

1 RUSS, AUGUST & KABAT  
 2 Marc A. Fenster, SBN 181067  
 Email: mfenster@raklaw.com  
 3 Reza Mirzaie (CA SBN 246953)  
 4 Email: rmirzaie@raklaw.com  
 Brian D. Ledahl (CA SBN 186579)  
 5 Email: bledahl@raklaw.com  
 6 Paul Kroeger (CA SBN 229074)  
 Email: pkroeger@raklaw.com  
 7 C. Jay Chung (CA SBN 252794)  
 8 Email: jchung@raklaw.com  
 Philip X. Wang (CA SBN 262239)  
 9 Email: pwang@raklaw.com  
 10 12424 Wilshire Boulevard, 12<sup>th</sup> Floor  
 Los Angeles, California 90025  
 11 Telephone: (310) 826-7474  
 12 Facsimile: (310) 826-6991

13 *Attorneys for Plaintiff*  
 14 *REALTIME ADAPTIVE STREAMING LLC*

15 **UNITED STATES DISTRICT COURT**  
 16 **CENTRAL DISTRICT OF CALIFORNIA**  
 17 **WESTERN DIVISION**

18 REALTIME ADAPTIVE STREAMING  
 19 LLC,  
 20 Plaintiff,  
 21 vs.  
 22 GOOGLE LLC, and YOUTUBE, LLC,  
 23 Defendants.

Case No. 2:18-cv-03629-GW-JC

**JURY TRIAL DEMANDED**

24  
 25 **SECOND AMENDED COMPLAINT FOR PATENT INFRINGEMENT**

26 This is an action for patent infringement arising under the Patent Laws of the  
 27 United States of America, 35 U.S.C. § 1 *et seq.* in which Plaintiff Realtime Adaptive  
 28 Streaming LLC (“Plaintiff” or “Realtime”) makes the following allegations against

1 Defendants Google LLC and YouTube, LLC. (collectively, “Defendants” or  
2 “YouTube”).

3 **PARTIES**

4 1. Realtime is a Texas limited liability company. Realtime has a place of  
5 business at 1828 E.S.E. Loop 323, Tyler, Texas 75701. Realtime has researched and  
6 developed specific solutions for data compression. As recognition of its innovations  
7 rooted in this technological field, Realtime holds multiple United States patents and  
8 pending patent applications.

9 2. Defendant Google LLC is a Delaware limited liability company with its  
10 principal place of business in Mountain View, California.

11 3. Defendant YouTube, LLC is a Delaware limited liability company with  
12 its principal place of business in San Bruno, California. YouTube, LLC is a wholly  
13 owned subsidiary of Google LLC.

14 4. Google LLC and YouTube, LLC have regular and established places of  
15 business in this District, including, e.g., at 340 Main St., Venice, CA 90291. They  
16 offer their products and/or services, including those accused herein of infringement, to  
17 customers and potential customers located in California and in this District. Google  
18 LLC and YouTube, LLC may be served with process through its registered agent for  
19 service at The Corporation Service Company (CSC- Lawyers Incorporating Service),  
20 2710 Gateway Oaks Drive, Suite 150N, Sacramento, California 95833.

21 **JURISDICTION AND VENUE**

22 5. This action arises under the patent laws of the United States, Title 35 of  
23 the United States Code. This Court has original subject matter jurisdiction pursuant to  
24 28 U.S.C. §§ 1331 and 1338(a).

25 6. This Court has personal jurisdiction over Defendants in this action  
26 because they have committed acts within the Central District of California giving rise  
27 to this action and has established minimum contacts with this forum such that the  
28 exercise of jurisdiction over Defendants would not offend traditional notions of fair

1 play and substantial justice. Defendants have committed and continue to commit acts  
2 of infringement in this District by, among other things, offering to sell and selling  
3 products and/or services that infringe the asserted patents.

4 7. Venue is proper in this District, e.g., under 28 U.S.C. § 1400(b).  
5 Defendants are registered to do business in California, and they have transacted  
6 business in the Central District of California and have committed acts of direct and  
7 indirect infringement in the Central District of California. Defendants have regular  
8 and established place(s) of business in this District, as set forth above.

9  
10 **THE PATENTS-IN-SUIT**

11 8. This action arises under 35 U.S.C. § 271 for Defendants' infringement of  
12 Realtime's United States Patent Nos. 7,386,046 (the "'046 patent"), 8,934,535 (the  
13 "'535 patent"), 9,769,477 (the "'477 patent"), RE46,777 (the "'777 patent"), and  
14 9,578,298 (the "'298 patent").

15  
16 9. The '046 patent, titled "Bandwidth Sensitive Data Compression and  
17 Decompression," was duly and properly issued by the United States Patent and  
18 Trademark Office ("USPTO") on June 10, 2008. A copy of the '046 patent is  
19 attached hereto as Exhibit A. Realtime is the owner and assignee of the '046 patent  
20 and holds the right to sue for and recover all damages for infringement thereof,  
21 including past infringement.  
22

23  
24 10. The '535 patent, titled "Systems and methods for video and audio data  
25 storage and distribution," was duly and properly issued by the USPTO on January 13,  
26 2015. A copy of the '535 patent is attached hereto as Exhibit B. Realtime is the  
27 owner and assignee of the '535 patent and holds the right to sue for and recover all  
28

1 damages for infringement thereof, including past infringement.

2 11. The '477 patent, titled "Video data compression systems," was duly and  
3 properly issued by the USPTO on September 19, 2017. A copy of the '477 patent is  
4 attached hereto as Exhibit C. Realtime is the owner and assignee of the '477 patent  
5 and holds the right to sue for and recover all damages for infringement thereof,  
6 including past infringement.  
7

8 12. The '777 patent, titled "Quantization for Hybrid Video Coding," was  
9 duly and properly issued by the USPTO on April 3, 2018. The '777 patent is a reissue  
10 of U.S. Pat. No. 8,634,462, which was issued on January 21, 2014. A copy of the  
11 '777 patent is attached hereto as Exhibit D. Realtime is the owner and assignee of the  
12 '777 patent and holds the right to sue for and recover all damages for infringement  
13 thereof, including past infringement.  
14

15 13. The '298 patent, titled "Method for Decoding 2D-Compatible  
16 Stereoscopic Video Flows," was duly and properly issued by the USPTO on February  
17 21, 2017. A copy of the '298 patent is attached hereto as Exhibit E. Realtime is the  
18 owner and assignee of the '298 patent and holds the right to sue for and recover all  
19 damages for infringement thereof, including past infringement.  
20  
21  
22

23  
24 **COUNT I**

25 **INFRINGEMENT OF U.S. PATENT NO. 7,386,046**

26 14. Plaintiff re-alleges and incorporates by reference the foregoing  
27 paragraphs, as if fully set forth herein.  
28

1           15. On information and belief, Defendants have made, used, offered for sale,  
2 sold and/or imported into the United States products that infringe the '046 patent, and  
3 continues to do so. By way of illustrative example, these infringing products include,  
4 without limitation, Defendants' streaming products/services such as, e.g., YouTube  
5 site, YouTube TV, YouTube Live Streaming Platform, YouTube App, etc., and all  
6 versions and variations thereof since the issuance of the '046 patent ("Accused  
7 Instrumentalities").  
8  
9

10           16. On information and belief, Defendants have directly infringed and  
11 continue to infringe the '046 patent, for example, through their sale, offer for sale,  
12 importation, use and testing of the Accused Instrumentalities, which practices the  
13 system claimed by, e.g., Claim 40 of the '046 patent, namely, a system, comprising: a  
14 data compression system for compressing and decompressing data input; a plurality  
15 of compression routines selectively utilized by the data compression system, wherein  
16 a first one of the plurality of compression routines includes a first compression  
17 algorithm and a second one of the plurality of compression routines includes a second  
18 compression algorithm; and a controller for tracking throughput and generating a  
19 control signal to select a compression routine based on the throughput, wherein said  
20 tracking throughput comprises tracking a number of pending access requests to a  
21 storage device; and wherein when the controller determines that the throughput falls  
22 below a predetermined throughput threshold, the controller commands the data  
23 compression engine to use one of the plurality of compression routines to provide a  
24  
25  
26  
27  
28

1 faster rate of compression so as to increase the throughput. Upon information and  
2 belief, Defendants use the Accused Instrumentalities to practice infringing methods  
3 for its own internal non-testing business purposes, while testing the Accused  
4 Instrumentalities, and while providing technical support and repair services for the  
5 Accused Instrumentalities to Defendants' customers. For example, the Accused  
6 Instrumentalities utilize H.264 video compression standard in delivering live video  
7 via HTTP Live Streaming (HLS) technology. For example, YouTube "transcodes  
8 your content into lower bit rates, including HLS and iOS." *See, e.g.,*  
9 <https://support.google.com/youtube/answer/6251900>. Moreover, YouTube also  
10 decompresses live streams "on game consoles and mobile devices via the YouTube  
11 app and m.youtube.com. *See, e.g.,* [https://support.google.com/youtube/answer/  
12 2853702?hl=en](https://support.google.com/youtube/answer/2853702?hl=en). Furthermore, according to HLS "protocol specification does not  
13 limit the encoder selection. However, the current Apple implementation should  
14 interoperate with encoders that produce MPEG-2 Transport Streams containing H.264  
15 video and AAC audio (HE-AAC or AAC-LC)." *See, e.g.,*  
16 [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conc  
17 eptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.  
18 html](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html). As another example, HLS developer guide also states: "HTTP Live Streaming  
19 supports switching between streams dynamically if the available bandwidth changes.  
20 The client software uses heuristics to determine appropriate times to switch between  
21 the alternates. Currently, these heuristics are based on recent trends in measured  
22  
23  
24  
25  
26  
27  
28

1 network throughput.” *See, e.g.,*

2 <https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.html>.

3  
4  
5  
6 As another example, YouTube specifies ‘a plurality of compression routines  
7 selectively utilized by the data compression system’ as follows:

8  
9 Video codec: H.264, 4.1 for up to 1080p 30 FPS  
10 H.264, 4.2 for 1080p 60 FPS  
11 H.264, 5.0 for 1440p 30 FPS  
12 H.264, 5.1 for 1440p 60 FPS  
13 H.264, 5.1 for 2160p 30 FPS  
14 H.264, 5.2 for 2160p 60 FPS

15 *See, e.g.,* <https://support.google.com/youtube/answer/2853702?hl=en>

16 Moreover, YouTube “will automatically transcode your live stream to create many  
17 different output formats so all of your viewers on all of their devices and networks can  
18 watch!” *See, e.g.* <https://support.google.com/youtube/answer/2853702?hl=en>.

19  
20 Furthermore, HLS developer guide states: “The current implementation of the client  
21 observes the effective bandwidth while playing a stream. If a higher-quality stream is  
22 available and the bandwidth appears sufficient to support it, the client switches to a  
23 higher quality. If a lower-quality stream is available and the current bandwidth  
24 appears insufficient to support the current stream, the client switches to a lower  
25 quality.” *See,*

1 e.g., [https://developer.apple.com/library/content/documentation/NetworkingInternet/](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html)  
2 [Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuest](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html)  
3 [ions.html](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html).

4  
5 17. The Accused Instrumentalities include a data compression system for  
6 compressing and decompressing data input. YouTube “transcodes your content into  
7 lower bit rates, including HLS and iOS.” *See, e.g.,*  
8 [https://support.google.com/youtube](https://support.google.com/youtube/answer/6251900)  
9 [/answer/6251900](https://support.google.com/youtube/answer/6251900). For example, YouTube’s streaming products/services utilizes  
10 H.264 compression standard. As another example, YouTube’s live streaming  
11 H.264 compression standard. As another example, YouTube’s live streaming  
12 platform utilizes H.264 compression standard.  
13

14  
15  
16 Video codec: H.264, 4.1 for up to 1080p 30 FPS  
17 H.264, 4.2 for 1080p 60 FPS  
18 H.264, 5.0 for 1440p 30 FPS  
19 H.264, 5.1 for 1440p 60 FPS  
20 H.264, 5.1 for 2160p 30 FPS  
21 H.264, 5.2 for 2160p 60 FPS

22 *See, e.g.,* <https://support.google.com/youtube/answer/2853702?hl=en>.

23 18. The Accused Instrumentalities include a plurality of compression  
24 routines selectively utilized by the data compression system, wherein a first one of the  
25 plurality of compression routines includes a first compression algorithm and a second  
26 one of the plurality of compression routines includes a second compression algorithm.  
27 For example, the Accused Instrumentalities utilize H.264, which include, e.g.,  
28



1 Context-Adaptive Variable Length Coding (“CAVLC”) entropy encoder and Context-  
 2 Adaptive Binary Arithmetic Coding (“CABAC”) entropy encoder. H.264 provides  
 3 for multiple different ranges of parameters (e.g., bitrate, resolution parameters, etc.),  
 4 each included in the “profiles” and “levels” defined by the H.264 standard. See  
 5 [http://www.axis.com/files/whitepaper/wp\\_h264\\_31669\\_en\\_0803\\_lo.pdf](http://www.axis.com/files/whitepaper/wp_h264_31669_en_0803_lo.pdf) at 5:  
 6

7  
 8 **4. H.264 profiles and levels**

9 The joint group involved in defining H.264 focused on creating a simple and clean solution, limiting  
 10 options and features to a minimum. An important aspect of the standard, as with other video standards,  
 11 is providing the capabilities in profiles (sets of algorithmic features) and levels (performance classes)  
 12 that optimally support popular productions and common formats.

13 H.264 has seven profiles, each targeting a specific class of applications. Each profile defines what  
 14 feature set the encoder may use and limits the decoder implementation complexity.

15 Network cameras and video encoders will most likely use a profile called the baseline profile, which is  
 16 intended primarily for applications with limited computing resources. The baseline profile is the most  
 17 suitable given the available performance in a real-time encoder that is embedded in a network video  
 18 product. The profile also enables low latency, which is an important requirement of surveillance video and  
 19 also particularly important in enabling real-time, pan/tilt/zoom (PTZ) control in PTZ network cameras.

20 H.264 has 11 levels or degree of capability to limit performance, bandwidth and memory requirements.  
 21 Each level defines the bit rate and the encoding rate in macroblock per second for resolutions ranging  
 22 from QCIF to HDTV and beyond. The higher the resolution, the higher the level required.

