Pacific Graphics 2019
C. Theobalt, J. Lee, and G. Wetzstein
(Guest Editors)

Volume 38 (2019), Number 7

Appearance Flow Completion for Novel View Synthesis

Hoang Le and Feng Liu

Portland State University, Oregon, USA

Abstract

Novel view synthesis from sparse and unstructured input views faces challenges like the difficulty with dense 3D reconstruction and large occlusion. This paper addresses these problems by estimating proper appearance flows from the target to input views to warp and blend the input views. Our method first estimates a sparse set 3D scene points using an off-the-shelf 3D reconstruction method and calculates sparse flows from the target to input views. Our method then performs appearance flow completion to estimate the dense flows from the corresponding sparse ones. Specifically, we design a deep fully convolutional neural network that takes sparse flows and input views as input and outputs the dense flows. Furthermore, we estimate the optical flows between input views as references to guide the estimation of dense flows between the target view and input views. Besides the dense flows, our network also estimates the masks to blend multiple warped inputs to render the target view. Experiments on the KITTI benchmark show that our method can generate high quality novel views from sparse and unstructured input views.

1. Introduction

Novel view synthesis addresses the problem of generating a new image at a new viewpoint from a set of input views. It has a wide variety of applications, such as video stabilization, teleconference, 3D video, and VR. It is a classic problem in computer graphics and vision and many algorithms are available [KLTS06, Sze10, ZC04].

Traditional novel view synthesis algorithms usually require a dense set of input views to reliably obtain an approximate 3D scene structure and render novel views [BBM*01, CDSHD13, LH96, NLB*, ZKU*04]. When only a sparse set of unstructured input views are provided, many areas in the input views only appear in one view. 3D reconstruction is difficult to perform in these areas. In addition, large baselines between input views also result in significant occlusions, which make 3D reconstruction and view synthesis challenging. Thus, the traditional methods often have difficulty in synthesizing novel views with sparse input views. The recent deep learning methods provide data-driven approaches to novel view synthesis [KWR16, SHL*18, ZTS*16]. Unlike conventional approaches that mostly rely on geometry, they can learn to handle challenging issues like occlusions and produce high-quality novel views. However, these methods still face challenges when the target viewpoint is distant from input ones.

This paper presents a novel view synthesis method that is able to generate novel views at distant viewpoints with sparse and unstructured input views. Our method builds upon the success of the research on sparse 3D reconstruction and the power of deep neural networks for image synthesis. Specifically, state-of-the-art structure from motion algorithms can now reliably perform sparse 3D reconstruction of a scene, often in form of 3D scene points [SF16]. While these 3D scene points alone are not sufficient for distant view synthesis, they can guide novel view synthesis [LGJA09]. Meanwhile, deep neural networks have been shown successful for image synthesis. This work is inspired by the appearance flow methods [SHL*18, ZTS*16] that estimate dense flows to guide the sampling of input images to generate the novel view. These methods provide an elegant way to handle occlusion and produce sharp images. However, they do not explicitly explore 3D geometry and often have difficulty with distant novel view synthesis from sparse input views. We hypothesize that with the guidance from the sparse flow that can be directly computed according to the sparse 3D scene points, distant novel views can be better rendered.

In this paper, we formulate novel view synthesis as an appearance flow completion problem. Specifically, given a sparse set of 3D points, we calculate the sparse flows from the target view to the input ones based on their camera poses. We then develop a deep neural network that takes the sparse flows as well as input views as input and outputs dense appearance flows and the mask maps. These outputs are used to warp and blend input views to generate the novel view. To further improve the estimation of the dense appearance flows, we calculate the sparse scene flow between input views as well as the dense flow between them, which can be readily calculated using a state-of-the-art optical flow method as the input views are known. The pair of sparse scene flow and dense optical flow provides a reference to guide the neural network to transform the sparse flows between the unknown target view and the input views to the dense flows. Compared to existing methods [SHL*18, ZKSE16], having such a reference makes it easy for the neural network to learn to correctly warp and blend input views by explicitly providing dense pixel correspondences between them and create high-quality views without blurring or ghosting artifacts.

We train and evaluate our deep neural network on the KITTI benchmark [GLU12]. Our experiments show that our method can generate high quality novel views from sparse and unstructured input views. The strength of our method comes from the two contributions of this paper to novel view synthesis. The first is the combination of the explicit use of sparse geometry and the power of our deep network for image synthesis, implemented in a deep neural network that uses a sparse set of 3D scene points to guide dense appearance flow estimation. The second is our idea of using a pair of sparse flow and dense flow between known views to help transform the sparse flows between the unknown target view and the input views to the dense appearance flows.

2. Related Work

Novel view synthesis is a classic problem in computer vision and graphics. A good survey can be found in [KLTS06, Sze10, ZC04] on traditional non-deep learning methods for novel view synthesis. These methods often require dense 3D reconstruction of the scene or its sparse proxy to generate the novel view [BBM*01, CDSHD13, LH96, NLB*, ZKU*04]. Many methods are available to handle the scenario when 3D reconstruction is unreliable. For example, Fitzgibbon *et al.* transformed the problem of reconstructing the 3D scene geometry to that of reconstructing the color to handle textureless regions and employed an image-based prior on the reconstruction to generate realistic synthetic views [FWZ05]. Recently, Penner and Zhang used a soft 3D representation to preserve depth uncertainty in the stages of 3D reconstruction and rendering. This soft 3D reconstruction enables high-quality, continuous, and robust novel view rendering [PZ17]. This paper aims to handle the challenging scenario of very sparse input views and significant occlusions and presents a deep neural network method that estimates appearance flows to render a novel view instead of relying on dense 3D reconstruction.

Our work is related to the recent methods on deep learning-based novel view synthesis methods. In their seminal work, Dosovitiskiy *et al.* [DSB15], Kulkarni *et al.* [KWKT15], Yang *et al.* [YRYL15], and Tatarchenko *et al.* [TDB16] investigated the use of deep neural networks for novel view synthesis. Their methods take as input a set of images of objects and render unseen views of the objects. Recently, Thies *et al.* developed an image guided neural object rendering method that decomposes images into view-dependent effects and diffuse images. Their method is able to generate novel views with highly realistic view-dependent appearance and minimizes the boundary and occlusion artifacts [TZT*18]. Compared to these methods, our work aims to generate novel views for general scenes instead of objects.

Our work is particularly relevant to the deep learning-based novel view synthesis methods for general scenes. The DeepStereo method from Flynn *et al.* builds a plane-sweep volume for each input image and then trains a deep neural network to blend them to generate a novel view [FNPS16]. Liu *et al.* explored the 3D geometry to synthesize a novel view by approximating a real-world scene with a fixed number of planes [LHS18]. Hedman *et al.* developed a deep neural network to learn to blend multiple warped input views [HPP*18]. Similarly, our method also learns to blend warped input views; however, our work learns to estimate dense appearance flows to warp input views instead of relying on multi-view stereo algorithms. Several recent deep learning methods have been presented to address the challenges of imperfect 3D reconstructions in image-based rendering. Thies *et al.* developed a deferred neural rendering paradigm that learns a novel neural texture representation, which is used by their neural rendering pipeline to produce realistic images given imperfect 3D input, enabling a wide variety of applications, such as novel view synthesis, scene editing, and animation synthesis [TZN19]. Sitzmann *et al.* presented a method to learn 3D feature embeddings, called DeepVoxels, to encode a posed view of a scene without explicitly modeling its geometry. Such 3D feature embeddings allow for consistent novel view synthesis [STH*19]. Compared to these methods, our work aims to address the extreme cases where input images are so sparse that

