MatchFormer: Interleaving Attention in Transformers for Feature Matching

Qing Wang*, Jiaming Zhang*, Kailun Yang†,
Kunyu Peng, and Rainer Stiefelhagen

Karlsruhe Institute of Technology, Germany
https://github.com/jamycheung/MatchFormer

Abstract. Local feature matching is a computationally intensive task at the subpixel level. While *detector-based* methods coupled with feature descriptors struggle in low-texture scenes, CNN-based methods with a sequential *extract-to-match* pipeline, fail to make use of the matching capacity of the encoder and tend to overburden the decoder for matching. In contrast, we propose a novel hierarchical *extract-and-match* transformer, termed as *MatchFormer*. Inside each stage of the hierarchical encoder, we interleave self-attention for feature extraction and cross-attention for feature matching, yielding a human-intuitive *extract-and-match* scheme. Such a match-aware encoder releases the overloaded decoder and makes the model highly efficient. Further, combining self- and cross-attention on multi-scale features in a hierarchical architecture improves matching robustness, particularly in low-texture indoor scenes or with less outdoor training data. Thanks to such a strategy, MatchFormer is a multi-win solution in efficiency, robustness, and precision. Compared to the previous best method in indoor pose estimation, our lite MatchFormer has only 45% GFLOPs, yet achieves a +1.3% precision gain and a 41% running speed boost. The large MatchFormer reaches state-of-the-art on four different benchmarks, including indoor pose estimation (ScanNet), outdoor pose estimation (MegaDepth), homography estimation and image matching (HPatch), and visual localization (InLoc).

Keywords: Feature Matching · Vision Transformers

1 Introduction

Matching two or more views of a scene is the core of many basic computer vision tasks, *e.g.*, Structure-from-Motion (SfM) [19,24], Simultaneous Localization and Mapping (SLAM) [4,11], relative pose estimation [16], and visual localization [31,37,46], *etc.* For vision-based matching, classical *detector-based* methods (see Fig. 1(a)), coupled with hand-crafted local features [10,30], are computationally intensive due to the high dimensionality of local features [31,55]. Recent works [22,27,42] based on deep learning focus on learning detectors

---

* Equal contribution
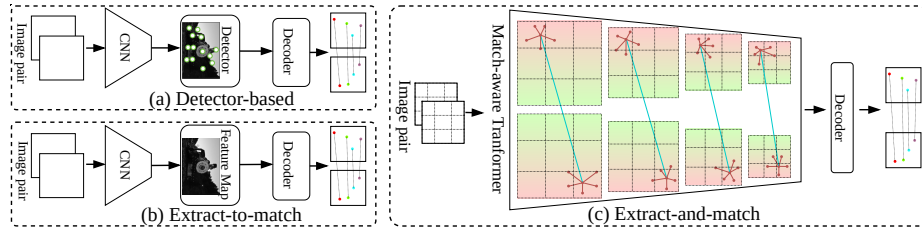† Correspondence: kailun.yang@kit.edu

Fig. 1. Feature matching pipelines. While (a) *detector-based* methods coupled with feature descriptors, (b) *extract-to-match* methods fail to make use of the matching capacity of the encoder. Self- and cross-attention are interleaved inside each stage of the match-aware transformer to perform a novel (c) *extract-and-match* pipeline.

and local descriptors using Convolutional Neural Networks (CNNs). Some partial transformer-based methods [36,14] only design an attention-based decoder and remain the *extract-to-match* pipeline (see Fig. 1(b)). For instance, while COTR [14] feeds CNN-extracted features into a transformer-based decoder, SuperGlue [32] and LoFTR [36] only apply attention modules atop the decoder. Overburdening the decoder, yet neglecting the matching capacity of the encoder, makes the whole model computationally inefficient.

Rethinking local feature matching, in reality, one can perform feature extraction and matching simultaneously by using a pure transformer. We propose an *extract-and-match* pipeline shown in Fig. 1(c). Compared to the *detector-based* methods and the *extract-to-match* pipeline, our new scheme is more in line with human intuition, which learns more respective features of image pairs while paying attention to their similarities [53]. To this end, a novel transformer termed *MatchFormer* is proposed, which helps to achieve multi-wins in precision, efficiency, and robustness of feature matching. For example, compared to LoFTR [36] in Fig. 2, MatchFormer with lower GFLOPs is more robust in low-textured scenes and achieves higher matching number, speed, and accuracy.

More specifically, for improving computational efficiency and the robustness in matching low-texture scenes, we put forward *interleaving* self- and cross-attention in MatchFormer to build a matching-aware encoder. In this way, the local features of the image itself and the similarities of its paired images can be learned simultaneously, so called *extract-and-match*, which relieves the overweight decoder and makes the whole model efficient. The cross-attention arranged in earlier stages of the encoder robustifies feature matching, particularly, in low-texture indoor scenarios or with less training samples outdoors, which makes MatchFormer more suitable for real-world applications where large-scale data collection and annotation are infeasible. To extract continuous patch information and embed location information, a novel *positional patch embedding (PosPE)* method is designed in the matching-aware encoder, which can enhance the detection of low-level features. Additionally, the lite and large versions *w.r.t.* feature resolutions, each with two efficient attention modules [34,43], are fully investigated to overcome the massive calculations in transformers when dealing with fine features. Furthermore, MatchFormer, with a hierarchical trans-
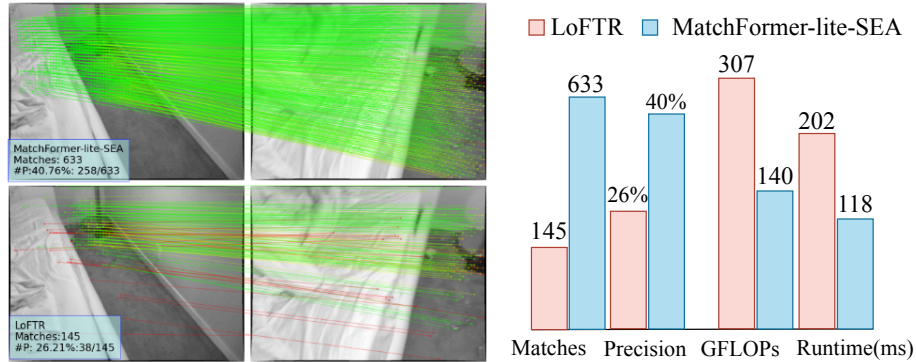
Fig. 2. Comparison between MatchFormer and LoFTR. With 45% GFLOPs of LoFTR, our efficient MatchFormer boosts the running speed by 41%, while delivering more robust matches and a higher matching precision on such a low-texture indoor scenario. Green color in the figure refers to correct matches and red color to mismatches.

former, conducts multi-level feature extraction in the encoder and multi-scale feature fusion in the decoder, which contribute to the robustness of matching. Finally, for the precision, extensive experiments prove that MatchFormer achieves state-of-the-art performances of indoor location estimation on ScanNet [7], outdoor location estimation on MegaDepth [18], image matching and homography estimation on HPatches [1], and visual localization on InLoc [37].

