A Probabilistic Approach to Spectral Graph Matching

Amir Egozi, *Student Member*, *IEEE*, Yosi Keller, and Hugo Guterman, *Member*, *IEEE*

Abstract—Spectral Matching (SM) is a computationally efficient approach to approximate the solution of pairwise matching problems that are *np*-hard. In this paper, we present a probabilistic interpretation of spectral matching schemes and derive a novel Probabilistic Matching (PM) scheme that is shown to outperform previous approaches. We show that spectral matching can be interpreted as a Maximum Likelihood (ML) estimate of the assignment probabilities and that the Graduated Assignment (GA) algorithm can be cast as a Maximum a Posteriori (MAP) estimator. Based on this analysis, we derive a ranking scheme for spectral matchings based on their reliability, and propose a novel iterative probabilistic matching algorithm that relaxes some of the implicit assumptions used in prior works. We experimentally show our approaches to outperform previous schemes when applied to exhaustive synthetic tests as well as the analysis of real image sequences.

Index Terms—Graphs, spectral matching, probabilistic matching, point matching

1 Introduction

Graph matching is a known problem in image and data analysis that relates to a gamut of research topics. Those include the registration of sets of points in $\mathbb{R}^d$ [21], [33], object recognition [2], shape retrieval [13], symmetry analysis [18], [6], and channel decoding.

The graph matching problem is depicted in Fig. 1, where, given a set of points, graph nodes can represent the points, while the graph edges encode their distances. Hence, graph matching paves the way for recovering point correspondences. Given two sets of points in $\mathbb{R}^d$ such that $S_1 = \{\mathbf{x}_i\}_{i=1}^{n_1}$ and $S_2 = \{\mathbf{y}_i\}_{i=1}^{n_2}$, the assignment problem is to recover the set of assignments

$$C \triangleq \{c_{ii'}\}_{i=1}^{n} = \{\mathbf{x}_i, \mathbf{y}_{i'}\}_{i=1}^{n}, \quad n \leq \min(n_1, n_2). \tag{1}$$

It is common to represent the correspondence by an *assignment matrix* $\mathbf{Z} \in \{0,1\}^{n_1 \times n_2}$ such that $z_{i,i'} = 1$ implies that $\mathbf{x}_i$ corresponds to $\mathbf{y}_{i'}$ and its row-wise vectorized replica $\mathbf{z} \in \{0,1\}^{n_1 n_2}$. The matching problem is defined by an unary affinity matrix $\mathbf{A} \in \mathbb{R}^{n_1 \times n_2}$ such that

$$a_{i,i'} = \Omega_1(c_{ii'}) \tag{2}$$

is the affinity of matching $\mathbf{x}_i$ to $\mathbf{y}_{i'}$ and Ω_1 is a unary affinity measure.

- *A. Egozi and H. Guterman are with the Department of Electrical Engineering, Ben Gurion University, Israel. E-mail: {agozi, hugo}@ee.bgu.ac.il.*
- *Y. Keller is with the Faculty of Engineering, Bar Ilan University, Israel. E-mail: yosi.keller@gmail.com.*

Manuscript received 22 Aug. 2010; revised 1 Oct. 2011; accepted 22 Jan. 2012; published online 13 Feb. 2012.
Recommended for acceptance by M. Meila.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2010-08-0648.
Digital Object Identifier no. 10.1109/TPAMI.2012.51.

The optimal assignment is thus given by

$$\mathbf{z}^* = \arg\max_{\mathbf{z}} \sum_{z_k=1} a_k = \arg\max_{\mathbf{z}} (\mathbf{z}^T \mathbf{a}), \quad \mathbf{z} \in \{0,1\}^{n_1 n_2} \quad s.t.\ \mathbf{Z}\underline{\mathbf{1}} \leq \underline{\mathbf{1}} \quad \text{and} \quad \mathbf{Z}^T\underline{\mathbf{1}} \leq \underline{\mathbf{1}}, \tag{3}$$

where $\mathbf{a} \in \mathbb{R}^{n_1 n_2}$ is a row-wise vectorized replica of $\mathbf{A}$ and $\mathbf{z}^*$ is a row-wise reshaped replica of $\mathbf{Z}^* \in \{0,1\}^{n_1 \times n_2}$.

The constraints in (3) imply that a point $\mathbf{x}_i \in S_1$ can only be matched to a single point in S_2 or not matched at all. The same applies to any point $\mathbf{y}_i \in S_2$. Equation (3) can be optimally solved by the Hungarian algorithm in polynomial time [25], binary linear programming [26], or approximated by Dynamic Programming [11].

In *pairwise* assignments we are given a pairwise affinity measure Ω_2 that scores the *joint* matching of $c_{ii'}$ and $c_{jj'}$:

$$\Omega_2 = \Omega_2(c_{ii'}, c_{jj'}),$$

implying that $\mathbf{x}_i$ is matched to $\mathbf{y}_{i'}$ and $\mathbf{x}_j$ to $\mathbf{y}_{j'}$. Pairwise affinities can encode geometric properties such as *local isometry*:

$$\Omega_2(c_{ii'}, c_{jj'}) = \exp\left\{-\frac{1}{\varepsilon^2}\left(\|\mathbf{x}_i - \mathbf{x}_j\|_2 - \|\mathbf{y}_{i'} - \mathbf{y}_{j'}\|_2\right)^2\right\}, \tag{4}$$

where $\varepsilon > 0$ is the kernel bandwidth. In contrast, unary affinities (2) encode vertex-to-vertex similarities such as correlations and differences of local descriptors (SIFT [22], Shape Context [1]).

The pairwise affinity matrix $\mathbf{A} \in \mathbb{R}^{n_1 n_2 \times n_1 n_2}$ is given by

$$a_{\substack{\frown \\ i,j}}^{\frown} = a_{(i-1)n_2+i',(j-1)n_2+j'} = \Omega_2(c_{ii'}, c_{jj'}), \tag{5}$$

and its structure is detailed in Appendix A, which can be found in the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2012.51. The optimal assignment C^* is the one maximizing the sum of corresponding pairwise affinities adhering to the same set of constraints as in (3). This yields the following optimization problem:

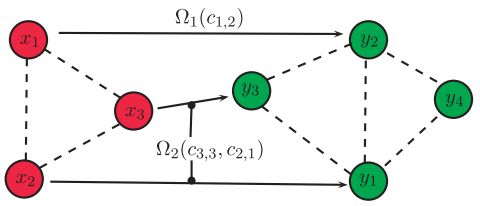


Fig. 1. The graph matching problem. Given two sets of points $S_1 = \{\mathbf{x}_i\}$ and $S_2 = \{\mathbf{y}_i\}$, $c_{1,2}$ indicates that x_1 is matched to y_2. The *unary* affinity $\Omega_1(c_{1,2})$ quantifies the affinity of machining x_1 to y_2. The *pairwise* affinity $\Omega_2(c_{3,3}, c_{2,1})$ quantifies the *joint* matching of x_3 to y_3, and x_2 to y_1. S_1 and S_2 might be of different sizes.

$$\begin{aligned} \mathbf{z}^* &= \arg\max_{\mathbf{z}} \left(\mathbf{z}^T \mathbf{A} \mathbf{z}\right), \quad \mathbf{z} \in \{0,1\}^{n_1 n_2} \\ &s.t.\ \mathbf{Z}\underline{\mathbf{1}} \le \underline{\mathbf{1}} \quad \text{and} \quad \mathbf{Z}^T\underline{\mathbf{1}} \le \underline{\mathbf{1}}, \end{aligned} \tag{6}$$

where $\mathbf{z}$ and $\mathbf{Z}$ are the same indicator vector/matrix pair as in (3).

In some applications, such as shape retrieval [13] and image matching [5], one can utilize unary similarities (based on SIFT and Shape Context) to reduce the assignment space of each vertex in S_1 to $K \ll n_2$ potential assignments in S_2. In practice, in these applications $K \le 10$.

