

Zoom-In-to-Check: Boosting Video Interpolation via Instance-level Discrimination

Liangzhe Yuan^{1*} Yibo Chen^{1*} Hantian Liu¹ Tao Kong^{1,2} Jianbo Shi¹

¹University of Pennsylvania ²Tsinghua University

{lzyuan, yibochen, lhantian, jshi}@seas.upenn.edu, taokongcn@gmail.com

Abstract

We propose a light-weight video frame interpolation algorithm. Our key innovation is an instance-level supervision that allows information to be learned from the high-resolution version of similar objects. Our experiment shows that the proposed method can generate state-of-the-art results across different datasets, with fractional computation resources (time and memory) of competing methods.

Given two image frames, a cascade network creates an intermediate frame with 1) a flow-warping module that computes coarse bi-directional optical flow and creates an interpolated image via flow-based warping, followed by 2) an image synthesis module to make fine-scale corrections. In the learning stage, object detection proposals are generated on the interpolated image. Lower resolution objects are zoomed into, and the learning algorithms using an adversarial loss trained on high-resolution objects to guide the system towards the instance-level refinement corrects details of object shape and boundaries.

1. Introduction

High fidelity video frame interpolation has usages in novel-view rendering, video compression, and frame rate conversion. Existing methods focus on achieving overall high-quality interpolation averaged over all regions of images. The lack of explicitly object instances modeling became the bottleneck for algorithm's improvement.

Flow-based image synthesis algorithms [1, 10, 19, 22] generate realistic colors and patterns by explicitly copying pixels from given frames. For challenging scenes with occlusion, complex deformation or fast motion, flow-based interpolation suffers due to inaccuracy in optical flow estimation algorithms. To compensate for optical flow error, [19, 22] added an additional network to refine the interpolation results, at a cost of much higher computational cost.

*indicates equal contribution

Supplementary video: https://youtu.be/q_wIRq26DY.



Figure 1: Object detection proposals allow region-of-interest (RoI) Zoom-In-to-Check. An adversarial discriminator is trained using high-resolution objects across the entire video against the current interpolated image region. From top to bottom: synthesized image by image synthesis loss only; by whole image adversarial discriminator; by proposed instance-level discriminator; and the ground truth.

A kernel-based interpolation approach achieves the same per-pixel mapping goal without requiring a precise per-pixel flow estimation. The size of blending kernels in a such

method directly restricts the motion that the network is able to capture. To capture larger motion, big kernels (51×51) are used in [20], which results in heavy memory and computation resource usage.

To hallucinate pixels on dis-occluded objects [15] or sharpen motion blurred objects [13] often a Generative Adversarial Network (GAN) is used. However, such generative models are susceptible to mode collapse, resulting in overfitting issues: when an object is blurry, it favors removing the object altogether.

We propose a lightweight video synthesis framework that takes advantage of a newly proposed instance-level adversarial training. Our system consists of a two-stage interpolation network: a cascade design with a flow-based module followed by a kernel-based module. The design substantially alleviates the computational resource at the inference stage, as it requires neither large scale network needed to estimate accuracy optical flow, nor large size kernel needed to preserve clean boundary and capture large motion.

From our experiments, we found image-level supervision has a tendency to remove object details, particularly when the optical flow is fuzzy. To alleviate this issue, we propose an instance-level discriminator to focus our system on the fine details of individual objects. However, if the ‘ground-truth’ reference images also lack details due to lower resolution or motion blur, there is no sufficient feedback to the network on how to correct its mistakes. Our key observation is that in the video we often have similar objects that appear at high-resolution with greater details. This allows the algorithm to learn not just from the current reference frame, but also from semantically similar objects at higher-resolution.

This design allows our network to leverage instance-level attention in learning and thus performs better in challenging scenes. To the best of our knowledge, we are the first to present an instance-level adversarial learning framework that effectively exploits the network’s capacity and achieves an accuracy-speed trade-off for video interpolation task. Using 78% computational time and 21% model parameters of SepConv [21], we achieve state-of-the-art interpolation quality.

2. Related Work

Optical flow estimation is a basic building block for video frame interpolation [10, 19, 29, 31]. In fact, the image interpolation quality has been used to evaluate the accuracy of optical flow estimation [1]. With rapidly improving quality of optical flow estimation, state-of-the-art optical flow methods [3, 8, 26] can serve as a strong baseline for video interpolation. The drawbacks for flow-based video interpolation include 1) producing artifacts around object boundary due to lack of occlusion reasoning, 2) training optical flow estimators requires task-specific datasets, and 3) the overall

algorithm is not end-to-end trainable.

One line of research focused on integrating optical flow into an end-to-end trainable video interpolation framework. Liu *et al.* [31] developed a network to extract per-pixel 3D optical flow vector across space and time in the input video. The intermediate image is generated by trilinear interpolation across the input video volume. The method obtains high-quality results in frame interpolation and their unsupervised flow estimation results are comparable to the state-of-the-art. However, [31] tends to fail when the scene contains repetitive patterns. The work by Jiang *et al.* [10] addressed the issue of occlusion by estimating bidirectional flow together with visibility mask, followed by a flow refinement network. Niklaus *et al.* [19] addressed the issue of inaccuracy of optical flow by retaining pixel-wise contextual information extracted from ResNet18 [7], and employ a synthesis network with a GridNet [4] architecture to generate the interpolated frame.

