ResMatch: Residual Attention Learning for Feature Matching

Yuxin Deng[1], **Kaining Zhang**[1], **Shihua Zhang**[1], **Yansheng Li**[2], **Jiayi Ma**[1*]

[1]Electronic Information School, Wuhan University, Wuhan 430072, China
[2]School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China
{dyx_acuo, zkn19961212, yansheng.li}@whu.edu.com, {suhzhang001, jyma2010}@gmail.com

Abstract

Attention-based graph neural networks have made great progress in feature matching. However, the literature lacks a comprehensive understanding of how the attention mechanism operates for feature matching. In this paper, we rethink cross- and self-attention from the viewpoint of traditional feature matching and filtering. To facilitate the learning of matching and filtering, we incorporate the similarity of descriptors into cross-attention and relative positions into self-attention. In this way, the attention can concentrate on learning residual matching and filtering functions with reference to the basic functions of measuring visual and spatial correlation. Moreover, we leverage descriptor similarity and relative positions to extract inter- and intra-neighbors. Then sparse attention for each point can be performed only within its neighborhoods to acquire higher computation efficiency. Extensive experiments, including feature matching, pose estimation and visual localization, confirm the superiority of the proposed method. Our codes are available at https://github.com/ACuOoOoO/ResMatch.

Introduction

Establishing point correspondences among images is a fundamental step in various computer vision tasks, such as Simultaneous Localization and Mapping (SLAM) (Fu et al. 2022), Structure-from-Motion (SfM) (Schonberger and Frahm 2016), and Multiview Stereo (MVS) (Seitz et al. 2006). To this end, features, which consist of descriptors, positions, *etc.* (Lowe 2004; Yan et al. 2022; Fan et al. 2022b), should be extracted and matched. The most common way to match features is searching nearest neighbor (NN) via the similarity of descriptors. However, due to the inherent ambiguity of descriptors, NN shows limited practicality in challenging scenes marked by significant changes in viewpoint or illumination, repetitive structures, and other complexities (Fan et al. 2022a, 2019).

Compared to NN, graph-based methods (Caetano et al. 2009; Sarlin et al. 2020) offer enhanced feature matching by considering additional properties like keypoint position, thereby enabling the discovery of more correspondences. However, crafting a graph-matching model that seamlessly optimizes across multiple information sources is challenging. Benefiting from the great potential of deep learning, SuperGlue (Sarlin et al. 2020) elegantly combines spatial information and visual appearance in an attention-based graph neural network (GNN) (Vaswani et al. 2017). Briefly, SuperGlue fuses descriptor and keypoint position in hyperspace, then uses sequential self- (intra-image) and cross- (inter-image) attention to aggregate valid information. Finally, the features augmented by the aggregated information can be precisely matched. Although SuperGlue has achieved favorable performance and led the current trend in feature matching (Lindenberger, Sarlin, and Pollefeys 2023; Sun et al. 2021), it remains unclear how features that entangle spatial and visual information interact with each other during iterative self- and cross-attention.

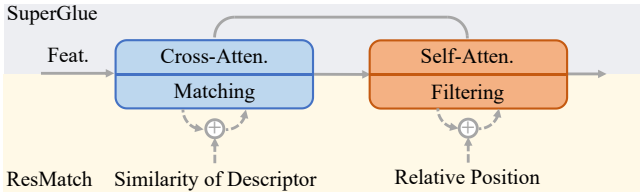

Figure 1: Interpretation of ResMatch. For the classical matching-and-filtering, the similarity of descriptors and relative positions are injected into cross- and self-attention.

We think cross-attention behaves like NN matching since it measures inter-image similarity and gathers messages from similar features. In terms of self-attention, it shares a similar formulation with the kernel-based match filter, VFC (Ma et al. 2014). Given noisy putative matches, VFC leverages kernel method to estimate the vector field, *i.e.*, optical flow between two images, which can be fitted by inliers only and used to filter outliers. The kernel measures correlation among intra-image points and the motion of a query point is calculated by the correlation-weighted sum of key motions. Similarly, self-attention measures intra-image correlation between features, which contain information of 'soft putative matches', and gathers information in a formulation of kernel method. So, self-attention might tend to predict vector field like VFC. Such evidence encourages us to promote agnostic attention-based networks from a traditional viewpoint of matching and filtering.

*Corresponding author

Intuitively, measuring visual/spatial correlation between inter-/intra-image features can be seen as the *basic function* of matching/vector-field-based filtering. In the iterative matching and filtering process, cross-attention should learn some *residual matching functions*. For example, it should additionally measure the correlation between the motions of query points and the positions of matching candidates. Moreover, self-attention should learn some *residual functions* of evaluating the reliability of key motions, consensus (a kind of correlation) between query and key motions, and even semantic correlation between query and key points.

However, limited by the dimension of space and the complexity of operations, attention-based neural networks might not be able to implement advanced matching or filtering functions, especially at the early stage. Then, original visual and spatial information turn noisy and incomplete in deep attentional propagation. In this case, the difficulty of leveraging information and learning functions is irreversibly increasing in deep layers, which limits the final performance. To tackle the problems, we can pre-compute precise basic functions on raw visual descriptors and positions. Then incorporate them into the deep layers with simple bypassing attention connections. In this way, the clean information can indicate the process in deep layers, and the limited capacity of the model can be reserved to learn the residual functions.

In this paper, we propose ResMatch, in which self- and cross-attention are reformulated as learning residual functions with reference to relative position and descriptor similarity as shown in Figure 1. Such residual attention learning brings a formidable and clean inductive bias, *i.e.*, a prior for easier optimization to attention-based feature matching networks. Moreover, we propose sparse ResMatch (sResMatch) equipped with KNN-based sparse attention to conserve computation cost. Specifically, we search k nearest inter-neighbors for each point as matching candidates according to the similarity of descriptors; k intra-neighbors for partial consensus modeling according to the relative positions. Sparse attention for a point is only computed within its neighborhood. So the computation cost of $\mathcal{O}(N^2)$ is reduced to $\mathcal{O}(kN)$ for two sets of N-point matching. Extensive experiments demonstrate that residual attention in ResMatch can yield significant improvements. The competitive performance of sResMatch supports our interpretation of attention while reducing the computation cost. Our contributions can be summarized as:

- We propose residual attention learning for feature matching, termed ResMatch. Simple bypassing injection of relative positions and the similarity of descriptors facilitates feature matching learning.
- We propose sResMatch with KNN-based linear attention. sResMatch verifies our analysis of attention for feature matching, while reducing the computation cost.
- Our models achieve remarkable performance in feature matching, pose estimation and visual localization tasks.

