On the matching precision of SIFT

Zhongwei Tang, Pascal Monasse, Jean-Michel Morel

ON THE MATCHING PRECISION OF SIFT

Zhongwei Tang[1], *Pascal Monasse*[1], *Jean-Michel Morel*[2]

[1]IMAGINE, LIGM, Université Paris-Est/ Ecole des Ponts ParisTech, France
[2]CMLA, Ecole Normale Supérieure de Cachan, France

ABSTRACT

Matching precision of scale-invariant feature transform (SIFT) is evaluated and improved in this paper. The aim of the paper is not to invent a new feature detector more invariant than the others. Instead, we focus on SIFT method and evaluate and improve the matching precision, defined as the root mean square error (RMSE) under ground truth geometric transform. Matching precision reflects to some extent *the average relative localization precision* between two images. For scale invariant feature detectors like SIFT, the matching precision decreases with the scale of features due to the sub-sampling in the scale space. We propose to cancel the sub-sampling to improve the matching precision. But in case of scale change, the improvement is marginal due to the coarse scale quantization in the scale space. One more sophisticated method is also proposed to improve the matching precision in case of scale change. These modifications can be easily extended to other scale invariant feature detectors.

Index Terms— Matching precision, localization precision, scale space, scale-invariant feature transform (SIFT)

1. INTRODUCTION

Recent years have seen the blossom of invariant feature detectors [??]. Even though they become more and more invariant to image geometric transform and illumination change, the matching precision has not been carefully studied. Matching precision seems similar to the repeatability, which is defined as the percentage of the repeatable features among all the matched features. One feature is considered repeatable if it and its matched feature are compatible with the ground truth transform within some loose precision threshold (1 pixels is chosen in [??]). However, matching precision directly measures the average residual error for all the correctly matched feature pairs with respect to the ground truth geometric transform. So even if a feature detector gives high repeatability, it does not necessarily mean that the matching precision is high.

The localization precision needs to be clarified here because it is a confusing concept with the matching precision. Localization precision is used to describe how close *on average* the position of detected feature is to the *true position*. In statistics terms, localization precision is called bias. A *statistical ensemble* is required to define the bias. In the case of feature detection, the underlying *statistical ensemble* can be interpreted as a collection of images, obtained by disturbing an ideal image by an underlying disturbance model, like noise model, blur model, illumination model, etc. Assume the true position of a feature point is known. The localization precision (bias) can be measured as the distance from the average position of the detected features by a particular feature detector on differently disturbed images to the true position. The localization precision is not uniform in the image domain, different from one feature point to another.

It is difficult to evaluate the localization precision in practice. The computation of localization precision first requires the ideal position of feature points, which is in fact ill-defined. Even if the ideal position of feature points is known, it is not clear what is an appropriate *statistical ensemble*, which can be deduced from an underlying noise model, a camera model or a model of lighting condition. So in practice, it is impossible to measure the localization precision. However, absolute localization precision is less interesting in computer vision tasks, where the correspondences are usually required. This motivates us to measure the matching precision instead of localization precision. In fact, matching precision reflects to some extent the localization precision. Given enough correspondences between two images, assume that all the feature points in one image are ideal and the ground truth transform between two images is known, then the matching precision is close to the localization precision under the hypothesis that the feature detector has the same localization precision on all the feature points in the other image and the local property of all the features composes an appropriate *statistical ensemble*. In reality feature points in both images are not ideal and their localization precision is different from point to point. So the matching precision measures the *average relative localization precision* of matchings between two images.

We first review SIFT method and improve its matching precision in Section 2. The improvement seems to be marginal in case of scale change between two images in the evaluation in Section 3, which motivates a more sophisticated improvement based on local filtering. The extension of the

Part of this work was supported by the Agence Nationale de la Recherche (ANR) project STEREO (program ASTRID 2012).

proposed improvement is briefly discussed in Section 4.

