Extreme Rotation Estimation using Dense Correlation Volumes

Ruojin Cai[1]   Bharath Hariharan[1]   Noah Snavely[1,2]   Hadar Averbuch-Elor[1,2]

[1]Cornell University   [2]Cornell Tech

Abstract

We present a technique for estimating the relative 3D rotation of an RGB image pair in an extreme setting, where the images have little or no overlap. We observe that, even when images do not overlap, there may be rich hidden cues as to their geometric relationship, such as light source directions, vanishing points, and symmetries present in the scene. We propose a network design that can automatically learn such implicit cues by comparing all pairs of points between the two input images. Our method therefore constructs dense feature correlation volumes and processes these to predict relative 3D rotations. Our predictions are formed over a fine-grained discretization of rotations, bypassing difficulties associated with regressing 3D rotations. We demonstrate our approach on a large variety of extreme RGB image pairs, including indoor and outdoor images captured under different lighting conditions and geographic locations. Our evaluation shows that our model can successfully estimate relative rotations among non-overlapping images without compromising performance over overlapping image pairs.[1]

1. Introduction

Estimating the relative pose between a pair of RGB images is a fundamental task in computer vision with applications including 3D reconstruction [39, 34], camera localization [3, 40, 42], simultaneous localization and mapping [8, 32] and novel view synthesis [29, 38]. Standard methods for computing relative pose are highly dependent on accurate correspondence. But what if the poses are so different that there is no overlap and hence no correspondence?

Our work takes a step towards this seemingly impossible goal of estimating relative pose for pairs of RGB images that have little or no overlap. In particular, we present a technique for estimating relative 3D rotation for a pair of images with (possibly) extreme relative motion. There are many applications where dense imagery is difficult to obtain that can benefit from rotation estimation from non-overlapping views
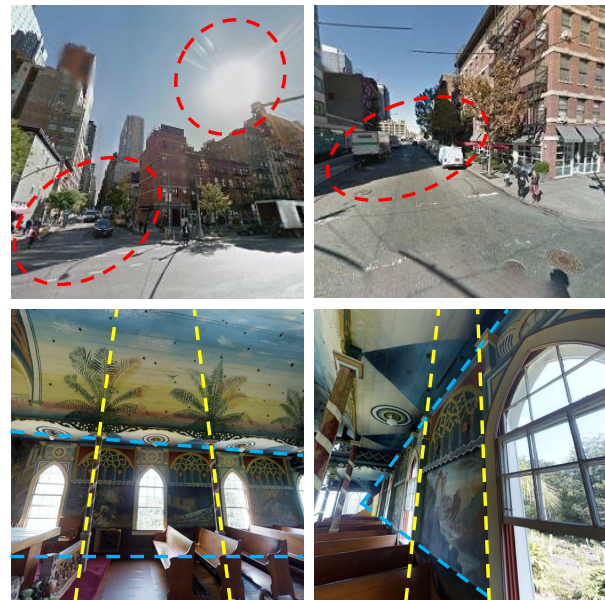

Figure 1: **How can we estimate relative rotation between images in extreme non-overlapping cases?** Above we show two non-overlapping image pairs capturing an urban street scene (top) and a church (bottom). Possible cues to their relationship include sunlight and direction of shadows in outdoor scenes (highlighted in red) and lines parallel in 3D in indoor scenes (marked with yellow and blue line segments), from which vanishing points can be derived.

[25, 1]. For example, when advertising homes on online real estate sites, users may only provide a small number of images—too sparse for current 3D reconstruction methods. Rotation estimation can simplify downstream tasks, such as 3D reconstruction from sparse views.

How can we reason about relative rotation in extreme non-overlapping settings? As humans, there are a number of cues we might leverage. Consider the two image pairs in Fig. 1. For the outdoor pair, we can infer relative orientation using illumination cues, e.g., by analyzing which buildings are lit or the directions of cast shadows. Geometric cues are also useful. For example, from the pair of indoor images we can infer a change in camera pitch from the set of parallel vertical
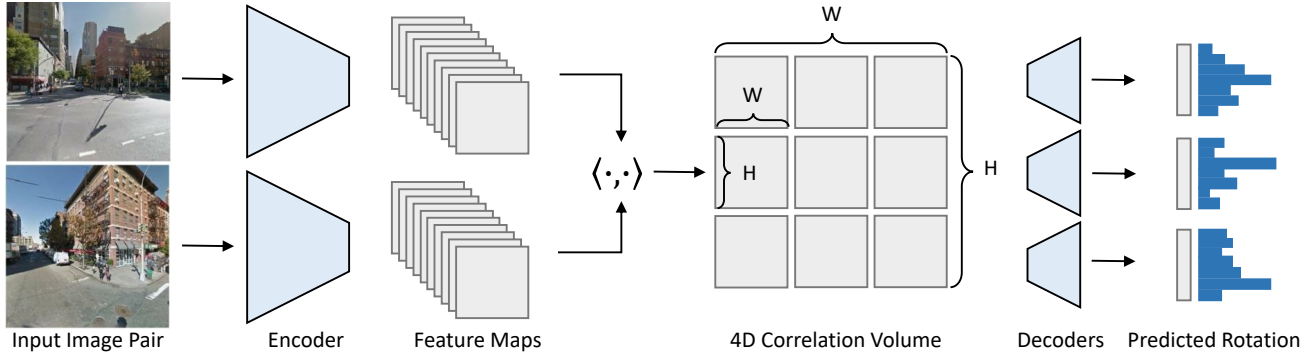
Figure 2: **Method overview.** Given a pair of images, a shared-weight Siamese encoder extracts feature maps. We compute a 4D correlation volume using the inner product of features, from which our model predicts the relative rotation (here, as distributions over Euler angles).

lines in 3D (colored in yellow) that suggest a vanishing point, and we can infer a rightward camera rotation by analyzing symmetries and the layout of the benches.

Given the presence of such "hidden" cues, one approach to computing relative rotation would be to explicitly learn such cues via supervision, e.g., by labeling vanishing points and learning to predict them. However, in addition to the drawbacks of requiring additional supervision, we do not want to restrict our model to a set of handcrafted cues which may or may not be relevant for the image pair provided at test time. Instead, we want to learn to predict relative rotation from pose supervision alone. As such, we ask: Can we guide the network to reason about such hidden cues implicitly? And what architecture would best achieve this goal?

Our key insight is that reasoning about cues such as vanishing points and illumination—while not achievable from direct feature correspondence alone—nonetheless can be realized through comparison of local properties like line orientations (in the case of vanishing points) and shadows and light sources (in the case of illumination). Crucially, any pair of points between the image pair can provide evidence for their geometric relationship.

