International Journal of Computer Vision manuscript No.
(will be inserted by the editor)

A Study of Parts-Based Object Class Detection Using Complete Graphs

Martin Bergtholdt, Jörg Kappes, Stefan Schmidt, Christoph Schnörr
University of Heidelberg
Dept. Mathematics and Computer Science

Received: date / Accepted: date

Abstract Object detection is one of the key components in modern computer vision systems. While the detection of a specific rigid object under changing viewpoints was considered hard just a few years ago, current research strives to detect and recognize *classes* of *non-rigid*, *articulated* objects. Hampered by the omnipresent confusing information due to clutter and occlusion, the focus has shifted from holistic approaches for object detection to representations of individual object parts linked by structural information, along with richer contextual descriptions of object configurations. Along this line of research, we present a practicable and expandable probabilistic framework for parts-based object class representation, enabling the detection of rigid and articulated object classes in arbitrary views. We investigate learning of this representation from labelled training images and infer globally optimal solutions to the contextual MAP-detection problem, using A^*-search with a novel lower-bound as admissible heuristic. An assessment of the inference performance of Belief-Propagation and Tree-Reweighted Belief Propagation is obtained as a by-product. The generality of our approach is demonstrated on four different datasets utilizing domain dependent information cues.

Keywords: object detection, object class recognition, graphical models, conditional random fields, classification, multi-class, learning, inference, optimization, single view, multiple view, 2D/3D pose

Speyerer Strasse 4-6
69115 Heidelberg
Germany
E-mail: bergtholdt@web.de
E-mail: kappes@math.uni-heidelberg.de
E-mail: schmidt@math.uni-heidelberg.de
E-mail: schnoerr@math.uni-heidelberg.de

1 Introduction

1.1 Motivation and Overview

Probabilistic approaches to object detection and recognition have become a focal point of computer vision research during the last years [50]. This trend has been spurred by a substantial amount of work on the extraction of locally invariant image features [44, 75], by the impact of contextual probabilistic modeling and machine learning [27], and through the feasibility of large-scale optimization and learning on standard PCs [5].

In this paper, we adopt the representation of object views by configurations of its parts and study the detection of object categories by combining state-of-the-art approaches from three major lines of research: Local feature extraction and statistical detection [44, 42], contextual modeling with both discriminative and generative random fields [71, 40, 39], and efficient inference with deterministic algorithms [67, 36]. The graphs underlying our probabilistic representation of object classes are complete, in order to take into account all potentially relevant relations between object parts, and to better cope with deficiencies of local part detectors through contextual inference.

Our objective is to assess the capability of this general approach for modeling and learning the variability of object classes, and for detecting corresponding objects in images. To this end, we consider three different and increasingly challenging categories: faces, human spines in 3D medical image data, and humans (Figure 1), and apply throughout the *same* strategy: Discriminative modeling of local appearance of object parts and generative modeling of the geometry of part configurations are combined in a probabilistic graphical model in order to complement one another. Detection is carried out for comparison both by standard Belief-Propagation (BP) and the related convex relaxation solved by Tree-Reweighted Belief Propagation (TRBP). In order to thoroughly assess the performance of these established methods from the optimization point-of-view[1], we compute the corresponding ground truth in terms of the global optimum using A^* search with a novel lower bound as admissible heuristic. By this, we also avoid mixing up imperfections of the model and learning on the one hand, and of inference on the other hand.

Because the variability of the three object categories differs considerably, our study reveals the strengths and limitations of the overall approach. While this study shows that it competitively copes with significant variation of object appearance in a purely 2D view-based manner, it still does not generalize to the reliable detection of highly-articulating humans that are dissimilar to the training set.

1.2 Related Work and Contribution

The literature on detection and recognition of objects and humans (people, pedestrians) is vast. Approaches vary considerably depending on the image cues and the prior knowledge used, and on the application area ranging from pure detection to pose recovery in 2D and 3D to tracking in cluttered scenes, in connection with surveillance tasks, driver-assistance systems, biometric recognition or human-machine interface design. No attempt will be made to provide a corresponding review here. We confine ourselves to pointing out a few key issues.

As a parts-based approach, our work differs from representations in terms of “bag of features” [15,45,61] that are sensitive to the robustness of local detection, and from approaches relying on silhouette-based representations that coarsely quantize the corresponding data manifold through clustering in order to handle articulation and different aspects [58,76,28]. Furthermore, purely contour-based approaches [1,46] relying on shape context [4] are likely to fail in scenes with cluttered background.

Our approach is view-based in order to keep its applicability to different object categories straightforward. We do not exploit category-specific 3D prior knowledge as e.g. in [8] for the case of humans, or as in [41,55,62,2] in very detailed form.

Rather, we integrate state-of-the-art approaches to robust feature extraction and fast detection [44,16,42] into a complete-graph based conditional random field model of configurations of parts [40,51,39], and assess its performance for view-based detection of humans, and also for detecting instances of less variable object categories (human spines and faces) for comparison.

Regarding occlusions of parts, a careful study of occlusion-sensitive local likelihoods was provided in [59]. Corresponding additional occlusion constraints create loops in the graphical model that are coped with approximate belief-propagation. In the same context, the authors of [54] point out the importance of keeping the number of parts variable. Inference involves local grouping and local optimization in a feed-forward manner whose performance, however, appears difficult to assess from the viewpoint of optimization.

Regarding probabilistic models of spatial part configurations, Gaussian distributions have been proposed in [48] that can only be accurate for restricted set of human poses, however. Likewise, computationally more convenient tree-structured models do not explicitly model relations between all object parts. As a consequence, they may tend to detect both arms and legs at the same position, for instance, and therefore have been mainly applied to views taken from a similar viewpoint [21,23]. To overcome this shortcoming, configurations are sampled from the tree-distribution and evaluted by a global objective function. Recent work [59,35,30] has shown, however, that using additional relations in terms of acyclic graphs can enforce correct configurations between body parts. We report experiments comparing tree-structured and non-tree-structured models in Sec. 5.5.

We point out, on the other hand, that tree-structured models benefit from a small number of parameters, and have recently shown to be extensible to weakly-supervised learning settings [24]. Furthermore, the paper [52] comprehensively elaborates on-line learning in order to adapt a general human model to specific detected object instances in the spatio-temporal context.

A notable difference to our work concerns the meaning of nodes. Whereas most approaches, e.g. [26,52,59,21], choose body parts as nodes we use in this work body joints. This results in a non-redundant parametrization of the articulated object and smaller domains for the random variables assigned to the nodes.

Markov random field (MRF) inference [72,67] is an established technique in computer vision. For a review of linear programming (LP) based relaxation, we refer to [70], for an experimental evaluation of various techniques in connection with standard applications to [65], and for performance bounds based on weak duality to [38]. In recent research, graphical structures that are more densely connected than image grid graphs

[1] This is by no means clear *a priori* because our graphs differ considerably from the more common regular grid-graphs in other problems having typically a smaller number of states.

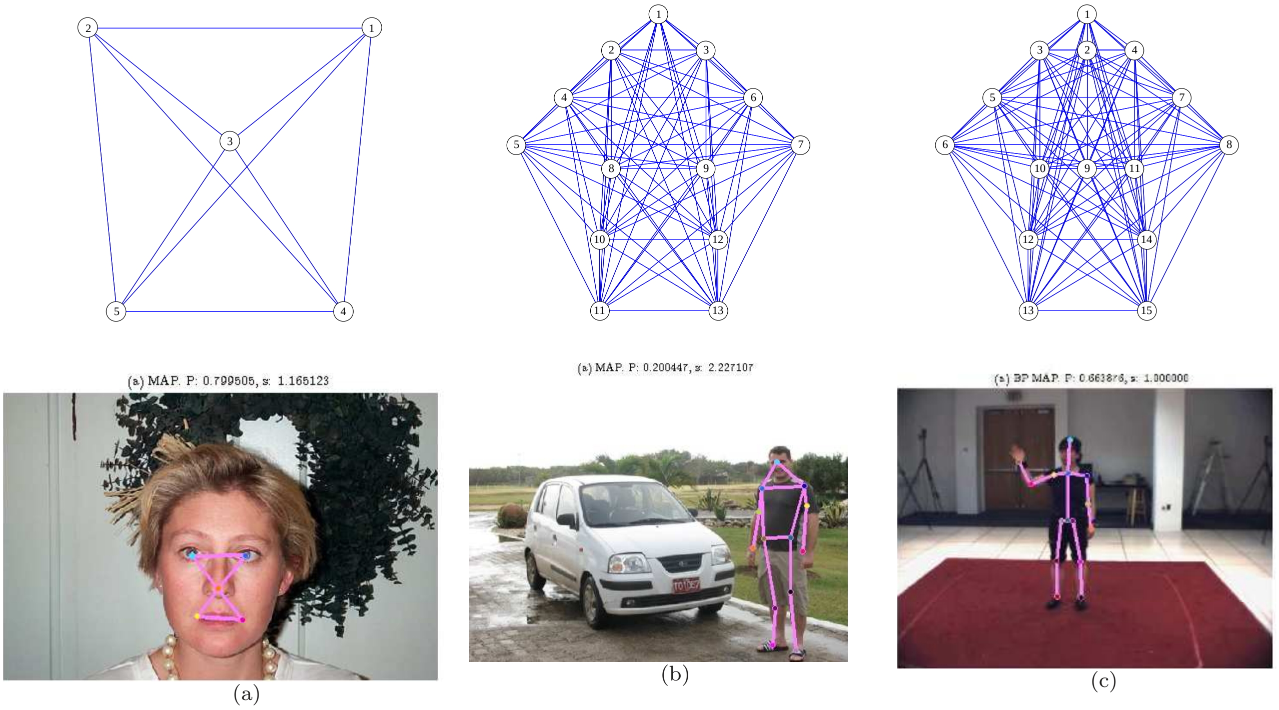

2
1
3
5
4
1
2
3
4
6
5
7
8
9
10
12
11
13
1
3
2
4
5
7
6
8
10
9
11
12
14
13
15
(a) MAP. P: 0.799505, s: 1.165123
(a) MAP. P: 0.200447, s: 2.227107
(a) BP MAP. P: 0.663876, s: 1.000000
(a)
(b)
(c)

Fig. 1 Top: Graphs for objects from three different data sets. Bottom: Detected configurations of object parts. The datasets are Face (a), Human (b), and HumanEva (c). Figure 18 shows a further example from 3D medical imaging. All these cases are handled by our approach in a uniform manner.

have become more important [37]. In this context, the ground truth evaluation of our model, utilizing complete graphs with a very large number of node states, sheds additional light on this important topic. Somewhat unexpected, the A^*-search technique based on a novel lower bound as admissible heuristic outperforms all techniques for *small* complete graphs, as e.g. used for face detection.

Organization

We introduce the basic notation in Section 2 and detail the components of our graphical model of object classes. In Section 3, we summarize an established variational approach to approximate inference and detail the lower bound estimate and the search algorithm for globally optimal, exact inference. The learning algorithm and the corresponding model parameters are considered in Section 4. A fairly comprehensive experimental evaluation using four different data sets along with a discussion is provided in Section 5. We conclude in Section 6 and point out directions of further research.

2 Graphical Model

In this section, we detail the components of our probabilistic representation of object views. After fixing some basic notation, we distinguish discriminative local models of object part appearance, and generative contextual models for the geometry of part configurations. Both components are combined in a probabilistic graphical model.

2.1 Basic Notation

We adopt the common notation in the literature (e.g., [14]). $|A|$ denotes the cardinality of a finite set A. For a fixed object category, let $G = (V, E)$ denote the respective graph depicted in Figure 1, with vertices $s \in V = \{1, 2, \dots, |V|\}$, And with a *complete* set of edges $st \in E \subset V \times V$, $S \neq t$. Edges are undirected, so we identify $st = ts$. We set $\mathcal{C} := V \cup E$. For a subgraph $T \subset G$, $E(T)$ denotes the corresponding set of edges of T.

As illustrated in Figure 1, each vertex s is uniquely assigned to a fixed part of objects of the category. The positions of parts $s \in V$ in a given image, that is its location x_s on a subset $\mathcal{X}_s \subset \mathbb{Z}^d$ of the regular image grid

$\mathbb{Z}^d$ of dimension $d = 2$ or $d = 3$, are given by the vector of random variables $x = (x_1, \ldots, x_{|V|})^\top$ defined over the graph G, and indexed by V. We denote $\mathcal{X}_1 \times \cdots \times \mathcal{X}_{|V|}$ with $\mathcal{X}$. For the sake of readability, we will use the common shorthand $x_S := (x_{s_1}, \ldots, x_{s_{|S|}})^\top$, $s_1, \ldots, s_{|S|} \in S$, for any $S \subseteq V$. In particular, $x_V = x$, and x_c, $c \in \mathcal{C}$, may either denote x_s, $s \in V$, or $(x_s, x_t)^\top$, $st \in E$.

The probability of a particular localization x of an object is modeled by the Gibbs distribution

$$p(x|\theta) = \frac{1}{Z(\theta)} \exp\big(-J(x|\theta)\big) , \tag{1a}$$

$$J(x|\theta) = \sum_{c \in \mathcal{C}} \theta_{c;x_c} , \tag{1b}$$

with the normalizing partition function $Z(\theta)$ ensuring $\sum_{x \in \mathcal{X}} p(x|\theta) = 1$. The term $J(x|\theta)$ is referred to as energy of the Gibbs distribution. $\theta_{c;x_c}$ denotes the potential of the part c of x as detailed in Section 2.2.

It will be convenient to interchangeably use another common parametrization of (1) in terms of all possible values of x. Defining the index set

$$\begin{aligned} \mathcal{I} :=& \big\{(s;j)\,,\ s \in V,\ j \in \mathcal{X}_s\big\} \\ &\cup\ \big\{(st;jk)\,,\ s,t \in V,\ j \in \mathcal{X}_s,\ k \in \mathcal{X}_t\big\} \end{aligned} \tag{2}$$

and corresponding indicator functions

$$\begin{aligned} \phi(x)_{s;j} &:= \begin{cases} 1 & \text{if } x_s = j \\ 0 & \text{otherwise} \end{cases} \\ \phi(x)_{st;jk} &:= \begin{cases} 1 & \text{if } x_s = j \ \wedge\ x_t = k \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{3}$$

we write with a corresponding vector $\phi(x) \in \mathbb{R}^{|\mathcal{I}|}$ and a parameter vector $\theta \in \mathbb{R}^{|\mathcal{I}|}$:

$$p(x|\theta) = \frac{1}{Z(\theta)} \exp\big(-J(x|\theta)\big) , \tag{4a}$$

$$J(x|\theta) = \big\langle \theta, \phi(x) \big\rangle = \sum_{\alpha \in \mathcal{I}} \theta_\alpha \phi(x)_\alpha . \tag{4b}$$

The functional dependence of θ on observed image data I will be detailed below.

2.2 Potential Functions

To cope with the large variability of image data and the complex dependencies therein, we transform the information into a set of *scalar-valued* feature functions. Each potential function $\theta_{c;x_c}$ in equation (1b) can be written as a weighted sum of the individual contributions:

$$\theta_{c;x_c} = \sum_{f \in \mathcal{F}} \lambda_{c,f} f_c(x_c) \tag{5}$$

with model weights $\lambda_{c,f}$, feature functions $f_c(x_c)$, $c \in \mathcal{C}$, $f \in \mathcal{F}$, where the function types are

$$\mathcal{F} := \text{functions of } \begin{cases} \text{appearance} & \text{if } c \in V\ , \\ \text{appearance, length,} & \\ \text{orientation, epipolar} & \text{if } c \in E\ . \end{cases} \tag{6}$$

Unary features may depend on one site x_s and the image I; *Pairwise* or *edge features* may depend on two sites (x_s, x_t) and the image. Input features to the feature functions are: SIFT-features [44], color features, edge length, edge orientation, and epipolar residuals. All features have the property to depend at most on two image sites, which allows us to compute exhaustively all unary terms and a sufficiently large set of edge terms. The features are described in detail in Sections 2.3, 2.4, and 2.5.

