

Neural Enhancement in Content Delivery Systems: The State-of-the-Art and Future Directions

Royson Lee^{†*}, Stylianos I. Venieris^{†*}, Nicholas D. Lane^{†,‡}

[†]Samsung AI Center, Cambridge [‡]University of Cambridge

* Indicates equal contribution.

ABSTRACT

Internet-enabled smartphones and ultra-wide displays are transforming a variety of visual apps spanning from on-demand movies and 360° videos to video-conferencing and live streaming. However, robustly delivering visual content under fluctuating networking conditions on devices of diverse capabilities remains an open problem. In recent years, advances in the field of deep learning on tasks such as super-resolution and image enhancement have led to unprecedented performance in generating high-quality images from low-quality ones, a process we refer to as neural enhancement. In this paper, we survey state-of-the-art content delivery systems that employ neural enhancement as a key component in achieving both fast response time and high visual quality. We first present the deployment challenges of neural enhancement models. We then cover systems targeting diverse use-cases and analyze their design decisions in overcoming technical challenges. Moreover, we present promising directions based on the latest insights from deep learning research to further boost the quality of experience of these systems.

CCS CONCEPTS

• **Computing methodologies** → *Computer vision tasks; Distributed computing methodologies.*

ACM Reference Format:

Royson Lee, Stylianos I. Venieris, Nicholas D. Lane. 2020. Neural Enhancement in Content Delivery Systems: The State-of-the-Art and Future Directions. In *Barcelona '20: International Workshop on Distributed Machine Learning, June 03–05, 2020, Barcelona, Spain*. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/1122445.1122456>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Barcelona '20, June 03–05, 2020, Barcelona, Spain

© 2020 Association for Computing Machinery.

ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00

<https://doi.org/10.1145/1122445.1122456>

1 INTRODUCTION

Internet content delivery has seen a tremendous growth over the past few years. Specifically, video traffic is estimated to account for 82% of global Internet traffic by 2022 – up from 75% in 2017 [7]. This growth is attributed to not only the rapid increase of Internet-enabled devices, but also the support for higher-resolution content. For instance, an estimated 66% of TV sets will support Ultra-High-Definition (4K) videos by 2023 as compared to 33% in 2018 [6]. Most importantly, content traffic such as live streaming, video conferencing, video surveillance, and both short- and long-form videos-on-demand, are expected to rise very quickly. To meet these demands, a new class of distributed systems has emerged. Such systems span from video analytics frameworks that co-optimize latency and accuracy [12, 46], to content delivery systems [36, 52] that aim to maximize the quality of experience (QoE), estimated based on the chosen bitrate and amount of rebuffering.

One of the primary challenges of distributed systems for content delivery is their reliance on networking conditions. Currently, the quality of the communication channel between client and server plays a key role in meeting the application-level performance needs due to the significant amount of transferred data and the tight latency targets. Nevertheless, in real-life mobile networks, the communication speed fluctuates and poor network conditions lead to excessive response times, dropped frames or video stalling, that rapidly degrade the user experience. This phenomenon is further amplified by the increasing number of users which compete for the same pool of network resources.

One recent key method that enables tackling this challenge is *neural enhancement* through super-resolution (SR) and image enhancement models. These models are capable of processing a low-resolution/quality image and generating a high-quality output. With the unprecedented performance of convolutional neural networks (CNNs), content delivery systems have begun integrating neural enhancement models as a core component. The primary paradigm of using neural enhancement models in content delivery systems comprises the transmission of compact low-resolution/quality content, often along with the associated model, followed by its subsequent quality enhancement on the receiver side through

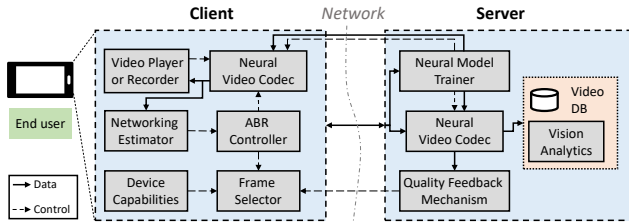


Figure 1: Architecture of content delivery systems.

a enhance-capable model [51]. In this manner, the transfer load is minimized, drastically reducing the network footprint and the corresponding bandwidth requirements.

Despite their benefits, integrating state-of-the-art neural enhancement models into visual content delivery systems introduces significant challenges. First, these models, especially SR models, have excessive computational demands that are measured up to the hundreds of TFLOPs per frame in order to achieve an upscaling of up to 4K/8K. With client platforms typically comprising devices with strict resource and battery constraints [2], clients are still struggling to execute neural enhancement models on-device while meeting the target quality [29]. This fact is aggravated by the stringent latency and throughput requirements that are imposed in order to sustain high QoE. Finally, enabling the deployment of such systems requires overcoming unique technical challenges stemming from the diversity of use-cases, spanning from on-demand video streaming [52] to video-conferencing [17].

