From Features to Attribute Graphs for Point Set Registration

Carlos Orrite(✉), Elias Herrero, and Mauricio Valencia

Instituto de Investigacion en Ingenieria de Aragon, University of Zaragoza,
Zaragoza, Spain
{corrite,jelias}@unizar.es, A01136483@itesm.mx
http://i3a.unizar.es/

Abstract. The traditional approach for point set registration is based on matching feature descriptors between the target object and the query image and then the fundamental matrix is calculated robustly using RANSAC to align the target in the image. However, this approach can easily fail in the presence of occlusion, background clutter and changes in scale and camera viewpoint, being the RANSAC algorithm unable to filter out many outliers. In our proposal the target is represented by an attribute graph, where its vertices represent salient features describing the target object and its edges encode their spatial relationships. The matched keypoints between the attribute graph and the descriptors in the query image are filtered taking into account features such as orientation and scale, as well as the structure of the graph. Preliminary results using the Stanford Mobile Visual search data set and the Stanford Streaming Mobile Augmented Reality Dataset show the best behaviour of our proposal in valid matches and lower computational cost in relation to the standard approach based on RANSAC.

Keywords: Graphs · Matching · RANSAC

1 Introduction

A large number of vision applications, such as visual correspondence, object matching, 3D reconstruction and motion tracking, rely on matching keypoints across images. Effective and efficient generation of keypoints from an image is a well-studied problem in the literature and is the basis of many Computer Vision applications. In spite of the large literature dealing with this issue, it remains a challenging topic for achieving stable and reliable matching results in a complex situation, facing illumination variation, shape and scale change, background clutter, appearance change, partial occlusions, etc.

The standard approach for feature matching relies on some steps. The first one is to find feature points in each image. Next, keypoint descriptors are matched for each pair of images using the approximate nearest neighbour (ANN). After matching features for an image pair the fundamental matrix is robustly

A. Morales et al. (Eds.): IbPRIA 2019, LNCS 11867, pp. 437–448, 2019.
https://doi.org/10.1007/978-3-030-31332-6_38

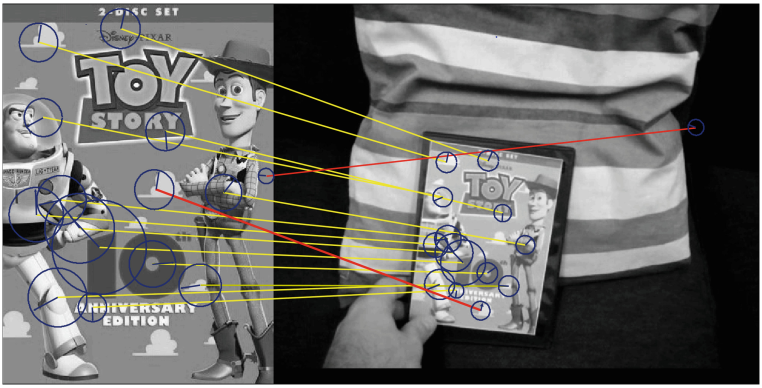

Fig. 1. Point set registration found with RANSAC. Feature descriptors are represented by a circle, pointing to the scale, and an arrow indicating the orientation. Inlier matches are represented in yellow, while the outliers are in red. (Color figure online)

estimated for the pair using Random Sample Consensus (RANSAC) algorithm [1] in order to determinate in-liners. Although RANSAC works well in many cases, there is no guarantee that it will obtain a reasonable solution even if there exists one. It can also be hard to determine if there is no solution at all.

Matching between images is accomplished by feature descriptors. It provides a list of candidate matching points between the object and the descriptors in the image. However, there exists some other keypoint attributes, such as scale or orientation, useful for pruning false matches. Inspired in fingerprint recognition we follow a similar feature matching process to remove wrong correspondences. So, when matching keypoints the correspondence among them has to be in accordance with the change of scale and orientation between both images. Figure 1 shows the matches after RANSAC between two images where some outliers are still present. Feature points in both images are represented by a circle, whose size is proportional to its scale, and an arrow that shows the main orientation of the descriptor. It is easily seen how the extreme outlier, depicted in red, can be removed as it does not exhibit a similar change in scale or orientation between both images.

