Bottom-up Recognition and Parsing of the Human Body

Praveen Srinivasan*
GRASP Laboratory, University of Pennsylvania
3330 Walnut St., Philadelphia, PA 19104
psrin@seas.upenn.edu

Jianbo Shi
GRASP Laboratory, University of Pennsylvania
3330 Walnut St., Philadelphia, PA 19104
jshi@cis.upenn.edu

Abstract

Recognizing humans, estimating their pose and segmenting their body parts are key to high-level image understanding. Because humans are highly articulated, the range of deformations they undergo makes this task extremely challenging. Previous methods have focused largely on heuristics or pairwise part models in approaching this problem. We propose a bottom-up parsing of increasingly more complete partial body masks guided by a parse tree. At each level of the parsing process, we evaluate the partial body masks directly via shape matching with exemplars, without regard to how the parses are formed. The body is evaluated as a whole, not the sum of its constituent parses, unlike previous approaches. Multiple image segmentations are included at each of the levels of the parsing, to augment existing parses or to introduce ones. Our method yields both a pose estimate as well as a segmentation of the human. We demonstrate competitive results on this challenging task with relatively few training examples on a dataset of baseball players with wide pose variation. Our method is comparatively simple and could be easily extended to other objects.

1. Introduction

Recognition, pose estimation and segmentation of humans and their body parts remain important unsolved problems in high-level vision. Action understanding and image search and retrieval are just a few of the areas that would benefit enormously from this task. There has been good previous work on this topic, but significant challenges remain ahead. We divide the previous literature on this topic into three main areas:

Top-down approaches: [4] developed the well-known pictorial structures (PS) method and applied it to human pose estimation. In the original formulation, PS does probablistic inference in a tree-structured graphical model, where the overall cost function for a pose decomposes across the edges and nodes of the tree, usually with the torso as the root. PS recovers locations, scales and orientations of rigid rectangular part templates that represent a body. Pairwise potentials were limited to simple geometric relations (relative position and angle), while unary potentials were based on image gradients or edge detections. The tree structure is a limitation since many cues (e.g., symmetry of appearance of right and left legs) cannot be encoded. [10] extended the original model to encode the fact that symmetric limb pairs have similar color, and that parts have consistent color or colors in general, but how to incorporate more general cues seems unclear. [11] track people by repeatedly detecting them with a top-down PS method. [14] introduced a non-parametric belief propagation method with occlusion reasoning to determine the pose. All these approaches estimate pose, and do not provide an underlying segmentation of the image. Their ability to utilize more sophisticated cues beyond pixel-level cues and geometric constraints between parts is limited.

Search approaches: [9] utilized heuristic-guided search, starting from limbs detected as segments from Normalized Cuts (NCut) ([3]), and extending the limbs into a full-body pose and segmentation estimate. A follow up to this, [8], introduced an Markov-Chain Monte Carlo (MCMC) method for recovering pose and segmentation. [6] developed an MCMC technique for inferring 3-D body pose from 2-D images, but used skin and face detection as extra cues. [15] utilized a combination of top-down, MCMC and local search to infer 2-D pose.

Bottom-up/Top-down approaches: [12] used bottom-up detection of parallel lines in the image as part hypotheses, and then combined these hypotheses into a full-body configuration via an integer quadratic program. [15] also fit into this category, as they use bottom-up cues such as skin pixel detection. Similarly, [5] integrated bottom-up skin color cues with a top-down, non-parametric belief propagation process. [8] use superpixels to guide their search. While [2] estimate only segmentation and not pose for horses and humans in upright, running poses, they best

*Partially supported by an NSF Graduate Fellowship.

utilize shape and segmentation information in their framework. [13] use bottom-up part detectors to detect part hypotheses, and then piece these hypotheses together using a simple dynamic programming (DP) procedure in much the same way as [4].

2. Overview of Our Parsing Method

Our goal is to combine a subset of salient shapes S (in our case, represented as binary masks, and provided by segmenting the image via NCut) detected in an image into a shape that is similar to that of a human body. Because the body has a very distinctive shape, we expect that it is very unlikely for this to occur by chance alone, and therefore should correspond to the actual human in the scene.