This paper provides a timely and up-to-date overview of the growing area of visual content delivery systems that employ neural enhancement. In particular, we first describe the typical architecture and major components of such systems. We then survey the state-of-the-art existing systems (Table 1) across diverse content delivery applications including on-demand video and image services, visual analytics, video-conferencing, live streaming and 360° videos. We conclude by discussing future directions for the design of neural visual content delivery systems that draw from the latest progress by the computer vision community and describe how they can be integrated to value-add existing systems.

2 VISUAL CONTENT DELIVERY SYSTEMS

Content delivery systems are a widely studied field in systems research [4, 18, 23, 36, 54]. Such systems aim to deliver content to the users with minimal latency and high visual quality while supporting a diverse set of client platforms, from powerful desktops to mobile devices. Fig. 1 depicts the typical distributed architecture of a content delivery system. Focusing on video-on-demand without loss of generality, the client selects an online video to play and its video player app sends a request to the server side. The server fetches the video from its database, breaks it into seconds-long segments and begins streaming them to the client.

Table 1: Overview of Visual Content Delivery Systems

System	Task	Ref. CNN Model	CNN Execution
Yeo <i>et al.</i> [51]	On-demand video delivery	VDSR [25]	Client
NAS [52]	On-demand video delivery	MDSR [31]	Client
PARSEC [8]	360° video delivery	MDSR [31]	Client
Mobi SR [29]	On-demand image delivery	RCAN [58]	Client
Supremo [53]	On-demand image delivery	IDN [19]	Server
CloudSeg [46]	Video segmentation	CARN [1]	Server
Dejavu [17]	Video-conferencing	EDSR [31]	Client
LiVeNAS [26]	Live streaming	MDSR [31]	Server

For such systems to meet the performance targets, the communication channel between client and server has to sustain high bandwidth throughout the streaming process. This constitutes a strong assumption that breaks for mobile clients where the connectivity conditions vary continuously. Hence, additional techniques, such as adaptive bitrate and neural enhancement, have been introduced that enable the dynamic adaptation to the varying quality of the channel.

Adaptive Bitrate. To remedy the dependence of content delivery systems on network conditions, adaptive bitrate (ABR) algorithms have emerged [18, 36, 54]. Under this scheme, the client device first monitors its instantaneous bandwidth (*Networking Estimator* in Fig. 1) to assess the current network state, or the occupancy of its playback buffer. Next, the *ABR Controller* tunes accordingly the per-segment bitrate that it requests from the server and configures the client’s codec to decode at the selected rate. On the remote side, the server encodes each video segment with the specified bitrate and streams them to the client. Overall, ABR techniques function as a way to control the network footprint *at run time* and thus helps to minimize rebuffering. Although ABR has been significantly improved through deep learning-based algorithms [36], it often fails in scarce network conditions as it relies solely on network resources.

Neural Enhancement. Neural enhancement aims to restore and recover the quality/resolution of visual input. As this problem is inherently ill-posed, most works enforce a strong prior to mitigate its ill-posed nature. To this end, most state-of-the-art approaches utilize CNNs to learn the prior as it results in superior visual performance. These methods train a model to either map a low- to a high-quality image using exemplar pairs [10, 25, 31] or exploit the internal recurring statistics of the image to enhance/upscale it [40].

The primary paradigm of using neural enhancement models in content delivery systems comprises the transmission of compact low-resolution/low-quality content followed by its subsequent enhancement on the receiver side through the enhance-capable models [51, 52]. In this manner, the transfer load is tunable by means of the upscaling factor (for SR) and the degree of compression, controlling the system’s network footprint and the associated bandwidth requirements. Therefore, neural enhancement opens up a new dimension

in the design space by introducing a trade-off between computational and network resources, effectively overcoming existing systems’ sole reliance on network resources. To this end, existing systems may choose to independently optimize the utilization of these neural enhancement models [26, 29] or integrate them within existing ABR algorithms [8, 52].

Challenges of Neural Enhancement. Despite their advantages, neural enhancement CNNs are extremely expensive in terms of both computational and memory burden; these models are orders of magnitude larger than image discriminative models, with efficiency-optimized SR models [11, 28] measured in GFLOPs, whereas their image classification counterparts [15] are measured in MFLOPs [2]. Although the memory footprint is hugely addressed by splitting the image into patches before processing, the computational cost is still a big challenge for real-time applications [8, 29].

Furthermore, models trained on standard datasets that aim to generalize across all videos/images result in outputs of varying performance upon deployment and often fail catastrophically on unexpected inputs [51]. On the other hand, tailoring a CNN towards a specific video/image helps to mitigate this drop in performance at the cost of additional training per video/image. In this respect, a *generalization-specialization trade-off* is exposed which system designers need to decide how to control based on the target use-case.

3 THE LANDSCAPE OF CNN-DRIVEN VISUAL CONTENT DELIVERY

Despite their deployment barriers, several recent frameworks have incorporated neural enhancement methods into their pipelines and introduced novel techniques for overcoming their challenges. In this context, we survey the state-of-the-art visual content delivery systems that leverage neural enhancement models, taxonomizing them based on the type of content (video or image), and provide an analysis of how they counteract **a** the excessive computational requirements and **b** the performance variability across different content.

