Correspondence Propagation with Weak Priors

Huan Wang[1], Shuicheng Yan[2], Jianzhuang Liu[1], Xiaoou Tang[1] and Thomas S. Huang[2]

[1]Department of Information Engineering, Chinese University of Hong Kong, Hong Kong
{hwang5,jzliu,xtang}@ie.cuhk.edu

[2]Beckman Institute, University of Illinois at Urbana-Champaign, USA
{scyan,huang}@ifp.uiuc.edu

Abstract

For the problem of image registration, the top few reliable correspondences are often relatively easy to obtain, while the overall matching accuracy may fall drastically as the desired correspondence number increases. In this paper, we present an efficient feature matching algorithm to employ sparse reliable correspondence priors for piloting the feature matching process. First, the feature geometric relationship within individual image is encoded as a spatial graph, and the pairwise feature similarity is expressed as a bipartite similarity graph between two feature sets; then the geometric neighborhood of the pairwise assignment is represented by a categorical product graph, along which the reliable correspondences are propagated; and finally a closed-form solution for feature matching is deduced by ensuring the feature geometric coherency as well as pairwise feature agreements. Furthermore, our algorithm is naturally applicable for incorporating manual correspondence priors for semi-supervised feature matching. Extensive experiments on both toy examples and real-world applications demonstrate the superiority of our algorithm over the state-of-the-art feature matching techniques.

Index Terms

Image Registration, Object Correspondence, Feature Matching, Weak Prior, Propagation.

I. Introduction

Feature matching of two objects is a fundamental problem for computer vision research, and a variety of computer vision tasks heavily rely on the feature matching results, such as object tracking [11] and recognition [15] [14], image warping [3] and stitching [7], and 3D reconstruction [2] [16] [1]. The feature matching accuracy may be affected by various factors including feature descriptors, similarity measurements, and matching approaches.

Substantive works have been devoted to seeking the correspondences between features extracted from two images. Among them, recently Grauman et al. [10] considers the image features as unordered elements in sets of different cardinalities and proposes a pyramid matching algorithm for pursuing inexact correspondences. Local feature plays an important role in this task, and the popular feature detectors, such as SIFT [14], salient region detector [12], as well as scale and affine invariant interest point detector [17], tend to output interest points or regions in a structured way. Also, it is observed that the salient points and SIFT features extracted from the images with similar structures often share similar local spatial distributions. Thus the feature location also conveys important information for feature matching. The works in [6] [20] and [18] present approaches for utilizing structure information. They formulate the feature matching problem with integer quadratic programming (IQP) or Semidefinite Programming (SDP) techniques, and hence severely suffer from the high computational cost. Leordeanu et al. [13] proposes a spectral analysis method for promoting feature matching accuracy with the geometric structure information and designs an iterative procedure to eliminate the conflicts among the derived correspondences. [9] adds affine constraints to the spectral matching formulation and proposes a normalization procedure to improve the matching accuracy.

One common issue encountered by all above feature matching algorithms is that the top few matches with the highest similarities are often very accurate, but the matching accuracy falls rapidly when the desired match number increases, especially for data with noises. Another issue arising in real-world applications is that the unsupervised feature matching algorithms often cannot provide sufficiently accurate results for the subsequent applications such as image stitching and object recognition. A natural question is how to incorporate extra clues for promoting feature matching performance. In this work, we present a solution for feature matching with the *reliable correspondence priors*, from the top few reliable correspondences obtained by either conventional feature matching algorithms or manual labeling.

First, the relative geometric relation of the feature pairs within an image is encoded as a spatial graph, and the matching assignments are considered as the vertices of the product graph constructed from two spatial graphs of the images to be matched. Then, based on the these spatial relations, the assignment neighborhoods are defined on the product graph and the point-to-point matchings are then propagated from those reliable correspondences to the
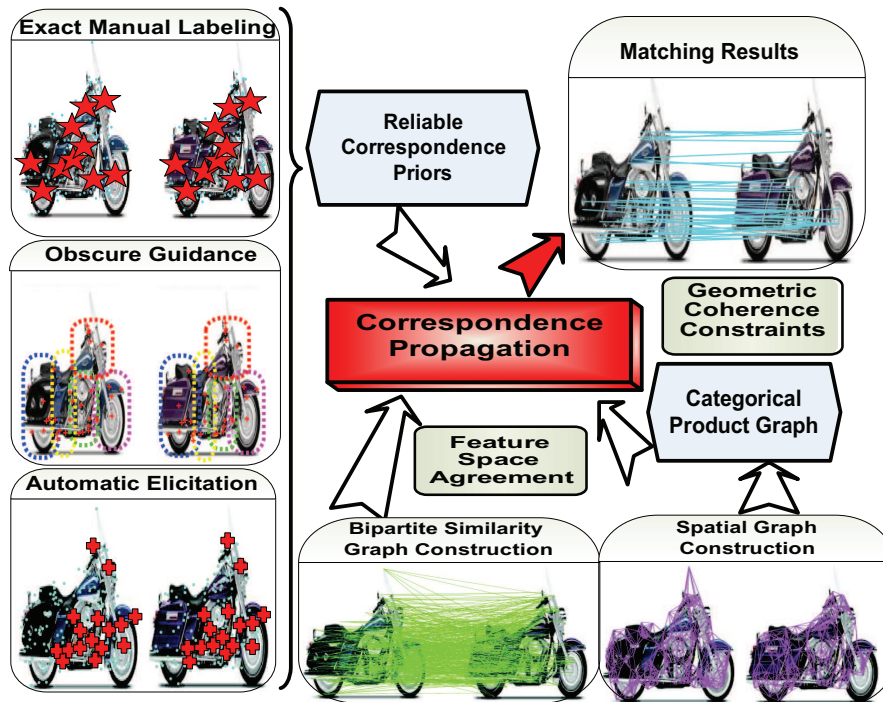
Fig. 1. Flowchart of correspondence propagation from reliable correspondence priors for feature matching.

remaining ones. Finally, we deduce an efficient closed-form solution for the feature matching problem by ensuring both spatial consistency and feature similarity agreements.

The works in [13], [6] and [18] also try to employ the feature location information for matching, while our work in this paper differs from them in that we make full use of the information provided by those reliable correspondences. Moreover, benefitting from the propagation property, our framework is easy to incorporate human interactions for the guidance of correspondence searching. An illustration of the whole framework for correspondence propagation from reliable correspondence priors is displayed in Figure 1.

