Beyond Vid2Vid: Few-Shot Video Synthesis for Road Scene Generation

Trenton Chang
Department of Computer Science
Stanford University
tchang97@stanford.edu

Akash Modi
Department of Mechanical Engineering
Stanford University
akmodi@stanford.edu

Roshan Toopal
Department of Mechanical Engineering
Stanford University
rtoopal@stanford.edu

Abstract

Generative Adversarial Networks (GANs) have achieved high performance on image generation, and groundbreaking work is currently being done on video generation, with recent improvements in generated video stability, blurriness, and realism. Building off of work by NVIDIA on Vid2Vid, we attempt a fewshot learning approach to allow the model to generalize to unseen city scenes.

1 Introduction

Generative Adversarial Networks (GANs) have achieved high performance on image, text, and audio generation, and groundbreaking work is currently being done on video generation, with recent improvements in generated video stability, blurriness, and realism. More specifically, NVIDIA has recently used GANs to create Vid2Vid model which takes video input and generates videos that match the input video with the style of videos previously seen in training. [1]

The problem with Vid2Vid is that learning to draw arbitrary city scenes requires retraining the entire network on representative city images, with their respective semantic segmentation maps. This is because Vid2Vid only learns how to draw, for example, a particular type of road surface or tree texture, as opposed to road surfaces and tree textures in general.

Wang et. al. built a novel weight generation network which takes a few example images as input at training time [2]. This network learns to dynamically change the weights of the pretrained generator, allowing the network to create videos of domains only seen through a select few images. However, since this requires the development and tuning of a separate network, this is still rather expensive. We propose an alternate method inspired by neural style transfer: given an input segmentation map on an unseen domain, we minimize some distance metric between the ground-truth image and the generated image.

2 Related Work

Our main inspiration for this project is NVIDIA's Vid2Vid network, which is detailed in Wang et. al. (2018). [1] Recently, they proposed a method for few-shot video synthesis which involved training a separate network to dynamically change the weights of the video generator at test time.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

3 Dataset

We will use Cityscapes, dashcam videos from cars driving through cities, to create our pretrained model, and the KITTI dataset for transfer learning. [3, 4] We thought that this would be ideal, since the distributions of city driving images are not so similar that this learning task would become infeasible, but could have meaningful qualitative differences such as building architecture, landscape, weather conditions, etc. that a transfer learning approach could "adapt" to. Specifically, the Cityscapes videos are from German street scenes in Stuttgart, and the KITTI dataset features scenes in the U.S. [3, 4]

4 Model

As input, at training time, Vid2Vid takes in a list of ground truth image frames and a list of corresponding semantic feature maps. We used NVIDIA's segmentation algorithm, pretrained on Cityscapes, to generate semantic feature maps [5]. The output is a segmentation mask, an optical flow map which serves as a proxy for change from frame-to-frame, and a hallucinated image.

Figure 1: Example segmentation mask (model input) colorized

In a mathematical notation: let $\hat{x}$ be the predicted video sequence, which we condition on segmentation mask $\mathbf{s}$. The key assumption Wang et. al. makes to facilitate video synthesis is a Markov assumption such that

$$p(\hat{x}_1^T \mid \mathbf{s}_1^T) = \prod_{i=1}^{T} (\hat{x}^T \mid \hat{x}_{T-L}^{T-1}, \mathbf{s}_{T-L}^{T-1}) \tag{1}$$

Furthermore, $\hat{x}$ is learned as a function of optical flow, segmentation mask, and previous generated frames. This essentially means that each frame depends only on the segementation mask and the images of the previous L frames.

During training, Wang et. al. adopts a coarse-to-fine approach. For example, to generate (1024 x 512) images, separate downsampled (512 x 256) and (256 x 128) images are also created as input, and a generator is trained for all of these scales. [1] The outputs of the smaller networks are concatenated with a selected layer in the larger-scale network, forming a U-net-like architecture.

We use the same architecture as Vid2Vid but modify the training scheme significantly for our few-shot approach, specifically setting up a transfer learning problem. We impose some differentiable metric J between $\hat{h}^T$ and ground-truth image x^T, so our problem becomes

$$\underset{\theta}{argmin}\ J(G(\mathbf{s}_{T-L}^{T-1}), x^T) \tag{2}$$

where θ are the parameters of G. L is a hyperparameter that represents how many frames into the past we look back. Using backpropagation, we update the weights of G.

4.1 Methods

The key problem is the development of a loss function. This function must not only accurately convey the subjective metric of image similarity, but also be continuous and differentiable almost everywhere and satisfy the triangle inequality. To this end, we test on average mean squared error (MSE) and structural similarity loss (SSIM) [6]. Separately, we applied a neural style transfer to the output of Nvidia's model, using a frame of the generated video as the content image, and using a frame of our recorded video on Stanford's campus as the style image. The output of the neural style transfer network was then recorded.

Once we selected a loss function to optimize, the model was trained via the ADAM optimizer with default parameters $\beta_1 = 0.9$ and β_2 = 0.999, testing on models trained at 10, 50. For structural similarity loss, we found the default setting of using an 11 x 11 kernel to yield the best results. We kept the default setting of "looking" 3 frames (0.1s on 30 FPS video) into the past. For all other settings, we retained the same settings specified by Wang et. al. at `https://github.com/NVIDIA/vid2vid`.

Our results, however, were still not very high quality and could be instantly discriminated by a human. To improve the results, based on the intuition that the initial layers of a convolutional net pick up lower-level features of the image, we thought that these layers were already well-trained and froze them, backpropagating only through advanced layers.

4.2 Results

First, we show the results of NVIDIA's pretrained model on Cityscapes as a baseline. The model outputs images with resolution 1024x512.