3.1 On-demand Content Delivery Systems

3.1.1 On-demand Video Streaming. Video-on-demand (VOD) services allow users to watch content at their suitable time from any Internet-enabled device. The user selects a video to watch which is then fetched and streamed by the video server to the user device. With the majority of VOD services being interactive, on-demand video streaming systems have to yield low response time and minimal rebuffering while not compromising visual quality in order to maximize the QoE. In achieving these targets, the bottleneck lies in the link between the video server and the client with the bandwidth of the connection directly affecting the end performance.

Yeo *et al.* [51] presented one of the first works that employed neural enhancement to overcome this limitation and offered a way to utilize the clients’ computational power. Specifically, the authors first grouped videos into clusters according to their category (basketball, athletics, *etc.*) and then trained a specialized SR model, VDSR [25], for each cluster, reducing the performance variation (challenge **b**) as compared to using a single generic model. They also proposed the use of more compact representations such as their edges or the luminance channel, on top of tuning the spatial size of the frames, to further reduce the use of both bandwidth and computational resources. Although these compact representations were shown not to work well in practice with the H.264 codec, they were adopted in later works, such as Dejavu [17], to tackle challenge **a** in different use-cases.

To handle the excessive computational needs of neural enhancement models (challenge **a**), the authors constrained their system to work with up to 720p videos and targeted homogeneous client platforms hosting powerful desktop-grade GPUs. This limitation was subsequently addressed to accommodate clients with heterogeneous computational capabilities in their extended proposed framework - NAS [52].

In NAS [52], the authors addressed the problem of heterogeneous clients through the use of early-exit SR models of varying sizes and computational workload, allowing each client to select the appropriate model (challenge **a**) based on their resource constraints. To this end, they extended previous reinforcement-learning-based ABR algorithms [36] to decide not only the bitrate, but also the fraction of the SR model to be transmitted for each video segment. Furthermore, the authors deploy a smaller variant of MDSR [31] quantized at 16-bit half-precision floating-point format and parallelize its execution across multiple GPUs on the client side. Finally, instead of categorizing videos into coarse clusters as in [51], NAS tackles challenge **b** by first pre-training a generic SR model and then fine-tuning a specialized model *for each video*. Overall, as shown in Fig. 2a, the client selects the bitrate b and the fraction j of the SR model for the i -th segment and receives the i -th low-resolution segment s_i^E and the associated fraction of the specialized model m_i^E for video v that are used locally by the *Super-resolution Processor* (SRP).

3.1.2 360° Video Streaming. Compared to regular videos, streaming 360° videos has significantly elevated bandwidth requirements. To alleviate this, existing systems employ viewport prediction techniques [13] which estimate which part of the video the user will look towards and only download this spatial content. However, accurate viewport prediction is still difficult to achieve, exacerbating the problem as missing patches of the current viewpoint are needed to be fetched at the time of viewing. Although neural enhancement models can be utilized to mitigate these challenges, the

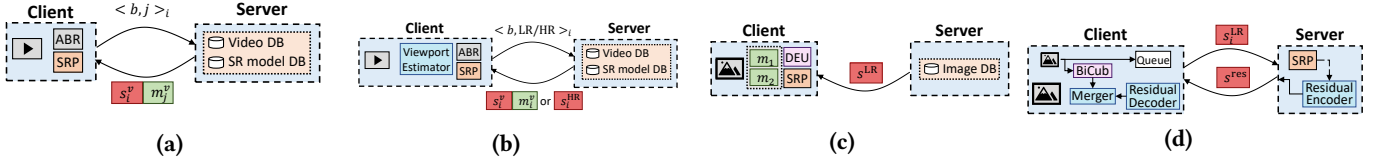


Figure 2: Overview of on-demand delivery systems: 2a) NAS [52], 2b) PARSEC [8], 2c) Mobi SR [29], 2d) Supremo [53].

larger spatial dimensions of 360° content further aggravates challenge **a** and calls for dedicated deployment solutions.

In this context, Dasari *et al.* [8] proposed a 360° video streaming framework named PARSEC (Fig. 2b). Unlike previous works, the authors extended the ABR algorithm to decide on the low-resolution (LR) patches to be upscaled and the bitrate of high-resolution (HR) patches to be downloaded based on 1) the networking conditions and the client’s computational resources, 2) the viewport prediction (see *Viewport Estimator* in Fig. 2b) and 3) the qualities (PSNR) of both the HR and the upscaled patches. Since the proposed ABR algorithm is designed to maximize QoE and can thus selectively decide which patches are upscaled or downloaded, challenge **b** is mitigated. To overcome challenge **a**, PARSEC combines an extreme upscaling factor of $\times 64$ that allows ultra-LR patches to be transmitted, together with a manually tuned efficient SR model, similar to that used by NAS, specialized for each video segment.

3.1.3 On-demand Image Delivery. As chipsets on commodity devices are gradually getting more powerful [2, 20], this enables many applications to run fully on-device, avoiding the latency and privacy issues of offloading. In this direction, Lee *et al.* [29] proposed Mobi SR (Fig. 2c), a system that capitalizes over the heterogeneous compute engines of modern smartphones, *e.g.* CPU, GPU and NPU, through a model selection mechanism to deliver rapid image super-resolution.