However, this filtering process may be not enough as it is based on point to point correspondences without taking into account the overall structure of the object. Some authors propose the modelling of the object by an attribute graph, where keypoints provided by the feature detector constitute the vertices of the graph. The topological relationship of these keypoints are preserved by the edge interconnections. The structure of the graph has itself a relevant information about the object and can be useful for filtering mismatches as we propose in this paper. Figure 3 shows a mismatch between three points. Orientation and change in scale is similar for the matched points, as they correspond to a similar cloud in the image. However, from a structural point of view, both triangles are not matched as they correspond to a different graph structure.

The main contributions of this paper are: the generation of an Attribute Graph (AG) and a matching filtering based on attribute and structure which overcome the limitations of RANSAC, mainly to assess if there is no matching between two images.

2 Related Work

2.1 Local Features from Images

Feature detection aims at finding some interesting points (features) in the image such as corners. On the other hand, descriptor extraction aims to represent those interesting points to later compare them with other interesting points (features) in a different image. Current methods for feature matching rely on well-known descriptors for detection and matching as the SIFT and SURF keypoint detector and descriptor, or more recent descriptors such as BRIEF or BRISK which exhibit an acceptable performance with the benefit of a low computational cost, see [2] for a deeper insight in comparison of feature detectors and descriptors for object class matching. Those sparse methods that find interest points and then match them across images become the de facto standard [3].

In recent years, many of these approaches have been revised using deep networks [3,4], which has led to a revival for dense matching [5,6]. Despite the great expectation raised by dense methods, they still tend to fail in complex scenes with occlusions.

Regardless the learning procedure used to get local features, sparse or dense methods, this paper tackles with the problem of how to combine them in a higher structure to cope with the limitations that both approaches still suffer.

2.2 Image-to-Image Alignment

Another problem that has attracted increasing attention over the past years is the localization problem, i.e. estimating the position of a camera giving an image. Several approaches have been suggested to solve this problem. Many of them have adopted an image retrieval approach, where a query image is matched to a database of images using visual features. Sometimes this is combined with a geometric verification step, but in many cases the underlying geometry is largely ignored, see [7].

Some works afford the representation of the target object by an attributed graph, where its vertices represent feature descriptors and its edges encode their spatial relationship which has already been proposed in [8]. The authors propose an attribute graph to represent the structure of a target object in the problem of tracking it along time. The structure of the model yields in the change of shape when the object moves, adapting continuously to the edge ratio in each triangle of the graph. As in many other computer vision problems when adapting the model according to the appearance of the object it can easily be degenerated to loose the target when the model no longer fit to the object.

2.3 Filtering Correspondences

Image alignment and structure-from-motion methods often use RANSAC to find optimal transformation hypotheses [1]. In the last decades many variants of the original RANSAC procedure have been proposed, we refer the reader to [9] for a performance evaluation of these methods.

A major drawback of these approaches is that they rely on small subsets of the data to generate the hypotheses, e.g., the 5-point algorithm or 8-point algorithm to retrieve the essential matrix. It requires that most false matches have to be removed in advance. As image pairs with large baselines and imaging changes will contain a large percentage of outliers, it makes RANSAC to fail in these kind of situations.

Recent works try to overcome this limitation by simultaneously rejecting outliers and estimating global motion. GMS [10] divides the image into multiple grids and forms initial matches between the grid cells. Although they show improvements over traditional matching strategies, the piecewise smoothness assumption is often violated in practice.

As mentioned previously, traditionally hand-crafted feature descriptions have been replaced by deep learning ones, which can be trained in an end-to-end way. However, RANSAC has not been used as part of such deep learning pipelines, because its hypothesis selection procedure is non-differentiable. As far as our knowledge, DSAC [11] is the only work to tackle spare outlier rejection in a differentiable way. However, this method is designed to mimic RANSAC rather than to outperform it, as we propose in this paper. Furthermore, it is specific to 3D to 2D correspondences, rather than point set registration [12].

3 Our Approach

3.1 Attribute Graph

Given a target image and a set of keypoints and its corresponding descriptors we first model the target by an attribute graph. Triangles are 2D entities that are able to describe the geometry of planar objects and more complex objects by a triangle mesh. In our approach, the object is modelled by a triangle mesh where each point in the mesh corresponds to a keypoint, which has associated a feature descriptor.

