

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SAMSUNG ELECTRONICS CO., LTD., and
SAMSUNG ELECTRONICS, AMERICA, INC.,
Petitioners,

v.

CERENCE OPERATING COMPANY,
Patent Owner.

Case No. IPR2024-01267
U.S. Patent No. 9,026,428

PATENT OWNER'S RESPONSE

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	PROCEDURAL BACKGROUND	4
III.	THE '428 PATENT.....	4
IV.	THE SCOPE AND CONTENT OF THE PRIOR ART	10
A.	<i>Arai</i>	10
B.	<i>Fenwick</i>	13
C.	<i>Baker</i>	15
V.	LEVEL OF ORDINARY SKILL IN THE ART	15
VI.	CLAIM CONSTRUCTION.....	16
VII.	APPLICABLE LEGAL STANDARDS	16
VIII.	THE PETITION FAILS TO ESTABLISH THE OBVIOUSNESS OF ANY CHALLENGED CLAIM	19
A.	Ground 1: Petitioners Fail To Show That Claims 1-3, 7-9, and 12- 13 Would Have Been Obvious Based on <i>Arai</i> In View Of <i>Fenwick</i>	19
1.	<i>Arai</i> And <i>Fenwick</i> Are Directed To Fundamentally Different Approaches To Character And Pattern Recognition Than The '428 Patent	19
a.	The '428 Patent	19
b.	<i>Arai</i>	23
c.	<i>Fenwick</i>	26
2.	Because <i>Arai</i> And <i>Fenwick</i> Are Directed To Fundamentally Different Approaches To Character And Pattern Recognition Than The '428 Patent, They Fail To Disclose Or Render Obvious Multiple Claim Limitations	29
a.	Claim 1	29
i.	Petitioners Identify The Wrong Disclosure In <i>Arai</i> For Limitations 1[d] And 1[e]	30
ii.	Element 1[f]: “identifying at least a first sequence portion of the first candidate sequence wherein the identifying is at least partially	

based on one or more rules or dictionaries of words accessible to the mobile device;”	41
iii. Element 1[g]: “removing the first sequence portion from the memory storage buffer, wherein the removing is performed in the absence of receiving user input editing the first candidate sequence, representing a space character, or selecting a word;”	44
b. Claim 2	58
c. Claim 3	59
d. Claim 7	61
e. Claim 8	61
f. Claim 9	61
g. Claim 12	66
h. Claim 13	66
B. Ground 2: Petitioners Fail To Show That Claims 9 and 12-13 Would Have Been Obvious Based on Arai In View Of Fenwick, And In Further View Of Baker	67
IX. CONCLUSION	69

TABLE OF AUTHORITIES

Cases

<i>Abbott Laboratories v. Sandoz, Inc.</i> , 544 F.3d 1341 (Fed. Cir. 2008).....	43
<i>Arctic Cat Inc. v. Bombardier Recreational Prods. Inc.</i> , 876 F.3d 1350 (Fed. Cir. 2017).....	17
<i>Belden Inc. v. Berk-Tek LLC</i> , 805 F.3d 1064 (Fed. Cir. 2015).....	17
<i>Graham v. John Deere Co. of Kansas City</i> , 383 U.S. 1 (1966)	16, 17
<i>Granite Constr. Co. v. United States</i> , 962 F.2d 998 (Fed. Cir. 1992).....	18
<i>In re Cyclobenzaprine Hydrochloride Extended-Release Capsule Patent Litig.</i> , 676 F.3d 1063 (Fed. Cir. 2012).....	17
<i>In re Kahn</i> , 441 F.3d 977 (Fed. Cir. 2006).....	18
<i>In re Magnum Oil Tools Int’l, Ltd.</i> , 829 F.3d 1364 (Fed. Cir. 2016).....	18, 57, 69
<i>In re Stepan Co.</i> , 868 F.3d 1342 (Fed. Cir. 2017).....	17
<i>Intelligent Bio-Systems, Inc. v. Illumina Cambridge Ltd.</i> , 821 F.3d 1359 (Fed. Cir. 2016).....	18
<i>Virtek Vision International ULC v. Assembly Guidance Systems, Inc.</i> , 97 F.4th 882 (Fed. Cir. 2024).....	18
<i>WBIP, LLC v. Kohler Co.</i> , 829 F.3d 1317 (Fed. Cir. 2016).....	17

Statutes

35 U.S.C. § 103	16
-----------------------	----

PATENT OWNER'S EXHIBIT LIST

Exhibit No.	Description
2001	Federal Court Management Statistics, Eastern District of Texas, 2019-2024
2002	Excerpt of Defendant's Amended Invalidity Contentions, (<i>Cerence Operating Company v. Samsung Electronics Co., Ltd.</i> , Case No. 2:23-cv-00482-JRG-RSP (E.D. Tex.))
2003	Weiss PHV
2004	Wang PHV
2005	Tsuei PHV
2006	Hyde PHV
2007	Declaration of Eric Koskinen in support of Patent Owner's Response ("Ex.2007 (Koskinen Decl.)")
2008	Curriculum Vitae of Eric Koskinen
2009	Declaration of Jonas Andersson in support of Patent Owner's Response ("Ex. 2005 (Andersson Decl.)")
2010	April 18, 2025 Deposition Transcript of Jean Ward ("Ex.2010 (Ward Dep. Tr.)")
2011	Lacerda, E.B., de Souza, T. V., Zanchettin. C., Rabelo, J. C., & Coutinho, L.D. (2017). <i>Handwriting Recognition: Overview, Challenges and Future rends</i> . In Bezerra, B.L., Zanchettin, C., Toselli, A.H., & Pirlo, G. (Eds.), <i>Handwriting: Recognition, Development and Analysis</i> (pp.3-32). Nova Science Publisher.
2012	Keysers, D., Deselaers, T., Rowley, H., Wang, L., & Carbone, V. (2017, June). <i>Multi-Language Online Handwriting Recognition</i> . IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), 1180- 1194.

I. INTRODUCTION

The Petition fails to demonstrate that any challenged claims of the '428 patent are unpatentable for three reasons.

First, Petitioners' primary references, *Arai* and *Fenwick*, are directed to entirely different recognition algorithms than the '428 patent. For example, the '428 patent seeks to solve problems with prior art character recognition systems and methods which did not provide contextual analysis for character and word recognition, and which further required the user to press a button or take some other action to determine when one character or word ends, which decreases the speed of character input by the users. To achieve this goal, the '428 patent uses "linguistic tools with an intermediary buffer," which "allows the system to use a 'history' of characters input by a user to increase the speed and accuracy of handwritten characters input to the system," and to further "automatically decide when to clear this intermediate buffer by analyzing the context of two or more characters or words." '428 patent, 2:59-62. The '428 patent thus allows for a faster and more accurate prediction of what words or phrases a user may be trying to enter.

Conversely, *Arai*, Petitioners' primary prior art reference, discloses a recognition process that is directed to the exact type of prior art systems and methods the '428 patent was trying to improve upon (i.e., a process which

provides **no** contextual analysis, **no** ability for predictive text generation, and **no** ability to automatically commit character sequences to an application and flush corresponding memory without additional user input). *See, e.g. Arai* ¶¶ [139]-[154] (walking through *Arai*’s recognition process).

Likewise, in contrast to the ’428 patent, which is a “linguistically aided autocommit system for handwriting recognition user interfaces” (’428 patent, 2:39-40), *Fenwick* merely discloses a spacing correction algorithm. *Fenwick*, 1:6-10; 1:34, 1:42-45. Specifically, *Fenwick* describes a system and process of storing characters in multiple memory buffers to identify potential word matches, so that it can append a space after any word which may have been inadvertently missed during an initial character recognition process. *Id.*

In short, *Arai* and *Fenwick* are fundamentally different recognition processes than the ’428 patent, and are the very types of prior art recognition processes that the ’428 patent sought to improve.

Second, because Petitioners’ primary references are directed to entirely different recognition processes than the ’428 patent, they fail to disclose or suggest multiple claim limitations, including limitations related to identifying “portions” of hypothetical candidate character strings which may be automatically committed to an application and flushed from memory.

Third, a POSITA would not be motivated to combine Petitioners' references and would not have a reasonable expectation of success in doing so. As an initial matter, a POSITA would not have been motivated to combine *Arai*, a Japanese patent application disclosing a handwriting recognition algorithm for the Japanese language (a language which does not have spaces between words) with *Fenwick*, a U.S. patent directed to correcting spacing errors which occur when inputting English language words.

Moreover, a POSITA would not have been motivated to combine a Japanese handwriting recognition algorithm with an English spacing correction algorithm because handwriting and character recognition processes have significant differences between different language models, and can even differ significantly for the same language depending on dialect, nationality, education, and other factors. Indeed, *both* parties' experts agree that there are significant differences in the structure and design of different language models, and that a number of challenges exist in attempting to combine recognition processes from different languages. But neither Petitioners, nor their expert, address how a POSITA would have a reasonable expectation of success in combining such different algorithms.

Patent Owner thus respectfully requests that the Board confirm patentability of the challenged claims.

II. PROCEDURAL BACKGROUND

On October 13, 2023, Patent Owner sued Samsung Electronics Co. Ltd. and Samsung Electronics America, Inc. (together, “Samsung” or “Petitioners”) in district court for infringing Patent Owner’s patents, including U.S. Patent No. 9,026,428 (the “’428 patent”). *Cerence Operating Company v. Samsung Electronics Co. Ltd. et al.*, 2:23-cv-00482, ECF No. 1 (E.D. Tex. Oct. 13, 2023).

On August 9, 2024, Petitioners filed their petition for *Inter Partes* Review, challenging claims 1-3, 7-9, and 12-13 of the ’428 patent. Petitioners structure their two asserted grounds for unpatentability as follows:

- **Ground 1:** Obviousness of claims 1-3, 7-9, and 12-13 in view of *Arai*¹ and *Fenwick*².
- **Ground 2:** Obviousness of claims 9 and 12-13 in view of *Arai*, *Fenwick*, and *Baker*³.

III. THE ’428 PATENT

The ’428 patent seeks to solve problems with prior art character recognition methods which did not provide contextual analysis for character and

¹ JP Pat. App. JP 2006-323502 A, published Nov. 30, 2006 (Ex. 1005 (certified English translation), “Arai”).

² U.S. Pat. No. 5,161,245, issued Nov. 3, 1992 (Ex. 1006, “Fenwick”).

³ EP Pat. App. EP0-830-668 B1, published Oct. 31, 2001 (Ex. 1007, “Baker”).

word recognition, and which further required the user to press a button or take some other action to determine when one character or word ends, which “decreases the speed of character/letter input by the users.” Ex.2007 (Koskinen Decl.), ¶39; ’428 patent, 1:31-42. As the ’428 patent explains:

The recognition engine or process provides an indication or output as to when one or more characters can be committed and sent to the application, as opposed to requiring a user to input of a space gesture, select a commit button, or other user input. In other words, the system provides an automatic and quicker recognition system.

’428 patent, 2:54-59. This analysis of when one or more characters can be committed to an application without further user input is based on innovative contextual analysis, using “elaborate linguistic knowledge” to assist in performing “identification of e.g., the end of one word and the beginning of the next without the need to detect a space gesture between two words.” Ex.2007 (Koskinen Decl.), ¶39; ’428 patent, 2:59-62. Specifically, “the use of linguistic tools with an intermediary buffer allows the system to use a ‘history’ of characters input by a user to increase the speed and accuracy of handwritten characters input to the system” which allows the system to “automatically decide when to clear this intermediate buffer by analyzing the context of two or more characters or words.” Ex.2007 (Koskinen Decl.), ¶39; ’428 patent, 2:59-62.

