Flow-Guided Online Stereo Rectification for Wide Baseline Stereo

Anush Kumar[1] Fahim Mannan[1] Omid Hosseini Jafari[1] Shile Li[1] Felix Heide[1, 2]
[1]Torc Robotics [2]Princeton University

Abstract

Stereo rectification is widely considered "solved" due to the abundance of traditional approaches to perform rectification. However, autonomous vehicles and robots in-the-wild require constant re-calibration due to exposure to various environmental factors, including vibration, and structural stress, when cameras are arranged in a wide-baseline configuration. Conventional rectification methods fail in these challenging scenarios: especially for larger vehicles, such as autonomous freight trucks and semi-trucks, the resulting incorrect rectification severely affects the quality of downstream tasks that use stereo/multi-view data. To tackle these challenges, we propose an online rectification approach that operates at real-time rates while achieving high accuracy. We propose a novel learning-based online calibration approach that utilizes stereo correlation volumes built from a feature representation obtained from cross-image attention. Our model is trained to minimize vertical optical flow as proxy rectification constraint, and predicts the relative rotation between the stereo pair. The method is real-time and even outperforms conventional methods used for offline calibration, and substantially improves downstream stereo depth, post-rectification. We release two public datasets (https://light.princeton.edu/online-stereo-rectification/), a synthetic and experimental wide baseline dataset, to foster further research.

1. Introduction

Wide baseline stereo methods with cameras separated meters apart have been proposed as a low-cost depth sensing method [5, 40, 41, 49] that allows, with double-digit megapixel resolutions, even long distance depth measurements beyond 100 meters range. Employed in autonomous trucks and large construction, farming robots, or UAVs [23], the mounted cameras experience significant vibrations which propagate to the stereo sensors leading to large deviations from offline calibration. Therefore, accurate online stereo calibration is essential for the functionality of these sensor systems as part of the autonomous decision-making stack, especially for large long-haul vehicles where the mounting structures for large baselines can stretch and twist due to temperature and stress gradients. The quality of the calibration, addressed by the proposed method (see Fig. 1), is essential to downstream tasks that aim to understand the environment around the vehicle, including stereo depth estimation, 3D object detection, semantic segmentation, and SLAM.

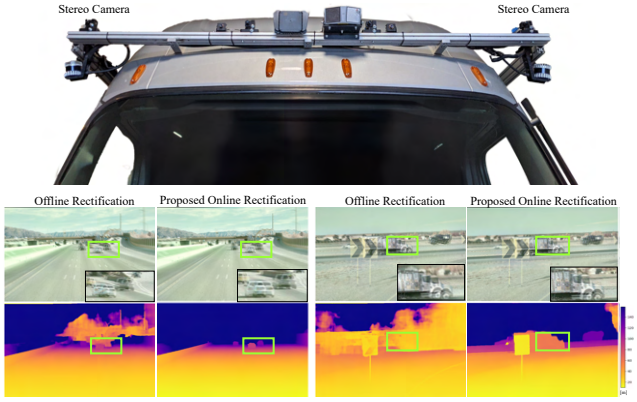

Figure 1. **Wide Baseline Stereo Online Calibration.** We address online calibration in wide baseline stereo setups mounted above the cab of a semi-truck with 2 meters baseline (top). Due to vibrations in a trucking scenario, the offline calibrations can be incorrect during a drive. We overlay rectified stereo pairs (middle) and the corresponding depth inferred by HITNet [51] (bottom), each column pair is an individual example. The proposed flow-guided online calibration allows for accurate stereo depth estimation in such a wide baseline setup. Please zoom in for details.

Since images exist in 2D space, stereo co-planarity enables efficient stereo matching and stereo depth estimation by nullifying the disparity in one of the axes, thereby reducing the matching search space to 1D. While perfect stereo co-planarity is difficult to achieve and maintain, stereo rectification aims to project a pair of images onto a common image plane such that there is no vertical disparity between corresponding pixels across the images.

Existing calibration and rectification methods typically rely on keypoint extraction and description methods [12, 53] using hand-crafted [4, 7, 30, 31, 35, 46, 47] and

learning-based [43, 48, 50] features. Stereo rectification can be addressed using such traditional calibration approaches. A line of work [2, 17, 20–22, 37] formulates the rectification constraint using epipolar geometry, and they rely on existing optimization techniques like Gradient Descent and Levenberg-Marquardt to arrive at a rectified stereo pair. Other approaches [19, 32, 34, 57] focus on improving feature extraction and matching techniques, fundamental matrix computation and extrinsics estimation. The majority of these methods [42] are not computationally permissive to deploy in an online setting due to expensive optimization and intermediary steps. Those that can be deployed online compromise on accuracy resulting in significantly deteriorated downstream performance. Dang *et al*. [13] extensively analyzes the error propagation from poor extrinsic to downstream tasks like 3D reconstruction.

As a result of these limitations, today's accurate stereo calibration methods require a separate calibration step that uses visual patterns with known geometry to determine the intrinsics and extrinsics of the setup. These approaches mandate calibration in an offline setting, and it ignores environmental effects experienced by the sensors when in use. During inference in the wild, this significantly reduces accuracy and performance for downstream tasks.

In our work, we propose an online rectification process, that can periodically re-rectify the sensors to address the poor calibration quality. To this end, we propose an online stereo pose estimation model that utilizes a correlation volume to determine relative pose between two cameras. We use a transformer encoder to produce strong feature representations built from global context and cross-view context. Our model is trained with weak supervision and a proxy rectification constraint which is computed in a self-supervised fashion. We use vertical optical flow to interpret the degree of vertical disparity and train the model to minimize it. Additionally, we find a 40% improvement in SIFT[35] and SuperGlue[48] keypoint-offset metric, when our novel self-supervised vertical flow constraint is employed. We further validate the method by measuring the effect of rectification quality on state-of-the-art stereo depth estimation models (DLNR [60], HITNet [51]). We improve MAE by 63% and 51% for depth estimation downstream using DLNR [60] and HITNet [51], when evaluated on our real-world Semi-Truck Highway dataset and KITTI dataset[18], respectively. We also validate the effectiveness of the method on a synthetic Carla dataset with artificially induced severe perturbations that rarely occur in real-world captures. We make the following contributions.

