UNITED STATES DISTRICT COURT
DISTRICT OF DELAWARE

VIDEOLABS, INC., and
VL COLLECTIVE IP LLC

Plaintiffs,

v.

META PLATFORMS, INC.;
INSTAGRAM, INC.; WHATSAPP LLC;
FACEBOOK TECHNOLOGIES, LLC

Defendants.

Civil Action No. 1:22-cv-00680-JHS

JURY TRIAL DEMANDED

FIRST AMENDED COMPLAINT

Plaintiffs VideoLabs, Inc. ("VL") and VL Collective IP LLC ("VL IP") (collectively "VideoLabs" or "Plaintiffs") file this First Amended Complaint against Defendants Meta Platforms, Inc. ("Meta"), Instagram, Inc. ("Instagram"), WhatsApp LLC ("WhatsApp"), and Facebook Technologies, LLC ("Facebook Technologies") (collectively, the "Meta Companies" or "Defendants"),[1] and in support thereof alleges as follows:

NATURE OF THE ACTION

1. Digital video has become fundamental to how society interacts, communicates, educates, and entertains. In fact, video consumption now accounts for more than 82% of all Internet traffic.[2] The ability to reliably provide high-quality video drives the growth of digital

[1] VideoLabs filed a stipulation dismissing Giphy, Inc. without prejudice on May 20, 2024. ECF No. 39.

[2] *See* Ex. 6, *The Sustainable Future of Video Entertainment*, INTERDIGITAL (Aug. 2020), https://www.interdigital.com/white_papers/the-sustainable-future-of-video-entertainment?submit_success=true (last visited May 24, 2022).

platforms that are increasingly integral to the global economy. As Meta's CEO, Mark Zuckerberg, said, "video is a mega trend" on the "same order as mobile.[3] Meta's COO, Sheryl Sandberg further explained that "[c]onsumer video is exploding on [Meta's] platform . . . and that really creates ad opportunities."[4]

2. The advent of high-quality video as a staple of digital consumption did not happen instantaneously. As with any complex technology, digital video presented implementation challenges. Many companies spent many years and resources to develop new and innovative technologies that guide how video is created, streamed, secured, managed, and consumed.

3. Various inventions and technological advances have transformed digital video. Some of these technologies, such as techniques to efficiently compress video file size, address central challenges to storing and transmitting video. Others enable video content to be efficiently and securely streamed to the many user devices that exist today. Yet others involve managing and organizing videos to provide viewers easier access to content and address how they interact with content. And others involve identifying the content of videos so as to better target users. Successful video streaming thus requires myriad technologies that necessarily coordinate with one another.

4. Because various companies played roles in developing the foundational technology for today's digital video, no single company can provide the high-quality video experiences that consumers have come to expect without using technology owned by other companies.

---

[3] Ex. 7, *Mark Zuckerberg sees video as a 'mega trend' and is gunning for* YouTube, CNBC (Feb. 2017), https://www.cnbc.com/2017/02/01/mark-zuckerberg-video-mega-trend-like-mobile.html (last visited May 24, 2022).

[4] *Id.*

5. The founders of VideoLabs recognized this problem and understood that collective action was needed to address it. If the companies that developed critical video technologies worked together, everyone could benefit: innovators could receive fair compensation for their contributions, companies deploying video technology could respect the innovators' patents and license them on affordable and predictable terms, and consumers could experience better and more affordable video technology.

6. In 2019, with support from widely-recognized industry leaders, VideoLabs launched a platform to achieve these goals. VideoLabs spent millions of dollars and thousands of hours analyzing the video space and identifying the patents that reflect the innovations with the highest impact. VideoLabs then compiled a portfolio of these core patents, obtaining them from leading companies, including Hewlett Packard Enterprise, Alcatel-Lucent S.A., Siemens AG, Swisscom AG, 3Com, Panasonic, LG, and Nokia.

7. VideoLabs then opened-up membership on its platform to all willing companies. In exchange for low-cost membership or licensing fees, VideoLabs provides access to its patent portfolio and a commitment to seek out the most important patents in the video industry and clear them. Many prominent companies recognized the benefits of the VideoLabs platform and worked with VideoLabs to efficiently and responsibly license its video technology patents.

8. Unfortunately, Defendants have not. The Meta Companies operate several of the world's most popular social media platforms, and in doing so, make extensive use of video technologies. They are enmeshed in practically every aspect of video, from creation to processing, delivery, targeting, and display.

9. VideoLabs has reached out to the Meta Companies multiple times over many years to alert it to its use of VideoLabs' patented technology and offer them the benefit of VideoLabs'

platform. On June 26, 2020, VideoLabs was told that the Meta Companies were not interested in good faith licensing discussions. Several follow-up attempts by VideoLabs were made in early and mid-2021, but VideoLabs received no response to those inquiries. Rather than meaningfully engage with VideoLabs, the Meta Companies have chosen to continue to free-ride on VideoLabs' patents and the significant innovations they represent.

10. Defendants' refusal to acknowledge VideoLabs' patents and offer fair compensation for their use violates the patent laws and undermines the viability of VideoLabs' platform. VideoLabs feels it has no recourse but to file this action to stop the Meta Companies unauthorized use of VideoLabs' patents.

11. This case is ultimately about ensuring the integrity of the patent system and compensating patent owners for their protected innovations. Respect for intellectual property, as the law requires, is essential to incentivize innovation and promote technological progress. Accordingly, VideoLabs brings this action under the patent laws, 35 U.S.C. § 1 *et seq.*, in order to stop the Meta Companies' willful infringement of U.S. Patent Nos. 7,769,238, 8,139,878, 7,970,059, 7,266,682, and 7436,980 (collectively, "patents-in-suit").

THE PARTIES

12. VL was founded in 2018 as part of an industry-sponsored and -funded effort to reduce the cost and risk of technological gridlock associated with diverse patent ownership. VL's leadership has decades of experience in intellectual property licensing, during which they have completed over 1,000 intellectual property transactions worldwide and drawn more than $6 billion in revenue.

13. VL is a corporation organized under the laws of the State of Delaware, with its principal place of business in Palo Alto, California.

14. VL IP was founded in 2019 as a subsidiary of VideoLabs, Inc.

15. VL IP is a corporation organized under the laws of the State of Delaware, with its principal place of business in Palo Alto, California.

16. On information and belief, Meta is a publicly traded corporation organized and existing under the laws of the State of Delaware and is registered to do business in the State of Delaware. Meta's headquarters are located at 1601 Willow Road, Menlo Park, California 94025.

17. On information and belief, Instagram, WhatsApp, and Facebook Technologies are each subsidiaries of Meta, and the companies are heavily intertwined, including with respect to the products accused herein of infringing VideoLabs' patents and the Defendants' infringement. The Defendants share many underlying technology, resources, platforms, and architecture, particularly with respect to the accused products and the Defendants' infringement. For example, Meta's recent "data center is intended to help support all of Meta's apps and services, including Facebook, WhatsApp, Instagram and Meta Quest."[5] Similarly, "WhatsApp relies on Meta data centers to provide its services to you."[6] As another example, Facebook Messenger and Instagram are connected and can be used together.[7] And content from Instagram can be shared on Facebook.[8]

[5] Ex. 8, *Meta to invest $800 million in new Central Texas data center*, THE DALLAS MORNING NEWS (Mar. 2022), https://www.dallasnews.com/business/technology/2022/03/31/meta-to-invest-800-million-in-new-central-texas-data-center (last visited May 24, 2022).

[6] Ex. 9, *Redirect to Meta*, WHATSAPP (2022), https://faq.whatsapp.com/general/redirect-to-meta/?lang=en (last visited May 24, 2022).

[7] Ex. 10, *How to Send Message on Instagram Without the Instagram App*, TECHWISER (May 2021), https://techwiser.com/how-to-connect-facebook-messenger-to-instagram (last visited May 24, 2022).

[8] Ex. 11, *How do I share my video on Instagram to my Facebook Page?*, INSTAGRAM HELP CENTER (2022), https://www.facebook.com/help/instagram/486878428409681 (last visited May 24, 2022).

Further, the Defendants also share legal policies, terms, and conditions.[9] As yet another example, Facebook Technologies works with Meta to provide Oculus (also called Quest) and Portal products and services.

JURISDICTION AND VENUE

18. This is an action for patent infringement arising under the patent laws of the United States. This Court has jurisdiction over the subject matter of this action under 28 U.S.C. §§ 1331 and 1338(a), as well as 28 U.S.C. § 1367(a).

19. This Court has personal jurisdiction, including general and specific jurisdiction, over Meta. Meta is incorporated under the laws of the State of Delaware. On information and belief, Meta conducts business in, has continuous and systematic contacts with, and has committed acts of patent infringement in the State of Delaware and in this District, and has established minimum contacts with this forum state such that the exercise of jurisdiction over Meta would not offend traditional notions of fair play and substantial justice. On information and belief, Meta markets, offers for sale, sells, and/or uses products and/or services, including those presently accused of infringement, in this District. Further on information and belief, Meta markets, offers for sale, and/or sells products and/or services, including those presently accused of infringement, to customers and potential customers in this District.

20. This Court has personal jurisdiction, including general and specific jurisdiction, over Instagram. Instagram is incorporated under the laws of the State of Delaware. On information and belief, Instagram conducts business in, has continuous and systematic contacts with, and has committed acts of patent infringement in the State of Delaware and in this District, and has

[9] Ex. 12, *Data Policy*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/519522125107875/?maybe_redirect_pol=0 (last visited May 24, 2022).

established minimum contacts with this forum state such that the exercise of jurisdiction over Instagram would not offend traditional notions of fair play and substantial justice. On information and belief, Instagram markets, offers for sale, sells, and/or uses products and/or services, including those presently accused of infringement, in this District. Further on information and belief, Instagram markets, offers for sale, and/or sells products and/or services, including those presently accused of infringement, to customers and potential customers in this District.

21. This Court has personal jurisdiction, including general and specific jurisdiction, over WhatsApp. WhatsApp is incorporated under the laws of the State of Delaware. On information and belief, WhatsApp conducts business in, has continuous and systematic contacts with, and has committed acts of patent infringement in the State of Delaware and in this District, and has established minimum contacts with this forum state such that the exercise of jurisdiction over WhatsApp would not offend traditional notions of fair play and substantial justice. On information and belief, WhatsApp markets, offers for sale, sells, and/or uses products and/or services, including those presently accused of infringement, in this District. Further on information and belief, WhatsApp markets, offers for sale, and/or sells products and/or services, including those presently accused of infringement, to customers and potential customers in this District.

22. This Court has personal jurisdiction, including general and specific jurisdiction, over Facebook Technologies. Facebook Technologies is incorporated under the laws of the State of Delaware. On information and belief, Facebook Technologies conducts business in, has continuous and systematic contacts with, and has committed acts of patent infringement in the State of Delaware and in this District, and has established minimum contacts with this forum state such that the exercise of jurisdiction over Facebook Technologies would not offend traditional

notions of fair play and substantial justice. On information and belief, Facebook Technologies markets, offers for sale, sells, and/or uses products and/or services, including those presently accused of infringement, in this District.

23. Further on information and belief, Facebook Technologies markets, offers for sale, and/or sells products and/or services, including those presently accused of infringement, to customers and potential customers in this District.

24. Venue is proper in this Court as to Meta under 28 U.S.C. §§ 1391 and 1400(b). Meta resides in this District. By virtue of choosing to incorporate in the State of Delaware, Meta has received the benefits and responsibilities offered to and expected of Delaware corporations. Meta must accordingly assume responsibilities to Delaware and its citizens.

25. Venue is proper in this Court as to Instagram under 28 U.S.C. §§ 1391 and 1400(b). Instagram resides in this District. By virtue of choosing to incorporate in the State of Delaware, Instagram has received the benefits and responsibilities offered to and expected of Delaware corporations. Instagram must accordingly assume responsibilities to Delaware and its citizens.

26. Venue is proper in this Court as to WhatsApp under 28 U.S.C. §§ 1391 and 1400(b). WhatsApp resides in this District. By virtue of choosing to incorporate in the State of Delaware, WhatsApp has received the benefits and responsibilities offered to and expected of Delaware corporations. WhatsApp must accordingly assume responsibilities to Delaware and its citizens.

27. Venue is proper in this Court as to Facebook Technologies under 28 U.S.C. §§ 1391 and 1400(b).

28. Facebook Technologies resides in this District. By virtue of choosing to incorporate in the State of Delaware, Facebook Technologies has received the benefits and

responsibilities offered to and expected of Delaware corporations. Facebook Technologies must accordingly assume responsibilities to Delaware and its citizens.

29. Further, on information and belief, Defendants have offered and sold, and continue to offer and sell, their infringing products and services in this District. On information and belief, Defendants design, use, distribute, sell, and/or offer to sell the infringing products and services in this District as well as to consumers and businesses in this District.

30. On information and belief, Defendants are large companies with global reach and billions of dollars of annual revenue. Litigating this case in this District is not clearly inconvenient and would serve the interests of justice. Further, litigating this case in this District serves the interests of judicial economy, including in light of related pending lawsuits in this District.[10]

31. Joinder of Defendants into a single action for patent infringement is proper pursuant to 35 U.S.C. § 299(a). The joinder of Defendants is proper because Defendants have been and are acting in concert, and are otherwise jointly or severally, or otherwise in concert with respect to the acts of infringement, which arise out of the same transaction, occurrence, or series of transactions or occurrences relating to the making, using, importing into the United States, offering for sale, or selling of the Accused Products. Further, joinder of Defendants is proper because there are questions of fact common to all Defendants.

32. Meta's Quarterly Earnings Report for the quarter ending March 31, 2022, states that Meta "report[s] our financial results for our two reportable segments: Family of Apps (FoA) and Reality Labs (RL). FoA includes Facebook, Instagram, Messenger, WhatsApp, and other services. RL includes augmented and virtual reality related consumer hardware, software, and

[10] *See Starz Entertainment, LLC v. VL Collective IP, LLC*, No. 1:21-cv-01448-CFC (D. Del. filed Oct. 13, 2021); *see also VideoLabs, Inc. v. Netflix, Inc.*, No. 1:22-cv-00229-CFC (D. Del. filed Feb. 23, 2022).

content."[11] On information and belief, Meta does not separately report revenue or financial results for the Accused Products in its filings to the Securities and Exchange Commission, but rather reports combined performance and financial results.

33. The applications and products Facebook, Instagram, WhatsApp, Oculus, and Portal are "own[ed]" by Meta.[12] On information and belief, Meta "owns" as well as "operates" Facebook, Instagram, and WhatsApp, including the development and support of these services.[13] On information and belief, this operation includes "shar[ing] information about" users within the Meta "family of companies," including Facebook, WhatsApp, and Instagram, "to facilitate, support and integrate . . . activities and improve [these] services."[14]

---

[11] Ex. 13, *Meta Reports First Quarter 2022 Results*, META (Apr., 2022), available at https://investor.fb.com/investor-news/press-release-details/2022/Meta-Reports-First-Quarter-2022-Results/default.aspx (last accessed May 24, 2022).

[12] *See Ex. 14, Facebook goes all-in on its 'metaverse' ambitions, consolidates itself, Instagram, WhatsApp and others under Meta*, BUSINESS INSIDER (Oct., 2021), https://www.businessinsider.in/tech/news/facebook-goes-all-in-on-its-metaverse-ambitions-consolidates-itself-instagram-whatsapp-and-others-under-meta/articleshow/87352752.cms#:~:text=Facebook%20has%20officially%20announced%20the,or%20any%20other%20existing%20platform (last visited May 24, 2022) ("Facebook has officially announced the name for its corporate entity that owns Facebook, Instagram, WhatsApp and Oculus. Facebook chief Mark Zuckerberg, at the Facebook Connect event, announced that the holding company will be called 'Meta.'"); *see also* Ex. 15, *Facebook owns the four most downloaded apps of the decade*, BBC (Dec., 2019), https://www.bbc.com/news/technology-50838013 (last visited May 24, 2022) ("The four most downloaded apps of the decade," Facebook, Facebook Messenger, WhatsApp, and Instagram, "are all owned by Facebook.").

[13] *See* Ex. 16, *The Meta Companies*, META (2022), https://www.facebook.com/help/111814505650678 (last visited May 24, 2022) ("Meta owns and operates each of the companies listed below," listing "Facebook Technologies, LLC" and "WhatsApp LLC."); *see also* Ex. 17, *Data Policy*, META (2022), https://help.instagram.com/155833707900388 (last visited May 24, 2022) ("We share information globally, both internally within the Meta Companies, and externally with our partners . . . [t]hese data transfers are necessary to provide the services set forth in the Meta Terms and Instagram Terms and to globally operate and provide our Products to you.").

[14] *See* Ex. 18, *The Meta Companies*, META (2022), https://www.facebook.com/help/111814505650678 (last visited May 24, 2022).

34. Accordingly, joinder of Defendants into a single action for patent infringement is proper under 35 U.S.C. § 299(a).

THE VIDEOLABS PATENTS-IN-SUIT

A. U.S. Patent Nos. 8,139,878, 7,769,238, and 7,970,059

35. U.S. Patent No. 8,139,878 (the "'878 Patent"), titled "Picture Coding Method and Picture Decoding Method," issued on March 20, 2012. VL owns all rights and title to the '878 Patent, as necessary to bring this action. A true and correct copy of the '878 Patent is attached as Exhibit 1.

36. U.S. Patent No. 7,769,238 (the "'238 Patent"), titled "Picture Coding Method and Picture Decoding Method," issued on August 3, 2010. VL owns all rights and title to the '238 Patent, as necessary to bring this action. A true and correct copy of the '238 Patent is attached as Exhibit 2.

37. U.S. Patent No. 7,970,059 (the "'059 Patent"), titled "Variable Length Coding Method and Variable Length Decoding Method," issued on June 28, 2011. VL owns all rights and title to the '059 Patent, as necessary to bring this action. A true and correct copy of the '059 Patent is attached as Exhibit 3.

38. The '878, '238, and '059 Patents (collectively, the "Coding Patents") were developed by engineers at Panasonic, one of the largest consumer electronics companies at the time of the invention and a major innovator in Internet technologies. In 2002, when patent applications were first filed for the Coding Patents, Panasonic was a world leader in digital video technologies.[15] Panasonic developed video coding technologies and designed consumer

---

[15] *See* Ex. 19, *Annual Report 2002*, National/Panasonic Matsuhita Electric, available at https://www.annualreportowl.com/Panasonic/2002/Annual%20Report (last accessed May 24, 2022).

electronics — including TVs, DVD players, and memory cards — for storing, processing, and displaying video content.[16]

39. Native video content is massive. Modern digital video cameras used by premier television and movie studios capture images at incredibly fast rates (ranging from 30 frames per second up to 300 frames per second) and extremely high resolutions (up to “5k,” or 5120 x 2880, for a total size of 14,745,600 pixels per frame). Storing just an hour of this raw content requires more than 300 GB of memory.[17] Most modern TVs, laptops, tablets, and smartphones cannot possibly store and play such large files.

40. Even if they could, there would be little point from the perspective of on-demand content delivery: Internet speeds are far too slow to stream such massive video files. The fact is that transmitting high quality audiovisual content is simply not possible without powerful compression technologies. Streaming even just standard high-definition content (720p) requires network bandwidth of approximately 1.5 Gbps,[18] which is about 35 times faster than the average Internet speed in the United States.[19] “Encoding” and “decoding,” which respectively refer to the processes of compressing and decompressing content, are thus essential to applications such as video streaming, digital television, and videoconferencing.

41. Encoding video content allows the content to be made small for storage and

---

[16] *See id.*

[17] *See* Ex. 20, *How Many GB Is a 2 Hour 4k Movie?*, GAMINGSECTION (Nov. 2020), https://gamingsection.net/news/how-many-gb-is-a-2-hour-4k-movie/ (last visited May 24, 2022).

[18] *See* Ex. 21, Bryan Samis, *Back to Basics: GOPs Explained*, AWS MEDIA BLOG (May 28, 2020), https://aws.amazon.com/blogs/media/part-1-back-to-basics-gops-explained/ (last visited May 24, 2022).

[19] *See* Ex. 22, *Average U.S. Internet Speed is 42.86 Mbps*, ETI (Feb. 2, 2021), https://etisoftware.com/resources/blog/report-average-u-s-internet-speed-is-42-86-mbps/ (last visited May 24, 2022).

transmission, while decoding permits the viewer to watch high-quality content on his or her device. In addition to making real-time streaming of content possible, every incremental increase in compression efficiency yields substantial benefits to companies that store, process, transmit, or access video. For example, if a video streaming company can cut the size of each of its movie files in half, then it reasons that it only needs half the numbers of servers to store its movies, half the network bandwidth to transmit its movies, and half of all other related expenses, such as energy costs and staffing resources.

42. The Coding Patents describe breakthrough techniques for encoding and decoding audiovisual content so that it can be transmitted and stored with fewer resources. The patents vastly improve upon existing methods, and the core technology they describe has been used throughout the industry for years as the gold standard for coding content.

1. Background On Coding Technology

43. Video "coding" refers to both the encoding and decoding of video content. Video compression techniques minimize the size of the data that is sent between the encoder and the decoder by removing redundancies and imperceivable changes and then efficiently representing the remaining data for transmission.

44. Video is comprised of a series of frames. These frames are successively output to create the moving pictures that we recognize as video.

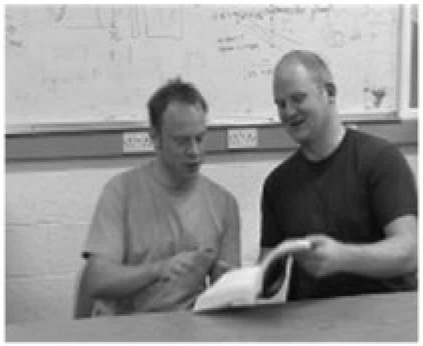


Figure 3.10 Frame 1

Figure 3.11 Frame 2

Ex. 23, Iain E. Richardson, *The H.264 Advanced Video Compression Standard* (2d. ed. 2010) (hereinafter "Richardson"), at 33.

45. In the early 2000s, certain techniques existed to reduce the amount of data needed to describe each frame without any loss in picture quality. For example, if there are a series of 50 white pixels in a row followed by 75 green pixels, then it is more efficient to store the fact that there are 50 white pixels followed by 75 green pixels than to store the value of all 125 pixels. This algorithm, which reduces the redundancy stemming from repeating pixels within a frame, yielded substantial benefits.

46. Video engineers also realized that, very often, not much changes between successive frames. In the images shown above, for example, the changes between frames 1 and 2 are largely concentrated in the area near the book. As a result, it is not necessary to send the complete data for every frame of a video. Instead, frames can be sent periodically at strategic points, such as when there is a scene change that creates major differences between successive frames. Those strategic frames — called "key frames" — could be used to "predict" other frames nearby in time by analyzing each frame and storing the differences from one frame to the next.

47. Further research yielded additional advances in what became known as predictive coding. Video engineers realized that it was advantageous to divide each frame into blocks, as shown below.

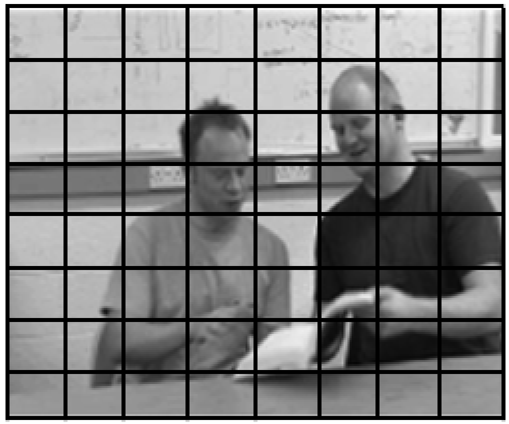


Figure 3.10 Frame 1

48. These blocks could be analyzed and used to predict the pixels in the same block in *surrounding* frames ("*inter*-picture prediction," also called "temporal compression"). Additionally, these blocks could be analyzed to predict the pixels in surrounding blocks in the *same* frame ("*intra*-picture prediction," also called "spatial compression"). While predictive coding does not always recreate frames that are identical to the original frames, the differences are so minor as to be imperceptible. For example, in the middle of an intense action sequence, a frame might display a pixel as blue even though it should be green because doing so enables the image to be represented more efficiently. This minor alteration from the original content will go unnoticed by the viewer, who is distracted by all the other activity.

49. Once redundancy in the video content has been minimized and imperceptible details have been streamlined, a process called "entropy encoding" further compresses the data by using as few bits to represent the data as possible, while still ensuring fidelity to the original visual content. This is achieved by allocating the fewest bits to commonly appearing bit sequences, and the most bits to infrequently occurring bit sequences. By way of analogy, when training your dog, the commands you use most frequently are likely the shortest, single-word commands, like "sit" and "no." But commands that you need less frequently may be longer, such as "wait for it" and

"roll over." In this way, over the course of a week, you expend fewer (verbal) resources. Entropy encoding applies this same principle to the bits of data that comprise video content.

50. There are standardized ways to represent sequences of bits, and depending on the type of entropy coding, these sequences are stored in either "coding tables" or "probability tables." Entropy coding involves selecting the optimal table for the data being transmitted and ensuring that the decoder knows the proper table to use when decoding the data.

51. It was in this context that the inventors of the Coding Patents made their contributions.

2. The '878 and '238 Patents

52. The '878 and '238 Patents are directed to a type of coding called "Context-based Adaptive Variable Length Coding," or "CAVLC." *See, e.g.*, Ex. 1, '878 Patent at col. 1, ll. 49-52.[20] The patents share the same specification and describe the same advances in coding technology but claim different aspects of the inventions. In particular, the claims of the '878 patent are directed to *encoding* audio and video content for transmission, while the claims of the '238 patent are directed to *decoding* the encoded content for playback.

53. When encoded, the image data in a particular image block is represented by, among other things, its "coefficients." *Id.* at col. 1, ll. 63-67; col. 7, ll. 38-43; col. 21, ll. 60-66; col. 25, ll. 29-36. Roughly speaking, larger coefficients for a block indicate a larger amount of changes in that block as compared with a reference block. *See id.* For many blocks, there are no such changes, and so all the coefficients have a value of zero. *See id.* at col. 21, ll. 60-66. The inventors of the '878 and '238 Patents recognized that these "zero-coefficient" blocks presented an opportunity for

[20] The '878 Patent and the '238 Patent share a specification. Accordingly, citations to the '878 Patent specification apply equally to the '238 Patent and vice versa; for simplicity, citations in the present section will be to the '878 Patent, but apply equally to the '238 Patent.

further compression. *See*, *e.g.*, *id.* at col. 1, ll. 49-52.

