This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

PanoPose: Self-supervised Relative Pose Estimation for Panoramic Images

Diantao Tu [1,2,3] Hainan Cui[1,2,3†] Xianwei Zheng [4] Shuhan Shen[1,2,3†]
[1]Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
[3]CASIA-SenseTime Research Group [4]The State Key Lab. LIESMARS, Wuhan University
tudiantao2020@ia.ac.cn, hncui@nlpr.ia.ac.cn, zhengxw@whu.edu.cn, shshen@nlpr.ia.ac.cn

Abstract

Scaled relative pose estimation, i.e., estimating relative rotation and scaled relative translation between two images, has always been a major challenge in global Structure-from-Motion (SfM). This difficulty arises because the two-view relative translation computed by traditional geometric vision methods, e.g. the five-point algorithm, is scaleless. Many researchers have proposed diverse translation averaging methods to solve this problem. Instead of solving the problem in the motion averaging phase, we focus on estimating scaled relative pose with the help of panoramic cameras and deep neural networks. In this paper, a novel network, namely PanoPose, is proposed to estimate the relative motion in a fully self-supervised manner and a global SfM pipeline is built for panorama images. The proposed PanoPose comprises a depth-net and a pose-net, with self-supervision achieved by reconstructing the reference image from its neighboring images based on the estimated depth and relative pose. To maintain precise pose estimation under large viewing angle differences, we randomly rotate the panoramic images and pre-train the pose-net with images before and after the rotation. To enhance scale accuracy, a fusion block is introduced to incorporate depth information into pose estimation. Extensive experiments on panoramic SfM datasets demonstrate the effectiveness of PanoPose compared with state-of-the-arts.

1. Introduction

Structure-from-motion (SfM) has always been an essential research field in computer vision. SfM consists of three key steps: feature extraction and matching, pairwise relative pose estimation, and global pose estimation. Based on the different strategies used in the third step, SfM can be categorized into incremental [8, 27] and global [7, 29]. Incremental SfM usually begins with two or three "seed" views and gradually adds more images to the model. To mitigate the accumulated errors, the Bundle Adjustment (BA) is performed after every few images are added. The complexity of BA problem increases with the number of images, leading to more and more time spent in BA, which finally affects the computation efficiency.

On the other hand, global SfM simultaneously estimates all camera poses, requiring BA only once with higher efficiency. The pose estimation in global SfM consists of rotation averaging and translation averaging, which compute the global rotation and translation, respectively. In the ideal case, the translation averaging holds the following equation

$$\mathbf{t}_j - \mathbf{t}_i = d_{ij}\mathbf{R}_i\mathbf{v}_{ij} \ , \tag{1}$$

where $\mathbf{t}_i$ and $\mathbf{t}_j$ are the camera center, $\mathbf{R}_i$ is the rotation from i-th camera to the world coordinate, $\mathbf{v}_{ij}$ is a unit vector representing the direction of relative translation between i-th and j-th camera, and $d_{ij} = \|\mathbf{t}_j - \mathbf{t}_i\|_2$ is the scale of relative translation. However, the relative translation derived from two images with traditional five-point algorithm[24] only contains the direction, leaving d_{ij} unknown. The scale ambiguity is one of the major challenges in global SfM and becomes particularly conspicuous when the camera moves along a straight path. Various translation averaging methods aim to resolve this by optimizing relative scales [23] or minimizing the angular error between direction vectors [38, 43]. Additionally, the relative pose estimation requires a rich textured environment with numerous reliable feature matching. In textureless scenes, relative pose estimation will produce large errors or degradation, affecting the overall SfM process.

To tackle these challenges, we focus on estimating the scaled relative pose. Inspired by the self-supervised monocular depth estimation [10, 42], we propose PanoPose, consisting of a depth-net and a pose-net. Different from the aforementioned method that focuses on accurate depth estimation and design of complex depth-net architecture, we give more attention to pose estimation and use a pre-trained model [37] as the backbone with differentiable pose pre-

†Corresponding author.

diction layers. To accommodate large angle differences in views, we utilize the characteristics of panoramic images to randomly rotate the images to pre-train the pose prediction layer. To further improve the accuracy of scale estimation, we propose a fusion block, integrating depth information into relative translation estimation. Moreover, to facilitate the scale prior, we propose a translation averaging method that directly optimizes the scale of relative translation. This method is complemented by an iteratively reweighted least squares (IRLS) strategy, enhancing accuracy and robustness. The PanoPose network and our proposed translation averaging method, as well as a rotation averaging method [5] form a comprehensive global SfM pipeline.

Why Pano? Panoramic cameras, with a simple camera model and omnidirectional field of view (FoV), outperform pinhole and multi-camera systems in image matching robustness and 3D reconstruction completeness. Thus, panoramic cameras are increasingly used in practical applications, especially in indoor environments, such as room layout estimation [16, 31] and depth estimation [28]. However, research specifically focused on panoramic camera 3D reconstruction is relatively limited. This research gap motivates our study.

To summarize, the main contributions of this paper are:

- We propose a self-supervised network that estimates the scaled relative pose between two panoramic images, mitigating the scale ambiguity in traditional approaches.
- We propose a pre-training strategy by randomly rotating images to improve the pose estimation accuracy under large viewing angle differences and a fusion block to incorporate depth information into pose estimation.
- Based on the network, we build a global SfM pipeline for panoramic images, and outperform state-of-the-art global SfM methods in panoramic datasets.

2. Related Work

2.1. Relative Pose Estimation

The traditional relative pose estimation method is the five-point algorithm [24], which estimates the essential matrix between images based on epipolar geometry and decomposes it to relative rotation and translation. However, the method has poor performance in textureless areas and is unable to estimate the scale of relative translation. Compared to the traditional method, the deep learning technique can alleviate the above problems, thus many researchers have focused on directly regressing the pose using neural network. These methods can be roughly divided into two categories, supervised and self-supervised.

