Geometrized Transformer for Self-Supervised Homography Estimation

Jiazhen Liu and Xirong Li*

Key Lab of DEKE, Renmin University of China

{liujiazhen, xirong}@ruc.edu.cn

https://github.com/ruc-aimc-lab/GeoFormer

Abstract

*For homography estimation, we propose Geometrized Transformer (*GeoFormer*), a new detector-free feature matching method. Current detector-free methods,* e.g. *LoFTR, lack an effective mean to accurately localize small and thus computationally feasible regions for cross-attention diffusion. We resolve the challenge with an extremely simple idea: using the classical RANSAC geometry for attentive region search. Given coarse matches by LoFTR, a homography is obtained with ease. Such a homography allows us to compute cross-attention in a focused manner, where key/value sets required by Transformers can be reduced to small fix-sized regions rather than an entire image. Local features can thus be enhanced by standard Transformers. We integrate GeoFormer into the LoFTR framework. By minimizing a multi-scale cross-entropy based matching loss on auto-generated training data, the network is trained in a fully self-supervised manner. Extensive experiments are conducted on multiple real-world datasets covering natural images, heavily manipulated pictures and retinal images. The proposed method compares favorably against the state-of-the-art.*

1. Introduction

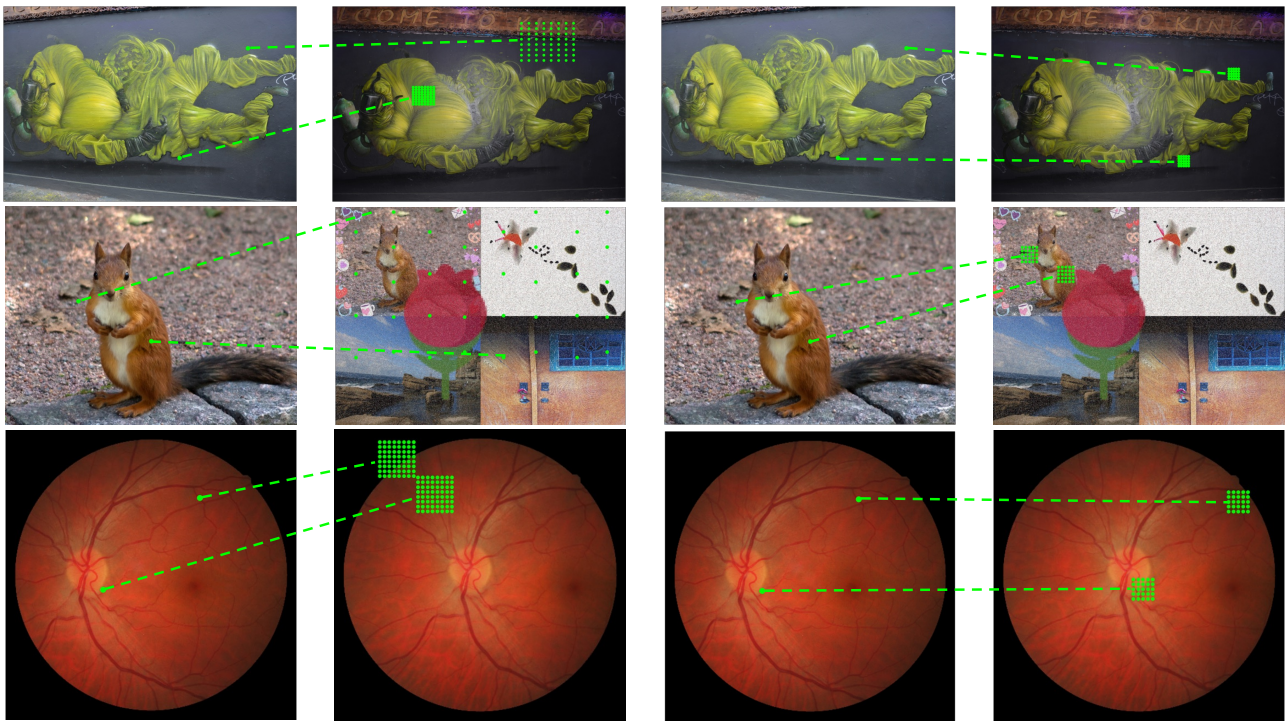
Homography estimation, also known as perspective transformation or planar projection, is a fundamental problem in computer vision and robotics. It involves estimating a 3×3 matrix that maps corresponding points in two images taken from different viewpoints, assuming that the scene is planar. Homography estimation has various applications including image/video stitching [32, 7], camera calibration [33], object recognition [17], and 3D reconstruction [19, 34], *etc*. The problem is challenging due to various factors such as occlusions, noise, and perspective distortions.

Various techniques have been developed to estimate homography efficiently and accurately. These include matching based methods [18, 4, 22] and unsupervised deep homography methods [10, 16, 31]. Matching based methods detect distinctive features such as keypoints or corners from given images and then match the features for homography estimation. SuperGlue [24], for instance, employs SuperPoint [4] for feature detection and description, and then uses Transformers [29] to enhance the features by self-attention and cross-attention. The method however suffers from the lack of discriminative features when dealing with textureless or blurry images [23]. Deep homography methods, on the other hand, directly minimize the photometric or feature differences between two images. As such, they can be sensitive to errors or ambiguities in the data, making them difficult to handle image pairs with a large baseline [16, 31, 10].

A novel trend is to develop *detector-free* feature matching methods, see NCNet [23], LoFTR [26], ASpanFormer [3] and DKM [6]. These methods find matches by dense pixel-to-pixel matching, with no need for keypoint detection. Conceptually, LoFTR can be viewed as a detector-free variant of SuperGlue. Being detector-free means the number of features to be updated by Transformers is equal to the number of pixels. In order to update high-resolution feature maps at affordable computational cost, LoFTR has to replace the normal Transformers with linear Transformers [11]. However, the latter tends to diffuse among large areas instead of focusing sharply on corresponding regions [3, 27]. Consequently, noise can be introduced during feature updating, resulting in incorrect matches. In order to localize attention regions that are computationally feasible for the normal Transformers, ASpanFormer regresses flow maps in each cross-attention phase. However, flow map regression can be error-prone, see Fig. 1a.

