Localized fusion of Shape and Appearance features for 3D Human Pose Estimation

S. Sedai
suman@csse.uwa.edu.au
M. Bennamoun
m.bennamou@csse.uwa.edu.au
D. Q. Huynh
du@csse.uwa.edu.au

School of Computer Science and
Software Engineering
The University of Western Australia
Crawley, Perth
Western Australia 6009

Abstract

This paper presents a learning-based method for combining the shape and appearance feature types for 3D human pose estimation from single-view images. Our method is based on clustering the 3D pose space into several modular regions and learning the regressors for both feature types and their optimal fusion scenario in each region. This way the complementary information of the individual feature types is exploited, leading to improved performance of pose estimation. We train and evaluate our method using a synchronized video and 3D motion dataset. Our experimental results show that the proposed feature combination method gave more accurate pose estimation than that from each individual feature type.

1 Introduction

Image based human pose estimation systems have a wide range of applications, including content-based image retrieval, character animation, human computer interaction (HCI) and visual surveillance. 3D human pose estimation from monocular (single-view) images is challenging because it involves not only estimating a large number of pose parameters but also dealing with artifacts such as clutters, occlusions, viewpoint changes and pose ambiguities. In order to accurately estimate the 3D pose from a single image, relevant and informative features should be utilized.

Shape-based features such as silhouettes are insensitive to background variations. However, one silhouette can be associated with more than one pose, resulting in ambiguities. Although such ambiguities can be resolved using temporal constancy [[illegible]], such temporal information is not always available, e.g in the case of pose estimation from single image. While appearance-based features, e.g image textures, are more discriminative than shape features, they are unstable due to background clutters and variations in the clothing of the human subject. Therefore, shape and appearance features are not self-sufficient; instead, they complement each other. A proper combination of these two types of features is thus desirable, as demonstrated in this paper, to improve the performance of pose estimation.

There are two basic types of approaches to 3D pose estimation from such image features. The first type is the *generative method*, where the pose that best explains the observed

© 2010. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

BMVC 2010 doi:10.5244/C.24.51

image features is determined from a population of hypothesized poses, e.g. [4, 6]. The second type is the *discriminative method*, where a direct mapping between image features and pose parameters is learned using training data. Approaches of the first type are, in general, inefficient as the correct 3D pose must be searched in a high dimensional pose space. Discriminative pose estimation approaches, on the other hand, can estimate the pose quickly once the mapping function is trained. Some discriminative approaches use a single mapping from feature space to pose space[2]. Others use a piecewise linear regression approach where multiple regressors are trained in different parts of the pose space to approximate the non-linear mapping from the feature space to the pose space [1, 7, 11].

In this paper, we present a discriminative approach that fuses shape- and appearance-based features in a learning framework for 3D human pose estimation. To the best of our knowledge, the combination of these two types of features by a discriminative method has not been explored in the context of 3D human pose estimation. Our method is based on clustering the pose space into several modular regions and learning the regressors for both feature types and their optimal fusion scenario in each region. The optimal fusion scenario is learned using a confidence measure of each feature type in the region. The concept is intuitive because parts of the pose space may prefer shape features over appearance features whereas some other parts may prefer the opposite. This way the complementary information of the individual feature types is exploited, leading to an improved performance of the pose estimation system. Moreover, when one of the feature type is corrupt, for example, due to the presence of noise or occlusion, the other feature type will compensate, allowing the 3D pose to be successfully estimated. We evaluate our fusion method using the HumanEva I dataset [10] and compare our approach with the approaches of [8] and [7]. The evaluation results show that our method is superior to both of them.

The rest of the paper is organized as follows. In Section 2, we describe the feature descriptors used for pose estimation. Section 3 presents our proposed feature combination method. Section 4 provides the experimental results and Section 5 concludes the paper.

