Part Segmentation of Visual Hull for 3D Human Pose Estimation

Atul Kanaujia
ObjectVideo,Inc.
akanaujia@objectvideo.com

Nicholas Kittens
ObjectVideo, Inc.
nkittens@gmail.com

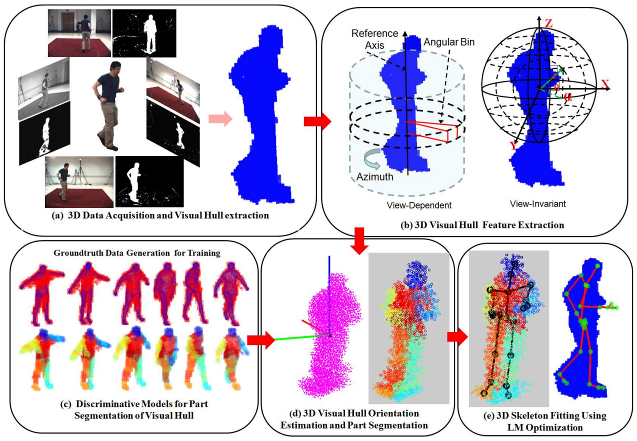
Narayanan Ramanathan
ObjectVideo,Inc.
nramanathan@objectvideo.com

Abstract

In this paper we present an algorithm for estimating 3D pose of human targets using multiple, synchronized video streams obtained from a set of calibrated visual sensors. Our method uses 3D visual hull, reconstructed from multi-view image silhouettes, to estimate skeleton and 3D pose of the human target. The key contribution of this work is to extend predictive human pose estimation algorithms used in the kinect gaming system to 3D visual hull data. In 3D space, viewpoint invariance is achieved by transforming the world reference frame to human centered reference frame. To do so, we first estimate the rigid body orientation and translation of the target from the shape of the visual hull. We then apply discriminative classifiers in the human centered reference frame to segment the 3D voxels of the visual hull into semantic part segments. The part clusters are then used to estimate a 3D pose that best aligns with the detected joint centers while conforming to the part non self-intersection constraints. Claims made in the work are supported by extensive experimental evaluation on both synthetic and real dataset.

1. Introduction

Depth sensors are being widely applied for acquiring range map of nearby human targets for 3D motion recovery and activity analysis. However, their limited ranging capability and use of infrared laser has restricted its application to outdoor environments and in surveillance scenarios. Tremendous progress has been made in sensor technology in recent years, enabling development of advanced video sensors with gigapixel resolution, that are capable of providing sufficient image resolution for detailed analysis of human targets. In the near future, these sensors will be capable of conducting wide area surveillance from a long standoff distance, at the same time providing sufficient image resolution for inferring subtle changes in target appearances. In this work, we present a fully automated system for estimating 3D skeleton and pose of the human targets from multi-view imagery. Our algorithm extends the core discriminative model learning method used in the kinect gaming platform[19] to estimate 3D human pose from 3D visual hull data. Specifically, we use intermediate body part representation to first segment a 3D visual hull into meaningful part clusters. We then use trained discriminative classifiers to classify 3D voxel into body parts. Finally, the identified part segments are used to fit the 3D skeleton using standard optimization algorithms.



(a) 3D Data Acquisition and Visual Hull extraction

(b) 3D Visual Hull Feature Extraction

(c) Discriminative Models for Part Segmentation of Visual Hull

(d) 3D Visual Hull Orientation Estimation and Part Segmentation

(e) 3D Skeleton Fitting Using LM Optimization

Figure 1. Overview of the proposed system for 3D human pose estimation from multi-view imagery;(a) 3D Data is acquired as volumetric reconstruction using space carving;(b) 3D shape descriptors are extracted to encode voxel distribution in 3D space ; (c) Predictive models are trained offline from the aligned visual hull data extracted from synthetic the silhouettes. Ground truth, part-segmented 3D mesh is aligned to the visual hull and the voxels are assigned part-labels of the nearest mesh vertex; (d) Predictors trained on the labeled data are used to estimate orientation of the visual hull and part-labels of the 3D voxels ; (e) Joint locations are estimated using mean-shift mode finding. 3D Pose is estimated by fitting skeleton using Levenberg-Marquardt(LM) algorithm

Contributions: We have developed a novel algorithm for estimating 3D pose from visual hull reconstructed from multi-view imagery. The main contribution of our algorithm is the use of intermediate part-based representation in our inference. We extend the shape context descriptor[5] to 3D space and use it to train predictors for segmenting 3D visual hull into semantically meaningful body regions. We develop a principled training scheme using motion capture sequences and 3D human shape model to train the discriminative classifiers and regression models for part segmentation. As our algorithm employs only scale normalized shape features, it is invariant to clothing, appearance and anthropometric variations of the human targets. We further extend the system to fit a skeleton to the joint locations estimated as the body part centers inferred in the classification step. In doing so, we incorporate additional constraints due to non self-intersection of body parts, anthropometric priors on the skeleton shapes and angular limits on the skeletal joints. We empirically show that this step indeed improves the joint location accuracy. Finally, the method allows efficient estimation of target specific skeleton.

