Probabilistic Subgraph Matching Based on Convex Relaxation

Christian Schellewald and Christoph Schnörr

Computer Vision, Graphics, and Pattern Recognition Group
Department of Mathematics and Computer Science;
University of Mannheim, D-68131 Mannheim, Germany

{cschelle,schnoerr}@ti.uni-mannheim.de
http://www.cvgpr.uni-mannheim.de

Abstract. We present a novel approach to the matching of subgraphs for object recognition in computer vision. Feature similarities between object model and scene graph are complemented with a regularization term that measures differences of the relational structure. For the resulting quadratic integer program, a mathematically tight relaxation is derived by exploiting the degrees of freedom of the embedding space of positive semidefinite matrices. We show that the global minimum of the relaxed convex problem can be interpreted as probability distribution over the original space of matching matrices, providing a basis for efficiently sampling all close-to-optimal combinatorial matchings within the original solution space. As a result, the approach can even handle completely ambiguous situations, despite uniqueness of the relaxed convex problem. Exhaustive numerical experiments demonstrate the promising performance of the approach which – up to a single inevitable regularization parameter that weights feature similarity against structural similarity – is free of any further tuning parameters.

1 Introduction

Recognition of objects by matching relational structures of local features is a key problem of computer vision. Since such structures were suggested for image analysis in 1971 by Barrow and Popplestone [1], a very broad range of approaches have been suggested to cope with the inherent combinatorial complexity of the corresponding matching problem. A non-exhaustive list of relevant work includes tree search algorithms [2], evolutionary strategies [3], spectral approaches [4, 5], the expectation maximization framework [6], matching of structures in terms of generalized maximum clique search [7], interpolation-based matching [8], metric embedding [9], matching by graph seriation and sequence alignment [10], and exact probabilistic inference using sparse graphical models with computationally feasible junction trees [11]. Since a general discussion of related work is beyond the scope of this contribution, we refer to [12] for a recent survey.

The specific motivation for our work comes from two different directions. The first line of research originates from independent work of Gold and Rangarajan

[13] and Ishii and Sato [14] on deterministic annealing strategies for the matching of relational structures with *equal* number of nodes over the convex hull of all permutation matrices (cf. also [15, 16]). The second line of research concerns specific instances of the general pattern of convex relaxations of combinatorial integer programming problems [17, 18] to the specific problem addressed in [13, 14], the quadratic assignment problem [19, 20]. Using established benchmark tests [21], a thorough experimental comparison of both approaches [22] revealed similar performance, provided the used parameters for deterministic annealing [13, 14] are optimized for each problem instance, whereas no parameter tuning is necessary for a convex relaxation approach [19, 20].

The applicability of approaches related to the quadratic assignment problem is limited to the matching of relational structures with feature sets of (almost) equal cardinality. From the viewpoint of computer vision, such approaches are not applicable to the more frequent scenario of matching *smaller model graphs* representing typical object views, to *larger scene graphs* representing current observations – see Figure 1 for an illustration.

The present paper is an attempt to overcome this limitation by a novel optimization approach to subgraph matching. From the computational viewpoint, we consistently use semidefinite relaxation, as motivated by the discussion above, and by its performance in connection various combinatorial problems in computer vision [23, 24]. Besides obtaining parameter-free algorithms, we point out an additional benefit of this relaxation strategy – the interpretation of the globally optimal solution to the relaxed problem as probability distribution over the original combinatorial solution space. While this possibility is obvious from the mathematical viewpoint, it is by no means clear that this conveys useful information about the complex original solution space. This interpretation shows, however, that uniqueness in a larger embedding space is associated with the explicit representation of multiple hypotheses in the original space. In particular, our approach can cope with ambiguous situations. This accounts for a another novel aspect of our contribution.

Organization. We design a variational problem to subgraph matching in section 2. The domain of the resulting quadratic functional is the space of binary matching matrices. The optimality of the matchings is defined in terms of feature similarities and structural similarities, weighted by a single regularization parameter. In section 3, a semidefinite relaxation of this combinatorial optimization problem is derived. The additional degrees of freedom of the larger embedding space are exploited to incorporate constraints of the original problem formulation, thus tightening the relaxation mathematically. A probabilistic interpretation of the corresponding globally optimal solution and its ability to cope with ambiguous situations, is discussed in section 4. We summarize exhaustive numerical experiments in section 5 that characterize the performance of our approach.

Notation. We will use the following notation throughout this paper:
$x^\top$: transpose of x; I_n: $n \times n$ unit matrix; e_n: vector of all ones: $(e_n)_i = 1$, $i = 1, \ldots, n$; E_{nn}: matrix of all ones: $E_{nn} = ee^\top$; $\mathrm{Tr}[X]$ trace of the matrix X; $A \otimes B$:

Kronecker product of matrices A and B; δ_{ij}: Kronecker delta: $\delta_{ij} = 1$ if $i = j$, and 0 otherwise; $\mathcal{M}^{n \times m}$: set of $n \times m$ matching matrices; diag(X): vector of the diagonal elements of the matrix X.

2 Variational Approach

In this paper, we consider undirected graphs $G = (V, E)$ with nodes $V = \{1, \ldots, n\}$ and edges $E \subset V \times V$. We denote the model graph with G_K and the scene graph with G_L. The corresponding sets V_K and V_L contain $K = |V_K|$ and $L = |V_L|$ nodes respectively. We assume $L \geq K$. Furthermore, we assume a distance function $w(i, j)$ to be given which measures the similarity of each pair of vertices $i \in V_K$ and $j \in V_L$.

Graphs representing object views are called *model graphs* or *object graphs* in this paper. In the same way as model graphs, *scene graphs* are computed by extracting local image features and spatial relationships in a preprocessing step. Our aim is to find a reasonable matching between the nodes of the model graph and the scene graph. In figure 1 a example for a subgraph matching problem is shown. The left object graph G_K has to be matched against the scene graph G_L. We do not discuss image preprocessing in this paper (cf. section 5.1, first paragraph) but assume the model and scene graphs to be given, along with a similarity between the nodes of the different graphs.