2 Feature Representation

Given an image, we first obtain the silhouette of a human subject using background subtraction [5]. We use *Histogram of Shape Context* (HoSC) [2] as our shape descriptor to encode the silhouette boundary of the human subject. First, *shape context (SC)* descriptors characterized by 12 angular bins and 5 radial bins are computed at 400 points regularly sampled along the boundary of the silhouette, as shown in Figure 1 (b). The HoSC descriptor is then constructed by *vector quantization,* which involves soft-voting each of the SC descriptors to each of the M entries of the SC feature dictionary. The SC feature dictionary is obtained once and for all by *k-means* clustering of the combined set of shape context vectors for all the training silhouettes. The voting are accumulated to obtain the M dimensional HoSC descriptor of the silhouette.

We use the *Histogram of Local Appearance Context* (HLAC) [9] as our appearance descriptor. The descriptor is defined in terms of the gradient orientation histograms of the image window containing the human subject. We obtain this image window using a bounding rectangle computed from the extermity of the silhouette. Alternatively, a human detector, such as [3], can also be used. Detailed description of the HLAC descriptor can be found in [9]. For the completeness of the paper, we briefly summarize the descriptor below. First, the image window is rescaled to the fixed size of 128×64. The HLAC descriptor of the image

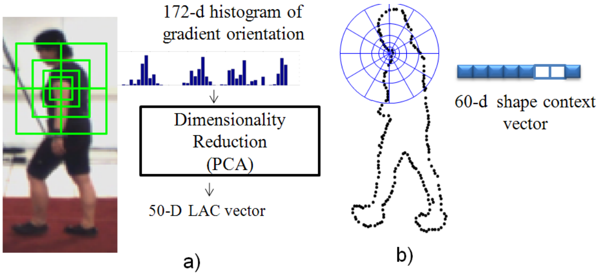

172-d histogram of gradient orientation
Dimensionality Reduction (PCA)
50-D LAC vector
a)
60-d shape context vector
b)

Figure 1: a) Examples of LAC descriptor used to construct the HLAC appearance descriptor [9] and b) *shape context* descriptor used to construct the HoSC shape descriptor [2].

window is then computed via two steps. In step 1, a *Local Appearance Context* (LAC) descriptor is computed on every 5th pixel sampled horizontally and vertically within the image window. To compute the *LAC* descriptor at a given point of an image region, the local region is partitioned into log-polar spatial blocks designated by 4 concentric squares having the respective widths of $12, 20, 30$ and 48 pixels, resulting in 16 spatial blocks as shown in Figure 1 (a). It is mentioned in [9] that this spatial block structure allows the descriptor to embody more contextual information of the local appearance features, making it more distinctive and robust to noise. The histogram of orientation of the image gradient computed in all of the 16 blocks are concatenated to form a single *LAC* vector of 172 dimensions, which are then reduced to 50 using PCA. In step 2, the histogram of these LAC vectors is computed via *vector quantization* to form an M-dimensional HLAC descriptor. This process is similar to the construction of the HoSC descriptor [2].

For both HoSC and HLAC descriptor, we found the optimal size of the feature dictionary (M) to be 200. For $M > 200$, the system did not exhibit any improvement on pose estimation. In all of our experiments, the dimensions of HLAC and HoSC descriptors were therefore set to 200.

3 Learning to Combine the Features

We use a supervised discriminative learning technique to estimate the 3D human pose from the shape and appearance descriptors extracted from an image. We learn a mapping from the shape descriptor space $\mathbf{x}_1 \in \mathbb{R}^{m_1}$ and the appearance descriptor space $\mathbf{x}_2 \in \mathbb{R}^{m_2}$ to the pose vector space $\mathbf{y} \in \mathbb{R}^d$ using the training data samples. The mapping function $\mathbf{y} = F(\mathbf{x}_1, \mathbf{x}_2)$ is a discriminative fusion model since it combines two feature types in discriminative fashion to estimate the pose. Once the fusion model is learned, the pose $\hat{\mathbf{y}}$ for a new image features $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ can then be estimated on the fly in the testing step.