- We provide a Semi-truck Highway driving dataset and a Carla dataset that capture calibration deterioration in real and synthetic settings, respectively.
- We propose a novel learning-based stereo calibration model that utilizes stereo correlation volume to infer relative pose between a stereo pair.
- We introduce a self-supervised vertical optical flow loss to train our model without the need for high-quality offline extrinsics.
- The proposed approach outperforms all tested existing methods in the keypoint-offset metric and on downstream stereo depth estimation on real data while deployable in an online real-time setting.

2. Related Work

Traditional Stereo Rectification Traditional approaches to stereo rectification usually focus on computing rectification homographies without prior knowledge of the extrinsics of the cameras [2]. The homographies are directly computed by optimizing on a formulation of the rectification constraint [21, 22, 37]. Although computationally costly, Rehder *et al*. [42] target bundle adjustment as a strategy to refine existing rectification quality. Hartley [22] propose to determine an arbitrary homography matrix for one of the cameras such that its epipole is pushed to infinity. Gluckman *et al*. [20] propose an optimization strategy that minimizes undersampling/oversampling of pixels after rectification. Fusiello et al. [17] optimizes for rectification homographies by minimizing the first-order approximation of the reprojection error termed Sampson error. Several methods [32, 34, 57] investigate decomposing the intermediate extrinsics and homographies into simpler transformations to optimize them individually.

A parallel direction focuses on the steps leading up to rectification, such as better establishing correspondences between views or improvements to fundamental matrix or extrinsics computation [55]. Georgiev *et al*. [19] remove erroneous or noisy matches using various filters, and Zilly *et al*. [61] break down the computation of the fundamental matrix into its Taylor-series expansion. Finally, Dang *et al*. [13] explore continuous online calibration for wide baseline scenarios and evaluate the effect of poor calibration on downstream tasks. While accurate, all of the above methods are computationally expensive and prohibit real-time calibration. The proposed method is real-time while even reducing error of offline calibration methods.

Learning-based Stereo Rectification Recently, learned methods attempt to circumvent errors in rectification. Li *et al*. [29] propose a strategy of both 1D search and 2D kernel-based search for depth estimation from stereo images and Luo *et al*. [36] address stereo rectification in laparoscopic images by introducing an intermediate vertical correction module to provide pixel-wise vertical corrections. Ji *et al*. [25] use a rectification network to predict rectification homographies for the task of view synthesis. Zhang *et al*. [59] employ a 4-point parametrization of the rectification homography, which is estimated by a dedicated subnetwork

and for downstream stereo matching. Wang *et al*. [56] focus solely on rectification alone using learned feature extractors. All of these methods propose refinement approaches to rectification instead of directly predicting rectification. We hope that the datasets introduced in this work can facilitate research towards rectification as a standalone task.

Learning-based Camera Pose Estimation Stereo rectification is also tangentially related to relative pose estimation between multi-view images. Given two overlapping views, learning to estimate the relative rotation and translation between the views results in us knowing the extrinsics between the two views. PoseNet [27] was one of the earliest attempts to perform end-to-end camera pose estimation using CNNs. Several works build on PoseNet [16, 26, 28, 38] by introducing elements to balance the losses from translation and rotation and architecture variations. Recently, DirectionNet [10] explores optimizing quaternions as continuous representations for rotations. Other works incorporate flow to refine pose estimates from a coarse initialization [37, 39]. These methods warp source images onto the target frame, while we directly optimize vertical flow which implicitly captures the rectification quality.

Several methods explore incorporating learned components into existing traditional approaches, for example DSAC [6] is a differentiable RANSAC method. Similarly Ling *et al*. [32] proposes a learned model to estimate the fundamental matrix given feature matches and descriptors from SuperGlue [48]. Rockwell *et al*. [44] propose using ViT [15] to estimate cross attention weights between stereo image patches and then finally regress pose by directly estimating the orthonormal bases obtained from SVD decomposition of the Essential Matrix. Roessle *et al*. [45] propose a graph model that builds features correspondences across multi-view images inspired by SuperGlue [48] followed by a coarse pose estimation using the 8-point algorithm which is further refined using a bundle adjustment step. Arnold *et al*. [1] also explore using 2D-2D, 3D-3D correspondences in loop with learned depth estimation models and cost-volume based approaches for pose estimation. These existing pose estimation methods do not optimize the rectification constraint but rather optimize with 3D pose or epipolar constraints. We find that applying pose estimations methods to rectification results in poor performance, see Sec. 5.

3. Flow-Guided Online Rectification

In this section, we present the online rectification model and flow-guided training approach. Local feature-based pose estimation methods can perform poorly due to rolling-shutter effects coupled with mechanical vibrations on highway trucking scenarios. To address these challenges, we predict pose directly from a stereo pair utilizing cross-attentional image features and stereo cost volume to optimize for pose. Furthermore, for real data we are only able to perform static offline calibration which does not hold under highway driving scenarios due to strong vibration. To this end, we rely on self-supervised vertical flow loss and use offline calibration as a semi-supervised rotation loss. As poor rotation can have a high impact on downstream tasks [13] and the stereo baseline is fixed, our model learns to estimate the relative rotation from a stereo pair.

Fig.2 shows the model architecture and training process. The model operates on a pair of images which are fed to a shared CNN (Fig.2a) to extract shift-equivariant features. The features are then rectified using a prior pose estimate which can either be from previous estimation or set to identity. This is followed by our feature enhancement step (Fig.2b), which comprises of a positional embedding step and a transformer encoder. The transformer encoder captures global information across both views using self-attention and cross-attention. Next, we employ a correlation volume, Fig.2(c), to establish matches based on the features extracted from the transformer encoder. The volume represents a match distribution for every pixel in the feature map. The correlation volume is then processed by a decoder to implicitly learn to discern noisy matches and predict a simplified rotation estimate, which undergoes further processing to produce the final relative rotation prediction. Given our rotation prediction, we rectify the input images and estimate the optical flow. With this flow estimate in hand, we minimize the vertical flow, Fig.2(e).

