Video-to-Video Synthesis

Ting-Chun Wang[1], **Ming-Yu Liu**[1], **Jun-Yan Zhu**[2], **Guilin Liu**[1],
Andrew Tao[1], **Jan Kautz**[1], **Bryan Catanzaro**[1]
[1]NVIDIA, [2]Massachusetts Institute of Technology
{tingchunw,mingyul,guilinl,atao,jkautz,bcatanzaro}@nvidia.com,
junyanz@mit.edu

Abstract

We study the problem of video-to-video synthesis, whose goal is to learn a mapping function from an input source video (e.g., a sequence of semantic segmentation masks) to an output photorealistic video that precisely depicts the content of the source video. While its image counterpart, the image-to-image synthesis problem, is a popular topic, the video-to-video synthesis problem is less explored in the literature. Without understanding temporal dynamics, directly applying existing image synthesis approaches to an input video often results in temporally incoherent videos of low visual quality. In this paper, we propose a novel video-to-video synthesis approach under the generative adversarial learning framework. Through carefully-designed generator and discriminator architectures, coupled with a spatial-temporal adversarial objective, we achieve high-resolution, photorealistic, temporally coherent video results on a diverse set of input formats including segmentation masks, sketches, and poses. Experiments on multiple benchmarks show the advantage of our method compared to strong baselines. In particular, our model is capable of synthesizing 2K resolution videos of street scenes up to 30 seconds long, which significantly advances the state-of-the-art of video synthesis. Finally, we apply our approach to future video prediction, outperforming several state-of-the-art competing systems. Code is available at our website.

1 Introduction

The capability to model and recreate dynamics of our visual world is essential to building intelligent agents. Apart from purely scientific interest, learning to synthesize continuous visual experiences has a wide range of applications in vision, robotics, and graphics. For example, in model-based reinforcement learning [1, 18], a video synthesis model finds use in approximating visual dynamics of the world for training the agent with less amount of real experience data. Using a learned video synthesis model, one can generate realistic videos without explicitly specifying scene geometry, materials, light transport, and their transformations, which would be cumbersome but necessary when using standard graphic rendering techniques [56].

The video synthesis problem exists in various forms, including future video prediction [12, 38, 35, 60, 54, 11, 52, 32] and unconditional video synthesis [53, 46, 51]. In this paper, we study a new form, which is referred to as the video-to-video synthesis problem. At the core, the problem is concerned with learning a mapping function that can convert an input video to an output video. To the best of our knowledge, a general-purpose solution to video-to-video synthesis has not yet been studied in the prior work, although its image counterpart, the image-to-image synthesis problem, is a popular research topic [26, 50, 2, 49, 62, 33, 34, 24, 63, 56]. Our method is inspired by previous application-specific video synthesis methods [47, 58, 45].

We cast the video-to-video synthesis problem as a distribution matching problem, where the goal is to train a model such that the conditional distribution of the synthesized videos given input

Preprint. Work in progress.

videos, resembles that of real videos. To this end, we leverage the generative adversarial learning framework [14]. Given paired input and output videos, we learn to map input videos to the output domain. With a carefully-designed generator network and a learning objective function, the proposed approach can learn to synthesize high-resolution, photorealistic, and temporally coherent videos. Moreover, we later extend our method to multimodal video synthesis. Conditioning on the same input, our model can produce videos with diverse appearances.

We conduct extensive experimental validation on several datasets on the task of converting a sequence of segmentation masks to photorealistic videos. Both quantitative and qualitative results indicate that the footage synthesized by our approach looks more photorealistic than strong baselines. We further demonstrate that the proposed approach is capable of generating photorealistic 2K resolution videos, up to 30 seconds long. Our method also grants users flexible high-level control on the video generation results. For example, a user can easily replace all the buildings with trees in a street view video. Code and additional results are available at our website.

2 Related Work

Generative Adversarial Networks (GANs). We build our model on GANs [14]. During GAN training, a generator and a discriminator are set up to play a zero-sum game. The goal of the generator is to generate realistic synthetic data so that the discriminator cannot differentiate between real and synthetic data. Although GANs originally used random samples drawn from some noise distribution as input [14, 42, 10], various forms of data can be used as input to the generator, including images [26, 62, 33], categorical labels [41, 40], and textual descriptions [43, 61]. Such conditional models are called conditional GANs, and allow flexible control over the output of the model. Our method belongs to the category of conditional video generation with GANs. However, instead of predicting future videos conditioning on the current observed images [38, 31, 53], our method synthesizes photorealistic videos conditioning on manipulable semantic representations, such as segmentation masks, sketches, or poses.

Image-to-image translation algorithms transfer an input image from one domain to a corresponding image in another domain. There exists a large body of work for this problem [26, 50, 2, 49, 62, 33, 34, 24, 63, 56]. Our approach is their video counterpart. In addition to ensuring that each video frame looks photorealistic, a video synthesis model also has to ensure that frames are temporally coherent, which is a challenging task, especially for a long duration video.

Unconditional video synthesis. Several works [53, 46, 51] extends the GAN framework for unconditional video synthesis, which aims at learning a generator for converting a random vector to a video. VGAN [53] uses a spatial-temporal convolutional network. TGAN [46] projects a video latent code to a set of image latent codes and uses an image generator. MoCoGAN [51] disentangles the latent space to motion and content subspaces and uses a recurrent neural network to generate a sequence of motion codes. Due to the unconditional setting, these methods often produce low-resolution and short-length videos.

Future video prediction. Conditioning on the current observed images, future video prediction models are trained to predict future video frames [12, 38, 35, 60, 54, 55, 11, 52, 32, 31]. These models are often trained by minimizing image reconstruction loss. As a result, they tend to generate blurry videos due to the classic regress-to-the-mean problem. Also, they fail to generate long duration videos even when incorporating generative adversarial training [38, 32]. The video-to-video synthesis problem is substantially different because it does not attempt to predict object motion or any other aspect of the future video frames. Instead, by conditioning on an existing video, our approach is capable of generating high-resolution and long-length videos.

