Structural Graph Matching Using the EM Algorithm and Singular Value Decomposition

Bin Luo and Edwin R. Hancock

Abstract—This paper describes an efficient algorithm for inexact graph matching. The method is purely structural, that is to say, it uses only the edge or connectivity structure of the graph and does not draw on node or edge attributes. We make two contributions. Commencing from a probability distribution for matching errors, we show how the problem of graph matching can be posed as maximum-likelihood estimation using the apparatus of the EM algorithm. Our second contribution is to cast the recovery of correspondence matches between the graph nodes in a matrix framework. This allows us to efficiently recover correspondence matches using singular value decomposition. We experiment with the method on both real-world and synthetic data. Here, we demonstrate that the method offers comparable performance to more computationally demanding methods.

Index Terms—Inexact graph matching, EM algorithm, matrix factorization, mixture models, Delaunay triangulations.

✦

1 INTRODUCTION

GRAPH matching is a task of pivotal importance in high-level vision since it provides a means by which abstract pictorial descriptions can be matched to one another. Unfortunately, since the process of eliciting graph structures from raw image data is a task of some fragility due to noise and the limited effectiveness of the available segmentation algorithms, graph matching is invariably approached by inexact means [38], [35]. Because of this, many high-level matching techniques have weakened the role of structural information and have relied more heavily on the use of attribute relations. This is disappointing since structural graph representations provide abstractions that convey important visual invariances. It is for this reason that we return to the problem of structural graph matching in this paper. We make two contributions: First, we aim to render the process robust to structural error using the apparatus of the EM algorithm. Second, we cast the resulting statistical utility measure into a matrix setting and show how matching can be realized using singular value decomposition.

1.1 Literature Review

We set our work in context with a brief review of the related literature. Some of the pioneering work on graph matching was undertaken in the early 1970's by Barrow and Popplestone [2] and by Fischler and Enschlager [21]. These

two studies provided proof of concept for the use of relational structures in high-level pictorial object recognition. Over the intervening three decades, there has been a sustained research activity. Broadly speaking, the work reported in the literature can be divided into three areas.

The first of these is concerned with defining a measure of relational similarity. Much of the early work here was undertaken in the structural pattern recognition literature. For instance, Shapiro and Haralick [38] showed how inexact structural representations could be compared by counting consistent subgraphs. This similarity measure was refined by Eshera and Fu [17] and by Sanfeliu and Fu [35] who showed how the concept of string edit distance could be extended to graphical structures. The formal basis of graph edit distance has recently been extended by Bunke and his coworkers [9], [6] who have shown, among other things, that the edit distance is related to the size of the maximum common subgraph. More recently, Tirthapura et al. have shown how the classical Levenshtein distance can be used to match shock graphs representing 2D skeletal shapes [43].

Much of the work described above adopts a heuristic- or goal-directed approach to measuring graph similarity. The second issue addressed in our literature survey is that of how to develop more principled statistical measures of similarity. This endeavor involves the modeling of the processes of structural error present in the graph-matching problem. Wong and You [50] made one of the first contributions here by defining an entropy measure for structural graph matching. Boyer and Kak [4] also adopted an information theoretic approach, but worked instead with attribute relations. Using a probabilistic relaxation framework Christmas et al. [11] have developed a statistical model for pairwise attribute relations. Working in the purely structural domain, Wilson and Hancock [49] have

- B. Luo is with the Department of Computer Science, University of York, York YO1 5DD, UK and Anhui University, People's Republic of China. E-mail: luo@cs.york.ac.uk.
- E.R. Hancock is with the Department of Computer Science, University of York, York YO1 5DD, UK. E-mail: erh@cs.york.ac.uk.

derived probability distributions for the relational errors that occur when there is significant graph corruption. More recently, Cross and Hancock [13] have developed a variant of the EM algorithm in which the structural error model of Wilson and Hancock [49] is used to improve the alignment of triangulated point-sets under perspective geometry.

The third issue is that of optimization. Here, there have been several attempts to use both continuous and discrete optimization methods to locate optimal graph matches. Turning our attention first to discrete optimization methods, there have been several attempts to apply techniques such as simulated annealing [24], genetic search [14], and tabu search [48] to the graph matching problem. However, continuous optimization methods provide attractive alternatives since their fixed points and convergence properties are usually better understood than their discrete counterparts. However, the main difficulty associated with mapping a discretely defined search problem onto a continuous optimization method is that of embedding. There are several ways in which this embedding can be effected for the problem of graph matching. The most straightforward of these is to pose the graph-matching problem as that of recovering a permutation matrix which preserves edge or adjacency structure. For instance, Kosowsky and Yuille have cast the problem into a statistical physics setting and have recovered a continuous representation of the permutation matrix using mean-field update equations [52]. Gold and Rangarajan [23] have exploited the stochastic properties of Sinkhorn matrices to recover the matches using a soft-assign update algorithm. Umeyama [45] takes a more conventional least-squares approach and shows how an eigendecomposition method can be used to recover the permutation matrix. An alternative representation has recently been developed by Pelillo [32] which involves an embedding based on the association graph. Matches are located by using the replicator equations of evolutionary game-theory to locate the maximal clique of the association graph, i.e., the maximum common subgraph, of the two graphs being matched. Subsequently, this method has also been applied to shock-graph matching [31].

Closely related to this work on recovering permutation structure by continuous embedding is the literature on spectral graph theory. This is a term applied to a family of techniques that aim to characterize the global structural properties of graphs using the eigenvalues and eigenvectors of the adjacency matrix [12]. In the computer vision literature, there have been a number of attempts to use spectral properties for graph matching, object recognition, and image segmentation. Umeyama has an eigendecomposition method that matches graphs of the same size [45]. Borrowing ideas from structural chemistry, Scott and Longuet-Higgins were among the first to use spectral methods for correspondence analysis [36]. They showed how to recover correspondences via singular value decomposition on the point association matrix between different images. In keeping more closely with the spirit of spectral graph theory, yet seemingly unaware of the related literature, Shapiro and Brady [39] developed an extension of the Scott and Longuet-Higgins method, in which point sets are matched by comparing the eigenvectors of the point proximity matrix. Here, the proximity matrix is constructed by computing the Gaussian weighted distance between points. The eigenvectors of the proximity matrices can be viewed as the basis vectors of an orthogonal transformation on the original point identities. In other words, the components of the eigenvectors represent mixing angles for the transformed points. Matching between different point-sets is effected by comparing the pattern of eigenvectors in different images. Shapiro and Brady's method can be viewed as operating in the attribute domain rather than the structural domain. Horaud and Sossa [27] have adopted a purely structural approach to the recognition of line-drawings. Their representation is based on the immanental polynomials for the Laplacian matrix of the line-connectivity graph. By comparing the coefficients of the polynomials, they are able to index into a large database of line-drawings. In another application involving indexing into large databases, Sengupta and Boyer [37] have used property matrix spectra to characterize line-patterns. Various attribute representations are suggested and compared. Shokoufandeh et al. [40] have shown how graphs can be encoded using local topological spectra for shape recognition from large databases.

Although formally elegant, the main limitation of these matrix methods is their inability to cope with graphs of different sizes. This means that they cannot be used when significant levels of structural corruption are present.

1.2 Motivation

From this review of the literature, we draw the following observation: First, the use of principled probabilistic methods for gauging similarity has met with considerable success for inexact graph matching. Second, although more computationally elegant, matrix methods for graph matching have failed to cope with the realistic cases of inexact graph matching or matching graphs of different size. Moreover, there has been little attempt to combine these two pieces of work.

Based on these observations, our aim in this paper is to cast the statistical matching of graphs into a matrix representation and to exploit singular value methods to efficiently recover correspondences. We commence by developing a likelihood function for the graph-matching problem. This treats the graph to be matched (the data graph) as observed data and the set of correspondences with the available model (the model graph) as hidden variables. Accordingly, we construct a mixture model over the set of correspondences between the nodes of the data graph and those of the model graph. We adopt a Bernoulli model for the probability distribution of the correspondence errors encountered in matching the data graph to the model graph. The existence, or otherwise, of

correspondence errors is gauged using the edge-consistency of the pattern of matches.

Using the likelihood function for the graph-matching problem, we develop an algorithm for recovering the pattern of correspondence matches. Since the likelihood function has a mixture-structure, we use the expectation-maximization (EM) algorithm of Dempster et al. [16] to iteratively estimate a set of assignment variables which indicate the state of match between the two graphs. The EM algorithm provides a principled way for recovering maximum-likelihood solutions to problems posed in terms of missing or hidden data. From a computational standpoint, the EM algorithm relies on interleaved iterative steps. In the maximization step, a parameter is estimated so as to maximize the value of the expected log-likelihood function. The expectation step updates the a posteriori probabilities of the hidden variables, which in their turn are needed to weight contributions to the expected log-likelihood function.

By adopting a matrix representation, we show that when the distribution of correspondence errors is modeled using a Bernoulli distribution, then the expected log-likelihood function of the EM algorithm is related to the weighted product of the adjacency matrices. The weighting facilitated by the EM algorithm provides a means of excluding structural errors. Using this principled similarity measure, the maximization step of the EM algorithm can be realized via singular value decomposition. Specifically, the diagonalization of the weighted adjacency measure delivers correspondence matches. The resulting matching method can be applied to graphs of different size. Moreover, as we will demonstrate in our experiments, it can accommodate severe levels of structural corruption.

Finally, it is worth comparing our iterative matrix-based graph-matching method with the use of subgraph isomorphisms for matching. This is a classical approach. One of the best-known algorithms for locating exact subgraph isomorphisms is that of Ullman [44], which uses tree search with backtracking. For inexact graph matching there are several extensions of the idea which include the edit-distance methods of Eshera and Fu [17] and Sanfeliu and Fu [35] and the method developed by Bunke and Allerman [7]. An important development is the association-graph method of Ambler et al. [1]. Here, subgraph isomorphisms which correspond to the maximum common subgraph are found by searching for the maximum clique of the association graph. The method has been used successfully for stereo correspondence by Horaud and Skordas [26]. Moreover, Wilson and Hancock [49] have shown how the association graph can be combined with a Bayesian discrete relaxation process to match graphs which are subject to considerable structural corruption. Pelillo et al. [31] have developed a probabilistic relaxation scheme for efficiently computing the size of the maximum common subgraph. Bunke has shown the relationship between the size of the maximum common subgraph and the graph edit distance [6]. Further work by Bunke and Messmer [8], has developed a technique in which the search for a maximum common subgraph can be rendered efficient using a subgraph decomposition and by tabulating the associated isomorphisms. The main similarities and difference between our EM algorithm and these subgraph isomorphism methods are as follows: The main difference is that we adopt a probabilistic framework and use optimization rather than search. Hence, while our method facilitates evidence combination, it is prone to convergence to local optima. The similarity of the method resides in the fact that the matrix product of the correspondence probabilities with the adjacency matrix of the two graphs plays a role which is reminiscent of an association structure.

2 A LIKELIHOOD FUNCTION FOR GRAPH MATCHING

Our overall goal in this paper is to develop a maximum-likelihood framework for structural graph matching. In this section, we develop the likelihood function underpinning our study. To commence, we must define some notation. We use the notation $G = (V, E)$ to denote the graphs under match, where V is the set of nodes and E is the set of edges. Our aim in matching is to associate nodes $V_D = \{x_1, x_2, \ldots \ldots x_{|V_D|}\}$ in a graph $G_D = (V_D, E_D)$ representing data to be matched against those from the set $V_M = \{y_1, y_2, \ldots \ldots, y_{|V_M|}\}$ in a graph $G_M = (V_M, E_M)$ representing an available model. Formally, the matching is represented by a function from the nodes in the data graph G_D to those in the model graph G_M. Suppose that the state of match between the two graphs is represented by the function $f : V_D \to V_M$ from the nodes of the data graph to those of the model graph. We will use Latin letters to denote nodes from the data graph and Greek letters to denote nodes from the model graph. Hence, the statement $f^{(n)}(a) = \alpha$ means that the node $a \in V_D$ is assigned the label or symbol $\alpha \in V_M$.