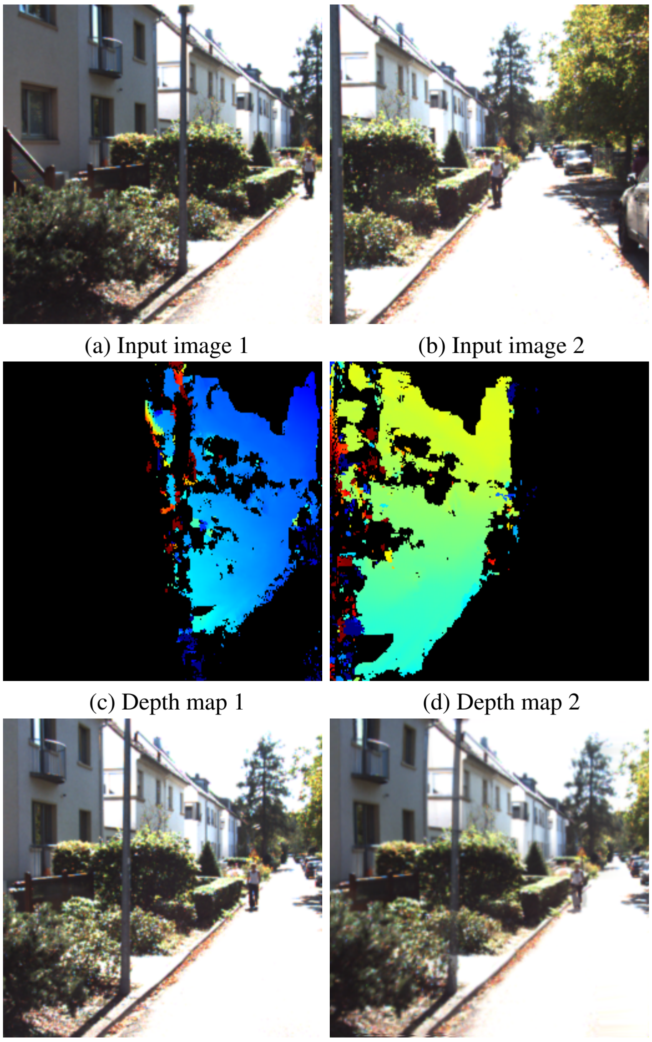

(a) Input image 1
(b) Input image 2
(c) Depth map 1
(d) Depth map 2
(e) Ground truth
(f) Our interpolation result

Figure 1: *Novel view synthesis from sparse input views. The large baseline between two input views makes the overlapping region small and leads to significant occlusion, which makes it difficult to obtain high-quality dense depth maps for the whole scene using a state of the art method, such as COLMAP [SZPF16]. Our method does not rely on dense 3D reconstruction and is able to generate high quality novel views.*

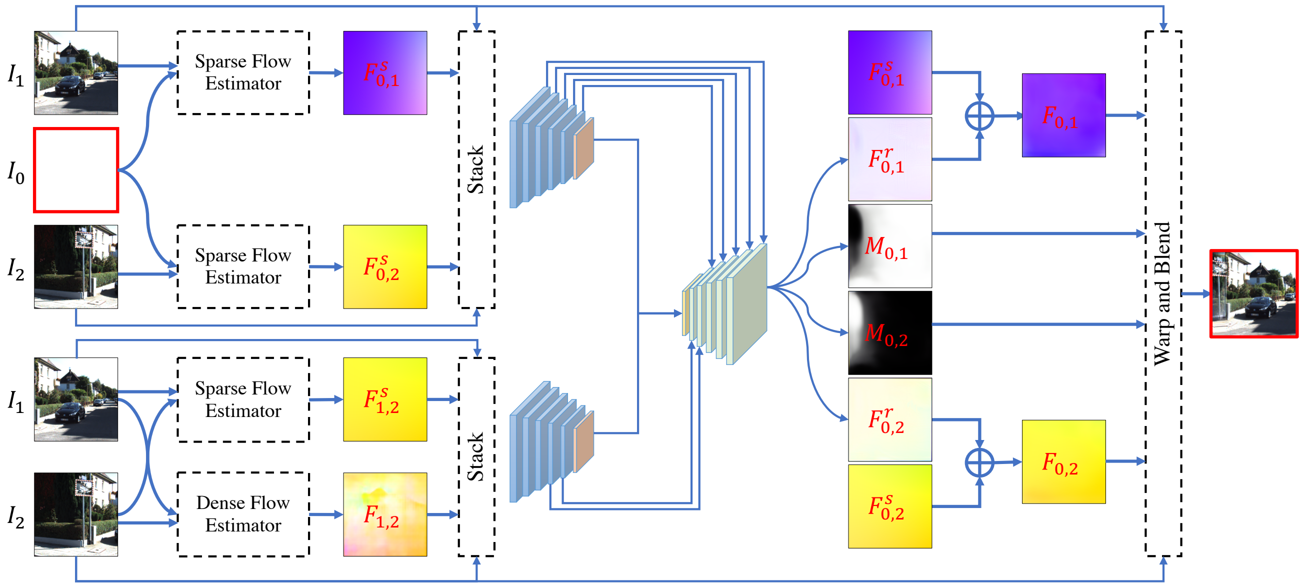


Figure 2: *The architecture of our deep novel view synthesis neural network.*

dense 3D estimation is almost impossible in a large portion of a scene. We, accordingly, propose to leverage an optical-flow-based approach and combine it with sparse 3D estimation to enable novel view synthesis for these extreme cases.

Our method is inspired by the work from Zhou *et al.* that estimates appearance flows for view synthesis [ZTS*16]. Recently, Sun *et al.* extended the appearance flow method to handle an arbitrary number of input views [SHL*18]. Our method also adopts the formulation of appearance flow for novel view synthesis. Our method explicitly incorporates sparse geometry into the estimation of dense appearance flow and can more reliably render a distant novel view. Moreover, we also explore the pair of sparse flow and dense flow between known views to guide the estimation of dense appearance flows from the unknown view to the input views.

Finally, deep neural networks have been shown successful for some specific novel view synthesis tasks. For instance, deep neural network algorithms are now able to interpolate high-quality frames, even at as a high resolution as 4K [BLM*19, JSJ*18, LYT*17, LLLC19, MDM*18, NL18]. Kalantari *et al.* developed a two-stage deep convolutional neural network that can expand views for light field imaging [KWR16]. In their recent work, Zhou *et al.* developed a stereo magnification method based on a new layered representation of multiplane images. They collected a large set of videos of static scenes to train a deep neural network that is able to extrapolate views from images captured by a narrow-baseline stereo cameras [ZTF*18]. Compared to these methods, our work aims to generate novel views at a more distant viewpoint.

3. Novel View Synthesis by Appearance Flow Completion

This paper considers the problem of novel view synthesis from sparse and unstructured views. This is a challenging task as given only a sparse set of unstructured views, the baselines between input views are often large, which makes the overlapping regions between input views small, leading to a large portion of the scene that is only covered by one input view and thus difficult to perform 3D reconstruction. As shown in Figure 1, almost half of each of the input views is in the monocular region and 3D reconstruction cannot be performed there. Moreover, the large baseline between input views leads to potentially a significant amount of occlusion, which both makes 3D reconstruction and view synthesis difficult.

Instead of relying on the dense 3D reconstruction of the scene, our method explores its sparse 3D proxy to guide novel view synthesis. The state-of-the-art research on 3D vision can now provide robust sparse 3D reconstruction from only a few unstructured input views [SF16]. The sparse 3D proxy, in the form of a set of 3D scene points, defines where a sparse set of points in the target novel view are mapped to the input views. This can be naturally integrated into the appearance flow approach, which renders a novel view by estimating dense flows from the target view to the input views [ZTS*16]. The seminal appearance flow work by Zhou *et al.* can create a sharp novel view and inspires our work; however, it does not explicitly explore the geometry of the scene and cannot handle distant novel view synthesis. Our method builds upon it by directly using the sparse 3D proxy of the scene to guide appearance flow estimation. Specifically, we calculate sparse appearance flows according to the set of 3D scene points and accordingly formulate novel view synthesis as appearance flow completion that aims to complete the originally sparse appearance flows. Without loss of generality, we consider the case of two input images. We render each target view I_0 from two input views I_1 and I_2.