In order to support local feature interaction and enhancement with standard Transformers, we propose in this paper Geometrized Transformer (GeoFormer). Our idea is extremely simple. Instead of flow map regression, we propose

*Corresponding author.



(a) ASpanFormer (b) GeoFormer (*this paper*)

Figure 1: **Visualizing cross-attention diffusion regions** of (a) ASpanFormer [3] and (b) the proposed GeoFormer. Per image pair, a specific query point is shown in green dot on the left-hand image, while its cross-attention region (*i.e*. the key / value set) is shown in green dots on the right-hand image. ASpanFormer uses flow map regression, which appears to be inaccurate. By contrast, GeoFormer exploits the RANSAC geometry to identify small yet geometrically verified regions.

to use the classical RANSAC geometry for attentive region search. As shown in Fig. 1b, the geometry allows us to compute cross-attention in a focused manner such that the key / value set required by the Transformers can be reduced to small fix-sized regions on the feature maps. Viewing these geometrically localized regions as keypoints detected at a coarse level, the proposed method essentially refines local features in a detector-based manner, whilst matching the refined features in a detector-free manner.

In sum, our main contributions are as follows:

- We propose GeoFormer, a new detector-free feature matching method for homography estimation. With the proposed sparse self-attention and focused cross-attention blocks, local feature updates are achieved with standard Transformers, which have more accurate attention diffusion regions than linear Transformers.
- GeoFormer is integrated with ease into LoFTR, see Fig. 2. By minimizing a multi-scale cross-entropy based matching loss on auto-generated training data, the entire network is end-to-end trained in a fully self-supervised manner.
- Experiments on natural images [2], heavily edited pictures [5] and retinal images [9] show that GeoFormer compares favorably against the state-of-the-art. Code is released.

2. Related Work

We categorize the current methods for homography estimation into the following three groups: deep homography [16, 31, 10], detector-based feature matching [4, 22, 15] and detector-free feature matching [23, 26, 3].

Deep homograpy. Deep homography methods aim to train a deep network that directly predicts the homography from a given pair of source and target images. A common objective is to minimize the distance of inlier regions from the warped source image to the target image. CA-Unsupervised [16] obtains the homography matrix from four regressed 2D offset vectors by solving a linear system. A mask is learned to skip outlier regions that interfere with homography training. Instead of regressing the corner offsets, BasesHomo [31] decomposes the homography matrix into 8 orthogonal flow bases and predicts weights of the bases. HomoGAN [10] addresses the problem of plane-induced parallax, using a coplanarity-aware GAN to let the model focus on the dominant plane. While these methods work for image pairs with a small baseline, *e.g*. consecutive video frames or photos captured by a dual-camera cellphone, they cannot handle large geometric changes.

Detector-based feature matching. In contrast to deep homography, a detector-based feature matching method typically works in four steps. That is, keypoint detection, local feature extraction per keypoint, content-based feature matching, and lastly homography fitting based on the matches. The classical SIFT detector finds corners and blobs in a scale-invariant manner [18]. SuperPoint is a deep learning based method, presenting a self-supervised solution named homographic adaptation to address the lack of keypoint labels [4]. Homographic adaptation is also shown to be effective for training deep learning based detectors for retinal images, either in a fully self-supervised manner, see GLAMpoints [28], or in a semi-supervised manner, see SuperRetina [15]. Keypoints detected by SuperPoint generally have high repeatability, but are not necessarily reliable. To remedy the issue, R2D2 produces two probabilistic maps to measure the reliability and the repeatability of each pixel being a keypoint [22]. SuperGlue adapts Transformers [29] to match two sets of local features produced by SuperPoint. A major drawback of detector-based feature matching methods is that they rely heavily on the quality of the detected keypoints, which are known to suffer from large viewpoint changes and textureless regions [23, 26].

Detector-free feature matching. Detector-free feature matching methods perform dense pixel-to-pixel matches, with no need for keypoint detection [23, 26, 3, 6]. Such a design makes it possible to find correspondences even in textureless areas. NCNet [23] constructs 4D cost volumes to enumerate all possible matches between two images. Given the quadratic complexity w.r.t. the number of pixels, feature maps used for matching have to be substantially downsized. DKM [6] improves dense matching with a kernel regression global matcher, the optimization of which requires depth information. LoFTR [26] improves over SuperGlue with Transformers to exploit self-/inter- correlations among all dense-positioned local features. To make its computation feasible, LoFTR uses linear Transformers rather than standard Transformers. Recent studies report that the linear Transformers tend to diffuse among large areas instead of focusing sharply on the corresponding regions [3, 27]. In order to use the standard Transformers with a modest cost, ASpanFormer [3] relies on regressed flow maps in each cross-attention phase to localize attentive regions. However, due to the inaccuracy of flow map regression, ASpanFormer may become unstable, see Fig. 1. By contrast, we propose to use the simple RANSAC geometry for attentive region search, resulting in a novel detector-free feature matching method that is highly effective for homography estimation.

3. Proposed Method

As our method is developed on the basis of LoFTR, we first describe that method briefly.

3.1. LoFTR in a Nutshell

Conceptually, LoFTR works as follows. Given a pair of (gray-scale) input images I_0 and I_1 sized[1] $w \times h$, a 2D-CNN (ResNetFPN [14]) is used to extract a coarse-level feature map C_l at $\frac{1}{8}$ of the original image dimension and a fine-level feature map F_l at $\frac{1}{2}$ of the original image dimension per image, $l = 0, 1$. The two coarse features C_0 and C_1 are then fed into a `coarse-matching` module, updated with linear Transformers based self attention (SA) and cross attention (CA), to produce a pixel-to-pixel confidence matrix P_c. Each element of P_c, accessed by $P_c(i, j)$, indicates the probability of pixel i in C_0 and pixel j in C_1 being a truly matched pair, with $1 \leq i, j \leq \frac{w \times h}{64}$. By thresholding P_c followed by mutual nearest neighbor search, an array of coarse-level matches $M_c = \{(i, j)\}$ is determined.

