Depth from Defocus Approaches for Video Depth Estimation

Zhengyang Wei

Abstract—Depth estimation remains challenging to perform with a single image due to the loss of three-dimension information. Inspired by depth from defocus and the scale-consistent video-based learning, we propose a video depth estimation method called DfD-SC-Depth, which applies unsupervised training with monocular snippets on the Depth from Defocus model. We demonstrate the performance of our proposed method and the baseline on the NYUv2 dataset and analyze the results qualitatively and quantitatively. Thanks to the combination of the strength of each technique, our method achieves better performance. Furthermore, this approach offers a possibility for using a phase-coded aperture camera’s video to improve the depth from defocus model without extra depth information.

Index Terms—Depth from Defocus, Video Depth Estimation, Video-based Learning

1 INTRODUCTION

Video depth information is important for robotics, autonomous driving, 3D reconstruction, and beyond [1]. Depth information can be acquired by expensive sensors like LIDAR and stereo cameras. Generating high-quality depth-from-color can inexpensively complement these sensors.

Monocular depth estimation (MDE) from a single color image is challenging. Estimating absolute depth from a color image is ill-posed without a second image for triangulation [2]. Most MDE approaches rely on contextual cues to evaluate the relative location of objects in an image [3]. Depth from focus processes are used for single image depth estimation and outperform many state-of-art methods [4]. Well-designed end-to-end coded aperture MDE technique can take full advantage of the monocular depth cue [5] and achieve higher accuracy for depth estimation.

Video depth estimation considers the depth of a single frame and the Spatio-temporal relationship between adjacent video frames, which improves the performance of depth estimation [2]. Furthermore, video-based learning doesn’t have an impact on single-image-based tasks. Still, it’s critical for video-based applications [6], which makes it a practical approach to improve the performance of depth estimation.

However, the depth from defocus method hasn’t been applied to video video-based learning for depth estimation yet.

This paper proposes a video depth estimation method called DfD-SC-Depth, which applies unsupervised training with monocular snippets on the Depth from Defocus model and compares the proposed method with our baselines qualitative and quantitatively. Specifically, our contributions are the following:

1) We proposed a framework that combined the unsupervised video-based training based on scale consistency with the depth from defocus method.
2) We conduct comparison experiments on NYUv2 dataset [7] and qualitatively and quantitatively analyze the results.
3) We proposed a feasible plan to improve models with videos captured by the phase-coded aperture camera without depth information.

2 RELATED WORK

2.1 Depth from defocus method for MDE

Besides pictorial cues, defocus blur is a vital depth cue for monocular depth estimation. The camera can produce defocus blur according to the depth of the object and camera settings when capturing images. So, applying it to depth estimation has become a trend.

Using images with defocus blur in deep learning approaches outperforms the use of all-in-focus images [8]. Conventional lenses or hand-crafted phase-coded apertures have been regarded as powerful depth estimation techniques [9]. With the development of deep optics, the End-to-End coded aperture was used to improve the defocus blur and encode more information [10]. Ikoma et al. proposed a framework for single RGB image depth estimation, including an occlusion-aware image formation model, a rotationally symmetric phase-coded aperture, and the corresponding preconditioning approach [5]. It can also estimate an all-in-focus image, which is helpful to extract the Spatio-temporal relationship between adjacent frames. The depth from defocus method is one of the most promising ways to solve the ill-posed problem.

2.2 Video depth estimation

Unlike depth estimation for a single image, video depth estimation usually takes information between frames into account. As a result, researchers attempt to design many
self-supervised or unsupervised approaches to realize the optimization of depth estimation models, mainly utilizing the frame information of videos.

Monodepth2 uses minimum reprojection loss to tackle occlusions between frames and auto-masking loss to ignore stationary pixels [2]. Packnet has symmetrical packing and unpacking blocks, and it uses the neighbor frames in temporal context to realize self-supervised scale-aware structure-from-motion [11]. SC-Depth penalizes the inconsistency of predicted depths of adjacent frames and resolves fine-scale details using a defocus-aware depth filter [13]. However, almost all the video depth estimation methods haven’t considered the depth from defocus approaches.

