Image Translation Between Video Games and Movies

May 2021

Word Count: 1499

1 Introduction

The aim of this report is to investigate the application of deep learning for image-to-image translation to the domains of video games and movies.

2 Preprocessing

The data provided for this task were a video file from the video game "Mafia" and three shorter video clips from mafia related movies and series: "The Irishman", "The Godfather (1-3)" and "The Sopranos". In total, 6000 frames were extracted, 3000 originating from the game while the rest was equally extracted from the three movie files (1000 each). This was achieved by dividing the files in equal timesteps to acquire samples from the entire videos. The resolution of these frames was initially 1280×720 pixels which is too large to use in deep learning models and was reduced to 256×256 as the training time would have been exponentially high. Moreover, the images were cropped to remove the parts were most subtitles lie improving performance and reducing artifacts.
While the test set could have been extracted from our training data by randomly shuffling and sampling our dataset, as the video files were short, many of the extracted frames pictured the same scene with similar character details. Thus, we also hold the end portion of each video for the testing dataset to test our model on data that it would not have seen before.

3 Frame-to-Frame Model

For this project we implemented and adapted CycleGAN [6]. CycleGAN implements both a ResNet and a U-Net architecture for the generator. ResNet is generally recommended due to their ability to extract image representations

[2]. U-Net showed output results similar to each input without much variation or change, while ResNet exhibited promising results with pronounced changes. It also had more artifacts present, shown in the images below.
Training and testing the models took place in Google Colab using the Tesla P100-PCIE-16GB GPU. Models were trained for 50 epochs of mini-batches. Learning rate was reduced when the loss did not significantly improve for 5 epochs. The ResNet architecture used included 9 blocks as per the original CycleGAN paper recommended for images of size 256×256.
We also utilise identity images, which are images translated to their original domain and cycle images, which are images translated to another domain and then translated back to their original domain. As the identity images had a high loss during training, the identity lambda was set to 10, increasing the influence of the identity images to the overall generator loss. Lastly, we use Batch Normalisation instead of Instance as better results were observed with it.

Figure 1: U-Net Images: Original, Identity, Translated, Cycle

The U-Net model failed to make significant changes within 50 epochs, however, the ResNet model while making more changes, introduced more artifacts as can be seen from Figure 2.

Figure 2: ResNet Images: Original, Identity, Translated, Cycle

In our results, some chessboard artifacts are present. The model also introduced artifacts in light sources as there is a difference between the ambient lighting that games use, compared to the more direct lighting conditions of the movies. It also performs worse in tasks with variable illumination, commonly found in darker images.
ResNet was faster to train at an average of 5.14 minutes per epoch while U-Net took 6.23 minutes.

4 Face-to-Face Models

4.1 Face Extraction

We apply Multi-Task Convolutional Neural Network (MTCNN) to the extracted frames from Section 2. MTCNN is a popular face and landmark detection architecture with multiple pre-trained models available online [5]. Using PyTroch's model, we pass the original size frames (1280×720), after which, if a face of at least 100 pixels is detected, the algorithm will crop the image around that face leaving a margin of 50 pixels. This is because the model is good enough to detect faces in the background of an image which when resized to the size of 256×256, become pixelated. In total, this process extracted 315 and 355 game and movie faces respectively.
While the number of available samples is low, extracting more faces would mean getting the same face from the same scene multiple times in our training dataset which could lead to overfitting of the data. Instead, we apply data augmentation techniques to assist the model with dealing with the low sample count. After the faces have been extracted, they are saved in h5 format for faster I/O. Similarly to before, the testing set is generated from the last part of each video which the model would not have seen during training.

4.2 Face-To-Face Model

As ResNet showed a greater number of changes, we use that architecture for our generator. The discriminator used before is a 5-layer patch discriminator implemented from the CycleGAN architecture. However, as the patterns for faces will be similar for both games and movies, the smaller details are more important to create better translated images. For that reason, we create a new discriminator using both the Patch Discriminator and a Pixel Discriminator. Thus, the performance is based on larger features which the patch discriminator should be able to detect, and on smaller features, handled by the pixel discriminator. As the memory requirements are very high during the use of two discriminators and training failed due to memory errors, we combine the two in a single discriminator named Branch Discriminator. While we are now able to use two discriminators, a disadvantage of this approach is that to compare the two, we need to reduce the output of the pixel discriminator using a maxpool and thus, lose some details that it provides. Below is a diagram of our proposed discriminator architecture.

