On the Improvement of Image Feature Matching under Perspective Transformations

Vilar Fiuza da Camara Neto
Núcleo de Estudo e Pesquisa em Computação
Fund. Centro de Análise, Pesq. e Inov. Tecnológica (FUCAPI)
Manaus, AM, Brazil
Email: neto@dcc.ufmg.br

Mario Fernando Montenegro Campos
Departamento de Ciência da Computação
Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte, MG, Brazil
Email: mario@dcc.ufmg.br

***Abstract*—This paper presents a novel methodology to perform consistent matching between visual features of a pair of images, particularly in the case of point-of-view changes between shots. Traditionally, such correspondences are determined by computing the similarity between descriptor vectors associated with each point which are obtained by invariant descriptors. Our methodology first obtains a coarse global registration among images, which constrains the correspondence space. Then, it analyzes the similarity among descriptors, thus reducing both the number and the severity of mismatches. The approach is sufficiently generic to be used with many feature descriptor methods. We present several experimental results that show significant increase both in accuracy and the number of successful matches.**

***Keywords*-visual features correspondence; image matching;**

I. Introduction

In Computer Vision, one often needs to determine correspondences between points of pairs of images. This problem is known as *matching* or *correspondence* and is a fundamental step to solve many other problems, such as multiple view geometry, image mosaicing, object tracking, visual odometry, scene mapping and monocular SLAM. Recently, a large number of works have been published that perform matching of small image patches based on visual feature descriptors, such as SIFT [1], [2], PCA-SIFT [3], GLOH [4], SURF [5], MSER [6] and ASIFT [7]. However, none of these methods addresses the (almost always expected) general geometric consistency between the transformation from one image to the other, which often generates correspondences between seemingly random points of the images.

To significantly avoid outliers, descriptor-based correspondence can be combined to robust methods that enforce geometric compatibility between the matched points. While RANSAC [8] comes to mind as a candidate, in the general case of unknown scene geometry it can only be used to estimate the simplest image relations, such as affine transformations or homographies; also, the large number of parameters to estimate (8 for a full homography between images) makes it unsuitable for real-time applications.

In a previous work, the authors [9] presented a methodology that uses geometric clues (position, orientation and size), already provided by feature descriptor algorithms, to search for ranges for translation, rotation and scaling that describe the relation between points of a pair of images. Although this method attempts to capture the global correspondence as a (bounded) affine transformation from one image to the other, it was shown that it suffices to describe even small perspective changes between shots. However, the original algorithm presents two main weaknesses:

1) it cannot cope with general point-of-view changes between shots (large perspective transformations and scenes with objects placed at distinct distances from the camera) nor with dynamic scenes; and
2) more importantly, it only compares descriptors that comply with the aforementioned transformation bounds, which sometimes causes the algorithm to blindly select the "best of the worst" pairing from a set of features that are utterly visually incompatible.

This paper presents an improvement of the aforementioned algorithm that corrects the second problem and minimizes the first one. We introduce a main loop that searches for *several* groups of transformation bounds, thus segmenting the images in regions for which geometric transformations are locally consistent. This makes our algorithm able to deal with dynamic scenes, since the region of the image corresponding to each moving object fits into the concept of local consistent geometric transformation. Also, we reject a pairing between two points if there exists an unbounded correspondence for which the descriptors are much more similar.

The rest of this paper is organized as follows: Section II gives an overview of the main issues behind feature descriptors and the matching process; Section III presents the proposed methodology, which is evaluated by several experiments whose results are shown in Section IV. Finally, Section V concludes with a discussion of the results and possible further directions.

II. Feature descriptors and matching

Given an input image q, a feature descriptor algorithm outputs a set of features, $\{F_{q,n}\}$, where each feature $F_{q,n}$ comprises the descriptor vector among some geometric information associated with the feature. Typically, most affine-invariant feature descriptors give at least the following data:

$$F_{q,n} = \langle x_{q,n}, y_{q,n}, s_{q,n}, \phi_{q,n}, \mathbf{d}_{q,n} \rangle, \quad (1)$$

where $(x_{q,n}, y_{q,n})$ are the coordinates of the feature centroid, in pixels; $s_{q,n}$ is a scale factor; $\phi_{q,n}$ is the orientation of some particular characteristic of the feature; and $\mathbf{d}_{q,n}$ is the descriptor vector. Traditionally, only the descriptor vectors are used to match pairs of points of two input images, as described below:

- Given the set of descriptors of all distinct features found in both images, $\mathcal{D}_q = \{\mathbf{d}_{q,1}, \ldots, \mathbf{d}_{q,N_q}\}$ for $q \in \{1, 2\}$, where $\mathbf{d}_{*,*}$ are vectors with K elements and N_q is the number of features detected in image q;
- given a *distance function*, $\mathrm{dist} : \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{R}^+$ (possibly, but not necessarily, the Euclidean distance), where $\mathbb{R}^+$ is the set of non-negative real numbers;
- for each descriptor $\mathbf{d}_{1,i}$, search for a $\mathbf{d}_{2,j}$ that minimizes the distance $\mathrm{dist}(\mathbf{d}_{1,i}, \mathbf{d}_{2,j})$.

The output is a set of pairs of indexes $\langle i, j \rangle$ of corresponding features. Formally, the set $\mathcal{M}$ of all pairs can be described by:

$$\mathcal{M} = \Big\{ \langle i, \arg\min_j \mathrm{dist}(\mathbf{d}_{1,i}, \mathbf{d}_{2,j}) \rangle \mid \forall k \neq j \big[\mathrm{dist}(\mathbf{d}_{1,i}, \mathbf{d}_{2,k}) \geq \tau \, \mathrm{dist}(\mathbf{d}_{1,i}, \mathbf{d}_{2,j}) \big] \Big\} \quad (2)$$

for a given *distinctiveness threshold* $\tau \geq 1$ that avoids ambiguous matches and minimizes matchings between not-much-alike features (i.e., it prevents forced pairing of features $F_{1,i}$ that have no particularly good correspondences in the other image). Unfortunately, the adoption of a distinctiveness threshold also avoids pairing of features that appear more than one time in the images, such as recurrent patterns or homogeneous textures. Ideally, matchings should be searched for all rich features, no matter how many times they appear replicated in the images. So, a better approach would be to rely on other constraints while keeping $\tau = 1$, effectively allowing matchings of repeated features.

Regarding the geometric data presented in Eq. (1), in general the scale factor $s_{q,n}$ and the orientation angle $\phi_{q,n}$ may have no meaning in image space, serving only to compare the geometry among local patches (around the neighborhood of $(x_{q,n}, y_{q,n})$) in both images. On the other hand, it also means that all "good" matches between two images can be grouped into regions for which all scale ratios $s_{2,j}/s_{1,i}$ and difference of orientations $\phi_{2,j} - \phi_{1,i}$ are somewhat consistent and therefore bounded. Intuitively there should also be some consistency between translation of the coordinates, $x_{2,j} - x_{1,i}$ and $y_{2,j} - y_{1,i}$ (albeit not trivially bounded, since translation may be embedded in a more general perspective transformation of each image region). The identification of these limits is the inspiration of our methodology, which is presented next.

III. Methodology

Our methodology is depicted in Figure 1, and it works as follows. Given $\mathcal{M}_{\text{init}} = \{\langle i, j \rangle\}$, the set of matchings

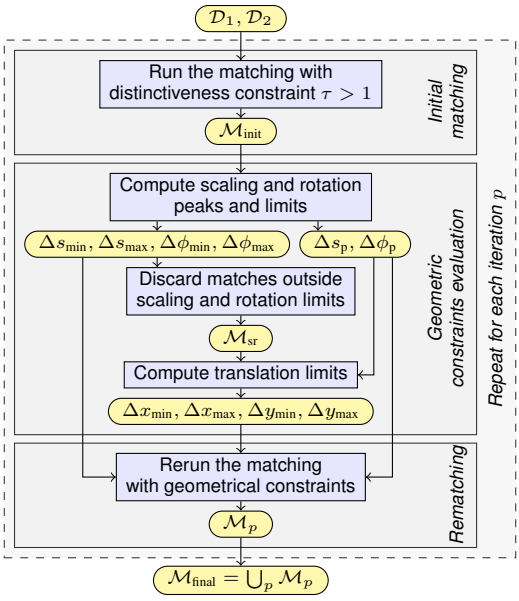

Figure 1. Overview of our methodology. From an initial matching set, we first analyze scaling, rotation and translation to evaluate the geometric constraints, then the matching is rerun under these constraints.

generated by any matching method (e.g., Eq. (2)), we first search for a region-wise consensus for scaling, rotation, and translation of the image features and define "acceptable" ranges for these transformations. We then build a new set of matchings based solely on geometric constraints (as if the distinctiveness threshold τ was set to 1). This entire procedure is repeated several times, in order to (ideally) cover all regions of the first image for which there are corresponding regions on the second image.

