Progressive Neighbor Consistency Mining for Correspondence Pruning

Xin Liu and Jufeng Yang*
TMCC, College of Computer Science, Nankai University, China
xinliu_0209@163.com, yangjufeng@nankai.edu.cn

Abstract

The goal of correspondence pruning is to recognize correct correspondences (inliers) from initial ones, with applications to various feature matching based tasks. Seeking neighbors in the coordinate and feature spaces is a common strategy in many previous methods. However, it is difficult to ensure that these neighbors are always consistent, since the distribution of false correspondences is extremely irregular. For addressing this problem, we propose a novel global-graph space to search for consistent neighbors based on a weighted global graph that can explicitly explore long-range dependencies among correspondences. On top of that, we progressively construct three neighbor embeddings according to different neighbor search spaces, and design a Neighbor Consistency block to extract neighbor context and explore their interactions sequentially. In the end, we develop a Neighbor Consistency Mining Network (NCMNet) for accurately recovering camera poses and identifying inliers. Experimental results indicate that our NCMNet achieves a significant performance advantage over state-of-the-art competitors on challenging outdoor and indoor matching scenes. The source code can be found at https://github.com/xinliu29/NCMNet.

1. Introduction

Estimating high-quality feature correspondences between two images is of crucial significance to numerous computer vision tasks, such as visual simultaneous localization and mapping (SLAM) [33], structure from motion (SfM) [41, 49], image fusion [31], and image registration [48, 51]. Off-the-shelf feature extraction methods [4, 29, 52] can be employed to establish initial correspondences. Due to complex matching situations (*e.g.*, severe viewpoint variations, illumination changes, occlusions, blurs, and repetitive structures), a great number of false correspondences, called outliers, are inevitable [19, 30]. To mitigate the negative impact of outliers for downstream tasks, correspondence pruning [5, 24, 53] can be implemented to further identify correct correspondences, also known as inliers, from initial ones. However, unlike images that contain sufficient information, *e.g.*, texture and RGB information, correspondence pruning is extremely challenging since the spatial positions of initial correspondences are discrete and irregular [55].

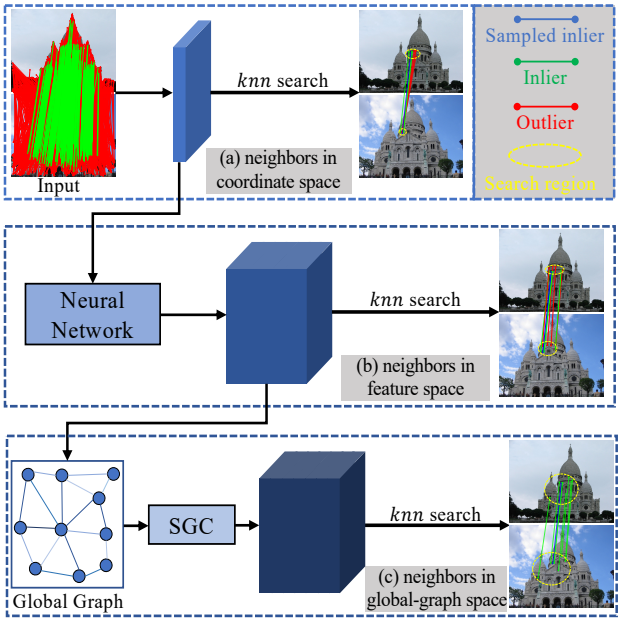


Figure 1. The acquisition process and visual comparison of (a) spatial neighbors, (b) feature-space neighbors, and (c) global-graph neighbors. **SGC**: the spectral graph convolution operation.

Intuitively, inliers commonly conform to consistent constraints (*e.g.*, lengths, angles, and motion) under the 2D rigid transformation, while outliers are randomly distributed, see the top left of Fig. 1. Therefore, the consistency of correspondences as a vital priori knowledge has been studied extensively to distinguish inliers and outliers [9, 13, 14], in which the neighbor consistency has received widespread attention [25, 26, 57]. For well-defined neighbors, previous approaches [5, 8, 27] employ k-nearest neighbor (knn) search in the coordinate space of raw correspondences to seek spatially consistent neighbors, denoted

* Corresponding Author

as spatial neighbors. Subsequently, several works, such as CLNet [57] and MS^{2}DG-Net [11], search for feature-consistent neighbors ($i.e.$, feature-space neighbors) by performing knn search in the feature space learned from the neural network. They devise various strategies to exploit the neighbor consistency of correspondences, and show satisfactory progress. Nevertheless, there may exist numerous outliers in the vicinity of an inlier due to the random distribution of outliers, especially for the challenging matching scenes. As shown in Fig. 1, in the coordinate and feature spaces, the searched neighbors of a sampled inlier (blue line) always contain some unexpected outliers (red line).

To tackle this issue, we propose a new global-graph space to seek consistent neighbors for each correspondence. Inspired by the fact that inliers have strong consistency at a global level [13, 22, 23], we first construct a weighted global graph, in which nodes denote all correspondences and edges represent their pairwise affinities calculated by the preliminary inlier scores. The dependence between two correspondences is determined to be tight if they have high scores simultaneously. Next, we use a spectral graph convolution operation [21,57] to further explore long-range dependencies among correspondences and increase the discrimination between inliers and outliers. We finally adopt knn search in the global-graph space to search for globally consistent neighbors, called global-graph neighbors as illustrated in Fig. 1(c). Noteworthily, the positions of global-graph neighbors are not required to be spatially close to the sampled inlier. In other words, this kind of neighbor has a large search region (see the ablation for quantitative results) due to our global operation.

Moreover, a single type of neighbor is inadequate for all complex matching situations. Therefore, we present a new Neighbor Consistency (NC) block to take full advantage of three types of neighbors and improve the robustness. Specifically, we progressively construct three neighbor embeddings according to the spatial, feature-space, and global-graph neighbors. To extract corresponding neighbor context and explore their interactions, we design two successive layers, $i.e.$, Self-Context Extraction (SCE) layer and Cross-Context Interaction(CCI) layer. The SCE layer is responsible for dynamically capturing intra-neighbor relations and aggregating their context, while the CCI layer fuses and modulates inter-neighbor interactive information. Finally, an effective Neighbor Consistency Mining Network (NCMNet) is developed to achieve correspondence pruning.

