Discriminative 3D Human Pose Estimation from Monocular Images via Topological Preserving Hierarchical Affinity Clustering

Weiwei Guo
Queen Mary, University of London, UK
National University of Defense Technology, China
weiwei.guo@elec.qmul.ac.uk

Ioannis Patras
Queen Mary, University of London, UK
i.patras@elec.qmul.ac.uk

Abstract

This paper presents a hierarchical approach to address the problem of 3D human body pose estimation from a single images. In order to deal with multimodality, we learn piecewise mappings from observations to human poses. We first construct a tree on the pose manifold by applying affinity propagation clustering at the different levels of the hierarchy. Support vector machines classifiers are then trained to learn traversing the tree so that new examples/observations can be classified to the clusters associated to the leaf nodes. Multi-valued Relevance Vector Machine (RVM) regressors are trained at each of the leaf nodes, so to learn local mappings from the observation to the pose space. We propose the use of a geodesic distance during clustering and describe a method for training multi-valued RVMs. The latter alleviates the need to train a separate RVM for each of the dimensions in the pose space. We validate the proposed method using the HumanEva dataset and show promising results.

1. Introduction

This paper addresses the problem of 3D human pose estimation from a single image. This is an active field in computer vision and has numerous applications, including human-computer interaction and surveillance. The approaches to the problem fall into two categories, namely the generative one and discriminative one. Methods that belong to the generative category, model the observations as a generative process from the pose parameters. Estimation is performed by optimizing over the pose space a likehood or in general a fitness measure between the images/observations predicted by the model and the actual observations. Such methods are computationally expensive and without good initialization get easily trapped into local extrema. This is due to the high dimensionality of the pose and the observation space, and due to the high complexity of the mapping due to (self)occlusions, background clutter etc. By contrast to the generative methods, the discriminative ones, learn a direct mapping from the observation space to the pose space, that is from visual observations to articulated body configurations. The mapping is learned in a supervised way utilizing a training data set. This category of methods has recently received considerable attention due to its simplicity, its computational efficiency and the fact that it does not require a good initialization.

Learning a mapping from the visual observations to body configuration is a challenging task. This is due to the high dimensionality of the state space and due to the fact that the conditional distribution of the body configuration given the observations is multi-modal. The later implies that the mapping is one to many. Recently some discriminative methods address the issue of multi-modality by clustering the data and learning different mappings for each cluster. A similar approach is to construct an observation-state database for near neighborhood retrieval of poses with similar visual appearance on which local regressors can be learned. Sminchisescu *et al*. [16] employ a Bayesian Mixture of Experts that deliver a probabilistic prediction in the form of a mixture of Gaussians. Other methods [6, 5] reduce the complexity of the problem by learning a low dimensional manifold on which certain human actions lie.

In this paper, we introduce an efficient discriminative method for human pose estimation that relies on a hierarchical clustering algorithm and local regressors. In the training stage, a pose tree is learned by affinity propagation clustering using a geometric similarity measure in the pose space. After the pose tree structure is constructed, multi-class classifiers are learned on the observation space at each non-leaf node, so that a new example can be classified to a leaf node by following the appropriate path in the tree. At each leaf node, a local regressor is learned in a supervised way during the training phase. During the test phase, once features are extracted, a new example traverses the tree by applying the learned classifiers at each non-leaf node. Finally, the pose is estimated by applying the learned regressor that is attached