For the supervised pose estimation, Kendall et al. [17] propose PoseNet that regresses the camera poses from a single image for camera relocalization. [1] and [14] use CNN to estimate motion. However, their work focuses on learning feature representation instead of ego-motion. Thus, the relative pose accuracy is not competitive with geometric methods. DeMoN [32] proposes a framework composed of multiple stacked encoder-decoder networks and estimates the optical flow, depth, and ego-motion. To estimate the relative pose with a wide baseline, DirectionNet [6] estimates discrete distributions of camera poses and introduces a novel parameterization to make the estimation tractable. RelPose [41] proposes an energy-based formulation to capture the uncertainty in relative poses and estimate the relative rotation for images of a generic object.

For self-supervised relative pose estimation, since adjacent images are used as supervision signals, it is often necessary to estimate the depth map simultaneously. SfMLearner [42] proposes a learning framework with a single-view depth-net and multi-view pose-net. The two networks are trained using a loss based on warping nearby images to the target with the estimated depth and pose. SfMLearner lays the foundation for self-supervised depth/pose estimation, and our method also uses a similar structure. UnDeepVO [18] uses stereo image pairs to train the network, so the network can recover the scaled depth map and relative motion. GeoNet [39] decomposes motion into rigid and non-rigid components and uses a joint learning framework to estimate depth, optical flow, and relative pose.

Some works focus on relative pose estimation using panoramic images. Hutchcroft et al. [13] propose CoVisPose that estimates relative pose for wide-baseline indoor panoramas. To directly regress the camera pose, the network jointly learns dense bidirectional visual overlap, correspondence, and room layout in a supervised manner. Thus, it is only suitable for indoor environments. Both [19] and [21] use a similar network architecture as SfMLearner and change the inputs to panoramic images.

The most similar work to ours is [35]. The main differences between our work and theirs are: 1) they project the images to cubic projection while we use the original equirectangular projection; 2) we add a fusion block to utilize the depth information from depth-net to better estimate relative translation scales.

2.2. Global Structure-from-Motion

Given the relative pose of image pairs, global SfM estimates all camera poses simultaneously, involving two steps: rotation averaging and translation averaging. Rotation averaging is computing the global rotation for each camera based on the relative rotations, while translation averaging estimates the global translation using pairwise relative translations and the global rotation. Govindu [11] gives the first rotation averaging method by representing the rotation in quaternion and solving the problem with linear least-squares fitting. To enhance the robustness of rotation averaging, Chatterjee and Govindu [5] optimize the problem

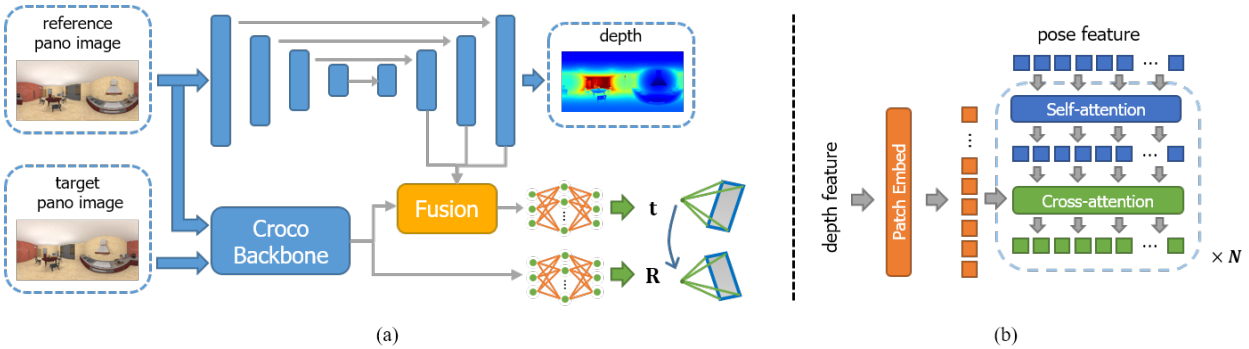


Figure 1. (a) is the overview of our network. (b) is the detailed architecture of the proposed fusion block.

under L_1-norm and proposed an IRLS strategy to further refine the result. Compared to rotation averaging, translation averaging is more difficult, since only the direction of relative translation is determined, leaving the scale unknown. The situation worsens when the baseline is short or the camera centers collinear. Jiang et al. [15] propose a linear method for estimating global translations by minimizing geometric error. Moulon et al. [23] optimize the problem under L_∞-norm and estimate the scale of translation within a triplet. BATA [43] uses a carefully designed simple bilinear objective function and introduces a variable to perform the requisite normalization. A pose-only solution was derived from [4], which gives a linear solution to the translation averaging problem. [22] proposed a framework that iteratively refines the relative translation direction using the point correspondences between two images.

3. Method

3.1. Overview

The overview of the proposed network is shown in Fig. 1, which comprises a CNN-based depth-net and a transformer-based pose-net. The depth-net takes a panoramic image as input and generates a dense depth map, while the pose-net takes pairs of images and directly regresses the 6-degree-of-freedom (6-DoF) relative pose. Different from other self-supervised methods [21, 34] that design complex depth-net for better depth estimation, we pay more attention to the accuracy of relative poses. Also, using a heavy depth-net can make the overall network hard to convex or stuck in local minimal, so the depth-net is as lightweight as possible. The choice of different depth-net architecture is discussed in supplementary material. Our depth-net architecture closely resembles Monodepth2 [10], which uses a ResNet-18 as an encoder and multiple deconvolution and upsampling block as a decoder. The depth-net outputs depth maps at different scales, $(H, W), (H/2, W/2), (H/4, W/4)$, where H and W are the height and width of the input image.

In the following, Sec. 3.2 introduces the detailed structure of our pose-net and the proposed depth fusion block, Sec. 3.3 presents the losses in our framework, Sec. 3.4 shows our rotation-only pre-training strategy, and the proposed global SfM pipeline is in Sec. 3.5.