Given M_c and the two fine-level features F_0 and F_1, a `fine-matching` module produces sub-pixel matches as follows. For each match $(i, j) \in M_c$, a local window of size $s \times s$ centered at $\hat{i} = i \times 4$ is cropped from F_0, and another local window of the same size centered at $\hat{j} = j \times 4$ is cropped from F_1. The cropped features, again enhanced by SA and CA, are used to compute the matching probability. By finding the best match of pixel $\hat{i}$ within the window in F_1, an array of sub-pixel matches M_f is obtained. More formally, we express the above process as follows:

$$\begin{cases} (C_0, F_0), (C_1, F_1) & \leftarrow \text{CNN}([I_0, I_1]), \\ P_c, M_c & \leftarrow \text{coarse-matching}(C_0, C_1), \\ M_f & \leftarrow \text{fine-matching}(F_0, F_1, M_c). \end{cases} \tag{1}$$

3.2. Geometrized Transformer

GeoFormer is designed to improve the coarse features, C_0 and C_1, with K / V information from *geometrically* matched areas rather than from the entire images, and thus generate coarse matches M_g more accurate than M_c. To that end, we first conduct the classical `RANSAC` algorithm on M_c, obtaining a homography matrix H_c and $\widetilde{M}_c$ that fits H_c. For the i-th pixel in C_0, its correspondence in C_1 is indexed by $H(i)$. Similarly, for the j-th pixel in C_1, its correspondence in C_0 is indexed by $H_c^{-1}(j)$. In order to update C_0 and C_1 subject to $\widetilde{M}_c$, we introduce sparse self-attention and focused cross-attention.

Sparse self-attention (`sparse-SA`). Given each feature in C_0 as a query vector Q of length d, we exclusively use pixels from $\widetilde{M}_c$ as the K / V set. Note that the size of $\widetilde{M}_c$ is substantially smaller than the number of pixels in C_0. Such a sparse K / V set allows us to update Q with a standard transformer. We use $\widetilde{C}_0$ to denote C_0 updated by `sparse-SA`. In a similar manner, we define $\widetilde{C}_1$.

[1] Letting the input images have the same size is merely for the ease of description.