23 See [https://en.wikipedia.org/wiki/H.264/MPEG-4\\_AVC](https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC):

Levels with maximum property values

Level	Max decoding speed		Max frame size		Max video bit rate for video coding layer (VCL) kbit/s			Examples for high resolution @ highest frame rate (max stored frames) <a href="#">Toggle additional details</a>
	Luma samples/s	Macroblocks/s	Luma samples	Macroblocks	Baseline, Extended and Main Profiles	High Profile	High 10 Profile	
1	380,160	1,485	25,344	99	64	80	192	176x144@15.0 (4)
1b	380,160	1,485	25,344	99	128	160	384	176x144@15.0 (4)
1.1	768,000	3,000	101,376	396	192	240	576	352x288@7.5 (2)
1.2	1,536,000	6,000	101,376	396	384	480	1,152	352x288@15.2 (6)
1.3	3,041,280	11,880	101,376	396	768	960	2,304	352x288@30.0 (6)
2	3,041,280	11,880	101,376	396	2,000	2,500	6,000	352x288@30.0 (6)
2.1	5,068,800	19,800	202,752	792	4,000	5,000	12,000	352x576@25.0 (6)
2.2	5,184,000	20,250	414,720	1,620	4,000	5,000	12,000	720x576@12.5 (5)
3	10,368,000	40,500	414,720	1,620	10,000	12,500	30,000	720x576@25.0 (5)
3.1	27,648,000	108,000	921,600	3,600	14,000	17,500	42,000	1,280x720@30.0 (5)
3.2	55,296,000	216,000	1,310,720	5,120	20,000	25,000	60,000	1,280x1,024@42.2 (4)
4	62,914,560	245,760	2,097,152	8,192	20,000	25,000	60,000	2,048x1,024@30.0 (4)
4.1	62,914,560	245,760	2,097,152	8,192	50,000	62,500	150,000	2,048x1,024@30.0 (4)
4.2	133,693,440	522,240	2,228,224	8,704	50,000	62,500	150,000	2,048x1,080@60.0 (4)
5	150,994,944	589,824	5,652,480	22,080	135,000	168,750	405,000	3,672x1,536@26.7 (5)
5.1	251,658,240	983,040	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@26.7 (5)
5.2	530,841,600	2,073,600	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@56.3 (5)

1           19. A video data block is organized by the group of pictures (GOP) structure,  
2 which is a “collection of successive pictures within a coded video stream.” *See*  
3 [https://en.wikipedia.org/wiki/Group\\_of\\_pictures](https://en.wikipedia.org/wiki/Group_of_pictures). A GOP structure can contain intra  
4 coded pictures (I picture or I frame), predictive coded pictures (P picture or P frame),  
5 bipredictive coded pictures (B picture or B frame) and direct coded pictures (D picture  
6 or D frames, or DC direct coded pictures which are used only in MPEG-1 video). *See*  
7 [https://en.wikipedia.org/wiki/Video\\_compression\\_picture\\_types](https://en.wikipedia.org/wiki/Video_compression_picture_types) (for descriptions of I  
8 frames, P frames and B frames); <https://en.wikipedia.org/wiki/MPEG-1#D-frames> (for  
9 descriptions of D frames). Thus, at least a portion of a video data block would also  
10 make up a GOP structure and could also contain I frames, P frames, B frames and/or  
11 D frames. The GOP structure also reflects the size of a video data block, and the GOP  
12 structure can be controlled and used to fine-tune other parameters (e.g. bitrate, max  
13 video bitrate and resolution parameters) or even be considered as a parameter by itself.

18           20. Based on the bitrate and/or resolution parameter identified (e.g. bitrate,  
19 max video bitrate, resolution, GOP structure or frame type within a GOP structure), a  
20 H.264-compliant system such as the Accused Instrumentalities would determine  
21 which profile (e.g., “baseline,” “extended,” “main”, or “high”) corresponds with that  
22 parameter, then select between at least two asymmetric compressors. If baseline or  
23 extended is the corresponding profile, then the system will select a Context-Adaptive  
24 Variable Length Coding (“CAVLC”) entropy encoder. If main or high is the  
25 corresponding profile, then the system will select a Context-Adaptive Binary  
26  
27  
28

1 Arithmetic Coding (“CABAC”) entropy encoder. *See*

2 <https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/>

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

	Baseline	Extended	Main	High	High 10
<b>I and P Slices</b>	Yes	Yes	Yes	Yes	Yes
<b>B Slices</b>	No	Yes	Yes	Yes	Yes
<b>SI and SP Slices</b>	No	Yes	No	No	No
<b>Multiple Reference Frames</b>	Yes	Yes	Yes	Yes	Yes
<b>In-Loop Deblocking Filter</b>	Yes	Yes	Yes	Yes	Yes
<b>CAVLC Entropy Coding</b>	Yes	Yes	Yes	Yes	Yes
<b>CABAC Entropy Coding</b>	No	No	Yes	Yes	Yes
<b>Flexible Macroblock Ordering (FMO)</b>	Yes	Yes	No	No	No
<b>Arbitrary Slice Ordering (ASO)</b>	Yes	Yes	No	No	No
<b>Redundant Slices (RS)</b>	Yes	Yes	No	No	No
<b>Data Partitioning</b>	No	Yes	No	No	No
<b>Interlaced Coding (PicAFF, MBAFF)</b>	No	Yes	Yes	Yes	Yes
<b>4:2:0 Chroma Format</b>	Yes	Yes	Yes	Yes	Yes
<b>Monochrome Video Format (4:0:0)</b>	No	No	No	Yes	Yes
<b>4:2:2 Chroma Format</b>	No	No	No	No	No
<b>4:4:4 Chroma Format</b>	No	No	No	No	No
<b>8 Bit Sample Depth</b>	Yes	Yes	Yes	Yes	Yes
<b>9 and 10 Bit Sample Depth</b>	No	No	No	No	Yes
<b>11 to 14 Bit Sample Depth</b>	No	No	No	No	No
<b>8x8 vs. 4x4 Transform Adaptivity</b>	No	No	No	Yes	Yes
<b>Quantization Scaling Matrices</b>	No	No	No	Yes	Yes
<b>Separate Cb and Cr QP control</b>	No	No	No	Yes	Yes
<b>Separate Color Plane Coding</b>	No	No	No	No	No
<b>Predictive Lossless Coding</b>	No	No	No	No	No

24 *See* [http://web.cs.ucla.edu/classes/fall03/cs218/paper/H.264\\_MPEG4\\_Tutorial.pdf](http://web.cs.ucla.edu/classes/fall03/cs218/paper/H.264_MPEG4_Tutorial.pdf) at

25 7:

26

27

28

The following table summarizes the two major types of entropy coding: Variable Length Coding (VLC) and Context Adaptive Binary Arithmetic Coding (CABAC). CABAC offers superior coding efficiency over VLC by adapting to the changing probability distribution of symbols, by exploiting correlation between symbols, and by adaptively exploiting bit correlations using arithmetic coding. H.264 also supports Context Adaptive Variable Length Coding (CAVLC) which offers superior entropy coding over VLC without the full cost of CABAC.

#### H.264 Entropy Coding – Comparison of Approaches

Characteristics	Variable Length Coding (VLC)	Context Adaptive Binary Arithmetic Coding(CABAC)
• Where it is used	MPEG-2, MPEG-4 ASP	H.264/MPEG-4 AVC (high efficiency option)
• Probability distribution	<b>Static</b> - Probabilities never change	<b>Adaptive</b> - Adjusts probabilities based on actual data
• Leverages correlation between symbols	<b>No</b> - Conditional probabilities ignored	<b>Yes</b> - Exploits symbol correlations by using "contexts"
• Non-integer code words	<b>No</b> - Low coding efficiency for high probability symbols	<b>Yes</b> - Exploits "arithmetic coding" which generates non-integer code words for higher efficiency

Moreover, the H.264 Standard requires a bit-flag descriptor, which is set to determine the correct decoder for the corresponding encoder. As shown below, if the flag = 0, then CAVLC must have been selected as the encoder; if the flag = 1, then CABAC must have been selected as the encoder. *See*

[https://www.itu.int/rec/dologin\\_pub.asp?lang=e&id=T-REC-H.264-201304-S!!PDF-E&type=items](https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-H.264-201304-S!!PDF-E&type=items) (Rec. ITU-T H.264 (04/2013)) at 80:

**entropy\_coding\_mode\_flag** selects the entropy decoding method to be applied for the syntax elements for which two descriptors appear in the syntax tables as follows:

- If **entropy\_coding\_mode\_flag** is equal to 0, the method specified by the left descriptor in the syntax table is applied (Exp-Golomb coded, see clause 9.1 or CAVLC, see clause 9.2).
- Otherwise (**entropy\_coding\_mode\_flag** is equal to 1), the method specified by the right descriptor in the syntax table is applied (CABAC, see clause 9.3).

21. After its selection, the asymmetric compressor (CAVLC or CABAC) will compress the video data to provide various compressed data blocks, which can be organized in a GOP structure (see above). *See* <https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/>:

1 **Entropy Coding**

2 For entropy coding, H.264 may use an enhanced VLC, a more complex context-adaptive  
3 variable-length coding (CAVLC) or an ever more complex Context-adaptive binary-arithmetic  
4 coding (CABAC) which are complex techniques to losslessly compress syntax elements in the  
video stream knowing the probabilities of syntax elements in a given context. The use of  
CABAC can improve the compression of around 5-7%. CABAC may requires a 30-40% of total  
processing power to be accomplished.

5 22. *See*  
6 [http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.602.1581&rep=rep1&type=](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.602.1581&rep=rep1&type=pdf)  
pdf at 13:

7 Typical compression ratios to maintain excellent quality are:

- 8 • 10:1 for general images using JPEG
- 9 • 30:1 for general video using H.263 and MPEG-2
- 10 • 60:1 for general video using H.264 and WMV9

11 *See* [http://www.ijera.com/papers/Vol3\\_issue4/BM34399403.pdf](http://www.ijera.com/papers/Vol3_issue4/BM34399403.pdf) at 2:

1 Most visual communication systems today  
2 use Baseline Profile. Baseline is the simplest H.264  
3 profile and defines, for example, zigzag scanning of  
4 the picture and using 4:2:0 (YUV video formats)  
5 chrominance sampling. In Baseline Profile, the  
6 picture is split in blocks consisting of 4x4 pixels,  
7 and each block is processed separately. Another  
8 important element of the Baseline Profile is the use  
9 of Universal Variable Length Coding (UVLC) and  
10 Context Adaptive Variable Length Coding  
11 (CAVLC) entropy coding techniques.

12 The Extended and Main Profiles includes  
13 the functionality of the Baseline Profile and add  
14 improvements to the predictions algorithms. Since  
15 transmitting every single frame (think 30 frames per  
16 second for good quality video) is not feasible if you  
17 are trying to reduce the bit rate 1000-2000 times,  
18 temporal and motion prediction are heavily used in  
19 H.264, and allow transmitting only the difference  
20 between one frame and the previous frames. The  
21 result is spectacular efficiency gain, especially for  
22 scenes with little change and motion.

23 The High Profile is the most powerful  
24 profile in H.264, and it allows most efficient coding  
25 of video. For example, large coding gain achieved  
26 through the use of Context Adaptive Binary  
27 Arithmetic Coding (CABAC) encoding which is  
28 more efficient than the UVLC/CAVLC used in  
Baseline Profile.

The High Profile also uses adaptive  
transform that decides on the fly if 4x4 or 8x8-pixel  
blocks should be used. For example, 4x4 blocks are  
used for the parts of the picture that are dense with  
detail, while parts that have little detail are  
transformed using 8x8 blocks.

23. The Accused Instrumentalities includes a controller for tracking  
throughput and generating a control signal to select a compression routine based on  
the throughput, wherein said tracking throughput comprises tracking a number of  
pending access requests to a storage device, and a controller where, when the  
controller determines that the throughput falls below a predetermined throughput  
threshold, the controller commands the data compression engine to use one of the  
plurality of compression routines to provide a faster rate of compression so as to

1 increase the throughput. For example, YouTube “will automatically transcode your  
2 live stream to create many different output formats so all of your viewers on all of  
3 their devices and networks can watch!”

4  
5 *See, e.g.* <https://support.google.com/youtube/answer/2853702?hl=en>.

6 In this regard, YouTube “transcodes your content into lower bit rates, including HLS  
7 and iOS.” *See, e.g.*, <https://support.google.com/youtube/answer/6251900>. As such,

8  
9 HLS (HTTP Live Streaming) “supports switching between streams dynamically if the  
10 available bandwidth changes. The client software uses heuristics to determine  
11 appropriate times to switch between the alternates. Currently, these heuristics are  
12 based on recent trends in measured network throughput.” *See, e.g.*,

13  
14 [https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.](https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.html)

15  
16  
17 html. “The current implementation of the client observes the effective bandwidth  
18 while playing a stream. If a higher-quality stream is available and the bandwidth  
19 appears sufficient to support it, the client switches to a higher quality. If a lower-  
20 quality stream is available and the current bandwidth appears insufficient to support  
21 the current stream, the client switches to a lower quality.” *See, e.g.*,

22  
23 [https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.](https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html)

24  
25  
26 html. The controller in the Accused Instrumentalities decides which compression (e.g.,  
27 CABAC, CAVLC, etc.) to use at a point in time based on parameters, for example,  
28

1 e.g., current or anticipated throughput. For example, when a low bandwidth is present,  
2 the Accused Instrumentalities select lower quality stream using a particular  
3 compression technique. As another example, when a high bandwidth is present, the  
4 Accused Instrumentalities select higher quality stream using another particular  
5 compression technique. For example, the Accused Instrumentalities' use of HTTP  
6 Live Streaming is directed to this selection. As another example, the Accused  
7 Instrumentalities' use of different "Profiles" of H.264 is directed to selecting lower  
8 quality stream using a particular compression technique (e.g., CABAC or CAVLC,  
9 etc.) for lower anticipated bandwidth situations, and selecting higher quality stream  
10 using a higher compression technique (e.g., CABAC or CAVLC, etc.) for higher  
11 anticipated bandwidth situations.  
12  
13  
14

15         24. On information and belief, Defendants also directly infringe and continue  
16 to infringe other claims of the '046 patent.  
17

18         25. On information and belief, all of the Accused Instrumentalities perform  
19 the claimed methods in substantially the same way, e.g., in the manner specified in the  
20 H.264 standard.  
21

22         26. On information and belief, use of the Accused Instrumentalities in their  
23 ordinary and customary fashion results in infringement of the methods and systems  
24 claimed by the '046 patent.  
25

26         27. On information and belief, Defendants have had knowledge of the '046  
27 patent since at least the filing of this Complaint or shortly thereafter, and on  
28



1 information and belief, Defendants knew of the '046 patent and knew of its  
2 infringement, including by way of this lawsuit. By the time of trial, Defendants will  
3 have known and intended (since receiving such notice) that its continued actions  
4 would actively induce and contribute to the infringement of the claims of the '046  
5 patent.  
6

7         28. Upon information and belief, Defendants' affirmative acts of making,  
8 using, and selling the Accused Instrumentalities, and providing implementation  
9 services and technical support to users of the Accused Instrumentalities, including,  
10 e.g., through training, demonstrations, brochures, installation and user guides, have  
11 induced and continue to induce users of the Accused Instrumentalities to use them in  
12 their normal and customary way to infringe the '046 patent. For example, Defendants  
13 adopted H.264 as its video codec in its YouTube TV, YouTube Live Streaming  
14 Platform, and YouTube App. As another example, YouTube on its help webpages  
15 states that "[A]n **encoder** compresses audio and video into a format that can be  
16 delivered to the YouTube platform. YouTube transcodes your content into lower bit  
17 rates, including HLS and iOS. This means your content will be available to as wide  
18 an audience as possible on desktop, mobile and tablet."  
19

20  
21  
22  
23 *See, e.g.* <https://support.google.com/youtube/answer/6251900>. For similar reasons,  
24 Defendants also induce their customers to use the Accused Instrumentalities to  
25 infringe other claims of the '046 patent. Defendants specifically intended and was  
26 aware that these normal and customary activities would infringe the '046 patent.  
27  
28

1 Defendants performed the acts that constitute induced infringement, and would induce  
2 actual infringement, with the knowledge of the '046 patent and with the knowledge, or  
3 willful blindness to the probability, that the induced acts would constitute  
4 infringement. For example, since filing of this action, Defendants know that the  
5 ordinary way of using HTTP Live Streaming—which is directed to choosing different  
6 compression techniques based on current or anticipated throughput—in the Accused  
7 Instrumentalities infringes the patent but nevertheless continues to promote HTTP  
8 Live Streaming to customers. The only reasonable inference is that Defendants  
9 specifically intend the users to infringe the patent. On information and belief,  
10 Defendants engaged in such inducement to promote the sales of the Accused  
11 Instrumentalities. Accordingly, Defendants have induced and continue to induce users  
12 of the Accused Instrumentalities to use the Accused Instrumentalities in their ordinary  
13 and customary way to infringe the '046 patent, knowing that such use constitutes  
14 infringement of the '046 patent. Accordingly, Defendants have been (as of filing of  
15 the original complaint), and currently are, inducing infringement of the '046 patent, in  
16 violation of 35 U.S.C. § 271(b).

