

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Google LLC,
Petitioner,

v.

Withrow Networks Inc.,
Patent Owner.

IPR2025-00775

U.S. Patent No. 10,771,849 B2

PETITIONER'S DEMONSTRATIVE SLIDES

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Illustrative Claim – Claim 1 of the '849 Patent

1. A method comprising:

receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

utilizing optimized decoding processes to maintain quality playback.

Grounds 1-5

Obviousness Grounds

Ground	Claims	Basis for Challenge Under § 103(a)
1	1-9	Carmel in view of Mattavelli
2	1-9	Carmel in view of Wang
3	1-9	Carmel in view of Wang and Wolters
4	1-9	Carmel in view of Wang, Wolters, Guo, Yu, and Forecast
5	1-9	Carmel in view of Mattavelli, Guo, Yu, and Forecast

Cooley

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Summary of the Issues

Ground 1 Issues:

- | | |
|---|--|
| 1 | Carmel & Carmel + Mattavelli teach “autonomously adjusting said selection and playback” |
| 2 | Carmel + Mattavelli teaches maintaining fluidity |
| 3 | Carmel + Mattavelli teaches “utilizing optimized decoding processes to maintain quality playback” and “fluidity, video quality and audio quality are maintained” |

Ground 2 Issues:

- | | |
|---|---|
| 1 | Carmel + Wang teaches “optimized decoding processes to maintain quality playback” and maintaining “fluidity, video quality and audio quality” |
| 2 | A POSA would have been motivated to combine Carmel and Wang’s teachings |

Additional Issues:

- | | |
|---|--|
| 1 | Wolters teaches optimized audio decoding processes for maintaining quality playback, including fluidity, video quality, and audio quality |
| 2 | Grounds 4-5 address limitations that PO does not dispute; and the Petition explains why a POSA would have been motivated to combine the references |
| 3 | Petitioner’s claim construction positions before the Board and the district court are consistent and justified; PO’s positions are not |
| 4 | PO fails to show that the “maintain quality playback” term is not a statement of intended use |

Relevant Claim Language

1. A method comprising:

receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained¹ by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters² and supplied host path identification, and by

utilizing optimized decoding processes to maintain quality playback.

6. A multimedia player comprising:

computer executable instructions stored on a non-transitory computer readable medium which when executed by a processor:

requests transmission of multimedia objects;

receives audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated object parameters and supplied host and path identification to form multimedia objects; and wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

plays back received multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained¹ by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and² by utilizing optimized decoding processes to maintain quality playback.

7. A system comprising

at least one centralized content server transmitting multimedia objects; said multimedia objects including segmented audio and video, and parameters and supplied host path identification;

at least one multimedia player configured to:

request transmission of multimedia objects;

receive audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated object parameters and supplied host and path identification to form multimedia objects; and wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

play back received multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained¹ by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and² by utilizing optimized decoding processes to maintain quality playback.

'849 patent, cls. 1, 6 & 7

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 33-50, 65-67;
Polish Decl. (Ex. 1002) at ¶¶256-264 5

Ground 1: Carmel + Mattavelli

Cooley

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Carmel (Ex. 1003)

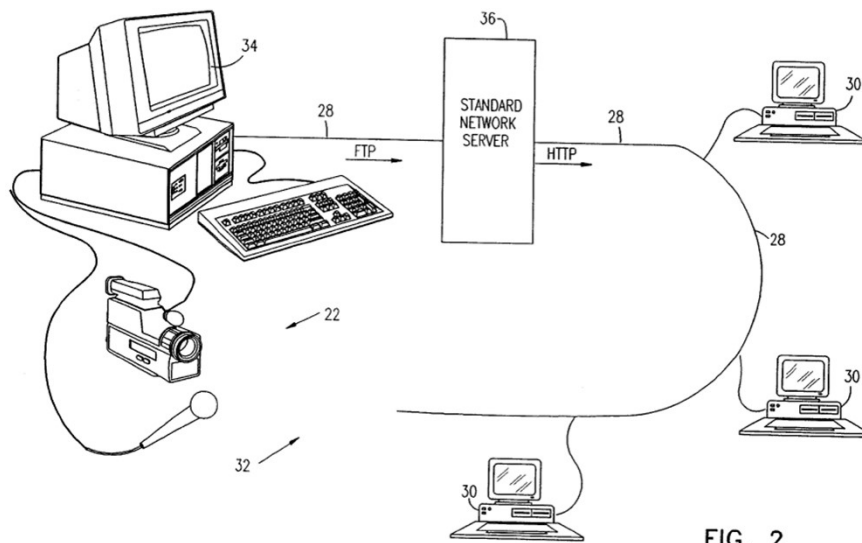
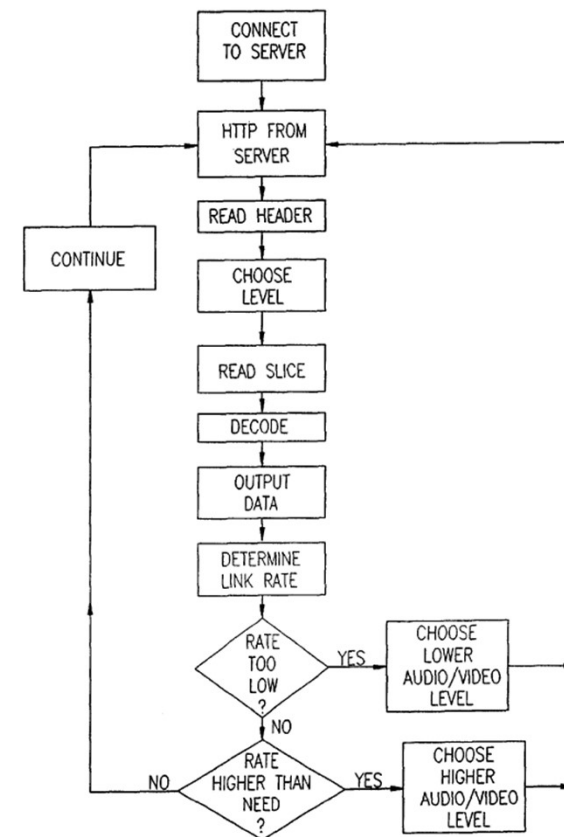


FIG. 2

FIG. 6B



Mattavelli (Ex. 1004)

COMPUTATIONAL GRACEFUL DEGRADATION FOR VIDEO DECODING

Marco Mattavelli, Fabian Brunton
Integrated Systems Center - CSI,
Swiss Federal Institute of Technology,
Lausanne CH-1015, SWITZERLAND
Tel: +41 21 6936079; fax: +41 21 6936063
e-mail: Marco.Mattavelli@epfl.ch

ABSTRACT

The implementation of software based video decoders is an advantageous solution over traditional dedicated hardware real-time systems. However, guaranteeing real-time performance, needed when processing video/audio bit-streams, by scheduling processing tasks of variable load and that might exceed the available resources, is an extremely difficult and in many cases an impossible challenge. In this paper we introduce the motivations and ideas of the Computational Graceful Degradation (CGD) approach for the implementation of video decoding. The goal is to provide results of possibly lower visual quality, but respecting the real-time constraints, when the required processing needs exceed the available processing resources. This approach is very attractive since it enables a more efficient usage of the hardware processing power without needing to implement resources for worst case complexity decoding. We show how such technique can be applied to the decoding of compressed video sequences. Simulation results for MPEG-4 and H.263 video compression standards showing some of the interesting achievements and potentialities of CGD approach are also reported. The presented approach can be virtually applied to any video compression standard and processor based platform, although the best performances are achieved when complexity prediction information provided as side information by the video standard is available. This option has been included in the new MPEG-4 standard.

1 INTRODUCTION

The complexity of video decoding is a fluctuating function of time [1]. By complexity we mean the processing time necessary to decode a compressed video bit-stream. Even considering a specific processor platform, a specific compression standard, with a relative level and profile, the complexity of video decoding is a variable quantity. Dedicated hardware platforms can guarantee worst case complexity decoding. In contrast, in the case of programmable platforms such a guarantee would imply the allocation of a processing power up to

two orders of magnitude higher than the one actually needed for an average decoding [2]. Conversely, a theoretical average value is not useful for an efficient scheduling of a decoding task. The average value for such considered sequence is not stationary, and its variation around the theoretical average can be so unpredictable that no useful scheduling policy can be devised. The first straightforward option is a constrained over-booking of the processing resources. The second is to accept interruptions in the audio/video output streams when the needed decoding resources exceed the ones available in the system. Obviously both options are not desirable being respectively over-estimated, or providing a quality of the services which may result severely perceptually unacceptable. The main rationale of the Computational Graceful Degradation (CGD) approach is to provide results of poorer quality when the results of the desirable quality cannot be obtained in real time. When the complexity of the task exceeds the available processing power, real-time services, possibly of degraded quality, are nevertheless guaranteed respecting all real-time constraints. This approach enables the correct scheduling of real-time tasks even when the computational system load fluctuates above the system capacity limits.

The concept of computational graceful system degradation using imperative results has previously been introduced in the field of real-time computing with applications to control system and communication. In these contexts imperative results are defined as intermediate results with the property that the more recent and immediate result is the closer one that approximates the final result (i.e. the closer only results to incrementally improving computations) [2, 3, 4]. In these cases the problem to be solved are more concerned with fault tolerance, I/O overheads, stability and resiliency to unexpected inputs, and the corresponding techniques are not suitable for the purposes of video decoding. In our case, the goal is to guarantee real-time video decoding with possible degradations of the image quality, i.e. reduction of the peak signal to noise ratio (PSNR), without losing parts of the image content. Such an

ABSTRACT

The implementation of software based video decoders is an advantageous solution over traditional dedicated hardware real-time systems. However, guaranteeing real-time performance, needed when processing video/audio bit-streams, by scheduling processing tasks of variable load and that might exceed the available resources, is an extremely difficult and in many cases an impossible challenge. In this paper we introduce the motivations and ideas of the Computational Graceful Degradation (CGD) approach for the implementation of video decoding. The goal is to provide results of possibly lower visual quality, but respecting the real-time constraints, when the required processing needs exceed the available processing resources. This approach is very attractive since it enables a more efficient usage of the hardware processing power without needing to implement resources for worst case complexity decoding. We show how such technique can be applied to the decoding of compressed video sequences. Simulation results for MPEG-4 and H.263 video compression standards showing some of the interesting achievements and potentialities of CGD approach are also reported. The presented approach can be virtually applied to any video compression standard and processor based platform, although the best performances are achieved when complexity prediction information provided as side information by the video standard is available. This option has been included in the new MPEG-4 standard.

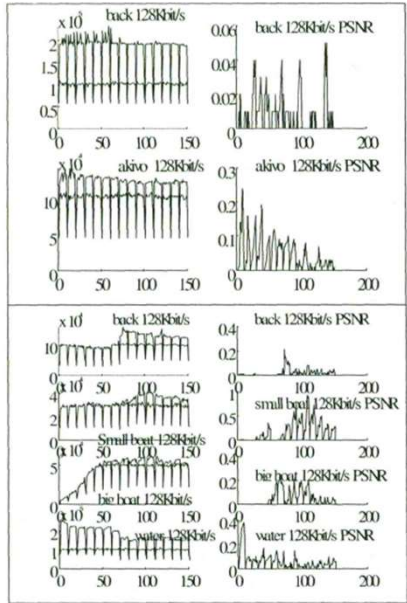


Figure 3: Left, decoding time results for the various VOP of sequences Akiyo and Coastguard, with and without imposing a bound. Right, the PSNR degradations versus the sequence VOP number.

Ground 1 – Issue 1: Carmel & Carmel + Mattavelli teach “autonomously adjusting said selection and playback”

1. A method comprising:

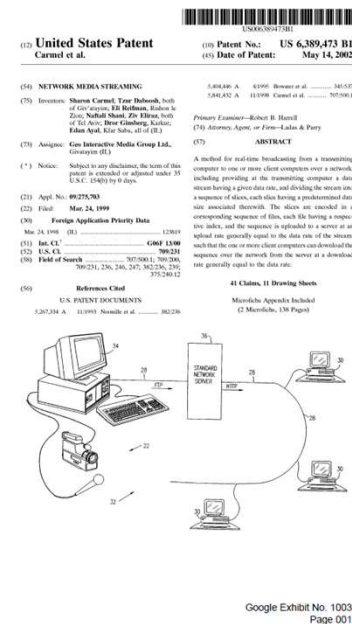
receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, **autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification**, and by

utilizing optimized decoding processes to maintain quality playback.

'849 patent, cl. 1.



COMPUTATIONAL GRACEFUL DEGRADATION FOR VIDEO DECODING

Marc Mattavelli, Sphero Brantton
Integrated Systems Center - CSI,
Swiss Federal Institute of Technology,
Lausanne CH-1015, SWITZERLAND
Tel: +41 21 6930579; fax: +41 21 6931603
e-mail: Marc.Mattavelli@epfl.ch

ABSTRACT

The implementation of software based video decoders in an asynchronous real-time system, where traditional dedicated hardware real-time systems. However, guaranteeing real-time performance, model when processing video/audio bit-streams, by scheduling processing tasks of variable load and that might exceed the available resources, is an extremely difficult and in many cases an impossible challenge. In this paper we introduce the motivation and ideas of the Computational Graceful Degradation (CGD) approach for the implementation of video decoding. The goal is to provide results of possibly lower visual quality, but respecting the real-time constraints, when the required processing needs exceed the available processing resources. This approach is very attractive since it enables a more efficient usage of the hardware processing power without needing to implement resources for worst case complexity decoding. We show how such technique can be applied to the decoding of compressed video sequences. Simulation results for MPEG-4 and H.263 video compression standards, during course of its increasing requirements and potentialities of CGD approach are also reported. The presented approach can be virtually applied to any video compression standard and processor based platform, although the best performance are achieved when complexity prediction information provided as side information to the video standard is available. This option has been included in the new MPEG-4 standard.

1. INTRODUCTION

The complexity of video decoding is a fluctuating function of them [1]. By complexity we mean the processing time necessary to decode a compressed video bit-stream. First considering a specific processing platform, a specific compression standard, with a relative level and profile, the complexity of video decoding is a variable quantity. Dedicated hardware platform can guarantee worst case complexity decoding. In contrast, in the case of programmable platform such a guarantee would imply the allocation of a processing power up to

two orders of magnitude higher than the one actually needed for an average decoding [2]. Conversely, a theoretical average value is not useful for an efficient scheduling of a decoding task. The average value for each considered sequence is not constant, and its variation around the theoretical average one are so unpredictable that no useful scheduling policy can be devised. The first straightforward option is a constant over-estimation of the processing resources. The second is to accept interruptions in the audio/video output streams when the needed decoding resources exceed the ones available in the system. Obviously both options are not desirable being respectively not economical, or providing a quality of the services which may really result progressively unacceptable. The main objective of the Computational Graceful Degradation (CGD) approach is to provide results of good quality when the results of the available quality cannot be obtained in real time. When the complexity of the task exceeds the available processing power, real-time services, possibly of degraded quality, are nevertheless guaranteed respecting all real-time constraints. This approach enables the current scheduling of real-time tasks even when the computational system load fluctuates above the system capabilities.

The concept of computational graceful system degradation using heuristic results has previously been introduced in the field of real-time computing with application to control systems and automation. In these contexts important results are defined as intermediate results with the property that the more recent and immediate result is the closest one that approximates the final result (i.e. the idea only works in non-monotonically improving computation [2, 3, 4]). In these cases the problem to be solved are more concerned with final (desired) LVN measures, stability and sensitivity to unexpected inputs, and the corresponding techniques are more concerned with the final result. In our case, the goal is to guarantee real-time video decoding with possible degradation of image quality (reduction of the peak signal to noise ratio (PSNR), without losing parts of the image content). Such an

1206
Google Exhibit No. 1004
Page 001

PO's half-proposed construction of “adjusting said selection and [independently adjusting] playback” is unsupported

“Carmel requests transmission of next segments with different rates in response to bandwidth limitations but does not discuss **adjusting playback of the already downloaded segments** according to multimedia object parameters and supplied host path identification.”

POR at 1

“The claim language is explicit: the player must adjust two distinct things, ‘selection’ and ‘playback,’ joined by the conjunctive ‘and.’ ... The specification confirms that ‘playback’ carries a meaning **independent** of ‘selection.’”

Sur-Reply at 2-3

- PO half-proposes a construction that incorrectly requires independent adjustments to selection and playback (POR at 1; Sur-reply at 2-3)
- PO also suggests that adjusting playback is something additional that must be done on already-downloaded objects. (POR at 1; Sur-reply at 2-3)

PO's half-proposed construction of “adjusting playback” is wrong and inconsistent with: (i) the claim language, (ii) the '849 written description, and (iii) PO and its expert's positions outside of this IPR

See also Petition at 39, 40-45; Reply at 4-11;
Polish Decl. (Ex. 1002) at ¶¶173-196

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

PO's half-proposed construction of “adjusting said selection and [independently adjusting] playback” is inconsistent with claim language

1. A method comprising:

...

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

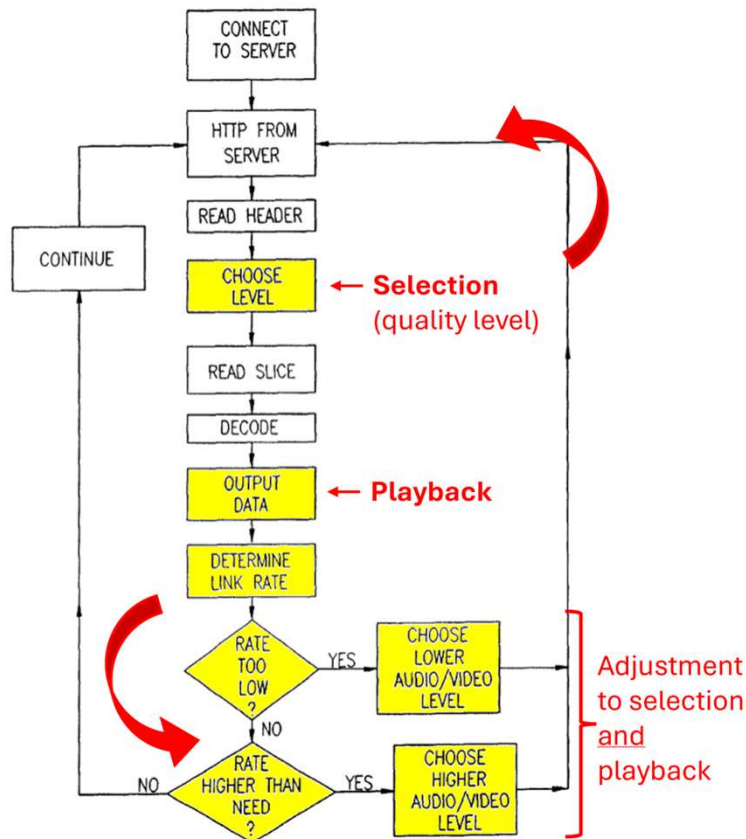
utilizing optimized decoding processes to maintain quality playback.