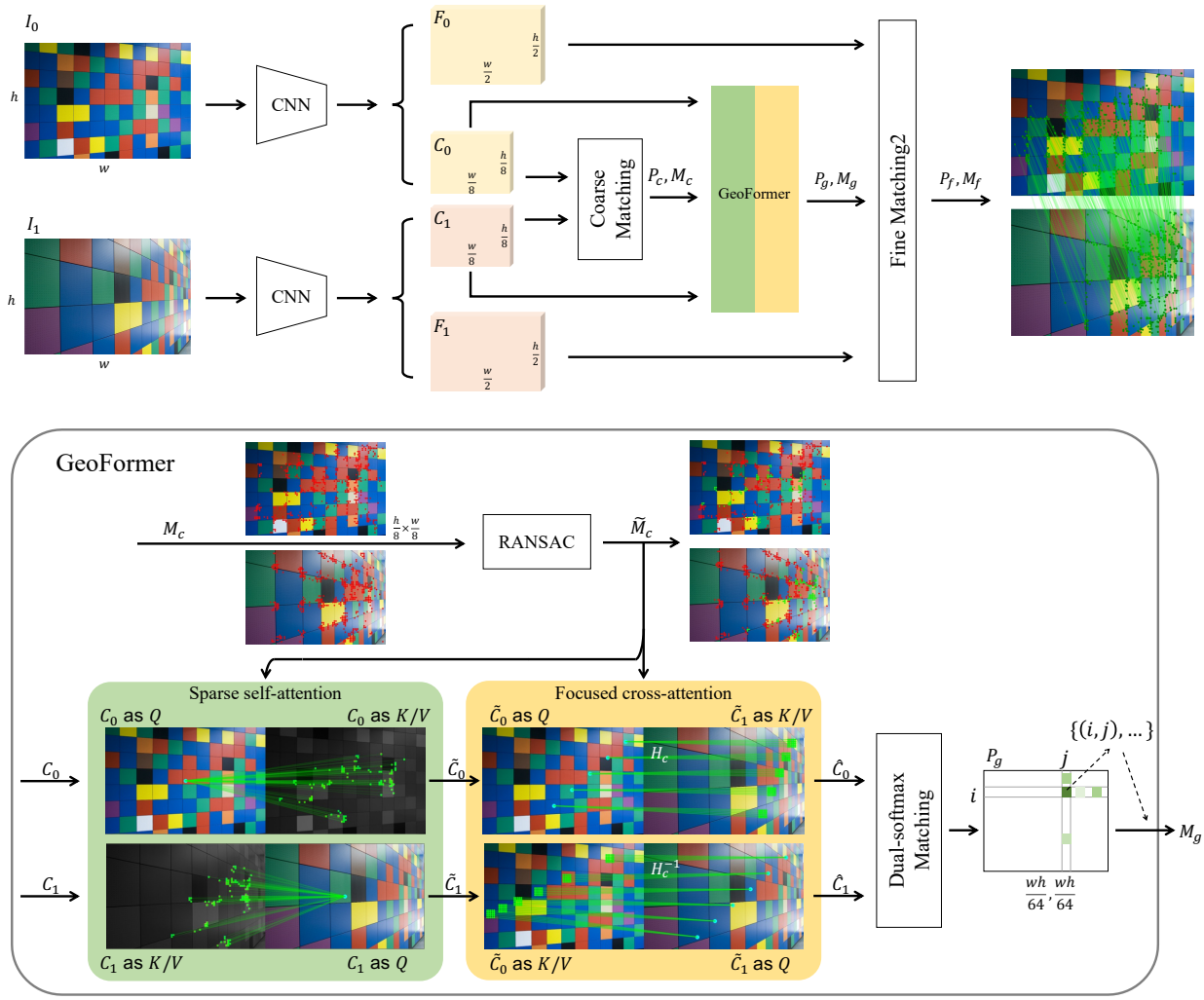


Figure 2: **Geometrized Transformer (GeoFormer) for detector-free local feature matching and subsequently homography estimation**. The `coarse matching` block, taken from LoFTR, produces pixel-to-pixel matches M_c at the $\frac{1}{8}$ scale. GeoFormer conducts `RANSAC` on M_c to estimate a homography H_c, and accordingly identify potentially common areas between the two input images I_0 and I_1. Using standard Transformers of $softmax(\frac{QK^T}{\sqrt{d_v}}V)$, GeoFormer computes self-attention in a sparse manner that the key K / value V set exclusively comes from the common areas. Meanwhile, cross-attention is computed in a focused manner that for a specific query point i in one image, its K / V set is a $s \times s$ local patch, shown in green squares, centered at $H_c(i)$ in the other image. GeoFormer outputs refined matches M_g, which are then fed to the `fine matching2` block, taken from LoFTR with small revision, for producing fine matches M_f at the $\frac{1}{2}$ scale.

Focused cross-attention (`focused-CA`). Similar to LoFTR, we also use a cross-attention (CA) transformer to let $\widetilde{C}_0$ and $\widetilde{C}_1$ interact and enhance mutually. However, in contrast to LoFTR that uses the global CA, we propose a focused CA based on our conjecture that interaction with geometrically matched areas is adequate. In particular, for each pixel i in $\widetilde{C}_0$, we use $H(i)$ to localize its correspondence in $\widetilde{C}_1$. Subsequently a local patch of $s \times s$ centered on $H(i)$ is cropped from $\widetilde{C}_1$ as the K / V set. We use $\hat{C}_0$ to denote $\widetilde{C}_0$ updated by `focused-CA`. In a similar manner, we define $\hat{C}_1$. Since s is small (5 in this work), `focused-CA` practically has a linear complexity w.r.t. to the number of query features. The geometrized selection of the K / V set is crucial for computing SA / CA using standard transformers.

While both LoFTR and GeoFormer run CA in a correspondence window, their choice of the query set differs. The query set in LoFTR is limited to regions localized by coarse matching. By contrast, with the RANSAC-obtained homography, GeoFormer allows each pixel to be used a query and accordingly have its feature updated via CA.

Given the self and mutually enhanced features $\hat{C}_0$ and $\hat{C}_1$, we follow LoFTR to produce coarse matches M_g. Specifically, we compute pixel-to-pixel similarities followed by a dual-softmax operation. This results in a confidence matrix P_g, where $P_g(i,j)$ indicates the probability of pixel i in $\hat{C}_0$ and pixel j in $\hat{C}_1$ being a true match. With threshold-based masking and mutual nearest neighbor search on P_g, M_g is obtained. Given C_0, C_1 and M_c as input, the workflow of GeoFormer is summarized as follows:

$$\begin{cases} H_c, \widetilde{M}_c & \leftarrow \text{RANSAC}(M_c), \\ \widetilde{C}_0, \widetilde{C}_1 & \leftarrow \text{sparse-SA}(C_0, C_1, \widetilde{M}_c), \\ \hat{C}_0, \hat{C}_1 & \leftarrow \text{focused-CA}(\widetilde{C}_0, \widetilde{C}_1, H_c), \\ P_g, M_g & \leftarrow \text{dual-softmax-matching}(\hat{C}_0, \hat{C}_1). \end{cases} \tag{2}$$

3.3. Adding GeoFormer to LoFTR

By substituting M_g for M_c in `fine-matching` in Eq. 1, GeoFormer can be integrated with ease into the LoFTR framework. Recall that the fine matching by LoFTR is asymmetric: A coarsely matched point $\hat{i}$ in F_0 is used to find matches in F_1, with the coordinate of $\hat{i}$ not adjustable. As a remedy, we introduce a symmetric matching strategy as follows. For each matched point $\hat{i}$ in F_0 and $\hat{j}$ in F_1, we crop two sets of local windows of size $s \times s$, which are then enhanced by LoFTR's SA and CA. Similar to generating confidence matrix P_g, we compute pixel-to-pixel similarities of two local windows followed by a dual-softmax operation. Threshold-based masking and then mutual nearest neighbor search are performed to find adjustable $(\hat{i}, \hat{j})$, which will be added to M_f. The new strategy is referred to as `fine-matching2`. Putting everything together, we have GeoFormer-embedded LoFTR as

$$\begin{cases} (C_0, F_0), (C_1, F_1) & \leftarrow \text{CNN}([I_0, I_1]), \\ P_c, M_c & \leftarrow \text{coarse-matching}(C_0, C_1), \\ P_g, M_g & \leftarrow \text{GeoFormer}(C_0, C_1, M_c), \\ P_f, M_f & \leftarrow \text{fine-matching2}(F_0, F_1, M_g). \end{cases} \tag{3}$$

3.4. Self-supervised Training

GeoFormer is trained in a self-supervised manner with no need of manual annotation. For the given image I_0, we make I_1 its geometric transformation by applying a controlled homography on I_0. With the homography, for each i in M_f, its true correspondence j in I_1 can be directly calculated. Accordingly, ground-truthed matches G_f w.r.t. M_f is generated on the fly. In a similar manner, we obtain ground-truthed matches G_c for both M_c and M_g. Data augmentation is performed by randomly sampling the homography. Varied random photometric distortions, *e.g.* brightness and contrast adjustment, motion blurring, and Gaussian noising, are also applied on the paired images.

As shown in Fig. 2, three pixel-to-pixel similarity matrices, *i.e.* P_c, P_g and P_f, are produced by the coarse matching module, GeoFormer and the fine matching module, respectively. Naturally, the matrices shall be close to their ground truth. By computing a cross-entropy loss per matrix and having the individual losses equally combined, we define a multi-scale loss as follows:

$$\begin{array}{l} -\frac{1}{|G_c|}\sum_{(i,j)\in G_c}\log(P_c(i,j)) \\ -\frac{1}{|G_c|}\sum_{(i,j)\in G_c}\log(P_g(i,j)) \\ -\frac{1}{|G_f|}\sum_{(i,j)\in G_f}\log(P_f(i,j)). \end{array} \tag{4}$$

Our training is performed by minimizing the multi-scale loss. Compared with previously used auxiliary losses such as flow map regression [3] that are indirectly related to the task, our loss terms are consistent by definition, optimizing the feature matching at the coarse ($\frac{1}{8}$) and fine ($\frac{1}{2}$) scales.

