A More Robust Feature Correspondence for More Accurate Image Recognition

Shady Y. El-Mashad* and Amin Shoukry*†
*Computer Science and Engineering Department
Egypt-Japan University for Science and Technology (E-JUST)
Alexandria, Egypt
email:{shady.elmashad, amin.shoukry}@ejust.edu.eg
†Computer and Systems Engineering Department
Alexandria University
Alexandria, Egypt

***Abstract*—In this paper, a novel algorithm for finding the optimal correspondence between two sets of image features has been introduced. The proposed algorithm pays attention not only to the similarity between features but also to the spatial layout of every matched feature and its neighbors. Unlike related methods that use geometrical relations between the neighboring features, the proposed method employes topology that survives against different types of deformations like scaling and rotation; resulting in more robust matching. The features are expressed as an undirected graph where every node represents a local feature and every edge represents adjacency between them. The topology of the resulting graph can be considered as a robust global feature of the represented object. The matching process is modeled as a graph matching problem; which in turn is formulated as a variation of the quadratic assignment problem. In this variation, a number of parameters are used to control the significance of global vs. local features to tune the performance and customize the model. The experimental results show a significant improvement in the number of correct matches using the proposed method compared to different methods.**

***Keywords*-Features Matching; Local Features; Global Features; Topological Relations; Graph Matching; Quadratic Assignment Problem.**

I. Introduction

Image matching or comparing images in order to obtain a measure of their similarity, is an important computer vision task. It is involved in many different applications, for instance object detection and recognition, image classification, content based image retrieval, video data mining, image stitching, stereo vision, and 3D object modelling. A general solution for identifying similarities between objects and scenes within a database of images is still a faraway goal. There are a lot of challenges to overcome such as viewpoint or lighting variations, deformations, and partial occlusions that may exist across different examples [1].

In computer vision, image features are generally classified into two categories which are local and global. A local feature is a property of an object located at a single point or small region. One of the key issues in dealing with local features is that there may be differing numbers of feature points in each image, making comparing images more convoluted. One advantage of using local features is that they may be used to recognize the object despite significant clutter and occlusion. The locality of the features preserves robustness to context change or occlusion, and the representation of the features is invariant to the different geometrical or photometric changes. They also do not require segmentation of the object from the background, unlike many texture features and shape features. In contrast, global features try to cover the information content of an image or an image patch, i.e. all pixels in a region/image are considered. Several object recognition systems use global features that describe an entire image. The majority of shape and texture descriptors fall into this kind. Such features are striking because they create very compact representations of images, where each image corresponds to a point in a high-dimensional feature space. Consequently, any standard classifier can be used. Moreover global features are insightful to clutter and occlusion. Accordingly it is either assumed that an image only contains a single object, or that a good segmentation of the object from the background is available [2].

In many applications, the matching procedure is considered as a crucial preliminary step. This procedure can be used as a pre-processing step to find relevant objects in different images. It can be used before using any geometric consistency algorithm like RANSAC (RANdom SAmple Consensus) [3] or a mean square error minimization [4]. In addition, the matching step can be used not only to find similarities between images, but also to decide whether an object exists or not. This is a useful task especially in the applications which uses a large database [5].

There are two levels to measure the similarity of images which are patch and image levels. In the former level which is patch-level, the distance between any two patches is calculated based on their descriptors. In the image level, the overall similarity between any two images is calculated which in most cases contain many patches. The Minkowski-type metric has been used to measure the distance between patches in most of researches. Suppose there are two patches represented by two vectors (x1,x2,...,xm), (y1,y2,...,ym), respectively. The Minkowski metric as shown in (1) is defined as:

$$D(X,Y) = (\sum_{i=1}^{P} |X_i - Y_i|^r)^{1/r} \tag{1}$$

Actually, it is the Euclidean distance (L2 distance) when $r = 2$, and It is the Manhattan distance (L1 distance) when $r = 1$ [6].

