

# A Dual Decomposition Approach to Feature Correspondence

Lorenzo Torresani, *Member, IEEE*, Vladimir Kolmogorov, and Carsten Rother, *Member, IEEE*

**Abstract**—In this paper, we present a new approach for establishing correspondences between sparse image features related by an unknown nonrigid mapping and corrupted by clutter and occlusion, such as points extracted from images of different instances of the same object category. We formulate this matching task as an energy minimization problem by defining an elaborate objective function of the appearance and the spatial arrangement of the features. Optimization of this energy is an instance of graph matching, which is in general an NP-hard problem. We describe a novel graph matching optimization technique, which we refer to as dual decomposition (DD), and demonstrate on a variety of examples that this method outperforms existing graph matching algorithms. In the majority of our examples, DD is able to find the global minimum within a minute. The ability to globally optimize the objective allows us to accurately learn the parameters of our matching model from training examples. We show on several matching tasks that our learned model yields results superior to those of state-of-the-art methods.

**Index Terms**—Graph matching, feature correspondence, dual decomposition

## 1 INTRODUCTION

FEATURE correspondence is the task of matching a sparse or dense set of image features across multiple images or volumes. It is one of the fundamental problems of computer vision and is a key ingredient in a wide range of applications, including object recognition, 3D reconstruction, mosaicing, motion segmentation, and image morphing. Several robust algorithms (see, e.g., [3], [24]) exist for registration of images of static scenes and for visual correspondence under rigid motion. These methods typically exploit powerful constraints (e.g., epipolar constraints) to reduce the search space and disambiguate the correspondence problem. However, such constraints do not apply in the case of nonrigid motion or when matching different object instances. A popular approach in these cases is to discard the information about the spatial layout of features and to find correspondences using appearance only. For example, many object recognition methods [15], [16], [20], [40], [45], [47], [49] represent images as orderless sets of local appearance descriptors known as bags of features. Recent work [11] has suggested that for many correspondence problems, learned appearance-based models perform similarly or better than state-of-the-art structural models exploiting information about spatial arrangement of features. This is primarily due to

the challenges posed by the optimization and training of structural models, which often require approximate solution of NP-hard problems. In this paper, we contrast this theory and demonstrate that a complex structural model for image matching can be learned and optimized successfully so as to obtain results superior to those achievable with models neglecting the spatial arrangement of features. We cast the visual correspondence problem as an energy minimization task by defining a complex image matching objective depending on 1) feature appearance, 2) geometric compatibility of correspondences, and 3) spatial coherence of matched features. Additionally, we impose a uniqueness constraint allowing at most one match per feature. We introduce a novel algorithm to minimize this function based on the dual decomposition approach (DD) from combinatorial optimization, see, e.g., [6], [12], [44], [36], [37], [25]. The DD method works by maximizing a lower bound on the energy function. The value of the lower bound can be used to gauge the distance from the global minimum and to decide when to stop the optimization in the event the global minimum cannot be found. For the majority of our examples DD finds the global minimum in reasonable time, and otherwise provides a solution whose cost is very close to the optimum. In contrast, previously proposed optimization methods such as [14], [17], [48] often fail to compute good solutions for our energy function. Our experimental evaluation shows that the model and the algorithm presented in this paper can be applied to a wide range of image matching problems with results matching or exceeding those of existing algorithms [11], [4].

### 1.1 Relation to Previous Work

**Models for feature matching.** Our approach is loosely related to algorithms that find visual correspondences by matching appearance descriptors under smooth, or piecewise smooth, spatial mappings. For example, Torr [42] describes a technique for estimating sparse correspondences using RANSAC under the assumption that the images

• L. Torresani is with the Department of Computer Science, Dartmouth College, 6211 Sudikoff Lab, Hanover, NH 03755.

E-mail: lorenzo@cs.dartmouth.edu.

• V. Kolmogorov is with the Institute of Science and Technology, IST Austria, Am Campus 1, 3400 Klosterneuburg, Austria.

E-mail: vladimir.kolmogorov@ist.ac.at.

• C. Rother is with Microsoft Research Cambridge, 7 JJ Thomson Avenue, Cambridge CB3 0FB, United Kingdom. E-mail: carrot@microsoft.com.

Manuscript received 12 Aug. 2011; revised 13 Feb. 2012; accepted 21 Apr. 2012; published online 1 May 2012.

Recommended for acceptance by N. Paragios.

For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2011-08-0544.

Digital Object Identifier no. 10.1109/TPAMI.2012.105.

contain a common set of rigidly moving objects. However, this piece-wise rigid motion assumption is not appropriate for deformable objects, such as human faces, or for different instances of an object class, such as different cars, related by highly nonlinear mappings. Other approaches [33], [38] have proposed constraining the correspondence problem by learning or hand-coding explicit models of how an object is allowed to deform using parametric 2D or 3D representations, such as linear eigenshapes or superquadrics. Unlike such approaches, our method does not make a parametric assumption about the transformation relating the input images, and thus can be used in a wider range of applications. Belongie et al. [4] inject spatial smoothness in the match by means of an iterative technique that alternates between finding correspondences using shape features and computing a regularized transformation aligning the matching features. The shape descriptors are recomputed in each iteration after the warping. Since the objective is changed at each iteration, the convergence properties of this algorithm are not clear. Our approach is most closely related to the work of Berg et al. [5], and Leordeanu and Hebert [28], who formulate visual correspondence as a graph matching problem by defining an objective including terms based on appearance similarity as well as geometric compatibility between pairs of correspondences. Our model differs from those in [5] and [28] in several ways. The methods proposed in [5] and [28] handle outliers by removing low-confidence correspondences from the obtained solutions. Instead, we include in our energy an explicit occlusion cost as, for example, previously done in [43]. Thus our algorithm solves for the outliers as part of the optimization. We add to the objective a spatial coherence term, favoring spatial aggregation of matched features, which reduces the correspondence error on our examples. We also show that geometric penalty functions defined in local neighborhoods provide more accurate correspondences than geometric costs defined over entire images, such as those used in [5] and [28]. Finally, we use the method of Liu et al. [31] to learn the optimal parameter values for the model from examples, thus avoiding the need for manual parameter tuning.

**Graph matching optimization.** Graph matching is a challenging optimization problem that received considerable attention in the literature (see [13] for a comprehensive survey of methods). Proposed techniques include the graduated assignment algorithm of Gold and Rangarajan [19], spectral relaxation methods [28], [14], the COMPOSE technique of Duchi et al. [17], and the algorithm based on covering trees proposed by Yarkony et al. [48]. Maciel and Costeira [32] reduce the problem to concave minimization and apply the exact method in [10]. Torr [43] and Schellewald and Schnörr [35] use semidefinite programming (SDP) relaxation for graph matching. Among these papers, only [32] and [35] report obtaining optimal (or near optimal) solutions. The method in [32] was tested only on a single example with quadratic costs. We conjecture that on practical challenging instances this method will suffer from an exponential explosion.<sup>1</sup> As shown in [14], the SDP

1. The method in [32] first selects a linear function  $E^-$  which is an underestimator on the original objective function  $E$ , i.e.,  $E^-(\mathbf{x}) \leq E(\mathbf{x})$  for all feasible solutions  $\mathbf{x}$ . It then visits all feasible solutions  $\mathbf{x}$  with  $E^-(\mathbf{x}) \leq E(\mathbf{x}^*)$  where  $E(\mathbf{x}^*)$  is the cost of the optimal solution. For each solution a linear program is solved.

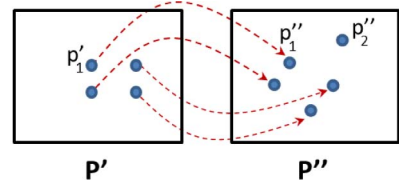


Fig. 1. An example of matching two feature point sets  $P'$  and  $P''$ , where  $|P'| = 4$  and  $|P''| = 5$ . Apart from feature  $p''_5$ , all other features have exactly one match. This means  $|A| = 20$  with 4 active and 16 inactive correspondences. For instance, it is  $x_{(p'_1, p''_1)} = 1$  and  $x_{(p'_1, p''_5)} = 0$ .

relaxation approach in [35] scales quite poorly and is too expensive for problems of reasonable size.

## 2 ENERGY FUNCTION

We now describe the energy function of our matching model. Let  $P'$  and  $P''$  be the sets of feature points extracted from the two input images. We denote with  $A \subseteq P' \times P''$  the set of potential assignments between features in the two sets. We will use the terms assignment and correspondence interchangeably to indicate elements of  $A$ . We represent a *matching configuration* between the two point sets as a binary valued vector  $\mathbf{x} \in \{0, 1\}^A$ . Each correspondence  $a \in A$  indexes an entry  $x_a$  in the vector  $\mathbf{x}$ . A correspondence  $a$  is active if  $x_a = 1$ , and it is inactive otherwise. Fig. 1 illustrates a small example. We define an energy function  $E(\mathbf{x})$  modeling our matching problem assumptions. This will allow us to formulate the matching task as minimization of  $E(\mathbf{x})$ . In this paper, we consider matching problems where at most one active correspondence per feature is allowed. This requirement is known as the uniqueness constraint and it is commonly used in correspondence problems. In order to enforce this condition we define the constraint set  $M$ :

$$M = \left\{ \mathbf{x} \in \{0, 1\}^A \mid \sum_{a \in A(p)} x_a \leq 1 \quad \forall p \in P \right\}, \quad (1)$$

where  $P = P' \cup P''$  is the set of features from both images and  $A(p)$  is the set of correspondences involving feature  $p$ . The goal is to find the configuration  $\mathbf{x} \in M$  minimizing  $E(\mathbf{x})$ . We define our energy as a weighted sum of four energy terms:

$$E(\mathbf{x}) = \lambda^{\text{app}} E^{\text{app}}(\mathbf{x}) + \lambda^{\text{occl}} E^{\text{occl}}(\mathbf{x}) + \lambda^{\text{geom}} E^{\text{geom}}(\mathbf{x}) + \lambda^{\text{coh}} E^{\text{coh}}(\mathbf{x}), \quad (2)$$

where  $\lambda^{\text{app}}$ ,  $\lambda^{\text{occl}}$ ,  $\lambda^{\text{geom}}$ , and  $\lambda^{\text{coh}}$  are scalar weights. We describe the energy terms below.

