Learning multiplane images from single views with self-supervision

Gustavo Sutter P. Carvalho*
Diogo C. Luvizon*
Antonio Joia Neto
André G. C. Pacheco
Otávio A. B. Penatti

AI R&D Lab
Samsung R&D Institute Brazil
Campinas, SP, 13097-160, Brazil

Abstract

Generating static novel views from an already captured image is a hard task in computer vision and graphics, in particular when the single input image has dynamic parts such as persons or moving objects. In this paper, we tackle this problem by proposing a new framework, called CycleMPI, that is capable of learning a multiplane image representation from single images through a cyclic training strategy for self-supervision. Our framework does not require stereo data for training, therefore it can be trained with massive visual data from the Internet, resulting in a better generalization capability even for very challenging cases. Although our method does not require stereo data for supervision, it reaches results on stereo datasets comparable to the state of the art in a zero-shot scenario. We evaluated our method on RealEstate10K and Mannequin Challenge datasets for view synthesis and presented qualitative results on Places II dataset.

1 Introduction

Predicting novel views from a single image is a key problem in computer vision and graphics, with potential applications related to image editing, photo animation, robotics, and virtual reality. This challenging task requires estimating the scene geometry while filling in occluded regions in the source image. For this reason, most of the current solutions require stereo data during inference [17, 25, 29], which eases the process of estimating the depth of the scene and helps inpainting occluded regions by gathering relevant visual information from different points of view. However, in many applications such as animating an old photo, multiple views are not available during inference.

Even during the training stage, multiple views of the same scene are not easy to obtain. Despite the fact that state-of-the-art neural network models require a large number of images in varied conditions for training, capturing a massive amount of stereo data with calibrated and synchronized cameras is an onerous process. As an alternative, structure-from-motion (SfM) and Multi-View Stereo (MVS) approaches such as COLMAP [22] have been used to estimate the camera parameters from videos [34]. However, since MVS depends on correspondence matching at different frames, MVS-based approaches might not be suitable for

 * These authors contributed equally.
 This work was funded by Samsung Eletrônica da Amazônia Ltda., through the project "Parallax Effect", within the scope of the Informatics Law No. 8248/91.

dynamic scenes with moving people and objects. Considering this limitation, one of the goals of our method is to learn to generate static novel views from challenging scenes, including snapshots of scenes with dynamic content or moving people, without explicitly depending on stereo data for it.

Although a few methods have been proposed to tackle novel view synthesis from a single image, they are based on point-cloud [18] or shape representations [24], which require rendering techniques that are not easily integrated into current deep learning frameworks, apart from limiting possible applications for the same reason, since some target devices have limited 3D rendering capabilities. As an alternative, multiplane images (MPI) [26] have been proposed to allow rendering of apparent 3D effects from a set of 2D images.

In this paper, we present CycleMPI, a new framework for learning an MPI representation in a self-supervised manner, which does not require multi-view stereo data for training. Instead of relying on multiple views of the same scene for supervision, we propose to learn an MPI with a cycle consistency supervision, as illustrated in Fig. 1, which depends only on single images and their correspondent (estimated) depth maps. Our method can be trained with massive data collected from the Internet, resulting in a robust solution even for zero-shot evaluation, as demonstrated by our results on challenging benchmarks.

The main contributions of our work are summarized as follows. We propose a new MPI learning framework based on cycle consistency loss. Through an extensive experimental evaluation, we demonstrated that our method can be trained with single view images and estimated depth maps. Our approach achieved results close to the state of the art on RealEstate10K in a zero-shot scenario, even when comparing to methods trained on this dataset or using stereo depth predictions during inference. In the remaining of this paper, we present a review of the literature in Section 2. In Section 3, we present our method. The experiments and our conclusions are presented in Sections 4 and 5, respectively.

2 Related work

In this section, we go through some of the most relevant work related to our method.

