UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

TESLA, INC.,
Petitioner,

v.

AUTONOMOUS DEVICES, LLC
Patent Owner

U.S. PATENT NOS. 10,607,134 and 11,113,585

DECLARATION OF JASON JANET

TABLE OF CONTENTS

I. Introduction....1

II. Summary of Opinions....3

III. Qualifications and Compensation....4

IV. Relevant Legal Standards....11

A. Claim Construction....12

B. Invalidity....14

C. Person of Ordinary Skill in the Art....17

V. Background of the Technology....18

VI. The Prior Art....21

A. U.S. No. 8,142,268 ("Thomas")....21

B. U.S. No. 9,434,072 ("Buehler")....30

C. U.S. 9,008,840 ("Ponulak")....33

VII. Claim Construction....34

1. "means for processing" ('585 Patent at claim 20)....35
2. "means for storing" ('585 at claim 20)....38
3. "means for generating or receiving…"; "means for determining …"; "means for executing…" ('134 Patent at claim 19 and claim 20)....40

VIII. Prosecution History....42

A. U.S. No. 11,113,585 ("the '585 Patent")....42

B. U.S. No. 10,607,134 (the '134 Patent)....44

IX. The '585 Patent....44

A. The Claims....44

Tesla Exhibit 1003.002

B. 585 Patent Grounds 1 and 2: Anticipation and/or Obviousness of Claims 1-7 and 10-20 based on Thomas.....58

1. Claim 1.....58

a. [1.1] A system comprising: one or more processors; and.....60

b. [1.2] one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program,.....61

c. [1.3] wherein the one or more processors are configured to perform at least: receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program; 68

d. [1.4] determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures; and.....72

e. [1.5] at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.....74

2. Claim 2.....75

a. [2.1] The system of claim 1, wherein the memory further stores at least a second one or more digital pictures correlated with a second one or more instruction sets for operating the first object of the first application program, ...75

b. [2.2] and wherein at least a portion of the first one or more digital pictures or at least a portion of the

first one or more instruction sets for operating the first object of the first application program is learned in a first learning process that includes operating the first object of the first application program at least partially by a first user, ... 76

c. [2.3] and wherein at least a portion of the second one or more digital pictures or at least a portion of the second one or more instruction sets for operating the first object of the first application program is learned in a second learning process that includes operating the first object of the first application program at least partially by the first user, ... 78

d. [2.4] and wherein the first user is: a human user, or a non-human user. 80

3. Claim 3 81

a. [3.1] The system of claim 1, wherein the memory further stores at least a second one or more digital pictures correlated with a second one or more instruction sets for operating the first object of the first application program, and 81

b. [3.2] wherein at least a portion of the first one or more digital pictures or at least a portion of the first one or more instruction sets for operating the first object of the first application program is learned in a first learning process that includes operating the first object of the first application program at least partially by a first user, and 81

c. [3.3] wherein at least a portion of the second one or more digital pictures or at least a portion of the second one or more instruction sets for operating the first object of the first application program is learned in a second learning process that includes operating the first object of the first application program at least partially by a second user, and 82

d. [3.4] wherein the first user is: a human user, or a non-human user, and wherein the second user is: a human user, or a non-human user. 83

4. Claim 4 83

a. [4.1] The system of claim 1, wherein the memory further stores at least a second one or more digital pictures correlated with a second one or more instruction sets for operating a third object of the first application program or a second object of the second application program, and 83
b. [4.2] wherein at least a portion of the first one or more digital pictures or at least a portion of the first one or more instruction sets for operating the first object of the first application program is learned in a first learning process that includes operating the first object of the first application program at least partially by a first user, and 85
c. [4.3] wherein at least a portion of the second one or more digital pictures or at least a portion of the second one or more instruction sets for operating the third object of the first application program or the second object of the second application program is learned in a second learning process that includes operating the third object of the first application program or the second object of the second application program at least partially by: the first user, or a second user, and 86
d. [4.4] wherein the first user is: a human user, or a non-human user, and wherein the second user is: a human user, or a non-human user..... 87
5. Claim 5 88
a. [5.1] The system of claim 1, wherein the new one or more digital pictures depict the at least the portion of the surrounding of the first object of the first application program, and..... 88
b. [5.2] wherein the first one or more instruction sets for operating the first object of the first application program are applied to the first object of the first application program, and wherein the first object of the first application program autonomously performs the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program. 88
6. Claim 6 89

Tesla Exhibit 1003.005

a. [6.1] The system of claim 1, wherein the new one or more digital pictures depict the at least the portion of the surrounding of the second object of the first application program, and.....89
b. [6.2] wherein the first one or more instruction sets for operating the first object of the first application program are applied to the second object of the first application program, and.....89
c. [6.3] wherein the second object of the first application program autonomously performs the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program. 92
7. Claim 7.....93
a. [7.1] The system of claim 1, wherein the new one or more digital pictures depict the at least the portion of the surrounding of the first object of the second application program, and.....93
b. [7.2] wherein the first one or more instruction sets for operating the first object of the first application program are applied to the first object of the second application program, and.....95
c. [7.3] wherein the first object of the second application program autonomously performs the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program. 97
8. Claim 10.....97
a. [10.1] The system of claim 1, wherein the new one or more digital pictures depict a third-person view of the at least the portion of the surrounding of: the first object of the first application program, the second object of the first application program, or the first object of the second application program, and.....97
[10.2] wherein the third-person view includes a view from a perspective of an observer of: the first object of the first application program, the second object of the first application program, or the first object of the second application program.....102

Tesla Exhibit 1003.006

9. Claim 11: The system of claim 1, wherein the new one or more digital pictures depict a top-down view of the at least the portion of the surrounding of: the first object of the first application program, the second object of the first application program, or the first object of the second application program, and wherein the top-down view includes a view from above of: the first object of the first application program, the second object of the first application program, or the first object of the second application program. 103
10. Claim 12 106
a. [12.1] The system of claim 1, wherein the one or more processors are further configured to perform at least: modifying: the first one or more instruction sets for operating the first object of the first application program, or a copy of the first one or more instruction sets for operating the first object of the first application program, and.....106
b. [12.2] wherein the executing the first one or more instruction sets for operating the first object of the first application program includes executing: the modified the first one or more instruction sets for operating the first object of the first application program, or the modified the copy of the first one or more instruction sets for operating the first object of the first application program, and.....109
c. [12.3] wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performing the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program includes the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performing one or more operations defined by: the modified the first one or more instruction sets for operating the first object of the first application program, or the

modified the copy of the first one or more instruction sets for operating the first object of the first application program. 109
11. Claim 13: The system of claim 1, wherein the new one or more digital pictures are generated at least by: generating one or more representations of one or more objects in the at least the portion of the surrounding of: the first object of the first application program, the second object of the first application program, or the first object of the second application program; and generating the new one or more digital pictures using the generated one or more representations of the one or more objects, and wherein the new one or more digital pictures depict: the generated one or more representations of the one or more objects, or a modified the generated one or more representations of the one or more objects.....110
12. Claim 14: The system of claim 1, wherein the first one or more instruction sets for operating the first object of the first application program include one or more information about one or more states of: the first object of the first application program, or a portion of the first object of the first application program.....112
13. Claim 15: The system of claim 1, wherein the first one or more digital pictures correlated with the first one or more instruction sets for operating the first object of the first application program include the first one or more digital pictures connected with the first one or more instruction sets for operating the first object of the first application program using at least one or more connections. 114
14. Claim 16.....117
- **a.** [16.1] The system of claim 1, wherein the system further comprising: a neural network that includes the first one or more digital pictures connected with the first one or more instruction sets for operating the first object of the first application program using at least one of: one or more connections, or one or more nodes, and 117
- **b.** [16.2] wherein the determining the first one or more instruction sets for operating the first object of the first application program based on the at

Tesla Exhibit 1003.008

least partial match between the new one or more digital pictures and the first one or more digital pictures includes applying the new one or more digital pictures to the neural network. 120

15. Claim 17 122

a. [17.1] The system of claim 1, wherein at least some elements of the system are included in: a single device, or multiple devices, and 122

b. [17.2] wherein the one or more processors include: one or more microcontrollers, one or more computing devices, or one or more electronic devices, and 122

c. [17.3] wherein the one or more memories include: one or more non-transitory machine readable media, one or more volatile memories, one or more non-volatile memories, one or more storage devices, or one or more storage systems, and 123

d. [17.4] wherein the first one or more digital pictures include: one or more still digital pictures, or one or more motion digital pictures, and 124

e. [17.5] wherein the new one or more digital pictures include: one or more still digital pictures, or one or more motion digital pictures, and 125

f. [17.6] wherein the first application program includes: a video game, a computer game, a simulation program, a program including text processing, a program including number processing, a program including picture processing, a program including object processing, or a program, and 125

g. [17.7] wherein the second application program includes: a video game, a computer game, a simulation program, a program including text processing, a program including number processing, a program including picture processing, a program including object processing, or a program, and 126

h. [17.8] wherein an instruction set of the first one or more instruction sets for operating the first object of the first application program includes at least

Tesla Exhibit 1003.009

one of: only one instruction, a plurality of instructions, one or more inputs, one or more commands … or one or more information. 127
16. Claim 18 128
a. [18.1] The system of claim 1, wherein the first one or more digital pictures are: one or more whole digital pictures, one or more representations of one or more whole digital pictures, one or more portions of at least one digital picture, one or more representations of one or more portions of at least one digital picture, one or more features, one or more representations of one or more features, one or more collections of pixels, or one or more collections of values, and....128
b. [18.2] wherein the new one or more digital pictures are: one or more whole digital pictures, one or more representations of one or more whole digital pictures, one or more portions of at least one digital picture, one or more representations of one or more portions of at least one digital picture, one or more features, one or more representations of one or more features, one or more collections of pixels, or one or more collections of values. 129
17. Claim 19 129
18. Claim 20 131
a. [20.1] A system comprising: means for processing; and....131
b. [20.2] means for storing machine readable code that, when executed by the means for processing, causes the means for processing to perform at least: 131
c. [20.3] accessing one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program;....132
d. [20.4] receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first

Tesla Exhibit 1003.010

application program, or a first object of a second application program;.....133
e. [20.5] determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures; and.....133
f. [20.6] at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.....134

C. Grounds 3 and 4: Anticipation of Claims 1, 8-9, 12, and 16-20 Based on Buehler; Obviousness of Claim 17 Based on Buehler.....134

1. Claim 1.....135
a. [1.1] A system comprising: one or more processors; and.....135
b. [1.2] one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program,.....135
c. [1.3] wherein the one or more processors are configured to perform at least: receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program;.....138
d. [1.4] determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures; and.....139

Tesla Exhibit 1003.011

e. [1.5] at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program. 141

2. Claim 8 142

a. [8.1] The system of claim 1, wherein the one or more processors are further configured to perform at least: 142

b. [8.2] modifying at least one of: the first one or more digital pictures, a copy of the first one or more digital pictures, the new one or more digital pictures, or a copy of the new one or more digital pictures, and 142

c. [8.3] wherein the at least partial match between the new one or more digital pictures and the first one or more digital pictures is determined at least by … (iii) determining at least partial match between the new one or more digital pictures and the modified the first one or more digital pictures, [or]…. 144

3. Claim 9 145

a. [9.1] The system of claim 1, wherein the new one or more digital pictures depict a first-person view of the at least the portion of the surrounding of: the first object of the first application program, the second object of the first application program, or the first object of the second application program, ... 145

b. [9.2] and wherein the first-person view includes a view from a perspective of: the first object of the first application program, the second object of the first application program, or the first object of the second application program. 147

4. Claim 12 147

Tesla Exhibit 1003.012

a. [12.1] The system of claim 1, wherein the one or more processors are further configured to perform at least: modifying: the first one or more instruction sets for operating the first object of the first application program, or a copy of the first one or more instruction sets for operating the first object of the first application program, and.....147
b. [12.2] wherein the executing the first one or more instruction sets for operating the first object of the first application program includes executing: the modified the first one or more instruction sets for operating the first object of the first application program, or the modified the copy of the first one or more instruction sets for operating the first object of the first application program, and.....149
c. [12.3] wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performing the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program includes the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performing one or more operations defined by: the modified the first one or more instruction sets for operating the first object of the first application program, or the modified the copy of the first one or more instruction sets for operating the first object of the first application program. 150
5. Claim 16.....151
a. [16.1] The system of claim 1, wherein the system further comprising: a neural network that includes the first one or more digital pictures connected with the first one or more instruction sets for operating the first object of the first application program using at least one of: one or more connections, or one or more nodes, and 151

Tesla Exhibit 1003.013

b. [16.2] wherein the determining the first one or more instruction sets for operating the first object of the first application program based on the at least partial match between the new one or more digital pictures and the first one or more digital pictures includes applying the new one or more digital pictures to the neural network. 155

6. Claim 17.....156

a. [17.1] The system of claim 1, wherein at least some elements of the system are included in: a single device, or multiple devices, and.....156

b. [17.2] wherein the one or more processors include: one or more microcontrollers, one or more computing devices, or one or more electronic devices, and.....157

c. [17.3] wherein the one or more memories include: one or more non-transitory machine readable media, one or more volatile memories, one or more non-volatile memories, one or more storage devices, or one or more storage systems, and 157

d. [17.4] wherein the first one or more digital pictures include: one or more still digital pictures, or one or more motion digital pictures, and.....158

e. [17.5] wherein the new one or more digital pictures include: one or more still digital pictures, or one or more motion digital pictures, and 158

f. [17.6] wherein the first application program includes: a video game, a computer game, a simulation program, a program including text processing, a program including number processing, a program including picture processing, a program including object processing, or a program, and.....159

g. [17.7] wherein the second application program includes: a video game, a computer game, a simulation program, a program including text processing, a program including number processing, a program including picture processing, a program including object processing, or a program, and.....159

Tesla Exhibit 1003.014

h. [17.8] wherein an instruction set of the first one or more instruction sets for operating the first object of the first application program includes at least one of: only one instruction, a plurality of instructions, one or more inputs, one or more commands…or one or more information. 160

7. Claim 18 161

a. [18.1] The system of claim 1, wherein the first one or more digital pictures are: one or more whole digital pictures, one or more representations of one or more whole digital pictures, one or more portions of at least one digital picture, one or more representations of one or more portions of at least one digital picture, one or more features, one or more representations of one or more features, one or more collections of pixels, or one or more collections of values, and.....161

b. [18.2] wherein the new one or more digital pictures are: one or more whole digital pictures, one or more representations of one or more whole digital pictures, one or more portions of at least one digital picture, one or more representations of one or more portions of at least one digital picture, one or more features, one or more representations of one or more features, one or more collections of pixels, or one or more collections of values. 162

8. Claim 19 162

9. Claim 20 164

a. [20.1] A system comprising: means for processing; and.....164

b. [20.2] means for storing machine readable code that, when executed by the means for processing, causes the means for processing to perform at least: 164

c. [20.3] accessing one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program;.....165

Tesla Exhibit 1003.015

d. [20.4] receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program;.....165
e. [20.5] determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures; and.....165
f. [20.6] at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.....166

D. Ground 5: Obviousness of Claim 16 based on Buehler in view of Ponulak.....166

1. Claim 16.....171
a. [16.1] The system of claim 1, wherein the system further comprising: a neural network that includes the first one or more digital pictures connected with the first one or more instruction sets for operating the first object of the first application program using at least one of: one or more connections, or one or more nodes, and 171
b. [16.2] wherein the determining the first one or more instruction sets for operating the first object of the first application program based on the at least partial match between the new one or more digital pictures and the first one or more digital pictures includes applying the new one or more digital pictures to the neural network. 174

Tesla Exhibit 1003.016

X. The ’134 Patent 175

A. The Claims 175

B. Grounds 1 and 2: Anticipation and/or Obviousness of Claims 1-20 based on Thomas 187

1. Claim 1 187

a. [1.1] A system comprising: one or more processors; and 189

b. [1.2] one or more non-transitory machine readable media storing machine readable code that, when executed by the one or more processors, causes the one or more processors to perform at least: 189

c. [1.3] accessing a first correlation including a first one or more object representations correlated with a first one or more instruction sets for operating a first avatar of an application, wherein the first one or more object representations represent one or more objects of the application; 190

d. [1.4] generating or receiving a second one or more object representations, wherein the second one or more object representations represent one or more objects of the application; 196

e. [1.5] determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second one or more object representations and the first one or more object representations; and 199

f. [1.6] at least in response to the determining, causing the first avatar of the application or a second avatar of the application to perform one or more operations defined by the first one or more instruction sets for operating the first avatar of the application at least by executing the first one or more instruction sets for operating the first avatar of the application. 201

2. Claim 2 203

a. [2.1] The system of claim 1, wherein the first one or more object representations include: one object representation, a stream of object representations, a

Tesla Exhibit 1003.017

collection of object representations, or a stream of collections of object representations, ...203
b. [2.2] and wherein the second one or more object representations include: one object representation, a stream of object representations, a collection of object representations, or a stream of collections of object representations, ...204
c. [2.3] and wherein the one or more objects of the application represented by the first one or more object representations are detected at a first time or during a first time period, and wherein the one or more objects of the application represented by the second one or more object representations are detected at a second time or during a second time period. 205
3. Claim 3 206
a. [3.1] The system of claim 1, wherein the determining the first one or more instruction sets for operating the first avatar of the application based on the at least partial match between the second one or more object representations and the first one or more object representations includes: 206
b. [3.2] determining that a number of at least partially matching portions of the second one or more object representations and portions of the first one or more object representations exceeds a threshold number or determining that a percentage of at least partially matching portions of the second one or more object representations and portions of the first one or more object representations exceeds a threshold percentage. 206
4. Claim 4 208
a. [4.1] The system of claim 1, wherein the first correlation is included in a knowledgebase, ...208
b. [4.2] and wherein the knowledgebase further includes a third correlation including a third one or more object representations correlated with a third one or more instruction sets for operating: the first avatar of the application, the second avatar of the

Tesla Exhibit 1003.018

application, a third avatar of the application, or an avatar of another application, and.....209
c. [4.3] wherein the third one or more object representations represent one or more objects of: the application, or the another application, and.....211
d. [4.4] wherein the first correlation is connected with the third correlation by a connection. 212
5. Claim 5: The system of claim 1, wherein the second avatar of the application performs the one or more operations defined by the first one or more instruction sets for operating the first avatar of the application. 214
6. Claim 6:.....217
a. [6.1] The system of claim 1, wherein the one or more objects of the application represented by the first one or more object representations include one or more objects of the application in the first avatar's surrounding, ...217
b. [6.2] and wherein the one or more objects of the application represented by the second one or more object representations include: one or more objects of the application in the first avatar's surrounding, or one or more objects of the application in the second avatar's surrounding, and 218
c. [6.3] wherein the first avatar's surrounding includes at least one of: a part of the application in an area of interest around the first avatar, a part of the application defined by a threshold distance from the first avatar, a part of the application relative to the first avatar, a part of the application around the first avatar, a part of the application that is shown to a user, a part of the application that is visible to a user, or a part of the application,.....220
7. Claim 7: The system of claim 1, wherein, to correlate the first one or more object representations with the first one or more instruction sets for operating the first avatar of the application, a determination is made that the first one or more instruction sets for operating the first avatar of the application temporally correspond to the first one or more object representations. 222
8. Claim 8.....225

Tesla Exhibit 1003.019

a. [8.1] The system of claim 1, wherein at least some elements of the system are included in: a single device, or multiple devices, ...225
b. [8.2] and wherein the one or more processors include: one or more microcontrollers, one or more computing devices, or one or more electronic devices, ...225
c. [8.3] and wherein the first correlation is stored in or on at least one of: at least one non-transitory machine readable medium of the one or more non-transitory machine readable media, another one or more non-transitory machine readable media, one or more volatile memories, one or more non-volatile memories, one or more storage devices, or one or more storage systems,.....226
d. [8.4] and wherein the application includes: a video game, a computer game, a simulation program, a program including text processing, a program including number processing, a program including picture processing, a program including object processing, or a program, ...226
e. [8.5] and wherein the application includes: one or more versions of the application, one or more upgrades of the application, one or more sequels of the application, one or more instances of the application, or one or more variations of the application,.....227
f. [8.6] and wherein the first avatar of the application is a first object of the application and the second avatar of the application is a second object of the application,.....228
g. [8.7] and wherein the first one or more object representations include one or more properties of the one or more objects represented by the first one or more object representations, and wherein the second one or more object representations include one or more properties of the one or more objects represented by the second one or more object representations, and 228

Tesla Exhibit 1003.020

h. [8.8] wherein an instruction set of the first one or more instruction sets for operating the first avatar of the application includes at least one of: only one instruction, a plurality of instructions, one or more inputs, one or more commands, one or more computer commands, one or more keywords, one or more symbols, one or more operators, one or more variables, one or more values, one or more objects, one or more object references, one or more data structures, one or more data structure references, one or more functions, one or more function references, one or more parameters, one or more signals, one or more characters, one or more digits, one or more numbers, one or more user operating directions, one or more user directions, one or more user inputs, one or more representations of one or more user actions, one or more representations of one or more user clicks, one or more binary bits, one or more assembly language commands, one or more states, one or more state representations, one or more codes, one or more data, or one or more information, and 229

i. [8.9] wherein: an object of the application of the one or more objects of the application represented by the first one or more object representations is the same as an object of the application of the one or more objects of the application represented by the second one or more object representations, multiple objects of the application of the one or more objects of the application represented by the first one or more object representations are the same as multiple objects of the application of the one or more objects of the application represented by the second one or more object representations, all objects of the application of the one or more objects of the application represented by the first one or more object representations are the same as all objects of the application of the one or more objects of the application represented by the second one or more object representations, or all

Tesla Exhibit 1003.021

objects of the application of the one or more objects of the application represented by the first one or more object representations are different than all objects of the application of the one or more objects of the application represented by the second one or more object representations.....230
9. Claim 9: The system of claim 1, wherein the first correlation is included in a first knowledge cell, and wherein the first knowledge cell is a data structure for storing, structuring, or organizing the first correlation.....233
10. Claim 10.....235
11. Claim 11.....236
a. [11.1] The system of claim 1, wherein the first correlation is included in a knowledgebase,.....236
b. [11.2] and wherein the knowledgebase further includes a third correlation including a third one or more object representations correlated with a third one or more instruction sets for operating the first avatar of the application, ...236
c. [11.3] and wherein the third one or more object representations represent one or more objects of the application, and.....237
d. [11.4] wherein at least a portion of the first correlation is learned in a learning process that includes operating the first avatar of the application at least partially by a user, and wherein at least a portion of the third correlation is learned in another learning process that includes operating the first avatar of the application at least partially by the user, and.....237
e. [11.5] wherein the user is: a human user, or a non-human user.....240
12. Claim 12.....241
a. [12.1] The system of claim 1, wherein the first correlation is included in a knowledgebase, and wherein the knowledgebase further includes a third correlation including a third one or more object representations correlated with a third one or more instruction sets for operating the first avatar of the application, and wherein the third one or more

Tesla Exhibit 1003.022

object representations represent one or more objects of the application, and 241
b. [12.2] wherein at least a portion of the first correlation is learned in a learning process that includes operating the first avatar of the application at least partially by a user, and..... 241
c. [12.3] wherein at least a portion of the third correlation is learned in another learning process that includes operating the first avatar of the application at least partially by another user, ... 241
d. [12.4] and wherein the user is: a human user, or a non-human user, and wherein the another user is: a human user, or a non-human user..... 243
13. Claim 13 244
a. [13.1] The system of claim 1, wherein the first correlation is included in a knowledgebase,..... 244
b. [13.2] and wherein the knowledgebase further includes a third correlation including a third one or more object representations correlated with a third one or more instruction sets for operating a third avatar of the application, and..... 244
c. [13.3] wherein the third one or more object representations represent one or more objects of the application,..... 246
d. [13.4] and wherein at least a portion of the first correlation is learned in a learning process that includes operating the first avatar of the application at least partially by a user, and..... 246
e. [13.5] wherein at least a portion of the third correlation is learned in another learning process that includes operating the third avatar of the application at least partially by: the user, or another user, and wherein the user is: a human user, or a non-human user, and wherein the another user is: a human user, or a non-human user. 246
14. Claim 14 247
a. [14.1] The system of claim 1, wherein the first correlation is included in a knowledgebase, and 247
b. [14.2] wherein the knowledgebase further includes a third correlation including a third one or more

Tesla Exhibit 1003.023

object representations correlated with a third one or more instruction sets for operating an avatar of another application, and.....247
c. [14.3] wherein the third one or more object representations represent one or more objects of the another application.....249
15. Claim 15: The system of claim 1, wherein the first one or more instruction sets for operating the first avatar of the application include one or more information about one or more states of: the first avatar of the application or a portion of the first avatar of the application. 250
16. Claim 16: The method system of claim 1, wherein: an element of the first correlation is deleted after the first correlation is generated, an element of the first correlation is modified after the first correlation is generated, an element of the first correlation is manipulated after the first correlation is generated, or an element is inserted into the first correlation after the first correlation is generated. 252
17. Claim 17 255
a. [17.1] The system of claim 1, wherein the machine readable code, when executed by the one or more processors, causes the one or more processors to further perform at least: modifying: the first one or more instruction sets for operating the first avatar of the application, or a copy of the first one or more instruction sets for operating the first avatar of the application, and 255
b. [17.2] wherein the executing the first one or more instruction sets for operating the first avatar of the application includes executing: the modified the first one or more instruction sets for operating the first avatar of the application, or the modified the copy of the first one or more instruction sets for operating the first avatar of the application, and 257
c. [17.3] wherein the performing, by the first avatar of the application or by the second avatar of the application, the one or more operations defined by the first one or more instruction sets for operating the first avatar of the application includes

Tesla Exhibit 1003.024

performing, by the first avatar of the application or by the second avatar of the application, one or more operations defined by: the modified the first one or more instruction sets for operating the first avatar of the application, or the modified the copy of the first one or more instruction sets for operating the first avatar of the application. 258

18. Claim 18 259

a. [18.1] The system of claim 1, wherein at least a portion of the first correlation is learned in a learning process that includes: operating the first avatar of the application at least partially by: a human user, or a non-human user: 259

b. [18.2] generating or receiving the first one or more object representations; 260

c. [18.3] and obtaining or receiving the first one or more instruction sets for operating the first avatar of the application. 260

19. Claim 19 260

a. [19.1] A system comprising: a memory that stores at least a first correlation including a first one or more object representations correlated with a first one or more instruction sets for operating a first avatar of an application, wherein the first one or more object representations represent one or more objects of the application; 260

b. [19.2] means for generating or receiving a second one or more object representations, wherein the second one or more object representations represent one or more objects of the application; 261

c. [19.3] means for determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second one or more object representations and the first one or more object representations; and 262

d. [19.4] means for executing, at least in response to the determining, the first one or more instruction sets for operating the first avatar of the application, wherein the first avatar of the application or a second avatar of the application performs one or

Tesla Exhibit 1003.025

more operations defined by the first one or more instruction sets for operating the first avatar of the application.....262

20. Claim 20: The system of claim 19, wherein the means for generating or receiving the second one or more object representations includes one or more processors, and wherein the means for determining the first one or more instruction sets for operating the first avatar of the application based on the at least partial match between the second one or more object representations and the first one or more object representations includes one or more processors, and wherein the means for executing, at least in response to the determining, the first one or more instruction sets for operating the first avatar of the application includes one or more processors. 263

C. Ground 2: Anticipation of Claims 1, 3, and 17 Based on Buehler 264

1. Claim 1 264

a. [1.1] A system comprising: one or more processors; and: 264

b. [1.2] and one or more non-transitory machine readable media storing machine readable code that, when executed by the one or more processors, causes the one or more processors to perform at least: 265

c. [1.3] accessing a first correlation including a first one or more object representations correlated with a first one or more instruction sets for operating a first avatar of an application, wherein the first one or more object representations represent one or more objects of the application;.....266

d. [1.4] generating or receiving a second one or more object representations, wherein the second one or more object representations represent one or more objects of the application; 268

e. [1.5] determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second

Tesla Exhibit 1003.026

one or more object representations and the first one or more object representations; and.....269
f. [1.6] at least in response to the determining, causing the first avatar of the application or a second avatar of the application to perform one or more operations defined by the first one or more instruction sets for operating the first avatar of the application at least by executing the first one or more instruction sets for operating the first avatar of the application. 272
2. Claim 3 274
a. [3.1] The system of claim 1, wherein the determining the first one or more instruction sets for operating the first avatar of the application based on the at least partial match between the second one or more object representations and the first one or more object representations includes:.....274
b. [3.2] determining that a number of at least partially matching portions of the second one or more object representations and portions of the first one or more object representations exceeds a threshold number or determining that a percentage of at least partially matching portions of the second one or more object representations and portions of the first one or more object representations exceeds a threshold percentage. 274

XI. Declaration.....280

Tesla Exhibit 1003.027

I. Introduction

1. I have been retained by Petitioner Tesla, Inc. ("Petitioner" or "Tesla"), as an independent expert witness in the above-captioned *inter partes* reviews ("IPRs"), in which Tesla has requested that the U.S. Patent and Trademark Office cancel as unpatentable all claims of U.S. Patent Nos. 10,607,134 ("the '134 Patent") and 11,113,585 ("the '585 Patent") (together, "the Simulation Patents").

2. This declaration sets forth my analyses and opinions based on my knowledge, experience, and the materials I have considered. As I explain below, it is my opinion that all claims of the '134 Patent and '585 Patent are directed to subject matter that was routine, conventional, and well known in the art before the patents' priority dates. As would be readily appreciated by one of skill in the art, the '134 Patent and '585 Patent claims are anticipated by the prior art references cited herein, and/or rendered obvious by the prior art in view of the knowledge of a person of ordinary skill in the art or the combination of prior art references described below.

3. I have reviewed the documents referenced in this declaration. I understand they have been submitted as exhibits in conjunction with Tesla's Petitions for IPR as set forth in the table below:

Exhibit No.	Description
Ex. 1001	U.S. Patent No. 10,607,134
Ex. 1001	U.S. Patent No. 11,113,585
Ex. 1002	Prosecution File History for U.S. Patent No. 10,607,134
Ex. 1002	Prosecution File History for U.S. Patent No. 11,113,585
Ex. 1005	U.S. Patent No. 8,142,268 ("Thomas")

Tesla Exhibit 1003.028

Exhibit No.	Description
Ex. 1006	U.S. Patent No. 9,434,072 ("Buehler")
Ex. 1007	U.S. Patent No. 9,008,840 ("Ponulak")
Ex. 1008	U.S. Patent App. Pub. No. 2003/0171146 A1
Ex. 1009	U.S. Patent App. Pub. No. 2003/0032467 A1
Ex. 1010	U.S. Patent No. 9,195,934
Ex. 1011	Silva, Marty, *A Visual History of American Football Games*, IGN, July 8, 2013, available at, https://www.ign.com/articles/2013/07/08/a-visual-history-of-football-video-games
Ex. 1012	Macy, Seth G., *A Visual History of Racing Games*, IGN, updated Sept. 6, 2017, available at, https://www.ign.com/articles/2015/03/27/a-visual-history-of-racing-games
Ex. 1013	*Final Fantasy XI Released*, IGN, updated June 16, 2012, available at, https://www.ign.com/articles/2003/10/29/final-fantasy-xi-released
Ex. 1014	Ellis, Tim, et al., *Ellipse Detection and Matching with Uncertainty*, pp. 137-144
Ex. 1015	Pomerleau, Dean A., *Neural Network Perception for Mobile Robot Guidance*, Feb. 16, 1992
Ex. 1016	Lee, Ten-min, et al., *Computer Simulation of Learning Experiments with Autonomous Mobile Robots* (Univ. of Manchester, March 26, 1999)
Ex. 1017	Durrant-Whyte, Hugh, et al., *Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms*
Ex..1018	Smith, Randall, et al., *Estimating Uncertain Spatial Relationships in Robotics,* pp. 167-193
Ex. 1019	Zimmer, Kyle, *Evolution of the Game Console*, *available at* citadel.sjfc.edu/students-virtual/k/knz03095/public_html/imd/museum/index.html
Ex. 1020	Stuart, Keith, *PS4 and Xbox One: so why aren't they backwards compatible?,* The Guardian, December 12, 2023, available at https://www.theguardian.com/technology/2013/dec/12/ps4-and-xbox-one-so-why-arent-they-backwards-compatible
Ex. 1021	Christian, Mark, *Saved Game History: Battery Backups, Memory Cards, and the Cloud,* Mailchimp, Feb. 21, 2019, available at https://tedium.co/2019/02/21/video-game-save-state-history/

II. Summary of Opinions

4. Based on my knowledge, experience, and the materials that I have reviewed, it is my opinion that claims 1-20 of the ’134 Patent and claims 1-20 of the ’585 Patent are anticipated and/or obvious. Specifically:

(i) Claims 1-20 of the ’134 Patent are anticipated by U.S. Patent No. 8,142,268 (“Thomas”) (Ex. 1005);

(ii) Claims 3, 5, and 12-14 of the ’134 Patent are obvious based on Thomas in view of the knowledge of a POSITA;

(iii) Claims 1, 3, and 17 of the ’134 Patent are anticipated by U.S. Patent No. 9,434,072 (“Buehler”) (Ex. 1006);

(iv) Claims 1-7 and 10-20 of the ’585 Patent are anticipated by Thomas;

(v) Claims 3-4, 6-7, and 10-12 of the ’585 Patent are obvious based on Thomas in view of the knowledge of a POSITA;

(vi) Claims 1, 8-9, 12, and 16-20 of the ’585 Patent are anticipated by Buehler;

(vii) Claim 17 of the ’585 Patent is obvious based on Buehler in view of the knowledge of a POSITA;

Tesla Exhibit 1003.030

(viii) Claim 16 of the ’134 Patent is obvious based on Buehler in view of U.S. Patent No. 9,008,840 (“Ponulak”) (Ex. 1007).

III. Qualifications and Compensation

5. I have a Ph.D. in Electrical & Computer Engineering from North Carolina State University, which I earned in 1998. Prior to that, I received a M.S. in Integrated Manufacturing Systems Engineering from North Carolina State University in 1994 and a B.S. in Mechanical Engineering from the University of Virginia in 1990.

6. Since 1991, I have been active in the robotics, sensors, artificial intelligence, pattern analysis, computer vision, image/video processing, and automation fields. I have authored numerous publications and have co-authored a textbook entitled *Computational Intelligence*. My Ph.D. dissertation is entitled *Pattern Analysis, Tracking and Control for Autonomous Vehicles Using Neural Networks*. It described the use of unsupervised training (i.e., the Hyper-Ellipsoid Clustering Neural Network, or "HECNN"), and supervised training (i.e., the Region-Feature Neural Network, or "RFNN") for dissimilar mobile robots to learn aspects of their common environments through multiple types of sensors (*e.g.*, acoustic rangefinding, infrared proximity, tactile and imaging), build knowledge libraries, and then transfer knowledge between for operational use and/or refinement. Among other outputs, the robots learned to recognize spaces, landmarks within those spaces,

Tesla Exhibit 1003.031

and then plan and track motion from the accumulated and shared knowledge. Efficiency and accuracy were shown to increase significantly with the combined unsupervised and supervised learning, and knowledge transfers.

7. I have designed, built, and marketed robots, automated systems, and components thereof, including ground mobile robots, unmanned aerial vehicles (UAV), unattended sensors automated storage and retrieval systems (ASRS), submersible mobile robots, proof-of-concept extra-terrestrial robots, manually manipulated ("twiddled") submersible foils, wall climbing mobile robots, and augmented reality systems.

8. I have taught the following courses: Introduction to Robotics and Automation (Duke and NCSU); Introduction to Control Theory (Duke and NCSU); Distributed Real-Time Controls (NCSU); and myriad independent studies courses in the areas of robotics, automation, artificial intelligence, autonomy, pattern analysis, and control systems. I have also served on several M.S.- and Ph.D.-level graduate student committees, designed qualifying exam problems, served on the NCSU IMSEI Board, and participated in curriculum development for undergraduate and graduate level programs.

9. I have also started, sponsored, and advised multiple champion student teams for international robot competitions including, but not limited to, the NASA/ASCE Extra-Terrestrial Robotics Competition, the DARPA Grand

Tesla Exhibit 1003.032

Challenge, the AUVSI/ONR Autonomous Underwater Vehicle Competition, and the European CLAWAR Wall-Climbing Robot competition.

10. I have initiated multiple unmanned aerial vehicle (UAV) projects at both academic and industry levels. Academic UAVs include, but are not limited to, Quadcopters with Hybrid Remote and Autonomous Control, Marsupial UAVs that Deploy and Recover Unmanned Ground Vehicles, and Wall-Climbing UAVs. Additionally, the AngelFish Cross-Domain Submersible UAV, a DARPA ASW program that I incubated and procured funding for at Teledyne, included a partnership with North Carolina State University.

11. In 1999, while I was employed by Nekton Research ("Nekton"), I captured and managed multiple programs sponsored by the Department of Defense (DoD) and private-sector companies that focused primarily on autonomous underwater vehicles (AUV), remotely operated vehicles (ROV), TwiddleFish toys, and indirect-fire projectiles. During my tenure, Nekton entered into a joint venture agreement with the founders of what eventually became known as Parata Systems – a pharmacy automation solutions company, which I supported launching.

12. After leaving Nekton in March 2002, I joined a then New Jersey-based company called Avionic Instruments, Inc ("Avionic"). While at Avionic, I continued developing and marketing robots and supported engineering related to various design, manufacturing, quality and assembly issues on the core aerospace product

Tesla Exhibit 1003.033

lines. Avionic product lines include, but are not limited to, ducted fans, transformer rectifier units (TRU), regulated TRU (RTRU), auxiliary power unit (APU) control systems, power distribution systems (PDU), frequency converters, corner clamps, and VRAM attractors/thrusters. Customers included DoD, NASA, Boeing, Sikorsky, Augusta-Westland, Dassault, and Lockheed-Martin.

13. In late 2004, soon after Avionic was sold, I, along with others from Avionic, secured funding to develop an alpha-level multi-dispenser robotic system. I wrote the RxMedic business plan and raised multiple rounds of venture capital. In late 2006, RxMedic (called "APDS" until November 2006) was launched as a stand-alone, sole-focus venture. After the operational launch of RxMedic, I served as General Manager and eventually Chief Technical Officer. Through my roles at RxMedic, I oversaw the development of the RxMedic ADS robot, managed the intellectual property portfolio, coordinated sales and marketing, and provided strategic, fiscal, and operational leadership. In May 2010, J.M Smith Corporation acquired RxMedic, and in an effort to assist in the change of ownership, I served as a director of RxMedic until May 2011.

14. Also in late 2004, after Avionic was sold, I, along with others from Avionic, secured funding for Vortex HC through DoD contracts and robot sales, to continue developing the VRAM Mobile Robot Platform (VMRP – a wall-climbing robot), the ARTEMIS AUV (a holonomic submersible robot for counter-mine and

Tesla Exhibit 1003.034

counter-obstacle operations), the submersible crawler, the nuclear-grade boiler water reactor (BWR) inspection robot, and other robot products centered around the VRAM. Some DoD programs were/are classified, for which I maintained a SECRET clearance at both the personal and facilities level, and served as the facilities security officer (FSO). In late 2006, corresponding to the full launch of RxMedic, Vortex HC technologies were largely licensed to Teledyne SeaBotix, SeaRobotics, and HDT.

15. However, I have continued to support Vortex HC licensees and customers to-date. After the sale of RxMedic, and after fulfilling my 12-month employment obligation, I joined Teledyne Technologies in Summer 2011. I served as the Senior Manager for the RTP division of Teledyne Scientific, and supported multiple DoD sponsored robotic-focused programs. Some of these programs were/are classified, for which I maintained a personal SECRET clearance. Among these programs, were cargo unmanned ground vehicles (CUGV), squad-level autonomous unmanned ground vehicles (UGV), autonomous underwater vehicles (AUV), unmanned underwater vehicles (UUV), and a cross-domain autonomous vehicle capable of transitioning between air-, surface- and underwater-domains. Cross-domain vehicles that were evaluated, designed, and prototyped included, but were not limited to, the AngelFish (later called "EagleRay") submersible AUV for anti-submarine warfare, and a ball-shaped robot for countermine operations on the ground, in the beach zone and in the surf zone. Sensor design, refinement, and signal

Tesla Exhibit 1003.035

processing was a major component of each program. Sensors employed include, but are not limited to: proximity sensors; ranging sensors; electro-optical imaging; long-, short-, and midwave infrared (IR); inertial measurement units (IMU); optical flow; and radiofrequency (RF). Additionally, control systems were designed, refined, and integrated into the aforementioned systems. Most control systems were closed-loop, in that they utilized sensor-based feedback; others were open-loop, where states were estimated with little or no feedback.

16. In late 2013, Avionic, a Transdigm business unit at that point, requested that I return to turn around a supply-chain issue, and assume management of its Sikorsky S97 Raider helicopter program. The S97 Raider is purported to be the fastest, most maneuverable helicopter, due to its coaxial, counter-rotating variable-pitch wings, and an aft-based push-propeller. Avionic also controlled two business units named Acme Aerospace (Acme) and Aerospace Cooling Solutions (ACS). Avionic, Acme and ACS supported the S97 program, which has met milestones and continues to produce multiple successful demonstrations. In 2013, Transdigm expanded my role to include directorship of the Avionic, Acme, and ACS sales and marketing team, and to report operational and financial status at six-week intervals. Transdigm required that I move my family to New Jersey in late 2014, which influenced my decision to resign and assume the role of CEO at Delta Five Systems in Raleigh, NC.

Tesla Exhibit 1003.036

17. As CEO of Delta Five, I led a private-equity backed, hospitality-focused venture, coordinating the company's launch and strategic path including an early stage pivot. Initially focused on back-end robotics and automation, Delta Five shifted to address the rampant bed bug problem—a top priority for hoteliers—with a novel unattended sensor and trap, called the Telemetered Pest Monitoring System (TPMS), which is based on computer vision and internet of things (IoT), and has proven capable of scaling to other pests. In addition to the TPMS, Delta Five developed a novel means to mass produce a natural, unscented aggregation pheromone that, in concert with placement and heat, lured invertebrates to the TPMS. After leading Delta Five and serving on its Board for four years, I resigned to join ARA, but continue serving Delta Five as an investor and supportive transition agent.