54. They realized that the decoder did not need to know every single time a zero-coefficient block existed; rather, the decoder needs to know only when blocks have *non-zero* coefficients. They devised a technique wherein data about zero-coefficient blocks are effectively not encoded at all, and only non-zero coefficient block data is stored and transmitted. *See*, *e.g.*, *id.* at col. 1, ll.49-52, 56-62; col. 1, l. 65 – col. 2, l. 10. The inventors thereby achieved nearly perfect compression for these zero-coefficient blocks by communicating them practically without sending any information whatsoever. *See id.* at col. 2, ll.11-14.

55. The inventors also made a substantial contribution to the efficiency of entropy coding. They recognized that the coefficients in neighboring blocks were a good predictor of the coefficients in the block being analyzed, and so could be used to select the optimal coding table for the block, yielding enhanced compression. *See, e.g.*, *id.* at col. 9, ll. 34-37; col. 13, ll. 4-11. Prior techniques lacked this level of sophistication. They did not take advantage of the predictive power provided by analyzing the coefficients of the surrounding blocks. They would also use the same coding table for both inter- and intra-predictive coding, which was inefficient because there could be significant differences between neighboring blocks in the current frame and blocks in subsequent frames. *See, e.g.*, *id.* at col. 1, ll. 33-38. Due to these limitations in the use of coding tables, compression efficiency in previously known entropy coding techniques would vary significantly between different types of content, and generally decreased as the quality of content increased. *Id.* at col. 1, ll. 39-44. These problems (and others) were overcome by the inventors of the ’878 and ’238 Patents.

3. The ’059 Patent

56. Similar to the ’878 and ’238 Patents, the ’059 Patent describes an advance in video compression that involves the novel use of tables. The ’059 Patent, however, uses a kind of entropy

coding referred to as "Context Adaptive Binary Arithmetic Coding," or "CABAC," that relies on probability tables. *See* Ex. 3, '059 Patent at col. 1, ll. 37-42.

57. CABAC achieves strong compression performance using arithmetic coding — a sophisticated approach to flexibly pack a string of numbers based on the probability that each next number will be a particular value. *See, e.g.*, *id.* at col. 1, ll. 11-14, 37-42. CABAC can optimize and adapt its selection of probability estimates for image data based on the context of the data. *See id.* at col. 2, ll. 33-51. CABAC uses "binary coding," which means that information can be represented only by a "0" or a "1." *See, e.g., id.* at col. 1, ll. 37-42. Once the data is binarized, it is arithmetically coded. *Id.* Arithmetic coding uses predefined probability tables to compress the data into its final bit stream before transmission. *See, e.g., id.* at col. 1, l. 60 – col. 2, l. 2. Probability tables are known to both the encoder and the decoder and referenced by a number (e.g., probability table #2). *See, e.g.*, *id.* at col. 6, ll.13-14.

58. Multiple probability tables are available when encoding content, and probability table selection is based on analyzing the data being arithmetically coded and the previously coded data. *See, e.g.*, *id. a*t col. 1, ll. 57-59; Fig. 2. The inventors of the '059 Patent recognized that it was advantageous to choose the probability table based on the current probability table and the absolute value of the data being coded. *See, e.g., id.* at col. 2, l. 52 – col. 3, l. 11. Further, a particular sequence of probability tables was determined in advance (e.g., probability table #2, #4, #3, #1), and the encoder (and thus the decoder) always proceed through the tables in that order, never reversing and stopping once the final table is reached. *See, e.g., id.* at col. 2, ll. 48-51. The inventors realized that following this approach takes advantage of the natural ordering of data and the context surrounding it. *See id.* at col. 2, ll. 52-56. The tables are adjusted to respond to the previous and current data being encoded, but importantly, there are limits on how and when the

probability table can change. *See id.* at col. 3, ll. 3-7. This adaptation, which gives higher priority to more recent observations, increases the efficiency of the coding. *See id.* at col. 3, ll. 7-11.

59. The innovations of all three Coding Patents provided a significant advance in compression that was recognized throughout the industry. In fact, the compression techniques of the Coding Patents are used in the ubiquitous video codec, H.264. H.264 was revolutionary in the video industry, as it provided a quantum leap of improvement over the video codecs that had previously been commonly used, such as Motion JPEG video and MPEG-2. In particular, H.264 "has an 80% lower bitrate than Motion JPEG video" and "the bitrate savings can be as much as 50% or more compared to MPEG-2."[21]

A. U.S. Patent No. 7,266,682

60. U.S. Patent No. 7,266,682 (the "'682 Patent"), titled "Method and System for Transmitting Data from a Transmitter to a Receiver and Transmitter and Receiver Therefore," issued on September 4, 2007. VL IP owns all rights and title to the '682 Patent, as necessary to bring this action. A true and correct copy of the '682 Patent is attached as Exhibit 4.

61. The original assignee of the '682 Patent was Siemens Corporate Research, Inc. ("Siemens"), one of the largest consumer electronics companies at the time of the invention and a major innovator in Internet technologies, including those related to transmitting audio and video. In 2001 alone, Siemens spent €6.6 billion in research and development.[22] At that time, Siemens recognized the importance of growing Internet technologies, and that security would be essential

[21] *See* Ex. 24, *What is H264 Encoding?*, BLACKBOX, https://www.blackbox.co.uk/gb-gb/page/38313/Resources/Technical-Resources/Black-Box-Explains/Multimedia/What-is-H264-video-encoding/ (last visited May 24, 2022).

[22] *See* Ex. 25, *Siemens Annual Report 2001*, available at https://web.lib.aalto.fi/fi/old/yrityspalvelin/pdf/2001/Esiemens.pdf (last accessed May 24, 2022).

to communicating and transmitting content over the Internet.[23]

62. It used to be very difficult to transmit real-time data (such as audio and video) over the Internet. In the mid-1990s the Real Time Transport Protocol (or RTP) was developed to handle such real-time Internet traffic in a standard way. RTP was developed by the Audio-Video Transport Working Group of the Internet Engineering Task Force (IETF) and first published in 1996 as RFC 1889.

63. RTP is designed for end-to-end, real-time transfer of streaming media. Indeed, RTP is regarded as the primary standard for audio/video transport in IP networks and is used with an associated profile and payload format. Today, RTP is used in communication and entertainment systems that involve streaming media, such as telephony, video teleconference applications including WebRTC, television services, and web-based push-to-talk features. The design of RTP is based on the architectural principle known as application-layer framing, where protocol functions are implemented in the application as opposed to the operating system's protocol stack.

64. However, typical RTP messaging was not very secure. The transmission networks carrying RTP payload packets were susceptible to security flaws at the transmitter, during transmission, and at the receiver. *See* Ex. 4, '682 Patent, col. 2, ln. 44 – col. 3, ln. 10. For example, an attacker could hide his/her attack within data during its transmission, and the receiver would — as a matter of course — decode it. *See id.* at col. 3, ll. 1-10. As another example, an attacker could interfere with the data transmission and read the data for itself. *See id.* at col. 2, ll. 47-54.

65. Ultimately, a security and authentication layer was introduced into RTP, known as Secure Real-Time Transport Protocol (SRTP), intended to provide encryption, message authentication and integrity, and replay attack protection to the RTP data. SRTP was published

[23] *See id.*

by the IETF in March 2004 as RFC 3711.[24]

66. Yet, years before the formalization and publication of SRTP, Siemens had already recognized, and solved, the RTP security problems. The '682 Patent prevents the security flaws by ensuring, at the receiver, that data being transmitted is not insecure or unwanted. *See* Ex. 35, '682 Patent, col. 3, ll. 45-48. Specifically, the transmitter is used to insert authentication data into the data packets before the data is transmitted. *See id.* at col. 3, ll. 49-52, col. 7, ll. 6-10. The authentication data is then transmitted together with the data packet and analyzed by the receiver to ensure that the transmitter and the receiver know each other. *See id.* at Fig. 3, col. 3, ll. 52-54, col. 5, ll. 32-36, col. 7, ll. 10-17. If the receiver knows the transmitter, the data is processed; if not, the data is rejected. *See id.* at col. 7, ll. 14-21.

67. Data transmission networks operate on a variety of levels of discreteness. Many people of skill in the art refer to the seven-layer OSI reference model to abstract the layers of functionality in these networks. *See, e.g., id.* at col. 1, ll. 19-45. The well-known OSI model breaks transmission networks into seven layers, each of which has a different functionality. *Id.* at col. 1, ll. 21-25. The lowest layer, layer 1, corresponds to the physical layer, where data and messages are transmitted from the transmitter to the receiver using a physical protocol. *Id.* at col. 1, ll.25-28. Each subsequent layer builds on the layer(s) before it, all the way up to layer 7, which corresponds to an application layer using an application protocol. *Id.* at col. 1, ll. 36-37. The '682 Patent optimizes the security of data transmission by performing authentication in the application layer. *See id.* at col. 3, ll. 60 – col. 4, ll. 10.

68. Authenticating in the application layer provides myriad benefits to the user and the

[24] *See* Ex. 83, *The Secure Real-time Transport Protocol,* IETF (Mar. 2004) (hereinafter "RFC 3711"), at 3, available at https://datatracker.ietf.org/doc/html/rfc3711 (last visited May 24, 2022).

network. For one, it permits reduced size of authentication data, including because it allows the transmitter and receiver to share secret information, such as a key, which can be economically used to generate message authentications. *See id.* at col. 3, ln. 60 – col. 4, ln. 10, col. 7, ll. 30-34.

69. Another benefit of the ’682 Patent inventor’s insight that authenticating should take place in the application layer is that it increases the transmission system’s performance and reduces the implementation complexity of the system. *See* ’682 Patent at col. 4, ll. 15-21. Authenticating in the application layer provides real time availability of the data. *See id.* at col. 3, ln. 60 – col. 4, ln. 10. Once the receiver determines that the transmitter is known, the message is immediately accepted. *See id.* at col. 4, ll. 3-7. If the receiver determines that the transmitter is not known, the message is immediately rejected; no buffering or loading are performed, and no further action is needed. *See id.* at col. 4, ll. 7-9.

70. The ’682 patent’s claims recite an invention that involves more than the performance of well-understood, routine, and conventional activities previously known to the person of ordinary skill in the art (“POSITA”). Ex. 86 (Stubblebine Decl.), ¶23. For example, claim 1 of the ’682 patent recites:

> 1. A method for transmitting data from a transmitter to a receiver, comprising:
> *providing transmitter-to-receiver authentication at a Real Time Transport Protocol (RTP) packet level as an application protocol on an application layer by inserting, at the transmitter, authentication data at end of a whole RTP packet payload*;
> ascertaining, by the receiver, whether the receiver knows the transmitter based on the RTP packet level authentication data; and
> accepting, by the receiver, the whole RTP packet payload, if the receiver knows the transmitter, and otherwise rejecting the whole RTP packet payload.

Ex. 4, ’682 patent, claim 1 (emphasis added). The italicized portion of claim 1 recites a non-conventional combination of elements that collectively require providing authentication at a particular place in a protocol stack (*i.e.*, "at a [RTP] packet level") as part of a protocol for a particular layer of the protocol stack (*i.e.*, "as an application protocol on an application layer") in a particular way (*i.e.*, "by inserting, at the transmitter, authentication data at end of a whole RTP packet payload"). *Id.*; Ex. 86 (Stubblebine Decl.), ¶23. This combination of elements (and its combination with the rest of claim 1) was not well-known, routine, or conventional to the POSITA for providing transmitter-to-receiver authentication. Ex. 86 (Stubblebine Decl.), ¶23.

71. As an initial matter, the POSITA would understand that "layers" refer to layers of the OSI reference model discussed in the '682 patent's specification, which includes an application layer, presentation layer, session layer, transport layer, network layer, data link layer, and physical layer. Ex. 4, '682 patent at col 1, ln. 19-61; Ex. 86 (Stubblebine Decl.), ¶24. The POSITA would also understand that in the context of the '682 patent, as a variation of the OSI model, "application layer" refers to "the layers above the transport layer (layers above 4)," and "application protocol" refers to "the protocol for communication between a transmitter application layer situated above the transport layer and a receiver application layer situated above the transport layer." Ex. 4, '682 patent at col 2, ln. 6-11; Ex. 86 (Stubblebine Decl.), ¶24. Thus, as the POSITA would understand and appreciate, particularly in view of the patent's lexicography, the claims of the '682 patent focus on providing authentication at an *RTP packet level* as an *application* protocol on an *application* layer, where authentication is specifically provided above (not at or below) the transport layer. Ex. 86 (Stubblebine Decl.), ¶24. This combination of elements was not well-known, routine, or conventional to the POSITA. Ex. 86 (Stubblebine Decl.), ¶24.

72. Conventional techniques of providing secure transmissions did not involve the claimed combination of "providing transmitter-to-receiver authentication at a Real Time Transport

Protocol (RTP) packet level as an application protocol on an application layer." Ex. 86 (Stubblebine Decl.), ¶25. Instead, known techniques involved providing security in different places at different protocol stack layers and in different ways. *Id*. For instance, some approaches utilized techniques like Internet Protocol Security ("IPSEC") and IP Encapsulating Security Payload ("ESP"), which attempted to secure data transmissions using mechanisms at the network layer (rather than above the transport layer) and could not ensure protection at the application layer. *See* Ex. 4, '682 patent at col. 3, ln. 16-41; Ex. 86 (Stubblebine Decl.), ¶25. It was also known to use transport protocols with security features, but, consistent with the conventional layer-respecting approach of the OSI model (whereby a lower layer provides "services" to an upper layer through a separating interface such that the upper layer does not need to concern itself with the functionalities of the lower layer), such transport protocols did not provide authentication at an RTP packet level as an application protocol on an application layer. Ex. 86 (Stubblebine Decl.), ¶25. For example, the RTP specification itself, while providing for encryption at the RTP packet level, expressly did not define authentication or message integrity services for the RTP packet level (much less "at a Real Time Transport Protocol (RTP) packet level as an application protocol on an application layer"). Ex. 87 (RFC 1889) at 50; Ex. 86 (Stubblebine Decl.), ¶25. As another example, a reference titled "Secure Transport Protocols for High-Speed Networks" by Basturk et al. distinguishes security features according to their associated layer (*i.e.*, Basturk distinguishes "application" level security from "transport" level security) and proposes a secure transport protocol that provides security at the transport layer rather than as an application protocol at the application layer, which is consistent with conventional layer-respecting approaches. Ex. 88 at 3-5; Ex. 86 (Stubblebine Decl.), ¶25.

73. The combination of elements recited in claim 1 of the '682 patent is an inventive

approach that diverges from conventional techniques by providing transmitter-to-receiver authentication "at a Real Time Transport Protocol (RTP) packet level" by inserting authentication data "at end of a whole RTP packet payload," *i.e.*, at a specific location, "as an application protocol on an application layer," *i.e.*, as part of a protocol for a particular layer of the protocol stack, which was not well-understood, routine, or conventional to the POSITA. Ex. 86 (Stubblebine Decl.), ¶26. Indeed, the RTP specification itself expected that authentication and integrity services would be provided by *lower* layer protocols—thus, lower than the application layer—in the future. Ex. 87 at 50; Ex. 86 (Stubblebine Decl.), ¶26. Furthermore, the authors of Basturk taught away from and discouraged modifying an application layer protocol to include the security features of Basturk's transport protocol, indicating that doing so would be "poor design," "belies the layered communication structure," would "adversely affect performance," would "mak[e] implementation significantly harder," would "duplicate" functionality, and "degrade performance." Ex. 88 at 4 n.3, 12; Ex. 86 (Stubblebine Decl.), ¶26. This further demonstrates that it was not well-known, routine, or conventional to the POSITA to "provid[e] transmitter-to-receiver authentication at a Real Time Transport Protocol (RTP) packet level as an application protocol on an application layer by inserting, at the transmitter, authentication data at end of a whole RTP packet payload" as recited in claim 1 of the '682 patent. Ex. 86 (Stubblebine Decl.), ¶26. To the contrary, this claimed combination was highly innovative and provided a specific and concrete solution to problems that had not been adequately addressed. *Id.*

74. The combination of elements recited in claim 1 of the '682 patent were not conventional authentication techniques. Ex. 86 (Stubblebine Decl.), ¶27. They were innovative and gave rise to the technical improvements described in the '682 patent's specification. *Id.* Claim 1's specific combination of elements results in the achievement of secure transport of Real Time

Transport Protocol (RTP) packets, and is beneficial to real-time performance and availability of the system subject to attacks—a technological improvement. *See* Ex. 4, '682 patent at col. 3, ln. 45 – col. 4, ln. 10; Ex. 86 (Stubblebine Decl.), ¶27. As the '682 patent explains, the invention solves the security problem by "extend[ing] the data" specifically of the application layer, which includes the data actually being communicated but excludes the extra administrative data of lower layers, "by using an application protocol on the application layer." *Id*. This enables the patented invention to achieve authentication using smaller data packets of the application layer (rather than larger packets of lower layers) to solve security problems left open by conventional techniques (including preventing denial-of-service attacks). *Id*. But, as the claims make clear, the authentication provided using the application protocol on the application layer is provided specifically "at a Real Time Transport Protocol (RTP) packet level … by inserting, at the transmitter, authentication data at end of a whole RTP packet payload." *Id*. That is, the claim specifically requires that the *application protocol* be used to provide authentication at the *Real Time Transport Protocol (RTP) packet level* by inserting authentication data at the end of the whole RTP packet payload. *Id*. This was not well-known, routine, or conventional to the POSITA. *Id*. The POSITA would have understood that conventional approaches to authentication at the time would not have involved "providing transmitter-to-receiver authentication at" the lower "Real Time Transport Protocol (RTP) packet level" specifically "as an application protocol on an application layer by inserting, at the transmitter, authentication data at end of a whole RTP packet payload" and "ascertaining, by the receiver, whether the receiver knows the transmitter based on the RTP packet level authentication data." *Id*. Indeed, the patentee distinguished this combination of elements from the "closest prior art" reference raised by the USPTO during prosecution of the '682 patent ("Sengodan"), which the Examiner agreed did not disclose or render obvious the combination of elements in claim 1. Ex. 89 at 583-585, 603-604 ("[A]t the time of Sengodan's invention, there is no mechanism for authentication at the RTP level and the claimed invention of inserting the authentication data at the end of the whole RTP packet is considered a different technique from inserting the authentication data at the end of the [Sengodan] packets. The cited prior art, either singularly or in combination fail to anticipate or render the claimed invention obvious."); Ex. 86 (Stubblebine Decl.), ¶27. This is consistent with a POSITA's understanding of the relevant art at the time of the '682 patent. *Id.*

75. The non-conventional approach claimed in the '682 patent provides enhanced security and allows the receiver to ascertain whether it knows the transmitter before accepting the whole RTP packet payload without compromising real-time performance and availability, *see* Ex. 4, '682 patent at col. 3, ln. 56 – col. 4, ln. 10, which was a problem with reliable delivery protocols that involved additional handshake procedures (like the well-known TCP protocol) that were avoided when real-time performance was needed. Ex. 86 (Stubblebine Decl.), ¶28. As the patentee explained during prosecution of the '682 patent, applications with real-time properties were of a special class that needed to be "as free of delays as possible," thus protocols with "complex" security features or "additional handshakes" were "not applicable to real-time capable communication protocols." Ex. 89 at 488-489; Ex. 86 (Stubblebine Decl.), ¶28. This is consistent with the POSITA's understanding of the state of the art at the time of the '682 patent. *Id.* Moreover, when the whole RTP packet is rejected, unwanted data is not buffered at the application level, which reduces jitter as buffers become full. *Id.*

B. U.S. Patent No. 7,436,980

76. U.S. Patent No. 7,436,980 (the "'980 Patent"), titled "Graphical Object Models For Detection And Tracking," issued on October 14, 2008. VL IP owns all rights and title to the '980

Patent, as necessary to bring this action. A true and correct copy of the '980 Patent is attached as Exhibit 5.

77. The original assignee of the '980 Patent is Siemens, one of the largest consumer electronics companies at the time of the invention and a major innovator in Internet technologies, including those related to audio and video. In 2001 alone, Siemens spent €5.1 billion in research and development, with more than 55% in "information and communications and automation and control technologies."[25] At that time, Siemens recognized the importance of computer vision, and of object detection and tracking in particular, and that it would have wide applicability in a variety of industries, including security, automation, medical, and automotive.

78. The '980 Patent is concerned with object detection. Object detection — determining and tracking what is within images and videos — is a vital part of the web and specifically of social media and communications platforms. "The goal of object detection is to predict a set of bounding boxes and category labels for each object of interest."[26] Meta uses object detection on facebook.com to, for example, improve content moderation, target advertising, describe objects for people with visual impairments, and, for many years, identify users in photos and videos. And advances in detection could help Meta "in tagging products for associated display within video content" as well as "applications related to AR and visual tools that could lead to much more advanced, more immersive Facebook functions."[27] At base, "[t]he objective of object

[25] *See* Ex. 68, *Siemens Annual Report 2004*, available at https://www.siemens.com/investor/pool/en/investor_relations/downloadcenter/e04_00_gb2004_1230305.pdf (last accessed May 24, 2022).

[26] Ex. 26, Nicolas Carion, et al., *End-to-End Object Detection with Transformers* (May 28, 2020).

[27] Ex. 27, *Facebook Outlines Advances in Computer Vision and Object Identification Tech*, SOCIAL MEDIA TODAY (Apr. 30, 2021), https://www.socialmediatoday.com/news/facebook-outlines-advances-in-computer-vision-and-object-identification-tec/599399 (last visited May 24, 2022).

detection is to develop computational models and techniques that provide one of the most basic pieces of information needed by computer vision applications: What objects are where?"[28]

79. Yet, object detection presents fundamental challenges. What seems simple for a human is a complex exercise for a computer, which in the past required great computational resources. It is difficult to program a computer to know a cat from a dog or a car from a boat, and, most importantly, to do so in a way that doesn't require the programmer to manually identify all possible objects. Object detection suffers from even more challenges: partially covered objects, rotated objects, scaled objects (such as a small house in the background of a photo), etc. Video presents the same difficulties and more. It was necessary to develop a way to quickly and confidently identify objects in images and videos.

80. To address these problems, the inventors of the '980 Patent came up with a method to detect and track an object within a video by measuring the object as a collection of components in each image and, using a spatio-temporal model, determining the probability that the object is in any given image that makes up the video. This approach was completely unheard of in the field of object detection, but after the invention of the '980 Patent, its method was widely adopted. While it was relatively common to "represent objects as collections of features with distinctive appearance, spatial extent, and position," it was unknown in the art to use those components in object detection. *See* Ex. 5, '980 Patent at col. 1, ll. 15-21. As is seen in Exhibits 26 and 28, the '980 patent's solution to the object detection problem was unknown at the time of the invention, and even, but for the '980 patent, for years after. They demonstrate the unique problems inherent in object detection and highlight the inventive and concrete nature of the '980 patent's solutions.

81. Critical to the insight of the '980 Patent is the use of a spatio-temporal model. "The

[28] Ex. 28, Zhengxia Zou, et al., *Object Detection in 20 Years* (May 16, 2019).

spatio-temporal model is a graphical model comprising nodes corresponding to each of the collection of components and to the object." '980 Patent at col. 1, ll. 41-42. "Each node in the graph represents either the object or a component of the object at time t." '980 Patent at col. 3, ll. 47-50. What that means is that the object (such as a pedestrian) and the components of that object (which are discussed in more detail below) are mapped using both "the temporal compatibility of object state between frames" (its state from second to second) and "the spatial compatibility of the object and its components" within a single frame. Hence, both temporal and spatial mapping.

82. The inventors of the '980 patent also recognized the value in taking a multi-layer approach. *See* '980 Patent at col. 4, ll. 18-19 ("Building the Graphical Model; For a single frame, objects are represented using a two-later spatial graphical model."). Before the invention, images and videos were analyzed as a single layer — that is, the image as it appears. In the '980 patent, the image can artificially be made into two layers: "The course, object [layer], [which] corresponds to an entire appearance model of the object" and "[t]he fine, component [layer], [which] includes a set of loosely connected parts." '980 Patent at col. 4, ll. 20-23. The object layer is in turn "connected to all constituent components." '980 Patent at col. 4, ll. 23-24. An example illustrates the two-layer innovation. For a pedestrian, the components might be the head, the left arm, the right arm, and the legs. Thus, the component layer would be made up of nodes corresponding to those components, while the object layer is just the object — the pedestrian. "Having a two-layer graphical model allows the inference process to reason explicitly about the object as a whole . . . and reduce the complexity of the graphical model by allowing the assumption of the conditional independence of components" '980 Patent at col. 4, ll. 37-41. In addition to speeding up object detection, the patented method, particularly the possibility of overlapping components, "facilitates detection of complex articulated objects as well as helps in handling

partial object occlusions [e.g., partially covered items] or local illumination changes [e.g., changes in light]." '980 Patent at col. 2, ll. 51-54.

83. The inventors of the '980 patent also developed ways to use the two-layer, component model to determine the probability that an object is in the image, using a mixture of M_{ij} Gaussians, an iterative Expectation-Maximation (EM) Method, an interactive Expectation-Maximation (EM) Method, and a non-parametric belief propagation PAMPAS, among other techniques. '980 Patent at col. 4, ll. 49 - col. 7, ll. 44. Using these innovations allow companies (such as the Meta Companies) to quickly, accurately, and cheaply identify and track objects in videos.

ACCUSED PRODUCTS

A. Meta Companies Coding Patents Accused Products

84. The "Meta Coding Patents Accused Products" refers to all Meta products, services, and functionalities that use H.264 entropy coding. This includes, for example, all versions and implementations of www.facebook.com, Facebook applications, Facebook Live, Facebook Watch, Workplace, and Facebook Messenger. It also includes the encoding/decoding of video (using H.264) by Instagram, WhatsApp, or Facebook Technologies that is at the direction, request, or control of Meta.

85. The "Instagram Coding Patents Accused Products" refers to all Instagram products, services, and functionalities that use H.264 entropy coding. This includes, for example, all versions and implementations of Instagram applications, Instagram Live, Instagram Stories, and Instagram Messenger. It also includes the encoding/decoding of video (using H.264) by a Meta Company that is at the direction, request, or control of Instagram.

86. On information and belief, in relevant aspects for the Instagram Coding Patents

Accused Products, Meta and Instagram act in concert with one another.

