Robust estimation of local affine maps and its applications to image matching

M. Rodríguez[†] G. Facciolo[†] R. Grompone von Gioi[†] P. Musé[§] J. Delon[‡]
† CMLA, ENS Paris-Saclay, France
§ IIE, Universidad de la República, Uruguay ‡ MAP5, Université Paris Descartes, France

Abstract

*The classic approach to image matching consists in the detection, description and matching of keypoints. This defines a zero-order approximation of the mapping between two images, determined by corresponding point coordinates. But the patches around keypoints typically contain more information, which may be exploited to obtain a first-order approximation of the mapping, incorporating local affine maps between corresponding keypoints. In this work, we propose a LOCal Affine Transform Estimator (*LOCATE*) method based on neural networks. We show that* LOCATE *drastically improves the accuracy of local geometry estimation by tracking inverse maps. A second contribution on guided matching and refinement is also presented. The novelty here consists in the use of* LOCATE *to propose new SIFT-keypoint correspondences with precise locations, orientations and scales. Our experiments show that the precision gain provided by* LOCATE *does play an important role in applications such as guided matching. The third contribution of this paper consists in a modification to the RANSAC algorithm, that uses* LOCATE *to improve the homography estimation between a pair of images. These approaches outperform RANSAC for different choices of image descriptors and image datasets, and permit to increase the probability of success in identifying image pairs in challenging matching databases. The source codes are available at:* `https://rdguez-mariano.github.io/pages/locate`.

1. Introduction

A physical object with smooth or piecewise smooth boundary captured by real cameras at different positions undergoes smooth apparent deformations. These regular deformations are locally well approximated by affine transforms of the image plane; indeed, for any smooth deformation, its first order Taylor approximation is an affine map. By focusing on local image regions, or patches, the perspective changes of objects can therefore be modeled by affine image deformations.

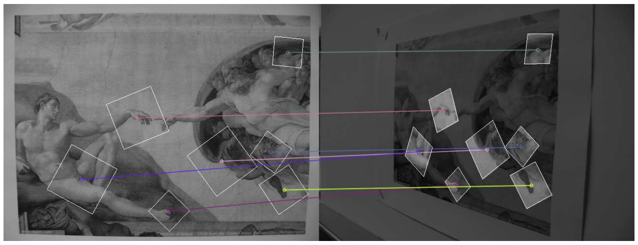


Figure 1. Some correspondences together with local affine maps estimated by the proposed LOCATE network. Patches on the target are warped versions of their corresponding query patch.

This observation has motivated the development of image comparison methods based on local descriptors that are as affine invariant as possible. The problem of constructing affine invariant image descriptors by using an affine Gaussian scale space, which is equivalent to simulating affine distortions followed by the heat equation, has a long history starting with [12, 4, 14, 15]. The idea of affine shape adaptation was used as a basis for the work on affine invariant interest points and affine invariant matching in [15, 3, 18, 19, 42, 41, 40], including the Harris-Affine and Hessian-Affine region detectors [18, 19]. Finally, the detectors MSER (Maximally Stable Extremal Region) [17] and LLD (Level Line Descriptor) [27, 28, 5] both rely on image level lines. Yet, the affine invariance of these descriptors in images acquired with real cameras is limited by the fact that optical blur and affine transforms do not commute, as shown in [26]. To overcome this limitation, the authors of [26] proposed to optically simulate affine transformations. This idea was also exploited in [29, 22, 38, 36] and more recently by the SIFT-AID method [37], which combines SIFT keypoints with a CNN-based patch descriptor trained to capture affine invariance. Another recent possibility to obtain affine invariance is by learning affine-covariant region representations [23], where a patch is normalized before description. The latter method together with the HardNet [21] descriptor was reported to be the state of the art in image matching under strong viewpoint changes for all detectors.

Image matching usually refers to estimating a global ho-

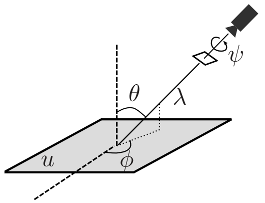


Figure 2. Geometric interpretation of equation (1).

mographic transform between two images. An established approach [10] consists in computing local image matches, which are then aggregated using the RANSAC (RANdom SAmple Consensus) algorithm [9] to estimate an homography. The same procedure is also used for fundamental matrix estimation.

Recently, CNN-based image matching approaches have been proposed for directly estimating global affine and homographic transformations [34, 6]. In [34], the POOL4 layer of the VGG-16 network [39] was used for acquiring features from images and correlation maps fed to a regression network that outputs the best affine transform that fits the query to the target image. In a direct approach, the authors of [6] trained a network to estimate the homography relating the query to the target image. Both [34, 6] were trained on synthetically generated images, but neither of them took into account the blur caused by camera zoom-out or tilt.

The objective of this work is to improve image matching by refining two stages of its pipeline. The improvement of homography estimation can be accomplished, on the one hand, by increasing the number of keypoint correspondences as well as their accuracy, and on the other hand by improving the RANSAC aggregation step. The contributions of this paper, detailed below, address all these issues:

1. We propose a LOCal Affine Transform Estimator (LOCATE) based on a neural network which estimates both the direct and inverse affine maps relating two patches, leading to a more accurate local geometry estimation.

2. To increase the number of correspondences we use the local affine information provided by LOCATE to guide the discovery of new candidates.

3. We introduce a reformulation of the consensus set (inliers) in RANSAC, incorporating the richer information provided by LOCATE, leading to an increase in the probability of success.

A prevalent element in this work is the LOCATE method, which yields a first-order approximation of the local geometry relating pairs of image patches, i.e, local affine maps or tangent planes, see Figure 1. The network architecture of LOCATE is a variation from the one in [6] that provides a two-way estimation, which leads to an increase in robustness relative to the former network. Another difference with respect to [6] is the use of affine simulated patches to train the networks. This simulation incorporates a realistic optical model that takes into account the blur caused by camera tilt and zoom [26]. This procedure allows to easily generate an arbitrarily large training set.

