Pose Search: retrieving people using their pose

Vittorio Ferrari
ETH Zurich
ferrari@vision.ee.ethz.ch

Manuel Marín-Jiménez
University of Granada
mjmarin@decsai.ugr.es

Andrew Zisserman
University of Oxford
az@robots.ox.ac.uk

Abstract

We describe a method for retrieving shots containing a particular 2D human pose from unconstrained movie and TV videos. The method involves first localizing the spatial layout of the head, torso and limbs in individual frames using pictorial structures, and associating these through a shot by tracking. A feature vector describing the pose is then constructed from the pictorial structure. Shots can be retrieved either by querying on a single frame with the desired pose, or through a pose classifier trained from a set of pose examples.

Our main contribution is an effective system for retrieving people based on their pose, and in particular we propose and investigate several pose descriptors which are person, clothing, background and lighting independent. As a second contribution, we improve the performance over existing methods for localizing upper body layout on unconstrained video.

We compare the spatial layout pose retrieval to a baseline method where poses are retrieved using a HOG descriptor. Performance is assessed on five episodes of the TV series 'Buffy the Vampire Slayer', and pose retrieval is demonstrated also on three Hollywood movies.

1. Introduction

The objective of this work is to retrieve those shots containing a particular human pose from a database of videos. We are interested in unconstrained video, such as movies or TV shows, and specify pose as a 2D spatial configuration of body parts. Often a pose, or pose sequence, characterizes a person's attitude or action.

Being able to search video material by pose provides another access mechanism over searching for shots containing a particular object or location [32], person [3, 22, 31], action [5, 19], object category or scene category (e.g. indoors/outdoors). We demonstrate in this work that from a single query frame we can retrieve shots containing that pose for different people, lighting, clothing, scale and backgrounds.

That pose retrieval is possible at all in uncontrolled video is due to the considerable progress over the past few years in detecting humans [9, 21] and human layout in still images [12, 23, 24, 28] and video [13, 27]. For human layout, early innovations succeeded for naked humans on uncluttered backgrounds [15], or by suppressing clutter by background subtraction [20]. With the addition of better segmentation [7, 29], better features, e.g. HOG [9], and more efficient inference on trees [12], humans and their limb configuration could be localized even with unknown clothing, in odd poses [24], and against very challenging backgrounds [13].

We investigate pose retrieval on two parallel threads. The first is the most flexible: we localize the 2D spatial layout of body parts for the upper body of all humans in the video (head, torso, lower and upper arms). Our approach is based on [13], and is aimed at near-frontal and near-rear poses (i.e. no profile views). We improve on this method, as described and quantified in section 2. Given this spatial layout, in section 3, we define three pose descriptors and associated similarity measures and compare their performance for pose retrieval. A query pose can be specified by a single example or a set of examples. In the latter case we first train a linear classifier on the provided examples, and compare performance between the two cases.

The second thread starts with a simple upper body human detector, and then computes a pose descriptor using HOG [9] (section 4). This provides a baseline comparison with the layout thread, and in a similar manner both query on a single pose and training a classifier are compared.

The methods we develop, representing the 2D spatial configuration, are complementary to the recent work on recognizing actions using low level spatio-temporal features [5, 10, 19, 25] or intermediate level features (sets of low level spatio-temporal features) [11], and can provide the starting point for action recognition using 2D silhouettes [14] or motion history images [6].

2. Human pose estimation

The task of pose estimation is to recover the articulated 2D pose of all visible persons in every video frame, as the

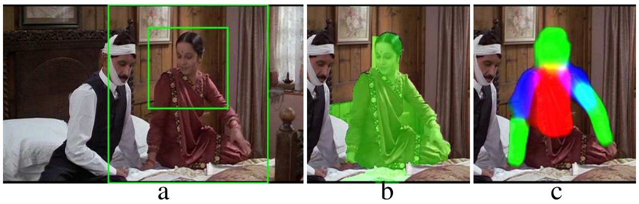

Figure 1. **Pose estimation.** (a) Input frame with detected upper-body (inner rectangle), and enlarged region for further processing (outer rectangle). (b) Foreground highlighting removes part of the background clutter. (c) Estimated pose, including our improvements over [13]. The color coding is: red = torso, blue = upper arms and head, green = lower arms. Color planes are superimposed, e.g. yellow indicates either lower arm and torso, or torso and head.

location, orientation and scale of their body parts. In this section we review our method for pose estimation [13] (subsection 2.1), and then describe two extensions to improve its performance (subsection 2.2). The explanations focus on the upper-body case, which has $N = 6$ body parts, but the method is applicable to full bodies as well (see section 4 in [13]).

2.1. The base method of Ferrari *et al* [13]

Our pose estimation system [13] enables fully automatic 2D human pose estimation in uncontrolled video, such as TV shows and movies. Direct pose estimation on this uncontrolled material is often too difficult, especially when knowing nothing about the location, scale, pose, and appearance of the person, or even whether there is a person in the frame or not. Therefore, in [13] we decompose the problem by first determining the approximate location and scale of the person in each frame using an upper body detector, and then applying a pose estimation method based on fitting pictorial structures (figure 1a). After applying the upper-body detector to every frame in the shot independently, the resulting detections are associated over time into *tracks*, each connecting the detections of the same person throughout a shot.

