Recovering Human Body Configurations using Pairwise Constraints between Parts

Xiaofeng Ren, Alexander C. Berg, Jitendra Malik
Computer Science Division
University of California, Berkeley, CA 94720
{xren,aberg,malik}@cs.berkeley.edu

Abstract

The goal of this work is to recover human body configurations from static images. Without assuming a priori knowledge of scale, pose or appearance, this problem is extremely challenging and demands the use of all possible sources of information. We develop a framework which can incorporate arbitrary pairwise constraints between body parts, such as scale compatibility, relative position, symmetry of clothing and smooth contour connections between parts. We detect candidate body parts from bottom-up using parallelism, and use various pairwise configuration constraints to assemble them together into body configurations. To find the most probable configuration, we solve an Integer Quadratic Programming *problem with a standard technique using linear approximations. Approximate IQP allows us to incorporate much more information than the traditional dynamic programming and remains computationally efficient.* 15 *hand-labeled images are used to train the low-level part detector and learn the pairwise constraints. We show test results on a variety of images.*

1. Introduction

The goal of this work is to take an image such as the one in Figure 1(a), detect a human figure, and find the configuration of parts (b). This is a very difficult problem, partly because human bodies are versatile, presenting a wide range of pose and aspects, many including self-occlusion, and partly because variations in clothing and background clutter deny a simple appearance model.

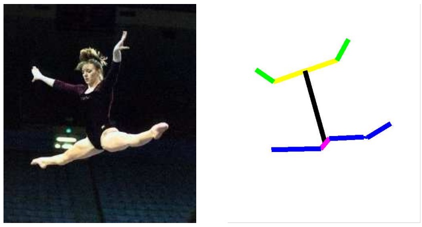


Figure 1: The challenge is to take an input image **(a)**, and recover the body configuration in **(b)**.

Given the seemingly insurmountable difficulties, many existing approaches to this problem make simplifications of one sort or another, either assuming knowledge of scale and appearance/color, or using motion information from video sequences for background subtraction, or limiting evaluation to restricted domains such as walking figures. In these cases, a canonical tree-based model is typically used to model body parts, where dynamic programming can be applied.

We tackle the problem in a more general setting. Without restrictions in pose, appearance, or background clutter, a tree-based model no longer suffices (*cf* [11]). Additional sources of information, not provided by tree based models, are required to succeed. For example, the symmetry of clothing is a powerful cue to constrain limb appearance. As another example, in Figure 1, what reveals the body position to us are the connection between the two upper legs and the relative geometric relationship between arms and legs, both of which are not in the traditional tree-based model.

It is an open question what models can express sufficient constraints and are computationally feasible. In this work, we develop a strategy that exploits a rich set of cues, defined on arbitrary pairs of parts, to constrain body configurations. We learn these constraints from empirical data and use *Integer Quadratic Programming* (IQP) to find the most probable configurations. IQP is a well-studied computational framework, where efficient approximations exist. Many cues for estimating human body configuration can be expressed as pairwise constraints. In our experiments we have found that IQP works well for this problem. The IQP framework allows incorporating much more information than dynamic programming on trees, and can handle a much larger set of candidate parts than a brute-force search strategy (as used in [11]).

2. Related Work

Finding people is a hard problem; yet it is a problem of great interest to both scientific researchers and engineers. One of the earliest lines of research on this problem is in the limited setting of detecting and tracking pedestrians. Starting with Hogg[5], there has been a great deal of work done on using 3D kinematic models for tracking [3]. These 3D models have a high degree of kinematic freedom and typically require hand initialization. Lee and Cohen [7] recently managed to use 3D models to detect people mostly in standing poses, making inference with Data-Driven MCMC.

More recent developments in pedestrian detection typically use a large amount of training data and make use of cleverly designed classifiers, the most successful of which is probably that of Viola et al. [18]. These template-based approaches do not recover joint locations, and have not yet been generalized to accommodate more pose variations.

Realizing the difficulties of using 3D part-based models, many researchers have used of 2D holistic exemplars for tracking and localizing body parts [17, 10, 16]. The main problem with such exemplar-based approaches is that they lack an intrinsic notion of parts, therefore having to deal with a combinatorial explosion when variations of pose, clothing, and clutter increase. Shakhnarovich et al. [14] take a brute-force approach to attacking this complexity explosion, using a variant of Locality Sensitive Hashing to speed up search. However, such an approach still requires millions of exemplars, if not more, even just for the upper body with common poses, background subtraction, and known scale.