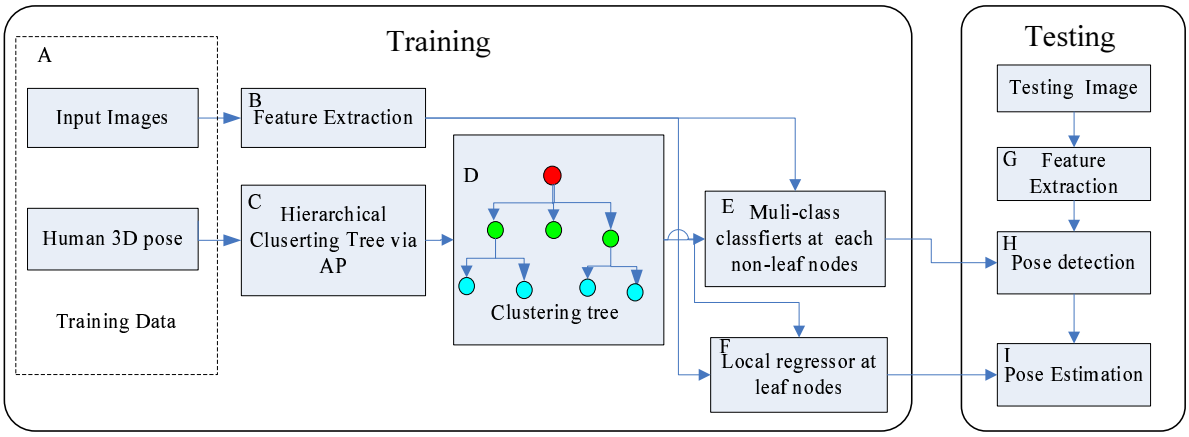

Figure 1. Overview of the proposed method.

to the leaf node to which the example is classified.

The overview of the proposed method is given in Fig. 1. The main contributions of our method are

- We propose the use of a recently developed clustering method for the bottom-up construction of the pose tree, namely the "Affinity propagation" [7] algorithm. In contrast to the widely used K-means clustering, it provides a mechanism that allows automatic selection of the number of clusters and produces better cluster centers in terms of the mean square error.
- We propose the use of a similarity measure that is based on geodesic distances to measure the affinity between the two poses. In contrast to the widely used Euclidean distance it preserves the manifold structure in the pose space. The motivation is the fact that human poses related to activities such a walking, running etc, lie on a low dimensional manifold in a high dimensional space.

The remainder of the paper is organized as follows. We review the related work on human pose estimation in section 2. Section 3 explains the hierarchical clustering by affinity propagation using the proposed geometric similarity, and gives a brief description of the multi-class SVMs that are used for discriminating between pose clusters. Section 4 describes the proposed extension for learning a multi-output Relevance Vector Machine regresssor using Expectation Maximization. In Section 5 we present the experimental results on the HumanEva–I dataset to validate our method. Finally, in Section 6 we draw some conclusions.

2. Related work

Regression-based human pose estimation has received extensive attention in the recent years due to the significant advantages in comparison to the generative methods, so far as the dependency on a good initialization and the computation efficiency are concerned. Agarwal and Triggs [1] used a sparse kernel regression to map human silhouettes to their corresponding pose coupled with temporal information to disambiguate estimates. In [16], Sminchisescu,*et al*. represented the conditional distribution of the pose given an observation as a Mixture of Experts. This is achieved by coupling a soft partitioning of the observation space with regressors/experts associated with each partition. Besides the direct functional approximation, local regression based on fast nearest neighborhood search have been proposed in [19, 15]. To speed up finding similar observations, an alternative method is to organize the data using trees. Gavrila [8] presents a probabilistic hierarchical exemplar-based shape detection approach for pose detection. While good detection performance is reported, the accuracy of detection-based methods is intrinsically limited by the granularity at which the detection 'pose classes/clusters' are determined. Recently, several authors (e.g. [6, 5, 9] exploited the fact that human motions and their corresponding appearances lie on a low dimensional manifold in a high dimensional space. The latent variables over the manifolds are used as an intermediate representation that connects the observation and the states, and can reduce the complexity and improve the generalization performance.

The work presented in this paper bears similarities to the works presented in [14] and [12]. In [14], Rogez *et al*. cluster the data in a hierarchical tree on a discretized torso manifold of poses. Subsequently, randomized forests are used to select optimal features for discriminating between these clusters. Okada *et al*. [12] clustered the poses using the K-means algorithm and used a kernel SVM that can select the relevant'pose-dependent' features to discriminate between the clusters. Here, we propose the construction of a pose tree by clustering using the Affinity Propagation algorithm in an hierarchical way. The latter is a recently proposed

method that is shown to significantly outperform K-means [7] especially for large scale problems, and provides an intuitive mechanism for the automatic selection of the number of clusters. Then a multi-class linear SVM is utilized to discriminate between the clusters. In contrast to [14], the proposed method does not require alignment of the original observation images. In addition, we describe a method for learning a multi-output sparse RVM-regressor instead of learning separate regressors over each output dimension within each clustered region.