One of the goals in this paper is to show how the two graphs can be matched using matrix factorization methods. Therefore, we introduce some matrix notation to represent the graphs. To this end, we define a $|V_D| \times |V_M|$ matching matrix $S^{(n)}$ whose elements are assignment variables which convey the following meaning

$$s_{a\alpha} = \begin{cases} 1 & \text{if } f(a) = \alpha \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

We represent the structure of the two graphs using a $|V_D| \times |V_D|$ adjacency matrix D for the data graph and a $|V_M| \times |V_M|$ adjacency matrix M for the model graph. The elements of the adjacency matrix for the data graph are defined as follows:

$$D_{ab} = \begin{cases} 1 & \text{if} (a, b) \in E_D \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

while those for the model graph are defined to be

$$M_{\alpha\beta} = \begin{cases} 1 & \text{if} (\alpha, \beta) \in E_M \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

Since we are working with undirected graphs, the two adjacency matrices are symmetric, i.e., $D = D^T$ and $M = M^T$.

Having introduced the necessary formalism, we now proceed to develop our maximum-likelihood framework for graph matching. We seek the matrix of assignment variables that maximizes the conditional likelihood of the observed data graph given the available model graph. Hence, we seek the matrix of assignment variables which satisfies the condition

$$S = \arg\max_{\hat{S}} P(G_D|G_M, \hat{S}). \tag{4}$$

Underpinning our model of the conditional-likelihood function is the idea that the correspondence matches assigned to the nodes of the data graph are hidden variables which have arisen through a noisy observation process. In other words, we must entertain the possibility that any single node of the data graph may be in correspondence with any of the nodes in the model graph. To capture this feature of the graph-matching problem, we construct a mixture model over the set of possible correspondences. We follow the standard approach to constructing the likelihood function for a mixture distribution. This involves factorizing the likelihood function over the observed data (i.e., the nodes of the data graph) and summing over the hidden or unobserved variables (i.e., the corresponding nodes in the model graph). As a result, we write

$$P(G_D|G_M, S) = \prod_{a \in V_D} \sum_{\alpha \in V_M} p(x_a|y_\alpha, S), \tag{5}$$

where $p(x_a|y_\alpha, S)$ is the probability that data-graph node a is in correspondence with the model-graph node α under the matrix of assignment variables S. This formula assumes that the nodes of the graph G_D are conditionally independent given the nodes of the graph G_M.

In order to proceed, we require a model for the observation density $p(x_a|y_\alpha, S)$. We commence from the definition of conditional probability and write,

$$P(x_a|y_\alpha, S) = \frac{P(x_a, y_\alpha, S)}{P(y_\alpha, S)}. \tag{6}$$

Under the assumption that the observation density is factorial over the parameters of the mixture model, i.e., the set of assignment variables, then we can write

$$P(x_a|y_\alpha, S) = \frac{\left\{\prod_{b \in V_D} \prod_{\beta \in V_M} P(s_{b\beta}|x_a, y_\alpha)\right\} P(x_a, y_\alpha)}{\left\{\prod_{b \in V_D} \prod_{\beta \in V_M} P(s_{b\beta}|y_\alpha)\right\} P(y_\alpha)}. \tag{7}$$

After some rearrangement using the definitions of conditional probability, we find that

$$P(x_a|y_\alpha, S) =$$
$$\frac{\left\{\prod_{b \in V_D} \prod_{\beta \in V_M} \frac{P(x_a|y_\alpha, s_{b\beta})P(y_\alpha|s_{b\beta})P(s_{b\beta})}{P(x_a, y_\alpha)}\right\} P(x_a, y_\alpha)}{\left\{\prod_{b \in V_D} \prod_{\beta \in V_M} \frac{P(y_\alpha|s_{b\beta})P(s_{b\beta})}{P(y_\alpha)}\right\} P(y_\alpha)}. \tag{8}$$

Canceling the terms $P(y_\alpha|s_{b\beta})$ and $P(s_{b\beta})$ which appear under the products in the numerator and denominator and collecting together terms, the above expression simplifies to

$$P(x_a|y_\alpha, S) = \left[\frac{1}{P(x_a|y_\alpha)}\right]^{|V_D| \times |V_M| - 1} \prod_{b \in V_D} \prod_{\beta \in V_M} P(x_a|y_\alpha, s_{b\beta}). \tag{9}$$

If we further assume that the data-graph node x_a is conditionally dependant on the model-graph node y_α only in the presence of the correspondence matches S, then $P(x_a|y_\alpha) = P(x_a)$. Hence, we can write

$$P(x_a|y_\alpha, S) = B_a \prod_{b \in V_D} \prod_{\beta \in V_M} P(x_a|y_\alpha, s_{b\beta}), \tag{10}$$

where the constant

$$B_a = \left[\frac{1}{P(x_a)}\right]^{|V_D| \times |V_M| - 1}$$

depends only on the identity of the data-graph node x_a.

Next, we develop a model for the probability distribution for the observed set of correspondences between the nodes of the data and the model graphs given the current set of assignment parameters, i.e., $P(x_a|y_\alpha, s_{b\beta})$. Our model draws on the recent work of Wilson and Hancock [49] and assumes that the observed data-graph nodes are derived from the model-graph nodes through a Bernoulli distribution. The parameter of this distribution is the probability of correspondence error P_e. The idea behind this model is that the data-graph node x_a can emit a symbol y_α drawn from the set of model-graph nodes. The probability that this symbol is the correct correspondence is $1 - P_e$, while the probability that it is in error is P_e. To gauge the correctness of the emitted symbol, we check whether the nodes a and b of the data graph are matched to a valid edge $(\alpha, \beta) \in E_m$ of the model graph. To test for edge-consistency, we make use of the quantity $D_{a,b}M_{\alpha\beta}s_{b\beta}$. This is unity if the label-assignment $f(b) = \beta$ can be made to node x_b, in such a way that the data-graph edge $(a, b) \in E_D$ is matched to an edge $(\alpha, \beta) \in E_m$ of the model graph. When this condition is not met, then the quantity is zero. In other words,

$$D_{a,b}M_{\alpha\beta}s_{b\beta} =$$
$$\begin{cases} 1 & \text{if } (a, b) \in E_D \text{ and } (\alpha, \beta) \in E_M \text{ and } f(b) = \beta \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

Using this switching property, the Bernoulli distribution becomes

$$P(x_a|y_\alpha, s_{b\beta}) = (1 - P_e)^{D_{ab}M_{\alpha\beta}s_{b\beta}} P_e^{1 - D_{ab}M_{\alpha\beta}s_{b\beta}}. \tag{12}$$

Using the factorial assumption and the distribution rule, the observation density becomes

$$P(x_a|y_\alpha, S) = B_a \prod_{b \in V_D} \prod_{\beta \in V_M} (1 - P_e)^{D_{ab}M_{\alpha\beta}s_{b\beta}} P_e^{1 - D_{ab}M_{\alpha\beta}s_{b\beta}}. \tag{13}$$

This expression is exponential in character. It can be rewritten as a natural exponential function

$$P(x_a|y_\alpha, S) = K_a \exp\left[\mu \sum_{b \in V_D} \sum_{\beta \in V_M} D_{ab} M_{\alpha\beta} s_{b\beta}\right], \quad (14)$$

where

$$\mu = \ln \frac{1 - P_e}{P_e} \quad (15)$$

and

$$K_a = P_e^{|V_D| \times |V_M|} B_a. \quad (16)$$

Finally, the likelihood function becomes

$$P(G_D|G_M, S) = \prod_{a \in V_D} \sum_{\alpha \in V_M} K_a \exp\left[\mu \sum_{b \in V_D} \sum_{\beta \in V_M} D_{ab} M_{\alpha\beta} s_{b\beta}\right] \quad (17)$$

and the corresponding log-likelihood function for the assignment matrix is

$$\mathcal{L}(S) = \sum_{a \in V_D} \log\left\{\sum_{\alpha \in V_M} K_a \exp\left[\mu \sum_{b \in V_D} \sum_{\beta \in V_M} D_{ab} M_{\alpha\beta} s_{b\beta}\right]\right\}. \quad (18)$$

Unfortunately, because of the mixture structure, the direct estimation of the matrix of assignment variables S from the log-likelihood function is not tractable in closed form. For this reason, in the next section, we explain how the expectation-maximization algorithm may be used instead.

3   EXPECTATION-MAXIMIZATION

Having developed our computational model which poses the graph-matching problem in a maximum-likelihood framework, in this section, we provide a concrete algorithm for recovering the parameters of the underlying mixture-model. We choose to use the EM algorithm originally introduced by Dempster et al. [16]. The utility measure underpinning the algorithm is the expected log-likelihood function. The basic idea underlying the algorithm is to iterate between the interleaved expectation and maximization steps until convergence is reached. Expectation involves updating the a posteriori probabilities of the missing data using the most recently available parameter estimates. In the maximization phase, the model parameters are recomputed to maximize the expected value of the incomplete data likelihood.

Several authors have considered how the problem of point pattern matching can be addressed using the EM framework. Utans recovers translation parameters [46], while Gold et al. are more ambitious in matching under affine transformation [23]. Recently, the EM algorithm has been exploited in the recovery of object pose by both Wells [47] and by Hornegger and Nieman [28]. In a demanding practical application, Moss and Hancock have shown how the algorithm can be used to register cartographic models against noisy and incomplete radar data [30]. In contrast to these approaches, the main contribution of this paper is to demonstrate the effectiveness of the algorithm in matching symbolic relational graphs without recourse to either an explicit transformational model or attribute information.

3.1   Expected Log-Likelihood Function

The utility measure underpinning the EM algorithm is the conditional expected log likelihood. The basic idea is to identify updated parameters that maximize their expected likelihood conditional upon the previously available iterates. This utility measure is frequently referred to as the incomplete data likelihood. In our matching problem, the parameters are the discrete matching assignments to the nodes of the data graph. Incompleteness originates from the fact that the matching configurations are not directly observable from the data. In other words, although we can observe the structure of the model and data graphs, the matching-function f is hidden from us. The incomplete data likelihood is obtained by weighting the individual contributions by the appropriate a posteriori matching probabilities. In a more general context, Dempster et al. [16] observed that maximizing a weighted log-likelihood function of this sort was equivalent to maximizing the conditional expectation of the likelihood for a new parameter set given an old parameter set.

For our graph-matching problem, and from the Bayes theorem and the well-known development of the EM algorithm [15], [10], [22], maximization of the expectation of the conditional likelihood is equivalent to maximizing the weighted log-likelihood function

$$\Lambda\left(S^{(n+1)}|S^{(n)}\right) = \sum_{a \in V_D} \sum_{\alpha \in V_M} P\left(y_\alpha|x_a, S^{(n)}\right) \ln P\left(x_a|y_\alpha, S^{(n+1)}\right), \quad (19)$$

where $S^{(n)}$ indicates the matrix of assignment variables taken at iteration n of the EM algorithm. Hence, the a posteriori correspondence matching probabilities computed at iteration n, i.e., $P(y_\alpha|x_a, S^{(n)})$ are used to weight the iteration $n + 1$ contributions to the log-likelihood function.

Using the expected log-likelihood function, the maximum-likelihood matrix of assignment variables is the one which satisfies the condition

$$S^{(n+1)} = \arg\max_{\hat{S}} \Lambda\left(\hat{S}|S^{(n)}\right). \quad (20)$$

One way to realize the update process is by parallel iterative local gradient ascent. In the next section, we show how the expected log-likelihood function can be recast in a matrix framework. This allows us to realize the update procedure more efficiently using singular value decomposition.

