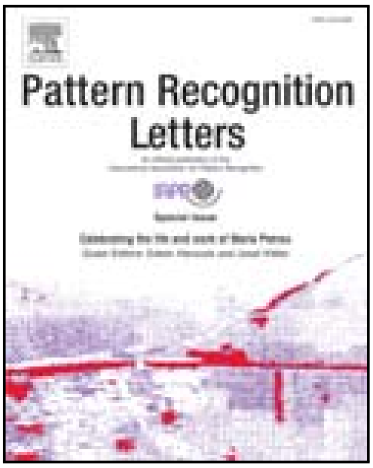

Multi-part Body Segmentation based on Depth Maps for Soft Biometry Analysis

Meysam Madadi, Sergio Escalera, Jordi Gonzàlez, F.Xavier Roca, Felipe Lumbreras

PII: S0167-8655(15)00036-7
DOI: 10.1016/j.patrec.2015.01.012
Reference: PATREC 6161

To appear in: *Pattern Recognition Letters*

Received date: 11 March 2014
Accepted date: 5 January 2015

Highlights

- A system for RGB-Depth human body segmentation and description is presented.
- Body clusters are automatically computed and a multi-class classifier is trained.
- 3D alignment is performed within an iterative 3D shape context fitting approach.
- We show robust biometry measurements by applying orthogonal plates to body hull.
- Results on a novel data set improve segmentation accuracy in relation to RF.

Multi-part Body Segmentation based on Depth Maps for Soft Biometry Analysis

Meysam Madadi[a,*], Sergio Escalera[a,b], Jordi Gonzàlez[a,c], F. Xavier Roca[a,c], Felipe Lumbreras[a,c]

[a]Computer Vision Center, Campus UAB, Edifici O, 08193, Bellaterra, Spain.
[b]Dept. Matemàtica Aplicada i Anàlisi, Universitat de Barcelona, Gran Via de les Corts Catalanes 585, 08007, Barcelona, Spain.
[c]Dept. Computer Science, Universitat Autònoma de Barcelona, Edifici Q, 08193, Bellaterra, Barcelona, Spain.

Abstract

This paper presents a novel method extracting biometric measures using depth sensors. Given a multi-part labeled training data, a new subject is aligned to the best model of the dataset, and soft biometrics such as lengths or circumference sizes of limbs and body are computed. The process is performed by training relevant pose clusters, defining a representative model, and fitting a 3D shape context descriptor within an iterative matching procedure. We show robust measures by applying orthogonal plates to body hull. We test our approach in a novel full-body RGB-Depth data set, showing accurate estimation of soft biometrics and better segmentation accuracy in comparison with random forest approach without requiring large training data.

Keywords: 3D shape context, 3D point cloud alignment, depth maps, human body segmentation, soft biometry analysis

1. Introduction

Soft biometrics in contrast to hard biometrics are traits of the human body, like color of the hair, skin, height and weight, that can be used to describe a person. These attributes have a lower power to discriminate and authenticate an individual, but they are easier to compute in comparison to hard biometrics.

Soft biometric traits have been used in video surveillance to track people with single camera systems or even with a discrete joint camera network (Demirkus et al. (2010); Denman et al. (2012); Ran et al. (2008)); as a preprocessing approach to help hard biometric systems to search databases faster or to increase reliability and accuracy (Guo et al. (2010); Mhatre et al. (2001)); and for other applications like person re-identification (Møgelmose et al. (2013)), supported diagnosis in clinical setups (Reyes et al. (2013)), or commercial tools like clothing sizing (Chen et al. (2011)), just to mention a few. Most surveillance systems using soft biometrics have integrated human height as one of their most important cues (Denman et al. (2012); Jeges et al. (2008); Ran et al. (2008)).

Velardo et al. (2011) proposed a weight estimation technique that computes weight by summation of coefficients of some soft biometrics like height and calf circumference. Since soft biometrics have semantic correlation in human metrology, these can be computed according to part relations. Recently, Adjeroh et al. (2010) studied the problem of predictability and correlation in human metrology applying some statistical measurements between different soft biometrics features in order to make correlation clusters among them to predict unknown body measurements. Samejima et al. (2012) used joints estimated by KinectSDK to estimate initial dimensions, afterwards multiple Regression of the 2 principal components of estimated body dimensions were applied to estimate other dimensions. Weiss et al. (2011) computed body measurements using a regression based approach from body parameters after an accurate scanning of the

[*] Corresponding author: mmadadi@cvc.uab.es (Meysam Madadi)

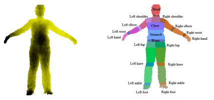

Figure 1: A typical depth image and defined segments.

body.

