Human Pose Estimation with Implicit Shape Models

Jürgen Müller and Michael Arens
Fraunhofer Institute of Optronics, System Technologies and Image Exploitation
Gutleuthausstr. 1, 76275 Ettlingen, Germany
juergen.mueller@iosb.fraunhofer.de, michael.arens@iosb.fraunhofer.de

ABSTRACT

We address the problem of articulated 2D human pose estimation in natural images. A well-known person detector – the Implicit Shape Model (ISM) approach introduced by Leibe et al. – is shown not only to be well suited to detect persons, but can also be exploited to derive a person's pose. Therefore, we extend the original voting approach of ISM and let all visual words that contribute to a person hypothesis also vote for the positions of the person's body parts. Since this approach is not constrained to a certain feature type and different feature types can even be fused during the pose estimation process, the approach is highly flexible. We show preliminary evaluation results of our approach using on the public available HumanEva dataset which comprises ground-truth pose data and thereby provides training and evaluation data.

Categories and Subject Descriptors

I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis

General Terms

Algorithms

Keywords

human pose estimation, motion tracking, video analysis, content-based image retrieval

1. INTRODUCTION

Event analysis for video sequences is a challenging task due to the inherent pattern recognition problems. Understanding human actions is a major goal in the field since humans play an important role as subjects in video events. The detection of humans and the analysis of the human pose opens the door to action analysis since an action can be recognized as a sequence of poses. While lot of progress has been made in detecting humans (e.g. [13], [3]) in images, robust 3D and 2D pose estimation in unconstrained environments can still be considered as an unsolved problem.

ARTEMIS'10, October 29, 2010, Firenze, Italy.

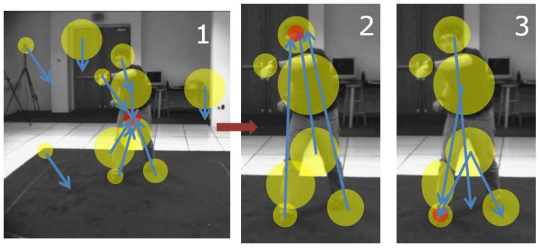


Figure 1: Overview of the proposed approach. 1: computed features (yellow discs) vote (blue arrows) for possible locations of the center of a person (red disc). 2+3: features that contributed to a person hypothesis are reused to vote for body part positions using an own ISM for each body part to be found (2: head, 3: right foot).

Several surveys are available that provide an overview over the different approaches to pose estimation (see e.g. [10]) but often different taxonomies are used to classify the methods. Techniques can be divided into top-down (e.g. comparing a projected 3D model of the human body with the current image) vs. bottom-up (e.g. body part detection and assembling). Pose estimation approaches can also be distinguished into model-free vs. model-based depending on whether the pose estimation process is guided by a model of the human body. Another classification possibility is to classify the approaches into discriminative vs. generative. A generative model learns the joint probability distribution $p(x, y)$ whereas a discriminative model learns the conditional probability distribution $p(y|x)$ with x=image features and y=pose. Due to the difficulty of the pose estimation task, several authors have suggested methods that reduce the pose search space.

When constraining the set of poses that can be recognized, the pose estimation task can be simplified since possible poses lie in a subspace of the space of all possible body part configurations. Urtasun et. al. [12] learn the mapping of tracked 2D human body points to a 3D pose via scaled Gaussian process latent variable models (SGPLVM) and show good results on restricted pose classes as e.g. gulfing poses. Upper body part pose estimation on a subset of images of the TV series Buffy are shown by Ferrari et al. [4] by a discriminative approach: the bounding box around a detected person is roughly segmented into person and background pixels and an appearance model of body parts is

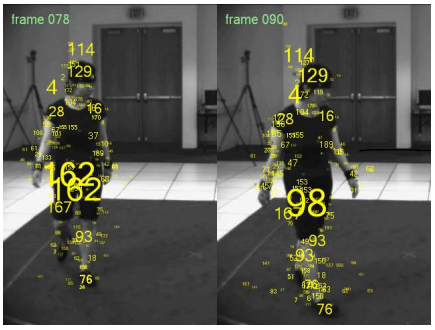


Figure 2: Visualization of positions and scales of SIFT features that contributed to the same person hypothesis.

learned by an iterative procedure of body part locating and updating the body part appearance model.

