Affine steerers for structured keypoint description

Georg Bökman[1], Johan Edstedt[2], Michael Felsberg[2], and Fredrik Kahl[1]

[1] Chalmers University of Technology
[2] Linköping University

Abstract. We propose a way to train deep learning based keypoint descriptors that makes them approximately equivariant for locally affine transformations of the image plane. The main idea is to use the representation theory of GL(2) to generalize the recently introduced concept of steerers from rotations to affine transformations. Affine steerers give high control over how keypoint descriptions transform under image transformations. We demonstrate the potential of using this control for image matching. Finally, we propose a way to finetune keypoint descriptors with a set of steerers on upright images and obtain state-of-the-art results on several standard benchmarks. Code will be published at github.com/georg-bn/affine-steerers.

Keywords: Keypoint description · Image matching · Equivariance

1 Introduction

Image matching is a critical component in a wide range of computer vision applications, including 3D scene reconstruction, stereo imaging, and motion tracking. Initially, the field advanced through the development of sophisticated engineering methods, tailored to achieve robust matching capable of handling significant changes in image scale and rotation, while being resilient to variations in lighting and different camera viewpoints. However, these manually crafted approaches have gradually been superseded by deep learning techniques, which offer a more adaptable and powerful framework for tackling the many other complexities of image matching. Unfortunately, these methods lost much of the robustness of the handcrafted approach. In this paper, we aim to further enhance the performance of deep learning based image matching by introducing a method specifically designed to improve robustness to affine distortions. This approach not only capitalizes on the strengths of deep learning but also revisits and revitalizes a foundational concept from early research in the field: ensuring robustness to affine distortions is essential for reliable image matching across wide baselines.

Our approach hinges on training neural network based keypoint descriptors that are approximately equivariant under local affine transformations. We train these networks by generalizing the *steerers* framework [9] from SO(2) to GL(2). We also find that training with affine steerers with heavy homography augmentation works well as pretraining before fine-tuning on upright images. This approach, while sacrificing some degree of equivariance, achieves new SotA results

Fig. 1: Qualitative example. We show qualitative matching results of our *AffSteer-B* descriptor which achieves SotA results for detector-descriptors on several benchmarks for image matching. We plot inliers after homography estimation with RANSAC.

on standard benchmarks for upright image matching. In particular, we improve on IMC22 [36] from 72.9 mAA@10 achieved by DeDoDe-B [22] to 77.3 mAA@10 with our model *AffSteer-B*. A qualitative example on images taken by the authors is shown in Figure 1. We also obtain competitive results on the rotation variant benchmark AIMS [75] by using our equivariant model *AffEqui-B*. Finally, we analyze the equivariance properties of our descriptors and outline promising directions for future research. In summary, our main contributions are:

1. Generalizing the steerers concept to image transformations that can be locally approximated by the affine group (Sections 2 and 3.1).
2. Describing how to train steerers for the affine group (Section 3.4).
3. Introducing a new training procedure for upright-specialized descriptors consisting of first pretraining with affine steerers and then fine-tuning on upright images through the max similarity method (Sections 3.5, 3.6).
4. Evaluating our methods on a wide range of standard benchmarks, with SotA results for detector-descriptor based methods (Section 4).
5. Critically examining the properties of our descriptors (Sections 4.1, 5).

1.1 Related work

Image matching. Image matching has classically been approached by detector and descriptor based methods [18, 20, 29, 47, 68, 84, 88, 91, 92] that detect keypoints [5, 77, 83, 87] and compute descriptions [2, 38, 41, 48, 57, 59, 78] typically matched using mutual nearest neighbors (MNNs). It was early on recognized that robustness to affine distortions plays an important role in order to achieve good performance [25,51,54,55,64]. ASIFT, which extends the Scale Invariant Feature Transform (SIFT) algorithm [47] by test time augmentation to offer affine invariance, was introduced in [89]. More recently, sparse keypoint matchers such as SuperGlue [70] and later works [15, 46, 73] aim to improve on the simple descriptor MNN matching using graph neural networks. Detector-free methods instead

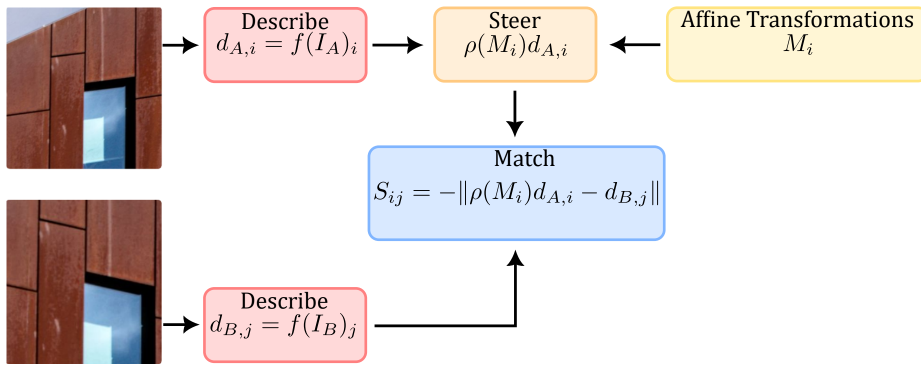

Fig. 2: Overview of the steering idea for keypoint matching. An affine steerer gives a way to modify descriptions as if they were obtained from warped images, without having to rerun the descriptor network on warped images. The steerer is a linear map and hence computationally light to use.

match on a coarse regular grid followed by sparse refinement with notable examples such as LoFTR [76], and numerous following works [10,13,28,37,50,85,90]. Finally, dense methods that aim to estimate matches on a fine regular grid such as GLU-Net [80] have lately received increased interest [21,23,61,79,81,82,94]. In this paper we focus on sparse keypoint descriptors that rely on MNN matching. This is the workhorse for numerous applications in computer vision. We use the SotA keypoint descriptor DeDoDe [22] as our baseline.

Affine correspondences. Our work is also related to methods that rely on affine correspondences. That is, apart from the keypoint location, also the affine transformation that maps one local neighborhood region to the corresponding one in the other image, is used in downstream tasks such as estimating the epipolar geometry [3,4,6,52]. There are also learned methods that provide an affine canonical frame around each keypoint, for instance, AffNet [60] which is important, *e.g.*, in image retrieval [62].

Equivariant representation learning. Self-supervised visual representations [33, 35,65,93] are typically learned by maximizing agreement between two views [14, 15,30], which is reminiscent of the image matching task, where we seek to learn local visual representations that match across views. Recent works [17,26,27,32, 39, 53, 66, 72] investigate explicitly learning equivariant representations, which makes the learned representations structured in the sense that it is possible to interpret certain changes in the latent space as specific changes to the input images. The equivariance can also be hard-coded into the network architecture, *e.g.*, for compact groups such as the rotation group [12,16,86]. This approach has been shown to work well for rotation invariant keypoint detection [34,42,69] and description [10, 43]. For the affine group acting on images, hard-coding equivariance has only been done approximately via sampling on the group and with small networks [49, 56], prohibiting the use of such approaches for SotA keypoint description. Unrelated to images, networks equivariant under the group SL(2) were considered for polynomial optimization problems in [40], using the same irreps from group representations on homogeneous polynomials that we will use for the group GL(2). In sharp contrast to hard-coding equivariance, it has been found that even without explicitly encouraging equivariance in learned representations, it often appears approximately because of symmetries in the data [7, 9, 11, 31, 44]. Closest to our work, Bökman *et al.* [9] propose a framework for learning approximately rotationally equivariant keypoint descriptors. We build on that work, generalizing it to the larger group of GL(2).

