Fixing Defect of Photometric Loss for Self-Supervised Monocular Depth Estimation

Shu Chen, Zhengdong Pu, Xiang Fan, and Beiji Zou

***Abstract*— View-synthesis-based methods have shown very promising results for the task of unsupervised depth estimation in single images. Most existing approaches synthesize a new image and employ it as the supervision signal for depth and pose prediction. There are two problems in these approaches: 1) There are many combinations of pose and depth that can synthesize a certain new image; therefore, reconstructing the depth and pose based on the view-synthesis method from only two images is an inherently ill-posed problem; 2) The model is trained under the photometric consistency assumption that the brightness or gradient is constant when applied to the video sequences. However, this assumption is easily violated in realistic scenes due to light changes, reflective surfaces and occlusions. To overcome the first drawback, we exploit the point cloud consistency constraint to eliminate ambiguity. To overcome the second drawback, we use threshold masks to filter dynamic and occluded points and introduce matching point constraints that implicitly encode the geometry relationship between two matched points to improve the precision of depth prediction. In addition, we employ epipolar constraints to compensate for the instability of the photometric error in textureless regions and varying illumination conditions. The experimental results on the KITTI, Cityscapes and NYUv2 datasets show that the method can improve the accuracy of depth prediction and enhance the robustness of the model in handling textureless regions and illumination changes. The code and data are available at https://github.com/XTUPRLAB/FixUnDepth.**

***Index Terms*— Photometric consistency, 3D reconstruction, epipolar geometry.**

I. Introduction

RECOVERING the 3D structure of a scene is a key problem in computer vision because it can be widely applied in many fields such as autonomous driving and virtual reality. This problem is traditionally studied by geometric methods. The invented techniques include simultaneous localization and mapping (SLAM) and structure from motion (SfM) according to whether they work on-line. Geometric methods commonly perform well in constrained surroundings. However, they fail in extreme conditions when there is drastic motion, textureless features or quick light changes. To overcome this problem, deep learning is employed to estimate the depth of a scene. The gains achieved by deep learning approaches against geometric methods mainly come from tremendous training data. Since deep models commonly are able to capture high-level semantic information from low level clue learning, deep learning approaches perform better even in ill-posed regions compared with geometric methods.

Deep learning based depth estimation approaches can be roughly divided into two categories: supervised and unsupervised. Compared with supervised methods, unsupervised approaches are more general and highly applicable because they do not require a large amount of labeled data. Most unsupervised methods for learning depth and ego-motion use a synthesized view as the supervisory signal [21]–[26]. The model is trained under the photometric consistency assumption that the brightness or gradient is constant when applied to the video sequences. This implies that the convergence to the local minimum yields the correct solution. In practice, this assumption is easily violated because the light is prone to change, especially in outdoor conditions. On the other hand, the synthesized view is mapped by projecting one image onto an adjacent camera according to the estimated depth and camera poses. This approach builds upon the insight that a geometric view synthesis system only performs consistently well when its intermediate predictions of the scene geometry and the camera poses correspond to the physical ground truth. It works when the synthesized view and the corresponding depth and camera poses have one-to-one correspondence. However, recovering the corresponding depth and camera poses from a certain synthesized view is an ill-posed problem, as shown in Fig. 1. In most cases, several possible outputs correspond to a given synthesized view. Therefore, the estimated depth and camera poses which produce the minimum of the objective function are not always the best ones because they are entangled with the depth and motion networks.

In this paper, we propose an unsupervised learning framework FixUnDepth to fix defect of photometric loss for monocular depth estimation. We argue that the precision of estimated depth can be improved when the uncertainty of camera poses is reduced. In this work, we are inspired by an intuition that the estimated camera poses are correct only when the poses between two cameras estimated from forward and backward

Manuscript received October 8, 2020; revised January 12, 2021 and February 15, 2021; accepted March 22, 2021. Date of publication March 24, 2021; date of current version March 9, 2022. This work was supported in part by the Natural Science Foundation of Hunan Province under Grant 2017JJ2252, in part by the Education Department of Hunan Province under Grant 16B258, and in part by the National Key Research and Development Program of China under Grant 2018AAA0102102. This article was recommended by Associate Editor J. Yu. *(Corresponding author: Shu Chen.)*

Shu Chen, Zhengdong Pu, and Xiang Fan are with the School of Computer Science, Xiangtan University, Xiangtan 411105, China, and also with the School of Cyberspace Security, Xiangtan University, Xiangtan 411105, China (e-mail: csu_cs@163.com; bobpuluck@163.com; franc_zi@163.com).

Beiji Zou is with the School of Computer Science and Engineering, Central South University, Changsha 410083, China (e-mail: bjzou@csu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2021.3068834.

Digital Object Identifier 10.1109/TCSVT.2021.3068834

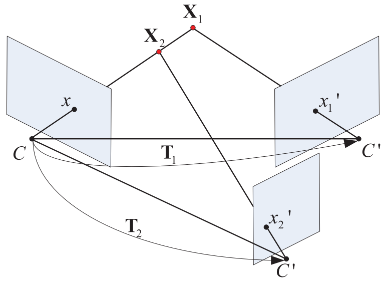


Fig. 1. This figure shows the estimated ambiguous 3D coordinates of a 2D point x, $\mathbf{X}_1$ and $\mathbf{X}_2$, under different poses $\mathbf{T}_1$ and $\mathbf{T}_2$. Notice that projected points x_1' and x_2' have the same image coordinates.

(transform one camera to another camera and vice versa) are completely inverse. In this work, we design two pose networks to predict the poses of two consecutive image pairs. Pose estimations of two consecutive frame pairs must follow the same principle that the two pose networks are formed as a twin network in which they share parameters. During the training process, the poses from forward and backward are estimated from two consecutive samples, and the pose consistency constraint is implicitly enforced by the epipolar geometry constraint. Additionally, we analyze the inherent drawbacks of photometric loss and propose using epipolar constraints to overcome this problem. We also use threshold masks [38] to filter the potential dynamic and occluded points which can reduce the adverse impact affection on the photometric loss. We explore the intermediate geometric information encoded by matching point constraints to guide the optimizer to move to the correct direction during training. The established geometry prior is obtained by the traditional feature extraction and matching approaches that are free from light changes. We also investigate the problem caused by multi-scales loss and try two different approaches in our experiments to find the best result.

Our main contributions are as follows: 1) Point cloud consistency is enforced to eliminate ambiguity in the estimated results. 2) The positions of the matched feature pairs constraint and the epipolar geometry constraint are employed to make up for the drawback of the photometric consistency assumption.

II. Related Work

A. Traditional Geometric Methods

There is a large body of work that uses geometric constraints to recover the camera motion and the structure of a scene [1]–[3]. Existing techniques can be roughly divided into two categories: indirect formulation and direct formulation. Indirect approaches first estimate an atomic model by extracting intermediate geometric representations, such as keypoint [4] and optical flow [5]. Each new coming image is then increasingly added to expand the structure. The accumulated geometric error is minimized either with sliding-window or period bundle adjustment [6]. Indirect approaches only use a particular intermediate feature to estimate the structure. That limits them to take advantage of the full image cues. As a consequence, indirect approaches may fail in some cases, such as low texture, stereo ambiguities, and occlusions, which commonly appear in natural scenes.

In comparison, direct approaches skip the feature extraction step and directly use the intensity of pixels in the image to optimize the photometric error. Most direct approaches employ a photometric error as well as a geometry prior to estimate dense or semi-dense geometry [7], [8]. The main drawback of adding a geometry prior is the introduction of correlations between geometry parameters, which renders a statistically consistent, joint optimization in real-time infeasible. Engel *et al.* [9] proposed to optimize a photometric error defined directly on the images, without incorporating a geometry prior. Since direct approaches can sample from across all available data, they generate a more complete model and lend more robustness in sparsely textured environments. However, this formulation tends to have a heavy load that requires a state-of-the-art GPU to run in real-time. Furthermore, with the presence of reflective surfaces, dynamic moving objects, and inaccurate photometric calibration, this formulation is less robust than indirect methods.

