Improving 3D Human Pose Estimation

Mirko Raca
CVLab, I&C, EPFL

***Abstract*—In attempts to improve on the existing technologies for estimating human pose we review two different approaches on estimation and one approach for representation of the human pose. We are showing the complexity of the problem and suggesting a research direction for non-invasive estimation of human posture by developing a generative model which is constrained with the time and tracking information of individual body parts which are available.**

***Index Terms*—3D pose, estimation, pedestrians, kinect, SCAPE, silhouette**

I. Introduction

HUMAN pose estimation has seen a lot of improvement over the years, but can not be considered nowhere near solved. The problem has several down-sides. The human body is typically modeled as a 55D system[1], by taking into consideration the major joints of the body. This alone gives us a huge number of poses. Many attempts were made into narrowing down the scope of the poses, either by taking into consideration the context of the action (examining only a specific range of motions)[2] or by trying to give a physical-world meaning of pose [3]. Other problem taken from the setup is that the human body is occluded not only by obstacles in the scene but by clothes and other apparel. For solving this input problem a number of motion-capturing systems have been developed and used, but these systems are typically complicated to set up, can only work in a controlled environments and reduce the mobility of the human user. The price of this capture is high and thus not applicable for real-life situations. Other specialized input devices as the Microsoft Kinect gives us a new set of input values which greatly improve the precision of the input, but are limited by the environmental factors.

Proposal submitted to committee: July 4th, 2011; Candidacy exam date: July 12th, 2011; Candidacy exam committee: Mark Pauly PhD, Pascal Fua PhD, Ronan Boulic PhD.

This research plan has been approved:

Date: ____________________

Doctoral candidate: ____________________
(name and signature)

Thesis director: ____________________
(name and signature)

Thesis co-director: ____________________
(if applicable) (name and signature)

Doct. prog. director: ____________________
(R. Urbanke) (signature)

On the other hand development of the cheap recording devices which are manufactured the distributed with everyday electronic devices gives us a large number of input sources for ad-hoc recording and using. This comes, of course, at a price of very low quality and no calibration.

As a reference for our goals we are considering a single non-invasive method which is based on regression and is based on small number of cues for the estimation[1], and yet still gives decent results. We consider the kinect-related paper[4] on pose estimation to give an example on how the novel inputs can significantly improve the state of the art. Last paper represents an attempt at giving a better visualization and reconstruction of the 3D pose[5], as an example of modeling the small deformations which make the movement believable.

The immediate challenge of the task set forth is to estimate the pose in a non-invasive way. To make the technology applicable, we're going to focus on the minimal number of input cues. Starting from the existing multi-camera system[6] we'll try to extend the functionality to include the pose estimation. When the favorable results are reached, the next steps will work on reducing the complexity of the setup by switching to a single-camera input and excluding the calibration data. Ideally, the goal is to reach a monocular, portable system for estimating 3D human pose.

II. 3D Human Pose from Silhouettes by Relevance Vector Regression

As said before, the variations of human pose represent a huge space, given that it doesn't only vary in joint angles, but also in dimensions of the subjects, colors and shape of clothing which can completely change our perception of the pose, and can easily confuse even the human eye. In search of a useful features to accurately estimate the pose, the work of A. Agarwal and B. Triggs[1] is being based on silhouettes as the input data.

The algorithm implies that we can confidently extract the silhouettes of the persons in the training and testing phases. The training phase consistency is assured by using high-contrast between the person and the background in a shadowless environment.

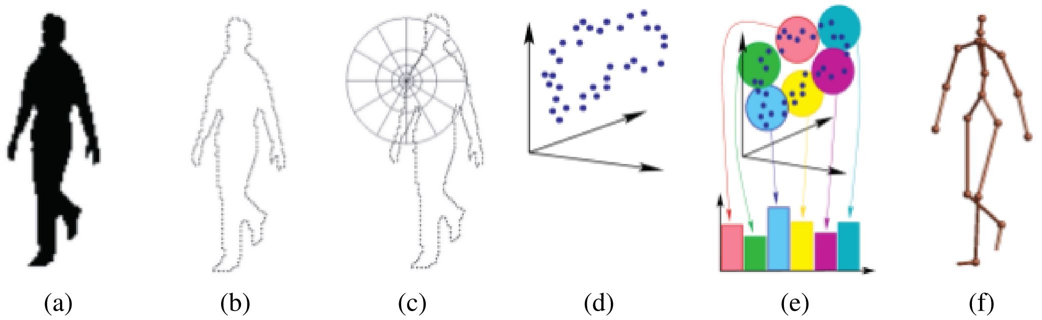


Fig. 1. Overview of the pose silhouettes method. (a) original silhouette (b) sampling points along the edge (c) log-polar bins for the classification of a single point (d) shape context distribution (e) codebook voting for 100D final distribution vector (f) final reconstruction of the pose

