BU CS TR-1999-017. Appears in **Proc. IEEE Computer Vision and Pattern Recognition**, June 2000

Inferring Body Pose without Tracking Body Parts

Rómer Rosales and Stan Sclaroff
Image and Video Computing Group - Computer Science Dept.
Boston University - Boston, MA 02215

Abstract

A novel approach for estimating articulated body posture and motion from monocular video sequences is proposed. Human pose is defined as the instantaneous two dimensional configuration (i.e.,the projection onto the image plane) of a single articulated body in terms of the position of a predetermined set of joints. First, statistical segmentation of the human bodies from the background is performed and low-level visual features are found given the segmented body shape. The goal is to be able to map these, generally low level, visual features to body configurations. The system estimates different mappings, each one with a specific cluster in the visual feature space. Given a set of body motion sequences for training, unsupervised clustering is obtained via the Expectation Maximization algorithm. For each of the clusters, a function is estimated to build the mapping between low-level features to 2D pose. Given new visual features, a mapping from each cluster is performed to yield a set of possible poses. From this set, the system selects the most likely pose given the learned probability distribution and the visual feature similarity between hypothesis and input. Performance of the proposed approach is characterized using real and artificially generated body postures, showing promising results.

1 Introduction

In recent years, there has been a great deal of interest in methods for tracking and analysis of human body motion by computer [1, 5, 21, 18, 22, 12, 15, 8, 19, 14, 20, 6]. Effective solutions would lead to breakthroughs in areas such as video coding, visual surveillance, human motion recognition, ergonomics, video indexing and retrieval, and human-computer interfaces, among others.

If the basic structure of the tracked body (its configuration) is reconstructed, motion analysis would be greatly simplified. In our everyday life, humans can easily estimate body part location and structure from relatively low-resolution images of the projected 3D world (*e.g.*,watching a video). Unfortunately, this problem is inherently difficult for a computer. The difficulty stems from the number of degrees of freedom in the human body, the complex underlying probability distribution, ambiguities in the projection of human motion onto the image plane, self-occlusion, insufficient temporal or spatial resolution, etc.

In this paper, we develop an approach for estimating human body pose given a single image or a monocular image sequence containing unoccluded bodies. Human pose is defined as the instantaneous two dimensional configuration (*i.e.*,the projection onto the image plane) of a single articulated body in terms of the position of a predetermined set of joints. Given a set of body motion sequences for training, a set of clusters is built in which each has statistically similar configurations according to a given measure and model. Then, for each of the clusters, a function that maps visual features to body pose is acquired via machine learning. Given new visual features, a mapping from each cluster is performed providing a set of possible poses. From this set, we extract the most likely pose given the learned probability distribution and the visual feature similarity between hypothesis and input.

In experiments, the resulting method efficiently learns from data sets of body configurations, and using this prior knowledge, how to map low-level visual features to a higher level representation like a set of joint positions of the body. This is a very important step considering that low-level visual features are relatively easily obtained using current vision techniques.

2 General Problem Definition

The problem of obtaining body pose (either 2D or 3D) from visual features can be thought of as an instance of the more general problem of estimating the function that maps elements of a given (cue) space to another (target) space from data. In our case this function seems to be highly complex, and the mapping is many to many (*e.g.*,same visual features can represent different body pose configurations and same body configurations can generate different visual features due to clothing, view-point, etc.).

Let us define $\Psi \subset \Re^t$ to be the set of sample data points from the target space and $\Upsilon \subset \Re^c$, with the same cardinality as Ψ, to be the set of sample data points from the cue space. Assume that for each element $\psi_i \in \Psi$ we know its counterpart $\upsilon_i \in \Upsilon$ (*i.e.*,the data is labeled), or that there is a way to generate υ_i, for example $\upsilon_i = \zeta(\psi_i)$. Note that if ζ is many-to-one, its inverse does not exist.