The optimization problem in (6) is a variant of the *quadratic binary programming*(QBP) and is known to be *np*-hard. It is a particular instance of a general problem whose implications are well beyond point matching. For instance, graph matching is equivalent to optimizing Markov Random Fields (MRFs) [8], where pairwise matching corresponds to solving 2D MRFs. It is also at the heart of the GrabCut supervised image segmentation of Rother et al. [30], and Raj and Zabih's work on discrete image deconvolution [27]. In these works, the MRF was solved by the Min-cut/Max-Flow scheme of Boykov and Kolmogorov [3]. In this paper, we exemplify the use of graph matching in point matching as it allows comparison of our results to previous works. Yet both the analysis and proposed matching scheme are applicable to a wider class of problems having a more general set of matching constraints than those in (6).

Although pairwise matching was found to be instrumental [2], [18], [15], some applications require high-order affinities. For instance, pairwise matching is susceptible to the scale differences between the sets of points, and using third-order affinities, one can define scale invariant similarity measures. A probabilistic triplets matching scheme was suggested by Zass and Shashua [37], while Chertok and Keller [5] presented a high-order Spectral Matching (SM) approach. Most matching schemes can recover partial matchings, implying that only subsets of size $n < \min(n_1, n_2)$ of S_1 and S_2 are matched. This is of particular interest in applications where the common set is embedded in clutter.

In this paper, we present two core contributions:

First, we extend the probabilistic model of spectral matching, suggested by Zass and Shashua [37] to show that the spectral matching scheme of Leordeanu and Hebert [21] can be interpreted as a maximum likelihood (ML) estimate of the assignment probabilities, given that the assignments of different points are statistically independent and that the affinity matrix $\mathbf{A}$ is an estimate of the joint assignment probability. We then show that given the same set of assumptions, the Graduated Assignment (GA) algorithm of Gold and Rangarajan [15] can be cast as a Maximum a Posteriori (MAP) estimate that utilizes a maximum entropy prior. We also reinterpret and justify the marginalization operation used by Zass and Shashua [37] in light of the probabilistic interpretation, and introduce an assignment ranking scheme. This allows us to rank the reliability of the spectral matchings and choose the subset of the most reliable assignments.

In our **second** contribution we derive a novel graph matching scheme that relaxes the probabilistic assumptions used in previous works. It iteratively refines the assignment and conditional matching probabilities, and is experimentally shown to compare favorably with previous state-of-the-art-matching schemes as the matching problem becomes difficult due to outliers and noise.

This paper is organized as follows: Section 2 discusses previous results in graph matching, while Section 3 presents a probabilistic analysis of spectral matching, Graduated Assignments and their variants. This paves the way for an analysis of the normalization and the assignment ranking schemes. The novel probabilistic matching (PM) is presented in Section 4, and is experimentally verified in Section 5 by applying it to synthetic as well as real data. Concluding remarks and future extensions are discussed in Section 6.

2 Related Work

A myriad of algorithms were proposed for matching graphs and sets of point in $\mathbb{R}^d$, and a comprehensive survey was conducted by Conte et al. [7]. One of the earliest algorithms for point matching is due to Scott and Longuet-Higgins [31] that solve the one-to-one matching by minimizing the sum of squared distances between matched points in both sets. They introduce a spectral formulation by computing a Gaussian-weighted affinity matrix $a_{ij} = \exp(-\|\mathbf{x}_i - \mathbf{y}_j\|^2/\varepsilon^2)$. $\mathbf{A}$ is replaced by $\mathbf{P} = \mathbf{U}\mathbf{V}$, where $\mathbf{U}$ and $\mathbf{V}$ are matrices whose columns are the left and right singular vectors of $\mathbf{A}$. The largest entries of $\mathbf{P}$ indicate strongly coupled points. This scheme implicitly assumes that $\mathbf{A}$ is an empirical approximation of the assignment matrix $\widehat{\mathbf{A}}$, for which we have $\widehat{\mathbf{A}}\underline{\mathbf{1}} = \widehat{\mathbf{A}}^T\underline{\mathbf{1}} = \underline{\mathbf{1}}$. It was observed by Cour et al. [9] that this spectral decomposition acts as a projection operator onto the space of assignment matrices.

Due to the equivalence between the graph matching problem (as defined in (6)) and a particular class of QBPs with corresponding sets of constraints that is known to be NP-hard, it follows that this computational problem is NP-hard [14]. Thus, it is commonly solved using relaxation-based approaches, where some of the constraints in (6) are dropped to derive low computational complexity approximations.

Leordeanu and Hebert [21] presented an efficient and robust solution to (6) by way of spectral relaxation:

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \frac{\mathbf{w}^T \mathbf{A} \mathbf{w}}{\mathbf{w}^T \mathbf{w}}, \quad \mathbf{w} \in \mathbb{R}^{n_1 n_2}. \tag{7}$$

Equation (7) is solved by computing the leading eigenvalue and corresponding eigenvector of $\mathbf{A}$. In point matching it is common to compute the affinity using a Gaussian kernel as in (4), guaranteeing that $\mathbf{A}$ is symmetric and nonnegative. Thus,

by the Peron-Frobenius theorem we have that the leading eigenvalue and eigenvector $\mathbf{w}^*$ of $\mathbf{A}$ are known to exist and $\mathbf{w}^*$ is nonnegative. Given the continuous solution $\mathbf{w}^*$, Leordeanu and Hebert [21] suggest discretizing $\mathbf{w}^*$ by a greedy approach and deriving the indicator vector $\mathbf{z}^*$. The assignment constraints in (6) are ignored in the spectral relaxation step (7) and induced during the discretization step.

The Graduated Assignment algorithm of Gold and Rangarajan [15] solves the graph matching problem by relaxing the constraint $\mathbf{z}^* \in \{0,1\}^{n_1 n_2}$ in (6) to $\mathbf{w} \in \mathbb{R}^{n_1 n_2}$; the derivative of the relaxed QBP is computed and an iterative update scheme is derived. The relaxation in this scheme boils down to a power iteration and thus precedes [21]. The GA utilizes the softassign operator that is parameterized by a continuation parameter, annealed after each iteration. It minimizes an objective function comprising of the quadratic term (as in (6)) and an entropy barrier function:

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \left(\mathbf{w}^T \mathbf{A} \mathbf{w} - \frac{1}{\beta} \sum_i w_i \log w_i \right) \quad s.t.\ \mathbf{W}\underline{\mathbf{1}} = \underline{\mathbf{1}} \ \text{ and } \ \mathbf{W}^T\underline{\mathbf{1}} = \underline{\mathbf{1}},\ \mathbf{w} \in \mathbb{R}^{n_1 n_2},\ \beta > 0. \tag{8}$$

The softassign operator allows us to approximate the solution of discrete assignment problems using continuous operators, and was first presented by Mjolsness [24] in the context of neural networks denoted as "assignment networks." The softassign was also applied by Rangarajan and Mjolsness to graph isomorphism estimation [28] and to the joint estimation of assignments and parametric motion [16].

The convergence of the GA was studied by Rangarajan et al. [29] via discrete time Lyapunov functions. Yuille and Rangarajan [36] introduced the Concave-Convex Optimization Procedure (CCCP) that is a general approach to iteratively minimizing objective functions consisting of a sum of a concave and convex functions. The CCCP is guaranteed to monotonically decrease the objective function and can be used to derive the GA by applying it to (8).

Similarly to the GA algorithm, van Wyk and van Wyk [34] based their iterative algorithm on the first derivative of the objective function. They suggest a novel approach for projecting an approximation of the current matching matrix onto the convex space of the matching constraints. In addition, their algorithm does not require an annealing parameter as the GA.