3.2   Matrix Representation

To commence, we note that, when the distribution function for the assignment variables is substituted from (14), the expected log-likelihood function becomes

$$\Lambda\left(S^{(n+1)}|S^{(n)}\right) =$$
$$\sum_{a \in V_D} \sum_{b \in V_D} \sum_{\alpha \in V_M} \sum_{\beta \in V_M} Q_{a\alpha}^{(n)}\left\{\ln K_a + \mu D_{ab} M_{\alpha\beta} s_{b\beta}^{(n+1)}\right\}, \quad (21)$$

where we have introduced the $|V_D| \times |V_M|$ matrix $Q^{(n)}$ whose elements $Q_{a\alpha}^{(n)} = P(y_\alpha|x_a, S^{(n)})$ are set equal to the a posteriori probability of correspondence match between the data-graph node a and the model-graph node α at iteration n of the EM algorithm.

The expression for the expected log-likelihood function $\Lambda(S^{(n+1)}|S^{(n)})$ simplifies if we note that the first term under the curly braces is simply proportional to the normalized probability mass over the state-space of the matching process. In other words, it contributes a constant amount

$$\sum_{a \in V_D} \sum_{\alpha \in V_M} Q_{a\alpha}^{(n)} \ln K_a = \sum_{a \in V_D} \ln K_a. \qquad (22)$$

Based on this observation, the critical quantity in determining the update direction for maximum-likelihood matches is

$$\hat{\Lambda}(S^{(n+1)}|S^{(n)}) = \sum_{a \in V_D} \sum_{b \in V_D} \sum_{\alpha \in V_M} \sum_{\beta \in V_M} Q_{a\alpha}^{(n)} D_{ab} M_{\alpha\beta} s_{b\beta}^{(n+1)}. \qquad (23)$$

To write this component of the expected log-likelihood function in matrix notation, we group the indices as follows:

$$\hat{\Lambda}(S^{(n+1)}|S^{(n)}) = \sum_{a \in V_D} \sum_{b \in V_D} \sum_{\alpha \in V_M} \sum_{\beta \in V_M} Q_{a\alpha}^{(n)} M_{\alpha\beta} \left(s_{\beta b}^{(n+1)}\right)^T D_{ba}^T. \qquad (24)$$

Using the repeated index contraction for matrix products, this is equal to

$$\hat{\Lambda}(S^{(n+1)}|S^{(n)}) = Tr\left[Q^{(n)} M \left(S^{(n+1)}\right)^T D^T\right], \qquad (25)$$

where $Q^{(n)}$ is the matrix of correspondence probabilities. Finally, using the fact that the trace of a product of matrices is invariant under cyclic permutation of the matrix order

$$\hat{\Lambda}(S^{(n+1)}|S^{(n)}) = Tr\left[D^T Q^{(n)} M \left(S^{(n+1)}\right)^T\right]. \qquad (26)$$

As a result, we confine our attention to the quantity $Tr[D^T Q^{(n)} M S^T]$. In Umeyama's [45] work the eigendecomposition method attempts to find the permutation matrix $\mathcal{P}$ which maximizes $D^T \mathcal{P} M$. The utility measure used by the EM algorithm can be regarded as a weighted version of Umeyama's least-squares criterion.

3.3 Maximization

The maximization step of the EM algorithm can be stated as that of recovering the set of correspondence indicators $S^{(n+1)}$ which satisfies the condition

$$S^{(n+1)} = \arg \max_{\hat{S}} Tr\left[D^T Q^{(n)} M \hat{S}^T\right]. \qquad (27)$$

In other words, the utility measure gauges the degree of correlation between the edge-sets of the two graphs under the weighted permutation structure induced by the correspondence probabilities.

To locate the updated set of correspondence indicators, we use the extremum principal reported by Scott and Longuet-Higgins [36]. Their result is as follows: Suppose that G is a positive definite $|V_D| \times |V_M|$ matrix. They have shown how the $|V_D| \times |V_M|$ orthogonal matrix R that maximizes the quantity $Tr[GR^T]$ may be found by performing singular value decomposition. To do this, they perform the matrix factorization $G = V\Delta U^T$, where V is a $|V_D| \times |V_D|$ orthogonal matrix, U is a $|V_M| \times |V_M|$ orthogonal matrix, and Δ is a $|V_D| \times |V_M|$ matrix whose off-diagonal elements $\Delta_{i,j} = 0$ if $i \neq j$ and whose "diagonal" elements $\Delta_{i,i}$ are nonzero. Suppose that E is the matrix obtained from Δ by making the diagonal elements $\Delta_{i,i}$ unity. The matrix R which maximizes $Tr[GR^T]$ is $R = VEU^T$. This extremum principle may be applied to our graph matching problem if we make the substitution $G = D^T Q^{(n)} M$ and perform the singular value decomposition $D^T Q^{(n)} M = V\Delta U^T$ to obtain R. This matrix satisfies the condition

$$R = \arg \max_{\hat{R}} Tr\left[D^T Q^{(n)} M \hat{R}^T\right]. \qquad (28)$$

Provided that the matrix $D^T Q^{(n)} M$ is positive-definite, then the elements of R are real.

Although this extremum principle is useful, it is not entirely suited to our needs. The reasons for this are that the elements of R cannot be interpreted as probabilities since they are neither guaranteed to be positive nor are they normalized. Furthermore, they cannot be interpreted as assignment indicators since they are not binary in nature. To overcome these difficulties, we follow Scott and Longuet-Higgins by testing the elements of R to obtain a matrix of binary correspondence indicators $S^{(n+1)}$. If the element $R_{a,\alpha}$ is the maximum value for both the row and column that contains it, then the assignment indicator $s_{a,\alpha}^{(n+1)}$ is set to unity. Otherwise, it is set to zero. As a result, the updated set of correspondence indicators is

$$s_{a\alpha}^{(n+1)} = \begin{cases} 1 & \text{if } R_{a\alpha} = \arg \max_{b\beta} R_{b\beta} \\ 0 & \text{otherwise.} \end{cases} \qquad (29)$$

There are alternatives to this decision step. For instance, we could use the Sinkhorn normalization idea of Gold and Rangarajan [23] to preprocess the matrix, or, we could apply a bipartite graph matching algorithm to the matrix R. The choice above is dictated by reasons of simplicity.

3.4 Expectation

In the expectation step of the EM algorithm, the a posteriori probabilities of the hidden data are computed from the component densities appearing in the mixture distribution. This is done by applying the Bayes theorem. At iteration $n + 1$, we have

$$P\left(y_\alpha|x_a, S^{(n+1)}\right) = \frac{p\left(x_a|y_\alpha, S^{(n)}\right) \pi_\alpha^{(n)}}{\sum_{\alpha \in V_M} p\left(x_a|y_\alpha, S^{(n)}\right) \pi_\alpha^{(n)}}, \qquad (30)$$

where

$$\pi_\alpha^{(n)} = \frac{1}{|V_D|} \sum_{a \in V_D} P\left(y_\alpha|x_a, S^{(n)}\right). \qquad (31)$$

We can re-express the a posteriori probabilities using the indicator variables in the following manner:
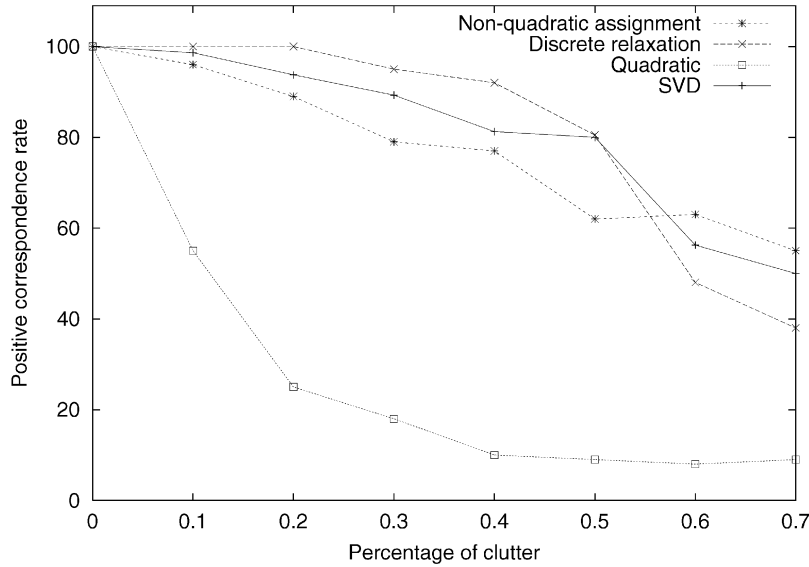
Fig. 1. Sensitivity study for graphs of different size.

$$Q_{a\alpha}^{(n+1)} =$$

$$\frac{K_a \exp\left[-\mu \sum_{b \in V_D} \sum_{\beta \in V_M} D_{ab} M_{\alpha\beta}\left(1 - s_{b\beta}^{(n)}\right)\right] \pi_\alpha^{(n)}}{\sum_{\alpha' \in V_M} K_a \exp\left[-\mu \sum_{b \in V_D} \sum_{\beta \in V_M} D_{ab} M_{\alpha\beta}\left(1 - s_{b\beta}^{(n)}\right)\right] \pi_{\alpha'}^{(n)}}.$$

$$(32)$$

Since the constant K_a depends only on the index of the model graph node x_a, it cancels between the numerator and the denominator.

At this point, it is worth pointing out that there are alternative views of the E-step of the EM algorithm. For instance, in the graphical models literature, it is common to introduce a set of indicator variables to model the affinities between observed and missing data. These variables may be updated in an optimization step which uses mean-field annealing.

4   SOFT ASSIGN

Before we proceed to experiment with the new graph matching process, it is interesting to briefly review the standard quadratic formulation of the matching problem investigated by Simic [41], Sugarnathan et al. [42] and Gold and Rangarajan [23]. Here, the aim has been to deploy continuous optimization methods such as the relatively heuristic graduated assignment [23], [3] or the more principled mean-field theory [25], [51], [34], [52], [33] to update a set of assignment variables representing the matching process. Although there are many variants of the idea, the common feature of these algorithms is to commence from the quadratic cost function

$$\mathcal{E}_H = -\frac{1}{2} \sum_{a \in V_D} \sum_{\alpha \in V_M} \sum_{b \in V_D} \sum_{\beta \in V_M} D_{ab} M_{\alpha\beta} s_{a\alpha} s_{b\beta}. \qquad (33)$$
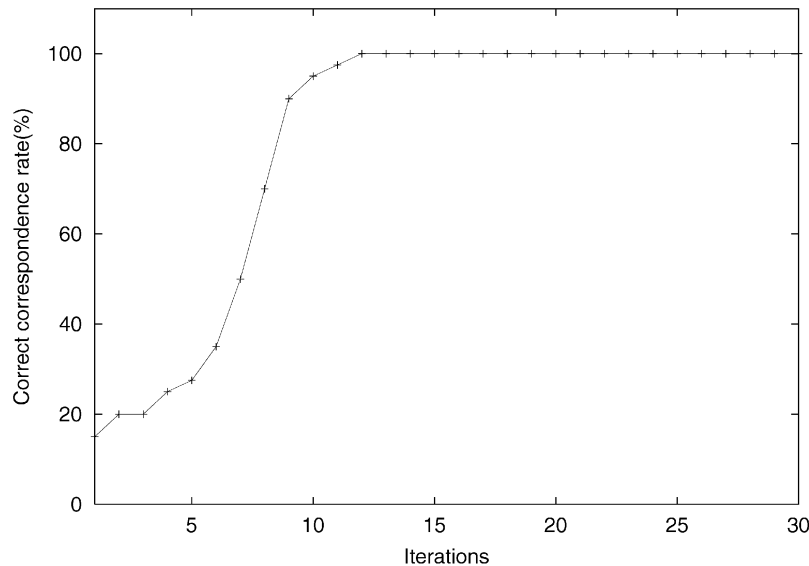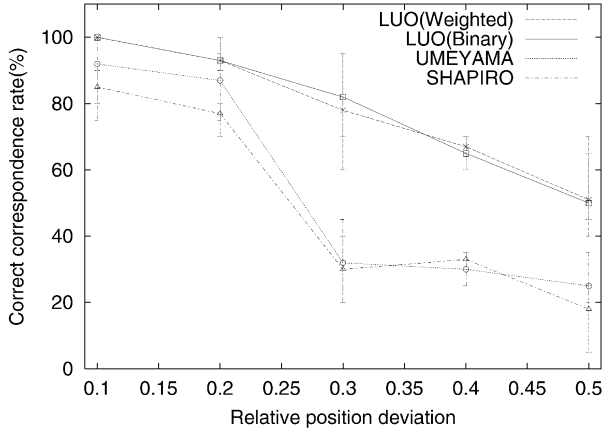

Fig. 2. Convergence rate.