3. Topological Preserving Hierarchical Clustering and Discrimination

Because the high dimensional pose states and their appearances exhibit multi-modal distributions, the inverse mapping from observations to the pose states is multi-modal. Therefore, most of discriminative methods propose piecewise mapping functions where each partition/cluster has its own mapping. Clearly, obtaining good clusters is essential for obtaining good approximations. Gavrila [8] addressed this NP-hard problem using Simulated Annealing, at the drawback of a high computation cost. K-means is a widely-used clustering algorithm but faces the challenging problem of automatic selection of the number of clusters, especially in high dimensional space. In this paper, we advocate the use of the recently developed "Affinity Propagation" (AP) algorithm, which poses the clustering problem as the one of selecting a number of prototypes such that the sum of similarities between the points and their exemplar prototypes is maximized [7]. This is achieved by posing the problem as inference in a factor-graph and finding in this way approximate MAP solutions of "prototypes" using the max(sum)-product algorithm. A hierarchical clustered tree can be formed by applying the AP algorithm recursively. In the next section, we introduce the affinity propagation clustering using a similarity measure that preserves the topology of the pose manifold. Subsequently, we outline the use of Support Vector Machines for traversing the tree and classifying a new example into one of the leaf nodes.

3.1. Topological Preserving Affinity Propagation on Manifold

Affinity propagation clustering requires the definition of a pairwise similarity measure in the appropriate space. Although the body pose space is clearly high dimensional it is argued that the postures during activities, like a walking, lie on a low dimensional manifold. To capture the topological structure of body configurations on the manifold we define the pairwise similarity measure $Sim(i,j)$ between two points i,j on the manifold, as their negative geodesic $d_{\mathcal{M}}(i,j)$ distance. The latter is calculated by computing the shortest path distance $d_G(i,j)$ in the near neighborhood graph, that is

$$Sim(i,j) = -d_G(i,j) = -d_{Dijkstra}(i,j) \tag{1}$$

In what follows, we review the Affinity Propagation algorithm, the details of which can be found in [7]. The goal is to select a number of prototype data points such that

$$\mathcal{S}(\mathbf{c}) = \sum_{i}^{N} Sim(i,c_i) + \sum_{k}^{N} \delta_k(\mathbf{c}) \tag{2}$$

$$\delta_k(\mathbf{c}) = \begin{cases} -\infty & \text{if } c_k \neq k \quad \text{but } \exists i : c_i = k \\ 0, & \text{otherwise} \end{cases}$$

is maximized. This is achieved by an iterative exchange of two kinds of competitive messages between data points i and k: "responsibility" $r(i,k)$, and the "availability" $a(i,k)$. The "responsibility"$r(i,k)$ is sent from a point i to a candidate exemplar point k, reflecting the accumulated evidence for how a well-suited point k is to serve as the exemplar point i, while the "availability"$a(i,k)$ sent from the candidate exemplar point k to the point i, reflects the accumulated evidence for how approximate it would be for point i to choose point k as its exemplar. To begin with, availabilities are initialized to zero. The responsibilities are computed using the rule

$$r(i,k) \leftarrow s(i,k) \; - \max_{k' \; s.t. k' \neq k} \left\{ a(i,k') + s(i,k') \right\} \tag{3}$$

For $k = i$, the self-responsibility $r(k,k)$ is set to the input preference that a point k is chosen as an exemplar tempered by how ill-suited it is to be assigned to another exemplar. The availability updates as

$$a(i,k) \leftarrow \min \left\{ 0, r(k,k) + \sum_{i' \; s.t. i' \notin \{i,k\}} \max \left\{ 0, r(i',k) \right\} \right\} \tag{4}$$

This update rule gathers evidence from the data point k as to whether each candidate exemplar would make a good exemplar. The self-availability update differently:

$$a(k,k) \leftarrow \sum_{i' \; s.t. i' \notin \{i,k\}} \max \left\{ 0, r(i',k) \right\} \tag{5}$$

This message reflects accumulated evidence that point k is an exemplar based on the positive responsibility sent to the candidate exemplar k from others. After convergence, for point i, its corresponding exemplar is

$$k^* = \arg\max_{k} \left\{ a(i,k) + r(i,k) \right\} \tag{6}$$

To obtain hierarchial exemplars, after the original clustering, we run AP iteratively using as data points the exemplars obtained by clustering at the lower level of the hierarchy. The process is continued until a pre-defined level is reached.

3.2. Cluster Discrimination using Multi-class SVM

The constructed tree contains clusters of human poses at different levels of resolution. Once it is constructed, a methodology is needed so that a novel example/image traverses the tree and is classified to a cluster that is associated to one of the leaf nodes. Clearly, the decision of which branch to follow should be based solely on the features extracted from the images. We pose the branching decisions as a multi-class classification problem, which is solved by training a multi-class linear SVM [3] at each non-leaf node of the tree.

