# LoFTR: Detector-Free Local Feature Matching with Transformers

Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, Xiaowei Zhou,
Zhejiang University, SenseTime Research

**摘要**

本文提出了一种新颖的图像局部特征匹配方法，不是按传统的步骤——图像特征检测、描述和匹配来顺序执行，而是首先在粗粒度上建立逐像素的密集匹配，然后在细粒度上完善精细匹配。与使用 cost volume 搜索对应关系的稠密匹配方法相比，本文使用的是 Transformer 的自注意力和交叉注意力层 (self and cross attention layers) 来获取两幅图像的特征描述符。基于特征检测器的方法在图像弱纹理区域通常难以产生可重复的兴趣点，而 Transformer 提供的全局感受野使图像能够在弱纹理区域产生密集匹配。本人使用 LoFTR 作为上游任务提取图像特征点，然后来完成图像拼接的下游任务，并与传统的基于 SIFT 以及 ORB 的图像拼接方法进行了对比实验。

**关键词：** 局部特征匹配；Transformer；图像拼接

## 1 引言

局部特征匹配在目标识别、图像配准、视觉跟踪、三维重建等方面都有广泛的应用。传统的局部特征匹配方法需要经过三个步骤，分别是图像特征检测、特征描述以及特征匹配。但是这种传统的局部特征匹配方法在低纹理、模式重复、视点变化、照明变化、运动模糊等情况可能无法在图像之间提取出足够的可重复的兴趣点，这种问题在室内的环境下尤为突出。[8, 12, 13] 提出了通过建立像素级密集匹配来解决这个问题，可以从密集匹配中选择置信度高的匹配，从而避免特征检测。但是在这些工作中，卷积神经网络提取的密集特征具有有限的接受野，可能无法区分不明显区域。

不明显的区域的对应关系不仅基于局部领域，而且基于更大全局背景，比如一些低纹理区域可以根据它们相对于边缘的相对位置来区分。基于上述观察，本文提出了一种新的无检测器的局部特征匹配方法——Local Feature TRansformer (LoFTR)。受开创性工作 SuperGlue [15] 的启发，本文使用 Transformer [17] 的自注意力和交叉注意力层一起处理 (变换) 从卷积 backbone 中提取的密集局部特征。以低特征分辨率 (图像维数的 1/8) 在两组变换后的特征之间提取密集匹配，并从这些密集匹配中选择高置信度的匹配，使用基于相关性的方法将其细化到亚像素级别。Transformer 的全局接受域和位置编码使变换后的特征表示具有上下文和位置依赖性。通过多次交错自注意力和交叉注意层，LoFTR 能够学习到在 ground-truth 匹配中显示的密集分布的全局一致匹配先验。本文还使用了线性 Transformer，降低了计算复杂度。

本文使用室内和室外数据集，在多个图像匹配和相机姿态估计任务中评估了所提出的方法。实验表明，LoFTR 在很大程度上优于基于检测器和无检测器的特征匹配基线。与基于检测器的基线方法相比，即使在低纹理、运动模糊或重复模式的无特征区域，LoFTR 也能产生高质量的匹配。

## 2 相关工作

### 2.1 基于检测器的局部特征匹配

基于检测器的方法是局部特征匹配的主要方法，在深度学习时代之前，许多工作在传统手工设计的局部特征上都取得了良好的表现。SIFT [11] 和 ORB [14] 是最成功的手工设计的局部特征，被广泛应用在许多 3D 计算机视觉任务中。基于学习的方法可以显著提高局部特征在更大视角和光照变化下的性能。LIFT [19] 和 MagicPoint [3] 是首批成功的基于学习的局部特征。它们采用的检测器设计是基于手工设计的方法，并取得了不错的性能。SuperPoint [4] 在 MagicPoint 的基础上，提出了一种通过自适应的自监督训练方法。

上述局部特征都是使用最近邻搜索来查找提取的兴趣点之间的匹配。SuperGlue [15] 创新性地提出了一种基于学习的局部特征匹配方法。SuperGlue 接受两组兴趣点及其描述符作为输入，并使用图神经网络 (GNN) 学习它们的匹配，这是 Transformer 的一般形式。由于可以通过数据驱动的方法学习特征匹配的先验知识，SuperGlue 取得了很好的效果，并开创了局部特征匹配的新技术。然而，作为一种依赖于检测器的方法，它还是具有无法在无特征区域中检测出可重复的兴趣点的根本缺点。本文工作受到 SuperGlue 的启发，在 GNN 中使用自注意力机制和交叉注意力机制在两组特征描述符之间传递消息，并且提出了一个无检测器的设计来避免特征检测器的根本缺点。