In summary, the contributions of this paper include:
- We rethink local feature matching and propose a new *extract-and-match* pipeline, which enables synchronization of feature extraction and feature matching. The optimal combination path is delivered when *interleaving* self- and cross-attention modules within each stage of the hierarchical structure to enhance multi-scale features.
- We propose a novel vision transformer, *i.e.*, *MatchFormer*, equipped with a robust hierarchical transformer encoder and a lightweight decoder. Including lite and large versions and two attention modules, four variants of Match-Former are investigated.
- We introduce a simple and effective positional patch embedding method, *i.e.*, *PosPE*, which can extract continuous patch information and embed location information, as well as enhances the detection of low level features.
- MatchFormer achieves state-of-the-art scores on matching low-texture indoor images and is superior to previous *detector-based* and *extract-to-match* methods in pose estimation, homography estimation, and visual localization.

2   Related Work

Local Feature Matching. *Detector-based* methods [10,6,13,23] usually include five steps: detecting interest points, calculating visual descriptors, searching for nearest neighbor matches, rejecting incorrect matches, and estimating geometric transformations. In *extract-to-match* methods [10,27,36,32,17,39,35,28] de-

signed for feature matching, CNNs are normally adopted to learn dense and discriminative features. CAPS [42] fuses multi-resolution features extracted by CNNs and obtains the descriptor of each pixel through interpolation. DSM [38] strengthens detection and refines the descriptors by merging various frames and multiple scales extracted by CNNs. DRC-Net [17] obtains CNN feature maps of two different resolutions, generates two 4D matching tensors, and fuses them to achieve high-confidence feature matching. D2Net [10] obtains valid key points by detecting the local maximum of CNN features. R2D2 [27] adapts dilated convolutions [5,47] to maintain image resolution and predict each key points and descriptors. COTR [14], LoFTR [36], and QuadTree [39] follow sequential *extract-to-match* processing. In this work, we consider that feature extraction and similarity learning through a transformer synchronously, can provide matching-aware features in each stage of the hierarchical structure.

Vision Transformer. Transformer [9] excels at capturing long-distance dependency [41], making it outstanding in vision tasks such as classification [20,40,48], detection [3,43,56], semantic segmentation [52,44,50], image enhancement [51], and image synthesis [12]. For local-feature matching, only attention blocks of transformers have been used in recent works. For example, SuperGlue [32] and LoFTR [36] applied self- and cross-attention to process the features which were extracted from CNNs. Yet, attention can actually function as the backbone module for feature extraction instead of only being used in the decoder for CNNs. This has been verified in ViT [9], but mainly for classification and segmentation tasks [52,43]. It remains unclear whether it is transferable to the image feature matching. When a pure transformer framework is used to process local feature matching, the computation complexity will be exceedingly large. Besides, transformers often lack and miss local feature information [48]. In this paper, we put forward a fully transformer image matching framework. In our model, we design positional patch embedding to enhance the feature extraction and introduce interleaving attention to achieve efficient and robust feature matching.

3   Methodology

3.1   MatchFormer

As illustrated in Fig. 3, MatchFormer employs a hierarchical transformer, which comprises four stages to generate high-resolution coarse and low-resolution fine features for local feature matching. In four stages, the self- and cross-attention are arranged in an *interleaving* strategy. Each stage consists of two components: one *positional patch embedding (PosPE)* module, and a set of efficient attention modules. Then, the multi-scale features are fused by an FPN-like decoder. Finally, the coarse and fine features are passed to perform the coarse-to-fine matching, as introduced in LoFTR [36].

Extract-and-Match Pipeline. Unlike the *extract-to-match* LoFTR using attention on a single-scale feature map and only after feature extraction, we combine self- and cross-attention inside the transformer-based encoder and apply
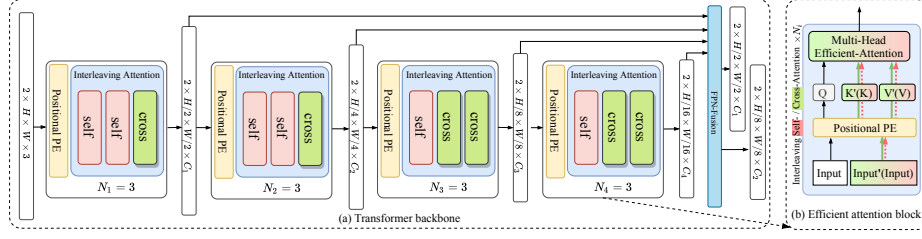
Fig. 3. MatchFormer architecture: (a) The transformer backbone generates high-resolution coarse features and low-resolution fine features; In (b), each attention block has interleaving-arranged self-attention (*w.r.t.* $\boldsymbol{Q}$, $\boldsymbol{K}$, $\boldsymbol{V}$ and red arrows) within the *input*, and cross-attention (*w.r.t.* $\boldsymbol{Q}$, $\boldsymbol{K'}$, $\boldsymbol{V'}$ and alternative green arrows) cross images (*input* and *input'*). Multi-head efficient-attention reduces the computation; Positional Patch Embedding (PE) completes the patch embedding and the position encoding.

on multiple feature scales (see Fig. 1). The combination of two types of attention modules enables the model to extract non-local features via self-attention and explore their similarities via cross-attention simultaneously, so called the *extract-and-match* scheme. As a new matching scheme, however, the difficulty lies in finding an effective and optimal combination strategy while maintaining the efficiency and robustness of the entire model. Thanks to the hierarchy nature of Transformers [9,43], we obtain two insights: (1) As the feature map at the shallow stage emphasizes textural information, *relatively* more self-attention are applied to extract the feature itself on the early stages. (2) As the feature map at the deep stage is biased toward semantic information, *relatively* more cross-attention are developed to explore the feature similarity on the later stages. These two observations lead us to design a novel *interleaving* strategy for joining self- and cross-attention.

Interleaving Self-/Cross-Attention. As shown in Fig. 3(a), the combination of self- and cross-attention modules are set at each stage in an *interleaving* strategy. Each block in Fig. 3(b) contains N attention modules, where each attention module is represented as self-attention or alternative cross-attention according to the input image pair. For self-attention, $\boldsymbol{Q}$ and ($\boldsymbol{K}$, $\boldsymbol{V}$) come from the same *input*, so the self-attention is responsible for feature extraction of the image itself. For cross-attention, ($\boldsymbol{K'}$, $\boldsymbol{V'}$) are from another *input'* of the image pair. Thus, the cross-attention learns the similarity of the image pair, resulting a match-aware transformer-based encoder. Within an attention block, self-attended features are extracted, while the similarity of the feature pair is located by the cross-attention. The strategy is more human-intuitive, which learns more respective features of image pairs while paying attention to their similarities.

