Auto-Rectify Network for Unsupervised Indoor Depth Estimation

Jia-Wang Bian, Huangying Zhan, Naiyan Wang, Tat-Jun Chin, Chunhua Shen, Ian Reid

Abstract—Single-View depth estimation using the CNNs trained from unlabelled videos has shown significant promise. However, excellent results have mostly been obtained in street-scene driving scenarios, and such methods often fail in other settings, particularly indoor videos taken by handheld devices. In this work, we establish that the complex ego-motions exhibited in handheld settings are a critical obstacle for learning depth. Our fundamental analysis suggests that the rotation behaves as noise during training, as opposed to the translation (baseline) which provides supervision signals. To address the challenge, we propose a data pre-processing method that rectifies training images by removing their relative rotations for effective learning. The significantly improved performance validates our motivation. Towards end-to-end learning without requiring pre-processing, we propose an Auto-Rectify Network with novel loss functions, which can automatically learn to rectify images during training. Consequently, our results outperform the previous unsupervised SOTA method by a large margin on the challenging NYUv2 dataset. We also demonstrate the generalization of our trained model in ScanNet and Make3D, and the universality of our proposed learning method on 7-Scenes and KITTI datasets.

Index Terms—Single-View Depth Estimation, Unsupervised Learning, Image Rectification

1 Introduction

Inferring 3D geometry from 2D images is a long-standing problem in robotics and computer vision. Depending on the specific use case, it is usually solved by Structure-from-Motion [1] or Visual SLAM [2]–[4]. Underpinning these traditional pipelines is searching for correspondences [5], [6] across multiple images and triangulating them via epipolar geometry [7]–[9] to obtain 3D points. Following the growth of deep learning-based approaches, Eigen et al. [10] show that the depth map can be inferred from a single color image by a CNN, which is trained with the ground-truth depth supervisions captured by range sensors. Subsequently a series of supervised methods [11]–[17] have been proposed and the accuracy of estimated depth is progressively improved.

Based on epipolar geometry [7]–[9], learning depth without requiring the ground-truth supervision has been explored. Garg et al. [18] showed that the single-view depth CNN can be trained from stereo image pairs with known baseline via photometric loss. Zhou et al. [19] further explored the unsupervised framework and proposed to train the depth CNN from unlabelled videos. They additionally introduced a Pose CNN to estimate the relative camera pose between consecutive frames, and the photometric loss remains as the main supervision signal. Following that, a number of of unsupervised methods have been proposed, which can be categorised into stereo-based [20]–[23] and video-based [24]–[32], according to the type of training data. Our work follows the latter paradigm, since unlabelled videos are easier to obtain in real-world scenarios.

Unsupervised methods have shown promising results in driving scenes, *e.g.*, KITTI [33] and Cityscapes [34]. However, as reported in [35], they usually fail in generic scenarios such as the indoor scenes in NYUv2 dataset [36]. For example, GeoNet [26], one of the state-of-the-art methods in KITTI, is unable to obtain reasonable results in NYUv2. To this end, [35] proposes to use optical flow as the supervision signal to train the depth CNN, and [37] also uses flow with epipolar geometry to estimate ego-motion to replace the Pose CNN. However, the reported depth accuracy [37] is still limited, *i.e.*, 0.189 in terms of *AbsRel*—see also qualitative results in Fig. 5.

Our work investigates the fundamental reasons behind poor results of unsupervised depth learning in indoor scenes. In addition to the usual challenges such as non-Lambertian surfaces and low-texture scenes, we identify the camera motion profile in the training videos as a critical factor that affects the training process. To develop this insight, we conduct an in-depth analysis of the effects of camera pose to current unsupervised depth learning framework. Our analysis shows that (i) fundamentally the camera rotation behaves as noise to training, while the translation contributes effective gradients; (ii) the rotation component dominates the motions in indoor videos captured using handheld cameras, while the opposite is true in autonomous driving scenarios.

Inspired by image rectification [38] which has been used a standard pre-processing for the stereo matching task [39], we propose to weakly rectify video frames for the effective depth learning. To be specific, we compute the relative rotation between image pairs and warp them to their common image plane, which results in rectified pairs that have no relative rotational motions, and we use these pairs for unsupervised depth learning. This is achieved by leveraging the traditional feature matching [5], [6] and epipolar geometry [8], [40], while no ground truth data is required. With our proposed data pre-processing, we demonstrate that existing state-of-the-art (SOTA) unsupervised methods [29], [32] can be trained well in the challenging indoor NYUv2 dataset [36]. Consequently, our results outperform the unsupervised SOTA [37] by a large margin.

Furthermore, towards end-to-end learning without requiring a manual pre-processing, we propose an Auto-Rectify Network (ARN), which draws philosophies from Spatial Transformer Network [41] and elegantly incorporates the proposed weak rectification into the modern deep learning framework. The ARN,

- *J.-W. Bian, H. Zhan, T.-J. Chin, C. Shen, and I. Reid are with The University of Adelaide and Australian Centre for Robotic Vision, Australia;*
- *N. Wang is with TuSimple, China.*

along with our proposed novel loss functions, can automatically learn to rectify images during the unsupervised training. The obtained results are comparable to training models on the pre-processed data. We make comprehensive ablation studies on the proposed methods, and validates the generalization of our trained model and learning method in several standard datasets, including ScanNet [42], 7-Scenes [43], and KITTI [33].

To summarize, our main contributions include:

- We theoretically analyze the relation between depth and two motion components in the warping. Along with the experimental analysis of the camera motion distribution in different scenarios, we answer the question why it is so challenging to train unsupervised depth CNNs from indoor videos captured by handheld cameras.
- We propose a data pre-processing method to address the complex motion, which significantly boosts the learned depth accuracy and validates our motivation.
- We propose an Auto-Rectify Network with novel loss functions towards end-to-end learning, resulting in an improved unsupervised depth learning framework.

2 Related Work

This paper address the challenges of unsupervised learning of monocular depth from motion-complex videos, which are often captured by a hand-held camera. Based on our findings that the rotation between image pairs makes depth learning more challenging, we propose two image rectification methods for removing the rotation by either data pre-processing or end-to-end learning. The related work is discussed in the following paragraphs.

Unsupervised Video Depth Estimation. Zhou et al. [19] propose the first video-based unsupervised learning method for monocular depth estimation. They rely on the color consistency between consecutive frames and the underlying 3D geometry constraint to train both depth and pose networks jointly. Following this seminal work, [26]–[29], [31], [32], [44], [45] propose different contributions to address the moving object issue in the dynamic scenes. [24], [25] propose additional geometric optimization to improve the pose estimation, and [46] proposes a better depth network architecture. The significantly improved results are presented, however, these methods only demonstrate high-quality depth estimation results in driving videos, *e.g.*, KITTI [33] or Cityscapes [34] datasets. As pointed by [35], many state-of-the-art methods such as GeoNet [26] fail to get reasonable results in NYUv2 indoor dataset [36]. We experiment with Monodepth2 [29], which sometimes also meets this issue. Through careful investigation, we identify the complex camera pose in the hand-held camera setting as the main challenge, and we address the challenge in this paper.

Before us, Zhou et al. [35] and Zhao et al. [37] have experimented in the indoor NYUv2 dataset. Specifically, [35] estimates the optical flow between image pairs, and then uses the flow instead of photometric loss to supervise the depth and pose based warping. [35] also removes purely rotated image pairs from training data, while the rotation in the remaining training image pairs are not addressed. We instead deal with the rotation in all image pairs automatically in this paper. [37] replaces the pose network with an optical flow assisted geometric module for pose estimation. However, they did not realize the rotation issue in the hand-held camera setting, and the performance is limited. Besides, a recent work [47] proposes to address the low-texture issue in the indoor scene by using patch-match and plane regularization. Their contributions are complementary to ours, and the results show that our method achievers a higher performance.

Image Rectification. Rectifying images for better depth estimation is not new in the computer vision community. For example, stereo rectification [38] makes left and right images captured by a stereo camera to be row-to-row aligned, which is a widely used pre-processing for stereo matching [39]. Besides, in the classic multi-view stereo system [48], images are also rectified before depth triangulation and optimization. We borrow this idea from classic geometry algorithms and integrate it into the deep learning based framework for monocular depth learning from motion-complex videos. A similar idea is used in [49] to learn the surface normal of tiled images. Moreover, our method is related to Spatial Transform Networks [50] and Homography estimation [51]. The former learns the image transform parameters and proposes a differentiable image warping, and the latter also learns the image rotation. The detailed differences are discussed in Sec. 5.2.

3 Problem Analysis

We first overview the unsupervised framework for depth learning. Then, we revisit the depth and ego-motion based image warping and demonstrate the relationship between depth and decomposed motions. Finally, we compare the ego-motion statistics in different scenarios to verify the impact of ego-motion on depth learning.