Related work: A comprehensive survey of initial work on marker-less motion-capture techniques from single and multi-view imagery is provided in [15]. Compared to earlier approaches[15] that modeled human shapes with cylindrical or superquadrics parts, current methods use more accurate modeling of 3D human shapes using SCAPE body models[3] or CAESAR dataset [2]. Multi-camera human pose estimation systems developed by Balan and Sigal[3, 4, 20]employed SCAPE(Shape Completion and Animation of People) data to model variability in 3D human shapes due to anthropometry, pose and body type. Gall et al.[7],[22] developed a combined skeleton and 3D shape based human models which they fit to multi-view imagery using a combined local and global optimization scheme. Their algorithm is a two-pass fitting approach. In the first pass, a skeleton with approximate skin is fit to the observation (silhouettes) from multiple sensors. They have proposed a combined optimization scheme for estimating 3D pose by first fitting the parts locally to the observed silhouettes and then projecting the 3D pose to a lower dimensional search space for global optimization. The 3D body shape is obtained by non-rigidly deforming the 3D mesh under the influence of the skeleton followed by the optimizing the mesh shape to match the observed silhouettes. An extension of the above work [8] used action based priors to improve pose tracking and 3D shape estimation from the multi-view image data. Moll et al.[17] proposed a multimodal system to improve 3D human pose and shape estimation from multi-view imagery by using both visual cues and global orientation information from inertial sensors. In parallel to these approaches, Munderman et al.[16]developed a SCAPE model with an underlying skeleton to track 3D poses of a human target in multi-view image sequences using an extension of Iterative Closest Point (ICP) algorithm. The method tracks by aligning the body parts of the human model to the visual hull reconstruction using an extension of Iterative Closest Point (ICP) algorithm. The system however requires manual initialization and critically depends on the quality of visual hull extraction. Hofmann et al.[13] developed upper body 3D pose estimation system from synchronized multi-view imagery using hierarchical shape matching and probabilistically voting based method. A number of recent works employing range sensor data for estimating 3D human pose have demonstrated remarkable pose estimation results at real-time processing rates. Notable among them are [10, 19, 11, 9]. Ganapathis et al.[10, 9] developed a fast human 3D pose tracking algorithm that runs faster than real-time using an extension of Iterative Closest Point (ICP) approach that can efficiently resolve part placement ambiguities by enforcing localization constraints in 3D space. Our approach most closely resembles the algorithms proposed in [19, 11] that form the core components of kinect gaming system. The work used single depth-image to accurately estimate 3D pose of the human without the need for temporal cues to resolve ambiguities. Most remarkably, the work also showed that pixel-wise classification (no contextual modeling) is sufficient for inferring 3D pose. Inspired by the success of this approach, we have developed algorithms to employ conventional multi-camera imagery to estimate 3D pose of human targets in a scene. The proposed algorithm has obvious advantages of exhibiting greater invariance to changes in both appearance and anthropometry of the targets as well as camera configurations.

2. System Overview

Fig. 1 shows the key components of the system. We obtain synchronized streams of multi-view imagery from a set of calibrated cameras and use it to generate a 3D volumetric reconstruction (visual hull) of the target using space carving (see fig. 1(a)). We use visual hull to extract a view-dependent and view-invariant 3D shape descriptor (fig. 1(b)) of the visual hull. We have synthetically generated a database of labeled examples of 3D poses to train discriminative classification and regression models using the two descriptors (fig. 1(c)). The training examples are generated by using motion capture data to animate randomly sampled, synthetic 3D mesh shapes (from a PCA based 3D human shape model). The regression model is trained to output 3D orientation of the target using view-dependent descriptor, and classifiers are trained for recognizing part segments of the visual hull. The pose estimation is done by first employing the regression model to remove rigid body motion (rotation, scaling and translation) in order to compute view-invariant shape context features. The part classifiers are then used to recognize 3D voxels of the visual hull (fig. 1(d)). Finally, an anthropometric skeleton is fitted to the joint centers obtained from the predicted part and joint segments (see fig. 1(e)). We go over each of the components in detail in the rest of the paper.

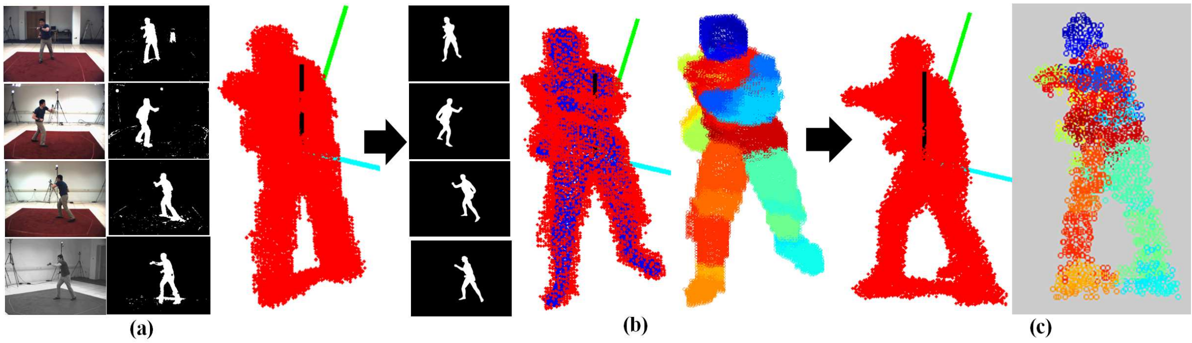


Figure 2. *(a)* Visual hull extracted from real HumanEva dataset with reference axis transformed to remove translation, scaling and rotation (root orientation) of the visual hull ; (b) Synthetic data images rendered using Autodesk Maya using the same camera configuration and 3D pose as the real data. Middle figure shows the aligned 3D mesh to the visual hull used for parts labeling of 3D voxels ; (c) Visual hull orientation and voxel classification results for real data using the learned regression and classification models