Video-to-video synthesis. While video super-resolution [47, 48] and video inpainting [57] can be considered as special cases of the video-to-video synthesis problem, existing approaches rely on problem-specific constraints in their algorithm designs. Hence, these methods cannot be easily applied to other video-to-video applications. Video style transfer, transferring the style of a reference image (e.g., painting) to a video (e.g., a natural scene video), is another related problem. Although existing methods [6, 16, 22, 45] cannot be directly applied to the video-to-video synthesis problems, we construct a strong baseline by combining a state-of-the-art video style transfer algorithm with a state-of-the-art image-to-image synthesis approach for validating the proposed approach.

3 Video-to-Video Synthesis

Let $\mathbf{s}_1^T \equiv \mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_T$ be a sequence of source images for video synthesis. For example, it can be a sequence of semantic segmentation masks or boundary maps. Let $\mathbf{x}_1^T \equiv \mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T$ be the sequence of corresponding real images. The goal of video-to-video synthesis is to learn a mapping function that can convert $\mathbf{s}_1^T$ to a sequence of output images, $\tilde{\mathbf{x}}_1^T \equiv \tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, ..., \tilde{\mathbf{x}}_T$, so that the conditional distribution of $\tilde{\mathbf{x}}_1^T$ given $\mathbf{s}_1^T$ is identical to the conditional distribution of $\mathbf{x}_1^T$ given $\mathbf{s}_1^T$.

$$p(\tilde{\mathbf{x}}_1^T|\mathbf{s}_1^T) = p(\mathbf{x}_1^T|\mathbf{s}_1^T). \tag{1}$$

Through matching the conditional video distributions, the model learns to generate photorealistic and temporally coherent output sequences as if they were captured by a video camera.

We propose a conditional GAN framework for the conditional video distribution matching task. Let G be a generator that maps an input source sequence to a corresponding output image sequence: $\mathbf{x}_1^T = G(\mathbf{s}_1^T)$. We train the generator by solving the minimax optimization problem given by

$$\max_D \min_G E_{(\mathbf{x}_1^T, \mathbf{s}_1^T)}[\log D(\mathbf{x}_1^T, \mathbf{s}_1^T)] + E_{\mathbf{s}_1^T}[\log(1 - D(G(\mathbf{s}_1^T), \mathbf{s}_1^T))] \tag{2}$$

where D is the discriminator. We note that as solving (2), we minimize the Jensen-Shannon divergence between $p(\tilde{\mathbf{x}}_1^T|\mathbf{s}_1^T)$ and $p(\mathbf{x}_1^T|\mathbf{s}_1^T)$ as shown by Goodfellow et. al. [14].

Solving the minimax optimization problem in (2) is a well-known, challenging task. Careful design of the network architecture and the objective function are required to achieve good performance as shown in the literature [27, 61, 56, 15, 37, 13, 39, 42, 10]. We follow the same spirit and propose a novel network design and novel objective function for video-to-video synthesis as detailed below.

Sequential Generator. To simplify the video-to-video synthesis problem, we make a Markov assumption where we factorize the conditional distribution $p(\tilde{\mathbf{x}}_1^T|\mathbf{s}_1^T)$ to a product form given by

$$p(\tilde{\mathbf{x}}_1^T|\mathbf{s}_1^T) = \prod_{t=1}^{T} p(\tilde{\mathbf{x}}_t|\tilde{\mathbf{x}}_{t-L}^{t-1}, \mathbf{s}_{t-L}^t). \tag{3}$$

In other words, we assume the video frames can be generated sequentially, and the generation of the t-th frame $\tilde{\mathbf{x}}_t$ only depends on 3 things: 1) the current source image $\mathbf{s}_t$, 2) the past L source images $\mathbf{s}_{t-L}^{t-1}$, and 3) the past L generated images $\tilde{\mathbf{x}}_{t-L}^{t-1}$. We use a feed forward network F to model sampling from the conditional distribution $p(\tilde{\mathbf{x}}_t|\tilde{\mathbf{x}}_{t-L}^{t-1}, \mathbf{s}_{t-L}^t)$ and let $\tilde{\mathbf{x}}_t = F(\tilde{\mathbf{x}}_{t-L}^{t-1}, \mathbf{s}_{t-L}^t)$. The final output $\tilde{\mathbf{x}}_1^T$ is then obtained by applying the function F in a recursive manner. In our experiments, we set $L = 2$.

Video signals have a characteristic that consecutive frames contain a large amount of repeating information. If the optical flow from the current frame to the next frame is known, we can use it to warp the current frame to generate an estimation of the next frame. This estimation would be largely correct except for the occluding areas. Based on this observation, we model F as

$$F(\tilde{\mathbf{x}}_{t-L}^{t-1}, \mathbf{s}_{t-L}^t) = (\mathbf{1} - \tilde{\mathbf{m}}_t) \odot \tilde{\mathbf{w}}_{t-1}(\tilde{\mathbf{x}}_{t-1}) + \tilde{\mathbf{m}}_t \odot \tilde{\mathbf{h}}_t \tag{4}$$

where $\odot$ is the element-wise product operator and $\mathbf{1}$ is an image of all ones. The definitions of the other terms in (4) are given below.