The idea of using context information for reducing the search space has recently been tackled by different authors. Yao et al. [15] show how objects and human poses can serve as a mutual context and ease the recognition of each other. While they show how the pose of sport equipment (e.g. a cricket bat) induces a strong bias onto the body pose, one can take advantage of the same idea in similar scenarios, e.g. the mutual context between the pose of music instruments and the body pose [14]. Karlinsky et al. [7] start at a good detectable body part (e.g. faces) and use chains of features to find the position of less easier detectable body parts (e.g. hands). While some authors directly estimate 3D poses based on 2D image evidence, the connection between 2D image features and 2D poses seems more obviously. Impressive results are shown by recent work of Andriluka et al. [1] which adopt an approach that first estimates 2D poses and use the 2D pose estimates to lift it up to 3D pose estimates. Within a Bayesian framework they estimate the probability of a sequence of consistent 2D poses and viewpoints over successive frames which they call 2D tracklets and use these 2D tracklets as evidence – instead of direct image evidence – for the estimation of 3D poses. All these approaches to pose estimation start with detecting the person within the image. For this task the ISM approach has proven to be quite successful. The core idea of our approach is to reuse these ISM image features that contributed to the detection of the person also for the detection of the person's body parts by learning a single ISM for each body part (see Fig. 1). In this sense we extend the ISM approach to a hierarchical one by first detecting the object center (person center) and then detecting the subparts (person's body parts). The main contribution of this paper is to show the feasibility of such a body part ISM approach for 2D human pose estimation.

Section 2 explains our method in detail. In section 3 we evaluate our approach systematically in different test scenarios and assess the quality of the estimated 2D poses. In Section 4 we conclude based on the results of our experiments.

2. METHOD

2.1 ISM for person detection

The Implicit Shape Model (ISM) introduced by Leibe et al. [8] can be used for detecting instances of arbitrary object

marker ID	marker name
0	head
1	upper spine
2	lower spine
3	right shoulder
4	right elbow
5	right hand
6	left shoulder
7	left elbow
8	left hand
9	right hip
10	right knee
11	right foot
12	left hip
13	left knee
14	left foot

Table 1: Each 2D pose is made up of the 2D positions of these 15 body parts / markers.

classes as persons, cars, motorbikes, cows, etc. The idea is to represent an object class by the typical distribution of small image patches relative to the object center, i.e. an object class is described by a tuple

$$ISM(C, O) = (C, P_C) \quad (1)$$

where C is a codebook containing descriptor vectors of image features that can appear on the object and P_C is a probability distribution that captures where the object center typically appears relative to the codevectors. An important requirement is that P_C represents the probability over the locations of the object center for a given codevector independently of all other codevectors. This allows for a simple voting scheme when using a learned ISM for object detection in a given image: having recognized an image feature known from the codebook as codevector v with $v = 1, ..., S$ where S is the number of codevectors, we can vote for the object center to be at the locations $\vec{l}$ relative to the position of the found feature v according to $P_C(\vec{l}|v)$ independently from all other found features $v' \neq v$.

When using scale-invariant interest point detectors and descriptors the observed image features do not only have a position but a scale. We have to take into account the scale of the features when learning $P_C(\vec{l}|v)$ by normalizing the voting offset vectors by the observed feature scale s_1 during learning and rescaling the voting offset vectors by the feature scale s_2 present during the voting procedure.

The ISM approach is attractive for person detection since it can handle cases in which persons are partly occluded by objects. Further it circumvents modeling the relationship between the presence / location of image feature and the location of the person center by a joint probability distribution which would require a huge set of training samples. Instead it integrates hints from individual image features by a voting scheme.

For learning a person ISM we create a codebook of SIFT features [9] in a first step by sampling SIFT interest points from a set of sample images of persons and perform an agglomerative clustering on the corresponding SIFT descriptor vectors to gain codevectors (prototypes, visual words, centroids). In a second step the positions of the person center relative to the codevectors are recorded to learn P_C. For an unknown image we can then sample SIFT interest points over the whole image area, assign the SIFT descriptor vectors to codevectors and use the probability distribution P_C to cast votes for the person center.