Here, we would like to highlight some aspects of our proposed Reliable Correspondence Propagation (RCP) algorithm:

1) RCP makes full use of the prior information of the sparse reliable correspondences, and is naturally applicable for incorporating the interactive manual labeling to further promote feature matching accuracy in a semi-supervised way.

2) The algorithmic objective provides a unified formulation that employs both the categorical product graph constructed from two spatial graphs for characterizing spatial coherency and the bipartite similarity graph for representing feature similarity agreements.

3) A closed-form solution is deduced with comparably low computational cost, and hence our algorithm is applicable to real-world image registration problems.

II. PROBLEM FORMULATION AND SOLUTION

A. Notations and Graph Construction

The two sets of features, e.g. extracted from SIFT [15], within two images to be matched are denoted as $\Phi^1 = \{\phi_1^1, \phi_2^1, ..., \phi_{N^1}^1\}$ and $\Phi^2 = \{\phi_1^2, \phi_2^2, ..., \phi_{N^2}^2\}$ with $\phi_i^k = \{f_i^k, x_i^k\}$, where f_i^k is the feature vector and x_i^k is the feature point location in the kth image ($k \in \{1, 2\}$).

Let $G^k = (V^k, E^k)$ be an undirected spatial graph with vertex set V^k and edge set E^k for the kth image. The edges in E^k reflect the geometric neighboring relations among the features, and can be defined in terms of k-nearest-neighbor or an ϵ-ball distance criteria in the feature position space. In addition, an adjacency/weight matrix W^k is defined for the graph G^k. One way to compute the weight matrix is directly based on the edge information, namely

$$w_{ij}^k = \begin{cases} 1 & \text{if } x_i^k \text{ and } x_j^k \text{ are connected,} \\ 0 & \text{else.} \end{cases}$$

There are also other ways for computing the similarity matrix, such as the heat kernel [4], i.e., $w_{ij}^k = e^{-\frac{\|x_i^k - x_j^k\|^2}{t}}$, where $t \in \mathbb{R}$ is a parameter to define the heat kernel.

To encode the pairwise feature similarity between two sets of features, we introduce the similarity graph, denoted as a triplet $G^{12} = (\Phi^1, \Phi^2, E^{12})$. The similarity graph G^{12} is a bipartite graph, and the weight matrix S of G^{12} are computed from the cosine distances of the feature pairs measured in the feature vector space.

B. Regularization on Categorical Product Graph

The feature matching process can be considered as seeking a binary function over the product set of Φ^1 and Φ^2:

$$\mathcal{M} : \Phi^1 \times \Phi^2 \rightarrow \{0, 1\},$$

where $\times$ denotes the set product and the function value 1 means matching and 0 for mismatching. To transduce the matching assignment from the reliable correspondence priors to the other feature pairs, we first give a neighborhood definition for the matching assignments.

Definition: Suppose $\Phi^1 = \{\phi_{i_1}^1, \phi_{i_2}^1, ..., \phi_{i_{N_1}}^1\}$ and $\Phi^2 = \{\phi_{i_1}^2, \phi_{i_2}^2, ..., \phi_{i_{N_2}}^2\}$ are the vertices of graph G^1 and G^2 respectively. Two assignments $m_{i_1 i_2} = \{\phi_{i_1}^1, \phi_{i_2}^2\}$ and $m_{j_1 j_2} = \{\phi_{j_1}^1, \phi_{j_2}^2\}$ are neighbors iff both pairs $\{\phi_{i_1}^1, \phi_{j_1}^1\}$ and $\{\phi_{i_2}^2, \phi_{j_2}^2\}$ are neighbors in G^1 and G^2 respectively, namely,

$$m_{i_1 i_2} \sim m_{j_1 j_2} \quad \text{iff} \quad \phi_{i_1}^1 \sim \phi_{j_1}^1 \quad \text{and} \quad \phi_{i_2}^2 \sim \phi_{j_2}^2, \tag{1}$$

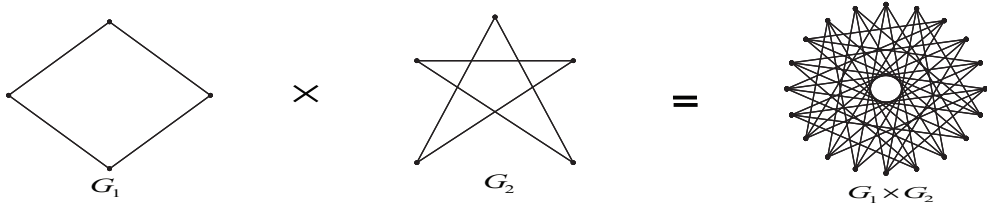where $a \sim b$ means a and b are neighbors on the corresponding graph.

Fig. 2. Demonstration of categorical graph product: graph G_1, graph G_2, and their categorical graph product $G_1 \times G_2$

Suppose binary weights are utilized. According to the definition (1), the assignment graph G^a is the categorical product graph of G^1 and G^2, i.e., $G^a = G^1 \times G^2$, and the adjacency of the assignments can be expressed as:

$$w^a_{m_{i_1 i_2} m_{j_1 j_2}} = w^1_{\phi^1_{i_1} \phi^1_{j_1}} w^2_{\phi^2_{i_2} \phi^2_{j_2}}. \tag{2}$$

An example of the categorical product graph is demonstrated in Figure 2.

Defined on the space of Cartesian product set, the assignment $\mathcal{M}$ can be regarded as a binary matrix of N_1 by N_2, i.e.,

$$M = \begin{pmatrix} m_{11} & m_{12} & ... & m_{1N_2} \\ m_{21} & m_{22} & ... & m_{2N_2} \\ ... & ... & ... & ... \\ m_{N_1 1} & m_{N_1 2} & ... & m_{N_1 N_2} \end{pmatrix}, m_{ij} \in \{0, 1\} \tag{3}$$

where the elements m_{ij} corresponds to the assignment of ϕ^1_i to ϕ^2_j. To facilitate the solution, we arrange the columns of M consequently to construct a vector $\vec{M}$, i.e.,

$$\vec{M} = vec(M) = [m_{11}, m_{21}, ..., m_{N_1 1}, m_{12}, m_{22},$$

$$..., m_{N_1 2}, ..., m_{1N_2}, m_{2N_2}, ..., m_{N_1 N_2}]^T, \tag{4}$$

where $vec(\cdot)$ is the vectorization operator.

