

# LeCoT: revisiting network architecture for two-view correspondence pruning

Luanyuan DAI, Xiaoyu DU &amp; Jinhui TANG\*

*School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China*

Received 20 May 2024/Revised 20 September 2024/Accepted 21 November 2024/Published online 4 January 2026

**Abstract** Two-view correspondence pruning aims to accurately remove incorrect correspondences (outliers) from initial ones and is widely applied to various computer vision tasks. Current popular strategies adopt multilayer perceptron (MLP) as the backbone, supplemented by additional modules to enhance the network ability to handle context information, which is a known limitation of MLPs. In contrast, we introduce a novel perspective for capturing correspondence context information without extra design modules. To this end, we design a two-view correspondence pruning network called LeCoT, which can naturally leverage global context information at different stages. Specifically, the core design of LeCoT is the spatial-channel fusion transformer block, a newly proposed component that efficiently utilizes both spatial and channel global context information among sparse correspondences. In addition, we integrate the proposed prediction block that utilizes correspondence features from intermediate stages to generate a probability set, which acts as guiding information for subsequent learning phases, allowing the network to more effectively capture robust global context information. Notably, this prediction block progressively refines the probability set, thereby mitigating the issue of information loss that is common in the traditional one. Extensive experiments prove that the proposed LeCoT outperforms state-of-the-art methods in correspondence pruning, relative pose estimation, homography estimation, visual localization, and 3D reconstruction tasks. The code is provided in <https://github.com/Dailuanyuan2024/LeCoT-Revisiting-Network-Architecture-for-Two-View-Correspondence-Pruning>.

**Keywords** correspondence pruning, transformer, spatial and channel information, context information, relative pose estimation

**Citation** Dai L Y, Du X Y, Tang J H. LeCoT: revisiting network architecture for two-view correspondence pruning. *Sci China Inf Sci*, 2026, 69(1): 112104, <https://doi.org/10.1007/s11432-024-4555-x>

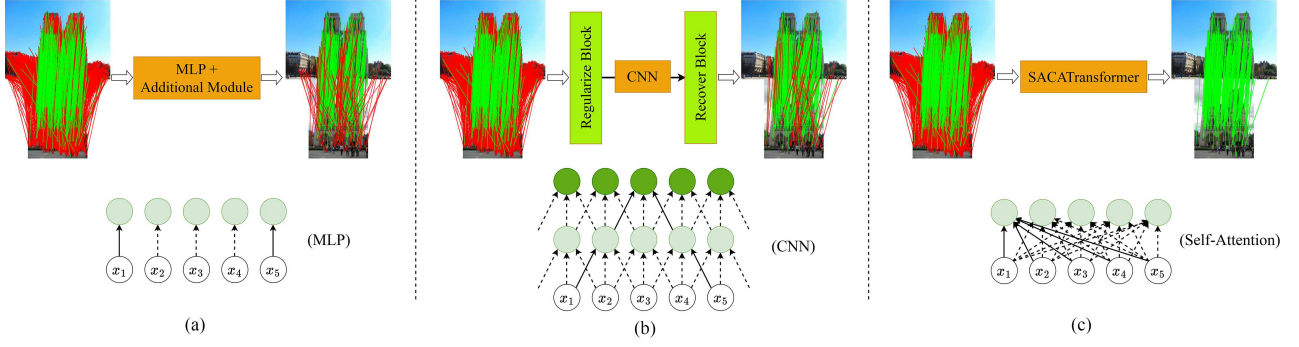
## 1 Introduction

Finding good-quality two-view correspondences is an elementary task in computer vision, which strives to build sparse correspondences between images from two different views and estimate the geometry relationship. It serves as the precondition for various challenging computer vision tasks, e.g., visual localization [1], simultaneous location and mapping (SLAM) [2], and point cloud registration [3].

The most common geometry matching pipeline includes two stages: feature extraction and matching, and correspondence pruning. In the initial stage, researchers expend considerable effort on local feature descriptors (such as SIFT [4] and SuperPoint [5]), and subsequently construct a putative correspondence set based on their similarities. Unfortunately, there are massive outliers because of rotations, viewpoint changes and illumination changes in image pairs. To mitigate the adverse effects of outliers, it is customary to eliminate them and choose a dependable subset predominantly comprising correct correspondences (i.e., inliers) to robustly estimate the relative pose. The process, named correspondence pruning, aims to remove outliers to precisely gain correspondences and geometry relationships.

Traditional correspondence pruning methods such as random sample consensus (RANSAC) [6], vector field consistency (VFC) [7], locality preserving matching (LPM) [8] and its variants [9–11] typically rely on handcrafted features and heuristic rules to identify and remove outliers in specific scenes. However, in real-world scenarios, there are complex scene variations and considerable outliers, which often render them ineffective. In contrast, deep learning-based techniques possess stronger learning and generalization capabilities, enabling them to learn abstract features and high-level representations from vast amounts of data. Therefore, researchers have turned their attention to deep learning-based techniques. In CNe [12], correspondence pruning is treated as a binary classification problem under the multilayer perceptron (MLP) framework. Unfortunately, MLP processes each sparse correspondence independently, as shown in Figure 1(a). The intrinsic individual feature extraction capability of MLP poses

\* Corresponding author (email: [jinhuitang@njust.edu.cn](mailto:jinhuitang@njust.edu.cn))



**Figure 1** (Color online) Comparative illustrations of correspondence pruning for three distinct frameworks: (a) MLP-based, (b) CNN-based, and (c) the proposed LeCoT, where the corresponding diagrams below are MLP, CNN, and self-attention (a crucial component of the Transformer architecture), respectively. The arrows represent the information transmission. The solid lines in (a) indicate that MLP processes each correspondence separately. The solid lines in (b) indicate the information transmission of a CNN with a convolutional kernel size of  $3 \times 3$  and stride 1. As the number of CNN layers increases, the receptive field gradually expands. The solid lines in (c) illustrate the information propagation of self-attention, where attention distributions are computed at each layer to determine the correlation between any two correspondences, thus aggregating global context information.

a significant problem: the lack of context information, which is crucial for handling two-view correspondences and geometry learning [13]. Therefore, current popular approaches [14–21] design additional modules, inserted into the MLP backbone, to capture context information, as depicted in Figure 1(a). Despite the promising performance has been attained with this framework, drawbacks still persist. Apparently, these challenges associated with these supplementary modules have impeded the acquisition and consolidation of context information, consequently restricting the performance of this framework (see Figure 1(a)).

Moreover, ConvMatch [13] advocates for the adoption of convolutional neural network (CNN) as the backbone to tackle this challenge, due to its ability to gradually integrate local and global information. While this approach partly alleviates the issue of MLP neglecting relationships among correspondences, some problems still endure. Theoretically, increasing the number of CNN layers can capture global context information from local context information. However, when considering the YFCC100M dataset [22] for relative pose estimation in ConvMatch [13], which involves  $N = 2000$  correspondences and utilizes the specified CNN architecture (with a  $3 \times 3$  convolutional kernel, a stride of 1, and no padding) as depicted in Figure 1(b), achieving comprehensive global context information would require approximately 1000 layers. This is obviously unrealistic for this task. Fortunately, ConvMatch [13] converts unordered correspondences into a  $16 \times 16$  vector field (data compression, as shown in Figure 1(b)), but this still requires 8 CNN layers to capture global context information. However, the study only uses 6 CNN layers, making it difficult to capture global context information adequately. Furthermore, the forced conversion of unordered data into ordered ones (data compression) can result in information loss and, disrupt the inherent structure and distribution of the data [23–25]. Additionally, the local perception mechanism in CNN may lead to a gradual loss of information propagation within the network, particularly concerning correlated information from distant positions [26, 27]. In this scenario, one might wonder how to further address this issue: whether to continue refining the context-capturing module to compensate for the inherently context-agnostic MLP, deepen the CNN-based network, or replace it directly with a network inherently possessing stronger context perception capabilities?

Transformer inherently possesses the outstanding capability to effectively capture global context information, as evidenced by its remarkable performance in computer vision tasks such as image classification [28] and semantic segmentation [29, 30]. Hence, we opt for Transformer over MLP or CNN to extract deep features for sparse correspondences, which can naturally address the issue of insufficient global context information among correspondences without additional module designs and data compression (see Figure 1(c)). Transformer is originally designed to process ordered data, primarily in natural language processing (NLP). However, sparse correspondences are unordered, so we choose Transformer without positional encoding. Following the idea, we construct a novel and straightforward two-view correspondence pruning framework, named LeCoT, which can naturally harness global context information from various stages without extra modules and data compression.

In conclusion, our primary contributions can be outlined as follows.

- From a new perspective, we propose a simple and effective two-view correspondence pruning framework LeCoT, without additional module designs and data compression, which can naturally utilize global context information at various stages.
- We introduce a novel spatial-channel fusion transformer block as the backbone to simultaneously leverage

spatial and channel global context information among sparse correspondences. To our knowledge, this is the first instance of utilizing the complete Transformer structure as the backbone to tackle the correspondence pruning problem.