Function  $E^{\text{app}}(\mathbf{x})$  favors correspondences between features having similar appearance. We define this function as a sum of unary terms:

$$E^{\text{app}}(\mathbf{x}) = \sum_{a \in A} \theta_a^{\text{app}} x_a. \quad (3)$$

For an assignment  $a = (p', p'') \in A$ ,  $\theta_a^{\text{app}}$  is the distance between appearance descriptors (such as Shape Context [4]) computed at points  $p'$  and  $p''$  in the respective images. We have used different appearance features depending on the task at hand (see Section 5).

The term  $E^{\text{occl}}(\mathbf{x})$  imposes a penalty for unmatched features. We define  $E^{\text{occl}}(\mathbf{x})$  to be the fraction of unmatched features in the smallest of the two feature sets. We can write this function as

$$E^{\text{occl}}(\mathbf{x}) = 1 - \frac{1}{\min\{|P'|, |P''|\}} \sum_{a \in A} x_a \quad (4)$$

by noting that  $\sum_{a \in A} x_a$  is equal to the number of distinct matched features in  $P'$  and  $P''$ ,  $\forall \mathbf{x} \in M$ . This result derives trivially from the uniqueness constraint.

The term  $E^{\text{geom}}(\mathbf{x})$  is a measure of geometric compatibility between active correspondences. This term is similar to the distortion costs proposed in [5] and [28]. Note, however, that the energy terms used in these previous approaches include distortion costs for all pairs of matched features, which results in energy functions penalizing any deviation from a global rigid transformation. Instead, our function  $E^{\text{geom}}(\mathbf{x})$  measures the geometric compatibility of correspondences only for *neighboring* features. We demonstrate that this model permits more flexible mappings between the two sets of features and yields more accurate correspondences. We use a “neighborhood system”  $N$  to specify the pairs of correspondences involved in our measure of geometric compatibility.  $N$  consists of all correspondence pairs defined over neighboring features

$$N = \{ \langle (p', p''), (q', q'') \rangle \in A \times A \mid p' \in N_{q'} \vee q' \in N_{p'} \vee p'' \in N_{q''} \vee q'' \in N_{p''} \}, \quad (5)$$

where  $N_p$  indicates the set of  $K$  nearest neighbors of  $p$  (computed in the set of features  $p$ ) and  $K$  is a positive integer value controlling the size of the neighborhood, which we call *geometric neighborhood size*. In practice, we set  $K$  in the range of 2 to 6.  $E^{\text{geom}}(\mathbf{x})$  is computed over pairs of active correspondences in the set  $N$ :

$$E^{\text{geom}}(\mathbf{x}) = \sum_{(a,b) \in N} \theta_{ab}^{\text{geom}} x_a x_b, \quad (6)$$

where

$$\theta_{ab}^{\text{geom}} = \eta(e^{\delta_{a,b}^2/\sigma_l^2} - 1) + (1 - \eta)(e^{\alpha_{a,b}^2/\sigma_a^2} - 1), \quad (7)$$

$$\delta_{(p',p''),(q',q'')} = \frac{\|p' - q'\| - \|p'' - q''\|}{\|p' - q'\| + \|p'' - q''\|}, \quad (8)$$

$$\alpha_{(p',p''),(q',q'')} = \arccos\left(\frac{p' - q'}{\|p' - q'\|} \cdot \frac{p'' - q''}{\|p'' - q''\|}\right). \quad (9)$$

Intuitively,  $\theta_{(p',p''),(q',q'')}^{\text{geom}}$  computes the geometric agreement between neighboring correspondences  $(p', p'')$ ,  $(q', q'')$  by evaluating how well the segment  $\overline{p'q'}$  matches the segment  $\overline{p''q''}$  in terms of both length and direction. The parameter  $\eta$  is a scalar value trading off the importance of preserving distances versus preserving directions. Note that while the geometric term encourages the transformation relating the two images to be approximately rigid within small neighborhoods, it still allows the global mapping to be highly nonrigid. In practice, the degree of nonrigidity allowed by the energy can be tuned via two parameters: the geometric neighborhood size ( $K$ ) and the scalar value

( $\lambda^{\text{geom}}$ ) defining the relative importance of the geometric term with respect to the other energy terms.

The term  $E^{\text{coh}}(\mathbf{x})$  favors the spatial proximity of matched features. It incorporates our prior knowledge that matched features should form spatially coherent regions within each image, corresponding to common objects or parts in the image pair, in analogy to coherence on a pixel grid, used, for example, in image segmentation. We define the cost  $E^{\text{coh}}(\mathbf{x})$  as the fraction of neighboring feature pairs with different matching status (this can be viewed as an MRF Potts model over feature occlusion). We now show how to write this function directly in terms of the solution  $\mathbf{x}$ . Let  $N_P$  be the set of pairs of neighboring features in the two images:

$$N_P = \{(p, q) \in (P' \times P') \cup (P'' \times P'') \mid p \in N_q \vee q \in N_p\}. \quad (10)$$

Then, we can express  $E^{\text{coh}}(\mathbf{x})$  as a sum of unary and pairwise terms:

$$E^{\text{coh}}(\mathbf{x}) = \frac{1}{|N_P|} \sum_{(p,q) \in N_P} V_{p,q}(\mathbf{x}), \quad (11)$$

where

$$V_{p,q}(\mathbf{x}) = \sum_{a \in A(p)} x_a + \sum_{b \in A(q)} x_b - 2 \sum_{a \in A(p), b \in A(q)} x_a x_b. \quad (12)$$

$V_{p,q}(\mathbf{x})$  is equal to 0 if  $p, q$  are either both matched or both unmatched;  $V_{p,q}(\mathbf{x})$  is equal to 1 otherwise.

**Feature correspondence as graph matching.** The problem defined above can be written as

$$\min_{\mathbf{x} \in M} E(\mathbf{x} \mid \bar{\theta}) = \sum_{a \in A} \bar{\theta}_a x_a + \sum_{(a,b) \in N} \bar{\theta}_{ab} x_a x_b, \quad (13)$$

where the constraint set  $M$  is given by (1). This problem is often referred to as *graph matching* in the literature [19], [11]. Features  $P'$  and  $P''$  are viewed as vertices of the two graphs. The pairwise term  $\bar{\theta}_{ab} x_a x_b$  with  $a = (p', p'')$ ,  $b = (q', q'')$  encodes compatibility between edges  $(p', q')$ ,  $(p'', q'')$  of the first and second graph, respectively, as well as spatial coherence of matched features, while unary term  $\bar{\theta}_a x_a$  measures similarity of vertices and other effects such as occlusion. We use  $\bar{\theta}$  to denote the vector obtained by concatenating the unary and pairwise vectors, i.e.,  $\bar{\theta} = (\bar{\theta}_a, \bar{\theta}_{ab})$ .

We now address the question of how to optimize (13). Unfortunately, this problem is NP-hard [19]. We propose using the *problem decomposition* approach (or *dual decomposition*) for graph matching, which we explain next.

### 3 PROBLEM DECOMPOSITION APPROACH

On a high level, the idea is to decompose the original problem into several “easier” subproblems for which we can efficiently compute a global minimum (or obtain a good lower bound). Combining the lower bounds for individual subproblems will then provide a lower bound for the original problem. The decomposition and the corresponding lower bound will depend on a parameter vector  $\theta$ . Hence, we then try to find a vector  $\theta$  that maximizes the

bound. This approach is well known in combinatorial optimization, where it is sometimes referred to as “dual decomposition” [6]. It was applied to quadratic pseudo-Boolean functions (i.e., functions of binary variables with unary and pairwise terms) by Chardaire and Sutter [12]. Their work is perhaps the closest to the method in this paper. As in [12], we use “small” subproblems for which the global minimum can be computed exactly in reasonable time. Our choice of subproblems for graph matching, however, is different from [12]. In particular, we exploit the structure of the problem, i.e., the structure of the constraint set  $M$  and the neighborhood relations between points. In contrast, subproblems proposed in [12] were for general quadratic pseudo-Boolean functions and the method was tested on randomly generated graphs.

In computer vision the decomposition approach is probably best known in the context of the MAP-MRF inference task. It was introduced by Wainwright et al. [44], who decomposed the problem into a convex combination of trees and proposed message passing techniques for optimizing vector  $\theta$ . These techniques do not necessarily find the best lower bound (see [22] or review article [46]). Storvik and Dahl [41], Schlesinger and Giginyak [36], [37], and Komodakis et al. [25] proposed using subgradient techniques [39], [6] for MRF optimization which guarantee to converge to a vector  $\theta$  giving the best possible lower bound.

### 3.1 Graph Matching via Problem Decomposition

We now apply this approach to the graph matching problem given by (13). We decompose (13) into subproblems characterized by vectors  $\theta^\sigma$ ,  $\sigma \in I$ , with positive weights  $\rho_\sigma$ . (These weights are chosen a priori and may affect the speed of convergence of the subgradient method in Section 3.4.) Here,  $I$  is a finite set of subproblem indexes. We will require the vector  $\theta = (\theta^\sigma \mid \sigma \in I)$  to be a  $\rho$ -reparameterization of the original parameter vector  $\bar{\theta}$  (see (13)) [44], i.e.,

$$\sum_{\sigma \in I} \rho_\sigma \theta^\sigma = \bar{\theta}. \quad (14)$$

For each subproblem  $\sigma$  we will define a lower bound  $\Phi_\sigma(\theta^\sigma)$  which satisfies

$$\Phi_\sigma(\theta^\sigma) \leq \min_{\mathbf{x} \in M} E(\mathbf{x} \mid \theta^\sigma). \quad (15)$$

It is easy to see that the function

$$\Phi(\theta) = \sum_{\sigma \in I} \rho_\sigma \Phi_\sigma(\theta^\sigma) \quad (16)$$

is a lower bound on the original function. Indeed, if  $\mathbf{x}^*$  is an optimal solution to (13), then from (14)-(16) we get

$$\Phi(\theta) \leq \sum_{\sigma \in I} \rho_\sigma \min_{\mathbf{x} \in M} E(\mathbf{x} \mid \theta^\sigma) \leq \sum_{\sigma \in I} \rho_\sigma E(\mathbf{x}^* \mid \theta^\sigma) = E(\mathbf{x}^* \mid \bar{\theta}).$$

In Section 3.2, we will describe the subproblems that we use. For each subproblem  $\sigma$  we have to specify the following: 1) constraints on vector  $\theta^\sigma$ , 2) function  $\Phi_\sigma(\theta^\sigma)$ , 3) an algorithm for computing  $\Phi_\sigma(\theta^\sigma)$ . In Section 3.4, we will discuss how to maximize the lower bound  $\Phi(\theta)$  using the subproblem solutions. Finally, in Section 3.5, we will describe how to obtain solution  $\mathbf{x} \in M$  for our original problem.