The affine information was already been used [7, 8] to predict location and pose from affine detectors like MSER [17], Harris-Affine[18] or Hessian-Affine [19]. We propose to complement the SIFT detector with a *guided matching* [10] step that increases the number of correct matches by sampling new keypoints surrounding the initial ones. LOCATE's accuracy in location, orientation and scale (i.e. rotation and position in the Gaussian pyramid) results in a drastic increase in the number of correspondences.

When estimating homographies from sets of correspondences with RANSAC, the use of first-order approximations allows to increase the performance in homography estimation. This has already been proposed in [32] by composing normalized affine maps provided by the Hessian Laplace detector. This detector can be replaced with Affnet [23] since it has been shown to produce more accurate affine maps. The LOCATE method can be used as well for the same purpose. In addition, we propose a modification in the RANSAC consensus step. Instead of defining inliers only by location agreement, we also consider the agreement in tilt, rotations and scale of the local affine maps. We will show how these modifications improve homography estimation from a set of SIFT-like matches.

The rest of this paper is organized as follows. Section 2 summarizes a formal methodology for simulating local viewpoint changes induced by real cameras, as required for training our network. The LOCATE method is introduced in Section 3. Section 4 and Section 5 present the proposed guided matching and our modified RANSAC step, respectively. The use of the proposed methods is illustrated with experiments in Section 6. Finally, Section 7 presents our concluding remarks.

2. Affine Maps and Homographies

As stated in [26, 35], a digital image $\mathbf{u}$ obtained by any camera at infinity is modeled as $\mathbf{u} = \mathbf{S}_1 \mathbb{G}_1 A u$, where $\mathbf{S}_1$ is the image sampling operator (on a unitary grid), A is an affine map, u is a continuous image and $\mathbb{G}_\delta$ denotes the convolution by a Gaussian kernel broad enough to ensure no aliasing by δ-sampling. This model takes into account the blur incurred when tilting or zooming a view. Notice that $\mathbb{G}_1$ and A generally do not commute.

Let $\mathcal{A}$ denote the set of affine maps and define $Au(x) = u(Ax)$ for $A \in \mathcal{A}$, where x is a 2D vector and Ax denotes function evaluation, $A\,(x)$. We define the set of invertible orientation preserving affinities $\mathcal{A}^+ = \{L + v \in \mathcal{A} \,|\, \det(L) > 0\}$ where L is a linear map and v a translation vector. We call $\mathcal{S}$ the set of similarity transformations, which are any combination of translations, rotations and zooms. Finally, we define the set $\mathcal{A}_*^+ = \mathcal{A}^+ \setminus \mathcal{S}$, where we exclude pure similarities. As it was pointed out in [26], every $A \in \mathcal{A}_*^+$ is uniquely decomposed as

$$A = \lambda R_1(\psi) T_t R_2(\phi), \tag{1}$$

where R_1, R_2 are rotations and $T_t = \begin{bmatrix} t & 0 \\ 0 & 1 \end{bmatrix}$ with $t > 1$, $\lambda > 0$, $\phi \in [0, \pi)$ and $\psi \in [0, 2\pi)$. Furthermore, the above decomposition comes with a geometric interpretation (see Figure 2) where the longitude ϕ and latitude $\theta = \arccos \frac{1}{t}$ characterize the camera's viewpoint angles (or tilt), ψ parameterizes the camera roll and λ corresponds to the camera zoom. The so-called optical affine maps involving a tilt t in the z-direction and zoom λ are formally simulated by:

$$\mathbf{u} \mapsto \mathbf{S}_1 A \mathbb{G}^z_{\sqrt{t^2-1}} \mathbb{G}_{\sqrt{\lambda^2-1}} I \mathbf{u}, \tag{2}$$

where I is the Shannon-Whittaker interpolator and the superscript z indicates that the operator takes place only in the z-direction. We denote by

$$\mathbb{A} := \mathbf{S}_1 A \mathbb{G}^z_{\sqrt{t^2-1}} \mathbb{G}_{\sqrt{\lambda^2-1}} I \,. \tag{3}$$

The operator $\mathbb{A}$ is not always invertible and therefore its application might incur a loss of information. We refer to [37] for an example where no optical transformation $\mathbb{A}$ is found between two views. With this in mind, we adopt the same data generation scheme proposed for training the affine invariant descriptors in [37]. That is, given an image $\mathbf{u}$ and a pair of optical affine maps $\mathbb{A}_1$ and $\mathbb{A}_2$, we simulate affine views $\mathbf{u}_1 = \mathbb{A}_1(\mathbf{u})$ and $\mathbf{u}_2 = \mathbb{A}_2(\mathbf{u})$. Our simulations involve maximal viewpoint angles of 75° with respect to $\mathbf{u}$. As for [37], the MS-COCO [13] dataset will provide instances of $\mathbf{u}$ in training and validation. Patch pairs seeing the same scene from $\mathbf{u}_1$ and $\mathbf{u}_2$ are said to belong to the same *class* and will be used to train the networks.

2.1. Local affine approximation of homographies

Let $H = (h_{ij})_{i,j=1,\dots,3}$ be the 3×3 matrix associated to the homography $\eta(\cdot)$. Let $\mathbf{x}$ be the homogeneous coordinates vector associated to the image point $x = (x_1, x_2)$ around which we want to determine the local affine map. We denote by $y = (y_1, y_2) = \left(\frac{(H\mathbf{x})_1}{(H\mathbf{x})_3}, \frac{(H\mathbf{x})_2}{(H\mathbf{x})_3} \right) = \eta(x)$ the image of x by the homography η.