- $\tilde{\mathbf{w}}_{t-1} = W(\tilde{\mathbf{x}}_{t-L}^{t-1}, \mathbf{s}_{t-L}^t)$ is the estimated optical flow from $\tilde{\mathbf{x}}_{t-1}$ to $\tilde{\mathbf{x}}_t$, and W is the optical flow prediction function. We estimate the optical flow using both input source images $\mathbf{s}_{t-L}^t$ and previously synthesized images $\tilde{\mathbf{x}}_{t-L}^{t-1}$. By $\tilde{\mathbf{w}}_{t-1}(\tilde{\mathbf{x}}_{t-1})$, we warp $\tilde{\mathbf{x}}_{t-1}$ based on $\tilde{\mathbf{w}}_{t-1}$.
- $\tilde{\mathbf{h}}_t = H(\tilde{\mathbf{x}}_{t-L}^{t-1}, \mathbf{s}_{t-L}^t)$ is the hallucinated image, an image generated from scratch.
- $\tilde{\mathbf{m}}_t = M(\tilde{\mathbf{x}}_{t-L}^{t-1}, \mathbf{s}_{t-L}^t)$ is the occlusion mask and the entry value is between 0 and 1. The function M is the occlusion mask prediction function. Note that our occlusion mask is soft instead of binary, to better handle the "zoom in" problem. For example, when an object is moving closer to the camera, if we simply keep warping the previous frames, the object will just become blurrier over time. By using a soft mask, we can gradually blend in the new pixels as the video plays.

We implement the functions M, W, and H using a residual network architecture [20]. To generate high-resolution videos, we apply a coarse-to-fine generator design proposed by Wang et. al. [56].

Using multiple discriminators has been shown beneficial in mitigating the model collapsing problem in GAN training [13, 51, 56]. We use two types of discriminators in our approach.

Conditional image discriminator D_I. The purpose of D_I is to ensure that each output frame resembles a real image with the same source image. It is a conditional discriminator. It should output 1 for a true pair $(\mathbf{x}_t, \mathbf{s}_t)$ and 0 for a fake one $(\tilde{\mathbf{x}}_t, \mathbf{s}_t)$.

Conditional video discriminator D_V. The purpose of D_V is to ensure that consecutive output frames resemble those in a real video with the same optical flow. It is also a conditional discriminator. While D_I conditions on the source image, D_V conditions on the flow. Let $\mathbf{w}_{t-K}^{t-2}$ be $K-1$ optical flow for the K consecutive real images $\mathbf{x}_{t-K}^{t-1}$. The discriminator D_V should output 1 for a true pair $(\mathbf{x}_{t-K}^{t-1}, \mathbf{w}_{t-K}^{t-2})$ and 0 for a fake one $(\tilde{\mathbf{x}}_{t-K}^{t-1}, \mathbf{w}_{t-K}^{t-2})$.

We introduce two sampling operators to facilitate the discussion. First, let ϕ_I be a random image sampling operator such that $\phi_I(\mathbf{x}_1^T, \mathbf{s}_1^T) = (\mathbf{x}_i, \mathbf{s}_i)$ where i is a random integer uniformly distributed from 1 to T. In other words, ϕ_I represents the operation of randomly sampling a pair of images from $(\mathbf{x}_1^T, \mathbf{s}_1^T)$. On the other hand, we define ϕ_V as a sampling operator that randomly retrieve K consecutive frames. Specifically, $\phi_V(\mathbf{w}_1^{T-1}, \mathbf{x}_1^T, \mathbf{s}_1^T) = (\mathbf{w}_{i-K}^{i-2}, \mathbf{x}_{i-K}^{i-1}, \mathbf{s}_{i-K}^{i-1})$ where i is a random integer uniformly distributed from $K+1$ to $T+1$. This operator allows a simple notation for retrieving K consecutive frames and the corresponding $K-1$ optical flow images from both the ground truth and generated videos. With ϕ_I and ϕ_V, we are ready to present our learning objective function.

Learning Objective Function. We train the sequential video synthesis function F by solving

$$\min_G \big(\max_{D_I} \mathcal{L}_I(F, D_I) + \max_{D_V} \mathcal{L}_V(F, D_V) \big) + \mathcal{L}_W(F), \tag{5}$$

where $\mathcal{L}_I$ is the GAN loss on images defined by the conditional image discriminator D_I, $\mathcal{L}_V$ is the GAN loss on K-consecutive frames defined by D_V, and $\mathcal{L}_W(F)$ is the flow estimation loss. In addition to the loss terms in (5), we also use the discriminator feature matching loss [30, 56] and VGG feature matching loss [8, 56] as they improve the convergence speed and training stability [56].

The GAN loss $\mathcal{L}_I$ is derived from the image-conditional GAN loss [26] by utilizing the operator ϕ_I

$$E_{\phi_I(\mathbf{x}_1^T, \mathbf{s}_1^T)}[\log D_I(\mathbf{x}_i, \mathbf{s}_i)] + E_{\phi_I(\tilde{\mathbf{x}}_1^T, \mathbf{s}_1^T)}[\log(1 - D_I(\tilde{\mathbf{x}}_i, \mathbf{s}_i))] \tag{6}$$

Similarly, the GAN loss $\mathcal{L}_V$ is given by

$$E_{\phi_V(\mathbf{w}_1^{T-1}, \mathbf{x}_1^T, \mathbf{s}_1^T)}[\log D_V(\mathbf{x}_{i-K}^{i-1}, \mathbf{w}_{i-K}^{i-2})] + E_{\phi_V(\mathbf{w}_1^{T-1}, \tilde{\mathbf{x}}_1^T, \mathbf{s}_1^T)}[\log(1 - D_V(\tilde{\mathbf{x}}_{i-K}^{i-1}, \mathbf{w}_{i-K}^{i-2}))] \tag{7}$$

Recall that the synthesized video $\tilde{\mathbf{x}}_1^T$ is obtained by recursively applying F to the generated images. The gradients from the discriminators are back-propagated to F through $\tilde{\mathbf{x}}_1^T$.