- “said selection and playback” recites two intertwined concepts:
 - “playing back” of received multimedia objects is tied to their “by selecting” in a manner that reflects available network bandwidth
 - adjustment is done according to (i) object parameters and (ii) supplied host path identification
- No support in claim language for an “already downloaded” or “post-download” construction (POR at 1; Sur-reply at 4)

See also Petition at 35-44; Reply at 4, 10-11;
Polish Decl. (Ex. 1002) at ¶¶188-193, 198

Carmel teaches “autonomously adjusting said selection and playback”

FIG. 6B



Carmel (Ex. 1003), Fig. 6B

FIG. 6B is a flow chart illustrating the operation of clients 30 in downloading and playing back multi-level data stream 41 (FIG. 3D) transmitted from server 36, in accordance with another preferred embodiment of the present invention. As in the method of FIG. 6A, each client 30 connects to the server, generally using a single HTTP link. After reading header 43 and, preferably, making an initial assessment of the link bandwidth, the client selects one of the available quality levels in the stream. Responsive to the selection, server 36 begins to transmit data slices at the chosen quality level. The slices are received, decoded and output by the client.

Periodically, client 30 makes an assessment of the rate of data transfer over the link from the server and, if necessary, changes the quality level accordingly. For example, if the rate is low, such that time stamps 59 indicate that the slices need to be played as fast as or faster than they are being received, the client will preferably select a lower quality level if one is available. On the other hand, if the rate is substantially higher than what is needed to receive the successive slices on time, the client may select a higher quality level to take advantage of the available bandwidth.

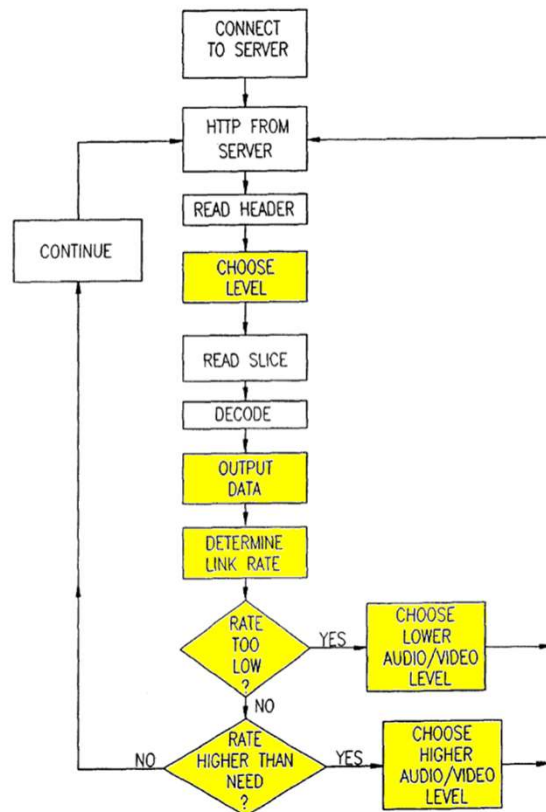
Carmel (Ex. 1003), 10:64-11:18

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 41-44; Reply at 2-4;
Polish Decl. (Ex. 1002) at ¶¶186-195 12

Carmel teaches “autonomously adjusting said selection and playback”

FIG. 6B



Carmel (Ex. 1003), Fig. 6B

In the case of a multimedia stream, client 30 reconstructs and **outputs** the multimedia data **for the appreciation of a user**. Time stamps in the data stream are used to synchronize the data, so that the multimedia sequence is **played back just as it was input at computer 34**, preferably with only a minimal necessary transmission and decoding delay.

Carmel (Ex. 1003), 10:48-54

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 36-37, 42-43; Reply at 2-4, 9-11;
Polish Decl. (Ex. 1002) at ¶¶190-192, 194 13

PO's half-proposed construction of “adjusting said selection and [independently adjusting] playback” is inconsistent with written description

On the client side, the number of multimedia objects that can be buffered in memory is based on the size of the first multimedia object. The amount of free memory available and the processing of further multimedia objects in the sequence can be optional and dependent on whether the implementation allows the modification of object parameters between multimedia objects (such as the size of the visual frame or the sample rate of the audio stream). The buffering and playback of multimedia objects in a synchronized fashion is critical to fluid playback.

'849 patent (Ex. 1001) at 3:36-45

The passage from the '849 specification PO identifies does not support PO's half-proposed construction of “adjusting said selection and [independently adjusting] playback”

1. A method comprising:

...

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

utilizing optimized decoding processes to maintain quality playback.

'849 patent:

After configuring the audio and/or video decoding components of the multimedia object player, the object player may choose to either delay playback until the multimedia object buffers in memory have filled or may begin playback immediately while requesting the next multimedia object concurrently. This decision can be based on the speed at which the multimedia objects are retrieved versus the playback time of each multimedia object, the latency of requests for multimedia objects or the number of multimedia objects that can be stored in memory at once.

...

The retrieval and playback of multimedia objects continues until the last multimedia object in the sequence has been completely retrieved and its playback has finished.

'849 patent (Ex. 1001) at 8:48-9:3
See PO Sur-Reply at 2-3

- No mention of “adjusting” playback – just a decision on whether to *delay* or *begin* playback
- No mention of “selecting” in a manner that reflects available bandwidth

Carmel's teachings materially match what PO identifies from the '849 specification as disclosing "adjusting playback"

Compare

'849 patent:

After configuring the audio and/or video decoding components of the multimedia object player, the object player may choose to either **delay playback** until the multimedia object buffers in memory have filled or may **begin playback** immediately while requesting the next multimedia object concurrently. **This decision can be based on the speed at which the multimedia objects are retrieved versus the playback time of each multimedia object**, the latency of requests for multimedia objects or the number of multimedia objects that can be stored in memory at once.

...

The retrieval and playback of multimedia objects continues until the last multimedia object in the sequence has been completely retrieved and its playback has finished.

'849 patent (Ex. 1001) at 8:48-9:3
See PO Sur-Reply at 2-3

Carmel:

FIG. 6B is a flow chart illustrating the operation of clients **30** in downloading and playing back multi-level data stream **41** (FIG. 3D) transmitted from server **36**, in accordance with another preferred embodiment of the present invention. As in the method of FIG. 6A, each client **30** connects to the server, generally using a single HTTP link. After reading header **43** and, preferably, making an initial assessment of the link bandwidth, the client selects one of the available quality levels in the stream. Responsive to the selection, server **36** begins to transmit data slices at the chosen quality level. The slices are received, decoded and output by the client.

Periodically, client **30** makes an **assessment of the rate of data transfer over the link from the server** and, if necessary, changes the quality level accordingly. **For example, if the rate is low, such that time stamps 59 indicate that the slices need to be played as fast as or faster than they are being received, the client will preferably select a lower quality level if one is available.** On the other hand, if the rate is substantially higher than what is needed to receive the successive slices on time, the client may select a higher quality level to take advantage of the available bandwidth.

Carmel (Ex. 1003), 10:64-11:18

See also Petition at 40-44; Reply at 2-4, 9-11;
Polish Decl. (Ex. 1002) at ¶¶186-198

Carmel teaches “autonomously adjusting said selection and playback”

Patent Owner’s expert, Mr. Goldman, acknowledged the connection taught by Carmel between download rate and the replay rate, and Carmel explicitly connects replay rate to playback.

41. Fourth, the ‘473 Patent ensures *real-time* delivery of multimedia streams by maintaining the upload (Claim 1), download (Claim 1) and replay (Claim 8) rates so they are generally *equal* to the assigned (given) data rate of the stream. For instance, the ‘473 Patent discloses an embodiment whereby the amount of data generated by the transmitting computer over a period of time (e.g., 10 seconds) is uploaded to a server within 10 seconds and is subsequently downloaded within 10 seconds by the client computer. Hence, there is a balance of flow between the production, transmission, and consumption of a real-time multimedia stream, resulting in a stable and scalable operation. The ‘473 Patent provides explicit steps for *rate control*, balancing the given data rate provided at the transmitting computer, the upload rate, download rate, and replay rate, including techniques for adjusting these rates synergistically, through selectable compression choices and other techniques. The invention’s rate control techniques are employed when the rates of the upload and download are slow compared to the data rate(s) of the multimedia stream(s), or when the upload and download rates are faster than the data rates of the multimedia streams. ‘473 Patent, 11:9-22.

Goldman Decl. from Carmel Reexam (Ex. 1081) at 11

8. A method according to claim 2, wherein the one or more client computers decode the sequence and play back the data stream responsive to the indices of the files, at a replay rate generally equal to the data rate.

Carmel (Ex. 1003) at cl. 8

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 39-40, 50-52; Reply at 8-9;
Polish Decl. (Ex. 1002) at ¶¶149, 169, 182
Goldman Tr. (Ex. 1085), 30:8-16

PO's half-proposed construction of “adjusting said selection and [independently adjusting] playback is inconsistent with PO's positions in the litigation

6 The remainder of this element introduces the concept of ABR:
7 playing back received multimedia objects . . . such that fluidity, video
8 quality and audio quality are maintained by selecting a plurality of said
9 multimedia objects that reflect available network bandwidth, autonomously
 adjusting said selection and playback according to the multimedia object
 parameters and supplied host path identification.
10 Media fluidity and quality are maintained by the client player through references to: (1) available
11 network bandwidth; and (2) multimedia object parameters. To overcome bandwidth limitations,
12 media is streamed in multiple quality levels. See 14:7-12 (“This process encodes a Supplied
13 Identification to each multimedia object to identify the . . . number of objects of a *particular*
14 multimedia stream.”); see also 3:18-20 (“the mobile client will . . . interact with any given media
15 sequence *on a per object basis*.”) Thus, object parameters (*e.g.*, data rate or size of visual frame)
16 are compared to bandwidth in conjunction with the location of the files (*e.g.*, URL) to stream the
17 media with the quality level being adjusted based on bandwidth (*e.g.*, lower bandwidth results in
18 a lower data rate, a lower visual frame, or both).³

Ex. 1083 at 0014
See Reply at 5

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 40-44; Reply at 4-8;
Polish Decl. (Ex. 1002) at ¶196 18

PO's half-proposed construction of "adjusting said selection and playback" is inconsistent with PO's position in the litigation

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by	The YT Player selects a playlist variant based on bandwidth: Using playback statistics, we are able to model the behavior of the player as it switches between resolutions. This allows us in effect to predict when an increased bitrate would be more likely to cause a player to switch to a lower resolution and thereby cancel the effect of bitrate increase in any one resolution. With this model, we are able to find better operating points for each video in the real world.¹ Ex. 18, <i>Making high quality video efficient</i>. YouTube engineering documentation explicitly states that the YouTube player (on the client) autonomously (i.e., without server-side intervention) adjusts the video playback.  YouTube Engineering and Developers Blog What's happening with engineering and developers at YouTube Making high quality video efficient Tuesday, April 24, 2018 YouTube works hard to provide the best looking video at the lowest bandwidth. One way we're doing that is by optimizing videos with bandwidth in mind. We recently made videos stream better – giving you higher-quality video by improving our videos so they are more likely to fit into your available bandwidth. When you watch a video the YouTube player measures the bandwidth on the client and adaptively chooses chunks of video that can be downloaded fast enough, up to the limits of the device's viewport, decoding, and processing capability. YouTube makes multiple versions of each video at different resolutions, with bigger resolutions having higher encoding bitrates. <i>Id.</i>
--	---

When you watch a video the YouTube player measures the bandwidth on the client and adaptively chooses chunks of video that can be downloaded fast enough, up to the limits of the device's viewport, decoding, and processing capability. YouTube makes multiple versions of each video at different resolutions, with bigger resolutions having higher encoding bitrates.

Reply at 4-5,
citing Ex1055 at 053

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 39; Reply at 4-5;
Polish Decl. (Ex. 1002) at ¶¶181, 196 19

Carmel's teachings materially match what PO alleged "explicitly" describes adjusting playback in its infringement allegations

Compare

PO's infringement allegations (Ex. 1055):

YouTube engineering documentation explicitly states that the YouTube player (on the client) autonomously (*i.e.*, without server-side intervention) adjusts the video playback.

When you watch a video the YouTube player measures the bandwidth on the client and adaptively chooses chunks of video that can be downloaded fast enough, up to the limits of the device's viewport, decoding, and processing capability. YouTube makes multiple versions of each video at different resolutions, with bigger resolutions having higher encoding bitrates.

Ex. 1055 at 053
See Reply at 2-5

Carmel:

FIG. 6B is a flow chart illustrating the operation of clients 30 in downloading and playing back multi-level data stream 41 (FIG. 3D) transmitted from server 36, in accordance with another preferred embodiment of the present invention. As in the method of FIG. 6A, each client 30 connects to the server, generally using a single HTTP link. After reading header 43 and, preferably, making an initial assessment of the link bandwidth, the client selects one of the available quality levels in the stream. Responsive to the selection, server 36 begins to transmit data slices at the chosen quality level. The slices are received, decoded and output by the client.

Periodically, client 30 makes an assessment of the rate of data transfer over the link from the server and, if necessary, changes the quality level accordingly. For example, if the rate is low, such that time stamps 59 indicate that the slices need to be played as fast as or faster than they are being received, the client will preferably select a lower quality level if one is available. On the other hand, if the rate is substantially higher than what is needed to receive the successive slices on time, the client may select a higher quality level to take advantage of the available bandwidth.

Carmel (Ex. 1003), 10:64-11:18

See also Petition at 35-44; Reply at 2-5;
Polish Decl. (Ex. 1002) at ¶¶181, 189-194, 196 20

Carmel + Mattavelli teaches “autonomously adjusting said selection and playback” even under PO’s incorrect, half-proposed construction

197. Additionally, as discussed in greater detail below for claim element 1[c.iv] and incorporated here, Mattavelli also teaches autonomously adjusting the playback of received multimedia objects according to multimedia object parameters. In short, Mattavelli teaches techniques for computational graceful degradation for decoding in which object parameters (including, *e.g.*, decoding complexity metadata that is available with the MPEG-4 standard) are used to determine when and by how much to adjust the decoding/playback of received video segments in view of processor constraints of the decoder/multimedia player. (See Mattavelli at 2309 (...), 2311 (...), 2311-12.)

Polish Decl. (Ex. 1002) at ¶197

Carmel + Mattavelli teaches “autonomously adjusting said selection and playback”

The implementation of software based video decoders is an advantageous solution over traditional dedicated hardware real-time systems. However, guaranteeing real-time performance, needed when processing video/audio bit-streams, by scheduling processing tasks of variable load and that might exceed the available resources, is an extremely difficult and in many cases an impossible challenge. In this paper we introduce the motivations and ideas of the Computational Graceful Degradation (CGD) approach for the implementation of video decoding. The goal is to provide results of possibly lower visual quality, but respecting the real-time constraints, when the required processing needs exceed the available processing resources. This approach is very attractive since it enables a more efficient usage of the hardware processing power without needing to implement resources for worst case complexity decoding. We show how such technique can be applied to the decoding of compressed video sequences. Simulation results for MPEG-4 and H.263 video compression standards showing some of the interesting achievements and potentialities of CGD approach are also reported. The presented approach can be virtually applied to any video compression standard and processor based platform, although the best performances are achieved when complexity prediction information provided as side information by the video standard is available. This option has been included in the new MPEG-4 standard.

The second strategy is to exploit the complexity information carried by the frames or VOP headers, like it can be done for MPEG-4 systems [6], and use it in two stages. The first stage predicts the decoding time needed for the current frame or VOP, as described in [5]. The second, on the basis of the predicted decoding time and on the complexity information (i.e. statistics of the algorithms necessary to correctly decode the frame), executes only the CGD algorithms that are just necessary to not exceed the target decoding time.

Mattavelli (Ex. 1004)
at 2311

Mattavelli (Ex. 1004)
at 2309

We have introduced the motivations and concepts of the CGD approach for the implementation of video decoders on processor based platforms. Simulation results for MPEG-4 video decoding demonstrate the successful implementation of the CGD approach. Obviously, the implementation of CGD features are not proposed to be used at their maximum degradation rate but as tool able to guarantee real-time performance in presence of peaks of decoding complexity or as a way to ease the scheduling problems given by temporary fluctuations of the available processing power. Moreover, the presented approach may enable the successful implementation of video decoding for levels and profiles that would be forbidden if the complexity of *worst case decoding* has to be guaranteed.

Mattavelli (Ex. 1004) at
2312

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 44-50; Reply at 11-12;
Polish Decl. (Ex. 1002) at ¶¶197-208, 209-222 22

Carmel + Mattavelli teaches “autonomously adjusting said selection and playback”

- In the POR, PO failed to challenge or dispute Petitioner’s showing that the complexity prediction information included in the MPEG-4 standard is an “object parameter” associated with multimedia objects (claim element 1[a.ii])
 - See Petition at 21-22, 44-45; POR at 33-34

also reported. The presented approach can be virtually applied to any video compression standard and processor based platform, although the best performances are achieved when complexity prediction information provided as side information by the video standard is available. This option has been included in the new MPEG-4 standard.

Mattavelli (Ex. 1004) at 2309

The second strategy is to exploit the complexity information carried by the frames or VOP headers, like it can be done for MPEG-4 systems [6], and use it in two stages. The first stage predicts the decoding time needed for the current frame or VOP, as described in [5]. The second, on the basis of the predicted decoding time and on the complexity information (i.e. statistics of the algorithms necessary to correctly decode the frame), executes only the CGD algorithms that are just necessary to not exceed the target decoding time.

Mattavelli (Ex. 1004) at 2311

Carmel + Mattavelli teaches “autonomously adjusting said selection and playback”

- PO attacks Mattavelli and Carmel individually; not the proposed combination of Carmel + Mattavelli (*E.g.*, PO Sur-Reply at 18-19)



UNITED STATES COURT OF APPEALS FOR THE
FEDERAL CIRCUIT

“Non-obviousness cannot be established by attacking references individually where the rejection is based upon the teachings of a combination of references.”

In re Merck & Co., Inc., 800 F. 2d 1091, 1097 (Fed. Cir. 1986)

1. A method comprising:

utilizing optimized decoding processes to maintain quality playback.

'849 patent, cl. 1.



Marcus Murravelli, Sylvain Brunetton
Integrated Systems Center - CSI,
Swiss Federal Institute of Technology,
Lausanne CH-1015, SWITZERLAND
Tel: +41 21 6936979; fax: +41 21 6934065
e-mail: marco.murravelli@epfl.ch

[illegible]

The concept of computational graph reduction using imperative rules has previously been introduced in the literature [1, 2]. In this paper, we show how to achieve this by using a declarative approach, achieved when complexity prediction information is provided in the form of a declarative constraint language. This option has been included in the new MPEG-7.

1 INTRODUCTION

The complexity of video decoding is a fluctuating function of time [1]. By complexity we mean the processing time necessary to decode a compressed video bit stream. Even considering a specific processor platform, a specific compression standard, with a relative level and profile, the complexity of video decoding is a variable quantity. Dedicated hardware platforms can guarantee worst case complexity decoding. In contrast, in the case of programmable platforms such as *gate arrays*, would imply the allocation of a processing power up to

The written description's only disclosure of “fluid” playback materially matches Carmel’s teachings relied on in the Petition

Compare

'849 patent:

The buffering and playback of multimedia objects in a synchronized fashion is critical to fluid playback.

'849 (Ex. 1001) at 3:43-45
See POR at 34

Carmel:

In the case of a multimedia stream, client 30 reconstructs and outputs the multimedia data for the appreciation of a user. Time stamps in the data stream are used to synchronize the data, so that the multimedia sequence is played back just as it was input at computer 34, preferably with only a minimal necessary transmission and decoding delay.