Figure 2, shown below, provides a high-level overview of the '428 patent's recognition process:

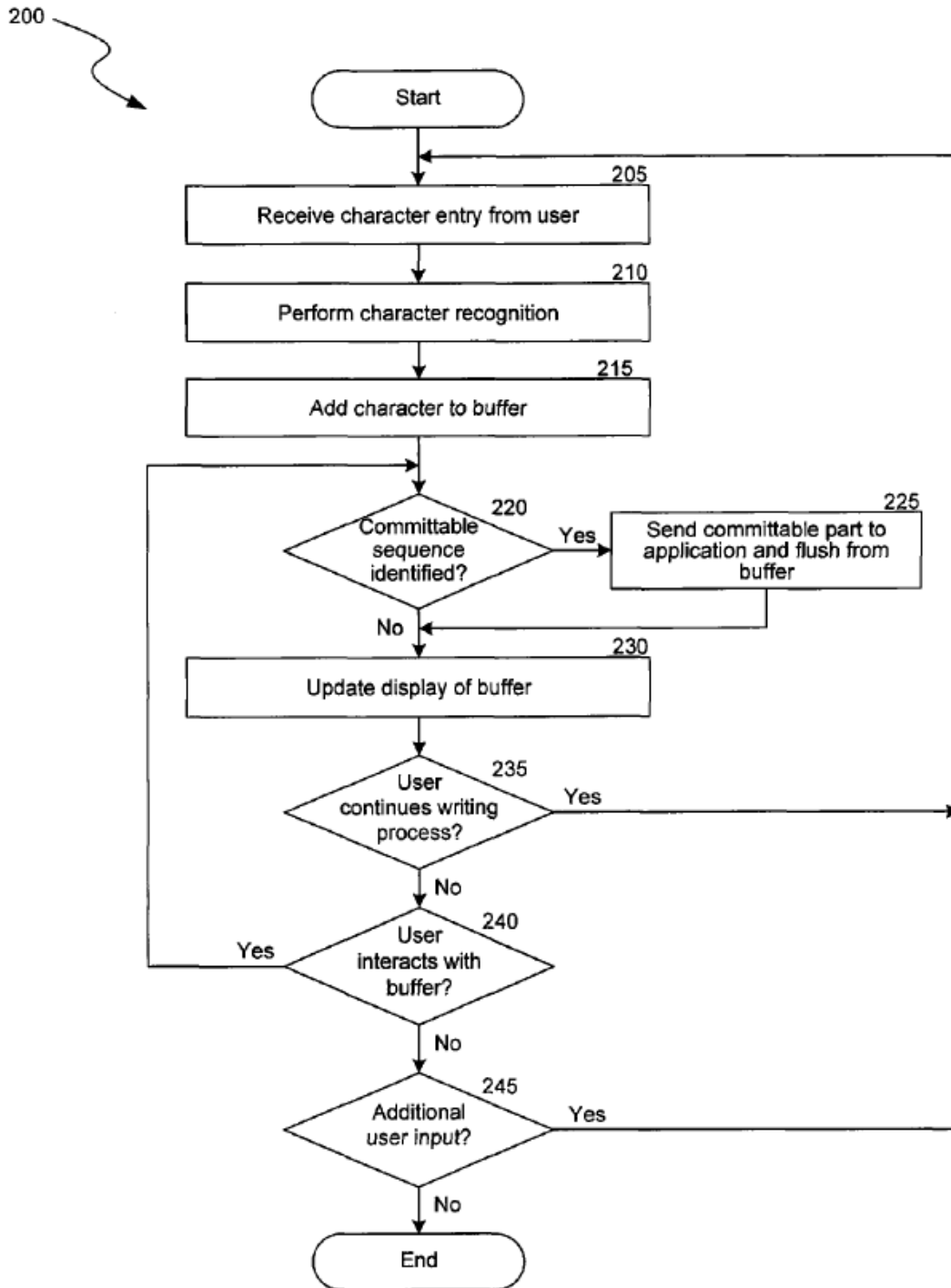


FIG. 2

Step 1: Receiving User Input

Referring to Figure 2 above, the recognition process begins at block 205 by allowing a user to enter the character through “one or a series of strokes or input features (e.g., using a finger and/or a stylus on a touch screen interface).” *Id.*, 4:36-38.

Step 2: Initial Character Recognition

Once this input from the user is received, at block 210 the process will perform character recognition on the input by, for example, “comparing a best match between one or more strokes/gestures input at a touch-sensitive input device to a database of recognized characters.” *Id.*, 4:41-43.

Step 3: Storing Recognized Characters In A Temporary Buffer

At block 215, after finding a best match, the process 200 adds the character to a temporary editable character buffer, which may be presented to the user. *Id.*, 4:43-46.

Step 4: Generating A Hypothetical Candidate Character Sequence

Next, at block 235, the process generates and displays hypothetical “candidate sequences” based on the current characters stored in the temporary memory buffer, which is a “hypothesis” of the character strings, words, or phrases that user may be trying to enter. *Id.*, 3:11-15, 5:13-30. The recognition process thus uses the intermediate buffer to “allow[] the system to use a

‘history’ of characters input by the user” to predict what the user may be trying to enter, “to increase the speed and accuracy of handwritten characters input to the system.” Ex.2007 (Koskinen Decl.), ¶39; ’428 patent, 2:66-3:2. The ’428 patent explains that the candidate character sequence may be determined to be the “best match or most probable candidate sequence, based on the user input, which is currently stored in the buffer.” ’428 patent, 6:4-8. The specification also explains that “[t]he recognition engine may select the candidate sequences 322 from one or more dictionaries contained therein based on, for example, user history and/or character context.” *Id.*, 6:61-65.

Step 5: Presenting The Hypothetical Candidate Sequence For Editing

This candidate sequence in the buffer can then be modified by the user in four ways: 1) the user can continue to enter more characters, which would cause the recognition engine to continue updating the buffer; 2) the user is able to select a candidate from the candidate list, which might be achieved with some gesture or specialized button; 3) the user can modify the text in the application directly; and finally, 4) the system may auto-commit at least a portion of the candidate sequence to an application. ’428 patent, 3:44-46; Ex.2007 (Koskinen Decl.), ¶48.

Step 6: Identifying A *Portion* Of The Candidate Sequence That May Be Sent To The Application And Cleared From Memory

Next, at block 220, the recognition process identifies whether any *portion* of the hypothetical candidate sequence may be committed to an application, based on the content of the temporary memory buffer. Ex.2007 (Koskinen Decl.), ¶54. As the specification explains “based on a sequence of one or more characters in the current buffer, the process 200 determines whether at least a *portion* of the sequence in the temporary editable character buffer is a committable sequence.” ’428 patent, 4:47-50 (emphasis added). The specification further explains that identification of a committable portion of the candidate sequence may be determined by “analyz[ing] the sequence of characters currently stored in the buffer for a match with certain rules, a database or dictionary.” *Id.*, 4:50-54.

If a committable sequence portion is identified in block 220, “the process 200 proceeds to block 225 in which the committable part of the sequence is sent to the current application and removed from the buffer.” *Id.*, 4:58-60. As the specification explains, “[t]he engine uses linguistics and or other heuristic rules to identify that part of the current content can be flushed from the engine buffer (and no longer be modifiable by the engine).”

Use Of Contextual Analysis

Finally, the '428 patent explains that the recognition process may use “contextual analysis” at each stage of the recognition process. Ex.2007 (Koskinen Decl.), ¶74; '428 patent, 3:2-4, 3:36-37, 4:50-58, 6:63-65, 9:42-49, 12:31-34, 13:26-34. Due to the innovative contextual analysis at each stage in the recognition process, the system provides an automatic and quicker recognition system than those disclosed in the prior art. Ex. 2007 (Koskinen Decl.), ¶74. Moreover, the '428 patent's use of advanced contextual analysis allows for character recognition across multiple languages. For example, as the specification explains:

‘if the system utilizes both Chinese and English language dictionaries for a bilingual user, then the user context database 626 may track the relative frequency with which the user selects Chinese words versus English words in order to determine the users ‘primary language,’ and then words from the dictionary may be preferred or presented first over words from the other dictionary.

'428 patent, 14:39-45.

IV. THE SCOPE AND CONTENT OF THE PRIOR ART

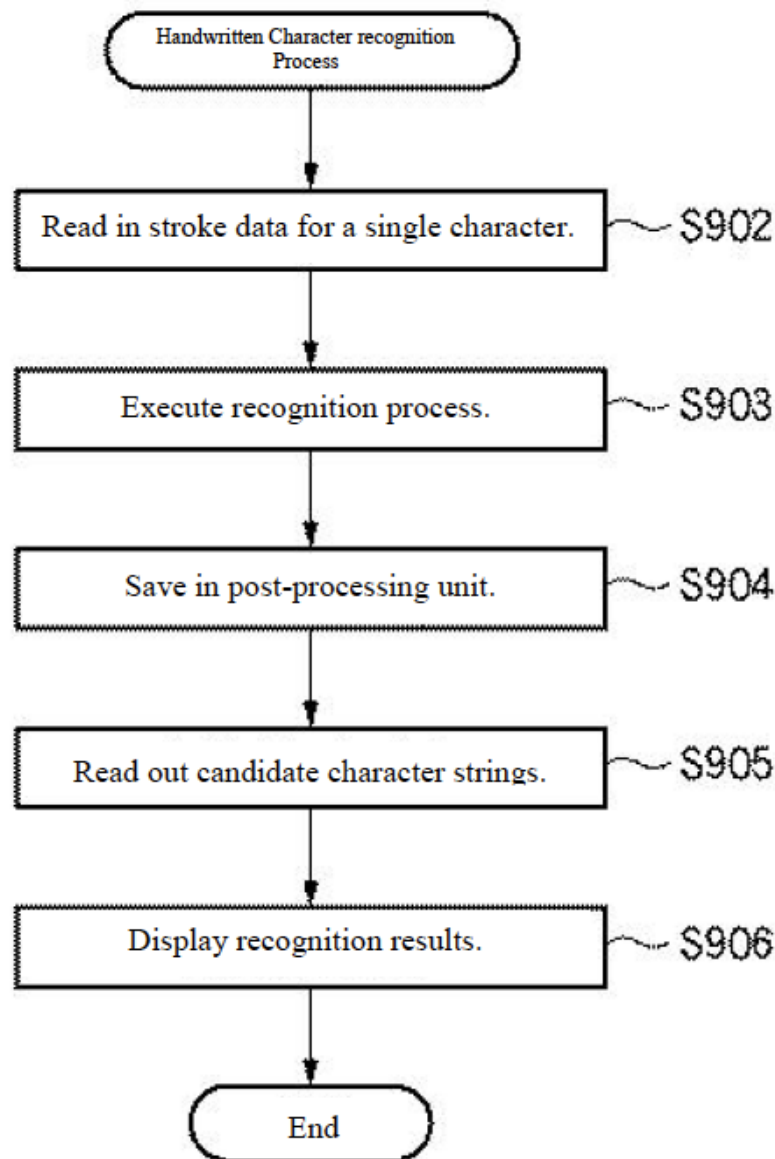
A. *Arai*

Arai is a Japanese patent application which discloses a system and method for recognizing handwritten Japanese characters. Ex. 2007 (Koskinen Decl.), ¶59. Specifically, the “invention relates to an information processing device and controlling method and program thereof for inputting stroke data

of a handwritten character and performing character recognition, and executing post-processing in relation to the character recognition result.” *Arai* ¶[0001].

The *Arai* character recognition process is shown in Figure 8 of the patent (shown below). As shown in Figure 8, the *Arai* character recognition process: 1) performs initial character recognition (steps 902-903); 2) performs post-processing using a common dictionary lookup to see whether an entered string of characters matches a known word (step 904); and 3) then displays the results to the user for editing (steps 905-906). Ex. 2007 (Koskinen Decl.), ¶60; *Arai* ¶¶[139]-[154], Fig. 8 (below).

[FIG. 8]



Notably, the post-processing continuation determination unit in Arai *relies on user input*, as the individual process types evaluate whether to stop post-processing or to append a character to the character string. Ex. 2007 (Koskinen Decl.), ¶¶61-63 Arai ¶¶[0085]-[0091], Fig. 4. Arai further requires the user to confirm that the character that is displayed is the character desired

by the user: “if the character displayed by the handwritten character input screen 53 is the character that is desired by the user, the user confirms this character to insert.” *Arai* ¶[0075]. *Arai* further requires hitting the Enter Key to send a word (or words) to the application. *Id.*, ¶[0091].