3. Feature Extraction

3D Data Acquisition: Silhouettes extracted from the image streams acquired from multiple calibrated sensors are used to reconstruct a 3D volumetric representation(visual hull) of the human target. We use an efficient background subtraction based on non-parametric kernel density estimate of stationary pixels [6] to extract silhouettes of moving targets. An octree-based fast iterative space carving algorithm is used to extract volumetric reconstruction of the target. The algorithm is initialized using single 3D cubic volume that completely encloses the working space of the acquisition system. The algorithm first determines whether a voxel is the boundary voxels and then iteratively subdivides it into eight parts (voxels) until the size of the voxels is less than the threshold size. The boundary voxels are determined based on whether its projection to the camera image plane lies totally inside, outside or on the boundary of the silhouette. This is efficiently done using the footprint test where the projection of the cube is approximated as a square and an integral image of the silhouette is used to identify whether the square is completely or partially inside the silhouette.

To classify a 3D voxel to a part, the 3D shape descriptors are used to encode its spatial relation with respect to other voxels of the visual hull. These shape descriptors vary smoothly in the feature space such that visual observations of a human in similar 3D poses and orientations have similar descriptor values while distant poses and orientations are mapped to distant points in the feature space. In order to train robust discriminative models that can be generalized across subject variations and illumination changes, it is important that these descriptors are distinctive and at the same time invariant to slight changes in the viewing angle and misalignment. Finally, a critical characteristic of the descriptor we seek is invariance to the orientation of the human. This is achieved by extracting descriptors in a human centered coordinate frame.

Shape Descriptor for Estimating Visual Hull Orientation: To estimate the human centered reference frame, we train discriminative models to directly predict the orientation of the human using the view-dependent shape descriptor of the 3D visual hull. We employ the efficient 3D visual hull shape descriptors proposed by Sangawa et. al [18] to train non-linear sparse Bayesian regression model (Relevance Vector Machine) for predicting orientation of the human target(see 1 (b)). The voxels from the visual hull are voted into radial, angular and axial bins of the cylindrical shaped descriptor. This encoding scheme has been demonstrated to be flexible to complex motions, and robust to noise due to shadows and inaccurate silhouette extraction. The reference axis is chosen as perpendicular to the ground plane and passing through the centroid of the visual hull. The height and radius of the cylinder is determined from the visual hull.

Shape Descriptors for Visual Hull Segmentation: We employ view invariant 3D Shape Context Histogram (SCH) to compactly encode spatial distribution information of the visual hull voxels. Shape context is extracted by uniformly sampling N voxels from the visual hull and uniformly voting into bins along the radius r, the elevation angles θ and the azimuth angles α (see fig. 1. The descriptors has been known to be robust to noise in the visual hull or erroneous estimates of the human orientation and are rendered view-invariant by transforming the reference axis according to the orientation of human.

4. Visual Hull Segmentation

3D Shape of a human target is a strong cue of its 3D pose. In order to estimate the 3D pose of the target, we adopt an intermediate step of first segmenting the voxels in the visual hull to different body segments and then fitting a 3D skeleton to the localized joint centers. Shotton et. al[19] proposed a novel technique of part segmentation by treating the problem as a per-pixel classification in their seminal work that forms the core algorithm in the Kinect gaming system. We extend their method to a generic 3D visual hull data wherein the objective is to recognize parts of voxels to estimate skeleton and 3D pose of the human target.

4.1. Synthetic Training Data Generation

The bottom-up (discriminative) predictors trained on synthetic data are used to predict orientation and classify a voxel into part segments(see Fig. 1(c)). The human body is divided into 31 segments that are chosen based on the body part regions that move rigidly for articulated human body movement. These regions correspond to either the joint centers or the part of the human body. We differentiate these two segments as the segments corresponding to the body parts are recognized with higher confidence (by classifiers) compared to those corresponding to the joint centers. Fig. 2 illustrates the entire process of generating training data from the motion capture data. The HumanEva motion capture data is imported to smoothly deform 3D mesh shapes using autodesk Maya software[14](see Fig. 2(b)). This is done by skinning a mean 3D mesh shape [12] to an average skeleton (learned from a space of 70 skeletons from CMU motion capture dataset[1]) in Maya. For every frame of the motion capture sequence, a 3D human shape is generated by randomly sampling coefficients of PCA based 3D human shape model. The coefficients are sampled in the range from -1.5σ to $+1.5\sigma$ of top 10 bases vectors. The silhouettes are rendered for the deformed 3D shape from 4 different viewpoints, arranged in the same spatial configuration as the test HumanEva data (see Fig. 2(b)). As the visual hull reconstructed from the rendered silhouettes is already aligned to the 3D mesh vertices, voxels are labeled based on their spatial proximity to the vertices of the aligned 3D mesh. In an ideal data generation scenario, each sampled shape should be deformed using a skeleton specific to its shape. However for training purposes, the current approach gives reasonably good approximation of the 3D shape of the body of a human target with sufficient variance in their anthropometry. Fig. 2(b) shows the 3D visual hull (shown in red) aligned with the 3D mesh (shown in blue) sampled from the 3D human shape based PCA model. Also shown are the labeled parts of the visual hull using nearest neighbor. Fig. 2(c) shows the visual hull reconstructed from the real silhouettes in the reference frame with orientation estimated using regression model and the estimated part segments classified by the discriminative classifiers.

