A Short Survey of Recent Advances in Graph Matching

Junchi Yan*
East China Normal University
IBM Research – China
jcyan@sei.ecnu.edu.cn

Xu-Cheng Yin
University of Science and Technology Beijing
xuchengyin@ustb.edu.cn

Weiyao Lin
Shanghai Jiao Tong University
wylin@sjtu.edu.cn

Cheng Deng
Xidian University
chdeng@mail.xidian.edu.cn

Hongyuan Zha
Georgia Institute of Technology
zha@cc.gatech.edu

Xiaokang Yang
Shanghai Jiao Tong University
xkyang@sjtu.edu.cn

ABSTRACT

Graph matching, which refers to a class of computational problems of finding an optimal correspondence between the vertices of graphs to minimize (maximize) their node and edge disagreements (affinities), is a fundamental problem in computer science and relates to many areas such as combinatorics, pattern recognition, multimedia and computer vision. Compared with the exact graph (sub)isomorphism often considered in a theoretical setting, inexact weighted graph matching receives more attentions due to its flexibility and practical utility. A short review of the recent research activity concerning (inexact) weighted graph matching is presented, detailing the methodologies, formulations, and algorithms. It highlights the methods under several key bullets, e.g. how many graphs are involved, how the affinity is modeled, how the problem order is explored, and how the matching procedure is conducted etc. Moreover, the research activity at the forefront of graph matching applications especially in computer vision, multimedia and machine learning is reported. The aim is to provide a systematic and compact framework regarding the recent development and the current state-of-the-arts in graph matching.

Keywords

Survey; graph matching; correspondence; similarity

1. INTRODUCTION

Graph matching, which refers to a class of computational problems of finding an optimal correspondence between the vertices of graphs to minimize (maximize) their node and edge disagreements (affinities), has been a fundamental problem in computer science and relates to many areas such as pattern recognition, computer vision, graphics, bioinformatics, whereby correspondence and similarity between graphs are required. One main advantage for describing the relational information by attributed graphs instead of vectors is that graphs allow for a more powerful representation of structural relations. In the general case, nodes and edges are assigned with arbitrary attributes [124, 4]. This modeling is rather flexible from the application-oriented point of view, as most real-world graphs fall into this category.

The GM problem can be divided into two general categories: exact matching and inexact matching. In the former task, a strict correspondence is required to be found, or at least among their substructures. In the latter, this requirement is substantially relaxed to find the bijection between the vertex that optimizes a certain affinity or distortion criterion – thus is also called error-tolerant/correcting GM in literature [14, 13], since matchings between non-identical graphs need to be addressed for real-world problems.

Different from the point based registration methods, such as RANSAC [45] and Iterative Closet Point (ICP) [151], GM methods incorporate both the unary *node-to-node*, and the second-order, or even higher-order, *(hyper)edge-to-(hyper)edge* structural similarities. By encoding such geometrical cues in the graph representation and matching process, GM methods can in general find better correspondence. Although extensive research has been done, deriving optimal GM is still a challenging problem in both theory and practice: in deed the GM problem can be formulated as a quadratic assignment problem (QAP) [77], being well-known NP-complete [49].

To our surprise, there is scarcely any focused whole-picture of up-to-date development on inexact weighted graph matching though it has already attracted considerable attentions due to its flexibility in modeling real-world problems.

Several surveys e.g. [15, 30, 46, 125] involve the broad literature on graph-based techniques, especially in the context of pattern recognition. They cover various areas beyond weighted graph matching, e.g. graph kernel, graph embedding, graph clustering, while devote only a minor portion of the space to the recent development of GM especially in computer vision and machine learning. For instance, despite the title of the survey [76], it is mostly dedicated to graph embedding and graph kernels. These concepts are related to GM as they help evaluate the similarity among graphs, but they are out of the scope of this survey. There are also

*Corresponding author. This work is partially supported by the Shanghai Science and Technology Committee (15JC1401700, 14XD1402100) NSFC 61471235 and Microsoft Research Asia Collaborative Research Award.

ICMR'16, June 06-09, 2016, New York, NY, USA

DOI: http://dx.doi.org/10.1145/2911996.2912035

a few specific reports on one or a few small areas related to GM. [106] presents a comparative evaluation of four exact algorithms for graph isomorphism and graph-subgraph isomorphism. [48] focuses on the graph edit distance based methods for both attributed and non-attributed graphs. On the other hand, are reviews on the applications of GM to computer vision and pattern recognition [31, 126]. However, many recent work studied in this paper are not covered in these surveys. One focused review on inexact weighted graph matching [131], which is the emphasis of this survey, is dated back one decade ago [18].

This paper is aimed to address the above gap between a too broad (but often rather general) survey and a too narrow one, especially for those emerging topics in GM, and the vitality of this survey involves two main regards:

i) It provides a systematic view of looking at the recent development on (inexact) weighted graph matching, concerning how the problem is formulated, two or multiple graphs are involved, how the affinity is modeled etc. Most of these perspectives are fresh compared with existing surveys, which we believe relates to the recent emerging research advances.

ii) This paper involves more recent GM works, especially from the computer vision and machine learning community. A majority of these works have not been covered in [125, 46, 76, 126], perhaps due to these existing surveys often focus on the area of pattern recognition. This survey aims to strengthen the tie between different communities.

The survey is organized as follows. We first give a brief on exact graph matching, which is commonly regarded too idealistic for practical problems. Then we are devoted to the recent advance in inexact weighted graph matching – a more realistic setting. It is followed by a study on the applications especially in multimedia and a conclusive remark.

2. BRIEF ON EXACT GRAPH MATCHING

Classical exact graph matching algorithms employ the concepts of graph and subgraph isomorphisms, such that requires a bijective node mapping between two (sub)graphs, and preserves the edges of both graphs with zero-distortion, which is termed isomorphism. Graph isomorphism, which involves to verify whether a zero-distortion node mapping exists, is known to be in NP, neither known to be in P nor NP-complete [49]. There are special structure where graph isomorphism can be solved in polynomial time. For instance, checking the isomorphism between planar graphs [42], or graphs with bounded vertex degree [78], is known to be solvable in a linear time .

One weaker form of exact GM is subgraph isomorphism, which requires the existence of an isomorphism between one graph and the subgraph of the other. Subgraph isomorphism is known to be NP-complete [32]. One more practical, while still NP-hard problem is the maximum common subgraph (MCS) [68]. It is aimed to find the largest subgraph in one graph that is isomorphic to a subgraph in the other.

3. INEXACT GRAPH MATCHING

GM is often integrated with an 'inexact' flavor to account for ubiquitous noises, involving finding a correspondence concerning minimum rather than zero-distortion, or equivalently a maximum affinity, when exact matching is impossible. In such cases, the concept of inexact [7], also termed as error-correcting [14] or error-tolerant [13] matching, is called for. Specifically, this survey pays more attention on the general quadratic assignment programming based formulations and the variants for weighted graphs [131], though there are similar forms under different residual norms [124, 4].