We therefore turn to *correlation volumes*, a tool used in correspondence tasks like optical flow or stereo. In a full correlation volume, every pair of points from feature maps derived from an image pair are compared. While dense correlation volumes have demonstrated superior performance for tasks like optical flow [46, 27, 43] and stereo matching [33, 23, 47, 14] that compare highly overlapping images, we find that they are also effective in finding implicit cues that are not in the form of direct correspondence. As such, we process image pairs—whether they overlap or not—by constructing a dense 4D correlation volume (see Fig. 3). This design allows us to both find explicit pixelwise correspondence, in the case of overlapping pairs, as well as leverage implicit cues for non-overlapping pairs.

To estimate the relative rotation, we process the correlation volume with another network that computes proba-

bilities estimates over a fine-grained discretization over the space of 3D rotations. Our framework is end-to-end trainable and optimizes simple loss formulations, bypassing difficulties associated with regressing 3D rotations.

We evaluate our method on a large variety of extreme RGB image pairs, including indoor and outdoor images captured in different geographic locations under varying illumination. We also show that our models yield state-of-the-art performance for overlapping pairs. Our models generalize surprisingly well to new data—e.g., training a model on outdoor scenes in Manhattan yields median errors below 6° for images captured in Pittsburgh and London.

2. Related Work

Extreme relative pose estimation. Non-overlapping images have been addressed in the context of related problems such as 2D alignment and mosaicing [36, 17], where the goal is to piece together images that have small gaps between them. Most prior work on the 3D relative pose estimation problem rely on correspondences or overlap, and hence are not well suited for handling extreme cases. Several works address extreme relative pose estimation between two input RGB-D scans [48, 49]. Provided with input depth values, these works perform scan completion and match the completed scans. Caspi and Irani [5] consider image sequences where the two cameras are rigidly attached and move jointly, and search for consistent temporal behavior. Littwin *et al*. [24] use inter-silhouette dissimilarities to estimate the relative rotations for a set of cameras. In this work, we propose a method for the more challenging problem of extreme relative rotation given only a pair of RGB images as input, without using depth, temporal coherence, or other forms of additional data.

Traditional pose estimation. The problem of relative pose estimation from overlapping views is traditionally divided into two sequential steps: correspondence estimation using local feature matching [26, 2], followed by epipolar

geometry–based pose estimation [15]. As reliable correspondence estimation can be challenging even for overlapping images, several recent works propose end-to-end convolutional neural networks that regress from images to relative pose [28, 12, 20].

In particular, several works focus on 3D rotations and propose parameterizations and architectures that allow for better estimation using neural networks [52, 21]. Zhou *et al.* [52] evaluate commonly used representations, including quaternions and Euler angles, and discuss issues relating to discontinuities in the rotation representation. Peretroukhin *et al.* [35] regress and combine multiple rotation estimates, producing probabilistic estimates. Mohlin *et al.* [31] estimate 3D rotation uncertainty using the matrix Fisher distribution. In our work, we discretize the space of 3D rotations and estimate a distribution over this space to avoid issues associated with direct regression to rotations.

Dense volumes for computing image relations. Dense volumes that encode pairwise pixel comparisons allow for an explicit representation of correspondence, and thus have been used in a number of tasks that require correspondence estimation. For instance, deep stereo matching approaches typically compute cost volumes that compare feature descriptors across disparity ranges [18, 19, 6, 50, 33, 23, 47, 14]. Cost volumes are also used to compute optical flow, in which case 2D displacements are encoded within a volumetric representation [46, 41, 27]. To define a matching cost, several works compute dense correlations between image features, forming correlation volumes [10, 41, 43]. In our work, we also form correlation volumes. However, unlike prior work, the input images are not necessarily highly correlated and therefore the correlation volumes are tasked with learning additional, more subtle, cues.

Several other recent techniques leverage dense volumes for reconstruction tasks. Zhou *et al.* [54] use cost volumes for the task of single-image reconstruction. Wei *et al.* [44] use them to refine structure-from-motion predictions. As part of their solution, they form a pose-based cost volume that is provided with initial camera pose parameters, then uniformly samples candidate poses around this initial estimate. In our case, we build dense volumes for all pairs of pixels and estimate rotations over the full space of possible rotations.

3. Method

Given a pair of RGB images (I_1, I_2), our goal is to estimate the 3D rotation matrix $\mathbf{R}$ between the two images. In order to allow for discovery and use of hidden cues in the difficult task of extreme rotation prediction, we use dense correlation volumes that allow for discovering implicit cues (Section 3.1). These dense volumes are fed to a fine-grained relative rotation classification network (Section 3.2). An overview of our approach is provided in Fig. 2.

Parameterization. We must choose a parameterization for 3D rotations suited to our problem. One standard representation for a 3D rotation matrix $\mathbf{R}$ is as three Euler angles $[\alpha, \beta, \gamma]$, denoting roll, pitch, and yaw angles, respectively:

$$\mathbf{R}(\alpha, \beta, \gamma) = \mathbf{R}_x(\alpha)\mathbf{R}_y(\beta)\mathbf{R}_z(\gamma) \quad (1)$$

This general parameterization can be integrated directly with our presented approach. However, we observe that for a wide variety of scenes the absolute pitch can be recovered from a single image using cues such as vanishing points. Furthermore, cameras are typically upright, *i.e.*, the input images have zero roll.

Given these observations, we can instead represent the relative orientation $\mathbf{R}$ with three angles $[\beta_1, \beta_2, \Delta\gamma]$, where $\Delta\gamma$ denotes the relative yaw angle, β_1 denotes the pitch of I_1 and β_2 denotes the pitch of I_2. Using this parameterization, the rotation matrices of the two images are defined as $\mathbf{R}_1(0, \beta_1, 0)$ and $\mathbf{R}_2(0, \beta_2, \Delta\gamma)$, and the relative rotation matrix is defined as $\mathbf{R} = \mathbf{R}_2\mathbf{R}_1^T$. In our experiments we show that encoding this prior knowledge into the parameterization leads to improved performance, although a generic parameterization also shows significant improvement over baseline methods.

3.1. Dense Correlation Volumes

Our proposed solution is inspired by traditional methods that accumulate evidence for global quantities from local evidence via voting schemes or other mechanisms. For instance, as intuition, consider the problem of detecting vanishing points from a single image. One approach is to have local features like line segments vote on vanishing points using accumulation methods similar to Hough transforms [16]. In our case, where we are given an image pair and want to estimate relative rotation, we observe that potentially any pair of image patches can provide evidence for the global geometric image relationship—for instance, two patches that support related vanishing point locations, or two patches that give evidence for light source directions.