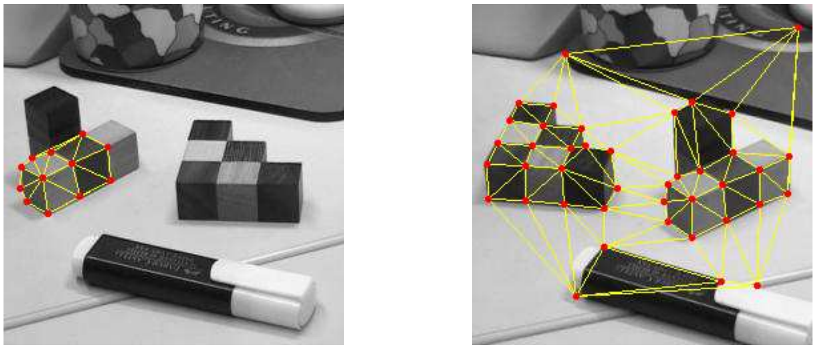


Fig. 1. The object graph G_K (left) with $K = 12$ nodes has to be matched against the scene graph (left) with $L = 41$ nodes. Local feature information is ambiguous.

2.1 Bipartite Matching

If we ignore the structure in both the model and the scene graph, then an optimal assignment of the K vertices of the model graph can be easily found as a matching in the bipartite graph $(V_K \cup V_L, E)$, with edges $(i,j) \in E$, defined for all pairs $i \in V_K, j \in V_L$ with corresponding weights $w(j,i)$.

Let $x \in \{0,1\}^{KL}$ denote the 0/1-indicator vector for the bipartite matching between the nodes of the object and scene graph. A element $X_{ji} = 1$ indicates that the node i of the first set V_K is matched to the node j in the second set V_L. The elements of the indicator vector are ordered as follows:

$$x = (X_{11}, \cdots, X_{L1}, X_{12}, \cdots, X_{L2}, \cdots, X_{1K}, \cdots, X_{LK})^\top . \tag{1}$$

Thus the indicator vector x can be interpreted as a sequence of appended columns of a matching matrix $X \in \mathcal{M}^{L \times K}$. With the same order we denote the weight vector $(w(1,1), \cdots w(L,K))^\top$ with w.

Then using $A_K = I_K \otimes e_L^\top$ and $A_L = e_K^\top \otimes I_L$, the optimal bipartite matching between the two node sets can be found by solving the following linear integer program:

$$\min_x \; w^\top x \quad \text{s.t.} \;\; A_K x = e_K \;, \; A_L x \leq e_L \;, \; x \in \{0,1\}^{KL} \tag{2}$$

The constraints ensure that the feasible vectors x all represent a bipartite mapping. The totally unimodular matrix $A = (A_K^\top, A_L^\top)^\top$ along with the integer valued data of the equality and inequality constraints guarantees that (2) can easily be solved by the following linear program which has a integral solution (cf., e.g., [25, 26]):

$$\min_x \; w^\top x \quad \text{s.t.} \;\; A_K x = e_K \;, \; A_L x \leq e_L \;, \; x \geq 0 \tag{3}$$

2.2 Quadratic Integer Program

To incorporate the relational structure of both the model graph and the scene graph, we extend the linear integer program (2) with a quadratic term $x^\top Q x$. The non-negative parameter $\alpha \in \mathbb{R}^+$ is added to control the influence of these additional costs. Formally the quadratic integer program then reads:

$$\min_x \; w^\top x + \alpha x^\top Q x \quad \text{s.t.} \;\; A_K x = e_K \;, \; A_L x \leq e_L \;, \; x \in \{0,1\}^{KL} \tag{4}$$

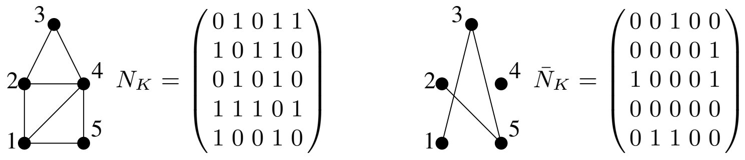
As before, the matching constraints are defined by the linear constraints. The matrix $Q \in \mathbb{R}^{KL \times KL}$ in the quadratic term of (4) to be specified below involves the symmetric 0/1-adjacency matrices N_K, N_L of the model graph and the scene graph, respectively, which encode the neighborhood structure in these two graphs. To simplify the notation we define also the *Complementary Adjacency Matrices.*

Definition 1. *Complementary Adjacency Matrices*

$$\bar{N}_L = E_{LL} - N_L - I_L \qquad \bar{N}_K = E_{KK} - N_K - I_K$$

These matrices can be interpreted as indicator matrices for *non-adjacent* nodes. They have the element $(\bar{N})_{ij} = 1$ if the corresponding nodes i and j are not directly connected in the graph.

For example, the adjacency matrix N_K and the appropriate complementary adjacency matrix for a house-like model graph are shown in figure 2. With this notation and referring to the order of the set of edges defined in (1), the symmetric *Relational Structure Matrix* Q in (4) incorporating the relational structure is defined in the following.



$$N_K = \begin{pmatrix} 0\,1\,0\,1\,1 \\ 1\,0\,1\,1\,0 \\ 0\,1\,0\,1\,0 \\ 1\,1\,1\,0\,1 \\ 1\,0\,0\,1\,0 \end{pmatrix} \qquad \bar{N}_K = \begin{pmatrix} 0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,1 \\ 1\,0\,0\,0\,1 \\ 0\,0\,0\,0\,0 \\ 0\,1\,1\,0\,0 \end{pmatrix}$$

Fig. 2. Example object graph and its adjacency matrix N_K along with its complementary adjacency matrix $\bar{N}_K$.