3.1. Feature Extraction

Given a set of stereo pairs I_l and $I_r \in \mathbb{R}^{H\times W\times 3}$, we encode these images using a weight-shared CNN backbone. The backbone comprises of 3 residual convolution blocks, resulting in a pair of feature maps f_l, $f_r \in \mathbb{R}^{h\times w\times c}$. These features are limited in capturing globally discriminative features, ultimately, due to the fact that convolution operations are local in nature. While global representations are important to reduce ambiguous matches, lower dimensional feature representations lend to compute and memory advantages in the following stages. The proposed method extracts global features while offering real-time runtime.

3.2. Positional Feature Enhancement

We take inspiration from transformer models [54], and employ positional encoding, specifically the 2D sine and cosine positional encoding (same as DETR [8]). The positional encoding is directly added to the CNN feature maps f_l, f_r, adding an extra layer of spatial information in addition to feature similarity during matching. This helps the model match features more consistently to estimate relative pose which is after all a spatial mapping of these features. We encode

$$p_{encoding} = [\sin(pos/C^{2k/d}), \cos(pos/C^{2k/d})] \quad (1)$$

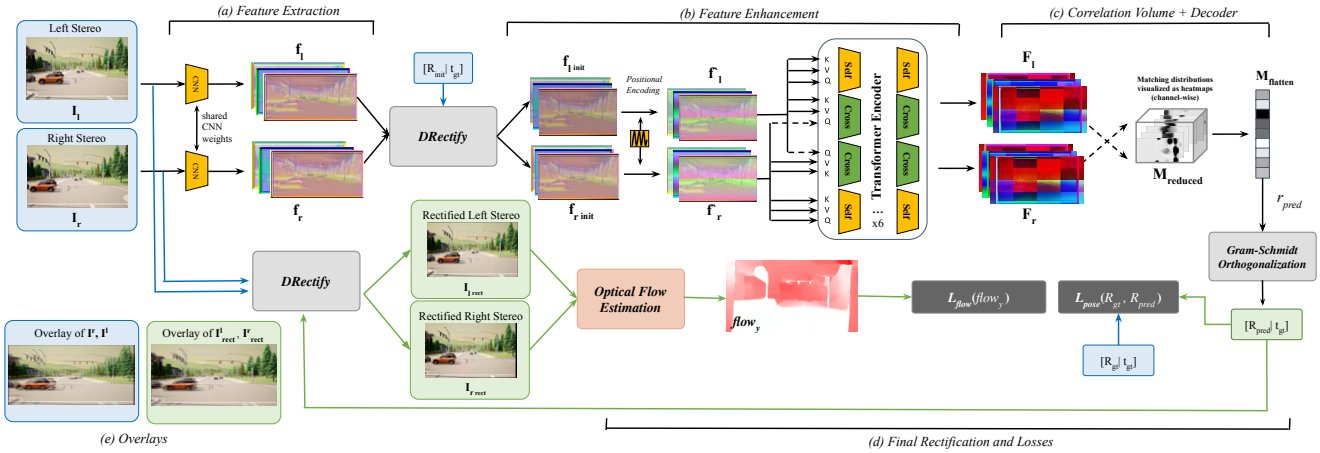

Figure 2. **Flow-Guided Online Rectification**. In Section 3 we go over the motivation and implementation of these sub-modules in our model. (a) Feature Extraction in Section 3.1, (b) Feature Enhancement in Section 3.2, and (c) Correlation Volume, Decoder and (d) Loss calculation in Section 3.5, 3.6. (e) The overlays are a simple illustration to understand the rectification quality, this is generated by a weighted addition of the stereo pairs.

$$f^{'}_{l_{2k}} = f_{l_{2k}} + p_{encoding}, \quad f^{'}_{r_{2k}} = f_{r_{2k}} + p_{encoding}, \quad (2)$$

where $C = 10000, d = 64, pos \in w \times h, k \in c$. Following, we apply a feature enhancement transformer. The encoder comprises of 6 self-attention blocks and 6 cross-attention blocks and a feed-forward network [54]. The keys, queries and values come from the same feature map in self-attention phase, while the cross-attention maps are estimated across the two features specifically keeping the key-value pairs from one set of features and querying from the other feature map [58]. We further enforce fine-grained global matching by computing the attention maps across the entire feature map as opposed to the windowed approach [58] using SWIN transformers [33],

$$F_l = T(K = f^{'}_l, V = f^{'}_l, Q = f^{'}_r), \quad (3)$$

$$F_r = T(K = f^{'}_r, V = f^{'}_r, Q = f^{'}_l). \quad (4)$$

Here, T is the transformer encoder and K, V, Q correspond to the Key, Value and Query inputs. The enhanced features F_l, F_r are used for the steps discussed in Section 3.4.

We perform the positional feature enhancement mentioned above after an initial rectification step, R_{init}, i.e.,

$$f_{l_{init}}, f_{r_{init}} = DRectify(f^{'}_l, f^{'}_r, P = [R_{init}|t]) \quad (5)$$

Here, R_{init} can either be identity (as used in our experiments) or rotation from previous estimation. Subsequently, F_l, F_r are computed from $f_{l_{init}}, f_{r_{init}}$ following the steps described in this section. This enables us to standardize the variation in pose to a certain extent, and further allows us to formulate the model predictions as a correction applied to R_{init} as discussed in Section 3.5

3.3. Differentiable Rectification

A crucial component in our model is the differentiable rectification module *DRectify*, enabling us to train end-to-end while inferring rectified images from the model pose predictions, which, in turn, enables us to add rectification constraints to the model during training. To define this operator, we assume two images $I_1, I_2 \in \mathbb{R}^{H \times W \times 3}$ and the relative pose between the sensors as $P = [R|t] \in SE(3)$ and intrinsics $K_1, K_2 \in \mathbb{R}^{3 \times 3}$, and we aim to project I_1, I_2 onto a common image plane using rectification rotations R_1, R_2 resulting in images $I_{1_{rect}}, I_{2_{rect}}$. We divide this as follows,

Estimating Rectification Rotations. Given relative pose information P this step involves computing rotations R_1, R_2 for each image I_1, I_2 using the horizontal baseline assumption [22].

Rectifying Images. Given rectification rotation R_1, R_2 for I_1, I_2, we reproject the image using these rotations and use differentiable grid-sampling [24] to sample the location of the new pixels resulting in $I_{1_{rect}}, I_{2_{rect}}$.