B. Supervised Depth Estimation

With the development of deep learning, some researchers have attempted to use CNN to estimate depth and poses. Most early works relied on the labeled data from depth sensors as supervision to learn depth [10], [11], [41]–[43]. Eigen *et al.* [10] used two stacked deep networks to find depth relations from a single image. Some improved techniques that built upon the success of this approach include refining results via a hierarchical condition random field (CRF) [12] and using a fully convolutional network (FCN) [13]. Since this task is inherently ambiguous, the generalization of these approaches is still questionable. Recent supervised approaches prefer a stereo set [39]. Flynn *et al.* [14] used a cost volume combined with a separate conditional color model to predict novel viewpoints in a multi-view stereo setting. Kendall *et al.* [15] proposed a novel end-to-end deep learning architecture for regressing disparity from a rectified pair of stereo images. Since the images on the stereo rig have a fixed and known transformation, the depth can be efficiently learned from that functional relationship.

Apart from learning depth only, some approaches have proposed to jointly estimate optical flow, depth, and motion [16]. Ummenhofer *et al.* [17] proposed a composition of multiple stacked encoder-decoder networks to estimate depth, motion, surface normals, and optical flow. They showed that learning these multiple tasks jointly leads to better performance. DeepTAM [18] estimated the poses and depth via two individual sub-networks indicating tracking and mapping respectively.

C. Unsupervised Learning From Video

Unsupervised or self-supervised learning approaches are more attractive because they do not require ground-truth. The self-supervised approaches are inspired by the idea of warping-based view synthesis [19], [20]. Garg *et al.* [21] presented an approach to recover depth from stereo pairs based on the photometric consistency constraint, where the camera motion between a stereo pair is known. Godard *et al.* [22] improved this approach using left-right consistency constraints. Zhou *et al.* [23] proposed unsupervised learning for depth and ego-motion estimation using only monocular video sequences. They estimated the depth and poses via two separate networks and the output of two networks were used to synthesize a new view that serves as the supervisory signal. Li *et al*. [24] used both spatial (between left-right pairs) and temporal (forward-backward) photometric warp error, which constrains the scene depth and camera motion to be in a common, real-world scale, and the same idea was proposed by Zhang *et al.* [25]. Mahjourian *et al*. [26] suggested to explicitly consider the inferred 3D geometry of the scene, enforcing consistency of the estimated 3D point clouds and ego-motion across consecutive frames. Since the depth value of input data can hardly be regressed exactly to the ground-truth value, Cao *et al*. [40] proposed to formulate depth estimation as a pixelwise classification task. To limit the effect of dynamic objects in the video sequences, Vijayanarasimhan *et al*. [27] explicitly modelled the object motion which is integrated with camera motion to estimate the optical flow. Yin and Shi [28] developed a similar learning method, GeoNet, a jointly unsupervised learning framework for monocular depth, optical flow, and ego-motion estimation from videos.

The above view-synthesis-based methods work under the photometric consistency assumption which is easily violated at the outdoor condition. To overcome this problem, Shen *et al*. [29] proposed to employ the epipolar geometry constraint. Although the epipolar geometry constraint can reduce the instability of the photometric error to some extent, the false-matched correspondences also introduce some errors. In this work, we also use the epipolar geometry constraint to improve the robustness. However, the constraint is not applied to the pairwise matching but to each point in the image. The correspondence of each point is calculated by projecting the point onto the other image based on the predicted depth and pose; in other words, the epipolar geometry constraint is implicitly embedded into the training process. Also, we introduce matching point constraints to guide the optimizer to converge to the correct solution. The coarse to fine idea is similar to [35] in which the sparse constraint enables the iteration to quickly move toward the neighborhood of the ground truth, which is further refined by the dense constraint. Additionally, most of the depth estimation works to employ multi-scales loss as the supervision, which is the sum of losses estimated at each scale. In the experiments, we find that single-scale loss is better than multi-scales loss because 'texture-copy' artifacts are exhibited in the intermediate lower resolution depth maps (details in the depth map incorrectly transferred from the color image). Therefore, the estimated photometric losses in the lower resolution depth maps are inaccurate that degrade the performance. In this work, we employ single-scale loss in the experiments.

III. FixUnDepth

A. Overview

Fig. 2 shows the pipeline of our FixUnDepth framework for depth and pose estimation. We use short image sequences of scenes captured by a moving camera to train our model. Given the image sequences include n frames and are denoted as $\langle I_1, I_2, \cdots, I_n\rangle$, we first segment every three consecutive images as a sample, and the step of sampling is one image, as described at the top of Fig. 2. Second, each sample is sequentially fed into the network where the input is three consecutive images, which is denoted as $\langle I_{t-1}, I_t, I_{t+1}\rangle$. The SURF features [30] at each frame are extracted and matched and the essential matrix between two consecutive frames is estimated by the matched correspondences from pre-processing. Our model is made up of two pose networks and one depth network. The pose network is a twin network of two pose networks that share parameters. We enforce this constraint because the networks are all used for pose estimation and they should not differ from each other. We employed the twin network that aims for better pose estimation by enforcing the pose consistency constraint. The depth map outputs the depth of each pixel in the frame. The pose network outputs the extrinsic parameters of the image. The output of each pose network is used to warp the source image into the target camera to synthesize a new image. The extrinsic parameters predicted by the first pose network are also employed to recover the essential matrix between the first two images according to the multi-view geometry knowledge. To better train our model, apart from using photometric reconstruction disparity loss (L_{pr}) and depth smoothness loss (L_{ds}) in our objective function, we also employed three other losses: positions of matched feature pairs loss (L_{pf}), epipolar geometry constraint loss (L_{eg}), and point cloud consistency loss (L_{pc}). L_{pr}, L_{pf} and L_{eg} are spatial losses which are illustrated by the dashed lines in Fig. 2. L_{pc} is the temporal loss which is illustrated by the dashed lines in Fig. 3. The architecture of our network is detailed in Section III.B.

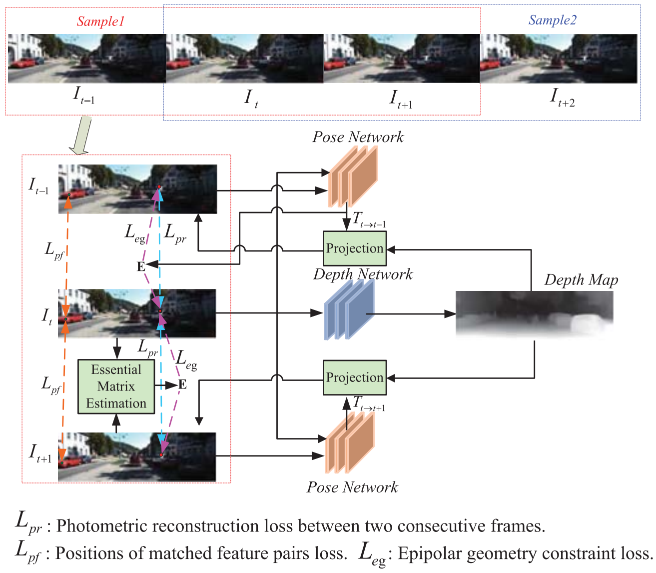


Fig. 2. Overview of FixUnDepth. The red points in I_{t-1}, I_{t+1} are the corresponding projected points of the red point in I_t, respectively. The green points in I_{t-1}, I_{t+1} are the matched features of the green point in I_t, respectively. Notice that two pose networks in the diagram share parameters.