4. Experiments

We evaluate GeoFormer on three distinct types of images, *i.e.* natural images with large variations in viewpoints and illumination [2], severely manipulated pictures [5], and retinal images with varied eye conditions [9]. An overview of our experimental data is given in Tab. 1.

Table 1: **Experimental data**. Models trained on Oxford-Paris are tested on HPatches and ISC-HE for homography estimation on generic images. Models trained on Lab-Aux will be evaluated on FIRE for retinal image registration.

Dataset	Image content	Images	Registered pairs
Training:			
Oxford-Paris	Outdoor / city scenes [20, 21]	11,455	auto-generated
Lab-Aux [15]	Color fundus photos	919	
Test:			
HPatches [2]	Planar photos with changes in photometry or geometry	696	580
ISC-HE	Severely manipulated pictures	372	186
FIRE [9]	Color fundus photos	129	134

4.1. Common Setup

We implement GeoFormer using PyTorch. Subject to our GPU resource (8 NVIDIA GeForce RTX 3090 cards), the larger dimension of training images is set to 640 for natural images and 768 for retinal images. The optimizer is Adam [12], with $\beta = (0.9, 0.999)$ and an initial learning rate of 0.001. Each mini-batch contains a single pair of images. The maximum number of training epochs is 10. At the inference stage, given the fine matches M_f, we use `cv2.findHomography` with RANSAC as the robust estimator for homography fitting. All evaluation is performed using an open-source toolbox[2].

[2] https://github.com/GrumpyZhou/image-matching-toolbox

4.2. Evaluation on Natural Images

4.2.1 Setup

Training data. We combine Oxford5K [20] and Paris6K [21], denoted by Oxford-Paris, as our training images. Registered image pairs are auto-generated, see Section 3.4.

Test data. We adopt the widely used HPatches [2]. The dataset has 57 sequences that undergo significant changes in illumination and 59 sequences that manifest considerable variations in viewpoints, rendering it a highly challenging benchmark for homography estimation.

Evaluation criteria. Following [4, 26], we compute the corner error between the images warped with the estimated homography matrix and the ground truth homography matrix as a correctness identifier. We report the area under the cumulative curve (AUC) of the corner error up to threshold values of 3, 5, and 10 pixels, respectively. All test images are resized with shorter dimensions equal to 480 [26].

Baselines. We compare the following three sorts of methods, *i.e*. deep homography estimation, detector-based matching, and detector-free matching. For a reproducible comparison, we prefer open-sourced methods, compiling a list of 11 baseline methods as follows:
i) *Deep homography estimation*: CA-Unsupervised[3][16], BasesHomo[4][31], and HomoGAN[5][10].
ii) *Detector-based matching*: SIFT[18]+RootSIFT[1], SuperPoint[6][4], SuperGlue[7][24], and R2D2[8][22].
iii) *Detector-free matching*: NCNet[9][23], LoFTR[10][26], ASpanFormer[11][3], and DKM[12][6].

Both ASpanFormer and our GeoFormer are developed based on LoFTR. So for a head-to-head comparison, the same training data is used to train these three models. DKM is also retrained. As DKM samples good samples at random, causing slightly varied performance per inference, we report its averaged result. As for other learning-based methods, we directly take their author-released models.

4.2.2 Results

Tab. 2 presents the AUC scores of various methods. Recall that the deep homography methods are specifically designed for image pairs with a small baseline [16]. Hence, they do not perform well on the challenging HPatches benchmark. We thus exclude them from the remaining experiments.

[3] https://github.com/JirongZhang/DeepHomography
[4] https://github.com/megvii-research/BasesHomo
[5] https://github.com/megvii-research/HomoGAN
[6] https://github.com/rpautrat/SuperPoint
[7] https://github.com/magicleap/SuperGluePretrainedNetwork
[8] https://github.com/naver/r2d2
[9] https://github.com/ignacio-rocco/ncnet
[10] https://github.com/zju3dv/LoFTR
[11] https://github.com/apple/ml-aspanformer
[12] https://github.com/Parskatt/DKM

Table 2: **Performance on HPatches**.

Method	Homography est. AUC			
	@3px	@5px	@10px	mAUC
Deep homography:				
CA-Unsupervised [16]	20.5	31.7	40.1	30.8
HomoGAN [10]	34.2	38.3	42.1	38.2
BasesHomo [31]	38.3	42.4	45.5	42.1
Detector-based matching:				
SuperPoint [4]	43.4	57.6	72.7	57.9
SIFT [18]	46.3	57.4	70.3	58.0
R2D2 [22]	50.6	63.9	76.8	63.8
SuperGlue [24]	53.9	68.3	81.7	68.0
Detector-free matching:				
DKM [6]	30.6	37.3	44.5	37.5
NCNet [23]	48.3	50.1	59.8	52.7
LoFTR [26]	58.5	69.8	81.1	69.8
ASpanFormer [3]	59.9	71.1	81.6	70.9
GeoFormer	**68.0**	**76.8**	**85.4**	**76.7**
1: w/o RANSAC	61.3	71.2	81.8	71.4
2: w/o Focused CA.	63.5	73.0	82.4	73.0
3: Linear Transformers	63.2	73.5	83.5	73.4
4: fine-matching	65.4	72.5	82.9	73.6
5: w/o Sparse SA.	66.0	75.4	84.8	75.4

Among the baselines, the better performance of ASpanFormer and LoFTR (both use Transformers) and the lower performance of NCNet (which uses no Transformers) than the detector-based methods justify the importance of Transformers for dense feature matching. The much lower performance of DKM is due to the absence of depth information in Oxford-Paris. Compared to ASpanFormer which uses the dense flow map to localize attention regions, GeoFormer has the largest improvement in terms of AUC@3px (68.0 versus 59.9). This result shows the accuracy of GeoFormer.