Positional Patch Embedding (PosPE). Typical transformers [9], split the image ($H{\times}W{\times}3$) into patches with size of $P{\times}P$ and then flatten these patches into sequence with a size of $N{\times}C$, where $N{=}HW/P^2$. The process is difficult to gather location information around patches. As a result, low-level feature information cannot be acquired directly through the standard process [48], which severely restricts the local feature matching. In the case of standard Patch Em-
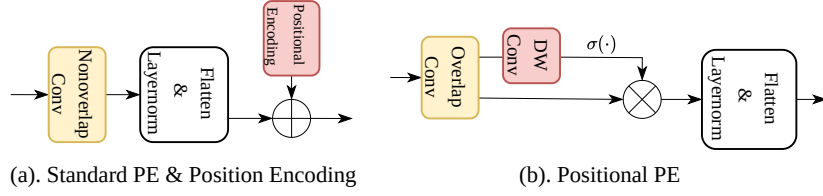
(a). Standard PE & Position Encoding   (b). Positional PE

Fig. 4. Comparison between different patch embedding modules.

beding (PE) in Fig. 4(a), the independent patch ignores the information around it and requires additional position encoding at the end. Therefore, we propose a simple but effective positional patch embedding (PosPE) method for capturing low feature information with few parameters, as shown in Fig. 4(b). It has a 7×7 convolution layer (with padding 3 and stride 2) in the first stage, and 3×3 convolution layers (all with padding 1 and stride 2) in later stages. A depth-wise 3×3 convolution is added to further enhance local features and encode positional information by its padding operation. The pixel-wise weights are then scaled by a sigmoid function $\sigma(\cdot)$ after the first step of convolution. Besides, our PosPE includes a first overlapping convolution that captures the continuous patch area information. PosPE augments the location information of patches and extracts denser features, which facilitates accurate feature matching.

Preliminaries on Efficient-Attention. After Patch Embedding, the query Q, key K, and value V are obtained, with the same $N \times C$ dimension according to the input resolution $N = H \times W$. The computation of the traditional attention is formulated as: $softmax((QK^T)/\sqrt{d})V$, where $\sqrt{d}$ is the scaling factor. However, the product of QK^T introduces a $O(N^2)$ complexity, which is prohibitive in large image resolutions and makes the model inefficient. To remedy this problem, we apply two kinds of efficient attention, *i.e.*, *Spatial Efficient Attention (SEA)* as in [43,44] or *Linear Attention (LA)* as in [34]. Then, $O(N^2)$ is reduced to $O(N^2/R)$ or $O(N)$. Hence, larger input feature maps can be well handled and processed while using a pure transformer-based encoder in the feature matching task.

Multi-scale Feature Fusion. Apart from the interleaving combination, there are four different stages in our hierarchical transformer encoder, in which the feature resolution shrinks progressively. Different from previous works [17,14] considering only the single-scale feature, MatchFormer fuses multi-scale features to generate dense and match-aware features for feature matching. As shown in Fig. 3(a), we flexibly adopt an FPN-like decoder in our architecture, because it can bring two benefits: (1) generating more robust coarse- and fine features for promoting the final matching; (2) creating a lightweight decoder without making the whole model computationally complex.

3.2   Model Settings

MatchFormer Variants. MatchFormer is available with its *lite* and *large* versions, as presented in Table 1. For the MatchFormer-lite models, we pick a lower resolution setting, which greatly increases the matching efficiency and ensures a certain matching accuracy. Therefore, we set MatchFormer-lite 4-stage features in the respective resolution of $\frac{1}{r_i} \in \{\frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}\}$ of the input. To promote

Table 1. MatchFormer-lite and -large with Linear Attention (LA) and Spatial Efficient Attention (SEA). C: the channel number of feature $\boldsymbol{F}$; K, S and P: the patch size, stride, and padding size of PosPE; E: the expansion ratio of MLP in an attention block; A: the head number of attention; R: the down-scale ratio of SEA.

Stage	MatchFormer-lite		MatchFormer-large		N_i
$\boldsymbol{F}_1$	$H/4{\times}W/4$	$K{=}7$, $S{=}4$, $P{=}3$, $E{=}4$	$H/2{\times}W/2$	$K{=}7$, $S{=}2$, $P{=}3$, $E{=}4$	
	$C_1{=}128$	**LA:** $A{=}8$; **SEA:** $A{=}1$, $R{=}4$	$C_1{=}128$	**LA:** $A{=}8$; **SEA:** $A{=}1$, $R{=}4$	${\times}3$
$\boldsymbol{F}_2$	$H/8{\times}W/8$	$K{=}3$, $S{=}2$, $P{=}1$, $E{=}4$	$H/4{\times}W/4$	$K{=}3$, $S{=}2$, $P{=}1$, $E{=}4$	
	$C_2{=}192$	**LA:** $A{=}8$; **SEA:** $A{=}2$, $R{=}2$	$C_2{=}192$	**LA:** $A{=}8$; **SEA:** $A{=}2$, $R{=}2$	${\times}3$
$\boldsymbol{F}_3$	$H/16{\times}W/16$	$K{=}3$, $S{=}2$, $P{=}1$, $E{=}4$	$H/8{\times}W/8$	$K{=}3$, $S{=}2$, $P{=}1$, $E{=}4$	
	$C_3{=}256$	**LA:** $A{=}8$; **SEA:** $A{=}4$, $R{=}2$	$C_3{=}256$	**LA:** $A{=}8$; **SEA:** $A{=}4$, $R{=}2$	${\times}3$
$\boldsymbol{F}_4$	$H/32{\times}W/32$	$K{=}3$, $S{=}2$, $P{=}1$, $E{=}4$	$H/16{\times}W/16$	$K{=}3$, $S{=}2$, $P{=}1$, $E{=}4$	
	$C_4{=}512$	**LA:** $A{=}8$; **SEA:** $A{=}8$, $R{=}1$	$C_4{=}512$	**LA:** $A{=}8$; **SEA:** $A{=}8$, $R{=}1$	${\times}3$
Output	Coarse: $H/4{\times}W/4, 128$ Fine: $H/8{\times}W/8, 192$		Coarse: $H/2{\times}W/2, 128$ Fine: $H/8{\times}W/8, 256$		

context learning for matching, feature embeddings with higher channel numbers are beneficial, which are set as $C_i{\in}\{128,192,256,512\}$ for four stages. In the MatchFormer-large models, higher resolution feature maps facilitate accurate dense matching. Hence, the $\frac{1}{r_i}$ and C_i are set as $\{\frac{1}{2},\frac{1}{4},\frac{1}{8},\frac{1}{16}\}$ and $\{128,192,256,512\}$ for the large MatchFormer.

Attention Module Variants. To fully explore the proposed *extract-and-match* scheme, each of the two MatchFormer variants has two attention variants. Here, we mainly investigate Linear Attention (LA) and Spatial Efficient Attention (SEA). Thus, there are four versions of MatchFormer as presented in Table 1. We found that they have different capabilities for recognizing features, making them suitable for various tasks. In the local feature matching, the density of features is different indoors and outdoors. We study the two kinds of attention in indoor (in Sec. 4.2) and outdoor (in Sec. 4.3) pose estimation, respectively.

4 Experiments

4.1 Implementation and Datasets

ScanNet. We use ScanNet [7] to train our indoor models. ScanNet is an indoor RGB-D video dataset with 2.5 million views in 1,513 scans with ground-truth poses and depth maps. The lack of textures, the ubiquitous self-similarity, and the considerable changes in viewpoint make ScanNet a challenging dataset for indoor image matching. Following [32], we select 230 million image pairs with the size of 640×480 as the training set and 1,500 pairs as the testing set.

MegaDepth. Following [10], we use MegaDepth [18] to train our outdoor models, which has 1 million internet images of 196 scenarios, and their sparse 3D point clouds are created by COLMAP [33]. We use 38,300 image pairs from 368 scenarios for training, and the same 1,500 testing pairs from [36] for evaluation.