5. Training Discriminative Models for Part Segmentation

The labeled visual hull data are used to train discriminative models to predict part labels of the 3D voxels. We train sparse Bayesian regression model(Relevance Vector Machine) to predict the 3D orientation of the visual hull using the view-dependent shape descriptor(see fig. 1(b)). The reference frame is first transformed to the centroid of the visual hull and rotated to remove its orientation. We train SVM classifiers for recognizing parts from the visual hull in the transformed reference frame using the view-invariant 3D shape context descriptor.

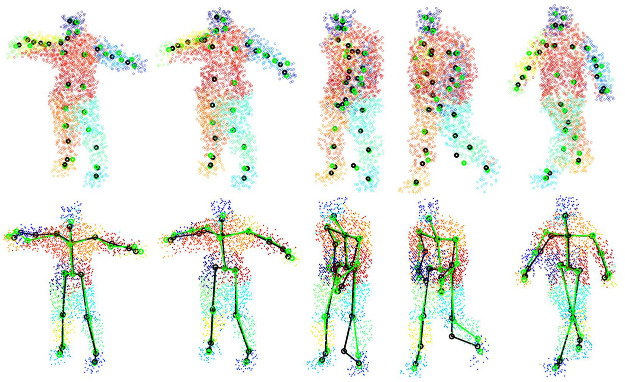


Figure 3. Joint location estimation for the synthetic data. (First row) Green markers are the part cluster means while the black markers are the joint locations obtained from mean shift clustering; Second row shows the skeleton estimated using mean shift (black) and the skeleton estimated using cluster centers as the joint locations (green). Notice that mean shift clustering improves the localization of the joint centers by using the part classification probability as an additional cue for computing similarity (or proximity) to the mean during clustering

Part Segmentation using Support Vector Machine: Support Vector Machine (SVM) is perhaps the most popular machine learning method for classification. SVM finds an optimal separating hyperplane by solving a quadratic progamming problem using Langragian multipliers. For all practical purposes linear SVM classifiers gives sufficient accuracy compared to other state of the art methods (such as Adaboost, Relevance Vector Machine, Gaussian Process Regression and Random Forest). We trained one-against-rest SVM classifiers for each part to classify a voxel into body part segments. SVM classifier typically gives signed distance from the separating hyperplane. We learn a logistic regression function using Iterative Reweighted Least Square (IRLS) optimization to calibrate the output response to a probability value

$$P(\mathbf{c}|f(\mathbf{x})) = \frac{1}{1+e^{-(Af(\mathbf{x})+B)}} \tag{1}$$

Where $f(\mathbf{x})$ denotes the signed distance response obtained from the SVM classifier for the input data $\mathbf{x}$ and $\mathbf{c}$ is the part classifier. The probability value is used to assigned part label to a voxel that has the highest probability. Since the framework requires detection of at least one part voxel, if no voxel is detected for a body part, we approximate the part center from top $K = 5$ voxels with highest confidence for that part.

6. Joint Location Extraction and Skeleton Fitting

The joint centers of a standard skeleton pose can be obtained by first computing the center of clusters for the body parts, and merging the centers to meaningful joint locations of the skeleton. However, doing so will not take into account the confidence of voxel classifications (from bottom-up models) in computing the joint centers. For example, the voxel classified to a part segment with low confidence should have lesser influence on the position of the joint center compared to the voxels strongly classified as a body part. Similar to [19], we use mean shift clustering that gives differential weights to voxels based on the probability of the part classifiers. Mean shift clustering is a non-parametric clustering algorithm that defines a density over a window in data space using a non-parametric form and iteratively computes a mean shift vector always pointing towards maximum increase in the density. The iteration eventually achieves a stationary point at the modes of the density. We define the kernel density as:

$$f_i(\mathbf{x}) = \sum_{i=1}^{M}\sum_{j=1}^{N} w_{ij}\text{exp}\{-\|\frac{\mathbf{x}-\mathbf{x}_j}{h_i}\|\} \tag{2}$$

here $\mathbf{x}$ denotes the location of voxels in 3D space, h_i is the bandwidth of the kernel for the i^{th} part of the total of M parts, x_j is the j voxel in the visual hull containing N voxels. The w_{ij} are the probability weights of associating j^{th} voxel to i^{th} part (mode) of the human body. Fig 3 shows the joint center locations as the cluster mean of voxels corresponding to a body part. The green markers show the cluster means and black markers show the joint locations obtained from mean shift clustering.