B. *Fenwick*

Fenwick is a U.S. Patent directed to “[a] pattern recognition system having inter-pattern spacing correction” which allows for “correcting spacing errors” that may occur when inputting text into the system. Ex. 2007 (Koskinen Decl.), ¶64; *Fenwick*, Abstract; *id.*, 1:6-10. As the patent explains:

In a typical pattern recognition system, various pattern groups (which may be as small as a single individual pattern) are formed from the pattern data by measuring the inter-pattern spacing distance between adjacent individual patterns. Adjacent patterns which are spaced apart from each other by less than a threshold inter-pattern distance are typically combined together to form one pattern group. Adjacent patterns spaced apart by more than the threshold amount are assumed to correspond to next adjacent pattern groups.

Fenwick, 1:32.

Fenwick further explains that “a significant problem” with prior pattern recognition techniques “is that a relatively large inter-pattern distance between adjacent patterns can sometimes be mistakenly identified as a space between pattern groups, resulting in a single pattern group being split into separate, and generally incomprehensible, parts.” *Id.*, 1:33-38. For example,

“sloppy handwriting” can create “irregular inter-character spacing” which “may result in the letters within a word being incorrectly recognized as belonging to separate words.” *Id.*, 1:38-45. As *Fenwick* further explains, “[i]nter-character spacing problems are of particular significance in handwriting recognition systems where writing styles and writing clarity can vary substantially.” *Id.*, 1:45-48.

To solve these spacing problems, *Fenwick* employs a system which: 1) stores each input character in one or more buffers in memory; 2) compares the last entry of each buffer against a dictionary of known words; 3) copies the last entries which match words to new buffers, while placing a space after the last entry in the old buffer; 4) deletes a buffer when a last entry from that buffer does not at least match the beginning of a known word in the dictionary; 5) repeats steps (1) through (4) until all the buffers have been deleted or until an end character, such as a carriage return or period has been stored in each buffer; 6) again compares the last entries from each buffer to the dictionary and deleting any buffer having a last entry which is not a known word in the dictionary or which does not end with a space followed by an end character; and then 7) saves and/or outputs the remaining words. *Id.*, Abstract; Cl. 1.

C. *Baker*

Baker is a European patent that “relates generally to methods and systems for word recognition, and more particularly, to methods and systems that employ context and meaning to recognize a word, or a series of words.”

Ex.2007 (Koskinen Decl.), ¶67; *Baker*, ¶[0001]. As the patent explains:

Accordingly, the invention, in one aspect, provides systems and methods that perform semantic analysis by analyzing language knowledge represented as, in one example, raw text, and employing the language knowledge for dynamically computing the semantic similarity between the words in a “sample passage” and the words in the “current passage” being analyzed.

Id., ¶[0024].

Notably, *Baker* is primarily directed toward improved speech recognition, rather than handwriting or character recognition. Ex. 2007 (Koskinen Decl.), ¶71; *Baker*, ¶¶[0004], [0006], [0032], [0036], [0046], [0068], [0078], [0083]-[0085], [0088]-[0090], [0092]-[0093], [0098], [0099], [0107], and Figure 2).

V. LEVEL OF ORDINARY SKILL IN THE ART

Petitioner proposes that a person of ordinary skill in the art (“POSITA”) at the time of the priority date would have had “at least a Bachelor’s of Science Degree (or equivalent) in electrical engineering, computer science, or computer engineering, or a related technical field and two years of experience working with handwriting recognition, user interfaces incorporating on-line

recognition technologies, or related technical fields.” Petition at 10–11. Petitioner also notes that “[a] greater amount of education could compensate for fewer years of work experience, and vice versa.” *Id.*, 11.

For the purposes of this response, Patent Owner does not dispute the POSITA definition proposed by Petitioner.

VI. CLAIM CONSTRUCTION

Petitioners state that “no express constructions of the claims are necessary” and state that “the claims should be given their plain and ordinary meaning.” Petition at 10–11. Patent Owner agrees that no construction of the terms is required to decide the patentability of the challenged claims. As Patent Owner explains further below, Petitioners’ invalidity arguments should be rejected when applying the plain and ordinary meaning to the relevant claim terms.

VII. APPLICABLE LEGAL STANDARDS

The U.S. Supreme Court set forth the framework for applying the statutory language of 35 U.S.C. § 103 in *Graham v. John Deere Co. of Kansas City*:

Under § 103, the scope and content of the prior art are to be determined; differences between the prior art and the claims at issue are to be ascertained; and the level of ordinary skill in the pertinent art resolved. Against this background, the obviousness or nonobviousness of the subject matter is determined. Such sec-

ondary considerations as commercial success, long felt but unsolved needs, failure of others, etc., might be utilized to give light to the circumstances surrounding the origin of the subject matter sought to be patented.

Graham v. John Deere Co. of Kansas City, 383 U.S. 1, 17–18 (1966).

An obviousness determination requires finding that a POSITA would have been motivated to modify or combine teachings in the prior art. *In re Stepan Co.*, 868 F.3d 1342, 1345–46 (Fed. Cir. 2017). “Whether an ordinarily skilled artisan would have been motivated to modify the teachings of a reference is a question of fact.” *WBIP, LLC v. Kohler Co.*, 829 F.3d 1317, 1327 (Fed. Cir. 2016) (citations omitted). “[W]here a party argues a skilled artisan would have been motivated to combine references, it must show the artisan ‘would have had a reasonable expectation of success from doing so.’” *Arctic Cat Inc. v. Bombardier Recreational Prods. Inc.*, 876 F.3d 1350, 1360–61 (Fed. Cir. 2017) (quoting *In re Cyclobenzaprine Hydrochloride Extended-Release Capsule Patent Litig.*, 676 F.3d 1063, 1068–69 (Fed. Cir. 2012)).

Moreover, obviousness concerns “whether a skilled artisan not only *could* have made but *would* have been motivated to make the combinations or modifications of prior art to arrive at the claimed invention.” *Belden Inc. v. Berk-Tek LLC*, 805 F.3d 1064, 1073 (Fed. Cir. 2015). The “mere fact that these possible arrangements existed in the prior art” does not provide a moti-

vation to substitute the arrangement in one prior art embodiment with the arrangement in another.” *Virtek Vision International ULC v. Assembly Guidance Systems, Inc.*, 97 F.4th 882, 887 (Fed. Cir. 2024). Because a motivation to modify or combine is part of the *prima facie* case for obviousness that must be set forth in the petition, a petitioner should not be permitted to present “an entirely new rationale to explain why one of skill in the art would have been motivated to combine” in its reply. *Intelligent Bio-Systems, Inc. v. Illumina Cambridge Ltd.*, 821 F.3d 1359, 1370 (Fed. Cir. 2016).

“[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.” *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006). The Board “may reject even uncontroverted expert testimony when it is intrinsically unpersuasive.” *Granite Constr. Co. v. United States*, 962 F.2d 998, 1006 (Fed. Cir. 1992). In short, petitioner cannot satisfy its burden of proving obviousness by employing “mere conclusory statements.” *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016).

VIII. THE PETITION FAILS TO ESTABLISH THE OBVIOUSNESS OF ANY CHALLENGED CLAIM

A. Ground 1: Petitioners Fail To Show That Claims 1-3, 7-9, and 12-13 Would Have Been Obvious Based on *Arai* In View Of *Fenwick*

Petitioners contend that *Arai* in view of *Fenwick* renders obvious claims 1-3, 7-9, and 12-13 of the '428 patent. Petitioners are wrong. *Arai* and *Fenwick* are directed to entirely different character and pattern recognition methods than the '428 patent and thus fail to disclose or suggest multiple claim limitations. Moreover, a POSITA would not have been motivated to combine *Arai*, which describes character recognition processes for Japanese (symbol-based) characters, with *Fenwick*, which describes a spacing correction algorithm for English (phonetic-based) characters. Additionally, because the processes for recognizing and disambiguating characters between these different languages differ considerably, a POSITA would not have a reasonable expectation of success in combining these references.

1. *Arai* And *Fenwick* Are Directed To Fundamentally Different Approaches To Character And Pattern Recognition Than The '428 Patent

a. The '428 Patent

As discussed above, the '428 patent seeks to solve problems with prior art character recognition methods which did not provide contextual analysis for character and word recognition, and which further required the user to

press a button or take some other action to determine when one character or word ends, which “decreases the speed of character/letter input by the users.”

Ex.2007 (Koskinen Decl.), ¶74; ’428 patent, 1:31-42.

This analysis of when one or more characters can be committed to an application without further user input is based on innovative contextual analysis, using “elaborate linguistic knowledge” to assist in performing “identification of e.g., the end of one word and the beginning of the next without the need to detect a space gesture between two words.” ’428 patent, 2:59-62; Ex. 2007 (Koskinen Decl.), ¶74. Specifically, “the use of linguistic tools with an intermediary buffer allows the system to use a ‘history’ of characters input by a user to increase the speed and accuracy of handwritten characters input to the system” and to “automatically decide when to clear this intermediate buffer by analyzing the context of two or more characters or words.” ’428 patent, 2:59-62; Ex.2007 (Koskinen Decl.), ¶74.⁴

⁴ The ’428 patent explains that “Context information is any information that may help inform the recognition engine of the users intent, and may include information Such as sentence or word-pair context, language context, application-specific context (e.g. data fields/tags), user-specific context, location, time of day etc.” ’428 patent, 12:34-39.

This novel process allows the '428 patent's recognition process to improve upon prior art systems which were unable to perform advanced contextual analysis throughout the character recognition process, and which lacked the ability to analyze portions of entered character sequences to determine whether to commit that portion of a character sequence to an application and flush any memory buffer *without* requiring additional user input. Ex. 2007 (Koskinen Decl.), ¶74.

As discussed above, *supra* Section III, the '428 patent's recognition process involves six primary steps: 1) receiving a handwriting character entry from the user; 2) performing initial character recognition on the user's input; 3) storing the recognized character in a temporary memory buffer; 4) generating a hypothetical candidate character sequence based on the context of the characters currently in the memory buffer; 5) presenting the hypothetical candidate character sequence to the user for editing; and 6) identifying if any portion of the candidate character sequence may be automatically committed to an application and flushed from the temporary memory buffer. Ex. 2007 (Koskinen Decl.), ¶76.

The '428 patent's recognition process differs from other prior handwriting recognition and post-processing methods in that it can perform contextual analysis at *multiple* different stages. Ex.2007 (Koskinen Decl.), ¶78. For