3.1 Some basic formulations

3.1.1 Second-order affinity matrix based model

Most GM methods try to find correspondences between two graphs e.g. [66, 50, 25, 17, 41]. In general, the two-graph matching problem can be formulated as a quadratic assignment problem (QAP). As a most general form, the Lawler's QAP [60] to maximize the affinity score is:

$$J(\mathbf{X}) = \mathrm{vec}(\mathbf{X})^\top \mathbf{K} \mathrm{vec}(\mathbf{X}) \quad (1)$$
$$\mathbf{X}\mathbf{1}_{n_2} = \mathbf{1}_{n_1}, \mathbf{X}^T\mathbf{1}_{n_1} \leq \mathbf{1}_{n_2}$$

where $\mathbf{X}$ refers to the assignment matrix between two graphs. The assignment constraint is one typical setting in GM. Here it refers to the two-way one-to-one node mapping: a node from graph $\mathcal{G}_1$ of node size n_1 can match at most one node in $\mathcal{G}_2$ of size n_2 and every node in $\mathcal{G}_2$ is corresponding to one node in $\mathcal{G}_1$. There is no (one/many)-to-many matchings between two graphs, though there is a line of studies on these more general cases [3, 145, 115, 37, 159] beyond the one-to-one mapping constraint. $\mathbf{K}$ is the second-order affinity matrix. A large portion of recent GM works including both two-graph matching e.g. [50, 25, 154] and multi-graph matching e.g. [135, 136] adopt the Lawler's formula.

The Koopmans-Beckmann's QAP [77] is another QAP form for two-graph matching, which can be written by

$$J(\mathbf{X}) = \mathrm{tr}(\mathbf{X}^\top \mathbf{F}_i \mathbf{X} \mathbf{F}_j) + \mathrm{tr}(\mathbf{K}_p^\top \mathbf{X}) \quad (2)$$

The vector $\mathbf{F}_i$, $\mathbf{F}_j$ are the weighted adjacency matrices and $\mathbf{K}_p$ is the node-to-node similarity matrix. Note the Koopmans-Beckmann's QAP can always be represented as a special case of the Lawler's by setting $\mathbf{K} = \mathbf{F}_j \otimes \mathbf{F}_i$.

3.1.2 Factorized affinity matrix

There are recent advances in the interpretation of the affinity matrix. [154] show how to factorize the affinity matrix as a Kronecker product of smaller matrices. One formulation for the undirected graphs is written by:

$$\mathbf{K} = (\mathbf{H}_2 \otimes \mathbf{H}_1)\mathrm{diag}(\mathrm{vec}(\mathbf{L}))(\mathbf{H}_2 \otimes \mathbf{H}_1)^T \quad (3)$$
$$\text{where} \quad \mathbf{H}_i = [\mathbf{G}_i, \mathbf{I}_{n_i}] \in \{0,1\}^{n_i \times (m_i + n_i)}, \quad i = 1, 2$$
$$\mathbf{L} = \begin{bmatrix} \mathbf{K}^q & -\mathbf{K}^q\mathbf{G}_2^T \\ -\mathbf{G}_1\mathbf{K}^q & \mathbf{G}_1\mathbf{K}^q\mathbf{G}_2^T + \mathbf{K}^p \end{bmatrix}$$

where n_i and m_i is the number of nodes and edges in graph i respectively and $\otimes$ is the Kronecker product operation between matrices. $\mathbf{K}^p \in \mathbb{R}^{n_1 \times n_2}$ denotes the node affinity matrix, and $\mathbf{K}^q \in \mathbb{R}^{m_1 \times m_2}$ for the edge affinity matrix. The graph structure is specified by the node-edge incidence matrix $\mathbf{G} \in \mathbb{R}^{n \times m}$ such that the non-zero elements in each column of $\mathbf{G}$ indicate the starting and ending nodes in the corresponding edge. The factorization provides a taxonomy for GM and reveals the connection among several methods. Readers are referred to [154] for greater details.

3.1.3 Higher-order affinity tensor based model

Based on tensor marginalization as adopted by several hypergraph matching works [148, 21, 39, 139]:

$$\mathbf{x}^* = \arg\max(\mathbf{H} \otimes_1 \mathbf{x} \otimes_2 \mathbf{x} \dots \otimes_m \mathbf{x}) \quad s.t. \qquad (4)$$
$$\mathbf{X}\mathbf{1}_{n_2} = \mathbf{1}_{n_1}, \mathbf{X}^T\mathbf{1}_{n_1} \leq \mathbf{1}_{n_2}, \mathbf{x} = \mathrm{vec}(\mathbf{X}) \in \{0,1\}^{n_1 n_2 \times 1}$$

where m is the affinity order and $\mathbf{H}$ is the m-order affinity tensor whose element encodes the affinity between two hyperedges from the graphs. $\otimes_k$ is the tensor product [63]. Readers are referred to Sec. 3.1 in [39] for details on tensor multiplication. The above works all assume the affinity tensor is invariant w.r.t. the index of the hyperedge pairs.

3.1.4 Graph edit distance based graph matching

[102] firstly introduces the graph edit distance (GED), as an important way to measure the similarity between pairwise graphs. GED is defined from the notion of edit path which corresponds to a sequence of elementary transformations of a graph into another. An standard edit operation, associated with non-negative value, is a transformation performed on the structure of a graph, restricted to be elementary: node or edge insertion, removal and substitution. The cost of the edit path is defined as the sum of all its elementary operation's costs [14] and the GED distance between two graphs can then be defined as the minimum cost required to transform one of the given graphs into the other. Computing the GED is known NP-complete and a classic method is by means of a tree search procedure that basically evaluates all possible node-to-node correspondences [102].

A number of approximate methods have been proposed to render the computation of graph edit distance feasible [48]. Among these, we examine two approximate formulations. One is called bipartite GED [98, 99] solved by linear assignment; the other is based on the quadratic assignment formulation [88, 10], which transforms the GED problem to a graph matching one, though computing the GED is generally not equivalent to solving a graph matching problem.

Bipartite GED [98] and its variants e.g. [109, 110] approximate the GED problem by a linear assignment problem, which can be solved efficiently via e.g. the Hungarian method [85]. These methods approximate graph structure by a node-to-node cost matrix that encodes local clique structure. However, edit operations on nodes and edges are not handled simultaneously, which limits the efficacy of such a first-order approximation strategy.

QAP based GED To improve the above Bipartite GED models, [88] firstly shows that the GED problem can be formalized as a Quadratic Assignment Programming problem, through the definition of fuzzy paths. In the more recent work, [10] then gives a further formal and more general analysis on how and in what conditions the GED can be transformed to a QAP form, which is directly related to weighted GM solvers e.g. [65] as used in [10].