While Eq. (2) can be used to evaluate the initial set of matched features $\mathcal{M}_{\text{init}}$, this would also cause our algorithm to be slower than current methods (since the total time required to run our methodology includes the time spent to compute $\mathcal{M}_{\text{init}}$). A more time-efficient method to evaluate the initial set is discussed next. Further details on the following steps will be described in Subsections III-B to III-D.

A. Initial matching

Evaluation of a matching set according to Eq. (2) has asymptotic time complexity of $O(KN_1N_2)$, where K is the descriptor vector length and N_1 and N_2 are the number of features found in each input image respectively. While reducing the asymptotic cost profile is not a trivial task, we can actually reduce the computational cost by taking only a subset of descriptors of $\mathcal{D}_1$ to evaluate $\mathcal{M}_{\text{init}}$. Although the resulting matching set is probably smaller than that evaluated by Eq. (2), this does not compromise the rest of the methodology, since $\mathcal{M}_{\text{init}}$ is used only to compute the limits for the rematching step.

We define $\mathcal{Z}$ as a random subset of $\mathcal{D}_1$, where the number

of elements of $\mathcal{Z}$ is defined as

$$|\mathcal{Z}| = \left\lfloor \frac{|\mathcal{D}_1|}{z} \right\rfloor \tag{3}$$

for a given *subsampling factor* $z \geq 1$. Then we redefine $\mathcal{M}_{\text{init}}$ by adapting the definition presented in Eq. (2):

$$\begin{aligned}\mathcal{M}_{\text{init}} = \Big\{ \langle i, \arg\min_j \text{dist}(\mathbf{d}_{1,i}, \mathbf{d}_{2,j}) \rangle \mid \\ \forall k \neq j \big[\text{dist}(\mathbf{d}_{1,i}, \mathbf{d}_{2,k}) \geq \tau \, \text{dist}(\mathbf{d}_{1,i}, \mathbf{d}_{2,j}) \big] \Big\} \\ \forall \mathbf{d}_{1,i} \in \mathcal{Z}. \end{aligned} \tag{4}$$

While the asymptotic time complexity is still the same (since $O(K\frac{N_1}{z}N_2) = O(KN_1N_2)$), the actual time required to compute $\mathcal{M}_{\text{init}}$ as in Eq. (4) will be significantly reduced for a large z when compared to the original proposition shown in Eq. (2). The optimal determination of z will not be addressed in this paper and will be left as future work.

B. Geometric constraints evaluation

The objective of this step is to define acceptable ranges for all geometric transformations considered here: scaling, rotation, and translation. The first two transformations (scaling and rotation) can be directly estimated from the scaling factors s and orientations ϕ computed by the feature detector (Eq. (1)). However, feature translations in image coordinates can only be consistently estimated after determining the other transformations. Thus, the evaluation of geometric constraints is performed in two steps: (i) scale and rotation and (ii) translation.

1) Scale and rotation constraints evaluation: For scale and rotation constraints evaluation, we first analyze the scaling ratios and rotation angles that occur between matched features. We then define an acceptable range for these transformations and discard all pairs that fall outside these constraints.

For scaling ratios, we proceed as follows: Given the set of all scaling ratios, represented by $\mathcal{S}$ and defined as:

$$\mathcal{S} \triangleq \big\{ s_{2,j}/s_{1,i} \big\} \quad \forall \langle i,j \rangle \in \mathcal{M}_{\text{init}}, \tag{5}$$

we first estimate the Probability Density Function (PDF) of $\mathcal{S}$ using Parzen windows [10] with a Gaussian kernel. For the bandwidth value we use the automated method discussed in [11]. From the estimated PDF we pick the following values:

- the mode of scale ratio (the value at the PDF's peak), Δs_{p};
- the largest scale ratio below Δs_{p} for which the likelihood equals 5% of the mode, Δs_{min}; and
- the smallest scale ratio above Δs_{p} for which the likelihood equals 5% of the mode, Δs_{max}.

