Learning to Parse Pictures of People

Remi Ronfard, Cordelia Schmid and Bill Triggs*

INRIA, 655 avenue de l'Europe, 38330, Montbonnot, France

Abstract Detecting people in images is a key problem for video indexing, browsing and retrieval. The main difficulties are the large appearance variations caused by action, clothing, illumination, viewpoint and scale. Our goal is to find people in static video frames using learned models of both the appearance of body parts (head, limbs, hands), and of the geometry of their assemblies. We build on Forsyth & Fleck's general 'body plan' methodology and Felzenszwalb & Huttenlocher's dynamic programming approach for efficiently assembling candidate parts into 'pictorial structures'. However we replace the rather simple part detectors used in these works with dedicated detectors learned for each body part using Support Vector Machines (SVMs) or Relevance Vector Machines (RVMs). We are not aware of any previous work using SVMs to learn articulated body plans, however they have been used to detect both whole pedestrians and combinations of rigidly positioned subimages (typically, upper body, arms, and legs) in street scenes, under a wide range of illumination, pose and clothing variations. RVMs are SVM-like classifiers that offer a well-founded probabilistic interpretation and improved sparsity for reduced computation. We demonstrate their benefits experimentally in a series of results showing great promise for learning detectors in more general situations.

Keywords: object recognition, image and video indexing, grouping and segmentation, statistical pattern recognition, kernel methods.

1 Introduction

Detecting people in images is an important practical challenge for content-based image and video processing. It is difficult owing to the wide range of appearances that people can have. There is a need for methods that can detect people in general everyday situations. For instance, actors in typical feature films are shown in a great variety of activities, scales, viewpoints and lightings. We can not rely on frequently-made simplifying assumptions such as non-occlusion, perfect background subtraction, *etc.*

To address this issue, Forsyth & Fleck introduced the general methodology of *body plans* [8] for finding people in images. However, they relied on simplistic body part detectors based on generalized cylinders. This is problematic, especially in the case of loose clothing. Similarly, Felzenszwalb & Huttenlocher [6] showed how dynamic programming could be used to efficiently group body plans cast as 'pictorial structures' [7], but they relied on simplistic colour-based part detectors. Both of these works make strong photometric assumptions about the body parts. We retain their ideas for composing parts into assemblies by building tree-structured models of people, but propose

* This work was supported by the European Union FET-Open research project VIBES

a more general approach to learning the body part detectors and the underlying geometric model, based on Support Vector Machines (SVM) [24,4] or Relevance Vector Machines (RVM) [22,23]. In the past, SVM classifiers have been learned for entire humans [18] and also for rigidly connected assemblies of subimages (typically, upper body, arms, and legs) [16], but not for flexibly articulated body models.

We present a series of experiments showing the promise of learning the articulated structure of people from training examples with hand-labelled body parts, using SVMs or RVMs. Our contribution is three-fold. Firstly, our feature set and training method builds reasonably reliable part detectors from as few as 100 hand-labelled training images, and the final RVM detectors are very efficient, often involving comparison with only 2–3 positive and 2–3 negative exemplars. Secondly, we sketch a method for learning a body joint model using the recently proposed Adaptive Combination of Classifiers (ACC) framework [16]. Thirdly, we describe an efficient decoder for the learned models, that combines kernel based detection with dynamic programming. Our initial experiments demonstrate that body part detectors learned with only 100 images from the MIT pedestrian database can give reliable detection with as few as 4 false alarms per image on this data set. This is remarkable as even humans often find it difficult to classify the isolated part subimages correctly. The detected parts can be efficiently assembled into correct body plans in 70% of cases.

The paper is structured as follows. We introduce our body plan model in §2, then discuss body part detectors learned by two competing algorithms, SVM and RVM, in §3. §4 presents our approach for learning and decoding body plans. Finally, §5 presents some results and discusses future work.

2 The Pictorial Structure of People

In the work of Marr & Nishihara [15] and others [10,19], people are described as hierarchical 3D assemblies of generalized cylinders and components. The position of a part C relative to its parent P is parametrized by C's position (p, r, θ) and angular orientation (ψ, ϕ, χ) in P's cylindrical coordinate system. Each joint is thus represented as a 6-vector, with discrete toleranced values for each parameter. They note that perspective projection makes many parameters unobservable and that the image signature of a joint is a pair of axes, but still emphasize, and attempt to recover, 3D structure.

Recovering articulated 3D models from single images is difficult. Felzenszwalb & Huttenlocher recently reconsidered Fischler & Elschlager's notion of *pictorial structure* [7] and demonstrated its usefulness for detecting people in indoor scenes [6]. Pictorial structures are collections of image parts arranged in deformable configurations. They are directly adapted to monocular observations. Similarly, Morris & Rehg argued that 3D tracking singularities can be removed using image based 'scaled prismatic models' [17] — essentially, pictorial structure models. Other 2D part-based models use image edges [25] or motion models derived from dense optical flow [14] as features for detection and/or tracking.