Fig. 3. Comparison of the four eigendecomposition methods for graphs with the same number of nodes.

Fig. 4. Fraction of edge errors as a function of the relative standard deviation of the point-position error.

Performance has been demonstrated to be enhanced if additional, sometimes nonquadratic, terms are added. Examples include node self-amplification term that encourages binary solutions [23] and the addition of a logarithmic barrier entropy that convexifies the energy [51]. To see the relationship between the graph-matching energy and the expected log-likelihood function, we note that

$$\mathcal{E}_H = -\frac{1}{2}Tr[D^T S M S^T]. \tag{34}$$

In other words, the weighting matrix Q is replaced by the assignment matrix S.

One of the simplest ways of updating the assignment variables is to use the soft-max ansatz of Bridle [5]. This ensures that the assignment variables remain constrained to lie within the range $[0,1]$ by adopting the update rule

$$s_{a\alpha} \leftarrow \frac{\exp\left[-\frac{1}{T}\frac{\partial \mathcal{E}_H}{\partial s_{a\alpha}}\right]}{\sum_{\alpha' \in V_M}\exp\left[-\frac{1}{T}\frac{\partial \mathcal{E}_H}{\partial s_{a\alpha'}}\right]}. \tag{35}$$
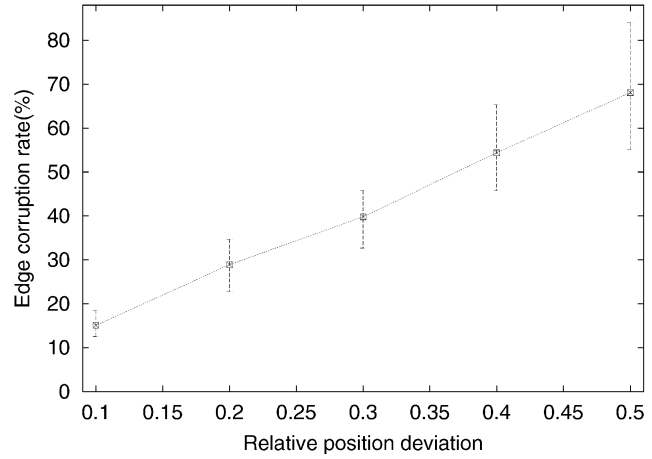
The temperature T is usually controlled using a slow exponential annealing schedule of the form suggested in [23]. For the quadratic graph-matching energy,

$$\frac{\partial \mathcal{E}_H}{\partial s_{b\beta}} = -\frac{1}{2}\sum_{a \in V_D}\sum_{\alpha \in V_M} D_{ab}M_{\alpha\beta}s_{a\alpha}. \tag{36}$$

More recently, Finch et al. [18] have developed a more sophisticated soft-assign graph-matching algorithm which revolves around optimizing the nonquadratic energy

$$\mathcal{E}_F = \sum_{a \in V_D} U_a, \tag{37}$$

where

$$U_a = \frac{\sum_{\alpha \in V_M} H_{a\alpha}\exp[-\mu H_{a_\alpha}]}{\sum_{\alpha \in V_M}\exp[-\mu H_{a_\alpha}]} \tag{38}$$

and

$$H_{a\alpha} = \sum_{b \in V_D}\sum_{\beta \in V_M} D_{ab}M_{\alpha\beta}(1 - s_{b\beta}). \tag{39}$$
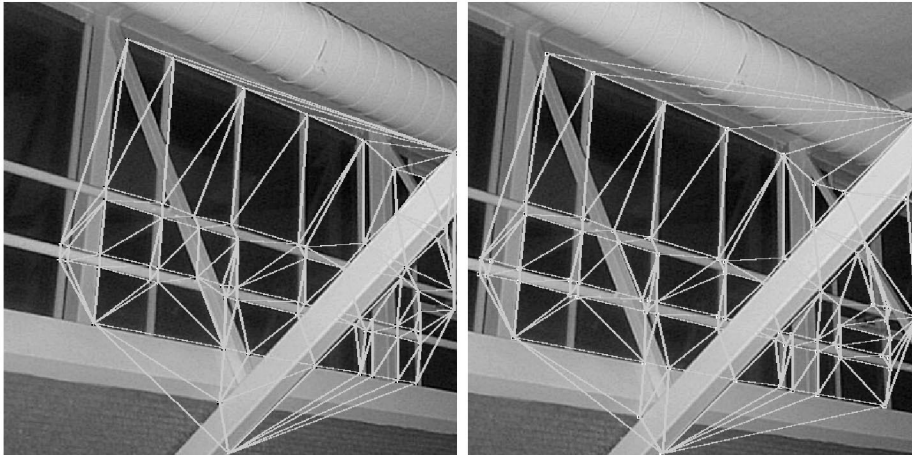

Fig. 5. Test images overlayed with Delaunay graphs.
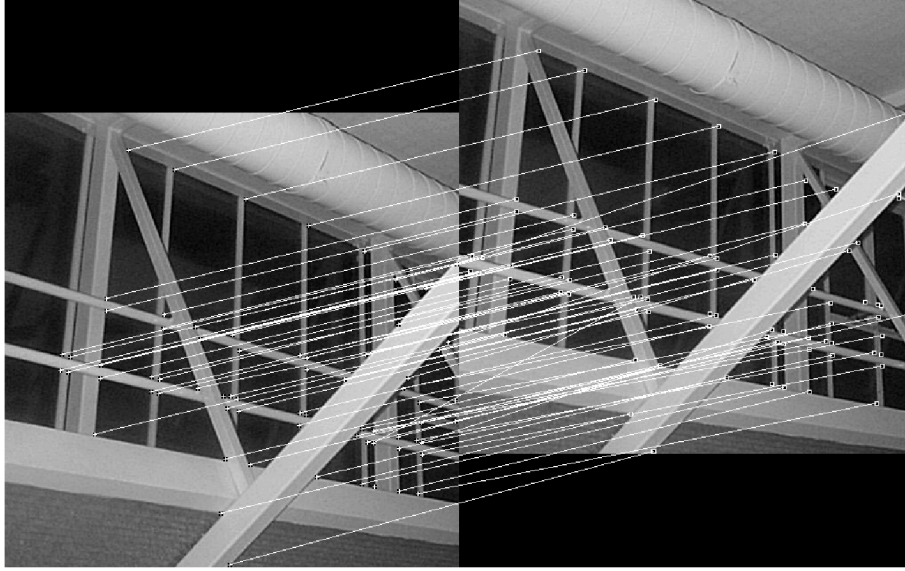
Fig. 6. Correspondences.

For this energy function, the partial derivative is given by

$$\frac{\partial \mathcal{E}_F}{\partial s_{b\beta}} = -\sum_{a\in V_D}\sum_{\alpha\in V_M} D_{ab} M_{\alpha\beta}\left[1-\mu\left(H_{a\alpha}-U_a\right)\right]Q_{a\alpha}. \quad (40)$$

We will use these two soft-assign graph matching methods for the purposes of experimental comparison. However, it is important to stress that the soft-assign update process adopted here is very simplistic and leaves considerable scope for further refinement. For instance, in [23] Sinkhorn matrices have been exploited to impose a permutation structure on the final solution.

5 EXPERIMENTS

In this section, we provide some experimental evaluation of the new graph-matching technique. There are two aspects to this study. We commence with a sensitivity study using synthetic data. The aim here is to evaluate how the new method performs under controlled structural corruption and to compare it with some alternatives reported elsewhere in the literature. The second part of the study evaluates the method on real-world data.

5.1 Sensitivity Study

Our sensitivity study is divided into two parts: First, we compare our method with some alternative methods for inexact-graph matching. These methods are capable of accommodating graphs of different size, but are not based on matrix factorization. Here, we investigate the effect of adding additional nodes to the graphs. The second class of methods used for comparison are those which rely on matrix-factorization techniques. These methods do not work when the graphs are of different size. Here, we keep the graphs of fixed equal size and investigate the effect of corrupting the pattern of edges.

5.1.1 Inexact Graph Matching

We commence by studying the effect of controlled structural error on the graphs being matched. The graphs used in our study are the Delaunay triangulations of randomly generated point-sets. The effects of structural error are simulated by deleting a predefined fraction of randomly selected nodes and retriangulating the remaining points.
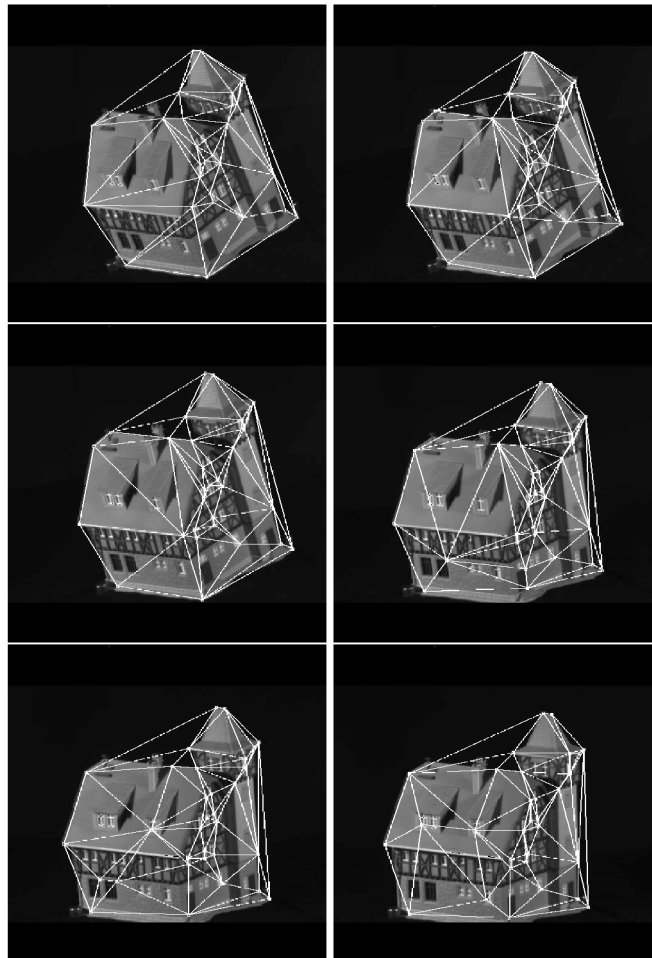

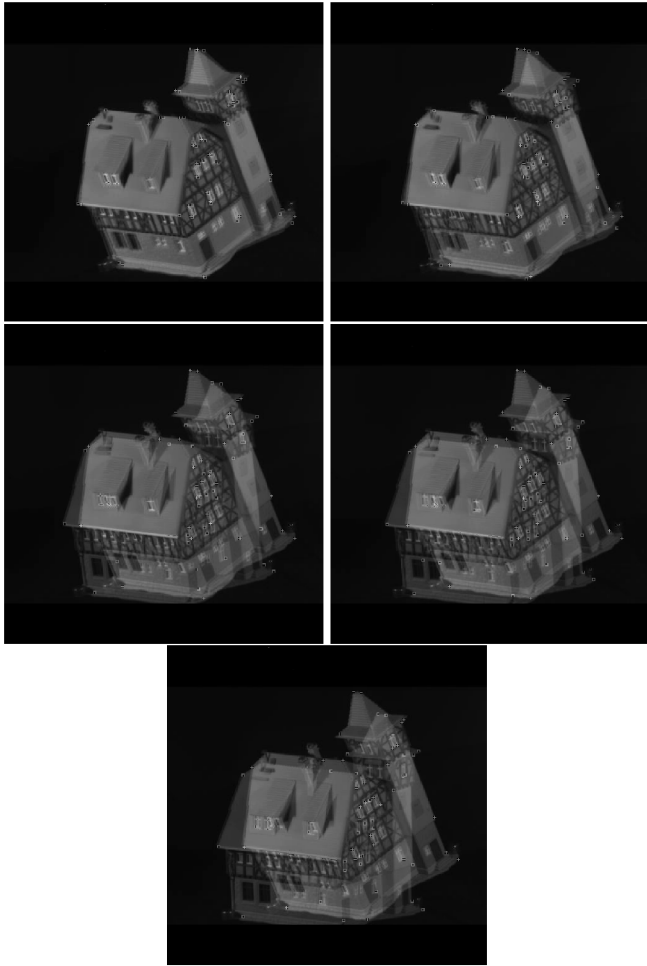Fig. 7. Delaunay graphs overlayed on the toy house images.

