Learning Graph Matching with GNCCP

Shaofeng Zeng
Faculty of Information Technology
Beijing University of Technology
Beijing, China
shaofeng_z@emails.bjut.edu.cn

Yujian Li*
Faculty of Information Technology
Beijing University of Technology
Beijing, China
liyujian@bjut.edu.cn

Zhaoying Liu
Faculty of Information Technology
Beijing University of Technology
Beijing, China

Too Edna
Faculty of Information Technology
Beijing University of Technology
Beijing, China

ABSTRACT
To improve the accuracy of graph matching problem, two strategies are frequently used, i.e., investigating more accurate optimization methods and constructing more accurate pairwise affinity functions. However, traditional methods have placed emphasizes on one of them. Here, we consider both. More specifically, we combine the graduated nonconvexity concavity relaxation procedure with the learning graph matching method. The former one is robust in dealing with graph matching problem, the latter one can construct the pairwise affinity functions by a supervised learning procedure. Experimental results show that the method perform better on CMU house/hotel data set, and competitively on synthetic noisy data set with the state of the art.

CCS Concepts
Computing methodologies→Artificial intelligence→Computer vision→Computer vision problems→Matching

Keywords
Graph matching; Supervised learning; GNCCP.

1. INTRODUCTION
Graph matching has been studied for decades in various fields such as pattern recognition, computer vision [1], [2], and operational research [3], [4]. Roughly speaking, graph matching aims at finding the optimal correspondences between two graph vertices. The correspondences can be one to one or one to many. In this paper, we investigate the problem of the same size graph matching with one to one correspondences. Then the solution of this kind of problem can be compactly represented as a permutation matrix.

In mathematics, graph matching problem is also known as quadratic assignment problem (QAP). Because of its combinatorial nature, exactly solving the problem will become much harder especially for large graphs. However, in practical application, searching exact solution is unnecessary and impractical. So a main body of researchers focus on exploiting more efficient and accurate methods to approximately solve it.

Among the various approximations, the continuous relaxation methods have attracted extensive attentions in recent years. These methods relax the discrete permutation constraint to a continuous one, which make the problem more flexible to solve. Leordeanu and Hebert [5] approximated the permutation matrix by a unit length vector. The optimal solution can be simply computed by the principal eigen-vector of the pairwise affinity matrix in the objective function. But if the eigen-gap is ambiguous, the solution may be bad. Gold and Rangarajan [6] proposed a graduated assignment algorithm (GA). GA finds a doubly stochastic matrix by iteratively solving a series of first order Taylor expansions of the objective function. However, GA is sensitive to the initializations and easily trapped in local extremes. Zaslavskiy *et al* [7] formulated the graph matching problem into a weighted linear combination of a convex and concave relaxation. In particular, they provided a path following algorithm (PATH) to minimize the concave relaxation which holds the same minima as the original matching problem. Theoretically speaking, PATH provides a method which can find a discrete and more accurate local minima by continuous optimization methods. This technique makes PATH among the most successful approximation methods. However, PATH can only apply to a special form of graph matching problem called *Koopmans-Beckmann* [8]. In order to make this excellent idea applicable to other related problems, Liu and Qiao [9] proposed the graduated nonconvexity and concavity procedure (GNCCP). Theoretically speaking, GNCCP can deal with any forms of graph matching problem, meanwhile, it achieves better or at least no worse performance than PATH.

All the methods listed above mainly focus on devising better approximation methods. Whereas, Caetano *et al* [10] proposed learning based method (LGM), to tackle the graph matching problem from the perspective of constructing more accurate pairwise affinity functions. For this purpose, LGM formulates the affinity functions between vertices and edges as a $\boldsymbol{w}$ parameterized one, which is estimated by a supervised learning procedure.

In this paper, we suggest to tackle the graph matching problem both by the improvement of the approximation technique and by the learning technique. To this end, we combine GNCCP with the learning method. Specifically, GNCCP is employed to solve a quadratic assignment problem involved in the learning procedure of LGM. This makes LGM more robust and stable than the original one, which further lead to better performance.