2 Background

The primary goal of keypoint matching is to identify 2D points across two images of the same scene, representing the same 3D points. A pair of corresponding 2D points is known as a correspondence. We adopt a standard three-stage method:

1. *Detection*: Identify K keypoints in each image.
2. *Description*: Generate keypoint descriptors as feature vectors in $\mathbb{R}^c$.
3. *Matching*: Descriptors are matched using mutual nearest neighbours.

This method encompasses both traditional techniques like SIFT [47] and modern deep learning approaches. A notable example is DeDoDe [22], which first optimizes keypoint detection from Structure from Motion reconstructions and then leverages the keypoint detector to train a descriptor through maximizing matching likelihood. Our focus is on improving the second stage by explicitly handling affine transformations via *steerers*, as introduced in [9] for rotation equivariant keypoint description. We generalize the formulation from rotations to locally affine transformations in the remainder of this section, before explaining how to implement and train affine steerers in the next section.

The general pipeline is outlined in Figure 2. We consider images as functions $I : \mathbb{R}^2 \to \mathbb{R}^3$ and general feature maps as functions $F : \mathbb{R}^2 \to \mathbb{R}^c$. Feature maps are obtained from feature extractors f, *i.e.*, we will write $f(I) = F$. Implicitly, we assume here that for a feature map F associated to an image I, the feature $F(x)$ at location x is associated to the image content $I(x)$ at x. The idea for keypoint description is that given a feature map F and keypoints x_i, we obtain keypoint descriptions d_i by evaluating F at the locations x_i. Images and feature maps can be warped by geometric image transformations $\phi : \mathbb{R}^2 \to \mathbb{R}^2$, by using the warp operator W_ϕ defined by

$$W_\phi[I](x) = I(\phi^{-1}(x)). \tag{1}$$

We will consider differentiable ϕ, so that they are locally approximated by affine transformations[3]. *I.e.*, for all $x \in \mathbb{R}^2$ there is an $M(x) \in \mathrm{GL}(2)$ s.t.

$$\phi(x + \Delta x) \approx \phi(x) + M(x)\Delta x, \tag{2}$$

for small Δx. $M : \mathbb{R}^2 \to \mathrm{GL}(2)$ maps image locations to local image transformations (affine warps) associated with ϕ at each point. For some classes of image transformations ϕ, $M(x)$ takes a special form. For instance, if $\phi \in \mathrm{SE}(2)$ is a roto-translation, then $M(x)$ will always be a rotation matrix and so the possible $M(x)$'s for roto-translations form the subgroup $\mathrm{SO}(2)$ of $\mathrm{GL}(2)$. Generally, given a set Φ of image transformations, we can associate with it the minimal subgroup of $\mathrm{GL}(2)$ that contains all possible local image transformations $M(x)$.

Given a feature extractor f, it is of interest how the extracted features change when the image is warped by W_ϕ. For example, we could have

$$f(W_\phi[I]) = W_\phi[f(I)], \tag{3}$$

so that the same feature is extracted regardless of whether I is warped or not. Introducing the notation F_ϕ for the feature map $f(W_\phi[I])$ extracted from an image warped by ϕ, we can evaluate (3) at $\phi(x)$ to get the form

$$F_\phi(\phi(x)) = F(x). \tag{4}$$

So the feature at position $\phi(x)$ in F_ϕ is the same as the feature at position x in F. When we have a collection of ϕ's forming a group structure, f satisfying (3) is said to to be equivariant. This is however the least interesting form of equivariance where f has to extract the exact same feature for a given image content, regardless of how the image is warped by ϕ. For example, a roto-translation-equivariant keypoint descriptor would describe all rotations of a corner patch with the same description according to (3). For this reason, descriptors satisfying (3) for the group of rotations and translations are sometimes also called invariant [10]. In [9] it was found that this leads to non-discriminative descriptions on upright images, and the solution was to allow more general transformations of the features using *steerers*. Steerers are group representations, a concept from abstract algebra which we introduce next.

Definition 1 (Group representation). *Given a group G and a vector space V, a group representation of G on V is a map $\rho : G \to \mathrm{GL}(V)$ such that*

$$\rho(g_2 g_1) = \rho(g_2)\rho(g_1) \quad \text{for all } g_1, g_2 \in G. \tag{5}$$

Here, $\mathrm{GL}(V)$ is the general linear group of V, *i.e.*, the group of all invertible linear maps $V \to V$ with composition as group operation. For instance, we write $\mathrm{GL}(\mathbb{R}^n) = \mathrm{GL}(n)$ for the group of invertible $n \times n$ matrices. A group representation of G on $\mathbb{R}^n$ is a map from G to invertible $n \times n$ matrices such that the group operation in G corresponds to matrix multiplication in $\mathrm{GL}(n)$. Equipped with this formalism, we can now define steerers.

[3] In practice, image warps corresponding to camera motions are piecewise continuous and piecewise differentiable. The discontinuities stem from motion boundaries, which we will ignore in the theoretical part of this paper.

Definition 2 (Steerer). *Let a set Φ of image transformations and the corresponding group $G < \mathrm{GL}(2)$ of local affine transformations $M(x)$ be given. For a feature extractor f mapping images $I : \mathbb{R}^2 \to \mathbb{R}^3$ to feature maps $F : \mathbb{R}^2 \to \mathbb{R}^c$, a* steerer *is a group representation ρ of G on $\mathbb{R}^c$ such that f becomes* equivariant *w.r.t. W_ϕ and ρ,* i.e.*,*

$$f(W_\phi[I]) = \rho(W_\phi[M])W_\phi[f(I)]. \tag{6}$$

Using the notation F_ϕ for the feature map $f(W_\phi[I])$ extracted from an image warped by ϕ, we can evaluate (6) at coordinate $\phi(x)$ as

$$F_\phi(\phi(x)) = \rho(M(x))F(x). \tag{7}$$

In words, the feature $F_\phi(\phi(x))$ describing the warped image at $\phi(x)$, is equal to the feature describing the original image at x, *i.e.* $F(x)$, up to multiplication by the steerer $\rho(M(x))$. The steerer compensates for the local image transformation $M(x)$ from coordinate x to $\phi(x)$. We call ρ a steerer even if (6) only holds approximately, as this will be the case in practice.

