NVSS: High-quality Novel View Selfie Synthesis

Jia-Wang Bian*, Huangying Zhan*, Ian Reid
University of Adelaide
{jiawang.bian, huangying.zhan, ian.reid}@adelaide.edu.au

Abstract

We present a novel method to synthesize novel view selfies from a mobile phone captured video. This is challenging due to the inconsistent geometry that is caused by the person's unavoidable movement. Recent methods reconstruct the whole deformable scene implicitly with a deformation field. We argue that they are inefficient and hard to fit diverse real-world videos. In contrast, we use an explicit reconstruction for generalization and efficiency, where we separately track, reconstruct, and synthesize the foreground and background to overcome the geometry inconsistency. Several novel and effective modules are proposed for better performance and visual results. We demonstrate the advantage of the proposed method against the existing alternatives in a collection of our captured selfie videos with the support of quantitative and qualitative results.

1. Introduction

Novel view synthesis (NVS) is one of the classic tasks in computer vision. The goal is to generate new viewpoint images of the scene, given a set of real images or video of the scene acquired from known viewpoints. NVS has a plenty of practical applications. For example, it enables photo/video editing software to manipulate objects in 3D space and helps create virtual reality environment. Previous works [29, 35, 36] usually assume that the scene is static, while it is often not true in real world scenarios. Geometric methods usually consider a highly constrained setup that multiple synchronized cameras capture videos at a same time. Dynamic scenes can then be considered as static given a collection of images taken at different viewpoints at the same timestamp under this setup. However, synthesizing a novel view image with a video footage captured by a single camera is a more challenging, general, and under-explored problem. Our work falls into this area and we approach it using a combination of multi-view geometry and deep learning based methods.

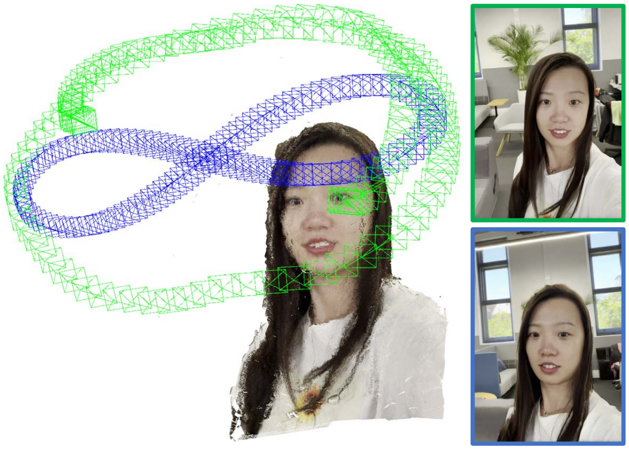

Figure 1: Novel view selfie synthesis. Our method takes a selfie video as input and renders novel images from new viewpoints. Green boxes stand for the camera path of original video, and blue boxes stand for the camera path of our rendered new video.

Modeling people with hand-held cameras is challenging due to the non-rigidity, *e.g.*, we cannot keep perfectly still when capturing selfie videos. This causes the scene geometry to be inconsistent across multiple views, making the static rendering methods [29, 35, 36] unfeasible and usually results in artifacts due to inaccuracies created by the scene dynamics. The bulk of this geometric inconsistency occurs between the static background (which *is* usually rigid) and the dynamic foreground, which for selfie videos is usually *mostly* rigid, but not the same rigid motion as the background. In this paper, we are interested in this special case of dynamic novel view synthesis.

Nerfies [31], a recent method that is based on implicit representation and volumetric rendering, uses two multilayer perceptron (MLP) networks to learn a deformation field for each image and a canonical template for all the observations. Although high-quality results are reported, the method is inefficient, *e.g.*, it requires multiple GPUs for training models on per video basis, and the training time is up to several days. Besides, in our own experiments we

[*] Indicates equal contributions and joint first authors.

have found that it is hard to learn the correct deformation, *e.g.*, see Fig. 5 which shows some of our data and note that the human pose looks incorrect in column 3. We conjecture that learning accurate deformation is a highly unconstrained problem, which requires exhaustive training and sometimes may converge to a local minimum.

Inspired by the ways that traditional geometric methods approach tracking and mapping in dynamic scenes [2, 43], which separately track (and reconstruct) static scene and dynamic objects, we propose to *track, reconstruct, and synthesize* the dynamic and static parts separately, avoiding their mutual inconsistency, instead of modeling the entire scene as deformable (as in [31]). The separate synthesized foreground/background images are fused eventually with a carefully designed fusion scheme. With the rapid development of deep learning in recent years, state-of-the-art semantic segmentation networks enable accurate object segmentation. In our case, dynamic foreground object (human) and static background (non-human) can be easily segmented with the aid of deep segmentation network [7].

The idea of separate reconstruction does not rely on specific representations, *e.g.*, implicit or explicit methods, and we adopt the latter for better generalization. We track and reconstruct the foreground/background using classic structure-from-motion (SfM) and multi-view stereo (MVS) systems [38, 39]. By excluding the foreground in the background reconstruction, our method is able to render a full background image. This enables a free combination with different foreground images, and it also can fill the occluded background in the original video. We explore a variety of ways to fuse the synthesized foreground/background images and find that a deep matting method [26] seamlessly fuse the synthesized images to a photo-realistic image as shown in Fig. 4 and Fig. 5.