Formally speaking, an attributed graph G consists of a set of vertices V, which are connected via a set of edges E. The edges E are inserted following the rules of the Delaunay triangulation. Hence, there is also a set of triangles F, where $c : F \longrightarrow V^3; c(f) = \{v_1, v_2, v_3\}$. The model stores attributes with vertices and triangles.

Attributes of Vertices: Each vertex $v \in V$ stores a set of attributes $\{\mathbf{p}, \beta, s\}$.

$\mathbf{p}$: $\mathbf{p}(v) = \{x, y\}^T$ is the 2D position of vertex v.

β: $\beta(v)$ is the orientation provided by the feature detector for this vertex.

s: $s(v)$ is the scale provided by the feature detector for this vertex.

Attributes of Triangles: Each triangle $f \in F$ stores a set of vertices $c(f) = \{v_1, v_2, v_3\}$ and barycentric angles A.

A: $A(f) = \{\alpha_1, \alpha_2, \alpha_3\}^T$ are the angles between any vertex and the barycentric, see Fig. 2.

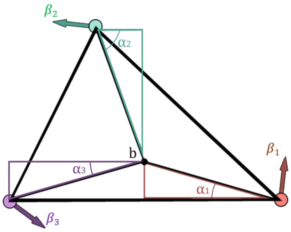

Fig. 2. Attributes for vertices and triangle.

3.2 Attribute Filtering

The standard approach for feature matching basically relies on three steps. The first one is to find feature points in each image (keypoints). Next, for each pair of images, keypoint descriptors are matched for each pair of images using the approximate nearest neighbour. After matching features for an image pair the fundamental matrix is robustly estimated for the pair using RANSAC, removing outliers.

In our proposal we follow the first two steps, i.e., feature detection and matching based on kd-tree and then use the graph structure to remove false matchings. We carry out two filtering processes one after the other. The first one is based on attributes and the other one on structure.

Matching between images is accomplished by feature descriptors. It provides a list of candidate matching points between the object and the descriptors in the image. However, other keypoint attributes, such as scale or orientation, useful for pruning false matches exist.

To recover the orientation between the model and the image the most probable orientation difference is found. The median is obtained and all the correspondences higher, or lower, a threshold from this value are rejected. To do so, the circular statistic is used due to the inherent nature of the data.

Afterwards, the same process is carried out using the scale attribute. For those correspondences exhibiting a difference in scale attribute higher or lower than the median are removed. In this paper we consider a scale threshold corresponding to the double or the half of the median scale change.

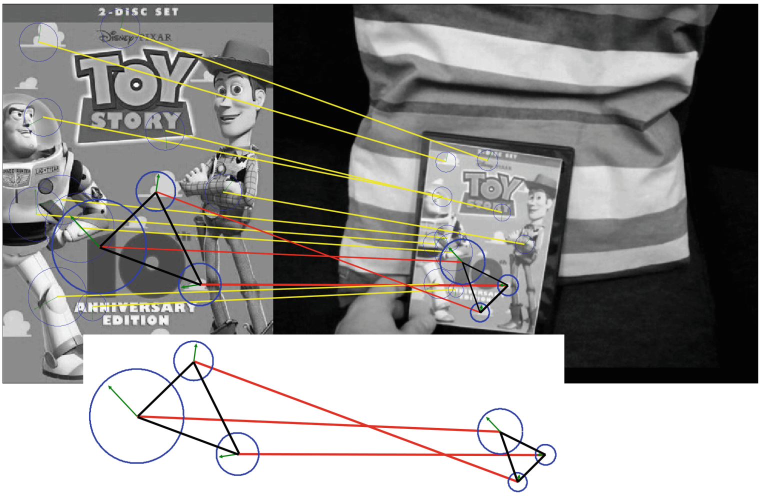

Fig. 3. Mismatch points filtered by structure.