- We use the proposed prediction block to progressively obtain the probability set from correspondence features learned at the intermediate stage, which can serve as guiding information to find robust global context information.
- The proposed LeCoT consistently surpasses current state-of-the-art networks in correspondence pruning, relative pose estimation, homography estimation, visual localization, and 3D reconstruction.

The remaining of this paper is organized as follows. In Section 2, we review correspondence pruning and Transformer related work. In Section 3, we present the overall framework and details of the proposed LeCoT. In Section 4, we show the experimental results. In Section 5, we make a conclusion.

## 2 Related work

### 2.1 Learning-based correspondence pruning networks

PointNet [31] pioneers learning-based approaches by employing an MLP-based backbone to classify and segment irregular point clouds. CNe [12] draws inspiration from the PointNet-like architecture, which is the first one to treat correspondence pruning as a binary classification problem with the MLP backbone. But the MLP backbone fails to fully capture context information from sparse correspondences, so various additional modules and strategies have been proposed.

Some networks [15, 16, 19, 32, 33] draw lessons from the attention machine [34] to aggregate information locally or globally. Specifically, ACNe [15] introduces a novel context normalization mechanism called attentive context normalization (ACN), which captures context information by applying attention weighting to local regions, thereby enhancing the model comprehension of input data. LAGA [16] captures context information locally and globally by the attention mechanism to enhance the network ability. MS<sup>2</sup>DGNet [19] leverages attention to cluster information within the constructed sparse semantic dynamic graph to enhance the network performance. Other networks [14, 17–19, 35–37] are influenced by GNNs to construct graphs in local or global levels. In particular, OANet [14] enhances the ability to handle spatial geometric transformations and ordering information between views effectively by incorporating a clustering layer into the intermediate of CNe [12], where the clustering layer is implemented based on GNNs. LMCNet [17] employs GNNs to achieve motion coherence and extends it as a differentiable layer in the network. CLNet [18] and U-Match [38] sequentially integrate local and global contexts by GNNs, progressively eliminating outliers. Based on CLNet [18], BCLNet [36] addresses the issue of ignoring interactions between different contexts. GCT [37] combines a Transformer and GNNs on an MLP-based backbone to provide additional information. TrGa [39] addresses the issue of information confusion in GNNs and adopts Transformers with stronger feature extraction capabilities as the backbone in correspondence pruning. In addition, LFC [40] and DeMatch [41] introduce a Siamese network and a diffusion model into the correspondence pruning process, respectively.

While the mentioned networks have advanced in the network structure and shown good performance, they all rely on the MLP-based backbone, which struggles to address the challenge of being context-agnostic. Hence, ConvMatch [13] proposes converting unordered correspondences into ordered vector fields and employs CNN as the backbone to tackle this issue. While it has made some progress, the conversion of unordered correspondences to ordered vector fields may result in information loss and disrupt the inherent structure and distribution of the data. Furthermore, the local perception mechanism in the CNN may lead to a gradual loss of information propagation, especially concerning distant information, making it difficult to capture global context information. Therefore, we construct a simple yet effective two-view correspondence pruning network LeCoT, from a fresh perspective, without additional modules and data compression, which can naturally leverage global context information.

### 2.2 Transformers in vision related tasks

Vision transformer (ViT) [28] first bridges the gap between NLP and computer vision (CV) by employing the standard Transformer [34] to handle full-sized images, achieving state-of-the-art results in the ImageNet [42] classification task. Inspired by the success of ViT, numerous diverse visual Transformers have been proposed for various computer vision tasks, yielding remarkable results. Compared to ViT [28], Image Transformer [43] boasts a more streamlined design and introduces several improvements, including a local self-attention mechanism and dynamic feature map resizing. TransUNet [44] is the first medical image segmentation framework that combines a Transformer with the U-Net, fully leveraging the strengths of the Transformer in capturing global information and the

U-Net in extracting local features, thereby achieving powerful image segmentation performance. SSFormers [45] is proposed for few-shot learning, which can identify task-relevant features and suppress task-irrelevant features. Segmenter [29] is the first one to successfully apply the Transformer architecture to the semantic segmentation task, achieving notable success in pixel-level image segmentation tasks. TransCrowd [46] redefines the weakly-supervised crowd counting problem from the perspective of sequence-to-count based on Transformer. Pyramid vision transformer (PVT) [47] introduces a pyramid-style attention mechanism, enabling the model to better handle image features at different scales. T2T-ViT [48] introduces a novel approach allowing training vision Transformer models from scratch on the ImageNet dataset, leveraging the strengths of both Transformer and CNN architectures, achieving performance comparable to pre-trained vision transformers without using pre-trained weights. DETR [49] is an end-to-end object detection network without redundancy boxes, which greatly optimizes and promotes the development of object detection. Swin Transformer [50] uses shifted windows to obtain multiple scale information to deal with different sizes of objects in each image. Inspired by BERT [51], MAE [52] masks a high proportion of each input image, so that it can achieve self-supervised learning. ViLT [53] uses Transformers well in the vision-and-language task, which greatly reduces network running cost and has comparable performance. To the best of our knowledge, the proposed LeCoT is the first two-view correspondence pruning network with the Transformer-based backbone, building upon the highly successful ViT [28].

### 3 Methodology

#### 3.1 Problem formulation

Given a pair of matching images  $(I, I')$ , we first use local features (SIFT [4], SuperPoint [5]) to detect keypoints and obtain descriptors. After that, we build an initial correspondence set  $C$  by an NN matcher:

$$C = \{c_1; c_2; \dots; c_N\} \in \mathbb{R}^{N \times 4}, \quad (1)$$

where  $c_i = (x_i, y_i, u_i, v_i)$  is defined as a correspondence between two keypoints  $(x_i, y_i)$  and  $(u_i, v_i)$  at the corresponding positions of the image pair, and both of them are normalized under camera intrinsics. But unfortunately, as mentioned earlier, the initial correspondence set usually contains a large portion of outliers. Therefore, our job is to eliminate outliers while preserving inliers as much as possible.

In this paper, we use the proposed LeCoT to achieve this goal. As shown in Figure 2, we introduce an improved verification framework inspired by [6, 18], which has two pruning modules. Our goal is to gain the final probability set  $P = \{p_1; p_2; \dots; p_N\}$  with  $p_i \in [0, 1]$ . Specifically, a guiding information probability set  $P_s$  is proposed to guide and adjust the training process. After that, we obtain the pruned guiding information probability set  $\hat{P}_s$ , the first estimated probability set  $\hat{P}_1$  and the first subset  $\hat{C}_1$  with a higher inlier rate after the first pruning module. Next,  $\hat{P}_s$ ,  $\hat{P}_1$  and  $\hat{C}_1$  are used to produce  $\hat{P}_2$  and  $\hat{C}_2$  in the second pruning module. Then, a weighted eight-point algorithm [12] (model estimation in Figure 2) is used to calculate an essential matrix  $\hat{E}$ . Ultimately, we use a verification operation to count the final probability set  $P$ . The above process can be written as

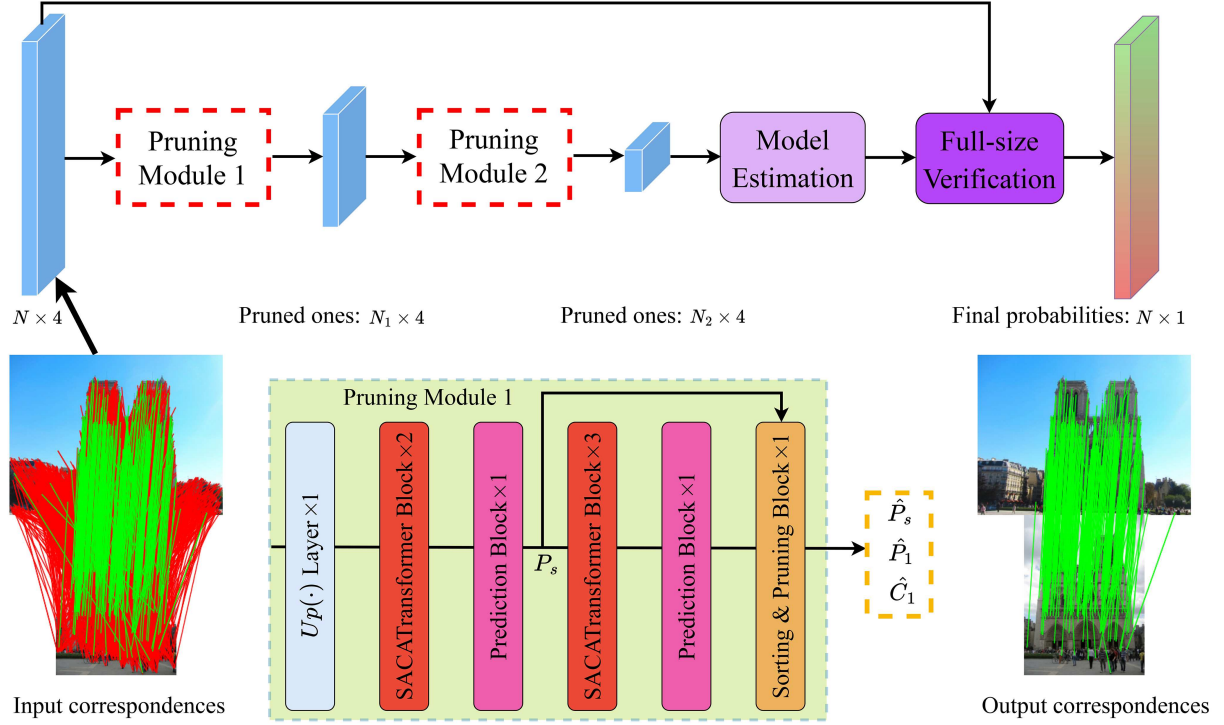
$$\begin{aligned} (\hat{P}_s, \hat{P}_1, \hat{C}_1) &= f_{1\phi}(C), \quad (\hat{P}_2, \hat{C}_2) = f_{2\psi}(\hat{P}_s, \hat{P}_1, \hat{C}_1), \\ \hat{E} &= g(\hat{P}_2, \hat{C}_2), \quad P = Ver(\hat{E}, C), \end{aligned} \quad (2)$$