More specifically, given a set of N training examples $\mathcal{D} = \left\{(x_i, y_i)\right\}_{i=1}^{N}$, each comprising of the features $x_i \in \mathcal{X} \subseteq \mathbb{R}^n$ and the associated label $y_i \in \mathcal{Y} = \{1, 2, \ldots, k\}$, we take the multi-class classifier of the form

$$\boldsymbol{H}_{\mathbf{M}} = \arg \max_{r=1,\ldots,k} \{\boldsymbol{M}_r \cdot x\}, \tag{7}$$

where $\mathbf{M}$ is a matrix of size $k \times n$ and $\boldsymbol{M}_r$ is the r^{th} row of $\mathbf{M}$. The optimal weight matrix $\mathbf{M}$ can be found using the max-margin criterion through standard quadratic programming techniques [3].

4. Local Regression using Multi-output RVM

In this section, we describe a multi-output Relevance Vector Machine (RVM) that is trained to map observations to states at each leaf node. The RVM is a regression method in the Bayesian framework, in which a prior distribution is introduced over the weights of the input examples that encourages most of weights to approach to zero. In comparison to Support Vector Machines this leads to considerably sparser representations whilst maintaining comparable generalization error. However, the original RVM proposed by Tipping in [18] maps the multidimensional input to a scalar output. For another kernel sparse regression technique—Gaussian Process, it is difficult to be extended to the case of multi outputs [2]. One approach for multi-output regression is to build separate RVMs for each dimension of the output. One major drawback of this solution is that this leads to separate sets of relevance vectors for each each output. Here, we propose to utilize the associated sparse weights in order to retain the same set of relevance vectors for all the dimensions of the output (i.e. the state).

In what follows, we derive a multi-output RVM in the EM framework with a slightly difference from the one presented in [17]. Our multi-output RVM is trained in a batch mode while the one presented in [17] is trained sequentially. Suppose we are given a set of N observations of input vector $\boldsymbol{x}$, denoted collectively by a data matrix $\mathbf{X}$ whose n^{th} row is $\boldsymbol{x}_n^T$ with $n = 1, \ldots, N$. The corresponding target values are given by $\mathbf{T} = (\boldsymbol{t_1}, \ldots, \boldsymbol{t_N})$ with $\boldsymbol{t}_i \in \mathbb{R}^L$. The output and input are linked by a generative model given by

$$\mathbf{T} = \mathbf{\Phi}\mathbf{W} + \epsilon, \tag{8}$$

where $\mathbf{\Phi}$ is the $N \times M$ design matrix with elements $\phi_{ni} = \phi_i(\boldsymbol{x}_n)$, ϵ is noise, and $\mathbf{W} = (\mathbf{w}_{\cdot 1}, \ldots, \mathbf{w}_{\cdot L}) = (\mathbf{w}_{1\cdot}, \ldots, \mathbf{w}_{M\cdot})$. Thus, the likehood function is given by

$$p\left(\mathbf{T}|\mathbf{X}, \mathbf{W}, \boldsymbol{\beta}\right) = \prod_{n=1}^{N} p\left(\mathbf{t}_n|\boldsymbol{x}_n, \mathbf{W}, \beta\right) \tag{9}$$

Next, we assume that $\mathbf{W}$ is row-sparse to encourage few nonzero rows in $\mathbf{W}$ by introducing a sparse prior with a set of separate hyperparameters shared by columns, thus taking the form

$$p\left(\mathbf{W}|\alpha\right) = \prod_{l=1}^{L} \mathcal{N}(\mathbf{w}_{\cdot l}|\mathbf{0}, \mathbf{A}) = \prod_{l=1}^{L}\prod_{m=1}^{M} \mathcal{N}\left(\mathbf{w}_{ml}|\mathbf{0}, \alpha_m^{-1}\right) \tag{10}$$

where $\mathbf{A} = diag(\alpha_m)$ and $\boldsymbol{\beta} = \{\beta_l\}_{l=1}^{L}$ is a set of row-shared hyperpaprameters. The complete-data log-likelihood function is then given by

$$\begin{aligned} \ln p\left(\mathbf{T}, \mathbf{W}|\boldsymbol{\alpha}, \boldsymbol{\beta}\right) \quad = \quad & \sum_{l=1}^{L}\left\{\sum_{n=1}^{N} -\frac{\beta_l}{2}\left(t_{nl} - \boldsymbol{\phi}^T(\boldsymbol{x}_n)\mathbf{w}_{\cdot l}\right)^2 \right. \\ & + \frac{N}{2}\ln\beta_l - \frac{1}{2}\mathbf{w}_{\cdot l}\mathbf{A}\mathbf{w}_{\cdot l}^T \\ & \left. + \frac{1}{2}\ln|\mathbf{A}|\right\} + const \end{aligned} \tag{11}$$

