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

Attorneys for Plaintiff
REALTIME ADAPTIVE STREAMING LLC

UNITED STATES DISTRICT COURT

CENTRAL DISTRICT OF CALIFORNIA

WESTERN DIVISION

REALTIME ADAPTIVE STREAMING LLC,

Plaintiff,

v.

NETFLIX, INC. and

NETFLIX STREAMING SERVICES, INC.,

Defendants.

Case No. 2:19-cv-06359-GW-JC

FIRST AMENDED COMPLAINT FOR PATENT INFRINGEMENT

JURY TRIAL DEMANDED

This is an action for patent infringement arising under the Patent Laws of the United States of America, 35 U.S.C. § 1 et seq. in which Plaintiff Realtime Adaptive Streaming LLC ("Plaintiff" or "Realtime") makes the following allegations against Defendants Netflix, Inc. and Netflix Streaming Services, Inc. (collectively "Netflix" or "Defendants")

PARTIES

1. Realtime is a Texas limited liability company. Realtime has a place of business at 66 Palmer Avenue, Suite 27, Bronxville, NY 10708. Realtime has researched and developed specific solutions for data compression, including, for example, those that increase the speeds at which data can be stored and accessed. As recognition of its innovations rooted in this technological field, Realtime holds multiple United States patents and pending patent applications.

2. Defendant Netflix, Inc. is a Delaware corporation, with its principal place of business at 100 Winchester Circle, Los Gatos, California 95032. Netflix, Inc. may be served with process by serving its registered agent, The Corporation Trust Company at the Corporation Trust Center, 1209 Orange Street, Wilmington, Delaware 19801.

3. Defendant Netflix Streaming Services, Inc. is a Delaware corporation, with its principal place of business at 100 Winchester Circle, Los Gatos, California 95032. Netflix Streaming Services, Inc. may be served with process by serving its registered agent, The Corporation Trust Company at the Corporation Trust Center, 1209 Orange Street, Wilmington, Delaware 19801.

4. Defendants have regular and established place of business in this District, including, e.g., in Los Angeles, CA. *See, e.g.,* https://jobs.netflix.com/locations/los-angeles-california

JURISDICTION AND VENUE

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

6. This Court has personal jurisdiction over Defendants in this action because Defendants have committed acts within this District giving rise to this action and has established minimum contacts with this forum such that the exercise of jurisdiction over Defendants would not offend traditional notions of fair play and substantial justice. Defendants have also committed and continue to commit acts of infringement in this District by, among other things, offering to sell and selling products and/or services that infringe the asserted patents.

7. Venue is proper in this district, *e.g.*, under 28 U.S.C. § 1400(b). Defendants has committed acts of direct and indirect infringement in this District, and has a regular and established place of business in this District, including, e.g., in Los Angeles.

THE PATENT-IN-SUIT

8. This action arises under 35 U.S.C. § 271 for Netflix's infringement of Realtime's United States Patent Nos. RE46,777 (the "'777 patent").

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

COUNT I

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

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

11. On information and belief, Netflix has made, used, offered for sale, sold and/or imported into the United States Netflix products that infringe the '777 patent, and continues to do so. By way of illustrative example, these infringing products include, without limitation, Netflix's streaming video service; Netflix's video encoders or codecs including x265 and libvpx (VP9); and Netflix's mobile encoders, encoders or codecs including VP9, VP9-Mobile, H.265/HEVC and all versions and variations thereof since the issuance of the '777 patent ("Accused Instrumentalities").

12. For example, in an entry of Netflix's "The Netflix Tech Blog," a test comparing several different encoders was described: "H.264/AVC is a very widely-used video compression standard on the Internet, with ubiquitous decoder support on web browsers, TVs, mobile devices, and other consumer devices. x264 is the most established open-source software encoder for H.264/AVC. HEVC is the successor to H.264/AVC and results reported from standardization showed about 50% bitrate savings for the same quality compared to H.264/AVC. **x265** is an open-source HEVC encoder, originally ported from the x264 codebase. Concurrent to HEVC, Google developed **VP9** as a royalty-free video compression format and released **libvpx** as an open-source software library for encoding VP9." The results of that test were as follows: "Here's a snapshot: x265 and libvpx demonstrate superior compression performance compared to x264, with bitrate savings reaching up to 50% especially at the higher resolutions. x265 outperforms libvpx for almost all resolutions and quality metrics, but the performance gap narrows (or even reverses) at 1080p." *See* https://medium.com/netflix-techblog/a-large-scale-comparison-of-x264-x265-and-libvpx-a-sneak-peek-2e81e88f8b0f.