A. Description of the features and regression method

The useful information from the silhouette is being converted into a shape context space. The initial information is meant to robustly encode the local histogram information. The localization is achieved by encoding the information received from regularly spaced points along the edge into log-polar bins. The log-polar bins encode the local shape information into a 60-D distribution (by using the 12 angular and 5 radial bins) as seen on Fig 1c. The information is retrieved from 400-500 points, which is equivalent to a single pixel spacing in a silhouette of 64x128 pixels. The similarity of the silhouettes then becomes a problem of matching shape context distributions. To do this efficiently we construct a codebook based on 100 centers space by running the k-means algorithm over all the points in all the training silhouettes (Fig 1d). Each point's 60-D distribution votes to a few nearest codebook centers with Gaussian weights (Fig 1e). The distributions are then reduced to comparison of 100-D histograms.

The regression process is trained on a set of training examples $\{(x_i, y_i)|i = 1...n\}$ from where we get a smooth reconstruction function $y = r(x)$. Here, $y \in R^m$ represents full-body pose as a 55-dimensional vector modeling 3 joint angles at 18 major body joints and the overall pose azimuth. $x \in R^d$ is the 100-D description of the pose in the shape context space. The function is a weighted linear combination of form $r(x) \equiv \sum_k a_k \phi_k(x)$ where ϕ are given as a prespecified set of scalar basis functions $\{\phi_k(k)|k = 1...p, \phi_k : R^d \Rightarrow R^m\}$. The final model can be represented as

$$y = Af(x) + \varepsilon \equiv \sum_{k=1}^{P} a_k \phi_k(x) + \varepsilon \quad (1)$$

ε represents the residual error vector, and A is a $A \equiv (a_1 a_2 ... a_p)$ is a $m \times p$ weight matrix. The prediction error in the y-space is measured with the Euclidian norm, which gives us the estimation problem

$$A := argmin_A \{\sum_{i=1}^{n} \|Af(x_i) - y_i\|^2 + R(A)\} \quad (2)$$

The $R(A)$ represents the regularizing element for A. By putting all of our output poses in the $m \times n$ matrix $Y \equiv (y_1 y_2 ... y_n)$, features in the $p \times n$ matrix $F \equiv (f(x_1) f(x_2) ... f(x_n))$ the final estimation problem is of form

$$A := argmin_A \{\|AF - Y\|^2 + R(A)\} \quad (3)$$

Two attempts at training the model were made with *(i)* dumped least squares regression and *(ii)* relevance vector regression[7].

Due to high dimensionality of the problem solving the problem as least-squares estimation would result in over-fitting and poor generalization. The term $R(A) \equiv \lambda \|A\|^2$ is used to penalize large coefficients in matrix A, and λ is the regularization parameter. The final formula for damped least squares regressor minimizes

$$\|A\tilde{F} - \tilde{Y}\|^2 := \|AF - Y\|^2 + \lambda \|A\|^2 \quad (4)$$

in which $\tilde{F} \equiv (F \ \lambda I)$ and $\tilde{Y} \equiv (Y \ 0)$ and the solution is calculated by solving linear system $A\tilde{F} = \tilde{Y}$. λ parameter needs to be large enough to prevent over-fitting, but not too large to cause over-damping.

The relevance vector machine regression have the advantage of producing very sparse models, which are expected to model the connection between the joints of the pose and the basis functions which are related. On the other hand, RVM tend to produce a highly non-convex model with many local minima. Authors claim that the RVMs tend to give comparable results despite this. The training is carried out by approximation of the $\nu log\|a\|$ regularizer with quadratic "bridges" $\nu(\|a\|/a_{scale})^2$. Two types of priors were introduced: *(i)* component-wise priors $R(A) = \nu \sum_{jk} log|A_{jk}|$ and *(ii)* column-wise priors $R(A) = \nu \sum_k log\|a_k\|$ where a_k is the k^{th} column of A. The solution is regularized as the weight vectors $a_k \in R^m$ are well-damped, and sparse in the sense that many of them are zero. This indicates that the regression is only taking into consideration the ones which are relevant for regression. The system was tested with two types of regression bases $f(x)$ *(i)*Linear basis $f(x) \equiv x$, meant to simply return the input vector so that the RVM can select relevant features (components of x). *(ii)* Kernel bases $f(x) = (K(x, x_1) ... K(x, x_n))^T$ meant to select relevant examples.