Here for simplicity we drop the conditional variable $\mathbf{X}$ from Eq. 11. By completing the square of Eq. 11, we can see the posterior distribution over weights is hence given by

$$p\left(\mathbf{w}_{\cdot l}|\mathbf{T}, \mathbf{A}, \boldsymbol{\beta}_l\right) = \mathcal{N}\left(\mathbf{w}_{\cdot l}|\mathbf{m}_l, \Sigma_l\right), \quad l = 1, \ldots, L \tag{12}$$

where the mean and covariance are

$$\mathbf{m}_l = \beta_l \Sigma_l \mathbf{\Phi}^T \mathbf{t}_{\cdot l} \tag{13}$$

$$\Sigma_l = \left(\mathbf{A} + \beta_l \mathbf{\Phi}^T \mathbf{\Phi}\right)^{-1} \tag{14}$$

Taking the expectation of Eq. 11 with respect to the posterior distribution $\mathbf{W}$ then gives

$$\begin{aligned} & \mathbb{E}\left[\ln p\left(\mathbf{T}, \mathbf{W}|\boldsymbol{\alpha}, \boldsymbol{\beta}\right)\right] = \\ & \sum_{l=1}^{L}\left\{\sum_{n=1}^{N} -\frac{\beta_l}{2}\mathbb{E}\left[\left(t_{nl} - \boldsymbol{\phi}^T(\boldsymbol{x}_n)\mathbf{w}_{\cdot l}\right)^2\right]\right. \\ & \left. + \frac{N}{2}\ln\beta_l - \frac{1}{2}\mathbb{E}\left[\mathbf{w}_{\cdot l}\mathbf{A}\mathbf{w}_{\cdot l}^T\right] + \frac{1}{2}\ln|\mathbf{A}|\right\} + const \end{aligned} \tag{15}$$

Setting the partial derivatives with respect to $\alpha_m, m = 1, \ldots, M$, $\beta_l, l = 1, \ldots, L$ to zero, we obtain the follow-

ing M step hyperparameters re-estimation equations

$$\alpha_m = \frac{\sum_{l=1}^{L} \gamma_{lm}}{\sum_{l=1}^{L} m_{lm}^2}, \qquad m = 1, \ldots, M \tag{16}$$

$$\beta_l = \frac{\| \mathbf{t}_{\cdot l} - \mathbf{\Phi} \mathbf{w}_{\cdot l} \|^2}{N - \sum_{m=1}^{M} \gamma_{lm}}, \quad l = 1, \ldots, L \tag{17}$$

where γ_{lm} is defined as follows:

$$\gamma_{lm} = 1 - \alpha_m \left(\Sigma_l\right)_{mm} \tag{18}$$

In the equation above, $\left(\Sigma_l\right)_{mm}$ is the m^{th} diagonal component of the posterior covariance Σ_l given by Eq. 14 with $l = 1, \ldots, L$, and $m = 1, \ldots, M$.

5. Experimental Results

We test our approach on the benchmark dataset HumanEva-I [11]. The dataset contains synchronized multi-view video and mocap data and depicts 3 subjects performing multiple activities. Here, we use the data from a single camera(C1), from the walking sequence for all three subjects. The training subset contains 612, 433 and 479 examples for S1, S2 and S3 respectively. The test subset contains 599, 433 and 412 examples for S1, S2 and S3 respectively.

Our features are based on the Histogram of Oriented Gradients [4]. Here, we extend it to hierarchical multi-level encoding as in [10]. However, while in [13], the HoG is extracted with a fixed spatial resolutions, here we vary the spatial resolutions to calculate HoG at three pyramid levels with 9 angular bins of gradient orientation $(0 \sim 180^\circ)$, unsigned. The final dimensionality of descriptors is 765. Similar to [13], the features are calculated within a region of interest (ROI). The latter is extracted by background substraction and shadow removal. Pose is encoded by the relative with respect to the location of the pelvis (torsoDistal)joint location of 19 3D joints. The errors are measured by the RMS absolute error between the ground truth $\boldsymbol{T}$ and the estimated locations of joints $\hat{\boldsymbol{T}}$ [11], in mm. That is,

$$D\left(\boldsymbol{T} - \hat{\boldsymbol{T}}\right) = \frac{1}{J} \sum_{j=1}^{J} \|\boldsymbol{t}_j - \hat{\boldsymbol{t}}_j\| \tag{19}$$

where $\boldsymbol{t}_j \in \mathbb{R}^3$ is the location of the j^{th} 3D joint, and $\boldsymbol{T} = \{\boldsymbol{t}_1, \ldots, \boldsymbol{t}_J\}$.