B. Network Architecture

Following [23], [28], we employ fully convolutional architecture to model the depth network as an encoder-decoder

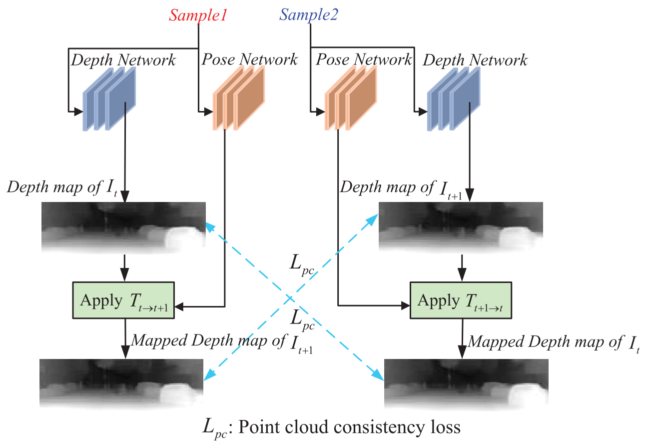


Fig. 3. Illustration of the temporal loss. The estimated point cloud from I_{t+1} and mapped point cloud, gained by transforming the estimated depth map of I_t, should be basically identical. The difference is measured by L_{pc}.

structure to generate dense depth maps. The encoder follows the basic structure of ResNet50. The decoder consists of deconvolution layers to enlarge the spatial feature maps to the same size as input. We use skip connections between encoder and decoder parts at different corresponding resolutions to fuse local detailed information. The depth is predicted at four different scales. The pose network predicts the camera rotation (represented in Euler angles) and translation by regressing the 6-DoF camera poses. The pose network is also modeled as encoder-decoder architecture. The encoder contains five convolutional layers which are followed by three convolutional layers and a global average pooling layer to predict the poses. The decoder consists of five deconvolution layers and four convolutional layers, and each deconvolution layer is followed by a convolutional layer except for the first deconvolution layer.

C. Training Loss

Zhou *et al.* [23] used the photometric loss for model training which commonly assumes that the brightness or gradient constancy constraint applies to the video sequence. In practice, this assumption is easily violated because the light is prone to change, especially in outdoor conditions. Comparatively, features are more robust, and most of the man-made features are free from light changes, such as SIFT [31] and ORB [32]. On the other hand, the update at each iteration during optimization depends on the difference of pixel intensity, which implies that this approach works only when the initialization of the pose is not far from the ground truth. Additionally, this loss fails to be applied in images that have little or non-texture where the gradient is not significant, which leads to early stop during optimization.

The loss of our approach is defined as a combination of five items, each item is controlled by a factor. Except for L_{pf} and L_{eg}, which are calculated at a single scale, all other loss functions can be formulated at a single scale and four different scales s, respectively. Each scale image is downsampled; therefore, the size of the last scale image is $\frac{1}{8}$ in width and height to the input.

$$L = \alpha L_{pr} + \beta L_{ds} + \lambda L_{pf} + \eta L_{eg} + \omega L_{pc}, \tag{1}$$

where L_{pr} represents the photometric reconstruction loss which constrains the warped image to appear similar to the corresponding training input; L_{ds} represents the depth smoothness loss; L_{pf} represents the positions of matched feature pairs loss which enforces the mapped features to coincide with the correspondences; L_{eg} represents the epipolar geometry constraint loss, and L_{pc} represents the point cloud consistency loss which imposes 3D geometric constraints. Next, we detail each component of our loss.

We define some symbols used in our losses. The input to the network is three consecutive frames, which are denoted as I_{t-1}, I_t and I_{t+1}, respectively. From the network we get two estimated ego-motions and the recovered depth map which are denoted as $T_{t\to t-1}$, $T_{t\to t+1}$ and D_t, respectively. The camera intrinsics matrix is denoted as K.

1) Photometric Reconstruction Loss: Similar to Zhou *et al.* [23], the photometric disparity error is employed as the loss to minimize and is defined as:

$$\begin{aligned} L_{pr} = &\sum_t \sum_i M_{t+1}(p_t(i)) \left| I_t(p_t(i)) - I_{t+1}(\hat{p}_{t+1}(i)) \right| \\ &+ \sum_t \sum_i M_{t-1}(p_t(i)) \left| I_t(p_t(i)) - I_{t-1}(\hat{p}_{t-1}(i)) \right|, \end{aligned} \tag{2}$$

where M_{t-1} and M_{t+1} represent the mask maps for images I_{t-1} and I_{t+1} as described below, respectively; $|.|$ denotes the L1 norm; $I_t(p_t(i))$ is the intensity of pixel i at time t; $I_{t+1}(\hat{p}_{t+1}(i)$ is the intensity of the mapped correspondence of pixel i at time $t+1$. We define the mapped point of p_t, $\hat{p}_{t+1}$, by projecting coordinates onto the view at time $t+1$ as

$$\hat{p}_{t+1} = K T_{t\to t+1} D_t(p_t) K^{-1} p_t. \tag{3}$$

$I_{t-1}(\hat{p}_{t-1}(i))$ is defined similarly.

Since occlusions and dynamic objects prevalently exist in realistic scenes, the prior works try to filter these erroneous regions by applying a per-pixel mask to the loss. However, since the mask is learned by a network, the inaccurate predicted mask can suffer from poor performance. In this work, similar to Godard *et al.* [38], the mask maps, M_{t-1} and M_{t+1}, are estimated as the binary which can filter pixels in a static camera, an object moving at equivalent relative translation to the camera, or a low texture region. We define M_{t-1} as follows, and M_{t+1} is defined similarly.

$$M_{t-1} = \begin{cases} 1, & |I_{t-1} - I_{t-1\to t}| < |I_{t-1} - I_t| \\ 0, & \text{otherwise,} \end{cases} \tag{4}$$

where I_{t-1} is the intensity of a pixel at time $t-1$; $I_{t-1\to t}$ is the intensity of the mapped correspondence of the pixel at time t, and I_t is the intensity of a pixel which has the same coordinates as the pixel at time t.

To better evaluate the quality of image predictions, we consider the structured similarity (SSIM) and define the final L_{pr}

as the combination of both L1 loss and SSIM loss.

$$\begin{aligned}L_{pr} &= \sum_t \left[(1-\rho)\sum_i \left[M_{t+1}(p_t(i))\left|I_t(p_t(i)) - I_{t+1}(\hat{p}_{t+1}(i))\right|\right]\right] \\ &\quad + \sum_t \rho \frac{1 - SSIM_{t\to t+1}}{2} \\ &\quad + \sum_t \left[(1-\rho)\sum_i \left[M_{t-1}(p_{t-1}(i))\left|I_t(p_t(i)) - I_{t-1}(\hat{p}_{t-1}(i))\right|\right]\right] \\ &\quad + \sum_t \rho \frac{1 - SSIM_{t\to t-1}}{2}. \end{aligned} \tag{5}$$

Following [22], [28], [29], [33], we set $\rho = 0.85$ in our work.

2) Depth Smoothness Loss: Since depth discontinuities often occur at image gradients, we use the edge-aware depth smoothness loss as usually done [22], [28], by penalizing the L1 norm of the depth gradients across adjacent pixels which are weighted by image gradients.

$$L_{ds} = \sum_t \sum_i \left|\nabla D_t\left(p_t(i)\right) \cdot \left(e^{-|\nabla I_t(p_t(i))|}\right)^T\right|, \tag{6}$$

where ∇ represents the 2D differential operator, and $p_t(i)$ is the pixel i at time t.