Our proposed discriminative fusion model consists of a two-tier hierarchy. At the first tier, the 3D pose space $\mathbf{y}$ is partitioned into clusters using *k-mean*s algorithm. In second tier, the individual shape regressors, appearance regressors and their optimal fusion model are trained for each cluster. The motivation behind the first level partition is intuitive since different data spaces may have preferences for different fusion strategies. For example, in some regions, the shape regressor may get more weight than its appearance regressor counterpart. Our approach performs localized fusion of more than one feature type in each region. The relation between the two specific feature types $\mathbf{x}_1$ and $\mathbf{x}_2$ and the 3D pose vector

$\mathbf{y}$ is approximated by the following fusion model:

$$\mathbf{y} = \sum_{k=1}^{K} f_k(\mathbf{x}_1, \mathbf{x}_2) g_k(\chi), \tag{1}$$

where K is the number of clusters that the pose space is partitioned into and χ is an m-dimensional vector obtained by concatenating $\mathbf{x}_1$ and $\mathbf{x}_2$. The partition function $g_k(\chi)$ is a multi-class classifier which gives the probability that the k^{th} cluster is selected for a given feature instance. In each cluster, the fusion of the shape and appearance regressors is performed locally as follows

$$f_k(\mathbf{x}_1, \mathbf{x}_2) = \sum_{j=1}^{2} f_{kj}(\mathbf{x}_j) \beta_{kj}(\chi), \tag{2}$$

where f_{kj} is a linear regressor that takes a feature vector $\mathbf{x}_j$ and outputs the 3D pose estimate for the k^{th} cluster; β_{kj} is a combining function which outputs a weight that is proportional to the confidence measure of the regressor f_{kj}. Since we are combining only two feature types, the combiner function is approximated by a two-class classifier using logistic regressor. Hence in each cluster, fusion is done by convex combination of the regressors corresponding to each feature type.

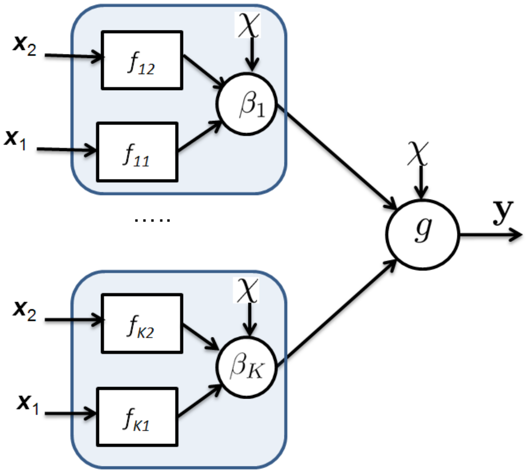


Figure 2: A block diagram showing our localized fusion method. The pose space is partitioned into K clusters and local fusion is performed for each cluster.

Figure 2 shows a block diagram for our localized fusion method. The pose space is partitioned into K clusters. For each cluster, the decision fusion of the local appearance regressors, f_{k1}, and the local shape regressor, f_{k2}, is performed using the weights given by the combiner function $\beta_k(\chi)$. The pose estimates from all the clusters are then aggregated using weights given by the partition function $g_k(\chi)$ to obtain the final output pose given by Equation (1). In the following subsection, we describe the parametric form of our regressors, partition function and combiner functions.

3.1 Model Parameters

The relation between each feature type and the 3D human pose in each cluster is approximated as $f_{kj}(\mathbf{x}_j) = \mathbf{W}_{kj}\mathbf{x}_j + \varepsilon_{kj}$, where $\mathbf{W}_{kj} \in \mathbb{R}^{d \times m_j}$ is a linear transformation matrix which maps $\mathbf{x}_j$ to $\mathbf{y}$; and ε_{kj} is a d-dimensional residual error vector. Modelling ε_{kj} as a Gaussian distribution with zero mean and covariance Λ_{kj}, the conditional pose distribution for the j^{th} feature type in the k^{th} cluster is $p_{kj}(\mathbf{y}|\mathbf{x}_j) = \mathbf{N}(\mathbf{W}_{kj}\mathbf{x}_j, \Lambda_{kj})$.

We model the combiner function, β_{kj}, by a logistic regressor. Since $\beta_{k2} = 1 - \beta_{k1}$,we only need to compute β_{k1}. Writing β_{k1} as β_k, the combiner function for the k^{th} cluster is approximated by $\beta_k(\chi) = 1/(1+\exp\,(-\lambda_k^T\chi))$ where λ_k is an m-dimensional weight vector.