We formulate this as a parsing problem, where we provide a set of parsing rules that lead to a parse (also represented by a binary mask) for the body, as see in Figures 1 and 2. A subset of the initial shapes S are then parsed into a body. The rules are unary or binary, and hence a non-terminal can create a parse by *composing* the parses of one or two children nodes (via the pixel-wise OR operator). In addition the parses for a node can be formed directly from a shape from S, in addition to being formed from a child/children. Traditional parsing methods (DP methods) that exploit a subtree independence (SI) property in their scoring of a parse can search over an exponential number of parses in polynomial time.

We can define a traditional context-free grammar as a tuple

$$\langle V, T, A, R, S \rangle \tag{1}$$

V are parse non-terminals and T are the terminals, where A is the root non-terminal,

$$R = \{A_i \rightarrow B_i, C_i\} \tag{2}$$

is a set of production rules with $A_i \in V$ and $B_i, C_i \in V \cup T$ (we restrict ourselves to binary rules, and unary rules by making C_i degenerate), and S_i is a score for using rule R_i. Further, for each image, a terminal $T_i \in T$ will have potentially multiple instantiations $t_i^j, j = 1, ..., n_i$ each with its own score u_i^j for using $T_i \rightarrow t_i^j$ in a parse. Each terminal instantiation $t_i^j \in S$, corresponds to an initial shape S drawn from NCut segmentation. If the root is $A \in V$, then we can compute the score of the best parse (and therefore the best parse itself) recursively as

$$P(A) = \max_{r_i | r_i = (A \rightarrow B_i, C_i)} (S_i + P(B_i) + P(C_i)) \tag{3}$$

However, this subtree independence property greatly restricts the type of parse scoring function (PSF) that can be used.

By contrast, our approach seeks to maximize a shape scoring function F_A for A that takes as input two specific child parses b_i^j and c_i^k (or one, as we allow unary rules) corresponding to rule $A \rightarrow B_i, C_i$:

$$P(A) = \max_{r_i = (A \rightarrow B_i, C_i)} \max_{j,k} \ (F_A(b_i^j, c_i^k)) \tag{4}$$

Recall that we represent a parse b_i^j or t_i^j as a binary mask, not as the parse rules and terminals that form it. Note that the exact solution requires all parses for the children as opposed to just the best, since the scoring function F_A does not depend on the scores of the child parses. Because the exact solution is intractable, we instead solve this approximately by greedily pruning parses to a constant number. However, we use a richer PSF that has no subtree independence property. We can view the differences between the two methods along two dimensions: **proposal** and **evaluation**.

Proposal: DP methods explore all possible parses, and therefore have a trivial proposal step. Our method recursively groups bottom-up body part parses into increasingly larger parts of the body until an entire body parse is formed. For example, a lower body could be formed by grouping two Legs, or a Thigh+Lower leg and a Lower leg, or taken directly from S. In the worst case, creating parses from two children with n parses each could create n^2 new parses. Therefore, pruning occurs at each node to ensure that the number of parses does not grow exponentially further up the tree. To prune, we eliminate redundant or low scoring parses. Because there is pruning, our method does not evaluate all possible parses. However, we are still able to produce high quality parses due to a superior evaluation function.

Evaluation: On the evaluation side, DP employs evaluation functions with special structure, limiting the types of evaluation functions that can be used. Usually, this takes the form of evaluating a parse according to the parse rule used (chosen from a very limited set of choices) and the scores of the subparses that compose it, as in Equation (3). However, this does not allow scoring of the parse in a holistic fashion. Figure 3 gives an example; two shapes that on their own are not clearly parts of a disk, but when combined together, very clearly form a disk. Therefore, we associate with each node i a scoring function F_i (as in Equation (4)) that scores parses not based on the scores of their constituent parses or the parse rule, but simply based on their shape. The scoring function also allows for pruning, as parses can be ranked and low-scoring parses can be discarded to control the number of parses. It is important to note that our choice of F_i does not exhibit an SI property. Because of this, we are primarily interested in the actual result of the parse, a binary mask, as opposed to how it was generated from child parses or from S. In contrast to DP methods, a parse is evaluated irrespective of how it was generated.