The first order Taylor approximation of η at x leads to

$$\eta(x + z) = v + L(x + z) + o(\|z\|), \tag{4}$$

where a brief computation shows that the vector v and the matrix L are determined through the following system of equations:

$$L = \begin{bmatrix} \frac{h_{11} - y_1 h_{31}}{h_{31}x_1 + h_{32}x_2 + h_{33}} & \frac{h_{12} - y_1 h_{32}}{h_{31}x_1 + h_{32}x_2 + h_{33}} \\ \\ \frac{h_{21} - y_2 h_{31}}{h_{31}x_1 + h_{32}x_2 + h_{33}} & \frac{h_{22} - y_2 h_{32}}{h_{31}x_1 + h_{32}x_2 + h_{33}} \end{bmatrix}, \tag{5}$$

$$v = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} - Lx. \tag{6}$$

This derivation allows us to compute the exact local affine approximation for a given homography. This will be useful to assess the accuracy of our method when using annotated datasets.

3. The Local Affine Transform Estimator

In this section we present the LOCal Affine Transform Estimator (LOCATE) network whose architecture is adopted from [6]. Unfortunately, the network as it is used in [6] often incurs in wrong geometry estimates in the presence of strong blur or tilt, even when trained for this task. To address this issue, LOCATE estimates the affine transform that maps query to target *and* target to query. As it will be shown in Section 6, the simultaneous estimation of both, the direct and inverse maps, significantly improves the network performance.

The LOCATE architecture, shown in Figure 3, consists of 4 blocks of two convolutional layers each followed by batch normalization and ReLU activations. The first block receives as input two patches in the form of a two channel image. Between each block a max-pooling layer is introduced. A 2D spatial dropout with a probability 0.5 is applied after the last convolutional layer followed by 2 fully connected layers. The last layer outputs a vector of dimension 16, corresponding to the coordinates of eight points, the four transformed patch corners in both directions. We also tested a network trained to directly estimate the six parameters of local affine maps (translation plus the parameters in Equation 1) but we observed that this choice led to worse performances.

As argued in [37], the affine approximation holds locally, which suggests the use of small patch sizes; on the other hand, small patches contain less information, leading to insufficient geometry anchors. As a compromise, we set the patch size to 60×60, which provides a good balance between locality and sufficient viewpoint information.

3.1. Training

The LOCATE network, as well as the network in [6], were trained with data generated as described in Section 2; more specifically with pairs of patches belonging to the same class and involving small differences in translation, rotation and zoom, but possibly large tilts. The resulting networks will lead to an affine approximation of the exact transformation relating two observations. Both networks are trained from scratch until reaching a *plateau* for the loss in training and validation. While training we also simulate contrast changes on all input patches.

Let A_1, A_2 denote two random affine maps and $\mathbb{A}_1$, $\mathbb{A}_2$ their respective optical simulations. We assume A_1 and A_2

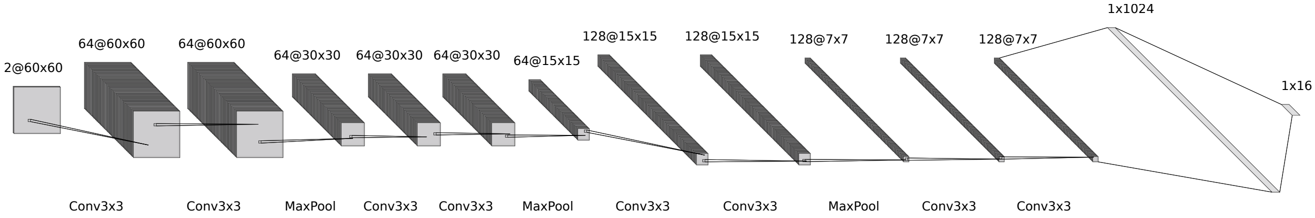


Figure 3. The proposed LOCATE network architecture. The last two layers are fully connected.

involve small perturbations in terms of similarity transformations. Let P_1 and P_2 be two square 60×60-patches simulated from a randomly chosen initial patch P by $\mathbb{A}_1$ and $\mathbb{A}_2$, respectively. Let $X = [x_1, x_2, x_3, x_4]$, where x_i are the 2D coordinates of the four corners of a patch following a fixed order. We also define 4- and 8-point ground truth parameterizations respectively for the network [6] and the LOCATE network,

$$\begin{array}{rcl} X^4 & := & A_1 A_2^{-1}(X), \\ X^8 & := & \left[A_1 A_2^{-1}(X), \, A_2 A_1^{-1}(X)\right], \end{array} \tag{7}$$

where $[\cdot,\cdot]$ denotes the concatenation of both vectors. Let $\mathcal{N}^k$ be one of the presented networks with k-point parameterization. Then the loss is defined as sum of the Euclidean norm between corresponding points:

$$\sum_{i=1}^{k} \|\mathcal{N}^k (P_1, P_2)_i - X_i^k\|_{L_2}, \tag{8}$$

where the sub-index i denotes the i-th element of the vector.

3.2. From patches in the Gaussian pyramid to local affine maps

The training process described above allows the networks to be easily coupled with matching methods based on the SIFT [16] detector. Indeed, a SIFT-like patch is simply the square crop at the origin of some similarity transformation (translation, rotation and zoom) of the original image; additionally, patches corresponding to matched keypoints should suffer small similarity deformations but possibly strong tilts.

Consider two 60×60-patches, P_q and P_t, coming from the Gaussian pyramid of the query and target images, respectively. Let c_q and c_t be their centers expressed in image coordinates. Let also A_q be the affine map that converts from the query image domain to patch coordinates; likewise A_t converts from target to patch coordinates. Note that the affinities A_q and A_t are pure similarities, combining just the translation, rotation and zoom corresponding to the location, orientation and scale associated to SIFT-like keypoints. Finally, in order to locally approximate the transformation between query and target images (centered at c_q and c_t), we only need the affine map relating P_q and P_t.