3.3 Structure Filtering

In spite of the attribute filtering some mismatch points, with similar change in orientation and scale, might remain, see Fig. 3 where a solid triangle suffers from a projective change. The structure filtering takes into account the Delaunay triangulation in the graph to analyse the consistency of the matching from a structural point of view. For every triangle in the graph the barycentric is calculated. Afterwards, for each vertex in the triangle we compute the angle between the vector given by the descriptor orientation and the vector joining the vertex with the barycentric. In a similar way, we compute the angle between the orientation of the descriptor in the image and the vector joining that keypoint and the coordinates of its barycentric. Computing the angle descriptor in relation to the barycentric vector is similar to a normalization process and therefore, the difference between both keypoints in the image and the target should be close to 0.

For the example shown in Fig. 3, it can be easily verified that the ordination of the points is not preserved, and therefore, the change of orientation between both triangles will not pass the structure filtering. Next step consists in identifying which is the wrong point in the triangle, or could be more than one. As every triangle is not isolated, but it belongs to a graph, we check for each vertex if it is or not in accordance with its neighbour triangles in the graph. In this way, we obtain a list of triangle orientation for any point in relation to its neighbours. Now we apply the voting rule, so that the point which has received more mismatches will be rejected. The process carries on till all vertices pass the triangle orientation exam.

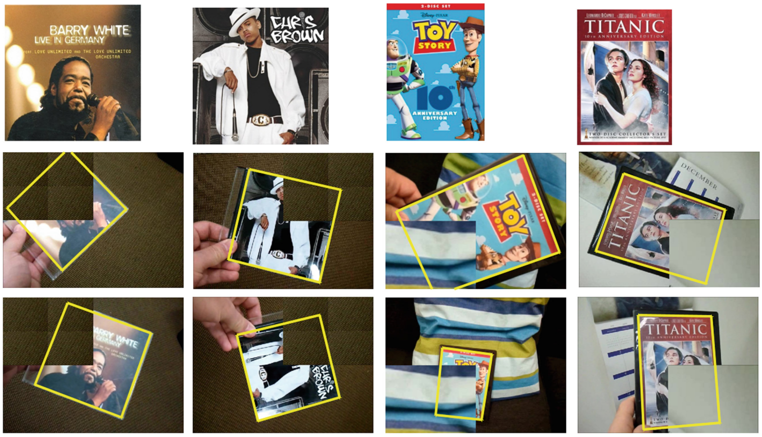

Fig. 4. Object detection in presence of scale and orientation changes, background clutter and partial occlusions for The Stanford Streaming Mobile Augmented Reality Dataset. From left to right: Barry White Moving, Chris Brown Moving, Toy Story Moving and Titanic Moving. At the top, the original DVD covers. Following rows show some frames suffering from different partial occlusions (first frame of the sequence on the middle row, last frame on the bottom row. The rectangular DVD cover is depicted in yellow after the four corners are detected.

4 Experiments

Next, we describe a set of experiments used to verify our theoretical results. We use two datasets: The Stanford Streaming Mobile Augmented Reality Dataset and The Stanford Mobile Visual Search data set. For the former dataset we report the mean error in the estimation of the target bounding box as well as the computational cost to carry out the object location. For the last dataset we report the image match accuracy.

All experiments have been implemented in Matlab. For both datasets we use the SIFT descriptor, running the code provided by [13]. The algorithm provides a feature vector of salient keypoints. For every keypoint the x and y coordinates, the scale and the orientation are given. Additionally, the code provides a 128-d histogram as feature descriptor for every keypoint. It is well known that Hellinger measures outperform Euclidean distance when comparing histograms. So, in our experiments we follow the so called RootSIFT approach proposed by [15] to improve the matching. As the Hellinger distance goes from 0 to 1, we use a 0.95 value as a threshold to valid the feature descriptor matching. Approximated nearest neighbour (ARR) is computed by kd-tree.