We develop a deep convolutional neural network to estimate the dense appearance flows from the sparse flow calculated from the sparse 3D proxy. As shown in Figure 2, given two input views I_1 and I_2, we first employ an off-the-shelf 3D reconstruction method [SZPF16] to estimate the camera parameters and a set of sparse 3D scene points. We calculate sparse appearance flows from the target view to each of the input views by projecting the

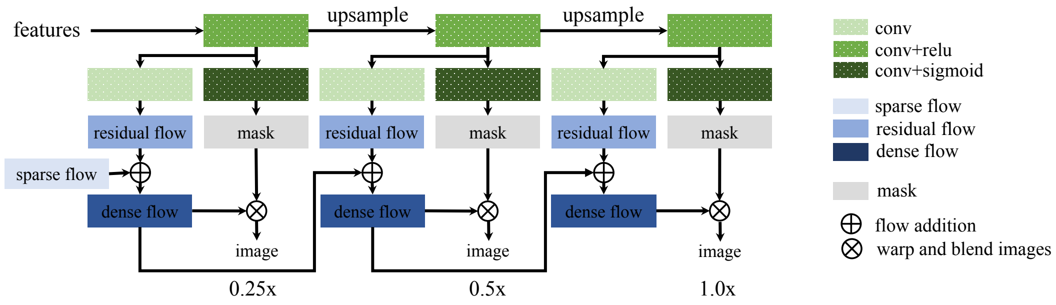


Figure 3: *The architecture of our multi-scale decoder.*

set of 3D scene points onto the target view. We then initialize the appearance flows for the undefined pixels in the target view by first estimating the best fitting homography between the target view and each input view from the sparse correspondences computed from the 3D scene points and then calculating the flows for undefined pixels using the estimated homography. Note, the consistency of initial flow fields from one target image to multiple different input images is respected by using the same set of scene points to estimate all homographies. In this way, we obtain the initialized appearance flows $F^s_{0,1}$ and $F^s_{0,2}$. The rest of this paper refers to the initialized appearance flows $F^s_{0,1}$ and $F^s_{0,2}$ as sparse flows to emphasize that only a sparse set of elements are accurate while most of them are approximated.

We stack the input views and the sparse appearance flows together and feed them into an encoder-decoder network with skip connections. This deep neural network outputs the appearance flows $F_{0,1}$ and $F_{0,2}$ and masks $\mathcal{M}_{0,1}$ and $\mathcal{M}_{0,2}$, which are used to synthesize the target view $\hat{I}_0$ as follows.

$$\hat{I}_0 = \mathcal{T}(I_1, F_{0,1}) \odot \mathcal{M}_{0,1} + \mathcal{T}(I_2, F_{0,2}) \odot \mathcal{M}_{0,2} \qquad (1)$$

where $\mathcal{T}$ back warps an image I guided by the appearance flows F.

As discussed in [ZTS*16], it is easier for a deep neural network to estimate short-range appearance flows to map an input view and the target. We share a similar observation. Therefore, our deep neural network estimates the residual flows $F^r_{0,1}$ and $F^r_{0,2}$ instead of the target appearance flows directly. These residual flows are added to the sparse appearance flows $F^s_{0,1}$ and $F^s_{0,2}$ to compute the target appearance flows $F_{0,1}$ and $F_{0,2}$, as shown in Figure 2.

3.1. Guided Appearance Flow Completion

We notice that often only a small number of 3D scene points can be estimated from input views. This makes it difficult for the neural network to estimate the dense appearance flows between the target and the input views. To address this problem, we have an observation that we can estimate both the sparse flows and dense flows between the two input views. This pair of sparse flows and dense flows can serve as a reference to guide learning to upgrade the sparse flows into the dense flows between the target view and the input views as all the views are imaged from the same scene. We use PWC-Net [SYLK18], a state-of-the-art optical flow algorithm, to estimate the dense appearance flows $F_{1,2}$ between two input views. We estimate the sparse flows between them in the same way as we estimate the sparse flows between the target and input views. Moreover, compared to existing methods [SHL*18,ZKSE16], such dense reference flows provide dense pixel correspondences between input views, making it easy for the neural network to learn to correctly warp and blend input views.

In theory, we can obtain dense 3D reconstruction from the dense flows between input views. In practice, we however find that the estimated dense flows are more of appearance flows than scene flows, which make them useful for view synthesis, but not sufficiently reliable for dense 3D reconstruction. This is consistent with the recent finding that optical flows need to be optimized for individual applications [XCW*19]. We therefore use the pair of sparse flows and dense flows between input views to guide the estimation of the target dense flows.

Figure 2 shows the full architecture of our fully convolutional neural network for novel view synthesis that aims to learn to complete the target appearance flows guided by the pair of sparse flows and dense flows between input views. In addition to the first encoder and the decoder described previously, it has a second encoder that takes as input the input views I_1 and I_2, the sparse flows $F^s_{1,2}$ and the dense flows $F_{1,2}$ between them. This encoder learns the correspondences between pixels in the two input views as well as the transformation from the sparse flows $F^s_{1,2}$ to the dense flows $F_{1,2}$. We concatenate the output features from the two encoders and use them as input for the decoder network.

To enable our neural network to estimate long-range flows, we furthermore adopt a multiple-scale iterative-refinement architecture, as illustrated in Figure 3. Specifically, our network outputs residual flows and corresponding masks to warp input images to the target at three different scales 0.25×, 0.5×, and 1.0×. At the scale 0.25×, our network first outputs the residual flows $F^{r,0.25}_{0,1}$ and $F^{r,0.25}_{0,2}$ to make correction for the initial sparse flows $F^{s,0.25}_{0,1}$ and $F^{s,0.25}_{0,2}$, and outputs predicted flows $F^{0.25}_{0,1}$ and $F^{0.25}_{0,2}$ at this scale. The flows at this scale aim to capture high-level structure of the images. We then use bilinear upsampling to convert these small-scale flows $F^{0.25}_{0,1}$ and $F^{0.25}_{0,2}$ to the next larger scales and use them as the initial flows $F^{s,0.5}_{0,1}$ and $F^{s,0.5}_{0,2}$ for the next refinement step. We repeat the same step for the last scale. This design is inspired by the prior work of Zhou *et al.* [ZTS*16], in which the authors discuss that using a multiple-scale architecture encourages the network to estimate long appearance flows better.

Table 1: *Qualitative comparison between our method and Appearance Flow [ZTS*16] and Multi-view to Novel. View [SHL*18].*

	MAD ↓	PSNR ↑	SSIM ↑	LPIPS ↓
Multi2Novel	0.1476	18.4938	0.6868	0.2154
App. Flow	0.2517	15.0071	0.5963	0.301
Ours	**0.1432**	**19.7985**	**0.7312**	**0.194**

Table 2: *Ablation study results.*

	MAD ↓	PSNR ↑	SSIM ↑	LPIPS ↓
w/o reference flow	0.1523	18.8214	0.7157	0.2001
w/o multi-scale	0.1534	19.0566	0.7184	0.2076
w/o perceptual loss	0.1542	18.6469	0.7128	0.2166
Ours	**0.1432**	**19.7985**	**0.7312**	**0.1940**

3.1.1. Loss functions

We consider two different loss functions to train our novel view synthesis network: the color loss and the feature loss. The color loss $\mathcal{L}_c$ measures the pixel-wise distance between the synthesized target image $\hat{I}_0$ and the ground truth I_0 as follows.

$$\mathcal{L}_c = \frac{1}{n}\|\hat{I}_0 - I_0\|_1, \tag{2}$$

where n is the number of pixels in I_0. The feature loss $\mathcal{L}_f$ focuses on the perceptual difference between the synthesized and the ground truth target image. As shown in many other image synthesis tasks [CK17, DB16, JAF16, LTH*16, ZKSE16, NML17b], perceptual loss helps deep neural networks to generate visually appealing images. We follow these existing methods and compute the feature loss using the feature maps from the last three layers of the VGG network [SZ14] pre-trained on the ImageNet dataset [DDS*09].

$$\mathcal{L}_f = \frac{1}{3}\sum_{l=1}^{3}\|\phi_l(\hat{I}_0) - \phi_l(I_0)\|_1, \tag{3}$$

where $\phi_l(I)$ is the feature map from the last l layers of the VGG network. We combine these two losses to train our neural network.

$$\mathcal{L} = \mathcal{L}_c + \alpha\mathcal{L}_f, \tag{4}$$

where α is a constant with value 0.001.