In the approach proposed in the present paper both local and global features are considered simultaneously. We try to retain the locality of the features advantages in addition to preserving the overall layout of the objects. The similarity between the local features has been used in conjunction with the topological relations between them as a global feature of the object. Once features and their descriptors have been extracted from two or more images, the next step is to establish a feature matching approach. Now, a set of query descriptors and a database of candidate descriptors have been given, the goal is to decide which features should be matched. This problem has been divided into two separate stages. The first stage is to choose a matching strategy that decides which pairs of corresponding features are valid to the next stage for further processing. The second stage is to devise a robust algorithm to apply the matching and taking into consideration the execution time of the algorithm [7].

This paper is organized as follow: Some related work are reviewed in section 2. Our proposed matching approach is illustrated in section 3. In section 4, Some experiments are conducted to evaluate the performance of the approach. Finally, the conclusion of this work and the recommendations for future work are presented in section 5 and 6, respectively.

II. Related work

The definition of a match depends on the matching strategy. All matching strategies compare each feature (descriptor) of the first image with each feature of the second image. Threshold based matching, nearest neighbour based matching (NN), and nearest neighbour distance ratio (NNDR) have been considered as the most popular matching strategies. In the case of **threshold based matching** two regions are matched if the distance between their descriptors is below a threshold. A descriptor can have several matches and several of them may be correct. In the case of **nearest neighbour based matching (NN)** two regions A and B are matched if the descriptor D_B is the nearest neighbour to D_A and if the distance between them is below a threshold. In this approach a descriptor has only one match. Finally, the **nearest neighbour distance ratio (NNDR)** is similar to the nearest neighbour matching except that the thresholding is applied to the distance ratio between the first and the second nearest neighbours. Thus, two regions, A and B, are matched if $(||D_A - D_B||/||D_A - D_C||) < threshold$. Where D_B is the first and D_C is the second nearest neighbours to D_A [8].

Recently, a number of researchers have used local features descriptors which have been extracted from robust and invariant interest points to measure the similarity between images or to find an object within an image.

K. Mikolajczyk and C. Schmid [9] proposed a new technique which depends on a voting-based indexing. The image is represented by a set of scale invariant points; which allows the computation of a scale invariant descriptor. Each scale-invariant interest point in a query image votes for images in the database which contain an interest point within a thresholded distance from itself.

T. Tuytelaars and L. V. Gool [10] have used the same concept like K. Mikolajczyk and C. Schmid. An affine moment invariants has been used to independently cast votes for similar database images.

F. Schaffalitzky and A. Zisserman [11] have used the voting technique to select candidate matches in matching scene problem. In addition, a number of steps to verify geometric consistency within larger neighbourhoods have been applied.

D. G. Lowe [4] proposed a new matching approach using distinctive invariant features for object recognition. objects key points are matched independently via a fast nearest-neighbour algorithm to all of the key points extracted from the database images. Consequently, a Hough transform to identify clusters belonging to a single object has been applied. Finally, verification through least-squares solution for consistent pose parameters has been used.

S. Lazebnik et al. [12] have represented textures by histograms of prototypical affine-invariant features. Then, exhaustive nearest-neighbour classification with EMD has been used.

The authors of [13] and [14] have used the text retrieval metaphor in the image matching problem. Vector quantization (VQ) has been applied to affine-invariant regions, which have been collected from images. Also, each image has been represented by a fixed-length vector, which is called bag-of-words. In [13], images in the database have been ranked in similarity to a user-segmented query region based on their frequency vectors normalized scalar product. While in [14], multi-class classifiers are trained using the frequency histograms as feature vectors.

S. Lazebnik et al. [15] proposed a framework for texture recognition based on local affine-invariant descriptors and their spatial layout. Expectation maximization (EM) algorithm has been used in order to cluster the invariant descriptors and assign class labels to descriptors in novel texture images, which has been refined with a relaxation step that uses neighbourhood co-occurrence statistics from the training set.