In Table 1, we summarize the quantitative results. The empirical probabilistic density functions of the error are also shown in Fig. 3. In order to demonstrate the influence of the choice of an appropriate similarity measure we report results for both a similarity measure that is based on the Euclidean distance and the proposed similarity that is based on a geodesic distance. In order to demonstrate the influence of the choice of the appropriate space on which the tree is constructed, we present results for trees constructed in two different spaces, the pose space and the observation space. It is clear that the geodesic distance outperforms the Euclidean distance and that clustering in the observation space is worse than clustering in the pose space. Let us here recall that in the proposed scheme an example traverses the classification tree and the pose is estimated by applying the regressor that is associated to the leaf node that the example in question reaches. In Table 1, next to the RMS error, we report estimation error in the case that the examples would end up to the 'correct' leaf node, that is if the classification problem was solved in the optimal way. It is interesting to note that quite similar results are reported for all variations using the geodesic distance. This indicates that the difference in performance should be mainly attributed to the complexity of the corresponding classification problems. Finally, in parentheses we report the number of the leaf nodes/clusters.

For comparison, we report the results using two baseline methods that are based on K-Nearest Neighbors (K taken to be 20 in our experiments). The first one [13], denoted by 'local WKNN', makes an estimation by taking the weighted average of the K nearest neighbours in the training set. We also presented the second one, denoted by 'Local KNN-RVM', which trains an RVM on the data from the K Nearest Neighbours, and uses it to make the prediction for the example in question. Fig.2 shows the estimation errors with different number of nearest neighbours. The comparison in Fig. 2 clearly shows the benefits of a local regression scheme in comparison to simple weighted NN [13], especially when a large number of nearest neighbours is used. Also, Table 1 shows that regression schemes that are based on clustering outperform both NN regression and WKNN. Note that the results on the baseline (Local WKNN) system are worse than the ones presented in [13] using the same approach, most probably due to the fact that we use a smaller number of training examples. However, by examining Table 1, it is clear that there is a significant difference in comparison to the proposed approach.

Space	Pose	Observation
Geometric	52.65/41.28(43)	60.47/38.59(147)
Euclidean	56.76/39.15(63)	65.40/36.89(194)
Baseline	Local WKNN	68.30
	Local KNN-RVM	60.11

Table 1. 3D pose estimation error (RMS in mm). We report the estimation error, the error if the example was classified to the best possible leaf node and in parentheses, the number of leaf nodes.

In Fig. 4, we superimpose a simple 3D model on the 2D observations. While the 3D to 2D projection is not accurate, as the 3D model is not meant to be used with a generative

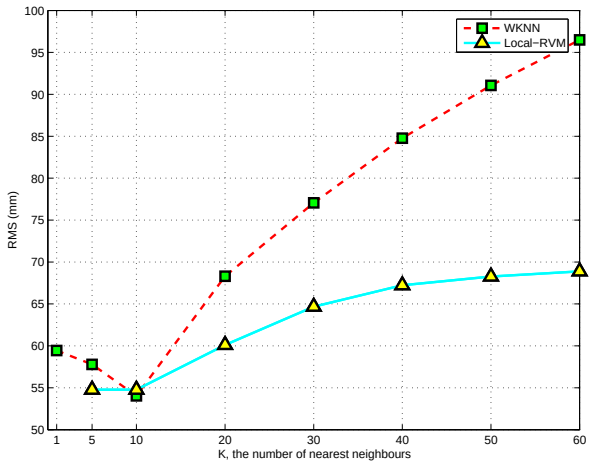

Figure 2. The estimation errors of WKNN and KNN-RVM using different numbers of nearest neighbours