Indeed the extraction of human body part traits in soft biometric systems, as other areas in computer vision, suffers from difficulties like illumination changes, cluttered and uncontrolled environments, and the fact of dealing with the articulated nature of the human body. Recently, Microsoft-Corp. (6/2012) has launched a low price multisensor device that uses pseudo random structured light technology that is capable of capturing RGB images and depth information simultaneously, which makes it possible to acquire 3D coordinates of pixels with high accuracy in indoor environments and overcome most of the difficulties aforementioned.

While most of the biometrics measurements are based on regression on some known body parameters, in this paper, first we accurately segment human limbs from a single depth image captured by a Kinect camera, and as a result we compute traits such as arm and leg lengths, and neck, chest, stomach, waist and hip sizes from segmented limbs. We use Kinect to get the human point clouds using background subtraction and depth thresholding from real user data, see in Figure 1 a typical pose, depth image, and the corresponding segments. As a first stage, we focus on human pose estimation as a multi-limb segmentation problem (Laxton (2007)). Two general approaches are defined for this task: model based and model free techniques. In model based approaches, a kinematic model approximates the shape of the body from measurements that best fit the observed image features (Bo and Sminchisescu (2010); Schwarz et al. (2011); Sminchisescu et al. (2011); Zhu and Fujimura (2010)). Andriluka et al. (2010) successfully combined bottom-up part-based models with 3D top-down priors and showed the models capable to deal with more complex poses. Ramanan (2006) proposed an edge based deformable model learned by a conditional random field, and used an iterative parsing as an energy minimization function to improve recognition. Several works are based on this approach as a first stage for human pose recovery and behavior analysis applications (Ferrari et al. (2009)). Recently, Ye et al. (2011) used an example-based approach which finds the pose from the nearest sample after registration.

The methods for human pose estimation based on depth data have mainly focused on model free approaches. Model free approaches use feature vectors to learn and map feature space to pose space. Recently, Shotton et al. (2011) proposed a random forest based approach to learn pixel labels from depth offsets, achieving robust segmentation results. This method has become one of the standard techniques for segmentation in depth data. However, this approach requires a huge dataset of real and synthetic labeled images as well as an expensive training procedure. Different works have focused on such a random forest segmentation approach to improve recognition of human body parts. Hernandez-Vela et al. (2012) applied graph cuts to perform a local and spatial optimization of random forest output probabilities in order to improve segmentation accuracy. Kohli et al. (2012) proposed a conditional regression forests approach applying a global latent variable that incorporates dependency between output variables, increasing body joint prediction.

In this work we use a model based system where labels of pixels are computed from a defined model after 3D alignment with the objective of performing soft biometrics analysis. For this task, we extract a depth image of each frame in the training set, and compute HOG features (Agarwal and Triggs (2006); Poppe (2007); Shakhnarovich et al. (2007)). The described data is clustered to group similar poses in the same class in order to find the closest model to the test sample as fast as possible at test time instead of searching all the data set. The number of clusters is defined using a Gaussian mixture in an EM algorithm. With such an optimization, we are able to accurately cluster training data in a problem-dependent way without the need of prefixing clustering parameters.

Subsequently, the model is aligned to the test body sample in the 3D space using 3D shape context descriptors and 3D thin plate spline (TPS). Using HOG as a pose recognizer does not require 3D shape context to be invariant to rotation or viewpoint changes, although 3D shape context can be rotated based on eigenvectors of the point cloud. For our task we apply Körtgen et al. (2003) 3D shape context for aligning point clouds of body hulls. In our procedure, a random number of pixels is selected and refined, removing nearest adjacent points, and then an iterative process finds the best matching points. Moreover each pixel in the body gets the nearest pixel label in the aligned model. As a result of this step we found accurate fitting of body parts without requiring expensive training procedures. Finally, joint points are computed from the segmented body parts. The intersection of a thin plate orthogonal to the body crossing each joint point and the body hull selects which pixels will be used for measuring the corresponding trait.

To validate our work, we need a motion capture data set with limbs pixel labels and traits ground truth. Therefore, We have validated our proposed system on a novel data set of human poses, showing high segmentation accuracy and soft biometrics estimation. In particular, we found better segmentation performance than random forest approach.

The rest of the paper is organized as follows: Section 2 presents the details of our method, then experiments and results are described in section 3, and finally we conclude our work in section 4.

2. Limbs labeling and size measurements

In this section we review 3D shape context and TPS, describe our system for human limb segmentation and soft biometrics computation, whose different modules are shown in Figure 2.

2.1. Training

The Histogram of Oriented Gradients (HOG) descriptor has been studied vastly in the domain of human detection and pose recognition. Here, the key idea is to use HOG as the depth descriptor of the human body on depth images, where the gradients of the depth image are the derivatives of the body hull surface.

Once HOG feature vectors have been computed, our approach is based on modeling homogeneous pose clusters within a training set of depth human poses using a multi-class classifier. Then the sample pose models are computed from the nearest neighbors of each cluster. We use a problem-dependent clustering strategy to group HOG feature vectors of poses, as described next.