As a first step, Mobi SR derives two model variants by applying a wide range of compression techniques on a reference model (RCAN [58]) and then assigns the resulting Pareto-optimal models to the different available compute engines. The authors showed that both large and small models perform similarly on hard-to-upscale patches, with difficulty quantified using the total-variation metric [39]. Leveraging this insight, they used a hardware-aware scheduler, (*Difficulty Evaluation Unit* (DEU)), to rapidly process hard-to-upscale patches using a more compact model (m_1) while feeding the easier patches to a larger model (m_2) to obtain higher quality. Hence, the image quality is maximized while meeting the applications’ latency constraints (challenge **a**).

With respect to challenge **b**, Mobi SR is optimized to achieve higher overall performance through a generic model and does not employ model specialization. Finally, although Mobi SR pushes on-device neural enhancement, its achieved processing rate is still not suitable for real-time video use-cases but is sufficient for reducing the data usage when using image-centric applications such as Instagram and Reddit.

3.1.4 Cloud-assisted Image Delivery. To accommodate real-time use-cases, Yi *et al.* [53] proposed Supremo (Fig. 2d), a framework that enables real-time on-device SR by selectively offloading computation to the cloud. Following other works such as Li *et al.* [26], Supremo uses a lightweight variant of a SR model to be run on the resource-rich server, mitigating challenge **a**, and performs patch selection to only transmit key patches. Specifically, Supremo’s patch selection mechanism starts by extracting the edges from each image and sorting them according to edge intensity. Next, depending on the networking conditions, latency requirements and their ranking, these patches are sent to the cloud to be upscaled through an SR model. To further reduce the network footprint required to download the super-resolved patches, Supremo exploits the sparsity of the difference between the super-resolved patches and bicubic-upscaled patches. As these differences are often very sparse, encoding them through the *Residual Encoder* results in a heavily compressed signal, thus minimizing bandwidth. Similar to Mobi SR, Supremo handles challenge **b** by employing a generic model that aims to maximize the average upscaling performance across all processed images.

3.2 Live Content Streaming Systems

3.2.1 Streaming for Video Analytics. Pipelines for video analytics [12, 16, 56] perform real-time intelligent tasks over user inputs in order to enable the development of novel application such as augmented and virtual reality apps [32]. Such tasks span from scene labeling and object detection to face recognition. To meet the real-time performance requirements across diverse hardware platforms, such systems often rely on cloud-centric solutions. In this setup, the client device transmits the input frames to a powerful server for analysis and collects back only the result.

Naturally, these video analytic frameworks can benefit from the transmission of lower-resolution/quality images under low-bandwidth settings. However, the use of low-resolution/quality images is known to reduce accuracy [5]. Therefore, these frameworks can use additional server-side computation and deploy neural enhancement models to minimize the accuracy loss of the targeted task. To achieve this goal, Wang *et al.* [46] proposed CloudSeg (Fig. 3a) that jointly trains an SR model (CARN [1]) together with its target analytics task, *i.e.* semantic segmentation (ICNet [59]). Specifically, they updated CARN using gradients computed from the content loss between the HR and the super-resolved image and the accuracy difference between using both images in ICNet.

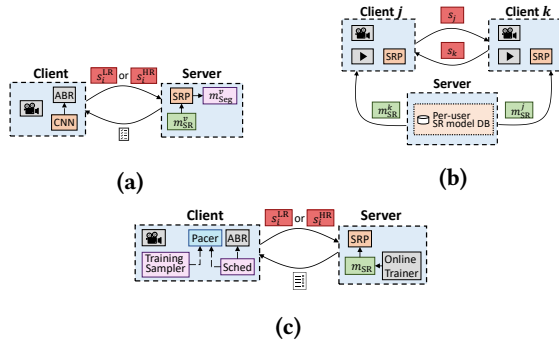


Figure 3: Overview of live streaming systems: 3a) Cl oudSeg [46], 3b) Dej avu [17], 3c) Li veNAS [26].

During inference, the SRP feeds both the LR image and super-resolved image to the pyramid segmentation model. Towards efficiency (challenge **a**), Cl oudSeg employs frame selection at the client side by deploying a small neural network that estimates pixel deviation in order to skip redundant stale frames. Finally, Cl oudSeg overcomes challenge **b** by falling back to streaming in high resolution whenever the analytics accuracy falls below a threshold.

3.2.2 Video-Conferencing. To sustain the interactive communication between the callers, video-conferencing requires low response time. In an effort to achieve that, existing services [14] often adopt conservative strategies that relax the bandwidth requirements, but also compromise visual quality.

In this context, Hu *et al.* [17] observed that in contrast to generic live streaming, video-conferencing has the property of visual similarity between *recurring* sessions and designed Dej avu (Fig. 3b) to exploit these unique computational and specialization opportunities. The developed system starts with the *offline* training of image enhancement models that are specialized *per caller* (challenge **b**). In this process, the model learns to improve the video quality by increasing its *encoding rate* without changing the resolution.