3 PROPOSED METHOD

In this section, we describe our proposed DfD-SC-Depth model. The overall pipeline is illustrated in Fig. 1.

3.1 Depth from Defocus Model

3.1.1 Image Formation Models

The imaging system’s depth-dependent point spread function (PSF) can be controlled by the surface height variation of the diffractive optical element DOE of the phase-coded aperture [14]. [14] proposes to use Equ. 1 to model the PSF.

\[
PSF(\rho, z, \lambda) = \frac{2\pi}{\lambda s} \int_0^{+\infty} r D(r, \lambda, z) P(r, \lambda) J_0(2\pi r \rho) dr \]  

(1)

\(\rho\) and \(r\) are the radial distance on the sensor and aperture planes; \(\lambda\) is the wavelength; \(s\) is the distance between lens and sensor, and \(J_0\) is the zeroth-order Bessel function. \(D(r, \lambda, z)\) is the defocus factor modeling the depth variation of the PSF for points at a distance \(z\) from the lens and given by

\[
D(r, \lambda, z) = \frac{z}{\lambda (r^2 + z^2)^{\frac{3}{2}}} e^{i \frac{2\pi}{\lambda}(\sqrt{r^2 + z^2} - \sqrt{r^2 + z^2}) + d^2} \]  

(2)

[15] proposed a radially symmetric DOE design which reduces the required memory and time for optimization. The corresponding phase delay \(P(r, \lambda)\) is defined as

\[
P(r, \lambda) = a(r) e^{i \frac{2\pi}{\lambda}(n(\lambda) - n_{air}) b(r)} \]  

(3)

Here \(n_{air}\) is the refractive index of air, and \(a(r)\) is the transmissivity of the phase mask. [5] adopts nonlinear differentiable image formation model to the wavelength- and depth-dependent PSF, which is given by Equ. 4, where \(l_k\) is the sub-images, and \(\alpha_k\) is the binary masks composed by quantized depth maps.

\[
E_k(\lambda) := PSF_k * \sum_{k'} \alpha_k' \\
\tilde{I}_k := \frac{PSF_k(\lambda) * I_k}{E_k(\lambda)} \\
\tilde{a}_k := \frac{PSF_k(\lambda) * \alpha_k(\lambda)}{E_k(\lambda)} \\
b(\lambda) = \sum_{k=0}^{K-1} \tilde{I}_k \prod_{k' = k+1}^{K-1} (1 - \tilde{a}_k') + \eta \\
\]  

(4)

The nonlinear model can generate realistic defocus images from color images and their corresponding densely labeled depth maps for our method.

3.1.2 UNet-based Estimation for Image and Depth

We adopt the architecture proposed in [5]. For defocus images produced by the nonlinear image formation models, Tikhonov-regularized least squares method is used to map the 2D image into a multilayer representation \(I^{(est)} \in \mathbb{R}^{M \times N \times K}\), as shown in Equ. 5. It can generate a layered representation with sharper details.

\[
l^{(est)} = \arg \min_{l \in \mathbb{R}^{M \times N \times K}} \| b - \sum_{k=0}^{K-1} PSF_k * l_k \|^2 + \gamma \| l \|^2 \\
\]  

(5)

Then, both the defocus image and its corresponding multilayer representation are concatenated, which is the input for the UNet architecture. And the detail of the implementation of the network is shown in TABLE 1.

3.2 Scale-consistent Depth Learning from Video

3.2.1 Framework overview

The framework is aimed to train the DfD depth network and pose network from unlabeled videos. For two adjacent frames \(I_a, I_b\) which are randomly sampled from a video, the model can respectively estimate their depth maps \(D_{a, r}, D_{b, r}\) and relative camera pose \(P_{ab}\). So the reference depth \(D_{a, r}\) can be synthesized with the source depth \(D_{b, r}\) by differentiable warping [6]. Then the reference depth maps can be used for training the whole model as shown in Fig. 1.

