

# Non-bayesian Graph Matching without Explicit Compatibility Calculations

Barend Jacobus van Wyk<sup>1,2</sup> and Michaël Antonie van Wyk<sup>3</sup>

<sup>1</sup> Kentron, a division of Denel, Centurion, South Africa

<sup>2</sup> University of the Witwatersrand, Johannesburg, South Africa  
ben.van.wyk@kentron.co.za

<sup>3</sup> Rand Afrikaans University, Johannesburg, South Africa  
mavw@ing.rau.ac.za

**Abstract.** This paper introduces a novel algorithm for performing Attributed Graph Matching (AGM). A salient characteristic of the Interpolator-Based Kronecker Product Graph Matching (IBKPGM) algorithm is that it does not require the explicit calculation of compatibility values between vertices and edges, either using compatibility functions or probability distributions. No assumption is made about the adjacency structure of the graphs to be matched. The IBKPGM algorithm uses Reproducing Kernel Hilbert Space (RKHS) interpolator theory to obtain an unconstrained estimate to the Kronecker Match Matrix (KMM) from which a permutation sub-matrix is inferred.

## 1 Introduction

An object can be described in terms of its parts, the properties of these parts and their mutual relationships. Representation of the structural descriptions of objects by attributed relational graphs reduces the problem of matching to an *Attributed Graph Matching* (AGM) problem.

According to [1], graph matching algorithms can be divided into two major approaches. In general, the first approach constructs a state-space, which is searched using heuristics to reduce complexity [2-7]. The second approach, which is the one adopted here, is based on function optimization techniques which include Bayesian, linear-programming, continuation, eigen-decomposition, polynomial transform, genetic, neural network and relaxation-based methods [1, 8-19].

Except for some earlier approaches not suited to sub-graph matching, such as [8-10], most optimization-based algorithms require the explicit calculation of compatibility values between vertices and edges, either using compatibility functions or probability distributions. In addition, some Bayesian-based approaches fail when graphs are fully connected [14-19]. The focus of this paper is on matching fully-connected, undirected attributed graphs, using a non-Bayesian optimization framework, without the explicit calculation of compatibility values or matrices.

The outline of the presentation is as follows: In section 2, we briefly review the concept of an attributed graph, formulate the *Attributed Graph Matching* (AGM)

problem, and introduce the *Kronecker Product Graph Matching* KPGM formulation. The *Interpolator-Based Kronecker Product Graph Matching* (IBKPGM) algorithm is presented in section 3. Numerical results obtained during the evaluation of our algorithm are presented in section 4.

## 2 KPGM Formulation

The focus of this paper is on matching graphs where a duplicate graph, say

$$G = (V, E, \{\mathbf{A}_i\}_{i=1}^r, \{\mathbf{B}_j\}_{j=1}^s) \quad (1)$$

is matched to a reference graph, say

$$G' = (V', E', \{\mathbf{A}'_i\}_{i=1}^r, \{\mathbf{B}'_j\}_{j=1}^s) \quad (2)$$

where  $\mathbf{A}_i \in \mathbb{R}^{n \times n}$ ,  $\mathbf{B}_j \in \mathbb{R}^{n \times 1}$ ,  $\mathbf{A}'_i \in \mathbb{R}^{n' \times n'}$  and  $\mathbf{B}'_j \in \mathbb{R}^{n' \times 1}$  represent the edge attribute adjacency matrices and vertex attribute vectors respectively. The reference and duplicate graphs each have  $r$  edge attributes and  $s$  vertex attributes. The number of vertices of  $G'$  (respectively,  $G$ ) is  $n' := |V'|$  (respectively,  $n := |V|$ ). Here we consider the general case of sub-graph matching. *Full-graph Matching* (FGM) refers to matching two graphs having the same number of vertices (i.e.  $n' = n$ ) while *Sub-graph Matching* (SGM) refers to matching two graphs having a different number of vertices (i.e.  $n' > n$ ).

We say that  $G$  is *matched* to some sub-graph of  $G'$  if there exists an  $n \times n'$  permutation sub-matrix  $\mathbf{P}$  such that  $\mathbf{A}_i = \mathbf{P}\mathbf{A}'_i\mathbf{P}^T$  and  $\mathbf{B}_j = \mathbf{P}\mathbf{B}'_j$  where  $i = 1, \dots, r$  and  $j = 1, \dots, s$ .