L. Torresani et al. [16] proposed an approach to find the correspondences between features extracted from a pair of images. The matching using the appearance and the spatial arrangement of the features has been formulated as an energy minimization problem. The geometric agreement between neighbouring correspondences edges in terms of both length and direction has been used. Although these combination achieve adequate results but they have a limitation in some computer vision challenges such as rotation and scaling.

III. Proposed Matching Approach

Basically, the choice of a metric is substantial for the matching of local features; therefore conventional matching approaches reduce the matching problem to a metric problem. These approaches depend mainly on finding the minimum distance between features (descriptors) in feature space as shown in (2), where D_{ij} is the similarity measure between feature i from the first image and feature j from the second image. X_{ij} is a matching between feature i and feature j, i.e. X_{ij} = 1 if feature i in the 1st image is mapped to feature j in the 2nd image and X_{ij} = 0 otherwise. Note that $X_{ij} \in \{0, 1\}$.

$$Min\ \ F = \sum_{\forall_{i,j}} D_{ij}\ X_{ij} \tag{2}$$

Limitations: The minimum distance between features deals with each feature individually rather than a group of features (object), i.e. it performs something similar in the meaning to local optimization. Consequently, the minimum distance between features can be misleading in some cases and as a result the performance of the algorithm deteriorates. In other words, the minimum distance criterion has no objection for a feature to be wrongly matched as long as it successfully achieves the minimum distance objective.

Consequently, there are limitations on using the metric solely to decide the features correspondence. A combination of the similarity between features and the topological relations between them is proposed to cope with these limitations and to improve the accuracy of the matching method. A new term, describing the neighbourhood/ topological relations between every pair of features has been added as shown below in (3).

$$Min\ \ F = \sum_{\forall_{i,j}} D_{ij}\ X_{ij} + \alpha \sum_{\forall_{i,j,k,l}} X_{ij}\ X_{kl}\ P_{ij,kl} \tag{3}$$

Subject to:

$$\sum_{j=1}^{n} X_{ij} \quad \leq\ 1 \qquad\qquad (a)$$

$$\sum_{i=1}^{m} X_{ij} \quad \leq\ 1 \qquad\qquad (b)$$

$$\sum_{i=1}^{m}\sum_{j=1}^{n} X_{ij} \quad = \ Min(m, n) \qquad\qquad (c)$$

The second term in (3) represents a penalty term over all pairs of features. $P_{ij,kl}$ is called a penalty matrix. It is used to penalize matching pairs of features X_{ij} in one image with corresponding pairs X_{kl} in the other image if they have different topologies. It is binary and of $(m \times n, m \times n)$ dimension; where m, n are the number of features in the first and the second images respectively. $P_{ij,kl} = 1$ if the features k, l in the second image have different topology when compared to features i, j in the first image. In other words, if any two features are neighbours to each other in the first image and matched to two features in the second image which are not neighbours to each other or vice versa. Hence a penalty term will be added to this matched pair.

(α) is called a topology coefficient. It indicates how much the matching algorithm depends on the topology between images and it will be adjusted according to the image type. If (α) is equal to zero, it means that the traditional metric matching technique is applied without any topology constraints. If (α) is large, it means that the matching algorithm greatly depends on the topology constraints. In the experiments, (α) was chosen in a range from 0 to 0.1.

The topology term has nearly no impact when the difference of similarities between the features is high. On the contrary, the topology term is effective and has a great impact when the features are similar to each other.

Constraints: Constraint (a): There exists at most one in every column of x. Constraint (b): There exists at most one in every row of x. Constraint (c): The summation of all rows and columns (all elements in x) must be equal to the minimum of m, n. The first two constraints ensure that every feature should match at most one feature. The constraint (c) enforces all features in the image which have less number of features to be matched.