There have been many approaches that explicitly model the human body as an assembly of 2D parts [15, 2, 6]. Typically they adopt a two-stage strategy: a bottom-up detector is applied on the image to extract candidate parts, then a top-down procedure makes inference about the configuration and finds the best assembly. Most approaches use the canonical tree model for body parts, hence solving the assembly problem efficiently with dynamic programming.

The work of Mori et al. [11] is the most relevant to us, also using a bottom-up strategy. Their work mainly relies on sophisticated low-level processing, using Normalized Cuts to find a few salient body parts. They then solve the assembly problem by brute-force search. Instead, we use a relatively simple low-level detector, and solve the assignment problem using Integer Quadratic Programming (IQP), which can systematically explore arbitrary pairwise constraints between parts.

3. Our Approach

The ultimate goal of our line of research is to develop a general method to recover configurations of human bodies, or other articulated objects, from static images. What characterizes an articulated object is that the object is made of a collection of simple rigid parts that are constrained under a global configuration. It is self-evident, therefore, that any approach without explicitly modeling the part structure would have great difficulties in handling pose variation, appearance change or background clutter.

There are in general two ways to detect parts of articulated objects: top-down and bottom-up. A typical top-down approach is to design rectangle-like filters or templates that model the shape of each object part, and match them to every possible location in the image. Such template matching is useful if one knows a priori the scale and the appearance of what he looks for. Because we aim at finding people in a general setting, this would require us to run part detectors at multiple scales, orientations, and aspect ratios (e.g., to account for foreshortening). We would find far too many candidate parts to be efficiently searched and assembled.

Our approach to finding people is to first detect candidate body parts from bottom-up, and then search for the combination of the candidate parts that is most probable for human bodies. Figure 2 shows an example of how information flows through various stages of the process:

Starting with the input image in Figure 2(a), we use the local *Probability of Boundary* (*Pb*) operator [9] to compute a soft edge map in Figure 2(b). We use Canny's hysteresis to convert the soft edge map into contours, and recursively split them into piecewise straight lines. We then use *constrained Delaunay triangulation* (CDT) to complete this scale-invariant discrete line structure into a triangulation (Figure 2(c)) [13].

We model a body part by a pair of parallel lines and build a discriminative part detector on the basis of the CDT triangulation. For each pair of edges in the triangulation, we use a logistic classifier to compute its low-level saliency as a body part. The logistic classifier is trained from 15 images extracted from a skating sequence performed by Tara Lipinski with hand-labeled parts.

Figure 2(d) shows the candidate parts detected in this image. Without the knowledge of scale or appearance, our part detector is fairly weak; there are a lot of false detections. After all, parallelism is a generic mid-level cue, and body parts by themselves are not distinctive. It is the configuration of parts that is distinctive for human bodies.

In Section 5 we define a variety of configuration constraints between pairs of parts. They go beyond the traditional tree-based model and incorporate constraints such as compatibility of part widths, symmetry of appearance, and smooth connectivity between parts. We learn the pairwise constraints from the same 15 hand-labeled Lipinski images.

To recover human body configuration is to assign part labels to detected candidate parts. Assuming that the pairwise constraints are independent, these constraints can be

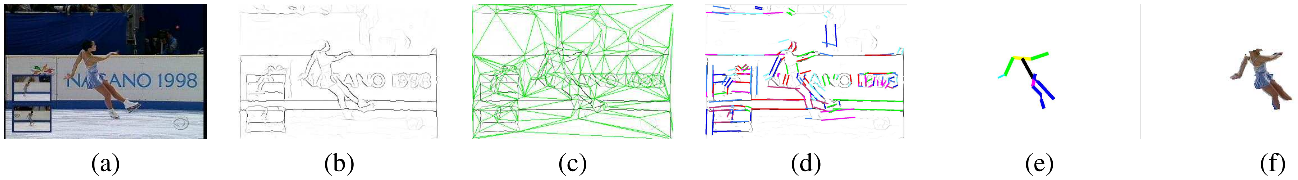


Figure 2: The processing pipeline: given an input image **(a)**, compute an edge map **(b)**, break this into segments and compute a constrained Delaunay triangulation **(c)**, identify part candidates by exploiting parallelism of part boundaries**(d)**, find a good configuration using Integer Quadratic Programming over pairwise constraints between body parts **(e),** use the labeled segments and stick figure to find an approximate segmentation of the figure **(f)**.

formulated as an assignment problem with a quadratic cost function. Hence we can compute the most probable body configurations by solving an *Integer Quadratic Programming* (IQP) problem. The IQP problem is solved by using an efficient linear approximation scheme to IQP [8, 1]. Once we have found the most probably configuration in Figure 2(e), it is straightforward to find the associated segmentation mask shown in Figure 2(f).