4.3. Evaluation on Manipulated Images

4.3.1 Setup

Models from Sec. 4.2 are adopted, without re-training.

Test data. We took images from the Facebook AI Image Similarity Challenge (ISC) [5], where an original image has been edited in varied manners, *e.g*. rotated and combined with another image, to create a severely manipulated image. Since the ISC image pairs are not registered, manual and collective labeling was performed, producing a number of 186 registered pairs in total. Each pair has 8 correspondences at minimum. We term the testset ISC-HE. Different from HPatches, ISC-HE has forgery characteristics such as watermarks, cutouts, and image stitching. Homography estimation on ISC-HE is thus even more challenging.

4.3.2 Results

As Tab. 3 shows, GeoFormer surpasses all baselines, although the performance difference is relatively small when

compared with the HPatches experiment. We attribute this to the extremely challenging nature of ISC-HE, see Fig. 1.

The detector-based methods, especially SuperGlue (mAUC 42.9) and SIFT (mAUC 42.4), now surpasses the best detector-free baseline, *i.e*. LoFTR (mAUC 41.5). We interpret the result as follows. ISC-HE images were heavily modified, with watermarks inserted and/or background replaced. Compared to the detector-free methods that perform dense feature matching, the detector-based methods are less sensitive to these modifications. Indeed, the better performance of GeoFormer than both detector-free and detector-based methods confirms our conjecture made in Sec. 1 that the proposed method combines the best of both worlds.

Table 3: **Performance on ISC-HE**.

Method	Homography est. AUC			
	@3px	@5px	@10px	mAUC
Detector-based matching:				
SuperPoint [4]	18.3	39.0	62.2	39.8
R2D2 [22]	18.2	39.6	62.9	40.2
SIFT [18]	**19.9**	42.4	65.0	42.4
SuperGlue [24]	19.6	42.2	66.9	42.9
Detector-free matching:				
DKM [6]	7.1	15.3	25.6	16.0
NCNet [23]	9.6	25.3	51.2	28.7
ASpanFormer [3]	18.0	39.2	62.0	39.7
LoFTR [26]	18.7	41.0	64.8	41.5
GeoFormer	**19.9**	**43.8**	**68.4**	**44.0**
1: w/o RANSAC	17.5	40.6	66.2	41.4
2: fine-matching	18.5	41.4	65.6	41.8
3: Linear Transformers	18.4	42.5	68.2	43.0
4: w/o Focused CA.	19.0	42.7	67.7	43.1
5: w/o Sparse SA.	19.6	43.7	68.2	43.8

4.4. Evaluation on Retinal Images

4.4.1 Setup

Training data. We adopt the Lab-Aux dataset [15], which has 919 color fundus photos in normal conditions.

Test data. We adopt the FIRE benchmark [9]: 129 images of size $2,912 \times 2,912$ acquired with a Nidek AFC-210 fundus camera (FOV of $45°$) and 134 registered image pairs. The pairs have been divided into three groups according to their registration difficulty: Category $\mathcal{S}$ (71 pairs with high overlap and no anatomical change), $\mathcal{A}$ (14 pairs with high overlap and large anatomical changes), and $\mathcal{P}$ (49 pairs with small overlap and no anatomical changes).

Evaluation criteria. Following [28, 15], the input image size for inference is 768×768. We report Area Under Curve (AUC) proposed in [9], estimating the expectation of the acceptance rates w.r.t. the decision threshold and thus reflects the overall performance. In particular, AUC per group, *i.e*. easy(S), moderate(A), and hard(P), is computed. In addition, we report three sorts of success rate, *i.e*. failed, inaccurate, and accurate, see [28] for more details. All the metrics are computed on the original size of 2912×2912.

Baselines. We re-use the previously evaluated baselines whenever applicable: SIFT, SuperPoint, SuperGlue, R2D2, NCNet, DKM, LoFTR, and ASpanFormer. In addition, we include the following three detector-based matching methods specifically designed for retinal image registration: REMPE [8], GLAMPoints [28] and SuperRetina [15].

4.4.2 Results

As Tab. 4 shows, GeoFormer obtains the best mAUC of 75.6 and an accurate rate of 98.51, marginally better than the best baseline, *i.e*. SuperRetina, which has mAUC of 75.5 and the same accurate rate of 98.51. Also notice that while REMPE has a lower mAUC of 72.0, the method tops the performance on the easy group (mAUC 95.8). It is worth pointing out that both SuperRetina and REMPE are designed specifically for retinal image registration. Moreover, SuperRetina is semi-supervised trained with a set of manually labeled keypoints, whilst REMPE takes around three minutes per registration. In such a context, the result that GeoFormer is on par with the state-of-the-art is appealing.

Table 4: **Performance on FIRE**.