Carmel (Ex. 1003), 10:48-54;
see also id., 7:28-41, 10:64-11:18, Fig. 6B

Patent Owner's IPR Position on "Fluidity"

"'Fluidity' refers to the smoothness and continuity of playback as perceived by the viewer, specifically the absence of jitter, freezes, or choppiness."

POR at 34

"Petitioner's assumption that playing back a sequence in real-time and in the order it was input is equivalent to maintaining fluidity is incorrect."

POR at 36

"In order for playback to be fluid, the three steps claimed in the '849 Patent must be taken."

POR at 35, 40

In the litigation, PO recognized that the claim language is satisfied when “[m]edia fluidity and quality are maintained” using multi-level streaming

6 The remainder of this element introduces the concept of ABR:
7 playing back received multimedia objects . . . such that fluidity, video
8 quality and audio quality are maintained by selecting a plurality of said
9 multimedia objects that reflect available network bandwidth, autonomously
10 adjusting said selection and playback according to the multimedia object
11 parameters and supplied host path identification.
12 Media fluidity and quality are maintained by the client player through references to: (1) available
13 network bandwidth; and (2) multimedia object parameters. To overcome bandwidth limitations,
14 media is streamed in multiple quality levels. See 14:7-12 (“This process encodes a Supplied
15 Identification to each multimedia object to identify the . . . number of objects of a *particular*
16 multimedia stream.”); see also 3:18-20 (“the mobile client will . . . interact with any given media
17 sequence *on a per object basis*.”) Thus, object parameters (*e.g.*, data rate or size of visual frame)
18 are compared to bandwidth in conjunction with the location of the files (*e.g.*, URL) to stream the
 media with the quality level being adjusted based on bandwidth (*e.g.*, lower bandwidth results in
 a lower data rate, a lower visual frame, or both).³

Ex. 1083 at 0014
See Reply at 5

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 40-44; Reply at 4-8;
Polish Decl. (Ex. 1002) at ¶196 28

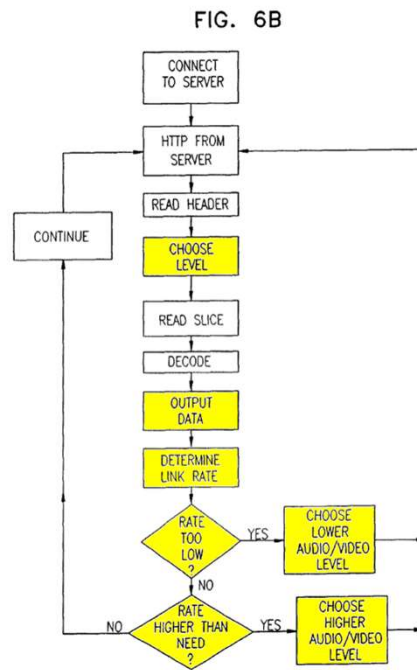
Carmel alone teaches maintaining fluidity in the face of bandwidth constraints

A method for **real-time broadcasting** from a transmitting computer to one or more client computers over a network, including providing at the transmitting computer a data stream having a given data rate, and dividing the stream into a sequence of slices, each slice having a predetermined data size associated therewith.

Carmel (Ex. 1003), Abstract

In the case of a multimedia stream, client 30 reconstructs and **outputs the multimedia data for the appreciation of a user**. Time stamps in the data stream are used to synchronize the data, so that the multimedia sequence is played back just as it was input at computer 34, preferably with only a minimal necessary transmission and decoding delay.

Carmel (Ex. 1003), 10:48-54



Carmel,
Figure 6B
(Ex1003)

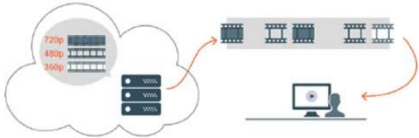
Preferably, each file also includes one or more time stamps, indicating a real time at which the data in the file were recorded or an elapsed time relative to the beginning of stream 40. ... Computer 34 monitors the time codes as file 40 is transmitted, and clients 30 similarly monitor the time codes as the file is received, in order to ensure that the transmission or reception is “keeping up” with the input of the data to the computer. In the event that a lag is detected, steps are taken to increase the data transmission or reception rate.

Carmel (Ex.1003), 7:28-42

Periodically, client 30 makes an **assessment** of the rate of data transfer over the link from the server and, if necessary, changes the quality level accordingly. For example, if the rate is low, such that time stamps 59 indicate that the slices need to be played as fast as or faster than they are being received, the client will preferably select a lower quality level if one is available. On the other hand, if the rate is substantially higher than what is needed to receive the successive slices on time, the client may select a higher quality level to take advantage of the available bandwidth.

Carmel (Ex.1003), 10:64-11:18

PO’s infringement allegations assert that “fluidity, video quality, and audio quality are maintained based on multiple versions of each stream that fit available bandwidth”

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by	YouTube playback fluidity, video quality, and audio quality are maintained based on multiple versions (green) of each video that reflect available bandwidth (yellow): <pre>#EXTM3U #EXT-X-INDEPENDENT-SEGMENTS #EXT-X-MEDIA:URI="https://manifest.goo #EXT-X-MEDIA:URI="https://manifest.goo #EXT-X-STREAM-INF: BANDWIDTH=174292, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=258776, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=372198, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=491883, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=663742, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=158993, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=153955, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=247948, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=332522, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=587726, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=716829, COO https://manifest.googlevideo.com/api/m #EXT-X-STREAM-INF: BANDWIDTH=1314966, CO https://manifest.googlevideo.com/api/m</pre> Video.	playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by	YouTube playback fluidity, video quality, and audio quality are maintained based on multiple versions of each stream that fit available bandwidth:  Engineering and Developers Blog What's happening with engineering and developers at YouTube Making high quality video efficient Tuesday, April 24, 2018 YouTube works hard to provide the best looking video at the lowest bandwidth. One way we're doing that is by optimizing videos with bandwidth in mind. We recently made videos stream better -- giving you higher-quality video by improving our videos so they are more likely to fit into your available bandwidth. When you watch a video the YouTube player measures the bandwidth on the client and adaptively chooses chunks of video that can be downloaded fast enough, up to the limits of the device's viewport, decoding, and processing capability. YouTube makes multiple versions of each video at different resolutions, with bigger resolutions having higher encoding bitrates.  Figure 1: HTTP-based Adaptive Video Streaming. Ex. 18. <i>Making high quality video efficient</i>, YouTube Engineering and Developers Blog, https://youtube-eng.googleblog.com/2018/04/making-high-quality-video-efficient.html (Apr. 24, 2018), at 1.
--	--	--	--

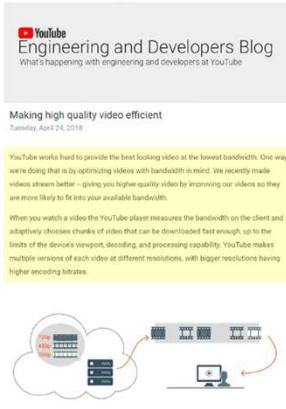
Ex. 1055 at 050-052
See Petition, 39; Reply, 16-17

Carmel teaches maintaining fluidity, video quality, and audio quality based on multiple versions of each stream that fit available bandwidth

Compare

PO's infringement allegations:

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by	YouTube playback fluidity, video quality, and audio quality are maintained based on multiple versions (green) of each video that reflect available bandwidth (yellow):
	YouTube playback fluidity, video quality, and audio quality are maintained based on multiple versions of each stream that fit available bandwidth:



Ex. 18, Making high quality video efficient, YouTube Engineering and Developers Blog, <https://youtube-eng.googleblog.com/2018/04/making-high-quality-video-efficient.html> (Apr. 24, 2018), at 1.

Ex. 1055 at 050-052
See Petition, 39; Reply, 16-17

PO Expert's declaration re: Carmel:

37. The '473 Patent further teaches that during real-time streaming, client computers may select a file from different quality levels corresponding to that slice of multimedia data depending on the existing network conditions, such as varying bandwidth constrictions and traffic bottlenecks.⁷

38. This feature, totally unavailable in the then-current state of the art was adopted several years later as standard in the industry. *Implementing Adaptive HTTP Streaming Using the Web*, pp. 1-2 (Exhibit E).

Goldman Decl. from Carmel Reexam (Ex. 1081) at 9-10
See also Carmel (Ex. 1003), 8:56-9:9; 10:64-11:18, Figs. 3D, 6B

PO's expert used the term "real-time" to characterize a streaming system/process that operates fast enough to provide fluid playback

E.g., re: the '849 patent's disclosure:

4 Q. Okay. So what does it mean to say 11:59:57
5 "consistent with the video codec specification and 12:00:03
6 displays a correct image"? 12:00:06
7 MR. GLENNON: Objection. Beyond the 12:00:10
8 scope. 12:00:11
9 THE WITNESS: I took that to mean that the 12:00:11
10 highest quality, given what these tools are doing, 12:00:19
11 which are trading off quality for realtime 12:00:23
12 processing or being able to deliver a fluid 12:00:27
13 picture, due to constraints and limitations in the 12:00:31
14 client and the network. 12:00:34

Goldman 7/21/25 Tr. (Ex. 1080), 65:4-14

1 Would it be fair to say that the goal for
2 fluidity of a video is that the viewer perception
3 of the video matches the in-person experience?
4 A I think that's fair.

Goldman 4/7/26 Tr. (Ex. 1085), 22:1-4

E.g., re: Carmel's teachings:

46. The invention claimed in the '473 Patent is directed to "real-time broadcasting" (claim 1, preamble). This is defined as the "simultaneous transmission of data to one or more clients matching the human perception of time or proceeding at the same rate as a physical or external process". In other words, the '473 Patent seeks to make the client's experience of watching a video being broadcast in accordance with the method disclosed therein to be as close as possible to matching a "live" or in-person experience.

Goldman Decl. from Carmel
Reexam (Ex. 1081) at 9-10

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 7,13, 37; Reply at 15, 23-24;
Polish Decl. (Ex. 1002) at ¶¶78, 114, 124, 178, 184-185 32

Carmel's teachings connect transmission/download rates to replay rates and playback fluidity

Again, Patent Owner's expert, Mr. Goldman, acknowledged the connection taught by Carmel between download rate and the replay rate

41. Fourth, the '473 Patent ensures real-time delivery of multimedia streams by maintaining the upload (Claim 1), download (Claim 1) and replay (Claim 8) rates so they are generally equal to the assigned (given) data rate of the stream. For instance, the '473 Patent discloses an embodiment whereby the amount of data generated by the transmitting computer over a period of time (e.g., 10 seconds) is uploaded to a server within 10 seconds and is subsequently downloaded within 10 seconds by the client computer. Hence, there is a balance of flow between the production, transmission, and consumption of a real-time multimedia stream, resulting in a stable and scalable operation. The '473 Patent provides explicit steps for rate control, balancing the given data rate provided at the transmitting computer, the upload rate, download rate, and replay rate, including techniques for adjusting these rates synergistically, through selectable compression choices and other techniques. The invention's rate control techniques are employed when the rates of the upload and download are slow compared to the data rate(s) of the multimedia stream(s), or when the upload and download rates are faster than the data rates of the multimedia streams. '473 Patent, 11:9-22.

Goldman Decl. from Carmel Reexam (Ex. 1081) at 11

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

8. A method according to claim 2, wherein the one or more client computers decode the sequence and play back the data stream responsive to the indices of the files, at a replay rate generally equal to the data rate.

Carmel (Ex. 1003) at cl. 8

See also Petition at 39-40, 50-52; Reply at 8-9;
Polish Decl. (Ex. 1002) at ¶¶149, 169, 182 33

The '849 patent uses the term “real-time” to characterize systems/processes that operate fast enough to provide fluid playback

'849 patent:

The current methods of distributing multimedia data over the wireless Internet to mobile clients are constrained by existing wireless bandwidth, and the **real-time decoding, processing and displaying of multimedia content** with limited hardware capabilities.

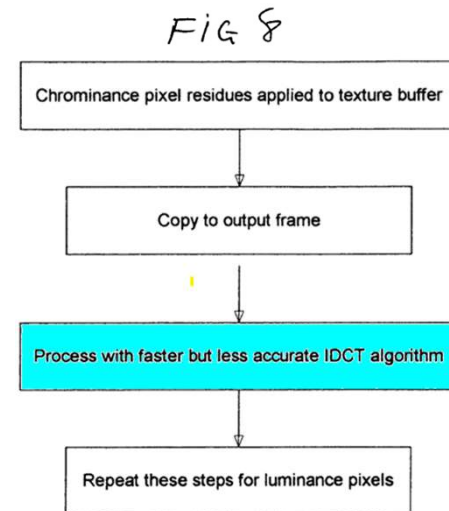
'849 (Ex. 1001) at 1:13-17

The Chen's IDCT algorithm is optimized, **but the processing time consumed by the standard Chen implementation is too great for real-time decoding on limited devices**. Hence, FIG. 11 is a flow diagram illustrating a novel use of Chen's algorithm in the multimedia object players **10** and **11**.

'849 (Ex. 1001) at 12:63-13:1

In cases where the device is **unable to decode the video stream in real-time, a faster but less accurate IDCT algorithm can be used to process these residue values**. Furthermore, to minimize the effect of the less accurate IDCT algorithm, this step is taken first on chrominance pixel residues, but can also occur for luminance pixel residues as required.

'849 (Ex. 1001) at 12:12-18, Fig. 8



PO's claimed steps that result in maintaining fluidity

1. A method comprising:

receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence **such that fluidity, video quality and audio quality are maintained** by ^{1a}selecting a plurality of said multimedia objects that reflect available network bandwidth, ^{1b}**autonomously adjusting said selection and playback** according to the multimedia object parameters and supplied host path identification, and by ²

utilizing optimized decoding processes to maintain quality playback.

6. A multimedia player comprising:

computer executable instructions stored on a non-transitory computer readable medium which when executed by a processor:

requests transmission of multimedia objects;

receives audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated object parameters and supplied host and path identification to form multimedia objects; and wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

plays back received multimedia objects in a sequence **such that fluidity, video quality and audio quality are maintained** by ^{1a}selecting a plurality of said multimedia objects that reflect available network bandwidth, ^{1b}**autonomously adjusting said selection and playback** according to the multimedia object parameters and supplied host path identification, and by ²**utilizing optimized decoding processes** to maintain quality playback.

7. A system comprising

at least one centralized content server transmitting multimedia objects; said multimedia objects including segmented audio and video, and parameters and supplied host path identification;

at least one multimedia player configured to:

request transmission of multimedia objects;

receive audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated object parameters and supplied host and path identification to form multimedia objects; and wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

play back received multimedia objects in a sequence **such that fluidity, video quality and audio quality are maintained** by ^{1a}selecting a plurality of said multimedia objects that reflect available network bandwidth, ^{1b}**autonomously adjusting said selection and playback** according to the multimedia object parameters and supplied host path identification, and by ²**utilizing optimized decoding processes** to maintain quality playback.

'849 patent, cls. 1, 6 & 7

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 33-50, 65-67;
Polish Decl. (Ex. 1002) at ¶¶256-264 35

Mattavelli teaches maintaining fluidity versus using un-fluid alternatives, such as skipping frames, that result in interruptions when resources are overtaxed

needed for an average decoding [1]. Conversely, a theoretical average value is not useful for an efficient scheduling of a decoding task. The average value for each considered sequence is not stationary, and its variation around the theoretical average one are so unpredictable that no useful scheduling policy can be realized. The first straightforward option is a sustained over-sizing of the processing resources. The second is to accept interruptions in the audio/video output streams when the needed decoding resources exceed the ones available in the system. Obviously both options are not desirable being respectively not economical, or providing a quality of the services which may easily result perceptually unacceptable. The main rationale of the Computational Graceful Degradation (CGD) approach is to provide results of poorer quality when the results of the desirable quality cannot be obtained in real time. When the complexity of the task exceeds the available processing power, real-time services, possibly of degraded quality, are nevertheless guaranteed respecting all real-time constraints. This approach enables the correct scheduling of real-time tasks even when the computational system load fluctuates above the system capabilities.

Mattavelli (Ex. 1004) at 2309

are not suitable for the purposes of video decoding. In our case, the goal is to guarantee real-time video decoding with possible degradations of the image quality, i.e. reductions of the peak signal to noise ratio (PSNR), without losing parts of the image content. Such alternative is certainly much more desirable than other straightforward options such as cutting image portions or simply skipping frames. These latter options, as can be easily understood, can have catastrophic consequences on prediction based coding systems. In H.263 decoders, for instance, skipping the decoding of a single frame or a portion of it, implies the skipping of all future incoming frames (or portion of them) before the next incoming Intra frame. The time interval of such skipping operation might cover many seconds and the information content loss might be very relevant. Such abrupt interruptions, which are perceptually very annoying, might be caused not only by the simple fluctuation of the decoding complexity load, but also by the fluctuations of the load of other non-video processing tasks which reduce the available processing resources. Moreover, such events in general are out of the control of the video decoder system and cannot be easily forecasted. Considering these facts, it is evident the importance of techniques able to filter out complexity decoding peak loads or able to reduce the average decoding computation load without causing perceptually annoying consequences. Skipping B-VOP could be, in some cases, a

Mattavelli (Ex. 1004)
at 2309-10

Carmel + Mattavelli teaches maintaining fluidity

- The combined teachings of Carmel and Mattavelli teach maintaining fluidity (and video and audio quality) in view of resource constraints, including constraints on bandwidth and processing resources

215. A person of ordinary skill in the art would have been motivated to combine the complementary methods and techniques taught by Carmel and Mattavelli to form a streaming system that is more robust and better able to handle these multiple known potential bottlenecks or constraints in the chain of transmission of a multimedia bitstream from its source to the end user in a way that provides the best possible playback experience. A person of ordinary skill in the art would have recognized that a multimedia streaming system would potentially be subject to constraints on (i) available network bandwidth, (ii) multimedia player processing resources, or (iii) both at the same time, and would have been motivated to apply Mattavelli's teachings regarding CGD decoding techniques to the multimedia player/decoder on the (wireless) client 30 taught by Carmel to mitigate/address both of these known problems or bottlenecks in providing the best possible streaming playback performance.

Polish Decl. (Ex. 1002) at ¶215

The Challenged Claims are not limited to mobile devices

1. A method comprising:

receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers **using a wireless connection**;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

utilizing optimized decoding processes to maintain quality playback.

'849 patent, cl. 1.

- Challenged claims recite using a “wireless connection;” not mobile device
- PO’s expert admitted that, e.g., laptops can use wireless connections

19 Q Is it a laptop on a wireless connection or
20 on a wired connection?

21 A Yes, it is on a -- it is on my home wi-fi
22 network.

1 Q Is that wireless?

2 A It is wireless.

Goldman 4/7/26 Tr. (Ex. 1085), 5:19-6:2

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See *also* Petition at 29-32; Reply at 15-16;
Polish Decl. (Ex. 1002) at ¶¶136, 160-164 38

Ground 1 – Issue 3: Carmel + Mattavelli teaches “utilizing optimized decoding processes to maintain quality playback” and “fluidity, video quality and audio quality are maintained”

1. A method comprising:

receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

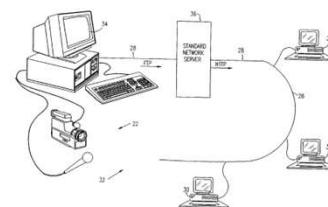
requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence **such that fluidity, video quality and audio quality are maintained** by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

utilizing optimized decoding processes to maintain quality playback.