When fully trained, the presented networks are expected to predict the movements of patch corners. Let $(x_i^q \leftrightarrow x_i^t)_{i=1,\dots,k}$ be a set of correspondences produced by one of the networks $\mathcal{N}^k$, where x_i^q and x_i^t denote query and target patch-coordinates, respectively, and k-point determines the point parameterization. Due to imprecisions in the prediction, these k correspondences are not necessarily related by an affinity. Then, the affine map A is estimated from the correspondences predicted by the network $\mathcal{N}^k$ as the solution of the linear least squares problem

$$\min_A \sum_{i=1}^{k} \left\|A x_i^q - x_i^t\right\|_{L_2}^2 . \tag{9}$$

Finally, around c_q, the local affine map transforming the query into the target (in image coordinates) is

$$A_{q \to t} = A_t^{-1} A A_q. \tag{10}$$

We call LOCATE the method returning $A_{q\to t}$ from the LOCATE network. Figure 4 visually shows estimated affine maps by the network [6] (4 points) and LOCATE, as well as their respective incurred geometric errors. Four random patch pairs from the validation dataset (synthetic data) reveal the Achilles heel of network [6]: zoom and translation. This visualization already justifies the use of the inverse information in the LOCATE method.

4. Refinement and Guided Matching

In this section, we present an iterative procedure that applies LOCATE to refine a set of existing matches, and then retrieves new ones by propagating the estimated local geometry. Think of the initial set of matches as correspondences resulting from a matching method, that includes both inliers and outliers. Each query and target keypoints have an associated location, orientation and scale (i.e. rotation and position in the Gaussian pyramid). The precise affine approximations between query and target obtained from LOCATE, allows to refine the matching by reducing the error in these three similarity parameters.

Furthermore, using the full affine transformations associated to the refined matches, allows to infer new match candidates by propagating the local geometry. The idea

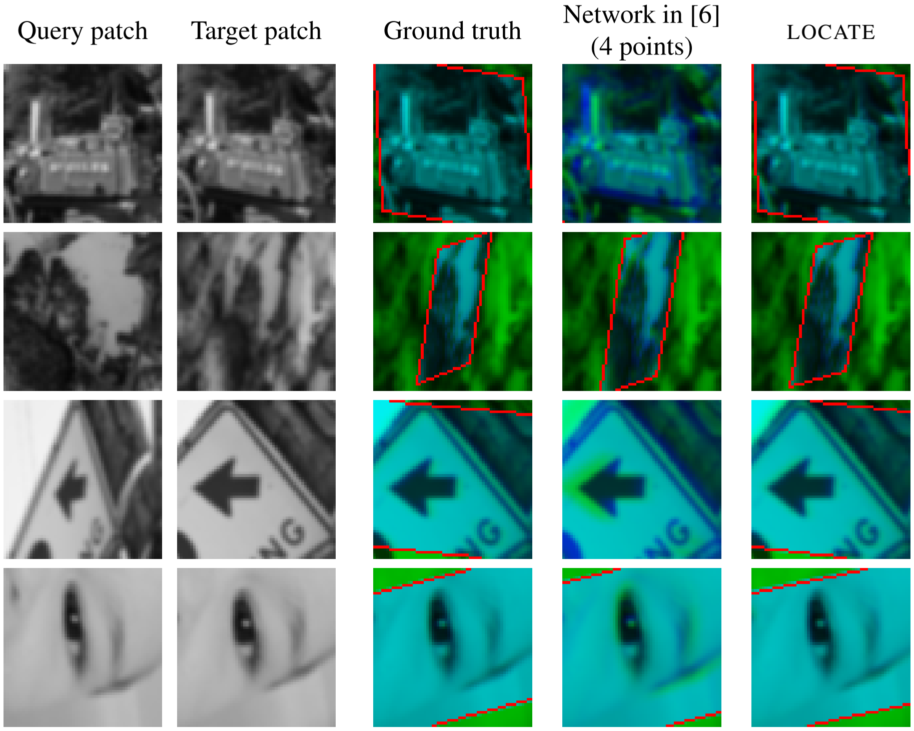


Figure 4. Four pairs of patches selected at random from the validation dataset and used as query and target input patches (columns 1-2). The three last columns show the drift error depicted by intense blue or intense green colors. Light blue means no error. Blue and green channels correspond to the target patch and a warped version of the corresponding query patch (the red line delimits its borders); the red channel is filled with zeros. 3rd column: groundtruth; 4th column: network in [6] (4 points); 5th column: LOCATE network. Input patches are shown without contrast difference for clear visualization.

of propagating the local geometry from a set of matches was already proposed in the literature [7, 8]. In these cases the location and pose are derived from affine detectors like MSER [17], Harris-Affine[18] or Hessian-Affine [19]. Despite the fact that SIFT keypoints are more robust to similarities (see [33]) than the previously mentioned ones, no SIFT-like affine guided matching procedure was proposed yet. The reason for this is that the first method allowing to infer affine maps between SIFT-like patches is Affnet, which was very recently proposed. As we will see in Section 6, LOCATE reaches higher accuracy than Affnet. Therefore, in this work we introduce guided matching based on the LOCATE method.

The procedure is as follows. For each query keypoint from a refined match, four new keypoints are generated at the NE, NW, SE, SW corners of the query patch domain. These points are then mapped into the target image domain with rotations and positions in the target Gaussian pyramid inferred from the affine decomposition in Equation 1. These four pairs of points will represent new tentative matches, and each tentative match is validated by computing a similarity score between corresponding patches. For this task, we use the BigAID descriptor [37] and the cosine proximity to measure the similarity.

This process can be iterated until some criteria is satisfied (e.g., a fixed number of iterations, the number of matches is stable, etc). In this paper, we fix the number of iterations to 4. Each keypoint information is refined only once. To avoid redundancy, new matches falling nearby existing matches are removed (a threshold of 4 pixels was used). Therefore, any valid match proposal will cover new areas connecting the query and target images.

5. Robust Homography Estimation

The standard RANSAC algorithm computes the parameters fitting a mathematical model from observed data in the presence of outliers. Numerous improvements have been proposed in the literature for RANSAC, see [24, 25, 30, 31], but the core idea remains the same.

