Finetuning Variational Autoencoders for the Restoration of Sports Video Highlights

Michael Doboli
Stanford University
Palo Alto, CA
mdoboli@stanford.edu

Siddharth Sharma
Stanford University
Palo Alto, CA
sidshr@stanford.edu

Abstract

As time has passed, cameras and video recorders have improved with quality. For this reason, past video footage may not necessarily be the most pleasant for modern viewers due to outdated quality and less capable technology. This is particularly relevant for sports data, since present-day viewers may not be able to fully enjoy old footage of their favorite athletes, matches, and sporting triumphs. Our work shows that it is possible to enhance old sports highlights by finetuning an existing variational autoencoder framework for a specific category of sports video data. We perform data-based finetuning, hyperparameter optimization, and postprocessing with the goal of various visual features being crisper and more visible. Extra care is taken to ensure that important features such as ball location or player movement is not obscured by our model. We use Wan et. al's publicly available pre-trained model as a foundation and baseline for our frame-by-frame restoration work of tennis video data from the Wimbledon Championships. [14]. Our results show both quantitative improvements via the FID score metric and qualitative enhancement of sports video data.

1. Introduction

Advances in Computer Vision have opened the possibility of re-constructing older images at a higher resolution. This is an especially salient problem given the number of old images that requires immediate restoration or risk being lost forever. Image restorations are used to not only improve the resolution of old images [8], but can also be used to remove scratches and physical aberrations to photos [14].

A field where image restoration techniques can be highly beneficial is old sports recordings. Namely, sports higlights from the 1990s and early 2000s are quite poor by modern standards, as seen in **Figure 1**. For modern viewers, such sports highlights can seem unwatchable, especially in sports such as tennis where low resolution can obscure crucial features such as ball location. This situation is quite unfortunate, especially given how low-resolution videos can often contain some of the most exciting and important moments in sports history.

In the world of professional sports, where billions of dollars are spent in sponsorships and contracts, the disparity in quality between footage taken only 20 or 30 years ago and now is quite stark. The occlusions and blurry movements can be irksome for modern viewers, and ultimately cause these highlights to not be as watched as more modern videos. This can ultimately lead to older videos being watched less and less over time, and ultimately forgotten. To prevent this from happening, we use Variational Autoencoders (VAEs) to restore past highlights with higher quality. We approach this problem by splitting videos into frames, using a finetuned model on each frame, and then applying postprocessing video normalization methods (sharpening and denoising) to ensure consistent and better quality. We then produce the restored video by stitching together the restored frames with the fps of the original video. To evaluate our model, we used both numerical metrics such as Frechet Inception Distance (FID) scores and qualitative analysis to determine which methodologies are most effective at restoring old sports videos. Our results show that our finetuning methods perform superior to the baseline general-purpose restoration model while maintaining consistency across various combinations of hyperparameters. We also demonstrate additional quantitative improvements via postprocessing methods.

2. Related Work

There has been much research on both removing scratches and improving image resolution. Since most sports highlights tend to not have physical aberrations, we consider literature that only focuses on super-resolution and

Figure 1. ATP 250 Basel 2001 Tennis Tournament. Extracted from https://www.youtube.com/watch?v=mK0Oq5_-O-Y

correcting image quality, shape, and discoloration. The first methods used to solve this were interpolation methods such as bicubic or Lanczos filtering [3] that aim to sharpen or smoothen certain features. Such interpolation methods often fail to capture the full complexity of the image [8], meaning that they are often incomplete for the purposes of generating realistic restored images. While interpolation methods were popular prior to the advent of deep learning methods, modern insights can do significantly better in solving this problem.

Deep Convolutional Neural Networks are one tool used to restore images that are realistic by human standards. Unlike interpolation or other geometric-based methods, Convolutional Neural Network (CNN)-based approaches leverage the training data in order to create a model that can specialize in restoring images within the problem space. Dong et al. developed on of the first such approaches in SRCNN, a deep convolutional network that learns mappings between low and high resolution images [2]. SCRNN showed that super-resolution could be achieved using deep neural network methods, and other work improved upon SRCNN by increasing training rate and generalizing to handle multiple upscale factors [7]. These methods first upscale an image from low resolution to a higher resolution using some sort of interpolation method (i.e., bicubic interpolation) before training a CNN upon this upscaled image. Shi et. al instead perform convolution on the low resolution image directly to learn low-level features before applying an interpolation method that is trained for the particular dataset . [13] This removes the need for more general filtering methods such as bicubic interpolation and achieves faster performance by avoiding this intermediate step. Altogether, CNN-based approaches have shown success in generating realistic images.