Related Works

Classical Feature Matching. NN is the simplest way to match features. To clean noisy matches, many post-processing methods are studied, such as mutual nearest neighbor (MNN), ratio test (RT) (Lowe 2004), filters based on local (Bian et al. 2017) or global (Ma et al. 2014; Lu et al. 2023b) consensus, and sampling-based robust solvers (Chum and Matas 2005; Barath et al. 2020). Especially, global consensus given by VFC (Ma et al. 2014) shares a similar formulation with self-attention, which encourages us to rethink self-attention as an outlier filter. Moreover, graph-based methods (Caetano et al. 2009; Torresani, Kolmogorov, and Rother 2008) seem more promising because more properties of features are utilized, but the difficulty in modeling hampers their popularity.

Deep Feature Matching. PointCN (Yi et al. 2018) takes an early effort to learn match filtering as a classification task. ConvMatch (Zhang and Ma 2023), an alternative, employs self-attention to model vector field consensus (Ma et al. 2014), which shares a similar motivation with us. Instead of filtering putative sets, SuperGlue (Sarlin et al. 2020) designs an attention-based GNN to match sparse features in a graph matching manner. To improve the practicability of SuperGlue, ParaFormer (Lu et al. 2023a) studies interactive hybrid attentions; SGMNet (Chen et al. 2021) and ClusterGNN (Shi et al. 2022) simplify $\mathcal{O}(N^2)$ cost of vanilla attention (Vaswani et al. 2017); IMP (Xue, Budvytis, and Cipolla 2023) and LightGlue (Lindenberger, Sarlin, and Pollefeys 2023) design efficient inference. By contrast, we tend to give insight on attention mechanisms. Moreover, rather than extracting sparse features before matching, LoFTR (Sun et al. 2021) finds correspondences for dense learnable features and then predicts precise locations for reliable correspondences. LoFTR has caught increasing interest in current years (Chen et al. 2022; Giang, Song, and Jo 2023; Tang et al. 2022; Xie et al. 2023), but we believe attention-based feature matching still deserves independent study since it is an essential step in LoFTR's pipeline.

Residual Learning. It has become a dispensable technique to avoid signal vanishing and facilitate optimization in deep learning (He et al. 2016). However, most residual connections are conducted on features, and only a few works learn residual attention in the bloom of Transformer (Vaswani et al. 2017; Dosovitskiy et al. 2020). Realformer (He et al. 2020) presents residual attention learning for Transformer in the natural language processing task, in which attention from previous layers is added to the current layer. EATransformer (Wang et al. 2023b) enhances the residual attention for a Transformer-based image classification network with learnable convolution. Compared with them, our residual attention is more simple and customized according to the nature of feature matching.

Efficient Attention. Vanilla attention for N-point matching takes a computation cost of $\mathcal{O}(N^2)$ (Vaswani et al. 2017). Linear Transformer (Katharopoulos et al. 2020) reduces the computation complexity to $\mathcal{O}(N)$ by decomposing softmax kernel, but might degenerate the performance on feature matching (Sun et al. 2021; Viniavskyi et al. 2022). SGMNet (Chen et al. 2021) searches several matches as seeds for down and up attentional pooling, which obtains linear complexity and satisfactory performance. Similarly, the variant of ParaFormer (Lu et al. 2023a) uses a part of

points for attentional pooling in U-Net architecture (Gao and Ji 2022). ClusterGNN (Shi et al. 2022) implements attention within k_c clusters, which leads to $\mathcal{O}(N^2/k_c)$ cost. Our sResMatch performs sparse attention for each point within a predefined neighborhood, which results in linear complexity.

Method

Attention-based Feature Matching Revisited

Given an image, feature extraction algorithms produce a set of local features, which consist of N keypoint positions $\boldsymbol{p}_i \in \mathbb{R}^2$ and corresponding visual descriptors $\boldsymbol{d}_i \in \mathbb{R}^c$, where $i \in \{1, 2, \ldots, N\}$ and c is the channel of descriptor. To match two feature sets from images A and B, SuperGlue (Sarlin et al. 2020) and its alternatives first fuse the spatial and visual information in c-dimensional space as:

$$^0\boldsymbol{x}_i = f_1(\boldsymbol{d}_i) + f_2(\boldsymbol{p}_i), \tag{1}$$

where $f(\cdot)$ is a multilayer perceptron (MLP), and the superscript 0 denotes the feature before the first attentional layer.

After that, two sets of fused features $^0\boldsymbol{X}^A$ and $^0\boldsymbol{X}^B \in \mathbb{R}^{N\times c}$ are fed into GNN, in which the basic operation, attention feed-forward $\text{Atten}(\boldsymbol{X}, \boldsymbol{Y})$ can be described as:

$$\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} = W_Q(\boldsymbol{X}), W_K(\boldsymbol{Y}), W_V(\boldsymbol{Y}), \tag{2}$$

$$S(\boldsymbol{X}, \boldsymbol{Y}) = \boldsymbol{Q}\boldsymbol{K}^T, \tag{3}$$

$$\tilde{\boldsymbol{X}} = Softmax(S(\boldsymbol{X}, \boldsymbol{Y}))\boldsymbol{V}, \tag{4}$$

$$\text{Atten}(\boldsymbol{X}, \boldsymbol{Y}) = \boldsymbol{X} + f_3(\boldsymbol{X} \| W_{\tilde{x}}(\tilde{\boldsymbol{X}})), \tag{5}$$

where $W(\cdot)$ denotes learnable linear layer, and $\|$ denotes concatenation of feature channel.

The attention operation is so-called self-attention if $\boldsymbol{X}$ and $\boldsymbol{Y}$ come from the same image and cross-attention otherwise. Eqs. (3) and (4) in cross-attention reduce to traditional NN, if $\boldsymbol{Q}$ and $\boldsymbol{K}$ carry visual information only and softmax approaches to argmax with large input magnitude. From this perspective, cross-attention can be viewed as a learnable matching process. In addition, from VFC (Ma et al. 2014) perspective, such operations on intra-image can be regarded as an interpolation procedure for vector field representation. The motion vector of a query point can be interpolated by measuring the correlation between it and key motions, then computing the weighted sum of reliable motions. The similarity of formulation and motivation between self-attention and VFC motivates us to rethink it from a perspective of vector-field-based filtering step.