The problem is to approximate a function that we will call $\xi : \Re^c \rightarrow \Re^t$ (not necessarily the inverse of ζ) that when given $\mathbf{x} \in \Re^c$, such that $\zeta(\mathbf{y}) = \mathbf{x}$, with $\mathbf{x}$ possibly not in Υ, $\xi(\mathbf{x})$ estimates $\hat{\mathbf{y}} \in \Re^t$ such that $\hat{\mathbf{y}}$ is close to $\mathbf{y}$ or $\mathbf{x}$ is close to $\zeta(\hat{\mathbf{y}})$, according to some distance measures.

This problem can be approached by minimizing:

$$\xi^* = \arg\min_{\xi} \sum_{i=1}^{l} \rho(\xi(\upsilon_i) - \psi_i), \qquad (1)$$

where l is the cardinality of Ψ or Υ [2, 9, 16], and ρ is an error function, i.e., a Euclidean norm or robust error norm.

The problem of function approximation from sparse data, sometimes regarded as the general machine learning problem, is known to be ill-posed [9, 2] if no further constraints are added, for example on the function space of ξ (*e.g.*,if ξ is constrained to be linear, a unique best solution exists).

In our case, Ψ represents the set of example human body poses, and Υ is the corresponding set of visual features taken from image projections under certain viewing conditions. We do not intend to solve the general problem of function approximation; instead, we address the specific problem of recovering pose parameters of an articulated body (the human body) from monocular visual features.

2.1 Body Pose from Visual Features

As stated above, our goal is to map visual features to likely body pose configurations. For training, motion capture can provide 3D marker positions and orientation of the human. Following a similar notation to that used above, the set of marker positions is denoted $\Psi_{3d} \subset \Re^t$.

Visual features generated by the three-dimensional object can be obtained by pointing a video camera at the given object, and analyzing the captured images. It is clear that these visual features depend on the camera parameters (*e.g.*,camera orientation, location, focal length, etc).

Alternatively, a computer graphics model of the 3D object (in our case, a human body model) can be used to render a set of images. These images simulate the visual appearance of the object in question, given pose and camera parameters. We call the rendering function $R : \Re^t \rightarrow \mathcal{I}$, where $\mathcal{I}$ is set of images at a given resolution. Optionally R can take a parameter θ indicating the camera point of view (or object orientation).

Images are an intermediate representation from which we can extract visual features using a function we denote by $V : \mathcal{I} \rightarrow \Re^c$. Following the definitions above, we have:

$$\zeta(\psi) = V(R(\psi)), \zeta : \Re^t \rightarrow \Re^c. \tag{2}$$

The set $\Upsilon_{3d} \subset \Re^c$ is formed by the visual features extracted from the images of Ψ_{3d}, using ζ. Our goal is to estimate the function denoted ξ, as defined above.

An alternative problem is to recover 2D marker positions, instead of 3D positions, from image features. By 2D marker positions, we mean the projection of the 3D markers onto the image plane. The 2D projections of the markers can be obtained from Ψ_{3d} to generate a data set $\Psi_{2d,\theta} \subset \Re^s$ of all frames viewed from camera orientation θ, and a distance to the object.

In the same way as in the 3D case, we can render 2D marker positions to form an image, this rendering function will be denoted $\hat{R} : \Re^s \rightarrow \mathcal{I}$, which is a 2D approximation of R. Note that having the set Ψ_{3d} from which $\Psi_{2d,\theta}$ was generated, we can obtain a more accurate rendering by using R on Ψ_{3d} at the appropriate orientation θ. When this is possible, we will use R instead of $\hat{R}$. To generate visual features from images, we can proceed as before, using V to generate the set $\Upsilon_{2d,\theta} \subset \Re^c$, which contains the visual features corresponding to the rendering of the set $\Psi_{2d,\theta}$. For notational convenience, we define $\Psi_{2d} = \bigcup_\theta \Psi_{2d,\theta}$, Υ_{2d} can be defined similarly. We also have:

$$\zeta_{2d}(\psi) = V(\hat{R}(\psi)), \zeta_{2d} : \Re^s \rightarrow \Re^c, \tag{3}$$

with $\psi \in \Psi_{2d}$. The problem is then to approximate ξ_{2d} (the 2D version of ξ)from data. In other words, given visual features, we want to find the likely 2D marker projections that generated them.