The reader should contrast (3) and (4) with (6) and (7). The more general (6) and (7) allow features to change when the image content is warped, but they must change in a predictible manner, specifically they must change by the linear map $\rho(M)$, *i.e.*, the steerer. Our definition of steerers generalizes the one in [9], from Φ only containing roto-translations, to arbitrary differentiable image transformations. Utilizing steerers, Bökman *et al.* [9] successfully trained rotation equivariant keypoint descriptors that match or surpass the performance of non-equivariant keypoint descriptors on upright images, while also facilitating rotation invariant matching. We will only be concerned with the case $G = \mathrm{GL}(2)$ in this paper. Specifically, having access to a steerer means that one can compute what the features $f(W_\phi[I])$ for a warped image according to any transformation ϕ would be, given only the features $f(I)$ from the original image. This enables test time augmentation in feature space, *i.e.*, computing features for multiple warps W_ϕ while only running the feature extractor f once.

3 Method

We outline the relevant mathematical theory for GL(2)-steerers in Section 3.1, followed by an a delineation from rotation steerers in Section 3.2 and an explanation of their integration with the keypoint descriptor neural network in Section 3.3. Subsequent sections, from Section 3.4 to Section 3.6, detail the comprehensive training steps for the entire pipeline.

3.1 Representation theory of GL(2)

In this paper, keypoint descriptions will be vectors in $\mathbb{R}^{256}$, which we will call description space. The aim of this section is to explain how we can build a steerer

ρ (Definition 2) for GL(2) on description space. From now we will write elements of GL(2) as matrices

$$M = \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}, \quad \text{s.t. } \alpha\delta - \beta\gamma \neq 0. \tag{8}$$

Understanding what representations ρ for GL(2) can look like is a well studied problem in mathematical representation theory, presented for instance in [63, Chapter 4]. We refer to [63] for details and will here limit ourselves to explaining the way of constructing representations ρ that we will use in the experiments.

Example 1 (Trivial representation). The simplest form of ρ is the constant map $\rho(M) = 1$, which trivially satisfies (5) and is a one-dimensional representation.

Example 2 (Determinant representation). Slightly more involved, we can construct representations using the determinant of M. For instance,

$$\rho(M) = \det(M)^n \quad \text{and} \quad \rho(M) = |\det(M)|^\xi \tag{9}$$

satisfy (5) for any choice of $n \in \mathbb{N}$ and $\xi \in \mathbb{R}$.

Example 3 (Standard representation). A straightforward way to construct a representation of GL(2) on $\mathbb{R}^2$ is to take the matrix itself: $\rho(M) = M$.

The easiest way to construct representations of larger dimensions is to stack smaller representations into block-diagonals. We denote by $X \oplus Y$ the block diagonal matrix with blocks X and Y. So, for instance, $\rho(M) = \bigoplus_{i=1}^{128} M$ is a representation on $\mathbb{R}^{256}$. It is also easy to see that any change of basis of a representation defines a new representation: $\tilde{\rho}(M) = Q^{-1}\rho(M)Q$, where Q is some invertible matrix. In our experiments we want to use a broad range of representations, and the idea will be to build them out of so called irreducible representations, irreps[4]. Given a set of irreps ρ_i we will form larger representations on description space by $\rho(M) = Q^{-1}(\bigoplus_{i=1}^{n} \rho_i(M))Q$.

To find irreps of GL(2), one can look at vector spaces of homogeneous polynomials in two variables. The following example demonstrates the general principle.

Example 4 (Homogeneous quadratics). Consider the vector space $\mathcal{H}_2$ of homogeneous quadratic polynomials in two variables x, y. It has a basis consisting of the monomials $y^2, 2xy, x^2$ and so any $q \in \mathcal{H}_2$ can be written

$$q(x, y) = a_0 y^2 + a_1 2xy + a_2 x^2 \tag{10}$$

[4] An irrep on V is a representation that does not leave any proper subspace $W \subset V$ invariant. Irreps can be thought of as fundamental building blocks of representations as many general representations can be decomposed into irreps. However, for GL(2), not all representations can be built out of its irreps. The standard counterexample is $\rho(M) = \left(\begin{smallmatrix} 1 & \log|\det M| \\ 0 & 1 \end{smallmatrix}\right)$ [63, Example 4.11].

with some coefficients $a_0, a_1, a_2 \in \mathbb{R}$. The reader will note that $\mathcal{H}_2$ is isomorphic to $\mathbb{R}^3$, since polynomials q are in bijection with triples (a_0, a_1, a_2). The reason to think about these triples as polynomials is that we can in a natural way find a representation of GL(2) on $\mathcal{H}_2$. We define the representation ρ_2 on $\mathcal{H}_2$ by

$$(\rho_2(M)q)(x, y) = q((x, y)M), \tag{11}$$

where $(x, y)M$ denotes multiplying the row-vector (x, y) by the matrix M. This satisfies (5) as

$$\begin{aligned}(\rho_2(M_2M_1)q)(x, y) &= q((x, y)(M_2M_1)) = q(((x, y)M_2)M_1) \\ &= (\rho_2(M_1)q)((x, y)M_2) = (\rho_2(M_2)(\rho_2(M_1)q))(x, y).\end{aligned} \tag{12}$$

We can rewrite ρ_2 as a representation on $\mathbb{R}^3$ by considering how the coefficients of q change under (11):

$$\begin{aligned}(\rho_2(M)q)(x, y) &= q((x, y)M) = q(\alpha x + \gamma y, \beta x + \delta y) \\ &= a_0(\beta x + \delta y)^2 + a_1 2(\alpha x + \gamma y)(\beta x + \delta y) + a_2(\alpha x + \gamma y)^2 \\ &= \left(\begin{pmatrix} \delta^2 & 2\gamma\delta & \gamma^2 \\ \beta\delta & \alpha\delta + \beta\gamma & \alpha\gamma \\ \beta^2 & 2\alpha\beta & \alpha^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix}\right) \cdot \begin{pmatrix} y^2 \\ 2xy \\ x^2 \end{pmatrix}.\end{aligned} \tag{13}$$

So we have found the action of ρ_2 on the polynomial coefficients $(a_0, a_1, a_2)^T \in \mathbb{R}^3$ and can by slight abuse of notation write $\rho_2(M)$ on matrix form as

$$\rho_2(M) = \begin{pmatrix} \delta^2 & 2\gamma\delta & \gamma^2 \\ \beta\delta & \alpha\delta + \beta\gamma & \alpha\gamma \\ \beta^2 & 2\alpha\beta & \alpha^2 \end{pmatrix}. \tag{14}$$

Constructing higher dimensional irreps can be done by following the recipe outlined in Example 4. The $(n + 1)$-dimensional vector space of homogeneous polynomials of degree n

$$\mathcal{H}_n = \text{span}\left\{\binom{n}{k} x^k y^{n-k}\right\}_{k=0}^{n} \tag{15}$$

has an associated representation ρ_n of GL(2) defined by (11). $\rho_n(M)$ can be written on matrix form similar to (14), where each entry of the matrix is a homogeneous degree n polynomial in $\alpha, \beta, \gamma, \delta$. A formula for $\rho_n(M)$ is given by [63, (2.6)].