We now observe that vertex attribute vectors converted to diagonal matrices, using the  $\text{diag}(\cdot)$  operation in linear algebra, satisfy the same expression as edge attribute matrices do, namely  $\text{diag}\mathbf{B}_j \approx \mathbf{P}\text{diag}(\mathbf{B}'_j)\mathbf{P}^T$ , with exact equality holding for the ideal case (i.e. when  $G$  is just a permuted sub-graph of  $G'$ ). This means that these converted vertex attribute vectors may be considered as additional edge attribute matrices. Observing that

$$\text{vec}(\mathbf{A}_i - \mathbf{P}\mathbf{A}'_i\mathbf{P}^T) \equiv \text{vec}\mathbf{A}_i - \Phi\text{vec}\mathbf{A}'_i \quad (3)$$

where  $\Phi = \mathbf{P} \otimes \mathbf{P}$  and  $\otimes$  denotes the *Kronecker Product* for matrices, the AGM problem can be expressed as

$$\min_{\Phi} \left( \sum_{i=1}^{r+s} \|\text{vec}\mathbf{A}_i - \Phi\text{vec}\mathbf{A}'_i\|^2 \right) \quad (4)$$

subject to  $\Phi = \mathbf{P} \otimes \mathbf{P}$ ,  $\mathbf{P} \in \text{Per}(n, n')$  where  $\text{Per}(n, n')$  is the set of all  $n \times n'$  permutation sub-matrices. Here  $\|\cdot\|$  denotes some matrix norm. Different norms will yield different solutions to the above minimization problem. The IBKPGM algorithm in essence obtains an approximation to  $\Phi$ , from which  $\mathbf{P} \in \text{Per}(n, n')$  can be derived.

The following definitions and observation are used in the sequel:

**Definition 1.** The matrix  $\Phi = \mathbf{P} \otimes \mathbf{P}$  minimizing Eq. (4), subject to  $\mathbf{P} \in \text{Per}(n, n')$ , is termed the Constrained Kronecker Match Matrix.

**Definition 2.** An unconstrained approximation to the Constrained Kronecker Match Matrix, say  $\hat{\Phi}$ , is termed the Unconstrained Kronecker Match Matrix.

**Definition 3.** An unconstrained approximation to the permutation matrix, say  $\hat{\mathbf{P}}$ , is termed the Unconstrained Permutation Sub-matrix.

**Observation 1** Given a Constrained Kronecker Match Matrix,  $\Phi = \mathbf{P} \otimes \mathbf{P}$ , such that  $\mathbf{P} \in \text{Per}(n, n')$ , we can retrieve the unknown permutation sub-matrix  $\mathbf{P}$  by

$$P_{ij} := \frac{\sum_{k,l} \Phi_{kl}}{n}, \quad (5)$$

where  $i = 1, \dots, n$ ,  $j = 1, \dots, n'$ ,  $k = (i-1)n + 1, \dots, (i-1)n + n$  and  $l = (j-1)n' + 1, \dots, (j-1)n' + n'$ .

The space generated by all Unconstrained Kronecker Match Matrices contains the space generated by all Constrained Kronecker Match Matrices. The general approach we follow is to first derive an unconstrained Kronecker Match Matrix from which we then infer a Constrained Kronecker Match Matrix.

### 3 The Interpolator-Based Approach

The *Interpolator Based Kronecker Product Graph Matching* (IBKPGM) algorithm uses *Reproducing Kernel Hilbert Space* (RKHS) interpolator theory, the general framework of which is presented in [20], to obtain an unconstrained approximation to  $\Phi$ . The RIGM algorithm [21] is based on the same theory, but the way in which the permutation sub-matrix  $\mathbf{P}$  is obtained, is completely different.

#### 3.1 The Interpolator Equation

The AGM process can be viewed to be comprising of the mapping

$$\mathbf{F} : \mathbb{R}^{n' \times n'} \rightarrow \mathbb{R}^{n \times n}, \quad \mathbf{F}(\mathbf{X}) := \mathbf{P} \mathbf{X} \mathbf{P}^T, \quad (6)$$

with the fixed parameter matrix  $\mathbf{P} \in \text{Per}(n, n')$  the unknown quantity to be determined. In doing so, the AGM problem is cast into a system identification problem which can be solved by selecting an appropriate RKHS-based interpolator to model the above mapping. This interpolator contains an approximation to  $\mathbf{P}$  implicitly. The input-output pairs used for establishing the interpolative constraints associated with  $\mathbf{F}$  is the set  $\{(\mathbf{X}_k, \mathbf{Y}_k)\}_{k=1}^{r+s}$ , where  $\mathbf{X}_k := \mathbf{A}'_k$  and  $\mathbf{Y}_k := \mathbf{A}_k$ .