4. Finding Body Parts

We choose to detect candidate body parts from bottom-up. Our approach is based on the following key observation: parts of a human body, or of an articulated object in general, are mostly characterized by a pair of parallel line segments. *Parallelism* or *Ebenbreite*, known from the early days of the Gestalt movement, is a fundamental and powerful principle in human vision. It is common understanding that, being a mid-level cue, the perception of parallelism occurs early in the visual pathway. Our approach here follows this theory: first we construct a discrete structure of edges in an image by grouping them into approximately straight contour elements. Then we use *constrained Delaunay triangulation* to complete the gaps between contour elements. Finally we train a classifier on a pair of elements to compute the probability of them forming the boundary of a body part.

4.1. Constrained Delaunay Triangulation

As the first step of our bottom-up processing, we use the local Pb operator [9] to compute a soft edge map. We use Canny's hysteresis trick to trace Pb edges in the image into continuous contours. We then recursively split these contours into pieces, until each contour element is approximately straight. This process gives us a discrete graph, the elements of which are straight contours of Pb edges. We note that this discretization is scale-invariant: a straight line, no matter how long it is, remains a single line in the graph.

We use *constrained Delaunay triangulation* to complete gaps between the detected contour elements. The constrained Delaunay triangulation (CDT) is a variant of the standard Delaunay triangulation (DT) in which a set of user-specified edges must lie in the triangulation. The details of this completion process can be found in [13].

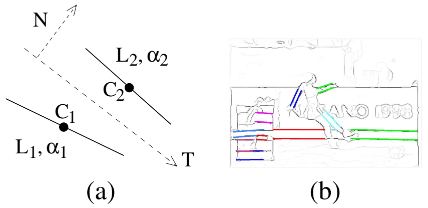


Figure 3: (a) Classifying parallel line segments. (b) Top 10 candidate parts detected in one image.

Figure 2(c) shows an example of the CDT triangulation. The linearized edges extracted from the Pb contours become constrained edges which we refer to as gradient edges or G-edges (black), and the rest are the completions by the CDT algorithm, which we refer to as completion edges or C-edges (green).

4.2. Finding Parallel Line Segments

We detect candidate body parts by finding well-aligned parallel lines in the CDT graph. Consider a pair of contour elements (Figure 3): let L denote the length of a contour element, α its orientation, $\vec{C}$ its center, and Pb the average contrast on this element. Let $\vec{T}$ denote the average tangent direction and $\vec{N}$ the normal direction. We define the following set of features: (1) **orientation consistency** $|\alpha_1 - \alpha_2|$; (2) **length consistency** $|L_1 - L_2|/(L_1 + L_2)$; (3) **low-level contrast** $|Pb_1 + Pb_2|$ and $|Pb_1 - Pb_2|$; (4) **distance between centers**, both in the normal direction $|(\vec{C}_1 - \vec{C}_2) \cdot \vec{N}|/(L_1 + L_2)$ and in the tangent direction $|(\vec{C}_1 - \vec{C}_2) \cdot \vec{T}|/(L_1 + L_2)$; and (5) **intervening contour** Pb_{IC}, where Pb_{IC} is the maximum Pb contrast on edges intersecting the straight line connecting $\vec{C}_1$ to $\vec{C}_2$ [1].

We train a simple logistic classifier to combine these features. For training, we use the hand-labeled body parts in the Lipinski dataset as positive examples. As negative examples, we use all pairs of contour elements whose centers are sufficiently close ($<= 5$ hops in the CDT graph).

[1] For simplicity we approximate this straight line by computing the shortest path between C_1 and C_2 in the CDT graph.

Most pairs of lines in the graph are unlikely to be body parts: either they are not parallel or too far away from each other. Hence we take a simplifying step that, for each line e in the CDT graph, we keep only the candidate limb segment using e with the best low-level score. Figure 3(b) shows the candidate parts with the top 10 low-level scores (cf Figure 2 showing all of the candidates above threshold). We note that the candidate parts are very different in scale and aspect ratio: given a pair of edges in the CDT graph, scale and aspect ratio are automatically determined. Therefore in our bottom-up detection step, there is no need to explicitly search over all possible scales and aspect ratios. And while a top-down rectangle detector would fire many times on the long parallel bars in the background, we only find one candidate part which will later be discarded due to its aspect ratio or because it is not compatible with enough other limb candidates.

5. Configuration Constraints

Most existing approaches model the human body as a tree of parts. The typical configuration constraints used in a tree model are positional and orientation constraints between adjacent parts, such as *torso-upper limb* connection, or *upper-lower limb* connection. These are, however, only a subset of the information that is available for recovering human body configurations.