Cour et al. [9] proposed two extensions to Leordeanu's work. First, they introduce affine constraints into the spectral decomposition, that encode the one-to-one matching constraints. Their second contribution is to apply bistochastic normalization to the Edge Similarity Matrix, which is a an equivalent representation of the pairwise affinity matrix.

A synergy of structural graph matching and the estimation of parametric motion models (affine, projective) was proposed by Cross and Hancock [10], and applied to the matching of 2D point-sets. Each point set was represented by a graph, and the point (node) correspondence was represented by a bipartite graph. The assignment probabilities are modeled via a mixture model, based on a Gaussian noise model with respect to the correspondence and motion parameters. This model is estimated using an Expectation Maximization (EM) scheme.

Lug and Hancock [23] presented an EM-based inexact graph matching scheme that only utilizes the edge (connectivity) structure of the graphs. Their probabilistic formulation casts the graph to be matched as observed data and the set of correspondences as the hidden variables. A mixture model is computed with respect to the node correspondences, and the correspondence errors are modeled via a Bernoulli distribution. The spectral approach of Scott and Longuet-Higgins [31] is used in the maximization step of the EM algorithm.

Graph embedding approaches to graph matching are based on embedding the graphs to be matched in a euclidean space, where the graph nodes are represented by the embedding coordinates. The premise is that such embeddings can be made invariant to rigid transformations of the nodes' locations. A spectral graph embedding scheme based on Kernel PCA was proposed by Wang and Hancock [35], that derived a one-to-one point matching scheme for points related by rigid motion. This approach is then extended to handle articulated motion by introducing label propagation into the iterative matching process.

The work of Zass and Shashua [37] is of particular interest as it introduces a probabilistic framework for hypergraph matching. They show that given that different assignments are statistically independent, high-order matching problems can be represented by a matrix constructed by Kronecker products. They also offer two computational contributions. First, they show that high-order affinity tensors can be marginalized into a 1D probability vectors. In their second contribution, the probability vector is refined by projecting it into the space of assignment vectors by minimizing a Bregman measure.

A spectral approach to high-order graph matching was proposed by Duchenne et al. [12]. The high-order matching is formulated as a tensor eigendecomposition problem, and is applied to point matching using an appropriate affinity measure. It is experimentally shown to outperform previous schemes.

3 Probabilistic Interpretation of Spectral Matching

In this section, we present a probabilistic interpretation of spectral matching schemes. As for notations, we follow the notations depicted in Fig. 1 and detailed in Appendix A, available in the online supplemental material. Let $P(c_{ii'})$ be the *assignment probability* of the i'th match, in which $\mathbf{x}_i \in S_1$ is assigned to $\mathbf{y}_{i'} \in S_2$. $P(c_{ii'}, c_{jj'})$ is the *pairwise assignment probability* such that $\mathbf{x}_i \in S_1$ matches $\mathbf{y}_{i'} \in S_2$ *and* $\mathbf{x}_j \in S_1$ matches $\mathbf{y}_{j'} \in S_2$, We denote $\mathbf{p}$ as the vector of assignment probabilities as in Appendix A, available in the online supplemental material.

We aim to relate these probabilities to the graph assignment problem. For that we utilize the probabilistic formulation of high-order graph matching proposed by Zass and Shashua [37]. Their work introduces the following *working* assumptions.

Assumption 1. *The pairwise affinity matrix A is an empirical estimate of the pairwise assignment probability*

$$\Omega_2\big(c_{ii'}, c_{jj'}\big) = P\big(c_{ii'}, c_{jj'}\big). \tag{9}$$

As we used a Gaussian kernel to compute A, we have that $\Omega_2(c_{ii'}, c_{jj'}) \in [0,1]$, where $\Omega_2(c_{ii'}, c_{jj'}) \approx 0$ corresponds to invalid matches ($P(c_{ii'}, c_{jj'}) \approx 0$), while $\Omega_2(c_{ii'}, c_{jj'}) \approx 1$ corresponds to jointly valid matches ($P(c_{ii'}, c_{jj'}) \approx 1$).

Assumption 2. *The assignments of different points $\mathbf{x}_i \in S_1$ are statistically independent. Thus, we have that*

$$P(c_{ii'}, c_{jj'}) = P(c_{ii'})P(c_{jj'}), \tag{10}$$

Substituting (9) in (10):

$$\Omega_2(c_{ii'}, c_{jj'}) = P(c_{ii'})P(c_{jj'}), \tag{11}$$

and by Rewriting (11) in matrix notation, we get that

$$\mathbf{A} = \mathbf{p}\mathbf{p}^T. \tag{12}$$

Hence, the assignment probability $\mathbf{p}$ can be estimated by computing the Rank-One-Approximation (ROA) of the affinity matrix $\mathbf{A}$. Following the Eckart-Young Theorem [20], the eigendecomposition of $\mathbf{A}$ is its optimal ROA in terms of the Frobenius norm

$$\mathbf{p}^* = \arg\min_{\mathbf{p}} \|\mathbf{A} - \mathbf{p}\mathbf{p}^T\|_{L_2}. \tag{13}$$

Thus, the eigendecomposition is a proxy for computing the ROA of the affinity matrix [17] without having to relate it to spectral relaxation or discrete optimization. It is common to compute the affinity using a Gaussian kernel, as in (4), insuring that $\mathbf{A}$ is symmetric and nonnegative. Thus, by the Perron-Frobenius theorem, we have that the leading eigenvector of $\mathbf{A}$ is known to exist and is nonnegative.

In general, Assumptions 1 and 2 might be invalid and are thus *working assumptions*. Yet they are implicitly used in the spectral matching approach of Leordeanu and Hebert [21] and the Graduated Assignment algorithm of Gold and Rangarajan [15].

It was shown by Chertok and Keller [5] that the discretization of the probabilities vector $\mathbf{p}$ can be formulated as a Maximum Likelihood estimate, where we maximize the overall probability of the chosen assignments, under the matching constraints:

$$\begin{aligned}\mathbf{z}^* = \arg\max_{\mathbf{z}}\big(\mathbf{z}^T\mathbf{p}\big) = \arg\max_{\mathbf{z}} \sum_{k \text{ s.t.} \mathbf{z}_k = 1} \mathbf{p}_k, \mathbf{z} \in \{0,1\}^{n_1 n_2} \\ s.t.\ \mathbf{Z}^*\underline{\mathbf{1}} \le \underline{\mathbf{1}} \quad \text{and} \quad (\mathbf{Z}^*)^T\underline{\mathbf{1}} \le \underline{\mathbf{1}},\end{aligned} \tag{14}$$

where $\mathbf{z} \in \{0,1\}^{n_1 n_2}$ and $\mathbf{Z} \in \{0,1\}^{n_1 \times n_2}$ are the assignment vector and matrix, respectively.

In general, (14) can be solved by binary linear programming [26], but in most assignment problems, one can use the greedy approach of Leordeanu and Hebert [21] or the Hungarian algorithm [25]. It follows that the spectral matching scheme of Leordeanu and Hebert [21] is comprised of two steps: The spectral decomposition in (13) provides an estimate of the assignment probabilities, while the discretizations in (14) are an ML estimate of the hard (binary) assignments.

We now provide a probabilistic interpretation of the GA algorithm [15]. The GA scheme iterates the following steps:

$$\mathbf{q}_t = Normalize(\mathbf{p}_t), \tag{15a}$$

$$\mathbf{r}_t = \exp(\beta_t \mathbf{q}_t), \tag{15b}$$

$$\mathbf{p}_{t+1} = \mathbf{A}\mathbf{r}_t, \tag{15c}$$

$$\beta_{t+1} = \beta_t \Delta\beta \ \ (\Delta\beta > 1), \tag{15d}$$

where $\mathbf{A}$ is the affinity matrix and $\beta_t > 0$ is an annealing parameter.