Skeleton Estimation from Anthropometric Prior: In order to fit a skeleton to the estimated joint locations, we first estimate skeleton specific to the human target. We model the parametric subspace of human skeletons using linear Principal Component Analysis(PCA). For a set of K initial frames, we project and back-project the joint center locations obtained for each frame to get a set of plausible skeleton shapes. The final skeleton is obtained as the mean of the set of skeleton obtained by back-projection.

Non Self-Intersection Constraints: We learn the parametric models for the space of human skeletons $\mathbf{L}$ and a coarse representation of the 3D human body $\mathbf{C}$ using cylindrical parts (see Fig4). The part dimensions of the coarse human shape model is obtained as average length and radius of human body parts computed from 3D human shape model acquired from MPI human shape dataset[12]. The shape parameters of the body parts $\mathbf{L} = [\mathbf{l}\ \mathbf{r}_1\ \mathbf{r}_2]$ include the length and the two radii of the tapered cylindrical human body parts. The cylindrical human model is used for computing self-intersection penalty during the skeleton fitting to the joint centers detected by the predictive models. The cost of parts self-intersection is computed as the difference between the shortest distance D between the two axes of the cylindrical body parts of radii R_1 and R_2. For the two intersecting parts, we add a penalty term proportional to $max(0, R_1 + R_2 - D_{12})$ (see fig. 4) in the overall cost function (discussed next). The optimistic bound, although overestimates the intersection scenarios, allows fast computation of the penalty terms for the intersecting body parts during the optimization.

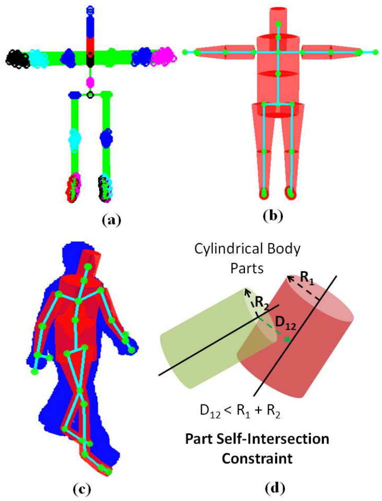


Figure 4. *(a)* Space of articulated human skeletons, *(b)* Coarse human cylindrical model with average radii estimated from part dimensions of MPI human shape dataset[12], *(c)* Coarse human model rigidly deformed for a given pose and aligned with the 3D visual hull ; (d) Parts self-intersection constraint used for fitting a skeleton to joint centers.

Levenberg-Marquardt(LM) Optimization: LM is a robust unconstrained optimization algorithm used for searching in parameter space of a non-linear function. The method minimizes the least square error between the observed data points and function values by iteratively improving the estimates of the function parameters. In our framework, we search over joint angles of a skeleton to estimate the 3D pose that best aligns the joint locations to the observed part cluster centers. Of the 31 part segments that we use in our framework, we associate part centers to the skeleton joints by categorizing the part segment as joint P^j or segments P^s. Based on our experimental evaluation, we have observed that parts corresponding to the body segments are detected with higher confidence compared to the parts corresponding to joint centers. We incorporate 4 types of cost terms in the error function to be minimized using the LM method.

$$
\begin{aligned}
\mathcal{C}(\Theta) = \sum_{i=1}^{M^j} \alpha_1 \left[P_i^j - J_i(\Theta)\right]^2 + \\
\sum_{k=1}^{M^s} \alpha_2 \left[P_k^s - \frac{(J_{k1}(\Theta) + J_{k2}(\Theta))}{2}\right]^2 + \qquad (3) \\
\sum_{(l,m)\epsilon C}^{I_{lm}} \alpha_3 \mathbb{I}(D_{lm} < (R_l + R_m)) + \\
\sum_{n\epsilon N_\theta}^{N_\theta} \alpha_4 \left[[\max(\theta_n, \theta_n^{max}) - \theta_n^{max}]^2 + \left[\theta_n^{min} - \min(\theta_n, \theta_n^{min})\right]^2\right]
\end{aligned}
$$

The first error term is for matching the skeletal joint location $J_i(\Theta)$ to the cluster centers of parts P_i^j corresponding to human body joint centers. The second error term is for matching the midpoint of the segments to the cluster centers of parts P_k^s corresponding to the human body segments. We always assign $\alpha_2 = 2 \times \alpha_1$ due to higher classification confidence of the part clusters corresponding to skeletal segments. Third error term is a fixed penalty term if two body parts intersect. We predefine the set C to contain possible pair of body parts that can intersect. These pairs include left-right lower and upper legs, left-right lower and upper arms and can intersect with each other or with the torso. The final cost term is due to constraints to enforce the joint angle limits $\left[\theta_n^{min}, \theta_n^{max}\right]$ for specific joints of the human skeleton. The iterative cost optimization algorithm based on LM incrementally improves initial parameters by solving a regularized least-square regression at each step. The initial estimate is obtained by computing joint angles with all degrees of freedom. The LM iteration then proceeds by restricting specific degrees of freedom and adding constraints shown in (3) in stages to allow gradual minimization of the cost.

7. Experimental Evaluation

We evaluated our system on HumanEva dataset. For generating synthetic training data we used Autodesk Maya[14] and PCA based human body shape model provided by Hasler *et.* al[12]. Training sequences are generated for each frame by randomly sampling 3D mesh shapes in the canonical pose and deforming the pose using Maya animation libraries.