Upon deployment, when a video-conference session is established, the associated caller-specific enhancement model is transferred from a server to the receiver (and vice versa) ($m_{SR}^{\{:,q\}}$ in Fig. 3b). During the call, the frames from the caller are re-encoded to lower quality prior to transmission, reducing the bandwidth usage, and the quality is enhanced on the receiver side through the caller-specific enhancement model.

To address challenge **a**, Dej avu uses several techniques. First, a drastically scaled-down variant of the EDSR model [31] is used for the quality-enhancing CNN. Next, the model is trained only on the luminance (Y) channel, instead of the typical RGB input. At the same time, a powerful GPU is assumed to be available on each calling side. Finally, Dej avu introduces a patch-scoring CNN that predicts the expected quality gain for each image patch. In this manner, only the top- k patches that are expected to yield the highest quality

improvement are processed by the quality-enhancing neural network to lower the run-time resource usage.

3.2.3 Live Video Streaming. In contrast to VOD services that focus on stored content, live streaming targets content produced in real-time. In this case, an additional bottleneck is introduced on the upstream client-to-server channel, as a degraded quality from the streaming user would propagate to the end users that watch the video. This property leads to additional challenges in sustaining high QoE. Moreover, while stored content in VOD or the recurrence of video-conferencing allows for offline specialization of the enhancement models, the real-time nature of live streaming requires online methods to tailor the models to the incoming video.

To tackle this problem, Kim *et al.* [26] proposed the Li veNAS system (Fig. 3c) that focuses on optimizing the upstream transmission from streamer to server. In this system, a pre-trained generic SR model resides on the server side. Similarly to NAS [52], the selected model is a lightweight variant of MDSR [31]. Upon deployment, the streamer uploads a series of low-resolution frames which are then enhanced at the server side by the SR processor (SRP in Fig. 3c).

To counteract the performance variability across diverse content (challenge **b**), Li veNAS introduces an online learning scheme that tailors the model to the particular unseen video. This scheme consists of selectively picking high-quality patches using the *Training Sampler* and transmitting them from the streamer to the server using the *Pacer* (Fig. 3c). Since the patches needed for online training share bandwidth with the video, it is crucial to send only the patches with the highest expected impact. Hence, the *Training Sampler* detects patches that are hard to compress with high quality, by calculating the PSNR between HR patches and their bilinearly-interpolated LR encoding and selecting the lowest-PSNR patches. The *Pacer*, on the other hand, is responsible for allocating the available bandwidth between the low-resolution patches to be upscaled and the high-resolution training patches by adaptively tuning the respective bitrate through a quality-optimizing scheduling algorithm (*Sched* in Fig. 3c).

On the server side, the transferred high-resolution patches are used by the *Online Trainer* to fine-tune the SR model. The invocation frequency of the Online Trainer is controlled based on an adaptive mechanism that detects training saturation or scene changes, by tracking the performance gain of the latest model as well as comparing to the initial model. Thus, the amount of training for each live stream is adapted to maximize performance without excessive resource usage. Finally, to alleviate challenge **a**, Li veNAS supports scale-out execution by parallelizing the SR computations across multiple GPUs (*e.g.* three GPUs for 1080p to 4K real-time enhancement), if available, on the server.

4 FUTURE DIRECTIONS

In this section, we propose various approaches that are drawn from the latest computer vision research and provide insights on how neural enhancement can further benefit content delivery systems.

Visual Quality. One of the main open challenges in neural enhancement algorithms is the design of a metric that will correspond well with human raters. Distortion-based metrics, such as PSNR or SSIM [47] have been extensively shown to improve image fidelity at the cost of perceptual quality, leading to blurry and unnatural outcomes [27]. On the other hand, optimizing only for a perceptual-based metric such as NIQE [37] and LPIPS [57]¹ will lead to more natural looking images at the expense of fidelity and therefore the occasional occurrence of image artefacts. Mathematically, there is a trade-off between fidelity and perceptual quality [3].

As all the existing neural enhancement frameworks (Section 3) train their models using a distortion-based metric, the outputs of these models are accurate, but may look unnatural. Although this will benefit video analytics frameworks, such as CloudSeg, having blurry outputs will undermine the goal of other content streaming systems such as Dejavu and LiveNAS. To close this gap, these systems can benefit further by utilizing recent methods proposed in computer vision to train and optimize their neural enhancement models. These works focus on striking an optimal point between image fidelity and perceptual quality by optimizing either for both metrics [28, 45], the interpolation between a distortion-based and perceptual-based output image [9] or model [44, 45], or introducing additional priors [34, 42] to alleviate undesirable artefacts caused by optimizing for perception-based metrics. Finally, for video-based solutions, temporal interpolation methods [48, 60] can enable a system to increase the achieved frame rate, and hence the QoE, by estimating intermediate frames rather than transferring them.

Efficiency-optimized Models. Most neural enhancement frameworks generally adopt popular full-blown SR models (Table 1) and revise them in order to speed up training and inference or fit into the limiting constraints for client-side computation. However, these revisions are usually sub-optimal or, in some cases, detrimental. For instance, PARSEC uses of batch normalization [21] to speed up training, reducing image fidelity [31] and introducing image artefacts [45]. Therefore, instead of naively scaling down and modifying full-blown SR models, these systems can leverage off-the-shelf manually-designed efficient models such as IDN [19] -already used by Supremo, automatically-designed variants

such as ESRN [43] and TPSR [28] or even binarized SR models [35, 50] in order to deliver higher-quality enhancement at a lower computational cost (challenge **a**).