The framework presented in [20] will now be used to find a solution to the AGM problem. Consider the space of all functions of the form  $F : \mathbb{R}^{(n')^2 \times 1} \rightarrow \mathbb{R}$ ,  $F(\mathbf{X}) = \sum_{i,j=1}^{n'} \varphi_{ij} X_{ij}$ . The component functions  $F_{ij}$  of the function  $\mathbf{F}$

clearly belong to this space. We choose as basis for this space the functions  $e_{ij}(\mathbf{X}) := X_{ij}$  where  $i, j = 1, \dots, n'$ . Next, we endow this space with the inner product  $(F, G) := \sum_{i,j=1}^{n'} \varphi_{ij} \gamma_{ij}$  to obtain an RKHS with reproducing kernel given by  $K(\mathbf{Y}, \mathbf{Z}) = \sum_{i,j=1}^{n'} e_{ij}(\mathbf{Y}) e_{ij}(\mathbf{Z}) = (\text{vec } \mathbf{Y})^T \text{vec } \mathbf{Z}$ . Here  $\text{vec}(\cdot)$  denotes the vectorization operation of linear algebra. Given the training set  $\{(\mathbf{A}'_k, \mathbf{A}_k)\}_{k=1}^{r+s}$  we therefore have the interpolative constraints,  $(F_{ij}, K(\mathbf{A}'_k, \cdot)) = F_{ij}(\mathbf{A}'_k) = A_{k|ij}$  for each  $k = 1, \dots, (r+s)$  and  $i, j = 1, \dots, n$ . As shown in [20], the minimum-norm interpolator has the form  $\tilde{F}_{ij}(\cdot) = \sum_{l=1}^{r+s} C_{l|ij} K(\mathbf{A}'_l, \cdot)$  where the coefficients  $C_{l|ij}$  are the unknown parameters. Evaluation of the function  $\tilde{F}_{ij}$  at the points  $\mathbf{X} = \mathbf{A}'_k$  yields the following system of simultaneous equations,

$$A_{k|ij} = \sum_{l=1}^{r+s} C_{l|ij} K(\mathbf{A}'_l, \mathbf{A}'_k), \quad k = 1, \dots, r+s, \quad (7)$$

for  $i, j = 1, \dots, n$ . Assembling these results into a matrix, we obtain

$$\mathbf{A}_k = \sum_{l=1}^{r+s} \mathbf{C}_l G_{lk} \quad (8)$$

where  $\mathbf{C}_l := (C_{l|ij})$ ,  $G_{lk} := K(\mathbf{A}'_l, \mathbf{A}'_k)$  and  $k = 1, \dots, r+s$ .

By introducing the matrices  $\mathbf{A} \in \mathbb{R}^{n^2 \times (r+s)}$  and  $\mathbf{G} \in \mathbb{R}^{(r+s) \times (r+s)}$  where  $\mathbf{A} := (\text{vec } \mathbf{A}_1, \dots, \text{vec } \mathbf{A}_{r+s})$  and  $\mathbf{G} := (\mathbf{G}_1, \dots, \mathbf{G}_{r+s})$ , we can express the complete problem in the form of a single matrix equation, namely  $\mathbf{A} = \mathbf{C} \mathbf{G}$ , where the only unknown is the matrix  $\mathbf{C} := (\text{vec } \mathbf{C}_1, \dots, \text{vec } \mathbf{C}_{r+s})$ . Conditions under which the Gram matrix  $\mathbf{G}$  is invertible are stated in [23]. These conditions are assumed to be satisfied by the attribute matrices of an attributed graph, and hence the coefficients of the interpolator are described by the matrix expression,

$$\mathbf{C} = \mathbf{A} \mathbf{G}^{-1}. \quad (9)$$

Up to this point the development coincides with that of the RIGM algorithm [21]. The way in which the permutation sub-matrix is inferred is totally different and is presented next.