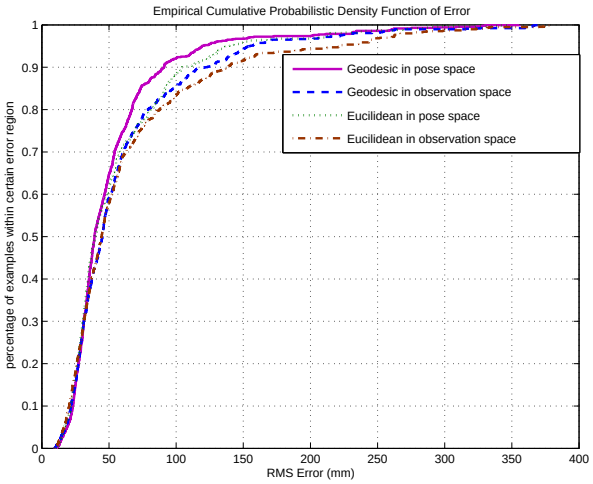

Figure 3. The empirical cumulative probabilistic density function of error (CDF) depicting the percentage of examples below a certain prediction error.

model, it is clear that the pose estimation is rather good and that the larger errors are associated with the extrema of the human body, that is they concern the location of the wrists and ankles.

6. Conclusions and Future Work

In this paper we have presented a novel approach to exemplar-based 3D human pose estimation from a monocular image. A tree is learned by hierarchical clustering on pose manifold via affinity propagation and a multi-output relevance vector machine regressor is developed for 3D pose estimation. The geodesic affinity propagation preserves the topological structure and find clusters on the pose manifold without the difficulty of the appropriate selection of number of clusters that the K-means algorithm has. Also the proposed local multi-output RVMs utilize the same set of relevance vectors across different dimensions of the output and result in reduced computational complexity in comparison to using multiple scalar regressors. We present results in the HumanEva dataset and show a significant improvement in comparison to the baseline method.

In future work, we will extend our method to different activities and a larger dataset, and take temporal constraints into consideration. We also intend to investigate other feature extraction and feature selection methods and deal with background clutter.

References

[1] A. Agarwal and B. Triggs. Recovering 3d human pose from monocular images. *IEEE Trans. on PAMI*, 28(1):44–58, 2006. 2

[2] Christopher. *Pattern Recognition and Machine Learning*. Springer, 2006. 4

[3] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *JMLR*, 2:265–292, December 2001. 4

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, volume 1, pages 886–893, 2005. 5

[5] C. Ek, P. Torr, and N. Lawrence. Gaussian process latent variable models for human pose estimation. In *Machine Learning for Multimodal Interaction*, Lecture Notes in Computer Science, pages 132–143. Springer, 2008. 1, 2

[6] A. Elgammal and C.-S. Lee. Inferring 3d body pose from silhouettes using activity manifold learning. In *CVPR*, 2004. 1, 2

[7] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315(5814):972–976, February 2007. 2, 3

[8] D. M. Gavrila. A bayesian, exemplar-based approach to hierarchical shape matching. *IEEE Trans. on PAMI*, 29(8):1408–1421, 2007. 2, 3

[9] F. Guo and G. Qian. Monocular 3d tracking of articulated human motion in silhouette and pose manifolds. *J. Image Video Process.*, 2008(3):1–18, 2008. 2

[10] A. Kanaujia, C. Sminchisescu, and D. Metaxas. Semi-supervised hierarchical models for 3d human pose reconstruction. In *CVPR*, pages 1–8, 2007. 5

[11] M. J. B. L. Sigal. Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical Report CS-06-08, Brown University, 2006. 5