where  $f_{1\phi}(\cdot)$  and  $\hat{P}_1$  are the first pruning module with learnable parameter  $\phi$  and the first estimated probability set, respectively;  $f_{2\psi}(\cdot, \cdot)$  and  $\hat{P}_2$  are the second pruning module's;  $g(\cdot, \cdot)$  denotes the weighted eight-point algorithm;  $Ver(\cdot, \cdot)$  is the full-size verification operation, in which correspondences that satisfy the estimated model  $\hat{E}$  are labeled as inliers, otherwise labeled as outliers.

#### 3.2 Main structure of LeCoT

The overall LeCoT architecture is simple and shown in Figure 2, which includes three main components: two pruning modules, a model estimation module, and a full-size verification module. The model estimation module utilizes the last pruned reliable correspondence subset  $\hat{C}_2 \in \mathbb{R}^{N_2 \times 4}$  to estimate the essential matrix, which is then passed to the full-size verification module to classify all correspondences. Particularly, each pruning module is made up of an  $Up(\cdot)$  layer, five SACATransformer blocks, two prediction blocks and a Sorting&Pruning block. Firstly, due to the low-dimensionality of the correspondence set hindering the extraction of deep features, as a routine procedure, we convert it into a high-dimensional correspondence feature set  $F \in \mathbb{R}^{N \times d}$  as the input correspondence feature set,

$$F = Up(C), \quad (3)$$



**Figure 2** (Color online) The pipeline of the proposed LeCoT. The input is an initial correspondence set  $C \in \mathbb{R}^{N \times 4}$  and the output is the final probability set  $P \in \mathbb{R}^{N \times 1}$  through an iterative pruning strategy. This process aims to distill more reliable candidates to estimate the parametric model. Each pruning module comprises an  $Up(\cdot)$  layer for dimensionality transformation, two proposed prediction blocks, five proposed spatial-channel fusion transformer (SACATransformer) blocks, and a sorting & pruning block.  $P_s$ ,  $\hat{P}_s$ ,  $\hat{P}_1$  and  $\hat{C}_1$  are the guiding information probability set, the pruned guiding information probability set, the first estimated probability set and the first subset with a higher inlier rate after the first pruning module, respectively.

where the  $Up(\cdot)$  layer is utilized to map the correspondence set from a low-dimensional space into a high-dimensional space. We opt for a simple MLP to fulfill this task.

Then, we utilize SACATransformer blocks to capture spatial and channel global context information for the unordered correspondence features, rather than context-agnostic MLP. In this process, we use the proposed prediction block (see Subsection 3.5) to obtain the guiding information  $P_s$ , which is then passed to the Sorting&Pruning block to obtain the pruned guiding information  $\hat{P}_s$ , which is one of the inputs of the second pruning module. By integrating the aforementioned modules, we establish a progressive correspondence pruning network.

### 3.3 Spatial-channel fusion transformer

A vanilla Transformer block consists of two PreNorms (PNs), one multi-head self-attention (MSA) and a FeedForward (FF), as depicted in Figure 3(a). We choose the layer normalization as the PreNorm, because it can better handle variable length data. Followed by ViT [28], a FeedForward is made up of two linear layers and a GELU layer. Notably, the multi-head self-attention mechanism in the vanilla Transformer block typically operates on the spatial dimension of the input sequence. Therefore, we refer to it as spatial multi-head self-attention (S-MSA), as illustrated in Figure 3. Taking the correspondence feature set  $F$  as an example, which is successively put into a PreNorm and an S-MSA followed by a residual structure. After that, the output  $\tilde{F}$  is passed through a PreNorm and a FeedForward followed by a residual structure to obtain  $\hat{F}$ . The above operations can be written as

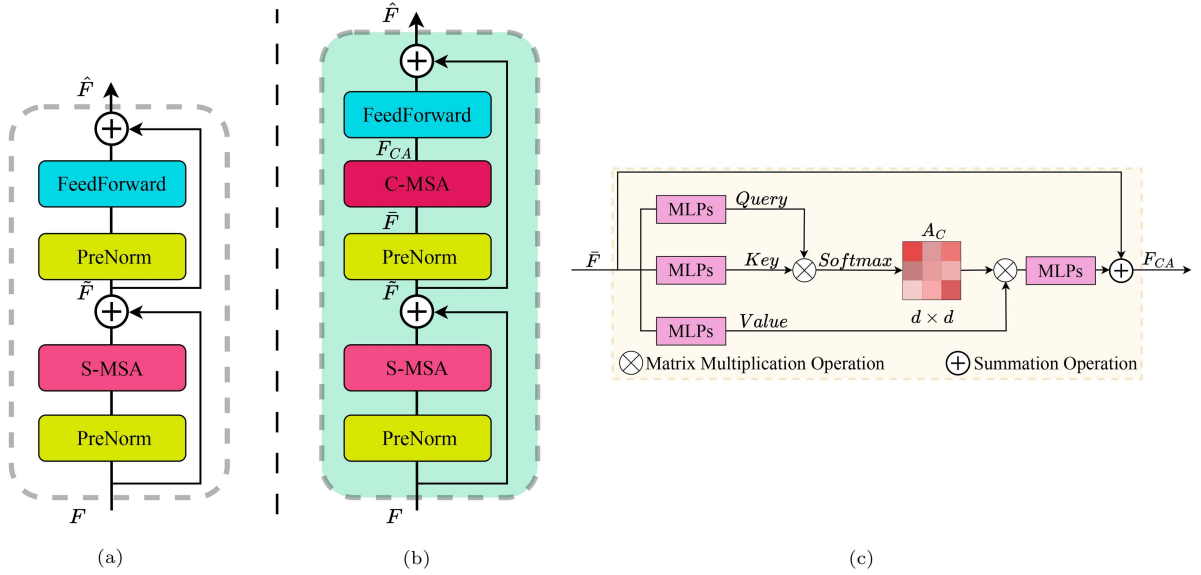
$$\tilde{F} = \text{S-MSA}(\text{PN}(F)) + F, \quad (4)$$

$$\hat{F} = \text{FF}(\text{PN}(\tilde{F})) + \tilde{F}. \quad (5)$$

However, we find that using the vanilla Transformer is not optimal because it lacks consideration of the global context information about correspondence features in the channel dimension. Therefore, we incorporate the proposed channel multi-head self-attention (C-MSA), between the PreNorm and the FeedForward of the vanilla Transformer, as shown in Figure 3(b). The reconfigured block is termed as spatial-channel fusion transformer, abbreviated as SACATransformer. Thus, Eq. (5) can be rewritten as

$$\hat{F} = \text{FF}(\text{C-MSA}(\text{PN}(\tilde{F}))) + \tilde{F}. \quad (6)$$





**Figure 3** (Color online) Structures of the vanilla Transformer block and the SACATransformer block. (a) The vanilla transformer block obtained from ViT; (b) the proposed SACATransformer block, which can capture global context information from spatial and channel dimensions; (c) the structure of channel multi-head self-attention (C-MSA).

We will provide a detailed explanation of C-MSA in Subsection 3.4.