### 3.2 Inferring the Unconstrained Kronecker Match Matrix

**Proposition 1.** *An Unconstrained Kronecker Match Matrix is given by*

$$\hat{\Phi} = \begin{pmatrix} \hat{\Phi}_1^T \\ \vdots \\ \hat{\Phi}_{n^2}^T \end{pmatrix} \quad (10)$$

where

$$\hat{\Phi}_i^T = \left( \sum_{l=1}^{r+s} C_{l|i} \text{vec } \mathbf{A}'_l \right)^T, \quad (11)$$

$\mathbf{C} := (\text{vec } \mathbf{C}_1, \dots, \text{vec } \mathbf{C}_{r+s}), \text{vec } \mathbf{C}_l := (C_{l|i})$  and  $i = 1, \dots, n^2$ .

*Proof.* Eq. (8) can be written in the form

$$\text{vec } \mathbf{A}_k = \text{vec} \left( \sum_{l=1}^{r+s} \mathbf{C}_l G_{l k} \right) = \sum_{l=1}^{r+s} \text{vec}(\mathbf{C}_l) G_{l k} = \mathbf{C} \mathbf{G}_k \quad (12)$$

with  $\mathbf{C} := (\text{vec } \mathbf{C}_1, \dots, \text{vec } \mathbf{C}_{r+s})$  and  $\mathbf{G}_k := (G_{1 k}, \dots, G_{r+s, k})^T$ . By comparing Eq. (12) with Eq. (3) it is clear that

$$\text{vec } \mathbf{A}_k = \mathbf{C} \mathbf{G}_k = \widehat{\Phi} \text{vec } \mathbf{A}'_k. \quad (13)$$

Expanding Eq. (13) and performing a row-wise decomposition we obtain

$$A_{k|i} = \left( \sum_{l=1}^{r+s} C_{l|i} \text{vec } \mathbf{A}'_l \right)^T \text{vec } \mathbf{A}'_k, \quad (14)$$

where  $\text{vec } \mathbf{A}_k := A_{k|i}, k = 1, \dots, r + s$  and  $i = 1, \dots, n^2$ . From Eq. (14) it follows that  $\widehat{\Phi}_i^T = \left( \sum_{l=1}^{r+s} C_{l|i} \text{vec } \mathbf{A}'_l \right)^T$  which concludes the proof.  $\square$

### 3.3 Inferring an Approximation to $\mathbf{P}$ from $\widehat{\Phi}$

**Observation 2** *If an Unconstrained Permutation Sub-matrix  $\widehat{\mathbf{P}}$  is used as the weight matrix of an optimal assignment problem, the solution of which is a permutation sub-matrix  $\overline{\mathbf{P}}$  representing the optimal assignment of  $\widehat{\mathbf{P}}$ , then the permutation sub-matrix  $\overline{\mathbf{P}}$  is the closest to  $\widehat{\mathbf{P}}$  in the mean-square-error sense, that is,  $\|\overline{\mathbf{P}} - \widehat{\mathbf{P}}\|_F$  is a minimum where  $\|\cdot\|_F$  denotes the Frobenius norm [22].*

Some optimal assignment algorithms might require  $\widehat{\mathbf{P}}$  to be a positive matrix. If  $\widehat{\mathbf{P}}$  is not a positive matrix, we can add a constant to  $\widehat{\mathbf{P}}$  to ensure that all its elements are non-negative.

**Proposition 2.** *Let  $\overline{\mathbf{P}}$  be the permutation sub-matrix closest to  $\widehat{\mathbf{P}}$  in the mean-square-error sense. If the Unconstrained Kronecker Match Matrix is given as  $\widehat{\Phi} = \widehat{\mathbf{P}} \otimes \widehat{\mathbf{P}}$ , then  $\overline{\mathbf{P}}$  will also be the permutation sub-matrix closest to  $\widehat{\mathbf{P}}$  in the same sense where*

$$\tilde{P}_{ij} := \frac{\sum_{k,l} \widehat{\Phi}_{kl}}{n}, \quad (15)$$

$\widehat{\Phi} := \widehat{\Phi}_{kl}, i = 1, \dots, n, j = 1, \dots, n', k = (i-1)n+1, \dots, (i-1)n+n, l = (j-1)n'+1, \dots, (j-1)n'+n'$  and  $\sum_{i,j} \tilde{P}_{ij}$  is positive.