The flow loss $\mathcal{L}_W$ includes two terms. The first is the endpoint error between the ground truth and the estimated flow, and the second is the warping loss when the flow is used to warp the previous frame to the next frame. Let $\mathbf{w}_t$ be the ground truth flow from $\mathbf{x}_t$ to $\mathbf{x}_{t+1}$. The flow loss $\mathcal{L}_W$ is given by

$$\mathcal{L}_W = \frac{1}{T-1} \sum_{t=1}^{T-1} \lambda_W \big(||\tilde{\mathbf{w}}_t - \mathbf{w}_t||_1 + \|\tilde{\mathbf{w}}_t(\mathbf{x}_t) - \mathbf{x}_{t+1}\|_1 \big) \tag{8}$$

where λ_W is the loss weight, which is set to 10 throughout the experiments.

Foreground-Background Prior. When using semantic segmentation masks as the source video, we can divide the image into foreground and background areas based on the semantic class. For example, buildings and roads belong to the background, while cars and pedestrians belong to the foreground. This foreground-background separation provides a strong prior, and we leverage it in the generator design to further improve the video-to-video synthesis performance.

We propose decomposing the image hallucination function into a foreground model $\tilde{\mathbf{h}}_{F,t} = H_F(\mathbf{s}_{t-L}^t)$ and a background model $\tilde{\mathbf{h}}_{B,t} = H_B(\tilde{\mathbf{x}}_{t-L}^{t-1}, \mathbf{s}_{t-L}^t)$. We note that background motion is a global transformation in general, where optical flow can be estimated quite accurately. As a result, background image synthesis can be generated accurately via warping. The task of the background hallucination function H_B is simply to synthesize background content in the occluded areas. On the other hand, the foreground objects often have large motion and only occupy a small portion of the image, which

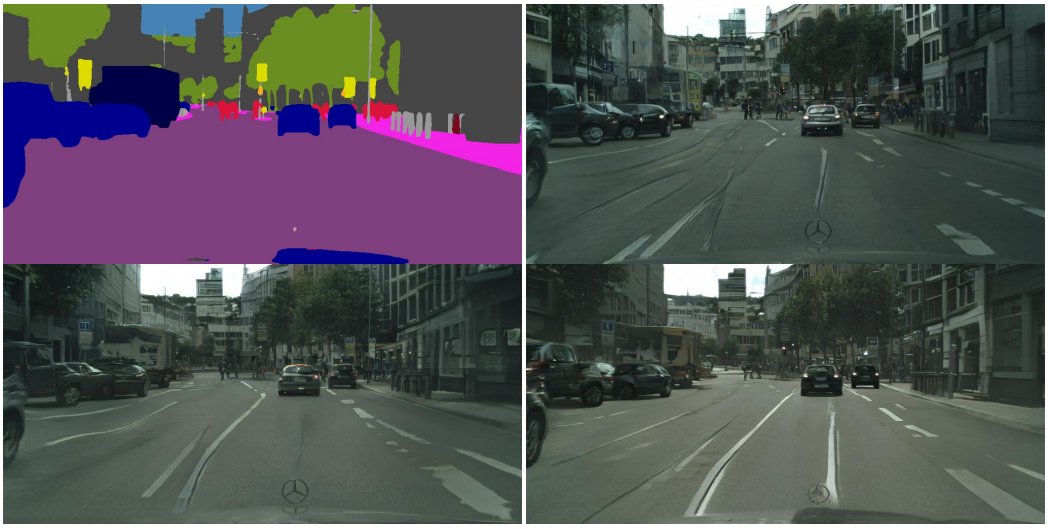

Figure 1: Cityscapes results. Top left: input. Top right: `pix2pixHD`. Bottom left: `COVST`. Bottom right: proposed. *The figure is best viewed with Acrobat Reader. Click the image to play the video clip.*

makes optical flow estimation difficult. The function H_F has to synthesize most of the foreground content from scratch. With this foreground–background prior, F is then given by

$$F(\tilde{\mathbf{x}}_{t-L}^{t-1}, \mathbf{s}_{t-L}^{t}) = (\mathbf{1} - \tilde{\mathbf{m}}_t) \odot \tilde{\mathbf{w}}_{t-1}(\tilde{\mathbf{x}}_{t-1}) + \tilde{\mathbf{m}}_t \odot \big((\mathbf{1} - \tilde{\mathbf{m}}_{B,t}) \odot \tilde{\mathbf{h}}_{F,t} + \tilde{\mathbf{m}}_{B,t} \odot \tilde{\mathbf{h}}_{B,t}\big) \quad (9)$$

where $\tilde{\mathbf{m}}_{B,t}$ is the background mask and is derived from $\mathbf{s}_t$. In practice, we find that by setting the foreground areas as occluded in the occlusion mask, the model converges much faster.

Multimodal Synthesis. The synthesis function F is a unimodal mapping function. Given an input source video, it can only generate one output video. In order to achieve multimodal synthesis, we adopt a feature embedding scheme proposed by Wang et. al. [56], which is applicable when the source video consists of instance-level semantic segmentation masks. Specifically, at training time, we train an image encoder E to encode the ground truth real image $\mathbf{x}_t$ into a d-dimensional feature map ($d = 3$ in our experiments). An instance feature averaging operation is then applied to the map so that all the pixels belonging to the same object have the same d-dimensional feature vectors. We will denote the instance-averaged feature map $\mathbf{z}_t$, which is concatenated with the input semantic segmentation mask $\mathbf{s}_t$ as the input to F. Once training is done, we apply E and the instance feature averaging operation to all the images in the training videos. We then fit a mixture of Gaussian distribution to the feature vectors belonging to the same object class. If there are N semantic classes, there are N mixture of Gaussian distributions. At test time, we sample a feature vector for each object instance in the video using the mixture of Gaussian distribution of the object class. With different feature vectors, the sequential generator F generates videos with different appearances.

4 Experiments

Implementation. We train our network in a spatio-temporally progressive manner. This means we start with generating low-resolution and few frames, and all the way up to generating full resolution and 30 (or more) frames. Our coarse-to-fine generator consists of 3 scales, which operates on 512×256, 1024×512, and 2048×1024 resolutions, respectively. We share all the weights in M and W except for the output layer. We use the multi-scale patch GAN discriminator architecture [56] for D_I. Our D_V is also multi-scale, working on different frame rates of the video to ensure both short-term and long-term consistency. More details are given in the appendix.