### 3.4 Channel multi-head self-attention

We modify the standard multi-head self-attention mechanism [34] by computing *Query*, *Key* and *Value* along the channel dimension. As illustrated in Figure 3(c), we first process the input features using simple MLPs to obtain *Query*, *Key* and *Value*. Then, we compute the similarity matrix  $A_C \in \mathbb{R}^{d \times d}$  by multiplying *Query* and *Key*, followed by a matrix multiplication operation between the similarity matrix  $A_C$  and *Value*. Finally, the resulting output is passed through MLPs to further learn abstract features within the input, followed by element-wise summation. The specific steps are as follows:

$$Query, Key, Value = \text{MLPs}(\tilde{F}), \quad (7)$$

$$A_C = \text{Softmax}(Query \times Key^T), \quad (8)$$

$$F_{CA} = \sum (A_C \times Value) + \tilde{F}, \quad (9)$$

where  $\times$  and  $\text{Softmax}(\cdot)$  present a matrix multiplication operation and a softmax operation, respectively;  $\sum$  represents the element-wise summation operation.

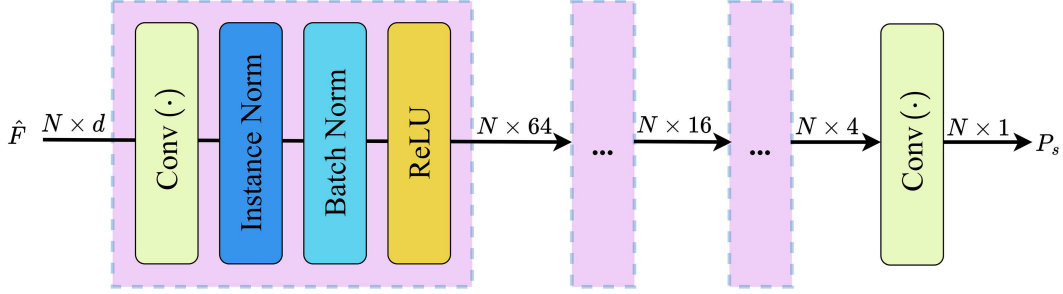
### 3.5 Prediction block

In Figure 2, we utilize the proposed prediction block to obtain the probability set as the guiding information, which can be written as

$$P_s = \text{Pre}(\hat{F}), \quad (10)$$

where  $\text{Pre}(\cdot)$  represents the proposed prediction block, which includes  $\text{Conv}(\cdot)$ , Instance Norm, Batch Norm and ReLU functions.

A common practice, such as in OA-Net++ [14], is to directly use a projection function to map the channel dimension from  $d$  to 1 as a basis for judging the correctness of correspondences. However, this approach may result in significant information loss and may not be sufficient to fully represent the input features [23]. Furthermore, the network employing the proposed prediction block yields superior performance compared to directly using a simple projection network, like  $\text{Conv}(\cdot)$ . Therefore, we opt for gradually reducing the dimensionality, as shown in Figure 4, which can mitigate information loss.



**Figure 4** (Color online) The structure of the prediction block. The input is the correspondence feature  $\hat{F}$  obtained by the first two SACA-Transformer blocks and the output is the guiding information  $P_s$ .

### 3.6 Loss function

Following previous studies [14, 18], we select a hybrid loss function to iteratively train the proposed LeCoT:

$$L_{oss} = L_e(\hat{E}, E) + \beta L_c, \quad (11)$$

where  $L_e$  is a geometric loss between an estimated model  $\hat{E}$  and the true one  $E$ ;  $\beta$  is a weighting factor to balance these two losses;  $L_c$  is a binary classification loss with a proposed adaptive temperature [18] and can be defined as

$$L_c = \sum_{i=1}^I \left( H(\sigma(\tau_i \odot \hat{P}_s), y) + H(\sigma(\tau_i \odot \hat{P}_i), y) \right), \quad (12)$$

where  $I$  is the iteration number;  $H(\cdot, \cdot)$  denotes a binary cross entropy function;  $\tau_i$  is an adaptive temperature vector to mitigate the influence of label ambiguity;  $\hat{P}_s$  and  $\hat{P}_i$  are the pruned guiding information and the  $i$ -th pruning module probability set, respectively;  $y$  is the binary ground truth label set;  $\odot$  and  $\sigma(\cdot)$  represent a Hadamard product and a sigmoid function, respectively.  $L_e$  is a geometric loss, which can be written as

$$L_e(E, \hat{E}) = \frac{(p'^T \hat{E} p)^2}{\|E p\|_{[1]}^2 + \|E p\|_{[2]}^2 + \|E^T p'\|_{[1]}^2 + \|E^T p'\|_{[2]}^2}, \quad (13)$$

where  $E$  and  $\hat{E}$  are the ground truth and estimated essential matrices, respectively;  $p$  and  $p'$  are virtual correspondence coordinates obtained by the ground truth  $E$ .

### 3.7 Implementation details

Network input is an  $N \times 4$  correspondence set, where  $N$  is up to 2000. There are two pruning modules in total, with a pruning rate of 0.5, resulting in a final set of  $N/4$  reliable correspondences. Channel dimension  $d$  is 128. SACATransformer block number  $L$  is 5, where 4-head attention is used. Batchsize is 32, and  $\beta$  in (11) is 0.5. We choose Adam optimizer with a learning rate of  $10^{-3}$ . And a warmup strategy is used to train LeCoT. Clearly, a linearly growing rate is used for the first 10k iterations, after that the learning rate begins to decrease and reduce for every 20k iterations with a factor of 0.4. Experiments are carried out on NVIDIA GTX 3090 GPUs.

## 4 Experiments

### 4.1 Correspondence pruning

Correspondence pruning is the process of identifying and eliminating outliers from initial correspondences to ensure accurate ones, which can enhance the accuracy and robustness of matching between the two views.

**Datasets.** We opt for Yahoo's YFCC100M [22] and SUN3D [54] datasets as outdoor and indoor scenes, respectively. In the outdoor dataset, 68 sequences serve as training data, while the remaining 4 sequences are allocated for testing. For the indoor dataset, 239 sequences are chosen for training, leaving 15 sequences for network evaluation.

**Evaluation protocols.** *Precision* ( $P$ ), *Recall* ( $R$ ) and *F-score* ( $F$ ) are selected as metrics in correspondence pruning. *Precision* ( $P$ ) measures the accuracy of correct matches by dividing the number of correct matches by the total number of samples classified as positive. *Recall* ( $R$ ) quantifies the completeness of correct matches by

**Table 1** Quantitative comparative results of correspondence pruning on YFCC100M and SUN3D with SIFT. The best result in each column is in bold.

Method	References	Size (MB)	YFCC100M (%)			SUN3D (%)		
			<i>P</i>	<i>R</i>	<i>F</i>	<i>P</i>	<i>R</i>	<i>F</i>
CNe [12]	CVPR2018	<b>0.39</b>	52.84	85.68	65.37	46.11	83.92	59.37
OA-Net++ [14]	ICCV2019	2.47	55.78	85.93	67.65	46.15	84.36	59.66
ACNe [15]	CVPR2020	0.41	55.62	85.47	67.39	46.16	84.01	59.58
LMC-Net [17]	CVPR2021	0.93	60.12	87.50	71.27	47.23	84.60	60.62
CL-Net [18]	ICCV2021	1.27	74.89	76.79	75.83	59.97	68.35	63.89
MS <sup>2</sup> DG-Net [19]	CVPR2022	2.61	59.11	88.40	70.85	46.95	84.55	60.37
ConvMatch [13]	AAAI2023	7.49	60.03	89.19	71.76	48.01	<b>85.19</b>	61.41
LFC [40]	ACM MM 2023	1.73	60.84	88.66	72.16	48.14	85.09	61.49
GCT [37]	AAAI 2024	4.09	77.00	79.02	78.00	61.12	69.34	64.97
DeMatch [41]	CVPR 2024	5.86	60.48	<b>89.79</b>	72.28	48.25	84.72	61.48
LeCoT (Ours)	–	3.74	<b>77.70</b>	80.43	<b>79.04</b>	<b>61.00</b>	68.77	<b>64.65</b>

assessing the ratio of the number of correct matches to the total number of samples actually belonging to the positive class. *F-score* (*F*) is the harmonic mean of *Precision* and *Recall*, providing a balanced assessment of a network performance.

**Baselines.** Ten learning-based networks (CNe [12], OA-Net++ [14], ACNe [15], LMC-Net [17], CL-Net [18], MS<sup>2</sup>DG-Net [19], ConvMatch [13], LFC [40], GCT [37], and DeMatch [41]) are chosen as baselines.

**Results.** Correspondence pruning is an important foundation for subsequent computer vision tasks and we test LeCoT and baselines on outdoor and indoor scenes. As summarized in Table 1, the proposed LeCoT outperforms other methods significantly on both *Precision* (*P*) and the comprehensive metric *F*, showing an approximately 1 absolute percentage points over the second-best network (GCT) on outdoor scenes. Additionally, we find that the parameter quantity of MS<sup>2</sup>DG-Net is most similar to our LeCoT from Table 1. Therefore, we conduct a partial comparison of typical visualization results between them in Figure 5. It is evident that LeCoT surpasses MS<sup>2</sup>DG-Net in correspondence pruning to enhance network performance. Specifically, the proposed LeCoT significantly outperforms the comparison MS<sup>2</sup>DG-Net in wide baselines (the 1st, 2nd, and 3rd columns), large changes in lighting (the 4th column), lack of texture information (the 5th and 8th columns) and repetitive structures (the 6th and 7th columns). In summary, the proposed LeCoT achieves optimal results in the comprehensive index *F* under indoor and outdoor scenes for the correspondence pruning task.