Definition 2. *Relational Structure Matrix*

$$Q = N_K \otimes \bar{N}_L + \bar{N}_K \otimes N_L \tag{5}$$

We explain in detail the two terms on the right hand side of (5) which are used to construct the matrix Q:

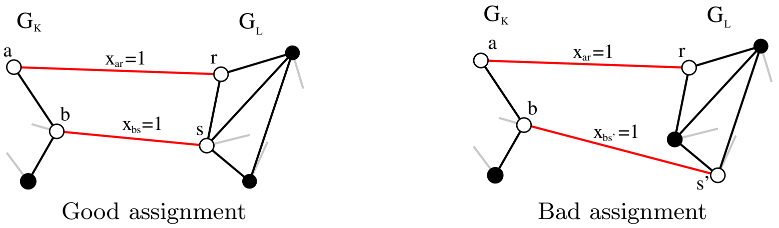
- The first term in the quadratic expression $x^\top Qx$ can be written as:

$$x^\top (N_K \otimes \bar{N}_L)x = \sum_{ar}^{KL} \sum_{bs}^{KL} (N_K)_{ab}(\bar{N}_L)_{rs} x_{ar} x_{bs} \tag{6}$$

  The interpretation of this term is that if two nodes a and b in the model graph are neighbors, $(N_K)_{ab} = 1$, then a good assignment (no costs) involves corresponding nodes r and s in the scene graph which are neighbors, too: $(\bar{N}_L)_{rs} = 0$. For such a configuration no cost is added in (6). Otherwise if the corresponding nodes r and s are no neighbors in the scene graph, $(\bar{N}_L)_{rs} = 1$, then a cost of 1 is added. This two configurations are visualized in figure 3.
- Analogously, the second term in $x^T Qx$ gives:

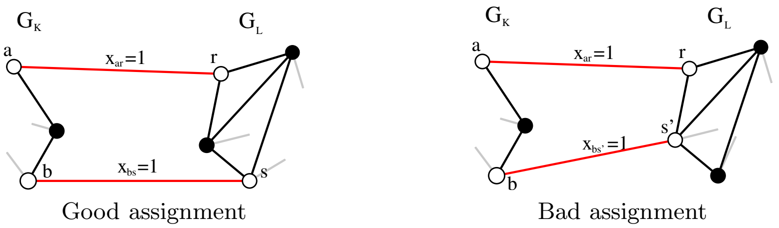
$$x^\top (\bar{N}_K \otimes N_L)x = \sum_{ar}^{KL} \sum_{bs}^{KL} (\bar{N}_K)_{ab}(N_L)_{rs} x_{ar} x_{bs} \tag{7}$$

  This term penalizes assignments where pairs of nodes in the object graph become neighbors in the scene graph which were not adjacent before. Figure 4 illustrates this situation in detail.



Good assignment Bad assignment

Fig. 3. Left: Adjacent nodes a and b in the model graph G_K are assigned to adjacent nodes r and s in the scene graph G_L. **Right.** Adjacent model nodes a and b are no longer adjacent in the scene graph G_L after the assignment. The left assignment leads to no additional costs while the right undesired assignment adds 1 to the cost term (6).



Good assignment Bad assignment

Fig. 4. Left: Nodes a and b which are not adjacent in the object graph G_K are assigned to nodes which are also not adjacent in the scene graph G_L. **Right:** A pair of nodes a and b become neighbors r and s' after assignment. The left assignment is associated with no additional costs in (7). The undesired assignment on the right side adds 1 to these costs.

Note that due to the symmetry of the quadratic cost term $x^\top Qx$, every difference in the compared structure of the two graphs is penalized with a cost of 2.

In contrast to the linear bipartite matching problem (2), the computation of the global optimum of the quadratic optimization problem (4), which incorporates the object and scene structure, is intrinsically difficult (NP-hard). Therefore, we derive in the next section a tractable convex relaxation of this NP-hard problem in order to compute a "good" local minimum.

3 Convex Problem Relaxation

The combinatorial subgraph matching approach (4) will be relaxed to a (convex) semidefinite program (SDP) which has the following standard form:

$$\begin{aligned} \min \quad & \mathrm{Tr}\left[\tilde{Q}X\right] \\ \text{s.t.} \quad & \mathrm{Tr}[A_i X] = c_i \quad \text{for} \quad i = 1, \ldots, m \\ & X \succeq 0 \end{aligned} \tag{8}$$

The last constraint in (8) says that X has to be positive semidefinite. We wish to emphasize once more that this convex optimization problem can be solved with standard methods like interior point algorithms. Note that the solution of the relaxation (8) provides a lower bound to (4).

Below, we describe step by step how we derive such a semidefinite program from (4). While in section 3.1, we derive an appropriate SDP objective function, we show in section 3.2 how the bipartite matching constraints can be incorporated into the SDP (8). For more information on semidefinite programming we refer to [27].

3.1 SDP Objective Function

In order to obtain an appropriate SDP relaxation for the combinatorial subgraph matching problem, we start with reformulating the objective function of (4) into a homogeneous quadratic form. And this can be stated directly in the appropriate trace formulation of objective function for the semidefinite relaxation (8) using the cyclic commutativity of the trace:

$$f(x) = w^\top x + \alpha x^\top Q x \;=\; \begin{pmatrix} 1 \; x^\top \end{pmatrix} \begin{pmatrix} 0 & \frac{1}{2}w^\top \\ \frac{1}{2}w & \alpha Q \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix} = \mathrm{Tr}\left[\tilde{Q}X\right] \tag{9}$$

Here we denote with $\tilde{Q} \in \mathbb{R}^{(KL+1)\times(KL+1)}$ and $X \in \mathbb{R}^{(KL+1)\times(KL+1)}$ the following symmetric matrices:

$$\tilde{Q} = \begin{pmatrix} 0 & \frac{1}{2}w^\top \\ \frac{1}{2}w & \alpha Q \end{pmatrix} \;, \quad X = \begin{pmatrix} 1 \\ x \end{pmatrix} \begin{pmatrix} 1 \; x^\top \end{pmatrix} = \begin{pmatrix} 1 & x^\top \\ x & xx^\top \end{pmatrix} \tag{10}$$

Besides being symmetric, the matrix X is positive semidefinite and has rank 1. We relax the objective function by dropping the rank 1 condition of X which makes the set of feasible matrices convex [27]. This lifts the original problem (4) defined in a vector space with dimension KL into the space of symmetric, positive semidefinite matrices with the dimension $(KL+1) \times (KL+1)$.