2.3 Object Appearance

Input features. Each feature function reduces a feature vector to a scalar.

The input feature vectors are computed from a window at each site, Figure 2 shows some example windows. For the 2D datasets (Human, HumanEva, Face, see Section 5) we compute

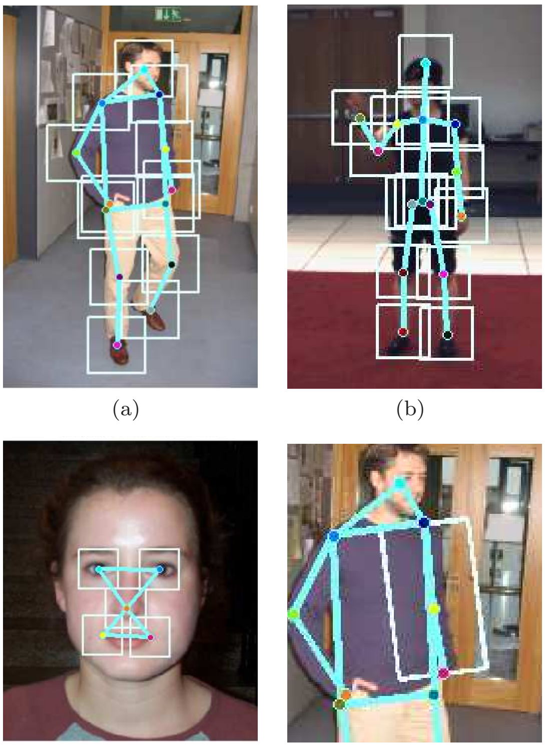
- SIFT features [44] with 8 × 8 spatial and 10 orientation bins at a fixed scale. Concatenation yields feature vectors of dimension $8 \times 8 \times 10 = 640$.
- Color features with 4 × 4 spatial bins in the L*a*b color space. Here each bin contains the average color of pixels falling inside it. Concatenation yields feature vectors of dimension $4 \times 4 \times 3 = 48$.

The same two features computed in windows aligned along the edge that connects two object parts have been used for pairwise appearance features, see Figure 2. For edges between physically connected body parts these pairwise appearance features are in fact “limb-like” as by construction they are invariant to translation, rotation and foreshortening.

For the 3D dataset (Spine) we compute

- Intensity features with 15×15×15 spatial bins which correspond one-to-one to the 3D window size of the input sub-volume. Concatenation yields feature vectors of dimension $15 \times 15 \times 15 = 3375$.

No pairwise appearance features have been used for this dataset.



(a)

(b)

(c)

(d)

Fig. 2 Ground truth configurations for the Human (a), HumanEva (b) and Face (c) datasets with corresponding (uniform) window sizes for local appearance computations. (d) shows a window for computation of pairwise appearance between left-shoulder and left-hand. Note the cyan-edges are only for visualization, and features are generally computed between *all* pairs.

Randomized classification trees. We use randomized classification trees [29] too compute scalar features given feature vectors. They allow for fast evaluation, are able to cope with large training data, and can be used to detect points of interest [42]. Randomized classification trees divide the feature space into arbitrary regions and build a statistic of the training features in each region. This process is repeated using many trees, e.g. about 100 in our case, and combining the individual statistics by averaging the class counts. By this the effect of the hard region boundaries is reduced. Training such a classifier amounts to creating a set of decision trees and collecting the statistics of the training data under the trees' classifications. The branching tests at tree-nodes that divide the feature space are chosen at random from a set of very simple tests, each involving only one or two feature space dimensions.

In particular, we have adopted three types of tests for branching

- T1: At each node of the tree, two dimensions dim_1 and dim_2 of an input feature vector v are chosen at random and their respective values compared. If $v(\mathrm{dim}_1) < v(\mathrm{dim}_2)$, we descend the left branch of the node, otherwise we descend the right branch.
- T2: At each node of the tree, one dimension dim of the input features and a threshold value val in the range of $v(\mathrm{dim})$ are chosen randomly. If $v(\mathrm{dim}) \leq \mathrm{val}$, we descend the left branch of the node, otherwise the right branch.
- T3: This test is only used for pairs of input features. At each node, two dimensions dim_1 and dim_2 are chosen randomly. For two input features v_1 and v_2, if $v_1(\mathrm{dim}_1) \leq v_2(\mathrm{dim}_2)$, we descend the left branch, otherwise the right.

Among the three tree types we selected the best performing as feature function generator for the input features. These are T1 for SIFT features and 3D Spine intensity features, T2 for color features, and T3 for pairs of color features for color similarity.

Building the statistics for the trees stopped when the number of training samples falling into a leaf was smaller than a given threshold (a value of 10 was used throughout the experiments), or if it only contained samples of a single part. This method seemed favourable compared to others defining a maximum depth of the trees, as in our case the tree depth automatically adapts to the number of training samples.

The overall performance and robustness against noise results from aggregation of the statistics over a large number of such tests that are distributed over the ensemble of decision trees. For Human, HumanEva and Face, we used 100 trees for the unary features and 70 for the pairwise features; for the Spine 150 trees were used. The class-specific scalar feature value is obtained by classifying a candidate feature vector, i.e. at each tree-node we descend into the corresponding sub-tree until we reach a leaf. The number of all training samples in the leafs, corresponding to the class and accumulated over all trees, divided by the number of all training samples accumulated over the respective leafs, yields the scalar feature. The final feature function value is obtained after a non-linear calibration method described next.

Feature calibration. The computed scalar features or classification scores have the property that higher values indicate higher probability that the feature being observed in the window at the site corresponds to a particular object-part. When combining different classification scores, it is important that they span comparable ranges [53]. In previous publications [6, 7], we have successfully applied a form of logistic regression to classifier scores, which were obtained using support vector machines (SVMs). The method and optimization was first proposed by Platt [49] for SVMs, as a reliability diagram[2] of their output showed a distinctive sigmoid-shape. For other types of classifiers, however, this may not be the case. For the randomized classification trees, we opted for *isotonic regression*, which was proposed by Zadrozny and Elkan [74] as an alternative for classifier calibration. Here a stepwise-constant isotonic – i.e. order-preserving – function is fitted by minimizing the mean-squared error to the empirical class membership probability using the pair-adjacent violators (PAV) algorithm [17]. This method allows arbitrary classification scores as long as higher scores indicate higher class membership probability. It has been noted [53] that for multi-class classification by combining binary classifiers, the one versus all (OVA) classification scheme for well calibrated classifiers yields similar results than more complicated schemes, e.g. methods inspired by error correcting codes. The effect of feature calibration is visualized in Figure 3.

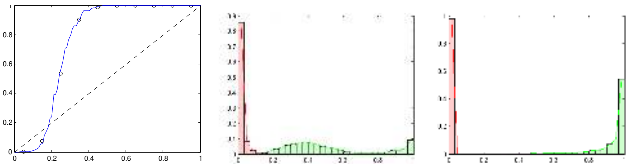
To reduce the set of functions for fitting and also the number of weight-parameters $\lambda_{c,f}$, we averaged the individual classifier scores of the SIFT and color classifiers, separately for each vertex and each edge, by taking their geometric mean in order to obtain a combined classifier score. For example, the *combined appearance score* for a node is computed as

$$s_{\text{appearance}} := \sqrt{s_{\text{SIFT}} \cdot s_{\text{color}}} \tag{7}$$

To these scores we fitted the isotonic function and refer to the resulting functions as "appearance probabilities" $p_{\text{a,c}}(x_c, I)$ for class $c \in \mathcal{C}$ at site location x_c for the image I. We have found that this yields also slightly better results with respect to classification than first fitting the isotonic function to each feature type and taking the geometric mean afterwards.

Finally, in terms of the the energy formulations (4b) and (5), the negative logarithm of the appearance probability yields the feature function

$$f_c(x_c) := -\ln p_{\text{a,c}}(x_c, I),\ f = \{\text{appearance}\}$$



(a) Reliability diagram (b) Histogram before calibration (c) Histogram after calibration

Fig. 3 Classifier calibration on appearance features for the right knee in the HumanEva dataset. (a) reliability diagram: x-axes denote the combined, uncalibrated classifier scores, y-axes denote the reliability, i.e. the observed relative frequency for an in-class feature having the respective classifier score; the blue curve denotes the non-parametric isotonic function fit, the black line corresponds to an ideal reliability and black dots indicate estimated reliability (using histograms). The isotonic functions follow the estimated reliability values closely. (b) and (c): Histograms of classifier scores before (b) and after calibration (c). In green the normalized histogram for in-class features, in red the normalized histogram for out-of-class features. Note that the calibrated scores are closer to ideal probabilistic scores, i.e. a score of 1 for in-class and 0 for out-of-class features.background

2.4 Object Shape

The feature functions for object shape are derived from simple 1D histograms. As input features we used the Euclidean distance between pairs of sites constituting an edge in the graph, and the absolute edge orientation. Unary terms, e.g. absolute part locations, were not used. In other words, we only model object shape, not absolute location. Whereas these features are thus invariant to object translation, the edge-length feature is not invariant to changes in scale and the orientation feature is not invariant to in-plane rotations. For training we normalized the scale by computing $r_{st} = \frac{\mu_{st}}{l_{st}}$ for all available edges $st \in E$ of the observed object, where μ_{st} denotes a normalized edge length and l_{st} is the observed length. We assume that variations in r are due to global scale and foreshortening. Foreshortening causes r_{st} to be overestimated as the observed l_{st} is shorter than the true edge-length. To account for foreshortening we assume that at least one edge is not foreshortened so that l_{st} is the true image-length of that edge and thus take the minimum over r_{st} as the scale normalization factor r. Clearly though, the effects of foreshortening will still hamper the length features. For inference on the test images we treat object scale as a latent variable for the length features, i.e. the features are computed after normalization with the hidden/unknown scale parameter r, that has to be inferred. In contrast we ignore the dependency of the orientation features to in-plane rotations for two reasons: (1) our particular objects usually have one predominant orientation in images and (2) where this assumption does not hold, e.g. standing vs. lying humans, this will be reflected in the histograms since these do allow for multiple modes. Clearly configurations that do not correspond to major modes will be hard to detect with this approach.

No calibration, as done for the appearance terms, was performed, as we assume that false edge candidates will follow uniform distributions, so we can expect that the reliability diagram is a straight line passing through the origin. We refer to the histogram outputs as length probability $p_{\text{l,c}}(x_c|r)$ and orientation proba-

[2] A reliability diagram plots the empirical class membership probability vs. the classification score, e.g. Figure 3.

bility $p_{\mathrm{o,c}}(x_c)$, where $c \in E$ and x_c denotes the two image sites corresponding to edge c. The feature functions for the energy formulation:

$$f_c(x_c) := -\ln p_{\mathrm{l,c}}(x_c|r),\ c \in E,\ f \in \{\text{length}\}\ ,$$
$$f_c(x_c) := -\ln p_{\mathrm{o,c}}(x_c),\ c \in E,\ f \in \{\text{orientation}\}\ ,$$

are again the negative logarithm of the histogram output.

2.5 Epipolar Constraints

For the HumanEva dataset, up to 7 calibrated images were taken at each time instant from different directions. We made use of this additional information by combining the configurations of all available cameras into a single model, where individual configurations in the images must satisfy additional *pairwise* constraints given by the epipolar geometry. For an image pair I_1, I_2 and two points $x_1 \in I_1$, $x_2 \in I_2$ in correspondence, i.e. imaging the same 3D world point, the epipolar constraint

$$x_1^\top F_{12}\, x_2 = 0$$

must be satisfied, where F_{12} is the fundamental matrix [32] of the image pair.

For the number of images used simultaneously, we use the original graph as depicted in Figure 1 (c) one for each view, augmented by edges between all parts with the same label in each combination of image pairs. The corresponding model graph is therefore not fully connected in this case. The input features for the additional edges are the algebraic residuals of the epipolar constraint $\left|x_{c,i}^\top F_{ij} x_{c,j}\right|$ for each part $c \in V$ and image pairs I_i, I_j. We compute 1D histograms of these features, analogously to the object shape features and refer to them as epipolar probability $p_{\mathrm{e,c}}(x_c)$. With a slight abuse of notation:

$$\begin{aligned} f_c(x_c) := -\ln p_{\mathrm{e,c}}(x_c),\ c = (s,i,j),\ s \in V, \\ i,j \in 1,\ldots,7,\ i \neq j,\ f \in \{\text{epipolar}\} \end{aligned}$$

2.6 Image Graph and Missing Parts

When building the image graph corresponding to a model-graph (Figure 1) in a bottom-up process we make in each step use of previous computations to prune the graph to manageable size in terms of computational effort and memory footprint. For a given test-image, we proceed as follows:

1. Compute the appearance probability $p_{\mathrm{a,c}}(x_c, I)$ for all parts $c \in V$ and all corresponding image sites $x_c \in \mathcal{X}$.
2. For a fixed, per-part threshold T_c, sample a set of candidate part locations $\mathcal{X}_c$ by including all sites with $p_{\mathrm{a,c}}(x_c, I) > T_c$. Additionally, we use non-maxima suppression to discard image sites nearby a sampled candidate.
3. Compute the lengths $p_{\mathrm{l,c}}(x_c|r)$ and orientation probabilities $p_{\mathrm{o,c}}(x_c)$ for each edge $c \in E$ and the set of sampled candidates. For the object scale r we have usually used 5 discrete settings. We employ hard thresholds for the edge-length, such that if it is smaller or larger than the any observed edge-length in the training set, the corresponding probability is set to zero.
4. Only for non-zero edge candidates, compute the appearance probability $p_{\mathrm{a,c}}(x_c)$, $c \in E$.

Figure 4 shows examples of appearance probability maps and candidate samples.

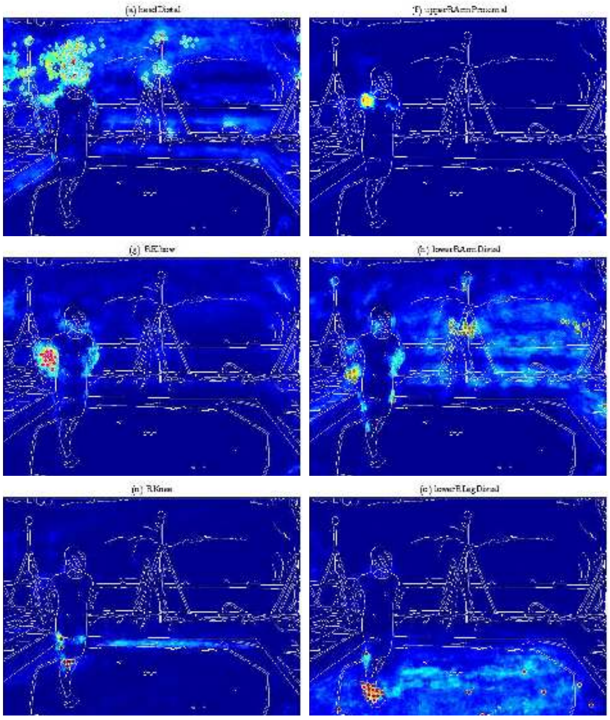


Fig. 4 Appearance probability maps $p_{\mathrm{a,s}(x_s,I)}$ and candidate samples $\mathcal{X}_s$ for an image of the HumanEva dataset. From left to right and top to bottom: headDistal, upperRightArmProximal, rightElbow, lowerRightArmDistal, rightKnee, lowerRightLegDistal.

By proceeding in a bottom-up manner, only a relevant subset of image sites are considered as possible sites for the object parts. Furthermore, the graph has a locality property, as candidates with large distance get an edge probability of zero, which is used to speed up subsequent inference. The thresholds T_c, $c \in V$ can be set by the user. In order to get a good compromise between missing detections and computational complexity, we have chosen the thresholds at the operating point where the individual classifiers maximize the F_1-measure [20] on all training features

$$F_1 := 2\,\frac{\#\text{true positive}}{\#\text{positive detections} + \#\text{positive features}}$$

The image graph is the input to the MAP inference problem, which is to match the model graph to the subgraph with minimal energy as described in Section 3.