3.1 Unsupervised depth learning from video

Following SfMLearner [19], plenty of unsupervised depth estimation methods have been proposed. The SC-Depth [32], which is the current SOTA framework, additionally constrains the geometry consistency over [19], leading to more accurate and scale-consistent results. In this paper, we use SC-Depth as our baseline, and we overview its pipeline in Fig. 1.

Forward. A training image pair (I_a, I_b) is first passed into a weight-shared depth CNN to obtain the depth maps (D_a, D_b), respectively. Then, the pose CNN takes the concatenation of two images as input and predicts their 6D relative camera pose P_{ab}. With the predicted depth D_a and pose P_{ab}, the warping flow between two images is generated according to Sec. 3.2.

Loss. First, the main supervision signal is the photometric loss L_P. It calculates the color difference in each pixel between I_a with its warped position on I_b using a *differentiable bilinear interpolation* [41]. Second, depth maps are regularized by the geometric inconsistency loss L_{GC}, where it enforces the consistency of predicted depths between different frames. Besides, a weighting mask M is derived from L_{GC} to handle dynamics and occlusions, which is applied on L_P to obtain the weighted L_P^M. Third, depth maps are also regularized by a smoothness loss L_S, which ensures that depth smoothness is guided by the edge of images. Overall, the objective function is:

$$L = \alpha L_P^M + \beta L_S + \gamma L_{GC}, \tag{1}$$

where α, β, and γ are hyper-parameters to balance different losses.

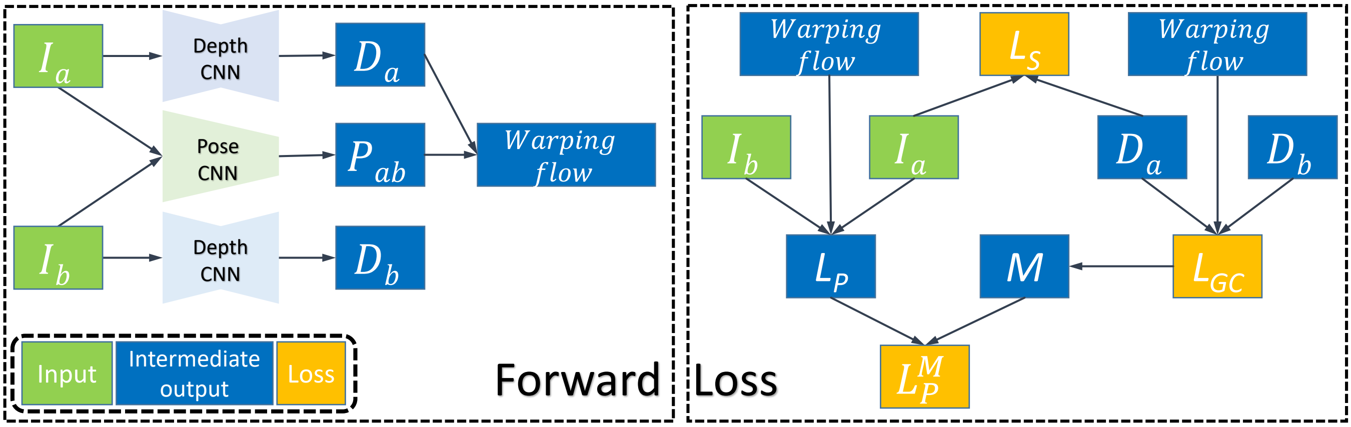


Fig. 1. Overview of SC-Depth [32]. Firstly, in the forward pass, training images (I_a, I_b) are passed into the network to predict depth maps (D_a, D_b) and relative camera pose P_{ab}. With D_a and P_{ab}, we obtain the warping flow between two views according to Eqn. 2. Secondly, given the warping flow, the photometric loss L_P and the geometry consistency loss L_{GC} are computed. Also, the weighting mask M is derived from L_{GC} and applied over L_P to handle dynamics and occlusions. Moreover, an edge-aware smoothness loss L_S is used to regularize the predicted depth map. Here we use this system as our baseline for problem analysis and illustrate our proposed components in Fig. 3.

3.2 Depth and camera pose based image warping

The image warping builds the link between networks and losses during training, *i.e.*, the warping flow is generated by network predictions (depth and camera motion) in forward pass, and the gradients are back-propagated from the losses via the warping flow to networks. Therefore, we investigate the warping to analyze the camera pose effects on the depth learning, which avoids involving image content factors, such as illumination changes and low-texture scenes.

Full transformation. The camera pose is composed of rotation and translation components. A point (u_1, v_1) in the first image is reprojected to (u_2, v_2) in the second image, which satisfies the following relationship:

$$\mathbf{K}^{-1}\left(d_2\begin{bmatrix}u_2\\v_2\\1\end{bmatrix}\right)=\mathbf{R}\mathbf{K}^{-1}\left(d_1\begin{bmatrix}u_1\\v_1\\1\end{bmatrix}\right)+\mathbf{t},\tag{2}$$

where d_i is the depth of this point in two images and $\mathbf{K}$ is the 3x3 camera intrinsic matrix. $\mathbf{R}$ is a 3x3 rotation matrix and $\mathbf{t}$ is a 3x1 translation vector. We decompose the full warping flow and discuss each component below.

Pure-rotation transformation. If two images are related by a pure-rotation transformation (*i.e.*, $\mathbf{t} = \mathbf{0}$), based on Eqn. 2, the warping satisfies:

$$d_2\begin{bmatrix}u_2\\v_2\\1\end{bmatrix}=\mathbf{K}\mathbf{R}\mathbf{K}^{-1}\left(d_1\begin{bmatrix}u_1\\v_1\\1\end{bmatrix}\right),\tag{3}$$

where $[\mathbf{K}\mathbf{R}\mathbf{K}^{-1}]$ is as known as the *homography matrix* $\mathbf{H}$ [8], and we have

$$\begin{bmatrix}u_2\\v_2\\1\end{bmatrix}=\frac{d_1}{d_2}\mathbf{H}\begin{bmatrix}u_1\\v_1\\1\end{bmatrix}=\mathbf{c}\begin{bmatrix}h_{11}&h_{12}&h_{13}\\h_{21}&h_{22}&h_{23}\\h_{31}&h_{32}&h_{33}\end{bmatrix}\begin{bmatrix}u_1\\v_1\\1\end{bmatrix},\tag{4}$$

where $c = \frac{d1}{d2}$, standing for the depth relation between two views, is determined by the third row of the above equation, *i.e.*, $c = 1/(h_{31}u_1 + h_{32}v_1 + h_{33})$. It indicates that we can obtain (u_2, v_2) without d_1. Specifically, solving the above equation, we have

$$\begin{cases}u_2=(h_{11}u_1+h_{12}v_1+h_{13})/(h_{31}u_1+h_{32}v_1+h_{33})\\v_2=(h_{21}u_1+h_{22}v_1+h_{23})/(h_{31}u_1+h_{32}v_1+h_{33}).\end{cases}\tag{5}$$

This demonstrates that the rotational flow in image warping is independent to the depth, and it is only determined by $\mathbf{K}$ and $\mathbf{R}$. Consequently, the rotational motion in image pairs cannot contribute effective gradients to supervise the depth CNN during training, even when it is correctly estimated. More importantly, if the estimated rotation is inaccurate, noisy gradients will arise and harm the depth CNN in backpropagation. Therefore, we conclude that the rotational motion behaves as the noise to unsupervised depth learning.

Pure-translation transformation. A pure-translation transformation means that $\mathbf{R}$ is an identity matrix in Eqn. 2. Then we have

$$d_2\begin{bmatrix}u_2\\v_2\\1\end{bmatrix}=d_1\begin{bmatrix}u_1\\v_1\\1\end{bmatrix}+\mathbf{K}\mathbf{t}=d_1\begin{bmatrix}u_1\\v_1\\1\end{bmatrix}+\begin{bmatrix}f_x&0&c_x\\0&f_y&c_y\\0&0&1\end{bmatrix}\begin{bmatrix}t_1\\t_2\\t_3\end{bmatrix},\tag{6}$$

where (f_x f_y) are camera focal lengths, and (c_x, c_y) are principal point offsets. Solving the above equation, we have

$$\begin{cases}d_2u_2=d_1u_1+f_xt_1+c_xt_3\\d_2v_2=d_1v_1+f_yt_2+c_yt_3\\d_2=d_1+t_3\end{cases}\quad\begin{cases}u_2=\frac{d_1u_1+f_xt_1+c_xt_3}{d_1+t_3}\\v_2=\frac{d_1v_1+f_yt_2+c_yt_3}{d_1+t_3}.\end{cases}\tag{7}$$

It shows that the translation vector $\mathbf{t}$ is coupled with the depth d_1 during the warping from (u_1, v_1) to (u_2, v_2). This builds the link between the depth CNN and the warping, so that gradients from the photometric loss can flow to the depth CNN via the warping. Therefore, we conclude that the translational motion provides effective supervision signals to depth learning.