The first term in (8) identifies with the quadratic term of the SM algorithm, yielding an estimate of the assignment probability, while the second is an entropy maximization term that acts as a prior. Thus, (15c) identifies with the power iteration [17] used in the SM algorithm. By applying the CCCP to the functional in (8), it follows that (15b)-(15d) constitute an entropy maximization scheme, while the coefficient β_t controls the tradeoff between the quadratic term (likelihood) and the prior. These two terms are contradictory since the optimal assignment $\mathbf{p}^*$ has a low entropy due to its sparsity. In contrast, the maximal entropy prior is maximized by a uniform vector $\mathbf{p}$. The entropy term allows the GA to avoid poor local maxima, and as β_t increases exponentially over the iterations, the tradeoff between the ML and the entropy terms evolves in favor of the ML solution.

Another observation relates to the work of Zass and Shashua [37] that proposed to marginalize affinity matrices and tensors to estimate the assignment probability $P(c_{ii'})$. This can be justified by utilizing Assumption 1:

$$P(c_{ii'}) = \sum_{c_{jj'}} P(c_{ii'}, c_{jj'}). \tag{16}$$

Although, computationally appealing, as it does not require computing an eigendecomposition, the marginalization is only equivalent to spectral matching in the outlier-free case, as it sums all of the elements in the affinity matrix. When outliers exist, (16) does not hold, and the marginalization does not allow the adaptive weighting used by the SM and GA schemes. This hampers the performance in the presence of outliers and will be experimentally demonstrated in Section 5.

3.1 Probability Functions Normalizations

The computation of the soft assignment vector $\mathbf{p}_t \in \mathbb{R}^{n_1 n_2}$ at iteration t, or equivalently, the corresponding soft assignment matrix $P_t \in \mathbb{R}^{n_1 \times n_2}$, requires their normalization. Typically, in a Power Iteration $\mathbf{p}_t$ is normalized to a unit L_2 in each iteration [17], while in the GA scheme P_t is bistochastically normalized using the Sinkhorn algorithm [32]. Following our probabilistic analysis, the normalization can be interpreted as a projection of P_t onto the space of probability functions. Hence, row-normalizing P_t is equivalent to setting $\sum_{i'} P(c_{ii'}) = 1$. This implies that each point in S_1 is assumed to be matched, and column-normalization relates to the points in S_2 mutatis mutandis. Bistochastic normalization corresponds to the assumption of one-to-one mapping. Hence, the choice of the normalization scheme

implicitly induces assignment constraints. For instance, in some problems, we match an outlier-free model/graph (WLOG assume it is S_1) to an acquired set of points S_2 that contains outliers. In such a setup, row-normalization will be more appropriate than column or bistochastic normalization. Applying the incorrect normalization might degrade the matching accuracy, as ideally for an outlier we expect to have $\sum_{i'} P(c_{ii'}) \approx 0$.

From an algebraic point of view, each of the normalization operations (rows, columns, and bistochastic) can be considered a projection operator onto a closed and convex set, as first noted by Hummel and Zucker [19]. Hence, introducing them into the different iterative schemes (SM, GA, PM) does not hamper the convergence properties.

The pairwise assignment probability $P(c_{ii'}, c_{jj'})$ can also be normalized in different ways, inducing different matching constraints. Thus, in an assignment problem where each point in S_1 is known to correspond to a point in S_2, but S_2 might contain outlier points, one would use

$$\widehat{P}(c_{ii'}, c_{jj'}) = P(c_{ii'}, c_{jj'}) \Big/ \sum_{i',j'} P(c_{ii'}, c_{jj'}),$$

thus inducing the constraint $\sum_{i',j'} \widehat{P}(c_{ii'}, c_{jj'}) = 1$. When the assignment is known to be one-to-one, we impose the constraint that $\sum_{i',j'} \widehat{P}(c_{ii'}, c_{jj'}) = 1$ and $\sum_{i,.j} \widehat{P}(c_{ii'}, c_{jj'}) = 1$ by applying the Sinkhorn algorithm [32] to $P(c_{ii'}, c_{jj'})$.

In general, given an assignment problem where both S_1 and S_2 might contain outliers, we found it best not to normalize $P(c_{ii'}, c_{jj'})$ as for an outlier point $\mathbf{x}_i \in S_1$, we should have $\sum_{i,j} \widehat{P}(c_{ii'}, c_{jj'}) \approx 0$, and applying any normalization, as discussed above, might induce erroneous assignment constraint on the outlier points and bias the solution.

3.2 Assignment Ranking

The probabilistic framework also provides an approach for ranking the reliability of the SM assignments. Given n_1 assignments computed via the SM, we aim to rank their reliability. This allows us to choose the "best" (most reliable) $\widehat{n} \ll n_1$ assignments as, in some applications, only $\widehat{n} \ll n_1$ are required. For instance, Chertok and Keller [6] used $\widehat{n} \ll n_1$ point matches to initialize a RANSAC-based parametric motion estimation to incorporate the prior knowledge that symmetric objects are related by rotations and reflections. Given the original matching problem, the RANSAC scheme might diverge due to the significant number of outliers. But, given the $\widehat{n} \ll n_1$ best SM matches, the number of outliers is reduced.

Our ranking approach is based on the probabilistic interpretation of the assignment vector $\mathbf{p}$. Given the hard assignment $\mathbf{z}^*$ computed via any of the discretization schemes, we consider the corresponding entries in the soft assignment vector $\mathbf{p}$. Following our probabilistic analysis, these are the assignment probabilities $P(c_{ii'})$ and can be used to rank the assignments

$$P(c_{i_1 i'_1}) \geq P(c_{i_2 i'_2}) \geq \cdots \geq P(c_{i_{n_1} i'_{n_1}}), \tag{17}$$

as an assignment with a higher probability $P(c_{ii'})$ is a more reliable one. We exemplify the validity of this approach in Section 5.

4 Probabilistic Graph Matching

In this section, we introduce the proposed probabilistic graph matching scheme (PM). The core of our approach is based on the observation that we can use the solution of the spectral matching algorithm [21] to refine the estimate of the affinity matrix $\mathbf{A}$ and then solve a new assignment problem based on the refined matrix $\mathbf{A}$. Namely, we can attenuate the affinities corresponding to matches with small matching probabilities and thus prune the affinity matrix $\mathbf{A}$. In the same vein, we aim to adaptively increase the entries in $\mathbf{A}$ corresponding to assignments with high matching probabilities. Specifically, given the *estimated* assignment probabilities $P(c_{ii'}) \approx 0$ corresponding to an unlikely assignment $c_{ii'}$, we attenuate the entries $P(c_{ii'}, c_{jj'})$, $j = 1..n_1$. Conversely, given an estimated high assignment probability $P(c_{ii'}) \approx 1$, we aim to increase the corresponding entries of $\mathbf{A}$.

The crux of our approach is to drive an iterative formulation where each iteration is comprised of two steps: The first estimates the assignment probabilities $P(c_{ii'})$ given the current estimate of the affinity matrix $\mathbf{A}$, and the second utilizes $P(c_{ii'})$ to refine $\mathbf{A}$, as previously discussed. The first step corresponds to applying the SM schemes, while the refinement of $\mathbf{A}$ should be carefully chosen to allow analytic interpretation and provable convergence.