87. The "WhatsApp Coding Patents Accused Products" refers to all WhatsApp products, services, and functionalities that use H.264 entropy coding. This includes, for example, all versions and implementations of WhatsApp applications, WhatsApp Messaging, and WhatsApp voice and video calls. It also includes the encoding/decoding of video (using H.264) by a Meta Company that is at the direction, request, or control of WhatsApp.

88. On information and belief, in relevant aspects for the WhatsApp Coding Patents Accused Products, Meta and WhatsApp act in concert with one another.

89. The "Facebook Technologies Coding Patents Accused Products" refers to all Oculus (Quest) and Portal products, services, and functionalities that use H.264 entropy coding. This includes, for example, all versions and implementations of Oculus Video, Oculus Rift, Oculus Rift S, Oculus Go, Oculus Quest, Meta Quest 2, Portal, Portal Mini, Portal+, Portal TV, and Portal Go. It also includes the encoding/decoding of video (using H.264) by a Meta Company that is at the direction, request, or control of Facebook Technologies.

90. On information and belief, in relevant aspects for the Facebook Technologies Coding Patents Accused Products, Meta and Facebook Technologies act in concert with one another.

91. H.264 is the name for technology described in an industry standard that is widely used to encode and decode streaming video.

92. H.264 reduces the file size of video files without any loss in quality of video, enabling companies to stream video in higher quality given the same network bandwidth.

93. H.264 focuses on the coding of the picture portions of the video content.

94. To this end, H.264 defines a format, or syntax, for compressed video and a method

for decoding this syntax to produce a displayable video sequence. An H.264 video encoder carries out prediction, transform, and encoding processes to produce a compressed H.264 bitstream. An H.264 video decoder carries out the complementary processes of decoding, inverse transform, and reconstruction to produce a decoded video sequence.

95. H.264 has been the dominant industry standard for compressing video for applications such as digital television, DVD video, video conferencing, and Internet video streaming. Standardizing video compression made it possible for products from different manufacturers to inter-operate. Recommendation H.264: Advanced Video Coding is a video decoding standard published by the international standards bodies ITU-T (International Telecommunication Union) and ISO/IEC (International Organisation for Standardisation / International Electrotechnical Commission) (attached as Ex. 29). It defines a format (syntax) for compressed video and a method for decoding this syntax to produce a displayable video sequence. Products that support H.264 encoding and decoding are compliant with the H.264 standard.

96. In H.264, picture data can be coded using context-adaptive binary arithmetic coding ("CABAC") or context-adaptive variable-length coding ("CAVLC"). Each coding algorithm provides different benefits.

97. CABAC highly compresses the picture data, but is computationally expensive to decode, while CAVLC is lower-complexity and more efficient.

98. The Meta Companies use H.264 coding to efficiently and seamlessly deliver video to their customers.

99. On information and belief, and according to Meta, the Meta products use H.264

and support coding in H.264.[29]

100. On information and belief, and according to Meta and Instagram, the Instagram products use H.264 and support coding in H.264.[30]

101. On information and belief, and according to Meta and WhatsApp, the WhatsApp products use H.264 and support coding in H.264.[31]

102. On information and belief, and according to Meta and Facebook Technologies, the Facebook Technologies products use H.264 and support coding in H.264.[32]

---

[29] *See* Ex. 30, *How Facebook encodes your videos*, ENGINEERING AT META (April 5, 2021), https://engineering.fb.com/2021/04/05/video-engineering/how-facebook-encodes-your-videos (last visited May 24, 2022); Ex. 31, *Live Video API-Reference*, META FOR DEVELOPERS (2022), https://developers.facebook.com/docs/live-video-api/reference (last visited May 24, 2022); Ex. 69, *Automatic Encoder Configuration API*, META FOR DEVELOPERS (2020), https://developers.facebook.com/docs/live-video-api/guides/automatic-encoder-configuration-api (last visited May 24, 2022); Ex. 32, *Graph API Version-Video Format*, META FOR DEVELOPERS (2020), https://developers.facebook.com/docs/graph-api/reference/video-format (last visited May 24, 2022); Ex. 33, *What video file formats can I upload on Facebook?*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/218673814818907 (last visited May 24, 2022); Ex. 34 *How do I optimize the quality of videos when uploading them to Workplace?*, WORKPLACE (2022), https://www.facebook.com/help/work/touch/885144448260143 (last visited May 24, 2022).

[30] *See* Ex. 35, *Sharing to Stories*, META FOR DEVELOPERS (2020), https://developers.facebook.com/docs/instagram/sharing-to-stories (last visited May 24, 2022); Ex. 36, *The Best Instagram Video Format And Specifications In 2022*, OBERLO (Dec. 29, 2021), https://www.oberlo.com/blog/best-instagram-video-format (last visited May 24, 2022).

[31] Ex. 37, *WhatsApp Business On-Premises API-Media*, META FOR DEVELOPERS (2020), https://developers.facebook.com/docs/whatsapp/on-premises/reference/media (last visited May 24, 2022); Ex. 38, *WhatsApp Business On-Premises API-Media*, META FOR DEVELOPERS (2020), https://developers.facebook.com/docs/whatsapp/api/messages/media (last visited May 24, 2022); Ex. 39, *WhatsApp Business Platform- On-Premises API Changelog Archive*, META FOR DEVELOPERS (2020), https://developers.facebook.com/docs/whatsapp/changelog/archive (last visited May 24, 2022); Ex. 40, *What Video does WhatsApp Support?*, LEAWO (2022), https://www.leawo.org/entips/whatapp-supported-video-1372.html (last visited May 24, 2022); Ex. 41, WABetaInfo (@WABetaInfo) Twitter (Sept. 24, 2016, 5:06), https://twitter.com/wabetainfo/status/779789143928926209.

[32] Ex. 42, *Encoding High-Resolution 360 and 180 Video for Oculus Quest and Oculus Go*, OCULUS FOR DEVELOPERS (Oct. 15, 2019), https://creator.oculus.com/blog/encoding-high-resolution-360-and-180-video-for-oculus-go (last visited May 24, 2022); Ex. 43, *How does*

B. Meta Companies '682 Accused Products

103. The "Meta '682 Accused Products" refers to all Meta products, services, and functionalities that implement, in whole or in part, Web Real-Time Communication ("WebRTC") or Secure Real-Time Transport Protocol (SRTP). This includes, for example, all versions and implementations of www.facebook.com, Facebook applications, Facebook Live, Facebook Watch, Workplace, and Facebook Messenger. It also includes the use of WebRTC or SRTP by any other Meta Company that is at the direction, request, or control of Meta.

104. The "Instagram '682 Accused Products" refers to all Meta products, services, and functionalities that implement, in whole or in part, WebRTC or SRTP. This includes, for example, all versions and implementations of Instagram applications, Instagram Live, Instagram Live Video Chat, Instagram Stories, and Instagram Messenger. It also includes the use of WebRTC or SRTP by any other Meta Company that is at the direction, request, or control of Instagram.

105. On information and belief, in relevant aspects for the Instagram '682 Accused Products, Meta and Instagram act in concert with one another.

106. The "WhatsApp '682 Accused Products" refers to all WhatsApp products, services, and functionalities that implement, in whole or in part, WebRTC or SRTP. This includes, for example, all versions and implementations of WhatsApp applications, WhatsApp Messaging, and WhatsApp voice and video calls. It also includes the use of WebRTC or SRTP by any other Meta

---

Oculus Link Work? The Architecture, Pipeline and AADT Explained, OCULUS FOR DEVELOPERS (Nov. 22, 2019), https://developer.oculus.com/blog/how-does-oculus-link-work-the-architecture-pipeline-and-aadt-explained (last visited May 24, 2022); Ex. 44, *Oculus HD*, accessible at https://silentsentinel.com/wp-content/uploads/2021/06/Oculus-HD-Datasheet-V4.1.pdf (last accessed May 24, 2020); Ex. 45, *Publishing to Oculus Video*, OCULUS CREATORS, https://creator.oculus.com/distribute/oculus-video/#:~:text=3840x1920%20resolution%20at%2030%20FPS,2048x2048%20resolution%20at%2060%20FPS (last visited May 24, 2022).

Company that is at the direction, request, or control of WhatsApp.

107. On information and belief, in relevant aspects for the WhatsApp ’682 Patents Accused Products, Meta and WhatsApp act in concert with one another.

108. The “Facebook Technologies ’682 Accused Products” refers to all Facebook Technologies ’682 products, services, and functionalities that implement, in whole or in part, WebRTC or SRTP. This includes, for example, all versions and implementations of Oculus Video, Oculus Rift, Oculus Rift S, Oculus Go, Oculus Quest, Meta Quest 2, Portal, Portal Mini, Portal+, Portal TV, and Portal Go. It also includes the use of WebRTC or SRTP by any other Meta Company that is at the direction, request, or control of Facebook Technologies.

109. On information and belief, in relevant aspects for the Facebook Technologies ’682 Patents Accused Products, Meta and Facebook Technologies act in concert with one another.

110. The “Meta Companies ’682 Accused Products” refers to the Meta ’682 Accused Products, the Instagram ’682 Accused Products, the WhatsApp ’682 Accused Products, and the Facebook Technologies’682 Accused Products, collectively.

111. WebRTC facilitates secure, real-time communication by permitting a web browser to request backend resources using available application programming interfaces, or APIs.

112. The Meta Companies use WebRTC to provide secure, real-time communication across browsers and mobile applications using WebRTC’s APIs in the provision of digital video.

113. On information and belief, all of the Meta Companies ’682 Accused Products use WebRTC, including Facebook Messenger Video Chat, Facebook Messenger Group Video Chat, Facebook Live, and VR Chat.[33] “Facebook’s mobile app and web client (accessible through a web

[33] SFHTML5, *Facebook Messenger RTC – The Challenges and Opportunities of Scale*, YouTube (Oct. 27, 2017), https://www.youtube.com/watch?v=F7UWvflUZocat, at 0:49-2:00;

browser) are both powered by WebRTC. By using Web Real-Time Communications, Messenger has brought voice and video calls to its users, and more recently, allows for co-broadcasting via Facebook Live. Additionally, Facebook has also incorporated WebRTC in VR Chat for video calls in Oculus, Workplace by Facebook, and IG Live Video Chat."[34] "[A]s a result, they have made communication across boundaries quicker and more reliable."[35]

114. "Over 400 million people call or video chat one another through Facebook Messenger using WebRTC every month."[36] "More than 1.3B people use Messenger every month," with "400M people us[ing] voice and video chat every month."[37] Facebook Messenger is the "2nd most popular iOS app of all time" and has "1B downloads on Android."[38]

115. Meta wants "to have the best media quality for all users."[39] Given the scale of Meta's products, this is a "really hard challenge."[40] Meta bases its peer-to-peer communications

---

Ex. 49, *WebRTC and SIP: 10 Applications that Might Surprise You*, TELNYX (Sept. 17, 2020), https://telnyx.com/resources/5-applications-that-demonstrate-the-power-of-webrtc-and-sip (last visited May 24, 2022); Ex. 50, *10 Massive Applications Using WebRTC*, BLOGGEEK.ME (Dec. 18, 2017), https://bloggeek.me/massive-applications-using-webrtc (last visited May 24, 2022); Ex. 51, *Oculus Browser Developer Release Notes*, OCULUS FOR DEVELOPERS, https://developer.oculus.com/documentation/web/browser-release-notes (last visited May 24, 2022); Ex. 52, *When Whatsapp Adds Voice Calling, WebRTC will be Used by 500 Million Users*, BLOGGEEK.ME (MAY 22, 2014) https://bloggeek.me/whatsapp-voice-calling-webrtc (last visited May 24, 2022).

[34] Ex. 53, *8 Powerful Applications Built Using WebRTC*, UNITED WORLD TELECOM BLOG https://www.unitedworldtelecom.com/learn/webrtc-applications (last visited May 24, 2022).

[35] *Id.*

[36] SFHTML5, *Facebook Messenger RTC – The Challenges and Opportunities of Scale*, YouTube (Oct. 27, 2017), https://www.youtube.com/watch?v=F7UWvflUZoc.

[37] *Id.* at 2:13.

[38] *Id.*

[39] *Id.* at 6:30

[40] *Id.* at 6:39.

on "WebRTC" as part of its strategy.[41] "Facebook is worth nearly $245 billion, when it makes use of a new technology such as WebRTC to enable audio/video calls, 600 million users make that technology (WebRTC) important. Not only does WebRTC gain incredible visibility in this use case, it also gets Facebook's vote of confidence opening the door for many others in the arena to follow suit."[42]

116. According to Meta, it "rewr[o]te [its] existing library from scratch using the latest version of the open source WebRTC library.[43] In fact, Facebook Messenger has been described as "one of, if not the, largest group of users using a WebRTC based applications."[44] Similarly, "WhatsApp has grown into a global messaging platform connecting users from around the globe quickly" and its "Android and iOS apps heavily use WebRTC . . . ."[45]

C. Meta Companies '980 Accused Products

117. "The Meta Companies '980 Accused Products" refers to all Meta, Instagram, WhatsApp, or Facebook Technologies products, services, functionalities, and features that

---

[41] *Id.* at 2:44; Ex. 54, *Facebook is right to defend the security of Messenger, and the solution is adding more security*, ACCESS NOW (August 23, 2018), https://www.accessnow.org/facebook-is-right-to-defend-the-security-of-messenger-and-the-solution-is-more-security (last visited May 24, 2022) ("Facebook's Messenger, including the voice (VoIP – Voice over Internet Protocol) functionality, utilizes an open protocol called WebRTC (Web Real Time Communications).").

[42] Ex. 55, *Why Does Facebook Use WebRTC?*, VERAVIEW BLOG (Jul. 16, 2015), http://veraview.com/what-is-webrtc (last visited May 24, 2022).

[43] *See* Ex. 56, *A smaller, faster video calling library for our apps*, ENGINEERING AT META (Dec. 21, 2020), https://engineering.fb.com/2020/12/21/video-engineering/rsys (last visited May 24, 2022).

[44] Ex. 57, *Facebook Messenger – The Biggest WebRTC App?*, WEBRTC WORLD (Jan. 13, 2016), https://www.webrtcworld.com/topics/webrtc-world/articles/415961-facebook-messenger-biggest-webrtc-app.htm (last visited May 24, 2022).

[45] Ex. 53, *8 Powerful Applications Built Using WebRTC*, UNITED WORLD TELECOM BLOG https://www.unitedworldtelecom.com/learn/webrtc-applications (last visited May 24, 2022).

implement, in whole or in part, object identification abilities. This includes, for example, all versions and implementations of Detectron, Detectron2, and D2Go.

118. On information and belief, in relevant aspects for the Meta Companies '980 Accused Products, the Meta Companies act in concert with one another.

119. "Detectron2 is Facebook AI Research's next generation library that provides state-of-the-art detection and segmentation algorithms . . . . It supports a number of computer vision research projects and production applications in Facebook."[46] "Since its release in 2018, the Detectron object detection platform has become one of Facebook AI Research (FAIR)'s most widely adopted open source projects."[47] The Meta Companies "built Detectron2 to meet the research needs of Facebook AI and to provide the foundation for object detection in production use cases at Facebook."[48] The Meta Companies are "expanding on Detectron2 with the introduction of Detectron2Go (D2Go), a new, state-of-the-art extension for training and deploying efficient deep learning object detection models on mobile devices and hardware."[49] "D2Go is already being used in Facebook's own development of computer vision models, specifically within FRL, where having hardware-aware, real-time models is essential for providing a great user experience – Facebook's 3D Photos feature being one such example."[50]

---

[46] Ex. 58, *Detectron2*, GITHUB (Nov. 15, 2021), https://github.com/facebookresearch/detectron2 (last visited May 24, 2022).

[47] Ex. 59, *Detectron2: A PyTorch-based modular object detection library*, META AI (Oct. 10, 2019), https://ai.facebook.com/blog/-detectron2-a-pytorch-based-modular-object-detection-library- (last visited May 24, 2022).

[48] *Id.*

[49] Ex. 48, *D2Go brings Detectron2 to mobile*, META AI (Mar. 4, 2021), https://ai.facebook.com/blog/d2go-brings-detectron2-to-mobile/ (last visited May 24, 2022).

[50] *Id.*

ALLEGATIONS OF PATENT INFRINGEMENT

COUNT I
INFRINGEMENT OF U.S. PATENT NO. 8,139,878

120. VideoLabs incorporates by reference the foregoing paragraphs of this First Amended Complaint as if fully set forth herein.

121. VL is the assignee and lawful owner of all right, title, and interest in and to the '878 Patent. The '878 Patent is valid and enforceable.

122. On information and belief, the Meta Companies have infringed and continue to infringe the '878 Patent in violation of 35 U.S.C. § 271(a), either literally or through the doctrine of equivalents, by making, using, selling, offering for sale, and/or importing into the United States products and/or methods that practice at least claim 1 of the '878 Patent, including with respect to the Meta Companies Coding Patents Accused Products.

123. On information and belief, the Meta Companies use the Meta Companies Coding Patents Accused Products for their own business purposes. In addition, the Meta Companies regularly conduct testing and troubleshooting of the Meta Companies Coding Patents Accused Products.[51] Further, VideoLabs believes companies related to the Meta Companies (e.g., other subsidiaries) use the '878 Meta Companies Accused Products.

124. On information and belief, the Meta Companies' infringement through their use of H.264 entropy encoding, described below, is exemplary of all of the Meta Companies' infringement with respect to all the Meta Companies Coding Patents Accused Products.

125. The Meta Companies Coding Patents Accused Products directly infringe at least

[51] *See, e.g.*, SFHTML5, *Facebook Messenger RTC – The Challenges and Opportunities of Scale*, YouTube (Oct. 27, 2017), https://www.youtube.com/watch?v=F7UWvflUZoc, at 10:24-12:06

claim 1 of the ’878 Patent, for example, by performing variable length encoding of blocks of picture data using the block data, inter-, intra-, and context-aware prediction to generate a predictive block, calculating a residual block using orthogonal transformation and quantization, and using the number of non-zero coefficients in the predicted block to encode the picture data at the encoder.

126. Each of the Meta Companies Coding Patents Accused Products meet every limitation of claim 1 of the ’878 Patent, which recites:

> 1. A transmitting apparatus which transmits multiplexed data which is obtained by multiplexing coded audio data and coded picture data, said transmitting apparatus comprising:
>
> an audio processing unit configured to code audio data to obtain coded audio data
>
> a picture coding unit configured to code picture data to obtain coded picture data; and
>
> a multiplexing unit configured to multiplex the coded audio data and the coded picture data to obtain multiplexed data,
>
> wherein said picture coding unit includes a block decoding unit configured to code a block image to obtain coded block data, the block image being obtained by dividing a picture signal into plural blocks, generating a residual block image from a block image of the respective blocks and a predictive block image obtained by intra-picture prediction or inter-picture prediction, and coding, on a block basis, coefficients obtained by performing orthogonal transformation and quantization on the residual block image,
>
> wherein said block coding unit includes:
>
> a coefficient number coding unit configured to code a total number of non-zero coefficients included in a current block to be coded, each of the non-zero coefficients being a coefficient having a value other than “0”;
>
> wherein said coefficient number coding unit includes:
>
> a determining unit configured to determine a predictive value for the number of non-zero coefficients included in the current block based on the number of non-zero coefficients included in a coded block located on a periphery of the current block;

> a selecting unit configured to select a variable length code table based on the determined predictive value; and
>
> a variable length coding unit configured to perform variable length coding on the total number of the non-zero coefficients included in the current block, by using the selected variable length code table.

127. Each of the Meta Companies Coding Patents Accused Products includes a transmitting apparatus that transmits multiplexed data, which is obtained by multiplexing coded audio data and coded picture data. H.264 carries audio and video multiplexed in a single stream. H.264 is directed to the picture portion of video, so devices containing H.264 encoders and decoders must multiplex/demultiplex audio and picture data in order to provide the H.264 picture data. Encoders multiplex the audio and pictures into a single stream, so that decoders receive the complete video presentation, including sound. Decoders receive the complete video presentation, including sound, and decode the stream to recreate the video. In the ISO Media File Format, which each accused Meta Companies Coding Patents product is capable of processing, a coded stream such as an H.264 video sequence or an audio stream is stored as a track, representing a sequence of coded data items or samples. Figure 8.32, below, illustrates an example of such multiplexed data, in which coded audio data ("audio track samples") and coded picture data ("video track samples") are multiplexed together.

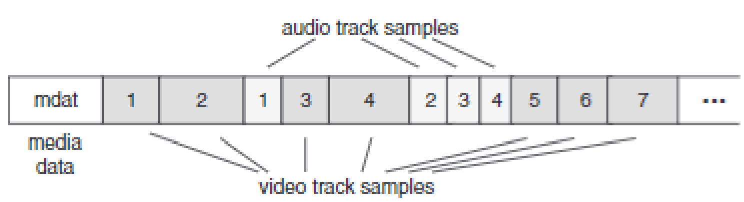


Figure 8.32 ISO Media File

Ex. 23, Richardson, at 247.

128. The Meta Companies Patents Accused Products have an audio processing unit configured to code audio data to obtain coded audio data. For example, the Meta Companies

Coding Patents Accused Products incorporate an audio codec that is configured to code audio data according to an input format, including without limitation, AAC. The audio data is encoded as coded audio data ("audio track samples").

129. The Meta Companies Coding Patents Accused Products further include a picture coding unit configured to code picture data to obtain coded picture data. For example, the Meta Companies Coding Patents Accused Products incorporate a H.264 video codec that is configured to code picture data to generate a H.264-compliant bitstream. The picture data is encoded as coded picture data ("video track samples").

130. The Meta Companies Coding Patents Accused Products include a multiplexing unit configured to multiplex the coded audio data and the coded picture data into multiplexed data. Encoders multiplex the audio track samples and video track samples so that decoders receive the complete video presentation, including sound. One representative example of multiplexed data generated by the Meta Companies Coding Patents Accused Products is shown in Figure 8.32 below, which shows the multiplexed stream of an ISO Media File including both coded audio track data and coded video track data.

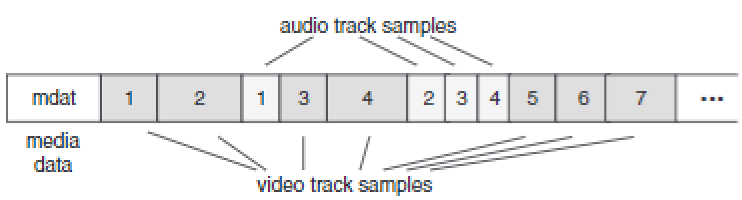


Figure 8.32 ISO Media File

Id. at 247.

131. The picture coding unit in the Meta Companies Coding Patents Accused Products includes a block coding unit configured to code a block image to obtain coded block data, the block image being obtained by dividing a picture signal into plural blocks.

132. In the Meta Companies Coding Patents Accused Products picture data is coded by an H.264-compliant encoder by dividing a picture signal into macroblocks, as shown in Figure 6-7:

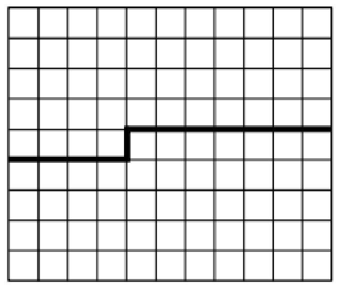


Figure 6-7 – A picture with 11 by 9 macroblocks that is partitioned into two slices

See ITU-T H.264, *Series H: Audiovisual and Multimedia Systems, Infrastructure of audiovisual services – Coding of moving video, Advanced video coding for generic audiovisual services*, Section 6.3, p. 25 (03/2009), at Figure 6-7.

133. The macroblock consists of a 16 x 16 block of luma samples and two corresponding blocks of chroma samples. A macroblock can be further partitioned for inter-prediction forming segmentations for motion representation as small as a block of 4 x 4 luma samples.[52]

134. In the Meta Companies Coding Patents Accused Products a residual block image is generated from the block image of the respective blocks and a predictive block image is obtained by intra-picture prediction or inter-picture prediction by an H.264-complaint encoder. For example, Figure 6.6 below shows a picture signal to be coded, with the macroblock being coded highlighted. The macroblock is predicted using neighboring, previously-encoded samples, as shown in Figure 6.7; because this prediction looks only to the other macroblocks of the same

[52] *See generally* Ex. 60, ITU-T H.264, *Series H: Audio Visual and Multimedia Systems, Infrastructure of Audiovisual Services – Coding of Moving Video, Advanced Video Coding for Generic Audio Visual Services* [hereinafter "H.264 Standard"], Section 0.6.3, p. 5 (09/2019).

picture, this is called intra-picture prediction. The predicted macroblock is shown in Figure 6.7. Figure 6.8 shows the prediction (Figure 6.7) subtracted from the original (Figure 6.6), which is called the residual.

Figure 6.6 QCIF frame with highlighted macroblock

Figure 6.7 Predicted luma frame formed using H.264 intra prediction

Figure 6.8 Residual after subtracting intra prediction

Ex. 23, Richardson at 141-143.

135. The Meta Companies Coding Patents Accused Products can use intra- and inter-picture coding on the macroblocks of a picture signal. Inter-picture coding predicts the value of the macroblock using temporal statistical dependencies between different pictures. Both types of prediction can be used by the H.264-compliant encoder to calculate the residual.

136. In the Meta Companies Coding Patents Accused Products the H.264-compliant encoder codes, on a block basis, coefficients obtained by performing orthogonal transformation and quantization on the residual block image. H.264 specifies an *entropy_coding_mode* flag that dictates the entropy encoding algorithm used to encode the picture data. When this flag is set to "0" the residual block data is coded using a CAVLC scheme.[53] In such case, the resulting prediction residual is split into 4x4 blocks, and transformation and quantization are applied. An integer transform is applied to the residual, outputting a set of coefficient weighting values. This process is shown in the figures below:

[53] *See* Ex. 60, H.264 Standard, Section 7.4.2.2, pp. 81-82 (09/2019).