In the case of homography estimation, the classic RANSAC algorithm returns the homography η_j computed in iteration j having the largest consensus of inliers among all iterations. The j-iteration can be described in two steps:

1. (Fitting) Randomly select s matches $(x_i \leftrightarrow y_i)_{i=1,\ldots,s}$ from the set of all matches (M_T) and compute the homography η_j that yields the best fit.

2. (Consensus) Count the number of matches from M_T that are within a distance threshold of κ (i.e. counting inliers).

Notice that steps 1-2 only take into account point coordinates. From now on, we call this method *RANSAC*. With eight degrees of freedom for a homography matrix and each match defining two equations, this implies $s = 4$. The following subsections support the claim that incorporating the local affine information can further improve the performance of the RANSAC algorithm.

5.1. Homography fitting from local affine maps

From Section 2.1 we know how to locally approximate a homography by an affine map. Conversely, the problem of determining a homography from a set of affine maps at different locations was addressed in [2, 32]. Let $x \leftrightarrow y$ be a match and $L = (l_{ij})_{i,j=1,2}$ the linear map in Equation 4. Then the unknown homography η must satisfy

$$E_{6\times 9} \cdot \vec{h} = \vec{0}, \tag{11}$$

where $E_{6\times 9}$ is the matrix

$$\begin{bmatrix} 1 & & & & & & -y_1 - l_{11}x_1 & -l_{11}x_2 & -l_{11} \\ & 1 & & & & & -l_{12}x_1 & -y_1 - l_{12}x_2 & -l_{12} \\ & & & 1 & & & -y_2 - l_{21}x_1 & -l_{21}x_2 & -l_{21} \\ & & & & 1 & & -l_{22}x_1 & -y_2 - l_{22}x_2 & -l_{22} \\ x_1 & x_2 & 1 & & & & -y_1x_1 & -y_1x_2 & -y_1 \\ & & & x_1 & x_2 & 1 & -y_2x_1 & -y_2x_2 & -y_2 \end{bmatrix}, \tag{12}$$

and $\vec{h} = [h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33}]^T$ is a vectorized version of the matrix H associated to η. The first four rows of $E_{6\times 9}$ are determined by Equation 5 and the last two are the classic equations derived from rewriting $\eta(x) = y$ in terms of $H\mathbf{x} = \mathbf{y}$.

Clearly, two matches with their corresponding local affine maps can over-determine the homography matrix. Indeed, putting everything together provides with 12 equations $\begin{bmatrix} E_1 \\ E_2 \end{bmatrix}_{12\times 9} \cdot \vec{h} = \vec{0}$, where E_i denotes the matrix E appearing in Equation 11 for each match. To avoid the solution $\vec{h} = \vec{0}$ we look for a unitary vector $\vec{h}$ minimizing $\left\| \begin{bmatrix} E_1 \\ E_2 \end{bmatrix} \cdot \vec{h} \right\|$, see [10] for more details.

We call *RANSAC$_{2pts}$* a RANSAC version in which the classic homography fitting of step 1 is replaced by the homography fitting of this section together with the LOCATE estimator. Note that RANSAC$_{\text{2pts}}$ only needs two samples at each iteration ($s = 2$).

5.2. Affine consensus for RANSAC homography

When matching two image patches, the transformation that relates them may not be consistent with the global transformation of the scene. This can be due to the presence of symmetric objects or even to failures in the matching process. For instance, suppose that two patches centered at the same scene location but with incoherent rotations are identified by a matching method. The symmetry issue is easy to address as usually we should have encountered as many keypoints as degrees of symmetry around the center; so at least two rotations will correspond. However, aberrant matches are not treated by the matching method nor by RANSAC. This problem can be circumvented by imposing consistency between the local affine maps and the proposed RANSAC model.

To impose local geometry consistency, most existing works [43, 22] propose to measure the incurred error in mapping keypoints of a match $x \leftrightarrow y$, e.g. $\|y - A(x)\| + \|x - A^{-1}(y)\|$. Unlike them we propose to enforce geometry consistency directly on the transformations parameters given by Equation 1. In other words, we use the affine information to redefine the consensus set of a model.

Inliers are now defined as follows. Let A_E and A_H be, respectively, the estimated affine map by the LOCATE method and the testing affine map computed from the testing homography (using Equation 5). Let also $[\lambda_E, \psi_E, t_E, \phi_E]$ and $[\lambda_H, \psi_H, t_H, \phi_H]$ be, respectively, the affine parameters of A_E and A_H. We define the α-vector between A_E and A_H as:

$$\begin{aligned} \alpha(A_E, A_H) \quad = \quad & \Big[\max\left(\tfrac{\lambda_E}{\lambda_H}, \tfrac{\lambda_H}{\lambda_E}\right), \angle(\psi_E, \psi_H), \\ & \max\left(\tfrac{t_E}{t_H} \tfrac{t_H}{t_E}\right), \angle(\phi_E, \phi_H) \Big], \end{aligned} \tag{13}$$

where $\angle(\cdot,\cdot)$ denotes the angular difference. To test consistency between A_E and A_H we add to the classic threshold on the Euclidean distance, four more thresholds on the α-vector. A perfect match would result in an α-vector equal to $[1, 0, 1, 0]$. If we assume independence on each dimension, the resulting probability of a match passing all thresholds is the multiplication of individual probabilities. With this in mind, we claim that rough thresholds are enough to obtain good performances and that there is no need to optimize them. Thus, we propose to further refine inliers by accepting only those matches also satisfying

$$\alpha(A_E, A_H) < \left[2, \frac{\pi}{4}, 2, \frac{\pi}{8}\right], \tag{14}$$

where the above logical operation is true if and only if it holds true for each dimension.

We call *RANSAC$_{affine}$* the version of RANSAC$_{\text{2pts}}$ that includes the affine consensus presented in this section.