Moving away from optical flow based methods, [17, 21, 20] eliminated the need of per-pixel explicit motion estimation. Meyer *et al.* [17] propagate predicted phase information across oriented multi-scale pyramid levels to cope with large motions. Niklaus *et al.* [20] estimate a spatially-adaptive convolution kernel for pixel synthesis for interpolation of two input frames. Although this method enables high-quality video frame interpolation, it is difficult to estimate all the kernels at once and the interpolation process is very memory intensive. In [21], the authors improve the efficiency by approximating a 2D kernel with a pair of 1D kernels. This work relieves the intensive memory requirement but the fundamental limitation still exists, where the capability of capturing large motion and flexibility in frame resolution is still limited by the kernel size, which is prohibitively expensive to increase.

A related but harder task is video frame extrapolation. This task contains a similar challenge of motion estimation and object completion on dis-occluded regions. Earlier approaches use variational models that can represent the inherent uncertainty in prediction. Mathieu *et al.* [16] developed a multi-scale conditional GAN architecture to improve the prediction. These methods suffer from blurriness and contain artifacts for large motion. Vondrick *et al.* [27] train a two-stream adversarial network that untangles foreground from background to predict into the future. Lee *et al.* [14] propose a stochastic video prediction model based on VAE-GAN for object synthesis and completion. Several recent works seek to learn a transformation from past pixels to the future directly. [28] untangles the memory of the past from the prediction of the future by learning to predict sampling kernels. [22] combines flow-based and kernel-based approaches to learn a model to predict a motion vector and a kernel simultaneously for each pixel.

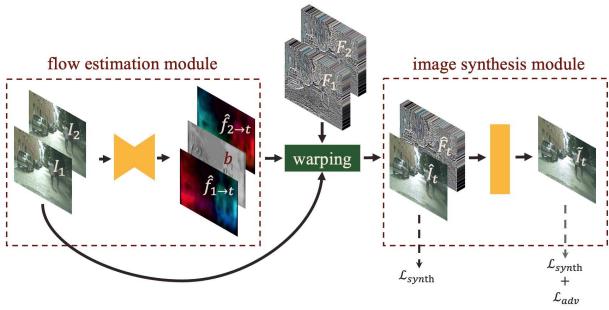


Figure 2: An overview of our model. The flow estimation module (left) takes two frames: \mathbf{I}_1 and \mathbf{I}_2 as input. It predicts the bidirectional optical flows $f_{1 \rightarrow t}$ and $f_{2 \rightarrow t}$ for coarse motion estimation, and a blending mask b for occlusion reasoning. The image synthesis module (right) takes images $\mathbf{I}_1, \mathbf{I}_2$, corresponding features $\mathbf{F}_1, \mathbf{F}_2$, estimated optical flows $f_{1 \rightarrow t}, f_{2 \rightarrow t}$ and blending mask b to synthesize target frame $\tilde{\mathbf{I}}_t$. Instance-level adversarial discrimination is further added on $\tilde{\mathbf{I}}_t$ to preserve sharper image details.

3. Method

3.1. Coarse Optical Flow Estimation

To compensate large displacement motion, we first estimate coarse optical flow to generate an initial interpolated frame $\hat{\mathbf{I}}_t$ given two consecutive video frames \mathbf{I}_1 and \mathbf{I}_2 . We use a U-Net like network to estimate bidirectional optical flows $f_{1 \rightarrow t}$ and $f_{2 \rightarrow t}$, which can be used to warp \mathbf{I}_1 and \mathbf{I}_2 respectively to designated time $\hat{\mathbf{I}}_t$. In the meantime, our network also predicts a per-pixel weighting mask b to blend two synthesized images into one. The blending mask b here can be seen as a confidence mask and it's designed to deal with occlusion. Inspired by [19], we employ a pre-trained feature extractor to extract high level features from both \mathbf{I}_1 and \mathbf{I}_2 , denoted as \mathbf{F}_1 and \mathbf{F}_2 respectively. Note that empirically the flow-based methods present a satisfying performance on most of the regions but often fail to cope with fine-grain details and complex motions. Thus our flow estimation module only serves as an initial step for video interpolation task.

3.2. Image Synthesis Module

We perform both pixel-level and semantic feature-level warping as shown in Fig.2. In detail, we feed images $\mathbf{I}_1, \mathbf{I}_2$, corresponding deep feature maps $\mathbf{F}_1, \mathbf{F}_2$, flows $f_{2 \rightarrow t}$, $f_{1 \rightarrow t}$ and mask b into later module for further refinement. In the image synthesis module, we use estimated bi-directional flow $f_{2 \rightarrow t}, f_{1 \rightarrow t}$ and blending mask b to warp both images and features into time t by bi-linear interpolation [9].

$$\hat{\mathbf{I}}_t = b \odot g(\mathbf{I}_1, f_{1 \rightarrow t}) + (1 - b) \odot g(\mathbf{I}_2, f_{2 \rightarrow t}) \quad (1)$$

$$\hat{\mathbf{F}}_t = b \odot g(\mathbf{F}_1, f_{1 \rightarrow t}) + (1 - b) \odot g(\mathbf{F}_2, f_{2 \rightarrow t}) \quad (2)$$