3.2 SDP Constraints

We wish to incorporate several constraints into the SDP relaxation by specifying appropriate constraint matrices $A_i \in \mathbb{R}^{(KL+1)\times(KL+1)}$. These SDP constraints will have the form:

$$\mathrm{Tr}[A_i X] = c_i \quad \text{for} \quad i = 1, \ldots, m$$

In particular, we introduce four types of constraints which correspond to the homogeneous formulation of the problem, the 0/1-integer constraints, and the bipartite matching constraints, respectively.

We next discuss in detail how the appropriate constraint matrices A_i can be defined in terms of the Kronecker delta which make the implementation of our approach easier:

- The first constraint we take into account results from the homogenization (9). To restrict the element $X_{11} = 1$ in the matrix X, we introduce a constraint matrix ${}^{\text{one}}A$ whose elements can be expressed as

$$ {}^{\text{one}}A_{kl} = \delta_{k1}\delta_{l1} \quad \text{for} \quad k,l = 1,\dots,KL+1 \ , $$

  where we make use of the Kronecker delta. Note that ${}^{\text{one}}A$ has only ${}^{\text{one}}A_{11} = 1$ as non-zero element.
- The second type of constraint we consider is derived from the integer constraints $x_i \in \{0,1\}, i = 1,\dots,KL$, which can be rewritten as $x_i^2 = x_i, i = 1,\dots,KL$. If we consider the matrix X before it is relaxed (see (10)) we observe that due to $x_i^2 = x_i$ the 0/1-integer elements on the diagonal of X must be equal to the 0/1-integer elements in the first column and row of X. Therefore the 0/1-integer constraints can be *weakly* enforced in the relaxed problem by requiring the first column and row of X to be equal to its diagonal. To implement these constraints, we introduce KL constraint matrices ${}^{\text{int}}A^j \in \mathbb{R}^{(KL+1)\times(KL+1)}, j = 2,\dots,KL+1$. We define these constraint matrices to have a 2 at the appropriate diagonal element and -1 at the corresponding elements in the first column and the first row. All other elements are zero. Using the Kronecker delta the elements of the j-th constraint matrix ${}^{\text{int}}A^j$ can be written as:

$$ {}^{\text{int}}A^j_{kl} = 2\delta_{kj}\delta_{lj} - \delta_{kj}\delta_{l1} - \delta_{lj}\delta_{k1} \quad \text{for} \quad k,l = 1,\dots,KL+1 $$

- The third type of constraint we take into account are the equality constraints $\sum_{j=1}^{L} x_{ij} = 1, i = 1,\dots,K$, which are part of the bipartite matching constraints in (4). They represent the constraint that each node of the smaller graph is mapped to exactly one node of the scene graph. We define K constraint matrices ${}^{\text{sum}}A^j \in \mathbb{R}^{(KL+1)\times(KL+1)}, j = 1,\dots,K$ which ensure (taking the order of the diagonal elements into account) that the sum of the appropriate portion of the diagonal elements of X is 1. We exploited again the fact that $x_i = x_i^2$ holds true for 0/1-variables. The matrix elements for the j-th constraint matrix ${}^{\text{sum}}A^j$ can be expressed as follows:

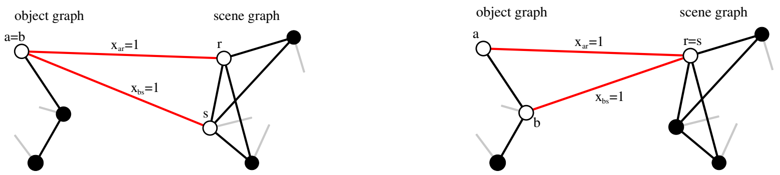
$$ {}^{\text{sum}}A^j_{kl} = \sum_{i=(j-1)L+1}^{jL+1} \delta_{ik}\delta_{il} \quad \text{for} \quad k,l = 1,\dots,KL+1 $$

- The fourth type of constraint is related to the observation that the bipartite matching constraints in (2) have a direct impact to certain matrix elements of the sub-matrix $\tilde{X} = xx^\top$ of X. If $x \in \{0,1\}^{KL}$ represents a bipartite matching then certain elements in $\tilde{X}$ must be zero. Affected elements can be determined by inspecting the following two cost terms which penalize matchings that do not meet the bipartite matching constraints.

$$x^\top (I_K \otimes (E_{LL} - I_L))x = \sum_{ar}^{KL} \sum_{bs}^{KL} (I_K)_{ab}(E_{LL} - I_L)_{rs} x_{ar} x_{bs} \tag{11}$$

$$x^\top ((E_{KK} - I_K) \otimes I_L)x = \sum_{ar}^{KL} \sum_{bs}^{KL} (E_{KK} - I_K)_{ab}(I_L)_{rs} x_{ar} x_{bs} \tag{12}$$