Hence, we propose to minimize the objective function

$$\begin{aligned}
&[P^*(c_{ii'}), P^*(c_{ii'}|c_{jj'})] \\
&\quad = \arg \min_{\substack{P(c_{ii'}) \\ P(c_{ii'}|c_{jj'})}} \sum_{ii'} \left(\left(\sum_{jj'} P(c_{ii'}|c_{jj'}) P(c_{jj'}) \right) - P(c_{ii'}) \right)^2 \\
&\qquad\qquad s.t.\ P(c_{ii'}) \text{ is a probability,}
\end{aligned} \tag{18}$$

where $P(c_{ii'}|c_{jj'})$ is the *conditional assignment probability*, that is, the probability of the assignment $c_{ii'}$, *given* that the assignment $c_{jj'}$ is valid. The assignment probability $P(c_{jj'})$ is an eigenvector of $P(c_{ii'}|c_{jj'})$, corresponding to an eigenvalue of 1:

$$\sum_{jj'} P(c_{ii'}|c_{jj'}) P(c_{jj'}) = \sum_{jj'} P(c_{ii'}, c_{jj'}) = P(c_{ii'}). \tag{19}$$

Thus, for a fixed $P(c_{ii'}|c_{jj'})$, the proposed formulation boils down to the regular SM scheme, and can be solved via the Power Iteration. Yet in our scheme we aim to update both $P(c_{ii'}|c_{jj'})$ and $P(c_{jj'})$.

The pairwise probability $P(c_{ii'}, c_{jj'})$ is not used as it cannot be easily updated: Having a high assignment probability $P(c_{ii'}) \approx 1$ does not imply that $P(c_{ii'}, c_{jj'}) \approx 1$ as we might have that $P(c_{jj'}) \approx 0$. In contrast, $P(c_{ii'}|c_{jj'})$ is asymmetric, and given that $P(c_{ii'}) \approx 1$, we can increase $P(c_{ii'}|c_{jj'})$ regardless of $P(c_{jj'})$.

Denote by $P_t(c_{ii'}|c_{jj'})$ and $P_t(c_{jj'})$ the estimate of $P(c_{ii'}|c_{jj'})$ and $P(c_{ii'})$ in iteration t, respectively. We propose to solve (18) by iterating the following two steps.

At iteration t we *fix* $P_t(c_{ii'}|c_{jj'})$ and compute a single iteration of the Power Iteration:

$$P_{t+1}(c_{ii'}) = \sum_{jj'} P_t(c_{ii'}|c_{jj'})P_t(c_{jj'}). \quad (20)$$

The estimate of the conditional assignment probability $P_t(c_{ii'}|c_{jj'})$ is refined by

$$P_{t+1}(c_{ii'}|c_{jj'}) = P_t(c_{ii'}|c_{jj'})\frac{P_{t+1}(c_{ii'})}{P_t(c_{ii'})}. \quad (21)$$

Due to the convergence properties of the Power Iteration, $P_{t+1}(c_{ii'})/P_t(c_{ii'}) \geq 1$ for valid assignments and $P_{t+1}(c_{ii'})/P_t(c_{ii'}) \leq 1$ for invalid ones. Hence, (21) adaptively increases the entries in $P_t(c_{ii'}|c_{jj'})$ corresponding to valid assignments and attenuates the invalid ones. The proposed PM scheme is summarized in Algorithm 1.

Algorithm 1. Probabilistic Graph Matching

1: Given the pairwise affinity matrix $\mathbf{A} \in \mathbb{R}^{\mathbf{n_1 n_2} \times \mathbf{n_1 n_2}}$, the number of iterations $IterNo$ and the threshold δ
2: Set $\mathbf{\Lambda}_0 = \mathbf{A}$ and $\mathbf{p}_0 = \frac{1}{n_2}\mathbf{1}$ where $\mathbf{p}_0 \in \mathbb{R}^{n_1 n_2}$
3: **for** $t = 0$ to $(IterNo - 1)$ **do**
4: $\quad \mathbf{q}_t = \mathbf{\Lambda}_t \mathbf{p}_t$
5: $\quad \mathbf{p}_{t+1} = Normalize(\mathbf{q}_t)$
6: $\quad \mathbf{\Lambda}_{t+1}(i,j) = \mathbf{\Lambda}_t(i,j)\frac{\mathbf{p}_{t+1}(i)}{\mathbf{p}_t(i)}$
7: $\quad$ if $\frac{\|\mathbf{p}_{t+1} - \mathbf{p}_t\|_2}{n_1 n_2} < \delta$, go to Step 9
8: **end for**
9: Discretize $\mathbf{p}_{t+1}$

We prove in Appendix B, available in the online supplemental material, that each of the consecutive steps in (20) and (21) monotonically reduces the objective function in (18). In practice, one can use different weighting terms in (21), such as $(\frac{P_{t+1}(c_{ii'})}{P_t(c_{ii'})})^2$, yielding similar experimental results, but we were unable to prove convergence for such terms.

The assignment probability is initialized to be $U[0,1]^n$, and we set $P_0(c_{ii'}|c_{jj'}) = \mathbf{A}$, that is, the affinity matrix used in the SM scheme. In each iteration, $P_t(c_{ii'})$ is normalized according to the schemes discussed in Section 3.1. As a stopping criterion we used a combination of 20 iterations, and a predefined threshold δ on the refinement of the probability vector. The estimated probability is then discretized using the discretization schemes discussed in Section 3. Similarly to the GA scheme, the probability estimate $P_t(c_{ii'})$ computed by the PM converges to hard (binary) assignments. Hence, both the greedy and Hungarian-based discretizations provide similar results. This formulation does not assume statistical independence among the different assignments, and thus relaxes Assumption 2 used by the Zass and Shashua model.

Fig. 2 illustrates the evolution of the conditional and assignment probabilities with respect to the PM iterations. We generate a set of $n = 10$ random points $\mathbf{x}_i \in \mathbf{U}[0,1]^2$ and create a noisy replica of it, $\mathbf{y}_i = \mathbf{x}_i + \gamma_i, \gamma_i \sim N(0, 0.1)$, thus matching two 10×10 graphs. Initially, $P_t(c_{ii'}|c_{jj'})$ (Fig. 2a) is given by the affinity matrix $\mathbf{A}$ and is thus symmetric, while $P_1(c_{ii'})$ (Fig. 2d) is nonuniform. After five iterations, $P_5(c_{ii'}|c_{jj'})$ (Fig. 2b) is asymmetric and consists of horizontal nonzero elements corresponding to assignments for which $P_5(c_{ii'}) > 0$. The assignment probability $P_5(c_{ii'})$ (Fig. 2e) is close to convergence. After 10 iterations (Fig. 2f), $P_{10}(c_{ii'})$ converges to a binary indicator vector, as expected.

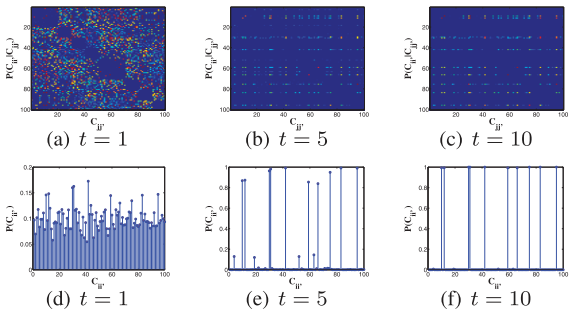


Fig. 2. The convergence of the proposed probabilistic matching scheme. The first row (a)-(c) shows the evolution of the conditional assignment probability $P_t(c_{ii'}|c_{jj'})$ with respect to the number of the iterations $t = 1, 5, 10$. In the second row (d)-(f), we depict the corresponding evolution of the assignment probability $P_t(c_{ii'})$.

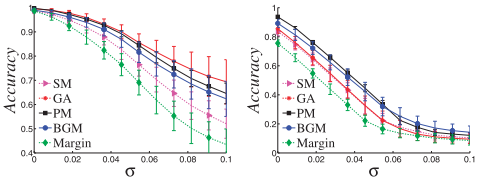
4.1 Computational Complexity

The computational complexity of the proposed PM scheme per iteration consists of the $O(n_1^2 K^2)$ operations required for the matrix-vector multiplication in step #4 of Algorithm 1, as well as $O(n_1 K)$ operations to row-normalize the assignment matrix and $O(n_1^2 K^2)$ operations needed to weigh the affinity matrix (step #6). Only step #6 is specific to the proposed PM scheme, whose overall complexity is $O(n_1^2 K^2)$, the same as the SM, GA and Balanced-Graph-Matching (BGM) [9] schemes.

