UNSUPERVISED VIDEO-TO-VIDEO TRANSLATION VIA SELF-SUPERVISED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing unsupervised video-to-video translation methods fail to produce translated videos which are frame-wise realistic, semantic information preserving and video-level consistent. In this work, we propose a novel unsupervised video-to-video translation model. Our model decomposes the style and the content, uses specialized encoder-decoder structure and propagates the inter-frame information through bidirectional recurrent neural network (RNN) units. The style-content decomposition mechanism enables us to achieve long-term style-consistent video translation results as well as provides us with a good interface for modality flexible translation. In addition, by changing the input frames and style codes incorporated in our translation, we propose a video interpolation loss, which captures temporal information within the sequence to train our building blocks in a self-supervised manner. Our model can produce photo-realistic, spatio-temporal consistent translated videos in a multimodal way. Subjective and objective experimental results validate the superiority of our model over existing methods.

1 INTRODUCTION

Recent image-to-image translation (I2I) works have achieved astonishing results by employing Generative Adversarial Networks (GANs) (Goodfellow et al., 2014). Most of the GAN-based I2I methods mainly focus on the case where paired data exists (Isola et al. (2017b), Zhu et al. (2017b), Wang et al. (2018b)). However, with the cycle-consistency loss introduced in CycleGAN (Zhu et al., 2017a), promising performance has been achieved also for the unsupervised image-to-image translation (Huang et al. (2018), Almahairi et al. (2018), Liu et al. (2017), Mo et al. (2018), Romero et al. (2018), Gong et al. (2019)). While there is an explosion of papers on I2I, its video counterpart is much less explored. Compared with the I2I task, video-to-video translation (V2V) is more challenging. Besides the frame-wise realistic and semantic preserving requirements, which is also required in the I2I task, V2V methods additionally need to consider the temporal consistency issue for generating sequence-wise realistic videos. Consequently, directly applying I2I methods on each frame of the video will not work out as those methods fail to utilize temporal information and can not assure any temporal consistency within the sequence.

In their seminal work, Wang et al. (2018a) combined the optical flow and video-specific constraints and proposed a general solution for V2V in a supervised way. Their sequential generator can generate long-term high-resolution video sequences. However, their vid2vid model (Wang et al. (2018a)) relies heavily on labeled data. Based on the I2I CycleGAN approach, previous methods on unsupervised V2V proposed to design spatio-temporal translator or loss to achieve more temporally consistent results while preserving semantic information. In order to generate temporally consistent video sequences, Bashkirova et al. (2018) proposed a 3DCycleGAN method which adopts 3D convolution in the generator and discriminator of the CycleGAN framework to capture temporal information. However, since the small 3D convolution operator (with temporal dimension 3) only captures dependency between adjacent frames, 3DCycleGAN can not exploit temporal information for generating long-term consistent video sequences. Furthermore, the vanilla 3D discriminator is also limited in capturing complex temporal relationships between video frames. As a result, when the gap between input and target domain is large, 3DCycleGAN tends to sacrifice the image-level reality and generates blurry and gray outcomes. Recently, Bansal et al. (2018) designed a Recycle loss for jointly modeling the spatio-temporal relationship between video frames. They trained a temporal predictor to predict the next frame based on two past frames, and plugged the temporal predictor in the cycle-loss to impose the spatio-temporal constraint on the traditional image translator. As the temporal predictors can be trained from video sequences in source and target domain in a self-supervised manner, the recycle-loss is more stable than the 3D discriminator loss proposed by Bashkirova et al. (2018). The RecycleGAN method of Bansal et al. (2018) achieved state-of-the-art unsupervised V2V results. Despite its success in translation scenarios with less variety, such as face-to-face or flower-to-flower, we have experimentally found that applying RecycleGAN to translate videos between domains with a large gap is still challenging. We think the following two points are major reasons which affect the application of RecycleGAN method in complex scenarios. On one hand, the translator in Bansal et al. (2018) processes input frames independently, which has limited capacity in exploiting temporal information; and on the other hand, its temporal predictor only imposes the temporal constraint between a few adjacent frames, the generated video content still might shift abnormally: a sunny scene could change to a snowy scene in the following frames. In a concurrent work, Chen et al. (2019) incorporate optical flow to add motion cycle consistency and motion translation constraints. However, their Motion-guided CycleGAN still suffers from the same two limitations as the RecycleGAN method.

In this paper, we propose UVIT, a novel method for unsupervised video-to-video translation. We assume that a temporally consistent video sequence should simultaneously be: 1) long-term style consistent and 2) short-term content consistent. Style consistency requires the whole video sequence to have the same style, it ensures the video frames to be overall realistic; while the content consistency refers to the appearance continuity of contents in adjacent video frames and ensures the video frames to be dynamically vivid. Compared with previous methods which mainly focused on imposing short-term consistency between frames, we have considered in addition the long-term consistency issue which is crucial to generate visually realistic video sequences.

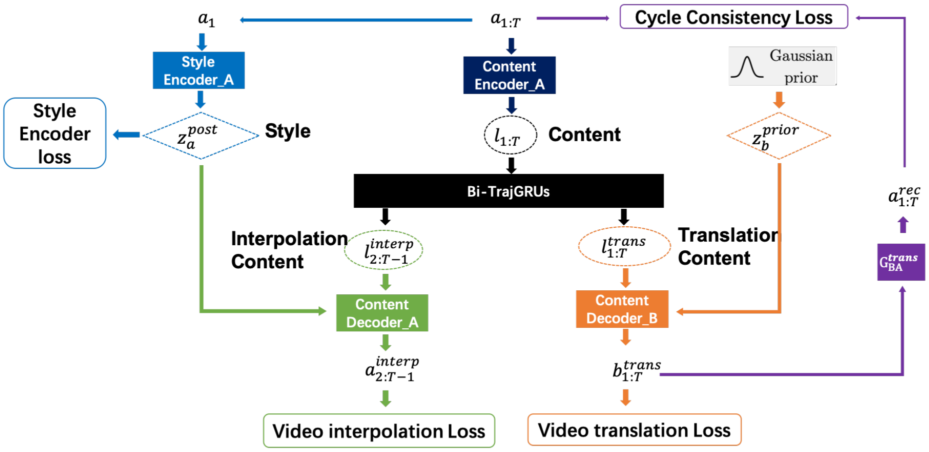

Figure 1: Overview of our proposed UVIT model: given an input video sequence, we first decompose it to the content by Content Encoder and the style by Style Encoder. Then the content is processed by special RNN units– TrajGRUs (Shi et al., 2017) to get the content used for translation and interpolation recurrently. Finally, the translation content and the interpolation content are decoded to the translated video and the interpolated video together with the style latent variable. We depict here the video translation loss (orange), the cycle consistency loss (violet), the video interpolation loss (green) and the style encoder loss (blue).