3D Human Pose Representation and Visual Hull Descriptors: Similar to the intermediate representation adopted by kinect [19], we represent human body using 31 body parts. Skeleton is modeled using 30 joints of varying degrees of freedom and a root joint for rigid translation and orientation. However, we used only 17 significant joints for fitting the skeleton to the part centers of the 3D visual hull. For learning the 3D orientation of the visual hull, we extract the cylindrical shape descriptor with 10 bins along the axis of the cylinder, 5 radial bins and 8 angular bins, thus generating a 3D descriptor of size 400. For learning orientation, we convert to angles to sinusoidal space and then reduce the dimensions to 3 using PCA. 3D Shape context descriptor has 10 radial bins, 10 bins along the elevation and 16 bins along the azimuth, generating a 1600 dimensional vector for the shape descriptor for a visual hull voxel. Both for orientation estimation and part segmentation, we sample 2000 voxels from the visual hull (40K voxels) uniformly. Whole visual hull shape is therefore represented by a 2000x1600 sized shape matrix.

Model /Axis	Rot. error along X-axis	Rot. error along Y-axis	Rot. error along Z-axis
Linear (S1 Jog)	15.52	11.64	26.72
Non-Linear (S1 Jog)	15.03	12.65	26.72
Linear (S2 Jog)	6.0	8.32	21.94
Non-Linear (S2 Jog)	5.34	6.34	19.04
Linear (S3 Jog)	8.95	14.64	22.74
Non-Linear (S3 Jog)	8.8	13.8	17.98
Linear (S2 Box)	10.67	17.75	14.90
Non-Linear (S2 Box)	6.28	7.55	13.59

Table 1. Visual hull orientation angle computation accuracy for the linear and non-linear regression models (using Relevance Vector Machine) trained on synthetic visual hull data and tested on real visual hull data of 3 different subjects and 2 actions (jogging and boxing)

Translation and Orientation Estimation: We always compute the 3D shape context descriptor for the visual hull in the reference frame centered at the root joint of the human. Translation is estimated as a fixed offset from the visual hull centroid. The 3 dimensional rotation is estimated using the RVM based regression models. Table 1 shows the orientation angle prediction accuracy of the RVM regression models trained on synthetic jogging and boxing motion capture sequence for the subject S2.

Part Segmentation Accuracy: We compared part classification accuracy using SVM based one-vs-rest classifier and Random Forest (RF) classifier. For the RF, we trained 300 trees with 10 features randomly selected at each node and maximum depth of each tree set to 20. Although the model trained using Random Forest occupied significantly larger disc space than the models trained using SVM, classification time using random forest was much lesser compared to SVM based classifiers. Fig. 5 compares the confusion matrix for classify 31 parts. The naming of the parts are as follows - *u* in prefix denote upper, *l* denotes left, *r* denotes right and *w* denotes lower. In our evaluation we did not observe any significant difference in prediction accuracy of random forest classifier compared to SVM classification. Our SVM based one-vs-rest classifier gave an accuracy of 54.633% whereas RF gave an overall accuracy of 55.05%

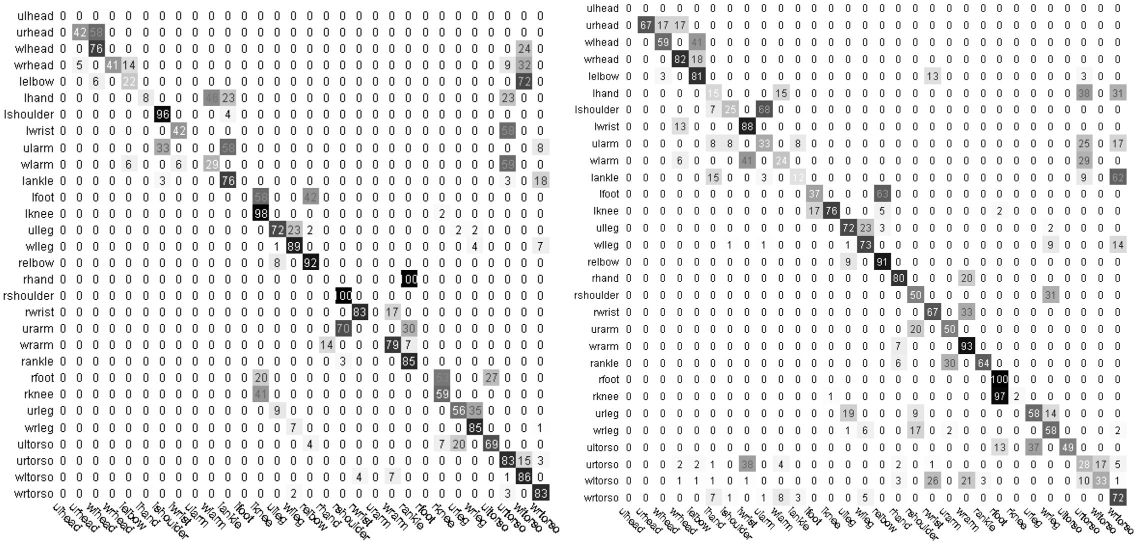


Figure 5. Confusion matrix for part classification *(left)* using Random Forest(RF) of depth 20 giving an overall recognition rate of 55.05% on 200 frames walking sequence and *(right)* using Support Vector Machine(SVM) giving an overall accuracy of 54.633%

on a synthetic walking sequence of 200 frames.