3 Related Work

One of the fundamental ideas in perception of human motion is the was of Johansson's moving light displays [13], where it was demonstrated that relatively little information (motion of a set of selected points on the body) is needed for humans to perform reconstruction of the body configurations. One of the first approaches related with walking people in real environments is due to [10]. The basic detection and registration technique used commonly is based on background segmentation, related to the work of Baumberg and Hogg [1], Bichsel [3], and others [22].

In order to find body parts using visual cues, [22] employed blob statistics and contour descriptions to roughly indicate where hands, feet, and torso were located. The system needed to be initialized with a certain configuration in which body part identification was easy to achieve. After this, their part labeling relied mostly on tracking blobs. In [12], some simple heuristics about body part relations were used to assist labeling and tracking.

In an alternative approach, model-based representations like [15, 8, 19, 14, 20, 7, 6], have been used. The models are generally articulated bodies comprised of 2D or 3D solid primitives. Most of these techniques require the use of multiple cameras, controlled viewing conditions, and/or user initialization. Also, model-based methods generally cannot recover from tracking errors in the middle of a sequence because they rely strongly on the accurateness of the estimate at the previous frame. Tracking errors may be common in real scenes due to low contrast, occlusions, changes in brightness, etc.

The main difference in our proposed approach with respect to the techniques mentioned above is that we do not try to match a body model to an image. We do not match image features from frame to frame; *e.g.*,image regions, points, articulated models. Therefore, we do not refer to our approach as tracking, *per se*. Instead, machine learning is used to map visual features to likely body configurations. Due to this, the pose estimation may not be as exact as the best performance of model-based tracking techniques; however, our approach tends to be more robust.

Previous learning based approaches include [18], where a statistical approach was taken for reconstructing the three-dimensional motions of a human figure, but assuming that 2D tracking of joints is given. They used a set of motion capture examples to build a Gaussian probability model for short human motion sequences.

The work most closely-related to our approach is [5], where the manifold of human body configurations was modeled via a hidden Markov model and learned via entropy minimization [5]. Because it models the dynamics of the time series (*i.e.*,human motion), the amount of data needed to create a good model (*i.e.*,approximate the manifold) is markedly larger than that needed when modeling single configurations.

Unlike previous learning based methods, our method does not attempt to model the dynamical system; instead, it relies only on instantaneous configurations. Even though this ignores information (*i.e.*,motion components) that can be useful for constraining the reconstruction process, it provides invariance with respect to speed (*i.e.*,sampling differences) and direction in which motions are performed. Furthermore, fewer training sequences are needed in learning a model. Finally, it should be noted that we employ a new step of feedback matching, which transforms the reconstructed configuration back to the visual cue space to choose among the set of reconstruction hypotheses.

4 Approach Overview

For clarity, we very briefly enumerate every step of the proposed approach. The steps are as follows:

1. A set of motion 3D capture sequences is obtained, $\Psi_{3d} \subset \Re^t$. A set of visual features Υ_{3d} is computed from images that the 3D body generated (using a computer graphics rendering function or simply captured by a video camera). By projecting the elements of Ψ_{3d} onto the image plane over a given number of views, we obtain as set of 2D marker positions Ψ_{2d}.

2. The set Ψ_{2d} is partioned into several exclusive subsets via unsupervised clustering. This yields a set Ω of m clusters. Each cluster corresponds to a group of similar pose parameters.

3. Given Ψ_{2d} and Υ_{2d}, for each cluster i, we approximate a mapping function P_i. By clustering our target space, the mapping can be approximated with simple functions, each responsible for a subset of the domain. We would hope that linear functions could do the mapping, but decided to estimate nonlinear functions; a multi-layer perceptron is trained for each cluster.

4. Novel data is presented in the form of human silhouettes. For each frame, visual features are extracted using $V : \mathcal{I} \rightarrow \Re^c$. Then, using P_i, a set of m projected marker positions per frame are estimated.

5. The series of possible m solutions provided for each frame is rendered to achieve images and their visual features are extracted. The best match with respect to the presented data can then be found via the maximum likelihood criterion. As an optional step, consistency in time can be enforced by observing some frames ahead.