B. Results

Given the inherent ambiguity of the input data, the method gives good results in the situations where the self-occlusion is limited. The mean estimation error over all joints for the

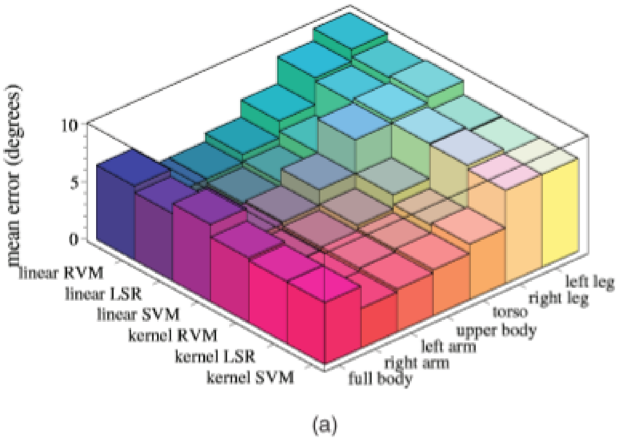


Fig. 2. (a) Graph of error in degrees per body part per regression method

Gaussian RVM is 6°. The results vary per body part depending primarily on how visible was the body part during the training phase. Comparison between different regression methods and kernel functions can be seen on Fig. 2. The authors used the opportunity to test the implicit feature selection, meaning to see if the silhouette points are indeed taken as the input for the expected joints of the body. Since the kernel methods hide this relation the test was only reproduced for linear kernels. For the purpose of this test a separate RVM regressor was trained for each of the 5 body parts - torso, two arms and two legs. While this test did show that the sampled silhouette points are grouped locally, their position did not correspond the expected results as seen on Fig. 3. This could mean that the training is putting the regression method on the a wrong set of points. Since the approximation was made only for walking motion, it remained to be determine if the similar results would arise from a training with a broader set of motions.

III. Real-Time Human Pose Recognition in Parts from Single Depth Images

In search for the new features, technologies such as PrimeSense's depth-camera made a big difference. The package was completed by using the Shotton's work previously applied to segmentation of the images[8], [9] and applying it to achieve a real-time full body pose reconstruction with good results[4].

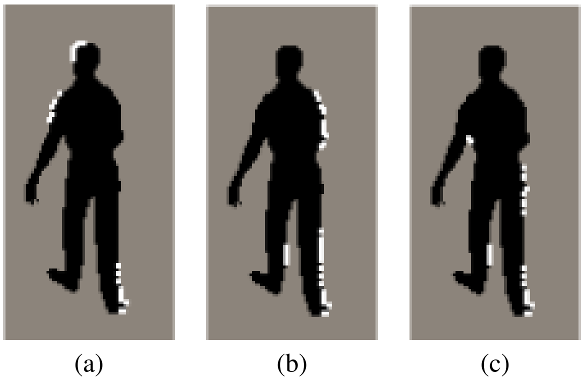


Fig. 3. Implicit feature selection. Selected silhouette points used for: (a) torso and neck (b) left arm (c) right arm

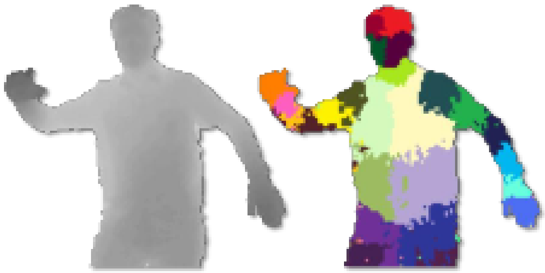


Fig. 4. Depth map and labeled body segments of the upper torso.

A. Body-part labeling and depth image features

The main contribution of the work is the substitution of the problem, by not taking into account the skeleton pose itself, but making a intermediate step and observing the human pose as an labeled depth surface. The problem is divided into two parts *(i)* finding the most probable configuration of the body segments viewed by the camera *(ii)* voting on the skeleton pose based on the found configuration.

Human body is divided into 31 segments which are marked to reflect the left/right orientation, localize important body joints or (if the body parts are not directly related to joints) to help connect the segments into whole and give a more probable voting material for the next step in processing. The segments in question are: LU/RU/LW/RW head, neck, L/R shoulder, LU/RU/LW/RW arm, L/R elbow, L/R wrist, L/R hand, LU/RU/LW/RW torso, LU/RU/LW/RW leg, L/R knee, L/R ankle, L/R foot (Left, Right, Upper, loWer). Part of the segmented depth map is shown on Fig. 4.