Detailed mathematical formulations for these two steps are described in our Supplementary Material.

3.4. Correlation Volume

We use a correlation volume to allow for global feature matching across the enhanced feature maps. We flatten both feature maps $F_l, F_r \in \mathbb{R}^{H \times W \times C}$ along $H \times W$. We then compute the correlation as follows [58],

$$M_{vol} = F_l(F_r)^T / \sqrt{C}, \in \mathbb{R}^{H \times W \times H \times W}. \quad (6)$$

This correlation volume computed across the flattened feature maps implicitly represents the matches across F_l and F_r. To further simplify this representation we apply a *softmax* along the last two dimensions of M_{vol},

$$M_{match} = softmax(M_{vol}). \quad (7)$$

M_{match} represents the likelihood of a match for a specific location in F_l to all locations in F_r. We let the model learn reliable and un-reliable matches by further processing M_{match} in the decoder as follows.

3.5. Decoder and Final Rectification

The decoder layers are comprised of a combination of 6x 3D Convolution and Average Pooling layers, the result, $M_{reduced} \in \mathbb{R}^{H \times W}$, representing the most likely matches from the distribution encoded into M_{match}. We then flatten $M_{reduced}$ to produce $M_{flatten}$ which is sent to our final linear layer to predict the relative rotation, r_{pred}. Since our model attempts to directly predict elements in the rotation matrix, the *Tanh* operator helps keep the predictions bounded $[-1, 1]$ and therefore stable. We have,

$$r_{correction} = Tanh(Linear(M_{flatten})) \in \mathbb{R}^{6\times 1}. \quad (8)$$

Borrowing from the extensive analysis from Chen et al. [10], we choose to use the 6D representation which focuses on predicting the x and y columns of the rotation matrix. This step is followed by Gram-Schmidt orthogonalization. Given the x and y columns as r_x, r_y,

$$r_{x_{norm}} = \frac{r_x}{||r_x||}, r_z = r_{x_{norm}} \times r_y, \quad (9)$$

$$r_{z_{norm}} = \frac{r_z}{||r_z||}, r_y = r_{z_{norm}} \times r_{x_{norm}}, \quad (10)$$

$$R_{GS} = [r_{x_{norm}}, r_y, r_{z_{norm}}] \in SO(3). \quad (11)$$

Next, we extract the x and y columns from R_{init} as $r_{init} \in \mathbb{R}^{6\times 1}$ and find that

$$r_{pred} = r_{init} + r_{correction} \quad (12)$$

$$R_{pred} = GramSchmidt(r_{pred_x}, r_{pred_y}) \quad (13)$$

is a valid rotation matrix, which can be used to rectify our input stereo pair I_l, I_r as

$$I_{l_{rect}}, I_{r_{rect}} = DRectify(I_l, I_r, [R_{pred}|t]), \quad (14)$$

where t is reused from ground truth pose information.

3.6. Training

Next, we describe the two main loss functions used to train our model. We carry over the notation from the previous section for convenience. The complete loss function is

$$L = \lambda_1 L_{rot} + \lambda_2 L_{flow}, \quad (15)$$

where $\lambda_1, \lambda_2 = 10, 0.1$ are scalar weights. Here L_{rot} is a pose loss supervised on ground truth calibration data, and L_{flow} is the self-supervised vertical-flow loss.

Self-supervised Vertical Flow Loss. Given our rectified image pairs, we use RAFT [52] pre-trained on KITTI [18] to infer the flow in the x-axis and y-axis. This loss function is self-supervised and is stable given the model predicts stable/valid rotation estimates. Since optical flow is an indirect method to establish dense correspondences across images, we are able to leverage this flow to implicitly add a rectification constraint. Since the goal of rectification is to nullify the disparity along the vertical axes, the vertical flow component contains information about the presence of vertical disparity. Hence our loss function component is designed as follows.

$$L_{flow} = \frac{1}{N} \sum_{i=1}^{N} |flow_y|, \quad (16)$$

with $flow_y = RAFT(I_{l_{rect}}, I_{r_{rect}})$ and N is the total number of pixels in the image. Interestingly, this loss function is applicable to rectify vertical stereo setups as well, here the goal would be to minimize $flow_x$ rather than $flow_y$.

Rotation Loss. We also employ a second supervised loss using a ground truth estimate of the rotation matrix R_{gt}. We use an ℓ_1 loss, yielding

$$L_{rot} = \|R_{gt} - R_{pred}\|_1 \quad (17)$$

The loss plays a crucial role in the early stages of training, allowing the model to narrow down the possible rotation estimates in the early training stages until reasonably stable rotation predictions are obtained and the self-supervised flow loss from above dominates.

3.7. Implementation

We implement our model on the PyTorch framework including *DRectify()*. The models are trained for 80-140 epochs depending on the dataset and the degree of misalignment between the images. We use the Adam Optimizer with a LR of $1e^{-4}$ coupled with exponential decay. We apply brightness, contrast, and color perturbations to the data as augmentations and interchange the left-right stereo pairs as an additional augmentation. We train on two NVIDIA-A40 GPUs with a batch size of 16. Our input resolution is 1024x512 pixels, while the main model is trained on one GPU we run flow estimation on the second GPU which is running inference on images of resolution 512x256. All inference time benchmarks in Table 3 are preformed on the NVIDIA-A40 GPU with a batch size of 1.

4. Stereo Rectification Datasets

The following section describes the real-world and synthetic datasets we use to train and evaluate the method. We note the high scene diversity and the task difficulty associated with both datasets, see Supplemental Document for details. Fig. 3 reports samples of the dataset and Table 1 lists the train-test splits. Both datasets contain samples of data from different sequences/recordings, which in turn means every frame in the dataset is a unique frame.

Figure 3. **Rectification Dataset Examples**. We introduce two new datasets. A first **Semi-Truck Highway dataset** (top three rows) is a wide baseline dataset captured on US highways. We also experiment with a synthetic **Carla dataset**, with rare and more extreme pose variations than even observed in typical truck drives (bottom three rows).