'849 patent, cl. 1.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE



Google Exhibit No. 1003
Page 001

COMPUTATIONAL GRACEFUL DEGRADATION FOR VIDEO DECODING

Marcus Mattavelli, System Architect
Integrated Systems Center, CHL
Swiss Federal Institute of Technology,
Lauterbach CH-1015, SWITZERLAND
Tel: +41 21 6930579; fax: +41 21 6930630
e-mail: Marcus.Mattavelli@epfl.ch

ABSTRACT

The implementation of software based video decoders in an asynchronous real-time system, where traditional dedicated hardware real-time systems. However, guaranteeing real-time performance, models when processing video/multimedia streams, by scheduling processing tasks of variable load and that might exceed the available resources, is an extremely difficult and in many cases an impossible challenge. In this paper we introduce the motivation and ideas of the Computational Graceful Degradation (CGD) approach for the implementation of video decoding. The goal is to provide results of possibly lower visual quality, but improving the real-time constraints, when the required processing needs exceed the available processing resources. This approach is very attractive since it enables a more efficient usage of the hardware processing power without needing to implement resources for worst case complexity decoding. We show how such technique can be applied to the decoding of compressed video sequences. Simulation results for MPEG-4 and H.263 video compression standards, during some of its interesting achievements and potentialities of CGD approach are also reported. The presented approach can be virtually applied to any video compression standard and processor based platform, although the best performance are achieved when complexity prediction information provided as side information to the video standard is available. This option has been included in the new MPEG-4 standard.

1. INTRODUCTION

The complexity of video decoding is a fascinating function of them [1]. By complexity we mean the processing time necessary to decode a compressed video stream. From complexity, a specific processing platform, a specific compression standard, with a relative level and profile, the complexity of video decoding is a variable quantity. Dedicated hardware platforms can guarantee worst case complexity decoding. In contrast, in the case of programmable platforms such a guarantee would imply the allocation of a processing power up to

two orders of magnitude higher than the one actually needed for an average decoding [2]. Conversely, a theoretical average value is not useful for an efficient scheduling of a decoding task. The average value for each considered sequence is not constant and its variation around the theoretical average one are so unpredictable that no useful scheduling policy can be devised. The first straightforward option is a constant over-estimation of the processing resources. The second is to accept interruptions in the audio/video output streams when the needed decoding resources exceed the ones available in the system. Obviously both options are not desirable being respectively not economical, or providing a quality of the services which may really result progressively unacceptable. The main objective of the Computational Graceful Degradation (CGD) approach is to provide results of good quality when the results of the available quality cannot be obtained in real time. When the complexity of the task exceeds the available processing power, real-time services, possibly of degraded quality, are nevertheless guaranteed respecting all real-time constraints. This approach enables the correct scheduling of real-time tasks even when the computational system load fluctuates above the system capabilities.

The concept of computational graceful system degradation using heuristic results has previously been introduced in the field of real-time computing with application to control systems and automation. In these contexts heuristic results are defined as intermediate results with the property that the more recent and immediate result is the closer one that approximates the final result (i.e. the idea only works in non-monotonically improving computational [2, 3, 4]). In these cases the problem to be solved are more concerned with final resources (CPU resources, stability and sensitivity to unexpected inputs, and the corresponding techniques are more concerned with the final result. In our case, the goal is to guarantee real-time video decoding with possible degradation of image quality (reduction of the peak signal to noise ratio (PSNR), without losing parts of the image content). Such a

See also Petition at 44-50, 50-55; Reply at 20-24; Polish Decl. (Ex. 1002) at ¶¶199-208, 209-222

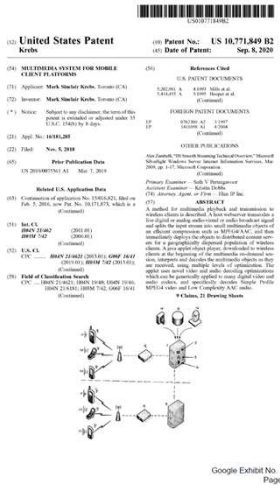
PO's critique of Mattavelli rests on an incorrect construction of "maintain" that is inconsistent with the claims, specification, and PO's positions outside of this IPR

"Degrading is not maintaining video quality."

POR at 45

- PO offers a *de facto* construction of "*maintaining*" fluidity, video quality, and audio quality to mean that there can never be **any** change (degradation) in **any** of these characteristics, regardless for how long, for what reason, or for what purpose (such as preserving fluidity at the expense of visual quality).
- PO's *de facto* construction of "maintaining" is incorrect and inconsistent with:
 - '849 claim language
 - '849 written description
 - PO expert's sworn testimony in district court

PO's *de facto* construction of “maintaining” is inconsistent with the Challenged Claims



1. A method comprising:

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence **such that fluidity, video quality and audio quality are maintained** by^{1a} selecting a plurality of said multimedia objects that reflect available network bandwidth,^{1b} **autonomously adjusting said selection and playback** according to the multimedia object parameters and supplied host path identification, and by

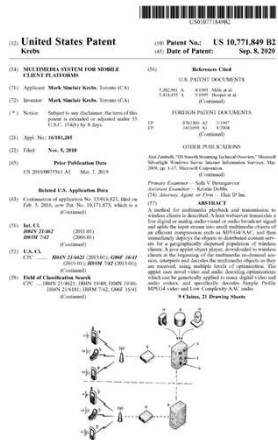
²utilizing optimized decoding processes to maintain
quality playback.

'849 patent, cl. 1.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See *also* Reply at 20-21;
Polish Decl. (Ex. 1002) at ¶¶199-208, 209-222 41

PO's *de facto* construction of “maintaining” is inconsistent with the ’849 specification



Google Exhibit No. 1001
Page 001

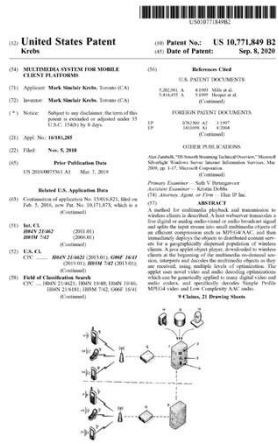
[T]he invention provides **novel optimizations for digital video decoding**. Some of these optimizations can then be used by an expert assessment process, whereby, the decoder maintains a state information list of shortcuts, related to perceived frame rate in a sorted list starting with those that **will decrease output quality the least, to those that will decrease output quality the most** but have the most impact on decoding speed. The client player-decoder *dynamically adjusts* how many shortcuts must be taken. These short-cuts are specifically designed to drastically reduce the number of computations necessary at certain critical steps in the video decoding process **at the cost video output quality**. This allows the video decoder to scale in complexity based on the processing power of the device being used. It also allows users to experience multimedia playback despite the limitations of the device they may be using.

'849 patent, 4:7-23

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Reply at 20-21;
Polish Decl. (Ex. 1002) at ¶¶64 42

PO's *de facto* construction of “maintaining” is inconsistent with the ’849 specification

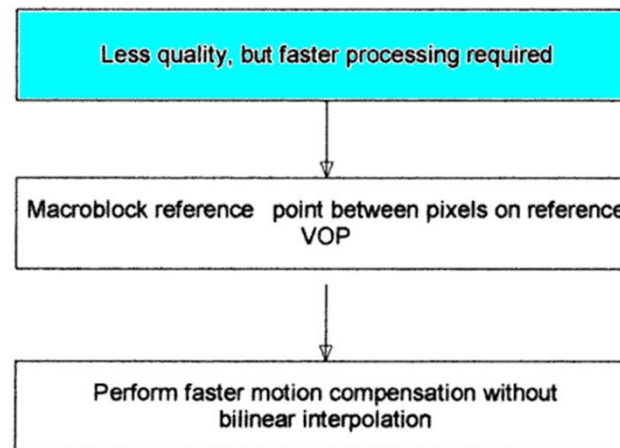


Google Exhibit No. 1001
Page 001

In addition to various arithmetic optimizations performed the video decoding process of the multimedia object players **10** and **11**, shown in the flow diagram of FIG. 9, uses **faster motion compensation without bilinear interpolation when less quality but faster processing is required.**

FIG 9

'849 patent, 12:12-29



'849 patent, Fig. 9

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Reply at 20-21;
Polish Decl. (Ex. 1002) at ¶¶64 43

PO's *de facto* construction of “maintaining” is inconsistent with the ’849 specification



Google Exhibit No. 1001
Page 001

The **highest quality** is consistent with the video codec specification and displays a correct image. A **medium quality level** indicates that certain time consuming short-cuts are made with some impact on image quality. A **low quality level** indicates that drastic reductions in display quality are made to **improve processing time**—the output video can be unrecognizable at times and as a result this level is used only in drastic cases of a sudden drop in processor availability.

'849 patent, 13:60-14:15, Fig. 13

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See *also* Reply at 20-21;
Polish Decl. (Ex. 1002) at ¶¶64 44

PO's *de facto* construction of “maintaining” is inconsistent with its expert’s positions outside of this IPR

2 Q. But isn't it the case that to meet the 10:56:46
3 definition of optimized decoding process, you 10:56:51
4 actually have to degrade some amount of audio and 10:56:55
5 video quality to reduce the complexity of the 10:56:58
6 calculations or the number of the calculations? 10:57:03
7 A. Well, yes. That's exactly what the patent 10:57:08
8 described, and it states that explicitly. 10:57:11

Goldman 7/21/25 Tr. (Ex. 1080), 37:2-8

PO's *de facto* construction of “maintaining” is inconsistent with its expert’s positions outside of this IPR

6 So my question to you is if you were 10:37:32
7 reducing certain aspects of quality but are 10:37:35
8 maintaining fluidity, would that satisfy this 10:37:39
9 claim element which refers to "utilizing optimized 10:37:45
10 decoding processes to maintain quality playback"? 10:37:52
11 A. Yes. 10:37:57

Goldman 7/21/25 Tr. (Ex. 1080), 25:6-11

13 So, yes, a POSA, from reading the patent 10:40:49
14 itself and understanding what the tool -- each 10:40:52
15 tool does and is understanding that the goal is to 10:40:54
16 have -- is to maintain quality playback, would 10:40:58
17 come to the same conclusion that the fluidity is 10:41:01
18 the more important factor, and it would sacrifice 10:41:04
19 some of the other elements to do that. 10:41:08
20 Again, it's not an absolute quality that's 10:41:10
21 being measured here. It's to maintain the best 10:41:12
22 quality you can, given network bandwidth and other 10:41:15
23 network impairments and the capability of the 10:41:19
24 device itself. 10:41:21

Goldman 7/21/25 Tr. (Ex. 1080), 27:13-24

PO's *de facto* construction of “maintaining” is inconsistent with its expert’s positions outside of this IPR

9 So, again, the metric that's being used -- 11:01:03
10 I'm using your term "metric" -- would be that by 11:01:07
11 using this complexity reducing tool, I reduce it 11:01:10
12 enough that I still -- I reduce the video and 11:01:15
13 audio quality enough so I still have fluidity. 11:01:19
14 And that is better than the result of not applying 11:01:21
15 the tool, because the complex calculations can't 11:01:25
16 be completed in time, and there will be drastic 11:01:28
17 degradation if you don't apply the tool. 11:01:33

Goldman 7/21/25 Tr. (Ex. 1080), 40:9-17

19 Q. And video output quality may sometimes in 13:00:23
20 the context of the patent be reduced in order to 13:00:27
21 maintain fluidity; is that fair? 13:00:30
22 A. It is. 13:00:32

Goldman 7/21/25 Tr. (Ex. 1080), 108:19-22

14 Q. So you're reading "audio quality" and 13:10:07
15 "video quality" here to mean the best possible 13:10:10
16 audio quality and video quality that can be 13:10:15
17 obtained while maintaining fluidity? 13:10:18
18 A. Correct. 13:10:22

Goldman 7/21/25 Tr. (Ex. 1080), 115:14-18

19 Q. Okay. And so the video quality may be 13:11:45
20 diminished in order to maintain fluidity? 13:11:48
21 A. Correct. 13:11:51

Goldman 7/21/25 Tr. (Ex. 1080), 116:19-21

Mattavelli teaches “utilizing optimized decoding processes to maintain quality playback” and “fluidity, video quality and audio quality are maintained”

Compare

PO’s expert on what the ’849 patent discloses:

19 Q. And video output quality may sometimes in 13:00:23
20 the context of the patent be reduced in order to 13:00:27
21 maintain fluidity; is that fair? 13:00:30
22 A. It is. 13:00:32

Goldman 7/21/25 Tr. (Ex. 1080), 108:19-22

14 Q. So you're reading "audio quality" and 13:10:07
15 "video quality" here to mean the best possible 13:10:10
16 audio quality and video quality that can be 13:10:15
17 obtained while maintaining fluidity? 13:10:18
18 A. Correct. 13:10:22

Goldman 7/21/25 Tr. (Ex. 1080), 115:14-18

19 Q. Okay. And so the video quality may be 13:11:45
20 diminished in order to maintain fluidity? 13:11:48
21 A. Correct. 13:11:51

Goldman 7/21/25 Tr. (Ex. 1080), 116:19-21

Mattavelli:

tation of video decoding. The goal is to provide results of possibly lower visual quality, but respecting the real-time constraints, when the required processing needs exceed the available processing resources. This approach is very attractive since it enables a more efficient usage of the hardware processing power without needing to implement resources for worst case complexity decoding. We show how such technique can be

Mattavelli (Ex. 1004) at 2309

ally unacceptable. The main rationale of the Computational Graceful Degradation (CGD) approach is to provide results of poorer quality when the results of the desirable quality cannot be obtained in real time. When the complexity of the task exceeds the available processing power, real-time services, possibly of degraded quality, are nevertheless guaranteed respecting all real-time constraints. This approach enables the cor-

Mattavelli (Ex. 1004) at 2309

Obviously for any Allocated Decoding Time the goal is to respect the real-time constraints with the lowest degradation in the video/audio content.

Mattavelli (Ex. 1004) at 2311-12

See also Petition at 44-50, 50-55; Reply at 20-22;
Polish Decl. (Ex. 1002) at ¶¶199-208, 209-222

Again: Mattavelli teaches maintaining fluidity versus using un-fluid alternatives, such as skipping frames, that result in interruptions when resources are overtaxed

needed for an average decoding [1]. Conversely, a theoretical average value is not useful for an efficient scheduling of a decoding task. The average value for each considered sequence is not stationary, and its variation around the theoretical average one are so unpredictable that no useful scheduling policy can be realized. The first straightforward option is a sustained over-sizing of the processing resources. The second is to accept interruptions in the audio/video output streams when the needed decoding resources exceed the ones available in the system. Obviously both options are not desirable being respectively not economical, or providing a quality of the services which may easily result perceptually unacceptable. The main rationale of the Computational Graceful Degradation (CGD) approach is to provide results of poorer quality when the results of the desirable quality cannot be obtained in real time. When the complexity of the task exceeds the available processing power, real-time services, possibly of degraded quality, are nevertheless guaranteed respecting all real-time constraints. This approach enables the correct scheduling of real-time tasks even when the computational system load fluctuates above the system capabilities.

Mattavelli (Ex. 1004) at 2309

are not suitable for the purposes of video decoding. In our case, the goal is to guarantee real-time video decoding with possible degradations of the image quality, i.e. reductions of the peak signal to noise ratio (PSNR), without losing parts of the image content. Such alternative is certainly much more desirable than other straightforward options such as cutting image portions or simply skipping frames. These latter options, as can be easily understood, can have catastrophic consequences on prediction based coding systems. In H.263 decoders, for instance, skipping the decoding of a single frame or a portion of it, implies the skipping of all future incoming frames (or portion of them) before the next incoming Intra frame. The time interval of such skipping operation might cover many seconds and the information content loss might be very relevant. Such abrupt interruptions, which are perceptually very annoying, might be caused not only by the simple fluctuation of the decoding complexity load, but also by the fluctuations of the load of other non-video processing tasks which reduce the available processing resources. Moreover, such events in general are out of the control of the video decoder system and cannot be easily forecasted. Considering these facts, it is evident the importance of techniques able to filter out complexity decoding peak loads or able to reduce the average decoding computation load without causing perceptually annoying consequences. Skipping B-VOP could be, in some cases, a

Mattavelli (Ex. 1004)
at 2309-10

Mattavelli still satisfies the “maintaining” limitations even under PO’s incorrect *de facto* construction

We have introduced the motivations and concepts of the CGD approach for the implementation of video decoders on processor based platforms. Simulation results for MPEG-4 video decoding demonstrate the successful implementation of the CGD approach. Obviously, the implementation of CGD features are not proposed to be used at their maximum degradation rate but as tool able to guarantee real-time performance in presence of peaks of decoding complexity or as a way to ease the scheduling problems given by temporary fluctuations of the available processing power. Moreover, the presented approach may enable the successful implementation of video decoding for levels and profiles that would be forbidden if the complexity of *worst case decoding* has to be guaranteed.

Mattavelli (Ex. 1004) at 2312

8	Q. Would that still be an optimized decoding	12:02:42
9	process?	12:02:44
10	A. Yes. Because you're using the processes	12:02:45
11	in order to make that -- you're using this expert	12:02:47
12	algorithm in order to make a determination of	12:02:51
13	whether or not you need to apply these tools.	12:02:53

Goldman 7/21/25 Tr. (Ex. 1080), 67:8-13

Carmel + Mattavelli teaches “utilizing optimized decoding processes to maintain quality playback,” including maintaining “audio quality”

1. A method comprising:

receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

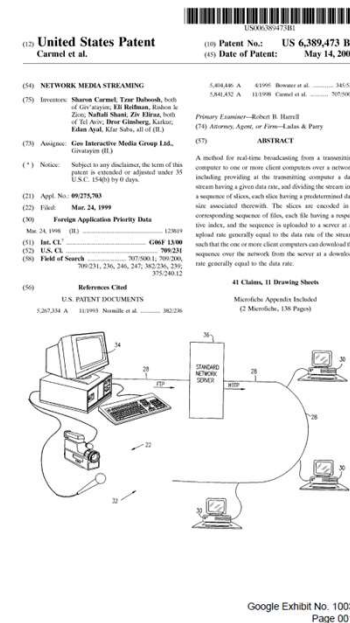
requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence **such that fluidity, video quality and audio quality are maintained** by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

utilizing optimized decoding processes to maintain quality playback.