One important cue missing from the tree model is the symmetry of clothing: corresponding parts, such as the two forearms, are usually clothed in the same way and thus similar in color. This cue can be very useful in identifying arm positions. Another example is the connectivity between two upper legs. They form a stereotypical "V"-shape, which is typically very salient and heavily exploited by the human visual system. There are many other useful cues between a pair of body parts.

5.1. Constraints between Parts

What is a good configuration? Individual parts have to be consistent with the global configuration. We approximate the global configuration consistency by defining pairwise constraints between parts.

Let c be a candidate part (two roughly parallel lines) detected from the image, and l be a part label (e.g., left upper leg). There are some simple unary constraints on this assignment (l, c):

1. aspect ratio $f_{aspect}(l, c)$: anthropometric data [12] provides us constraints on the aspect ratio of each individual part. Parts can be and often are foreshortened; however, the aspect ratio $length/width$ can only be smaller, but not much larger, than the expected aspect ratio;

2. low-level score $f_{lowlevel}(l, c)$: for a candidate part c, we have a measure of the low-level saliency $P_{lowlevel}(c)$ from the part detector, the posterior from the logistic classifier. We use $f_{lowlevel}(l, c) = \log(P_{lowlevel}(c))$. This feature is independent of the label l.

The unary constraints are very weak in nature. Without knowing the global scale or its relations to other parts, a candidate part can be labeled as almost anything, e.g., a torso or a lower leg. More important for recovering configurations are the constraints between parts. We define the following set of cues between a pair of assignments (l_1, c_1) and (l_2, c_2):

1. scale consistency $f_{scale}(l_1, c_1, l_2, c_2)$: body parts are roughly speaking cylindrical. Although *length* is unreliable because of foreshortening, *width* is a good estimate of the global object scale. Let $w_1 = width(c_1)$ and $w_2 = width(c_2)$, compute the ratio $r = (w_1 - w_2)/(w_1 + w_2)$; we compare r to $\tilde{r} = (\tilde{w}_1 - \tilde{w}_2)/(\tilde{w}_1 + \tilde{w}_2)$, where $\tilde{w}_1 = width(l_1)$ and $\tilde{w}_2 = width(l_2)$ are the "expected" widths of these two part labels, as given in anthropometric statistics. This is invariant to a global change in scale.

2. appearance consistency: the appearance of corresponding parts are similar; this constraint is valid for certain pairs of parts, e.g., between upper legs and between lower legs. Let (L, A, B) be the average color of a part, we compute the difference $f_L = |L_1 - L_2|$ and $f_{ab} = |(A_1, A_2) - (B_1, B_2)|$.

3. orientation consistency $f_{orient}(l_1, c_1, l_2, c_2)$: let α be the orientation of a part, we compute the difference $f_{orient} = |\alpha_1 - \alpha_2|$. This cue is useful mostly for adjacent body parts. Because of the large variations in pose in the training data, the orientation consistency turns out to be a weak cue.

4. connectivity: adjacent body parts should be adjacent in the image; this is the most commonly used cue in constraining possible body configurations. However, connectivity means much more. The V-shape constraint between two upper legs is one example. Another one is the connectivity cue between an upper arm and an upper leg, i.e., there is typically a "smooth" contour connecting them through the torso. We discuss more about how to quantify this smooth connectivity in the next section.

5.2. Smooth Connections between Parts

To quantify the smoothness of a connection, we compute the shortest path between two parts in the CDT graph (Figure 4(a)), where the path goes through contour elements, i.e. edges in the CDT graph G_{CDT}, instead of pixels. In the CDT graph, a path could go through G-edges as well as C-edges (gaps). We would like to tolerate small gaps but avoid jumping large distances along C-edges. Therefore, we raise the cost of traveling along a C-edge by a constant ratio (3.0).

We can compute a number of connectivity cues for a pair of edges (a_1, a_2): $dist(a_1, a_2)$, the total distance of the

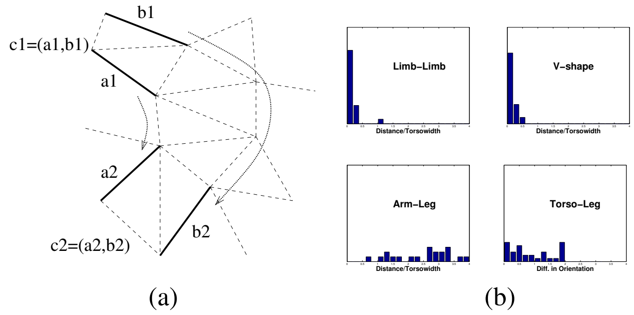


Figure 4: (a) Defining pairwise connectivity constraints on parts. (b) Empirical distributions of distance constraints from training data.