Missing parts. There are two natural reasons for missing parts. Firstly, the part can be occluded by another part or object. Secondly, the part may not be contained in the image section. Moreover, by employing the thresholds T_c, some of the parts may be missed during sampling. Therefore, we include a special candidate m_c for each part and derive corresponding appearance probabilities, edge-length probabilities, edge-orientation probabilities, and epipolar probabilities. "Special" means that m_c has no location, hence feature functions cannot be computed in the usual way.

In Bayesian inference, for each hypothesized miss in a configuration, we need to marginalize over the image domain to account for the hypothesis. However, the number of hypotheses already grows combinatorially: As every possible combination of present and missed parts is conceivable, the number of hypotheses is $2^{|V|}$. Moreover, computing all the edge terms is computationally unattractive as their number grows quadratically with the number of image sites or pixels.

Instead, we propose the following, more efficient approximation. First, instead of marginalization, a common approximation is to search for the maximally likely missing part, which also better suits MAP-inference in Section 3. Note that the highest attainable appearance probability for the missing part is exactly the threshold T_c. So we set

$$p_{\mathrm{a,c}}(m_c) := T_c,\ c \in V \tag{8}$$

Next, we define the edge probabilities for the missing candidate. Assuming that the miss is only caused by the local appearance probability lying below the threshold, but that pairwise edge probabilities would not be affected by this "failure" to recognize the part, we argue that the true part would lead to *typical* edge probabilities, in which case we define the edge probabilities by their typical values. We have chosen the mean of each of the appearance, length, and orientation probabilities for the three types of edge terms using a validation dataset $\mathcal{D}_V$

$$\left.\begin{aligned} p_{\mathrm{a,c}}(m_c) &:= \underset{x_c, I \in \mathcal{D}_V}{\mathrm{mean}}\ p_{a,c}(x_c, I) \\ p_{\mathrm{l,c}}(m_c) &:= \underset{x_c, I \in \mathcal{D}_V}{\mathrm{mean}}\ p_{l,c}(x_c) \\ p_{\mathrm{o,c}}(m_c) &:= \underset{x_c, I \in \mathcal{D}_V}{\mathrm{mean}}\ p_{o,c}(x_c) \\ p_{\mathrm{e,c}}(m_c) &:= \underset{x_c, I \in \mathcal{D}_V}{\mathrm{mean}}\ p_{e,c}(x_c) \end{aligned}\right. \quad \forall c \in E \tag{9}$$

where the x_c denote the true locations of the parts in the respective image. One might argue that the miss of the part may originate from other causes, in particular from occluding objects or self-occlusion, for which the part appearance probability at the true location will certainly be below T_c, as well as its edge-appearance counterparts. So the estimates serve as an optimistic guess and for experiments that rely on this heuristic, we have introduced the weight parameter $\gamma \leq 1$ by which we multiply the appearance probabilities for missing parts and edges.

We have found that the above heuristic already gives quite reasonable results and, where model complexity or an insufficient number of training data did not allow for maximum likelihood learning as proposed in Section 4.1[3], we successfully used this method instead.

In view of alternative approaches [63, 13, 11] that recreate a small number of candidates after few iterations of belief propagation, advantages of our model include independency of the inference method (any technique can be used), and feature functions for new candidates need not be computed in each step. Natural occlusion, however, has still to be modeled by an extra candidate.

3 Inference

In this section, we focus on inference algorithms to compute the Maximum-A-Posterior (MAP) configuration x by minimizing the energy in (4).

From the viewpoint of inference, the design of a graphical model amounts to a difficult compromise between sufficient expressiveness of a model to accommodate the complexity of real visual data and computational manageability. As detailed in the previous section, we restrict our model to a second order MRF which appears to be sufficiently powerful for representing contextual relations.

Regarding computational tractability, we investigate competitively in Section 5 different established inference techniques including *(Loopy) Belief Propagation*

[3] When performing maximum likelihood learning, we only use one feature function for a missing node and one feature function for a missing edge, i.e. the three edge terms are combined.

(BP) [72] and *Tree-Reweighted Belief Propagation (TRBP)* [67]. To this end, we introduce a novel admissible heuristic [6] employing a tree-based lower bound estimate, in order to compute ground truth (global optimum) with A^*-search for not too large problem instances.

We sketch TRBP below and then detail A^*-search tailored to our problem class. For ordinary BP, we refer to [72].

3.1 Tree-Reweighted Belief Propagation (TRBP)

Wainwright [67] proposed a convex relaxation of the intractable problem to compute the most likely configuration x by minimizing the energy in (4)[4]

$$\Phi(\theta) := \min_x J(x|\theta_T) = \max_x \langle \theta, \phi(x) \rangle \ . \tag{10}$$

Representing the parameter vector θ by a convex combination of parameters θ_T,

$$\theta = \sum_{T \in \mathcal{T}} \rho_T \theta_T \ , \quad \rho_T > 0 \ , \quad \sum_{T \in \mathcal{T}} \rho_T = 1 \ ,$$

corresponding to the set $\mathcal{T}$ of all spanning trees $T \subset G$ as tractable substructures of the underlying graph G, the convexity of $\Phi(\cdot)$ and Jensen's inequality yield the upper bound

$$\Phi(\theta) = \Phi\Big(\sum_{T \in \mathcal{T}} \rho_T \theta_T \Big) \leq \sum_{T \in \mathcal{T}} \rho_T \Phi(\theta_T) \ . \tag{11}$$

Minimizing this upper bound is a convex optimization problem. The corresponding dual program is the linear program (LP) [67]

$$\max_{\nu \in P_\nu} \langle \theta, \nu \rangle \ , \tag{12a}$$

$$P_\nu := \mathbb{R}_+^{|\mathcal{I}|} \cap \Big\{ \nu \ \Big| \ \sum_{j \in \mathcal{X}_s} \nu_{s;j} = 1 \ , \ \sum_{j \in \mathcal{X}_s} \nu_{st;jk} = \nu_{t;k} \ , \ \forall s,t \in V \ , \ \forall k \in \mathcal{X}_t \Big\} \ . \tag{12b}$$

The set P_ν in (12b) constitutes a computationally feasible outer-approximation of the marginal polytope, and the pseudo-max-marginals ν are related to, and computed by messages propagated along the edges of G [67]. The basic update equations read

$$\hat{M}_{ts}^{n+1}(i) = \max_{j \in \mathcal{X}_t} \bigg\{ \exp\left(\frac{\theta_{st;ij}}{\rho_{st}} + \theta_{t;j} \right) \cdot \frac{\prod_{v \in N(t) \setminus \{s\}} M_{vt}^n(j)^{\rho_{vt}}}{M_{st}^n(j)^{(1-\rho_{st})}} \bigg\} \ , \tag{13}$$

$$M_{ts}^{n+1}(i) = M_{ts}^n(i)^{1-\beta} \cdot \hat{M}_{ts}^{n+1}(i)^{\beta} \ , \tag{14}$$

where ρ_{st} denotes the relative frequency [5] of edge st belonging to a tree in $\mathcal{T}$, i.e. $st \in E(T)$, $T \in \mathcal{T}$, and where $N(t)$ is the set of all vertices adjacent to node $t \in V$.

Algorithm 1 Tree Reweighted Belief Propagation

```
[ $x$ ] ← **TRBP** ( $\theta$, $\rho$, $N$, $\beta$ )
$\forall st \in E,\ i \in \mathcal{X}_s,\ j \in \mathcal{X}_t : M_{st}^0(j) \leftarrow 1,\ M_{ts}^0(i) \leftarrow 1$
**for** $n = 1 \dots N$ **do**
    **for all** $st \in E$ **do**
        Compute updates (13) and (14) for both edge-directions
        and all $i$'s.
    **end for**
**end for**
**for all** $s \in V$ **do**
    $x_s \leftarrow \arg\max_{i \in \mathcal{X}_s} \exp(\theta_{s;i}) \prod_{t \in N(s)} M_{ts}(i)^{\rho_{st}}$
**end for**
```

As for standard BP, this algorithm is exact for acyclic graphs. For cyclic graphs fixed points are related to stationarity points of the dual LP (12). Convergence can be enforced by damping the update equations with a factor $\beta < 1$ [67], but cannot be guaranteed in general. A modification of TRBP by Kolmogorov [36] converges to a vector satisfying the weak tree agreement condition. We cannot apply it here because a required monotonicity property does not hold as we do allow more general potential functions.

3.2 Global Optima Via Lower Bounds and A^* Search

The A^*-algorithm is an established technique in order to cope with very large state spaces in the dynamic programming (DP) framework, when searching for a globally optimal solution to intricate problems – see [31, 47, 73, 12]. For applications in computer vision we refer to e.g. [25, 12, 48, 22]. The optimal solution is computed in terms of the shortest path within a weighted graph that represents the whole configuration space and the corresponding costs defined by the objective function. Its performance depends on devising a heuristic that estimates the costs of unexplored paths from the current node representing a partial solution, to a terminal node indicating a complete solution. In order to find the *global* minimum, the heuristic has to be *admissible*, i.e. it has to provide a *lower bound.* While this ensures global optimality once the search terminates, its complexity may be exponential in the problem size. Lower bounds on the runtime can be given in some cases when, e.g., the estimated error does not grow faster than the

[4] In order to conform to the literature and to avoid confusion of the reader, we temporarily reverse – in this subsection only – the sign of the representation (4b), $J(x|\theta) = -\langle \theta, \phi(x) \rangle$, that is we maximize the right-hand side. In other sections of the manuscript, we prefer the energy interpretation of θ_α.

[5] Due to the symmetry of complete graphs, we simply have $\rho_s t = |E(T)|/|E|$

logarithm of the true minimal costs, then the time complexity is polynomial [47,56]. It is not clear, however, how to achieve this in practice.

We detail next the *admissible* search heuristic, and subsequently the A^*-search algorithm used to compute ground truth for evaluating approximate MAP-inference algorithms. Interestingly, it turned out that for small problem sizes, e.g. when using small complete graphs for face detection, our algorithm outperforms BP-based algorithms not only with respect to optimality (by construction), but also with respect to *runtime*.

3.2.1 Admissible Search Heuristic

We transform the MAP inference problem to a shortest path problem with respect to a search tree[6] $T^* = (V^*, E^*)$. To define T^*, we assume to be given

- a spanning tree $T \subset G$ that is used to compute a lower bound. T is determined depending on the application. For instance, in the case of the HumanEva dataset, we use the tree T that covers the physical edges displayed in cyan in Figure 2.
- an arbitrary but fixed ordering of all nodes $V = \{1, 2, \dots, |V|\}$ of G.

An example tree T^* for a simple graph is shown in Figure 5.

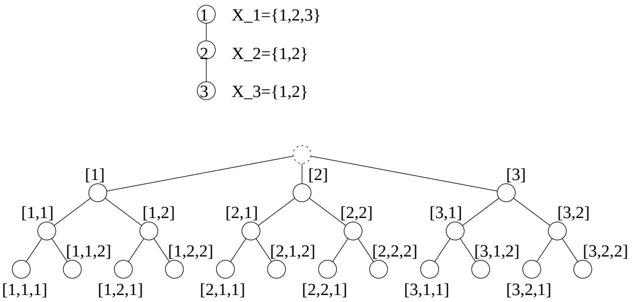


Fig. 5 Top: Graph $G = (V, E)$ and state candidates, bottom: Corresponding search-tree T^*.

Nodes V^* of T^* then represent all possible (sub) configurations x_S, $|S| = 0, 1, \dots, |V|$, where $S \subseteq V$ respects the order of V. With slight abuse of notation, we write

$$v^* = x_{v^*} = \{x_1, x_2, \dots, x_{|v^*|}\} \tag{15}$$

to emphasize this identification between nodes $v^* \in V^*$ and partial assignment of values to x. Complete configurations x_{v^*}, $|v^*| = |V|$, form the leaves of T^*. We denote the root by [ ] (empty configuration). Formally, we have

$$V^* = \{[]\} \cup \bigcup_{s \in V} \bigotimes_{t=1}^{s} \mathcal{X}_t \ ,$$

where s runs through V in the predefined order.

Nodes u^*, v^* are adjacent in T^*, $u^*v^* \in E^*$, if the two configurations they represent differ by an additional single random variable $x_{|v^*|}$,

$$E^* = \big\{u^*v^* \in V^* \times V^* \ \big|\ |u^*| + 1 = |v^*|\,,\ u^* \subset v^*\big\}\ .$$

Each of these edges carries the weight

$$w(u^*, v^*) = \theta_{t,x_t} + \sum_{s=1}^{|u^*|} \theta_{st;x_s x_t}\ , \quad t = |v^*|\ , \tag{16}$$

that equals the additional energy due to extending a partial solution u^* to v^* (recall notation (15))

Lemma 1 *For all $v^* \in V^*$, the distance $d([\,], v^*)$ in T^* equals the energy $J(v^*)$.*

Proof For the unique path with nodes $\{[\,], v_1^*, \dots, v_{|v^*|}^*\}$ in T^*, we obtain with (16)

$$\begin{aligned} d([\,], v^*) &= \sum_{s=1}^{|v^*|} w(v_{s-1}^*, v_s^*) = \sum_{s=1}^{|v^*|} \Big(\theta_{s;x_s} + \sum_{t=1}^{s-1} \theta_{t,s;x_t,x_s}\Big) \\ &= \sum_{s=1}^{|v^*|} \theta_{s;x_s} + \sum_{\substack{ts \in E \\ s,t \le |v^*|}} \theta_{t,s;x_t,x_s} = J(v^*) \end{aligned}$$

Definition 1 For $u^*, v^* \in V^*$ with $|u^*| \le |v^*|$, we define the search heuristic

$$\begin{aligned} H(v^*|u^*) := \min_{\substack{x \in \mathcal{X} \\ x|_{v^*} = v^*}} \Bigg[& \sum_{\substack{t \in V \\ t > |v^*|}} \theta_{t;x_t} + \sum_{\substack{st \in E \\ s \le |u^*|\,,\ t > |v^*|}} \theta_{s,t;x_s,x_t} \\ & + \sum_{\substack{s > |u^*| \\ t > |v^*|}} \Bigg(\sum_{st \in E(T)} \theta_{s,t;x_s,x_t} + \sum_{st \in E \setminus E(T)} \min_{x_s \in \mathcal{X}_s} \theta_{s,t;x_s,x_t} \Bigg) \Bigg] \end{aligned} \tag{17}$$

Proposition 31. *The heuristic* (17) *is admissible, i.e. it provides a lower bound of the energy corresponding to any path from v^* to a leaf node.*

Proof The minimal energy corresponding to a path from v^* to a leaf node is

$$\min_{\substack{x \in \mathcal{X} \\ x|_{v^*} = v^*}} \sum_{t > |v^*|} \Big(\theta_{t;x_t} + \sum_{st \in E} \theta_{s,t;x_s,x_t}\Big) \tag{18}$$

The edge set of the second term can be split in two sets corresponding to $s \le |u^*|$ and $s > |u^*|$, respectively, and the latter set can be further split into the set of tree-edges $st \in E(T)$ and its complement $st \in E \setminus E(T)$. Minimizing independently the last term with respect to x_s, as in (17), provides a lower bound.

[6] Note that the tree T^* and corresponding nodes V^* and edges E^*, that is all variables labelled with a $*$, refer to the configuration space for inference. They should not be confused with spanning trees $T \subset G$ or nodes V of the underlying graph G.