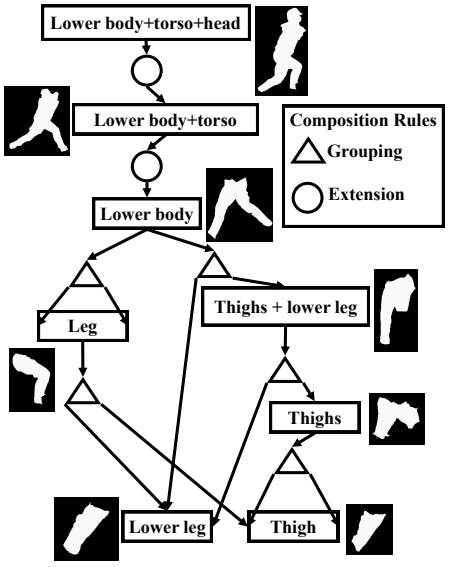

Figure 1. Our body parse tree, shown with an exemplar shape from our training set for each node; the exemplars are used for shape scoring. Shape parsing begins at the leaf nodes of thigh and lower leg and proceeds upwards. Note that in addition to composing parses from children nodes, parses can always come from the initial shapes S.

- {Lower leg, Thigh} → Leg
- {Thigh, Thigh} → Thighs
- {Thighs, Lower leg} → Thighs+Lower leg
- {Thighs+Lower leg, Lower leg} → Lower body
- {Leg, Leg} → Lower body
- {Lower body} → Lower body+torso
- {Lower body+torso} → Lower body+torso+head

Figure 2. Our parse rules. We write them in reverse format to emphasize the bottom-up nature of the parsing.

Figure 3. The two shapes on the left bear little resemblance to a disk in isolation. However, when combined, the disk is clear.

2.1. Multiple Segmentations

To initialize our bottom-up parsing, we need a set of intial shapes S. [9] noted that human limbs tend to be salient regions that NCut segmentation often isolate as a single segment. To make this initial shape generation method more robust, we consider not one segmentation as in [9], but 12 different segmentations provided by NCut. We vary the number of segments from 5 to 60 in steps of 5, giving a total of 390 initial shapes per image. This allows us to segment out large parts of the body that are themselves salient, e.g. the lower body may appear as a single segment, as well as smaller parts like individual limbs or the head. Figure 4 shows for an image 2 of the 12 segmenations with overlaid boundaries. Segments from different segmentations can overlap, or be contained within another. In our system, these segments are all treated equally. These initial shapes could be generated by other methods besides segmentation, but we found segmentation to be very effective.

Figure 4. Two segmentations of an image, 10 and 40 segments. Red lines indicate segment boundaries for 10 segments, green lines indicate boundaries for 40 segments, and yellow indicates boundaries common to both segmentations (best viewed in color).

2.2. Shape Comparison

For each node i, we have an associated shape scoring function F_i. For the root node, this ranks the final parses for us. For all other nodes, F_i ranks parses so that they can be pruned. All the shape scoring functions operate the same way: we match the boundary contour of the mask that represents a parse against boundary contours from a set of exemplar shapes using the inner-distance shape context (IDSC) of [7].

The IDSC is an extension of the original shape context proposed in [1]. In the original shape context formulation, given a contour of n points $x_1, ..., x_n$, a shape context was computed for point x_i by the histogram

$$\#(x_j, j \neq i : x_j - x_i \in bin(k)) \tag{5}$$

Ordinarily, the inclusion function $x_j - x_i \in bin(k)$ is based on the Euclidean distance $d = \| x_j - x_i \|_2$ and the angle $acos((x_j - x_i)/d)$. However, these measures are very sensitive to articulation. The IDSC replaces these with an *inner-distance* and an *inner-angle*.