18. In 2018, I joined ARA as sector CTO. In this role, my primary responsibilities included leading the commercialization and technology transition efforts, which require technical, business, and corporate development, as well as the raising of capital and managing matrixed rosters. I assumed my other current role at ARA, Corporate Director of IP and Product Commercialization, in 2020. In this role, I evaluate, quantify, present, and recommend portfolios of IP, product lines, and ventures on the basis of competitive landscapes, addressable markets, sales strategy,

Tesla Exhibit 1003.037

resource availability and technology readiness. I also spearhead investments and provide general venture oversight.

19. My curriculum vitae is attached as Appendix 1 to my report, and provides further information about my experience, expertise, and presentations.

20. Through my professional experience, I have gained extensive expertise in robotics, sensors, artificial intelligence, automation, computer vision, and image/video processing.

21. I am being compensated at my standard rate of $500/hour. My compensation is in no way contingent upon my opinions or the outcome of the proceeding.

IV. Relevant Legal Standards

22. I am not an attorney, and therefore my understanding of patent law and the legal standards set forth in this report is based on explanations provided to me by counsel.

23. I understand that for any claim of a patent to claim priority to an earlier application (*i.e.*, to benefit from the earlier application's filing date), the claims of the later patent must be fully supported by the disclosure of the earlier patent application to which priority is claimed. I understand that in order for the claims to be supported, the earlier application's disclosure must be sufficient to

Tesla Exhibit 1003.038

allow a person of ordinary skill in the art to reasonably conclude that the inventors were in possession of the claimed invention.

24. I understand that the ’134 Patent was filed on December 19, 2016 and does not claim priority to any other application, and the ’585 Patent was filed on January 1, 2019 and claims priority to a series of continuation applications dating back to August 23, 2016 (App. No. 15/245,046). For the purposes of my opinions, I have considered the state of the art as of and shortly before these respective priority dates (*i.e.*, December 19, 2016 and August 23, 2016), and the level of knowledge that a POSITA would have possessed at that time. Unless I state otherwise, whenever I refer to any principle or technical subject matter as having been known or understood, this is meant to denote the knowledge and understanding of a POSITA at or prior to August 23, 2016.[1]

A. Claim Construction

25. It is my further understanding that the numbered paragraphs at the end of the disclosure of a U.S. patent are the patent “claims” that define the metes and

[1] It is my opinion that there is no appreciable difference between the state of the art as of August 23, 2016, and as of December 19, 2016, as it relates to the subject matter claimed in the ’134 Patent and ’585 Patent.

Tesla Exhibit 1003.039

bounds of the alleged invention. I understand that the claims of the Simulation Patents are what is being challenged in the present IPR proceedings.

26. I have been informed that, in this proceeding, the Board must determine the scope of the claims by giving the claims their ordinary and customary meaning in light of the specification, as the claims would be interpreted by one of ordinary skill in the art.

27. I understand that patent claims generally include a "transitional" term or phrase, such as "consisting" or "comprising," which may connect the preamble of the claim to the body of the claim. I have been informed that if a claim uses the term "consisting" as a transition term, that means that the claim is a "closed" claim, which means that the claim is limited to the claim features that follow the transition term and nothing else. On the other hand, I understand that the transition term "comprising" denotes an "open" claim, which means that the claim is not limited to only the features recited in the claim, and could encompass the listed elements as well as other unrecited elements.

28. I understand that 35 U.S.C. § 112 (f) created an exception to the general rule of claim construction called a "means plus function" limitation. These types of terms and limitations should be interpreted to cover only the corresponding structure described in the specification, and equivalents thereof. I also understand that a limitation is presumed to be a means plus function limitation if: (a) the claim

Tesla Exhibit 1003.040

limitation uses the phrase "means for"; (b) the "means for" is modified by functional language; and (c) the phrase "means for" is not modified by sufficient structure for achieving the specified function.

29. I understand that a structure is considered structurally equivalent to the corresponding structure identified in the specification only if the differences between them are insubstantial. For instance, if the structure performs the same function in substantially the same way to achieve substantially the same result. I further understand that a structural equivalent must have been available at the time of the issuance of the claim.

B. Invalidity

30. I understand that Tesla bears the burden of proving that the challenged claims of the Simulation Patents are invalid, and must prove this by a preponderance of the evidence, which means that invalidity must be shown to be more likely than not.

31. I have been asked to consider whether the claims of the Simulation Patents are anticipated by the prior art. I understand that for prior art to anticipate a claim, the reference must disclose every element of the claim.

32. I have also been asked to consider whether the claims of the Simulation Patents would have been obvious. I understand that this analysis must be conducted from the perspective of the person of ordinary skill in the art, and

Tesla Exhibit 1003.041

that I must evaluate whether the skilled artisan would consider any differences between the prior art and what is claimed to have been obvious. To make this assessment, I have been informed that the concept of patent obviousness involves four factual inquiries: (1) the scope and content of the prior art; (2) the differences between the claimed invention and the prior art; (3) the level of ordinary skill in the art; and (4) secondary considerations of non-obviousness. I have been instructed that one must not engage in hindsight. Rather, I understand that one should instead consider what the person of ordinary skill in the art would have reason to pursue further, and steps that were routinely done, such as in response to known problems, steps, or obstacles.

33. It is my understanding that the following is a non-exhaustive list of rationales that may support the obviousness of an invention: combining prior art elements according to known methods to yield predictable results; simple substitution of one known element for another to obtain predictable results; use of a known technique to improve a similar device (method, or product) in the same way; applying a known technique to a known device (method, or product) ready for improvement to yield predictable results; choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success; and some teaching, suggestion, or motivation in the prior art that would have led a

Tesla Exhibit 1003.042

POSITA to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

34. It is my understanding that the motivation to combine prior art references may be implicit and may be found in the knowledge of one of ordinary skill in the art, or in the nature of the problem to be solved. Specifically, it is my understanding that an implicit motivation to combine exists not only when a suggestion may be gleaned from the prior art as a whole, but when the "improvement" is technology-independent and the combination of references results in a product or process that is more desirable, for example because it is stronger, cheaper, cleaner, faster, lighter, smaller, more durable, or more efficient. It is my further understanding that the motivation to combine references may be found in the nature of the problem to be solved where prior art references are directed to precisely the same problem.

35. I also understand that prior art may be relied on for its express disclosure and teachings. I also understand that the prior art may be relied upon for a teaching of features that are necessarily present in the prior art reference even if that specific feature is not expressly or explicitly disclosed.

36. I understand that before reaching any final conclusion on obviousness, the obviousness analysis requires consideration of objective indicia of non-obviousness, if any such indicia are offered. These must be considered to ensure

that, for example, there were not some unanticipated problems, obstacles or hurdles that may seem easy to overcome in hindsight, but which were not readily overcome prior to the relevant invention date of the patents/claims at issue here. I understand that these objective indicia are also known as "secondary considerations of non-obviousness," and may include long-felt but unmet need and unexpected results, among others. I also understand, however, that any offered evidence of secondary considerations of non-obviousness must be commensurate with the scope of the challenged claims. This means that for any offered evidence of secondary considerations of non-obviousness to be given substantial weight, I understand the proponent of that evidence must establish a "nexus" or a sufficient connection or tie between that evidence and the merits of the claimed invention, which I understand specifically incorporates any novel element(s) of the claimed invention. If the secondary consideration evidence offered actually results from something other than the merits of the claim, then I understand that there is no nexus or tie to the claimed invention. I also understand it is the Patent Owner who has the burden of proving that a nexus exists, and I understand that secondary considerations will not overcome a strong showing of obviousness.

C. Person of Ordinary Skill in the Art

37. I have been asked to review the Simulation from the perspective of a person of ordinary skill in the art ("POSITA") as of the earliest claimed priority date

Tesla Exhibit 1003.044

for the patents— August 23, 2016 and December 19, 2016. I have been asked to evaluate the disclosure and claims of the Simulation Patents.

38. It is my opinion that a POSITA relevant to the Simulation Patents would have had at least: (1) a bachelor's degree in electrical engineering, computer engineering, computer science, systems engineering, or equivalent course work with two to three years of work experience in computer vision, image/video processing, and/or machine learning; or (2) a master's degree in electrical engineering, computer engineering, computer science, systems engineering, or equivalent course work with a focus in computer vision, image/video processing, and/or machine learning.

39. Based on my education, training and experience, it is my opinion that I can accurately represent the views of a POSITA as of the earliest claimed priority date of December 19, 2016, as to at least claims 1-20 of the '134 Patent, and as of August 23, 2016, as to at least claims 1-20 of the '585 Patent. The opinions I provide in this declaration are provided using the viewpoint of the POSITA as of these dates.

V. Background of the Technology

40. The asserted claims of the Simulation Patents are directed to the well-known practice of employing methods, devices, systems and/or applications that "generally relate to computing devices and/or systems … generally directed to devices, systems, methods, and/or applications for learning an avatar's or an application's operation in various circumstances, storing this knowledge in a

Tesla Exhibit 1003.045

knowledgebase (i.e. neural network, graph, sequences, etc.), and/or enabling autonomous operation of the avatar or the application." '134 Patent, Abstract. Devices' operations have been learned by intelligent systems, at least as far back as several decades. Teach pendants, for example, have been integral to robots dating back to at least the 1990's, where human operators would guide robots through various tasks based on sensory feedback, and the robots would learn to operate autonomously (i.e., without constant operator control) based on perceived circumstances. Simulations have also been used for the same purpose, also at least as far back as the 1990's. *See, e.g.,* Ex. 1016 (learning autonomous robots' interactions with their surroundings using Nomadic Technologies simulator and a neural network simulator), Ex. 1017 (discussing the history of using algorithms to solve robotic mapping and navigation problem), Ex. 1018 (discussing creating a map for a robot including through use of a simulator. "In the figures in this paper, the plotted points show the actual locations of objects, ***which are known only by the simulator*** and displayed for our benefit.").

41. Other types of supervised, unsupervised, and/or human-in-the-loop (HITL) training have been applied to mobile robots for decades, as well. For example, AAAI and IJCAI competitions from the mid 1990's demonstrated mobile robots that would learn to recognize objects and features within perceived environments and, based on supervised training, initiate one or more actions

Tesla Exhibit 1003.046

including avoidance, pursuit, and manipulation. These mobile robots utilized computer vision, whiskers, ultrasonic, and infrared sensors for perception. Many would also evaluate the scene(s) utilizing a variety of well-known sensor processing techniques such as color histograms, transforms, moments, codes, masks, and neural networks. Feedback from operator(s) in the loop might adjust, refine and optimize parameters (*e.g.*, neural network synaptic weights and biases), and then be stored as "knowledge." Multi-mobile robot systems, sometimes referred to as "multi-agent systems," from as far back as the 1990's were also shown to exchange knowledge and/or compile their respective observations to create a universal knowledge base that can be broadly used by any of a multitude of mobile robots. During subsequent operation, "knowledge" can be locked, deleted, added, and/or modified, depending on the demand for adaptability.

42. At its core, both supervised and unsupervised learning utilizes pattern analysis from well-known machine learning models, including statistics, Euclidean distances, Mahalanobis distances, moments, discriminators, masks, and the like. Unsupervised learning employs algorithms that learn patterns that were not previously "labeled" or "identified"; and learning features or other aspects about input data (*e.g.*, images) can be used for clustering and/or certain aspects of supervised learning. In contrast, supervised learning maps input data, including meta-level features and topographies gleaned from unsupervised learning, to target

Tesla Exhibit 1003.047

outputs. For example, the target output of a mobile robot vehicle might be to avoid obstacles and/or react to traffic guidance mechanisms, and a human in the loop would provide the mapping from input data (*e.g.*, images) to safe mobile robot vehicle operation. Supervised and unsupervised learning models that integrate HITL allow humans to provide direct feedback to the model(s) for predictions, identification, and decision making. As with most control models, training and operation can be based on individual images, image sequences, accumulated image aspects, and even the rate at which aspects of images change. Through single and multiple events, the model refines its understanding of the world around it, and significantly improves its abilities to, for example, safely control a driver-optional mobile robot. Examples of HITL AI models abound, including supervised and unsupervised models, and were well understood at least as far back as the 1990s.

43. As described in more detail below, these machine learning principles have also been applied in the prior art to computer simulations of robots, and other autonomous devices, as well as video games.

VI. The Prior Art

A. U.S. No. 8,142,268 ("Thomas")

44. Thomas is a U.S. patent filed on November 30, 2005, which issued on March 27, 2012. Thomas discloses a simulator and database used to simulate the behavior of a user in a computer program, such as a multiplayer video game. I

Tesla Exhibit 1003.048

understand that Thomas was not of record during the prosecution of the applications that became the '134 Patent or the '585 Patent.

45. Thomas discloses a system and method for "simulating a particular user's behavior in an interactive computer system." Ex.1005, Abstract. The system includes a (1) learning mode and (2) a simulation/playback mode.

46. During learning mode, the simulator records "the user's decision/input, such as a pressed button, combination of buttons, or joystick move of a controller, in response to the particular situation and/or event." *Id*., 4:67-5:5. The simulator can be applied to "a multi-user interactive computer game, wherein each user controls one or more avatars, such as a sports game." *Id*., 4:25-27. In a sports game, the user's input controls an avatar representing a player, causing the avatar to perform an action such as shooting or passing the ball. The simulator records the user's input linked with the circumstances at the time of the event, such as the time remaining in the game, the score, which avatar is currently controlling the ball and the spatial position of avatars on the virtual field or court:

> During the learning mode, the simulator is configured to monitor the user's interaction with the computer program for one or more events and analyze one or more current circumstances during the occurrence of the one or more events. ***The simulator is further configured to record one or more user inputs in response to the occurrence of the one or more events, store the one or more user inputs in one or more***

Tesla Exhibit 1003.049

data fields, and associate the one or more data fields with the one or more events and the one or more current circumstances.

***

Each event 685 and situation 655 are defined by a set of one or more parameters. In the instance where the computer program 110 is a multi-user interactive computer game, wherein each user controls one or more avatars, such as a sports game or a first person shooter game, events 685 and situations 655 may be defined by a number of parameters. One common parameter that defines a situation may be time 660, e.g., time left in a particular session or game, and/or time played in a particular session or game. ***Another may include spatial position of avatars controlled by the user 665 on the interface screen. For example, in football, the spatial position may be position on a football field, as will be explained in more detail below.*** In the case of a first person shooter game, spatial position may be location of controlled avatar within the virtual reality of the game. ***Another parameter may be the spatial position(s) of the avatars controlled by one or more user opponents 675 of the game.*** Another parameter may be the score if applicable to the computer program 110.

***

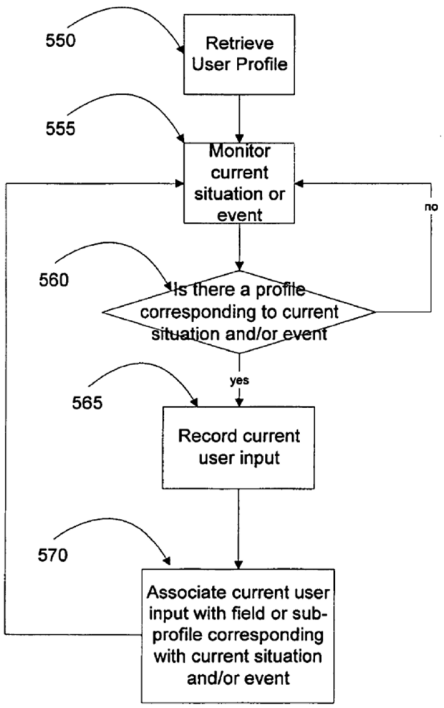
For example, in the game of basketball, if a user is controlling a basketball player near the three-point line and sees two defenders, one on each side, ***the user may choose to shoot a jump shot or split the***

Tesla Exhibit 1003.050

defenders-responses that may be recorded in the profile 650 and linked with opponent's avatar's position.[2]

Id., Abstract, 4:23-40, 5:13-18.

47. Figure 3a, reproduced below, illustrates the learning mode process:



Ex. 1005, Fig. 3a

48. The data recorded during learning mode "enables the analysis of a user's behavior, tendencies, and inclinations for a given event." *Id*., 5:24-25; *see also* 9:62-63. In simulation (also called playback) mode, the simulator uses the data recorded during learning mode to autonomously simulate the user. During

[2] Emphasis added unless otherwise noted

Tesla Exhibit 1003.051

simulation mode, the simulator "monitors and analyzes the operation of the computer program in search of particular events and/or situations," analyzing factors such as the spatial locations of player avatars. *Id*., 5:34-36. When the simulator detects a particular situation associated with data recorded during learning mode, "the simulator 120 can invoke a corresponding reaction based on the stored data, thereby simulating the user." *Id*., 5:28-29.

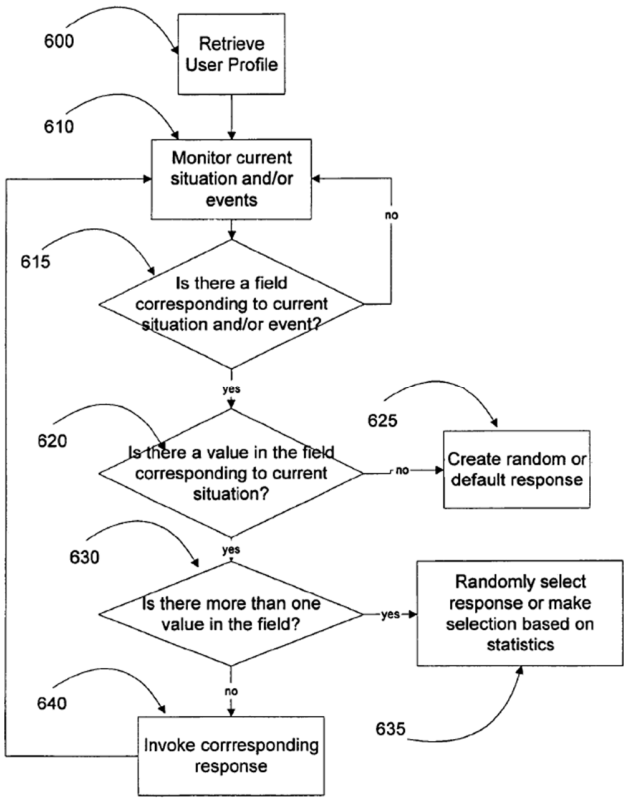
> For an offensive rebound event, the simulator 120, during learning mode 550, will record which avatar grabbed the rebound and whether the user immediately passed the ball, which indicates a "reset" type rebound, or whether the user shoots the ball, which indicates a "shoot" type rebound. Again, current circumstances are recorded for the event. ***During playback 600, the simulator 120 will analyze the current circumstances during the occurrence of an offensive rebound event and invoke the proper response accordingly, e.g., whether to invoke a "shoot" rebound or a "reset" rebound.***
>
> ***
>
> During playback 600, for a pass out of shot event, the simulator 120 will analyze how well the passing avatar and the receiving avatar are covered. ***The simulator 120 will invoke a pass if the current circumstances meet the criteria, e.g., how much the passing and receiving avatars are covered.***
>
> ***
>
> For example, in the case of a football game, if the simulated user is on its own twenty yard line and just failed to reach the first down mark

Tesla Exhibit 1003.052

> after a third down play, e.g., it is fourth down and one, the simulator 120 can analyze the time left on the play clock and the score. If the time left on the play clock is less than two minutes, and simulated user is losing by 5 points, then the simulator 120 will analyze such a situation to determine whether the user would kick a field goal or try to reach the end zone for a touchdown. … ***If there is a stored user input value, (decision block 630) then the simulator 120 will generate a response corresponding to that value, which would represent the second user's response to the occurring situation or event (action block 640), and then continue monitoring (action block 610).***

Id., 7:45-54, 7:59-64, 5:45-62.

49. Figure 3b, reproduced below, illustrates the simulation mode process:



Ex. 1005, Fig. 3b

50. The inputs recorded during learning mode are stored in a "data structure, generally referred to as a user's profile, corresponding to the first user. Generally, the data is logically linked to an event, or situation, that initiated the user's input." *Id*., 3:58-61. During simulation mode, if a "second user desires to interact with the first user, but the first user is not available, the simulator 120 will then retrieve input data from the first user … from the database 130 (action block 220) and simulate the first user's behavior based on the retrieved data (action block 230)." *Id*., 3:64-4:2. In some embodiments, where the simulator and database reside on a

Tesla Exhibit 1003.054

server, "any participating user can upload its own user profile for access by any other participating user, and further any participating users can download or access any user profile available in the database 130." *Id*., 19:58-62.

51. The '134 and '585 Patent specifications describe an example embodiment where the system learns the behavior of a player in a video game, such as a first person shooter or racing game. '134 Patent[3] at 163:54-56 ("Examples of 3D Computer Game 18*a* include a ***first shooter game, a flight simulation, a driving simulation, and/or others***"); '585 Patent[4] at 163:20-22 ("Examples of 3D Computer Game 18 a include a ***first shooter game, a flight simulation, a driving simulation, and/or others***"). The user can control the operations of an avatar through inputting directions in a game controller, keyboard, joystick or other input device, causing the avatar to execute operations such as moving, maneuvering, shooting or jumping. '585 at 163:36-42 ("User 50 can interact…through ***inputting operating directions (i.e. manipulating game controller elements, pressing keyboard buttons, tilting a joystick, etc.) via Human-machine Interface 23 such as a game controller,***

[3] Citations to the "'134 Patent" herein refer to Ex. 1001 in the IPR Proceeding concerning the '134 Patent.

[4] Citations to the "'585 Patent" herein refer to Ex. 1001 in the IPR Proceeding concerning the '585 Patent.

Tesla Exhibit 1003.055

keyboard, joystick, or other input device. For instance, ***responsive to User's 50 manipulating one or more game controller elements***, Processor 11 or 3D Computer Game 18 a may cause ***Avatar 180a to move, maneuver, shoot, jump, and/or perform other operations***"); '134 at 163:56-64 ("Avatar 605 may be or include Soldier 605 a within 3D Computer Game 18 a. ***Soldier 605a can be controlled by User 50 (i.e. game player, etc.) through inputting operating directions via Human-machine Interface 23 such as a game controller, keyboard, joystick, or other input device***. For instance, responsive to User's 50 manipulating one or more game controller elements, ***Soldier 605a may be caused to move, maneuver, shoot, jump, and/or perform other operations***.").

52. The system in Thomas can also be used in a first person shooter or racing game. Ex. 1005, 4:24-27 ("In the instance where the computer program 110 is a multi-user interactive computer game, wherein each user controls one or more avatars, such as a sports game or ***a first person shooter game***…."); 7:7-9 ("Other computer programs 110 can include individual sports video games, such as golf, boxing, ***racing***, etc."). In a football or basketball video game, a user can use a game controller to direct an avatar to execute operations such as moving, maneuvering, jumping or passing/shooting a ball. *See id*., 4:67-5:16 ("[T]he simulator 120 will record the user's decision/input, such as a ***pressed button, combination of buttons, or joystick move of a controller***… For example, in the game of basketball,…***the***

Tesla Exhibit 1003.056

user may choose to shoot a jump shot or split the defenders."); 9:23-25 ("the user's input ***generally includes controlling the movements and actions of one or more of the avatars*** 320."). Thus, Thomas discloses video game embodiments that correspond to an exemplary embodiment disclosed in the Simulation Patents.

B. U.S. No. 9,434,072 ("Buehler")

53. Buehler is a U.S. patent filed on September 17, 2012, which issued on September 6, 2016. Buehler discloses training vision-guided robots through two-way communication with a human trainer. I understand that Buehler was not of record during the prosecution of the applications that became the '134 Patent or the '585 Patent.

54. Buehler discloses systems and methods "for robot learning and robot training that involve two-way communication between a vision-guided robot and a human trainer." Ex. 1006, 1:65-67. Buehler describes a robotic system, where the robot can include an optical camera and an end-effector, such as a gripper. *Id*., 6:52-54 ("The robot includes at least one movable appendage with ***an end-effector (e.g., a gripper)*** for performing the action on the object."); 8:20-24 ("The input devices include … ***one or more cameras (or other optical sensors)*** and associated image-processing hardware and/or software."). The robot captures and stores camera images of objects, along with associated routines for completing tasks. *Id*., 2:6-12 ("For example, ***the robot may use a camera to acquire images of its***

Tesla Exhibit 1003.057

environment….”); 3:36-39 (“***relevant information about the accomplished task***, e.g., the established grasp point and motions of the gripper and/or robot arm, ***may be stored along with the visual model of the object***.”). When a robot encounters a new object, it uses its cameras to capture real-time images depicting the object. The robot compares the newly captured images to stored images and templates depicting previously encountered objects. If the real-time images match a stored image, the system executes the associated object-specific routine. *Id*., 18:50-54 (“The detection algorithm 508 ***compares real-time camera images against the template*** and ***determines, based on a similarity metric, whether the real-time image contains the object of interest***.”); 22:42-44 (“If the robot detects, for instance, a conveyor belt, it may select an action for picking up objects from the conveyor belt.”); 3:40-42 (“When the robot later, in execution mode, ***detects an object of the same type, it may utilize the stored information to execute an object-specific pick-up routine***.”).

55. The robotic system has two modes: training mode and execution mode. In training mode, “a user is present and engaged with the robot, teaching the robot how to perform certain tasks.” *Id*., 10:46-48. In execution mode, the robot “can generally not appeal to the user for help, and its primary focus is on executing a task specified in the training mode.” *Id.*, 10:49-51. In training mode, the system can overlay computer graphics on the real-time images, including a graphical representation of the robot and its components, such as its end-effector. When the

Tesla Exhibit 1003.058

acquired images correspond to stored image data, the virtual representation of the robot simulates performing a task based on a previously stored object-specific routine. If the user is not satisfied with the simulated robot's performance, the user and/or the robot can modify the routine until the action is performed correctly:

> ***Following creation of the visual model, the robot may test its performance by simulating detection of the object using the model*** (step 404).
> ***
> Rather than physically attempting to grasp the object, ***the robot may overlay virtual grippers onto a camera view of the object, and adjust the location of the virtual grippers based on input from the user***, such as user-guided movement of the robot arm. When the virtual gripper appears to be in the proper position relative to the robot, the robot may try to grip and pick up the object with its physical gripper. If the attempt fails, the robot returns to the virtual-gripper adjustment step. Otherwise, ***relevant information about the accomplished task, e.g., the established grasp point and motions of the gripper and/or robot arm, may be stored along with the visual model of the object***.
> ***
> Following the user's approval of the selected visual model, ***the robot may overlay virtual grippers on the image (or video stream) of the object to indicate where the grasp point would be***, given the current arm location and orientation (step 606). … If the virtual gripper is not positioned around the proper part of the object (as determined in step 608), the user may adjust the arm to roughly establish the proper grasp point (step 610). The robot may then fine-tune the arm location to

Tesla Exhibit 1003.059

> optimize the gripper location relative to the rough grasp point on the object (step 612). ***This process of coarse and fine adjustments of the gripper by the human trainer and the robot, respectively, may be iterated until the user is satisfied with the grasp point and gripper orientation shown in the visual display***.

Id., 19:34-36, 3:28-39, 21:46-64.

C. U.S. 9,008,840 ("Ponulak")

56. Ponulak is a U.S. patent filed on April 19, 2013, which issued on April 14, 2015. Ponulak discloses a method for training robots through reinforced learning. Ponulak, 1:37-39. In relevant part, Ponulak discloses a "spiking neural network" generating "motor commands" for autonomously operating a robot based on an input digital image. Ex. 1007, 8:31-33 ("[C]ertain aspects of the disclosure can best be understood in the context of robotic adaptive control system comprising a spiking neural network…."), 9:1-21 ("The output 108 may comprise one or more motor commands…. The signal 106 may comprise an array of pixel values in the input image…."). Ponulak further describes that the neural network "may comprise a hierarchy of processing blocks (cells blocks). In some implementations, the lowest level L1 cell 1152 of the apparatus 1150 may comprise logic and memory blocks…." *Id*., 36:10-13. Ponulak further describes that certain cells may be devoted to object detection, while other interconnected cells may be devoted to determining motor commands based on the input image data:

Tesla Exhibit 1003.060

> A number of ***cell blocks*** may be arranged in a cluster and may ***communicate with one another via local interconnects*** 1162, 1164. Individual clusters may form higher level cell, e.g., cell L2, denoted as 1154 in FIG. 11 d. Similarly, ***several L2 clusters may communicate with one another via a second level interconnect*** 1166….
>
> Different cell levels (e.g., L1, L2, L3) of the apparatus 1150 may be configured to perform functionality various levels of complexity. In some implementations, individual L1 cells may process in parallel different portions of the visual input (e.g., encode individual pixel blocks, and/or encode motion signal), with the L2, L3 cells performing progressively ***higher level functionality (e.g., object detection)***. Individual ***ones of L2, L3, cells may perform different aspects of operating a robot*** with one or more L2/L3 cells processing visual data from a camera, and ***other L2/L3 cells operating motor control block*** for implementing lens motion what tracking an object or performing lens stabilization functions.

Id., 36:16-41.

VII. Claim Construction

57. I understand that the words of the claims of a patent are generally given the ordinary and customary meaning as interpreted by a person of ordinary skill in the art at the time of invention. I have been informed that patent claims are construed in light of the claim language, the specification (including the drawings), and the prosecution history (including the references cited during the prosecution of the patent).

Tesla Exhibit 1003.061

1. “means for processing” (’585 Patent at claim 20)

58. Claim 20 of the ’585 Patent requires a “means for processing” that performs the following functions:

- “accessing one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program”
- “receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program….”
- “determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures”
- “at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program…autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.”

’585 Patent, claim 20.

Tesla Exhibit 1003.062

59. Claim 20 of the ’585 Patent does not describe any corresponding structure for these claimed functions. Accordingly, in determining the corresponding structure for these claimed functions, a POSITA would have looked to the ’585 Patent specification. As described below, the specification describes that one or more processors or processor circuits can perform the claimed functions of accessing a memory, receiving a new digital picture, determining[5] the first one or more instruction sets, and executing the first one or more instruction sets:

> ***[O]ne or more processor circuits***…perform operations comprising: ***accessing a memory unit*** that stores a plurality of digital pictures correlated with one or more instruction sets for operating an object of an application, ***receiving a new digital picture*** from a renderer,…***anticipating the one or more instruction sets*** for operating the object of the application correlated with the first digital picture based on at least a partial match between the new digital picture and the first digital picture…[and] ***causing an execution of the one or more instruction sets*** for operating the object of the application correlated with the first digital picture, the causing performed in response to the anticipating the one or more instruction sets for operating the object of the application.
>
> ***

[5] The Specification refers to “determining the first one or more instruction sets” as “anticipating the one or more instruction sets.”

Tesla Exhibit 1003.063

> ***[T]he one or more processors are configured to perform at least: receiving or generating a new one or more digital pictures*** that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program; ***receiving or generating a new one or more digital pictures*** that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program; at least in response to the determining, ***executing the first one or more instruction sets for operating the first object of the first application program***, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.
>
> ***
>
> ***Where a reference to a processor or processor circuit is used herein, it should be understood that one or more processors or processor circuits can be utilized instead***. For example, one or more parallel processors or processor circuits can be utilized. Any other arrangement or plurality of processors or processor circuits can be utilized.

’585 Patent, 28:8-37; *see also id*., claim 1 (requiring a processor that performs the receiving, determining, and executing steps); *id*.,’585 Patent, 79:29-34.

Tesla Exhibit 1003.064

60. Accordingly, a POSITA would have understood that the corresponding structure for the "means for processing" can be a processor or processor circuit that performs the claimed functions.

2. "means for storing" ('585 at claim 20)

61. Claim 20 of the '585 Patent also requires a "means for storing machine readable code," but does not otherwise describe a structure for performing the "storing" function. The specification of the '585 Patent describes storing machine readable code using a storage or memory. The specification, for example, describes that the code for implementing the claimed functions is referred to as Application Program 18 that is executed by processor 11:

> ***Application Program 18*** (also referred to as program, computer program, application, script, ***code***, or other similar name or reference) comprises instructions that can ***provide functionality when executed by processor 11***. Application program 18 can be implemented in a high-level procedural or object-oriented programming language, or in a low-level machine or assembly language. Any language used can be compiled, interpreted, or otherwise translated into machine language.

'585 Patent, 73:24-32; *see also id*. at 73:49-57 ("Examples of Application Program 18 include computer game, virtual world, web browser, media application, word processing application, spreadsheet application, database application, forms-based application, global positioning system (GPS) application, 2D application, 3D

Tesla Exhibit 1003.065

application, operating system, factory automation application, device control application, vehicle control application, and/or other application or program").

62. Figure 1 of the '585 Patent, reproduced below, illustrates that Application Program 18 is stored in Storage 27.

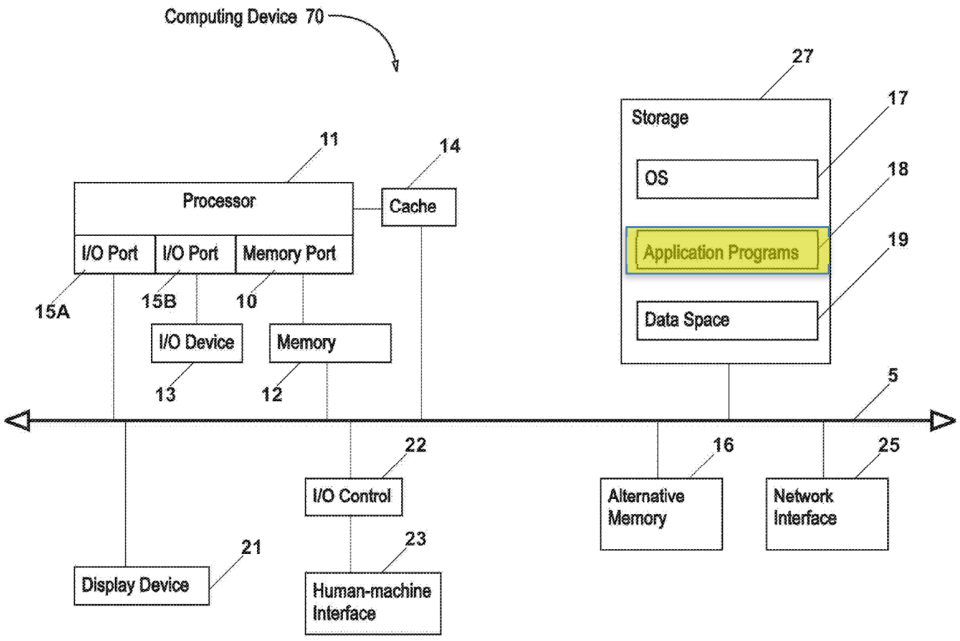


FIG. 1

'585 Patent, Figure 1 (annotated).

The specification further describes that Storage 27 "***may include memory***, which provides an operating system, also referred to as OS 17, additional ***application programs 18 operating*** on OS 17, and/or data space 19 in which additional data or

information can be stored" (*i.e.*, the memory stores application program 18). *See* '585 Patent, 71:42-46.

63. Accordingly, a POSITA would have understood the corresponding structure for storing machine-readable code to be a storage or memory.

3. "means for generating or receiving…"; "means for determining …"; "means for executing…" ('134 Patent at claim 19 and claim 20)

64. Claim 19 of the '134 Patent requires a "means for generating or receiving a second one or more object representations…," a "means for determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second one or more object representations and the first one or more object representations…," and a "means for executing, at least in response to the determining, the first one or more instruction sets for operating the first avatar of the application…." '134 Patent, claim 19. The '134 specification describes "one or more processor circuits…perform operations comprising: receiving a first collection of object representations[,]…anticipating[6] the first one or more instruction sets for operating the avatar of the application correlated with the first collection of object representations based on at least a partial

[6] As noted above, use of the term "anticipating" in the specification corresponds to "determining" in the claims of the 134 Patent.

Tesla Exhibit 1003.067

match between the new collection of object representations and the first collection of object representations…[and] causing an execution of the first one or more instruction sets for operating the avatar of the application." '134 Patent, 17:19-54. Claim 1 of the '134 Patent also requires the generating/receiving function, the determining function, and the executing function be performed by one or more processors:

> one or more non-transitory machine readable media storing machine readable code that, when executed by the one or more processors, ***causes the one or more processors to perform at least…generating or receiving a second one or more object representations***, wherein the second one or more object representations represent one or more objects of the application; ***determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second one or more object representations and the first one or more object representations***; and at least in response to the determining, ***causing the first avatar of the application or a second avatar of the application to perform one or more operations defined by the first one or more instruction sets*** for operating the first avatar of the application at least by ***executing the first one or more instruction sets for operating the first avatar of the application.***

'134 Patent, claim 1.

65. Additionally, claim 20 of the '134 Patent requires that the corresponding structure for the functions claimed in claim 19 "includes one or more processors." '134 Patent, claim 20.

Tesla Exhibit 1003.068

66. Accordingly, a POSITA would have understood that the corresponding structure for the functions set forth in claim 20 of the '134 Patent can comprise a processor or processor circuit.

VIII. Prosecution History

A. U.S. No. 11,113,585 ("the '585 Patent")

67. The application that would issue as the '585 Patent was filed on January 1, 2019, as a continuation of application No. 15/835,434, which issued as U.S. 10,210,434 and was filed on December 7, 2017, which was itself a continuation of application No. 15/245,046, which issued as U.S. 9,864,933 and was filed on August 23, 2016. '585 File History, .005, .106-107.[7]

68. On December 16, 2020, the examiner issued a non-final rejection, which rejected then pending claims 1-16 for indefiniteness. *Id.*, .139, .322-.323. Claims 1, 2, 5-8, 12, 13, 15-17, 19, and 20 were rejected as anticipated by Perez (App. No. 2011/0007079) under 35 U.S.C. § 102. *Id.* Claims 3, 4, 14, and 18 were rejected as obvious over Perez in view of Archibald (App. No. 2013/0329946) under 35 U.S.C. § 103. *Id.*, .328. Claims 1-8 and 12-20 were rejected on the ground of nonstatutory double patenting as unpatentable over claims from other patents and pending applications of the Applicant: U.S. 9,864,933, U.S. 10,102,449, U.S.

[7] Citations to the "'585 File History" herein refer to Ex. 1002 in the IPR Proceeding concerning the '585 Patent.

Tesla Exhibit 1003.069

10,210,434, U.S. 10,223,621, U.S. 10,452,974, App. No. 16/120,253 and App. No. 16/584,736. *Id.*, .330-.340. Claims 9-11 were objected to as dependent claims based on rejected independent claims, but were found to be otherwise allowable provided that the applicant amended them to overcome the Section 112 indefiniteness rejection. *Id.*, .341.

69. On March 28, 2021, the Applicant amended the pending claims to incorporate various features of claims 9-11 into independent claim 1. *Id.*, .395, .423. In response, the Examiner issued a Notice of Allowance for the amended claims. The Examiner determined that the claims were allowable over Perez because Perez did not disclose the limitation "depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program." *Id.*, .439; '585 Patent, claim 1. The Examiner determined that the cameras in the Perez reference captured the user's gesture, but did not capture the avatar's surroundings. '585 File History, .439.

70. The examiner did not appear to review or address the disclosures in Thomas, Buehler, or Ponulak during prosecution of the '585 Patent application.

Tesla Exhibit 1003.070

B. U.S. No. 10,607,134 (the ’134 Patent)

71. The application that would issue as the ’134 Patent was filed on December 19, 2016 and did not claim priority to any other application. ’134 File History, .050, .216.[8]

72. On April 25, 2019, the Applicant requested a preliminary amendment of the claims. *Id.*, .266. On August 10, 2019, the Applicant requested another preliminary amendment of the claims. *Id.*, .293. On November 25, 2019, the Applicant initiated an examiner interview, where the Applicant expressed a desire to further amend the claims, while the Examiner informed the Applicant of a nonstatutory double patenting rejection based on U.S. Patent No. 10,402,731. *Id.*, .333-.334. In response, the Applicant filed a terminal disclaimer, which resulted in allowance of the claims. *Id.*, .307-.308, .312.

73. The examiner did not appear to review or address the disclosures in Thomas, Buehler, or Ponulak during prosecution of the ’134 Patent application.

IX. The ’585 Patent

A. The Claims

74. I understand that the petition challenges all the claims of the ’585 Patent. Independent claim 1 of the ’585 Patent is directed to a system or method in

[8] Citations to the “’134 File History” herein refer to Ex. 1002 in the IPR Proceeding concerning the ’134 Patent.

Tesla Exhibit 1003.071

which an object of an application autonomously executes operations defined by instruction set(s) corresponding to new digital picture(s) based on a comparison to previously stored digital picture(s). As I explain in this Declaration, a POSITA, in and prior to 2016, would have understood that there was nothing inventive about this combination of features, which were well known in the art at the time. Claim 1 is reproduced below:

> A system comprising:
> (a) one or more processors; and
> (b) one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program, wherein the one or more processors are configured to perform at least:
> (c) receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program; determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures; and
> (d) at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more

Tesla Exhibit 1003.072

operations defined by the first one or more instruction sets for operating the first object of the first application program.

'585 Patent, claim 1.

75. Independent claim 19 is nearly identical to claim 1, but discloses a method instead of a system.

76. Independent claim 20 is also nearly identical to claim 1, but discloses a system including a "means for processing" and a "means for storing."

77. Dependent claims 2-18, which all depend from claim 1, can be summarized as follows in the table below based on the additional steps or features that they require. As I further explain in this Declaration, the steps and features added by these dependent claims were also well-known at or prior to 2016, and a POSITA would have understood that there was nothing inventive about the combination of steps and features recited in these claims.