Following this line of research, we represent people using a 2D articulated appearance model composed of 15 part-aligned image rectangles surrounding the projections of body parts: the complete body, the head, the torso, and the left and right upper arms,

forearms, hands, thighs, calves and feet, numbered from 1 to 15 as in Figure 1. Each body part P_i is a rectangle parametrized in image coordinates by its centre $[x_i, y_i]$, its length or size s_i and its orientation θ_i. A coarse resolution whole-body image is also included in case 'the whole is greater than the sum of the parts'. During training and detection, we discretize the admissible range of sizes and orientations. As discussed later, we use a range of 8 scales, and 36 orientations equally spaced every 10 degrees. 14 body joints connect the parts: the plexus between body and torso, the neck between head and torso, the hips between torso and thighs, the knees between thighs and calves, the ankles between calves and feet, the shoulders between torso and upper arms, the elbows between upper arms and forearms and the wrists between forearms and hands. Figure 1 shows the body model in average position, using a single aspect ratio of 16:9 for all body parts.

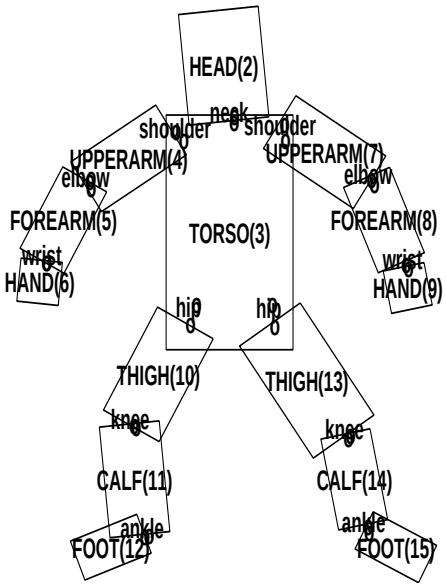

Figure 1. Our articulated body model with its 14 joints and 15 body parts.

Expressed in terms of the probabilistic formulation of pictorial structure, the posterior likelihood of there being a body with parts P_i at image locations l_i ($i \in \{1...15\}$) is the product of the *data likelihoods for the 15 parts* (*i.e.* the classification probabilities for the observed subimages at the given part locations to be images of the required parts) and the *prior likelihoods for the 14 joints* (*i.e.* the probabilities for a coherent body to generate an image with the given relative geometric positionings between each part and its parent in the body tree). The negative log likelihood for the whole body assembly $A = \{l_1, \ldots, l_{15}\}$ can thus be written as follows, where E is the list of body

joints ('edges' of the body tree):

$$L(A) \;=\; -\sum_i \log p_i(l_i) - \sum_{(ij)\in E} d_{ij}(l_i, l_j)$$

Felzenszwalb & Huttenlocher model body parts as constant color regions of known shapes and body joints as rotational joints. In this paper, we machine-learn the 29 distributions $p_i(l_i)$ and $d_{ij}(l_i, l_j)$ from sets of positive and negative examples. We model the part and articulation likelihoods using linear Support Vector or Relevance Vector Machines. Our work can be viewed as an extension of Mohan's recent work on *combined classifiers* [16], where 'component' classifiers are trained separately for the limbs, torso and head based on image pixel values, and 'combination' classifiers are trained for the assemblies based on the scores of the component classifiers in fixed image regions. However, we learn part-aligned, rather than image-aligned, classifiers for each body part, and we extend the 'combination' classifier to include deformable, articulated structures rather than rigid assemblies.

3 Detecting Body Parts

In our model, learning each body part amounts to estimating its probability given the observed image distribution at its location. Detecting and labelling body parts is a central problem in all component-based approaches. Clearly the image must be scanned at all relevant locations and scales, but there is also a question of how to handle different part orientations, especially for small, mobile, highly articulated parts such as arms and hands. One can work either in the image frame, trying to build a general detector that is capable of finding the part whatever its orientation, or in a part-aligned frame, building a detector that works for just one orientation and scanning this over all relevant orientations. The part-aligned approach has the potential to produce simpler detectors from less (but better labelled) training data, and the advantage that it also recovers the part orientation. Which approach is faster or better must depend on the relative complexity and reliability of all-orientation and one-orientation detectors, but in general it is difficult to build good transformation invariance into general-purpose detectors. The image-frame approach is well adapted to pedestrian detection applications such as Mohan's [16], where one wants a relatively coarse whole person detector for distant people with similar poses (mainly standing or walking). But our ultimate goal is to detect people and label them with detailed part locations, in applications where the person may be in any pose and partly occluded. For this we believe that the part-based body plan approach is preferable.

Our detector works with a generalized feature pyramid spanning 8 scales and 36 orientations $0° \ldots 350°$. During training, the articular structure of each training image is clicked, and for each designated part a 14×24 subimage aligned with its axes and scaled to its size is extracted as shown in Figure 2. We learn 15 Support Vector or Relevance Vector Machines for the individual parts and the whole body, and during detection run each of them over the scale-orientation-position feature pyramid, then assemble the results as discussed in the next section.