In this work, our contributions include (a) we propose a general and efficient framework for synthesizing dynamic scenes from novel viewpoints that first separately track, reconstruct, and synthesize static/dynamic regions, then fuse the synthesized images into a photo-realistic image; (b) we propose full background synthesis which allows for free combination with foreground images, and it also can complete the occluded background in the original video; (c) we propose a mask estimation method which aggregates source view masks, and the estimated mask is refined by a matting method to seamlessly fuse the rendered foreground and background images together.

2. Related Work

Image-based rendering has a long history in computer vision and graphics. Shum and Kang [40] provide a review of early methods, and seminal works after that include [6, 11, 13, 17, 19, 33]. These methods are mainly based on multi-view geometry. Recently, deep learning-based methods [1, 9, 10, 16, 35, 36, 41, 42] are proposed, which also rely on geometric reconstruction. Our method is most related to SVS [36], which operates on the geometric scaffold that is reconstructed using the classic structure-from-motion [38] and multi-view stereo [38]. It aggregates source view features on the scaffold surface to render the target view, showing high-quality results in static scenes. We follow this idea and contribute by extending the static scene reconstruction to dynamic selfie videos.

Implicit 3D representations [8, 28, 30] are recently popular in computer vision, and NeRF [29] shows that it can be used with the volumetric rendering for synthesizing photo-realistic images. A series of following works [12, 14, 21, 22, 31, 32, 34, 44] extend NeRF to dynamic scenes, in which Nerfies [31] is the most related to ours. It implicitly reconstructs the entire deformable scene with a template geometry for all observations and a deformation field for each image. Although excellent results are reported, the method is inefficient because it requires training on per test video, due to the nature of implicit representation. Compared with these methods, (a) our method generalizes well to different scenes and hence is more efficient; (b) these methods are able to represent other non-rigid scenes, while our method is only applicable to specific scenes such as selfies due to the human-centric reconstruction and synthesis; (c) our idea of separate reconstruction can also be used to learn the implicit representation, which reduces the geometry inconsistency significantly and simplifies the learning.

The domain-specific knowledge can be used to reconstruct faces [3, 4, 5] and human bodies [24, 46]. However, these methods often lack details such as hair and eyeglasses. Compared with them, our method doesn't rely on domain-specific knowledge, and it is able to render details.

3. Method

High-quality 3D human scanning and rendering from the scan are challenging as it is difficult for a human to keep perfectly still while taking a selfie video. In contrast, scanning and rendering a static scene is relatively easy. In this work, we propose to disentangle the foreground (*i.e.* human) from the background scene, and we assume the human mainly undergoes a small rigid motion. As a result, both parts can be considered as static w.r.t. the camera, and hence we can reconstruct them individually using rigid methods. A full background rendering method from geometric reconstruction is presented in Sec. 3.1. In Sec. 3.2, we present a method to render the high-quality human images with a carefully designed masking scheme and a random sampling-based inlier frame detection method. Finally, we discuss the seamless fusion of rendered results in Sec. 3.3 to form a photo-realistic novel view image.

3.1. Full Background Synthesis

In this section, we propose full background rendering from a collection of N selfie photos. This allows seamless fusion with any rendered foreground in our later stage. However, it is non-trivial because the background in source views is always partially occluded. First, we reconstruct a 3D background model based on multi-view geometry methods, in which we remove foreground regions. We then map the novel view pixels to source views via projective geometry. Color features on valid background regions are extracted from the source view images and aggregated together. Finally, a neural renderer is adopted to convert the aggregated features to a target color image.

Tracking and Reconstructing Background. First, we segment images into foreground and background regions. After experimenting with different semantic segmentation networks, we choose the DeepLabV3 [7] model that is pretrained on COCO dataset [23]. "Person" segments are considered as foreground while the remaining regions are considered as background. Second, we use the COLMAP [38] system for estimating camera intrinsic $\boldsymbol{K}_b$ and extrinsics $\{\boldsymbol{P}_{b,i}\}_{i=1}^N$. Towards accurate camera poses estimation for background reconstruction, we remove the DoG keypoints [25] detected in foreground regions before running feature matching and bundle adjustment. Third, with the estimated camera poses, we conduct the dense reconstruction using the classic MVS system [39], which estimates a semi-dense depth map, $D_{b,i}$, for each image. Finally, we fuse all depth maps $\{D_{b,i}\}_{i=1}^N$ into a point cloud and fit a surface mesh $\mathcal{M}_b$ [20] from the point cloud. This removes outliers significantly and densely represents the 3D geometry of the background scene.

Background Mapping. In order to render a novel view image, denoted as the target image I_t, we need to establish dense mappings between the target view and background regions in the source views, thus the color features from the source views can be aggregated and converted to target colors via neural rendering. To establish the mapping, we cast rays through a processed mesh $\mathcal{M}'_b$, in which the foreground mesh is removed, from the inquired target camera pose $\boldsymbol{P}_t$ to derive the target background depth D_t. It is used with the pre-computed $\{\boldsymbol{P}_{b,i}\}_{i=1}^N$ for mapping. The details are explained as follows.

The mesh $\mathcal{M}_b$ derived from reconstruction contains the noisy foreground. We hope to remove that to ensure that the derived target depths are sourcing from the background. It is non-trivial to operate in mesh space $\mathcal{M}_b$. We propose to mask out foreground from the estimated depth maps using the segmentation mask before fusing them to the point cloud and mesh instead. Note that we dilate the foreground mask for completely removing the foreground because the segmentation is imperfect in the first place. As a result, we obtain a pure background mesh $\mathcal{M}'_b$.