We now describe our base method [13] in more detail, as we will improve on it below. Following **upper body detection**, an area of quite some size needs to be searched for body parts (it needs to cover out-stretched arms for example). This *enlarged region* is derived from the detection as shown in figure 1a. Within this region **foreground highlighting** is applied by using GrabCut [29] to determine a foreground area where the human may be, and exclude part of the background clutter (figure 1b), and thus ease the task of subsequent model fitting. GrabCut requires an initialization for the foreground and background regions. They are provided in [13] as the portions of the enlarged region likely to contain the head and torso (for foreground) and away from this area for background.

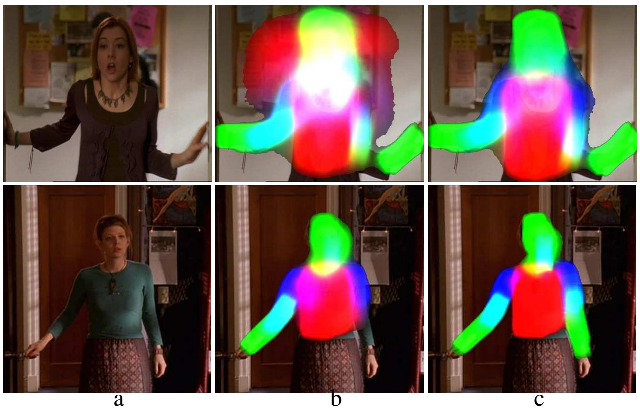

Figure 2. **Improved pose estimation.** (a) Input frame. (b-top) Pose estimated using base method [13]. (c-top) The improvement by adding position priors. (b-bottom) Pose estimated using the kinematic tree model of the base method. (c-bottom) The improvement by adding the the repulsive model.

The next stage, **parsing**, fits a pictorial structure model [12], restricting the search to the foreground area. We use the method of Ramanan [26], which captures the appearance and spatial configuration of body parts. A person's body parts are tied together in a tree-structured conditional random field. Parts, l_i, are oriented patches of fixed size, and their position is parameterized by location (x, y) and orientation θ. The posterior of a configuration of parts $L = \{l_i\}$ given an image I is

$$P(L|I) \propto \exp\left(\sum_{(i,j)\in E} \Psi(l_i, l_j) + \sum_i \Phi(l_i|I)\right) \qquad (1)$$

The pairwise potential $\Psi(l_i, l_j)$ corresponds to a spatial prior on the relative position of parts and embeds the kinematic constraints (e.g. the upper arms must be attached to the torso). The unary potential $\Phi(l_i|I)$ corresponds to the local image evidence for a part in a particular position (likelihood). As the model structure E is a tree, inference is performed exactly by sum-product Belief Propagation.

Since the appearance of the parts is initially unknown, a first inference uses only edge features in Φ. This delivers soft estimates of body part positions, which are used to build appearance models of the parts and background (color histograms). Inference in then repeated with Φ using both edges and appearance. This *parsing* technique simultaneously estimates pose and appearance of parts.

For each body part, parsing delivers a posterior marginal distribution over location and orientation (x, y, θ) (more details in section 3.1).

2.2. Improvements over the base method

We present here two extensions to [13] and a quantitative evaluation demonstrating how they improve pose estimation performance. Better pose estimates are desirable, because we expect they will in turn lead to better pose retrieval.

Position priors. The implicit assumption exploited by [13] is that people appear *upright* in the image. This underpins the design of the helpful preprocessing stages such as upper-body detection and foreground highlighting.

Here, we present an additional way to take advantage of the upright assumption. We extend the model (1) by adding priors $\Upsilon(l_{head}), \Upsilon(l_{torso})$ requiring the orientation of the torso and head to be near-vertical. $\Upsilon(\cdot)$ gives uniform probability to a few values of θ around vertical, and zero probability to other orientations. This further reduces the search space for torso and head, thus improving the chances that they will be correctly estimated (figure 2-top). Moreover, it also benefits the pose estimation for the arms, because the torso induces constrains on their position through the kinematic prior Ψ.

Repulsive model. A well-known problem with pictorial structure models is that different body parts can take on similar (x, y, θ) states, and therefore cover the same image pixels. Typically this happens for the left and right lower arms, when the image likelihood for one is substantially better than the likelihood for the other. It is a consequence of the model being a *tree*, assuming conditional independence between the left and right arms. This is referred to as the *double-counting problem* and has been noted by other authors [30]. One solution, adopted in previous work, is to explicitly model limb occlusion by introducing layers into the model [2, 17, 30], though the graphical model is then no longer a tree.

