FINNiGAN

Frame Interpolation using Generative Adversarial Networks

Authors: Mark Koren, Kunal Menda, Apoorva Sharma

Introduction

Streaming video services are some of the most recognizable brands in technology today. One of the hardest problems in this exciting field is that of frame interpolation.

Frame interpolation, for the purposes of this project, is the action of generating a frame for a video, given the immediate frames occurring sequentially before and after. This allows a video to have its frame rate enhanced, which is a process known as *upsampling*. In the general upsampling task, we cannot assume access to ground truth for the interpolated frames.

High fidelity upsampling can be applied to *video compression*, since could store only key frames of the video, and interpolate at playtime to fill in the missing parts. For compression tasks, the original high frame rate video exists, and thus the ground truth for the interpolated frames is available.

Problem Statement

The goal of this project is to present a network architecture that is capable of taking a video with some specified frame-rate producing a 2x up-sampled version of the video.

With reference to Figure 1, the network will execute this task by first down-sampling the 30 fps video to 15 fps by hiding every other frame, and training itself to best regenerating the hidden frames.

After training, the trained network will then attempt to generate the a frame in-between every two frames of the original 30 fps video, producing a 60 fps output.

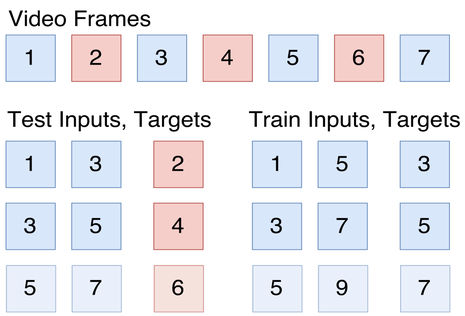

Figure 1 - Example showing how a sequence of frames from a source video can be sorted to create train and test datasets. In each case, the target is the frame temporally between the two input frames.

Results

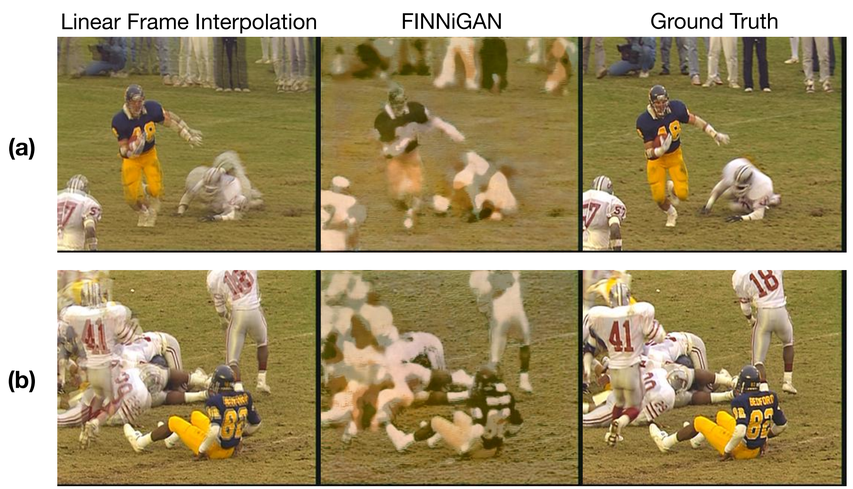

Figure 3 - Frame interpolation on the "football" dataset. Video at http://bit.ly/2rvs57a.

The images generated and shown in Figures 3-4 are **test** examples. A single network was trained to generate the video from which the fames shown in Figure 3 a,b were drawn, and another was trained for those shown in Figure 4 a,b. The proposed method reproduces the overall structure accurately, eliminating ghosting artifacts that appear with typical motion-compensated frame interpolation techniques.