Our contributions are three-fold: (1) Based on the fact that inliers have strong consistency at a global level, we propose a novel global-graph space to seek consistent neighbors for each correspondence. (2) We present a new NC block to progressively mine the consistency of three types of neighbors by extracting intra-neighbor context and exploring inter-neighbor interactions in a sequential manner. (3) We develop an effective NCMNet for correspondence pruning, obtaining considerable performance gains when compared to state-of-the-art works.

2. Related Work

RANSAC-Related Methods. RANSAC [13] is one of the most renowned handcrafted techniques over the past decades, which adopts a hypothesize-and-verify framework. To be specific, RANSAC iteratively samples a minimal subset of data to hypothesize a parametric model, and then verifies the model's reliability by counting the number of supported inliers. Based on this framework, its variants, such as MLESAC [44], USAC [37], and MAGSAC [2], utilize different strategies to improve the efficiency and effectiveness. These methods are still viewed as standard solutions for getting accurate inliers and estimating reliable parametric models. However, they are sensitive to outliers [18, 30, 56], therefore, their performance will be limited when initial correspondences are heavily contaminated by outliers.

Learning-Based Methods. With the booming of deep learning [15, 17, 46, 58], some pioneer works, such as DSAC [6], LFGC [53], and DFE [38], adopt neural networks to remove outliers, and obtain competitive results. Particularly, inspired by PointNet [35,36], LFGC [53] casts the correspondence pruning as a labeling outlier/inlier task and a regressing essential matrix task, which designs a permutation-equivariant network structure based on Multi-Layer Perceptrons (MLPs) to effectively process irregular and unordered data. By taking initial correspondences as inputs, this network is able to predict the inlier weights and corresponding essential matrix. Follow-up works using this de facto standard improves the network performance by designing different network structures. For example, OANet [54] clusters input correspondences based on a differentiable pooling operator to exploit local context. The original order of correspondences is then recovered by a differentiable unpooling operation. To obtain both local and global contexts, ACNe [42] presents a simple attentive context normalization. T-Net [60] devises a T-shaped structure for adequately integrating the output features of all sub-networks. The above-mentioned methods implicitly capture contextual information through well-designed network structures. However, they rarely explore geometric properties of correspondences, and remain vulnerable to the negative effect of numerous outliers.

Consistency of Correspondences. Inliers of two matching images tend to be consistent while outliers are disorganized [14] under the 2D rigid transformation. Exploring the consistency of correspondences to remove outliers has gained extensive attention in past decades [28, 30, 34]. For example, BF [23] and CODE [22] leverage the global consistency to discern the difference between inliers and out-

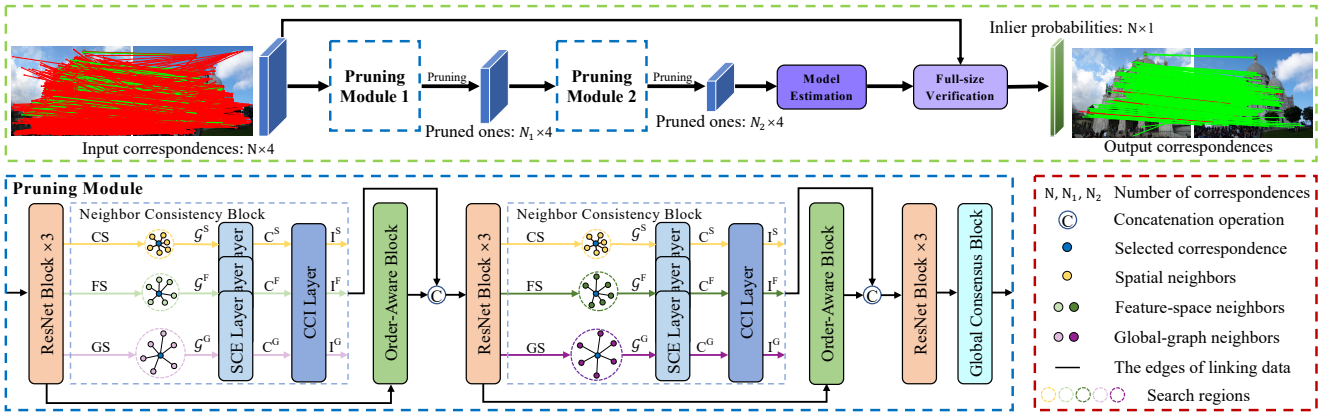


Figure 2. Pipeline of our NCMNet. It takes $N \times 4$ initial correspondences as inputs and outputs $N \times 1$ inlier probabilities by an iterative pruning strategy, which distills more reliable candidates to estimate the parametric model. Each pruning module contains several off-the-shelf network structures and the newly proposed Neighbor Consistency block (**CS**: the coordinate space, **FS**: the feature space, **GS**: the global-graph space).

liers. GMS [5] and LPM [32] exploit the local consistency by finding consistent spatial neighbors. Although these traditional methods have demonstrated decent performance, they still have difficulty in facing challenging matching scenes and require elaborate parameter tuning. Recently, several correspondence pruning works explore the consistency and aggregate context in a learning-based manner. They seek consistent neighbors for each correspondence, such as spatial neighbors in LMCNet [27], compatibility-specific neighbors in NM-Net [55], as well as feature-space neighbors in CLNet [57] and MS^{2}DG-Net [11]. They then design different network modules or learning paradigms to aggregate neighbor information. However, these neighbors are inadequate due to the extremely irregular distribution of plentiful outliers. Therefore, we develop a new global-graph space with a large neighbor search region by exploring long-range dependencies between correspondences. We further design a Neighbor Consistency block to accomplish the context extraction and interaction of the neighbors in different spaces.

3. Methodology

3.1. Problem Formulation

Given a matching image pair, we can utilize any existing feature extraction methods [12, 29] to detect feature keypoints and construct descriptors. Then, a set $S = \{s_1, s_2, ..., s_N\} \in \mathbb{R}^{N\times 4}$ containing N initial correspondence can be set up by the brute-force matching of descriptor similarities. s_i denotes the i-th initial correspondence that contains two normalized coordinates in two matching images by camera intrinsics. In practice, the set S usually has a high proportion of outliers, therefore, our aim is to identify inliers and eliminate outliers as much as possible.