Now the assignment is converted into a function on the $N_1 N_2$ dimensional vector space and thus the adjacency matrix W^a of the assignments is an $N_1 N_2$ by $N_1 N_2$ matrix, i.e.,

$$W^a = W^{2T} \otimes W^1, \tag{5}$$

where $\otimes$ is the Kronecker product operator and the corresponding graph G^a is the categorical product graph of G^1 and G^2. Note that the adjacency matrix of the categorical product graph can also be defined as $W^a = W^1 \otimes W^2$ if we rearrange the sequence of assignments while here we adopt the first definition so that the assignment arrangement is coherent with that of $\vec{M}$. When the W^1 and W^2 are not binary, the adjacency matrix W^a calculated from (5) can still capture the relative geometric relations of the assignments.

To introduce a spatial consistency for the assignments, we make the assumption that the neighboring vertices on the categorical product graph share similar assignment values. This is quite natural in the representation of

structural feature sets, since in real-world applications, the feature points that constitute certain kind of structures are often extracted together and thus the features are often matched *set* by *set*. Emphasizing this assumption can also transduce the *reliable correspondence priors* derived from manual labeling or automatic approaches to their neighboring assignments and then the assignments are propagated along the categorical product graph until a final balance is drawn.

In our framework, the spatial consistency assumption is fulfilled by a graph Laplacian penalty item in the objective. According to the spectral graph theory [5] [19], penalizing the Graph Laplacian leads to a solution with the locality preserving property. The Graph Laplacian can be expressed as:

$$\vec{M}^T L^a \vec{M}^T = \frac{1}{2} \sum_{ij} w_{ij}^a (m_i^v - m_j^v)^2,$$

where m_i^v is the ith element of $\vec{M}$, $L^a = D^a - W^a$ is the Laplacian matrix of the categorical product graph and D^a is a diagonal matrix with $D_{ii}^a = \sum_j W_{ij}^a$. If m_i^v and m_j^v are adjacent in the graph, i.e., the w_{ij}^a is large, the minimizing of the objective will lead to a small distance between m_i^v and m_j^v, and then the reliable prior correspondence can be propagated along with this spatial consistency property.

C. Consistency in Feature Domain and Soft Constraints

Besides the geometric consistency, we also emphasize the coherence in the feature domain. The pairwise feature agreement is encoded by the N_1 by N_2 adjacency matrix S of the similarity graph G^{12}. The coherence of the feature similarity is then converted into the maximization of item,

$$|S \odot M|_s = vec(S) \cdot vec(M) = \vec{S}^T \vec{M},$$

$$\text{w.r.t.} \quad m_i^v \in \{0, 1\}, \tag{6}$$

where $\odot$ is the matrix Hardamard product, $|A|_s$ returns the sum of all the elements in matrix A, $\vec{S}$ is the vectorization of the matrix S, and the operator $\cdot$ is the inner product of two vectors.

Finally, for those one-to-one correspondence configurations, a soft penalty is introduced, i.e.,

$$\sum_{i=1}^{N_1} (|A_1^i \odot M|_s - 1)^2 + \sum_{i=1}^{N_2} (|A_2^i \odot M|_s - 1)^2, \tag{7}$$

where A_1^i is an N_1 by N_2 coefficient matrix with 1 in the ith row and 0 for others; A_2^i is an N_1 by N_2 coefficient matrix with 1 in the ith column and 0 for other elements. The first term tends to matching each feature in the first image to a feature with the largest similarity in the second one, and the second term tends to matching each feature in the second image with a feature with the highest similarity in the first one.

Vectorizing the coefficient matrices A_1^i and A_2^i and arranging the derived column vectors, we construct the constraint coefficient matrices $\hat{A}_1$ and $\hat{A}_2$:

$$\hat{A}_1(:,i) = vec(A_1^i), \quad \hat{A}_2(:,i) = vec(A_2^i).$$

Then the item (7) can be expressed as:

$$Tr((\hat{A}_1^T \vec{M} - e_{N_1})^T(\hat{A}_1^T \vec{M} - e_{N_1}))$$
$$+ Tr((\hat{A}_2^T \vec{M} - e_{N_2})^T(\hat{A}_2^T \vec{M} - e_{N_2})), \tag{8}$$

where $\hat{A}_1 = e_{N_2} \otimes I_{N_1}$ is an $N_1 N_2$ by N_1 matrix, $\hat{A}_2 = I_{N_2} \otimes e_{N_1}$ is an $N_1 N_2$ by N_2 matrix, e_N is an N dimensional column vector of 1 and I_N is an N by N identity matrix.

Note that for the one-to-one correspondence, we can also impose hard constraints, i.e.,

$$\hat{A}_1^T \vec{M} = e_{N_1} \quad \text{or} \quad \hat{A}_2^T \vec{M} = e_{N_2}, \tag{9}$$

but these conditions may not be satisfied, since the feature extracted in one image may not have a correspondence in the other image due to the noise, occlusion or the inequality of the feature set cardinality. Thus we adopt a soft penalty in the objective and the affine constraints are consequently removed from the formulation.

D. Inhomogeneous Pair Labeling

Since the one-to-one matching is optimized on the product graph of the two input graphs, the number of variables can be extremely large and it grows rapidly with the increase of the input vertex number. The number of features extracted depends on various factors such as the feature extractors, the complexity of surroundings, the scales searched for maximum and the size of images. Generally, the assignment variables are highly redundant. Substantive assignment variables are dispensable due to the low similarity, or, large feature distances between the involved feature pairs. We call these assignments *inhomogeneous pairs*. Rather than simply removing them, in our framework the *mismatch* information of those inhomogeneous pairs is also employed. Specifically, they are assigned as 0's, which indicate that the corresponding feature pairs will not be matched, i.e.,

$$M_{i,j} = \vec{M}_{i+(j-1)\times N_1} \Leftarrow 0 \text{ if } \{\phi_i^1, \phi_j^2\} \in \Psi, \tag{10}$$

where Ψ is the set of inhomogeneous pairs. Then the *mismatch* information of those inhomogeneous pairs is also utilized to guide the solution and transduced to the remaining ones.

E. Reliable Correspondence Propagation

In the following the known correspondences including some reliable correspondences and certain number of inhomogeneous pairs are called labeled assignments or labeled feature pairs. We arrange the matching variables so that the labeled assignments are placed ahead, i.e.,

$$\vec{M}^* = [\vec{M}^l; \vec{M}^u], \tag{11}$$

where $\vec{M}^l$ represents the assignments of the labeled feature pairs, $\vec{M}^u$ corresponds to the assignment values of the remaining unlabeled feature pairs to be estimated. $\vec{M}^*$ is the rearranged assignment vector.