The inner-distance between x_i and x_j is the shortest path between the two points traveling through the interior of the mask. This distance is less sensitive to articulation. The

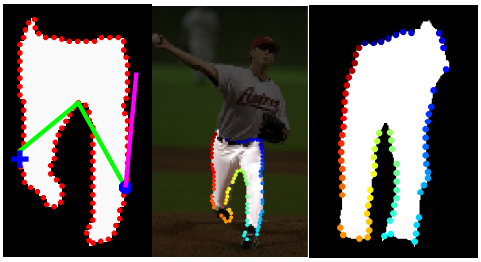

Figure 5. IDSC Computation. **Left:** We show: shortest interior path (green) from start (blue dot) to end (blue cross); boundary contour points (red); contour tangent at start (magenta). The length of interior path is the inner-distance; the angle between contour tangent and the start of the interior path is the inner-angle. **Center:** Lower body mask parse; colored points indicate correspondence established by IDSC matching with exemplar on **right**.

inner-angle between x_i and x_j is the angle between the contour tangent at the point x_i and tangent at x_i of the shortest path leading from x_i to x_j. Figure 5 shows the interior shortest path and contour tangent.

The inner-distances are normalized by the mean inner-distance between all pairs $\{(x_i, x_j)\}$, $i \neq j$ of points. This makes the IDSC scale invariant, since angles are also scale-invariant. The inner-angles and normalized log inner-distances are binned to form a histogram, the IDSC descriptor. For two shapes with points $x_1, ..., x_n$ and $y_1, .., y_n$, IDCSs are computed at all points on both contours. For every pair of points x_i, y_j, a matching score between the two associated IDCSs is found using the Chi-Square score ([1]). This forms an n-by-n cost matrix, which is used as input to a standard DP algorithm for string matching, allowing us to establish correspondence between the points on the two contours. The algorithm also permits occlusion of matches with a user-specified penalty. We try the alignment at several different, equally spaced starting points on the exemplar mask to handle the cyclic nature of the closed contours, and keep the best scoring alignment (and the score). Because the DP algorithm minimizes a cost (smaller is better), we multiply the score it returns by -1 to keep consistent with our desire to maximize F and all F_i. The complexity of the IDSC computation and matching is dominated by the matching; with n contour points and s different starting points, the complexity is $O(sn^2)$.

2.3. Parse Rule Application Procedure

Our parsing process consists of five basic steps that can be used to generate the parses for each node. For a particular node A, given all the parses for all children nodes, we perform the following steps:

Algorithm 1: P_A = Parse(A, S): for a particular image, given initial segments S and part name A, produce ranked and pruned parses for A.

Input: Part name A and initial shapes S
Output: P_A: set of ranked and pruned parses for A
$P_A = S$; `// Include all of` S `as parse candidates`
foreach *rule* $\{B_i, C_i\} \rightarrow A$ *(or* $B_i \rightarrow A$*)* **do**
  P_{B_i} = Parse(B_i, S); `// Recurse`
  P_{C_i} = Parse(C_i, S); `// If binary rule, recurse`
  $P_A = P_A \cup$ Group(P_{B_i}, P_{C_i}) (or Extend(P_{B_i})); `// Add to parses of A`
end
P_A = RankByShapeMatchingScore(P_A);
P_A = Prune(P_A); `// Prune redundant/low scoring parses`
return P_A; `// Return parses`

2.3.1 Parse rules

Segment inclusion: applies to all nodes We include by default all the masks in S as parses for A. This allows us to cope with an input image that is itself a silhouette, which would not necessarily be broken into different limbs, for example. A leg will often appear as a single segment, not as separate segments for the thigh and lower leg; it is easier to detect this as a single segment, rather than trying to split segments into two or more pieces, and then recognize them separately. For nodes in the parse tree with no children, this is their only source of masks.

Grouping: $\{B, C\} \rightarrow A$ For binary rules, we can compose parses from two children such as grouping two legs into a lower body, e.g. {Leg, Leg} $\rightarrow$ Lower body. For each child, based on the alignment of the best matching exemplar to the child, we can predict which part of the segment boundary is likely to be adjacent to another part.

A pair of masks, b from B and c from C, are taken if the two masks are within 30 pixels of each other (approximately 1/10th of the image size in our images), and combined with the pixel-wise OR operator. Because we need a single connected shape for shape comparison, if the two masks are not directly adjacent we search for a mask from the segmentations that is adjacent to both, and choose the smallest such mask m. m is then combined with b and c into a single mask with a single connected component. If no such mask exists, we just keep the larger of a and b. Figure 6 provides an example of the parse rule, {Leg,Leg} $\rightarrow$ LowerBody.