'849 patent, cl. 1.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE



COMPUTATIONAL GRACEFUL DEGRADATION FOR VIDEO DECODING

Marcus Mattavelli, Ephraim Brannstrom
Integrated Systems Center, CS3,
Swiss Federal Institute of Technology,
Lausanne CH-1015, SWITZERLAND
Tel: +41 21 6930579; fax: +41 21 6930303
e-mail: Marcus.Mattavelli@epfl.ch

ABSTRACT

The implementation of software based video decoders in an asynchronous real-time system, where traditional dedicated hardware real-time systems. However, guaranteeing real-time performance, needed when processing video/audio bit-streams, by scheduling processing tasks of variable load and that might exceed the available resources, is an extremely difficult and in many cases an impossible challenge. In this paper we introduce the motivation and ideas of the Computational Graceful Degradation (CGD) approach for the implementation of video decoding. The goal is to provide results of possibly lower visual quality, but improving the real-time constraints, when the required processing time exceeds the available processing resources. This approach is very attractive since it enables a more efficient usage of the hardware processing power without needing to implement resources for worst case complexity decoding. We show how such technique can be applied to the decoding of compressed video sequences. Simulation results for MPEG-4 and H.263 video compression standards, during case of the increasing requirements and potentialities of CGD approach are also reported. The presented approach can be virtually applied to any video compression standard and processor based platform, although the best performance are achieved when complexity prediction information provided as side information to the video standard is available. This option has been included in the new MPEG-4 standard.

1. INTRODUCTION

The complexity of video decoding is a fascinating function of them [1]. By complexity we mean the processing time necessary to decode a compressed video bit-stream. First considering a specific processor platform, a specific compression standard, with a relative level and profile, the complexity of video decoding is a variable quantity. Dedicated hardware platforms can guarantee worst case complexity decoding. In contrast, in the case of programmable platforms such a guarantee would imply the allocation of a processing power up to

two orders of magnitude higher than the one actually needed for an average decoding [2]. Conversely, a theoretical average value is not useful for an efficient scheduling of a decoding task. The average value for each considered sequence is not constant, and its variation around the theoretical average one are so unpredictable that no useful scheduling policy can be obtained. The first straightforward option is a constant over-estimation of the processing resources. The second is to accept interruptions in the audio/video output streams when the needed decoding resources exceed the ones available in the system. Obviously both options are not desirable being respectively not economical, or providing a quality of the services which may really result progressively unacceptable. The main objective of the Computational Graceful Degradation (CGD) approach is to provide results of poorer quality when the results of the desirable quality cannot be obtained in real time. When the complexity of the task exceeds the available processing power, real-time services, possibly of degraded quality, are nevertheless guaranteed respecting all real-time constraints. This approach enables the correct scheduling of real-time tasks even when the computational system load fluctuates above the system capabilities.

The concept of computational graceful system degradation using improve results has previously been introduced in the field of real-time computing with application to control systems and automation. In these contexts imperative results are defined as intermediate results with the property that the more recent and immediate result is the closer one that approximates the final result (i.e. the idea only works in monotonically improving computations [2, 3, 4]). In these cases the problem to be solved are more concerned with final responses, i.e. the results, stability and accuracy to unexpected inputs, and the corresponding techniques are more concerned with the final results. In our case, the goal is to guarantee real-time video decoding with possible degradation of the image quality (reduction of the peak signal to noise ratio (PSNR), without losing parts of the image frames). Such a

1306
Google Exhibit No. 1004
Page 001

See also Petition at 49-50; Reply at 26;
Polish Decl. (Ex. 1002) at ¶¶206-208 51

Carmel + Mattavelli teaches “utilizing optimized decoding processes to maintain quality playback,” including maintaining “audio quality”

- Claim 1 (and claims 6 & 7) only require use of “optimized decoding processes”
 - Not optimized *video* decoding processes **and** optimized *audio* decoding processes
- Use of optimized *video* decoding processes can act to maintain fluidity, video quality, and audio quality

9. The method according to claim 1, wherein the object parameters and supplied identification of the first player received multimedia object define a sequence of similar multimedia objects and configure the player accordingly, using object parameters and **video decoding optimization, such that fluidity, video and audio quality are maintained.**

'849 patent, cl. 9

Carmel + Mattavelli teaches “utilizing optimized decoding processes to maintain quality playback,” including maintaining “audio quality”

- To the extent it has not forfeited this argument by failing to raise it until its Sur-Reply (*compare* POR at 42-47 with Sur-Reply at 18), PO’s assertion that Mattavelli fails to teach maintaining audio quality is also mistaken
- Mattavelli teaches its CGD techniques are beneficial to maintaining fluidity, video, and **audio** quality

Obviously for any Allocated Decoding Time the goal is to **respect the real-time constraints with the lowest degradation in the video/*audio* content.**

Mattavelli (Ex. 1004) at 2311-12

206 ... This is so because “audio compression is built on many of the same principles as video compression” (EX1020 at 322) and a person of ordinary skill in the art would have appreciated and found obvious that the same graceful degradation principles taught by Mattavelli would be beneficial to employ in circumstances where the audio decompression demands were high/complex but the decoder processing resources were constrained. ...

Polish Decl. (Ex.1002) at ¶206

Ground 2: Carmel + Wang

Cooley

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Ground 2 – Issue 1: Carmel + Wang teaches “optimized decoding processes to maintain quality playback” and maintaining “fluidity, video quality and audio quality”

1. A method comprising:

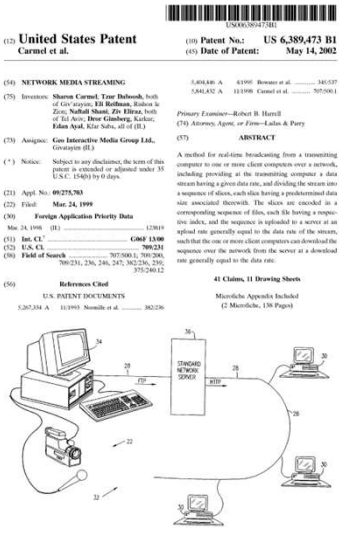
receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

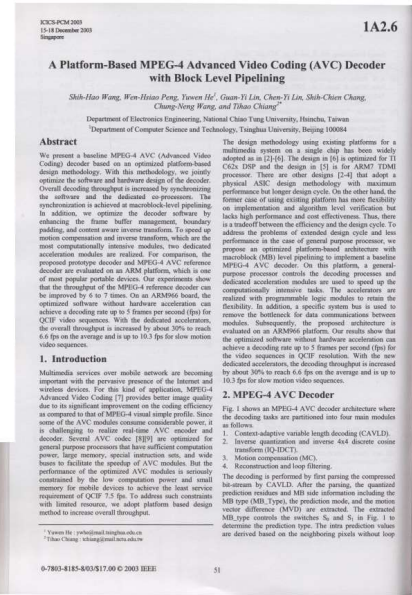
playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence **such that fluidity, video quality and audio quality are maintained** by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

utilizing **optimized decoding processes to maintain quality playback.**

’849 patent, cl. 1.



Google Exhibit No. 1003
Page 001



Google Exhibit No. 1007
Page 001

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 70-78; Reply at 24-26;
Polish Decl. (Ex. 1002) at ¶¶276-300 55

Wang teaches “optimized decoding processes to maintain quality playback” and maintaining “fluidity, video quality and audio quality”

280. Wang teaches a “platform-based MPEG-4 Advanced Video Coding (AVC) decoder with block level pipelining” that **optimizes** the decoding hardware and software so that resource-constrained mobile devices can decode **MPEG-4 AVC encoded data fast enough to utilize this compression standard and benefit from the higher image quality it can provide.**

Polish Decl. (Ex.1002) at ¶280

Multimedia services over mobile network are becoming important with the pervasive presence of the Internet and wireless devices. For this kind of application, MPEG-4 Advanced Video Coding [7] provides better image quality due to its significant improvement on the coding efficiency as compared to that of MPEG-4 visual simple profile. Since some of the AVC modules consume considerable power, it is challenging to realize real-time AVC encoder and decoder. Several AVC codec [8][9] are optimized for general purpose processors that have sufficient computation power, large memory, special instruction sets, and wide buses to facilitate the speedup of AVC modules. But the performance of the optimized AVC modules is seriously constrained by the low computation power and small memory for mobile devices to achieve the least service requirement of QCIF 7.5 fps. To address such constraints with limited resource, we adopt platform based design method to increase overall throughput.

Wang (Ex.1007) at 51

Wang teaches “optimized decoding processes”

We present a baseline MPEG-4 AVC (Advanced Video Coding) decoder based on an **optimized** platform-based design methodology. With this methodology, we jointly optimize the software and hardware design of the decoder. Overall decoding throughput is increased by synchronizing the software and the dedicated co-processors. The synchronization is achieved at macroblock-level pipelining. In addition, we **optimize** the decoder software by enhancing the frame buffer management, boundary padding, and content aware inverse transform. To speed up motion compensation and inverse transform, which are the most computationally intensive modules, two dedicated acceleration modules are realized. **For comparison, the proposed prototype decoder and MPEG-4 AVC reference decoder are evaluated on an ARM platform, which is one of most popular portable devices. Our experiments show that the throughput of the MPEG-4 reference decoder can be improved by 6 to 7 times. On an ARM966 board, the optimized software without hardware acceleration can achieve a decoding rate up to 5 frames per second (fps) for QCIF video sequences. With the dedicated accelerators, the overall throughput is increased by about 30% to reach 6.6 fps on the average and is up to 10.3 fps for slow motion video sequences.**

Wang (Ex.1007) at 51

4.3 Software Optimization

To retain the flexibility and portability across different platforms, our software optimization is mainly from algorithm level improvement. Our optimization techniques are as follows.

A. Frame buffer management

We improve the frame buffer management to save execution time for memory copy as follows.

1. The frame copy is done by swapping the pointers of frame buffers.
2. The motion compensated frame or intra prediction pixels are stored directly in the decoded frame buffer to avoid memory copy.
3. The decoded residuals are stored in the frame buffers with 4-byte alignment to match the 32-bit access architecture for ARM.

B. Boundary padding for motion compensation

Costly branching instructions for boundary checking can be saved if we perform padding in advance. Since unconstrained motion vector and the 6-tap filter [1, -5, 20, 20, -5, 1] are used for MC with half-pel interpolation, the locations of the reference pixels could exceed the frame boundaries. To avoid costly branching instructions, the reference frame are padded before MC. The padding length for the left and upper boundaries is (16+2) points while the padding length for the right and bottom boundaries is (16+3) points. In addition to padding, we also speed up the interpolation by replacing the multiplication for fixed filter tap weight with a table look-up.

C. Content aware IQ-IDCT

Due to correlation removal from inter frame prediction; most DCT coefficients after quantization are zero. Table 3 shows the percentage of all zero blocks in fast motion and slow motion sequences. For fast motion sequences, about 25% of DCT blocks are classified as the all zero case. For slow motion sequences, the percentage is increased to 47%. Besides, we also find that a large percentage of DCT blocks only have one DC coefficient. Thus, we can classify DCT blocks into three cases and conduct IQ-IDCT differently as follows:

Wang (Ex.1007) at 53-54

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 70-78; Reply at 24-26;
Polish Decl. (Ex. 1002) at ¶¶276-300 57

Patent Owner's Position

“Wang’s decoder is intended to improve throughput, not maintain quality playback.”

POR at 49

“Optimized decoding processes” increase decoding speed to provide and maintain quality playback

- PO draws a false distinction between improving throughput/decoding speed and maintaining quality playback (POR at 49-52)
- The '849 written description and PO's expert acknowledge the link between optimizations that increase decoding speed and maintaining quality playback

In cases where the device is unable to decode the video stream in real-time, **a faster but less accurate IDCT algorithm** can be used to process these residue values.

'849 Patent, 12:12-15

In addition to various arithmetic optimizations performed the video decoding process of the multimedia object players 10 and 11, shown in the flow diagram of FIG. 9, uses **faster motion compensation** without bilinear interpolation **when less quality but faster processing is required**.

'849 Patent, 12:25-29

21	A. Yes. So it says on page 10, and this is	10:45:46
22	paragraph 27: As used in the specification, the	10:45:52
23	optimizations -- in quotes -- taught therein all	10:46:07
24	seek to reduce the complexity of -- that is,	10:46:13
25	simplify -- the decoding processes, to maintain a	10:46:19
1	<u>smooth video stream, that otherwise could not be</u>	10:46:22
2	<u>completed in time using unabridged computations.</u>	10:46:25

Goldman 7/21/25 Tr. (Ex. 1080), 29:21-30:2

“Optimized decoding processes” increase decoding speed to provide and maintain quality playback

- PO’s expert acknowledges the link between optimizations that increase decoding speed and maintaining quality playback

21	So the complexity reducing is you're	10:33:44
22	simplifying the equation so that you can actually	10:33:47
23	do the calculations <u>in time to keep fluidity</u>	10:33:50
24	<u>moving</u> , with the understanding that a POSA would	10:33:53
25	have, is that if you didn't apply these and you	10:33:57
1	had bandwidth restrictions or the capability of	10:34:01
2	the device is such that it didn't have the	10:34:05
3	processing power or memory capability to do the	10:34:07
4	functions as described in the stream, that it	10:34:11
5	would not be able to maintain fluidity and	10:34:14
6	audio/video quality.	10:34:18

Goldman 7/21/25 Tr. (Ex. 1080), 22:21-23:6

“Optimized decoding processes” increase decoding speed to provide and maintain quality playback

Compare

'849 patent:

In FIG. 1 the multimedia object player 10 is a downloadable Java (J2M2) applet and **the described audio and video decoder optimizations and strategies**, FIGS. 7-13 and FIGS. 15-20, as applied to standard MPEG4 and AAC decoding **make it possible for the multimedia object player 10 to playback live music and video, at acceptable frame rates (5-15 fps), on limited, cell phone handsets.**

'849 Patent, 18:64-19:3

Wang:

In addition, we **optimize** the decoder software by enhancing the frame buffer management, boundary padding, and content aware inverse transform. To speed up motion compensation and inverse transform, which are the most computationally intensive modules, two dedicated acceleration modules are realized. **For comparison, the proposed prototype decoder and MPEG-4 AVC reference decoder are evaluated on an ARM platform, which is one of most popular portable devices. Our experiments show that the throughput of the MPEG-4 reference decoder can be improved by 6 to 7 times.** On an ARM966 board, the **optimized software** without hardware acceleration can achieve a decoding rate up to 5 frames per second (fps) for QCIF video sequences. **With the dedicated accelerators, the overall throughput is increased by about 30% to reach 6.6 fps on the average and is up to 10.3 fps for slow motion video sequences.**

Wang (Ex.1007) at 51

Carmel + Wang teaches “utilizing optimized decoding processes to maintain quality playback,” including maintaining “audio quality”

- Claim 1 (and claims 6 & 7) only require use of “optimized decoding processes”
 - Not optimized *video* decoding processes **and** optimized *audio* decoding processes
- Use of optimized *video* decoding processes can act to maintain fluidity, video quality, and audio quality

9. The method according to claim 1, wherein the object parameters and supplied identification of the first player received multimedia object define a sequence of similar multimedia objects and configure the player accordingly, using object parameters and **video decoding optimization, such that fluidity, video and audio quality are maintained.**

'849 patent, cl. 9

Carmel + Wang teaches “utilizing optimized decoding processes to maintain quality playback,” including maintaining “audio quality”

- Even if optimized audio decoding processes were required by the claims, *arguendo*, optimized audio decoding processes are taught and rendered obvious by Wolters (Ground 3)

Ground 2 – Issue 2: A POSA would have been motivated to combine Carmel and Wang’s teachings

1. A method comprising:

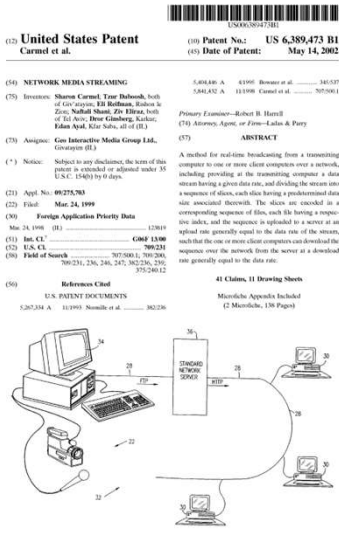
receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence **such that fluidity, video quality and audio quality are maintained** by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

utilizing **optimized decoding processes to maintain quality playback.**

’849 patent, cl. 1.



Google Exhibit No. 1003
Page 001



Google Exhibit No. 1007
Page 001

Patent Owner's Position

“Carmel seeks to implement media streaming over existing client hardware and software.”

POR at 53

“Wang on the other hand is directed to an improved hardware and software decoder at the client devices.”

POR at 54

“Implementing Wang’s decoder in Carmel would require upgrading each client’s personal computer or workstation to include the improved decoder hardware and software. That is, every customer of Carmel’s Internet service provider would have to buy new hardware with Wang’s prototype decoder architecture.”

POR at 54

A POSA would have been motivated to combine Carmel and Wang's teachings

- Carmel's teachings are applicable to a wide-variety of client computing devices—including mobile devices—as patent owner's expert recognized

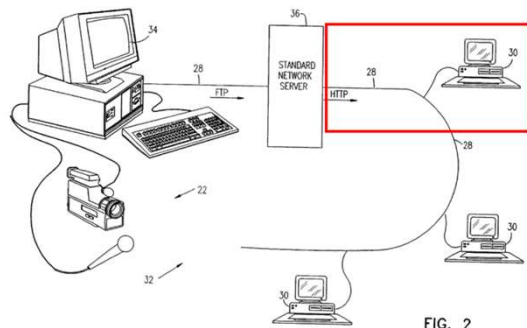


FIG. 2

Carmel (Ex.1003) at Fig. 2

The term “**network**” shall mean at least two computers connected for communications through communication lines which can be hard wired **or virtual, such as satellite communications....** Such a network consists of multiple computers ... all connected through telephone lines, **or other communication links both physical and wireless.**

Carmel '432 (Ex.1009) at 3:21-35

42. The '473 Patent also disclosed various rate-control techniques based on the local bandwidth throughput conditions (e.g., varying compression and the use of multiple streams at different quality levels),¹² thus addressing the issues of (a) **varying bandwidth conditions** (which at the time was impacted by, for example, dial-up connections and **mobile networks**), and (b) providing the best video quality at a fraction of the cost required by technologies known at the time. For at least these reasons, the '473 Patent's techniques were non-conventional at the time of the patent, and represented a clear departure from the way others in the field worked. *Implementing Adaptive HTTP Streaming Using the Web* (Exhibit E), p. 2.

Goldman Decl. from Carmel Reexam (Ex1081) at 11-12

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 29-32, 33-34; Reply at 27-28;
Polish Decl. (Ex. 1002) at ¶¶86-87, 160-164 66

A POSA would have been motivated to combine Carmel and Wang's teachings

- Carmel is not limited to using only what already exists on a client's device. Rather, Carmel expressly teaches providing a client with a Java applet or plug-in.

Similarly, because HTTP is supported by substantially all modern Web browsers, **clients 30 will typically need only add a Java applet or plug-in** to their existing Web browsers, as described further hereinbelow, **in order to receive and play back the broadcast.**

Carmel (Ex.1003) at 7:12-17

A POSA would have been motivated to combine Carmel and Wang's teachings

- ARM platform not specialized hardware – 75% of mobile device market at time
- Only one or some client(s) using an ARM hardware platform in a broadcasting system implementing Carmel's teachings would have been sufficient to motivate a POSA to combine Carmel and Wang's teachings

3.2 Proposed Architecture

Fig. 2 shows our emulation configuration to test our new architecture for the MPEG-4 AVC decoder. The proposed architecture is compliant with the MPEG-4 AVC baseline profile of level 1. Table 1 summarizes its main parameters. The architecture is emulated on an ARM development board. Since the ARM platform is one of the most typical portable devices and takes almost 75% of the market for the mobile devices. In addition, the ARM platform provides a general purpose ARM966 CPU core for data flow control and a logic module for multiple dedicated accelerators. The ARM also provides an industry standardized AHB for high-speed computation and emulations.