3.2 Affinity/cost modeling

3.2.1 Attribute/affinity learning

A large number of GM methods are fed with a manually-predefined affinity matrix/tensor and keep it unchanged during the entire matching process. This would be too restricted to cope with noisy real-world data. Recent work leverage various leaning algorithms for computing the optimal affinity matrix [66, 17, 67, 23, 54], and these methods in general fall into either supervised [17], unsupervised [66], or semi-supervised [67] learning paradigm, based on to what extent the supervision information is used.

The authors in [23] present a unified parameterized graph structure learning model, and solve it in a max-margin framework. They define a joint feature map by the vector form:

$$\Phi(\mathcal{G}, \mathcal{G}', \pi) = [\cdots, s_v(a_i, a_{\pi(i)}), \cdots, s_e(a_{ij}, a_{\pi(i)\pi(j)}) \cdots]^T$$

By introducing weights on all elements of this feature map, one obtains a score function: $S(\mathcal{G}, \mathcal{G}', \pi, \beta) = \beta\Phi(\mathcal{G}, \mathcal{G}', \pi)$ where β is a weight vector of the node and edge similarity.

As observed by [23], many previous learning based GM methods can be viewed as a special case for the above formulation. Specifically, [17] uses a 60-dimensional similarity function s_v for appearance similarity and a simple binary similarity s_e for edges. [123] adopts 2-dimensional s_v and s_e functions to measure appearance similarity, geometric compatibility and occlusion. [66] employs a multi-dimensional s_e to model similarity without considering s_v.

3.2.2 Graph edit cost learning

In [14], the authors theoretically prove that the graph edit costs is critical to the performance of the GED based methods. In complex tasks a manual procedure for cost setting is difficult, or even impossible to apply. To address this issue, [87] aims to learn the edit cost by a probabilistic framework, to reduce the intra-class edit distance and increase the inter-class one. Alternatively, [86] proposes to use self-organizing maps to learn the edit cost.

3.3 Joint matching multiple graphs

Dating back to the simpler setting of matching a set of point sets based on their node-wise attributes[95, 36], here is an emerging line of work for devising advanced multi-graph matching methods [117, 55, 93, 135, 118, 134, 20, 114], because directly using on-the-shelf two-graph solvers is incapable of fully utilizing the cross-graph information of affinity and matching consistency. Most multi-graph matching methods involve the following aspects:

i) One limitation of employing a two-graph matching method for multi-graph is that the so-called cycle-consistency need additional care. Consider one toy example, for graphs $\mathcal{G}_i$, $\mathcal{G}_j$, $\mathcal{G}_a$, $\mathcal{G}_b$ of equal size with no outliers, the four matching solutions between two graphs $\mathbf{X}_{ij}$, $\mathbf{X}_{ia}$, $\mathbf{X}_{aj}$, $\mathbf{X}_{ib}$, which are computed by two-graph matching independently, can lead to cycle-inconsistency: $\mathbf{X}_{ia}\mathbf{X}_{aj} \neq \mathbf{X}_{ib}\mathbf{X}_{bj}$ [135].

Many multi-graph matching models capture the above consistency often in an iterative or a one-shot setting:

Iterative methods aim to maximize the affinity and meanwhile account for consistency. [135, 136] enforce the cycle-consistency by obeying the strict equality constraints $\mathbf{X}_{ij} = \mathbf{X}_{ib}\mathbf{X}_{bj}$ over the whole iterative variable updating procedure. The performance is often sensitive to the starting solution and variable rotating order – inherently vulnerable to error accumulation. Similar hard constraint is devised by [118] which extends the Graduated Assignment algorithm for two-graph matching [50] to multi-graph one. [134, 133] propose a more flexible and robust mechanism, where they gradually infuse consistency over iterations.

One-shot methods try to enforce overall consistency immediately. [93, 20, 158] achieve consistency as a post-step whose inputs are the putative two-graph matchings $\{\mathbf{X}\}_{i,j=1}^N$ obtained by a two-graph matching solver. Since no reinforce-

ment is performed to boost the affinity, these methods are found often sensitive to the quality of initial inputs.

ii) In a line of recent work [118, 135, 136, 134, 133], the authors write out the objective for multi-graph matching by adding up pairwise affinity terms $\{\text{vec}(\mathbf{X})^\top \mathbf{K} \text{vec}(\mathbf{X})\}_{i,j=1}^N$, and the two-graph matchings $\{\mathbf{X}\}_{i,j=1}^N$ are iteratively updated such that strictly, or gradually satisfy the cycle-consistency constraints. While for the two non-iterative methods [93, 20], affinity information is only considered locally for a pair of graphs at hand. The affinity information is ignored in their post-step where consistency is enforced.

Based on the above two groups of methods, the authors in [138] make two observations: first, cycle-inconsistency is caused by the use of two-graph matchings – a distributed and redundant representation for multi-graph matching; second, for affinity modeling, existing multi-graph matching methods are mostly based on the two-graph affinity representation. Motivated by this, they present a novel formulation where the affinity among graphs is encoded by a matrix stacked by the vectorized attributes of graphs, and the variables are reduced to a set of non-redundant bases inherent free from the consistency constraint.

3.4 Higher-order graph matching

Higher-order attributes for matching hypergraphs have recently drawn considerable attentions, with the aim of improving the matching robustness by incorporating higher-order cues. Third-order models, for instance using the angles of sampled triplets [39, 63, 139], are often explored in for their tradeoff between efficiency and modeling capability.

The approximate linear assignment framework is often inherited for extending the second-order methods [50, 64, 65, 25, 121] to their hypergraph variants [19, 39, 63, 139]. In the seminal work, [39] resorts to a tensor to encode the higher-order affinity to generalize second-order to higher-order. They also extend the spectral matching method [64] via a multi-dimensional power method [51]. Re-weighted Random Walks is devised for second-order [25], and subsequently for higher-order [63]. Chang and Kimia [19] generalize GAGM [50] to the higher-order. The probabilistic method [148] assumes the higher-order correspondences are conditional independence. This hurts its applicability and performance [63]. [139] generalize the convergence analysis and algorithm from the second-order [121] to higher-order. The local property of graph structure is taken special care in [156] such that the matching accuracy is improved.

There are also specialized hypergraph matching solvers [8, 94]. The former is tailored to finding 3-D and 2-D projection correspondence. The latter assumes the affinity tensor is redundant as it requires there are many tuples of feature points whose corresponding angles are close to each other.

Another interesting work [89] proposes a novel flexible tensor block coordinate ascent method for hypergraph matching, with the guarantee of monotonic ascent in the matching score on the set of discrete assignment matrices.

3.5 Recent representative solvers

We refer the readers to a partial review of recent GM methods, of which most are after year 2000, under several bullets to highlight the main characters. Note that due to the complexity and non-triviality of GM algorithms, they are often mixed by multiple features, and the bullets below are not strictly orthogonal to each other in algorithm design.