### 2.2 无检测器的局部特征匹配

无检测器的方法移除了特征检测阶段，直接产生稠密特征描述符或者稠密特征匹配。稠密特征匹配的思想可以追溯到 SIFT Flow [10]。与基于检测器的方法类似，最近邻搜索通常被用于稠密特征描述符匹配的后处理步骤。NCNet [13] 提出了一种不同的方法，以端对端的方式直接学习稠密匹配。它构建四维成本量来枚举图像之间所有可能的匹配，并使用四维卷积来正则化成本量并强制所有匹配之间的邻域一致性。稀疏 NCNet [12] 在 NCNet 的基础上进行了改进，使其在使用稀疏卷积时效率更高。与本文的工作相同，DRC-Net [8] 也遵循了这一思路，并提出了一种由粗到精的匹配方法，能够以更高的精度产生稠密匹配。

### 2.3 视觉任务中的 Transformer

由于 Transformer 的简单性和计算高效性，Transformer 已经成为 NLP 序列标注任务中的标准模块。近年来，Transformers 在图像分类 [5]、目标检测 [1]、语义分割 [18] 等任务中也取得了非常好的效果。由于需要将查询向量和键向量进行点乘，所以普通 Transformer 的计算量随着输入序列长度呈二次增长。最近在处理长序列问题上，提出了许多 Transformer 的有效变体 [2, 6, 7]。由于这些工作中没有对输入数据作前提假设，因此它们也非常适合处理图像。

# 3 本文方法

## 3.1 本文方法概述

本文提出了一个无检测器的局部特征匹配方法 LoFTR，方法总框架如图 1所示。本文方法可以分为四个模块，分别是局部特征提取模块、粗粒度局部特征 Transformer 模块、粗粒度特征匹配模块以及细粒度特征匹配模块。在局部特征提取模块中，使用 CNN 从输入的图像对中提取出粗粒度特征以及细粒度特征。然后在粗粒度局部特征 Transformer 模块中将粗粒度特征 flatten 为一维向量，并加入位置编码，并由 LoFTR 模块对其进行处理。接着在粗粒度特征匹配模块中利用可微匹配层对经过 LoFTR 模块变换后的特征进行匹配，得到置信度矩阵。根据置信度阈值和相互最近邻方法对置信矩阵中的匹配进行选择，得到粗粒度匹配预测。在细粒度特征匹配模块中，对于每个选定的粗粒度匹配预测，从细粒度特征中裁剪出一个局部窗口。粗粒度匹配预测将在这个局部窗口内被细化到亚像素级别，作为最终的匹配预测。