To achieve this goal, we develop an effective Neighbor Consistency Mining Network (NCMNet) as illustrated in Fig. 2. The iterative pruning strategy [57] is utilized as our main framework, which has the ability to explicitly reduce the adverse impact caused by numerous outliers. To be specific, our NCMNet first uses two sequential pruning modules to process the input set S, and outputs corresponding results $(S_1, o_1) = f_{\theta 1}(S)$ and $(S_2, o_2) = f_{\theta 2}(S_1)$, in which $S_1 \in \mathbb{R}^{N_1\times 4}$ and $S_2 \in \mathbb{R}^{N_2\times 4}$ are two pruned correspondence sets, where $N > N_1 > N_2$. $f_{\theta 1}(\cdot)$ and $f_{\theta 2}(\cdot)$ with relevant parameters $\theta 1$ and $\theta 2$ denote two sequential pruning modules, where o_1 and o_2 represent their final logit values. o_2 is additionally processed by a ResNet block and an MLP layer to compute the inlier weight set w_2 as an auxiliary input of model estimation. Next, we estimate a parametric model (*i.e.*, essential matrix $\hat{E}$) according to the set S_2 and w_2. Finally, we utilize $\hat{E}$ to do a full-size verification on the set S, which can avoid some inliers to be removed incorrectly in the pruning process. Our whole framework can be expressed as:

$$\hat{E} = g(S_2, w_2), \tag{1}$$

$$w = v(\hat{E}, S), \tag{2}$$

where weighted eight-point algorithm $g(\cdot)$ is used for model estimation [53, 57]. $v(\cdot)$ represents a full-size verification operation using the epipolar constraint [14]. w denotes the inlier probabilities of all input correspondences.

3.2. Global-Graph Space

Leveraging the priori knowledge that inliers usually have strong consistency to each other while outliers scatter randomly, we explore the neighbor consistency for better distinguishing both inliers and outliers. In this paper, we utilize three different neighbor search spaces to seek consis-

tent neighbors. The coordinate space is the network input $S \in \mathbb{R}^{N\times 4}$, where the last dimension is omitted for simplicity. The feature space is the middle feature map $F \in \mathbb{R}^{N\times d}$ processed by several ResNet blocks, where d is the channel dimension. We perform knn search in the S and F to search for spatial k-nearest neighbors and feature-space k-nearest neighbors, respectively, for each correspondence s_i. Meanwhile, we propose a global-graph space for complementing the two spaces.

Specifically, we first compute the preliminary inlier weights w^p according to the F:

$$w^p = \mathrm{ReLU}(\tanh(\mathrm{MLP}(F))), \tag{3}$$

where $\mathrm{MLP}(\cdot)$ is an MLP layer for reducing the channel dimension to 1. Activation functions $\tanh(\cdot)$ and $\mathrm{ReLU}(\cdot)$ are used for computing weights. Then, we construct a weighted global graph $\mathcal{G}^g = \{\mathcal{V}^g, \mathcal{E}^g\}$, in which nodes $\mathcal{V}^g$ represent all correspondences, and undirected edges $\mathcal{E}^g$ link each correspondence pair by the associated weights $w^p_{ij} = w^p_i \cdot w^p_j, 1 \le i, j \le N$. A high association can be established only when two correspondences have high inlier weights simultaneously, otherwise there will be weak or no link. Therefore, we can construct a weighted adjacency matrix $A = w^p_{ij} \in \mathbb{R}^{N\times N}$, which explicitly describes long-range dependencies between correspondences. Finally, we use the spectral graph convolution operation [21, 57] to obtain our global-graph space:

$$F^g = \sigma(LFW^g), \tag{4}$$

where $L = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$ is the graph Laplacian matrix that modulates the F into the spectral domain. $\tilde{A} = A + I_N$ is the adjacency matrix with an added self-connection, and I_N represents a diagonal identity matrix to avoid the degeneracy of this formulation. $\tilde{D}_{ii} = diag(\sum_j \tilde{A}_{ij})$ denotes the diagonal degree matrix of $\tilde{A}$. W^g is the trainable weight. $\sigma(\cdot)$ represents an activation function ($e.g.$, $\mathrm{ReLU}(\cdot)$ in this paper). $F^g \in \mathbb{R}^{N\times d}$ is our global-graph space, which reflects the consistency of correspondences well from the global aspect, especially for inliers since they have high associations with each other. We can acquire global-graph k-nearest neighbors of each correspondence s_i by performing knn search in the F^g. Noteworthily, our global-graph neighbors have a large neighbor search region due to the gains of long-range dependencies.

3.3. Neighbor Consistency Block

To cope with complex matching situations, we propose a Neighbor Consistency (NC) block to progressively mine the consistency of different types of neighbors. Our NC block has three key parts: neighbor embedding construction, Self-Context Extraction (SCE) layer, and Cross-Context Interaction(CCI) layer.

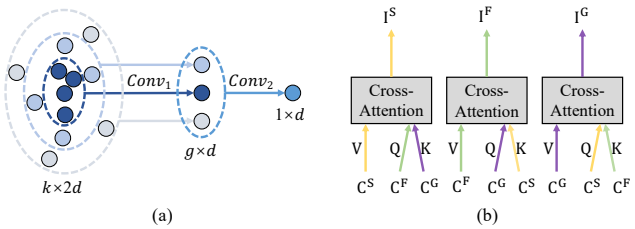


Figure 3. (a) The grouped convolution manner in the SCE layer. (b) The details of the CCI layer.

Neighbor embedding construction. We construct three individual neighbor embeddings $\mathcal{G}^S_i = \{\mathcal{V}^S_i, \mathcal{E}^S_i\}$, $\mathcal{G}^F_i = \{\mathcal{V}^F_i, \mathcal{E}^F_i\}$ and $\mathcal{G}^G_i = \{\mathcal{V}^G_i, \mathcal{E}^G_i\}$ for each correspondence s_i according to its spatial, feature-space and global-graph neighbors. For a $\mathcal{G}^S_i$, nodes $\mathcal{V}^S_i = \{s^S_{i1}, ..., s^S_{ik}\}$ denote the spatial k-nearest neighbors of s_i, and directed edges $\mathcal{E}^S_i = \{e^S_{i1}, ..., e^S_{ik}\}$ link s_i and its spatial neighbors in $\mathcal{V}^S_i$. We use the same edge construction as [47, 57]:

$$e^S_{ij} = [f_i, f_i - f^S_{ij}], j = 1, 2, ..., k \tag{5}$$

where f_i, f^S_{ij} are feature maps of the correspondence s_i and its j-th spatial neighbor s^S_{ij} in the $F = \{f_1, f_2, ..., f_N\}$. $f_i - f^S_{ij}$ denotes the residual feature map. $[\cdot, \cdot]$ represents the concatenation operation along the channel dimension. $\mathcal{G}^S \in \mathbb{R}^{N\times k\times 2d}$ is the spatial neighbor embedding of all correspondences. Similarly, we can obtain the feature-space neighbor embedding $\mathcal{G}^F \in \mathbb{R}^{N\times k\times 2d}$ and the global-graph neighbor embedding $\mathcal{G}^G \in \mathbb{R}^{N\times k\times 2d}$.