Figure 2: Frame 1, NVIDIA Pretrained Vid2Vid network

Applying this model with no further training to KITTI yields:

Figure 3: Frame 1, NVIDIA Pretrained Vid2Vid network on KITTI

Figure 4: Frame 1, generated from SSIM loss function, 10 iterations

Figure 5: Frame 1, generated from SSIM loss function, 50 iterations

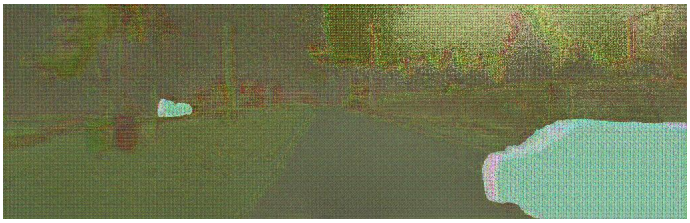


Figure 6: 1st Frame generated from MSE loss function, 10 iterations

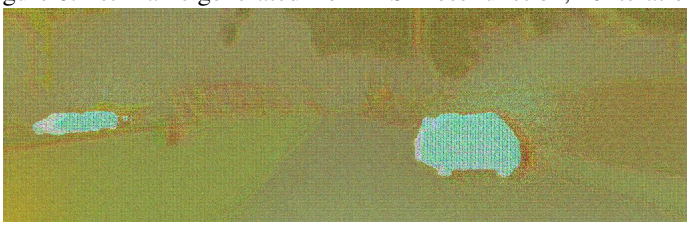
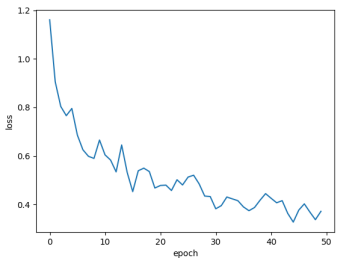


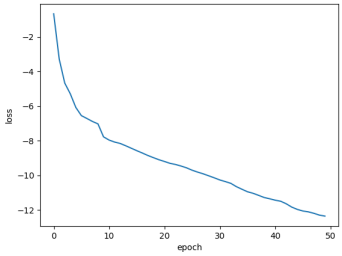
Figure 7: 20th Frame generated from MSE loss function, 10 iterations

We notice that the transfer learning approach we employed results in massive image degradation. However, frame-to-frame stability remains reasonable. Viewing the sequence of generated images as a "video," the model tracks object movement fairly realistically.

For completeness, we have also included selected training loss curves on MSE and SSIM.



(a) MSE Loss Over 50 Epochs



(b) SSIM Loss Over 50 Epochs

The below table lists the models we tested along with the loss functions used, with the training error and test error achieved on each set per image. The pretrained model was trained on a set of 180 frames of video, following NVIDIA's original training scheme, and was tested on a 30 frame training set. Our few-shot model provides a 32-frame sequence as supplementary training data, and tests on a 145 frame sequence. Because of the performance gains we observed in smaller-scale experiments, we froze the first 3 ResNet blocks (of 9) in each generator.

Table 1: Results on Fewshot

Model	**Loss Function**	**Training Err.** *(per img)*	**Test Err.** *(per img)*
Pretrained Baseline	MSE	N/A	0.4425
Pretrained Baseline	SSIM	N/A	-0.1000
Pretrained + 10 epochs	MSE	0.0212	0.0048
Pretrained + 50 epochs	MSE	0.0128	0.0012
Pretrained + 10 epochs	SSIM	-0.2680	0.0030
Pretrained + 50 epochs	SSIM	-0.4258	-0.0001

We omit the training error for the pretrained baseline model, since its training scheme is different from the modified training scheme used for the fewshot approach.

5 Discussion

The accuracy of the generated videos can be qualitatively analyzed by the human eye, to determine whether the GAN has produced a comparable video to the environment depicted in the few shot inputs. This "human preference ratio" is used in many GAN evaluations, including Wang et. al. [1], but we felt that this metric was inappropriate because of the low quality of our images and opted for something more quantitative. The results from above also show high variance, or overfitting, when using SSIM.

However, it is much more difficult to quantitatively score how realistic an image is. Given that the models are not yet realistic, our primary goal was to determine which loss function and training parameters created closest images to reality.

6 Future Work

This transfer learning approach of finetuning a pretrained model is theoretically sound, but due to the high bias of the output results we were able to achieve, we could significantly increase our training time. Other options include modifying the architecture of Vid2Vid itself, but given the complexity of the model (over 1000 layers), much deeper analysis of the model architecture would be required.

Coming up with a loss function that yields 1) a monotonically decreasing loss curve and 2) captures image structure would be ideal. MSE meets neither criterion, being vulnerable to various image transformations, while SSIM meets the second only. In the future, we could try various weighted combinations of different loss functions as well.

7 References

[1] Wang, T.C., Liu, M.C., Zhu, M.C. et al. "Video-to-Video Synthesis." arXiv:1808.06601 [cs.CV]. 3 Dec 2018.

[2] Wang, T.C., Liu, M.C., Tao, A. et al. "Few-shot Video-to-Video Synthesis." arXiv:1910.12713 [cs.CV]. 28 Oct 2019

[3] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[4] A. Geiger, P. Lenz, C. Stiller and R. Urtasun,"Vision meets Robotics: The KITTI Dataset," International Journal of Robotics Research (IJRR), 2013.

[5] Yi Zhu*, Karan Sapra*, Fitsum A. Reda, Kevin J. Shih, Shawn Newsam, Andrew Tao, Bryan Catanzaro, "Improving Semantic Segmentation via Video Propagation and Label Relaxation", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2019, https://nv-adlr.github.io/publication/2018-Segmentation

[6] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, Apr. 2004.