The depth features are based on the principles that they have to be light in terms of computation (to accommodate the real-time requirements of the Kinect platform). Each feature is calculated as

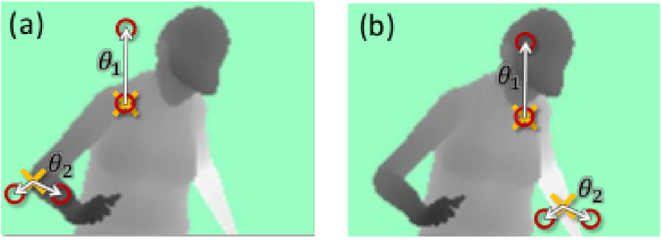
$$f_\theta(I,x) = d_I\left(x + \frac{u}{d_I(x)}\right) - d_I\left(x + \frac{v}{d_I(x)}\right) \tag{5}$$

and illustrated at Fig. 5. The randomly chosen offsets u and v from the original point x are used to retrieve the probability $P_t(c|I,x)$ from a random decision-tree forest, where c represents the class of body part. The parameter $d_I(x)$ is a normalization factor which scales the offsets depending on how far away is the pixel located from the camera.

The decision-tree forest is (for the examples given in the paper) formed out of 3 trees of 20 depth steps. The decision features at each level are selected from randomly chosen f_θ and τ parameters, where τ serves as a threshold to choose the left or right branch of the tree. Each tree is formed with the procedure:

1) Randomly select a splitting feature $\phi = (\vartheta, \tau)$
2) Split the set of training examples $Q = (I, x)$ to the left and right subsets

$$Q_l(\phi) = \{(I,x) | f_\vartheta(I,x) < \tau\}$$



(a) (b)

Fig. 5. Depth image features. Yellow cross is the pixel being classified, red circles represent the depth locations *u,v* being considered by the classifier f_θ.

$$Q_r(\phi) = Q \setminus Q_l(\phi) \tag{6}$$

3) Compute the largest information gain as

$$\phi^* = argmax_\phi G(\phi) \tag{7}$$

$$G(\phi) = H(Q) - \sum_{s\in\{l,r\}} \frac{|Q_s(\phi)|}{|Q|} H(Q_s(\phi)) \tag{8}$$

with $H(Q)$ being the Shannon's entropy, computed as the normalized histogram of body parts labels $l_I(x)$ for all $(I, x) \in Q$.

4) If we found the large enough $G(\phi^*)$, and the depth of the tree is below the maximum, recurse for the left and right subsets $Q_l(\phi^*)$ and $Q_r(\phi^*)$.

Decision trees are particularly interesting because of their easy implementation and parallelization properties. This combined with the simple decision features gives us a fast system with reliable outputs, which can be easily sped-up even further. The downsides of the method lies in the vast amount of needed training data, which will be discussed in the next sections.

B. Joint position

After the image has being processed by the decision forest, it gives us a probability of which body part each pixel in the image represents. To actually get the 3D pose of the skeleton, these conclusions need to be processed to check the join position and combine those into a proper 3D pose. Since the probabilities are calculated on per-pixel bases, the output of the previous step potentially contains a large number of outliers. This is corrected by using a local mode-finding approach. This is based on mean shift algorithm [10] with a Gaussian kernel. The density estimator per body part is defined as

$$f_c(\hat{x}) \propto \sum_{i=1}^{N} w_{ic} exp(-||\frac{\hat{x} - \hat{x}_i}{b_c}||^2) \tag{9}$$

where $\hat{x}$ is a coordinate in 3D space, N the number of image pixels, w_{ic} pixel weighting, $\hat{x}_i$ the re-projection of the image pixel x_i into world space given depth $d_I(x_i)$ and b_c is a learned per-part bandwidth. The w_{ic} considers the inferred body part probability at the pixel and the world surface area of the pixel as

$$w_{ic} = P(c|I, x_i) \cdot d_I(x_i)^2 \tag{10}$$

Mean shift finds modes in this density, and all pixels with learned probability threshold above λ_c are used as starting points for part c. The final confidence estimate is given as a sum of pixel weights reaching each mode. At this point we have an accurate estimate of the surface of the human body parts. To get the 3D position of the joint, the mode of each labeled surface is pushed back into the scene by a learned offset ζ_c.

C. Training and results evaluation

The downside of the method is the large quantity of training needed to properly prepare the decision trees for all the variations of the human shape, due to large number of poses, shapes, clothing etc. The problem was solved by applying the limited number of motion captures to a large number of generated models. The virtual mannequins have the advantage of easily being partitioned in the segmented parts, and also that they can be easily augmented to different proportions, clothes etc. The training set consisted of 5000 depth images with ground truth body parts labels and joint positions.

The training is limited in sense that there is an assumption about the context of usage. The user is directed towards the camera ±120° and the range of actions is typical for usage with the gaming console (menu choosing, driving, shooting, hitting etc.).