22 29. Defendants have also infringed, and continue to infringe, claims of the  
23 '046 patent by offering to commercially distribute, commercially distributing, making,  
24 and/or importing the Accused Instrumentalities, which are used in practicing the  
25 process, or using the systems, of the '046 patent, and constitute a material part of the  
26 invention. Defendants know the components in the Accused Instrumentalities to be  
27  
28

1 especially made or especially adapted for use in infringement of the '046 patent, not a  
2 staple article, and not a commodity of commerce suitable for substantial noninfringing  
3 use. For example, the ordinary way of using HTTP Live Streaming—which is  
4 directed to choosing different compression techniques based on current or anticipated  
5 throughput—infringes the patent, and as such, is especially adapted for use in  
6 infringement. Moreover, there is no substantial noninfringing use, as HTTP Live  
7 Streaming is directed to choosing different compression techniques based on current  
8 or anticipated throughput. Accordingly, Defendants have been (as of filing of the  
9 original complaint), and currently is, contributorily infringing the '046 patent, in  
10 violation of 35 U.S.C. § 271(c).

14 30. By making, using, offering for sale, selling and/or importing into the  
15 United States the Accused Instrumentalities, and touting the benefits of using the  
16 Accused Instrumentalities' compression features, Defendants have injured Realtime  
17 and is liable to Realtime for infringement of the '046 patent pursuant to 35 U.S.C. §  
18 271.  
19

21 31. As a result of Defendants' infringement of the '046 patent, Plaintiff  
22 Realtime is entitled to monetary damages in an amount adequate to compensate for  
23 Defendants' infringement, but in no event less than a reasonable royalty for the use  
24 made of the invention by Defendants, together with interest and costs as fixed by the  
25 Court.  
26

27 **COUNT II**

28 **INFRINGEMENT OF U.S. PATENT NO. 8,934,535**

1           32. Plaintiff re-alleges and incorporates by reference the foregoing  
2 paragraphs, as if fully set forth herein.

3           33. On information and belief, Defendants have made, used, offered for sale,  
4 sold and/or imported into the United States products that infringe the ‘535 patent, and  
5 continues to do so. By way of illustrative example, these infringing products include,  
6 without limitation, YouTube’s streaming products/services such as, e.g., YouTube  
7 site, YouTube TV, YouTube Live Streaming Platform, YouTube App, etc., and all  
8 versions and variations thereof since the issuance of the ‘535 patent (“Accused  
9 Instrumentalities”).  
10  
11

12           34. On information and belief, Defendants have directly infringed and  
13 continues to infringe the ‘535 patent, for example, through its own use and testing of  
14 the Accused Instrumentalities, which when used, practices the method claimed by  
15 Claim 1 of the ‘535 patent, namely, a method, comprising: determining a parameter or  
16 attribute of at least a portion of a data block having audio or video data; selecting an  
17 access profile from among a plurality of access profiles based upon the determined  
18 parameter or attribute; and compressing the at least the portion of the data block with  
19 one or more compressors using asymmetric data compression and information from  
20 the selected access profile to create one or more compressed data blocks, the  
21 information being indicative of the one or more compressors to apply to the at least  
22 the portion of the data block. Upon information and belief, Defendants use the  
23 Accused Instrumentalities to practice infringing methods for its own internal non-  
24  
25  
26  
27  
28

1 testing business purposes, while testing the Accused Instrumentalities, and while  
2 providing technical support and repair services for the Accused Instrumentalities to  
3 Defendants' customers.

4  
5 35. For example, the Accused Instrumentalities utilize H.264 video  
6 compression standard in delivering live video via HTTP Live Streaming (HLS)  
7 technology. For example, YouTube "transcodes your content into lower bit rates,  
8 including HLS and iOS." *See, e.g.,*

9  
10 <https://support.google.com/youtube/answer/6251900>. Furthermore, according to  
11 HLS "protocol specification does not limit the encoder selection. However, the current  
12 Apple implementation should interoperate with encoders that produce MPEG-2  
13 Transport Streams containing H.264 video and AAC audio (HE-AAC or AAC-LC)."  
14 *See, e.g.,*

15  
16 [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conc  
17 eptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.  
18 html](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html). As another example, HLS developer guide also states: "HTTP Live Streaming

19 supports switching between streams dynamically if the available bandwidth changes.  
20 The client software uses heuristics to determine appropriate times to switch between  
21 the alternates. Currently, these heuristics are based on recent trends in measured  
22 network throughput." *See, e.g.,*

23  
24  
25  
26 [https://developer.apple.com/library/content/documentation/NetworkingInternet/Conc  
27 eptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.  
28](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.html)

1 html. Moreover, HLS developer guide states: “The current implementation of the  
2 client observes the effective bandwidth while playing a stream. If a higher-quality  
3 stream is available and the bandwidth appears sufficient to support it, the client  
4 switches to a higher quality. If a lower-quality stream is available and the current  
5 bandwidth appears insufficient to support the current stream, the client switches to a  
6 lower quality.” *See,*  
7  
8 *e.g.,* [https://developer.apple.com/library/content/documentation/NetworkingInternet/  
9 Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuest  
10 ions.html.](https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html)  
11

12  
13 36. The Accused Instrumentalities determine a parameter or attribute of at  
14 least a portion of a data block having audio or video data. For example, YouTube  
15 “automatically detects the stream resolution and frame rate. When you are live, we’ll  
16 transcode to lower resolutions so all of your fans can enjoy your stream no matter the  
17 quality of their Internet connection.” *See, e.g.*  
18 [https://support.google.com/youtube/answer/2853700?hl=en&ref\\_topic=6136989](https://support.google.com/youtube/answer/2853700?hl=en&ref_topic=6136989)  
19

20  
21 As shown below, examples of such parameters include bitrate (or max video bitrate)  
22 and resolution parameters. Different parameters correspond with different end  
23 applications. H.264 provides for multiple different ranges of such parameters, each  
24 included in the “profiles” and “levels” defined by the H.264 standard. *See*  
25 [http://www.axis.com/files/whitepaper/wp\\_h264\\_31669\\_en\\_0803\\_lo.pdf](http://www.axis.com/files/whitepaper/wp_h264_31669_en_0803_lo.pdf) at 5:  
26  
27  
28

#### 4. H.264 profiles and levels

The joint group involved in defining H.264 focused on creating a simple and clean solution, limiting options and features to a minimum. An important aspect of the standard, as with other video standards, is providing the capabilities in profiles (sets of algorithmic features) and levels (performance classes) that optimally support popular productions and common formats.

H.264 has seven profiles, each targeting a specific class of applications. Each profile defines what feature set the encoder may use and limits the decoder implementation complexity.

Network cameras and video encoders will most likely use a profile called the baseline profile, which is intended primarily for applications with limited computing resources. The baseline profile is the most suitable given the available performance in a real-time encoder that is embedded in a network video product. The profile also enables low latency, which is an important requirement of surveillance video and also particularly important in enabling real-time, pan/tilt/zoom (PTZ) control in PTZ network cameras.

H.264 has 11 levels or degree of capability to limit performance, bandwidth and memory requirements. Each level defines the bit rate and the encoding rate in macroblock per second for resolutions ranging from QCIF to HDTV and beyond. The higher the resolution, the higher the level required.

See [https://en.wikipedia.org/wiki/H.264/MPEG-4\\_AVC](https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC):

Levels with maximum property values

Level	Max decoding speed		Max frame size		Max video bit rate for video coding layer (VCL) kbit/s			Examples for high resolution @ highest frame rate (max stored frames) <a href="#">Toggle additional details</a>
	Luma samples/s	Macroblocks/s	Luma samples	Macroblocks	Baseline, Extended and Main Profiles	High Profile	High 10 Profile	
1	380,160	1,485	25,344	99	64	80	192	176x144@15.0 (4)
1b	380,160	1,485	25,344	99	128	160	384	176x144@15.0 (4)
1.1	768,000	3,000	101,376	396	192	240	576	352x288@7.5 (2)
1.2	1,536,000	6,000	101,376	396	384	480	1,152	352x288@15.2 (6)
1.3	3,041,280	11,880	101,376	396	768	960	2,304	352x288@30.0 (6)
2	3,041,280	11,880	101,376	396	2,000	2,500	6,000	352x288@30.0 (6)
2.1	5,068,800	19,800	202,752	792	4,000	5,000	12,000	352x576@25.0 (6)
2.2	5,184,000	20,250	414,720	1,620	4,000	5,000	12,000	720x576@12.5 (5)
3	10,368,000	40,500	414,720	1,620	10,000	12,500	30,000	720x576@25.0 (5)
3.1	27,648,000	108,000	921,600	3,600	14,000	17,500	42,000	1,280x720@30.0 (5)
3.2	55,296,000	216,000	1,310,720	5,120	20,000	25,000	60,000	1,280x1,024@42.2 (4)
4	62,914,560	245,760	2,097,152	8,192	20,000	25,000	60,000	2,048x1,024@30.0 (4)
4.1	62,914,560	245,760	2,097,152	8,192	50,000	62,500	150,000	2,048x1,024@30.0 (4)
4.2	133,693,440	522,240	2,228,224	8,704	50,000	62,500	150,000	2,048x1,080@60.0 (4)
5	150,994,944	589,824	5,652,480	22,080	135,000	168,750	405,000	3,672x1,536@26.7 (5)
5.1	251,658,240	983,040	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@26.7 (5)
5.2	530,841,600	2,073,600	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@56.3 (5)

37. A video data block is organized by the group of pictures (GOP) structure, which is a “collection of successive pictures within a coded video stream.” See [https://en.wikipedia.org/wiki/Group\\_of\\_pictures](https://en.wikipedia.org/wiki/Group_of_pictures). A GOP structure can contain intra coded pictures (I picture or I frame), predictive coded pictures (P picture or P frame), bipredictive coded pictures (B picture or B frame) and direct coded pictures (D picture

1 or D frames, or DC direct coded pictures which are used only in MPEG-1 video). *See*  
2 [https://en.wikipedia.org/wiki/Video\\_compression\\_picture\\_types](https://en.wikipedia.org/wiki/Video_compression_picture_types) (for descriptions of I  
3 frames, P frames and B frames); <https://en.wikipedia.org/wiki/MPEG-1#D-frames> (for  
4 descriptions of D frames). Thus, at least a portion of a video data block would also  
5 make up a GOP structure and could also contain I frames, P frames, B frames and/or  
6 D frames. The GOP structure also reflects the size of a video data block, and the GOP  
7 structure can be controlled and used to fine-tune other parameters (e.g. bitrate, max  
8 video bitrate and resolution parameters) or even be considered as a parameter by itself.

11 38. The Accused Instrumentalities selects an access profile from among a  
12 plurality of access profiles based upon the determined parameter or attribute. For  
13 example, based on the bitrate and/or resolution parameter identified (e.g. bitrate, max  
14 video bitrate, resolution, GOP structure or frame type within a GOP structure), any  
15 H.264-compliant system such as the Accused Instrumentalities would determine  
16 which profile (e.g., “baseline,” “extended,” “main”, or “high”) corresponds with that  
17 parameter, then select between at least two asymmetric compressors. If baseline or  
18 extended is the corresponding profile, then the system will select a Context-Adaptive  
19 Variable Length Coding (“CAVLC”) entropy encoder. If main or high is the  
20 corresponding profile, then the system will select a Context-Adaptive Binary  
21 Arithmetic Coding (“CABAC”) entropy encoder. Both encoders are asymmetric  
22 compressors because it takes a longer period of time for them to compress data than to  
23 decompress data. *See* <https://sonnati.wordpress.com/2007/10/29/how-h-264-works->  
24  
25  
26  
27  
28



1 part-ii/  
 2

	<b>Baseline</b>	<b>Extended</b>	<b>Main</b>	<b>High</b>	<b>High 10</b>
<b>I and P Slices</b>	Yes	Yes	Yes	Yes	Yes
<b>B Slices</b>	No	Yes	Yes	Yes	Yes
<b>SI and SP Slices</b>	No	Yes	No	No	No
<b>Multiple Reference Frames</b>	Yes	Yes	Yes	Yes	Yes
<b>In-Loop Deblocking Filter</b>	Yes	Yes	Yes	Yes	Yes
<b>CAVLC Entropy Coding</b>	Yes	Yes	Yes	Yes	Yes
<b>CABAC Entropy Coding</b>	No	No	Yes	Yes	Yes
<b>Flexible Macroblock Ordering (FMO)</b>	Yes	Yes	No	No	No
<b>Arbitrary Slice Ordering (ASO)</b>	Yes	Yes	No	No	No
<b>Redundant Slices (RS)</b>	Yes	Yes	No	No	No
<b>Data Partitioning</b>	No	Yes	No	No	No
<b>Interlaced Coding (PicAFF, MBAFF)</b>	No	Yes	Yes	Yes	Yes
<b>4:2:0 Chroma Format</b>	Yes	Yes	Yes	Yes	Yes
<b>Monochrome Video Format (4:0:0)</b>	No	No	No	Yes	Yes
<b>4:2:2 Chroma Format</b>	No	No	No	No	No
<b>4:4:4 Chroma Format</b>	No	No	No	No	No
<b>8 Bit Sample Depth</b>	Yes	Yes	Yes	Yes	Yes
<b>9 and 10 Bit Sample Depth</b>	No	No	No	No	Yes
<b>11 to 14 Bit Sample Depth</b>	No	No	No	No	No
<b>8x8 vs. 4x4 Transform Adaptivity</b>	No	No	No	Yes	Yes
<b>Quantization Scaling Matrices</b>	No	No	No	Yes	Yes
<b>Separate Cb and Cr QP control</b>	No	No	No	Yes	Yes
<b>Separate Color Plane Coding</b>	No	No	No	No	No
<b>Predictive Lossless Coding</b>	No	No	No	No	No

22 See [http://web.cs.ucla.edu/classes/fall03/cs218/paper/H.264\\_MPEG4\\_Tutorial.pdf](http://web.cs.ucla.edu/classes/fall03/cs218/paper/H.264_MPEG4_Tutorial.pdf) at

23 7:  
 24  
 25  
 26  
 27  
 28

1 The following table summarizes the two major types of entropy coding: Variable Length  
 2 Coding (VLC) and Context Adaptive Binary Arithmetic Coding (CABAC). CABAC offers  
 3 superior coding efficiency over VLC by adapting to the changing probability distribution  
 4 of symbols, by exploiting correlation between symbols, and by adaptively exploiting bit  
 5 correlations using arithmetic coding. H.264 also supports Context Adaptive Variable Length  
 6 Coding (CAVLC) which offers superior entropy coding over VLC without the full cost of  
 7 CABAC.

#### 8 **H.264 Entropy Coding – Comparison of Approaches**

9 <b>Characteristics</b>	10 <b>Variable Length Coding (VLC)</b>	11 <b>Context Adaptive Binary Arithmetic Coding(CABAC)</b>
12 • Where it is used	MPEG-2, MPEG-4 ASP	H.264/MPEG-4 AVC (high efficiency option)
13 • Probability distribution	<b>Static</b> - Probabilities never change	<b>Adaptive</b> - Adjusts probabilities based on actual data
14 • Leverages correlation between symbols	<b>No</b> - Conditional probabilities ignored	<b>Yes</b> - Exploits symbol correlations by using "contexts"
15 • Non-integer code words	<b>No</b> - Low coding efficiency for high probability symbols	<b>Yes</b> - Exploits "arithmetic coding" which generates non-integer code words for higher efficiency

16 Moreover, the H.264 Standard requires a bit-flag descriptor, which is set to determine  
 17 the correct decoder for the corresponding encoder. As shown below, if the flag = 0,  
 18 then CAVLC must have been selected as the encoder; if the flag = 1, then CABAC  
 19 must have been selected as the encoder. *See*

20 [https://www.itu.int/rec/dologin\\_pub.asp?lang=e&id=T-REC-H.264-201304-S!!PDF-](https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-H.264-201304-S!!PDF-E&type=items)  
 21 [E&type=items](https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-H.264-201304-S!!PDF-E&type=items) (Rec. ITU-T H.264 (04/2013)) at 80:

22 **entropy\_coding\_mode\_flag** selects the entropy decoding method to be applied for the syntax elements for which two  
 23 descriptors appear in the syntax tables as follows:

- 24 – If **entropy\_coding\_mode\_flag** is equal to 0, the method specified by the left descriptor in the syntax table is applied  
 (Exp-Golomb coded, see clause 9.1 or CAVLC, see clause 9.2).
- 25 – Otherwise (**entropy\_coding\_mode\_flag** is equal to 1), the method specified by the right descriptor in the syntax table  
 26 is applied (CABAC, see clause 9.3).