An important observation has been that theoretical loss functions may not align perfectly with visual similarity. Since any model is trained to minimize its loss function, choosing a loss function that aligns with visual intuition is of upmost importance. Johnson et al. note that humans often correlate high-level features with an image's overall quality, thus meaning that pixel-based loss functions may not be optimal [6]. Thus, rather than minimizing pixel-to-pixel loss, they instead optimize perceptual loss from a pretrained loss network, which generates more aesthetically pleasing results. Bruna et al. take a more statistical-minded approach to this observation by showing that features generated by Convolutional Networks have less variance and more resistance to deformation than pixel-based data when performing conditional probability-based image restoration [1]. Since generating realistic images is the top priority, the choice of loss function can play a role in shaping the ultimate appearance of a restored image.

Convolutional networks generally function by taking a low resolution image, learning important features using convolution, and using these learned features to generate realistic restored images. A fundamentally different way of examining this problem is to instead consider it as a problem of learning a mapping from one domain to another. For problems of this kind, such as style transfer [9] and image inpainting (filling missing pixels so that the resulting image looks realistic) [15], generative adversarial networks (GANs) have been used to learn domain mappings. GANs train both a generator that can map between two different domains and a discriminator that determines if the mapping can produce results indistinguishable from the desired output. Ledig et al. use GANs in a two-step process: a generator network first creates high-resolution images, and then a discriminator differentiates the generated super-resolution image from a high resolution image [8]. Generative models can provide state-of-the-art results that look more realistic than convolutional networks alone [8], and are hence ultimately what we choose to finetune. The specific model which we finetune is Wan et al., since it is made for the specific purpose of restoring old images (as opposed to the more general task of super-resolution) [14].

3. Methods

3.1. Variational Autoencoders

As discussed in the literature view, we seek to finetune variational autoencoders (VAEs) to restore sports highlights. Similar to a generative adversarial network, a variational autoencoder employs regularized encodings during the training in order to ensure an effective latent space. VAEs use two main networks for training: an encoder and decoder. The encoder is the network that produces the "new features" representation from the "old features" representation while the decoder network conducts the reverse process [11]. The goal of an autoencoder model is to find the best encoder-decoder pair, namely one that maximizes the encoded information while minimizing the reconstruction

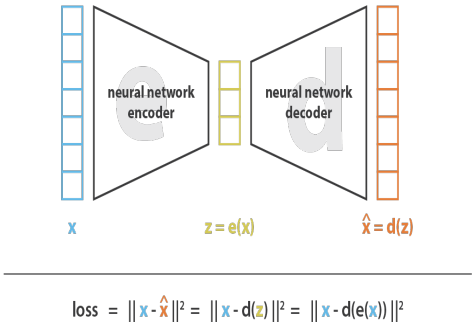

Figure 2. Variational Autoencoder. Extracted from https://towardsdatascience.com/understanding-variational-autoencoders-vaes-f70510919f73

error when decoding. This best encoding-decoding pair is learned through an iterative optimization process such as gradient descent. The diagram in **Figure 2** illustrates the high level intuition behind an autoencoder.

A variational autoencoder is simply an autoencoder where the loss function is composed of a reconstruction term (that makes the encoding-decoding scheme efficient) and a regularisation term. These enable optimal generations of new data. The loss function is given below:

$$\begin{aligned} \text{loss} &= ||x - \hat{x}||^2 + KL[N(\mu_x, \sigma_x), N(0, I)] \\ &= ||x - d(z)||^2 + KL[N(\mu_x, \sigma_x), N(0, I)] \end{aligned}$$

As seen in this equation, the regularization is given by the Kulback-Leibler (KL) divergence between the returned distribution and a standard Gaussian. It also penalizes a difference between a generated image and a real image, which is reasonable considering the goal is to make the generated and real image as similar as possible. We will now begin a discussion of the methods used for our work. In this project, we explicitly finetune existing VAE models to best suit the restoration of sports highlights. We mainly build off the work of Wan et al. as a foundation for our project [14]. This work specifically uses two variational autoencoders, one to transform old (degraded) images and one to transform clean (restored) images into two latent spaces. This domain gap between the two latent spaces is then minimized via the training of an adversarial discriminative model. For faces in particular, the authors of this work also apply a face refinement network.