Fig. 8. Superimposed image pairs illustrating the differences in viewpoint.

We compare the performance of our new matching method with three alternatives. These are the dictionary-based relaxation scheme of Wilson and Hancock [49], the quadratic assignment method of Gold and Rangarajan [23], and the nonquadratic graduated assignment method of Finch et al. [19]. Fig. 1 compares the four algorithms. Here, we show the fraction of correct correspondences as a function of the fraction of nodes deleted from the graphs. The main feature to note is that the new graph matching method delivers performance that is intermediate between the discrete relaxation method and the nonlinear graduated assignment method. This is an interesting observation when we compare the computational overheads associated with the three methods.

To provide some indication of the iterative properties of our new algorithm, Fig. 2 shows the fraction of correct correspondences as a function of iteration number. The method takes approximately 15 iterations to converge. Moreover, there is significant improvement in each of the iterations. At this point, it is worth pointing out the relative complexities of the different methods investigated here. The SVD can be computed in order $|V|^3$ per iteration, where $|V|$ is the number of nodes in the graph. The Wilson and Hancock method, on the other hand, requires $|V|^2 C^3$ computations per iteration for exact graph matching and $4^C C |V|^2$ computations per iteration for inexact graph matching, where C is the average degree of nodes of the graph. The quadratic assignment method has complexity of order $|V|^3$ per iteration. The Wilson and Hancock method also converges in about 10-15 iterations, while the quadratic assignment method takes 100s of iterations. In other words, our new SVD-based method is both accurate and efficient.

5.1.2 Factorization Methods

In this section, we provide a comparison between two methods for weighted-graph-matching which share with
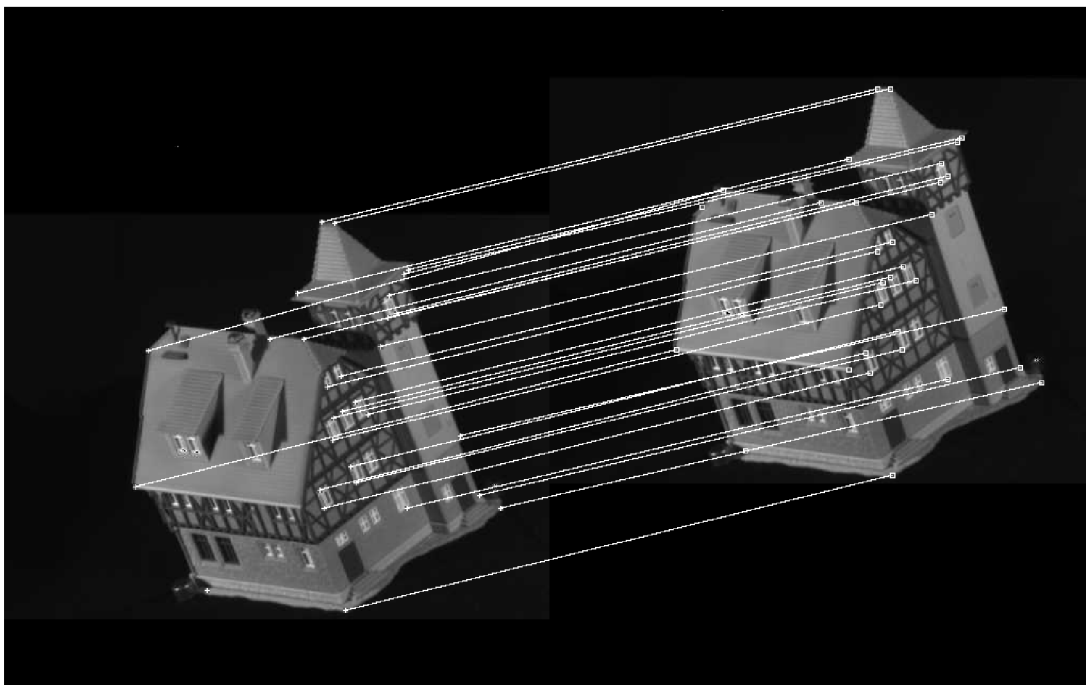

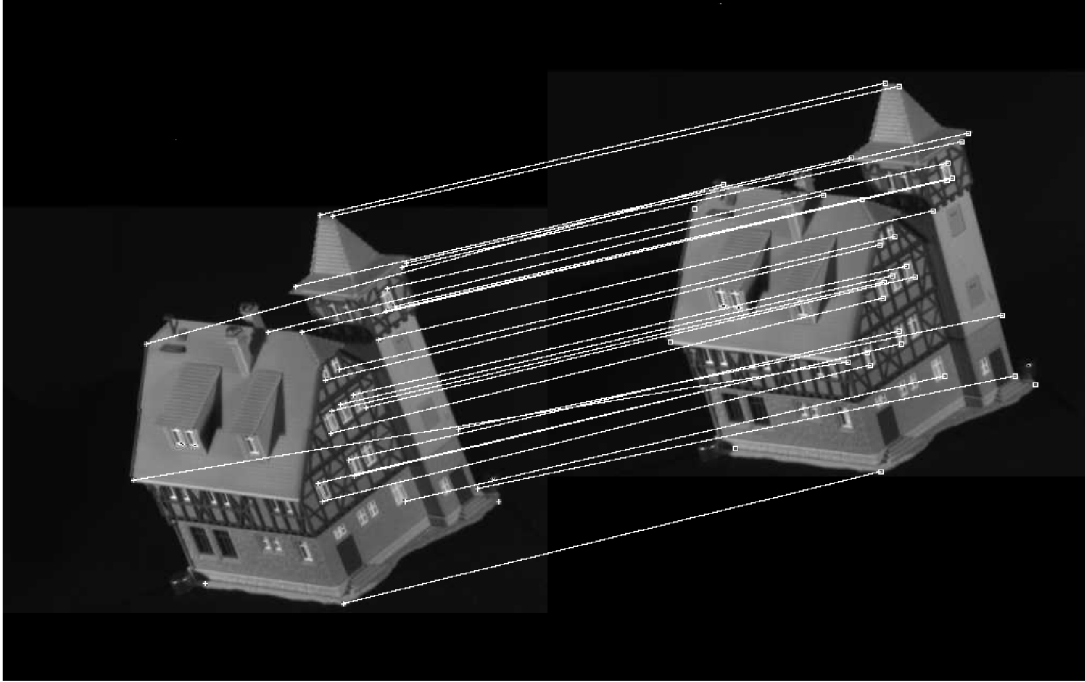Fig. 9. Correspondences between the first and the second images.

Fig. 10. Correspondences between the first and the third images.

our own method the feature of relying on matrix factorization. The methods selected for this comparison are:

- Umeyama's weighted-graph-matching method which seeks the permutation matrix $\mathcal{P}$ that minimizes quantity $J(\mathcal{P}) = ||\mathcal{P}M - D||$ [45]. The method performs the singular value decompositions $M = U_M \Delta_M U_M^T$ and $D = U_D \Delta_D U_D^T$, where the Us are orthogonal matrices and the Δs are diagonal matrices.

Once these factorizations have been performed, the required permutation matrix is $\mathcal{P} = U_D U_M^T$.

- Shapiro and Brady's [39] weighted-graph-matching method which uses the modal structure of the two-weighted adjacency matrices D and M. The modal structure of the two adjacency graphs is obtained by solving the eigenvalue equation $D\phi_l^D = \lambda_l \phi_l^D$, where λ_l is the lth eigenvalue of the adjacency matrix D and ϕ_l^D is the corresponding
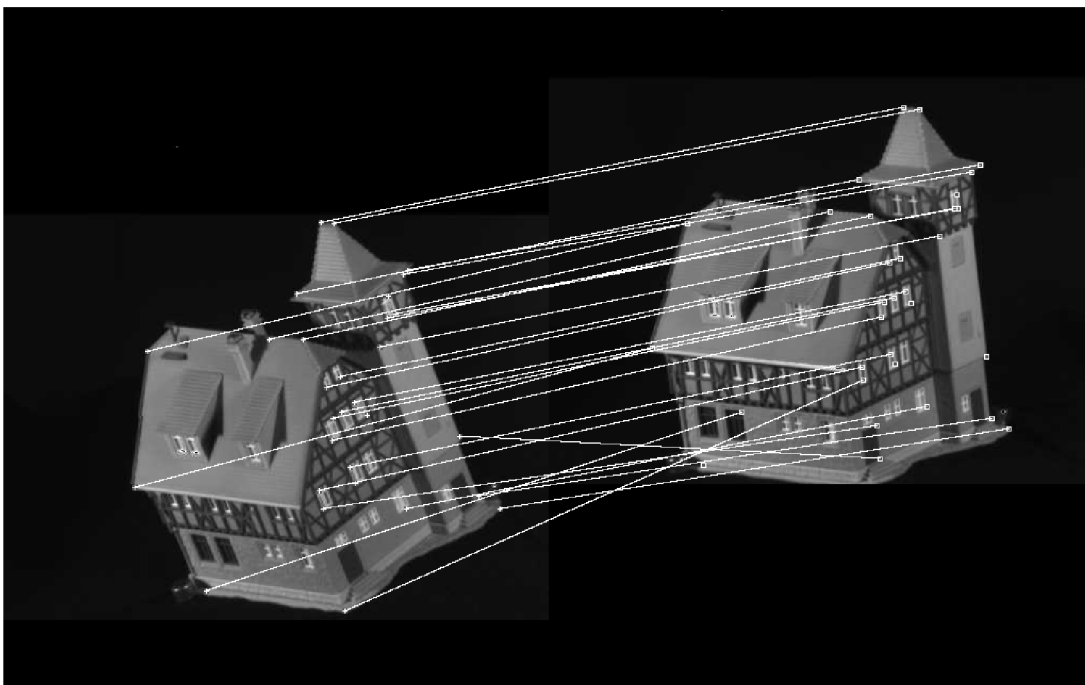

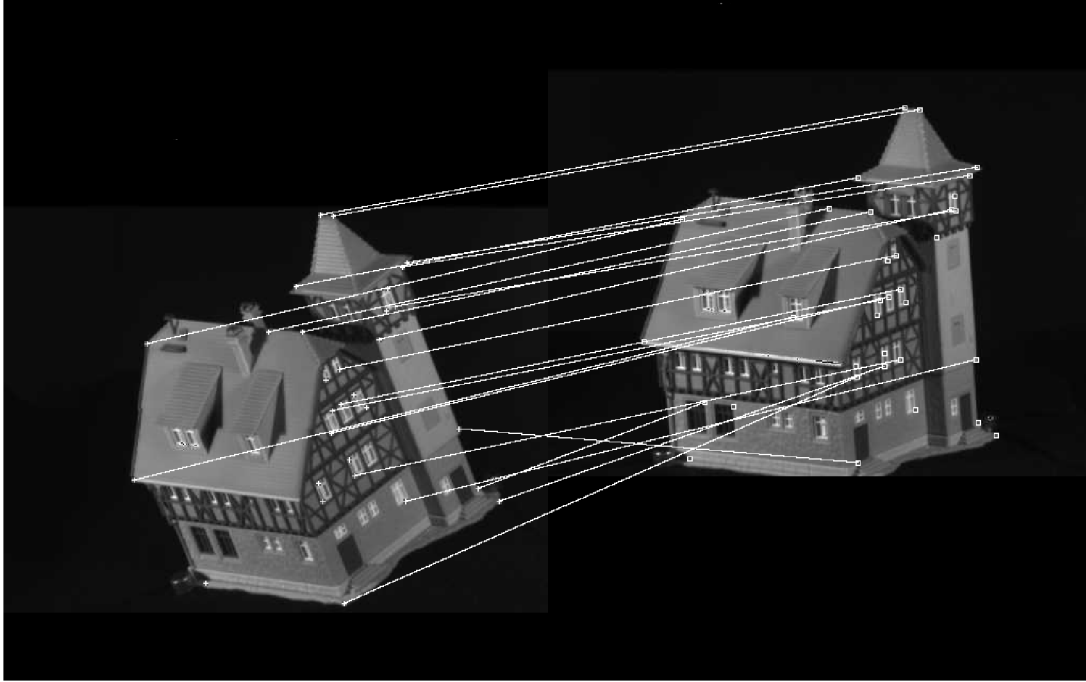Fig. 11. Correspondences between the first and the fourth images.