Single-image view synthesis. The goal of single-image view synthesis is to generate images of a scene from novel viewpoints using only one image as input. It states a harder problem than interpolating between two or more views of the same scene, given that occluded information that might be revealed on the multi-image setting is not available on single-image approaches. Tucker and Snavely [28] proposed to learn an MPI representation from a stereo pair of images and an additional depth supervision. Despite achieving good results on static scenes, their method still requires multi-view stereo data for training, which hinders the generalization capability. Huang et al. [11] proposed a view synthesis method based on generative networks. In their approach, a scene is artificially generated from segmentation masks. Differently, in our work, we target to generate an MPI from an existing photo. Other more *ad hoc* methods generate an intermediate representation based on a predefined sequence of steps, involving depth estimation, image inpainting, and additional filtering stages [16, 24]. This strategy usually results in better generalization, but has the drawback of requiring an elevated computational cost during inference and complex designing requirements. A common characteristic of previous methods, as in [24, 28], is the requirement for precise depth maps during training, which are generally estimated by structure-from-motion.

Monocular depth estimation. Single-image view synthesis is highly dependent on the depth of the scene, as previously noted in [28]. In recent years, many works tackled monoc-

ular depth estimation with leaning-based methods, in particular deep neural networks. The use of standard convolutional networks with a scale-invariant loss function was first introduced by Eigen et al. [7], followed by the use of more complex methods and architectures that further improved the results [2, 4, 14]. To deal with poor generalization, Ranftl et al. [20] combined multiple datasets using a flexible loss function, resulting in impressive zero-shot performance. Recently, methods based on the Vision Transfomer (ViT) architecture [6] have achieved state-of-the-art results for the task [21] [3]. Contemporary methods use monocular depth estimation to improve robustness and generalization of stereo depth estimation [31] and optical flow [1]. In a similar fashion, our work explore monocular depth estimation to improve the generalization of novel view synthesis using MPI representation.

Cycle consistency loss. The use of cycle consistency loss, also referred to as reconstruction loss in some contexts, has recently gained relevance as an objective function in unsupervised and self-supervised problems. In general, this framework is used to learn transformations over the original data without having necessarily a set of transformed samples for supervision. In order to obtain a supervision signal, the inverse transformation is applied to the intermediate representations, allowing for the reconstruction of the input samples. In this direction, cycle consistency has been used for image generation with conditional GANs, where the learned transformation is the mapping from one object category to another [13, 35] or even in a many-to-many fashion [5]. Hani et al [8] use a consistency loss on the features of the learned transformation to perform novel-view synthesis of 3D shapes without a ground-truth 3D model. Their method, however, diverges from ours by using a neural scene representation and needing two images of an object during training. In our method, we explore the idea of cycle consistency loss to allow a self-supervised training strategy that enables our network to learn inpainting information from single-view images.

3 Method

The goal of our method is to synthesize novel views from a single image, through an intermediate MPI representation, without requiring stereo data during training. For this, we propose a neural network that predicts an MPI from a single view. Our neural network is supervised by two main components: depth estimation loss and visual consistency loss. The depth estimation loss can be supervised with predictions (pseudo ground-truth) from a teacher model. This prevents our method from requiring data with precise and dense depth maps, which is hard to obtain on large scale and in varied conditions. The visual consistency is enforced by image losses, which also include inverse projection and cyclic losses. In this section, we briefly review the MPI representation and present our framework in detail.

3.1 Background on multiplane images

A *multiplane image* (MPI) [26, 34] can be defined as a set of front-parallel planes, where each plane is an RGBA image, i.e., it encodes color and transparency, and is placed at a fixed depth w.r.t. the camera. Different from other representations, like LDI [23] or point-cloud [18], MPI is capable of encoding complex scenes, including transparency and soft edges, can be easily rendered in a fully differentiable way, and benefits real-time applications.

MPI representation. Given an input image $\mathbf{I}$ of shape $H \times W \times 3$ and a neural network f_θ, a multiplane image is obtained by:

$$\{(c_i, \alpha_i)\}_{i=1}^{D} = f_\theta(\mathbf{I}), \tag{1}$$

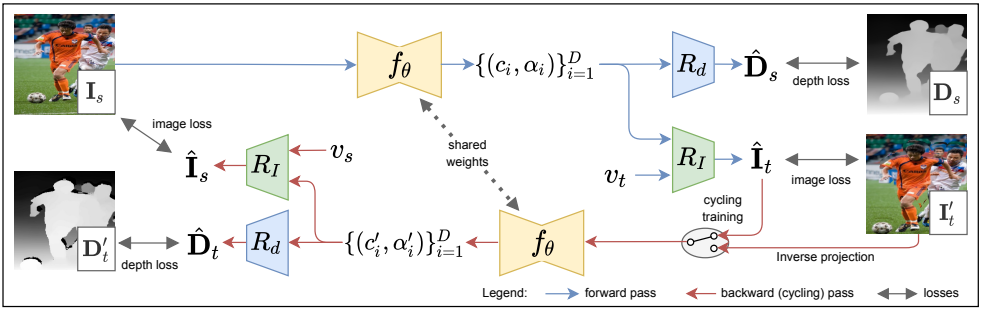

Figure 1: Proposed CycleMPI: a source image $\mathbf{I}_s$ is fed into a neural network f_θ, which produces an MPI representation w.r.t. the *source viewpoint* v_s. This MPI is then rendered in the target viewpoint v_t, resulting in $\hat{\mathbf{I}}_t$, which can be fed back into the network. The new MPI produced w.r.t. to the *target viewpoint* is then rendered in the source viewpoint, closing the loop for cyclic supervision. Depth maps are used as additional supervisions.

where c_i and α_i represent the RGB color and alpha values of the i^{th} image plane, respectively, and D is the total number of planes, assuming $i = 1$ the farthest plane. Since directly predicting the RGBA values for each image plane would be highly over-parameterized, it is more common to predict a background image and the alpha values of each image plane, as in [34]. In such a manner, the colors of each plane are defined by a combination of the source image $\mathbf{I}$ and the predicted background image $\hat{\mathbf{I}}_b$:

$$c_i = w_i \odot \mathbf{I} + (1 - w_i) \odot \hat{\mathbf{I}}_b, \tag{2}$$

where $\odot$ is the Hadamard product. The blending weights w_i are computed from α_i [28]:

$$w_i = \alpha_i \prod_{j=i+1}^{D} (1 - \alpha_j), \tag{3}$$

which is essentially the over-composite operation in the alphas. Therefore, the actual output of the neural network can be simply the background image $\hat{\mathbf{I}}_b$ and the alpha values $\{\alpha_i\}_{i=1}^D$.

Warping. The MPI representation can be rendered at different points of view through warping and compositing processes. Warping is implemented as a standard inverse homography [9], where each image plane is projected from the source to a target viewpoint, considering a given intrinsic and extrinsic camera parameters and the depth d_i of each image plane. More specifically, if we consider the warping operation $\mathcal{W}_{v_s,v_t}$ from the source viewpoint v_s to the target viewpoint v_t, the projected color and alpha planes are given by:

$$c'_i = \mathcal{W}_{v_s,v_t}(d_i, c_i), \quad \alpha'_i = \mathcal{W}_{v_s,v_t}(d_i, \alpha_i). \tag{4}$$

Since we consider images from unknown cameras and MPI layers at fixed depths, we can assume standard intrinsics based on the image size and virtual viewpoints with normalized camera pose. For more details about the warping process, please refer to [34] and [28].

Compositing. After warping, an MPI can be rendered into a target viewpoint v_t by the over-composite operation [19]. As noted in [28], the over-compositing operation can be performed on both image and depth domains. In the latter, a depth map can be directly regressed from the alpha channels, which is particularly useful to enforce structural constraints. Considering

w'_i as the blending weights from the warped alpha layers α' (from Equation 3), the predicted target image and the regressed depth map (w.r.t. the source) are obtained by:

$$\hat{\mathbf{I}}_t = \sum_{i=1}^{D}(c'_i w'_i), \quad \hat{\mathbf{D}}_s = \sum_{i=1}^{D}(d_i^{-1}\alpha_i w_i), \tag{5}$$

where d_i^{-1} is the disparity of the i^{th} plane (we use disparity and depth interchangeably).

3.2 Proposed approach

In our method, we assume that the ground truth for the target image $\mathbf{I}_t$ is not available. Although this is a common scenario, where stereo data is not available for training, this case cannot be handled by previous approaches based on MPI learning, as discussed in Section 2. For making it possible to train our model from end-to-end and without requiring stereo data, we propose two key strategies, described as follows.