Here, in order to alleviate the double-counting problem we propose a simple and effective method (figure 2-bottom). We add to the kinematic tree model two *repulsive edges*, connecting the left upper arm to the right upper arm, and the left lower arm to the right lower arm. Again, the model is no longer a tree. These new edges carry a repulsive prior $\Lambda(l_i, l_j)$ which gives lower probability when parts l_i and l_j overlap than when they don't. Therefore, the extended model *prefers* configurations of body parts where the left and right arms are not superimposed, but it does not forbid them. If the image evidence in their favor is strong enough, inference will return configurations with overlapping arms. This properly reflects our prior knowledge that the arms occlude each other in a minority of images. Approximate inference in the extended graphical model is performed with sum-product Loopy Belief Propagation [4].

Impact on performance. The qualitative benefit of the two improvements are illustrated in figure 2. A quantitative evaluation is obtained using the stickmen annotated data [1] and the performance measure given in [13]. The original algorithm of [13] correctly estimates 56% of the 1458 body parts in this dataset. Using the position priors above brings performance to 65.9%, and this rises to 69.5% with small additional corrections to the kinematic prior, and to 72.2% by using also the proposed repulsive model. This is a substantial improvement over [13]. Note, we perform pose estimation in every frame independently, as we do not transfer appearance models between frames, nor use temporal priors (both are used in [13]).

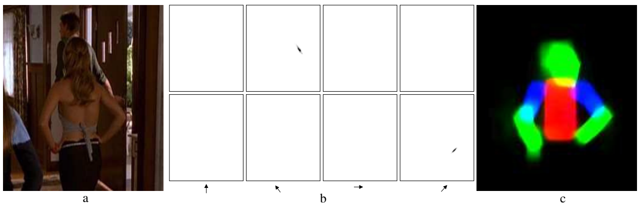

Figure 3. **Detailed pose estimate.** (a) Input frame (cropped to the enlarged region, as in figure 1a). (b) Estimated pose for right upper arm (RUA, top) and right lower arm (RLA bottom). Each row shows the posterior marginal $P(l_i = (x, y, \theta))$ as a function of (x, y) for four values of θ (out of 24). (c) Visualization obtained by convolving rectangles representing body parts, with their corresponding posterior.

3. Articulation-based Pose Retrieval

In this section we present our pose retrieval approach, which is based on the articulated pose estimates from section 2. Later, in section 4, we explore an alternative system based on simpler, lower level features (HOG).

We define *pose retrieval* as the task of retrieving shots containing any person in a given pose from a (possibly large) database of videos (*retrieval database*). Analogous to image retrieval the user can specify the target pose by selecting a single frame containing it. This query frame is not required to belong to the retrieval database. *External queries* are also supported.

As a second mode of operation, a set of training frames containing the desired pose can be provided, typically covering various persons in diverse environments. In this mode, the system has the opportunity to *learn a classifier* specific to the desired pose. We refer to the two modes as *query mode* (subsection 3.2), and *classifier mode* (subsection 3.3) respectively.

3.1. Pose descriptors

The procedure in section 2 outputs a track of pose estimates for each person in a shot. For each frame in a track, the pose estimate $E = \{E_i\}_{i=1..N}$ consists of the posterior marginal distributions $E_i = P(l_i = (x, y, \theta))$ over the position of each body part i (figure 3b), where N is the number of parts. Location (x, y) is in the scale-normalized coordinate frame centered on the person's head delivered by the initial upper body detection, making the representation translation and scale invariant. Moreover, the pose estimation process factors out variations due to clothing and background, making E well suited for pose retrieval, as it conveys a purely spatial arrangements of body parts.

In this section we present three pose descriptors derived from E. Of course there is a wide range of descriptors that could be derived and here we only probe three points, vary-

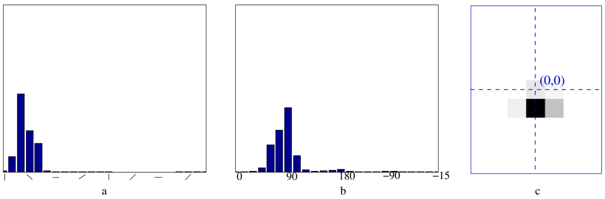

Figure 4. **Descriptor B.** (a) Distribution over orientations (x-axis) for RUA $P(l^o_{RUA} = \theta)$ from figure 3b. (b) Distribution over relative orientation (x-axis) from RUA to RLA $P(r(l_{RUA}, l_{RLA}) = \rho)$, in degrees. (c) Distribution over relative location (x-axis) from RUA to RLA $P(l^{xy}_{RLA} - l^{xy}_{RUA} = \delta)$.

ing the dimension of the descriptor and what is represented from E. Each one is chosen to emphasize different aspects, e.g. whether absolute position (relative to the original upper body detection) should be used, or only relative (to allow for translation errors in the original detection).

Descriptor A: part positions. A simple descriptor is obtained by downsizing E to make it more compact and robust to small shifts and intra-class variation. Each E_i is initially a $141 \times 159 \times 24$ discrete distribution over (x, y, θ), and it is resized down separately to $20 \times 16 \times 8$ bins. The overall descriptor $d_A(E)$ is composed of the 6 resized E_i, and has $20 \times 16 \times 8 \times 6 = 15360$ values.

Descriptor B: part orientations, relative locations, and relative orientations. The second descriptor encodes the relative locations and relative orientations between pairs of body parts, in addition to absolute orientations of individual body parts.