3.5.1 (Quasi-)discrete methods

Several methods tend to directly compute the solution in discrete assignment space. Integer Projected Fixed Point (IPFP) [65] is one of the most popular (quasi) discrete graph matching solvers in this spirit. By a similar first-order linearization idea, [139] proposes a full discrete method for hyper-graph matching. [1] devises a tailored Tabu search for graph matching. Another line of work are sampling based methods which directly generate discrete solutions: the Markov chain Monte Carlo model [62] and its efficient extension based on Sequential Monte Carlo Sampling [119].

3.5.2 Spectral, SDP and double-stochastic relaxation

There are different relaxation strategies used in graph matching to mitigate the hard combinatorial problem in nature, and enable practitioners to bring gradient-descent-type algorithms to bear [80]. Here we briefly mention three popular relaxation methodologies.

Spectral matching methods e.g. [57] are based on the fact that the eigenvalues of a matrix keep unchanged whatever the rows and columns are permuted, which are often efficient but more sensitive to noises. [64] proposes to relax the assignment matrix into a unit vector, and the relaxed solution can be obtained by computing the maximum eigen vector of the affinity matrix. [35] uses the affine constrain to better approximate the raw problem.

Semidefinite-programming (SDP) is a general tool for combinatorial problems, and SDP methods [122, 107] approximate GM by relaxing the non-convex constraint to a new one: $\mathbf{Y} = \text{vec}(\mathbf{X})\text{vec}(\mathbf{X})^T$ where $\mathbf{Y}$ is an introduced variable, to a convex semi-definite one: $\mathbf{Y} - \text{vec}(\mathbf{X})\text{vec}(\mathbf{X})^T \succeq 0$. Given $\mathbf{Y}$, $\mathbf{X}$ can be obtained by a winner-take-all strategy [107], or a randomized algorithm [122]. Though there is some theoretical bound on its approximation capability, while in practice it is often computationally expensive.

Most GM methods adopt the doubly-stochastic relaxation on $\mathbf{X}$ – the convex hull on the assignment matrix. GM is transformed to convex quadratic programming that enable different solvers e.g. [50, 64, 65]. However, the applicability of such a relaxation is poorly understood until the recent theoretical findings [80, 2]. Among them, a line of methods adopt a probabilistic paradigm. [41] presents a method for maximum-likelihood estimation of the assignment matrix, as an probabilistic interpretation of the spectral matching algorithm. [52, 25] adopts random walk based methods which are inspired from PageRank. The work [148] proposes a convex relative-entropy error from a probabilistic interpretation to the hypergraph matching problem.

3.5.3 Continuation and path-following methods

One thread of these work adopt a path-following strategy [147, 74, 73, 154, 155]. The authors in [154, 155] propose an affinity matrix factorization technique, based on which a convex-concave relaxation procedure controlled by a continuation parameter is devised, and it is iteratively solved by an modified Frank-Wolfe algorithm [47]. [147, 74] devise similar path-following methods independent of the factorization form of affinity matrix. [73, 75] further propose several implicit convex-concave path-following algorithms, which are more flexible because it does not require the two graphs are of equal size and can handle partial permutation matrix.

Graduated assignment [50] and its variants [19, 118] are another line of continuation methods where a deterministic

annealing procedure is performed in the continuous space. Though in general its convergence cannot be guaranteed [121], while adding an appropriate identity matrix to the affinity matrix can ensure it converges to a fixed point in the assignment space as proved in [97] by constructing discrete time Lyapunov functions. [139] generalizes this to tensor.

3.6 Other techniques

3.6.1 Progressive methods

Recently, progressive methods which iteratively enrich the candidate matches and reject the wrong ones have attracted a lot of attentions due to its high precision/recall and efficiency. [26] propagates the matches to neighbor points and refine the candidates by graph matching, where the former part promotes recall and the latter boosts the precision. In [143], a geometric consistency based on a high-order affine invariant constraint is established and a Markov Random Field optimization encoding both appearance and geometric cues is used to reject mismatches. [127] addresses the challenges of outliers and many-to-many node correspondence by a density maximization framework which maximizes the values of a proposed graph density estimator both locally and globally. [84] also proposes a progressive framework by hierarchically matching the sub-structures of the graphs.

3.6.2 Integrated matching pipeline

There are several integrated systems where GM is adopted. [123] allows nodes to be assigned in an unmatchable status in an energy function induced on the Markov Random Field. However, its complex objective function is designed that account for various similarity measurements *e.g.* appearance descriptors, occlusions, spatial proximity. [29] integrates the point detection and matching synergically, which shows robustness to visual problems by fusing various information.

3.6.3 Miscellaneous methods

In the recent work, [90] proposes the clique-graph, and a clique-graph matching method by preserving global and local structures. To address the problem of outliers, [27] presents a max-pooling based approach which is effective in handling massive outliers in one graph. In [79], the authors devise a seeded graph matching model embedding the graphs into a common Euclidean space and matching across embedded graphs, derived from the Joint Optimization of Fidelity and Commensurability [96] algorithm. In particular, [3] offers a game-theoretic perspective that formulates the matching problem as a non-cooperative game. This mechanism enable its capability of dealing with the general many-to-many matching problem. Their method is further improved in [12] by a fast evolutionary approach.

3.6.4 Relation to point registration

For point set registration, the goal is usually finding the correspondence and the geometric transformation between points such that the sum of distances between aligned point sets is minimized. One fundamental difference between GM and point set registration [151, 137] is that the former is a non-parametric model in terms of the transformation between two point sets while the latter assumes a parametric transformation function. The parameterization can vary dependent on the specific problem, such as similarity, affine and RBF non-rigid transformation as used in 2-D space.

Recently, by the factorization formulation, [154] shows that the point registration problem is a special case for the GM problem that only involve the linear term, when the transformation regularization term is given.

There is also a trend for the mixture of graph matching models and registration methods, as evidenced from the recent studies [103, 104, 137, 154].

4. APPLICATION AND DATASET

This section provides a study on several representative areas, and in particular for multimedia retrieval and analysis.

4.1 Applications

Multimedia and computer vision Graph representation of the objects that flexibly captures the geometrical characteristics, calls for GM techniques: image registration [112], understanding [43], extrapolation [128] and recognition [7], object recognition [40] and tracking [91], scene understanding [129] and parsing [144], weak-perspective 3-D reconstruction [158], action recognition [142], robotics [105], video surveillance [72] and person re-identification [113].

In particular, there are various multimedia retrieval applications where graph matching is intensively used, such as motion retrieval [132], object retrieval in video [22] and image [61], content-based image retrieval [9, 53], near-duplicate image detection [149, 152], shape retrieval [56], satellite imagery retrieval [59], CAD component detection [33] etc.

Other scenarios [120] releases a subgraph matching tool for biological graphs. [71] applies the spectral method to the problem of aligning multiple protein networks, and metabolic networks for [141]. [146] formulates to a GM problem and solves it by a path-following algorithm. [44] uses GM and sparse coding to infer the brain connectivity from alignment-free functional magnetic resonance imaging (fMRI) data.