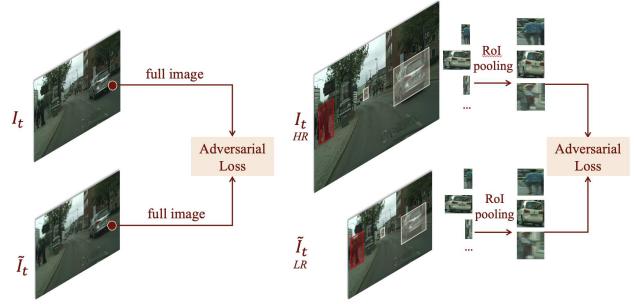


Figure 3: Image level adversarial learning v.s. proposed instance level adversarial learning. We crop RoIs from high resolution images and resize them into constant size patches, which are used to train our low resolution images. This forces the system to focus on refining details and boundaries of instances.

where $g(\mathbf{I}, f)$ is the bi-linear warping function that takes a warping map f to warp a tensor \mathbf{I} to $\hat{\mathbf{I}}$ and \odot is an element-wise multiplication operator. Then we concatenate warped features $\hat{\mathbf{F}}_t$ and image $\hat{\mathbf{I}}_t$ and feed it into the image synthesis layers. Different from [19], in which the author used a giant GridNet [4] to refine the image, we simply use three convolutional layers with kernel size 9 to approximate a large receptive field. We will show that this approximation is enough to get good performance with our proposed instance-level adversarial loss.

3.3. Instance-level Discriminator

Flow-guided warped image $\hat{\mathbf{I}}_t$ generated from the two previous stages has two problems: (a) as the optical flow is trained on the whole image, it often results in twisted and blurry boundaries, as shown in Fig.1; b) optical flow estimation fails to dis-occlude objects in the images, which are common cases the interpolation algorithm needs to deal with. To address the issues, we use the adversarial learning [5] to empower the model on synthesizing instances and recovering structural patterns. In the experiments, we explore two algorithm variations for video interpolation: (a) directly discriminating on the whole image, and (b) zoom-in on object instance area, as shown in Fig.3.

Direct adversarial learning on the whole image makes the generated $\tilde{\mathbf{I}}_t$ looks more realistic compared with the real image \mathbf{I}_t . However, since the majority of the image is usually the background, the image-level supervision provides a uniform gradient across the whole image, such that the semantic details are ignored and the optimization of the foreground is diminished.

The instance-centered learning forces the model to pay more attention to instances, especially on small-scale objects. Given an image \mathbf{I}_t , we use region proposal method [6] to generate several regions of interest (RoI). If we have access to the high-resolution images during training, we crop

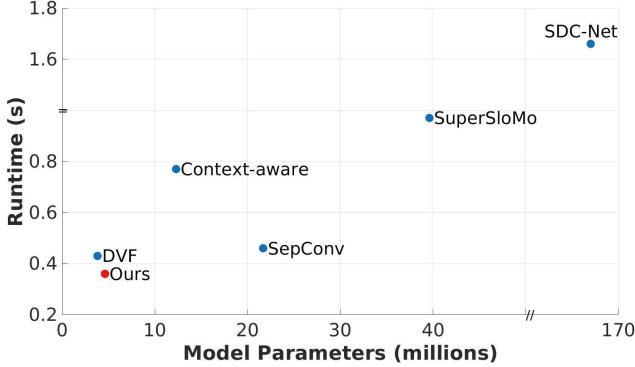


Figure 4: Trained with the proposed instance-level adversarial loss, our model generates the best results with minimal number of parameters and least run-time on interpolating FHD resolution images, comparing to other methods.

the corresponding RoIs from high-resolution images and use them to guide the synthesis of low-resolution results. We perform *RoIAlign* as described in [6] to pool the RoIs from \mathbf{I}_t and $\hat{\mathbf{I}}_t$ into patches with fixed size of $h \times w$. The *RoIAlign* can achieve two effects: a) through bi-linear interpolation, the gradient can be backpropagated to the exact pixel location and previous modules, thus the total network can be updated end-to-end; b) reshaping operation naturally realizes zoom-in effect, balancing network’s focus on close and far away, large and small objects. The reshaped RoIs of different objects are illustrated in Fig.3.

There are two ways to choose how many RoIs per image used for training: we can either choose a fixed number of RoIs with highest response from region proposal network, or use RoIs whose score are above a certain threshold of non-maximum suppression during region proposing. In our experiments, we found the final interpolation quality is not sensitive to the number of RoIs per image used for training. Using 10-30 region proposals per image during training leads to ± 0.002 *STD* of SSIM and ± 0.08 *STD* of IE/PSNR in testing. We empirically choose 16 RoIs per image in the training stage.

A discriminator with spectral normalization [18] is employed to examine only on the specific RoIs instead of on the whole image. The details of adversarial loss \mathcal{L}_{adv} are described in the next section.

3.4. Training Objectives

We use two losses to train the network: a global interpolation loss \mathcal{L}_{int} and an instance adversarial loss \mathcal{L}_{adv} .

$$\mathcal{L} = \mathcal{L}_{int} + \mathcal{L}_{adv} \quad (3)$$

Interpolation Loss. For the global interpolation loss, we first minimize the robust ℓ_1 norm [25] on the per-pixel color difference, which is used in recent self-supervised optical flow estimation work [30]. We further constrain the first-

order gradient difference between interpolated image and the groundtruth to be consistent, which further improves the reconstruction quality [16]. The above photometric losses are computed as

$$\begin{aligned} \mathcal{L}_{ph} = & \rho(\tilde{\mathbf{I}} - \mathbf{I}_{gt}) \\ & + \rho\left(\frac{\partial \tilde{\mathbf{I}}}{\partial x} - \frac{\partial \mathbf{I}_{gt}}{\partial x}\right) + \rho\left(\frac{\partial \tilde{\mathbf{I}}}{\partial y} - \frac{\partial \mathbf{I}_{gt}}{\partial y}\right) \end{aligned} \quad (4)$$

$$\rho(x) = (x^2 + \epsilon^2)^\alpha \quad (5)$$

where $\rho(\cdot)$ is the robust ℓ_1 norm also known as Charbonnier norm.