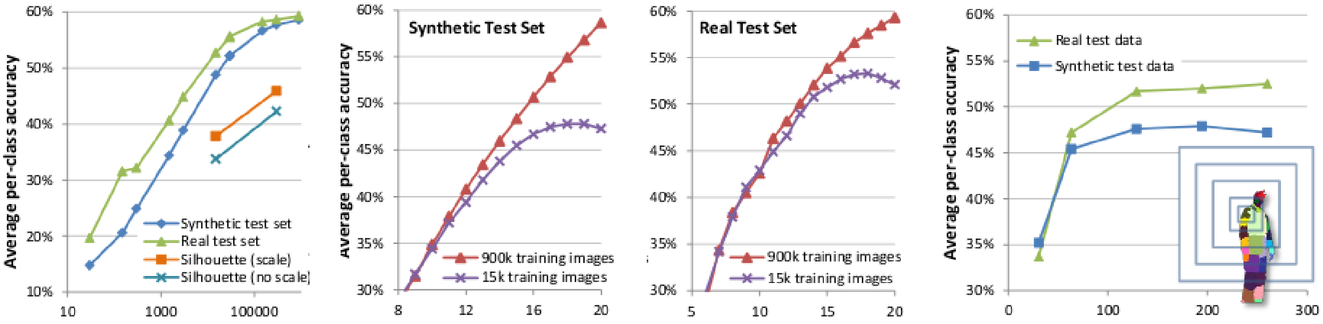
The second thing that was analyzed is the sensitivity of the algorithm to different training parameters as shown on Fig. 6. Depending on the number of training images Fig. 6a we see an approximately logarithmic growth until 100k images which could be due to the depth of the decision trees. This is assumed to be the most important parameter and is evaluated on Fig. 6b,c. Depending on the number of used training images the, we reach a quicker over-fitting of the trees (at about 17 levels of depth) with the smaller number of training images. The graph indicates that even greater precision could be reached with over 20 levels of depth, with the greater cost in memory and processing time. Maximum probe offset (how far away can the feature probes be distant from the pixel being classified) showed that it makes little difference over 129 pixel meters, meaning that the context size for classifying each pixel has it's limit and we need not take into account the entire image for a certain body part.

IV. SCAPE: Shape Completion and Animation for People

Among typical representations of the reconstructed human skeleton as a stick-figure, SCAPE model introduced an attempt of more complete and realistic visualization of the human models. Relying on precise information about the surface of the human body acquired by laser scans, SCAPE is trying to model the surface deformations based on the joint position.

A. Data acquisition and representation

The model is focused on modeling the deformations of the surface of human body. Input is based on data acquired by 3D body-scans and tries to compensate the difficulties and problems which the procedure introduces. The idea is to reproduce a human shape or body movement on the minimal



(a) (b) (c) (d)

Fig. 6. Classification dependency on different training parameters. (a) Number of training images (log scale) (b),(c) Depth of trees on synthetic and real test set (d) Maximum probe offset

number of inputs by reusing generic surface deformation model which can be adjusted based on known parts of the body scanned.

The training data is a full body scans of several subjects in different positions (Fig 7). Having the same subject in several positions allows us to observe the changes of the surface dependent on the pose of the subject. This gives us one scan which is referred as *template mash* which is aimed to be as complete as possible by means of hole-filling algorithms[11]. The rest of the poses are referred to as *instance mashes* and are brought to a correspondence with the template mash. This is done by manually placing a number of markers and then replicating them over the surface with a correlated correspondence algorithm[5]. The skeleton of the pose is automatically recovered as a 16 part structure by the algorithm[12].

The preprocessing steps play important role in producing surface mashes with a constant number of triangles and vertices. The data is organized as the instance mash $X = \{V_x, P_x\}$, where $V_x = \{x_1, ..., x_M\}$ is the set of vertices and $P_x = \{p_1, ..., p_P\}$ is the set of triangles. The scans are grouped in two ways: *(i)* same person in different poses and *(ii)* different persons in a similar pose. $Y^i = \{y_1^i, ..., y_M^i\}$ represents the instance mesh points.

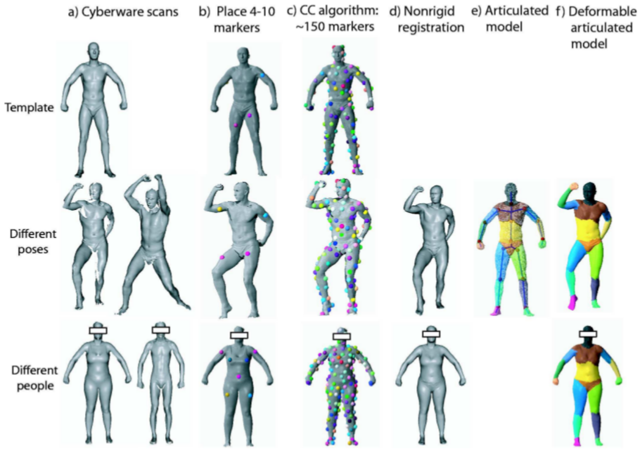


Fig. 7. Different stages of model acquisition

B. *Pose deformation*

The important idea in pose deformation is that the model is separating the rigid and non-rigid deformations of the surface and is trying to model the non-rigid deformations as a function of the angles of two nearest joints. This gives us a realistic and generic method for modeling the surface deformations.