Tesla Exhibit 1003.073

Dependent Claim	Additional Features Claimed
Claim 2	In addition to the limitations of claim 1, claim 2 requires that at least a portion of the first correlation is learned in a learning process, where an object of the application is operated by a human or non-human user, and that the memory stores a second correlation, which is learned in a separate learning process where the same user operates the same object. + **1st correlation** learned with operation of **1st object** by **1st user** ○ **1st user** is a human or non-human user + **2nd correlation** learned in **2nd learning process** with operation of **1st object** by **1st user**

Tesla Exhibit 1003.074

Dependent Claim	Additional Features Claimed
Claim 3	In addition to the limitations of claim 1, claim 3 requires that at least a portion of the first correlation is learned in a learning process, where an object of the application is operated by a human or non-human user, and that the memory stores a second correlation, which is learned in a separate learning process where a different user operates the same object. + **1st correlation** learned with operation of **1st object** by **1st user** ○ **1st user** is a human or non-human user + **2nd correlation** learned in **2nd learning process** with operation of **1st object** by **2nd user** ○ **2nd user** is a human or non-human user

Tesla Exhibit 1003.075

Dependent Claim	Additional Features Claimed
Claim 4	In addition to the limitations of claim 1, claim 4 further requires that at least a portion of the first correlation is learned in a learning process, where an object of the application is operated by a human or non-human user, and a second correlation including instructions for operating a third object of the application or a second object of a different application, learned in a separate learning process where the second/third object is operated by a user. + **1st correlation** learned with operation of **1st object** by **1st user** ○ **1st user** is a human or non-human user + **2nd correlation** for operating **3rd object** of **1st application** or **2nd object** of **2nd application** + **2nd correlation** learned in **2nd learning process** with user operation of **3rd object** of **1st application** or **2nd object** of **2nd application**

Tesla Exhibit 1003.076

Dependent Claim	Additional Features Claimed
Claim 5	In addition to the limitations of claim 1, claim 5 further requires that the new digital picture(s) include at least part of the surroundings of a first object of the first application, and the first instruction set(s) be applied to the first object of the first application, which autonomously executes the defined operations. + **New digital pictures (DP)** include surroundings of **1st object** + **1st object** autonomously executes **1st instructions**
Claim 6	In addition to the limitations of claim 1, claim 6 further requires that the new digital picture(s) include at least part of the surroundings of a second object of the first application and that the first instructions for operating the first object be applied to the second object, which autonomously executes the defined operations. + **New digital pictures (DP)** include surroundings of **2nd object** + **2nd object** autonomously executes **1st instructions** for **1st object**

Tesla Exhibit 1003.077

Dependent Claim	Additional Features Claimed
Claim 7	In addition to the limitations of claim 1, claim 7 further requires that the new digital picture(s) include at least a portion of the surrounding of the first object of a separate application and that the first instructions for the first object of the first application be applied to the first object of the second application, causing the object to autonomously perform the defined operations. + **New DP** include surroundings of **1st object** of **2nd application** + **1st object** of **2nd application** autonomously executes **1st instructions** for **1st object** of **1st application**
Claim 8	In addition to the limitations of claim 1, claim 8 further requires modifying the first or new digital pictures and determining partial match through one of eight options, including determining partial match between the modified first digital picture(s) and the new digital picture(s). + Modifying **1st DP** or **new DP** + Partial match includes modified DP

Tesla Exhibit 1003.078

Dependent Claim	Additional Features Claimed
Claim 9	In addition to the limitations of claim 1, claim 9 further requires that the new digital picture(s) depict a first-person view including at least a portion of the object's surrounding from the perspective of the first or second object of the first application or the first object of a separate application. + **New DP** depicts 1st person POV of the surrounding of **1st object**, **2nd object** of **1st application** or **1st object** of **2nd application** o 1st person POV = perspective from **1st object**, **2nd object** of **1st application** or **1st object** of **2nd application**

Tesla Exhibit 1003.079

Dependent Claim	Additional Features Claimed
Claim 10	In addition to the limitations of claim 1, claim 10 further requires that the new digital picture(s) depict a third-person view including at least a portion of the object's surrounding from the perspective of an observer of the first or second object of the first application or the first object of a separate application. + **New DP** depict 3rd person POV of the surrounding of **1st object**, **2nd object** of **1st application** or **1st object** of **2nd application** o 3rd person POV = perspective from observer of **1st object**, **2nd object** of **1st application** or **1st object** of **2nd application**

Tesla Exhibit 1003.080

Dependent Claim	Additional Features Claimed
Claim 11	In addition to the limitations of claim 1, claim 11 further requires that the new digital picture(s) depict a top-down view including at least a portion of the object's surrounding from above the first or second object of the first application or the first object of a separate application. + **New DP** depict top-down POV of the surrounding of **1st object**, **2nd object** of **1st application** or **1st object** of **2nd application** o Top-down POV = perspective from above **1st object**, **2nd object** of **1st application** or **1st object** of **2nd application**

Tesla Exhibit 1003.081

Dependent Claim	Additional Features Claimed
Claim 12	In addition to the limitations of claim 1, claim 12 further requires modifying the first instructions for the first object of the application and executing the modified instructions, causing the first or second object, or the first object of a separate application, to autonomously perform the modified instructions. + Modifying **1st instructions** for **1st object** + **1st object**, **2nd object** of **1st application** or **1st object** of **2nd application** autonomously perform modified instructions

Tesla Exhibit 1003.082

Dependent Claim	Additional Features Claimed
Claim 13	In addition to the limitations of claim 1, claim 13 further requires generating representation(s) of one or more objects in the surroundings of the first or second object of the first application or first object of a separate application program, which are used to generate new digital picture(s) depicting the generated representation(s). + Generating object representations (OR) o OR = representations of objects in the surroundings of the **1st object**, **2nd object** of **1st application** or **1st object** of **2nd application** + OR used to generate **new DP** depicting OR
Claim 14	In addition to the limitations of claim 1, claim 14 further requires the first instructions for operating the first object of the first application include information about the state of the first object. + **1st instructions** include state of **1st object** of **1st application**

Dependent Claim	Additional Features Claimed
Claim 15	In addition to the limitations of claim 1, claim 15 further requires the first correlation to include one or more connections connecting the first digital picture(s) with the first instruction set(s) for operating the first object of the first application. + **1st correlation** includes connection between **1st DP** and **1st instructions**
Claim 16	In addition to the limitations of claim 1, claim 16 further requires a neural network where the first digital picture(s) are connected to the first instruction set(s) for operating the first object of the first application program through a connection or node. + Neural network containing **1st DP** connected with **1st instructions** using a connection or node o Neural network=data structure converting input to output
Claim 17	In addition to the limitations of claim 1, claim 17 further requires all of the structures listed in claim 1 to include one of the various listed embodiments. + Various embodiments

Tesla Exhibit 1003.084

Dependent Claim	Additional Features Claimed
Claim 18	In addition to the limitations of claim 1, claim 18 requires the first and new digital pictures listed in claim 1 to comprise one of the various listed embodiments, such as a whole digital picture or part of a digital picture. + **1st DP** = one of the listed embodiments + **New DP** = one of the listed embodiments

B. 585 Patent Grounds 1 and 2: Anticipation and/or Obviousness of Claims 1-7 and 10-20 based on Thomas

78. As set forth in detail below, claims 1-7 and 10-20 of the '585 Patent are anticipated and/or rendered obvious by Thomas. The discussion below specifies where each element of the aforementioned claims is found in Thomas, and includes a detailed explanation of the significance of the quotations and citations.

1. Claim 1

79. As described below, Thomas anticipates claim 1 of the '585 Patent. Thomas includes several examples that demonstrate anticipation, including the pump fake (Ex.1005, 8:8-15) and pass out of shot (Ex.1005, 7:55-64) examples. The tables below identify how those examples map to various elements of claim 1:

Tesla Exhibit 1003.085

Pump Fake (Ex.1005, 8:8-15)	• A first one or more digital pictures = representations, features, and/or values of the spatial position of defensive players recorded during learning mode • A first one or more instruction sets = user inputs to cause the user-controlled avatar to pump fake during learning mode • A first object of a first application program = an avatar that pump fakes during learning mode based on user controls, which is later autonomously operated during simulation mode • A new one or more digital pictures = video game displayed on interface screen during simulation mode and the corresponding representations, features, and/or values of the spatial position of defensive players
Pass Out of Shot (Ex.1005, 7:55-64)	• A first one or more digital pictures = representations, features, and/or values of the spatial position of defensive players recorded during learning mode • A first one or more instruction sets = user inputs to cause the user-controlled avatar to pass the ball during learning mode • A first object of a first application program = an avatar that passes during learning mode

Tesla Exhibit 1003.086

	based on user controls, which is later autonomously operated during simulation mode • A new one or more digital pictures = video game displayed on interface screen during simulation mode and the corresponding representations, features, and/or values of the spatial position of defensive players

a. [1.1] A system comprising: one or more processors; and

Limitation [1.1]	Claimed Features
	System including processors

80. Thomas discloses an interactive computer system for simulating a user's behavior that includes "one or more processors":

> A computer program product comprising a computer-usable medium having a sequence of instructions stored thereon, which, when ***executed by a processor***, causes said ***processor*** to execute a first process for learning the behavior of a first user interacting with a multi-user computer game…
>
> ***
>
> ***The system*** of claim 13, wherein the computer video game is located in a ***local processor*** and the simulator and the database are located in a remote server accessible by the local processor via a wide area network.
>
> ***

Tesla Exhibit 1003.087

> The system of claim 13, wherein the computer video game, the simulator, and the database are located in a remote server accessible by the ***local processor*** via a wide area network.
>
> ***
>
> The system of claim 13, wherein the computer video game is located in a ***local processor*** and the simulator and the database are located on portable memory device enabled to be directly connected to the local processor.

Ex.1005, claim 1; claim 17; claim 18; claim 19;

> The field of the invention relates to interactive computer systems and more particularly to systems and methods for simulating a particular user's behavior in an interactive ***computer system***. In one embodiment, a computer program product includes a simulator configured to simulate a user interacting with a computer program

see id., abstract.

b. [1.2] one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program,

Limitation [1.2]	Claimed Features
	Memories store a **1st correlation** + **1st correlation** = DP + associated instructions + DP = Digital pictures

81. The '585 Patent describes that an "object" can comprise an avatar, a user-controllable object, and/or an object controlled by the system: "In some aspects,

Tesla Exhibit 1003.088

Object 180 includes an avatar. In other aspects, ***Object 180 includes a user-controllable object***. In further aspects, ***Object 180 includes an object controlled by a system***, program, and/or other mechanism for controlling or operating an object." '585 Patent, 81:15-19; *see also id*., 2:29-31 ("In further embodiments, the application includes a computer game, a virtual world, a CAD application, or a computer application."). The user-controlled and simulated avatars disclosed in Thomas, as described in more detail below, are objects of an application as described in the '585 Patent.

82. Thomas discloses that the simulator stores data recorded during learning mode in a database or memory. *See* Ex. 1005, 3:56-57 ("The collected data is then ***stored in the database*** 130 (action block 210)."), claim 19 ("the simulator and the database are located on ***portable memory device*** enabled to be directly connected to the local processor"). Thomas discloses that, during learning mode, the simulator records and stores a user's inputs for operating a first object in an application, such as an avatar in a computer game, (i.e. an instruction set for operating a first object of a first application program), which are "associated with" or "linked to" the situation or event that "initiated the user's input." *Id*., 9:43-45, 11:25-26.

> A user's profile can include one or more play book profiles 500. A play book profile 500 includes a plurality of offensive and defensive situation profiles 510 that include a plurality of play calling profiles

Tesla Exhibit 1003.089

530. … ***A play calling profile 530 stores information about the user's choice of play and the choices made within the play***.

***

During the learn mode, if a first user is using the computer program 110, ***the simulator 120 collects and evaluates data about the first user's input when interacting with the computer program*** 110 (action block 200). ***The collected data is then stored in the database*** 130 (action block 210). … ***Generally, the data is logically linked to an event, or situation, that initiated the user's input.***

***

If the simulator 120 identifies a particular situation and/or event occurring, defined by parameters such as those above (decision block 560), then the simulator 120 will record the user's decision/input, such as a pressed button, combination of buttons, or joystick move of a controller, in response to the particular situation and/or event (action block 565) and store the user's input in the field or sub-profile associated with the particular situation and/or event (action block 570)…. In other words, when a user responds to an event, the ***user's response, when recorded, is not only associated with the occurrence of that event, but also with the situation during the occurrence of that event, which is defined by one or more current circumstances***.

Id., 10:6-14, 3:53-61, 4:62-5:18, 8:8-11. A POSITA would have understood the "association" between the user's commands (*i.e.*, instruction set) and the situation/event comprises a correlation between them.

Tesla Exhibit 1003.090

83. Thomas additionally discloses that the event linked to the user's input can be defined by circumstances in the game, such as the virtual weather, road conditions or the spatial position of avatars:

> ***One common parameter that defines a situation*** may be time … Another ***may include spatial position of avatars controlled by the user 665 on the interface screen***. For example, in football, the spatial position may be position on a football field…. Another parameter may be the ***spatial position(s) of the avatars controlled by one or more user opponents*** 675 of the game.
>
> ***
>
> Thus, the data recorded can be associated with both an event and a situation. In other words, when a user responds to an event, the ***user's response, when recorded, is not only associated with the occurrence of that event, but also with the situation during the occurrence of that event, which is defined by one or more current circumstances***. For example, in the game of basketball, ***if a user is controlling a basketball player near the three-point line and sees two defenders, one on each side, the user may choose to shoot*** a jump shot or split the defenders-responses that may be ***recorded in the profile 650 and linked with opponent's avatar's position.***
>
> ***
>
> ***The interface 300 generally includes a graphical representation of a football field 310, and two sets of avatars*** 320, each representing a football team set to oppose each other. One of the sets of avatars 320

Tesla Exhibit 1003.091

represent a defensive team…. The other set of avatars 320 represents an offensive team….

***

One of ordinary skill in the art would also appreciate that ***other circumstances 680 may be included as parameters, such as virtual weather in games such as golf, or road conditions of racing games***.

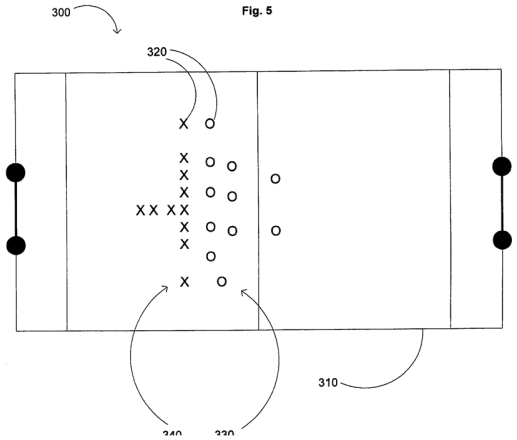
Id., 4:29-39, 5:7-18, 7:65-8:1, 8:67-9:9, 9:43-53, 4:51-54.

84. Claim 1 requires that the memory "stores a first one or more digital pictures." Claim 18 of the '585 Patent describes that the claimed "digital picture" can take a number of forms, including that the first "***one or more digital pictures are***: one or more whole digital pictures, one or more representations of one or more whole digital pictures, one or more portions of at least one digital picture, ***one or more representations of one or more portions of at least one digital picture***, ***one or more features***, one or more representations of one or more features, one or more collections of pixels, ***or one or more collections of values***." '585 Patent, claim 18, *see also id*., 9:14-16 ("A portion of the new or the first digital picture may include a region, a feature, or pixel."). The specification further describes that the term "digital picture" encompasses formats used in non-graphical Application Programs that are never displayed. *Id*., 82:67-83:6 ("In some implementations of ***non-graphical Application Programs 18*** such as simulations, calculations, and/or others, Renderer 91 may render or generate one or more Digital Pictures 525 of Object's 180 visual

Tesla Exhibit 1003.092

surrounding or of views of Application Program 18 to facilitate learning functionalities herein where ***the one or more Digital Pictures 525 are never displayed.***"). Claim 18 clearly demonstrates that the term "digital picture" is not limited to part of or an entire digital image. Rather, a POSITA would understand that the term "digital picture" in the '585 Patent claims encompasses at least one or more representations of one or more portions of at least one digital picture, one or more features, or one or more collections of values.

85. Thomas describes a video game interface including a graphical representation of video game circumstances and events, such as a football field, as illustrated in Figure 5, reproduced below:



Ex. 1005, Figure 5

A POSITA would have understood that the interface displays a stream of digital pictures, depicting features such as the virtual weather, road conditions and the spatial position of avatars. *See* '585 Patent, 151:60-63 ("In other embodiments, a

Tesla Exhibit 1003.093

stream of digital pictures (i.e. motion picture, video, etc.) may include one or more digital pictures. A stream of digital pictures comprises any type or form of digital motion picture.") A POSITA would have also understood that the spatial position of an avatar, the virtual weather, and the virtual road conditions disclosed by Thomas each comprise a representation of a portion of a digital picture depicted in the game situation, a feature, and/or a collection of values. Therefore, a POSITA would have understood that the game circumstances recorded during learning mode in Thomas, (*e.g.*, the spatial position of defenders, the weather, road conditions, etc.) comprise digital pictures as that term is used in the '585 Patent.

86. The examples identified above confirm that Thomas discloses memory storing correlations between a user's inputs (first one or more instruction sets) controlling an avatar in a computer video game (operating a first object of an application) and circumstances in the video game such as the weather, road conditions or spatial location of the avatars (first one or more digital pictures). In the pump fake example, Thomas describes that the system memory stores a command for causing an avatar to pump fake (*i.e.*, an instruction set) that is correlated with a first digital picture (*i.e.*, the spatial position of the defensive avatars). Similarly, in the pass out of shot example, Thomas describes that the system memory stores a command for causing an avatar to pass (*i.e.*, an instruction

Tesla Exhibit 1003.094

set) that is correlated with a first digital picture (*i.e.*, the spatial position of the defensive avatars):

> For a ***pass out of shot*** event, the simulator 120, during learn mode 550, ***records*** a shot type, ***which avatar passed, and which avatar was passed to***. For current circumstances, the simulator 120 ***records how open the receiving avatar was and how covered the passing avatar was***.
>
> ***
>
> For a ***pump fake event***, during the learning mode 550, the simulator 120 ***records how often a user pump fakes***. For current circumstances, the simulator 120 ***records how covered the user was and user's location***.

Ex. 1005, 7:55-59, 8:8-11.

c. [1.3] wherein the one or more processors are configured to perform at least: receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program;

Limitation [1.3]	Claimed Features
	Processors receiving/generating **new DP** including the surroundings of **1st object** of **1st application**, **2nd object** of **1st application** or **1st object** of **2nd application**

87. In simulation mode, Thomas discloses generating new digital pictures on the interface screen that depict the surroundings of a first object of an application (*i.e.*, the same avatar that the user controlled during learning mode that the system is now simulating during simulation mode). Thomas describes that the surroundings of

Tesla Exhibit 1003.095

the first object include, *inter alia*, the field/court, the virtual weather, the virtual road conditions, and the avatars of the defensive players surrounding the first object:

> FIG. 1 shows a diagram of an interactive computer system 100 that includes an interactive computer program 110 configured to have a plurality of users interact with each other, such as a video game, ***a simulator 120 operatively coupled with the program 110 that is configured to simulate a particular user's decisions and behavior***, and a database 130 coupled with the simulator 120, which stores data used by the simulator 120 to calculate and generate the decisions and behavior. ***The simulator 120 generally operates in two modes****:* (1) learn, or record, mode and ***(2) simulation, or playback, mode****.*
>
> ***
>
> Turning to FIG. 5, a diagram representing ***a video game interface 300 is shown. The interface 300 generally includes a graphical representation of a football field 310, and two sets of avatars 320****,* each representing a football team set to oppose each other.
>
> ***
>
> One common parameter that defines a situation may be time…. Another may include ***spatial position of avatars controlled by the user 665 on the interface screen. For example, in football, the spatial position may be position on a football field****….*
>
> ***
>
> One of ordinary skill in the art would also appreciate that ***other circumstances 680 may be included as parameters, such as virtual weather in games such as golf, or road conditions of racing games****.*

Tesla Exhibit 1003.096

***

During playback mode 600, ***the simulator 120 will monitor for a shoot event and analyze the location of the simulated user and the defensive coverage. If criteria is met, then simulator 120 will invoke a pump fake.***

Id., 3:40-51, 8:66-9:3, 4:29-35, 4:51-54, 8:11-15.

88. Unlike Perez, Thomas explicitly describes generating and displaying information about the *surroundings* of the first object (*i.e.*, the avatar), such as the field/court, spatial positions of other players around the simulated avatar, and the weather, as described above.

89. A POSITA would have also understood that the digital pictures generated during simulation mode by the processor for analysis include at least a feature, a collection of values or a representation of a portion of the digital picture displayed on the interface screen and comprise digital pictures as described by the '585 Patent. Thomas additionally discloses that a processor analyzes the circumstances depicted on the interface screen, such as the spatial location of other players in a sports game, the virtual weather, and the virtual road conditions, and compares the data to circumstances recorded during learning mode. Ex.1005, 8:11-14 ("During playback mode 600, the simulator 120 will monitor for a shoot event and analyze the location of the simulated user and the defensive coverage…."); *id.*, claim 1 ("a sequence of instructions…which, when executed by a processor, causes

Tesla Exhibit 1003.097

said processor to execute a first process…"). A POSITA would have understood that the processor receives at least a representation of a portion of the generated new digital picture (*e.g.*, the spatial position of the defenders), including the surroundings of a first object (*i.e.*, an avatar) in order to compare the new circumstance to the circumstances recorded during learning mode. *See also* Section IX.B.1.d below.

90. The pump fake and pass out of shot examples meet this limitation. As identified in the excerpt below, in each situation the system generates and receives during simulation mode new digital pictures that depict the surroundings of the first object (*i.e.*, the avatar that was controlled by the user during learning mode that the system is now simulating), such as the spatial position of the defenders relative to the first object (*i.e.*, how well the avatar is "covered"):

> During playback 600, for a pass out of shot event, the simulator 120 will ***analyze how well the passing avatar and the receiving avatar are covered***. The simulator 120 ***will invoke a pass*** if the current circumstances meet the criteria, e.g., how much the passing and receiving avatars are covered.
>
> ***
>
> During playback mode 600, the simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and the defensive coverage***. If criteria is met, then simulator 120 ***will invoke a pump fake***.

Ex. 1005, 7:59-64, 8:11-15.

91. Although not required by claim 1, Thomas also discloses a "second object of the application program." In the pass out of shot example above, for

Tesla Exhibit 1003.098

example, the "second object of the application program" corresponds to another avatar on the user-controlled team that receives the pass from the user-controlled avatar. A POSITA would understand that once that avatar receives the pass, the user would control that second avatar during learning mode, which the simulator would simulate in simulation mode.

d. [1.4] determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures; and

Limitation [1.4]	Claimed Features
	1st instructions determined by partial match between **1st DP** and **new DP**

92. The specification of the '585 Patent describes that the partially matching portions of digital pictures "may be determined factoring in at least one of: a type of a feature, an importance of a feature, a location of a feature, a threshold for a similarity in a feature, or a threshold for a difference in a feature" and/or by comparing one or more "portions (i.e. regions, features, pixels, etc.)." '585 Patent, 9:41-45, 114:32-33.

93. Thomas discloses comparing the first digital pictures recorded during learning mode with the new digital pictures generated during simulation mode. Thomas describes that the data recorded and stored during learning mode "enables the analysis of a user's behavior, tendencies, and inclinations for a given event." Ex.

Tesla Exhibit 1003.099

1005, 5:24-25. In simulation mode, when digital pictures representing a new situation in the video game are generated or received, the simulator determines the instructions for operating an avatar by matching the new digital pictures with the first digital pictures recorded in learning mode (*e.g.*, the spatial position of defensive players). *Id*., 5:25-29 ("[t]hus…when the simulator 120 is in simulation mode, and ***a particular event and/or situation occurs, the simulator 120 can invoke a corresponding action based on the stored data, thereby simulating the user***."). If the new digital pictures generated and/or received during simulation mode at least partially match the first digital pictures recorded during learning mode (*e.g.* the spatial positions of the defenders is the same or the user is facing the same players on the opposing team), the system determines an instruction set to be applied to the first object of the application (*e.g.* an avatar representing a basketball player). *See also* Section IX.B.1.e below. This is described in the pump fake and pass out of shot examples in Thomas:

> During playback 600, for a pass out of shot event, the simulator 120 will ***analyze how well the passing avatar and the receiving avatar are covered***. The simulator 120 ***will invoke a pass if the current circumstances meet the criteria, e.g., how much the passing and receiving avatars are covered***.
>
> ***
>
> During playback mode 600, the simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and the defensive***

Tesla Exhibit 1003.100

coverage. ***If criteria is met***, then simulator 120 ***will invoke a pump fake***.

Ex. 1005, 7:59-64, 8:11-15; *id*. at 7:59-64, 10:56-11:22, 12:63-13:1, 12:25-29, 13:5-11.

e. [1.5] at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.

Limitation [1.5]	Claimed Features
	Executing **1st instructions** by **1st object**, **2nd object** or **1st object** of **2nd application** autonomously

94. As I explained above for limitation [1.4], Thomas discloses determining the instruction sets for operating an object of the application (*e.g.* an avatar in a video game) by comparing and matching new digital pictures generated or received during simulation mode with first digital pictures recorded during learning mode. *See* Section IX.B.1.d above.

95. Thomas describes that in simulation mode, when the system detects a particular situation or event (*i.e.*, at least a partial match), the system simulates a user by invoking a corresponding action for operating an avatar (i.e. first object of the first application program) based on data obtained during learning mode. For example, in a basketball game in simulation mode, an avatar representing a

Tesla Exhibit 1003.101

basketball player will pass, shoot or pump fake if the spatial position of the avatars on the interface screen match corresponding data stored during learning mode. This execution step is also disclosed in the pump fake and pass out of shot examples described in Thomas:

> During playback 600, for a pass out of shot event, the simulator 120 will analyze how well the passing avatar and the receiving avatar are covered. ***The simulator 120 will invoke a pass if the current circumstances meet the criteria,*** e.g., how much the passing and receiving avatars are covered.
>
> ***
>
> For a pump fake event…simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and the defensive coverage. If criteria is met, then simulator 120 will invoke a pump fake.***

Ex. 1005, 7:59-64, 8:8-15; *see also id*. at 7:50-54 ("During playback 600, the simulator 120 will ***analyze the current circumstances during the occurrence of an offensive rebound event and invoke the proper response accordingly***, e.g., whether to invoke a "shoot" rebound or a "reset" rebound.").

2. Claim 2

a. [2.1] The system of claim 1, wherein the memory further stores at least a second one or more digital pictures correlated with a second one or more instruction sets for operating the first object of the first application program,

	Claimed Features

Tesla Exhibit 1003.102

Limitation [2.1]	Memory stores **2nd correlation**

96. Thomas discloses that the memory can store additional correlations, including a second correlation, where each correlation includes digital picture(s) correlated with instruction set(s) for operating an avatar in the game (the first object of the first application). As described above, each game circumstance (*i.e.*, digital pictures comprising the spatial position of defensive players) is correlated with instruction(s) for operating the avatar based on the user's response, including, for example, the pump fake event (first correlation) and pass out of shot event (second correlation:

> For a pass out of shot event, the simulator 120, during learn mode 550, ***records a shot type, which avatar passed***, and which avatar was passed to. For current circumstances, the simulator 120 ***records how open the receiving avatar was and how covered the passing avatar was***.
> ***
> For a ***pump fake event***, during the learning mode 550, the simulator 120 ***records how often a user pump fakes***. For current circumstances, the simulator 120 ***records how covered the user was and user's location***.

Ex. 1005, 7:55-59, 8:8-11.

b. [2.2] and wherein at least a portion of the first one or more digital pictures or at least a portion of the first one or more instruction sets for operating the first object of the first application program is learned in a first learning process that includes operating the first object

Tesla Exhibit 1003.103

of the first application program at least partially by a first user,

Limitation [2.2]	Claimed Features
	1st correlation learned with operation of **1st object** by **1st user**

97. Thomas discloses a learning process where a user controls an avatar (first object of the first application) and the simulator records the user's response to a game circumstance or event, characterized by parameters such as the spatial position of avatars (first digital picture):

> A computer program product comprising a computer-usable medium having a sequence of instructions stored thereon, which, when executed by a processor, causes said processor to ***execute a first process for learning the behavior of a first user interacting with a multi-user computer game***…
>
> ***
>
> ***A first user controls one of the teams*** 330/340. The other of the teams 330/340 may be controlled by the computer program 110 or by a second user. In addition, ***the other of the teams 330/340 may be controlled by a simulator 120, which includes a learning mode and a simulation mode***.

Ex. 1005, claim 1, 9:30-34.

98. A POSITA would understand that each situation during the same game is a separate learning process, such as when the system learns the pump fake (*i.e.*, a first learning process):

For a ***pump fake event***, during the learning mode 550, the simulator 120 ***records how often a user pump fakes***. For current circumstances, the simulator 120 ***records how covered the user was and user's location***.

Id., 8:8-11.

c. [2.3] and wherein at least a portion of the second one or more digital pictures or at least a portion of the second one or more instruction sets for operating the first object of the first application program is learned in a second learning process that includes operating the first object of the first application program at least partially by the first user,

Limitation [2.3]	Claimed Features
	2nd correlation learned in a second learning process with operation of **1st object** by **1st user**

99. Thomas discloses that the second correlation (*i.e.*, second instruction set correlated with second digital picture) is learned in a second learning process in which the first object (*i.e.*, the user-controlled avatar) is operated at least partially by the first user. Thomas discloses multiple learning processes can occur in a single game session as the user invokes different actions- such as pump faking or passing out of shot. In particular, when the system learns a pass out of shot correlation during learning mode, a POSITA would understand that comprises a second learning process because it occurs at a different time than the pump fake correlation was learned:

> ***For a pass out of shot event***, the simulator 120, during learn mode 550, ***records a shot type, which avatar passed***, and which avatar was passed to. For current circumstances, the simulator 120 ***records how open*** the receiving avatar was and how covered the passing avatar was.

Id., 7:55-59.

100. Thomas also discloses that a second correlation is learned in a second learning process in which the user operates the first object of the application because it describes that the system learns additional correlations (*i.e.*, a second correlation) in different games (*e.g.*, 30 different games) in which the user operates in learning mode:

> Turning to FIG. 11 a, at step 570, the ***simulator 120 will check if one or more user inputs have already been recorded for the current situation and/or event (decision block 575). If so, then the simulator 120 will associate one or more values with each different type of user input based on the statistical occurrence of each input compared to other inputs*** (action block 577)….
>
> ***
>
> ***During learning mode***, the simulator 120 retrieves the user profile 400 of the user to be simulated (action block 800).… If there is existing data in the playbook profile 500, then the simulator 120 will evaluate whether the data is up to date (action block 840). ***Preferably, data should evolve as the user's interaction evolves. In other words, data recorded thirty games ago may have a lesser impact on the total data available.***

Id., 6:10-16, 10:19-32.

Tesla Exhibit 1003.106

d. [2.4] and wherein the first user is: a human user, or a non-human user.

Limitation [2.4]	Claimed Features
	1st user is a human or non-human user

101. Thomas discloses that the first user is a human user or non-human user playing a game. In particular, Thomas describes that the system learns the instruction sets in learning mode based on user controls provided via control buttons and/or a joystick, which a POSITA would understand is done by a human user.[9] *Id*., 7:28-32 ("If the ***user is controlling the team*** taking the ball inbound, then the simulator 120 records who the inbound was passed; whether the user hurried the inbound, which can be in the form of a ***quick tap of a control button***; whether the user ran an avatar around and then passed…."); *id*., 4:67-5:3 ("the simulator 120 will record the user's decision/input, ***such as a pressed button, combination of buttons, or joystick move of a controller***, in response to the particular situation and/or event"). Although not required for this limitation, Thomas discloses multiple users, including a second user. Ex. 1005, 19:49-50 ("Multiple users can compete against

[9] Even if Patent Owner were to argue that Thomas does not disclose a "human" user, Thomas still meets this limitation because it describes that a "user" controls the first object, which would be either a human or non-human user.

each other …."). For the same reasons I explained for the first user, the second user is a human or non-human user.

3. Claim 3

a. [3.1] The system of claim 1, wherein the memory further stores at least a second one or more digital pictures correlated with a second one or more instruction sets for operating the first object of the first application program, and

102. As I discussed above for limitation [2.1], Thomas discloses a memory storing a second correlation, including second digital picture(s) (representations, features, and/or values of game circumstances displayed on the interface screen) and correlated instruction set(s) for operating the first object of the first application (an avatar). For example, Thomas can store a first correlation for a pump fake event and a second correlation for a pass out of shot event. *See* Section IX.B.2.a above.

b. [3.2] wherein at least a portion of the first one or more digital pictures or at least a portion of the first one or more instruction sets for operating the first object of the first application program is learned in a first learning process that includes operating the first object of the first application program at least partially by a first user, and

103. As I discussed above for limitation [2.2], Thomas discloses a learning process where a user operates a first object of the application (an avatar) and the system records the user's instructions for operating the object correlated with one or more first digital pictures (*e.g.*, representations of the spatial position of the defensive players). *See* Section IX.B.2.b above.

Tesla Exhibit 1003.108

c. [3.3] wherein at least a portion of the second one or more digital pictures or at least a portion of the second one or more instruction sets for operating the first object of the first application program is learned in a second learning process that includes operating the first object of the first application program at least partially by a second user, and

Limitation [3.3]	Claimed Features
	2nd correlation learned in **2nd learning process** with operation of **1st object** by **2nd user**

104. Thomas discloses that the memory can record and store profiles for multiple users, where each profile includes one or more correlations between the user's instructions for operating an avatar (first object) and the circumstances the user was responding to (digital pictures):

> Turning to FIG. 6, ***the inputs for a user are stored in user profiles 400, which, as described above, may be individual data structures in a database*** 130. The general structure of a user profile 400 for this particular video game is shown in FIG. 6….
>
> ***
>
> ***Multiple users can compete against each other either on a single console 750/740 or from multiple consoles*** 750/740. ***For example, a first user can compete on a first video game console 760 against a second user competing on a second video game console*** 760…. In the case where the simulator 120 and database 130 resides on a server 710, ***any participating user can upload its own user profile for access by any other participating user, and further any participating users can download or access any user profile available in the database*** 130.

Ex. 1005, 9:36-43, 19:49-62.

Tesla Exhibit 1003.109

105. A POSITA would have understood, and it would have been obvious, that when a second user plays with the same team and avatars as a first user (*e.g.*, different users playing with the Los Angeles Lakers), the digital pictures and instruction sets (*i.e.*, a first and second digital picture correlated with a first and second instruction set respectively) for operating the same avatar on the Lakers (*i.e.*, the first object) would be learned in a second learning process that includes operating that avatar (*i.e.*, the first object) by a second user.

d. [3.4] wherein the first user is: a human user, or a non-human user, and wherein the second user is: a human user, or a non-human user.

Limitation [3.4]	Additional Features to Claim 2
	+ **2nd user** = human or non-human use is a human or non-human user

106. As I discussed above for limitation [2.4], Thomas discloses that the first user and second user are human users or non-human users. *See* Section IX.B.2.d above.

4. Claim 4

a. [4.1] The system of claim 1, wherein the memory further stores at least a second one or more digital pictures correlated with a second one or more instruction sets for operating a third object of the first application program or a second object of the second application program, and

	Additional Features to Claim 2

Tesla Exhibit 1003.110

Limitation [4.1]	+ **2nd correlation** for operating **3rd object** of **1st application** or **2nd object** of **2nd application**

107. As I discussed above for limitation [2.1], Thomas discloses a memory storing multiple correlations between instruction sets and digital pictures for operating avatars in a video game (*i.e.*, objects of an application program), such as a first correlation associated with a pump fake event and a second correlation associated with a pass out of shot event. *See* Section IX.B.2.a above. Thomas further discloses that a user can control multiple avatars in learning mode, such as different avatars on the same team. Ex. 1005, 4:24-28 ("In the instance where the computer program 110 is a multi-user interactive computer game, wherein each user ***controls one or more avatars*** …"). Accordingly, Thomas discloses and it would have been obvious that the system stores first and second digital picture(s) correlated with first and second instruction sets for multiple avatars on a sports team, such as first, second, and third avatars on the same time (*i.e.*, first, second, and third objects of the application):

> Another type of parameter, in the case of a computer ***game involving one or more avatars***, is the archetype …. ***[A]rchetypes can include true point guard, slasher, shooter, pure shooter, low post scorer, midrange scorer, and role player.***
> ***

For each event, during learning mode 550, the ***simulator 120 records the archetype of the avatar that initiated a monitored event.***

Ex. 1005, 4:45-51, 8:38-40. Thus, Thomas discloses storing first and second digital pictures correlated with first and second instruction sets for operating a third object of the first program (*i.e.*, any of three players on a basketball team that the user controlled during learning mode).

108. Thomas also describes that a user's profile can be accessed by other users. Ex. 1005, 19:60-62 ("[A]ny participating users can download or access any user profile available in the database…."). Thus, when a second user downloads a particular profile for use on his or her console, the digital pictures and instruction from the profile would be used to operate first and second objects (*i.e.*, first and second avatars) in a second application program (*i.e.*, the video game on a different user's console). Thomas therefore also discloses storing digital pictures correlated with instruction sets for operating a second object of a second application program.

b. [4.2] wherein at least a portion of the first one or more digital pictures or at least a portion of the first one or more instruction sets for operating the first object of the first application program is learned in a first learning process that includes operating the first object of the first application program at least partially by a first user, and

109. As I discussed above for limitation [2.2], Thomas discloses a learning process whereby a user operates an avatar (*i.e.*, a first object) and the system records correlations between "digital pictures" (*e.g.*, the spatial position of the avatars) and

Tesla Exhibit 1003.112

"instruction set(s) for operating the first object" (the user's inputs controlling the avatar). *See* Section IX.B.2.b above.

c. [4.3] wherein at least a portion of the second one or more digital pictures or at least a portion of the second one or more instruction sets for operating the third object of the first application program or the second object of the second application program is learned in a second learning process that includes operating the third object of the first application program or the second object of the second application program at least partially by: the first user, or a second user, and

Limitation [4.3]	Claimed Features
	2nd correlation learned in **2nd learning process** with user operation of **3rd object** of **1st application** or **2nd object** of **2nd application**

110. Thomas discloses learning additional instructions for operating the third avatar on the team (third object of the application) in response to a second game event (second digital pictures) through a separate learning process, where a user operates the third avatar. As I explained above, digital pictures and instruction sets are recorded during different instances in the same game session and across multiple game sessions. *See* Section IX.B.2.c above; Ex.1005, 10:27-32 ("If there is existing data in the playbook profile 500, then the simulator 120 will evaluate whether the data is up to date (action block 840). Preferably, data should evolve as the user's interaction evolves. In other words, ***data recorded thirty games ago may have a***

Tesla Exhibit 1003.113

lesser impact on the total data available.”). Each recording comprises a separate learning process, whether in the same instance of the game or across different games, including a second learning process for operating the third avatar on the team (third object of the application) in response to a second game event (second digital pictures).

111. As I described above for Limitations [4.1]-[4.2], a learning process can involve a user controlling a third avatar in the first application, or a second avatar in a second application (such as a second user operating the first, second, and third avatars in a different copy of the video game). *See* Ex.1005, 19:49-53 (“***Multiple users can compete*** against each other either on a single console 750/740 or from multiple consoles 750/740. For example, ***a first user can compete on a first video game console*** 760 against ***a second user competing on a second video game console*** 760.”), 4:24-28 (“In the instance where the computer program 110 is a multi-user interactive computer game, wherein each user ***controls one or more avatars***…”); *see* Section IX.B.4.a-b above.

d. [4.4] wherein the first user is: a human user, or a non-human user, and wherein the second user is: a human user, or a non-human user.

112. As I discussed above for limitation [3.4], Thomas discloses multiple users, including a first and a second user, which are human users or non-human users. *See* Section IX.B.3.d above.

Tesla Exhibit 1003.114

5. Claim 5

a. [5.1] The system of claim 1, wherein the new one or more digital pictures depict the at least the portion of the surrounding of the first object of the first application program, and

113. As I discussed above for limitation [1.3], Thomas discloses that the new digital pictures generated in simulation mode depict part of the avatar's surroundings (*i.e.*, the surroundings of the first object), including the location of other players and the virtual court or field, such as the location of defensive players when the user invokes a pump fake. *See* Section IX.B.1.c above.

b. [5.2] wherein the first one or more instruction sets for operating the first object of the first application program are applied to the first object of the first application program, and wherein the first object of the first application program autonomously performs the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.

Limitation	Claimed Features
[5.2]	**1st object** autonomously executes **1st instructions**

114. As explained for Limitation [1.5], Thomas discloses that when the simulator detects a situation in simulation mode corresponding to a situation recorded in learning mode, the simulator applies the correlated instruction set, causing the avatar to autonomously execute the defined operation, such as performing a pump fake. *See* Section IX.B.1.e above.

Tesla Exhibit 1003.115

6. Claim 6

a. [6.1] The system of claim 1, wherein the new one or more digital pictures depict the at least the portion of the surrounding of the second object of the first application program, and

Limitation [6.1]	Claimed Features
	New DP include surroundings of **2nd object**

115. Thomas discloses that the new digital pictures generated during simulation mode and displayed on the video game interface screen include the surroundings of a second avatar on the football or basketball team (*i.e.*, a second object of the application), such as the virtual court/field and the locations of other players around the second avatar, such as during a pump fake event (*i.e.*, a second avatar would include a second player on the simulated user's team) or a pass out of shot event (*i.e.*, the second avatar is the one that receives the pass):

> For a pass out of shot event, the simulator 120, during learn mode 550, records a shot type, ***which avatar passed, and which avatar was passed to***…. During playback 600, for a pass out of shot event, the simulator 120 will analyze how well ***the passing avatar and the receiving avatar*** are covered….