The same procedure is applied to find the peak and limits of the rotation angle (difference of orientations). Given the set of all rotation angles, $\mathcal{R}$, defined as:

$$\mathcal{R} \triangleq \big\{ \phi_{2,j} - \phi_{1,i} \big\} \quad \forall \langle i,j \rangle \in \mathcal{M}_{\text{init}}, \tag{6}$$

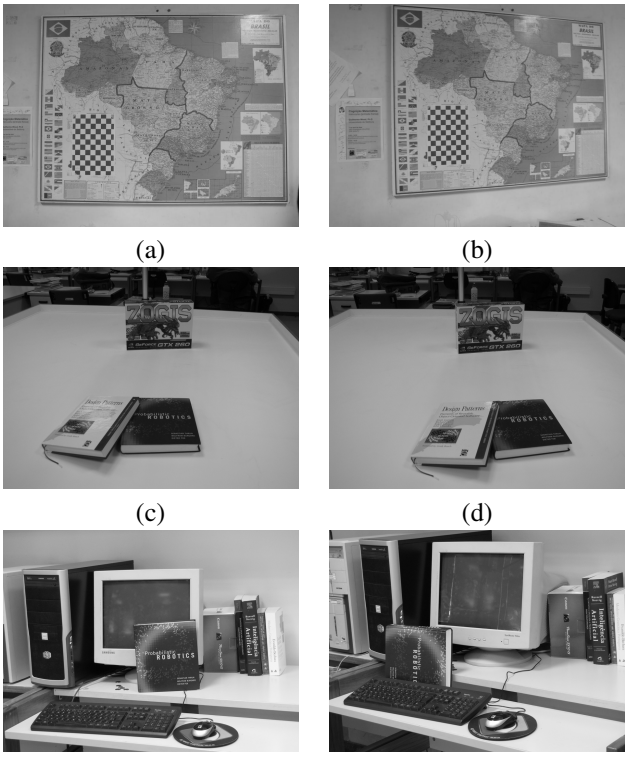

Figure 2. Test cases evaluated in this paper: (a)-(b) WallMap: point-of-view change of planar scene (ground truth homography is available); (c)-(d) BooksInTable: groups of objects in distinct depths; (e)-(f) MovingBook: non-planar, dynamic scene.

we estimate the PDF of $\mathcal{R}$, again using Parzen windows. However, since angles form a cyclic group, the estimated PDF must be confined to a (2π rad)-wide window centered on the likelihood peak $\Delta\phi_{\text{p}}$, i.e., the PDF is defined over the domain $(\Delta\phi_{\text{p}} - \pi \text{ rad}, \Delta\phi_{\text{p}} + \pi \text{ rad}]$. The acceptable range limits $\Delta\phi_{\text{min}}$ and $\Delta\phi_{\text{max}}$ are defined from this PDF exactly as Δs_{min} and Δs_{max}.

Finally, we build the set $\mathcal{M}_{\text{sr}} \subseteq \mathcal{M}_{\text{init}}$ containing only matching pairs whose scaling and rotation transformations fall within the limits previously defined:

$$\begin{aligned}\mathcal{M}_{\text{sr}} \triangleq \big\{ \langle i,j \rangle \in \mathcal{M}_{\text{init}} \mid \\ (\Delta s_{\text{min}} \leq s_{2,j}/s_{1,i} \leq \Delta s_{\text{max}}) \wedge \\ \wedge (\Delta\phi_{\text{min}} \leq \phi_{2,j} - \phi_{1,i} \leq \Delta\phi_{\text{max}}) \big\}, \end{aligned} \tag{7}$$

where $\phi_{2,j} - \phi_{1,i}$ is normalized to the interval $(\Delta\phi_{\text{p}} - \pi \text{ rad}, \Delta\phi_{\text{p}} + \pi \text{ rad}]$.

2) Translation constraints evaluation: As explained before, translation is analyzed on the transformed (scaled and rotated) feature coordinates. For each pair $\langle i,j \rangle \in \mathcal{M}_{\text{sr}}$, we define the *transformed displacement vector* $\vec{\mathbf{v}}_{i,j}$ as:

$$\vec{\mathbf{v}}_{i,j} \triangleq \begin{bmatrix} \Delta x_{i,j} \\ \Delta y_{i,j} \end{bmatrix} \triangleq \begin{bmatrix} x_{2,j} \\ y_{2,j} \end{bmatrix} - \Delta s_{\text{p}} \begin{bmatrix} \cos\Delta\phi_{\text{p}} & -\sin\Delta\phi_{\text{p}} \\ \sin\Delta\phi_{\text{p}} & \cos\Delta\phi_{\text{p}} \end{bmatrix} \begin{bmatrix} x_{1,i} \\ y_{1,i} \end{bmatrix}. \tag{8}$$