Implementation Settings. On the indoor dataset ScanNet, MatchFormer is trained using Adam [15] with initial learning rate and batch size, setting for the

Table 2. Indoor pose estimation on ScanNet. The AUC of three different thresholds and the average matching precision (P) are evaluated.

Method	Pose estimation AUC (%)			P
	@5°	@10°	@20°	
ORB [30]+GMS [2] CVPR'17	5.21	13.65	25.36	72.0
D2-Net [10]+NN CVPR'19	5.25	14.53	27.96	46.7
ContextDesc [22]+RT [21] CVPR'19	6.64	15.01	25.75	51.2
SP [8]+NN CVPRW'18	9.43	21.53	36.40	50.4
SP [8]+PointCN [45] CVPR'18	11.40	25.47	41.41	71.8
SP [8]+OANet [49] ICCV'19	11.76	26.90	43.85	74.0
SP [8]+SuperGlue [32] CVPR'20	16.16	33.81	51.84	84.4
LoFTR [36] CVPR'21	22.06	40.80	57.62	87.9
LoFTR [36]+QuadTree [39] ICLR'22	23.90	43.20	60.30	89.3
MatchFormer-lite-LA	20.42	39.23	56.82	87.7
MatchFormer-lite-SEA	22.89	42.68	60.66	89.2
MatchFormer-large-LA	24.27	43.48	60.55	89.2
MatchFormer-large-SEA	**24.31**	**43.90**	**61.41**	**89.5**

lite version at 3×10^{-3} and 4, and for the large version at 3×10^{-4} and 2. In the case of the outdoor dataset MegaDepth, MatchFormer is trained using Adam with initial learning rate and batch size, setting for the lite version at 3×10^{-3} and 2, and for the large version at 3×10^{-4} and 1. To compare LoFTR and Match-Former at different data scales on outdoor pose estimation task, both use 8 A100 GPUs, otherwise use 64 A100 GPUs following LoFTR [36]. We perform Image Matching, Homography Estimation, and InLoc Visual Localization experiments using the model trained with MatchFormer-large-LA on MegaDepth.

4.2   Indoor Pose Estimation

Indoor pose estimation is highly difficult due to wide areas devoid of textures, a high degree of self-similarity, scenes with complicated 3D geometry, and frequent perspective shifts. Faced with all these challenges, MatchFormer with interleaved self- and cross-attention modules still functions well as unfolded in the results.

Metrics. Following [32], we provide the area under the cumulative curve (AUC) of the pose error at three different thresholds $(5°, 10°, 20°)$. The camera pose is recovered by using RANSAC. We report the matching precision (P), the probability of a true match if its epipolar is smaller than 5×10^{-4}.

Quantitative Results. As shown in Table 2, MatchFormer demonstrates exceptional performance on the low-texture indoor pose estimation task. The matching precision (P) of MatchFormer-large-SEA reaches the state-of-the-art level of 89.5%. Benefiting from the *extract-and-match* strategy, MatchFormer-large-SEA can bring +5.1% improvement over the *detector-based* SuperGlue, +1.6% over the *extract-to-match* LoFTR. Pose estimation AUC of MatchFormer is also significantly superior to *detector-based* SuperGlue. Compared to LoFTR, Match-Former provides a more pronounced pose estimation AUC by boosting (+2.25%, +3.1%, +3.79%) at three thresholds of $(5°, 10°, 20°)$. The LoFTR model is recently adapted by a complex decoder with QuadTree Attention [39]. However, MatchFormer maintains its lead (+0.41%, +0.70%, +1.11%) with the *extract-and-match* strategy. Additionally, compared to LoFTR, our lightweight MatchFormer-lite-SEA has only 45% GFLOPs, yet achieves a +1.3% precision gain and a 41% running speed boost. More details of the efficiency comparison will be presented
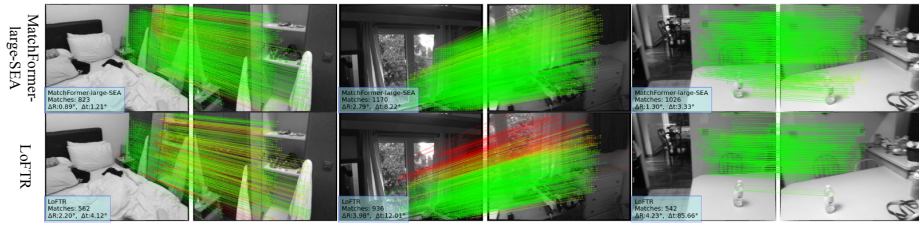
Fig. 5. Qualitative visualization of MatchFormer and LoFTR [36]. MatchFormer achieves higher matching numbers and more correct matches in low-texture scenes.
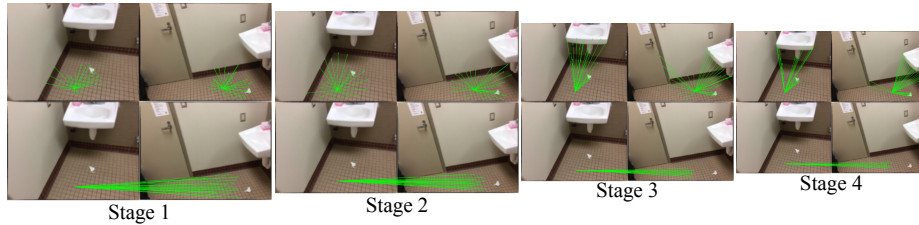

Fig. 6. Visualization of self- and cross-attention at 4 stages of MatchFormer. Cross-attention focuses on learning the similarity across paired images and gradually refines the matching range, while self-attention focuses on detecting features of the image itself and enabling long-range dependencies.

in Table 7. Comparing SEA and LA, we found that the spatial scaling operation in SEA has benefits in handling low-texture features, thus it is more suited for indoor scenes and provides better results.

Qualitative Results. The indoor matching results are in Fig. 5. In challenging feature-sparse indoor scenes, it can reliably capture global information to assure more matches and high accuracy. Thus, the pose solved by matching prediction has a lower maximum angle error (ΔR) and translation error (Δt). Due to the hierarchical transformer and interleaving-attention design, the receptive field of MatchFormer exceeds that of CNN-based methods. It confirms that applying cross-attention modules earlier for learning feature similarity robustifies low-texture indoor matching, which is in line with our *extract-and-match* pipeline.

Self- and Cross-attention Visualization. To further investigate the effectiveness of interleaving attention in MatchFormer, the features of self- and cross-attention modules in four stages are shown in Fig. 6. Self-attention connects obscure points with surrounding points, while cross-attention learns relationship between points across images. Specifically, self-attention enables the query point to associate surrounding textural features in the shallow stage, and it enables the query point to connect to semantic features in the deep stage. As the model deepens, cross-attention will narrow the range of query points detected across images, rendering the matching much easier and more fine-grained. Finally, these four stages of features are blended, empowering the model to perform accurate feature matching in low-texture scenes.

Table 3. Outdoor pose estimation on MegaDepth. † represents training on different percentages of datasets, which requires 8 GPUs for training.