2. SIFT METHOD REVIEW AND IMPROVEMENT

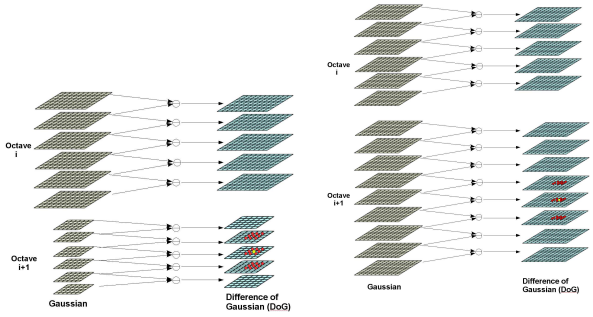
The SIFT method [1] is a complete image comparison algorithm composed of scale-invariant feature detector, gradient-based descriptor and descriptor matching. The features are detected in the scale space of normalized Laplacian, approximated by the difference of Gaussian due to the computational efficiency:

$$D(x,y,\sigma) = \Big(G(x,y,k\sigma) - G(x,y,\sigma)\Big) \circledast I(x,y) \quad (1)$$
$$\approx (k-1)\sigma^2 \Delta\Big(G(x,y,\sigma) \circledast I(x,y)\Big)$$

$G(\cdot,\cdot,\sigma)$ is the Gaussian function with standard deviation σ and $\circledast$ is the convolution operation. Remark that the normalized Laplacian gives scale invariance to the Laplacian threshold in SIFT method. Images of same size with increasing Gaussian blur compose one octave in the scale space. And there is a 2-subsampling between two adjacent octaves to simulate the camera zoom (Fig. 1a). One octave contains $Ninter + 3$ Gaussian blurred images of the same size which are used to compute $Ninter + 2$ difference-of-Gaussian (DoG) images ($Ninter = 3$ by default). The Gaussian blur is increased with the multiplicative factor $2^{1/Ninter}$. The 2-subsampling is performed on the image in octave which contains two times the blur of the initial image in the same octave. This convolution-subsampling procedure is repeated until the image is too small for feature detection. It is easy to see that the sampling with respect to the blur is the same for all octaves. So one image has the same nature as its counterparts in the other octaves. This process simulates camera zoom and explains why SIFT method is scale invariant.

Only local extrema with strong Laplacian value in $Ninter$ DoG images in the middle of each octave are detected as features. Once a feature is extracted, its 3D position (position and scale) is refined by a 3D interpolation, which makes SIFT points sub-pixel precise. Each feature is assigned a descriptor which is constructed by using the gradient information in the neighborhood. Finally, their 3D coordinate (location and scale) is projected back to the original image.

Remark that the principal error source in SIFT method is that the detected features in scale space are projected back to the original image. Assume a feature located at $\mathbf{x}$ with ideal position $\mathbf{x_0}$ disturbed by the error ε: $\mathbf{x} = \mathbf{x_0} + \varepsilon$. If this feature is detected in the i-th octave in SIFT scale space, then its final position is $2^i\mathbf{x} = 2^i\mathbf{x_0} + 2^i\varepsilon$. The error is increased by the factor 2^i. This inspires us to cancel the sub-sampling between octaves. The new schema is shown in Fig. 1b. Although this seems to be a one-step modification to SIFT method, there are some details to discuss.



(a) (b)

Fig. 1: 1a The scale space of SIFT. 1b The improved SIFT schema with sub-sampling canceled.

2.1. Blur

Blur plays a very important role in SIFT method because the scale invariance of SIFT is in fact blur invariance and is achieved by simulating the camera blur under different resolutions. SIFT method is based on the assumption that a Gaussian convolution can well approximate the blur introduced by camera system and gives an aliasing-free image subsampling. In fact, it is shown in [2] that a well-sampled image should contain a Gaussian blur of standard deviation $\beta = 0.8$. Therefore an aliasing-free t-subsampling should be preceded by a Gaussian blur of about $\beta \times \sqrt{t^2 - 1}$. However, if we cancel the sub-sampling between adjacent octaves, it is equivalent to up-sample images in the scale space and the following conditions should be satisfied:

$$\Delta\left(u(\frac{x}{2},\frac{y}{2})\right) = \frac{1}{4}(\Delta u)\left(\frac{x}{2},\frac{y}{2}\right) \text{ and } \frac{\partial\left(u\left(\frac{x}{2},\frac{y}{2}\right)\right)}{\partial \bullet} = \frac{1}{2}\frac{\partial u}{\partial \bullet}\left(\frac{x}{2},\frac{y}{2}\right) \quad (2)$$

which means that the Laplacian is 4 times smaller and the gradient is 2 times smaller if an image is up-sampled by 2. For digital images, the above relationships are valid only if the image u is smooth enough. Fig. 2 shows a test for a natural image. The image is first convolved by a Gaussian blur, then followed by an up-sampling by factor 2. The Laplacian and gradient module are computed on the original image and the up-sampled image respectively. Note m the ratio of Laplacian before and after 2-upsampling and n the ratio of gradient norm before and after 2-upsampling. By computing the average and standard deviation of m and n with respect to the added blur, it appears that the above conditions are satisfied only if the added Gaussian blur is bigger than 1.6. This makes the image blur equal to $\sqrt{1.6^2 + 0.8^2} \approx 1.8$ if the original image is assumed to already contain a blur 0.8. Similar results are obtained with another dozen of natural images. This experiment is complementary to the one dealing with aliasing-free sub-sampling in [2]. Our conclusion is that a good image for feature analysis must contain at least a 1.8 Gaussian blur. There are two ways to achieve this quantity of blur: either we

convolve the input image with enough Gaussian blur, or we pre-zoom the input image by 2 to increase the blur to be 1.6, close to 1.8 as required (if the initial image contains Gaussian blur 0.8). This step of pre-zoom is optional in original SIFT in order to increase the number of features.

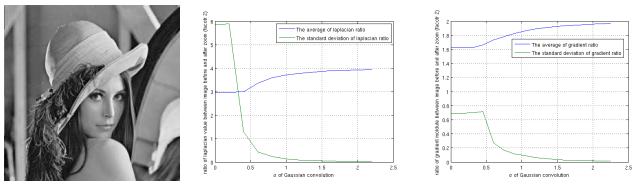


Fig. 2: Image on the left is convolved by a Gaussian function with standard deviation σ before it is up-sampled by factor 2. The Laplacian value and gradient modulus before and after the up-sampling are compared. Middle: the average and standard deviation of ratio of the Laplacian value before and after 2-upsampling. Right: the average and standard deviation of ratio of gradient modulus before and after 2-upsampling.

2.2. 3D refinement

Once the local extrema are extracted in 3D scale space, their position can be refined under the assumption that the DoG image can be locally approximated by a second order Taylor expansion. Given a local extrema located at $\mathbf{x} = (x, y, \sigma)$, the DoG function $D(\mathbf{x})$ is expanded at $\mathbf{x}$ by: $D(\mathbf{x} + \mathbf{\Delta x}) = D(\mathbf{x}) + \mathbf{\Delta x}^T \frac{\partial D}{\partial \mathbf{x}} + \mathbf{\Delta x}^T \frac{\partial^2 D}{\partial \mathbf{x}^2} \mathbf{\Delta x}$. The peak of this function is attained when its derivative is set to be zero, which gives the offset $\mathbf{\Delta x} = (\Delta x, \Delta y, \Delta\sigma)^T = \left(\frac{\partial^2 D}{\partial \mathbf{x}^2}\right)^{-1} \frac{\partial D}{\partial \mathbf{x}}$. Sub-pixel precision is thus obtained and the final position is $\mathbf{x} + \mathbf{\Delta x}$.

If the sub-sampling is canceled, the blur between two adjacent scales in octaves increases also by factor 2, 4, 16, $\cdots$. Thus the scale space is sampled more and more sparsely through octaves. This makes it more difficult for the 3D interpolation refinement to produce a precise result. In addition, the SIFT descriptor is constructed approximately on these sparse intervals without really interpolating a new interval. This makes descriptors less accurate. To compensate this effect, the number of intervals is increased with the same factor through octaves (see Fig. 1b). This means that the up-sampling is performed also in the scale direction, just as that in the x and y directions in image.