We finally measure both color and perceptual loss at three different scales and combine them into a single loss to train our network.

$$\mathcal{L} = \sigma_1\mathcal{L}_{1.0} + \sigma_2\mathcal{L}_{0.5} + \sigma_3\mathcal{L}_{0.25} \tag{5}$$

where $\mathcal{L}_{1.0}$, $\mathcal{L}_{0.50}$, $\mathcal{L}_{0.25}$ are the losses measured at three different scales. Empirically, we set $\sigma_1 = 0.25$, $\sigma_2 = 0.5$, and $\sigma_3 = 1$.

4. Experiments

We experiment with our method on the KITTI dataset [GLU12], which often serves as a benchmark for real-scene novel view synthesis. We follow the same procedure as the recent work [SHL*18, ZTS*16] to sample input frames and the corresponding target frames. Specifically, we use the 11 frame sequences that are provided with ground truth camera poses. In each sequence, the source and target frames are randomly sampled so that they are separated by at most 10 frames. We use the same training and testing split provided by [SHL*18] to train and evaluate our method. Please refer to [SHL*18, ZTS*16] for more details on how to sample the training and testing tuples. We implemented our appearance flow completion network using TensorFlow. We trained our network with using the Adam optimizer [KB14] with an initial learning rate of 7×10^{-5}. The network was trained for one million iterations for roughly 3 days on an NVIDIA GTX 1080 Ti GPU.

We compare our method with two state-of-the-art deep learning-based novel view synthesis methods Appearance Flow [ZTS*16] and Multi-view to Novel View (Multi2Novel) [SHL*18] both quantitatively and qualitatively. We also perform ablation study to further understand our method. Please refer to our supplementary video demo to further examine the visual quality of our results.

We qualitatively measure the quality of novel view synthesis results using four metrics, including Mean Absolute Difference (MAD), Peak Signal to Noise (PSNR), Structural Similarity (SSIM) [WBSS04], and LPIPS, which measures the perceptual difference between a synthesized image and the ground truth [ZIE*18]. For MAD and LPIPS, a small value indicates a better quality while for SSIM and PSNR, a large value indicates a better quality. As reported in Table 1, our method achieves better numerical scores than Appearance Flow and Multi2Novel on the well-known KITTI benchmark [GLU12] according to all the four metrics. Figure 4 shows some visual comparisons. These examples show that our method is able to generate novel views with less ghosting and blurry artifacts (Row 1, 4 and 5), and less distortion (Row 2 and 3).

Because the distance between viewpoints has a strong effect on the quality of a synthesized novel view, we evaluate the novel view synthesis methods with different distances between the target and input viewpoints. We approximate the viewpoint distance as the time difference between the video frames in the KITTI benchmark because each of these videos was captured using the same moving camera [GLU12]. For every tuple of the target and two input frames, we use the smaller distance between the target and one of the input frames. Figure 5 shows how each method performs with different viewpoint distances. For each distance in the horizontal axis, we compute the average score of all the samples that have their frame distances smaller than that particular distance. Indeed, the performance of all methods decreases when the distance between the target and input views increases. Nevertheless, our method is more robust than the competing methods.

4.1. Ablation study

To further assess our method, we conduct an ablation study by training separated models while leaving a major component out. As reported in Table 2, leaving any component out compromises the performance of our method. As shown in Figure 6, the perceptual loss helps producing sharper images and reducing visual distortion and ghosting artifacts. Our multi-scale architecture helps achieve better blending input images captured at distant viewpoints, which is consistent with the observation from [ZTS*16]. Moreover, Figure 7 shows that the pair of sparse flows and dense flows between input views can serve as a reference to guide our network

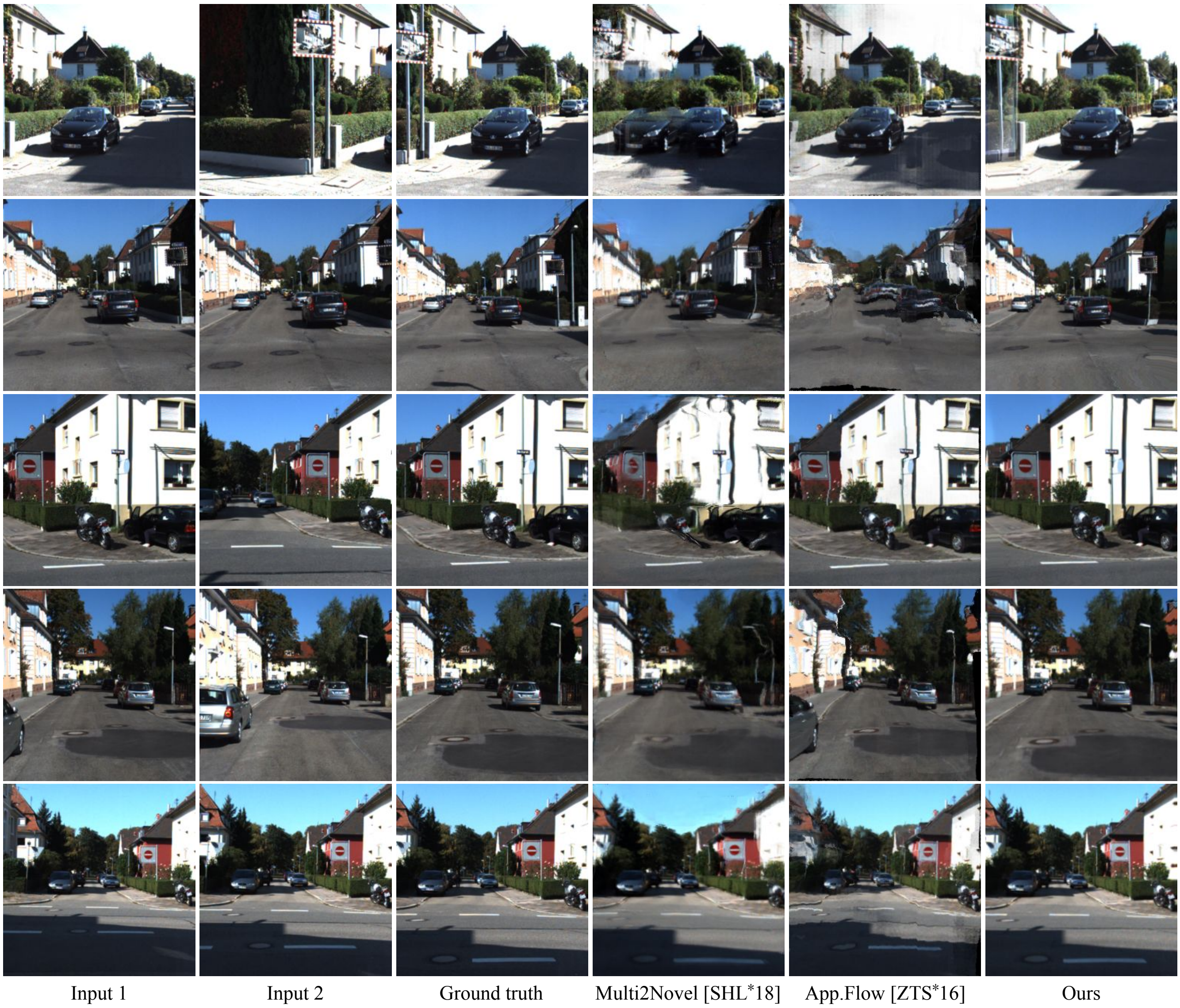


Figure 4: *Visual comparison between our method and the state-of-the-art methods.*