First, we warp the source image into the target viewpoint, considering a Lambertian projection of non-occluded regions, based on the corresponding depth map and using morphological filtering to reduce artifacts. This allows us to recover an approximation of $\mathbf{I}_t$, designated by $\mathbf{I}'_t$, which has been demonstrated by our experiments to be sufficient and effective for learning an MPI representation that encodes non-occluded regions from the source image. This step is designated as *forward pass* in Fig. 1. *Second*, in order to enforce the network to learn to generate color textures for occluded regions, we propose a training scheme based on cycle consistency: the neural network should be able to predict an MPI from the *target image* that allows recovering the *source image*. In other words, when considering a cycle consistency training, instead of only feeding the source image as input to the neural network, we also input the warped images $\mathbf{I}'_t$ or $\hat{\mathbf{I}}_t$ (simply referred as $\mathbf{I}_t^{(\cdot)}$) into f_θ, which produces an MPI w.r.t. the *target viewpoint*, which corresponds to the *backward pass* in Fig. 1. Then, this MPI is projected back to the *source viewpoint*, where the supervision signal is the *source image* $\mathbf{I}_s$.

Note that both the warped image $\mathbf{I}'_t$ and the predicted view $\hat{\mathbf{I}}_t$ can be used as input in this process. To make the difference clear, we refer to the former as *inverse projection training* and to the latter as *cyclic training*. This strategy enforces the network to learn to inpaint the occluded regions since the source image has all the pixel information that was occluded in $\mathbf{I}_t^{(\cdot)}$. This process is illustrated in Fig. 1, and the details of each part are presented next.

3.2.1 Self-supervision signal

Let us consider a source image $\mathbf{I}_s$ captured from viewpoint v_s. Our goal in this step is to reconstruct an approximation of the target image $\mathbf{I}_t$ from viewpoint v_t, so this reconstructed image could be used as an auxiliary supervision signal. For this, we consider a depth map $\mathbf{D}_s$ as additional information from the source viewpoint, which can be obtained with state-of-the-art monocular depth estimation models (e.g., [21]). Then, each pixel from the source image is warped to the target viewpoint v_t, as illustrated in Fig. 2.

Image warping and filtering. A naive forward warping would result in sparkle image artifacts or "flying pixels", as shown in Fig. 2 (c). Instead, we propose an alternative warping process, where the source image is split into an MPI-like representation, directly obtained by slicing the image accordingly to the depth map. More specifically, we quantize the depth map $\mathbf{D}_s$ into integer values $i \in [1,\dots,D]$ expanding it as a binary representation $\mathbf{q}$ of shape

$H \times W \times D$. Then, each binary mask $\mathbf{q}_i$ (of shape $H \times W$) is filtered by a sequence of morphological operations to remove flying pixels and to expand the boundaries of connected regions by a few pixels. In practice, we perform a morphological erosion followed by a dilation, with equivalent filters of size 3×3 and 7×7, respectively. Note that the filters' size might be adapted to different image resolutions.

Background filling. The warped image and its mask (Fig. 2 d-e) would be enough for training a network with L1 loss in the forward pass. However, we may need additional loss functions to improve the quality of predictions, e.g., the VGG perceptual loss [12]. For this, we require a dense target and predicted images. We followed the strategy proposed in [31] for filling in the holes in the warped images with real image samples. In our case, we use the source image of the same sample flipped horizontally. Although this strategy is very simple and efficient, the results are generally plausible in a global perspective, as in Fig. 2 (f).

(a) (b) (c) (d) (e) (f)

Figure 2: Reconstruction of the target image from the source image (a) and depth map (b); (c) naive forward projection and (d) forward projection after morphological filtering; (e) the resulting mask of valid pixels and (f) the reconstructed approximation $\mathbf{I}'_t$ of the target image.