The second term of the interpolation loss is perceptual loss [11]. It quantifies the network higher-level feature reconstruction quality and thus makes more visually plausible image interpolation results. Our experiments show that the perceptual loss enables the network to learn to reconstruct crispy image boundary. The perceptual loss is defined as

$$\mathcal{L}_{pe} = |\Phi(\tilde{\mathbf{I}}) - \Phi(\mathbf{I}_{gt})|_1 \quad (6)$$

in which the $\Phi(\cdot)$ is the feature extraction function and in our work, we use the latent features from VGG-16 [23]. We apply photometric loss and perceptual loss on both the initial interpolated image $\hat{\mathbf{I}}$ and the synthesized image $\tilde{\mathbf{I}}$. We also constrain the first-order gradient of bi-directional optical flow $f_{1 \rightarrow t}, f_{2 \rightarrow t}$ and the corresponding blending mask b to be locally smooth, resulting in smoothness loss \mathcal{L}_s .

The above loss functions applied on full images mainly guide our network for the coarse level interpolation and we group them as the interpolation loss,

$$\mathcal{L}_{synth} = \lambda_0 \mathcal{L}_{ph} + \lambda_1 \mathcal{L}_{pe} + \lambda_2 \mathcal{L}_s \quad (7)$$

Adversarial Loss. In order to deal with complex scenarios and enlarge model capacity, we utilize another network $D(\cdot)$ to discriminate synthesized images. The adversarial loss consists of two parts, namely the generator loss and the discriminator loss. Let the $(\tilde{\mathbf{P}}_i, \mathbf{P}_i)$ refer to a pair of synthesized and groundtruth RoIs, where $i = 1, \dots, N$. The discriminator will examine each one of them and the adversarial losses are formulated as:

$$\begin{aligned} \mathcal{L}_d = & \frac{1}{N} \sum_{i=1}^N \mathbb{E}[\min(0, -1 - D(\tilde{\mathbf{P}}_i))] \\ & + \mathbb{E}[\min(0, -1 + D(\mathbf{P}_i))] \\ \mathcal{L}_g = & - \frac{1}{N} \sum_{i=1}^N \mathbb{E}[D(\tilde{\mathbf{P}}_i)]; \mathcal{L}_{adv} = \lambda_3 \mathcal{L}_d + \lambda_4 \mathcal{L}_g \end{aligned} \quad (8)$$

3.5. Training Details

The network is trained on a mixer of UCF101 [24] and CityScapes [2] dataset. We randomly pick four triplets in

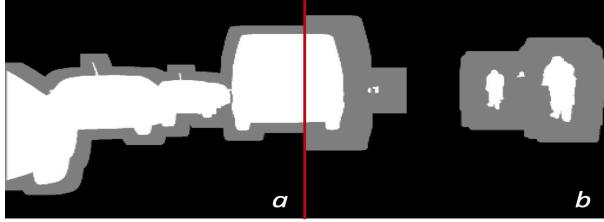


Figure 5: Example of a trimap mask using groundtruth segmentation (white) with 12(a)/24(b) pixels dilation (gray). Evaluation is done on gray and white area.

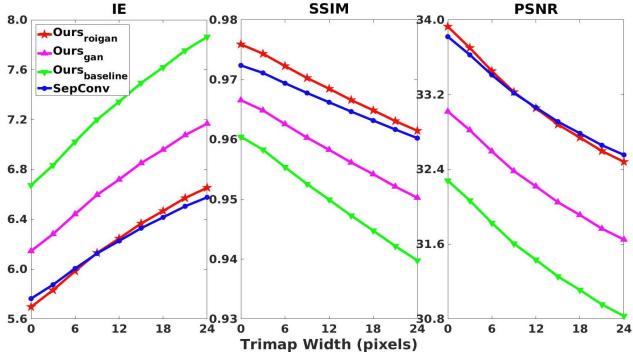


Figure 6: Evaluation metrics on trimaps with various widths on CityScapes dataset. DVF [31] is excluded in the plot for better visualization.



Figure 7: Higher SSIM score from Ours_{roigan} indicates we preserve more perceptual structures than SepConv [21].

every video clip of UCF101 and one triplet in every sequence of CityScapes training set, which gives us around 26k triplets in all. In practice, as our proposed training pipeline is self-contained and does not need labels, any collection of video clips are sufficient to train our network. We keep UCF101 original image size and down-sample CityScapes images to 256×512 . Note that we use the high-resolution version of images in CityScapes dataset to supervise adversarial learning. Forming high-resolution and low-resolution training pair is the key to our learning algorithm. During training, we randomly crop a 256×256 region of triplets as input. We also randomly flip images for data augmentation. The size of output from *RoIAlign* is set to be 64×64 . An Adam optimizer [12] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ is used with initial learning rate $1e-4$, which is decayed exponentially by a factor of 0.1 for every 10 epochs and clipped at $1e-8$ during training. Also, we added decayed random noise to ‘real’ images and