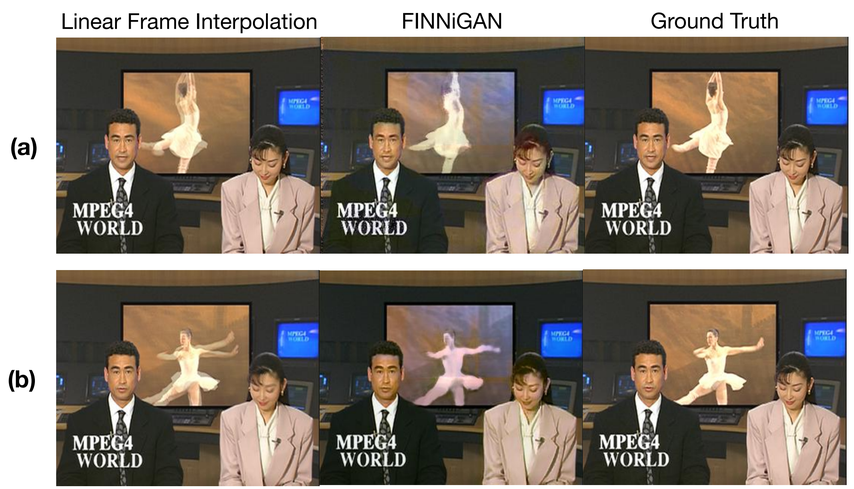

Figure 4 - Frame interpolation on the "news" dataset. Video at http://bit.ly/2rFl2Y3.

Though the network is able to predict the structure of a frame that would lie in between the two inputs, it is easy to see that the generated images lack sharp details in parts of the scene that are moving. Additionally, the network does not yet faithfully reproduce color information. This may be due to the chosen weighting between the MS-SSIM loss and other loss components, as MS-SSIM ignores color dissimilarity.

Network Architecture

We have developed a GAN architecture for the task of frame interpolation. Figure 2 shows this architecture. Note that the generator here is modeled similarly to the U-net encoder/decoder model used by Isola et al [1].

The generator weights are updated to *maximize* the dis_fake_loss, i.e to fool the discriminator as much as possible. Meanwhile, the discriminator weights are updated to *minimize* the sum of both dis_real_loss and dis_fake_loss. The weights of the generator were also updated to minimize the L1 loss and a structural similarity (MS-SSIM) loss between the generated frames and ground truth middle frames. The two generator objectives are combined linearly with a relative weighting hyperparameter.

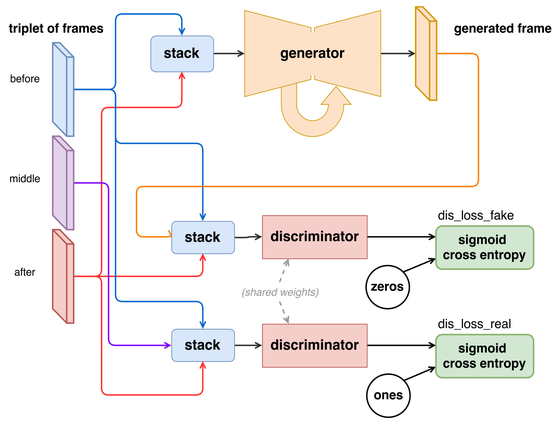

Figure 2 - Network architecture. The generator weights are updated on a combination of maximizing dis_loss_fake and minimizing L1 and MS-SSIM loss between the generated frame and the ground truth middle frame. The discriminator weights are updated to minimize both losses

Conclusion & Future Work

The results of this project show the promise of applying modern machine learning techniques to frame interpolation. Much of the architecture for this project was inspired by work released in the last couple years. As work in this field continues to progress, especially as it relates to high quality, large resolution images, results for this task will improve in kind.

At present, there are some possible changes that are worth exploring. Using a Recurrent Neural Net may allow the network to more easily learn how structures change temporally. Recent work in increasing GAN stability could improve convergence speed and generator accuracy. We also believe that training this generator on a structural loss alone, rather than adversarially, and then applying a pix2pix GAN on the generated frame to improve the realism and fill in the texture.

[1] Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." *arXiv preprint arXiv:1611.07004* (2016).