SCE layer. Once the three neighbor embeddings are constructed, we need to consider how to effectively mine intra-neighbor consistency information. A straightforward way is to employ popular pooling operations, $e.g.$, average-pooling and max-pooling. However, these operations may discard the underlying relationships among graph nodes. Therefore, to take full advantage of the graph structure of neighbor embeddings, we propose an SCE layer. Considering the fact that graph nodes are sorted by the similarity principle of knn search within different spaces, the SCE layer utilizes a grouped convolution manner [57] to dynamically capture the relationships of neighbors and aggregate the neighbor context along graph edges.

Concretely, as illustrated in Fig. 3(a), for the graph $\mathcal{G}_i \in \mathbb{R}^{k\times 2d}$ of correspondence s_i, its nodes are divided into g groups depending on the affinities to the anchor node, where each group has k/g nodes. We utilize two successive convolution layers followed by one Batch Normalization (BN) [16] layer with ReLU to process the graph. This operation can be written as:

$$C_i = (Conv_2(Conv_1(\mathcal{G}_i)), \tag{6}$$

where $Conv_1(\cdot)$ and $Conv_2(\cdot)$ represent the convolution layers with $1 \times \frac{k}{g}$ kernels and $1 \times g$ kernels, respectively.

We omit the BN and ReLU for simplicity. $C_i \in \mathbb{R}^{1\times d}$ is the output result of graph $\mathcal{G}_i$. In our NC block, we utilize three parallel SCE layers to individually process three neighbor embeddings and get three corresponding neighbor context features $\{C^S, C^F, C^G\} \in \mathbb{R}^{N\times d}$.

CCI layer. When three neighbor context features are obtained, we want to fuse and modulate inter-neighbor information in a collaborative manner. Therefore, we develop a CCI layer based on a cross-attention operation as shown in Fig. 3(b). In detail, our CCI layer has three parallel cross-attention branches, and each of them takes one neighbor context feature as values V, while the other two features are treated as queries Q and keys K. Similar to self-attention [45], we first generate $\mathrm{V} \in \mathbb{R}^{N\times d}$ and $\{\mathrm{Q}, \mathrm{K}\} \in \mathbb{R}^{N\times \frac{d}{r}}$ according to corresponding neighbor context features via an individual MLP layer followed by one BN layer with ReLU, where r is the channel reduction ratio for reducing parameters. Then, we use a matrix multiplication between Q and the transpose of K followed by a softmax function to calculate an attention weight matrix $A_w \in \mathbb{R}^{N\times N}$. In the end, we utilize the A_w, which measures the correlation between correspondences obtained from two neighbor context features, to enhance the V. The example of the first cross-attention branch can be formulated as:

$$I^S = \alpha(MLPs(A_w \mathrm{V})) + C^S, \tag{7}$$

where $MLPs(\cdot)$ contains one MLP layer, one BN layer, and one ReLU. α is a learned weight for controlling the influence, which is initialized to 0. I^S is the final output of the first cross-attention branch, where the response of each position is a weighted sum calculated from the other two features at all positions. Therefore, the inliers in three neighbor context features can achieve mutual gains, thus further improving the difference between correspondences. Similarly, we can obtain the outputs I^F and I^G. Three neighbor interaction features $\{I^S, I^F, I^G\} \in \mathbb{R}^{N\times d}$ are final outputs of our NC block.

3.4. Neighbor Consistency Mining Network

As illustrated in Fig. 2, NCMNet comprises two core pruning modules to distill reliable candidates for estimating accurate essential matrix and inlier probabilities. Each pruning module consists of some existing network structures ($i.e.$, ResNet block [53], Order-Aware block [54], and Global Consensus block [57]) and our proposed NC block. ResNet block is a basic correspondence processing structure containing two MLP layers and some normalization operations. Order-Aware block is designed for implicitly capturing local and global contexts by a clustering manner, in which the number of clusters is set to 250. Global Consensus block encodes global context based on Graph Convolutional Network [21] to estimate final logit values for pruning correspondences. It is worth noting that the feature-space and global-graph neighbors found by learnable spaces are dynamic. Therefore, we propose a progressive neighbor refinement processing ($i.e.$, using two NC blocks in each pruning module) to increase the reliability of neighbors and extract rich neighbor context.

3.5. Loss Function

The proposed NCMNet is optimized by a hybrid loss as benchmarks [53, 54]:

$$\mathcal{L} = \mathcal{L}_c(o_m, y_m) + \beta \mathcal{L}_e(E, \hat{E}), \tag{8}$$

where β is a weighting parameter. The classification loss $\mathcal{L}_c(\cdot)$ is formulated as:

$$\mathcal{L}_c(o_m, y_m) = \sum_{m=1}^{M} H(\tau_m \odot o_m, y_m), \tag{9}$$

where $H(\cdot)$ is a binary cross entropy loss. $\odot$ represents the Hadamard product. o_m denotes the logit value of the m-th pruning module. y_m is the ground-truth correspondence label determined by epipolar distances with a threshold of 10^{-4}. τ_m is an adaptive temperature vector [57] for alleviating the influence of label ambiguity. M is the number of pruning modules. The regression loss $\mathcal{L}_e(\cdot)$ is a geometry loss [38], which is formulated as follows:

$$\mathcal{L}_e(E, \hat{E}) = \frac{(p'^T \hat{E} p)^2}{\|Ep\|^2_{[1]} + \|Ep\|^2_{[2]} + \|E^T p'\|^2_{[1]} + \|E^T p'\|^2_{[2]}}, \tag{10}$$

where $c_{[i]}$ is the i-th element of vector c. Virtual correspondence coordinates p and p' are generated by the ground truth essential matrix E.

4. Experiments

4.1. Evaluation Protocols

Datasets. We construct experiments to showcase the capability of NCMNet on outdoor and indoor scenes as benchmark [54]. The YFCC100M [43] dataset from Yahoo, which contains 100 million tourist images, has been utilized as the outdoor scene. The SUN3D [50] dataset including a large number of video frames has been selected as the indoor scene. We test methods on known and unknown scenes following the data division of [54].