Image Rescaling. One of the key benefits of deploying an SR model as a neural enhancement unit is its ability to work without the high-resolution ground-truth. However, in many content delivery settings, the ground-truth is available. Hence, many of these works can leverage the downscaling operation during the upscaling process using neural image rescaling techniques to counteract performance variability (challenge **b**) and further boost image reconstruction. For instance, a downscaling CNN can be trained jointly with an existing SR model as shown in [24] and techniques such as encoder-decoder frameworks and invertible neural networks can also be utilized as shown in [30] and [49] respectively.

Despite its benefits, neural-based image rescaling incurs an additional cost of executing a downscaling neural network as compared to that of interpolation methods, utilizing additional computational resources for a more robust improvement in visual quality. Therefore, image rescaling techniques may be more suitable for on-demand video systems, such as NAS and PARSEC, in which the downscaling cost is an offline one-time cost across videos.

Meta-learning. To mitigate challenges **a** and **b**, many systems pre-train their generic neural enhancement networks offline before fine-tuning, either offline or online, for each specific image/video. For instance, NAS first trains a generic model before using its weights to fine-tune a separate model for each video in order to amortize the one-time offline training cost. In order to speed up and improve the performance during the fine-tuning process, these works can adopt a meta-learning approach in order to find a more optimal set of initialization parameters for fine-tuning. Specifically, pre-training a neural enhancement model via meta-learning on an external dataset will then require fewer gradient updates during the fine-tuning stage, therefore requiring less computational resources and leading to better performance as compared to naively fine-tuning [38, 41].

5 CONCLUSION

As the demand for content traffic grows over the upcoming years, the use of neural enhancement models will gain more traction in content delivery system design not only in existing applications but also emerging technologies such as augmented reality/virtual reality [22, 33] and telepresence [55]. Concurrently, as our everyday devices get more powerful, these models will ultimately run fully on-device, shifting the focus from accommodating stringent computational budgets to maximizing visual quality. By integrating ideas from both the computer vision and systems communities, we envision to align both ends towards more effective and deployable neural enhancement.

¹These metrics are accessible and easily computable using code provided by the respective authors.