To operationalize this intuition, we devise a network structure that performs pairwise comparisons between all pairs of features across the two images. We first use convolutional networks with shared weights to extract dense feature descriptors $\mathbf{f}(I_i) \in \mathbb{R}^{K \times H/4 \times W/4}$, where K is the number of channels, W is the image width and H is the image height. We then compute a 4D dense correlation volume $\mathcal{C}(\mathbf{f}(I_1), \mathbf{f}(I_2))$, such that for each pair of spatial positions (p, q) in I_1 and (r, s) in I_2, we define the correlation score at (p, q, s, r) as the dot product of the corresponding vectors in the feature maps of the two images:

$$\mathcal{C}(\mathbf{f}(I_1), \mathbf{f}(I_2); p, q, r, s) = \mathbf{f}(I_1; p, q) \cdot \mathbf{f}(I_2; r, s), \quad (2)$$

where $\mathbf{f}(I; x, y)$ denotes the feature vector for image I at spatial position (x, y). Fig. 3 illustrates the correlation vol-
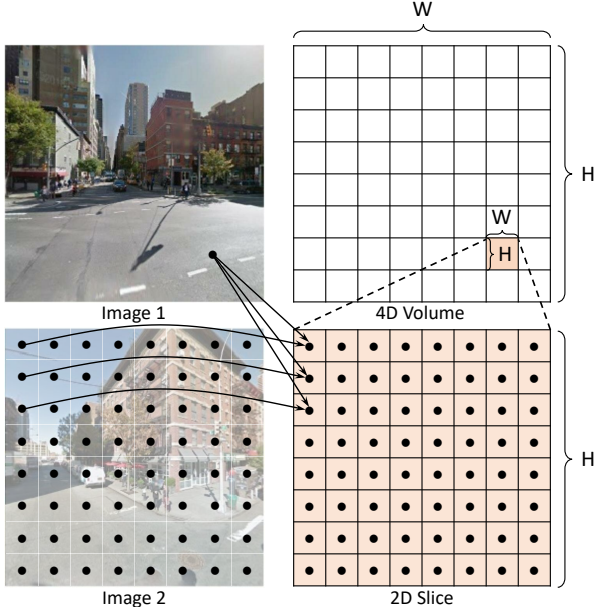
Figure 3: **4D correlation volumes.** A 4D correlation volume is calculated from a pair of image feature maps. Given a feature vector from Image 1, we compute the dot product with all feature vectors in Image 2, and build up a 2D slice of size $H \times W$. Combining all 2D slices over all feature vectors in Image 1, we obtain a 4D correlation volume of size $H \times W \times H \times W$.
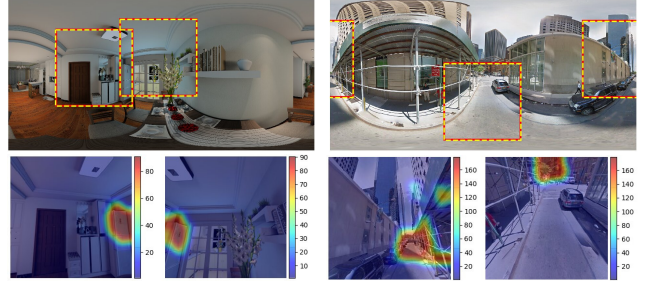

Figure 4: **Visualizing cues detected by our model for overlapping (left) and non-overlapping (right) image pairs.** We show regions which, when blocked, affect the rotation error, with warmer colors depicting larger errors (according to their associated color bars). The full panoramas are shown above, with the ground-truth and predicted perspective image regions marked in red and yellow, respectively. These visualizations suggest that our method reasons about pointwise correspondences for overlapping pairs (*e.g.* on top of the refrigerator) and implicit cues for non-overlapping pairs (*e.g.* related to vanishing points).

ume for an image pair. Note that the correlation volume $\mathcal{C}$ can be computed efficiently using matrix multiplication.

Unlike prior work that uses correlation volumes to directly predict pixel-to-pixel correspondence as in optical flow [46, 27, 43] or stereo [33, 23, 14], our correlation volumes are implicitly assigned a dual role which emerges through training on both overlapping and non-overlapping pairs. When the input image pair contains significant overlap, pointwise correspondence can be computed and transferred onward to the rotation prediction module. When the input image pair contains little to no overlap, the correlation volume can assume the novel role of detecting implicit cues.

We visualize this dual role in Fig. 4, where we create a heatmap for each image in a pair occluding each image region in turn with a sliding window before feeding it to our network, in order to assess each region's approximate importance towards computing the relative pose [51]. As illustrated in the left pair, covering the region of overlap significantly affect the model's prediction for overlapping pairs. For non-overlapping pairs, covering regions corresponding to a strong vanishing point yields a steep drop in performance. Please refer to the supplementary material for additional visualizations.

Our dense correlation volume $\mathcal{C}$ is provided as input to a rotation classification network g_σ that is tasked with predicting the relative rotation.

3.2. Relative Rotation Classification

The relative rotation classification network g_σ is constructed from three identical networks (without weight sharing), where each network predicts one of the angles describing the relative rotation according to our three angle parameterization. Learning-based pose estimation methods typically regress to rotation and translation parameters [28, 12]. However, commonly used 3D rotation representations, including Euler angles and quaternions, are discontinuous and hence challenging for direct regression in deep networks. To overcome the problem of regressing discontinuous 3D rotations, prior works have suggested using higher dimensional continuous representations, e.g., in 5D and 6D [52].

As an alternative to directly regressing the relative angles, we discretize the space of rotations, such that for each angle we estimate a probability distribution over N bins. We empirically set $N = 360$, and let each bin capture an angle in the range $[-180°, 180°]$. This discretization is related to concurrent work that discretizes 3D rotations in terms of rotation matrix columns [7]. Our fine-grained discrete angle parameterization enables using a simple cross-entropy loss to train our network. Overall, a sum of three cross-entropy loss functions is used, one per angle. During training, the ground truth is set using one-hot vectors (that empirically yielded similar results as smoothed vectors).

4. Experiments

To validate our approach, we conduct extensive experiments on a variety of extreme RGB image pairs capturing both indoor and outdoor scenes. We compare with several baseline techniques, evaluating performance on both overlapping and non-overlapping image pairs. We also demonstrate

the generalization power of our model, testing it on image pairs from unseen cities. Finally, we present an ablation study to examine the impact of the different components in our proposed approach. Additional results and implementation details are provided in the supplementary material.