4.1. Semi-Truck Highway Dataset (Real)

Setup: The stereo setup is mounted on a large semi-truck at approximately 3m from the ground. The cameras are mounted on a rigid bar using an adjustable custom-made mount and calibrations are performed at the beginning and end of the drives to ensure correctness. Our main sensor used in this dataset is the OnSemi AR0820 cameras which are built around a 1/2-inch CMOS sensor recording raw data in the RCCB format. Our setup consists of 4 synchronized AR0820s in a 2m wide baseline arrangement, with the baseline varying between 0.6m and 0.7m. The cameras record images at $15Hz$ at a resolution of 3848×2168 pixels.

Scene Diversity: The dataset consists of recordings from geographically diverse locations in New Mexico and Virginia. The dataset comprises of diverse scenes from Highway and Urban areas, note given our capture vehicle is a semi-truck we do not collect samples from dense urban scenes. In addition, we also provide data with diverse natural lighting conditions ranging from afternoon, evening and night scenarios as well.

The dataset consists of 50,029 unique stereo pairs. We capture with 4 cameras and offline calibration parameters for each camera. The 4 camera setup allowing one to use up to 6 different combinations of stereo pairs, or even 3 or 4 cameras simultaneously. This is also essential for the rectification task considering the error in relative pose (from calibration) between any two cameras is unique. The data is sampled from scenarios where downstream stereo tasks performed inadequately due to poor rectification quality. The collected data validates the need for online rectification approaches due in long-haul journeys.

4.2. Carla Dataset (Synthetic)

Setup: We rely on the Carla driving simulator [14] to simulate a further synthetic dataset with known ground truth pose information, and perhaps more importantly, the ability to simulate extreme pose deviations that are rare in real captures. Our sensor setup here consists of 3 RGB Cameras, separated by a baseline of 0.8m each mounted on a regular traffic vehicle. Each camera records at $30hz$ capturing images of resolution 2560×1440.

Dataset	Training Set	Test Set	Synthetic
Semi-Truck Highway	40,007	10,022	N
KITTI [18]	6,587	3,252	N
Carla	6,949	773	Y

Table 1. Dataset Statistics. We list here the the train/test splits we use for our novel pose datasets and KITTI[18].

Scene Diversity: We simulate in a largely urban scene with traffic and some additional highway scenes. In order to add diversity we take advantage of lighting and weather controls in Carla, to generate random scenes with a wide range of environmental effects. Similar to the trucking dataset we capture scenes with a variety of natural conditions such as dawn, morning, afternoon, dusk. We also add random weather such as fog, rain, overcast and sunny scenes. Since the scenario remains unchanged throughout the recording, this results in highly similar scenes with different illumination and weather conditions.

This dataset consists of 7,722 stereo pairs, although smaller in comparison to the Trucking dataset we introduce significant perturbations, in the range of [-1, 1]°, to the camera extrinsics in all three axes of rotation. This makes it more challenging to evaluate on, and useful to indicate robustness (or lack thereof) in most rectification approaches. It is important to note no perturbations are introduced to the translation components. As above, the multi-camera setup enables us to sample different pairs for stereo rectification.

Dataset	Method	MAE	SIFT Offset *(pixels)*	SuperGlue Offset *(pixels)*	Vertical Flow *(pixels)*
(a) **Semi-Truck Highway**	Unrectified	0.23	8.21	3.01	3.25
	GT (Offline Calibration)	-	2.91	0.77	1.34
	SIFT + LO-RANSAC [11, 35]	0.26	21.10	17.61	18.11
	SuperGlue + MAGSAC [3, 48]	0.15	14.97	11.96	12.10
	LOFTR + MAGSAC [3, 50]	0.15	14.94	11.87	11.95
	RPNet [16]	0.18	12.10	9.19	9.39
	DirectionNet [10]	0.28	8.41	6.10	6.44
	ViTPose [44]	0.07	4.53	2.37	2.83
	Ours (*w/o OF*)	0.023	3.13	0.97	1.43
	Ours (*w/ OF*)	**0.015**	**2.67**	**0.59**	**1.05**
(b) **KITTI**[18]	Unrectified	0.31	16.55	8.25	9.6
	GT	-	1.65	0.43	0.33
	SIFT + LO-RANSAC [11, 35]	0.08	1.82	0.63	0.61
	SuperGlue + MAGSAC [3, 48]	0.06	2.85	1.67	1.69
	LOFTR + MAGSAC [3, 50]	0.04	2.14	0.97	0.96
	RPNet [16]	0.06	1.74	0.59	0.52
	DirectionNet [10]	0.06	2.13	0.94	0.94
	ViTPose [44]	0.03	1.95	0.71	0.73
	Ours (*w/o OF*)	0.015	1.73	0.53	0.51
	Ours (*w/ OF*)	**0.011**	**1.44**	**0.36**	**0.28**
(c) **Carla**	Unrectified	0.27	13.32	13.76	14.17
	GT	-	3.22	0.45	0.49
	SIFT + LO-RANSAC [11, 35]	0.11	3.6	0.82	1.23
	SuperGlue + MAGSAC [3, 48]	0.071	5.77	2.96	3.21
	LOFTR + MAGSAC [3, 50]	0.075	7.29	4.53	4.52
	RPNet [16]	0.13	7.72	5.52	5.53
	DirectionNet [10]	0.133	13.47	10.64	11.12
	ViTPose [44]	0.06	4.34	1.45	1.72
	Ours (*w/o OF*)	0.13	7.68	5.53	5.67
	Ours (*w/ OF*)	**0.05**	*3.55*	*0.64*	*1.003*

Table 2. Quantitative Evaluation on the (a) Semi-Truck Highway, (b) KITTI [18] and (c) Carla Datasets. Evaluation on unrectified images and ground truth images are included, and we evaluate the proposed method when trained without optical flow (OF) self-supervision.

5. Experiments

In the following, we validate the proposed method by evaluating on the test sets defined above and comparing baseline approaches. We also confirm the effectiveness of the design choices with ablation experiments.

Baselines. We compare our models to traditional approaches and learned approaches to camera pose estimation. The learned components are incorporated in two ways. First, we employ at keypoint-based approaches based on hand-crafted [35] and learned [48, 50] features coupled with robust estimators such as ℓ_0-RANSAC and MAGSAC [3, 11]. Second, we compare our method to existing state-of-the-art end-to-end pose estimation models, including En *et al*. [16] (RPNet), Chen *et al*. [10] (DirectionNet) and Rockwell *et al*. [44] (we dub this approach ViTPose). We find that all methods perform well on existing pose estimation datasets, e.g., MatterPort3D [9], but struggle when evaluated on the wide baseline data.