We collect a partial list of other applications: grid anomaly detection [5], business process management [38], schema [81] and RDF [11] matching, network monitoring [116], case based reasoning [101], task assignment [111], information fusion [130], document classification [108], semantic search [153], assembling [6], product design [100], among others.

4.2 Public datasets

There are several benchmark datasets available online like: i) CMU motion sequences [154, 135]; ii) Willow-ObjectClass dataset collected and annotated by [23]; iii) QAPLib benchmark dataset [16]; iv) Tarragona Databases [34, 83]; v) human connectomics by DT-MRI scans as used in [80]; vi) Fish and character shape dataset from UCF shape dataset [28] which has been historically widely used for ICP registration algorithms, but recently also used in GM [137, 154].

5. CONCLUSION AND OUTLOOK

We have presented a focused survey on various graph matching methods, whereby methods are categorized by different topics. In particular, it covers many recent papers in the computer vision and machine learning communities.

Looking forward, we think the scalability of GM methods is the key issue for its real-world impact. This is because not only in multimedia applications, but also a wide spectrum of miscellaneous problems e.g. heterogenous social network [150], protein networks [146], smart grids [5], CAD [92], head [140] and face [70, 69] where graph representation is natural,

call for scalable GM algorithms. The matching procedure is also expected to handle the dynamic change of the graph, where dynamic assignment algorithms e.g. [82] are suited.

Back to multimedia and computer vision, one specific direction is developing principled methods for weakly-supervised object discovery and localization in images [24]. Another potential is developing tailored GM method integrated in the system for addressing the object (weak) reconstruction from multiple images where several recent advancements have been made [158, 58]. For social media, we are witnessing the emerging problems for aligning multiple heterogenous social network [150] for cross-network data mining, and the behavior data-generated diffusion network structure [157].

6. REFERENCES

[1] K. Adamczewski, Y. Suh, and K. Lee. Discrete tabu search for graph matching. In *ICCV*, 2015.

[2] Y. Aflalo, A. Bronstein, and R. Kimmel. On convex relaxation of graph isomorphism. *Proceedings of the National Academy of Sciences*, 112(10):2942–2947, 2015.

[3] A. Albarelli, S. Bulo, A. Torsello, and M. Pelillo. Matching as a non-cooperative game. In *ICCV*, pages 1319–1326, 2009.

[4] H. A. Almohamad and S. O. Duffuaa. A linear programming approach for the weighted graph matching problem. *PAMI*, 1993.

[5] A. Anwar and A. N. Mahmood. Anomaly detection in electric network database of smart grid: Graph matching approach. *Electric Power Systems Research*, 133:51–62, 2016.

[6] A.S.Deshmukh, A. G. Banerjee, S. GuPta, and R. Sriram. Content-based assembly search: A step towards assembly reuse. *ComPuter-Aided Design*, pages 244–261, 2008.

[7] S. Auwatanamongkol. Inexact graph matching using a genetic algorithm for image recognition. *Pattern Recognition Letters*, 28(12):1428–1437, 2007.

[8] M. Bansal and K. Daniilidis. Geometric polynomial constraints in higher-order graph matching. *arXiv preprint arXiv:1405.6261*, 2014.

[9] S. Berretti, A. D. Bimbo, and E. Vicario. Efficient matching and indexing of graph models in content-based retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1089–1105, 2001.

[10] S. Bougleux, L. Brun, V. Carletti, P. Foggia, B. Gaüzère, and M. Vento. A quadratic assignment formulation of the graph edit distance. *arXiv preprint arXiv:1512.07494*, 2015.

[11] M. Brocheler, A. Pugliese, and V. Subrahmanian. Dogma: A disk-oriented graph matching algorithm for rdf database. In *ISWC*, 2009.

[12] S. R. Bulò, M. Pelillo, and I. M. Bomze. Graph-based quadratic optimization: A fast evolutionary approach. *Computer Vision and Image Understanding*, 115(7):984–995, 2011.

[13] H. Bunke. Error-tolerant graph matching: a formal framework and algorithms. In *Advances in Pattern Recognition*, pages 1–14. 1998.

[14] H. Bunke. Error correcting graph matching: on the influence of the underlying cost function. *PAMI*, 21(9), 1999.

[15] H. Bunke. Graph matching: theoretical foundations, algorithms, and applications. In *Vision Interface*, 2000.

[16] R. Burkard, S. Karisch, and F. Rendl. Qaplib–a quadratic assignment problem library. *Journal of Global optimization*, 10(4):391–403, 1997.

[17] T. Caetano, J. McAuley, L. Cheng, Q. Le, and A. J. Smola. Learning graph matching. *TPAMI*, 31(6):1048–1058, 2009.

[18] R. M. Cesar, E. Bengoetxea, I. Bloch, and P. Larrañaga. Inexact graph matching for model-based recognition: Evaluation and comparison of optimization algorithms. *Pattern Recognition*, 38(11):2099–2113, 2005.

[19] M. C. Chang and B. B. Kimia. Measuring 3d shape similarity by graph-based matching of the medial scaffolds. *Computer Vision and Image Understanding*, pages 707–720, 2011.

[20] Y. Chen, G. Leonidas, and Q. Huang. Matching partially similar objects via matrix completion. In *ICML*, 2014.

[21] M. Chertok and Y. Keller. Efficient high order matching. *TPAMI*, 2010.

[22] F. Chevalier, J. Domenger, J. Benois-Pineau, and M. Delest. Retrieval of objects in video by similarity based on graph matching. *Pattern Recognition Letters*, 28(8):939–949, 2007.

[23] M. Cho, K. Alahari, and J. Ponce. Learning graphs to match. In *ICCV*, 2013.

[24] M. Cho, S. Kwak, C. Schmid, and J. Ponce. Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals. In *CVPR*.

[25] M. Cho, J. Lee, and K. M. Lee. Reweighted random walks for graph matching. In *ECCV*, 2010.

[26] M. Cho and K. M. Lee. Progressive graph matching: Making a move of graphs via probabilistic voting. In *CVPR*, 2012.

[27] M. Cho, J. Sun, O. Duchenne, and J. Ponce. Finding matches in a haystack: a max-pooling strategy for graph matching in the presence of outliers. In *CVPR*, 2014.

[28] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *CVIU*, 2003.

[29] T. Collins, P. Mesejo, and A. Bartoli. An analysis of errors in graph-based keypoint matching and proposed solutions. In *ECCV*, 2014.

[30] D. Conte, P. Foggia, C. Sansone, and M. Vento. Thirty years of graph matching in pattern recognition. *IJPRAI*, 2004.

[31] D. Conte, P. Foggia, C. Sansone, and M. Vento. How and why pattern recognition and computer vision applications use graphs. In *Applied Graph Theory in Computer Vision and Pattern Recognition*, pages 85–135. Springer, 2007.