We model the K-class classifier, g_k, by multiple two-class-classifiers using *one-against the rest* strategy. For each class, we train a two class classifier $g'_k(\chi) = 1/(1+\exp\,(-v_k^T\chi))$ to discriminate between the k^{th} cluster and the rest of the clusters, where v_k is an m-dimensional parameter weight vector. The posterior probability for the k^{th} class is computed as $g_k = g'_k/\sum g'_j$ to ensure $\sum g_k = 1$. Hence, our model consists of training of $2K$ linear regressors and $2K$ classifiers.

In this paper, we adopt the Relevance Vector Machine (RVM) [12] technique to train regressors and classifier because it gives a sparse model while maintaining good generalization performance. The RVM uses the Bayesian learning technique to find the weights of the regressors and classifiers from training data. It uses a Gaussian prior over the weight parameters, where the prior is controlled by an independent set of hyperparameters for each weight. The point estimate of the hyperparameters is done by optimizing the marginal likelihood (a.k.a type-II likelihood) [13]. The posterior distribution of the weights is then directly computed from the hyperparameter estimates. In RVM, the posterior distribution of most of the weights sharply peaks around zero. This implies that the weight values are zero for most of the feature components that are irrelevant to pose estimation. The RVM is particularly useful in our context, because only a subset of components of each feature type that are important to pose estimation is selected for each cluster during training. This sparse feature selection property of the RVM makes the fusion of two feature types more optimal.

3.2 Training steps of the fusion model

The parameters of our discriminative fusion to be estimated are $\mathbf{W}_{kj}$, Σ_{kj}, v_k, λ_k, for $k = 1, \cdots, K$; $j = 1, 2$. We use N training examples $\mathbf{T} = \{\mathbf{x}_1^{(i)}, \mathbf{x}_2^{(i)}, \mathbf{y}^{(i)}\}_{i=1}^N$ to estimate the value of these parameters. The training steps of our fusion model are given below.

1. Divide the pose space $\mathbf{Y} = \left\{\mathbf{y}^{(i)}\right\}_{i=1}^{N}$ into K clusters using *k-means*.

2. For each cluster, train each linear RVM regressors f_{kj} with the training set $\left\{\mathbf{x}_j^{(i)}, \mathbf{y}^{(i)}\right\}$, for all i, such that $y^{(i)} \in k^{\text{th}}$ cluster using the method described in [13]. This gives a weight matrix $\mathbf{W}_{kj}$. The covariance matrix is obtained from the error $\mathbf{e}_{k,j}^{(i)} = \mathbf{y}^{(i)} - \mathbf{W}_{kj}\mathbf{x}_j^{(i)}$ as $\Lambda_{kj} = \frac{1}{N_k}\sum_{i=1}^{N_k}\left(\mathbf{e}_{k,j}^{(i)}\right)\left(\mathbf{e}_{k,j}^{(i)}\right)^T$, where N_k is the number of training data that belongs to the k^{th} cluster. Since the output dimensions are assumed to be independent, we retain only the diagonal elements of these covariance matrices.

3. For each cluster, train the RVM classifier β_k with the training data $\left\{\chi^{(i)}, c_k^{(i)}\right\}$, for all

i, such that $y^{(i)} \in k^{th}$ cluster using the method described in [13]. This gives the weight vector λ_k. The confidence measure of $c_k^{(i)}$ of the regressor f_{k1} on the k^{th} cluster is computed as,

$$c_k^{(i)} = \frac{p_{k1}(\mathbf{y}^{(i)}|\mathbf{x}_1^{(i)})}{\sum_{j=1}^{2} p_{kj}(\mathbf{y}^{(i)}|\mathbf{x}_j^{(i)})}. \tag{3}$$

4. Train the two class RVM classifier for the k^{th} cluster, g'_k, with the training data $\left\{\chi^{(i)}, l_k^{(i)}\right\}_{i=1}^{N}$ where $l_k^{(i)} = 1$ if $\mathbf{y}^{(i)} \in k^{\text{th}}$ cluster otherwise 0 using the method described in [13]. This gives the weight vector v_k. The value of g_k can be obtained by normalizing the value of g'_k as $g_k = g'_k / \sum_{j=1}^{K} g'_j$.