Fig. 12. Correspondences between the first and the fifth images.

eigenvector. The eigenvectors are ordered according to the size of the associated eigenvalues and are used as the columns of the modal matrix $\Phi_D = \left(\phi_1^D, \phi_2^D, \phi_3^D, ...\right)$. This procedure is repeated to construct a second modal matrix Φ_M for the model-graph adjacency matrix M. The column index of these two modal matrices refers to the order of the eigenvalues while the row-index is the index of the nodes in the graphs. Shapiro and Brady find correspondences by locating pairs of rows which have minimum distance, i.e.,

$$s_{a,\alpha} = \begin{cases} 1 & \text{if } \alpha = \arg\min_{\alpha'} \sum_{l=1}^{N} ||\Phi_D(a,l) - \Phi_M(\alpha',l)||^2 \\ 0 & \text{otherwise.} \end{cases}$$
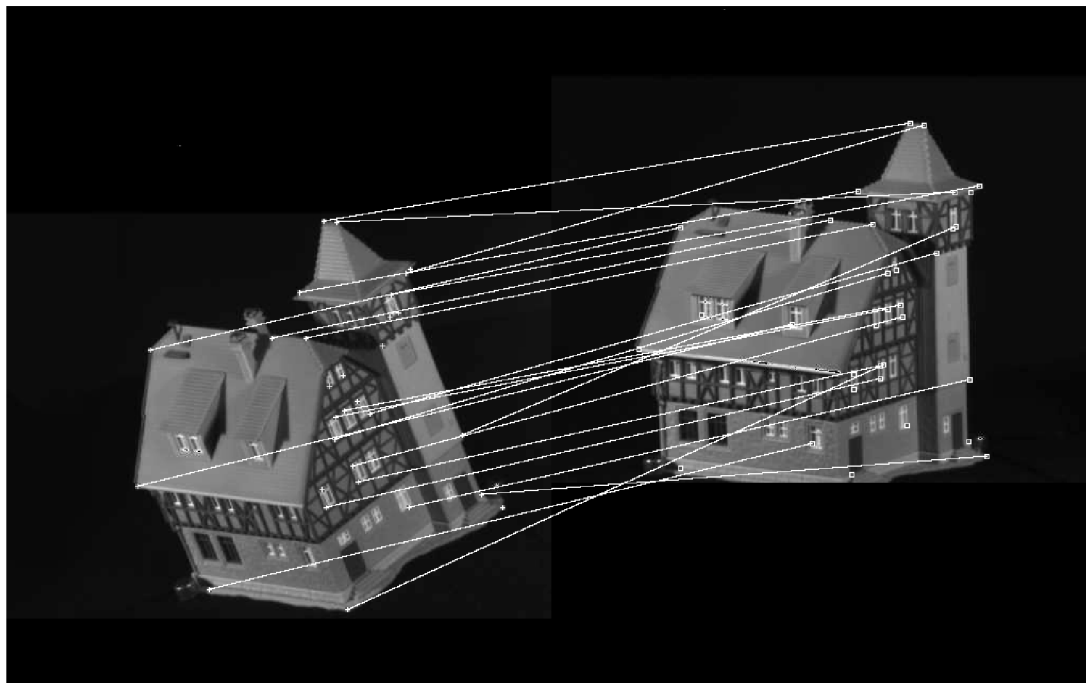
$$(41)$$

Fig. 13. Correspondences between the first and the sixth images.

TABLE 1
Summary of Experimental Results for
the House Sequence Images

Images	Corners	Correct correspondences	False correspondences	No correspondences
house 1	30	–	–	–
house 2	32	29	0	1
house 3	32	28	1	1
house 4	31	23	5	2
house 5	30	11	10	9
house 6	30	5	16	9

These two methods rely on weighted adjacency matrices rather than the binary ones defined earlier. To conduct our experiments, we have generated random 2D point-sets. We use the positions of these points to generate the weights of the adjacency matrix. Suppose that $\vec{x}_\alpha^M$ and $\vec{x}_\beta^M$ represent the coordinate vectors associated with the nodes indexed a and b. The weight associated with the edge connecting the nodes is

$$M_{\alpha\beta} = \exp\left[-k||\vec{x}_\alpha^M - \vec{x}_\beta^M||^2\right]. \tag{42}$$

These two methods are not effective when the graphs under study contain different numbers of nodes. To compare with our method, therefore, we have kept the number of points fixed and have added Gaussian errors to the point positions. The parameter of the noise process is the standard deviation of the positional jitter. In our experiments, we express this parameter as a fraction of the average minimum distance between points (the relative standard deviation). It is important to stress that the

methods compared here use different representations of the arrangement of the points. The Shapiro and Brady and Umeyama methods use the weighted adjacency matrix. Our method, on the other hand, uses a binary-adjacency matrix to represent the Delaunay triangulation of the points.

In Fig. 3, we show the fraction of correct correspondences as a function of the relative standard deviation for our new method (bold solid curve), Umeyama's [45] method (faint solid curve) and the method of Shapiro and Brady [39] (dash-dotted curve). The main feature to note is that our method outperforms the two alternatives. There is little to distinguish the performance of the Shapiro and Brady [39] and Umeyama [45] methods. Both fail abruptly once the relative standard deviation exceeds 0.2, i.e., the noise standard deviation is greater than 20 percent of the average closest point distance. Our method, on the other hand, degrades almost linearly with the noise standard deviation. However, it must be stressed that the results are not completely comparable. In the case of Shapiro and Brady, and Umeyama [45], we are measuring the sensitivity of the method to noise on the entries of the weighted-adjacency matrices. In the case of our method, we are measuring the sensitivity of the method to errors in the edge-sets of the graphs used for matching.

To show that the point-jitter does indeed result in significantly different adjacency matrices, in Fig. 4 we show the fraction of edge differences as function of the relative standard deviation of the positional jitter. The fraction of edge errors in the unweighted adjacency graphs is defined to be
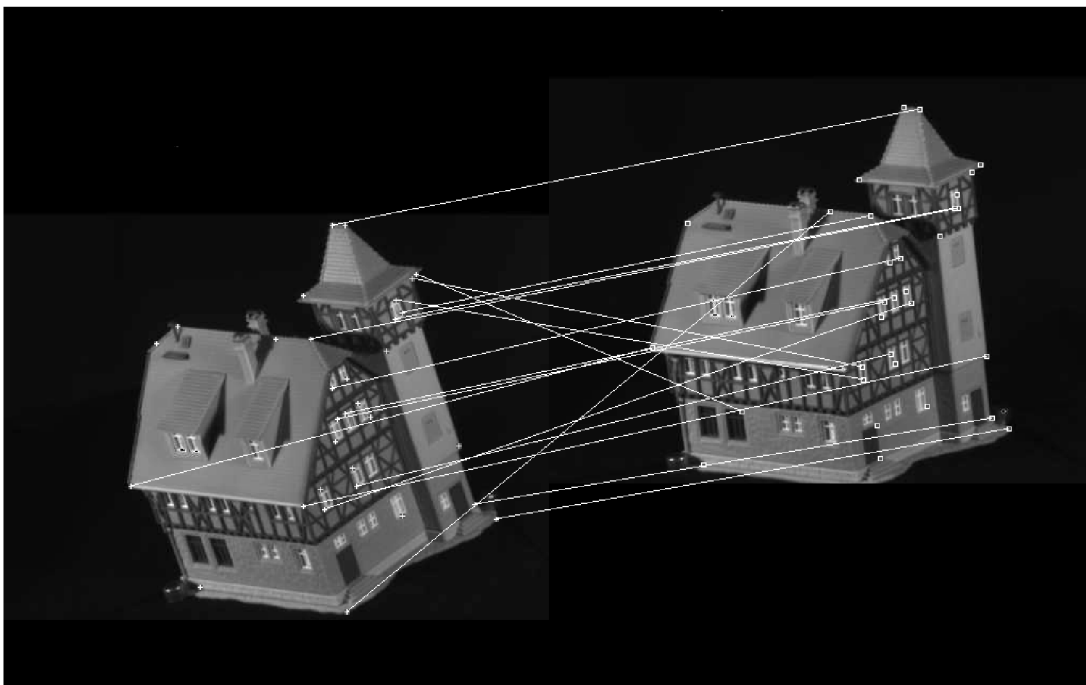

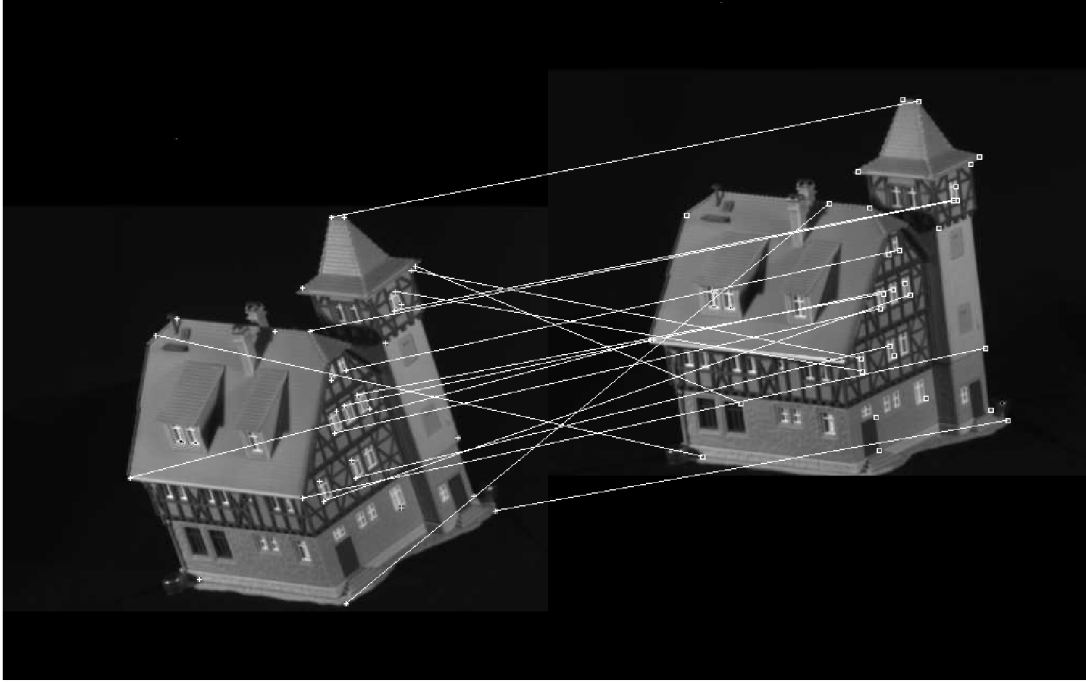Fig. 14. Correspondences from the Umeyama algorithm.