### 3.2 Graph Matching Subproblems

Intuitively, a good decomposition should use subproblems that satisfy the following: 1) A subproblem should include as many variables as possible while ensuring that the computation time stays reasonable; 2) variables in a subproblem should not be independent, i.e., changing one of them should have some influence on other variables.

Motivated by these design principles, we considered three types of subproblems: a “linear” subproblem, a “maxflow” subproblem, and “local” subproblems. It will turn out that the first two subproblems are not so important in practice compared to the local subproblems, but we still decided to include them since they are rather natural and were used in the literature before (e.g., the first one was used in [17]).

**Linear subproblem.** In our first subproblem, which we denote by the index “ $L$ ,” we require all pairwise terms to be zero:  $\theta_{ab}^L = 0$  for  $(a, b) \in N$ . In such a case (13) can be solved exactly in polynomial time, for example using the Hungarian algorithm [2]. (This is often known as the *linear assignment problem*.)

**Maxflow subproblem.** In the second subproblem, which we denote by the index “ $F$ ,” we do not put any restrictions on the vector  $\theta^F$ . To get a lower bound, we ignore the uniqueness constraint  $\sum_{a \in A(p)} x_a \leq 1$  and leave only the discreteness constraint:  $x_a \in \{0, 1\}$ . If the function  $E(\mathbf{x} \mid \theta^F)$  is submodular (i.e., coefficients  $\theta_{ab}^F$  are nonpositive for all pairwise terms  $(a, b) \in N$ ), then we can compute a global minimum using a maxflow algorithm. With arbitrary  $\theta_{ab}^F$  the problem becomes NP-hard [7]. We use the *roof duality* relaxation [21], [23] to get a lower bound  $\Phi_F(\theta^F)$  on the problem. It can be defined as the optimal value of the following linear program:

$$\Phi_F(\theta^F) = \min_{\mathbf{x}} \sum_{a \in A} \theta_a^F x_a + \sum_{(a,b) \in N} \theta_{ab}^F x_{ab}, \quad (17)$$

$$\text{subject to } \begin{cases} 0 \leq x_a \leq 1 & \forall a \in A \\ x_{ab} \leq x_a, x_{ab} \leq x_b & \forall (a, b) \in N \\ x_{ab} \geq x_a + x_b - 1 & \forall (a, b) \in N \\ x_{ab} \geq 0 & \forall (a, b) \in N. \end{cases}$$

This relaxation can be solved in polynomial time by computing a maximum flow in a graph with  $2(|A| + 1)$  nodes and  $O(|A| + |N|)$  edges [8], [7].

**Local subproblems.** For our last set of subproblems we use an explicit search to compute the global minimum. We have tested two different techniques: a straightforward exhaustive search and a branch-and-bound search (see Section 3.3 for details). The latter one gives a considerable speed up, as we will document in Section 5.

For efficiency reasons, we need to make sure that subproblems are sufficiently small. We use the following technique: For each point  $p \in P$  (we remind the reader that  $P = P' \cup P''$ ), we choose  $N_p^d \subseteq P$  to be the set of  $K^d$  nearest points in the same image where  $K^d$  is a small constant, e.g., 2 or 3. (The superscript  $d$  stands for “decomposition.”) We then consider the subproblem that involves only assignments in the set  $A(N_p^d) = \{(p', p'') \in A \mid p' \in N_p^d \vee p'' \in N_p^d\}$  and the edges between those assignments. More precisely, we require vector  $\theta^p$  corresponding to this subproblem to satisfy the following constraints:

$$\begin{aligned}\theta_a^p &= 0, & \text{if } a \notin A(N_p^d), \\ \theta_{ab}^p &= 0, & \text{if } a \notin A(N_p^d) \text{ or } b \notin A(N_p^d).\end{aligned}$$

These constraints imply that we can fix assignments  $a \in A - A(N_p^d)$  to 0 when computing the minimum  $\min_{\mathbf{x} \in M} E(\mathbf{x} \mid \boldsymbol{\theta}^p)$ . Then we get a graph matching problem where the set of points in one of the images is  $N_p^d$ .

### 3.3 Solving Local Subproblems

As mentioned above, we used two strategies for computing the global minimum for local subproblems. These strategies are described below.

**Exhaustive search.** Assume that the set  $P'$  has a smaller size than  $P''$  (the other case is symmetric). First, we select an ordering of points in  $P'$ ; thus, we can assume that  $P' = \{1, 2, \dots, |P'|\}$ . We then use a depth-first search to go through all labelings  $\mathbf{x} \in M$ . We start with the zero labeling, in which all assignments are passive. At depth  $d$  ( $d = 1, \dots, |P'|$ ) we pick point  $p = d \in P'$  and explore  $|A(p)| + 1$  possible branches for  $p$ . (In each branch we either make one of assignments in  $A(p)$  active or declare all assignments in  $A(p)$  to be passive.) If we detect a violation of the uniqueness constraint in  $P''$ , then we backtrack. For each depth  $d$  we maintain the cost of the current labeling. Updating this cost at depth  $d$  takes  $O(d)$  time. (For this we need to store an  $|A| \times |A|$  matrix of costs in the memory.) The running time of this approach is  $O(|P'| \cdot (\max_{p \in P'} |A(p)| + 1)^{|P'|})$ .

**Branch-and-bound search.** Our second strategy is quite similar to the one above, but with the following difference: Consider the branch at depth  $d$  with the current labeling  $\mathbf{x} \in M$ ; thus, we made decisions for points  $\{1, \dots, d\} \subset \{1, \dots, |P'|\}$  and  $x_a = 0$  for all  $a = (p', p'')$  with  $p' \in \{d+1, \dots, |P'|\}$ . We compute a lower bound on the energy of labelings in this branch via

$$E^{\text{bound}}(\mathbf{x}, d) = E(\mathbf{x}) + \sum_{p=d+1}^{|P'|} \min_{a \in A(p)} \left[ \bar{\theta}_a^{\text{bound}} + \sum_{b \in A} \bar{\theta}_{ab} x_b \right] x_a,$$

where  $\bar{\theta}_a^{\text{bound}} = \bar{\theta}_a + \sum_{p'=d+1}^{|P'|} \min\{0, \min_{b \in A(p')} \bar{\theta}_{ab}\}$ . Values  $\bar{\theta}_a^{\text{bound}}$  are precomputed in  $O(|A|^2)$  time; then computing  $E^{\text{bound}}(\mathbf{x}, d)$  takes  $O(|\{a = (p', p'') \in A, p' > d\}| \cdot d)$  time. If  $E^{\text{bound}}(\mathbf{x}, d)$  is greater than or equal to the current best known energy, then we backtrack. To prune incorrect branches earlier, it is important to have a good initial labeling. We compute the initial labeling using two passes (forward and backward) of coordinate descent (ICM) applied to the zero labeling  $\mathbf{x} = \mathbf{0}$ . Note the lower bound described above appears to be rather crude. One potential way of improving it (not addressed in this paper) is to somehow reparameterize vector  $\boldsymbol{\theta}$  first. Also, it is not difficult to design better bounds at the expense of additional computations.

### 3.4 Lower Bound Optimization

In the previous sections we described constraints on vector  $\boldsymbol{\theta}$  and a lower bound  $\Phi(\boldsymbol{\theta})$  consisting of  $|P| + 2$  subproblems. It can be seen that  $\Phi$  is a concave function of  $\boldsymbol{\theta}$ . Furthermore, the constraints on  $\boldsymbol{\theta}$  yield a convex set that we will denote as  $\Omega$ . This set is defined by the reparameterization (14) and constraints on individual subproblems  $\boldsymbol{\theta}^\sigma \in \Omega_\sigma$  given by equalities  $\theta_a^\sigma = 0$ ,  $\theta_{ab}^\sigma = 0$

for certain assignments  $a$  and edges  $(a, b)$ . Let  $I_a, I_{ab} \subseteq I$  be the subsets of subproblem indexes for which elements  $\theta_a^\sigma$ ,  $\theta_{ab}^\sigma$ , respectively, are **not** constrained to be 0. Thus, assignment  $a \in A$  is involved in subproblems  $\sigma \in I_a$ , and edge  $(a, b) \in N$  is involved in subproblems  $\sigma \in I_{ab}$ . We have

$$\Omega = \left\{ \boldsymbol{\theta} = (\boldsymbol{\theta}^\sigma \mid \sigma \in I) \mid \begin{array}{l} \theta_a^\sigma = 0 \quad \forall a \in A, \quad \sigma \in I - I_a \\ \theta_{ab}^\sigma = 0 \quad \forall (a, b) \in N, \quad \sigma \in I - I_{ab} \\ \sum_{\sigma} \rho_\sigma \boldsymbol{\theta}^\sigma = \bar{\boldsymbol{\theta}} \end{array} \right\}.$$

Similarly to [12], [36], [37], [25], we use a projected subgradient method [39], [6] for maximizing  $\Phi(\boldsymbol{\theta})$  over  $\Omega$ . One iteration is given by

$$\boldsymbol{\theta} := \mathcal{P}_\Omega(\boldsymbol{\theta} + \lambda \mathbf{g}),$$

where  $\mathcal{P}_\Omega$  is the operator that projects a vector to  $\Omega$ ,  $\mathbf{g}$  is a subgradient of  $\Phi(\boldsymbol{\theta})$ , and  $\lambda > 0$  is a step size.

**Projection.** To project vector  $\boldsymbol{\theta}$  to  $\Omega$ , we first compute vector  $\bar{\boldsymbol{\theta}} = \sum_{\sigma} \rho_\sigma \boldsymbol{\theta}^\sigma$  and then update  $\boldsymbol{\theta}$  as follows:  $\theta_a^\sigma := 0$  for  $\sigma \in I - I_a$ ,  $\theta_{ab}^\sigma := 0$  for  $\sigma \in I - I_{ab}$ ,

$$\begin{aligned}\theta_a^\sigma &:= \theta_a^\sigma + \rho_\sigma \frac{\bar{\theta}_a - \hat{\theta}_a}{\sum_{\sigma \in I_a} \rho_\sigma^2} & \forall \sigma \in I_a, \\ \theta_{ab}^\sigma &:= \theta_{ab}^\sigma + \rho_\sigma \frac{\bar{\theta}_{ab} - \hat{\theta}_{ab}}{\sum_{\sigma \in I_{ab}} \rho_\sigma^2} & \forall \sigma \in I_{ab}.\end{aligned}$$

**Subgradient computation.** A subgradient of function  $\Phi(\boldsymbol{\theta})$  is given by

$$\mathbf{g} = \sum_{\sigma \in I} \rho_\sigma \mathbf{g}^\sigma,$$

where  $\mathbf{g}^\sigma$  is a subgradient of function  $\Phi_\sigma(\boldsymbol{\theta}^\sigma)$ . If the latter function is the global minimum of  $E(\mathbf{x} \mid \boldsymbol{\theta}^\sigma)$  (which is the case for  $\sigma \in I - \{F\}$ ), then we can take  $\mathbf{g}_a^\sigma = x_a^\sigma$ ,  $\mathbf{g}_{ab}^\sigma = x_a^\sigma x_b^\sigma$ , where  $\mathbf{x}^\sigma$  is a global minimizer of  $E(\mathbf{x} \mid \boldsymbol{\theta}^\sigma)$ . For the maxflow subproblem a subgradient can be computed as  $\mathbf{g}^F = \mathbf{x}^F$ , where  $\mathbf{x}^F$  is an optimal solution of linear program (17). For a formal proof we refer the reader to [6, Section 6.1].