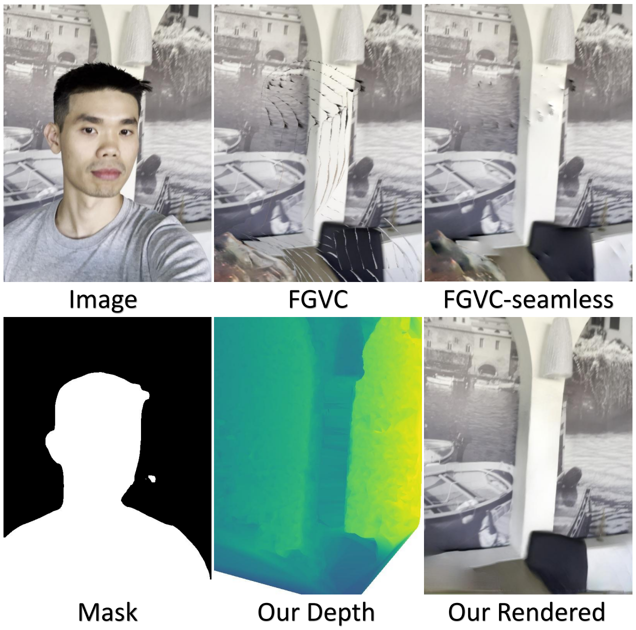

Figure 2: Full background synthesis. Our method can render full background images, and hence it enables completing the occluded background in the original video. We compare with the most recent video completion method FGVC [15]. The full video comparison is attached in the supplementary material.

After obtaining $\mathcal{M}'_b$, a target view depth at a novel viewpoint can be derived by ray-casting. An example of rendered background depth is illustrated in Fig. 2, where white regions in the depth map are not reconstructed in the mesh, and we fill them with infinite positive values. Next, a dense correspondence, $[\boldsymbol{x}_t, \boldsymbol{x}_{t,i}]$, between the target view and i-th source view can be found via

$$\boldsymbol{x}_{t,i} = \boldsymbol{K_b}\boldsymbol{T}_t^i\boldsymbol{K}_b^{-1}(\boldsymbol{x}_t D_t), \tag{1}$$

where $\boldsymbol{T}_t^i = \boldsymbol{P}_{b,i}\boldsymbol{P}_t^{-1}$ is the relative camera transformation between the two views and $\boldsymbol{x}_t$ is the pixel (homogeneous) coordinates. However, the established dense mapping is not necessarily the background-background matches since D_t can also be reprojected to foreground regions in source views. Therefore, not only computing depth consistency as conducted in [35, 36] to remove inaccurate matches, but also we eliminate invalid background-foreground matches via semantic consistency. A match is regarded as invalid if it falls into the foreground regions in source views. The dilated mask is used to avoid wrong segmentation labels. As a result, only valid background-background matches will be used in the following rendering step.

Neural Rendering. The mapping obtained above can be used to aggregate source view features such that the aggregated features can be blended into a target image. Firstly, we extract source image features via a U-Net based encoder [37], then we aggregate features $\{v_{t,i}\}_{i=0}^{N}$ for each target pixel using the valid background-background matches. Finally, we adopt an off-the-shelf rendering network [36], which takes aggregated features as input and outputs the target color, where the view-dependent effects are considered. The network is trained on Tanks and Temples dataset [18], and we find it generalizing well to our selfie videos without finetuning. A dense background rendering example is shown in Fig. 2 (Our Rendered). We refer readers to [36] for more details about feature extraction, feature aggregation, and color regression.

3.2. Foreground Synthesis

Foreground synthesis shares a similar idea as the background synthesis in Sec. 3.1. We track and reconstruct a 3D human model from the source images first, followed by establishing mapping of foreground regions between target view and source views. Once the dense correspondences are obtained, the color features are aggregated from source views and blended into a single foreground image. However, the difference is that we assume that the foreground (human) is dynamic but almost rigid, while background is always static. The dynamic nature causes geometric inconsistency between two parts, and we show that individual foreground reconstruction is helpful in reconstructing high-quality 3D human models. Moreover, we present a voting-based masking scheme from multi-view information. The mask generated will be used for fusion in later stage. Lastly, we explicitly remove frames containing geometric inconsistent foreground using a RANSAC-based inlier frame detection, which further improves the reconstruction result.

Tracking and Reconstructing Foreground. Although people cannot keep perfectly still while taking selfie videos, we observe that the movement is usually a small rigid motion, *i.e.* the body shape doesn't change much during capturing. Therefore, we can still use COLMAP [38] and MVS [39] for camera tracking and dense reconstruction as presented in Sec. 3.1. In contrast, we remove the keypoints detected in background regions for camera tracking, thus we construct a point cloud and surface mesh for foreground mapping. We discuss the benefit of this individual foreground reconstruction as follows.

First, the SfM concept builds upon the static world assumption, where we estimate the camera poses and depths w.r.t. the static world. However, in this tracking and reconstruction, we treat the dynamic person as a static object and it allows estimating the camera poses relative to the static human. Hence, this camera trajectory may be different from the actual path taken by the camera relative to the static world since the camera trajectory takes both actual camera motion and human motion into account.