5 Modeling the Configuration Space

Motion capture data Ψ_{3d} will be used to train our model. Motion capture data provides 3D position information about the location of a set of markers. In the case, the set of markers roughly corresponds to a subset of major human body joints. This set of marker is fixed and determined beforehand.

3D marker positions are projected into 2D marker positions Ψ_{2d}, using a perspective camera located at a fixed height and distance from the center of the body. This projection is repeated by rotating the camera around the main axis of the human, at fixed increments of θ. In our experiments $d\theta = \pi/16$. Note that we can make the set Ψ_{2d} as dense as we want by sampling at more camera orientations. To account for a wider variety of viewing conditions, we could sample the whole viewing sphere. Differences in the camera-object distance could be avoided in principle by choosing scale invariant image features.

Given marker positions for a human body in a particular frame, we can render its visual appearance using computer graphics techniques. In our case, we specify the structure of the connections between markers, and use cylinders to connect them. Fig. 1 shows two elements of the set Ψ_{2d}, and the corresponding rendered binary images from which visual features Υ_{2d} are extracted. For this implementation we chose Hu moments [11] as our visual features, mainly due to their ease of computation and their invariance to translation, scaling and rotation on the image plane.

Formally, we have the following data sets:

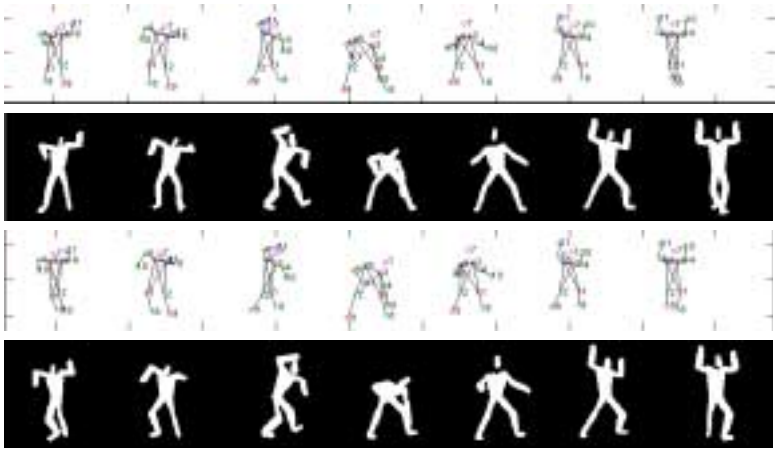
1. Ψ_{3d} is the set of 3D body configurations expressed in marker positions, obtained using motion capture.

2. $\Psi_{2d} = \{\psi_{2d} \in \Re^s | C(\psi_{3d}, \theta) = \psi_{2d}\}$, with θ varying from 0 to 2π using a fixed increment, and C the transformation that projects spatial coordinates to the image plane, using viewpoint θ.

3. $\Upsilon_{2d} = \{\upsilon_{2d} \in \Re^c | \zeta(\psi_{3d}, \theta) = \upsilon_{2d}\}$, with $\zeta = V \circ R$, mapping 3D marker positions to image features.

5.1 Clustering body configurations

It would be ideal if the mapping from Υ_{2d} to Ψ_{2d} were simple. Unfortunately this mapping is highly ambiguous. For example, if moments on binary images (*e.g.*,body silhouettes) are used as visual features, a person facing forward would generate very similar image moments to another facing backwards. Image moments provide a descriptor that does not encode many of the degrees of freedom of 2D



(a) $\psi_{2d} = C(\psi_{3d}, 0) \in \Psi_{2d,0}$

(a) $R(\Psi_{3d}, 0)) \in \mathcal{I}$

(b) $\psi_{2d} = C(\psi_{3d}, 3\pi/16) \in \Psi_{2d,3\pi/16}$

(b) $R(\Psi_{3d}, 3\pi/16)) \in \mathcal{I}$

Figure 1: The data used for training is formed by 2D, projected marker positions Ψ_{2d} and the visual features extracted from the binary masks obtained via rendering of a 3D computer graphics model Υ_{2d}. This rendering is repeated by rotating the virtual camera around the main axis of the human, at fixed increments of θ. The binary masks are used to obtain visual features. Here we show some frames from the same sequence viewed from two different camera locations: (a) $\theta = 0$ rads, (b) $\theta = 6\pi/32$ rads.