Extension: $\{B\} \rightarrow A$ For unary rules we generate parses by projecting an expected location for an additional part based on correspondence with exemplars. This is useful when bottom-up detection of a part by shape, such as

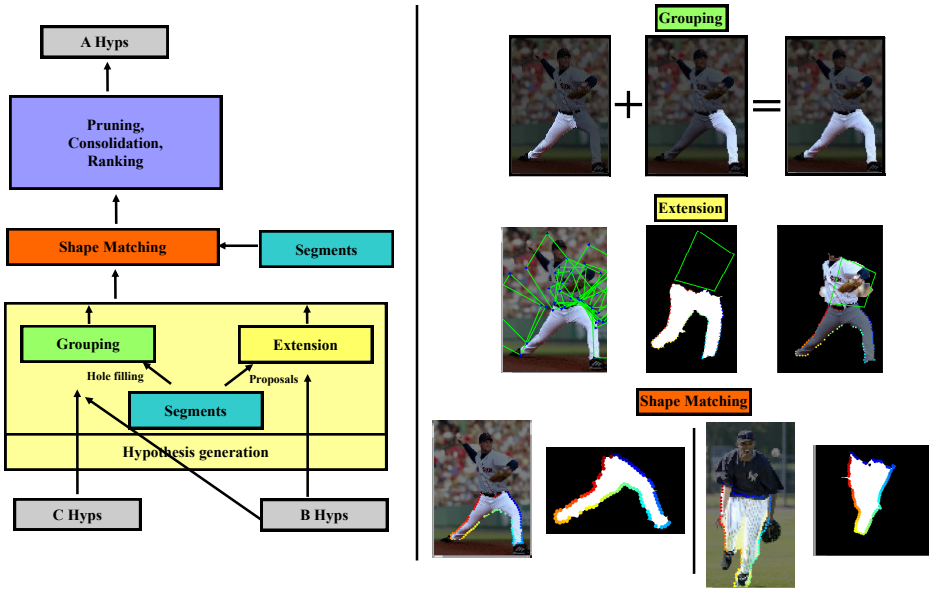

Figure 6. **Left:** parse rule application procedure. For binary rules, all pairs of child parses that are within 10 pixels of each other are composed via grouping, with hole filling provided by segments if needed. For unary rules, the child parses undergo extension using projected quadrilaterals and segment proposals. Shape matching is performed on both the original segments as well as the composed parses. For leaf nodes, shape matching is performed only on the segments. After shape matching, the parses are consolidated, pruned and ranked. **Right:** Grouping: two legs, on the left, are grouped into a lower body parse, on the right. Extension: the leftmost image shows a lower body parse with multiple different torso quadrilaterals projected from exemplars on to the image using the correspondence between the lower body parse and the lower body exemplars; the center image shows the exemplar with its torso quadrilateral that yielded the best torso parse, seen in the right image. Shape matching: two examples of shape matching. The lower body on the right was detected directly from the segments S, underscoring the importance of injecting the shapes from S into all levels of the parse tree.

the torso or head, is difficult due to wide variation of shape, or lack of distinctive shape. Once we have a large piece of the body (at least the lower body), it is more reliable to directly project a position for other parts. Given a parse of the lower body and its correspondence to a lower body exemplar shape, we can project the exemplar's quadrilateral representing the torso on to the parse (we estimate a transform with translation, rotation and scale based on the correspondence of two contour points closest to the two bottom vertices of the torso quadrilateral).

Similarly, given a mask for the lower body and torso, and its correspondence to exemplars, we can project quadrilaterals for the head. With these projected quadrilaterals, we look for all masks in S which have at least half their area contained within the quadrilateral, and combine these with the existing mask to give a new parse. For each parse/exemplar pair, we compose a new parse.

2.3.2 Complexity Control

Scoring Once parses have been composed, they are scored by matching to the nearest exemplar with IDSCs and DP. Correspondence is also established with the exemplar, providing an estimate of pose.