In situations where two persons pass by it may be not

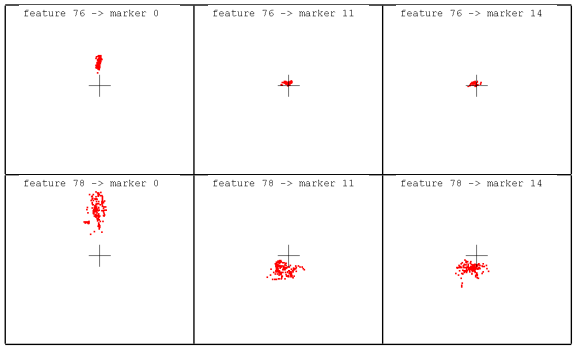


Figure 3: Examples of feature → marker voting maps. First row: feature #76 to marker head, left foot, right foot. Second row: feature #78 to marker head, left foot, right foot.

clear which image feature belongs to which person hypothesis due to the possibility that a feature contributes votes for both person hypotheses. For this, we adopt the approach described in [5] and track SIFT features by assigning each SIFT feature to a person hypothesis and use a confidence value that is increased when the feature is found again and decreased when it cannot be confirmed in the next frame. This makes it easier to keep track of which features belong to which person hypotheses when two or more persons overlap in the image.

For further details about the ISM the reader is referred to the original work of Leibe et al. [8].

2.2 ISMs for body parts

Fig. 2 shows SIFT features found in two sample frames from the public available HumanEva dataset[1]. The numbers represent the codevector IDs within our SIFT codebook used for person detection while the text size indicates the scale of the respective SIFT feature. Such *feature clouds* contain not only information about the person center, but also worthwhile hints for the positions of body parts. Some of these features can directly be associated to certain body parts, as has been shown in [6]. Consider e.g. feature #129 (head), feature #93 (legs), feature #76 (feet/shoes), feature #28 (right shoulder), and feature #16 (left shoulder). Note how the text size of features #129 and #76 slightly becomes bigger as the feature scale of the corresponding image structures (head, shoe) becomes bigger in frame 90 compared to frame 78.

Though such apparent one-to-one feature ↔ body part mappings seem tempting, we have to keep in mind that the same feature #76 which appeared on the foot may appear on other similar looking body parts and even on the background.

Beside features that appear more or less directly on the center of certain body parts we can also observe features that do not allow a unique assignment to a single body part. Consider as an example the features #162 and #98: these features are typically found at a big scale in scale space, i.e. cover a big part of the person's image structure. The corresponding descriptor vectors describe parts of the hips and the thighs. Nevertheless, the systematic behavior where they typically appear carries valuable information about the position of some body parts.

[1] http://vision.cs.brown.edu/humaneva/

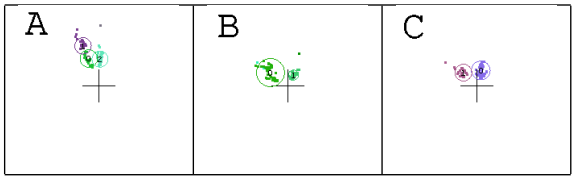


Figure 4: Examples of mean shift clustered <*feature,marker*> voting maps of feature #75 → to marker right shoulder (A), right hand (B), left hand (C)

The core idea of our approach is to use all these image features – independent of whether they are associated only with a single body part or whole pose sub configurations – by learning an ISM for each body part that captures the relationship between their presence and the location of body markers.

2.3 Learning body part ISMs

The observation that such feature clouds can carry valuable information for 2D pose estimation leads directly to the idea to learn a separate ISM for each body part.

Table 1 lists the 15 body parts we try to find. The positions of these markers relative to the person center (bounding box center if bounding boxes are used) defines the 2D pose we try to estimate, i.e. a vector

$$\vec{p} = \{\vec{m_i} = (x_i, y_i) : i = 0, .., 14\} \quad (2)$$

These markers correspond to a subset of the markers attached to the subjects in the HumanEva data sets which come with synchronized video and 3D motion capture data streams. Using the extrinsic and intrinsic camera parameters that are provided as well with the dataset for each video camera, we can project the 3D positions of the markers into the 2D camera image to get the 2D image positions of the markers. This procedure yields the training data, N pairs of images $\vec{I}$ and corresponding 2D poses $\vec{p}$:

$$\{(\vec{I_j}, \vec{p_j}) : j = 1, .., N\} \quad (3)$$

Since we do not work on pixel images $\vec{I_j}$ but extract SIFT interest points and descriptor vectors within the images, our training samples are pairs

$$\{(\vec{F_j}, \vec{p_j}) : j = 1, .., N\} \quad (4)$$

where $\vec{F_j}$ is the set of all image features

$$\vec{F_j} = \{\vec{f_k} = (s_k, x_k, y_k, v_k, b_k) : k = 1, .., K\} \quad (5)$$

where each image feature $\vec{f_k}$ has a scale s_k, a position (x, y), and an assigned codevector ID v_k with a belief value b_k that represents the belief that the found feature descriptor vector corresponds to the codevector v_k.

In the first step, learning an ISM for each body part (ISM_{head}, $ISM_{rightfoot}$, $ISM_{leftfoot}$ etc.) means building up sample lists

$$S_{v,m} = \{(x, y)_i : i = 1, ..., N\} \quad (6)$$

based on our N training pairs $\vec{F_j}$ for each codevector v and body part / marker m that captures the observed positions of marker $m = 0, ..., 14$ relative to the feature center f_k.

In Fig. 3 we show six examples of such sample *feature to marker* voting lists generated from the *HumanEva subject 1 walking* sequence. The marker votes are displayed with the feature forming the coordinate origin. Note that the

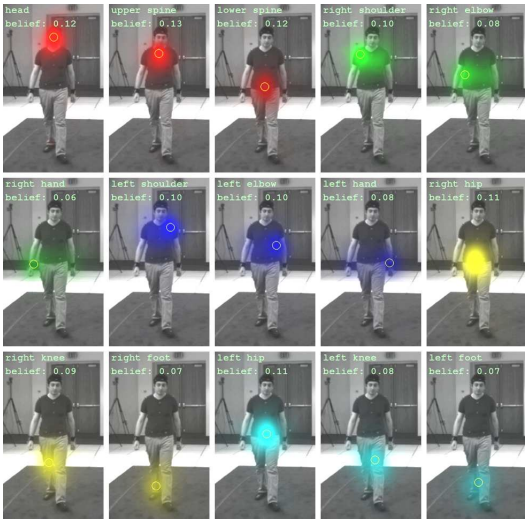


Figure 5: To find each body part we use a separate map that integrates all 2D Gaussian feature votes for that body part.

marker vote vectors were rescaled to unit feature scale. The presumption that feature #76 appears mainly on the feet (marker 11,14) is confirmed. The first map shows that a highly probable location of the head is expected above this feature.

The second row shows an example of another feature that has not such a clear explanatory power concerning the position of head, left and right foot. The votes are far more spread. Nethertheless, finding feature #78 also gives some weak hints about the height where we can find marker 0, 11, and 14.

In the original work of Leibe et al. such sample lists are directly taken as a non-parametric representation of the feature → marker probability distributions. While this allows to approximate the true distribution in as much detail as the training data permits, it has a severe drawback if we have a lot of such samples.

For a codebook with S codevectors, M markers (here: $M = 15$) and N available training frames with ground truth poses we would have to store $S \cdot M \cdot N$ samples. While this might still be unproblematic concerning data storage complexity for some hundreds of codevectors S and some thousands of training frames N due to the fact that the voting samples are only 2D vectors, this approach becomes problematic when we come to the point where we want to use this representation of a probability density for estimating the marker positions and time is a critical issue because we aim to reach real-time pose estimation. Having found $T < S$ different codevectors in an image (the number of instances of features found in frame can even be bigger than S), we have to traverse $T \cdot M \cdot N$ of the samples in the database and transform each sample into a vote cast. It now depends heavily on how expensive the vote cast is whether this fits to our time constraints or not.