Wang (Ex.1007) at 52

A POSA would have been motivated to combine Carmel and Wang's teachings

- PO in no way shows that combining Carmel and Wang's teachings would render Carmel inoperable for its alleged "intended purpose"
- PO's challenge to the motivation to combine Carmel and Wang is an improper bodily incorporation argument.

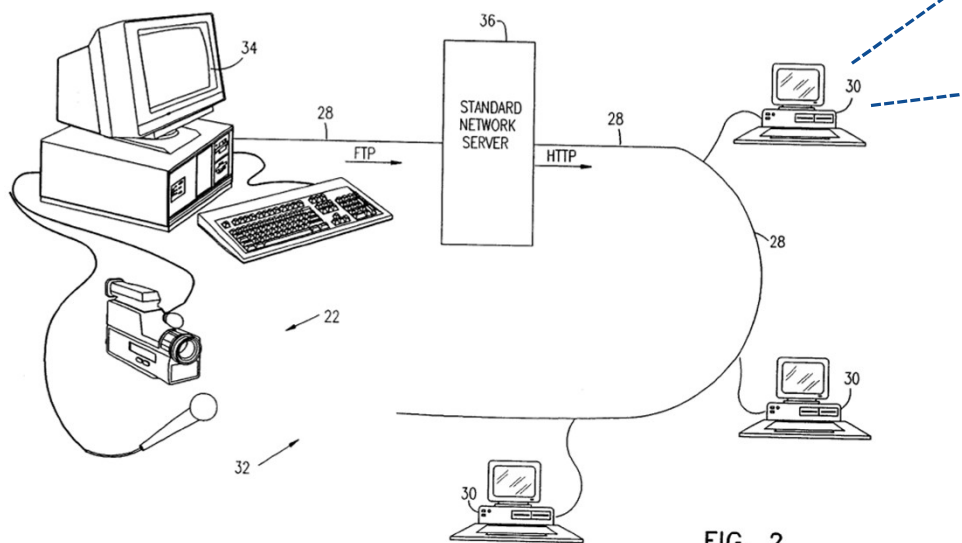


UNITED STATES COURT OF APPEALS FOR THE
FEDERAL CIRCUIT

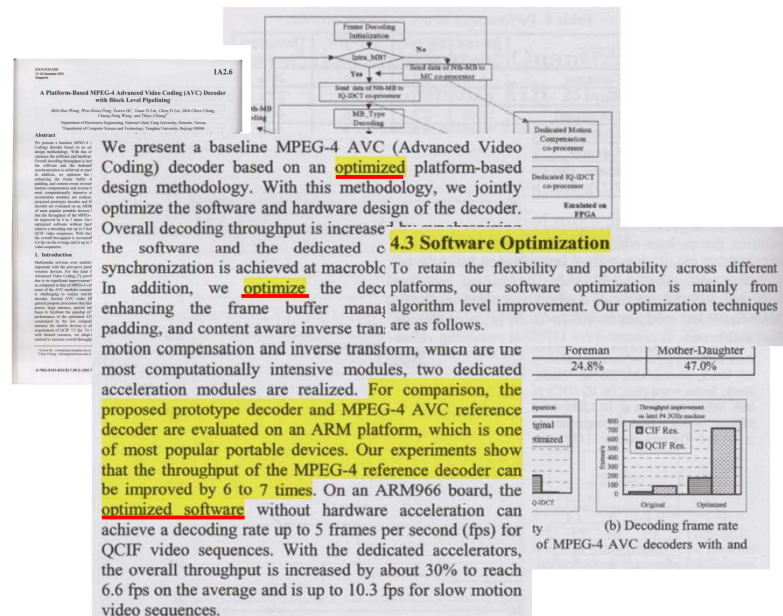
“The test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference; nor is it that the claimed invention must be expressly suggested in any one or all of the references. Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art.”

Innovative Memory Sys., Inc. v. Micron Tech., Inc., 784 F. App'x 771, 776–77 (Fed. Cir. 2019),
quoting *In re Keller*, 642 F.2d 413, 425 (CCPA 1981)

Carmel + Wang



Carmel (Ex.1003)



Wang (Ex.1007)

Other Issues

Cooley

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Additional Issue 1: Ground 3 – Wolters teaches optimized audio decoding processes for maintaining quality playback, including fluidity, video quality, and audio quality

United States Patent
Carmel et al.

Patent No. US 6,389,473 B1
Date of Patent: May 14, 2002

Network Media Streaming

Abstract: A method for real-time broadcasting from a transmitting computer to one or more client computers over a network, including providing at the transmitting computer a data stream having a given data rate, and dividing the stream into a sequence of slices, each slice having a predetermined data size associated therewith. The slices are encoded in a corresponding sequence of files, each file having a respective index, and the sequence is uploaded to a server at an upload rate generally equal to the data rate of the stream, such that the one or more client computers can download the sequence over the network from the server at a download rate generally equal to the data rate.

41 Claims, 11 Drawing Sheets

References Cited

U.S. Patent Documents

5,267,034 A 11/1995 Nicolaou et al. 362/236

Google Exhibit No. 1003
Page 001

Carmel (Ex.1003)

Platform-Based MPEG-4 Advanced Video Coding (AVC) Decoder with Block Level Pipelining

Shih-Hao Wang, Wen-Hsiao Peng, Tawen He*, Guan-Ti Lin, Chen-Ti Lin, Shih-Chien Chang

Abstract: The design methodology using existing platform's multimedia system on a single chip has been adopted in [1]-[3]. The design in [1] is optimized for Cx2x DSP and the design in [2] is for ARM7T processor. There are other designs [2]-[4] that add physical ASIC design methodology with multi-performance but longer design cycle. On the other hand, former case of using existing platform has more focus on implementation and algorithm level optimization, lacks high performance and cost effectiveness. Thus, is a tradeoff between the efficiency and the design cycle address the problems of crossbar design cycle and performance in the case of general purpose processors. propose an optimized platform-based architecture: multiblock (MB) level pipelining to implement a basic MPEG-4 AVC decoder. On this platform, a general purpose processor controls the decoding process. dedicated acceleration modules are used to speed up computationally intensive tasks. The accelerators realized with programmable logic modules to ensure flexibility. In addition, a specific system has a use remove the bottleneck for data communications between modules. Subsequently, the proposed architecture evaluated on an ARM7T processor. Our results show the optimized software without hardware acceleration achieve a decoding rate up to 5 frames per second (fps) the video sequences in QCIF resolution. With the dedicated accelerators, the decoding throughput is increased by about 30% to reach 6.6 fps on the average and is up to 10.3 fps for slow motion video sequences.

1. Introduction

Multimedia services over mobile network are becoming important with the pervasive presence of the Internet and wireless devices. For this kind of applications, MPEG-4 Advanced Video Coding (AVC) provides better image quality due to its significant improvement on the coding efficiency in comparison to that of MPEG-4 visual sample quality. Since some of the AVC modules consume considerable power, it is challenging to realize real-time AVC decoder and encoder. Several AVC code (H.264) are optimized for general purpose processors that have sufficient computation power, large memory, special instruction sets, and wide buses to facilitate the operating of AVC modules. But the performance of the optimized AVC modules is seriously constrained by the low computation power and small memory for mobile devices to achieve the least service requirement of QCIF 7.5 fps. To address such constraints with limited resources, we adopt platform-based design method to increase overall throughput.

2. MPEG-4 AVC Decoder

Fig. 1 shows an MPEG-4 AVC decoder architecture where the decoding tasks are partitioned into four main modules as follows:

1. Content-adaptive variable length decoding (CAVLD): binary quantization and inverse fast discrete cosine transform (IQ-DCT).
2. Motion compensation (MC).
3. Reconstruction and loop filtering.
4. The decoding is performed by first parsing the compressed bit-stream by CAVLD. After the parsing, the quantization prediction indices and MB side information including MB type (MB_Type), the prediction mode, and the one vector difference (MVD) are extracted. The MB_Type controls the switches S₀ and S₁ in Fig. 1. After the prediction type, the intra prediction values are derived based on the neighboring pixels without

Google Exhibit No. 1007
Page 001

Wang (Ex.1007)

Audio Engineering Society
Convention Paper 5871

Presented at the 115th Convention
2003 October 10-13 New York, NY, USA

A closer look into MPEG-4 High Efficiency AAC

Martin Wolters*, Kristofer Kjølting¹, Daniel Hoenen¹, Heiko Puschmann²

*Coding Technologies, Deutschherren 13-15, 90429 Nürnberg, Germany
¹Coding Technologies, Dillwegweg 66, 11557 Stockholm, Sweden
Correspondence should be addressed to Martin Wolters (wol@codingtechnologies.com)

ABSTRACT

MPEG Spectral Band Replication (SBR) is the newest compression technology available as part of the MPEG standards. It is combined with MPEG Advanced Audio Coding (AAC) and improves coding efficiency by more than 30%. The resulting scheme is called High-Efficiency AAC (HE-AAC). This presentation explains MPEG SBR and its integration into the existing MPEG-4 bitstream format. The SBR technology itself as well as the implications on systems based on MPEG-4 technology are described. Signaling through MPEG-4 Systems and other transport formats is introduced and typical applications and usage scenarios are listed.

1. INTRODUCTION

MPEG-4 High Efficiency AAC (HE-AAC) is not a replacement for AAC, but rather a supplement which extends the reach of high-quality MPEG-4 Audio to much lower bitrates. High Efficiency AAC decoders will decode both plain AAC and the enhanced AAC-plus SBR. The result is a backward compatible extension of the standard. Listening tests show that MPEG-4 HE-AAC offers a significant benefit over proprietary codecs and over AAC with extensions, and place it clearly as the most efficient audio codec in existence. Coding Technologies and its customers have been using this technology for a number

Google Exhibit No. 1041
Page 001

Wolters (Ex.1004)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 78-80; Reply at 28-29; Polish Decl. (Ex. 1002) at ¶¶301-304 72

Wolters teaches optimized audio decoding processes for maintaining quality playback, including fluidity, video quality, and audio quality

- Although not required by the Challenged Claims, Wolters teaches optimized **audio** decoding processes that also satisfy the maintaining playback quality and “fluidity, video quality and audio quality are maintained” limitations.

MPEG-4 HE AAC is a proven technology which has been widely deployed and is ready for use today. Reference decoder source code will soon be made available through MPEG and **optimized source code** for both encoders and decoders is available for license from companies such as Coding Technologies. **Optimized binary implementations** for Win32, Linux, MacOS X, ST Micro, ARM, Motorola, TI C64x & C55x, and Trimedia are presently available and other firmware implementations are being completed.

Wolters (Ex.1004) at 12

- A POSA would have been motivated to use MPEG-4 High Efficiency Advanced Audio Coding (HE-AAC) and combine Wolters’s teachings with those of Carmel and Wang

304. ... A person of ordinary skill in the art would have understood and been motivated to also use HE AAC audio coding, as taught by Wolters, for the same reason—that is because it provides high compression efficiency allowing transmission of high quality audio using less data, and by employing optimized decoding processes, a person of ordinary skill in the art would **maximize the available processing resources to deliver the highest quality possible/practicable**. ... (EX1041 at 13 (“By coupling HE AAC with MPEG-4 Video, more bits can be allocated to the video signal without degrading the quality of the audio signal.”)).

Polish Decl. (Ex. 1002) at ¶304

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 78-80; Reply at 28-29;
Polish Decl. (Ex. 1002) at ¶¶301-304 73

Wolters teaches optimized audio decoding processes for maintaining quality playback, including fluidity, video quality, and audio quality

- PO's critique of Ground 3: (i) HE-AAC provides two variants, a high-quality (HQ) variant and a low-power (LP) variant; (ii) only the LP variant would be appropriate for mobile devices at the time; (iii) the LP variant purportedly does not maintain quality, but reduces quality (*compared to the HQ variant*) (See POR at 55-56)
- Inconsistent with how PO has interpreted the '849 patent and claims outside of this IPR:

4 A. So the way that the patent is written, it 10:32:41
5 talks about maintaining quality. So it's not a 10:32:46
6 baseline against an absolute value amount. It's a 10:32:50
7 baseline of what can be done in relationship to a 10:32:54
8 few points that are very clearly defined in the 10:32:59
9 patent. It is the network bandwidth and is the 10:33:02
10 processing power and memory capabilities of the 10:33:06
11 device. 10:33:10

Goldman 7/21/25 Tr. (Ex. 1080), 22:4-11

20 Again, it's not an absolute quality that's 10:41:10
21 being measured here. It's to maintain the best 10:41:12
22 quality you can, given network bandwidth and other 10:41:15
23 network impairments and the capability of the 10:41:19
24 device itself. 10:41:21

Goldman 7/21/25 Tr. (Ex. 1080), 27:20-24

- PO again improperly attacks the references individually and fails to address the proposed combination

“Nor does the Petition identify anything in Wolters that addresses maintaining fluidity or video quality, both of which are required by claim element 1[c].

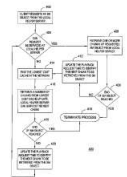
POR at 56

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 78-80; Reply at 28-29;
Polish Decl. (Ex. 1002) at ¶¶301-304 74

Additional Issue 2: Grounds 4-5 address limitations that PO does not dispute; and the Petition explains why a POSA would have been motivated to combine the references

- (12) **United States Patent**
Gao et al.
- (13) **Patent No.:** US 7,938,312 B1
Jul. 8, 2013
- (14) **METHOD AND SYSTEM FOR CATCHING**
UNDESIRABLE E-MAILS
Gao et al. 20090000000
- (15) **Inventory:**
Markus A. Hoffmann, Tübingen, DE (US)
Markus A. Hoffmann, Tübingen, DE (US)
Toshiba Electronics Europe Ltd., Farnborough, UK
IBM Research, Farnborough, UK (US)
Toshiba Electronics Europe Ltd., Farnborough, UK (US)
- (16) **Assignee:** Toshiba Electronics Europe Ltd., Farnborough, UK (US)
- (17) **Attorney:** *See below*
- (18) **Abstract:** A method to dynamically detect the time of the receipt of e-mail and decide whether the receipt of the e-mail is a spam message.
- (19) **Appl. No.:** 2008/0036180
- (20) **Publ. No.:** 2008/0036180
- (21) **Int. Cl.:** H04L 29/06
H04L 29/12
H04L 29/18
H04L 29/22
H04L 29/24
H04L 29/28
H04L 29/30
H04L 29/32
H04L 29/34
H04L 29/36
H04L 29/38
H04L 29/40
H04L 29/42
H04L 29/44
H04L 29/46
H04L 29/48
H04L 29/50
H04L 29/52
H04L 29/54
H04L 29/56
H04L 29/58
H04L 29/60
H04L 29/62
H04L 29/64
H04L 29/66
H04L 29/68
H04L 29/70
H04L 29/72
H04L 29/74
H04L 29/76
H04L 29/78
H04L 29/80
H04L 29/82
H04L 29/84
H04L 29/86
H04L 29/88
H04L 29/90
H04L 29/92
H04L 29/94
H04L 29/96
H04L 29/98
H04L 30/00
H04L 30/02
H04L 30/04
H04L 30/06
H04L 30/08
H04L 30/10
H04L 30/12
H04L 30/14
H04L 30/16
H04L 30/18
H04L 30/20
H04L 30/22
H04L 30/24
H04L 30/26
H04L 30/28
H04L 30/30
H04L 30/32
H04L 30/34
H04L 30/36
H04L 30/38
H04L 30/40
H04L 30/42
H04L 30/44
H04L 30/46
H04L 30/48
H04L 30/50
H04L 30/52
H04L 30/54
H04L 30/56
H04L 30/58
H04L 30/60
H04L 30/62
H04L 30/64
H04L 30/66
H04L 30/68
H04L 30/70
H04L 30/72
H04L 30/74
H04L 30/76
H04L 30/78
H04L 30/80
H04L 30/82
H04L 30/84
H04L 30/86
H04L 30/88
H04L 30/90
H04L 30/92
H04L 30/94
H04L 30/96
H04L 30/98
H04L 31/00
H04L 31/02
H04L 31/04
H04L 31/06
H04L 31/08
H04L 31/10
H04L 31/12
H04L 31/14
H04L 31/16
H04L 31/18
H04L 31/20
H04L 31/22
H04L 31/24
H04L 31/26
H04L 31/28
H04L 31/30
H04L 31/32
H04L 31/34
H04L 31/36
H04L 31/38
H04L 31/40
H04L 31/42
H04L 31/44
H04L 31/46
H04L 31/48
H04L 31/50
H04L 31/52
H04L 31/54
H04L 31/56
H04L 31/58
H04L 31/60
H04L 31/62
H04L 31/64
H04L 31/66
H04L 31/68
H04L 31/70
H04L 31/72
H04L 31/74
H04L 31/76
H04L 31/78
H04L 31/80
H04L 31/82
H04L 31/84
H04L 31/86
H04L 31/88
H04L 31/90
H04L 31/92
H04L 31/94
H04L 31/96
H04L 31/98
H04L 32/00
H04L 32/02
H04L 32/04
H04L 32/06
H04L 32/08
H04L 32/10
H04L 32/12
H04L 32/14
H04L 32/16
H04L 32/18
H04L 32/20
H04L 32/22
H04L 32/24
H04L 32/26
H04L 32/28
H04L 32/30
H04L 32/32
H04L 32/34
H04L 32/36
H04L 32/38
H04L 32/40
H04L 32/42
H04L 32/44
H04L 32/46
H04L 32/48
H04L 32/50
H04L 32/52
H04L 32/54
H04L 32/56
H04L 32/58
H04L 32/60
H04L 32/62
H04L 32/64
H04L 32/66
H04L 32/68
H04L 32/70
H04L 32/72
H04L 32/74
H04L 32/76
H04L 32/78
H04L 32/80
H04L 32/82
H04L 32/84
H04L 32/86
H04L 32/88
H04L 32/90
H04L 32/92
H04L 32/94
H04L 32/96
H04L 32/98
H04L 33/00
H04L 33/02
H04L 33/04
H04L 33/06
H04L 33/08
H04L 33/10
H04L 33/12
H04L 33/14
H04L 33/16
H04L 33/18
H04L 33/20
H04L 33/22
H04L 33/24
H04L 33/26
H04L 33/28
H04L 33/30
H04L 33/32
H04L 33/34
H04L 33/36
H04L 33/38
H04L 33/40
H04L 33/42
H04L 33/44
H04L 33/46
H04L 33/48
H04L 33/50
H04L 33/52
H04L 33/54
H04L 33/56
H04L 33/58
H04L 33/60
H04L 33/62
H04L 33/64
H04L 33/66
H04L 33/68
H04L 33/70
H04L 33/72
H04L 33/74
H04L 33/76
H04L 33/78
H04L 33/80
H04L 33/82
H04L 33/84
H04L 33/86
H04L 33/88
H04L 33/90
H04L 33/92
H04L 33/94
H04L 33/96
H04L 33/98
H04L 34/00
H04L 34/02
H04L 34/04
H04L 34/06
H04L 34/08
H04L 34/10
H04L 34/12
H04L 34/14
H04L 34/16
H04L 34/18
H04L 34/20
H04L 34/22
H04L 34/24
H04L 34/26
H04L 34/28
H04L 34/30
H04L 34/32
H04L 34/34
H04L 34/36
H04L 34/38
H04L 34/40
H04L 34/42
H04L 34/44
H04L 34/46
H04L 34/48
H04L 34/50
H04L 34/52
H04L 34/54
H04L 34/56
H04L 34/58
H04L 34/60
H04L 34/62
H04L 34/64
H04L 34/66
H04L 34/68
H04L 34/70
H04L 34/72
H04L 34/74
H04L 34/76
H04L 34/78
H04L 34/80
H04L 34/82
H04L 34/84
H04L 34/86
H04L 34/88
H04L 34/90
H04L 34/92
H04L 34/94
H04L 34/96
H04L 34/98
H04L 35/00
H04L 35/02
H04L 35/04
H04L 35/06
H04L 35/08
H04L 35/10
H04L 35/12
H04L 35/14
H04L 35/16
H04L 35/18
H04L 35/20
H04L 35/22
H04L 35/24
H04L 35/26
H04L 35/28
H04L



Google Exhibit No. 1051
Page 001

Guo (Ex.1051)

US2007099389B1

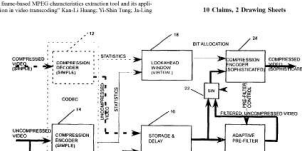
(c) **United States Patent**
Yu et al.