In these experiments we compared the performance of the proposed algorithm to that of RANSAC and a couple of its variants, i.e., MSAC, and MLESAC. For all these methods, the number of iterations was set to 100, and in each iteration the fundamental matrix was calculated using the eight-point algorithm, the Chi squared probability threshold for inliers is set to 0.99 and the noise standard

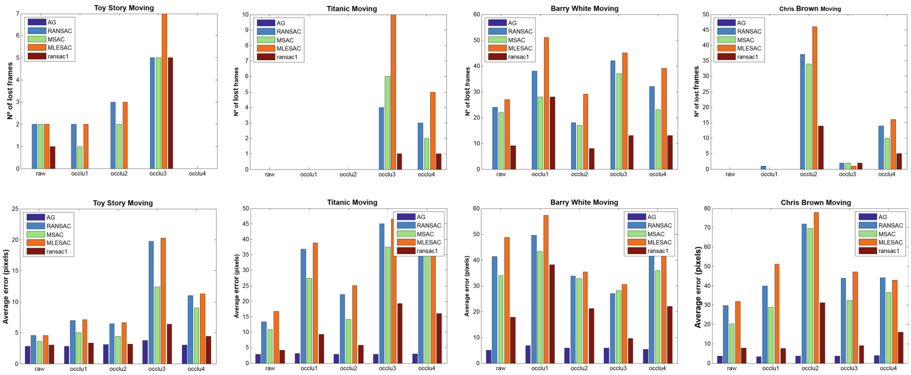

Fig. 5. Top: lost frames in any dataset. Bottom: average error for the four corners obtained from the right matched frames.

deviation to 2, see [14]. Additionally, we use another implementation of RANSAC provided by [16], that we denoted as ransac1.

4.1 The Stanford Streaming Mobile Augmented Reality Dataset

The Stanford Streaming Mobile Augmented Reality Dataset [17] contains 23 different objects of interest, divided to four categories: Books, CD covers, DVD covers and Common Objects.Each video is 100 frames long, recorded at 30 fps with resolution 640 × 480. For each video, we provide a clean database image (no background noise) for the corresponding object of interest. In our experiments we use those videos corresponding to moving objects recorded with a moving camera, i.e., Barry White Moving, Chris Brown Moving, Toy Story Moving, Titanic Moving. These videos help to study the effect of background clutter when there is a relative motion between the object and the background. The dataset authors provide the ground-truth localization information for these videos, where they manually define a bounding quadrilateral around the object of interest in each video frame. To make the experiments even more challenging we have simulated the partial occlusion of the object using a square with a texture similar to that of the target. As the object is being moved along the scene, we check the behaviour of all approaches considering four different occlusion placements, see Fig. 4.

Figures 5 (bottom) show the number of frames where the matching has failed or the error in any of the four corners is longer than the maximum image size, as well as, the mean error, given in pixels, for the four corners corresponding to the correct frames (on the top). Some conclusions can be drawn from this Figure. The most relevant one is that our approach, based on an Attribute Graph (AG) and filtering, outperforms the results provided by the RANSAC algorithm and its variants. No frame is lost in our approach. Besides, the average error for all frames in any dataset, for all occlusions, is lower in AG.

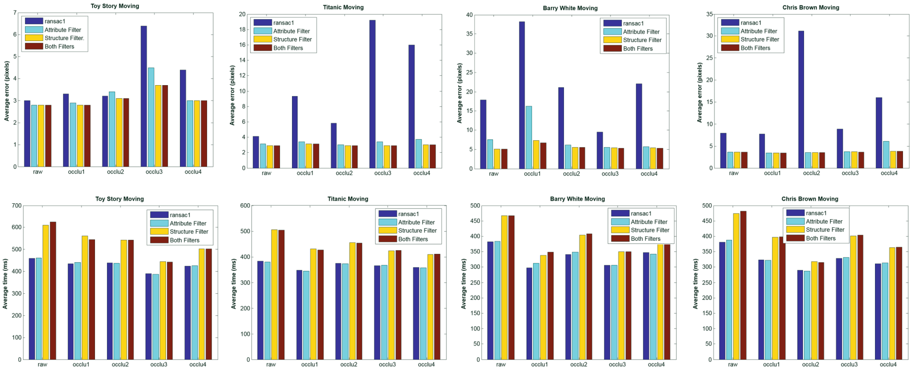

Fig. 6. Applying the attribute and structural filters after RANSAC. On the top: the average error (in pixels) for all datasets. On the bottom: the average computational cost (in ms) taken by any of the proposals.