To avoid such performance problems we suggest to turn the non-parametric sample list representation of the probability densities into parametric density representations. Currently, we reduce the list of samples by identifying voting clusters. For this we start a Mean Shift clustering [2] on each of the $S \cdot 15$ (S codevectors, 15 markers) sample distributions to identify voting clusters. For the mean shift kernel we use a Gaussian kernel. As a result we obtain a list of L clusters for each <*feature, marker*> pair as in Fig. 4. In addition to the cluster centers, we also compute how many samples are assigned to the corresponding cluster and the mean distance of these samples to the cluster center $\tilde{d}$ and use $r = 2\tilde{d}$ as a rough approximation of the cluster radius. This gives us a parametric description of the vote distribution, i.e.

$$C_{v,m} = \{C_l = (x, y, r, n, o) : l = 1, ..., L\} \quad (7)$$

where (x, y) is the center of the cluster in the feature coordinate system, r is the cluster radius, n is the number of samples that were assigned to this cluster, and o is the ratio $\frac{n}{N}$ that tells us which fraction of the total number of vote samples are represented by this cluster.

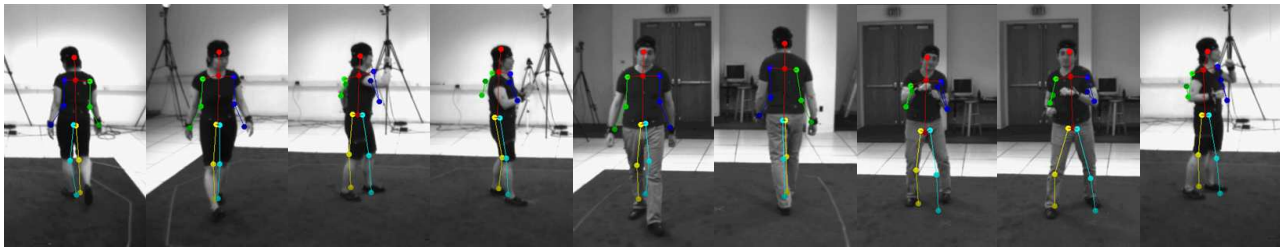
2.4 Using body part ISMs for pose estimation

A high number of samples n within a voting cluster (i.e. high ratio o) can only be interpreted as a high probability to find the corresponding marker within the cluster borders if we make sure that our training data is to some extent balanced in the sense that we do not have a strong bias for certain poses within the set of training poses. As an example imagine the situation that we detect a feature X at the right hand during 100 frames in which the person lifts up the right arm. The head may now be found below feature X. If we further have 10 frames where the persons right hand is beside its right hip, the head can now be found above this feature. In both cases we will find clear <feature X, right hand> votes, but we will find 10 times more samples in the <feature X, head> vote map in the bottom of this map than in the top of this map.

In such unbalanced training set scenarios we have to be careful how to interpret n and o since the non-parametric approach to represent the vote distribution directly by the set of all samples as done in the original ISM approach becomes critical when coming to the voting phase: then we have no abstraction of such pose biases available and would vote 100 times the head to be somewhere below feature X and 10 times the head to be somewhere above feature X.

In contrast, using the parametric cluster representation of the vote distribution gives us the possibility to consider a high number of samples within a vote cluster just as an artifact of an unbalanced training set and let each cluster take part in equal measure in the voting process.

For estimating a pose given an unknown image with a detected person, we use the features that contributed to the person hypotheses. As in training, our perception of the person in frame j is the set of features $\vec{F}_j$ (see def. (5)). For each feature $f_k \in \vec{F}_j$ and marker m we now have a parametric representation of the vote distribution in form of $C_{v_k,m}$ available. This list of clusters has to be converted into votes for the positions of the markers m. For this, we maintain a 2D voting map $V_m \in \mathbb{R}^2$ for each marker m where we can integrate votes from the different features f_k (see Fig. 5). The origin of V_m is set identical to the person center. Each vote cluster $C_l \in C_{v_k,m}$ is then converted into a vote into V_m. Since we consider the area covered by the cluster C_l as a rough approximation where we can find the marker m with decreasing probability to the border of the cluster area, we add a 2D Gaussian $f(x, y)$ with variance $r \cdot c$ (c is a constant, r the cluster radius) into V_m for each feature and its corresponding clusters, i.e.