3.2.2 Training Loss Function

[6] proposed a kind of objective function for scale-consistent model, which is formulated as follows:

\[
L = \alpha L_p^M + \beta L_S + \gamma L_G \\
\alpha, \beta, \gamma \text{ are the loss weighting terms.} \\
L_p^M \text{ stands for the photometric loss } L_P \text{ weighted by the proposed } M_a \text{ and is defined as:} \\
L_p^M = \frac{1}{|V|} \sum_{p \in V} (M_a(p) \cdot L_P(p)) \\
\text{ with the synthesized } I'_a \text{ and the reference image } I_a \text{ from predicted depth } D_a \text{ and pose } P_{ab}, \text{ } L_P \text{ is formulated as:} \\
L_P = \frac{1}{|V|} \sum_{p \in V} \left( \lambda \| I_a(p) - I'_a(p) \|_1 + (1 - \lambda) \frac{1 - \text{SSIM}_{aw}(p)}{2} \right) \\
\]  

(6)

(7)

(8)
Fig. 1. Our Framework. Given two adjacent coded images randomly sampled from the video, their depth maps and all-in-focus images are first estimated by the DID models. Then the relative camera pose is calculated by the PoseNet. With the predicted depth and pose, the reference depth for the first frame is synthesized with the estimated depth of the second frame and camera pose. Then the network is supervised by the photometric loss weighted by a self-discover mask, the smoothness loss, and the geometry consistent loss. Finally, the losses are averaged over valid areas determined by an auto-mask.

SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1) (2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1) (\sigma_x^2 + \sigma_y^2 + C_2)} \tag{9}

where \( V \) is the set of valid points that are successfully projected from \( I_a \) to the image plane of \( I_b \); \( p \) is the generic point in \( V \); \( x, y \) stands for two \( 3 \times 3 \) patches around the central pixel; \( C_1 \) and \( C_2 \) are constants; \( \mu \) and \( \sigma \) are mean and variance of the image color respectively. And the depth inconsistency map \( D_{\text{diff}} \) for each \( p \in V \) and self-discovered mask \( M_a \) are defined as:

\[ D_{\text{diff}}(p) = \frac{|D_a^x(p) - D_b^x(p)|}{D_a^x(p) + D_b^x(p)} \tag{10} \]

\[ M_a(p) = \begin{cases} 1 & \text{if } \|I_a(p) - I_a'(p)\|_1 < \|I_a(p) - I_b(p)\|_1 \\ 0 & \text{otherwise} \end{cases} \tag{14} \]

where \( M_a \) is a binary mask for each point in \( V \), and \( I_a' \) is the warped image from \( I_b \) using the estimated depth and pose. It removes the points where the identity mapping results in a lower loss.

4 Experimental Results

This section discusses the design of our experiments and then analyzes and evaluates the qualitative and quantitative results.

4.1 Dataset

The NYU-Depth V2 (NYUv2) data set is comprised of video sequences from a variety of indoor scenes as recorded by both the RGB and Depth cameras [7]. We choose it for our experiments since it has densely labeled depth maps which can help us generate defocus images. And we use the officially provided 654 densely labeled images for testing. For the training data, we use videos of 27 different scenes.

L_S stands for the edge-aware smoothness loss. Formally,

\[ L_S = \sum_p \left( e^{-\nabla I_a(p) \cdot \nabla D_a(p)} \right)^2 \tag{12} \]

where \( \nabla \) is the first derivative along spatial directions.

\( L_G \) is the geometric consistency loss which can be computed with the inconsistency map.

\[ L_G = \frac{1}{|V|} \sum_{p \in V} D_{\text{diff}}(p) \tag{13} \]

which minimizes the geometric inconsistency of predicted depths for two adjacent frames.

What’s more, the loss is averaged over valid points, which are determined by the auto-mask \( M_a \) proposed in (Godard et al. 2019). For each \( p \in V \), we have

\[ M_a(p) = \begin{cases} 1 & \text{if } \|I_a(p) - I_a'(p)\|_1 < \|I_a(p) - I_b(p)\|_1 \\ 0 & \text{otherwise} \end{cases} \tag{14} \]

1. https://www.cs.huji.ac.il/~wyweiss/Colorization/
4.4 Qualitative Results

4.4.1 Results

The qualitative comparison with the baselines is shown in Fig. 2 and the detail of the comparison is shown in Fig. 3.