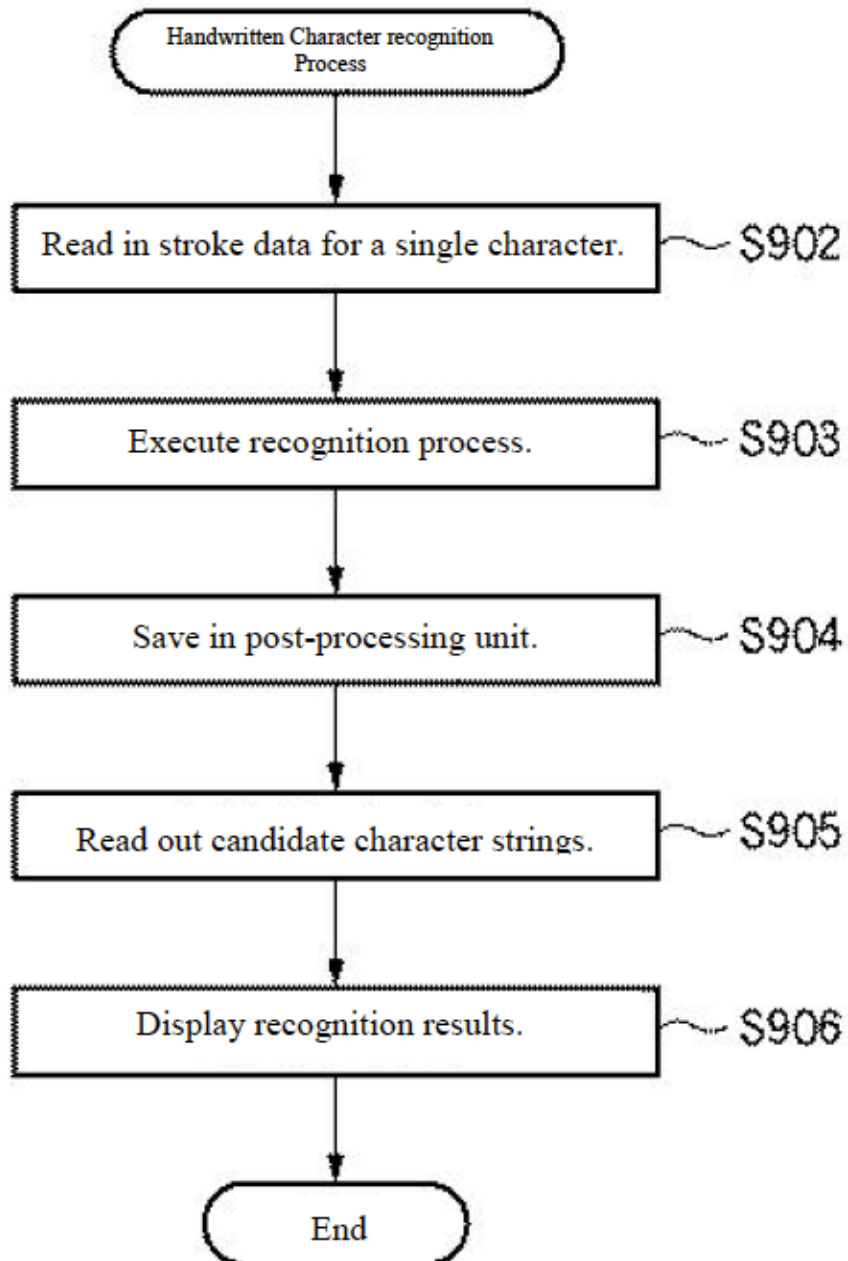
example, the specification explains that contextual analysis may be applied: 1) during the initial character recognition process ('428 patent, 12:31-34); 2) during the initial selection of the hypothetical character candidate sequences (*id.*, 6:63-65, 13:26-34); 3) during the subsequent identification of any sub-portions of the character candidate sequences which may be automatically committable to the application and flushed from the memory buffer (*id.*, 4:50-58, 9:42-49); 4) when deciding whether to clear the temporary memory buffer (*id.*, 3:2-4); and 5) even after words are committed to the application (*id.*, 3:36-37). Ex.2007 (Koskinen Decl.), ¶78.

Moreover, the '428 patent's novel process involving multiple stages of contextual analysis allows for the application of character recognition across *multiple* languages, a significant challenge at the time of the invention due to the vast differences in character recognition and post-processing algorithms used for different languages. Ex. 2007 (Koskinen Decl.), ¶74; *see also* Ex. 2009 (Andersson Decl.) ¶¶3-15 (discussing the challenges of combining handwriting recognition and post-processing algorithms used for different languages); Ex. 2010 (Ward Dep. Tr.), 28:1-10, 28:24-29:1, 29:11-19. 32:1-12, 33:1-7, 34:2-6, 34:8-24, 35:3-21, 35:24-36:2, 36:10-13, 50:1-51:14, 51:24-52:3, 52:20-53:10, 53:17-18, and 57:9-15.

b. *Arai*

Conversely, as shown in Figure 8 of *Arai* (below), along with the corresponding disclosure in *Arai*'s specification (*see* ¶¶ [139]-[154]), *Arai*'s recognition process is directed to the exact type of prior art systems and methods that the '428 patent was trying to improve upon (i.e., a process which provides ***no*** contextual analysis, ***no*** ability for predictive text generation, and ***no*** ability to automatically commit portions of candidate character sequences to an application and flush corresponding memory ***without*** additional user input). Ex. 2007 (Koskinen Decl.), ¶80. Specifically, as shown below in Figure 8, the *Arai* recognition process simply performs initial character recognition (steps 902-903), performs basic post-processing using a common dictionary lookup to see whether an entered string of characters matches a known word (step 904), and then displays the results to the user for editing (steps 905-906). *Id.*

[FIG. 8]



In short, while *Arai* performs an initial character recognition, and then generates a probable candidate character sequence to display to the user for editing, *Arai*'s process fails to perform the '428 patent's step of *further* iden-

tifying a *portion* of any probable candidate character string that may be automatically committed to an application and flushed from the memory buffer (*without requiring further user input*). Ex. 2007 (Koskinen Decl.), ¶80. Thus, *Arai* is missing the primary feature the '428 patent explains is important for improving speed and accuracy over the prior art. *Id.*, ¶74. In fact, contrary to the '428 patent's novel approach of automatically committing character sequences to an application, *Arai*'s post-processing unit expressly ***requires*** input to proceed. *Id.*, ¶81; *Arai* ¶¶[0085]-[0091], Fig. 4. Indeed, as the Petition admits, "*Arai requires* hitting the Enter Key to send a word (or words) to the application (*Arai* ¶[0091])." Petition at 31 (emphasis added).

Moreover, while the '428 patent further enhances the speed of its recognition process by using contextual analysis to predict which words or phrases the user may be trying to enter, *Arai*'s identification of probable character sequences involves a mere dictionary lookup to determine if the current character string matches an exact full word in a dictionary. Ex. 2007 (Koskinen Decl.), ¶83; *Arai* ¶[146]. Indeed, *Arai* lacks any disclosure of using contextual analysis at ***any*** stage of the recognition process, further limiting the speed and accuracy of *Arai*'s recognition process when compared to the '428 patent's process. Ex. 2007 (Koskinen Decl.), ¶¶83-84.

Finally, while the '428 patent's use of advanced contextual analysis allows for character recognition across multiple languages (*supra* Section III), *Arai*'s lack of advanced contextual analysis significantly limits its ability to be effectively used with more than one language. Ex. 2007 (Koskinen Decl.), ¶84; Ex. 2009 (Andersson Decl.) ¶¶3-15; Ex. 2010 (Ward Dep. Tr.), 28:1-10, 28:24-29:1, 29:11-19, 32:1-12, 33:1-7, 34:2-6, 34:8-24, 35:3-21, 35:24-36:2, 36:10-13, 50:1-51:14, 51:24-52:3, 52:20-53:10, 53:17-18, and 57:9-15.

c. *Fenwick*

In contrast to the '428 patent, which is a “linguistically aided autocommit system for handwriting recognition user interfaces” ('428 patent, 2:39-40), *Fenwick* merely discloses a spacing correction algorithm. *Supra* Section IV; Ex. 2007 (Koskinen Decl.), ¶85; *Fenwick*, 1:6-10; 1:34, 1:42-45. Specifically, *Fenwick* describes a system and process of storing characters in multiple memory buffers to identify potential word matches, so that it can append a space after any word which may have been inadvertently missed during an initial character recognition process. *Id.* After a full word has been identified and a space appended to the word, the word and space may be committed to an application and the associated buffers cleared. *Supra* Section IV; Ex. 2007 (Koskinen Decl.), ¶86.

Fenwick's process of generating **multiple** memory buffers to identify full words and append spaces to said words differs considerably from the '428 patent's process, which utilizes a **single** intermediary memory buffer to store character sequences so that the process can then use the contents of that single memory buffer for contextual analysis to predict what words or phrases the user may be attempting to enter. Ex. 2007 (Koskinen Decl.), ¶87; '428 patent, 12:31-50. Thus, while the '428 patent takes input characters that have been recognized and stored in memory and uses those characters to generate hypothetical character candidate sequences (to predict what the user may be trying to enter), *Fenwick* completely lacks this feature, instead merely segmenting received input into a series of memory buffers in an attempt to separate recognized words from one another (to append spaces between said words). Ex.2007 (Koskinen Decl.) ¶¶87-88.

Moreover, since *Fenwick* lacks the '428 patent's feature of generating hypothetical character candidate sequences to aid in predicting what the user may be trying to enter, *Fenwick* necessarily lacks the '428 patent's feature of displaying those hypothetical character candidate sequences to the user for potential editing. *Id.*

Nor does *Fenwick* perform any analysis of whether any *portion* of a character sequence may be automatically committed to an application, as *Fenwick* is limited to determining whether full words have been identified (to append spaces to the end of those full words), and then committing those full words and appended spaces to an application. Ex. 2007 (Koskinen Decl.), ¶¶86-88. This is contrasted against '428 patent's process, which allows for auto-committing *portions* of character sequences, based on its contextual analysis. '428 patent, 3:44-46 ("The engine uses linguistics and or other heuristic rules to identify that part of the current content can be flushed from the engine buffer"); Ex. 2007 (Koskinen Decl.), ¶¶86-88.

Finally, as with *Arai*, *Fenwick*'s lack of advanced contextual analysis significantly limits its ability to be effectively used with more than one language. Ex. 2007 (Koskinen Decl.), ¶¶86-88; Ex. 2009 (Andersson Decl.) ¶¶3-15; Ex. 2010 (Ward Dep. Tr.), 28:1-10, 28:24-29:1, 29:11-19, 32:1-12, 33:1-7, 34:2-6, 34:8-24, 35:3-21, 35:24-36:2, 36:10-13, 50:1-51:14, 51:24-52:3, 52:20-53:10, 53:17-18, and 57:9-15.

2. Because *Arai* And *Fenwick* Are Directed To Fundamentally Different Approaches To Character And Pattern Recognition Than The '428 Patent, They Fail To Disclose Or Render Obvious Multiple Claim Limitations

Because *Arai* and *Fenwick* are directed to entirely different character and pattern recognition processes than the '428 patent, they fail to disclose or render obvious multiple claim limitations. Ex. 2007 (Koskinen Decl.), ¶90.

a. Claim 1

The steps of Claim 1 of the '428 patent are reproduced below:

1[pre]: A method for data input on a touchscreen of a mobile device, the method comprising:

1[a]: receiving a first user input, wherein the first user input is handwritten input received on the touchscreen of the mobile device;

1[b]: determining a first recognized sequence based on the first user input; wherein the first recognized sequence is a best match of the first user input;

1[c]: writing the first recognized sequence to a memory storage buffer of the mobile device;

1[d]: generating a first candidate sequence, wherein the first candidate sequence is a best match of one or more recognized sequences;

1[e]: presenting the first candidate sequence to the user, wherein the presenting includes displaying the first candidate sequence to the user and allowing the user to edit the first candidate sequence;

1[f]: identifying at least a first sequence portion of the first candidate sequence wherein the identifying is at least partially

based on one or more rules or dictionaries of words accessible to the mobile device;

1[g]: removing the first sequence portion from the memory storage buffer, wherein the removing is performed in the absence of receiving user input editing the first candidate sequence, representing a space character, or selecting a word; and

1[h]: sending the first sequence portion to an application running on the mobile device.

'428 patent, cl. 1.

As discussed further below, Petitioners mischaracterize the disclosure in *Arai* to make *Arai*'s recognition algorithm appear to have more steps than it actually does, in order to map *Arai*'s method onto claim 1 of the '428 patent. Specifically, Petitioners rely on the wrong disclosure in *Arai* for limitations 1[d] and 1[e]. When looking at *Arai*'s actual disclosure, it is clear that *Arai* does not disclose or suggest limitation 1[f], as Petitioners assert, and *Arai* in view of *Fenwick* does not disclose or suggest limitations 1[g] and 1[h].

i. Petitioners Identify The Wrong Disclosure In *Arai* For Limitations 1[d] And 1[e]

Before getting into why Petitioners have failed to meet their burden in establishing that *Arai* discloses or suggests limitation 1[f], or that *Arai* in view of *Fenwick* discloses or suggests limitations 1[g] and 1[h], it is important to note that Petitioners and their expert Mr. Ward mischaracterize the disclosure in *Arai* and identify the **wrong** steps in *Arai*'s handwriting recognition process

as purportedly disclosing limitations 1[d] and 1[e] of the '428 patent, i.e. “generating a first candidate sequence, wherein the first candidate sequence is a best match of one or more recognized sequences” and “presenting the first candidate sequence to the user, wherein the presenting includes displaying the first candidate sequence to the user and allowing the user to edit the first candidate sequence.” Ex. 2007 (Koskinen Decl.), ¶¶91-92. Specifically, as discussed further below, Petitioners and their expert improperly point to the same disclosure in *Arai* regarding *Arai*’s initial character recognition step for **both** limitation 1[b] (i.e., “determining a first recognized sequence...”) **and** limitation 1[d] (i.e., “generating a first candidate sequence...”). *Id.* This is done so that Petitioners can characterize as *Arai* has having more process steps than what is actually disclosed. *Id.*

a. *Arai*’s Actual Recognition Process

As discussed above, the *Arai* recognition process simply reads in character input, performs initial character recognition, executes basic post-processing using a common dictionary lookup to see whether an entered string of characters matches a known word, and then displays the results to the user for editing. Ex. 2007 (Koskinen Decl.), ¶93.