Ex. 1005, 7:55-64, 8:8-15 (describing a pump fake event for a basketball player, who a POSITA would understand has multiple teammate avatars).

b. [6.2] wherein the first one or more instruction sets for operating the first object of the first application

Tesla Exhibit 1003.116

program are applied to the second object of the first application program, and

Limitation [6.2]	Claimed Features
	1st instructions for 1st object applied to 2nd object

116. Thomas discloses operating a particular avatar through default instructions when the game encounters a situation for which there is no correlated instruction set. Thomas, 5:54-58 ("***If there is no value, i.e., stored user input***, recorded for the given the event and/or situation, then the simulator 120 will generate a random or ***default response*** representing the second user's simulated input, or response."). A POSITA would have understood that if there is no recorded response for a situation, the situation is correlated with the default instructions. For example, a POSITA would understand and it would have been obvious to store a default instruction to shoot or pass correlated with the simulated avatar and defensive coverage being located on a particular place on the court.[10] Thomas additionally discloses that avatars representing basketball players can be classified as one of several archetypes. *Id.*, 4:45-51 ("Another type of parameter, in the case of a computer game involving one or more avatars, is the archetype…. ***[A]rchetypes can***

[10] Without such a correlation, for example, the default instruction could result in a simulated avatar shooting the basketball when positioned on the other end of the court from the basket.

include true point guard, slasher, shooter, pure shooter, low post scorer, midrange scorer, and role player. For each event, during learning mode 550, the ***simulator 120 records the archetype of the avatar that initiated a monitored event***."). A POSITA would have understood and it would have been obvious that avatars of the same archetype (*e.g.*, shooters) would follow the same default instructions in the absence of a learned instruction set. A POSITA would have understood that the same default instruction set would be applied to a first and second avatar of the same archetype (first and second object), such as two shooters in a basketball game. Accordingly, the first one or more instruction sets for operating the first object of the first application program are applied to the second object of the first application program.

117. Thomas also renders this limitation obvious for a second reason. In particular, Thomas discloses storing instruction sets correlated with digital pictures for a first avatar that are learned in learning mode, and that the system records the archetype associated with that avatar, as described above. *See also* Section IX.B.1.b above. It would have been obvious to a POSITA that a learned instruction set for a first avatar would be applied to a second avatar of the same archetype if there is no stored correlation for the second avatar, such as multiple avatars that are shooters, low post scorers, etc. For example, Thomas discloses that the system can invoke a stored instruction set in response to a situation based on time, score, and the spatial

Tesla Exhibit 1003.118

position of the defense without requiring that the instruction set be performed by the identical avatar. *See e.g.*, Ex.1005, 7:39-44 ("During playback mode 600…the simulator 120 will…analyze the current situation, such as time, score, and spatial positions of defensive avatars, to determine which action to invoke that represents the simulated user."). This disclosure confirms that Thomas discloses and/or renders obvious applying an instruction set for a first avatar (*i.e.*, first object) to a second avatar (*i.e.*, second object) when other situation parameters sufficiently match.

c. [6.3] wherein the second object of the first application program autonomously performs the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.

Limitation [6.3]	Claimed Features
	2nd object autonomously executes **1st instructions** for **1st object**

118. As I explained above for Limitations [6.1]-[6.2], Thomas discloses multiple avatars, including a second avatar (*i.e.*, second object), which autonomously perform defined operations when an instruction set is applied. When instructions for operating a first avatar are applied to a different avatar, for example a second avatar representing a different player on the same team that has the same archetype, the second avatar autonomously performs the operation defined by the instruction sets for operating the first avatar.

Tesla Exhibit 1003.119

7. Claim 7

a. [7.1] The system of claim 1, wherein the new one or more digital pictures depict the at least the portion of the surrounding of the first object of the second application program, and

Limitation [7.1]	Claimed Features
	New DP include surroundings of **1st object** of **2nd application**

119. Thomas discloses that a user can download the profile of any other participating user to use on the same or a different console. When a second user downloads the same profile as a first user for use in simulation mode, the new digital pictures generated in simulation mode depict the surroundings of a particular avatar on the second user's video game (first object of the second application program), such as the locations of other players around the avatar in a basketball game when the avatar pump fakes:

> Multiple users can compete against each other either on a single console 750/740 or from multiple consoles 750/740…. ***In the case where the simulator 120 and database 130 resides on a server 710, any participating user can upload its own user profile for access by any other participating user, and further any participating users can download or access any user profile available in the database*** 130.
>
> ***
>
> During simulation mode, if a second user is using the computer program 110 and the second user desires to interact with the first user, but the first user is not available, ***the simulator 120 will then retrieve***

Tesla Exhibit 1003.120

input data from the first user, i.e., the first user's profile, from the database 130 (action block 220) and simulate the first user's behavior based on the retrieved data (action block 230).

Ex. 1005, 19:49-62, 3:62-4:2; *see also* Section IX.B.1.c.

120. As I explain below, in relation to limitation [14.2] for the '134 Patent (*see* Section X.B.14.b below), the two users can include, for example, users playing on the same type of device or different types of devices. For example, a first user could download a profile to play a video game in simulation on a Playstation (*i.e.*, a first application program), while a second user could download a profile to play the same type of video game in simulation mode on a Playstation or PC (*i.e.*, a second application program), as described in Thomas. Ex. 1005, 19:35-40 (describing that the system can be implemented on a computer workstation, Xbox, or Playstation). As I explained above for limitation [1.3], in simulation mode, the simulator generates new digital pictures including at least part of the surroundings of a simulated avatar, such as the basketball court and the other avatars representing players on the court. Thus, when the first user downloads a profile to use in simulation mode on an Xbox the new digital pictures would depict the surroundings of the avatars (*e.g.*, a first object and the other avatars around the first object) of a first application program, and when the second user downloads a profile to use in simulation mode on an Xbox, PC, or Playstation the new digital pictures would

Tesla Exhibit 1003.121

depict the surroundings of the avatars (*e.g.*, a first object) of a second application program.

b. [7.2] wherein the first one or more instruction sets for operating the first object of the first application program are applied to the first object of the second application program, and

Limitation [7.2]	Claimed Features
	1st instructions for **1st object** applied to **1st object** of **2nd application**

121. As explained above, a second user can download the same profile as a first user for use in simulation mode. When the second user uses the same profile as the first user, that will result in the instruction set for operating the first object of the first application program being applied to the first object in a second application program (*i.e.*, simulation of the same avatar in a video game played by different users on the same or different consoles) such that the first object of the second application program autonomously performs the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program. For example, if a simulated avatar pump fakes in simulation mode on a first user's console, the same simulated avatar on the second user's console will also pump fake when the same situation occurs if the first and second user both download the same profile for use in simulation mode. As described in Thomas below, for example, a first user can retrieve a profile for simulation mode

Tesla Exhibit 1003.122

(including a first one or more instruction sets for operating the first object of the first application program):

> During simulation mode, if a second user is using the computer program 110 and the second user desires to interact with the first user, but the first user is not available, ***the simulator 120 will then retrieve input data from the first user, i.e., the first user's profile, from the database 130 (action block 220) and simulate the first user's behavior based on the retrieved data (action block 230)***.
>
> ***
>
> ***During simulation, or playback, mode, as shown in FIG. 3b, when a first user desires to interact with a particular second user via the computer program 110, the simulator 120 retrieves the second user's profile*** *(*action block 600), if it has been created.
>
> ***
>
> [A]ny participating user can upload its own user profile for access by any other participating user, and further ***any participating users can download or access any user profile available in the database 130***.

Ex. 1005, 3:63-4:2, 5:30-62, 19:57-62. If a second user downloads the same profile as well to use in simulation mode, a POSITA would recognize that this would result in the first one or more instruction sets for operating the first object of the first application program (as stored in the profile) being applied to the first object of the second application program (*i.e.*, a different user playing the game in simulation

Tesla Exhibit 1003.123

mode on the same or different console). Therefore, Thomas discloses and/or renders this limitation obvious.

c. [7.3] wherein the first object of the second application program autonomously performs the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.

Limitation [6.3]	Claimed Features
	1st object of **2nd application** autonomously executes **1st instructions** for **1st object**

122. As explained above, when a second user downloads the same profile as a first user for use in simulation mode, an avatar in the second user's game (first object of the second application) autonomously performs the operations defined by the instruction set(s) for operating the corresponding avatar in the first user's game (first object of the first application).

8. Claim 10

a. [10.1] The system of claim 1, wherein the new one or more digital pictures depict a third-person view of the at least the portion of the surrounding of: the first object of the first application program, the second object of the first application program, or the first object of the second application program, and

Limitation [10.1]	Claimed Features
	New DP depict 3rd person POV of the surrounding of **1st object**, **2nd object** or **1st object** of **2nd application**

Tesla Exhibit 1003.124

123. Thomas discloses and it would have been obvious to a POSITA that the new digital pictures in Thomas would depict a third-person view of the surroundings of an avatar (*i.e.*, the first object), for example a view that includes "a third-person view or perspective such as a view that shows an avatar as well as objects around the avatar from an observer's point of view." '585 Patent, 82:8-11. As I explained above, in simulation mode, Thomas describes generating new digital pictures that can be displayed on a video game interface screen. Moreover, the generated digital pictures depict an object (such as an avatar in a sports games or a vehicle in a racing game) and its surroundings (such as the virtual field, court or road or other players around the avatar). Thomas describes that its system and methods can be used to simulate a user's behavior in a basketball, football, golf or racing game:

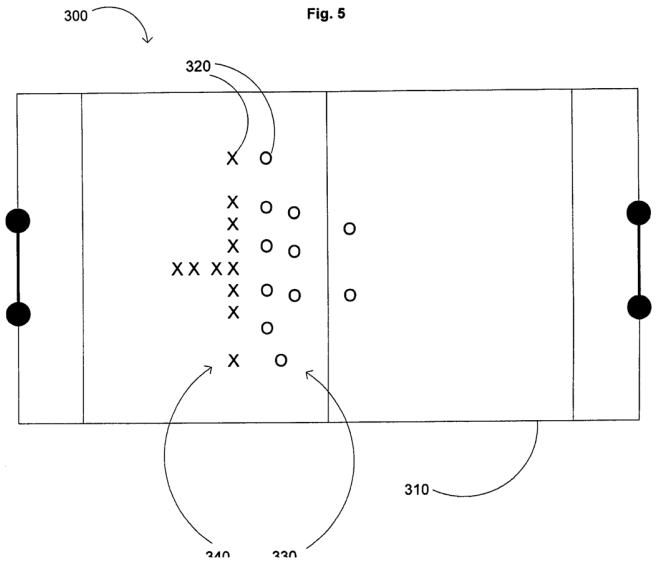
> For example, ***in the game of basketball***, if a user is controlling a basketball player near the three-point line…the user may choose to shoot a jump shot….
>
> ***
>
> ***For example, in football, the spatial position may be position on a football field***, as will be explained in more detail below. ***In the case of a first person shooter game, spatial position may be location of controlled avatar within the virtual reality of the game***. ***Another parameter may be the spatial position(s) of the avatars controlled by one or more user opponents 675 of the game***…. ***One of ordinary skill in the art would also appreciate that other circumstances 680 may be***

Tesla Exhibit 1003.125

included as parameters, such as virtual weather in games such as golf, or road conditions of racing games.

Ex.1005, 5:13-16, 4:19-54.

124. Figure 5 of Thomas, reproduced below, illustrates a third person aerial view of a football game, depicting the players on the field, which would include the surroundings of the simulated avatar (*i.e.*, the first object). Ex. 1005, 8:66-9:3 ("Turning to FIG. 5, ***a diagram representing a video game interface 300 is shown***. The interface 300 generally includes ***a graphical representation of a football field 310, and two sets of avatars 320, each representing a football team set to oppose each other***.").

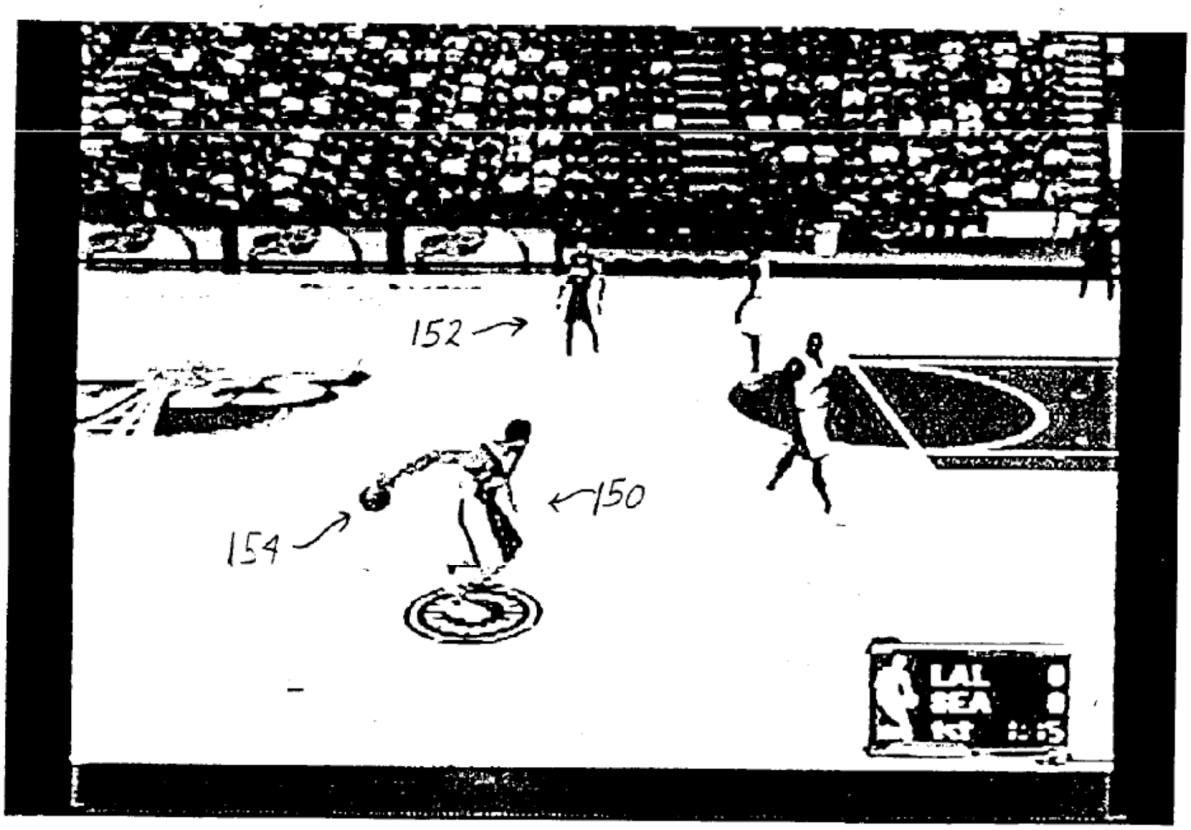


Ex. 1005, Fig. 5

125. It also would have been obvious to include a third person view because a POSITA would have understood that prior art sports and racing games commonly

Tesla Exhibit 1003.126

showed the game from the third-person perspective of a viewer from the stands in a basketball game, from behind the quarterback in a football game, or from behind the racing vehicle in a racing game, generating pictures that showed a first object of the application (*e.g.,*, a player avatar, vehicle, etc.) and its surroundings, such as a virtual field/court or racetrack including teammates and opponents around the user-controlled avatar. For example, the below illustration in a prior art patent illustrates a third-person view generated in a virtual basketball game, including a view that illustrates the surroundings of a simulated avatar 150 (*i.e.*, a first object):



Ex. 1008, Fig. 4

Tesla Exhibit 1003.127

126. In addition, a 2015 IGN article on the history of racing games describes that racing games have featured a third-person, behind-the-car view since 1976 and that a third-person, behind-the-car perspective is "common in racing games." Ex. 1012.004. The article describes and provides screenshots from multiple third-person racing games. An example from Pole Position (released in 1982) is reproduced below:



Ex. 1012.007

127. A 2013 IGN article on the history of football games identifies multiple examples of football games featuring a third-person point of view, including Madden NFL '95 (1994), NFL Blitz (1997) and Madden 2002 (2001). Ex. 1011.005-.008. An

example from Madden 2002 (released in 2001) is reproduced below, which shows a third person view of the avatars (*i.e.*, objects) from the sideline:



Ex. 1011.008

[10.2] wherein the third-person view includes a view from a perspective of an observer of: the first object of the first application program, the second object of the first application program, or the first object of the second application program.

Limitation [10.2]	Claimed Features
	3rd person POV = perspective from observer of **1st object**, **2nd object** or **1st object** of **2nd application**

128. Thomas discloses and it would have been obvious to a POSITA that the third-person view would be from the perspective of an observer of the first avatar

(*i.e.*, first object). As explained above, Figure 5 of Thomas discloses an aerial third person view of the football field from the perspective of an observer. And for the reasons explained above, it also would have been obvious to include a third person view from the perspective of an observer of a simulated avatar (*i.e.*, a first object), given that doing so was common in prior art sports video games. A POSITA would have understood that the perspective of an observer could be from an observer in the stands or watching the game on television.

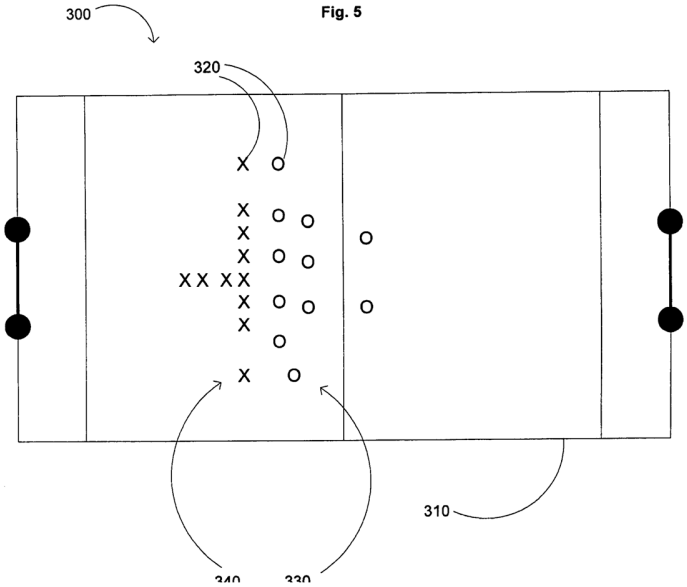
9. Claim 11: The system of claim 1, wherein the new one or more digital pictures depict a top-down view of the at least the portion of the surrounding of: the first object of the first application program, the second object of the first application program, or the first object of the second application program, and wherein the top-down view includes a view from above of: the first object of the first application program, the second object of the first application program, or the first object of the second application program.

Claim 11	Claimed Features
	New DP depict top-down POV of the surrounding of **1st object**, **2nd object** or **1st object** of **2nd application** + Top-down POV = perspective from above **1st object**, **2nd object** or **1st object** of **2nd application**

129. Thomas discloses and it would have been obvious to a POSITA to design the system in Thomas so that the new digital pictures depict a top-down view

Tesla Exhibit 1003.130

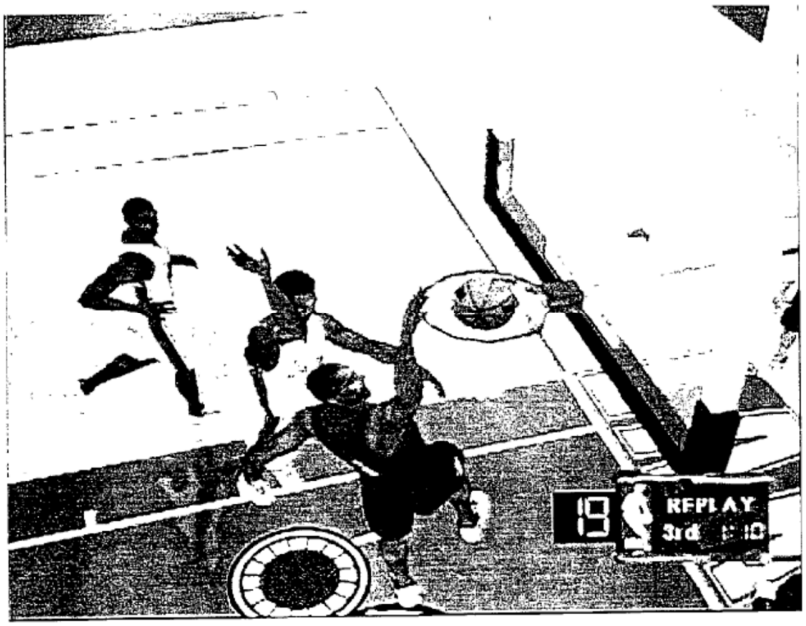
of the surroundings of an object. Figure 5 depicts a third-person, aerial, top-down view of a football game, including the surroundings of a first object (*i.e.*, the field and other avatars on the field). Ex. 1005, 8:66-9:3 ("Turning to FIG. 5, a diagram representing a video game interface 300 is shown. The interface 300 generally includes a graphical representation of a football field 310, and two sets of avatars 320, each representing a football team set to oppose each other."); *see* Section IX.B.8 above.



Ex. 1005, Fig. 5

130. It also would have been obvious to provide a top-down view that depicts the surroundings of a first object because a POSITA would have understood that prior art sports and racing games commonly depicted a top-down view of the game, including an object of the application and its surroundings, such as a virtual basketball court and other players around the object. For example, the below

illustration in a prior art patent illustrates a top-down view generated in a virtual basketball game, including an avatar (*i.e.*, first object) and its surroundings (*i.e.*, the court, basket, and other avatars):



Ex. 1009, Fig. 6

131. IGN's article on the history of racing games also demonstrates that it would have been obvious to provide a top-down view in other types of video games, as illustrated below in Gran Track 10 (released in 1974) and Rally X (released in 1980). Ex. 1012.003-.006. An example from Rally X is reproduced below:

Tesla Exhibit 1003.132



Ex. 1012.006

10. Claim 12

a. [12.1] The system of claim 1, wherein the one or more processors are further configured to perform at least: modifying: the first one or more instruction sets for operating the first object of the first application program, or a copy of the first one or more instruction sets for operating the first object of the first application program, and

Limitation [12.1]	Claimed Features
	Modifying **1st instructions** for **1st object**

132. Thomas discloses and it would have been obvious to a POSITA that the processors modify the instructions for operating a first object (such as an avatar).

Thomas describes that as a user interacts with the system, over the course of a game or over multiple game sessions, the simulator updates the instructions stored in the playbook profile. Ex. 1005, 10:27-32 ("If there is existing data in the playbook profile 500, then the simulator 120 will evaluate ***whether the data is up to date*** (action block 840). Preferably, ***data should evolve as the user's interaction evolves***. In other words, ***data recorded thirty games ago may have a lesser impact on the total data available***."). A POSITA would have understood and it would have been obvious that updating the data includes modifying the recorded instructions to reflect how the user has modified their response to a particular situation, defined by parameters such as the spatial position of avatars.

133. Thomas also explicitly discloses modifying the first one or more instruction sets for operating the first object. In particular, Thomas teaches that the user's profile can include a kick profile, which stores instructions regarding how a particular user kicks a football:

> Turning to FIG. 6, a kick profile 420 stores information as to how well a user kicks the ball during play. ***This includes how many times a user has kicked the ball, the minimum and maximum distances that a user has kicked the ball, and the total power used***. There is a kick profile for when a user kicks off, punts, and kicks field goals.

Ex. 1005, 12:42-47.

Tesla Exhibit 1003.134

134. As the user continues to interact with the system during learning mode, the simulator updates the kick profile. Each time the user kicks the ball, the simulator modifies the instruction set by increasing the kick count and/or modifying the kick power range. If the user exceeds their minimum or maximum power, the instruction sets are modified to change the range of power the simulator kicks the ball with in simulation mode:

> ***When recording data for a kick profile 420, the simulator 120 may check to see if the user's power has exceeded the minimum or maximum, increment the kick count, and add the power to the total power accumulated***. When the simulator 120 is in simulation mode, based on the user's profile 400, and the simulated user is in a kick situation, the simulator 120 may determine the power for kick based on the corresponding kick profile 420[.]
>
> ***
>
> Turning to FIG. 6, a kick profile 420 stores information as to how well a user kicks the ball during play. This includes how many times a user has kicked the ball, the minimum and maximum distances that a user has kicked the ball, and the total power used …. When the simulator 120 is in simulation mode, based on the user's profile 400, and the simulated user is in a kick situation, the ***simulator 120 may determine the power for kick based on the corresponding kick profile*** 420, e.g., whether the power should be above the average or below the average and/or whether the power should be ***a random value between the maximum and minimum powers recorded***.

Ex. 1005, 12:42-57.

Tesla Exhibit 1003.135

135. Thus, Thomas discloses modifying a first instruction set(s) directing the avatar to perform the kicking operation.

b. [12.2] wherein the executing the first one or more instruction sets for operating the first object of the first application program includes executing: the modified the first one or more instruction sets for operating the first object of the first application program, or the modified the copy of the first one or more instruction sets for operating the first object of the first application program, and

Limitation [12.2]	Claimed Features
	Executing modified **1st instructions**

136. As explained for limitation [12.1], Thomas discloses and/or renders obvious that the system would execute the most recent, updated instructions (the modified instructions if the instructions have been modified) for operating an avatar in a particular situation, such as kicking the ball using the modified kick profile. *See also* Ex. 1005, 10:30-32 ("***data recorded thirty games ago may have a lesser impact on the total data available***.").

c. [12.3] wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performing the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program includes the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performing one or more

Tesla Exhibit 1003.136

operations defined by: the modified the first one or more instruction sets for operating the first object of the first application program, or the modified the copy of the first one or more instruction sets for operating the first object of the first application program.

Limitation [12.3]	Claimed Features
	1st object, **2nd object** or **1st object** of **2nd application** autonomously perform modified instructions

137. This limitation can be satisfied through any combination of the listed options, including if the first object of the application (such as a first avatar) autonomously performs the one or more operations defined by the modified first one or more instruction sets. As explained in limitation [12.1], Thomas discloses and it would have been obvious that when the simulator executes the modified instruction set correlated with a particular situation (*e.g.*, executing the modified kick profile), as described above for limitations [5.2] and [12.2], the avatar would autonomously perform the operations defined by the modified instruction set. *See* Section IX.B.5.b and IX.B.10.a-b above.

11. Claim 13: The system of claim 1, wherein the new one or more digital pictures are generated at least by: generating one or more representations of one or more objects in the at least the portion of the surrounding of: the first object of the first application program, the second object of the first application program, or the first object of the second application program; and generating the new one or more digital pictures using the generated one or more representations of the one or more objects, and wherein the

Tesla Exhibit 1003.137

new one or more digital pictures depict: the generated one or more representations of the one or more objects, or a modified the generated one or more representations of the one or more objects.

Claim 13	Claimed Features
	Generating object representations (OR) + OR = representations of objects in the surroundings of the **1st object**, **2nd object** or **1st object** of **2nd application** OR used to generated **new DP** depicting OR

138. Thomas describes a simulation mode in which the system generates and the interface depicts a simulated avatar player (first object of the application), as well as the avatar's surroundings, such as a virtual field or court, and other players. For example, when the first object is an avatar representing a basketball player, the generated pictures include representations of defensive players around the avatar. *See* Ex. 1005, 8:11-14 ("During playback mode 600, the simulator 120 will monitor for a ***shoot event*** and analyze the location of ***the simulated user and the defensive coverage***."). Thomas therefore discloses generating one or more representations of objects in the surroundings of the first object (*i.e.*, simulated avatar), such as the defensive players, court, and basketball; *see also* Section IX.B.1.c above.

139. Thomas additionally discloses that the simulator uses the generated object representations to generate new digital pictures displayed on the interface screen, which depict the spatial position of other players around the simulated avatar

Tesla Exhibit 1003.138

(*i.e.*, the new digital pictures depict the generated one or more representations of the one or more objects in the surroundings of the first object). *See* Ex. 1005, 4:31-33 ("Another may include ***spatial position of avatars*** controlled by the user 665 ***on the interface screen***."), 8:11-14 ("***During playback mode*** 600, the simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and the defensive coverage***"), 8:66-9:3 ("Turning to FIG. 5, a diagram representing a video game interface 300 is shown. The interface 300 generally includes a ***graphical representation of a football field 310, and two sets of avatars 320, each representing a football team set to oppose each other.***").

12. Claim 14: The system of claim 1, wherein the first one or more instruction sets for operating the first object of the first application program include one or more information about one or more states of: the first object of the first application program, or a portion of the first object of the first application program.

Claim 14	Claimed Features
	1st instructions include state of **1st object**

140. Thomas discloses that the recorded instruction sets can include information about the state of the simulated avatar (first object), such as the avatar's archetype, the avatar's current circumstances in the game, whether the avatar possesses the ball (*e.g.*, for a pump fake), and whether the avatar is able to execute a given operation, such as kicking a football:

Tesla Exhibit 1003.139

Another type of parameter, in the case of a computer game involving one or more avatars, is the ***archetype of the avatar***. For example, an archetype can be defined by ***relative size, strength, and speed***. In the case of a basketball game, wherein avatars represent players of the game, archetypes can include true point guard, slasher, shooter, pure shooter, low post scorer, midrange scorer, and role player.

***

For a pump fake event, during the learning mode 550, the simulator 120 records how often a user pump fakes. For current circumstances, ***the simulator 120 records how covered the user was and user's location***. During playback mode 600, ***the simulator 120 will monitor for a shoot event and analyze the location of the simulated user*** and the defensive coverage. ***If criteria is met, then simulator 120 will invoke a pump fake***.

***

For an offensive rebound event, the simulator 120, during learning mode 550, ***will record which avatar grabbed the rebound*** and whether the user immediately passed the ball, which indicates a “reset” type rebound, or whether the user shoots the ball, which indicates a “shoot” type rebound. Again, current circumstances are recorded for the event. ***During playback 600, the simulator 120 will analyze the current circumstances during the occurrence of an offensive rebound event and invoke the proper response accordingly, e.g., whether to invoke a “shoot” rebound or a “reset” rebound***.

***

[A] kick profile 420 stores information as to how well a user kicks the ball during play…. When the simulator 120 is in simulation mode, based on the user's profile 400, ***and the simulated user is in a kick situation***, the simulator

Tesla Exhibit 1003.140

120 may determine the power for kick based on the corresponding kick profile 420, e.g., whether the power should be above the average or below the average and/or whether the power should be a random value between the maximum and minimum powers recorded.

***

Another sub-profile that can be included is the special moves profile 460. ***A user can invoke special moves if the user's avatars 320 are in possession of the ball.*** In such a situation, the simulator 120 may examine where the closest two defending avatars 320 are in relation to the ball carrying avatar 320 ("the ball carrier") on the football field 310.

***

For example, in the case of a football game, ***a player's avatar is running down the sideline*** and encounters an opposing player's avatar (an event). In one hundred (100) occurrences of the event, if the user spun left twenty (20) times, right thirty (30) times, and juked fifty (50) times, the simulator 120 will assign values of twenty (20), thirty (30), and fifty (50) to the respective responses.

Ex. 1005, 4:45-51, 8:8-15, 7:45-54, 12:42-57, 12:63-13:1, 6:16-27, 4:29-31.

13. Claim 15: The system of claim 1, wherein the first one or more digital pictures correlated with the first one or more instruction sets for operating the first object of the first application program include the first one or more digital pictures connected with the first one or more instruction sets for operating the first object of the first application program using at least one or more connections.

Claim 15	Claimed Features
	1st correlation includes connection between **1st DP** and **1st instructions**

Tesla Exhibit 1003.141

141. Thomas discloses a connection between the digital pictures and correlated instructions recorded in learning mode (*i.e.*, the first instruction set and first digital picture).

142. The ’585 patent specification describes that a connection may include or be associated with a numeric attribute or parameter, such as weight or occurrence count. ’585 Patent, 107:42-45 (“Connection 853 may include or be associated with a value such as a symbolic label or ***numeric attribute*** (*i.e*. ***weight***, cost, capacity, length, etc.).”); *see also id*., 109:57-60 (“Each of Connection 853z1 and Connection 853z2 may include or be associated with ***occurrence count, weight, and/or other parameter or data***.”).

143. Thomas discloses that in learning mode, the simulator records how often a user responds to a particular situation (first digital pictures) by invoking a certain action (instruction set for operating the first object). If a user encounters the same circumstances more than once, the user may invoke different actions on different occurrences. When the user has responded in multiple ways to a given situation (*i.e*., provided different instruction sets), the simulator assigns a numerical occurrence connection value between the situation (*i.e.*, the digital picture) and each correlated instruction set, where the connection is a numerical value reflecting how frequently the user instructed the avatar to perform the action:

Tesla Exhibit 1003.142

If there is more than one behavior recorded, the selection is based on statistical data (action block 1210).

***

During learning mode, at step 570, if the user has invoked more than one type of response, ***the simulator 120 can record each type of response and apply a statistical, or weighted, value to a response based on its statistical occurrence compared to other types of responses***. Turning to FIG. 11 a, at step 570, the simulator 120 will check if one or more user inputs have already been recorded for the current situation and/or event (decision block 575). If so, then ***the simulator 120 will associate one or more values with each different type of user input based on the statistical occurrence of each input compared to other inputs*** (action block 577). For example, in the case of a football game, a player's avatar is running down the sideline and encounters an opposing player's avatar (an event). In one hundred (100) occurrences of the event, if the user spun left twenty (20) times, right thirty (30) times, and juked fifty (50) times, the simulator 120 ***will assign values of twenty (20), thirty (30), and fifty (50) to the respective responses***….

Ex. 1005, 11:33-35, 6:5-27. Thomas further describes using the numerical occurrence values to select an instruction set during simulation mode from the various options for that particular situation:

If there is more than one value (decision block 630), then the actual second user may have invoked more than one type of response to the particular situation or event while being recorded. In such a case…***the***

Tesla Exhibit 1003.143

simulator 120 can select its response based on statistics or mathematical formula, e.g., based on the number of times the actual user invoked one response over another (action block 635). Such statistics may indicate tendencies demonstrated by the simulated user.

***

During simulation mode…when the simulated opponent's avatar is running down the sideline and encounters an opposing player's avatar and each of the above responses are given the respective range of values, ***the simulator 120 will randomly select a number from 1 to 100. If the number 53 is selected, then the avatar will juke (51-100 are associated with juke).***

Ex. 1005, 5:63-6:4, 6:28-41.

14. Claim 16

a. [16.1] The system of claim 1, wherein the system further comprising: a neural network that includes the first one or more digital pictures connected with the first one or more instruction sets for operating the first object of the first application program using at least one of: one or more connections, or one or more nodes, and

Limitation [16.1]	Claimed Features
	Neural network containing **1st DP** connected with **1st instructions** using a connection or node + Neural network = data structure converting input to output

Tesla Exhibit 1003.144

144. The ’585 Patent describes that a neural network comprises a data structure configured to compute an output from an input. ’585 Patent, 98:24-2 (“The features of known persons, objects, and/or their activities can be stored in ***any data structure or repository (i.e. neural network***….)”), 107:45-49 (“A computational model can be utilized to ***compute values from inputs*** based on pre-programmed or learned function or method. ***For example, a neural network may include one or more input neurons that can be activated by inputs***.”). For example, the specification describes that a neural network in the context of the ’585 Patent can include only a single layer of logic, meaning that the neural network can accept in an input and provide a corresponding output based on a single equation and/or comparison (*i.e.*, such as comparing two digital pictures). ’585 Patent, 107:51-53 (“Neural networks may range from those with ***only one layer of single direction logic*** to multi-layer of multi-directional feedback loops.”).

145. The ’585 Patent specification also describes that a “node” is synonymous with a knowledge cell (’585 Patent, 102:10-22, 111:61-63). A “knowledge cell” can “include any data structure or arrangement” and can facilitate storing correlations. *Id*., 20:41-42, 153:46-48 (“Therefore, knowledge cell may include any data structure or arrangement that can facilitate such storing.”).

146. Thomas discloses a neural network that uses connections and nodes to connect digital pictures and instruction sets. Thomas describes compiling digital

Tesla Exhibit 1003.145

pictures correlated with instruction sets into profiles (data structures), where each digital picture and user input is stored in a field or sub-profile (node). Ex. 1005, 4:67-5:5 ("[T]he simulator 120 will record the user's decision/input … and store the user's input in the field or sub-profile associated with the particular situation …."), 4:8-10 ("[T]he inputs for a user are stored in data structures herein referred to as user profiles, which may be individual data structures in a database 130."), 4:14-19 ("A user profile can contain a sub-profile related to one or more events 685 and/or situations 655 that occur within the computer program 110 during operation. An event 685 is a particular action occurring, whereas a situation 655 is the surrounding circumstances during which an event 685 may take place."); *see also* Section IX.B.13 above.

147. In simulation mode, Thomas describes that its neural network receives and analyzes digital pictures (inputs) and outputs instruction sets representing the correlated user instructions (outputs). Ex. 1005, 5:24-29 ("The stored data enables the analysis of a user's behavior, tendencies, and inclinations for a given event.… [W]hen the simulator 120 is in simulation mode, and ***a particular event and/or situation occurs, the simulator 120 can invoke a corresponding reaction*** based on the stored data….").

148. Thomas also describes that there are connections between digital pictures and instruction sets in its neural network. In particular, Thomas describes

Tesla Exhibit 1003.146

that stored digital pictures are connected to certain instruction sets by a connection, where the connection is a weighted numerical value reflecting the frequency of a given response for a particular first digital picture (*i.e.*, a picture stored during learning mode). Ex. 1005, 6:19-23 ("In one hundred (100) occurrences of the event, if the user spun left twenty (20) times, right thirty (30) times, and juked fifty (50) times, the simulator 120 ***will assign values of twenty (20), thirty (30), and fifty (50) to the respective responses***."), 6:39-41 ("[T]he simulator 120 will randomly select a number from 1 to 100. If the number 53 is selected, then the avatar will juke (51-100 are associated with juke)."), 11:33-35 ("If there is more than one behavior recorded, ***the selection is based on statistical data*** (action block 1210)."); *see also* Section IX.B.13 above.

b. [16.2] wherein the determining the first one or more instruction sets for operating the first object of the first application program based on the at least partial match between the new one or more digital pictures and the first one or more digital pictures includes applying the new one or more digital pictures to the neural network.

Limitation [16.2]	Claimed Features
	Applying neural network to **new DP**

149. Thomas discloses applying the new digital pictures generated during simulation mode to the neural network to determine how to operate an avatar (*i.e.*, determining the first instruction set). As I explained above, Thomas describes

Tesla Exhibit 1003.147

analyzing whether the situation generated during simulation mode partially matches a situation recorded during learning mode. *See* Section IX.B.1.d above. Thomas does so by applying the new digital pictures to the neural network to assess whether there is at least a partial match to the first digital picture stored in the network (for instance, the digital pictures concerning the defensive coverage in a basketball or football game). Thomas also discloses that the simulator can determine the instruction sets for operating the avatar using the connections (weighted values based on the occurrence count of a response):

> In one hundred (100) occurrences of the event, if the user spun left twenty (20) times, right thirty (30) times, and juked fifty (50) times, the simulator 120 ***will assign values of twenty (20), thirty (30), and fifty (50) to the respective responses***….
>
> ***
>
> ***During simulation mode, the simulator 120 can randomly invoke a response to a situation and/or event based on weighted statistical values of the stored inputs***…. In the example above, ***when the simulated opponent's avatar is running down the sideline*** and encounters an opposing player's avatar and each of the above responses are given the respective range of values, ***the simulator 120 will randomly select a number from 1 to 100. If the number 53 is selected, then the avatar will juke (51-100 are associated with juke).***

Ex. 1005, 6:19-41.