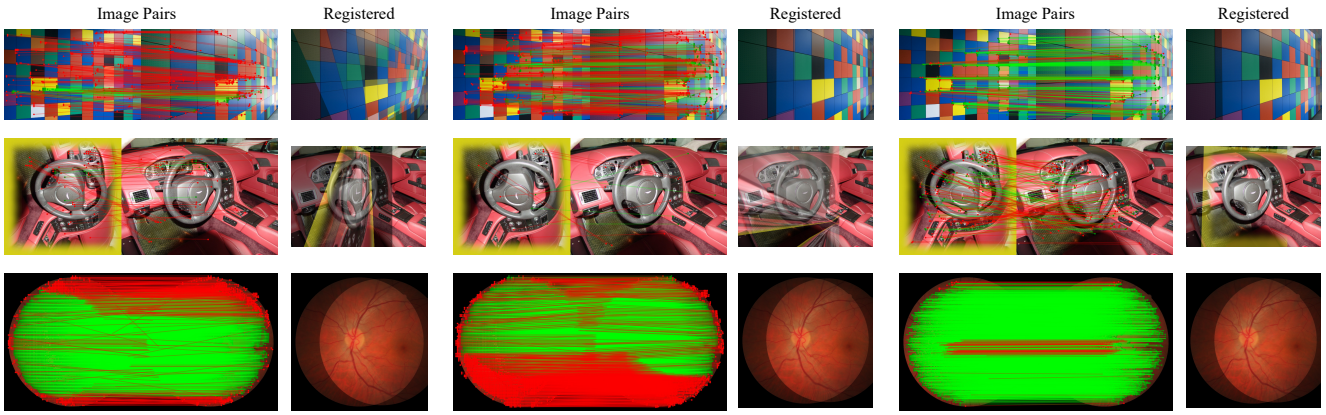
Method	Homography est. AUC				Success rate		
	Easy	Mod	Hard	mAUC	Failed	Inaccurate	Accurate
Detector-based matching:							
SIFT [18]	90.3	47.4	34.1	57.3	0	20.15	79.85
GLAMPoints [28]	82.5	51.7	49.0	61.1	0	7.46	92.54
SuperPoint [4]	88.2	64.9	49.0	67.4	0	5.22	94.78
SuperGlue [24]	88.5	68.9	48.8	68.7	0.75	3.73	95.52
R2D2 [22]	92.8	66.6	54.0	71.1	0	4.48	95.52
REMPE [8]	**95.8**	66.0	54.2	72.0	0	2.99	97.01
SuperRetina [15]	94.0	**78.3**	54.2	75.5	0	1.49	**98.51**
Detector-free matching:							
DKM [6]	93.1	60.3	20.6	58.0	0	24.06	75.94
NCNet [23]	81.7	60.9	41.0	61.2	0	14.18	85.82
LoFTR [26]	92.0	71.1	35.9	66.3	0	3.01	96.99
ASpanFormer [3]	92.1	70.3	49.5	70.6	0	8.27	91.73
GeoFormer	94.4	76.6	**55.9**	**75.6**	0	1.49	**98.51**
1: w/o RANSAC	90.9	61.7	49.3	67.3	0	4.48	95.52
2: w/o Focused CA.	93.4	77.1	46.1	72.2	0	8.27	91.73
3: Lineal Transformer	91.3	72.0	55.5	72.9	0	3.01	96.99
4: fine-matching	94.0	74.9	54.3	74.4	0	3.76	96.24
5: w/o Sparse SA.	93.9	74.6	55.3	74.6	0	1.49	**98.51**

4.5. Understanding GeoFormer

We conduct ablation studies as follows. Some qualitative results are shown in Fig. 3.

RANSAC. Recall that without RANSAC, only the query points in M_c are updated via the focused CA module. As Tab. 2, 3 and 4 show, GeoFormer w/o RANSAC leads a clear performance drop in mAUC (HPatches 76.7 $\rightarrow$ 71.4, ISC-HE 44.0 $\rightarrow$ 41.4, FIRE 75.6 $\rightarrow$ 67.3).

Sparse SA / Focused CA. Removing either sparse SA or focused CA leads to consistent performance decrease, with



(a) LoFTR (b) ASpanFormer (c) GeoFormer

Figure 3: **Homography estimation by LoFTR, ASpanFormer and GeoFormer**. Inliers / outliers determined by RANSAC are shown in green / red color. From top to bottom are samples from HPatches, ISC-HE and FIRE, respectively.

the latter corresponding to a larger performance drop. Focused CA is more important for enhancing local features.

Linear Transformers? We substitute linear Transformers for standard Transformers in sparse-SA and focused-CA. Such a replacement hurts the performance (HPatches 76.7 → 73.4, ISC-HE 44.0 → 43.0, FIRE 75.6 → 72.9).

Fine-matching versus Fine-matching2. Using LoFTR's `fine-matching` as an alternative to our `fine-matching2` degenerates the performance consistently (HPatches 76.7 → 73.6, ISC-HE 44.0 → 41.8, FIRE 75.6 → 74.4). Note that the `fine-matching` run here uses GeoFormer, so it is better than the original LoFTR.

LoFTR w/ and w/o GeoFormer. On average, LoFTR w/o GeoFormer has 2,633 matches with an inlier rate of 87%. By contrast, LoFTR with GeoFormer has 2,155 matches with an inlier rate of 90%. The result indicates that the latter finds fewer yet more precise matches.

Possible failure cases. GeoFormer runs RANSAC on the coarse matches by LoFTR to obtain an initial homography guess. If the coarse matches fail in the first place, GeoFormer shall fail. Such cases occur rarely.

Trained on MegaDepth [13]. GeoFormer again outperforms the baselines for homography estimation, see Tab. 5.

Applicability to other tasks? As Tab. 5 shows, GeoFormer is marginally worse than LoFTR for visual localization. Results on InLoc [30] are given in the supplement.

Table 5: **Multi-task evaluation**. Visual localization is tested on Aachen Day-Night v1.0 (local feature evaluation) [25]. Training data: MegaDepth.

	Homography estimation			**Visual localization**		
	HPatches	ISC-HE	FIRE	0.25m,2°	0.5m,5°	5m,10°
LoFTR	75.0	39.3	73.6	**79.6**	**91.8**	100.0
DKM	79.8	41.0	74.8	78.6	85.7	100.0
GeoFormer	**79.9**	**44.7**	**75.7**	77.6	86.7	100.0

Efficiency. As shown in Tab. 6, compared to LoFTR, the inference runtime and memory overhead of GeoFormer increase by 24 ms and 184 MB, respectively.

Table 6: **Runtime and memory per batch**. Each batch has a pair of 640×480 images. GPU: NVIDIA RTX 3090.

	Runtime (milliseconds)		**GPU memory** (MB)	
	training	*inference*	*training*	*inference*
LoFTR	330	77	8,719	3,089
GeoFormer	540	101	11,056	3,273
DKM	400	208	11,217	7,627

5. Conclusions

GeoFormer is a new detector-free feature matching method for self-supervised homograhy estimation. Extensive experiments on three dataset, *i.e*. HPatches, ISC-HE and FIRE, support the following conclusions. Compared with the state-of-the-art trained on our data, *i.e*. ASpanFormer on HPatches, SuperGlue on ISC-HE and SuperRetina on FIRE, GeoFormer is clearly better than ASpanFormer and SuperGlue, and marginally better than SuperRetina. While both sparse self-attention and focused cross-attention are necessary, the latter is more important for local feature enhancement. By exploiting the RANSAC geometry, GeoFormer essentially enhances local features in a detector-based manner, and meanwhile performs feature matching in a detector-free manner. As such, the proposed method combines the best of the two worlds.