To cope with the problem of determining the exact number of clusters, we estimate the optimum number of clusters by combining the EM and k-means algorithms as proposed in Lee et al. (2006): let $X = \{x_1, ..., x_N\}, x_i \in R^d$ be a given data set. An iterative algorithm starts from M_{min} to M_{max}, $M \in \{1, ..., N\}$, and at each iteration EM algorithm is initialized with the clusters of X obtained from k-means ($k = M$); then the parameters of the mixture model and the posterior probabilities of the members of X are computed. At the end of each iteration the mutual relationship between every two mixtures is measured as:

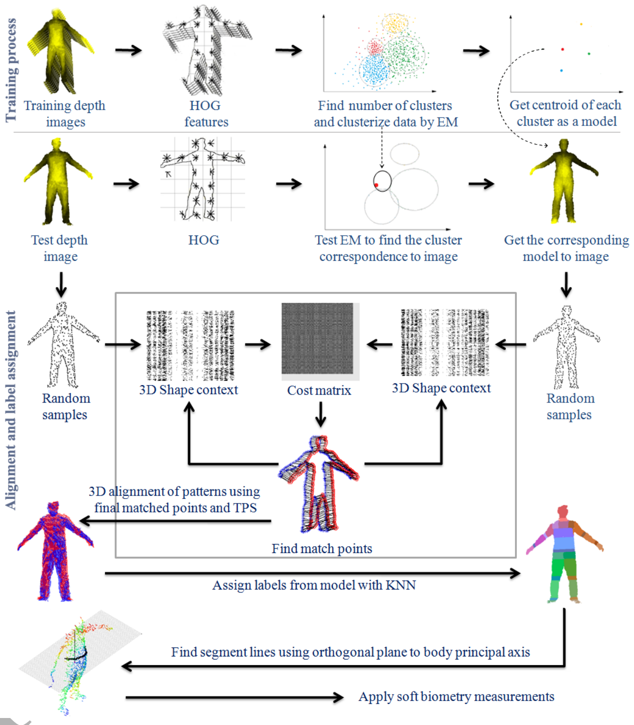
Training process
Training depth images
HOG features
Find number of clusters and clusterize data by EM
Get centroid of each cluster as a model
Test depth image
HOG
Test EM to find the cluster correspondence to image
Get the corresponding model to image
Alignment and label assignment
Random samples
3D Shape context
Cost matrix
3D Shape context
Random samples
3D alignment of patterns using final matched points and TPS
Find match points
Assign labels from model with KNN
Find segment lines using orthogonal plane to body principal axis
Apply soft biometry measurements

Figure 2: Process diagram.

$$\psi(i, j) = p(i, j) \log_2 \frac{p(i, j)}{p(i)p(j)}, i = 1, ..., j, \qquad (1)$$

where $p(i) = \frac{1}{N} \sum_{n=1}^{N} p(i|x_n)$ is the probability of the mixture i and $p(i, j) = \frac{1}{N} \sum_{n=1}^{N} p(i|x_n)p(j|x_n)$ is the joint probability of i and j mixtures. For any composition of $i, j \in \{1, ..., M\}$, if $\psi(i, j) > 0$, then i and j mixtures are considered statistically dependent so the process finishes and $M - 1$ is returned as the most suitable number of mixtures.

One of the limitations of such an approach is when there is no 'meaningful' mixture, i.e. when the number of training poses is low or when the data does not follow normal distributions. In the case of straggly or scarce data, algorithm goes to reach M_{max} where each data is assumed as a cluster itself. When the data distributions are not normal, we can tune M_{min} and M_{max} to solve this problem. After estimating the optimal M, the EM is trained and the labels and feature vectors of each model are kept. EM algorithm shows better cluster results than k-means, besides we can keep the parameters of EM and retrieve them later in order to predict the cluster of a new feature vector applying the posterior probabilities of the feature vector and the mixtures. This is useful to retrieve the closest model to test point cloud in an efficient way at test step.

Moreover, extracted nearest model should be aligned to the input body point cloud. For this purpose, we extract sample points from each body point cloud to decrease the overall fitting time while preserving performance as follows: we first select a number of random points and then refine them by eliminating undesired nearest adjacent points till reach a constant portion of each point cloud. The refinement step includes computing the Euclidean distances among all the selected points, finding the shortest distance, and removing one of the ending points[1]. This process enables us to consider a normal distribution for the points and, at the same time, alignment is more accurate in the edges.