1 2 3 4 5 6 7 8 9

Figure 6: One pose estimation example from each of the experiments 1-9.

$$V_m(x,y) = \sum_{\vec{f_k}=(s_k,x_k,y_k,v_k,b_k)\in \vec{F_j}} \sum_{C_l \in C_{v_k,m}} f_{\vec{f_k},C_l}(x,y) \quad (8)$$

with

$$f_{\vec{f_k},C_l}(x,y) = A\frac{1}{2\pi\sigma^2} e^{-\left(\frac{(x-x_o)^2}{2\sigma^2}+\frac{(y-y_o)^2}{2\sigma^2}\right)} \quad (9)$$

where (x_0, y_0) is the position of feature f_k within the voting map (i.e. its position relative to the person center) plus the (feature scale s_k rescaled) voting offset vector defined by the cluster center C_l. As mentioned before, for the variance of these 2D Gaussians we use

$$\sigma^2 = r \cdot c \quad (10)$$

where r is the radius of the current vote cluster and A is the amplitude (height) of the Gaussian bell that is set to $\frac{1}{|C_{v_k,m}|}$. A compensates for different numbers of voting clusters we have found for different features. This makes sure that a feature $\vec{f_1}$ that has e.g. 10 voting clusters, compared to a feature $\vec{f_2}$ that has only 1 vote cluster does not get 10 times more weight when casting votes. Since the integral of a 2D Gaussian is 1, the integral of the function in (9) is A. For a feature $\vec{f_k}$ that has $|C_{v_k,m}|$ voting clusters for marker m, we create $|C_{v_k,m}|$ 2D gaussians with an integral of $\frac{1}{|C_{v_k,m}|}$ each, thus the total voting mass induced by a feature $\vec{f_k}$ sums up to 1. In a nutshell: each feature has the same weight when voting for the position of a body part. Depending on the features and training data used, a set of maxima will emerge within each voting map V_m as shown in Fig. 5 due to the superposition of the Gaussian bells. These modes could be identified by a Mean Shift procedure as well and be considered as candidates for the body part positions. Currently we take the maximum vote out of each voting map and consider it as the body part position as a single maximum appears relatively clear in each body part voting map. Thus, the final 2D pose estimate $\vec{p} = \{\vec{m_i} : i = 0, .., 14\}$ is given by

$$\vec{m_i} = \arg\max_{(x,y)} V_i(x,y) \quad (11)$$

3. EXPERIMENTS

The HumanEva dataset helps us to test different pose estimation scenarios systematically, since it comes with 4 different persons $S_1, ..., S_4$, performing different movements (poses) recorded by different cameras. Each movement sequence consists of about 300-600 frames. Table 2 shows the definition of our test scenarios. Testing the pose estimation on *S1 walking, cam2* sequence means e.g. estimating 2D poses for 549 frames.

For assessing the quality of an estimated pose, we compute the average distance in pixels between the markers in the ground truth 2D pose vector $\vec{p}$ and the estimated pose vector $\tilde{\vec{p}}$, i.e.

$$Error(\vec{p}) = \sum_{i=0}^{14} \frac{|\vec{m_i} - \vec{m_i}|}{15} \quad (12)$$

and for assessing the quality of pose estimation in one of the test scenarios 1-9, we compute the average pose error, averaged over all frames of the corresponding test sequence. This simple measure was suggested by Sigal and Black [11] to make results of different authors in 2D human pose estimation comparable. All experiments $1-9$ were conducted with SIFT features within a person hypotheses tracking framework as described in [5]. Table 3 shows the pose estimation results achieved with our method and Fig. 6 shows a single sample of an estimated pose of each experiment conducted.

The results in Fig. 6 show that we can roughly estimates a 2D pose. The upper body part pose estimates for the arms in experiment where the viewpoints are different between training and testing are wrong. This indicates that the learned body part ISMs are viewpoint specific and do not generalize enough to such strong viewpoint changes.

The last row of Table 3 shows the pose estimation errors per body part averaged over all nine experiments. We can clearly see that it is much more easy to locate markers that are more or less fixed to the torso as head, upper/lower spine, the shoulders, and the hips since these body parts do not undergo strong articulations. In contrast, body parts that undergo high articulations since they are at the end of a leg or arm kinematic actuator chain as feet and hands are more difficult to locate.