After L layers hybrid attention, information of $^0\boldsymbol{X}^A$ and $^0\boldsymbol{X}^B$ is propagated into $^L\boldsymbol{X}^A$ and $^L\boldsymbol{X}^B$, respectively. Finally, the correlation matrix, $W_L(^L\boldsymbol{X}^A)W_L(^L\boldsymbol{X}^B)^T$, and a learnable dustbin are fed into Sinkhorn algorithm (Cuturi 2013) or dual softmax function (Sun et al. 2021) to acquire an assignment matrix. A cross-entropy loss can be constructed to train the network by increasing the matching probability of inliers in the assignment matrix. The architecture of feature matching is briefly illustrated in Figure 2.

Residual Attention Learning

Motivation. SuperGlue can be interpreted as an iterative feature matching-and-filtering process. In the process, cross-attention should learn the *basic function* of measuring visual

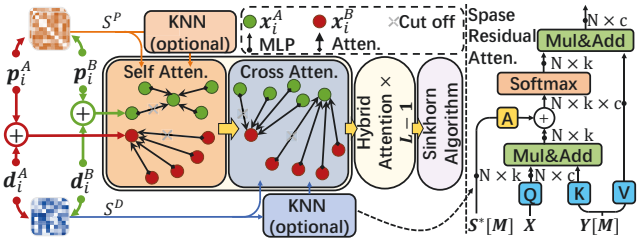

Figure 2: The architecture of ResMatch. Point-to-point relative position S^P and similarity of descriptors S^D are injected into the self- and cross-attention, respectively. Sparse residual attention propagation which is only conducted within neighborhoods, is optional.

for matching. Besides that, the advanced attention-based matching step should learn *residual functions* of matching query points and candidates with the predicted motions. Specifically, it should find the correspondence of a point in the area determined by its motion vector. Given the information of putative motion vectors, *i.e.*, raw matches embedded in the features, self-attention should estimate field consensus, *i.e.*, interpolate the approximate motion vector for each point. However, such interpolation should involve *basic functions* of measuring the spatial correlation between query and key points, then accumulating key motions like bilinear interpolation. Beside that, it should cover *residual functions* of evaluating the reliability of the key points, so the reliable ones can contribute more to the prediction. Moreover, it might additionally learn the semantic point correlation to help motion estimation since points on one object might have consistent motions.

From this perspective, cross- and self-attention can be optimized by injecting the similarity of raw descriptors and the relative positions into the attention score for two reasons: *i)* The networks might not be so versatile to learn complete functions of matching and filtering. Thus, pre-computing basic functions, *i.e.*, measuring the basic correlations, then injecting them into attention score can make model concentrate on learning residual functions with the limited capacity. *ii)* Information in raw inputs is cleaner and more complete compared to the one in intermediate features. So spatial and visual relationships computed on raw inputs and connected to deep attention blocks might be more reliable to initialize and indicate the matching-and-filtering process.

Residual Cross-attention. Since the matching step needs measure inter-image visual similarity, we achieve this by:

$$S^D(\boldsymbol{X}, \boldsymbol{Y}) = f_4(\boldsymbol{D}^X)f_4(\boldsymbol{D}^Y)^T, \tag{6}$$

where $\boldsymbol{D}^X \in \mathbb{R}^{N\times c}$ denotes the descriptor set corresponding to $\boldsymbol{X}$. Then we add S^D into Eq. (3) after affine modulation and activation:

$$S^{D'}(\boldsymbol{X}, \boldsymbol{Y}) = \text{LReLU}(\lambda^D S^D(\boldsymbol{X}, \boldsymbol{Y}) + \beta^D), \tag{7}$$

$$S^{ResD}(\boldsymbol{X}, \boldsymbol{Y}) = S(\boldsymbol{X}, \boldsymbol{Y}) + S^{D'}(\boldsymbol{X}, \boldsymbol{Y}), \tag{8}$$

where LReLU denotes leaky rectified linear unit, λ and β are learnable affine parameters which are unique for different

layers. S^D is broadcast to multi-heads with unshared parameters. In this way, $S(\boldsymbol{X}, \boldsymbol{Y})$ in cross attention can be seen as a residual function with reference to $S^D(\boldsymbol{X}, \boldsymbol{Y})$ and facilitate the learning of complete matching function. As shown in Figure 3, the similarity of mismatches decreases fast, which indicates that the cross-attention acts as NN.

Residual Self-attention. Vector field consensus modeling in self-attention needs clean spatial information and measure intra-image spatial correlation (Ma et al. 2014). So we employ a nonlinear neural network to pre-compute the spatial relationship, *i.e.*, relative positions, as

$$S^P(\boldsymbol{X}, \boldsymbol{X}) = f_5(\boldsymbol{P}^X) f_5(\boldsymbol{P}^X)^T, \tag{9}$$

where $\boldsymbol{P}^X$ denotes the keypoint positions corresponding to $\boldsymbol{X}$. And then, we inject the relative positions into self-attention like residual cross-attention:

$$S^{P'}(\boldsymbol{X}, \boldsymbol{X}) = \text{LReLU}(\lambda^P S^P(\boldsymbol{X}, \boldsymbol{X}) + \beta^P), \tag{10}$$

$$S^{ResP}(\boldsymbol{X}, \boldsymbol{X}) = S(\boldsymbol{X}, \boldsymbol{X}) + S^{P'}(\boldsymbol{X}, \boldsymbol{X}). \tag{11}$$

As shown in Figure 3, self-attention is assembled in local areas due to strong spatial correlation. Importantly, the topological structures of self-attention in two views are similar, in which several correspondences with dark red links can be easily found. It means these correspondences are reliable, so the correlation, *i.e.*, the kernel value, between them and the query point is high. Consequently, their motion would contribute more to the prediction. Generally, the visualization in Figure 3 validates our analysis of attention mechanisms.

Bypassing Attention Adjustment. As shown in Figure 3, the similarity of descriptors might not be indicative enough to guide matching in deep layers. And relative positions, which only involve point-to-point spatial relationship, does not fulfill the demand of field consensus modeling. Therefore, we adjust these bypassing attentions, *i.e.*, S^D and S^P, at a middle layer (4th):

$${}^4S^D(\boldsymbol{X}, \boldsymbol{Y}) =^0 S^D(\boldsymbol{X}, \boldsymbol{Y}) + f_6({}^4\boldsymbol{X}) f_6({}^4\boldsymbol{Y})^T, \tag{12}$$

$${}^4S^P(\boldsymbol{X}, \boldsymbol{X}) =^0 S^P(\boldsymbol{X}, \boldsymbol{X}) + f_7({}^4\boldsymbol{X}) f_7({}^4\boldsymbol{X})^T, \tag{13}$$

where f_6 and f_7 are task-specific information decoders. ${}^0S$ denotes the bypassing attention computed before the first attention layer with Eqs. (6) and (9). ${}^4S$ replaces ${}^0S$ in residual attention learning, *i.e.*, Eqs. (7) and (11) since the 4th layer. Both ${}^0S$ and ${}^4S$ are computed once.