Subsequently, we compute the normalized gradients of each axis as the mean values and angles of the gradients for selected points in the depth image. Assume $P_{N\times3}$ is the matrix of selected x, y, z points in the depth image coordinates of the model, $t_{N\times1}$ is the matrix of gradient angles and $g_{N\times2}$ is the matrix of normalized gradients. Then, the point cloud for the model is defined as:

$$P_{tan} = P + \alpha \left[g \circ \left[\cos(t) \mid \sin(t) \right] \mid 0 \right]_{N\times3}, \qquad (2)$$

where α is a static constant and $A \circ B$ is the Hadamard product of A and B. We use P_{tan} later to compute new gradient angles matrix of the model after converting it to real world coordinates.

2.1.1. Point matching

Here, we employ the basic shape context of Belongie et al. (2002) extended to 3D data. We propose to use exponential space for the radius of nested spheres (n_r) as:

$$r_i = \frac{10^{i/n_r} - 1}{9} \times max(\{\|x - y\| \,|\, x, y \in \omega\}), i = 1, ..., n_r, \quad (3)$$

where ω is the set of inlier points (vs. outliers). This space partitioning forces shape context to be more sensitive to near samples to the sphere bin than farther ones. After computing the shape context histograms for all selected points, the best matched points between all pairs of points on the model and the input test pattern are found. Such a matching process minimizes the overall matching cost, for which a cost table performs matching based on the histogram similarity and appearance similarity of the points. As in Belongie et al. (2002), we use χ^2 test to find the histogram similarity cost and the gradient angular difference polarity to find the appearance cost between pairs of (p_i, q_j) points of the two point clouds. So the cost function is defined as:

$$C(p_i, q_j) = \frac{1}{2} \left((1-\alpha) \sum_{k=1}^{\frac{n_r n_\theta^2}{2}} \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)} + \alpha(1 - \cos(t_i - t_j)) \right), \qquad (4)$$

where n_θ must be an even number, $h_i(k)$ and $h_j(k)$ denote the k-th bin of the histogram, and t_i and t_j are the gradient angles at p_i and q_j, respectively. The appearance cost acts as a penalty function causing smooth alignments on the surfaces, while the α coefficient controls this smoothing factor.

Redundant, "dummy" points are also added to the cost table with a constant cost to control the sensitivity of the shape context to noise as in Belongie et al. (2002). Therefore, points that do not match any other with a lower value than this dummy cost will be considered as outliers. In hard assignments, each point matches exactly one point in the cost table so that the overall cost is minimum. This task, commonly referred to as Linear Assignment Problem (LAP), can be solved using Jonker and Volgenant (1987).

2.1.2. Transformations

Sample points are aligned after matching to generate new coordinates and gradient angles which will be used

[1] In order to perform this task efficiently, we set an infinity value to that value in the distance table without recomputing the whole table, setting the flag of the removed point to undesired. At the end, we add the desired points to the final list.

in next iteration to refine the final matched points. This alignment task is done by generating an interpolation matrix using the best matches of random samples and thin plate spline (see Belongie et al. (2002); Bookstein (1989)) in the form of:

$$T = \begin{bmatrix} K_{N\times N} & [1|P_{N\times 3}] \\ [1|P_{N\times 3}]^{\top} & 0 \end{bmatrix}^{-1}_{(N+4)^2} \begin{bmatrix} Q_{N\times 3} \\ 0 \end{bmatrix}_{(N+4)\times 3}, \quad (5)$$

where K is a kernel matrix, P is the best model matching points matrix, and Q is the best test pattern matching points matrix. K is computed as:

$$K_{ij} = \begin{cases} \|P_i - P_j\| \log(\|P_i - P_j\|) & \text{if } i \neq j, \\ \lambda & \text{if } i = j, \lambda > 0, \end{cases} \quad (6)$$

where $\lambda = \alpha^2 \lambda_0$ is a regularization parameter used to smooth the interpolation. The term α is defined as:

$$\alpha = \frac{1}{N} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \|P_i - P_j\|, \quad (7)$$

and λ_0 is a scaling factor. We change 0 values of $\left\|P_i - P_j\right\|$ in the indices matrix K to 1 to avoid generating $-\infty$ in logarithm. Then, model sample points are mapped to their interpolated locations in the test pattern using the interpolation matrix T and the same procedure specified in equation 5 in the form $[K|1|O]T$, where O is the point cloud model and $K_{ij} = \|O_i - P_j\| \log(\|O_i - P_j\|)$. These new mapped points are sent to the next iteration along with new gradient angles. New gradient angles are updated in a procedure as follows: image coordinates of the mapped model samples are computed, P_{tan} points computed in previous sections are mapped using the matching points and the alignment procedure, and their image coordinates are estimated; finally the angles extracted from the differences of the coordinates of 2D mapped model and P_{tan} sample points are returned as new gradient angles. Figure 3 illustrates an alignment example within the described iterative process.