The probability $P(l^o_i = \theta)$ that part l_i has orientation θ is obtained by marginalizing out location (figure 4a)

$$P(l^o_i = \theta) = \sum_{(x,y)} P\left(l_i = (x, y, \theta)\right) \qquad (2)$$

The probability $P(r(l^o_i, l^o_j) = \rho)$ that the relative orientation $r(l^o_i, l^o_j)$ from part l_i to l_j is ρ is

$$P(r(l^o_i, l^o_j) = \rho) = \sum_{(\theta_i, \theta_j)} P(l^o_i = \theta_i) \cdot P(l^o_j = \theta_j) \cdot \mathbf{1}(r(\theta_i, \theta_j) = \rho) \qquad (3)$$

where $r(\cdot, \cdot)$ is a circular difference operator, and the indicator function $\mathbf{1}(\cdot)$ is 1 when the argument is true, and 0 otherwise. This sums the product of the probabilities of the parts taking on a pair of orientations, over all pairs leading to relative orientation ρ (figure 4b). It can be implemented efficiently by building a 2D table $T(l^o_i, l^o_j) = P(l^o_i = \theta_i) \cdot P(l^o_j = \theta_j)$ and summing over the diagonals (each diagonal corresponds to a different ρ).

The probability $P(l^{xy}_i - l^{xy}_j = \delta)$ of relative location $\delta = (\delta_x, \delta_y)$ is built in an analogous way (figure 4c). It involves the 4D table $T(l^x_i, l^y_i, l^x_j, l^y_j)$, and summing over lines corresponding to constant δ.

By recording geometric relations between parts, this descriptor can capture local structures characteristic for a pose, such as the right angle between the upper and lower arm in the 'hips' pose (figure 7). Moreover, locations of individual parts are not included, only relative locations between parts. This makes the descriptor fully translation invariant, and unaffected by inaccurate initial detections.

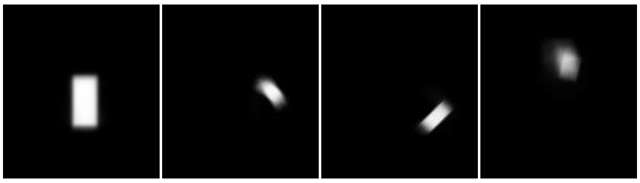

Figure 5. **Descriptor C.** Soft-segmentations for torso, RUA, RLA and head from figure 3b (displayed here in full resolution; the actual descriptor is downsized).

To compose the overall descriptor, a distribution over θ is computed using (2) for each body part, and distributions over ρ and over δ are computed (3) for each pair of body parts. For the upper-body case, there are 15 pairs and the overall descriptor is the collection of these $6 + 15 + 15 = 36$ distributions. Each orientation distribution, and each relative orientation distribution, has 24 bins. The relative location is downsized to 7×9, resulting in $24 \cdot 6 + 24 \cdot 15 + 9 \cdot 7 \cdot 15 = 1449$ total values.

Descriptor C: part soft-segmentations. The third descriptor is based on soft-segmentations. For each body part l_i, we derive a soft-segmentation of the image pixels as belonging to l_i or not. This is achieved by convolving a rectangle representing the body part with its corresponding distribution $P(l_i)$. Every pixel in the soft-segmentation takes on a value in $[0, 1]$, and can be interpreted as the probability that it belongs to l_i (figure 5).

Each soft-segmentation is now downsized to 20×16 for compactness and robustness, leading to an overall descriptor of dimensionality $20 \times 16 \times 6 = 1920$. As this descriptor captures the silhouette of individual body parts separately, it provides a more distinctive representation of pose compared to a single global silhouette, e.g. as used in [5, 16].

3.2. Query mode

In query mode, the user specifies the target pose with a single frame q. Through the techniques above, for every person in a shot of the retrieval database we obtain a series of pose descriptors d_f, one per video frame f in the track.

In order to search the database for shots containing the target pose, we need (i) a similarity measure between pose descriptors, for comparing the query d_q to descriptors d_f from the database, and (ii) a strategy to score a shot, based on the similarity scores to all the descriptors it contains. The final output of the pose retrieval system is a list of all shots, ranked by their score.

Similarity measures. Each descriptor type (A–C) has an accompanying similarity measure $\mathrm{sim}(d_q, d_f)$:

Descriptor A. The combined Bhattacharyya similarity ρ of the descriptor d^i for each body part i: $\mathrm{sim}_{\mathrm{A}}(d_q, d_f) = \sum_i \rho(d_q^i, d_f^i)$. As argued in [8], $\rho(a, b) = \sum_j \sqrt{a(j) \cdot b(j)}$ is a suitable measure of the similarity between two discrete distributions a, b (with j running over the histogram bins).
Descriptor B. The combined Bhattacharyya similarity over all descriptor components: orientation for each body part, relative orientation and relative location for each pair of body parts.
Descriptor C. The sum over the similarity of the soft-segmentations d^i for each part: $\mathrm{sim}_{\mathrm{C}}(d_q, d_f) = \sum_i d_q^i \cdot d_f^i$. The dot-product $\cdot$ computes the overlap area between two soft-segmentations, and therefore is a suitable similarity measure.