We train our model for 40 epochs using the ADAM [29] optimizer with $lr = 0.0002$ and $(\beta_1, \beta_2) = (0.5, 0.999)$ with the LSGAN [37] loss. The training is performed on an NVIDIA DGX1 machine. Due to the large image resolution, even with one short video per batch, we still have to use all the GPUs in DGX1 (8 V100 GPUs, each with 16GB memory) for training. We do model parallelism where we distribute the generator computation task to 4 GPUs and the discriminator computation task to the other 4 GPUs. Training generally takes about 10 days for 2K resolution.

Table 1: Comparison between competing video-to-video synthesis approaches on Cityscapes.

Inception Net. of FID	I3D	ResNeXt
`pix2pixHD`	5.57	0.18
`COVST`	5.55	0.18
proposed	**4.66**	**0.15**

Human Preference Score	short seq.	long seq.
proposed / `pix2pixHD`	**0.87** / 0.12	**0.83** / 0.17
proposed / `COVST`	**0.84** / 0.16	**0.80** / 0.20

Table 2: Ablation study. We compare the proposed approach to its 3 variants.

Human Preference Score	
proposed / `no background-foreground prior`	**0.80** / 0.20
proposed / `no conditional video discriminator`	**0.84** / 0.16
proposed / `no flow warping`	**0.67** / 0.33

Table 3: Comparison between competing future video synthesis approach on Cityscapes.

Inception Net. of FID	I3D	ResNeXt
PredNet	11.18	0.59
MCNet	10.00	0.43
proposed	**3.44**	**0.18**

Human Preference Score	
proposed / PredNet	**0.92** / 0.08
proposed / MCNet	**0.98** / 0.02

Datasets. We evaluate the proposed approach on several datasets.

- **Cityscapes** [9]. The dataset consists of street scene videos captured in several cities in Germany where the image resolution is 2048×1024. Only a subset of images in the videos contain ground truth semantic segmentation masks. We use those images to train a DeepLabV3 semantic segmentation network [7] and apply the trained network to segment all the videos to obtain the input source videos. We use the optical flow extracted by FlowNet2 [25] as the ground truth flow $\mathbf{w}$ to train F. We treat the instance segmentation masks computed by the Mask R-CNN [19] as our instance-level ground truth. In summary, the training set contains 2975 videos, each with 30 frames. The validation set consists of 500 videos, each with 30 frames. Finally, we test our method on 3 long sequences from a test set, with 600, 1100, and 1200 frames, respectively. We will show that although we train our model using short videos, it is capable of synthesizing long videos.
- **Apolloscape** [23] consists of 73 street scene videos captured in Beijing, where the video length varies from 100 to 1000 frames. Similar to Cityscapes, Apolloscape is constructed for the image/video semantic segmentation task. But we use it for synthesizing videos using the semantic segmentation mask. We split the dataset into half for training and validation.
- **Face video dataset.** We use the real videos in the FaceForensics dataset [44], which contains 854 videos of news briefing from different reporters. We use this dataset for the sketch video to face video synthesis task. To extract sequences of sketches from a video, we first apply a landmark detection algorithm [28] to find the facial landmarks in each frame. The facial landmarks are then connected together to create the face sketch. To model the background, we extract Canny edges outside the face regions. We split the dataset into 704 videos for training and 150 videos for validation.
- **Dance video dataset.** We download a set of dance videos from YouTube for the pose video to human motion video synthesis task. Each video is about $3 \sim 4$ minutes at a resolution of 1280×720, and we crop the central 512×720 regions. To extract the human body pose, we use the DensePose [17] and the OpenPose [3] algorithms, and directly concatenate the results together. The training set consists of a dance video from a single dancer[1], while the test set consists of videos of other dance motions or from other dancers.

Baselines. We compare our approach to several baseline methods using the same training data.

- `pix2pixHD` [56] is the state-of-the-art image-to-image synthesis approach. As applying the approach to the video-to-video synthesis task, we simply process images in the source videos independently and assemble the output images to construct the output video.

[1]We thank Lisa Rhee for allowing us to use her videos for training.

- `COVST` is our strong baseline, derived from the coherent video style transfer work [6] by replacing the stylization network with `pix2pixHD`. The key idea in `COVST` is to warp high-level deep features using optical flow for achieving temporally coherent outputs. No additional adversarial training is applied. We feed in ground truth optical flow to `COVST`, which is impractical for real applications. In contrast, our model estimates optical flow from source videos.

Evaluation metrics. We use both subjective and objective metrics for performance evaluation.

- **Human Preference Score.** We perform human subjective test for evaluating the visual quality of synthesized videos. We use the Amazon Mechanical Turk (AMT) platform. Each question is an AB test, where an AMT worker is first shown two videos at a time (results synthesized by two different algorithms) and then asked which one looks more like a video captured by a real camera. We specifically ask the worker to check for both temporal coherence and image quality. A worker must have a life-time task approval rate greater than 98% to participate in the evaluation. For each question, we gather answers from 10 different workers. The human preference score of an algorithm is then given by the ratio that the algorithm outputs are preferred.
- **Frechet Inception Distance (FID)** [21] is a popular metric for generative model evaluation, because it correlates well with visual quality of generated samples. The FID was originally developed for image evaluation. We propose a variant for video evaluation, which measures both visual quality and temporal consistency. Specifically, we use a pretrained CNN-based video recognition network as a feature extractor by removing the last few layers from the network. This feature extractor will be our inception network. We apply the inception network as a spatial-temporal CNN to each of the synthesized videos. This renders a spatial-temporal map of feature vectors. We gather the feature vectors from all the synthesized videos and compute their mean $\tilde{\mu}$ and covariance matrix $\tilde{\Sigma}$. We also perform the same processing to the ground truth videos. The FID is then given by $\|\mu - \tilde{\mu}\|^2 + \text{Tr}\Big(\Sigma + \tilde{\Sigma} - 2\sqrt{\Sigma\tilde{\Sigma}}\Big)$. We use two different pretrained CNN-based video recognition networks in our evaluation, I3D [4] and ResNeXt [59].