REFERENCES

- [1] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. 2018. Fast, Accurate, and Lightweight Super-Resolution with Cascading Residual Network. In *European Conference on Computer Vision (ECCV)*.
- [2] Mario Almeida, Stefanos Laskaridis, Ilias Leontiadis, Stylianos I. Venieris, and Nicholas D. Lane. 2019. EmBench: Quantifying Performance Variations of Deep Neural Networks Across Modern Commodity Devices. In *The 3rd International Workshop on Deep Learning for Mobile Systems and Applications (EMDL)*. 6.
- [3] Yochai Blau and Tomer Michaeli. 2018. The Perception-Distortion Tradeoff. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [4] S. Borst, V. Gupta, and A. Walid. 2010. Distributed Caching Algorithms for Content Distribution Networks. In *2010 Proceedings IEEE INFOCOM*.
- [5] T. Chen, L. Ravindranath, Shuo Deng, P. Bahl, and H. Balakrishnan. 2015. Glimpse: Continuous, Real-Time Object Recognition on Mobile Devices. In *SensSys '15*.
- [6] Cisco. 2020. *Cisco Annual Internet Report (2018–2023) White Paper*. Technical Report. Cisco Systems, Inc. <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html> [Retrieved: October 13, 2020].
- [7] Cisco. 2020. *Cisco Visual Networking Index (VNI) Complete Forecast Update, 2017 - 2022*. Technical Report. Cisco Systems, Inc. https://www.cisco.com/c/dam/m/en_us/network-intelligence/service-provider/digital-transformation/knowledge-network-webinars/pdfs/1213-business-services-ckn.pdf [Retrieved: October 13, 2020].
- [8] Mallesh Dasari, A. Bhattacharya, Santiago Vargas, Pranjal Sahu, A. Balasubramanian, and S. Das. 2020. Streaming 360-Degree Videos Using Super-Resolution. In *IEEE Conference on Computer Communications (INFOCOM)*.
- [9] Xin Deng. 2018. Enhancing Image Quality via Style Transfer for Single Image Super-Resolution. *IEEE Signal Processing Letters* (2018).
- [10] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2016. Image Super-Resolution Using Deep Convolutional Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2016).
- [11] Chao Dong, Chen Change Loy, and Xiaoou Tang. 2016. Accelerating the Super-Resolution Convolutional Neural Network. In *European Computer on Computer Vision (ECCV)*.
- [12] Kuntai Du, Ahsan Pervaiz, Xin Yuan, Aakanksha Chowdhery, Qizheng Zhang, Henry Hoffmann, and J. Jiang. 2020. Server-Driven Video Streaming for Deep Learning Inference. *SIGCOMM* (2020).
- [13] Ching-Ling Fan, J. Lee, Wen-Chih Lo, C. Huang, Kuan-Ta Chen, and C. Hsu. 2017. Fixation Prediction for 360 Video Streaming in Head-Mounted Virtual Reality. In *Network and Operating System Support for Digital Audio and Video*.
- [14] Sadjad Fouladi, John Emmons, Emre Orbay, C. Wu, Riad S. Wahby, and Keith Winstein. 2018. Salsify: Low-Latency Network Video through Tighter Integration between a Video Codec and a Transport Protocol. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*.
- [15] A. Howard, Mark Sandler, G. Chu, Liang-Chieh Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Quoc V. Le, and H. Adam. 2019. Searching for MobileNetV3. *International Conference on Computer Vision (ICCV)* (2019).
- [16] Kevin Hsieh, Ganesh Anantharayanan, Peter Bodik, Shivaram Venkataraman, Paramvir Bahl, Matthai Philipose, Phillip B. Gibbons, and Onur Mutlu. 2018. Focus: Querying Large Video Datasets with Low Latency and Low Cost. In *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation (OSDI)*. 269–286.
- [17] Pan Hu, Rakesh Misra, and Sachin Katti. 2019. Dejavu: Enhancing Videoconferencing with Prior Knowledge. In *HotMobile*.
- [18] Te-Yuan Huang, Ramesh Johari, Nick McKeown, Matthew Trunnell, and Mark Watson. 2014. A Buffer-Based Approach to Rate Adaptation: Evidence from a Large Video Streaming Service. In *SIGCOMM*.
- [19] Zheng Hui, Xiumei Wang, and Xinbo Gao. 2018. Fast and Accurate Single Image Super-Resolution via Information Distillation Network. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [20] Andrey Ignatov, Radu Timofte, Andrei Kulik, Seungsoo Yang, Ke Wang, Felix Baum, Max Wu, Lirong Xu, and Luc Van Gool. 2019. AI Benchmark: All About Deep Learning on Smartphones in 2019. In *International Conference on Computer Vision (ICCV) Workshops*.
- [21] S. Ioffe and Christian Szegedy. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning (ICML)*.
- [22] Puneet Jain, Justin Manweiler, and Romit Roy Choudhury. 2015. Overlay: Practical Mobile Augmented Reality. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*.
- [23] Junchen Jiang, Vyas Sekar, and Hui Zhang. 2012. Improving Fairness, Efficiency, and Stability in HTTP-Based Adaptive Video Streaming with FESTIVE. In *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies (CoNEXT)*. 97–108.
- [24] Heewon Kim, Myungsub Choi, Bee Lim, and Kyoung Mu Lee. 2018. Task-Aware Image Downscaling. In *European Conference on Computer Vision (ECCV)*.
- [25] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. 2016. Accurate Image Super-Resolution using Very Deep Convolutional Networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [26] Jae-Hong Kim, Youngmok Jung, H. Yeo, Juncheol Ye, and D. Han. 2020. Neural-Enhanced Live Streaming: Improving Live Video Ingest via Online Learning. In *SIGCOMM*.
- [27] C. Ledig, L. Theis, Ferenc Huszar, J. Caballero, Andrew Aitken, Alykhan Tejani, J. Totz, Zehan Wang, and W. Shi. 2017. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [28] Royson Lee, L. Dudziak, M. Abdelfattah, Stylianos I. Venieris, H. Kim, Hongkai Wen, and N. Lane. 2020. Journey Towards Tiny Perceptual Super-Resolution. In *European Conference on Computer Vision (ECCV)*.
- [29] Royson Lee, Stylianos I. Venieris, L. Dudziak, S. Bhattacharya, and N. Lane. 2019. MobiSR: Efficient On-Device Super-Resolution through Heterogeneous Mobile Processors. In *The 25th Annual International Conference on Mobile Computing and Networking (MobiCom)*.
- [30] Y. Li, D. Liu, H. Li, Lianghuan Li, Z. Li, and F. Wu. 2019. Learning a Convolutional Neural Network for Image Compact-Resolution. *IEEE Transactions on Image Processing (TIP)* (2019).
- [31] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. 2017. Enhanced Deep Residual Networks for Single Image Super-Resolution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*.
- [32] Luyang Liu, Hongyu Li, and Marco Gruteser. 2019. Edge Assisted Real-Time Object Detection for Mobile Augmented Reality. In *The 25th Annual International Conference on Mobile Computing and Networking (MobiCom)*.
- [33] Luyang Liu, Hongyu Li, and Marco Gruteser. 2019. Edge Assisted Real-Time Object Detection for Mobile Augmented Reality. In *The 25th Annual International Conference on Mobile Computing and Networking*

(MobiCom).