To solve (17), we use the method of [8] that reduces the problem to a minimum  $s$ - $t$  cut/maximum flow problem in an appropriately constructed graph. We refer to [7], [23] for the description of this reduction; here we just mention one additional detail. The method produces a half-integer optimal solution  $x_a^F \in \{0, 0.5, 1\}$  for all assignments  $a \in A$ . Given this solution, we compute an optimal solution for variables  $x_{ab}^F$  by minimizing (17) while keeping variables  $x_a^F$  fixed. This minimization problem is easy to solve since the problem decouples. We obtain the following rule: If  $(x_a^F, x_b^F) \neq (0.5, 0.5)$ , then  $x_{ab}^F = x_a^F x_b^F$ ; otherwise  $x_{ab}^F = 0$  if  $\theta_{ab}^F \leq 0$  (i.e., the corresponding term is submodular) and  $x_{ab}^F = 0.5$  if  $\theta_{ab}^F > 0$ .

**Step size.** An important issue in the subgradient method is the choice of the step size  $\lambda$ . We used an adaptive technique mentioned in [6]. We set  $\lambda = \alpha(\Phi(\boldsymbol{\theta}^*) + \delta - \Phi(\boldsymbol{\theta})) / \|\mathbf{g}\|^2$ , where  $\alpha$  is a constant (1 in our experiments),  $\boldsymbol{\theta}^*$  is the best vector found so far (i.e., the vector giving the best lower bound), and  $\delta$  is a positive number which is updated as follows: If the last iteration improved the best lower bound  $\Phi(\boldsymbol{\theta}^*)$ , then  $\delta$  is increased by a certain factor

(1.5 in our experiments); otherwise it is decreased by a certain factor (0.95).

**Restarting the subgradient method.** In our implementation we also used the following technique borrowed from [30]. If the best value of the lower bound  $\Phi(\theta^*)$  has not changed during  $\gamma$  iterations, then we replace  $\theta$  with  $\theta^*$ . In the beginning  $\gamma = 20$  and, after every restart, it is updated as  $\gamma := \min\{\gamma + 10, 50\}$ .

### 3.5 Solution Computation

To conclude the description of the method, we need to specify how to obtain solution  $\mathbf{x} \in M$ . If the linear subproblem were to be included in the decomposition, then it is natural to use its minimum  $\mathbf{x}^L$  in each iteration since  $\mathbf{x}^L$  is guaranteed to satisfy the uniqueness constraint. However, we excluded this subproblem for the experiments presented in this paper (for reasons explained below). Hence, we computed the solution in a given iteration as follows: Starting with labeling  $\mathbf{x} = 0$ , we go through local subproblems  $\sigma \in I - \{L, F\}$  and assignments  $a$  involved in  $\sigma$  (in a fixed order), set  $x_a = 1$  if  $x_a^\sigma = 1$  and this operation preserves the uniqueness constraint on  $\mathbf{x}$ .

We maintain the solution with the smallest energy computed so far, and output it as a result of the method.

### 3.6 Properties of Decomposition

Of course, it is not necessary to use all subproblems described in Section 3.2. The only requirement is that each assignment  $a \in A$  and edge  $(a, b) \in N$  should be covered by at least one subproblem (i.e.,  $I_a$  and  $I_{ab}$  should be nonempty); otherwise the projection operation would be undefined. In this section, we study how the choice of subproblems affects the optimal value of the lower bound  $\max_{\theta \in \Omega} \Phi(\theta)$ . Without loss of generality we can assume  $\rho_\sigma = 1$  for all  $\sigma \in I$ . (Indeed, weights  $\rho_\sigma$  may affect the speed of the subgradient method, but they do not affect the value of the optimal bound since the transformation  $\rho_\sigma := \rho_\sigma / \gamma$ ,  $\theta^\sigma := \theta^\sigma \cdot \gamma$  with  $\gamma > 0$  preserves the bound.)

First, we compare the bound provided by our decomposition to the bound of the following approach. Let us rewrite the energy  $E$  (13) in such a way that the uniqueness constraint is implicitly encoded in the energy so that optimization can be performed in the full domain  $\mathbf{x} \in \{0, 1\}^A$ , without the restriction that  $\mathbf{x} \in M$ . This new energy, which we denote by  $E'$ , consists of unary and quadratic terms. Thus, we can apply the *roof duality* relaxation [21], [7], [23] to get a lower bound on function (13). We will refer to this approach (and to this bound) as *QPBO*. Formally, the new function  $E'$  is obtained by replacing uniqueness constraints  $\sum_{a \in A(p)} x_a \leq 1$  with pairwise terms  $Cx_ax_b$  for  $a, b \in A(p)$ ,  $a \neq b$ , where  $C$  is a sufficiently large constant:

$$E'(\mathbf{x}) = E(\mathbf{x}) + \sum_{p \in P' \cup P''} \sum_{a, b \in A(p), a \neq b} Cx_ax_b,$$

Clearly, the minimization problem,

$$\min_{\mathbf{x} \in \{0, 1\}^A} E'(\mathbf{x}), \quad (18)$$

is equivalent to (13).

Let us now compare the bound of our decomposition method with the QPBO bound. (Later we will also explore the performance of QPBO experimentally.)

**Lemma 1.** *If the set  $I$  includes the linear and maxflow subproblems, then the optimal value of the lower bound  $\Phi(\theta^*)$  is the same as or larger than the QPBO bound.*

A proof is given in Appendix A, which can be found in the Computer Society Digital Library at <http://doi.ieeecomputersociety.org/10.1109/TPAMI.2012.105>. In this proof, we derive the LP relaxation solved by the decomposition approach in the case when  $I = \{L, F\}$ .

The next lemma shows that the linear and maxflow subproblems are often not essential. (A proof is given in Appendix B, which is available in the online supplemental material.)

**Lemma 2.**

1. *Suppose that for each point  $p \in P$  there exists a local subproblem  $\sigma \in I - \{L, F\}$  which covers all assignments in  $A(p)$ , i.e.,  $\sigma \in I_a$  for all  $a \in A(p)$ . Then adding or removing the linear subproblem will not affect the optimal value of the lower bound of the decomposition approach.*
2. *Suppose that each assignment  $a \in A$  and each edge  $(a, b) \in N$  are covered by at least one local subproblem, i.e., there exist subproblems  $\sigma \in I - \{L, F\}$  with  $\sigma \in I_a$  and subproblems  $\sigma \in I - \{L, F\}$  with  $\sigma \in I_{ab}$ . Then adding or removing the maxflow subproblem will not affect the optimal value of the lower bound.*

It can be seen that our choice of local subproblems always satisfies conditions of part 1. Thus, the linear subproblem would not help (assuming that we can compute the optimal lower bound). Note, we described this subproblem partly because it was used in previous work: In [17] Duchi et al. computed *exact* min-marginals for the linear subproblem in the belief propagation framework.

Next, we ask whether or not conditions of part 2 are satisfied. The answer depends on the structures of the neighborhood systems  $N_p$  used for constructing the energy function and  $N_p^d$  used for constructing local subproblems. Recall that  $N_p$  and  $N_p^d$  are controlled by parameters  $K$  and  $K^d$ , respectively. If  $K \leq K^d$ , then conditions of part 2 are always satisfied; otherwise some edges may not be covered, and so including the maxflow subproblem may improve the optimal bound.

## 4 MODEL LEARNING

In most of our experiments we learned all model parameters from ground truth correspondences, as explained next. However, for some experiments there is no training data available. In that case we set all model parameters to the following default values:  $\lambda^{\text{app}} = \lambda^{\text{occl}} = \lambda^{\text{geom}} = \lambda^{\text{coh}} = 1$ ,  $\eta = 0.5$ ,  $\sigma_l^2 = 0.5$ ,  $\sigma_\alpha^2 = 0.9$ .

The energy model defined in (2) is parameterized by a set of parameters, denoted here with  $\psi = \{\lambda^{\text{app}}, \lambda^{\text{occl}}, \lambda^{\text{geom}}, \lambda^{\text{coh}}, \eta, \sigma_l^2, \sigma_\alpha^2\}$ . In addition, the energy depends on input features sets  $P'$  and  $P''$  extracted from the images. Here we highlight this dependence on parameters and input points, by writing the energy function as  $E(\mathbf{x}; P', P'', \psi)$ . We now consider the problem of learning parameters  $\psi$  from a set of  $n$  training matching examples defined by pairs of feature sets  $\{(P'_1, P''_1), \dots, (P'_n, P''_n)\}$  and “ground truth” correspondences

$\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ , typically specified by the user. We use the Nonlinear Inverse Optimization (NIO) algorithm described in [31]. In our context, the objective of this method corresponds to minimizing the gap in energy value between the user-provided ground truth correspondences and the matching configurations estimated via energy optimization. In other words, we minimize the following objective  $G(\boldsymbol{\psi})$ :

$$G(\boldsymbol{\psi}) = \sum_{i=1}^n E(\mathbf{x}_i; P'_i, P''_i, \boldsymbol{\psi}) - \min_{\mathbf{x} \in M} E(\mathbf{x}; P'_i, P''_i, \boldsymbol{\psi}).$$

Let  $\boldsymbol{\lambda} = \{\lambda^{\text{app}}, \lambda^{\text{occl}}, \lambda^{\text{geom}}, \lambda^{\text{coh}}\}$  and  $\boldsymbol{\sigma}^2 = \{\sigma_l^2, \sigma_o^2\}$ . In order to avoid degenerate solutions with  $\boldsymbol{\lambda} = \mathbf{0}$  and to obtain positive values for the term weights  $\lambda_i$  and variances  $\sigma_i^2$ , we define reparameterizations  $\lambda_i = e^{\nu_i} / \sum_j e^{\nu_j}$ ,  $\sigma_i^2 = e^{\epsilon_i}$ , and minimize  $G$  with respect to  $\{\boldsymbol{\epsilon}, \boldsymbol{\nu}\} = \boldsymbol{\psi}$ . As in [31], we optimize  $G$  via gradient descent with line search. The gradient of  $G$  is locally approximated by optimizing the energies given the current estimate of the parameters in each iteration. The learning procedure was initialized using default parameters (see above).