IC4E 2018, January 11–13, 2018, San Diego, CA, USA

ACM ISBN 978-1-4503-5485-1/18/01…$15.00
https://doi.org/10.1145/3183586.3183608

2. THE GRAPH MATCHING PROBLEM

In this paper, a graph G is denoted as $G=\left(\{V_i\},\{E_{ij}\}\right)$, where, $\{V_i\}$ and $\{E_{ij}\}$ is the sets of vertices and edges of G respectively; E_{ij} represents an edge with two endpoints V_i and V_j. Given another graph $H=\left(\{V_a\},\{E_{ab}\}\right)$, define $\mathrm{G}=\left(G,H\right)$, and suppose $i,j,a,b\in\{1,2,...k\}$, then the graph matching problem can be expressed as,

$$\max_{\boldsymbol{X}\in\mathrm{P}}\mathrm{F}\left(\boldsymbol{X};\mathrm{G}\right)=\mathrm{vec}(\boldsymbol{X})^{\mathrm{T}}\boldsymbol{K}\mathrm{vec}(\boldsymbol{X}) \tag{1}$$

where, $\mathrm{vec}(\boldsymbol{X})$ is a vector reshaped by matrix $\boldsymbol{X}$, P denotes the set of permutation matrices.

$$\mathrm{P}=\left\{\boldsymbol{X}\mid\boldsymbol{X}_{ia}\in\{0,1\},\sum_{i=1}^{k}\boldsymbol{X}_{ia}=1,\sum_{a=1}^{k}\boldsymbol{X}_{ia}=1,\forall i,a\right\}$$

Different values of $\boldsymbol{X}_{ia}$ represent different matching results: if vertex V_i matches vertex V_a, then $\boldsymbol{X}_{ia}=1$, otherwise, $\boldsymbol{X}_{ia}=0$. $\boldsymbol{K}$ is a $n^2\times n^2$ square matrix with pairwise affinities.

Relaxation methods such as PATH, GNCCP solve problem (1) by relaxing P to its convex hull, i.e. the doubly stochastic matrices set denoted by D,

$$\mathrm{D}=\left\{\boldsymbol{X}\mid\boldsymbol{X}_{ia}\geq 0,\sum_{i=1}^{k}\boldsymbol{X}_{ia}=1,\sum_{a=1}^{k}\boldsymbol{X}_{ia}=1,\forall i,a\right\}$$

Then the relaxation of problem (1) can be formulated as,

$$\max_{\boldsymbol{X}\in\mathrm{D}}\mathrm{F}\left(\boldsymbol{X};\mathrm{G}\right)=\mathrm{vec}(\boldsymbol{X})^{\mathrm{T}}\boldsymbol{K}\mathrm{vec}(\boldsymbol{X}) \tag{2}$$

After solving (2), we need to project the doubly stochastic matrix back to its nearest permutation matrix. This problem is usually represented as,

$$\boldsymbol{P}^{*}=\arg\max_{\boldsymbol{P}\in\mathrm{P}f\ \ s}\mathrm{tr}\left(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{X}\right) \tag{3}$$

Problem (3) is called linear assignment problem, which can be optimized by Hungarian method [11] or greedy method. From another perspective, we are actually use the divide and conquer technique to solve graph matching problem: we divide the graph matching problem into two sub-problems—a relaxation problem and a linear assignment problem. And both of the two problems are easier to solve than the original one.

3. LGM AND ITS LEARNING PROCEDURE

3.1 The Model of LGM

LGM constructs a $\boldsymbol{w}$ parameterized graph matching model which can be expressed as follows,

$$\max_{\boldsymbol{X}\in\mathrm{P}}\mathrm{F}\left(\boldsymbol{X};\mathrm{G},\boldsymbol{w}\right)=\mathrm{vec}(\boldsymbol{X})^{\mathrm{T}}\boldsymbol{K}\left(\boldsymbol{w}\right)\mathrm{vec}(\boldsymbol{X})$$

More specifically, each element $\boldsymbol{K}\left(\boldsymbol{w}\right)_{ia;jb}$ of $\boldsymbol{K}\left(\boldsymbol{w}\right)$ is computed as follows,

$$\boldsymbol{K}\left(\boldsymbol{w}\right)_{ia;jb}=\begin{cases}\phi_1\left(V_i,V_a,\boldsymbol{\omega}_1\right) & \text{if } i=j \text{ and } a=b\\ \phi_2\left(E_{ij},E_{ab},\boldsymbol{\omega}_2\right) & \text{if } i\neq j \text{ and } a\neq b\\ 0 & \text{otherwise}\end{cases} \tag{4}$$

where, $\boldsymbol{w}=\left[\boldsymbol{\omega}_1;\boldsymbol{\omega}_2\right]$, $\phi_1\left(V_i,V_a,\boldsymbol{\omega}_1\right)$ and $\phi_2\left(E_{ij},E_{ab},\boldsymbol{\omega}_2\right)$ are the affinity functions between pairs $\left(V_i,V_a\right)$ and $\left(E_{ij},E_{ab}\right)$, respectively.