13. In addition, another article on the website "Arstechnica" mentions that "There's also the matter of hardware decoding support for 10-bit HEVC, the 4K codec used by Netflix and other streaming services." *See* https://arstechnica.com/gadgets/2016/11/netflix-4k-streaming-pc-kaby-lake-cpu-

windows-10-edge-browser/.

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

15. The Accused Instrumentalities reduce temporal redundancy by block based motion compensated prediction in order to establish a prediction error signal. For example, clause 8.5.3 Decoding process for prediction units in inter prediction mode and the subclauses thereof of the HEVC Spec describe the block based motion compensation techniques used in the decoding process. *See also, e.g.,* IEEE HEVC at 1651-1652 6) Motion compensation: Quarter-sample precision is used for the MVs, and 7-tap or 8-tap filters are used for interpolation of fractional-sample positions (compared to six-tap filtering of half-sample positions followed by linear interpolation for quarter-sample positions in H.264/MPEG-4 AVC). Similar to H.264/MPEG-4 AVC, multiple reference pictures are used. For each PB, either one or two motion vectors can be transmitted, resulting either in

unipredictive or bipredictive coding, respectively. As in H.264/MPEG-4 AVC, a scaling and offset operation may be applied to the prediction signal(s) in a manner known as weighted prediction.").

16. The Accused Instrumentalities perform quantization on samples of the prediction error signal or on coefficients resulting from a transformation of the prediction error signal into the frequency domain to obtain quantized values, representing quantized samples or quantized coefficients respectively. For example, the quantization parameter and the scaling (inverse quantization) are defined in definitions 3.112 (page 10) and 3.131 (page 11), respectively, the usage of the scaling process in the decoding being described in clause and 8.6 Scaling, transformation and array construction process prior to deblocking filter process of the HEVC Spec. *See also, e.g.,* IEEE HEVC at 1652 ("8) Quantization control: As in H.264/MPEG-4 AVC, uniform reconstruction quantization (URQ) is used in HEVC, with quantization scaling matrices supported for the various transform block sizes.").

17. The Accused Instrumentalities perform a method wherein the prediction error signal includes a plurality of subblocks each including a plurality of quantized values. For example, the quantized samples or transform coefficients from the subblock are scaled and transformed as described in above mentioned clause 8.6 of the HEVC Spec. *See also, e.g.,* IEEE HEVC at 1652 ("Prediction units and prediction blocks (PBs): The decision whether to code a picture area using interpicture or intrapicture prediction is made at the CU level. A PU partitioning structure has its root at the CU level. Depending on the basic prediction-type decision, the luma and chroma CBs can then be further split in size and predicted from luma and chroma prediction blocks (PBs). HEVC supports variable PB sizes from 64×64 down to 4×4 samples.").

18. The Accused Instrumentalities perform a method of calculating a first quantization efficiency for the quantized values of at least one subblock of the

plurality of subblocks; setting the quantized values of the at least one subblock to all zeroes; calculating a second quantization efficiency for the at least one subblock while all of the quantized values are zeroes; selecting which of the first and second quantization efficiencies is a higher efficiency; and selecting, for further proceeding, the at least one subblock with the quantized values prior to setting the quantized values of the at least one subblock to all zeroes if the first quantization efficiency is higher and selecting the at least one subblock with the quantized values set to zero, for further proceeding, if the second quantization efficiency is higher. For example, the bitstream resulting from the encoding as described in this last item of the claim contains all the relevant information as needed by the decoder for proper decoding. If the coefficients of the subblock are set to zero as a consequence of the efficiency calculation, the coded_sub_block_flag, as described in clause 7.4.9.11 Residual coding semantics, HEVC Spec, is set to 0, indicating that all the 16 coefficients of the coded sub block have been set to 0: "coded_sub_block_flag[xS][yS] specifies the following for the sub-block at location (xS, yS) within the current transform block, where a sub-block is a (4x4) array of 16 transform coefficient levels: – If coded_sub_block_flag[xS][yS] is equal to 0, the 16 transform coefficient levels of the sub-block at location (xS, yS) are inferred to be equal to 0."

