Improving the Performance of Web-Streaming by Super-Resolution Upscaling

Yuriy Reznik
yreznik@brightcove.com
Brightcove Inc
Seattle, USA

Nabajeet Barman
nbarman@brightcove.com
Brightcove UK Ltd
London, United Kingdom

Patrick Wagstrom
pwagstrom@brightcove.com
Brightcove Inc
Boston, USA

ABSTRACT

In recent years, we have seen significant progress in advanced image and video upscaling techniques, sometimes called super-resolution, or AI-based upscaling. Such algorithms are now broadly available in the forms of software SDKs, as well as functions natively supported by modern graphics cards. However, to take advantage of such technologies in video streaming applications, one needs to (a) add support for super-resolution upscaling in the video rendering chain, (b) develop means for quantifying the effects of using different upscaling techniques on perceived quality, and c) modify streaming clients to use such more advanced scaling techniques in a way that leads to improvements in quality, efficiency, or both.

In this paper, we discuss several techniques addressing these challenges. We first present an overview of super resolution technology. We review available SDKs and libraries for adding super-resolution functionality in streaming players. We next propose a parametric quality model suitable for modeling the effects of different upscaling techniques. We validate it by using an existing widely used dataset with subjective scores. And finally, we present an improved adaptation logic for streaming clients, allowing them to save bandwidth while maintaining quality at the level achievable by standard scaling techniques. Our experiments show that this logic can reduce streaming bitrates by up to 38.9%.

KEYWORDS

Adaptive Streaming, Video Streaming, Super Resolution, Machine Learning, Deep Learning, Upsampling, Quality Enchancement

1 INTRODUCTION

Adaptive streaming, where the media playback is adapted based on the changing network conditions, is one of the fundamental technologies enabling a good user experience. This has resulted in the increasing growth and popularity of streaming services over the past two decades. In adaptive streaming of videos, the video is encoded in different representations, often called renditions. One of the most widely used adaptive streaming formats used by most over-the-top (OTT) service providers is HTTP-based Adaptive Streaming (HAS), where the streaming takes place over reliable transport protocols such as TCP.

1.1 HTTP-based Adaptive Streaming

In a typical HAS solution, such as HTTP Live Streaming (HLS) [7] and Dynamic Adaptive Streaming over HTTP (DASH) [8], the video is encoded in multiple resolution-bitrate pairs. The streaming client (player), depending on the available network throughput, buffer status, and player size, selects the appropriate rendition for playback [20]. The player at the end-user device typically upscales (or, in some cases, downscales) the videos to fit the player/window resolution. In the special case of web streaming, as discussed by the authors in [23], player window size significantly impacts the selection of streams. In such systems where the videos are delivered embedded in web pages, the network bandwidth is no longer the only factor influencing the selection of streams. Many modern streaming clients also consider the player (window) size as one of the factors in their adaptation logic [21]. However, in most cases, the adaptation logic is very simplistic, e.g., limiting the upscale factor, selecting the nearest matching resolution in the ladder, etc. Such simple resolution adaptation algorithms do not necessarily account for viewing setup parameters such as pixel density and viewing angle. More importantly, to the best of the authors' knowledge, the existing resolution adaptation algorithms do not account for the effect of upsampling methods being used on the client side.

1.2 Advanced AI-based Upsampling Algorithms

Traditionally, image/video scaling in web browsers has been implemented by using classical signal processing-based techniques such as *bi-cubic* interpolation or *sinc* and *lanczos* filters. More recently, however, there has been a growing interest and work in the field of AI/ML-based upscaling, often termed as Super Resolution (SR) techniques [12, 26]. Such algorithms are primarily based on deep learning technologies such as Convolutional Neural Networks (CNNs) and, more recently, Generative Adversarial Networks (GANs). Such advanced SR algorithms are typically designed and used to perform ×2 and ×4 upsampling of lower-resolution images and videos. These newly designed SR algorithms are shown

Table 1: Examples of existing SR algorithms.