Correspondingly, the constraint coefficient matrices $\hat{A}_1$, $\hat{A}_2$ and the vectorized adjacency matrix $\vec{S}$ of the similarity graph are also rearranged, so that,

$$\hat{A}_1^* = [\hat{A}_1^l; \hat{A}_1^u], \hat{A}_2^* = [\hat{A}_2^l; \hat{A}_2^u], \quad \text{and} \quad \vec{S}^* = [\vec{S}^l; \vec{S}^u], \tag{12}$$

where $\hat{A}_1^l$, $\hat{A}_2^l$, and $\vec{S}^l$ are the coefficients and vectorized adjacency sub-matrix of the similarity graph for the labeled assignments respectively; $\hat{A}_1^u$, $\hat{A}_2^u$, and $\vec{S}^u$ are the coefficients and vectorized adjacency sub-matrix for the unlabeled assignments; and $\hat{A}_1^*$, $\hat{A}_2^*$, and $\vec{S}^*$ are the rearranged coefficients and vectorized similarity graph adjacency matrix.

Due to the variable rearrangement, the vertex order in the categorical product graph is also modified. The rearranged adjacency matrix W^{a*} and the corresponding Laplacian matrix L^{a*} are

$$W^{a*} = \begin{pmatrix} W_{ll}^a & W_{lu}^a \\ W_{ul}^a & W_{uu}^a \end{pmatrix}, \; L^{a*} = \begin{pmatrix} L_{ll}^a & L_{lu}^a \\ L_{ul}^a & L_{uu}^a \end{pmatrix}. \tag{13}$$

Integrating all factors and we get the final optimization formulation for our feature matching framework:

$$\min_{\vec{M}^*} -\vec{S}^{*T}\vec{M}^* + \lambda \vec{M}^{*T} L^{a*} \vec{M}^* +$$
$$\gamma(Tr((\hat{A}_1^{*T}\vec{M}^* - e_{N_1})^T(\hat{A}_1^{*T}\vec{M}^* - e_{N_1}))$$
$$+ Tr((\hat{A}_2^{*T}\vec{M}^* - e_{N_2})^T(\hat{A}_2^{*T}\vec{M}^* - e_{N_2})))$$
$$w.r.t. \quad m_i^* \in \{0, 1\}, \quad i \in \{1, 2, ..., N_1 N_2\} \tag{14}$$

where m_i^* is the ith element of $\vec{M}^*$, λ and γ are coefficients controlling the balance among feature similarity, spatial coherency and one-to-one penalty.

We relax the binary integer optimization problem to real values by discarding the constraints in (14) and the formulation is converted to an unconstrained quadratic optimization. Take the derivative w.r.t. $\vec{M}^*$ and substitute the equation (11), we obtain a closed-form relation between the labeled and unlabeled assignments:

$$\vec{M}^u = C_{uu}^{-1}(B_u - C_{ul}\vec{M}^l), \tag{15}$$

Algorithm 1 Elicit k correspondences. [Input: M]

1: Output the correspondence $m_{ij} = \{\phi_i^1, \phi_j^2\} = \arg\max_{\phi_i^1 \phi_j^2} M$.

2: Remove from M all potential assignments in conflict with m_{ij}.

3: If column or row dimension of M becomes 0 or if the output correspondence number reaches k, stop; otherwise, go back to step 1.

---

where

$$C = \begin{pmatrix} C^{ll} & C^{lu} \\ C^{ul} & C^{uu} \end{pmatrix}$$
$$= \gamma(\hat{A}_1^* \hat{A}_1^{*T} + \hat{A}_2^* \hat{A}_2^{*T}) + \lambda L^{12*} \tag{16}$$

and

$$B = \begin{pmatrix} B^l \\ B^u \end{pmatrix}$$
$$= \gamma(\hat{A}_1^* e_{N_1} + \hat{A}_2^* e_{N_2}) + \frac{1}{2}\vec{S}^* \tag{17}$$

F. Rearrangement and Discretizing

To get the original assignment $\mathcal{M}$, we first take the inverse process of the element arrangement described above and convert $\vec{M}^*$ to $\vec{M}$, then reshape the derived assignment vector into the N_1 by N_2 matrix M. Since the assignment variables have been relaxed, we tried two discretization strategies: thresholding and eliciting. Setting a threshold for discretization is natural and it can determine the correspondence number automatically. This strategy is also suitable for the cases in which the correspondences are not required to be one-to-one. On the other hand, in case a fixed number of one-to-one correspondences are needed, we design an iterative correspondence eliciting procedure, which is displayed in Algorithm 1. Finally the whole algorithmic process is listed in Algorithm 2.

III. ALGORITHMIC ANALYSIS

A. Selection of Reliable Correspondences

The accuracy of those reliable correspondences are critical for final performance. One way to obtain these reliable correspondences in the automatic matching configuration is simply to pick up a few pairs with the highest similarity scores while the correspondences derived in this way may be clustered together and their guidance for the correspondence searching is thus limited. The work [8] proposes an Adaptive Non-Maximal Suppression (ANMS) strategy to elicit a fixed number of interest features and at the same time keep the the selected interest points spatially well distributed. In this paper, we adopt the Correspondence Elicit Procedure described in Algorithm 1 and the first several correspondences produced are regarded to be reliable in the automatic matching configuration.

Algorithm 2 Reliable Correspondence Propagation

1: **Graph Construction:** Contruct the spatial graphs G^1 and G^2 from the feature locations and calculate the adjacency matrix for the categorical product graph using $W^a = W^{2T} \otimes W^1$. Construct the bipartite similarity graph G^{12} according pairwise feature similarity.

2: **Constraint Coefficient Matrix Initialization:** Initialize the constraint coefficient matrices $\hat{A}_1$ and $\hat{A}_2$ according to the cardinality of input feature sets.

3: **Assignment Labeling:** Initialize the corresponding assignments for those reliable pairs as 1 and set the assignment variables as 0 for those inhomogeneous pairs with low similarity values.

4: **Correspondence Propagation:** Rearrange the assignment variables, the adjacency matrices, the constraint coefficient matrices so that the labeled assignments are placed in front of the unlabeled variables and calculate the closed-form solution in (15).

5: **Rearrangement:** Take the inverse process of the arrangement in step 4 and get the correspondences using the strategies described in Sec-II.F.