3.2. Pose-net with Fusion Block

Transformer[33] has been widely used in panorama depth estimation [28, 44], but its potential in pose estimation is not fully exploited. Also, training vision transformers in a self-supervised manner is challenging and demands a substantial amount of data. Thus, we adopt Croco [37] as our pose-net backbone, which is a transformer-based network trained unsupervisedly via cross-view completion. After the Croco, input images are represented as high-dimensional features, and we use a rotation estimator and a translation estimator to directly regress the relative pose. The rotation estimator is an MLP that consists of two hidden layers with a hidden dimension of 1024. The output of the rotation estimator is a 9-dimension vector and we use Procrustes mapping to project it to the closet rotation matrix. According to [3], the Procrustes representation has better performance than the rotation-vector and quaternion representation. The translation estimator is similar to the rotation estimator and the difference is the output is a 3-dimension vector.

To incorporate depth information into the relative translation estimation process and enhance scale accuracy, a depth fusion block is introduced before the translation estimator. The detailed structure of our fusion block is illustrated in Fig. 1(b). It consists of multiple transformer blocks, with each block encompassing both a self-attention layer and a cross-attention layer. The inputs to this fusion block are twofold: the depth feature, derived from the depth-net at various scales, and the pose feature, which originates from the Croco backbone. In the fusion block, we incorporate three transformer blocks to fuse depth information from $(H, W), (H/2, W/2), (H/4, W/4)$. For a specific depth feature map from scale i, it is firstly divided into non-overlap patches with patch size $p \times p$, where $p = 16/2^i$. Then, a convolution layer with kernel size equal to patch size is applied to each patch. The output channel of this con-

volution layer matches the dimensions of the pose feature. A standard transformer block with self-attention and cross-attention is employed to fuse the information from different networks. The pose features are first passed through the self-attention layer and fused with the embedded depth feature by the cross-attention layer. In each attention layer, we use the Rotary Positional Embedding (RoPE) [9] which encodes the relative positioning of feature pairs when computing attention.

3.3. Loss

Following a similar approach to [42], we formulate our problem as minimizing the photometric reprojection error. For each pixel $\mathbf{p}$ in reference image I_r, its pixel coordinates can be projected to the target image I_t as

$$\mathbf{p}' = \pi^{-1}\left(\mathbf{R}\pi(\mathbf{p}, d) + \mathbf{t}\right) . \tag{2}$$

Here, $(\mathbf{R}, \mathbf{t})$ is the relative pose and d is the depth of $\mathbf{p}$ in depth map D_r. $\pi(*)$ is the transformation from the image coordinate to local camera coordinate, while $\pi^{-1}(*)$ represents the inverse transformation. We employ bilinear sampling to sample the target image and get the color value of $\mathbf{p}'$. After all pixels in I_r are projected to I_t, the reconstructed image I'_r is derived. Following [10], the L1-norm and SSIM [36] are used to compute the photometric loss as

$$\begin{aligned} \mathcal{L}_p &= \sum_{\mathbf{p}\in I_r} L_{ssim}(\mathbf{p}) + L_1(\mathbf{p}) \\ L_{ssim}(\mathbf{p}) &= \frac{\alpha}{2}\left(1 - SSIM(W_{\mathbf{p}}, W'_{\mathbf{p}})\right) , \\ L_1(\mathbf{p}) &= (1-\alpha)\|I_r(\mathbf{p}) - I'_r(\mathbf{p})\|_1 \end{aligned} \tag{3}$$

where $L_{ssim}(\mathbf{p})$ and $L_1(\mathbf{p})$ are the SSIM loss and L1 loss for each pixel, respectively. W_p represents the image patch centered at $\mathbf{p}$ in I_r and $W'_{\mathbf{p}}$ is the image patch centered at $\mathbf{p}$ in I'_r. The patch size is set to 7×7. $SSIM(W_{\mathbf{p}}, W'_{\mathbf{p}})$ computes the image similarity between two image patches. α is a weight to balance the SSIM and L1 difference and it is set to 0.85 as other self-supervised methods.

In addition to photometric loss, an edge-aware smooth loss is applied to the estimated depth to reduce noise. The loss is computed both horizontally and vertically, and the expression is

$$\begin{cases} D^*_r = & D_r/\bar{d_r} \\ \mathcal{L}_e = & |\nabla_x D^*_r| e^{-|\nabla_x I_r|} + |\nabla_y D^*_r| e^{-|\nabla_y I_r|} \end{cases} . \tag{4}$$

Here, $\bar{d_r}$ is the average of all depth values and D^*_r is the mean-normalized depth that can discourage shrinking of the estimated depth. ∇_x and ∇_y denote computing the second order gradient in different axes.

To further enhance the precision of our pose-net's pose estimation, we introduce the pose consistency loss. Given the input images as I_r and I_t, the relative pose output by the network is $(\mathbf{R}_1, \mathbf{t}_1)$. We exchange the order of the input images and obtain an output relative pose of $(\mathbf{R}_2, \mathbf{t}_2)$. Theoretically, $(\mathbf{R}_1, \mathbf{t}_1)$ and $(\mathbf{R}_2, \mathbf{t}_2)$ should be exact opposites. Our pose consistency loss is based on this assumption, which can be expressed as

$$\mathcal{L}_c = \|\mathbf{R}_1 - \mathbf{R}_2^T\|_F + \beta\|\mathbf{t}_1 - \mathbf{R}_2^T\mathbf{t}_2\|_2^2 . \tag{5}$$

$\|\cdot\|_F$ is the Frobenius norm between two matrix and $\beta = 10$ to balance the rotation error and translation error.

The final loss is

$$\mathcal{L}_{final} = \mathcal{L}_p + \lambda_e\mathcal{L}_e + \lambda_c\mathcal{L}_c , \tag{6}$$

where λ_e and λ_c are the weights for the depth smoothness loss and the pose consistency loss.