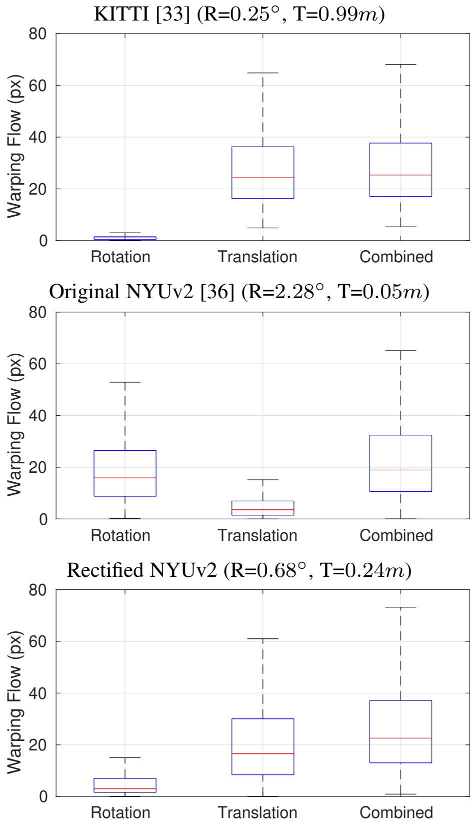


Fig. 2. Distribution of inter-frame camera motions and warping flows. "Rectified" stands for the proposed pre-processing method described in Sec. 4. In each figure, the first row shows the averaged magnitude of camera poses, *i.e.*, **R** for rotation and **T** for translation, and the plot shows the distribution of decomposed warping flow magnitudes (px) of randomly sampled points.

3.3 Distribution of decomposed camera motions

Inter-frame camera motions and warping flows. Fig. 2 shows the camera motion statistics on KITTI [33] and NYUv2 [36] datasets. KITTI is pre-processed by removing static images, as done in [19], [32]. We pick one image of every 10 frames in NYUv2, which is denoted as *Original NYUv2*. Then we apply the proposed pre-processing (Sec. 4) to obtain *Rectified NYUv2*. For all datasets, we compare the decomposed camera pose of their training image pairs *w.r.t.* the absolute magnitude and inter-frame warping flow[1]. Specifically, we compute the averaged warping flow of randomly sample points in the first image using the ground-truth depth and pose. For each point (u_1, v_1) that is warped to (u_2, v_2), the flow magnitude is $\sqrt{(u_2-u_1)^2+(v_2-v_1)^2}$. Fig. 2 shows that the rotational flows dominates the flows in *Original NYUv2* but it is opposite in KITTI. Along with the conclusion in Sec. 3.2 that the depth is supervised by the translation while the rotation behaves as the noise, this answers the question why unsupervised depth learning methods that obtain state-of-the-art results in driving scenes often fail in indoor videos. Besides, the results on *Rectified NYUv2* demonstrate that our proposed data pre-processing can effectively address the issue.

4 Proposed Data Processing

The above analysis suggests that unsupervised depth learning favours image pairs those have small rotational and moderate translational motions for training. However, unlike driving sequences, videos captured by handheld cameras tend to have significant rotational motions, as shown in Fig. 2. In this section, we describe the proposed method to rectify images for effective learning in Sec. 4.1, and discuss the method in Sec. 4.2.

4.1 Weak Rectification

Sampling image pair candidates. Given a high frame rate video, *e.g.*, 30fps, we first downsample it temporally to remove redundant images, *i.e.*, extract one key (1st) frame from every m frames. Then, instead of only considering adjacent frames as a pair, we pair up each image with its following k frames as pair candidates. Here, we use $m = 10$ and $k = 10$ in NYUv2 [36].

Two-view relative pose estimation. For each pair candidate, we first generate correspondences across images by using SIFT [5] features, and we then apply the *ratio test* [5] with GMS [32] filter to find good matches. Second, with the selected correspondences, we estimate the *essential matrix* using the five-point algorithm [40] within a RANSAC [53] framework. Then the relative camera pose is decomposed from the essential matrix. This two-view image matching pipeline has a well-established technique in Computer Vision. See more technical details in [8], [9].

3-DoF weak rectification. Given an image pair with the pre-computed relative pose, we warp both images to a common plane using the their relative rotation matrix $\mathbf{R}$. Specifically, (i) we fist convert $\mathbf{R}$ to rotation vector $\mathbf{r}$ using Rodrigues formula [54] to obtain half rotation vectors for two images (*i.e.*, $\frac{\mathbf{r}}{2}$ and $-\frac{\mathbf{r}}{2}$), and then we convert them back to rotation matrices $\mathbf{R_1}$ and $\mathbf{R_2}$. (ii) Given $\mathbf{R_1}$, $\mathbf{R_2}$, and camera intrinsic $\mathbf{K}$, we warp images to a new common image plane according to Eqn. 5. Then in the common plane, we crop their overlapped rectangular regions to obtain the weakly rectified pairs.

4.2 Discussion of the proposed weak rectification

Inspiration from Stereo Rectification [38]. The image rectification has been standard techniques for pre-processing images in the stereo matching task [39], which simplifies the problem by rectifying images so that corresponding points in two images have identical vertical coordinates. Drawing inspiration from it, we propose to weakly rectify images for easing the task of unsupervised depth learning from video. Compared with the standard rectification [38], we only consider the rotation for image warping and deliberately ignore the translation, which results in weakly rectified pairs that have 3-DoF translational motions, instead of 1-DoF motions as in [38], due to two reasons: (i) the standard rectification is used for imaged pairs that are captured by two horizontal cameras, and using it in video cases, especially the forward-motion pairs, often cause wired results such as extremely deformed images [38]; (ii) the rigorous 1-DoF rectification is unnecessary for unsupervised depth learning, because the motions in all three directions are associated with the depth during image warping, as shown in Eqn. 7. An experimental example is that the existing methods [19], [26], [28], [29], [32] that are trained on KITTI videos, where image pairs have 3-DoF translational

[1] We first compute the rotational flow using Eqn. 5, and then we get the translational flow by subtracting rotational flow from the overall warping flow. Here, the translational flow is also called *residual parallax* in [52], where it is used to compute depth from correspondences and relative camera poses.

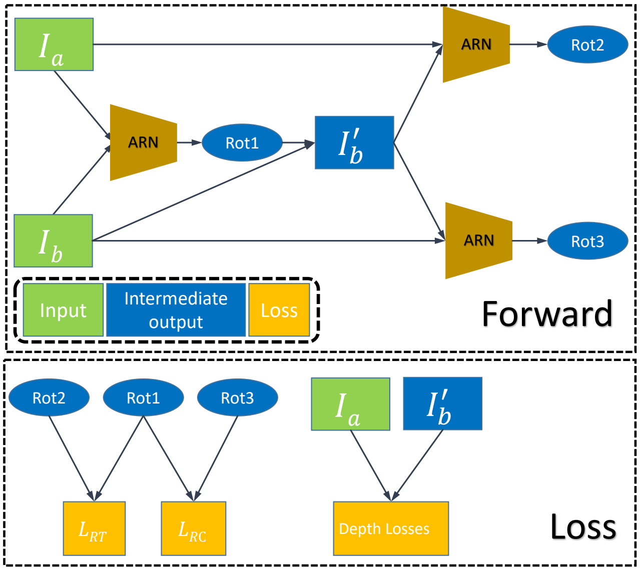


Fig. 3. Proposed Auto-Rectify Network (ARN) with loss functions. We use ARN to predict the relative rotation between two input images (I_a, I_b), and warp I_b to obtain the I_b', which is supposed to be well-aligned with I_a in terms of rotation. Then we use the image pair (I_a, I_b') for subsequent depth learning, as described in Fig. 1. The proposed loss functions are used to regularize the training of ARN.

motions, can show comparable results to methods those are trained on KITTI stereo pairs [18], [20], [21].

How to deal with unstable rectification? This is unavoidable due to the nature of geometric methods. For example, we may obtain significantly deformed images due to the inaccurate rotation estimation, which could be caused by insufficient correspondence search. However, this does not matter in our problem, because our goal is to construct good data for effective training, while we do not require that the rectification on all data is perfect. In practice, we set thresholds to remove invalid pairs, *e.g.*, based on correspondence numbers and aspect ratios of the rectified images.

Can we do rectification in end-to-end fashion? Although using our pre-processed data can lead to better performance (Tab. 1) and faster convergence (Fig. 6), the end-to-end learning approach is more favorable. For example, it benefits post-processing during inference, as recently discussed in [31], [55]. We here propose the data pre-processing solution mainly to validate our motivation, and we show how to encapsulate this idea into an end-to-end learning framework in the next section.

5 Proposed End-to-End Method

In this section, we present the proposed Auto-Rectify Network (ARN) with novel loss functions in Sec. 5.1 for end-to-end learning, and we discuss our design in Sec. 5.2.

5.1 Auto-Rectify Network with losses

Fig. 3 illustrates the proposed methods. ARN takes a concatenation of two images as input and outputs their 3-DoF relative rotation. Then we warp I_b to obtain I_b' using the predicted rotation parameters, which is supposed to be co-planar with I_a. Then we pass the image pair (I_a, I_b') to our baseline framework for training. Fig. 3 shows the pipeline and Fig. 4 shows the visualized results.