6. Experiments

To the best of our knowledge, the most suitable and effective means of estimating affine maps connecting two

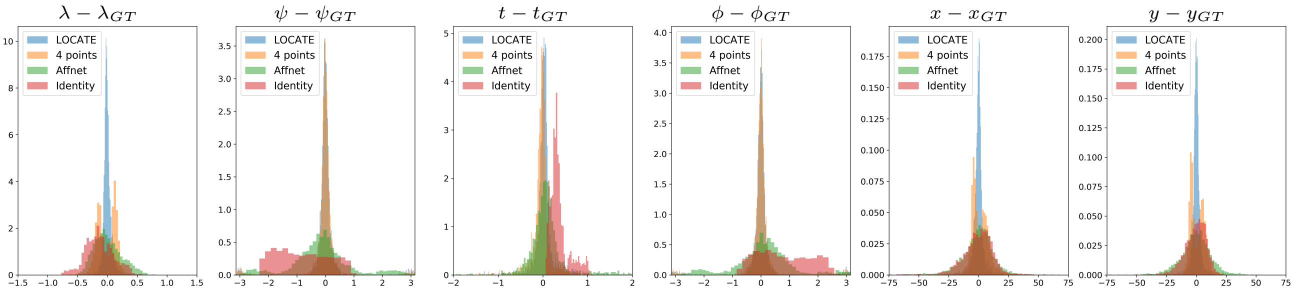
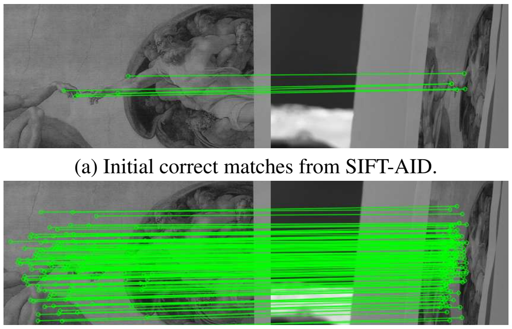


Figure 5. Affine error prediction in terms of the affine decomposition of Equation 1 (namely zoom λ, camera rotation ψ, tilt t, tilt direction ϕ, and translation x, y), for the proposed LOCATE method, the network [6] (4 points), the Affnet method [23] and the identity map method. The used dataset [37] contains 3352 patch pairs with corresponding groundtruth. The sub-index GT means groundtruth, conversely, no sub-index stands for estimated parameters.



(a) Initial correct matches from SIFT-AID.

(b) Homography consistent matches after guiding with LOCATE.

Figure 6. Guided matching for the adam pair, EVD [22].

patches are: Affnet [23], the network [6], and now the LOCATE method. The procedure described in Subsection 3.2 works for both networks: [6] and LOCATE. On the other hand, Affnet was conceived to predict normalizing ellipse shapes for single patches based on a 3-variable parametrization. The connection provided by two Affnet-normalizing affine maps for the query and target patches is richer than each normalizing transformation. Indeed, for different choices of $A_1 = T_1 R_1$ and $A_2 = T_2 R_2$ one would need the four parameters (zoom, camera rotation, tilt and tilt direction) in Equation 1 in order to express $A_2 A_1^{-1}$. However, Affnet does not estimate translations. We claim that the LOCATE method out-performs the other two state-of-the-art methods in terms of precision.

Please note that the networks were trained exclusively with simulated patches, let us now try on real patches. The passage from affine cameras to real cameras is a big gap to fill by both [6] and LOCATE networks. We expect them to generalize the affine world to all sorts of geometry as long as the Taylor approximation holds.

Does precision really matter? As a first evaluation of the precision in a realistic environment we used the viewpoint dataset from SIFT-AID [37], consisting of five pairs of images with their groundtruth homographies and 3352 true matches. Notice that Equations 5-6 allow us to compute groundtruth local affine maps around each match. Figure 5 shows the accuracy of Affnet [23], the 4 points network [6] and LOCATE, represented by error density functions with respect to the affine decomposition appearing in Equation 1. Ideally, we expect a Dirac delta function centered at 0 for a perfect method. This is approximately true for the LOCATE method. The experiment also illustrates the failure of the network [6] in predicting zoom and translation (as shown in Figure 4). Note in Figure 5 that translations from the Affnet [23] method do not quite match those from the Identity method; this difference can be explained by the connecting mapping itself as $A_{1\rightarrow 2}(\mathbf{x}) = A_2\left(A_1^{-1}\mathbf{x} - A_1^{-1}\mathbf{c}\right) + \mathbf{c}$ is different from $A_2 A_1^{-1}\mathbf{x}$, where $\mathbf{c}$ denotes the center of patch domain and A_i are the estimated affine maps by Affnet. LOCATE, with the only addition of tracking points movements associated to the inverse affine map, obtains better result than [6]. As expected, both [6] and LOCATE perform better than Affnet [23]. Indeed, Affnet analyzes one patch at a time, whereas [6] and LOCATE have access to both patches simultaneously. However, in practice, using Affnet involves less computations.

The following experiment shows that the precision improvement of LOCATE indeed results in better guided image matching performance. Table 1 shows that LOCATE has the overall best performance of all methods. LOCATE usually boost the number of inliers as well as the ratio of inliers while always being the lowest or close to lowest average pixel error. By construction, this boost in inliers means that new areas are connected between the image pairs, see Figure 6 for an example. Moreover, the probability of success of RANSAC USAC [30] is not diminished with respect to the matching method itself, this is observed in the "None" rows of Table 1. We remark the capacity of our guided matching method to expand true matches while keeping the number of false matches low.