3.4. Rotation Only Pre-training

In practical applications, such as when capturing images handheld, the camera may undergo rapid rotations, resulting in significant perspective changes between consecutive frames. When using an ordinary pinhole camera, this can lead to small overlapping between images, potentially yielding inaccurate relative pose estimates. Panoramic cameras have an omnidirectional field of view and are robust to rapid changes in perspective. This is one of the key reasons that we choose panoramic cameras.

The main supervision signal of self-supervised methods comes from the reprojection error of the image. Thus, current self-supervised methods assume relatively smooth camera movements and small variations in viewing angles between adjacent frames. However, when the angular disparity between two images is substantial, the images themselves can differ significantly. The depth estimated by the depth-net is also inaccurate, which will lead to large errors in reprojection. In such cases, the network optimization process may be misguided, potentially leading to convergence in local optima. To address this challenge, we propose a rotation-only pre-training strategy, which utilizes the inherent geometry of panoramic images.

Specifically, for an image I, a rotation axis (r_x, r_y, r_z) and an angle r_d are randomly selected. Since the panoramic image has an omnidirectional FoV and can be rotated without requiring depth information, I is rotated with the pre-generated rotation and gets I_r. We use the pose-net to predict the relative rotation between I and I_r and the loss $\|\mathbf{R} - \mathbf{R}^*\|_F$ is used to supervise the network, where $\mathbf{R}^*$ is the generated rotation and $\mathbf{R}$ is the predicted. In most cases, the camera is mainly rotated around the y-axis, we take advantage of this feature and require the generated rotation axis to satisfy $|r_y| > |r_x| + |r_z|$. While this strategy provides pre-training exclusively for relative rotation, it proves effective in enhancing relative pose estimation, particularly in scenarios with significant differences in viewing angles.

3.5. Proposed SfM Pipeline

We further propose an SfM pipeline that leverages our PanoPose as the relative pose estimator. In this pipeline, the method proposed by [5] is employed for rotation averaging. Since the network can estimate scaled relative translation, the translation averaging problem can be solved by optimizing it under L_2-norm with an iteratively reweighted least squares (IRLS) scheme, namely L2IRLS. The problem is formulated as

$$\begin{aligned} \arg\min \quad & \sum_{(i,j)} w_{ij} \left\| \mathbf{t}_i - \mathbf{R}_{ij}\mathbf{t}_j - \lambda_{ij}\mathbf{t}_{ij} \right\|_2^2 \\ s.t. \quad & 0.5 s_{ij} \leq \lambda_{ij} \leq 1.5 s_{ij},\ \mathbf{t}_1 = (0,0,0)^T \end{aligned}, \quad (7)$$

where $(\mathbf{R}_{ij}, \mathbf{t}_{ij})$ is the relative pose, $\mathbf{t}_i$ is the global translation for i-th image, λ_{ij} is the scale of relative translation and s_{ij} is the initial scale derived from the network. In the first constraint, we allow λ_{ij} to fluctuate within a certain range. The second constraint is to remove translation ambiguity. $w_{ij} = \left(\left\| \mathbf{t}_i - \mathbf{R}_{ij}\mathbf{t}_j - \lambda_{ij}\mathbf{t}_{ij} \right\|_2^2 + \delta \right)^{-1/2}$ is the weight of each cost, and δ is set to 0.01. The minimizing process will iterate 2-3 times before converging, and the weight w_{ij} is initialized to 1 in the first iteration.

4. Experiment

4.1. Implementation Details

Our PanoPose is implemented using PyTorch [26] framework and Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. Two RTX A6000 GPUs are used to train the network with a batch size of 10. A pre-trained depth-net from [10] is employed to give a better initialization. In each experiment dataset, the network is pre-trained for 10 epochs with the proposed rotation-only pre-training strategy. Then, the network is trained for 40 epochs. During training, all images are resized to 320×640 and the learning rate is $1e$-5.

4.2. Evaluation Metrics and Datasets

For the evaluation of relative pose, relative rotation error (RRE), relative translation angle error (RTAE), and relative scale error (RSE) are used. RRE represents the accuracy of relative rotation, RTAE shows the accuracy of relative translation direction, and RSE reflects the accuracy of the relative translation scale. As for the evaluation of global pose, absolute rotation error (ARE) and absolute translation error (ATE) are used. Detailed calculation methods for these error metrics are in the supplementary material. To verify the performance of our method, Experiments are conducted on the following five datasets, from small indoor environments to large outdoor scenes.

PanoSUNCG. PanoSUNCG[34] is a synthetic indoor environment dataset, which contains 103 scenes of the SunCG dataset [30] and has 25,000 panoramic images. All images are provided with dense depth maps and ground truth poses. In our experiments, the official training and testing splits are used, where 80 scenes for training and 23 scenes for testing. For each image, we find its 5 nearest neighbor images based on the position of the camera center and randomly select two of them to generate image pairs, which results in 50,000 image pairs.

Mapillary Metropolis. The Mapillary Metropolis dataset [1] contains 10,274 panoramic images with ground truth poses. Additionally, we render a sparse depth map for each image based on the LiDAR reconstruction result provided by the dataset. In our experiments, the official training and validating splits are used, which include 6845 and 3347 images, respectively. Since the interval between image sampling is relatively long, we use two frames that are adjacent in time as an image pair.

360VO Dataset. The 360VO dataset [12] is a synthetic dataset rendered from the urban scene [20]. It contains 10 sequences (Seq 0 to Seq 9) with a total of 23,000 panoramic images, each with a ground truth pose. Since the dataset is designed for visual odometry, it lacks depth information and predefined train/validation/test splits. We use sequences 1, 4, and 9 for test and other sequences for training. For each image, we use its 10 nearest neighbors to generate image pairs, which results in 230,000 pairs.

Building and Campus. Among the data sets mentioned above, only the Mapillary Metropolis dataset is based on real-world scenes. However, the images are uniformly sampled at a fixed distance of 6 meters, resulting in a sparsely connected pose graph that makes global pose estimation challenging. Thus, we have collected our own datasets with an Insta 360 ONE X2 panoramic camera, namely Building and Campus. The Building dataset is collected around several buildings and the Campus dataset is collected on a vast campus. The ground truth poses are acquired by an RTK GNSS. We use the panoramic MVS method to generate the sparse depth map for each image. The Building dataset contains 1424 images for training and 300 images for testing, while the Campus uses 6125 images for training and 1,500 images for testing. For each image, the image pairs are generated with its nearest 7 neighbors.