Metrics. To evaluate the rectification quality, we introduce a key-point offset metric in Table 2. The metric first finds keypoints using existing methods followed by matching across the stereo images. We then compute the average offset along the y-axis of the keypoint matches for two keypoint types, SIFT [35] and Superglue [48]. We also evaluate the rectification with Mean Absolute Error (MAE) between the Ground Truth rectified images and the rectified images from each method. Finally, we report vertical flow offset measurements (which our method is trained to minimize). We also report angular errors rotation estimates in the Supplementary Material.

Method	Inference time *(msec)*
SIFT + LO-RANSAC [11, 35]	338
SuperGlue + MAGSAC [3, 48]	106
LOFTR + MAGSAC [3, 50]	173
RPNet [16]	577
DirectionNet [10]	556
ViTPose [44]	88
Ours	**86**

Table 3. Inference Time. We measure inference time for our method and competing approaches on GPU hardware, see text.

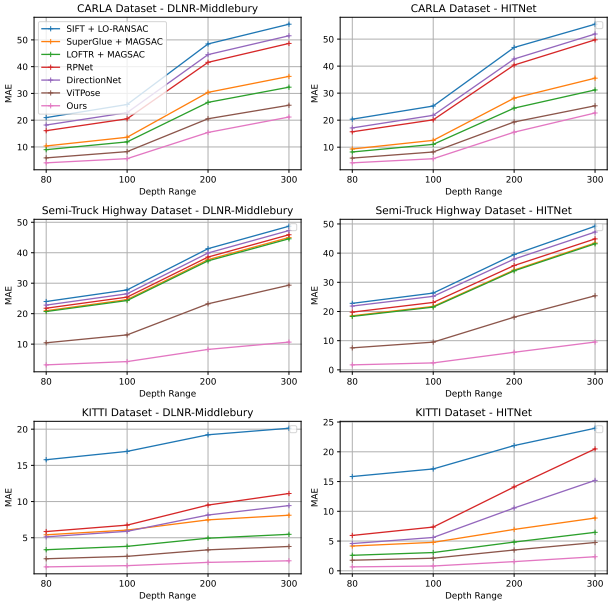

Figure 4. MAE of depth from HITNet [51] and DLNR [60] across the depth range, with inputs obtained from stereo images rectified by with several rectification methods alongside the proposed.

Additionally, we evaluate downstream depth estimation models for insight into the correlation between rectification quality and stereo depth estimation. We include the Mean Absolute Error (MAE) computed between ground truth depth and depth estimated from different rectification methods. Further discussion on the downstream evaluation is listed in the Supplementary Material.

5.1. Quantitative Analysis and Ablation Studies

We report the evaluation results in Table 2. The KITTI[18] evaluations, all with a narrow baseline and little pose variation, reveal our method outperforms other approaches despite these approaches showing competitive results. Furthermore, our method is favorable, even outperforming Ground Truth (offline calibration) as well. Our evaluations on the Carla dataset and Semi-Truck Highway dataset with severe pose variations capture robustness (or lack thereof) in all the methods. The proposed approach fares best in all metrics overall, outperforming Ground Truth (offline

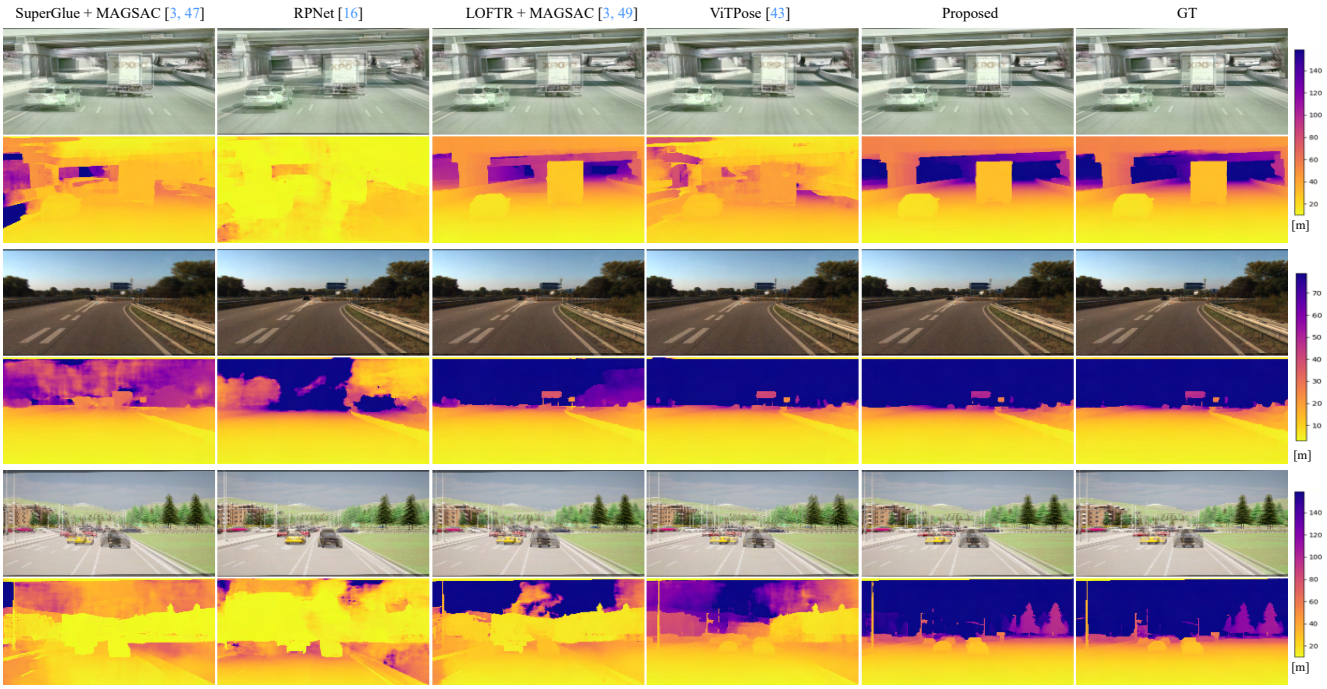

Figure 5. **Qualitative Assessment**. We overlay here the rectified left-right stereo pairs. Each column represents a different rectification method. To visually evaluate the rectification quality, focus on an object in the scene and compare the vertical disparity. Every row is accompanied by the corresponding depth inferred from HITNet [51]. This grid contains samples from (Top) Semi-Truck Highway dataset, (Middle) KITTI [18] and (Bottom) Carla dataset. As seen above, our method is robust to different scenes and pose variations and our quantitative results (Table 2) supports this further.

calibration) in Semi-Truck Highway and comparable to Ground Truth (simulation calibration) in Carla. Additionally, the evaluations validate the importance of the novel self-supervised vertical flow loss (Ours (*w/o OF*)).