## 4.2 Relative pose estimation

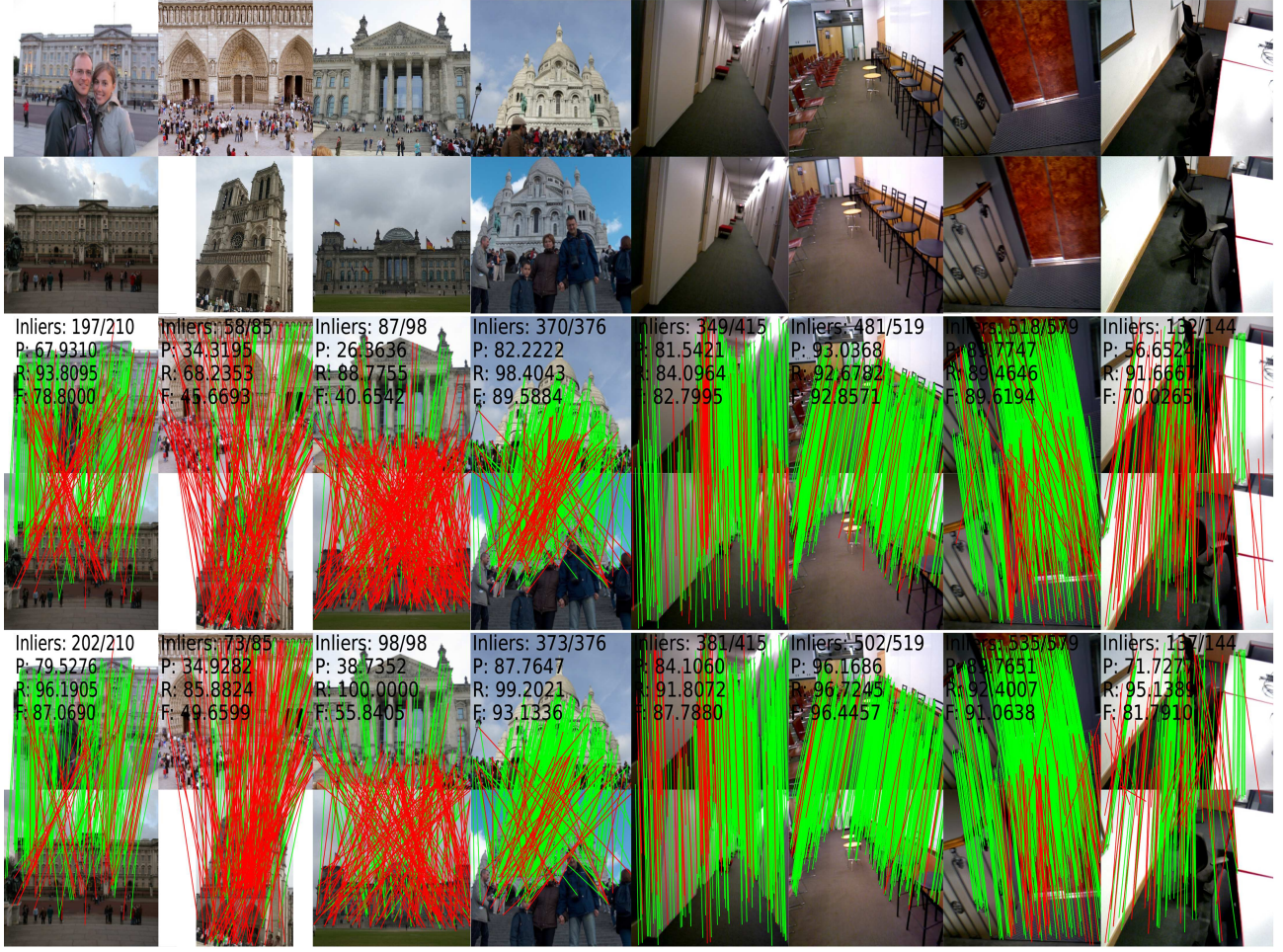
The accuracy of pose estimation can significantly reflect the performance of correspondence pruning methods. Therefore, we evaluate the proposed LeCoT and other networks on indoor and outdoor datasets preprocessed with SIFT and SuperPoint to assess their performance.

**Evaluation protocols.** We choose the weighted eight-point algorithm to estimate an essential matrix based on the predicted true correspondences, which is decomposed into a rotation vector and a translation vector. Subsequently, error metrics refer to the angular differences between the calculated rotation/translation vectors and labels. We select the mean average precision for angular errors less than or equal to 5° and less than or equal to 20° as the default metrics to evaluate the network in relative pose estimation, denoted as mAP5° and mAP20°.

**Outdoor relative pose estimation results.** The quantitative comparative experimental results of the proposed LeCoT and other state-of-the-art networks for relative pose estimation in outdoor scenes are summarized in Table 2. We choose mAP5° and mAP20° as evaluation metrics, and our LeCoT performs best on both of them. Specifically, LeCoT improves 1.54% on mAP5° than GCT (the second best network), but the number of parameters is less than half of that. Compared to MS<sup>2</sup>DG-Net, which has a similar network parameter quantity to ours, LeCoT has about 16 percentage points higher on mAP5°. As shown in the first four columns of Figure 5, our LeCoT can remove masses of outliers, while MS<sup>2</sup>DG-Net cannot. From this, it can be seen that the proposed LeCoT can perform well in relative pose estimation under the outdoor scenes, as it can naturally capture global context information from sparse correspondences without additional modules or data compression limitations.

**Indoor relative pose estimation results.** Additionally, we also compare these networks in indoor scenes. Due to a lack of texture information in indoor scenes, as shown in the right of Figure 5, it is more difficult to estimate relative pose. Even so, our LeCoT still performs best, as shown in Table 2. Clearly, LeCoT is 24.89% better than MS<sup>2</sup>DG-Net on mAP5°. In conclusion, the proposed LeCoT obtains superior performance in the relative pose





**Figure 5** (Color online) Partial typical visualization results on YFCC100M and SUN3D datasets with SIFT. From top to bottom: input image pairs, results of MS²DG-Net and the proposed LeCoT. Green and red lines are inliers and outliers, respectively. The top left corner of each image displays *Inliers*, *P*, *R*, and *F*.

**Table 2** Evaluation on YFCC100M and SUN3D with SIFT for relative pose estimation. mAP5° and mAP20° are reported, and the best result in each column is in bold.

Method	References	Size (MB)	YFCC100M (%)		SUN3D (%)	
			mAP5°	mAP20°	mAP5°	mAP20°
CNe [12]	CVPR2018	<b>0.39</b>	23.95	52.44	9.30	25.01
OA-Net++ [14]	ICCV2019	2.47	39.33	66.93	16.18	32.40
ACNe [15]	CVPR2020	0.41	33.06	62.91	18.86	36.04
LMC-Net [17]	CVPR2021	0.93	47.50	73.07	16.82	34.05
CL-Net [18]	ICCV2021	1.27	51.80	75.76	17.03	34.16
MS²DG-Net [19]	CVPR2022	2.61	49.13	76.04	17.84	36.21
ConvMatch [13]	AAAI2023	7.49	57.03	78.74	21.65	47.04
LFC [40]	ACM MM 2023	1.73	56.15	79.13	19.78	46.41
GCT [37]	AAAI 2024	4.09	63.80	82.86	21.05	46.75
DeMatch [41]	CVPR 2024	5.86	59.50	79.93	22.09	47.01
LeCoT (Ours)	—	3.74	<b>65.34</b>	<b>82.71</b>	<b>22.28</b>	<b>47.05</b>

estimation task under outdoor and indoor scenes.

**Generalization ability test.** Existing two-view correspondence pruning networks often exhibit poor generalization performance to different descriptors and scenes. For instance, models trained on datasets with SIFT struggle to adapt to the datasets preprocessed with SuperPoint, or models performing well in outdoor scenes may fail in indoor scenes. However, the proposed LeCoT, leveraging a full Transformer backbone, can effortlessly capture global context information, thereby mitigating the loss of generalization ability. Specifically, all models are

**Table 3** Generalization ability test on YFCC100M, SUN3D and PhotoTourism with different feature extractors, including ORB, SuperPoint and SIFT. mAP5° is reported. The best result in each column is in bold.