4.3. Relative Pose Estimation

In this experiment, we compare our method against some state-of-the-art self-supervised pose estimation methods, including SfMLearner [42], MonoDepth2 [10], NonLocal-DPT [40], and BiFuse++ [35]. SfMLearner and MonoDepth2 are designed for normal pinhole cameras, but they can process input images at arbitrary resolutions. So we retrained the network with their default settings on our experiment dataset to ensure fair comparisons. In [40], the

[1] https://www.mapillary.com/dataset/metropolis

Dataset	Method	Mean RRE	Med RRE	Mean RTAE	Med RTAE	Mean RSE	Med RSE
360VO-Seq1	SfMLearner [42]	3.2378	0.8158	5.2945	1.0116	0.2529	0.1307
	MonoDepth2 [10]	2.1394	0.2341	4.9963	0.6907	0.1931	0.1457
	NonLocal-DPT [40]	1.5878	0.1872	5.0870	0.9376	0.2391	0.1833
	BiFuse++ [35]	2.5309	0.2056	10.9642	3.7566	0.5800	0.2651
	Five-point [24]	**0.0294**	**0.028**	**0.0976**	**0.0713**	1.8622	0.5087
	PanoPose	0.1520	0.0791	0.9390	0.4494	**0.1630**	**0.1449**
Mapillary Metropolis	SfMLearner [42]	2.8751	0.8528	2.9834	0.9398	0.2534	0.1852
	MonoDepth2 [10]	2.5383	0.9582	2.1671	0.5328	0.1577	0.0995
	NonLocal-DPT [40]	2.1328	0.6906	2.3511	0.4885	0.0934	0.0675
	BiFuse++ [35]	2.6898	0.5574	33.5934	32.3928	0.1952	0.1346
	Five-point [24]	**0.1691**	**0.0685**	**1.281**	**0.2741**	**0.0133**	**0.0056**
	PanoPose	1.7228	0.2683	1.7661	0.4006	0.0217	0.0101
PanoSUNCG	SfMLearner [42]	1.2548	0.4922	1.8113	1.0338	0.3562	0.1855
	MonoDepth2 [10]	1.9562	0.8227	2.9704	1.1927	**0.2877**	0.0962
	NonLocal-DPT [40]	2.0587	0.9024	2.9158	1.1361	0.3091	0.0781
	BiFuse++ [35]	5.6007	1.6863	5.0651	2.5947	0.7836	0.1531
	Five-point [24]	0.3453	0.1507	3.4659	1.5776	0.5875	0.366
	PanoPose	**0.1559**	**0.0560**	**0.4253**	**0.2874**	1.2295	**0.0115**
Building	SfMLearner [42]	0.6557	0.3744	1.4321	0.4402	0.1039	0.0810
	MonoDepth2 [10]	**0.1317**	0.1119	1.5811	0.4729	0.1232	0.1012
	NonLocal-DPT [40]	0.4535	0.2243	1.9583	0.5702	0.1325	0.0905
	BiFuse++ [35]	1.6576	0.3897	2.0919	1.4433	0.2212	0.1592
	Five-point [24]	0.3961	**0.0685**	15.182	1.4175	0.3406	0.1066
	PanoPose	0.2009	0.1427	**0.4653**	**0.3892**	**0.0935**	**0.0733**
Campus	SfMLearner [42]	0.3395	0.1039	2.6042	0.6923	0.7214	0.0993
	MonoDepth2 [10]	0.2815	0.1359	2.7216	0.5135	0.8789	0.0677
	NonLocal-DPT [40]	0.2284	0.0932	**2.0066**	0.9848	0.5692	0.0934
	BiFuse++ [35]	0.9410	0.3369	2.6573	0.6410	66.8591	0.1102
	Five-point [24]	0.1525	**0.0396**	2.7876	**0.3605**	1.0854	**0.0378**
	PanoPose	**0.1094**	0.0862	2.2683	0.4644	**0.2563**	0.0519

Table 1. Relative pose evaluation result on different datasets. The unit of relative rotation error (RRE) and relative translation angle error (RTAE) is degree, and relative scale error (RSE) is unitless. The mean and median errors are demonstrated in the table. The best result is shown in **bold** and the second best is shown with under line.

authors assume that panoramic images are aligned with the direction of gravity, restricting their network to estimate 4-DOF poses. We modified their network to output 6-DOF poses. For a full comparison, the traditional five-point algorithm [24] is incorporated, in which we extract RootSIFT [2] features and use brute force matching strategy to compute essential matrix and decompose it into relative pose. Since the five-point method cannot estimate the scale, all relative translations are set to unit vectors.

The evaluation result of the relative pose is shown in Tab. 1. For the sake of brevity, we have excluded the results in 360VO-Seq4 and 360VO-Seq9, which are available in the supplementary material. Compared to the learning-based methods, the traditional five-point algorithm has more advantages in estimating relative rotation and translation direction. For both RRE and RTAE, the five-point algorithm consistently achieves the best or second-best results, except for the Building dataset, where it exhibits higher errors in relative translation angle estimation. These advantages in accuracy can be attributed to the omnidirectional field of view (FoV) of the panoramic image and the precise RootSIFT feature matching. Due to the omnidirectional FoV, even if a part of the image is occluded, relative pose estimation can be performed relying on feature points in other areas. In contrast, network-based methods estimate a mask to deal with occlusion areas, which is unreliable in some cases and can yield suboptimal results in certain scenarios. In the datasets used for experiments, most scenes exhibit rich textures, leading to accurate relative pose estimation of the five-point method. However, in the synthetic indoor environment of the PanoSUNCG dataset, which includes textureless areas, the five-point method exhibits reduced performance compared to our PanoPose. In Mapillary Metropolis datasets, the five-point method also achieves the lowest relative scale error. This is because the images in this dataset are regularly sampled at intervals of 6 meters. The relative translation scale obtained by the five-point method is fixed at 1 meter (unit vector), a low error can be obtained by multiplying the scaling factor of 6.