Let's illustrate the work by a simple example. Suppose there are two images. Each one has three features. The target is to decide which of these features in an image should be matched to which feature in the other image. Fig. 1, depicts the features in each image. If the distance between any two features in the same image is less than a threshold then there is an edge linking these two features. Any two features having an edge between them are called neighbours to each other. According to the example features 2 and 3 in the first image are neighbours as well as features 1 and 2 in the second image. Adjacency matrix of each image has been created from the idea of the neighbourhood's features as illustrated in table I. The penalty matrix has been constructed from the adjacency matrix of the two images as depicted in table II. There are only three different values which are 0, 1 and not valid in the penalty matrix. Actually, it can be said that there are only two values because zero and not valid are the same in the implementation. Now, we illustrate how to calculate three different random values from the table I. The cell F11 and F32 means that feature 1 in the first image is matched to feature 1 in the second image. And feature 3 in the first image is matched to feature 2 in the second image. As shown in fig.1 and table I, features 1 and 3 in the first

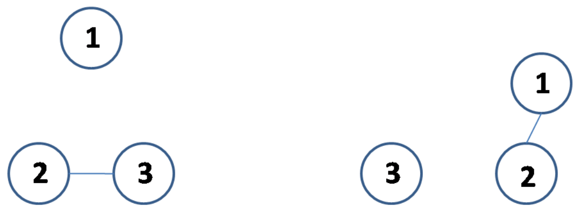

Figure 1. Undirected graph for two images

Table I
ADJACENCY MATRIX FOR TWO IMAGES. IMAGE1 (LEFT) AND IMAGE2 (RIGHT)

	F1	F2	F3		F1	F2	F3
F1	0	0	0	F1	0	1	0
F2	0	0	1	F2	1	0	0
F3	0	1	0	F3	0	0	0

image are not neighbours and feature 1 and 2 in the second image are neighbours, so in this case we add 1 in this cell which means there is a penalty that should be paid. The cell F31 and F22 means that feature 3 and 2 in the first image has been matched to feature 1 and 2 in the second image respectively. It is clearly shown in fig. 1 that features 3 and 2 in the first image are neighbours and feature 1 and 2 in the second image are also neighbours, so in this case we add 0 in this cell which means that there is no penalty that should be paid. The cell F11 and F12 means that feature 1 in the first image has been matched to feature 1 and 2 in the second image, which is not valid for a feature to match two features or to be matched by two features.

To cope with the previous limitation (enforcing all features to be matched), we should get rid of the hard constraint (c). The constraint number (c) has been canceled, and a new term (soft constraint) has been added to the objective function $\beta\ (Min(m,n)\ -\sum_{\forall_{i,j}} X_{ij}$ as shown in (4).

$$\begin{aligned} Min\ \ F = \sum_{\forall_{i,j}} D_{ij}\ X_{ij} + \alpha \sum_{\forall_{i,j,k,l}} X_{ij}\ X_{kl}\ P_{ij,kl} \\ + \beta\ (Min(m,n)\ - \sum_{\forall_{i,j}} X_{ij}) \end{aligned} \tag{4}$$

Subject to:

$$\sum_{j=1}^{n} X_{ij} \quad \leq\ 1 \qquad (a)$$

$$\sum_{i=1}^{m} X_{ij} \quad \leq\ 1 \qquad (b)$$

Where (β) is called a threshold coefficient. It indicates how much the matching algorithm depends on the features matching threshold. It will be adjusted according to the image type. In the experiments, (β) was chosen in a range from 0 to 0.5.

This new term doesn't enforce the matching algorithm to match the whole features but it only adds a penalty which is proportional to the difference between the total number of features and the matched features. i.e. if all the features have been matched. Therefore, there is no extra penalty and this term will be zero and since the difference between the matched features and the total features number has been increased, the penalty term related to the threshold will increase too.