Pruning Many parses are either low-scoring or redundant or both. We prune away these parses with a simple greedy technique: we order the parses by their shape score, from highest to lowest (best to worst). We add the best parse to a representative set, and eliminate all other parse which are similar to the just added parse. We then recurse on the remaining parses until the representative set reaches a fixed size. For mask similarity we use a simple mask overlap score O between masks a and b:

$$O(a,b) = \frac{area(a \bigcap b)}{area(a \bigcup b)} \tag{6}$$

where $\bigcap$ performs pixel-wise AND, and $area(m)$ is simply the count of pixels with value 1 in the mask. If $O(a,b)$ is greater than a particular threshold, a and b are considered to be similar. After this step, we have a pruned set of parses that can be passed higher in the tree, or to evaluate in the end if the node A is the root. Figure 6 illustrates the stages of the parsing process for generating the parse for a single node. Also included are examples of grouping, extension, and shape matching/scoring.

Algorithm 1 sums up the parsing process for a particu-

lar part A, given intial set of shapes S from segmentation. It recursively generates parses for the children parts, and therefore to parse the torso+lower body+head (TLBH), we would call $Parse(TLBH, S)$. Note that if the part is a child in the parse tree, then no recursion occurs, and only the shapes S can form parses.

3. Results

We present results on the baseball dataset used in [9] and [8]. This dataset contains challenging variations in pose and appearance. We used 15 images to construct shape exemplars, and tested on $|I| = 39$ images. To generate the IDSC descriptors, we used the code provided by the authors of [7]. Boundary contours of masks were computed and resampled to have 100 evenly-spaced points. The IDSC histograms had 5 distance and 12 angle bins (in $[0, 2\pi]$). The occlusion penalty for DP matching of contours was 0.6 * (average match score), and 10 different alignments were used to initialize contour registration. For pruning, we used a threshold of 0.95 for the overlap score to decide if two masks were similar (a, b are similar $\iff O(a, b) \geq 0.95$) for the lower body+torso and lower body + torso + head, and 0.75 for all other pruning. In all cases, we pruned to 50 parses.

For parsing via grouping of parses from two different nodes, we can compose at most $50^2 = 2500$ parses. In practice, we typically found this to be between 500 and 1500 parses. For parsing via extension, for each of the 50 child parses, we create 15 new parses, 1 per exemplar, for a total of 750 parses. For each node, we examine an additional 390 parses from S. Given that there are 8 nodes, 2 extension relationships, and 5 grouping relationships, this gives an upper bound # of $2500 * 5 + 750 * 2 + 390 * 8 = 17120$ parses. With 15 exemplars, the number of shape comparisons is at most $15 * 17120 = 256800$.

Because we limit ourselves to shape cues, the best mask (in terms of segmentation and pose estimate) found by the parsing process is not always ranked first; although shape is a very strong cue, it alone is not quite enough to always yield a good parse. We expect that incorporating other cues would allow us to rank the best parse at, or very close to, the top. Our main purpose was to investigate the use of global shape features over large portions of the body via shape parsing. We evaluate our results in two different ways: segmentation score and projected joint position error. To the best of our knowledge, we are the first to present both segmentation and pose estimation results on this task.

3.1. Segmentation Scoring

We present our results in terms of an overlap score for a mask with a ground truth labeling. Our parsing procedure results in 50 final masks per image, ranked by their shape score. We compute the overlap score $O(m, g)$ between each mask m and ground truth mask g. We then compute the cumulative maximum overlap score through the 50 masks. For an image i with ranked parses $p_1^i, ...p_n^i$, we compute overlap scores $o_1^i, ..., o_n^i$. From these scores, we compute the cumulative maximum $C^i(k) = \max(o_1^i, ..., o_k^i)$. The cumulative maximum gives us the best mask score we can hope to get by taking the top k parses.

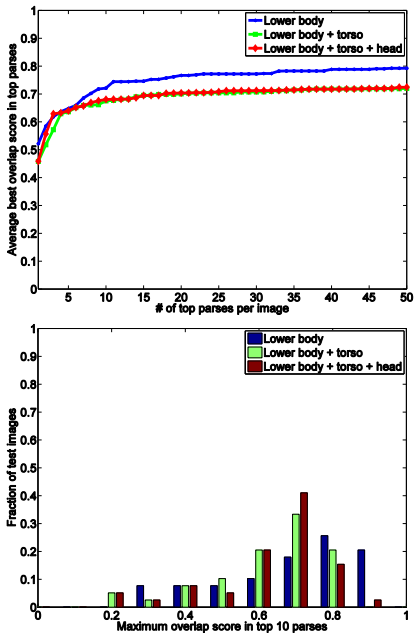

Figure 7. **Top:** We plot the average of each image's maximum overlap score as a function of the number of final parses retained, and do this for each region. **Bottom:** To give greater insight into the distribution of overlap scores, we focus on the top 10 parses, and histogram the best overlap score out of the top 10 for each image and region.