Sparse Residual Attention

Motivation. It is expensive to reconstruct N^2 putative matches and gather all messages of them in the iterative matching steps. It is reasonable to select a subset of candidates for matching according to the similarity of descriptors S^D. Moreover, interpolation often favors to gather values in neighborhoods. Motion vectors predicted by VFC (Ma et al. 2014) also highly depend on the value of the neighbors due to the nature of local affinity (Ma et al. 2019). So we can utilize only a few neighbors as keys for predicting the motion vector of a query point. Considering *i)* the attention propagation works as an iterative matching-and-filtering process, *ii)* bypassing S^D and bypassing S^P are believed to domain the attention in our ResMatch, we can conduct KNN on S^D and S^P to sparsify residual attention as discussed above.

Figure 3: Illustration of attention score. Red full lines denote the top 16 attention scores for the center query point, and green dash lines denote the 16 neighbors according to bypassing S^D and S^P. Query points in images are correspondences in two views.

KNN-based Sparse Attention. We find $k/2$ matching candidates for each point according to S^D. For filtering, we also mine k nearest neighbors (KNN) for local consensus modeling according to S^P. So KNN-based sparse residual attention for a query feature $\boldsymbol{x}_i$ can be formulated briefly as:

$$\boldsymbol{M}_i = \text{KNN}(\boldsymbol{x}_i, S^*), S^* \in \{S^P, S^D\}, \tag{14}$$

$$S(\boldsymbol{x}_i, \boldsymbol{Y}[\boldsymbol{M}_i]) = \boldsymbol{x}_i \boldsymbol{K}[\boldsymbol{M}_i]^T + S^*[i, \boldsymbol{M}_i], \tag{15}$$

$$\tilde{\boldsymbol{x}}_i = Softmax(S)\boldsymbol{V}[\boldsymbol{M}_i], \tag{16}$$

where $\boldsymbol{M}_i \in \mathbb{N}^k$ stores the indices of k nearest neighbors of $\boldsymbol{x}_i$ according to S^P or S^D, and $[\cdot]$ denotes the indexing operation. Bypassing attention $S^*[i, \boldsymbol{M}_i]$ is crucial for differential KNN and overall learning. KNN is conducted both at the initial step and after adjustment.

Intuitively, more reliable neighbors should be involved for local consensus modeling in sparse self-attention, while the number of matching candidates in cross-attention can be relatively small for distinguishing descriptors. So, at the initial step, we mine k neighbors for self-attention and $k/2$ for cross-attention; After bypassing attention adjustment, only $k/4$ neighbors are mined for cross-attention. k is empirically set to a constant 64 for balancing performance and efficiency. In this way, we obtain computation cost linear to the number of query points.

Implementation Details

The networks equipped with full and sparse residual attention learning are termed ResMatch and sResMatch, respectively. Our networks consist of sequential 9 blocks of 4-head hybrid attention, of which feature dimension is consistent with the input descriptor. Following SGMNet (Chen et al. 2021), we train the network on the GL3D (Shen et al. 2018). In the training, 1k features are extracted for each image. 10 iterations of Sinkhorn algorithm are performed to obtain the assignment matrix. And cross-entropy loss same as SuperGlue (Sarlin et al. 2020) is conducted on the final matching

Figure 4: Samples of RootSIFT matching results.The green/red lines link the inliers/outliers.

probability. The networks are trained in 450000 iterations with batch size of 16.

Experiments

We evaluate our methods for two-view image matching on three datasets, including YFCC100M (Thomee et al. 2016), ScanNet (Dai et al. 2017), and FM-Bench (Bian et al. 2019). We use Aachen Day-Night V1.1 (Sattler et al. 2018) to further verify the applicability of our methods in visual localization task. In all datasets, our methods are employed to match three kinds of features including handcrafted Root-SIFT (Lowe 2004; Arandjelović and Zisserman 2012), DOG+HN (Lowe 2004; Mishchuk et al. 2017), and SuperPoint (DeTone, Malisiewicz, and Rabinovich 2018). The performance is compared to NN with RT (Lowe 2004), learnable filter ConvMatch (Zhang and Ma 2023), and feature matching GNNs including SuperGlue (Sarlin et al. 2020), SGMNet (Chen et al. 2021), ParaFormer (Lu et al. 2023a), and LightGlue (9 layers) (Lindenberger, Sarlin, and Pollefeys 2023). All GNNs are trained on GL3D. Some samples of RootSIFT matching are shown in Figure 4.

Image Matching

YFCC100M contains 100 million outdoor photos collected from the Internet, along with 72 3D models reconstructed in the SfM pipeline (Thomee et al. 2016; Heinly et al. 2015). Following SGMNet (Chen et al. 2021), we select 4000 pairs of images from 4 models for testing. Up to 2k features are extracted for each image by three feature extraction methods and matched by different matchers. Finally, we estimate relative camera pose with predicted matches and the robust solver, RANSAC (Fischler and Bolles 1981). Three metrics are reported in Table 1: *i)* Approximate AUC at different thresholds of estimated rotation and translation error, *ii)* the ratio of the number of correct matches to extracted keypoints, known as matching score, and *iii)* the inlier rate of predicted matches, known as mean precision.

As shown in Table 1, our residual attention learning brings certain improvements for three kinds of feature matching on most metrics. Especially, we boost matching precision of RootSIFT+SuperGlue and DOG+HN+SuperGlue by over 8% and matching score by about 2%, which imposes a positive effect on pose estimation accuracy. It is worth mentioning that KNN-based sResMatch keeps the admirable performance while saving computation costs in theory.