Figure 2: The cluster means obtained after performing unsupervised clustering of the data points in 2D marker space. Note that opposite configurations are clustered separately. For example, one can see that there is a cluster for the figure facing forward, and another one backward. This separation is important because visual features alone cannot resolve this ambiguity. Complexity of the mapping is reduced if clusters are trained separately.

markers. Therefore, it is possible that drastically different body configurations have similar image moments.

The way we approach the problems mentioned above is by first creating clusters of statistically homogeneous data points in the marker space. We used 2D projected markers to try to generate clusters that can be described by a Gaussian probability distribution. This is an unsupervised clustering task, for which we use the EM algorithm.

Let us denote $\Theta_i = (\mu_i, \Sigma_i)$ to be the learned distribution parameters for cluster i. For each data point $\psi \in \Psi$, we can assign it to a cluster, by just finding the ML (Maximun Likelihood) estimate.

$$i = \arg\max_j(P(\Theta_j|\psi)) = \arg\max_j(\mathcal{N}(\psi, \mu_j, \Sigma_j)), \quad (4)$$

where i is the label of the cluster to which we assigned this data point ψ.

Fig. 2 shows the mean configuration of a set of 15 clusters found by this method using the set Ψ_{2d}. By splitting the body configuration space into homogeneous regions, it becomes feasible to approximate a more specialized (and simpler) map from a visual feature space. This will reduce the ambiguities mentioned above. For example, in Fig. 2 we can see that there are mean configurations facing forward and backward.

5.2 Training the Map from Visual Features to Body Configurations

Once data points are divided into clusters, the system must learn cluster-dependent mappings that take visual features to body configurations. For each cluster, we used a neural network to train how to map inputs (from Υ_{2d}) to outputs (from Ψ_{2d}) in a supervised fashion. A multi-layer perceptron with one hidden layer is chosen to do this [4]. The explicit expression for this function is:

$$\hat{y}_k = g_2(\sum_{j=0}^{l_2} w_{kj}^{(2)} g_1(\sum_{i=0}^{l_1} w_{ji}^{(1)} x_i)), \quad (5)$$

where $\hat{\mathbf{x}} \in \Psi$ is the visual feature vector at a given instant, $\mathbf{y}$ is the estimated marker configuration, $\mathbf{w}^{(1)}$ and $\mathbf{w}^{(2)}$ are each layer's synaptic weights and biases, g_1 and g_2 are a sigmoidal and linear function respectively.

This architecture was chosen because it can approximate some non-linear mappings [17] instead of just linear ones, and the training is relatively simple, given the data. The parameters of this network were estimated via Levenberg-Marquardt optimization to update the weights and biases. The system creates a set $\Omega = \{P_1, P_2, ..., P_m\}$ of m multi-layer perceptrons, each trained to a particular cluster of body configurations.

6 Synthesizing Body Configurations

When novel data $\mathbf{x} \in \Re^c$ is presented (*i.e.*,features computed from an image frame), the likely 2D marker positions are estimated using the cluster-dependent functions P_i. This yields a set of hypothetical body configurations $\hat{\mathbf{Y}} = \{\hat{\mathbf{y}}_k\}$. There is a total of m hypotheses per frame $\mathbf{x}$. The question is, how to choose from this set of hypotheses? We approach this problem by creating another mapping or function P_b that estimates visual features from 2D marker positions. There are different alternatives for doing this.

One alternative is to simply use each $\hat{\mathbf{y}}_k$ to render an image using $P_b = \hat{R}$ (*i.e.*,using computer graphics), and then find its visual features via V. Recall that $\hat{R}$ is a 2D approximation of the 3D rendering function R. It is an approximation because the object to be rendered is three-dimensional, and here the hypotheses $\hat{\mathbf{Y}}$ are 2D markers.