Each triangle edge is modeled as vector relative to the first triangle point $\hat{\nu} = x_{k,j} - x_{k,1}, j = 2, 3$, where k is the triangle number. The non-rigid deformation of each triangle k in the pose i is modeled with a 3×3 matrix Q_k^i. The rigid part of the deformation is common to all triangles which are connected to the certain body part $l[k]$ and is represented as the matrix R_l^i which gives us the formula

$$\nu_{k,j}^i = R_{l[k]}^i Q_k^i \hat{\nu}_{k,j}, j = 2, 3 \tag{11}$$

To estimate the deformations needed to transform the template mesh to an instance mesh, we try to minimize the function

$$argmin_{y_1,...,y_M} \sum_k \sum_{j=2,3} ||R_{l[k]}^i Q_k^i \hat{\nu}_{k,j} - (y_{j,k} - y_{1,k})||^2 \tag{12}$$

This formulation is still under-constrained, so to predict the Q values we add a smoothness parameter between neighboring triangles

$$argmin_{y_1,...,y_M} \sum_k \sum_{j=2,3} ||R_{l[k]}^i Q_k^i \hat{\nu}_{k,j} - \nu_{k,j}^i||^2 + \\ w_s \sum_{k1,k2adj} I(l_{k_1} = l_{k_2})||Q_{k_1}^i - Q_{k_2}^i||^2 \tag{13}$$

Where $I()$ is the indicator function, $w_s = 0.001\rho$ and ρ is the resolution of model mesh X.

The representation of the joint rotations, if the joints rotation matrices are R_{l_1} and R_{l_2} we can represent the relative joint rotation as $R_{l_1}^T R_{l_2}$ and convert it to a twist angle as

$$t = \frac{||\theta||}{2\sin||\theta||} \begin{bmatrix} m_{32} - m_{23} \\ m_{13} - m_{31} \\ m_{21} - m_{12} \end{bmatrix}$$

$$\text{with } \theta = \cos\left(\frac{tr(M) - 1}{2}\right)^{-1} \tag{14}$$

The twist vector represents the axis of rotation with it's direction and the intensity of vector represents the amount

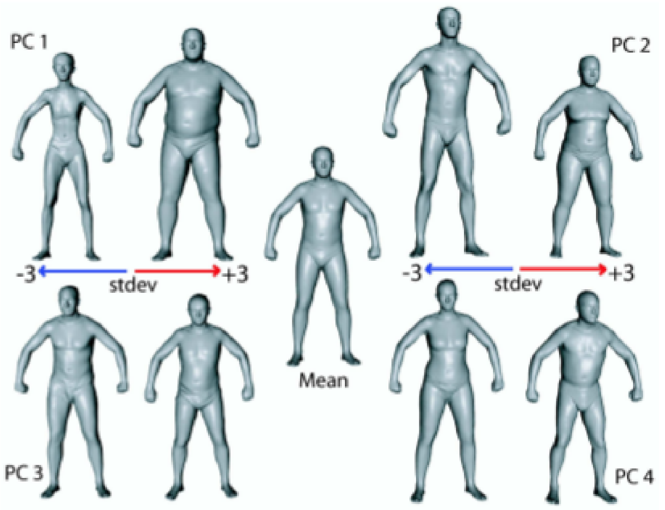


Fig. 8. First four principle components of the body-shape space.

of rotation. We use this representation of rotation matrices to model the rotations of two neighboring joints of each triangle $\Delta r^i_{l[k]} = (\Delta r^i_{l[k],1}, \Delta r^i_{l[k],2})$. To model the dependency of each element $q^i_{k,lm}$ of the matrix Q^i_k to the two twist vectors, we can represent it as

$$q^i_{k,lm} = a^T_{k,lm} \begin{bmatrix} \Delta r^i_{l[k]} \\ 1 \end{bmatrix} l,m = 1,2,3 \tag{15}$$

which gives us a 9×7 matrix of a_k elements to fit for each triangle of the mesh, which is represented as $Q^i_k = \mathcal{Q}_{a_k}(\Delta r^i_{l[k]})$. Given the joint angles from our input model, and deformation matrices Q^i_k which were constructed from the equation (13), we can find the a_k values by minimizing

$$argmin_{a_{k,lm}} \sum_i \left([\Delta r^i 1]a_{k,lm} - q^i_{k,lm}\right)^2 \tag{16}$$

It is possible to prune the model by modeling some joints with less then 3 degrees of freedom. The classification which dimension to reduce is based on the size of eigenvalues. This principle ended up reducing the model by one third of the parameters.