More general representations can be obtained by multiplying with determinant based representations and we define

$$\rho_{n,\xi}(M) = |\det(M)|^{\xi - \frac{n}{2}} \rho_n(M). \tag{16}$$

where the power $-\frac{n}{2}$ is introduced to normalize the determinant of $\rho_{n,0}$ to ± 1. These $\rho_{n,\xi}$ are irreps and form the basic building blocks of our steerers. The

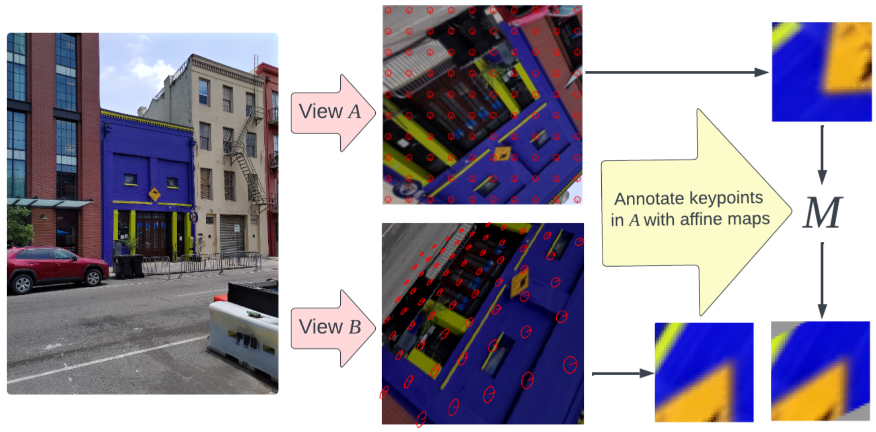

Fig. 3: Illustration of the data generation pipeline. Given two views A, B of a scene, we compute the image warp from one to the other and use it to both find corresponding keypoints and annotate the keypoints in A with affine transformations M that locally approximate the image warp from A to B. In the illustration above, we show two views obtained as homographies of a single image, as used in the pretraining step described in Section 3.5, but it is also possible to obtain the two views, *e.g.*, by taking two photos of the same location as is the case when we train on MegaDepth (Section 3.4). The red circles in A are warped to the ellipses in B, and we illustrate the obtained affine map M for one keypoint pair on the right hand side of the figure. In practice we use DeDoDe keypoints but here we illustrate with a regular grid in A.

steerers we use in the experiments are formed as

$$\rho(M) = Q^{-1}\left(\bigoplus_j \rho_{n_j,\xi_j}(M)\right)Q, \tag{17}$$

where $Q \in \mathbb{R}^{256\times 256}$ is a learnt change of basis matrix initialized to identity, ξ_j are learnt determinant scalings and $n_j \in \{0, 1, 2, 3, 4\}$ are chosen to get an equal distribution of dimensions of the different orders $0, \ldots, 4$. The cutoff at 4 is motivated by the fact that it covers rotation frequencies up to 4 when a rotation R is fed into ρ, and higher frequencies were not useful in prior work [9, 24].

3.2 Comparison to rotation steerers

If we input a rotation matrix R into (17), we obtain a steerer like those considered in [9]. In that case, $\det(R) = 1$ so the determinant scaling does not affect the steerer. The main novelty in this work is that we can now steer the much larger class of local transformations GL(2). (17) specifies a structured way for the descriptor to respond to affine transformations, where the determinant scaling captures how much the descriptor varies with the scale of the transform M.

3.3 Max similarity matcher revisited

Given two descriptions $d_1, d_2 \in \mathbb{R}^{256}$, we want to be able to say how close they are to each other in order to determine correspondences between keypoints in different images. The similarity measure used in [9, 22] is the cosine similarity. This similarity is however unsuited for affinely steered descriptions, as the steerer (17) can change the norm of a description. We therefore use Euclidean norm instead and measure the similarity of d_1 and d_2 by $-\|d_1 - d_2\|$. In practice, given two images A and B, we find K keypoints in each image, describe them with descriptions and measure all pairwise similarities $S \in \mathbb{R}^{K\times K}$. We get matches from the pairwise similarity matrix S by taking mutual nearest neighbours and putting a minimum threshold on the dual softmax score of S.

The similarity of descriptions can be more robustly computed when a steerer is available. In [9], several methods were suggested, one of which is the *max similarity* method. The similarity between d_1 and d_2 is then

$$\max_{M\in\mathcal{M}} -\|\rho(M)d_1 - d_2\|, \tag{18}$$

where $\mathcal{M}$ is a set of transformations that can be selected to balance robustness over a range of transformations with inference speed and discriminability. We propose to learn $\mathcal{M}$ by backpropagating through (18) during finetuning (Section 3.6), although this limits us to a small set $\mathcal{M}$ for computational reasons.

3.4 Training the descriptor and steerer on MegaDepth

We follow [9] and train the descriptor network f and steerer ρ (17) jointly. We train on MegaDepth [45] similar to [9, 22], but in contrast to them use heavy affine augmentations of the images. Given an image pair A, B, we detect K DeDoDe keypoints [22] $x_{A,i}$ $(i = 1, \dots, K)$, resp. $x_{B,j}$ $(j = 1, \dots, K)$ per image, and obtain descriptions $d_{A,i}$, resp. $d_{B,j}$ by evaluating the extracted feature maps $f(A), f(B)$ at the keypoint locations. Ground truth matches between keypoints in the two images are obtained as in [22]: by using the known image depths and relative pose we warp the keypoints from A to B and from B to A and take matches as all mutual closest neighbours with distances below 0.5% of the image width. For the ground truth matches, we also compute local affine maps M_i from A to B by warping a small octagon of points around each keypoint in A to B and taking the least squares estimated affine map from the eight points in A to the eight points in B. The local affine maps M_i are used to steer the descriptions in A during training in order to align them with the ones in B. We also estimate a global affine M by taking the least squares estimate of all ground truth matches. This M is used to steer all descriptions in A which do not have a match in B. In summary, during training, the similarity between a description $d_{A,i}$ in A and one $d_{B,j}$ in B is

$$S_{ij} = -\|\rho(M_i)d_{A,i} - d_{B,j}\|. \tag{19}$$

We form the similarity matrix $S \in \mathbb{R}^{K\times K}$ by considering all pairs of descriptions in A and B and compute the loss as in [22] by taking the negative dual log-softmax of S and taking the mean over the ground truth matches.

3.5 Pretraining on homographies

Training or pretraining descriptors and matchers on homographies is a common strategy [18, 29, 46, 71]. The idea is to take a single image and match it with a homography warped (and photometrically augmented) version of itself. Homography pretraining is a good fit for affine steering, since a homography applied to an image induces an affine transform between each point in the original image and the corresponding point in the warped image. We visualize the data generation pipeline in Figure 3.

We find that pretraining on homographies is not very helpful for affine steerers trained by augmentations of MegaDepth. However, pretraining on homographies gives a good starting point for the fine-tuning approach explained next.