3.2.2 Training with reverse projection and cyclic loss

The approximation of the target image reconstructed by our method does not provide a direct supervision signal for learning to inpaint occluded regions in the scene, since it does not have information corresponding to the occluded pixels in the source image. To handle this, we propose a new training strategy, where the input of the network is the approximation $\mathbf{I}'_t$ or the prediction $\hat{\mathbf{I}}_t$, and $\mathbf{I}_s$ is used for supervision. In this case, we can re-write Equation 1 as:

$$\{(c_i^t, \alpha_i^t)\}_{i=1}^{D} = f_\theta(\mathbf{I}_t^{(\cdot)}), \tag{6}$$

where c_i^t and α_i^t are the color and alpha values w.r.t. the target viewpoint. The same process of warping and compositing an MPI, described in Section 3.1, can be applied to the new (c_i^t, α_i^t) values, considering a proper swapping between the source and target viewpoints, which results in the MPI $\{(c_i'^t, \alpha_i'^t)\}_{i=1}^{D}$. Finally, we can render the estimated image in the *source viewpoint* and the depth map in the *target viewpoint* (from Equation 5):

$$\hat{\mathbf{I}}_s = \sum_{i=1}^{D}(c_i'^t w_i'^t), \quad \hat{\mathbf{D}}_t = \sum_{i=1}^{D}(d_i^{-1}\alpha_i^t w_i^t), \tag{7}$$

where $c_i'^t$ and $w_i'^t$ represent the color values and blending weights after warping from the *target* to the *source* viewpoint, and w_i^t is the blending weights from the target viewpoint.

Depth map warping and filtering. From Equation 7, we can supervise $\hat{\mathbf{I}}_s$ with the source image $\mathbf{I}_s$. However, a supervision signal is not available for $\hat{\mathbf{D}}_t$. A naive approach would be to estimate $\mathbf{D}_t$ using a monocular depth estimation method from $\mathbf{I}'_t$. This approach would be computationally expensive during training. Instead, we warp the depth map $\mathbf{D}_s$ to the target

viewpoint, using a similar warping approach as described in Section 3.2.1. In this case, we performed a morphological erosion and dilation with equivalent filters of size 3×3 and 5×5. This process results in $\mathbf{D}'_t$, as illustrated in Fig. 1.

3.2.3 Neural network model

We implement the neural network f_θ with the EfficientNet model [27] cut at block 5 as encoder and a sequence of upsampling and depth-wise convolutions as the decoder. The main outputs of the network are i) the background image with shape $H \times W \times 3$ and ii) the alpha values with shape $H \times W \times (D-1)$, since the last alpha value is always opaque. In addition, the network also outputs intermediate depth predictions at scales $H/2 \times W/2$ and $H/4 \times W/4$, which are also supervised during training to help the network to learn the structure of the scene with multi-resolution supervision. The intermediate depth predictions are performed by casting depth regression as a multi-layer classification problem, similar to [3]. More details about the network implementation and the intermediate supervisions are in the supplementary material.

3.2.4 Training and loss functions

The network f_θ can be trained in three different ways: i) direct projection, where the source image $\mathbf{I}_s$ is used as input and $\mathbf{I}'_t$ and $\mathbf{D}_s$ are used as supervisions; ii) inverse projection, where the reconstructed target image $\mathbf{I}'_t$ is used as input and $\mathbf{I}_s$ and $\mathbf{D}'_t$ are used as supervision, or iii) cyclic training, where we first perform a direct projection, then we do an inverse projection using as input to the neural network the predicted view $\hat{\mathbf{I}}_t$. In practice, when we refer to *inverse projection training*, we alternate between direct projection and inverse projection randomly with equal probability. During training, we also generate random target viewpoints v_t (assuming v_s as the identity) with small camera movements.

For supervising the estimated depth maps, we use a common loss function composed of an L1 plus multi-scale gradient matching terms [14]:

$$\mathcal{L}_{depth} = \|\mathbf{D} - \hat{\mathbf{D}}\| + \alpha \sum_k (\|\nabla_x(\mathbf{D}^k - \hat{\mathbf{D}}^k)\| + \|\nabla_y(\mathbf{D}^k - \hat{\mathbf{D}}^k)\|), \tag{8}$$

where $\mathbf{D}$ and $\hat{\mathbf{D}}$ are the reference and predicted depths, and k is the scale. Similarly to [14], we used $k = 4$ scales and $\alpha = 0.5$.