shortest path $a_1 \rightarrow a_2$ (relative to torso width, which we can estimate from the two parts under consideration given anthropometric data); $gap(a_1, a_2)$, the total length of C-edges or gaps on this path; $angle(a_1, a_2)$, the maximum change of angle along the path; and $turn(a_1, a_2)$, the number of "turns" or T-junctions along this path (where traveling through a T-junction is counted as a "turn" if it does not go through the pair of edges that form the minimum angle at this junction).

Now consider a pair of candidates c_1 and c_2: each part has two bounding edges, (a_1, b_1) and (a_2, b_2) respectively. We find a correspondence between these two pairs of edges, such that $dist(a_1, a_2)$, the distance between (a_1, a_2), are the minimum of all four combinations; and (b_1, b_2) are the two other edges. Thus we can define two sets of connectivity cues between parts, one from the path $(a_1 \rightarrow a_2)$ and one from $(b_1 \rightarrow b_2)$. Sometimes both of them are very informative, such as the connectivity between an upper leg and the lower leg, as we expect both sides to be well connected. Sometimes only one of them is useful, such as the connectivity between an upper arm and an upper leg.

5.3. Torso-Limb Constraints

The torso has a special role as the hub that connects all the limbs together. The constraints that follow are designed to make the optimization problem somewhat easier.

1. Torso orientation $f_{torsoorient}(c)$: we assume that the torso is oriented upward. Let θ be the leaning angle of the candidate part c, we compute $f_{torsoorient} = |tan(\theta)|$;

2. Left/right disambiguation: given a torso candidate c and its two bounding edges a and b, we know which edge is the left side and which is the right side. When we compute the connectivity of the left (or right) limbs to the torso, we only consider the connectivity to the left (or right) side of the torso.

3. Arm-Leg disambiguation: given the torso orientation, we assume that the center of upper legs cannot be higher (along torso orientation) than the top of the torso, and upper arms cannot be lower than the bottom of the torso.

5.4. Learning the Constraints

We use the 15 hand-labeled images of an ice-skater, Lipinski, for training. With such a limited amount of training data, it would be difficult to learn the interactions between all the pairwise constraints. Thus, for simplicity, we assume that the constraints are all independent of each other, and they have a Gaussian distribution. 15 images are sufficient for us to estimate the mean and standard deviation from the empirical data. For some constraints, such as the distance between upper leg and lower leg, we know it should be zero in the ideal case. In such cases we fix the mean of this constraint to be zero.

Figure 4(b) shows a few empirical distributions of distance constraints. The distance between two upper legs (the V-shape) is typically zero or very small. In comparison, the distance between upper legs and upper arms is less reliable and has a high variance. The orientation constraint between torso and upper leg turns out to be weak (in many cases > 90 degrees), as the ice-skater in the training images goes through a wide range of poses.

Another observation is that these distributions are clearly non-Gaussian. We leave it for future work to build better parametric models for each type of constraint, possibly with more training data.

5.5. Non-traditional Constraints

We have introduced a number of configuration constraints that are not found in the traditional tree-based models, for example: f_{ab}, the symmetry of color between corresponding left/right limbs; $f_{dist(V)}$, the V-shape distance or the distance between two upper legs; and $f_{gap(UA,UL)}$, the gap (C-edges on the shortest path) between upper arms and upper legs. Based on the training images, we attempt to quantify their usefulness and compare to traditional cues, such as $f_{dist(L,L)}$, the distance between adjacent limbs, and $f_{dist(L,T)}$, the distance between upper legs and torsos.

Let $f(l_1, c_1, l_2, c_2)$ be a constraint under study, and let Z be a binary variable such that $Z = 1$ iff a random assignment (l_1, c_1, l_2, c_2) is good (i.e., in the groundtruth). One way to quantify the information in f is to measure the relative information gain $\mathbf{I}(Z, f)/\mathbf{H}(Z)$, where $\mathbf{H}(Z)$ is the entropy or uncertainty of the label Z and $\mathbf{I}(Z, f)$ is the mutual information between Z and f.

Constraint	Relative Information
f_{ab}	6.1%
$f_{dist(V)}$	34.9%
$f_{gap(UA,UL)}$	10.4%
$f_{dist(L,L)}$	44.5%
$f_{dist(L,T)}$	27.1%

We find that these non-traditional constraints do contain a significant amount of information about the configuration, comparable to the traditional tree-based constraints. Of course, this is only the marginal information gain; more important questions would be how much extra information they provide and how much they help improve the end performance. These are subtle questions and we will try to address them in future work.