2.2. Label assignment

In order to cope with self-occlusions, we maintain a complete point cloud for each sample model. Once alignment has been done, the complete model is transformed to the test point cloud using the matching points of the former described approach through TPS. This warped complete model is used to complete occluded parts and assign labels. We can easily estimate the label of each point using its nearest neighbor pixel label after alignment of the point clouds by applying the matching points and the transformation procedure described above. Unfortunately, assignments of labels from 3D nearest neighbors cause problems in the case of imperfect alignments and broken segments. To minimize this issue, as well as noise points, we proposed to train SVM directly on 3D coordinates of warped model using a linear kernel and predict test points labels from it. SVM makes a bound around points and tune the assignments.

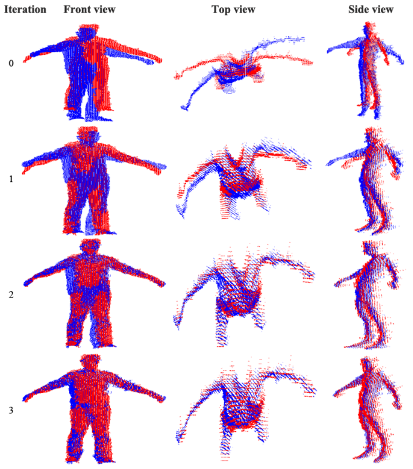

Figure 3: Iterative alignment process shows how points get closer at each step.

2.3. Size measurements

To measure size, we complete body from complete warped model. Therefore, we save occluders labels for the model image. After label assignment, we use those test points with occluder label as a mask and select those non visible points of model inside the mask as completion points. Figure 4 shows how occludees are completed. Following this procedure, the lengths of arms and legs are easily obtained after extracting the points of their joints like shoulder, elbow and wrist for arms; or hip, knee and ankle for legs. But the most challenging part in size measurements lies in the estimation of the circumference of

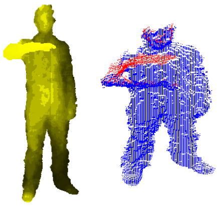

Figure 4: Red points are non visible points of the model inside occluders mask. Left image is the corresponding depth map.

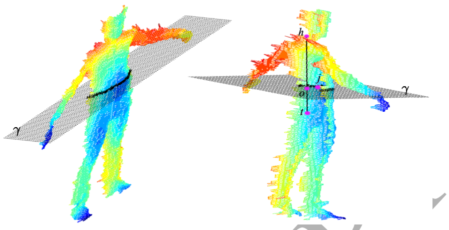

Figure 5: Constructing trait curve for size measurement using orthogonal plane γ to the body principal axis.

body traits like neck, chest or waist. Depending on the training data, one could add more body parameters like principle components as degree of obesity and using such a more robust model, full size measurements are possible from body completion. This can obtain even more accurate results in multi-view systems which require a point cloud nearest neighbor based approach.

Next we describe a geometrical approach to compute camera view circumference of such traits: we estimate the orthogonal plane to the body principal axis so that the intersection of this plane and the body hull surface is used for estimating those measurements. Since the principal axis of the body is the symmetry axis of it, we assume that this axis starts at the mean point of the hip edge and ends at the mean point of the neck edge. This edge lies on the segment after estimating labels. The accuracy of principal axis finding strongly depends on the segmentation accuracy. Similarly, the principal axis of the neck starts at the mean point of the neck edge and ends at the head joint.

As shown in Figure 5, let h be the head point and t be the tail point of the principal axis, j be the joint point of a segment, γ be the orthogonal plane to the principal axis crossing at j, and o be the intersection of γ and the principal axis. In this assumption, o is unknown and we compute it using other known points as:

$$o = \alpha(h - t) + t, \tag{8}$$

where α is the plane γ factor computed as:

$$\alpha = \frac{\sum(h - t) \circ (m - t)}{\sum(h - t)^2}. \tag{9}$$

Let p be a point on body hull that belongs to the selected segment. Point p lies on γ plane if and only if $\overrightarrow{op} \cdot \overrightarrow{oh} = 0$ is satisfied. Since the body hull point cloud is a discrete surface, we threshold the dot product for all points in the segment to estimate the intersection curve. However, the resulting narrow strip of points is still not appropriate for measurements. We divide the resulting strip into non-overlapping segments to extract the mean point of each segment, and then consider the Euclidean distance between neighbor segments. Applying such an interpolation reduces affects of boundary points noises. These measurements can be used in regression based approaches as initial parameters.

For small segments like neck, using completed point cloud and tuning the weights of each segment in SVM segmentation improves segment line analysis. An important parameter is the threshold of dot product which can be tuned for different point cloud densities and segments.

3. Experiments and results

To evaluate our method, we have created a dataset [2] manually labeled containing 1155 frames of 38 individuals, 7 females and 31 males, with a resolution of 640×480 pixels captured by a Kinect using the OpenNI library OpenNI (6/2012). Each frame consists of RGB image, and depth information and label of each body pixel, a complete model in the self-occlusion cases, as well as, ground truth values of the front views of limbs sizes with a ±20 mm human error in measurements, and a file containing occluder segments. Subjects rotate facing the camera in a range of ±60° such that the whole body was observable.