Feature (#)	Matcher	AUC @5°	AUC @10°	AUC @20°	M.S.	Prec.
RootSIFT (2k)	NN	48.25	58.16	68.13	4.44	56.38
	ConvMatch*	59.23	68.54	77.56	8.03	67.91
	SuperGlue	61.45	71.23	80.62	17.37	74.91
	SGMNet	62.72	72.52	81.48	17.08	**86.08**
	ParaFormer	61.80	71.70	81.18	17.09	76.29
	ResMatch	**63.92**	73.33	82.15	**19.34**	83.54
	sResMatch	63.51	**73.52**	**82.26**	18.65	81.43
DOG+HN (2k)	NN	53.05	63.11	73.37	8.44	55.96
	ConvMatch*	59.55	69.46	75.22	13.53	60.66
	SuperGlue	61.12	70.88	80.56	16.92	75.37
	SGMNet	62.98	72.81	82.02	17.87	83.23
	ParaFormer	62.00	71.86	81.21	18.01	78.40
	ResMatch	**64.70**	**74.43**	**83.20**	**18.76**	**86.27**
	sResMatch	64.35	74.06	82.90	18.71	81.77
SuperPoint (2k)	NN	33.40	42.66	53.40	7.83	36.14
	ConvMatch*	50.28	61.25	71.78	12.58	68.34
	SuperGlue	60.45	70.71	80.00	19.47	80.22
	SuperGlue*	67.10	76.18	84.37	21.58	88.64
	SGMNet	61.22	71.02	80.45	22.36	85.44
	ParaFormer	61.75	72.03	81.23	22.31	81.28
	LightGlue	62.90	73.01	81.48	22.78	85.60
	ResMatch	**63.03**	**73.04**	**81.94**	**23.06**	**85.99**
	sResMatch	62.40	72.62	81.65	21.82	83.05

Table 1: Results on YFCC100M, where AUC denotes the accuracy of the estimated poses, M.S. denotes matching score, and Prec. denotes precision. Bold indicates the best. * denotes the official model trained in other protocols.

ScanNet is a comprehensive indoor dataset with ground-truth poses and depth (Dai et al. 2017). The indoor images in ScanNet lack textures and exhibit distinct variance in depth, which obstructs feature matching. Following SuperGlue, we select 1500 wide-baseline pairs of images for the test after overlap validation. Images in test are resized to a fixed size of 640×480. The same evaluation pipeline and metrics in YFCC100M are employed in ScanNet. Furthermore, we compare ResMatch to SuperGlue on advanced features: DISK (Tyszkiewicz, Fua, and Trulls 2020), ALIKED (Zhao et al. 2023), and AWDesc (Wang et al. 2023a).

Our ResMatch outperforms counterparts with distinct margins for pose estimation as shown in Table 2. Especially, there is an about 3% average improvement of AUC in different matching protocols, which demonstrates that the solid inductive bias provided by residual attention learning benefits feature matching in challenging scenes. Moreover, the remarkable performance of sResMatch reveals sparsifying self- and cross-attention according to relative positions and the similarity of descriptors is effective, which confirms our interpretation of attention-based feature matching networks.

FM-Bench comprises four subsets (CPC, T&T, TUM and KITTI) covering driving, indoor SLAM and wide-baseline scenarios (Bian et al. 2019). We choose CPC, T&T, TUM for test. We only perform 10 iterations for Sinkhorn algorithm, which saves half test time. Fundamental matrices are estimated on predicted matches and the estimation is considered

Feature (#)	Matcher	AUC @5°	AUC @10°	AUC @20°	M.S.	Prec.
RootSIFT (2k)	SuperGlue	30.08	41.30	53.00	9.08	38.44
	SGMNet	30.66	41.13	52.80	8.79	45.54
	ParaFormer	32.62	43.32	54.21	9.09	38.38
	ResMatch	**35.40**	**46.53**	**57.97**	**10.19**	44.94
	sResMatch	35.20	46.13	57.85	9.51	**46.09**
DOG+HN (2k)	SuperGlue	34.06	44.60	56.03	8.90	40.36
	SGMNet	34.53	45.67	57.68	9.33	45.11
	ParaFormer	33.06	44.73	56.28	9.23	40.70
	ResMatch	**38.10**	**49.47**	**60.80**	**10.30**	**45.74**
	sResMatch	37.93	48.80	60.31	9.73	42.68
SuperPoint (1k)	SuperGlue	34.18	44.35	54.89	16.25	46.16
	SuperGlue*	37.93	49.70	62.34	18.50	47.32
	SGMNet	35.40	44.83	55.80	**16.86**	47.83
	ParaFormer	34.96	45.66	56.85	16.62	46.64
	LightGlue	37.06	46.86	57.83	16.40	47.80
	ResMatch	**37.87**	47.27	**58.40**	16.64	**48.46**
	sResMatch	36.46	**47.50**	58.12	16.79	46.83
DISK (2k)	SuperGlue	28.00	38.83	50.12	13.43	43.62
	ResMatch	32.00	42.03	53.80	14.22	48.02
ALIKED (2k)	SuperGlue	34.06	43.90	55.56	13.46	46.42
	ResMatch	37.07	47.77	59.08	14.51	48.80
AWDesc (2k)	SuperGlue	39.33	50.20	61.48	12.09	42.38
	ResMatch	42.86	54.26	65.37	11.96	48.19

Table 2: Wide-baseline indoor pose estimation in ScanNet. Approximate AUC, matching score (M.S.) and matching precision (Prec.) are reported.

correct if its normalized symmetric epipolar distance (Zhang 1998) is lower than 0.05. In Table 3, we report recall of fundamental matrix estimation and the mean number of correspondences (Corr.) after RANSAC processing.

As we can see, our methods show advantages on most metrics. However, SGMNet seems to perform better on TUM, while our sResMatch yields a degeneration. This might be caused by low resolution, poor quality and lack of textures in TUM indoor images. Forcing feature extraction methods to extract dense and unreliable features troubles the subsequent matching. In this case, our sResMatch might be struck in unreliable local consensus between two subsets, which finally leads to relatively weak performance.

Visual Localization

Visual localization is a common task to verify the applicability of feature matching algorithms. Thus, we integrate several feature matching algorithms into the state-of-the-art visual localization pipeline Hierarchical Localization (HLoc) (Sarlin et al. 2019) and run the pipeline on the Aachen Day-Night V1.1 dataset, which consists of 6697 reference images and 1015 (824 daytime, 191 nighttime) query images. For each image, we extract up to 8k RootSIFT, 4k DOG+HN and SuperPoint associated with NetVLAD (Arandjelovic et al. 2016) global feature. To save test time, we only perform 50 iterations of Sinkhorn algorithm for matching. The localization accuracy under different thresholds is reported in Table 4. As we can see, our methods achieve high scores on all metrics, which confirms the applicability of residual attention learning in the downstream tasks. Moreover, our proposals demonstrate scalability to the number of features, as evidenced by the favorable performance on 8k RootSIFT and 4K DOG+HN, which is several times larger than the training size of 1k.