[32] D. J. Cook and L. B. Holder. *Mining graph data*. John Wiley & Sons, 2006.

[33] L. Cordella, P. Foggia, C. Sansone, and M. Vento. Fast graph matching for detecting cad image components. In *ICPR*, 2000.

[34] X. Cortés and F. Serratosa. Learning graph-matching edit-costs based on the optimality of the oracle's node correspondences. *Pattern Recognition Letters*, 56:22–29, 2015.

[35] T. Cour, P. Srinivasan, and J. Shi. Balanced graph matching. In *NIPS*, 2006.

[36] S. Deb, M. Yeddanapudi, K. Pattipati, and Y. Bar-Shalom. A generalized sd assignment algorithm for multisensor-multitarget state estimation. *IEEE Transactions on Aerospace and Electronic Systems*, 33(2):523–538, 1997.

[37] M. F. Demirci, B. Platel, A. Shokoufandeh, L. L. Florack, and S. J. Dickinson. The representation and matching of images using top points. *Journal of Mathematical Imaging and Vision*, 35(2):103–116, 2009.

[38] R. Dijkman, M. Dumas, Z. Gao, and L. Garcia-Banuelos. Graph matching algorithms for business process model similarity search. In *Business Process Management*, pages 48–63. Springer, 2009.

[39] O. Duchenne, F. Bach, I. Kweon, and J. Ponce. A tensor-based algor ithm for high-order graph matching. *PAMI*, 2011.

[40] O. Duchenne, A. Joulin, and J. Ponce. A graph-matching kernel for object categorization. In *ICCV*, 2011.

[41] A. Egozi, Y. Keller, and H. Guterman. A probabilistic approach to spectral graph matching. *TPAMI*, 2013.

[42] D. Eppstein. Subgraph isomorphism in planar graphs and related problems. In *SODA*, 1995.

[43] M. Eshera and K.-S. Fu. An image understanding system using attributed symbolic representation and inexact graph-matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (5):604–618, 1986.

[44] M. Fiori, P. Sprechmann, J. Vogelstein, P. Muse, and G. Sapiro. Robust multimodal graph matching: Sparse coding meets graph matching. In *NIPS*, 2013.

[45] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 1981.

[46] P. Foggia, G. Percannella, and M. Vento. Graph matching and learning in pattern recognition in the last 10 years. *IJPRAI*, 33(1), 2014.

[47] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 1956.

[48] X. Gao, B. Xiao, D. Tao, and X. Li. A survey of graph edit distance. *Pattern Analysis and Applications*, 2010.

[49] M. R. Garey and D. S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. Freeman and Co., NY, USA, 1990.

[50] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *TPAMI*, 1996.

[51] G. Golub and C. Loan. Matrix computations. *Johns Hopkins University Press*, 1996.

[52] M. Gori, M. Maggini, and L. Sarti. Exact and approximate graph matching using random walks. *PAMI*, 2005.

[53] A. Hlaoui and S. Wang. A new algorithm for graph matching with application to content-based image retrieval. In *Structural, Syntactic, and Statistical Pattern Recognition*, pages 291–300. 2002.

[54] N. Hu, R. M. Rustamov, and L. Guibas. Graph mmatching with anchor nodes: a learning approach. In *CVPR*, 2013.

[55] Q. Huang, G. Zhang, L. Gao, S. Hu, A. Butscher, and L. Guibas. An optimization approach for extracting and encoding consistent maps in a shape collection. *ACM Transactions on Graphics (TOG)*, 2012.

[56] B. Huet and E. Hancock. Shape retrieval by inexact graph matching. In *IEEE International Conference on Multimedia Computing and Systems*, volume 1, pages 772–776, 1999.

[57] U. Kang, M. Hebert, and S. Park. Fast and scalable approximate spectral graph matching for correspondence problems. *Information Sciences*, 220:306–318, 2013.

[58] A. Kar, S. Tulsiani, J. Carreira, and J. Malik. Category-specific object reconstruction from a single image. In *CVPR*, pages 1966–1974, 2015.

[59] A. Kitamoto, C. Zhou, and M. Takagi. Similarity retrieval of noaa satellite imagery by graph matching. In *IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology*, pages 60–73, 1993.

[60] E. L. Lawler. The quadratic assignment problem. *Management Science*, pages 586–599, 1963.

[61] J. Lebrun, P. Gosselin, and S. Philipp-Foliguet. Inexact graph matching based on kernels for object retrieval in image databases. *Image and Vision Computing*, 29(11):716–729, 2011.

[62] J. Lee, M. Cho, and K. Lee. A graph matching algorithm using data-driven markov chain monte carlo sampling. In *ICPR*, 2010.

[63] J. Lee, M. Cho, and K. M. Lee. Hyper-graph matching via reweighted randomwalks. In *CVPR*, 2011.

[64] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *ICCV*, 2005.

[65] M. Leordeanu, M. Hebert, and R. Sukthankar. An integer projected fixed point method for graph matching and map inference. In *NIPS*, 2009.

[66] M. Leordeanu, R. Sukthankar, and M. Hebert. Unsupervised learning for graph matching. *Int. J. Comput. Vis.*, pages 28–45, 2012.

[67] M. Leordeanu, A. Zanfir, and C. Sminchisescu. Semi-supervised learning and optimization for hypergraph matching. In *ICCV*, 2011.

[68] G. Levi. A note on the derivation of maximal common subgraphs of two directed or undirected graphs. *Calcolo*, 9(4):341–352, 1973.

[69] C. Li, Q. Liu, W. Dong, X. Zhu, J. Liu, and H. Lu. Human age estimation based on locality and ordinal information. *IEEE Transaction on Cybernetics*, 2015.

[70] C. Li, Q. Liu, J. Liu, and H. Lu. Learning ordinal discriminative features for age estimation. In *CVPR*, 2012.

[71] C. S. Liao, K. Lu, M. Baym, R. Singh, and B. Berger. Isorankn: spectral methods for global alignment of multiple protein networks. *Bioinformatics*, 2009.

[72] L. Lin, Y. Lu, Y. Pan, and X. Chen. Integrating graph partitioning and matching for trajectory analysis in video surveillance. *IEEE Transactions on Image Processing*, 21(12):4844–4857, 2012.

[73] Z. Liu and H. Qiao. Gncgcp - graduated nonconvexity and graduated concavity procedure. *IEEE PAMI*, 2014.

[74] Z. Liu, H. Qiao, and L. Xu. An extended path following algorithm for graph-matching problem. *TPAMI*, pages 1451–1456, 2012.

[75] Z. Liu, H. Qiao, X. Yang, and S. Hoi. Graph matching by simplified convex-concave relaxation procedure. *IJCV*, 2014.

[76] L. Livi and A. Rizzi. The graph matching problem. *Pattern Analysis and Applications*, 16(3):253–283, 2013.