4.1. Datasets

StreetLearn [30] is an outdoor dataset that contains approximately 143K panoramic views covering the cities of Manhattan and Pittsburgh. We focus on a set of roughly 56K images from Manhattan, randomly allocating 1,000 panoramic views for testing.

SUN360 [45] is an indoor dataset that contains 9,962 panoramic views of different scenes downloaded from the Internet, grouped into 50 different categories. We use approximately 7,500 panoramas for training and 830 among the remaining panoramas for testing.

InteriorNet [22] is a synthetic indoor dataset. We use a subset of InteriorNet that contains 10,050 panoramic views from 112 different houses, where 82 houses are allocated for training and the remaining 30 houses for testing.

For each panoramic view, we randomly sample 100 perspective views to obtain images with a resolution of 256×256 and a $90°$ FoV. We sample images distributed uniformly over the range of $[-180, 180]$ for yaw angles. This yields, for example, an average of 3147 samples (per $1°$ bin) in our StreetLearn dataset. We assume zero roll (and demonstrate in the supplementary material that our models are insensitive to small roll angles at test time). To avoid generating textureless images that focus on the ceiling/sky or the floor, we limit the range over pitch angles to $[-30, 30]$ for the indoor datasets and $[-45, 45]$ for the outdoor dataset. Furthermore, in the InteriorNet dataset, since the dataset is synthetic and panoramas are rendered at random camera positions—leading to cases where the image is too close to the geometry (*e.g.* the scene is largely occluded by a close object or most of the panorama observes a wall), we use MiDaS [37], a single view depth estimation method, to filter out images that are too close the scene.

From these perspective images, we construct datasets with and without camera translations, in order to understand the effect of camera translation on our rotation estimation problem. Datasets without translations are constructed by sampling image pairs originating from the same panorama, and datasets with translations, denoted as InteriorNet-T and StreetLearn-T, are constructed by pairing up images from different panoramas using translation distances smaller than 3m and 10m, respectively (SUN360 is constructed from Internet panoramas that are not physically related in space). For all datasets, we have ∼1M training pairs sampled from the same panorama, ∼700K training pairs sampled from different panoramas, and 1K test pairs. There is no overlap between train and test scenes.

4.2. Baselines

SIFT-based relative rotation estimation. A geometry-based technique that computes SIFT [26] features and then estimates a rotation matrix with a 2-point algorithm [4] for image pairs from the same panorama, or an essential matrix for image pairs with translation, using RANSAC [13].

Learning-based feature matching, using pretrained networks for interest point detection and description (with model fitting as described above). We evaluate a pretrained SuperPointNet [9] (hereby called SuperPoint) and D2-Net [11] (see supplemental material for results).

End-to-end relative rotation regression, where image features are concatenated and fed to a regression model. We evaluate models predicting a continuous representation in 6D (hereby denoted as Reg6D) as proposed by Zhou *et al.* [52] and additional representations, including quaternions as proposed by En *et al.* [12], in the supplementary material.

For all end-to-end techniques, we also train models with *only* overlapping pairs to better understand the impact of training models with non-overlapping image pairs.

4.3. Evaluation Metrics

For an image pair, let $\mathbf{R}$ be the predicted rotation matrix and $\mathbf{R}^*$ be ground truth rotation matrix. We follow prior work and report the geodesic error $\arccos\left(\frac{tr(\mathbf{R}^T\mathbf{R}^*)-1}{2}\right)$.

To analyze the performance of methods across different overlap ratios, we divide the test image pairs into three categories: (i) **large**, indicating highly overlapping pairs (relative rotations up to $45°$), (ii) **small**, indicating pairs that partially overlap (relative rotation angles $\in [45°, 90°]$), and (i) **none**, indicating pairs with no overlap (relative rotations $>90°$).

4.4. Quantitative Evaluation

Table 1 reports the mean and median geodesic error, as well as the percentage of image pairs with a relative rotation error under $10°$. Qualitative results are presented in Figure 7. We analyze the results according to the amount of overlap:

Overlapping cases. For image pairs sampled from the same panorama, our model produces very accurate results for both indoor and outdoor scenes, with mean errors of $4.31°$, $6.13°$, and $3.23°$ for small overlapping pairs. For the regression baselines, adding non-overlapping pairs during training causes the regression baseline performance to suffer on overlapping pairs, while our method does not see such a drop.

The performance of models trained on datasets with camera translations is somewhat lower. For these, the overlapping regions can be smaller due to the camera motion. In particular, for StreetLearn-T, the translations are large (up to 10m), which can have a more dramatic effect on the overlap region. Nonetheless, our method still achieves low median errors for datasets with translation (around $3°$).