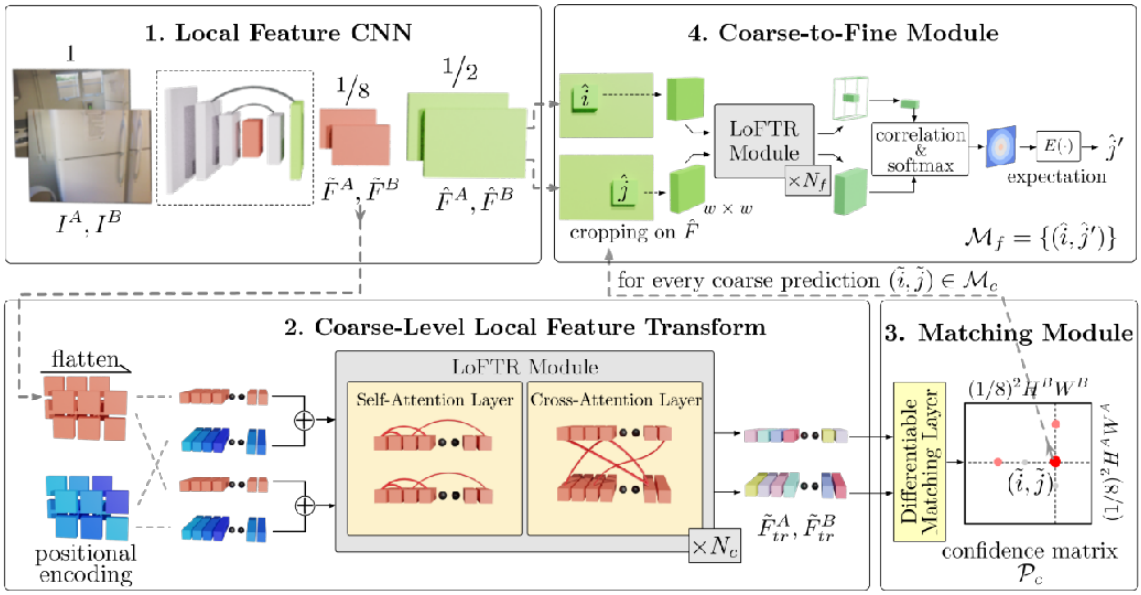



图 1. 方法框架图

## 3.2 局部特征提取模块

CNN 具有局部性和平移等变性归纳偏置，适合提取局部特征。在局部特征提取模块，首先使用 FPN [9] 架构提取输入图像对的多级特征。FPN 是依赖于通过自顶向下的路径和横向连接将低分辨率、语义强的特性与高分辨率、语义弱的特性结合起来的体系结构。本文使用原始图像维度的 1/8 提取粗粒度特征，使用原始图像维度 1/2 提取细粒度的特征。这样的降采样操作减少了后续的 LoFTR 模块的输入长度，降低了计算成本。

## 3.3 粗粒度局部特征 Transformer 模块

在 DETR [1] 之后，本文在 Transformer 中使用了标准位置编码的 2D 扩展。与 DETR 不同，本文只将它们添加到主干输出中一次。位置编码以正弦格式给出每个元素唯一的位置信

息。首先将从局部特征提取模块提取到的粗粒度特征 flatten 为向量，然后加上位置编码，因此变换后的特征将与位置变得相关，从而可以在不清晰区域中产生匹配。

LoFTR 模块由自注意力和交叉注意力层组成。对于自注意力层，输入特征是相同的，即同一张图片的特征；对于交叉注意力层，输入特征是不同的，即不同图片的特征。在 LoFTR 模块中，将自注意力层和交叉注意力层交错多次。它的核心思想其实就是对特征进行变换，融合本张图片的邻域信息以及融入待匹配图像的信息，最终得到比较容易匹配的特征。

### 3.4 粗粒度特征匹配模块

对于从上一模块中得到的比较容易匹配的特征，建立一个粗粒度特征匹配模块。可以应用两种类型的可微匹配层，分别是 optimal transport(OT) 层 [15] 和 dual-softmax operator [13, 16]。首先通过 $S(i,j) = \frac{1}{\tau}\langle \tilde{F}^A_{\text{tr}}(i), \tilde{F}^B_{\text{tr}}(j)\rangle$ 计算变换特征之间的得分矩阵。当使用 OT 进行匹配时，$-S$ 可以作为部分分配问题的成本矩阵。当使用 dual-softmax operator 时，匹配概率 $P_c$ 通过以下公式获得：

$$P_c(i,j) = \text{softmax}(S(i,\cdot))_j \cdot \text{softmax}(S(\cdot,j))_i \tag{1}$$

基于置信矩阵 $P_c$，选择置信度高于阈值 $\theta_c$ 的匹配，并进一步实施相互最近邻（MNN）准则，该准则能够过滤掉可能的离群粗粒度匹配。本文将粗粒度匹配预测表示为：

$$M_c = \left\{(\tilde{i},\tilde{j}) | \forall (\tilde{i},\tilde{j}) \in \text{MNN}(P_c), P_c(\tilde{i},\tilde{j}) \geq \theta_c\right\} \tag{2}$$

### 3.5 细粒度特征匹配模块

在建立粗粒度匹配后，需要将这些匹配细化到原始图像分辨率。首先使用基于相关性的方法，确定每个粗粒度匹配在细粒度特征图中的位置。然后在这个区域裁剪 $w \times w$ 大小的块。将 $w \times w$ 的块放入上述 LoFTR 模块中，得到新特征向量 $\hat{F}^A_{\text{tr}}(\hat{i})$ 和 $\hat{F}^B_{\text{tr}}(\hat{j})$，它们分别以点 $\hat{i}$ 和 $\hat{j}$ 为中心。计算 $\hat{F}^A_{\text{tr}}(\hat{i})$ 的中心向量和 $\hat{F}^B_{\text{tr}}(\hat{j})$ 每个向量的相关性并归一化，最后计算概率分布的期望，得到 B 图上与 $\hat{i}$ 点最匹配的点 $\hat{j}'$，表示为 $M_f = (\hat{i},\hat{j}')$。