Evaluation metrics. The error metrics are determined by the angular differences between the estimated rotation/translation vectors (recovered from the essential matrix) and the ground truth ones. We utilize mAP with different thresholds as the evaluation metric of methods.

4.2. Implementation Details

In our implementation, SIFT [29] has been adopted to establish $N = 2000$ initial correspondences, and channel

Table 1. Quantitative comparison results on YFCC100M [43] and SUN3D [50]. mAP5° (%) on known and unknown scenes is given. **Bold** indicates the best.

Methods	YFCC100M		SUN3D	
	Known	Unknown	Known	Unknown
RANSAC [13]	30.19	40.83	19.13	14.57
DEGENSAC [10]	21.00	27.65	16.01	11.01
GC-RANSAC [1]	30.43	41.58	18.86	14.14
MAGSAC [2]	32.80	41.61	20.35	16.24
MAGSAC++ [3]	30.48	40.95	18.90	14.19
LFGC [53]	16.87	25.95	11.55	09.30
DFE [38]	18.02	30.29	14.44	12.34
OANet++ [54]	33.96	38.95	20.86	16.18
ACNe [42]	29.17	33.06	18.86	14.12
SuperGlue [40]	35.00	48.12	22.50	17.11
LMCNet [27]	33.73	47.50	19.92	16.82
T-Net [60]	41.33	48.20	22.38	17.24
MS^{2}DG-Net [11]	39.68	48.20	22.20	17.84
MSA-Net [59]	39.53	50.65	18.64	16.86
CLNet [57]	39.16	53.10	20.35	17.03
NCMNet	**52.33**	**63.43**	**26.12**	**20.66**

dimension d is 128. For the iterative pruning strategy [57], we utilize two sequential pruning modules with a pruning rate of 0.5. The neighbor number k is empirically set to 9 and 6 in two pruning modules, respectively. In the SCE layer of two pruning modules, the number of groups g is set as 3 and 2, respectively. The channel reduction ratio r in the CCI layer is set to 4. Following [54], we adopt Adam [20] optimizer with a batchsize of 32 and a fixed learning rate of 10^{-3} to train networks. The training period is set to 500k iterations. The weight parameter β in Eq. 8 is initialized to 0, and subsequently fixed as 0.5 after the first 20k iterations.

4.3. Comparisons

We compare NCMNet with some advanced works, including traditional methods [1–3, 10, 13] and learning-based methods [11, 27, 38, 40, 42, 53, 54, 57, 59, 60]. For traditional works, we remove the poor initial correspondences by adopting the ratio test [29] with a threshold of 0.8, since their performance drops extremely with the high ratio outliers. We employ the released model of SuperGlue and retrain the other network models.

The comparative results on YFCC100M and SUN3D are provided in Table 1. It is apparent that the proposed NCMNet achieves the most exceptional performance in all settings. For example, our method obtains outstanding performance improvements over the second-best works by 11.00% and 10.33% on both known and unknown outdoor scenes, respectively. Our NCMNet also has significant performance gains on indoor scenes compared with all baselines.

Table 2. Performance comparisons when using SIFT [29] and SuperPoint [12] on unknown YFCC100M [43]. mAP5° **without/with** RANSAC [13] as a post-processing step is reported.

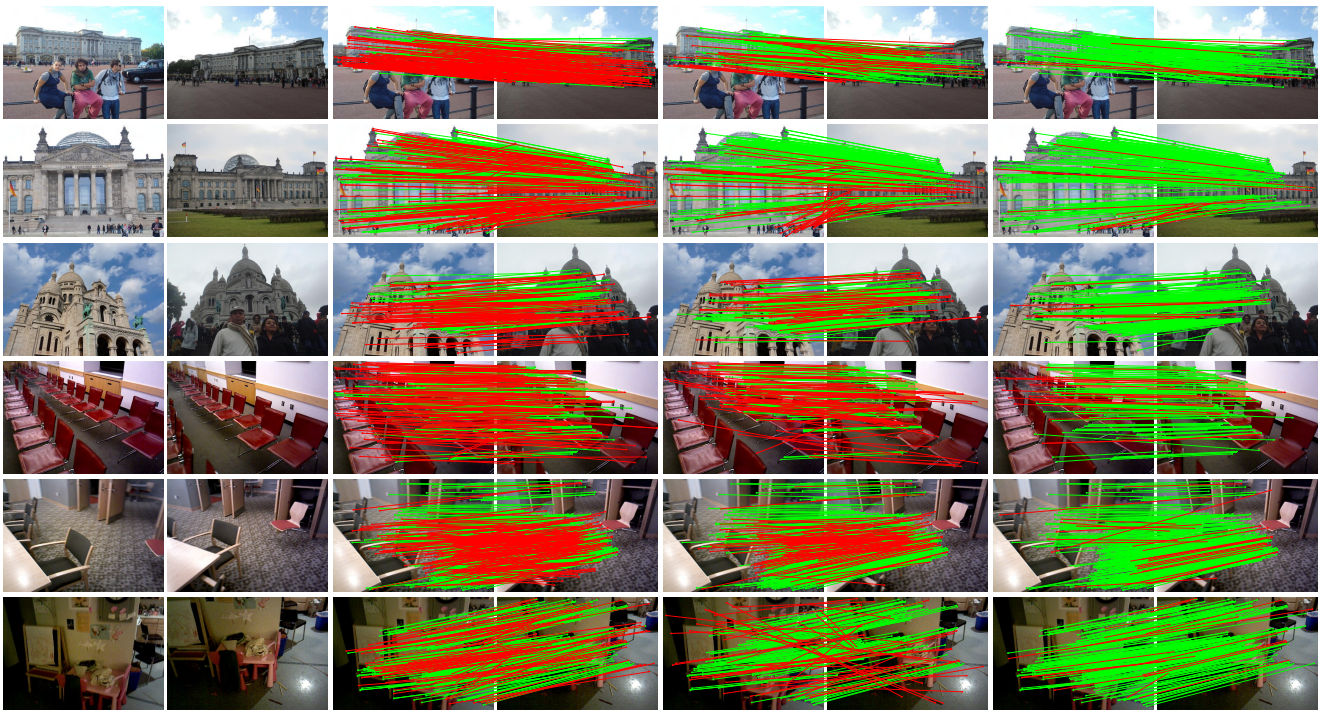
Methods	SIFT [29]		SuperPoint [12]	
	-	RANSAC	-	RANSAC
RANSAC [13]	-	40.83	-	34.38
LFGC [53]	25.95	50.00	24.25	42.57
OANet++ [54]	38.95	52.59	35.27	45.45
T-Net [60]	48.20	55.85	40.08	47.83
MS^{2}DG-Net [11]	48.20	57.15	37.38	46.48
MSA-Net [59]	50.65	56.28	38.53	47.50
CLNet [57]	53.10	59.13	39.19	48.15
NCMNet	**63.43**	**63.33**	**48.20**	**52.20**

Table 3. Generalization ability of networks on YFCC100M [43] and PhotoTourism [19] with different feature extraction methods, including ORB, SuperPoint(SP), and SIFT. mAP5° is reported.