Rearranging (17),

$$
\begin{aligned}
H(v^*|u^*) := \min_{\substack{x\in\mathcal{X}\\ x|_{v^*}=v^*}} \Bigg[\sum_{\substack{t\in V\\ t>|v^*|}} \Bigg(& \theta_{t;x_t} + \sum_{\substack{st\in E\\ s\le|u^*|}} \theta_{s,t;x_s,x_t} \\
& + \sum_{\substack{st\in E\setminus E(T)\\ s>|u^*|}} \min_{x_s\in\mathcal{X}_s} \theta_{s,t;x_s,x_t} \Bigg) \\
& + \sum_{\substack{st\in E(T)\\ s>|u^*|,t>|v^*|}} \theta_{s,t;x_s,x_t} \Bigg]
\end{aligned}
\tag{19}
$$

shows that evaluating the lower bound amounts to a *tree-structured* inference problem that can be efficiently carried out using standard message passing.

3.2.2 A^*-Search

We use a heap as data structure in order to handle storage of the large amount of data accumulated during the search, and for efficiently determining the next partial configuration v^* to be extended towards a globally optimal solution.

If it happens that the algorithm exceeds the available memory, we discard 50% of those hypotheses for the global solutions having the highest energy estimates. This is not a problem: In practice, because most paths have very high energies; in theory, because we keep track of the lowest energy of this set, so as to be able to verify global optimality after termination.

Algorithm 2 shows the pseudo-code of the A^*-search used in this paper for the specific case $|u^*| = |v^*| - 1$. We point out that *opt* = FALSE after termination does not imply that the global optimum has not been found. Rather, it implies that global optimality cannot be guaranteed.

This algorithm can be easily modified in various meaningful ways. For example, concerning the heuristic (17), the tighter bound $H(v^*|v^*)$ could be used, but at considerably higher computational costs. The other extreme is $|u^*| = 0$, independently of v^*, i.e. to evaluate just once the estimates $H(v^*|[\ ])$ that are, of course, much less tight.

Our experiments indicate that the choice above, $|u^*| = |v^*| - 1$, is a good compromise.

Algorithm 2 A^*-Search for MAP-Inference

```
[ $x$, $opt$ ] ← AStar ( $\theta$, $T$ )
$v^* \leftarrow [\,]$, $\tau \leftarrow +\infty$
while $|v^*| < |V|$ do
    $u^* \leftarrow v^*$
    compute $H(v^*|u^*)$ , $\forall v^*$ , $|v^*| = |u^*| + 1$
    for $i \in \mathcal{X}_{|u^*|+1}$ do
        $v^* \leftarrow \{u^*, i\}$
        insert$\{v^*, J(v^*) + H(v^*|u^*)\}$ into the heap
        if size = maxsize then
            $\Delta \leftarrow$ lowest value of the 50% worst energy estimates
            $\tau \leftarrow \min(\tau, \Delta)$
        end if
    end for
    $v^* \Leftarrow$ getMin(heap)
end while
$x \leftarrow v^*$, $opt \leftarrow$ FALSE
if $J(x) \le \tau$ then
    $opt \leftarrow$ TRUE
end if
```

4 Model Learning

4.1 Parameter Initialization

Along with the feature functions f described in Section 2, we have to learn model parameters λ for the computation of θ. Recall that (5)

$$\theta_{c;x_c} = \sum_{f\in\mathcal{F}} \lambda_{c,f} f_c(x_c)\ .$$

For every vertex and edge, and for each corresponding feature, a single model parameter has to be estimated. The reasoning for computing an initial guess is as follows:

Initially neglecting all structural information given by the edge terms, we can conceive a detector for recognizing an object by detecting its parts individually. Assuming that all part detectors are independent, the overall probability is the product of the individual probabilities (naive Bayes classifier)

$$p_{a,V}(x) \propto \prod_{c\in V} p_{a,c}(x_c)\ .$$

We initialize all λ parameters corresponding to vertex appearance features with 1. Including the edge appearance probabilities gives us a complementary view of the same probabilistic event. So we could set $p_{a,E}(x) \propto \prod_{c\in E} p_{a,c}(x_c)$. The number of edges is far greater than the number of vertices, however, and edge features may overlap by construction, i.e. the independence assumption of the individual classifiers does not hold. Therefore, if their individual contribution is comparable to the part probabilities, the overall final probability will be much lower than the one above, using parts alone. To account for this, we combine the probabilities in a "products of expert" model [33]

$$p_{a,E}(x) \propto \prod_{c\in E} p_{a,c}(x_c)^{\lambda_{c,a}}.$$

Expecting edge appearance to be of similar quality as the one for vertices, we set

$$\lambda_{c,a} := \frac{|V|}{|E|},\ \forall c \in E,$$

i.e. $\frac{2}{|V|-1}$ for a fully connected graph. Assuming further that length and orientation probabilities are equally informative, their respective λ parameters are initialized in the same way. Now each type of feature (part appearance, edge appearance, length, orientation) gives rise to an *expert*, and the overall probability is again their combination using the "products of expert" model, where we weighted their contribution based on intuition as $\xi_{\text{a,V}} = 0.5$ for the node appearance, $\xi_{\text{a,E}} = 0.25$ for edge appearance, $\xi_{\text{l,E}} = 0.125$ for length, and $\xi_{\text{o,E}} = 0.125$ for orientation. In conclusion we define the initial values of the λ parameters to be

$$\lambda_{c,f} := \begin{cases} \xi_{\text{a,V}} & \text{, if } c \in V \text{ ,} \\ \frac{2}{(|V|-1)}\, \xi_{\text{f,E}} & \text{, if } c \in E \text{ ,} \end{cases} \tag{20}$$

Note that after taking the negative logarithm, the exponential mixture parameters $\lambda_{c,f}$ become the factors in the energy formulation.

The effect of the λ parameters on the individual feature functions corresponds to smoothing of the marginal statistics of the feature function if $\lambda < 1$, and to sharpening of the statistics for $\lambda > 1$, see Figure 6.

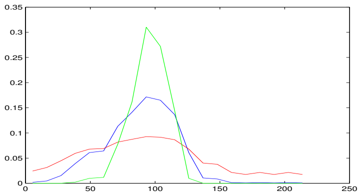


Fig. 6 Effect of the λ parameter on the marginal statistics of a feature function. The figure shows the histogram for the edge length between the left elbow and left hand of the Human dataset. Blue: original distribution, red: histogram after λ-smoothing with $\lambda = 0.3$, green: with $\lambda = \frac{1}{0.3}$, after renormalization.

In practice, the assumptions made here do not strictly hold, since image patches may overlap and features are in general not equally informative. Optimization of the λ parameters can be done in the conditional random field (CRF) framework by maximizing the log-likelihood of the ground truth for a set of training samples [40], as described next.

4.2 Parameter Learning

Given i.i.d. training images with known ground truth $x^1, \ldots, x^D$, we maximize the likelihood function

$$\begin{aligned} L(\lambda) &= \prod_{d=1}^{D} p(x^d|\theta^d) \\ &= \prod_{d=1}^{D} \left[\frac{1}{Z(\theta^d(\lambda))} \exp\left(- \sum_{\alpha \in \mathcal{I}^d} \theta^d(\lambda)_\alpha \phi(x^d)_\alpha \right) \right] \end{aligned} \tag{21}$$

where $\mathcal{F}$, see (5), includes also missing nodes and missing edges to optimize their respective λ parameters as well.

While the optimization of equation (21), or equivalently its logarithm, is the commonly applied approach and a number of algorithms for exact and approximate solutions have been proposed, see e.g. [64,51,66,43], there are two shortcomings of this formulation. First, the number of parameters is large in our case (e.g. 429 for the Human dataset). Secondly, the likelihood function does not variate smoothly in vicinity of the ground truth. In particular, when labelling ground truth configurations, a user is confronted with the difficult decision to label a *single* point on a joint or body part in arbitrary views, as well as deciding if a part is still visible or should be labelled as missed. Without further proof, we feel that different users will most likely label different configurations for the same image. To tackle both problems at the same time, we relax (21) by also including configurations that are similar to the "ground truth" as positive examples. For this we weight part candidates by their distance to their ground truth location x_s^d until reaching a certain distance threshold D_s

$$\overline{w}_s(i) = \begin{cases} 1 - 0.5 \cdot \frac{|x_{s;i} - x_s^d|}{D_s} & \text{, if } |x_{s;i} - x_s^d| < D_s \text{ ,} \\ \delta & \text{, if } x_{s;i} \text{ missing candidate} \\ 0 & \text{, otherwise,} \end{cases} \tag{22}$$

where δ is the penalty for hiding a part. See Figure 4.2 for a visualization of the distance thresholds used.

We maximize a *weighted* log-likelihood, with $w_s(i) = \frac{\overline{w}_s(i)}{\sum_{j \in \mathcal{X}_s} \overline{w}_s(j)}$, such that $\forall s \in V : \sum_x w_s(x_s) = 1$ and $W(x) := \prod_{s \in V} w_s(x_s)$, $\sum_{x \in \mathcal{X}} W(x) = 1$. The corresponding *smoothed* log-likelihood function behaves more gently in the neighborhood of the labelled ground truth, and is maximized using gradient ascent.

$$\begin{aligned} l(\lambda) &= \sum_d \sum_{x \in \mathcal{X}^d} W(x) \cdot \ln p(x|\theta^d) \\ &= \sum_d \sum_{x \in \mathcal{X}^d} W(x) \cdot \Big(- \langle \theta^d(\lambda), \phi(x) \rangle - \ln Z\big(\theta^d(\lambda)\big) \Big) \end{aligned}$$

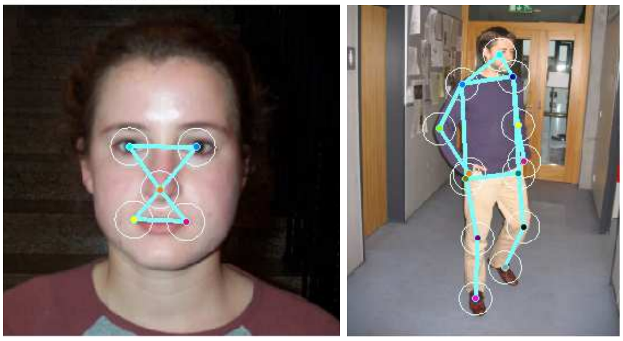


Fig. 7 Distance thresholds used for the Face and Human datasets for learning and evaluation. Circles visualize the (uniform) maximum distance to the manual ground truth location (cyan) for a part to be considered a positive hit.

(23)

Note that compared to (21), where each ground truth configuration x^d is used only once, there is now a weighted summation over the configuration space $\mathcal{X}^d$, where we have used the output after candidate sampling (c.f. Sec. 2.6) as configuration space $\mathcal{X}^d$, thus including the effects of candidate misses due to suboptimal classification.

This can be considered as a minimization of Kullback-Leibler divergence $D_{KL}(W||p)$ between a desired distribution given by $W(x)$ and the estimates given by the model $p(x|\theta)$:

$$
\begin{aligned}
D_{KL}(W||p) =& \sum_x W(x) \cdot \ln \frac{W(x)}{p(x|\theta)} \\
=& \sum_x W(x) \cdot \ln W(x) - \sum_x W(x) \cdot \ln p(x|\theta) \\
=& \text{const} - \sum_x W(x) \cdot \ln p(x|\theta)\,.
\end{aligned}
\tag{24}
$$

The gradients are computed by the partial derivatives as:

$$\frac{\partial}{\partial \lambda_{c;f}} \langle \theta^d(\lambda), \phi(x) \rangle = f_c^d(x_c) \tag{25}$$

$$\frac{\partial}{\partial \lambda_{c;f}} \ln Z(\theta^d(\lambda)) = - \sum_{x_c \in \mathcal{X}_c^d} \mathbb{E}\left[\phi(x)_{c,x_c}\right] \, f_c^d(x_c) \tag{26}$$

$$
\begin{aligned}
\frac{\partial}{\partial \lambda_{c;f}} l(\lambda) = \sum_d \Big(&- \sum_{x_c \in \mathcal{X}_c^d} w_c^d(x_c) f_c^d(x_c) \\
&+ \sum_{x_c \in \mathcal{X}_c^d} \mathbb{E}\left[\phi(x)_{c,x_c}\right] \, f_c^d(x_c) \Big)\,.
\end{aligned}
\tag{27}
$$

The computation of $\mathbb{E}\left[\phi(x)_{c,x_c}\right]$ is known to be difficult for general graphs. However, we can obtain good approximations using BP/TRBP for fixed λ. The derivation is the same as in Section 3.1, we only exchange the max-product- by the sum-product-semiring. Using this approximate gradient we perform K gradient ascent steps to optimize the parameters. The value of the step size parameter was chosen $\eta = 0.01$.

Algorithm 3 Learn parameter λ

```
[ $\lambda$ ] ← **Learn** ( $\theta^{[1,\dots,D]}$,$w^{[1,\dots,D]}$, $\lambda^0$, $\eta$ )
  $k \leftarrow 0$
  **for** $k = 1,\dots,K$ **do**
    **for** $d = 1,\dots,D$ **do**
      $\forall c \in C$ , $\forall x_c \in \mathcal{X}_c$ , compute $b_c^d(x_c) \approx \mathbb{E}\left[\phi(x)_{c,x_c}\right]$ with TRBP.
    **end for**
    **for** $c \in C$ **do**
      **for** $f$ **do**
        $\lambda_{c,f}^{k+1} \leftarrow \lambda_{c,f}^{k} + \eta \sum_d \sum_{x_c \in \mathcal{X}_c^d} \left[ \left( -w_c^d(x_c) + b_c^d(x_c) \right) f_c^d(x_c) \right]$
      **end for**
    **end for**
  **end for**
```

5 Experiments and Discussion

5.1 Performance measures

Classifier performance. To estimate the performance of the part classifiers, we used several measures that are common in literature. In the following definitions, n indexes test instances and c denotes a particular class-label. The optimal probability for a test vector x is denoted by

$$p_c(x) = \begin{cases} 1, & \text{if } x \in \text{class } c \\ 0, & \text{otherwise} \end{cases}$$

and the estimated value given by the classifier is denoted by $\hat{p}_c(x)$.

- Precision recall curves (PR) and area under the curve (APR).
- Equal error rate (EER), i.e. the error rate at the point where the false positive rate is equal to the false negative rate, reported for each class individually or as mean over all classes.
- Mean cross entropy

$$\text{MCE} = -\frac{1}{N \cdot C} \sum_{n=1}^{N} \sum_{c=1}^{C} p_c(x_n) \ln \hat{p}_c(x_n)$$

The cross entropy is a measure of how good the estimated probabilities approximate the true probabilities. Cross entropy is defined as

$$\mathrm{CE} = -\sum_{n=1}^{N} p(x_n) \ln \hat{p}(x_n) = H(p) + D_{KL}(p \,\|\, \hat{p}) \; .$$

In this definition we set $0 \cdot \ln(0) = 0$ as $\lim_{x \longrightarrow 0} x \cdot \ln(x) = 0$. And as $p_c(x_n)$ is 0 or 1, depending on the true class c_n of x_n, MCE can further be simplified to

$$-\frac{1}{N \cdot C} \sum_{n=1}^{N} \ln \hat{p}_{c_n}(x_n)$$

In this case, this directly relates to the negative data log-likelihood $-\ln \hat{p}(x_1, \ldots, x_N, c_1, \ldots, c_n)$, and because $H(p) = 0$, also to $D_{KL}(p \,\|\, \hat{p})$.

- Confusion matrix gives the outcome of classification. Each row gives the instances of an actual class and each column the instances of the prediction. We normalize the confusion matrix by its row-sums, i.e. each row shows how many percent of instances of that respective class have been predicted as any of the classes. The diagonal of the confusion matrix gives the accuracy for each class. Cohen's κ [10] value is a summary measure for a confusion matrix and can be interpreted as the level of agreement between truth and prediction where 1 is total agreement.

Localization performance. To measure the performance of the whole framework with respect to localizing an object by its parts, we use the following measures.