5 Experimental Results

We applied the proposed approach to synthetically generated random graphs as well as real images. In the synthetic graphs simulations we followed the experimental framework used in previous works [15], [37]. It allows us to conduct large-scale simulations and quantify the accuracy and resiliency of the proposed scheme, and compare it to contemporary state-of-the-art algorithms. The same synthetic graph having a known ground truth is input to the different matching schemes at a time. We also experiment with the *Hotel* and *House* sequences and compare our results to state-of-the art schemes: the publicly available[1] implementation of the probabilistic matching scheme by Zass and Shashua [37] (Margin), where we applied the marginalization and probabilistic optimization to a pairwise affinity matrix. We also implemented the Spectral Matching scheme of Leordeanu and Hebert [21], the Balanced-Graph-Matching by Cour et al. [9], and the Graduated Assignment algorithm of Gold and Rangarajan [15]. For these we based our code on the implementation by Timothee Cour.[2] In all trials *the same* affinity matrix was the input to all of the reference schemes (and ours). We ran all schemes until $\frac{\|\mathbf{p}_{t+1} - \mathbf{p}_{t+1}\|_2}{n_1 n_2} < \delta$, $\delta = 10^{-3}$, or 20 iterations were executed. We experimentally verified that allowing more iterations or decreasing δ did not improve the results. We used the Hungarian algorithm to discretize all schemes as it provided better results for the SM and BGM. The GA and the proposed PM schemes converge to hard assignment solutions; hence both the Hungarian and greedy discretizations perform similarly.

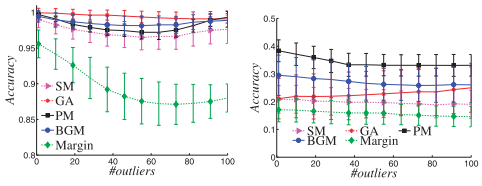
1. The code by Ron Zass is available at: http://www.cs.huji.ac.il/~zass/gm/.
2. http://www.seas.upenn.edu/~timothee/software/graph_matching/graph_matching.html.



(a) $n = 200, K = 10$ (b) $n = 200, K = 50$

(c) $n = 200, K = 100$ (d) $n = 200, K = 200$

Fig. 3. Noise test: Matching accuracy with respect to varying intensity of an additive noise $N(0, \sigma)$ that were added to the matched sets of points. In each simulation, we match 200 points with a varying number of K potential assignments. We compared the Spectral Matching [21], Graduated Assignment [15], proposed Probabilistic Matching, Balanced-Graph-Matching [9], and probabilistic Marginalization [37] (Margin).



(a) $n = 200, K = 10$ (b) $n = 200, K = 50$

(c) $n = 200, K = 100$ (d) $n = 200, K = 200$

Fig. 4. Outliers test results where outliers are introduced to both matched sets. We show the matching accuracy versus the number of outliers for graphs with $n = 200$ nodes, and a varying number of K possible assignments per point. We compared the Spectral Matching [21], Graduated Assignment [15], proposed Probabilistic Matching, Balanced-Graph-Matching [9], and probabilistic Marginalization [37] (Margin).

5.1 Matching of Random Synthetic Graphs

In this set of trials we measure the accuracy and resiliency of our approach to additive noise and outliers. In each trial we generated a set of $n = 200$ random points $S_1 = \{\mathbf{x}_i\}_{i=1}^{n}$, $\mathbf{x}_i \in \mathbb{R}^2$, $\mathbf{x}_i \sim U[0, 1]^{2\times 1}$, and denote this randomly generated set as the *source set*. In the noise trials we added noise to S_1:

$$\mathbf{y}_i = \mathbf{x}_i + \gamma_i, \quad \forall \mathbf{x}_i \in S_1, \quad \gamma_i \sim N(0, \sigma),$$

and denote the distorted source set as the *target set* $S_2 = \{\mathbf{y}_i\}_{i=1}^{n}$, where $\mathbf{x}_i$ corresponds to $\mathbf{y}_i$. In applications such as image matching this simulates image distortions as well as localization inconsistencies of the interest points' detectors. In this trial, the matching is one-to-one as we did not add outliers. In real matching scenarios, descriptors can be used to prune the set of possible assignment per point $\mathbf{x}_i \in S_1$. We applied the different schemes with $K = \{10, 50, 100, 200\}$. This provides a varying degree of matching difficulty. Using $K = 10$ corresponds to having highly discriminative local features, and the larger K is the more difficult the matching.

We measure the matching accuracy as the ratio of correctly matched points to the total number of points that could potentially be matched. In all simulations we used the affinity measure given in (4) with the same kernel bandwidth $\varepsilon = \frac{1}{n}$. For a given noise variance we repeated the same experiment 1,000 times by generating 1,000 random sets S_1 (source graphs) and adding noise. We report the mean accuracy and standard deviation over the different trials.

The results are depicted in Fig. 3, where we vary the number of possible assignments per point K. For the easiest setup in Fig. 3a with $K = 10$, the PM is second to the GA. Yet, as K increases in Figs. 3b, 3c, and 3d and the matching problem becomes more difficult, the proposed PM scheme proved superior, while the GA came in second.

Partial matching is of particular interest, as one-to-one matching is rarely present in actual applications. A feature points detector might detect points in one image, many of them lacking corresponding counterparts in the other image. Hence, in the outlier tests we added points both to the source and target sets. We generate random sets with $n = 200$ points and add 100 random outliers to the source set S_1. Those have no correspondences in the second graph. Their potential K matches per point are picked randomly. For the inliers we insert the valid match as one of the K potential matches. We also added a varying number of outliers to the target set S_2, and random noise $N(0, \sigma)$, $\sigma = 0.05$. The results depicted in Fig. 4 show that the PM outperformed the other schemes as the matching problem became challenging for $K = \{100, 200\}$.

5.2 Assignment Ranking

We exemplify the assignment ranking approach introduced in Section 3.2 by analyzing the noise simulation results for matching $n_1 = 200$ points to $n_2 = 200$ points using $K = 50$ and the SM scheme. The average accuracy results are presented in Fig. 3b, while the average assignment accuracy *per ranking* is shown in Fig. 5. The per ranking results were averaged over 1,000 simulations, where in each simulation we ranked the assignments by sorting their assignment probabilities according to Section 3.2. The validity of the ranking scheme is exemplified by comparing the accuracy

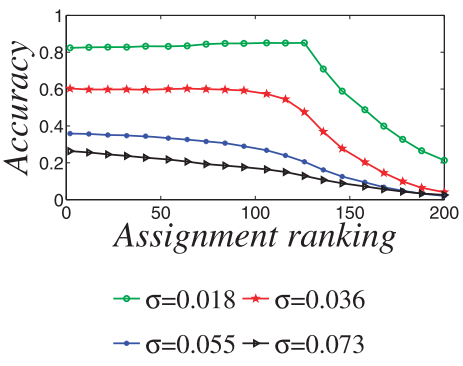
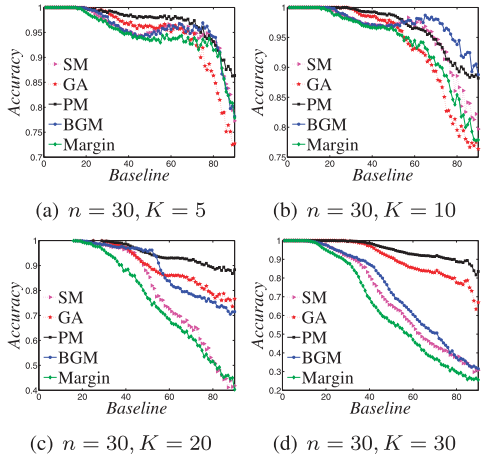


Fig. 5. Probabilisitc ranking of the spectral matching results. We show the average matching accuracy versus the probabilistic ranking of the assignments for several additive noise levels $N(0, \sigma)$ that were added to the matched sets of points.