Once the model is trained, we can use Equation (1) to estimate the pose from the feature types $\mathbf{x}_1$ and $\mathbf{x}_2$ in the testing phase.

4 Results

We trained and evaluated our proposed 3D human pose estimation method using the HumanEva I data set [10] provided by Brown University. The data set contains video frames and corresponding 3D poses of three subjects carrying out actions such as walking, jogging and boxing. Altogether, our training set consists of 3600 images and our test set consists of 3864 images for camera views C1, C2 and C3. For each image, the corresponding ground truth 3D pose is given by the *x,y,z* coordinates of the 14 human body joint locations relative to the location of the pelvis joint. The dimension of a pose vector is thus 42. The data set was originally partitioned into training, validation and testing sets. However, since the ground truth of the testing set was not provided, we used the validation set as our test set. Table 1 shows the number of images in the training and testing sets for each action.

Table 1: Snapshot of the training and testing sets

Event	Training	Testing
Walking	1528	1476
Jog	1027	1155
Box	1045	1233
All	3600	3864

For all images in the training and the testing sets, we extracted 200 dimensional HLAC appearance descriptor vectors and 200-dimensional HoSC shape descriptor vectors following the process described in Section 2 on page 2. We trained the proposed discriminative fusion framework using the extracted shape descriptor vectors, appearance descriptor vectors and the corresponding pose vectors of all the images in training set using the training steps described in Section 3.2. We also trained the regressors using the shape and appearance descriptors individually. For all of these cases, we found the optimal number of clusters to be $K = 5$. For $K > 5$ the pose estimation performance did not improve.

We then predicted the poses of the images in the testing set using the trained models. The final output of the fusion model is a 42-dimensional pose vector, which collectively denotes a particular pose. We then computed the absolute error between the estimated pose $\hat{\mathbf{y}}$ and the ground-truth pose $\mathbf{y}$, each of dimension d to evaluate the performance of our approach. The absolute error is given by $e = \frac{1}{d}\sum_{i=1}^{d}|y_i - \hat{y}_i|$, which measures the mean absolute distance per joint in *mm* between the ground truth pose and the estimated pose.

Table 2 compares the average error of the regressors trained on each feature type alone with the average error of their fusion using our approach for each test action sequence and for each camera view. These experimental results show that our proposed fusion method gives much lower average error than the average error of individual shape and appearance regressors. The lower standard deviation suggests that the poses estimated using our fusion method have higher confidence values than those poses estimated using each shape and appearance regressor alone. Figure 3(a)-3(d) respectively shows the affinity matrix of the ground truth poses, the output poses estimated from the shape features alone, the appearance features alone, and our proposed fusion method. It can be observed that the output poses of the shape regresssor are different from the ground truth for most of the frames. This is because the shape features output incorrect poses due to forward backward ambiguities and self occlusions. For the appearance regressor, these ambiguous estimates are corrected to some extent but there are still inaccurate pose estimates due to clutters and noise included in appearance features. The accuracy of the pose estimates is significantly improved when the two features are combined as the affinity matrix of the poses estimated using our proposed fusion method is more similar to the affinity matrix of the ground truth poses.

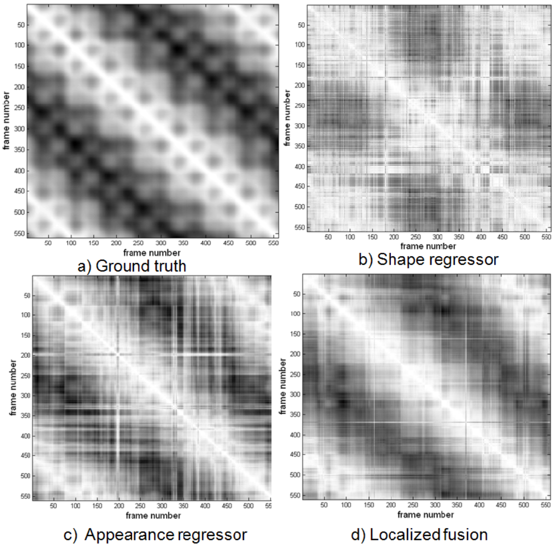
Table 2: Comparison of the average absolute test errors and the standard deviation (shown in bracket) in *mm* for the shape regressor, appearance regressor and our proposed localized fusion method. The comparisons are shown for each of the test action set and their combination set for each view C1, C2 and C3.

