Image Enhancement

Kai-Wei Yang, Prasaanth Radhakrishnan, Sagnik Chowdhury, Huyen Nguyen
Computer Engineering Department
San José State University (SJSU)
San José, CA, USA
Email: {kai-wei.yang, prasaanth.radhakrishnan, sagnik.chowdhury, huyen.nguyen}@sjsu.edu

***Abstract*—Convolutional neural networks (CNNs) have revolutionized many areas that were previously challenging to solve with traditional programming methods. One such area that shows great potential is super-resolution, which involves improving the quality of low-resolution images. Super-resolution techniques have many practical applications that include medical and scientific imaging and also have applications in multimedia and entertainment. Due to the substantial amount of visual data transmitted over the internet in real-time, compression algorithms have become a necessity. Super-resolution has the potential to function as a useful method of compression.**

Video streaming is widely used globally, but they often suffer from poor video quality due to variations in internet speed and connection quality. This can significantly detract from the user experience. While upgrading to a better internet connection may not be immediately possible for all users, real-time resolution upscaling could potentially mitigate these issues. The chat application could enhance the quality of images by utilizing AI-based techniques to improve the resolution. Furthermore, the model could identify the internet speed and modify the resolution as necessary, allowing low-level hardware devices to provide a superior chat experience.

I. Introduction

Because of the pandemic, many events and meetings have become virtualized among enterprises and universities. Therefore many video chat applications, including Zoom, Teams, and WhatsApp, have prevailed in the world to achieve remote communication for users worldwide. With the growth of communication applications, the demand for the Internet is required. To provide the users with faster data transmission speed and quality of service for users, the internet bandwidth where the data size is an important factor in the video chat application user experience. The Internet bandwidth strongly depends on the strength of the Internet signal and the speed of the Internet. Presently, the coverage of the internet grid and the speed of the internet differ from area to area. It is better to adjust the data size based on the internet bandwidth to achieve faster transmission speed. From the video chat application side, super-resolution can be used to improve image quality for the end user. Our research is to test various super-resolution techniques and report on their strengths and weaknesses, and we will prototype a simple video chat application based on our findings.

A. Subject

Due to the shift towards remote work, many individuals have had to rely on video chat applications for work meetings and personal communication, as well as streaming platforms that are used for entertainment. The rise of real-time video communication with the growth of video conferencing, online education, telemedicine, and cloud games has become more prevalent due to the advancement of the internet. Companies are now competing to attract the most customers by providing a seamless user experience. To achieve this, some companies like YouTube use Dynamic Adaptive Streaming, which adjusts the bitrates based on network bandwidth to ensure smooth streaming. [1]

B. Purpose

The success of a video chat application depends on multiple factors, such as the user interface, video quality, accessibility, network conditions, video codecs, and the protocol utilized. Each of these elements can have a notable influence on user experience, although video quality is the most important aspect. Poor algorithms can greatly affect the overall quality of the video, leading to dissatisfaction among users. The aim of this study is to examine these factors individually and provide an analysis of the most optimal combination. Although there are various algorithms available for enhancing video quality, like ABR and SR, there are many limitations to them. Identifying a suitable method for these challenges could significantly improve the Quality of Experience for users.

C. Project Focus

The primary focus of this paper is to explore Super-Resolution Algorithms, and how they can be integrated into a video streaming pipeline. The analysis will be centered around popular chat and streaming applications and will aim to understand why they offer superior user experiences compared to their competitors. This analysis will take into account various factors such as overall application analysis, AI and Deep-Learning models, network protocols, and Video Coding. The paper will highlight the importance of having a well-functioning application and address the challenges faced by the industry, along with potential solutions to overcome them.

D. Scope

"Video traffic accounts for over 70 percent of all downstream internet traffic and this number is expected to grow" [2]. Video streaming companies utilize various algorithms and deep learning models which quickly enhance the quality of