(a) $n = 30, K = 5$ (b) $n = 30, K = 10$

(c) $n = 30, K = 20$ (d) $n = 30, K = 30$

Fig. 6. Hotel series analysis results. We tracked $n = 30$ features points over the 101 frames of the sequence, using a varying number of K possible assignments, and show the matching accuracy versus the baseline, that is, the temporal differential (in frames) between matched frames. We compared the Spectral Matching [21], Graduated Assignment [15], proposed Probabilistic Matching, Balanced-Graph-Matching [9], and probabilistic Marginalization [37] (Margin).

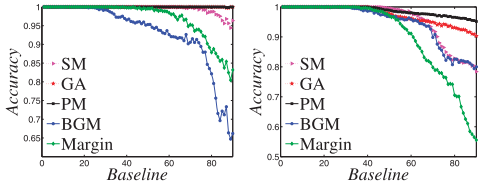
results in both figures for a given noise level. For instance, consider the accuracy curves for $\sigma \approx 0.036$ in Fig. 5 and compare it to the overall accuracy of the SM in Fig. 3b. It follows that the average assignment accuracy achieved by choosing *all* of the assignments is 43 percent, but by choosing the leading 100 assignments we can achieve an accuracy of close to 60 percent. In particular, there is a significant difference in the average accuracy of the best versus the worst assignments.

5.3 Image Sequence Matching

We applied our approach to the CMU *Hotel* and *House* sequences.[3] These were used in multiple works [12], [33], [4] and provide a baseline for comparison. In order to assess the matching accuracy we tracked landmark feature points that were manually labeled and tracked in all frames [4]. This allows us to compare the performance of the different schemes over a varying temporal baseline—the larger the temporal baseline (differential) between the frames is, the larger the relative deformation and the more difficult the matching. We matched each frame to the frames succeeding it and computed the average matching error per temporal baseline.

As in the noise tests in the previous section, we utilized the number of possible assignments K as a mean to vary the matching difficulty. For that we used shape-context shape descriptors [1] as a similarity measure between points. Hence, when matching a point in a particular frame it can only be matched to its K nearest shape descriptors. In some previous works [12], [33], [4], the shape descriptors were used as a unary term in the quadratic matching formulation. Such formulations require balancing the unary versus the pairwise terms that encode descriptors versus geometric similarity, respectively. We found it best to avoid using the unary terms explicitly, and use them to reduce the assignment space to the K nearest points.

3. Available at: http://vasc.ri.cmu.edu/idb/html/motion/hotel/index.html.



(a) $n = 30, K = 20$ (b) $n = 30, K = 30$

Fig. 7. *House* series analysis results. We tracked $n = 30$ features points over the 111 frames of the sequence, using a varying number of K possible assignments, and show the matching accuracy versus the baseline, that is, the temporal differential (in frames) between matched frames. We compared the Spectral Matching [21], Graduated Assignment [15], proposed Probabilistic Matching, Balanced-Graph-Matching [9], and probabilistic Marginalization [37] (Margin).

The results for the *Hotel* sequence are depicted in Fig. 6. The sequence consists of 101 frames and 30 landmark points and we vary $K = \{5, 10, 20, 30\}$. For $K = 5$, the matching problem is well constrained and all schemes perform similarly. The decline in the accuracy for large baseline values is due to the inaccuracy of the shape descriptors as the shape deformations become substantial. Hence, the true matching is not within the $K = 5$ nearest neighbors of the SCs.

For $K = 10$, the proposed PM scheme came second to the BGM, whose bistochastic normalization is well suited for the one-to-one matching in this sequence. As the assignment problem becomes more difficult ($K = \{20, 30\}$), the PM outperforms the other schemes, with the GA coming in second.

We repeated the above experiment with the *House* sequences that consist of 30 landmark points and 111 frames. The motion in this sequence is more rigid than in the *Hotel* sequence and easier to match. The results reported in Fig. 7 show that for $K = 20$ both the GA and the PM perform similarly, but as the matching difficulty increased to $K = 30$, the PM proved superior with the minimal average error rate of ≈ 2 percent.

It is worth noting that superior results were reported by Torresani et al. [33] that achieved an accuracy of 100 percent for the *Hotel* sequence. Their approach differs from ours, as the PM does not utilize unary affinity explicitly as in [33], and the *Dual Decomposition* is significantly slower.

5.4 Timing Results

The timing results are reported in Fig. 8. The measurements were taken on a 2.2 GHz computer running a Matlab implementation, where we input the same affinity matrix of varying size to the different matching algorithms. We applied 10 iterations of each scheme and, for each affinity matrix, the experiment was repeated 1,000 times. Although all of the schemes are of complexity $O(n_1^2 K^2)$, we got different timing results, where the PM and Balanced-Graph-Matching required the most computational time. We attribute this to our Matlab-based implementation as the matrix-vector multiplication in Step #4 of Algorithm 1 (and in the SM, GM, and BGM) requires $O(n_1^2 K^2)$ operations, the same as the weighting of the affinity matrix within the PM, and the normalization of the affinity matrix in the BGM algorithm. Yet the matrix-vector multiplication is implemented via an optimized internal Matlab function, while the others are implemented in Matlab code that is less

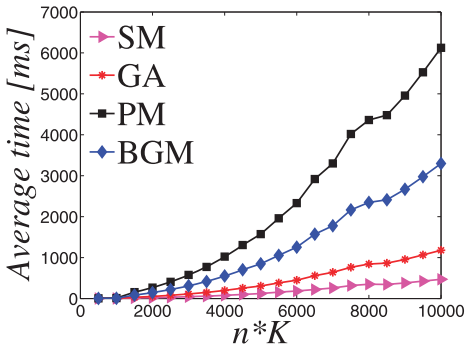


Fig. 8. Timing results for the matching schemes. We measure the run time with respect to the dimensions of the matching problem $n \cdot K$. We compared the Spectral Matching [21], Graduated Assignment [15], proposed Probabilistic Matching, and Balanced-Graph-Matching [9].

efficient. Note that even for the problem of largest dimensions $n_1 K = 10{,}000$, the running time of our approach is just ≈ 5 s, and can be significantly reduced by an optimized C++ implementation.

6 Summary and Discussion

In this paper, we presented a probabilistic interpretation of the spectral matching approach of Leordeanu and Hebert [21]. It is shown to be a maximum-likelihood estimate of the assignment probabilities, while assuming that the affinity matrix is an estimate of the joint assignment probability. We further analyze the GA algorithm of Gold and Rangarajan [15] and show it to be a MAP extension of the SM approach (although it precedes the SM chronologically). Based on our probabilistic analysis, we propose a spectral assignment ranking scheme that allows us to choose the most reliable matchings. We introduce a novel probabilistic formulation of quadratic matching that relaxes some of the assumptions used by the SM scheme. Our approach is experimentally shown to outperform previous schemes, especially as the matching problem becomes difficult due to the presence of noise and outliers.

In the future, we aim to reformulate the SM scheme by recalling that the ML estimate is computed via a Rank-One-Approximation (ROA) that is optimal with respect to the Frobenius norm. Computing the ROA with respect to a Bregman measure seems more appropriate and might result in improved results. The PM approach also seems promising for analyzing general graphs where the adjacency matrix is binary-valued. Such graphs are common in domains such as protein structure prediction and the analysis of social and biological networks.

Acknowledgments

The authors would like to thank the associate editor, Professor Marina Meila, and the anonymous reviewers for their valuable comments and constructive suggestions that improved the quality and clarity of the paper.