Main Results. We compare the proposed approach to the baseline methods on the Cityscapes benchmark where we apply the learned models to synthesize 500 short video clips in the validation set. Quantitative and qualitative evaluation results are reported in Table 1, which show that the videos synthesized by the proposed approach have a smaller FID and are much more preferred by the human subjects. We also report the human preference scores on 3 long test videos. Again, the videos rendered by our approach are considered more realistic by the human subjects. The performance scores for the Apolloscape dataset are given in the appendix.

In Figures 1 and 2, we visualize the video synthesis results. Although each image rendered by `pix2pixHD` is photorealistic, the resulting video lacks temporal coherence. The road lane markings and building appearances are inconsistent across frames. While improving upon `pix2pixHD`, `COVST` still suffers from temporal inconsistency. On the contrary, our approach renders high-resolution, photorealistic, and temporally consistent video output. We are able to generate 30-second long videos, showing that our approach synthesizes convincing videos of longer length.

We conduct an ablation study to analyze several of our design choices. Specifically, we create 3 variants. In one variant, we do not use the foreground-background prior, which is termed `no background-foreground prior`. That is, instead of using (9), we use (4). The second variant is `no conditional video discriminator` where we do not use D_V for training. In the last variant, we remove W and M from F in (4) and only use H for synthesis. This variant is referred to as `no flow warping`. We use human preference score on Cityscapes for this ablation study. As shown in Table 2, without the ablated component, the visual quality of video outputs drops significantly.

Multimodal Results. In Figure 3, we visualize multimodal synthesis results. In this example, we keep the sampled feature vectors of all the object instances in the video the same except for the road instance. As shown in the figure, we synthesize temporally smooth videos with different road surface appearances.

Semantic Manipulation. Our approach also allows the user to semantically manipulate the source videos for diverse outputs. For example, a user could add a car to the scene and synthesize a new video with one more car. In Figure 4, we show an example of changing the semantic labels. In the left video, we replace all trees with buildings in the original segmentation masks and synthesize a new video. On the right, we show the result of replacing buildings with trees.

Figure 2: Apolloscape results. Left: `pix2pixHD`. Center: `COVST`. Right: proposed. The input semantic segmentation mask video is shown in the left video. *The figure is best viewed with Acrobat Reader. Click the image to play the video clip.*

Figure 3: Example multi-modal video synthesis results. These synthesized videos contain different road surfaces. *The figure is best viewed with Acrobat Reader. Click the image to play the video clip.*

Figure 4: Example results of changing input semantic segmentation masks to generate diverse videos. Left: tree→building. Right: building→tree. The original video is shown in Figure 3. *The figure is best viewed with Acrobat Reader. Click the image to play the video clip.*

Sketch-to-Video Synthesis. We train a sketch-to-face synthesis video model using the real face videos in the FaceForensics dataset [44]. As shown in Figure 5, our model can convert sequences of sketches to photorealistic output videos.

Pose-to-Video Synthesis. In addition, we apply our framework to the task of converting sequences of human poses to photorealistic output videos. We note that the image counterpart is studied in a recent work [36], and a similar idea that works on dance videos is also proposed in [5]. As shown in Figure 6, our model learns to synthesize photorealistic output dance videos that contain unseen body shapes and motions in the training time.

Future Video Synthesis. We show an extension of our approach to the future video synthesis task—given few observed frames of a video, learn to predict the future video. We decompose the task into two sub-tasks: 1) synthesizing future semantic segmentation masks using the observed frames, and 2) converting the synthesized segmentation masks into videos. After extracting the segmentation masks from the few observed frames, we train a sequential generator similar to (7) to predict future semantic segmentation masks via solving a variant of (5). We then use the proposed video-to-video synthesis approach to convert the predicted semantic segmentation masks to a future video.

We conduct both quantitative and qualitative evaluations with comparisons to two start-of-the-art approaches: PredNet [35] and MCNet [52]. As shown in Table 3, our approach renders smaller FIDs and our resulting videos are more preferred by the human subjects. In Figure 7, we visualize the future video synthesis results. While the image quality of the results from the competing algorithms degrades significantly over time, ours remains consistent.

Figure 5: Example sketch-to-face video results. *The figure is best viewed with Acrobat Reader. Click the image to play the video clip.*

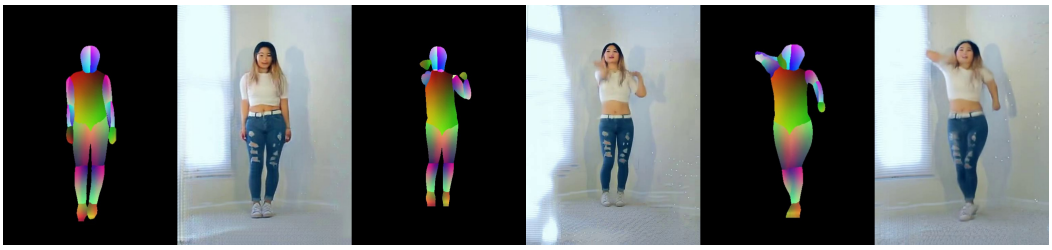

Figure 6: Example pose-to-dance video results. *The figure is best viewed with Acrobat Reader. Click the image to play the video clip.*

Figure 7: Future video synthesis results. Top left: ground truth. Top right: PredNet [35] Bottom left: MCNet [52]. Bottom right: ours. *The figure is best viewed with Acrobat Reader. Click the image to play the video clip.*

Limitations and Future Work. Our method fails in a couple of situations although outperforming previous methods by a large margin. For example, our model struggles in synthesizing turning cars due to insufficient information in label maps. We suspect that this could be potentially addressed by adding additional 3D information, such as depth maps. Furthermore, our model still can not guarantee that an object has a consistent appearance across the whole video. Occasionally, a car may change its color gradually. We believe that this issue can be resolved if we incorporate object tracking information, since we can use it to enforce that the same object shares the same appearance throughout the entire video. Finally, when we perform semantic manipulations such as turning trees into buildings, there are some visible artifacts as building and trees have different label shapes. This might be resolved if we train our model with coarser semantic labels as the trained model would be less affected by the label shapes.