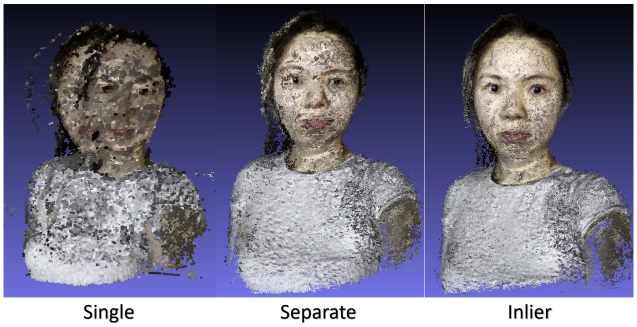

Figure 3: Point cloud visualization. Previous methods reconstruct an entire scene in a single model, which causes poor representation for foreground due to geometry inconsistency. We propose to separate two parts and detect inlier frames for foreground reconstruction.

Second, separate reconstruction avoids jointly modeling a dynamic object and a static scene under the single static world assumption, which causes inconsistent geometry on foreground as shown in Fig. 3 (Single). By separating them, the reconstruction accuracy for both regions can be improved. An example of the reconstructed foreground point cloud is shown in Fig. 3, which demonstrates the advantage of our separate reconstruction over reconstructing the whole scene together. Moreover, the quantitative results in Tab. 2 show the improved reconstruction accuracy, as represented by the decreasing in reprojection error of face landmarks from *Error-Singles* to *Error-Ours*.

Mask Aggregation. Similar to background rendering, we find correspondences between target and source views using the reconstructed mesh, and we use the pre-trained model in SVS [36] for rending images. The rendered image consists a photo-realistic human foreground with noisy background, since we perform the human-centric tracking and reconstruction. To determine the foreground mask accurately, we aggregate foreground/background labels from searched source pixels. Specifically, we regard one target pixel as foreground when more than half of its correspondences are predicted as foreground in source views. Otherwise, it is regarded as background. The results in Tab. 3 demonstrate that our estimated mask (Fig. 4(b)) is highly consistent with the predicted segmentation mask of the target view, which also reflects the high accuracy of our reconstruction.

Analyzing Reconstruction Accuracy. We propose a simple method to verify the accuracy of foreground reconstruction, which checks the reprojection error of facial landmarks. This is based on the fact that faces are usually not occluded in selfie videos. The method can also be used in extreme cases to remove outlier frames that are not well-registered. The algorithm is described below.

First, we detect 2D facial landmarks in all images using

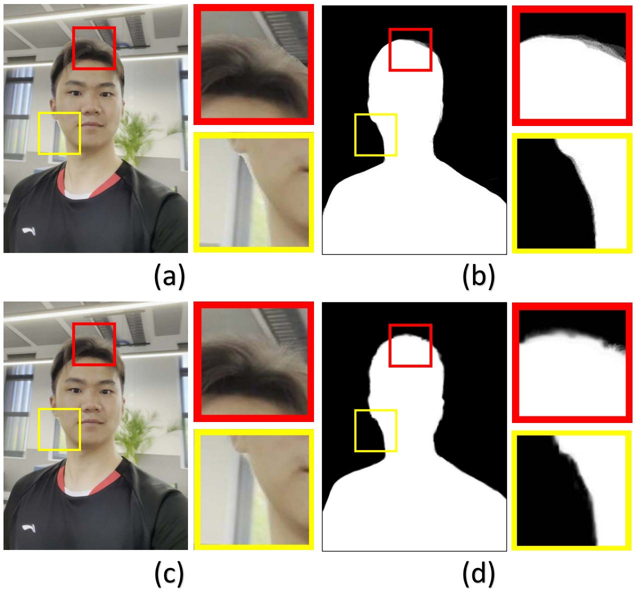

Figure 4: Mask and fusion. We show the hard fusion (a) with the estimated mask (b) and the matting fusion (c) with the alpha map (d).

the face mesh by MediaPipe [27]. With the estimated camera poses in foreground reconstruction, we randomly sample $m = 10$ images and triangulate 3D facial point clouds from the detected 2D landmarks. Second, we project the triangulated 3D facial points to all images. The projected 2D facial points is thus compared with the detected facial landmarks to obtain a reprojection error. We regard an image as an inlier if the averaged reprojection error is below a threshold, *e.g.*, $3px$. Third, we repeat the above steps for $n = 30$ times, and we save the best model that finds the most inlier frames. The reprojection error and inlier numbers can be used to analyze the reconstruction accuracy, as shown in Tab. 3. We provide an example of a reconstructed point cloud from only inlier frames in Fig. 3.

3.3. Background and Foreground Fusion

In previous sections, we introduce how to generate the background and foreground images with the segmentation mask. Here we introduce two fusion schemes, namely hard fusion and matting fusion.

Hard Fusion. Given the foreground mask by aggregation, a straightforward fusion approach is to crop the foreground image (*i.e.*, human) using the mask and paste it onto the dense background image obtained in Sec. 3.1. It works well in most regions, but an artifactual object boundary may appear as shown in Fig. 4(a).

Matting Fusion. Toward seamless fusion, we use the image matting method [26] to refine the estimated mask. An example is provided in Fig. 4. First, we dilate the mask (b) by $5px$ to generate a trimap, which indicates the opaque and unknown regions. Then we feed the trimap and foreground image into the matting network to obtain an alpha map α, as visualized in (d). Finally, we fuse foreground (F_i) and background (B_i) images by alpha matting,

$$I_i = \alpha_i * F_i + (1 - \alpha_i) * B_i \tag{2}$$

The fused result I_i, as visualized in (c), shows a smoother object boundary than the Hard Fusion result (a). Moreover, the quantitative results in Tab. 1 prove the advantage of Matting Fusion over Hard Fusion.