Feature (#)	Matcher	%Reall (#Corr.) CPC	%Reall (#Corr.) T&T	%Reall (#Corr.) TUM
RootSIFT (4k)	SuperGlue	57.3 (240)	86.1 (418)	62.5 (797)
	SGMNet	61.2 (**289**)	84.3 (**468**)	**67.2** (**945**)
	ParaFormer	58.4 (243)	86.3 (428)	61.6 (838)
	ResMatch	62.4 (273)	**88.0** (464)	65.4 (910)
	sResMatch	**63.7** (261)	86.5 (445)	62.7 (845)
DOG+HN (4k)	SuperGlue	58.3 (236)	87.2 (415)	63.5 (794)
	SGMNet	60.9 (268)	86.3 (433)	64.8 (895)
	ParaFormer	60.2 (250)	87.3 (433)	63.9 (840)
	ResMatch	**65.2** (**286**)	**89.1** (**479**)	**67.2** (**956**)
	sResMatch	64.1 (269)	87.8 (451)	62.4 (861)
SuperPoint (4k)	SuperGlue	68.8 (346)	95.0 (482)	59.1 (803)
	SGMNet	69.0 (**380**)	93.7 (**511**)	62.6 (**923**)
	ParaFormer	70.0 (351)	94.6 (483)	58.3 (798)
	ResMatch	72.3 (366)	95.1 (505)	**64.2** (848)
	sResMatch	**75.6** (342)	**95.6** (478)	58.6 (728)

Table 3: Fundamental matrix estimation on FM-Bench. %Recall denotes the precision of fundamental matrix estimation, and #Corr. denotes the number of inliers.

Feature (#)	Matcher	(0.25m,2°) / (0.5m,5°) / (1.0m,10°) Day	(0.25m,2°) / (0.5m,5°) / (1.0m,10°) Night
RootSIFT (8k)	SuperGlue	82.8/ 91.1/ 96.8	51.8/ 62.3/ 84.3
	SGMNet	82.8/ 90.5/ 97.2	56.0/ 72.3/ 89.5
	ParaFormer	82.9/ 90.8/ 96.5	52.9/ 67.0/ 84.8
	ResMatch	84.8/ 91.4/ 97.3	56.5/ 74.9/ 92.1
	sResMatch	86.5/ 93.4/ 97.9	57.6/ 72.3/ 90.6
DOG+HN (4k)	SuperGlue	84.6/ 92.0/ 97.1	49.2/ 63.9/ 80.6
	SGMNet	84.8/ 93.1/ 97.3	65.4/ 81.2/ 90.6
	ParaFormer	82.9/ 91.3/ 97.0	50.8/ 64.9/ 83.8
	ResMatch	85.6/ 92.8/ 98.5	64.4/ 80.1/ 93.7
	sResMatch	85.0/ 93.1/ 97.5	65.4/ 82.2/ 92.7
SuperPoint (4k)	SuperGlue	87.9/ 93.4/ 97.8	70.2/ 89.5/ 97.4
	SuperGlue*	89.4/ 96.5/ 99.4	75.4/ 91.1/ 99.5
	SGMNet	87.0/ 94.3/ 98.4	69.6/ 87.4/ 97.9
	ParaFormer	87.1/ 93.7/ 97.9	69.1/ 88.0/ 97.4
	ResMatch	87.5/ 94.3/ 98.5	68.6/ 90.6/ 99.0
	sResMatch	88.6/ 95.0/ 98.9	71.2/ 89.5/ 98.4

Table 4: Visual localization on Aachen V1.1. */*/* denotes the visual localization accuracy under thresholds of (0.25m, 2°)/(0.5m, 5°)/(1.0m, 10°).

Discussion

Ablation Study

In this paper, we propose residual self-, cross-attention and their corresponding sparse versions for feature matching. To investigate the effectiveness of each proposal, we conduct ablation study with 2k RootSIFT on YFCC100M (Thomee et al. 2016; Heinly et al. 2015) and ScanNet (Dai et al. 2017). Results of ablation study are reported in Table 5.

As we can see, ResMatch without residual cross-attention produces larger degeneration of performance than ResMatch without residual self-attention. The reason might be that the information of visual appearance in c-D raw descriptors is compressed by f_1 to make room for 2-D positional infor-

AUC@20° (%)	YFCC100M	ScanNet
SuperGlue	80.62	53.00
ResMatch	82.15	57.97
w/o ResSelfAtten	82.00	57.16
w/o ResCrossAtten	81.35	55.47
w/o Adjustment	81.42	56.31
sResMatch	82.26	57.85
w/ FullSelfAtten	82.33	58.12
w/ FullCrossAtten	81.92	58.04
k=32	80.78	56.34

Table 5: Ablation study on YFCC100M and ScanNet. AUC under a threshold of 20° is reported.

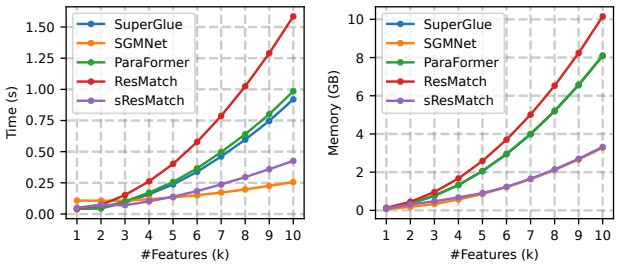

Figure 5: Computation efficiency of feature matching networks on an NVIDIA RTX3090 GPU. The memory consumption of SuperGlue is close to ParaFormer.

Layers	1	2	3	6	9
SuperGlue	39.93	44.35	47.63	52.25	53.00
SGMNet	40.46	41.95	43.85	52.46	52.80
ResMatch	46.57	50.65	53.35	56.30	57.97
sResMatch	46.86	50.50	52.86	56.18	57.85

Table 6: The impact of numbers of layers for RootSIFT matching. AUC@20° on ScanNet is reported.

mation in the c-D fused hyperspace. The information loss confuses the matching function in ResMatch without residual cross-attention and SuperGlue (Sarlin et al. 2020). Conversely, 2-D positional information mapped into c-D intermediate features has been complete and clean enough for filtering of ResMatch without residual self-attention. However, residual self-attention still improves the SuperGlue with certain margins, which demonstrates the significance of bypassing injection of relative position. Moreover, bypassing attention adjustment would facilitate the learning in deep layers as proved by the ablation study.

For sResMatch, the ablation study suggests our sparsification principle ($k = 64$) for attention yield indistinct changes in performance. We even can find some improvements brought by sparse attention in Tables 3 and 4, in which large numbers of features are extracted. The reason might be that the information of all points is aggregated into a limited feature space. And the mixed information is too ambiguous to model precise vector field. By contrast, the sparsification tightens the solution space of modeling and obtains more precise model after limited iterations. However, too sparse attention with $k = 32$ leads to significant drop because matching candidates are too few to cover the correspondence and the neighborhoods in self-attention are too small to cover enough inliers for local consensus modeling.