Figure 2. A hand-labelled training image from the MIT database and its extracted body part subimages. Reading vertically from left to right: left upper arm, forearm, hand; left thigh, calf and foot; head, torso and whole body; right thigh, calf, foot; right upper arm, forearm and hand.

3.1 Feature Sets

The problem of choosing features for object recognition has received a lot of interest in recent years and numerous feature sets have been suggested, including image pixel values, wavelet coefficients and Gaussian derivatives. Wavelets are currently popular, but as a general representation for human body parts it is unclear whether standard (rectangular) or non-standard (square) wavelet constructions are most suitable [9,16]. Heisele *et al* obtained better results for their SVM face detector using gray levels rather than Haar wavelets [9]. Some authors also feel that wavelets are unsuitable as a general image representation because they represent point events rather than line or curve ones, and instead propose ridgelets and curvelets [2,5]. These might prove useful for detecting human limbs.

Here we leave such issues for future work and use a feature set consisting of the Gaussian filtered image and its first and second derivatives. Although simple, these features seem to represent the variations of body part detail effectively over a range of scales and orientations. The feature vector for an image rectangle at location-scale-orientation $[x_i, y_i, s_i, \theta_i]$ contains the absolute values of the responses of the six Gaussian $\sigma = 1$ filters $\{G, \nabla_x G, \nabla_y G, \nabla_{xx} G, \nabla_{xy} G, \nabla_{yy} G\}$ in the rectangle's (rescaled and reoriented) 14×24 window. There are thus $14 \times 24 \times 6 = 2016$ features per window. For color images we use only the luminance values Y. The absolute values of the filter responses are normalized across each image. The extracted features are not required to be scale- or orientation-invariant. On the contrary, we seek features that are tuned to

the characteristic scales and orientations of the detail in the aligned body-part images. Some examples of the feature vectors are shown in Figure 3.

To implement this, the Gaussian filters are computed using 9 rotated images from 0 to 80 degrees and 8 scales. We resample according to scale in each window, so the standard deviation of the filters in their resampled 14×24 windows is always 1. For any given size and orientation, we select the feature vector that best approximates the part-aligned region as an axis-aligned rectangle of height 24. This choice of primitives makes reasonably few assumptions about the nature of the features to be learned, which can be arbitrary combinations of shape, luminance and texture.

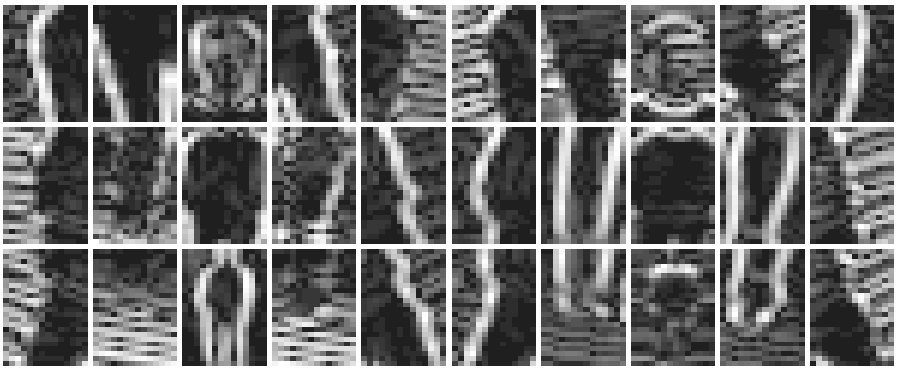

Figure 3. The $\nabla_x G$ and $\nabla_y G$ feature images for the example in Figure 2.

3.2 Training

Using the 2016-dimensional feature vectors for all body parts in the training set, we trained two linear classifiers for each part, one using a Support Vector Machine and the other using a Relevance Vector Machine. SVMs and RVMs are grounded on statistical learning results that suggest that they should give good classification performance even when there are relatively few training examples. Here we decided to put this claim to a severe test by training on the smallest sets of examples that give reasonable results — in our case, about 100.

We trained the 15 part classifiers individually against a common 'background' data set consisting of random pieces of the training images that do not contain people. Note that we are not attempting to learn isolated part detectors or multi-class part-type classifiers, but *reliable filters* for rejecting non-parts within an articulated body plan framework. We expect the overlap in appearance between different parts to be significant, but we do not want this to cause missed detections in ambiguous cases.

Support Vector Machines: SVMs are discriminant classifiers that give a yes/no decision, not a probability. However in our experiments we treat the SVM scores (scalar

products in feature space) as if they were log likelihoods for the body parts given the image values[1].