Translation limits are defined as follows: Given a two dimensional histogram over x and y axes of vectors $\vec{\mathbf{v}}_{i,j}$, the rectangle that covers the connected island of all histogram bins around the histogram peak defines the acceptable ranges for both axes: $[\Delta x_{\text{min}}, \Delta x_{\text{max}}]$, and $[\Delta y_{\text{min}}, \Delta y_{\text{max}}]$.

C. Rematching

From the constraints evaluated during the previous step, we now build the set of matching pairs $\mathcal{M}_p$ for the p-th region:

$$\begin{aligned}\mathcal{M}_p \triangleq \big\{ \langle i, \arg\min_j \operatorname{dist}(\mathbf{d}_{1,i}, \mathbf{d}_{2,j}) \rangle \mid \\ (\Delta s_{\text{min}} \leq s_{2,j}/s_{1,i} \leq \Delta s_{\text{max}}) \wedge \\ \wedge\, (\Delta \phi_{\text{min}} \leq \phi_{2,j} - \phi_{1,i} \leq \Delta \phi_{\text{max}}) \wedge \\ \wedge\, (\Delta x_{\text{min}} \leq \Delta x_{i,j} \leq \Delta x_{\text{max}}) \wedge \\ \wedge\, (\Delta y_{\text{min}} \leq \Delta y_{i,j} \leq \Delta y_{\text{max}}) \wedge \\ \wedge\, (\operatorname{dist}(\mathbf{d}_{1,i}, \mathbf{d}_{2,j}) \leq \eta \operatorname{dist}(\mathbf{d}_{1,i}, \mathbf{d}_{2,k})) \big\} \\ \forall i \in \{1, \ldots, N_1\} \text{ and } j, k \in \{1, \ldots, N_2\}. \end{aligned} \quad (9)$$

The most important aspect of Eq. (9) is the last acceptance condition: it states that a pair $\langle i, j \rangle$ will be accepted only if the distance between their descriptors are not much larger (based on a threshold $\eta \geq 1$) than the distance from $\mathbf{d}_{1,i}$ to *any other* descriptor of the second image. In other words, if a pair $\langle i, k \rangle$ is significantly more visually compatible than the candidate pair $\langle i, j \rangle$ (and, of course, $\langle i, k \rangle$ does not meet the transformation bounds), then the candidate is rejected.

D. Multiple region correspondence

The matchings recovered by the previous steps may not cover the entire image space. In order to provide a more comprehensive coverage, first one needs to remove all matched features from the input sets,

$$\left.\begin{array}{l} \mathcal{D}_1 \leftarrow \mathcal{D}_1 - \{F_{1,i}\} \\ \mathcal{D}_2 \leftarrow \mathcal{D}_2 - \{F_{2,j}\} \end{array}\right\} \forall \langle i, j \rangle \in \mathcal{M}_p, \quad (10)$$

and then repeat the entire procedure (Subsections III-A to III-C) a finite number of times. After the execution of all iterations, the final set of matchings, $\mathcal{M}_{\text{final}}$, is the union of the matchings $\mathcal{M}_p$ obtained in each iteration. The ideal number of iterations depends on several factors and will not be addressed in this paper.

IV. Experiments and Results

In this section we compare three matching methods: the classical one, represented by Eq. (2); the original algorithm previously published by the authors [9], referred here as "Original"; and the improved version proposed in this paper. They were all implemented in Matlab, with some routines implemented in C++ for the sake of speed. The test platform consists of an Intel® 2.67GHz Core™ i7 processor with 12GB of RAM, running a 64-bit GNU/Linux Ubuntu box (kernel v2.6.28). Images were taken using a Canon PowerShot SX10 IS digital camera with 10 megapixels and subsampled to 1/16 of the original dimensions, resulting in images of 912×684 pixels.