*Proof.* Observe that  $\frac{\sum_{k,l} \widehat{\Phi}_{kl}}{n} = \widehat{P}_{ij} \sum_{p,q} \widehat{P}_{pq}$  where  $k = (i-1)n+1, \dots, (i-1)n+n, l = (j-1)n'+1, \dots, (j-1)n'+n', p = 1, \dots, n, q = 1, \dots, n', i = 1, \dots, n$  and  $j = 1, \dots, n'$ . This leads to

$$\overline{\mathbf{P}} = \alpha \widehat{\mathbf{P}} \quad (16)$$

where  $\alpha$  is a constant. Noting that the optimal assignment procedure yields the same answer when its weight matrix is multiplied by a positive constant concludes the proof.  $\square$

By assuming that  $\widehat{\Phi}$ , as given by Proposition 1, is the Kronecker Product of an Unconstrained Permutation Sub-matrix  $\widehat{\mathbf{P}}$  with itself, we can therefore use Proposition 2 to obtain a permutation sub-matrix  $\overline{\mathbf{P}}$  which will serve as an approximation to  $\mathbf{P}$ . The resultant algorithm has a complexity of  $O(n^4)$  when  $n > (r + s)^2$  and the Kuhn-Munkres optimal assignment algorithm is used to approximate  $\mathbf{P}$ .

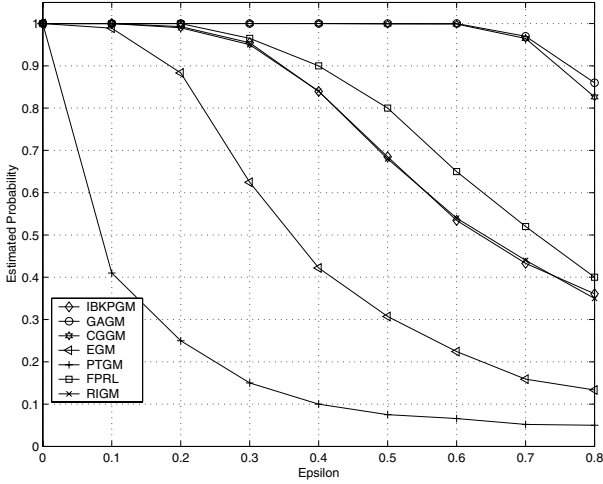
## 4 Numerical Experiments

In order to evaluate the performance of the IBKPGM algorithm, the following procedure was used: Firstly, the parameters  $n'$ ,  $n$ ,  $r$  and  $s$  were fixed. For every iteration, a reference graph  $G'$  was generated randomly with all attributes distributed between 0 and 1. An  $n \times n'$  permutation sub-matrix,  $\mathbf{P}$ , was also generated randomly, and then used to permute the rows and columns of the edge attribute adjacency matrices and the elements of the vertex attribute vectors of  $G'$ . Next, an independently generated noise matrix (vector, respectively) was added to each edge attribute adjacency matrix (vertex attribute vector, respectively) to obtain the duplicate graph  $G$ . The element of each noise matrix or vector was obtained by multiplying a random variable—uniformly distributed on the interval  $[-1/2, 1/2]$ —by the noise magnitude parameter  $\varepsilon$ . Different graph matching algorithms were then used to determine a permutation sub-matrix which approximates the original permutation sub-matrix  $\mathbf{P}$ . The proportion correct vertex-vertex assignments was calculated for a given value of  $\varepsilon$  after every 300 trials for each algorithm. From a probabilistic point of view, this approximates how well an algorithm performs for a given noise magnitude.

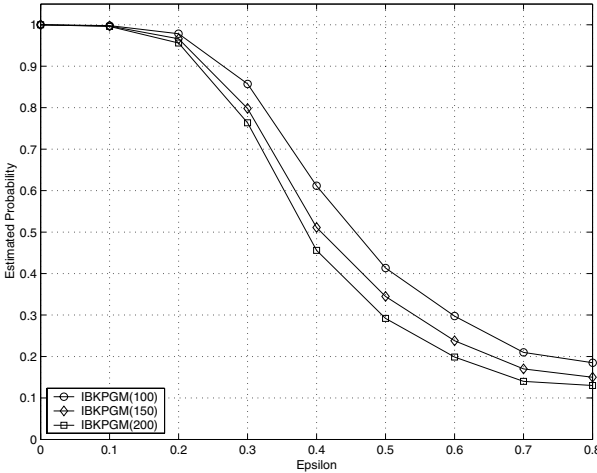
### 4.1 Numerical Results

The performance of the IBKPGM algorithm was compared to the performance of the GAGM [1], EGM [10], CGGM [24], PTGM [9] and RIGM [21] algorithms. A comparison is also made against the *Faugeras-Price Relaxation Labelling* (FPRL) algorithm [25]. The computational complexity of the IBKPGM algorithm is  $O(n^4)$ .