6. Integer Quadratic Programming

Recovering human body configuration in an image can be formulated as an assignment of body part labels $\{l_i\}$ to candidate body parts $\{c_j\}$. We use the unary and pairwise constraints f_k introduced in the last section to define the "goodness" of an assignment. We model each constraint f_k with a Gaussian distribution with parameters (μ_k, σ_k). Let $\bar{f}_k = (f_k - \mu_k)^2 / \sigma_k^2$. Assuming that all the constraints are independent results in a product of Gaussians as the probabilistic model. Finding the maximum likelihood assignment $\pi : \{l_i\} \to \{c_j\}$ is then equivalent to minimizing the sum:

$$\sum_{l_1,l_2} \sum_k \bar{f}_k \left(l_1, \pi\left(l_1\right), l_2, \pi\left(l_2\right)\right) + \sum_l \sum_k \bar{f}_k \left(l, \pi(l)\right) \quad (1)$$

where l_i is a part label, and $\pi(l_i)$ is the part candidate assigned to l_i by π. The first sum is over pairwise constraints $\bar{f}_k$'s. For instance one $\bar{f}_k\left(l_1, \pi\left(l_1\right), l_2, \pi\left(l_2\right)\right)$ might measure the scale consistency of labeling one part candidate as a leg and another as a torso [2]. The second sum is over each limb label l where $\bar{f}_k\left(l, \pi(l)\right)$ are the unary constraints (e.g., the low-level saliency score).

Integer Quadratic Programming Minimizing Equation 1 can be written as an integer quadratic programming problem (IQP). The assignment π is represented by a binary vector x. Each entry x_i indicates whether one particular part candidate $c(i)$ is labeled with a particular part label $l(i)$. In order for x to represent a valid assignment there is a constraint that for each part label $\hat{l}$, $\sum_{i:l(i)=\hat{l}} x_i = 1$. We can now write the integer quadratic programming problem:

$$\begin{aligned} \min Q(x) =& x'Hx + c'x \\ \text{subject to } & Ax = b, \quad x \in \{0,1\}^n \end{aligned} \quad (2)$$

here H is a matrix representing the pairwise consistency,

$$H_{i,j} = \sum_k \bar{f}_k \left(l(i), c(i), l(j), c(j)\right)$$

similarly we have $c_i = f(l(i), c(i))$. Finally $Ax = b$ expresses the constraints that x represents a valid assignment (as above).

The binary vector x that minimizes Equation 2 corresponds to the assignment that minimizes Equation 1 and therefore has the maximum likelihood under our model.

Linear Bound A linear bounding function $L(x)$ is constructed so that $L(x) < Q(x)$ for all x. Note that from this point forward the constraints from Equation 2 are assumed, but not written.

$$q_i = \min_x \sum_j H(i,j) x_j \quad (3)$$

If x_i indicates assigning limb $l(i)$ to candidate $c(i)$, then $q_i + c_i$ is a lower bound for the cost contributed to any assignment mapping $l(i)$ to $c(i)$. Now we can write the bounding function, $L(x) = \sum_i (q_i + c_i) x_i$. Finding the x that minimizes L and finding the q_i in Equation 3 subject to the constraints in Equation 2 is simple because the vertices of the constraint polytopes lie only on integer coordinates. As a result the integer linear programming problems can be relaxed to linear programming problems without changing the optima.

This construction follows [8] and [1], and is a standard bound for a quadratic program.

Greedy Search Starting from the assignment that minimizes L we perform a greedy local search considering up to two changes in the assignment at a time. Considering two changes is important in order to move both upper and lower parts of a limb out of a poor configuration.

Complexity Integer quadratic programming can be quite challenging, it is after all an NP-hard problem; but it turns out the instances generated as described above are not so difficult. A simple linear approximation followed by a greedy local search produces reasonably good results.

There are more complex approximations to IQP, using semidefinite programming (SDP), with guaranteed bounds on approximation error [4]. However, in this work a simple approximation produces results within the approximation bound and with significantly lower time and space complexity than [4].

If n is the length of x, computing the x that minimizes $L(x)$ (including computing all of the q_i) takes $O(n^2)$ operations with a very small constant. Each gradient descent step requires approximately the same number of operations. As a comparison SDP techniques are polynomial in n^2, effectively many, many times slower as $n \sim 1300$.