3) Positions of Matched Feature Pairs Loss: At the preprocessing, we extracted SURF features at each frame and matched them between two consecutive frames. The image coordinates of the matched correspondences i at frames t and $t+1$ are denoted as $f_t(i)$ and $f_{t+1}(i)$, respectively. Then, the position disparity error between frames t and $t+1$ is defined as

$$L_{t+1}^{pf} = \sum_t \sum_i M_{t+1}\left|f_{t+1}(i) - \hat{f}_{t+1}(i)\right|, \tag{7}$$

where $\hat{f}_{t+1}(i)$ is the mapped feature of $f_t(i)$ according to (3), and $|.|$ denotes the L1 norm.

Similarly, the position disparity error between frames t and $t-1$ is obtained L_{t-1}^{pf}. The final position loss is defined as $L_{pf} = L_{t+1}^{pf} + L_{t-1}^{pf}$.

4) Epipolar Geometry Constraint Loss: The epipolar geometry constraint loss is defined as

$$\begin{aligned}L_{eg} &= \sum_t \sum_i M_{t+1}(p_t(i))\left|p_t(i)^T K^T E_{t\to t+1} K \hat{p}_{t+1}(i)\right| \\ &\quad + \sum_t \sum_i M_{t-1}(p_t(i))\left|p_t(i)^T K^T E_{t\to t-1} K \hat{p}_{t-1}(i)\right|, \end{aligned} \tag{8}$$

where K is the camera intrinsics matrix; $E_{t\to t+1}$ is the estimated essential matrix derived from the predicted pose between frames t and $t+1$; $E_{t\to t-1}$ is the estimated essential matrix derived from the predicted pose between frames t and $t-1$ and defined as follows according to [44].

$$E_{t\to t-1} = \left[\mathbf{t}_{t\to t-1}\right]_\times \mathbf{R}_{t\to t-1}, \tag{9}$$

where $\mathbf{R}_{t\to t-1}$ and $\mathbf{t}_{t\to t-1}$ are the predicted rotation and translation from $T_{t\to t-1}$, respectively, and $\left[\mathbf{t}_{t\to t-1}\right]_\times$ is the corresponding skew-symmetric matrix of $\mathbf{t}_{t\to t-1} = [t_1, t_2, t_3]$ and defined as

$$\left[\mathbf{t}_{t\to t-1}\right]_\times = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}. \tag{10}$$

$\hat{p}_{t+1}(i)$ denotes the mapped correspondence of point $p_t(i)$ according to (3). $E_{t-1\to t}$ and $\hat{p}_{t-1}(i)$ are defined similarly, but $E_{t-1\to t}$ is obtained by a perspective-n-point (PNP) approach which uses the matched pairs as inputs. We define $\hat{p}_{t+1}$ as

$$\hat{p}_{t+1} = K\left(T_{t+1\to t}\right)^{-1} D_t(p_t) K^{-1} p_t, \tag{11}$$

where $D_t(p_t)$ is the predicted depth of point p_t. Notice that the transform matrix is the inverse of $T_{t+1\to t}$ which implicitly constrains the ego-motion and the opposite of two consecutive frames should be basically inverse.

5) Point Cloud Consistency Loss: Similarly, we enforce 3D geometric consistency in the loss based on an intuition that the estimated point cloud from sample $t+1$ and mapped point cloud, gained by transforming the point cloud estimated from sample t, should be basically identical if the estimated depth is correct. We formulate this constraint as

$$\begin{aligned}L_{pc} &= \sum_t \sum_i (E_{t+1} + E_{t-1}), \\ E_{t+1} &= M_t\left(p_t(i)\right)\Big|T_{t\to t+1} D_t\left(p_t(i)\right) K^{-1} p_t(i) \\ &\qquad - D_{t+1}\left(\hat{p}_{t+1}(i)\right) K^{-1} \hat{p}_{t+1}(i)\Big|, \\ E_{t-1} &= M_t\left(p_t(i)\right)\Big|T_{t\to t-1} D_t\left(p_t(i)\right) K^{-1} p_t(i) \\ &\qquad - D_{t-1}\left(\hat{p}_{t-1}(i)\right) K^{-1} \hat{p}_{t-1}(i)\Big|, \end{aligned} \tag{12}$$

where M represents the mask, and $\hat{p}_{t+1}(i)$ and $\hat{p}_{t-1}(i)$ are defined as in (3).

IV. Experiments

A. Experimental Details

We use the publicly available TensorFlow framework to implement our code and evaluate the performance of our system on the KITTI [34], Cityscapes [37] and NYUv2 [45] datasets.

1) Datasets: We employ the KITTI dataset as the main dataset for training and evaluation. The KITTI dataset is the commonly used benchmark for evaluating depth and ego-motion accuracy. The KITTI dataset includes evaluation benchmarks for several computer vision and robotic tasks such as stereo, optical flow, visual odometry, SLAM, 3D object detection, and 3D object tracking. We only use its raw form in our experiments. The raw dataset is divided into the categories 'Road', 'City', 'Residential', 'Campus', and 'Person'. For each sequence, the raw data, object annotations, and a calibration file are provided. The dataset contains 42,382 rectified stereo pairs; the resolution of a typical image is 1242 × 375.

The Cityscapes dataset is comprised of a large, diverse set of stereo video sequences recorded in streets from 50 different cities. Five thousand of these images have high-quality pixel-level annotations; 20,000 additional images have coarse annotations to enable methods that leverage large volumes of weakly-labeled data. This dataset brings higher resolution, image quality, and variety compared to KITTI, while having a similar setting.

NYUv2 dataset is one of the largest RGB-D datasets for indoor scene reconstruction. It contains a raw dataset, which includes RGB and depth video sequences belonging to 464 scenes, and a fine dataset containing 1449 densely labeled RGB and depth pairs. In accordance with the official setting, 795 images of the total 1449 images of the fine dataset are used as training data, and 654 images are used for testing.

2) Training Details: Similar to Godard *et al.* [38], we use data augmentation to expand the training set. The inputs are first with horizontal flips, and are then randomly training augmentations with 50% probability: random brightness, contrast, saturation, and hue jitter with respective ranges of $\pm 0.2, \pm 0.2, \pm 0.2$, and ± 0.1. The mini-batch size and the sequence lengths are set to be 4 and 3, respectively. We train the network using Adam optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$. The learning rate is initially set to be 0.0002 and decreases to 1/10 time after the 15th epoch. The whole training process takes around 18 epochs to converge. The loss weights are set as $\alpha = 1.0, \beta = 0.1, \lambda = 0.8, \eta = 0.1, \omega = 0.2$.

3) Evaluation Metrics: We employ the metrics proposed in [10], [21] to evaluate our depth predictions. The employed metrics are defined as follows: D and D^{gt} are denoted as the depth predictions and the ground truth, respectively, and N represents the number of valid pixels in the ground truth.