C. Body-shape deformation

To model the shape differences between different subjects, the model introduces another set of matrices $S^i = \{S^i_k : k = 1...P\}$ in which the i parameter indicates the different person, while the k is still representing the triangle of the surface. The new formula for deforming a triangle from the original template into a triangle of an instance model, we get

$$\nu^i_{k,j} = R^i_{l[k]} S^i_k Q^i_k \hat{\nu}_{k,j}, j = 2,3 \tag{17}$$

The S^i parameters are modeled with the idea that their origin is in the linear subspace, and thus can be modeled by the PCA

$$S^i = \mathcal{S}_{U,\mu}(\beta^i) = U\beta + \mu \tag{18}$$

For a given set of S^i_k we can estimate the PCA parameters U, μ and mesh-specific coefficient β^i. To get the S^i_k we estimate them by solving

$$argmin_{S^i} \sum_k \sum_{j=2,3} ||R^i_k S^i_k Q^i_k \hat{\nu}_{k,j} - \nu^i_{k,j}||^2 + w_s \sum_{k_1,k_2 adj} ||S^i_{k_1} - S^i_{k_2}||^2 \tag{19}$$

The R^i joint rotations are given by the preprocessing, from which we can calculate the $Q^i_k = \mathcal{Q}_{a_k}(\Delta r^i_{l[k]})$ based on the description from the previous section. This lives us with the unknown S^i_k to be estimated. $w_s \sum_{k_1,k_2 adj} ||S^i_{k_1} - S^i_{k_2}||^2$ part of the equation enforces smoothness of the adjected deformations.

Given the parameters β for body shape and R for rotation of the rigid parts, we can find vertices Y that minimize the objective

$$E_H[Y] = \sum_k \sum_{j=2,3} ||R_k \mathcal{S}_{U,\mu}(\beta) \mathcal{Q}_{a_k}(\Delta r_{l[k]}) \hat{\nu}_{j,k} - (y_{j,k} - y_{1,k})||^2 \tag{20}$$

which produces a consistent mesh of a person within the reasonable body-shape scope and pose parameters.

D. Applications

The model is applied to three tasks *(i)* shape completion *(ii)* partial view completion *(iii)* motion capture animation.

Shape completion is the task of extending the previously specified task in formula (20) with the constraints given by already scanned points. The final formulation then becomes

$$E_H[Y] + w_Z \sum_{l=1}^{L} ||y_l - z_l||^2 \tag{21}$$

where w_z represent the weight factor that balances between the fit of the markers and consistency of the model and z_l are the given coordinates of the known points of the mesh. The parameters of rotation R and body shape β are not known.

The nature of the equation (21) is non-linear, non-convex, and subjected to local minima. The routine used in the paper for effectively minimizing is to minimize each of the (R,β and Y) parameters separately. The optimization takes into consideration (in that particular order) R, Y and finally β. The iterative nature of the procedure approximates $R^{new} = (I + \hat{t})R^{old}$ where $t = (t_1, t_2, t_3)$ is the twist vector and

$$\hat{t} = \begin{pmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{pmatrix}$$

Due to an observation that the Q matrices are more affected by general R then the ΔR an added term penalizes if two adjected joints have a large difference in rotation. The final process is then described with steps:

- Update R while keeping the current values for β and Q. The final equation is

$$argmin_t \sum_k \sum_{j=2,3} ||(I + \hat{t}_{l_k})R^{old} SQ\hat{\nu}_{j,k} - (y_{j,k} - y_{1,k})||^2 + w_T \sum_{l_1,l_2 adj} ||t_{l_1} - t_{l_2}||^2 \tag{22}$$

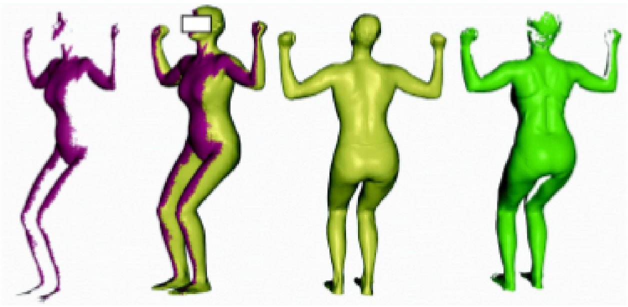


Fig. 9. Partial pose scan, pose completion and comparison between the SCAPE completion and the original pose.

after the R has been updated, we update the ΔR and Q accordingly to the new value of R.

- Update Y while β and R are fixed, by optimizing equation (21). By this, we get S and Q matrices.
- Update β by optimizing equation (21), R and Q are fixed. The objective function is reduced to

$$\sum_{k} \sum_{j=2,3} ||R_k \overline{(U\beta + \mu)}_k Q \hat{\nu}_{j,k} - (y_{j,k} - y_{1,k})||^2 \qquad (23)$$

Partial view completion (Fig 9) also relies on the equation (21). With given partial scan of the human body, the points of correspondence with the generic model can be manually set. Additional points are being produced with CC algorithm[12]. The generic SCAPE model recovered this way can be used to fill in the missing parts with generic information. The model can not recover any specifics of the person based on minor information present on the given scan part.