Methods	YFCC100M		PhotoTourism	
	ORB [39]	SP [12]	SIFT [29]	SP [12]
LFGC [53]	7.88	15.48	14.37	10.78
OANet++ [54]	11.58	21.50	32.11	23.26
T-Net [60]	13.70	23.13	41.87	28.87
MS^{2}DG-Net [11]	13.00	22.85	38.20	27.64
CLNet [57]	14.70	26.78	39.47	20.30
NCMNet	**19.95**	**33.20**	**54.73**	**30.60**

We further take into account the case of using a learning-based feature extraction method to establish the initial correspondence set. We employ SuperPoint [12], which designs a fully-convolutional model to detect pixel-level keypoints and construct corresponding descriptors by a self-supervised framework. Meanwhile, a robust model estimator RANSAC [13] with a threshold of 0.001 has been adopted as a post-processing step of learning-based methods. The comparative results on unknown YFCC100M are shown in Table 2. Our NCMNet still obtains the best results when initial correspondences are established by SuperPoint. In addition, RANSAC as post-processing is able to further increase the performance, especially for those methods that perform poorly ($e.g.$, LFGC and OANet++). However, the performance of NCMNet with RANSAC slightly drops when using SIFT. This is because our method has obtained sufficiently accurate correspondence results, therefore, RANSAC cannot further distill suitable inliers to improve the accuracy of estimated camera poses.

Furthermore, we evaluate the generalization ability of networks for various datasets with different combinations of feature extraction methods. PhotoTourism is a phototourism dataset derived from the image matching challenge



Original image pair (a) OANet++ [54] (b) CLNet [57] (c) NCMNet

Figure 4. Visualization results of correspondence pruning. The top three examples derive from unknown YFCC100M and the rest examples come from unknown SUN3D. Outliers (red lines) and inliers (green lines) are exhibited.

Table 4. Ablation studies regarding performance gains of the key components in each pruning module. **IPS**: the iterative pruning strategy. **SCE**: the Self-Context Extraction layer. **CCI**: the Cross-Context Interaction layer. **PNR**: the progressive neighbor refinement processing. **OA**: the Order-Aware block.

IPS	SCE	CCI	PNR	OA	mAP5°	mAP20°
✓					53.10	76.11
✓	✓				56.50	78.34
✓	✓	✓			58.63	80.03
✓	✓	✓	✓		61.73	81.46
✓	✓	✓	✓	✓	63.43	82.46

benchmark [19]. ORB [39] is a fast and accurate detector-descriptor technique based on BRIEF [7]. All network models are trained on YFCC100M with SIFT, where the pruning rate in CLNet and NCMNet is set as 1 when using SuperPoint. As shown in Table 3, our NCMNet produces superior results in all settings, owing to the extraction and interaction of different types of neighbors. This demonstrates the robustness of our approach to different matching situations.

The visualized comparison results of NCMNet and the other two baselines [54, 57] for correspondence pruning are shown in Fig. 4. For challenging outdoor and indoor matching scenes, such as large viewpoint variations, illumination changes, textureless objects, and repetitive structures, our method obtains reliable pruning results.

4.4. Ablation Studies

We further construct ablation studies to examine the contributions of different components in the proposed NCMNet on the unknown YFCC100M [43] dataset. Here, we adopt both mAP5° and mAP20° to evaluate methods.

Main components. In our NCMNet, we utilize the iterative pruning strategy [57] as the network framework. We therefore evaluate performance gains of the main components in each pruning module over the baseline [57]. The SCE layer is used for extracting the intra-neighbor context, meanwhile the CCI layer is designed to explore the inter-neighbor interaction. We adopt the progressive neighbor refinement processing to improve the reliability of dynamic neighbors, and the Order-Aware block to implicitly capture local and global contextual information. As reported in Table 4, the performance gradually improves as the SCE layer and CCI layer are incrementally added to the baseline. When employing the progressive neighbor refinement processing and combining the Order-Aware block, our method achieves the best performance improvements.

Three types of neighbors. We give a visual comparison of three types of neighbors as illustrated in Fig. 1.

Table 5. The effectiveness of concurrently using three types of neighbors. **SN**: the spatial neighbors. **FN**: the feature-space neighbors. **GN**: the global-graph neighbors.

	Three SN	Three FN	Three GN	SN+FN+GN
mAP5°	61.40	62.60	61.73	63.43
mAP20°	81.26	81.74	81.31	82.46

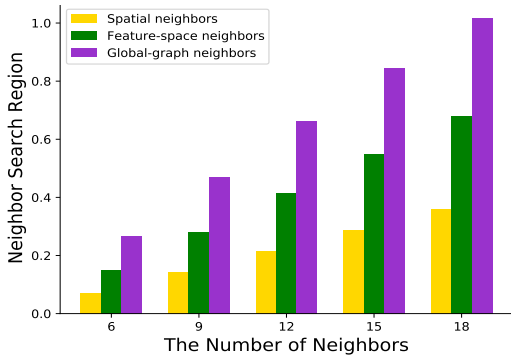


Figure 5. The illustration of mean neighbor search region (%) for all inliers in terms of different neighbor numbers of k.

Here, we report quantitative results on the mean neighbor search region of all inliers in Fig. 5. For different numbers of k-nearest neighbors, the global-graph neighbors of inliers have larger neighbor search regions than the other two due to the consideration of long-range dependencies among correspondences. Moreover, to demonstrate that employing three types of neighbors simultaneously is rational, we use three same neighbor embeddings in the NC block for comparison. Table 5 reports the comparative results. When only one type of neighbor is used, the network's performance degrades, which demonstrates the complementarity of the three types of neighbors.