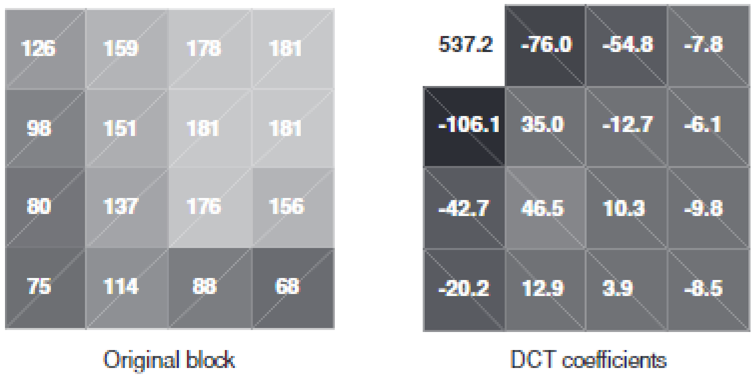


Figure 3.31 Close-up of 4 × 4 block; DCT coefficients

Ex. 23, Richardson, at 47.

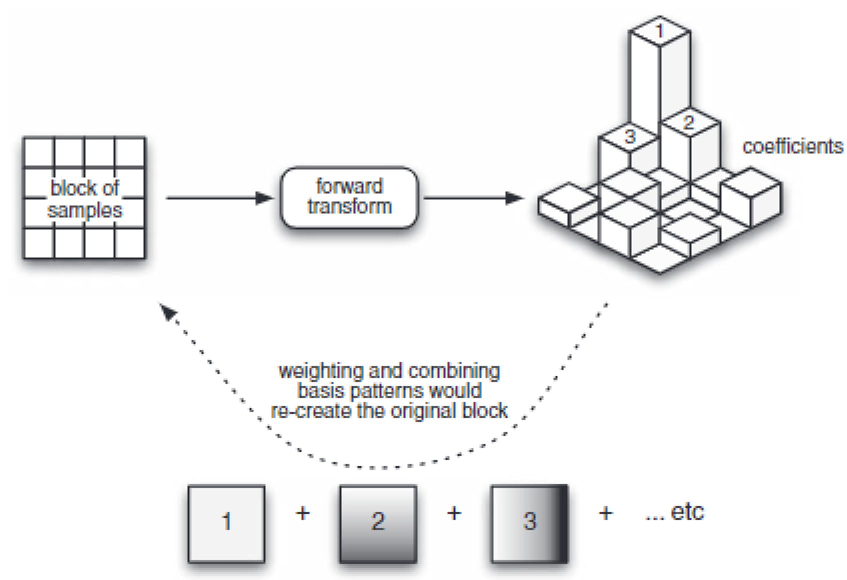


Figure 4.11 Forward transform

Ex. 23, Richardson, at 88.

137. The coefficients are then quantized, meaning that insignificant coefficient values are rounded down (for example, to zero), while a small number of significant, non-zero coefficients are retained. The quantization step is shown in Figure 4.12:

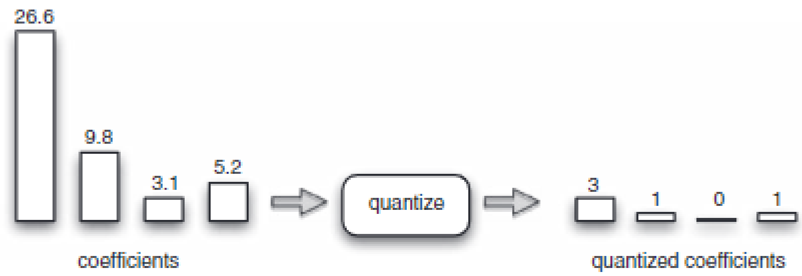


Figure 4.12 Quantization example

Ex. 23, Richardson, at 88.

138. The encoding process is shown in Figure 4.4:

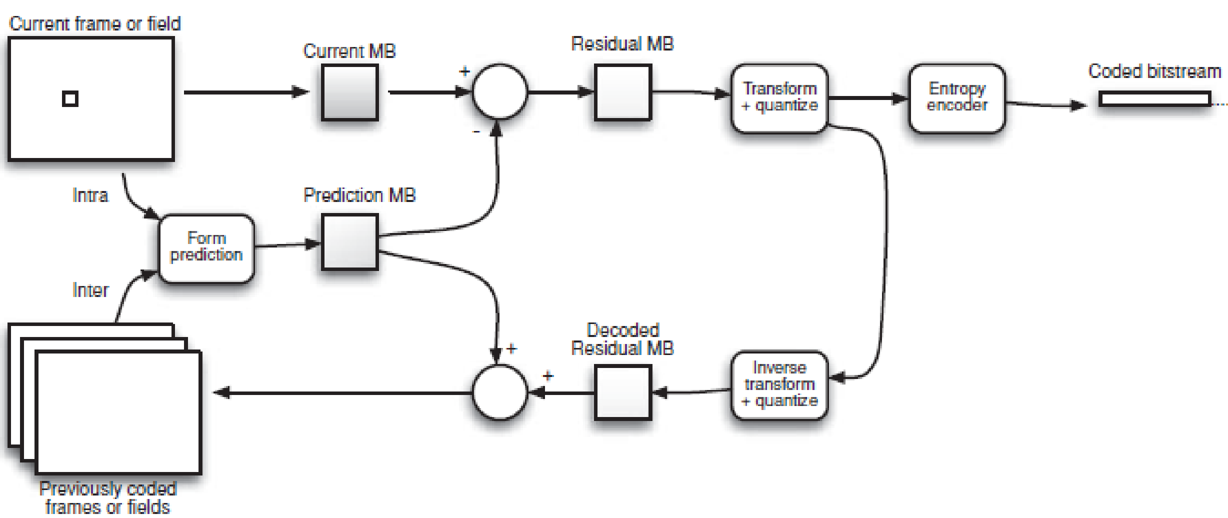


Figure 4.4 Typical H.264 encoder

Ex. 23, Richardson, at 84.

139. In the Meta Companies Coding Patents Accused Products the block coding unit includes a coefficient number coding unit configured to code a total number of non-zero coefficients included in a current block to be coded, where each of the non-zero coefficients having a value other than "0". In H.264 CAVLC encoding generally, which the Meta Companies Coding Patents Accused Products support, the total number of non-zero coefficients included in a current block to be coded is derived in order to generate a H.264-compliant bitstream.

140. To this end, the coefficient number coding unit in the Meta Companies Coding

Patents Accused Products includes a determining unit configured to determine a predictive value for the number of non-zero coefficients included in the current block based on the number of non-zero coefficients included in a block located on a periphery of the current block. Since CAVLC is a context-adaptive variable length coding technique, the number of non-zero coefficients in neighboring blocks is correlated as part of the entropy coding process. An H.264-compliant encoder uses previously-processed macroblocks to help encode the currently-processed macroblock. The number of non-zero coefficients in previously-processed blocks on the periphery—including the blocks to the left and above the current macroblock—are used to predict the number of non-zero coefficients in the current block. This use of the blocks of the periphery is shown in Figure 6-14:

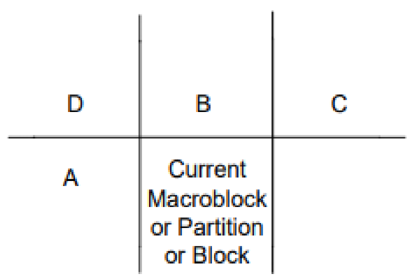


Figure 6-14 – Determination of the neighbouring macroblock, blocks, and partitions (informative)

Ex. 60, H.264 Standard, at 33.

141. The H.264-compliant encoder in the Meta Companies Coding Patents Accused Products obtains the number of non-zero coefficients in the left and above blocks to set the variable *nC*, the prediction of the current macroblock's number of non-zero coefficients based on the neighboring macroblocks' number of non-zero coefficients.[54]

142. The coefficient number coding unit in the Meta Companies Coding Patents Accused Products also includes a selecting unit configured to select a variable length code table

[54] *See* Ex. 60, H.264 Standard, Section 9.2.1, pp. 214-216 (09/2019).

based on the determined predictive value. The H.264-compliant encoder in the Meta Companies Coding Patents Accused Products uses the predictive value *nC* to select one of six variable length coding tables specified in Table 9-5 of the H.264 standard in order to generate a H.264-compliant bitstream.[55]

143. This selection of a variable length code table based on a determined predictive value is exemplified in Figure 7.19:

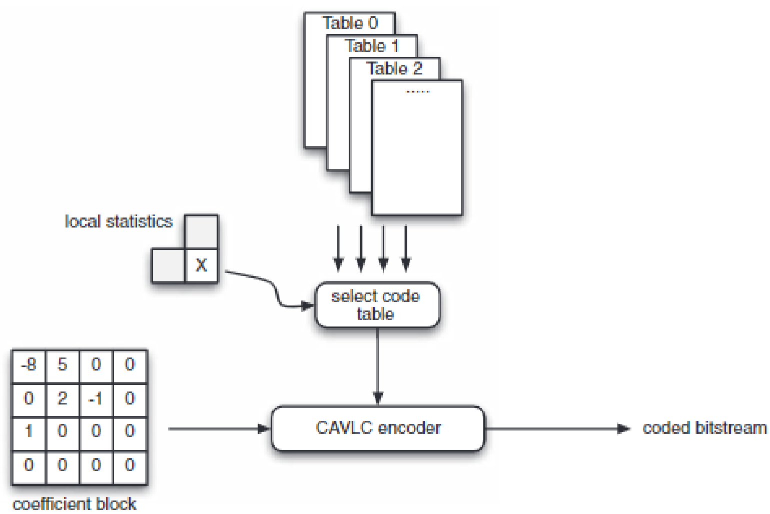


Figure 7.19 CAVLC encoder overview

Ex. 23, Richardson at 211.

144. The coefficient number coding unit in the Meta Companies Coding Patents Accused Products further includes a variable length coding unit configured to perform variable length coding on the number of the non-zero coefficients included in the current block, by using the selected variable length code table. Figure 7.19, *supra*, shows a **C**ontext-**A**daptive **V**ariable-**L**ength **C**oding ("CAVLC") coding unit. The unit uses the selected variable length code table to perform variable length coding on the syntax element *coeff_token* representing the number of non-

[55] *See* Ex. 60, H.264 Standard, Section 9.2.1 and Table 9-5, pp. 214-218 (09/2019).

zero coefficients of the current macroblock to generate a H.264-compliant bitstream.[56]

145. VideoLabs representatives reached out to Meta representatives at least as early as October 2, 2019 regarding VideoLabs' platform and to gauge the Meta Companies' interest in joining as a partner or member. In October, 2019, VideoLabs representatives presented VideoLabs' platform to representatives of the Meta Companies. VideoLabs representatives and Meta representatives spoke between December and February, and met in person on January 30, 2020, during which VideoLabs presented them with an updated proposal. On June 3, 2020, VideoLabs representatives reached out again and stated that "With regards to Facebook, and based on our analysis so far, we have determined that VideoLabs' current patent portfolio is relevant to the majority of Facebook's annual revenue in some way and we have 17 unique claim charts completed or in development (and other evidence of use) so far related to Facebook Messenger, Facebook Live, Oculus, Instagram, Facebook Workplace, Portal, Facebook Ads, and backend infrastructure" and asked them to engage in "good faith licensing discussions" under a proposed NDA. VideoLabs representatives sent another updated proposal on March 23, 2020. VideoLabs representatives reached out again on June 26, 2020, March 16, 2021, and May 4, 2021 to "request[] good faith licensing negotiations" regarding the number of patents relevant to Facebook and other Meta services. On June 26, 2020, VideoLabs confirmed in an email that representatives of the Meta Companies said on a recent call that the Meta Companies were not interested in good faith licensing discussions. To date, months later, the Meta Companies have not reengaged with VideoLabs.

---

[56] *See generally* Ex. 60, H.264 Standard, Section 9.2.1 and Table 9-5, pp. 214-218 (09/2019).

146. The Meta Companies of course know how their products operate, and on information and belief, they investigated the ’878 Patent and the infringement of the Meta Companies Coding Patents Accused Products. The Meta Companies have been given further notice of the ’878 Patent and their infringement of the ’878 Patent through VideoLabs’ May 23, 2022 letter, the filing of the VideoLabs’ May 24, 2022 Complaint, and this First Amended Complaint. On information and belief, the Meta Companies are either knowingly infringing the ’878 Patent or are willfully blind to their infringement, and continue to act in wanton disregard of VideoLabs’ patent rights.

147. Despite becoming aware of or willfully blinding themselves to their infringement of the ’878 Patent, the Meta Companies have nonetheless continued to engage in and have escalated their infringing activities by continuing to develop, advertise, make available, and use the infringing functionalities of the Meta Companies Coding Patents Accused Products. On information and belief, the Meta Companies have made no attempts to design around the ’878 Patent or otherwise stop their infringing behavior.

148. The Meta Companies’ infringement of the ’878 Patent therefore has been and remains willful.

149. The Meta Companies also indirectly infringe the ’878 Patent by inducing others to infringe and contributing to the infringement of others, including other Defendants, third party users of the Meta Companies Patents Accused Products in this District and throughout the United States. As described above, on information and belief, the Meta Companies have known about the family of patents including the ’878 Patent since at least January 30, 2020, and have known about the ’878 Patent since May 23, 2021.

150. On information and belief, the Meta Companies have actively induced the

infringement of the ’878 Patent under 35 U.S.C. § 271(b) by actively inducing the infringement of the Meta Companies Coding Patents Accused Products by third parties in the United States. The Meta Companies knew or were willfully blind to the fact that their conduct would induce these third parties to act in a manner that infringes the ’878 Patent in violation of 35 U.S.C. § 271(a).

151. The Meta Companies actively encouraged and continue to actively encourage third parties to directly infringe the ’878 Patent by, for example, marketing the Coding Patents Accused Products and infringing functionalities to consumers; working with consumers to implement, install and/or operate the Coding Patents Accused Products and infringing functionalities; fully supporting and managing consumers’ continuing use of the Coding Patents Accused Products and infringing functionalities; and providing technical assistance to consumers during their continued use of the Coding Patents Accused Products and infringing functionalities.[57]

---

[57] *See e.g.,* Ex. 71, *Using Facebook-Your Photos and Videos-Videos*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/1738143323068602 (last visited May 24, 2022); Ex. 72, *I can’t add a video on Facebook*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/215726848451641 (last visited May 24, 2022); Ex. 73, *Audio and Video Calling*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/messenger-app/1673374996287506 (last visited May 24, 2022); Ex. 74, *How do I video chat with someone or a group in Messenger?*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/messenger-app/1414800065460231 (last visited May 24, 2022); Ex. 75, *Instagram Features-Videos*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/381435875695118 (last visited May 24, 2022); Ex. 76, *How do I share a video on Instagram?*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/456185931138729 (last visited May 24, 2022); Ex. 77, *How do I share my video on Instagram to my Facebook Page?*, INSTAGRAM HELP CENTER (2022), https://www.facebook.com/help/instagram/486878428409681 (last visited May 24, 2022); Ex. 78, *Oculus Support*, https://www.youtube.com/channel/UCxLTeQQFvA5_eb0VkrLIq6w (last visited May 24, 2022); Ex. 79, *Viewing your own 360 videos on your Gear VR*; META, https://support.giphy.com/hc/en-us/articles/360019674452-How-To-Make-A-GIF (last visited May 24, 2022); Ex. 80, *How to make a video call*, WHATSAPP, https://faq.whatsapp.com/android/voice-and-video-calls/how-to-make-a-video-call/?lang=en (last visited May 24, 2022); Ex. 81, *I get a message that my video is too long and it won’t send*, META, https://faq.whatsapp.com/general/i-get-a-message-that-my-video-is-too-long-and-it-wont-send/?lang=en (last visited May 24, 2022).

152. On information and belief, the Meta Companies contributorily infringe the ’878 Patent under 35 U.S.C. § 271(c) by importing, selling, and/or offering to sell within the United States the Meta Companies Coding Patents Accused Products (or components thereof) that constitute a material part of the claimed invention and are not staple articles of commerce suitable for substantial non-infringing use. For example, the hardware and/or software for encoding content with H.264 using CAVLC is material, has no insubstantial non-infringing uses, and is known by the Meta Companies to be especially made or adapted for use in a manner that infringes the ’878 Patent.

COUNT II
INFRINGEMENT OF U.S. PATENT NO. 7,769,238

153. VideoLabs incorporates by reference the foregoing paragraphs of this First Amended Complaint as if fully set forth herein.

154. VL is the assignee and lawful owner of all right, title, and interest in and to the ’238 Patent. The ’238 Patent is valid and enforceable.

155. On information and belief, the Meta Companies have infringed and continue to infringe the ’238 Patent in violation of 35 U.S.C. § 271(a), either literally or through the doctrine of equivalents, by making, using, selling, offering for sale, and/or importing into the United States products and/or methods that practice claim 1 of the ’238 Patent, including with respect to the Meta Companies Coding Patents Accused Products.

156. On information and belief, the Meta Companies use the Meta Companies Coding Patents Accused Products for their own business purposes. In addition, the Meta Companies regularly conduct testing and troubleshooting of the Meta Companies Coding Patents Accused

Products.[58] Further, VideoLabs believes companies related to the Meta Companies (e.g., other subsidiaries) use the '238 Meta Companies Accused Products.

157. On information and belief, the Meta Companies' infringement through its use of H.264 entropy decoding, described below, is exemplary of all of their infringement with respect to all the Meta Companies Coding Patents Accused Products.

158. The Meta Companies Coding Patents Accused Products directly infringe claim 1 of the '238 Patent by performing variable length decoding of coded blocks of picture data using the coded block data, inter-, intra-, and context-aware prediction to generate a predictive block, calculating a residual block using orthogonal transformation and quantization, and using the number of non-zero coefficients in the predicted block to reconstruct the picture data at the decoder.

159. Each of the Meta Companies Coding Patents Accused Products meet every limitation of claim 1 of the '238 Patent, which recites:

> 1. A receiving apparatus which receives multiplexed data which is obtained by multiplexing coded audio data and coded picture data, said receiving apparatus comprising:
>
> a demultiplexing unit configured to separate the multiplexed data into the coded audio data and the coded picture data;
>
> an audio processing unit configured to decode the separated coded audio data; and
>
> a picture decoding unit configured to decode the separated coded picture data,
>
> wherein said picture decoding unit includes a block decoding unit configured to decode coded block data included in the coded picture data, the coded block data being obtained by dividing a picture signal into plural blocks, generating a residual block image from a block image of the respective blocks and a predictive block image

[58] *See, e.g.*, SFHTML5, *Facebook Messenger RTC – The Challenges and Opportunities of Scale*, YouTube (Oct. 27, 2017), https://www.youtube.com/watch?v=F7UWvflUZoc, at 10:24-12:06

> obtained by intra-picture prediction or inter-picture prediction, and coding, on a block basis, coefficients obtained by performing orthogonal transformation and quantization on the residual block image,
>
> said block decoding unit includes:
>
> a coefficient number decoding unit configured to decode the coded block data to obtain the number of non-zero coefficients which are coefficients included in a current block to be decoded and having a value other than "0";
>
> a unit configured to obtain coefficients corresponding to a residual block image of the current block by decoding the coded block data;
>
> a unit configured to obtain the residual block image of the current block by performing inverse quantization and inverse orthogonal transformation on the coefficients corresponding to the residual block image of the current block; and
>
> a reproducing unit configured to reproduce a block image of the current block, from the obtained residual block image and a predictive block image obtained by intra-picture prediction or inter-picture prediction,
>
> said coefficient number decoding unit includes:
>
> a determining unit configured to determine a predictive value for the number of non-zero coefficients included in the current block based on the number of non-zero coefficients included in a decoded block located on a periphery of the current block;
>
> a selecting unit configured to select a variable length code table based on the determined predictive value; and
>
> a variable length decoding unit configured to perform variable length decoding on a coded stream which is generated by coding the number of the non-zero coefficients included in the current block, by using the selected variable length code table.

160. Each of the Meta Companies Coding Patents Accused Products includes a receiving apparatus that receives multiplexed data, which is obtained by multiplexing coded audio data and coded picture data. H.264 carries audio and video multiplexed in a single stream. H.264 is directed to the picture portion of video, so devices containing H.264 encoders and decoders must multiplex/demultiplex audio and picture data in order to obtain the H.264 picture data. Encoders multiplex the audio and pictures into a single stream, so that decoders receive the complete video

presentation, including sound. Decoders receive the complete video presentation, including sound, and decode the stream to recreate the video. In the ISO Media File Format, which each accused H.264 product is capable of processing, a coded stream such as an H.264 video sequence or an audio stream is stored as a track, representing a sequence of coded data items or samples. Figure 8.32, below, illustrates an example of such multiplexed data, in which coded audio data ("audio track samples") and coded picture data ("video track samples") are multiplexed together.

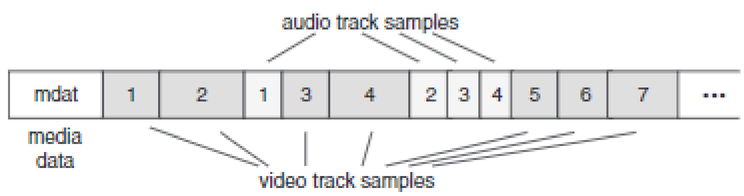


Figure 8.32 ISO Media File

Ex. 23, Richardson, at 247.

161. The Meta Companies Coding Patents Accused Products include a demultiplexing unit configured to separate the multiplexed data into the coded audio data and the coded picture data. Since the coded audio data and the coded picture data are decoded by separate respective decoders, it is necessary to first demultiplex the audio track samples and video track samples in order to reproduce the encoded data in its entirety. Figure 8.32 below shows the multiplexed stream of an ISO Media File as received at the demultiplexing unit, which includes coded audio data ("audio track samples") and coded picture data ("video track samples"). The coded audio data and the coded picture data are demultiplexed and sent to respective decoders for further processing.

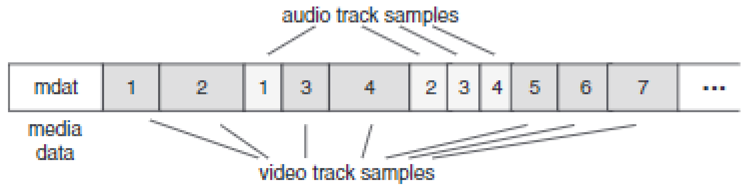


Figure 8.32 ISO Media File

Ex. 23, Richardson, at 247.

162. The Meta Companies Coding Patents Accused Products have an audio processing unit configured to decode the separated coded audio data. For example, the Meta Companies Coding Patents Accused Products incorporate an audio codec that is configured to decode audio data according to an input format, including without limitation, AAC.

163. The Meta Companies Coding Patents Accused Products further include a picture decoding unit configured to decode the separated coded picture data. For example, the Meta Companies Coding Patents Accused Products incorporate a H.264 video codec that is configured to decode a H.264 encoded bitstream.

164. The picture decoding unit in the Meta Companies Coding Patents Accused Products includes a block decoding unit configured to decode coded block data included in the coded picture data, the coded block data being obtained by dividing a picture signal into plural blocks.

165. The Meta Companies Coding Patents Accused Products decode picture data that has been coded by an H.264-compliant encoder which divides a picture signal into macroblocks, generates a residual block image from a block image of the respective blocks and a predictive block image obtained by intra-picture prediction or inter-picture prediction, and codes, on a block basis, coefficients obtained by performing orthogonal transformation and quantization on the residual block image to generate a H.264-compliant bitstream.

166. The process of dividing the picture signal into macroblocks is shown in Figure 6-7:

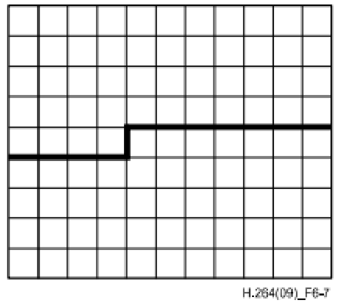


Figure 6-7 – A picture with 11 by 9 macroblocks that is partitioned into two slices

Id. at Figure 6-7.

167. The macroblock consists of a 16 x 16 block of luma samples and two corresponding blocks of chroma samples. A macroblock can be further partitioned for inter-prediction forming segmentations for motion representation as small as a block of 4 x 4 luma samples.[59]

168. A residual block image is generated from the block image of the respective blocks and a predictive block image is obtained by intra-picture prediction or inter-picture prediction by an H.264-compliant encoder. For example, Figure 6.6 below shows a picture signal to be coded, with the macroblock being coded highlighted. The macroblock is predicted using neighboring, previously-encoded samples, as shown in Figure 6.7; because this prediction looks only to the other macroblocks of the same picture, this is called intra-picture prediction. The predicted macroblock is shown in Figure 6.7. Figure 6.8 shows the prediction (Figure 6.7) subtracted from the original (Figure 6.6), which is called the residual.

[59] *See* Ex. 60, H.264 Standard, Section 0.6.3, p. 5 (09/2019).

Figure 6.6 QCIF frame with highlighted macroblock

Figure 6.7 Predicted luma frame formed using H.264 intra prediction

Figure 6.8 Residual after subtracting intra prediction

Ex. 23, Richardson, at 141-143.

169. Intra- and inter- picture prediction can be used by the H.264-compliant encoder to calculate the residual. Inter-picture coding predicts the value of the macroblock using temporal statistical dependencies between different pictures.

170. An H.264-compliant encoder codes, on a block basis, coefficients obtained by performing orthogonal transformation and quantization on the residual block image. H.264 specifies an *entropy_coding_mode* flag that dictates the entropy encoding algorithm used to encode the picture data. When this flag is set to "0" the residual block data is coded using a CAVLC scheme.[60] In such case, the resulting prediction residual is split into 4x4 blocks, and transformation and quantization are applied. An integer transform is applied to the residual, outputting a set of coefficient weighting values. This process is shown in the figures below:

[60] *See* Ex. 60, H.264 Standard, Section 7.4.2.2, pp. 81-82 (09/2019).

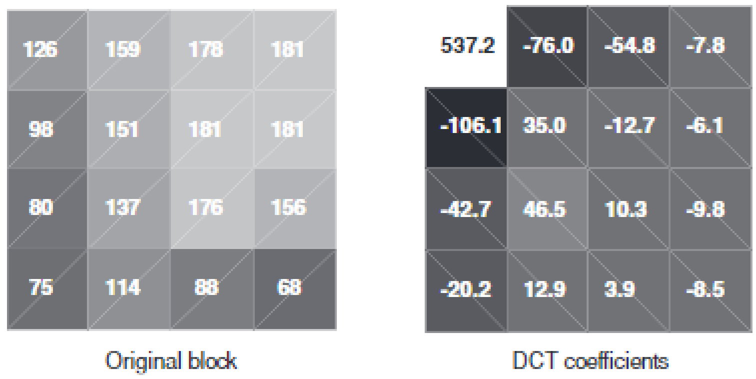


Figure 3.31 Close-up of 4 × 4 block; DCT coefficients

Ex. 23, Richardson, at 47.