[2] This dataset will be made publicly available.

We used a 10-fold cross validation over all 1155 frames to generate the results. The segmentation error per frame is the proportion of mislabeled pixels in relation to the total number of pixels. Then, the overall error is averaged.

We have used a block of 9 × 6 including 2 × 2 cells and 8 orientation bins for HOG. Figure 6 illustrates some clusters contents and shows how the poses and bodies are clusterized together. We achieved the best number of clusters at 90 in a range of [15..100] clusters. Although the results show the robustness of our approach to the HOG parameters, we found that the parameter values give the best alignment results: $n_\theta = 8$, $n_r = 15$, 500 random samples, dummy cost 0.1, 35% of samples as additional dummy points, appearance cost weight 0.15, and 4 iterations. We set the parameter λ_0 to 1000 for first iteration to have an affine transformation and br^{k-1} for next iterations where $b = 0.9$, $r = 1.7$, and k is the iteration number.

In order to compare our approach, we have considered the random forest (RF) pixel labeling approach as defined in Shotton et al. (2011). In particular, the RF implementation computes the weights of each body part label as:

$$W_l = 0.5 + \frac{P(l)}{\sqrt{\sum_{i=1}^{N} P(i)^2}}, \tag{10}$$

where $P(l)$ is the probability of the label l and is equal to the averaged proportion of the number of pixels with label l compare to the total number of pixels in the body for some random images. In essence, this weight adjusts the probability of small vs. large segments. We compared to Shotton et al. (2011) just in case of segmentation, because the outputs of this approach is noisy and is not consistent with our biometrics measurements. The whole approach is implemented in C++ using the OpenCV library, and the computational time for the complete soft biometric estimation is around 50 seconds: less than 1s for nearest model finding, 2s for point sampling, 14s for alignment and 33s for label assignment.

3.1. Segmentation results

Given the results shown in Figure 7, we fixed the number of cluster to be 90 and progressively increasing the number of training images in order to test our multi-part segmentation methodology. The results in Figure 7 illustrate a low sensitivity of our approach to the number of training data: human body segmentation accuracy is improved between 10% and 20% compared to RF. On the other hand, RF trend shows that segmentation errors remain stable and is not able to improve for higher amounts of training data. We plot the qualitative results of segmentation in Figure 8, where the influence of alignment in the segmentation is shown. Segmentation errors occur in the areas for which alignment is not perfect, so SVM has incorrect labels. Another source of error comes from inconsistencies of manual labeling for different samples. The results in the image shows how the approach copes with the self occlusions.

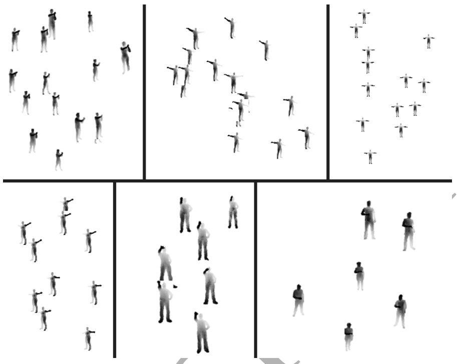

Figure 6: Some typical clusters are shown in this image among HOG and EM. The number of clusters is estimated for every combination of parameters. We employed randomly 90% of data to train 10% to test.

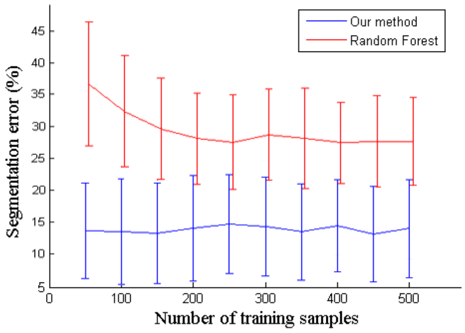

Figure 7: In our approach, segmentation error remains stable in different amount of training data in comparison to RF. Average error percentage for our method and RF at the best case is 13.55 ± 7.39 and 32.26 ± 10.03, respectively.