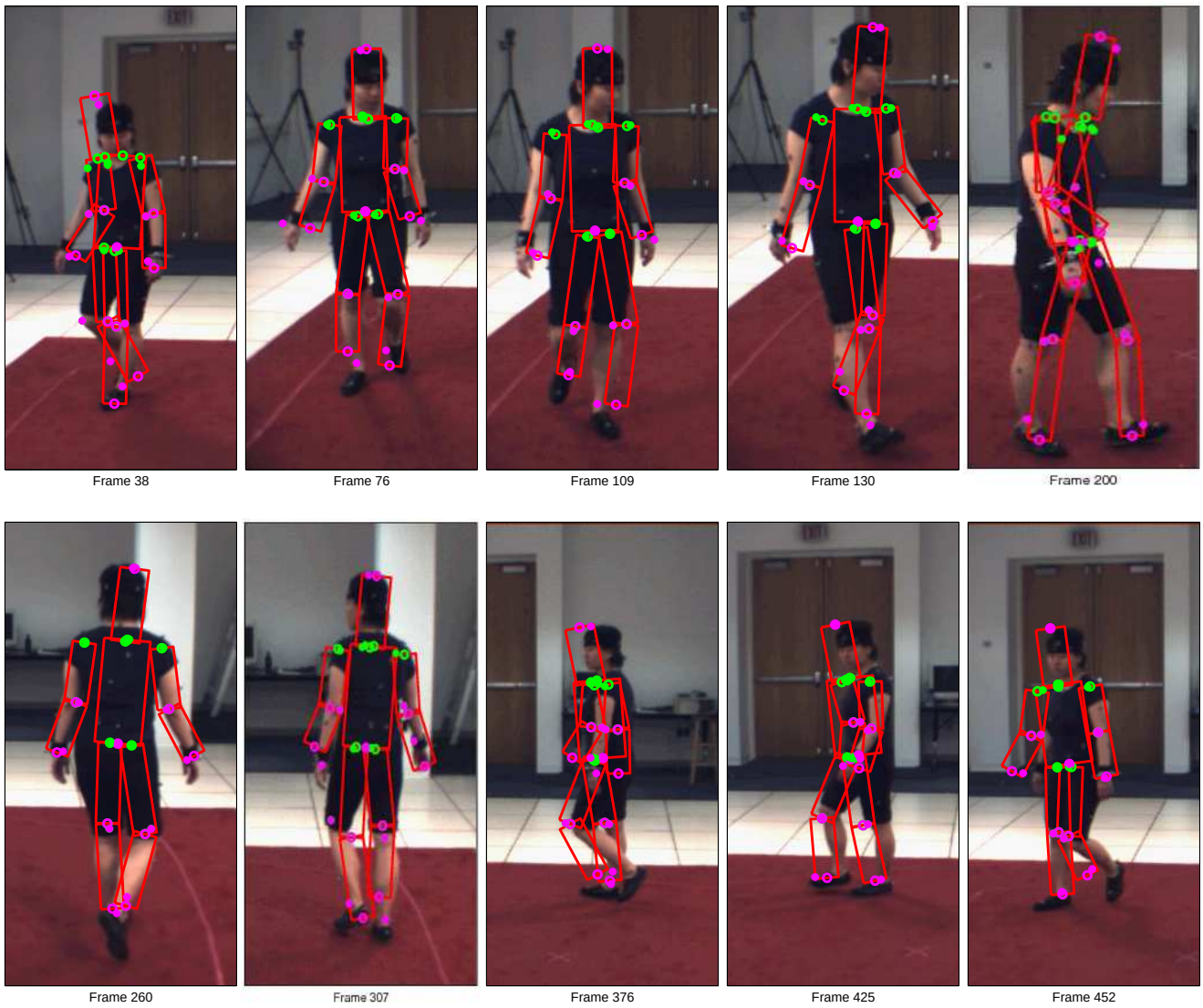

Figure 4. Pose estimation results on HumanEva–I dataset. The larger errors occur at the positions of the four end joints, that is the wrists and the ankles.

[12] R. Okada and S. Soatto. Relevant feature selection for human pose estimation and localization in cluttered images. In *ECCV*, pages 434–445, 2008. 2

[13] R. Poppe. Evaluating example-based pose estimation: Experiments on the humaneva sets. In *CVPR 2nd Workshop on Evaluation of Articulated Human Motion and Pose Estimation (EHuM2)*, 2007. 5

[14] G. Rogez, J. Rihan, S. Ramalingam, C. Orrite, and P. H. S. Torr. Randomized trees for human pose detection. In *CVPR*, pages 1–8, 2008. 2, 3

[15] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. In *ICCV*, pages 750–757 vol.2, 2003. 2

[16] C. Sminchisescu, A. Kanaujia, and D. N. Metaxas. Bm3e : Discriminative density propagation for visual tracking. *IEEE Trans. on PAMI*, 29(11):2030–2044, 2007. 1, 2

[17] A. Thayananthan. *Template-based Pose Estimation and Tracking of 3D Hand Motion*. PhD thesis, Department of Engineering,University of Cambridge, 2005. 4

[18] M. E. Tipping. Sparse bayesian learning and the relevance vector machine. *JMLR*, (1):211–244, 2001. 4

[19] R. Urtasun and T. Darrell. Sparse probabilistic regression for activity-independent human pose inference. In *CVPR*, pages 1–8, 2008. 2