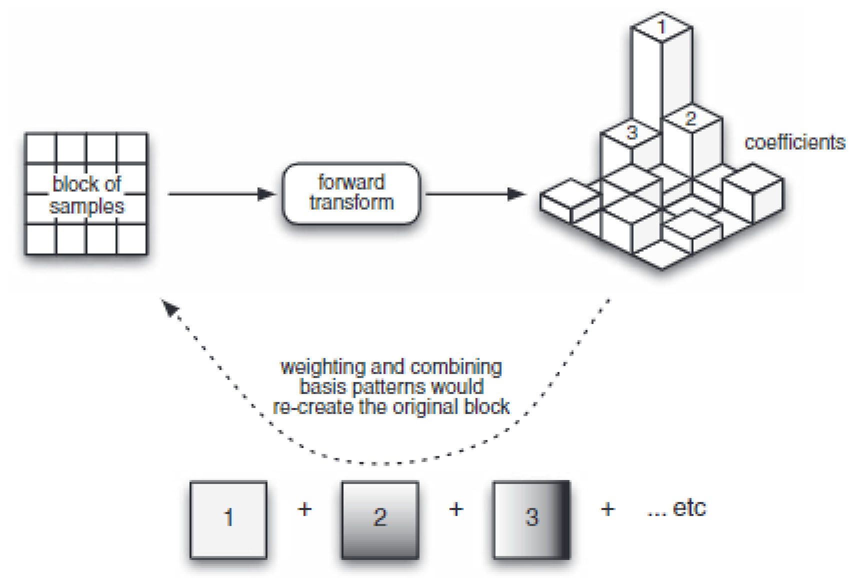


Figure 4.11 Forward transform

Ex. 23, Richardson at 88.

171. The coefficients are then quantized, meaning that insignificant coefficient values are rounded down (for example, to zero), while a small number of significant, non-zero coefficients are retained. The quantization step is shown in Figure 4.12:

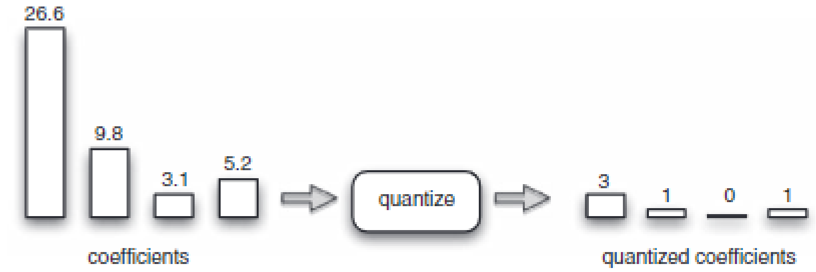


Figure 4.12 Quantization example

Ex. 23, Richardson, at 88.

172. These quantized coefficients are rescaled to obtain similar coefficients at the decoder, as shown in Figure 4.14:

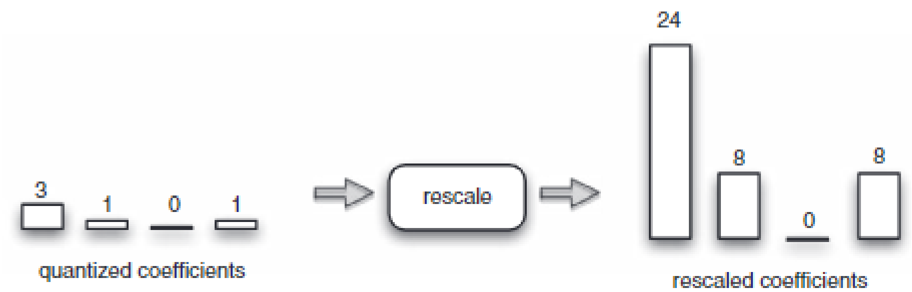


Figure 4.14 Rescaling example

Ex. 23, Richardson at 89.

173. The encoding process is shown in Figure 4.4:

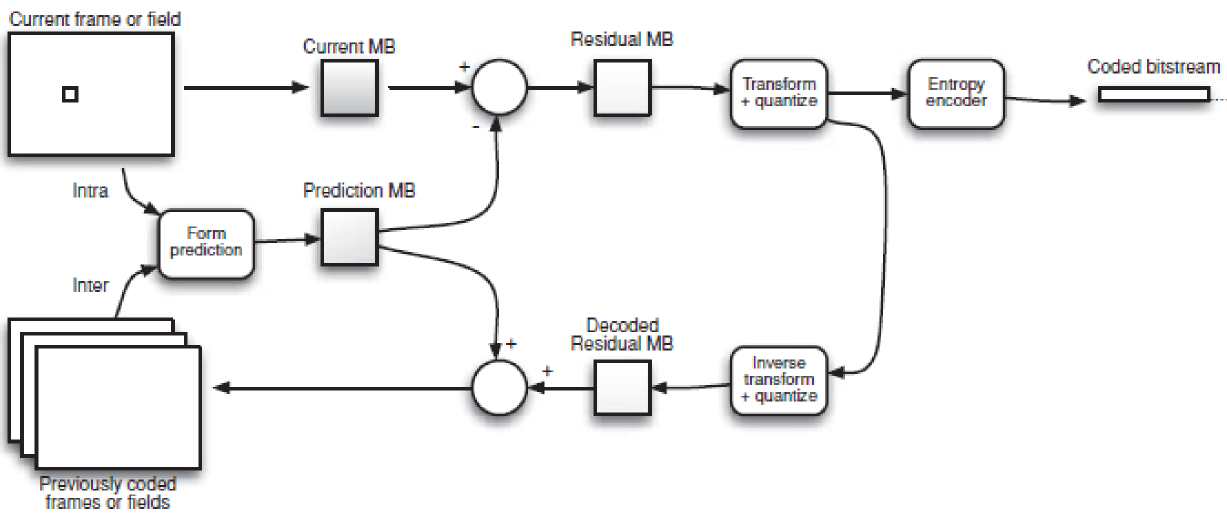


Figure 4.4 Typical H.264 encoder

Ex. 23, Richardson, at 84.

174. In the Meta Companies Coding Patents Accused Products the block decoding unit includes a coefficient number decoding unit configured to decode the coded block data to obtain the number of non-zero coefficients which are coefficients included in a current block to be decoded and have a value other than "0." In H.264, the total number of non-zero coefficients in the current block is encoded by the variable *coeff_token*. The coefficient number decoding unit in the Meta Companies Coding Patents Accused Products is configured to decode the coded block data by executing the function TotalCoeff(*coeff_token*) to return the number of non-zero transform coefficient levels derived from *coeff_token*.[61] A "level" in this context is the value of a transform coefficient prior to scaling.[62]

175. The block decoding unit in the Meta Companies Coding Patents Accused Products

[61] *See* Ex. 60, H.264 Standard, Section 7.4.5.3.2, p. 110 (09/2019).

[62] *See* Ex. 60, H.264 Standard, Section 3.175 (transform coefficient level definition), p. 15 (09/2019).

also includes a unit configured to obtain coefficients corresponding to a residual block image of the current block by decoding the coded block data. H.264-compliant decoders read the encoded residual macroblock data and extract the coefficients located in that data. This extraction is called parsing. The parsed data is then inverse quantized and inverse orthogonal transforms are applied to reconstruct the residual macroblock. The H.264 decoding process in this instance employs a CAVLC parsing process to obtain a list of transform coefficient levels (coeffLevel) of the luma block or the chroma block.[63]

176. The block decoding unit in the Meta Companies Coding Patents Accused Products further includes a unit configured to obtain the residual block image of the current block by performing inverse quantization and inverse orthogonal transformation on the coefficients corresponding to the residual block image of the current block. Once the residual luma and chroma coefficient blocks are extracted, the H.264-compliant decoder calculates a quantization parameter, and carries out inverse quantization and inverse transformation to (re)produce the residual sample blocks.[64]

177. The block decoding unit in the Meta Companies Coding Patents Accused Products also includes a reproducing unit configured to reproduce a block image of the current block, from the obtained residual block image and a predictive block image obtained by intra-picture prediction or inter-picture prediction. For each macroblock, the H.264 decoder forms an identical prediction to the one created by the encoder using inter-picture prediction from previously-decoded frames or intra-picture prediction from previously-decoded samples in the current frame. The decoder adds the prediction to the decoded residual to reconstruct a decoded macroblock which can then

---

[63] *See* Ex. 60, H.264 Standard, Section 9.2-9.2.4, pp. 214-223 (09/2019).

[64] *See* Ex. 23, Richardson, at 96.

be displayed as part of a video frame. This reconstruction of a macroblock at the decoder can be seen in Figure 4.17:

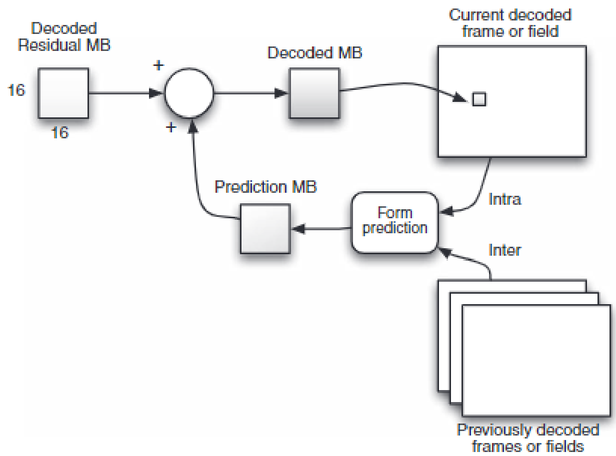


Figure 4.17 Reconstruction flow diagram

Ex. 23, Richardson at 91.

178. The decoding process is shown in Figure 4.21:

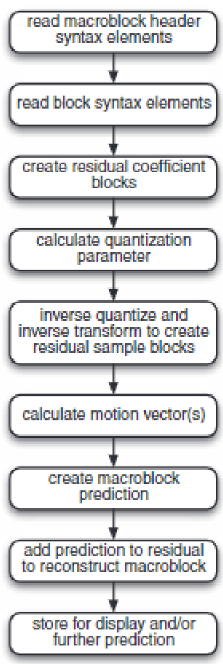


Figure 4.21 P-macroblock decoding process

Ex. 23, Richardson, at 96.

179. The coefficient number decoding unit in the Meta Companies Coding Patents Accused Products includes a determining unit configured to determine a predictive value for the number of non-zero coefficients included in the current block based on the number of non-zero coefficients included in a decoded block located on a periphery of the current block. Since CAVLC is a context-adaptive variable length coding technique, the number of non-zero coefficients in neighboring blocks is correlated as part of the entropy decoding process. An H.264-compliant decoder uses previously-processed macroblocks to help decode the currently-processed macroblock. The number of non-zero coefficients in previously-processed blocks on the periphery — including the blocks to the left and above the current macroblock — are used to predict the number of non-zero coefficients in the current block. This use of the blocks of the periphery is shown in Figure 6-14:

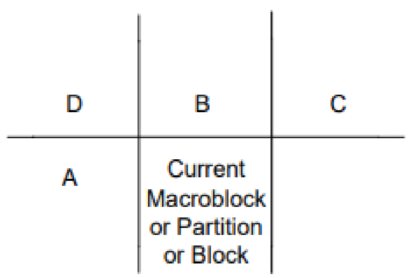


Figure 6-14 – Determination of the neighbouring macroblock, blocks, and partitions (informative)

Ex. 60, H.264 Standard, at 33.

180. The H.264-compliant decoder in the Meta Companies Coding Patents Accused Products obtains the number of non-zero coefficients in the left and above blocks to set the variable *nC*, the prediction of the current macroblock's number of non-zero coefficients based on the neighboring macroblocks' number of non-zero coefficients.[65]

181. The coefficient number decoding unit in the Meta Companies Coding Patents

[65] *See* Ex. 60, H.264 Standard, Section 9.2.1, pp. 214-216 (09/2019).

Accused Products also includes a selecting unit configured to select a variable length code table based on the determined predictive value. The H.264-compliant decoder in the Meta Companies Coding Patents Accused Products uses the predictive value *nC* to select one of six variable length coding tables specified in Table 9-5 of the H.264 standard in order to decode the H.264-compliant bitstream.[66]

182. The coefficient number decoding unit in the Meta Companies Coding Patents Accused Products further includes a variable length decoding unit configured to perform variable length decoding on a coded stream which is generated by coding the number of the non-zero coefficients included in the current block, by using the selected variable length code table. At the H.264-compliant CAVLC decoder, the selected variable length code table will be used to decode the syntax element *coeff_token*, in the H.264-compliant bitstream, which represents the number of non-zero coefficients in the macroblock.[67]

183. VideoLabs representatives reached out to Meta representatives at least as early as October 2, 2019 regarding VideoLabs' platform and to gauge the Meta Companies' interest in joining as a partner or member. In October, 2019, VideoLabs representatives presented VideoLabs' platform to representatives of the Meta Companies. VideoLabs representatives and Meta representatives spoke between December and February, and met in person on January 30, 2020, during which VideoLabs presented them with an updated proposal. On June 3, 2020, VideoLabs representatives reached out again and stated that "With regards to Facebook, and based on our analysis so far, we have determined that VideoLabs' current patent portfolio is relevant to the majority of Facebook's annual revenue in some way and we have 17 unique claim charts

[66] *See* Ex. 60, H.264 Standard, Section 9.2.1 and Table 9-5, pp. 214-218 (09/2019).

[67] *See generally* Ex. 60, H.264 Standard, Section 9.2.1 and Table 9-5, pp. 214-218 (09/2019).

completed or in development (and other evidence of use) so far related to Facebook Messenger, Facebook Live, Oculus, Instagram, Facebook Workplace, Portal, Facebook Ads, and backend infrastructure" and asked them to engage in "good faith licensing discussions" under a proposed NDA. VideoLabs representatives sent another updated proposal on March 23, 2020. VideoLabs representatives reached out again on June 26, 2020, March 16, 2021, and May 4, 2021 to "request[] good faith licensing negotiations" regarding the number of patents relevant to Facebook and other Meta services. On June 26, 2020, VideoLabs confirmed in an email that representatives of the Meta Companies said on a recent call that the Meta Companies were not interested in good faith licensing discussions. To date, months later, the Meta Companies have not reengaged with VideoLabs.

184. The Meta Companies of course know how their products operate, and on information and belief, the Meta Companies investigated the '238 Patent and their infringement of the Meta Companies Coding Patents Accused Products. The Meta Companies have been given further notice of the '238 Patent and their infringement of the '238 Patent through VideoLabs' May 23, 2022 letter, VideoLabs' May 24, 2022 Complaint, and the filing of this First Amended Complaint. On information and belief, the Meta Companies are either knowingly infringing the '238 Patent or are willfully blind to its infringement, and continue to act in wanton disregard of VideoLabs' patent rights.

185. Despite becoming aware of or willfully blinding itself to its infringement of the '238 Patent, the Meta Companies have nonetheless continued to engage in and has escalated their infringing activities by continuing to develop, advertise, make available, and use the infringing functionalities of the Meta Companies Coding Patents Accused Products. On information and

belief, the Meta Companies have made no attempts to design around the ’238 Patent or otherwise stop their infringing behavior.

186. The Meta Companies’ infringement of the ’238 Patent therefore has been and remains willful.

187. The Meta Companies also indirectly infringe the ’238 Patent by inducing others to infringe and contributing to the infringement of others, including other Defendants and third-party users of the Meta Companies Coding Patents Accused Products in this District and throughout the United States. As described above, on information and belief, the Meta Companies have known about the family of patents including the ’238 Patent since at least January 30, 2020, and have known about the ’238 Patent since May 23, 2021.

188. On information and belief, the Meta Companies have actively induced the infringement of the ’238 Patent under 35 U.S.C. § 271(b) by actively inducing the infringement of the Meta Companies Coding Patents Accused Products by third parties in the United States. The Meta Companies knew or were willfully blind to the fact that their conduct would induce these third parties to act in a manner that infringes the ’238 Patent in violation of 35 U.S.C. § 271(a).

189. The Meta Companies actively encouraged and continue to actively encourage third parties to directly infringe the ’238 Patent by, for example, marketing the Coding Patents Accused Products and infringing functionalities to consumers; working with consumers to implement, install and/or operate the Coding Patents Accused Products and infringing functionalities; fully supporting and managing consumers’ continuing use of the Coding Patents Accused Products and

infringing functionalities; and providing technical assistance to consumers during their continued use of the Coding Patents Accused Products and infringing functionalities.[68]

190. On information and belief, the Meta Companies contributorily infringe the '238 Patent under 35 U.S.C. § 271(c) by importing, selling, and/or offering to sell within the United States the Meta Companies Coding Patents Accused Products (or components thereof) that constitute a material part of the claimed invention and are not staple articles of commerce suitable for substantial non-infringing use. For example, the hardware and/or software for decoding content with H.264 using CAVLC is material, has no insubstantial non-infringing uses, and is known by the Meta Companies to be especially made or adapted for use in a manner that infringes the '238 Patent.

COUNT III

[68] *See e.g.,* Ex. 71, *Using Facebook-Your Photos and Videos-Videos*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/1738143323068602 (last visited May 24, 2022); Ex. 72, *I can't add a video on Facebook*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/215726848451641 (last visited May 24, 2022); Ex. 73, *Audio and Video Calling*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/messenger-app/1673374996287506 (last visited May 24, 2022); Ex. 74, *How do I video chat with someone or a group in Messenger?*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/messenger-app/1414800065460231 (last visited May 24, 2022); Ex. 75, *Instagram Features-Videos*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/381435875695118 (last visited May 24, 2022); Ex. 76, *How do I share a video on Instagram?*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/456185931138729 (last visited May 24, 2022); Ex. 77, *How do I share my video on Instagram to my Facebook Page?*, INSTAGRAM HELP CENTER (2022), https://www.facebook.com/help/instagram/486878428409681 (last visited May 24, 2022); Ex. 78, *Oculus Support*, https://www.youtube.com/channel/UCxLTeQQFvA5_eb0VkrLIq6w (last visited May 24, 2022); Ex. 79, *How To Make A GIF*; GIPHY SUPPORT, https://support.giphy.com/hc/en-us/articles/360019674452-How-To-Make-A-GIF (last visited May 24, 2022); Ex. 80, *How to make a video call*, WHATSAPP, https://faq.whatsapp.com/android/voice-and-video-calls/how-to-make-a-video-call/?lang=en (last visited May 24, 2022); Ex. 81, *I get a message that my video is too long and it won't send*, META, https://faq.whatsapp.com/general/i-get-a-message-that-my-video-is-too-long-and-it-wont-send/?lang=en (last visited May 24, 2022); Ex. 85, *Viewing your own 360 videos on your Gear VR*, META, https://store.facebook.com/help/quest/articles/headsets-and-accessories/oculus-go-and-gear-vr/gear-vr-personal-360-videos (last visited May 24, 2022).

INFRINGEMENT OF U.S. PATENT NO. 7,970,059

191. VideoLabs incorporates by reference the foregoing paragraphs of this First Amended Complaint as if fully set forth herein.

192. VL is the assignee and lawful owner of all right, title, and interest in and to the ’059 Patent. The ’059 Patent is valid and enforceable.

193. On information and belief, the Meta Companies have infringed and continue to infringe the ’059 Patent in violation of 35 U.S.C. § 271(a), either literally or through the doctrine of equivalents, by making, using, selling, offering for sale, and/or importing into the United States products and/or methods that practice at least claim 3 of the ’059 Patent, including with respect to the Meta Companies Coding Patents Accused Products.

194. On information and belief, the Meta Companies use the Meta Companies Coding Patents Accused Products for their own business purposes. In addition, the Meta Companies regularly conduct testing and troubleshooting of the Meta Companies Coding Patents Accused Products.[69] Further, VideoLabs believes companies related to the Meta Companies (e.g., other subsidiaries) use the ’059 Meta Companies Coding Patents Accused Products.

195. On information and belief, the Meta Companies’ infringement through their use of H.264 entropy coding, described below, is exemplary of all of the Meta Companies’ infringement with respect to all the Meta Companies Coding Patents Accused Products.

196. The Meta Companies Coding Patents Accused Products directly infringe at least claim 3 of the ’059 Patent by, for example, performing arithmetic decoding of coded blocks of picture data, where the probability table to be used is switched to another probability table in one

[69] *See, e.g.*, SFHTML5, *Facebook Messenger RTC – The Challenges and Opportunities of Scale*, YouTube (Oct. 27, 2017), https://www.youtube.com/watch?v=F7UWvflUZoc, at 10:24-12:06

direction, when the arithmetic-coded absolute values of the coefficient values include an absolute value exceeding a predetermined threshold value.

197. Each of the Meta Companies Coding Patents Accused Products meet every limitation of claim 3 of the ’059 Patent, which recites:

> 3. A decoding method comprising:
>
> receiving multiplexed data obtained by multiplexing (i) coded picture data that is obtained by coding a moving picture and (ii) audio data that is obtained by coding an audio signal;
>
> demultiplexing the multiplexed data received in said receiving into the coded picture data and the audio data;
>
> decoding the coded picture data into a first bit of binary data corresponding to each absolute value of coefficients of a two-dimensional array of frequency components, on a block basis, according to a predetermined scanning order starting at a high frequency component toward a low frequency component by using a plurality of probability tables, the coefficients being generated by frequency transformation performed on picture data of a block which has a predetermined size of pixels;
>
> switching between the plurality of probability tables, from a current probability table for the first bit of the binary data corresponding to a first coefficient to be decoded, to a new probability table for the first bit of the binary data corresponding to a second coefficient to be decoded, based on a result of a comparison between an absolute value of the first coefficient to be decoded and a predetermined threshold value; and
>
> decoding audio data;
>
> wherein, in said switching, the switching between the plurality of probability tables (i) is performed in a predetermined one direction within each block such that each of the probability tables, which has been used for performing arithmetic decoding on the first bit of the binary data corresponding to an already decoded coefficient before switching to the new probability table, is not used within each block after switching to the new probability table, and (ii) is not performed in the direction opposite to the predetermined one direction regardless of the result of the comparison,
>
> wherein, within each block, if a predetermined one of the plurality of probability tables has been used to perform arithmetic decoding, in said switching, the switching between the plurality of probability tables is not performed regardless of the result of the comparison,

> wherein the coded picture data and the audio data are coded by an arithmetic coding apparatus,
>
> wherein the arithmetic coding apparatus includes: a coefficients scanning unit configured to scan coefficients of frequency components, which are generated by frequency transformation performed on the picture data of a block which has a predetermined size of pixels, in a predetermined scanning order starting at a high frequency component toward a low frequency component;
>
> a converting unit configured to convert each absolute value of the coefficients into binary data;
>
> an arithmetic coding unit configured to perform arithmetic coding on a first bit of the binary data corresponding to each absolute value of the coefficients according to the predetermined scanning order by using a plurality of probability tables:
>
> a switching unit configured to switch between the plurality of probability tables, from a current probability table for the first bit of the binary data corresponding to a first coefficient to be coded, to a new probability table for the first bit of the binary data corresponding to a second coefficient to be coded, based on a result of a comparison between an absolute value of the first coefficient to be coded and a predetermined threshold value; and
>
> an audio coding unit configured to code an audio signal, wherein said switching unit is configured (i) to switch between the plurality of probability tables in a predetermined one direction within each block such that each of the probability tables, which has been used for performing arithmetic coding on the first bit of the binary data corresponding to an already coded coefficient before switching to the new probability table, is not used within each block after switching to the new probability table, and (ii) not to switch between the plurality of probability tables in the direction opposite to the predetermined one direction regardless of the result of the comparison, and
>
> wherein, within each block, if a predetermined one of the plurality of probability tables has been used to perform arithmetic coding, said switching unit is configured not to switch between the plurality of probability tables regardless of the result of the comparison.

198. Each of the Meta Companies Coding Patents Accused Products perform decoding by receiving multiplexed data, which is obtained by multiplexing coded picture data and coded audio data, and then demultiplexing the multiplexed data into coded picture data and coded audio

data. H.264 carries audio and video multiplexed in a single stream. H.264 is directed to the picture portion of video, so devices containing H.264 encoders and decoders must multiplex/demultiplex picture and audio data in order to obtain the H.264 picture data. Encoders multiplex the audio and pictures into a single stream, so that decoders receive the complete video presentation, including sound. Decoders receive the complete video presentation, including sound, and decode the stream to recreate the video. In the ISO Media File Format, which each accused H.264 product is capable of processing, a coded stream such as an H.264 video sequence or an audio stream is stored as a track, representing a sequence of coded data items or samples. Figure 8.32, below, illustrates an example of such multiplexed data, in which coded audio data ("audio track samples") and coded picture data ("video track samples") are multiplexed together:

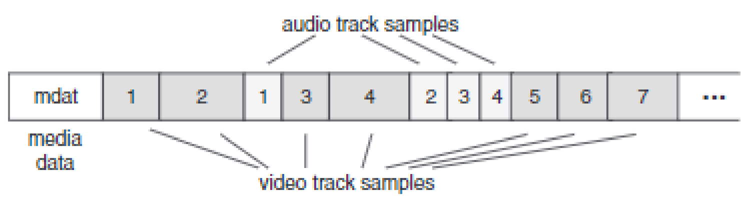


Figure 8.32 ISO Media File

Ex. 23, Richardson, at 247.

199. In the Meta Companies Coding Patents Accused Products, since the coded audio data and the coded picture data are decoded by separate respective decoders, it is necessary to first demultiplex the audio track samples and video track samples in order to reproduce the encoded data in its entirety. Figure 8.32 below shows the multiplexed stream of an ISO Media File as received, which includes coded audio data ("audio track samples") and coded picture data ("video track samples"). The coded audio data and the coded picture data are demultiplexed and sent to respective decoders for further processing.

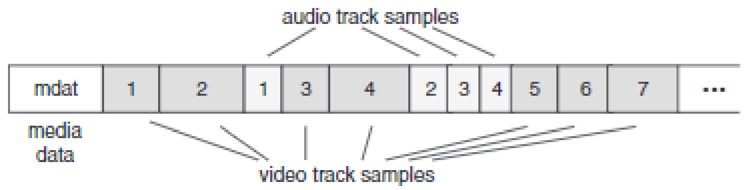


Figure 8.32 ISO Media File

Ex. 23, Richardson, at 247.

200. H.264-compliant decoders use one of two entropy decoders (CAVLC and CABAC) to extract binary data from a coded bitstream. Entropy encoders compress the parameters of the H.264 prediction model to remove statistical redundancy in the data, and then produce a H.264-compliant compressed bit stream or file for storage and/or transmission.