SR Implementation	Implementation	Reference
VDSR	Open-source	Kim et al. [10]
EDVR	Open-source	Wang et al. [24]
COMISR	Open-source	Google [11]
BasicVSR	Open-source	Chan et al. [5]
WebGL in video.js	WebGL-based, open source	N. Chadwick and M. Matthew [4]
RTX Video Super Resolution	Proprietary, SR support in MS Edge and Chrome	Nvidia [22]
MAXINE	Proprietary	Nvidia [17]
FidelityFX™ Super Resolution (FSR) 2	Open-source	AMD [3]

to outperform traditional upsampling methods when evaluating the reconstruction quality on various datasets in terms of objective quality metrics such as PSNR and SSIM [25] and in some cases, subjectively using mean opinion scores [12, 26].

Due to their improved performance over existing traditional upsampling algorithms, such advanced algorithms are getting increasingly popular, with many companies offering such AI-based upscaling solutions[1 2]. Table 1 presents a summary of a few available super-resolution algorithms. Such works can broadly be classified as academic/research implementations (e.g., [5, 10, 24]), WebGL-based implementations that may be used in web browsers with js-based players (e.g., [4]) and implementations supported by native features of GPUs (e.g., [3, 17]).

1.3 Open Questions

The rapidly evolving field of super resolution brings with it a new range of questions that must be answered before such technologies can be adopted into mainstream video streaming applications. A few of these are highlighted next.

(1) *What are the advantages of SR over traditional scaling?* Many of the proposed models have been evaluated on a small number of datasets. Their performance evaluation for real-world video streaming applications considering adaptive streaming applications and their benefits compared to traditional upscaling algorithms remains an open question.
(2) *How to model/quantify super-resolution scaling capability?* Most of the performance metrics used to quantify existing SR techniques have been limited to PSNR and, in some cases, SSIM [25]. However, such metrics are often limited in terms of their correlation with human-perceived quality and, oftentimes, unsuitable for measuring the quality of AI-based algorithms [28].
(3) *How to use SR for improved image/video delivery?* Advanced AI-based upsampling algorithms and newer, alternative content such as HDR result in increased quality saturation at lower angular resolutions. Understanding the impacts of the encoded video resolution and scaling algorithms on the perceived quality is essential for applications to allow them to select the optimal renditions [18, 20]. Such intelligent adaptation algorithms, considering the effect of upsampling algorithms, can result in significant bandwidth and storage savings.

[1]https://www.topazlabs.com/topaz-video-ai
[2]https://www.avclabs.com/

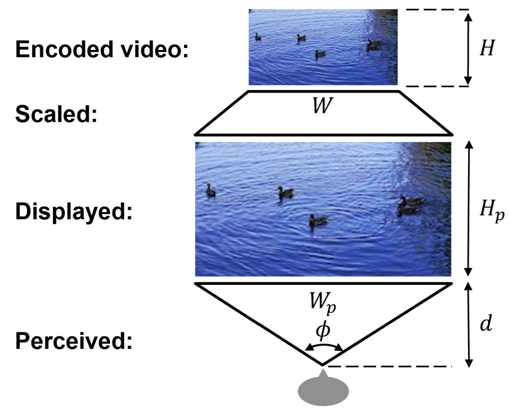

Figure 1: Parameters of video reproduction setup.

1.4 Contributions

This paper presents several contributions which can help understand and quantify the impacts of more advanced, AI-based upscaling algorithms with a focus on optimal rendition selection by streaming players/clients. We first present and discuss a generalized model of the well-known Westerink and Roufs model, which uses angular parameters, viewing angle, and angular resolution to predict the perceived picture quality [15, 27]. The generalized model provides additional parameters which allow it to be tuned to adapt to the differences in perceived picture quality due to the use of different upsampling algorithms. Using the generalized model, we present an improved adaptation logic for video streaming clients, considering the player size and the upsampling algorithm used at the client for upscaling the received video. We conclude the paper with some results for a sample case, demonstrating the utility of the proposed model for selecting an optimal set of renditions, resulting in significant bandwidth savings.