Neighbor context aggregation. We design a grouped convolution manner to dynamically extract the neighbor context of each neighbor embedding in the SCE layer. Here, we compare it with some other aggregation manners, including the average-pooling layer, max-pooling layer, and convolution layer with $1 \times k$ kernels. The comparative results are shown in Table 6, our grouped convolution manner outperforms all competitors with a suitable model size, indicating its efficacy.

Inlier ratio of inputs. The inlier ratio (ir) of inputs can greatly affect the performance of traditional methods, such as RANSAC [13] and its variants [1, 2, 10]. As a result, we test the influence of the inlier ratio for our network as illustrated in Fig. 6. To set up initial correspondences with different inlier ratios as network inputs, we use Lowe's ratio test (rt) [29] with different thresholds during descriptor matching, where network models are retrained under corresponding training sets. As opposed to traditional methods, our method with the low inlier ratios works well. While the ratio test is useful for reducing outliers of inputs, it has the unintended consequence of discarding many important inliers, hence diminishing overall accuracy. The results also indicate that our network is more suitable for challenging scenarios, $i.e.$, there exist many outliers but sufficient inliers in initial correspondences.

Table 6. Quantitative comparisons of different context aggregation manners in the SCE layer. **"Avg-pooling & MLPs"** aggregates neighbor context with an average-pooling layer and two successive MLP layers with BN and ReLU. **"Max-pooling"** indicates a max-pooling layer. In addition, the model size (MB) is reported.

	mAP5°	mAP20°	Size
Avg-pooling & MLPs	61.48	81.53	3.68
Max-pooling & MLPs	62.75	81.86	3.68
$1 \times k$ kernels Conv.	62.88	81.91	6.04
Grouped Conv.	63.43	82.46	4.77

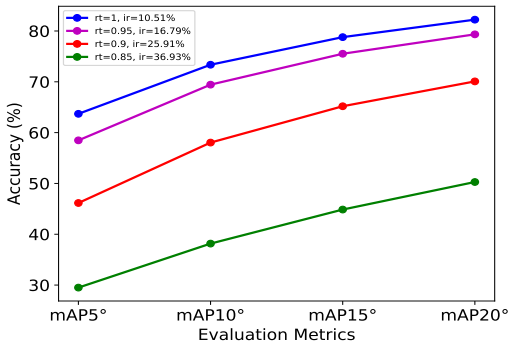


Figure 6. Influence of different inlier ratios of inputs. mAP with different error thresholds is reported.

5. Conclusion

In this paper, we develop an effective Neighbor Consistency Mining Network (NCMNet) for challenging correspondence pruning. We propose a novel global-graph space, which explicitly captures long-range dependencies among correspondences, to seek consistent neighbors. For adapting various matching situations, we further design a new Neighbor Consistency block that progressively mines the consistency of different types of neighbors. We construct extensive experiments on public benchmarks to verify NCMNet's effectiveness and generalization ability, showing remarkable superiority over the state-of-the-arts.

Acknowledgments. This work was supported by the National Key Research and Development Program of China Grant (NO. 2018AAA0100400), Natural Science Foundation of Tianjin, China (NO.20JCJQJC00020), and Fundamental Research Funds for the Central Universities.

References

[1] Daniel Barath and Jiří Matas. Graph-cut ransac. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6733–6741, 2018. 6, 8

[2] Daniel Barath, Jiri Matas, and Jana Noskova. Magsac: marginalizing sample consensus. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10197–10205, 2019. 2, 6, 8

[3] Daniel Barath, Jana Noskova, Maksym Ivashechkin, and Jiri Matas. Magsac++, a fast, reliable and accurate robust estimator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1304–1312, 2020. 6

[4] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Proceedings of the European Conference on Computer Vision*, pages 404–417. Springer, 2006. 1

[5] JiaWang Bian, Wen-Yan Lin, Yasuyuki Matsushita, Sai-Kit Yeung, Tan-Dat Nguyen, and Ming-Ming Cheng. Gms: Grid-based motion statistics for fast, ultra-robust feature correspondence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4181–4190, 2017. 1, 3

[6] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6684–6692, 2017. 2

[7] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *Proceedings of the European Conference on Computer Vision*, pages 778–792, 2010. 7

[8] Luca Cavalli, Viktor Larsson, Martin Ralf Oswald, Torsten Sattler, and Marc Pollefeys. Handcrafted outlier detection revisited. In *Proceedings of the European Conference on Computer Vision*, pages 770–787, 2020. 1

[9] Hsin-Yi Chen, Yen-Yu Lin, and Bing-Yu Chen. Co-segmentation guided hough transform for robust feature matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(12):2388–2401, 2015. 1

[10] Ondrej Chum, Tomas Werner, and Jiri Matas. Two-view geometry estimation unaffected by a dominant plane. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 772–779, 2005. 6, 8

[11] Luanyuan Dai, Yizhang Liu, Jiayi Ma, Lifang Wei, Taotao Lai, Changcai Yang, and Riqing Chen. Ms2dg-net: Progressive correspondence learning via multiple sparse semantics dynamic graph. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8973–8982, 2022. 2, 3, 6

[12] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 224–236, 2018. 3, 6

[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*, 24(6):381–395, 1981. 1, 2, 6, 8

[14] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2003. 1, 2, 3

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 2

[16] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015. 4

[17] Guoli Jia and Jufeng Yang. S2-ver: Semi-supervised visual emotion recognition. In *Proceedings of the European Conference on Computer Vision*, pages 493–509, 2022. 2

[18] Xingyu Jiang, Jiayi Ma, Guobao Xiao, Zhenfeng Shao, and Xiaojie Guo. A review of multimodal image matching: Methods and applications. *Information Fusion*, 73:22–71, 2021. 2

[19] Yuhe Jin, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. *International Journal of Computer Vision*, 129(2):517–547, 2021. 1, 6, 7

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[21] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016. 2, 4, 5

[22] Wen-Yan Lin, Fan Wang, Ming-Ming Cheng, Sai-Kit Yeung, Philip HS Torr, Minh N Do, and Jiangbo Lu. Code: Coherence based decision boundaries for feature correspondence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(1):34–47, 2017. 2

[23] Wen-Yan Daniel Lin, Ming-Ming Cheng, Jiangbo Lu, Hongsheng Yang, Minh N Do, and Philip Torr. Bilateral functions for global motion modeling. In *Proceedings of the European Conference on Computer Vision*, pages 341–356, 2014. 2

[24] Xin Liu, Guobao Xiao, Riqing Chen, and Jiayi Ma. Pgfnet: Preference-guided filtering network for two-view correspondence learning. *IEEE Transactions on Image Processing*, pages 1367 – 1378, 2023. 1