For supervising the image reconstruction, we used three different losses:

$$\mathcal{L}_{pix} = \|\mathbf{I} - \hat{\mathbf{I}}\|, \quad \mathcal{L}_{vgg} = \sum_j \|\Phi_j(\mathbf{I}) - \Phi_j(\hat{\mathbf{I}})\|, \quad \mathcal{L}_{style} = \sum_j \|G_j^\Phi(\mathbf{I}) - G_j^\Phi(\hat{\mathbf{I}})\|, \tag{9}$$

where $\mathbf{I}$ and $\hat{\mathbf{I}}$ are the reference and predicted views, Φ is the 16-layer VGG model, G^Φ is the Gram matrix from VGG features, and j indexes the first three VGG blocks. Please refer to [12] for more details about VGG and style losses. The final loss is given by:

$$\mathcal{L}_{total} = \mathcal{L}_{depth} + \mathcal{L}_{pix} + \beta \mathcal{L}_{vgg} + \gamma \mathcal{L}_{style} \tag{10}$$

4 Experiments

In this section, we evaluated our method considering a challenging case where no stereo data is available for training, i.e., our model is trained using a single view dataset with no

Method	SSIM ↑		PSNR ↑		LPIPS ↓	
	$n=6$	$n=10$	$n=6$	$n=10$	$n=6$	$n=10$
Single-View Synthesis [28]	0.844	0.795	23.50	20.80	0.138	0.209
3D-Photography [24]	0.859	0.795	24.30	21.27	0.087	0.141
Adaptive-MPI [16]	0.804	0.735	21.09	18.23	0.113	0.187
Ours	0.842	0.786	22.79	19.96	0.115	0.176
Ours + fine-tuning on RE10K	0.845	0.788	22.85	19.89	0.129	0.196

Table 1: Comparison with previous methods on RealEstate10K. We fine-tuned our model using 10% of training images from RE10K (without ground-truth depths). Note that [28] was trained on RE10K with ground-truth depths and [24] uses multiple views during inference to estimate depth.

Method	FID ↓	SSIM ↑	PSNR ↑	LPIPS ↓
Single-View Synthesis [28]	21.24	0.617	13.56	0.514
Ours	16.10	0.633	14.02	0.479

Table 2: Comparison of MPI end-to-end approaches on Mannequin Challenge dataset.

ground truth depth nor target images, and is evaluated on well-known stereo datasets from the literature, which characterizes a zero-shot evaluation scenario.

4.1 Datasets and evaluation metrics

Training datasets. For training, we consider two different datasets: Places II [33] and a dataset with images from Flickr [16], here referred to as Flickr-Pictures dataset. The former is composed of about 1.8M images and the latter has about 95K images in high resolution, which were resized to 1024×768 at least, keeping the original ratio. For both datasets, we computed the depth maps for supervision using the available model from [21].

Evaluation datasets. RealEstate10K [34] is a common dataset for novel view synthesis, composed of 70K sequences extracted from YouTube videos. We used the evaluation set composed of 1500 testing triplets proposed by [24], choosing the source and target images to be 6 and 10 frames apart. We also evaluated our method on the Mannequin Challenge dataset [15], which is traditionally a depth estimation dataset based on structure-from-motion (SfM). This is a challenging dataset since it contains complex scenes with many people and often with blurry images. The evaluation was performed on the official test split, choosing the source and target frames to be 3 frames apart.

4.2 Implementation details

We trained our model using Adam as optimizer, with parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, and learning rate 10^{-3}, with batches of 4 images for about 10 million iterations. We set the optimization parameters of our method as $\beta = 0.01$ and $\gamma = 0.0001$, according to our ablation studies in the Appendices.