The best 2D pose estimation errors on the HumanEva data are reported by Andriluka et al. [1] to the best of our knowledge. The authors report a mean 2D pose estimation error of 10-11 pixels. Though their are several differences between the constraints made here and their approach. First, Andriluka et al. additionally learn a viewpoint specific model of the subjects S1 to S3 in advance before the actual pose estimation process starts and further use viewpoint classifiers. Second, results are reported only for a single experiment on the HumanEva data: training on all available training sequences of S3 and testing on the S2 walking sequence cam1+cam2. Third, temporal 2D pose coherency information between frames is used and exploited by a HMM based tracking procedure.

exp no.	test on same (training vs. test data)			training data	test data
	person	viewpoint	pose types		
1	yes	no	yes	S1 walking, cam1	S1 walking, cam2
2	yes	no	yes	S1 walking, cam1+cam3	S1 walking, cam2
3	yes	no	yes	S1 boxing, cam1	S1 boxing, cam2
4	yes	no	yes	S1 boxing, cam1+cam3	S1 boxing, cam2
5	no	yes	yes	S1 walking, cam1	S2 walking, cam1
6	no	yes	yes	S1+S3 walking, cam1	S2 walking, cam1
7	no	yes	yes	S1 boxing, cam1	S2 boxing, cam1
8	no	yes	yes	S1+S3 boxing, cam1	S2 boxing, cam1
9	no	no	no	S2 walking, cam1	S1 boxing, cam2

Table 2: Experiments conducted for evaluation of the body part ISM approach.

marker/exp	1	2	3	4	5	6	7	8	9	∅ (std)
head	8,0 (2,2)	10,7 (2,3)	5,2 (1,7)	12,3 (1,4)	23,9 (2,5)	14,1 (2,6)	15,2 (2,7)	13,5 (2,7)	5,2 (1,7)	**12,0 (2,2)**
upper spine	7,6 (2,2)	9,2 (2,1)	10,6 (1,8)	16,0 (1,6)	15,2 (2,2)	10,2 (2,3)	16,7 (2,7)	16,6 (2,6)	10,6 (1,8)	**12,5 (2,1)**
lower spine	7,7 (2,2)	7,8 (2,3)	12,5 (1,9)	16,4 (1,6)	11,7 (2,2)	11,1 (2,3)	11,4 (2,4)	13,2 (2,5)	12,5 (1,9)	**11,6 (2,1)**
right shoulder	13,3 (3,1)	13,3 (2,3)	25,6 (2,7)	21,5 (2,3)	25,3 (3,7)	17,1 (3,1)	14,3 (2,7)	15,1 (3,0)	25,6 (2,7)	**19,0 (2,9)**
right elbow	11,7 (2,6)	13,3 (2,6)	54,8 (4,7)	55,7 (4,4)	23,1 (3,9)	20,0 (3,4)	17,1 (3,1)	20,2 (3,2)	54,8 (4,7)	**30,1 (3,6)**
right hand	35,0 (5,2)	35,6 (5,1)	72,8 (4,6)	68,8 (4,3)	46,4 (5,6)	43,2 (5,5)	40,8 (3,5)	36,1 (3,8)	72,8 (4,6)	**50,2 (4,7)**
left shoulder	7,1 (2,3)	8,3 (2,3)	23,1 (2,3)	14,1 (2,4)	20,4 (2,8)	15,6 (2,5)	16,3 (3,0)	17,5 (3,0)	23,1 (2,3)	**16,1 (2,5)**
left elbow	8,5 (2,3)	9,2 (2,2)	28,7 (3,5)	22,7 (3,5)	19,8 (3,8)	20,3 (3,9)	20,0 (3,5)	22,5 (3,3)	28,7 (3,5)	**20,0 (3,3)**
left hand	18,6 (3,6)	17,2 (3,4)	22,4 (3,7)	34,4 (3,8)	23,7 (4,3)	22,7 (4,1)	24,4 (3,9)	23,3 (3,8)	22,4 (3,7)	**23,2 (3,8)**
right hip	8,3 (2,1)	9,1 (2,1)	18,6 (2,2)	19,0 (2,0)	13,3 (2,2)	12,4 (2,3)	12,0 (2,4)	13,4 (2,5)	18,6 (2,2)	**13,9 (2,2)**
right knee	16,4 (2,7)	16,6 (2,8)	12,4 (2,6)	10,0 (1,9)	21,2 (3,3)	20,1 (3,3)	17,3 (2,9)	20,1 (2,7)	12,4 (2,6)	**16,3 (2,8)**
right foot	24,4 (3,4)	23,7 (3,4)	12,9 (2,2)	32,6 (5,0)	31,6 (4,3)	31,9 (4,4)	30,3 (3,1)	22,0 (3,3)	12,9 (2,2)	**24,7 (3,5)**
left hip	9,0 (2,3)	9,7 (2,3)	7,4 (1,6)	14,4 (1,5)	13,2 (2,7)	13,4 (2,8)	11,6 (2,4)	13,1 (2,5)	7,4 (1,6)	**11,0 (2,2)**
left knee	15,5 (3,2)	16,6 (3,2)	13,3 (1,5)	28,7 (3,2)	19,8 (3,3)	21,4 (3,4)	22,2 (2,6)	23,4 (2,2)	13,3 (1,5)	**19,3 (2,7)**
left foot	24,2 (3,6)	24,5 (3,6)	21,7 (1,9)	19,1 (2,2)	30,4 (4,2)	30,4 (4,2)	41,1 (2,5)	42,4 (2,4)	21,7 (1,9)	**28,4 (2,9)**
∅ (std)	**14,4 (2,9)**	**15,0 (2,8)**	**22,8 (2,6)**	**25,7 (2,7)**	**22,6 (3,4)**	**20,3 (3,4)**	**20,7 (2,9)**	**20,8 (2,9)**	**22,8 (2,6)**	