Tesla Exhibit 1003.148

15. Claim 17

a. [17.1] The system of claim 1, wherein at least some elements of the system are included in: a single device, or multiple devices, and

Limitation [17.1]	Claimed Features
	Single device or multiple devices

150. Thomas discloses that some elements of the claimed system can reside on a single or multiple devices, such as residing "on a single client console" (a single device) or on a "server 710 accessible by one or more client consoles" (multiple devices, including the server and the consoles). Ex. 1005, 19:35-43 ("Turning to FIG. 10, the computer program 110, simulator 120, and database 130, as shown in FIG. 1, ***can reside on a single client console***, such as the workstation 750 or a video game console 760, such as an Xbox or Playstation. Alternatively, or in addition, the computer program 110, the ***simulator 120, and database 130 can reside on a server 710 accessible by one or more client consoles*** 750/740….").

b. [17.2] wherein the one or more processors include: one or more microcontrollers, one or more computing devices, or one or more electronic devices, and

Limitation [17.2]	Claimed Features
	Processors = Microcontrollers, computing devices or electronic devices

Tesla Exhibit 1003.149

151. Thomas disclose the processors can include a video game console, which a POSITA would have understood to be an electronic device and/or a computing device. Ex. 1005, 19:35-39 ("Turning to FIG. 10, the computer program 110, simulator 120, ***and database 130, as shown in FIG. 1, can reside on a single client console***, such as the workstation 750 or a video game console 760, such as an Xbox or Playstation."), claim 19 ("The system of claim 13, ***wherein the computer video game is located in a local processor*** and the simulator and the database are located on portable memory device enabled to be directly connected to the local processor.").

c. [17.3] wherein the one or more memories include: one or more non-transitory machine readable media, one or more volatile memories, one or more non-volatile memories, one or more storage devices, or one or more storage systems, and

Limitation [17.3]	Claimed Features
	Memories = storage system, storage device or other embodiments

152. Thomas discloses that the memories can include a database stored on a portable memory device or server, i.e. a storage system and/or storage device. Ex. 1005, 19:40-43 ("[T]he ***computer program 110, the simulator 120, and database 130 can reside on a server*** 710 accessible by one or more client consoles …."), claim 19 ("The system of claim 13, wherein the computer video game is located in a local processor and the simulator and ***the database are located on portable***

Tesla Exhibit 1003.150

memory device enabled to be directly connected to the local processor."); *see also* Section B.1.b-c above.

d. [17.4] wherein the first one or more digital pictures include: one or more still digital pictures, or one or more motion digital pictures, and

Limitation [17.4]	Claimed Features
	1st DP = still DP or motion DP

153. Thomas discloses generating motion digital pictures during learning mode, which are depicted on the interface screen during learning mode, representing the circumstances in the video game. Ex. 1005, 10:19-36 ("During learning mode … The computer system initializes the game system features, e.g., controllers, animations, collisions, behaviors, etc. (action block 810).… After the game starts, and the game system components are updated, e.g., clocks, teams, cameras, etc. (action block 850), the simulator 120 monitors to determine whether a situation for special moves has occurred (action block 860)."). Thomas further describes that the system records features, values, and/or representations of portions of the digital pictures, such as the spatial position of players when a user invokes an action. *Id*., 5:15-18 ("[T]he user may choose to shoot a jump shot or split the defenders-responses that may be ***recorded in the profile 650 and linked with opponent's avatar's position.***"). As explained for claim 1, the features, values, and representations of portions of the digital pictures that Thomas records during

Tesla Exhibit 1003.151

learning mode (*i.e.*, the first digital picture) comprise "digital pictures" as that term is used in the '585 Patent claims. *See* Section IX.B.1.b above.

e. [17.5] wherein the new one or more digital pictures include: one or more still digital pictures, or one or more motion digital pictures, and

Limitation [17.5]	Claimed Features
	New DP = still DP or motion DP

154. Thomas discloses generating motion digital pictures, which are depicted on the interface screen during simulation mode, representing the circumstances in the video game. *See* Section VIII.B.1.c above. Thomas also describes analyzing values, features, and representations of at least portions of the motion digital pictures generated during simulation mode, such as the positions of defenders, and compares them to features, values, and/or representations of the first digital pictures recorded in learning mode. Ex. 1005, 8:11-14 ("During playback mode 600, the simulator 120 will monitor for a shoot event and analyze the location of the simulated user and the defensive coverage….").

f. [17.6] wherein the first application program includes: a video game, a computer game, a simulation program, a program including text processing, a program including number processing, a program including picture processing, a program including object processing, or a program, and

	Claimed Features

Tesla Exhibit 1003.152

Limitation [17.6]	**1st application** = video game, computer game or other embodiment

155. Thomas discloses that the simulation can be applied to a video game or computer game (first application program). Ex. 1005, 19:49-53 ("Multiple users can compete against each other either on a single console 750/740 or from multiple consoles 750/740. For example, a first user can compete on a ***first video game console*** 760 against a second user competing on a second video game console 760."); *id*., 8:55-62 ("***Another example embodiment*** of a computer system 100 is a video game system…. ***NFL 2K5 includes a computer program 110 in the form of an video game to be executed on a video game console***….").

g. [17.7] wherein the second application program includes: a video game, a computer game, a simulation program, a program including text processing, a program including number processing, a program including picture processing, a program including object processing, or a program, and

Limitation [17.7]	**Claimed Features**
	2nd application = video game, computer game or other embodiment

156. Thomas discloses that the simulation can be applied to a video game or computer game played on multiple user consoles, including a video game on a second console (second application program). Ex. 1005, 19:49-53 ("Multiple users can compete against each other either on a single console 750/740 or from multiple

Tesla Exhibit 1003.153

consoles 750/740. For example, a first user can compete on a first video game console 760 against a second user competing on a ***second video game console*** 760."). A POSITA would have understood Thomas discloses a second application program, which comprises a copy of the computer game and/or video game on the same type of console (Xbox), or a version of the video game for a different type of console (*i.e.*, Playstation or PC).

h. [17.8] wherein an instruction set of the first one or more instruction sets for operating the first object of the first application program includes at least one of: only one instruction, a plurality of instructions, one or more inputs, one or more commands … or one or more information.

Limitation [17.8]	Claimed Features
	1st instructions = instructions, input or other embodiment

157. This limitation is satisfied if the instruction set contains any one of: an instruction, a plurality of instructions, one or more inputs, or one or more commands. Thomas describes instruction sets including at least one instruction, input or command directing an avatar to perform an action, such as an instruction or command to shoot a basketball by providing an input, such as pressing a button or combination of buttons. Ex. 1005, 4:67-5:1 ("[S]imulator 120 will record the user's decision/input, such as a pressed button, combination of buttons…."), 5:13-17 ("For example, in the game of basketball, if a user is controlling a basketball player near

the three-point line and sees two defenders, one on each side, the ***user may choose to shoot a jump shot … that may be recorded*** in the profile ….”), 5:26-29 (“[W]hen the simulator 120 is in simulation mode, and a particular event and/or situation occurs, the simulator 120 can ***invoke a corresponding reaction based on the stored data***, thereby simulating the user”).

16. Claim 18

a. [18.1] The system of claim 1, wherein the first one or more digital pictures are: one or more whole digital pictures, one or more representations of one or more whole digital pictures, one or more portions of at least one digital picture, one or more representations of one or more portions of at least one digital picture, one or more features, one or more representations of one or more features, one or more collections of pixels, or one or more collections of values, and

Limitation [18.1]	Claimed Features
	1st DP = representation of a portion of a DP, one or more features, or one or more collections of values

158. This claim is satisfied if the first digital pictures comprise any of the listed embodiments. As I explained above in relation to limitation [1.2], Thomas discloses that the first digital pictures recorded during learning mode are at least representations of a portion of a digital picture, one or more features, or one or more collections of values. *See* Section IX.B.1.b above.

Tesla Exhibit 1003.155

b. [18.2] wherein the new one or more digital pictures are: one or more whole digital pictures, one or more representations of one or more whole digital pictures, one or more portions of at least one digital picture, one or more representations of one or more portions of at least one digital picture, one or more features, one or more representations of one or more features, one or more collections of pixels, or one or more collections of values.

Limitation [18.2]	Claimed Features
	New DP = whole DP, representation of a portion of a DP, one or more features, or one or more collections of values

159. This claim is satisfied if the new digital pictures comprise any of the listed embodiments. During simulation mode, Thomas discloses generating new whole digital pictures, displayed on the video game interface, as I explained for claim 1. *See* Section IX.B.1.c above. As also explained for claim 1, Thomas describes that the processor receives at least representations of a portion of the digital picture, one or more features, or one or more collections of values for the new digital pictures, which are compared to those stored during learning mode. *See* Section IX.B.1.c-d above.

17. Claim 19

160. Independent claim 19 is a method claim that corresponds to system claim 1. My analysis above explaining how Thomas discloses the limitations of claim 1 applies to the limitations of claim 19 as set forth in the chart below.

Claim 19	Corresponding Analysis for Claim 1
[19.1] A method implemented using a computing system that includes one or more processors, the method comprising: accessing one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program;	*See* analysis for limitations [1.1] and [1.2] above. Sections IX.B.1.a-b.
[19.2] receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program;	*See* analysis for limitation [1.3] above. Section IX.B.1.c.
[19.3] determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures; and	*See* analysis for limitation [1.4] above. Section IX.B.1.d.
[19.4] at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.	*See* analysis for limitation [1.5] above. Section IX.B.1.e.

Tesla Exhibit 1003.157

18. Claim 20

a. [20.1] A system comprising: means for processing; and

Limitation [20.1]	Claimed Features
	System + means for processing

161. As I discussed above in regards to claim construction, the corresponding structure for this limitation is a processor or processor circuit. Thomas discloses a system including a processor. Ex. 1005, claim 19 ("The system of claim 13, wherein the computer video game is located in a ***local processor*** and the simulator and the database are located on portable memory device enabled to be directly connected to the ***local processor***."). Thomas also discloses running the simulator on a video game console, which a POSITA would have understood includes a processor. *Id*., 8:55-62 ("***Another example embodiment*** of a computer system 100 is a video game system … NFL 2K5 includes a computer program 110 in the form of an video game to be executed on a ***video game console***…."); Ex. 1019 (discusses how video game consoles have included processors since the 1970s), Ex. 1020 (discusses Playstation 1 (released 1994) and Playstation 2 (released 2000) include processors).

b. [20.2] means for storing machine readable code that, when executed by the means for processing, causes the means for processing to perform at least:

	Claimed Features

Tesla Exhibit 1003.158

Limitation [20.2]	Means for storing

162. As I discussed above in regards to claim construction, the corresponding structure for this limitation is storage or a memory. Thomas discloses storing the code for operating the simulator and database on a memory device that can be directly connected to the processor. Ex. 1005, claim 19 ("The system of claim 13, wherein the computer video game is located in a local processor and the simulator and the database are ***located on portable memory device*** enabled to be directly connected to the local processor."). Thomas also discloses running the simulator on a video game console, which a POSITA would have understood includes a memory that stores code. *Id*., 8:55-62 ("***Another example embodiment*** of a computer system 100 is a video game system … NFL 2K5 includes a computer program 110 in the form of an video game to be executed on a ***video game console***…."); Ex. 1019 (discusses how video game consoles have included memories since the 1970s); Ex. 1021 (discusses history of memory in video game consoles starting in the 1970s).

c. [20.3] accessing one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program;

163. As I described above for claim 1, Thomas discloses accessing memories that store digital pictures correlated with instruction sets for operating a first object

Tesla Exhibit 1003.159

of the application. Moreover, the claims of Thomas confirm that the processor in Thomas performs the accessing function. *See* Ex. 1005, claims 1-2 (requiring that the processor execute a first process and second process, which include "retrieving the one or more data fields" from the memory).

d. [20.4] receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program;

164. As I discussed above for limitation [1.3], Thomas discloses receiving or generating new digital pictures depicting part of the surrounding of the first object of the application. Moreover, the claims of Thomas confirm that the processor in Thomas performs the generating and receiving functions. *See* Ex. 1005, claims 1-2 (requiring that the processor execute a first process and second process, which include "monitoring the computer game for one or more events…").

e. [20.5] determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures; and

165. As I discussed above for limitation [1.4], Thomas discloses determining the instruction sets for operating the first object of the application based at least partial match between the digital pictures recorded during learning mode and the digital pictures generated/received during simulation mode. Moreover, the claims of

Tesla Exhibit 1003.160

Thomas confirm that the processor in Thomas performs the determining function. *See* Ex. 1005, claims 1-2 (requiring that the processor execute a first process and second process, which include "invoking a response …").

f. [20.6] at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.

166. As I discussed above for limitation [1.5], Thomas discloses executing the first instruction set, causing the first object of the first application to perform the defined operations. Moreover, the claims of Thomas confirm that the processor in Thomas performs the executing function. *See* Ex. 1005, claims 1-2 (requiring that the processor execute a first process and second process, which include "invoking a response …").

C. Grounds 3 and 4: Anticipation of Claims 1, 8-9, 12, and 16-20 Based on Buehler; Obviousness of Claim 17 Based on Buehler

167. As set forth in detail below, claims 1, 8-9, 12, and 16-20 of the '585 Patent are anticipated by Buehler. The discussion below specifies where each element of the aforementioned claims is found in Buehler, and includes a detailed explanation of the significance of the quotations and citations.

Tesla Exhibit 1003.161

1. Claim 1

a. [1.1] A system comprising: one or more processors; and

Limitation [1.1]	Claimed Features
	System including processors

168. Buehler discloses a robotic system, where the robot includes a processor. Ex. 1006, 8:10-1 ("Robots … include one or more appendages with end-effectors for manipulating objects … and (as part of the robot itself or associated therewith as part of a robot system) a control system…."), 4:51-53 ("The image-processing module and the control system comprise sets of instructions stored in memory and executable by ***a processor***.").

b. [1.2] one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program,

Limitation [1.2]	Claimed Features
	Memories store a **1st correlation** + First correlation = DP + associated instructions + DP = Digital pictures

169. Buehler discloses a memory that stores template digital pictures of objects correlated with routines for operating the robot, such as picking up the object.

170. More specifically, Buehler discloses "a ***data storage medium*** (e.g., system memory or a non-volatile storage medium such as, e.g., a hard drive) storing a plurality of ***visual models and associated object representations***," which can

Tesla Exhibit 1003.162

include template digital pictures or features extracted from a digital image. Ex. 1006, 5:27-31. *See also*:

> In some embodiments, the learning process involves applying a number of ***different visual models, each including one or more vision algorithm and associated object representations***, and selecting the visual model that works best for the object at hand. ***For example, some objects may be best recognized based on features extracted from the images (such as, e.g., parameters describing a certain shape), whereas others may be identified based on straightforward image correlation with a template image*** of the object. Which visual model is most suitable for the object may be determined based on user feedback and/or self-assessment of the robot. In addition to ***saving the selected visual model in memory for later use***, the robot may also obtain non-visual information about the object and store it in association with the visual model.
>
> ***
>
> The system memory 224 contains partitions and/or buffers to store, ***in digital (pixel) form, images obtained by the cameras*****,** as well as portions of ***one or more databases for representing task and object representations as described*** below.

Ex. 1006, 3:7-21, 10:22-26.

> [R]elevant information about the ***accomplished task … may be stored along with the visual model of the object***. When the robot later, in execution mode, detects an object of the same type, it may utilize the stored information to execute an object-specific pick-up routine.

Id., 3:36-42.

Tesla Exhibit 1003.163

171. Buehler discloses storing a routine, such as an "object-specific pick-up routine," (instruction set) "along with" (correlated with) the template digital pictures:

> [R]elevant information about the accomplished task, e.g., ***the established grasp point and motions of the gripper and/or robot arm, may be stored along with the visual model of the object***. When the robot later, in execution mode, ***detects an object of the same type, it may utilize the stored information to execute an object-specific pick-up routine***.
>
> ***
>
> In some embodiments, the ***robot stores a plurality of action templates***, e.g., in the form of prototypes for manipulations, and the robot and/or user may select one of the templates to indicate a present training objective. For example, the user may select an action template from a list, or ***the robot may select the action template automatically based on sensor input and context. If the robot detects, for instance, a conveyor belt, it may select an action for picking up objects from the conveyor belt.***

Ex. 1006, 3:36-42, 22:35-44.

172. Buehler further discloses generating a virtual representation of the robot and its end-effector to simulate the performance of tasks using the stored routines (*i.e.*, instruction sets), such as grasping an object. The virtual robot and end-effector simulate the robot's operations based on a stored instruction set associated with the target object (operating a first object of the first application according to a first instruction set). A POSITA would have understood that the virtual

Tesla Exhibit 1003.164

representation of the robot and its end-effector comprises a first object of a first application, where the first application program is the simulation in training mode that demonstrates expected performance of the task using the stored instructions:

> [T]he robot may ***overlay virtual grippers on the image*** (or video stream) of the object to indicate where the grasp point would be, given the current arm location and orientation (step 606). Thus…the robot can give the user a sense of the likelihood of a successful pick-up.
> ***
> In this method, the robot selects an object in the field of view and displays the selection (including the camera view of the selected object) on a robot-controlled visual display, ***overlays a representation of an end-effector onto the camera view, and displays performance of an action on the object by the end-effector***…. In some embodiments, the robot ***selects the action from a plurality of stored action templates.***

Ex. 1006, 21:47-54, 6:31-48.

c. [1.3] wherein the one or more processors are configured to perform at least: receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program;

Limitation [1.3]	Claimed Features
	Processors receiving/generating **new DP** including the surroundings of **1st object** of **1st application**, **2nd object** of **1st application** or **1st object** of **2nd application**

Tesla Exhibit 1003.165

173. As I discussed above for limitation [1.1], Buehler discloses that its system is performed by one or more processors. Buehler additionally discloses generating new digital pictures by capturing images with the camera. Buehler further describes that the new digital pictures captured by the camera depict the surroundings of the virtual representation of the robot's end-effector (a first object of the application program) because the system overlays the virtual grippers onto the captured image in the simulation program:

> For example, the ***robot may use a camera to acquire images of its environment (constituting real-world data), and overlay computer graphics onto the image*** (such as, e.g., an object outline in the camera image, indicating selection of the outlined object by the robot, and/or ***a representation of the robot's end-effector***, indicating the position of the end-effector relative to the object).
>
> ***
>
> [T]he robot may overlay ***virtual grippers on the image (or video stream) of the object to indicate where the grasp point would be, given the current arm location and orientation (step 606)***.

Ex. 1006, 2:6-12, 21:47-50.

d. [1.4] determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures; and

Limitation [1.4]	Claimed Features
	1st instructions determined by partial match between **1st DP** and **new DP**

Tesla Exhibit 1003.166

174. Buehler discloses comparing new digital images to stored image data (first digital images). In particular, Buehler describes comparing the new images captured by the robot camera to stored image templates to determine if there is at least a partial match, and determining an instruction set (*i.e.*, an object specific routine) if there is a match:

> The method includes causing a robot to execute the following steps: selecting an object in a camera field of view, ***using a visual model to computationally identify the object based on a stored representation thereof associated with the model, determining whether an object-specific manipulation routine is stored in association with the stored representation***, and, if so, ***executing the object-specific manipulation routine***….
>
> ***
>
> More generally, ***representations of objects, or classes of objects, may have specific routines for picking up or otherwise manipulating the object associated with them….*** Thus, the robot's end-effector would undergo different motions to open a cylindrical container with a lid than it would to open a cardboard box. ***The robot may learn these special routines during training, and associate them with particular visual representations***.

Ex. 1006, 7:6-15, 22:13-30.

175. Buehler further describes that the routine determined based on matching the new digital pictures to the stored first digital pictures is used to operate

Tesla Exhibit 1003.167

a visual representation of the robot (*i.e.*, the first object of the application) to simulate the robot's performance of a task, such as grasping an object:

> Following the user's approval of the selected visual model, the robot may overlay ***virtual grippers on the image (or video stream) of the object to indicate where the grasp point would be, given the current arm location and orientation (step 606)***. Thus, without the need to actually lower the gripper and bring it in contact with the object … ***the robot can give the user a sense of the likelihood of a successful pick-up***.

Ex. 1006, 21:43-54.

e. [1.5] at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.

Limitation [1.5]	Claimed Features
	Executing **1st instructions** by **1st object**, **2nd object** or **1st object** of **2nd application** autonomously

176. As explained above, Buehler discloses that the robot determines the appropriate object-specific routine based on at least partial match between new images captured by the robot's camera and stored image data. Buehler further discloses that when the robot detects a match, the system overlays a virtual representation of its end-effector on the newly captured images and simulates performing a task using the correlated object-specific routine (*i.e.*, the system

executes the instruction set such that the first object autonomously performs the operation defined by the instruction set):

> Following the user's approval of the selected visual model, ***the robot may overlay virtual grippers on the image (or video stream) of the object to indicate where the grasp point would be, given the current arm location and orientation***….
>
> ***
>
> In this method, the robot selects an object in the field of view …, ***overlays a representation of an end-effector onto the camera view, and displays performance of an action on the object by the end-effector***.

Ex. 1006, 21:46-50, 6:31-37.

2. Claim 8

a. [8.1] The system of claim 1, wherein the one or more processors are further configured to perform at least:

177. As I discussed above in relation to claim 1, Buehler discloses a system including processors.

b. [8.2] modifying at least one of: the first one or more digital pictures, a copy of the first one or more digital pictures, the new one or more digital pictures, or a copy of the new one or more digital pictures, and

Limitation [8.2]	Additional Features to Claim 1
	Modifying **1st DP** or **new DP**

178. Buehler discloses overlaying digital images with computer-generated graphics, such as an outline around an object of interest. In training mode, a user can

Tesla Exhibit 1003.169

modify and manipulate the computer generated graphics (*i.e.*, the digital picture) to better correspond to the object of interest:

> For example, the robot may show an image (or video, i.e., image stream) of its environment, ***and indicate selection of an object in the field of view by drawing an outline of the object in the image***…. [T]he user may have ***the opportunity to provide feedback by manipulating the computer-generated graphics***. For example, if the object outline does not encompass the entire object or, conversely, includes areas not corresponding to the object, ***the user may drag vertices of the outline to a desired position to achieve better correspondence.***

Ex. 1006, 12:43-55.

179. Buehler further describes that the system stores the modified images and, on subsequent use, compares new images captured by the camera's sensors to the modified images. The modified images accordingly correspond to the claimed first digital pictures:

> Sensor input 500 includes, typically, a video stream of images from one or more robot cameras, e.g., from a camera integrated into an end-effector and pointing at the object of interest … ***The contextual information 502 relevant to object recognition includes a previously acquired image or video stream annotated with labeled pixels*** (e.g., contour outlines, highlights, etc.) that indicate which pixels belong to the object and which do not.

Ex. 1006, 17:28-38.

Tesla Exhibit 1003.170

c. [8.3] wherein the at least partial match between the new one or more digital pictures and the first one or more digital pictures is determined at least by … (iii) determining at least partial match between the new one or more digital pictures and the modified the first one or more digital pictures, [or]….

Limitation [8.3]	Additional Features to Claim 1
	Partial match includes modified DP

180. This limitation provides eight options for determining a partial match. The limitation is satisfied if the partial match is determined by any one of the specified options. My analysis below is based on Buehler's disclosure of (iii), determining a partial match between the modified first digital picture and the new digital picture.

181. Buehler discloses that the above-described modified digital images captured in training mode (modified first one or more digital pictures) are stored and compared to new images generated and received by the robotic system. A partial match is determined by comparing the modified images to the newly generated images:

> Sensor input 500 includes, typically, a video stream of images from one or more robot cameras, e.g., from a camera integrated into an end-effector and pointing at the object of interest … ***The contextual information 502 relevant to object recognition includes a previously acquired image or video stream annotated with labeled pixels*** (e.g., contour outlines, highlights, etc.) that indicate which pixels belong to the object and which do not.

Tesla Exhibit 1003.171

***

The detection algorithm 508 ***compares real-time camera images against the template and determines, based on a similarity metric, whether the real-time image contains the object of interest***.

Ex. 1006, 17:28-38, 18:50-54.

3. Claim 9

a. [9.1] The system of claim 1, wherein the new one or more digital pictures depict a first-person view of the at least the portion of the surrounding of: the first object of the first application program, the second object of the first application program, or the first object of the second application program,

Limitation [9.1]	Additional Features to Claim 1
	New DP depict 1st person POV of the surrounding of **1st object**, **2nd object** or **1st object** of **2nd application** 1st person POV = perspective from **1st object**, **2nd object** or **1st object** of **2nd application**

182. The '585 Patent describes that a first-person view "may include a first-person view or perspective such as a view through an avatar's eyes that shows objects around the avatar, but does not typically show the avatar itself. First-person view may sometimes include the avatar's hands, feet, other body parts, and/or objects that the avatar is holding." '585 Patent, 82:3-8.

183. Buehler discloses newly generated digital pictures, captured from the perspective of the robot, including pictures from cameras in the robot's chest and wrists:

Tesla Exhibit 1003.172

> Further, the robot 100 includes five cameras. It has ***one camera 109 in each of its two wrists so that the robot 100 can "see" objects it is about to pick up*** and adjust its grippers 106 accordingly. Further, ***it has two cameras 110, side by side, in its chest that provide a wide-angle view of the workspace and allow the robot 100 to visually detect equipment or objects to be manipulated***.

Ex. 1006, 8:57-63. The new digital pictures depict the surroundings of the robot, such as objects located near the robot (e.g., objects it is about to pick up), and also depict surroundings of the virtual representation of the robot in the simulation program (*i.e.*, the first object).

184. As explained above, Buehler generates new digital pictures that include the objects captured by the robot camera in conjunction with a virtual representation of the robot end effector. *Id.*, 21:46-50 ("Following the user's approval of the selected visual model, ***the robot may overlay virtual grippers on the image (or video stream) of the object to indicate where the grasp point would be….***"). A POSITA would have understood that the new images generated by the robot including the virtual end effector would depict at least a portion of the robot and end-effector's surroundings (such as the object the robot is intended to interact with) from the first person view of the virtual representation of the robot. This corresponds to a first person view as described in the '585 Patent because it would provide a view from the virtual robot's chest that does not depict the robot itself (only its end effector), and otherwise only depicts its surroundings (i.e., an object to pick up).

Tesla Exhibit 1003.173

b. [9.2] and wherein the first-person view includes a view from a perspective of: the first object of the first application program, the second object of the first application program, or the first object of the second application program.

Limitation [9.2]	Additional Features to Claim 1
	1st person POV = perspective from **1st object**, **2nd object** or **1st object** of **2nd application**

185. As I explained above for limitation [9.1], a POSITA would have understood that the newly captured digital picture(s) depict the targeted object from the first person perspective of the robot (the first object of the application), as viewed from the robot's chest or wrist.

4. Claim 12

a. [12.1] The system of claim 1, wherein the one or more processors are further configured to perform at least: modifying: the first one or more instruction sets for operating the first object of the first application program, or a copy of the first one or more instruction sets for operating the first object of the first application program, and

Limitation [12.1]	Claimed Features
	Modifying **1st instructions** for **1st object**

186. As discussed above in regards to limitation [1.1], Buehler discloses that the system includes processors. Buehler discloses a training process where the user and/or the robot iteratively modify the instructions for grasping an object with the

Tesla Exhibit 1003.174

virtual end-effector (first object of the first application program) until the user is satisfied with the simulated performance of the task:

> If the virtual gripper is not positioned around the proper part of the object (as determined in step 608), ***the user may adjust the arm*** to roughly establish the proper grasp point (step 610). The robot may then fine-tune the arm location to optimize the gripper location relative to the rough grasp point on the object (step 612). ***This process of coarse and fine adjustments of the gripper by the human trainer and the robot, respectively, may be iterated until the user is satisfied with the grasp point and gripper orientation shown in the visual display***.
>
> ***
>
> For example, the user may provide feedback by moving a robot appendage. ***The display of the robot's performance of the action on the object by the end-effector and feedback are repeated until the action is performed correctly on the display***. If the end-effector is a gripper and the action includes gripping the object, the feedback may indicate whether the displayed grippers have gripped a proper part of the displayed object. In some embodiments, the robot selects the action from a plurality of stored action templates.
>
> ***
>
> Thus, the robot's end-effector would undergo different motions to open a cylindrical container with a lid than it would to open a cardboard box. The robot may learn these special routines during training, and associate them with particular visual representations. ***Alternatively, the routines may be programmed into the robot at the outset and altered, as necessary, during training as noted above***.

Tesla Exhibit 1003.175

***

The user interface includes **a *robot-controlled visual display for visually displaying performance of the action on the object in a camera view thereof using a robot-generated graphic representation of the end-effector***, and one or more input devices for receiving feedback from the trainer indicating whether the action is performed correctly on the display…. ***The control system is adapted to modify performance of the action on the display in response to the feedback until the action is performed correctly on the display***.

Ex. 1006, 21:54-64, 6:31-48; *id*., 22:26-32, 6:58-7:3.

b. [12.2] wherein the executing the first one or more instruction sets for operating the first object of the first application program includes executing: the modified the first one or more instruction sets for operating the first object of the first application program, or the modified the copy of the first one or more instruction sets for operating the first object of the first application program, and

Limitation [12.2]	Claimed Features
	Executing modified **1st instructions**

187. Buehler discloses that, during training mode, after the user and/or system modify the instructions, the program simulates operating the virtual end-effector and robot using the modified instruction set (executes the modified first instruction set). *Id.*, 6:37-43 ("Based on the display, feedback is provided to the robot, indicating whether the action is performed correctly on the display….***The display of the robot's performance of the action on the object by the end-effector***

Tesla Exhibit 1003.176

and feedback are repeated until the action is performed correctly on the display."); 6:65-7:1 ("***The control system is adapted to modify performance of the action on the display in response to the feedback until the action is performed correctly on the display***."); 21:54-64 ("If the virtual gripper is not positioned around the proper part of the object…the user may adjust the arm to roughly establish the proper grasp point (step 610). ***The robot may then fine-tune the arm location*** to optimize the gripper location…. ***This process of coarse and fine adjustments of the gripper by the human trainer and the robot, respectively, may be iterated until the user is satisfied*** with the grasp point and gripper orientation shown in the visual display.").

c. [12.3] wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performing the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program includes the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performing one or more operations defined by: the modified the first one or more instruction sets for operating the first object of the first application program, or the modified the copy of the first one or more instruction sets for operating the first object of the first application program.

Limitation [12.3]	Claimed Features
	1st object, **2nd object** or **1st object** of **2nd application** autonomously perform modified instructions

188. This limitation can be satisfied through any combination of the listed options. My analysis is based on Buehler's disclosure of the first object of the first application program autonomously performing the one or more operations defined by the first one or more instruction sets for operating the first object of the first application program including the first object of the first application program autonomously performing one or more operations defined by: the modified the first one or more instruction sets for operating the first object of the first application program.

189. Buehler discloses that the first object of the first application (the virtual end-effector) autonomously performs one or more operations defined by the modified instruction set. As I explained above in regards to limitation [12.2], after the user and/or system modify the instructions, the virtual representation of the robot and its end-effector (the first object) autonomously simulates performing the task based on the modified routine (performs the modified first instruction set for operating the first object of the first application). *Id.*, 6:40-43 ("The display of the robot's performance of the action on the object by the end-effector and feedback are repeated until the action is performed correctly on the display.").

5. Claim 16

a. [16.1] The system of claim 1, wherein the system further comprising: a neural network that includes the first one

Tesla Exhibit 1003.178

or more digital pictures connected with the first one or more instruction sets for operating the first object of the first application program using at least one of: one or more connections, or one or more nodes, and

Limitation [16.1]	Claimed Features
	Neural network containing **1st DP** connected with **1st instructions** using a connection or node + Neural network = data structure converting input to output

190. As I explained above in relation to the Thomas patent, a neural network comprises a data structure configured to compute an output from an input, and a node is any data structure that stores the requisite data. *See* Section IX.B.14.a above (Thomas Limitation 16.1). Buehler describes storing instruction sets that are connected to visual models of target objects in a data structure. Ex. 1006, 3:36-42 ("[R]elevant information about the ***accomplished task…may be stored along with the visual model of the object***. When the robot later, in execution mode, detects an object of the same type, it may utilize the stored information to execute an object-specific pick-up routine."), 10:52-11:18 ("To facilitate the robot's autonomous performance of tasks in execution mode, the task-execution module 268 may utilize representations of tasks, and of objects to which the tasks pertain, that have been specified during training mode…. Of course, the prototype library 272, instance database 274, and user models 276 may, during operation, be loaded into system

Tesla Exhibit 1003.179

memory 224."). The stored tasks and template digital pictures in Buehler are connected via a connection because they are stored together such that a particular task can be output when the system detects that a new digital picture matches a template picture, as described below.

191. Buehler further describes that its system converts input digital pictures obtained via its cameras into output instructions for the robot and performing a task or simulating performance of a task. *Id.*, 7:6-15 ("The method includes…selecting an object in a camera field of view, using a visual model to computationally ***identify the object based on a stored representation thereof***…determining whether ***an object specific manipulation routine is stored in association with the stored representation***, and, if so, executing the object-specific manipulation routine."). Thus, Buehler describes that its neural network includes first digital pictures connected with first instruction sets for operating a first object (i.e., the virtual robot) using at least one connection and one node.

192. A POSITA would have understood it has been known the art since at least 1993 to apply a neural network to the operation of an autonomous mobile robot with visual sensors, such as cameras. Ex. 1015.010 ("In this dissertation I develop techniques which enable artificial neural networks (ANNs) to learn the visual processing required for mobile robot guidance. The power and flexibility of these techniques are demonstrated in two domains, wheeled vehicle navigation, and

Tesla Exhibit 1003.180

legged robot foot positioning."), 1015.011 ("To navigate effectively in a complex environment, a mobile robot must be equipped with sensors for gathering information about its surroundings. In this work, I have chosen imaging sensors as the primary source of information because of their ability to quickly provide dense representations of the environment. The imaging sensors actually employed include ***color and black-and-white video cameras***, a scanning laser rangefinder and a scanning laser reflectance sensor."). Pomerleau describes a system that can be used to learn how to position a foot appendage of a robot, which a POSITA would have understood to correspond to the robot in Buehler learning how to position a robot's end-effort appendage. *Id.*, 1015.018 ("The crucial task in this domain is to ***precisely position the foot of the robot*** in order to anchor it without damaging either the space station or the robot."). Like the neural network described in Buehler, Pomerleau describes a neural network with inputs comprising images that the robot processes to extract important features and outputs comprising motor commands to operate the robot. *Id.*, 1015.022 ("As a result of these difficulties with preprocessing images for mobile robot guidance, the philosophy in this work has been to keep the system general by performing a minimal amount of preprocessing. The idea is to let the network learn not only to combine important features to determine the most appropriate response in the current situation, but ***also to determine what the most important features are and how to detect them in the input image***."), 1015.029("In

the case of mobile robot guidance, ***the network's output represents a specific motor response along a continuum.*** For instance, in autonomous driving, the network's output represents the steering direction appropriate for the current situation, ranging continuously from sharp left to sharp right. Similarly, in the foot placement task for the walking robot, the network's output represents the displacement of the robot's foot from a 'safe' position to step."). Pomerleau describes a system where a car learns how to drive based on sensor data through correlating images with the instructions of a human driver, which a POSITA would understand to correspond to the training mode described in Buehler. *Id.*, 1015.050 ("Autonomous driving has a substantial advantage as a domain for supervised learning: the ready availability of a teaching signal in the form of the human driver's current steering direction. In theory it should be possible to teach a network to imitate a person as they drive using the current sensor image as input and the person's current steering direction as the desired output.").

b. [16.2] wherein the determining the first one or more instruction sets for operating the first object of the first application program based on the at least partial match between the new one or more digital pictures and the first one or more digital pictures includes applying the new one or more digital pictures to the neural network.

Limitation [16.2]	Claimed Features
	Applying neural network to **new DP**

Tesla Exhibit 1003.182

193. As I explained above, when new digital pictures are input into the neural network in Buehler, the neural network outputs the correlated instruction set for completing a task upon detecting at least a partial match between the new digital pictures and the first digital pictures (i.e., the templates stored in the neural network). Ex. 1006, 7:6-14 ("The method includes…selecting an object in a camera field of view, using a visual model to computationally ***identify the object based on a stored representation thereof***…determining whether ***an object-specific manipulation routine is stored in association with the stored representation***, and, if so, executing the object-specific manipulation routine.").

6. Claim 17

a. [17.1] The system of claim 1, wherein at least some elements of the system are included in: a single device, or multiple devices, and

Limitation [17.1]	Claimed Features
	Single device or multiple devices

194. Buehler discloses that some elements of the claimed system can "be integrated into the robot" or be "kept (at least in part) in a separate control facility in communication with the robot," i.e. be included in a single device or multiple devices. *Id.*, 11:25-27 ("[C]ontrol system 200 and its various components may be integrated into the robot, or kept (at least in part) in a separate control facility in communication with the robot.").

Tesla Exhibit 1003.183

b. [17.2] wherein the one or more processors include: one or more microcontrollers, one or more computing devices, or one or more electronic devices, and

Limitation [17.2]	Claimed Features
	Processors = Microcontrollers, computing devices or electronic devices

195. Buehler discloses processors including a computing device. *Id.*, 10:5-12 ("[T]he robot controller . . . [is] implemented on a general-purpose computer . . . [that] includes a central processing unit (CPU) 222 (e.g., a quad-core Intel processor)….").

c. [17.3] wherein the one or more memories include: one or more non-transitory machine readable media, one or more volatile memories, one or more non-volatile memories, one or more storage devices, or one or more storage systems, and

Limitation [17.3]	Claimed Features
	Memories = storage system, storage device, or non-volatile memory

196. Buehler discloses a "system memory 224, and non-volatile mass storage devices (such as one or more hard disks and/or optical storage units)…." *Id.*, 10:10-14; *see also id.,* 5:27-31 ("In some embodiments, the control system includes a data storage medium (e.g., system memory or a non-volatile storage medium such

as, e.g., a hard drive) storing a plurality of visual models and associated object representations.").

d. [17.4] wherein the first one or more digital pictures include: one or more still digital pictures, or one or more motion digital pictures, and

Limitation [17.4]	Claimed Features
	1st DP = still DP or motion DP

197. Buehler discloses recording and storing first digital pictures that are still digital pictures (e.g., images) or motion digital pictures (*e.g.*, video streams). *Id.*, 17:34-36 ("The contextual information 502 relevant to object recognition includes a previously ***acquired image or video stream***….").

e. [17.5] wherein the new one or more digital pictures include: one or more still digital pictures, or one or more motion digital pictures, and

Limitation [17.5]	Claimed Features
	New DP = still DP or motion DP

198. Buehler discloses that the new digital pictures generated and received from the robot camera can be still digital pictures (e.g., images) or motion digital pictures (e.g., video streams), which are displayed in the simulation program. *Id.*, 21:46-48 ("Following the user's approval of the selected visual model, the robot may overlay virtual grippers on ***the image (or video stream)***….").

Tesla Exhibit 1003.185

f. [17.6] wherein the first application program includes: a video game, a computer game, a simulation program, a program including text processing, a program including number processing, a program including picture processing, a program including object processing, or a program, and

Limitation [17.6]	Claimed Features
	1st application = simulation program or other embodiment

199. Buehler discloses a first application program including a simulation program for simulating performance of tasks during training mode. *Id.*, 2:43-45 ("As part of the training process, the robot ***may repeatedly simulate performance of its task***, each time adjusting its actions in executing the task based on user feedback."); 13:55-61 ("In many cases, the ***robot simply simulates what it would do in execution mode*** (e.g., select an object), and augments the simulation with intuitive information (e.g., an outline of the object) that helps the user assess the robot's performance (e.g., decide whether the selection is correct) and provide feedback to alter task-execution if appropriate (e.g., adjust the outline).").

g. [17.7] wherein the second application program includes: a video game, a computer game, a simulation program, a program including text processing, a program including number processing, a program including picture processing, a program including object processing, or a program, and

Limitation [17.7]	Claimed Features
	2nd application = simulation program or other embodiment

Tesla Exhibit 1003.186

200. Buehler discloses and/or renders obvious a second application program also including a simulation program. Buehler describes that its system can be used to train a variety of industrial robots designed to perform different tasks. *Id.*, 1:25 ("Industrial robots perform a variety of tasks…."). A POSITA would have understood and found obvious that when Buehler's system is applied to a second of the same type of robot or to a different model of robot, the robot would include a second application program that includes a simulation program to simulate the second robot's interactions with target objects during training mode, as described above.

h. [17.8] wherein an instruction set of the first one or more instruction sets for operating the first object of the first application program includes at least one of: only one instruction, a plurality of instructions, one or more inputs, one or more commands…or one or more information.

Limitation [17.8]	Claimed Features
	1st instructions = instructions, input or other embodiment

201. Buehler discloses that the instructions for operating the virtual robot and its end effector include input from a user and/or the robot's control system provided during training mode, and/or one or more commands or instructions provided by the user or programmed into the robot (e.g., a routine for picking up an object). *Id.*, 22:26-32 ("[T]he ***routines may be programmed into the robot*** at the

Tesla Exhibit 1003.187

outset and altered, as necessary, during training."), 21:54-64 ("If the virtual gripper is not positioned around the proper part of the object (as determined in step 608), the ***user may adjust the arm to roughly establish the proper grasp point*** (step 610). The robot may then fine-tune the arm location to optimize the gripper location relative to the rough grasp point on the object (step 612). This process of coarse and fine adjustments of the gripper by the human trainer and the robot, respectively, may be iterated until the user is satisfied with the grasp point and gripper orientation shown in the visual display.").

7. Claim 18

a. [18.1] The system of claim 1, wherein the first one or more digital pictures are: one or more whole digital pictures, one or more representations of one or more whole digital pictures, one or more portions of at least one digital picture, one or more representations of one or more portions of at least one digital picture, one or more features, one or more representations of one or more features, one or more collections of pixels, or one or more collections of values, and

Limitation [18.1]	Claimed Features
	1st DP = representation of a portion of a DP or other embodiment

202. As I explained above, in relation to limitation [1.2], Buehler discloses that the first digital picture(s) are whole or a portion of a whole digital pictures, such as images previously captured by the robot's cameras and sensors, stored in the robot's memory. *See* Section IX.C.1.b above. Buehler also describes that the first

Tesla Exhibit 1003.188

digital pictures can comprise one or more features or collections of pixels, which are used to compare the templates to newly captured digital pictures by the camera. *See* Section IX.C.1.d above (describing that the pictures can be stored in pixel form).

b. [18.2] wherein the new one or more digital pictures are: one or more whole digital pictures, one or more representations of one or more whole digital pictures, one or more portions of at least one digital picture, one or more representations of one or more portions of at least one digital picture, one or more features, one or more representations of one or more features, one or more collections of pixels, or one or more collections of values.

Limitation [18.2]	Claimed Features
	New DP = whole DP, representation of a portion of a DP or other embodiment

203. As I explained above, in relation to limitation [1.3], Buehler discloses that the new digital picture(s) are whole or a portion of a whole digital pictures captured by the robot's cameras and sensors. *See* Section IX.C.1.c above.

8. Claim 19

204. Independent claim 19 is a method claim that corresponds to system claim 1. My analysis above explaining how Buehler discloses the limitations of claim 1 applies to the limitations of claim 19 as set forth in the chart below.

Tesla Exhibit 1003.189

Claim 19	Corresponding Analysis for Claim 1
[19.1] A method implemented using a computing system that includes one or more processors, the method comprising: accessing one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program;	*See* analysis for limitations [1.1] and [1.2] above. *See* Section IX.C.1.a-b above.
[19.2] receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program;	*See* analysis for limitation [1.3] above. *See* Section IX.C.1.c above.
[19.3] determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the new one or more digital pictures and the first one or more digital pictures; and	*See* analysis for limitation [1.4] above. *See* Section IX.C.1.d above.
[19.4] at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.	*See* analysis for limitation [1.5] above. *See* Section IX.C.1.e above.