The transductive property of our algorithm makes it easy to incorporate human interactions for the correspondence searching and a semi-supervised matching framework is naturally derived. In this work, two configurations of human interactions are used:

Exact Pairwise Correspondence Labeling: In this configuration, the users are asked to give exact correspondence labeling for the guidance of matching, and the assignments labeled by human are used as reliable correspondence priors in the feature matching process.

Obscure Correspondence Guidance: To facilitate the user labeling, we also provide an obscure matching scheme in which the user only has to describe a rough correspondence of image parts. The procedures used in the automatic matching configuration are then employed to extract reliable correspondences within the indicated corresponding areas.

B. Computational Complexity

The complexity of the inverse operation for an n by n matrix is $\mathbf{O}(n^3)$, which is greater than the spectral algorithms ($\mathbf{O}(n^2)$). However, the matrix C_{uu} in our algorithm is sparse and exploiting this sparsity, the computational cost can be greatly reduced. Also, efficient parallel algorithms exist for the gaussian elimination procedure in the computation of the sparse matrix inversion problem and thus the computation time can be further shortened. Another factor affecting the computation cost is the candidate matching variable number, which determines the dimension of the matrix C_{uu}. In our experiments, 6000 assignments with the largest similarity scores are fetched as matching
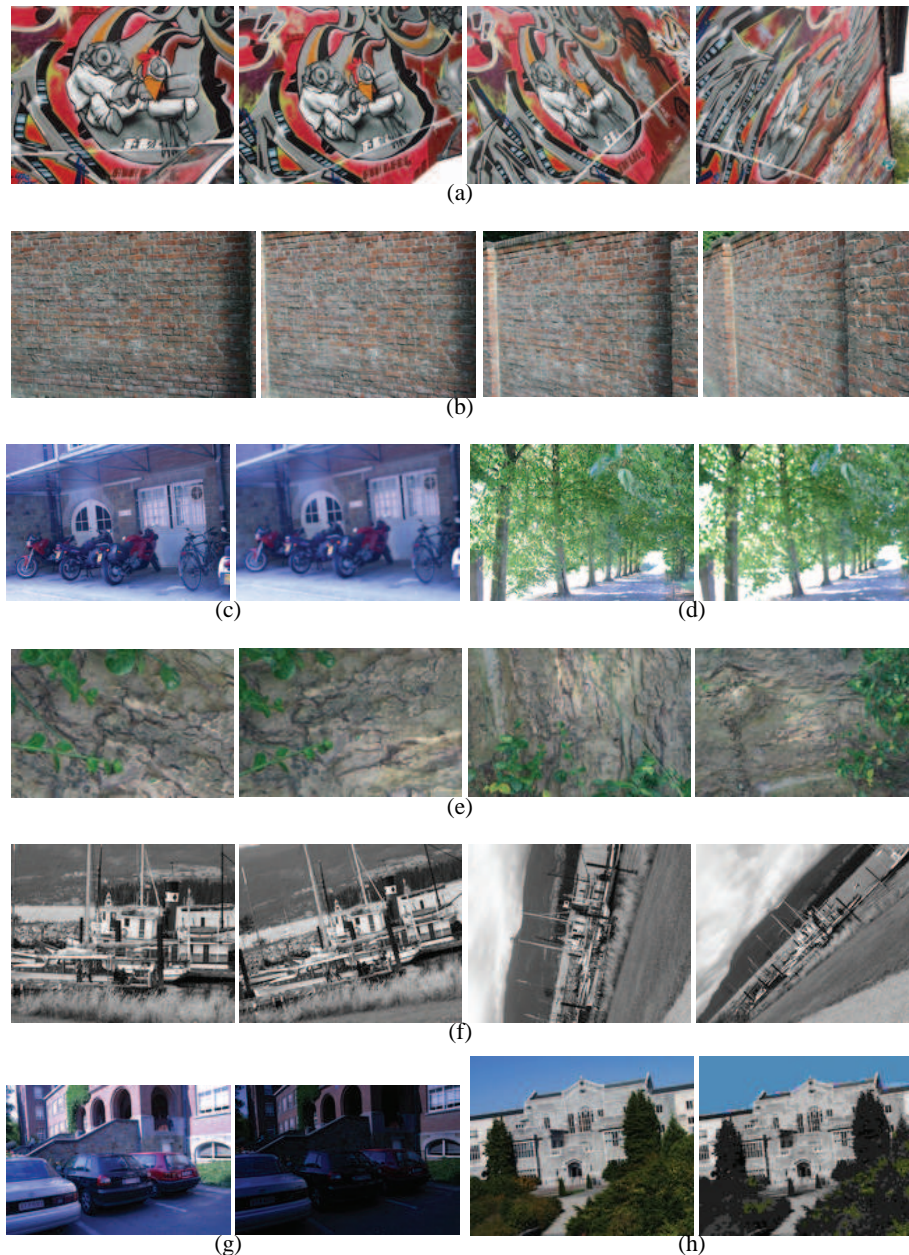
Fig. 3. Oxford real Image Transformation Database. The transformations include viewpoint variation ((a) Graffiti (b) Wall sequence), image blur ((c) bikes and (d) trees sequence), zoom and rotation ((e) bark and (f) boat sequence), illumination change ((g) leuven) and JPEG compression ((h) UBC).

candidates and the variable number can be adjusted according to the requirement of the applications. Our algorithm is much faster than the QP and SDP based algorithms and is applicable for the large scale real-world applications.

IV. APPLICATIONS AND EXPERIMENTS

In this section, our algorithm is systematically evaluated in two settings: unsupervised and semi-supervised. In the unsupervised setting, those reliable correspondences are derived automatically; while in the semi-supervised setting, the reliable correspondence priors are labeled manually. In all the experiments, the SIFT [14] descriptor is used for feature extraction and representation; the spatial graph is constructed using 10-nearest neighbors and the

weights for the spatial graphs are calculated using heat kernels $K(x, y) = \exp\{-\|x - y\|^2/\delta_o^2\}$ with parameters $\delta_o = 2^{1/2.5}\delta$ applied, where δ is the standard deviation of the feature locations. For the similarity graph, 16 nearest neighbors are used and the cosine distance is directly used as the graph weight. The coefficient λ is empirically set as 0.4 and γ is set as 0.05. In the inhomogeneous pair labeling process, we keep 6000 pairs with the top similarities as candidate matchings and others are labeled 0. The performance of our algorithm is systematically compared with the state-of-the-art feature matching algorithms, such as the spectral correspondence technique (SC) [13] and the matching algorithm used in [14] (SM), which compares the distance of the closest neighbor to that of the second-closest neighbor. We take the N_1 by N_2 pairwise similarity matrix as the input M for the Correspondence Eliciting Procedure (CE) and the matching scores are also reported. The QP and SDP based algorithms are inapplicable for comparison due to the large number of features involved. For the adjacency matrix M in the spectral correspondence technique [13], we assign a score that is linearly increasing with the cosine distance between the feature and its candidate corresponding feature to the diagonal element. Since the adjacency matrix of the categorical product graph in our algorithm represents the geometrical relations of assignments, the non-diagonal elements of M is set using the corresponding elements in W^a.