3. MATCHING PRECISION EVALUATION

Matching precision is evaluated on pairs of images under different geometric transforms: translation, rotation, zoom, affine transform. The translations are respectively $(45, 32)$, $(45.1, 32.1)$, $(45.3, 32.3)$, $(45.5, 32.5)$, $(45.7, 32.7)$ and $(45.9, 32.9)$. The rotations varies from $15°$ to $85°$ with the step of $10°$. The zooming factors are respectively $2^{1/6}$, $2^{2/6}$, $2^{3/6}$, $2^{4/6}$, $2^{5/6}$ and $2^{6/6}$. The affine transform A is of the parametric form $A = R_1 T_t R_2$ with R_1 and R_2 rotation of $24°$ and $37°$ respectively, and T_t is the tilt in the x direction with compression factor $t = 2^{1/12}$, $2^{2/12}$, $2^{3/12}$, $2^{4/12}$, $2^{5/12}$ and $2^{6/12}$ respectively.

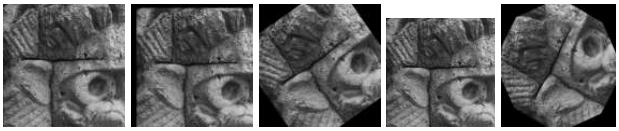


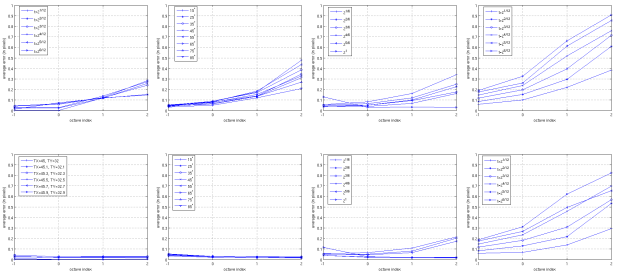
Fig. 3: Some images used in the experiments. The leftmost one is the reference image and the others are synthetized from it.

The reference image is fixed and the second image is synthesized from the reference one according to the transform by 7th-order spline interpolation to avoid the "ringing" artifact introduced by Fourier interpolation. The pre-zoom of SIFT method is activated and the number of octaves is fixed to be four (the octave index begins with -1). Only local extrema with strong Laplacian value are detected as features and are matched. A RANSAC-like algorithm is performed on the matchings of each octave respectively in order to remove the "outliers" and to estimate the geometric transform for each octave. The matching precision is then computed as the root mean square error (RMSE) on each octave with respect to the estimated transform respectively.

It is shown in Fig. 4 that the matching precision of original SIFT decreases through octaves. This is not surprising because all detected features are finally projected back to the original image. For improved SIFT, the matching precision is kept or even improved through octaves when there is no scale change between two images, namely translation and rotation. However, when a scale change is present, namely zoom and affine transform, the gain in precision is marginal. This is mainly due to the fact that the matched features are detected at different scales (blur) when the transform between two images contains a scale change. The 3D refinement being sensitive to blur, the precision of the refined position and scale for the matched features will be different. So the matching precision, which can be interpreted to some extent as the average relative localization precision, is lower than that in case of translation and rotation.

If we know in advance the transformation between two images is some parametric transform, the above problem of blur inconsistency can be solved to some extent by estimating then compensating the transform using the most precise matchings in the first octave. But in practice, the parametric form of the transformation between two images is unknown due to the 3D effect of scene. Here we concentrate on the cases where the transformation between two images is smooth and can be locally approximated by homography. Although this does not apply when the scene is really 3D, there are already many applications based on this assumption, like image stitching and camera calibration.