4.4.2 Analysis and Evaluation

Our DfD-SC-Depth method visually obtains the depth maps closest to the ground truth. DfD-SC-Depth can protect the sharper edges for the estimated depth maps and get more accurate depth for the area far from the camera compared with SC-Depth. DfD-SC-Depth can also modify some apparent errors in the depth maps of the DfD model.

Our method can successfully capture edges of the doors, walls, windows in Row 1, 4, 5 of Fig. 2, and the details are shown in Fig. 3; while SC-Depth leads to blurring artifact and DfD method leads to aliasing artifact.

Our DfD-SC-Depth can also preserve the shape of objects in the images. For example, we can tell from the chairs and tables obtained by our method easily in Row 1, 6 of Fig. 2. However, it’s difficult to point out where the chairs are in depth-maps obtained by DfD and SC-Depth in Row 1 of Fig. 2.

SC-Depth sometimes might fail to accurately get the depth for the area far from the camera. For example, in Row 2, 3, 4 of Fig. 2, the deepest regions of the SC-Depth’s depth map are much different from the ground truth, but our method’s results are almost consistent with the ground truth in these regions.

There are some apparent errors in the depth map of the DfD method, such as the right-bottom of Row 2 of Fig. 2 and right-top of Row 7 of Fig. 2. Fortunately, our DfD-SC-Depth can also effectively modify these artifacts.
4.5 Quantitative Results

4.5.1 Results

The quantitative comparison with baselines is shown in TABLE 2.

<table>
<thead>
<tr>
<th></th>
<th>SC-Depth</th>
<th>DfD</th>
<th>DfD-SC-Depth</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSE ↓</td>
<td>0.399</td>
<td>0.426</td>
<td>0.372</td>
</tr>
<tr>
<td>AbsRel ↓</td>
<td>0.214</td>
<td>0.215</td>
<td>0.198</td>
</tr>
<tr>
<td>Log10 ↓</td>
<td>0.089</td>
<td>0.102</td>
<td>0.095</td>
</tr>
<tr>
<td>$a_1$ ↑</td>
<td>0.682</td>
<td>0.649</td>
<td>0.688</td>
</tr>
<tr>
<td>$a_2$ ↑</td>
<td>0.828</td>
<td>0.830</td>
<td>0.846</td>
</tr>
<tr>
<td>$a_3$ ↑</td>
<td>0.927</td>
<td>0.916</td>
<td>0.923</td>
</tr>
</tbody>
</table>

4.5.2 Analysis and Evaluation

As shown in TABLE 2, the RMSE, AbsRel, $a_1$, and $a_2$ of our proposed DfD-SC-Depth method are much better than the baselines. However, Log10 and $a_3$ of SC-Depth are a little better than our method, which might be caused by the fact that SC-Depth can yield fewer points that deviate too far from the ground truth. However, our approach can achieve better accuracy overall.

5 Conclusion

5.1 Limitations and Future Work

First, this method is sensitive to the learning rate since there is no ground truth for training data, and the reference depth is generated by the predicted depth and pose. So if the learning rate is too large, the model might jump in the wrong direction.

Second, we use the coded defocus images generated by RGBD images. And since the size of densely labeled images we can acquire is very limited, our experiment isn’t conducted on a significant number of unlabeled training data as expected. Therefore, the final results may not reach the best situation.

In the future, if we can use the phase-coded aperture camera to capture videos, we will validate our method with more training data. We also consider combining other types of video consistency information with depth from defocus methods.

5.2 Conclusion

We successfully applied unsupervised video-based training on depth from the defocus method. We experimentally demonstrate that our approach produces better performances than the baseline on the NYUv2 dataset. Even though the improvement is relatively slight, the method is still inspiring. Using the phase-coded aperture camera to capture videos or continuous images, we might improve the models without extra depth information. That ensures the method is feasible for real-life applications and can be widely applied to specific scenes.
ACKNOWLEDGMENTS

I would like to express my greatest gratitude to Prof. Gordon Wetzstein for his fascinating lectures and well-prepared projects and my project mentor, Mark Nishimura, for giving me useful insights.

REFERENCES