To simultaneously impose style and content consistency, we adopt an encoder-decoder architecture as the video translator. Given an input frame sequence, a content encoder and a style encoder firstly extract its content and style information, respectively. Then, a bidirectional recurrent network propagates the inter-frame content information. Updating this information with the single frame content information, we get the spatio-temporal content information. At last, making use of the conditional instance normalization (Dumoulin et al. (2016), Perez et al. (2018)), the decoder takes the style information as the condition and utilizes the spatio-temporal content information to generate the translation result. An illustration of the proposed architecture can be found in figure 1. By applying the same style code to decode the content feature for a specific translated video, we can produce a long-term consistent video sequence, while the recurrent network helps us combine multi-frame content information to achieve content consistent outputs. The conditional decoder also provides us with a good interface to achieve modality flexible video translation.

Besides using the style dependent content decoder and bidirectional RNNs to ensure long-term and short-term consistency, another advantage of the proposed method lies in our training strategy. Due to our flexible Encoder-RNN-Decoder architecture, the proposed translator can benefit from the highly structured video data and being trained in a self-supervised manner. Concretely, by removing content information from frame t and using posterior style information, we use our Encoder-RNN-Decoder translator to solve the video interpolation task, which can be trained by video sequences in each domain in a supervised manner. In the RecycleGAN method, Bansal et al. (2018) proposed to train video predictors and plugged them into the GAN losses to impose spatio-temporal constraints. They utilize the structured video data in an indirect way: using video predictor trained in a supervised way to provide spatio-temporal loss for training video translator. In contrast, we use the temporal information within the video data itself, all the components, i.e. Encoders, RNNs and Decoders, can be directly trained with the proposed video interpolation loss. The processing pipelines of using our Encoder-RNN-Decoder architecture for the video interpolation and translation tasks can be found in figure 2, more details of our video interpolation loss can be found in section 2.

The main contributions of our paper are summarized as follows:

1. a novel Encoder-RNN-Decoder framework which decomposes content and style for temporally consistent and modality flexible unsupervised video-to-video translation;
2. a novel video interpolation loss that captures the temporal information within the sequence to train translator components in a self-supervised manner;
3. extensive experiments showing the superiority of our model at both video and image level.

2 Proposed UVIT (Unsupervised VIdeo-to-video Translation)

2.1 Problem setting

Let A be the video domain A, $a_{1:T} = \{a_1, a_2, ..., a_T\}$ be a sequence of video frames in A, let B be the video domain B, $b_{1:T} = \{b_1, b_2, ..., b_T\}$ be a sequence of video frames in B. For example, they can be sequences of semantic segmentation labels or scene images. Our general goal of unsupervised video-to-video translation is to train a translator to convert videos between domain A and domain B with many-to-many mappings, so that the distribution of the translated video would be close to that of the real target domain video. More concretely, to generate the style consistent video sequence, we assume each video frame has a style latent variable z. Let $z_a \in Z_A$ and $z_b \in Z_B$ be the style latent variables in domain A and B, respectively. Our target is to align the conditional distribution of translated videos and target domain videos, i.e. $P(b_{1:T}^{trans}|a_{1:T}, z_b) \approx P(b_{1:T}|a_{1:T}, z_b)$ and $P(a_{1:T}^{trans}|b_{1:T}, z_a) \approx P(a_{1:T}|b_{1:T}, z_b)$. The style information can be drawn from the prior or encoded from the style encoder in an example-based way. In addition, taking the prior subset information (rain, snow, day, night, etc.) as label and incorporating that into the style code, we can also achieve deterministic control for the style of the output.

2.2 Translation and Interpolation pipeline

In this work, we assume a shared content space such that corresponding frames in two domains are mapped to the same latent content code just like UNIT (Liu et al., 2017). To achieve the goal of unsupervised video-to-video translation, we propose an Encoder-RNN-Decoder translator which contains the following components:

- Two content encoders CE_A and CE_B, which extract the frame-wise content information in the common spatial content space (e.g., $CE_A(a_t) = l_t$).
- Two style encoders SE_A and SE_B, which encode video frames to the respective style domains (e.g., $SE_A(a_{1:T}) = z_a^{post}$). Here, z_a^{post} is the posterior style latent variable. In practice, we usually take the first frame to conduct style encoding($SE_A(a_1) = z_a^{post}$).
- Two Trajectory Gated Recurrent Units (TrajGRUs) (Shi et al., 2017) $TrajGRU_{forw}$ and $TrajGRU_{back}$, which propagate the inter-frame content information in the forward and the backward direction to form the forward l_t^{forw} and backward l_t^{back} content recurrently.

- One Merge Module $Merge$, which adaptively combine l_t^{forw} and l_t^{back}. Without the l_t from the current frame, it gets the interpolation content l_t^{interp}. Using l_t to update the l_t^{interp}, it gets the translation content l_t^{trans}.
- Two conditional content decoders CD_A and CD_B, which take the spatio-temporal content information and the style code to generate the output frame. It can produce the interpolation frame (e.g., $CD_A(l_t^{interp}, z_a^{post}) = a_t^{interp}$) or the translation frame (e.g., $CD_B(l_t^{trans}, z_b^{prior}) = b_t^{trans}$). Here, z_b^{prior} is the prior style latent variable of domain A drawn from the prior distribution.

Combining the above components, we achieve two conditional video translation mappings: $G_{AB}^{trans} : A \times Z_B \mapsto B^{trans}$ and $G_{BA}^{trans} : B \times Z_A \mapsto A^{trans}$. In order to achieve the style-consistent translation result, we let all the frames in a video sequence to share the same style code z_a (z_b). Besides imposing long-term style consistency, another benefit of the conditional generator is modality flexible translation. By assigning partial dimension of the style code to encode subset labels in the training phase, we are able to control the subset style of the translated video in a deterministic way.

As we propose to use the video interpolation loss to train the translator components in a self-supervised manner, here we also define the video interpolation mappings: $G_A^{interp} : A \times Z_A \mapsto A^{interp}$ and $G_B^{interp} : B \times Z_B \mapsto B^{interp}$. Though the interpolation mapping is conducted within each domain, the interpolation and translation mappings use exactly the same building blocks. An illustration of the translation and interpolation mappings are provided in figure 2.