5.2. Effect on Downstream Depth Estimation

We evaluate the effect of improved rectification on downstream task performance and pass the rectified images using different approaches to two SOTA stereo models (HITNet [51] and DLNR [60]). Fig. 4 shows quantitative comparisons of MAE over different distances (note this MAE metric is different from Table 2 as it is computed against Ground Truth Depth) and Fig. 5 shows qualitative comparisons of rectification and resulting stereo depth. We observe a general trend of poor depth estimates at larger distances, which can be attributed to the model itself rather than the rectification. With this in mind, when using the proposed method, we measure a 17% improvement on average over [0-300]m in MAE on the Carla dataset using DLNR, and a 10% improvement on average over [0-300]m on HITNet. On KITTI, we observe the MAE to be similar to other methods but we do improve on these metrics by 51% over the [0-300]m range on both SOTA models. Finally, we measure a large performance gain in MAE on the Semi-Truck Highway dataset, with over a 63% improvement on average on [0-300]m range in both models. This validates that the proposed model performs better in rectification and results in higher accuracy depth estimates.

6. Conclusion

We propose an online stereo rectification method for wide baseline stereo setups, which aims to address the calibration degradation that occurs due to environmental effects and prolonged exposure. These effects manifest as vibrations, stretch and twist due to temperature and stress gradients. Our method hinges on weak supervision from offline calibration and self-supervision using vertical flow. We take a stereo correlation volume-based approach to establish correspondences and estimate relative rotation between a stereo pair. We train and evaluate the approach on two novel wide-baseline stereo datasets, one captured with a semi-truck on highways, and another simulated one with extreme pose variations. Our method compares favorably to existing traditional and learned pose estimation and online calibration methods in terms of calibration accuracy and the accuracy of downstream stereo depth. Exciting future directions include multi-scale iterative refinement of calibration and simultaneous multi-camera rectification.

References

[1] Eduardo Arnold, Jamie Wynn, Sara Vicente, Guillermo Garcia-Hernando, 'Aron Monszpart, Victor Adrian Prisacariu, Daniyar Turmukhambetov, and Eric Brachmann. Map-free visual relocalization: Metric pose relative to a single image. In *European Conference on Computer Vision*, 2022. 3

[2] Nicholas Ayache and Charles D. Hansen. Rectification of images for binocular and trinocular stereovision. *[1988 Proceedings] 9th International Conference on Pattern Recognition*, pages 11–16 vol.1, 1988. 2

[3] Dániel Baráth, Jana Noskova, Maksym Ivashechkin, and Jiri Matas. Magsac++, a fast, reliable and accurate robust estimator. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1301–1309, 2019. 7

[4] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Computer Vision – ECCV 2006*, pages 404–417, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. 1

[5] Paul Beardsley, Phil Torr, and Andrew Zisserman. 3d model acquisition from extended image sequences. In *Computer Vision — ECCV '96*, pages 683–695, Berlin, Heidelberg, 1996. Springer Berlin Heidelberg. 1

[6] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac — differentiable ransac for camera localization. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2492–2500, 2016. 3

[7] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *Computer Vision – ECCV 2010*, pages 778–792, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. 1

[8] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *ArXiv*, abs/2005.12872, 2020. 3

[9] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3D Vision (3DV)*, 2017. 7

[10] Kefan Chen, Noah Snavely, and Ameesh Makadia. Wide-baseline relative camera pose estimation with directional learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3257–3267, 2021. 3, 5, 7

[11] Ondřej Chum, Jiri Matas, and Josef Kittler. Locally optimized ransac. In *DAGM-Symposium*, 2003. 7

[12] Gabriela Csurka, Christopher R. Dance, and M. Humenberger. From handcrafted to deep local features. *arXiv: Computer Vision and Pattern Recognition*, 2018. 1

[13] Thao Dang, Christian Hoffmann, and Christoph Stiller. Continuous stereo self-calibration by camera parameter tracking. *IEEE Transactions on Image Processing*, 18(7):1536–1550, 2009. 2, 3

[14] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16, 2017. 6

[15] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 3

[16] Sovann En, Alexis Lechervy, and Frédéric Jurie. Rpnet: an end-to-end network for relative camera pose estimation. In *ECCV Workshops*, 2018. 3, 7

[17] Andrea Fusiello and Luca Irsara. Quasi-euclidean uncalibrated epipolar rectification. *2008 19th International Conference on Pattern Recognition*, pages 1–4, 2008. 2

[18] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013. 2, 5, 6, 7, 8

[19] Mihail Georgiev, Atanas P. Gotchev, and Miska M. Hannuksela. A fast and accurate re-calibration technique for misaligned stereo cameras. *2013 IEEE International Conference on Image Processing*, pages 24–28, 2013. 2

[20] Joshua Gluckman and Shree K. Nayar. Rectifying transformations that minimize resampling effects. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, 1:I–I, 2001. 2

[21] Peter Hansen, Hatem Alismail, Peter Rander, and Brett Browning. Online continuous stereo extrinsic parameter estimation. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1059–1066, 2012. 2

[22] Richard I. Hartley. Theory and practice of projective rectification. *International Journal of Computer Vision*, 35:115–127, 1999. 2, 4

[23] Timo Hinzmann, Tim Taubner, and Roland Siegwart. Flexible stereo: Constrained, non-rigid, wide-baseline stereo vision for fixed-wing aerial platforms. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2550–2557, 2018. 1