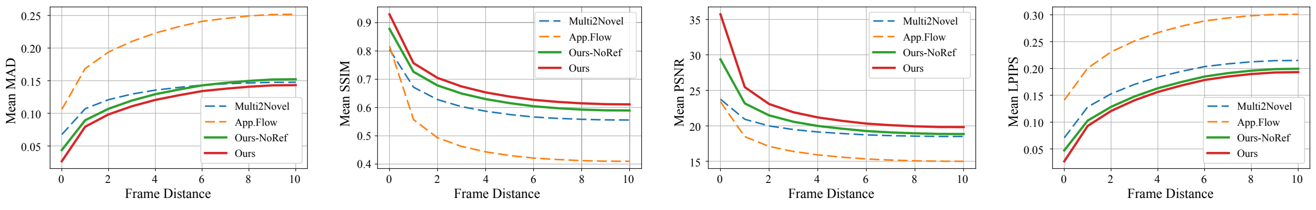


Figure 5: *As the viewpoint distance increases, the performance of each novel view synthesis method decreases. Nevertheless, our method is robust against the increasing viewpoint distance, compared to the other methods.*

to estimate dense appearance flows to synthesize the target view. Note, existing appearance flow-based synthesis approaches independently predict the flows to map pixels from each input view to the target and then combine the resulting warped images using predicted masks [ZTS*16, SHL*18]. This task is challenging since the network needs to estimate the dense flows from the target view to different input views consistently so that the warped images can be blended without blurring or ghosting artifacts. We address this problem by leveraging the power of a state-of-the-art optical flow algorithm to directly estimate the pixel correspondences

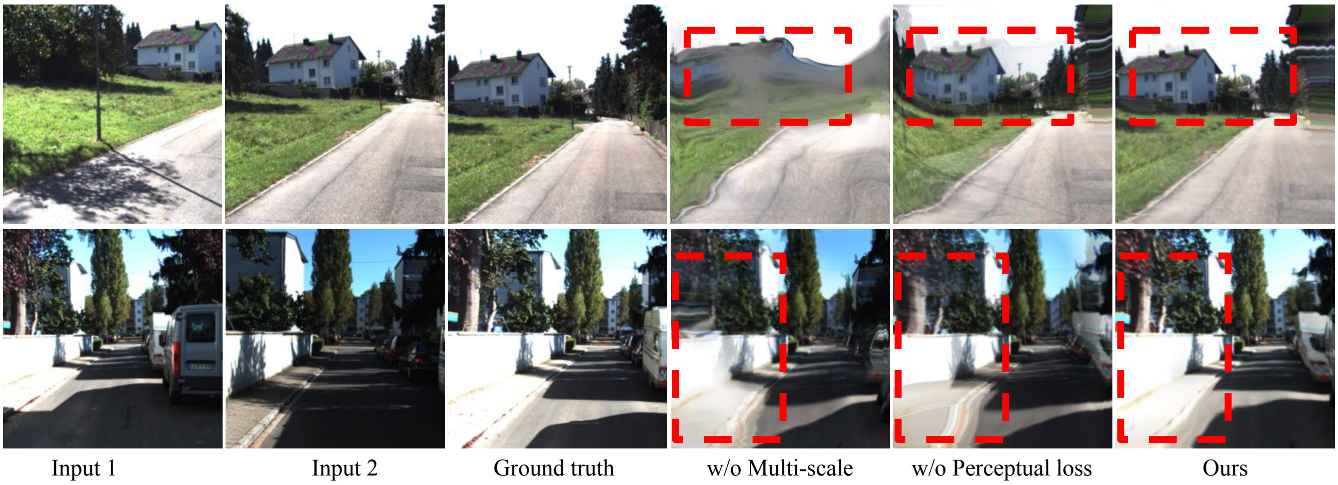


Figure 6: *Effects of our multi-scale architecture and perceptual loss function on the visual quality of novel view synthesis results.*

(dense flows) between the input images, and use them to guide the estimation of target dense flows as well as masks. As shown in Figure 7, our method with the reference flows is able to produce high quality novel views free from blurring or ghosting artifacts.

4.2. Applications

Below we discuss how our algorithm can be used for two novel view synthesis applications. All of our results are generated from only two input images, including the video results. That is, given an input video of n frames, at each time, we only take two consecutive frames as input and generate frames between them. Once we finish processing all the $n-1$ frame pairs, we assemble all the frames together into the final video.

4.2.1. Video frame interpolation

Video frame interpolation can be considered as a special case of novel view synthesis where one or more intermediate frames are interpolated between every two consecutive input frames. Great progress has been made for this problem [NML17a, NML17b, NL18, LLLC19, BLM*19]. We compare our method with two recent video frame interpolation methods, namely Cyclic Frame Generation (CyclicGen) [LLLC19] and Depth-aware Frame Interpolation (DAIN) [BLM*19]. CyclicGen is purely appearance based and DAIN explores the understanding of scene depth for video frame interpolation. In this experiment, we sample input frames from the KITTI benchmark that are 0.4 seconds away from each other and then use each algorithm to generate intermediate frames. As shown in Figure 8, our method can often produce more realistic results with less visual artifacts, especially for the cases that the input images were captured at distant viewpoints.

4.2.2. Free viewpoint navigation

Novel view synthesis is a critical technology for free viewpoint navigation [CDSHD13]. We compare our method with the recent Multi2Novel method for this task [SHL*18]. For a fair comparison, we conduct this experiment on the KITTI dataset since we are using the trained model of Multi2Novel on the KITTI dataset directly. In this experiment, we linearly interpolate the camera locations between any two input views. We use the Slerp algorithm [Sho85] to interpolate the camera orientations. We then use a novel view synthesis algorithm to render a view at each new viewpoint.

We evaluate novel view synthesis algorithms in two cases: (i) intermediate locations between any two input views (interpolation), and (ii) locations before and after input views (extrapolation). Figure 9 shows visual examples of the smooth transiting results generated by our method in comparison with Multi2Novel. These results show that our method can successfully interpolate and extrapolate novel frames from only two input frames. The transition between consecutive synthesized frames is smooth enough to enable free-viewpoint navigation, as shown in the supplementary video demo.

4.3. Discussion

Many image-based rendering (IBR) algorithms rely on Multi-View Stereo (MVS) algorithms to perform dense 3D reconstruction of a scene. MVS often works well for a sequence of 10 frames or more, and thus these IBR methods often work well with many input views. However, when the overlap among these frames is small, MVS methods sometimes cannot work well. For example, in the KITTI dataset, when we sample multiple frames from a video captured by a camera on a forward-moving car, MVS cannot generate satisfactory results as many parts of the scene is only captured in one or at most two views, which makes dense 3D reconstruction difficult and leads to unsatisfactory novel view synthesis results. In contrast, our method addresses such a challenging problem by estimating appearance flows from the target view to the input views and employing a neural network to blend them.

While this paper focuses on novel view synthesis from only two input views, our method can be extended to handle more input images. Specifically, given multiple input images, we can apply the same off-the-shelf method as described in our paper to first estimate robust sparse flows between the target view and each of the multiple input images. We feed these sparse flows and input images into our view synthesis network. We then train our network to learn to transform these sparse flows into dense flows between the target and the input images. Our method uses these flows to warp input images to the target and combine multiple warped images using corresponding mask maps to render the target view.