Overlap	Method	InteriorNet			InteriorNet-T			SUN360			StreetLearn			StreetLearn-T		
		Avg(°↓)	Med(°↓)	10°(%↑)	Avg(°↓)	Med(°↓)	10°(%↑)	Avg(°↓)	Med(°↓)	10°(%↑)	Avg(°↓)	Med(°↓)	10°(%↑)	Avg(°↓)	Med(°↓)	10°(%↑)
Large	SIFT* [26]	6.09	4.00	84.86	7.78	2.95	55.52	5.46	3.88	93.10	5.84	3.16	91.18	18.86	3.13	22.37
	SuperPoint* [9]	5.40	3.53	87.10	5.46	2.79	65.97	4.69	3.18	92.12	6.23	3.61	91.18	6.38	1.79	16.45
	Reg6D [52]-o	5.43	3.87	87.10	10.45	6.91	67.76	7.18	5.79	81.28	3.36	2.71	97.65	12.31	6.02	69.08
	Reg6D [52]	9.05	5.90	68.49	17.00	11.95	41.79	16.51	12.43	40.39	11.70	8.87	58.24	36.71	24.79	23.03
	Ours-o	**1.53**	1.10	**99.26**	**2.89**	1.10	**97.61**	**1.00**	**0.94**	**100.00**	**1.19**	1.02	**99.41**	9.12	2.91	**87.50**
	Ours	1.82	**0.88**	98.76	8.86	1.86	93.13	1.37	1.09	99.51	1.52	1.09	**99.41**	24.98	**2.48**	78.95
Small	SIFT* [26]	24.18	8.57	39.73	18.16	10.01	18.52	13.71	6.33	56.77	16.22	7.35	55.81	38.78	13.81	5.68
	SuperPoint* [9]	16.72	8.43	21.58	11.61	5.82	11.73	17.63	7.70	26.69	19.29	7.60	24.58	6.80	6.85	0.95
	Reg6D [52]-o	17.83	9.61	51.37	21.87	11.43	44.14	18.61	11.66	39.85	7.95	4.34	87.71	15.07	7.59	63.41
	Reg6D [52]	25.71	15.56	33.56	42.93	28.92	23.15	42.55	32.11	9.40	24.77	15.11	30.56	46.61	34.33	13.88
	Ours-o	6.45	1.61	95.89	10.24	1.38	89.81	3.09	1.41	98.50	2.32	1.41	98.67	13.04	3.49	84.23
	Ours	**4.31**	**1.16**	**96.58**	30.43	2.63	74.07	6.13	1.77	95.86	3.23	1.41	98.34	27.84	3.19	74.76
None	SIFT* [26]	109.30	92.86	0.00	93.79	113.86	0.00	127.61	129.07	0.00	83.49	90.00	0.38	85.90	106.84	0.38
	SuperPoint* [9]	120.28	120.28	0.00	–	–	0.00	149.80	165.24	0.00	–	–	0.00	–	–	0.00
	Reg6D [52]-o	88.89	79.24	0.98	110.50	116.25	1.17	101.21	99.64	0.94	133.08	167.19	2.46	132.67	158.45	1.88
	Reg6D [52]	48.36	32.93	10.82	60.91	51.26	11.14	64.74	56.55	3.77	28.48	18.86	24.39	49.23	35.66	11.86
	Ours	**37.69**	**3.15**	**61.97**	**49.44**	**4.17**	**58.36**	**34.92**	**4.43**	**61.39**	**5.77**	**1.53**	**96.41**	**30.98**	**3.50**	**72.69**
All	SIFT* [26]	13.68	5.04	45.80	12.24	5.69	24.60	18.12	5.02	34.00	17.29	5.53	32.50	36.00	6.03	5.40
	SuperPoint* [9]	8.19	4.08	41.40	6.62	3.38	25.90	11.09	4.00	25.80	11.52	4.80	22.90	6.42	2.62	2.80
	Reg6D [52]-o	34.51	9.71	50.40	48.27	15.59	37.40	60.15	34.51	27.60	73.36	20.03	44.30	77.09	55.29	31.60
	Reg6D [52]	25.90	13.02	40.70	40.38	23.35	25.30	49.05	34.37	12.70	24.51	15.31	32.00	46.50	33.14	29.90
	Ours	**13.49**	**1.18**	**86.90**	**29.68**	**2.58**	**75.10**	**20.45**	**2.23**	**78.30**	**4.40**	**1.44**	**97.50**	29.85	**3.20**	74.30

Table 1: **Rotation estimation evaluation on the InteriorNet, the SUN360, and the StreetLearn datasets.** We report the mean and median geodesic error in degrees, and the percentage of pairs with a relative rotation error under 10° for different levels of overlap, as detailed in Section 4.3. Models trained only on overlapping pairs are denoted with "-o". *Errors are computed only over successful image pairs, for which these algorithms output an estimated rotation matrix (failure over more than 50% of the test pairs is shown in gray).

SuperPoint [9] achieves the smallest average errors on StreetLearn-T large and small overlapping cases (6.38° and 6.80°). However, both SIFT and SuperPoint do not always output an answer (model fitting-based techniques require a sufficient number of detected inliers). Only successful image pairs are considered for evaluation, and hence these errors should be interpreted as errors over pairs for which they produced an answer (full numbers provided in the supplemental material). Our method still significantly outperforms SuperPoint in terms of the fraction of pairs with less than 10° error across all datasets, suggesting that SuperPoint fails to give an answer.

Non-overlapping cases. Due to insufficient correspondences on image pairs with large viewpoint changes, feature matching-based methods (and regression models trained on only overlapping pairs) unsurprisingly fail on pairs with no overlap. The median errors of the strongest regression baseline are all above 18° (and for several datasets are much larger). In contrast, our method yields median errors consistently below 5°, while the mean error rises to 6°–49°. This indicates that our method is usually surprisingly accurate, but sometimes makes large errors. It turns out that such errors are primarily due to ambiguities, as detailed below.

To better understand cases when our method produces incorrect rotations, we can look at the full distributions predicted by our method. Because our network outputs a probability distribution over 3D rotations, we can compute a top-2 error (i.e., the smaller of the errors over both the most likely

and second most likely prediction), to see whether the correct rotation angles have high peaks in the learned probability distributions. On the StreetLearn-T dataset, the mean error significantly drops, from 24.98° to 6.49° on large overlapping cases, 27.84° to 10.39° on small overlapping cases, and 30.98° to 15.72° on non-overlapping cases. This gap illustrates that the model is able to predict with large probability the ground truth angle, but is confused with other likely rotation estimates. For instance, given two images facing two different roads at the crossing, there is an ambiguity between 90° and 180° rotations. We provide full distributions of prediction errors (that demonstrate, for example, that there are error modes at 90° and 180° rotations) and qualitative results of failure cases in the supplemental material.

4.5. Generalization to Other Datasets

To demonstrate the generalization power of our model, we test the model trained on StreetLearn on outdoor images from Pittsburgh and London, using images from the Holicity dataset [53], and compare to the regression model of Zhou *et al.* [52]. Test images are split according to the overlap levels detailed in Section 4.3. Figure 6 illustrates several qualitative examples, also demonstrating how different these samples are (*e.g.* captured in rural areas or on water).

Results are reported in Figure 5. Our StreetLearn-trained model generalizes well to the other outdoor datasets, with all median errors below 6°. In comparison, the median errors for Reg6D on small or non-overlapping pairs are larger on

Figure 5: **Generalization to new locations.** We show the median geodesic rotation error over test images from Manhattan, Pittsburgh, and London for the models trained on Manhattan. Results are reported for our model and the regression models proposed by Zhou *et al.* [52] (Reg6D-o is trained only on overlapping pairs).