For example, *Arai*’s recognition process is explained in paragraphs [0139] through [0155] of *Arai*, as well as Figure 8 of *Arai* (which shows a

flowchart of *Arai*'s recognition process). Ex. 2007 (Koskinen Decl.), ¶94. As *Arai*'s specification explains, at step S902 of Figure 8, "stroke data for a single character is read into a work area." *Arai*, ¶[0141]; Ex. 2007 (Koskinen Decl.), ¶94.

Next, at step S903, character recognition is performed on the stroke data. *Arai*, ¶[0142]; Ex. 2007 (Koskinen Decl.), ¶95. This process involves calculating "evaluation values" relating to "the values for the degree of matching" of the stroke data to "reference character dictionary data." *Id.* The character codes and the evaluation values corresponding to the reference character dictionary data that have the "highest degrees of matching" are then stored as the "recognition results" for each individual character entered by the user. *Id.* For example, *Arai* provides an example of two individual characters entered by the user "工場", with the recognition process first determining a set of evaluation values for the first character "工" and then subsequently determining a second set of evaluation values for the second character "場". *Arai* ¶¶[0144]-[0145]. Importantly, at this point in the *Arai* recognition process, only *individual* characters have been analyzed, i.e., *no* analysis has been performed on any combination of characters to facilitate generating any candidate character sequences. Ex. 2007 (Koskinen Decl.), ¶95.

Subsequently, at step S904, the recognition result candidates for each individual character entered by the user are sent to the post-processing unit 34, which then performs post-processing on the *combined* character sequences in memory. Ex. 2007 (Koskinen Decl.), ¶96. Specifically, *Arai* explains that “in the post-processing unit 34 the chain probabilities for the character strings *combining these candidates* are calculated using a dictionary for post-processing.” *Arai* ¶[0146] (emphasis added). The post-processing unit 34 then “determines the candidate character string with the highest chain probabilities among these accommodations.” *Id.*

Finally, at steps S905 and S906, the candidate character string is read out from the post-processing unit 34 and the results are displayed for the user to edit. *Id.*, ¶¶[0147]-[0149].

Thus, while *Arai* performs an initial character recognition (i.e., steps S902-S903), and then generates a probable candidate character sequence to display to the user for editing (steps 903-906), *Arai*’s process fails to perform the ’428 patent’s steps of *further* identifying a *portion* of any probable candidate character string that may be automatically committed to an application and flushed from the memory buffer (without requiring further user input). Ex. 2007 (Koskinen Decl.), ¶98.

b. Petitioners' Mischaracterization Of *Arai*'s Recognition Process

Faced with this glaring deficiency in *Arai*'s disclosure, Petitioners and their expert attempt to “double dip” from *Arai*'s disclosed method, pointing to the *same* disclosure in *Arai* for both limitation 1[b] (the initial character recognition limitation), as well as limitation 1[d] (the “generating a first candidate sequence...” limitation). Ex. 2007 (Koskinen Decl.), ¶99. This allows Petitioners and their expert to then mischaracterize *Arai* as performing more steps than are actually occurring. *Id.*

For example, for limitation 1[b] (“determining a first recognized sequence...”), Petitioners point to *Arai*'s initial character recognition process, i.e., the determination of the highest ranked “recognition results candidates” via calculation of the “evaluation values” for each individual character entered. Petition at 17-19. Indeed, Petitioners cite to and walk through the *same* paragraphs Patent Owner discusses above regarding *Arai*'s recognition process, i.e. paragraphs [0142] through [0145] (discussing *Arai*'s calculation of evaluation values and walking through an example of how these recognition results are processed for two sample characters entered by the user, i.e., the characters “工” and “場”). *Id.*

Then, for limitation 1[c] (“writing the first recognized sequence to a memory storage buffer ...”), Petitioners point to the same paragraphs discussed above regarding the individual character recognition results being stored in RAM. Petition at 20-21. Petitioners also again rely on *Arai*’s example walk through of the process, citing to *Arai*’s example of how each individual “character’s recognition results (e.g., each of 工 and 場) are written to a memory storage buffer” *Id.*, 20-21.

Thus, at this point, Petitioners have ***already*** relied on *Arai*’s initial character recognition process (i.e., the determination of the highest “evaluation values” for each individual character entered) for limitation 1[b] and further relied on the storage of these highest-ranked “evaluation values” in memory for limitation 1[c]. Ex. 2007 (Koskinen Decl.), ¶102. Following *Arai*’s example walkthrough of its recognition process, at this point both individual characters “工” and “場” have ***already*** been evaluated and stored in memory together. *Id.*

Petitioners then improperly attempt to rely on the same disclosure and process steps in *Arai* for limitation 1[d], which requires the ***additional*** processing step of “generating a first candidate sequence, wherein the first candidate sequence is a best match of one or more recognized sequences.”

Petition at 21-22. Specifically, Petitioners assert that the recognized characters “工” and “場” form a first candidate sequence “工場”, and that “[b]ecause *Arai* generates 工場 using an ‘evaluation value’ and ‘first-ranked recognition result,’ a POSITA would have understood this first candidate sequence 工場 to be a best match of the first and second recognized sequences (工 and 場).” *Id.* But, as discussed above, Petitioners **already** relied on *Arai*’s determination of “evaluation values” and “first ranked recognition results” for each individual character for limitations 1[b], and the storage of these individual characters together for limitation 1[c]. Ex. 2007 (Koskinen Decl.), ¶103.

Moreover, even if Petitioners had not already relied on these same steps in *Arai* for earlier limitations, Petitioners cannot rely on these steps to satisfy limitation 1[d] of the ’428 patent, because limitation 1[d] requires an **additional** processing step (beyond the initial individual character recognition process step), i.e., processing of the **combination** of characters stored in memory to generate the hypothetical “candidate sequence.” *Id.*, ¶104.

As discussed above, the very first place that *Arai* performs post-processing on the **combined** character string in memory (i.e., generating a “best match of one or more recognized sequences” as required by limitation 1[d]), is **after** the initial character recognition process is performed in *Arai* (step

S903), i.e., when those individual character results are sent to the post-processing unit to calculate the “chain probabilities for the character strings ***combining these candidates.***” Arai, ¶[0146] (discussing step S904); Ex. 2007 (Koskinen Decl.), ¶105.

Indeed, Petitioners’ expert Mr. Ward confirmed during his deposition that “generating a first candidate sequence” is a ***separate*** distinct step from the earlier character recognition step of claim 1, and that (contrary to Petitioners’ arguments) the candidate sequence is not just the output of the initial individual character recognition step; it is instead an additional step of determining a hypothesis of what character sequence the user is trying to enter based on the combination of characters in the memory buffer. *See, e.g.*, Ex.2010 (Ward Dep. Tr.), 106:4-10 (“Q. Since these are different steps in the method, the determining of first recognized sequence is a separate step from the subsequent generating a first candidate sequence; correct? ***A. It’s listed here as a distinct step, yes.***”); *id.*, 109:22-110:8 (“Q. So to my point then, the generation of the first candidate sequence is doing something different than the initial best match determination that's happening at the earlier step with determining a first recognized sequence ... ***A. They are discrete steps, yes, sir.***”); *id.*, 126:10-20 (“Q. These candidate sequences are not just the combination of the individual characters that have been entered; right? They are

instead hypotheses of what the user may be trying to enter; right? *A. I believe you're paraphrasing but, yes....*"); *id.*, 12522-126:3 (“Q. So when the recognition engine is creating these candidate sequences, it's trying to hypothesize what character sequences the user is trying to enter based on analyzing the current content of the characters in their memory buffer; correct? *A. In part, yes.*”).

Further confirmation that Petitioners and their expert are mischaracterizing *Arai's* recognition process can be found in their arguments regarding limitation 1[e]. Ex. 2007 (Koskinen Decl.), ¶115. For example, limitation 1[e] of the '428 patent requires “presenting the first candidate sequence to the user, wherein the presenting includes displaying the first candidate sequence to the user and allowing the user to edit the first candidate sequence.” For this limitation, Petitioners rely on disclosure from *Arai* that the example characters “工場” are displayed “as the recognition result in the input character buffer area 61.” Petition at 22 (citing *Arai* ¶[0151]). Petitioners then cite to disclosures in *other* parts of *Arai* regarding the ability to edit characters in the input area. *Id.* (citing *Arai* ¶¶[0051], [0088], [0089]).

However, when explaining how its method works “as a time series” (i.e., in linear order), *Arai* makes it clear that editing of the characters in the

input area does not occur until *after* the post-processing unit performs its analysis of the full character strings in memory to calculate the chain probabilities. See *Arai* ¶¶[0150]-[0154]. Put simply, the example characters “工場” that Petitioners rely on as being displayed are *not* editable by the user as required by limitation 1[e], because (once again) they are *not* the candidate character sequence determined by the post-processing unit in *Arai*, they are merely the results of the initial individual character recognition step. Ex. 2007 (Koskinen Decl.), ¶116; see also Ex. 2010 (Ward Dep. Tr.), 210:18-21 (testifying that *Arai* allows editing *after* the post-processing step).

To be clear, Patent Owner does not dispute that *Arai* discloses or suggests limitation 1[d] requiring “generating a first candidate sequence, wherein the first candidate sequence is a best match of one or more recognized sequences,” as this occurs in *Arai* at step S904, when the post processing unit calculates “the chain probabilities for the character strings” and then “determines the candidate character string with the highest chain probabilities....” *Arai*, ¶[0146]. Ex. 2007 (Koskinen Decl.), ¶117. Nor does Patent Owner dispute that *Arai* discloses or suggests limitation 1[e], requiring “presenting the first candidate sequence to the user, wherein the presenting includes displaying the first candidate sequence to the user and allowing the user to edit

the first candidate sequence,” as this occurs in *Arai* at steps S905-S906, when the candidate sequence is displayed to the user for editing.

However, while Patent Owner does not dispute that limitations 1[d] and 1[e] are disclosed or suggested in *Arai*, it is still important to clarify that Petitioners are identifying the wrong disclosure in *Arai* for these limitations because Petitioners’ mischaracterization allows them to stretch *Arai*’s process out into more steps than actually occur.

For example, by incorrectly relying on *Arai*’s initial character recognition process (i.e., step 903) for limitation 1[d] (requiring “generating a first candidate sequence...”), Petitioners are then able to misrepresent *Arai*’s sole post-processing step S904 (i.e., calculating “the chain probabilities for the character strings”) as being the **subsequent** processing steps required by the ’428 patent, including limitations 1[f] through 1[h] requiring “identifying at least a portion of the candidate sequence” that may be automatically committed to an application and flushed from the memory buffer. *Id.*, ¶118.

But as explained above, and addressed further below for each corresponding limitation, *Arai* does **not** disclose or suggest any subsequent processing steps after the generation of the candidate character sequence, and thus does not disclose or suggest limitations 1[f] through 1[h], requiring **fur-**

ther processing to identify at least a *portion* of said candidate character sequence (or automatically committing said portion to an application and flushing said portion from the memory buffer). *Id.*, ¶119.