The main idea is to use the local filter by homography

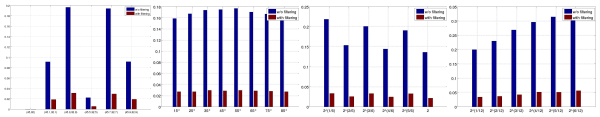


(a) translation (b) rotation (c) zoom (d) affine transform

Fig. 4: The matching precision for the original SIFT and improved SIFT under different transforms. The x-axis is the octave index, from -1 to 2. The y-axis the average residual error (in pixels). Top row: original SIFT. Bottom row: improved SIFT.

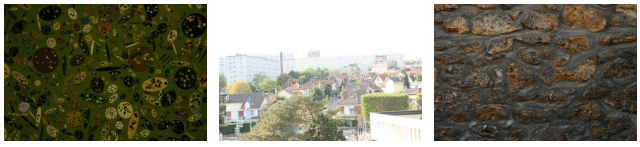
to increase the matching precision. We detect a dense set of SIFT matching between two images. To have enough matchings for natural images, we consider every pixel as a feature point in the first two octaves of improved SIFT scheme. Only a low Laplacian threshold is used to eliminate very unstable feature points. The matching process is accelerated by dividing the image domain into several blocks and the resulting matchings are pruned by vector filter to obtain reliable matchings. For each SIFT matching, 100 neighboring matchings around it are used to estimate the best local homography in the least-square sense. Then one point in each SIFT matching is adjusted according to its corresponding homography. We evaluate the whole procedure on the different geometric transformations as before. The matchings in octave -1 and 0 are mixed together to have dense matchings over image domain. The evaluation then gives the average matching error without distinguishing their octave index. In Fig. 5, without the local filtering, the matching precision of improved SIFT is coarse due to the relaxed criteria in feature detection. But the precision is largely improved with the local filtering. The default of this method is that not many features can be matched by SIFT under the transformations too "affine", even though almost each pixel is considered as a feature point. This will degrade the filtering performance of local homographies. This explains why the precision decreases in the case of affine transformation with the increase of the parameter from $2^{1/12}$ to $2^{6/12}$. This is in fact not a problem because we can always make two images to be more similar by transforming one image using a coarse homography between them, even if the relation between both images is not a homography.

For real images, the matching precision is also affected by the noise in the image. To test the performance of the algorithm for real images, a Canon EOS 30D SLR camera with EFS $18-55$mm lens was used to take photos. Three pairs of images were tested (Fig. 6): the first pair for a planar abstract painting, the second pair for an infinite homography, the third pair for a distant wall with small camera motion. Ideally the underlying geometric transformation is a homography. But for real images, the homography cannot be used to measure the matching precision due to the lens distortion. Even though the maximal focal length (55mm) was chosen to avoid the lens distortion as much as possible, there still exists small distortion. The bivariate polynomial model, an universal and practical distortion model, which is also consistent with a homography, was used here to evaluate the matching precision. The result is recapitulated in Fig. 7. It seems that the local filtering technique is very efficient to increase the matching precision even with noisy images.



(a) translation (b) rotation (c) zoom (d) affine transform

Fig. 5: The matching precision of improved SIFT without/with local filter under different geometric transformations.



(a) drawing (b) house (c) wall

Fig. 6: Three pairs of images taken by Canon EOS 30D SLR camera with focal length 55mm.

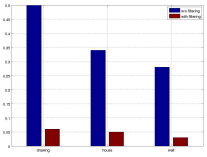


Fig. 7: The matching precision of improved SIFT without/with local filter on real image pairs evaluated by a 10th-degree bivariate polynomial.

4. CONCLUSION

The matching precision of SIFT method is reviewed and improved. We show that a simple canceling of sub-sampling in the scale space can improve the matching precision if there is no scale change between images. In the presence of scale change, we propose to improve the matching precision by local homography filtering if the image transform is smooth. In the future work, we would like to improve the matching precision for more general transforms, including transforms induced by 3D scene. We also plan to extend the work to other feature detectors.

5. REFERENCES

[1] David G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.

[2] J.M. Morel and G.Yu, "On the consistency of the sift method," *Preprint, CMLA, ENS-Cachan*, 2009.