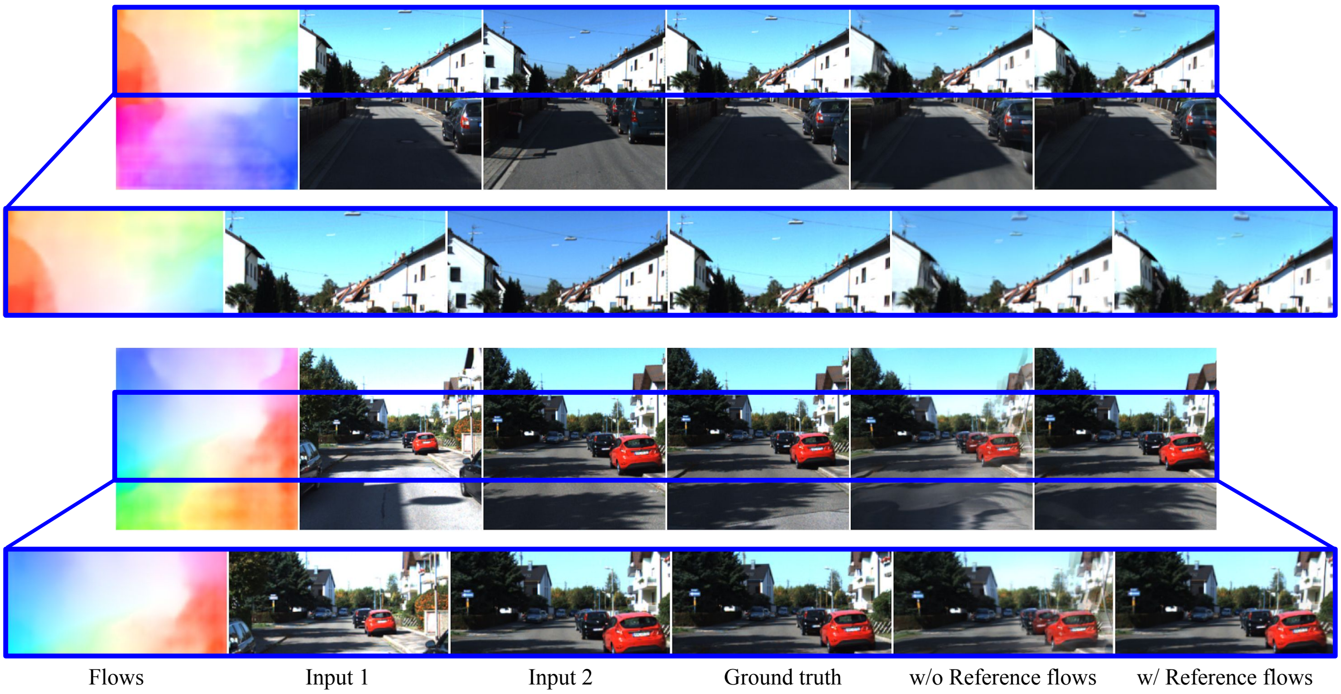


Figure 7: *Effect of reference flows. By directly estimating dense flows and using them to guide the estimation of the target dense appearance flows and masks, our results avoid ghosting or blurring artifacts.*

We also tested our method on the challenging IBR dataset from Chaurasia *et al.* [CDSHD13]. We sampled pairs of frames that are four frames apart from each other. Given such challenging examples, our method is able to synthesize realistic views in general, as shown in Figure 10(d) and (f). We do notice that our results sometimes contain noticeable ghosting artifacts in the regions with very strong parallax (Figure 10(e)). Such artifacts can be mostly attributed to two factors. First, no sparse flows in these regions are available, such as the tall tree and the bench chair in Figure 10(a) and (c). Second, the reference flows are misleading, such as those for the tall tree when we use an off-the-shelf optical flow algorithm to estimate them due to large motion there (as shown in Figure 10(b)). Note, some state-of-the-art novel view synthesis methods, such as [PZ17], can better handle this challenging example than ours, as more input frames are used for 3D reconstruction in their methods. In contrast, our method only uses two input frames and thus tries to address a more challenging scenario where only a very sparse set of input views are available. We conducted such experiments to thoroughly examine the performance of our method. As shown in this experiment, the performance of our method can be compromised by the insufficient sparse flows and reference flows.

In our experiments, we pre-process input images to obtain sparse 3d reconstruction and optical flow using off-the-shelf computer vision algorithms. At the rendering stage, our network takes 0.06, 0.13, 0.21, and 0.64 seconds to generate a novel view of size 512×512, 800×800, 1024×1024, and 1764×1764 respectively on a workstation with an Intel I7-7700 CPU and an NVIDIA 1080Ti GPU. We believe that this performance can be further improved to enable real-time image-based rendering in the future with better code optimization.

5. Conclusion

This paper presented a method that is able to generate a novel view from sparse and unstructured input ones. This method considers novel view synthesis as a dense appearance flow estimation problem and explores reliable sparse 3D geometry and deep neural network to estimate the dense appearance flow from the target view to the input views. Specifically, this method calculates the sparse flow for a sparse set of 3D scene points and trains a deep neural network to transform the sparse flow to the dense appearance flows. This method also explores the readily available pair of sparse flows and the corresponding dense flows between input views to guide the estimation of the unknown dense appearance flows. The experiments show that this paper can generate state-of-the-art novel views at distant viewpoints.

Acknowledgment.
This project is supported by a gift award from Intel.

References

[BBM*01] Buehler C., Bosse M., McMillan L., Gortler S., Cohen M.: Unstructured lumigraph rendering. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), pp. 425–432. 1, 2

[BLM*19] Bao W., Lai W.-S., Ma C., Zhang X., Gao Z., Yang M.-H.: Depth-aware video frame interpolation. In *IEEE Conferene on Computer Vision and Pattern Recognition* (2019). 3, 7, 9

[CDSHD13] Chaurasia G., Duchene S., Sorkine-Hornung O., Drettakis G.: Depth synthesis and local warps for plausible image-based navigation. *ACM Trans. Graph. 32*, 3 (2013). 1, 2, 7, 8, 11

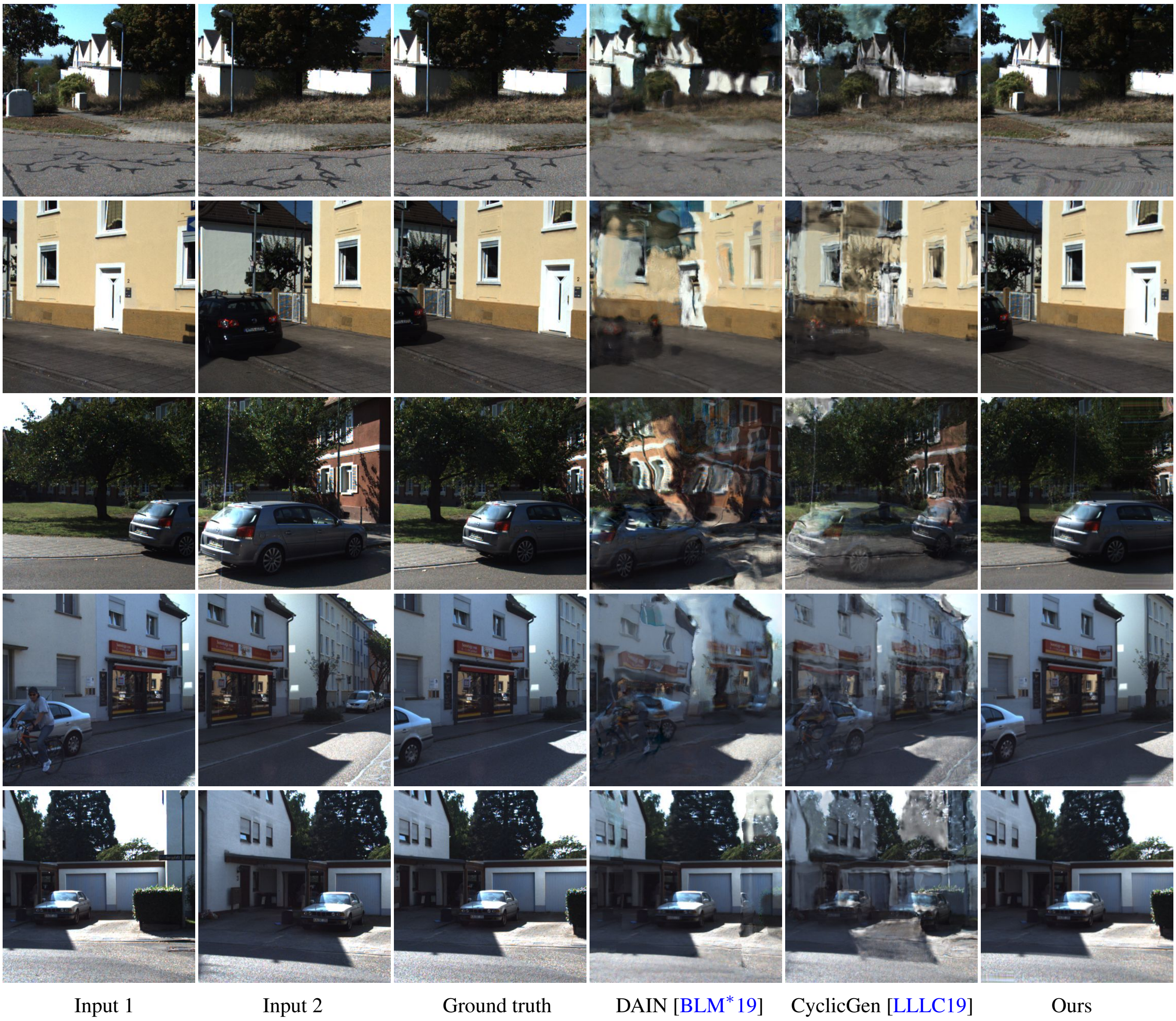


Figure 8: *Comparisons with video frame interpolation methods.*

[CK17] Chen Q., Koltun V.: Photographic image synthesis with cascaded refinement networks. In *IEEE International Conference on Computer Vision* (Oct 2017). 5