Method	Size (MB)	YFCC100M (%)			SUN3D (%)		PhotoTourism (%)	
		SIFT	ORB	SuperPoint	SIFT	SuperPoint	SIFT	SuperPoint
CNe [12]	<b>0.39</b>	47.98	10.28	38.33	12.01	13.24	34.35	28.73
OA-Net++ [14]	2.47	52.18	14.03	40.25	14.13	13.52	42.30	29.85
CL-Net [18]	1.27	58.95	16.80	39.98	14.48	11.81	45.92	28.94
MS <sup>2</sup> DG-Net [19]	2.61	57.68	14.63	41.68	15.16	14.11	46.62	32.94
ConvMatch [13]	7.49	58.95	15.65	45.07	15.31	14.23	48.59	33.41
LFC [40]	1.73	58.63	16.60	40.53	15.32	14.27	47.59	32.88
GCT [37]	4.09	62.70	19.40	37.50	15.73	11.06	49.05	27.26
DeMatch [41]	5.86	60.48	16.60	44.35	14.34	13.74	47.15	32.33
LeCoT (Ours)	3.74	<b>64.8</b>	<b>20.45</b>	<b>45.18</b>	<b>16.35</b>	<b>14.43</b>	<b>49.85</b>	<b>34.37</b>

**Table 4** Efficiency evaluation. mAP5° and the average runtime (ART, ms) of each image pair on YFCC100M with SIFT of different networks are reported. The best result on each row is in bold.

	RANSAC [6]	ConvMatch [13]	LFC [40]	GCT [37]	DeMatch [41]	LeCoT
mAP5° (↑)	9.08	57.03	56.28	63.80	59.50	<b>65.34</b>
ART (ms↓)	353.10	92.78	125.48	117.47	99.22	<b>74.32</b>

pre-trained on YFCC100M with SIFT and are directly evaluated on other datasets (e.g., YFCC100M, SUN3D and PhotoTourism [55]) with different combinations of feature extractors (e.g., SIFT, SuperPoint and ORB [56]). We use RANSAC as a post-processing step. In detail, PhotoTourism is an additional dataset, which includes a large number of photos taken by tourists during their travels, and ORB is a fast feature extractor. As shown in Table 3, we find that our LeCoT ranks first in all cases. This is because the proposed LeCoT can obtain adequate spatial and channel global context information, so that it has strong robustness and generalization abilities.

**Efficiency.** We summarize the experimental results of traditional RANSAC and several deep learning-based networks in Table 4, in which mAP5° and average runtime (ART, ms) are reported. Notably, these selected deep learning-based networks all achieve over 55 in mAP5° on YFCC100M with SIFT. The proposed LeCoT achieves the best performance and efficiency. Specifically, compared to RANSAC, the performance of our network is 719.60% of RANSAC, while the ART is only 21.05% of RANSAC. This demonstrates that the proposed LeCoT is both effective and efficient.

### 4.3 Homography estimation

Homography estimation is a critical task in computer vision, aiming to infer the perspective transformation between images. Typically, it involves calculating a homography matrix that describes this transformation relationship by analyzing corresponding points in images. We conduct homography estimation experiments using the direct linear transform (DLT) estimator on the HPatches benchmark [57].

**Dataset.** HPatches benchmark [57] comprises a total of 116 scenes, with 59 scenes exhibiting significant viewpoint changes and 57 scenes captured under varying illumination conditions. Each scene consists of 6 images, one serving as the reference image and the others as target images with ground-truth, amounting to 696 images in total, each image pair of which is detected up to 4000 keypoints with SIFT followed by a NN matcher.

**Baselines.** Eight learning-based networks (CNe [12], OA-Net++ [14], CL-Net [18], MS<sup>2</sup>DG-Net [19], ConvMatch [13], LFC [40], GCT [37], and DeMatch [41]) are chosen as baselines.

**Evaluation protocols.** We follow the recommendation of SuperPoint [5], using homography error to classify the estimation results as accurate or inaccurate, and thresholds are set at 3/5/10 pixels, respectively.

**Results.** Comparison results in Table 5 indicate that the proposed LeCoT, utilizing the Transformer as the network backbone, outperforms in all scenes. Particularly, LeCoT achieves an improvement of 3.84 points over the second best network (DeMatch) on ACC.3px. This is attributed to the fact that the proposed LeCoT can obtain spatial and channel global context information among sparse correspondences, which proves advantageous for homography estimation even in challenging scenarios.

### 4.4 Visual localization

Visual localization refers to estimating 6-degree of freedom (DOF) relative pose (3-DOF for rotation and another 3-DOF for translation) from given images to a 3D model. It has many important practical applications, such as

**Table 5** Evaluation of homography estimation on HPatches. Accuracy (ACC.) at different error thresholds is reported. The best result in each column is in bold.

Method	Size (MB)	HPatches (%)		
		ACC.3px	ACC.5px	ACC.10px
CNe [12]	<b>0.39</b>	38.97	51.55	65.34
OA-Net++ [14]	2.47	39.83	52.76	62.93
CL-Net [18]	1.27	43.10	55.69	68.10
MS <sup>2</sup> DG-Net [19]	2.61	41.21	50.17	62.59
ConvMatch [13]	7.49	43.73	55.78	66.12
LFC [40]	1.73	44.81	56.83	68.01
GCT [37]	4.09	47.58	57.96	68.25
DeMatch [41]	5.86	48.09	58.95	68.21
LeCoT (Ours)	3.74	<b>51.93</b>	<b>60.34</b>	<b>68.59</b>

**Table 6** Evaluation visual localization on Aachen Day-Night. We report the percentage of correctly localized queries (0.25 m, 2°), (0.5 m, 5°) and (1.0 m, 10°), respectively. The best result in each column is in bold.

Method	Size (MB)	Day	Night
		(0.25 m, 2°)/(0.5 m, 5°)/(1.0 m, 10°)	
CNe [12]	<b>0.39</b>	81.3/91.4/95.9	68.4/78.6/87.8
OA-Net++ [14]	2.47	82.3/91.9/96.5	71.4/79.6/90.8
CL-Net [18]	1.27	83.3/92.4/ <b>97.0</b>	71.4/80.6/ <b>93.9</b>
MS <sup>2</sup> DG-Net [19]	2.61	82.8/92.1/96.8	70.4/82.7/ <b>93.9</b>
ConvMatch [13]	7.49	81.8/91.8/96.6	69.4/79.7/89.9
LFC [40]	1.73	82.9/92.0/96.2	70.8/81.3/91.3
GCT [37]	4.09	83.2/92.1/96.5	71.0/81.4/91.6
DeMatch [41]	5.86	83.6/ <b>92.5</b> /96.7	71.1/81.9/92.3
LeCoT (Ours)	3.74	<b>83.7/92.5/97.0</b>	<b>71.6/82.8/93.9</b>

robot grasping and control, automatic navigation, and augmented reality. The proposed LeCoT and other networks are integrated into the official HLoc framework [1] to accomplish visual localization.

**Dataset.** We choose Aachen Day-Night benchmark [58] to test network performance. There are 922 query images (824 daytime and 98 nighttime) obtained by mobile phones and 4328 reference ones with the ancient European town style.

**Baselines.** We choose the same baselines as the homography estimation task in Subsection 4.3. All the networks are based on the NN strategy.

**Evaluation protocols.** Each image is extracted up to 4096 keypoints with SIFT and followed by an NN strategy. Then, a new SfM model is triangulated from the given poses and image pairs are found by image retrieval. Finally, we will perform visual localization. We choose the percentage of correctly localized queries at specific distances and orientation thresholds as evaluation matrices.

**Results.** As shown in Table 6, our LeCoT performs best in daytime and nighttime scenes with all the thresholds, especially the lowest threshold, which can demonstrate that LeCoT can work in visual localization.

#### 4.5 3D reconstruction

3D reconstruction refers to the process of reconstructing the three-dimensional shape and structure of an object or scene from two-dimensional data, such as images or videos, which is widely used in fields such as virtual reality (VR), augmented reality (AR), autonomous driving, and robot navigation.

**Dataset.** We select Alamo [59] as the test dataset, which contains 2915 images captured from the Alamo Mission in San Antonio, Texas. These images have been taken from multiple different viewpoints, aiming to capture the details of the building as well as the structure of the surrounding environment.

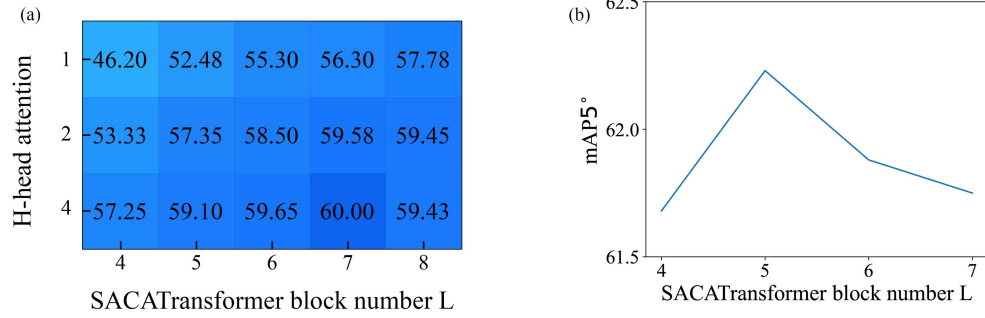
**Baselines.** We choose ratio test + mutual check (R+M), OA-Net+++ [60] and MS<sup>2</sup>DG-Net [19] as baselines. All the networks are trained on YFCC100M with SIFT.