Tesla Exhibit 1003.190

9. Claim 20

a. [20.1] A system comprising: means for processing; and

Limitation [20.1]	Claimed Features
	System + means for processing

205. As I explained above in relation to claim construction, the corresponding structure for this limitation includes a processor. As I discussed above in relation to claim [1], Buehler discloses a system comprising a processor. *See* Section IX.C.1.a above.

b. [20.2] means for storing machine readable code that, when executed by the means for processing, causes the means for processing to perform at least:

Limitation [20.2]	Claimed Features
	Means for storing

206. As I explained above in relation to claim construction, the corresponding structure for this limitation is storage or a memory. Buehler discloses that the machine readable code (*i.e.*, sets of instructions) are stored in the robot's storage and/or memory. Ex. 1006, 10:5-14 ("[T]he robot controller and high-level computational functionality are implemented on a general-purpose computer 210…. The general-purpose computer 210 includes a central processing unit (CPU) 222…*system memory 224, and non-volatile mass storage devices (such as one or more hard disks and/or optical storage units*) 226."), 4:51-56 ("The image

Tesla Exhibit 1003.191

processing module and the control system comprise *sets of instructions stored in memory* and executable by a processor."), 12:21-25 (describing that robot-user interaction is "controlled and processed by associated ***program code***").

c. [20.3] accessing one or more memories that store at least a first one or more digital pictures correlated with a first one or more instruction sets for operating a first object of a first application program;

207. As I explained above in relation to limitation [1.2], Buehler discloses that the processor accesses memories storing digital pictures correlated with instruction sets for operating a first object of the application. *See* Section IX.C.1.b above.

d. [20.4] receiving or generating a new one or more digital pictures that depict at least a portion of a surrounding of: the first object of the first application program, a second object of the first application program, or a first object of a second application program;

208. As I explained above in relation to limitation [1.3], Buehler discloses that the processor receives and generates new digital pictures depicting part of the surrounding of the simulated robot and end-effector, such as an object to be picked up. *See* Section IX.C.1.c above.

e. [20.5] determining the first one or more instruction sets for operating the first object of the first application program based on at least partial match between the

Tesla Exhibit 1003.192

new one or more digital pictures and the first one or more digital pictures; and

209. As I explained above in relation to limitation [1.4], Buehler discloses that the processor determines the instruction sets for operating the first object of the application based at least partial match between the digital pictures recorded during training mode and the digital pictures generated by the robot's cameras. *See* Section IX.C.1.d above.

f. [20.6] at least in response to the determining, executing the first one or more instruction sets for operating the first object of the first application program, wherein the first object of the first application program, the second object of the first application program, or the first object of the second application program autonomously performs one or more operations defined by the first one or more instruction sets for operating the first object of the first application program.

210. As I explained above in relation to limitation [1.5], Buehler discloses that the processor executes the first instruction set, causing the first object of the first application to autonomously perform the defined operations. *See* Section IX.C.1.e above.

D. Ground 5: Obviousness of Claim 16 based on Buehler in view of Ponulak

211. A POSITA would have been motivated to combine Buehler with Ponulak. As I described above in relation to claim 16, since 1992 a POSITA would have understood that a neural network could be applied to the operation of an autonomous mobile robot with visual sensors, such as cameras, as described in

Tesla Exhibit 1003.193

Pomerleau. *See* Section IX.C.5 above (Buehler limitation 16.1). A POSITA would have been motivated to combine Buehler with the neural network disclosed in Ponulak for several reasons. First, both references describe using digital image inputs to detect objects and provide instructions for operating a robot, including for industrial applications. Ex. 1007, 1:41-45, Ex. 1006, 1:25-26. This would have suggested to a POSITA that Ponulak's neural network for detecting objects and providing instructions to autonomously operate a robot would have been applicable to Buehler. Given these similarities, the combination would have been obvious as the application of Ponulak's technique (*i.e.*, using a neural network for robot object detection and instruction selection) to improve a similar device (Buehler's robot).

212. Second, a POSITA would have been motivated to utilize Ponulak's neural network for image detection in Buehler given the advantages described therein. In particular, Ponulak describes that use of its neural network and adaptive predictor apparatus may allow "for faster and smoother anticipatory avoidance behavior" and "may speed-up and/or improve reliability of object detection…." Ex. 1007, 38:34-41. Ponulak also describes that its system can use an image recognition technique described in the incorporated patent App. 13/756,382 (U.S. Patent No. 9,195,934) that "may provide a more timely response upon receiving input data (e.g., characterized by a shorter delay between receipt of the input and provision of the output). Faster response may be advantageous in various applications…such as, for

Tesla Exhibit 1003.194

example collision avoidance." Ex. 1010, 30:31-43; *see also* Ex. 1007, 11:57-67 (incorporating by reference App No. 13/756,382). A POSITA would have understood that these advantages of Ponulak's system would have been desirable for Buehler's robotic system. In particular, a POSITA would have recognized that industrial robots, as described in Buehler, would benefit from more accurate object detection to ensure that the robot performed the correct instruction set. A POSITA also would have recognized that Buehler's robot would have benefited from faster object detection so that the robot could better avoid colliding with other robots and/or other objects that would likely be moving in an industrial setting.

213. A POSITA would additionally have been motivated to combine Buehler with Ponulak because they describe similar systems intended for the same field of use. For example, both systems can be applied to a robot intended for industrial uses. Ex. 1007, 1:41-45 ("Robotic devices may be used in a variety of applications, such as manufacturing, medical, safety, military, exploration, and/or other applications. Some existing robotic devices (e.g., manufacturing assembly and/or packaging) may be programmed in order to perform desired functionality."), Ex. 1006, 1:25-31 ("Industrial robots perform a variety of tasks involving the movement and manipulation of various objects. A typical industrial robot as used, e.g., in a manufacturing environment, may have one or more arms equipped with grippers that allow the robot to pick up, transport, and put down objects, and a

Tesla Exhibit 1003.195

machine-vision system that guides the robot in the performance of its tasks."). Ponulak describes a system that associates sensory data inputs with output instructions for operating a robot, where the sensory data can include a detected image. Ex. 1007, 2:3-7 ("***configuring the adaptive controller, using one or more processors, to generate a control output based on a sensory input and the reinforcement signal***, the sensory input including information associated with an environment of the robotic apparatus"), 27:65-67 ("In one or more implementations, examples of aspects of ***the sensory input may include an object being detected in the input***"). Buehler and Ponulak both describe systems that can detect features of a target object. Ex. 1006, 3:7-13 ("some objects may be best recognized based on features extracted from the images…."); Ex. 1007, 33:12-16 ("[I]ndividual subsequent layers of detectors may…*detect more complex features* and objects (as compared to the *features detected* by the preceding detector layer)."). As I explained above, Buehler similarly discloses a system associating commands for operating a robot with sensor data including detected objects.

214. A POSITA accordingly would have understood that the neural network for operating an autonomous robot described by Ponulak, which "generate[s] a control output based on sensory input," would be applicable to the system described in Buehler for operating an autonomous robot by executing instruction based on data from attached cameras. *Id.*, 2:4-5.

Tesla Exhibit 1003.196

215. Ponulak and Buehler also both describe a training mode, where the robot simulates detecting a target object and performing a task based on object-specific instructions. *Id.*, 28:5-8 ("In some implementation, the sensory input may be received based on performing one or more training trials (e.g., as the trials described with respect to FIGS. 2B-2C above) of a robotic apparatus."), 26:57-27:5 ("The environment (e.g., shown by the rectangle 600 in FIG. 6) may comprise one or more obstacles 608 and/or walls 602. ***The training may comprise obstacle avoidance training based on one or more reinforcement events*** depicted by stars 614.…In some implementations, as the robotic device 610 approaches the obstacle, the reinforcement signal may comprise negative reinforcement."); Ex. 1006, 21:43-54 ("***The robot may repeat the process of selecting (step 602) and testing (step 604) visual models until the user confirms the robot's ability to successfully detect the object***. Following the user's approval of the selected visual model, the robot may overlay virtual grippers on the image (or video stream) of the object to indicate where the grasp point would be, given the current arm location and orientation (step 606). Thus, without the need to actually lower the gripper and bring it in contact with the object…the robot can give the user a sense of the likelihood of a successful pick-up.").

216. Buehler and Ponulak are directed to common subject matter- operating autonomous and semi-autonomous robots, for uses including industrial application.

Tesla Exhibit 1003.197

A POSITA would have been familiar with the teachings of both patents and would have understood that, as I explained above, combining their teachings would have beneficial.

217. A POSITA also would have expected to succeed in combining Buehler and Ponulak, and found predictable that the robots described in Buehler could be combined with the neural network described in Ponulak for advantageous results. As described above, a POSITA would have expected to succeed in combining Buehler and Ponulak given the substantially similar and complimentary subject matter, as I described above. Moreover, it was well-known in the art how to incorporate a neural network into a system for determining instruction sets for autonomously operating a robot based on visual inputs captured by the robot (e.g., digital pictures), as I described above for the Pomerleau reference. *See* Section IX.C.5 above.

218. For at least these reasons, it would have been obvious to a POSITA to combine Buehler and Ponulak.

1. Claim 16

a. [16.1] The system of claim 1, wherein the system further comprising: a neural network that includes the first one or more digital pictures connected with the first one or more instruction sets for operating the first object of the first application program using at least one of: one or more connections, or one or more nodes, and

	Claimed Features

Tesla Exhibit 1003.198

Limitation [16.1]	Neural network containing **1st DP** connected with **1st instructions** using a connection or node + Neural network = data structure converting input to output

219. A POSITA would have found the neural network described by limitation 16.1 obvious in view of Buehler and Ponulak. As I explained above in relation to claim 1, Buehler describes storing correlations between sensor data depicting a target object (first digital pictures) and instructions for operating a virtual representation of the robot and its end-effector (first object of the first application). Ponulak describes a neural network that converts inputs comprising sensor data containing detected objects (digital pictures) into outputs comprising motor commands for operating a robot (instruction sets). Ex. 1007, 9:1-20 ("The output 108 may comprise one or more motor commands…. [T]he signal 106 may comprise an array of pixel values in the input image…."). For example, Ponulak describes that the input can comprise a digital picture in the form of visual input received via an interface. *Id.*, 36:42-43 ("The neuromorphic apparatus 1150 may receive input (e.g., visual input) via the interface 1160").

220. Ponulak, in turn, describes connections between different clusters and cells, which fill different roles in the neural network, such as detecting objects or determining motor commands. *Id.*, 36:16-18 ("A number of *cell blocks* may be

arranged in a cluster and may ***communicate with one another via local interconnects***"), 36:20-21 ("Similarly, several L2 clusters may communicate with one another ***via a second level interconnect***…."), 36:35-40 ("the L2, L3 cells performing progressively higher level functionality (e.g., object detection). ***Individual ones of L2, L3, cells may perform different aspects*** of operating a robot with one or more L2/L3 cells ***processing visual data from a camera***, and other L2/L3 cells ***operating motor control block***….").

221. It would have been obvious to apply Ponulak's neural network to Buehler's system. In the combination, it would have been obvious to a POSITA to incorporate Buehler's object detection functionality in one or more cells (including stored image templates and the algorithm for comparing templates to newly captured images) that are connected with one or more other cells containing motor commands determined based on the comparison of image data. As described by Ponulak, each cell or cluster of cells could perform the described functions and produce output including the target object and the associated motor commands. *Id.*, 36:49-51 ("The output may include one or more of an indication of recognized object or a feature, a motor command"). Ponulak illustrates the functionality of its neural network in Figure 11D, reproduced below, and annotated to illustrate its combination with Buehler's object detection and instruction set determination:

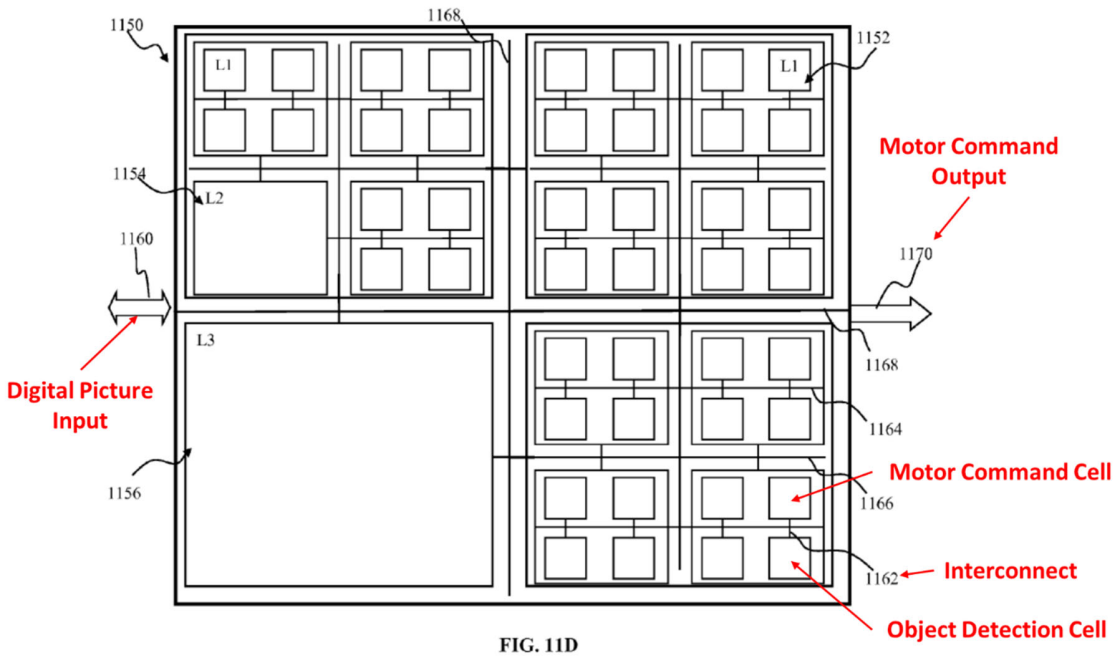


FIG. 11D

Ex. 1007, Figure 11D (annotated)

b. [16.2] wherein the determining the first one or more instruction sets for operating the first object of the first application program based on the at least partial match between the new one or more digital pictures and the first one or more digital pictures includes applying the new one or more digital pictures to the neural network.

Limitation [16.2]	Claimed Features
	Applying neural network to **new DP**

222. As I explained above, Ponulak describes a neural network that can be applied to sensor data input to detect a particular object (a new digital picture) and output the associated motor command (instruction set). *Id.*, 36:42-51 ("The neuromorphic apparatus 1150 may receive input (e.g., visual input)…. The output

may include…a motor command….”). It would have been obvious to convert the sensor input into motor commands based on a process of partially matching digital pictures as described by Buehler, as I explained above in relation to claim 1. For example, the system would analyze the new sensor data input (new digital pictures) and the image features stored in Ponulak’s neural network (first digital pictures) to detect at least a partial match. *Id.*, 33:12-16 (“[I]ndividual subsequent layers of detectors may be configured to ***receive signals from the previous detector layer, and to detect more complex features and objects***….”). Based on at least a partial match, Ponulak’s neural network applied to Buehler’s system would determine and execute an instruction set, such as one or more motor commands for operating the robot. *Id.*, 36:42-51 (“The neuromorphic apparatus 1150 may receive input (e.g., visual input)…. The output may include…a motor command….”); *see also* Section IX.C.1.d above (Buehler describing matching digital pictures to determine instruction sets for operating a virtual robotic end effector).

X. The ’134 Patent

A. The Claims

223. I understand that the petition challenges claims 1-20 of the ’134 Patent. Independent claim 1 of the ’134 Patent is directed to a system that determines an instruction set for operating an avatar of an application by comparing and matching a first stored object representation with a second

Tesla Exhibit 1003.202

generated or received object representation. As I explain in this Declaration, a POSITA, in and prior to 2016, would have understood that there was nothing inventive or non-obvious about the components of this system, which were all well known in the art at the time. Claim 1 is reproduced below:

> 1. A system comprising:
> (a) one or more processors; and
> (b) one or more non-transitory machine readable media storing machine readable code that, when executed by the one or more processors, causes the one or more processors to perform at least:
> (c) accessing a first correlation including a first one or more object representations correlated with a first one or more instruction sets for operating a first avatar of an application, wherein the first one or more object representations represent one or more objects of the application;
> (d) generating or receiving a second one or more object representations, wherein the second one or more object representations represent one or more objects of the application;
> (e) determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second one or more object representations and the first one or more object representations;
> (f) at least in response to the determining, causing the first avatar of the application or a second avatar of the application to perform one or more operations defined by the first one or more instruction sets for operating the first avatar of the application at least by executing the first

one or more instruction sets for operating the first avatar of the application.

224. Independent claim 10 is nearly identical to claim 1, but discloses a method instead of a system.

225. Independent claim 19 of the ’134 Patent is directed to a nearly identical system claim 1, except that claim 19 includes means-plus-function claim limitations. As I explain in this Declaration, a POSITA, in and prior to 2016, would have understood that there was nothing inventive about the components of this system, which were all well known in the art at the time. Claim 19 is reproduced below:

A system comprising:

(a) a memory that stores at least a first correlation including a first one or more object representations correlated with a first one or more instruction sets for operating a first avatar of an application, wherein the first one or more object representations represent one or more objects of the application;

(b) means for generating or receiving a second one or more object representations, wherein the second one or more object representations represent one or more objects of the application;

(c) means for determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second one or more object representations and the first one or more object representations; and

Tesla Exhibit 1003.204

(e) means for executing, at least in response to the determining, the first one or more instruction sets for operating the first avatar of the application, wherein the first avatar of the application or a second avatar of the application performs one or more operations defined by the first one or more instruction sets for operating the first avatar of the application.

226. Dependent claims 2-9 and 11-18, which all depend from claim 1, and claim 20, which depends from claim 19, can be summarized as follows in the table below based on the additional steps or features that they require. As I further explain in this Declaration, the steps and features added by these dependent claims were also well-known at or prior to 2016, and a POSITA would have understood that there was nothing inventive about the combination of steps and features recited in these claims.

Dependent Claim	Additional Features Claimed
Claim 2	In addition to the limitations of claim 1, claim 2 further requires that the first and second object representations each include at least one object representation and that the first and second object representations be detected at different times. + **1st object representations (OR)** and **2nd OR** include at least one OR + **1st OR** and **2nd OR** detected at different times

Tesla Exhibit 1003.205

Dependent Claim	Additional Features Claimed
Claim 3	In addition to the limitations of claim 1, claim 3 further requires determining that the number or percentage of partially matching portions of the first and second object representations exceeds a threshold. + Partial match exceeds a threshold
Claim 4	In addition to the limitations of claim 1, claim 4 further requires a knowledgebase storing at least a first correlation and a third correlation, where the third correlation includes at third object representation(s) representing object(s) of an application, and a connection connecting the first and third correlations. + Knowledgebase + **3rd correlation** (**3rd OR** + **3rd instructions**) ○ **3rd OR** represents objects of **the application** or **another application** + Connection between **1st correlation** and **3rd correlation**

Tesla Exhibit 1003.206

Dependent Claim	Additional Features Claimed
Claim 5	In addition to the limitations of claim 1, claim 5 further requires a second avatar of the application execute the first instructions for operating the first avatar. + **2nd avatar** executes **1st instructions** for **1st avatar**
Claim 6	In addition to the limitations of claim 1, claim 6 further requires the first object representation(s) represent object(s) in the surrounding of the first avatar and the second object representation(s) represent object(s) in the surrounding of the first avatar or a second avatar, where an avatar's surroundings include one of the various listed embodiments. + **1st OR** includes objects in surroundings of **1st avatar** + **2nd OR** includes objects in surroundings of **1st avatar** or **2nd avatar** o Surroundings=one of the listed embodiments
Claim 7	In addition to the limitations of claim 1, claim 7 further requires a temporal correspondence between the first instruction set(s) and first object representation(s). + Temporal correspondence between **1st instructions** and **1st OR**

Tesla Exhibit 1003.207

Dependent Claim	Additional Features Claimed
Claim 8	In addition to the limitations of claim 1, claim 8 further requires all of the structures listed in claim 1 to include one of the listed embodiments. + Various listed embodiments
Claim 9	In addition to the limitations of claim 1, claim 9 further requires at least one knowledge cell, where a knowledge cell is a data structure storing a collection of object representations correlated with an instruction set. + Knowledge cell o Knowledge cell = Data structure storing correlation

Tesla Exhibit 1003.208

Dependent Claim	Additional Features Claimed
Claim 11	In addition to the limitations of claim 1, claim 11 further requires a knowledgebase storing at least a first correlation and a third correlation, the third correlation includes at third object representation(s) representing object(s) of an application, and at least a portion of the first correlation is learned in a learning process involving a human or non-human user operating an avatar and that at least a portion of the third correlation is learned in a separate learning process where the same user operates the same avatar. + Knowledgebase + **3rd correlation** (**3rd OR** + **3rd instructions**) + **1st correlation** and **3rd correlation** learned with operation of the **1st avatar** by **1st user** o User=human or non-human user

Tesla Exhibit 1003.209

Dependent Claim	Additional Features Claimed
Claim 12	In addition to the limitations of claim 1, claim 12 further requires a knowledgebase storing at least a first correlation and a third correlation, the third correlation includes at third object representation(s) representing object(s) of an application, and the third correlation is at least partly learned in a separate learning process where a different human or non-human user operates the same avatar. + Knowledgebase + **3rd correlation** (**3rd OR** + **3rd instructions**) + **3rd correlation** learned in **another learning process** with operation of the **1st avatar** by **another user**

Tesla Exhibit 1003.210

Dependent Claim	Additional Features Claimed
Claim 13	In addition to the limitations of claim 1, claim 13 further requires a knowledgebase storing at least a first correlation and a third correlation, the third correlation includes at third object representation(s) representing object(s) of an application, the first correlation is learned in a learning process where the first avatar is operated by a user and the third correlation is at least partly learned in a separate learning process where a different human or non-human user operates a third avatar. + Knowledgebase + **3rd correlation** = **3rd OR** + **3rd instructions** for operating **3rd avatar** + **3rd correlation** learned in **another learning process** through operation of the **3rd avatar** by **another user**

Tesla Exhibit 1003.211

Dependent Claim	Additional Features Claimed
Claim 14	In addition to the limitations of claim 1, claim 14 further requires knowledgebase storing at least a first correlation and a third correlation, the third object representation(s) representing object(s) of another application and instructions for operating the third avatar of another application. + Knowledgebase + **3rd correlation** = **3rd OR** + **3rd instructions** for **3rd avatar** of **another application** + **3rd OR** represent objects **another application**
Claim 15	In addition to the limitations of claim 1, claim 15 further requires that the first instructions include information about the state of the first avatar. + **1st instructions** include state of the **1st avatar**

Tesla Exhibit 1003.212

Dependent Claim	Additional Features Claimed
Claim 16	In addition to the limitations of claim 1, claim 16 further requires that an element of the first correlation is deleted/modified/manipulated/inserted after the first correlation is generated. + Element of **1st correlation** is deleted/modified/manipulated/inserted
Claim 17	In addition to the limitations of claim 1, claim 17 requires modifying the first instructions and executing the modified instructions. + Modification of **1st instructions** + **1st avatar** executes modified **1st instructions**
Claim 18	In addition to the limitations of claim 11, claim 18 further requires generating or receiving the first object representation(s) and obtaining or receiving the first instructions. + Generating/receiving **1st OR** + Obtaining/receiving **1st instructions**

Dependent Claim	Additional Features Claimed
Claim 20	In addition to the limitations of claim 19, claim 20 further requires that the means for generating or receiving, the means for determining and the means for executing include one or more processors. + Means for generating/receiving + means for determining + means for executing include processors

B. Grounds 1 and 2: Anticipation and/or Obviousness of Claims 1-20 based on Thomas

227. As set forth in detail below, claims 1-20 of the ’134 Patent are anticipated and/or rendered obvious by Thomas. The discussion below specifies where each element of the aforementioned claims is found in Thomas, and includes a detailed explanation of the significance of the quotations and citations.

1. Claim 1

228. As described below, Thomas anticipates claim 1 of the ’134 Patent. Thomas includes several examples that demonstrate anticipation, including the pump fake (Ex. 1005, 8:8-15) and pass out of shot (*id.,* 7:55-64) examples. The tables below identify how those examples map to various elements of claim 1:

Tesla Exhibit 1003.214

Pump Fake (Ex. 1005, 8:8-15)	• A first one or more object representations = spatial position of defensive players recorded during learning mode • A first one or more instruction sets = user inputs to cause the user-controlled avatar to pump fake during learning mode • A first avatar of an application = an avatar that pump fakes during learning mode based on user controls, which is later autonomously operated during simulation mode • A second one or more object representations = video game displayed on interface screen during simulation mode, including spatial position of defensive players
Pass Out of Shot (Ex. 1005, 7:55-64)	• A first one or more object representations = spatial position of defensive players recorded during learning mode • A first one or more instruction sets = user inputs to cause the user-controlled avatar to pass the ball during learning mode • A first avatar of an application = an avatar that passes during learning mode based on user controls, which is later autonomously operated during simulation mode • A second one or more object representations = video game displayed on interface screen

Tesla Exhibit 1003.215

	during simulation mode, including spatial position of defensive players

a. [1.1] A system comprising: one or more processors; and

Limitation [1.1]	Claimed Features
	System including processors

229. As I explained above in regards to limitation [1.1] of the '585 Patent, Thomas discloses a system including one or more processors. *Id.*, abstract ("The field of the invention relates to… systems and methods for simulating a particular user's behavior in an interactive computer ***system***"), *id.*, claim 1 ("[A] computer-usable medium having a sequence of instructions stored thereon, which…causes said ***processor*** to execute a first process….").

b. [1.2] one or more non-transitory machine readable media storing machine readable code that, when executed by the one or more processors, causes the one or more processors to perform at least:

Limitation [1.2]	Claimed Features
	+ machine readable media + storing code + executed by processors + causes processors to perform

Tesla Exhibit 1003.216

230. The '134 Patent specification describes that "non-transitory machine readable medium" comprises "***all machine-readable media*** except for a transitory, propagating signal," '134 Patent, 64:26-28, and includes "any memory" or "storage medium." *Id*., 64:12-25. The specification further describes that "code" can comprise "instructions that can provide functionality when executed by processor." *Id*. at 60:53-56.

231. Thomas discloses this limitation because it describes "a ***computer-usable medium having a sequence of instructions stored thereon***, which…causes said processor to execute a first process," where a sequence of instructions would be understood to refer to "code" as described above. Ex. 1005, claim 1; *see also id*., 19:39-43 ("Alternatively, or in addition, the computer program 110, the simulator 120, and database 130 can ***reside on a server 710*** accessible by one or more client consoles 750/740 via a network, local area or wide area, such as the Internet 770."), claim 19 ("the simulator and the database are located on ***portable memory device*** enabled to be directly connected to the local processor."); *see also id.*, 13:60-19:34 (a list of stored information in machine readable form).

c. [1.3] accessing a first correlation including a first one or more object representations correlated with a first one or more instruction sets for operating a first avatar of an application, wherein the first one or more object representations represent one or more objects of the application;

	Claimed Features

Tesla Exhibit 1003.217

Limitation [1.3]	Accessing a **1st correlation** + **1st correlation** = **1st OR** + **1st instructions** + OR = object representations representing objects of the application

232. Thomas discloses this limitation, and specifically describes that, during learning mode, the simulator records and stores a user's inputs for operating a first avatar in the game (i.e. an instruction set for operating a first avatar of a first application program), which are "associated with" or "linked to" (*i.e.*, correlated with) the situation or event that "initiated the user's input." *Id.*, 9:43-45, 11:25-26.

> During the learn mode, if a first user is using the computer program 110, ***the simulator 120 collects and evaluates data about the first user's input when interacting with the computer program*** 110 (action block 200). ***The collected data is then stored in the database*** 130 (action block 210)….***Generally, the data is logically linked to an event, or situation, that initiated the user's input.***
>
> ***
>
> Next, the simulator 120 monitors and analyzes the user's interaction with the program 110 (action block 555). ***If the simulator 120 identifies a particular situation and/or event occurring, defined by parameters such as those above (decision block 560), then the simulator 120 will record the user's decision/input, such as a pressed button, combination of buttons, or joystick move of a controller, in response to the particular situation and/or event (action block 565) and store the user's input in the field or sub-profile associated with the particular situation and/or event*** (action block 570)…. [T]he data

Tesla Exhibit 1003.218

> recorded can be associated with both an event and a situation. In other words, when a user responds to an event, the ***user's response, when recorded, is not only associated with the occurrence of that event, but also with the situation during the occurrence of that event, which is defined by one or more current circumstances***.

Id., 3:53-61, 4:62-5:13.

233. Thomas additionally discloses that the associated situation/event can be defined by circumstances depicted on the interface screen and recorded by the system, such as the spatial position of other avatars or road conditions in the video game:

> ***One common parameter that defines a situation*** may be time….Another ***may include spatial position of avatars controlled by the user 665 on the interface screen***. For example, in football, the spatial position may be position on a football field…. Another parameter may be the spatial position(s) of the avatars controlled by one or more user opponents 675 of the game.
>
> ***
>
> For a catch and shoot event, during the learning mode 550, ***the simulator 120 records how often a user catches and shoots right after the catch and records, as a current circumstance, the location of the user's avatar that catches and shoots.***
>
> ***
>
> Thus, the data recorded can be associated with both an event and a situation. In other words, when a user responds to an event, the ***user's***

Tesla Exhibit 1003.219

> ***response, when recorded, is not only associated with the occurrence of that event, but also with the situation during the occurrence of that event, which is defined by one or more current circumstances***.
>
> ***
>
> One of ordinary skill in the art would also appreciate that ***other circumstances 680 may be included as parameters, such as*** virtual weather in games such as golf, or road ***conditions of racing games***.
>
> ***
>
> [T]he simulator 120 will record the user's decision/input, such as a pressed button, combination of buttons, or joystick move of a controller, in response to the particular situation…and store the user's input in the field or sub-profile associated with the particular situation

Id., 4:29-39, 5:7-13, 4:51-54, 4:67-5:5; *see also id.,* 8:67-9:9, 9:43-53, 5:7-18

234. The specification of the '134 Patent describes an "object representation" as a representation of any object in an application, such as a "person, animal, tree, rock, building, vehicle, and/or others in a context of a computer game"," which can include "any information related to an object," such as the object's location. '134 Patent, 69:3-8 ("Object Representation 625 may include a representation of an object . . . in Avatar's 605 surrounding within Application Program 18. As such, Object Representation 625 may include any information related to an object. In other designs, Object Representation 625 may include or be replaced with an object itself . . . ."); 70:32-41 ("Examples of Object Properties 630

Tesla Exhibit 1003.220

(i.e. also referred to simply as object properties, etc.) include existence of Object 615, type of Object 615 (i.e. ***person***, animal, tree, rock, building, vehicle, etc.), identity of Object 615 (i.e. name, identifier, etc.), distance of Object 615, bearing/angle of Object 615, ***location of Object 615*** (i.e. distance and bearing/angle from a known point, coordinates, etc.), shape/size of Object 615 (i.e. scale, height, width, depth, computer model, etc.), activity of Object 615 (i.e. motion, gestures, etc.), and/or other properties of Object 615."). Accordingly, the spatial position of avatars and the road conditions disclosed in Thomas are object representations that represent one or more objects of the application (*i.e.*, the video game).

235. Thomas further discloses that during simulation mode, the simulator accesses the stored data (including the instruction sets correlated with object representations) to simulate a user's control of the avatars in response to a situation or event:

> ***During simulation mode***, if a second user is using the computer program 110 and the second user desires to interact with the first user, but the first user is not available, ***the simulator 120 will then retrieve input data from the first user, i.e., the first user's profile, from the database 130 (action block 220) and simulate the first user's behavior based on the retrieved data*** (action block 230).

Ex. 1005, 3:62-4:2; *see also id.*, 5:30-61.

Tesla Exhibit 1003.221

236. The pump fake and pass out of shot examples identified above confirm that Thomas discloses this limitation. For example, Thomas describes that during simulation mode, the system accesses stored correlations between a user's inputs (first one or more instruction sets) for controlling an avatar in a computer video game (a first avatar of an application) and object representations that represent one or more objects of the application, such as the spatial position of other avatars in the basketball game. In the pump fake example, Thomas describes that the system accesses a command for causing an avatar to pump fake (*i.e.*, an instruction set) that is correlated with a first object representation (*i.e.*, the spatial position of the defensive avatars). Similarly, in the pass out of shot example Thomas describes that the system accesses a stored command for causing an avatar to pass (*i.e.*, an instruction set) that is correlated with a first one or more object representations (*i.e.*, the spatial position of the defensive avatars):

> For a ***pass out of shot*** event, the simulator 120, during learn mode 550, records a shot type, ***which avatar passed, and which avatar was passed to***. For current circumstances, the simulator 120 ***records how open the receiving avatar was and how covered the passing avatar was***. During playback…simulator 120 will ***invoke a pass if the current circumstances meet the criteria***, e.g., how much the ***passing and receiving avatars are covered***.
>
> ***

Tesla Exhibit 1003.222

For a ***pump fake event***, during the learning mode 550, the simulator 120 ***records how often a user pump fakes***. For current circumstances, the simulator 120 ***records how covered the user was and user's location***. During playback mode 600, the simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and the defensive coverage. If criteria is met***, then simulator 120 ***will invoke a pump fake***.

Id., 7:55-64, 8:8-15.

d. [1.4] generating or receiving a second one or more object representations, wherein the second one or more object representations represent one or more objects of the application;

Limitation [1.4]	Claimed Features
	Generating or receiving **2nd OR**

237. In simulation/playback mode, Thomas discloses generating new object representations (*i.e.*, a second one or more object representations) during simulation or playback mode, which include object representations such as the spatial positions of other avatars and the virtual road conditions in a video game:

> ***The simulator 120 generally operates in two modes***: (1) learn, or record, mode and ***(2) simulation, or playback, mode***.
>
> ***
>
> One common parameter that defines a situation may be time 660, e.g., time left in a particular session or game, and/or time played in a particular session or game. Another may include ***spatial position of avatars controlled by the user 665 on the interface screen***. For

Tesla Exhibit 1003.223

example, in football, the spatial position may be position on a football field, as will be explained in more detail below.

Id., 3:49-51, 5:34-40, 4:29-35.

238. Thomas additionally discloses that the simulator receives the second one or more object representations generated during simulation mode, and monitors and analyzes the circumstances of the game, including the spatial location of other players in a sports game or the virtual road conditions in a racing game, to compare the object representations generated during simulation mode to those recorded during learning mode:

> During simulation, or playback, mode, as shown in FIG. 3b, when a first user desires to interact with a particular second user via the computer program 110, the simulator 120 retrieves the second user's profile (action block 600), if it has been created. Then, ***the simulator 120 monitors and analyzes the operation of the computer program in search of particular events and/or situations*** (action block 610).
>
> ***
>
> During playback mode 600, when an inbound event occurs, the simulator 120 will retrieve the recorded data associated with the inbound event and also ***analyze the current situation, such as time, score, and spatial positions of defensive avatars, to determine which action to invoke that represents the simulated user***.

Id., 7:39-44.

Tesla Exhibit 1003.224

239. As explained above and in more detail below, POSITA would have understood that the simulator/processor receives the second object representation(s) that are generated during simulated/playback mode to compare the new circumstance to the circumstances recorded during learning mode. *See* Section below.

240. In the pump fake and pass out of shot examples, Thomas describes that the system generates a second one or more object representations comprising the defensive coverage (*i.e.*, the spatial position of the defensive avatars), which the processor receives to compare to the first object representations of the defensive coverage that were stored in learning mode:

> For a ***pass out of shot*** event, the simulator 120, during learn mode 550, records a shot type, ***which avatar passed, and which avatar was passed to***. For current circumstances, the simulator 120 ***records how open the receiving avatar was and how covered the passing avatar was***. During playback…simulator 120 will ***invoke a pass if the current circumstances meet the criteria***, e.g., how much the ***passing and receiving avatars are covered***.
>
> ***
>
> For a ***pump fake event***, during the learning mode 550, the simulator 120 ***records how often a user pump fakes***. For current circumstances, the simulator 120 ***records how covered the user was and user's locati***on. During playback mode 600, the simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and***

Tesla Exhibit 1003.225

the defensive coverage. If criteria is met, then simulator 120 ***will invoke a pump fake***.

Id., 7:55-64, 8:8-15.

e. [1.5] determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second one or more object representations and the first one or more object representations; and

Limitation [1.5]	Claimed Features
	Determining **1st instructions** from partial match between **1st OR** and **2nd OR**

241. Thomas discloses comparing the circumstances recorded during learning mode with the new circumstances generated during simulation mode, which includes comparing the first object representations stored during learning mode to the second object representations generated and received during simulation mode. Thomas describes that the data recorded and stored during learning mode "enables the analysis of a user's behavior, tendencies, and inclinations for a given event [and,] [t]hus…when the simulator 120 is in simulation mode, and a particular event and/or situation occurs, the simulator 120 can invoke a corresponding action based on the stored data, thereby simulating the user." *Id.,* 5:24-29. If the object representations generated/received during simulation mode (*e.g.* the spatial positions of defenders) at least partially match the object representations previously recorded during learning mode (*e.g.* the spatial positions of defenders), the system determines that

the same avatar the user controlled during learning mode should execute the associated instruction set:

> For a catch and shoot event, during the learning mode 550, ***the simulator 120 records how often a user catches and shoots right after the catch and records, as a current circumstance, the location of the user's avatar that catches and shoots***. During playback mode 600, the simulator 120 will monitor when the simulated user catches the ball and analyze the user's location on the basketball court. ***If the simulated user is located in the typical spot for catch and shoot, then the simulator 120 will have the simulated user immediately shoot*** the ball after the catch.
>
> ***
>
> During playback mode 600, when an inbound event occurs, the simulator 120 will retrieve the recorded data associated with the inbound event and also ***analyze the current situation, such as time, score, and spatial positions of defensive avatars, to determine which action to invoke that represents the simulated user.***

Id., 7:65-8:7, 7:39-44.

242. As described above for the pump fake and pass out of shot examples, Thomas describes determining either a pump fake or pass instruction in response to detecting at least a partial match between the first object representations (*i.e.* the spatial position of defenders recorded during learning mode) and the second object representations (*i.e.*, the spatial position of defenders generated and received during simulation mode. *Id.*, 8:8-15 ("[T]he simulator 120 ***records how covered the user***

Tesla Exhibit 1003.227

was and user's location. During playback mode 600, the simulator…***analyze the location of the simulated user and the defensive coverage***. ***If criteria is met, then simulator 120 will invoke a pump fake***."), 7:55-64 (describing determining a pass instruction when defensive coverage generated during simulation mode matches coverage recorded during learning mode).

f. [1.6] at least in response to the determining, causing the first avatar of the application or a second avatar of the application to perform one or more operations defined by the first one or more instruction sets for operating the first avatar of the application at least by executing the first one or more instruction sets for operating the first avatar of the application.

Limitation [1.6]	Claimed Features
	1st instructions executed by **1st avatar** or **2nd avatar**

243. As I explained above for limitation [1.5], Thomas discloses determining the instruction sets for operating an avatar by comparing the new circumstances generated during simulation mode with the circumstances recorded during learning mode, including a comparison of object representations. Thomas also discloses causing the first avatar of the application to perform the instruction set for operating the first avatar at least by executing the determined instruction set.

244. Thomas describes that in simulation mode, when the system detects a particular situation or event (*i.e.* matches the object representations), the system simulates a user by invoking the corresponding instruction set recorded during

learning mode. For example, the system causes the first avatar (*i.e.*, the avatar being simulated based on the user's inputs during learning mode) to perform the stored instruction set, such as pump faking or passing the ball:

> For a pump fake event, during the learning mode 550, the simulator 120 records how often a user pump fakes. For current circumstances, the simulator 120 records how covered the user was and user's location. During playback mode 600, the simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and the defensive coverage. If criteria is met, then simulator 120 will invoke a pump fake.***
>
> ***
>
> During playback 600, for a pass out of shot event, the simulator 120 will analyze how well the passing avatar and the receiving avatar are covered. ***The simulator 120 will invoke a pass if the current circumstances meet the criteria,*** e.g., how much the passing and receiving avatars are covered.

Id., 8:8-15, 7:59-64.

245. Although it is not required by this limitation, Thomas also discloses a "second avatar of the application," which can be any second player on the user-controlled team in a sports game during learning mode that the system simulates during simulation mode. *Id.,* 7:2-4 ("For example, the computer program 110 can be ***a team sports video game, such as football, basketball, hockey, baseball***, etc.").

Tesla Exhibit 1003.229

2. Claim 2

a. [2.1] The system of claim 1, wherein the first one or more object representations include: one object representation, a stream of object representations, a collection of object representations, or a stream of collections of object representations,

Limitation [2.1]	Claimed Features
	1st OR include at least one OR

246. This limitation is satisfied if the first object representation(s) include any of the listed embodiments. Thomas describes that the object representations recorded during learning mode can include one object representation (*e.g.* the location of a single defender) or a collection of object representations (*e.g.* the locations of multiple defensive players). In the pump fake example, Thomas describes at least one object representation associated with covering the user controlled avatar:

> For a pump fake event, during the learning mode 550, the simulator 120 records how often a user pump fakes. ***For current circumstances, the simulator 120 records how covered the user was and user's location***.

Id., 8:8-11. In the pass out of shot example, Thomas discloses a collection of object representations (*i.e.*, at least one defender guarding the passing avatar and one defender guarding the avatar that receives the pass):

> For a ***pass out of shot*** event, the simulator 120, during learn mode 550, records a shot type, ***which avatar passed, and which avatar was passed***

to. For current circumstances, the simulator 120 ***records how open the receiving avatar was and how covered the passing avatar was***.

Id., 7:55-59.

b. [2.2] and wherein the second one or more object representations include: one object representation, a stream of object representations, a collection of object representations, or a stream of collections of object representations,

Limitation [2.2]	Claimed Features
	2nd OR include at least one OR

247. This limitation is satisfied if the second object representation(s) include any of the listed embodiments. Thomas describes that the object representations generated or received during simulation mode can include one object representation (*e.g.* the location of a single defender) or a collection of object representations (*e.g.* the locations of multiple defensive players). As explained above, the pump fake example includes at least one second object representation (one defender covering the pump faking avatar), while the pass out of shot example includes a second object representation that is a collection of object representations (one defender covering the passing player and one defender covering the receiving player):

> During playback mode 600, the simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and the defensive coverage. If criteria is met, then simulator 120 will invoke a pump fake.***

***

During playback 600, for a pass out of shot event, the simulator 120 will analyze how well the passing avatar and the receiving avatar are covered. ***The simulator 120 will invoke a pass if the current circumstances meet the criteria,*** e.g., how much the passing and receiving avatars are covered.