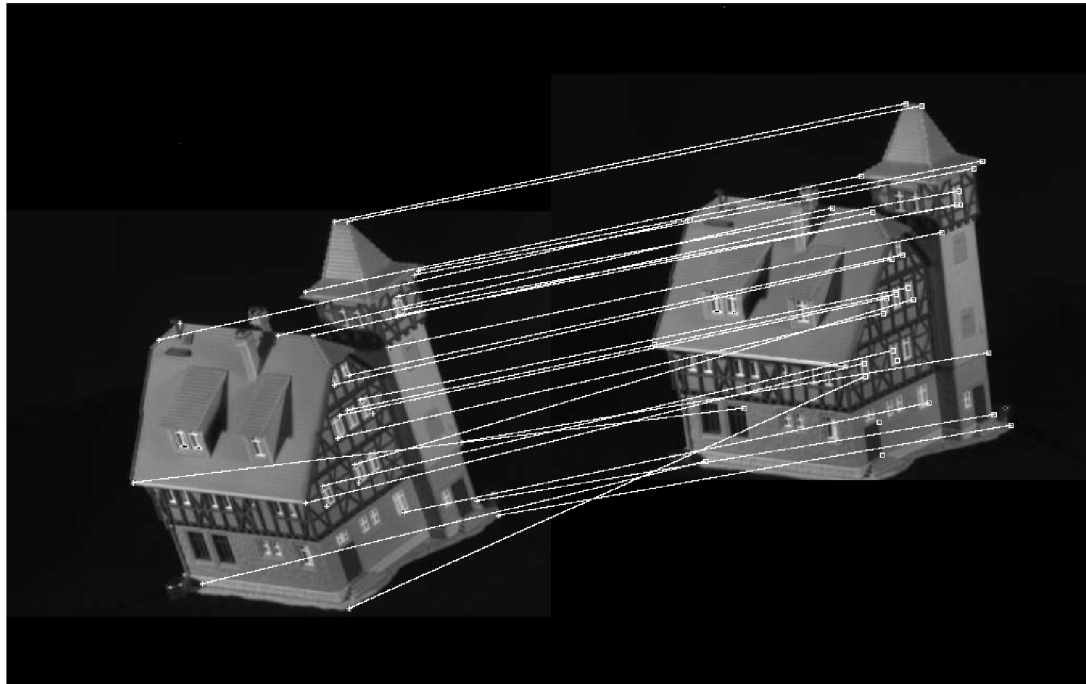Fig. 15. Correspondences from the Shapiro algorithm.

Fig. 16. Correspondences from the unweighted variant of our algorithm.

$$F = \frac{\sum_{a=1}^{N} \sum_{b=1}^{N} |D_{ab} - \sum_{c=1}^{N} S_{ac}M_{cb}|}{\sum_{a=1}^{N} \sum_{b=1}^{N} |M_{ab}|}, \qquad (43)$$

where N is the number of nodes being matched. From Fig. 4 it is clear, that the fraction of edge errors is 50 percent when the relative standard deviation of the positional jitter is 0.3. In other words, our method is finding 70 percent of the correct correspondences, even when 50 percent of the entries in the data-graph adjacency matrix are in error.

Finally, we illustrate the results obtained when we apply our method to the weighted-adjacency matrix rather than the binary-adjacency matrix. The dot-dashed curve in Fig. 3 shows the fraction of correct correspondences as a function of the relative standard deviation of the point-position jitter. The method performs considerably better than the Shapiro and Brady, and Umeyama methods. However, there is little to distinguish its performance from that obtained with the binary-adjacency matrix.
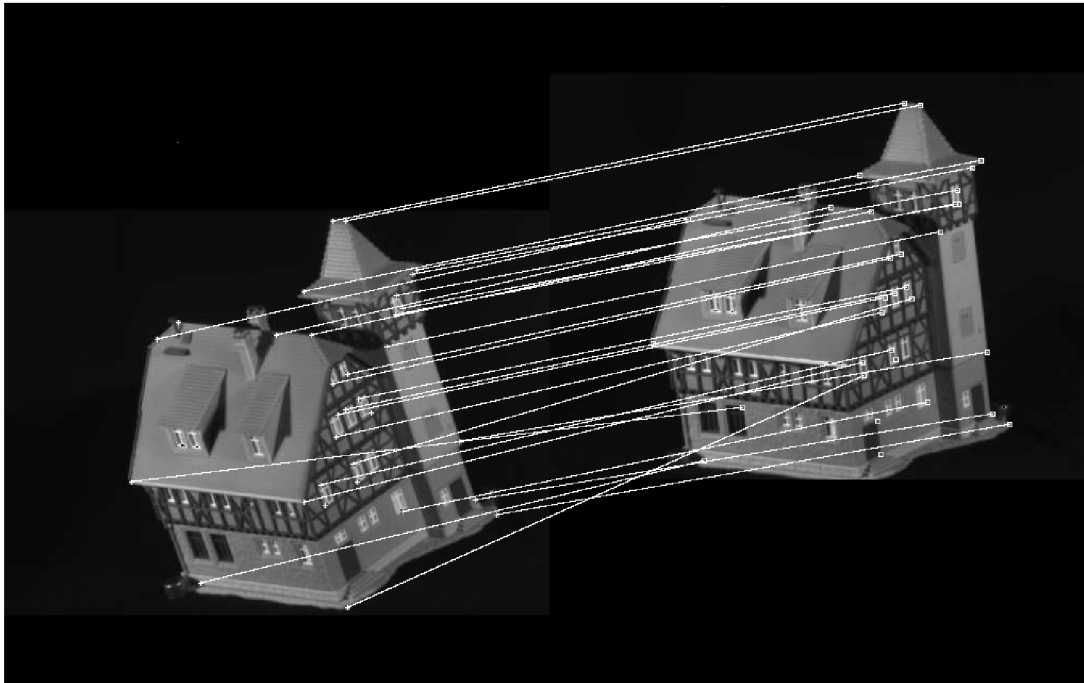
Fig. 17. Correspondences from the weighted variant of our algorithm.

5.2 Real-World Data

We commence our real-world evaluation of the graph-matching method on images of indoor scenes. Here, we are concerned with matching the Delaunay triangulations of corner-features. We use the corner detector recently reported by Luo et al. [29] to extract point features. Fig. 5 shows two examples of the indoor images used in our study. Superimposed on the images are the detected corners and their associated Delaunay triangulations. The two images are taken from different viewpoints. There is rotation, scaling, and perspective distortion present. Moreover, several of the objects in the scene are at different depths and move relative to one another. As a result, there are significant structural differences in the two Delaunay graphs. Fig. 6 shows the correspondences between the corners as lines between the two images. After checking by hand, the fraction of correct correspondences is 77 percent.

We repeat this set of experiments using images taken from the CMU/VASC model-house sequence. The images used in our study are shown in Fig. 7 and correspond to different camera viewing directions. The detected corner features and their Delaunay triangulations are overlayed on the images. There are clearly significant structural differences in the graphs. By superimposing the first image on the subsequent images in the sequence, Fig. 8 illustrates the differences in viewing angle. Figs. 9, 10, 11, 12, and 13 show the results obtained when we match the first image to each of the subsequent images in the sequence. The results are summarized in Table 1. Here, we list the number of detected corners in the images being matched, the number of corners that are in correct correspondence, the number of corners that are in error, and the number of corners for which there are no correspondences (i.e., there is no row and column maximum). The method breaks down after the fourth image in the sequence.

To provide some comparison, we have selected a pair of images which contain the same number of corner points (image 2 and image 4). Although the number of corners is the same, there are differences in the both identities of the detected points and their structural arrangement. For these images, we compare the matches returned by the unweighted and weighted versions of our algorithm (referred to as Luo), the method of Umeyama and the method of Shapiro and Brady. The results are shown in Figs. 14, 15, 16, and 17 and the numbers of correct matches are summarized in Table 2. From these results, it is clear that the new method returns considerably better matches.

The final real-world example is furnished by matching features in aerial infrared images. The main structure in these images is a road network. The features used in our matching experiments are the junctions in the road network. This data was used in the recent studies by both Wilson and Hancock [49] and Finch et al. [20]. Fig. 18 shows the correspondences. Here, manual checking reveals that 58 percent of the correspondences are correct. By contrast, the fractions of correct correspondences from the two recent studies are respectively 34 percent and 47 percent.

TABLE 2
Summary of the Comparison of the Three Matching Algorithms

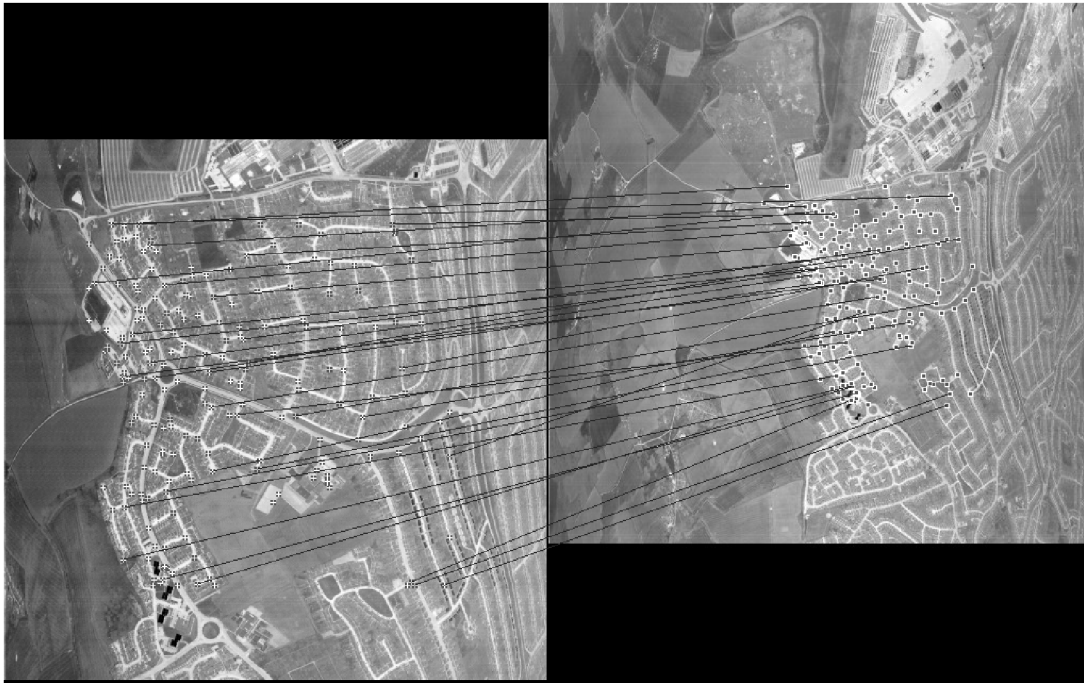| Methods	Correct correspondences	False correspondences	No correspondences
Luo(Weighted)	22	6	3
Luo(Unweighted)	22	6	3
Umeyama	6	11	14
Shapiro	6	11	14

Fig. 18. Correspondences for the aerial infrared image pair.

6 CONCLUSIONS

Our main contributions in this paper are twofold: First, we have cast the problem of graph matching into a maximum-likelihood framework by constructing a mixture model over the set of hidden correspondences and adopting a Bernoulli model for the distribution of edge-matching errors. Second, we have used the apparatus of the EM algorithm to show how the problem of estimating the correspondence indicators may be cast into a compact matrix setting. This allows us to use singular value decomposition to estimate the correspondence indicators in the M-step. The result is an efficient algorithm that can be used to accurately match inexact graphs under considerable levels of structural corruption.

When viewed from the perspective of recent work on matrix-based graph matching, the important contribution of this paper is to show how point-sets of different sizes can be matched using singular value decomposition.

REFERENCES

[1] A.P. Ambler, H.G. Barrow, C.M. Brown, R.M. Burstall, and R.J. Popplestone, "A Versatile Computer-Controlled Assembly System," *Proc. Third Int'l Conf. Artifical Intelligence,* pp. 298-307, 1973.