[77] E. M. Loiola, N. M. de Abreu, P. O. Boaventura-Netto, P. Hahn, and T. Querido. A survey for the quadratic assignment problem. *EJOR*, pages 657–90, 2007.

[78] E. M. Luks. Isomorphism of graphs of bounded valence can be tested in polynomial time. *Journal of Computer and System Sciences*, 1982.

[79] V. Lyzinski, S. Adali, J. T. Vogelstein, Y. Park, and C. E. Priebe. Seeded graph matching via joint optimization of fidelity and commensurability. *stat*, 1050:16, 2014.

[80] V. Lyzinski, D. E. Fishkind, M. Fiori, J. T. Vogelstein, C. E. Priebe, and G. Sapiro. Graph matching: Relax at your own risk. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(1):60–73, 2016.

[81] S. Melnik, H. Garcia-Molina, and E. Rahm. A versatile graph matching algorithm and its application to schema matching. *Data Engineering*, 2002.

[82] G. A. Mills-Tettey, A. Stentz, and M. B. Dias. The dynamic hungarian algorithm for the assignment problem with changing costs. 2007.

[83] C. F. Moreno-García and F. Serratosa. Correspondence consensus of two sets of correspondences through optimisation functions. *Pattern Analysis and Applications*, pages 1–13, 2015.

[84] P. Morrison and J. J. Zou. Inexact graph matching using a hierarchy of matching processes. *Computational Visual Media*, 1(4):291–307, 2015.

[85] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial & Applied Mathematics*, 5(1):32–38, 1957.

[86] M. Neuhaus and H. Bunke. Self-organizing maps for learning the edit costs in graph matching. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 35(3):503–514, 2005.

[87] M. Neuhaus and H. Bunke. Automatic learning of cost functions for graph edit distance. *Information Sciences*, 177(1):239–247, 2007.

[88] M. Neuhaus and H. Bunke. A quadratic programming approach to the graph edit distance problem. In *Graph-Based Representations in Pattern Recognition*, pages 92–102. 2007.

[89] Q. Ngoc, A. Gautier, and M. Hein. A flexible tensor block coo rd inate ascent scheme for hypergraph matching. In *CVPR*, 2015.

[90] W. Nie, A. Liu, Z. Gao, and Y. Su. Clique-graph matching by preserving global & local structure. In *CVPR*, 2015.

[91] W. Nie, A. Liu, Y. Su, H. Luan, Z. Yang, L. Cao, and R. Ji. Single/cross-camera multiple-person tracking by graph matching. *Neurocomputing*, 2014.

[92] Z. Niu, R. Martin, F. Langbein, and M. Sabin. Rapidly finding cad features using database optimization. *Computer-Aided Design*, 69:35–50, 2015.

[93] D. Pachauri, R. Kondor, and S. Vikas. Solving the multi-way matching problem by permutation synchronization. In *NIPS*, 2013.

[94] S. Park, S. Park, and M. Heber. Fast and scalable approximate spectral matching for higher-order graph matching. *PAMI*, 2013.

[95] A. Poore. Multidimensional assignment formulation of data association problems arising from multitarget and multisensor tracking. *Computational Optimization and Applications*, 3(1):27–57, 1994.

[96] C. E. Priebe, D. J. Marchette, Z. Ma, S. Adali, et al. Manifold matching: Joint optimization of fidelity and commensurability. *Brazilian Journal of Probability and Statistics*, 27(3):377–400, 2013.

[97] A. Rangarajan, A. Yuille, and E. Mjolsness. Convergence properties of the softassign quadratic assignment algorithm. *Neural Computation*, 1999.

[98] K. Riesen and H. Bunke. Approximate graph edit distance computation by means of bipartite graph matching. *Image and Vision Computing*, 27(7):950–959, 2009.

[99] K. Riesen, M. Ferrer, and H. Bunke. Approximate graph edit distance in quadratic time. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2015.

[100] C. Romanowski, R. Nagi, and M. Sudit. Data mining in an engineering design environment: Or applications from graph matching. *Computers & operations research*, 33(11):3150–3160, 2006.

[101] K. Sanders, B. Kettler, and J. Hendler. The case for graph-structured representations. In *Case-Based Reasoning Research and Development*, pages 245–254. 1997.

[102] A. Sanfeliu and K.-S. Fu. A distance measure between attributed relational graphs for pattern recognition. pages 353–362, 1983.

[103] G. Sanromà, R. Alquézar, and F. Serratosa. Attributed graph matching for image-features association using sift descriptors. In *Structural, Syntactic, and Statistical Pattern Recognition*, pages 254–263. 2010.

[104] G. Sanromà, R. Alquézar, and F. Serratosa. Smooth simultaneous structural graph matching and point-set registration. In *Graph-Based Representations in Pattern Recognition*, pages 142–151. Springer, 2011.

[105] G. Sanromà Güell, R. Alquézar Mancho, and F. Serratosa Casanelles. Graph matching using sift descriptors: an application to pose recovery of a mobile robot. In *5th International Conference on Computer Vision Theory and Applications*, pages 249–254, 2010.

[106] M. D. Santo, P. Foggia, C. Sansone, and M. Vento. A large database of graphs and its use for benchmarking graph isomorphism algorithms. *Pattern Recognition Letters*, 24(8):1067–1079, 2003.

[107] C. Schellewald and C. Schnörr. Probabilistic subgraph matching based on convex relaxation. In *EMMCVPR*, 2005.

[108] A. Schenker, M. Last, H. Bunke, and A. Kandel. Classification of web documents using graph matching. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(03):475–496, 2004.

[109] F. Serratosa. Fast computation of bipartite graph matching. *Pattern Recognition Letters*, 45:244–250, 2014.

[110] F. Serratosa. Speeding up fast bipartite graph matching through a new cost matrix. *International Journal of Pattern Recognition and Artificial Intelligence*, 29(02):1550010, 2015.

[111] C. Shen and W. Tsai. A graph matching approach to optimal task assignment in distributed computing systems using a minimax criterion. *IEEE Transactions on Computers*, 100(3):197–203, 1985.

[112] D. Shen and C. D. Hammer. Hierarchical attribute matching mechanism for elastic registration. *TMI*, 2002.

[113] Y. Shen, W. Lin, J. Yan, M. Xu, J. Wu, and J. Wang. Person re-identification with correspondence structure learning. In *ICCV*, 2015.

[114] X. Shi, H. Ling, W. Hu, J. Xing, and Y. Zhang. Tensor power iteration for multi-graph matching. In *CVPR*, 2016.

[115] A. Shokoufandeh, Y. Keselman, M. F. Demirci, D. Macrini, and S. Dickinson. Many-to-many feature matching in object recognition: a review of three approaches. *Computer Vision, IET*, 6(6):500–513, 2012.

[116] P. Showbridge, M. Kraetzl, and D. Ray. Detection of abnormal change in dynamic networks. In *Information, Decision and Control*, pages 557–562, 1999.