To understand the behavior of the cumulative maximum over the entire dataset, we compute $M(k) = \frac{1}{|I|}\sum_{i=1}^{|I|} C^i(k)$, or the average of the cumulative maximum over all the test images for each $k = 1, ..., n$ ($n = 50$ in our case). This is the average of the best overlap score we could expect out of the top k parses for each image. We consider this a measure of both precision and recall; if our parsing procedure is good, it will have high scoring masks (recall) when k is small (precision). On top in Figure 7, we plot $M(k)$ against k for three different types of masks composed during our parsing process: lower body, lower body+torso, and lower body + head + torso. We can see that in the top 10 masks, we can expect to find a mask that is similar to the ground truth mask desired, with similarity 0.7 on average. This indicates that our parsing process does a good job of both generating parses as well as ranking them.

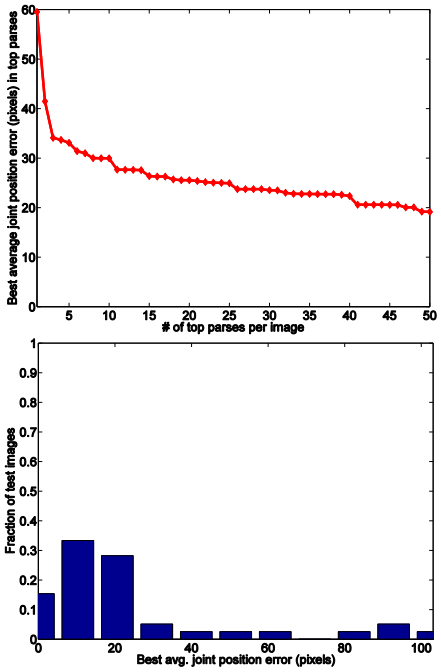

Figure 8. **Top:** We plot the average, across all images, of the minimum average joint error in the top k parses as a function k, the number of parses retained. **Bottom:** Taking the top 10 parses per image, for each image we compute the minimum average joint error from these top 10. We then histogram these values to show that taking 10 parses is likely to lead to recall of a good body parse. We can see that the vast majority of average errors are roughly 20 pixels or less.

While the above plot is informative, we can obtain greater insight into the overlap scores by examining all $C^i(k)$, $i = 1, ..., |I|$ for a fixed $k = 10$. We histogram the values of $C^i(10)$ on the bottom in Figure 7. We can see that most of the values are in fact well over 0.5, clustered mostly around 0.7. This confirms our belief that the parsing process is effective in both recalling and ranking parses, and that shape is a useful cue for segmenting human shape.

3.2. Joint Position Scoring

We also examinine the error in joint positions predicted by the correspondence of a parse to the nearest exemplar. We take 5 joints: head-torso, torso-left thigh, torso-right thigh, left thigh-left lower leg, right thigh-right lower leg. The positions of these joints are marked in the exemplars, and are mapped to a body parse based on the correspondence between the two shapes. For a joint with position j in the exemplar, we locate the two closest boundary contour points p, q in the exemplar that have corresponding points p', q' in the shape mask. We compute a rotation, scaling and translation that transforms p, q to p', q', and apply these to j to obtain a joint estimate j' for the parse mask. We compare j' with the ground truth joint position via Euclidean distance. For each mask, we compute the average error over the 5 joints. Given these scores, we can compute statistics in the same way as the overlap score for segmentation. On the top in Figure 8 we plot the average cumulative *minimum* $M(k)$, which gives the average best-case average joint error achieveable by keeping the top k masks. We see again that in the top 10 masks, there is a good chance of finding a mask with relatively low average joint error. On the bottom in Figure 8, we again histogram the data when $k = 10$.