2 MEASURING THE EFFECT OF SCALING ON PERCEIVED QUALITY

2.1 Viewing Setup

Table 2 presents a list of the main parameters of the video, player, and characteristics of the viewing setup. Figure 1 illustrates a typical video reproduction chain explaining the relationship between the various parameters. In this figure, an encoded video sequence of size $W \times H$ [pixels] is scaled to fit a player/display of size $W_p \times H_p$ [pixels]. Parameter d [inches] denotes the viewing distance between the observer and the display. We next introduce two angular metrics as typically used to quantify perceptual effects [1, 27].

2.1.1 Viewing Angle. The viewing angle, ϕ, describes the horizontal angular size of the video as visible to the user. Considering a video player window of size $W_p \times H_p$, display pixel density ρ, and viewing distance d, the viewing angle ϕ can be computed as:

$$\phi = 2 \arctan \left(\frac{W_p}{2d\rho} \right). \tag{1}$$

Table 2: Key parameters of video and viewing setup.

Parameter	Description	Units
W_p	display/player window width	pixels
H_p	display/player window height	pixels
W	horizontal image resolution	pixels
H	vertical image resolution	pixels
d	viewing distance	inches
ρ	display pixel density	pixels per inch

2.1.2 Angular Resolution. Angular resolution μ effectively describes the Nyquist frequency of the video, presented in angular units. Using the same parameters as defined above and considering that the resolution of the video being played is $W \times H$ [pixels], the angular resolution, μ, of the video at that resolution can be computed as:

$$\mu = \left(2 \arctan\left(\frac{W_p}{W d \rho}\right)\right)^{-1}. \tag{2}$$

2.2 Encoding Ladders

When video content is prepared for media delivery, it is typically encoded into multiple streams (or *renditions*). Such streams typically differ by choices of resolutions and bitrates used for encoding. Table 3 shows an example ladder consisting of 13 streams encoded using HEVC [9] at different bitrates. This ladder covers all standard video resolutions ranging from 108p to 4K [6].

Subsequently, in referring to encoding ladders, we will use H_i and W_i to denote the height and width of i-th rendition in the ladder, $i = 1, \ldots, n$, and where n represents the number of streams in the ladder. In this work, we will assume that all the renditions in an adaptive bitrate ladder use the same aspect ratio (e.g., 16:9). This simplification will allow for the specification of a single resolution parameter, e.g., height H_i, to derive the other. For simplicity, we will also assume that renditions' resolutions form an ascending order: $H_1 \leq H_2 \leq \ldots \leq H_n$.

2.3 Video Players

We next note that the size of a video player window generally depends on user preferences and the type of user device. E.g., on PCs, some users may prefer to play videos full screen, while others may play them as they appear on web pages, subject to the position and size of the browser's window [23].

In general, we will assume that the video player size is $W_p \times H_p$ [pixels] and that the player employs an adaptation logic selecting streams with a resolution that closely matches the current player size, subject to the available network bandwidth and some other constraints [20].

2.4 Quality Model Based on Viewing Setup Parameters

A well-known technique for assessing the perceived quality based on the parameters of viewing setup is the Westerink and Rouf (WR) model [27]. It establishes that subjective image quality is influenced independently by both the viewing angle of the projected image and the angular resolution of the projected picture on display. The

Table 3: Example encoding ladder.

Stream	Codec	Width [pixels]	Height [pixels]	Framerate [fps]	Bitrate [kbps]
1	HEVC	192	108	59.94	260
2	HEVC	320	180	59.94	500
3	HEVC	384	216	59.94	640
4	HEVC	480	270	59.94	930
5	HEVC	640	360	59.94	1350
6	HEVC	768	432	59.94	1960
7	HEVC	960	540	59.94	2550
8	HEVC	1280	720	59.94	3690
9	HEVC	1600	900	59.94	5350
10	HEVC	1920	1080	59.94	6950
11	HEVC	2560	1440	59.94	11130
12	HEVC	3200	1800	59.94	16140
13	HEVC	3840	2160	59.94	23400

model since then has been validated by others and used in many works [1, 13, 19] and more recently in [15].