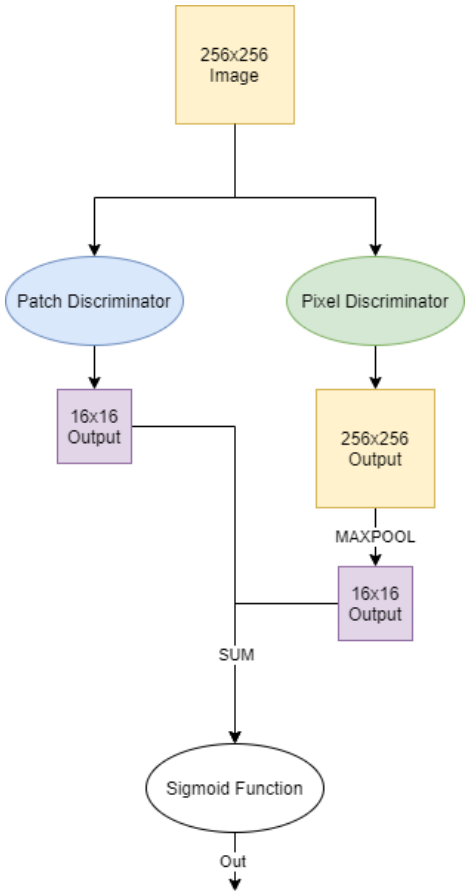

Figure 3: Branch Discriminator Architecture

Furthermore, for the Face-To-Face model, better results were shown through the use of a fixed learning rate of 0.0002 while the number of mini-batches in an epoch was smaller due to the smaller dataset used. The learning rate of the discriminator was also reduced to 2×10^{-5}. Below are the resulting images.

Figure 4: ResNet-9 Faces: Original, Identity, Translated, Cycle

The results showcase the ability of the model to adapt the domain of an image between video game and movie, by changing lighting, shadows, colours and contrast accordingly. Moreover, movie footage converted to game footage exhibit less detailed features while the reverse is true for video game faces converted to movie faces. Moreover, the identity and cycle images closely resemble the original faces.

4.3 Data Augmentation

As the outputs of the model have been capturing the colour palette of the movie and game domains well and have adjusted the lighting in both, we are mainly interested to assist the network better learn structural features between games and movies. Thus, random rotation and random horizontal flip were used so the model can ideally learn better feature representations. No colour augmentation was used such as colour jitter, as we are mainly interested in improving the landmark features of the images.

Figure 5: Images Trained with Augmentation

While the performance is similar to the model without augmented data, this model is also able to better handle faces at different angles and orientations as can be seen by the results presented above. Due to the larger Branch Discriminator, training took longer at an average of 6.47 minutes per epoch.

4.4 Realism

There is a strong dominance of male characters both in the video game and movie footage. Thus, the network struggles to translate images of female faces. Similarly, there are more older people in both datasets and thus, the model exhibits better results when there are wrinkles and other such characteristics present. Examining the translated game faces, sharp edges near the jaw and nose are different from the more rounded features present in the movie faces and are an obvious sign of the game footage.

5 Real World Application

5.1 Original Footage

Translating frames using the face-to-face model introduces some artifacts on areas that the model has not seen before. Cropping the faces generates better results, where faces look similar between frames, even when posture changes. Still, the lack of features in the video game faces makes them unrealistic. The final running time of this process was 4.73

5.1.1 Improving Results

As the image sizes needed would be larger than the 256×256 we propose translating the images in a resized representation and then using a subsequent neural network, to increase the resolution of the image. This can be achieved by using a Super-Resolution-GAN (SRGAN) which has showcased optimal results [3]. Moreover, a network that only focuses on one direction could be utilised that does not require any auxiliary discriminators or generators. Models such as FastCUT while more complicated can achieve better results than CycleGAN in reduced training time [4].

5.2 Pipeline Model

Another improvement on converting video games to videos is by using parts of the rendering pipeline. By having access to the 3D meshes, we can train a network to construct these in a more realistic way. This can be done without annotating the images as the input to the network would be a movie image we wish to reconstruct, and the network would generate the necessary meshes and lighting conditions to approximate it. Thus, we then compare the output of the network with the original image by a simple $L1$ loss function to get a measure of how close the output is to the input.
Similarly to this proposed architecture, NVIDIA utilises high-resolution images to train their DLSS architecture to produce better tessellation with less computing resources by training a neural network to fill in missing information

[1]. Thus, we can use and ideal movie-like image to teach a neural net to best compose the necessary vectors to approximate it.

References

[1] Andrew Burnes. *NVIDIA DLSS 2.0: A Big Leap In AI Rendering*. 2020. URL: https://www.nvidia.com/en-us/geforce/news/nvidia-dlss-2-0-a-big-leap-in-ai-rendering/.

[2] Xiaohan Jin, Ye Qi, and Shangxuan Wu. *CycleGAN Face-off*. 2018. arXiv: 1712.03451 [cs.CV].

[3] Christian Ledig et al. *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network*. 2017. arXiv: 1609.04802 [cs.CV].

[4] Taesung Park et al. *Contrastive Learning for Unpaired Image-to-Image Translation*. 2020. arXiv: 2007.15651 [cs.CV].

[5] Kaipeng Zhang et al. "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks". In: *IEEE Signal Processing Letters* 23.10 (Oct. 2016), pp. 1499–1503. ISSN: 1558-2361. DOI: 10.1109/lsp.2016.2603342. URL: http://dx.doi.org/10.1109/LSP.2016.2603342.

[6] Jun-Yan Zhu et al. *Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks*. 2020. arXiv: 1703.10593 [cs.CV].