Overlap	Method	Rotation Error			Yaw Error		Pitch Error	
		Avg(°)	Med(°)	10°(%)	Avg(°)	Med(°)	Avg(°)	Med(°)
Large	w/o RP	1.02	1.05	100.00	0.48	0.47	0.52	0.48
	w/o $\mathcal{L}_{cls}, \mathcal{C}$	40.19	33.41	8.87	33.22	23.19	16.50	14.01
	w/o $\mathcal{L}_{cls}$	7.93	6.96	78.33	5.07	3.90	3.22	2.91
	w/o $\mathcal{C}$	43.79	43.44	6.90	43.74	43.44	1.07	0.74
	Ours (full)	1.37	1.09	99.51	0.57	0.51	1.03	0.73
Small	w/o RP	1.67	1.44	99.62	0.83	0.61	0.79	0.64
	w/o $\mathcal{L}_{cls}, \mathcal{C}$	67.97	60.91	0.38	61.53	54.19	19.71	15.59
	w/o $\mathcal{L}_{cls}$	15.75	12.65	34.59	10.28	6.35	5.81	4.31
	w/o $\mathcal{C}$	65.31	69.32	15.79	65.13	69.31	2.04	1.10
	Ours (full)	6.13	1.77	95.86	4.70	0.72	1.76	0.97
None	w/o RP	39.01	25.81	23.92	36.32	21.17	5.96	3.39
	w/o $\mathcal{L}_{cls}, \mathcal{C}$	83.29	75.08	0.56	78.42	68.89	18.16	14.72
	w/o $\mathcal{L}_{cls}$	46.26	39.53	2.64	40.98	34.08	13.93	11.08
	w/o $\mathcal{C}$	121.16	130.91	0.19	121.10	130.91	1.97	1.26
	Ours (full)	34.92	4.43	61.39	34.19	3.51	1.91	1.36
All	w/o RP	21.37	3.98	59.50	19.60	1.87	3.48	1.09
	w/o $\mathcal{L}_{cls}, \mathcal{C}$	70.46	57.87	2.20	64.75	52.70	18.24	14.72
	w/o $\mathcal{L}_{cls}$	30.36	19.97	26.50	25.52	12.61	9.60	5.88
	w/o $\mathcal{C}$	90.60	89.37	5.70	90.51	89.35	1.81	1.10
	Ours (full)	20.45	2.23	78.30	19.52	1.07	1.69	1.07

Table 2: **Ablation study**, evaluating the effect of our rotation parameterization (RP), classification objective ($\mathcal{L}_{cls}$) and correlation volumes ($\mathcal{C}$) on SUN360.

the other cities by 5-10°, suggesting that this baseline is more sensitive to image appearance. For overlapping pairs, we can see that Reg6D-o obtains only slightly larger median errors on the other cities (but still larger than ours), while it completely fails on non-overlapping pairs.

This experiment, and also the strong performance obtained on SUN360, a dataset constructed from Internet panoramas originating from various sources, suggest that we are not simply learning spurious correlations (*e.g.* learning pose based on panorama stitching artifacts), as our models can successfully predict relative rotations for pairs that are significantly different from those they were trained on.

4.6. Ablation Study

We perform an ablation study to analyze the effect of individual algorithmic components (see Table 2). In particular,

we replace (i) the rotation parameterization, (ii) dense correlation volumes with concatenated image features, and (iii) the relative rotation classification objective with a regression loss, regressing to a continuous representation in 6D.

For (i), instead of predicting the absolute pitch and the relative yaw, we predict the roll, pitch and yaw $[\alpha, \beta, \gamma]$ (directly decomposed from the relative rotation matrix). The evaluation suggests that our parameterization generally does not have a large effect on the final result. However, decomposing the absolute pitch per image is helpful to accurately predict the pitch angle in non-overlapping cases.

Dense correlation volumes yield significant improvements, particularly over relative yaw predictions, and boost performance over regression models, indicating that networks with correlation volumes can better reason over geometric relations between images with little to no overlap.

The impact of the classification loss is visible for both overlapping and non-overlapping cases, especially for pitch predictions. Using a regression loss instead of a classification loss also degrades yaw accuracy for overlapping cases. The significant improvements in the median error (*e.g.* 39.53° to 4.43° for non-overlapping pairs) illustrates that predicting a distribution over discretized angles enables the model to learn richer information in comparison to a regression model.