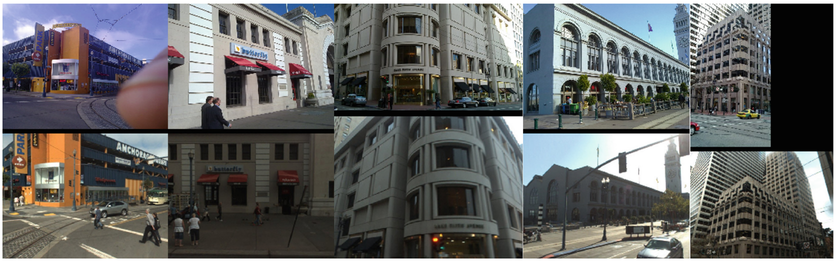

Fig. 7. Landmark dataset. The query image (top), the reference image (bottom)

In relation to the RANSAC algorithms taken from the literature we have found that the so called ransac1 exhibits the best performance among them. Therefore, we have run a second experiment to check the improvement of this algorithm in conjunction with our approach. Figure 6 shows on the top the average error for four different situations: (a) running the RANSAC algorithm alone, (b) applying afterwards the attribute filter or (c) the structure filter and (d) both filters after RANSAC. It is noticeable how the average error of RANSAC is highly reduced after applying our filtering approach.

Figure 6 on the bottom shows the average time (in ms) taken to compute any of these four experiments. As it can be seen, the RANSAC algorithm consumes most of the time, being the attribute filter practically negligible. Therefore, we can pay for a low increase in computational cost to obtain a lower error in localization.

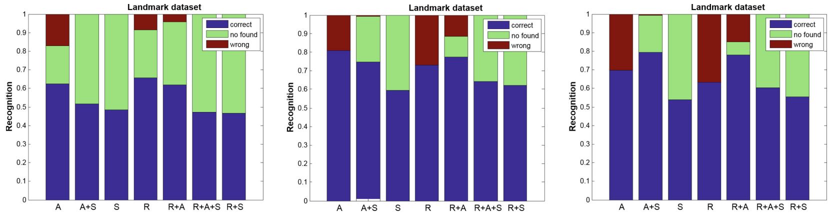

Fig. 8. Matching accuracy for the Landmark dataset for three different thresholds: 0.95; 0.90, 0.85 from left to right

4.2 The Stanford Mobile Visual Search Data Set

Outdoor applications pose additional challenges to those shown in the previous experiments where highly textured rigid planar objects taken under controlled lighting conditions were used. Buildings, on the other hand, tend to have fewer features, exhibit repetitive structures and their 3-D geometric distortions are not captured by simple affine or projective transformations. In order to check our proposal for this kind of applications we have selected The Stanford Mobile Visual Search data set, which exhibits some characteristics lacking in the previous data sets: rigid objects, widely varying lighting conditions, perspective distortion, foreground and background clutter and query data collected from heterogeneous low and high-end camera phones, [18]. The Landmarks dataset has been used for the experiments as it constitutes one of the most challenging dataset in relation to image registration. Figure 7 shows some examples.

By this new set of experiments we intend to asses the performance of the matching filtering processes discussed in this paper, i.e., orientation+scale attribute filtering, structure filtering and RANSAC. In this regard, we run a set of experiments denoted as: A (Attribute filtering); S (Structure filtering); A+S (Structure filtering after Attribute filtering); R (RANSAC); R+A (Attribute filtering after RANSAC); R+S (Structure filtering after RANSAC); R+A+S (Structured filtering after Attribute after RANSAC).

We run a new experiment with the Lanmark dataset for image retrieval taking a query image from the query subset and see which one corresponds to the reference subset. Taking into account the eight-point problem, we consider that if there are not at least eight point correspondences between the query and the reference image we label it as no found. Both subsets, query and reference, have the same number of pictures (N = 501). So far, we have performed several experiments using the Hellinger distance for feature matching with a threshold value equal to 0.95. This parameter can be relevant when dealing with outdoor images which tend to have fewer features and lighting conditions can change significantly at different times of the day. Therefore, in the new experiments three different values have been used, i.e., 0.95, 0.90 and 0.85.

Figure 8 shows the results for the three thresholds under consideration for all the filtering combinations previously mentioned. In order to analyse these graphs we have to consider two aspects: on the one hand, the recognition matching has to be the highest possible and on the other hand, the wrong mismatches have to be the lowest. In this regards we can notice that the attribute filtering and the RANSAC filter exhibit the highest number of wrong mismatches of all approaches. It is worth noticing that the combination of structure filtering after attribute filtering improve the results provided by the structure filtering alone. We can see how A+S outperforms S approach and the same can be said for R+A+S in relation to R+S. Finally, it is shown that the A+S approach outperforms the R+A+S for this dataset.