- ii. **Element 1[f]: “identifying at least a first sequence portion of the first candidate sequence wherein the identifying is at least partially based on one or more rules or dictionaries of words accessible to the mobile device;”**

Arai does not disclose or suggest this limitation. Ex. 2007 (Koskinen Decl.), ¶120.⁵

Petitioners rely on *Arai*’s post-processing step S904 (i.e., calculation of the chain probabilities for the character strings) as satisfying this limitation. *See* Petition at 23-26. But as discussed above, step S904 of *Arai*’s recognition process is where *Arai* generates the candidate character sequence. Ex. 2007 (Koskinen Decl.), ¶121. At **no** point during this post-processing step, or any time after this step, does *Arai* perform any **additional** processing to identify any “first sequence portion” of said candidate character sequence, as required by limitation 1[f]. *Id.*, ¶¶121-129.

Indeed, Petitioners’ arguments regarding this limitation borderline on nonsensical. Specifically, because there is no disclosure in *Arai* of identifying any “portions” of the candidate character sequence, Petitioners simply assert

⁵ Petitioners do not contend that *Arai*, in view of *Fenwick*, discloses this limitation.

that step S904's identification of the character string with the highest chain probability (i.e., “工場” in the example shown in *Arai*) **necessarily** includes an identification of a sub-portion of the candidate character string. Petition at 26. As Petitioners argue, because the example character string includes both the individual characters “工” and “場”, then “工” is a portion of the first candidate sequence 工場 (‘identifying at least a first sequence portion of the first candidate sequence’).” *Id.*

Petitioners' argument is, of course, plainly wrong. If the mere generation of the candidate sequence of limitation 1[d] **automatically** satisfied the separate processing step of limitation 1[f] (requiring identification of at least a portion of the candidate character sequence), then limitation 1[f] would be superfluous to claim 1. Ex. 2007 (Koskinen Decl.), ¶130; *see also Intel Corp. v. Qualcomm Inc.*, 21 F.4th 801, 810 (Fed. Cir. 2021) (“It is highly disfavored to construe terms in a way that renders them void, meaningless, or superfluous”).

Moreover, even if *Arai*'s recognition process did perform some identification of a “portion” of the candidate character sequence—and it does not—Petitioners do not identify **any** disclosure in *Arai* of said portion being identified “at least partially based on one or more rules or dictionaries of words accessible to the mobile device” as required by limitation 1[f]. Ex. 2007

(Koskinen Decl.), ¶130. Specifically, Petitioners point to the fact that the chain probabilities are calculated by referencing dictionary data. Petition at 24-25. But the result of *Arai*'s dictionary comparison is merely the **full** candidate character string (i.e., “工場,” in the example shown in *Arai*) (see *Arai* ¶¶[0146]-[0152]); the dictionary lookup is **not** being used to identify of any “portion” of the candidate character sequence as required by limitation 1[f]. Ex. 2007 (Koskinen Decl.), ¶130.

In short, Petitioners' attempts to cobble together the requirements of limitation 1[f] from multiple disparate parts of *Arai* fail in view of the plain disclosure in *Arai*'s specification. See e.g., *Abbott Laboratories v. Sandoz, Inc.*, 544 F.3d 1341, 1348 (Fed. Cir. 2008) (refusing to reverse the district court's validity finding, stating, “in addressing the question of obviousness a judge must not pick and choose isolated elements from the prior art and combine them so as to yield the invention in question if such a combination would not have been obvious at the time of the invention”). Here, contrary to Petitioners' assertions, *Arai* fails to disclose or suggest identifying any “first sequence portion of the first candidate sequence” and further fails to disclose or suggest said “first sequence portion” being identified “at least partially based on one or more rules or dictionaries....” Ex. 2003 (Koskinen Decl.), ¶132.

- iii. **Element 1[g]: “removing the first sequence portion from the memory storage buffer, wherein the removing is performed in the absence of receiving user input editing the first candidate sequence, representing a space character, or selecting a word;”**

Arai in view of *Fenwick* fails to disclose or suggest this limitation. Ex. 2007 (Koskinen Decl.), ¶133.

As discussed above, *Arai* does not disclose or suggest identification of any “first sequence portion of the first candidate sequence,” as required by limitation 1[f]. Ex. 2007 (Koskinen Decl.), ¶134. Thus, *Arai* necessarily can neither disclose nor suggest the subsequent limitations requiring action on that first sequence portion (i.e., limitations 1[g] and 1[h]). *Id.* Additionally, because Petitioners do not contend that *Fenwick* discloses or suggests identification or use of any first sequence portion of the first candidate sequence, *Arai* and *Fenwick* together also fail to disclose or suggest this limitation. *Id.*

Arai in view of *Fenwick* further fails to disclose or suggest the remainder of this limitation, i.e., removing the first sequence portion from the memory buffer, “wherein the removing is performed in the absence of receiving user input editing the first candidate sequence, representing a space character, or selecting a word.” *Id.*, ¶134. As Petitioners admit, “*Arai* is silent on when and under what conditions the results are remove from RAM.” Petition

at 26. And while Petitioners point to disclosure in *Fenwick* regarding automatically deleting identified words from memory buffers, as discussed below, a POSITA would not have been motivated to combine *Arai* and *Fenwick* and would further not have a reasonable expectation of success in combining these references. Ex. 2007 (Koskinen Decl.), ¶134.

First, a POSITA would not have been motivated to combine *Arai* and *Fenwick*, because, as discussed above, *Arai* is a Japanese patent application disclosing a handwriting recognition algorithm for the Japanese language (a language which does not utilize spaces between words), while *Fenwick* is an English patent disclosing an algorithm for correcting spacing errors which commonly occur when reading in English characters. Ex. 2007 (Koskinen Decl.), ¶135. Simply put, a POSITA would not have been motivated to combine an algorithm for fixing spacing errors between words with an algorithm designed to recognize characters and words in a language which does not utilize spaces between words. *Id.*

Second, a POSITA would not have been motivated to combine *Arai*'s Japanese handwriting recognition algorithm with *Fenwick*'s English spacing correcting algorithm, because both the initial recognition process, as well as the post-processing that occurs on any recognized characters, differs substantially between different languages. *Id.*, ¶136; *see also* Ex. 2009 (Andersson

Decl.), ¶¶3-15; Ex. 2010 (Ward Dep Tr.), 28:1-10, 28:24-29:1, 29:11-19, 32:1-12, 33:1-7, 34:2-6, 34:8-24, 35:3-21, 35:24-36:2, 36:10-13, 50:1-51:14, 51:24-52:3, 52:20-53:10, 53:17-18, and 57:9-15.

For example, the differences between different language recognition models and challenges with combining different models was frequently discussed in academic publications prior to the priority date of the '428 patent.

Ex. 2007 (Koskinen Decl.), ¶137. For example, as explained in *Handwriting:*

Recognition, Development and Analysis:

There are several factors that make character recognition a more complex task than digit recognition. In this scenario, one takes into account the existence of a greater number of classes to be recognized (this set can be composed of digits, uppercase, lowercase or accented characters, punctuation and symbols). The variation of the calligraphy of the individuals during the act of writing over time, are characteristics that also make the process of recognition of handwritten characters a complex activity. It is, therefore, noticeable that there is a high level of variability of the instances that can be correctly assigned to the same class. On the other hand, the high similarity between some distinct characters also increases the occurrence of false positives. The task still becomes more challenging, as it is usually specific to each domain and application. ***In this way, the techniques established and used for the recognition of a certain family of characters, such as Latin/Roman, can not be applied in the same way for characters of different origin, such as Indus, Chinese or Arab.***

Ex. 2011, Lacerda, E.B., de Souza, T. V., Zanchettin. C., Rabelo, J. C., & Coutinho, L.D. (2017). *Handwriting Recognition: Overview, Challenges and Future Trends*. In Bezerra, B.L., Zanchettin, C., Toselli, A.H., & Pirlo, G.

(Eds.), *Handwriting: Recognition, Development and Analysis* (pp.3-32). Nova Science Publisher.

Similarly, as explained in *Multi-Language Online Handwriting Recognition*:

There is strong variability in writing style between different groups of people and between individuals, for example preference for printed style versus cursive writing or the age and location where writing was taught for the same language. There is strong variability even for the same person, depending on the input speed, sloppiness, or context in which the text is input, e.g., while walking and on a mobile phone. Many ambiguities are inherent in the input, e.g., a small stroke could indicate an apostrophe or could just be accidental 'noise'. Also, many characters are virtually indistinguishable without context. e.g., a single round stroke could be any of the Unicode characters shown in Fig. 2. Similar ambiguities exist for other basic shapes such as horizontal or vertical strokes. ***For scripts with thousands of characters, machine learning is still a challenge, as is the gathering of proper training and evaluation data and the coverage of all common input symbols.*** Handwriting input often has a non-monotonic relationship to the output which is caused by delayed strokes (e.g., t-strokes and i-dots in English) or corrections that are entered later (e.g., a forgotten letter is inserted into a word.)

Ex. 2012, Keyzers, D., Deselaers, T., Rowley, H., Wang, L., & Carbone, V. (2017, June). *Multi-Language Online Handwriting Recognition*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), 1180- 1194.

Moreover, as explained in the declaration of Mr. Jonas Andersson, one of the named inventors of the '428 patent, and a software engineer who has

worked in the handwriting recognition field for more than two decades, handwriting recognition systems “are typically divided into two separate parts,” the lower-level “single character recognition” (“SCR”) part, which “is the process for solving the initial character recognition problem,” and then the higher-level “multiple character recognition” (“MCR”) part, which “attempts to recognize a sequence of strokes that represents one or more characters.” Ex.2009 (Andersson Decl.), ¶¶4-5.

As Mr. Andersson explains, “both the lower-level SCR, as well as the higher-level MCR processes differ substantially depending on the target language being analyzed.” *Id.*, ¶7. Thus, “[w]hile Cerence and its predecessors have handwriting recognition algorithms for the Alphabetic languages (i.e., for recognition of letters in scripts like Latin, to be used for example for the English language) as well as handwriting recognition algorithms for CJK languages (Chinese, Japanese, Korean),” “Cerence utilizes different engineering teams for each, with each developing different handwriting recognition technologies.” *Id.*

Indeed, Mr. Andersson provides a specific example of how language models may be different between Alphabetic type languages (like English) and CJK type languages (Chinese, Japanese, Korean), explaining that:

For example, the SCR process, where an input stroke sequence is matched against character templates in a database, requires

much more strokes to be matched for CJK languages, because there are a lot more characters in the CJK languages than in the English language and each CJK character contains on average a lot more strokes than an English letter. This means that each SCR operation takes a lot more time for the CJK languages than for English.

Additionally, because there are so many strokes in each CJK character, there are a lot more possible segmentations in each CJK writing than in English writing. So, in total, an MCR technology that is based on the robust segmentation by recognition strategy would have to do many more and much heavier SCR operations to solve the segmentation problem, i.e. the MCR operation would be much slower for each CJK language than for English language.

Id., ¶¶8-9.

Due to the added complexities of a higher-level MCR process for CJK languages, Mr. Andersson explains that “the engineering team working on the CJK handwriting recognition technologies did not fully develop a higher-level MCR process for CJK languages until more than a decade after a higher-level MCR process was developed for Alphabetic type languages, like English.”

Id., ¶10.

With respect to challenges that occur at the lower-level SCR stage, Mr. Andersson further explains that “there can be variances in handwriting style, multiple possible recognition results for a particular input shape, and misidentifications of shapes due to certain characters having similar stroke patterns.”

Id., ¶11. And Mr. Andersson confirms that “[t]hese challenges are exacerbated if attempting to combine multiple language models, which necessarily increases the potential recognition results, as well as the potential for misidentification of characters.” *Id.*

With respect to challenges that occur at the higher-level MCR stage, Mr. Andersson further explains that “one problem to be solved in the segmentation problem, i.e., identifying when one character ends and another begins.” *Id.*, ¶12. Moreover, “the content, order, and shape of characters entered by the user may vary significantly even for a single language, depending on the user’s education, nationality, handwriting style, and the context of the input being received.” *Id.* Thus, “MCR models can vary significantly depending on the application, even for a single language.” *Id.*

Moreover, “as with the SCR stage, these challenges in the MCR stage are exacerbated if attempting to combine multiple language models.” *Id.*, ¶13. For example, “different language models would utilize different dictionary or database lookups, different contextual analysis, different formatting assumptions, and different processes for determining whether one recognized character is associated with another.” *Id.*

In short, “handwriting recognition models differ substantially depending on the target language being recognized, including differences in both the

lower-level SCR stage (i.e., the initial character recognition stage) as well as the higher-level MCR stage (i.e., the segmentation post-processing stage).” *Id.*, ¶15. These differences “make[] combining recognition models for different languages challenging.” *Id.* Moreover, “[d]ue to these differences, Cerence and its predecessors needed to use and in fact used different engineering teams in different countries across the world to develop different handwriting recognition.” *Id.*

Mr. Andersson also confirmed that, based on his experience, “these challenges existed as of October 15, 2012,” the priority date of the ’428 patent. *Id.*, ¶14.