Can RANSAC$_{\text{affine}}$ improve homography estimation? In the previous paragraphs we established the precision of

Matching method	Guiding affine map	SIFT-AID dataset [37]					EVD dataset [22]					OxAff Viewpoint dataset [20]				
		S	$\boxed{5}$	inl.	AvE	R	S	$\boxed{15}$	inl.	AvE	R	S	$\boxed{10}$	inl.	AvE	R
SIFT-AID	None	500	5	508	6.2	0.24	100	1	162	6.2	0.11	1000	10	1840	4.1	0.43
	Identity	487	5	114	6.3	0.33	100	1	19	6.9	0.46	1000	10	1546	4.6	0.62
	LOCATE	500	5	**1438**	5.2	**0.44**	200	2	**862**	**3.8**	0.49	1000	10	**7198**	**2.7**	**0.71**
	4 points	500	5	1166	**5.1**	0.41	200	2	548	4.1	0.46	1000	10	6725	2.8	0.70
	Affnet	487	5	328	7.0	0.31	103	2	142	6.7	**0.50**	1000	10	2223	5.4	0.57
SIFT-Affnet	None	400	4	99	**3.8**	**0.79**	235	3	13	7.9	0.64	1000	10	1185	**2.1**	**0.96**
	Identity	300	3	32	4.2	0.71	0	0	0	-	-	895	9	1336	3.5	0.94
	LOCATE	400	4	**620**	4.7	0.72	200	2	151	5.6	**0.98**	1000	10	**6871**	2.5	**0.96**
	4 points	400	4	448	4.6	0.73	101	2	**169**	**3.1**	0.94	1000	10	6164	2.7	0.94
	Affnet	400	4	78	5.8	0.69	100	1	28	5.6	0.86	1000	10	1724	4.8	0.88

Table 1. Guided matching and refinement performances on three viewpoint datasets with seed correspondences from two affine invariant matching methods: SIFT-AID [37] and SIFT-Affnet[23]-HardNet[21] (SIFT-Affnet). After refinement and guiding on each image pair, RANSAC-USAC [30] is run 100 times to measure its probability of success in retrieving corresponding ground truth homographies. Legend: S - the number of successes (bounded by 100×$\boxed{\text{number}}$); the number of correctly matched image pairs; inl. - the average number of correct inliers; AvE - the average pixel error; R - the ratio of inliers/total. The $\boxed{\text{numbers}}$ of image pairs in a dataset are boxed.

Matching method	Homography Estimator	EF dataset [44]				EVD dataset [22]				OxAff dataset [20]				SymB dataset [11]			
		S	$\boxed{33}$	inl.	AvE	S	$\boxed{15}$	inl.	AvE	S	$\boxed{40}$	inl.	AvE	S	$\boxed{46}$	inl.	AvE
RootSIFT	RANSAC	2403	26	51	3.2	0	0	0	-	3806	39	580	1.2	2693	31	102	2.8
	$\text{RANSAC}_{\text{2pts}}$	2633	28	46	3.7	0	0	0	-	3893	39	566	1.2	3219	34	84	3.3
	$\text{RANSAC}_{\text{affine}}$	**2805**	**30**	28	3.4	0	0	0	-	**3899**	**39**	404	1.1	**3297**	**36**	54	3.4
SIFT-AID	RANSAC	879	23	78	6.6	82	1	40	7.8	3600	39	1477	4.8	1014	19	450	6.8
	$\text{RANSAC}_{\text{2pts}}$	1829	27	84	6.1	99	1	72	6.3	3917	40	1459	4.5	1867	30	327	6.5
	$\text{RANSAC}_{\text{affine}}$	**1996**	**30**	39	5.8	**166**	**5**	37	8.2	**3939**	**40**	852	4.0	**2341**	**38**	138	6.6
SIFT-Affnet	RANSAC	2475	25	47	3.7	200	2	16	8.0	**4000**	**40**	805	2.3	2999	31	108	3.5
	$\text{RANSAC}_{\text{2pts}}$	2707	28	43	3.6	**300**	**3**	10	7.6	**4000**	**40**	805	2.3	3268	34	99	3.4
	$\text{RANSAC}_{\text{affine}}$	**2826**	**29**	29	3.5	200	2	12	7.4	**4000**	**40**	562	2.2	**3285**	**36**	65	3.5

Table 2. Homography estimation performances for RANSAC, $\text{RANSAC}_{\text{2pts}}$ and $\text{RANSAC}_{\text{affine}}$ for three matching methods: RootSIFT [1], SIFT-AID [37], and SIFT-Affnet[23]-HardNet[21] (SIFT-Affnet). Each RANSAC ran for 1000 internal iterations. To measure probability of success, all RANSACs were run 100 times on resulting matches from each pair of images. Legend: S - the number of successes (bounded by 100×$\boxed{\text{number}}$); the number of correctly matched image pairs; inl. - the average number of correct inliers; AvE - the average pixel error. The $\boxed{\text{numbers}}$ of image pairs in a dataset are boxed.

the local affine maps provided by the LOCATE method. We now focus on the evaluation of the three variants of RANSAC. In order to highlight the benefits of local geometry in estimating homographies, we drop all the improvements in RANSAC USAC [30] and head back to the base RANSAC. But notice that most improvements proposed in RANSAC USAC [30] can also be applied to $\text{RANSAC}_{\text{2pts}}$ and $\text{RANSAC}_{\text{affine}}$. The following experiment was conducted on four well known datasets for homography estimation. All datasets include groundtruth homographies that were used to verify accuracy. First, local features were detected and matched, then each homography estimation method (RANSAC, $\text{RANSAC}_{\text{2pts}}$ and $\text{RANSAC}_{\text{affine}}$) was applied and we declared a success if at least 80% of inliers (in consensus with the estimated homography) were in consensus with the groundtruth homography. The two steps of RANSAC (fitting and consensus) are iterated a 1000 times for each of the three variants. Therefore, the processing time spent in applying LOCATE could be compensated later on by decreasing the number of internal iterations. For equal settings, rows 'None' in Table 1 and rows 'RANSAC' in Table 2 do not correspond; this is because RANSAC USAC [30] was used in the former while baseline RANSAC in the latter.

7. Conclusions

We proposed a CNN based method to locally estimate affine maps between images. Our experiments show that the LOCATE method provides accurate first-order approximations of local geometry. This information proved to be valuable for two applications: Guided matching of SIFT keypoints with precise locations, orientations and scales; and homography estimation, for which we presented a RANSAC version that systematically improved results in four well known datasets [44, 22, 20, 11]. Training LOCATE to handle occlusions as well as applications to stereo matching will be explored in future work.