Computation Efficiency

We compare the computation efficiency of our networks to SuperGlue (Sarlin et al. 2020), SGMNet (Sarlin et al. 2020), ParaFormer (Lu et al. 2023a). The computation cost versus the numbers of 128-D features, are drawn in Figure 5. In theory, residual attention learning should take little additional time over SuperGlue. However, the cost of ResMatch is out of our expectation in practice. Similarly, some operations of sResMatch are hard to optimize in programming. For example, KNN in Equation (14) takes 20% time consumption for 8k features matching, and indexing operations in Equations (15) and (16) take 23%. For memory cost, our sResMatch with $k = 64$ is competitive with SGMNet, which is one of the major reasons why k is set to 64.

Impact of The Number of Layers

We think residual attention learning can facilitate the matching-and-filtering process, especially at the initial stage. If our analysis is solid, our residual attention might give more improvements with fewer layers. As we can see in Table 6, compared to SuperGlue, our methods yield over 6% improvements in networks with 2 layers, while smaller than 5% with more than 6 layers. These evidences confirm that our residual attention learning can better initialize the matching-and-filtering step. Moreover, benefiting from the clean basic functions injected to the deep layers, our networks seem to obtain better results by stacking more layers, while the performance of SuperGlue and SGMNet seems saturated with more than 6 layers. It is worth mentioning that the computation cost of our sResMatch can be further optimized by point pruning and early stop, which is studied by recent works IMP (Xue, Budvytis, and Cipolla 2023) and LightGlue (Lindenberger, Sarlin, and Pollefeys 2023).

Conclusion

In this paper, we rethink self- and cross-attention in feature matching networks from a viewpoint of feature matching and filtering. For the viewpoint, self- and cross-attention are reformulated as learning residual functions with reference to the basic functions of measuring spatial and visual correlation, then added by relative position and the similarity of descriptors to facilitate the learning, respectively. ResMatch equipped with the proposed residual attention, obtains promising performance in extensive experiments. Furthermore, we perform sparse self- and cross-attention of each point only with its intra- and inter-neighborhoods, which are mined according to the two kinds of references. sResMatch with sparse residual attention not only reduces the computation cost, but also verifies the significance of residual attention learning with competitive performance.

In summary, we bridge the gap between the interpretable matching-and-filtering pipeline and agnostic attention-based feature matching networks empirically. Comprehensive experiments confirm the validity of our analysis and proposals.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (62276192).

References

Arandjelovic, R.; Gronat, P.; Torii, A.; Pajdla, T.; and Sivic, J. 2016. NetVLAD: CNN architecture for weakly supervised place recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 5297–5307.

Arandjelović, R.; and Zisserman, A. 2012. Three things everyone should know to improve object retrieval. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2911–2918.

Barath, D.; Noskova, J.; Ivashechkin, M.; and Matas, J. 2020. MAGSAC++, a fast, reliable and accurate robust estimator. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1304–1312.

Bian, J.; Lin, W.-Y.; Matsushita, Y.; Yeung, S.-K.; Nguyen, T.-D.; and Cheng, M.-M. 2017. Gms: Grid-based motion statistics for fast, ultra-robust feature correspondence. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 4181–4190.

Bian, J.-W.; Wu, Y.-H.; Zhao, J.; Liu, Y.; Zhang, L.; Cheng, M.-M.; and Reid, I. 2019. An Evaluation of Feature Matchers for Fundamental Matrix Estimation. In *Proc. Brit. Mach. Vis. Conf.*

Caetano, T. S.; McAuley, J. J.; Cheng, L.; Le, Q. V.; and Smola, A. J. 2009. Learning graph matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(6): 1048–1058.

Chen, H.; Luo, Z.; Zhang, J.; Zhou, L.; Bai, X.; Hu, Z.; Tai, C.-L.; and Quan, L. 2021. Learning to match features with seeded graph matching network. In *Proc. IEEE Int. Conf. Comput. Vis.*, 6301–6310.

Chen, H.; Luo, Z.; Zhou, L.; Tian, Y.; Zhen, M.; Fang, T.; McKinnon, D.; Tsin, Y.; and Quan, L. 2022. Aspanformer: Detector-free image matching with adaptive span transformer. In *Proc. Europ. Conf. Comput. Vis.*, 20–36.

Chum, O.; and Matas, J. 2005. Matching with PROSAC-progressive sample consensus. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 220–226.

Cuturi, M. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Adv. Neural Inf. Process. Syst.*, 26.

Dai, A.; Chang, A. X.; Savva, M.; Halber, M.; Funkhouser, T.; and Nießner, M. 2017. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 5828–5839.

DeTone, D.; Malisiewicz, T.; and Rabinovich, A. 2018. Superpoint: Self-supervised interest point detection and description. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 224–236.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. Int. Conf. Learn. Represent.*

Fan, B.; Kong, Q.; Wang, X.; Wang, Z.; Xiang, S.; Pan, C.; and Fua, P. 2019. A performance evaluation of local features for image-based 3D reconstruction. *IEEE Trans. Image Process.*, 28(10): 4774–4789.

Fan, B.; Yang, Y.; Feng, W.; Wu, F.; Lu, J.; and Liu, H. 2022a. Seeing through darkness: Visual localization at night via weakly supervised learning of domain invariant features. *IEEE Trans. Multimedia.*

Fan, B.; Zhou, J.; Feng, W.; Pu, H.; Yang, Y.; Kong, Q.; Wu, F.; and Liu, H. 2022b. Learning semantic-aware local features for long term visual localization. *IEEE Trans. Image Process.*, 31: 4842–4855.

Fischler, M. A.; and Bolles, R. C. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6): 381–395.

Fu, Q.; Yu, H.; Wang, X.; Yang, Z.; He, Y.; Zhang, H.; and Mian, A. 2022. Fast ORB-SLAM without Keypoint Descriptors. *IEEE Trans. Image Process.*, 31: 1433–1446.

Gao, H.; and Ji, S. 2022. Graph U-Nets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(9): 4948–4960.

Giang, K. T.; Song, S.; and Jo, S. 2023. TopicFM: Robust and Interpretable Feature Matching with Topic-assisted. In *Proc. AAAI Conf. Artif. Intell.*

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 770–778.