Indeed, Petitioners *own* expert, Mr. Ward, testified *exhaustively* during his deposition regarding the challenges faced in performing individual character recognition and post-processing, as well as the additional challenges that may be present for different languages. For example, Mr. Ward testified that “recognizing handwritten input like recognizing speech input is a complex and difficult problem.” Ex. 2010 (Ward Dep Tr.), 28:24-29:1; *id.*, 29:19 (“It’s a complex problem.”). Indeed, to solve this problem, Mr. Ward testified that “you need to use everything you can to try and come up with a successful solution,” including using “context information which might be incorporated in the recognition itself,” as well as “application-specific information,” and

“post processing for things done after some initial character recognition.” *Id.*, 28:1-10.

Mr. Ward further testified that post-processing in recognition algorithms “can be many different forms,” such as “comparing against a list of acceptable inputs or patterns for acceptable inputs,” or “going back and revisiting the recognition results based on further information that was not available at the time of the input.” *Id.*, 29:11-18.

Mr. Ward further testified about the challenges that may occur when attempting to perform character recognition and post-processing on even a *single* language. For example, Mr. Ward testified that: 1) “there could be more than one possible recognition result for the particular input shape or particular user input that can vary depending on handwriting style or handwriting variability” (*id.*, 32:1-6); 2) recognized characters may be confused because they have nearly identical shapes, but are intended to be different characters depending on the context, “such as the exact same symbol may represent a 1 or an I or a lower case L” (*id.*, 32:7-12); and 3) “[u]nusually spelled proper names or incomplete words make it hard to perform simple dictionary or database checks” (*id.*, 33:1-7). Mr. Ward further confirmed that these types of problems would exist as of the priority date of the ’428 patent, as these are problems that would “need to be dealt with in general.” *Id.*, 34:2-6.

Indeed, when asked what additional challenges a POSITA would face when attempting to do handwriting, character, or word recognition for multiple languages, Mr. Ward testified that algorithms may differ *even for the same language*, due to “different writing styles that depend on where someone went to school or when someone went to school,” as well as “dialect differences or national differences in what words are or how certain things are written such as dates.” *Id.*, 34:8-19; *id.*, 35:24-36:2 (“*you could have very different algorithms applied even to the same language or different combinations of algorithms applied to the same language.*”). Mr. Ward further testified that there may be differences in “various abbreviations,” and that “formatting differences in punctuation” “might be different in different countries or in different places even within the same language.” *Id.*, 34:20-24.

With respect to the challenges of combining multiple different language models, Mr. Ward testified that “[i]f you’re dealing with different languages, you may need to deal with additional characters or different characters or characters written in a different form.” *Id.*, 34:1-35:3. Indeed, Mr. Ward testified that:

Depending on the language, some languages, the way writing is taught is more strict about stroke order and stroke direction than in the educational systems commonly used in other countries or other languages. If you're dealing with multiple languages, you may need to deal with multiple languages or multiple character sets at the same time which might even have very similar input

or handwriting forms or characters that can be in one language or another language.

Id., 35:3-13.

As Mr. Ward explained, “[y]ou may have a language that incorporates multiple algorithms even for what you might consider the same steps such as recognizing shapes or user input.” *Id.*, 36:10-13. Mr. Ward further confirmed that even the process of a basic dictionary lookup may differ between different languages, explaining that:

If you had a word list or dictionaries, those may be organized in different fashions in different languages. The alphabetization or dictionary sorting order of a language which nominally uses the same character set or a similar character set or another could be different. Dictionaries can be arranged around strokes in the characters rather than letters.

Id., 35:14-21. Mr. Ward further explained that “[i]f you're dealing with multiple languages, there are also things that tend to be associated with a particular language, not just a particular education for diacritical marks like umlauts in German, accents in French.” *Id.*, 51:24-52:3.

Mr. Ward even specifically testified about challenges that may occur when attempting to design an algorithm which could process both English and Japanese characters, explaining that “you could have the same handwritten input that would ... just as plausibly [be] a Roman – we’ll say Roman letter character or a Katakana character.” *Id.*, 50:11-21. And further that there may

be “differences in the behavior of people writing the connected strokes,” including “which stroke order or stroke direction was more consistent in Katakana and also Kanji characters versus what we experienced in actual use in the U.S. with Roman alphabet characters.” *Id.*, 50:1-51:5. As Mr. Ward testified, “some of the problems would be that depending on the language ... you may have wider set of shapes including a number of just ambiguous inputs” noting that “Katakana had something like 44 characters and was a syllabary.” *Id.*, 51:10-14.

Mr. Ward further testified about differences in the post-processing portions of any recognition algorithm used between different languages such as English and Japanese, explaining that “you would probably have a different dictionary or different set of context phrases used for account Katakana and Japanese and for English” (*id.*, 52:20-53:10) and “there may be other context differences as well” (*id.*, 53:17-18).

Mr. Ward further testified that different language recognition models may utilize system memory differently, as such that “ROM space or table space” would be different. *Id.*, 57:9-15.

Given all of the challenges discussed above, including those identified by both Mr. Andersson, and Petitioners’ own expert Mr. Ward, a POSITA would not have been motivated to combine *Arai* (a Japanese patent application

directed to a handwriting recognition algorithm for the Japanese language), with *Fenwick* (a spacing correction algorithm for the English language). Ex. 2007 (Koskinen Decl.), ¶157. Nor would a POSITA have had a reasonable expectation of success in attempting to combine these very different methods, particularly in view of the significant differences between these types of algorithms discussed above. *Id.*

Importantly, neither Petitioners, nor their expert Mr. Ward, address these challenges, particularly the challenges in attempting to combine a Japanese handwriting recognition and post-processing algorithm with an English spacing correction algorithm. *Id.*, ¶158. Nor do Petitioners or their expert Mr. Ward explain how or why a POSITA would have a reasonable expectation of success in doing so, particularly in view of the challenges discussed above. *Id.*

Instead, Petitioners and their expert Mr. Ward merely conclude (without support), that combining *Arai* and *Fenwick* would be obvious as simply “applying a known technique” (i.e. *Fenwick*) to “a known method” (i.e., *Arai*), “to yield predictable results.” Petition at 32 (citing Ward Decl., ¶140). But Petitioners’ assertions are contradicted by the record evidence, including Mr. Ward’s *own* deposition testimony; and Mr. Ward’s conclusory assertions regarding motivation to combine and any reasonable expectations of success are

entitled to **no** weight. *See In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016) (“To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements.”).

Third, Petitioners’ arguments regarding motivation to combine lack merit. Ex. 2007 (Koskinen Decl.), ¶160. Specifically, Petitioners assert that “[a] POSITA would have been motivated to combine Arai and Fenwick, because Arai seeks to solve the problem of a user being ‘forced to perform other input operations in addition to the actual intended character input, adversely affecting a natural use experience.’” Petition at 31 (citing *Arai* ¶[0004]). As Petitioners explain, “[a] POSITA would have been motivated to incorporated Fenwick’s teachings regarding automatically sending recognized words to an application into Arai because this would have furthered Arai’s purpose of having a natural, uninterrupted flow in entering characters without having to press extra keys.” *Id.*, (citing Ward Decl., ¶139).

But Petitioners’ argument that automatically sending words to an application would further *Arai*’s purported goal of improving natural use experience is **irrelevant** to limitation [g], which relates to automatically deleting the associated memory buffer, **not** automatically sending character strings to an application. Ex. 2007 (Koskinen Decl.), ¶161. Indeed, the requirement of automatically sending sequence portions to an application is found in the next

claim limitation (i.e. limitation 1[h]). *Id.* Importantly, neither Petitioners, nor their expert Mr. Ward explain why a POSITA would be motivated to combine *Arai*'s recognition algorithm with *Fenwick*'s feature of automatically deleting memory buffers. *Id.*

As Mr. Koskinen explains, automatically flushing the associated memory buffer is not a prerequisite to automatically sending a sequence portion to an application, as an algorithm could choose to keep identified sequences in memory for additional contextual analysis, or simply flush the memory buffer based on some other criteria, such as an automatic timeout, after a certain number of characters have been entered, or based on some other contextual information. *Id.*, ¶162.

In short, as discussed above, a POSITA would not have been motivated to combine *Arai* and *Fenwick* to achieve the '428 patent's method, and would not have had a reasonable expectation of success in doing so. *Id.*, ¶163. Thus, once again, *Arai* in view of *Fenwick* fails to disclose or suggest limitation [g].

b. Claim 2

Claim 2 depends on claim 1, and thus Petitioners have failed to demonstrate that *Arai* in view of *Fenwick* discloses or renders obvious the limitations of this claim for the same reasons discussed with respect to claim 1. Ex. 2007 (Koskinen Decl.), ¶164.

c. Claim 3

Claim 3 depends on claim 1, and thus Petitioners have failed to demonstrate that *Arai* in view of *Fenwick* discloses or renders obvious the limitations of this claim for the same reasons discussed with respect to claim 1. Ex. 2007 (Koskinen Decl.), ¶165.

Moreover, *Arai* in view of *Fenwick* further neither discloses nor renders obvious the additional limitations of claim 3. *Id.*, ¶166. Specifically, claim 3 requires “The method of claim 1, further comprising: determining at least a second candidate sequence based on the first user input; and presenting at least the second candidate sequence to the user.” ’428 patent, cl. 3. However, neither *Arai*, nor *Fenwick*, discloses presenting any “second candidate sequence” to the user. Ex. 2007 (Koskinen Decl.), ¶166.

While *Arai* discloses calculating the chain probabilities of character sequences, and even gives an example of multiple different candidate combinations being generated (*see, e.g., Arai* ¶[0146]), *Arai*’s specification explains that only the “the candidate character string with the highest chain probability” is displayed to the user for editing (*see Arai* ¶¶[0146], [0152]). Thus, *Arai* does not disclose presenting any “second candidate sequence” to the user. Ex. 2007 (Koskinen Decl.), ¶168. Nor does *Fenwick* disclose any disclosure of multiple candidate sequences to the user.

Moreover, Petitioners' arguments for why *Arai* purportedly discloses this limitation lack merit. *Id.*, ¶168. Specifically, as with claim 1, Petitioners cite to the **wrong** disclosure in *Arai*, once again attempting to pass off *Arai*'s initial character recognition process as somehow generating "candidate sequences." *Id.*; Petition at 36-37. For example, Petitioners cite to an example in *Arai* where initial evaluation values are calculated for the **individual** character entered '工', and then note that *Arai* discloses allowing a user to select individual characters displayed in the "character buffer area 61," which allows "a group of the first through fourth candidates for that character" to be displayed. *Id.*, 36 (citing *Arai* ¶[0086]).

But as evident from the express language Petitioners cite, this is not a list of hypothetical candidate character sequences (i.e., potential words or strings of characters the user may be trying to enter), this is simply a display of the rank-ordered recognition results for the **individual character** '工' which were determined during the initial individual character recognition step in *Arai*. Ex. 2007 (Koskinen Decl.), ¶169; *Arai* ¶[0086] ("a group of the first through fourth candidates **for that character** will be displayed").