Shot scores. The score of a shot is set to that of the best scoring track, i.e. the person considered most likely to be carrying out the query pose. We propose here different strategies for scoring a track:
One-to-one. The track score is simply the maximum similarity of d_q to every frame: $\max_i \mathrm{sim}(d_q, d_i)$.
Top-k average. The track score is the average over the top k frames most similar to d_q.
Query interval. Consider a short time interval around the query frame q. The score of a track frame is the maximum similarity over this query interval. This improves results when pose estimation performs better in a frame near q.

The last two strategies can be combined, resulting in a track score integrating several query frames *and* several track frames.

3.3. Classifier mode

In classifier mode, a set $\mathcal{S}^+$ of training frames is made available to the system. $\mathcal{S}^+$ includes all frames containing the target pose, from a small number of videos V (e.g. from examples of that pose from a number of shots covering different people and clothing). For frames containing multiple people, $\mathcal{S}^+$ also indicates *which* of them is performing the target pose. A discriminative classifier specific to the desired pose is first learnt, and then used for scoring shots from the retrieval database.

Training a classifier. A linear SVM is trained from $\mathcal{S}^+$ and a negative training set $\mathcal{S}^-$ of frames not containing the target pose. $\mathcal{S}^-$ is constructed by randomly sampling frames from V, and then removing those in $\mathcal{S}^+$. The descriptors presented in subsection 3.1 are extracted for all frames in $\mathcal{S}^+$ and $\mathcal{S}^-$, and presented as feature vectors to the SVM trainer. For a frame of $\mathcal{S}^+$, only the descriptor corresponding to the person performing the pose is included.

Optionally, $\mathcal{S}^+$ can be augmented by perturbing the original pose estimates E with small scalings, rotations, and translations before computing their descriptors. As noted by [18], this practice improves the generalization ability of the classifier. The augmented $\mathcal{S}^+$ is 9 times larger.

Searching the database. When searching the database the SVM classifier is applied to all descriptors, and the output distance to the hyperplane is used as a score. Therefore, the SVM plays the same role as the similarity measure in query mode. Apart from this, the system operates as in query mode, including using different shot scoring strategies (e.g. top-k average) as above. The classifier mode has the potential to be more accurate than query mode, as it explicitly learns to distinguish the target pose from others. As an additional benefit, with the linear SVM we can learn which of the components of the feature vector are important from the hyperplane weighting.

4. HOG-based Pose Retrieval

We describe now our baseline pose retrieval system, which uses a Histograms of Oriented Gradients (HOG) [9] descriptor for each upper body detection in a track, rather than the pose descriptors computed from the pictorial structure inference. In order to be able to capture the pose at all in a descriptor, the window must be enlarged over the size of the original upper body detection, and we use here the enlarged region show in figure 1a (the same region is used as the starting point for fitting the articulated model). For the HOG computation this is resized to a standard 116×130 pixels (width $\times$ height).

We employ the HOG pose descriptor for pose retrieval in the same manner as the descriptors of section 3:

Query mode. The HOG-based query mode proceeds as in section 3.2, using the negative Euclidean distance between two HOG descriptors as a similarity measure. Other than scale and translation invariance we do not expect this descriptor to have the same invariances as the articulated model descriptors (such as clothing invariance). In particular, we expect it to be very sensitive to background clutter since every gradient in the enlarged region counts.

Classifier mode. Here a classifier is trained for specific poses, e.g. hips, using the same training data as in section 3.3. This has a similar objective to the keyframe pose search of [19] (e.g. a classifier for the pose of coffee at the mouth). As in [9], we use a round of bootstrapping to improve the performance. The classifier from the first round is applied to a large set of negative frames from the training videos (constructed as $\mathcal{S}^-$ in section 3.3). In the second round we add to the negative set the most positively scoring negative frames, so as to double its size, and the classifier is then re-trained.

We would expect the classifier to learn to suppress background clutter to some extent, so that this mode would have superior performance over the query mode. Also, this is a

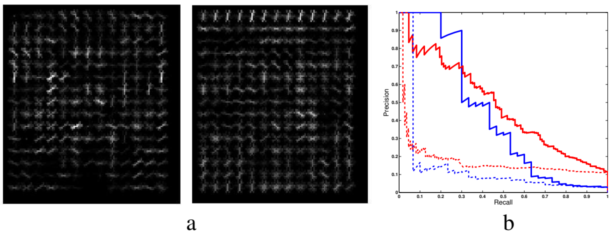

Figure 6. **(a) HOG-based hips classifier.** Left: positive values of the hyperplane learnt by the SVM. The hips-pose is just visible in the orientation flow on the left side. Right: negative values. **(b) Example precision/recall plots for experiment 1 (section 5.1)**, showing performance of descriptor C (solid) on hips (blue) and rest (red) poses, each for the query leading to the best AP. The dashed curves show the same for HOG, also for the queries leading to its best AP.

linear classifier, so the weight vector can be visualized to determine which areas of the descriptors are significant for discriminating a pose. An example is shown in figure 6a, with the details of the training given in section 5.2.