UCF101 [24]	IE	SSIM	PSNR
DVF [31]	11.54	0.869	29.70
SepConv [21]	11.28	0.875	30.29
SuperSloMo [10]	10.87	0.885	30.48
Ours _{baseline}	11.23	0.876	30.08
Ours _{gan}	11.66	0.870	29.85
Ours _{roigan}	10.92	0.882	30.23

CityScapes [2]	IE	SSIM	PSNR
DVF [31]	17.49	0.722	23.88
SepConv [21]	7.85	0.923	30.92
SuperSloMo [10]	—	—	—
Ours _{baseline}	9.38	0.890	29.31
Ours _{gan}	9.04	0.902	29.93
Ours _{roigan}	8.03	0.925	30.77

Table 1: Quantitative evaluation of different methods on CityScapes¹ and UCF101², including Interpolation Error (IE) [1], Peak-Signal-To-Noise (PSNR), and Structural-Similarity-Image-Metric (SSIM). Lower IE and higher SSIM and PSNR indicate better quality.

scheduled to train the discriminator more to smooth the adversarial learning. The weights for different losses are set as $(\lambda_0, \lambda_1, \lambda_2, \lambda_3, \lambda_4) = (1, 1, 0.01, 0.1, 0.01)$.

4. Experiments

To evaluate our method, we quantitatively and qualitatively compare it with several state-of-art video frame interpolation methods. Namely, Deep Voxel Flow (DVF) [31] is a flow warping method for video interpolation; Separable adaptive Convolutions (SepConv) [21] is a kernel based method with the adaptive separable convolutions; SuperSloMo [10] uses a cascaded optical flow estimator to interpolate video frames. We refer Ours_{roigan} as our network trained with proposed instance-level adversarial loss, Ours_{gan} as the model trained with the adversarial loss on overall image, and Ours_{baseline} as the model trained without any adversarial loss.

We compare the algorithm results on two different datasets, CityScapes [2] and UCF101 [24]. CityScapes contains different objects, e.g. cars, people, traffic lights, etc., with various size and distance, which is good to differentiate algorithms’ interpolation abilities on small objects and partial occlusions. UCF101 contains people activities, e.g. boating, making-up, boxing, etc., which is good to show results on fast motion and complex deformation.

¹SuperSloMo [10] is not open-sourced so we don’t have their results on CityScapes dataset.

²We re-run the evaluation on the synthesis images provided by [10].

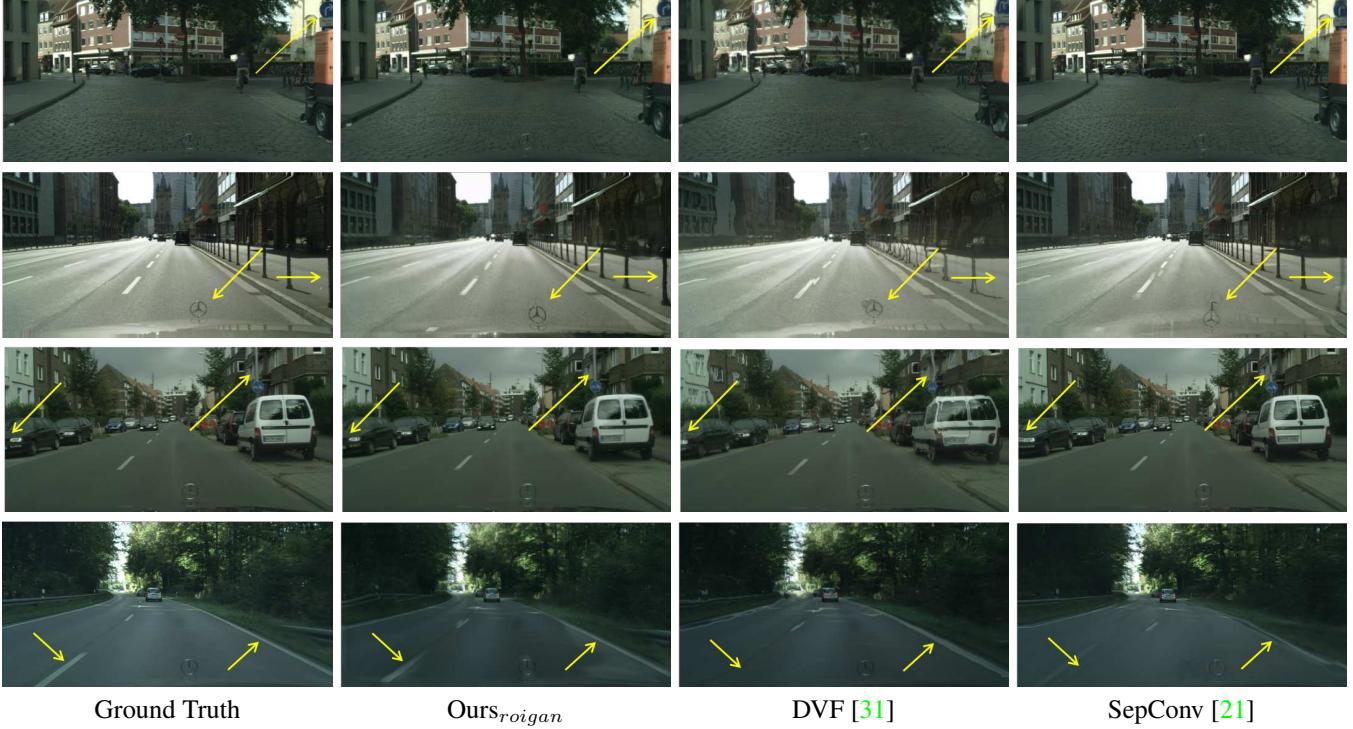


Figure 8: Qualitative results from different methods on CityScapes dataset. Best viewed in color.