Our PanoPose outperforms other self-supervised methods in most datasets, except for the relative rotation estimation on Building and mean RTAE on Campus. On 360VO-

Dataset	TA method	Init Rel Pose	Mean ARE(°)	Med ARE(°)	Mean ATE(m)	Med ATE(m)
360VO-Seq3	BATA[43]	Five-point	12.4805	7.9057	2.6858	2.1478
		PanoPose	13.4524	9.9413	2.2941	1.8877
	LUD[25]	Five-point	**12.4512**	7.8862	2.7598	2.3717
		PanoPose	13.4209	9.9148	2.1823	1.9502
	L2IRLS	Five-point	13.2952	**7.3977**	3.1602	2.4316
		PanoPose	13.2376	9.4842	**1.9853**	**1.3688**
360VO-Seq6	BATA[43]	Five-point	11.1983	7.3028	1.6403	1.6572
		PanoPose	13.5741	10.6381	1.2708	1.3216
	LUD[25]	Five-point	11.0981	**7.2018**	1.7205	1.6399
		PanoPose	13.5074	10.6592	1.1363	1.2586
	L2IRLS	Five-point	**11.0658**	7.2726	2.8248	2.2934
		PanoPose	13.4502	8.9259	**0.9815**	**0.9771**
Building	BATA[43]	Five-point	0.4963	0.4952	3.5968	3.1559
		PanoPose	2.7754	2.8502	3.0632	2.5502
	LUD[25]	Five-point	0.4914	0.4872	3.8921	3.3175
		PanoPose	2.8346	2.9134	3.1973	2.7261
	L2IRLS	Five-point	**0.4895**	**0.4915**	4.1518	3.4175
		PanoPose	2.7865	2.9091	**2.6129**	**2.0475**

Table 2. Global pose evaluation result on different datasets. We demonstrate the mean and median absolute rotation error (ARE) and absolute translation error (ATE). The best results are shown in **bold**.

Seq1 and PanoSUNCG, our method significantly surpasses the comparative self-supervised methods. Compared with the best results of the comparative method, our PanoPose reduces the mean RRE, median RRE, mean RTAE, and median RTAE by 87%, 61%, 81%, and 36%. Focusing on the scale error, PanoPose outperforms other self-supervised methods on median RSE in all datasets. As for the mean RSE, the proposed PanoPose achieves the best result except for the PanoSUNCG dataset, where certain image pairs exhibit significant scale errors, contributing to an overall higher average error.

4.4. Global Pose Estimation

To further validate the effectiveness of PanoPose, we employ the relative poses estimated by PanoPose and the traditional five-point method as initial values and combine them with different translation averaging methods. Subsequently, we calculate the absolute error between the estimated poses and the ground truth. We use [5] as the rotating averaging method. For the translation averaging method, BATA[43], LUD[25], and L2IRLS (Eq. (7)) are chosen. Notably, we do not consider [4] and [22] as they require feature point correspondence, while PanoPose only provides relative pose. The results are summarized in Tab. 2.

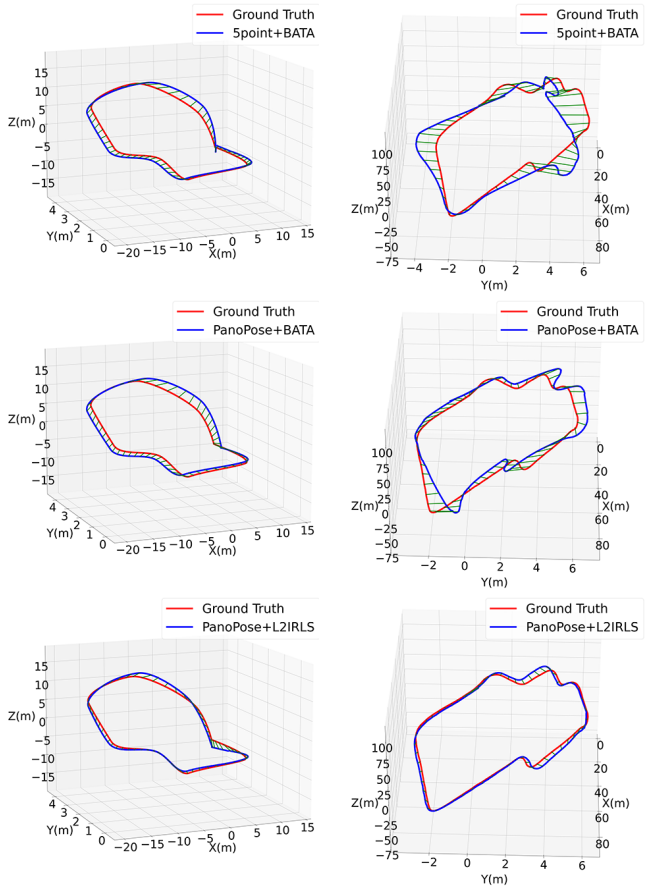
As can be seen from the table, the choice of translation averaging methods has little impact on the final rotation error (Mean ARE and Med ARE). Moreover, when using the five-point method, the absolute rotation tends to be more accurate, owing to the precise estimation of relative rotation between image pairs.

Focusing on the absolute translation error, using our network for estimating relative poses outperforms the five-point method. Across the experimental datasets, changing the initial relative pose estimation from the five-point algorithm to PanoPose reduces the mean ATE and median ATE by 29% and 27% respectively. The combination of our network and the proposed L2IRLS translation averaging strategy significantly outperforms others in terms of ATE. This superior performance can be attributed to L2IRLS's direct optimization of the relative translation scale, which is provided by PanoPose with relatively high precision. Conversely, when the five-point algorithm is paired with L2IRLS, the resulting ATE is the highest. This is because the traditional method cannot estimate the scale, resulting in a large initial scale error received by L2IRLS, which in turn leads to performance degradation. On the other hand, BATA and LUD do not use scale as an optimization variable, thus avoiding scale-related errors.