[25] Yizhang Liu, Yanping Li, Luanyuan Dai, Taotao Lai, Changcai Yang, Lifang Wei, and Riqing Chen. Motion consistency-based correspondence growing for remote sensing image matching. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2021. 1

[26] Yizhang Liu, Yanping Li, Luanyuan Dai, Changcai Yang, Lifang Wei, Taotao Lai, and Riqing Chen. Robust feature matching via advanced neighborhood topology consensus. *Neurocomputing*, 421:273–284, 2021. 1

[27] Yuan Liu, Lingjie Liu, Cheng Lin, Zhen Dong, and Wenping Wang. Learnable motion coherence for correspondence pruning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3237–3246, 2021. 1, 3, 6

[28] Yizhang Liu, Brian Nlong Zhao, Shengjie Zhao, and Lin Zhang. Progressive motion coherence for remote sensing image matching. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13, 2022. 2

[29] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 1, 3, 5, 6, 8

[30] Jiayi Ma, Xingyu Jiang, Aoxiang Fan, Junjun Jiang, and Junchi Yan. Image matching from handcrafted to deep features: A survey. *International Journal of Computer Vision*, 129(1):23–79, 2021. 1, 2

[31] Jiayi Ma, Han Xu, Junjun Jiang, Xiaoguang Mei, and Xiao-Ping Zhang. Ddcgan: A dual-discriminator conditional generative adversarial network for multi-resolution image fusion. *IEEE Transactions on Image Processing*, 29:4980–4995, 2020. 1

[32] Jiayi Ma, Ji Zhao, Junjun Jiang, Huabing Zhou, and Xiaojie Guo. Locality preserving matching. *International Journal of Computer Vision*, 127(5):512–531, 2019. 3

[33] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, 2015. 1

[34] Andriy Myronenko, Xubo Song, and Miguel Carreira-Perpinan. Non-rigid point set registration: Coherent point drift. *Advances in Neural Information Processing Systems*, 19, 2006. 2

[35] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017. 2

[36] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, pages 5099–5108, 2017. 2

[37] Rahul Raguram, Ondrej Chum, Marc Pollefeys, Jiri Matas, and Jan-Michael Frahm. Usac: A universal framework for random sample consensus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):2022–2038, 2012. 2

[38] René Ranftl and Vladlen Koltun. Deep fundamental matrix estimation. In *Proceedings of the European Conference on Computer Vision*, pages 284–299, 2018. 2, 5, 6

[39] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2564–2571, 2011. 6, 7

[40] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4938–4947, 2020. 6

[41] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4104–4113, 2016. 1

[42] Weiwei Sun, Wei Jiang, Eduard Trulls, Andrea Tagliasacchi, and Kwang Moo Yi. Acne: Attentive context normalization for robust permutation-equivariant learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11286–11295, 2020. 2, 6

[43] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016. 5, 6, 7

[44] Philip HS Torr and Andrew Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156, 2000. 2

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. 5

[46] Lijuan Wang, Guoli Jia, Ning Jiang, Haiying Wu, and Jufeng Yang. Ease: Robust facial expression recognition via emotion ambiguity-sensitive cooperative networks. In *Proceedings of the ACM International Conference on Multimedia*, pages 218–227, 2022. 2

[47] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *Acm Transactions on Graphics*, 38(5):1–12, 2019. 4

[48] Guobao Xiao, Jiayi Ma, Shiping Wang, and Changwen Chen. Deterministic model fitting by local-neighbor preservation and global-residual optimization. *IEEE Transactions on Image Processing*, 29:8988–9001, 2020. 1

[49] Guobao Xiao, Hanzi Wang, Jiayi Ma, and David Suter. Segmentation by continuous latent semantic analysis for multi-structure model fitting. *International Journal of Computer Vision*, pages 1–23, 2021. 1

[50] Jianxiong Xiao, Andrew Owens, and Antonio Torralba. Sun3d: A database of big spaces reconstructed using sfm and object labels. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1625–1632, 2013. 5, 6

[51] Zhuoqian Yang, Yang Yang, Kun Yang, and Zi-Quan Wei. Non-rigid image registration with dynamic gaussian component density and space curvature preservation. *IEEE Transactions on Image Processing*, 28(5):2584–2598, 2018. 1

[52] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *Proceedings of the European Conference on Computer Vision*, pages 467–483, 2016. 1

[53] Kwang Moo Yi, Eduard Trulls, Yuki Ono, Vincent Lepetit, Mathieu Salzmann, and Pascal Fua. Learning to find good correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2666–2674, 2018. 1, 2, 3, 5, 6

[54] Jiahui Zhang, Dawei Sun, Zixin Luo, Anbang Yao, Lei Zhou, Tianwei Shen, Yurong Chen, Long Quan, and Hongen Liao. Learning two-view correspondences and geometry using order-aware network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5845–5854, 2019. 2, 5, 6, 7

[55] Chen Zhao, Zhiguo Cao, Chi Li, Xin Li, and Jiaqi Yang. Nm-net: Mining reliable neighbors for robust feature correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 215–224, 2019. 1, 3

[56] Chen Zhao, Zhiguo Cao, Jiaqi Yang, Ke Xian, and Xin Li. Image feature correspondence selection: a comparative study and a new contribution. *IEEE Transactions on Image Processing*, 29:3506–3519, 2020. 2

[57] Chen Zhao, Yixiao Ge, Feng Zhu, Rui Zhao, Hongsheng Li, and Mathieu Salzmann. Progressive correspondence pruning by consensus learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6464–6473, 2021. 1, 2, 3, 4, 5, 6, 7

[58] Sicheng Zhao, Guoli Jia, Jufeng Yang, Guiguang Ding, and Kurt Keutzer. Emotion recognition from multiple modalities: Fundamentals and methodologies. *IEEE Signal Processing Magazine*, 38(6):59–73, 2021. 2

[59] Linxin Zheng, Guobao Xiao, Ziwei Shi, Shiping Wang, and Jiayi Ma. Msa-net: Establishing reliable correspondences by multiscale attention network. *IEEE Transactions on Image Processing*, 31:4598–4608, 2022. 6

[60] Zhen Zhong, Guobao Xiao, Linxin Zheng, Yan Lu, and Jiayi Ma. T-net: Effective permutation-equivariant network for two-view correspondence learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1950–1959, 2021. 2, 6