Id., 8:8-15, 7:59-64.

c. [2.3] and wherein the one or more objects of the application represented by the first one or more object representations are detected at a first time or during a first time period, and wherein the one or more objects of the application represented by the second one or more object representations are detected at a second time or during a second time period.

Limitation [2.3]	Claimed Features
	1st OR and **2nd OR** detected at different times

248. As I explained above for Limitation [1.3], the object(s) represented by the first object representation(s) (*e.g.* the spatial positions of avatars during learning mode) are detected and recorded during learning mode. The object(s) represented by the second object representation(s) (*e.g.* the spatial positions of avatars during simulation mode) are detected and recorded during simulation mode. A POSITA would have understood that the simulator detects and records the first object representations in learning mode during a first time period and generates and detects

the second object representations in simulation mode during a separate, second time period.

3. Claim **3**

a. [3.1] The system of claim 1, wherein the determining the first one or more instruction sets for operating the first avatar of the application based on the at least partial match between the second one or more object representations and the first one or more object representations includes:

249. As I discussed above for limitation [1.5], the first instruction set(s) for operating the first avatar of the application are determined based on partial match between the object representation(s) generated or received during simulation mode (second one or more object representations) and the object representation(s) recorded during learning mode (first one or more object representations). *See* Section X.B.1.e above.

b. [3.2] determining that a number of at least partially matching portions of the second one or more object representations and portions of the first one or more object representations exceeds a threshold number or determining that a percentage of at least partially matching portions of the second one or more object representations and portions of the first one or more object representations exceeds a threshold percentage.

Limitation	Additional Features to Claim 1
[3.2]	Partial match exceeds a threshold

250. The ’134 Patent specification describes that the threshold can comprise any percentage or number, including one. *See* ’134 Patent, 115:2-7 (“[S]ubstantial

Tesla Exhibit 1003.233

similarity can be achieved when at least a threshold number (i.e. 1, 2, 3, 6, 11, etc.) or percentage (i.e. 55%, 61%, 78%, 82%, 99%, etc.) of Object Properties 630 or portions thereof…match or substantially match.").

251. Thomas discloses and/or renders obvious that if the number of matching portions between the first and second object representations (*e.g.* the spatial positions of avatars generated during simulation mode and recorded during learning mode) exceeds a threshold number, the simulator executes the associated instruction set. As described above, during simulation mode the system analyzes the defensive coverage (i.e., the second object representations) and assesses whether it at least partially matches the defensive coverage recorded during learning mode (i.e., the first object representations). A POSITA would have understood and it would have been obvious that determining whether the criteria is met (i.e., whether there is at least a partial match) would involve determining whether the matching portions exceed a numerical threshold, such as the number of defenders and the distance of the defenders from the simulated avatar (i.e., the first avatar), as described in the pump fake (i.e., the criteria) and pass out of shot (i.e., how much the avatars are covered) examples:

> During playback mode 600, the simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and the defensive coverage. If criteria is met***, then simulator 120 will invoke a pump fake.
> ***

Tesla Exhibit 1003.234

> During playback 600, for a pass out of shot event, the simulator 120 will analyze how well the passing avatar and the receiving avatar are covered. ***The simulator 120 will invoke a pass if the current circumstances meet the criteria,*** e.g., ***how much*** the passing and receiving avatars are covered.

Ex. 1005, 8:11-15, 7:59-64.

4. Claim 4

a. [4.1] The system of claim 1, wherein the first correlation is included in a knowledgebase,

Limitation	Claimed Features
[4.1]	Knowledgebase

252. The '134 Patent specification describes a knowledgebase as comprising or including "any data structure or arrangement capable of storing knowledge of Avatar's 605 operation in circumstances including objects with various properties." '134 Patent, 103:42-45.

253. Thomas discloses a database storing user profiles (*i.e.*, a knowledgebase), which stores correlations between situations/events (i.e., first object representations) and instructions for operating an avatar:

> FIG. 1 shows a diagram of an interactive computer system 100 that includes an interactive computer program 110…that is configured to simulate a particular user's decisions and behavior, and a ***database 130 coupled with the simulator 120, which stores data used by the simulator 120 to calculate and generate the decisions and behavior***.

Tesla Exhibit 1003.235

***

During the learn mode, if a first user is using the computer program 110, the simulator 120 collects and evaluates data about the first user's input when interacting with the computer program 110 (action block 200). The collected data is then stored in the database 130 (action block 210). ***The data is stored in a data structure, generally referred to as a user's profile, corresponding to the first user. Generally, the data is logically linked to an event, or situation, that initiated the user's input.***

Ex. 1005, 3:41-49, 3:53-61.

254. A POSITA would have understood that a database comprises a knowledgebase because the database is a data structure capable of storing of an avatar's operations in various circumstances.

b. [4.2] and wherein the knowledgebase further includes a third correlation including a third one or more object representations correlated with a third one or more instruction sets for operating: the first avatar of the application, the second avatar of the application, a third avatar of the application, or an avatar of another application, and

Limitation [4.2]	Claimed Features
	3rd correlation (3rd OR + 3rd instructions)

255. Thomas discloses that the knowledgebase includes multiple correlations, including a first, second and third correlation, for autonomously operating an avatar in different situations during the game, each including the user's instructions for operating an avatar and the correlated object representations (such

Tesla Exhibit 1003.236

as the spatial positions of the avatars). As I explained above, Thomas discloses storing correlations associated with a pump fake (first correlation) and pass out of shot (second correlation event), and further discloses storing correlations (i.e., at least a third correlation) for other events, such as an inbound, offensive rebound, etc.

> [A]n event may be ***an inbound event, an offensive rebound event, a pass out of shot event***…. ***During each of these decisions made by the user, the simulator 120 will record the current circumstances***, for example, ***whether the defense is playing a full court press*** or how much time is left in the game, which may indicate whether a user will hurry the inbound pass.

Id., 7:22-38.

256. Thomas additionally discloses that the simulator can store multiple correlations, including a first, second and third correlation, for the same situation encountered at different times, where the situation is correlated with different commands (i.e., instruction sets) that the user provides in response each time the event occurs:

> ***For example, in the case of a football game, a player's avatar is running down the sideline and encounters an opposing player's avatar (an event).*** In one hundred (100) occurrences of the event, ***if the user spun left twenty (20) times, right thirty (30) times, and juked fifty (50) times, the simulator 120 will assign values of twenty (20), thirty (30), and fifty (50) to the respective responses.*** Alternatively, each response can be assigned a range of values, such as 1-20, 21-50, and 51-100, respectively, or 1-2, 3-5, and 6-10….

Tesla Exhibit 1003.237

Id., 6:16-27. In the example above, there are three correlations, each of which includes a different response (juking, spin left, spin right) to the same situation/event.

c. [4.3] wherein the third one or more object representations represent one or more objects of: the application, or the another application, and

Limitation [4.3]	Claimed Features
	3rd OR represents objects of **the application** or **another application**

257. As I described above for limitation [4.2], Thomas discloses at least three correlations, each of which includes one an object representation that represents one or more objects of the application (i.e., spatial position of defenders) correlated with a response to the situation/event. In the example above, the correlations associated with (1) juke, (2) spin left, and (3) spin right each includes an object representation of the spatial position of the defensive players. A POSITA would understand that this teaching would also apply to a basketball game, where the system would store at least three correlations corresponding to a particular situation/event, each of which would include different instruction sets (*e.g.*, pump fake, shot, pass) for responding to a situation/event (*e.g.*, the spatial position of defenders at a certain point during the game), as explained for Limitation [4.2].

d. [4.4] wherein the first correlation is connected with the third correlation **by a connection.**

Limitation [4.4]	Claimed Features
	Connection between **1st correlation** and **3rd correlation**

258. Thomas discloses a connection between correlations, including between a first correlation and a third correlation.

259. The '134 Patent specification describes that a connection may include or be associated with a numeric attribute or parameter, such as weight or occurrence count. '134 Patent, 105:35-38 ("Connection 853 may include or be associated with a value such as a symbolic label or ***numeric attribute*** (i.e. weight, cost, capacity, length, etc.)."); *see also id*., 107:50-53 ("Each of Connection 853z1 and Connection 853z2 may include or be associated with ***occurrence count, weight, and/or other parameters or data***.").

260. Thomas discloses that in learning mode, the simulator records how often a user responds to a particular situation by invoking a certain action. If a user encounters the same situation more than once, the user may invoke different actions on different occasions. When the user has provided multiple responses to a given situation (*e.g.*, the same spatial position of defenders), the simulator generates multiple correlations for the situation, each of which would correlate that situation with a different response that the user has provided during learning mode. As described below, Thomas describes that in this situation the system connects the

Tesla Exhibit 1003.239

three correlations via a numerical occurrence value, which the system uses to determine the instruction set to invoke when the situation occurs during simulation mode:

> If there is more than one value (decision block 630), then the actual second user may have invoked more than one type of response to the particular situation or event while being recorded. In such a case, the simulator 120 can randomly select from one of the values, or ***the simulator 120 can select its response based on statistics or mathematical formula, e.g., based on the number of times the actual user invoked one response over another (action block 635). Such statistics may indicate tendencies demonstrated by the simulated user.***
>
> ***
>
> [T]he ***simulator 120 can record each type of response and apply a statistical, or weighted, value to a response based on its statistical occurrence compared to other types of responses***…. ***[S]imulator 120 will associate one or more values with each different type of user input based on the statistical occurrence of each input compared to other inputs*** (action block 577)…. In one hundred (100) occurrences of the event, if the user spun left twenty (20) times, right thirty (30) times, and juked fifty (50) times, the simulator 120 ***will assign values of twenty (20), thirty (30), and fifty (50) to the respective responses***.
>
> ***
>
> ***During simulation mode, the simulator 120 can randomly invoke a response to a situation and/or event based on weighted statistical***

Tesla Exhibit 1003.240

> ***values of the stored inputs***…. The stored input associated with the selected value will then be invoked. In the example above, ***when the simulated opponent's avatar is running down the sideline and encounters an opposing player's avatar and each of the above responses are given the respective range of values, the simulator 120 will randomly select a number from 1 to 100. If the number 53 is selected, then the avatar will juke (51-100 are associated with juke).***

Ex. 1005, 5:63-4, 6:7-23, 6:28-41. As explained above, a POSITA would understand that this teaching would also apply to a basketball video game, wherein an avatar may shoot, pump fake, or pass at different times in response to the same situation (*i.e.*, spatial position of defensive avatars).

5. Claim 5: The system of claim 1, wherein the second avatar of the application performs the one or more operations defined by the first one or more instruction sets for operating the first avatar of the application.

Claim 5	Claimed Features
	2nd avatar executes **1st instructions** for **1st avatar**

261. Thomas discloses and/or renders obvious claim 5. Thomas describes operating a particular avatar through random or default instructions when the game encounters a situation for which there is no correlated instruction set. *Id.,* 5:54-58 ("***If there is no value, i.e., stored user input***, recorded for the given the event and/or situation, then the simulator 120 will generate a random or ***default response*** representing the second user's simulated input, or response."). A POSITA would

Tesla Exhibit 1003.241

have understood and it would have been obvious that a "default response" would be an instruction set (ex. preprogrammed directions for operating an avatar in response to a situation) that is correlated with one or more object representations (ex. the spatial position of avatars in the situation in response to which the response is invoked). As explained above, a POSITA would have understood that a pre-programmed default response would be correlated with a situation (*e.g.*, spatial position of defensive avatars and the simulated avatar) so that the simulated avatar would operate autonomously in an appropriate manner (*i.e.*, not shooting the basketball when positioned at the opposite end of the court from the basket). Thomas additionally discloses that avatars representing basketball players can be classified as one of several archetypes, and that instruction sets are associated with the archetype of the avatar that initiated the event. *Id.,* 4:45-51 ("Another type of parameter, in the case of a computer game involving one or more avatars, is the archetype….For example, an archetype can be defined by relative size, strength, and speed…..***[A]rchetypes can include true point guard, slasher, shooter, pure shooter, low post scorer, midrange scorer, and role player***."), 8:38-40 ("For each event, during learning mode 550, the ***simulator 120 records the archetype of the avatar that initiated a monitored event***.").

262. A POSITA would have understood and it would have been obvious to a POSITA that avatars of the same archetype (ex. shooters) would follow the same

Tesla Exhibit 1003.242

default instructions in the absence of a learned instruction set. Accordingly, when a second avatar possesses the ball in a game and encounters a situation for which there is no associated correlation in simulation mode, a second avatar would perform the same instruction set that a first avatar would perform when encountering a situation in simulation mode for which there is no stored correlation.

263. Thomas also renders this limitation obvious for a second reason. In particular, Thomas discloses storing instruction sets correlated with object representations for a first avatar that are learned in learning mode, and that the system records the archetype associated with that avatar, as described above. *See* Section X.B.1.c above. It would have been obvious to a POSITA that a learned instruction set for a first avatar would be applied to a second avatar of the same archetype if there is no stored correlation for the second avatar, such as multiple avatars that are shooters, low post scorers, etc. For example, Thomas discloses that the system can invoke a stored instruction set in response to a situation based on time, score, and the spatial position of the defense without requiring that the instruction set be performed by the identical avatar. *See e.g.*, Ex. 1005, 7:39-44 ("During playback mode 600…the simulator 120 will…analyze the current situation, such as time, score, and spatial positions of defensive avatars, to determine which action to invoke that represents the simulated user."). This disclosure demonstrates that Thomas discloses and/or renders obvious applying an instruction

Tesla Exhibit 1003.243

set for a first avatar (*i.e.*, first object) to a second avatar (*i.e.*, second object) when certain situation parameters sufficiently match, even if the avatars are not the same.

6. Claim 6:

a. [6.1] The system of claim 1, wherein the one or more objects of the application represented by the first one or more object representations include one or more objects of the application in the first avatar's surrounding,

Limitation [6.1]	Claimed Features
	1st OR includes objects in surroundings of **1st avatar**

264. Thomas discloses that the first object representations represent objects in a first avatar's surroundings. For example, the first object representations recorded during learning mode can include object representations of other player avatars in the surroundings of the user-controlled avatar (*i.e.*, the first avatar), such as defenders:

> For example, in the game of basketball, if a user is controlling a basketball player near the three-point line and ***sees two defenders, one on each side,*** the user may choose to shoot a jump shot or split the defenders-***responses that may be recorded in the profile 650 and linked with opponent's avatar's position.***
>
> ***
>
> For a pass out of shot event, the simulator 120, during learn mode 550, records a shot type, which avatar passed, and which avatar was passed to. ***For current circumstances, the simulator 120 records how open the receiving avatar was and how covered the passing avatar was***.

Tesla Exhibit 1003.244

***

For a pump fake event, during the learning mode 550, the simulator 120 records how often a user pump fakes. ***For current circumstances, the simulator 120 records how covered the user was and user's location***.

***

[A]n event may be an inbound event, an offensive rebound event, a pass out of shot event…. ***During each of these decisions made by the user, the simulator 120 will record the current circumstances, for example, whether the defense is playing a full court press*** or how much time is left in the game, which may indicate whether a user will hurry the inbound pass.

Id., 5:13-18, 7:55-59, 8:8-11, 7:22-38.

b. [6.2] and wherein the one or more objects of the application represented by the second one or more object representations include: one or more objects of the application in the first avatar's surrounding, or one or more objects of the application in the second avatar's surrounding, and

Limitation [6.2]	Claimed Features
	2nd OR includes objects in surroundings of **1st avatar** or **2nd avatar**

265. Thomas discloses that the second object representations represent objects of the application in the first avatar's surroundings. For example, the second object representations generated during simulation mode include representations of

Tesla Exhibit 1003.245

defensive avatar players (*i.e.*, objects of the application) in the surroundings of the simulated avatar that was user controlled during learning mode (*i.e.*, the first avatar):

> For a pump fake event, during the learning mode 550, the simulator 120 records how often a user pump fakes. For current circumstances, the simulator 120 records how covered the user was and user's location. During playback mode 600, ***the simulator 120 will monitor for a shoot event and analyze the location of the simulated user and the defensive coverage.*** If criteria is met, then simulator 120 will invoke a pump fake.
>
> ***
>
> During playback 600, for a pass out of shot event, ***the simulator 120 will analyze how well the passing avatar and the receiving avatar are covered.*** The simulator 120 will invoke a pass if the current circumstances meet the criteria, e.g., how much the passing and receiving avatars are covered.

Id., 8:8-15, 7:59-64.

266. Thomas additionally discloses that the object representations can include objects in the surroundings of a second avatar. In simulation mode, the simulator can operate multiple avatars on a simulated user's team, including a second avatar. *Id.,* 7:2-4 ("For example, the computer program 110 can be a ***team sports video game, such as football, basketball, hockey, baseball***, etc."). The generated second object representations include representations of opposing players around a second avatar on the simulated team, such as a second avatar on a basketball team that receives a pass in the example above.

Tesla Exhibit 1003.246

c. [6.3] wherein the first avatar's surrounding includes at least one of: a part of the application in an area of interest around the first avatar, a part of the application defined by a threshold distance from the first avatar, a part of the application relative to the first avatar, a part of the application around the first avatar, a part of the application that is shown to a user, a part of the application that is visible to a user, or a part of the application,

[6.4] and wherein the second avatar's surrounding includes at least one of: a part of the application in an area of interest around the second avatar, a part of the application defined by a threshold distance from the second avatar, a part of the application relative to the second avatar, a part of the application around the second avatar, a part of the application that is shown to a user, a part of the application that is visible to a user, or a part of the application.

Limitation	Claimed Features
[6.3] and [6.4]	Surroundings = one of the listed embodiments

267. Thomas discloses that the first avatar's surroundings include an area of interest around the first avatar in the application, such as the position of the defensive players near the avatar on a basketball court or football field, as described in the pump fake and pass out of show examples above. A POSITA would understand that the areas of interest are also visible to the user in learning and simulation mode because the players and the virtual field or court are depicted on the interface screen:

> During playback mode 600, the simulator 120 will monitor for a shoot event and analyze the location of the simulated user and the defensive coverage. If criteria is met, then simulator 120 will invoke a pump fake.

Tesla Exhibit 1003.247

> ***
>
> During playback 600, for a pass out of shot event, ***the simulator 120 will analyze how well the passing avatar and the receiving avatar are covered.***
>
> ***
>
> Turning to FIG. 5, a diagram representing a video game interface 300 is shown. ***The interface 300 generally includes a graphical representation of a football field 310, and two sets of avatars 320, each representing a football team set to oppose each other***. One of the sets of avatars 320 represent a defensive team, or team on defense 330, wherein each avatar 320 represents a defensive player, i.e., defensive linemen, linebackers, and defensive backs. The other set of avatars 320 represents an offensive team, or team on offense 340, wherein each avatar 320 represents an offensive player, i.e., offensive linemen, backs, receivers, and a quarterback.
>
> ***
>
> ***The simulator 120 further defines a logical half circle behind the ball carrier on the football field 310. If one of the two closest defending avatars 320 is in this area, the simulator 120 sets a bit in an index in the special moves profile 460 to indicate this as well.***

Id., 8:11-15, 7:59-61, 8:66-9:9, 13:7-11.

268. Thomas also discloses that the second avatar's surroundings include an area of interest around the second avatar (*e.g.*, the avatar that receives a pass in a

Tesla Exhibit 1003.248

basketball game). Thomas discloses that the application can include multiple avatars, including a first and a second avatar, representing different players on a football or basketball team. The second avatar's surroundings also include the virtual field or court and the defensive players thereon and are displayed on the interface screen, which is an area of interest around the second avatar in the application that is visible to the user, as discussed above.

7. Claim 7: The system of claim 1, wherein, to correlate the first one or more object representations with the first one or more instruction sets for operating the first avatar of the application, a determination is made that the first one or more instruction sets for operating the first avatar of the application temporally correspond to the first one or more object representations.

Claim 7	Claimed Features
	Temporal correspondence between **1st instructions** and **1st OR**

269. The specification of the '134 Patent describes that instruction sets that "temporally correspond" with object representations includes instruction sets executed at the time the first object representations are generated. '134 Patent, 101:38-43 ("Instruction Sets 526 and/or Extra Info 527 ***that temporally correspond*** to a Collection of Object Representations 525 may include Instruction Sets 526 used and/or Extra Info 527 ***obtained at the time*** of generating the Collection of Object Representations 525."); *id.*, 29:4-8 ("The one or more instruction sets that temporally correspond to the first stream of collections of object representations may include

Tesla Exhibit 1003.249

one or more instruction sets ***executed at a time of generating the first stream of collections of object representations***."); *see also id.,* 101:49-55 ("For example, Instruction Sets 526 and/or Extra Info 527 that temporally correspond to a Collection of Object Representations 525 may include Instruction Sets 526 used and/or Extra Info 527 obtained within 50 milliseconds, 1 second, 3 seconds, 20 seconds, 1 minute, 41 minutes, 2 hours, or any other time period before and/or after generating the Collection of Object Representations 525.").

270. Thomas discloses that a determination is made that the first one or more instruction sets for operating the first avatar of the application temporally correspond to the first one or more object representations. In particular, Thomas describes that during learning mode the avatar executes the user's commands at the same time object representations representing the game situation are generated. For example, in a game of basketball, an avatar may execute a user's command to pump fake or pass at the same time the defenders are generated and displayed on the interface screen (i.e., when the first object representations are generated), which are also recorded by the simulator at the same time:

> During learning mode 550, as shown in FIG. 3a, the simulator 120 ***records a user's input as the user is interacting with the game***.
>
> ***
>
> For a pass out of shot event, the simulator 120, during learn mode 550, records a shot type, which avatar passed, and which avatar was passed

Tesla Exhibit 1003.250

> to. ***For current circumstances, the simulator 120 records how open the receiving avatar was and how covered the passing avatar was***.
>
> ***
>
> For a pump fake event, during the learning mode 550, the simulator 120 records how often a user pump fakes. ***For current circumstances, the simulator 120 records how covered the user was and user's location***.

Ex. 1005, 9:35-36, 7:55-59, 8:8-11.

271. Additionally, Thomas discloses that a determination is made that the instruction sets temporally correspond to the first object representations because it describes that the system stores temporal information associated with each correlation, such as the specific time in the game (*i.e.* temporal information). *Id.*, 4:29-31 ("One common parameter that defines a situation may be time 660, e.g., time left in a particular session or game, and/or time played in a particular session or game."). In simulation mode, the simulator analyzes temporal information to determine the appropriate instruction set for a situation or event:

> For example, in the case of a football game, if the simulated user is on its own twenty yard line and just failed to reach the first down mark after a third down play, e.g., it is fourth down and one, the simulator 120 can analyze the time left on the play clock and the score. ***If the time left on the play clock is less than two minutes, and simulated user is losing by 5 points, then the simulator 120 will analyze such a situation to determine whether the user would kick a field goal or try to reach the end zone for a touchdown***.

Tesla Exhibit 1003.251

***

During each of these decisions made by the user, the simulator 120 will record the current circumstances, for example, whether the defense is playing a full court press or how much time is left in the game, which may indicate whether a user will hurry the inbound pass. During playback mode 600, when an inbound event occurs***, the simulator 120 will retrieve the recorded data associated with the inbound event and also analyze the current situation, such as time***, score, and spatial positions of defensive avatars***, to determine which action to invoke that represents the simulated user.***

Id., 5:45-54, 7:34-44.

8. Claim 8

a. [8.1] The system of claim 1, wherein at least some elements of the system are included in: a single device, or multiple devices,

Limitation	Claimed Features
[8.1]	Single device or multiple devices

272. As I explained above for limitation [17.1] of the '585 Patent, Thomas discloses that elements of the system can be included in a single device or multiple devices. *See* Section IX.B.15.a above.

b. [8.2] and wherein the one or more processors include: one or more microcontrollers, one or more computing devices, or one or more electronic devices,

	Claimed Features

Tesla Exhibit 1003.252

Limitation [8.2]	Processors = Microcontrollers, computing devices or electronic devices

273. As I explained above for limitation [17.2] of the '585 Patent, Thomas discloses processors including electronic devices. *See* Section IX.B.15.b above.

c. [8.3] and wherein the first correlation is stored in or on at least one of: at least one non-transitory machine readable medium of the one or more non-transitory machine readable media, another one or more non-transitory machine readable media, one or more volatile memories, one or more non-volatile memories, one or more storage devices, or one or more storage systems,

Limitation [8.3]	Claimed Features
	1st correlation stored on storage system, storage device or other embodiments

274. As I explained above for limitation [17.3] of the '585 Patent, Thomas discloses a first correlation stored in a storage device or storage system, such as a memory or servier. *See* Section IX.B.15.c above (explaining that the database can be stored in a portable memory device or server); *see also* Section X.B.1.c (explaining that the correlations are stored in a database).

d. [8.4] and wherein the application includes: a video game, a computer game, a simulation program, a program including text processing, a program including number processing, a program including picture

processing, a program including object processing, or a program,

Limitation [8.4]	Claimed Features
	Application = video game, computer game or other embodiment

275. As I explained above for limitation [17.6] of the '585 Patent, Thomas discloses that the application program can comprise a video game or a computer game. *See* Section IX.B.15.f above.

e. [8.5] and wherein the application includes: one or more versions of the application, one or more upgrades of the application, one or more sequels of the application, one or more instances of the application, or one or more variations of the application,

Limitation [8.5]	Claimed Features
	Application = version of the application or other embodiment

276. Thomas discloses that the application program can comprise a version of a video game or a computer game, such as NFL 2K5 or a basketball video game. Ex. 1005, 8:55-60 ("Another example embodiment of a computer system 100 is a video game system that simulates an American football game and enables a first user to play the game against a computer opponent or a second user, and which is embodied in ***Visual Concepts' ESPN NFL 2K5™ ("NFL 2K5") video game, a product currently available in the market***.").

Tesla Exhibit 1003.254

f. [8.6] and wherein the first avatar of the application is a first object of the application and the second avatar of the application is a second object of the application,

Limitation [8.6]	Claimed Features
	1st avatar = 1st object **2nd avatar= 2nd object**

277. As I explained above for claim 1, the first and second avatars comprise objects of the application. For example, the first and second avatars can represent players in a football or basketball video game application. *See* Section X.B.1.c above.

g. [8.7] and wherein the first one or more object representations include one or more properties of the one or more objects represented by the first one or more object representations, and wherein the second one or more object representations include one or more properties of the one or more objects represented by the second one or more object representations, and

Limitation [8.7]	Claimed Features
	1st OR includes properties of represented objects **2nd OR** includes properties of represented objects

278. The '134 Patent specification describes that a property of an object can include the object's location. '134 Patent, 70:32-40 ("***Examples of Object Properties 630 (i.e. also referred to simply as object properties, etc.) include*** existence of Object 615, type of Object 615 (i.e. person, animal, tree, rock, building, vehicle, etc.), identity of Object 615 (i.e. name, identifier, etc.), distance of Object 615,

bearing/angle of Object 615, ***location of Object 615*** (i.e. distance and bearing/angle from a known point, coordinates, etc.), shape/size of Object 615 (i.e. scale, height, width, depth, computer model, etc.)…"). As I explained above for claim 1, Thomas discloses that the first one or more object representations recorded during learning mode and the second one or more object representations generated during simulation mode include one or more properties of objects of the application, such as the archetypes of basketball players or the spatial positions (i.e., location) of players on the football field or basketball court. *See* Section X.B.1.c-d and Section X.B.5 above.

h. [8.8] wherein an instruction set of the first one or more instruction sets for operating the first avatar of the application includes at least one of: only one instruction, a plurality of instructions, one or more inputs, one or more commands, one or more computer commands, one or more keywords, one or more symbols, one or more operators, one or more variables, one or more values, one or more objects, one or more object references, one or more data structures, one or more data structure references, one or more functions, one or more function references, one or more parameters, one or more signals, one or more characters, one or more digits, one or more numbers, one or more user operating directions, one or more user directions, one or more user inputs, one or more representations of one or more user actions, one or more representations of one or more user clicks, one or more binary bits, one or more assembly language commands, one or more states, one or more state representations, one or more codes, one or more data, or one or more information, and

	Claimed Features

Tesla Exhibit 1003.256

Limitation [8.8]	Instruction set includes instructions, inputs, commands or other listed embodiment

279. This limitation is satisfied by an instruction set that includes one or more instructions, inputs, or commands. Thomas discloses the first one or more instruction set(s) for operating the first avatar include at least one instruction, a plurality of instructions or at least one input or command. In particular, Thomas describes that the first one or more instruction sets includes "a pressed button, combination of buttons, or joystick move of a controller, in response to the particular situation and/or event." Ex. 1005, 4:64-5:5. Thomas further describes that these instructions, inputs, or commands include an instruction to pump fake or pass the basketball to a certain player:

> For a pass out of shot event, the simulator 120, during learn mode 550, ***records a shot type, which avatar passed, and which avatar was passed to***.
>
> ***
>
> For a ***pump fake event***, during the learning mode 550, the simulator 120 ***records how often a user pump fakes***.

Id., 7:55-57, 8:8-9.

i. [8.9] wherein: an object of the application of the one or more objects of the application represented by the first one or more object representations is the same as an object of the application of the one or more objects of

Tesla Exhibit 1003.257

the application represented by the second one or more object representations, multiple objects of the application of the one or more objects of the application represented by the first one or more object representations are the same as multiple objects of the application of the one or more objects of the application represented by the second one or more object representations, all objects of the application of the one or more objects of the application represented by the first one or more object representations are the same as all objects of the application of the one or more objects of the application represented by the second one or more object representations, or all objects of the application of the one or more objects of the application represented by the first one or more object representations are different than all objects of the application of the one or more objects of the application represented by the second one or more object representations.

Limitation [8.9]	Claimed Features
	1st OR and **2nd OR** represent same object(s) or other embodiments

280. This limitation can be satisfied through any of the listed embodiments. Thomas discloses that (1) an object of the application of the one or more objects of the application represented by the first one or more object representations is the same as an object of the application of the one or more objects of the application represented by the second one or more object representations, and that (2) multiple objects of the application of the one or more objects of the application represented by the first one or more object representations are the same as multiple objects of

Tesla Exhibit 1003.258

the application of the one or more objects of the application represented by the second one or more object representation, both of which satisfy this limitation.

281. For example, during learning mode, the simulator records a player profile including information about other players on the field (first object representations), such as the location of defensive players during a pump fake or pass out of shot event. Ex. 1005, 7:55-64 ("For a pass out of shot…the simulator 120 records how open the receiving avatar was and how covered the passing avatar was."), 8:8-11 ("For a pump fake event…the simulator 120 records how covered the user was and user's location."). Thomas further describes that during simulation mode the simulator monitors the game for the same situation or event, which a POSITA would understand includes identifying at least one of the same defensive players (i.e., an object represented by the second one or more object representations) that correspond to those generated and recorded during learning mode:

> For a pump fake event, ***during the learning mode*** 550, the simulator 120 records how often a user pump fakes…***how covered the user was and user's location***. During playback mode 600, the simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and the defensive coverage. If criteria is met***, then simulator 120 will invoke a pump fake.

Id., 8:8-15.

282. Thomas similarly discloses that an object of the one or more first object representations is the same as an object of the one or more second object

Tesla Exhibit 1003.259

representations in the context of a football video game. In one example, Thomas describes a situation in which a user double teams the best receiver on the opposing team (an object) in learning mode, which the simulator simulates in simulation mode (i.e., when an object of the one or more second object representations is the same). *Id.,* 12:21-29 ("***For a defensive play calling profile 530, the simulator 120 may store*** whether and ***how a user adjusts defensive players for a particular play***, which avatar 320 the user typically controls, and where the user typically lines up the avatars 320. Other information may include when the user… makes adjustments to ***double team the best receiver on the opposing team***."). A POSITA would readily recognize that the teachings of the football example (*e.g.*, detecting that the "best receiver" is the same in the first and second object representations) would apply to the basketball game, such as how the simulated user would defend the opposing team in certain situations during the basketball game.

9. Claim 9: The system of claim 1, wherein the first correlation is included in a first knowledge cell, and wherein the first knowledge cell is a data structure for storing, structuring, or organizing the first correlation.

Claim 9	Claimed Features
	Knowledge cell + Knowledge cell = Data structure storing correlation

283. The '134 Patent specification describes that a knowledge cell can comprise "any data structure or arrangement" storing "one or more collections of object representations correlated with any instruction sets into the knowledge cell."

Tesla Exhibit 1003.260

’134 Patent, 154:28-31; *see id.,*154:28-36 (“Correlating may include structuring or generating a knowledge cell (i.e. Knowledge Cell 800, etc.) and storing one or more collections of object representations correlated with any instruction sets into the knowledge cell. Therefore, ***a knowledge cell may include any data structure or arrangement that can facilitate such storing***.”).

284. Thomas discloses a knowledge cell that is a data structure for storing, structuring, or organizing the first correlation because it describes that the system can include a data structure in the form of a user profile, which stores user inputs (i.e., instruction sets) linked (i.e., correlated) with sitautions/events (including object representations such as the spatial position of defenders).

> During learning mode, the simulator 120 records a user's input as the user is interacting with the game. Turning to FIG. 6, ***the inputs for a user are stored in user profiles 400, which, as described above, may be individual data structures in a database 130***. The general structure of a user profile 400 for this particular video game is shown in FIG. 6. ***A user profile includes sub-profiles, e.g., 410, each containing fields or additional sub profiles that store the user's inputs in an organized and logical manner. Generally, as mentioned above, the user's input is logically linked to an event and/or situation that initiated the user's input.***

Ex. 1005, 9:35-45.

Tesla Exhibit 1003.261

285. Thomas further describes that user profiles are stored "on a server 710 accessible by one or more client consoles 750/740." *Id.*, 19:39-43. A POSITA would have understood a user profile to comprise a knowledge cell as described by the '134 Patent.

10. Claim 10

286. Independent claim 10 is a method claim that corresponds to system claim 1. My analysis above explaining how Thomas discloses the limitations of claim 1 applies to the limitations of claim 19 as set forth in the chart below.

Claim 10	**Corresponding Analysis for Claim 1**
[10.1] A method implemented using a computing system that includes one or more processors, the method comprising:	*See* analysis for limitation [1.1] above. *See* Section X.B.1.a-b above
[10.2] accessing a memory that stores at least a first correlation including a first one or more object representations correlated with a first one or more instruction sets for operating a first avatar of an application, wherein the first one or more object representations represent one or more objects of the application;	*See* analysis for limitation [1.3] above. *See* Section X.B.1.c above
[10.3] generating or receiving a second one or more object representations, wherein the second one or more object representations represent one or more objects of the application;	*See* analysis for limitation [1.4] above. *See* Section X.B.1.d above

Tesla Exhibit 1003.262

Claim 10	Corresponding Analysis for Claim 1
[10.4] determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second one or more object representations and the first one or more object representations; and	*See* analysis for limitation [1.5] above. *See* Section X.B.1.e above
[10.5] at least in response to the determining, executing the first one or more instruction sets for operating the first avatar of the application, wherein the first avatar of the application or a second avatar of the application performs one or more operations defined by the first one or more instruction sets for operating the first avatar of the application.	*See* analysis for limitation [1.6] above. *See* Section X.B.1.f above

11. Claim 11

a. [11.1] The system of claim 1, wherein the first correlation is included in a knowledgebase,

287. As I explained above with respect to limitation [4.1], Thomas discloses a knowledgebase. *See* Section X.B.4.a above.

b. [11.2] and wherein the knowledgebase further includes a third correlation including a third one or more object representations correlated with a third one or more instruction sets for operating the first avatar of the application,

288. As I explained above with respect to limitation [4.2], Thomas discloses a knowledgebase including a third correlation between a third object representation(s) and a third instruction set(s). For example, Thomas discloses

Tesla Exhibit 1003.263

instruction sets for (1) a pump fake event, (2) pass out of shot event, and (3) a rebound or inbound. *See* Section X.B.4.b above (claim 1) (describing pump fake and pass out of shot events); Ex. 1005, 7:22-36 ("[A]n event may be ***an inbound event***, an offensive rebound event, a pass out of shot event…. ***During each of these decisions*** made by the user, the simulator 120 ***will record the current circumstances***….").

c. [11.3] and wherein the third one or more object representations represent one or more objects of the application, and

289. As I explained above with respect to limitation [11.2] and for limitations [4.2]-[4.3], Thomas discloses that the third correlation, such as the correlation associated with one of three situations in a basketball game, includes third object representations representing one or more objects of the application, such as defensive players. *See* Section X.B.4.b-c above.

d. [11.4] wherein at least a portion of the first correlation is learned in a learning process that includes operating the first avatar of the application at least partially by a user, and wherein at least a portion of the third correlation is learned in another learning process that includes operating the first avatar of the application at least partially by the user, and

Limitation [11.4]	Additional Features to Claim 4
	1st correlation and **3rd correlation** learned with operation of the **1st avatar** by **1st user**

Tesla Exhibit 1003.264

290. Thomas discloses a first learning process (*i.e.*, learning mode) in which a user controls an avatar and the simulator records the user's input in response to a situation or event (*i.e.*, a first correlation), which includes object representations such as the spatial position of avatars, such as when the user pump fakes during learning mode. Thomas at 5:5-7 ("A user's ***response to the occurrence of an event can differ depending upon the surrounding circumstances or situation***….").

> For a pump fake event, ***during the learning mode 550, the simulator 120 records how often a user pump fakes. For current circumstances, the simulator 120 records how covered the user was and user's location***.

Id., 8:8-11.

291. Thomas also discloses learning additional correlations for different situations and events (*i.e.*, second correlation and third correlation) in separate learning processes. As the user continues to interact with the system, the system records additional object representations and correlated instruction sets for subsequent, distinct game events, including a first, second and third correlations (*e.g.*, a pass out of shot event, inbound event, rebound event, etc.). *Id.,* 7:22-64 (describing recording situations and instruction sets during catch and shoot event, pump fake event, pass out of shot, and defensive steal event). Each recording comprises a separate learning process, including a first, second and third learning process:

Tesla Exhibit 1003.265

> ***A user's response to the occurrence of an event can differ depending upon the surrounding circumstances or situation. Thus, the data recorded can be associated with both an event and a situation.*** In other words, when a user responds to an event, the user's response, when recorded, is not only associated with the occurrence of that event, but also with the situation during the occurrence of that event, which is defined by one or more current circumstances.
>
> ***
>
> For example, in the case of a basketball game, ***an event may be an inbound event, an offensive rebound event, a pass out of shot event, a catch and shoot event, a pump fake event, defensive steal event, and/or a shot event***…. ***During each of these decisions made by the user, the simulator 120 will record the current circumstances….***

Id., 5:5-13, 7:22-36.

292. Thomas also discloses this limitation because it describes that when the user controls the avatar in different game sessions during learning mode, the system records additional correlations as described above. Each recording comprises a separate learning process, which a POSITA would understand includes learning first, second, and third correlations:

> ***During learning mode***, the simulator 120 retrieves the user profile 400 of the user to be simulated (action block 800). The computer system initializes the game system features, e.g., controllers, animations, collisions, behaviors, etc. (action block 810)…. If there is existing data in the playbook profile 500, then the simulator 120 will evaluate

Tesla Exhibit 1003.266

whether the data is up to date (action block 840). Preferably, data should evolve ***as the user's interaction evolves***. In other words, ***data recorded thirty games ago may have a lesser impact on the total data available.***

Id., 10:19-32. Thomas therefore discloses that the first correlation is learned in one learning process, while the third correlation can be learned in another learning process (*i.e.*, subsequent play of the same game or a different instance of the game), both of which include operating the first avatar of the application at least partially by the user.

e. [11.5] wherein the user is: a human user, or a non-human user.

Limitation [11.5]	Additional Features to Claim 4
	1st user is a human or non-human user

293. As I explained above with respect to limitation [2.4] of the '585 Patent, Thomas discloses a human user playing the video game or computer game, such as a human user[11] providing input commands to a controller. *Id.,* 4:67-5:3 ("the simulator 120 will record the user's decision/input, ***such as a pressed button,***

[11] A POSITA would readily understand that a human user provides the input commands to a controller. In the event Patent Owner contends that Thomas does not refer to a "human" user, a POSITA would understand the user is a non-human user.

Tesla Exhibit 1003.267

combination of buttons, or joystick move of a controller, in response to the particular situation and/or event").

12. Claim 12

a. [12.1] The system of claim 1, wherein the first correlation is included in a knowledgebase, and wherein the knowledgebase further includes a third correlation including a third one or more object representations correlated with a third one or more instruction sets for operating the first avatar of the application, and wherein the third one or more object representations represent one or more objects of the application, and

294. As I explained above in regards to limitation [4.1]-[4.3], Thomas discloses a knowledgebase storing a first and third correlation including a third one or more object representations correlated with a third one or more instruction sets for operating the first avatar of the application, where the third correlation includes a third object representation(s) representing one or more objects of the application. *See* Section X.B.4.a-c above.

b. [12.2] wherein at least a portion of the first correlation is learned in a learning process that includes operating the first avatar of the application at least partially by a user, and

295. As I explained above in regards to limitation [11.2], Thomas discloses that the first correlation is at least partially learned in a learning process where a user operates an avatar.

c. [12.3] wherein at least a portion of the third correlation is learned in another learning process that includes

Tesla Exhibit 1003.268

operating the first avatar of the application at least partially by another user,

Limitation [12.3]	Additional Features to Claim 11
	3rd correlation learned in **another learning process** with operation of the **1st avatar** by **another user**

296. Thomas discloses and/or renders obvious this limitation. Thomas describes that the memory can record and store profiles for multiple users, where each profile includes one or more correlations between the user's instructions for operating a particular avatar and a particular situation or event (including object representations of the spatial position of the defensive player avatars):

> ***Multiple users can compete against each other either on a single console 750/740 or from multiple console*****s** 750/740. ***For example, a first user can compete on a first video game console 760 against a second user competing on a second video game console*** 760. Whether the simulator 120 is on one of the client consoles 740/750 or a server 710, the simulator 120 can operate in learning and/or simulation mode as long as the simulator 120 is communicatively coupled with a copy of the interactive computer program 110. In the case where the simulator 120 and database 130 resides on a server 710, ***any participating user can upload its own user profile for access by any other participating user, and further any participating users can download or access any user profile available in the database*** 130.
>
> ***
>
> Turning to FIG. 6, ***the inputs for a user are stored in user profiles 400, which, as described above, may be individual data structures in a***

Tesla Exhibit 1003.269

> ***database*** 130. The general structure of a user profile 400 for this particular video game is shown in FIG. 6. A user profile includes Sub-profiles, e.g., 410, each containing fields or additional Sub profiles that store the user's inputs in an organized and logical manner.