19. When coded_sub_block_flag[xS][yS] has not been set equal to 0, the position in the array of non 0 coefficients can be determined as follows:

– Otherwise (coded_sub_block_flag[xS][yS] is equal to 1), the following applies:

– If (xS, yS) is equal to (0, 0) and (LastSignificantCoeffX, LastSignificantCoeffY) is not equal to (0, 0), at least one of the 16 sig_coeff_flag syntax elements is present for the sub-block at location (xS, yS) .

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

When coded_sub_block_flag[xS][yS] is not present, it is inferred as follows:

– If one or more of the following conditions are true, coded_sub_block_flag[xS][yS] is inferred to be equal to 1:

– (xS, yS) is equal to (0, 0)

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

– Otherwise, coded_sub_block_flag[xS][yS] is inferred to be equal to 0.

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

20. The infringement of the Accused Instrumentalities is also shown by way of considering the reference software (*see, e.g.,* https://hevc.hhi.fraunhofer.de/). Setting the flag RDOQ=true in the encoder configuration file enables rate-distortion-optimized quantization for transformed TUs. This feature is implemented in the HM reference software as function xRateDistOptQuant in file TComTrQuant.cpp. In the function xRateDistOptQuant, the efficiency for setting all quantized values to zero is calculated and stored in the variable d64BestCost. In the variable iBestLastIdxP1, a 0 is stored indicating that all values starting from the 0th position are set to zero. Afterwards, the efficiency for keeping quantized values unequal to zero is calculated and stored in the variable totalCost. The variable iBestLastIdxP1 is adjusted correspondingly to values unequal to 0. The two efficiencies d64BestCost and totalCost are compared, and selecting for further proceeding either quantized values, which are all set to zero or quantized values, which are not all set to zero. All values starting from the position defined by the variable iBestLastIdxP1 are set

to zero.

```
Double  d64BestCost         = 0;
Int     ui16CtxCbf          = 0;
Int     iBestLastIdxP1      = 0;
if( !pcCU->isIntra( uiAbsPartIdx ) && isLuma(compID) && pcCU->getTransformIdx( uiAbsPartIdx ) == 0 )
{
  ui16CtxCbf   = 0;
  d64BestCost  = d64BlockUncodedCost + xGetICost( m_pcEstBitsSbac->blockRootCbpBits[ ui16CtxCbf ][ 0 ] );
  d64BaseCost += xGetICost( m_pcEstBitsSbac->blockRootCbpBits[ ui16CtxCbf ][ 1 ] );
}
else
{
  ui16CtxCbf   = pcCU->getCtxQtCbf( rTu, channelType );
  ui16CtxCbf  += getCBFContextOffset(compID);
  d64BestCost  = d64BlockUncodedCost + xGetICost( m_pcEstBitsSbac->blockCbpBits[ ui16CtxCbf ][ 0 ] );
  d64BaseCost += xGetICost( m_pcEstBitsSbac->blockCbpBits[ ui16CtxCbf ][ 1 ] );
}
```

21. Calculation of the efficiency for setting all quantized values to zero and storing the result in the variable d64BestCost:

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

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

```
Bool bFoundLast = false;
for (Int iCGScanPos = iCGLastScanPos; iCGScanPos >= 0; iCGScanPos--)
{
  UInt uiCGBlkPos = codingParameters.scanCG[ iCGScanPos ];

  d64BaseCost -= pdCostCoeffGroupSig [ iCGScanPos ];
  if (uiSigCoeffGroupFlag[ uiCGBlkPos ])
  {
    for (Int iScanPosinCG = uiCGSize-1; iScanPosinCG >= 0; iScanPosinCG--)
    {
      iScanPos = iCGScanPos*uiCGSize + iScanPosinCG;

      if (iScanPos > iLastScanPos) continue;
      UInt   uiBlkPos     = codingParameters.scan[iScanPos];

      if( piDstCoeff[ uiBlkPos ] )
      {
        UInt   uiPosY       = uiBlkPos >> uiLog2BlockWidth;
        UInt   uiPosX       = uiBlkPos - ( uiPosY << uiLog2BlockWidth );

        Double d64CostLast= codingParameters.scanType == SCAN_VER ? xGetRateLast( uiPosY, uiPosX, compID ) :
                                                                    xGetRateLast( uiPosX, uiPosY, compID );
        Double totalCost = d64BaseCost + d64CostLast - pdCostSig[ iScanPos ];
```