Obviously, parameter $\boldsymbol{w}$ makes LGM more flexible and efficient. Appropriate $\boldsymbol{w}$ can construct more accurate affinity functions, which certainly improve the matching performance.

3.2 The Learning Procedure

In this paper, $\boldsymbol{w}$ is estimated by a supervised learning procedure. The training set consists of N samples, represented as $\left\{\left(\mathrm{G}^1,\boldsymbol{X}^1\right),...,\left(\mathrm{G}^N,\boldsymbol{X}^N\right)\right\}$. The n-th sample G^n is a pair of graphs, denoted as $\mathrm{G}^n=\left(G^n,H^n\right)$, $\boldsymbol{X}^n$ is the label, i.e., the ground truth matching. Define the discriminant function as $g\left(\mathrm{G};\boldsymbol{w}\right)=\arg\max_{\boldsymbol{X}\in\mathrm{P}}\mathrm{F}\left(\boldsymbol{X};\mathrm{G},\boldsymbol{w}\right)$, the learning procedure mainly focuses on minimizing the following expression,

$$\frac{1}{N}\sum_{n=1}^{N}\Delta\left(g\left(\mathrm{G}^n,\boldsymbol{w}\right),\boldsymbol{X}^n\right)+\lambda\Omega\left(\boldsymbol{w}\right) \tag{5}$$

The first part of (5) is the empirical risk which measures the loss of the discriminate function on all training samples; the second part is the regularization term which can avoid over fitting problem; parameter λ trades off the data fitting against generalization ability.

The learning problem for graph matching is more complicated than the common machine learning problems, such as classification and regression. The input and output of the discriminate function is structured objects; and the discriminate function itself is an optimization problem; although the regularization term can be convex, the empirical risk is not; additionally, large equivalent of $\boldsymbol{w}$s exist for the same empirical risk. So reference [10] applies Bundle method to minimize a convex upper bound of (5) instead. Specifically, Bundle method solves the optimization problem by a series of linear approximations using Taylor expansions. It should be noted that one key step of the Bundle method involves solving the following problem for each training samples,

$$\arg\max_{\boldsymbol{X}\in\mathrm{D}}\mathrm{F}\left(\boldsymbol{X};\mathrm{G}^n,\boldsymbol{w}\right)+\Delta\left(\boldsymbol{X},\boldsymbol{X}^n\right) \tag{6}$$

here, $\Delta\left(\boldsymbol{X},\boldsymbol{X}^n\right)$ is the loss function computed by Hamming loss shown as follows.

$$\Delta\left(\boldsymbol{X},\boldsymbol{X}^n\right)=1-\frac{1}{\left\|\boldsymbol{X}^n\right\|_F^2}\mathrm{tr}\left(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}^n\right) \tag{7}$$

Problem (6) is indeed a QAP which is maximized by GA in reference [10].

4. LGM WITH GNCCP

As is shown in [10], the inexact nature of GA not only bring trouble to the adjustment of parameter λ in formula (5) but also lead to the poor performance of LGM. So we use a more robust method called GNCCP instead. The model of GNCCP is formulated as,

$$F_{\zeta}(\boldsymbol{X})_{\boldsymbol{X}\in\mathrm{D}}=\begin{cases}(1-\zeta)F(\boldsymbol{X})+\zeta\operatorname{tr}(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}) & \text{if } 1\geq\zeta\geq 0\\(1+\zeta)F(\boldsymbol{X})+\zeta\operatorname{tr}(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}) & \text{if } 0>\zeta\geq -1\end{cases}\tag{8}$$

In implementation, GNCCP set $\zeta=1$ and gradually decrease it until $\zeta=-1$, meanwhile, solve $F_{\zeta}(\boldsymbol{X})$ for each new ζ with Frank-Wolfe algorithm[12], using the result of the previous $F_{\zeta}(\boldsymbol{X})$ as the starting point. The authors of [9] prove that formula (8) will be a convex and concave problems by some $\zeta<1$ and $\zeta>-1$ respectively. So we exactly start from solving a convex graph matching problem and end with a concave one. GNCCP benefits a lot from this particular solving method: 1) Optimizing the convex problem first will give a global extreme which may near the true result of the concave one. 2) Theoretically, we can indeed get a permutation matrix by optimizing the concave graph matching problem on set D [13]; 3) GNCCP can naturally avoid some errors brought about by the discretization procedure of (3); 4) By gradually decrease ζ from 1 to -1, the GNCCP can avoid more unwanted local extremes.