3.6 Fine-tuning on upright images

We propose a way to fine-tune a descriptor and steerer jointly to obtain descriptions tailored for upright images. We train through the *max similarity* score (18), where we use a set of only three affine transformations $\mathcal{M} = \{I, M_1, M_2\}$, letting M_1, M_2 be learnable "prototypes". The aim is that the learnable M_1, M_2 will contain affine transformations such that the net does not have to become invariant under M_1, M_2 while still enabling matching that is invariant under M_1, M_2 through (18). Note that in contrast to the earlier described training approaches with the loss computed from (19) we do not enforce a connection between M_1, M_2 and affine warps of the image at this stage. Hence there are no guarantees that steering by an $M \in \mathrm{GL}(2)$ will correspond to warping the input image by M any longer. For instance, steering by M could correspond to warping the input image by $\tilde{M}M\tilde{M}^{-1}$ for some $\tilde{M}$, which would mean that the steerer that actually corresponds to warping by M is $\tilde{\rho}(M) = \rho(\tilde{M})^{-1}\rho(M)\rho(\tilde{M})$. Even more generally, the steerer could learn to incorporate other image transformations than affine warps into the learnt prototypes M_1, M_2. We show numerically how well the steerer works for affine warps after fine-tuning in Section 4.1.

4 Experiments

Experiment details and ablations can be found in the appendix. We follow [22] and consider two descriptor networks, a B-model with a VGG-19 backbone [74] and a larger G-model with a DINOv2 ViT backbone [19, 65]. In the appendix we also present an experiment with the smaller XFeat descriptor [67].

For both B- and G-networks, we have two versions. One version is optimized for upright images by pretraining on homographies (Section 3.5) and fine-tuning on upright MegaDepth pairs (Section 3.6). These models are called *AffSteer-B* and *AffSteer-G*. During inference, we use the set of three affine transformations $\mathcal{M}$ learnt during training for the *max similarity* matching method (18).

The second version is trained on Megadepth pairs that are affinely augmented (Section 3.4). This version is not as good on upright images, but can be steered

Table 1: Megadepth-1500. Relative pose estimation, measured in AUC (higher is better).

Method ↓ AUC →	@5°	@10°	@20°
SuperPoint [18] CVPRW'18	31.7	46.8	60.1
DISK [84] NeurIPS'20	36.7	52.9	65.9
ALIKED [91] IEEE-TIM'23	41.9	58.4	71.7
SiLK [29] ICCV'23	39.9	55.1	66.9
DeDoDe-B [22] 3DV'24	49.4	65.5	77.7
Steerers-B-C4-Perm [9]	51	67	79
AffEqui-B	46.1	62.4	74.8
AffSteer-B	**52.7**	**68.9**	**81.0**
DeDoDe-G [22] 3DV'24	52.8	69.7	82.0
Steerers-G-C4-Perm [9]	52	69	81
AffEqui-G	50.6	67.2	80.1
AffSteer-G	**53.7**	**70.0**	**82.1**

Table 2: WxBS [58]. Fundamental matrix estimation, measured in mAA at 10px (higher is better).

Method ↓ mAA@ →	10px
DISK [84] NeurIPS'20	35.5
DeDoDe-B [22] 3DV'24	45.9 ± 1.2
Steerers-B-C4-Perm [9]	**51.0** ± 1.5
AffSteer-B	50.0 ± 1.4
DeDoDe-G [22] 3DV'24	**57.7** ± 1.1
Steerers-G-C4-Perm [9]	57.0 ± 1.4
AffSteer-G	57.3 ± 1.2

with affine warps and is more equivariant. These models are called *AffEqui-B* and *AffEqui-G*. We compare the equivariance and steering capabilities of the versions in Section 4.1.

We evaluate on several common benchmarks for upright image matching. Here we consistently set new state-of-the-art results for detector-descriptor based methods. Dense methods such as the SotA [23] are heavier and take two images jointly as input, allowing them to compute features in each image conditioned on the other image. Descriptor based methods cannot do this and so we don't compare to dense methods here. Generally, we divide the tables into three sections of first works not based on DeDoDe [22], then methods with the same network architecture as DeDoDe-B and finally methods with the same network architecture as DeDoDe-G.

MegaDepth-1500 relative pose. We follow LoFTR [76] and evaluate on a held out part of MegaDepth. This is a standard benchmark for matching upright images. We present results in Table 1, where we see consistent improvements for the upright optimized *AffSteer*-models, while the *AffEqui*-models perform competitively with previous approaches.

WxBS Fundamental matrix estimation. Results for the WxBS benchmark [58] and are shown in Table 4. Here we find that the results fluctuate a lot due to RANSAC and dual-soft-max thresholds. We report results with the best found thresholds for each method and mean and standard deviation over 10 evaluations.

Image Matching Challenge 2022. Results for the Image Matching Challenge 2022 [36] are presented in Table 3. Remarkably, we obtain a large improvement for *B*-size variants, even outperforming all *G*-size variants with *AffSteer-B*. As

Table 3: IMC2022 [36]. Relative pose on the hidden test set, measured in mAA (higher is better).

Method ↓ mAA →	@10
DISK [84] NeurIPS'20	64.8
ALIKED [91] IEEE-TIM'23	64.9
SiLK [29] ICCV'23	68.5
DeDoDe-B [22] 3DV'24	72.9
Steerers-B-C4-Perm [9] CVPR'24	73.4
AffSteer-B	**77.3**
DeDoDe-G [22] 3DV'24	75.8
Steerers-G-C4-Perm [9] CVPR'24	75.5
AffSteer-G	**76.8**

Table 4: HPatches [1] Homography. Corner error, measured in AUC (higher is better).

Method ↓ AUC@ →	3px	5px	10px
DISK [84] NeurIPS'20	60.3	71.4	81.8
ALIKED [91] IEEE-TIM'23	61.6	73.1	83.5
SiLK [29] ICCV'23	66	-	-
DeDoDe-B [22] 3DV'24	68.2	77.9	86.4
Steerers-B-C4-Perm [9] CVPR'24	69.5	78.7	87.0
AffSteer-B	**70.1**	**79.1**	**87.3**
DeDoDe-G [22] 3DV'24	67.1	77.3	86.3
Steerers-G-C4-Perm [9] CVPR'24	66.8	76.7	85.9
AffSteer-G	**68.2**	**78.1**	**86.9**

Table 5: AIMS [75]. Measured in average precision (higher is better). The subset "North Up", contains image pairs with small relative rotation and "All Others" contains image pairs with large relative rotations. "All" includes all image pairs.

Matching method	Descriptor	North Up	All Others	All
Max similarity 8	Steerers-B-SO2-Spread [9]	60	57	58
Procrustes	Steerers-B-SO2-Freq1 [9]	**64**	**59**	**60**
Max similarity 8	*AffEqui-B*	59	58	59
Max similarity 12	*AffEqui-B*	61	**59**	**60**

the test set is hidden, we unfortunately cannot offer an explanation as to what types of image pairs we are handling better than previous works.