For the Face and Human dataset we give

- Number of true positives (TP), i.e. ground truth is present and the estimated position is within the distance threshold (c.f. Figure 4.2).
- Number of outliers (OUT), i.e. ground truth is present and the estimated position is outside the distance threshold.
- Number of false negatives (FN), i.e. ground truth is present but the part has been labelled missing.
- Number of true negatives (TN), i.e. ground truth is occluded and part has been labelled missing.
- Number of false positives (FP), i.e. ground truth is occluded and part has been labelled present.
- 2D relative localization error for TP: The distance of the part, after MAP inference, to its ground truth location normalized by an instance-specific distance. Normalization for the Face dataset is with respect to the distance between the eyes, for the Human dataset it is the mean of the distance between the left-hip and left-sholder, and the distance between the right-hip and right-shoulder. We have chosen these normalization distances because they rarely suffer from foreshortening.

For the HumanEva dataset we give

- 2D localization error: The distance of the part, after MAP inference, to its ground truth location in pixel.
- 3D localization error: The distance in mm between the 3D ground truth location and the 3D location after triangulation using several synchronized 2D images.

The 3D localization error is also used for the Spine dataset.

5.2 Face

The model for the Face dataset consists of 5 parts: left and right eye, nose, left and right side of the mouth, see Figure 2. We used the Caltech face dataset [68] consisting of 450 frontal faces. Frames $[1 \ldots 89]$ (4 subjects, 3 male, 1 female, no beards, no glasses) constitute the training set; frames $[90 \ldots 165]$ (5 subjects, 4 male – two with beards, 1 female, no glasses) constitute the validation set; frames $[166 \ldots 450]$ (18 subjects, 10 male – one with beard – one with glasses, 8 female) constitute the test set, where frames $[328 \ldots 336]$ have been rescaled by factor 3 so that the faces appear approximately at the same scale as the rest and frames $[400, 402, 403]$ have been omitted from evaluation (artificial paintings) making a total of 282 test frames. The shape features are more susceptible to changes in scale than the appearance features, therefore for inference we compute the MAP over 5 discrete scale settings $([0.8, 0.9, 1.0, 1.1, 1.2])$ for the edge-length features.

Generic background was obtained from 45 images without people/faces, but featuring scenes where people normally occur.

Part Detection. Results from the part classifiers are summarized in Figure 8. Even though classification performance is already very good for the uncalibrated classifier (mean APR: 0.9771) it further increases after calibration (mean APR: 0.9989), while the mean equal error rate drops from 2.80% to 1.52%[7]. More important when used as component in a probabilistic model is the decrease in mean cross entropy, which is even more pronounced: from 4.88% to 1.16%.

[7] Note that changes in mean APR and mean EER are due to renormalization of the probabilities after calibration.

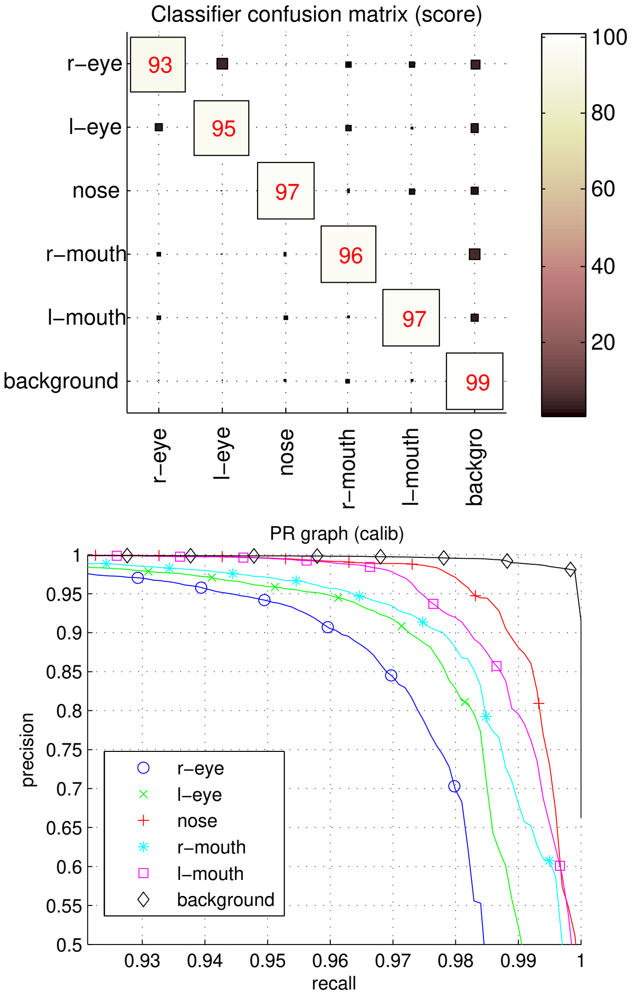


Fig. 8 Classification results for the Face dataset. The performance is shown for the calibrated combined appearance classifier (7) on the test vectors from the Caltech Face dataset. Top: confusion matrix (rows: true class, columns: estimated class). Bottom: (zoomed) precision recall graph. The classification performance is already extremely good for this dataset. Only very few misclassifications remain in the background class and between left and right eyes, which is also reflected in the precision recall graph. As a comparison, the diagonal of the confusion matrix for the uncalibrated classifier is 91.39, 91.75, 94.99, 86.52, 93.05, 99.97 (values in %). Respective κ values [10] are 94.5% for uncalibrated and 96.5% for calibrated.

Learning. We used the CRF learning algorithm [3] to compute optimal λ parameters on the training and calibration sets, using BP and TRBP for learning, and contrast them to the heuristic initialization (20). δ in eq. (22) was set to 0.5. We also compare a simpler tree graph in form of a star with the nose as center to the completely connected graph. To compare λ values for different learning methods we rescaled them so that their sum equals one (MAP inference will not be effected by this). This means for the heuristic (20) that the sum of all λ values for nodes is $\sum_{c \in V} \lambda_{\mathrm{c,a}} = \xi_{\mathrm{a,V}} = 0.5$ and the sum of all λ values for edges is analogously $\xi_{\mathrm{a,E}} + \xi_{\mathrm{l,E}} + \xi_{\mathrm{o,E}} = 0.5$. Interestingly for the CRF learning these values shifted towards nodes 0.83 for CRF-BP, and 0.61 for CRF-TRBP, and 0.93 for the tree model. The edge terms are effectively zero for the edge appearance and length features (CRF-BP: 0.01, 0; CRF-TRBP: 0.02, 0.01; tree: 0, 0), compared to orientation (CRF-BP: 0.16, CRF-TRBP: 0.35, tree: 0.06). The reason for this could be due to the hard thresholds for edge length employed during sampling (c.f. Sec. 2.6) false edges already have zero probability and especially the length term is thus uninformative for this dataset, whereas orientation could still separate between left and right for pairwise parts. Also we can see that more weight is allocated to the edge terms for the complete graph compared to the tree.

Localization. In Table 2 we show results for the localization performance on the test set for the different models. Baseline is given by ground truth, i.e. the user labelling and "baseline" which is the best localization possible given the reduced image information after candidate sampling (c.f. Sec. 2.6) by simply picking the nearest neighbor to the ground truth for each part in the set of candidates or the missing candidate if the ground truth is occluded. The models compared are summarized in Table 1, they are: the three complete graphs with the different learning methods (heuristic, CRF learning with BP, CRF learning with TRBP) (3)-(5); the tree graph with the heuristic initialization (taking into account the reduced number of edges in (20)) (6) and CRF learning using BP (7)[8]. And a decoupled graph without any structural edge information (8). Results are very close to optimality stemming from the good performance of the part-classifiers as indicated by the 93.5% of true positives obtained with the decoupled graph. Still more structure does improve the results: True positive rates increase for the tree models to 96.4% for the heuristic and 97.5% for the learned model, and to 96.7% (heuristic), 97.6% (CRF-BP), 98.3% (CRF-TRBP) for the complete graphs.

Overall there is no significant difference of the models given by the heuristic and the ones obtained by CRF learning on this dataset. Albeit the former tends to occlude more parts, it in turn produces less outliers. This seems a general trade-off and is largely effected by the user parameters γ for the heuristic (here set to 0.8) and the penalty δ given to the missing candidate in (22). Also the tree models show comparable performance owing to the rigid structure of the faces in the images. We find that this is a comparatively simple dataset and our

[8] Note that for trees the computation of $\mathbb{E}\left[\phi(x)_{c,x_c}\right]$ in (27) is exact using BP.

Table 1 Identification numbers (ID) used for the models employed for the Face and Human evaluation. Baseline is given in form of hand labelled ground truth, and "baseline" which uses the nearest candidate to ground truth after sampling part-candidates. Evaluated are three models with completely connected graphs, two models on a tree-shaped graph, and a decoupled graph.

ID	graph type	learning method
(1)	ground truth	
(2)	baseline	
(3)	complete	heuristic
(4)	complete	CRF-BP
(5)	complete	CRF-TRBP
(6)	tree	heuristic
(7)	tree	CRF-BP
(8)	decoupled	heuristic

Table 2 Localization performance for the Face test set. Three models using complete graph as underlying structure have been learned using the heuristic (3), and CRF with BP (4) and with TRBP(5). Simpler graphs in form of a star with nose as center ("tree") (6+7) and a completely decoupled graph (8), where all nodes are independent, are also presented as comparison, c.f. Tab. 1 for ID. Positive results are in green, errors in red. In bold is the best number for each column, without (1) and (2). For discussion see Sections 5.2 and 5.5.

ID	TP	OUT	FN	TN	FP	μ_d	σ_d
(1)	1409	0	0	1	0	0	0
(2)	1406	3	0	1	0	0.04	0.04
(3)	1362	**5**	42	**1**	**0**	**0.08**	**0.05**
(4)	1375	32	1	0	1	0.12	0.08
(5)	**1385**	22	2	0	1	0.12	0.07
(6)	1358	25	26	0	1	0.10	0.16
(7)	1374	30	5	0	1	0.12	0.08
(8)	1318	91	**0**	0	1	0.18	0.65

framework performs quite well. For comparable object-classes with simple structure the presented methods are well suited for detection.

Example configurations after MAP-inference are shown in Figure 9.

5.3 Human

As depicted in Figure 2, the model for the Human dataset consists of 13 parts or joints: head(1), shoulders(2), elbows(2), hands(2), hip(2), knees(2), and feet(2). We have used a total of 2401 images consisting of images from private collections, images taken from the Internet and images of the PASCAL Visual Object Class (VOC) Challenge 2006 [19] and 2007 [18]. For the non-object class, we used the same background images as for the Face dataset. A total of 1243 images was used as training set to learn the feature functions for appearance and geometry, 717 images were used as validation set for the classifier calibration, and 441 images remained as test set. The test images were taken from the PASCAL VOC Challenge 2007 for the Person Layout task.

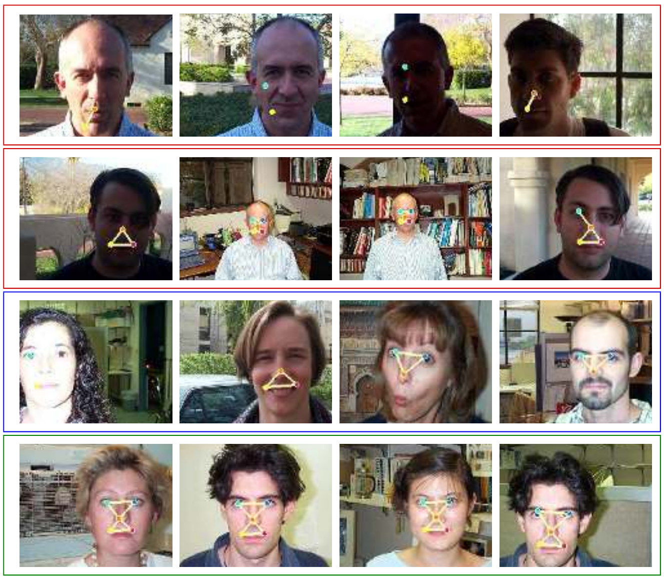


Fig. 9 Images with face configurations (slightly cropped). Rows 1 and 2 (red): 8 worst configurations with respect to the mean distance to ground truth. 3rd row (blue) configurations with least parts detected not contained in rows 1 and 2. Last row (green) 4 configurations with highest confidence, i.e. the exponential of the negative energy or unnormalized probability. We throughout obtain good performance on this data as only the worst 3 images can be considered wrong configurations.

Part Detection. Confusion matrix and precision recall graph for the calibrated classifiers are shown in Figure 10. As a comparison, the diagonal of the confusion matrix (in % for uncalibrated ("score") and calibrated ("calib") classifiers is

	head	l-shoul	r-shoul	l-elbow	l-hand	r-elbow	r-hand
score	87.01	64.57	73.24	18.51	16.47	23.76	19.49
calib	81.04	56.72	67.76	24.36	26.47	26.77	28.53

	l-hip	r-hip	l-knee	l-foot	r-knee	r-foot	backgro
score	37.33	37.31	29.63	36.09	33.23	34.18	89.01
calib	31.41	31.23	32.56	38.76	30.75	37.14	91.78

Respective means, i.e. mean accuracy, are 42.84% and 43.23%. Cohen's κ values are 44.50% and 44.93% respectively. Mean APR increases from 42.76% to 45.18% after calibration. Mean EER drops from 22.09 to 20.77. As stand-alone classifier there doesn't seem to be significant improvement, the classes just seem slightly more balanced. MCE drops more significantly from 14.9% to 11.2%, but is still much higher than for the Face dataset (1.16%). There is quite distinctive behaviour for different body parts for this difficult dataset. Especially hands and elbows, whose appearance vary significantly due to articulations in the images, give poor results.

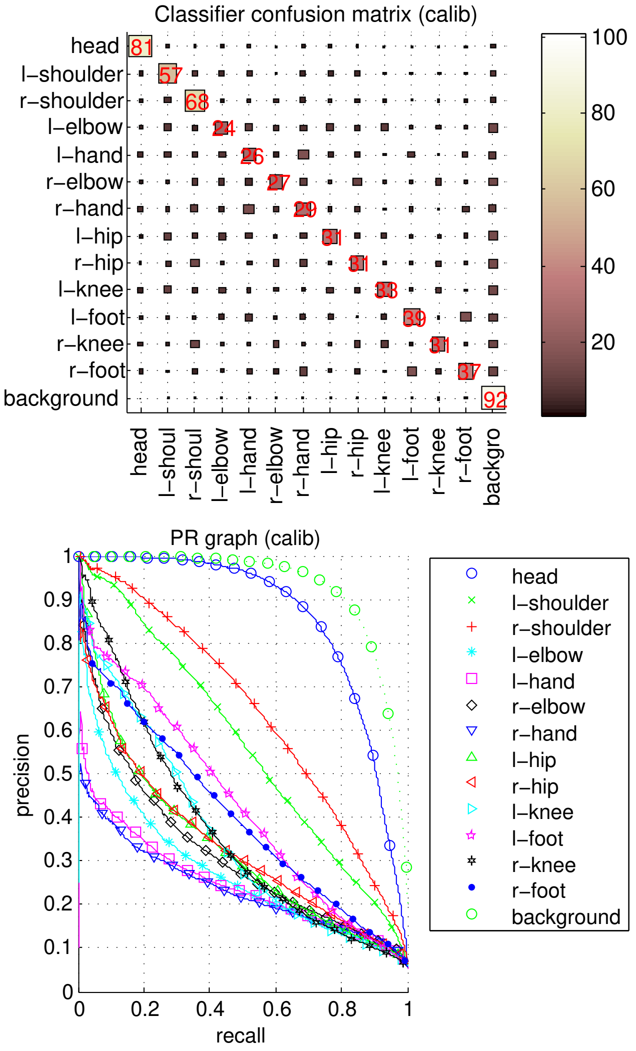


Fig. 10 Classification results for the Human dataset after calibration. Top: confusion matrix (rows: true class, columns: estimated class). Bottom: precision recall graph. Especially hands and elbows are difficult to recognize, followed by hip and knees. In contrast the performance for head, shoulders and background is much superior. As can be seen from the confusion matrix, this is also true because of ambiguities between left and right body parts.