(16) Patent No.: **US 7,999,389 B1**
Date of Patent: **Aug. 29, 2006**

(51) **CLASS. CONC'D WITH PICTURE-BASED LOOKUP-WAY INFORMATION**

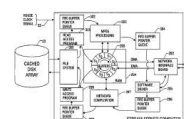
(54) **Picture Data: Displacement**
Yu, Pei-Ing; Duan, Hsin-Chia; Chang, Jui-Hsin
Intel Corporation, et al., Inventors
Intel Corporation, et al., Assignors

(52) **U.S. CLASS.**
359/224, 359/225, 359/226, 359/227, 359/228, 359/229, 359/230, 359/231, 359/232, 359/233, 359/234, 359/235, 359/236, 359/237, 359/238, 359/239, 359/240, 359/241, 359/242, 359/243, 359/244, 359/245, 359/246, 359/247, 359/248, 359/249, 359/250, 359/251, 359/252, 359/253, 359/254, 359/255, 359/256, 359/257, 359/258, 359/259, 359/260, 359/261, 359/262, 359/263, 359/264, 359/265, 359/266, 359/267, 359/268, 359/269, 359/270, 359/271, 359/272, 359/273, 359/274, 359/275, 359/276, 359/277, 359/278, 359/279, 359/280, 359/281, 359/282, 359/283, 359/284, 359/285, 359/286, 359/287, 359/288, 359/289, 359/290, 359/291, 359/292, 359/293, 359/294, 359/295, 359/296, 359/297, 359/298, 359/299, 359/300, 359/301, 359/302, 359/303, 359/304, 359/305, 359/306, 359/307, 359/308, 359/309, 359/310, 359/311, 359/312, 359/313, 359/314, 359/315, 359/316, 359/317, 359/318, 359/319, 359/320, 359/321, 359/322, 359/323, 359/324, 359/325, 359/326, 359/327, 359/328, 359/329, 359/330, 359/331, 359/332, 359/333, 359/334, 359/335, 359/336, 359/337, 359/338, 359/339, 359/340, 359/341, 359/342, 359/343, 359/344, 359/345, 359/346, 359/347, 359/348, 359/349, 359/350, 359/351, 359/352, 359/353, 359/354, 359/355, 359/356, 359/357, 359/358, 359/359, 359/360, 359/361, 359/362, 359/363, 359/364, 359/365, 359/366, 359/367, 359/368, 359/369, 359/370, 359/371, 359/372, 359/373, 359/374, 359/375, 359/376, 359/377, 359/378, 359/379, 359/380, 359/381, 359/382, 359/383, 359/384, 359/385, 359/386, 359/387, 359/388, 359/389, 359/390, 359/391, 359/392, 359/393, 359/394, 359/395, 359/396, 359/397, 359/398, 359/399, 359/400, 359/401, 359/402, 359/403, 359/404, 359/405, 359/406, 359/407, 359/408, 359/409, 359/410, 359/411, 359/412, 359/413, 359/414, 359/415, 359/416, 359/417, 359/418, 359/419, 359/420, 359/421, 359/422, 359/423, 359/424, 359/425, 359/426, 359/427, 359/428, 359/429, 359/430, 359/431, 359/432, 359/433, 359/434, 359/435, 359/436, 359/437, 359/438, 359/439, 359/440, 359/441, 359/442, 359/443, 359/444, 359/445, 359/446, 359/447, 359/448, 359/449, 359/450, 359/451, 359/452, 359/453, 359/454, 359/455, 359/456, 359/457, 359/458, 359/459, 359/460, 359/461, 359/462, 359/463, 359/464, 359/465, 359/466, 359/467, 359/468, 359/469, 359/470, 359/471, 359/472, 359/473, 359/474, 359/475, 359/476, 359/477, 359/478, 359/479, 359/480, 359/481, 359/482, 359/483, 359/484, 359/485, 359/486, 359/487, 359/488, 359/489, 359/490, 359/491, 359/492, 359/493, 359/494, 359/495, 359/496, 359/497, 359/498, 359/499, 359/500, 359/501, 359/502, 359/503, 359/504, 359/505, 359/506, 359/507, 359/508, 359/509, 359/510, 359/511, 359/512, 359/513, 359/514, 359/515, 359/516, 359/517, 359/518, 359/519, 359/520, 359/521, 359/522, 359/523, 359/524, 359/525, 359/526, 359/527, 359/528, 359/529, 359/530, 359/531, 359/532, 359/533, 359/534, 359/535, 359/536, 359/537, 359/538, 359/539, 359/540, 359/541, 359/542, 359/543, 359/544, 359/545, 359/546, 359/547, 359/548, 359/549, 359/550, 359/551, 359/552, 359/553, 359/554, 359/555, 359/556, 359/557, 359/558, 359/559, 359/560, 359/561, 359/562, 359/563, 359/564, 359/565, 359/566, 359/567, 359/568, 359/569, 359/570, 359/571, 359/572, 359/573, 359/574, 359/575, 359/576, 359/577, 359/578, 359/579, 359/580, 359/581, 359/582, 359/583, 359/584, 359/585, 359/586, 359/587, 359/588, 359/589, 359/590, 359/591, 359/592, 359/593, 359/594, 359/595, 359/596, 359/597, 359/598, 359/599, 359/600, 359/601, 359/602, 359/603, 359/604, 359/605, 359/606, 359/607, 359/608, 359/609, 359/610, 359/611, 359/612, 359/613, 359/614, 359/615, 359/616, 359/617, 359/618, 359/619, 359/620, 359/621, 359/622, 359/623, 359/624, 359/625, 359/626, 359/627, 359/628, 359/629, 359/630, 359/631, 359/632, 359/633, 359/634, 359/635, 359/636, 359/637, 359/638, 359/639, 359/640, 359/641, 359/642, 359/643, 359/644, 359/645, 359/646, 359/647, 359/648, 359/649, 359/650, 359/651, 359/652, 359/653, 359/654, 359/655, 359/656, 359/657, 359/658, 359/659, 359/660, 359/661, 359/662, 359/663, 359/664, 359/665, 359/666, 359/667, 359/668, 359/669, 359/670, 359/671, 359/672, 359/673, 359/674, 359/675, 359/676, 359/677, 359/678, 359/679, 359/680, 359/681, 359/682, 359/683, 359/684, 359/685, 359/686, 359/687, 359/688, 359/689, 359/690, 359/691, 359/692, 359/693, 359/694, 359/695, 359/696, 359/697, 359/698, 359/699, 359/700, 359/701, 359/702, 359/703, 359/704, 359/705, 359/706, 359/707, 359/708, 359/709, 359/710, 359/711, 359/712, 359/713, 359/714, 359/715, 359/716, 359/717, 359/718, 359/719, 359/720, 359/721, 359/722, 359/723, 359/724, 359/725, 359/726, 359/727, 359/728, 359/729, 359/730, 359/731, 359/732, 359/733, 359/734, 359/735, 359/736, 359/737, 359/738, 359/739, 359/740, 359/741, 359/742, 359/743, 359/744, 359/745, 359/746, 359/747, 359/748, 359/749, 359/750, 359/751, 359/752, 359/753, 359/754, 359/755, 359/756, 359/757, 359/758, 359/759, 359/760, 359/761, 359/762, 359/763, 359/764, 359/765, 359/766, 359/767, 359/768, 359/769, 359/770, 359/771, 359/772, 359/773, 359/774, 359/775, 359/776, 359/777, 359/778, 359/779, 359/780, 359/781, 359/782, 359/783, 359/784, 359/785, 359/786, 359/787, 359/788, 359/789, 359/790, 359/791, 359/792, 359/793, 359/794, 359/795, 359/796, 359/797, 359/798, 359/799, 359/800, 359/801, 359/802, 359/803, 359/804, 359/805, 359/806, 359/807, 359/808, 359/809, 359/810, 359/811, 359/812, 359/813, 359/814, 359/815, 359/816, 359/817, 359/818, 359/819, 359/820, 359/821, 359/822, 359/823, 359/824, 359/825, 359/826, 359/827, 359/828, 359/829, 359/830, 359/831, 359/832, 359/833, 359/834, 359/835, 359/836, 359/837, 359/838, 359/839, 359/840, 359/841, 359/842, 359/843, 359/844, 359/845, 359/846, 359/847, 359/848, 359/849, 359/850, 359/851, 359/852, 359/853, 359/854, 359/855, 359/856, 359/857, 359/858, 359/859, 359/860, 359/861, 359/862, 359/863, 359/864, 359/865, 359/866, 359/867, 359/868, 359/869, 359/870, 359/871, 359/872, 359/873, 359/874, 359/875, 359/876, 359/877, 359/878, 359/879, 359/880, 359/881, 359/882, 359/883, 359/884, 359/885, 359/886, 359/887, 359/888, 359/889, 359/890, 359/891, 359/892, 359/893, 359/894, 359/895, 359/896, 359/897, 359/898, 359/899, 359/900, 359/901, 359/902, 359/903, 359/904, 359/905, 359/906, 359/907, 359/908, 359/909, 359/910, 359/911, 359/912, 359/913, 359/914, 359/915, 359/916, 359/917, 359/918, 359/919, 359/920, 359/921, 359/922, 359/923, 359/924, 359/925, 359/926, 359/927, 359/928, 359/929, 359/930, 359/931, 359/932, 359/933, 359/934, 359/935, 359/936, 359/937, 359/938, 359/939, 359/940, 359/941, 359/942, 359/943, 359/944, 359/945, 359/946, 359/947, 359/948, 359/949, 359/950, 359/951, 359/952, 359/953, 359/954, 359/955, 359/956, 359/957, 359/958, 359/959, 359/960, 359/961, 359/962, 359/963, 359/964, 359/965, 359/966, 359/967, 359/968, 359/969, 359/970, 359/971, 359/972, 359/973, 359/974, 359/975, 359/976, 359/977, 359/978, 359/979, 359/980, 359/981, 359/982, 359/983, 359/984, 359/985, 359/986, 359/987, 359/988, 359/989, 359/990, 359/991, 359/992, 359/993, 359/994, 359/995, 359/996, 359/997, 359/998, 359/999, 359/1000, 359/1001, 359/1002, 359/1003, 359/1004, 359/1005, 359/1006, 359/1007, 359/1008, 359/1009, 359/1010, 359/1011, 359/1012, 359/1013, 359/1014, 359/1015, 359/1016, 359/1017, 359/1018, 359/1019, 359/1020, 359/1021, 359/1022, 359/1023, 359/1024, 359/1025, 359/1026, 359/1027, 359/1028, 359/1029, 359/1030, 359/1031, 359/1032, 359/1033, 359/1034, 359/1035, 359/1036, 359/1037, 359/1038, 359/1039, 359/1040, 359/1041, 359/1042, 359/1043, 359/1044, 359/1045, 359/1046, 359/1047, 359/1048, 359/1049, 359/1050, 359/1051, 359/1052, 359/1053, 359/1054, 359/1055, 359/1056, 359/1057, 359/1058, 359/1059, 359/1060, 359/1061, 359/1062, 359/1063, 359/1064, 359/1065, 359/1066, 359/1067, 359/1068, 359/1069, 359/1070, 359/1071, 359/1072, 359/1073, 359/1074, 359/1075, 359/1076, 359/1077, 359/1078, 359/1079, 359/1080, 359/1081, 359/1082, 359/1083, 359/1084, 359/1085, 359/1086, 359/1087, 359/1088, 359/1089, 359/1090, 359/1091, 359/1092, 359/1093, 359/1094, 359/1095, 359/1096, 359/1097, 359/1098, 359/1099, 359/1100, 359/1101, 359/1102, 359/1103, 359/1104, 359/1105, 359/1106, 359/1107, 359/1108, 359/1109, 359/1110, 359/1111, 359/1112, 359/1113, 359/1114, 359/1115, 359/1116, 359/1117, 359/1118, 359/1119, 359/1120, 359/1121, 359/1122, 359/1123, 359/1124, 359/1125, 359/1126, 359/1127, 359/1128, 359/1129, 359/1130, 359/1131, 359/1132, 359/1133, 359/1134, 359/1135, 359/1136, 359/1137, 359/1138, 359/1139, 359/1140, 359/1141, 359/1142, 359/1143, 359/1144, 359/1145, 359/1146, 359/1147, 359/1148, 359/1149, 359/1150, 359/1151, 359/1152, 359/1153, 359/1154, 359/1155, 359/1156, 359/1157, 359/1158, 359/1159, 359/1160, 359/1161, 359/1162, 359/1163, 359/1164, 359/1165, 359/1166, 359/1167, 359/1168, 359/1169, 359/1170, 359/1171, 359/1172, 359/1173, 359/1174, 359/1175, 359/1176, 359/1177, 359/1178, 359/1179, 359/1180, 359/1181, 359/1182, 359/1183, 359/1184, 359/1185, 359/1186, 359/1187, 359/1188, 359/1189, 359/1190, 359/1191, 359/1192, 359/1193, 359/1194, 359/1195, 359/1196, 359/1197, 359/1198, 359/1199, 359/1200, 359/1201, 359/1202, 359/1203, 359/1204, 359/1205, 359/1206, 359/1207, 359/1208, 359/1209, 359/1210, 359/1211, 359/1212, 359/1213, 359/1214, 359/1215, 359/1216, 359/1217, 359/1218, 359/1219, 359/1220, 359/1221, 359/1222, 359/1223, 359/1224, 359/1225, 359/1226, 359/1227, 359/1228, 359/1229, 359/1230, 359/1231, 359/1232, 359/1233, 359/1234, 359/1235, 359/1236, 359/1237, 359/1238, 359/1239, 359/1240, 359/1241, 359/1242, 359/1243, 359/1244, 359/1245, 359/1246, 359/1247, 359/1248, 359/1249, 359/1250, 359/1251, 359/1252, 359/1253, 359/1254, 359/1255, 359/1256, 359/1257, 359/1258, 359/1259, 359/1260, 359/1261, 359/1262, 359/1263, 359/1264, 359/1265, 359/1266, 359/1267, 359/1268, 359/1269, 359/1270, 359/1271, 359/1272, 359/1273, 359/1274, 359/1275, 359/1276, 359/1277, 359/1278, 359/1279, 359/1280, 359/1281, 359/1282, 359/1283, 359/1284, 359/1285, 359/1286, 359/1287, 359/1288, 359/1289, 359/1290, 359/1291, 359/1292, 359/1293, 359/1294, 359/1295, 359/1296, 359/1297, 359/1298, 359/1299, 359/1300, 359/1301, 359/1302, 359/1303, 359/1304, 359/1305, 359/1306, 359/1307, 359/1308, 359/1309, 359/1310, 359/1311, 359/1312, 359/1313, 359/1314, 359/1315, 359/1316, 359/1317, 359/1318, 359/1319, 359/1320, 359/1321, 359/1322, 359/1323, 359/1324, 359/1325, 359/1326, 359/1327, 359/1328, 359/1329, 359/1330, 359/1331, 359/1332, 359/1333, 359/1334, 359/1335, 359/1336, 359/1337, 359/1338, 359/1339, 359/1340, 359/1341, 359/1342, 359/1343, 359/1344, 359/1345, 359/1346, 359/1347, 359/1348, 359/1349, 359/1350, 359/1351, 359/1352, 359/1353, 359/1354, 359/1355, 359/1356, 359/1357, 359/1358, 359/1359, 359/1360, 359/1361, 359/1362, 359/1363, 359/1364, 359/1365, 359/1366, 359/1367, 359/1368, 359/1369, 359/1370, 359/1371, 359/1372, 359/1373, 359/1374, 359/1375, 359/1376, 359/1377, 359/1378, 359/1379, 359/1380, 359/1381, 359/1382, 359/1383, 359/1384, 359/1385, 359/1386, 359/1387, 359/1388, 359/1389, 359/1390, 359/1391, 359/1392, 359/1393, 359/1394, 359/1395, 359/1396, 359/1397, 359/1398, 359/1399, 359/1400, 359/1401, 359/1402, 359/1403, 359/1404, 359/1405, 359/1406, 359/1407, 359/1408, 359/1409, 359/1410, 359/1411, 359/1412, 359/1413, 359/1414, 359/1415, 359/1416, 359/1417, 359/1418, 359/1419, 359/1420, 359/1421, 359/1422, 359/1423, 359/1424, 359/1425, 359/1426, 359/1427, 359/1428, 359/1429, 359/1430, 359/1431, 359/1432, 359/1433, 359/1434, 359/1435, 359/1436, 359/1437, 359/1438, 359/1439, 359/1440, 359/1441, 359/1442, 359/1443, 359/1444, 359/1445, 359/1446, 359/1447, 359/1448, 359/1449, 359/1450, 359/1451, 359/1452, 359/1453, 359/1454, 359/1455, 359/1456, 359/1457, 359/1458, 359/1459, 359/1460, 359/1461, 359/1462, 359/1463, 359/1464, 359/1465, 359/1466, 359/1467, 359/1468, 359/1469, 359/1470, 359/1471, 359/1472, 359/1473, 359/1474, 359/1475, 359/1476, 359/1477, 359/1478, 359/1479, 359/1480, 359/1481, 359/1482, 359/1483, 359/1484, 359/1485, 359/1486, 359/1487, 359/1488, 359/1489, 359/1490, 359/1491, 359/1492, 359/1493, 359/1494, 359/1495, 359/1496, 359/1497, 359/1498, 359/1499, 359/1500, 359/1501, 359/1502, 359/15



Google Exhibit No. 1061
Page 001

Yu (Ex.1061)

[illegible]

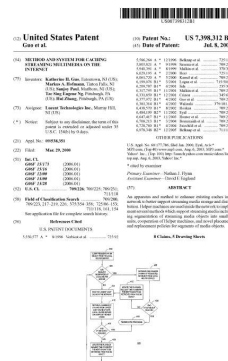
Google Exhibit No. 1016
Page 001

Forecast (Ex.1016)