7. Experimental Results

We have tested our algorithm on a variety of images, including extracted frames from a skating sequence of Michel Kwan, and other gymnastic images. Examples are shown in Figure 7, with recovered body configurations and the associated segmentation masks.

To obtain these results, we use a standard 9-part model for the human body, i.e., torso plus left/right upper/lower

[2] Note that for some pairs of limbs and features, $\bar{f}_k$ will be zero – for instance $\bar{f}_k$ is zero for all k whenever l_1 is a lower arm and l_2 is a lower leg, as we do not include such constraints in the model.

legs/arms. The number of candidate parts detected per image is in the range of $100 \sim 200$. We use the set of Gaussian parameters learned from a different skating sequence with a different skater, Lipinksi. Because Gaussian models are not very accurate for some features, we choose to "cut off" the features at 4σ; i.e., any cost term above 4σ is considered as being infinity.

Because of the heterogeneous nature of the constraints, our IQP problem is hard to solve. We have found that, occasionally, the linear approximation to IQP fails to locate the correct body parts. The linear approximation step in Equation 3 basically tries to find for each line (or each possible labeling) the best consistent assignment. This is done without considering the pairwise constraints between other parts. Therefore the linear approximation could fail and the gradient descent that follows may not be able to correct the errors.

To remedy this problem, we make use of an empirical observation: that although torsos typically have poor low-level saliency, they are the most constrained part of the body and therefore can be most reliably detected in the linear approximation scheme. We use the following two-step strategy: in the first step, we run the linear approximation to obtain a shortlist of 5 best torso candidates. In the second step, we go through the shortlist, pick one candidate part, fix its label to be the torso and re-solve the IQP problem, with the same cost matrix H. Fixing the torso is appealing because it helps constrain all the upper legs/arms in the configuration. Finally we pick the solution that has the lowest cost Q.

8. Conclusion

In this work we develop a strategy to use pairwise constraints between human body parts to recover body configurations from static images. We detect candidate body parts from bottom-up using parallelism cues to search a discrete graph structure given by Constrained Delaunay Triangulation. Finding the configuration of a human body is then an assignment problem: for each body part label, we decide which candidate part should be assigned to that label. We formulate the assignment as a *Integer Quadratic Programming* (IQP) problem.

As compared to the traditional tree-based model and the associated dynamic programming algorithm, IQP allows us to incorporate a much richer set of constraints, namely arbitrary constraints between pairs of body parts. This includes the important cues such as the symmetry of clothing, the canonical V-shape between upper legs, and the smooth contour connectivity between arms and legs. As compared to a brute-force search approach in [11], we are able to handle a much larger set of candidate parts and do not rely on the availability of a few being very salient. We have found that a two-step strategy using the linear approximation of IQP works well for our assignment problem, produces satisfactory results on a variety of images without relying on extensive low-level processing, and is computationally efficient. We believe that the IQP formulation will find more and more use in detecting articulated objects.

Acknowledgments. This research was supported by ONR N00014-01-1-0890 (MURI) and NSF ITR IIS-00-85864.

References

[1] A. Berg, T. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondence. In *CVPR*, 2005.

[2] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient matching of pictorial structures. In *CVPR*, pages 66–73, 2000.

[3] D. M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding: CVIU*, 73(1):82–98, 1999.

[4] M. Goemans and D. Williamson. Imroved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6):1115–1145, 1995.

[5] D. Hogg. Model-based vision: A program to see a walking person. *Image and Vision Computing*, 1(1):5–20, 1983.

[6] S. Ioffe and D. Forsyth. Probabilistic methods for finding people. *Int. Journal of Computer Vision*, 43(1):45–68, 2001.

[7] M. W. Lee and I. Cohen. Proposal maps driven mcmc for estimating human body pose in static images. In *CVPR*, volume 2, pages 334–341, 2004.

[8] J. Maciel and J. Costeira. A global solution to sparse correspondence problems. *IEEE Trans. PAMI*, 25(2):187–199, 2003.

[9] D. Martin, C. Fowlkes, and J. Malik. Learning to find brightness and texture boundaries in natural images. *NIPS*, 2002.

[10] G. Mori and J. Malik. Estimating human body configurations using shape context matching. In *ECCV*, volume 3, pages 666–680, 2002.

[11] G. Mori, X. Ren, A. Efros, and J. Malik. Recovering human body configurations: Combining segmentation and recognition. In *CVPR*, volume 2, pages 326–333, 2004.

[12] NIST. Anthrokids - anthropometric data of children, http://ovrt.nist.gov/projects/anthrokids/, 1977.

[13] X. Ren, C. Fowlkes, and J. Malik. Scale-invariant contour completion using conditional random fields. In *ICCV*, 2005.