5. Conclusion

We presented a method for estimating 3D rotations between a pair of RGB images in extreme settings where the images have little to no overlap. We demonstrate our method on a wide variety of example pairs and show that our model works surprisingly well on non-overlapping pairs, and can generalize to new scenes (including new cities) compared to the training data. Our key technical contribution is the use of dense correlation volumes that perform all-pairs feature correlations for this pose estimation problem, enabling our network to detect both explicit and implicit cues. While we focus on relative rotations in our work, we believe that our method could be extended to compute whole camera poses, including camera intrinsics like focal length. However, we found that applying our approach directly to estimate relative translations did not work well for non-overlapping pairs. Addressing translations may require new architectures or by first predicting and removing rotation, as recently proposed by Chen *et al.* [7]. In the future, we would also like to explore use of our internal features for other geometric tasks, like lighting or depth prediction—we believe that the use of non-overlapping relative pose as a proxy task may yield image features that are useful for a range of other tasks.
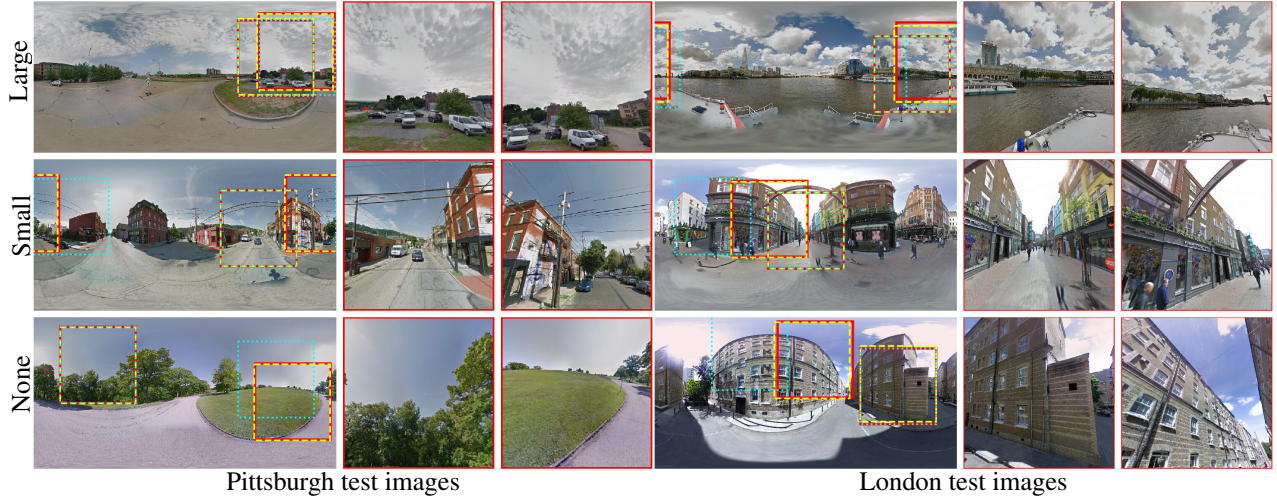
Figure 6: **Generalization from Manhattan to Pittsburgh and London.** Above we show results obtained by our StreetLearn model on image pairs from new, never-before-seen cities. The full panoramas are shown on the left with the ground-truth perspective images marked in red. We show our predicted viewpoints (in yellow) and the result by the Reg6D regression model [52] (in blue).
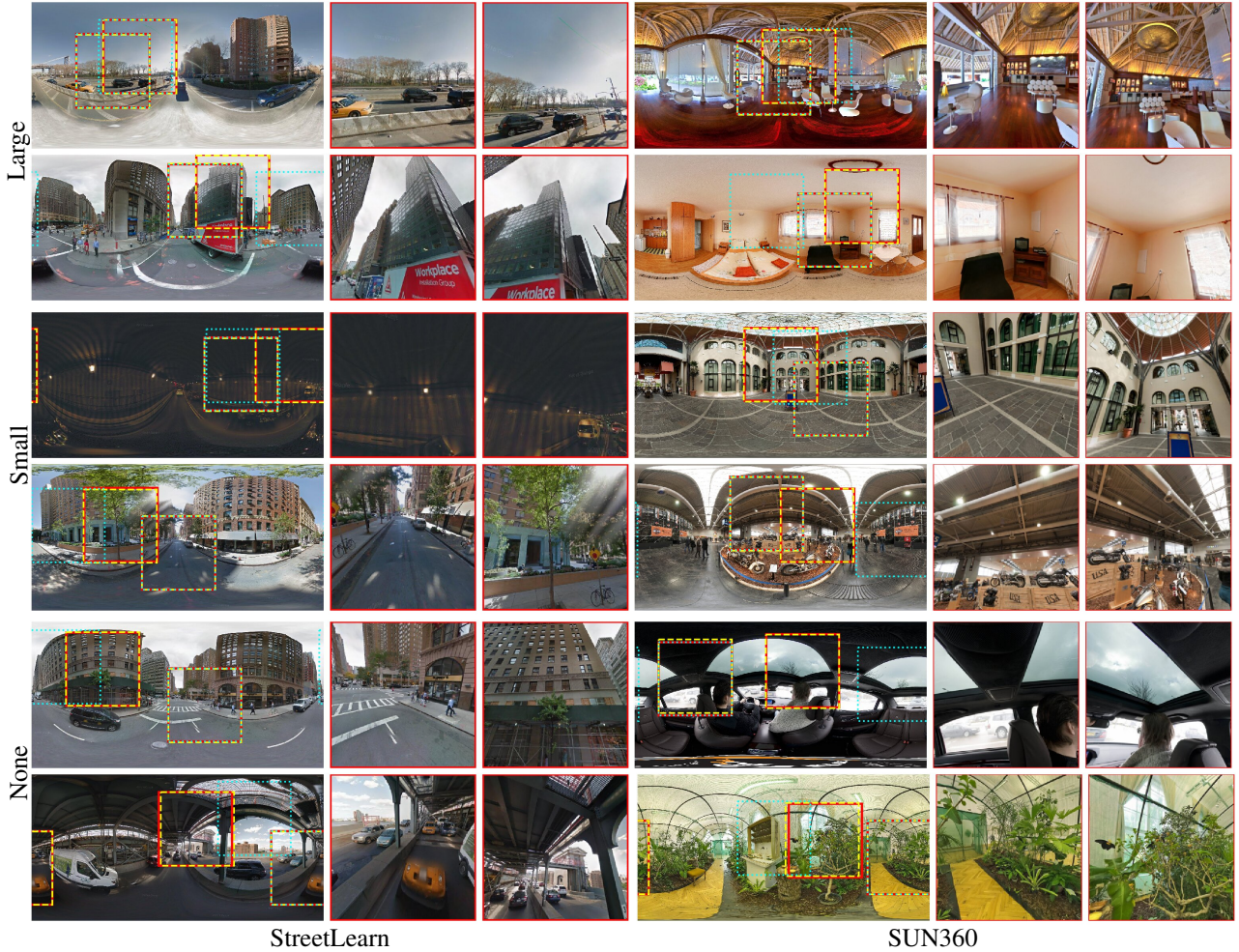

Figure 7: **Predicted rotation results.** Full panoramas are shown on the left, with the ground-truth perspective images marked in red. We show our predicted viewpoints (in yellow) and results from the Reg6D regression model [52] (in blue).

References

[1] Hadar Averbuch-Elor and Daniel Cohen-Or. Ringit: Ring-ordering casual photos of a temporal event. *ACM TOG*, 34(3), 2015.

[2] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *ECCV*, pages 404–417. Springer, 2006.

[3] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *CVPR*, pages 6684–6692, 2017.

[4] Matthew Brown, Richard I Hartley, and David Nistér. Minimal solutions for panoramic stitching. In *CVPR*, pages 1–8. IEEE, 2007.

[5] Yaron Caspi and Michal Irani. Aligning non-overlapping sequences. *IJCV*, 48(1):39–51, 2002.

[6] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *CVPR*, pages 5410–5418, 2018.

[7] Kefan Chen, Noah Snavely, and Ameesh Makadia. Wide-baseline relative camera pose estimation with directional learning. In *CVPR*, 2021.

[8] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE TPAMI*, 29(6):1052–1067, 2007.

[9] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPRW*, 2018.

[10] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *ICCV*, pages 2758–2766, 2015.

[11] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-Net: A Trainable CNN for Joint Detection and Description of Local Features. In *CVPR*, 2019.

[12] Sovann En, Alexis Lechervy, and Frédéric Jurie. Rpnet: An end-to-end network for relative camera pose estimation. In *ECCV*, pages 0–0, 2018.

[13] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM (CACM)*, 24(6):381–395, 1981.

[14] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *CVPR*, pages 2495–2504, 2020.

[15] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[16] Paul VC Hough. Method and means for recognizing complex patterns, Dec. 18 1962. US Patent 3,069,654.

[17] Hui Huang, Kangxue Yin, Minglun Gong, Dani Lischinski, Daniel Cohen-Or, Uri M Ascher, and Baoquan Chen. "mind the gap": Tele-registration for structure-driven image completion. *ACM TOG*, 32(6):174–1, 2013.