5. Experiments

We present experiments on a video database consisting of TV show episodes and Hollywood movies. For each video the following steps are carried out: first it is partitioned into shots; then an upper body (UB) detector is run on every frame and tracked through the shot using [13]; for each track, we apply our improved pose estimation algorithm from section 2 on every frame with a detection; and finally for each detection we have three descriptors (A–C) computed from the fitted articulated model, and a HOG descriptor of the enlarged region (which is used for the baseline comparisons).

Video data and ground truth labelling. We show quantitative evaluations on five episodes of the TV series 'Buffy the Vampire Slayer' (episodes 2–6 of the fifth season, a total of 1394 shots containing any upper body, or about 130000 frames). In addition, we also show retrieval examples on three Hollywood movies, 'Gandhi', 'Four Weddings and a Funeral', and 'Love Actually', for a total of 1303 shots with upper bodies (about 210000 frames).

For the five Buffy episodes every shot is ground truth labelled as to which of three canonical poses it contains: hips, rest, and folded (figure 7). Three labels are possible indicating whether the shot contains the pose, does not contain the pose, of if the frame is ambiguous for that pose. Ambiguous cases, e.g. when one arm is occluded or outside the image, are ignored in both training and testing. The statistics for these poses are given in table 1. As the ground truth labelling of these episodes is algorithm independent, we use it to assess precision/recall performance for the target poses, and to compare different descriptors and search options. We have released this ground truth annotation on the web [1].

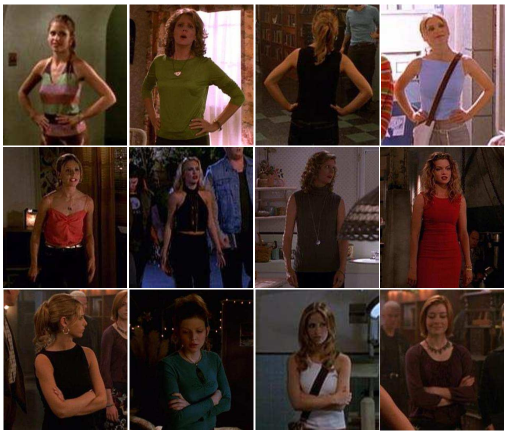

Figure 7. **Canonical poses labelled in ground truth.** From top to bottom: four instances of hips, rest, and folded. These are also used as queries in section 5.1.

5.1. Experiment 1: Query mode – Buffy

For each pose we select 7 query frames from the 5 Buffy episodes. Having several queries for each pose allows to average out performance variations due to different queries, leading to more stable quantitative evaluations. Each query is searched for in all 5 episodes, which form the retrieval database for this experiment. For each query, performance is assessed by the average precision (AP), which is the area under the precision/recall curve. As a summary measure for each pose, we compute the mean AP over its 7 queries (mAP). Three queries for each pose are shown in figure 7. In all quantitative evaluations, we run the search over all shots containing at least one upper body track.

Shot scores. We investigate the impact of the different strategies for scoring tracks, while keeping the descriptor fixed to A (section 3.2). Both ideas of *query interval* and *top-k average* bring a visible improvement. We found a query interval of 5 frames and $k = 10$ to perform best, and to improve mAP for 'hips' to 26.3%, from the 21.5% achieved by the straightforward *one-to-one* approach. In the following experiments, we leave these parameters fixed at this optimal value.

Descriptors. As table 1 shows, pose retrieval based on articulated pose estimation performs substantially better than the HOG baseline (section 4), on all poses, and for all three descriptors we propose (section 3.1). As the query pose occurs infrequently in the database, absolute performance is much better than chance (e.g. 'hips' occurs only in 3% of the shots), and we consider it very good given the high challenge posed by the task [1]. While most of the top 10 shots re-

[1] The pose retrieval task is harder than simply classifying images into three pose classes. For each query the entire database of 5 full-length episodes is searched, which contains many different poses.