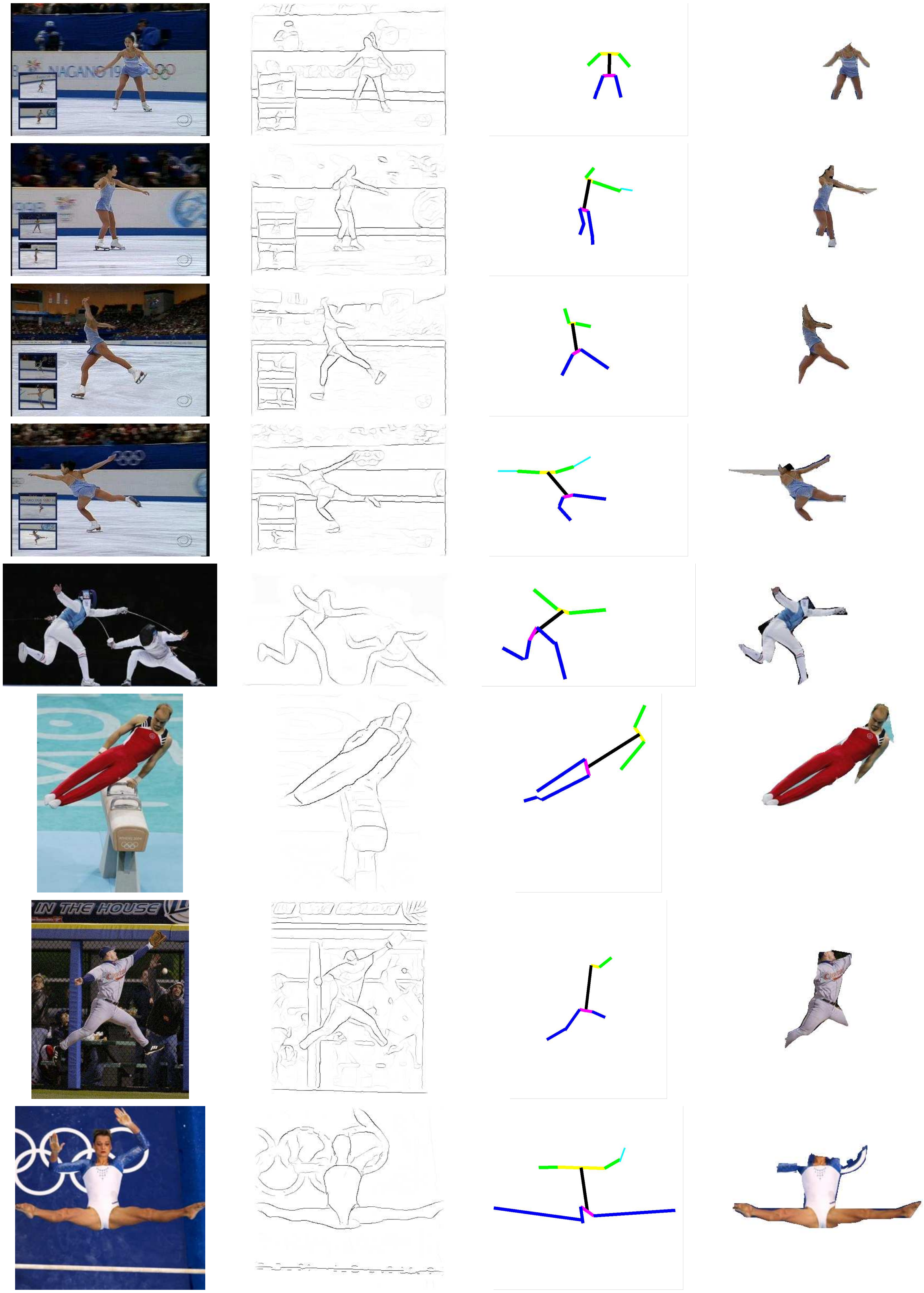
[14] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter sensitive hashing. In *ICCV*, volume 2, pages 750–757, 2003.

[15] Y. Song, L. Goncalves, and P. Perona. Unsupervised learning of human motion. *IEEE Trans. PAMI*, 25(7):814–827, 2003.

[16] J. Sullivan and S. Carlsson. Recognizing and tracking human action. In *ECCV*, volume 1, pages 629–644, 2002.

[17] K. Toyama and A. Blake. Probabilistic exemplar-based tracking in a metric space. In *ICCV*, volume 2, pages 50–57, 2001.

[18] P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *ICCV*, pages 734–741, 2003.



(a) (b) (c) (d)