27 The Accused Instrumentalities compresses the at least the portion of the data block  
 28 with one or more compressors using asymmetric data compression and information

1 from the selected access profile to create one or more compressed data blocks, the  
2 information being indicative of the one or more compressors to apply to the at least  
3 the portion of the data block. The controller in the Accused Instrumentalities decides  
4 which compression (e.g., CABAC, CAVLC, etc.) to use at a point in time based on  
5 parameters, for example, e.g., current or anticipated throughput. For example, when a  
6 low bandwidth is present, the Accused Instrumentalities select lower quality stream  
7 using a particular compression technique. As another example, when a high  
8 bandwidth is present, the Accused Instrumentalities select higher quality stream using  
9 another particular compression technique. For example, the Accused  
10 Instrumentalities' use of HTTP Live Streaming is directed to this selection. As another  
11 example, the Accused Instrumentalities' use of different "Profiles" of H.264 is  
12 directed to selecting lower quality stream using a particular compression technique  
13 (e.g., CABAC or CAVLC, etc.) for lower anticipated bandwidth situations, and  
14 selecting higher quality stream using a higher compression technique (e.g., CABAC  
15 or CAVLC, etc.) for higher anticipated bandwidth situations.

16  
17  
18  
19  
20  
21 39. Moreover, compression techniques utilized in H.264 standard are  
22 asymmetric.

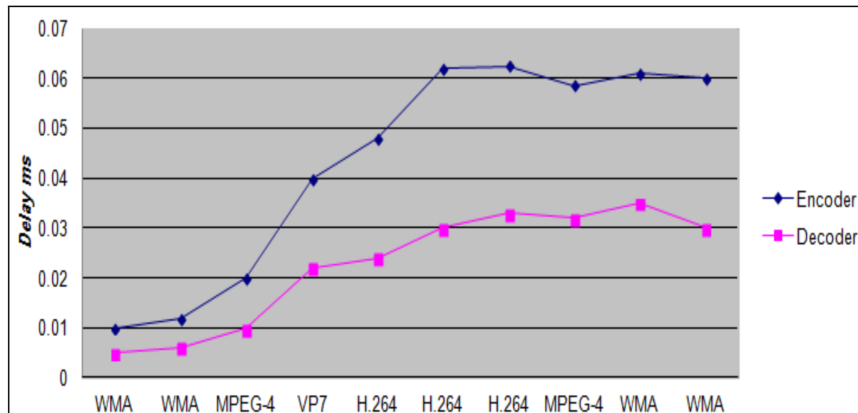


Figure 4: Encoder and Decoder Delay for Variety of Video Codec's

It is clear from figure 4, decoder delay in the most of video codec's can be assumed half of the encode delay.

See e.g., <http://www.iiste.org/Journals/index.php/NCS/article/viewFile/11072/11373>

40. YouTube “will automatically transcode your live stream to create many different output formats so all of your viewers on all of their devices and networks can watch!” See, e.g. <https://support.google.com/youtube/answer/2853702?hl=en>.

In particular, the Accused Instrumentalities compress the at least the portion of the data block with the selected one or more asymmetric compressors to provide one or more compressed data blocks, which can be organized in a GOP structure (see above). After its selection, the asymmetric compressor (CAVLC or CABAC) will compress the video data to provide various compressed data blocks, which can also be organized in a GOP structure. See <https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/>:

### Entropy Coding

For entropy coding, H.264 may use an enhanced VLC, a more complex context-adaptive variable-length coding (CAVLC) or an ever more complex Context-adaptive binary-arithmetic coding (CABAC) which are complex techniques to losslessly compress syntax elements in the video stream knowing the probabilities of syntax elements in a given context. The use of CABAC can improve the compression of around 5-7%. CABAC may requires a 30-40% of total processing power to be accomplished.

1 *See*

2 <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.602.1581&rep=rep1&type=pdf> at 13:

3 Typical compression ratios to maintain excellent quality are:

- 4 • 10:1 for general images using JPEG
- 5 • 30:1 for general video using H.263 and MPEG-2
- 6 • 60:1 for general video using H.264 and WMV9

7 *See* [http://www.ijera.com/papers/Vol3\\_issue4/BM34399403.pdf](http://www.ijera.com/papers/Vol3_issue4/BM34399403.pdf) at 2:

8 Most visual communication systems today  
9 use Baseline Profile. Baseline is the simplest H.264  
10 profile and defines, for example, zigzag scanning of  
11 the picture and using 4:2:0 (YUV video formats)  
12 chrominance sampling. In Baseline Profile, the  
13 picture is split in blocks consisting of 4x4 pixels,  
14 and each block is processed separately. Another  
15 important element of the Baseline Profile is the use  
16 of Universal Variable Length Coding (UVLC) and  
17 Context Adaptive Variable Length Coding (CAVLC) entropy coding techniques.

18 The Extended and Main Profiles includes  
19 the functionality of the Baseline Profile and add  
20 improvements to the predictions algorithms. Since  
21 transmitting every single frame (think 30 frames per  
22 second for good quality video) is not feasible if you  
23 are trying to reduce the bit rate 1000-2000 times,  
24 temporal and motion prediction are heavily used in  
25 H.264, and allow transmitting only the difference  
26 between one frame and the previous frames. The  
27 result is spectacular efficiency gain, especially for  
28 scenes with little change and motion.

29 The High Profile is the most powerful  
30 profile in H.264, and it allows most efficient coding  
31 of video. For example, large coding gain achieved  
32 through the use of Context Adaptive Binary  
33 Arithmetic Coding (CABAC) encoding which is  
34 more efficient than the UVLC/CAVLC used in  
35 Baseline Profile.

36 The High Profile also uses adaptive  
37 transform that decides on the fly if 4x4 or 8x8-pixel  
38 blocks should be used. For example, 4x4 blocks are  
39 used for the parts of the picture that are dense with  
40 detail, while parts that have little detail are  
41 transformed using 8x8 blocks.

42 41. YouTube will “automatically archive the event up to 12 hours and make  
43 it available in the Video Manager.” *See, e.g.*

44 [https://support.google.com/youtube/answer/2853700?hl=en&ref\\_topic=6136989](https://support.google.com/youtube/answer/2853700?hl=en&ref_topic=6136989)

1 Thus, on information and belief, the Accused Instrumentalities store at least a portion  
2 of the one or more compressed data blocks in buffers, hard disk, or other forms of  
3 memory/storage.

4  
5 42. On information and belief, Defendants also directly infringe and continue  
6 to infringe other claims of the '535 patent.

7  
8 43. On information and belief, all of the Accused Instrumentalities perform  
9 the claimed methods in substantially the same way, e.g., in the manner specified in the  
10 H.264 standard.

11  
12 44. On information and belief, use of the Accused Instrumentalities in their  
13 ordinary and customary fashion results in infringement of the methods and systems  
14 claimed by the '535 patent.

15  
16 45. On information and belief, Defendants have had knowledge of the '535  
17 patent since at least the filing of this Complaint or shortly thereafter, and on  
18 information and belief, Defendants knew of the '535 patent and knew of its  
19 infringement, including by way of this lawsuit. By the time of trial, Defendants will  
20 have known and intended (since receiving such notice) that its continued actions  
21 would actively induce and contribute to the infringement of the claims of the '535  
22 patent.  
23

24  
25 46. Upon information and belief, Defendants' affirmative acts of making,  
26 using, and selling the Accused Instrumentalities, and providing implementation  
27 services and technical support to users of the Accused Instrumentalities, including,  
28

1 e.g., through training, demonstrations, brochures, installation and user guides, have  
2 induced and continue to induce users of the Accused Instrumentalities to use them in  
3 their normal and customary way to infringe the ‘535 patent by practicing a method,  
4 comprising: determining a parameter of at least a portion of a data block; selecting one  
5 or more asymmetric compressors from among a plurality of compressors based upon  
6 the determined parameter or attribute; compressing the at least the portion of the data  
7 block with the selected one or more asymmetric compressors to provide one or more  
8 compressed data blocks; and storing at least a portion of the one or more compressed  
9 data blocks. For example, Defendants adopted H.264 as its video codec in its  
10 YouTube TV, YouTube Live Streaming Platform, and YouTube App. As another  
11 example, YouTube on its help webpages states that “[A]n **encoder** compresses audio  
12 and video into a format that can be delivered to the YouTube platform. YouTube  
13 transcodes your content into lower bit rates, including HLS and iOS. This means your  
14 content will be available to as wide an audience as possible on desktop, mobile and  
15 tablet.”  
16

17  
18  
19  
20  
21 *See, e.g.* <https://support.google.com/youtube/answer/6251900>. For similar reasons,  
22 Defendants also induce their customers to use the Accused Instrumentalities to  
23 infringe other claims of the ‘535 patent. Defendants specifically intended and was  
24 aware that these normal and customary activities would infringe the ‘535 patent.  
25

26 Defendants performed the acts that constitute induced infringement, and would induce  
27 actual infringement, with the knowledge of the ‘535 patent and with the knowledge, or  
28

1 willful blindness to the probability, that the induced acts would constitute  
2 infringement. For example, since filing of this action, Defendants know that the  
3 ordinary way of using HTTP Live Streaming—which is directed to choosing different  
4 compression techniques based on current or anticipated throughput—in the Accused  
5 Instrumentalities infringes the patent but nevertheless continues to promote HTTP  
6 Live Streaming to customers. The only reasonable inference is that Defendants  
7 specifically intend the users to infringe the patent. On information and belief,  
8 Defendants engaged in such inducement to promote the sales of the Accused  
9 Instrumentalities. Accordingly, Defendants have induced and continue to induce users  
10 of the Accused Instrumentalities to use the Accused Instrumentalities in their ordinary  
11 and customary way to infringe the ‘535 patent, knowing that such use constitutes  
12 infringement of the ‘535 patent. Accordingly, Defendants have been (as of filing of  
13 the original complaint), and currently is, inducing infringement of the ‘535 patent, in  
14 violation of 35 U.S.C. § 271(b).

15  
16  
17  
18  
19 47. Defendants have also infringed, and continues to infringe, claims of the  
20 ‘535 patent by offering to commercially distribute, commercially distributing, making,  
21 and/or importing the Accused Instrumentalities, which are used in practicing the  
22 process, or using the systems, of the ‘535 patent, and constitute a material part of the  
23 invention. Defendants know the components in the Accused Instrumentalities to be  
24 especially made or especially adapted for use in infringement of the ‘535 patent, not a  
25 staple article, and not a commodity of commerce suitable for substantial noninfringing  
26  
27  
28



1 use. For example, the ordinary way of using HTTP Live Streaming—which is  
2 directed to choosing different compression techniques based on current or anticipated  
3 throughput—infringes the patent, and as such, is especially adapted for use in  
4 infringement. Moreover, there is no substantial noninfringing use, as HTTP Live  
5 Streaming is directed to choosing different compression techniques based on current  
6 or anticipated throughput. Accordingly, Defendants have been (as of filing of the  
7 original complaint), and currently is, contributorily infringing the ‘535 patent, in  
8 violation of 35 U.S.C. § 271(c).  
9  
10

11 48. By making, using, offering for sale, selling and/or importing into the  
12 United States the Accused Instrumentalities, and touting the benefits of using the  
13 Accused Instrumentalities’ compression features, Defendants have injured Realtime  
14 and is liable to Realtime for infringement of the ‘535 patent pursuant to 35 U.S.C. §  
15 271.  
16  
17

18 49. As a result of Defendants’ infringement of the ‘535 patent, Plaintiff  
19 Realtime is entitled to monetary damages in an amount adequate to compensate for  
20 Defendants’ infringement, but in no event less than a reasonable royalty for the use  
21 made of the invention by Defendants, together with interest and costs as fixed by the  
22 Court.  
23  
24

### 25 COUNT III

#### 26 INFRINGEMENT OF U.S. PATENT NO. 9,769,477

27 50. Plaintiff re-alleges and incorporates by reference the foregoing  
28

1 paragraphs, as if fully set forth herein.

2 51. On information and belief, Defendants have made, used, offered for sale,  
3 sold and/or imported into the United States products that infringe the '477 patent, and  
4 continues to do so. By way of illustrative example, these infringing products include,  
5 without limitation, YouTube's streaming products/services such as, e.g., YouTube  
6 site, YouTube TV, YouTube Live Streaming Platform, YouTube App, etc., and all  
7 versions and variations thereof since the issuance of the '477 patent ("Accused  
8 Instrumentalities").  
9  
10

11 52. On information and belief, Defendants have directly infringed and  
12 continues to infringe the '477 patent, for example, through its sale, offer for sale,  
13 importation, use and testing of the Accused Instrumentalities that practice Claim 1 of  
14 the '477 patent, namely, a system, comprising: a plurality of different asymmetric data  
15 compression encoders, wherein each asymmetric data compression encoder of the  
16 plurality of different asymmetric data compression encoders is configured to utilize  
17 one or more data compression algorithms, and wherein a first asymmetric data  
18 compression encoder of the plurality of different asymmetric data compression  
19 encoders is configured to compress data blocks containing video or image data at a  
20 higher data compression rate than a second asymmetric data compression encoder of  
21 the plurality of different asymmetric data compression encoders; and one or more  
22 processors configured to: determine one or more data parameters, at least one of the  
23 determined one or more data parameters relating to a throughput of a communications  
24  
25  
26  
27  
28

1 channel measured in bits per second; and select one or more asymmetric data  
2 compression encoders from among the plurality of different asymmetric data  
3 compression encoders based upon, at least in part, the determined one or more data  
4 parameters.  
5

6 53. For example, the Accused Instrumentalities utilize H.264 video  
7 compression standard in delivering live video via HTTP Live Streaming (HLS)  
8 technology. For example, YouTube “transcodes your content into lower bit rates,  
9 including HLS and iOS.” *See, e.g.,*  
10 <https://support.google.com/youtube/answer/6251900>. Furthermore, according to  
11 HLS “protocol specification does not limit the encoder selection. However, the current  
12 Apple implementation should interoperate with encoders that produce MPEG-2  
13 Transport Streams containing H.264 video and AAC audio (HE-AAC or AAC-LC).”  
14 *See, e.g.,*  
15 [https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions](https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html).  
16  
17 html. As another example, HLS developer guide also states: “HTTP Live Streaming  
18 supports switching between streams dynamically if the available bandwidth changes.  
19  
20 The client software uses heuristics to determine appropriate times to switch between  
21 the alternates. Currently, these heuristics are based on recent trends in measured  
22 network throughput.” *See, e.g.,*  
23  
24 [https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions](https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html).  
25  
26  
27  
28

1 eptual/StreamingMediaGuide/UsingHTTPLiveStreaming/UsingHTTPLiveStreaming.  
2 html. Moreover, HLS developer guide states: “The current implementation of the  
3 client observes the effective bandwidth while playing a stream. If a higher-quality  
4 stream is available and the bandwidth appears sufficient to support it, the client  
5 switches to a higher quality. If a lower-quality stream is available and the current  
6 bandwidth appears insufficient to support the current stream, the client switches to a  
7 lower  
8 quality.” *See,*  
9 *e.g.*, [https://developer.Apple.com/library/content/documentation/NetworkingInternet/  
10 Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuest  
11 ions.html.](https://developer.Apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/FrequentlyAskedQuestions/FrequentlyAskedQuestions.html)  
12  
13

14 54. The Accused Instrumentalities include a plurality of different asymmetric  
15 data compression encoders, wherein each asymmetric data compression encoder of the  
16 plurality of different asymmetric data compression encoders is configured to utilize  
17 one or more data compression algorithms, and wherein a first asymmetric data  
18 compression encoder of the plurality of different asymmetric data compression  
19 encoders is configured to compress data blocks containing video or image data at a  
20 higher data compression rate than a second asymmetric data compression encoder of  
21 the plurality of different asymmetric data compression encoders. H.264 provides for  
22 multiple different ranges of parameters (e.g., bitrate, max video bitrate, resolution  
23 parameters, etc.), each included in the “profiles” and “levels” defined by the H.264  
24 standard. *See* [http://www.axis.com/files/whitepaper/wp\\_h264\\_31669\\_en\\_0803\\_lo.pdf](http://www.axis.com/files/whitepaper/wp_h264_31669_en_0803_lo.pdf)  
25  
26  
27  
28

1 at 5:

2 **4. H.264 profiles and levels**

3 The joint group involved in defining H.264 focused on creating a simple and clean solution, limiting  
4 options and features to a minimum. An important aspect of the standard, as with other video standards,  
5 is providing the capabilities in profiles (sets of algorithmic features) and levels (performance classes)  
6 that optimally support popular productions and common formats.