[DB16] Dosovitskiy A., Brox T.: Generating images with perceptual similarity metrics based on deep networks. In *Advances in Neural Information Processing Systems* (2016), pp. 658–666. 5

[DDS*09] Deng J., Dong W., Socher R., Li L.-J., Li K., Fei-Fei L.: Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition* (2009), pp. 248–255. 5

[DSB15] Dosovitskiy A., Springenberg J. T., Brox T.: Learning to generate chairs with convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1538–1546. 2

[FNPS16] Flynn J., Neulander I., Philbin J., Snavely N.: Deepstereo: Learning to predict new views from the world's imagery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 5515–5524. 2

[FWZ05] Fitzgibbon A., Wexler Y., Zisserman A.: Image-based rendering using image-based priors. *International Journal of Computer Vision 63*, 2 (2005), 141–151. 2

[GLU12] Geiger A., Lenz P., Urtasun R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition* (2012), pp. 3354–3361. 2, 5

[HPP*18] Hedman P., Philip J., Price T., Frahm J.-M., Drettakis G., Brostow G.: Deep blending for free-viewpoint image-based rendering. *ACM Trans. Graph. 37*, 6 (2018). 2

[JAF16] Johnson J., Alahi A., Fei-Fei L.: Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision* (2016), vol. 9906, pp. 694–711. 5

[JSJ*18] Jiang H., Sun D., Jampani V., Yang M.-H., Learned-Miller E., Kautz J.: Super SloMo: High quality estimation of multi-

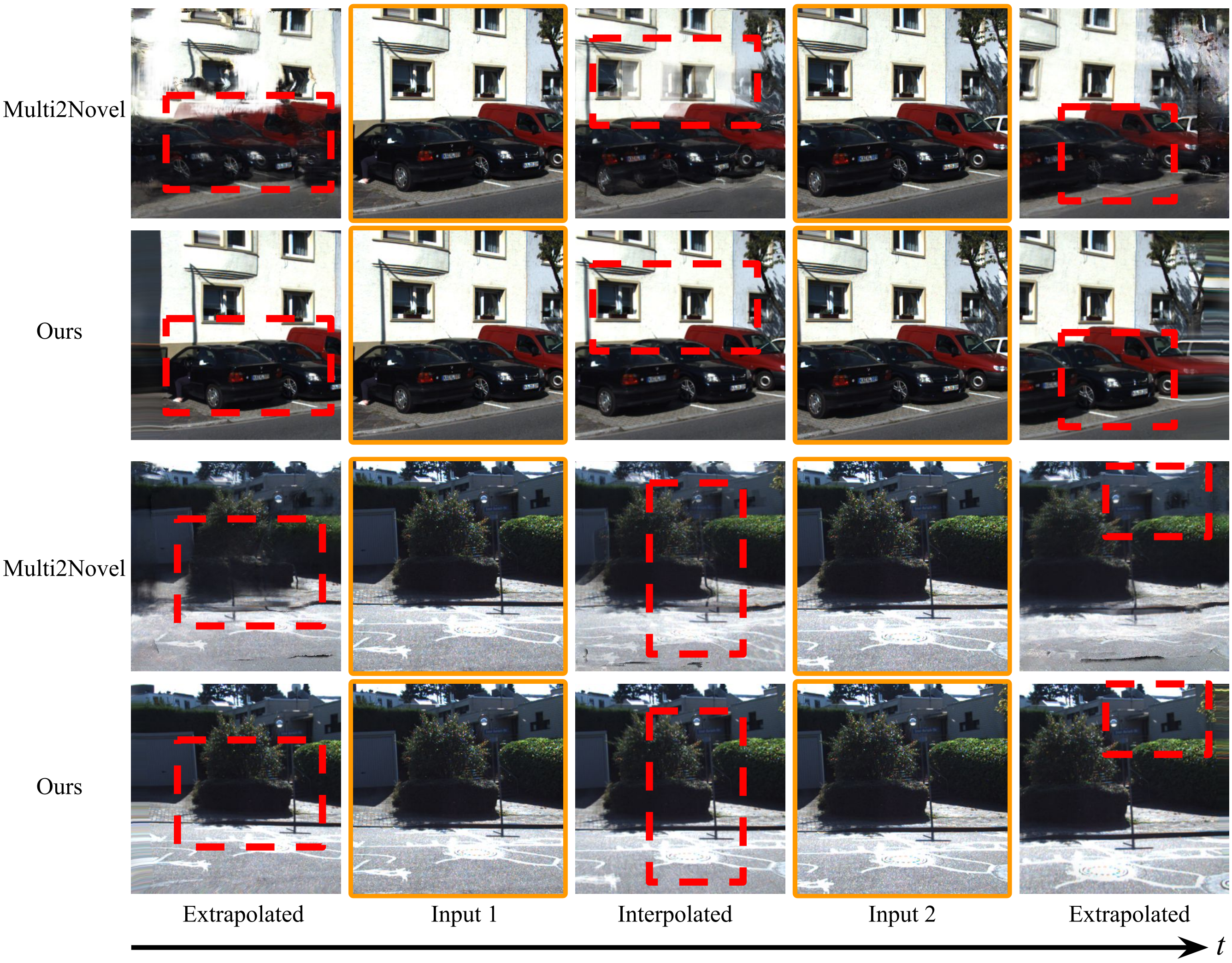


Figure 9: *Free viewpoint navigation enabled by novel view synthesis. interpolation and extrapolation. Our method is able to both interpolate and extrapolate new frames from only two input frames. Our results are typically free from blurring, ghosting artifacts or distortion.*

ple intermediate frames for video interpolation. In *IEEE Conference on Computer Vision and Pattern Recognition* (June 2018). 3

[KB14] Kingma D. P., Ba J.: Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014). 5

[KLTS06] Kang S. B., Li Y., Tong X., Shum H.: Image-based rendering. *Foundations and Trends in Computer Graphics and Vision 2*, 3 (2006). 1, 2

[KWKT15] Kulkarni T. D., Whitney W. F., Kohli P., Tenenbaum J. B.: Deep convolutional inverse graphics network. In *NIPS* (2015), pp. 2539–2547. 2

[KWR16] Kalantari N. K., Wang T., Ramamoorthi R.: Learning-based view synthesis for light field cameras. *ACM Trans. Graph. 35*, 6 (2016), 193:1–193:10. 1, 3

[LGJA09] Liu F., Gleicher M., Jin H., Agarwala A.: Content-preserving warps for 3d video stabilization. *ACM Trans. Graph 28*, 3 (2009), 44. 1

[LH96] Levoy M., Hanrahan P.: Light field rendering. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques* (1996), pp. 31–42. 1, 2

[LHS18] Liu M., He X., Salzmann M.: Geometry-aware deep network for single-image novel view synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 4616–4624. 2

[LLLC19] Liu Y.-L., Liao Y.-T., Lin Y.-Y., Chuang Y.-Y.: Deep video frame interpolation using cyclic frame generation. 3, 7, 9