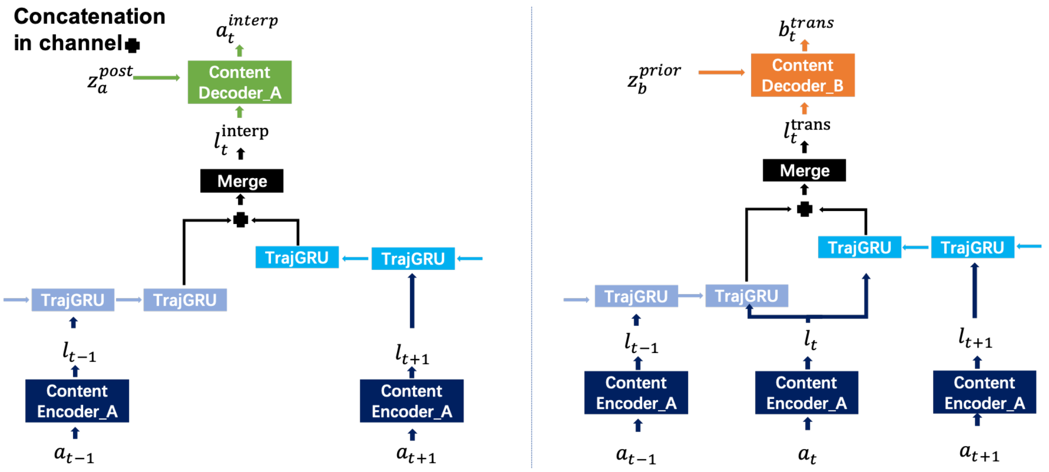

Figure 2: Video interpolation (left) and video translation (right): two processes share modules organically. The input latent content is processed by the Merge Module to merge information from TrajGRUs in both the forward and the backward direction. The translated content (l_t^{trans}) is obtained by updating interpolated content (l_t^{interp}) with the content (l_t) from the current frame (a_t).

2.3 Loss function

2.3.1 Generator loss

Video translation loss. The translated video frames should be similar to the real samples in the target domain. Both the image-level discriminator and the video-level discriminator are added to ensure the image-level quality and the video-level quality. Here we adopt relativistic LSGAN loss (Jolicoeur-Martineau (2018), Mao et al. (2017)). Such loss for domain B can be listed as:

$$L_B^{trans} = \frac{1}{T}\sum_{i=1}^{i=T}[D_B^{img}(b_i^{trans}) - D_B^{img}(b_i) - 1]^2 + [D_B^{vid}(b_{1:T}^{trans}) - D_B^{vid}(b_{1:T}) - 1]^2 \quad (1)$$

Here, $b_{1:T}^{trans}$ are the translated frames from time 1 to T: $b_{1:T}^{trans} = G_{AB}^{trans}(a_{1:T}, z_b^{prior})$. D_B^{img} is the image-level discriminator for domain B , D_B^{vid} is the video-level discriminator for domain B. Translation loss for domain A (L_A^{trans}) is defined in the same way.

Video interpolation loss. The interpolated video frames should be close to the ground truth frames. At the same time, they should be realistic compared to other frames in the domain. This loss term (in domain A) is as follows:

$$L_A^{interp} = \frac{1}{(T-2)}(\lambda_{interp} \parallel a_{2:T-1} - a_{2:T-1}^{interp} \parallel_1 + \sum_{i=2}^{i=T-1} [D_A^{img}(a_i^{interp}) - D_A^{img}(a_i) - 1]^2) \quad (2)$$

Here, because of the characteristic of bidirectional TrajGRUs, only frames from time 2 to $T-1$ are taken to compute the video interpolation loss. $a_{2:T-1}$ are the real frames in domain A, $a_{2:t-1}^{interp}$ are the interpolated frames. The former part of the loss is the supervised L_1 loss. The later part of the loss is the GAN loss computed on the image-level discriminator D_A^{img} in domain A. λ_{interp} is used to control the weight between two loss elements. $B(L_B^{interp})$ is defined in the same way.

Cycle consistency loss. This loss is added to ensure semantic consistency. This loss term (in domain A) is defined as:

$$L_A^{cycle} = \frac{\lambda_{cycle}}{T} \parallel a_{1:T} - a_{1:T}^{rec} \parallel_1 \quad (3)$$

$a_{1:T}^{rec}$ are the reconstructed frames of domain A from time 1 to T by the translation model, $a_{1:T}^{rec} = G_{BA}^{trans}(b_{1:T}^{trans}, z_a^{post})$. The λ_{cycle} is the cycle consistency loss weight. L_B^{cycle} is defined in the same way.

Style encoder loss. To train the style encoder, the style reconstruction loss and style adversarial loss are defined as follows:

$$L_{Z_A}^{style} = \lambda_{rec} \parallel z_a^{rec} - z_a^{prior} \parallel_1 + [D_{Z_A}(z_a^{post}) - D_{Z_A}(z_a^{prior}) - 1]^2 \quad (4)$$

Here, z_a^{rec} is the reconstructed style latent variable of domain A, $z_a^{rec} = SE_A(a_1^{trans})$. λ_{rec} is the style reconstruction loss weight. Such loss for Z_B ($L_{Z_B}^{style}$) is defined in the same way.

2.3.2 Discriminator loss

There are the image-level discriminator loss ($L^{GAN}_{D_A^{img}}$, $L^{GAN}_{D_B^{img}}$), the video-level discriminators loss ($L^{GAN}_{D_A^{vid}}$, $L^{GAN}_{D_B^{vid}}$) and the style latent variable discriminator loss ($L^{GAN}_{D_{Z_A}}$, $L^{GAN}_{D_{Z_B}}$). The detailed loss functions are attached in Appendix A.1.

2.3.3 Full Objective

Our objective for the Generator:

$$L_G^{total} = L_A^{trans} + L_B^{trans} + L_A^{interp} + L_B^{interp} + L_A^{cycle} + L_B^{cycle} + L_{Z_A}^{style} + L_{Z_B}^{style} \quad (5)$$

Here G are the generator modules, which consist of CE_A, CE_B, SE_A, SE_B, $TrajGRU_{forw}$, $TrajGRU_{back}$, $Merge$, CD_A and CD_B.