3.2. Architecture

The key insight of Wan et al. is that when using generative models, the difference between old images and synthetically generated old images can be quite large. For example, **Figure 3** and **Figure 4** it is clear how a synthesized old

Figure 3. Real Old Picture. Extracted from https://www.youtube.com/watch?v=Q5bhszQq9eA

Figure 4. Synthesized Old Picture. Extracted from https://www.youtube.com/watch?v=Q5bhszQq9eA

image can lack features such as scratches, discolorations, or other physical damage. Even though a new image can be converted to grayscale, such an image can differ greatly from a true old image.

Since synthesized and real old pictures can differ significantly, the solution is to map the two pictures onto a common latent space. Since this latent space will have reduced dimensionality from the original, finding an accurate mapping is an easier problem to solve than the original. The structure of the different mappings can be seen in **Figure 5**.

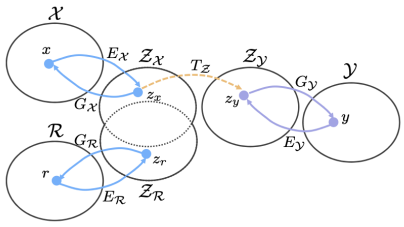

Figure 5. Mapping of Latent Spaces. $\mathcal{R}$ represents the real old image space, $\mathbb{X}$ represents the space of synthesized old images, and $\mathcal{Y}$ represents the space of real modern images. $\mathcal{Z}_{\mathcal{X}}, \mathcal{Z}_{\mathcal{R}}, \mathcal{Z}_{\mathcal{Y}}$ are the corresponding latent spaces. Extracted from [14]

The model goes through three general steps:

Step 1 - Encoding: In the first step, the model encodes the space of real old images $\mathcal{R}$ and the space of synthetically generated images $\mathcal{X}$ into the latent spaces $\mathcal{Z}_{\mathcal{R}}$ and $\mathcal{Z}_{\mathcal{X}}$, respectively using the same Variational Autoencoder

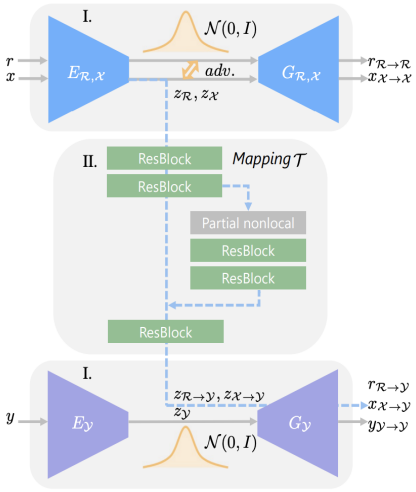

Figure 6. Two Variational Autoencoders and mapping between them. $E_{\mathcal{R},\mathcal{X}}$ and $G_{\mathcal{R},\mathcal{X}}$ is the encoder-generator pair for the VAE that maps the old frames to a latent space. $E_{\mathcal{Y}}$ and $G_{\mathcal{Y}}$ is the encoder-generator pair for the VAE that maps output images to a latent space. The mapping $\mathcal{T}$ has a partial nonlocal step that can perform specialized tasks such as scratch removal and face detection. Extracted from [14]

(VAE). The reasoning behind using the same VAE is that we ultimately aim for $\mathcal{Z}_{\mathcal{X}}$ to be as close as possible to $\mathcal{Z}_{\mathcal{R}}$. We denote this first VAE as VAE_1. During this same step, a second VAE, VAE_2, is used to encode the space of restored images into a separate latent space $\mathcal{Z}_{\mathcal{Y}}$. Both VAE_1 and VAE_2 are trained with a similar loss function that penalizes the latent space for diverging from a Gaussian prior. Furthermore, the model also contains a Least Squares Adversarial Network (LSGAN) loss term that provides a further boost for the two latent spaces to match. The LSGAN loss function has been shown to produce higher quality images [10], and in this case provides further reassurance that $\mathcal{Z}_{\mathcal{X}}$ and $\mathcal{Z}_{\mathcal{R}}$ will indeed be similar.