Absolute Relative Difference:

$$Abs\ Rel = \frac{1}{N}\sum_{i=1}^{N}\left|D - D^{gt}\right|/D^{gt}.$$

Squared Relative Difference:

$$Sq\ Rel = \frac{1}{N}\sum_{i=1}^{N}\left(D - D^{gt}\right)^2/D^{gt}.$$

Root Mean Squared Error:

$$RMSE=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(D - D^{gt})^2}.$$

Root Mean Squared Error in Log Space:

$$RMSE\ log=\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\log(D) - \log(D^{gt}))^2}.$$

Absolute Difference in Log Space:

$$log10 = \frac{1}{N}\sum_{i=1}^{N}\left|\log(D) - \log\left(D^{gt}\right)\right|.$$

Accuracy with Threshold:

$$\%\text{ of } D \text{ s.t.}\max\left(\frac{D}{D^{gt}}, \frac{D^{gt}}{D}\right) = \delta < thr.$$

TABLE I

ABLATION STUDY ON OUR APPROACH USING SINGLE SCALE LOSS

Method	Error (lower is better)				Accuracy (higher is better)		
	Abs Rel	*Sq Rel*	*RMSE*	*RMSE log*	$\delta <$ 1.25	$\delta <$ 1.25^2	$\delta <$ 1.25^3
Basic	0.141	1.064	5.349	0.2176	0.824	0.942	0.9758
Basic+pf	0.140	1.046	5.304	0.2151	0.825	0.942	0.9760
Basic+pf+pc	0.138	1.027	5.226	0.2139	0.826	0.942	0.9769
Basic+pf+pc+ms	0.135	0.987	5.209	0.2132	0.829	**0.943**	**0.9772**
Basic+pf+pc+ms+eg	**0.134**	**0.979**	**5.169**	**0.2124**	**0.832**	**0.943**	0.9769

TABLE II

ABLATION STUDY ON OUR APPROACH USING MULTI-SCALES LOSS

Method	Error (lower is better)				Accuracy (higher is better)		
	Abs Rel	*Sq Rel*	*RMSE*	*RMSE log*	$\delta <$ 1.25	$\delta <$ 1.25^2	$\delta <$ 1.25^3
Basic	0.147	1.305	5.535	0.2261	0.809	0.936	0.9743
Basic+pf	0.145	1.090	5.445	0.2167	0.810	0.938	0.9776
Basic+pf+pc	0.143	1.056	5.435	0.2165	0.811	0.938	0.9780
Basic+pf+pc+ms	0.140	1.025	5.442	0.2162	0.811	0.939	0.9780
Basic+pf+pc+ms+eg	**0.139**	**1.006**	**5.440**	**0.2154**	**0.812**	**0.940**	**0.9789**

TABLE III

EXPERIMENTAL RESULTS ON DIFFERENT RESOLUTIONS

Resolution	Error (lower is better)				Accuracy (higher is better)		
	Abs Rel	*Sq Rel*	*RMSE*	*RMSE log*	$\delta <$ 1.25	$\delta <$ 1.25^2	$\delta <$ 1.25^3
416X128	0.134	0.979	5.169	0.2124	0.832	0.944	0.976
832X256	**0.129**	**0.976**	**4.958**	**0.2035**	**0.848**	**0.951**	**0.979**

TABLE IV

QUANTITATIVE COMPARISON RESULTS WITH/WITHOUT OUR LOSSES WHILE VGG NET IS EMPLOYED AS THE BACKBONE

Method	Error (lower is better)			Accuracy (higher is better)		
	Abs Rel	*Log10*	*RMSE*	$\delta <$ 1.25	$\delta <$ 1.25^2	$\delta <$ 1.25^3
VGG-Net	0.142	0.062	5.46	0.819	0.936	0.973
VGG-Net + Ours	**0.140**	**0.061**	**5.40**	**0.821**	**0.938**	**0.975**

B. KITTI Ablation Study

In this section, the effectiveness of positions of matched feature pairs loss (pf), threshold masks (ms), point cloud consistency loss (pc), and epipolar geometry constraint loss (eg) are validated by ablation study. The results have been verified on single scale loss, multi-scales loss, and different resolutions, respectively.

Prior works use multi-scale depth prediction to relieve the gradient locality of the bilinear sampler and to prevent the training objective of getting stuck in local minima. However, this operation exhibits 'texture-copy' artifacts in the inferred disparities. In this work, we investigate two kinds of loss established approaches:

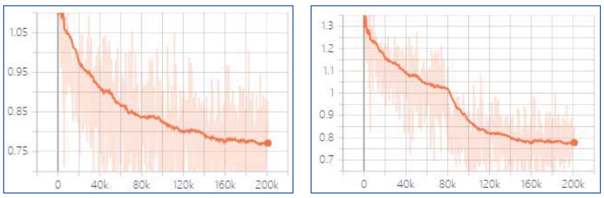
Single scale loss: predicting the depth map at the input resolution and estimating the photometric error as the supervision signal.

Multi-scales loss: predicting the depth maps at multiple resolutions and calculating the loss. The combination of the individual losses is served as the total loss.

TABLE V

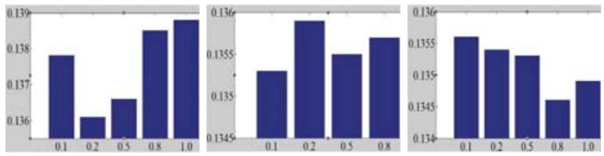
COMPARISON RESULTS ON THE KITTI AND CITYSCAPES DATASETS. K DENOTES THE TRAINING AND TEST ON THE KITTI DATASET. C DENOTES THE TRAINING AND TEST ON THE CITYSCAPES DATASET. (D) REPRESENTS SUPERVISED APPROACHES. (S) REPRESENTS STEREO DATA AS INPUT, AND (M) REPRESENTS MONOCULAR DATA AS INPUT. THE BEST RESULT IS MARKED IN BOLD. NOTE: FOR FAIR COMPARISONS, THE MONO2 [38] APPROACH WITHOUT PRETRAINING IN THE IMAGENET DATASET WAS USED

Method	Dataset	Error (lower is better)					Accuracy (higher is better)		
		Abs Rel	*Sq Rel*	*RMSE*	*RMSE log*	*log10*	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Eigen et al. [10]	K(D)	0.203	1.548	6.307	0.282	-	0.702	0.890	0.958
Liu et al. [11]	K(D)	0.202	1.614	6.523	0.275	-	0.678	0.895	0.965
Godard et al. [22]	K(S)	0.148	1.344	5.927	0.247	-	0.803	0.922	0.964
Zhan et al. [25]	K(S)	0.144	1.391	5.869	0.241	-	0.803	0.928	0.969
Han Yan et al. [42]	K(D)	0.134	-	4.72	-	0.057	0.829	0.950	0.98
Y Cao et al. [40]	K(D)	0.180	-	6.31	-	0.072	0.771	0.917	0.966
Y Cao et al. [41]	K(D)	0.142	-	5.06	-	0.058	0.829	0.943	0.982
Gan et al. [47]	K(D)	**0.098**	**0.666**	**3.933**	**0.173**	-	**0.890**	**0.964**	**0.985**
Zhou et al. [23]	K(M)	0.183	1.595	6.709	0.270	-	0.734	0.902	0.959
Mahjourian et al. [26]	K(M)	0.163	1.240	6.220	0.250	-	0.762	0.916	0.968
Geonet [28]	K(M)	0.155	1.296	5.857	0.233	-	0.793	0.931	0.973
Matchvo [29]	K(M)	0.156	1.309	5.730	0.236	-	0.797	0.929	0.969
Bian et al. [33]	K(M)	0.137	1.089	5.439	0.217	-	0.830	0.942	0.975
Mono2 [38] w/o pt	K(M)	0.132	1.044	5.142	0.210	-	0.845	0.948	0.977
Ours	K(M)	**0.129**	**0.976**	**4.958**	**0.203**	**0.056**	**0.848**	**0.951**	**0.979**
Zhou et al. [23]	K+C(M)	0.198	1.836	6.565	0.275	-	0.718	0.901	0.960
Godard et al. [22]	K+C(S)	0.124	1.076	5.311	0.219	-	0.847	0.942	0.973
Geonet [28]	K+C(M)	0.153	1.328	5.737	0.232	-	0.802	0.934	0.972
Matchvo [29]	K+C(M)	0.152	1.205	5.564	0.227	-	0.800	0.935	0.973
Bian et al. [33]	K+C(M)	0.128	1.047	5.234	0.208	-	0.846	0.947	0.976
Ours	K+C(M)	**0.118**	**0.909**	**4.816**	**0.195**	**0.051**	**0.876**	**0.955**	**0.980**



(a) (b)

Fig. 4. Comparison of the convergence curves of our approach with the base approach during model training. (a) The convergence curve of the base approach. (b) The convergence curve of our approach.