References

[1] S. Belongie, J. Malik, and J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 4, pp. 509-522, Apr. 2002.

[2] A.C. Berg, T.L. Berg, and J. Malik, "Shape Matching and Object Recognition Using Low Distortion Correspondences," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 26-33, June 2005.

[3] Y. Boykov and V. Kolmogorov, "An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 9, pp. 1124-1137, Sept. 2004.

[4] T. Caetano, J. McAuley, L. Cheng, Q. Le, and A. Smola, "Learning Graph Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 31, no. 6, pp. 1048-1058, June 2009.

[5] M. Chertok and Y. Keller, "Efficient High Order Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 32, no. 12, pp. 2205-2215, Dec. 2010.

[6] M. Chertok and Y. Keller, "Spectral Symmetry Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 32, no. 7, pp. 1227-1238, July 2010.

[7] D. Conte, P. Foggia, C. Sansone, and M. Vento, "Thirty Years of Graph Matching in Pattern Recognition," *Int'l J. Pattern Recognition and Artificial Intelligence,* vol. 18, no. 3, pp. 265-298, 2004.

[8] T. Cour and J. Shi, "Solving Markov Random Fields with Spectral Relaxation," *J. Machine Learning Research—Proc. Track,* vol. 2, pp. 75-82, 2007.

[9] T. Cour, P. Srinivasan, and J. Shi, "Balanced Graph Matching," *Advances in Neural Information Processing Systems 19,* B. Schölkopf, J. Platt, and T. Hoffman, eds., pp. 313-320, MIT Press, 2007.

[10] A.D. 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.

[11] S. Dasgupta, C.H. Papadimitriou, and U. Vazirani, *Algorithms.* McGraw-Hill, Inc., 2008.

[12] O. Duchenne, F. Bach, I. Kweon, and J. Ponce, "A Tensor-Based Algorithm for High-Order Graph Matching," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* June 2009.

[13] A. Egozi, Y. Keller, and H. Guterman, "Improving Shape Retrieval by Spectral Matching and Meta Similarity," *IEEE Trans. Image Processing,* vol. 19, no. 5, pp. 1319-1327, May 2010.

[14] M.R. Garey and D.S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness.* W.H. Freeman & Co., 1990.

[15] 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.

[16] S. Gold, A. Rangarajan, C.-P. Lu, S. Pappu, and E. Mjolsness, "New Algorithms for 2D and 3D Point Matching: Pose Estimation and Correspondence," *Pattern Recognition,* vol. 31, no. 8, pp. 1019-1031, 1998.

[17] G.H. Golub and C.F. van Loan, *Matrix Computations,* third ed. Johns Hopkins Univ. Press, 1996.

[18] J.H. Hays, M. Leordeanu, A.A. Efros, and Y. Liu, "Discovering Texture Regularity via Higher-Order Matching," *Proc. Ninth European Conf. Computer Vision,* pp. 522-535, May 2006.

[19] R.A. Hummel and S.W. Zucker, "On the Foundations of Relaxation Labeling Processes," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 5, no. 3, pp. 267-287, May 1983.

[20] R. Johnson, "On a Theorem Stated by Eckart and Young," *Psychometrika,* vol. 28, no. 3, pp. 259-263, Sept. 1963.

[21] M. Leordeanu and M. Hebert, "A Spectral Technique for Correspondence Problems Using Pairwise Constraints," *Proc. 10thIEEE Int'l Conf. Computer Vision,* vol. 2, pp. 1482-1489, Oct. 2005.

[22] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int'l J. Computer Vision,* vol. 20, pp. 91-110, 2003.

[23] B. Luo and E. Hancock, "Structural Graph Matching Using the EM Algorithm and Singular Value Decomposition," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 10, pp. 1120-1136, Oct. 2001.

[24] E. Mjolsness, "Bayesian Inference on Visual Grammars by Neural Nets that Optimize," Report YALEU/DCS/TR- 854, May 1991.

[25] J. Munkres, "Algorithms for the Assignment and Transportation Problems," *J. SIAM,* vol. 5, no. 1, pp. 32-38, 1957.

[26] G.L. Nemhauser and L.A. Wolsey, *Integer and Combinatorial Optimization.* John Wiley and Sons, 1988.

[27] A. Raj and R. Zabih, "A Graph Cut Algorithm for Generalized Image Deconvolution," *Proc. 10th IEEE Int'l Conf. Computer Vision,* pp. 1048-1054, 2005.

[28] A. Rangarajan and E. Mjolsness, "A Lagrangian Relaxation Network for Graph Matching," *IEEE Trans. Neural Networks,* vol. 7, no. 6, pp. 1365-1381, Nov. 1996.
[29] A. Rangarajan, A. Yuille, and E. Mjolsness, "Convergence Properties of the Softassign Quadratic Assignment Algorithm," *Neural Computation,* vol. 11, no. 6, pp. 1455-1474, 1999.
[30] C. Rother, V. Kolmogorov, and A. Blake, "'Grabcut' Interactive Foreground Extraction Using Iterated Graph Cuts," *Proc. ACM Siggraph '04,* pp. 309-314, 2004.
[31] G. Scott and H. Longuet-Higgins, "An Algorithm for Associating the Features of Two Images," *Royal Soc.,* vol. B-244, pp. 21-26, 1991.
[32] R. Sinkhorn and P. Knopp, "Concerning Nonnegative Matrices and Doubly Stochastic Matrices," *Pacific J. Math.,* vol. 21, no. 2, pp. 343-348, 1967.
[33] L. Torresani, V. Kolmogorov, and C. Rother, "Feature Correspondence via Graph Matching: Models and Global Optimization," *Proc. 10th European Conf. Computer Vision,* pp. 596-609, 2008.
[34] B. van Wyk and M. van Wyk, "A POCS-Based Graph Matching Algorithm," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 11, pp. 1526-1530, Nov. 2004.
[35] H.F. Wang and E.R. Hancock, "Correspondence Matching Using Kernel Principal Components Analysis and Label Consistency Constraints," *Pattern Recognition,* vol. 39, no. 6, pp. 1012-1025, 2006.
[36] A.L. Yuille and A. Rangarajan, "The Concave-Convex Procedure," *Neural Computation,* vol. 15, no. 4, pp. 915-936, 2003.
[37] R. Zass and A. Shashua, "Probabilistic Graph and Hypergraph Matching," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 1-8, June 2008.

Amir Egozi received the MSc degree in computer and electrical engineering from Ben-Gurion University, Beer-Sheva, Israel, in 2007, where he is currently pursuing the PhD degree. His research interests include computer vision, shape analysis, graph theory, and assignment problems. He is a student member of the IEEE.

Yosi Keller received the BSc degree in electrical engineering from the Technion-Israel Institute of Technology, Haifa, and the MSc and PhD degrees in electrical engineering from Tel-Aviv University, Tel-Aviv, in 1994, 1998, and 2003, respectively. From 1994 to 1998, he was an R&D officer in the Israeli Intelligence Forces. From 2003 to 2006, he was a Gibbs Assistant Professor with the Department of Mathematics, Yale University. He is a senior lecturer in the Faculty of Engineering at Bar Ilan University, Israel. His research interests include graph-based data analysis, optimization, and spectral graph theory-based dimensionality reduction.

Hugo Guterman received the electronic engineering degree from the National Technological University of Buenos Aires, Argentina, and the MSc and PhD degrees in computer and electrical engineering from Ben-Gurion University, Beer-Sheva, Israel, in 1978, 1982, and 1988, respectively. From 1988 to 1990, he was a postdoctoral fellow at MIT. Since 1990, he has been in the Department of Computer and Electrical Engineering at Ben-Gurion University. His research interests include control, image and signal processing, neural networks and fuzzy logic, electrochemical processes, robotics, biotechnology, and biosensors. He is a member of the IEEE.