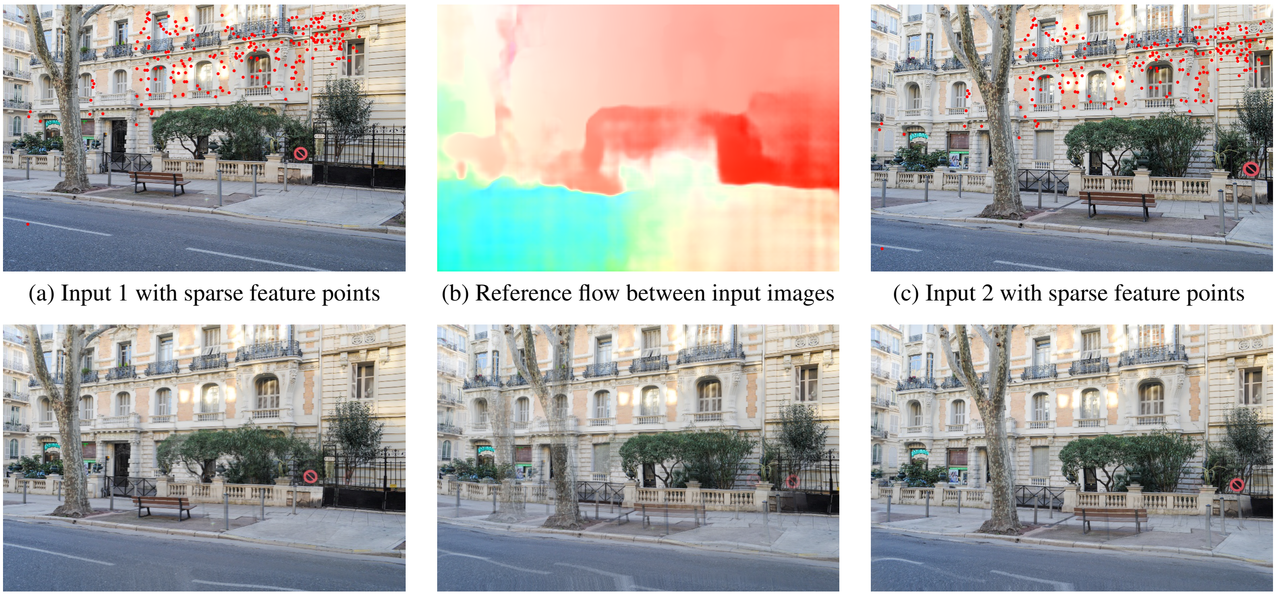
[LTH*16] Ledig C., Theis L., Huszar F., Caballero J., Aitken A. P., Tejani A., Totz J., Wang Z., Shi W.: Photo-realistic single image super-resolution using a generative adversarial network. *arXiv/1609.04802* (2016). 5

[LYT*17] Liu Z., Yeh R. A., Tang X., Liu Y., Agarwala A.: Video frame synthesis using deep voxel flow. In *IEEE International Conference on Computer Vision* (Oct 2017). 3

[MDM*18] Meyer S., Djelouah A., McWilliams B., Sorkine-Hornung A., Gross M., Schroers C.: PhaseNet for video frame interpolation. In *IEEE Conference on Computer Vision and Pattern Recognition* (June 2018). 3

[NL18] Niklaus S., Liu F.: Context-aware synthesis for video frame interpolation. In *IEEE Conference on Computer Vision and Pattern Recognition* (June 2018). 3, 7

[NLB*] Ng R., Levoy M., Brédif M., Duval G., Horowitz M.,



(a) Input 1 with sparse feature points

(b) Reference flow between input images

(c) Input 2 with sparse feature points

(d) Our result

(e) Our result with noticeable artifacts

(f) Our result

Figure 10: *Our results on the image-based rendering dataset from Chaurasia* et al. *[CDSHD13]. The second row shows the novel view synthesis results at different viewpoints.*

Hanrahan P.: Light field photography with a hand-held plenoptic camera. In *Stanford Computer Science Technical Report CSTR 2, no. 11 (2005): 1-11*. 1, 2

[NML17a] Niklaus S., Mai L., Liu F.: Video frame interpolation via adaptive convolution. In *IEEE Conference on Computer Vision and Pattern Recognition* (July 2017). 7

[NML17b] Niklaus S., Mai L., Liu F.: Video frame interpolation via adaptive separable convolution. In *IEEE International Conference on Computer Vision* (Oct 2017). 5, 7

[PZ17] Penner E., Zhang L.: Soft 3d reconstruction for view synthesis. *ACM Transactions on Graphics (TOG) 36*, 6 (2017), 235. 2, 8

[SF16] Schonberger J. L., Frahm J.-M.: Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 4104–4113. 1, 3

[SHL*18] Sun S.-H., Huh M., Liao Y.-H., Zhang N., Lim J. J.: Multi-view to novel view: Synthesizing novel views with self-learned confidence. In *European Conference on Computer Vision* (2018), pp. 155–171. 1, 3, 4, 5, 6, 7

[Sho85] Shoemake K.: Animating rotation with quaternion curves. In *ACM SIGGRAPH computer graphics* (1985), vol. 19, ACM, pp. 245–254. 7

[STH*19] Sitzmann V., Thies J., Heide F., Niessner M., Wetzstein G., Zollhofer M.: Deepvoxels: Learning persistent 3d feature embeddings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2019), pp. 2437–2446. 2

[SYLK18] Sun D., Yang X., Liu M.-Y., Kautz J.: Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 8934–8943. 4

[SZ14] Simonyan K., Zisserman A.: Very deep convolutional networks for large-scale image recognition. *arXiv/1409.1556* (2014). 5

[Sze10] Szeliski R.: *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010. 1, 2

[SZPF16] Schönberger J. L., Zheng E., Pollefeys M., Frahm J.-M.: Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)* (2016). 2, 3

[TDB16] Tatarchenko M., Dosovitskiy A., Brox T.: Multi-view 3D models from single images with a convolutional network. In *European Conference on Computer Vision* (2016), pp. 322–337. 2

[TZN19] Thies J., Zollhöfer M., NieM.: Deferred neural rendering: Image synthesis using neural textures. *ACM Trans. Graph. 38*, 4 (2019), 66:1–66:12. 2

[TZT*18] Thies J., Zollhöfer M., Theobalt C., Stamminger M., Niessner M.: Ignor: Image-guided neural object rendering. *arXiv:1811.10720* (2018). 2

[WBSS04] Wang Z., Bovik A. C., Sheikh H. R., Simoncelli E. P.: Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing 13*, 4 (2004), 600–612. 5

[XCW*19] Xue T., Chen B., Wu J., Wei D., Freeman W. T.: Video enhancement with task-oriented flow. *International Journal of Computer Vision* (2019). 4

[YRYL15] Yang J., Reed S. E., Yang M., Lee H.: Weakly-supervised disentangling with recurrent transformations for 3D view synthesis. In *NIPS* (2015), pp. 1099–1107. 2

[ZC04] Zhang C., Chen T.: A survey on image-based rendering - representation, sampling and compression. *Signal Processing: Image Communication 19*, 1 (2004), 1–28. 1, 2

[ZIE*18] Zhang R., Isola P., Efros A. A., Shechtman E., Wang O.: The unreasonable effectiveness of deep features as a perceptual metric. *arXiv preprint* (2018). 5

[ZKSE16] Zhu J., Krähenbühl P., Shechtman E., Efros A. A.: Generative visual manipulation on the natural image manifold. In *European Conference on Computer Vision* (2016), pp. 597–613. 1, 4, 5

[ZKU*04] Zitnick C. L., Kang S. B., Uyttendaele M., Winder S., Szeliski R.: High-quality video view interpolation using a layered representation. *ACM Trans. Graph. 23*, 3 (2004), 600–608. 1, 2

[ZTF*18] Zhou T., Tucker R., Flynn J., Fyffe G., Snavely N.: Stereo magnification: learning view synthesis using multiplane images. *ACM Transactions on Graphics 37*, 4 (2018), 65. 3

[ZTS*16] Zhou T., Tulsiani S., Sun W., Malik J., Efros A. A.: View synthesis by appearance flow. In *European conference on computer vision* (2016), Springer, pp. 286–301. 1, 3, 4, 5, 6