[18] Alex Kendall, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry. End-to-end learning of geometry and context for deep stereo regression. In *ICCV*, pages 66–75, 2017.

[19] Sameh Khamis, Sean Fanello, Christoph Rhemann, Adarsh Kowdle, Julien Valentin, and Shahram Izadi. Stereonet: Guided hierarchical refinement for real-time edge-aware depth prediction. In *ECCV*, pages 573–590, 2018.

[20] Zakaria Laskar, Iaroslav Melekhov, Surya Kalia, and Juho Kannala. Camera relocalization by computing pairwise relative poses using convolutional neural network. In *ICCVW*, pages 929–938, 2017.

[21] Jake Levinson, Carlos Esteves, Kefan Chen, Noah Snavely, Angjoo Kanazawa, Afshin Rostamizadeh, and Ameesh Makadia. An analysis of svd for deep rotation estimation. In *NeurIPS*, 2020.

[22] Wenbin Li, Sajad Saeedi, John McCormac, Ronald Clark, Dimos Tzoumanikas, Qing Ye, Yuzhong Huang, Rui Tang, and Stefan Leutenegger. Interiornet: Mega-scale multi-sensor photo-realistic indoor scenes dataset. In *BMVC*, 2018.

[23] Zhengfa Liang, Yulan Guo, Yiliu Feng, Wei Chen, Linbo Qiao, Li Zhou, Jianfeng Zhang, and Hengzhu Liu. Stereo matching using multi-level cost volume and multi-scale feature constancy. *IEEE TPAMI*, 2019.

[24] Etai Littwin, Hadar Averbuch-Elor, and Daniel Cohen-Or. Spherical embedding of inlier silhouette dissimilarities. In *CVPR*, pages 3855–3863, 2015.

[25] Chenxi Liu, Alex Schwing, Kaustav Kundu, Raquel Urtasun, and Sanja Fidler. Rent3d: Floor-plan priors for monocular layout estimation. In *CVPR*, 2015.

[26] David G Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[27] Yao Lu, Jack Valmadre, Heng Wang, Juho Kannala, Mehrtash Harandi, and Philip Torr. Devon: Deformable volume network for learning optical flow. In *The IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2705–2713, 2020.

[28] Iaroslav Melekhov, Juha Ylioinas, Juho Kannala, and Esa Rahtu. Relative camera pose estimation using convolutional neural networks. In *International Conference on Advanced Concepts for Intelligent Vision Systems (ACIVS)*, pages 675–687. Springer, 2017.

[29] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.

[30] Piotr Mirowski, Andras Banki-Horvath, Keith Anderson, Denis Teplyashin, Karl Moritz Hermann, Mateusz Malinowski, Matthew Koichi Grimes, Karen Simonyan, Koray Kavukcuoglu, Andrew Zisserman, et al. The StreetLearn environment and dataset. *arXiv preprint arXiv:1903.01292*, 2019.

[31] David Mohlin, Josephine Sullivan, and Gérald Bianchi. Probabilistic orientation estimation with matrix fisher distributions. *NeurIPS*, 33, 2020.

[32] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam

system. *IEEE Transactions on Robotics (IEEE T-RO)*, 31(5):1147–1163, 2015.

[33] Guang-Yu Nie, Ming-Ming Cheng, Yun Liu, Zhengfa Liang, Deng-Ping Fan, Yue Liu, and Yongtian Wang. Multi-level context ultra-aggregation for stereo matching. In *CVPR*, pages 3283–3291, 2019.

[34] Onur Ozyesil, Vladislav Voroninski, Ronen Basri, and Amit Singer. A survey of structure from motion. *arXiv preprint arXiv:1701.08493*, 2017.

[35] Valentin Peretroukhin, Brandon Wagstaff, and Jonathan Kelly. Deep probabilistic regression of elements of so (3) using quaternion averaging and uncertainty injection. In *CVPRW*, pages 83–86, 2019.

[36] Yair Poleg and Shmuel Peleg. Alignment and mosaicing of non-overlapping images. In *The IEEE International Conference on Computational Photography (ICCP)*, pages 1–8. IEEE, 2012.

[37] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE TPAMI*, 2020.

[38] Gernot Riegler and Vladlen Koltun. Free view synthesis. In *ECCV*, 2020.

[39] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016.

[40] Johannes L Schönberger, Marc Pollefeys, Andreas Geiger, and Torsten Sattler. Semantic visual localization. In *CVPR*, pages 6896–6906, 2018.

[41] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: CNNs for optical flow using pyramid, warping, and cost volume. In *CVPR*, pages 8934–8943, 2018.

[42] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense matching and view synthesis. In *CVPR*, pages 7199–7209, 2018.

[43] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, pages 402–419. Springer, 2020.

[44] Xingkui Wei, Yinda Zhang, Zhuwen Li, Yanwei Fu, and Xiangyang Xue. Deepsfm: Structure from motion via deep bundle adjustment. In *ECCV*, pages 230–247. Springer, 2020.

[45] Jianxiong Xiao, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Recognizing scene viewpoint using panoramic place representation. In *CVPR*, pages 2695–2702. IEEE, 2012.

[46] Jia Xu, René Ranftl, and Vladlen Koltun. Accurate optical flow via direct cost volume processing. In *CVPR*, pages 1289–1297, 2017.

[47] Gengshan Yang, Joshua Manela, Michael Happold, and Deva Ramanan. Hierarchical deep stereo matching on high-resolution images. In *CVPR*, pages 5515–5524, 2019.

[48] Zhenpei Yang, Jeffrey Z Pan, Linjie Luo, Xiaowei Zhou, Kristen Grauman, and Qixing Huang. Extreme relative pose estimation for rgb-d scans via scene completion. In *CVPR*, 2019.

[49] Zhenpei Yang, Siming Yan, and Qixing Huang. Extreme relative pose network under hybrid representations. In *CVPR*, 2020.

[50] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *ECCV*, pages 767–783, 2018.

[51] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014.

[52] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *CVPR*, pages 5745–5753, 2019.

[53] Yichao Zhou, Jingwei Huang, Xili Dai, Linjie Luo, Zhili Chen, and Yi Ma. HoliCity: A city-scale data platform for learning holistic 3D structures. 2020. arXiv:2008.03286 [cs.CV].

[54] Yichao Zhou, Shichen Liu, and Yi Ma. Learning to detect 3d reflection symmetry for single-view reconstruction. *arXiv preprint arXiv:2006.10042*, 2020.