Relevance Vector Machines: RVMs [22,23] are Bayesian kernel methods that choose sparse basis sets using an 'Automatic Relevance Determination' [1] style prior that pushes non-essential weights to zero. They do not usually give significantly better error rates than the corresponding SVMs, but they do often give similar results with many fewer kernels. The functional form of the final classifier is the same as that of an SVM — only the fitted weights are different. Here we use logistic linear discriminant RVMs, whose output directly models the log-odds for a part versus a non-part at the given point. In this paper, we use RVMs mainly to reduce the number of kernels ('relevance vectors') and hence the computational complexity. The trained RVM classifiers typically use only 2–3 positive and 2–3 negative relevance vectors each, as compared to 100–200 support vectors for a comparable SVM classifier.

Currently we train the linear RVMs to make sparse use of *examples*, but they could also be trained to make sparse use of *features*. This would potentially mean that fewer image features would have to be extracted, and hence that the method would run faster. We plan to investigate this in future work.

3.3 Detection

We detect all of the body parts at once, in a single scan over the orientation-scale pyramid. The detection score for each part reduces to a simple convolution product against a mask containing the discriminant sum of weighted support or relevance vectors. Conceptually, this amounts to generalized template matching over images of local feature vectors, with weighted sums of training examples as templates. The nonlinearity of the process is hidden in the rectified, normalized local feature vectors. For efficiency in the assembly stage, we currently retain only the 50 best candidates for each part. The observed detection rates suggest that this strategy suffices for simple images, but it is not ideal for robustness against occlusions and we ultimately plan to use a more sophisticated strategy based on adaptive thresholds.

4 Parsing the body tree

In a non-articulated, image-aligned method such as that of Mohan [16], assembling the part detections is relatively straightforward: decompose the search window into subwindows, keep the highest score for the appropriate part in each subwindow, and compose the scores into a single, low-dimensional feature vector. Given these second-stage feature vectors, a single linear SVM can be learned for the overall body detection.

In our articulated, part-aligned method, the composition of part-models is only slightly more difficult, and can be cast as a combinatorial search: from all detected

[1] A more principled approach to converting the scores of a discriminant classifier to probabilities is as follows: run the detector over a validation set and fit density models to its positive-example and negative-example output scores. At any given score, the ratio of the positive-example density to the negative-example one is an estimate of the positive-to-negative odds ratio for detections at that score.

parts, search for the assemblies looking most like people. Since assemblies are naturally described as trees, efficient dynamic programming algorithms can be used to build the second-stage classifier, as we now describe.

4.1 Parsing/decoding algorithm

Given N candidate body part locations l_{kn} detected by each body part classifier C_k, we are looking for a 'parse' of the scene into one or more 'body trees'. One important subproblem is to assign a 'valid detection' or 'false alarm' label to each candidate, based not only on the candidate's scores, but on the local configuration between the candidates and its neighbours. Our approach relies on an extension of the Viterbi decoding algorithm, as described by Ioffe & Forsyth [13] and Felzenszwalb & Huttenlocher [6], which we sketch only briefly here. Given the detection scores $D_k(l_{kn})$ for all candidates $n = 1...N$, we search for the best candidate as a function of their direct parents $pa(n)$ in the body tree. For the leaves (i.e. hands, feet and head), this is computed by algorithm 1:

Algorithm 1 leaf location

$B_k(l_{jm}) = \min_{\{n=1...N\}} -D_k(l_{kn}) + d_{kj}(l_{kn}, l_{jm})$
$l_k^*(l_{jm}) = \arg\min_{\{n=1...N\}} -D_k(l_{kn}) + d_{kj}(l_{kn}, l_{jm})$

Based on this computation, we can score candidates from the bottom up, using the recursion algorithm 2:

Algorithm 2 bottom up

$B_k(l_{jm}) = \min_{\{n=1...N\}} -D_k(l_{kn}) + d_{kj}(l_{kn}, l_{jm}) + \Sigma_{\{c|k=pa(c)\}} B_c(l_{kn})$
$l_k^*(l_{jm}) = \arg\min_{\{n=1...N\}} -D_k(l_{kn}) + d_{kj}(l_{kn}, l_{jm}) + \Sigma_{\{c|k=pa(c)\}} B_c(l_{kn})$

At the root node we obtain the simple formula 3 for scoring the high level hypotheses.

Algorithm 3 root location

$B_r = \min_{\{n=1...N\}} -D_r(l_{rn}) + \Sigma_{\{c|r=pa(c)\}} B_c(l_{rn})$
$L_r^* = \arg\min_{\{n=1...N\}} -D_r(l_{rn}) + \Sigma_{\{c|r=pa(c)\}} B_c(l_{rn})$

Choosing the most probable root node, we can then assign the other nodes in a top down fashion by choosing $L_k^* = l_k^*(L_{pa(k)})$ for each node given its parent. Note that this algorithm has a complexity $O(MN^2)$ with M the number body parts and N the number of candidates per body part. As an example of the detection results obtained with this method, Figure 6 shows the three most probable parses for four test images, ranked in order of decreasing likelihood.