See *also* Petition at 80-86; Reply at 29;
Polish Decl. (Ex. 1002) at ¶¶305-315 75

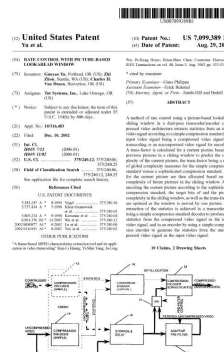
Grounds 4-5 address limitations that PO does not dispute

- Grounds 4 & 5 address limitations that PO does not dispute are satisfied by Carmel + Mattavelli (Ground 1) and/or Carmel + Wang (Ground 2)
 - Guo (Ex. 1051) – multiple servers to stream multimedia content – (possibly) element 1[b.ii]
 - Yu (Ex. 1061) – transcoding compressed multimedia to a different standard – claims 3 & 8
 - Forecast (Ex. 1016) – splitting compressed video at an I-frame – claims 4 & 5



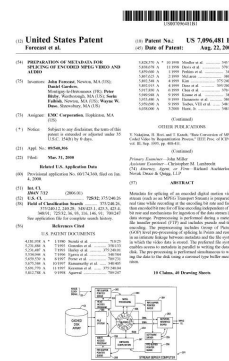
Google Exhibit No. 1051
Page 001

Guo (Ex.1051)



Google Exhibit No. 1061
Page 001

Yu (Ex.1061)



Google Exhibit No. 1016
Page 001

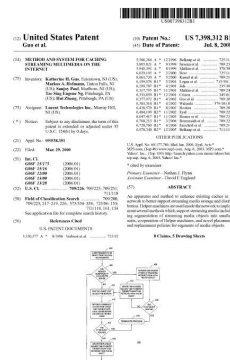
Forecast (Ex.1016)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 80-86; Reply at 29;
Polish Decl. (Ex. 1002) at ¶¶305-315 76

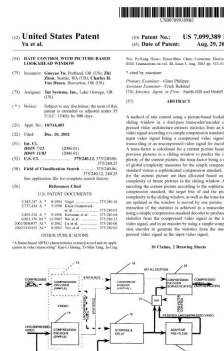
Grounds 4-5: the Petition explains why a POSA would have been motivated to combine the proposed teachings

- The teachings from Guo (multiple servers), Yu (transcoding), and Forecast (splitting at an I-frame) concern only *server-side* preparation and broadcasting of a multimedia stream
- The Petition appropriately explains why a POSA would have been motivated to combine these teachings with Carmel's teachings regarding server-side preparation and broadcasting of a multimedia stream (See Petition at 80-86)



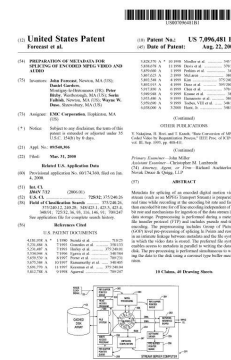
Google Exhibit No. 1051
Page 001

Guo (Ex.1051)



Google Exhibit No. 1061
Page 001

Yu (Ex.1061)



Google Exhibit No. 1016
Page 001

Forecast (Ex.1016)

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 80-86; Reply at 29;
Polish Decl. (Ex. 1002) at ¶¶305-315 77

Additional Issue 3: Petitioner’s claim construction positions before the Board and the district court are consistent and justified; PO’s positions are not

Petition for *Inter Partes* Review of
U.S. Patent No. 10,771,849 B2

IPR2025-00775
U.S. Pat. No. 10,771,849

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Google LLC,
Petitioner,

Google LLC,
Petitioner,

v.

v.

Withrow Networks Inc.,
Patent Owner

Withrow Networks Inc.,
Patent Owner

IPR2025-00775

Case No. IPR2025-00775
U.S. Patent No. 10,771,849

U.S. Patent No. 10,771,849 B2
Issue Date: September 8, 2020

PATENT OWNER RESPONSE

Title: MULTIMEDIA SYSTEM FOR MOBILE CLIENT PLATFORMS

PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 10,771,849 B2

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 45-46 & n.8; Reply at 30-31;

Petitioner's claim construction positions before the Board and the district court are consistent and justified

- PO has forfeited its institution-related argument by failing to raise it in its Request for Discretionary Denial, Patent Owner Preliminary Response, or Request for Director Review of the Institution decision
- The Petition explained that, while the outer bounds of certain terms (“optimized” “maintain quality playback”) of the challenged claims were not adequately defined, a POSA would nonetheless have understood the teachings from the asserted references to fall within the core scope of these terms.

Carmel in view of Mattavelli teaches and renders obvious configuring a Carmel's multimedia player to utilize optimized decoding processes to play multimedia objects in a sequence such that fluidity and video and audio quality playback are maintained.⁸ (Polish, ¶¶199-200.)

⁸ While the precise bounds of this claim term (and similar claim terms such as claim 3's recitation of “optimal” encoding formats and rates) are not defined, Mattavelli, Wang, Wolters, and Carmel teach and render obvious the use of optimized decoding processes to maintain quality playback and/or optimal encoding formats and rates, falling within the core scope of these terms, as explained below.

Petition at 45-46 & n.8

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 45-46 & n.8; Reply at 30-31;

Petitioner's claim construction positions before the Board and the district court are consistent and justified



“Petitioner’s explanation may have risen to a sufficient level, for example, if Petitioner had shown that, notwithstanding the alleged indefiniteness of the claim term, an ordinarily skilled artisan would understand that the asserted art satisfies the claim limitation.”

Tesla, Inc. v. Intell. Ventures II LLC, IPR2025-00340, Paper 18 at 3-4 (Director Nov. 5, 2025) (informative)

Petitioner's claim construction positions before the Board and the district court are consistent and justified

- Notwithstanding the indefiniteness of certain claim terms, the Petition and its supporting evidence shows that a POSA would have understood the teachings of Carmel, Mattavelli, and Wang to use the same language as challenged claims and fall within the core scope of these terms

FIG. 3D is a block diagram that schematically illustrates a file format of a **multi-level data stream** 41 ... Each time slice in stream 41 includes multimedia data at **multiple quality levels**.

Carmel (Ex. 1003), 8:42-57

In the case of a multimedia stream, client 30 reconstructs and outputs the multimedia data for the appreciation of a user. Time stamps in the data stream are used to **synchronize the data**, so that the multimedia sequence is **played back** just as it was input at computer 34, preferably with only a minimal necessary transmission and decoding delay.

Carmel (Ex. 1003), 10:48-54

Periodically, client 30 makes an assessment of the rate of data transfer over the link from the server and, if necessary, changes the **quality level** accordingly. For example, if the rate is low, such that time stamps 59 indicate that the slices need to be played as fast as or faster than they are being received, the client will preferably select a **lower quality level** if one is available. On the other hand, if the rate is substantially higher than what is needed to receive the successive slices on time, the client may select a **higher quality level** to take advantage of the available bandwidth.

Carmel (Ex. 1003), 10:64-11:18

Petitioner's claim construction positions before the Board and the district court are consistent and justified

- Notwithstanding the indefiniteness of certain claim terms, the Petition and its supporting evidence shows that a POSA would have understood the teachings of Carmel, Mattavelli, and Wang to use the same language as challenged claims and fall within the core scope of these terms

ally unacceptable. The main rationale of the Computational Graceful Degradation (CGD) approach is to provide results of poorer quality when the results of the desirable quality cannot be obtained in real time. When the complexity of the task exceeds the available processing power, real-time services, possibly of degraded quality, are nevertheless guaranteed respecting all real-time constraints.

Mattavelli (Ex. 1004)
at 2309, 2311-12

This approach enables the core-Obviously for any Allocated Decoding Time the goal is to respect the real-time constraints with the lowest degradation in the video/audio content. Associating a degradation measure to each w_i^{CGD} it is possible to minimize the degradation for any given T_{Alloc} . The challenge is to find the set $S_{a,b}$ of couple of coefficients a_i, b_i , with $i \in S_{CGD}$, that satisfy minimize the video degradations expressed as difference with the non degraded output:

$$S_{a,b} = \arg\min\{a, b\}_{i \in S_{CGD}} : (PSNR - PSNR(a, b)). \quad (5)$$

The determination of the set $S_{a,b}$ can be done by means of many suitable optimization method or heuristics. The described techniques can be easily extended to the case of sequences containing multiple VOP at the same sampling time t .

We present a baseline MPEG-4 AVC (Advanced Video Coding) decoder based on an optimized platform-based design methodology. With this methodology, we jointly optimize the software and hardware design of the decoder.

Overall decoding throughput is increased by the software and the dedicated co-pi synchronization is achieved at macroblock-level. In addition, we optimize the decoder enhancing the frame buffer management padding, and content aware inverse transform

4.3 Software Optimization

To retain the flexibility and portability across different platforms, our software optimization is mainly from algorithm level improvement. Our optimization techniques are as follows.

motion compensation and inverse transform, which are the most computationally intensive modules, two dedicated acceleration modules are realized. For comparison, the proposed prototype decoder and MPEG-4 AVC reference decoder are evaluated on an ARM platform, which is one of most popular portable devices. Our experiments show that the throughput of the MPEG-4 reference decoder can be improved by 6 to 7 times. On an ARM966 board, the optimized software without hardware acceleration can achieve a decoding rate up to 5 frames per second (fps) for QCIF video sequences. With the dedicated accelerators, the overall throughput is increased by about 30% to reach 6.6 fps on the average and is up to 10.3 fps for slow motion video sequences.

Wang (Ex.1007) at 51, 53-54

Patent Owner's claim construction positions before the Board and the district court are inconsistent

- In contrast, Patent Owner has taken unjustified, inconsistent claim construction positions before the Board and the district court on the meaning and scope of the challenged claims

Compare

PO's infringement allegations (Ex. 1055):

YouTube engineering documentation explicitly states that the YouTube player (on the client) autonomously (*i.e.*, without server-side intervention) adjusts the video playback.

When you watch a video the YouTube player measures the bandwidth on the client and adaptively chooses chunks of video that can be downloaded fast enough, up to the limits of the device's viewport, decoding, and processing capability. YouTube makes multiple versions of each video at different resolutions, with bigger resolutions having higher encoding bitrates.

Ex. 1055 at 053
See Reply at 4-5

POR and Sur-Reply

“Carmel requests transmission of next segments with different rates in response to bandwidth limitations but does not discuss adjusting playback of the already downloaded segments according to multimedia object parameters and supplied host path identification.”

POR at 1

“From start to finish, Carmel adjusts only which stream segment the client requests from the server. It says nothing about adjusting playback of multimedia objects after those objects have been received.”

Sur-Reply at 3

Patent Owner's claim construction positions before the Board and the district court are inconsistent

- In contrast, Patent Owner has taken unjustified, inconsistent claim construction positions before the Board and the district court on the meaning and scope of the challenged claims

Compare

PO's expert in the litigation on the '849 patent:

14	Q. So you're reading "audio quality" and	13:10:07
15	"video quality" here to mean the best possible	13:10:10
16	audio quality and video quality that can be	13:10:15
17	obtained while maintaining fluidity?	13:10:18
18	A. Correct.	13:10:22

Goldman 7/21/25 Tr. (Ex. 1080), 115:14-18

19	Q. Okay. And so the video quality may be	13:11:45
20	diminished in order to maintain fluidity?	13:11:48
21	A. Correct.	13:11:51

Goldman 7/21/25 Tr. (Ex. 1080), 116:19-21

POR and Sur-Reply

“Degrading is not maintaining video quality.”

POR at 45

“Petitioner’s theory therefore depends on treating degradation of video quality as maintenance of video quality.”

Sur-Reply at 17

Additional Issue 4: PO fails to show that the “maintain quality playback” term is not a statement of intended use

1. A method comprising:

receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

utilizing optimized decoding processes **to maintain quality playback.**

'849 patent, cl. 1.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Reply at 32 85

PO fails to show that the “maintain quality playback” term is not a statement of intended use

- PO confusingly asserts that the functional term “to maintain quality playback” is structural simply because PO claims it modifies the “optimized decoding processes” term (POR at 62-63)
- PO offers no support for the proposition that a functional statement of intended use is transformed into a structural limitation when purportedly used as a modifier

1. A method comprising:

receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

utilizing optimized decoding processes **to maintain quality playback**.

'849 patent, cl. 1.

PO fails to show that the “maintain quality playback” term is not a statement of intended use

- PO did not clearly rely on the “maintain quality playback” term to distinguish the cited prior art during prosecution

As mentioned in the amendments made to address the 112 issues, applicant has narrowed the “metadata” to only claim parameters and supplied host path identification, neither of which are taught by Hemy (or by Vetro or Chang, for that matter).

Applicant has also clarified that adjusting selection and playback is done autonomously (ie. without server-side intervention).

In Vetro “the predictor estimates the current network state” ..“in real time” ...the predictor constrains bitstream data... and this real time update is “directed to the transcoder and the content classifier” and “rate quality trade-offs” for the system to maintain “the semantic fidelity of the content even when the bitstream needs to be further compressed to meet network and user device characteristics”. The “meta-data” among “other features of content characteristics”“are used to make rate-quality trade-offs in the content-network manager.”

The content-network manager uses this information, not the player.

The predictor and transcoder determine encoding rate on the server side, before transmission, in real time. The player makes no bitstream control decision in this process. The content-network manager, which assesses network bandwidth using the predictor, transcodes the input bitstream into a suitable encoding rate, and makes the real time determination of what will be played by the player. The content-network manager also utilizes information it may have about the player platform to further define the transmitted bitstream. Hence, on Page 6 of the office action about quotes Vetro, “the adaptable stream delivery system maintains semantic fidelity of the content even when the bitstream needs to be further compressed to meet network and user device characteristics, etc. “

Maintaining semantic fidelity and further compression are NOT the same as optimized player decoding. Semantic fidelity is based on extracted bitstream characteristics for encoding and transmission control by using the Vetro predictor model and transcoder to adaptively adjust the bitstream for network bandwidth and to tailor it to operational player characteristics/limitations. Maintaining smooth quality playback, based on playback efficiency in my specification, involves other factors, and involves dynamic player executed configurations and optimized decoding, based on the parsed multimedia objects.

’849 patent prosecution history (Ex. 1008) at 580-82.

PO fails to show that the “maintain quality playback” term is not a statement of intended use

- To the extent “maintain quality playback” should be interpreted consistently across claims, it is a statement of intended use in claim 7 as well

1. A method comprising:

receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

playing back received multimedia objects by the multimedia player, wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

utilizing optimized decoding processes **to maintain quality playback.**

'849 patent, cl. 1

7. A system comprising

at least one centralized content server transmitting multimedia objects; said multimedia objects including segmented audio and video, and parameters and supplied host path identification;

at least one multimedia player configured to:

request transmission of multimedia objects;

receive audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated object parameters and supplied host and path identification to form multimedia objects; and wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

play back received multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by utilizing optimized decoding processes **to maintain quality playback.**

'849 patent, cl. 7

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

See also Petition at 66-67; Reply at 32;
Polish Decl. (Ex. 1002) at ¶¶112, 260-262 88

Additional Slides

Cooley

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Undisputed elements not challenged by Patent Owner

- Among other things, PO did not dispute or challenge in its POR:
 - Carmel and Mattavelli (Ground 1) and Carmel and Wang (Ground 2) satisfy claim elements 1[pre]-1[c.i], and the corresponding elements of independent claims 6 & 7
 - Carmel and Mattavelli (Ground 1) and Carmel and Wang (Ground 2) satisfy the additional limitations recited in dependent claims 2-5 and 8-9
 - Carmel teaches “autonomous” adjustments by a client multimedia player
 - Mattavelli and Wang teach “optimized decoding processes”
 - The rationales for combining Carmel and Mattavelli’s teachings

The parties agree: high quality, fluid playback is a goal of any real-time system

“High quality fluid playback is a general goal of any real-time system.”

POR at 39, citing EX2014, ¶110

- This undisputed goal and knowledge of a POSA informs how a POSA would have understood Carmel, Mattavelli, and Wang’s teachings (See, e.g., Polish Decl. (Ex. 1002) ¶¶177-184, 192-193)
- Carmel, Mattavelli, and Wang’s teachings are consistent with and in service of this undisputed goal

PO ignored how the Petition considers the 1[c.-numbered] elements together

1. A method comprising:

receiving audio and video segments encoded in a digital encoding format and with an encoding rate; wherein, said audio and video segments are associated with object parameters and supplied host path identification to form multimedia objects;

requesting by a multimedia player, transmission of said multimedia objects; wherein said multimedia objects are located using http and received by said multimedia player from servers using a wireless connection;

[1.c.i] playing back received multimedia objects by the multimedia player, [1.c.ii] wherein the multimedia player is configured to play multimedia objects in a sequence such that fluidity, video quality and audio quality are maintained by selecting a plurality of said multimedia objects that reflect available network bandwidth, [1.c.iii] autonomously adjusting said selection and playback according to the multimedia object parameters and supplied host path identification, and by

[1.c.iv] utilizing optimized decoding processes to maintain quality playback.

'849 patent, cl. 1.

⁷ Carmel's teachings regarding subsequent adjustments to this selection based on changes to available bandwidth are discussed below for element 1[c.iii]. (Polish, ¶182.)

Petition at 39 n.7

Additionally, as discussed in greater detail below for claim element 1[c.iv] and incorporated here, Mattavelli also teaches autonomously adjusting the playback of received multimedia objects by the client/decoder/multimedia player according to multimedia object parameters.

Petition at 44-45

Carmel in view of Mattavelli teaches and renders obvious configuring a Carmel's multimedia player to utilize optimized decoding processes to play multimedia objects in a sequence such that fluidity and video and audio quality playback are maintained.

Petition at 45

Carmel and Mattavelli are both directed to processing multimedia data ultimately for playback "for the appreciation of a user" (*see* Carmel, 10:49-50), with both references teaching techniques for providing the highest perceived audio and video output quality possible or practicable in view of system constraints, while ensuring playback continuity. (*E.g.*, Carmel, 10:48-54, 11:9-18, 7:35-49, Figs. 3D, 6B; Mattavelli, 2309, 2310.)

Petition at 50-51

See also Petition at 39, n. 7, 44-45, 50-52; Reply at 12-13; Polish Decl. (Ex. 1002) at ¶¶174, 187, 197 92

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Carmel teaches “autonomously adjusting said selection and playback”

PO’s expert, Mr. Goldman, acknowledged that ABR (adaptive bitrate) streaming was taught by Carmel

37. The ‘473 Patent further teaches that during real-time streaming, client computers may select a file from different quality levels corresponding to that slice of multimedia data depending on the existing network conditions, such as varying bandwidth constrictions and traffic bottlenecks.⁷

38. This feature, totally unavailable in the then-current state of the art was adopted several years later as standard in the industry. *Implementing Adaptive HTTP Streaming Using the Web*, pp. 1-2 (Exhibit E).

⁷ See ‘473 Patent, 3:5-14 (“In other preferred embodiments, the slices are provided by the server at multiple resolution or quality levels. Each such level has a different degree of data compression, and thus corresponds to a different data bandwidth requirement. The client or the server monitors the data transfer rate of a data link opened therebetween and selects the level that is appropriate to the link bandwidth. If the monitored data transfer rate changes during transmission, the quality level is preferably reselected accordingly.”).