Camera	Actions/Feature Types	Fusion	Shape	Appearance
C1	Walking	**26.96** (**12.23**)	39.58 (19.22)	32.22 (14.99)
	Jog	**27.40** (**13.61**)	48.74 (23.99)	29.82 (13.67)
	Boxing	**32.54** (**14.13**)	42.29 (18.62)	42.32 (22.06)
	Combined	**29.89** (**14.56**)	41.34 (20.74)	36.42 (15.12)
C2	Walking	**24.92**(**11.34**)	39.04(22.49)	29.35(15.20)
	Jog	**26.60**(**16.01**)	40.04(20.00)	30.43(13.90)
	Boxing	**35.96**(**22.79**)	45.51(24.21)	36.94(24.03)
	Combined	**28.68**(**14.85**)	39.43(21.62)	33.70(16.09)
C3	Walking	**24.56**(**11.20**)	36.69(20.95)	28.77(14.46)
	Jog	**25.12**(**12.08**)	40.72(19.27)	31.15(17.23)
	Boxing	**33.65**(**19.46**)	44.89(23.78)	34.56(17.14)
	Combined	**30.03**(**17.92**)	40.07(19.50)	33.27(14.22)

Table 3: Comparison of the average absolute errors in (*mm*) of our approach with other approaches. The training and testing sets are taken from the Walking sequence of camera C1 for each subject.

Method/Subject	S1	S2	S3
Pope [8]	41.24	39.56	55.27
Okada[7]	41.19	35.03	37.69
Our method	**32.71**	**22.07**	**36.52**

We compared the performance of our approach with the approaches of [7, 8] in Table 3. Both approaches [7, 8] use the HOG [3] based appearance descriptor. We used the walking



a) Ground truth

b) Shape regressor

c) Appearance regressor

d) Localized fusion

Figure 3: Comparison of the ground truth and estimated poses using affinity matrices. Affinity matrix of a) the ground truth poses, b) the estimated poses using shape regressor, c) the estimated poses using the appearance regressor, and d) the estimated poses using our proposed fusion method. The brightness of the pixels in each matrix denotes the degree of similarity of the two poses. The off-diagonal similarity is due to circular motion of the subject. The ground truth and estimated poses are taken from test walking sequences of the subject S1 and the camera C1.

sequence of the three subjects S1, S2 and S3 taken from camera C1 for comparison. The same set was used by [7, 8]. The result shows that our method gives lower errors than both approaches. Figure 4 compares estimated poses as the 3D human body model along with the estimation error for our fusion and non-fusion cases. These results show that the estimated poses using each individual shape and appearance regressor have larger errors compared to the poses estimated using our fusion method. Similarly, Figure 5 displays some of the output poses predicted using our fusion method alongside with their estimation errors. Our experiments show that the proposed fusion method can effectively combine more than one feature type to improve the pose estimation performance.

5 Conclusions

We have presented a learning-based discriminative fusion method for combining shape and appearance features for 3D human pose estimation. Our method effectively combines the shape and appearance feature types by learning their locally optimal fusion. We have experimentally shown that our proposed fusion method produced more accurate pose estimations in comparison with those from shape and appearance features alone. Our fusion model is effective because it combines different feature types by first learning the input specific fusion strategy. Although this paper describes an application of the proposed fusion method to 3D human pose estimation, our method is equally applicable to other systems using dis-