Another alternative is to approximate the function P_b from data (*i.e.*,Υ_{2d} and Ψ_{2d}). That is, we could estimate the parameters of P_b using the approach outlined in Sec. 5.2. Then given $\hat{\mathbf{y}}_k$, we can obtain an estimate $\hat{\mathbf{x}}_k = P_b(\mathbf{y}_k)$. This function P_b would avoid the need of a probably expensive rendering of the m hypotheses.

In our system, we chose the second alternative. This was done using the sets Υ_{2d} and Ψ_{2d}. Because this mapping uses data rendered with knowledge of 3D information, it is very likely to have accuracy advantages over the simpler transformation $\hat{R}$.

Given the set $\hat{\mathbf{Y}}$ of hypotheses about the body configuration, we find the most accurate hypothesis by minimizing:

$$i = \arg\min_j (P_b(\hat{\mathbf{y}}_j) - \mathbf{x}_j)^\top \Sigma_\Upsilon^{-1} (P_b(\hat{\mathbf{y}}_j) - \mathbf{x}_j), \quad (6)$$

where Σ_Υ is the covariance matrix of the elements in the set Υ_{2d} and i is the neural network label that best matched the visual feature observed.

As a further refinement step, because neighboring frames are generally from similar configurations, we have obtained slightly better performance if consistency in time is enforced. Therefore, after we obtain the best P_k to use for a given frame, if this network differs from that chosen in the previous frame, we wait for the next frames to arrive (generally 2 or 3) to decide whether to use this new P_k. If within this window the new frames are consistent with the change (use the same P_k), then the new P_k is used; if not, then the previous network is used instead. Although a probabilistic dynamical model would be advantageous, this proved to be an effective culling mechanism to avoid spurious individual reconstructed frames.

7 Experiments

In order to evaluate the performance of our approach, we first conducted experiments in which we had knowledge of the *best* reconstruction. Using the data sets Ψ_{2d} and Υ_{2d}, we performed clustering and training, taking out the sequence with the specific orientation that would be used for testing. We also took out its neighboring views, the opposite view and its neighbors. View orientations were sampled every $2\pi/32$ radians, for a total of 32 orientations.

The training data set consisted of five sequences with an average of about 200 frames each, sampled at the orientations above mentioned and at 30 frames/sec. The activities include dance, walking/grabbing/throwing, walking/crouching-down/turning/walking, and walking in circle. The 3D motion-capture data was obtained from http://www.biovision.com. This data consisted of position information of 37 markers, from which we chose a subset of 11 considered by us the most informative ones.

Fig. 3 shows the reconstruction obtained by our approach when images of the *destroy* sequence were shown. This sequence exhibits very challenging configurations and orientations. The view angles used were 0 and $12\pi/32$ radians respectively. The agreement between reconstruction and ground-truth is easy to perceive for all sequences. Note that for self-occluding configurations, reconstruction is harder, but still the estimate is close. The error is mainly due to the inadequacy of the feature and image representation to separate configurations that are different in the marker space, but similar in feature space.

Using the training and testing procedure described above, we measured the average marker error (measured as the distance between reconstructed and ground-truth projected marker position). After testing all the sequences, the mean and variance marker displacement was 0.0428 and 0.0011 units respectively. As a point or reference, the height of the figure was approximately 1.4 units on average. Therefore, the mean marker displacement was approximately 3% and its variance 0.07% of the body height.

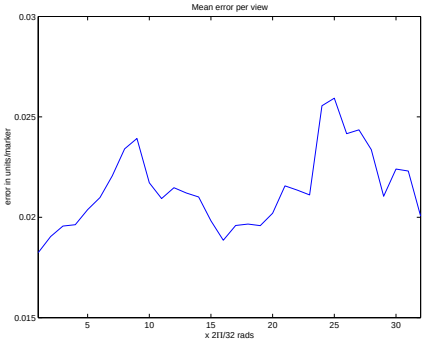


Figure 4: Measure of the mean marker error per view angle. Figures were aligned to always face forward for the 0 radians view angle. This was done so that we could measure the error with respect to body orientation, not camera orientation. Recall that bodies can turn in a sequence. Angles are sampled every $2\pi/32$ radians starting at $2\pi/32$ radians. Note that the error is bigger for orientations close to $\pi/2$ and $3\pi/2$ radians.