To provide a qualitative comparison of the various combinations of relative pose estimation methods and translation averaging strategies, we visualize the estimated camera global pose in Fig. 2. Comparing the first row and second row of Fig. 2, it is clear that transitioning from the traditional five-point method to PanoPose results in more accurate pose estimation. The third row is the result generated by our network and L2IRLS, and it has the best alignment with the ground truth.

4.5. Ablation Studies

With the same conditions, we validate the key components of our network by conducting an ablation study on the experiment datasets, and the results are summarized in Tab. 3. For the sake of brevity, only the average value of the relative pose error is reported.



(a) 360VO-Seq3 (b) Building

Figure 2. Global pose estimation result on different datasets. Ground truth is shown in red and the estimated camera trajectory is blue. The first row is the result of the five-point and BATA. The second row is our PanoPose and BATA. The last row is PanoPose and the proposed L2IRLS.

From Tab. 3, we can observe that the rotation-only pre-training has a significant improvement in relative rotation estimation. Employing this strategy leads to a 60% reduction in the RRE. The depth fusion block can improve the accuracy of relative translation scales. Across the experimental datasets, utilization of this module resulted in an RSE reduction of 22%, 44%, and 73% respectively.

Since the block is inserted in the process of relative translation estimation, theoretically, it also influences translation direction estimation. From the experiment result, we observe that the fusion block has a positive effect on translation direction estimation in 360VO-Seq4 and PanoSUNCG datasets while leading to worse results on Campus. When the pre-training strategy and fusion module are used together, the relative pose estimation accuracy is greatly improved, especially the relative rotation and relative translation scales. However, the improvement in the relative translation direction is smaller. This is because the two key modules we proposed focus on rotation and scale respectively.

Dataset	Rot	Fusion	RRE	RTAE	RSE
360VO-Seq4			0.1795	0.6924	0.2317
	✓		0.0687	0.5841	0.2428
		✓	0.1716	0.5209	0.1806
	✓	✓	**0.0656**	**0.4809**	**0.1770**
Pano-SUNCG			0.8459	0.8492	3.0462
	✓		0.2237	0.5773	3.3820
		✓	0.7114	0.6891	1.6933
	✓	✓	**0.1559**	**0.4253**	**1.2295**
Campus			0.2476	2.6856	0.8419
	✓		0.1404	3.1764	1.8562
		✓	0.2599	2.8506	**0.2314**
	✓	✓	**0.1094**	**2.2683**	0.2563

Table 3. Ablation study of the proposed rotation-only pre-training strategy and fusion block, which are represented by Rot and Fusion, respectively. The best result is shown in **bold**.

5. Conclusion

In this paper, we propose PanoPose, a fully self-supervised network for panoramic image relative pose estimation. Our PanoPose is composed of a depth-net and a pose-net to estimate the dense depth map and relative pose, and the main supervision is the photometric loss between the reference image I_r and the reconstructed image I'_r. To further improve pose estimation accuracy, we add a fusion block to leverage depth-net information, a rotation-only pre-training strategy, and pose consistency loss. Since PanoPose is a self-supervised method, in practical applications, it can be trained directly on the target dataset, which greatly improves its scope of application. PanoPose can also be applied to normal pinhole images by removing the rotation-only pretraining stage, which utilizes the inherited geometry of panoramic images.

Limitions. Despite its advancements, PanoPose still has its drawbacks. Because the network uses transformer-based Croco as the backbone, the amount of calculation is relatively large, which affects the efficiency of training and inference. Additionally, the cross-dataset generalization capabilities of PanoPose are still limited, which is also a key issue for the learning-SfM method to become practical.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (No. U22B2055, U23A20386, 62273345 and 62073320), the Beijing Natural Science Foundation (No. L223003), the Key R&D Project in Henan Province (No. 231111210300), and the Open Fund of Wuhan University-Huawei Geoinformatics Innovation Laboratory Under Grant TC20210901025.

References

[1] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *IEEE International Conference on Computer Vision (ICCV)*, pages 37–45, 2015. 2

[2] Relja Arandjelović and Andrew Zisserman. Three things everyone should know to improve object retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2911–2918, 2012. 6

[3] Romain Brégier. Deep regression on manifolds: a 3d rotation case study. In *International Conference on 3D Vision (3DV)*, pages 166–174, 2021. 3

[4] Qi Cai, Lilian Zhang, Yuanxin Wu, Wenxian Yu, and Dewen Hu. A pose-only solution to visual reconstruction and navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 45(1):73–86, 2021. 3, 7

[5] Avishek Chatterjee and Venu Madhav Govindu. Robust relative rotation averaging. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(4):958–972, 2017. 2, 5, 7

[6] Kefan Chen, Noah Snavely, and Ameesh Makadia. Wide-baseline relative camera pose estimation with directional learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3258–3268, 2021. 2

[7] Hainan Cui, Shuhan Shen, Xiang Gao, and Zhanyi Hu. Batched incremental structure-from-motion. In *International Conference on 3D Vision (3DV)*, pages 205–214, 2017. 1

[8] Hainan Cui, Shuhan Shen, and Zhanyi Hu. Tracks selection for robust, efficient and scalable large-scale structure from motion. *Pattern Recognition*, 72:341–354, 2017. 1

[9] Philipp Dufter, Martin Schmitt, and Hinrich Schütze. Position information in transformers: An overview. *Computational Linguistics*, 48(3):733–763, 2022. 4

[10] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *IEEE International conference on Computer Vision (ICCV)*, pages 3828–3838, 2019. 1, 3, 4, 5, 6

[11] Venu Madhav Govindu. Combining two-view constraints for motion estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages II–II, 2001. 2