Table 3: Results for experiments 1-9. Errors are specified relative to ground truth marker positions (pixel distance, standard deviation in parentheses).

In contrast the approach presented here is up to this point rather puristic since it is based on a per-frame estimation basis that solely uses the features already used to detect the person. This makes the approach easy to implement as an addition on top of a person ISM detector.

4. CONCLUSIONS

We have presented a hierarchical extension of the ISM approach to 2D human body pose estimation. The experimental results are only preliminary but suggest that the approach can be used for a rough 2D pose estimation.

Note that the approach is up to this point completely model-free, i.e. no pre-knowledge about the human body has been used so far. Additional knowledge about possible and typical configurations of the human body parts, i.e. modeling the relationship between the body part positions, is supposed to improve the pose estimation results shown in the experiments section.

The attractivity of our approach lies in its straightforwardness: we can recycle features that have already been used for person detection to detect body parts as well using the same technique (ISM).

5. REFERENCES

[1] M. Andriluka, S. Roth, and B. Schiele. Monocular 3d pose estimation and tracking by detection. In *Proc. of CVPR 2010*, San Francisco, USA, 2010.

[2] D. Comaniciu and P. Meer. Distribution free decomposition of multivariate data. *Pattern Analysis and Applications*, 2:22–30, 1998.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. of CVPR 2005*, pages 886–893, Washington, DC, USA, 2005.

[4] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Progressive search space reduction for human pose estimation. In *Proc. of CVPR 2008*, pages 1–8, 2008.

[5] K. Jüngling and M. Arens. Detection and tracking of objects with direct integration of perception and expectation. In *VS09*, pages 1129–1136, 2009.

[6] K. Jüngling and M. Arens. Feature based person detection beyond the visible spectrum. *Workshop on Object Tracking and Classification Beyond and in the Visible Spectrum (OTCBVS-2009). In conjunction with CVPR 2009*, pages 30–37, 2009.

[7] L. Karlinsky, M. Dinerstein, D. Harari, and S. Ullman. The chains model for detecting parts by their context. In *Proc. of CVPR 2010*, San Francisco, USA, 2010.

[8] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *IJCV*, 77(1-3):259–289, 2008.

[9] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, 2004.

[10] R. Poppe. Vision-based human motion analysis: An overview. *Computer Vision and Image Understanding*, 108:4–18, 2007.

[11] L. Sigal and M. J. Black. Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical report, 2006.

[12] R. Urtasun, D. Fleet, A. Hertzmann, and P. Fua. Priors for people tracking from small training sets. In *Proc. of ICCV 2005*, pages 403–410, 2005.

[13] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 57(2):137–154, 2004.

[14] B. Yao and L. Fei-Fei. Grouplet: a structured image representation for recognizing human and object interactions. In *Proc. of CVPR 2010*, San Francisco, USA, 2010.

[15] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *Proc. of CVPR 2010*, San Francisco, USA, 2010.