Method	Data percent	Pose estimation AUC (%) @5°	@10°	@20°	P
SP [8]+SuperGlue [32] CVPR'20	100%	42.18	61.16	75.95	–
DRC-Net [17] NeurIPS'20	100%	27.01	42.96	58.31	–
LoFTR [36] CVPR'21	100%	52.80	69.19	81.18	94.80
MatchFormer-lite-LA	100%	48.74	65.83	78.81	97.55
MatchFormer-lite-SEA	100%	48.97	66.12	79.07	97.52
MatchFormer-large-LA	100%	**52.91** (+0.11)	**69.74** (+0.55)	**82.00** (+0.82)	**97.56** (+2.76)
Robustness with less training data and fewer GPU resources:					
LoFTR†	10%	38.81	54.53	67.04	83.64
MatchFormer†	10%	42.92 (+4.11)	58.33 (+3.80)	70.34 (+3.30)	85.08 (+1.44)
LoFTR†	30%	47.38	64.77	77.68	91.94
MatchFormer†	30%	49.53 (+2.15)	66.74 (+1.97)	79.43 (+1.75)	94.28 (+2.34)
LoFTR†	50%	48.68	65.49	77.62	92.54
MatchFormer†	50%	50.13 (+1.45)	66.71 (+1.22)	79.01 (+1.39)	94.89 (+2.35)
LoFTR†	70%	49.08	66.03	78.72	93.86
MatchFormer†	70%	51.22 (+2.14)	67.44 (+1.41)	79.73 (+1.01)	95.75 (+1.89)
LoFTR†	100%	50.85	67.56	79.96	95.18
MatchFormer†	100%	**53.28** (+2.43)	**69.74** (+2.18)	**81.83** (+1.87)	**96.59** (+1.41)

4.3   Outdoor Pose Estimation

Outdoor pose estimation presents unique challenges compared to indoors. In particular, outdoor scenes have greater variations in lighting and occlusion. Still, Matchformer achieves outstanding performance in outdoor scenes.

Metrics. We present the same AUC of the pose error as in the indoor pose estimation task. The matching precision pipolar distance threshold is 1×10^{-4}.

Results. As shown in Table 3, MatchFormer noticeably surpasses the *detector-based* SuperGlue and DRC-Net, as well as the *extract-to-match* LoFTR. Our MatchFormer-lite-LA model also achieves great performance. It can deliver a higher matching precision (P) with 97.55%, despite being much lighter. Note that MatchFormer-large-SEA using the partially optimized SEA will raise an out-of-memory issue. Here, we recommend to use the memory-efficient LA in the high-resolution outdoor scenes. Our MatchFormer-large-LA model achieves consistent state-of-the-art performances on both metrics of AUC and P.

Robustness and Resource-Efficiency. It is reasonable to evaluate the robustness of the model when only less training data and fewer training resources are available in practical applications. Therefore, we further train MatchFormer-large-LA and LoFTR (marked with † in Table 3) using different percentages of datasets and on fewer resources with 8 GPUs. First, compared to LoFTR†, MatchFormer† obtains consistent improvements on different constrained data scales, *i.e.*, the first {10,30,50,70,100} percentages of the original dataset. It proves that MatchFormer has more promise in data-hungry real-world applications. Second, training with the same 100% data on different GPU resources, LoFTR† has $(-1.95\%, -1.63\%, -1.22\%)$ performance drops at three AUC thresholds of $(5°, 10°, 20°)$ when using 8 GPUs instead of 64 GPUs. In contrast, Match-Former maintains the stable and surprising accuracy, which shows that our method is more resource-friendly and easier to reproduce.
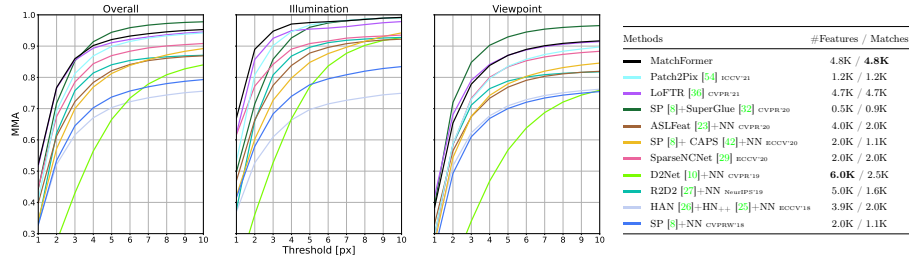
Methods	#Features / Matches
MatchFormer	4.8K / **4.8K**
Patch2Pix [54] ICCV'21	1.2K / 1.2K
LoFTR [36] CVPR'21	4.7K / 4.7K
SP [8]+SuperGlue [32] CVPR'20	0.5K / 0.9K
ASLFeat [29]+NN CVPR'20	4.0K / 2.0K
SP [8]+ CAPS [42]+NN ECCV'20	2.0K / 1.1K
SparseNCNet [29] ECCV'20	2.0K / 2.0K
D2Net [19]+NN CVPR'19	**6.0K** / 2.5K
R2D2 [27]+NN NeurIPS'19	5.0K / 1.6K
HAN [26]+HN++ [25]+NN ECCV'18	3.9K / 2.0K
SP [8]+NN CVPRW'18	2.0K / 1.1K

Fig. 7. Image matching on HPatches. The mean matching accuracy (MMA) at thresholds from [1,10] pixels, and the number of matches and features are reported.

4.4 Image Matching

Metrics. On the standard image matching task of HPatches sequences based on sequences with illumination or viewpoint change, we evaluate MatchFormer by detecting correspondences between pairs of input images. Following the experimental setup of Patch2Pix [54], we report the mean matching accuracy (MMA) at thresholds from [1,10] pixels, and the number of matches and features.

Results. Fig. 7 illustrates the results for the experiments with illumination and viewpoint changes, along with the MMA. Under varying illumination conditions, our method provides the best performance. On overall (the threshold ≤3 pixels), Matchformer performs optimally at precision levels. While other methods can only account for lighting changes or viewing angles changes, MatchFormer is reasonably compatible and maintains its functionality when the viewpoint changes. Thanks to the match-aware encoder, a larger number of features and matches, both 4.8K, are obtained. The results reveal the effectiveness of our *extract-and-match* strategy for image matching.

4.5 Homography Estimation

Metrics. To evaluate how the matches contribute to the accuracy of the geometric relations estimation, we assess MatchFormer in the homography estimation on HPatches benchmark [1]. The proportion of accurately predicted homographies with an average corner error distance less than 1/3/5 pixels is reported.

Results. As shown in Table 4, the large-LA MatchFormer achieves excellent performance on the HPatches benchmark in homography estimation. It reaches the best level in the face of illumination variations, delivering the accuracy of (0.75,0.95,0.98) at 1/3/5 pixel errors. Additionally, MatchFormer obtains highest number of matches with 4.8K. To evaluate the robustness and resource-efficiency, we also execute experiments with varying dataset percentages in Table 4. Compared to LoFTR†, MatchFormer† performs significantly better in homography experiments, and is relatively unaffected by the limited training data. Match-Former trained with 30% data has a better performance in illumination variations. One reason is that the accuracy of the geometry relation estimation is related to accurate matches, as well as the distribution and number of matches [54]. Training with fewer GPUs on 100% data, while LoFTR† has noticeable performance drops, MatchFormer† maintains stable performance and requires fewer

Table 4. Homography estimation on HPatches. † represents training on different percentages of datasets, which requires 8 GPUs for training.