4.1. Ablation Studies

Baseline. Considering our baseline is methodologically similar to [19, 31], comparing metrics within our methods ($Ours_{baseline}$, $Ours_{roigan}$) would serve as an ablation study to show the effectiveness of the proposed instance-level discrimination. Results in both Tab.1 and Fig.6 show $Ours_{roigan}$ consistently outperforms or on-par with $Ours_{baseline}$ across datasets. This shows from one aspect that the proposed instance-level discrimination can improve algorithm performance.

Adversarial Training. We verify the advantages of using adversarial learning to improve video interpolation performance. From experiment on both datasets, training with adversarial loss gives us sharper boundaries in images. In Fig.1, we show an example of the effectiveness of adversarial loss. From zoomed-in figures, we can see adversarial loss helps preserve edges and shapes. This can be attributed to the adversarial loss better facilitating image synthesis module’s learning and potentially correcting the inaccurate optical flow estimation.

Interestingly, we find training model with image level adversarial loss would lead to a local-minimal solution sometimes. This phenomenon is especially noted when testing on CityScapes dataset. Since the image-level adversarial training does not explicitly constrain the instances, the network tends to erase the uncertain objects in the scene and recover the background. This is because the data distribu-

tion is dominated by rigid objects and background such that training with image level adversarial loss leads to a biased learning result. In the next part, we discuss the proposed instance-level discrimination which would potentially fix this issue.

Instance-level Discriminator. We further verify the advantages of introducing an attention mechanism in adversarial training, which greatly improves the video interpolation performance as a result. From experiments, it is shown that training with instance-level discrimination gives us sharper boundaries on small, thin objects and image details. With the adversarial loss, both rigid moving object and non-rigid human body shape are preserved better than baseline method with, as we can see in Fig.1.

In Table 1, we show $Ours_{roigan}$ method outperforms $Ours_{baseline}$ and $Ours_{gan}$ method on all three standards in CityScapes dataset. In UCF101 dataset, as the RoI size in the image is quite close to the entire image size, the instance-level discriminator model and full-image-level discriminator model perform considerably similar. Qualitatively, results in Fig.9 still show better interpolation results on instances by our methods, due to the instance-level adversarial training. Noticeably, we also measured all three metrics using the trimaps of ‘human’, ‘vehicles’ groups with various dilation widths in CityScapes dataset to quantitatively illustrate the proposed instance-level discrimination improves synthesized instance quality. Trimaps are generated using the groundtruth segmentation masks, as shown in