He, R.; Ravula, A.; Kanagal, B.; and Ainslie, J. 2020. Realformer: Transformer likes residual attention. *arXiv preprint arXiv:2012.11747*.

Heinly, J.; Schonberger, J. L.; Dunn, E.; and Frahm, J.-M. 2015. Reconstructing the world* in six days*(as captured by the yahoo 100 million image dataset). In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 3287–3295.

Katharopoulos, A.; Vyas, A.; Pappas, N.; and Fleuret, F. 2020. Transformers are rnns: Fast autoregressive transformers with linear attention. In *Proc. Int. Conf. Mach. Learn.*, 5156–5165.

Lindenberger, P.; Sarlin, P.-E.; and Pollefeys, M. 2023. LightGlue: Local Feature Matching at Light Speed. In *Proc. IEEE Int. Conf. Comput. Vis.*

Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.*, 60(2): 91–110.

Lu, X.; Yan, Y.; Kang, B.; and Du, S. 2023a. ParaFormer: Parallel Attention Transformer for Efficient Feature Matching. In *Proc. AAAI Conf. Artif. Intell.*

Lu, Y.; Ma, J.; Fang, L.; Tian, X.; and Jiang, J. 2023b. Robust and Scalable Gaussian Process Regression and Its Applications. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 21950–21959.

Ma, J.; Zhao, J.; Jiang, J.; Zhou, H.; and Guo, X. 2019. Locality preserving matching. *Int. J. Comput. Vis.*, 127: 512–531.

Ma, J.; Zhao, J.; Tian, J.; Yuille, A. L.; and Tu, Z. 2014. Robust point matching via vector field consensus. *IEEE Trans. Image Process.*, 23(4): 1706–1721.

Mishchuk, A.; Mishkin, D.; Radenovic, F.; and Matas, J. 2017. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Adv. Neural Inf. Process. Syst.*, 4829–4840.

Sarlin, P.-E.; Cadena, C.; Siegwart, R.; and Dymczyk, M. 2019. From Coarse to Fine: Robust Hierarchical Localization at Large Scale. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 12716–12725.

Sarlin, P.-E.; DeTone, D.; Malisiewicz, T.; and Rabinovich, A. 2020. Superglue: Learning feature matching with graph neural networks. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 4938–4947.

Sattler, T.; Maddern, W.; Toft, C.; Torii, A.; Hammarstrand, L.; Stenborg, E.; Safari, D.; Okutomi, M.; Pollefeys, M.; Sivic, J.; et al. 2018. Benchmarking 6dof outdoor visual localization in changing conditions. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 8601–8610.

Schonberger, J. L.; and Frahm, J.-M. 2016. Structure-from-motion revisited. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 4104–4113.

Seitz, S. M.; Curless, B.; Diebel, J.; Scharstein, D.; and Szeliski, R. 2006. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 519–528.

Shen, T.; Luo, Z.; Zhou, L.; Zhang, R.; Zhu, S.; Fang, T.; and Quan, L. 2018. Matchable Image Retrieval by Learning from Surface Reconstruction. In *Proc. Asian Conf. Comput. Vis.*

Shi, Y.; Cai, J.-X.; Shavit, Y.; Mu, T.-J.; Feng, W.; and Zhang, K. 2022. Clustergnn: Cluster-based coarse-to-fine graph neural network for efficient feature matching. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 12517–12526.

Sun, J.; Shen, Z.; Wang, Y.; Bao, H.; and Zhou, X. 2021. LoFTR: Detector-free local feature matching with transformers. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 8922–8931.

Tang, S.; Zhang, J.; Zhu, S.; and Tan, P. 2022. Quadtree Attention for Vision Transformers. In *Proc. Int. Conf. Learn. Represent.*

Thomee, B.; Shamma, D. A.; Friedland, G.; Elizalde, B.; Ni, K.; Poland, D.; Borth, D.; and Li, L.-J. 2016. YFCC100M: The new data in multimedia research. *Commun. ACM*, 59(2): 64–73.

Torresani, L.; Kolmogorov, V.; and Rother, C. 2008. Feature correspondence via graph matching: Models and global optimization. In *Proc. Europ. Conf. Comput. Vis.*, 596–609.

Tyszkiewicz, M.; Fua, P.; and Trulls, E. 2020. DISK: Learning local features with policy gradient. In *Adv. Neural Inf. Process. Syst.*, 14254–14265.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Adv. Neural Inf. Process. Syst.*, volume 30.

Viniavskyi, O.; Dobko, M.; Mishkin, D.; and Dobosevych, O. 2022. Openglue: Open source graph neural net based pipeline for image matching. *arXiv preprint arXiv:2204.08870*.

Wang, C.; Xu, R.; Lv, K.; Xu, S.; Meng, W.; Zhang, Y.; Fan, B.; and Zhang, X. 2023a. Attention Weighted Local Descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(9): 10632–10649.

Wang, Y.; Yang, Y.; Li, Z.; Bai, J.; Zhang, M.; Li, X.; Yu, J.; Zhang, C.; Huang, G.; and Tong, Y. 2023b. Convolution-enhanced Evolving Attention Networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(7): 8176–8192.

Xie, T.; Dai, K.; Wang, K.; Li, R.; and Zhao, L. 2023. DeepMatcher: A Deep Transformer-based Network for Robust and Accurate Local Feature Matching. *arXiv preprint arXiv:2301.02993*.

Xue, F.; Budvytis, I.; and Cipolla, R. 2023. IMP: Iterative Matching and Pose Estimation with Adaptive Pooling. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 21317–21326.

Yan, P.; Tan, Y.; Xiong, S.; Tai, Y.; and Li, Y. 2022. Learning Soft Estimator of Keypoint Scale and Orientation with Probabilistic Covariant Loss. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 19406–19415.

Yi, K. M.; Trulls, E.; Ono, Y.; Lepetit, V.; Salzmann, M.; and Fua, P. 2018. Learning to find good correspondences. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2666–2674.

Zhang, S.; and Ma, J. 2023. ConvMatch: Rethinking Network Design for Two-View Correspondence Learning. In *Proc. AAAI Conf. Artif. Intell.*, 3472–3479.

Zhang, Z. 1998. Determining the epipolar geometry and its uncertainty: A review. *Int. J. Comput. Vis.*, 27: 161–195.

Zhao, X.; Wu, X.; Chen, W.; Chen, P. C.; Xu, Q.; and Li, Z. 2023. ALIKED: A Lighter Keypoint and Descriptor Extraction Network via Deformable Transformation. *IEEE Trans. Instrum. Meas.*, 72: 5014016.