References

[1] R. Arandjelovic and A. Zisserman. Three things everyone should know to improve object retrieval. In *CVPR*, pages 2911–2918, 2012.

[2] D. Barath and L. Hajder. Novel ways to estimate homography from local affine transformations. 2016.

[3] A. Baumberg. Reliable feature matching across widely separated views. *CVPR*, 1:774–781, 2000.

[4] J. Blom. Topological and Geometrical Aspects of Image Structure. *University of Utrecht*, 1992.

[5] F. Cao, J.-L. Lisani, J.-M. Morel, P. Musé, and F. Sur. *A Theory of Shape Identification*. Springer Verlag, 2008.

[6] D. DeTone, T. Malisiewicz, and A. Rabinovich. Deep image homography estimation. *arXiv preprint arXiv:1606.03798*, 2016.

[7] E. Farhan and R. Hagege. Geometric expansion for local feature analysis and matching. *SIAM Journal on Imaging Sciences*, 8(4):2771–2813, 2015.

[8] E. Farhan, E. Meir, and R. Hagege. Local Region Expansion: a Method for Analyzing and Refining Image Matches. *Image Processing On Line*, 7:386–398, 2017.

[9] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[10] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[11] D. C. Hauagge and N. Snavely. Image matching using local symmetry features. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 206–213. IEEE, 2012.

[12] T. Iijima. Basic equation of figure and and observational transformation. *Systems, Computers, Controls*, 2(4):70–77, 1971.

[13] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014.

[14] T. Lindeberg. *Scale-Space Theory in Computer Vision.* Royal Institute of Technology, Stockholm, Sweden, 1993.

[15] T. Lindeberg and J. Garding. Shape-adapted smoothing in estimation of 3-D depth cues from affine distortions of local 2-D brightness structure. *ECCV*, pages 389–400, 1994.

[16] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[17] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *IVC*, 22(10):761–767, 2004.

[18] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. *ECCV*, 1:128–142, 2002.

[19] K. Mikolajczyk and C. Schmid. Scale and Affine Invariant Interest Point Detectors. *IJCV*, 60(1):63–86, 2004.

[20] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, R. Kadir, and L. Van Gool. A comparison of affine region detectors. *International journal of computer vision*, 65(1-2):43–72, 2005.

[21] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Advances in Neural Information Processing Systems*, pages 4826–4837, 2017.

[22] D. Mishkin, J. Matas, and M. Perdoch. MODS: Fast and robust method for two-view matching. *CVIU*, 141:81–93, 2015.

[23] D. Mishkin, F. Radenovic, and J. Matas. Repeatability is not enough: Learning affine regions via discriminability. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 284–300, 2018.

[24] L. Moisan, P. Moulon, and P. Monasse. Automatic Homographic Registration of a Pair of Images, with A Contrario Elimination of Outliers. *IPOL*, 2:56–73, 2012.

[25] L. Moisan, P. Moulon, and P. Monasse. Fundamental Matrix of a Stereo Pair, with A Contrario Elimination of Outliers. *IPOL*, 6:89–113, 2016.

[26] J.-M. Morel and G. Yu. ASIFT: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences*, 2(2):438–469, 2009.

[27] P. Musé, F. Sur, F. Cao, and Y. Gousseau. Unsupervised thresholds for shape matching. *ICIP*, 2003.

[28] P. Musé, F. Sur, F. Cao, Y. Gousseau, and J.-M. Morel. An A Contrario Decision Method for Shape Element Recognition. *IJCV*, 69(3):295–315, 2006.

[29] Y. Pang, W. Li, Y. Yuan, and J. Pan. Fully affine invariant SURF for image matching. *Neurocomputing*, 85:6–10, 2012.

[30] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm. USAC: a universal framework for random sample consensus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):2022–2038, 2013.

[31] M. Rais, G. Facciolo, E. Meinhardt-Llopis, M. J.-M., B. A., and C. B. Accurate motion estimation through random sample aggregated consensus. *CoRR*, abs/1701.05268, 2017.

[32] C. Raposo and J. P. Barreto. Theory and practice of structure-from-motion using affine correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5470–5478, 2016.

[33] I. Rey-Otero, M. Delbracio, and J.-M. Morel. Comparing feature detectors: A bias in the repeatability criteria. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 3024–3028. IEEE, 2015.

[34] I. Rocco, R. Arandjelovic, and J. Sivic. Convolutional neural network architecture for geometric matching. *TPAMI*, 2018.

[35] M. Rodriguez, J. Delon, and M. J.-M. Covering the space of tilts. application to affine invariant image comparison. *SIIMS*, 11(2):1230–1267, 2018.

[36] M. Rodriguez, J. Delon, and J.-M. Morel. Fast affine invariant image matching. *IPOL*, 8:251–281, 2018.

[37] M. Rodriguez, G. Facciolo, R. Grompone von Gioi, P. Musé, J.-M. Morel, and J. Delon. Sift-aid: boosting sift with an affine invariant descriptor based on convolutional neural networks. In *ICIP*, Sep 2019.

[38] M. Rodriguez and R. Grompone von Gioi. Affine invariant image comparison under repetitive structures. In *ICIP*, pages 1203–1207, Oct 2018.

[39] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[40] T. Tuytelaars and L. Van Gool. Wide baseline stereo matching based on local, affinely invariant regions. *BMVC*, pages 412–425, 2000.

[41] T. Tuytelaars and L. Van Gool. Matching Widely Separated Views Based on Affine Invariant Regions. *IJCV*, 59(1):61–85, 2004.

[42] T. Tuytelaars, L. Van Gool, and Others. Content-based image retrieval based on local affinely invariant regions. *Int. Conf. on Visual Information Systems*, pages 493–500, 1999.

[43] Y. Zheng and D. Doermann. Robust point matching for nonrigid shapes by preserving local neighborhood structures. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):643–649, 2006.

[44] C. L. Zitnick and K. Ramnath. Edge foci interest points. In *2011 International Conference on Computer Vision*, pages 359–366. IEEE, 2011.