Our objective for the Discriminator:

$$L_D^{total} = L^{GAN}_{D_A^{img}} + L^{GAN}_{D_B^{img}} + L^{GAN}_{D_A^{vid}} + L^{GAN}_{D_B^{vid}} + L^{GAN}_{D_{Z_A}} + L^{GAN}_{D_{Z_B}} \quad (6)$$

Here, D are discriminator modules, which consist of $D_A^{img}, D_B^{img}, D_A^{vid}, D_B^{vid}, D_{Z_A}, D_{Z_B}$.

We aim to solve the optimization problem:

$$\begin{aligned} G^* &= argmin_G L_G^{total} \\ D^* &= argmin_D L_D^{total} \end{aligned} \quad (7)$$

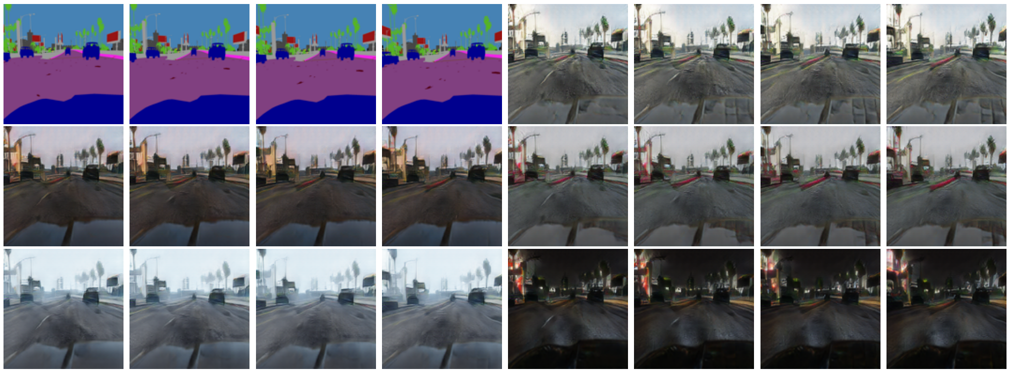

Figure 3: Label-to-image: multi-subset outcomes. Top left: input semantic labels; Top right: translated day video; Middle left: translated sunset video; Middle right: translated rain video; Bottom left: translated snow video; Bottom right: translated night video.

3 Experiments

Implementation details: Our model is trained with 6 frames per batch, with a resolution of 128 × 128. This enables us to train our model with a single Titan Xp GPU. During test time, we follow the experimental setting of Wang et al. (2018a) and load video clips with 30 frames. These 30 frames are divided into 7 smaller sequences of 6 frames with overlap. They all share the same style code to be style consistent. Please note that our model can be easily extended to process video sequences with any lengths. Details of the network architecture are attached in Appendix A.2.

Datasets: We use the Viper dataset (Richter et al., 2017). Viper has semantic label videos and scene image videos. There are 5 subsets for the scene videos: day, sunset, rain, snow and night. The large diversity of scene scenarios makes this dataset a very challenging testing bed for the unsupervised V2V task. We quantitatively evaluate translation performance by different methods on the image-to-label and the label-to-image mapping tasks. We further conduct the translation between different subsets of the scene videos for qualitative analysis.

3.1 Ablation Study

Before comparing the proposed UVIT with state-of-the-art approaches, we first conduct ablation study experiments to emphasize our contributions. We provide experimental results to show the effect of style-conditioned translation and the effectiveness of the proposed video interpolation loss.

Conditional Video Translation: UVIT utilizes an Encoder-RNN-Decoder architecture and adopts a conditional decoder to ensure the generated video sequence to be style consistent. The conditional decoder also provides us with a good interface to achieve modality flexible video translation. In our implementation, we use a 21-dimensional vector as the style latent variable to encode the subset label as well as the stochastic part. By changing the subset label, we are able to control the subset style of the generated video in a deterministic way. Meanwhile, by changing the stochastic part, we can generate various video sequences in a stochastic way. In figure 3, we use the same semantic label sequence to generate video sequences with different sub-domain labels. In figure 4, inducing the same subset label – sunset but changing the stochastic part of the style latent variable, we present different sunset videos generated from the same semantic label sequence. Figure 3 and figure 4 clearly show the effectiveness of the proposed conditional video translation mechanism. Please note that the training of our method does not rely on the subset labels, we incorporate subset labels for the purpose of a deterministic controllable translation. Without the subset labels, we can still generate multimodal style consistent results in a stochastic way.

Video Interpolation Loss: In this part, we provide ablation experiments to show the effectiveness of the proposed video interpolation loss. We conduct ablation studies on both the image-to-label and the label-to-image tasks. Besides comparing UVIT with and without video interpolation loss, we also train UVIT with image reconstruction loss (Huang et al., 2018), which only uses image-level information to train encoder-decoder architectures in a self-supervised manner. We denote UVIT

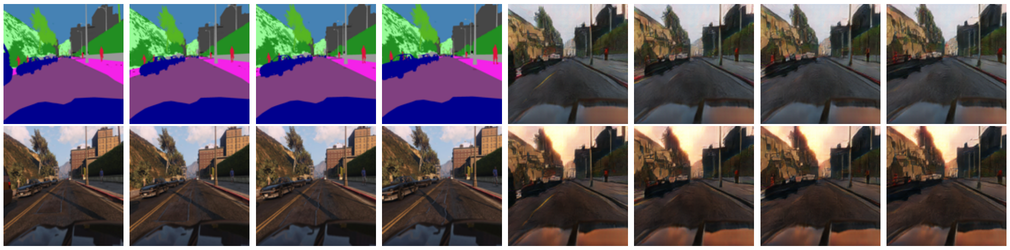

Figure 4: Label-to-image: translation with multimodality. Top left: label inputs; Top right: translated video 1; Bottom left: ground truth; Bottom right: translated video 2. By incorporating different style codes, UVIT could generate different translation results.