The first of these two terms penalizes non-unique assignments of model nodes to scene nodes. Analogously, the second term penalizes assignments where different nodes of the model graph are mapped to the same node in the scene graph. Thus, in summary, the two terms penalize all assignments which do no lead to a bipartite matching. Figure 5 illustrates such configurations in detail. All integer solutions $\tilde{X} = xx^\top \in \mathbb{R}^{KL \times KL}$, where x represents a bipartite matching, have zero-values at those matrix positions where $I_K \otimes (E_{LL} - I_L)$ and $(E_{KK} - I_K) \otimes I_L$ have non-zero elements. Accordingly, we want to force the corresponding elements in $X \in \mathbb{R}^{(KL+1)\times(KL+1)}$ to be zero. Fortunately, this can be achieved with the constraint matrices ${}^{\text{zeros1}}A^{ars}, {}^{\text{zeros2}}A^{\hat{s}\hat{a}\hat{b}} \in \mathbb{R}^{(KL+1)\times(KL+1)}$ which are determined by the indices a, r, s and $\hat{s}, \hat{a}, \hat{b}$. They have the following matrix elements



Bad assignment

Bad assignment

Fig. 5. Assignments which do not lead to bipartite matchings are penalized by the quadratic terms (11) and (12).

$${}^{\text{zeros1}}A_{kl}^{ars} = \delta_{k,(aL+r+1)} \delta_{l,(aL+s+1)} + \delta_{k,(aL+s+1)} \delta_{l,(aL+r+1)} \quad , \tag{13}$$

$${}^{\text{zeros2}}A_{kl}^{\hat{s}\hat{a}\hat{b}} = \delta_{k,(\hat{s}K+\hat{b}+1)} \delta_{l,(\hat{s}K+\hat{a}+1)} + \delta_{k,(\hat{s}K+\hat{a}+1))} \delta_{l,(\hat{s}K+\hat{b}+1)} \quad , \tag{14}$$

where $k, l = 1, \ldots, KL+1$. Note that each of these matrices has only two non-zero matrix elements at symmetric positions. The indices a, r, s and $\hat{s}, \hat{a}, \hat{b}$ attain all valid combinations of the following triples where $s > r$ and $\hat{b} > \hat{a}$:

$$(a, r, s): \; a = 1, \ldots, K; \; r = 1, \ldots, L; \; s = (r+1), \ldots, L$$

$$(\hat{s}, \hat{a}, \hat{b}): \; \hat{s} = 1, \ldots, L; \; \hat{a} = 1, \ldots, K; \; \hat{b} = (\hat{a}+1), \ldots, K$$

With this we define $(LL - L)K/2 + (KK - K)L/2$ additional constraints that ensure zero-values at the corresponding matrix positions of X.

Altogether we have the following $1 + KL + K + (LL - L)K/2 + (KK - K)L/2$ SDP constraints:

$$\begin{aligned}
\mathrm{Tr}[{}^{\mathrm{one}}AX] &= 1 \\
\mathrm{Tr}[{}^{\mathrm{int}}A^j X] &= 0 \quad \text{for} \quad j = 2\ldots, KL+1 \\
\mathrm{Tr}[{}^{\mathrm{sum}}A^j X] &= 1 \quad \text{for} \quad j = 1, \ldots, K \\
\mathrm{Tr}[{}^{\mathrm{zeros1}}A^{ars} X] &= 0 \quad \forall (a, r, s) \quad , \mathrm{Tr}[{}^{\mathrm{zeros2}}A^{\hat{s}\hat{a}\hat{b}} X] = 0 \quad \forall (\hat{s}, \hat{a}, \hat{b})
\end{aligned}$$

The name *gangster operator* was introduced in [28] for the last two constraint operators because they "shoot holes",i.e. zeros, into the matrix X.
We note here that we dropped the additional linear *inequality* constraints of the bipartite matching, $\sum_{i=1}^K x_{ij} \leq 1, \forall j$, which, in principle, can be incorporated by lifting schemes (see e.g. [27]). This, however, would considerably increase the number of constraints and slow down the computation. Our experiments (section 5) show that this does not compromise the performance of our approach.

4 Combinatorial Solutions by Post-Processing

The diagonal elements of the global optimum $X_{bound} \in \mathbb{R}^{(KL+1)\times(KL+1)}$ to the semidefinite relaxation (8) can be interpreted as a non-integer approximation $\hat{x}_{sol} = \mathrm{diag}(X_{bound})$ to the solution of (4). Omitting the first element in $\hat{x}_{sol} \in \mathbb{R}^{KL+1}$, which was added due to the homogenization (9), we obtain the approximation $x_{sol} \in \mathbb{R}^{KL}$ for the indicator vector $x \in \{0, 1\}^{KL}$.

4.1 Probabilistic Interpretation of the Non-Integer Solution

According to the constraints $A_K x_{sol} = e_K$, we have for each node i of the model graph $\sum_j^L (x_{sol})_{ji} = 1$, $i = 1, \ldots, K$. Hence, $(x_{sol})_{ji}$ may be considered as the probability that model node i matches to scene node j. To illustrate this interpretation, figure 6 shows a completely ambiguous situation, whereas figure 7 depicts for each of the five model nodes $i = 1, \ldots, 5$ the values $(x_{sol})_{ji}$. The presence of equally likely matchings clearly shows that multiple plausible hypotheses for matchings can be represented through the convex problem relaxation. As explained next, and as validated in section 5, this property can be exploited to compute the final matching from x_{sol}.

4.2 Post-Processing

Winner-Take-All Strategy. An obvious strategy for determining the final matching is to compute among all binary vectors x representing valid matchings the vector that is maximally aligned with x_{sol}:

$$\max_x x_{sol}^\top x \quad \text{s.t. } A_K x = e_K \ , \ A_L x \leq e_L \ , \ x \in \{0, 1\}^{KL} \tag{15}$$

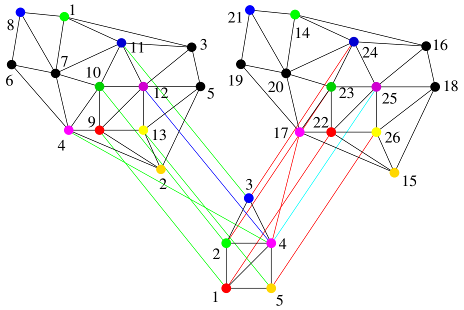


Fig. 6. Ambiguous situation with *two* global optima. The node colors indicate the similarity between the nodes of the object and the scene graph. Despite convexity and uniqueness, the semidefinite relaxation is able to represent multiple hypotheses for matching – see figure 7.