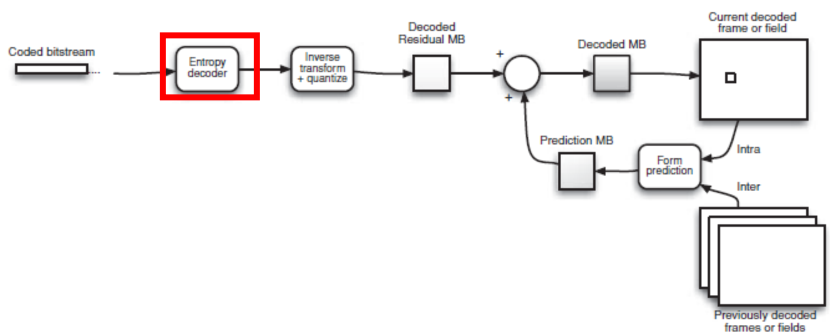


Figure 4.5 Typical H.264 decoder

Ex. 23, Richardson, at 85.

201. In the Meta Companies Coding Patents Accused Products, the H.264 decoder decodes the coded picture data into a first bit of binary data corresponding to each absolute value of coefficients of a two-dimensional array of frequency components, on a block basis, according to a predetermined scanning order starting at a high frequency component toward a low frequency component by using a plurality of probability tables, where the coefficients are generated by frequency transformation performed on picture data of a block which has a predetermined size of

pixels.

202. The H.264-compliant compressed bit stream or file sequence representing the coded picture data consists of coded prediction parameters, coded residual coefficients, and header information.

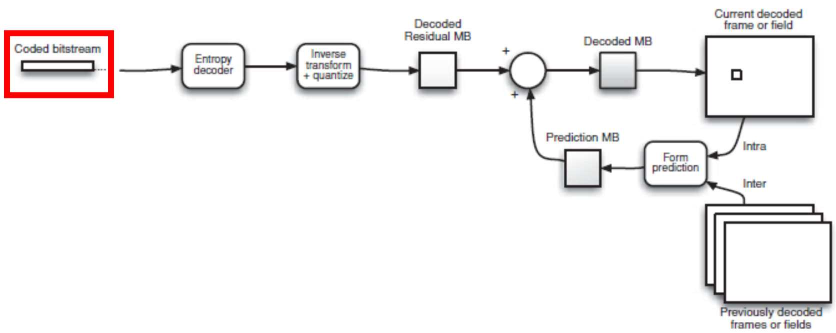


Figure 4.5 Typical H.264 decoder

Ex. 23, Richardson, at 85.

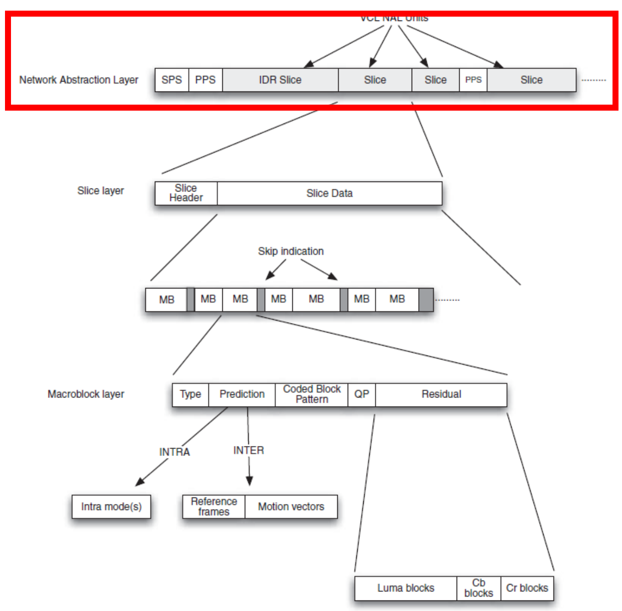


Figure 5.1 Syntax overview

Ex. 23, Richardson, at 100.

203. In accordance with the H.264 standard, H.264-compliant decoders, such as those in the Meta Companies Coding Patents Accused Products, operate on a macroblock, consisting of a

16 x 16 block of luma samples and two corresponding blocks of chroma samples. For example, picture data is coded in accordance with H.264 by dividing a picture signal into macroblocks, as shown in Figure 6-7:

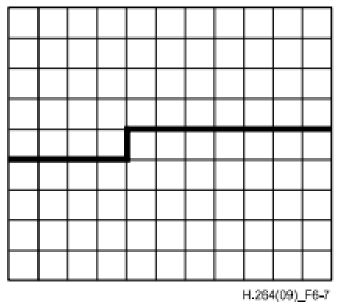


Figure 6-7 – A picture with 11 by 9 macroblocks that is partitioned into two slices

Id. at Figure 6-7.

204. The picture macroblocks are used to generate a residual block image. This residual may be obtained using intra- or inter- picture prediction. For example, Figure 6.6 below shows a picture signal to be coded, with the macroblock being coded highlighted. The macroblock is predicted using neighboring, previously-encoded samples, as shown in Figure 6.7; because this prediction looks only to the other macroblocks of the same picture, this is called intra-picture prediction. The predicted macroblock is shown in Figure 6.7. Figure 6.8 shows the prediction (Figure 6.7) subtracted from the original (Figure 6.6), which is called the residual.

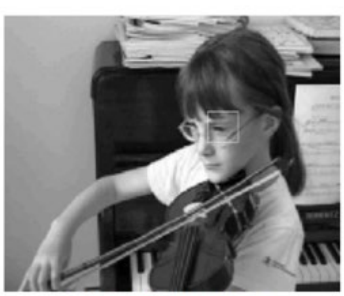


Figure 6.6 QCIF frame with highlighted macroblock

Figure 6.7 Predicted luma frame formed using H.264 intra prediction

Figure 6.8 Residual after subtracting intra prediction

Ex. 23, Richardson, at 141-143.

205. H.264-compliant encoders can use intra- and inter- picture coding on the macroblocks of a picture signal. Inter-picture coding predicts the value of the macroblock using temporal statistical dependencies between different pictures. Both types of prediction can be used to calculate the residual.

206. The resulting prediction residual is split into 4x4 blocks, and transformation and quantization are applied. An integer transform is applied to the residual, outputting a set of coefficient weighting values. This process is shown in the figures below:

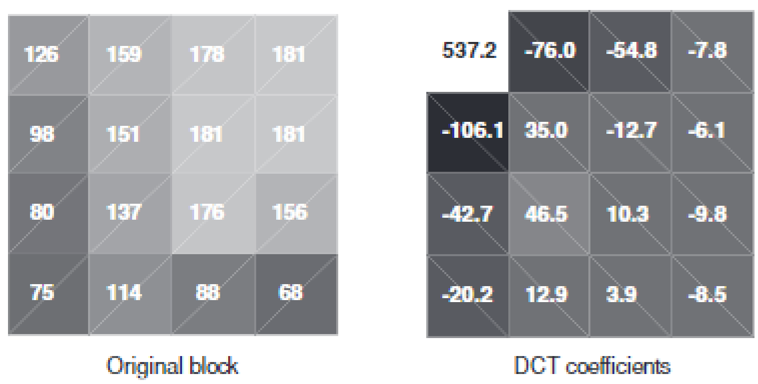


Figure 3.31 Close-up of 4 × 4 block; DCT coefficients

Ex. 23, Richardson, at 47.

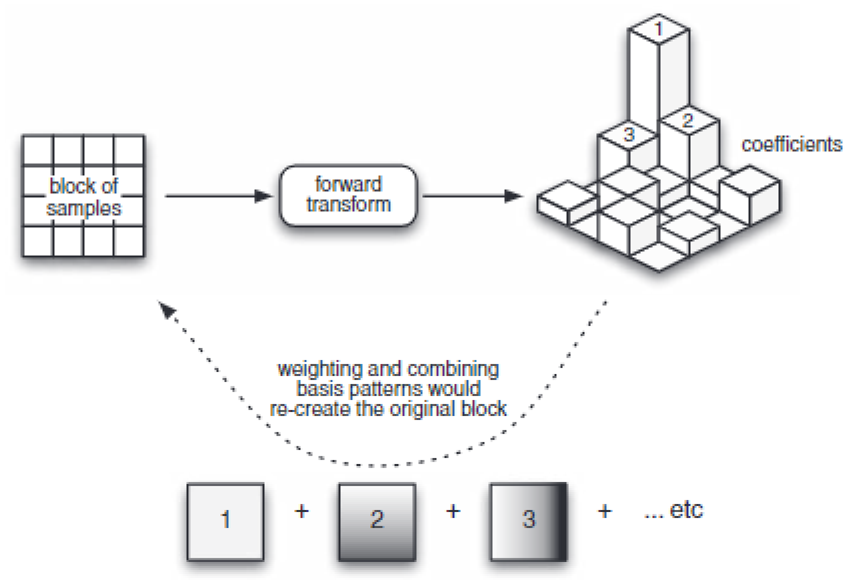


Figure 4.11 Forward transform

Ex. 23, Richardson, at 88.

207. H.264 specifies an *entropy_coding_mode* flag that dictates the entropy encoding algorithm used to encode the picture data. When this flag is set to "1" the residual block data is coded using a CABAC scheme.[70] In H.264, CABAC coding a data symbol involves binarizing the frequency transform coefficients, in scan order, and then further encoding the binary codes. Since CABAC is a context-adaptive binary arithmetic coding technique, it relies on probability model ("context model") selection for one or more bins of the binarized code. Context models and binarization schemes are specified in the H.264 standard.[71] The context model stores the probability of each bin being "1" or "0". An H.264-compliant arithmetic coder then encodes each bin according to the selected probability model and the selected context model is updated based

[70] *See* Ex. 60, H.264 Standard, Section 7.4.2.2, pp. 81-82 (09/2019).

[71] *See* Ex. 60, H.264 Standard, Section 9.3, pp. 223-278 (09/2019).

on the actual coded value for further encoding.[72] The H.264 decoder in the Meta Companies Coding Patents Accused Products reverses this process to decode the picture data.

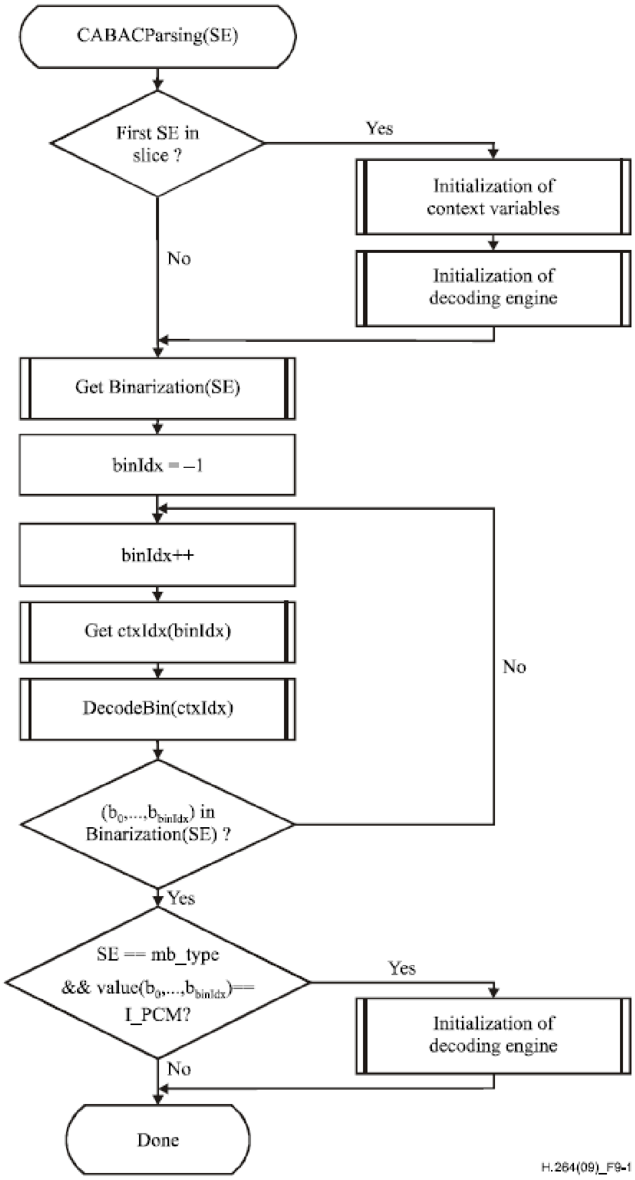


Figure 9-1 – Illustration of CABAC parsing process for a syntax element SE (informative)

208. The Meta Companies Coding Patents Accused Products perform CABAC decoding by switching between the plurality of probability tables, from a current probability table for the

[72] *See* Ex. 60, H.264 Standard, Sections 7.4.5.3.3, 9.3, and Figure 9-1, pp. 110-111, 223-278 (09/2019).

first bit of the binary data corresponding to a first coefficient to be decoded, to a new probability table for the first bit of the binary data corresponding to a second coefficient to be decoded.

209. In accordance with H.264 standard requirements, at the beginning of each coded slice, the context models are initialized. Initializing the context models produces context model tables that are accessed by index "ctxIdx."[73] The CABAC decoding engine in the Meta Companies Coding Patents Accused Products applies a specific table to decode each bin of a H.264 syntax element, such as the residual block. In particular, the H.264 standard defines six sets of probability tables assigned to residual block types. Residual blocks in categories 0-4 are assigned a group of tables starting at table ctxIdx_227. LumaLevel8x8 blocks (category 5) are assigned a group of tables starting at table ctxIdx_426. The first table within a group is denoted as ctxIdxOffset.[74]

---

[73] *See* Ex. 60, H.264 Standard, Section 9.3.1, Table 9-11, pp. 225-228 (09/2019).

[74] *See* Ex. 60, H.264 Standard, Section 9.3.3.1.3, Tables 9-34 and 9-42, pp. 249-251, 267-270.

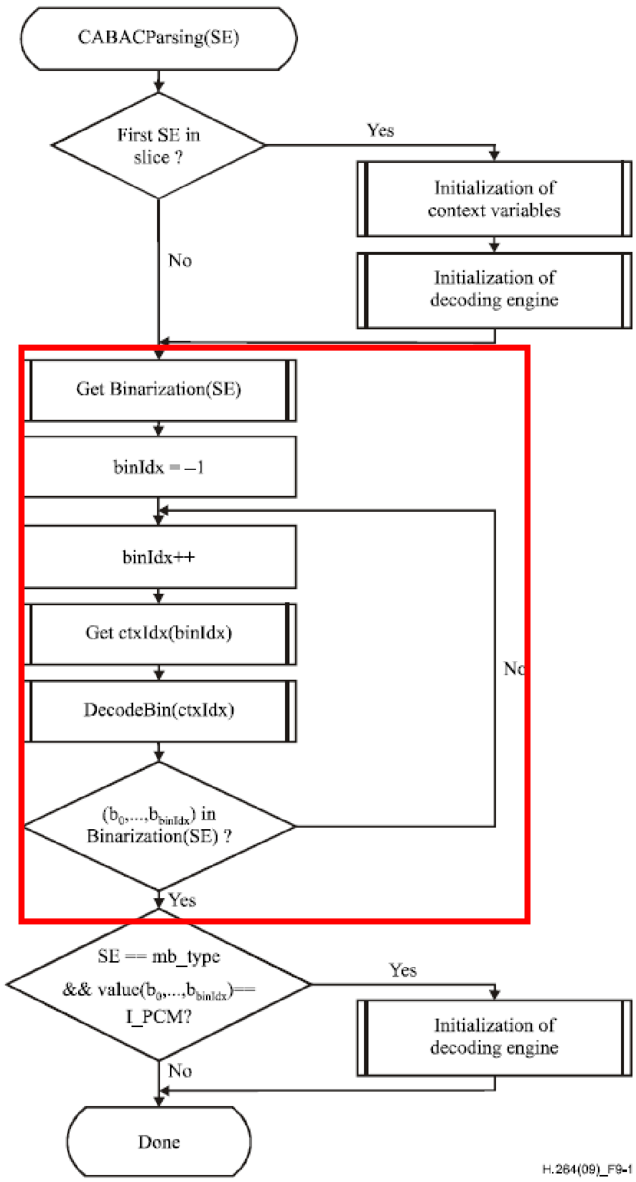


Figure 9-1 – Illustration of CABAC parsing process for a syntax element SE (informative)

Ex. 60, H.264 Standard, at 225.

210. To effect the table switching mandated by the H.264 standard, the CABAC decoder in the Meta Companies Coding Patents Accused Products computes index ctxIdx to access the table for decoding the first bit (b0) corresponding to a first transform coefficient to be decoded. To decode the first bit (b0) corresponding to a second transform coefficient to be decoded the

CABAC decoder computes another index ctxIdx to access the corresponding table.[75] Index ctxIdx is specified by the H.264 standard to be the sum of ctxIdxOffset and ctxIdxBlockCatOffset(ctxBlockCat) and variable ctxIdxInc.[76]

211. The Meta Companies Coding Patents Accused Products switch between the plurality of probability tables based on a result of a comparison between an absolute value of the first coefficient to be decoded and a predetermined threshold value. Equation 9-23 of the H.264 Specification mandates that the context increment index "ctxIdxInc" is incremented by one (i.e., switching to a new probability table correlated to ctxIdx), so long as the absolute value of the corresponding decoded coefficient is 1 and the number of the previous decoded single bit coefficients is less than 4.

Let numDecodAbsLevelEq1 denote the accumulated number of decoded transform coefficient levels with absolute value equal to 1, and let numDecodAbsLevelGt1 denote the accumulated number of decoded transform coefficient levels with absolute value greater than 1. Both numbers are related to the same transform coefficient block, where the current decoding process takes place. Then, for decoding of coeff_abs_level_minus1, ctxIdxInc for coeff_abs_level_minus1 is specified depending on binIdx as follows:

– If binIdx is equal to 0, ctxIdxInc is derived by

$$ctxIdxInc = ((numDecodAbsLevelGt1 \ != \ 0) \ ? \ 0 : Min(4, 1 + numDecodAbsLevelEq1)) \quad (9\text{-}23)$$

Ex. 60, H.264 Standard, at 270, Section 9.3.3.1.

212. The Meta Companies Coding Patents Accused Products also decode audio data. For example, the Meta Companies Coding Patents Accused Products incorporate an audio codec that is configured to decode audio data according to an input format, including without limitation, AAC.

[75] *See* Ex. 60, H.264 Standard, Section 9.3.3.1.3, Equations 9-23 and 9-24, pp. 267-270 (09/2019).

[76] *See* Ex. 60, H.264 Standard, Section 9.3.3.1, Table 9-40, pp. 257-259 (09/2019).

213. In the Meta Companies Coding Patents Accused Products the switching between the plurality of probability tables is performed in a predetermined one direction within each block such that each of the probability tables which has been used for performing arithmetic decoding on the first bit of binary data corresponding to an already decoded coefficient before switching to the new probability table is not used within each block after switching to the new probability table, and is not performed in the opposite direction to the predetermined one direction regardless of the result of the comparison. In accordance with Equation 9-23, the H.264-compliant CABAC decoder in the Meta Companies Coding Patents Accused Products switches between the plurality of probability tables in a predetermined increasing direction (i.e., from ctxIdxInc 1 to 4) and will not reverse direction, as long as the absolute value of the corresponding decoded transform coefficient is 1 and the absolute value of the previously decoded coefficient is not greater than 1 (i.e., the switching increment ctxIdxInc for bin0 of a transform coefficient increases monotonically from 1 to 4 with each successive coefficient to be decoded and will not reverse direction).[77]

214. Similarly, in the Meta Companies Coding Patents Accused Products within each block if a predetermined one of the plurality of probability tables has been used to perform arithmetic decoding, in said switching, the switching between the plurality of probability tables is not performed regardless of the result of the comparison. For H.264-compliant CABAC decoders, such as those in the Meta Companies Coding Patents Accused Products, Equation 9-23 mandates that no switching between the plurality of tables is performed when the table ctxIdx corresponding to ctxIdxInc = 4 has been used to decode a transform coefficient with absolute value 1 and bin0 of the next single-bit coefficient is received. In such case, the same ctxIdx table will be used for decoding additional trailing 1s without further table switching. Additionally, after decoding a

[77] *See generally* H.264 Standard, Equation 9-23, p. 270 (09/2019).

transform coefficient with absolute value greater than 1, the table ctxIdx corresponding to ctxIdxInc = 0 will be used and no further switching will occur.[78]

215. H.264-compliant decoders, such as those in the Meta Companies Coding Patents Accused Products, are configured to decode coded picture data, including picture data that has already been encoded by an arithmetic coding apparatus using CABAC entropy encoding. Products capable of decoding H.264-compliant coded picture data, including the Meta Companies Coding Patents Accused Products, typically also incorporate an audio codec to decode related audio data previously encoded in accordance with various audio coding standards. An arithmetic coding apparatus, whether embodied in the Meta Companies Coding Patents Accused Products or otherwise, typically constitutes a H.264-compliant codec for coding picture data and one or more audio codecs for coding the audio data.

216. A H.264-compliant arithmetic coding apparatus, whether embodied in the Meta Companies Coding Patents Accused Products or otherwise embodied separately, includes a coefficient scanning unit configured to scan coefficients of frequency components, which are generated by frequency transformation performed on the picture data of a block which has a predetermined scanning order starting at a high frequency component toward a low frequency component. As noted previously, H.264-compliant encoders operate on a macroblock, consisting of a 16 x 16 block of luma samples and two corresponding blocks of chroma samples. A macroblock can be further portioned for inter-prediction forming segmentations for motion representation as small as 4 x 4 luma samples in size.[79] Two main coding types are specified in H.264, intra-coding and inter-coding. Intra-coding is done without reference to other pictures

---

[78] *See* Ex. 60, H.264 Standard, Equation 9-23, p. 270 (09/2019).

[79] *See* Ex. 60, H.264 Standard, Section 0.6.3, p. 5 (09/2019).

while inter-coding uses inter-prediction of each block of sample values from some previously decoded picture.[80] H.264 decoding is based on the use of a block-based transform method for spatial redundancy removal. The resulting residual block is split into 4 x 4 blocks. These residual blocks are converted into the transform domain where they are quantized.[81] H.264 specifies an *entropy_coding_mode* flag that dictates the entropy encoding algorithm used to encode the picture data. When this flag is set to "1" the residual block data is coded using a CABAC scheme.[82]

217. A H.264-compliant arithmetic coding apparatus, whether embodied in the Meta Companies Coding Patents Accused Products or otherwise embodied separately, includes a converting unit configured to convert each absolute value of the coefficients into binary data. In H.264 CABAC encoding generally, coding a data symbol involves binarizing the frequency transform coefficients, in scan order, and then further encoding the binary codes.

218. A H.264-compliant arithmetic coding apparatus, whether embodied in the Meta Companies Coding Patents Accused Products or otherwise embodied separately, includes an arithmetic coding unit configured to perform arithmetic coding on a first bit of the binary data corresponding to each absolute value of the coefficients according to the predetermined scanning order by using a plurality of probability tables. Since CABAC is a context-adaptive binary arithmetic coding technique, it relies on probability model ("context model") selection for one or more bins of the binarized code. Context models and binarization schemes are defined in the H.264 standard.[83] The context model stores the probability of each bin being "1" or "0". An

[80] *See* Ex. 60, H.264 Standard, Section 0.6.1, p. 4 (09/2019).

[81] *See* Ex. 60, H.264 Standard, Section 0.6.4, p. 5 (09/2019).

[82] *See* Ex. 60, H.264 Standard, Section 7.4.2.2, pp. 81-82 (09/2019).

[83] *See* Ex. 60, H.264 Standard, Section 9.3, pp. 223-278 (09/2019).

arithmetic coder then encodes each bin according to the selected probability model and the selected context model is updated based on the actual coded value for further encoding.[84]

219. A H.264-compliant arithmetic coding apparatus, whether embodied in the Meta Companies Coding Patents Accused Products or otherwise embodied separately, includes a switching unit configured to switch between the plurality of probability tables, from a current probability table for the first bit of the binary data corresponding to a first coefficient to be coded, to a new probability table for the first bit of the binary data corresponding to a second coefficient to be coded, based on a result of a comparison between an absolute value of the first coefficient to be coded and a predetermined threshold value. In accordance with H.264 CABAC encoding, at the beginning of each coded slice, the context models are initialized. Initializing the context models produces context model tables that are accessed by index "ctxIdx".[85] The CABAC encoder applies a specific table to encode each bin of a H.264 syntax element, such as a residual block. In particular, the H.264 standard defines six sets of probability tables assigned to residual block types. Residual blocks in categories 0-4 are assigned a group of tables starting at table ctxIdx_227. LumaLevel8x8 blocks (category 5) are assigned a group of tables starting at table ctxIdx_426. The first table within a group is denoted as ctxIdxOffset.[86]

220. H.264-compliant encoders, whether embodied in the Meta Companies Coding Patents Accused Products or embodied separately, necessarily incorporate switching logic to effect switching between probability tables in accordance with H.264 requirements when performing CABAC entropy encoding. Specifically, to encode the transform coefficient block, H.264-

[84] *See* Ex. 60, H.264 Standard, Sections 7.4.5.3.3, 9.3, and Figure 9-1, pp. 110-111, 223-278 (09/2019).

[85] *See* Ex. 60, H.264 Standard, Section 9.3.1, Table 9-11, pp. 225-228 (09/2019).

[86] *See* Ex. 60, H.264 Standard, Section 9.3.3.1.3, Tables 9-34 and 9-42, pp. 249-251, 267-270 (09/2019).

compliant encoders must implement switching logic that follows the mandates of Equations 9-23 and 9-24 of the H.264 Standard, depending on whether the absolute value of transform coefficients is greater than or equal to 1:[87]

– If binIdx is equal to 0, ctxIdxInc is derived by

$$\text{ctxIdxInc} = ((\text{numDecodAbsLevelGt1} \;!=\; 0) \;?\; 0 : \text{Min}(4, 1 + \text{numDecodAbsLevelEq1})) \qquad (9\text{-}23)$$

– Otherwise (binIdx is greater than 0), ctxIdxInc is derived by

$$\text{ctxIdxInc} = 5 + \text{Min}(4 - ((\text{ctxBlockCat} == 3) \;?\; 1 : 0), \text{numDecodAbsLevelGt1}) \qquad (9\text{-}24)$$

221. To effect the mandated table switching, the CABAC entropy encoder computes index ctxIdx to access the table for encoding the first bit (b0) corresponding to a first transform coefficient to be encoded. To encode the first bit (b0) corresponding to a second transform coefficient to be encoded, the CABAC entropy encoder computes another index ctxIdx to access the corresponding table.[88] Index ctxIdx is specified in the H.264 Standard to be the sum of ctxIdxOffset and ctxIdxBlockCatOffset(ctxBlockCat) and variable ctxIdxInc.[89] In accordance with Equation 9-23, as long as the absolute value of the corresponding encoded transform coefficient is 1 and the number of previous encoded single bit coefficients is less than 4, the context increment index ctxIdxInc is incremented by 1 (i.e., the switching unit switches to a new probability table correlated to ctxIdx).