Criterion	Model	Day	Sunset	Rain	Snow	Night
mIoU	UVIT (ours)	**12.29**	**12.66**	**13.03**	**11.77**	**9.79**
	UVIT w/o vi-loss (ours)	9.54	10.11	10.72	10.21	8.61
	UVIT w/o vi w ir loss (ours)	8.47	9.01	9.46	8.35	8.34
AC	UVIT(ours)	**17.5**	**17.46**	**17.66**	**16.73**	**14.23**
	UVIT w/o vi-loss (ours)	14.37	15.23	15.24	14.95	12.83
	UVIT w/o vi w ir loss (ours)	12.34	12.74	13.20	12.44	12.29
PA	UVIT(ours)	**62.85**	**61.21**	**62.21**	**59.77**	**56.84**
	UVIT w/o vi-loss (ours)	52.23	53.39	56.35	54.9	50.89
	UVIT w/o vi w ir loss (ours)	53.33	55.15	55.25	52.16	51.05

Table 1: Ablation study: image-to-label (Semantic segmentation). More details can be found in Section 3.1.

trained without video interpolation loss as “UVIT w/o vi-loss” and UVIT trained without video interpolation loss but with image reconstruction loss as “UVIT w/o vi w ir loss”.

We follow the experimental setting of RecycleGAN (Bansal et al., 2018) and use semantic segmentation metrics to evaluate the image-to-label results quantitatively. We report the Mean Intersection over Union (mIoU), Average Class Accuracy (AC) and Pixel Accuracy (PA) achieved by different methods in Table 1. For the label-to-image task, we use the Fréchet Inception Distance (FID) (Heusel et al., 2017) to evaluate the feature distribution distance between translated videos and ground truth videos. The same as vid2vid (Wang et al., 2018a), we use the pretrained I3D (Carreira & Zisserman, 2017) model to extract features from videos. We use the semantic labels from the respective sub-domains to generate videos and evaluate the FID score on all the subsets of the Viper dataset. The FID score achieved by the proposed UVIT and its ablations can be found in Table 2. On both the image-to-label and label-to-image tasks, the proposed video interpolation loss plays a crucial role for UVIT to achieve good translation results. In addition, compared with the image-level image reconstruction loss, video interpolation loss could effectively incorporate temporal information, and delivers better video-to-video translation results.

Criterion	Model	Day	Sunset	Rain	Snow	Night
FID	UVIT (ours)	**14.32**	**15.21**	**18.39**	**16.34**	**16.39**
	UVIT w/o vi-loss (ours)	22.32	20.48	24.02	18.59	18.12
	UVIT w/o vi w ir loss (ours)	18.30	19.32	24.67	19.89	21.29

Table 2: Ablation study: Label-to-image FID. More details can be found in Section 3.1.

3.2 Comparison of UVIT with State-of-the-Art Methods

Image-to-label mapping: We use exactly the same setting as our ablation study to compare UVIT with RecycleGAN in the image-to-label mapping task. The mIoU, AC and PA value by the proposed

UVIT and competing methods are listed in Table 3. The results clearly validate the advantage of our method over the competing approaches in terms of preserving semantic information.

Criterion	Model	Day	Sunset	Rain	Snow	Night	All
mIoU	UVIT (ours)	**12.29**	12.66	**13.03**	**11.77**	**9.79**	**11.94**
	RecycleGAN (Reproduced)[1]	10.31	11.18	11.26	9.81	7.74	10.11
	RecycleGAN (Reported) [2]	8.5	**13.2**	10.1	9.6	3.1	8.9
	Cycle-GAN	3.39	3.82	3.02	3.05	7.76	4.1
AC	UVIT (Ours)	**17.5**	**17.46**	**17.66**	**16.73**	**14.23**	**16.62**
	RecycleGAN (Reproduced)[1]	15.78	15.8	15.95	15.56	11.46	14.84
	RecycleGAN (Reported)[2]	12.6	13.2	10.1	13.3	5.9	12.4
	Cycle-GAN	7.83	8.56	7.91	7.53	11.12	8.55
PA	UVIT (ours)	**62.85**	61.21	**62.21**	**59.77**	**56.84**	**60.51**
	RecycleGAN (Reproduced)[1]	54.68	55.91	57.72	50.84	49.10	53.65
	RecycleGAN (Reported)[2]	48.7	**70.0**	60.1	58.9	33.7	53.7
	Cycle-GAN	15.46	16.34	12.83	13.2	49.03	19.59

Table 3: Quantitative comparison between UVIT and baseline approaches on the image-to-label (Semantic segmentation) task. More details can be found in Section 3.2.

Criterion	Model	Day	Sunset	Rain	Snow	Night
FID	UVIT (ours)	**14.32**	**15.21**	**18.39**	**16.34**	**16.39**
	Improved RecycleGAN	16.50	17.68	23.61	20.01	19.56
	RecycleGAN	17.10	18.87	21.60	25.35	28.28

Table 4: Quantitative comparison between UVIT and baseline approaches on the label-to-image task. More details can be found in Section 3.2.

Criterion	Comparing models	Video level	Image level
Human Preference Score	UVIT (ours) / Improved RecycleGAN	**0.67** / 0.33	**0.66** / 0.34
	UVIT (ours) / 3DCycleGAN	**0.75** / 0.25	**0.70** / 0.30
	UVIT (ours) / vid2vid	0.49 / **0.51**	–
	UVIT (ours) / CycleGAN	–	**0.61** / 0.39

Table 5: Label-to-image: Human Preference Score. Vid2vid is a supervised method and the other methods are unsupervised approaches, more details can be found in Section 3.2.

Label-to-image mapping: In this setting, we compare the quality of the translated video sequence by different methods. We firstly report the FID score (Heusel et al., 2017) on all the sub-domains of the Viper dataset in the same setting as our ablation experiments. As the original RecycleGAN method can not produce long-term style consistent video sequences, we also report the results achieved by our improved version of the RecycleGAN. Concretely, we develop a conditional version which formally controls the style of generated video sequences in a similar way as our UVIT model, and denote the conditional version as improved RecycleGAN. The FID results by different methods are shown in Table 4. The proposed UVIT achieves better FID on all the 5 sub-domains.

To thoroughly evaluate the visual quality of the video translation results, we conduct subjective evaluation on the Amazon Mechanical Turk (AMT) platform. We compare the proposed UVIT with 3DCycleGAN and RecycleGAN. The video-level and image-level human preference scores (HPS) are reported in Table 5. For reference, we also compare the video-level quality between UVIT and the supervised vid2vid model (Wang et al., 2018a). Meanwhile, image-level quality comparison between UVIT and CycleGAN (the image translation baseline) is also included. Table 5 clearly

[1] The result is reproduced by us. The output would be in a resolution of 256×256, we then downscale it to 128×128 to compute the statistics.

[2] This is the result reported in the original paper with a resolution of 256×256.