7 H.264 has seven profiles, each targeting a specific class of applications. Each profile defines what  
8 feature set the encoder may use and limits the decoder implementation complexity.

9 Network cameras and video encoders will most likely use a profile called the baseline profile, which is  
10 intended primarily for applications with limited computing resources. The baseline profile is the most  
11 suitable given the available performance in a real-time encoder that is embedded in a network video  
12 product. The profile also enables low latency, which is an important requirement of surveillance video and  
13 also particularly important in enabling real-time, pan/tilt/zoom (PTZ) control in PTZ network cameras.

14 H.264 has 11 levels or degree of capability to limit performance, bandwidth and memory requirements.  
15 Each level defines the bit rate and the encoding rate in macroblock per second for resolutions ranging  
16 from QCIF to HDTV and beyond. The higher the resolution, the higher the level required.

17 55. See [https://en.wikipedia.org/wiki/H.264/MPEG-4\\_AVC](https://en.wikipedia.org/wiki/H.264/MPEG-4_AVC):

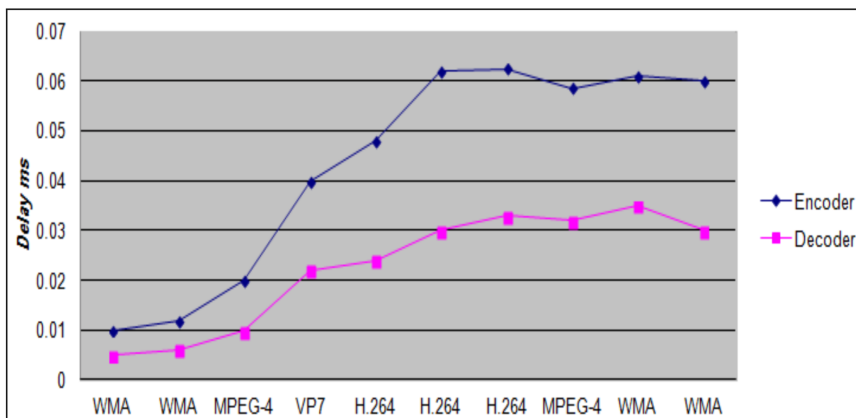
Levels with maximum property values

Level	Max decoding speed		Max frame size		Max video bit rate for video coding layer (VCL) kbit/s			Examples for high resolution @ highest frame rate (max stored frames) <input type="button" value="Toggle additional details"/>
	Luma samples/s	Macroblocks/s	Luma samples	Macroblocks	Baseline, Extended and Main Profiles	High Profile	High 10 Profile	
1	380,160	1,485	25,344	99	64	80	192	176x144@15.0 (4)
1b	380,160	1,485	25,344	99	128	160	384	176x144@15.0 (4)
1.1	768,000	3,000	101,376	396	192	240	576	352x288@7.5 (2)
1.2	1,536,000	6,000	101,376	396	384	480	1,152	352x288@15.2 (6)
1.3	3,041,280	11,880	101,376	396	768	960	2,304	352x288@30.0 (6)
2	3,041,280	11,880	101,376	396	2,000	2,500	6,000	352x288@30.0 (6)
2.1	5,068,800	19,800	202,752	792	4,000	5,000	12,000	352x576@25.0 (6)
2.2	5,184,000	20,250	414,720	1,620	4,000	5,000	12,000	720x576@12.5 (5)
3	10,368,000	40,500	414,720	1,620	10,000	12,500	30,000	720x576@25.0 (5)
3.1	27,648,000	108,000	921,600	3,600	14,000	17,500	42,000	1,280x720@30.0 (5)
3.2	55,296,000	216,000	1,310,720	5,120	20,000	25,000	60,000	1,280x1,024@42.2 (4)
4	62,914,560	245,760	2,097,152	8,192	20,000	25,000	60,000	2,048x1,024@30.0 (4)
4.1	62,914,560	245,760	2,097,152	8,192	50,000	62,500	150,000	2,048x1,024@30.0 (4)
4.2	133,693,440	522,240	2,228,224	8,704	50,000	62,500	150,000	2,048x1,080@60.0 (4)
5	150,994,944	589,824	5,652,480	22,080	135,000	168,750	405,000	3,672x1,536@26.7 (5)
5.1	251,658,240	983,040	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@26.7 (5)
5.2	530,841,600	2,073,600	9,437,184	36,864	240,000	300,000	720,000	4,096x2,304@56.3 (5)

18 56. A video data block is organized by the group of pictures (GOP) structure,  
19 which is a “collection of successive pictures within a coded video stream.” See  
20 [https://en.wikipedia.org/wiki/Group\\_of\\_pictures](https://en.wikipedia.org/wiki/Group_of_pictures). A GOP structure can contain intra  
21 coded pictures (I picture or I frame), predictive coded pictures (P picture or P frame),  
22  
23  
24  
25  
26  
27  
28

1 bipredictive coded pictures (B picture or B frame) and direct coded pictures (D picture  
 2 or D frames, or DC direct coded pictures which are used only in MPEG-1 video). See  
 3 [https://en.wikipedia.org/wiki/Video\\_compression\\_picture\\_types](https://en.wikipedia.org/wiki/Video_compression_picture_types) (for descriptions of I  
 4 frames, P frames and B frames); <https://en.wikipedia.org/wiki/MPEG-1#D-frames> (for  
 5 descriptions of D frames). Thus, at least a portion of a video data block would also  
 6 make up a GOP structure and could also contain I frames, P frames, B frames and/or  
 7 D frames. The GOP structure also reflects the size of a video data block, and the GOP  
 8 structure can be controlled and used to fine-tune other parameters (e.g. bitrate, max  
 9 video bitrate and resolution parameters) or even be considered as a parameter by itself.

10  
 11  
 12  
 13 57. Moreover, compression algorithms utilized in H.264 standard are  
 14 asymmetric.



17  
 18  
 19  
 20  
 21  
 22  
 23  
 24 Figure 4: Encoder and Decoder Delay for Variety of Video Codec's

25 It is clear from figure 4, decoder delay in the most of video codec's can be assumed half of the encode delay.

26 See e.g., <http://www.iiste.org/Journals/index.php/NCS/article/viewFile/11072/11373>

27 58. The Accused Instrumentalities include one or more processors configured  
 28 to: determine one or more data parameters, at least one of the determined one or more

1 data parameters relating to a throughput of a communications channel measured in  
2 bits per second; and select one or more asymmetric data compression encoders from  
3 among the plurality of different asymmetric data compression encoders based upon, at  
4 least in part, the determined one or more data parameters. For example, YouTube  
5 “automatically detects the stream resolution and frame rate. When you are live, we’ll  
6 transcode to lower resolutions so all of your fans can enjoy your stream no matter the  
7 quality of their Internet connection.” *See, e.g.*

8  
9  
10 [https://support.google.com/youtube/answer/2853700?hl=en&ref\\_topic=6136989](https://support.google.com/youtube/answer/2853700?hl=en&ref_topic=6136989)

11 Moreover, based on the bitrate and/or resolution parameter identified (e.g. bitrate, max  
12 video bitrate, resolution, GOP structure or frame type within a GOP structure), any  
13 H.264-compliant system such as the Accused Instrumentalities would determine  
14 which profile (e.g., “baseline,” “extended,” “main”, or “high”) corresponds with that  
15 parameter, then select between at least two asymmetric compressors. If baseline or  
16 extended is the corresponding profile, then the system will select a Context-Adaptive  
17 Variable Length Coding (“CAVLC”) entropy encoder. If main or high is the  
18 corresponding profile, then the system will select a Context-Adaptive Binary  
19 Arithmetic Coding (“CABAC”) entropy encoder. Both encoders are asymmetric  
20 compressors because it takes a longer period of time for them to compress data than to  
21 decompress data.  
22  
23  
24  
25

26 See e.g., <http://www.iiste.org/Journals/index.php/NCS/article/viewFile/11072/11373>  
27  
28

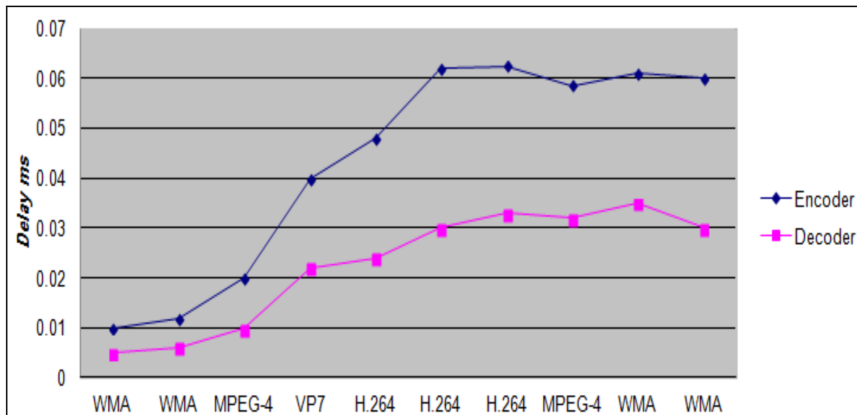


Figure 4: Encoder and Decoder Delay for Variety of Video Codec's

It is clear from figure 4, decoder delay in the most of video codec's can be assumed half of the encode delay.

See <https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/>

	Baseline	Extended	Main	High	High 10
<b>I and P Slices</b>	Yes	Yes	Yes	Yes	Yes
<b>B Slices</b>	No	Yes	Yes	Yes	Yes
<b>SI and SP Slices</b>	No	Yes	No	No	No
<b>Multiple Reference Frames</b>	Yes	Yes	Yes	Yes	Yes
<b>In-Loop Deblocking Filter</b>	Yes	Yes	Yes	Yes	Yes
<b>CAVLC Entropy Coding</b>	Yes	Yes	Yes	Yes	Yes
<b>CABAC Entropy Coding</b>	No	No	Yes	Yes	Yes
<b>Flexible Macroblock Ordering (FMO)</b>	Yes	Yes	No	No	No
<b>Arbitrary Slice Ordering (ASO)</b>	Yes	Yes	No	No	No
<b>Redundant Slices (RS)</b>	Yes	Yes	No	No	No
<b>Data Partitioning</b>	No	Yes	No	No	No
<b>Interlaced Coding (PicAFF, MBAFF)</b>	No	Yes	Yes	Yes	Yes
<b>4:2:0 Chroma Format</b>	Yes	Yes	Yes	Yes	Yes
<b>Monochrome Video Format (4:0:0)</b>	No	No	No	Yes	Yes
<b>4:2:2 Chroma Format</b>	No	No	No	No	No
<b>4:4:4 Chroma Format</b>	No	No	No	No	No
<b>8 Bit Sample Depth</b>	Yes	Yes	Yes	Yes	Yes
<b>9 and 10 Bit Sample Depth</b>	No	No	No	No	Yes
<b>11 to 14 Bit Sample Depth</b>	No	No	No	No	No
<b>8x8 vs. 4x4 Transform Adaptivity</b>	No	No	No	Yes	Yes
<b>Quantization Scaling Matrices</b>	No	No	No	Yes	Yes
<b>Separate Cb and Cr QP control</b>	No	No	No	Yes	Yes
<b>Separate Color Plane Coding</b>	No	No	No	No	No
<b>Predictive Lossless Coding</b>	No	No	No	No	No

See [http://web.cs.ucla.edu/classes/fall03/cs218/paper/H.264\\_MPEG4\\_Tutorial.pdf](http://web.cs.ucla.edu/classes/fall03/cs218/paper/H.264_MPEG4_Tutorial.pdf) at 7:



The following table summarizes the two major types of entropy coding: Variable Length Coding (VLC) and Context Adaptive Binary Arithmetic Coding (CABAC). CABAC offers superior coding efficiency over VLC by adapting to the changing probability distribution of symbols, by exploiting correlation between symbols, and by adaptively exploiting bit correlations using arithmetic coding. H.264 also supports Context Adaptive Variable Length Coding (CAVLC) which offers superior entropy coding over VLC without the full cost of CABAC.

**H.264 Entropy Coding – Comparison of Approaches**

Characteristics	Variable Length Coding (VLC)	Context Adaptive Binary Arithmetic Coding(CABAC)
• Where it is used	MPEG-2, MPEG-4 ASP	H.264/MPEG-4 AVC (high efficiency option)
• Probability distribution	<b>Static</b> - Probabilities never change	<b>Adaptive</b> - Adjusts probabilities based on actual data
• Leverages correlation between symbols	<b>No</b> - Conditional probabilities ignored	<b>Yes</b> - Exploits symbol correlations by using "contexts"
• Non-integer code words	<b>No</b> - Low coding efficiency for high probability symbols	<b>Yes</b> - Exploits "arithmetic coding" which generates non-integer code words for higher efficiency

59. Moreover, the H.264 Standard requires a bit-flag descriptor, which is set to determine the correct decoder for the corresponding encoder. As shown below, if the flag = 0, then CAVLC must have been selected as the encoder; if the flag = 1, then CABAC must have been selected as the encoder. See [https://www.itu.int/rec/dologin\\_pub.asp?lang=e&id=T-REC-H.264-201304-S!!PDF-E&type=items](https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-H.264-201304-S!!PDF-E&type=items) (Rec. ITU-T H.264 (04/2013)) at 80:

**entropy\_coding\_mode\_flag** selects the entropy decoding method to be applied for the syntax elements for which two descriptors appear in the syntax tables as follows:

- If **entropy\_coding\_mode\_flag** is equal to 0, the method specified by the left descriptor in the syntax table is applied (Exp-Golomb coded, see clause 9.1 or CAVLC, see clause 9.2).
- Otherwise (**entropy\_coding\_mode\_flag** is equal to 1), the method specified by the right descriptor in the syntax table is applied (CABAC, see clause 9.3).