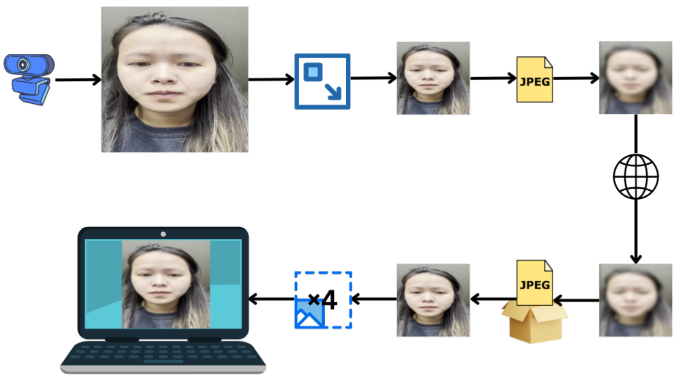


Figure 1: Our video streaming pipeline from camera to display

images and videos. The proposed model in this paper aims to improve existing technologies, specifically when tested under different network conditions. It is important to consider various factors in this process as customers are always seeking the best services and content. Choosing appropriate techniques can increase the number of viewers or subscribers and result in greater revenue for content providers. As electronic devices' usage continues to grow, it is essential to consistently enhance and stay competitive. A video chat application that provides a good Quality Experience can improve meeting settings, boost productivity, and benefit businesses that use online streaming platforms.

II. Project Architecture

A. Overview

Our prototype video chat application will use a peer-to-peer (P2P) TCP connection for streaming content between two clients. A server will be used to help with establishing the connection between the clients. The displayed image on the client side will be enhanced using the chosen super-resolution model. Because video chat applications can be used on such a broad array of devices we are looking into a variety of models that offer a range of trade-offs between speed and quality.

B. Streaming Pipeline

We are keeping our pipeline as simple as possible in order to reduce any extraneous factors that might impact our results. Each client will capture a frame from their camera, downscale it to 1/4 the size required by the receiver, and apply Jpeg compression to further reduce the size. The compressed data will be transmitted over the TCP connection. On the receiving end, the data will be decoded and upscaled by a factor of 4 using the chosen enhancement technique.

C. Handshake Server

The handshake server exists to simplify the process of connecting with another user. In order to establish a TCP connection between two nodes one node needs to be a server while the other becomes a client. The server then waits for the client to connect to them. A P2P connection requires both parties to coordinate so that each party knows what role to

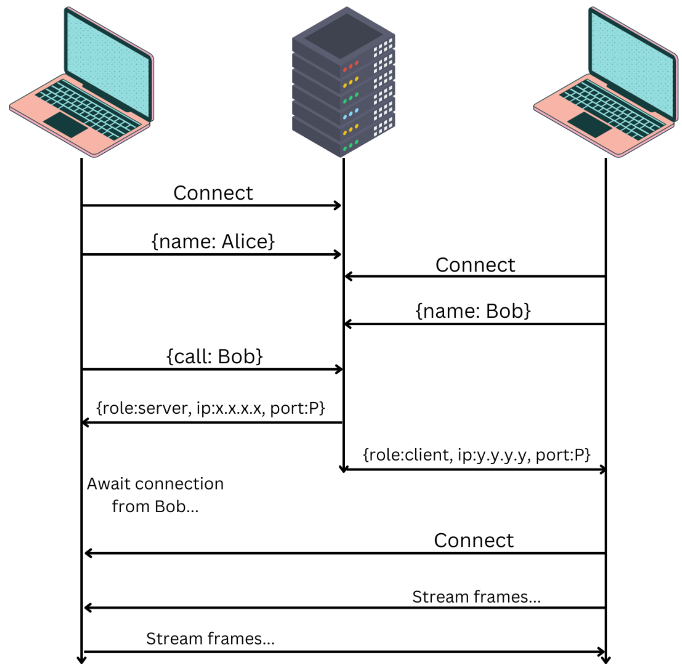


Figure 2: Our simple handshake protocol for establishing the P2P connection between two users

play. This is not possible if the two parties have no other means of communicating with each other prior to establishing the connection.

To solve this problem the handshake server acts as a moderator between users. Both parties connect to the handshake server as clients. Both clients can then provide a human-friendly identifier, such as a name, which allows them to find each other from a list of 'active' users. When a user wants to call another user, the server selects a common port number for both users, exchanges the IP addresses of the two users, and assigns the role of 'server' to the caller, and the role of 'client' to the 'callee'.

The caller then starts their server and waits for the callee to connect to them. The callee connects using the received IP and port information. The caller can additionally use the callee's IP address to filter out any other connections that may come in.

In a practical implementation, the server would have a secure database of users, and several network security measures in place, but that is beyond the scope of our prototype.