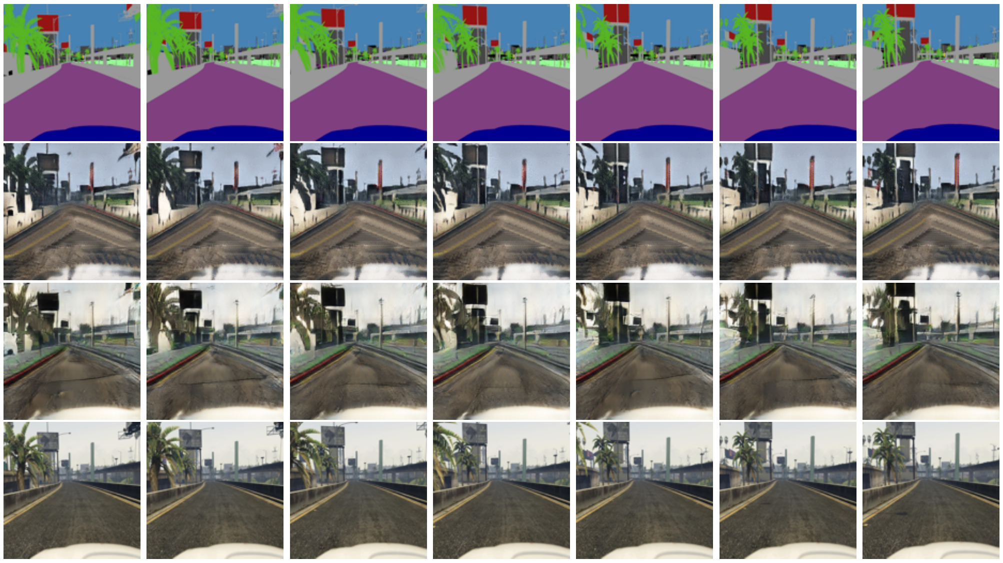

Figure 5: Label-to-image: qualitative comparison. First row: label inputs; Second row: improved RecycleGAN outputs; Third row: UVIT outputs. Fourth row: ground truth.

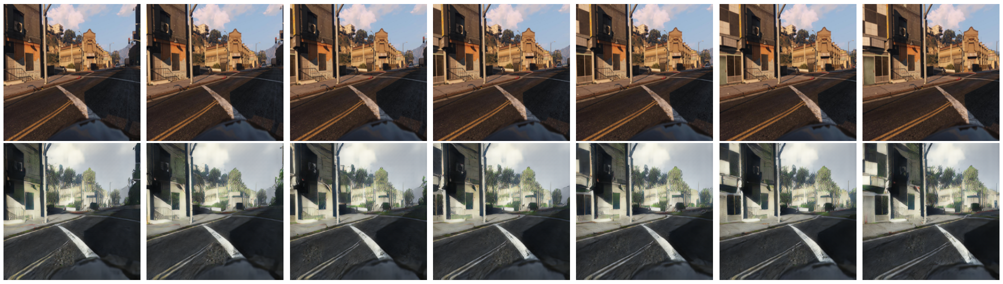

Figure 6: Viper sunset to viper day. Top: input sunset video; Bottom: translated day video.

demonstrates the effectiveness of our proposed UVIT model. In the video-level comparison, our unsupervised UVIT model outperforms the competing unsupervised RecycleGAN and 3DCycleGAN by a large margin, and achieves comparable results with the supervised benchmark. In the image-level comparison, UVIT achieves better HPS than both the V2V competing approaches and the image-to-image baseline. A qualitative example in figure 5 also shows that UVIT model produces a more content consistent video sequence. It could not be achieved by simply introducing the style control without the specialized network structure to record the inter-frame information.

3.3 Other Tasks

Besides translating video sequences between image and label domains, we also train models to translate video sequences between different image subsets and different video datasets. In figure 6, we provide visual examples of video translation from Sunset to Day scenes in the Viper dataset. More results of translation between Viper and Cityscapes (Cordts et al., 2016) datasets can be found in our Appendix.

4 Conclusion

In this paper, we have proposed UVIT, a novel method for unsupervised video-to-video translation. A novel Encoder-RNN-Decoder architecture has been proposed to decompose style and content in the video for temporally consistent and modality flexible video-to-video translation. In addition, we have designed a video interpolation loss which utilizes highly structured video data to train our translators in a self-supervised manner. Extensive experiments have been conducted to show the effectiveness of the proposed UVIT model. Without using any paired training data, the proposed UVIT model is capable of producing excellent multimodal video translation results, which are image-level realistic, semantic information preserving and video-level consistent.

REFERENCES

Amjad Almahairi, Sai Rajeswar, Alessandro Sordoni, Philip Bachman, and Aaron Courville. Augmented cyclegan: Learning many-to-many mappings from unpaired data. *arXiv preprint arXiv:1802.10151*, 2018.

Aayush Bansal, Shugao Ma, Deva Ramanan, and Yaser Sheikh. Recycle-gan: Unsupervised video retargeting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 119–135, 2018.

Dina Bashkirova, Ben Usman, and Kate Saenko. Unsupervised video-to-video translation. *arXiv preprint arXiv:1806.03698*, 2018.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

Yang Chen, Yingwei Pan, Ting Yao, Xinmei Tian, and Tao Mei. Mocycle-gan: Unpaired video-to-video translation. *arXiv preprint arXiv:1908.09514*, 2019.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223, 2016.

Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*, 2016.

Rui Gong, Wen Li, Yuhua Chen, and Luc Van Gool. Dlow: Domain flow for adaptation and generalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2477–2486, 2019.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

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. In *Advances in Neural Information Processing Systems*, pp. 6626–6637, 2017.

Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 172–189, 2018.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint*, 2017a.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017b.

Alexia Jolicoeur-Martineau. The relativistic discriminator: a key element missing from standard gan. *arXiv preprint arXiv:1807.00734*, 2018.

Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *Advances in neural information processing systems*, pp. 700–708, 2017.

Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2794–2802, 2017.

Sangwoo Mo, Minsu Cho, and Jinwoo Shin. Instagan: Instance-aware image-to-image translation. *arXiv preprint arXiv:1812.10889*, 2018.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Stephan R Richter, Zeeshan Hayder, and Vladlen Koltun. Playing for benchmarks. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2213–2222, 2017.

Andrés Romero, Pablo Arbeláez, Luc Van Gool, and Radu Timofte. Smit: Stochastic multi-label image-to-image translation. *arXiv preprint arXiv:1812.03704*, 2018.