Table II
PENALTY MATRIX EXAMPLE

	F11	F21	F31	F12	F22	F32	F13	F23	F33
F11	X	X	X	X	1	1	X	0	0
F21	X	X	X	1	X	0	0	X	1
F31	X	X	X	1	0	X	0	1	X
F12	X	1	1	X	X	X	X	0	0
F22	1	X	0	X	X	X	0	X	1
F32	1	0	X	X	X	X	0	1	X
F13	X	0	0	X	0	0	X	X	X
F23	0	X	1	0	X	1	X	X	X
F33	0	1	X	0	1	X	X	X	X

The objective function now which should be minimized depends not only on the metric and topology as before but also on a threshold.

For example, if we have 10 features. The threshold with the metric and topology terms can choose to match only 8 from the 10 features. Although, there will be a penalty added concerning the threshold term since it matches only 8 features out of 10 equal to (10-8)* . Therefore, the cost of matching these 8 features (min. distance + topology + threshold) will be less than matching all the features (min. distance + topology).

This problem is quadratic-objective subject to linear constraints. It is called Mixed Integer Quadratic Programming (MIQP) problem [17]. Mixed Integer Quadratic Programming (MIQP) problem should be in the form:

$$X' \times H \times X + F \times X$$

Consequentially, our problem should be rewritten into this form. To do this equation (4) should be rewritten as in (5):

$$\begin{aligned} Min\ \ F = \sum_{\forall_{i,j}} D_{ij}\ X_{ij} + \alpha \sum_{\forall_{i,j,k,l}} X_{ij}\ X_{kl}\ P_{ij,kl} \\ + \beta\ Min(m,n)\ - \beta \sum_{\forall_{i,j}} X_{ij} \end{aligned} \tag{5}$$

The term β Min(m,n) is a constant term, so it has no effect while solving. Equation (5) has been reduced to (6):

$$\begin{aligned} Min\ \ F = \sum_{\forall_{i,j}} D_{ij}\ X_{ij} + \alpha \sum_{\forall_{i,j,k,l}} X_{ij}\ X_{kl}\ P_{ij,kl} \\ - \beta \sum_{\forall_{i,j}} X_{ij} \end{aligned} \tag{6}$$

Then, the non quadratic term has been combined as shown in (7):

$$Min\ \ F = \sum_{\forall_{i,j}} X_{ij}(D_{ij} - \beta) + \alpha \sum_{\forall_{i,j,k,l}} X_{ij}\ X_{kl}\ P_{ij,kl} \quad (7)$$

Algorithm (1) gives a summary of the proposed local features matching algorithm, which depends not only on the similarity between features but also on the topological relations between them.

Algorithm 1 Local Feature Matching

Input: A pair of Images, topology coefficient (α), and threshold coefficient (β).

1) For every image:
   a) Detect local features (select strongest 100);
   b) Extract a descriptor for every feature;
2) For every feature (descriptor) in the 1st image: Calculate the similarity between it and all the features in the 2nd image;
3) Penalize any pair of features that matches to a pair of different topology;
4) Compute the objective function using (7) (features similarity and topological constraints);

Output: List of features correspondences.

IV. Experiments

The proposed interest point matching method has been tested on a number of images from a dataset with extra synthetic deformation.

Dataset: Columbia Object Image Library (COIL-100) [18] has been used in the experiments. COIL-100 is a database of color images which has 7200 images of 100 different objects (72 images per object). Each object was placed at the center of a motorized turntable with a black background. This turntable was rotated through 360 degrees. The objects were acquired with a fixed color camera at every 5 degrees of rotation. Consequently, these collections of objects have a wide diversity of complex geometric and reflectance characteristics.

Features Detection and extraction: SURF (Speeded Up Robust Features) [19][20] has been used to detect and extract the interest points in every image. The SURF has been used because it is faster than other descriptors and at the same time it has adequate accuracy. The detection process is based on an integer approximation to the determinant of Hessian blob detector. The speed gain is due to the use of an intermediate image representation known as "integral images", which makes a significant reduction in the number of operations for simple box convolutions, independent of the chosen scale. The descriptor is based on sums of Haar wavelet components.