(a) (b) (c)

Fig. 5. The absolute relative difference while λ, η and ω are set as different values. (a) The reconstructed absolute relative difference when loss includes L_{pr}, L_{ds} and L_{pc}, and ω is set as 0.1, 0.2, 0.5, 0.8 and 1.0, respectively. (b) The reconstructed absolute relative difference when loss includes L_{pr}, L_{ds}, L_{pc} and L_{eg}, and η is set as different factors. (c) The reconstructed absolute relative difference when loss includes $L_{pr}, L_{ds}, L_{pc}, L_{eg}$ and L_{pf}, and λ is set as different factors.

We evaluate the effectiveness of different modules on these proposed methods. Tables I and II show the comparative results on the resolution of 416×218. The basic approach only used the photometric reconstruction disparity loss and the depth smoothness loss. We notice that each loss function can improve depth estimation accuracy. From tables I and II, we observe that our approach using single scale loss is better. We think the gain by employing single scale loss may be from the loss estimated at the input resolution in which the 'texture-copy' artifacts are prohibited. Table III shows evaluation metrics on different image resolutions. We notice that a higher resolution can achieve better performance.

We report the convergence of the proposed method as compared to the base approach in Fig. 4. From Fig. 4, we notice that our approach is more stable during the optimization process which may be caused by the additional constraints. During iteration, we remove the positions of matched feature pairs loss after the 80 thousandth step, which causes an inflection point in Fig. 4. Since the loss is almost not changed after 15 epochs, we stop the training at this time step.

We also conducted experiments to determine the most effective combination of different weights. Since the photometric reconstruction loss and the depth smoothness loss are employed by most state-of-the-art, we set their weights as $\alpha = 1.0$ and $\beta = 1.0$ according to the suggested set, respectively. During the experiments, these two weights are fixed. Fig. 5 shows the absolute relative difference while λ, η and ω are set as different values. Since the sample space of these weights is huge, to ease the experiments, we only consider five different factors: 0.1, 0.2, 0.5, 0.8 and 1.0. To vividly demonstrate the effect of different factors, the results are depicted as three histograms where lower is better. The first chart in Fig. 5 shows how the point cloud consistency loss affects the reconstruction results while $\alpha = 1.0, \beta = 0.1$, and ω is set as different values. We observe that point cloud consistency loss gets the maximum effect while ω is set as 0.2. The second chart in Fig. 5 shows the effect of epipolar geometry constraint loss while α, β and ω are set as $\alpha = 1.0$, $\beta = 0.1$ and $\omega = 0.2$, respectively. We notice that the best value of η is set as 0.1. The third chart in Fig. 5 depicts the experimental results while $\alpha = 1.0, \beta = 0.1, \eta = 0.1$ and $\omega = 0.2$, and λ is set as different factors. The optimal value of

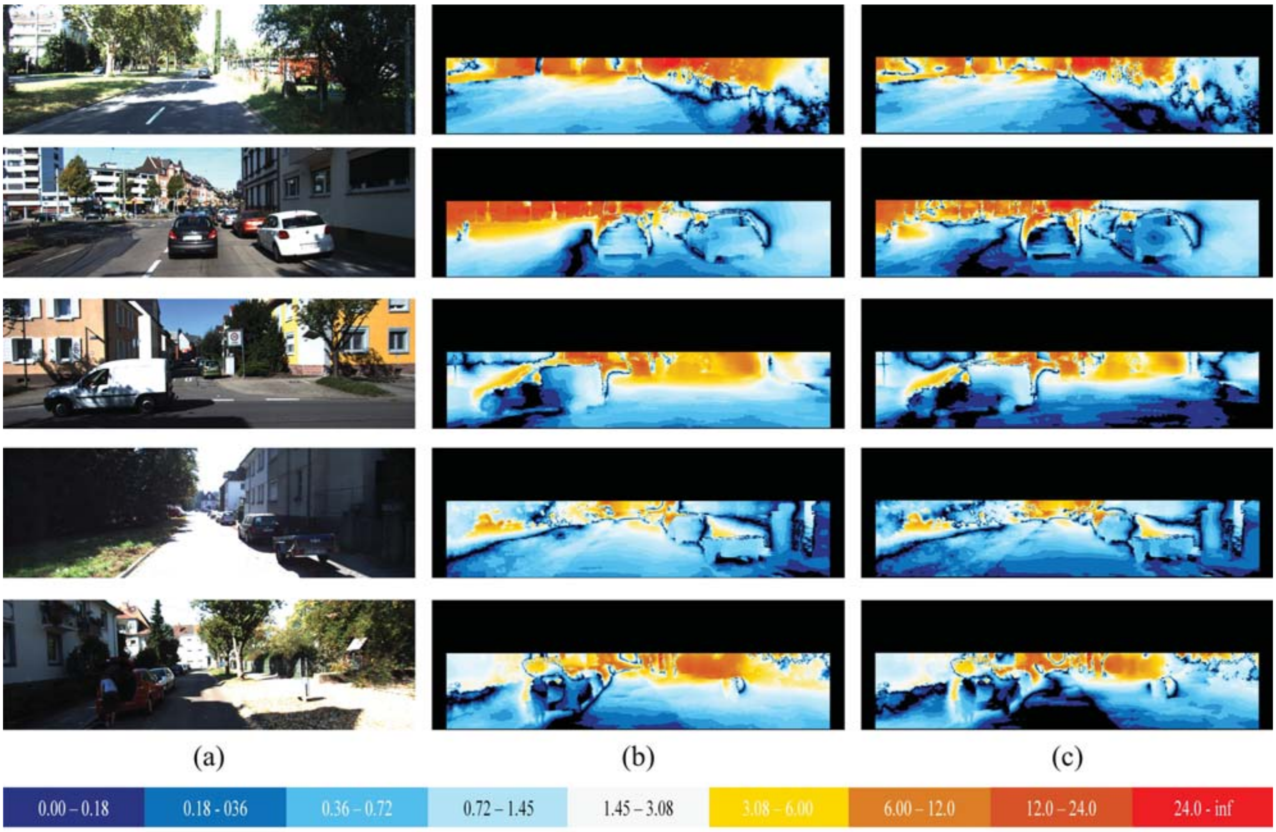


Fig. 6. Qualitative comparison results with or without the epipolar geometry constraint loss. (a) The input image. (b) The errors with respect to ground truth without the epipolar geometry constraint loss. (c) The errors with respect to ground truth with the epipolar geometry constraint loss. The last line reports the color code used to display the seriousness of the shortcomings.