Learning. We used the CRF learning algorithm [3] to compute optimal λ parameters on the calibration set, using BP for learning on the complete graph (CRF-BP) and on a simpler tree graph[9]. δ in eq. (22) was set to 0.01. The normalized λ values for the heuristic again sum to 0.5 for nodes: $\sum_{c \in V} \lambda_{\mathrm{c,a}} = \xi_{\mathrm{a,V}} = 0.5$ as well as for edges: $\xi_{\mathrm{a,E}} + \xi_{\mathrm{l,E}} + \xi_{\mathrm{o,E}} = 0.25 + 0.125 + 0.125 = 0.5$. The corresponding values for the complete model (CRF-BP) are: 0.1818 for nodes and 0.8182 for edges (appearance: 0.0895, length: 0.2164, orientation: 0.5124). For the tree they are: 0.4247 for nodes and 0.5753 for edges (appearance: 0.0623, length: 0.2194, orientation: 0.2936). Contrary to the Face dataset, here the weights actually give more influence to the edge terms and less influence to the node terms. This is much more so for the complete graph than for the tree. It might indicate that shape is actually the more informative cue for this class as the part appearance performs so poorly for this difficult data.

Table 3 Localization performance for the Human test set. Compared are the complete graph learned heuristically (3) and with CRF-BP (4), the tree graph with heuristic (6) and learned (7), and the decoupled graph (8), c.f. Tab. 1 for ID. Positive results are in green, errors in red. In bold is the best number for each column, without (1) and (2). For discussion see Sections 5.3 and 5.5.

ID	TP	OUT	FN	TN	FP	μ_d	σ_d
(1)	2381	0	0	791	0	0	0
(2)	1494	887	0	791	0	0.27	0.36
(3)	628	940	813	526	265	0.55	0.58
(4)	**734**	**868**	779	**541**	**250**	**0.41**	**0.45**
(6)	581	1196	604	298	493	0.77	0.74
(7)	709	998	674	456	335	0.54	0.62
(8)	592	1762	**27**	90	701	0.94	0.79

Localization To simplify evaluation, we only consider images containing only one person (244 frames). For input, images are rescaled using the method in Sec. 2.4. The shape term was optimized over 5 discrete scale settings ([0.9, 0.95, 1.0, 1.05, 1.1]). The results of the localization are summarized in Table 3. And Figure 11 gives an overview of the number of correct parts per image, i.e. the number of true positives (TP) plus the number of true negatives (TN). Clearly using CRF learning (Sec. 4.2) on the complete graph produces the strongest model for this dataset.

For this best performing model (CRF-BP), Table 4 shows the localization results for each part individually. The rigid upper body parts head and shoulders are detected rather well, with moderate levels of outliers. Next come hip, followed by elbows. Most outliers occur with the hands that are the most difficult parts to detect, both with respect to appearance and shape due to articulation. Many of the images only contain partial configurations, especially images where only the upper body is visible and the rest is occluded. Our algorithm can handle these cases, indicated by the high levels of true negatives for the knees and feet.

Example configurations obtained after MAP-inference for the learned model are shown in Figure 12.

[9] Due to the computational burden, we omitted a comparison to CRF-TRBP

Table 4 Localization performance for 244 Human test images, shown for each part individually for the model learned with CRF-BP. Overall performance for head and shoulders is quite good, whereas the articulating parts, especially hands, are much harder to detect. This could be expected given the performance of the part classifiers and also the shape features are less discriminative for these parts.

	head	l-shoul	r-shoul	l-elbow	l-hand	r-elbow	r-hand	l-hip	r-hip	l-knee	l-foot	r-knee	r-foot	total
TP	160.00	94.00	125.00	41.00	31.00	47.00	31.00	58.00	56.00	26.00	22.00	24.00	19.00	734.00
OUT	55.00	51.00	73.00	76.00	94.00	92.00	100.00	79.00	74.00	42.00	26.00	58.00	48.00	868.00
FN	29.00	91.00	40.00	79.00	81.00	58.00	73.00	50.00	63.00	62.00	54.00	52.00	47.00	779.00
TN	0.00	6.00	2.00	25.00	16.00	18.00	16.00	37.00	28.00	87.00	116.00	87.00	103.00	541.00
FP	0.00	2.00	4.00	23.00	22.00	29.00	24.00	20.00	23.00	27.00	26.00	23.00	27.00	250.00

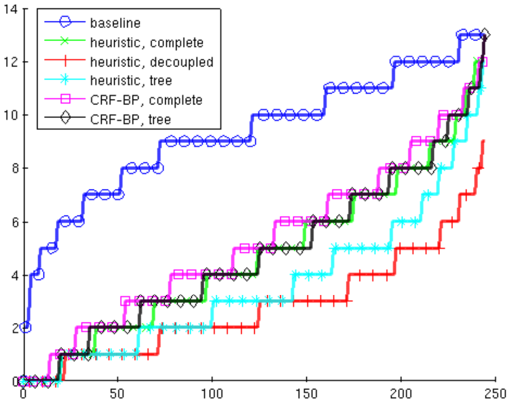


Fig. 11 Number of correct parts (TP+TN) per frame on 244 Human test images. Note, the integral under each curve corresponds to the total number of correct parts. For discussion see Sections 5.3 and 5.5.

5.4 Inference performance

Throughout the experiments, we checked the performance of A^*, BP and TRBP for the Face and the Human datasets. Due to the strong geometrical constraints, graphical inference for face detection is comparatively easy. All three methods show good performance and suboptimality to this small degree does not seem to be an issue. Somewhat unexpectedly, in addition to being globally optimal by definition, A^* is here also the fastest method!

For the complex Human problem class, the median runtime of A^* with 0.9680 (heuristic model) 0.9890 (CRF-BP) seconds is still reasonably fast. In a few, unpredictable cases it might however take several minutes.

Regarding optimization performance, we obtain an ambivalent rating between the methods. For the heuristic models it seems that BP performs worst and we get

$$A^* \geq TRBP \geq BP \ ,$$

whereas the models learned with the CRF framework are better solved with BP, thus

$$A^* \geq BP \geq TRBP \ ,$$

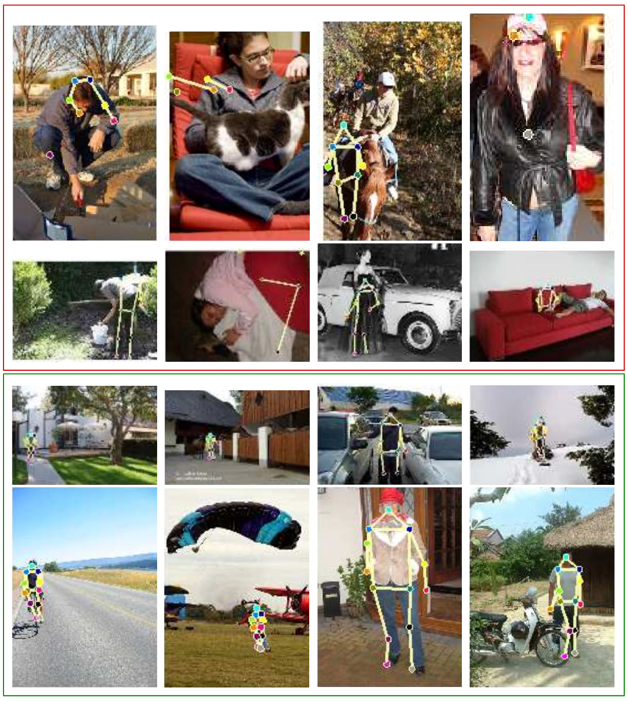
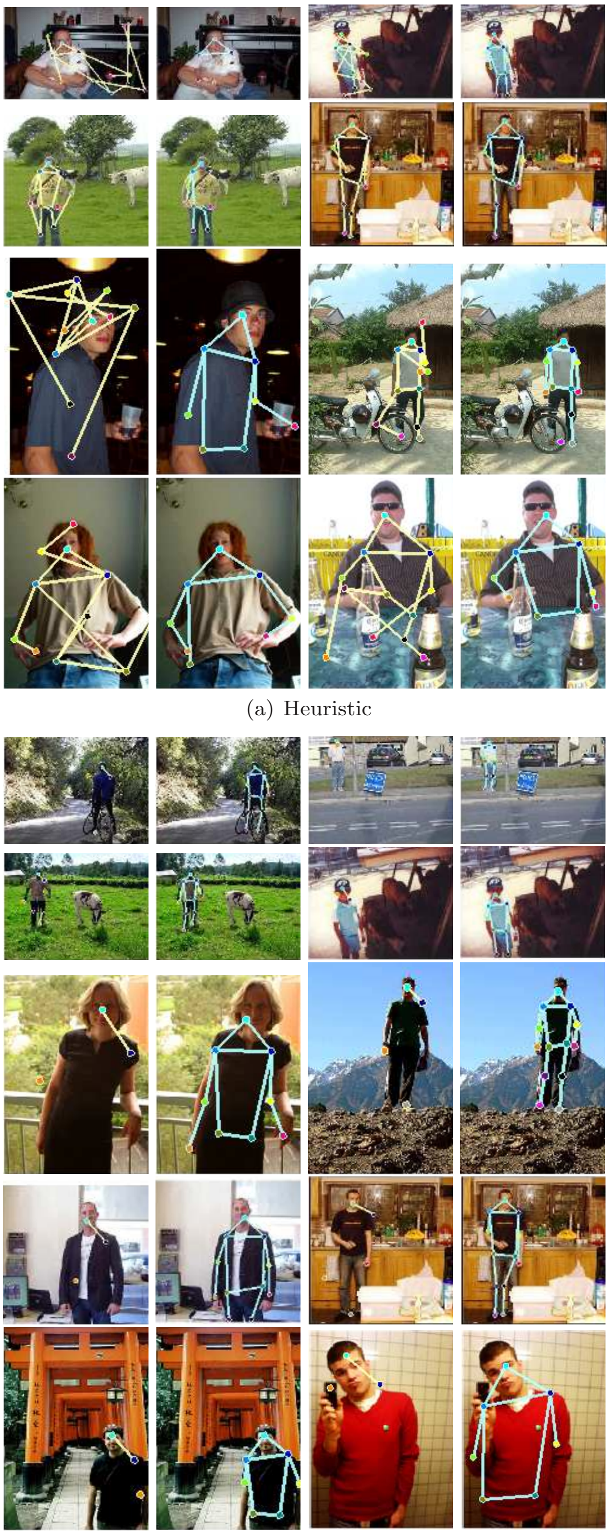


Fig. 12 Human configurations for the PASCAL VOC Challenge 2007 for the Person Layout task. Top 2 rows (red): configurations that were too difficult for our approach. Bottom 2 rows (green): 8 Configurations with high confidence. In general, standing humans without too much occlusion can be localized well.

as is quantitatively shown in Table 5. For the complex human class, suboptimal inference *is* an issue, in particular for TRBP! Figure 13 illustrates this point by a range of typical examples, where the problems seem to arise in particular in dealing with occluded/missing parts: Suboptimal configurations are due to occluding too few parts for the heuristic model and too many parts for the learned model.

5.5 Comparison to Tree Graphs

As mentioned in previous sections, we compared the fully connected graphical models also to simpler tree structured models. Quantitative results are presented in Tables 2 and 3 and also in figure 11. Decline in lo-



(a) Heuristic

(b) CRF-BP

Fig. 13 Images where BP and TRBP do not converge to the global optimum. The first image of each pair shows the result of BP or TRBP (yellow graph), the second the global optimum computed with A^* (cyan graph). When suboptimality occurs, BP and TRBP have the same problem with handling occlusion: Depending on the underlying model they produce too many (CRF-BP) or not enough (heuristic) occluded parts. Images shown are typical in this respect.

Table 5 Mean values of inference time (in seconds) and confidence (exp of the neg. energy) relative to the optimum given by A^* for inference on the same graphs. For BP and TRBP, these values correspond to 1000 iterations. The small differences in confidence values of all three methods for faces indicate that this class is comparatively easy to handle. Furthermore, for graphs with a small number of vertices, A^* outperforms all methods also with respect to runtime! For more complex classes like "Human", the differences between A^* and the approximate inference engines is much more pronounced.

Dataset		A^*	BP	TRBP
Face	time	0.0377	0.2224	0.1463
heuristic	confidence	1.0000	0.9801	0.9915
Face	time	0.0345	0.1847	0.2488
CRF-BP	confidence	1.000	0.9990	0.9975
Face	time	0.0413	0.1916	0.2108
CRF-TRBP	confidence	1.0000	0.9960	0.9949
Human	time	4.7552	1.8276	0.8511
heuristic	confidence	1.0000	0.5442	0.6524
Human	time	5.6601	0.6552	0.3080
CRF-BP	confidence	1.0000	0.9983	0.9092

calization performance is mainly due to our particular handling of occlusion/missing parts that is quite different to other methods: For a tree structured graph if one of the parts is missing the inferred configuration is the result of two or more independent subgraphs. For the Face set with strong part classifiers this usually produces only marginally worse results, as most parts are found and there is few occlusion. A few configurations where missing parts where inferred, however, illustrate this difficulty for tree-structured graphs, see Fig. 14. For the Human dataset where the framework has to deal with highly elevated levels of occlusion this can cause for example legs to appear inside the body, see Fig. 15. This shows that for our framework dens graphs with additional structural information are a necessity.

The confusion between left and right body parts that are reported by other using tree models (e.g. [21]), and that usually require sampling from the tree-distribution and evaluation of another global cost-function, is immaterial in our framework, due to our particular shape features including absolute angles. This is, however, only true if articulating parts do not cross as opposed to, e.g. folded arms.

5.6 HumanEva

The HumanEva dataset [60] consists of several sequences of four subjects (S1, . . . , S4) performing five different tasks (walking, jogging, gesturing, boxing and throwing/catching). The sequences for HumanEvaI were taken with 7 synchronized cameras (3 color, 4 gray-scale), for HumanEvaII there are 4 synchronized color cameras.

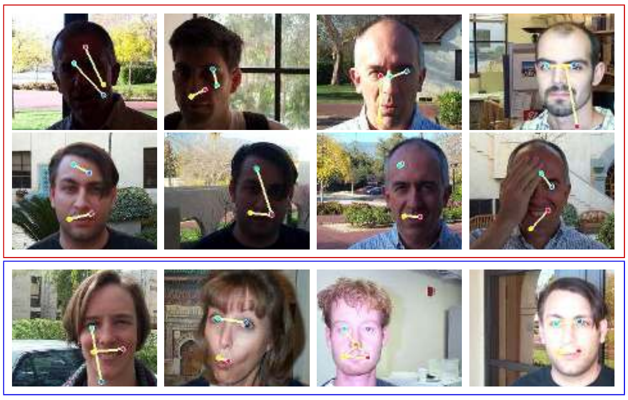
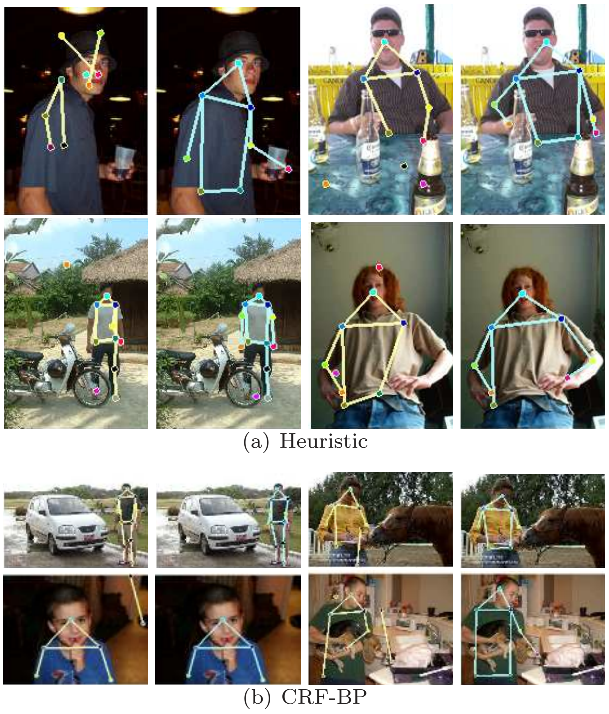


Fig. 14 Images with face configurations (slightly cropped) for the tree graph. Rows 1 and 2 (red): 8 worst configurations with respect to the mean distance to ground truth. 3rd row (blue) configurations with least parts detected not contained in rows 1 and 2. The 8 worst configurations and the first of the last line can be considered wrong. Compare this to only 2 wrong configurations in Fig. 9. Problems occur for this dataset in cases when one or more parts are missing as there is then no structural information to other parts. This is especially severe here when the nose is missing as it is the center of the star-shaped tree. Note that drawn edges do not correspond to the underlying tree-graph but are the same as in Fig. 9.