Joint Localization Accuracy: We evaluated our algorithms on real HumanEva I dataset[21] on jogging and walking sequences for subjects S1, S2 and S3. In the experiments we trained the classifiers on synthetic data obtained by importing HumanEva motion capture sequences to animate and deform 3D human mesh. We used only 100 frames (51 from S2 jog and 49 from S2 Box) to train the classifiers. The models trained on synthetic data was used to segment and estimate pose of visual hull computed from the real data sequence (500 frames each). Table 2 compares the average joint location error (in mm) for different sequences. For each sequence we show the avg. joint estimation accuracy with using ground truth orientation and with predicting the orientation using the learned RVM regression. Notice as shown in Table 1 that although the average orientation error is $15^o - 30^o$ for some sequences, there is only a small degradation in the joint estimation accuracy. The table also shows that skeleton fitting always improves the joint estimation accuracy.

Data Seq.	Joint Loc. Error (mm) Before Skeleton Fit	Joint Loc. Error (mm) After Skeleton Fit
With GT Orient. (S2 Jog)	75.0301	71.261
With Pred. Orient. (S2 Jog)	77.1602	74.1372
With GT Orient. (S3 Jog)	83.745	79.57
With Pred. Orient. (S3 Jog)	84.25	81.04
With Pred. Orient. (S1 Jog)	90.952	85.5072
With GT Orient. (S2 Box)	75.062	81.205
With Pred. Orient. (S2 Box)	88.58	90.72

Table 2. Average joint estimation accuracy on jog and box sequences of subjects S1, S2 and S3 in HumanEva dataset. The two columns compare the joint prediction accuracy before and after fitting the skeleton to the joint locations estimated as centers of part clusters. The table also compares joint estimation accuracy using ground truth(GT) orientation and using a learned regression model to estimate orientation of the visual hull

8. Conclusion

In this work we have developed and evaluated a part-based algorithm for estimating 3D pose of human targets by segmenting its 3D visual hull reconstructed from synchronized video streams acquired from 4 calibrated visual sensors. A significant advantage of our approach is that it can be trained on only synthetically generated data. Our pose estimation framework employs generic 3D visual hull shape descriptors that are sufficiently discriminative, invariant to target appearance, robust to perturbations in the input data and can be applied to infer poses of humans of varied anthropometry. Experimental evaluation gave promising results both in terms of joint prediction accuracy and their ability to generalize to different human subjects.

Acknowledgements: This work was supported by Air Force Research Lab, contract number FA8650-10-M-6094.

References

[1] Cmu motion capture dataset http://mocap.cs.cmu.edu. -, 2000.

[2] B. Allen, B. Curless, and Z. Popovic. The space of human body shapes: reconstruction and parameterization from range scans. *SIGGRAPH*, 2003.

[3] A. Balan, L. Sigal, M. Black, J. Davis, and H. Haussecker. Detailed human shape and pose from images. *CVPR*, 2007.

[4] A. O. Balan and M. J. Black. The naked truth: Estimating body shape under clothing. In *ECCV (2)*, pages 15–29, 2008.

[5] S. Belongie, J. Malik, and J. Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *NIPS*, pages 831–837, 2000.

[6] A. M. Elgammal, D. Harwood, and L. S. Davis. Non-parametric model for background subtraction. In *ECCV (2)*, pages 751–767, 2000.