**Evaluation protocols.** Following OA-Net+++ [60], we choose the following evaluation metrics: (1) the number of successfully registered/aligned images (Reg.), (2) the number of points in the sparse 3D point cloud (Sparse), (3) the number of points in the dense 3D point cloud (Dense), (4) the average track length (TL), (5) the average number of observations per 3D point (Obs.), and (6) the reprojection error (Reproj.). The higher the values of metrics (1)–(5), the better the network performance, and the opposite holds for the rest.



**Table 7** Comparative results of 3D reconstruction on Alamo. R+M: ratio test + mutual check. The best result in each column is in bold. The arrows ( $\uparrow/\downarrow$ ) denote that higher/lower values indicate better performance, respectively.

Method	Images	Reg. ( $\uparrow$ )	Sparse ( $\uparrow$ )	Dense ( $\uparrow$ )	TL ( $\uparrow$ )	Obs. ( $\uparrow$ )	Reproj. ( $\downarrow$ )
R+M		921	192714	3515294	<b>11.22</b>	2347	<b>0.63</b> <i>px</i>
OA-Net+++ [60]	2915	888	251375	3164383	10.59	2997	0.77 <i>px</i>
MS <sup>2</sup> DG-Net [19]		921	250328	3258545	10.46	2842	0.77 <i>px</i>
LeCoT		<b>996</b>	<b>318021</b>	<b>3806193</b>	10.76	<b>2999</b>	0.73 <i>px</i>

**Figure 6** (Color online) (a) Relationship between the number of SACATransformer blocks and the number of attention heads.  $mAP5^\circ$  (%) is reported. (b) Relationship between  $mAP5^\circ$  (%) and the number of SACATransformer blocks, when  $H = 4$  and  $po = 2$ .  $po = i$  represents the guiding probability set obtained at the  $i$ -th SACATransformer block.**Table 8** Evaluation on outdoor scenes with SIFT for relative pose estimation.  $mAP5^\circ$  and  $mAP20^\circ$  are reported.  $po = i$  represents the guiding probability set obtained at the  $i$ -th Transformer block. The best result in each row is in bold.

	$L = 7, H = 4$		$L = 5, H = 4$			
	–	–	$po = 1$	$po = 2$	$po = 3$	$po = 4$
$mAP5^\circ$ (%)	60.00	59.10	60.05	<b>62.23</b>	61.25	59.33
$mAP20^\circ$ (%)	80.61	80.37	81.02	<b>81.31</b>	80.61	80.56

**Results.** Following [60], we conduct 3D reconstruction experiments on the COLMAP platform [61]<sup>1)</sup>. For each image, we first extract up to 4000 feature points using SIFT descriptors, then apply various models trained on the YFCC100M dataset to filter out outliers. These filtered points are then integrated into the official COLMAP pipeline to evaluate the performance of these different models. As shown in Table 7, the proposed LeCoT achieves the best performance in terms of Reg., Sparse, Dense and Obs., and also demonstrates competitive results in the TL and Reproj., which can prove the generalizability of the proposed LeCoT.

#### 4.6 Ablation studies

In this subsection, ablation studies about the LeCoT structure and its details on the outdoor dataset with SIFT are reported.

**Relationship between  $H$  and  $L$ .** The relationship between the SACATransformer block number  $L$  and attention head number  $H$  has been discussed in Figure 6(a). When  $H = 1$  and  $L$  increases from 4 to 8, network performance gradually increases. Meanwhile, we find that when  $L = 4$  and  $H$  goes from 1 to 4, the network performance also gradually improves. In Figure 6(a), the color of the squares transitions from light to dark, representing an improvement in the network performance from poor to good. We find that when  $L = 7$  and  $H = 4$ , the network performs best, rather than simply choosing  $L = 8$  and  $H = 4$ . Hence, we choose this combination ( $L = 7$  and  $H = 4$ ).

**How can additional guiding information help?** To obtain stronger global context information at different stages, we additionally introduce guiding information to guide network training. If the distance between the guiding information and Sorting&Pruning Block is too close, the guiding information and the pruning module information  $\hat{P}_i$  may be too similar. While the distance is too far, the guiding information is inaccurate. As shown in Table 8, we find that  $po = 2$  is a suitable position to obtain guiding information. Observing Figure 6(b), we can find that when  $po = 2$  and  $H = 4$ , the model with  $L = 5$  performs best.

**Is transformer structure necessary?** In the first two rows of Table 9, there is an interesting phenomenon that if we only borrow attentions from the vanilla Transformer block, network performance will rapidly decrease. This

1) <https://github.com/colmap/colmap>.

**Table 9** Ablation experiments about spatial-channel fusion transformer design. LeCoT (Attention) represents only using attention instead of the full Transformer structure. LeCoT (Vanilla) presents using the vanilla Transformer. S-MSA: spatial multi-head self-attention; C-MSA: channel multi-head self-attention. The best result in each column is in bold.

	mAP5° (%)	mAP20° (%)
LeCoT (Attention)	39.88	60.98
LeCoT (Vanilla)	62.23	81.31
LeCoT (C-MSA + S-MSA)	62.30	81.53
LeCoT (C-MSA + C-MSA)	63.50	82.32
LeCoT (S-MSA + S-MSA)	63.85	82.38
LeCoT (Ours, S-MSA + C-MSA)	<b>65.34</b>	<b>82.71</b>

**Table 10** Ablation experiments about the prediction block. LeCoT (simple projection) employs the common strategy by directly applying a projection function (Conv( $\cdot$ )) to reduce the channel dimension from  $d$  to 1 to identify inliers, while LeCoT (Ours, prediction block) adopts the proposed prediction block. The best result in each column is in bold.

	mAP5° (%)	mAP20° (%)
LeCoT (simple projection)	64.05	81.98
LeCoT (Ours, prediction block)	<b>65.34</b>	<b>82.71</b>

is very surprising, probably because there are no MLPs or CNNs for feature extraction. However, if the complete Transformer structure is used, the issue will disappear.

**Spatial-channel fusion transformer design.** Comparing with the third to sixth rows in Table 9, we find that utilizing S-MSA first and then C-MSA is much better than utilizing C-MSA first, utilizing C-MSA twice, and utilizing S-MSA twice. Moreover, all these four combinations outperform the vanilla Transformer. This is because the proposed SACATransformer can simultaneously consider the spatial and channel similarities between correspondence features, while the vanilla Transformer cannot.

**Prediction block vs. simple projection.** In Table 10, we present the performance of the proposed prediction block and a simple projection function (Conv( $\cdot$ )), such as in OA-Net++ [14]. It can be found that the network using the proposed prediction block outperforms significantly. This is because directly mapping features from a high dimension  $d$  to 1 may result in information loss [23]. In contrast, the proposed prediction block, as shown in Figure 4, gradually reduces the dimension, thus mitigating this issue to a certain extent.

## 5 Conclusion

In this work, from a new perspective, we propose a simple yet effective two-view correspondence pruning network LeCoT, without additional module designs and data compression, which can naturally capture global context information at various stages. In particular, we propose a spatial-channel fusion transformer block as the backbone to simultaneously capture spatial and channel global context information. Additionally, we utilize the proposed prediction block to capture a probability set from the middle stage, as guiding information, which can make the proposed LeCoT to more effectively acquire global context information at different stages. Extensive experiments demonstrate that the proposed LeCoT outperforms existing state-of-the-art networks. Using Transformers as a backbone will be a new perspective in correspondence pruning, and we will look further into this in our subsequent work.

**Acknowledgements** This work was supported by National Natural Science Foundation of China (Grant No. 62172226). Luanyuan DAI acknowledges the support of the China Scholarship Council Program (Grant No. 202306840103).