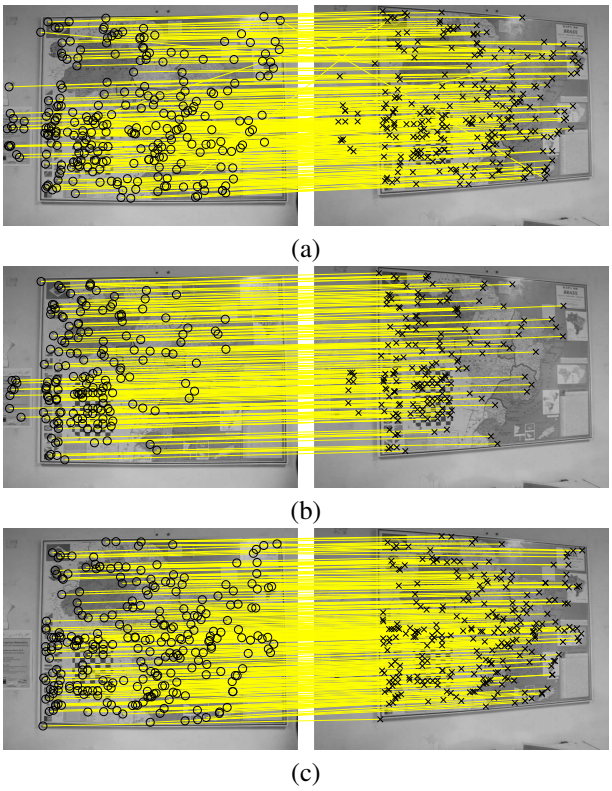

Figure 3. Results of the WallMap test case with (a) classical method, (b) original method with $z = 1/2$, and (c) improved method with 2 iterations and $z = 1/2$. For visualization purposes, only 10% of the matches were plotted in each case. Notice that the classical method matches gives a number of severe outliers. The original method fails to capture the essence of the perspective transformation and (incorrectly) matches only a small number of features in the right portion of the images. Our improved method behaves much like the classical method, but does not generate visible outliers.

Some images of different scenes, shown in Figure 2, were used to compare our proposed method with the others. Pairs of these images compose the test cases evaluated in our experiments. Notice that these images were selected to compare the efficiency of the three methods in face of perspective transformations (changes in the point-of-view) and dynamic scenes. In all cases, feature sets $\mathcal{D}_1$ and $\mathcal{D}_2$ are evaluated using the SIFT algorithm [1], [2].[1]

The set of matchings for each test set are presented in Figures 3, 4 and 5. All tests show a consistent behavior: While the classical matching method is able to match a large number of features, outliers occur frequently. The original method fails to provide extensive coverage of the images, since the underlying geometric model cannot cope with general perspective transformations. On the other hand, the improved method not only provides extensive coverage of the images,

[1] Although we use SIFT algorithm in our experiments, please recall that our methodology is not specifically tied to it. Any feature descriptor algorithm that outputs the data presented in Eq. (1) can be used instead.

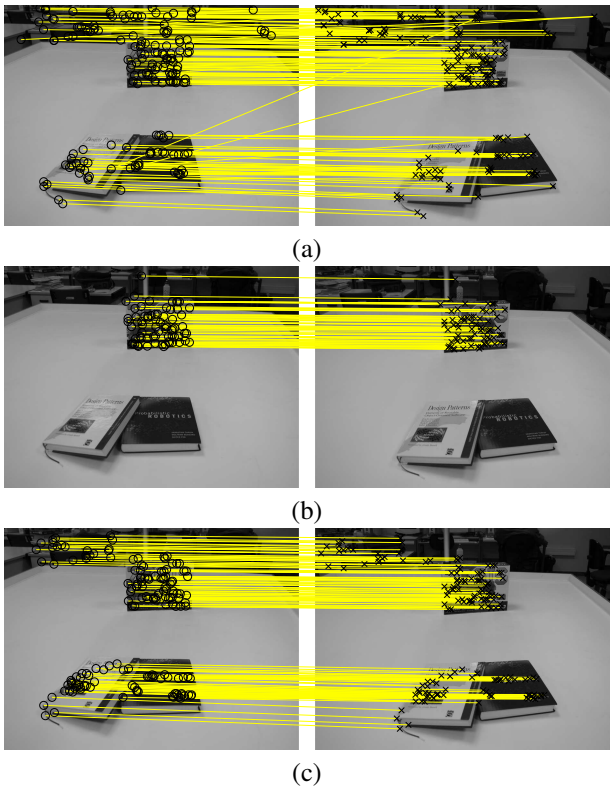

Figure 4. Results of the BooksInTable test case with (a) classical method, (b) original method with $z = 1/2$, and (c) improved method with 3 iterations and $z = 1/2$. For visualization purposes, only 25% of the matches were plotted in each case. Notice that in this case the classical method behaves very well, except for some outliers. The original method only matches the background object. Our improved method selects three groups of correspondences (one per iteration), effectively segmenting the image into regions of objects for which the distances to the camera are locally consistent.