A. Automatic Feature Matching on Oxford Image Transformation Database

In this subsection, the unsupervised version of our algorithm is evaluated on the Oxford real image transformation database[1]. The Oxford database is a benchmark database for the feature descriptor evaluation. It contains eight subsets for six different geometric and photometric real image transformations, including zoom, rotation, viewpoint change, image blur, JPEG compression, and light variation. Two different scene types are involved for the case of rotation, viewpoint change, and blur: one contains homogeneous regions with distinctive edge boundaries and the other contains repeated textures of different forms, which facilitates us to analyze the effect of changing the image conditions and the scene type separately. Some images in oxford database are demonstrated in Figure 3. The image width and height are resized to $1/5$ of the original ones and for each image, 100-500 SIFT descriptors are extracted. Since the homographies between the reference image and other images in each particular subset are given, we can derive the ground truth matches for the evaluation.

40-180 assignments are extracted as the reliable correspondences using Algorithm 1 in the evaluation. The matching score is calculated as the ratio between the number of correct matches and the smaller value of detected feature numbers from the image pair. The detailed results are demonstrated in Figure 4-7. It is observed that our algorithm generally reaches a higher accuracy compared with the state-of-the-art techniques and the algorithmic performance is stable over all the subsets. Although in some situation such as the JPEG compression the spectral

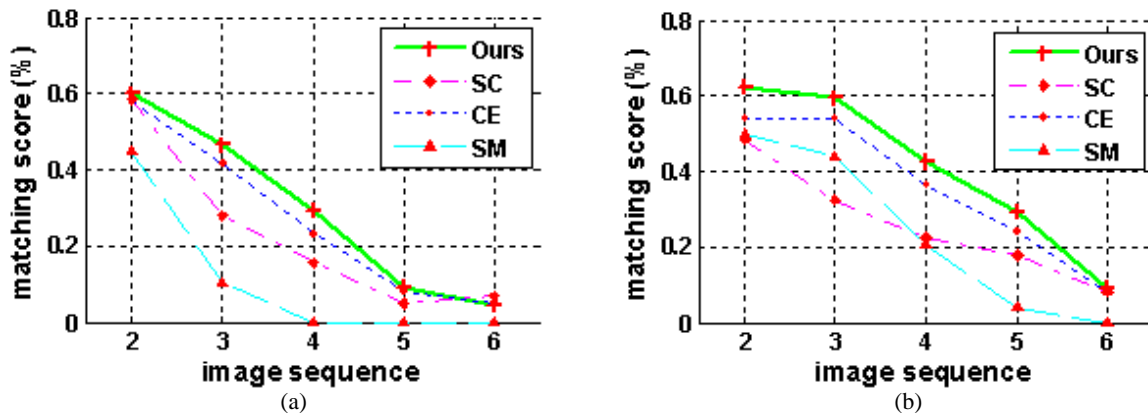[1]http://www.robots.ox.ac.uk/ vgg/ research/affine.

Fig. 4.   Automatic feature matching score on (a) Graffiti and (b) Wall sequence for viewpoint change.
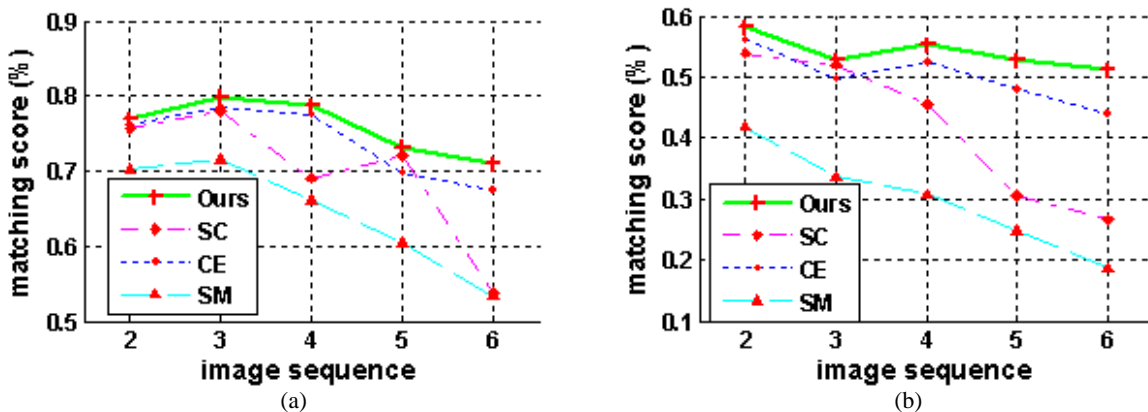

Fig. 5.   Automatic feature matching score on (a) bikes and (b) trees sequence for image blur.

technique shows an excellent performance, it is not so stable in most cases.

Though the spectral based technique also employs geometric information as well as feature similarity in the matching process, our algorithm generally produces a better performance. The main reason is that our algorithm essentially puts different weights on the correspondences and the reliable correspondences are emphasized, while this information is ignored in other state-of-the-art feature matching algorithms.