Figure 4.1 presents the estimated probability of correct vertex-vertex match as a function of noise magnitude  $\varepsilon$  for the case  $(n, r, s) = (30, 3, 3)$ . Globally, the performance curves for the GAGM and CGGM algorithms are closely spaced and well separated from the performance curves of the IBKPGM and FPRL algorithms for large values of  $\varepsilon$ , which in turn are well separated from the performance curves of the PTGM and EGM algorithms. In figure 2, the performance of the IBKPGM algorithm is shown for values of  $n = 100, 150$ , and  $200$  where  $r = 3$  and  $s = 3$ . Figure 3 depicts the estimated probability of correct vertex-vertex

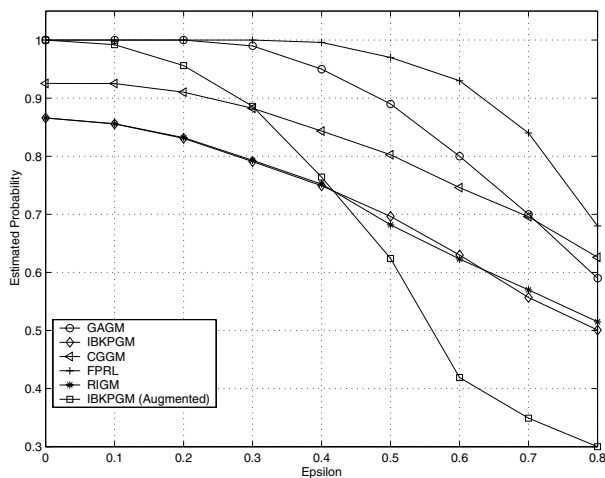


**Fig. 1.** Matching of  $(30,3,3)$  attributed graphs: Estimated probability of correct vertex-vertex matching versus  $\epsilon$



**Fig. 2.** Matching of  $(100,3,3)$ ,  $(150,3,3)$  and  $(200,3,3)$  attributed graphs: Estimated probability of correct vertex-vertex matching versus  $\epsilon$

matching for the case  $(n'/n, r, s) = (15/5, 5, 5)$ . The results of this experiment indicate that the estimated probability of a correct vertex-vertex match is higher than 0.8 for noise values up to nearly 0.3 when a third of the vertices are missing. When only two or three vertices are missing, the experiment is trivial, since the IBKPGM algorithm almost always finds the correct vertex-vertex match when no noise is present. The curve IBKPGM(Augmented) indicates the performance



**Fig. 3.** Matching of (15/5,5,5) attributed graphs: Estimated probability of correct vertex-vertex matching versus  $\varepsilon$

of the IBKPGM algorithm when five additional attributes were added to each edge and vertex of each graph. The additional attributes were derived from the existing attributes by squaring them. This procedure significantly improves the sub-graph matching performance of the IBKPGM algorithm for small values of  $\varepsilon$ .

## 5 Conclusion

The *Kronecker Product Graph Matching* (KPGM) formulation was presented and the *Interpolator-Based Kronecker Product Graph Matching* (IBKPGM) algorithm, based on this formulation, was introduced. The IBKPGM algorithm incorporates a general approach to a wide class of graph matching problems.

It was demonstrated that the performance of the IBKPGM algorithm is comparable to the performance of a typical gradient-based relaxation method such as the FPRL algorithm when performing full-graph matching. The performance curves of the IBKPGM are almost identical to those of the RIGM [21] algorithm. This phenomenon cannot be explained at present, and is a topic for further investigation.