Method	Data percent	Overall	Illumination Accuracy (%,$\epsilon < 1/3/5$ px)	Viewpoint	#Matches
SP [8] ᴄᴠᴘʀᴡ'18	100%	0.46/0.78/0.85	0.57/0.92/0.97	0.35/0.65/0.74	1.1K
D2Net [10] ᴄᴠᴘʀ'19	100%	0.38/0.71/0.82	0.66/0.95/**0.98**	0.12/0.49/0.67	2.5K
R2D2 [27] ɴᴇᴜʀIᴘs'19	100%	0.47/0.77/0.82	0.63/0.93/**0.98**	0.32/0.64/0.70	1.6K
ASLFeat [23] ᴄᴠᴘʀ'20	100%	0.48/0.81/0.88	0.62/0.94/**0.98**	0.34/0.69/0.78	2.0K
ASLFeat [23] ᴄᴠᴘʀ'20 + ClusterGNN [35]	100%	0.51/0.83/0.89	0.61/**0.95**/0.98	0.42/0.72/**0.82**	-
SP [8] + SuperGlue [32] ᴄᴠᴘʀ'20	100%	0.51/0.82/0.89	0.60/0.92/**0.98**	0.42/0.71/0.81	0.5K
SP [8] + CAPS [42] ᴇᴄᴄᴠ'20	100%	0.49/0.79/0.86	0.62/0.93/**0.98**	0.36/0.65/0.75	1.1K
SP [8] + ClusterGNN [35] ᴄᴠᴘʀ'22	100%	0.52/**0.84**/**0.90**	0.61/0.93/**0.98**	**0.44**/**0.74**/0.81	-
SIFT + CAPS [42] ᴇᴄᴄᴠ'20	100%	0.36/0.77/0.85	0.48/0.89/0.95	0.26/0.65/0.76	1.5K
SparseNCNet [29] ᴇᴄᴄᴠ'20	100%	0.36/0.65/0.76	0.62/0.92/0.97	0.13/0.40/0.58	2.0K
Patch2Pix [54] ᴄᴠᴘʀ'21	100%	0.50/0.79/0.87	0.71/**0.95**/**0.98**	0.30/0.64/0.76	1.3K
LoFTR [36] ᴄᴠᴘʀ'21	100%	**0.55**/0.81/0.86	0.74/**0.95**/**0.98**	0.38/0.69/0.76	4.7K
MatchFormer	100%	**0.55**/0.81/0.87	**0.75**/**0.95**/**0.98**	0.37/0.68/0.78	**4.8K**
Robustness with less training data and fewer GPU resources:					
LoFTR†	10%	0.50/0.78/0.84	0.74/0.95/**0.98**	0.28/0.63/0.71	3.6K
MatchFormer†	10%	0.50/0.78/0.84	0.72/0.93/0.97	0.30/0.64/0.71	4.0K
LoFTR†	30%	0.52/0.80/0.86	0.74/0.96/**0.98**	0.32/**0.66**/0.74	4.1K
MatchFormer†	30%	**0.57**/**0.81**/0.86	**0.78**/**0.97**/0.98	0.36/**0.66**/0.74	4.4K
LoFTR†	50%	0.52/0.79/0.85	0.73/0.95/**0.98**	0.32/0.65/0.73	4.1K
MatchFormer†	50%	0.54/0.78/0.85	0.75/0.95/**0.98**	0.35/0.62/0.74	**4.5K**
LoFTR†	70%	0.52/0.79/0.85	0.74/0.94/**0.98**	0.31/0.64/0.73	4.1K
MatchFormer†	70%	0.55/0.79/0.86	0.76/0.94/**0.98**	0.35/0.64/0.75	**4.5K**
LoFTR†	100%	0.52/0.79/0.86	0.74/0.93/**0.98**	0.32/0.65/0.74	4.2K
MatchFormer†	100%	0.54/0.79/**0.87**	0.74/0.95/**0.98**	**0.36**/**0.66**/**0.77**	**4.5K**

Table 5. Visual localization on InLoc. We report the percentage of correctly localized queries under specific error thresholds, following the HLoc [31] pipeline.

Method	Localized Queries (%, $0.25m/0.5m/1.0m$, $10°$)	
	DUC1	DUC2
SP [8] + NN ᴄᴠᴘʀᴡ'18	40.4 / 58.1 / 69.7	42.0 / 58.8 / 69.5
D2Net [10] + NN ᴄᴠᴘʀ'19	38.4 / 56.1 / 71.2	37.4 / 55.0 / 64.9
R2D2 [27] + NN ɴᴇᴜʀIᴘs'19	36.4 / 57.6 / 74.2	45.0 / 60.3 / 67.9
SP [8] + SuperGlue [32] ᴄᴠᴘʀ'20	49.0 / 68.7 / 80.8	53.4 / **77.1** / 82.4
SP [8] + CAPS [42] + NN ᴇᴄᴄᴠ'20	40.9 / 60.6 / 72.7	43.5 / 58.8 / 68.7
SP [8] + ClusterGNN [35] ᴄᴠᴘʀ'22	47.5 / 69.7 / 79.8	53.4 / **77.1** / 84.7
ASLFeat [23] + SuperGlue [32] ᴄᴠᴘʀ'20	51.5 / 66.7 / 75.8	53.4 / 76.3 / 84.0
ASLFeat [23] + ClusterGNN [35] ᴄᴠᴘʀ'22	**52.5** / 68.7 / 76.8	55.0 / 76.0 / 82.4
SIFT + CAPS [42] + NN ᴇᴄᴄᴠ'20	38.4 / 56.6 / 70.7	35.1 / 48.9 / 58.8
SparseNCNet [29] ᴇᴄᴄᴠ'20	41.9 / 62.1 / 72.7	35.1 / 48.1 / 55.0
Patch2Pix [54] ᴄᴠᴘʀ'21	44.4 / 66.7 / 78.3	49.6 / 64.9 / 72.5
LoFTR-OT [36] ᴄᴠᴘʀ'21	47.5 / 72.2 / 84.8	54.2 / 74.8 / **85.5**
MatchFormer	46.5 / **73.2** / **85.9**	**55.7** / 71.8 / 81.7

training resources for success. These experiments sufficiently prove that our new *extract-and-match* pipeline has higher robustness than the *extract-to-match* one used in previous methods.

4.6   Visual Localization on InLoc

Metrics. A robust local feature matching method ensures accurate visual localization. To evaluate our local feature matching method MatchFormer, we test it on the InLoc [37] benchmark for visual localization. Referring to SuperGlue [32], we utilize MatchFormer as the feature matching step to complete the visual localization task along the localization pipeline HLoc [31].

Results. As shown in Table 5, on the InLoc benchmark for visual localization, MatchFormer reaches a level comparable to the current state of art methods SuperGlue and LoFTR. Interleaving attention in the MatchFormer backbone

Table 6. Ablation study with different structures, attention arrangements and PEs.