### 3.6 损失函数定义

粗粒度级别的损失函数是 OT 层或 dual-softmax operator 返回的置信矩阵 $P_c$ 上的负对数似然损失。本文遵循 SuperGlue [15]，在训练期间使用相机姿势和深度图来计算置信矩阵的 ground-truth 标签。本文将 ground-truth 粗粒度匹配定义为两组 1/8 分辨率网格的相互最近邻，两个栅格之间的距离通过其中心位置的重投影距离来测量。对于 OT 层，本文使用与 [15] 中相同的损失函数。当使用 dual-softmax operator 进行匹配时，最小化网格的负对数似然损失：

$$L_c = -\frac{1}{|\ M_c^{gt}\ |} \sum_{(\tilde{i},\tilde{j}) \in M_c^{gt}} \log P_c(\tilde{i},\tilde{j}) \tag{3}$$

细粒度级别损失函数使用的是 $l_2$ 级细粒度损失。对于每个查询点 $\hat{i}$，本文通过计算相应 heatmap 的总方差 $\sigma^2(\hat{i})$ 来测量其不确定性。目标是优化具有低不确定性的精确位置，从而产生细粒度级别的最终加权损失函数：

$$L_f = \frac{1}{\mid M_f \mid} \sum_{(\hat{i},\hat{j}')\in M_f} \frac{1}{\sigma^2(\hat{i})} \| \hat{j}' - \hat{j}'_{gt} \|_2 \tag{4}$$

其中 $\hat{j}'_{gt}$ 是通过使用地面实况相机姿态和深度将每个 $\hat{i}$ 从 $\hat{F}^A_{tr}(\hat{i})$ 扭曲到 $\hat{F}^B_{tr}(\hat{j})$ 来计算的。计算 $L_f$ 时，如果 $\hat{i}$ 的扭曲位置落在 $\hat{F}^B_{tr}(\hat{j})$ 的局部窗口之外，则忽略 $(\hat{i},\hat{j}')$。在训练期间，梯度不会通过 $\sigma^2(\hat{i})$ 反向传播。

最终损失是粗粒度级别损失和细粒度级别损失之和。

$$L = L_c + L_f \tag{5}$$

# 4 复现细节

## 4.1 与已有开源代码对比

本次复现工作是在现有代码的基础上进行的，主要参考了现有代码的整体架构和网络模型。在现有代码的基础上，我将 LoFTR 作为上游任务提取图像特征点，来完成图像拼接的下游任务。并与传统的基于 SITF 以及 ORB 的图像拼接方法在指纹图像、室内图像以及室外图像上的拼接效果进行对比，实验表明，使用 LoFTR 提取特征点的方法得到了最佳的图像拼接效果。

## 4.2 实验环境搭建

实验平台环境为 Linux 系统，GPU 为 TITAN Xp 12GB 显存，CPU 为 16GB 6 核处理器。实验语言使用的是 Python3.7 版本，编码框架使用的是 Pytorch，版本为 1.10.1，cuda 版本为 11.2。

## 4.3 创新点

使用 LoFTR 作为上游任务提取特征点，完成图像拼接下游任务。即使是在低纹理、模式重复的区域，LoFTR 也能产生密集匹配，因此使用 LoFTR 提取出的特征进行图像拼接，能够产生比较理想的图像拼接效果。

# 5 实验结果分析

## 5.1 复现结果

对 LoFTR 的性能进行测试，分别在室内和室外数据集上的姿态估计任务使用 AUC 指标评估 LoFTR 性能，并与论文中所给出的数据进行对比，结果如表 1以及表 2所示。其中 OT 表示粗粒度匹配使用的是 optimal transport 层，DS 表示粗粒度匹配用的是 dual-softmax operator。