GNCCP provides a generic framework to realize the convex concave relaxation procedure for graph matching problem. Moreover, GNCCP exhibits a better or at least a no worse performance than PATH.

5. EXPERIMENTAL RESULTS

In this section, we compare the proposed method (LGM-GNCCP) with some classical ones on the CMU house/hotel data sets and synthetic noisy data set. We are interested in comparison of the GA, GNCCP, and traditional learning method for LGM (LGM-GA)

To simplify the calculation and get more better performance, we set $\phi_1(V_i,V_a,\boldsymbol{\omega}_1)=0$ for all i and a; and $\phi_2(E_{ij},E_{ab},\boldsymbol{\omega}_2)$ is computed as $\phi_2(E_{ij},E_{ab},\boldsymbol{\omega}_2)=\boldsymbol{\omega}_2^{\mathrm{T}}\boldsymbol{D}_{ia;jb}$, where $\boldsymbol{D}_{ia;jb}$ is computed as,

$$\boldsymbol{D}_{ia;jb}=-\left[(d_{ij}-d_{ab})^2;(\sigma_{ij}-\sigma_{ab})^2;(\alpha_{ij}-\alpha_{ab})^2;(\beta_{ij}-\beta_{ab})^2\right]$$

Details of $d_{ij}(d_{ab})$, $\sigma_{ij}(\sigma_{ab})$, $\alpha_{ij}(\alpha_{ab})$ and $\beta_{ij}(\beta_{ab})$ are as follows,

$$d_{ij}=Eu(V_i,V_j)/\max_{x,y}\left(Eu(V_x,V_y)\right)$$

$$\sigma_{ij}=\arccos\left(\overrightarrow{V_iV_j},\overrightarrow{OV_i}\right)/\max_{x,y}\left(\arccos\left(\overrightarrow{V_xV_y},\overrightarrow{OV_x}\right)\right)$$

$$\alpha_{ij}=\arccos\left(\overrightarrow{V_iV_j},\boldsymbol{h}\right)/\max_{x,y}\left(\arccos\left(\overrightarrow{V_xV_y},\boldsymbol{h}\right)\right)$$

$$\beta_{ij}=\arccos\left(\overrightarrow{OV_i},\overrightarrow{OV_j}\right)/\max_{x,y}\left(\arccos\left(\overrightarrow{OV_x},\overrightarrow{OV_y}\right)\right)$$

where $Eu(\cdot)$ is the Euclid distance; $\arccos(\cdot)$ is the inverse of cosine; O represents the point (0,0); $\boldsymbol{h}=[0,1]$. For GA and GNCCP, we just set $\boldsymbol{\omega}_2=[1;1;1;1]$.

5.1 CMU House/Hotel Data Sets

The CMU house data set [14] contains 111 frames of toy house images. Each frame has 30 hand labeled landmarks. The hotel data set [15] contains 101 frames of toy hotel images. Each frame also has 30 landmarks.

Fig. 1 shows the experimental results of the four methods on house data set. The horizontal axis is the baseline which means the separation of two image numbers; the vertical axis is the normalized Hamming loss on test set. From the figure we can see that all the methods except GA achieve the all zero losses. Note that GNCCP achieves the same performance with two learning methods, i.e., LGM-GA and LGM-GNCCP, which shows its robustness.

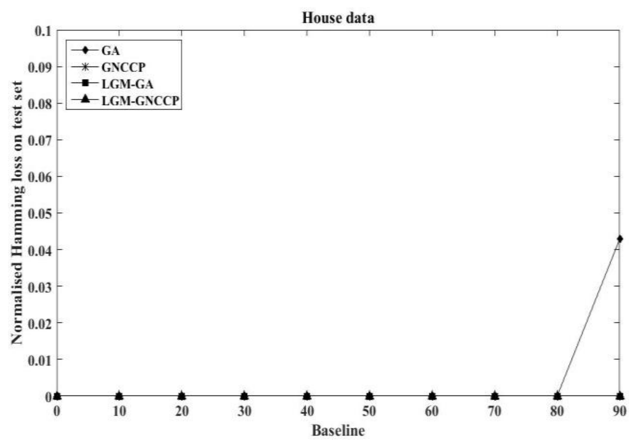

Figure 1. Performances of four methods on the house data.