In this work, we use the Generalized WR (GWR) model proposed in [15]:

$$Q(\phi, \mu) = \log\left(\alpha + \beta\left(1 + \left(\frac{\phi}{\phi_s}\right)^{-k}\right)^{-\frac{\gamma}{k}}\left(1 + \left(\frac{\mu}{\mu_s}\right)^{-l}\right)^{-\frac{\delta}{l}}\right) \tag{3}$$

where γ, δ, k, l, ϕ_s, and μ_s are model parameters controlling the behavior with respect to viewing angle and angular resolution. As we will show later, this allows the model to be tuned to consider differences in HDR and SDR content and upsampling algorithms.

As reported in [15], the fit of this model to the six modern datasets considering different viewing setups (UHD TV to smartphones and tablets) and resolutions (QCIF to 4k/UHD) yields the following model parameters: α= 2.72, β=145.69, γ = 1.55, δ=2.12, k=6.01, l=2.11, ϕ_s=35.0, and μ_s=16.93. This model outperforms the original WR model on the same datasets. The use of this generalized WR model as a basis for deriving more complex quality models, also incorporating distortion measures, was recently reported in [16].

3 MEASURING THE EFFECTS OF SUPER RESOLUTION

For AI/ML-based upsampling algorithms such as Super Resolution (SR), when compared to the traditional algorithms such as Nearest Neighbour (NN) or bi-cubic (BC) interpolation algorithms, it is observed that MOS scores often reach saturation at lower angular resolutions. In this section, we discuss how the proposed model can be re-tuned to consider the differences in subjective perception due to different upsampling algorithms. We first introduce the dataset used, and then we discuss how the GWR model discussed in Section 2.4 can be retuned to different upsampling algorithms.

3.1 Dataset

The open-source *BVI* dataset [14] provides subjective scores considering three different upsampling algorithms. The dataset consists of 24 10-bit five-second source reference videos sequences of 60 fps encoded in three different resolutions (1920×1080, 960×540 and

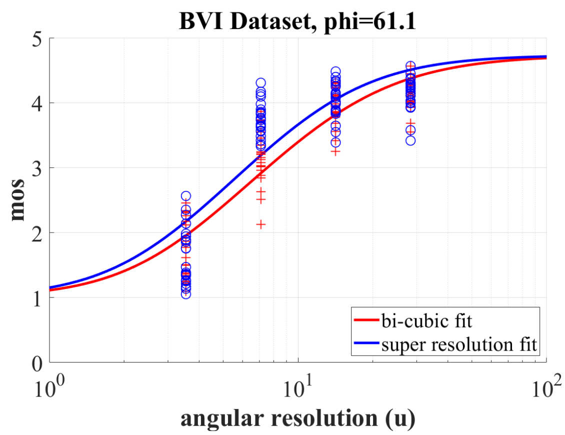

Figure 2: MOS vs Angular resolution (cpd) plot considering bi-cubic and Super Resolution scaling.

480×270), which were then upsampled to the source resolution of 3840x2160p using three different upsampling algorithms: bi-cubic (BC), Nearest Neighbour (NN) and Super Resolution (SR) [10]. The upscaled videos were then displayed to test subjects on a display measuring 65.4 × 36.8 cm of BT.2020 color space (full range) at a viewing distance of 1.5H.

3.2 Model Refit to Different Upsampling Algorithms

Since the new dataset is of a different content type (10-bit, HDR) than the GWR model was designed for, we first perform a refit of the GWR model using all MOS scores from the BVI dataset. Allowing for different values of α and β, as well as adding a common scale factor, ϵ for γ and δ, we obtain the new parameter values as: $\alpha = 2.72, \beta = 106.91, \epsilon = 1.08, \gamma = 1.55\epsilon, \delta = 2.12\epsilon$. However, fit alone to account for differences in content type is insufficient. For improved performance, one needs to consider the differences due to upsampling algorithm. This requires re-tuning the model to each specific upsampling algorithm subset of the dataset, which is discussed next.

To account for the differences in upsampling algorithms, we now fit the GWR model to BC and SR upsampling algorithms subsets of the BVI dataset corresponding to the upsampling method used. To adapt to the differences, we allow only parameters μ_s and l to vary (since they control the model behavior wrt μ). Rest all parameter values are set to default (as obtained from the refit to the whole of the BVI dataset considering all three upsampling algorithms).