## References

- 1 Sarlin P E, Cadena C, Siegwart R, et al. From coarse to fine: robust hierarchical localization at large scale. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019. 12716–12725
- 2 Mur-Artal R, Montiel J M M, Tardos J D. ORB-SLAM: a versatile and accurate monocular SLAM system. IEEE Trans Robot, 2015, 31: 1147–1163
- 3 Bai X Y, Luo Z X, Zhou L, et al. Pointdsc: robust point cloud registration using deep spatial consistency. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021. 15859–15869
- 4 Lowe D G. Distinctive image features from scale-invariant keypoints. Int J Comput Vision, 2004, 60: 91–110
- 5 DeTone D, Malisiewicz T, Rabinovich A. SuperPoint: self-supervised interest point detection and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018. 224–236
- 6 Fischler M A, Bolles R C. Random sample consensus. Commun ACM, 1981, 24: 381–395



- 7 Ma J, Zhao J, Tian J, et al. Robust point matching via vector field consensus. *IEEE Trans Image Process*, 2014, 23: 1706–1721
- 8 Ma J, Zhao J, Jiang J, et al. Locality preserving matching. *Int J Comput Vis*, 2019, 127: 512–531
- 9 Liu Y, Li Y, Dai L, et al. Robust feature matching via advanced neighborhood topology consensus. *Neurocomputing*, 2021, 421: 273–284
- 10 Liu Y Z, Zhao B N, Zhao S J, et al. Progressive motion coherence for remote sensing image matching. *IEEE Trans Geosci Remote Sensing*, 2022, 60: 1–13
- 11 Liu Y Z, Zhou W W, Li Y P, et al. ROSE: rotation-invariant sequence-aware consensus for robust correspondence pruning. In: *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024. 7754–7763
- 12 Yi K M, Trulls E, Ono Y, et al. Learning to find good correspondences. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2666–2674
- 13 Zhang S, Ma J. ConvMatch: rethinking network design for two-view correspondence learning. *AAAI*, 2023, 37: 3472–3479
- 14 Zhang J H, Sun D W, Luo Z X, et al. Learning two-view correspondences and geometry using order-aware network. In: *Proceedings of the IEEE International Conference on Computer Vision*, 2019. 5845–5854
- 15 Sun W W, Jiang W, Trulls E, et al. ACNe: attentive context normalization for robust permutation-equivariant learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 11286–11295
- 16 Dai L, Liu X, Liu Y, et al. Enhancing two-view correspondence learning by local-global self-attention. *Neurocomputing*, 2021, 459: 176–187
- 17 Liu Y, Liu L J, Lin C, et al. Learnable motion coherence for correspondence pruning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 3237–3246
- 18 Zhao C, Ge Y X, Zhu F, et al. Progressive correspondence pruning by consensus learning. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 6464–6473
- 19 Dai L Y, Liu Y Z, Ma J Y, et al. MS2DG-Net: progressive correspondence learning via multiple sparse semantics dynamic graph. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 8973–8982
- 20 Ye X, Zhao W, Lu H, et al. Learning second-order attentive context for efficient correspondence pruning. *AAAI*, 2023, 37: 3250–3258
- 21 Li Z Z, Zhang S H, Ma J Y. U-Match: two-view correspondence learning with hierarchy-aware local context aggregation. In: *Proceedings of International Joint Conference on Artificial Intelligence*, 2023. 1169–1176
- 22 Thomee B, Shamma D A, Friedland G, et al. YFCC100M: the new data in multimedia research. *Commun ACM*, 2016, 59: 64–73
- 23 Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks. *Science*, 2006, 313: 504–507
- 24 Bengio Y, Courville A, Vincent P. Representation learning: a review and new perspectives. *IEEE Trans Pattern Anal Mach Intell*, 2013, 35: 1798–1828
- 25 LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*, 2015, 521: 436–444
- 26 Zhang Q S, Wu Y N, Zhu S C. Interpretable convolutional neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 8827–8836
- 27 Zeiler M D, Fergus R. Visualizing and understanding convolutional networks. In: *Proceedings of European Conference on Computer Vision*, 2014. 818–833
- 28 Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16×16 words: transformers for image recognition at scale. 2020. [ArXiv:2010.11929](https://arxiv.org/abs/2010.11929)
- 29 Strudel R, Garcia R, Laptev I, et al. Segmenter: transformer for semantic segmentation. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 7262–7272
- 30 Wang W, Yang X, Tang J. Vision transformer with hybrid shifted windows for gastrointestinal endoscopy image classification. *IEEE Trans Circuits Syst Video Technol*, 2023, 33: 4452–4461
- 31 Qi C R, Su H, Mo K C, et al. PointNet: deep learning on point sets for 3D classification and segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 652–660
- 32 Liao T, Zhang X, Xu Y, et al. SGA-Net: a sparse graph attention network for two-view correspondence learning. *IEEE Trans Circuits Syst Video Technol*, 2023, 33: 7578–7590
- 33 Li Z Z, Jiang J, Ma J Y. Seed to prune: a seeded graph neural network for two-view correspondence learning. *IEEE Trans Neural Networks Learn Syst*, 2024, 36: 12199–12213
- 34 Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: *Proceedings of Advances in Neural Information Processing Systems*, 2017. 5998–6008
- 35 Dai L Y, Du X Y, Zhang H W, et al. MGNet: learning correspondences via multiple graphs. *AAAI*, 2024, 38: 3945–3953
- 36 Miao X, Xiao G, Wang S, et al. BCLNet: bilateral consensus learning for two-view correspondence pruning. *AAAI*, 2024, 38: 4225–4232
- 37 Guo J, Xiao G, Wang S, et al. Graph context transformation learning for progressive correspondence pruning. *AAAI*, 2024, 38: 1968–1975
- 38 Li Z, Zhang S, Ma J. U-Match: exploring hierarchy-aware local context for two-view correspondence learning. *IEEE Trans Pattern Anal Mach Intell*, 2024, 46: 10960–10977
- 39 Dai L Y, Du X Y, Tang J H. TRGA: reconsidering the application of graph neural networks in two-view correspondence pruning. In: *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024. 5633–5642
- 40 Wang L B, Wu J, Fang X Y, et al. Local consensus enhanced Siamese network with reciprocal loss for two-view correspondence learning. In: *Proceedings of the 31st ACM International Conference on Multimedia*, 2023. 5235–5243
- 41 Zhang S H, Li Z Z, Gao Y, et al. DeMatch: deep decomposition of motion field for two-view correspondence learning. In: *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024. 20278–20287

- 42 Deng J, Dong W, Socher R, et al. ImageNet: a large-scale hierarchical image database. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2009. 248–255
- 43 Touvron H, Cord M, Sablayrolles A, et al. Going deeper with image transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021. 32–42
- 44 Chen J N, Lu Y Y, Yu Q H, et al. TransUNet: transformers make strong encoders for medical image segmentation. 2021. ArXiv:2102.04306
- 45 Chen H X, Li H X, Li Y H, et al. Sparse spatial transformers for few-shot learning. *Sci China Inf Sci*, 2023, 66: 210102
- 46 Liang D K, Chen X W, Xu W, et al. TransCrowd: weakly-supervised crowd counting with transformers. *Sci China Inf Sci*, 2022, 65: 160104
- 47 Wang W H, Xie E Z, Li X, et al. Pyramid vision transformer: a versatile backbone for dense prediction without convolutions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021. 568–578
- 48 Yuan L, Chen Y P, Wang T, et al. Tokens-to-token ViT: training vision transformers from scratch on ImageNet. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021. 558–567
- 49 Carion N, Massa F, Synnaeve G, et al. End-to-end object detection with transformers. In: Proceedings of European Conference on Computer Vision, 2020. 213–229
- 50 Liu Z, Lin Y T, Cao Y, et al. Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021. 10012–10022
- 51 Devlin J, Chang M W, Lee K, et al. BERT: pre-training of deep bidirectional transformers for language understanding. 2018. ArXiv:1810.04805
- 52 He K M, Chen X L, Xie S N, et al. Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022. 16000–16009
- 53 Kim W J, Son B K, Kim I D. ViLT: vision-and-language transformer without convolution or region supervision. In: Proceedings of International Conference on Machine Learning, 2021. 5583–5594
- 54 Xiao J X, Owens A, Torralba A. SUN3D: a database of big spaces reconstructed using SfM and object labels. In: Proceedings of the IEEE International Conference on Computer Vision, 2013. 1625–1632
- 55 Jin Y, Mishkin D, Mishchuk A, et al. Image matching across wide baselines: from paper to practice. *Int J Comput Vis*, 2021, 129: 517–547
- 56 Rublee E, Rabaud V, Konolige K, et al. ORB: an efficient alternative to SIFT or SURF. In: Proceedings of International Conference on Computer Vision, 2011. 2564–2571
- 57 Balntas V, Lenc K, Vedaldi A, et al. Hpatches: a benchmark and evaluation of handcrafted and learned local descriptors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017. 5173–5182
- 58 Sattler T, Maddern W, Toft C, et al. Benchmarking 6DoF outdoor visual localization in changing conditions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018. 8601–8610
- 59 Wilson K, Snavely N. Robust global translations with 1DSfM. In: Proceedings of European Conference on Computer Vision, 2014. 61–75
- 60 Zhang J, Sun D, Luo Z, et al. OANet: learning two-view correspondences and geometry using order-aware network. *IEEE Trans Pattern Anal Mach Intell*, 2022, 44: 3110–3122
- 61 Schonberger J L, Frahm J M. Structure-from-motion revisited. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016. 4104–4113