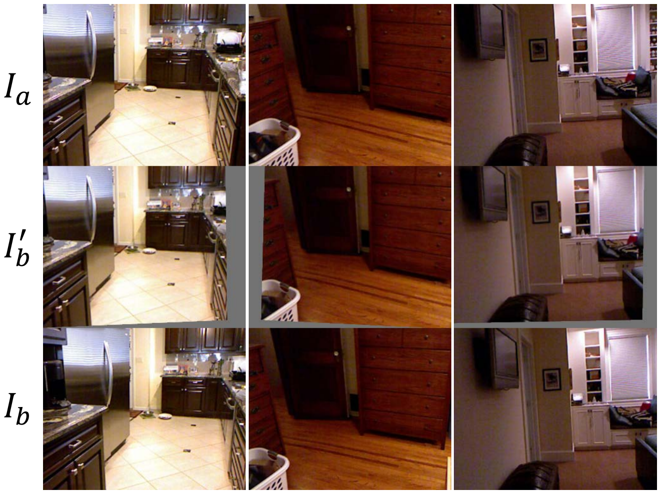


Fig. 4. Samples of ARN warped results. I_a, I_b are input images, and I_b' is the warped image by ARN. Note the grey board of I_b', which stands for the zero-padding in image warping.

Network Architecture. The network architectures of ARN is almost the same as Pose CNN, and the difference is that ARN outputs a 3-DoF rotation instead of full 6-DoF pose. We use ResNet-18 [56] encoder to extract features from a concatenation of two images, and regress their rotation parameters using several convolution layers. In order to enable two frames as input, we modify the first layer to have six channels instead of three.

Loss Functions. First, as we expect that I_b' is well-aligned with I_a *w.r.t.* rotation, $Rot2$ is minimized. However, this leads to a trivial solution (*i.e.*, the network always outputs zero), which inhabits the network ability for learning rotation and thus rectification. As a regularization, we simultaneously maximize $Rot1$. This relies in the assumption that the warped pairs are more well-aligned than original pairs. This allows us to define a Rotation-Triple loss L_{RT}. Formally,

$$L_{RT} = max(||Rot2||_1 - ||Rot1||_1 + \delta, 0), \tag{8}$$

where δ is the margin, which we empirically set to be 0.1. Besides, as I_b' is manually warped from I_b using $Rot1$, which is the ground truth of $Rot3$, the prediction of ARN on pair (I_b, I_b'). Therefore, we define an Rotation-Consistent loss L_{RC}:

$$L_{RC} = ||Rot3 - Rot1||_1, \tag{9}$$

where $||\cdot||_1$ stands for the L_1 loss. We validate the effects of both losses in Tab. 8.

5.2 Discussion of the proposed ARN

Inspiration from Spatial Transformer Network [41]. The STN is proposed to warp a single image (or its feature map) to the canonical view using the automatically learned spatial transformation parameters, and this has shown advantages to supervised learning tasks, *e.g.*, image classification. In this paper, drawing inspiration from STN, we propose ARN to learn the relative rotation between two images and rectify images by warping the second image using the learned rotation parameters to a new view, which is supposed to be better aligned with the first image. We demonstrate that using the rectified pairs leads to significantly

TABLE 1
Single-view depth estimation results on NYUv2 [36].

Methods	Supervision	Error ↓			Accuracy ↑		
		AbsRel	Log10	RMS	δ_1	δ_2	δ_3
Make3D [57]	✓	0.349	-	1.214	0.447	0.745	0.897
Depth Transfer [58]	✓	0.349	0.131	1.21	-	-	-
Liu et al. [59]	✓	0.335	0.127	1.06	-	-	-
Ladicky et al. [60]	✓	-	-	-	0.542	0.829	0.941
Li et al. [61]	✓	0.232	0.094	0.821	0.621	0.886	0.968
Roy et al. [62]	✓	0.187	0.078	0.744	-	-	-
Liu et al. [11]	✓	0.213	0.087	0.759	0.650	0.906	0.976
Wang et al. [63]	✓	0.220	0.094	0.745	0.605	0.890	0.970
Eigen et al. [12]	✓	0.158	-	0.641	0.769	0.950	0.988
Chakrabarti et al. [13]	✓	0.149	-	0.620	0.806	0.958	0.987
Laina et al. [14]	✓	0.127	0.055	0.573	0.811	0.953	0.988
Li et al. [15]	✓	0.143	0.063	0.635	0.788	0.958	0.991
DORN [16]	✓	0.115	0.051	0.509	0.828	0.965	0.992
VNL [17]	✓	**0.108**	**0.048**	**0.416**	**0.875**	**0.976**	**0.994**
MovingIndoor [35]	✗	0.208	0.086	0.712	0.674	0.900	0.968
TrainFlow [37]	✗	0.189	0.079	0.686	0.701	0.912	0.978
Monodepth2 [29]	✗	0.176	0.074	0.639	0.734	0.937	0.983
P^{2}Net (3-frame) [47]	✗	0.159	0.068	0.599	0.772	0.942	0.984
P^{2}Net (5-frame) [47]	✗	0.147	0.062	0.553	0.801	0.951	0.987
Baseline (SC-Depth [32])	✗	0.159	0.068	0.608	0.772	0.939	0.982
Ours-DP	✗	0.143	0.060	0.538	0.812	0.951	0.986
Ours-ARN	✗	**0.138**	**0.059**	**0.532**	**0.820**	**0.956**	**0.989**

TABLE 2
Unsupervised single-view depth estimation results on KITTI [33].

Methods	Error ↓				Accuracy ↑		
	AbsRel	SqRel	RMS	RMSLog	δ_1	δ_2	δ_3
SfMLearner [19]	0.208	1.768	6.856	0.283	0.678	0.885	0.957
Vid2Depth [25]	0.163	1.240	6.220	0.250	0.762	0.916	0.968
DDAD [24]	0.151	1.257	5.583	0.228	0.810	0.936	0.974
GeoNet [26]	0.155	1.296	5.857	0.233	0.793	0.931	0.973
DF-Net [27]	0.150	1.124	5.507	0.223	0.806	0.933	0.973
Struct2Depth [44]	0.141	1.026	5.291	0.215	0.816	0.945	0.979
DW [30]	0.128	0.959	5.230	0.212	0.845	0.947	0.976
GL-Net [31]	0.135	1.070	5.230	0.210	0.841	0.948	0.980
CC [28]	0.140	1.070	5.326	0.217	0.826	0.941	0.975
EPC++ [64]	0.141	1.029	5.350	0.216	0.816	0.941	0.976
Monodepth2 [29]	0.115	0.882	4.701	0.190	0.879	0.961	0.982
TrainFlow [37]	0.113	**0.704**	4.581	0.184	0.871	0.961	**0.984**
Insta-DM [45]	0.112	0.777	4.772	0.191	0.872	0.959	0.982
PackNet [46]	**0.107**	0.802	**4.538**	**0.186**	**0.889**	**0.962**	0.981
Baseline (SC-Depth [32])	0.119	**0.857**	4.950	0.197	0.863	0.957	**0.981**
Ours	**0.118**	0.861	**4.803**	**0.193**	**0.866**	**0.958**	**0.981**

TABLE 3
Zero-shot generalization results on ScanNet [42].

Methods	Supervision	Error ↓			Accuracy ↑		
		AbsRel	Log10	RMS	δ_1	δ_2	δ_3
Laina et al. [14]	✓	0.141	0.059	0.339	0.811	0.958	0.990
VNL [17]	✓	**0.123**	**0.052**	**0.306**	**0.848**	**0.964**	**0.991**
TrainFlow [37]	✗	0.179	0.076	0.415	0.726	0.927	0.980
SC-Depth [32]	✗	0.169	0.072	0.392	0.749	0.938	0.983
Ours	✗	**0.156**	**0.066**	**0.361**	**0.781**	**0.947**	**0.987**

better performance than original pairs for unsupervised depth learning in motion-complex dataset—see Tab. 1.

Relation to learning homography. Although both methods estimate a relative rotation, homography estimators solve rotation of purely rotated image pairs for accurate registration, while the goal of our proposed ARN is to rectify image pairs with general motions. In the special case where input images are related by only rotational motions, our ARN is equivalent to the homography estimator. Therefore, we propose the L_{RC} (Eqn. 9) for supervising the network training in such case. Specifically, $Rot3$ is the prediction of ARN on the purely rotated pairs (I_b, I'_b), which is manually rotated by $Rot1$, so we minimize the difference between $Rot3$ and $Rot1$ to encourage accurate rotation predictions.