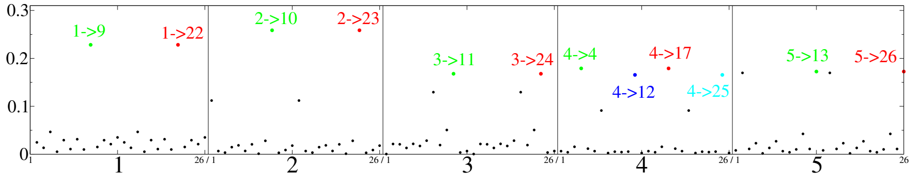


Fig. 7. The non-integer solution x_{sol} for the ambiguous matching situation shown in figure 6. The plot is subdivided into $K = 5$ segments, with the i-th segment ($i \in \{1, \ldots, K\}$) representing all possible matchings from the model node i to all $L = 26$ nodes in the scene graph. In each segment the probabilities sum up to one.

The exact solution to (15), denoted with x^*_{lin}, can be computed by solving a linear program because the constraint matrices A_K and A_L are total unimodular (cf. section 2.1). According to the probabilistic interpretation, x^*_{lin} represents the most probable matching.

Sampling. To exploit alternative hypotheses for valid matchings as well, we may randomly select a node $i \in \{1, \ldots, K\}$ of the model graph and assign to it a scene node $j \in \{1, \ldots, L\}$ by sampling from the distribution $(x_{sol})_{ji}, j = 1, \ldots, L$. This assignment is only accepted if it results in a valid matching representing an *improved* combinatorial solution. In our experiments, we conducted $10 \cdot KL$ such sampling steps, starting with the solution x^*_{lin} to (15). The resulting matching is denoted with $x^*_{sampling}$.

5 Experiments

5.1 Real World Example

For the problem shown in figure 1, we computed feature similarities (weights w) by determining the earth mover distance [29] between local gray value histograms for each node. We point out that more elaborate feature selection or even learning is beyond the scope of this paper, whose main focus is the optimization.

To demonstrate that the regularization in (4) is indeed necessary, we first computed the bipartite matching (3). The corresponding solution, shown in figure 8, just picks the locally best fitting scene graph nodes. Only the assignments drawn in red are correct.

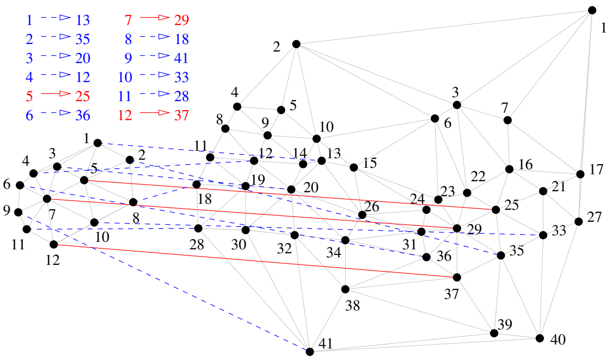


Fig. 8. Matching computed by the linear program (3) without regularization, i.e. sensitivity to relational structure. Only 3 out of the 12 assignments are correct (marked with red).

The non-integer solution x_{sol} obtained by the SDP relaxation (8) of the approach (4), is shown in figure 9, where $(x_{sol})_{ji}$ is plotted for each model node $i = 1, \dots, K$. Only a few mappings $i \mapsto j$ have significantly large probabilities $(x_{sol})_{ji}$. The most likely assignments are marked with red, and some alternative candidates with $(x_{sol})_{ji} \geq 0.1$ are marked with green. The corresponding matchings are shown in figure 10. The red assignments correspond to the optimal combinatorial solution which, in turn, corresponds to the solution x^*_{lin} to (15). Sampling, therefore, cannot lead to further improvements, in this case.

5.2 Generating Random Problem Instances

In order to get a more complete picture of the performance of our approach, we conducted a large series of experiments with randomly generated problem instances. Two kinds of experiments were considered:

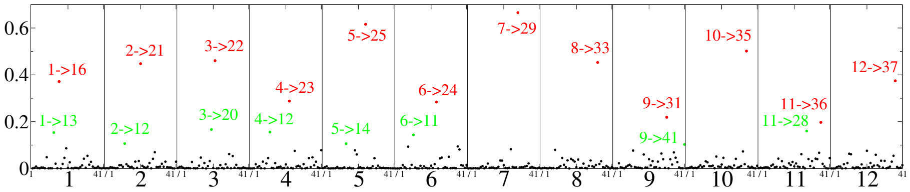


Fig. 9. The non-integer solution $(x_{sol})_{ji}$ for each model node $i = 1, \ldots, K = 12$, as obtained by the SDP relaxation. Only a few assignments have significantly large probabilities. The most likely ones are marked with red and green, respectively.

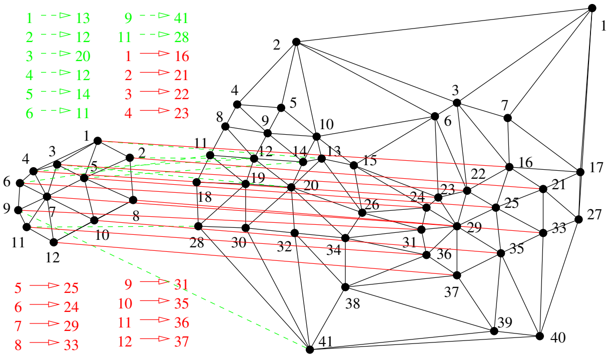


Fig. 10. SDP relaxation with winner take all post-processing results in the correct matching. An alternative hypothesis is shown by the green line segments.