[12] Huajian Huang and Sai-Kit Yeung. 360vo: Visual odometry using a single 360 camera. In *International Conference on Robotics and Automation (ICRA)*, pages 5594–5600, 2022. 5

[13] Will Hutchcroft, Yuguang Li, Ivaylo Boyadzhiev, Zhiqiang Wan, Haiyan Wang, and Sing Bing Kang. Covispose: Covisibility pose transformer for wide-baseline relative pose estimation in 360° indoor panoramas. In *European Conference on Computer Vision (ECCV)*, pages 615–633, 2022. 2

[14] Dinesh Jayaraman and Kristen Grauman. Learning image representations tied to ego-motion. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1413–1421, 2015. 2

[15] Nianjuan Jiang, Zhaopeng Cui, and Ping Tan. A global linear method for camera pose registration. In *IEEE International Conference on Computer Vision (ICCV)*, pages 481–488, 2013. 3

[16] Zhigang Jiang, Zhongzheng Xiang, Jinhua Xu, and Ming Zhao. Lgt-net: Indoor panoramic room layout estimation with geometry-aware transformer network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1654–1663, 2022. 2

[17] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2938–2946, 2015. 2

[18] Ruihao Li, Sen Wang, Zhiqiang Long, and Dongbing Gu. Undeepvo: Monocular visual odometry through unsupervised deep learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 7286–7291, 2018. 2

[19] Yuyan Li, Zhixin Yan, Ye Duan, and Liu Ren. Panodepth: A two-stage approach for monocular omnidirectional depth estimation. In *International Conference on 3D Vision (3DV)*, pages 648–658, 2021. 2

[20] Liqiang Lin, Yilin Liu, Yue Hu, Xingguang Yan, Ke Xie, and Hui Huang. Capturing, reconstructing, and simulating: the urbanscene3d dataset. In *European Conference on Computer Vision (ECCV)*, pages 93–109, 2022. 5

[21] Mengyi Liu, Shuhui Wang, Yulan Guo, Yuan He, and Hui Xue. Pano-sfmlearner: Self-supervised multi-task learning of depth and semantics in panoramic videos. *IEEE Signal Processing Letters*, 28:832–836, 2021. 2, 3

[22] Lalit Manam and Venu Madhav Govindu. Correspondence reweighted translation averaging. In *European Conference on Computer Vision (ECCV)*, pages 56–72, 2022. 3, 7

[23] Pierre Moulon, Pascal Monasse, and Renaud Marlet. Global fusion of relative motions for robust, accurate and scalable structure from motion. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3248–3255, 2013. 1, 3

[24] David Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 26(6):756–770, 2004. 1, 2, 6

[25] Onur Ozyesil and Amit Singer. Robust camera location estimation by convex programming. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2674–2683, 2015. 7

[26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems (NeurIPS)*, 32, 2019. 5

[27] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016. 1

[28] Zhijie Shen, Chunyu Lin, Kang Liao, Lang Nie, Zishuo Zheng, and Yao Zhao. Panoformer: Panorama transformer for indoor 360° depth estimation. In *European Conference on Computer Vision (ECCV)*, pages 195–211, 2022. 2, 3

[29] Noah Snavely, Steven M. Seitz, and Richard Szeliski. Photo tourism: Exploring photo collections in 3d. In *ACM SIGGRAPH 2006 Papers*, pages 835–846, 2006. 1

[30] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene comple-

tion from a single depth image. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1746–1754, 2017. 5

[31] Cheng Sun, Chi-Wei Hsiao, Min Sun, and Hwann-Tzong Chen. Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1047–1056, 2019. 2

[32] Benjamin Ummenhofer, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox. Demon: Depth and motion network for learning monocular stereo. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5038–5047, 2017. 2

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017. 3

[34] Fu-En Wang, Hou-Ning Hu, Hsien-Tzu Cheng, Juan-Ting Lin, Shang-Ta Yang, Meng-Li Shih, Hung-Kuo Chu, and Min Sun. Self-supervised learning of depth and camera motion from 360 videos. In *Asian Conference on Computer Vision (ACCV)*, pages 53–68, 2018. 3, 5

[35] Fu-En Wang, Yu-Hsuan Yeh, Yi-Hsuan Tsai, Wei-Chen Chiu, and Min Sun. Bifuse++: Self-supervised and efficient bi-projection fusion for 360 depth estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 45(5):5448–5460, 2022. 2, 5, 6

[36] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing (TIP)*, 13(4):600–612, 2004. 4

[37] Philippe Weinzaepfel, Vincent Leroy, Thomas Lucas, Romain Brégier, Yohann Cabon, Vaibhav Arora, Leonid Antsfeld, Boris Chidlovskii, Gabriela Csurka, and Jérôme Revaud. Croco: Self-supervised pre-training for 3d vision tasks by cross-view completion. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:3502–3516, 2022. 1, 3

[38] Kyle Wilson and Noah Snavely. Robust global translations with 1dsfm. In *European Conference on Computer Vision (ECCV)*, pages 61–75, 2014. 1

[39] Zhichao Yin and Jianping Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1983–1992, 2018. 2

[40] Ilwi Yun, Hyuk-Jae Lee, and Chae Eun Rhee. Improving 360 monocular depth estimation via non-local dense prediction transformer and joint supervised and self-supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3224–3233, 2022. 5, 6

[41] Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose: Predicting probabilistic relative rotation for single objects in the wild. In *European Conference on Computer Vision (ECCV)*, pages 592–611, 2022. 2

[42] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1851–1858, 2017. 1, 2, 4, 5, 6

[43] Bingbing Zhuang, Loong-Fah Cheong, and Gim Hee Lee. Baseline desensitizing in translation averaging. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4539–4547, 2018. 1, 3, 7

[44] Chuanqing Zhuang, Zhengda Lu, Yiqun Wang, Jun Xiao, and Ying Wang. Spdet: Edge-aware self-supervised panoramic depth estimation transformer with spherical geometry. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 45(10):12474–12489, 2023. 3