D. Super Resolution Models

Super-Resolution (SR) models are generally used to generate a higher-resolution image or video from a lower-resolution version. Multi-frame video super-resolution (VSR) is an approach that uses lower-generation video frames to generate a higher-resolution video frame and hence enhance the quality of the video. This is done by using neighboring frames and their information to enhance lower-quality videos. "Multi-frame video super-resolution(VSR) aims to restore a high-resolution video from both its corresponding low-resolution frame and multiple neighboring frames, in order to make

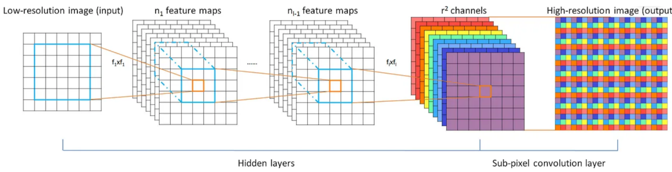


Figure 3: The network structure of ESPCN

full use of the inter-frame information" [3]. Various deep-learning algorithms are used for this but the most popularly used ones are called Generative Adversarial Networks (GAN). GAN mainly "consists of a generator and a discriminator" [4] network to generate high-quality frames or images. In regards to Image super-resolution, there exist two techniques namely Single-Image "Super-Resolution (SISR) and Multi-Image Super-Resolution (MISR)" [4]. There has been a vast improvement in these models over the years but there is still a long way to go as there exist multiple technologies and the best one needs to be used and researched more thoroughly to implement this more frequently. The following models have been utilized to analyze the application's functionality and perform benchmarking tasks. Below are some of the models we are evaluating. Since our use case is live camera streams our access to adjacent frames is limited, therefore we will be limiting our scope to single-image techniques.

1) ESPCN: ESPCN stands for "Enhanced Super-Resolution Convolutional Network." It is a deep learning architecture specifically designed for the task of single-image super-resolution. The ESPCN model utilizes convolutional neural networks (CNNs) "to learn the nonlinear correlation mapping between low-resolution and high-resolution video frame patches" [5]. It works in a patch-based manner, where small patches from the low-resolution image are used as input to predict the corresponding high-resolution patches. By using the hierarchical features learned by the CNNs, ESPCN can recover minute details and enhance the resolution of the output image.

"The input of the ESPCN network is a low-resolution image, and features are extracted through three convolutional layers to obtain a feature image with the channel number r^2 and the same image size as the input image" [6]. The network structure can be seen in Figure 3.

2) ESRGAN: "Generative Adversarial Network (GAN) is a type of neural network architecture consisting of two parts: a generator and a discriminator" [3] "ESRGAN further improves the restored Image quality of SRGAN" [7]. ESRGAN incorporates a perceptual loss function, which measures the perceptual similarity between the generated and target images, to guide the training process. "ESRGAN does not use batch normalization" [7].

3) RCAN: The RCAN (Residual Channel Attention Networks) model is a "deep learning-based single-image super-resolution restoration (SRR) method" [5] that was introduced in a research paper published in 2018 [8]. It utilizes residual learning and channel attention mechanisms. The authors propose the use of a residual in residual (RIR) structure for the architecture. The RIR structure contains a number of residual groups (RGs), each containing multiple residual channel attention blocks (RCABs). Furthermore, the RGs use long skip connections, whereas the RCABs use short skip connections. The outputs of the RCABs are summed with the original input to form the residual connection, which allows the model to capture high-frequency information from the original image.

4) SRResNet: "This model has similar deep structure to SRGAN but does not have the discriminator. It is a model in which we deprive the adversarial components of SRGAN. Similar to SRGAN, the generator attempts to produce an image that is close to the ground truth" [9]. SRResNet employs a deep convolutional neural network with skip connections, allowing the network to learn residual information between low-resolution and high-resolution image patches.

III. System Design

One of the requirements of compression algorithms used in video streaming is that they can adapt to changing network bandwidth conditions. However, most SISR models only support discrete scale factors, which makes them unsuitable, on their own, as an adaptive compression system. "Continuous-scale SR, which aims to use a single model to process arbitrary (integer or non-integer) scale factors, is still a challenging task" [10]. Furthermore, as our tests have shown, traditional image compression algorithms do a much better job of reducing the size without losing high-frequency details.

Thus, our compression pipeline will combine a single discrete downsampling of the input image with traditional Jpeg compression. When the network bandwidth cannot handle a high level of Jpeg compression the image will be downsampled by a factor of 4 prior to Jpeg encoding and transmission. This reduces the size of the image data by a factor of 16. On the receiving end, if the decoded image has a low resolution, the system will upscale it by a factor of 4 using the super-resolution technique supported on that device. If no super-resolution is supported, it will simply apply bicubic upsampling.