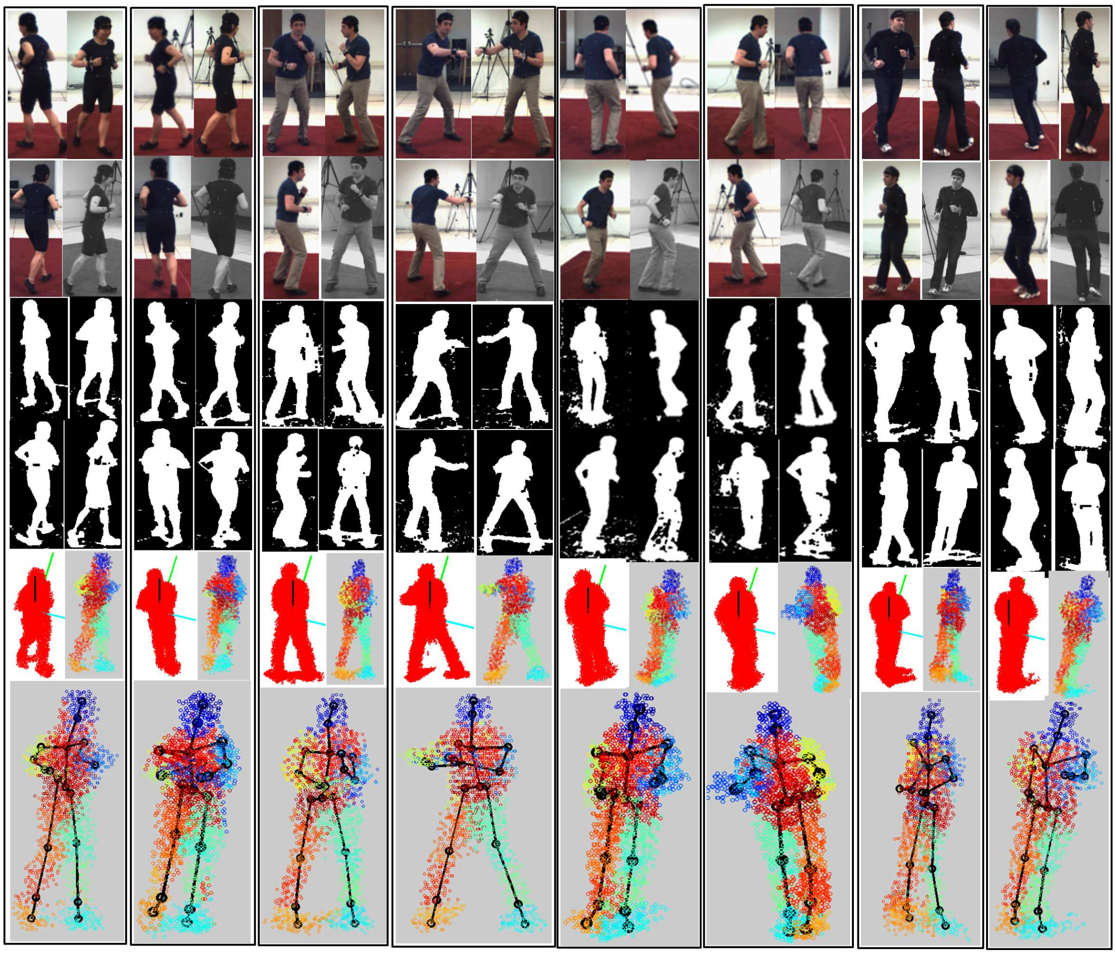


Figure 6. 3D pose estimation results on selected images from the HumanEva dataset that was used for evaluating the system. Each column shows the image and corresponding silhouette acquired from 4 sensors, the reconstructed visual hull in the human centered reference axis, part classification of visual hull voxels and the fitted 3D skeleton. We show results on jogging and boxing sequences for the 3 subjects of HumanEva data

[7] J. Gall, C. Stoll, E. de Aguiar, C. Theobalt, B. Rosenhahn, and H.-P. Seidel. Motion capture using joint skeleton tracking and surface estimation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1746–1753, 2009.

[8] J. Gall, A. Yao, and L. J. V. Gool. 2d action recognition serves 3d human pose estimation. In *ECCV (3)*, pages 425–438, 2010.

[9] V. Ganapathi, C. Plagemann, D. Koller, and S. Thrun. Real time motion capture using a single time-of-flight camera. *CVPR*, 27(3), 2010.

[10] V. Ganapathi, C. Plagemann, D. Koller, and S. Thrun. Real-time human pose tracking from range data. In *Proceedings of the 12th European conference on Computer Vision - Volume Part VI*, ECCV'12, pages 738–751, Berlin, Heidelberg, 2012. Springer-Verlag.

[11] R. B. Girshick, J. Shotton, P. Kohli, A. Criminisi, and A. W. Fitzgibbon. Efficient regression of general-activity human poses from depth images. In *ICCV*, pages 415–422, 2011.

[12] N. Hasler, C. Stoll, M. Sunkel, B. Rosenhahn, and H.-P. Seidel. A statistical model of human pose and body shape. In P. Dutr'e and M. Stamminger, editors, *Computer Graphics Forum (Proc. Eurographics 2008)*, volume 2, Munich, Germany, Mar. 2009.

[13] M. Hofmann and D. M. Gavrila. Multi-view 3d human pose estimation combining single-frame recovery, temporal integration and model adaptation. *In ACM Transactions on Graphics*, 27(3), 2009.

[14] http://www.autodesk.com. Maya autodesk. *CVPR*, 2009.

[15] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2-3):90–126, 2006.

[16] L. Mündermann, S. Corazza, and T. P. Andriacchi. Accurately measuring human movement using articulated icp with soft-joint constraints and a repository of articulated models. In *CVPR*, 2007.

[17] G. Pons-Moll, A. Baak, T. Helten, M. Müller, H.-P. Seidel, and B. Rosenhahn. Multisensor-fusion for 3d full-body human motion capture. In *CVPR*, pages 663–670, 2010.

[18] M. Shimosaka, Y. Sagawa, T. Mori, and T. Sato. 3d voxel based online human pose estimation via robust and efficient hashing. In *ICRA '09. IEEE International Conference on Robotics and Automation*, pages 3577–3582, 2009.

[19] J. Shotton, A. W. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, pages 1297–1304, 2011.

[20] L. Sigal, A. O. Balan, and M. J. Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, editors, *NIPS*. MIT Press, 2007.

[21] L. Sigal and M. J. Black. Humaneva: Synchronized video and motion capture-dataset for evaluation of articulated human motion. *Technical Report CS-06-08, Brown University, 2006 http://vision.cs.brown.edu/humaneva/*, 2006.

[22] C. Stoll, J. Gall, E. de Aguiar, S. Thrun, and C. Theobalt. Video-based reconstruction of animatable human characters. *ACM Trans. Graph.*, 29(6):139, 2010.