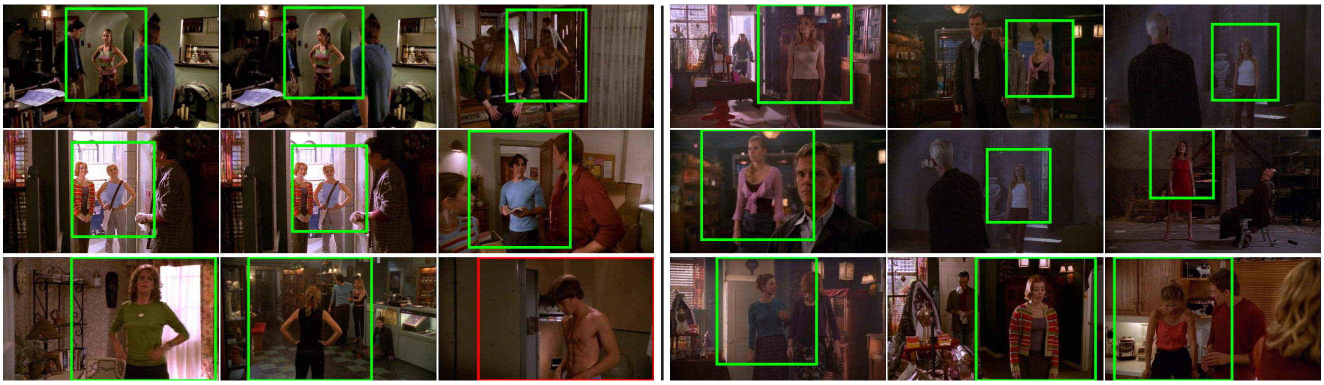

Figure 8. **Query mode. Left: Hips.** Top 9 returned shots for the result with the highest mAP (45.6). The query is the first frame. Notice how Joyce (Buffy's mother) is returned at rank #7. The system also returns a box around the person with pose most similar to the query (marked green when correct, and red otherwise). **Right: Rest.** Top 9 returned shots for the result with the highest mAP (61.5). Again, the query is the first frame. Note, the variety of clothing, backgrounds, and people retrieved starting from a single query frame.

	A	B	C	HOG	instances	chance
hips	**26.3**	24.8	25.5	8.0	31 / 983	3.2 %
rest	38.7	**39.9**	34.0	16.9	108 / 950	11.4 %
folded	14.5	**15.4**	14.3	8.1	49 / 991	4.9 %

Table 1. **Experiment 1.** *Query mode (test set = episodes 1–6). For each pose and descriptor, the table reports the mean average precision (mAP) over 7 query frames. The fifth column shows the number of instances of the pose in the database, versus the total number of shots searched (the number of shot varies due to different poses having different numbers of shots marked as ambiguous in the ground-truth). The last column shows the corresponding chance level.*

turned by our system are typically correct, precision is only high at low recall, which explains why the mAP are quite below 100% (see figure 6b). Notice how HOG also performs better than chance, because shots with frames very similar to the query are highly ranked, but it fails to generalize.

As shown in figure 8, our method succeeds in returning different people, wearing different clothes, at various scales, background, and lighting conditions, starting from a *single* query frame. Interestingly, no single descriptor outperforms the others for all poses, but the more complex descriptors A and B do somewhat better than C on average.

5.2. Experiment 2: Classifier mode – Buffy

We evaluate here the classifier mode. For each pose we use episodes 2 and 3 as the set V used to train the classifier (section 3.3). The positive training set $\mathcal{S}^+$ contains all time intervals over which a person holds the pose (also marked in the ground-truth). The classifier is then tested on the remaining episodes (4,5,6). Again we assess performance using mAP. In order to compare fairly to query mode, for each pose we re-run using only query frames from episodes 2 and 3 and searching only on episodes 4–6 (there are 3 such queries for hips, 3 for rest, and 2 for folded). Results are given in table 2.

Several interesting observations can be made. First, the three articulated pose descriptors A–C do better than HOG on hips and rest also in classifier mode. This highlights their suitability for pose retrieval. On folded, descriptor C performs about as well as HOG. Second, when compared on the same test data, HOG performs better in classifier mode than in query mode, for all poses. This confirms our expectations from section 4, as it can learn to suppress background clutter and to generalize to other clothing/people, to some extent. Third, the articulated pose descriptors, which do well already in query mode, benefit from classifier mode when there is enough training data (i.e. on the rest pose). There are only 16 instances of hips in episodes 2 and 3, and 11 of folded, whereas there are 39 of rest. To further complicate the learning task, not all training poses are correctly estimated (see evaluation in section 2.2). This phenomenon is consistent over all three descriptors.

5.3. Experiment 3: Hollywood movies

To test the generalization ability of the proposed pose representation even further, we search three hollywood movies using several queries from 'Gandhi' (figure 9). As the figure shows, our method can retrieve a variety of different poses, and finds matches across different movies.

6. Conclusions

We have demonstrated that query based pose retrieval is possible on video material of high difficulty and variety. This opens up the possibility of further video analysis looking at combinations of poses over time, and over several characters within the same shot (interactions). Analogous pose search methods can also be developed for other (non-human) animals. We are currently investigating clustering on poses, so that typical poses of characters can be discov-