5 Conclusion

We present a general video-to-video synthesis framework based on conditional GANs. Through novel generator and discriminator network designs inspired by visual dynamics priors, we achieve synthesizing high-resolution, photorealistic, and temporally consistent videos. Its extension to the future video synthesis task also compares favorably against the state-of-the-art.

References

[1] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38, 2017.

[2] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[3] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[4] J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[5] C. Chan, S. Ginosar, T. Zhou, and A. A. Efros. Everybody dance now. In *European Conference on Computer Vision (ECCV) Workshop*, 2018.

[6] D. Chen, J. Liao, L. Yuan, N. Yu, and G. Hua. Coherent online video style transfer. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[7] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[8] Q. Chen and V. Koltun. Photographic image synthesis with cascaded refinement networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[9] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[10] E. Denton, S. Chintala, A. Szlam, and R. Fergus. Deep generative image models using a Laplacian pyramid of adversarial networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[11] E. L. Denton and V. Birodkar. Unsupervised learning of disentangled representations from video. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[12] C. Finn, I. Goodfellow, and S. Levine. Unsupervised learning for physical interaction through video prediction. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[13] A. Ghosh, V. Kulharia, V. Namboodiri, P. H. Torr, and P. K. Dokania. Multi-agent diverse generative adversarial networks. *arXiv preprint arXiv:1704.02906*, 2017.

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

[15] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[16] A. Gupta, J. Johnson, A. Alahi, and L. Fei-Fei. Characterizing and improving stability in neural style transfer. *arXiv preprint arXiv:1705.02092*, 2017.

[17] R. A. Güler, N. Neverova, and I. Kokkinos. Densepose: Dense human pose estimation in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[18] D. Ha and J. Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

[19] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[20] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[21] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[22] H. Huang, H. Wang, W. Luo, L. Ma, W. Jiang, X. Zhu, Z. Li, and W. Liu. Real-time neural style transfer for videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[23] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang. The apolloscape dataset for autonomous driving. *arXiv preprint arXiv:1803.06184*, 2018.

[24] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz. Multimodal unsupervised image-to-image translation. *arXiv preprint arXiv:1804.04732*, 2018.

[25] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[26] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[27] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations (ICLR)*, 2018.

[28] D. E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 2009.

[29] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[30] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. In *International Conference on Machine Learning (ICML)*, 2016.

[31] A. X. Lee, R. Zhang, F. Ebert, P. Abbeel, C. Finn, and S. Levine. Stochastic adversarial video prediction. *arXiv preprint arXiv:1804.01523*, 2018.

[32] X. Liang, L. Lee, W. Dai, and E. P. Xing. Dual motion gan for future-flow embedded video prediction. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[33] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[34] M.-Y. Liu and O. Tuzel. Coupled generative adversarial networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[35] W. Lotter, G. Kreiman, and D. Cox. Deep predictive coding networks for video prediction and unsupervised learning. In *International Conference on Learning Representations (ICLR)*, 2017.

[36] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool. Pose guided person image generation. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[37] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[38] M. Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. In *International Conference on Learning Representations (ICLR)*, 2016.

[39] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[40] T. Miyato and M. Koyama. cgans with projection discriminator. In *International Conference on Learning Representations (ICLR)*, 2018.

[41] A. Odena, C. Olah, and J. Shlens. Conditional image synthesis with auxiliary classifier gans. In *International Conference on Machine Learning (ICML)*, 2017.

[42] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2015.

[43] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In *International Conference on Machine Learning (ICML)*, 2016.

[44] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner. Faceforensics: A large-scale video dataset for forgery detection in human faces. *arXiv preprint arXiv:1803.09179*, 2018.

[45] M. Ruder, A. Dosovitskiy, and T. Brox. Artistic style transfer for videos. In *German Conference on Pattern Recognition*, 2016.

[46] M. Saito, E. Matsumoto, and S. Saito. Temporal generative adversarial nets with singular value clipping. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[47] E. Shechtman, Y. Caspi, and M. Irani. Space-time super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 27(4):531–545, 2005.

[48] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[49] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[50] Y. Taigman, A. Polyak, and L. Wolf. Unsupervised cross-domain image generation. In *International Conference on Learning Representations (ICLR)*, 2017.

[51] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz. Mocogan: Decomposing motion and content for video generation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[52] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee. Decomposing motion and content for natural video sequence prediction. In *International Conference on Learning Representations (ICLR)*, 2017.

[53] C. Vondrick, H. Pirsiavash, and A. Torralba. Generating videos with scene dynamics. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[54] J. Walker, C. Doersch, A. Gupta, and M. Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *European Conference on Computer Vision (ECCV)*, 2016.

[55] J. Walker, K. Marino, A. Gupta, and M. Hebert. The pose knows: Video forecasting by generating pose futures. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[56] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[57] Y. Wexler, E. Shechtman, and M. Irani. Space-time video completion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[58] Y. Wexler, E. Shechtman, and M. Irani. Space-time completion of video. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 29(3), 2007.