4.2 Learning the body tree

The cost functions used in our body tree model are based on geometric constraints on the relative positions of parts at a body articulation, as in Felzenszwalb & Huttenlocher [6]. Essentially, the articulation model is a linear combination of the differences between two joint locations, as predicted separately by the two body parts meeting at the articulation.

Algorithm 4 joint distance(l_i, l_j)

Compute joint location x_{ij}, y_{ij} given first body part location l_i
Compute joint location x_{ji}, y_{ji} given second body part location l_j
Return $d_{ij} = w^x_{ij}|x_{ij} - x_{ji}| + w^y_{ij}|y_{ij} - y_{ji}| + w^\theta_{ij}|\theta_i - \theta_j - \theta_{ij}| + w^s_{ij}|\log \frac{s_i}{s_j} - \log s_{ij}|$

Each body joint is parametrized by the relative sizes s_{ij} and angles θ_{ij} between its parts, and the four rigidity parameters $w^x_{ij}, w^y_{ij}, w^\theta_{ij}, w^s_{ij}$ governing the admissible range of apparent deformations of the articulation in position, size and orientation. We learned the relative sizes s_{ij} and angles θ_{ij} of each articulation by simply taking the average relative positions of all pairs of body parts over the training set.

To learn the rigidity parameters, we again used a Support Vector Machine. For each articulation A_{ij} between parts P_i and P_j, we learned a 'combination classifier' based on a five-dimensional feature vector $F^0_i = D_i + D_j$, $F^x_i = |x_{ij} - x_{ji}|$, $F^y_i = |y_{ij} - y_{ji}|$, $F^\theta_i = |\theta_i - \theta_j - \theta_{ij}|$, $F^s_i = |\log \frac{s_i}{s_j} - \log s_{ij}|$.

Using positive and negative examples from our training set, we used a linear SVM classifier to learn a set of weights $w^0_{ij}, w^x_{ij}, w^y_{ij}, w^\theta_{ij}, w^s_{ij}$ such that the score is positive for all positive example, and negative for all negative examples. We experimentally verified that the learned weights have the expected signs, $w^0_{ij} > 0$ and $w^x_{ij} < 0, w^y_{ij} < 0, w^\theta_{ij} < 0, w^s_{ij} < 0$, so that the learned model can indeed be related to the log-likelihood of the articulation

$$L(A_{ij}) = F^0_i - \frac{|w^x_{ij}|}{w^0_{ij}} F^x_i - \frac{|w^y_{ij}|}{w^0_{ij}} F^y_i - \frac{|w^\theta_{ij}|}{w^0_{ij}} F^\theta_i - \frac{|w^s_{ij}|}{w^0_{ij}} F^s_i$$

In our experiments with the MIT pedestrian database, the learned models performed slightly better than the naive approach of assigning equal weights to all parameters and all articulations, and we expect the method to be of even greater benefit for dealing with the more complicated cases of people in action such as running or jumping.

5 Implementation and results

We implemented and tested our method in Matlab. The system consists of several components. There is an interactive program for hand-labelling examples and storing the locations of the body joints and parts. Another function computes image pyramids and extracts image signatures at all locations x, y, s, θ . These are used both to generate feature vectors for SVM/RVM training, and to perform detection against the learned models. Finally, a parser based on the above dynamic programming approach reads

candidate locations from the 15 body part detectors and produces a ranked list of candidate assemblies.

We used MIT's public domain program SvmFu-3.0 to train the SVM classifiers. We trained the RVM classifiers in Matlab using a new algorithm that will be described in detail elsewhere.

5.1 Experimental setup

We selected 100 frontal images from the MIT pedestrian database and labelled their 15 parts, as shown in Figure 2. Each example is labelled by clicking 14 body joints. Occluded parts are clicked at their most likely (hidden) location, but flagged as occluded. Only visible parts are used to train the image part models, but the hidden parts can be included when training the geometric models. We also picked 5 background regions in each image, for use as negative examples. As a result, each body part classifier was trained with slightly less than 100 positive examples, and 500 negative examples.

Separate examples are needed for training and testing, so we selected and labelled another 100 images from the MIT pedestrian database to serve as a test set. This was used to evaluate the body part and assembly detectors.

5.2 Detection of body parts.

Detectors are traditionally compared by tracing ROC curves, i.e. true detection rates (recall) as a function of false alarm rates ($1-$precision). In our case the detectors must be tuned to function as filters, so most important parameter is the false alarm rate needed to achieve 'total recall'. Hence, we compared the two detectors by measuring the false detection rates required to detect all visible body parts in our test set. The resulting true positive rates for each part detector are shown in Figure 4.

As can be seen, individual part images are not very discriminative, so the absolute false alarm rates remain quite high. In fact, they become still higher (up to 15:1) once confusions between parts are included. Even so, the linking stage manages to resolve most of the ambiguity, and the number of candidates that have to be examined remains quite tractable, at most about 75 candidates per part for these images. Ignoring spatial contiguity, the worst-case number of body joint hypotheses is therefore $14 \times 75^2 = 78750$. In practice, we observed an average number closer to $14 \times 20^2 = 5600$ and used 50 candidates as a safe bet in all of our experiments. The RVM classifiers perform only slightly worse than their SVM counterparts, with mean false detection rates of 80.1% and 78.5% respectively. This is remarkable given the very small number of relevance vectors used by the RVM detectors. For the purpose of rapid filtering, the advantages of the RVM clearly outweigh their inconvenience.