IV. Evaluation Methodology

Our goal is to evaluate a number of different super-resolution techniques on a variety of GPU and CPU devices so that we can select the appropriate model based on the capabilities of the device our application will be running on. We will measure the performance using the elapsed time from when the super-resolution workload is dispatched to when it is completed. We will also measure the quality of each technique by measuring the "peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM)" [5] scores on a short video. We will use a video in order to keep our measurements fair and consistent across all test environments.

PSNR is a measurement used to assess the quality of a reconstructed or compressed signal compared to the original signal. "Among the existing methods, PSNR for its simplicity is the most known and widely used measure. The PSNR of a video sequence can be calculated either as an average of individual frames' PSNR or can be based on the average of individual frames' distortions" [11]. We will use the former approach. A higher PSNR value indicates less distortion and better quality. It is commonly used to evaluate image and video compression algorithms, providing a numerical measure of how well the compressed version resembles the original. Higher PSNR values indicate better quality and less noticeable artifacts in the compressed signal.

SSIM is a metric used to evaluate the similarity between two images. It takes into consideration not only the differences between individual pixels but also the structural patterns, luminance, contrast, and texture information. "The SSIM evaluation method of traditional image calculates the SSIM index of the local block through the sliding window in a distorted image, and finally obtains the SSIM evaluation value of the entire image by normalizing all the local block evaluation indicators" [12]. The SSIM index ranges from 0 to 1, with values closer to 1 indicating higher similarity and better image quality. We will use the mean SSIM of individual frames as the final score for the video.

Each frame of the video will be downsampled to 128x128, and used as input to a 4x super-resolution model, resulting in a 512x512 output. For the ground truth, we will resize the original frame to 512x512 to match the dimensions of the generated output. The PSNR and SSIM scores will be evaluated based on the generated output and the resized ground truth images.

However, our objective is not only to accurately reconstruct the original image but to produce an image that is visually appealing to the end user. So, we will also make subjective judgments on the visual appeal of the outputs from each model, regardless of how accurate they are. For the subjective assessment, we will use live camera footage in order to judge the quality of the output under more realistic circumstances. We will consider two key factors.

The first factor is the perceived clarity of the image, that is to say, the sharpness of edges and high-frequency details. This is the most important factor, in our opinion, because blurry or muddy edges are perceived as being of lower quality. Since in the final application, the user will have no ground truth to compare against, the illusion of higher quality is more important than the accurate reconstruction of the image.

The second factor is temporal stability. Since we are looking exclusively at single-image techniques, our models are at risk of producing inconsistent outputs between consecutive frames. This can lead to an unpleasant flickering effect even if the frame-to-frame inconsistency is small.

We will grade each model on these two factors on a scale of 1 to 10.

V. Results

A. Assessment of Output Quality

Table I shows the PSNR and SSIM scores achieved by our selected models as well as for simple bicubic upsampling. As we can see, the PSNR scores are largely very similar for most of the models, and bicubic upsampling seems to outperform many of the SR models. This result is surprising, as visual inspection of the output (Figure 5) shows a clear difference in quality across the various techniques. The SSIM scores appear to be much more representative of the actual quality of the generated outputs. Hence, we will focus on the SSIM score in our final assessment.

Model	PSNR	SSIM
Bicubic	25.822	0.702
ESPCN	25.892	0.701
SRResNet	25.461	0.721
ESRGAN	25.851	0.71
RCAN	25.677	0.737

Table I: Comparison of Measured Quality

Model	Clarity	Temporal Stability
Bicubic	3	10
ESPCN	5	10
SRResNet	7	7
ESRGAN	6	9
RCAN	8	8

Table II: Our subjective assessment of the models on a scale of 1-10

Table II shows our subjective grades for each SR technique, including bicubic upsampling. As we can see every model has some strengths and weaknesses.

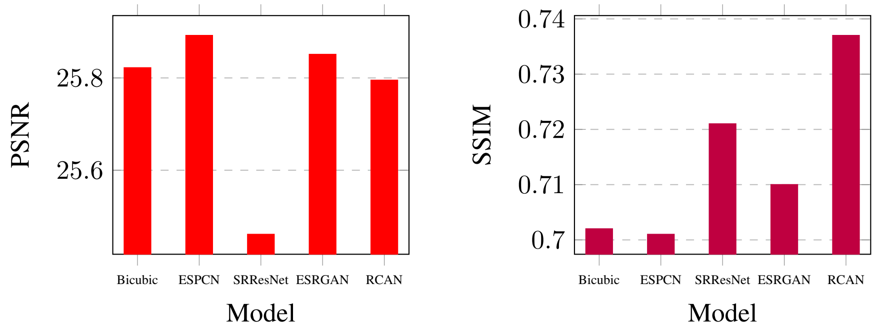
In terms of image clarity, we rated bicubic upsampling the lowest since we can clearly see pixelation artifacts along high-contrast edges, and the image is quite muddy in regions with high-frequency details. ESPCN is a significant improvement as it does a good job smoothing out the pixelation artifacts, however, the resulting image is still quite blurry, and it struggles with recovering high-frequency details. ESRGAN produced a sharper image compared to ESPCN but still struggled with high-frequency details, such as hair strands. SRResNet and RCAN both produced very crisp and clear images, however, the output from SRResNet had a more synthetic look to it, so we gave RCAN a higher grade.