How to deal with purely rotated pairs? As Eqn. 5 shows that the rotation behaves as noises during training, so that purely rotated pairs are invalid for training the depth CNN. Previous methods [35], [37] manually remove that from the dataset. However, we show that in our end-to-end learning framework, the purely rotated images could be automatically masked out. Specifically, for the image pair (I_a, I_b) that is related by purely rotational motions, after the ARN-based warping, the new pair (I_a, I'_b) would be well-aligned and become *stationary pairs*. In this scenario, the auto-mask [29] which was proposed to remove stationary points, are used in our framework for removing purely rotational points. We implement this component in our baseline framework [32], and the effects of that is shown in Tab. 9.

Can we reuse the PoseNet as ARN? It sounds possible because the ARN has a very similar network structure with the PoseNet. Specifically, we can use only the rotation output of the PoseNet in the first stage for rectification and the full pose prediction in the second stage for image warping. However, in practice, we are hard to make the network converge during training. We conjecture that it depends on the model initialization and the knowledge conflicts between two steps. More specifically, in the first stage it needs to learn coarse but big rotations, while in the second stage it needs to learn fine but small pose residuals. Therefore, we suggest not sharing parameters between the ARN and PoseNet.

6 Experiments

6.1 Implementation Details

We use SC-Depth [32] as our baseline, which is publicly available. We use the default hyper-parameters. To be specific, we use $\alpha = 1$, $\beta = 0.1$, $\gamma = 0.5$ in Eqn. 1. For training ARN, we use the weights 0.5 for L_{RT} and 0.1 for L_{RC}. We train models for 50 epochs using the Adam [65] optimizer with the learning rate being 0.0001. The batch size equals to 8 for 3-frame inputs, and it is 16 for pairwise inputs, as the former is used as two pairs during training. Besides, to demonstrate that our proposed pre-processing is universal to different methods, we experiment with Monodepth2 [29] (ResNet-18) in ablation studies, where we use the default parameters and train models for 50 epochs.

6.2 Dataset and Metrics

NYUv2 [36]. The dataset is composed of indoor video sequences recorded by a handheld Kinect RGB-D camera at 640 × 480 resolution. The dataset contains 464 scenes taken from three cities. We use the officially provided 654 densely labeled images for testing, and use the rest 335 sequences (no overlap with testing scenes) for training (302) and validation (33). The raw training

TABLE 4
Camera pose estimation results on ScanNet [42].

Method	rot (deg)	tr (deg)	tr (cm)
MovingIndoor [35]	1.96	39.17	1.40
Monodepth2 [29]	2.03	41.12	0.83
P^{2}Net [47]	1.86	**35.11**	0.89
Ours	**1.82**	39.41	**0.55**

TABLE 5
Zero-shot generalization results on Make3D [57].

Methods	Supervision	AbsRel	SqRel	RMS
Karsch [58]	✓	0.428	5.079	8.389
Liu [59]	✓	0.475	6.562	10.05
Laina [14]	✓	**0.204**	**1.840**	**5.683**
SfMLearner [19]	✗	0.383	5.321	10.470
DDVO [24]	✗	0.387	4.720	8.090
Monodepth2 [29]	✗	0.322	3.589	7.417
Ours	✗	**0.305**	**2.869**	**7.320**

TABLE 6
Fine-tuned results on 7-Scenes [43]. The model is pre-trained on NYUv2. We fine-tune models in each scene for 3 epochs, which consumes less than 10 minutes.

Scenes	No Fine-tuning		After Fine-tuning	
	AbsRel	Acc (δ_1)	AbsRel	Acc (δ_1)
Chess	0.179	0.689	**0.150**	**0.780**
Fire	0.163	0.751	**0.105**	**0.918**
Heads	0.171	0.746	**0.143**	**0.833**
Office	0.146	0.799	**0.128**	**0.855**
Pumpkin	0.120	0.841	**0.097**	**0.922**
RedKitchen	0.136	0.822	**0.124**	**0.853**
Stairs	0.143	0.794	**0.134**	**0.823**

TABLE 7
Effects of our proposed data processing (DP) on NYUv2 [36].

Methods	With	Error ↓			Accuracy ↑		
	DP	AbsRel	Log10	RMS	δ_1	δ_2	δ_3
SC-Depth [32]	✗	0.159	0.068	0.608	0.772	0.939	0.982
	✓	**0.143**	**0.060**	**0.538**	**0.812**	**0.951**	**0.986**
Monodepth2 [29]	✗	0.176	0.074	0.639	0.734	0.937	0.983
	✓	**0.151**	**0.064**	**0.559**	**0.795**	**0.947**	**0.985**

sequences contain $268K$ images. It is first downsampled 10 times to remove redundant frames. We train on the pre-processed data (*i.e.*, it generates $67K$ rectified pairs) and train directly on original data with ARN, respectively. The images are resized to 320×256 resolution for training.

KITTI [33]. The dataset contains driving videos in outdoor driving scenes, and we use it to investigate whether the proposed ARN has an adverse impact on motion-simple data. We use Eigen [10]'s splits and follow SC-Depth [32] for training and evaluation. The image is re-scaled to 832×256 resolution for training.

ScanNet [42]. The dataset provides RGB-D videos of 1513 indoor scans, captured by handheld devices. We use officially released test set, which is originally used to evaluate the semantic labeling task, for evaluating the generalization of our trained model. It contains total 2135 color images (1296×968), and corresponding ground truth depths (640×480). We re-scale the depth prediction to GT resolution for evaluation.

7-Scenes [43]. The dataset contains 7 indoor scenes, and each scene contains several video sequences (500-1000 frames per sequence), which are captured by a Kinect camera at 640×480 resolution. We follow the official train/test split for each scene. For testing, we simply extract the first image from every 10 frames, and for training, we fine-tune the model that is pre-trained on NYUv2 to demonstrate the universality of the proposed ARN.

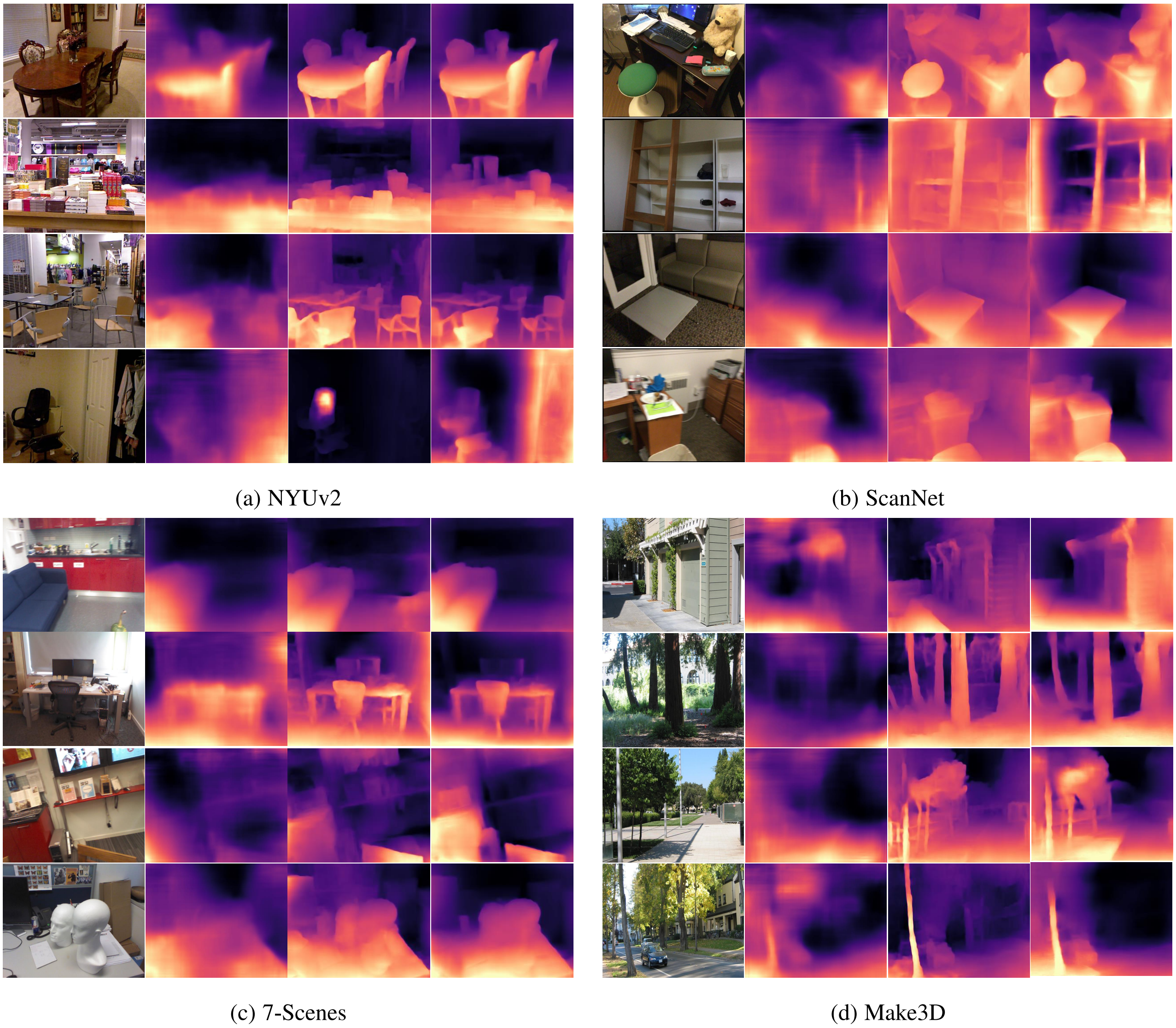
Make3D [57]. Following previous unsupervised methods [19], [29], we test the zero-shot generalization of our trained models on the Make3D dataset. It contains 134 in-the-wild images for testing. However, caution should be taken, as the ground truth depth and input images are not well aligned, causing potential evaluation issues.