Also note that the worst results are obtained for the torso (3) and head (2) models. The torso is probably the hardest body part to detect as it is almost entirely shapeless. It is probably best detected indirectly from geometric clues. In contrast, the head is known to contain highly discriminant features, but the training images contain a wide range of poses and significantly more training data (and perhaps some bootstrapping on false alarms) is probably needed to build a good detector.

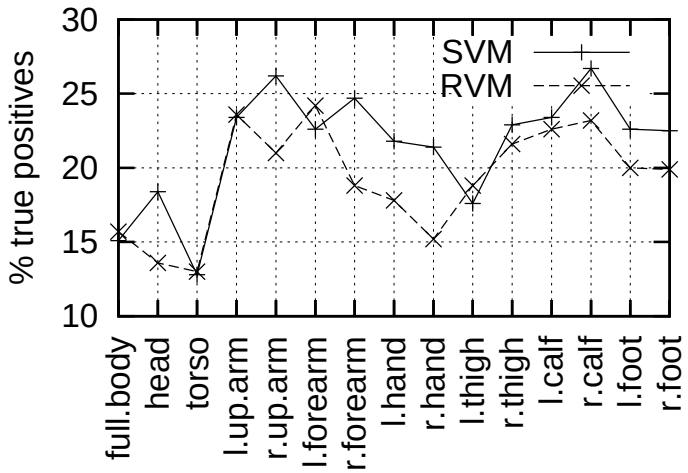

Figure 4. True positive rates for SVM and RVM body part detectors.

5.3 Detection of body trees

We evaluated the final body detector by visually comparing the best (highest probability) three configurations returned with the correct interpretation in each of the 100 test set images. Thus, the task was purely that of detecting humans using the 50 best candidates for each body part and the body tree model. Our first experiment used 100 training exemples. We obtained correct detections rates of 72 % using RVM scores and 83 % using SVM scores, while using a naive geometric model with uniform rigidity parameters for all of the body joints. We then learned a geometric model using labelled body joints from the 100 training images. We used the correct assemblies as positive examples, and circular permutations of the body parts as negative ones. Using the learned model, the correct detection rates improved to 74 % and 85 %. We should note that detection is a relatively easy task with this data set, and our method should be evaluated also with regards to the pose estimates. We plan to investigate this area quantitatively in later work. Qualitatively, we noted that a majority of the body parts were correctly positioned in only 36 % of the test images for RVM and 55 % for SVM.

In a second experiment, we increased the size of the training set to 200 examples. This resulted in a slight increase of the detection rates, to 76 % for SVM and 88 % for RVM, and a much vaster improvement of the pose estimates, resulting in qualitatively correct poses in 54 % of the test examples for RVM and 75 % for SVM.

6 Discussion and Future Work

The good detection rates achieved by the method make a convincing case that the body-plan strategy is applicable to real problems in image and video indexing. We plan to extend this work to video, where we hope to improve the detection rates even further