Bicubic upsampling and ESPCN both had extremely stable outputs with no visible flickering of any sort. ESRGAN was nearly as good, but some minor jittering could be seen in high-contrast areas. RCAN had slightly more visible instability compared to ESRGAN, but was still quite usable. SRResNet produced the most jarring instabilities, especially in regions with high-frequency details, where certain lines would warp from frame to frame. This warping effect likely also contributed to the synthetic look of its output as previously mentioned.

We think that RCAN produced the best results overall, as it had the most crisp output with an acceptable level of temporal instability.

B. Performance Benchmarks

Tables III and IV show the performance benchmarks on several CUDA-enabled GPUs and several CPUs respectively. Interestingly, these results do not indicate a straightforward correlation between hardware spec and performance. Looking



(a) PSNR of models

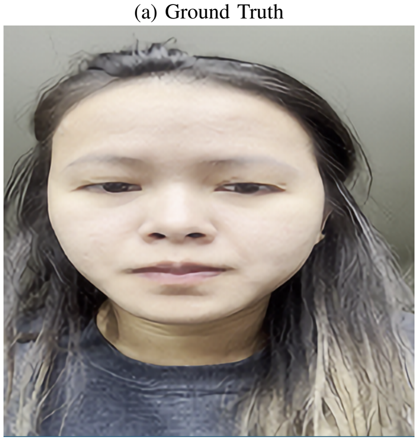
(b) SSIM of models

Figure 4: Super Resolution Quality Comparison

(a) Ground Truth

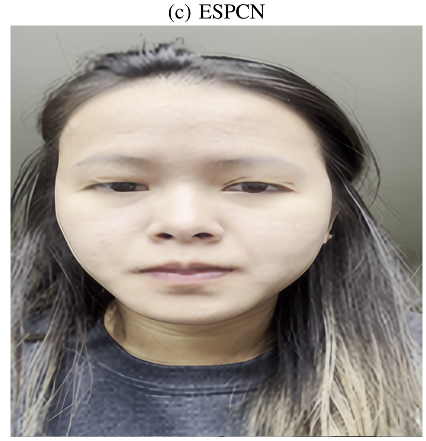
(b) Bicubic

(c) ESPCN



(d) SRResNet

(e) ESRGAN



(f) RCAN

Figure 5: Visual comparison of the result from each model

Model	RTX 3080	RTX 2080 Ti	GTX 1060
ESPCN	0.0391	0.0376	0.0140
SRResNet	0.0401	0.0441	0.0183
ESRGAN	0.1101	0.1372	0.0922
RCAN	0.1244	0.1739	0.1310

Table III: Evaluation time (in seconds) on various CUDA-enabled GPUs

Model	Intel Core i5 1.4 GHz	Intel Xeon 3.6 GHz
ESPCN	0.0120	0.0195
SRResNet	0.7918	0.9416
ESRGAN	0.7320	0.7568
RCAN	2.8208	3.2479

Table IV: Evaluation time (in seconds) on various CPU hardware

at figures 6a and 6b we see that certain models may perform better on weaker devices.

Naturally, there are various factors besides the processor that affect the resulting performance, such as the memory bandwidth and caches, as well as how well the model's code

Quality	Size (bytes)	PSNR	SSIM
100	110010.52	47.910548	0.990402544
95	44649.44	44.39635322	0.979030016
90	29358.62	43.38208614	0.974286942
85	23148.75	42.94004699	0.974393202
80	18948.99	42.22797621	0.969291413
75	15461.27	42.25610693	0.967825580
70	14049.53	42.08512243	0.965785578
65	12932.72	41.25732254	0.959294827
60	12158.78	41.02974747	0.962076474
55	11504.08	40.76484524	0.959217297

Table V: JPEG Compression Results

itself is optimized. However, it is clear that a GPU is required in order to use most of the models. In order to have a pleasant experience we need to achieve a frame rate of at least 24Hz, which gives us a budget of 0.04 seconds to execute the SR technique. Only ESPCN achieved that target on the CPU devices that we tested with.