Figure 2 and Table 4 present the results for the fit for two of the upsampling algorithms, one traditional (bi-cubic) and one AI-based (Super Resolution). Based on the custom fit for both upsampling algorithms in Figure 2, it is obvious that SR upsampling reaches higher MOS at lower angular resolutions (μ) as compared to traditional bi-cubic upsampling. Consequently, the degree to which one needs to upscale videos to achieve a particular MOS is less in the case of SR upsampling than in BC upsampling. Service providers can use such information to compute relative savings in encoding resolutions or optimal encoding ladder generation.

Table 4: Model refit parameters for each upsampling algorithm.

Upsampling Algorithm	μ_s	l
Bi-cubic (BC)	13.93	1.76
Super Resolution (SR)	12.24	2.06

4 SUPER RESOLUTION AWARE ADAPTATION ALGORITHMS FOR STREAMING CLIENTS

As discussed earlier, in a modern-era adaptive streaming system delivering videos embedded in web pages, the stream selection logic is jointly influenced by both available network bandwidth and output video size (player size). While adaptation to network bandwidth has been widely studied [2], stream adaptation based on the resolution of available renditions and the player size has received little attention [18]. In this section, we present a discussion about improving the ABR algorithms considering the second aspect, i.e., adaptation to resolution, while also taking into account the effect of client-side upsampling algorithms (traditional vs super resolution based upsampling).

4.1 Algorithms

We present two optimal rendition resolution selection algorithms based on the player size and the upsampling algorithm. The first algorithm, Algorithm 1, finds the best rendition from a list of available renditions, which delivers the best possible quality for a given viewing setup (viewing distance, player size, and device type). The GWR model (3) is used to guide the best rendition selection. In this particular algorithm, we assume that playback is done by using bi-cubic upsampling. Therefore, the GWR model employed is the one tuned for the bi-cubic upsampling method (μ_s = 13.93 and l =1.76, cf. Table 4).

The second algorithm, Algorithm 2, is an extension of Algorithm 1, designed to achieve the same perceptual quality but in a system employing Super Resolution (SR) upsampling. To achieve this effect, Algorithm 2 takes as input the best rendition previously found by Algorithm 1, as well as the quality achieved by using bi-cubic upsampling, and then tries to find a new rendition with the smallest possible resolution, such that the resulting quality produced by SR upsampling matches the one achieved earlier. To estimate perceived quality, Algorithm 2 uses the GWR model tuned for the SR upsampling method (μ_s = 13.93 and l=1.76).

With such a modified selection, Algorithm 2 allows the player to pull streams with fewer pixels and reduced bitrate while delivering quality comparable to one deliverable by the players using bi-cubic upscaling. Naturally, in practice, both algorithms would have to be more complex and consider various additional conditions, variables, and constraints (bandwidth, CPU load, battery charge, potentially changing aspect ratios, and framerates of videos across renditions, etc.). Nevertheless, the algorithms capture the core optimization logic as needed for adaptation to resolutions, viewing conditions, and upscaling methods.

Algorithm 1: Optimal Rendition Resolution Selection Based on Player Size and BC Upsampling Algorithm

Data:
Viewing angle, ϕ
Angular resolution, μ
Available video rendition heights, $H_{renditions} = H_1, ...H_n$
Player Window Height, H_p
Distance from the display, d
Effective pixel density of the screen, ρ
Client Side Upsampling Algorithm, UP_{algo} = BC
Model fit values, α=2.72, β=106.91, ϵ=1.08, γ=1.55ϵ, δ=2.12ϵ.