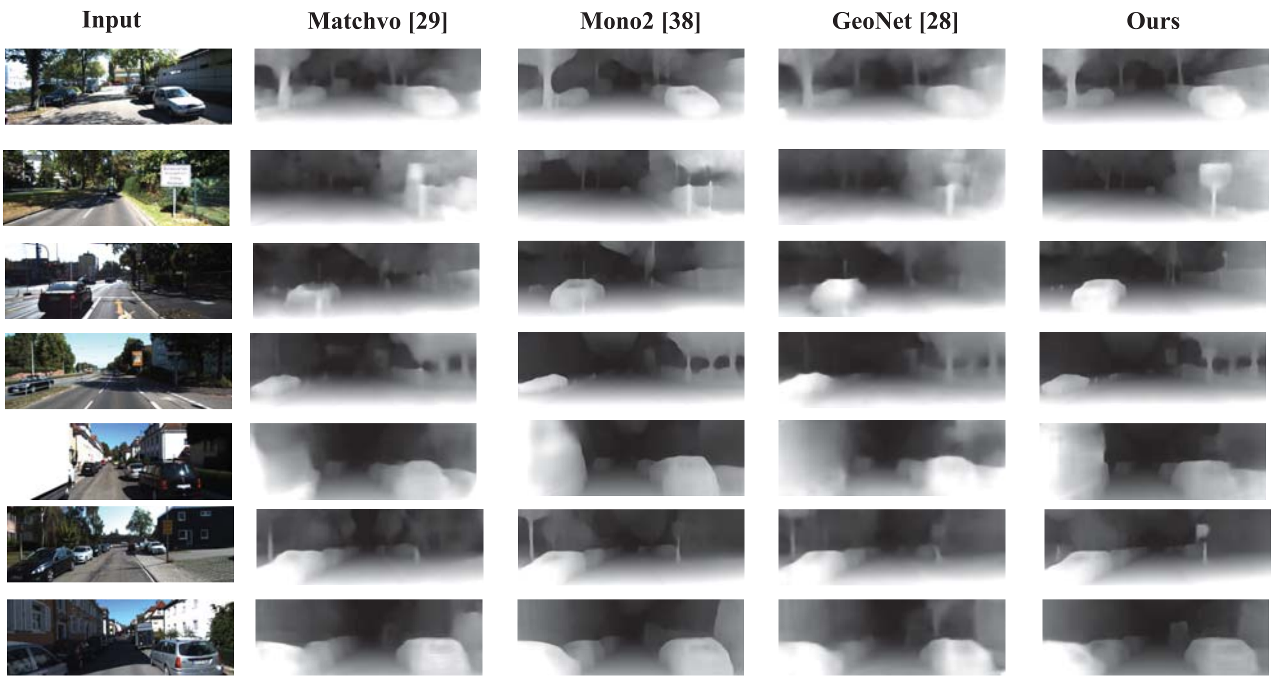


Fig. 7. Comparison of single-view depth estimation between other approaches and ours.

TABLE VI

COMPARISON RESULTS ON THE NYUv2 DATASET. N DENOTES THE TRAINING AND TEST ON THE NYUv2 DATASET. (D) REPRESENTS SUPERVISED APPROACHES, AND (M) REPRESENTS MONOCULAR DATA AS INPUT. THE BEST RESULT IS MARKED IN BOLD

Method	Dataset	Error (lower is better)			Accuracy (higher is better)		
		Abs Rel	*Sq Rel*	*RMSE*	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Liu et al. [11]	N(D)	0.230	0.095	0.824	0.614	0.883	0.971
Eigen et al. [10]	N(D)	0.215	-	0.907	0.611	0.887	0.971
Semi [46]	N(D)	0.183	0.077	0.704	0.713	0.931	0.984
Chen et al. [49]	N(D)	0.144	-	0.518	0.815	0.960	0.989
Yin et al. [50]	N(D)	0.134	-	0.485	0.829	0.956	0.980
Fu et al. [48]	N(D)	**0.115**	**0.051**	**0.509**	**0.828**	**0.965**	**0.992**
Zhou et al. [23]	N(M)	0.208	0.086	0.712	0.674	0.900	0.968
Ours	N(M)	**0.165**	**0.069**	**0.587**	**0.765**	**0.936**	**0.981**

λ is set as 0.8. Therefore, we know that the combination of the last three parameters in Eq. (1) gets the maximum effect while they are set as $\lambda = 0.8$, $\eta = 0.1$ and $\omega = 0.2$, respectively.

Fig. 6 shows qualitative comparison results for some of the scenes in KITTI with or without the epipolar geometry constraint loss. As evidenced by the error images, with the epipolar geometry constraint loss, our approach is able to recover the correct depth and produce the minimal error.

To demonstrate the effect of our approach, we conducted experiments on the KITTI dataset in which VGG net is employed as the backbone. Table IV shows the quantitative comparison results with or without our losses. We notice that the performance can be improved where our proposed losses are concerned.

C. Comparisons With the State-of-the-Art

1) Depth Estimation: For fair comparisons with other approaches, we only use the raw form of KITTI in our experiments. Similar to Zhou *et al.* [23], we use the train/validation split provided by Eigen and Fergus [36]. Each sample includes three consecutive frames where the second frame is the target image and the other frames are the source image. The total number of samples is 44,540 where 40,109 were used for training and 4,431 for validation. We experimented on two resolutions of input and conclude that higher resolution is better.

2) Depth Results on KITTI and Cityscapes Datasets: Table V shows the quantitative comparison results on the KITTI and Cityscapes datasets. Our approach outperformed all other approaches except the approach proposed in [47], but this approach is a supervised approach that can take advantage of the ground truth. A qualitative evaluation is shown in Fig. 7. Our approach can preserve the depth boundaries, especially in thin structures, as shown in the 2^{nd}, 4^{th}, and 6^{th} rows where the boundary of the reconstructed small sign in our approach is clear while it is relatively obscure in other approaches. Since most approaches are designed for multiple tasks, our approach is more competitive in runtime.

3) Depth Results on NYUv2 Dataset: We train our model on the NYUv2 dataset, and compare our approach against several prior works. Table VI reports the quantitative comparison results, from which we can see that the performance of our approach is inferior to some supervised approaches. The reason is two-fold: 1) the large scale of rotation and translation of the camera are not suitable for self-supervised depth estimation approaches; 2) the supervised approaches can take advantage of the ground truth. However, our approach still performs better than some supervised approaches and the self-supervised approach proposed in [23]. Some qualitative results are illustrated in Fig. 8. Our approach yields visually reasonable results.

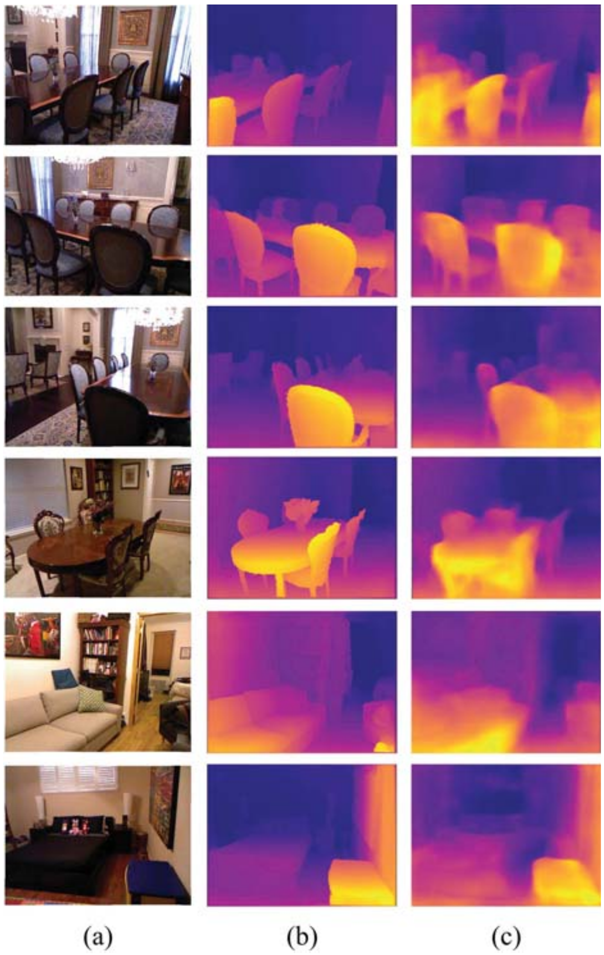


Fig. 8. Qualitative results on NYUv2 dataset. (a) Image; (b) Ground truth; (c) Ours.