- [34] Cheng Ma, Yongming Rao, Yean Cheng, Ce Chen, Jiwen Lu, and J. Zhou. 2020. Structure-Preserving Super Resolution With Gradient Guidance. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [35] Y. Ma, Hongyu Xiong, Zhe Hu, and L. Ma. 2019. Efficient Super Resolution Using Binarized Neural Network. *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2019).
- [36] Hongzi Mao, R. Netravali, and M. Alizadeh. 2017. Neural Adaptive Video Streaming with Pensieve. *SIGCOMM* (2017).
- [37] Anish Mittal, Rajiv Soundararajan, and Alan C. Bovik. 2013. Making a "Completely Blind" Image Quality Analyzer. *IEEE Signal Processing Letters* (2013).
- [38] Seobin Park, Jinsu Yoo, Donghyeon Cho, Jiwon Kim, and Tae Hyun Kim. 2020. Fast Adaptation to Super-Resolution Networks via Meta-Learning. In *European Conference on Computer Vision (ECCV)*.
- [39] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. 1992. Nonlinear Total Variation Based Noise Removal Algorithms. *Phys. D* 60, 1-4 (Nov. 1992), 259–268.
- [40] Assaf Shocher, N. Cohen, and M. Irani. 2018. "Zero-Shot" Super-Resolution Using Deep Internal Learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [41] Jae Woong Soh, Sunwoo Cho, and N. I. Cho. 2020. Meta-Transfer Learning for Zero-Shot Super-Resolution. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [42] Jae Woong Soh, G. Y. Park, Junho Jo, and N. I. Cho. 2019. Natural and Realistic Single Image Super-Resolution With Explicit Natural Manifold Discrimination. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [43] Dehua Song, Chang Xu, Xu Jia, Yiyi Chen, Chunjing Xu, and Yunhe Wang. 2020. Efficient Residual Dense Block Search for Image Super-Resolution. In *Association for the Advancement of Artificial Intelligence (AAAI)*.
- [44] Xintao Wang, K. Yu, C. Dong, X. Tang, and Chen Change Loy. 2019. Deep Network Interpolation for Continuous Imagery Effect Transition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [45] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Chen Change Loy, Yu Qiao, and Xiaoou Tang. 2018. ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks. In *European Conference on Computer Vision Workshops (ECCVW)*.
- [46] Yiding Wang, Weiyan Wang, Junxue Zhang, J. Jiang, and K. Chen. 2019. Bridging the Edge-Cloud Barrier for Real-time Advanced Vision Analytics. In *HotCloud*.
- [47] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing (TIP)* (2004).
- [48] X. Xiang, Yapeng Tian, Yulun Zhang, Y. Fu, J. Allebach, and Chenliang Xu. 2020. Zooming Slow-Mo: Fast and Accurate One-Stage Space-Time Video Super-Resolution. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020).
- [49] Mingqing Xiao, Shuxin Zheng, Chang Liu, Yaolong Wang, Di He, Guolin Ke, Jiang Bian, Zhouchen Lin, and Tie-Yan Liu. 2020. Invertible Image Rescaling. In *European Conference on Computer Vision (ECCV)*.
- [50] Jingwei Xin, Nannan Wang, Xinrui Jiang, Jie Li, Heng Huang, and Xinbo Gao. 2020. Binarized Neural Network for Single Image Super Resolution. In *European Conference on Computer Vision (ECCV)*.
- [51] H. Yeo, Sunghyun Do, and D. Han. 2017. How will Deep Learning Change Internet Video Delivery?. In *HotNets*.
- [52] H. Yeo, Youngmok Jung, Jaehong Kim, Jinwoo Shin, and D. Han. 2018. Neural Adaptive Content-aware Internet Video Delivery. In *13th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*.
- [53] Junheon Yi, Seongwon Kim, Joongheon Kim, and Sunghyun Choi. 2020. Supremo: Cloud-Assisted Low-Latency Super-Resolution in Mobile Devices. *IEEE Transactions on Mobile Computing (TMC)* (2020).
- [54] Xiaoqi Yin, A. Jindal, V. Sekar, and B. Sinopoli. 2015. A Control-Theoretic Approach for Dynamic Adaptive Video Streaming over HTTP. In *SIGCOMM*.
- [55] E. Zakharov, Aleksei Ivakhnenko, Aliaksandra Shysheya, and V. Lempitsky. 2020. Fast Bi-layer Neural Synthesis of One-Shot Realistic Head Avatars. In *European Conference on Computer Vision (ECCV)*.
- [56] Haoyu Zhang, Ganesh Ananthanarayanan, Peter Bodik, Matthai Philipose, Paramvir Bahl, and Michael J. Freedman. 2017. Live Video Analytics at Scale with Approximation and Delay-tolerance. In *Proceedings of the 14th USENIX Conference on Networked Systems Design and Implementation (NSDI)*. 377–392.
- [57] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [58] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. 2018. Image Super-Resolution Using Very Deep Residual Channel Attention Networks. In *European Conference on Computer Vision (ECCV)*.
- [59] Hengshuang Zhao, Xiaojuan Qi, Xiaoyong Shen, J. Shi, and J. Jia. 2018. ICNet for Real-Time Semantic Segmentation on High-Resolution Images. In *European Conference on Computer Vision (ECCV)*.
- [60] Liad Pollak Zuckerman, S. Bagon, Eyal Naor, George Pisha, and M. Irani. 2020. Across Scales & Across Dimensions: Temporal Super-Resolution using Deep Internal Learning. In *European Conference on Computer Vision (ECCV)*.