B. Influence of Reliable Correspondence Number

In the unsupervised configuration, the performance of our algorithm relies on the accuracy of the reliable correspondences, which also deteriorates as the correspondence number increases. It is interesting and necessary to evaluate the performance of our algorithm with respect to the number of automatically selected *reliable* correspondences. Figure 8 shows the correct matching number versus the number of reliable correspondences automatically derived. We can observe that the correct match number increases along with the increase of the reliable
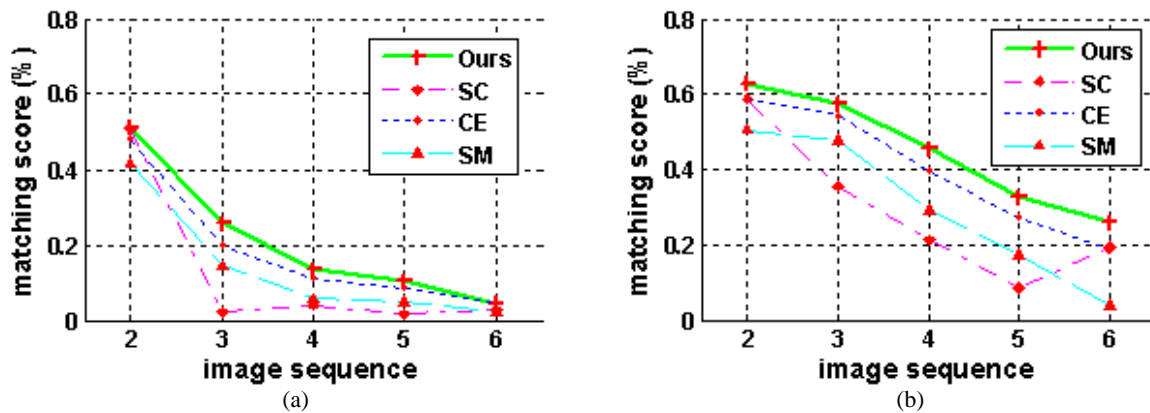
Fig. 6. Automatic feature matching score on (a) bark and (b) boat sequence for zoom and rotation.
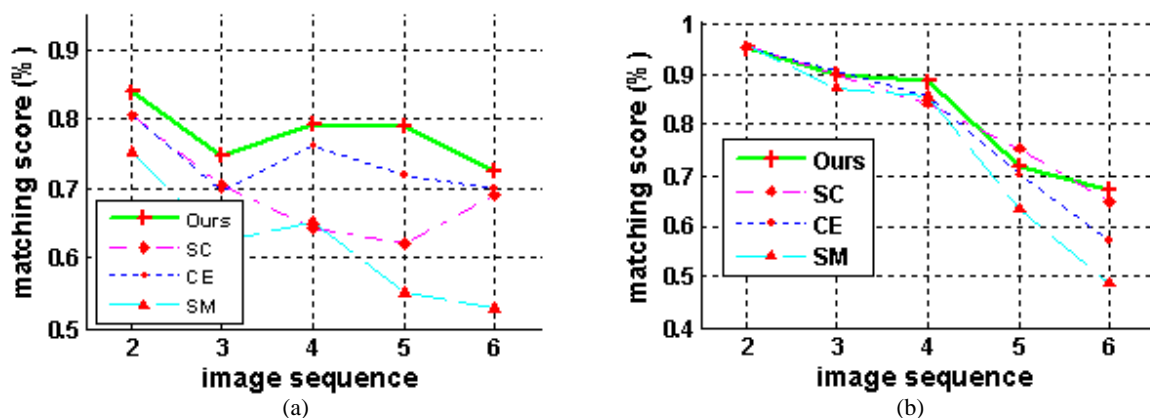

Fig. 7. Automatic feature matching score on (a) leuven for illumination variation and (b) UBC for JPEG compression.

correspondence number within a reasonable range, and then the accuracy falls when the reliable correspondence number becomes too large to give an accurate guidance.

C. Matching Demonstration on Object Recognition Databases

In this subsection, we evaluate our algorithm on the Caltech 101 Object Recognition database[2] and ETH-80 database [3]. Four categories of images are used in this demonstration, i.e., the *motorbikes* and *face* images from Caltech 101 database as well as the *dog* and *horse* images from the ETH-80 database. Since for the objects of different types, the correspondences may not be one-to-one, a threshold of 0.01 is used in the discretization process and thus the correspondence number is determined automatically. For comparison, the matchings with the largest k cosine distances are also plotted as baseline, where k is the number of correspondences determined by our algorithm. The matching results are demonstrated by Figure 9-11, in which the reliable correspondences drawn by hand are marked by red stars, the obscure guidance indicated by human interaction is described by rectangles of different

[2]http://www.vision.caltech.edu/Image Datasets/Caltech101/
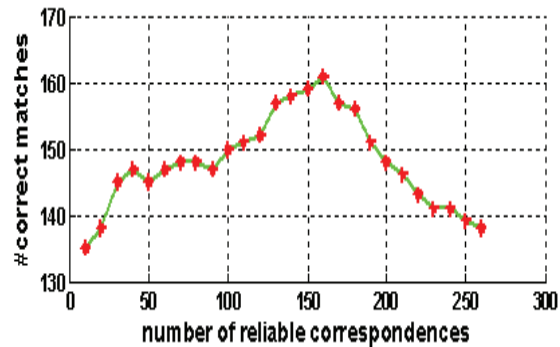[3]http://www.vision.ethz.ch/projects/categorization/

Fig. 8. The number of correct matches v.s. the number of automatically selected reliable correspondences on the first two images of Graffiti database.
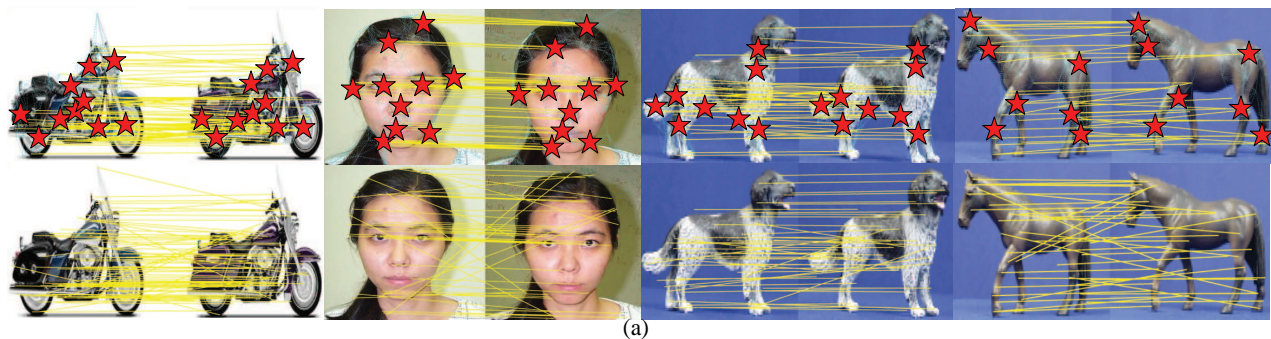

(a)

Fig. 9. Semi-supervised RCP results by manual pairwise correspondence labeling (1st row) v.s. baseline algorithm (2nd row).

colors and the automatically derived reliable correspondences are plotted by small crosses. The correspondence number of the two figures within the same column is the same. From the results we can observe that the matching accuracy is boosted with the guidance of the manually labeled correspondences, and the unsupervised version of our algorithm is also superior over the baseline algorithm.