222. Typical audio encoders, whether embodied in the Meta Companies Coding Patents Accused Products or embodied separately, as an element of the arithmetic coding apparatus, include an audio coding unit configured to code an audio signal in a standard audio format.

[87] *See* Ex. 60, H.264 Standard, Section 9.3.3.1.3, Equations 9-23 and 9-24, pp. 267-270 (09/2019).

[88] *See* Ex. 60, H.264 Standard, Section 9.3.3.1.3, Equations 9-23 and 9-24, pp. 267-270 (09/2019).

[89] *See* Ex. 60, H.264 Standard, Section 9.3.3.1, Table 9-40, pp. 257-259 (09/2019).

223. In H.264-compliant encoders, whether embodied in the Meta Companies Coding Patents Accused Products or embodied separately, the switching unit is configured to switch between the plurality of probability tables in a predetermined one direction within each block such that each of the probability tables, which has been used for performing arithmetic coding on the first bit of the binary data corresponding to an already coded coefficient before switching to the new probability table, is not used within each block after switching to the new probability table and not to switch between the plurality of probability tables in the direction opposite to the predetermined one direction regardless of the result of the comparison. In accordance with Equation 9-23, for H.264-compliant encoders, the switching between the plurality of tables is performed in a predetermined increasing direction (i.e., from ctxIdxInc 1 to 4) and will not reverse direction, as long as the absolute value of the corresponding encoded transform coefficient is 1 and the absolute value of the previous encoded coefficient is not greater than 1 (i.e., the switching increment ctxIdxInc for bin0 of a transform coefficient increases monotonically from 1 to 4 with each successive coefficient to be encoded and will not reverse direction).[90]

224. Similarly, in H.264-compliant encoders, whether embodied in the Meta Companies Coding Patents Accused Products or embodied separately, within each block, if a predetermined one of the plurality of the probability tables has been used to perform arithmetic coding, said first switching unit is configured not to switch between the plurality of probability tables regardless of the result of the comparison. Equation 9-23 mandates that no switching between the plurality of tables is performed when the table ctxIdx corresponding to ctxIdxInc = 4 has been used to encode a transform coefficient with absolute value 1 and bin0 of the next single-bit coefficient is to be coded. In such case the same ctxIdx table will be used for encoding additional trailing 1s without

[90] *See* Ex. 60, H.264 Standard, Equation 9-23, pp. 270 (09/2019).

further table switching. Additionally, after encoding a transform coefficient with an absolute value greater than 1, the table ctxIdx corresponding to ctxIdxInc = 0 will be used for subsequent coefficient coding and no further switching will occur.[91]

225. On information and belief, to the extent applicable, VideoLabs has complied with 35 U.S.C. § 287(a) with respect to the '059 Patent.

226. VideoLabs representatives reached out to Meta representatives at least as early as October 2, 2019 regarding VideoLabs' platform and to gauge the Meta Companies' interest in joining as a partner or member. In October, 2019, VideoLabs representatives presented VideoLabs' platform to representatives of the Meta Companies. VideoLabs representatives and Meta representatives spoke between December and February, and met in person on January 30, 2020, during which VideoLabs presented them with an updated proposal. On June 3, 2020, VideoLabs representatives reached out again and stated that "With regards to Facebook, and based on our analysis so far, we have determined that VideoLabs' current patent portfolio is relevant to the majority of Facebook's annual revenue in some way and we have 17 unique claim charts completed or in development (and other evidence of use) so far related to Facebook Messenger, Facebook Live, Oculus, Instagram, Facebook Workplace, Portal, Facebook Ads, and backend infrastructure" and asked them to engage in "good faith licensing discussions" under a proposed NDA. VideoLabs representatives sent another updated proposal on March 23, 2020. VideoLabs representatives reached out again on June 26, 2020, March 16, 2021, and May 4, 2021 to "request[] good faith licensing negotiations" regarding the number of patents relevant to Facebook and other Meta services. On June 26, 2020, VideoLabs confirmed in an email that representatives of the Meta Companies said on a recent call that the Meta Companies were not interested in good faith

[91] *See* Ex. 60, H.264 Standard, Equation 9-23, p. 270 (09/2019).

licensing discussions. To date, months later, the Meta Companies have not reengaged with VideoLabs.

227. The Meta Companies of course know how their products operate, and on information and belief, the Meta Companies investigated the ’059 Patent and their infringement of the Meta Companies Coding Patents Accused Products. The Meta Companies have been given further notice of the ’059 Patent and their infringement of the ’059 Patent through VideoLabs’ May 23, 2022 letter, VideoLabs’ May 24, 2022 Complain, and the filing of this First Amended Complaint. On information and belief, the Meta Companies are either knowingly infringing the ’059 Patent or are willfully blind to their infringement, and continue to act in wanton disregard of VideoLabs’ patent rights.

228. Despite becoming aware of or willfully blinding itself to its infringement of the ’059 Patent, the Meta Companies have nonetheless continued to engage in and has escalated their infringing activities by continuing to develop, advertise, make available, and use the infringing functionalities of the Meta Companies Coding Patents Accused Products. On information and belief, the Meta Companies have made no attempts to design around the ’059 Patent or otherwise stop their infringing behavior.

229. The Meta Companies’ infringement of the ’059 Patent therefore has been and remains willful.

230. The Meta Companies also indirectly infringe the ’059 Patent by inducing others to infringe and contributing to the infringement of others, including other Defendants and third-party users of the Meta Companies Coding Patents Accused Products in this District and throughout the United States. As described above, on information and belief, the Meta Companies have known about the family of patents including the ’059 Patent since at least January 30, 2020 and have

known about the ’059 Patent since May 23, 2021.

231. On information and belief, the Meta Companies have actively induced the infringement of the ’059 Patent under 35 U.S.C. § 271(b) by actively inducing the infringement of the Meta Companies Coding Patents Accused Products by third parties in the United States. The Meta Companies knew or were willfully blind to the fact that their conduct would induce these third parties to act in a manner that infringes the ’059 Patent in violation of 35 U.S.C. § 271(a).

232. The Meta Companies actively encouraged and continue to actively encourage third parties to directly infringe the ’059 Patent by, for example, marketing the Coding Patents Accused Products and infringing functionalities to consumers; working with consumers to implement, install and/or operate the Coding Patents Accused Products and infringing functionalities; fully supporting and managing consumers’ continuing use of the Coding Patents Accused Products and infringing functionalities; and providing technical assistance to consumers during their continued use of the Coding Patents Accused Products and infringing functionalities.[92]

[92] *See e.g.,* Ex. 71, *Using Facebook-Your Photos and Videos-Videos*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/1738143323068602 (last visited May 24, 2022); Ex. 72, *I can’t add a video on Facebook*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/215726848451641 (last visited May 24, 2022); Ex. 73, *Audio and Video Calling*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/messenger-app/1673374996287506 (last visited May 24, 2022); Ex. 74, *How do I video chat with someone or a group in Messenger?*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/messenger-app/1414800065460231 (last visited May 24, 2022); Ex. 75, *Instagram Features-Videos*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/381435875695118 (last visited May 24, 2022); Ex. 76, *How do I share a video on Instagram?*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/456185931138729 (last visited May 24, 2022); Ex. 77, *How do I share my video on Instagram to my Facebook Page?*, INSTAGRAM HELP CENTER (2022), https://www.facebook.com/help/instagram/486878428409681 (last visited May 24, 2022); Ex. 78, *Oculus Support*, https://www.youtube.com/channel/UCxLTeQQFvA5_eb0VkrLIq6w (last visited May 24, 2022); Ex. 79, *Viewing your own 360 videos on your Gear VR*; META, https://support.giphy.com/hc/en-us/articles/360019674452-How-To-Make-A-GIF (last visited May 24, 2022); Ex. 80, *How to make a video call*, WHATSAPP,

233. On information and belief, the Meta Companies contributorily infringe the '059 Patent under 35 U.S.C. § 271(c) by importing, selling, and/or offering to sell within the United States the Meta Companies Coding Patents Accused Products (or components thereof) that constitute a material part of the claimed invention and are not staple articles of commerce suitable for substantial non-infringing use. For example, the hardware and/or software for decoding content with H.264 using CABAC is material, has no insubstantial non-infringing uses, and is known by the Meta Companies to be especially made or adapted for use in a manner that infringes the '059 Patent.

COUNT IV
INFRINGEMENT OF U.S. PATENT NO. 7,266,682

234. VideoLabs incorporates by reference the foregoing paragraphs of this First Amended Complaint as if fully set forth herein.

235. VL IP is the assignee and lawful owner of all right, title, and interest in and to the '682 Patent. The '682 Patent is valid and enforceable.

236. On information and belief, the Meta Companies have infringed and continue to infringe the '682 Patent in violation of 35 U.S.C. § 271(a), either literally or through the doctrine of equivalents, by using methods and/or taking steps that practice at least claim 1 of the '682 Patent, including with respect to the Meta Companies '682 Accused Products.

237. On information and belief, the Meta Companies use the Meta Companies '682 Accused Products for their own business purposes. In addition, the Meta Companies regularly

---

https://faq.whatsapp.com/android/voice-and-video-calls/how-to-make-a-video-call/?lang=en (last visited May 24, 2022); Ex. 81, *I get a message that my video is too long and it won't send*, META, https://faq.whatsapp.com/general/i-get-a-message-that-my-video-is-too-long-and-it-wont-send/?lang=en (last visited May 24, 2022).

conduct testing and troubleshooting of the Meta Companies ’682 Accused Products.[93] Further, VideoLabs believes companies related to the Meta Companies (e.g., other subsidiaries) use the Meta Companies ’682 Accused Products.

238. On information and belief, the Meta Companies’ infringement through its use of WebRTC, described below, is exemplary of all of Meta Companies' infringement with respect to all the Meta Companies ’682 Accused Products.

239. The Meta Companies ’682 Accused Products directly infringe at least claim 1 of the ’682 Patent by, for example, performing SRTP/SRTCP authentication via streaming using the WebRTC standard.

240. The Meta Companies ’682 Accused Products meet every limitation of claim 1 of the ’682 Patent, which recites:

> 1. A method for transmitting data from a transmitter to a receiver, comprising:
>
> providing transmitter-to-receiver authentication at a Real Time Transport Protocol (RTP) packet level as an application protocol on an application layer by inserting, at the transmitter, authentication data at end of a whole RTP packet payload;
>
> ascertaining, by the receiver, whether the receiver knows the transmitter based on the RTP packet level authentication data; and
>
> accepting, by the receiver, the whole RTP packet payload, if the receiver knows the transmitter, and otherwise rejecting the whole RTP packet payload.

241. The Meta Companies ’682 Accused Products practice a method for transmitting data from a transmitter to a receiver. The WebRTC framework provides support for direct interactive rich communication using audio, video, text, collaboration, games, etc., between two

[93] *See, e.g.*, SFHTML5, *Facebook Messenger RTC – The Challenges and Opportunities of Scale*, YouTube (Oct. 27, 2017), https://www.youtube.com/watch?v=F7UWvflUZoc, at 10:24-12:06

peers' web browsers.[94]

242. The Meta Companies '682 Accused Products provide transmitter-to-receiver authentication at a Real Time Transport Protocol (RTP) packet level as an application protocol on an application layer. The accused WebRTC Products provide secure authentication by using RTP payloads. *See* Ex. 82, WebRTC Spec at 8 ("WebRTC Endpoints . . . MUST employ the full RTP/SAVPF profile to protect all RTP and RTPC packets that are generated (i.e., implementations MUST use SRTP and SRTCP)"). SRTP, a profile of RTP, is a real-time transport protocol which provides confidentiality, message authentication, and replay protection to the RTP traffic and to the control traffic for RTP, RTCP (the Real-time Transport Control Protocol).[95] SRTP is an application protocol operating on an application layer of the standard OSI model, residing between the RTP application and the transport layer.[96]

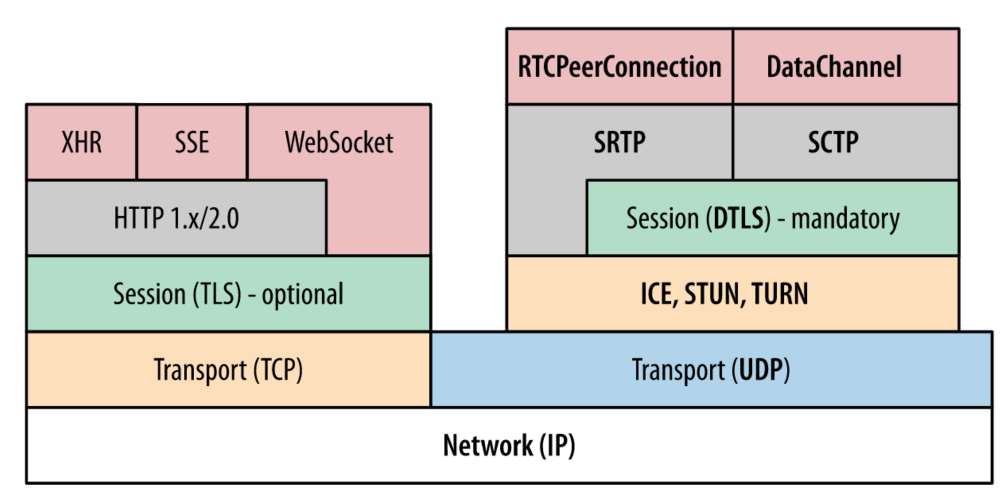
243. WebRTC uses DTLS-SRTP to add encryption, message authentication and integrity, and replay attack protection. As such, SRTP is a key component of the security in WebRTC. With WebRTC, SRTP is used for encrypting media streams. The image below illustrates the role of SRTP in WebRTC.[97]

[94] *See* Ex. 82, RTCWEB Working Group, Web Real-Time Communication (WebRTC): Media Transport and Use of RTP (June 12, 2015) (hereinafter "WebRTC Spec"), at 1.

[95] *See* Ex. 83, *The Secure Real-time Transport Protocol,* IETF (Mar. 2004) (hereinafter "RFC 3711"), at 3, available at https://datatracker.ietf.org/doc/html/rfc3711 (last visited May 24, 2022).

[96] *See* Ex. 83, RFC 3711, Section 3, p. 5.

[97] *See* Ex. 61, Ilya Grigorik, *High Performance Browser Networking, WebRTC: Browser APIs and Protocols*, at Figure 18-3, available at https://hpbn.co/webrtc/.



244. The Meta Companies provide this authentication by inserting, at the transmitter, authentication data at end of a whole RTP packet payload. As shown below, the Meta Companies insert an authentication tag at the end of a whole RTP packet payload. The authentication tag is used to carry message authentication data.[98]

[98] *See* Ex. 83, RFC 3711, Section 3.1, pp. 6-7.

```
3.1.  Secure RTP

   The format of an SRTP packet is illustrated in Figure 1.

        0                   1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+<+
     |V=2|P|X|  CC   |M|     PT      |       sequence number         | |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |
     |                           timestamp                           | |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |
     |           synchronization source (SSRC) identifier            | |
     +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+ |
     |            contributing source (CSRC) identifiers             | |
     |                               ....                            | |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |
     |                   RTP extension (OPTIONAL)                    | |
   +>+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |
   | |                          payload  ...                         | |
   | |                               +-------------------------------+ |
   | |                               | RTP padding   | RTP pad count | |
   +>+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+<+
   | ~                     SRTP MKI (OPTIONAL)                       ~ |
   | +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |
   | :                 authentication tag (RECOMMENDED)              : |
   | +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ |
   |                                                                   |
   +- Encrypted Portion*                      Authenticated Portion ---+
```

Ex. 83, RFC 3711, at 6.

245. Further, the Meta Companies ’682 Accused Products ascertain, by the receiver, whether the receiver knows the transmitter based on the RTP packet level authentication data. In WebRTC, “[t]o authenticate and decrypt an SRTP packet, the receiver SHALL do” tasks including “verification of the authentication tag,” which is performed “using the rollover counter . . . , the authentication algorithm indicated in the cryptographic context, and the session authentication key.”[99]

246. Finally, the Meta Companies ’682 Accused Products accept, by the receiver, the whole RTP packet payload, if the receiver knows the transmitter, and otherwise reject the whole RTP packet payload. Specifically, at the receiver, “[i]f the result is ‘AUTHENTICATION FAILURE’ . . . , the packet MUST be discarded from further processing.”[100] However, if the

[99] *See* Ex. 83, RFC 3711, Section 3.3, p. 12.

[100] *See* Ex. 83, RFC 3711, Section 3.3, p. 12.

authentication is successful, the receiver "[d]ecrypt[s] the Encrypted Portion of the packet . . . using the decryption algorithm indicated in the cryptographic context, the session encryption key and salt (if used)… with the index [of the SRTP packet]."[101]

247. On information and belief, to the extent applicable, VideoLabs has complied with 35 U.S.C. § 287(a) with respect to the '682 Patent.

248. VideoLabs representatives reached out to Meta representatives at least as early as October 2, 2019 regarding VideoLabs' platform and to gauge the Meta Companies' interest in joining as a partner or member. In October, 2019, VideoLabs representatives presented VideoLabs' platform to representatives of the Meta Companies. VideoLabs representatives and Meta representatives spoke between December and February, and met in person on January 30, 2020, during which VideoLabs presented them with an updated proposal. On June 3, 2020, VideoLabs representatives reached out again and stated that "With regards to Facebook, and based on our analysis so far, we have determined that VideoLabs' current patent portfolio is relevant to the majority of Facebook's annual revenue in some way and we have 17 unique claim charts completed or in development (and other evidence of use) so far related to Facebook Messenger, Facebook Live, Oculus, Instagram, Facebook Workplace, Portal, Facebook Ads, and backend infrastructure" and asked them to engage in "good faith licensing discussions" under a proposed NDA. VideoLabs representatives sent another updated proposal on March 23, 2020. VideoLabs representatives reached out again on June 26, 2020, March 16, 2021, and May 4, 2021 to "request[] good faith licensing negotiations" regarding the number of patents relevant to Facebook and other Meta services. On June 26, 2020, VideoLabs confirmed in an email that representatives of the Meta Companies said on a recent call that the Meta Companies were not interested in good faith

[101] *See* Ex. 83, RFC 3711, Section 3.3, p. 12.

licensing discussions. To date, months later, the Meta Companies have not reengaged with VideoLabs.

249. The Meta Companies of course know how their products operate, and on information and belief, they investigated the ’682 Patent and their infringement of the Meta Companies ’682 Accused Products. The Meta Companies have been given further notice of the ’682 Patent and its infringement of the ’682 Patent through VideoLabs’ May 23, 2022 letter, VideoLabs’ May 24, 2022 Complaint, and the filing of this First Amended Complaint. On information and belief, the Meta Companies are either knowingly infringing the ’682 Patent or are willfully blind to its infringement, and continues to act in wanton disregard of VideoLabs’ patent rights.

250. Despite becoming aware of or willfully blinding itself to its infringement of the ’682 Patent, the Meta Companies have nonetheless continued to engage in and have escalated their infringing activities by continuing to develop, advertise, make available, and use the infringing functionalities of the Meta Companies ’682 Accused Products. On information and belief, the Meta Companies have made no attempts to design around the ’682 Patent or otherwise stop their infringing behavior.

251. The Meta Companies’ infringement of the ’682 Patent therefore has been and remains willful.

252. The Meta Companies also indirectly infringe the ’682 Patent by inducing others to infringe and contributing to the infringement of others, including other Defendants and third-party users of the Meta Companies ’682 Accused Products in this District and throughout the United States. As described above, on information and belief, the Meta Companies has known about the family of patents including the ’682 Patent since at least January 30, 2020, and have known about

the '682 Patent since May 23, 2021.

253. On information and belief, the Meta Companies have actively induced the infringement of the '682 Patent under 35 U.S.C. § 271(b) by actively inducing third parties in the United States to use methods and/or take steps that practice at least claim 1 of the '682 Patent with respect to the Meta Companies '682 Accused Products. The Meta Companies knew or were willfully blind to the fact that their conduct would induce these third parties to act in a manner that infringes the '682 Patent in violation of 35 U.S.C. § 271(a).

254. The Meta Companies actively encouraged and continue to actively encourage third parties to directly infringe the '682 Patent by, for example, marketing the '682 Accused Products and WebRTC to consumers; working with consumers to implement, install and/or operate the '682 Accused Products and WebRTC; fully supporting and managing consumers' continuing use of the '682 Accused Products and WebRTC; and providing technical assistance to consumers during their continued use of the '682 Accused Products and WebRTC.[102]

[102] *See e.g.,* Ex. 71, *Using Facebook-Your Photos and Videos-Videos*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/1738143323068602 (last visited May 24, 2022); Ex. 72, *I can't add a video on Facebook*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/215726848451641 (last visited May 24, 2022); Ex. 73, *Audio and Video Calling*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/messenger-app/1673374996287506 (last visited May 24, 2022); Ex. 74, *How do I video chat with someone or a group in Messenger?*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/messenger-app/1414800065460231 (last visited May 24, 2022); Ex. 75, *Instagram Features-Videos*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/381435875695118 (last visited May 24, 2022); Ex. 76, *How do I share a video on Instagram?*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/456185931138729 (last visited May 24, 2022); Ex. 77, *How do I share my video on Instagram to my Facebook Page?*, INSTAGRAM HELP CENTER (2022), https://www.facebook.com/help/instagram/486878428409681 (last visited May 24, 2022); Ex. 78, *Oculus Support*, https://www.youtube.com/channel/UCxLTeQQFvA5_eb0VkrLIq6w (last visited May 24, 2022); Ex. 79, *Viewing your own 360 videos on your Gear VR*; META, https://support.giphy.com/hc/en-us/articles/360019674452-How-To-Make-A-GIF (last visited May 24, 2022); Ex. 80, *How to make a video call*, WHATSAPP,

255. On information and belief, the Meta Companies contributorily infringe the ’682 Patent under 35 U.S.C. § 271(c) by importing, selling, and/or offering to sell within the United States the Meta Companies ’682 Accused Products (or components thereof) that constitute a material part of the claimed invention and are not staple articles of commerce suitable for substantial non-infringing use. For example, the hardware and/or software for using WebRTC is material, has no insubstantial non-infringing uses, and is known by the Meta Companies to be especially made or adapted for use that practices at least claim 1 of the ’682 Patent with respect to the Meta Companies ’682 Accused Products.

COUNT V
INFRINGEMENT OF U.S. PATENT NO. 7,436,980

256. VideoLabs incorporates by reference the foregoing paragraphs of this First Amended Complaint as if fully set forth herein.

257. VL IP is the assignee and lawful owner of all right, title, and interest in and to the ’980 Patent. The ’980 Patent is valid and enforceable.

258. On information and belief, the Meta Companies have infringed and continue to infringe the ’980 Patent in violation of 35 U.S.C. § 271(a), either literally or through the doctrine of equivalents, by using methods and/or taking steps that practice at least claim 1 of the ’980 Patent, including with respect to the Meta Companies ’980 Accused Products.

259. On information and belief, the Meta Companies use the Meta Companies ’980 Accused Products for its own business purposes. In addition, the Meta Companies regularly conduct testing and troubleshooting of the Meta Companies ’980 Accused Products. Further,

---

https://faq.whatsapp.com/android/voice-and-video-calls/how-to-make-a-video-call/?lang=en (last visited May 24, 2022); Ex. 81, *I get a message that my video is too long and it won’t send*, META, https://faq.whatsapp.com/general/i-get-a-message-that-my-video-is-too-long-and-it-wont-send/?lang=en (last visited May 24, 2022).

VideoLabs is informed and believes companies related to the Meta Companies (e.g., other subsidiaries) use the Meta Companies ’980 Accused Products.

260. On information and belief, the Meta Companies’ infringement through its use of object detection, described below, is exemplary of all of the Meta Companies’ infringement with respect to all the Meta Companies ’980 Accused Products.

261. The Meta Companies ’980 Accused Products directly infringe at least claim 1 of the ’980 Patent by, for example, detecting an object in an image of video by providing a spatio-temporal model for the object; measuring the object as a collection of components; determining a probability that the object is in the image, and comparing the probabilities for each image to a threshold for detecting the object (“Meta Companies Object Detection”).

262. The Meta Companies Object Detection meets every limitation of claim 1 of the ‘980 Patent, which recites:

> 1. A computer implemented method for object detection comprising:
>
> providing a spatio-temporal model for an object to be detected;
>
> providing a video comprising a plurality of images including the object;
>
> measuring the object as a collection of components in each image;
>
> determining a probability that the object is in each image; and
>
> detecting the object in any image upon comparing the probabilities for each image to a threshold for detecting the object.

263. The Meta Companies Object Detection provides a computer implemented method for object detection.

> We built Detectron2 to meet the research needs of Facebook AI and to provide the foundation for object detection in production use cases at Facebook. We are now using Detectron2 to rapidly design and train the next-generation pose detection models that power Smart Camera, the AI camera system in Facebook's Portal video-calling devices. By relying on Detectron2 as the unified library for object detection across research and production use cases, we are able to rapidly move research ideas into production models that are deployed at scale.

Source: Ex. 59, *Detectron2: A PyTorch-based modular object detection library*, META AI (Oct. 10, 2019), https://ai.facebook.com/blog/-detectron2-a-pytorch-based-modular-object-detection-library-/ (last visited May 24, 2022).

264. The Meta Companies Object Detection provides a spatio-temporal model for an object to be detected. For example, the Meta Companies Object Detection supports myriad spatio-temporal models to perform object detection functions.

> New models and features: Detectron2 includes all the models that were available in the original Detectron, such as Faster R-CNN, Mask R-CNN, RetinaNet, and DensePose. It also features several new models, including Cascade R-CNN, Panoptic FPN, and TensorMask, and we will continue to add more algorithms. We've also added features such as synchronous Batch Norm and support for new datasets like LVIS.