Step 2 - Mapping: Once the synthetic and real old images are aligned to a common latent space, the next step is to map this latent space onto the latent space of real images $\mathcal{Z}_{\mathcal{Y}}$. We can make the assumption that $\mathcal{Z}_{\mathcal{R}} \approx \mathcal{Z}_{\mathcal{X}}$, so we train the mapping $\tau : \mathcal{Z}_{\mathcal{X}} \to \mathcal{Z}_{\mathcal{X}}$ and assume that it will generalize to $\mathcal{Z}_{\mathcal{R}}$ as an input space. $\mathcal{T}$ also benefits from the fact that $\mathcal{Z}_{\mathcal{X}}$ is a synthesized version of $\mathcal{Z}_{\mathcal{Y}}$. In other words, we can pair images $\{x, y\}$ where $x \in \mathcal{Z}_{\mathcal{X}}$ and $y \in \mathcal{Z}_{\mathcal{Y}}$ such that x is a synthesized version of y. Hence, having $\mathcal{T}$ learn the mapping from x to y is a feasible task given that x itself is the result of a mapping applied to y. The loss function for the mapping between these two latent spaces is

$$\mathcal{L}_{\mathcal{T}}(x, y) = \lambda \mathcal{L}_{\mathcal{T},\ell} + \mathcal{L}_{\mathcal{L},\text{GAN}} + \lambda' \mathcal{L}_{\text{FM}}$$

where λ, λ' are learnable parameters, $\mathcal{L}_{\mathcal{T},\ell} = \mathbb{E}\|\mathcal{T}(z_x) - z_y\|$ is the L1 loss distance between where z_x is mapped and z_y, $\mathcal{L}_{\mathcal{L},\text{GAN}}$ is a Least Squares GAN loss term, and $\mathcal{L}_{\text{FM}}$ is a loss function that measures how closely the Adversial Least Squares GAN term is following a pre-trained VGG network (this term is used solely to ensure stability while training).

Step 3 - Decoding: Upon learning the mapping $\mathcal{T}$, the model must decode the latent space representation into a restored image. During the encoding step we used the encoder $E_{\mathcal{Y}}$ to encode from a real image to the latent space; in the decoding step we step we use the corresponding generator $G_{\mathcal{Y}}$ to transform an element of the latent space into a restored image.

3.3. Finetuning Methods

We present a finetuned pipeline of generative methods, based on Wan et al.'s baseline model [14], for the task of sports video restoration. Note that Wan et al.'s work was used as the foundation for our finetuning and all methods in this section extended this existing codebase. We divide our finetuning methods into three classes of work: data-based finetuning, hyperparameter optimization, and postprocessing.

For the data finetuning step, we use our extracted dataset of images to run three explicit steps: the training of the domain A domain a variational autoencoder (the mapping from old photos to a latent space), the training of the domain B variational autoencoder (the mapping from modern photos to a latent space), and the training of mappings (mapping the two latent spaces together.) We choose to use a VAE framework over vanilla autoencoders due to the denser latent representation provided by VAEs [11]. Using synthetic image pairs $\{x, y\}$, we map the latent space to generalize the restorations. Note that we did not use the face enhancement network that was provided in the original architecture since faces are not key features in our model. We also trained the mappings without the specific steps for scratch detection. This is due to the fact that scratches are not relevant occlusions for tennis highlights.

As part of the finetuning for our custom tennis dataset, we also perform hyperparameter optimization. We specifically manipulate parameters such as the learning rate (α), batch size, and weight initialization. The baseline model of Wan et al. [14] specifically uses Adam as its optimizer and we kept this as a constant given Adam is the current state-of-the-art for computing adaptive learning updates for each parameter.

Beyond hyperparameter optimization, we also employ

post processing methods to enhance our results. We specifically perform image sharpening as well as Gaussian denoising to enhance the quality of lines and other key edges present in tennis footage. For image sharpening, we apply a 2D filter based on a given kernel and for Gaussian denoising, we use `opencv`'s bilateral filter [5]. This technique reduces noise while maintaining edges, thus giving our restorations a smoother look. For this step of post-processing the restorations, we also considered other techniques such as Sobel image derivatives, Laplacians, contrast and brightness scaling, and histogram equalization.