4.3 Comparison with previous methods

RealEstate10K. We compare our method with [28], [24], and [16] on RealEstate10K, evaluating the SSIM [30], PSNR, and LPIPS [32] metrics between the images produced by each

Training strategy on Places II						Validation on RE10K		
$\mathcal{L}_{depth}$	$\mathcal{L}_{pix}$	$\mathcal{L}_{vgg}$	$\mathcal{L}_{style}$	Inverse proj.	Cyclic	SSIM ↑	PSNR ↑	LPIPS ↓
✓						0.734	17.699	0.237
	✓					0.758	19.349	0.280
✓	✓					0.760	19.473	0.215
✓	✓	✓				0.752	19.332	0.195
✓	✓	✓	✓			0.735	18.748	0.183
✓	✓	✓	✓	✓		0.761	19.556	**0.182**
✓	✓	✓	✓		✓	**0.765**	**19.773**	**0.182**

Table 3: Ablation study considering different training strategies in our method.

method and the ground truth. The results are presented in Table 1. We replicated the results from all previous methods using the author's implementation and pre-trained models. Our results are close to the state of the art, even considering that [28] was trained on RealEstate10K and [24] uses multiple views for depth estimation, while our method was trained only with single images from Places II and does not use multiple views during inference. We also fine-tuned our model using 10% of training images from RealEstate10K, which resulted in a slight improvement in the SSIM metric.

Mannequin Challenge. One advantage of our self-supervised method is its capability to generalize on challenging scenarios. To demonstrate this, we evaluated our method on the Mannequin Challenge, which contains complex scenes with people and moving objects, and is more than fifty times smaller than RealEstate10K, making it unfeasible for training deep networks for view synthesis. Another challenge in this dataset is the wide variety of scene scales, making it hard for any scale-free 3D representation to generate novel views correctly aligned with the ground truth data. We refer to the Appendix for examples of the misalignment problem. For this reason, we also reported results using the Fréchet Inception Distance (FID) [10], an evaluation metric suited for ill-posed problems, to deal with misalignment issues. We focus on MPI models trained end-to-end, comparing our self-supervised model trained on Places II with Single-View Synthesis [28] trained on RealEstate10K. Results on Table 2 show the superiority of our model in this challenging case.

4.4 Ablation studies

We also performed ablation experiments with different training strategies of our method to better understand their relative importance, which is presented in Table 3. In all these experiments, we trained our model with images from Places II for 500K iterations, using a fixed image resolution of 384×288 pixels. We tested several combinations of the loss functions in the final loss, considering inverse projection and cyclic training. We can notice that depth and pixel losses are complementary for learning. Although these two losses alone already resulted in satisfying SSIM and PSNR metrics, the LPIPS result is poor, which reflects the low texture quality of the generated views. For this, we included the VGG and style losses, which increase LPIPS metric substantially. From our experiments, we noticed that LPIPS is very sensitive to blurry images, therefore this metric is relevant for more realistic image predictions. We can observe that both inverse projection and cyclic training have improved all the metrics consistently, which supports our main idea.

In order to better understand the proposed cyclic training strategy in regard of the availability of target views and different depth estimation models, we performed additional experiments in Table 4. First, we trained our model on RE10K considering a simple forward

Training strategy			Validation on RE10K		
Training dataset	Depth model	Cyclic	SSIM ↑	PSNR ↑	LPIPS ↓
RE10K	DPT		0.767	20.729	0.207
RE10K	DPT	✓	0.772	20.959	0.181
Places365	DPT		0.735	18.748	0.183
Places365	DPT	✓	0.765	19.773	0.182
Places365	AdaBins	✓	0.760	19.948	0.229

Table 4: Comparison of different training data and depth maps by DPT [21] and AdaBins [3].

training (first row) and cyclic training, which improved all the metrics. One could also argue that the DPT model [21] was trained with multiple datasets, including stereo data. We show that our approach does not depend on stereo data by comparing DPT and AdaBins [3], which was trained NYU depth, a Kinect based single depth dataset. From this evaluation, we can observe that both models resulted in close SSIM and PSNR metrics, although AdaBins degraded the LPIPS metric.

Finally, we presented some qualitative results from our method in Fig. 3. This evidences the capability of our approach to generalize well on different and challenging types of images. The main failures of our method are due to wrong depth predictions of our model, which is a consequence of incorrect pseudo ground-truth depth used during training. Blurry images are also observed for large baselines, which is a common problem on MPI based methods. Please see the supplementary material and video demonstration of the method for more results.

Figure 3: Results on Places II: source image (top), new view (bottom), and selected planes.