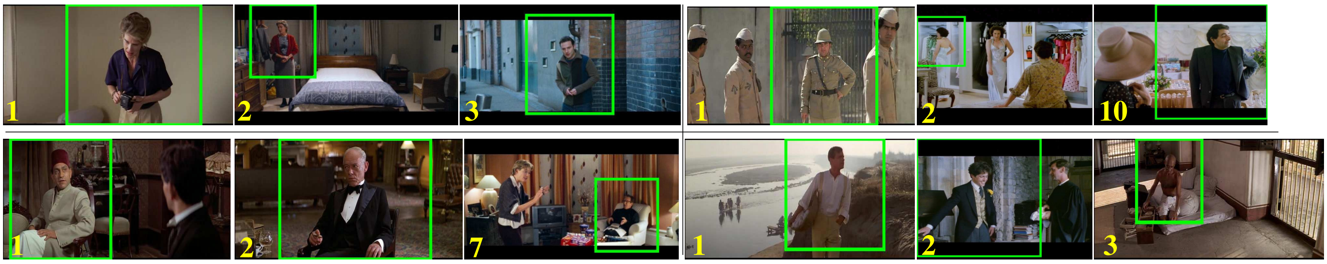

Figure 9. **Retrieval on hollywood movies.** Three of the top 10 returned shots for each of 4 queries (rank marked on the bottom left). The queries are from 'Gandhi' and the search is over all of 'Gandhi', 'Four Weddings and a Funeral', and 'Love Actually'. The first image is the query in each case. Notice the difference in illumination conditions, background, clothing and gender of the person between the query and the returned shots. Also, several correct shots from 'Four Weddings and a Funeral' and 'Love Actually' are returned, given a query from 'Gandhi' (e.g. 2nd and 3rd on top-left and top-right).

	Classifier Mode				**Query mode**			
	A	B	C	HOG	A	B	C	HOG
hips	9.2	16.8	10.8	6.8	33.9	19.9	21.3	1.7
rest	48.2	38.7	41.1	18.4	36.8	31.6	29.3	15.2
folded	8.6	12.1	13.1	13.6	9.7	10.9	9.8	10.2

Table 2. **Experiment 2.** *Left columns: classifier mode (test set = episodes 4–6). Right columns: query mode on same test episodes 4–6 and using only queries from episodes 2 and 3. Each entry reports AP for a different combination of pose and descriptor, averaged over* 3 *runs (as the negative training samples* $\mathcal{S}^-$ *are randomly sampled).*

ered from video material.

Acknowledgements. We thank Deva Ramanan for the image parsing code [26] and Marcin Eichner for suggesting corrections to the kinematic prior. We are grateful for support from EU Project CLASS, the Swiss National Science Foundation (SNSF), the Royal Academy of Engineering, Microsoft, and the Spanish projects TIN2005-01665 and Consolider Ingenio-2010 (MIPRCV).

References

[1] http://www.robots.ox.ac.uk/∼vgg/research/pose_estimation/index.html.

[2] A. Agarwal and B. Triggs. Tracking articulated motion using a mixture of autoregressive models. In *ECCV*, 2004.

[3] O. Arandjelovic and A. Zisserman. Automatic face recognition for film character retrieval in feature-length films. In *CVPR*, 2005.

[4] C. Bishop. Pattern recognition and machine learning. *Springer*, 2006.

[5] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, 2005.

[6] A. Bobick and J. Davis. The recognition of human movement using temporal templates. *IEEE PAMI*, 23(3):257–267, 2001.

[7] Y. Y. Boykov and M. P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images. In *ICCV*, volume 2, pages 105–112, 2001.

[8] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. In *IEEE PAMI*, 2002.

[9] N. Dalal and B. Triggs. Histogram of Oriented Gradients for Human Detection. In *CVPR*, volume 2, pages 886–893, 2005.

[10] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *ICCV VS-PETS*, 2005.

[11] A. Fathi and G. Mori. Action recognition by learning mid-level motion features. In *CVPR*, 2008.

[12] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *IJCV*, 61(1), 2005.

[13] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Progressive search space reduction for human pose estimation. In *CVPR*, Jun 2008.

[14] N. Ikizler and P. Duygulu. Human action recognition using distribution of oriented rectangular patches. In *ICCV workshop on Human Motion Understanding*, 2007.

[15] S. Ioffe and D. Forsyth. Finding people by sampling. In *ICCV*, 1999.

[16] Y. Ke, R. Sukthankar, and M. Hebert. Spatio-temporal shape and flow correlation for action recognition. In *CVPR*, 2007.

[17] M. P. Kumar, P. H. S. Torr, and A. Zisserman. Learning layered pictorial structures from video. In *ICVGIP*, pages 148–153, 2004.

[18] I. Laptev. Improvements of object detection using boosted histograms. In *BMVC*, 2006.

[19] I. Laptev and P. Perez. Retrieving actions in movies. In *ICCV*, 2007.

[20] M. Lee and R. Nevatia. Body part detection for human pose estimation and tracking. In *Workshop on Motion and Video Computing*, 2007.

[21] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *CVPR*, 2005.

[22] P. Li, H. Ai, Y. Li, and C. Huang. Video parsing based on head tracking and face recognition. In *CIVR*, 2007.

[23] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *ECCV*. Springer-Verlag, May 2004.

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

[25] J. Niebles and L. Fei-Fei. A hierarchical model model of shape and appearance for human action classification. In *CVPR*, 2007.

[26] D. Ramanan. Learning to parse images of articulated bodies. In *NIPS*. MIT Press, 2006.

[27] D. Ramanan, D. A. Forsyth, and A. Zisserman. Tracking people by learning their appearance. *IEEE PAMI*, 29(1):65–81, Jan 2007.

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

[29] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *SIGGRAPH*, 2004.

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

[31] J. Sivic, M. Everingham, and A. Zisserman. Person spotting: video shot retrieval for face sets. In *CIVR*, 2005.

[32] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, 2003.