Random Subgraph Problems. An object graph with K nodes is randomly created with an edge probability[1] equal to 0.5. Then the scene graph is created by copying the object graph and enlarging it to L nodes by adding $(L-K)$ random nodes and edges with edge probability 0.2. Hence, the model graph always forms a subgraph of the scene graph in this series of experiments. Costs $\{w_{ji}\}$ are selected randomly within the small range $0.4 \ldots 0.6$ if the mapping of the object node i to the scene node j represents a desired mapping. Otherwise the costs are set randomly to a value within the wider range $0.4 \ldots 1.0$. Note that the wider range includes the small range, which increases the probability that some undesired mappings have cheaper assignment costs w_{ji} than the desired mapping. Therefore, the linear matching approach (2), which ignores the graph structure, is likely to fail in all experiments.

We fixed the size of the object graph to $K = 9$ and varied the scene graph size from $L = 14$ to $L = 30$.

[1] The edge probability is the probability that an edge of the underlying complete graph is present.

By construction, we know which subgraph of the scene corresponds to the object in each experiment. Accordingly, the corresponding matching is defined to be ground truth, irrespective of the existence of other accidental matchings with a better objective value (4) in some rare cases.

Random Model and Scene Graphs. In this series of experiments both object and scene graphs with K and L nodes, respectively, are created randomly, and independently from each other. The similarities w_{ji} are set randomly within the range $0.4 \ldots 1.0$. The edge probability was set to 0.4 for the smaller model graph, and to 0.3 for the larger scene graph.

To compute ground truth, we have to rely on exhaustive search, forcing us to limit[2] the maximum size of the scene graph to $L = 19$.

For each size L, we created 1000 problem instances. The value of the regularization parameter α was 0.15 and 0.3 for the subgraph and purely random problems, respectively.

5.3 Evaluation

Fraction of Optimally Solved Problem Instances. Figure 11 shows the percentage of optimally[3] solved problem instances for various problem sizes K and L. We observe that for almost all smaller subgraph problems the global

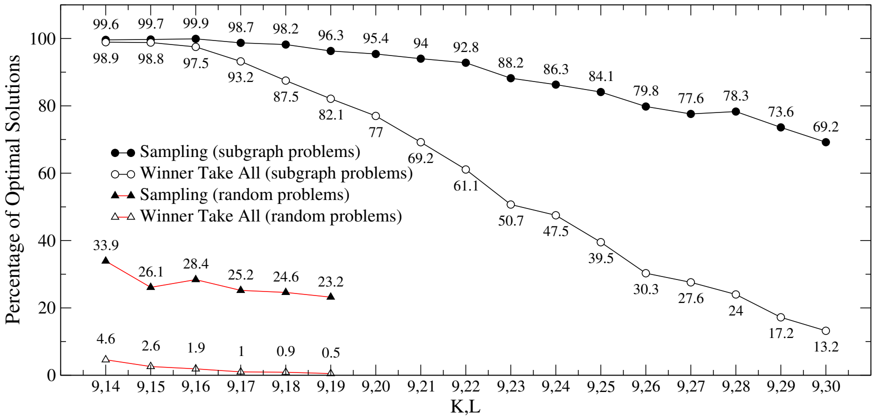


Fig. 11. Fraction of optimally solved problem instances for increasing problem sizes K and L. The sampling post-processing step significantly increases this fraction by exploiting the information in the non-integer solution vector x_{sol}.

[2] The number of possible assignments growth as $L!/(L-K)!$.

[3] We count a solution as optimal if it has an equal or better objective value than the objective value of the correct matching.

optimum is obtained by the convex relaxation followed by the winner take all post-processing step. The main observation, however, is that the sampling was always able to significantly increase the fraction of global optimal solutions. This confirms the usefulness of a probabilistic interpretation of the non-integer solution vector x_{sol} to the SDP relaxation. Furthermore, figure 11 shows that the more structured subgraph matching problems are easier to solve than the purely random problem instances.

Quality of Optima. To get a more accurate picture of the performance of our approach, we investigated the quality of the combinatorial solutions. To this end, we computed the mean values along with the standard deviation of the ratios f^*_{lin}/f^*_{opt} and $f^*_{sampling}/f^*_{opt}$ for the solutions f^*_{lin} and $f^*_{sampling}$ obtained by the winner take all and the sampling post-processing step, respectively. A ratio close to 1 indicates that the obtained solution is close to the objective value f^*_{opt} of the true matching. The results are shown in table 1. We observe that the sampling post-processing step always improves the results obtained by the winner take all post-processing, and that the corresponding mean $f^*_{sampling}$ is very close to f^*_{opt}. For example, for the larger subgraph problems with $K = 9$, $L = 30$, the sampling improves the deviation from 35% to only 5%. Again, the random problems turn out to be more difficult to solve, but sampling still leads to good solutions close to the global optimum.

	K,L	9,15	9,17	9,19	9,25	9,30
subgraph problems	f^*_{lin}/f^*_{opt}	1.00 ± 0.01	1.01 ± 0.04	1.02 ± 0.08	1.17 ± 0.21	1.35 ± 0.24
	$f^*_{sampling}/f^*_{opt}$	0.99 ± 0.01	0.99 ± 0.01	1.00 ± 0.02	1.02 ± 0.06	1.05 ± 0.09
random problems	f^*_{lin}/f^*_{opt}	1.58 ± 0.36	1.74 ± 0.42	1.99 ± 0.49	n.a.	n.a.
	$f^*_{sampling}/f^*_{opt}$	1.10 ± 0.12	1.11 ± 0.12	1.14 ± 0.14	n.a.	n.a.

Table 1. Mean and standard deviation of the optima relative to the correct solution. A ratio close to 1 indicates that the computed solution is close to f^*_{opt}. Winner take all post-processing is always inferior to sampling. Note that the ratio for the subgraph problems can become smaller than 1 due to accidental matchings with smaller objective function value than that of the correct matching.