Evaluation metrics. Following previous methods [11], [14], [16], [17], we use the mean absolute relative error (AbsRel), mean log10 error (Log10), root mean squared error (RMS), root mean squared log error (RMSLog), and the accuracy under threshold ($\delta_i < 1.25^i$, $i = 1, 2, 3$). As unsupervised methods cannot recover the metric scale, we multiply the predicted depth maps by a scalar that matches the median with that of the ground truth, as in [19], [29], [32]. The predicted depths are capped at $80m$ in KITTI, $70m$ in Make3D, and $10m$ in all indoor datasets.

6.3 Results

Results on NYUv2. Tab. 1 shows the single-view depth estimation results on NYUv2 [36]. It shows that our method clearly outperforms previous unsupervised methods [29], [37], [47]. See qualitative results of depth estimation in Fig. 5 and the visualization of 3D point clouds in Fig. 8. Besides, compared with SC-Depth [32], which is our baseline method, the higher performance demonstrates the efficacy of our proposed DP and ARN. Note that NYUv2 dataset is so challenging that many previous unsupervised methods such as GeoNet [26] fail to get reasonable results, as discussed in [35]. We find that Monodepth2 [29] sometimes also meets this issue, and we report its best result of multiple runs. Moreover, our method outperforms a series of fully supervised methods [11]–[13], [15], [57]–[63]. However, there still has a gap to the state-of-the-art supervised method [17].

Results on KITTI. Other than indoor datasets, we also report the results on the KITTI outdoor driving dataset [33]. Note that in the driving dataset, the image rotation is very small because most image pairs have a simple forward motion, and in this scenario our proposed rotation rectification is not necessary. Therefore, we do not run the data processing (DP) and only evaluate the effects of our proposed ARN. The results are summarized in Tab. 2. It shows that the performance of our method is slightly lower than the state-of-the-art methods [29], [37], [46]. However, compared with SC-Depth [32], which is our baseline method, the proposed



(a) NYUv2

(b) ScanNet

(c) 7-Scenes

(d) Make3D

Fig. 5. Qualitative results. Left to right: RGB, TrainFlow [37], Monodepth2 [29], and Ours. The models are trained on NYUv2 [36].

ARN module can lead to a minor improvement. It demonstrates that the proposed ARN has no adverse impact when training on motion-simple datasets.

Generalization results on ScanNet. Tab. 3 shows the zero-shot generalization results on ScanNet [42], where all models are trained on NYUv2 [36]. The results demonstrate that our trained models generalize well to new dataset. See qualitative results in Fig. 5. Besides, following [47], we provide the pose estimation results on ScanNet dataset, where 2000 image pairs from diverse scenes are selected by [66]. The results demonstrate that our method outperforms other unsupervised alternatives.

Generalization results on Make3D. Tab. 5 shows the zero-shot generalization results on Make3D [57]. Note that it is very challenging because our models are trained on indoor NYUv2 dataset but tested on in-the-wild outdoor images. Here, even though other unsupervised methods are trained on KITTI outdoor dataset, in which the images are arguably more similar to Make3D images, the results show that our indoor trained models outperform other unsupervised outdoor trained models.

Fine-tuned results on 7-Scenes. The nature of unsupervised learning makes our method easy to be fine-tuned in new datasets, *i.e.*, we can fine-tune models there without the requirement for the ground-truth labels. Tab. 6 shows the quantitative fine-tuned results on 7-Scenes [43], where the models were pre-trained on NYUv2 [36]. The results shows that our model not only generalizes well to new datasets, but also a quick fine-tuning can improve the performance significantly. This has important implications to real-world applications. Also, the results demonstrate that our proposed ARN performs well in different scenarios.

Timing. It takes 28 hours to train the models for 50 epochs on original NYUv2 dataset and 25 hours on the rectified data, measured in a single 16GB NVIDIA V100 GPU. It takes 44 hours when training with the proposed ARN. The inference speed is about 210 FPS in a NVIDIA RTX 2080 GPU, where images are resized to 320 × 256 before feeding to the network.

6.4 Ablation studies

Effects of the proposed DP. Tab. 7 shows the ablation study results on NYUv2. For both SC-Depth [32] and Monodepth2 [29]

TABLE 8
Effects of the proposed loss functions on NYUv2 [36].

With	With	Error ↓			Accuracy ↑		
L_{RT}	L_{RC}	AbsRel	Log10	RMS	δ_1	δ_2	δ_3
✗	✗	0.150	0.064	0.564	0.797	0.946	0.985
✓	✗	0.145	0.062	0.560	0.802	0.948	0.987
✗	✓	0.148	0.063	0.562	0.798	0.950	0.987
✓	✓	**0.138**	**0.059**	**0.532**	**0.820**	**0.956**	**0.989**

TABLE 9
Effects of Auto-Mask (AM) and ImageNet Pretrain (IP) on NYUv2 [36].

With	With	Error ↓			Accuracy ↑		
AM	IP	AbsRel	Log10	RMS	δ_1	δ_2	δ_3
✓	✗	0.164	0.069	0.596	0.768	0.937	0.982
✗	✓	0.153	0.065	0.571	0.790	0.945	0.986
✓	✓	**0.138**	**0.059**	**0.532**	**0.820**	**0.956**	**0.989**

frameworks, training on our pre-processed data leads to significantly improved results than using original dataset. It demonstrates that the proposed data processing method is independent to the method chosen, and our motivation of removing rotation is correct. Moreover, we visualize the quantitative learning curves in Fig. 6 for more detailed comparison.

Effects of the proposed ARN. The results in Tab. 1 show that the proposed ARN improves the depth estimation accuracy significantly on NYUv2. Besides, the results in Tab. 2 show that the performance improvement by ARN is minor on KITTI, since the camera rotation in driving scenes is almost small. As shown in Fig. 2(a), the ego-motion in NYUv2 is more complex than in KITTI. This indicates that using ARN is beneficial to training on motion-complex data (see Fig. 6), while it has less impact when training on motion-simple data.

Effects of the proposed losses. Tab. 8 shows the ablation study results of the proposed loss functions on NYUv2. Note that the ARN could converge well even without applying the proposed loss functions during training, since it can get supervision signals from the photometric loss and geometry consistency loss, contributed to the differentiable warping. This is very important because it makes our system really end-to-end trainable, leading to improved performance against the two-step learning—our data preprocessing solution. Besides, the results show that the proposed L_{RT} and L_{RC} can effectively regularize the network during training and lead to a higher performance. Moreover, Tab. 9 shows the effects of ImageNet pretraining and Auto-Mask, which is used to remove purely rotational motions.

Rotation removal by ARN. Fig. 7 shows that the proposed ARN can effectively reduce the relative image rotations, where the model is trained on NYUv2 without fine-tuning on 7-Scenes. However, when compared with traditional geometry based methods, the accuracy is lower, and the generalization is worse. Besides, Fig. 4 shows several ARN warped results during training, and Fig. 6 demonstrates that this is effective for improving depth learning. Therefore, we conclude that ARN is effective for assisting the depth learning, while not comparable to geometry-based methods in terms of rotation estimation.

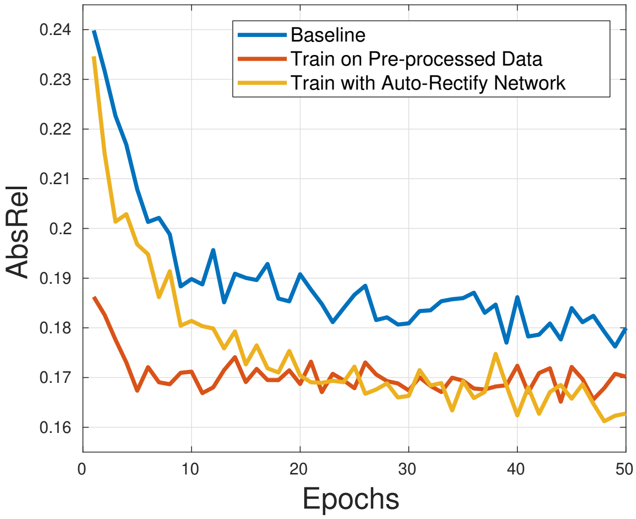
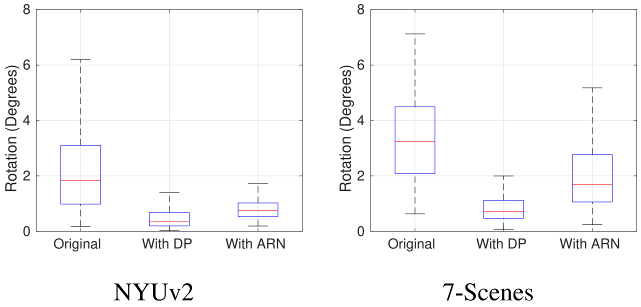


Fig. 6. Validation loss during training on NYUv2 [36].