3.4. Dataset

For the general problem of sports footage restoration, we chose to demonstrate the capabilities of finetuning specifically for tennis. Different sports can look quite different, and even within sports different videos could be significantly different (for example, different tennis court colors), so in this work we finetune our models based on a consistent Wimbledon tennis video dataset. We chose tennis for four reasons: camera angles are relatively consistent, the crowd and background image features are out of the main frame during play, data for long matches is publicly available for extraction, and important artifacts such as the ball and player's rackets make it an interesting problem. We also chose to procure our tennis data from the Wimbledon Championships tournament since all players are mandated to dress in the same white color and court surfaces are consistent (a green color for grass). This helps to reduce noise and enables the model to focus on key features such as the players, ball, court, etc.

To finetune the domain-based approach of Wan et al. [14], we constructed three datasets: one containing non-synthesized old images in RGB format, one containing non-synthesized old images in grayscale, and one containing modern sports images in RGB format. Wan et al.'s original paper uses the Pascal VOC dataset since it is a commonly used dataset for object detection [14]; however, we simply sample JPEG images for each. Each of our three training datasets contains approximately 2,000 frames (corresponding to 80 seconds) of tennis match footage from the Wimbledon championships. In this case, we extracted data from two matches: our modern match data came from Roger Federer versus Novak Djokovic (Wimbledon Final 2019) and our older mach data came from Pete Sampras versus Jim Courier (Wimbledon Final 1993). To attain the grayscaled old images, we grayscaled each image from the RGB old images folder and created a new directory. Examples of images in the training data are provided in **Figure 7** and **Figure 8**:

Our validation dataset is composed of 300 old RGB

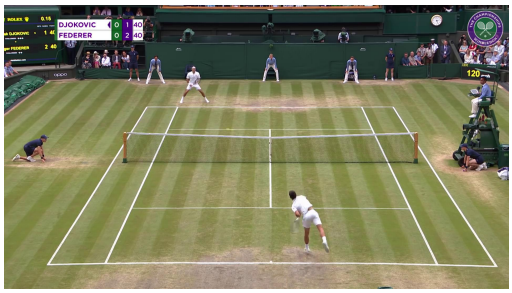

Figure 7. Modern RGB Image (Groundtruth). Extracted from https://www.youtube.com/watch?v=TUikJi0Qhhw

Figure 8. Old RGB Image. Extracted from https://www.youtube.com/watch?v=4hOSgdC5WbI

images, specifically extracted from Roger Federer versus Yevgeny Kafelnikov (Wimbledon First Round 2000). To test our various finetuned models, we used a dataset of approximately 300 frames from Rafael Nadal versus Nick Krygios (Wimbledon Fourth Round 2019). As part of our testing procedure, we chose to perform restorations on blurred frames of modern footage (synthetic data). This enables us to match the workflow used by Wan et al.

4. Experiments

For our experiments, we created multiple models with different hyperparameters for the same three training datasets (modern RGB images, old RGB images, old grayscaled images). Our experimental procedure consists of the following steps given the three assembled datasets:

- Train the Domain A VAE
- Train the Domain B VAE
- Train the mappings using the Domain A and Domain B training results
- Perform inference (restorations) using the checkpoints from the three training steps

For each step of the training process, it was possible to apply various optimizations and manually adjust learning parameters. As referenced in the methods section above,

we exclusively modified the weight initialization technique, learning rate, and batch size while keeping all data consistent. The specific combination of hyperparameters was determined based on deviation from the default. The baseline method of Wan et al. [14] used a learning rate of $2 \cdot 10^{-4}$, a batch size of 100, and Xavier weight initialization. For our experiments, we maintained batch sizes of 32 and below since the AWS instance and Google Colaboratory environments that we used did not have the CUDA memory to manage experiments with batch sizes greater than 32. In fact, when we trained the domain mappings with a batch size of 32, we ran into CUDA memory errors and had to adjust to a batch size of 16 just for this step. For this reason, the main batch size we used was 16 and for all trainings of the mappings, we used a batch size of 16. All experiment combinations are detailed below:

Experiment	Learning Rate	Batch Size	Weights
1	$2 \cdot 10^{-4}$	16	Xavier
2	$2 \cdot 10^{-4}$	16	Kaiming
3	$2 \cdot 10^{-3}$	16	Xavier
4	$2 \cdot 10^{-2}$	16	Kaiming
5	$2 \cdot 10^{-5}$	16	Xavier
6	$2 \cdot 10^{-3}$	16	Kaiming
7	$2 \cdot 10^{-3}$	16	Kaiming
8	$2 \cdot 10^{-4}$	32	Xavier

As seen in the table above, batch size refers to the number of examples in each batch seen by the model during the training of the domain A VAE and the domain B VAE. The learning rates recorded above were applied during all three steps. Following the training of models, we applied the postprocessing techniques that were discussed in the finetuning methods section above. To perform a comparison to the baseline method and detect which finetuned models perform best for the task of restoration, we used Frechet Inception Distance (FID), a metric that enables calculation of the distance between feature vectors for groundtruth images and generated images. Developed by Heusel et al. [4], FID estimates quality of generated images using the Inception v3 model. We chose to use FID due to its well-known correlation with human judgement of visual quality; lower FID scores correspond to higher quality generations. Using the `PyTorch-FID` module [12], we calculate FID scores as follows:

$$d^2 = ||\mu_1 - \mu_2||^2 + Tr(C_1 + C_2 - 2 * \sqrt{C_1 * C_2}) \quad (1)$$

For our work, we computed FID scores on two datasets: restorations of synthetically blurred modern images and the original modern image. We specifically used Gaussian blur to simulate blurring of the groundtruth RGB modern images.

Figure 9. Output from domain B training

5. Results

Although our model is meant to restore videos of sports, it ultimately learns to restore individual frames of sports videos before stitching those frames together. Hence, all training and evaluation occurs on the frame-level, where videos are used primarily to produce a final output.

5.1. Training

When finetuning, the main hyperparameters that we adjusted were learning rate, batch size, and weight initialization. An Adam optimizer was used on 25 epochs, and the three different training steps (training the two VAEs and mapping) took around 6.5 hours to run on each experiment. The choice of 25 epochs was meant to accommodate the enhanced dataset of 2000 training images – the full 200 epochs that Wan et al. used would have been computationally infeasible for running more than one experiment.

While training, the model outputted results from its domain A, domain B, and mappings training. We present an example output of the domain B training visualization in Figure 9, where the highlighted regions indicate that VAE learns high-level features such as line-placement and player position. Given the relatively well-ordered structure of tennis data, the model learning these high-level features is crucial, since ultimately it shows the model is effective at mapping the input image to the latent space.

5.2. Evaluation

To evaluate the model, a novel 300 image dataset was chosen from the 2021 Wimbeldon Championships. Since the size of our training dataset had 2000 images, the validation set size of 300 was picked to approximate the standard 9:1 ratio of train to validation set size. Each element in the validation set was distorted with a Gaussian Blur in order to approximate an old sports higlight. This Gaussian Blur was necessary since no dataset currently exists that pairs an old tennis video with a restored version of it, so the Gaussian Blur was intended to approximate an old tennis video. Our model was then applied on this blurred image, and using the `pytorch-fid` library [12] we computed the FID