Method	Self	Cross	PosPE	StdPE	Pose estimation AUC (%)			P
					@5°	@10°	@20°	
LoFTR [36] CVPR'21					15.47	31.72	48.63	82.6
❶ Convolution					7.36	18.17	32.21	76.1
❷ Self-only	✓				9.48	22.68	38.10	81.3
❸ Cross-only		✓			13.88	29.98	46.89	84.4
❹ Sequential	✓	✓			14.75	31.03	48.27	85.0
❺ Sequential	✓	✓	✓		17.32	34.85	52.71	85.8
❻ Interleaving	✓	✓		✓	16.53	34.63	52.31	85.9
❼ Interleaving	✓	✓	✓		**18.01**	**35.87**	**53.46**	**86.7**

Table 7. Efficiency analysis. Runtime in ms, GFLOPs @ 640×480.

Method	#Params	GFLOPs	Runtime	P
LoFTR [36]	**11**	307	202	87.9
LoFTR [36]+QuadTree [39]	13	393	234	89.3
MatchFormer-lite-LA	22	**97**	140	87.7
MatchFormer-large-LA	22	389	246	87.8
MatchFormer-lite-SEA	23	140	**118**	89.2
MatchFormer-large-SEA	23	414	390	**89.5**

enables robust local feature matching in indoor scenes with large low-texture areas and repetitive structures.

4.7 MatchFormer Structural Study

Performing the *extract-and-match* strategy in a pure transformer, the layout between self- and cross-attention co-existing inside each stage of MatchFormer is a critical point to achieve efficient and robust feature matching. The structural study is conducted to explore the sweet spot to arrange attention modules.

Ablation Study of Interleaving. To verify the rationality of the model design, models in Table 6 are ablated according to different backbone structures, attention arrangements and patch embedding modules. Models are trained with 10% data of ScanNet. Such a setting is one for efficiency and another is that the robustness between models is validated with less training data. By comparing ❶ and ❷, we establish that the transformer with self-attention significantly improves the matching precision (P, +5.2%) compared to utilizing the convolutional extractor, which shows the long-range dependency can robustify the local feature matching. While the structure in ❷ contains only self-attention in between, the structure in ❸ with cross-attention can bring a +3.1% performance gain, which demonstrates the benefits of leaning feature similarity inside a transformer. The sequential structures (❹❺) apply pure self-attention in the early stages and pure cross-attention in the later stages, while our interleaving structures (❻❼) apply mix self-/cross-attention in each stage. Our structures improve the overall performance, which adaptively inserts self-/cross-attention in multi-scale stages, and it is in line with our statement about the *extract-and-match* strategy in transformers. The comparison between ❹ and ❺ indicates that the proposed PosPE is capable of completing the fixed position encoding and it comes with a +0.8% gain. Our PosPE in ❼ can enhance the accuracy by +0.8% compared with standard PE (StdPE) in ❻, demonstrating that PosPE is more robust. Our interleaving model in ❼ surpasses LoFTR by a large margin (+4.1% @ P), indicating that MatchFormer is more robust, not only in low-texture indoor scenes, but also with less training data.

Feature Maps Comparison. As shown in Fig. 8, we visualize the feature maps of the ablation experiment ❷ and ❼ of Table 6. In both shallow and deep layers, our interleaving attention structure enables MatchFormer to capture dense features and learn feature similarities, such as the paired regions highlighted
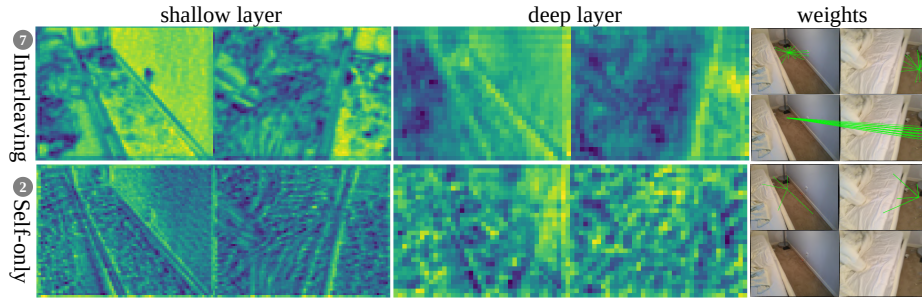
Fig. 8. Feature maps comparison between interleaving and self-only attention in shallow and deep layers (from the last layer of the stage-2 and stage-3, respectively).

in yellow. The model with only self-attention tends to extract features in each individual image and neglects the matching-aware features across images, *i.e.*, without cross-attention weights. As a result, the self-attention model without cross-attention model will be incapable of matching local features when the image features are sparse (*i.e.*, low-texture scenes).

Runtime and Efficiency Analysis. Aside from verifying the effectiveness of arranging self- and cross-attention in an interleaving manner, MatchFormer is still supposed to be computationally efficient. The comparisons of efficiency results including the #Parameters (M), GFLOPs, and runtime (ms) are detailed in Table 7. Based on a 3080Ti GPU, MatchFormer is compared against the previous transformer-based LoFTR. We quantify the average runtime it takes for MatchFormer to complete a single image pair on the ScanNet test set, which includes 1,500 pairs of images in the resolution of 640×480. MatchFormer-lite-SEA is clearly much faster, speeding up the matching process by 41%, although a higher number of parameters is required. Additionally, we compute the GFLOPs of the two approaches to determine their computing costs and storage demands. The GFLOPs of MatchFormer-lite-SEA are only 45% of those of LoFTR. Yet, our model achieves a +1.3% precision gain. Thanks to interleaving self- and cross-attention in between, our lite and large MatchFormers achieve state-of-the-art performances with respect to previous methods on various tasks.

5   Conclusions

Rethinking local feature matching from a novel *extract-and-match* perspective with transformers, we propose the MatchFormer framework equipped with a matching-aware encoder by interleaving self- and cross-attention for performing feature extraction and feature similarity learning synchronously. MatchFormer circumvents involving a complex decoder as used in the *extract-to-match* methods and adopts a lightweight FPN-like decoder to fuse multi-scale features. Experiments show that MatchFormer achieves state-of-the-art performances in indoor and outdoor pose estimation on the ScanNet and MegaDepth benchmarks, and in both homography estimation and image matching on the HPatches benchmark, as well as in visual localization on the InLoc benchmark.