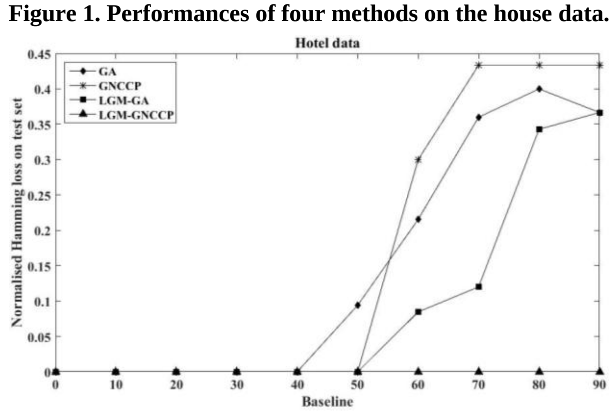

Figure 2. Performances of four methods on the hotel data.

Fig. 2 shows the performances of the same four methods on hotel data. The losses of GA and GNCCP grow rapidly after the baselines of 40, 50, respectively. Two learning methods outperform GA and GNCCP after the baseline of 50. Additionally, LGM-GNCCP achieves all zero losses, which significantly outperforms LGM-GA.

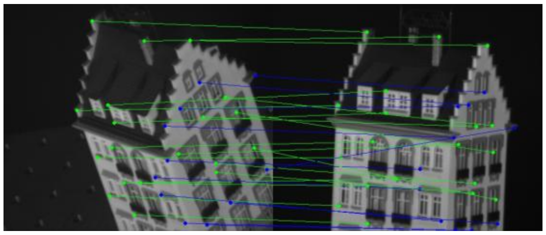

Figure 3. A matching example of LGM-GA (baseline=90,hotel).

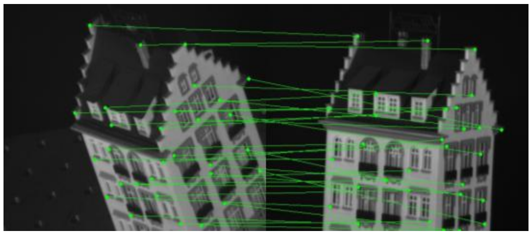

Figure 4. A matching example of LGM-GNCCP (baseline=90,hotel).

On the whole, the relatively better performances of two learning methods shows the effectiveness of the learning technique. Moreover, LGM-GNCCP outperforms both LGM-GA and GNCCP, which verifies the feasibility of our methods.

In Figure 3-4, we present two matching examples for GNCCP and LGM-GNCCP. The two images are the 3rd and the 93th frames from the hotel data set. A green line linking two points gives a correct match, a blue line a wrong match. Totally, there are 11 and 0 pairs of points mismatched for LGM-GA and LGM-GNCCP, respectively.

5.2 Synthetic Noisy Data Set

We also test LGM-GNCCP on synthetic noisy data set [10]. This data set contains 200 images. Each image has 35 landmarks. And we gradually apply noise with standard deviation 20 pixels for the landmarks.

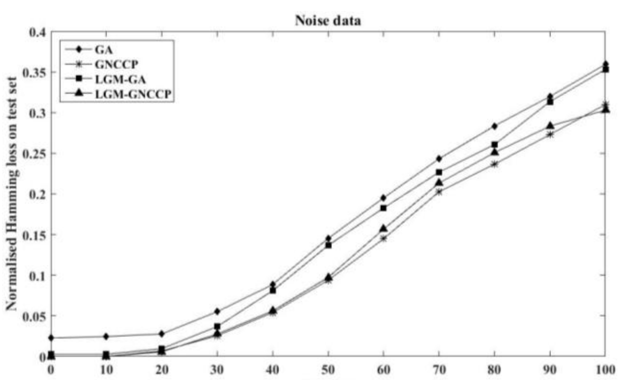

Figure 5. Performances of four methods on the noise data.