HPatches Homography Estimation. We evaluate on the HPatches Homography benchmark [1, 76]. We present results in Table 4. Interestingly, B-size DeDoDe variants do better on homographies than G-size methods generally.

Astronaut Image Matching Challenge (AIMS). To check how well our equivariant models handle rotations, we compare against rotation steerers [9] on AIMS [75]. Here we test a *max similarity* matcher with 8 or 12 rotations and use the *AffEqui-B* model, since [9] also use a B-model in their evaluation. We show competitive performance even though our method is not specialized for rotations. In particular, we cannot use the Procrustes matcher proposed in [9], but show that we can match its performance by increasing the amount of rotations in the max similarity matcher.

4.1 Evaluating the steering capabilities

To evaluate how well the steerer works for affine steering, we take a subset of MegaDepth images and investigate how many matches are obtained when

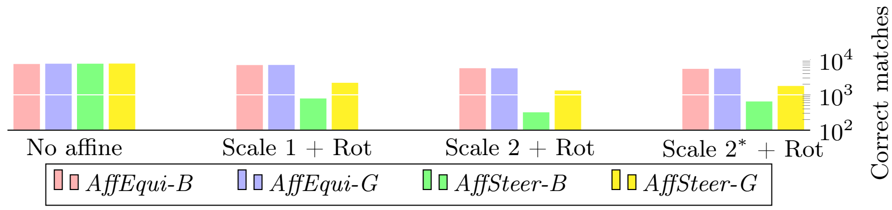

Fig. 4: Measuring the affine steering capabilities of the different descriptors. We count correct matches when using the steerer with the ground truth affine map on affinely augmented images. We vary the affine maps by scale. "+ Rot" means that all inplane rotations are included. "Scale 2*" means that the scale of the affine map M is maximum 2, but the matrix that is fed to the steerer is $M/\sqrt{\det M}$ and so has determinant 1.

matching with an *oracle steerer* to a affinely augmented version of each image. In other words, we measure description similarity by

$$-\|\rho(M_i)d_{A,i} - d_{B,j}\|, \tag{20}$$

where M_i is the correct affine warp. We will measure the number of correct matches across various difficulties of affine warps. If the steerer works well, then we should get similarly many matches in the warped cases as in the case where there is no warp. We show the results in Figure 4. It is clear that the steerer works well for *AffEqui* as expected. To further confirm this, we evaluate *AffEqui* on HPatches with an oracle steerer that steers the descriptions with the correct $M_i \in \mathrm{GL}(2)$. The results are (71.2, 80.8, 89.2) AUC@(3px, 5px, 10px) for *AffEqui-B* and (70.9, 80.1, 88.7) for *AffEqui-G*. These numbers are comparable to using the SotA dense matcher RoMa [23], which achieves (71.3, 80.6, 88.5). While the comparison is unfair in the sense that we have access to the correct M_i, this still indicates a large potential of affinely equivariant descriptors. We hope that it will be possible in the future to leverage this potential.

As seen in Figure 4, for *AffSteer-G* the steerer still works quite well, but the steering of scale seems to have been destroyed by the fine-tuning, since we get more matches when normalizing the correct affine map M to unit determinant before feeding it into the steerer. For *AffSteer-B* the steerer does not work well. A possible explanation is given in Section 3.6, the steerer $\rho(M)$ might no longer directly encode an affine image warp by M but some other transformation.

5 Conclusion

We presented a generalization of the steerers framework to locally affine transformations. This led to affinely equivariant descriptors *AffEqui-B*, *AffEqui-G*. These produced results below the state-of-the-art, but showed promise for combining with estimation of local affine transforms, which we explored through the use of an oracle method. Then we proposed a fine-tuning method for upright images, producing the state-of-the-art descriptors *AffSteer-B* and *AffSteer-G*.

Acknowledgements

This work was supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP), funded by the Knut and Alice Wallenberg Foundation and by the strategic research environment ELLIIT, funded by the Swedish government. The computational resources were provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS) at C3SE, partially funded by the Swedish Research Council through grant agreement no. 2022-06725, and by the Berzelius resource, provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre.

References

1. Balntas, V., Lenc, K., Vedaldi, A., Mikolajczyk, K.: HPatches: A benchmark and evaluation of handcrafted and learned local descriptors. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5173–5182 (2017)
2. Balntas, V., Riba, E., Ponsa, D., Mikolajczyk, K.: Learning local feature descriptors with triplets and shallow convolutional neural networks. In: Bmvc (2016)
3. Barath, D., Mishkin, D., Cavalli, L., Sarlin, P.E., Hruby, P., Pollefeys, M.: Affineglue: Joint matching and robust estimation. arXiv preprint arXiv:2307.15381 (2023)
4. Barath, D., Polic, M., Förstner, W., Sattler, T., Pajdla, T., Kukelova, Z.: Making affine correspondences work in camera geometry computation. In: European Conf. Computer Vision (ECCV). p. 723–740 (2020)
5. Barroso-Laguna, A., Riba, E., Ponsa, D., Mikolajczyk, K.: Key.Net: Keypoint detection by handcrafted and learned cnn filters. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5836–5844 (2019)
6. Bentolila, J., Francos, J.M.: Conic epipolar constraints from affine correspondences. Computer Vision and Image Understanding (CVIU) **122**, 105–114 (2014)
7. Bruintjes, R.J., Motyka, T., van Gemert, J.: What affects learned equivariance in deep image recognition models? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 4838–4846 (June 2023)
8. Brynte, L., Iglesias, J.P., Olsson, C., Kahl, F.: Learning structure-from-motion with graph attention networks. In: IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2024)
9. Bökman, G., Edstedt, J., Felsberg, M., Kahl, F.: Steerers: A framework for rotation equivariant keypoint descriptors. In: IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2024)
10. Bökman, G., Kahl, F.: A case for using rotation invariant features in state of the art feature matchers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5110–5119 (2022)
11. Bökman, G., Kahl, F.: Investigating how ReLU-networks encode symmetries. In: Thirty-seventh Conference on Neural Information Processing Systems (2023), `https://openreview.net/forum?id=8lbFwpebeu`
12. Bökman, G., Kahl, F., Flinth, A.: ZZ-Net: A universal rotation equivariant architecture for 2d point clouds. In: IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2022)