[24] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. *ArXiv*, abs/1506.02025, 2015. 4

[25] Dinghuang Ji, Junghyun Kwon, Max McFarland, and Silvio Savarese. Deep view morphing. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7092–7100, 2017. 2

[26] Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6555–6564, 2017. 3

[27] Alex Kendall, Matthew Koichi Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2938–2946, 2015. 3

[28] Zakaria Laskar, Iaroslav Melekhov, Surya Kalia, and Juho Kannala. Camera relocalization by computing pairwise relative poses using convolutional neural network. *2017 IEEE*

International Conference on Computer Vision Workshops (ICCVW), pages 920–929, 2017. 3

[29] Jiankun Li, Peisen Wang, Pengfei Xiong, Tao Cai, Zi-Ping Yan, Lei Yang, Jiangyu Liu, Haoqiang Fan, and Shuaicheng Liu. Practical stereo matching via cascaded recurrent network with adaptive correlation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16242–16251, 2022. 2

[30] Tony Lindeberg. Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention. *Int. J. Comput. Vision*, 11(3): 283–318, 1993. 1

[31] Tony Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30:79–116, 1998. 1

[32] Yonggen Ling and Shaojie Shen. High-precision online markerless stereo extrinsic calibration. *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1771–1778, 2016. 2, 3

[33] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021. 4

[34] Charles T. Loop and Zhengyou Zhang. Computing rectifying homographies for stereo vision. *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, 1:125–131 Vol. 1, 1999. 2

[35] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.*, 60(2):91–110, 2004. 1, 2, 7

[36] Huoling Luo, Congcong Wang, Xingguang Duan, Hao Liu, Ping Wang, Qingmao Hu, and Fucang Jia. Unsupervised learning of depth estimation from imperfect rectified stereo laparoscopic images. *Computers in biology and medicine*, 140:105109, 2021. 2

[37] John Mallon and Paul F. Whelan. Projective rectification from the fundamental matrix. *Image Vis. Comput.*, 23:643–650, 2005. 2, 3

[38] Iaroslav Melekhov, Juho Kannala, and Esa Rahtu. Relative camera pose estimation using convolutional neural networks. *ArXiv*, abs/1702.01381, 2017. 3

[39] Chethan Parameshwara, Gokul Hari, Cornelia Fermuller, Nitin J. Sanket, and Yiannis Aloimonos. Diffposenet: Direct differentiable camera pose estimation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6835–6844, 2022. 3

[40] P. Pritchett and A. Zisserman. Wide baseline stereo matching. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 754–760, 1998. 1

[41] Philip Pritchett and Andrew Zisserman. Matching and reconstruction from widely separated views. In *3D Structure from Multiple Images of Large-Scale Environments*, pages 78–92, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg. 1

[42] Eike Rehder, Christian Kinzig, Philipp Bender, and Martin Lauer. Online stereo camera calibration from scratch. *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 1694–1699, 2017. 2

[43] Jérôme Revaud, César Roberto de Souza, M. Humenberger, and Philippe Weinzaepfel. R2d2: Reliable and repeatable detector and descriptor. In *Neural Information Processing Systems*, 2019. 2

[44] C. Rockwell, Justin Johnson, and David F. Fouhey. The 8-point algorithm as an inductive bias for relative pose prediction by vits. *2022 International Conference on 3D Vision (3DV)*, pages 1–11, 2022. 3, 7

[45] Barbara Roessle and Matthias Nießner. End2end multi-view feature matching with differentiable pose optimization. 2022. 3

[46] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *Computer Vision – ECCV 2006*, pages 430–443, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. 1

[47] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571, 2011. 1

[48] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. *CoRR*, abs/1911.11763, 2019. 2, 3, 7

[49] Cordelia Schmid and Roger Mohr. Matching by local invariants. Research Report RR-2644, INRIA, 1995. 1

[50] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8918–8927, 2021. 2, 7

[51] Vladimir Tankovich, Christian Häne, S. Fanello, Yinda Zhang, Shahram Izadi, and Sofien Bouaziz. Hitnet: Hierarchical iterative tile refinement network for real-time stereo matching. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14357–14367, 2020. 1, 2, 7, 8

[52] Zachary Teed and Jia Deng. RAFT: recurrent all-pairs field transforms for optical flow. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part II*, pages 402–419. Springer, 2020. 5

[53] P. H. S. Torr and A. Zisserman. Feature based methods for structure and motion estimation. In *Vision Algorithms: Theory and Practice*, pages 278–294, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg. 1

[54] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Neural Information Processing Systems*, 2017. 3, 4

[55] Jialiang Wang, Daniel Scharstein, Akash Bapat, Kevin Blackburn-Matzen, Matthew Yu, Jonathan Lehman, Suhib Alsisan, Yanghan Wang, Sam S. Tsai, Jan-Michael Frahm, Zijian He, Péter Vajda, Michael F. Cohen, and Matthew Uyttendaele. A practical stereo depth system for smart glasses. *ArXiv*, abs/2211.10551, 2022. 2

[56] Yuxing Wang, Yawen Lu, and Guoyu Lu. Stereo rectification based on epipolar constrained neural network. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2105–2109, 2021. 3

[57] Ruichao Xiao, Wenxiu Sun, Jiahao Pang, Qiong Yan, and Jimmy S. J. Ren. Dsr: Direct self-rectification for uncalibrated dual-lens cameras. *2018 International Conference on 3D Vision (3DV)*, pages 561–569, 2018. 2

[58] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 8111–8120. IEEE, 2022. 4

[59] Xuchong Zhang, Yongli Zhao, Hang Wang, Han Zhai, Hongbin Sun, and Nanning Zheng. End-to-end learning of self-rectification and self-supervised disparity prediction for stereo vision. *Neurocomputing*, 494:308–319, 2022. 2

[60] Haoliang Zhao, Huizhou Zhou, Yongjun Zhang, Jie Chen, Yitong Yang, and Yong Zhao. High-frequency stereo matching network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1327–1336, 2023. 2, 7, 8

[61] Frederik Zilly, Marcus Müller, Peter Eisert, Peter Kauff, and Heinrich-Hertz-Institut Einsteinufer. Joint estimation of epipolar geometry and rectification parameters using point correspondences for stereoscopic tv sequences. 2010. 2