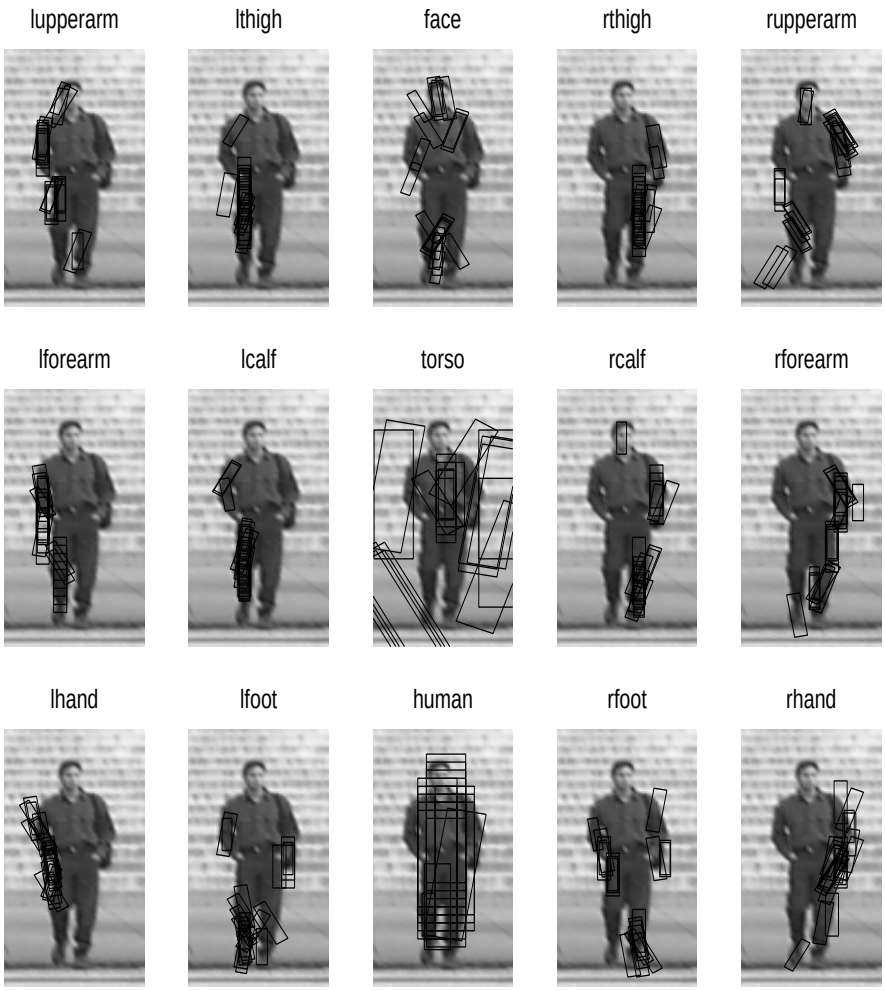

Figure 5. Part detection results from test collection.

by making use of temporal and kinematic constraints. But the construction of the image pyramid is computationally expensive, and we plan to move to a more efficient implementation, which could rely on a more thorough selection of the feature vectors. One way to do this will be to use RVM classifiers that learn relevant features rather than relevant examples. As a complement, Sidenbladh & Black's [20,21] approach for learning the image statistics of people vs. background could prove useful for learning better models by selecting better features. In the assembly phase, the complexity of the dynamic programming algorithm is quadratic n the number of candidate parts which need to be stored, which in turn depends on the precision of the individual body part detectors. By fine-tuning the body part detectors, we expect to achieve significant improvements also in the overall performance of the global detector.

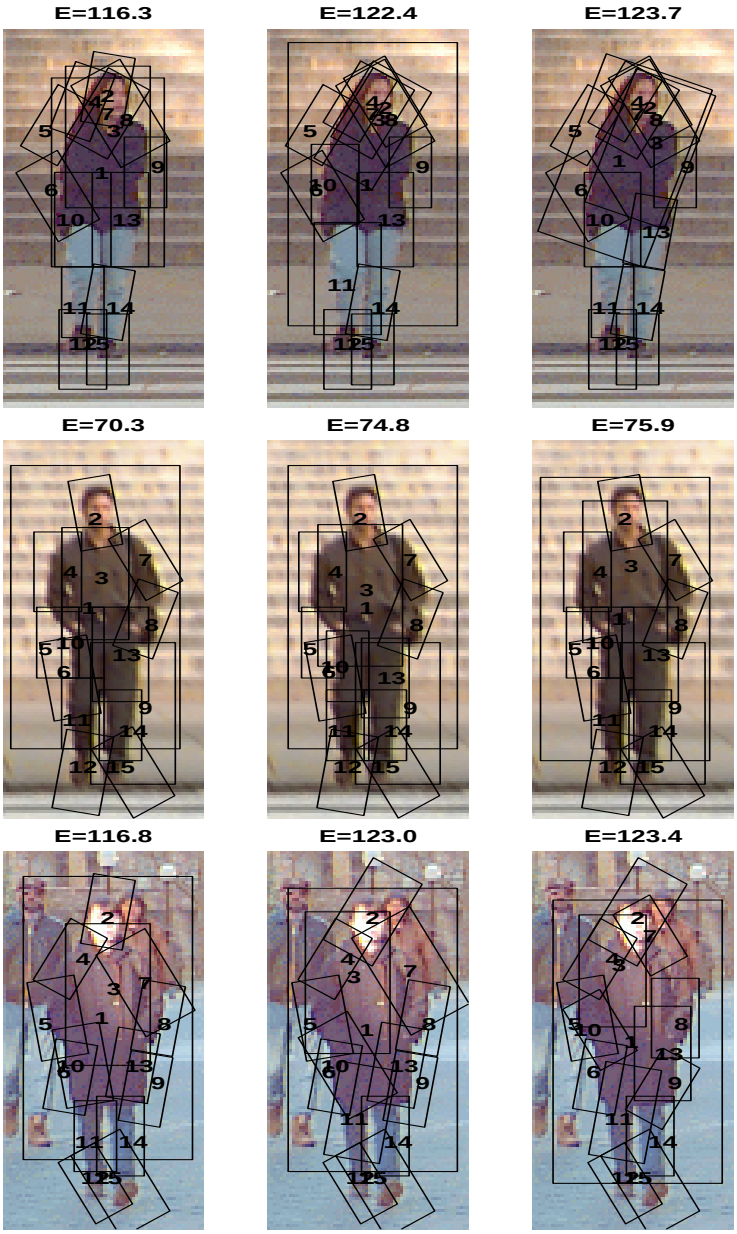

Figure 6. Ranked detections and their energies, using the learned body model and SVM scores.