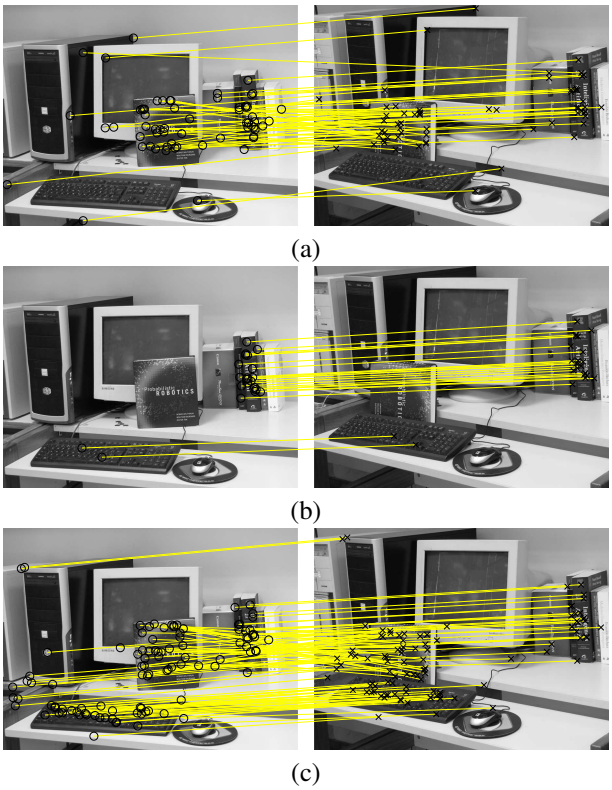

Figure 5. Results of the MovingBook test case with (a) classical method, (b) original method with $z = 1/20$, and (c) improved method with 4 iterations and $z = 1/20$. For visualization purposes, only 10% of the matches were plotted in each case. Notice that the classical method matches both the static (computer) and dynamic (book) portions of the scene, but also gives a number of severe outliers. On the other hand, the original method only matched a small, static portion of the scene. Our improved method correctly finds matches in the entire image space.

but also avoids severe mismatches.

Table I shows the number of matches obtained with each method for each test case after 30 runs and respective timings. For the WallMap test case, the mean number of matches is similar for all methods. However, notice that the random selection of features in the original method can significantly affect the number of matches, which ranges from less than 1,400 to more than 3,400. Our methodology behaves much more steadily in this matter. In the remaining test cases the original method behaved poorly when compared to the other two, since the geometric transformation between the image sets cannot be accurately represented by a single bounded affine transform. On the other hand, the time spent by our improved algorithm suggests that the original algorithm may be a better choice if speed is a main concern and if the images do not undergo significative perspective transformations.

The WallMap test case consists of two images with a checkerboard pattern with known size. This allows for a direct quantitative evaluation of the results based on image homography. Thus, we define the *error vector* $\vec{\epsilon}_{i,j}$ as the difference between observed (from feature detector and matcher) and predicted (from homography) coordinates:

$$\vec{\epsilon}_{i,j} \triangleq \begin{bmatrix} x_{2,j} \\ y_{2,j} \end{bmatrix} - H\left(\begin{bmatrix} x_{1,i} \\ y_{1,i} \end{bmatrix} \right), \tag{11}$$

Table I
MEAN, MINIMUM AND MAXIMUM NUMBER OF MATCHES AND EXECUTION TIME OBTAINED WITH EACH METHOD IN EACH TEST CASE AFTER 30 RUNS.

		Number of matches			*Mean exec.*
Test case	*Method*	*Mean*	*Min*	*Max*	*time [s]*
WallMap	Classical	2488	2488	2488	5.75
	Original	2688	1364	3475	3.14
	Improved	2652	2518	2840	13.39
BooksInTable	Classical	669	669	669	0.48
	Original	268	235	284	0.37
	Improved	606	375	844	1.58
MovingBook	Classical	625	625	625	0.93
	Original	250	134	428	0.64
	Improved	776	526	1707	4.58

Table II
STATISTICS OF ERROR MEASUREMENTS OBTAINED FOR EVALUATED METHODS.