Xingjian Shi, Zhihan Gao, Leonard Lausen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun Woo. Deep learning for precipitation nowcasting: A benchmark and a new model. In *Advances in neural information processing systems*, pp. 5617–5627, 2017.

Chuan Wang, Haibin Huang, Xiaoguang Han, and Jue Wang. Video inpainting by jointly learning temporal structure and spatial details. *arXiv preprint arXiv:1806.08482*, 2018a.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8798–8807, 2018b.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2223–2232, 2017a.

Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *Advances in Neural Information Processing Systems*, pp. 465–476, 2017b.

A Appendix

A.1 Detailed loss function for the discriminator

Image level discriminator loss. This loss term (for D_A^{img} in domain A) is defined as follows:

$$\begin{aligned} L_{D_A^{img}}^{GAN} = \frac{1}{2(T-2)} \sum_{i=2}^{i=T-1} [D_A^{img}(a_i) - D_A^{img}(a_i^{interp}) - 1]^2 \\ + \frac{1}{2T} \sum_{i=1}^{i=T} [D_A^{img}(a_i) - D_A^{img}(a_i^{trans}) - 1]^2 \end{aligned} \tag{8}$$

$L_{D_B^{img}}^{GAN}$ for domain B is defined in the same way.

Video level discriminator loss. This loss term (for D_A^{vid} in domain A) is defined as follows:

$$L_{D_A^{vid}}^{GAN} = [D_A^{vid}(a_{1:T}) - D_A^{vid}(a_{1:T}^{trans}) - 1]^2 \tag{9}$$

$L_{D_B^{vid}}^{GAN}$ for domain B is defined in the same way.

Style latent variable discriminator loss. This loss term (for D_{Z_A} in style domain A) is defined as follows:

$$L_{D_{Z_A}}^{GAN} = [D_{Z_A}(z_a^{prior}) - D_{Z_A}(z_a^{post}) - 1]^2 \tag{10}$$

$L_{D_{Z_B}}^{GAN}$ for style domain B is defined in the same way.

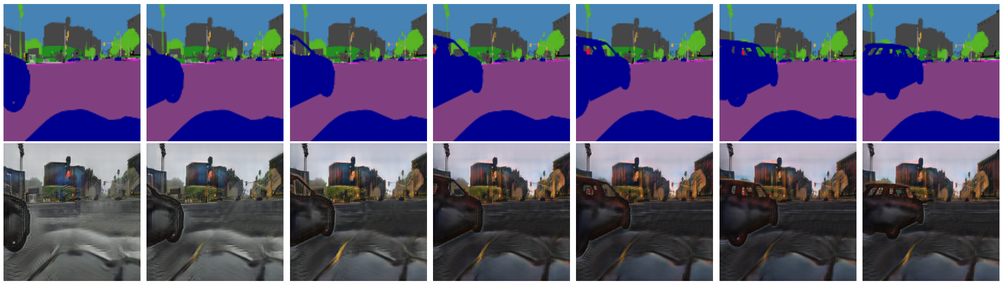

Figure 7: Style inconsistency of RecycleGAN. Top: label inputs; Bottom: RecycleGAN outputs. It is evident that the frames produced by RecycleGAN fail to be style consistent. The first frame is in the rain scenario, the following frames gradually turns to the sunset scene images.

A.2 Network structure

The Content Encoder and Content Decoder are similar to the Conditional Instance Normalized (CIN) ResnetGenerator used in the MUNIT (Huang et al., 2018). For the Content Encoder, after the 3 layers downsampling Convolutional Neural Networks (CNNs) followed by Instance Normalization (IN), we add 3 Resnet Blocks. For the Content Decoder, following the 3 CIN-Resnet Blocks, 3 layers upsampling CNNs are added with CIN.

The Trajectory Gated Recurrent Units (TrajGRUs) (Shi et al., 2017) can actively learn the location-variant structure in the video data. It uses the input and hidden state to generate the local neighborhood set for each location at each time, thus warping the previous state to compensate for the motion information. We take two TrajGRUs to propagate the inter-frame information in both directions in the shared content space.

For the image-level discriminators D_A^{img} and D_B^{img}, the architecture is based on the the PatchGANs (Isola et al., 2017a). Video-level discriminators D_A^{vid} and D_B^{vid} are are also similar to PatchGANs, but with 3D convolutional filters. The style latent variable discriminators D_{Z_A} and D_{Z_B} are used with the same architecture as Augmented CycleGAN by Almahairi et al. (2018).

A.3 Additional training details

With the video being in a resolution of 128 × 128, we use a single Titan Xp GPU to train our network for 3 to 4 days to get a mature model. Due to the GPU memory limitation, the batch size is set to be one. Currently, the frame per clip is 6. Feeding more frames per clip may improve the ability of our model to capture the content dependency in a longer range. However, it requires more GPU memory. The same requirement holds if we want to achieve a higher resolution and display more details.

A.4 Additional experiment results

An example of style inconsistency of RecyceGAN is shown in figure 7. A qualitative example of the mapping between images and labels can be found at figure 8, which shows that our UVIT model can output semantic preserving and consistent segmentation labels. More results on the label-to-image mapping comparison of UVIT and Improved RecycleGAN are plotted in figure 9 and figure 10. More results on label sequences to image sequences with multimodality are plotted in figure 11. The Cityscapes (Cordts et al. (2016)) dataset has real-world street scene videos. As a supplement, we conduct qualitative analysis on the translation between scene videos of Cityscapes and Viper dataset. The result is organized in figure 12.

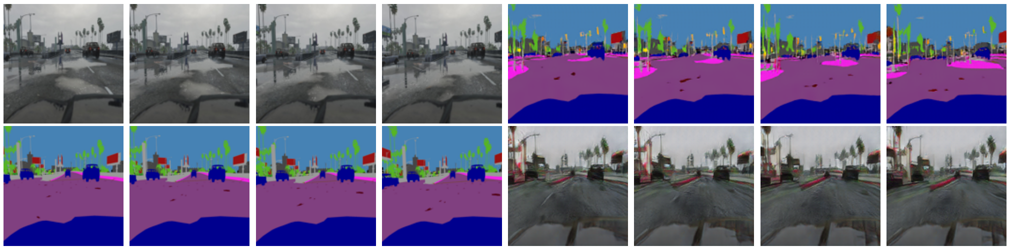

Figure 8: Image-to-label and lable-to-image. Top left: real images inputs; Top right: generated semantic labels; Bottom left:real semantic label inputs; Bottom right: generated images.