Result: Best rendition height (BC Upsampling), $H_{best_{BC}}$, and Best MOS (BC Upsampling), $MOS_{best_{BC}}$

```
best_mos = 0;
best_rendition-index = 1;
for i ← 1 to n do
    Calculate Viewing angle φ
    Calculate Angular resolution μ
    μs = 13.93; l = 1.76 ;  /* BC Upsampling, Table 4 */
    Calculate MOS, Q(φ, μ) ;          /* Using Eqn 3 */
    if MOS is ≥ best_mos then
        best_mos = MOS ;
        best_rendition-index = i ;
    end
end
H_bestBC = H_renditions(best_rendition-index)
MOS_bestBC = best_mos
```

Algorithm 2: Optimal Rendition Resolution Selection Based on Player Size and SR Upsampling Algorithm

Data:
Viewing angle, ϕ
Angular resolution, μ
Available video rendition heights, $H_{renditions} = H_1, ...H_n$
Player Window Height, H_p
Distance from the display, d
Effective pixel density of the screen, ρ
Model fit values, α = 2.72, β = 106.91, ϵ = 1.08, γ = 1.55ϵ, δ = 2.12ϵ.
MOS values from Algorithm 1, $MOS_{best_{BC}}$

Result: Best rendition height (Super Resolution Upsampling), $H_{best_{SR}}$, and Best MOS (Super Resolution Upsampling), $MOS_{best_{SR}}$

```
best_mos = MOS_bestBC ;     /* MOS from Algorithm 1 */
best_rendition-index = 1;
for i ← 1 to n do
    Calculate Viewing angle φ
    Calculate Angular resolution μ
    μs = 12.24; l = 2.06 ;  /* SR Upsampling, Table 4 */
    Calculate MOS, Q(φ, μ) ;          /* Using Eqn 3 */
    if MOS is ≥ best_mos then
        best_mos = MOS ;
        best_rendition-index = i ;
        break; ; /* Minimum Rendition found, exit */
    end
end
H_bestSR = H_renditions(best_rendition-index)
MOS_bestSR = best_mos
```

4.2 Simulation Results

We next study the performance that may be achieved by employing both proposed algorithms.

As a viewing environment, we will assume the same reproduction setup as was used in the BVI dataset [14]: 31″ 4K monitor, 1.5H viewing distance. This is the same dataset that was used earlier to retune the GWR model to different upsampling methods. The encoding profile and parameters of streams produced for an example video (sequence "Crosswalk" from BVI dataset [14]), as shown in Table 3, were used in our experiment. Notably, this profile covers the entire set of resolutions as allowed for DVB-DASH [6], ranging from 108p to 2160p.

In our simulation, we also assume that video player window size H_p can vary in the same range: $H_p \in [108, .., 2160]$. For each possible player resolution H_p, we use Algorithm 1 and Algorithm 2 to find the best matching renditions. We present the outputs of our selection algorithms in Figures 3, 4, and 5.

In Figure 3, we show the rendition resolution (height H) as selected based on player window size. It can be observed that for large player sizes, the SR-based algorithm picks progressively smaller resolution renditions as compared to the algorithm relying on standard bi-cubic based rendering. In Figure 4, we show the effect of such selections on the rendition bitrates and hence bandwidth used by streaming clients. It can be observed that, again, the SR-based algorithm picks renditions with substantially smaller bandwidth. In particular, we see that in the high-resolution regime, the player employing SR with Algorithm 2 can deliver up to 38.94% in bitrate savings. Finally, in Figure 5, we show quality vs. bitrate plots achievable by systems with traditional (bi-cubic) vs AI-based (super resolution) upscaling. It can be observed that SR-based systems are considerably more effective.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we have briefly reviewed the existing SR-based upsampling algorithms and discussed problems still existing in enabling their use in streaming video applications. As one such problem, we have identified the lack of quality metric capturing the effects of SR algorithms on video quality. To address this problem, we have proposed to use a generalized Westerink-Roufs model with a subset of parameters tuned to different upsampling techniques. We have shown that this method works using the BVI dataset. Another problem was the need to modify clients' algorithms to take advantage of SR rendering. Towards this end, we have proposed two algorithms, one finding the renditions delivering the best quality and the other matching quality achievable by clients with normal scaling but lowering the bitrate. We have shown a simulation of the operation of both algorithms, demonstrating their effectiveness. Specifically, we noted that in high-player resolution regimes, the use of SR-based rendering could lead to up to 38.94% in bitrate savings. We find these results highly promising, showcasing the