(a) Heuristic

(b) CRF-BP

Fig. 15 Comparison of the tree graphs to their respective fully connected models. Left images are inferred results using the tree, right images using the complete graphs. If parts are missing the tree-graphs get disconnected and independent subgraphs are matched, leading e.g. to spurious legs inside the body: (a) 1st row, (b) all except first pair. Also if a body part is not at its usual location relative to other parts this can cause confusion between symmetric parts (folded arms, (b) 2nd row, right image.

The sequences are separated into "Training", "Validation" and "Test" sets. The respective number of images in each set are given in [60]. The 2D labelling of parts (or joints) is similar to our labelling of the Human dataset, but here they do not correspond to visual features, but to the back-projection of their 3D counterparts, see Figure 2 for comparison. The 15 parts are shown in Figure 1. The "Training" images have been used to train the local appearance classifiers, as well as the geometry terms. The "Validation" set has been used for classifier calibration.In Figure 16 the confusion matrix and the precision recall curves are shown for the "Validation" set, as ground truth for the "Test" set is not available to us.

Classifier performance. Overall the classifier performance is much better than for the Human dataset as more training data is available and the test set is more similar to the training set. Mean APR increases from 92.87% to 95.43 for calibrated data, mean EER drops from 3.12% to 2.35%, Cohen's κ increases slightly from 89.78% to 90.35%, and MCE decreases significantly from 5.00% to 1.53%, but note that the test set in this case is the same as used for calibration.

Due to the complex graphs, no CRF learning has been performed on this dataset. The presented results where obtained using only the initialization heuristic in Sec. 4.1.

Triangulation. The HumanEva dataset is the only dataset with 3rd-party ground truth data available, which allows objective measurements of the geometrical error. To obtain a 3D configuration for corresponding synchronized 2D images, we triangulate a 3D configuration as follows.

1. For all image pairs (I_i, I_j), $i, j \in \{1, \dots, \#\text{cameras}\}$, $i < j$, of a synchronized frame, and for all parts $s \in V$ that are not occluded in either image, triangulate a 3D point $X_{s,ij}$ using standard stereo-triangulation.
2. Calculate the vector-median [69] to find the median 3D position, i.e. the 3D point X_s for part s that minimizes the sum of Euclidean distances to all candidate 3D points,

$$X_s = \min_X \sum_{ij} \|X - X_{s,ij}\| \,. \tag{28}$$

The vector-median has some beneficial properties: the median X_s lies in the convex hull spanned by $\{X_{s,ij}\}_{(ij)}$ and, like the scalar-median, is robust against outliers [69]. Additionally, it is rotationally invariant. For theoretical background and implementation details we refer to [3].

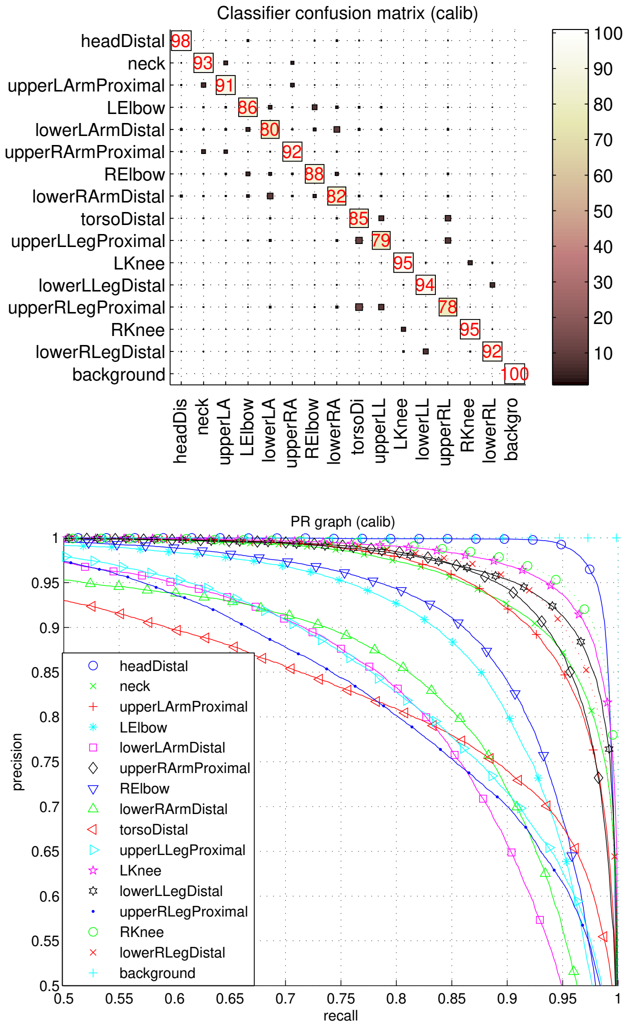


Fig. 16 Classification results for the HumanEva dataset after calibration. Top: confusion matrix, bottom: PR curve (zoomed). The confusion matrix is here exemplary for the ambiguities between left and right body parts.

Localization performance. In Table 6 we summarize the results of the localization performance for the HumanEvaI and HumanEvaII dataset, by comparing two approaches: (1) inferring configurations for each image of a frame individually, and (2) using the additional epipolar constraints and inferring a configuration for all images of a frame at once (indicated by the extension "e" in the table). For the second method we used BP for approximate MAP-inference as these graphs have $7 \times 15 = 105$ nodes and inference using A^* is no longer feasible. Localization errors are reported for 3D and for 2D. For the shape terms, global scale r was inferred over 5 discrete scale settings ([0.80.91.01.11.2]). For 2D we also report the resulting error after back-projection of the inferred 3D locations which yields large improvements for the method without epipolar features, but almost no change in error when including epipolar features from the start, thus indicating that in deed the inclusion of epipolar features leads to more consistent 2D configurations. Sample configurations for different tasks after MAP inference are shown in Figure 17. Our results indicate that if the training and test data come from similar distributions, as is the case for the HumanEvaI dataset, then our method works very well in almost all cases. Also when the test set changes, e.g. when the subjects have different clothing as is the case for HumanEvaII, but is still similar to the training, we can achieve competitive results with our method *without using background subtraction, temporal context or 3D kinematics.* Thus our method can be used in contexts where the camera is not fixed and for (re-)initialization of tracking algorithms.

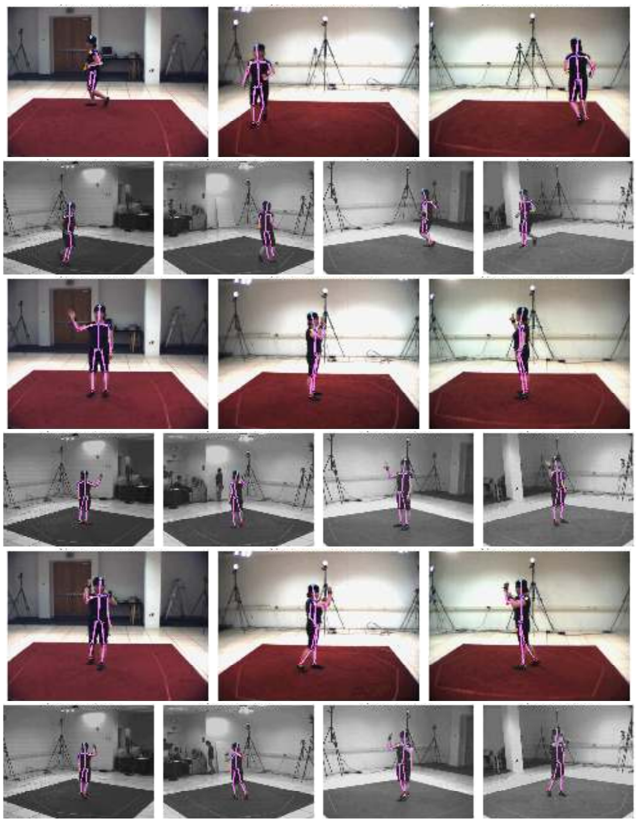


Fig. 17 Configurations for the HumanEva test set using MAP inference on the graphs including epipolar constraints. 3 complete frames are shown, i.e. all seven images of subject S1 performing the tasks jogging (rows 1-2), gestures (rows 3-4) and box (rows 5-6). We depict typical errors that can occur for some frames: 1st set: one foot is consistently matched to a wrong location (near the other foot). 3rd set: one arm is also matched consistently to a location near the hip, where it is often found during training and which leads to false detections in this case.

Table 6 Localization results for the HumanEvaI (I:) and HumanEvaII (II:) test sets (Combo).
Settings: We took every 20th frame of the test set for each subject S2 to S4 for HumanEvaI and S2 and S4 for HumanEvaII. For subject S1 the gray-scale images were not available for action "Cobmo". We chose the "Cobmo" set as it includes all types of activities. We show the median and the mean error. 3D error is in millimeter, 2D errors are in pixel. If a part is labeled as occluded/missing it is not taken into account in the error-measure. The extension "e" indicates the model including the epipolar constraints. We report mean error (μ), standard deviation (σ), median (0.5) and 90 percent quantile (0.9). The bold values indicate the best 3D and 2D error for each subject.
Discussion: 2D error improves a lot with back projection when using the method without epipolar features. When epipolar features are included then the back projection method does not seem to change much the 2D errors (except for I:S4 where it apparently produces one or more outliers – see mean and standard deviation for this case). This could indicate that the additional *epipolar features already enforce consistent results over the individual images* so that the back projection can not really improve the 2D error. 3D errors do not present significant differences neither do 2D errors after back projection or with epipolar features. Overall the resulting errors for HumanEvaII are a little higher than the ones reported by others [34,9], *but without using background subtraction, temporal context or 3D kinematics.* Thus our method is especially attractive for (re-)initialization.

Data	3D				2D				3D→2D			
	μ	σ	0.5	0.9	μ	σ	0.5	0.9	μ	σ	0.5	0.9
I:S2	**39.49**	**3.28**	**39.41**	**43.28**	6.68	3.75	5.80	8.01	**4.82**	1.52	**4.64**	7.71
I:S3	95.33	48.30	84.21	183.36	15.15	9.63	12.97	29.37	11.74	**6.57**	10.00	**21.09**
I:S4	**197.96**	**61.77**	**192.32**	**292.84**	28.77	12.18	26.36	44.95	**24.58**	**9.74**	**22.60**	**36.61**
II:S2	207.48	90.85	**185.55**	338.71	29.75	15.51	26.16	50.66	**25.48**	13.18	**22.65**	**42.92**
II:S4	292.17	103.46	**283.49**	419.35	46.63	24.32	41.66	77.21	**38.82**	16.32	**36.30**	59.35

Data	3De				2De				3De→2D			
	μ	σ	0.5	0.9	μ	σ	0.5	0.9	μ	σ	0.5	0.9
I:S2	41.26	5.69	42.90	46.19	5.76	1.54	5.45	8.28	5.20	**1.42**	4.83	**7.67**
I:S3	**92.04**	**47.89**	**76.61**	**180.33**	12.17	6.99	9.93	21.91	**11.59**	6.73	**9.36**	21.19
I:S4	261.15	381.54	202.77	310.91	25.66	11.19	23.68	39.50	52.83	373.37	24.74	40.58
II:S2	211.40	**81.24**	200.51	**335.77**	27.76	12.65	25.42	46.47	27.13	**12.11**	25.18	45.32
II:S4	**290.98**	**78.56**	289.96	**397.70**	40.04	13.77	37.64	56.94	39.20	**13.31**	37.16	**55.24**

5.7 Spine Labeling in 3D Magnet Resonance Images

Our approach is not limited to face or person detection in 2D images. A related field that we investigate is the detection and labeling of anatomical structures in medical images. In this context, we experiment with magnetic resonance images of the human spine column, in which we automatically localize and identify the intervertebral discs using the parts-based model described in this paper. Applications include labeled visualization, initialization of segmentation algorithm and statistical shape analysis for deformation-related pathologies (e.g. scoliosis). The 3D images are low-resolution ($224 \times 224 \times 180 \approx 9 \cdot 10^6$ voxels) T1-weighted fast field-echo images of the total spine. The fact that the sought discs have ambiguous local appearance, or, due to pathologies, might be degenerated or missing completely, is particularly challenging. Therefore, exploiting global context and permitting missing parts in the model are essential for successful labeling.

We used a simplified version of the described model for these data. Because of the limited training set of 30 3D-images, we modeled geometric features, i.e. pairwise part distances or pairwise displacements, as truncated Gaussians. The model contains 26 vertices corresponding to the center positions of the intervertebral structures, 23 of them being discs in the anatomical sense lying between mobile vertebrae. We trained an ensemble of 150 randomized trees without calibration on a set of volume patches around the ground truth locations as image features, augmented by resampled, deformed copies, and background.

Using a leave-one-out procedure for evaluation, the model was fit to the test image by generating the 10 candidates for each part having the strongest responses in the tree classifier, and inferring the optimal configuration using A^*. The global scale parameter r was here estimated based on the first fit and used to refine the geometry prior, leading to a better subsequent fit by compensating for patient height. Finally, for parts marked as missing, we predict a position relative to the ones found in a postprocessing step. Figure 18 shows an example result of the procedure, illustrating localization and labeling of the model's parts in an unseen image.

We achieved an average part detection rate above 90% and an average part distance to ground truth of 5.1mm. Details are reported in [57].

The computational bottleneck of the detector in this application is not the inference stage, as one might expect, but rather the application of the tree classifier on each of the $\sim 9{\cdot}10^6$ voxels. Cascaded classifiers and parallelization might be viable approaches to resolve this issue.