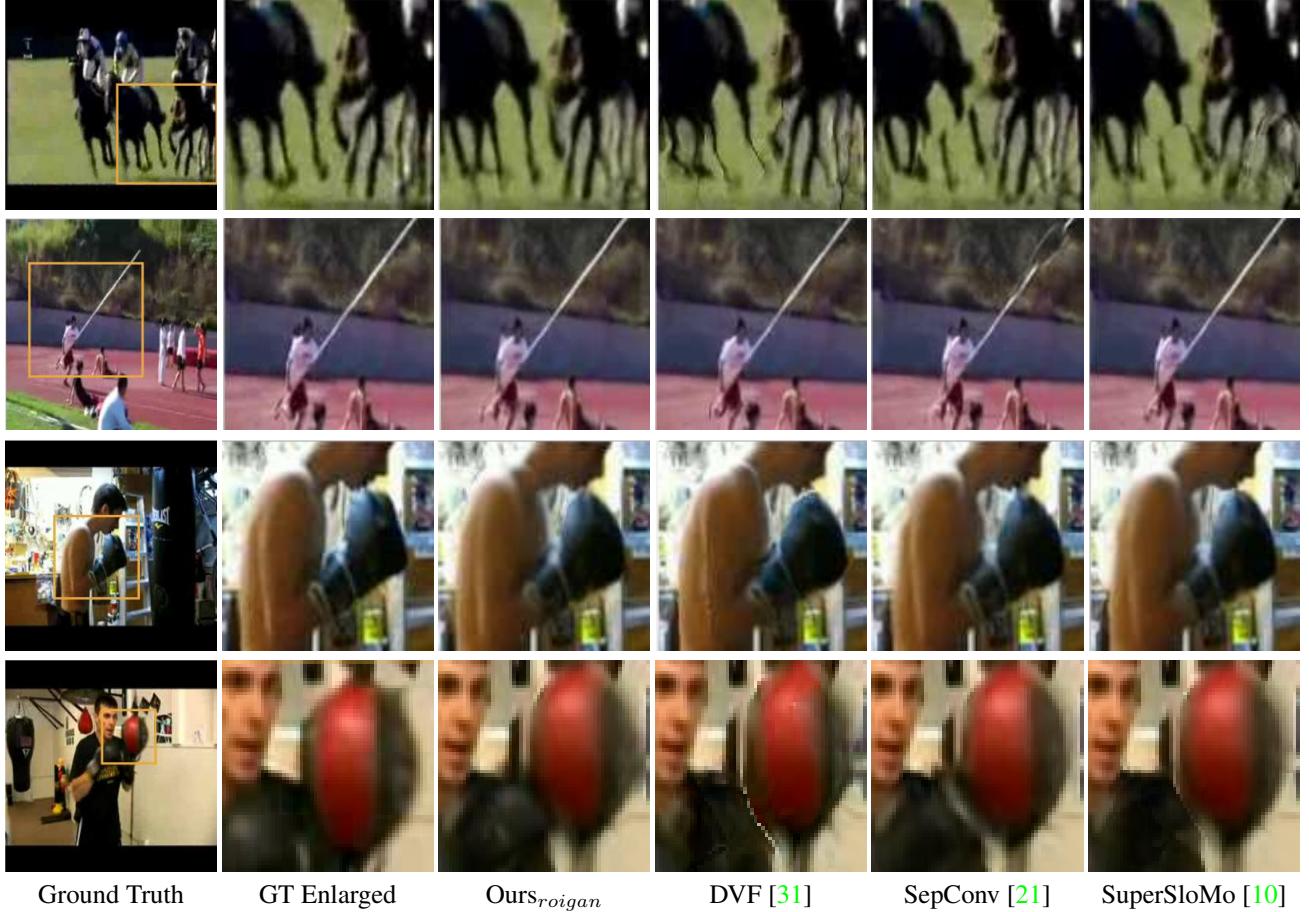


Figure 9: Qualitative results from different methods on UCF101 dataset. Best viewed in color.

Fig.5. In Fig.6, we show our method achieves the best performance on object instances. When dilation width is less than 12 pixels, $Ours_{roigan}$ performs the best. As the trimap width grows to over 12 pixels, more background pixels are included such that $Ours_{roigan}$ performs slightly worse than SepConv [21] on IE and PSNR. Introducing the region proposals and zooming into them force the network to focus on details and to utilize the fine-grained information for learning filters. By formulating video interpolation problem as perturbing semantic objects in image space, the pixel-level motion estimation can be better grouped and updated.

Training with High Resolution Patches. We also study the effects of training with different image resolution. Due to data augmentation and the concerns of training speed, researchers used to down-sample high-resolution images or crop part of images for training. However, high-resolution images often preserve fine-grained information and it can potentially improve algorithms performance. In our model, we train our proposed model with instance-level discriminator on real image patches from high-resolution images. More specifically, based on the region proposals we crop the ‘fake’ RoIs from synthesized images and the corresponding

‘real’ patches from its high-resolution counterpart, forming low-resolution high-resolution pairs. The high-resolution patches ultimately force the generator to super-resolve and synthesize the details on low-resolution images. From Table 1, we show that using high-resolution patches to train the network with region based adversarial training boosts performances beyond both the baseline model and the model using full image adversarial training. Fig.6 also shows training with high-resolution image patches consistently improves interpolation qualities on instances.

4.2. Quantitative Evaluation

We achieve the highest SSIM across datasets consistently, both on the foreground and full images. We compare our approach with state-of-the-art video interpolation methods, including Separable adaptive Convolution (SepConv) [21], and Deep Voxel Flow (DVF) [31] on both UCF101 and CityScapes dataset. As shown in Table 1, our method achieves the best SSIM score on CityScapes dataset. Table 1 also demonstrates the quantitative results on UCF101 dataset, where we also compare with the SuperSloMo [10]. We re-run the evaluation on the images provided by [31]

and [10], and images generated from [21]. All metrics are computed under the motion masks provided by [31], which highlights the capabilities to cope with regions of motion and occlusion. Our method achieves the highest SSIM score among the lightweight models and performs comparably to the heavy model, SuperSloMo [10]. We also show the proposed model achieves the highest SSIM on instances in Fig.6. As the SSIM metric measures perceptual and structural similarity, it serves as a strong cue that the proposed method can render the most realistic scene and structural details as shown in Fig.7.

4.3. Qualitative Results

We also present qualitative comparisons with other methods. In Fig.8, we present the comparison on different street scenes under various lighting condition. It is obvious that DVF[31] generates the most artifacts such as distortion of the whole scene, unrealistic deformation of cars and buildings, misalignment of white lines and etc.. SepConv[21] is capable of dealing with motion within their kernel size, but it consistently results in severe blur and artifacts near the image boundary, as shown in all of our examples. Our proposed approach is particularly good at recovering fine-grained details, for example, the traffic sign in the first example. Also, it fills up the occluded regions in a natural and realistic way, such as the white lines on the road in the fourth example. Fig.9 shows a qualitative comparison on UCF101. It is hard for DVF[31] to handle the occlusion as shown in the second example, although it was trained on UCF101. SepConv[21] is observed to have frequent duplicate artifacts, such as splits of horse legs and vault pole. SuperSloMo[10] performs well in most scenes but sometimes fails in the refinement of details in small scale such as the chin of the boxing player, and legs of running horses. Our proposed method enables the reconstruction of the fine-grained details and thus is capable of interpolating the challenging scenes.