[117] A. Solé-Ribalta and F. Serratosa. Models and algorithms for computing the common labelling of a set of attributed graphs. *CVIU*, 2011.

[118] A. Sole-Ribalta and F. Serratosa. Graduated assignment algorithm for multiple graph matching based on a common labeling. *IJPRAI*, 2013.

[119] Y. Suh, M. Cho, and K. M. Lee. Graph matching via sequential monte carlo. In *ECCV*, 2012.

[120] Y. Tian, R. Mceachin, C. Santos, and J. Patel. Saga: a subgraph matching tool for biological graphs. *Bioinformatics*, 2007.

[121] Y. Tian, J. Yan, H. Zhang, Y. Zhang, X. Yang, and H. Zha. On the convergence of graph matching: Graduated assignment revisited. In *ECCV*, 2012.

[122] P. H. S. Torr. Solving markov random fields using semidefinite programmin. In *AISTATS*, 2003.

[123] L. Torresani, V. Kolmogorov, and C. Rother. Feature correspondence via graph matching: Models and global optimization. In *ECCV*, 2008.

[124] S. Umeyama. An eigendecomposition approach to weighted graph matching problems. *TPAMI*, 1988.

[125] M. Vento. A long trip in the charming world of graphs for pattern recognition. *Pattern Recognition*, 48(2):291–301, 2015.

[126] M. Vento and P. Foggia. Graph matching techniques for computer vision. *Graph-Based Methods in Computer Vision: Developments and Applications: Developments and Applications*, page 1, 2012.

[127] C. Wang, L. Wang, and L. Liu. Density maximization for improving graph matching with its applications. *IEEE Transactions on Image Processing*, 2015.

[128] M. Wang, Y. Lai, Y. Liang, R. R. Martin, and S.-M. Hu. Biggerpicture: data-driven image extrapolation using graph matching. *ACM Transactions on Graphics*, 33(6), 2014.

[129] W. Wang, W. Lin, Y. Chen, J. Wu, J. Wang, and B. Sheng. Finding coherent motions and semantic regions in crowd scenes: a diffusion and clustering approach. In *ECCV*, 2014.

[130] M. L. Williams, R. C. Wilson, and E. Hancock. Multiple graph matching with bayesian inference. *Pattern Recognition Letters*, pages 1275–1281, 1997.

[131] R. C. Wilson and E. R. Hancock. A bayesian compatibility model for graph matching. *Pattern Recognition Letters*, 17(3):263–276, 1996.

[132] Q. Xiao, Y. Wang, and H. Wang. Motion retrieval using weighted graph matching. *Soft Computing*, 19(1), 2015.

[133] J. Yan, M. Cho, H. Zha, X. Yang, and S. Chu. Multi-graph matching via affinity optimization with graduated consistency regularization. *TPAMI*, 2016.

[134] J. Yan, Y. Li, W. Liu, H. Zha, X. Yang, and S. Chu. Graduated consistency-regularized optimization for multi-graph matching. In *ECCV*, 2014.

[135] J. Yan, Y. Tian, H. Zha, X. Yang, Y. Zhang, and S. Chu. Joint optimization for consistent multiple graph matching. In *ICCV*, 2013.

[136] J. Yan, J. Wang, H. Zha, and X. Yang. Consistency-driven alternating optimization for multigraph matching: A unified approach. *IEEE Transactions on Image Processing*, 2015.

[137] J. Yan, J. Wang, H. Zha, X. Yang, and S. Chu. Multi-view point registration via alternating optimization. In *AAAI*, 2015.

[138] J. Yan, H. Xu, H. Zha, X. Yang, H. Liu, and S. Chu. A matrix decomposition perspective to multiple graph matching. In *ICCV*, 2015.

[139] J. Yan, C. Zhang, H. Zha, W. Liu, X. Yang, and S. Chu. Discrete hyper-graph matching. In *CVPR*, 2015.

[140] Y. Yan, E. Ricci, R. Subramanian, G. Liu, O. Lanz, and N. Sebe. A multi-task learning framework for head pose estimation under target motion. 2016.

[141] Q. Yang and S. Sze. Path matching and graph matching in biological networks. *Journal Of Computational Biology*, 2007.

[142] B. Yao and F. Li. Action recognition with exemplar based 2.5 d graph matching. In *ECCV*, 2012.

[143] C. Yu, L. Tian, H. Hu, Y. Duan, and J. Zhou. Progressive feature matching via triplet graph. In *ICIP*, 2015.

[144] T. Yu and R. Wang. Scene parsing using graph matching on street-view data. *CVIU*, 2016.

[145] M. Zaslavskiy, F. Bach, and J.-P. Vert. Many-to-many graph matching: a continuous relaxation approach. In *Machine Learning and Knowledge Discovery in Databases*, pages 515–530. Springer, 2010.

[146] M. Zaslavskiy, F. R. Bach, and J.-P. Vert. Global alignment of proteinÍcprotein interaction networks by graph matching methods. *Bioinformatics*, 2009.

[147] M. Zaslavskiy, F. R. Bach, and J.-P. Vert. A path following algorithm for the graph matching problem. *TPAMI*, 2009.

[148] R. Zass and A. Shashua. Probabilistic graph and hypergraph matching. In *CVPR*, 2008.

[149] D. Zhang and S. Chang. Detecting image near-duplicate by stochastic attributed relational graph matching with learning. In *ACM MM*, pages 877–884, 2004.

[150] J. Zhang and P. Yu. Multiple anonymized social networks alignment. In *ICDM*, 2015.

[151] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *IJCV*, 1994.

[152] W.-L. Zhao, C.-W. Ngo, H.-K. Tan, and X. Wu. Near-duplicate keyframe identification with interest point matching and pattern learning. *IEEE Transactions on Multimedia*, 9(5):1037–1048, 2007.

[153] J. Zhong, H. Zhu, J. Li, and Y. Yu. Conceptual graph matching for semantic search. In *Conceptual structures: Integration and interfaces*, pages 92–106. 2002.

[154] F. Zhou and F. Torre. Factorized graph matching. *IEEE PAMI*, 2016.

[155] F. Zhou and F. D. Torre. Deformable graph matching. In *CVPR*, 2013.

[156] J. Zhou, H. Yan, and Y. Zhu. Local topology preserved tensor models for graph matching. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 2153–2157, 2015.

[157] K. Zhou, H. Zha, and L. Song. Learning triggering kernels for multi-dimensional hawkes processes. In *ICML*, pages 1301–1309, 2013.

[158] X. Zhou, M. Zhu, and K. Daniilidis. Multi-image matching via fast alternating minimization. In *ICCV*, 2015.

[159] H. Zhu, D. Liu, S. Zhang, Y. Zhu, L. Teng, and S. Teng. Solving the many to many assignment problem by improving the kuhn–munkres algorithm with backtracking. *Theoretical Computer Science*, 2016.