## 5 EXPERIMENTS

We start with a description of the different methods considered in our experiments (Section 5.1). Then, in Section 5.2, we present results on various image datasets.

### 5.1 Algorithms

In our experiments, we compare the following methods:

**DD:** This is our dual decomposition procedure. We used  $K^d = \min\{K, 4\}$ , where  $K$  is the geometric neighborhood size. Motivated by the insights in Section 3.6, we did not use the linear subproblem. We set  $\rho_\sigma = 1$  for all other subproblems  $\sigma$ . We used a maximum of 10,000 iterations and stopped earlier if the gap between the lower bound and the cost became smaller than  $10^{-6}$ .

**HUNG:** As in [11], we also tested the Hungarian algorithm using an energy consisting only of linear terms. On problems with occlusions, we used our occlusion cost in addition to the appearance term, i.e.,  $E^{\text{HUNG}}(\mathbf{x}) = \lambda^{\text{app}} E^{\text{app}}(\mathbf{x}) + \lambda^{\text{occl}} E^{\text{occl}}(\mathbf{x})$ . The Hungarian method finds the global optimum for this energy. We denote with LearnHUNG the Hungarian method applied to a learned version of the linear matching problem: We use the algorithm of Caetano et al. [11] to learn individual weights for the features of the descriptor defining the appearance term of (3).

**FUSION:** This algorithm was introduced in [27] for MRF optimization with multiple labels. We propose using it for graph matching as follows: First, we generate 256 solutions by applying one pass of coordinate descent (ICM) to zero labeling using random orders. (Different orders of visiting assignments usually yield different solutions.) We then “fuse” together pairs of solutions using the binary tree structure until a single solution remains. Fusion of solutions  $\mathbf{x}'$ ,  $\mathbf{x}''$  is defined as follows: First, we fix all assignments  $a \in A$  for which  $\mathbf{x}'$  and  $\mathbf{x}''$  agree, i.e.,  $x'_a = x''_a$ . Then we convert the obtained graph matching problem to a quadratic pseudo-Boolean optimization problem as described in Section 3.6. Finally, we run the QPBO-PI method [34] starting either with labeling  $\mathbf{x}'$  if  $E(\mathbf{x}' | \bar{\boldsymbol{\theta}}) < E(\mathbf{x}'' | \bar{\boldsymbol{\theta}})$  or

with  $\mathbf{x}''$  otherwise. The produced solution  $\mathbf{x}$  is guaranteed to have the same or smaller cost than the costs of  $\mathbf{x}'$  and  $\mathbf{x}''$ .

In Fig. 4b, we show a plot of the energy as a function of time. Clearly, this plot depends on the order of fusions. We used the following order: We always pick the leftmost node of the binary tree whose parents are available for fusion. Thus, the plots are independent of the number of initial solutions (256 in our case).

**BP:** We converted graph matching to a quadratic pseudo-Boolean optimization problem and ran the max-product belief propagation algorithm. We also tested applying the roof duality approach instead of BP, but the results were quite discouraging (see Section 5.2).

**CTB:** We tested the quadratic assignment solver proposed by Yarkony et al. [48], using software made available to us by the authors. This method operates in two stages: First, it uses tree-reweighted belief propagation to minimize a lower bound on the original objective defined by a covering tree of the matching graph; then, in a second stage, one-to-one correspondences are established by applying bipartite linear matching to the min-marginal costs computed with the covering tree.

**SMAC:** We ran the spectral relaxation method of Cour et al. [14] using the graduated assignment algorithm [19] for discretization. Since SMAC imposes affine constraints on the solution, we applied this algorithm only to datasets without outliers, where the one-to-one affine constraint is satisfied. In principle, SMAC could handle outliers by the introduction of dummy nodes. However, this would increase the number of variables and potentially make the problem harder to solve.

**COMPOSE:** We reimplemented the “COMPOSE” algorithm described in [17]. The problem was cast as assigning a label from the set  $A(p) \cup \{\text{“occlusion”}\}$  to each point  $p \in P'$ . Min-marginals for the linear subnetwork were computed via  $O(|A| + |P'|)$  calls to the Dijkstra algorithm. As in [17], we used Residual Belief Propagation (RBP) [18] with damping = 0.3 for computing pseudo min-marginals for the “smoothness” subnetwork containing pairwise terms  $\theta_{ab} x_a x_b$ . However, in our experiments messages did not converge, so we set an additional termination criterion for RBP: We stop it after passing  $20|N|$  messages. As in [17], we computed the configuration by looking at individual messages at each node. We did not use damping for the outer loop since otherwise the produced configurations usually did not satisfy the uniqueness constraint. We also informally tested the COMPOSE method with our representation, which labels each assignment as 0 or 1. To compute min-marginals for the smoothness subnetwork we tried both a maxflow algorithm (in the case of submodular potentials) and RBP. However, it did not seem to improve the results, and the issue with convergence remained.

**IPFP:** This is the Integer Projected Fixed Point method of Leordeanu et al. [29]. It is an iterative procedure involving repeated alternation of two steps: The first step improves the current solution in the discrete domain via a linearization of the objective; in the second step the direction obtained from the first step is used to maximize the original quadratic energy in the continuous domain. As in [29], we test IPFP by itself (using a flat uniform continuous solution



as initialization), but also by applying it to initial solutions computed via Spectral Matching [28] (SM+IPFP), and the Hungarian method (HUNG+IPFP). We do not report the results obtained using only Spectral Matching since, as already observed in [29], we found them to be widely inferior to those produced by SM+IPFP.

## 5.2 Comparative Results

**Hotel sequence: wide baseline matching.** In this section, we report results on the CMU “hotel” sequence.<sup>2</sup> As in previous work [11], we use this dataset to assess the performance of graph matching methods and ignore the rigid motion constraint that could be exploited using alternative wide-baseline matching algorithms [42]. We reproduce the experimental setup described in [11] using the same manual labeling of 30 landmark points and the same subset of 105 frame pairs. As in this previous work, we adopt as unary terms the distances between Shape Context descriptors. However, we replace the pairwise terms proposed in [11] with our geometric energy function  $E^{\text{geom}}(\mathbf{x})$ , using  $K = 2$ . Due to the absence of outliers, we remove  $E^{\text{coh}}(\mathbf{x})$  from our energy and use a large constant value for  $\lambda^{\text{occl}}$ . We set the remaining parameters to default values, as defined in Section 4. We set  $A = P' \times P''$ . Fig. 2a shows the matching error obtained by optimizing this model with different methods. We also include in the plot the performance of the Hungarian method (HUNG and LearnHUNG). For LearnHUNG, we learned the feature weights using as training data the Hotel frames not included in the 105 test pairs (for a total of 3,655 training pairs). Note in the figure the large variations in matching accuracy obtained with the different approaches: BP and DD are by far the best methods with errors approaching 0 percent. Note that the error obtained with our model and our optimization is over 50 times smaller than the errors reported in [11]. In particular, our approach greatly outperforms LearnHUNG, i.e., the Hungarian method applied to a learned appearance model. This clearly underscores the benefit of using the spatial arrangement of the points as additional information for the matching. As shown in Fig. 2b, for all 105 image pairs, DD converged to the global minimum of our objective (convergence to the global minimum is detected when the lower bound and the energy have the same value). In 2 out of 105 test cases, BP yields a suboptimal solution that nevertheless matches the ground truth. The average runtime of our method on an Intel Core Duo CPU @ 3 GHz is 12.9 seconds per image pair when using exhaustive search to solve the local subproblems. The average runtime is reduced to 1.4 seconds when using the branch and bound strategy to compute the global minimum of the local subproblems. As a reference, SMAC and IPFP are by far the fastest quadratic assignment methods considered in our comparison, with average running times of 0.04 and 0.2 seconds, respectively. We found that on these matching problems, applying QPBO on the energy  $E'$  (18), does not provide any labeling at all.

**House dataset: model and optimization comparison with CTB.** In [48], Yarkony et al. provide an in-depth comparison between CTB and DD applied to wide baseline matching of images from the CMU “House” dataset. Their

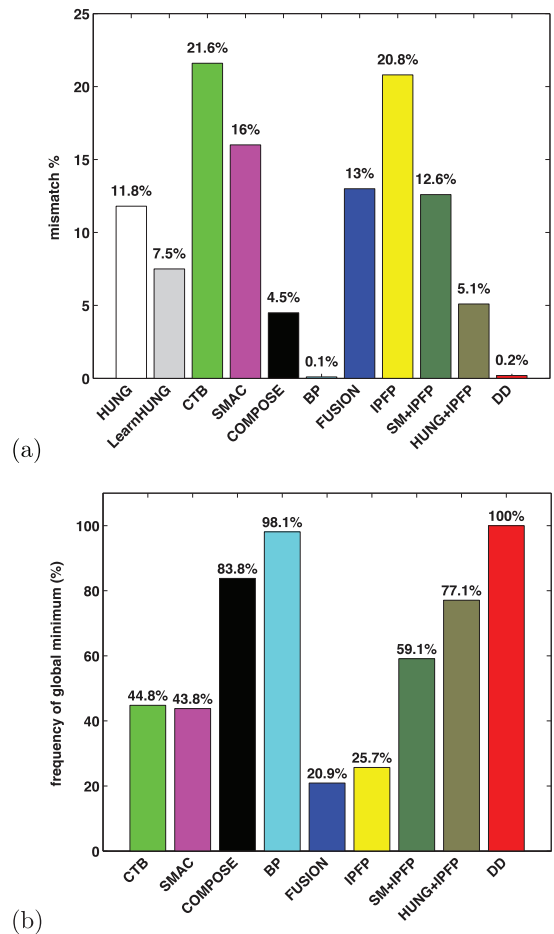


Fig. 2. Results on the Hotel sequence ( $|P'| = |P''| = 30$ ,  $|A| = 900$ ). (a) Mismatch percentages for HUNG and different optimization methods applied to our energy model. (b) Frequency of convergence to global minimum.

analysis suggests that when the two algorithms are applied to their proposed quadratic energy model (for details about the energy function see [48]), CTB provides a significant speedup and a slightly better matching accuracy (85.6 percent for CTB versus 81.8 percent for DD). Using the software kindly provided by the authors, we were able to reproduce these results. However, using our branch and bound approach (instead of exhaustive search) to solve the local subproblems significantly reduces the computational cost gap between the two methods (average runtime is 14.6 seconds for CTB and 59.3 seconds for DD using branch and bound). Furthermore, we found that the two methods are fairly similar in terms of energy values, with DD being slightly superior: 49 out of 105 times DD returns lower energy values than CTB and 20 out of 105 times the two methods output the same value.