An interesting observation is that ESPCN when run on the CPU actually outperformed some of the high-end GPUs. This could be due to the fact that its architecture is so simple that it overshadows the cost of transferring the batch data to GPU memory. This is a promising result, as it means that we could achieve significant performance gains by running the model on an integrated GPU.

Conversely, SRResNet ran nearly as fast as ESPCN on GPUs, and significantly faster than ESRGAN and RCAN, but was incredibly slow on CPUs, even more so than ESRGAN. It is very likely that with a properly optimized implementation using a vendor-agnostic API such as Vulkan, SRResNet could achieve the performance goal even on relatively low-end GPU devices.

C. *JPEG Compression*

We ran some tests to compare the compression factor from downsampling the image against using traditional compression algorithms. Since our architecture is based on SISR techniques we chose Jpeg as our image compression algorithm, instead of a video codec. We used the same test video that we used to evaluate the models and measured the size of the encoded data as well as PSNR and SSIM scores of the subsequently decoded image against the original. We repeated this test for a range of compression qualities.

Table V shows the data size in bytes, PSNR, and SSIM scores corresponding to the encoding quality used. From figure 7 we can see that the data size decreases with quality, as do the PSNR and SSIM, however, both scores are still significantly higher than those achieved by the SR techniques, even for the lowest quality that we tested.

As for size, the original 512x512 input is 768KB. And the downsampled 128x128 image is 48KB. In comparison, the encoded Jpeg size at 95% quality is 44KB, already smaller than the downsampled image. Thus downsampling alone is not an effective strategy for image compression and must be used alongside other traditional compression methods for best results.

D. *Front End features*

For the front end, users could utilize the ports of video streaming and audio on the server to chat with other users. After the users connect to the server, they could choose to join the chatting room with or without the super-resolution recovery. If the users choose the option with super-resolution, the server would output the processed user data to other users. In contrast, if the users choose to join without super-resolution, the server would output the original data to the chat room. This would allow the user to choose the optimal option based on varying network bandwidths and conditions.

VI. Related Works

Zoom is considered an advanced video chat application that employs AI to deliver an enhanced video conference experience. Zoom's technology architecture is patented and is now a cloud-based technology. It avoids using peer-to-peer (P2P) technology because it is unreliable for networks with different speeds.

A. *Zoom*

The compression technique used by Zoom can recognize important parts of an image and enhance its quality while leaving out irrelevant sections. This technology is also used in the "Blur Background" feature which highlights only the person's face. This feature is particularly helpful when the user has low internet speed.

"Zoom—an innovative videoconferencing platform—has a number of unique features that enhance its potential appeal to qualitative and mixed-methods researchers" [13]. This is a well-known application used for video conferencing and is a suitable application to compare with. "Its relative ease of use, cost-effectiveness, data management features, and security options" [13] are some of the features why this application is used globally. It has recently gained traction during the COVID pandemic to conduct class lectures and even company meetings.

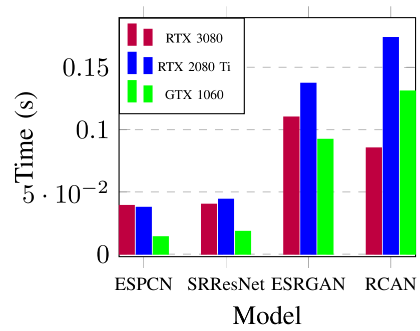
B. *Deep Learning*

Deep Learning uses multiple computational layers to learn and train models. Neural networks within this technique identify crucial features of datasets to attain the best results. To accomplish this, methods like eliminating noise from images, implementing CNNs, and enhancing LR inputs into HR outputs are used.

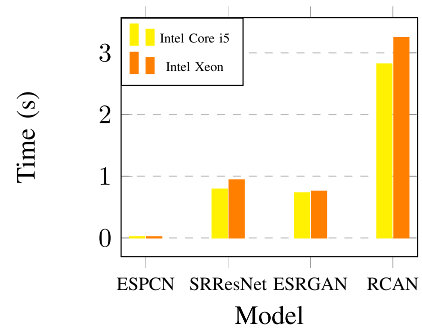
Deep learning has had a huge impact on image generation and networks giving very high-quality outputs. "These tools have recently reached a great deal of popularity in computer vision and computer graphics applications, such as motion transfer, creation of video portraits, deep fakes, image-to-image translation etc" [14].