NYUv2 7-Scenes

Fig. 7. Effects of proposed rectification methods. We test the data pre-processing (DP) and ARN on the validation sequence. The ARN model is trained on NYUv2 [36].

7 Conclusion

In this paper, we analyze the impact of ego-motion on the indoor unsupervised depth learning problem. Our analysis shows that (i) rotational motions dominate the camera motions in videos taken by handheld devices, and (ii) rotation behaves as noises while translation contributes effective signals to training. Based on the analysis, we propose a data pre-processing method to rectify the training image pairs. The results show that training on rectified image pairs can effectively improve the performance. Besides, we propose an Auto-Rectify Network with novel loss functions, which are integrated into the existing self-supervised learning framework for end-to-end learning. The comprehensive evaluation results in different datasets demonstrate the efficacy and universality of our proposed methods.

Acknowledgments

This work was in part supported by the Australian Centre of Excellence for Robotic Vision CE140100016, and the ARC Laureate Fellowship FL130100102 to Prof. Ian Reid. We thank anonymous reviewers for their valuable suggestions.

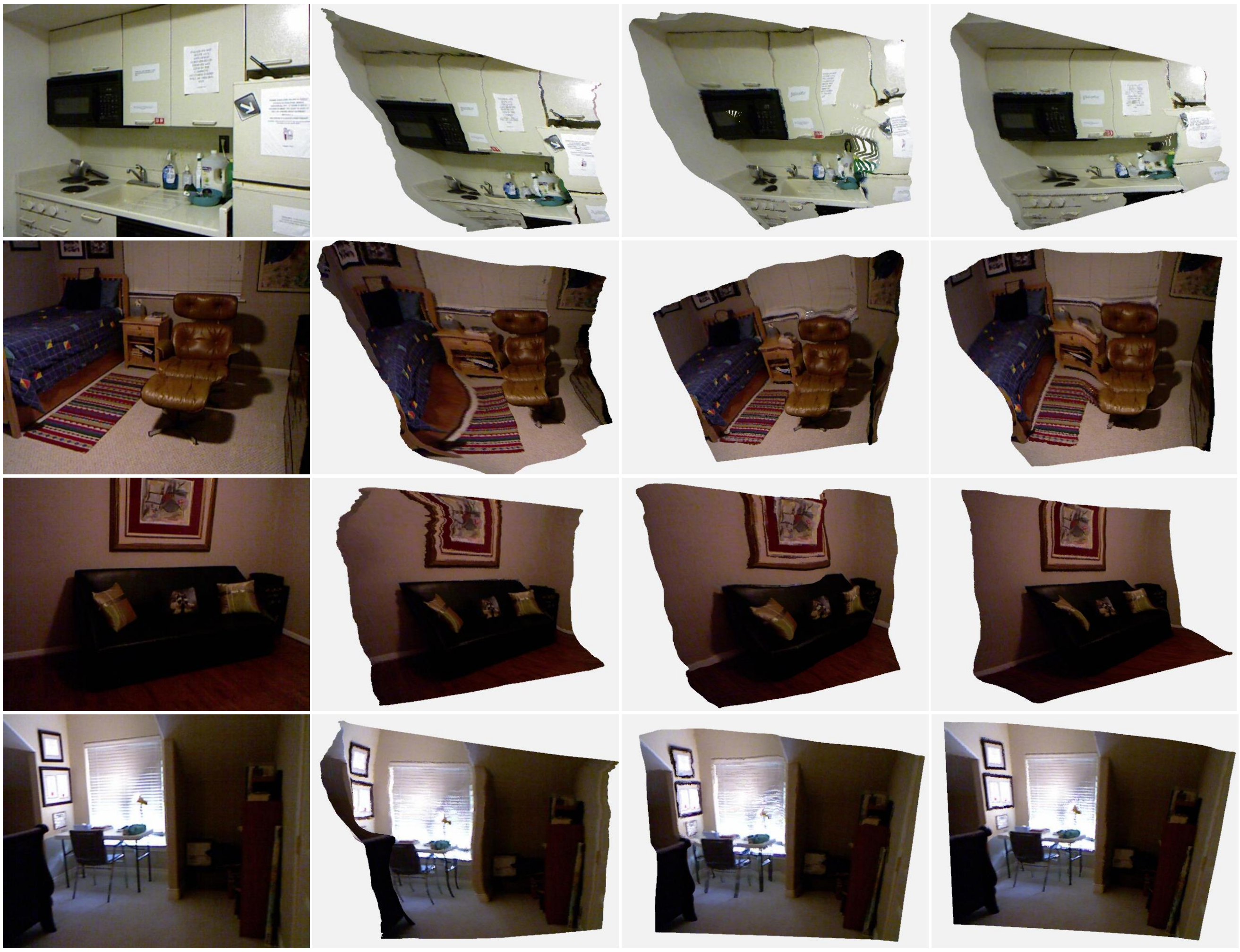


Fig. 8. Visualisation of 3D point clouds on NYUv2 [36]. Left to right: RGB, TrainFlow [37], Monodepth2 [29], and Ours.

References

[1] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[2] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "Monoslam: Real-time single camera slam," *IEEE Transactions on Pattern Recognition and Machine Intelligence (TPAMI)*, 2007.

[3] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "Dtam: Dense tracking and mapping in real-time," in *IEEE International Conference on Computer Vision (ICCV)*, 2011.

[4] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: a versatile and accurate monocular slam system," *IEEE Transactions on Robotics (TRO)*, 2015.

[5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal on Computer Vision (IJCV)*, 2004.

[6] J.-W. Bian, W.-Y. Lin, Y. Liu, L. Zhang, S.-K. Yeung, M.-M. Cheng, and I. Reid, "GMS: Grid-based motion statistics for fast, ultra-robust feature correspondence," *International Journal on Computer Vision (IJCV)*, 2020.

[7] Z. Zhang, "Determining the epipolar geometry and its uncertainty: A review," *International Journal on Computer Vision (IJCV)*, 1998.

[8] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[9] J.-W. Bian, Y.-H. Wu, J. Zhao, Y. Liu, L. Zhang, M.-M. Cheng, and I. Reid, "An evaluation of feature matchers for fundamental matrix estimation," in *British Machine Vision Conference (BMVC)*, 2019.

[10] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Neural Information Processing Systems (NeurIPS)*, 2014.

[11] F. Liu, C. Shen, G. Lin, and I. Reid, "Learning depth from single monocular images using deep convolutional neural fields," *IEEE Transactions on Pattern Recognition and Machine Intelligence (TPAMI)*, 2016.

[12] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[13] A. Chakrabarti, J. Shao, and G. Shakhnarovich, "Depth from a single image by harmonizing overcomplete local network predictions," in *Neural Information Processing Systems (NeurIPS)*, 2016.

[14] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab, "Deeper depth prediction with fully convolutional residual networks," in *3DV*, 2016.

[15] J. Li, R. Klein, and A. Yao, "A two-streamed network for estimating fine-scaled depth maps from single rgb images," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[16] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, "Deep ordinal regression network for monocular depth estimation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[17] W. Yin, Y. Liu, C. Shen, and Y. Yan, "Enforcing geometric constraints of virtual normal for depth prediction," in *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[18] R. Garg, V. K. BG, G. Carneiro, and I. Reid, "Unsupervised cnn for single view depth estimation: Geometry to the rescue," in *European Conference on Computer Vision (ECCV)*, 2016.

[19] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[20] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[21] H. Zhan, R. Garg, C. Saroj Weerasekera, K. Li, H. Agarwal, and

I. Reid, "Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[22] H. Zhan, C. S. Weerasekera, R. Garg, and I. Reid, "Self-supervised learning for single view depth and surface normal estimation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[23] J. Watson, M. Firman, G. J. Brostow, and D. Turmukhambetov, "Self-supervised monocular depth hints," in *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[24] C. Wang, J. Miguel Buenaposada, R. Zhu, and S. Lucey, "Learning depth from monocular videos using direct methods," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[25] R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[26] Z. Yin and J. Shi, "GeoNet: Unsupervised learning of dense depth, optical flow and camera pose," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[27] Y. Zou, Z. Luo, and J.-B. Huang, "DF-Net: Unsupervised joint learning of depth and flow using cross-task consistency," in *European Conference on Computer Vision (ECCV)*, 2018.