Because *Arai* does not disclose or suggest presenting at least a second candidate sequence to the user, and Petitioners do not point to any disclosure in *Fenwick* for this additional limitation, *Arai* in view of *Fenwick* does not

disclose or render obvious the additional limitations of claim 3. Ex. 2007 (Koskinen Decl.), ¶170.

d. Claim 7

Claim 7 depends on claim 1, and thus Petitioners have failed to demonstrate that *Arai* in view of *Fenwick* discloses or renders obvious the limitations of this claim for the same reasons discussed with respect to claim 1. Ex. 2007 (Koskinen Decl.), ¶171.

e. Claim 8

Claim 8 depends on claim 1, and thus Petitioners have failed to demonstrate that *Arai* in view of *Fenwick* discloses or renders obvious the limitations of this claim for the same reasons discussed with respect to claim 1. Ex. 2007 (Koskinen Decl.), ¶172.

f. Claim 9

Arai in view of *Fenwick* does not disclose or render obvious the limitations of claim 9. Ex. 2007 (Koskinen Decl.), ¶173. Claim 9 is reproduced below:

9[pre]: At least one non-transitory computer-readable storage medium storing instructions for data input, wherein the data input is performed at a wireless device having a processor and a memory, comprising:

9[a]: receiving a first user input and at least a second user input;

9[b]: wherein the first and second user input are handwritten input received on a handwriting input area of the wireless device;

9[c]: determining a first sequence based on a best match of the first and second user inputs, wherein the first sequence includes at least the first input;

9[d]: storing the first sequence to a memory of the mobile device;

9[e]: automatically determining that at least a portion of the sequence matches a word or phrase in a database of acceptable words and/or phrases without the user inputting a space or selecting a word; and

9[f]: sending the first sequence to an application or service running on, or accessible by, the wireless device, wherein the sending comprises removing the first sequence from the memory storage buffer, and wherein the sending is performed in the absence of user editing of the first sequence.

As shown above, independent claim 9 of the '428 patent is the system version of the '428 patent's method claim 1, and thus *Arai* in view of *Fenwick* does not disclose or render obvious claim 9 for similar reasons as discussed above with respect to claim 1.

For example, with respect to limitation 9[c], “determining a first sequence based on a best match of the first and second user inputs, wherein the first sequence includes at least the first input,” this limitation parallels limitation 1[d] of claim 1 (i.e., “generating a first candidate sequence, wherein the

first candidate sequence is a best match of one or more recognized sequences”). Ex. 2007 (Koskinen Decl.), ¶175. And as Petitioners did with limitation 1[d] of claim 1, Petitioners once again mischaracterize *Arai*’s disclosure to identify the **wrong** disclosure in *Arai* for limitation 9[c] of claim 9, once again pointing to *Arai*’s initial character recognition process for this limitation, rather than appropriately pointing to *Arai*’s post-processing unit’s later calculation of the chain probabilities for the entered character string. See Petition at 41-42; Ex. 2007 (Koskinen Decl.), ¶175.

With respect to limitation 9[e], which requires “automatically determining that at least a portion of the sequence matches a word or phrase in a database of acceptable words and/or phrases without the user inputting a space or selecting a word,” this limitation parallels limitation 1[f] of claim 1 (i.e., “identifying at least a first sequence portion of the first candidate sequence wherein the identifying is at least partially based on one or more rules or dictionaries of words accessible to the mobile device.”). Ex. 2007 (Koskinen Decl.), ¶176. And as Petitioners did with limitation 1[f] of claim 1, Petitioners once again mischaracterize the disclosure in *Arai* to point to *Arai*’s sole post-processing step of calculating the chain probabilities as satisfying this limitation. Petition at 42-45. But, as with limitation 1[f], Petitioners fail to point to

any disclosure in *Arai* of determining whether any “portion” of an entered character sequence matches anything. Ex. 2007 (Koskinen Decl.), ¶176.

Specifically, with respect to limitation 9[e], Petitioners assert that “[b]ecause *Arai* continues post-processing while the user continues handwritten character input, a POSITA would have understood that *Arai* determines that at least a portion of the sequence matches a word or phrase without the user entering a space or selecting a word.” Petition at 45 (citing Ward Decl., ¶181). But once again, Petitioners are wrong. *Arai* clearly discloses only determining whether a ***full character string*** appears in the dictionary lookup, not whether any sub-portions of any character string matches any database or dictionary. Ex. 2007 (Koskinen Decl.), ¶177; *see also Arai* ¶[0146] (showing example outputs of *Arai*’s post-processing unit 34). Thus, as with limitation 1[f] of claim 1, *Arai* fails to disclose or suggest limitation 9[e] of claim 9. Ex. 2007 (Koskinen Decl.), ¶177.

Additionally, *Arai* fails to disclose or suggest automatically determining that at least a portion of the sequence matches a word or phrase in a “database of acceptable words and/or phrases.” Indeed, Petitioners recognize that *Arai* fails to disclose referencing any “database of acceptable words and/or phrases,” but nevertheless assert that “[a] POSITA would have understood” *Arai*’s disclosure of “dictionary data” to encompass a “database of acceptable

words and/or phrases.” Petition at 44. Not so. Here, claim 1 of the ’428 patent uses the express term “dictionary” to define its claim scope, while claim 9 uses “a database of acceptable words and/or phrases.” To the extent that the applicants intended for these claims to have the same scope during prosecution of the ’428 patent, they could have used the same terminology for both claims 1 and 9. They did not. Under the doctrine of claim differentiation, the term “dictionary” of claim 1 is not equivalent to the term “a database of acceptable words and/or phrases” of claim 9. Indeed, the fact that Petitioners assert a *separate* grounds of invalidity for claim 9 just to add another prior art reference (*Baker*) for the disclosure of “a database of acceptable words and/or phrases” confirms that Petitioners are well aware that *Arai* fails to disclose or suggest this limitation.

Finally, with respect to limitation 9[f], which requires “sending the first sequence to an application or service running on, or accessible by, the wireless device, wherein the sending comprises removing the first sequence from the memory storage buffer, and wherein the sending is performed in the absence of user editing of the first sequence,” this limitation parallels limitations 1[g] and 1[h] of claim 1 (i.e., “removing the first sequence portion from the memory storage buffer, wherein the removing is performed in the absence of receiving user input editing the first candidate sequence, representing a space

character, or selecting a word” and “sending the first sequence portion to an application running on the mobile device.”). Ex. 2007 (Koskinen Decl.), ¶178. And as Petitioners did with limitations 1[g] and 1[h] of claim 1, Petitioners rely on the combination of *Arai* and *Fenwick* to disclose or suggest limitation 9[f]. Petition at 45-47. But as with limitations 1[g] and 1[h] of claim 1 discussed above, *Arai* and *Fenwick* fail to disclose or suggest limitation 9[f]. Ex. 2007 (Koskinen Decl.), ¶178. Moreover, as discussed above, a POSITA would not have been motivated to combine *Arai* and *Fenwick*, and would not have had a reasonable expectation of success in doing so. *Id.*

g. Claim 12

Claim 12 depends on claim 9, and thus Petitioners have failed to demonstrate that *Arai* in view of *Fenwick* discloses or renders obvious the limitations of this claim for the same reasons discussed with respect to claim 9. Ex. 2007 (Koskinen Decl.), ¶180.

h. Claim 13

Claim 13 depends on claim 9, and thus Petitioners have failed to demonstrate that *Arai* in view of *Fenwick* discloses or renders obvious the limitations of this claim for the same reasons discussed with respect to claim 9. Ex. 2007 (Koskinen Decl.), ¶181.

B. Ground 2: Petitioners Fail To Show That Claims 9 and 12-13 Would Have Been Obvious Based on Arai In View Of Fenwick, And In Further View Of Baker

In their second ground for invalidity, Petitioners assert that *Arai* in combination with *Fenwick* and *Baker* renders obvious claims 9 and 12-13 of the '428 patent. Petition at 48- 51. As Petitioners explain, their second ground for invalidity is identical to their arguments for claims 9 and 12-13 discussed above for Petitioners first ground for invalidity, except that Petitioners' second ground further adds *Baker* for the disclosure of the limitation “a database of acceptable words and/or phrases” found in claim 9. *Id.*

Because Petitioners do not rely on the addition of *Baker* to disclose or suggest any other limitations beyond “a database of acceptable words and/or phrases,” Petitioners have not shown that *Arai* in combination with *Fenwick* and *Baker* renders obvious claims 9 and 12-13 of the '428 patent for the same reasons discussed above with respect to Petitioners' first invalidity ground. Ex. 2007 (Koskinen Decl.), ¶183.

Additionally, as with the combination of *Arai* and *Fenwick*, a POSITA would not have been motivated to combine the Japanese patent application *Arai* (disclosing a Japanese handwriting recognition algorithm) with a European patent specification *Baker* (disclosing word recognition algorithms for Latin or Cyrillic-based words), because both the initial recognition process, as

well as the post-processing that occurs on any recognized characters, differs substantially between different languages, and even differs substantially between different dialects between the same language. Ex. 2007 (Koskinen Decl.), ¶184; *see also* Ex. 2009 (Andersson Decl.), ¶¶3-15; Ex. 2005 (Ward Dep Tr.), 28:1-10, 28:24-29:1, 29:11-19, 32:1-12, 33:1-7, 34:2-6, 34:8-24, 35:3-21, 35:24-36:2, 36:10-13, 50:1-51:14, 51:24-52:3, 52:20-53:10, 53:17-18, and 57:9-15.

Once again, neither Petitioners, nor their expert Mr. Ward, address these challenges, particularly the challenges in attempting to combine a Japanese handwriting recognition and post-processing algorithm with an algorithm for Latin or Cyrillic-based words. Ex. 2007 (Koskinen Decl.), ¶185. Nor do Petitioners or their expert Mr. Ward explain how or why a POSITA would have a reasonable expectation of success in doing so, particularly in view of the challenges discussed above. *Id.*

Instead, once again, Petitioners and their expert Mr. Ward merely conclude (without support), that combining *Arai* and *Baker* would be as obvious as simply “applying a known technique” (i.e. *Baker*) to “a known device” (i.e., *Arai-Fenwick*), “to yield predictable results.” Petition at 50 (citing Ward Decl., ¶197). But as discussed above, Petitioners assertions are contradicted by the record evidence, including Mr. Ward’s *own* deposition testimony; and

Mr. Ward’s conclusory assertions regarding motivation to combine and any reasonable expectations of success are entitled to ***no*** weight. *See In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d at 1380 (“To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements.”).

IX. CONCLUSION

For each of the above reasons, the patentability of all challenged claims should be upheld.

Respectfully submitted,

Dated: April 25, 2025

/ Qi (Peter) Tong /

Qi (Peter) Tong (Reg. No. 74,292)
RUSS AUGUST & KABAT
8080 N. Central Expy, Suite 1503
Dallas, TX 75206
(310) 826-7474

Attorney for Patent Owner

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. §42.24(d), Patent Owner hereby certifies, in accordance with and reliance on the word count provided by the word-processing system used to prepare this **PATENT OWNER'S RESPONSE**, that the number of words in this paper is 13,694 Pursuant to 37 C.F.R. § 42.24, this word count is in compliance with the word limit set forth in 37 C.F.R. § 42.24(b)(2) excluding the portions exempted under 37 C.F.R. § 42.24(a)(1).

/Qi Tong/
Qi (Peter) Tong (Reg. No. 74,292)

CERTIFICATE OF SERVICE

The undersigned hereby certifies that the above document was served on April 25, 2025, by filing this document through the Patent Trial and Appeal Case Tracking System as well as delivering a copy via electronic mail upon the following attorneys of record for the Petitioner:

Ali R. Sharifahmadian
Patrick Reidy
Douglas L. Clark
Albert J. Boardman
ARNOLD & PORTER KAYE SCHOLER LLP
601 Massachusetts Ave., NW
Washington, DC 20001-3743
Telephone: 202-942-5000
Email: ali.sharifahmadian@arnoldporter.com
Email: patrick.reidy@arnoldporter.com
Email: douglas.clark@arnoldporter.com
Email: albert.boardman@arnoldporter.com
Email: xSamsungCerenceAP@arnoldporter.com

Dated: April 25, 2025

/ Qi (Peter) Tong /

Qi (Peter) Tong (Reg. No. 74,292)
RUSS AUGUST & KABAT
8080 N. Central Expy, Suite 1503
Dallas, TX 75206
(310) 826-7474

Attorney for Patent Owner