Source: Ex. 59, *Detectron2: A PyTorch-based modular object detection library*, META AI (Oct. 10, 2019), https://ai.facebook.com/blog/-detectron2-a-pytorch-based-modular-object-detection-library-/ (last visited May 24, 2022).

COCO Object Detection Baselines

Faster R-CNN:

Name	lr sched	train time (s/iter)	inference time (s/im)	train mem (GB)	box AP	model id	download
R50-C4	1x	0.551	0.102	4.8	35.7	137257644	model | metrics
R50-DC5	1x	0.380	0.068	5.0	37.3	137847829	model | metrics
R50-FPN	1x	0.210	0.038	3.0	37.9	137257794	model | metrics
R50-C4	3x	0.543	0.104	4.8	38.4	137849393	model | metrics
R50-DC5	3x	0.378	0.070	5.0	39.0	137849425	model | metrics
R50-FPN	3x	0.209	0.038	3.0	40.2	137849458	model | metrics
R101-C4	3x	0.619	0.139	5.9	41.1	138204752	model | metrics
R101-DC5	3x	0.452	0.086	6.1	40.6	138204841	model | metrics
R101-FPN	3x	0.286	0.051	4.1	42.0	137851257	model | metrics
X101-FPN	3x	0.638	0.098	6.7	43.0	139173657	model | metrics

RetinaNet:

Name	lr sched	train time (s/iter)	inference time (s/im)	train mem (GB)	box AP	model id	download
R50	1x	0.205	0.041	4.1	37.4	190397773	model | metrics
R50	3x	0.205	0.041	4.1	38.7	190397829	model | metrics
R101	3x	0.291	0.054	5.2	40.4	190397697	model | metrics

RPN & Fast R-CNN:

Name	lr sched	train time (s/iter)	inference time (s/im)	train mem (GB)	box AP	prop. AR	model id	download
RPN R50-C4	1x	0.130	0.034	1.5		51.6	137258005	model | metrics
RPN R50-FPN	1x	0.186	0.032	2.7		58.0	137258492	model | metrics
Fast R-CNN R50-FPN	1x	0.140	0.029	2.6	37.8		137635226	model | metrics

Source: Ex. 84, *Detectron2 Model Zoo and Baselines*, https://github.com/facebookresearch/detectron2/blob/main/MODEL_ZOO.md (last visited May 24, 2020).

265. The Meta Companies Object Detection provides a video comprising a plurality of images including the object. Video comprising a plurality of images is provided to Detectron2 for object detection through a Python command line interface, where the video file reference is passed as a command line argument.

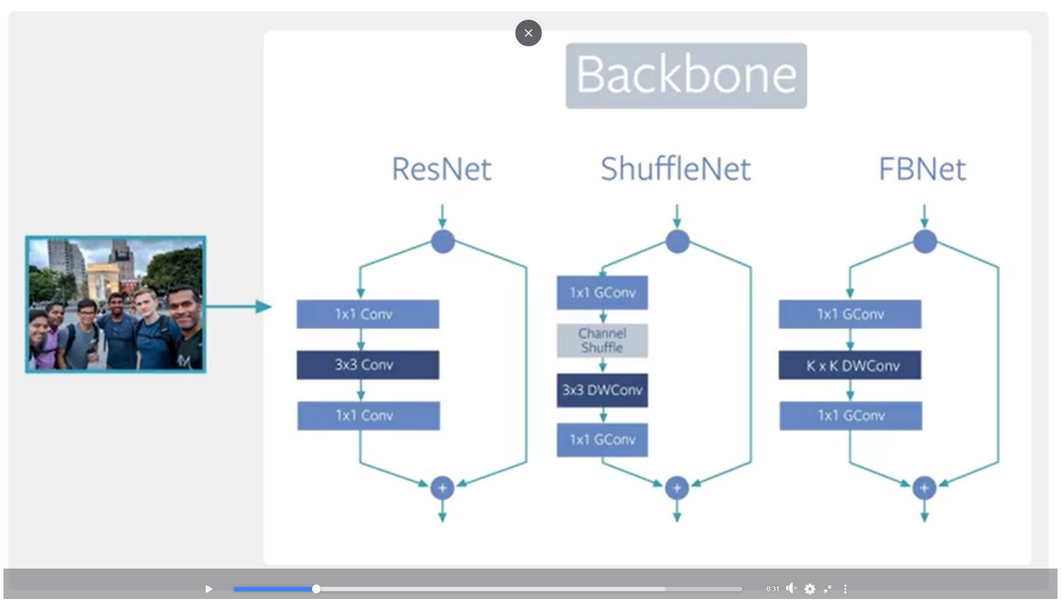
```
cd demo/
python demo.py --config-file ../configs/COCO-InstanceSegmentation/mask_rcnn_R_50_FPN_3x.yaml \
  --input input1.jpg input2.jpg \
  [--other-options]
  --opts MODEL.WEIGHTS detectron2://COCO-InstanceSegmentation/mask_rcnn_R_50_FPN_3x/137849600/mo
```

For details of the command line arguments, see `demo.py -h` or look at its source code to understand its behavior. Some common arguments are:

- To run **on your webcam**, replace `--input files` with `--webcam`.
- To run **on a video**, replace `--input files` with `--video-input video.mp4`.
- To run **on cpu**, add `MODEL.DEVICE cpu` after `--opts`.
- To save outputs to a directory (for images) or a file (for webcam or video), use `--output`.

Source: Ex. 62, *Getting Started with Detectron2*, DETECTRON2, available at https://detectron2.readthedocs.io/en/latest/tutorials/getting_started.html (last visited May 24, 2020).

266. The Meta Companies Object Detection measures the object as a collection of components in each image. The object detection models employed in Detectron2 and D2Go use convolutional approaches to measure each object in the image as a collection of components.



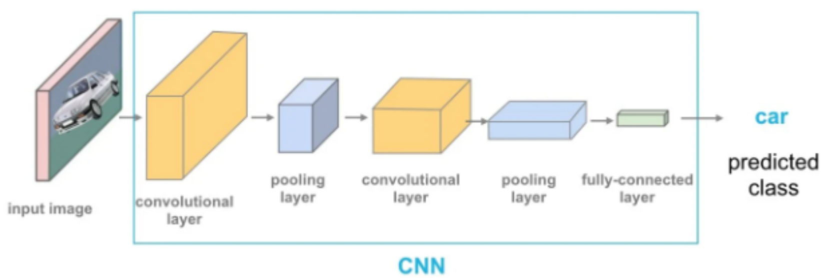
Source: Ex. 59, *Detectron2: A PyTorch-based modular object detection library*, META AI (Oct. 10, 2019), https://ai.facebook.com/blog/-detectron2-a-pytorch-based-modular-object-detection-library-/ (last visited May 24, 2022).

267. These convolutional approaches process image data using a layered approach. Convolutional neural networks measure an image as made of constituent layers.

The Convolutional Neural Network Architecture consists of three main layers:

1. **Convolutional layer:** This layer helps to abstract the input image as a feature map via the use of filters and kernels.
2. **Pooling layer:** This layer helps to downsample feature maps by summarizing the presence of features in patches of the feature map.
3. **Fully connected layer:** Fully connected layers connect every neuron in one layer to every neuron in another layer.

Combining the layers of a CNN enables the designed neural network to learn how to identify and recognize the object of interest in an image. Simple Convolutional Neural Networks are built for image classification and object detection with a single object in the image.



Concept of the CNN architecture: How a convolutional neural network works.

Source: Ex. 63, E. Odemakinde, *Mask R-CNN: A Beginner's Guide*, VISO.AI, available at https://viso.ai/deep-learning/mask-r-cnn/#:~:text=Mask%20R%2DCNN%20is%20a,segmentation%20mask%20for%20each%20instance (last accessed May 24, 2022).

268. Another way the Meta Companies Object Detection processes image data is by using masks. Masks use spatial segmentation maps to measure and detect the objects in an image.

Figure 1. Selected output of *TensorMask*, our proposed framework for performing *dense sliding-window instance segmentation*. We treat dense instance segmentation as a prediction task over *structured* 4D tensors. In addition to obtaining competitive quantitative results, TensorMask achieves results that are *qualitatively* reasonable. Observe that both small and large objects are well delineated and more critically *overlapping objects are properly handled*.

Source: Ex. 64, X. Chen, et al., *TensorMask: A Foundation for Dense Object Segmentation*, FACEBOOK AI RESEARCH (FAIR), at Fig. 1 (Aug. 27, 2019), available at https://arxiv.org/pdf/1903.12174.pdf (last accessed May 24, 2022).

Our main insight is that the core concepts for defining dense mask representations, as well as effective realizations of these concepts in neural networks, are both lacking. Unlike bounding boxes, which have a fixed, low-dimensional representation regardless of scale, segmentation masks can benefit from richer, more structured representations. For example, each mask is itself a 2D spatial map, and masks for larger objects can benefit from the use of larger spatial maps. Developing effective representations for dense masks is a key step toward enabling dense instance segmentation.

Source: Ex. 64, X. Chen, et al., *TensorMask: A Foundation for Dense Object Segmentation*, FACEBOOK AI RESEARCH (FAIR), at Fig. 1 (Aug. 27, 2019), available at https://arxiv.org/pdf/1903.12174.pdf (last accessed May 24, 2022).

269. The Meta Companies Object Detection determines a probability that the object is in each image. On information and belief, the Meta Companies' Detectron2 and D2Go perform a series of object detection and object tracking functions in order to determine the probability that an object is detected in each image.

Source: Ex. 65, G. Tanner, *D2Go – Use Detectron2 on mobile devices* (Mar. 20, 2021), available at https://gilberttanner.com/blog/d2go-use-detectron2-on-mobile-devices (last accessed May 24, 2022).

270. The Meta Companies' Detectron2 and D2Go maintain a list of objects detected within each image as separate "instances." The Meta Companies specify the probability of each instance being in the image by a "score" field, which indicates or reflects the probability that the object is in the image. These probability scores are returned by the "inference_single_image" function.

class `detectron2.structures.Instances`(*image_size: Tuple[int, int], **kwargs: Any*)

Bases: `object`

This class represents a list of instances in an image. It stores the attributes of instances (e.g., boxes, masks, labels, scores) as "fields". All fields must have the same `__len__` which is the number of instances.

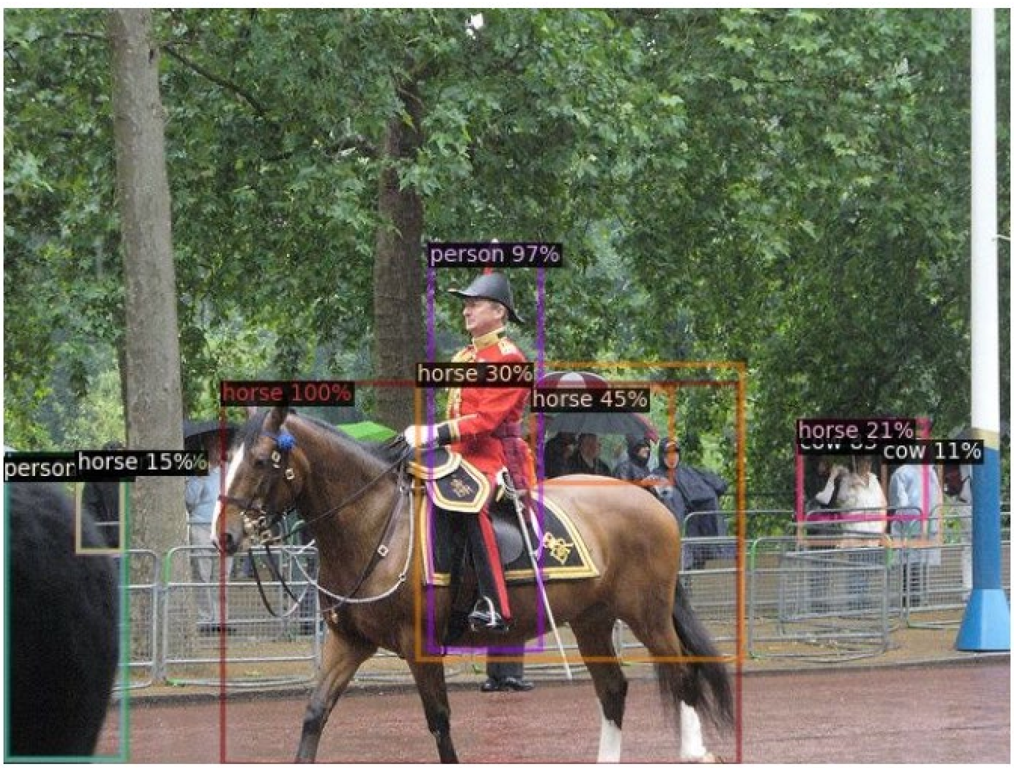
Source: Ex. 66, *detectron2.structures*, DETECTRON2, available at https://detectron2.readthedocs.io/en/latest/modules/structures.html (last accessed May 24, 2022).

`inference_single_image`(*anchors: List[detectron2.structures.Boxes], box_cls: List[torch.Tensor], box_delta: List[torch.Tensor], image_size: Tuple[int, int]*)

Single-image inference. Return bounding-box detection results by thresholding on scores and applying non-maximum suppression (NMS).

Source: Ex. 66, *detectron2.structures*, DETECTRON2, available at https://detectron2.readthedocs.io/en/latest/modules/structures.html (last accessed May 24, 2022).

271. The Meta Companies Object Detection detects the object in any image upon comparing the probabilities for each image to a threshold for detecting the object. The Meta Companies' Detectron2 and D2Go perform a series of object detection and object tracking functions in order to determine the probability that an object is detected in each image.



Source: Ex. 65, G. Tanner, *D2Go – Use Detectron2 on mobile devices* (Mar. 20, 2021), available at https://gilberttanner.com/blog/d2go-use-detectron2-on-mobile-devices (last accessed May 24, 2022).

272. The probabilities that an object is in each image are compared to a threshold for detecting the object. Specifically, "inference_single_image" is an object detection and tracking function which can be configured to filter out detected objects that are below a probability threshold, "self.test_score_thresh." Detected objects that fall below this threshold probability value are filtered out of the object detection results for the images.

inference_single_image(*anchors: List[detectron2.structures.Boxes], box_cls: List[torch.Tensor], box_delta: List[torch.Tensor], image_size: Tuple[int, int]*)

Single-image inference. Return bounding-box detection results by thresholding on scores and applying non-maximum suppression (NMS).

Source: Ex. 66, *detectron2.structures*, DETECTRON2, available at https://detectron2.readthedocs.io/en/latest/modules/structures.html (last accessed May 24, 2022).

```
def inference_single_image(
    self,
    anchors: List[Boxes],
    box_cls: List[Tensor],
    box_delta: List[Tensor],
    image_size: Tuple[int, int],
):
    """
    Single-image inference. Return bounding-box detection results by thresholding
    on scores and applying non-maximum suppression (NMS).
```

Source: Ex. 67, *Source code for detectron2.modeling.meta_arch.retinanet*, DETECTRON2, available at https://detectron2.readthedocs.io/en/v0.5/_modules/detectron2/modeling/meta_arch/retinanet.html (last accessed May 24, 2022).

```
# Iterate over every feature level
for box_cls_i, box_reg_i, anchors_i in zip(box_cls, box_delta, anchors):
    # (HxWxAxK,)
    predicted_prob = box_cls_i.flatten().sigmoid_()

    # Apply two filtering below to make NMS faster.
    # 1. Keep boxes with confidence score higher than threshold
    keep_idxs = predicted_prob > self.test_score_thresh
    predicted_prob = predicted_prob[keep_idxs]
    topk_idxs = nonzero_tuple(keep_idxs)[0]

    # 2. Keep top k top scoring boxes only
    num_topk = min(self.test_topk_candidates, topk_idxs.size(0))
    # torch.sort is actually faster than .topk (at least on GPUs)
    predicted_prob, idxs = predicted_prob.sort(descending=True)
    predicted_prob = predicted_prob[:num_topk]
    topk_idxs = topk_idxs[idxs[:num_topk]]
```

Source: Ex. 67, *Source code for detectron2.modeling.meta_arch.retinanet*, DETECTRON2, available at https://detectron2.readthedocs.io/en/v0.5/_modules/detectron2/modeling/meta_arch/retinanet.html (last accessed May 24, 2022).

273. On information and belief, to the extent applicable, VideoLabs has complied with 35 U.S.C. § 287(a) with respect to the '980 Patent.

274. VideoLabs representatives reached out to Meta representatives at least as early as October 2, 2019 regarding VideoLabs' platform and to gauge the Meta Companies' interest in joining as a partner or member. In October, 2019, VideoLabs representatives presented VideoLabs' platform to representatives of the Meta Companies. VideoLabs representatives and Meta representatives spoke between December and February, and met in person on January 30, 2020, during which VideoLabs presented them with an updated proposal. On June 3, 2020, VideoLabs representatives reached out again and stated that "With regards to Facebook, and based on our analysis so far, we have determined that VideoLabs' current patent portfolio is relevant to the majority of Facebook's annual revenue in some way and we have 17 unique claim charts completed or in development (and other evidence of use) so far related to Facebook Messenger,

Facebook Live, Oculus, Instagram, Facebook Workplace, Portal, Facebook Ads, and backend infrastructure" and asked them to engage in "good faith licensing discussions" under a proposed NDA. VideoLabs representatives sent another updated proposal on March 23, 2020. VideoLabs representatives reached out again on June 26, 2020, March 16, 2021, and May 4, 2021 to "request[] good faith licensing negotiations" regarding the number of patents relevant to Facebook and other Meta services. On June 26, 2020, VideoLabs confirmed in an email that representatives of the Meta Companies said on a recent call that the Meta Companies were not interested in good faith licensing discussions. To date, months later, the Meta Companies have not reengaged with VideoLabs.

275. The Meta Companies of course know how their products operate, and on information and belief, they investigated the '980 Patent and their infringement of the Meta Companies '980 Accused Products. The Meta Companies have been given further notice of the '980 Patent and their infringement of the '980 Patent through VideoLabs' May 23, 2022 letter, VideoLabs' May 24, 2022 Complaint, and the filing of this First Amended Complaint. On information and belief, the Meta Companies are either knowingly infringing the '980 Patent or are willfully blind to its infringement, and continue to act in wanton disregard of VideoLabs' patent rights.

276. Despite becoming aware of or willfully blinding itself to its infringement of the '980 Patent, the Meta Companies have nonetheless continued to engage in and have escalated their infringing activities by continuing to develop, advertise, make available, and use the infringing functionalities of the Meta Companies '980 Accused Products. On information and belief, the Meta Companies have made no attempts to design around the '980 Patent or otherwise stop their infringing behavior.

277. The Meta Companies' infringement of the '980 Patent therefore has been and remains willful.

278. The Meta Companies also indirectly infringe the '980 Patent by inducing others to infringe and contributing to the infringement of others, including other Defendants and third-party users of the Meta Companies '980 Accused Products in this District and throughout the United States. As described above, on information and belief, the Meta Companies have known about the family of patents including the '980 Patent since at least January 30, 2020, and have known about the '980 Patent since May 23, 2021.

279. On information and belief, the Meta Companies have actively induced the infringement of the '980 Patent under 35 U.S.C. § 271(b) by actively inducing third parties in the United States to use methods and/or take steps that practice at least claim 1 of the '980 Patent with respect to the Meta Companies '980 Accused Products. The Meta Companies knew or were willfully blind to the fact that their conduct would induce these third parties to act in a manner that infringes the '980 Patent in violation of 35 U.S.C. § 271(a).

280. The Meta Companies actively encouraged and continue to actively encourage third parties to directly infringe the '980 Patent by, for example, marketing and offering the Meta Companies '980 Accused Products to consumers; working with consumers to implement, install and/or operate the Meta Companies '980 Accused Products; fully supporting and managing consumers' continuing use of the Meta Companies '980 Accused Products; and providing technical assistance to consumers during their continued use of the Meta Companies '980 Accused Products.[103]

---

[103] *See e.g.,* Ex. 71, *Using Facebook-Your Photos and Videos-Videos*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/1738143323068602 (last visited May 24,

281. The Meta Companies induce third parties to infringe the ’980 Patent at least by encouraging them to use Meta Companies Object Detection, which constitutes infringement of the ’980 Patent. For example, the Meta Companies advertise and promote Meta Companies Object Detection on its website. The Meta Companies advertise that Meta Companies Object Detection facilitates “flexible and extensible” object detection, “fast training on single or multiple GPU servers,” “includes high-quality implementations of state-of-the-art object detection algorithms,” and facilitates “rapid[]” and “at scale” production models.[104] In response, consumers acquire, develop, configure, and operate the Facebook Object Detection library in an infringing manner.

282. On information and belief, the Meta Companies contributorily infringe the ’980 Patent under 35 U.S.C. § 271(c) by importing, selling, and/or offering to sell within the United

---

2022); Ex. 72, *I can’t add a video on Facebook*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/215726848451641 (last visited May 24, 2022); Ex. 73, *Audio and Video Calling*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/messenger-app/1673374996287506 (last visited May 24, 2022); Ex. 74, *How do I video chat with someone or a group in Messenger?*, FACEBOOK HELP CENTER (2022), https://www.facebook.com/help/messenger-app/1414800065460231 (last visited May 24, 2022); Ex. 75, *Instagram Features-Videos*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/381435875695118 (last visited May 24, 2022); Ex. 76, *How do I share a video on Instagram?*, INSTAGRAM HELP CENTER (2022), https://help.instagram.com/456185931138729 (last visited May 24, 2022); Ex. 77, *How do I share my video on Instagram to my Facebook Page?*, INSTAGRAM HELP CENTER (2022), https://www.facebook.com/help/instagram/486878428409681 (last visited May 24, 2022); Ex. 78, *Oculus Support*, https://www.youtube.com/channel/UCxLTeQQFvA5_eb0VkrLIq6w (last visited May 24, 2022); Ex. 79, *Viewing your own 360 videos on your Gear VR*; META, https://support.giphy.com/hc/en-us/articles/360019674452-How-To-Make-A-GIF (last visited May 24, 2022); Ex. 80, *How to make a video call*, WHATSAPP, https://faq.whatsapp.com/android/voice-and-video-calls/how-to-make-a-video-call/?lang=en (last visited May 24, 2022); Ex. 81, *I get a message that my video is too long and it won’t send*, META, https://faq.whatsapp.com/general/i-get-a-message-that-my-video-is-too-long-and-it-wont-send/?lang=en (last visited May 24, 2022).

[104] *See* Ex. 59, *Detectron2: A PyTorch-based modular object detection library*, META AI (Oct. 10, 2019), https://ai.facebook.com/blog/-detectron2-a-pytorch-based-modular-object-detection-library-/ (last visited at May 24, 2022).

States the Meta Companies ’980 Accused Products (or components thereof) that constitute a material part of the claimed invention and are not staple articles of commerce suitable for substantial non-infringing use. For example, the hardware and/or software for using object detection is material, has no insubstantial non-infringing uses, and is known by the Meta Companies to be especially made or adapted for use that practices at least claim 1 of the ’980 Patent with respect to the Meta Companies ’980 Accused Products.

PRAYER FOR RELIEF

WHEREFORE, VideoLabs prays for judgment as follows:

a) That the Meta Companies directly and/or indirectly infringe the ’878, ’238, ’059, ’682, and ’980 Patents;

b) That such infringement is willful;

c) That the Meta Companies and their respective officers, directors, agents, partners, servants, employees, attorneys, licensees, successors, and assigns, and those in active concert or participation with any of them, be permanently enjoined from engaging in infringing activities with respect to the ’878, ’238, ’059, ’682, and ’980 Patents;

d) In the alternative, in the event injunctive relief is not granted as requested by VideoLabs, an award of a mandatory future royalty payable on each future product sold by the Meta Companies that is found to infringe one or more claims of the ’878, ’238, ’059, ’682, and ’980 Patents, and on all future products which are not colorably different from products found to infringe;

e) That the Meta Companies be required to pay VideoLabs’ damages in an amount adequate to compensate VideoLabs for their infringement, but in no event less than a reasonable royalty under 35 U.S.C. § 284, including supplemental damages for any continuing post-verdict infringement up until entry of judgment and beyond, with accounting, as needed;

f) That VideoLabs be awarded all statutory and actual damages to which it is entitled, including the profits reaped by the Meta Companies through its illegal conduct, and prejudgment and post-judgment interest;

g) That VideoLabs be awarded enhanced damages, up to and including trebling of the damages awarded to VideoLabs;

h) That VideoLabs be awarded recovery of the costs of this suit, including reasonable attorneys' fees; and

i) That VideoLabs be awarded such other and further relief as this Court deems just and proper.

DEMAND FOR JURY TRIAL

283. VideoLabs hereby demands a jury trial on its claims for patent infringement and any and all issues triable of right before a jury.

Dated: June 6, 2024

Respectfully submitted,

/s/ Brian E. Farnan

Brian E. Farnan (Bar No. 4089)
Michael J. Farnan (Bar No. 5165)
919 N. Market St., 12th Floor
FARNAN LLP
Wilmington, DE 19801
Telephone: (302) 777-0300
Fax: (302) 777-0301
bfarnan@farnanlaw.com
mfarnan@farnanlaw.com

Christine E. Lehman
clehman@reichmanjorgensen.com
REICHMAN JORGENSEN LEHMAN & FELDBERG LLP
1909 K Street NW, Suite 800
Washington, DC 20006
Telephone: (202) 894-7325
Fax: (650) 623-1449

Courtland L. Reichman
creichman@reichmanjorgensen.com

Jaime F. Cardenas-Navia
jcardenas-navia@reichmanjorgensen.com
Michael Matulewicz-Crowley

mmatulewicz-crowley@reichmanjorgensen.com
REICHMAN JORGENSEN LEHMAN &
FELDBERG LLP
750 Third Avenue, Suite 2400
New York, NY 10017
Telephone: (212) 381-1965
Facsimile: (650) 623-1449

REICHMAN JORGENSEN LEHMAN &
FELDBERG LLP
100 Marine Parkway, Suite 300
Redwood Shores, California 94065
Telephone: (650) 623-1401
Fax: (650) 623-1449

Taylor N. Mauze
tmauze@reichmanjorgensen.com
REICHMAN JORGENSEN LEHMAN &
FELDBERG LLP
7500 Rialto Blvd., Ste. 1-250
Austin, Texas 78735
Telephone: (650) 623-1401
Facsimile: (650) 623-1449

Attorneys for Plaintiffs VideoLabs, Inc. and VL Collective IP LLC