5 Conclusions

In this paper, we presented a new method for learning an MPI representation with intermediate depth and cyclic supervision. The proposed method can be trained with single view data by exploring inverse projection and cyclic training strategies for learning to inpaint occluded regions in the scene. We compared our approach with recent methods for single view synthesis from the literature. Our method was evaluated in a zero-shot scenario and achieved results comparable to the results in the state of the art, even when compared to methods trained on the target datasets. We also presented ablation studies to show the relative importance of each component of the proposed framework, which evidenced the benefits of our method.

References

[1] Filippo Aleotti, Matteo Poggi, and Stefano Mattoccia. Learning optical flow from still images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15201–15211, 2021.

[2] Ibraheem Alhashim and Peter Wonka. High quality monocular depth estimation via transfer learning. *arXiv e-prints*, abs/1812.11941:arXiv:1812.11941, 2018.

[3] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4009–4018, June 2021.

[4] Weifeng Chen, Zhao Fu, Dawei Yang, and Jia Deng. Single-image depth perception in the wild. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[5] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[7] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems*, pages 2366–2374, 2014.

[8] Nicolai Hani, Selim Engin, Jun-Jee Chao, and Volkan Isler. Continuous object representation networks: Novel view synthesis without target view supervision. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6086–6099. Curran Associates, Inc., 2020.

[9] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[10] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *arXiv preprint arXiv:1706.08500*, 2017.

[11] Hsin-Ping Huang, Hung-Yu Tseng, Hsin-Ying Lee, and Jia-Bin Huang. Semantic view synthesis. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 592–608, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58610-2.

[12] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[13] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks, 2017.

[14] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[15] Zhengqi Li, Tali Dekel, Forrester Cole, Richard Tucker, Noah Snavely, Ce Liu, and William T. Freeman. Learning the depths of moving people by watching frozen people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[16] Diogo C Luvizon, Gustavo Sutter P Carvalho, Andreza A dos Santos, Jhonatas S Conceicao, Jose L Flores-Campana, Luis GL Decker, Marcos R Souza, Helio Pedrini, Antonio Joia, and Otavio AB Penatti. Adaptive multiplane image generation from a single internet picture. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2556–2565, 2021.

[17] 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 *ECCV*, 2020.

[18] Simon Niklaus, Long Mai, Jimei Yang, and Feng Liu. 3d ken burns effect from a single image. *ACM Transactions on Graphics*, 38(6):184:1–184:15, 2019.

[19] Thomas Porter and Tom Duff. Compositing digital images. In *Proceedings of the 11th Annual Conference on Computer Graphics and Interactive Techniques*, page 253–259. ACM, 1984.

[20] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[21] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. *ArXiv preprint*, 2021.

[22] Johannes L. Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113. IEEE. ISBN 978-1-4673-8851-1. doi: 10.1109/CVPR.2016.445.

[23] Jonathan Shade, Steven Gortler, Li-wei He, and Richard Szeliski. Layered depth images. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '98, page 231–242, New York, NY, USA, 1998. Association for Computing Machinery. ISBN 0897919998. doi: 10.1145/280814.280882.

[24] Meng-Li Shih, Shih-Yang Su, Johannes Kopf, and Jia-Bin Huang. 3d photography using context-aware layered depth inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[25] Pratul P. Srinivasan, Richard Tucker, Jonathan T. Barron, Ravi Ramamoorthi, Ren Ng, and Noah Snavely. Pushing the boundaries of view extrapolation with multiplane images. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 175–184, 2019. doi: 10.1109/CVPR.2019.00026.

[26] Richard Szeliski and Polina Golland. Stereo matching with transparency and matting. *International Journal of Computer Vision*, 32(1):45–61, 1999.

[27] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR, 2019.

[28] Richard Tucker and Noah Snavely. Single-view view synthesis with multiplane images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[29] Shubham Tulsiani, Richard Tucker, and Noah Snavely. Layer-structured 3d scene inference via view synthesis. In *ECCV*, 2018.

[30] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. doi: 10.1109/TIP.2003.819861.

[31] Jamie Watson, Oisin Mac Aodha, Daniyar Turmukhambetov, Gabriel J. Brostow, and Michael Firman. Learning stereo from single images. In *European Conference on Computer Vision (ECCV)*, 2020.

[32] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

[33] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[34] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. In *SIGGRAPH*, 2018.

[35] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.