V. CONCLUSION AND FUTURE WORK

In this paper, we analyzed the limitation of self-supervised depth estimation based on only photometric loss. To overcome this limitation, we employed the positions of matched feature pairs loss as the geometry prior to guide the optimizer to move toward the neighborhood of the ground truth and exploit the point cloud consistency constraint to eliminate ambiguity. Additionally, we employed epipolar constraints to eliminate the ambiguity of pose estimation. As a result, the precision of depth prediction increased. We also investigated the problem of 'texture-copy' artifacts and proposed two different solutions. The experimental results on the KITTI, Cityscapes and NYUv2 datasets show that our approach performed better than other approaches. In the future, we intend to exploit the temporal information by using long video sequences as input and use the RNN network to encode the relation.

REFERENCES

[1] K. Konolige and M. Agrawal, "FrameSLAM: From bundle adjustment to real-time visual mapping," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1066–1077, Oct. 2008.

[2] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4104–4113.

[3] P. Moulon, P. Monasse, and R. Marlet, "Global fusion of relative motions for robust, accurate and scalable structure from motion," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 3248–3255.

[4] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015.

[5] R. Ranftl, V. Vineet, Q. Chen, and V. Koltun, "Dense monocular depth estimation in complex dynamic scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4058–4066.

[6] B. Triggs, P. F. McLauchlan, R. Hartley, and A. W. Fitzgibbon, "Bundle adjustment—A modern synthesis," in *Proc. Int. Workshop Vis. Algorithms*, 1999, pp. 298–372.

[7] J. Engel, J. Sturm, and D. Cremers, "Semi-dense visual odometry for a monocular camera," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1449–1456.

[8] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: LargeScale direct monocular SLAM," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2014, pp. 834–849.

[9] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 611–625, Mar. 2018.

[10] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2014, pp. 1–9.

[11] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2024–2039, Oct. 2016.

[12] B. Li, C. Shen, Y. Dai, A. van den Hengel, and M. He, "Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1119–1127.

[13] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 239–248.

[14] J. Flynn, I. Neulander, J. Philbin, and N. Snavely, "Deep stereo: Learning to predict new views from the world's imagery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5515–5524.

[15] A. Kendall *et al.*, "End-to-end learning of geometry and context for deep stereo regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 66–75.

[16] N. Mayer *et al.*, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4040–4048.

[17] B. Ummenhofer *et al.*, "DeMoN: Depth and motion network for learning monocular stereo," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5622–5631.

[18] H. Zhou, B. Ummenhofer, and T. Brox, "DeepTAM: Deep tracking and mapping," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 851–868.

[19] C. Wang, J. M. Buenaposada, R. Zhu, and S. Lucey, "Learning depth from monocular videos using direct methods," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2022–2030.

[20] Y. Zhong, Y. Dai, and H. Li, "Self-supervised learning for stereo matching with self-improving ability," 2017, *arXiv:1709.00930*. [Online]. Available: http://arxiv.org/abs/1709.00930

[21] R. Garg, V. K. Bg, G. Carneiro, and I. Reid, "Unsupervised CNN for single view depth estimation: Geometry to the rescue," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2016, pp. 740–756.

[22] C. Godard, O. M. Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6602–6611.

[23] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6612–6621.

[24] R. Li, S. Wang, Z. Long, and D. Gu, "UnDeepVO: Monocular visual odometry through unsupervised deep learning," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2018, pp. 7286–7291.

[25] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. M. Reid, "Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 340–349.

[26] R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised learning of depth and ego-motion from monocular video using 3D geometric constraints," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5667–5675.

[27] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki, "SfM-Net: Learning of structure and motion from video," 2017, *arXiv:1704.07804*. [Online]. Available: http://arxiv.org/abs/1704.07804

[28] Z. Yin and J. Shi, "GeoNet: Unsupervised learning of dense depth, optical flow and camera pose," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1983–1992.

[29] T. Shen *et al.*, "Beyond photometric loss for self-supervised ego-motion estimation," in *Proc. Int. Conf. Robot. Automat. (ICRA)*, May 2019, pp. 6359–6365.

[30] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2016, pp. 404–417.

[31] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[32] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2564–2571.

[33] J. Bian *et al.*, "Unsupervised scale-consistent depth and ego-motion learning from monocular video," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2019, pp. 1–11.

[34] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013.

[35] S. Chen, L. Liang, W. Liang, and H. Foroosh, "3D pose tracking with multi-template warping and SIFT correspondences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 11, pp. 2043–2055, Nov. 2016.

[36] D. Eigen and R. Fergus, "Predicting depth, surface normal and semantic labels with a common multi-scale convolutional architecture," in *Proc. Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2650–2658.

[37] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[38] C. Godard, O. M. Aodha, M. Firman, and G. Brostow, "Digging into self-supervised monocular depth estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3827–3837.

[39] J. Zhang, Y. Cao, Z.-J. Zha, Z. Zheng, C. W. Chen, and Z. Wang, "A unified scheme for super-resolution and depth estimation from asymmetric stereoscopic video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 3, pp. 479–493, Mar. 2016.

[40] Y. Cao, Z. Wu, and C. Shen, "Estimating depth from monocular images as classification using deep fully convolutional residual networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 11, pp. 3174–3182, Nov. 2018.

[41] Y. Cao, T. Zhao, K. Xian, C. Shen, Z. Cao, and S. Xu, "Monocular depth estimation with augmented ordinal depth relationships," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 8, pp. 2674–2682, Aug. 2020.

[42] H. Yan, X. Yu, Y. Zhang, S. Zhang, X. Zhao, and L. Zhang, "Single image depth estimation with normal guided scale invariant deep convolutional fields," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 1, pp. 80–92, Jan. 2019.

[43] H. Mohaghegh, N. Karimi, S. M. R. Soroushmehr, S. Samavi, and K. Najarian, "Aggregation of rich depth-aware features in a modified stacked generalization model for single image depth estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 3, pp. 683–697, Mar. 2019.

[44] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2004, p. 257.

[45] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 746–760.

[46] R. Ji *et al.*, "Semi-supervised adversarial monocular depth estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2410–2422, Oct. 2020.

[47] Y. Gan, X. Xu, W. Sun, and L. Lin, "Monocular depth estimation with affinity, vertical pooling, and label enhancement," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 232–247.

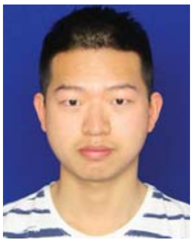
[48] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2002–2011.

[49] Y. Chen, H. Zhao, and Z. Hu, "Attention-based context aggregation network for monocular depth estimation," 2019, *arXiv:1901.10137*. [Online]. Available: http://arxiv.org/abs/1901.10137

[50] W. Yin, Y. Liu, C. Shen, and Y. Yan, "Enforcing geometric constraints of virtual normal for depth prediction," in *Proc. Int. Conf. Comput. Vis.*, Dec. 2019, pp. 5683–5692.

Shu Chen received the Ph.D. degree from Central South University, China. He is currently an Associate Professor with the School of Computer Science, Xiangtan University, China, and the School of Cyberspace Security, Xiangtan University. His research interests include computer vision, human motion tracking, and 3D scene reconstruction.

Zhengdong Pu is currently pursuing the M.S. degree in computer application technology with Xiangtan University, China. His research interests include 3D depth reconstruction and scene reconstruction.



Xiang Fan is currently pursuing the M.S. degree in computer application technology with Xiangtan University, China. His research interests include 3D depth reconstruction and scene reconstruction.

Beiji Zou received the B.S. degree in computer science from Zhejiang University, China, in 1982, the M.S. degree from Tsinghua University, specializing in CAD and computer graphics, in 1984, and the Ph.D. degree in control theory and control engineering from Hunan University, in 2001. He is currently a Professor with the School of Computer Science and Engineering, Central South University, China. His research interests include computer graphics, CAD technology, and image processing.