13. Cao, C., Fu, Y.: Improving transformer-based image matching by cascaded capturing spatially informative keypoints. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 12129–12139 (October 2023)
14. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021)
15. Chen, H., Luo, Z., Zhang, J., Zhou, L., Bai, X., Hu, Z., Tai, C.L., Quan, L.: Learning to match features with seeded graph matching network. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6301–6310 (2021)
16. Cohen, T., Welling, M.: Group equivariant convolutional networks. In: International conference on machine learning. pp. 2990–2999. PMLR (2016)
17. Cohen, T.S., Welling, M.: Transformation properties of learned visual representations. ICLR 2015 (arXiv:1412.7659) (2014)
18. 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. pp. 224–236 (2018)
19. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021), `https://openreview.net/forum?id=YicbFdNTTy`
20. Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., Sattler, T.: D2-Net: A Trainable CNN for Joint Detection and Description of Local Features. In: Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2019)
21. Edstedt, J., Athanasiadis, I., Wadenbäck, M., Felsberg, M.: DKM: Dense kernelized feature matching for geometry estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (2023)
22. Edstedt, J., Bökman, G., Wadenbäck, M., Felsberg, M.: DeDoDe: Detect, Don't Describe – Describe, Don't Detect for Local Feature Matching. In: 2024 International Conference on 3D Vision (3DV). IEEE (2024)
23. Edstedt, J., Sun, Q., Bökman, G., Wadenbäck, M., Felsberg, M.: RoMa: Robust dense feature matching. In: IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2024)
24. Felsberg, M., Sommer, G.: Image features based on a new approach to 2d rotation invariant quadrature filters. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) Computer Vision — ECCV 2002. pp. 369–383. Springer Berlin Heidelberg, Berlin, Heidelberg (2002)
25. Forssén, P.E., Lowe, D.G.: Shape descriptors for maximally stable extremal regions. In: 2007 IEEE 11th International Conference on Computer Vision. pp. 1–8. IEEE (2007)
26. Garrido, Q., Assran, M., Ballas, Nicolas Bardes, A., Najman, L., LeCun, Y.: Learning and leveraging world models in visual representation learning. arXiv preprint arXiv:2403.00504 (2024)
27. Garrido, Q., Najman, L., Lecun, Y.: Self-supervised learning of split invariant equivariant representations. In: International Conference on Machine Learning. PMLR (2023)
28. Giang, K.T., Song, S., Jo, S.: TopicFM: Robust and interpretable topic-assisted feature matching. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37 (2023)

29. Gleize, P., Wang, W., Feiszli, M.: SiLK: Simple Learned Keypoints. In: ICCV (2023)
30. Grill, J.B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al.: Bootstrap your own latent-a new approach to self-supervised learning. Advances in neural information processing systems **33**, 21271–21284 (2020)
31. Gruver, N., Finzi, M.A., Goldblum, M., Wilson, A.G.: The lie derivative for measuring learned equivariance. In: The Eleventh International Conference on Learning Representations (2023), `https://openreview.net/forum?id=JL7Va5Vy15J`
32. Gupta, S., Robinson, J., Lim, D., Villar, S., Jegelka, S.: Structuring representation geometry with rotationally equivariant contrastive learning. arXiv preprint arXiv:2306.13924 (2023)
33. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: 2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06). vol. 2, pp. 1735–1742. IEEE (2006)
34. Han, J., Ding, J., Xue, N., Xia, G.S.: Redet: A rotation-equivariant detector for aerial object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2786–2795 (2021)
35. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 16000–16009 (June 2022)
36. Howard, A., Trulls, E., Yi, K.M., Mishkin, D., Dane, S., Jin, Y.: Image matching challenge 2022 (2022), `https://kaggle.com/competitions/image-matching-challenge-2022`
37. Huang, D., Chen, Y., Liu, Y., Liu, J., Xu, S., Wu, W., Ding, Y., Tang, F., Wang, C.: Adaptive assignment for geometry aware local feature matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5425–5434 (2023)
38. Jonsson, E., Felsberg, M.: Efficient computation of channel-coded feature maps through piecewise polynomials. Image and Vision Computing **27**(11), 1688–1694 (2009)
39. Koyama, M., Fukumizu, K., Hayashi, K., Miyato, T.: Neural fourier transform: A general approach to equivariant representation learning. In: The Twelfth International Conference on Learning Representations (2024), `https://openreview.net/forum?id=eOCvA8iwXH`
40. Lawrence, H., Harris, M.T.: Learning polynomial problems with $sl(2,\mathbb{R})$ equivariance. In: The Twelfth International Conference on Learning Representations (2023)
41. Lee, J., Jeong, Y., Cho, M.: Self-supervised learning of image scale and orientation. In: 31st British Machine Vision Conference 2021, BMVC 2021, Virtual Event, UK. BMVA Press (2021), `https://www.bmvc2021-virtualconference.com/programme/accepted-papers/`
42. Lee, J., Kim, B., Cho, M.: Self-supervised equivariant learning for oriented keypoint detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4847–4857 (2022)
43. Lee, J., Kim, B., Kim, S., Cho, M.: Learning rotation-equivariant features for visual correspondence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21887–21897 (2023)
44. Lenc, K., Vedaldi, A.: Understanding image representations by measuring their equivariance and equivalence. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2015)

45. Li, Z., Snavely, N.: Megadepth: Learning single-view depth prediction from internet photos. In: IEEE Conf. Computer Vision and Pattern Recognition (CVPR). pp. 2041–2050 (2018)
46. Lindenberger, P., Sarlin, P.E., Pollefeys, M.: LightGlue: Local Feature Matching at Light Speed. In: IEEE Int'l Conf. Computer Vision (ICCV) (2023)
47. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int'l J. Computer Vision (IJCV) **60**, 91–110 (2004)
48. Luo, Z., Shen, T., Zhou, L., Zhang, J., Yao, Y., Li, S., Fang, T., Quan, L.: Contextdesc: Local descriptor augmentation with cross-modality context. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2527–2536 (2019)
49. MacDonald, L.E., Ramasinghe, S., Lucey, S.: Enabling equivariance for arbitrary lie groups. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8183–8192 (June 2022)
50. Mao, R., Bai, C., An, Y., Zhu, F., Lu, C.: 3DG-STFM: 3d geometric guided student-teacher feature matching. In: Proc. European Conference on Computer Vision (ECCV) (2022)
51. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. Image and vision computing **22**(10), 761–767 (2004)
52. Matas, J., Obdrzalek, T., Chum, O.: Local affine frames for wide-baseline stereo. In: 2002 International Conference on Pattern Recognition. vol. 4, pp. 363–366. IEEE (2002)
53. Melnyk, P., Felsberg, M., Wadenbäck, M.: Steerable 3D spherical neurons. In: Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., Sabato, S. (eds.) Proceedings of the 39th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 162, pp. 15330–15339. PMLR (17–23 Jul 2022), `https://proceedings.mlr.press/v162/melnyk22a.html`
54. Mikolajczyk, K., Schmid, C.: Scale & affine invariant interest point detectors. Int'l J. Computer Vision (IJCV) **60**, 63–86 (2004)
55. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Trans. Pattern Analysis and Machine Intelligence (T-PAMI) **27**(10), 1615–1630 (2005)
56. Mironenco, M., Forré, P.: Lie group decompositions for equivariant neural networks. In: The Twelfth International Conference on Learning Representations (2024), `https://openreview.net/forum?id=p34fRKp8qA`
57. Mishchuk, A., Mishkin, D., Radenovic, F., Matas, J.: Working hard to know your neighbor's margins: Local descriptor learning loss. Advances in neural information processing systems **30** (2017)
58. Mishkin, D., Matas, J., Perdoch, M., Lenc, K.: Wxbs: Wide baseline stereo generalizations. arXiv preprint arXiv:1504.06603 (2015)
59. Mishkin, D., Radenovic, F., Matas, J.: Repeatability is not enough: Learning affine regions via discriminability. In: Proceedings of the European conference on computer vision (ECCV). pp. 284–300 (2018)
60. Mishkin, D., Radenović, F., Matas, J.: Repeatability is not enough: Learning affine regions via discriminability. In: European Conf. Computer Vision (ECCV). p. 287–304 (2018)
61. Ni, J., Li, Y., Huang, Z., Li, H., Bao, H., Cui, Z., Zhang, G.: Pats: Patch area transportation with subdivision for local feature matching. In: The IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR) (2023)