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

23. Comparing the two efficiencies d64BestCost and totalCost:

```
if( totalCost < d64BestCost )
{
  iBestLastIdxP1  = iScanPos + 1;
  d64BestCost     = totalCost;
}
```

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

24. Selecting for further proceeding either quantized values, which are all

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

set to zero or quantized values, which are not all set to zero:

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

25. Therefore, from at least the above, Netflix has directly infringed and continues to infringe the '777 patent, for example, through its own use and testing of the Accused Instrumentalities, which when used, practices the system claimed by Claim 1 of the '777 patent, namely, a method for coding a video signal using hybrid coding, comprising: reducing temporal redundancy by block based motion compensated prediction in order to establish a prediction error signal; performing quantization on samples of the prediction error signal or on coefficients resulting from a transformation of the prediction error signal into the frequency domain to obtain quantized values, representing quantized samples or quantized coefficients respectively, wherein the prediction error signal includes a plurality of subblocks each including a plurality of quantized values; calculating a first quantization efficiency for the quantized values of at least one subblock of the plurality of subblocks; setting the quantized values of the at least one subblock to all zeroes; calculating a second quantization efficiency for the at least one subblock while all of the quantized values are zeroes; selecting which of the first and second quantization efficiencies is a higher efficiency; and selecting, for further proceeding, the at least one subblock with the quantized values prior to setting the quantized values of the at least one subblock to all zeroes if the first quantization efficiency is higher and selecting the at least one subblock with the quantized values set to zero, for further proceeding, if the second quantization efficiency is higher. Upon information and belief, Netflix uses the Accused Instrumentalities to practice infringing methods for its own internal non-testing business purposes,

while testing the Accused Instrumentalities, and while providing technical support and repair services for the Accused Instrumentalities to their customers.

26. On information and belief, Netflix also directly infringes and continues to infringe other claims of the '777 patent.

27. On information and belief, all of the Accused Instrumentalities perform the claimed methods in substantially the same way, e.g., in the manner specified in the HEVC (or H.265) standard.

28. On information and belief, use of the Accused Instrumentalities in their ordinary and customary fashion results in infringement of the methods and/or systems claimed by the '777 patent.

29. On information and belief, Netflix has had knowledge of the '777 patent since at least the filing of this Complaint or shortly thereafter, and on information and belief, Netflix knew of the '777 patent and knew of its infringement, including by way of this lawsuit. By the time of trial, Netflix will have known and intended (since receiving such notice) that its continued actions would actively induce and contribute to the infringement of the claims of the '777 patent.

30. Upon information and belief, Netflix's affirmative acts of making, using, and selling the Accused Instrumentalities, and providing implementation services and technical support to users of the Accused Instrumentalities, including, e.g., through training, demonstrations, brochures, installation and user guides, have induced and continue to induce users of the Accused Instrumentalities to use them in their normal and customary way to infringe the '777 patent by practicing a method for coding a video signal using hybrid coding, comprising: reducing temporal redundancy by block based motion compensated prediction in order to establish a prediction error signal; performing quantization on samples of the prediction error signal or on coefficients resulting from a transformation of the prediction error signal into the frequency domain to obtain quantized values,