4. Experiments

4.1. Evaluation Dataset and Metric

Dataset. We evaluate the proposed method on a collection of our captured selfie videos. During capture, users usually undergo a small rigid motion including small movement and rotation. We use the front camera of an iPhone 12 Pro Max to capture selfie videos. The video length ranges from 5 to 15 seconds with $4K$ resolution, and we extract $100+$ images per video. Notice that the method proposed in this work is a general framework, and network training is not required on new videos, unlike the prior arts using implicit representation [29, 31].

Metric. We use three widely-used evaluation metrics for analyzing results. The PSNR and MS-SSIM [45] are used to evaluate the low-level image differences to the ground truth, and we use the LPIPS [47] metric to evaluate the high-level perception differences. It is based on convolutional features, and it better correlates with human perception.

4.2. Evaluation Results

Novel View Synthesis. We compare our method with the state-of-the-art, including NeRF [29], Nerfies[1] [31], FVS [35], and SVS [36]. The quantitative results are reported in Tab. 1, and the qualitative comparison is shown in Fig. 5. NeRF and Nerfies are trained and validated on all test images due to implicit representation, while there is no training on test sequences required in other methods. FVS, SVS and Ours firstly track and reconstruct the scene using all the frames. Since our available ground truth is the captured video itself, for evaluation purpose, we use $\{\boldsymbol{P}_i\}_{n=1}^{N}$ as the novel viewpoints and use the remaining images in the video as the source views. The synthesized target image I_t at viewpoint $\boldsymbol{P}_i$ is evaluated against I_i. 960×540 image resolution is adopted in these methods. As NeRF and Nerfies are trained and tested on 480×270 resolution images, so we re-scale the ground truth and the results of all methods to this resolution for a fair comparison.

[1] We use the official COLAB version provided by the authors.

Table 1: Novel view synthesis results. The values in brackets stand for image numbers. The PSNR (P) and MS-SSIM (M) indicate low-level image errors, and the LPIPS (L) indicate high-level perceptual errors. Ours-H stands for Hard Fusion, while Ours-M is Matting Fusion. Foreground (FG) and background (BG) synthesis are also evaluated individually for ablation study, where SVS can be viewed as our baseline.

	Methods	A (136)			B (93)			C (124)			D (123)			E (136)		
		P↑	M↑	L↓	P↑	M↑	L↓	P↑	M↑	L↓	P↑	M↑	L↓	P↑	M↑	L↓
Full	NeRF	30.955	0.976	0.162	18.638	0.632	0.610	28.729	0.949	0.188	29.890	0.955	0.214	28.019	0.959	0.185
	Nerfeis	16.758	0.623	0.486	15.640	0.556	0.458	18.404	0.596	0.418	17.027	0.537	0.458	15.594	0.605	0.392
	FVS	30.660	0.981	0.089	27.584	0.974	0.080	24.520	0.928	0.166	25.611	0.946	0.128	30.877	0.986	0.078
	SVS	30.584	0.983	**0.083**	27.345	0.967	0.087	25.308	0.933	0.160	26.032	0.938	0.135	29.581	0.980	0.079
	Ours-H	30.564	0.987	0.087	28.106	0.980	0.077	29.078	0.974	0.111	29.607	0.982	0.093	29.638	0.988	0.079
	Ours-M	**31.226**	**0.988**	**0.083**	**28.399**	**0.981**	**0.074**	**29.547**	**0.975**	**0.106**	**30.668**	**0.984**	**0.088**	**31.132**	**0.990**	**0.074**
FG	SVS	37.102	0.992	0.043	33.686	0.988	0.032	29.809	0.959	0.071	29.892	0.962	0.070	33.510	0.988	0.040
	Ours	**39.450**	**0.995**	**0.042**	**34.762**	**0.992**	**0.027**	**33.022**	**0.984**	**0.041**	**35.619**	**0.993**	**0.038**	**36.312**	**0.996**	**0.030**
BG	SVS	**34.276**	**0.995**	**0.028**	30.987	0.984	0.042	32.121	0.985	0.057	33.042	0.978	0.040	**34.816**	**0.995**	**0.031**
	Ours	33.253	0.994	0.036	**31.736**	**0.991**	**0.039**	**35.492**	**0.994**	**0.051**	**35.730**	**0.995**	**0.039**	34.302	0.994	0.039

NeRF [29] does not take dynamics into consideration which results in (1) difficulty in learning the implicit representation, revealed by the noisy synthesis in sequence B; (2) ghosting/blurring effect as shown in other videos. However, low-level metrics (PSNR and SSIM) are "friendly" on blurry result as discussed in the supplementary material so NeRF does not show the significant downgraded results in PSNR and SSIM metrics on most videos, while the qualitative result does not match with the quantitative evaluation. However, it is reflected in high-level evaluation metrics like perception error (LPIPS) in all the sequences.

Nerfies [31] reconstructs the entire deformable scene by learning a deformation field for each image. It indeed shows better visual results (Fig. 5) than NeRF, but it is not reflected in the quantitative results. Besides, we find it difficult to learn accurate deformation in our data, *e.g.*, the human orientation is different from the original image in Fig. 5. We suppose that the performance can be improved by more exhaustive training. However, we argue that this solution is inefficient and learning the deformation field from appearance is very challenging. Compared with it, our method makes the problem easier by separate modeling. Human is considered "static" w.r.t. camera in the separate foreground reconstruction such that the dynamics are suppressed.