Further work will be needed for assessing the correctness of the detection and pose estimation results in a more systematic way and for 'bootstrapping' the learned models (adding examples on which our current model fails, and retraining). Even without boot-

strapping, we have verified experimentally that the quality of the body part classifiers is improved significantly by increasing the size of the training data. We will need to quantify this observation in future work.

We also plan to extend the method to handle multiple persons in a greater variety of backgrounds and poses, by explicitly representing occlusions in the decoding process as in the work of Coughlan et al. [3] or by introducing mixtures of partial body trees, as in the recent proposal made by Ioffe and Forsyth [11,12]. The cost functions used to evaluate the assembly of the body plans could also benefit from a richer geometric model and additional photometric constraints (e.g. similarity of color and texture between the body parts for the same person). There are cases where we would like to move even further away from the human anatomic model, and replace it with a small set of 'clothing models', which could be learned in much the same way and provide additional flexibility. Those are avenues for further experimental work.

7 Conclusion

Detecting humans is a challenging problem in computer vision, with considerable practical implications for content-based indexing. We believe we have reached three useful concusions with the work reported in this paper. Firstly, it is possible to learn appearance models for human body parts from examples and to use them as input to a body plan parser, at least for a modest-size problem such as pedestrian detection. Secondly, we have been able to learn geometric models for the combination of the detected parts, allowing us to robustly estimate the likelihood of a body part assembly, without recourse to sampling or HMM distributions, which require thousands of examples to be learned efficiently. Thirdly, the learned models lead to an efficient decoding algorithm that combines kernel based learning and dynamic programming techniques, and is simple enough to be extended to video sequences.

References

1. C. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, 1995.
2. E. Candes and D. Donoho. Curvelets – a surprisingly effective nonadaptive representation for objects with edges. In L. L. Schumaker et al., editor, *Curves and Surfaces*. Vanderbilt University Press, 1999.
3. J. Coughlan, D. Snow, C. English, and A. Yuille. Efficient optimization of a deformable template using dynamic programming. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2000.
4. N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and other Kernel Based Learning Methods*. Cambridge University Press, 2000.
5. M. Do and M. Vetterli. Orthonormal finite ridgelet transform for image compression. In *Int. Conference on Image Processing*, volume 2, pages 367–370, 2000.
6. Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient matching of pictorial structures. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2000.
7. M. Fischler and R. Elschlager. The representation and matching of pictorial structures. *IEEE Trans. Computer*, C-22:67–92, 1973.

8. D. Forsyth and M. Fleck. Body plans. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1997.
9. B. Heisele, T. Poggio, and M. Pontil. Face detection in still gray images. Technical report, AI Memo 1687, Massachusetts Institute of Technology, 2000.
10. D. Hogg. Model-based vision: A program to see a walking person. *Image and Vision Computing*, 1(1):5–20, 1983.
11. Sergey Ioffe and David Forsyth. Human tracking with mixtures of trees. In *Proc. Int. Conf. Computer Vision*, 2001.
12. Sergey Ioffe and David Forsyth. Mixtures of trees for object recognition. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2001.
13. Sergey Ioffe and David Forsyth. Probabilistic methods for finding people. *Int. J. of Computer Vision*, 43(1), 2001.
14. S. Ju, M. Black, and Y. Yacoob. Cardboard people: a parameterized model of articulated image motion. In *Int. Conference on Automatic Face and Gesture Recognition*, 1996.
15. D. Marr and H.K. Nishihara. Representation and recognition of the spatial organization of three dimensional structure. *Proceedings of the Royal Society of London B*, 200:269–294, 1978.
16. Anuj Mohan, Constantine Papageorgiou, and Tomaso Poggio. Example-based object detection in images by components. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 23(4), 2001.
17. D. Morris and J. Rehg. Singularity analysis for articulated object tracking. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1998.
18. C. Papageorgiou. Object and pattern detection in video sequences. Technical report, Master's thesis, Massachusetts Institute of Technology, 1997.
19. K. Rohr. Incremental recognition of pedestrians from image sequences. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 8–13, 1993.
20. H. Sidenbladh and M. Black. Learning the statistics of people in images and video. *Int. Journal of Computer Vision*, 2001.
21. H. Sidenbladh, F. Torre, and M. Black. A framework for modeling the appearance of 3d articulated figures. In *Int. Conference on Automatic Face and Gesture Recognition*, 2000.
22. M. Tipping. The relevance vector machine. In *Advances in Neural Information Processing Systems*. Morgan Kaufmann, 2000.
23. M. E. Tipping. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, 2001.
24. V. Vapnik. *Statistical Learning Theory*. Wiley, 1998.
25. Liang Zhao and Chuck Thorpe. Recursive context reasoning for human detection and parts identification. In *IEEE Workshop on Human Modeling, Analysis and Synthesis*, 2000.