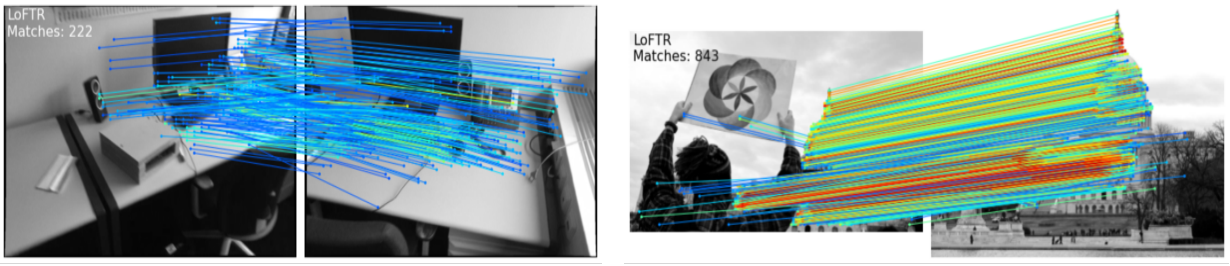
图 2是 LoFTR-DS 对图像进行局部特征匹配的结果，包括室内图像和室外图像，图中的连线的颜色越红表示匹配的可信度越高，越蓝表示匹配的可信度越低。可以看到，LoFTR 能够产生密集的、可信度较高的匹配。即使是在室内的低纹理区域也可以产生密集匹配。

表 1. 复现情况对比 (室内)

| Method | @5° | @10° | @20° |
|---|---|---|---|
| LoFTR-OT-原 | 21.51 | 40.39 | 57.96 |
| LOFTR-DS-原 | 22.06 | 40.8 | 57.62 |
| LoFTR-OT-复现 | 21.40 | 40.24 | 57.45 |
| LOFTR-DS-复现 | 22.15 | 40.78 | 57.76 |

表 2. 复现情况对比 (室外)

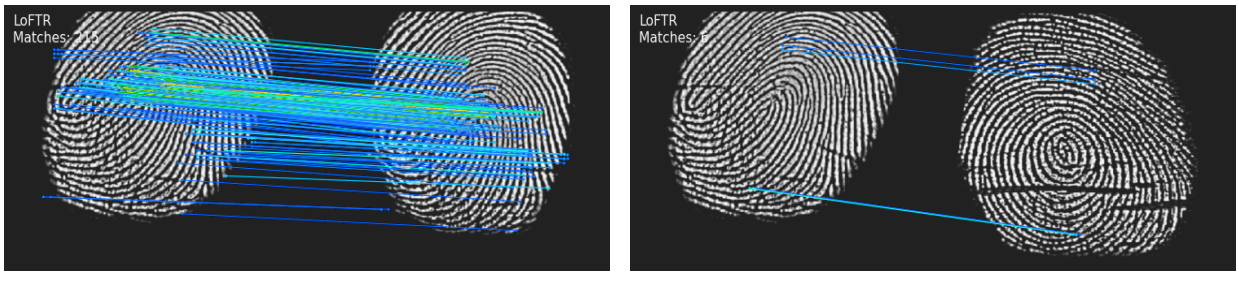
| Method | @5° | @10° | @20° |
|---|---|---|---|
| LoFTR-OT-原 | 50.31 | 67.14 | 79.93 |
| LOFTR-DS-原 | 52.8 | 69.19 | 81.18 |
| LoFTR-OT-复现 | 51.29 | 67.92 | 80.06 |
| LOFTR-DS-复现 | 52.8 | 69.19 | 81.19 |




(a) (b)

图 2. LoFTR 进行局部匹配: (a) 室内; (b) 室外

图 3是使用 LoFTR-DS 对传统 2D 指纹图像进行局部特征匹配的结果，其中图 3(a) 是来自同于一根手指的指纹图像的匹配结果，图 3(b) 是来自不同手指的指纹图像的匹配结果。通过图 3可知，即使在指纹图像这种模式重复的区域，LoFTR 也能够产生比较密集的匹配。并且对于来自不同手指的指纹图像，LoFTR 也不会错误地产生大量匹配。




(a) (b)

图 3. LoFTR 进行局部匹配: (a) 同一根手指的指纹图像; (b) 不同手指的指纹图像

### 5.2 图像拼接

分别使用基于 SIFT、基于 ORB 以及基于 LoFTR 的方法进行图像拼接，观察最终的效果。图 4是使用 LoFTR 提取指纹图像特征后进行拼接的效果。此外，通过计算拼接图像时的特征点筛选率来对比不同方法产生的特征点的质量好坏，如表 3所示。

(a) (b) (c)

图 4. 指纹图像拼接: (a) left 图; (b)right 图; (c) 拼接后

表 3. 拼接指纹图像的特征点筛选率对比

| Method | 匹配点数 | 筛选后 | 筛选率 (%) |
|---|---|---|---|
| SIFT | 800 | <4 | 无法拼接 |
| ORB | 310 | <4 | 无法拼接 |
| LoFTR | 1313 | 639 | 51.33 |