[28] A. Ranjan, V. Jampani, K. Kim, D. Sun, J. Wulff, and M. J. Black, "Competitive Collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[29] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth prediction," in *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[30] A. Gordon, H. Li, R. Jonschkowski, and A. Angelova, "Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras," in *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[31] Y. Chen, C. Schmid, and C. Sminchisescu, "Self-supervised learning with geometric constraints in monocular video: Connecting flow, depth, and camera," in *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[32] J.-W. Bian, H. Zhan, N. Wang, Z. Li, L. Zhang, C. Shen, M.-M. Cheng, and I. Reid, "Unsupervised scale-consistent depth learning from video," *International Journal on Computer Vision (IJCV)*, 2021.

[33] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets Robotics: The kitti dataset," *IJRR*, 2013.

[34] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[35] J. Zhou, Y. Wang, K. Qin, and W. Zeng, "Moving indoor: Unsupervised video depth learning in challenging environments," in *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[36] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *European Conference on Computer Vision (ECCV)*, 2012.

[37] W. Zhao, S. Liu, Y. Shu, and Y.-J. Liu, "Towards better generalization: Joint depth-pose learning without posenet," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[38] A. Fusiello, E. Trucco, and A. Verri, "A compact algorithm for rectification of stereo pairs," *Machine Vision and Applications*, 2000.

[39] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Recognition and Machine Intelligence (TPAMI)*, 2007.

[40] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE Transactions on Pattern Recognition and Machine Intelligence (TPAMI)*, 2004.

[41] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Neural Information Processing Systems (NeurIPS)*, 2015.

[42] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[43] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon, "Scene coordinate regression forests for camera relocalization in rgb-d images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[44] V. Casser, S. Pirk, R. Mahjourian, and A. Angelova, "Depth prediction without the sensors: Leveraging structure for unsupervised learning from monocular videos," in *Association for the Advancement of Artificial Intelligence (AAAI)*, 2019.

[45] S. Lee, S. Im, S. Lin, and I. S. Kweon, "Learning monocular depth in dynamic scenes via instance-aware projection consistency," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

[46] V. Guizilini, R. Ambrus, S. Pillai, A. Raventos, and A. Gaidon, "3d packing for self-supervised monocular depth estimation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[47] Z. Yu, L. Jin, and S. Gao, "P^{2}net: Patch-match and plane-regularization for unsupervised indoor depth estimation," in *European Conference on Computer Vision (ECCV)*, 2020.

[48] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *European Conference on Computer Vision (ECCV)*, 2016.

[49] T. Do, K. Vuong, S. I. Roumeliotis, and H. S. Park, "Surface normal estimation of tilted images via spatial rectifier," in *European Conference on Computer Vision (ECCV)*, 2020.

[50] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Neural Information Processing Systems (NeurIPS)*, 2015.

[51] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Deep image homography estimation," *arXiv preprint arXiv:1606.03798*, 2016.

[52] Z. Li, T. Dekel, F. Cole, R. L. Tucker, N. Snavely, C. Liu, and W. T. Freeman, "MannequinChallenge: Learning the depths of moving people by watching frozen people." *IEEE Transactions on Pattern Recognition and Machine Intelligence (TPAMI)*, 2020.

[53] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, 1981.

[54] E. Trucco and A. Verri, *Introductory techniques for 3-D computer vision*. Prentice Hall Englewood Cliffs, 1998, vol. 201.

[55] X. Luo, J.-B. Huang, R. Szeliski, K. Matzen, and J. Kopf, "Consistent video depth estimation," in *ACM Transactions on Graphics (SIGGRAPH)*, 2020.

[56] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[57] A. Saxena, S. H. Chung, and A. Y. Ng, "Learning depth from single monocular images," in *Neural Information Processing Systems (NeurIPS)*, 2006.

[58] K. Karsch, C. Liu, and S. B. Kang, "Depth transfer: Depth extraction from video using non-parametric sampling," *IEEE Transactions on Pattern Recognition and Machine Intelligence (TPAMI)*, 2014.

[59] M. Liu, M. Salzmann, and X. He, "Discrete-continuous depth estimation from a single image," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[60] L. Ladicky, J. Shi, and M. Pollefeys, "Pulling things out of perspective," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[61] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[62] A. Roy and S. Todorovic, "Monocular depth estimation using neural regression forest," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

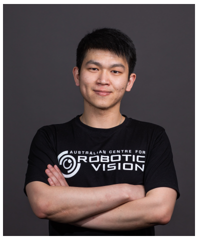
[63] P. Wang, X. Shen, Z. Lin, S. Cohen, B. Price, and A. L. Yuille, "Towards unified depth and semantic prediction from a single image," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[64] C. Luo, Z. Yang, P. Wang, Y. Wang, W. Xu, R. Nevatia, and A. Yuille, "Every pixel counts++: Joint learning of geometry and motion with 3d holistic understanding," *IEEE Transactions on Pattern Recognition and Machine Intelligence (TPAMI)*, vol. 42, no. 10, pp. 2624–2641, 2019.

[65] D. P. Kingma and J. Ba, "ADAM: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[66] Z. Teed and J. Deng, "Deepv2d: Video to depth with differentiable structure from motion," in *International Conference on Learning Representations (ICLR)*, 2018.

Jia-Wang Bian is a PhD candidate at the University of Adelaide and an Associated PhD researcher with the Australian Centre for Robotic Vision (ACRV). He is advised by Prof. Ian Reid and Prof. Chunhua Shen. His research interests lie in the field of computer vision, deep learning, and robotics. Jiawang received his B.Eng. degree from Nankai University, where he was advised by Prof. Ming-Ming Cheng. He was a research assistant at the Singapore University of Technology and Design (SUTD). Also, Jiawang did research intern jobs in many companies, including Advanced Digital Sciences Center (ADSC), Huawei, TuSimple, Amazon, and Meta.



Huangying Zhan is a Postdoctoral Researcher at the University of Adelaide, where he is working with Prof. Ian Reid @ Uni.Adelaide and Dr. Hamid Rezatofighi @ Monash University. He is associated with The Australian Institute for Machine Learning (AIML) @ UniAdelaide and Vision & Learning for Autonomous AI Lab(VL4AI) @ Monash University. His research interests include deep learning and its application in robotic vision and visual navigation. Piror to that, he received his Ph.D. degree from the University of Adelaide and he was affiliated with the Australian Centre for Robotic Vision, where he was advised by Prof. Ian Reid and Prof. Gustavo Carneiro. Previously, he received my B.Eng degree in Electronic Engineering (first class honors) from The Chinese University of Hong Kong (CUHK), where he was advised by Prof. Xiaogang Wang. Also, he was a visiting student in the Unmanned Systems Research Group at The National University of Singapore, where he worked with Prof. Ben M. Chen.

Naiyan Wang is currently the chief scientist of TuSimple. he leads the algorithm research group in the Beijing branch. Before this, he got his PhD degree from CSE department, HongKong University of Science and Technology in 2015. His supervisor is Prof. Dit-Yan Yeung. He got his BS degree from Zhejiang University, 2011 under the supervision of Prof. Zhihua Zhang. His research interest focuses on applying statistical computational model to real problems in computer vision and data mining. Currently, He mainly works on the vision based perception and localization part of autonomous driving. Especially He integrates and improves the cutting-edge technologies in academia, and makes them work properly in the autonomous truck.

Tat-Jun Chin received his PhD in Computer Systems Engineering from Monash University in 2007, which was supported by the Endeavour Australia-Asia Award, and a Bachelor in Mechatronics Engineering from Universiti Teknologi Malaysia in 2004, where we won the Vice Chancellor's Award. He currently holds the SmartSat CRC Professorial Chair of Sentient Satellites at The University of Adelaide. He is also the Director of Machine Learning for Space at The Australian Institute for Machine Learning. Tat-Jun's research interest lies in optimisation for computer vision and machine learning, and their application to robotic vision, space and smart cities. He has published more than 100 research articles on the subject, and has won several awards for his research, including a CVPR award (2015), a BMVC award (2018), Best of ECCV (2018), two DST Awards (2015, 2017) and IAPR Award (2019).

Chunhua Shen is a Professor of Computer Science at the University of Adelaide, Australia.

Ian Reid is the Head of the School of Computer Science at the University of Adelaide, and the senior researcher at the Australian Institute for Machine Learning. He is a Fellow of Academy of Technological Sciences and Engineering and held an ARC Australian Laureate Fellowship there 2013-18. He is also Deputy Director of the ARC Centre of Excellence in Robotic Vision, in which capacity he also co-chaired sub-committees of the National Roadmap for Robotics produced and published by the Centre in 2018. Between 2000 and 2012 he was a Professor of Engineering Science at the University of Oxford. where, together with long-time colleague Prof. David Murray, he ran the Active Vision Group which is part of the wider Robotics Research Group. He received a BSc in Computer Science and Mathematics with first class honours from University of Western Australia in 1987 and was awarded a Rhodes Scholarship in 1988 to study at the University of Oxford, where he obtained a D.Phil. in 1992.