Ex. 1005, 19:49-62, 9:36-43.

297. A POSITA would have understood and it would have been obvious that when a second user operates the same avatars as the first user, such as avatars representing players on the Los Angeles Lakers, the system would record and store multiple correlations (*i.e.*, at least first, second, and third correlations comprising first, second, and third object representations correlated with first, second, and third instruction sets) for operating the same avatar (*i.e.*, the same player on the Los Angeles Lakers), which any other user could download for use in simulation mode. Accordingly, a POSITA would have recognized that two different users playing with the Los Angeles Lakers during learning mode would result in at least a portion of a third correlation being learned in another learning process that includes operating the first avatar by another user.

d. [12.4] and wherein the user is: a human user, or a non-human user, and wherein the another user is: a human user, or a non-human user.

Limitation [12.4]	Additional Features to Claim 11
	Another user is a human users

Tesla Exhibit 1003.270

298. As I explained above with respect to limitation [3.4] of the '585 Patent, Thomas discloses multiple human or non-human users, including a user and another user playing the video game or computer game, resulting in the system storing multiple user profiles. Ex. 1005, 4:67-5:3 ("the simulator 120 will record the user's decision/input, ***such as a pressed button, combination of buttons, or joystick move of a controller***, in response to the particular situation and/or event"), 8:8-9 ("For a pump fake event, during the learning mode 550, the simulator 120 records how often ***a user pump fakes***.").

13. Claim 13

a. [13.1] The system of claim 1, wherein the first correlation is included in a knowledgebase,

299. As I explained above in respect to limitation [4.1], Thomas discloses a knowledgebase including a first correlation. *See* Section X.B.4.a above.

b. [13.2] and wherein the knowledgebase further includes a third correlation including a third one or more object representations correlated with a third one or more instruction sets for operating a third avatar of the application, and

Limitation [13.2]	Additional Features to Claim 11
	3rd correlation = **3rd OR** + **3rd instructions** for operating **3rd avatar**

300. As I explained above with respect to claim 4, Thomas describes that in learning mode the system can detect and record multiple correlations (*i.e.*, first, second, and third correlations) "related to one or more events 685 and/or situations

Tesla Exhibit 1003.271

655 that occur within the computer program 110 during operation," such as for a pass out of shot, pump fake, rebound, etc. *Id.,* 4:14-16; *see* Section X.B.4.b above.

301. Thomas additionally describes that a user can operate multiple avatars during learning mode (*i.e.*, first, second, and third avatars). Thomas describes and a POSITA would have understood that the simulator records correlations including instruction sets for operating multiple avatars, such as avatars representing different players on the same team in a sports video game, including a first, second and third avatars (such as three different avatars on the same basketball team):

> Another type of parameter, in the case of a computer ***game involving one or more avatars***, is the archetype…. For example, an archetype can be defined by relative size, strength, and speed…. **[*A]rchetypes can include true point guard, slasher, shooter, pure shooter, low post scorer, midrange scorer, and role player*.**
>
> ***
>
> For each event, during learning mode 550, the simulator 120 records the archetype of the avatar that initiated a monitored event.

Ex. 1005, 4:45-51, 8:38-40.

302. Thomas therefore discloses a third correlation including a third one or more object representations correlated with a third one or more instruction sets (*i.e.*, the spatial position of defensive players and corresponding instructions for a pump fake, pass out of shot, defensive rebound) for operating a third avatar of the

Tesla Exhibit 1003.272

application (*i.e.*, any of three of more players on a basketball team that the user controlled during learning mode).

c. [13.3] wherein the third one or more object representations represent one or more objects of the application,

303. As I explained above in respect to limitation [4.3], Thomas discloses a third object representation(s) that represents one or more objects of the application, such as the defensive players and their spatial position. *See* Section X.B.4.c above.

d. [13.4] and wherein at least a portion of the first correlation is learned in a learning process that includes operating the first avatar of the application at least partially by a user, and

304. As I explained above in respect to limitation [11.4], Thomas discloses a first correlation at least partially learned in a learning process where a user operates a first avatar of the application. *See* Section X.B.11.d above.

e. [13.5] wherein at least a portion of the third correlation is learned in another learning process that includes operating the third avatar of the application at least partially by: the user, or another user, and wherein the user is: a human user, or a non-human user, and wherein the another user is: a human user, or a non-human user.

Limitation [13.5]	Additional Features to Claim 11
	3rd correlation learned in **another learning process** through operation of the **3rd avatar** by **another user**

305. As I explained for limitation 11.4, Thomas discloses that different correlations (i.e., first, second, and third correlations) are learned in different

Tesla Exhibit 1003.273

learning processes in which the user operates different avatars on the same team during the same game or different learning processes that occur when the user plays multiple games. *See* Section X.B.11.d above.

306. And as explained above for limitation 13.2 above, Thomas discloses that the correlations (including a third correlation) are learned in a learning process that includes operating the avatars (including a third avatar) at least partially by a human user or non-human user. *See also* Section X.B.11.d-e above.

14. Claim 14

a. [14.1] The system of claim 1, wherein the first correlation is included in a knowledgebase, and

307. As I explained above in respect to limitation [4.1], Thomas discloses a knowledgebase storing a first correlation. *See* Section X.B.4.a above.

b. [14.2] wherein the knowledgebase further includes a third correlation including a third one or more object representations correlated with a third one or more instruction sets for operating an avatar of another application, and

Limitation [14.2]	Additional Features to Claim 4
	3rd correlation = **3rd OR** + **3rd instructions** for **3rd avatar** of **another application**

308. As I explained above in respect to limitations [4.2] and [13.2], Thomas discloses a user profile (i.e., a knowledgebase) containing multiple correlations, including a third correlation between a third object representation(s) and a third

Tesla Exhibit 1003.274

instruction set(s) for operating an avatar in a video game. *See* Section X.B.4.b and X.B.13.b above.

309. Thomas also describes that players can utilize the disclosed system while playing on different "client console(s)," including "workstation 750" and "video game console 760, such as an Xbox or Playstation." Ex. 1005, 19:35-43. Moreover, the '134 Patent describes that games on different devices, such as "a computer game, a virtual world, a 2D or 3D graphics application [or] a web browser," comprise different applications. '134 Patent, 170:62-172:3.

310. Thomas describes that the simulator can store profiles for any participating user that any other user can download and access on the same or a different device. Ex. 1005, 19:57-62 ("any participating user can upload its own user profile for access by any other participating user, and further any participating users can download or access any user profile available in the database 130."). In simulation mode, when a user (*e.g.* User A) selects another user's profile, the simulator operates the avatars on User A's console according to the correlations stored in the profile. *Id.*, 3:63-4:2 ("the simulator 120 will then retrieve input data from the first user, i.e., the first user's profile, from the database 130 (action block 220) and simulate the first user's behavior based on the retrieved data (action block 230)."). When multiple users, such as User A and User B, download the same profile, the simulator uses the same stored correlations (including a third correlation) to

Tesla Exhibit 1003.275

operate the same avatars in both User A and User B's versions of the game (an application and another application).

311. A POSITA would have understood that different users could play the same game from different consoles and users could download profiles originally recorded on a different device- such as a Playstation and XBox. Cross-console play for multiplayer games has been known in the art since at least 2003, when FFIX introduced cross-play between users playing FFIX designed for Windows on their computers and users playing FFIX designed for Playstation on their consoles. Ex. 1016. It would have been obvious to a POSITA that two users playing on different devices could download the same profile, where the profile included correlations (including a third correlation) for operating an avatar on a first application and another application. For example, two different users could download the same profile for use on their versions of the game-either both on a Playstation or one a PlayStation and one on a PC. In either situation, the downloaded correlations (including a third correlation) would be used to operate an avatar on an application and another application.

c. [14.3] wherein the third one or more object representations represent one or more objects of the another application.

Limitation [14.3]	Additional Features to Claim 1
	3rd OR represent objects **another application**

Tesla Exhibit 1003.276

312. As I explained above, multiple users can download a profile storing correlations recorded on another user's game, on the same or a different console, including a third correlation. When multiple users download the same profile from the simulator, the simulator uses the stored correlations to invoke the proper instruction set for events and situations in each user's game. When a second user (ex. User B) uses a profile, the third object representation(s) represent object(s) in User B's game (another application), such as the defensive players in User B's game. For example, the first application could be a PC game, and the another application could be a game played by another user on a PC or a Playstation.

15. Claim 15: The system of claim 1, wherein the first one or more instruction sets for operating the first avatar of the application include one or more information about one or more states of: the first avatar of the application or a portion of the first avatar of the application.

Claim 15	Claimed Features
	1st instructions include state of the **1st avatar**

313. Thomas discloses that the instructions recorded during learning mode (*i.e.*, the first one or more instruction sets) can include information about the state of the avatar, such as the avatar's archetype, whether the avatar possesses the ball (*e.g.*, for passing, shooting, or a pump fake), its ability to kick a football, and the amount of time left in the game the avatar is playing:

> For a pump fake event, during the learning mode 550, the simulator 120 records how often a user pump fakes. For current circumstances, the

simulator 120 records how covered the user was and user's location. During playback mode 600, the simulator 120 will monitor for a shoot event and ***analyze the location of the simulated user and the defensive coverage. If criteria is met, then simulator 120 will invoke a pump fake.***

***

For an offensive rebound event, ***the simulator 120, during learning mode 550, will record which avatar grabbed the rebound and whether the user immediately passed the ball***, which indicates a "reset" type rebound, or whether the user shoots the ball, which indicates a "shoot" type rebound. Again, current circumstances are recorded for the event. ***During playback 600, the simulator 120 will analyze the current circumstances during the occurrence of an offensive rebound event and invoke the proper response accordingly, e.g., whether to invoke a "shoot" rebound or a "reset" rebound***.

***

[A] kick profile 420 stores information ***as to how well a user kicks the ball during play***.…When the simulator 120 is in simulation mode, based on the user's profile 400, ***and the simulated user is in a kick situation***, the simulator 120 may determine the power for kick based on the corresponding kick profile 420, e.g., whether the power should be above the average or below the average and/or whether the power should be a random value between the maximum and minimum powers recorded.

***

Tesla Exhibit 1003.278

Another sub-profile that can be included is the special moves profile 460. ***A user can invoke special moves if the user's avatars 320 are in possession of the ball.*** In such a situation, the simulator 120 may examine where the closest two defending avatars 320 are in relation to the ball carrying avatar 320 ("the ball carrier") on the football field 310.

***

Another type of parameter, in the case of a computer game involving one or more avatars, is the ***archetype of the avatar***. For example, an archetype can be defined by ***relative size, strength, and speed***. In the case of a basketball game, wherein avatars represent players of the game, archetypes can include true point guard, slasher, shooter, pure shooter, low post scorer, midrange scorer, and role player.

***

One common parameter that defines a situation may be time 660, e.g., time left in a particular session or game, and/or time played in a particular session or game.

***

For a defensive steal event, the simulator 120, during learning mode 550, records how often a user attempts to steal and the average time between attempts. As current circumstances, the simulator 120 records the ***user's distance to the ball, number of fouls to give in a quarter, and the skill level of the avatar with the ball***.

Ex. 1005, 8:8-15, 7:45-54, 12:42-57, 12:63-13:1, 4:45-51, 4:29-31, 8:16-21.

16. Claim 16: The method system of claim 1, wherein: an element of the first correlation is deleted after the first

Tesla Exhibit 1003.279

correlation is generated, an element of the first correlation is modified after the first correlation is generated, an element of the first correlation is manipulated after the first correlation is generated, or an element is inserted into the first correlation after the first correlation is generated.

Claim 16	Claimed Features
	Element of **1st correlation** is deleted/modified/manipulated/inserted

314. The '134 Patent specification describes that correlations can include "Extra Info," including "any information useful in comparisons or decision making performed in autonomous Avatar 605 operation". '134 Patent, 95:50-53; 98:25-30 ("***Object Representations 525*** representing Objects 615 in Avatar's 605 surrounding some of which may not be ***correlated with any Instruction Sets 526 and/or Extra Info 527***.").

315. Thomas discloses the elements of claim 16 because it describes that an element of the first correlation is inserted and/or modified after the first correlation is generated. In particular, Thomas describes that if the simulator encounters a situation or event for which there is existing data in the playbook profile (i.e. after the first correlation is generated), the simulator evaluates whether the correlation is "up to date." Ex. 1005, 10:27-29. Thomas further describes that as a user continues to interact with the system, data in the user's profile, including elements of a first correlation, "evolve as the user's interaction evolves. In other words, data recorded thirty games ago may have a lesser impact on the total data available." *Id.,* 10:27-

Tesla Exhibit 1003.280

32. As described above, for example, Thomas describes modifying a correlation comprising an instruction set for kicking a football, such as the range of power.

316. Thomas further describes that if a user invokes more than one response to the same situation/event, the system stores multiple correlations linking the same object representation(s) with different instruction set(s), and that each correlation is assigned a numerical weight value that corresponds to the frequency with which a user invoked a particular response to the same situation during learning mode:

> If there is more than one value (decision block 630), then the actual second user may have invoked more than one type of response to the particular situation or event while being recorded. In such a case, the simulator 120 can randomly select from one of the values, or ***the simulator 120 can select its response based on statistics or mathematical formula, e.g., based on the number of times the actual user invoked one response over another (action block 635). Such statistics may indicate tendencies demonstrated by the simulated user.***
>
> ***
>
> Turning to FIG. 11a, at step 570, the simulator 120 will check if one or more user inputs have already been recorded for the current situation and/or event (decision block 575). If so, then ***the simulator 120 will associate one or more values with each different type of user input based on the statistical occurrence of each input compared to other inputs*** (action block 577). For example, in the case of a football game, a player's avatar is running down the sideline and encounters an opposing player's avatar (an event). In one hundred

Tesla Exhibit 1003.281

(100) occurrences of the event, if the user spun left twenty (20) times, right thirty (30) times, and juked fifty (50) times, the simulator 120 ***will assign values of twenty (20), thirty (30), and fifty (50) to the respective responses***….It should be noted that the weighted values can be calculated based on any number of responses and any number of types of responses.

Id., 5:63-6:4, 6:10-27. A POSITA would have recognized that the numerical values described in Thomas correspond to "extra info" for a correlation as described in the '134 Patent.

317. As such, a POSITA would have understood that the numerical weight value associated with how frequently a user invoked a particular response to a situation/event (i.e., the numerical weight for each correlation) would be modified and/or inserted into the correlation each time the user invoked a command in response to a particular situation/event after the correlation was first generated.

17. Claim 17

a. [17.1] The system of claim 1, wherein the machine readable code, when executed by the one or more processors, causes the one or more processors to further perform at least: modifying: the first one or more instruction sets for operating the first avatar of the application, or a copy of the first one or more instruction sets for operating the first avatar of the application, and

Limitation	Claimed Features
[17.1]	Modification of **1st instructions**

Tesla Exhibit 1003.282

318. As I explained above, Thomas describes that the simulator evaluates whether a user's profile is up to date after each game. *Id.*, 10:29-32 ("Preferably, data should evolve as the user's interaction evolves. In other words, data recorded thirty games ago may have a lesser impact on the total data available."). Moreover, Thomas describes that a user's profile can include a kick profile, which stores information on how a particular user kicks a football, such as the minimum and maximum power/distance:

> Turning to FIG. 6, a kick profile 420 stores information as to how well a user kicks the ball during play. ***This includes how many times a user has kicked the ball, the minimum and maximum distances that a user has kicked the ball, and the total power used***. There is a kick profile for when a user kicks off, punts, and kicks field goals.

Id., 12:42-47.

319. Thomas further describes that as the user continues to interact with the system during learning mode, the simulator updates the kick profile. Each time the user kicks the ball, for example, the simulator modifies the instruction set by increasing the kick count and/or modifying the minimum and maximum distance/power values in the kick profile:

> ***When recording data for a kick profile 420, the simulator 120 may check to see if the user's power has exceeded the minimum or maximum, increment the kick count, and add the power to the total***

Tesla Exhibit 1003.283

power accumulated. the simulator 120 is in simulation mode, based on the user's profile 400, and the simulated user is in a kick situation, the simulator 120 may determine the power for kick based on the corresponding kick profile 420[.]

***

Turning to FIG. 6, ***a kick profile 420 stores information as to how well a user kicks the ball during play. This includes how many times a user has kicked the ball, the minimum and maximum distances that a user has kicked the ball, and the total power used***… When the simulator 120 is in simulation mode, based on the user's profile 400, and the simulated user is in a kick situation, the simulator 120 may determine the power for kick based on the corresponding kick profile 420, e.g., whether the power should be above the average or below the average and/or whether the power should be a random value between the maximum and minimum powers recorded.

Id., 12:47-54, 12:42-62.

320. Thus, Thomas discloses modifying a first instruction set(s) associated with the avatar performing a kick operation.

b. [17.2] wherein the executing the first one or more instruction sets for operating the first avatar of the application includes executing: the modified the first one or more instruction sets for operating the first avatar of the application, or the modified the copy of the first one or more instruction sets for operating the first avatar of the application, and

	Claimed Features

Limitation [17.2]	Executing modified **1st instructions**

321. Thomas discloses that executing the first one or more instruction sets for operating the first avatar includes executing the modified instruction set. Thomas describes, for example, that in simulation mode, during a kick situation, the simulator operates the avatar according to the kick profile. *Id.*, 12:51-57 ("[T]he simulator 120 is in simulation mode, based on the user's profile 400, and the simulated user is in a kick situation, the simulator 120 may determine the power for kick based on the corresponding kick profile 420[.]"), 12:46-47 ("There is a kick profile for when a user kicks off, punts, and kicks field goals."). As I explained above, the kick profile is modified as the user interacts with the system in learning mode, including by modifying the power/distance range. Accordingly, a POSITA would recognize that executing the instruction set in Thomas includes executing the modified instruction set associated with kicking the football for kickoffs, punts or field goals.

c. [17.3] wherein the performing, by the first avatar of the application or by the second avatar of the application, the one or more operations defined by the first one or more instruction sets for operating the first avatar of the application includes performing, by the first avatar of the application or by the second avatar of the application, one or more operations defined by: the modified the first one or more instruction sets for operating the first avatar of the application, or the

Tesla Exhibit 1003.285

modified the copy of the first one or more instruction sets for operating the first avatar of the application.

Limitation [17.3]	Claimed Features
	1st avatar performs modified 1st instructions

322. As I explained above, during a kick situation in simulation mode, an avatar representing the kicker kicks the ball according to the kick profile and the kick profile is modified as the user interacts with the system in learning mode. Thomas therefore discloses that the kicker (a first avatar) performs the operation defined by the modified instruction set by kicking the ball as directed by the modified kick profile (*e.g.*, a modified kick profile with modified minimum and/or maximum distances/power).

18. Claim 18

a. [18.1] The system of claim 1, wherein at least a portion of the first correlation is learned in a learning process that includes: operating the first avatar of the application at least partially by: a human user, or a non-human user:

323. As I explained above in regards to claims 11.4 and 11.5, Thomas discloses a system where a portion of the first correlation is learned in a learning process where a human user or non-human user at least partially operates a first avatar of the application. *See* Section X.B.11.d-e above.

Tesla Exhibit 1003.286

b. [18.2] generating or receiving the first one or more object representations;

Limitation [18.2]	Additional Features to Claim 11
	Generating/receiving **1st OR**

324. As I explained above in regards to claim 1, Thomas discloses this limitation because it describes that during learning mode, the system generates and stores (*i.e.*, receives) information concerning situations/events, including object representations of the defensive players and their spatial position in a video game. *See* Section X.B.1.c above.

c. [18.3] and obtaining or receiving the first one or more instruction sets for operating the first avatar of the application.

Limitation [18.3]	Additional Features to Claim 11
	Obtaining/receiving **1st instructions**

325. As I explained above in relation to claim 1, Thomas also discloses that during learning mode the system stores (*i.e.*, obtains and receives) the user's commands (*i.e.*, instruction sets) associated with particular situations and/or events during the game. *See* Section X.B.1.c above.

19. Claim 19

a. [19.1] A system comprising: a memory that stores at least a first correlation including a first one or more object representations correlated with a first one or more instruction sets for operating a first avatar of an application, wherein the first one or more object

representations represent one or more objects of the application;

326. As I explained above in relation to limitation [1.3], Thomas discloses a memory that stores a first correlation including a first object representation(s) representing one or more objects of the application and a first instruction set(s) for operating an avatar in the application. *See* Section X.B.1.c above; *see also* Section X.B.8.c above.

b. [19.2] means for generating or receiving a second one or more object representations, wherein the second one or more object representations represent one or more objects of the application;

Limitation [19.2]	Additional Features to Claim 1
	Means for generating/receiving **2nd OR**

327. As I discussed above in relation to claim construction, the corresponding structure for this limitation is one or more processors or processor circuits. As I explained above in relation to claim 1, Thomas discloses that during simulation mode, a processor generates and/or receives a second object representation(s) representing one or more objects of the application (ex. the spatial locations of players). *See* Section X.B.1.d above. For example, Claims 1 and 2 of Thomas describe that the processor generates and receives the second one or more object representations to monitor the game for one or more events. Ex. 1005, claims

Tesla Exhibit 1003.288

1-2 (requiring that the processor execute a first process and second process, which include "monitoring the computer game for one or more events…").

c. [19.3] means for determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second one or more object representations and the first one or more object representations; and

Limitation [19.3]	Additional Features to Claim 1
	Means for determining **1st instructions** through partial match between **1st OR** and **2nd OR**

328. As I discussed above in relation to claim construction, the corresponding structure for this limitation is one or more processors or processor circuits. As I explained above in relation to claim 1, Thomas discloses that during simulation mode, a processor determines the first instruction set(s) for operating an avatar based on partial match between the first object representation(s) recorded during learning mode and the second object representation(s) generated during simulation mode. *See* Section X.B.1.e above. Claims 1 and 2 of Thomas also describe that the processor performs the determining function to invoke a response during simulation mode. Ex. 1005, claims 1-2 (requiring that the processor execute a first process and second process, which include "invoking a response …").

d. [19.4] means for executing, at least in response to the determining, the first one or more instruction sets for operating the first avatar of the application, wherein the first avatar of the application or a second avatar of the

Tesla Exhibit 1003.289

application performs one or more operations defined by the first one or more instruction sets for operating the first avatar of the application.

Limitation [19.4]	Additional Features to Claim 1
	Means for executing **1st instructions** for operating **1st avatar**

329. As I discussed above in relation to claim construction, the corresponding structure for this limitation is one or more processors or processor circuits. As I explained above in relation to claim 1, Thomas discloses that during simulation mode, a processor executes the first instruction set(s) causing a first avatar of the application to perform the defined operation(s). *See* Section X.B.1.f above. Claims 1 and 2 of Thomas also describe that the processor performs the executing function by "invoking a response" during simulation mode. Ex. 1005, claims 1-2 (requiring that the processor execute a first process and second process, which include "invoking a response …").

20. Claim 20: The system of claim 19, wherein the means for generating or receiving the second one or more object representations includes one or more processors, and wherein the means for determining the first one or more instruction sets for operating the first avatar of the application based on the at least partial match between the second one or more object representations and the first one or more object representations includes one or more processors, and wherein the means for executing, at least in response to the determining, the first one or more instruction sets for operating the first avatar of the application includes one or more processors.

Claim 20	Additional Features to Claim 19

Tesla Exhibit 1003.290

	Means for generating and receiving/determining/executing include processors

330. Claim 20 requires that the corresponding structure for each of the functions claimed in claim 19 ("means for generating or receiving," means for determining," and "means for executing") includes one or more processors. As I explained above in relation to claim 19, Thomas discloses that the generating/receiving, determining and executing functions are performed by processors.

C. Ground 2: Anticipation of Claims 1, 3, and 17 Based on Buehler

331. As set forth in detail below, claims 1, 3 and 17 of the '134 Patent are also anticipated by Buehler. The discussion below specifies where each element of the aforementioned claims is found in Buehler, and includes a detailed explanation of the significance of the quotations and citations.

1. Claim 1

a. [1.1] A system comprising: one or more processors; and:

Limitation	Claimed Features
[1.1]	System including processors

332. As I explained above in regards to limitation [1.1] of the '585 Patent, Buehler discloses a system including one or more processors. Ex. 1006, 4:51-53

Tesla Exhibit 1003.291

("The image-processing module and the control system comprise sets of instructions stored in memory and executable by ***a processor***."); 8:10-15 (describing a robot system).

b. [1.2] and one or more non-transitory machine readable media storing machine readable code that, when executed by the one or more processors, causes the one or more processors to perform at least:

Limitation [1.2]	Claimed Features
	+ machine readable media + storing code + executed by processors + causes processors to perform

333. The '134 Patent specification describes that "non-transitory machine readable medium" comprises "***all machine-readable media*** except for a transitory, propagating signal," '134 Patent at 64:26-28, and includes "any memory" or "storage medium." *Id*., 64:12-25.

334. Buehler discloses a hard disk, optical storage unit or memory (i.e. non-transitory machine readable media) storing sets of instructions (machine readable code) executable by processor. Ex. 1006, 10:5-14 ("[T]he robot controller and high-level computational functionality are implemented on a general-purpose computer 210…. The general-purpose computer 210 includes a central processing unit (CPU)

Tesla Exhibit 1003.292

222…***system memory 224, and non-volatile mass storage devices (such as one or more hard disks and/or optical storage units***) 226."), 4:51-56 ("The image-processing module and the control system comprise ***sets of instructions stored in memory*** and executable by a processor."), 12:21-25 ("describing that the user-robot interaction is "controlled and processed by associated ***program code***").

c. [1.3] accessing a first correlation including a first one or more object representations correlated with a first one or more instruction sets for operating a first avatar of an application, wherein the first one or more object representations represent one or more objects of the application;

Limitation	Claimed Features
[1.3]	Accessing a **1st correlation** + **1st correlation = 1st OR + 1st instructions** + OR = object representations representing objects of the application

335. Buehler discloses accessing a memory storing object representations (*e.g.*, template images or stored representations) correlated with object-specific routines (*i.e.*, instruction sets) for operating the robot and end-effector. As I describe below for claim 17, Buehler also discloses that the instruction sets correlated to the representations of object can be learned during training mode where a user partially operates the robots. *See* Section X.C.3 below.

> The method includes causing a robot to execute the following steps: selecting an object in a camera field of view, using a visual model to computationally ***identify the object based on a stored representation***

Tesla Exhibit 1003.293

thereof associated with the model, determining whether ***an object specific manipulation routine is stored in association with the stored representation***, and, if so, executing the object-specific manipulation routine.

***

The system memory 224 contains partitions and/or buffers to store, in digital (pixel) form, images obtained by the cameras, as well as portions of one or more ***databases for representing task and object representations as described below***.

***

In some embodiments, the selection is based on a visual model and one or more associated object representations stored in memory. ***The object representation may include an image template, coordinates and texture descriptors associated with object keypoints, and/or parameters associated with an object shape.***

***

In some embodiments, the ***robot stores a plurality of action templates***, e.g., in the form of prototypes for manipulations, and the robot and/or user may select one of the templates to indicate a present training objective. For example, the user may select an action template from a list, or ***the robot may select the action template automatically based on sensor input and context. If the robot detects, for instance, a conveyor belt, it may select an action for picking up objects from the conveyor belt.***

Id., 7:6-14, 10:22-26, 5:7-12, 22:35-44.

Tesla Exhibit 1003.294

336. The "stored representations" or template images described in Buehler comprise object representations representing objects of the application because they represent objects that the robot may encounter, which the system matches with newly captured images to determine an instruction set for operating an avatar of the robot and its end effector in a computer simulation or virtual representation, as described below:

> In this method, the robot selects an object in the field of view and displays the selection (including the camera view of the selected object) on a robot-controlled visual display, ***overlays a representation of an end-effector onto the camera view, and displays performance of an action on the object by the end-effector***…. In some embodiments, the robot ***selects the action from a plurality of stored action templates***.
>
> ***
>
> [T]he robot may ***overlay virtual grippers on the image*** (or video stream) of the object to indicate where the grasp point would be, given the current arm location and orientation (step 606). Thus,… the robot can ***give the user a sense of the likelihood of a successful*** pick-up.

Id., 6:31-48, 21:46-54.

d. [1.4] generating or receiving a second one or more object representations, wherein the second one or more object representations represent one or more objects of the application;

Limitation [1.4]	Claimed Features
	Generating or receiving **2nd OR**

Tesla Exhibit 1003.295

337. Buehler discloses that when the robot encounters a new object, the robot generates a second object representation representing the object by using its camera to acquire real-time images of the object and overlaying the images with computer graphics, including graphics representing the robot's end-effector. The robot compares these images to stored image data to determine the appropriate object-specific routine. A POSITA would understand these images include a second object representation representing an object of the application because the images depict an object in the virtual simulation that the virtual robotic end effector will act upon:

> For example, ***the robot may use a camera to acquire images of its environment (constituting real-world data), and overlay computer graphics onto the image*** (such as, e.g., an object outline in the camera image, indicating selection of the outlined object by the robot, and/or ***a representation of the robot's end-effector***, indicating the position of the end-effector relative to the object).
>
> ***
>
> The robot may overlay ***virtual grippers on the image (or video stream) of the object to indicate where the grasp point would be, given the current arm location and orientation (step 606).***

Id., 2:6-12, 21:47-50.

e. [1.5] determining the first one or more instruction sets for operating the first avatar of the application based on at least partial match between the second one or more

Tesla Exhibit 1003.296

object representations and the first one or more object representations; and

Limitation [1.5]	Claimed Features
	Determining **1st instructions** from partial match between **1st OR** and **2nd OR**

338. Buehler discloses comparing captured images of an object (second object representations) to representations or templates of objects stored in the robot's memory (first object representations). If the newly captured images at least partially match the stored templates or representations, the robot will execute the associated object-specific routine (determining the first instruction set):

> The method includes causing a robot to execute the following steps: selecting an object in a camera field of view, ***using a visual model to computationally identify the object based on a stored representation thereof associated with the model, determining whether an object specific manipulation routine is stored in association with the stored representation***, and, if so, ***executing the object-specific manipulation routine***.
>
> ***
>
> In some embodiments, ***the visual representation of an object informs the robot beyond the purpose of detecting and localizing that type of object in images. For example, knowing the shape of an object may help the robot choose an appropriate grasping strategy***. If the representation describes, for instance, an object with a hollow (e.g., cylindrical) body and a handle (e.g., akin to a mug), the robot may attempt to pick up the object by its handle. More generally,

representations of objects, or classes of objects, may have specific routines for picking up or otherwise manipulating the object associated with them. Container objects, for example, may have associated routines for opening or closing the containers, which vary between classes of containers. Thus, the robot's end-effector would undergo different motions to open a cylindrical container with a lid than it would to open a cardboard box. ***The robot may learn these special routines during training, and associate them with particular visual representations***.

***

From one or more annotated camera views identifying the object of interest, the learning algorithm 504 extracts one or more respective image templates of the object by tightly cropping around the object outline. A similarity threshold for each template can be determined using conventional cross-validation techniques which exercise the detection algorithm on training data. The training data can be the original data given to the learning algorithm, newly acquired training images of the object, or modified versions of training images that simulate noise or other expected image transformations. ***The detection algorithm 508 compares real-time camera images against the template and determines, based on a similarity metric, whether the real-time image contains the object of interest.***

Id., 7:6-15, 22:13-30, 18:40-54.

339. Buehler further describes that if the newly acquired image (i.e., a second object representation) corresponds to stored image data (i.e., a first object representation), the system uses a visual representation of the robot (first avatar) to

Tesla Exhibit 1003.298

execute the associated routine (*i.e.*, instruction set) and simulate the robot's performance of a task, such as grasping the object:

> The robot may repeat the process of selecting (step 602) and testing (step 604) visual models until the user confirms the robot's ability to successfully detect the object….
>
> Following the user's approval of the selected visual model, the robot may overlay ***virtual grippers on the image (or video stream) of the object to indicate where the grasp point would be, given the current arm location and orientation (step 606)***. Thus, without the need to actually lower the gripper and bring it in contact with the object…***the robot can give the user a sense of the likelihood of a successful pick-up***.

Id., 21:43-54.

f. [1.6] at least in response to the determining, causing the first avatar of the application or a second avatar of the application to perform one or more operations defined by the first one or more instruction sets for operating the first avatar of the application at least by executing the first one or more instruction sets for operating the first avatar of the application.

Limitation [1.6]	Claimed Features
	1st instructions executed by **1st avatar** or **2nd avatar**

340. As I explained above, Buehler discloses that the robot determines the appropriate object-specific routine based on at least partial match between the real-time images captured by the robot's camera and the stored templates or

Tesla Exhibit 1003.299

representations. Buehler further discloses that when the robot detects a match, the system overlays a virtual representation of its end-effector on the captured images and executes the instruction set to cause the virtual robot and its end effector (i.e., the first avatar) to simulate performing a task such as grasping the object using the correlated object-specific routine (executing an operation defined by the first instruction set):

> Once the robot has received the user's confirmation, ***it may analyze the pixels corresponding to the object and selects a suitable visual representation and algorithm to reliably detect and localize the object enclosed by the agreed upon outline*** (step 602, which is the same as or similar to step 402 of FIG. 4)….
>
> Following the user's approval of the selected visual model, ***the robot may overlay virtual grippers on the image (or video stream) of the object to indicate where the grasp point would be, given the current arm location and orientation***….
>
> ***
>
> In this method, the robot selects an object in the field of view…, ***overlays a representation of an end-effector onto the camera view, and displays performance of an action on the object by the end-effector***….
>
> ***
>
> ***To facilitate the robot's autonomous performance of tasks in execution mode, the task-execution module 268 may utilize***

Tesla Exhibit 1003.300

representations of tasks, and of objects to which the tasks pertain, that have been specified during training mode.

Id., 21:33-54, 6:31-37, 10:52-55.

2. Claim 3

a. [3.1] The system of claim 1, wherein the determining the first one or more instruction sets for operating the first avatar of the application based on the at least partial match between the second one or more object representations and the first one or more object representations includes:

341. As I discussed above for limitation [1.5], the first instruction sets for operating the virtual representation of the robot and its end-effector are determined based on partial match between the real-time images of an object (second one or more object representations) and the previously stored models or representations (first one or more object representations). *See* Section X.C.1.e above.

b. [3.2] determining that a number of at least partially matching portions of the second one or more object representations and portions of the first one or more object representations exceeds a threshold number or determining that a percentage of at least partially matching portions of the second one or more object representations and portions of the first one or more object representations exceeds a threshold percentage.

Limitation	Additional Features to Claim 1
[3.2]	Partial match exceeds a threshold

342. As I explained above, the ’134 Patent specification describes that the threshold number or percentage can comprise any percentage or number, including

Tesla Exhibit 1003.301

one. *See* '134 Patent, 115:2-7 ("[S]ubstantial similarity can be achieved when at least a threshold number (i.e. 1, 2, 3, 6, 11, etc.) or percentage (i.e. 55%, 61%, 78%, 82%, 99%, etc.) of Object Properties 630 or portions thereof…match or substantially match.").

343. Buehler discloses determining that a number of at least partially matching portions exceeds a threshold number or percentage. In particular, Buehler describes that when its system matches newly captured images to stored template images (*i.e.*, comparing first and second object representations), the system uses a metric or score (*i.e.*, a threshold number) to assess whether there is a match. If the metric or score exceeds a "metric" or "score" (*i.e.*, a numerical threshold), the system determines that the object representations at least partially match:

> The shape of an ellipse can be fully described with three parameters, e.g., the orientation and the lengths of major and minor axes. The parameter values may be determined using a least-squares method, or any other suitable regression approach. ***The learning algorithm may also provide an error metric or confidence metric that indicate how closely the object can be fitted with an ellipse. In general, if the error exceeds a specified threshold, the representation is deemed invalid.***
>
> ***
>
> In a third class of visual models, an image of the object itself, or multiple images of the object (e.g., taken from different perspectives), serve as the representation, ***along with an image-similarity***

Tesla Exhibit 1003.302

> ***threshold***…. ***The detection algorithm 508 compares real-time camera images against the template and determines, based on a similarity metric, whether the real-time image contains the object of interest***. For example, the detection algorithm may run a normalized cross-correlation of the stored image template with a presently acquired image, resulting in a score for reach pixel. ***Scores above a set threshold (e.g., the threshold determined by the learning algorithm based on anticipated noise levels etc.), may then be clustered to determine the potential location of the object in the image***.
>
> ***
>
> ***The representation of the object may include***, for example, parameters associated with a shape (e.g., an ellipse), coordinates and texture descriptors associated with keypoints, ***or an image template and associated similarity threshold***.

Ex. 1006, 17:58-18:7, 18:37-60, 5:50-54.

344. One method Buehler describes for matching representations objects encountered in real time with previously stored visual models is ellipse fitting. *Id.*, 17:60-18:7 ("In the first type of visual model, the object is represented with a two-dimensional shape, such as an ***ellipse***…. ***The shape of an ellipse can be fully described with three parameters, e.g., the orientation and the lengths of major and minor axes.*** The parameter values may be determined using a least-squares method, or any other suitable regression approach…. In general, ***if the error exceeds a specified threshold***, the representation is deemed invalid."). A POSITA would have understood that the method of ellipse fitting to correspond to a known technique for

Tesla Exhibit 1003.303

image recognition based on a partial match, including a threshold number. *See, e.g.* Ex. 1014 at 1014.002 ("In this paper we consider two aspects of the wider task of matching image to model primitives. Firstly, we extract ellipses from the image data and use the (singly or in groups) to estimate viewpoint and infer structures in the scene. Secondly, we match these structures to model structures."), 1014.004("We select a value for the threshold…in order to provide a suitable trade-off between accepting a high proportion of good features, and rejecting poor ones. In noisier images, the threshold can be reduced to make the test more selective.").

3. Claim 17

a. [17.1] The system of claim 1, wherein the machine readable code, when executed by the one or more processors, causes the one or more processors to further perform at least: modifying: the first one or more instruction sets for operating the first avatar of the application, or a copy of the first one or more instruction sets for operating the first avatar of the application, and

Limitation	Claimed Features
[17.1]	Modification of **1st instructions**

345. As I explained above for claim 1, Buehler discloses machine readable code executed by processors. As I explained above for limitation [12.1] of the '585 Patent, Buehler discloses modifying the instruction sets for operating a virtual representation of the robot and its end-effector (a first avatar of the application)

Tesla Exhibit 1003.304

during training mode, such that the user and/or robotic system can modify the instruction set for operating the virtual end effector by adjusting the routine after the system simulates performance of the task:

> If the virtual gripper is not positioned around the proper part of the object (as determined in step 608), ***the user may adjust the arm*** to roughly establish the proper grasp point (step 610)…. ***This process of coarse and fine adjustments of the gripper by the human trainer and the robot, respectively, may be iterated until the user is satisfied with the grasp point and gripper orientation shown in the visual display***.
> ***
>
> The robot may learn these special routines during training, and associate them with particular visual representations. ***Alternatively, the routines may be programmed into the robot at the outset and altered, as necessary, during training as noted above***.

Ex. 1006, 21:54-64, 22:28-32.

b. [17.2] wherein the executing the first one or more instruction sets for operating the first avatar of the application includes executing: the modified the first one or more instruction sets for operating the first avatar of the application, or the modified the copy of the first one or more instruction sets for operating the first avatar of the application, and

Limitation [17.2]	Claimed Features
	Executing modified **1st instructions**

Tesla Exhibit 1003.305

346. As I explained above for limitation [12.2] of the '585 Patent, Buehler discloses that when the robot simulates performing a task after modifying the associated routine, the robot executes the modified instruction set for operating the virtual representation of the robot and its end-effector (first avatar of the application) so that the user can assess whether the task is performed properly. *Id.*, 6:37-43 ("Based on the display, feedback is provided to the robot, indicating whether the action is performed correctly on the display…***The display of the robot's performance of the action on the object by the end-effector and feedback are repeated until the action is performed correctly on the display.***").

c. [17.3] wherein the performing, by the first avatar of the application or by the second avatar of the application, the one or more operations defined by the first one or more instruction sets for operating the first avatar of the application includes performing, by the first avatar of the application or by the second avatar of the application, one or more operations defined by: the modified the first one or more instruction sets for operating the first avatar of the application, or the modified the copy of the first one or more instruction sets for operating the first avatar of the application.

Limitation [17.3]	Claimed Features
	1st avatar performs modified **1st instructions**

347. This limitation can be satisfied through any combination of the listed embodiments. As I explained above for limitation [12.3] of the '585 Patent, Buehler

Tesla Exhibit 1003.306

discloses that the performing by the first avatar of the application the one or more operations defined by the first one or more instruction sets for operating the first avatar of the application includes performing the modified the first one or more instruction sets for operating the first avatar of the application. As explained above for limitations [17.1] and [17.2], Buehler discloses that the system executes and the virtual robot (*i.e.*, the first avatar) performs the modified instruction set after it is updated by the user and/or system.

XI. Declaration

348. I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Dated: June 30, 2023 By: 

Jason Janet

Tesla Exhibit 1003.307