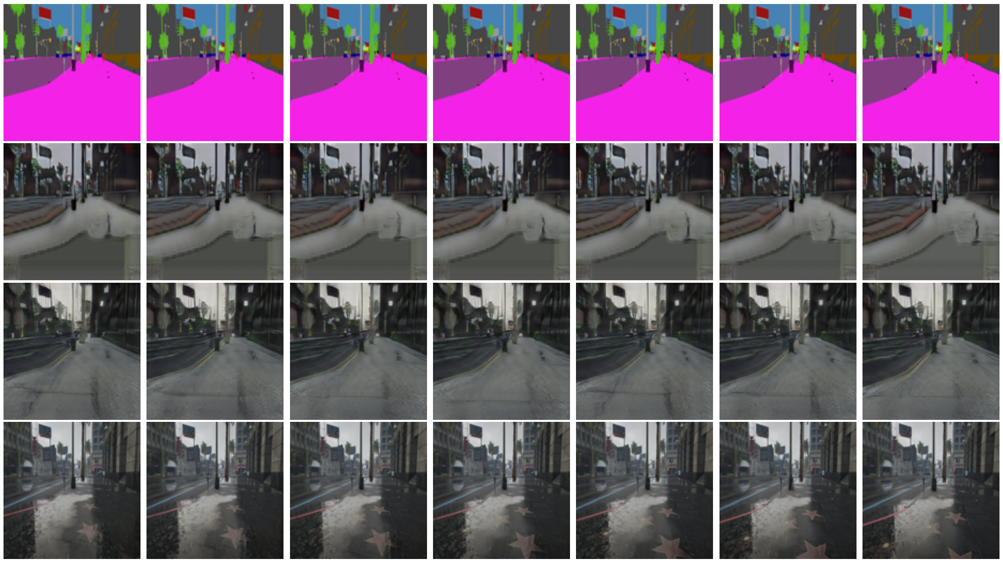

Figure 9: Label-to-image: qualitative comparison. First row: label inputs; Second row: improved RecycleGAN outputs; Third row: UVIT outputs. Fourth row: ground truth. Incomplete translation of the large road in improved RecycleGAN outputs becomes better for the UVIT model.

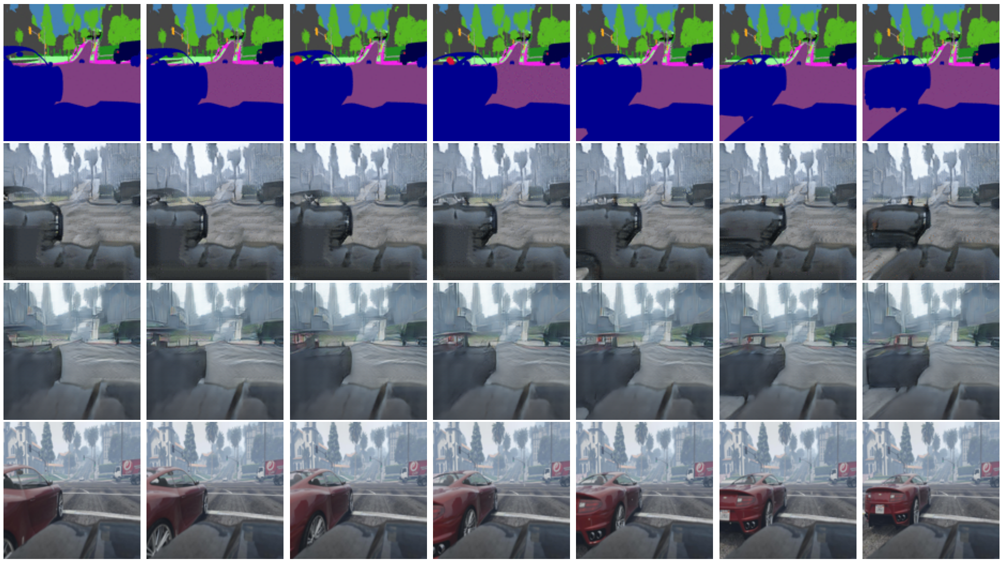

Figure 10: Label-to-image: qualitative comparison. First row: label inputs; Second row: improved RecycleGAN outputs; Third row: UVIT outputs. Fourth row: ground truth. The boundary between different cars is clearer in the UVIT model output.

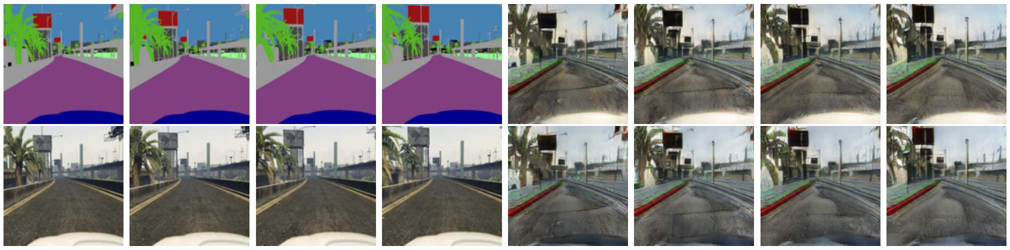

Figure 11: Label-to-image: translation with multimodality. The translated day sequences have different car style versions. Top left: label inputs; Top right: translated video 1; Bottom left: ground truth; Bottom right: translated video 2.

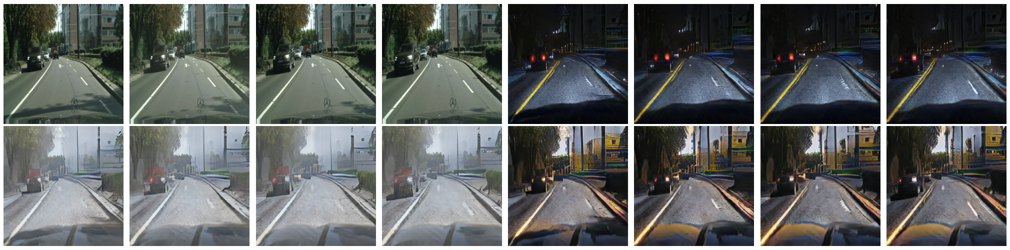

Figure 12: Cityscapes to Viper translation. Top left: input Cityscapes video; Top right: translated Viper video in the night scenario; Bottom left: translated Viper video in the snow scenario; Bottom right: translated Viper video in the sunset scenario.