60. The processor in the Accused Instrumentalities decides which compression (e.g., CABAC, CAVLC, etc.) to use at a point in time based on parameters, for example, e.g., current or anticipated throughput. For example, when a low bandwidth is present, the Accused Instrumentalities select lower quality stream using a particular compression technique. As another example, when a high

1 bandwidth is present, the Accused Instrumentalities select higher quality stream using  
2 another particular compression technique. For example, the Accused  
3 Instrumentalities' use of HTTP Live Streaming is directed to this selection. As another  
4 example, the Accused Instrumentalities' use of different "Profiles" of H.264 is  
5 directed to selecting lower quality stream using a particular compression technique  
6 (e.g., CABAC or CAVLC, etc.) for lower anticipated bandwidth situations, and  
7 selecting higher quality stream using a higher compression technique (e.g., CABAC  
8 or CAVLC, etc.) for higher anticipated bandwidth situations.

9 61. After its selection, the asymmetric compressor (CAVLC or CABAC) will  
10 compress the video data to provide various compressed data blocks, which can be  
11 organized in a GOP structure (see above). *See*

12 <https://sonnati.wordpress.com/2007/10/29/how-h-264-works-part-ii/>:

### 13 **Entropy Coding**

14 For entropy coding, H.264 may use an enhanced VLC, a more complex context-adaptive  
15 variable-length coding (CAVLC) or an ever more complex Context-adaptive binary-arithmetic  
16 coding (CABAC) which are complex techniques to losslessly compress syntax elements in the  
video stream knowing the probabilities of syntax elements in a given context. The use of  
CABAC can improve the compression of around 5-7%. CABAC may requires a 30-40% of total  
processing power to be accomplished.

17 *See*

18 <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.602.1581&rep=rep1&type=pdf> at 13:

19 Typical compression ratios to maintain excellent quality are:

- 20 • 10:1 for general images using JPEG
- 21 • 30:1 for general video using H.263 and MPEG-2
- 22 • 60:1 for general video using H.264 and WMV9

23 *See* [http://www.ijera.com/papers/Vol3\\_issue4/BM34399403.pdf](http://www.ijera.com/papers/Vol3_issue4/BM34399403.pdf) at 2:  
24  
25  
26  
27  
28

1 Most visual communication systems today  
2 use Baseline Profile. Baseline is the simplest H.264  
3 profile and defines, for example, zigzag scanning of  
4 the picture and using 4:2:0 (YUV video formats)  
5 chrominance sampling. In Baseline Profile, the  
6 picture is split in blocks consisting of 4x4 pixels,  
7 and each block is processed separately. Another  
8 important element of the Baseline Profile is the use  
9 of Universal Variable Length Coding (UVLC) and  
10 Context Adaptive Variable Length Coding  
11 (CAVLC) entropy coding techniques.

12 The Extended and Main Profiles includes  
13 the functionality of the Baseline Profile and add  
14 improvements to the predictions algorithms. Since  
15 transmitting every single frame (think 30 frames per  
16 second for good quality video) is not feasible if you  
17 are trying to reduce the bit rate 1000-2000 times,  
18 temporal and motion prediction are heavily used in  
19 H.264, and allow transmitting only the difference  
20 between one frame and the previous frames. The  
21 result is spectacular efficiency gain, especially for  
22 scenes with little change and motion.

23 The High Profile is the most powerful  
24 profile in H.264, and it allows most efficient coding  
25 of video. For example, large coding gain achieved  
26 through the use of Context Adaptive Binary  
27 Arithmetic Coding (CABAC) encoding which is  
28 more efficient than the UVLC/CAVLC used in  
Baseline Profile.

The High Profile also uses adaptive  
transform that decides on the fly if 4x4 or 8x8-pixel  
blocks should be used. For example, 4x4 blocks are  
used for the parts of the picture that are dense with  
detail, while parts that have little detail are  
transformed using 8x8 blocks.

62. On information and belief, Defendants also directly infringe and continue  
to infringe other claims of the '477 patent.

63. On information and belief, all of the Accused Instrumentalities perform  
the claimed methods in substantially the same way, e.g., in the manner specified in the  
H.264 standard.

64. On information and belief, use of the Accused Instrumentalities in their  
ordinary and customary fashion results in infringement of the methods and systems  
claimed by the '477 patent.

65. On information and belief, Defendants have had knowledge of the '477

1 patent since at least the filing of this Complaint or shortly thereafter, and on  
2 information and belief, Defendants knew of the '477 patent and knew of its  
3 infringement, including by way of this lawsuit. By the time of trial, Defendants will  
4 have known and intended (since receiving such notice) that its continued actions  
5 would actively induce and contribute to the infringement of the claims of the '477  
6 patent.  
7

8  
9 66. Upon information and belief, Defendants' affirmative acts of making,  
10 using, and selling the Accused Instrumentalities, and providing implementation  
11 services and technical support to users of the Accused Instrumentalities, including,  
12 e.g., through training, demonstrations, brochures, installation and user guides, have  
13 induced and continue to induce users of the Accused Instrumentalities to use them in  
14 their normal and customary way to infringe the '477 patent by using a system  
15 comprising: a plurality of different asymmetric data compression encoders, wherein  
16 each asymmetric data compression encoder of the plurality of different asymmetric  
17 data compression encoders is configured to utilize one or more data compression  
18 algorithms, and wherein a first asymmetric data compression encoder of the plurality  
19 of different asymmetric data compression encoders is configured to compress data  
20 blocks containing video or image data at a higher data compression rate than a second  
21 asymmetric data compression encoder of the plurality of different asymmetric data  
22 compression encoders; and one or more processors configured to: determine one or  
23 more data parameters, at least one of the determined one or more data parameters  
24  
25  
26  
27  
28

1 relating to a throughput of a communications channel measured in bits per second;  
2 and select one or more asymmetric data compression encoders from among the  
3 plurality of different asymmetric data compression encoders based upon, at least in  
4 part, the determined one or more data parameters. For example, Defendants adopted  
5 H.264 as its video codec in its YouTube TV, YouTube Live Streaming Platform, and  
6 YouTube App. As another example, YouTube on its help webpages states that  
7 “[A]n **encoder** compresses audio and video into a format that can be delivered to the  
8 YouTube platform. YouTube transcodes your content into lower bit rates, including  
9 HLS and iOS. This means your content will be available to as wide an audience as  
10 possible on desktop, mobile and tablet.” *See, e.g.*  
11 <https://support.google.com/youtube/answer/6251900>. For similar reasons, Defendants  
12 also induce their customers to use the Accused Instrumentalities to infringe other  
13 claims of the ’477 patent. Defendants specifically intended and was aware that these  
14 normal and customary activities would infringe the ’477 patent. Defendants  
15 performed the acts that constitute induced infringement, and would induce actual  
16 infringement, with the knowledge of the ’477 patent and with the knowledge, or  
17 willful blindness to the probability, that the induced acts would constitute  
18 infringement. For example, since filing of this action, Defendants know that the  
19 ordinary way of using HTTP Live Streaming—which is directed to choosing different  
20 compression techniques based on current or anticipated throughput—in the Accused  
21 Instrumentalities infringes the patent but nevertheless continues to promote HTTP  
22  
23  
24  
25  
26  
27  
28

1 Live Streaming to customers. The only reasonable inference is that Defendants  
2 specifically intend the users to infringe the patent. On information and belief,  
3 Defendants engaged in such inducement to promote the sales of the Accused  
4 Instrumentalities. Accordingly, Defendants have induced and continue to induce users  
5 of the Accused Instrumentalities to use the Accused Instrumentalities in their ordinary  
6 and customary way to infringe the '477 patent, knowing that such use constitutes  
7 infringement of the '477 patent. Accordingly, Defendants have been (as of filing of  
8 the original complaint), and currently are, inducing infringement of the '477 patent, in  
9 violation of 35 U.S.C. § 271(b).  
10  
11  
12

13 67. Defendants have also infringed, and continues to infringe, claims of  
14 the '477 patent by offering to commercially distribute, commercially distributing,  
15 making, and/or importing the Accused Instrumentalities, which are used in practicing  
16 the process, or using the systems, of the '477 patent, and constitute a material part of  
17 the invention. Defendants know the components in the Accused Instrumentalities to  
18 be especially made or especially adapted for use in infringement of the '477 patent,  
19 not a staple article, and not a commodity of commerce suitable for substantial  
20 noninfringing use. For example, the ordinary way of using HTTP Live Streaming—  
21 which is directed to choosing different compression techniques based on current or  
22 anticipated throughput—infringes the patent, and as such, is especially adapted for use  
23 in infringement. Moreover, there is no substantial noninfringing use, as HTTP Live  
24 Streaming is directed to choosing different compression techniques based on current  
25  
26  
27  
28

1 or anticipated throughput. Accordingly, Defendants have been (as of filing of the  
2 original complaint), and currently is, contributorily infringing the '477 patent, in  
3 violation of 35 U.S.C. § 271(c).  
4

5 68. By making, using, offering for sale, selling and/or importing into the  
6 United States the Accused Instrumentalities, and touting the benefits of using the  
7 Accused Instrumentalities' compression features, Defendants have injured Realtime  
8 and is liable to Realtime for infringement of the '477 patent pursuant to 35 U.S.C. §  
9 271.  
10

11 69. As a result of Defendants' infringement of the '477 patent, Plaintiff  
12 Realtime is entitled to monetary damages in an amount adequate to compensate for  
13 Defendants' infringement, but in no event less than a reasonable royalty for the use  
14 made of the invention by Defendants, together with interest and costs as fixed by the  
15 Court.  
16  
17

18 **COUNT IV**

19 **INFRINGEMENT OF U.S. PATENT NO. RE46,777**

20  
21 70. Plaintiff re-alleges and incorporates by reference the foregoing  
22 paragraphs, as if fully set forth herein.  
23

24 71. On information and belief, Defendants have made, used, offered for sale,  
25 sold and/or imported into the United States products that infringe the '777 patent, and  
26 continues to do so. By way of illustrative example, these infringing products include,  
27 without limitation, Defendants' products and services that implement the High  
28

1 Efficiency Video Coding (HEVC; also known as H.265) standard (e.g., YouTube,  
2 Google Photos, Chromecast Ultra, Google Duo, etc.), and all versions and variations  
3 thereof since the issuance of the '777 patent ("Accused Instrumentalities").  
4

5 72. On information and belief, Defendants have directly infringed and  
6 continues to infringe the '777 patent, for example, through its sale, offer for sale,  
7 importation, use and testing of the Accused Instrumentalities, which practices the  
8 method claimed by Claim 1 of the '777 patent, namely, a method for coding a video  
9 signal using hybrid coding, comprising: reducing temporal redundancy by block based  
10 motion compensated prediction in order to establish a prediction error signal;  
11 performing quantization on samples of the prediction error signal or on coefficients  
12 resulting from a transformation of the prediction error signal into the frequency  
13 domain to obtain quantized values, representing quantized samples or quantized  
14 coefficients respectively, wherein the prediction error signal includes a plurality of  
15 subblocks each including a plurality of quantized values; calculating a first  
16 quantization efficiency for the quantized values of at least one subblock of the  
17 plurality of subblocks; setting the quantized values of the at least one subblock to all  
18 zeroes; calculating a second quantization efficiency for the at least one subblock while  
19 all of the quantized values are zeroes; selecting which of the first and second  
20 quantization efficiencies is a higher efficiency; and selecting, for further proceeding,  
21 the at least one subblock with the quantized values prior to setting the quantized  
22 values of the at least one subblock to all zeroes if the first quantization efficiency is  
23  
24  
25  
26  
27  
28



1 higher and selecting the at least one subblock with the quantized values set to zero, for  
2 further proceeding, if the second quantization efficiency is higher. Upon information  
3 and belief, Defendants use the Accused Instrumentalities to practice infringing  
4 methods for its own internal non-testing business purposes, while testing the Accused  
5 Instrumentalities, and while providing technical support and repair services for the  
6 Accused Instrumentalities to Defendants' customers.  
7

8  
9 73. For example, the Accused Instrumentalities utilize the HEVC standard.  
10 *See, e.g.*, [https://support.google.com/youtube/troubleshooter/2888402?hl=en&vid=0-](https://support.google.com/youtube/troubleshooter/2888402?hl=en&vid=0-315731740723-1524058922197)  
11 [315731740723-1524058922197](https://support.google.com/youtube/troubleshooter/2888402?hl=en&vid=0-315731740723-1524058922197) (“Supported YouTube file formats ... HEVC  
12 (h265)”); <https://plus.google.com/+PeggyKTC/posts/VTD6DM5Rxvx> (“Google  
13 Photos now lets iOS11 users back up HEIF photos and HEVC videos”);  
14 <https://developers.google.com/cast/docs/media#image-formats> (“Supported Media for  
15 Google Cast ... Video codecs (Chromecast Ultra) ... HEVC / H.265”);  
16 <https://wccftech.com/google-duo-v26-brings-support-h-265/> (“Google Duo V26  
17 Brings Support for H.265”).  
18  
19  
20

21 74. The HEVC Specification (e.g., ITU-T H.265 Series H: Audiovisual and  
22 Multimedia Systems, “Infrastructure of audiovisual services – Coding of moving  
23 video” High efficiency video coding) (“HEVC Spec”) sets forth standard that is  
24 followed by HEVC compliant devices, and is relevant to both decoding and encoding  
25 that are performed pursuant to the HEVC standard.  
26

27 75. The Accused Instrumentalities performs a method for coding a video  
28

1 signal using hybrid coding. For instance, Accused Instrumentalities performs a  
2 method for coding a video signal using hybrid coding when performing coding using  
3 HEVC. For example, the aim of the coding process is the production of a bitstream,  
4 as defined in definition 3.12 of the ITU-T H.265 Series H: Audiovisual and  
5 Multimedia Systems, “Infrastructure of audiovisual services – Coding of moving  
6 video” High efficiency video coding (“HEVC Spec”): “bitstream: A sequence of bits,  
7 in the form of a NAL unit stream or a byte stream, that forms the representation of  
8 coded pictures and associated data forming one or more coded video sequences  
9 (CVSs).” *See also, e.g.*, “Overview of the High Efficiency Video Coding (HEVC)  
10 Standard” by Gary J. Sullivan, Fellow, IEEE, Jens-Rainer Ohm, Member, IEEE,  
11 Woo-Jin Han, Member, IEEE, and Thomas Wiegand, Fellow, IEEE, published in  
12 IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO  
13 TECHNOLOGY, VOL. 22, NO. 12, DECEMBER 2012 (“IEEE HEVC”) (“The video  
14 coding layer of HEVC employs the same hybrid approach (inter-/intrapicture  
15 prediction and 2-D transform coding) used in all video compression standards since  
16 H.261”). *See also, e.g.*, HEVC Spec at 0.7 “Overview of the design characteristics.”

22 76. The Accused Instrumentalities reduce temporal redundancy by block  
23 based motion compensated prediction in order to establish a prediction error signal.  
24 For instance, the Accused Instrumentalities reduce temporal redundancy by block  
25 based motion compensated prediction in order to establish a prediction error signal  
26 when performing HEVC encoding. For example, clause 8.5.3 Decoding process for  
27  
28

1 prediction units in inter prediction mode and the subclauses thereof of the HEVC Spec  
2 describe the block based motion compensation techniques used in the decoding  
3 process, which indicate that block based motion compensation prediction is used in  
4 encoding to reduce temporal redundancy and establish a prediction error signal. *See*  
5 *also, e.g., IEEE HEVC at 1651-1652* 6) Motion compensation: Quarter-sample  
7 precision is used for the MVs, and 7-tap or 8-tap filters are used for interpolation of  
8 fractional-sample positions (compared to six-tap filtering of half-sample positions  
9 followed by linear interpolation for quarter-sample positions in H.264/MPEG-4  
10 AVC). Similar to H.264/MPEG-4 AVC, multiple reference pictures are used. For each  
11 PB, either one or two motion vectors can be transmitted, resulting either in  
12 unipredictive or bipredictive coding, respectively. As in H.264/MPEG-4 AVC, a  
13 scaling and offset operation may be applied to the prediction signal(s) in a manner  
14 known as weighted prediction.”).