FVS [35] and SVS [36] show similar results since their reconstructions are the same, *e.g.*, they reconstruct the scene as if it is static. This allows for generating smooth/blurry qualitative result and showing good scores in evaluation metrics, but it brings about many artifacts due to inaccurate reconstruction. For example, objects are blurred and deformed because non-match pixels are fused. Moreover, their performance drops significantly when the geometry inconsistency between the foreground and background is significant, *e.g.*, in the sequence C and D (Tab. 1). Compared with them, our method shows better results due to more accurate reconstruction, contributed by separate modeling.

New Camera Path Video. The original video is used above for quantitative evaluation. To further validate our method and compare the method with others in completely novel viewpoints, we provide two novel camera paths for qualitative evaluation, including an interpolated path and a spiral path. First, we extend the original video (20fps) to a high frame rate (60fps) video by interpolating the original camera trajectories. Note that we have two camera paths for the foreground and background, and we interpolate them independently and fuse the rendered results together. In this case, the viewpoint difference to the original one is small. Second, we render a spiral path [31], as shown in Fig. 1. This is very challenging due to large viewpoint changes. Nevertheless, our method still shows the compelling results in most videos. We attach the rendered videos in the supplementary materials.

Background Completion. Our background synthesis can be used for video completion, and we show a visual comparison to FGVC [15] in Fig. 2, which is based on optical flow and achieves state-of-the-art performance. We discuss the advantages and disadvantages of both methods below: (a) Our method works better than FGVC in static scenes by leveraging a globally consistent reconstruction. However, FGVC works well in general dynamic scenes since 3D reconstruction is not considered in most video completion frameworks. (b) Our method is faster than FGVC, *e.g.*, our method takes about one hour for dense reconstruction and rendering, while FGVC requires up to four hours to complete a video. Both methods are tested on the video with 100+ images and 960×540 resolution. (c) Our method can render a novel viewpoint background image, while FGVC can only complete the original video. A video comparison is provided in the supplementary materials.

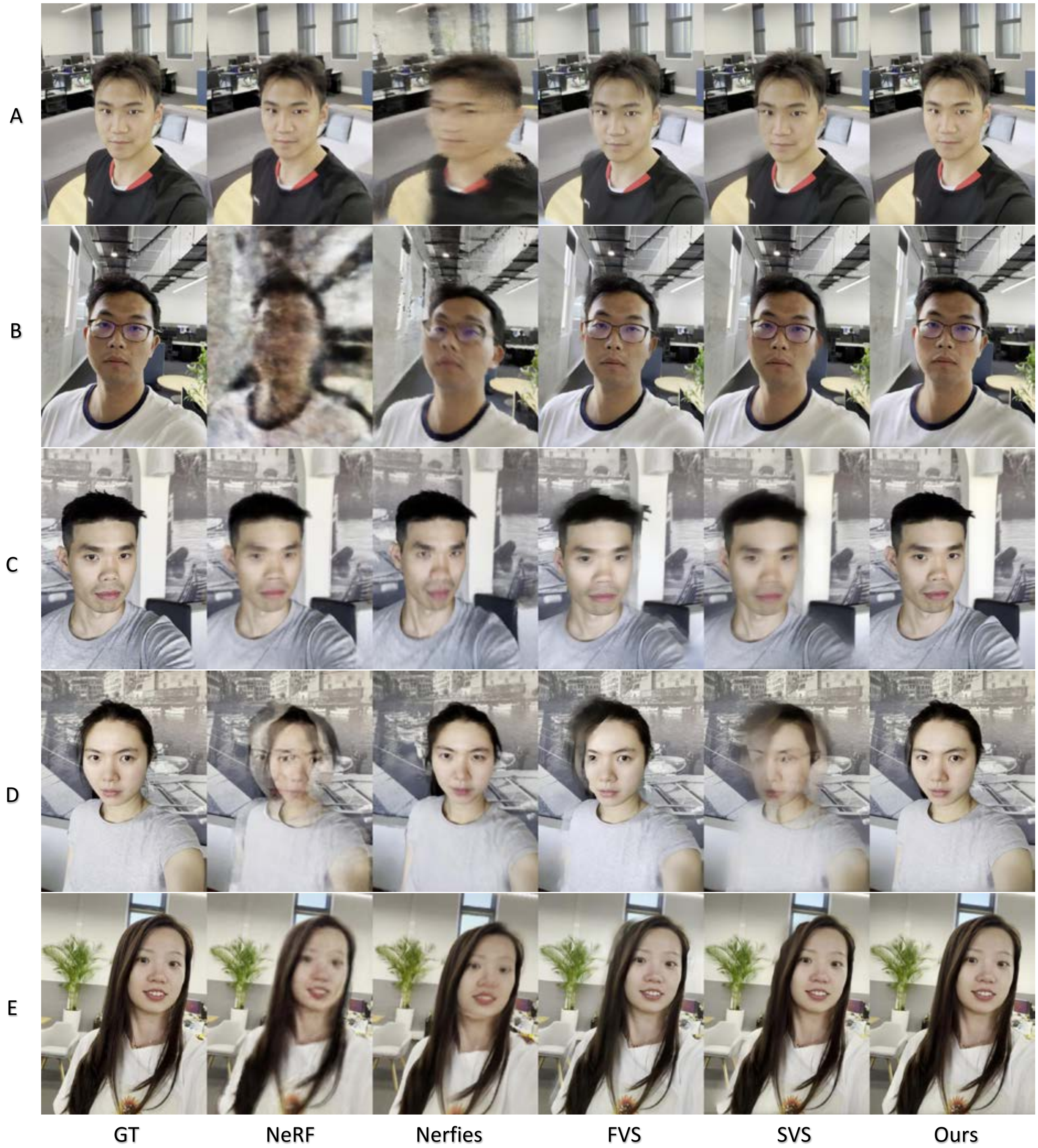

Figure 5: Synthesized novel views. The dataset name is consistent with Tab. 1. Images are cropped for visualization. Find the full video comparison in the supplementary material.