We also measured the average marker error per body orientation. For this we have to rotate the 3D figures so that their orientation corresponds to the orientation tested. Recall that in the original sequences, bodies are not always

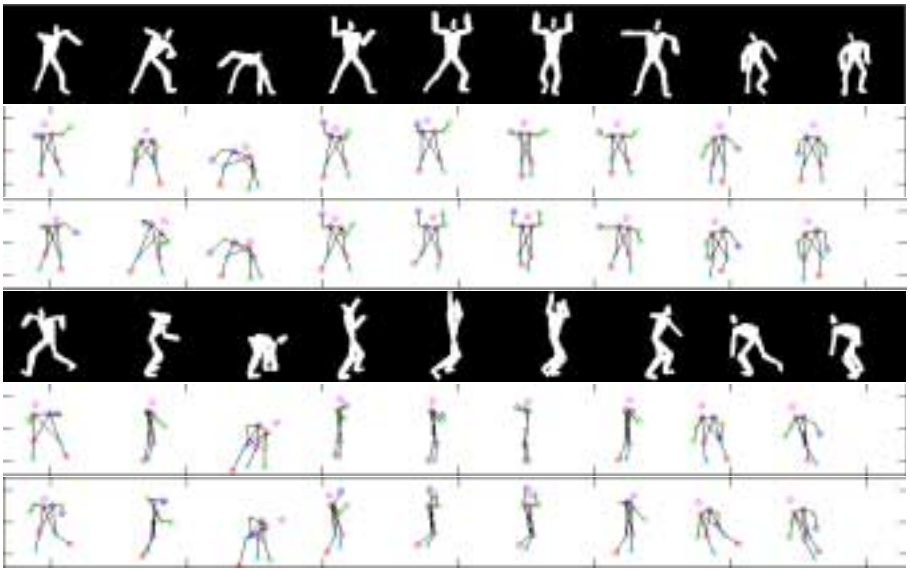


Figure 3: Example reconstruction of the *destroy* sequence, each set (3 rows each) consists of input images, reconstruction, and ground-truth. Results are shown every 25th frame. View angles are 0 and $12\pi/32$ radians. The obtained reconstruction visually agrees with the perfect output for all views. Note that this sequence has challenging configurations, body orientation is also recovered correctly.

facing a fixed point. Angles are sampled every $\pi/16$ radians starting at 0 radians, which corresponds to the person always facing to the camera. Note that the error is bigger for orientations closer to $\pi/2$ and $3\pi/2$ radians. This intuitively agrees with the notion that at those angles (side-view), there is less visibility of the body parts. This performance is very promising considering the complexity of the task and the simplicity of the approach.

7.1 Experiments using Real Visual Cues

For our next example, in Fig. 5 we now test the system against real segmented visual data, obtained from observing and tracking and human subject. Reconstruction for several relatively complex action sequences are shown below each sequence. Note that even though the characteristics of the segmented body differ from the ones used for training, good performance is achieved. Most frames are visually close to what can be thought as the right pose reconstruction. Body orientation also is correct not just for frontal views.

However, it is possible to see some erroneous configurations. We believe the source of error is mainly due to several reasons: 1.) insufficient data to account for given configurations that cannot just be obtained by interpolating surrounding ones (*e.g.*,raising arms up / pointing both arms to same direction), 2.) possible need of more clusters or approximating functions with more specialized domains (in cue space), 3.) differences in body characteristics used for training/testing, and 4.) little discriminative power of the chosen image features (Hu moments, which reduce the image interpretation to a 10-dimensional vector). Despite these errors, the experimental results are encouraging when compared with previous results.

8 Conclusion

We have presented a novel technique that allows the reconstruction of human body pose from low-level visual features. Because of the complexity of the mapping, we clustered the space of body configurations into approximately homogeneous configurations, showing improved results. The proposed approach is both simple and powerful. Our ideas are different from tracking approaches in that we do not try to match body parts from frame to frame.