References

1. Balntas, V., Lenc, K., Vedaldi, A., Mikolajczyk, K.: HPatches: A benchmark and evaluation of handcrafted and learned local descriptors. In: CVPR (2017) 3, 11
2. Bian, J., Lin, W.Y., Matsushita, Y., Yeung, S.K., Nguyen, T.D., Cheng, M.M.: GMS: Grid-based motion statistics for fast, ultra-robust feature correspondence. In: CVPR (2017) 8
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV (2020) 4
4. Chen, H., Hu, W., Yang, K., Bai, J., Wang, K.: Panoramic annular SLAM with loop closure and global optimization. Applied Optics (2021) 1
5. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence (2018) 4
6. Cheng, R., Wang, K., Lin, L., Yang, K.: Visual localization of key positions for visually impaired people. In: ICPR (2018) 3
7. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scan-Net: Richly-annotated 3D reconstructions of indoor scenes. In: CVPR (2017) 3, 7
8. DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperPoint: Self-supervised interest point detection and description. In: CVPRW (2018) 8, 10, 11, 12
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021) 4, 5
10. Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., Sattler, T.: D2-net: A trainable CNN for joint detection and description of local features. CVPR (2019) 1, 3, 4, 7, 8, 11, 12
11. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. IEEE Transactions on Pattern Analysis and Machine Intelligence (2018) 1
12. Esser, P., Rombach, R., Ommer, B.: Taming transformers for high-resolution image synthesis. In: CVPR (2021) 4
13. Fang, Y., Wang, K., Cheng, R., Yang, K.: CFVL: A coarse-to-fine vehicle localizer with omnidirectional perception across severe appearance variations. In: IV (2020) 3
14. Jiang, W., Trulls, E., Hosang, J., Tagliasacchi, A., Yi, K.M.: COTR: Correspondence transformer for matching across images. In: ICCV (2021) 2, 4, 6
15. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015) 7
16. Li, S., Yuan, L., Sun, J., Quan, L.: Dual-feature warping-based motion model estimation. In: ICCV (2015) 1
17. Li, X., Han, K., Li, S., Prisacariu, V.: Dual-resolution correspondence networks. NeurIPS (2020) 3, 4, 6, 10
18. Li, Z., Snavely, N.: MegaDepth: Learning single-view depth prediction from internet photos. In: CVPR (2018) 3, 7
19. Lindenberger, P., Sarlin, P.E., Larsson, V., Pollefeys, M.: Pixel-perfect structure-from-motion with featuremetric refinement. In: ICCV (2021) 1
20. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: ICCV (2021) 4

21. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (2004) 8
22. Luo, Z., Shen, T., Zhou, L., Zhang, J., Yao, Y., Li, S., Fang, T., Quan, L.: ContextDesc: Local descriptor augmentation with cross-modality context. In: CVPR (2019) 1, 8
23. Luo, Z., Zhou, L., Bai, X., Chen, H., Zhang, J., Yao, Y., Li, S., Fang, T., Quan, L.: ASLFeat: Learning local features of accurate shape and localization. In: CVPR (2020) 3, 11, 12
24. Ma, W.C., Yang, A.J., Wang, S., Urtasun, R., Torralba, A.: Virtual correspondence: Humans as a cue for extreme-view geometry. In: CVPR (2022) 1
25. Mishchuk, A., Mishkin, D., Radenovic, F., Matas, J.: Working hard to know your neighbor's margins: Local descriptor learning loss. NeurIPS (2017) 11
26. Mishkin, D., Radenovic, F., Matas, J.: Repeatability is not enough: Learning affine regions via discriminability. In: ECCV (2018) 11
27. Revaud, J., De Souza, C., Humenberger, M., Weinzaepfel, P.: R2D2: Reliable and repeatable detector and descriptor. In: NeurIPS (2019) 1, 3, 4, 11, 12
28. Revaud, J., Leroy, V., Weinzaepfel, P., Chidlovskii, B.: PUMP: Pyramidal and uniqueness matching priors for unsupervised learning of local descriptors. In: CVPR (2022) 3
29. Rocco, I., Arandjelović, R., Sivic, J.: Efficient neighbourhood consensus networks via submanifold sparse convolutions. In: ECCV (2020) 11, 12
30. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: An efficient alternative to SIFT or SURF. In: ICCV (2011) 1, 8
31. Sarlin, P.E., Cadena, C., Siegwart, R., Dymczyk, M.: From coarse to fine: Robust hierarchical localization at large scale. In: CVPR (2019) 1, 12
32. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperGlue: Learning feature matching with graph neural networks. In: CVPR (2020) 2, 3, 4, 7, 8, 10, 11, 12
33. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: CVPR (2016) 7
34. Shen, Z., Zhang, M., Zhao, H., Yi, S., Li, H.: Efficient attention: Attention with linear complexities. In: WACV (2021) 2, 6
35. Shi, Y., Cai, J.X., Shavit, Y., Mu, T.J., Feng, W., Zhang, K.: ClusterGNN: Cluster-based coarse-to-fine graph neural network for efficient feature matching. In: CVPR (2022) 3, 12
36. Sun, J., Shen, Z., Wang, Y., Bao, H., Zhou, X.: LoFTR: Detector-free local feature matching with transformers. In: CVPR (2021) 2, 3, 4, 7, 8, 9, 10, 11, 12, 13
37. Taira, H., Okutomi, M., Sattler, T., Cimpoi, M., Pollefeys, M., Sivic, J., Pajdla, T., Torii, A.: InLoc: Indoor visual localization with dense matching and view synthesis. In: CVPR (2018) 1, 3, 12
38. Tang, S., Tang, C., Huang, R., Zhu, S., Tan, P.: Learning camera localization via dense scene matching. In: CVPR (2021) 4
39. Tang, S., Zhang, J., Zhu, S., Tan, P.: Quadtree attention for vision transformers. ICLR (2022) 3, 4, 8, 13
40. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: ICML (2021) 4
41. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017) 4
42. Wang, Q., Zhou, X., Hariharan, B., Snavely, N.: Learning feature descriptors using camera pose supervision. In: ECCV (2020) 1, 4, 11, 12

43. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: ICCV (2021) 2, 4, 5, 6

44. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: SegFormer: Simple and efficient design for semantic segmentation with transformers. In: NeurIPS (2021) 4, 6

45. Yi, K.M., Trulls, E., Ono, Y., Lepetit, V., Salzmann, M., Fua, P.: Learning to find good correspondences. In: CVPR (2018) 8

46. Yoon, S., Kim, A.: Line as a visual sentence: Context-aware line descriptor for visual localization. IEEE Robotics and Automation Letters (2021) 1

47. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. ICLR (2016) 4

48. Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Tay, F.E.H., Feng, J., Yan, S.: Tokens-to-token ViT: Training vision transformers from scratch on ImageNet. In: ICCV (2021) 4, 5

49. Zhang, J., Sun, D., Luo, Z., Yao, A., Zhou, L., Shen, T., Chen, Y., Quan, L., Liao, H.: Learning two-view correspondences and geometry using order-aware network. In: ICCV (2019) 8

50. Zhang, J., Yang, K., Constantinescu, A., Peng, K., Müller, K., Stiefelhagen, R.: Trans4Trans: Efficient transformer for transparent object segmentation to help visually impaired people navigate in the real world. In: ICCVW (2021) 4

51. Zhang, Z., Jiang, Y., Jiang, J., Wang, X., Luo, P., Gu, J.: STAR: A structure-aware lightweight transformer for real-time image enhancement. In: ICCV (2021) 4

52. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H.S., Zhang, L.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: CVPR (2021) 4

53. Zhong, G., Pun, C.M.: Subspace clustering by simultaneously feature selection and similarity learning. Knowledge-Based Systems (2020) 2

54. Zhou, Q., Sattler, T., Leal-Taixe, L.: Patch2Pix: Epipolar-guided pixel-level correspondences. In: CVPR (2021) 11, 12

55. Zhou, Z., Wu, Q.M.J., Wan, S., Sun, W., Sun, X.: Integrating SIFT and CNN feature matching for partial-duplicate image detection. IEEE Transactions on Emerging Topics in Computational Intelligence (2020) 1

56. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable DETR: Deformable transformers for end-to-end object detection. In: ICLR (2021) 4