The similarity of the test pattern to the estimated model

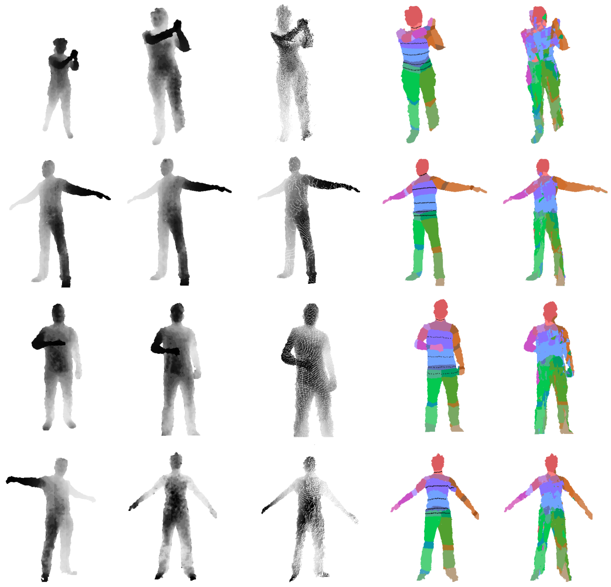

Figure 8: Qualitative results. First column shows the nearest model found, second column is the test sample, third column is the warped model after registration, and the next two columns belong to our approach and RF segmentation respectively. Black points correspond to segment lines used for measurements. It can be seen that segment lines accuracy has a direct relation with the segmentation accuracy and purity.

plays also an important role. In this case, higher number of random samples will generate better alignment and segmentation result, and higher complexity instead. Using perfect alignment parameters implies a low number of iterations, whereas a high number of iterations will reduce the accuracy dramatically in some cases. We observed that the affine behavior of λ_0 at first iteration generated quite better transformation results. We also implemented soft-assignment vs. hard-assignment which is faster but no difference in accuracy was found.

3.2. *Biometric estimation*

We show in Figure 9 the average limbs size errors among all subjects. This shows that the data distribution among all individuals is not normal and some data is more challenging for measurements. Notice that segment lines in different parts lie into the segments according to the Figure 8 even for small segments like neck.

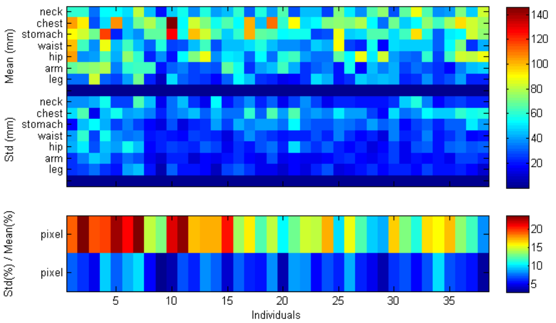

Figure 9: Overall size error per person in mm.

The accuracy of measurements is directly related to the accuracy of the segmentation and database labels: chest has the highest size errors (because clothes affect mostly on this part) whereas arm and leg have the lowest error values. Other source of errors are the affect of clothes in some poses as well as human faults in taking groundtruth. Besides self occlusions problem for example in the chest part has been solved by completing the point cloud. Table 1 summarizes the average mean and standard deviation errors in mm per limb.

Since we used geometrical approach for body measurements in this work, an extension to a multi-camera system utilizing occlusion completion would generate more accurate body base lines beside quite accurate results for full body circumference. Although we just considered one view circumference in this work, but this is the direct consequence of the results of our work.

4. Conclusions

We have introduced human segmentation as an intermediate stage for accurate soft biometric measurements applying an effective geometrical approach. As a result, an effective and accurate approach for body part size estimation from a static depth image is achieved. After clustering HOG feature vectors from depth images, the 3D shape context descriptor is used to match the points of the test pattern to the nearest estimated model. The alignment of point clouds is achieved using the TPS transformation which assigns the label of the nearest pixels. Both qualitative and quantitative results demonstrate that we improve

Table 1: The average mean and standard deviation error in mm for all the data.

	Neck	Chest	Stomach	Waist	Hip	Arm	Leg
Our method	**55.76±33.57**	**69.47±49.58**	**64.63±39.84**	**46.60±31.45**	**55.61±34.35**	**41.44±25.65**	**30.65±24.07**

human segmentation precision between 10% and 20% using a reduced set of training poses, as compared to random forest.

As a future work, we plan to parallelize and introduce our methodology in real-time scenarios, such as intelligent surveillance or size clothing estimation for e-commerce and retail purposes.

Acknowledgments

This work has been supported by the European project DiCoMa TSI-020400-2011-55; along with the Spanish projects TIN2009-14501-C02-01, TIN2009-14501-C02-02, TIN2012-39051, TIN2013-43478-P and TIN-2011-29494-C03-02.

References

Adjeroh, D., Cao, D., Piccirilli, M., Ross, A., 2010. Predictability and correlation in human metrology. IEEE International Workshop on Information Forensics and security .

Agarwal, A., Triggs, B., 2006. A local basis representation for estimating human pose from cluttered images. ACCV 1, 50–59.

Andriluka, M., Roth, S., Schiele, B., 2010. Monocular 3d pose estimation and tracking by detection.

Belongie, S., Malik, J., Puzicha, J., 2002. Shape matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 509–522.

Bo, L., Sminchisescu, C., 2010. Twin gaussian processes for structured prediction. IJCV 87, 28–52.