We also tried applying the two optimization methods to our energy model. We used the same features and settings as for the House dataset experiment. The results are summarized in Table 1 (top rows): When using our energy model, DD is far superior to CTB according to all performance measures (correspondence accuracy, frequency of convergence to global minimum, as well as runtime). One of the key differences between the energy model of [48] and ours is that in the former the pairwise interaction terms are defined only between correspondences involving neighboring features in

2. <http://vasc.ri.cmu.edu/idb/html/motion/hotel/index.html>.



TABLE 1

House Data Set Experiment: Performance of DD and CTB Applied to Our Energy Model (Top Two Rows) and the Model of [48] (Bottom Two Rows)

		matching accuracy (%)	global minimum freq. (%)	lower energy (count)	average runtime (sec)
our model	DD	100	100	99	4.8
	CTB	65.7	5.7	0	41.5
model [48]	DD	82.5	57.1	49	59.3
	CTB	84.9	n/a	36	14.6

The column “lower energy” lists the number of instances (in total 105) where the respective method was better than the other. Note, for some instances both methods have the same energy. The table clearly shows that DD applied to our model gives the best results.

the reference (left) image whereas in our model we include pairwise terms when two points are neighbors in either image (see (5)). As a consequence, the number of pairwise terms in the energy function of [48] is much smaller than in our model and the corresponding graphical model over  $\mathbf{x}$  becomes sparse. This enables faster optimization by the tree-reweighted belief propagation employed by CTB. Conversely, for this dataset our higher connectivity model is more successfully optimized by DD. Overall, the combination of our energy model and our optimization method provides by far the best results in terms of both speed and accuracy. As a reference, on this sequence HUNG yields an accuracy of 44.6 percent. LearnHUNG achieves an accuracy of 68.2 percent when using the remaining House frames to learn the linear matching objective with the algorithm of Caetano et al. Again, this contrasts with the results of [11], where it was shown that linear matching applied to learned appearance models performed better than state-of-the-art quadratic assignment methods. Our powerful optimization method allows our approach to effectively exploit the spatial information in the quadratic objective and to greatly outperform learned linear matching models.

**Matching MNIST digits.** Here we describe experiments on images of handwritten digits from the MNIST dataset [26]. For training, we randomly sampled from this dataset one image pair for each of the 10 digits. We repeated the same procedure to generate a test set of 10 pairs of the same digits. From each pair we extracted point sets  $P'$  and  $P''$  by uniformly sampling 100 points along the Canny edges of each image, using the procedure described in [4]. We defined the unary potentials  $\theta_{(p',p'')}^{\text{app}}$  to be the euclidean distances between Shape Context descriptors computed at points in  $P'$  and  $P''$ . We formed the set of candidate assignments  $A \in P' \times P''$  by selecting the five most similar features, in terms of Shape Context distance, for each point  $p \in P$ . We collected ground truth correspondences in the set  $(P' \times P'')$  for each of the 20 image pairs. The parameters of our model were learned from the 10 training image pairs with NIO (see Section 4). Fig. 3a shows that the matching accuracy on the test set critically depends on the ability to globally optimize the energies during the model learning stage. The left plot reports the frequency of convergence to a global minimum during learning, plotted as a function of  $K$  (the number of nearest neighbors used for the computation of the geometric penalty terms). The second plot shows the test set matching error of DD with learned versus default

parameters. Matching error here is measured as percentage of incorrect correspondences with respect to ground truth.<sup>3</sup> We can see that the matching is much more accurate when using the parameters for which DD more frequently reached global optimality during learning. Interestingly, although the frequency of global minimum convergence increases slightly when varying  $K$  from 2 to 4, the matching error remains roughly the same. This suggests that geometric penalty terms defined over small neighborhoods are sufficient to spatially regularize the correspondences. Thus, models involving geometric costs defined over all pairs of matched features, such as those used in [5], [28], may be unnecessarily restrictive for many applications, in addition to being more difficult to optimize.

Based on these results, we have used the model with  $K = 3$  for the subsequent MNIST experiments described below. Fig. 4a shows the normalized energy values obtained by different optimization methods on the 10 test image pairs, using the learned parameters. For each family of results we performed an *additive* normalization so that for each image pair the energy of the best method becomes value 1. On 9 out of the 10 test image pairs, DD reaches global optimality and provides the minimum energy value on all examples. FUSION, BP, and COMPOSE find the global minimum only on two images. FUSION finds solutions with energy values very close to those obtained by DD. COMPOSE and BP provide considerably higher energy values on some of the examples.

We have also attempted to minimize the energy by running the QPBO algorithm on the equivalent quadratic pseudo-Boolean optimization problem, i.e., energy  $E'$  (18). This algorithm produces partial labelings that are part of a global optimum. However, we found that on our MNIST matching problems, QPBO labeled, on average, only 16 percent of the correspondences, with only 0.12 percent of these assignments corresponding to active correspondences. We also tried to apply the PROBE method [9], [34] to get more labeled nodes. However, in practice we were unable to do so due to the high computational cost of running PROBE on our problem instances.

Fig. 4b shows minimization performance as a function of time, evaluated on a sample image pair. Fig. 4c shows how optimization performance translates into correspondence accuracy. Again, we find that DD and FUSION are the best methods. The parameters used for the energy of HUNG were learned from the training examples using NIO with Hungarian matching for optimization. On these instances the simple appearance-based model used by HUNG gives poor accuracy. We also evaluated variations of the energy model defined in (2), obtained by dropping the spatial coherence term and by forcing all points to be matched (implemented by fixing  $\lambda^{\text{occl}}$  to a large value). The parameters of these modified models were learned again with NIO, using DD for optimization during both training and testing. We see from Fig. 3b that both the spatial coherence prior and the occlusion term, make the matching

3. In order to account for a certain degree of inherent ambiguity in the selection of ground truth correspondences on these images, we did not consider it an error if a point was assigned to any of the 3-nearest neighbors of the correct feature. Declaring a point with a ground truth correspondence an outlier (or vice versa) was counted as an error.

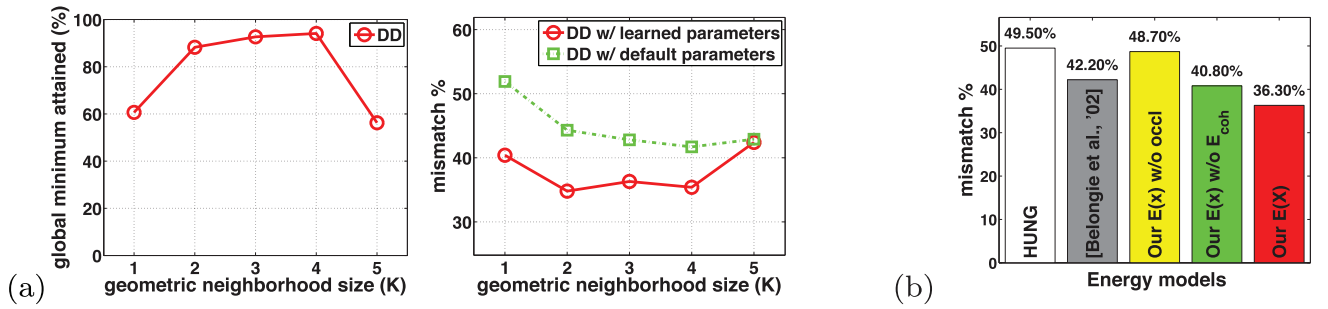


Fig. 3. Evaluation on MNIST digits ( $|P'| = |P''| = 100$ ,  $|A| = 695$  on average). (a) Correlation between learning accuracy and matching performance: The left plot shows the frequency of global minimum convergence during learning versus  $K$ ; the right plot shows mismatch error on test set. (b) Mismatch error with different energy models.

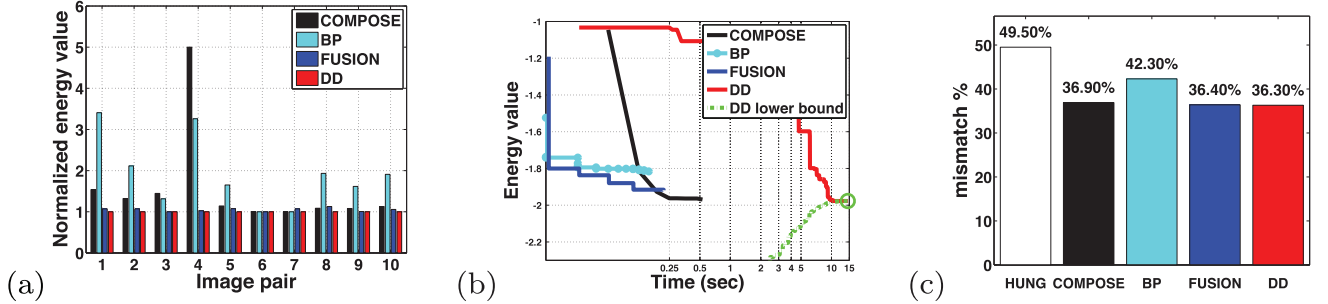


Fig. 4. Experiments on the MNIST dataset. (a) Normalized energy values. (b) Optimization performance versus runtime. (c) Mismatch error comparison between HUNG and different optimization methods using our energy.

considerably more accurate. We also report the matching error given by the model and optimization method of Belongie et al., which was applied in [4] to MNIST digit examples. For this experiment, we use the source code provided by the authors and the settings described in [4]. Our approach performs better than this state-of-the-art method. Fig. 5 illustrates some of the matches obtained with DD and HUNG. Our model yields more accurate and geometrically consistent correspondences.