18 77. The Accused Instrumentalities perform quantization on samples of the  
19 prediction error signal or on coefficients resulting from a transformation of the  
20 prediction error signal into the frequency domain to obtain quantized values,  
21 representing quantized samples or quantized coefficients respectively. For instance,  
22 the Accused Instrumentalities perform quantization on samples of the prediction error  
23 signal or on coefficients resulting from a transformation of the prediction error  
24 signal into the frequency domain to obtain quantized values, representing quantized samples  
25 or quantized coefficients respectively when performing HEVC coding. For example,  
26  
27  
28

1 the quantization parameter and the scaling (inverse quantization) are defined in  
2 definitions 3.112 (page 10) and 3.131 (page 11), respectively, the usage of the scaling  
3 process in the decoding being described in clause and 8.6 Scaling, transformation and  
4 array construction process prior to deblocking filter process of the HEVC Spec, which  
5 indicate quantization as claimed when doing HEVC encoding. *See also, e.g., IEEE*  
6 *HEVC at 1652 (“8) Quantization control: As in H.264/MPEG-4 AVC, uniform*  
7 *reconstruction quantization (URQ) is used in HEVC, with quantization scaling*  
8 *matrices supported for the various transform block sizes.”).*

11 78. The Accused Instrumentalities perform a method wherein the prediction  
12 error signal includes a plurality of subblocks each including a plurality of quantized  
13 values. For instance, the Accused Instrumentalities perform a method wherein the  
14 prediction error signal includes a plurality of subblocks each including a plurality of  
15 quantized values when performing HEVC encoding. For example, the quantized  
16 samples or transform coefficients from the subblock are scaled and transformed as  
17 described in above mentioned clause 8.6 of the HEVC Spec, indicating prediction  
18 error signal as claimed when doing HEVC encoding. *See also, e.g., IEEE HEVC at*  
19 *1652 (“Prediction units and prediction blocks (PBs): The decision whether to code a*  
20 *picture area using interpicture or intrapicture prediction is made at the CU level. A PU*  
21 *partitioning structure has its root at the CU level. Depending on the basic prediction-*  
22 *type decision, the luma and chroma CBs can then be further split in size and predicted*  
23 *from luma and chroma prediction blocks (PBs). HEVC supports variable PB sizes*  
24  
25  
26  
27  
28

1 from  $64 \times 64$  down to  $4 \times 4$  samples.”).

2           79. The Accused Instrumentalities perform a method of calculating a first  
3 quantization efficiency for the quantized values of at least one subblock of the  
4 plurality of subblocks; setting the quantized values of the at least one subblock to all  
5 zeroes; calculating a second quantization efficiency for the at least one subblock while  
6 all of the quantized values are zeroes; selecting which of the first and second  
7 quantization efficiencies is a higher efficiency; and selecting, for further proceeding,  
8 the at least one subblock with the quantized values prior to setting the quantized  
9 values of the at least one subblock to all zeroes if the first quantization efficiency is  
10 higher and selecting the at least one subblock with the quantized values set to zero, for  
11 further proceeding, if the second quantization efficiency is higher. For instance, the  
12 Accused Instrumentalities perform a method of calculating a first quantization  
13 efficiency for the quantized values of at least one subblock of the plurality of  
14 subblocks; setting the quantized values of the at least one subblock to all zeroes;  
15 calculating a second quantization efficiency for the at least one subblock while all of  
16 the quantized values are zeroes; selecting which of the first and second quantization  
17 efficiencies is a higher efficiency; and selecting, for further proceeding, the at least  
18 one subblock with the quantized values prior to setting the quantized values of the at  
19 least one subblock to all zeroes if the first quantization efficiency is higher and  
20 selecting the at least one subblock with the quantized values set to zero, for further  
21 proceeding, if the second quantization efficiency is higher when performing HEVC  
22  
23  
24  
25  
26  
27  
28

1 encoding. For example, the bitstream resulting from the encoding as described in this  
2 last item of the claim contains all the relevant information as needed by the decoder  
3 for proper decoding. If the coefficients of the subblock are set to zero as a  
4 consequence of the efficiency calculation, the coded\_sub\_block\_flag, as described in  
5 clause 7.4.9.11 Residual coding semantics, HEVC Spec, is set to 0, indicating that all  
6 the 16 coefficients of the coded sub block have been set to 0:  
7

8 “coded\_sub\_block\_flag[ xS ][ yS ] specifies the following for the sub-block at  
9 location ( xS, yS ) within the current transform block, where a sub-block is a (4x4)  
10 array of 16 transform coefficient levels: – If coded\_sub\_block\_flag[ xS ][ yS ] is equal  
11 to 0, the 16 transform coefficient levels of the sub-block at location ( xS, yS ) are  
12 inferred to be equal to 0.”  
13  
14

15 80. When coded\_sub\_block\_flag[ xS ][ yS ] has not been set equal to 0, the  
16 position in the array of non 0 coefficients can be determined as follows:  
17

18 – Otherwise (coded\_sub\_block\_flag[ xS ][ yS ] is equal to 1), the following  
19 applies:  
20

21 – If ( xS, yS ) is equal to ( 0, 0 ) and ( LastSignificantCoeffX,  
22 LastSignificantCoeffY ) is not equal to ( 0, 0 ), at least one of the 16  
23 sig\_coeff\_flag syntax elements is present for the sub-block at location  
24 ( xS, yS ).  
25

26 – Otherwise, at least one of the 16 transform coefficient levels of the sub-  
27 block at location ( xS, yS ) has a non zero value.  
28

1 When coded\_sub\_block\_flag[ xS ][ yS ] is not present, it is inferred as follows:

2 – If one or more of the following conditions are true,

3 coded\_sub\_block\_flag[ xS ][ yS ] is inferred to be equal to 1:

4 – ( xS, yS ) is equal to ( 0, 0 )

5 – ( xS, yS ) is equal to ( LastSignificantCoeffX >> 2 ,

6 LastSignificantCoeffY >> 2 )

7 – Otherwise, coded\_sub\_block\_flag[ xS ][ yS ] is inferred to be equal to 0.

8  
9  
10 HEVC Spec at 7.4.9.11 Residual coding semantics. Therefore, even though the  
11 coding algorithms than can be used for reaching specific efficiency targets are not  
12 specified by the HEVC Spec (as stated in clause 0.7), this particular combination of  
13 choices produces a valid bitstream that has to be decoded by a conformant decoder.  
14

15 81. The infringement of the Accused Instrumentalities is also shown by way  
16 of considering the reference software (*see, e.g.,* <https://hevc.hhi.fraunhofer.de/>). The  
17 Accused Instrumentalities implement HEVC encoding in the same or substantially the  
18 same manner as the reference software, i.e., in infringing manner. Setting the flag  
19 RDOQ=true in the encoder configuration file enables rate-distortion-optimized  
20 quantization for transformed TUs. This feature is implemented in the HM reference  
21 software as function xRateDistOptQuant in file TComTrQuant.cpp. In the function  
22 xRateDistOptQuant, the efficiency for setting all quantized values to zero is calculated  
23 and stored in the variable d64BestCost. In the variable iBestLastIdxP1, a 0 is stored  
24 indicating that all values starting from the 0th position are set to zero. Afterwards, the  
25  
26  
27  
28

1 efficiency for keeping quantized values unequal to zero is calculated and stored in the  
 2 variable totalCost. The variable iBestLastIdxP1 is adjusted correspondingly to values  
 3 unequal to 0. The two efficiencies d64BestCost and totalCost are compared, and  
 4 selecting for further proceeding either quantized values, which are all set to zero or  
 5 quantized values, which are not all set to zero. All values starting from the position  
 6 defined by the variable iBestLastIdxP1 are set to zero.  
 7

8  
 9 82. Calculation of the efficiency for setting all quantized values to zero and  
 10 storing the result in the variable d64BestCost:

```

11
12 Double d64BestCost = 0;
13 Int ui16CtxCbf = 0;
14 Int iBestLastIdxP1 = 0;
15 if( !pcCU->isIntra( uiAbsPartIdx ) && isLuma(compID) && pcCU->getTransformIdx( uiAbsPartIdx ) == 0 )
16 {
17     ui16CtxCbf = 0;
18     d64BestCost = d64BlockUncodedCost + xGetICost( m_pcEstBitsSbac->blockRootCbpBits[ ui16CtxCbf ][ 0 ] );
19     d64BaseCost += xGetICost( m_pcEstBitsSbac->blockRootCbpBits[ ui16CtxCbf ][ 1 ] );
20 }
21 else
22 {
23     ui16CtxCbf = pcCU->getCtxQtCbf( rTu, channelType );
24     ui16CtxCbf += getCBFContextOffset(compID);
25     d64BestCost = d64BlockUncodedCost + xGetICost( m_pcEstBitsSbac->blockCbpBits[ ui16CtxCbf ][ 0 ] );
26     d64BaseCost += xGetICost( m_pcEstBitsSbac->blockCbpBits[ ui16CtxCbf ][ 1 ] );
27 }
  
```

18 HEVC Reference Software (<https://hevc.hhi.fraunhofer.de/>).

19 83. Calculating the efficiency for keeping quantized values unequal to zero  
 20 and storing the result in the variable totalCost:

```

21
22 Bool bFoundLast = false;
23 for (Int iCGScanPos = iCGLastScanPos; iCGScanPos >= 0; iCGScanPos--)
24 {
25     UInt uiCGBlkPos = codingParameters.scanCG[ iCGScanPos ];
26     d64BaseCost -= pdCostCoeffGroupSig [ iCGScanPos ];
27     if (uiSigCoeffGroupFlag[ uiCGBlkPos ])
28     {
29         for (Int iScanPosinCG = uiCGSize-1; iScanPosinCG >= 0; iScanPosinCG--)
30         {
31             iScanPos = iCGScanPos*uiCGSize + iScanPosinCG;
32             if (iScanPos > iLastScanPos) continue;
33             UInt uiBlkPos = codingParameters.scan[iScanPos];
34             if ( piDstCoeff[ uiBlkPos ] )
35             {
36                 UInt uiPosY = uiBlkPos >> uiLog2BlockWidth;
37                 UInt uiPosX = uiBlkPos - ( uiPosY << uiLog2BlockWidth );
38                 Double d64CostLast= codingParameters.scanType == SCAN_VER ? xGetRateLast( uiPosY, uiPosX, compID ) :
39                                     xGetRateLast( uiPosX, uiPosY, compID );
40                 Double totalCost = d64BaseCost + d64CostLast - pdCostSig[ iScanPos ];
  
```



1 HEVC Reference Software (<https://hevc.hhi.fraunhofer.de/>).

2 84. Comparing the two efficiencies d64BestCost and totalCost:

```
3  
4 if( totalCost < d64BestCost )  
5 {  
6     iBestLastIdxP1 = iScanPos + 1;  
7     d64BestCost = totalCost;  
8 }
```

9 HEVC Reference Software (<https://hevc.hhi.fraunhofer.de/>).

10 85. Selecting for further proceeding either quantized values, which are all set  
11 to zero or quantized values, which are not all set to zero:

```
12 //==== clean uncoded coefficients =====  
13 for ( Int scanPos = iBestLastIdxP1; scanPos <= iLastScanPos; scanPos++ )  
14 {  
15     piDstCoeff[ codingParameters.scan[ scanPos ] ] = 0;  
16 }
```

17 HEVC Reference Software (<https://hevc.hhi.fraunhofer.de/>).

18 86. On information and belief, Defendants also directly infringe and continue  
19 to infringe other claims of the '777 patent.

20 87. On information and belief, all of the Accused Instrumentalities perform  
21 the claimed methods in substantially the same way, e.g., in the manner specified in the  
22 HEVC standard.

23 88. On information and belief, use of the Accused Instrumentalities in their  
24 ordinary and customary fashion results in infringement of the methods claimed by the  
25 '777 patent.

26 89. On information and belief, Defendants have had knowledge of the '777  
27 patent since at least the filing of this Complaint or shortly thereafter, and on  
28

1 information and belief, Defendants knew of the ‘777 patent and knew of its  
2 infringement, including by way of this lawsuit. By the time of trial, Defendants will  
3 have known and intended (since receiving such notice) that its continued actions  
4 would actively induce and contribute to the infringement of the claims of the ‘777  
5 patent.  
6

7           90. Upon information and belief, Defendants’ affirmative acts of making,  
8 using, and selling the Accused Instrumentalities, and providing implementation  
9 services and technical support to users of the Accused Instrumentalities, including,  
10 e.g., through training, demonstrations, brochures, installation and user guides, have  
11 induced and continue to induce users of the Accused Instrumentalities to use them in  
12 their normal and customary way to infringe the ‘777. For example, Defendants  
13 adopted HEVC as its video codec in its products and services. For similar reasons,  
14 Defendants also induce customers to use the Accused Instrumentalities to infringe  
15 other claims of the ‘777 patent. Defendants specifically intended and was aware that  
16 these normal and customary activities would infringe the ‘777 patent. Defendants  
17 performed the acts that constitute induced infringement, and would induce actual  
18 infringement, with the knowledge of the ‘777 patent and with the knowledge, or  
19 willful blindness to the probability, that the induced acts would constitute  
20 infringement. For example, since filing of this action, Defendants know that the  
21 ordinary way of using HEVC in the Accused Instrumentalities infringes the patent but  
22 nevertheless continues to promote HEVC to customers. The only reasonable  
23  
24  
25  
26  
27  
28

1 inference is that Defendants specifically intends the users to infringe the patent. On  
2 information and belief, Defendants engaged in such inducement to promote the sales  
3 of the Accused Instrumentalities. Accordingly, Defendants have induced and continue  
4 to induce users of the Accused Instrumentalities to use the Accused Instrumentalities  
5 in their ordinary and customary way to infringe the '777 patent, knowing that such use  
6 constitutes infringement of the '777 patent. Accordingly, Defendants have been (as of  
7 filing of the original complaint), and currently are, inducing infringement of the '777  
8 patent, in violation of 35 U.S.C. § 271(b).

11 91. Defendants have also infringed, and continues to infringe, claims of the  
12 '777 patent by offering to commercially distribute, commercially distributing, making,  
13 and/or importing the Accused Instrumentalities, which are used in practicing the  
14 process, or using the systems, of the '777 patent, and constitute a material part of the  
15 invention. Defendants knows the components in the Accused Instrumentalities to be  
16 especially made or especially adapted for use in infringement of the '777 patent, not a  
17 staple article, and not a commodity of commerce suitable for substantial noninfringing  
18 use. For example, the ordinary way of using HEVC infringes the patent, and as such,  
19 is especially adapted for use in infringement with no substantial noninfringing use.  
20 Accordingly, Defendants have been (as of filing of the original complaint), and  
21 currently are, contributorily infringing the '777 patent, in violation of 35 U.S.C. §  
22 271(c).

27 92. By making, using, offering for sale, selling and/or importing into the  
28

1 United States the Accused Instrumentalities, and touting the benefits of using the  
2 Accused Instrumentalities' compression features, Defendants have injured Realtime  
3 and is liable to Realtime for infringement of the '777 patent pursuant to 35 U.S.C. §  
4  
5 271.