[2] H.G. Barrow and R.J. Popplestone, "Relational Descriptions in Picture Processing," *Machine Intelligence,* vol. 5, pp. 377-396, 1971.

[3] R.E. Blake, *Development of an Incremental Graph Matching Device,* vol. 30, of NATO ASI series, pp. 355-366. Berlin: Spinger-Verlag, 1987.

[4] K. Boyer and A. Kak, "Structural Stereopsis for 3D Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 10, no. 2, pp. 144-166, Feb. 1988.

[5] J.S. Bridle, "Training Stochastic Model Recognition Algorithms can Lead to Maximum Mutual Information Estimation of Parameters," *Proc. Neural Information Processing Systems,* pp. 211-217, 1990.

[6] H. Bunke, "Error Correcting Graph Matching: On the Influence of the Underlying Cost Function," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 21, no. 9, pp. 917-922, Sept. 1999.

[7] H. Bunke and G. Allermann, "Inexact Graph Matching for Structural Pattern Recognition," *Pattern Recognition Letters,* vol. 1, pp. 245-253, 1983.

[8] H. Bunke and B.T. Messmer, "Recent Advances in Graph Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 11, pp. 169-203, 1997.

[9] H. Bunke and K. Shearer, "A Graph Distance Metric Based on the Maximal Common Subgraph," *Pattern Recognition Letters,* vol. 19, pp. 255-259, 1998.

[10] C. Bishop, *Neural Networks for Pattern Recognition.* Oxford: Clarendon Press, 1995.

[11] W.J. Christmas, J. Kittler, and M. Petrou, "Structural Matching in Computer Vision Using Probabilistic Relaxation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 8, pp. 749-764, Aug. 1995.

[12] F.R.K. Chung, *Spectral Graph Theory,* Am. Math. Soc., ed., CBMS Series 92. 1997,

[13] A.D.J. Cross and E.R. Hancock, "Graph Matching with a Dual-Step EM Algorithm," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 11, pp. 1236-1253, Nov. 1998.

[14] A.D.J. Cross, R.C. Wilson, and E.R. Hancock, "Inexact Graph Matching with Genetic Search," *Pattern Recognition,* vol. 30, no. 6, pp. 953-970, 1997.

[15] B.D. Ripley, *Pattern Recognition and Neural Networks.* New York: Cambridge Univ. Press, 1996.

[16] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum-Likelihood from Incomplete Data via the EM Algorithm," *J. Royal Statistical Soc. Series B,* vol. 39, pp. 1-38, 1977.

[17] M.A. Eshera and K.S. Fu, "An Image Understanding System Using Attributed Symbolic Representation and Inexact Graph-Matching," *J. Assoc. Computing Machinery,* vol. 8, no. 5, pp. 604-618, 1986.

[18] A.M. Finch, R.C. Wilson, and E.R. Hancock, "Softening Discrete Relaxation," *Advances in Neural Information Processing Systems,* M. Jordan, M. Mozer, and T. Petsche, eds., vol. 9, pp. 438-444, 1997.

[19] A.M. Finch, R.C. Wilson, and E.R. Hancock, "An Energy Function and Continuous Edit Process for Graph Matching," *Neural Computation,* vol. 10, no. 7, pp. 1873-1894, 1998.

[20] A.M. Finch, R.C. Wilson, and E.R. Hancock, "Symbolic Graph Matching with the EM Algorithm," *Pattern Recognition,* vol. 31, no. 11, pp. 1777-1790, 1998.

[21] M. Fischler and R. Elschlager, "The Representation and Matching of Pictorical Structures," *IEEE Trans. Computers,* vol. 22, no. 1, pp. 67-92, Jan. 1973.

[22] G.J. McLachlan and K.E. Basford, *Mixture Models: Inference and Applications to Clustering.* New York: Marcel Dekker, 1988.

[23] S. Gold and A. Rangarajan, "A Graduated Assignment Algorithm for Graph Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 18, no. 4, pp. 377-388, Apr. 1996.

[24] L. Herault, R. Horaud, F. Veillon, and J.J. Niez, "Symbolic Image Matching by Simulated Annealing," *Proc. British Machine Vision Conf.,* pp. 319-324, 1990.

[25] T. Hoffmann and J.M. Buhmann, "Pairwise Data Clustering with Deterministic Annealing," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 1, pp. 1-14, Jan. 1997.

[26] R. Horaud and T. Skordas, "Stereo Correspondence through Feature Grouping and Maximal Cliques," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 11, no. 11, pp. 1168-1180, Nov. 1989.

[27] R. Horaud and H. Sossa, "Polyhedral Object Recognition by Indexing," *Pattern Recognition,* vol. 28, no. 12, pp. 1855-1870, 1995.

[28] J. Hornegger and H. Niemann, "Statistical Learning, Localization, and Identification of Objects," *Proc. Int'l Conf. Computer Vision,* pp. 914-919, 1995.

[29] B. Luo, A.D.J. Cross, and E.R. Hancock, "Corner Detection via Topographic Analysis of Vector Potential," *Pattern Recognition Letters,* vol. 20, pp. 635-650, 1999.

[30] S. Moss and E.R. Hancock, "Registering Incomplete Radar Images with the EM Algorithm," *Image and Vision Computing,* vol. 15, pp. 637-648, 1997.

[31] M. Pelillo, K. Siddiqi, and S. W. Zucker, "Matching Hierarchical Structures Using Association Graphs," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 21, no. 11, pp. 1105-1120, Nov. 1999.

[32] M. Pellilo, "Replicator Equations, Maximal Cliques, and Graph Isomorphism," *Neural Computation,* vol. 11, no. 8, pp. 1933-1955, 1999.

[33] C. Peterson and B Soderberg, "A New Method for Mapping Optimisation Problems," *Int'l J. Neural Systems,* vol. 1, pp. 2-33, 1989.

[34] A. Rangarajan, S. Gold, and E. Mjolsness, "A Novel Optimizing Network Architecture with Applications," *Neural Computation,* vol. 8, pp. 1041-1060, 1996.

[35] A. Sanfeliu and K. S. Fu, "A Distance Measure between Attributed Relational Graphs for Pattern Recognition," *IEEE Trans. Systems, Man, and Cybernetics,* vol. 13, no. 3, pp. 353-362, May 1983.

[36] G.L. Scott and H.C. Longuet-Higgins, "An Algorithm for Associating the Features of 2 Images," *Proc. Royal Soc. London Series B,* vol. 244, no. 1309, pp. 21-26, 1991.

[37] K. Sengupta and K.L. Boyer, "Modelbase Partitioning Using Property Matrix Spectra," *Computer Vision and Image Understanding,* vol. 70, no. 2, pp. 177-196, 1998.

[38] L.G. Shapiro and R.M. Haralick, "A Metric for Comparing Relational Descriptions," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 7, no. 1, pp. 90-94, Jan. 1985.

[39] L.S. Shapiro and J.M. Brady, "Feature-Based Correspondence—An Eigenvector Approach," *Image and Vision Computing,* vol. 10, pp. 283-288, 1992.

[40] A. Shokoufandeh, S.J. Dickinson, K. Siddiqi, and S.W. Zucker, "Indexing Using a Spectral Encoding of Topological Structure," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 491-497, 1999.

[41] P.D. Simic, "Constrained Nets for Graph Matching and other Quadratic Assignment Problems," *Neural Computation,* vol. 3, pp. 268-281, 1991.

[42] P.N. Suganthan, E.K. Teoh, and D.P. Mital, "Pattern-Recognition by Graph Matching Using the Potts MFT Neural Networks," *Pattern Recognition,* vol. 28, no. 7, pp. 997-1009, 1995.

[43] S. Tirthapura, D. Sharvit, P. Klein, and B.B. Kimia, "Indexing Based on Edit-Distance Matching of Shape Graphs," *Multimedia Storage and Archiving Systems III,* vol. 3527, pp. 25-36, 1998.

[44] J.R. Ullman, "An Algorithm for Subgraph Isomorphism," *J. ACM,* vol. 23, no. 1, pp. 31-42, Jan. 1976.

[45] S. Umeyama, "An Eigen Decomposition Approach to Weighted Graph Matching Problems," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 10, pp. 695-703, 1988.

[46] J. Utans, "Mixture Models and EM Algorithms for Object Recognition within Compositional Hierarchies," ICSI Berkeley Technical Report TR-93-004, 1993.

[47] W.M. Wells, "Map Model Matching," *IEEE Conf. Computer Vision and Pattern Recognition,* pp. 486-492, 1991.

[48] M.L. Williams, R.C. Wilson, and E.R. Hancock, "Deterministic Search for Relational Graph Matching," *Pattern Recognition,* vol. 32, no. 7, pp. 1255-1271, 1999.

[49] R.C. Wilson and E.R. Hancock, "Structural Matching by Discrete Relaxation," *IEEE Pattern Analysis and Machine Intelligence,* vol. 19, no. 6, pp. 634-648, June 1997.

[50] A.K.C. Wong and M. You, "Entropy and Distance of Random Graphs with Application to Structural Pattern Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 7, no. 5, pp. 509-609, Sept. 1985.

[51] A.L. Yuille and J.J. Kosowsky, "Statistical Physics Algorithms that Converge," *Neural Computation,* vol. 6, pp. 341-356, 1994.

[52] A.L. Yuille, P. Stolorz, and J. Utans, "Statistical Physics, Mixtures of Distributions, and the EM Algorithm," *Neural Computation,* vol. 6, pp. 334-340, 1994.

Bin Luo received the BEng degree in electronics and the MEng degree in computer science from Anhui University of China in 1984 and 1991, respectively. He was a lecturer from 1992 to 1993 and an associate professor from 1993 to 1998 at Anhui University. He has been a professor at the same university since 1998. Presently, he is a PhD student and a research associate at the University of York, United Kingdom. His current research interests include image and graph matching, statistical pattern recognition, and image feature extraction.

Edwin R. Hancock studied physics as an undergraduate at the University of Durham and graduated with honors in 1977. He remained at Durham to complete the PhD degree in the area of high-energy physics in 1981. Following this, he worked for 10 years as a researcher in the fields of high-energy nuclear physics and pattern recognition at the Rutherford-Appleton Laboratory (now the Central Research Laboratory of the Research Councils). During this period, he also held adjunct teaching posts at the University of Surrey and the Open University. In 1991, he moved to the University of York as a lecturer in the Department of Computer Science. He was promoted to senior lecturer in 1997 and to reader in 1998. In 1998, he was appointed to chair in computer vision. He now leads a group of some 15 faculty, research staff, and PhD students working in the areas of computer vision and pattern recognition. His main research interests are in the use of optimization and probabilistic methods for high and intermediate level vision. He is also interested in the methodology of structural and statistical pattern recognition. He is currently working on graph matching, shape-from-X, image databases, and statistical learning theory. His work has found applications in areas such as radar terrain analysis, seismic section analysis, remote sensing, and medical imaging. Professor Hancock has published more than 60 journal papers and 200 refereed conference publications. He was awarded the Pattern Recognition Society medal in 1991 for the best paper to be published in the journal *Pattern Recognition*. The journal also awarded him an outstanding paper award in 1997. He is a member of the editorial boards of the journals *IEEE Transactions on Pattern Analysis and Machine Intelligence* and *Pattern Recognition*. He has also been a guest editor for special editions of the journals *Image and Vision Computing and Pattern Recognition* and he is currently a guest editor of a special edition of *IEEE Transactions on Pattern Analysis and Machine Intelligence* devoted to energy minimization methods in computer vision. He has been on the program committees for numerous national and international meetings. In 1997, he established a new series of international meetings on energy minimization methods in computer vision and pattern recognition. In 2000, he became a fellow of the International Association for Pattern Recognition (IAPR).

▷ **For more information on this or any other computing topic, please visit our Digital Library at** http://computer.org/publications/dlib.