4.4. Discussion

Our network achieves the state-of-art video interpolation results using minimal model parameters and running fastest at inference time, illustrated in Fig.4. During training, we zoom into instances and train our model by discriminating on the rescaled RoIs. The scaling on instances due to the physical distance and the zooming-in helps the network learn more structural and general filters that not only recover crispy boundary on objects but also structural patterns in the background, e.g. pole, traffic sign, etc. even though they are not explicitly trained by the discriminator. Also, inspired by the super-resolution literature, we expect our model learn to super-resolve and render semantic details by training with high-resolution patches. At inference time, only the flow estimation module and image synthesis



Figure 10: Failure cases on CityScapes dataset. Left Column: The network tends to erase objects and recover background to overfit to the training objectives. Right Column: Our model may fail in cluttered scenes.

module are needed, resulting in fast inference time. As a result, it costs our network 0.36s to run on a 1024×2048 image.

Our network still has several limitations. For large non-rigid body movements, the interpolated objects are slightly distorted. As the right column in Fig.10 shows, cluttered scenes will lead to failure case. Large overlapping instances with mutual occlusion makes the system hard to dis-occlude individual objects. Adversarial learning is also likely to overfit to some data points in the training data. For example, when the motion estimation is blurry, the synthesis module tends to remove the uncertainty and choose to safely reconstruct the background, as shown in Fig.10 left column. Finally, larger models with better optical flow estimation would generate better results than ours on a clean and texture-rich area (ground).

5. Conclusions

We demonstrate a lightweight video interpolation framework that can retain instance level object details. We use a flow estimation module to synthesize the intermediate frame followed by a light-weight image synthesis module to correct detailed shape errors. The network is trained by a region based discriminator which utilizes high-resolution image patches to supervise low-resolution RoIs, constraining instances in images to look realistic. Due to the modularity, our proposed adversarial training strategy can be universally used as a training block to improve algorithm performance. In the future, we hope to improve the model design to compensate some drawbacks in our model, e.g. employing deformable convolutions to tackle large motions and complex deformations. We also want to further expand our work to video prediction task.

6. Acknowledgement

We gratefully appreciate support through Honda Research Institute Curious Minded Machine program.

References

[1] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 92(1):1–31, Mar 2011. 1, 2, 5

[2] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 4, 5

[3] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2758–2766, 2015. 2

[4] D. Fourure, R. Emonet, E. Fromont, D. Muselet, A. Tréneau, and C. Wolf. Residual conv-deconv grid network for semantic segmentation. In *Proceedings of the British Machine Vision Conference*, 2017, 2017. 2, 3

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014. 3

[6] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017. 3, 4

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015. 2

[8] 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)*, volume 2, page 6, 2017. 2

[9] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015. 3

[10] H. Jiang, D. Sun, V. Jampani, M. Yang, E. G. Learned-Miller, and J. Kautz. Super slomo: High quality estimation of multiple intermediate frames for video interpolation. *CoRR*, abs/1712.00080, 2017. 1, 2, 5, 7, 8

[11] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, pages 694–711. Springer, 2016. 4

[12] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[13] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*, 2015. 2

[14] A. X. Lee, R. Zhang, F. Ebert, P. Abbeel, C. Finn, and S. Levine. Stochastic adversarial video prediction. *arXiv preprint arXiv:1804.01523*, 2018. 2

[15] X. Liang, L. Lee, W. Dai, and E. P. Xing. Dual motion GAN for future-flow embedded video prediction. *CoRR*, abs/1708.00284, 2017. 2

[16] M. Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. *CoRR*, abs/1511.05440, 2015. 2, 4

[17] S. Meyer, A. Djelouah, B. McWilliams, A. Sorkine-Hornung, M. H. Gross, and C. Schroers. Phasenet for video frame interpolation. *CoRR*, abs/1804.00884, 2018. 2

[18] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018. 4

[19] S. Niklaus and F. Liu. Context-aware synthesis for video frame interpolation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 2, 3, 6

[20] S. Niklaus, L. Mai, and F. Liu. Video frame interpolation via adaptive convolution. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

[21] S. Niklaus, L. Mai, and F. Liu. Video frame interpolation via adaptive separable convolution. In *IEEE International Conference on Computer Vision*, 2017. 2, 5, 6, 7, 8

[22] F. A. Reda, G. Liu, K. J. Shih, R. Kirby, J. Barker, D. Tarjan, A. Tao, and B. Catanzaro. Sdc-net: Video prediction using spatially-displaced convolution. In *The European Conference on Computer Vision (ECCV)*, September 2018. 1, 2

[23] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[24] K. Soomro, A. R. Zamir, and M. Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012. 4, 5

[25] D. Sun, S. Roth, and M. J. Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *International Journal of Computer Vision*, 106(2):115–137, 2014. 4

[26] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8934–8943, 2018. 2

[27] C. Vondrick, H. Pirsiavash, and A. Torralba. Generating videos with scene dynamics. *CoRR*, abs/1609.02612, 2016. 2

[28] C. Vondrick and A. Torralba. Generating the future with adversarial transformers. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2992–3000, July 2017. 2

[29] T.-C. Wang, J.-Y. Zhu, N. K. Kalantari, A. A. Efros, and R. Ramamoorthi. Light field video capture using a learning-based hybrid imaging system. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2017)*, 36(4), 2017. 2

[30] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis. Eventflownet: Self-supervised optical flow estimation for event-based cameras. *arXiv preprint arXiv:1802.06898*, 2018. 4

[31] X. T. Y. L. Ziwei Liu, Raymond Yeh and A. Agarwala. Video frame synthesis using deep voxel flow. In *Proceedings of International Conference on Computer Vision (ICCV)*, October 2017. 2, 5, 6, 7, 8