5 Conclusions

In this paper we have tackled with the problem of feature matching for object detection by modelling the object as an attribute graph. Keypoints provided by the feature detector constitute the vertices of the graph. The topological relationship of these keypoints is preserved by the edge interconnections. Matching filtering takes into account attributes of the nodes, such as orientation or scale, as well as the structure of the graph.

Experiments with the Stanford Streaming Mobile Augmented Reality Dataset have confirmed the better performance of our approach in relation to traditional methods based on RANSAC and variants. Moreover, the matching filtering approach based on attributes and structure has even worked well for matching outdoor images, such as landmarks, where their 3-D geometric distortions are not captured by simple affine or projective transformations. One of the most remarkable property of our approach is that it does not generate false image correspondences, as other RANSAC-based approaches do.

Acknowledgmentes. This work was supported by the Spanish project TIN2017-88841-R (Ministerio de Economa Industria y Competitividad/FEDER, UE).

References

1. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**, 381–395 (1981)
2. Hietanen, A., Lankinen, J., Buch, A.G., Kämäräinen, J.-K., Küger, N.: A Comparison of feature detectors and descriptors for object class matching. Neurocomputing **184**, 3–12 (2016)
3. Ono, Y., Trulls, E., Fua, P., Yi, K.M.: LF-Net: learning local features from images. NIPS 6234–6244 (2018)
4. Detone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: self-supervised interest point detection and description. In: CVPR Workshop on Deep Learning for Visual SLAM (2018)
5. Vijayanarasimhan, S., Ricco, S., Schmid, C., Sukthankar, R., Fragkiadaki, K.: SfM-Net: learning of structure and motion from video. arXiv:1704.07804 (2017)
6. Zhou, T., Brown, M., Snavely, N., Lowe, D.G.: Unsupervised learning of depth and ego-motion from video. In: CVPR-2017 (2017)

7. Svärm, L., Enqvist, O., Oskarsson, M., Kahl, F.: Accurate localization and pose estimation for large 3D models. In: CVPR-2014 (2014)
8. Artner, N.M., Kropatsch, W.G.: Structural cues in 2D tracking: edge lengths vs. barycentric coordinates. In: Ruiz-Shulcloper, J., Sanniti di Baja, G. (eds.) CIARP 2013. LNCS, vol. 8259, pp. 503–511. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41827-3_63
9. Choi, S., Kim, T., Yu, W.: Performance evaluation of RANSAC family. In: BMVC, pp. 1–12 (2009)
10. Bian, J., Lin, W., Matsushita, Y., Yeung, S., Nguyen, T., Cheng, M.: GMS: grid-based motion statistics for fast, ultra-robust feature correspondence. In: CVPR (2017)
11. Brachmann, E., et al.: DSAC - differentiable RANSAC for camera localization. ARXIV (2018)
12. Moo Yi, K., Trulls, E., Ono, Y., Lepetit, V., Salzmann, M., Fua, P.: Learning to find good correspondences. In: CVPR-2018 (2018)
13. Vedaldi, A., Fulkerson, B.: VLFeat: an open and portable library of computer vision algorithms (2008). http://www.vlfeat.org/
14. Zuliani, M.: RANSAC toolbox for matlab (2008)
15. Arandjelović, R., Zisserman, A.: Three things everyone should know to improve object retrieval. In: CVPR (2012)
16. Kovesi, P.: RANSACFITFUNDMATRIX fits fundamental matrix using RANSAC (2005). http://www.csse.uwa.edu.au/
17. Makar, M., Tsai, S.S., Chandrasekhar, V., Chen, D.M., Girod, B.: Inter-frame coding of canonical patches for mobile augmented reality. In: Proceedings of IEEE International Symposium on Multimedia (ISM) (2012)
18. Chandrasekhar, et al.: The stanford mobile visual search data set. In: Proceedings of the Second Annual ACM Conference on Multimedia Systems, pp. 117–122 (2011)