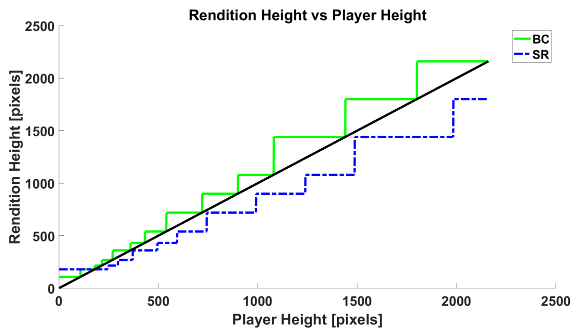

Figure 3: Best rendition resolutions found by using Algorithm 1 (green), and Algorithm 2 (blue).

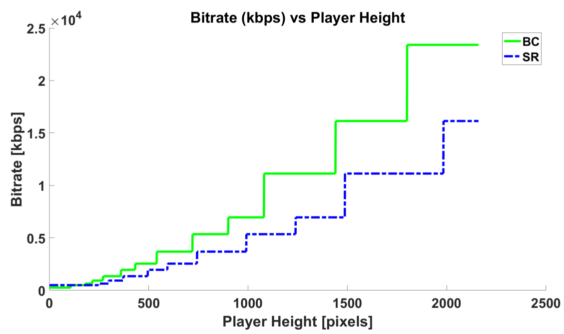

Figure 4: Bitrates of renditions selected by using Algorithm 1 (green), and Algorithm 2 (blue).

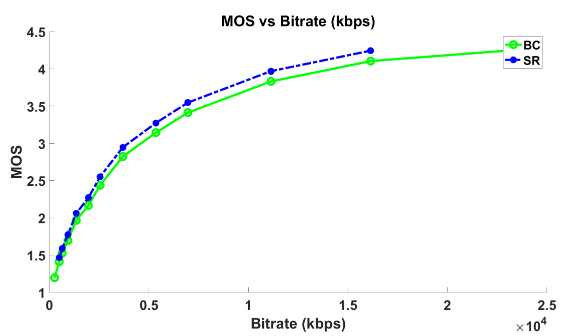

Figure 5: MOS vs Bitrate plots of players employing traditional (green) vs SR-based upscaling (blue) and adaptation.

potential for significant gains that may be achieved by using this class of techniques in streaming.

REFERENCES

[1] P.G.J. Barten. 1989. The effects of picture size and definition on perceived image quality. *IEEE Transactions on Electron Devices* 36, 9 (1989), 1865–1869.

[2] A. Bentaleb, B. Taani, A. C. Begen, C. Timmerer, and R. Zimmermann. 2019. A Survey on Bitrate Adaptation Schemes for Streaming Media Over HTTP. *IEEE Communications Surveys & Tutorials* 21, 1 (2019), 562–585.

[3] A. Blake-Davies. 2022. AMD FidelityFX Super Resolution 2.2. Retrieved 19-Feb-2023 from https://community.amd.com/t5/gaming/amd-fidelityfx-super-resolution-2-2-racing-into-more-games-and/ba-p/563910

[4] N. Chadwick and M. McClure. 2020. Super resolution upsampling in Video.js. Retrieved 19-Feb-2023 from https://github.com/muxinc/videojs-super-resolution

[5] K. K. Chan, X. Wang, K. Yu, C. Dong, and C. Loy. 2021. BasicVSR: The Search for Essential Components in Video Super-Resolution and Beyond. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 4945–4954.

[6] Digital Video Broadcasting (DVB). 2019. Digital Video Broadcasting (DVB) BlueBook A168 (Draft ETSI TS 103 285 V1.3.1). Retrieved 9-Feb-2023 from https://t.ly/5OQa

[7] Apple Inc. 2017. ISO/IEC 23009-1:2019 Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats. Retrieved 16-Feb-2023 from https://tools.ietf.org/html/rfc8216

[8] ISO/IEC. 2019. SO/IEC 23009-1:2022 Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats. Retrieved 13-Feb-2023 from https://www.iso.org/standard/83314.html

[9] ITU-T. 2021. ITU-T Recommendation H.265 : High efficiency video coding. Retrieved 21-Feb-2023 from https://www.itu.int/rec/T-REC-H.265

[10] J. Kim, J. K. Lee, and K. M. Lee. 2016. Accurate Image Super-Resolution Using Very Deep Convolutional Networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1646–1654.