[59] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[60] T. Xue, J. Wu, K. Bouman, and B. Freeman. Visual dynamics: Probabilistic future frame synthesis via cross convolutional networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[61] H. Zhang, T. Xu, H. Li, S. Zhang, X. Huang, X. Wang, and D. Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[62] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[63] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman. Toward multimodal image-to-image translation. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

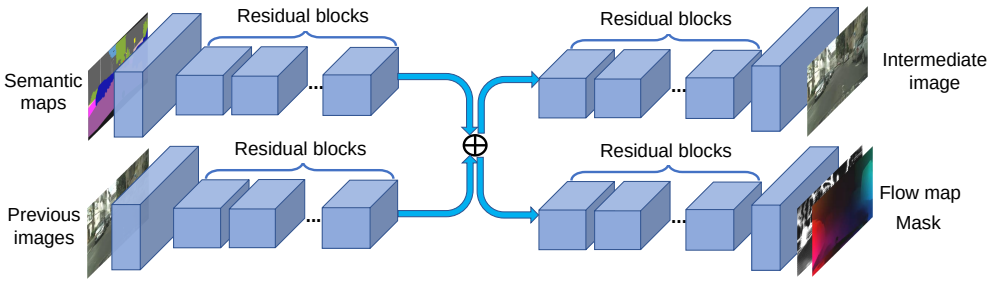

Figure 8: The network architecture (G_1) for low-res videos. Our network takes in a number of semantic label maps and previously generated images, and outputs the intermediate frame and the flow map along with the mask.

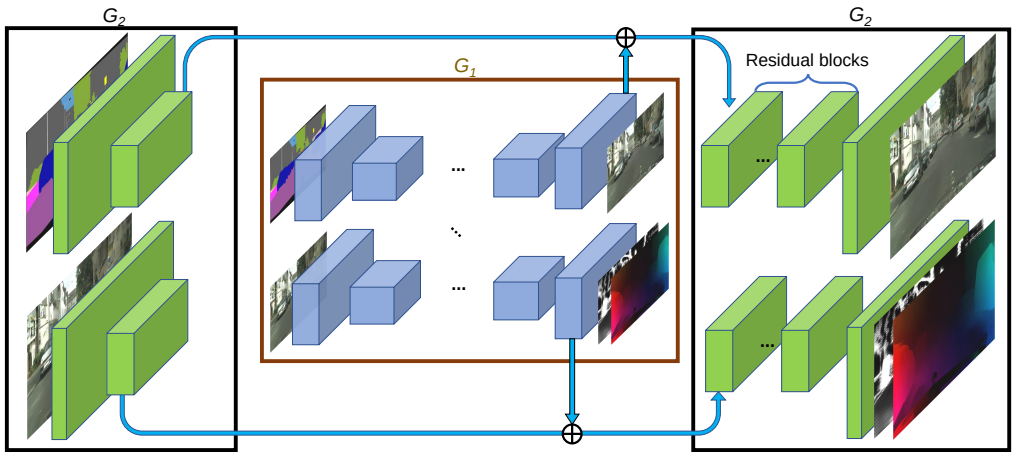

Figure 9: The network architecture (G_2) for higher resolution videos. The label maps and previous frames are downsampled and fed into the low-res network G_1. Then, the features from the high-res network and the last layer of the low-res network are summed and fed into another series of residual blocks to output the final images.

A Network Architecture

A.1 Generators

Our network adopts a coarse-to-fine architecture. For the lowest resolution, the network takes in a number of semantic label maps $\mathbf{s}_{t-L}^{t}$ and previously generated frames $\tilde{\mathbf{x}}_{t-L}^{t-1}$ as input. The label maps are concatenated together and undergo a number of residual blocks to form intermediate high-level features. Same with the previously generated images. Then, these two intermediate layers are added, and fed into two separate residual networks to output the hallucinated image $\tilde{\mathbf{h}}_t$, and the flow map $\tilde{\mathbf{w}}_t$ as well as the mask $\tilde{\mathbf{m}}_t$ (Figure 8).

Next, to build from low-res results to higher-res results, we apply another network G_2 on top of the low-res network G_1 (Figure 9). Specifically, the inputs are first downsampled and fed into G_1. Then, we extract features from the last feature layer of G_1, and add it to the feature layer of G_2. These summed features are then fed into another series of residual blocks to output the higher resolution images.

A.2 Discriminators

For discriminators, we adopt the multi-scale patch GAN architecture. To perform our temporally multi-scale approach, we subsample the actual/generated sequences by different amounts to generate different inputs to the temporal discriminators. In the finest scale, we take K consecutive frames in the original sequence as input. In the next scale, we subsample the video by a factor of K (i.e.,

skipping every $K-1$ intermediate frames), then take consecutive K frames in this new sequence as input. We do this for up to 3 scales in our implementation, and found that this helps us ensure both short-term and long-term consistency.

A.3 Feature matching loss

In our learning objective function, we also add VGG feature matching loss and discriminator feature matching loss to improve the training stability. For VGG feature matching loss, we use VGG as a feature extractor and minimize L1 losses between the extracted features from the real and the generated images. Specifically, we add $\sum_i \frac{1}{P_i}[||\psi^{(i)}(\mathbf{x}) - \psi^{(i)}(G(\mathbf{s}))||_1]$ to our objective, where $\psi^{(i)}$ denotes the i-th layer with P_i elements of the VGG network. Similarly, we adopt the discriminator feature matching loss, to match the statistics of features extracted by the GAN discriminators.

B Evaluation for the Apolloscape Dataset

We provide both the FID and the human preference score on the Apolloscape dataset. For both metrics, our method outperforms the other baselines.

Table 4: Comparison between competing video-to-video synthesis approaches on Apolloscape.

Inception Net. of FID	I3D	ResNeXt
`pix2pixHD`	2.33	0.128
`COVST`	2.36	0.128
proposed	**2.24**	**0.125**

Human Preference Score	
proposed / `pix2pixHD`	**0.61** / 0.39
proposed / `COVST`	**0.59** / 0.41