## References

1. Gold, S., Rangarajan, A.: A Graduated Assignment Algorithm for Graph Matching, *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. 18(4)(1996)377–388 74, 79
2. You, M., Wong, K. C.: An Algorithm for Graph Optimal Isomorphism, *Proc. ICPR*, (1984) 316–319
3. Tsai, W.-H., Fu, K.-S.: Error-Correcting Isomorphisms of Attributed Relation Graphs for Pattern Recognition, *IEEE Trans. Syst. Man Cybern.*, Vol. 9(1979)757–768



4. Tsai, W.-H., Fu, K.-S.: Subgraph Error-Correcting Isomorphisms for Syntactic Pattern Recognition, *IEEE Trans. Systems, Man, Cybernetics*, Vol. 13(1983)48–62
5. Depiero, F., Trivedi, M., Serbin, S.: Graph Matching using a Direct Classification of Node Attendance, *Pattern Recognition*, Vol. 29(6)(1996)1031–1048
6. Eshera, M. A., Fu, K.-S.: A Graph Distance Measure for Image Analysis, *IEEE Trans. Syst. Man Cybern.*, Vol. 14(3)(1984)
7. Bunke, H., Messmer, B.: Recent Advances in Graph Matching, *Int. J. Pattern Recognition Artificial Intell.*, Vol. 11(1)(1997)169–203
8. Almohamad, H. A. L., Duffuaa, S. O.: A Linear Programming Approach for the Weighted Graph Matching Problem, *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. 15(5)(1993)522–525
9. Almohamad, H. A. L.: Polynomial Transform for Matching Pairs of Weighted Graphs, *Applied Mathematical Modelling*, Vol. 15(4)(1991)216–222 79
10. Umeyama, S.: An Eigendecomposition Approach to Weighted Graph Matching Problems, *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. 10(5)(1988)695–703 79
11. Hummel, R. A., Zucker, S. W.: On the Foundations of Relaxation Labelling Processes, *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. 5(3)(1983)267–286
12. Peleg, S.: A New Probabilistic Relaxation Scheme, *IEEE Trans. Patt. Anal. Machine Intell.*, 2(4)(1980)362–369
13. Christmas, W. J., Kittler, J., Petrou, M.: Structural Matching in Computer Vision using Probabilistic Relaxation, *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. 17(8)(1995)749–764
14. Finch, A. M., Wilson, R. C., Hancock, R.: Symbolic Matching with the EM Algorithm, *Pattern Recognition*, Vol. 31(11)(1998)1777–1790
15. Williams, M. L., Wilson, R. C., Hancock, E. R.: Multiple Graph Matching with Bayesian Inference, *Pattern Recognition Letters*, Vol. 18(1997)1275–1281
16. Cross, A. D. J., Hancock, E. R.: Graph Matching with a Dual Step EM Algorithm, *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. 20(11)(1998)1236–1253
17. Wilson, R. C., Hancock, E. R.: A Bayesian Compatibility Model for Graph Matching, *Pattern Recognition Letters*, Vol. 17(1996)263–276
18. Cross, A. D. J., Wilson, C., Hancock, E. R.: Inexact Matching Using Genetic Search, *Pattern Recognition*, Vol. 30(6)(1997)953–970
19. Wilson, R. C., Hancock, E. R.: Structural Matching by Discrete Relaxation, *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. 19(8)(1997)634–648
20. van Wyk, M. A., Durrani, T. S.: A Framework for Multi-Scale and Hybrid RKHS-Based Approximators, *IEEE Trans. Signal Proc.*, Vol. 48(12)(2000)3559–3568 76, 77
21. van Wyk, M. A., Durrani, T. S., van Wyk, B. J.: A RKHS Interpolator-Based Graph Matching Algorithm, To appear in *IEEE Trans. Patt. Anal. Machine Intell.*, July 2002 76, 77, 79, 81
22. Van Wyk, M. A., Clark, J.: An Algorithm for Approximate Least-Squares Attributed Graph Matching, in *Problems in Applied Mathematics and Computational Intelligence*, N. Mastrokakis (ed.), World Science and Engineering Society Press, (2001) 67-72 78
23. Luenberger, D. G.: *Optimization by Vector Space Methods*, New York, NY: John Wiley & Sons, 1969 77
24. van Wyk, B. J., van Wyk, M. A., Virolleau, F.: The CGGM Algorithm and its DSP implementation, *Proc. 3rd European DSP Conference on Education and Research, ESIEE-Paris*, 20-21 September, 2000 79

25. Faugeras, O. D., Price, K. E.: Semantic Description of Aerial Images Using Stochastic Labeling, *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. 3(6)(1981)633–642 79