Acknowledgements. This work was supported by the National High Level Hospital Clinical Research Funding (2022-PUMCH-C-61), NSFC (62172420), Tencent Marketing Solution Rhino-Bird Focused Research Program, and Public Computing Cloud, Renmin University of China.

References

[1] Relja Arandjelović and Andrew Zisserman. Three things everyone should know to improve object retrieval. In *CVPR*, 2012. 6

[2] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *CVPR*, 2017. 2, 5, 6

[3] Hongkai Chen, Zixin Luo, Lei Zhou, Yurun Tian, Mingmin Zhen, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. ASpanFormer: Detector-free image matching with adaptive span transformer. In *ECCV*, 2022. 1, 2, 3, 5, 6, 7

[4] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperPoint: Self-supervised interest point detection and description. In *CVPRW*, 2018. 1, 2, 3, 6, 7

[5] Matthijs Douze, Giorgos Tolias, Zoë Papakipos, Ed Pizzi, Lowik Chanussot, Filip Radenovic, Tomas Jenicek, Maxim Maximov, Laura Leal-Taixé, Ismail Elezi, Ondřej Chum, and Cristian Canton Ferrer. The 2021 Image Similarity Dataset and Challenge. *arXiv e-prints*, 2021. 2, 5, 6

[6] Johan Edstedt, Ioannis Athanasiadis, Mårten Wadenbäck, and Michael Felsberg. Dkm: Dense kernelized feature matching for geometry estimation. In *CVPR*, 2023. 1, 3, 6, 7

[7] Heng Guo, Shuaicheng Liu, Tong He, Shuyuan Zhu, Bing Zeng, and Moncef Gabbouj. Joint video stitching and stabilization from moving cameras. *IEEE Transactions on Image Processing*, 2016. 1

[8] Carlos Hernandez-Matas, Xenophon Zabulis, and Antonis A Argyros. REMPE: Registration of retinal images through eye modelling and pose estimation. *IEEE Journal of Biomedical and Health Informatics*, 24(12):3362–3373, 2020. 7

[9] Carlos Hernandez-Matas, Xenophon Zabulis, Areti Triantafyllou, Panagiota Anyfanti, Stella Douma, and Antonis A Argyros. FIRE: Fundus image registration dataset. *Modeling and Artificial Intelligence in Ophthalmology*, 1(4):16–28, 2017. 2, 5, 7

[10] Mingbo Hong, Yuhang Lu, Nianjin Ye, Chunyu Lin, Qijun Zhao, and Shuaicheng Liu. Unsupervised homography estimation with coplanarity-aware GAN. In *CVPR*, 2022. 1, 2, 6

[11] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *ICML*, 2020. 1

[12] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[13] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018. 8

[14] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 3

[15] Jiazhen Liu, Xirong Li, Qijie Wei, Jie Xu, and Dayong Ding. Semi-supervised keypoint detector and descriptor for retinal image matching. In *ECCV*, 2022. 2, 3, 5, 7

[16] Shuaicheng Liu, Nianjin Ye, Chuan Wang, Kunming Luo, Jue Wang, and Jian Sun. Content-aware unsupervised deep homography estimation and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (01):1–1, 2022. 1, 2, 6

[17] Sina Lotfian and Hassan Foroosh. View-invariant object recognition using homography constraints. In *ICIP*, 2017. 1

[18] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 1, 3, 6, 7

[19] Christopher Mei, Selim Benhimane, Ezio Malis, and Patrick Rives. Efficient homography-based tracking and 3-d reconstruction for single-viewpoint sensors. *IEEE Transactions on Robotics*, 24(6):1352–1364, 2008. 1

[20] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*, 2007. 5, 6

[21] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *CVPR*, 2008. 5, 6

[22] Jerome Revaud, Philippe Weinzaepfel, César Roberto de Souza, and Martin Humenberger. R2D2: Repeatable and reliable detector and descriptor. In *NeurIPS*, 2019. 1, 2, 3, 6, 7

[23] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. NCNet: Neighbourhood consensus networks for estimating image correspondences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(2):1020–1034, 2022. 1, 2, 3, 6, 7

[24] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. SuperGlue: Learning feature matching with graph neural networks. In *CVPR*, 2020. 1, 6, 7

[25] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *CVPR*, 2018. 8

[26] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. LoFTR: Detector-free local feature matching with transformers. In *CVPR*, 2021. 1, 2, 3, 6, 7

[27] Shitao Tang, Jiahui Zhang, Siyu Zhu, and Ping Tan. Quadtree attention for vision transformers. In *ICLR*, 2022. 1, 3

[28] Prune Truong, Stefanos Apostolopoulos, Agata Mosinska, Samuel Stucky, Carlos Ciller, and Sandro De Zanet. GLAMpoints: Greedily learned accurate match points. In *ICCV*, 2019. 3, 7

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 1, 3

[30] Erik Wijmans and Yasutaka Furukawa. Exploiting 2d floorplan for building-scale panorama rgbd alignment. In *CVPR*, 2017. 8

[31] Nianjin Ye, Chuan Wang, Haoqiang Fan, and Shuaicheng Liu. Motion basis learning for unsupervised deep homography estimation with subspace projection. In *ICCV*, 2021. 1, 2, 6

[32] Julio Zaragoza, Tat-Jun Chin, Michael S Brown, and David Suter. As-projective-as-possible image stitching with moving dlt. In *CVPR*, 2013. 1

[33] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11):1330–1334, 2000. 1

[34] Zhongfei Zhang and Allen R Hanson. 3d reconstruction based on homography mapping. In *ARPA Image Understanding Workshop*, 1996. 1