62. Obdržálek, Š., Matas, J.: Local affine frames for image retrieval. In: International Conference on Image and Video Retrieval. pp. 318–327. Springer (2002)
63. Olver, P.J.: Classical invariant theory. No. 44 in London Mathematical Society Student Texts, Cambridge University Press (1999)
64. Olver, P.J., Qu, C., Yang, Y.: Feature Matching and Heat Flow in Centro-Affine Geometry. SIGMA. Symmetry, Integrability and Geometry: Methods and Applications **16**, 093 (2020). https://doi.org/10.3842/SIGMA.2020.093, https://www.emis.de/journals/SIGMA/2020/093/
65. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Howes, R., Huang, P.Y., Xu, H., Sharma, V., Li, S.W., Galuba, W., Rabbat, M., Assran, M., Ballas, N., Synnaeve, G., Misra, I., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: DINOv2: Learning robust visual features without supervision. arXiv:2304.07193 (2023)
66. Park, J.Y., Biza, O., Zhao, L., van de Meent, J.W., Walters, R.: Learning symmetric embeddings for equivariant world models. arXiv preprint arXiv:2204.11371 (2022)
67. Potje, G., Cadar, F., Araujo, A., Martins, R., Nascimento, E.R.: Xfeat: Accelerated features for lightweight image matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2682–2691 (2024)
68. Revaud, J., De Souza, C., Humenberger, M., Weinzaepfel, P.: R2d2: Reliable and repeatable detector and descriptor. Advances in Neural Information Processing Systems (NeurIPS) **32** (2019)
69. Santellani, E., Sormann, C., Rossi, M., Kuhn, A., Fraundorfer, F.: S-trek: Sequential translation and rotation equivariant keypoints for local feature extraction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9728–9737 (2023)
70. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperGlue: Learning feature matching with graph neural networks. In: IEEE Conf. Computer Vision and Pattern Recognition (CVPR) (2020)
71. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4938–4947 (2020)
72. Shakerinava, M., Mondal, A.K., Ravanbakhsh, S.: Structuring representations using group invariants. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) Advances in Neural Information Processing Systems. vol. 35, pp. 34162–34174. Curran Associates, Inc. (2022), https://proceedings.neurips.cc/paper_files/paper/2022/file/dcd297696d0bb304ba426b3c5a679c37-Paper-Conference.pdf
73. Shi, Y., Cai, J.X., Shavit, Y., Mu, T.J., Feng, W., Zhang, K.: Clustergnn: Cluster-based coarse-to-fine graph neural network for efficient feature matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12517–12526 (2022)
74. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015)
75. Stoken, A., Fisher, K.: Find my astronaut photo: Automated localization and georectification of astronaut photography. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 6196–6205 (June 2023)

76. Sun, J., Shen, Z., Wang, Y., Bao, H., Zhou, X.: LoFTR: Detector-free local feature matching with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8922–8931 (2021)
77. Tian, Y., Barroso Laguna, A., Ng, T., Balntas, V., Mikolajczyk, K.: Hynet: Learning local descriptor with hybrid similarity measure and triplet loss. Advances in neural information processing systems **33**, 7401–7412 (2020)
78. Tian, Y., Yu, X., Fan, B., Wu, F., Heijnen, H., Balntas, V.: Sosnet: Second order similarity regularization for local descriptor learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11016–11025 (2019)
79. Truong, P., Danelljan, M., Gool, L.V., Timofte, R.: GOCor: Bringing Globally Optimized Correspondence Volumes into Your Neural Network. Advances in Neural Information Processing Systems **33** (2020)
80. Truong, P., Danelljan, M., Timofte, R.: GLU-Net: Global-local universal network for dense flow and correspondences. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6258–6268 (2020)
81. Truong, P., Danelljan, M., Timofte, R., Van Gool, L.: PDC-Net+: Enhanced Probabilistic Dense Correspondence Network. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)
82. Truong, P., Danelljan, M., Van Gool, L., Timofte, R.: Learning accurate dense correspondences and when to trust them. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5714–5724 (2021)
83. TUZNIK, S.L., OLVER, P.J., TANNENBAUM, A.: Equi-affine differential invariants for invariant feature point detection. European Journal of Applied Mathematics **31**(2), 277–296 (2020). `https://doi.org/10.1017/S0956792519000020`
84. Tyszkiewicz, M., Fua, P., Trulls, E.: Disk: Learning local features with policy gradient. Advances in Neural Information Processing Systems (NeurIPS) **33**, 14254–14265 (2020)
85. Wang, Q., Zhang, J., Yang, K., Peng, K., Stiefelhagen, R.: MatchFormer: Interleaving attention in transformers for feature matching. In: Asian Conference on Computer Vision (2022)
86. Weiler, M., Cesa, G.: General e (2)-equivariant steerable cnns. Advances in neural information processing systems **32** (2019)
87. Yan, P., Tan, Y., Xiong, S., Tai, Y., Li, Y.: Learning soft estimator of keypoint scale and orientation with probabilistic covariant loss. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19406–19415 (2022)
88. Yi, K.M., Trulls, E., Lepetit, V., Fua, P.: Lift: Learned invariant feature transform. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14. pp. 467–483. Springer (2016)
89. Yu, G., Morel, J.M.: ASIFT: An algorithm for fully affine invariant comparison. "Image Processing On Line" **1**, 11–38 (2011)
90. Yu, J., Chang, J., He, J., Zhang, T., Yu, J., Feng, W.: ASTR: Adaptive spot-guided transformer for consistent local feature matching. In: The IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR) (2023)
91. Zhao, X., Wu, X., Chen, W., Chen, P.C.Y., Xu, Q., Li, Z.: Aliked: A lighter keypoint and descriptor extraction network via deformable transformation. IEEE Transactions on Instrumentation & Measurement **72**, 1–16 (2023). `https://doi.org/10.1109/TIM.2023.3271000`, `https://arxiv.org/pdf/2304.03608.pdf`

92. Zhao, X., Wu, X., Miao, J., Chen, W., Chen, P.C., Li, Z.: Alike: Accurate and lightweight keypoint detection and descriptor extraction. IEEE Transactions on Multimedia (2022)
93. Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., Kong, T.: Image BERT pre-training with online tokenizer. In: International Conference on Learning Representations (2022), `https://openreview.net/forum?id=ydopy-e6Dg`
94. Zhu, S., Liu, X.: PMatch: Paired masked image modeling for dense geometric matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)