Lastly, we show several example segmentations/registrations of images in Figure 9. Note that with the exception of the arms, our results are comparable to those of [8] (some of the images are the same), and in some cases our segmentation is better. As noted in [8], although quantitative measures may seem poor (e.g., average joint position error), qualitatively the results seem good.

4. Conclusion

In summary, we present a shape parsing method that constructs and verifies shapes in a bottom-up fashion. In contrast to traditional bottom-up parsing, our scoring functions at each node do not exhibit a SI property; instead, we score shapes against a set of exemplars using IDSCs, which convey global shape information over both small and large regions of the body. We also infuse the parsing process with multiple image segmentations as a pool of shape candidates at all levels, in contrast to typical parsing which only utilizes local image features at the leaf level.

We demonstrated competitive results on the challenging task of human pose estimation, on a dataset of baseball players with substantial pose variation, using only the cue of shape, while most other works use more cues. To the best of our knowledge, we are the first to present both quantitative segmentation and pose estimation results on this task. Note that in general, we need not start parsing with the legs only; it would be entirely feasible to add other nodes (e.g. arms) as leaves. A limitation of our method is that we have a fixed parsing procedure (starting from the lower body and going up); we will seek to remedy this in future work.

Further, we use larger shapes (composed of multiple body limbs) than typical pose estimation methods. Unlike most other related work, shape is our only cue. We expect that results would be improved with the introduction of color, texture and other cues. The notion of layers may also be useful in handling occlusion, as well as describing the shape relation of arms to the torso, since the arms often overlap the torso. Better grouping techniques (ones that introduce fewer parses) are a good idea, since this would save substantial computation (DP for contour alignment is expensive).

Figure 9. We present some of our body detection results. The segmentation of the person has been highlighted and the contour drawn as colored dots, indicating correspondence to the best matching exemplar. All the parses were the top scoring parses for that image (images are ordered row-major), with the exception of images 4 (2nd best), 8 (3rd best), 6 (3rd best). Some images were cropped and scaled for display purposes only. Full body overlap scores for each image (images are ordered row-major): 0.83, 0.66, 0.72, 0.74, 0.76, 0.70, 0.44, 0.57 and 0.84. Average joint position errors for each image: 12.28, 28, 27.76, 10.20, 18.87, 17.59, 37.96, 18.15, and 27.79.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 2002.

[2] E. Borenstein and J. Malik. Shape guided object segmentation. In *CVPR 2006*.

[3] T. Cour, F. Benezit, and J. Shi. Spectral segmentation with multiscale graph decomposition. In *CVPR 2005*.

[4] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *IJCV*, (1):55–79, January 2005.

[5] G. Hua, M.-H. Yang, and Y. Wu. Learning to estimate human pose with data driven belief propagation. In *CVPR 2005*.

[6] M. W. Lee and I. Cohen. Proposal maps driven mcmc for estimating human body pose in static images. *CVPR 2004*.

[7] H. Ling and D. W. Jacobs. Using the inner-distance for classification of articulated shapes. In *CVPR 2005*.

[8] G. Mori. Guiding model search using segmentation. In *ICCV 2005*.

[9] G. Mori, X. Ren, A. A. Efros, and J. Malik. Recovering human body configurations: combining segmentation and recognition. In *CVPR 2004*.

[10] D. Ramanan. Learning to parse images of articulated bodies. In *NIPS 2007*.

[11] D. Ramanan, D. A. Forsyth, and A. Zisserman. Strike a pose: Tracking people by finding stylized poses. In *CVPR 2005*.

[12] X. Ren, A. C. Berg, and J. Malik. Recovering human body configurations using pairwise constraints between parts. In *ICCV 2005*.

[13] R. Ronfard, C. Schmid, and B. Triggs. Learning to parse pictures of people. In *ECCV 2002*.

[14] L. Sigal and M. J. Black. Measure locally, reason globally: Occlusion-sensitive articulated pose estimation. In *CVPR 2006*.

[15] J. Zhang, J. Luo, R. Collins, and Y. Liu. Body localization in still images using hierarchical models and hybrid search. In *CVPR 2006*.