Running Time. Our method takes about one hour for tracking and reconstruction from 100+ images with 960×540 resolution, and it renders a novel view image at 1 fps in a single Nvidia RTX-2080 GPU. FVS and SVS take half of the time for dense reconstruction because our method reconstructs two parts separately. The image rendering speed

Table 2: Reconstruction accuracy. Previous methods reconstruct the whole scene in a single coordinate system, and Ours conducts separate reconstruction. We use reprojection error of face landmarks for analysis, and we regard images with a low reprojection error ($< 3px$) as inlier.

Dataset	A	B	C	D	E
Frame	136	93	124	123	136
Error-Single (px)	3.53	4.87	5.42	26.11	6.39
Error-Ours (px)	**1.47**	**2.36**	**2.85**	**2.87**	**2.18**
InlierNum.-Single	68	23	25	1	23
InlierNum.-Ours	**136**	**83**	**95**	**71**	**123**

for these two methods is similar to ours. NeRF [29] requires about ten hours for training a video with 480 × 270 resolution in the same device, and it takes about 6 seconds to render a new image. Nerfies requires 8 GPUs for training the full-featured model, and we train the author-provided COLAB version due to equipment limitation. It requires about 10+ hours for training using Google TPUs, and the inference speed is about 2 seconds per image.

4.3. Ablation Study

Separate Reconstruction. Our major motivation in this work is overcoming the geometry inconsistency between foreground and background by separate modeling. The baseline method is SVS [36], which reconstructs the whole scene together. We evaluate the foreground/background synthesis individually and present the result in Tab. 1, where we use DeepLabV3 [7] to predict the segmentation mask, and we erode both regions by $11px$ for avoiding including the object boundaries in the evaluation. The results demonstrate that our foreground rendering is clearly better than SVS, due to more accurate representation by converting the dynamic foreground to a "static" scene w.r.t. the camera. Our background synthesis also shows more accurate geometry than that of SVS by excluding the foreground in tracking and reconstruction. In sequences A and E, our method is comparable to SVS. We conjecture that it is because we may remove too much foreground in the process.

Reconstruction Accuracy. We analyze the foreground reconstruction accuracy using the reprojection error of the facial landmarks (Sec. 3.2) and present the result in Tab. 2. It shows that the reprojection error of separate reconstruction has been significantly reduced from a single reconstruction. The number of inlier frames is also reported. A visual comparison is provided in Fig. 3, where using only the inlier frames for foreground reconstruction leads to better visual performance. We mainly use this method for analysis purposes in this work, without deducting the outlier frames from the original source set. It is because our separate reconstruction is good enough for our collected videos and there is no much difference in quantitative evaluation. Nevertheless, this method can be applied for videos with inconsistent foreground geometry and produce a more stable synthesis for rendering novel view images. We attach demo videos in the supplementary material.

Table 3: Mask evaluation. We compare our mask that is aggregated from source views to the mask that is predicted on the target view by DeepLabV3 [7].

Dataset	A	B	C	D	E
IoU (%)	99.4	99.5	98.9	99.1	99.4

Aggregated Mask. To validate the accuracy of our aggregated mask from source views, as introduced in Sec. 3.2, we compare it with the mask predicted by segmentation networks. Here we use DeepLabV3 [7] to predict the mask for all images and use it as ground truth. We use the intersection over union (IoU) metric for evaluation and present the result in Tab. 3, which demonstrates that the aggregated mask is highly consistent (99%) with the ground truth.

Matting Fusion. We evaluate the effectiveness of the fusion schemes proposed in Sec. 3.3. The quantitative comparison between Hard Fusion and Matting fusion is presented in Tab. 1, where Matting Fusion (Ours-M) consistently outperforms Hard Fusion (Ours-H) in all sequences. As discussed in Sec. 3.3 and shown in Fig. 4, Matting Fusion further details the foreground/background mask with better object boundaries. Moreover, since the input to the matting algorithm [26] includes a rough mask (the aggregated mask in our case) and a reference image (rendered foreground image), the matting result also depends on the quality of rendered foreground images. This reflects the good foreground rendering ability of our method.

5. Conclusion

In this paper, we propose a novel system to reconstruct selfie videos casually captured by mobile phones for free-viewpoint rendering. Inspired by traditional geometric methods, we separately track, reconstruct, and synthesize background and dynamic foreground to avoid their geometry inconsistency. By using several excellent computer vision techniques in our system, including geometric reconstruction, semantic segmentation, neural rendering, and image matting, the proposed method is able to render full background and accurate foreground images, and seamlessly fuse the synthesized foreground/background images together. Leveraging geometric methods and deep learning modules, our system can perform novel view synthesis in an efficient way. More importantly, the method can be applied to any selfie videos without further training. Both quantitative and qualitative results demonstrate the advantage of our proposed method over existing alternatives.