Table III
The experimental results summary

	Correct Matches	Possible Matches	Detection Rate
Threshold	420	1185	0.35
NN	395	1185	0.33
NNDR	375	1185	0.32
Proposed	725	1185	0.62

Evaluation criterion: For each pair of images, every interest point in image 1 is compared to all interest points in image 2 by comparing their descriptors which have been extracted by SURF. The detection rate of the best N matches has been calculated in order to measure the performance. The detection rate R is defined as the ratio between the number of correct matches and the number of all possible matches [8].

$$\text{R} = \frac{\text{Number of Correct Matches}}{\text{Number of possible Matches}}$$

Experiment results: A Receiver Operating Characteristic (ROC) based criterion has been used to show the detection rates versus the number of most similar matches allowed (N). The ROC curves are shown in table III and fig.2. The experiments have been done using three state-of-the-art strategies which are Threshold, Nearest Neighbour (NN) and Nearest Neighbour Distance Ratio (NNDR) in addition to the proposed system. Ten objects of the aforementioned dataset have been chosen to perform the experiments as depicted in fig.3. These objects with extra synthetic deformations such as rotation, scaling, partial occlusion and heavy noise have been used for this purpose. In addition, a duplication of the same object has been found in the same image with deformations, but one as a whole and one as parts to make the matching more challenging and to test the principle goal of the new matching strategy as shown in fig.4.

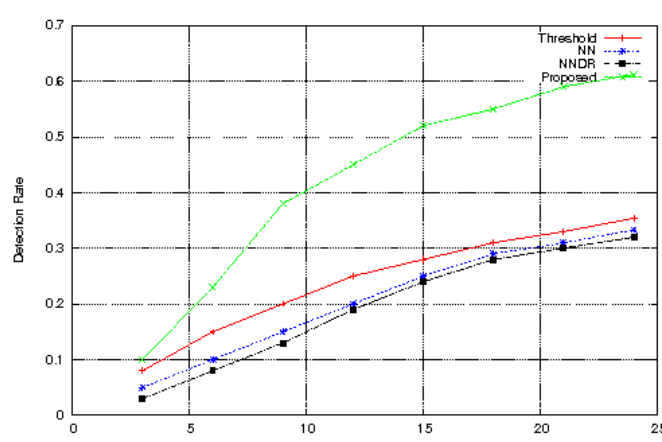

Figure 2. ROC curve for features matching experiments

Figure 3. Examples of objects from the COIL-100 dataset used for the evaluation

Figure 4. Some image pairs with synthetic deformations and duplication

Table IV depicts some experimental results. Each row in the table represents an instance. The first three columns show the results using three state-of-the-art strategies which are the threshold, the nearest neighbour (NN), and nearest neighbour distance ratio (NNDR) respectively. The last column represents the proposed strategy. To illustrate the advantage of the proposed strategy, let's take a closer look to the second experiment. In this experiment, the candidate object is subjected to rotation and an exact but partitioned copy of the object is added to the image making the matching process more challenging. The total number of possible matches is 28. The threshold, NN, NNDR successfully match 20, 12, and 12 features respectively. On the other hand, the proposed approach successfully match 24. In addition the proposed approach eliminated the false matches. From these experiments, the proposed strategy demonstrates very promising performance in accuracy when compared to other state-of-the-art strategies.

V. Conclusions

In this paper, a novel approach for features matching that can be used to serve object recognition is introduced. In the introduced approach, both local and global features are considered simultaneously and a set of control parameters is employed to tune the performance by adjusting the significance of global vs. local features. A major contribution of this research is considering the topological relations between the local features as a global feature of the object. From experimental results, it was found that the number of correctly matched features is increased. Moreover, wrong matches between visually similar features are eliminated. Experimental results show the superior performance of this approach, with accuracy 62% in the detection rate.