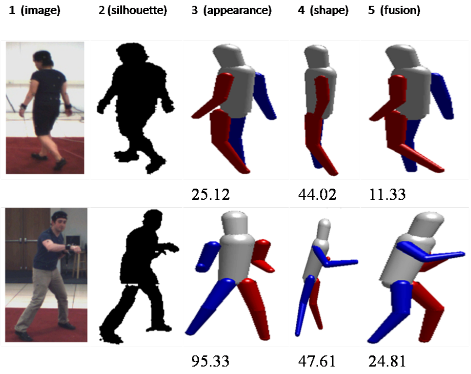


Figure 4: a) Visual comparison of the 3D poses estimated from appearance features (shown in column 3), the shape features (shown in column 4) and their fusion using our method (shown in column 5). The 3D poses are rendered as cylinders with red denoting left limbs and blue denoting right limbs. Below each output pose, the estimation error (in *mm*) is shown. It is evident that fusion of two types of the features improves the accuracy of the pose estimation.

criminative prediction models. The advantage of our approach is that the human pose can be estimated quickly. However, to learn the fusion model, a large number of training images and ground truth poses are required. While images can be easily acquired, ground truth poses are difficult to obtain. To overcome this problem, we are currently looking to extend the technique using a semi-supervised learning so that a more accurate fusion model can be trained by utilizing the training images that do not have the ground truth poses.

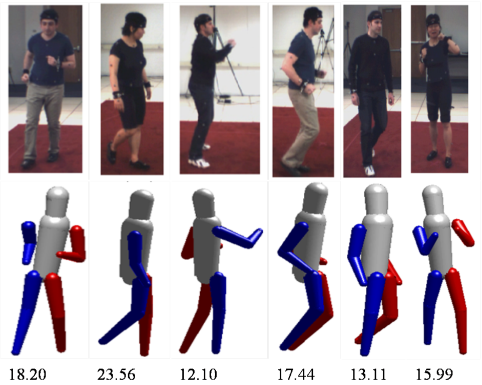


Figure 5: Some images from the test set and their corresponding 3D poses estimated by fusion of the shape and appearance feature types. Below each output pose, the estimation error (in *mm*) with respect to the ground truth pose is shown.

References

[1] A. Agarwal and B. Triggs. Monocular human motion capture with a mixture of regressors. *Computer Vision and Pattern Recognition*, 3:72–72, 2005.

[2] Ankur Agarwal and Bill Triggs. Recovering 3D human pose from monocular images. *IEEE Transactions on PAMI*, 28(1), 2006.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *IEEE Conference on CVPR*, 1:886–893 vol. 1, 2005.

[4] J. Deutscher and I. D. Reid. Articulated body motion capture by stochastic search. *Int'l Journal of Computer Vision*, 61(2), 2005.

[5] A. Elgammal, R. Duraiswami, D. Harwood, and L.S. Davis. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *Proceedings of the IEEE*, (7), 2002.

[6] Mun Wai Lee and I. Cohen. A model-based approach for estimating human 3D poses in static images. *IEEE Transactions on PAMI*, 28(6):905–916, 2006.

[7] Ryuzo Okada and Stefano Soatto. Relevant feature selection for human pose estimation and localization in cluttered images. In *European Conference on Computer Vision*, pages 434–445, 2008.

[8] R.W. Poppe. Evaluating example-based pose estimation: Experiments on the humaneva sets. In *Workshop on EHuM at CVPR*, pages 1–8, June 2007.

[9] Suman Sedai, Mohammed Bennamoun, and Du Huynh. Context-based appearance descriptor for 3D human pose estimation from monocular images. In *DICTA*, pages 484–491, 2009.

[10] Leonid Sigal and Michael J. Black. HumanEva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical Report CS-06-08, Brown University, 2006. Honeywell Research.

[11] Cristian Sminchisescu, Atul Kanaujia, Zhiguo Li, and Dimitris Metaxas. Discriminative density propagation for 3D human motion estimation. In *IEEE Conference on CVPR*, pages 390–397, 2005.

[12] Michael E. Tipping. The relevance vector machine. *Advances in neural information processing systems*, 12:652–658, 2000.

[13] Michael E. Tipping and Anita Faul. Fast marginal likelihood maximisation for sparse bayesian models. In *Proceeding of the AISTATS*, pages 3–6, 2003.