6 Conclusion

We proposed and investigated a novel approach to subgraph matching in computer vision using regularized bipartite matching, semidefinite relaxation, and a corresponding probabilistic post-processing step. A salient property of our approach is its mathematical simplicity: high-quality approximate solutions can be computed by just solving a *convex* optimization problem. As a consequence, no additional tuning parameters related to search heuristics, etc. are needed, apart from a single regularizing parameter penalizing structural differences of matchings. Extensive numerical experiments revealed a surprisingly good quality of the suboptimal solutions. Our approach provides a basis for learning optimal feature configurations for object recognition in future work.

References

1. H. G. Barrow and R.J. Popplestone. Relational descriptions in picture processing. *Machine Intelligence*, 6:377–396, 1971.
2. J.R. Ullmann. An algorithm for subgraph isomorphism. *Journal of the ACM*, 23(1):31–42, 1976.
3. A.D.J. Cross, R.C. Wilson, and E.R. Hancock. Inexact graph matching using genetic search. *Pattern Recog.*, 30(6):953–970, 1997.
4. S. Umeyama. An eigendecomposition approach to weighted graph matching problems. *IEEE Trans. Patt. Anal. Mach. Intell.*, 10(5):695–703, 1988.
5. B. Luo and E.R. Hancock. Structural graph matching using the em algorithm and singular value decomposition. *IEEE Trans. Patt. Anal. Mach. Intell.*, 23(10):1120–1136, 2001.
6. A.D.J. Cross and E.R. Hancock. Graph matching with a dual-step em algorithm. *IEEE Trans. Patt. Anal. Mach. Intell.*, 20(11):1236–1253, 1998.
7. M. Pavan and M. Pelillo. Dominant sets and hierarchical clustering. In *Proc. ICCV 2003 - 9th IEEE International Conference on Computer Vision*, volume 1, pages 362–369, 2003.
8. B.J. Van Wyk and M.A. Van Wyk. Kronecker product graph matching. *Patt. Recognition*, 36(9):2019–2030, 2003.
9. M.F. Demirci, A. Shoukoufandeh, Y. Keselman, S. Dickinson, and L. Bretzner. Many-to-many matching of scale-space feature hierarchies using metric embedding. In L.D. Griffin and M. Lillholm, editors, *Scale-Space 2003*, volume 2695 of *Lecture Notes in Computer Science*, pages 17–32. Springer, 2003.
10. A. Robles-Kelley and E.R. Hancock. Graph edit distance from spectral seriation. *IEEE Trans. Patt. Anal. Mach. Intell.*, 27(3):365–378, 2005.
11. T. Caelli and T.S. Caetano. Graphical models for graph matching: Approximate models and optimal algorithms. *Patt. Recog. Letters*, 26(3):339–346, 2005.
12. D. Conte, P. Foggia, C. Sansone, and M. Vento. Thirty years of graph matching in pattern recognition. *IJPRAI*, 18(3):265–298, 2004.
13. S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *IEEE Trans. Patt. Anal. Mach. Intell.*, 18(4):377–388, 1996.
14. S. Ishii and M. Sato. Doubly constrained network for combinatorial optimization. *Neurocomputing*, 43:239–257, 2002.
15. J.J. Kosowsky and A.L. Yuille. The invisible hand algorithm: Solving the assignment problem with statistical pyhysics. *Neural Networks*, 7(3):477–490, 1994.
16. A. Rangarajan, A. Yuille, and E. Mjolsness. Convergence properties of the softassign quadratic assignment algorithm. *Neural Computation*, 11(6):1455–1474, 1999.
17. F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5(1):13–51, 1995.
18. S. Poljak, F. Rendl, and H. Wolkowicz. A recipe for semidefinite relaxation for 0-1 quadratic programming. *Journal of Global Optimization*, (7):51–73, 1995.
19. Q. Zhao, S.E. Karisch, F. Rendl, and H. Wolkowicz. Semidefinite programming relaxations for the quadratic assignment problem. *J. Combinat. Optimization*, 2(1):71–109, 1998.
20. K. M. Anstreicher and N. W. Brixius. A new bound for the quadratic assignment problem based on convex quadratic programming. *Mathematical Programming*, 89(3):341–357, 2001.

21. R. E. Burkard, S. Karisch, and F. Rendl. QAPLIB-A Quadratic Assignment Problem Library. *European Journal of Operational Research*, 55:115–119, 1991.
22. C. Schellewald, S. Roth, and C. Schnörr. Evaluation of convex optimization techniques for the weighted graph-matching problem in computer vision. In B. Radig and S. Florczyk, editors, *Mustererkennung 2001*, volume 2191 of *Lect. Notes Comp. Science*, pages 361–368, Munich, Germany, Sept. 12–14 2001. Springer.
23. J. Keuchel, C. Schnörr, C. Schellewald, and D. Cremers. Binary partitioning, perceptual grouping, and restoration with semidefinite programming. *IEEE Trans. Patt. Anal. Mach. Intell.*, 25(11):1364–1379, 2003.
24. J. Keuchel, M. Heiler, and C. Schnörr. Hierarchical image segmentation based on semidefinite programming. In *Pattern Recognition, Proc. 26th DAGM Symposium*, volume 3175 of *Lecture Notes in Computer Science*, pages 120–128. Springer, 2004.
25. Alexander Schrijver. *Theory of linear and integer programming.* John Wiley & Sons, Inc. New York, NY, USA, 1986.
26. B. Korte and J. Vygen. *Combinatorial Optimization: Theory and Algorithms.* Algorithms and Combinatorics 21. Springer, Berlin Heidelberg New York, 2000.
27. H. Wolkowicz, R. Saigal, and L. Vandenberghe, editors. *Handbook of Semidefinite Programming*, Boston, 2000. Kluwer Acad. Publ.
28. Ph. L. Toint. On sparse and symmetric matrix updating subject to a linear equation. *Mathematics of Computation*, 31:954–961, 1977.
29. Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.