VI. Future Work

After the proof of concept of the aforementioned approach has been shown, a lot of work should be done to generalize this local features matching approach and achieve high degree of robustness and computational efficiency. First, a preprocessing step is needed to automatically decide the parameters values ($alpha$, $beta$) should be done. These values may depend on images size, number of extracted features in each image and images resolution. Second, an optimization to the algorithm to be more computationally efficient should be made without any loss in the algorithm accuracy. This may be done by using metaheuristic methods [21] alone or in conjunction with other exact methods. Finally, applying the proposed matching algorithm to different features extraction techniques and decide the pros. and cons. of using every one.

Acknowledgment

This research has been supported by the Ministry of Higher Education (MoHE) of Egypt through a Ph.D. fellowship. Our sincere thanks to Egypt-Japan University for Science and Technology (E-JUST) for guidance and support.

References

[1] B. Zitova and J. Flusser, "Image registration methods: a survey," *Image and vision computing*, vol. 21, no. 11, pp. 977–1000, 2003.

[2] C.-R. Shyu, C. Brodley, A. Kak, A. Kosaka, A. Aisen, and L. Broderick, "Local versus global features for content-based image retrieval," in *Content-Based Access of Image and Video Libraries, 1998. Proceedings. IEEE Workshop on.* IEEE, 1998, pp. 30–34.

[3] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *International Journal of Computer Vision*, vol. 74, no. 1, pp. 59–73, 2007.

[4] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[5] N. Snavely, S. M. Seitz, and R. Szeliski, "Modeling the world from internet photo collections," *International Journal of Computer Vision*, vol. 80, no. 2, pp. 189–210, 2008.

[6] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma, "A survey of content-based image retrieval with high-level semantics," *Pattern Recognition*, vol. 40, no. 1, pp. 262–282, 2007.

[7] R. Szeliski, *Computer vision: algorithms and applications*. Springer, 2011.

[8] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 10, pp. 1615–1630, 2005.

[9] ——, "Indexing based on scale invariant interest points," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 1. IEEE, 2001, pp. 525–531.

[10] T. Tuytelaars and L. Van Gool, "Content-based image retrieval based on local affinely invariant regions," in *Visual Information and Information Systems*. Springer, 1999, pp. 493–500.

[11] F. Schaffalitzky and A. Zisserman, "Automated scene matching in movies," in *Image and Video Retrieval*. Springer, 2002, pp. 186–197.

[12] S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using affine-invariant regions," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2. IEEE, 2003, pp. II–319.

[13] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 1470–1477.

[14] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Workshop on statistical learning in computer vision, ECCV*, vol. 1, 2004, p. 22.

[15] S. Lazebnik, C. Schmid, and J. Ponce, "Affine-invariant local descriptors and neighborhood statistics for texture recognition," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 649–655.

[16] L. Torresani, V. Kolmogorov, and C. Rother, "Feature correspondence via graph matching: Models and global optimization," in *Computer Vision–ECCV 2008*. Springer, 2008, pp. 596–609.

[17] O. Nissfolka and R. P. T. W. F. Janssonb, "A mixed integer quadratic reformulation of the quadratic assignment problem with rank-1 matrix," in *11th International Symposium on Process Systems Engineering: Pse2012*, vol. 15. Elsevier, 2012, p. 360.

[18] S. Nayar, S. Nene, and H. Murase, "Columbia object image library (coil 100)," *Department of Comp. Science, Columbia University, Tech. Rep. CUCS-006-96*, 1996.

[19] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Computer Vision–ECCV 2006*. Springer, 2006, pp. 404–417.

[20] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[21] I. Boussaïd, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Information Sciences*, 2013.

Table IV
SOME MATCHING EXAMPLES