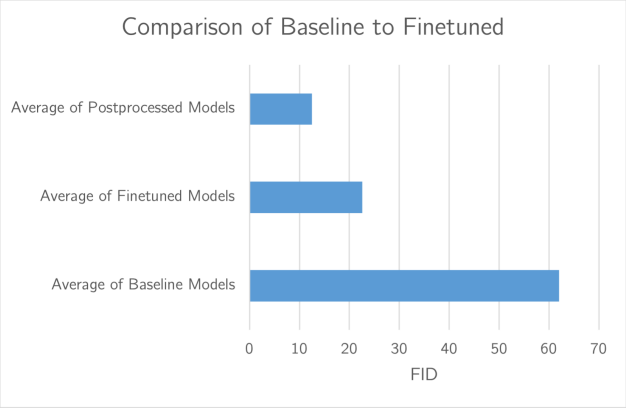

Figure 10. Finetuned, Baseline, Finetuned Processed Results

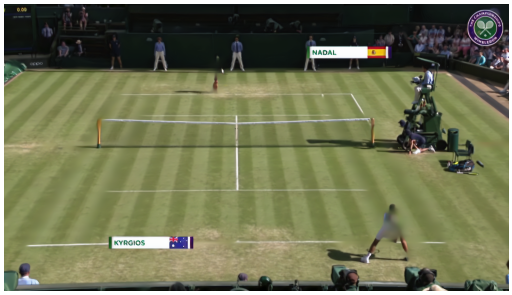

Figure 11. Baseline output

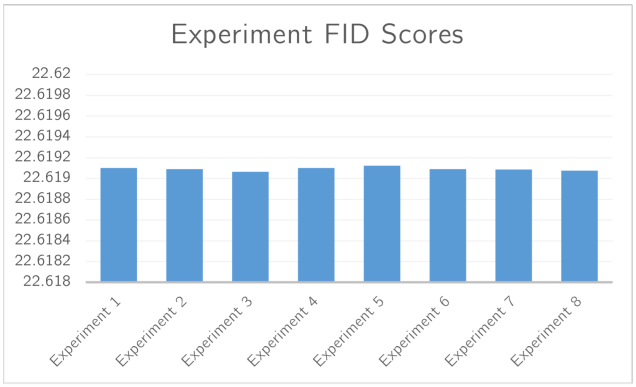

Figure 12. FID Scores for Base Finetuned Models

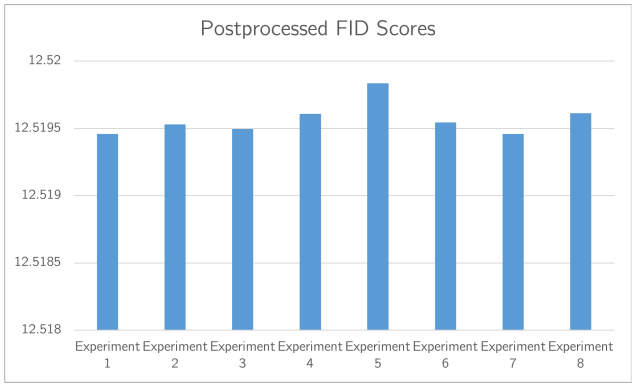

Figure 13. FID Scores for Postprocessed Finetuned Models

score between the restored image and the original. This FID score reflects how well the model was at restoring high-level visual features of the image; a lower FID score indicates more similarity between the restored image and the original image.

The primary result from the finetuning process is that the average of finetuned models performed signficantly better than the baseline model. This is likely since the baseline model is meant to perform general image enhancements such as scratch detection and physical distortion. We also hypothesize that the baseline model performs general enhancements due to the fact that it is trained on a large-scale Pascal VOC object detection dataset that contains a wide variety of environments. Since the tennis dataset featured none of these more broad attributes, the baseline model was too general to properly solve this specific problem. The specific deficits can be observed in **Figure 11**, where it is clear that the baseline model distorts key features such as players and lines.

The finetuned model however had surprisingly small sensitivity to the different training parameters. **Figure 12** displays this consistency among different parameters. Kaiming weight initialization generally performed marginally better than Xavier weight initialization, which is reasonable since the Convolutional Layers within the VAE used ReLU activation functions instead of Sigmoid activation functions [14]. A larger batch size also showed a potential correlation with lower FID score, but this was not explored futher due to CUDA memory constraints.

The main conclusion that can be derived from the model's consistency across parameters is that all combinations of hyperparameters generally led to similar model outcomes. This is potentially a unique aspect of our model, since it shows that the model is robust to poor hyperparameter initialization. The model could have also suffered from overfitting due to the training set being only one match and Wimbledon matches generally having variation in court color and lighting. One way to ameliorate this problem would be to diversify the training set with different matches or perhaps even different court surfaces. However, given that adding any more images would have made the training dataset computationally intractable (in fact we needed to make modifications to the existing codebase to accommodate a $2,000$ image training dataset) and we wished to finetune on the Wimbledon dataset, we did not pursue this route. Actions that were taken to prevent overfitting were regularization terms and a variety of different court angles in its training set. We also observe that our dataset may