**Estimating long range nonrigid motion.** We now describe results on the task of estimating large-disparity motion. For this experiment we used four (time-separated) video frames of a child jumping. We matched each image to every other image, for a total of six matches. The motion between any pair of these pictures is very large and highly nonrigid. There is self-occlusion created by the motion of arms and torso, and occlusion due to a tricycle positioned between the child and the camera. Feature points were extracted by running the Harris corner detector on each image. We used euclidean distances of geometric blur descriptors [5] computed at each feature point, both for selecting assignments in  $A$  (by choosing the five most similar features for each point  $p \in P$ ) as well as for calculating the unary terms of our energy. We learned the parameters in our model by applying the NIO algorithm to ground truth correspondences of two image pairs from a separate sequence containing the same child walking. Here we report results using  $K = 6$ . Fig. 6 shows two matching examples from this experiment and correspondences found with HUNG and DD. Note the ability of our system to cope well with occlusion and the presence of multiple motions. DD converged to a global minimum on all the image pairs in this experiment (see Figs. 7a, and 7b). Fig. 7c reports the percentage of correspondence errors (including mismatches as well as missed assignments).

**Matching faces.** Our final experiment is carried out on a set of eight face images of distinct individuals with different facial expressions. We used two of these images for learning the model parameters given their manually labeled correspondences. We then exhaustively matched the remaining 6 images, for a total of 15 test image pairs. Point sets  $P'$  and  $P''$  and candidate assignments  $A$  for each image pair were formed by matching geometric blur descriptors [5] computed along Canny edges in each image, using an iterative procedure. Starting from empty sets  $P' = P'' = A = \{\}$ , we alternate selection of a new point  $p$  from either the left or the right image by choosing the edge point (among those not yet considered) having minimum geometric blur distance to

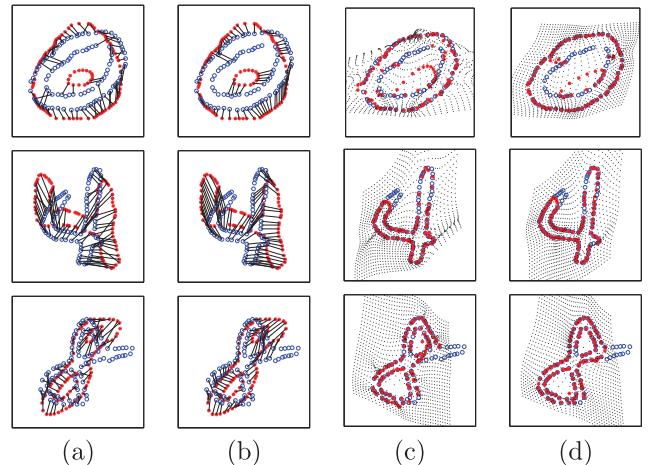


Fig. 5. Correspondences obtained with (a) Hungarian matching and (b) DD. The last two columns show grids warped by the thin-plate spline computed from the correspondences of (c) Hungarian matching and (d) DD.

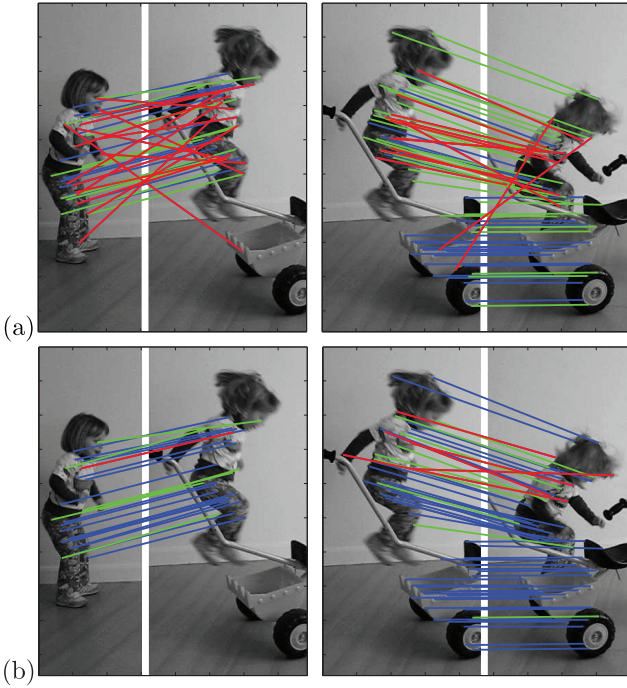


Fig. 6. Estimating human motion ( $|P'| = 118$ ,  $|P''| = 172$ ,  $|A| = 1,128$  on average). Correspondences computed with (a) the Hungarian method and (b) DD. Correct correspondences are shown in blue, missed assignments in green, and incorrect correspondences in red.

points in the other image. We add the three best assignments involving point  $p$  to  $A$  and the corresponding points to  $P'$  and  $P''$ . We then introduce an inhibition window around point  $p$  so that no other points in that neighborhood will be selected. We repeat this procedure 600 times. On average, this yields point sets with more than 900 points in each image, and a

set  $A$  with over 1,700 potential assignments. Here, we used geometric neighborhood size  $K = 6$ , and again defined the unary term to be the euclidean distance between geometric blur descriptors. Fig. 8a shows the normalized energy values obtained with the optimization methods in our comparison. FUSION is the best performing method 93.33 percent of the times, while DD is the best on the remaining 6.67 percent cases. On all image pairs DD and FUSION obtain very similar values, but on none of these challenging matches are they able to reach the lower bound. By dropping the value of  $K$  to 4, we found that DD can reach the global minimum in most of the cases, although the correspondences are slightly less accurate than when using  $K = 6$ . Here BP performs rather poorly. Fig. 8b illustrates the correspondences found by HUNG and DD on one of these image pairs.

## 6 CONCLUSIONS

We have introduced novel models and optimization algorithms for feature correspondence. We believe we are the first to demonstrate graph matching techniques capable of reaching global optimality on various real-world image matching problems.

There are several directions for future work that could potentially improve our method. The subgradient ascent scheme that we employ is known to have a slow convergence rate (sublinear in the worst case); one could test alternative techniques from the convex optimization literature. Also, the branch-and-bound computation for local subproblems could potentially be sped up by exploiting a better bound (the bound described in Section 3.3 appears to be rather weak).

A software implementation of our method is available online [1].

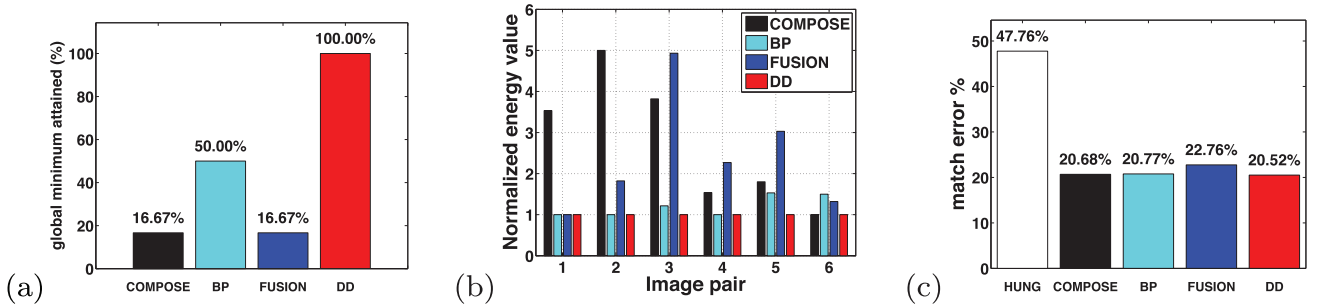


Fig. 7. Experiments on human motion frames. (a) Frequency of convergence to global minimum. (b) Normalized energy values. (c) Correspondence error.

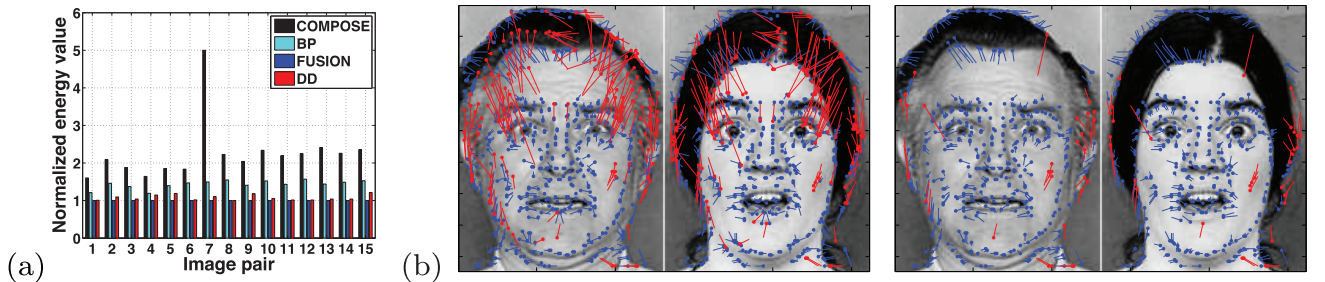


Fig. 8. Results on face images ( $|P'| = 922$ ,  $|P''| = 915$ ,  $|A| = 1782$ , on average). (a) Normalized energy values. (b) Correspondences found with Hungarian matching (left) and DD (right). Assignments are shown as feature displacements in each image. Red lines denote incorrect assignments and blue lines correct matches.



## ACKNOWLEDGMENTS

The authors are grateful to Timothee Cour, Praveen Srinivasan, and Jianbo Shi for providing the software of their SMAC algorithm. They thank John Duchi, Gal Elidan, and Danny Tarlow for sharing code and answering questions about the COMPOSE method. They thank Julian Yarkony and Charless Fowlkes for making the code implementing their CTB algorithm available to them. Thanks to Tiberio Caetano for providing the software implementing the "learning graph matching" method as well as the Hotel and House feature data. Finally, they are grateful to Marius Leordeanu for making the code for his IPFP algorithm publicly available. This research was funded in part by Microsoft Research.