	Error measurements	
Method	*RMSE [px]*	*MAE [px]*
Classical	80.88	21.71
Original	12.13	5.99
Improved	7.25	4.85

where $H : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ is the homography function which maps two dimensional coordinates from the first image to the coordinate space of the second image. The homography is estimated using Bouguet's *Camera Calibration Toolbox for Matlab* [12] based on the checkerboard calibration patterns present in the images.

Based on the set of error vectors we analyze two global error measurements: the *Root Mean Square Error*, RMSE($\{\vec{\epsilon}_{i,j}\}$), and the *Mean Absolute Error*, MAE($\{\vec{\epsilon}_{i,j}\}$). Smaller values obtained for RMSE and MAE indicate better results. RMSE is particularly sensitive to outliers (i.e., a single big value of $\|\vec{\epsilon}_{i,j}\|$) than MAE. Results are presented in Table II. The expressive decrease of the RMSE measurement clearly shows that the general transformation estimated by our method is closer to the recovered homography than those estimated by other methods.

V. CONCLUSIONS AND FUTURE WORK

We have presented a novel methodology for feature matching in a pair of images that provides more accurate results. Compared to current methods based on similarity between descriptor vectors alone, our approach is able to significantly reduce the occurrence of mismatches, as evidenced by the considerable decrease in both RMSE and MAE error measurements. Since RMSE is very sensitive to outliers, the expressive decrease in its magnitude, when compared to typical approaches in the literature, is a clear indication that occurrence of feature mismatch is much less frequent in the proposed method. Also, as we have shown, the increase in the number of successful matches suggests that other techniques are likely to discard correct matches due to the use of a distinctiveness threshold. This undesirable effect is not present in the proposed method, since no distinctiveness threshold is used.

Currently our method depends on the manual determination of the number of iterations, which limits its applicability to completely automated tasks. It is possible to circumvent this limitation by analyzing the number of new correspondences obtained in each iteration, thus stopping the loop if the probability of further growing the number of correspondences is small. The determination of this criteria is currently being addressed by the authors. Another possible approach is to use clustering techniques to segment the image prior to the main loop and perform each iteration based on the transformation parameters corresponding to each cluster. Also, it is worth noting that the current implementation of our methodology is largely unoptimized and thus it takes much more time to execute than the compared methods.

ACKNOWLEDGMENTS

This work was partially supported by *Fundação Centro de Análise, Pesquisa e Inovação Tecnológica (Analysis, Research and Technological Innovation Center) — FUCAPI*, Manaus, AM, Brazil.

REFERENCES

[1] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. ICCV*, vol. 2, 1999, pp. 1150–1157.

[2] ——, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004. [Online]. Available: citeseer.ist.psu.edu/lowe04distinctive.html

[3] Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," in *Proc. CVPR*, vol. 2, 2004, pp. 506–513. [Online]. Available: citeseer.comp.nus.edu.sg/688805.html

[4] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *PAMI*, vol. 27, no. 10, pp. 1615–1630, 2005. [Online]. Available: http://lear.inrialpes.fr/pubs/2005/MS05

[5] H. Bay, T. Tuytelaars, and L. J. V. Gool, "SURF: Speeded Up Robust Features," in *Proc. ECCV*, A. Leonardis, H. Bischof, and A. Pinz, Eds., 2006, pp. 404–417.

[6] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image and Vision Computing*, vol. 22, no. 10, pp. 761–767, 2004.

[7] J.-M. Morel and G. Yu, "ASIFT: A new framework for fully affine invariant image comparison," *SIAM J. of Imaging Sciences (SIIMS)*, vol. 2, no. 2, pp. 438–469, 2009.

[8] 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*, vol. 24, no. 6, pp. 381–395, 1981. [Online]. Available: http://portal.acm.org/citation.cfm?id=358669.358692

[9] V. F. da Camara Neto and M. F. M. Campos, "An improved methodology for image feature matching," in *Proc. SIBGRAPI*, 2009.

[10] E. Parzen, "On estimation of a probability density function and mode," *Annals of Math. Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[11] Z. I. Botev, "A novel nonparametric density estimator," The University of Queensland, Tech. Rep., 2006.

[12] J.-Y. Bouguet, "Camera calibration toolbox for Matlab," Available at http://www.vision.caltech.edu/bouguetj/calib_doc, 2008.