Human pose reconstruction is a particularly hard problem because this mapping is highly ambiguous. We have obtained excellent results even using a very simple set of image features, such as image moments. Choosing the best subset of image features from a given set is by itself a complex problem, and a topic of on-going research.

The implemented algorithm for reconstruction runs in linear time $O(M)$ with respect to the number of clusters M. Also it scales linearly for sequences, for a sequence of length N, the complexity is $O(NM)$. The method is by itself causal, but performance improved slightly when looking two or three frames ahead.

The current implementation was tested in recovering the pose for both generated and real visual data. The artificially generated data was used for measuring the performance of the approach, real data showed its applicability. The results are encouraging in considering the complexity of the task and in comparison with results reported for previous methods.

Figure 5: Reconstruction for three different real action sequences obtained from tracking a human subject(every 30th frame shown).

Acknowledgments

We thank Matt Brand for inspiring machine learning discussions. This work was supported in part through ONR Young Investigator Award N00014-96-1-0661, and NSF grants IIS-9624168 and EIA-9623865.

References

[1]A. Baumberg and D. Hogg. Learning flexible models from image sequences. In *ECCV*, 1994.

[2]M. Bertero, T. Poggio, and V. Torre. Ill-posed problems in early vision. *Proc. of the IEEE*, (76) 869-889, 1988.

[3]M. Bichsel. Segmenting simply connected moving objects in a static scene. *PAMI*, 16 (11):1138-1142, 1994.

[4]C. Bishop. *Neural networks for pattern recognition*. Oxford Univ. Press, 1995.

[5]M. Brand. Shadow puppetry. In *ICCV*, 1999.

[6]C. Bregler. Tracking people with twists and exponential maps. In *CVPR98*, 1998.

[7]T.J. Cham and J. Rehg. A multiple hypothesis approach to figure tracking. In *CVPR*, 1999.

[8]D. Gavrila and L. Davis. Tracking of humans in action: a 3-d model-based approac. In *Proc. ARPA Image Understanding Workshop, Palm Springs*, 1996.

[9]F. Girosi, M. Jones, and T. Poggio. Regularization theory and neural network architectures. *Neural Computation*, (7) 219-269, 1995.

[10]D. Hogg. *Interpreting Images of a Known Moving Object*. PhD thesis, University of Sussex, 1984.

[11]M. K. Hu. Visual pattern recognition by moment invariants. *IRE Trans. Inform. Theory*, IT(8), 1962.

[12]L. Davis I. Haritaouglu, D. Harwood. Ghost: A human body part labeling system using silhouettes. In *Intl. Conf. Pattern Recognition*, 1998.

[13]G. Johansson. Visual perception of biological motion and a model for its analysis. *Perception and Psychophysics*, 14(2): 210-211, 1973.

[14]S. Ju, M. Black, and Y. Yacoob. Cardboard people: A parameterized model of articulated image motion. In *Proc. Gesture Recognition*, 1996.

[15]I. Kakadiaris, D. Metaxas, and R. Bajcsy. Active part-decomposition, shape and motion estimation of articulated objects: A physics-based approach. In *CVPR*, 1994.

[16]N. Kolmogorov and S.V. Fomine. *Elements of the Theory of Functions and Functional Analysis*. Dover, 1975.

[17]A. Lapedes and R Farber. How neural nets work. *Neural Information Processing Systems*, 442-456, 1988.

[18]M. Leventon and W. Freeman. Bayesian estimation of 3-d human motion. Technical Report TR 98-06, Mitsubishi Electric Research Labs, 1998.

[19]A. Pentland and B. Horowitz. Recovery of non-rigid motion and structure. *PAMI*, 13(7):730–742, 1991.

[20]J. M. Regh and T. Kanade. Model-based tracking of self-occluding articulated objects. In *ICCV*, 1995.

[21]R. Rosales and S. Sclaroff. 3d trajectory recovery for tracking multiple objects and trajectory guided recognition of actions. In *CVPR*, 1999.

[22]C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: Real time tracking of the human body. *PAMI*, 19(7):780-785, 1997.