6 93. As a result of Defendants' infringement of the '777 patent, Plaintiff  
7 Realtime is entitled to monetary damages in an amount adequate to compensate for  
8 Defendants' infringement, but in no event less than a reasonable royalty for the use  
9 made of the invention by Defendants, together with interest and costs as fixed by the  
10 Court.  
11

12  
13 **COUNT V**

14 **INFRINGEMENT OF U.S. PATENT NO. 9,578,298**

15 94. Plaintiff re-alleges and incorporates by reference the foregoing  
16 paragraphs, as if fully set forth herein.  
17

18 95. On information and belief, Defendants have made, used, offered for sale,  
19 sold and/or imported into the United States products that infringe the '298 patent, and  
20 continues to do so. By way of illustrative example, these infringing products include,  
21 without limitation, Defendants' products and services that implement the High  
22 Efficiency Video Coding (HEVC; also known as H.265) standard (YouTube, Google  
23 Photos, Chromecast Ultra, Google Duo, Android 5.0+, etc.), and all versions and  
24 variations thereof since the issuance of the '298 patent ("Accused Instrumentalities").  
25  
26

27 96. On information and belief, Defendants have directly infringed and  
28

1 continues to infringe the '298 patent, for example, through its sale, offer for sale,  
2 importation, use and testing of the Accused Instrumentalities, which practices the  
3 method claimed by Claim 1 of the '298 patent, namely, a method for processing a  
4 video stream of digital images, the method comprising the steps of: receiving the  
5 video stream which comprises at least one composite frame (FC), each composite  
6 frame containing a pair of stereoscopic digital images (L,R) according to a  
7 predetermined frame packing format; generating an output video stream which can be  
8 reproduced on a visualization apparatus, receiving metadata which determine an area  
9 occupied by one of the two images within said composite frame (FC), said metadata  
10 indicating either a geometry of the frame packing format or a frame packing type of  
11 said composite frame (FC); determining the area in the composite frame (FC) which is  
12 occupied by said one image of the stereoscopic pair within the composite frame based  
13 on said metadata; decoding only that part of the composite frame (FC) which contains  
14 said one image to be displayed, and generating an output frame containing said  
15 decoded image. Upon information and belief, Defendants use the Accused  
16 Instrumentalities to practice infringing methods for its own internal non-testing  
17 business purposes, while testing the Accused Instrumentalities, and while providing  
18 technical support and repair services for the Accused Instrumentalities to Defendants'  
19 customers.

20  
21  
22  
23  
24  
25  
26 97. For example, the Accused Instrumentalities utilize the HEVC standard.  
27  
28 *See, e.g.*, <https://support.google.com/youtube/troubleshooter/2888402?hl=en&vid=0->

1 315731740723-1524058922197 (“Supported YouTube file formats ... HEVC  
2 (h265”); <https://plus.google.com/+PeggyKTC/posts/VTD6DM5Rxxv> (“Google  
3 Photos now lets iOS11 users back up HEIF photos and HEVC videos”);  
4 <https://developers.google.com/cast/docs/media#image-formats> (“Supported Media for  
5 Google Cast ... Video codecs (Chromecast Ultra) ... HEVC / H.265”);  
6 <https://wccftech.com/google-duo-v26-brings-support-h-265/> (“Google Duo V26  
7 Brings Support for H.265”); [https://developer.android.com/guide/topics/media/media-  
8 formats.html](https://developer.android.com/guide/topics/media/media-formats.html) (“Video formats and codecs ... H.265 HEVC ... (Android 5.0+)”).

11 98. The Accused Instrumentalities receive the video stream which comprises  
12 at least one composite frame (FC), each composite frame containing a pair of  
13 stereoscopic digital images (L,R) according to a predetermined frame packing format.  
14 For example, in the Accused Instrumentalities, the coded bitstream when it contains a  
15 stereoscopic video in one of the frame packing arrangements such as side-by-side or  
16 top-and-bottom or segmented rectangular frame packing format as defined in the  
17 following sections of the ITU-T H.265 Series H: Audiovisual and Multimedia  
18 Systems, “Infrastructure of audiovisual services – Coding of moving video” High  
19 efficiency video coding (“HEVC Spec”): D.2.16 Frame packing arrangement SEI  
20 message syntax, D.3.16 Frame packing arrangement SEI message semantics, D.2.29  
21 Segmented rectangular frame packing arrangement SEI message syntax, D.3.29  
22 Segmented rectangular frame packing arrangement SEI message semantics. Annex C,  
23 Annex D (Supplemental enhancement information), and Annex E (Video usability  
24  
25  
26  
27  
28

1 information) are each “an integral part of the Recommendation,” i.e., the HEVC Spec.  
2 HEVC Spec at Annex C, D, E. The Accused Instrumentalities implement SEI, VUI,  
3 and other items as specified in the HEVC Spec, and as stated herein.  
4

5 99. The Accused Instrumentalities generate an output video stream which  
6 can be reproduced on a visualization apparatus. For example, in the Accused  
7 Instrumentalities, the output of the decoding process as defined above is a sequence of  
8 decoded pictures. *See, e.g.*, HEVC Spec at 3.39 (“3.39 decoded picture: A decoded  
9 picture is derived by decoding a coded picture”). Decoded pictures are the input of  
10 the display process. *Id.* at 3.47 (“3.47 display process: A process not specified in this  
11 Specification having, as its input, the cropped decoded pictures that are the output of  
12 the decoding process.”).  
13  
14

15 100. The Accused Instrumentalities receive metadata which determine an area  
16 occupied by one of the two images within said composite frame, said metadata  
17 indicating either a geometry of the frame packing format or a frame packing type of  
18 said composite frame. For example, in the Accused Instrumentalities, the HEVC spec  
19 provides the default display window parameter to support 2D compatible decoding of  
20 stereo formats. *See, e.g.*, HEVC Spec (“NOTE 9 – The default display window  
21 parameters in the VUI parameters of the SPS can be used by an encoder to indicate to  
22 a decoder that does not interpret the frame packing arrangement SEI message that the  
23 default display window is an area within only one of the two constituent frames.”).  
24  
25  
26

27 101. The Accused Instrumentalities determine the area in the composite frame  
28

1 (FC) which is occupied by said one image of the stereoscopic pair within the  
 2 composite frame based on said metadata. For example, in the Accused  
 3 Instrumentalities, the default display window parameter has been defined to support  
 4 this application. The parameter syntax is defined in clause E.2.1 VUI parameters  
 5 syntax, the semantics thereof being described in clause E.3.1 VUI parameters  
 6 semantics. The usage of the Default Display Window for signaling the 2D single view  
 7 in a stereoscopic frame packing format is illustrated in Note 9 of clause D.3.16 and  
 8 Note 3 in Clause D.3.29 cited above.

11 102. The Accused Instrumentalities decode only that part of the composite  
 12 frame which contains said one image to be displayed. For example, in the Accused  
 13 Instrumentalities, tiles are intended to support independent decoding of different  
 14 picture regions. Clause 7.4.3.2.1 cited above illustrates the process to convert CTB  
 15 picture scan in CTB tile scan to enable independent decoding of the tile. *See also*  
 16 HEVC Spec:

**row\_height\_minus1**[ *i* ] plus 1 specifies the height of the *i*-th tile row in units of coding tree blocks.

The following variables are derived by invoking the coding tree block raster and tile scanning conversion process as specified in clause 6.5.1:

- The list **CtbAddrRsToTs**[ *ctbAddrRs* ] for *ctbAddrRs* ranging from 0 to **PicSizeInCtbsY** – 1, inclusive, specifying the conversion from a CTB address in the CTB raster scan of a picture to a CTB address in the tile scan,
- the list **CtbAddrTsToRs**[ *ctbAddrTs* ] for *ctbAddrTs* ranging from 0 to **PicSizeInCtbsY** – 1, inclusive, specifying the conversion from a CTB address in the tile scan to a CTB address in the CTB raster scan of a picture,
- the list **TileId**[ *ctbAddrTs* ] for *ctbAddrTs* ranging from 0 to **PicSizeInCtbsY** – 1, inclusive, specifying the conversion from a CTB address in tile scan to a tile ID,
- the list **ColumnWidthInLumaSamples**[ *i* ] for *i* ranging from 0 to **num\_tile\_columns\_minus1**, inclusive, specifying the width of the *i*-th tile column in units of luma samples,
- the list **RowHeightInLumaSamples**[ *j* ] for *j* ranging from 0 to **num\_tile\_rows\_minus1**, inclusive, specifying the height of the *j*-th tile row in units of luma samples.

The values of **ColumnWidthInLumaSamples**[ *i* ] for *i* ranging from 0 to **num\_tile\_columns\_minus1**, inclusive, and **RowHeightInLumaSamples**[ *j* ] for *j* ranging from 0 to **num\_tile\_rows\_minus1**, inclusive, shall all be greater than 0.

The array **MinTbAddrZs** with elements **MinTbAddrZs**[ *x* ][ *y* ] for *x* ranging from 0 to  $(\text{PicWidthInCtbsY} \ll (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) - 1$ , inclusive, and *y* ranging from 0 to  $(\text{PicHeightInCtbsY} \ll (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) - 1$ , inclusive, specifying the conversion from a location (*x*, *y*) in units of minimum transform blocks to a transform block address in z-scan order, is derived by invoking the z-scan order array initialization process as specified in clause 6.5.2.



1           103. The Accused Instrumentalities generate an output frame containing said  
2 extracted image. For example, in the Accused Instrumentalities, there is an output of  
3 the tile decoding process. *See, e.g.*, HEVC Spec at 8.1.1 (“8.1.1 General...Input to  
4 this process is a bitstream. Output of this process is a list of decoded pictures.”).

5  
6           104. On information and belief, Defendants also directly infringes and  
7 continues to infringe other claims of the ‘298 patent.  
8

9           105. On information and belief, all of the Accused Instrumentalities perform  
10 the claimed methods in substantially the same way, e.g., in the manner specified in the  
11 HEVC standard.  
12

13           106. On information and belief, use of the Accused Instrumentalities in their  
14 ordinary and customary fashion results in infringement of the methods claimed by the  
15 ‘298 patent.  
16

17           107. On information and belief, Defendants have had knowledge of the ‘298  
18 patent since at least the filing of this Complaint or shortly thereafter, and on  
19 information and belief, Defendants knew of the ‘298 patent and knew of its  
20 infringement, including by way of this lawsuit. By the time of trial, Defendants will  
21 have known and intended (since receiving such notice) that its continued actions  
22 would actively induce and contribute to the infringement of the claims of the ‘298  
23 patent.  
24  
25

26           108. Upon information and belief, Defendants’ affirmative acts of making,  
27 using, and selling the Accused Instrumentalities, and providing implementation  
28

1 services and technical support to users of the Accused Instrumentalities, including,  
2 e.g., through training, demonstrations, brochures, installation and user guides, have  
3 induced and continue to induce users of the Accused Instrumentalities to use them in  
4 their normal and customary way to infringe the '298. For example, Defendants  
5 adopted HEVC as its video codec in its products and services. For similar reasons,  
6 Defendants also induces its customers to use the Accused Instrumentalities to infringe  
7 other claims of the '298 patent. Defendants specifically intended and was aware that  
8 these normal and customary activities would infringe the '298 patent. Defendants  
9 performed the acts that constitute induced infringement, and would induce actual  
10 infringement, with the knowledge of the '298 patent and with the knowledge, or  
11 willful blindness to the probability, that the induced acts would constitute  
12 infringement. For example, since filing of this action, Defendants know that the  
13 ordinary way of using HEVC in the Accused Instrumentalities infringes the patent but  
14 nevertheless continues to promote HEVC to customers. The only reasonable  
15 inference is that Defendants specifically intend the users to infringe the patent. On  
16 information and belief, Defendants engaged in such inducement to promote the sales  
17 of the Accused Instrumentalities. Accordingly, Defendants have induced and continue  
18 to induce users of the Accused Instrumentalities to use the Accused Instrumentalities  
19 in their ordinary and customary way to infringe the '298 patent, knowing that such use  
20 constitutes infringement of the '298 patent. Accordingly, Defendants have been (as of  
21 filing of the original complaint), and currently is, inducing infringement of the '298  
22  
23  
24  
25  
26  
27  
28

1 patent, in violation of 35 U.S.C. § 271(b).

2 109. Defendants have also infringed, and continue to infringe, claims of the  
3 ‘298 patent by offering to commercially distribute, commercially distributing, making,  
4 and/or importing the Accused Instrumentalities, which are used in practicing the  
5 process, or using the systems, of the ‘298 patent, and constitute a material part of the  
6 invention. Defendants know the components in the Accused Instrumentalities to be  
7 especially made or especially adapted for use in infringement of the ‘298 patent, not a  
8 staple article, and not a commodity of commerce suitable for substantial noninfringing  
9 use. For example, the ordinary way of using HEVC infringes the patent, and as such,  
10 is especially adapted for use in infringement with no substantial noninfringing use.  
11 Accordingly, Defendants have been (as of filing of the original complaint), and  
12 currently are, contributorily infringing the ‘298 patent, in violation of 35 U.S.C. §  
13 271(c).

14 110. By making, using, offering for sale, selling and/or importing into the  
15 United States the Accused Instrumentalities, and touting the benefits of using the  
16 Accused Instrumentalities’ compression features, Defendants have injured Realtime  
17 and is liable to Realtime for infringement of the ‘298 patent pursuant to 35 U.S.C. §  
18 271.

19 111. As a result of Defendants’ infringement of the ‘298 patent, Plaintiff  
20 Realtime is entitled to monetary damages in an amount adequate to compensate for  
21 Defendants’ infringement, but in no event less than a reasonable royalty for the use  
22  
23  
24  
25  
26  
27  
28

1 made of the invention by Defendants, together with interest and costs as fixed by the  
2 Court.

3 **PRAYER FOR RELIEF**

4  
5 WHEREFORE, Plaintiff Realtime respectfully requests that this Court enter:

- 6 a. A judgment in favor of Plaintiff that Defendants have infringed, literally  
7 and/or under the doctrine of equivalents the '046, '535, '477, '777, and  
8 '298 patents (the "asserted patents");
- 9  
10 b. A judgment and order requiring Defendants to pay Plaintiff its damages,  
11 costs, expenses, and prejudgment and post-judgment interest for its  
12 infringement of the asserted patents, as provided under 35 U.S.C. § 284;
- 13  
14 c. A judgment and order requiring Defendants to provide an accounting and  
15 to pay supplemental damages to Realtime, including without limitation,  
16 prejudgment and post-judgment interest;
- 17  
18 d. A permanent injunction prohibiting Defendants from further acts of  
19 infringement of the asserted patents;
- 20  
21 e. A judgment and order finding that this is an exceptional case within the  
22 meaning of 35 U.S.C. § 285 and awarding to Plaintiff its reasonable  
23 attorneys' fees against YouTube; and
- 24  
25 f. Any and all other relief as the Court may deem appropriate and just under  
26 the circumstances.

27 **DEMAND FOR JURY TRIAL**

1 Plaintiff, under Rule 38 of the Federal Rules of Civil Procedure, requests a trial  
2 by jury of any issues so triable by right.  
3  
4

5 Respectfully Submitted,

6 Dated: November 8, 2018

/s/ Marc A. Fenster

7 RUSS AUGUST & KABAT  
8 Marc A. Fenster, SBN 181067  
9 Email: mfenster@raklaw.com  
10 Reza Mirzaie (CA SBN 246953)  
11 Email: rmirzaie@raklaw.com  
12 Brian D. Ledahl (CA SBN 186579)  
13 Email: bledahl@raklaw.com  
14 Paul Kroeger (CA SBN 229074)  
15 Email: pkroeger@raklaw.com  
16 C. Jay Chung (CA SBN 252794)  
17 Email: jchung@raklaw.com  
18 Philip X. Wang (CA SBN 262239)  
19 Email: pwang@raklaw.com

20 *Attorneys for Plaintiff*  
21 *REALTIME ADAPTIVE STREAMING LLC*  
22  
23  
24  
25  
26  
27  
28

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28

**CERTIFICATE OF SERVICE**

I hereby certify that the foregoing document was served on all counsel of record via electronic service on November 8, 2018.

/s/ Marc A. Fenster