Figure 5 is the experimental results on noise data. From the figure we can see that GNCCP achieves the best performance except baseline=100. LGM-GA outperforms GA, while LGM-GNCCP outperforms LGM-GA under all baselines. Moreover, LGM-GNCCP has a similar performance with GNCCP. The performance of LGM-GNCCP seems different from our expectation. This mainly because of the randomness of the noise. Although GNCCP is robust to noise, it is very hard for the learning procedure to handle the random noise.

6. CONCLUSION

In this paper, we have proposed a new learning procedure with GNCCP for learning graph matching method. The new method is called LGM-GNCCP. Because of the following two advantages, LGM-GNCCP can achieve more accurate performances: 1) LGM-GNCCP can learn the affinity functions according to the pattern of the graph data; 2) GNCCP provides a more accurate approximation to solve quadratic assignment problem, which indeed improves the learning procedure and the final matching results. We compared LGM-GNCCP with three state-of-the-art methods: GA, GNCCP and LGM-GA, the experimental results show that LGM-GNCCP can achieve relatively good performances on CMU house/hotel data. Moreover, it is competitive with other methods on random noise data. Particular, the linear weighted Koopmans-Beckmann's graph matching problem which can be seen as a special form of LGM, can also be solved by LGM-GNCCP. As future work, we would like to explorer learning method with better performance for noise data.

7. ACKNOWLEDGMENT

This research was financially supported in part by National Natural Science Foundation of China (61175004), the China Postdoctoral Science Foundation funded project (2015M580952) and Beijing Postdoctoral Research Foundation (2016ZZ-24).

8. REFERENCES

[1] O. Duchenne, A. Joulin and J. Ponce, "A graph-matching kernel for object categorization," *Proc. IEEE International Conference on Computer Vision*, IEEE Computer Society Press, Nov. 2011, 1792-1799, doi: 10.1109/ICCV.2011.6126445.

[2] E. Serradell, M. A. Pinheiro, R. Sznitman, J. Kybic, F. Moreno-Noguer and P. Fua, "Non-rigid graph registration using active testing search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37, 3, Mar., 2015, 625-638.

[3] K.M. Anstericher and N.W. Brixius, "Solving quadratic assignment problems using convex quadratic programming relaxations," *Optimization Methods and Software*, 16, 1-4, Jan., 2001, 49-68.

[4] A. Misevicius, "A tabu search algorithm for the quadratic assignment problem," *Computational Optimization and Applications*, 30, 1, Jan., 2005, 95-111.

[5] M. Leordeanu and M. Hebert, "A spectral technique for correspondence problems using pairwise constraints," *Proc. IEEE International Conference on Computer Vision*, IEEE Computer Society Press, 2, Oct. 2005, 1482-1489, doi: 10.1109/ICCV.2005.20.

[6] S. Gold and A. Rangarajan, "A graduated assignment algorithm for graph matching," *IEEE Transactions on Pattern analysis Machine Intelligence*, 18, Apr., 1996, 377-388.

[7] M. Zaslavskiy, F. R. Bach and J.P. Vert, "A path following algorithm for the graph matching problem," *IEEE Transactions on Pattern Analysis Machine Intelligence*, 31, 12, Dec., 2009, 2227-2242.

[8] T. C Koopmans and M. Beckmann, "Assignment problems and the location of economic activities," *Econometrica*, 25, 1, 1957, 53-76.

[9] Z.Y. Liu and H. Qiao, “GNCCP-graduated nonconvexity and concavity procedure,” *IEEE Transactions on Pattern Analysis Machine Intelligence*, 36, 6, Nov., 2014, 1258-1267.

[10] T. S. Caetano, J. J. McAuley, C. Li, Q. V. Le, and A. J. Smola, “Learning graph matching,” *IEEE Transactions on Pattern Analysis Machine Intelligence*, 31, 6, Jun., 2009, 1048-1058.

[11] H. W. Kuhn, “The hungarian method for the assignment problem,” *Naval Research Logistics Quarterly*, 2, 1/2, 1955, 83- 97.

[12] M. Frank and P. Wolfe, “An algorithm for quadratic programming,” *Naval Research Logistics Quarterly*, 3, 1/2, Mar., 1956, 95-110.

[13] J. Maciel and J. P. Costeira, “A global solution to sparse correspondence problems,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25, 2, Feb., 2003, 187-199.

[14] CMU “house” data set, http://vasc.ri.cmu.edu/idb/html/motion/house/index.html

[15] CMU “hotel” data set, http://vasc.ri.cmu.edu//idb/html/motion/hotel/index.html