References

[1] Kara-Ali Aliev, Dmitry Ulyanov, and Victor Lempitsky. Neural point-based graphics. *arXiv preprint arXiv:1906.08240*, 2019. 2

[2] Berta Bescos, José M Fácil, Javier Civera, and José Neira. Dynaslam: Tracking, mapping, and inpainting in dynamic scenes. *IEEE Robotics and Automation Letters*, 2018. 2

[3] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 1999. 2

[4] James Booth, Anastasios Roussos, Allan Ponniah, David Dunaway, and Stefanos Zafeiriou. Large scale 3d morphable models. *International Journal on Computer Vision (IJCV)*, 2018. 2

[5] Sofien Bouaziz, Yangang Wang, and Mark Pauly. Online modeling for realtime facial animation. *ACM Transactions on Graphics (TOG)*, 2013. 2

[6] Gaurav Chaurasia, Sylvain Duchene, Olga Sorkine-Hornung, and George Drettakis. Depth synthesis and local warps for plausible image-based navigation. *ACM Transactions on Graphics (TOG)*, 2013. 2

[7] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 3, 8

[8] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[9] Inchang Choi, Orazio Gallo, Alejandro Troccoli, Min H Kim, and Jan Kautz. Extreme view synthesis. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 2

[10] Peng Dai, Yinda Zhang, Zhuwen Li, Shuaicheng Liu, and Bing Zeng. Neural point cloud rendering via multi-plane projection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[11] Abe Davis, Marc Levoy, and Fredo Durand. Unstructured light fields. In *Computer Graphics Forum*, 2012. 2

[12] Yilun Du, Yinan Zhang, Hong-Xing Yu, Joshua B. Tenenbaum, and Jiajun Wu. Neural radiance flow for 4D view synthesis and video processing. *arXiv preprint arXiv:2012.09790*, 2020. 2

[13] Andrew Fitzgibbon, Yonatan Wexler, and Andrew Zisserman. Image-based rendering using image-based priors. *International Journal on Computer Vision (IJCV)*, 2005. 2

[14] Guy Gafni, Justus Thies, Michael Zollhöfer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4D facial avatar reconstruction. *https://arxiv.org/abs/2012.03065*, 2020. 2

[15] Chen Gao, Ayush Saraf, Jia-Bin Huang, and Johannes Kopf. Flow-edge guided video completion. In *European Conference on Computer Vision (ECCV)*, 2020. 3, 6

[16] Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. Deep blending for free-viewpoint image-based rendering. *ACM Transactions on Graphics (TOG)*, 2018. 2

[17] Peter Hedman, Tobias Ritschel, George Drettakis, and Gabriel Brostow. Scalable inside-out image-based rendering. *ACM Transactions on Graphics (TOG)*, 2016. 2

[18] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (TOG)*, 2017. 4

[19] Johannes Kopf, Michael F Cohen, and Richard Szeliski. First-person hyper-lapse videos. *ACM Transactions on Graphics (TOG)*, 2014. 2

[20] Patrick Labatut, Jean-Philippe Pons, and Renaud Keriven. Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. In *IEEE International Conference on Computer Vision (ICCV)*, 2007. 3

[21] Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, and Zhaoyang Lv. Neural 3d video synthesis, 2021. 2

[22] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. *https://arxiv.org/abs/2011.13084*, 2020. 2

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014. 3

[24] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM Transactions on Graphics (TOG)*, 2015. 2

[25] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 2004. 3

[26] Hao Lu, Yutong Dai, Chunhua Shen, and Songcen Xu. Index networks. *IEEE Transactions on Pattern Recognition and Machine Intelligence (PAMI)*, 2020. 2, 5, 8

[27] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*, 2019. 5

[28] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[29] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 5, 6, 8

[30] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[31] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo-Martin Brualla. Deformable neural radiance fields. *arXiv preprint arXiv:2011.12948*, 2020. 1, 2, 5, 6

[32] Sida Peng, Yuanqing Zhang, Yinghao Xu, Qianqian Wang,

Qing Shuai, Hujun Bao, and Xiaowei Zhou. Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. *arXiv preprint arXiv:2012.15838*, 2020. 2

[33] Eric Penner and Li Zhang. Soft 3d reconstruction for view synthesis. *ACM Transactions on Graphics (TOG)*, 2017. 2

[34] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-NeRF: Neural radiance fields for dynamic scenes. *https://arxiv.org/abs/2011.13961*, 2020. 2

[35] Gernot Riegler and Vladlen Koltun. Free view synthesis. In *European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 3, 5, 6

[36] Gernot Riegler and Vladlen Koltun. Stable view synthesis. *arXiv preprint arXiv:2011.07233*, 2020. 1, 2, 3, 4, 5, 6, 8

[37] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 2015. 4

[38] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 3, 4

[39] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 2, 3, 4

[40] Harry Shum and Sing Bing Kang. Review of image-based rendering techniques. In *Visual Communications and Image Processing 2000*, 2000. 2

[41] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *ACM Transactions on Graphics (TOG)*, 2019. 2

[42] Justus Thies, Michael Zollhöfer, Christian Theobalt, Marc Stamminger, and Matthias Nießner. Image-guided neural object rendering. In *International Conference on Learning Representations (ICLR)*, 2020. 2

[43] Philip HS Torr. Geometric motion segmentation and model selection. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 356(1740):1321–1340, 1998. 2

[44] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a deforming scene from monocular video. *https://arxiv.org/abs/2012.12247*, 2020. 2

[45] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, 2003. 5

[46] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[47] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 5