[11] Y. Li, P. Jin, F. Yang, C. Liu, M. Yang, and P. Milanfar. 2021. COMISR: Compression-Informed Video Super-Resolution. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE Computer Society, Los Alamitos, CA, USA, 2523–2532.

[12] H. Liu, Z. Ruan, P. Zhao, C. Dong, F. Shang, Y. Liu, L. Yang, and R. Timofte. 2022. Video Super-Resolution Based on Deep Learning: A Comprehensive Survey. *Artif. Intell. Rev.* 55, 8 (Dec 2022), 5981–6035.

[13] M. Lombard, T. B. Ditton, M. E. Grabe, and R. D. Reich. 1997. The role of screen size in viewer responses to television fare. *Communication Reports* 10, 1 (1997), 95–106.

[14] A. Mackin, M. Afonso, F. Zhang, and D. Bull. 2018. A Study of Subjective Video Quality at Various Spatial Resolutions. In *2018 25th IEEE International Conference on Image Processing (ICIP)*. 2830–2834.

[15] N. Barman, R. Vanam and Y. Reznik. 2022. Generalized Westerink-Roufs Model for Predicting Quality of Scaled Video. In *The 14th International Conference on Quality of Multimedia Experience (QoMEX)*. Lippstadt, Germany, 1–6.

[16] N. Barman, R. Vanam and Y. Reznik. 2022. Parametric Quality Models for Multi-screen Video Systems. In *10th European Workshop on Visual Information Processing (EUVIP)*. Lisbon, Portugal, 1–6.

[17] Nvidia. 2022. Maxine. Retrieved 19-Feb-2023 from https://developer.nvidia.com/maxine

[18] Y. Reznik, N. Barman, and R. Vanam. 2022. Optimal rendition resolution selection algorithm for web streaming players. In *Applications of Digital Image Processing XLV*, Vol. 12226. International Society for Optics and Photonics, SPIE, 122260F.

[19] Y. A. Reznik. 2021. Average Performance of Adaptive Streaming. In *2021 Data Compression Conference (DCC)*. 263–272.

[20] Y. A. Reznik, K. O. Lillevold, A. Jagannath, and X. Li. 2021. Towards Understanding of the Behavior of Web Streaming. In *2021 Picture Coding Symposium (PCS)*. 1–5.

[21] Y. A. Reznik, K. O. Lillevold, and R. Vanam. 2021. Perceptually Optimized ABR Ladder Generation for Web Streaming. *Electronic Imaging* 2021, 3 (2021), 75–1–75–11.

[22] S. Sen. 2023. Nvidia RTX Video Super Resolution: 4K AI upscaling for Chrome and Edge Browsers . Retrieved 19-Feb-2023 from https://www.neowin.net/news/ces-23-nvidia-outs-rtx-4070-ti-new-rtx-video-super-resolution-for-microsoft-edge--chrome/

[23] Thiago Teixeira, Bo Zhang, and Y. Reznik. 2021. Adaptive Streaming Playback Statistics Dataset. In *Proceedings of the 12th ACM Multimedia Systems Conference* (Istanbul, Turkey) *(MMSys '21)*. 248–254.

[24] X. Wang, K. K. Chan, K. Yu, C. Dong, and C. Loy. 2019. EDVR: Video Restoration With Enhanced Deformable Convolutional Networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE Computer Society, Los Alamitos, CA, USA, 1954–1963.

[25] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. 2004. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.

[26] Z. Wang, J. Chen, and S. C. H. Hoi. 2021. Deep Learning for Image Super-Resolution: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43, 10 (2021), 3365–3387.

[27] J. H. D. M. Westerink and J. A. J. Roufs. 1989. Subjective Image Quality as a Function of Viewing Distance, Resolution, and Picture Size. *SMPTE Journal* 98, 2 (1989), 113–119.

[28] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 586–595.