6 Conclusions and Outlook

This paper presented a general approach on part-based object detection using graphical models. Beside appli-

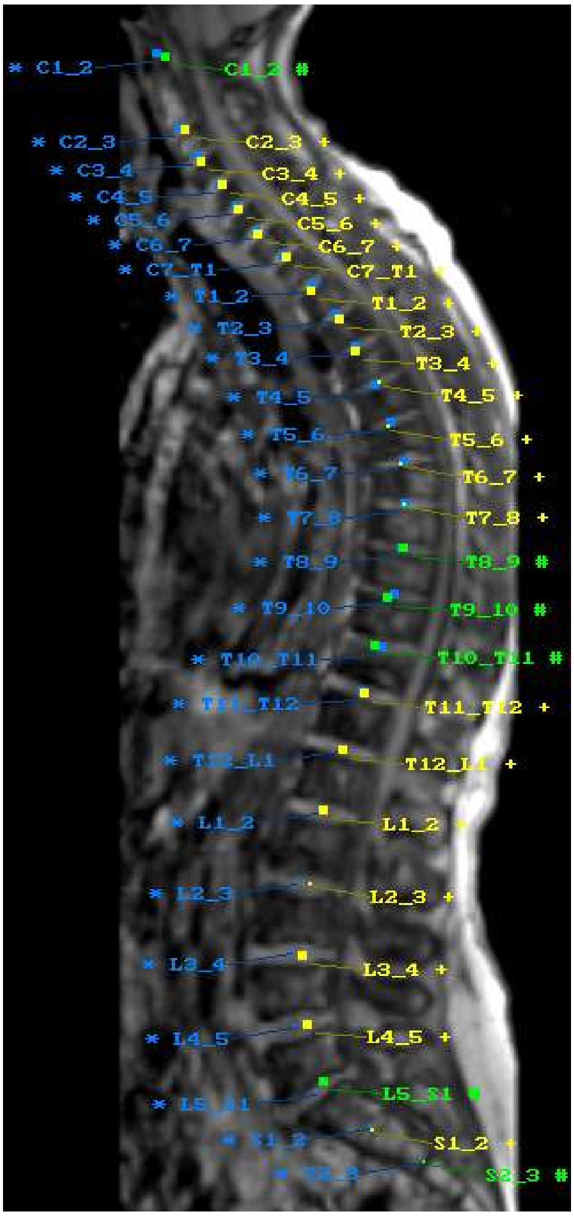


Fig. 18 MR spine labeling result. Yellow labels show the MAP estimate(+), green labels represent parts inferred by postprocessing(#), and blue labels show the ground truth annotation(*), done independently. Larger dots indicate positions more closely located to the viewing plane. We achieved an average part detection rate above 90% and an average part distance to ground truth of 5.1mm. Details are reported in [57].

cability to a range of object classes, a key objective of this work concerns the modelling of *articulated objects* where parts do not have a fixed relative geometrical position. Hence, a major issue in connection with object detection is the large variability of their appearance, especially for the Human data. We have tackled this by using *discriminative* classifiers and including them in a probabilistic CRF framework.

We recapitulate important aspects and differences of our approach compared to related work, especially in the context of human detection:

– A configuration is completely defined by the location of the parts, which in the case of humans are the joints as opposed to parametrization of the limbs. We feel that this non-redundant parametrization is beneficial as the search space for the parameters is smaller.
– No special form of the input features is assumed. Hence, the presented framework can easily be expanded using more features. The inclusion is straightforward if the features only depend on the configuration of two parts. While having more feature functions certainly makes CRF-learning more difficult, *there is no impact on the MAP-inference*, because feature functions are combined to a single potential for each node and edge before inference. The extension to features using triples or more parts is also conceivable. However, the computational complexity increases rapidly with the number of parts.
– Using feature calibration (Section 2.3) together with the heuristic for initializing the λ-parameters (20)), it is not even necessary to re-learn the model.
– The bottom-up process uses information at early stages to keep the image graphs small for inference, thus allowing for the computation of structural information in form of pairwise features without the suffering from the quadratic complexity that is usually involved.
– For fully connected model-graphs up to not more than 15 vertices, *exact* MAP-inference can be computed in the order of seconds on standard PC using A^*-search and our novel admissible heuristic presented in Section 3.2. For larger model-graphs efficient iterative algorithms using variants of loopy belief propagation provide often equitable approximate solutions. But suboptimality may also degrade detection quality (cf. Fig. 13).

Future Work. Using completely connected graphs for model representation certainly introduces redundancies that give the opportunity for averaging out the "noise" of individual features when inferring difficult object configurations and allows for efficient handling of occlusion (c.f. Sec. 2.6). In some cases, however, we would like to increase detection speed by removing redundant computations in the detection phase. In future work we would like to apply variations of the CRF learning algorithm 3 by the inclusion of additional prior terms on the λ-parameters that favor sparse solutions and also penalize costly computations. With respect to A^* and our proposed heuristic, one opportunity to improve speed that we have not yet investigated is optimizing the underlying tree for the heuristic. The trees we have used correspond roughly to a natural ordering of the parts as, e.g. given by the kinematic chain for the Human dataset. In general, other trees might be more efficient using the most informative parts first. Another aspect not addressed in this work is the inference of multiple objects. In fact, for the Human dataset we used a simple greedy method of finding one object at a time and removing its bounding box for further inference. A systematic analysis of how to model multiple objects efficiently in a single framework is part of our ongoing research.

Acknowledgements This work has been partially supported by the European Marie Curie research training network VISIONTRAIN and by Philips Research Europe, Hamburg. We also thank the anonymous reviewers for their detailed and constructiv criticism that helped to improve the original version of the manuscript.

References

1. Agarwal, A., Triggs, B.: Recovering 3D human pose from monocular images. IEEE PAMI **28**(1), 44–58 (2006)
2. Balan, A., Black, M., Haussecker, H., Sigal, L.: Shining a light on human pose: On shadows, shading and the estimation of pose and shape. In: Proc. ICCV (2007)
3. Becker, F.: Matrix-valued filters as convex programs. Master's thesis, CVGPR group, University of Mannheim (2004)
4. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Trans. Patt. Anal. Mach. Intell. **24**(4), 509–522 (2002)
5. Bennett, K., Parrado-Hernández, E.: The interplay of optimization and machine learning research. J. Mach. Learning Res. **7**, 1265–1281 (2006)
6. Bergtholdt, M., Kappes, J.H., Schnörr, C.: Graphical knowledge representation for human detection. In: International Workshop on The Representation and Use of Prior Knowledge in Vision. Springer (2006)
7. Bergtholdt, M., Kappes, J.H., Schnörr, C.: Learning of graphical models and efficient inference for object class recognition. In: 28th Annual Symposium of the German Association for Pattern Recognition (2006)
8. Bray, M., Kohli, P., Torr, P.: Posecut: Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In: Proc. ECCV, pp. 642–655 (2006)
9. Cheng, S.Y., Trivedi, M.M.: Articulated human body pose inference from voxel data using a kinematically constrained gaussian mixture model (2006). In CVPR EHuM2: 2-nd Workshop on Evaluation of Articulated Human Motion and Pose Estimation
10. Cohen, J.: A coefficient of agreement for nominal scales. Educational and Psychological Measurement **20**, 37–46 (1960)
11. Coughlan, J., Shen, H.: Shape matching with belief propagation: Using dynamic quantization to accomodate occlusion and clutter. In: CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 12, p. 180. IEEE Computer Society, Washington, DC, USA (2004)
12. Coughlan, J., Yuille, A.: Bayesian A^* tree search with expected O(N) node expansions: applications to road tracking. Neural Computation **14**(8), 1929–1958 (2002)
13. Coughlan, J.M., Ferreira, S.J.: Finding deformable shapes using loopy belief propagation. In: ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part III, pp. 453–468. Springer-Verlag, London, UK (2002)
14. Cowell, R., Dawid, A., Lauritzen, S., Spiegelhalter, D.: Probabilistic Networks and Expert Systems. Springer (2003)
15. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: ECCV Int. Workshop on Stat. Learn. in Comp. Vis. (2004)
16. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR, pp. 886–893 (2005)
17. DeGroot, M.H., Fienberg, S.E.: The comparison and evaluation of forecasters. Statistician **32**(1), 12–22 (1982)
18. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html
19. Everingham, M., Zisserman, A., Williams, C.K.I., Van Gool, L.: The PASCAL Visual Object Classes Challenge 2006 (VOC2006) Results. http://www.pascal-network.org/challenges/VOC/voc2006/results.pdf
20. Fawcett, T.: ROC graphs: Notes and practical considerations for researchers (2004)
21. Felzenszwalb, P., Huttenlocher, D.: Pictorial structures for object recognition. IJCV **61**(1), 55–79 (2005)
22. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 264–271 (2003)
23. Fergus, R., Perona, P., Zisserman, A.: A sparse object category model for efficient learning and exhaustive recognition. In: CVPR (2005)
24. Fergus, R., Perona, P., Zisserman, A.: Weakly supervised scale-invariant learning of models for visual recognition. Int. J. Comp. Vision **71**(3), 273–303 (2007)
25. Fergus, R., Weber, M., Perona, P.: Efficient methods for object recognition using the constellation model. Tech. rep., California Institute of Technology (2001)
26. Ferrari, V., Marin-Jimenez, M., Zisserman, A.: Progressive search space reduction for human pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2008)
27. Frey, B., Jojic, N.: A comparison of algorithms for inference and learning in probabilistic graphical models. IEEE Trans. Patt. Anal. Mach. Intell. **27**(9), 1392–1416 (2005)
28. Gavrila, D.: A bayesian, exemplar-based approach to hierarchical shape matching. IEEE Trans. Patt. Anal. Mach. Intell. **29**(8), 1408 – 1421 (2007)
29. Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. Machine Learning **36**(1), 3–42 (2006)
30. Gupta, A., Mittal, A., Davis, L.S.: Constraint integration for efficient multiview pose estimation with self-occlusions. IEEE Trans. Pattern Anal. Mach. Intell. **30**(3), 493–506 (2008)
31. Hart, P., Nilsson, N., Raphael, B.: A formal basis for the heuristic determination of minimum cost paths. IEEE Tr. Syst. Sci. Cybernetics **4**, 100–107 (1968)
32. Hartley, R.I.: Estimation of relative camera positions for uncalibrated cameras. In: (LNCS), vol. 588, pp. 589–587. ECCV, Springer-Verlag (1992)
33. Hinton, G.E.: Training products of experts by minimizing contrastive divergence. Neural Comput. **14**(8), 1771–1800 (2002)
34. Howe, N.R.: Recognition-based motion capture and the humaneva ii test data (2007). In CVPR EHuM2: 2-nd Workshop on Evaluation of Articulated Human Motion and Pose Estimation

35. Jiang, H., Martin, D.R.: Global pose estimation using non-tree models. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2008)
36. Kolmogorov, V.: Convergent tree-reweighted message passing for energy minimization. IEEE Trans. Patt. Anal. Mach. Intell. **28**(10), 1568–1583 (2006)
37. Kolmogorov, V., Rother, C.: Comparison of energy minimization algorithms for highly connected graphs. In: Proc. ECCV (2006)
38. Komodakis, N., Tziritas, G.: Approximate labeling via graph cuts based on linear programming. PAMI **29**(8), 2649–2661 (2007)
39. Kumar, S., Hebert, M.: Discriminative random fields. Int. J. Comp. Vision **68**(2), 179–201 (2006)
40. Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: International Conference on Machine Learning (ICML) (2001)
41. Lee, M.W., Cohen, I.: A model-based approach for estimating human 3D poses in static images. IEEE PAMI **28**(6), 905–916 (2006)
42. Lepetit, V., Fua, P.: Keypoint recognition using randomized trees. PAMI **28**(9), 1465–1479 (2006)
43. Levin, A., Weiss, Y.: Learning to combine bottom-up and top-down segmentation. In: ECCV, pp. IV: 581–594 (2006)
44. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV **60**(2), 91–110 (2004)
45. Mikolajczyk, K., Schmid, C., Zisserman, A.: Human detection based on a probabilistic assembly of robust part detectors. In: ECCV. Springer (2004)
46. Mori, G., Malik, J.: Recovering 3D human body configurations using shape contexts. IEEE PAMI **28**(7), 1052–1062 (2006)
47. Pearl, J.: Heuristics: Intelligent Search Strategies for Computer Problem Solving. Addison-Wesley (1984)
48. Pham, T.V., Smeulders, A.W.M.: Object recognition with uncertain geometry and uncertain part detection. CVIU **99**(2), 241–258 (2005)
49. Platt, J.: Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In: Advances in Large Margin Classifiers, pp. 61–74. MIT Press (2000)
50. Ponce, J., Hebert, M., Schmid, C., Zisserman, A. (eds.): Toward Category-Level Object Recognition, *LNCS*, vol. 4170. Springer (2006)
51. Quattoni, A., Collins, M., Darrell, T.: Conditional random fields for object recognition. In: NIPS (2004)
52. Ramanan, D., Forsyth, D.A., Zisserman, A.: Tracking people by learning their appearance. IEEE PAMI **29**(1), 65–81 (2007)
53. Rifkin, R., Klautau, A.: In defense of one-vs-all classification. Journal of Machine Learning Research **5**, 101–141 (2004)
54. Roberts, T., McKenna, S., Ricketts, I.: Human pose estimation using partial configurations and probabilistic regions. Int. J. Comp. Vision **73**(3), 285–306 (2007)
55. Rosenhahn, B., Brox, T., Weickert, J.: Three-dimensional shape knowledge for joint image segmentation and pose tracking. Int. J. Comp. Vision **73**(3), 243–262 (2007)
56. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach. Pearson Education (2003)
57. Schmidt, S., Kappes, J.H., Bergtholdt, M., Pekar, V., Dries, S., Bystrov, D., Schnörr, C.: Spine detection and labeling using a parts-based graphical model. In: N. Karssemeijer, B. Lelieveldt (eds.) Information Processing in Medical Imaging, no. 4584 in LNCS, pp. 122–133. Springer (2007). DOI 10.1007/978-3-540-73273-0_11
58. Seemann, E., Leibe, B., Schiele, B.: Multi-aspect detection of articulated objects. In: CVPR (2006)
59. Sigal, L., Black, M.: Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In: CVPR, vol. 2 (2006)
60. Sigal, L., Black, M.J.: Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Tech. Rep. CS-06-08, Brown University, Department of Computer Science, Providence, RI (2006)
61. Sivic, J., Russell, B., Efros, A., Zisserman, A., Freeman, W.: Discovering objects and their locations in images. In: ICCV. IEEE (2005)
62. Sminchisescu, C., Kanaujia, A., Metaxas, D.: Bm^{3}e: Discriminative density propagation for visual tracking. IEEE Trans. Patt. Anal. Mach. Intell. **29**(11), 2030–2044 (2007)
63. Sudderth, E., Ihler, A., Freeman, W., Willsky, A.: Nonparametric belief propagation. In: CVPR '03: Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2003)
64. Sutton, C., McCallum, A., Rohanimanesh, K.: Dynamic conditional random fields: Factorized probabilistic models for labeling and segmenting sequence data. Journal of Machine Learning Research **8**, 693–723 (2007)
65. Szeliski, R., Zabih, R., Scharstein, D., Veksler, O., Kolmogorov, V., Agarwala, A., Tappen, M., Rother, C.: A comparative study of energy minimization methods for markov random fields. In: Proc. ECCV (2006)
66. Wainwright, M.: Estimating the wrong markov random field: Benefits in the computation-limited setting. In: Y. Weiss, B. Schölkopf, J. Platt (eds.) Advances in Neural Information Processing Systems 18, pp. 1425–1432. MIT Press, Cambridge, MA (2006)
67. Wainwright, M., Jaakola, T., Willsky, A.: Map estimation via agreement on trees: message-passing and linear programming. IEEE Trans. Inform. Theory **51**(11), 3697– 3717 (2005)
68. Weber, M., Welling, M., Perona, P.: Unsupervised learning of models for recognition. In: ECCV, pp. 18–32 (2000)
69. Welk, M., Weickert, J., Becker, F., Schnörr, C., Feddern, C., Burgeth, B.: Median and related local filters for tensor-valued images. Signal Process. **87**(2), 291–308 (2007)
70. Werner, T.: A linear programming approach to max-sum problem: A review. IEEE Trans. Patt. Anal. Mach. Intell. **29**(7), 1165–1179 (2007)
71. Winkler, G.: Image Analysis, Random Fields and Markov Chain Monte Carlo Methods. Springer (2006)
72. Yedida, J.S., Freeman, W.T., Weiss, Y.: Constructing free-energy approximations and generalized belief propagation algorithms. IEEE Trans. Information Theory **51**(7), 2282–2312 (2005)
73. Yuille, A., Coughlan, J.: An A^* perspective on deterministic optimization for deformable templates. Pattern Recognition **33**(4), 603–616 (2000)
74. Zadrozny, B., Elkan, C.: Transforming classifier scores into accurate multiclass probability estimates. In: KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 694–699. ACM (2002)
75. Zhang, J., Marszalek, M., Lazebnik, S., Schmid, C.: Local features and kernels for classification of texture and object categories: A comprehensive study. Int. J. Comp. Vision **73**(2), 213–238 (2007)
76. Zhang, L., Nevatia, R., Wu, B.: Detection and tracking of multiple humans with extensive pose articulation. In: Proc. ICCV (2007)