## REFERENCES

- [1] <http://pub.ist.ac.at/~vnk/software/GraphMatching-v1.0.src.tar.gz>, 2011.
- [2] R. Ahuja, T. Magnanti, and J. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.
- [3] P.N. Belhumeur, "A Binocular Stereo Algorithm for Reconstructing Sloping, Creased, and Broken Surfaces in the Presence of Half-Occclusion," *Proc. Fourth IEEE Int'l Conf. Computer Vision*, May 1993.
- [4] S. Belongie, J. Malik, and J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509-522, Apr. 2002.
- [5] A. Berg, T. Berg, and J. Malik, "Shape Matching and Object Recognition Using Low Distortion Correspondence," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2005.
- [6] D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.
- [7] E. Boros and P. Hammer, "Pseudo-Boolean Optimization," *Discrete Applied Math.*, vol. 123, nos. 1-3, pp. 155-225, 2002.
- [8] E. Boros, P.L. Hammer, and X. Sun, "Network Flows and Minimization of Quadratic Pseudo-Boolean Functions," Technical Report RRR 17-1991, RUTCOR, May 1991.
- [9] E. Boros, P.L. Hammer, and G. Tavares, "Preprocessing of Unconstrained Quadratic Binary Optimization," Technical Report RRR 10-2006, RUTCOR, Apr. 2006.
- [10] A. Cabot and R. Francis, "Solving Certain Nonconvex Quadratic Minimization Problems by Ranking the Extreme Points," *Operations Research*, vol. 18, no. 1, pp. 82-86, 1970.
- [11] T.S. Caetano, L. Cheng, Q.V. Le, and A.J. Smola, "Learning Graph Matching," *Proc. 11th IEEE Int'l Conf. Computer Vision*, 2007.
- [12] P. Chardaire and A. Sutter, "A Decomposition Method for Quadratic Zero-One Programming," *Management Science*, vol. 41, no. 4, pp. 704-712, 1995.
- [13] D. Conte, P. Foggia, C. Sansone, and M. Vento, "Thirty Years of Graph Matching in Pattern Recognition," *Int'l J. Pattern Recognition and Artificial Intelligence*, vol. 18, no. 3, pp. 265-298, 2004.
- [14] T. Cour, P. Srinivasan, and J. Shi, "Balanced Graph Matching," *Proc. Neural Information Processing Systems*, 2007.
- [15] G. Csurka, C. Bray, C. Dance, and L. Fan, "Visual Categorization with Bags of Keypoints," *Proc. Workshop Statistical Learning in Computer Vision*, 2004.
- [16] G. Dorko and C. Schmid, "Selection of Scale-Invariant Parts for Object Class Recognition," *Proc. Ninth IEEE Int'l Conf. Computer Vision*, 2003.
- [17] J. Duchi, D. Tarlow, G. Elidan, and D. Koller, "Using Combinatorial Optimization within Max-Product Belief Propagation," *Proc. Neural Information Processing Systems*, 2007.
- [18] G. Elidan, I. McGraw, and D. Koller, "Residual Belief Propagation: Informed Scheduling for Asynchronous Message Passing," *Proc. 22nd Conf. Uncertainty in AI*, 2006.
- [19] S. Gold and A. Rangarajan, "A Graduated Assignment Algorithm for Graph Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 377-388, Apr. 1996.
- [20] K. Grauman and T. Darrell, "The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features," *Proc. 10th IEEE Int'l Conf. Computer Vision*, 2005.
- [21] P.L. Hammer, P. Hansen, and B. Simeone, "Roof Duality, Complementarity and Persistency in Quadratic 0-1 Optimization," *Math. Programming*, vol. 28, pp. 121-155, 1984.
- [22] V. Kolmogorov, "Convergent Tree-Reweighted Message Passing for Energy Minimization," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1568-1583, Oct. 2006.
- [23] V. Kolmogorov and C. Rother, "Minimizing Non-Submodular Functions with Graph Cuts—A Review," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 7, pp. 1274-1279, July 2007.
- [24] V. Kolmogorov and R. Zabih, "Computing Visual Correspondence with Occlusions Using Graph Cuts," *Proc. Eighth IEEE Int'l Conf. Computer Vision*, 2001.
- [25] N. Komodakis, N. Paragios, and G. Tziritas, "MRF Optimization via Dual Decomposition: Message-Passing Revisited," *Proc. 11th IEEE Int'l Conf. Computer Vision*, 2007.
- [26] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.
- [27] V. Lempitsky, C. Rother, and A. Blake, "LogCut—Efficient Graph Cut Optimization for Markov Random Fields," *Proc. 11th IEEE Int'l Conf. Computer Vision*, 2007.
- [28] M. Leordeanu and M. Hebert, "A Spectral Technique for Correspondence Problems Using Pairwise Constraints," *Proc. 10th IEEE Int'l Conf. Computer Vision*, 2005.
- [29] M. Leordeanu, M. Hebert, and R. Sukthankar, "An Integer Projected Fixed Point Method for Graph Matching and Map Inference," *Advances in Neural Information Processing Systems 22*, pp. 1114-1122, 2009.
- [30] C. Lim and H.D. Sherali, "Convergence and Computational Analyses for Some Variable Target Value and Subgradient Deflection Methods," *Computational Optimization and Applications*, vol. 34, no. 3, pp. 409-428, 2006.
- [31] C.K. Liu, A. Hertzmann, and Z. Popović, "Learning Physics-Based Motion Style with Nonlinear Inverse Optimization," *ACM Trans. Graphics*, vol. 24, no. 3, pp. 1071-1081, 2005.
- [32] J. Maciel and J. Costeira, "A Global Solution to Sparse Correspondence Problems," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 187-199, Feb. 2002.
- [33] D. Metaxas, E. Koh, and N. Badler, "Multi-Level Shape Representation Using Global Deformations and Locally Adaptive Finite Elements," *Int'l J. Computer Vision*, vol. 25, no. 1, pp. 49-61, 1997.
- [34] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer, "Optimizing Binary MRFs via Extended Roof Duality," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2007.
- [35] C. Schellewald and C. Schnörr, "Probabilistic Subgraph Matching Based on Convex Relaxation," *Proc. Fifth Int'l Conf. Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2005.
- [36] M.I. Schlesinger and V.V. Giginyak, "Solution to Structural Recognition (MAX,+)-Problems by Their Equivalent Transformations. Part 1," *Control Systems and Computers*, vol. 1, pp. 3-15, 2007.
- [37] M.I. Schlesinger and V.V. Giginyak, "Solution to Structural Recognition (MAX,+)-Problems by Their Equivalent Transformations. Part 2," *Control Systems and Computers*, vol. 2, pp. 3-18, 2007.
- [38] S. Sclaroff and A. Pentland, "Modal Matching for Correspondence and Recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, no. 6, pp. 545-561, June 1995.
- [39] N.Z. Shor, *Minimization Methods for Nondifferentiable Functions*. Springer-Verlag, 1985.
- [40] J. Sivic, B. Russell, A. Efros, and A. Zisserman, "Discovering Object Categories in Image Collections," *Proc. IEEE Int'l Conf. Computer Vision*, 2005.
- [41] G. Storvik and G. Dahl, "Lagrangian-Based Methods for Finding MAP," *IEEE Trans. Image Processing*, vol. 9, no. 3, pp. 469-479, Mar. 2000.
- [42] P.H.S. Torr, "Geometric Motion Segmentation and Model Selection," *Philosophical Trans. Royal Soc.*, vol. 356, pp. 1321-1340, 1998.
- [43] P.H.S. Torr, "Solving Markov Random Fields Using Semi Definite Programming," *Proc. Artificial Intelligence and Simulation Conf.*, 2003.
- [44] M. Wainwright, T. Jaakkola, and A. Willsky, "MAP Estimation via Agreement on Trees: Message-Passing and Linear-Programming Approaches," *IEEE Trans. Information Theory*, vol. 51, no. 11, pp. 3697-3717, Nov. 2005.

- [45] C. Wallraven, B. Caputo, and A. Graf, "Recognition with Local Features: The Kernel Recipe," *Proc. Ninth IEEE Int'l Conf. Computer Vision*, 2003.
- [46] T. Werner, "A Linear Programming Approach to Max-Sum Problem: A Review," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 7, pp. 1165-1179, July 2007.
- [47] J. Willamowski, D. Arregui, G. Csurka, C. Dance, and L. Fan, "Categorizing Nine Visual Classes Using Local Appearance Descriptors," *Proc. ICPR Workshop Learning for Adaptable Visual Systems*, 2004.
- [48] J. Yarkony, C.C. Fowlkes, and A.T. Ihler, "Covering Trees and Lower-Bounds on Quadratic Assignment," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010.
- [49] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local Features and Kernels for Classification of Texture and Object Categories: An In-Depth Study," Technical Report RR-5737, INRIA Rhone-Alpes, Nov. 2005.



**Lorenzo Torresani** received the Laurea degree in computer science with *summa cum laude* honors from the University of Milan, Italy, in 1996, and the MS and PhD degrees in computer science from Stanford University in 2001 and 2005, respectively. He is an assistant professor in the Computer Science Department at Dartmouth College. In the past, he has worked at several industrial research labs, including Microsoft Research Cambridge, Like.com, and Digital Persona. His research interests include computer vision, machine learning, and computer animation. In 2001, Torresani and his coauthors received the Best Student Paper Award at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). He is the recipient of a US National Science Foundation CAREER Award. He is a member of the IEEE.



**Vladimir Kolmogorov** received the PhD degree in computer science from Cornell University in 2003. After spending two years at Microsoft Research, Cambridge, as an associate researcher, he joined University College London (UCL) as a lecturer. In September 2011, he moved to the Institute of Science and Technology (IST), Austria, as an assistant professor. His main research interests include discrete optimization algorithms for problems in computer vision and other fields. His papers received several awards: a best paper award at ECCV 2002, a best paper honorable mention award at CVPR 2005, and an honorable mention, outstanding student paper award (to M. Pawan Kumar) at NIPS 2007. He held a Royal Academy of Engineering/EP SRC Research Fellowship in 2006-2011.



**Carsten Rother** received the diploma degree with distinction in 1999 from the University of Karlsruhe, Germany, and the PhD degree from the Royal Institute of Technology Stockholm, Sweden, supervised by Stefan Carlsson and Jan-Olof Eklundh. Since 2003, he has been a researcher at Microsoft Research Cambridge, United Kingdom. His research interests include the field of "physics-based scene recovery and understanding," in particular segmentation, matting, stereo, and object recognition, as well as in the areas of "vision for graphics," and "Markov random fields for vision." He has published more than 65 articles (h-index 29) in international conferences and journals. He won best paper honorable mention awards at ACCV '10, CHI '07, CVPR '05, and the best paper award at the Indian Conference on Computer Vision '10. He was awarded the DAGM Olympus prize in 2009. He has influenced various Microsoft products, in particular GrabCut for Office 2010 and AutoCollage. He serves on the program committees of major conferences (e.g., SIGGRAPH, ICCV, ECCV, CVPR, and NIPS), and has been an area chair for ICCV '11, ECCV '12, BMVC '08-'12, and DAGM '10-'12. He is a member of the IEEE.

► For more information on this or any other computing topic, please visit our Digital Library at [www.computer.org/publications/dlib](http://www.computer.org/publications/dlib).