拼接时使用 RANSAC 算法来筛选出符合条件的匹配特征点，筛选率越低说明产生的匹配特征点的质量越好。由表 3可知，由于 SIFT 和 ORB 方法不能够产生足够的符合要求的特征点，因此拼接失败。而 LoFTR 能够产生大量的特征点，并且筛选后仍然剩余大量的符合要求的特征点，从而得到了不错的指纹图像拼接结果。

图 5和表 4是对室外图像进行拼接的结果。

表 4. 拼接室外图像的特征点筛选率对比

| Method | 匹配点数 | 筛选后 | 筛选率 (%) |
|---|---|---|---|
| SIFT | 801 | 39 | 95.13 |
| ORB | 293 | 38 | 87.03 |
| LoFTR | 1249 | 998 | 20.10 |

根据拼接的效果图 5可以知道，使用 LoFTR 提取特征点然后进行拼接时，原图产生了最小的偏移和扭曲，并且拼接线处相比于 SIFT 以及 ORB，LoFTR 的对齐效果最佳。根据表 4可知，LoFTR 产生了最多的质量高的特征点，具有最低的特征筛选率。

图 6和表 5是对室内图像进行拼接的结果。

与室外图像的拼接结果相同，使用 LoFTR 提取室内图像特征点后拼接产生了最佳的拼接效果以及最低的特征点筛选率。

(a) (b) (c)

(d) (e)

图 5. 室外图像拼接: (a) left 图; (b)right 图; (c)SIFT; (d)ORB; (e)LoFTR

(a) (b) (c)

(d) (e)

图 6. 室内图像拼接: (a) left 图; (b)right 图; (c)SIFT; (d)ORB; (e)LoFTR

表 5. 拼接室内图像的特征点筛选率对比

| Method | 匹配点数 | 筛选后 | 筛选率 (%) |
|---|---|---|---|
| SIFT | 422 | 28 | 93.36 |
| ORB | 231 | 24 | 89.61 |
| LoFTR | 857 | 757 | 11.67 |

## 6 总结与展望

本文提出了一种新的无检测器局部特征匹配方法 LoFTR，它能以粗到细的方式与 Transformer 建立精确的密集匹配。提出的 LoFTR 模块利用 Transformer 中的自注意力层和交叉注意力层将局部特征转换为和上下文以及位置相关的特征，使得 LoFTR 能够在低纹理或模式重复的无特征区域上获得高质量的匹配。实验表明，在多数据集上 LoFTR 在姿态估计和视觉定位方面取得了最好的性能。LoFTR 为局部图像特征匹配中的无检测器方法提供了新的方向，可以扩展到更具挑战性的场景。

复现工作中，对 LoFTR 方法进行了复现，并且将 LoFTR 作为上游任务提取图像特征点，运用到图像拼接这一下游任务中去，取得了不错的图像拼接效果。此外，可以学习 LoFTR 的由粗到细的匹配的思想，来完成更多的图像匹配任务。

## 参考文献


[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[2] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

[3] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Toward geometric deep slam. *arXiv preprint arXiv:1707.07410*, 2017.

[4] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 224–236, 2018.

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[6] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pages 5156–5165. PMLR, 2020.

[7] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*, 2020.

[8] Xinghui Li, Kai Han, Shuda Li, and Victor Prisacariu. Dual-resolution correspondence networks. *Advances in Neural Information Processing Systems*, 33:17346–17357, 2020.

[9] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[10] Ce Liu, Jenny Yuen, and Antonio Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):978–994, 2010.

[11] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110, 2004.

[12] Ignacio Rocco, Relja Arandjelović, and Josef Sivic. Efficient neighbourhood consensus networks via submanifold sparse convolutions. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, pages 605–621. Springer, 2020.

[13] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. *Advances in neural information processing systems*, 31, 2018.

[14] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee, 2011.

[15] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020.

[16] Michał Tyszkiewicz, Pascal Fua, and Eduard Trulls. Disk: Learning local features with policy gradient. *Advances in Neural Information Processing Systems*, 33:14254–14265, 2020.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[18] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *European conference on computer vision*, pages 108–126. Springer, 2020.

[19] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14*, pages 467–483. Springer, 2016.