C. *Generative Adversarial Network*

"Generative Adversarial Network (GAN) is a type of neural network architecture consisting of two parts: a generator and a discriminator. The generator creates new synthetic data"

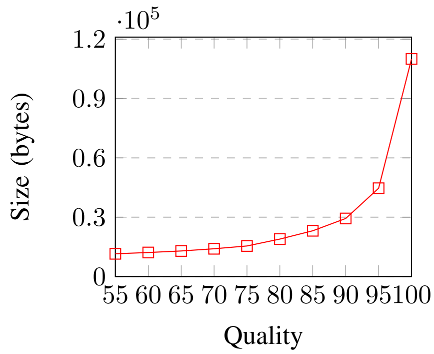


(a) Evaluation Time on CUDA-enabled GPUs

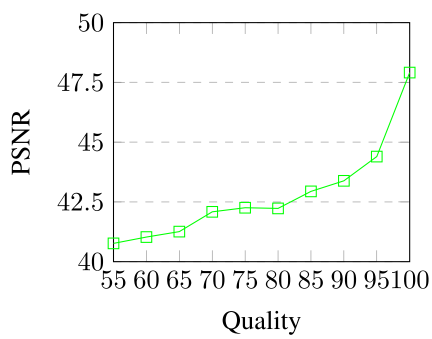


(b) Evaluation Time on CPUs

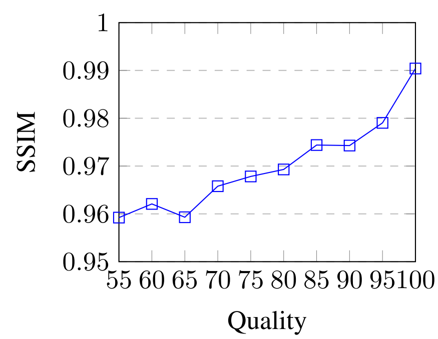
Figure 6: Super Resolution Performance Comparison



(a) Size vs Quality



(b) PSNR vs Quality



(c) SSIM vs Quality

Figure 7: JPEG Compression Results

[4] within a given domain, while the discriminator classifies examples as either real or fake. GANs are widely used in image and video generation tasks and have shown promising results in a variety of applications.

VII. Conclusion

A. Application Implementation

The section outlines the benefits and shortcomings of the application developed and its implementation.

1) Advantages of the implementation: The implementation of this application offers several benefits to both users and service/content providers. One such advantage is that users are more likely to continue using or switch to this application if they have an improved quality of experience (QoE) while video chatting or streaming. This, in turn, can boost the user's productivity and generate higher revenue for the companies. By using this implementation, even though a GPU is used they can get improved video qualities in low network areas.

Running the implementation system on the server side offers several advantages for determining the bit rate of a user. Many applications today use HTTP-based streaming and chat based on ABR, which means that implementing this system would require minimal changes to the applications. Data may need to be logged and saved if computations based on that data are necessary. This system's ABR algorithms are based on observations of past decisions across many video streaming experiments. Super Resolution and DL techniques can also enhance the video further when the ABR fails to deliver the best video quality due to low bandwidth. As all the processing occurs on the client side, there is a reduction in bandwidth usage, and the incoming video stream remains unaffected. Additionally, the added computation for determining the compression level can increase the customer's QoE.

It can improve the streaming quality on demand by the customer and not need the use of GPU when it's not requested by the consumer. This can be particularly useful in situations where internet bandwidth is limited, and the streaming service is unable to provide high-quality video. By enhancing the video quality, super-resolution can provide a more immersive viewing experience, reducing viewer frustration and increasing engagement.

2) Disadvantages of the implementation: While the proposed application comes with several benefits, it also has some downsides. One of the disadvantages of the Super Resolution system is that all training is done locally to save bandwidth, which may not be the most appropriate approach. By training Super Resolution algorithms on the network, it can receive faster access to the real-world network conditions that we use on a daily basis. Nevertheless, despite constant updates, it would still be slower than real-time training. Such training can also limit available bandwidth, so it is necessary to determine the appropriate balance between local and online training. In addition, there is an increase in the computational overhead for the client.

Furthermore, the application may encounter some general issues. The hardware's computation capability may be necessary for running SR to enhance video quality, and the improvement in Quality will depend on system capabilities. Additionally, the software designed may occasionally not get back the original image or video that has been downscaled, which could cause SR to malfunction, which can cause unprecedented blur and flicker in the video stream.