representing quantized samples or quantized coefficients respectively, wherein the prediction error signal includes a plurality of subblocks each including a plurality of quantized values; calculating a first quantization efficiency for the quantized values of at least one subblock of the plurality of subblocks; setting the quantized values of the at least one subblock to all zeroes; calculating a second quantization efficiency for the at least one subblock while all of the quantized values are zeroes; selecting which of the first and second quantization efficiencies is a higher efficiency; and selecting, for further proceeding, the at least one subblock with the quantized values prior to setting the quantized values of the at least one subblock to all zeroes if the first quantization efficiency is higher and selecting the at least one subblock with the quantized values set to zero, for further proceeding, if the second quantization efficiency is higher. For example, Netflix adopted HEVC (or H.265) as its video codec in its products/services, such as its streaming services, and uses HEVC (or H.265) as an encoder, encode or codec. For similar reasons, Netflix also induces its customers to use the Accused Instrumentalities to infringe other claims of the '777 patent. Netflix specifically intended and was aware that these normal and customary activities would infringe the '777 patent. Netflix performed the acts that constitute induced infringement, and would induce actual infringement, with the knowledge of the '777 patent and with the knowledge, or willful blindness to the probability, that the induced acts would constitute infringement. On information and belief, Netflix engaged in such inducement to promote the sales of the Accused Instrumentalities. Accordingly, Netflix has induced and continue to induce users of the Accused Instrumentalities to use the Accused Instrumentalities in their ordinary and customary way to infringe the '777 patent, knowing that such use constitutes infringement of the '777 patent. Accordingly, Netflix has been, and currently is, inducing infringement of the '777 patent, in violation of 35 U.S.C. § 271(b).

31. Netflix has also infringed, and continues to infringe, claims of the

'777 patent by offering to commercially distribute, commercially distributing, making, and/or importing the Accused Instrumentalities, which are used in practicing the process, or using the systems, of the '777 patent, and constitute a material part of the invention. Netflix knows the components in the Accused Instrumentalities to be especially made or especially adapted for use in infringement of the '777 patent, not a staple article, and not a commodity of commerce suitable for substantial noninfringing use. Accordingly, Netflix has been, and currently is, contributorily infringing the '777 patent, in violation of 35 U.S.C. § 271(c).

32. By making, using, offering for sale, selling and/or importing into the United States the Accused Instrumentalities, and touting the benefits of using the Accused Instrumentalities' compression features, Netflix has injured Realtime and is liable to Realtime for infringement of the '777 patent pursuant to 35 U.S.C. § 271.

33. As a result of Netflix's infringement of the '777 patent, Plaintiff Realtime is entitled to monetary damages in an amount adequate to compensate for Netflix's infringement, but in no event less than a reasonable royalty for the use made of the invention by Netflix, together with interest and costs as fixed by the Court.

PRAYER FOR RELIEF

WHEREFORE, Plaintiff Realtime respectfully requests that this Court enter:

a. A judgment in favor of Plaintiff that Netflix has infringed, literally and/or under the doctrine of equivalents, the '777 patents.

b. A judgment and order requiring Netflix to pay Plaintiff its damages, costs, expenses, and prejudgment and post-judgment interest for its infringement of the asserted patents, as provided under 35 U.S.C. § 284;

c. A judgment and order requiring Netflix to provide an accounting and to pay supplemental damages to Realtime, including without limitation, prejudgment and post-judgment interest;

d. A judgment and order finding that this is an exceptional case within the meaning of 35 U.S.C. § 285 and awarding to Plaintiff its reasonable attorneys' fees against Netflix; and

e. Any and all other relief as the Court may deem appropriate and just under the circumstances.

DEMAND FOR JURY TRIAL

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

Respectfully submitted,

DATED: September 20, 2019

RUSS, AUGUST & KABAT

By: /s/ *Reza Mirzaie*

Marc A. Fenster (CA SBN 181067)
Email: mfenster@raklaw.com
Brian D. Ledahl (CA SBN 186579)
Email: bledahl@raklaw.com
Reza Mirzaie (CA SBN 246953)
Email: rmirzaie@raklaw.com
Paul Kroeger (CA SBN 229074)
Email: pkroeger@raklaw.com
C. Jay Chung (CA SBN 252794)
Email: jchung@raklaw.com
Philip X. Wang (CA SBN 262239)
Email: pwang@raklaw.com
12424 Wilshire Boulevard, 12th Floor
Los Angeles, CA 90025
Telephone: 310/826-7474
Facsimile 310/826-6991

Attorneys for Plaintiff
Realtime Adaptive Streaming LLC