Bookstein, F.L., 1989. Principal warps: Thin-plate splines and the decomposition of deformations. IEEE Transactions on Pattern Analysis and Machine Intelligence 11, 567–585.

Chen, Y., Robertson, D.P., Cipolla, R., 2011. A practical system for modelling body shapes from single view measurements, in: BMVC, pp. 1–11.

Demirkus, M., Garg, K., Guler, S., 2010. Automated person categorization for video surveillance using soft biometrics. doi:`10.1117/12.851424`.

Denman, S., Bialkowski, A., Fookes, C., Sridharan, S., 2012. Identifying customer behaviour and dwell time using soft biometrics. Springer 409, 199–238.

Ferrari, V., Marin-Jimenez, M., Zisserman, A., 2009. 2d human pose estimation in tv shows.

Guo, G., Mu, G., Ricanek, K., 2010. Cross-age face recognition on a very large database: The performance versus age intervals and improvement using soft biometric traits. ICPR , 3392–3395.

Hernandez-Vela, A., Zlateva, N., Marinov, A., Reyes, M., Radeva, P., Dimov, D., Escalera, S., 2012. Human limb segmentation in depth maps based on spatio-temporal graph cuts optimization. JAISE 4, 535–546.

Jeges, E., Kispal, I., Hornak, Z., 2008. Measuring human height using calibrated cameras. Proceedings of HSI , 755–760.

Jonker, R., Volgenant, A., 1987. A shortest augmenting path algorithm for dense and sparse linear assignment problems. Computing 38, 325–340.

Kohli, P., Sun, M., Shotton, J., 2012. Conditional regression forests for human pose estimation. 2012 IEEE Conference on Computer Vision and Pattern Recognition 0, 3394–3401.

Körtgen, M., Novotni, M., Klein, R., 2003. 3d shape matching with 3d shape contexts, in: In The 7th Central European Seminar on Computer Graphics.

Laxton, B., 2007. Monocular Human Pose Estimation. Master's thesis.

Lee, Y., Lee, K.Y., Lee, J., 2006. The estimating optimal number of gaussian mixtures based on incremental k-means for speaker identification.

Mhatre, A., Palla, S., Chikkerur, S., Govindaraju, V., 2001. Efficient search and retrieval in biometric databases, spie defense and security, in: Symposium, March-2005.

Microsoft-Corp., 6/2012. Available at `http://www.xbox.com/kinect`.

Møgelmose, A., Clapés, A., Bahnsen, C., Moeslund, T., Escalera, S., 2013. Tri-modal Person Re-identification with RGB, Depth and Thermal Features. IEEE.

OpenNI, 6/2012. Available at `http://openni.org/`.

Poppe, R., 2007. Evaluating example-based pose estimation: experiments on the HumanEva sets. Technical Report TR-CTIT-07-72.

Ramanan, D., 2006. Learning to parse images of articulated bodies. NIPS , 1129–1136.

Ran, Y., Rosenbush, G., Zheng, Q., 2008. Computational approaches for real-time extraction of soft biometrics. ICPR , 1–4.

Reyes, M., Clapés, A., Ramírez, J., Revilla, J.R., Escalera, S., 2013. Automatic digital biometry analysis based on depth maps. Computers in Industry , –.

Samejima, I., Maki, K., Kagami, S., Kouchi, M., Mizoguchi, H., 2012. A body dimensions estimation method of subject from a few measurement items using kinect, in: Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on, pp. 3384–3389. doi:`10.1109/ICSMC.2012.6378315`.

Schwarz, L.A., Mkhitaryan, A., Mateus, D., Navab, N., 2011. Estimating human 3d pose from time-of-flight images based on geodesic distances and optical flow. Automatic Face and Gesture Recognition and Workshops , 700–706.

Shakhnarovich, G., Viola, P., Darrell, T., 2007. Fast pose estimation with parameter-sensitive hashing. ICCV 2, 750–757.

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A., 2011. Real-time human pose recognition in parts from single depth images.

Sminchisescu, C., Bo, L., Ionescu, C., Kanaujia, A., 2011. Feature-based pose estimation. Visual Analysis of Humans , 225–251.

Velardo, C., Dantcheva, A., D'Angelo, A., Dugelay, J.L., 2011. Bag of soft biometrics for person identification. Multimedia Tools Appl. 51, 739–777.

Weiss, A., Hirshberg, D., Black, M.J., 2011. Home 3d body scans from noisy image and range data. ICCV , 1951–1958.

Ye, M., Wang, X., Yang, R., Ren, L., Pollefeys, M., 2011. Accurate 3d pose estimation from a single depth image, in: Proceedings of the 2011 International Conference on Computer Vision, IEEE Computer Society. pp. 731–738.

Zhu, Y., Fujimura, K., 2010. A bayesian framework for human body pose tracking from depth image sequences. Sensors 10, 5280–5293.