V. CONCLUSION AND FUTURE WORKS

In this paper, we proposed an efficient feature matching framework that transduces certain number of reliable correspondences to the remaining ones by utilizing both geometric coherency constraints and feature agreements. Furthermore, the framework is naturally extended to incorporate human interactions for promoting feature matching performance. Experimental results showed that our algorithm, both semi-supervised and unsupervised versions, achieves a higher matching accuracy compared to the state-of-the-art techniques. We are planning to further investigate our algorithm with other feature descriptors and explore the combination with the ANMS strategy for reliable correspondence selection.

REFERENCES

[1] Y. Ahn, T. Yoon, and T. Schenk. Reconstruction of 3d object space from imagery by feature-based matching. In *Geoscience and Remote Sensing Symposium (IGARSS)*, July 2005.
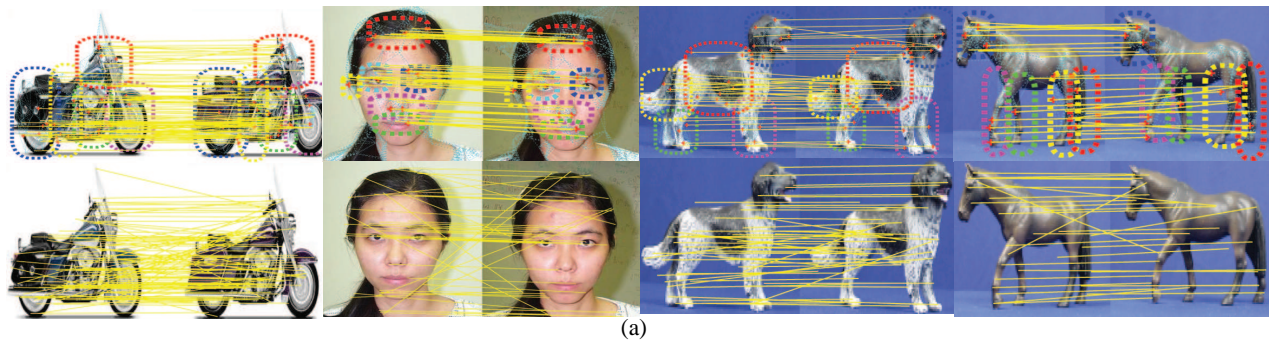
Fig. 10. Semi-supervised RCP results of obscure correspondence guidance (1st row) v.s. baseline algorithm (2nd row).
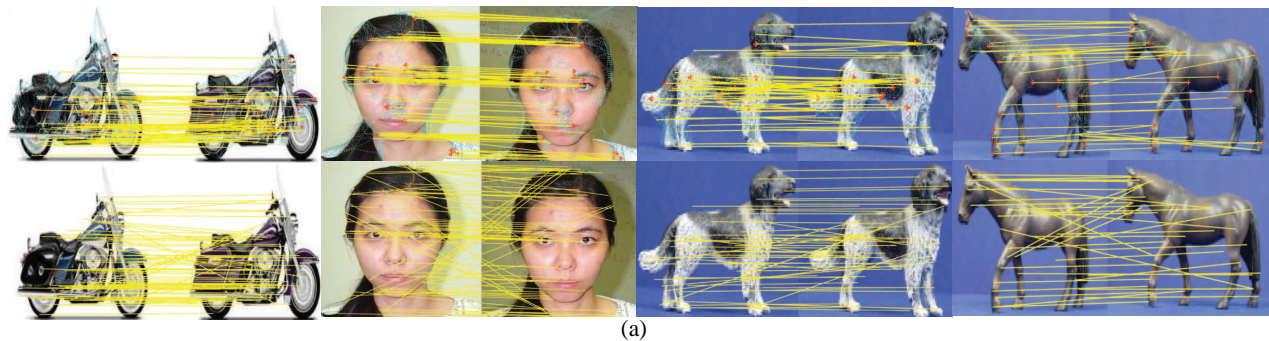

Fig. 11. Matching results of unsupervised RCP (1st row) v.s. baseline algorithm (2nd row).

[2] C. Baillard, C. Schmid, A. Zisserman, and A. Fitzgibbon. Automatic line matching and 3d reconstruction of buildings from multiple views. In *ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery, IAPRS Vol.32, Part 3-2W5*, pages 69–80, September 1999.

[3] T. Beier and S. Neely. Feature-based image metamorphosis. In *Special Interest Group on GRAPHics and Interactive Techniques (SIGGRAPH)*, pages 35–42, 1992.

[4] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. In *Neural Computation*, 15(6), 2003.

[5] M. Belkin, P. Niyogi, and V. Sindhwani. On manifold regularization. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTAT)*, 2005.

[6] A. C. Berg, T. L. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondences. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[7] M. Brown and D. G. Lowe. Recognising panoramas. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pages 1218–1227, 2003.

[8] M. Brown, R. Szeliski, and S. Winder. Multi-image matching using multi-scale oriented patches. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[9] T. Cour, P. Srinivasan, and J. Shi. Balanced graph matching. In *Neural Information Processing Systems (NIPS)*, 2007.

[10] K. Grauman and T. Darrell. The pyramid match kernel: discriminative classification with sets of image features. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2005.

[11] K. Hariharakrishnan and D. Schonfeld. Fast object tracking using adaptive block matching. In *IEEE TRANSACTIONS ON MULTIMEDIA*, 7(5), October 2005.

[12] T. Kadir, A. Zisserman, and M. Brady. An affine invariant salient region detector. In *European Conference on Computer Vision (ECCV)*, 2004.

[13] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2005.

[14] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision (IJCV)*, 2003.

[15] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of IEEE International Conference on Computer Vision (ICCV), Volume 2*, page 1150, Washington, DC, USA, 1999. IEEE Computer Society.

[16] T. Luhmann and W. Tecklenburg. 3-d object reconstruction from multiple-station panorama imagery. In *Proceedings of the ISPRS working group V/1 'Panoramic Photogrammetry Workshop'*, February 2004.

[17] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. In *International Journal of Computer Vision (IJCV)*, 2004.

[18] C. Schellewald and C. Schnörr. Probabilistic subgraph matching based on convex relaxation. In *Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, 2005.

[19] J. Shi and J. Malik. Normalized cuts and image segmentation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(8):888–905, 2000.

[20] H. Bai and E. Hancock. Graph matching using spectral embedding and semidefinite programming. In *British Machine Vision Conference (BMVC)*,2004.