Another disadvantage is the need for a GPU with high performance as Super Resolution is a computation-intensive algorithm and we need to test further with different GPUs to get consistent results and learn how to reduce flickering and improve qualities of high-frequency areas in the video stream if it is downscaled.

B. Future Work

This article emphasizes the need for research on various types of hardware devices to support different AI systems. It is suggested that running multiple simulations and tests on devices that have varying hardware capabilities can significantly improve the system and provide better results. Testing with other codecs, ABR, SR, CNN-based image compression techniques, and n/w architectures can also provide better testing results and provide insights on how the system can be improved. It is crucial to strike a balance between additional encoding and storage requirements that come with ABR and SR, which can be achieved through further testing. Releasing the product in a beta stage and allowing users to test it would help identify any issues, and the developers can make necessary changes based on user logs. Furthermore, we need a push for deep learning frameworks that are platform agnostic so that a broader array of devices, including mobile, can benefit from GPU acceleration.

References

[1] Eilwoo Baik et al. "VSync: Cloud based video streaming service for mobile devices". In: *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*. 2016, pp. 1–9. DOI: 10.1109/INFOCOM.2016.7524567.

[2] Zaixi Shang et al. "Study of the Subjective and Objective Quality of High Motion Live Streaming Videos". In: *IEEE Transactions on Image Processing* 31 (2022), pp. 1027–1041. DOI: 10.1109/TIP.2021.3136723.

[3] Bin Yang, Juhao Jiang, and Guannan Chen. "Multi-frame Video Super-resolution Based on Efficient and Parallel Network". In: *2022 IEEE 2nd International Conference on Software Engineering and Artificial Intelligence (SEAI)*. 2022, pp. 69–73. DOI: 10.1109/SEAI55746.2022.9832294.

[4] Wenming Yang et al. "Deep Learning for Single Image Super-Resolution: A Brief Review". In: *IEEE Transactions on Multimedia* 21.12 (Dec. 2019), pp. 3106–3121. DOI: 10.1109/tmm.2019.2919431. URL: https://doi.org/10.1109%2Ftmm.2019.2919431.

[5] Xuelong Li et al. "A multi-frame image super-resolution method". In: *Signal Processing* 90 (Feb. 2010), pp. 405–414. DOI: 10.1016/j.sigpro.2009.05.028.

[6] Jingyi Du, Qingli Liu, and Kang Chen. "License plate super-resolution reconstruction based on improved ESPCN network". In: *2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*. 2019, pp. 122–125. DOI: 10.1109/IMCEC46724.2019.8984050.

[7] Chengkun Song et al. "Low Resolution Face Recognition System Based on ESRGAN". In: *2021 3rd International Conference on Applied Machine Learning (ICAML)*. 2021, pp. 76–79. DOI: 10.1109/ICAML54311.2021.00024.

[8] Yulun Zhang et al. *Image Super-Resolution Using Very Deep Residual Channel Attention Networks*. 2018. arXiv: 1807.02758 [cs.CV].

[9] Takahiro Shindo et al. "Super Resolution for QR Code Images". In: *2022 IEEE 11th Global Conference on Consumer Electronics (GCCE)*. 2022, pp. 274–277. DOI: 10.1109/GCCE56475.2022.10014154.

[10] Hanlin Wu, Ning Ni, and Libao Zhang. *Scale-Aware Dynamic Network for Continuous-Scale Super-Resolution*. 2021. arXiv: 2110.15655 [cs.CV].

[11] A. T. Nasrabadi et al. "Investigating the PSNR calculation methods for video sequences with source and channel distortions". In: *2014 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting*. 2014, pp. 1–4. DOI: 10.1109/BMSB.2014.6873482.

[12] Yufeng Zhou et al. "Weighted-to-Spherically-Uniform SSIM Objective Quality Evaluation for Panoramic Video". In: *2018 14th IEEE International Conference on Signal Processing (ICSP)*. 2018, pp. 54–57. DOI: 10.1109/ICSP.2018.8652269.

[13] Mandy M. Archibald et al. "Using Zoom Videoconferencing for Qualitative Data Collection: Perceptions and Experiences of Researchers and Participants". In: *International Journal of Qualitative Methods* 18 (2019), p. 1609406919874596.

[14] Goluck Konuko, Giuseppe Valenzise, and Stéphane Lathuilière. "Ultra-Low Bitrate Video Conferencing Using Deep Image Animation". In: *2022 IEEE International Conference on Image Processing (ICIP)*. 2022, pp. 3515–3520. DOI: 10.1109/ICIP46576.2022.9897526.