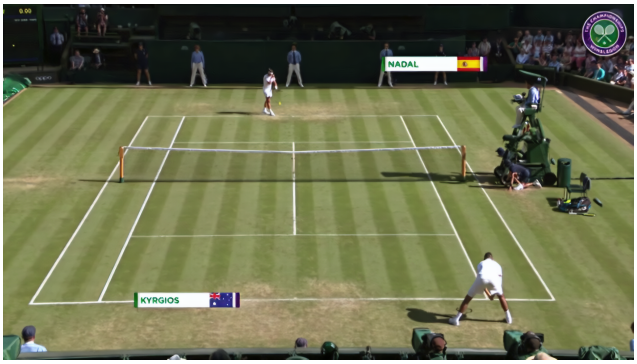

Figure 14. Restored image (same frame as **Figure 11**)

be prone to producing very similar FID scores due to all the images being taken from a single court (Center Court at Wimbledon) for our training data. Hence, all restoration techniques produced fairly similar adjustments and hence FID scores.

As mentioned in our finetuning methods section, we also performed a layer of postprocessing on our validation set generations. This was accomplished by the application of a kernel filter for sharpening and Gaussian denoising (smoothing). Based on the FID scores indicated in **Figure 13**, we show that our novel postprocessing steps have stronger similarity with the groundtruth images (due to consistently lower FID scores), thus further improving our restoration work. We show a single frame example for our finetuned restoration in **Figure 14**.

Compared to **Figure 11** (the baseline's restoration of the same frame), we see that our restoration has improved visual quality, stronger edges, and clearer artifacts. Thus, given these qualitative visually improved restorations and quantitative FID improvements, we are confident that our model is well-suited for tasks of sports image restoration.

6. Conclusion

In this work, we establish that finetuning VAEs can be used to restore old sports highlights compared to previous baseline results. With additional postprocessing techniques, videos can achieve even lower FID scores than using a finetuned model alone. These same techniques can be applied to other court types and perhaps even other sports. These techniques can be used to restore old sports highlights for modern audiences to appreciate.

Some extensions of this project could be to add more training epochs, diversify and enlarge the training set, as well as experimenting with restoration of different sports. Given further compute power, these tasks would be a natural corollary of the finetuning work we accomplished. We believe this work can provide value to both sports-watching communities as well as machine learning researchers and practitioners given our improved restorations and finetuning of a deep-learning framework for sports video finetuning.

7. Contributions & Acknowledgements

We used the publicly available code at https://github.com/microsoft/Bringing-Old-Photos-Back-to-Life for our baseline model. Siddharth's work focused more on data preprocessing, loading models, getting the initial training scripts to run, and preparing experiments. Michael's work focused more on postprocessing the results, creating scripts and folders to run experiments on AWS, and computing metrics such as FID. We both equally contributed to this paper.

References

[1] Joan Bruna, Pablo Sprechmann, and Yann LeCun. Super-resolution with deep convolutional sufficient statistics. *arXiv preprint arXiv:1511.05666*, 2015. 2

[2] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):295–307, 2015. 2

[3] Claude E Duchon. Lanczos filtering in one and two dimensions. *Journal of Applied Meteorology and Climatology*, 18(8):1016–1022, 1979. 2

[4] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 6

[5] Itseez. Open source computer vision library. https://github.com/itseez/opencv, 2015. 5

[6] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 2

[7] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1646–1654, 2016. 2

[8] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 1, 2

[9] Chuan Li and Michael Wand. Combining markov random fields and convolutional neural networks for image synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2479–2486, 2016. 2

[10] Xudong Mao, Qing Li, Haoran Xie, Raymond Y.K. Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 4

[11] Joseph Rocca. Understanding variational autoencoders (vaes), Mar 2021. 2, 4

[12] Maximilian Seitzer. pytorch-fid: FID Score for PyTorch. https://github.com/mseitzer/pytorch-fid, August 2020. Version 0.2.1. 6

[13] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016. 2

[14] Ziyu Wan, Bo Zhang, Dongdong Chen, Pan Zhang, Dong Chen, Jing Liao, and Fang Wen. Bringing old photos back to life. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2747–2757, 2020. 1, 2, 3, 4, 5, 6, 7

[15] Raymond Yeh, Chen Chen, Teck Yian Lim, Mark Hasegawa-Johnson, and Minh N Do. Semantic image inpainting with perceptual and contextual losses. *arXiv preprint arXiv:1607.07539*, 2(3), 2016. 2