Motion capture animation implies that we have an generic initial scan of the person we with to animate. This provides us with the body-shape parameters. The animated sequence is archived by using these parameters with the series of marker positions which represent body pose in each frame. The algorithm uses the pose of the body from the previous frame as the starting point of the optimization for the next frame.

E. Results

The model is focused on modeling the deformations of the surface of the human body. Although this can appear to have limited usability in the commercial use by usage of naked bodies only, the dependency between non-rigid and rigid deformations is interesting in it's application to other non-rigid deformations e.g. clothing. Due to nature of cloth, this isn't a simple substitution of the model, and would require a intermediate representation which is broad enough to model the deformation of the surface and is also applicable to a number of cloth styles which could be applied on top of that.

As the authors themselves notice, the model is also disregarding further physical properties of the material which the surface is modeling. E.g. the muscle deformation are not dependent on the shape properties of the person in question, meaning that the entire surface is always presumed to represent muscle mass, which is seldom the case.

V. Current work

The work until this point has been based on existing technologies for multiple person tracking[13], [14] and facing the challenges that lie with difficult viewing conditions.

A. Shadow removal

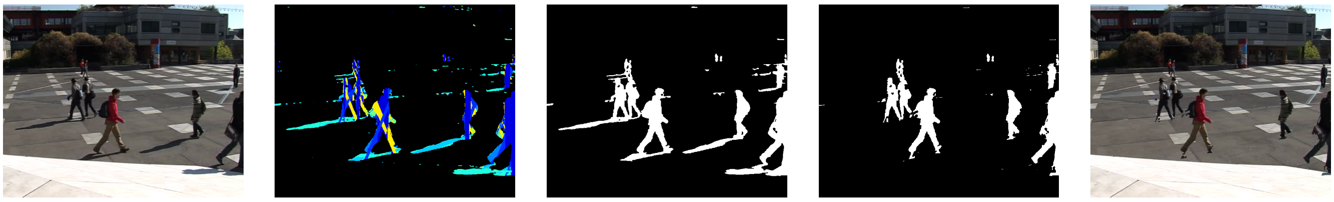
The first problem which was worked on was removal of shadows for the purpose of better tracking of pedestrians. The problem had two major difficulties *(i)* low quality of video footage *(ii)* dark cloths of the pedestrians in the footage which combined with the high contrast of the video proved to be indifferentiable from the shadows themselves. A number of techniques based on machine learning were tried. The idea of using SVM with a color-based feature vectors and background information proved to give limited results. Among tried color combinations of RGB, HSV, grayscale, with added background color information of the same pixel, feature vector with grayscale values of background and foreground actually gave the best results. Further information of the entropy, texture etc. were not useful due to the errors induced by video compression methods.

The final solution (Fig 10) was based on the concept that shadows are consistent in the color and direction. The foreground pixels (after background subtraction and smoothing preprocessing steps such as connected component size filter, erosion and dilation) were sorted into 8 bins based on the angle in which we had the biggest number of foreground pixels. By specifying the directions which are expected for the shadows, we were able to remove enough of the shadows pixels to improve the tracking of the algorithm[14]. This approach remains sensitive to large groups of individuals walking together in close proximity, because this gives us mis-classifications for certain bins. The visual results are not always visually pleasing, but they provide enough statistical information for the tracking algorithm.

B. Consistency of human silhouette

The problem of recognizing outlines of humans in a crowded scene was also worked on by attempt of building a library of common outlines per person. The tracking algorithm[14] provides us with a bounding boxes of persons in individual views (Fig 11b). By using the overlap check of the bounding boxes, we can determine when the person is isolated in the video and store those outlines to our pose library. For the problematic frames (Fig 11a), when the person is overlapping with other persons, we can apply a generative approach - by using the background model and poses of the isolated persons in the library, we can try to reconstruct the scene. The end result gives us the silhouettes of each person (Fig 11c).

Comparison is done by simple cross-correlation between the original image and generated image. Since the library is limited in size, and even the best match may not have a 100% overlap of the foreground pixels, after the best fit has been found, we apply the *region growing algorithm* meaning that the regions of assigned foreground pixels are being expanded on account of regions of un-assigned pixels.



(a) (b) (c) (d) (e)

Fig. 10. Removing shadows (a) Original frame (b) Dominant orientations of foreground pixels colored by different bins (c) Binary foreground before shadow removal (d) Binary foreground after shadow removal (e) Final frame

VI. Thesis plan

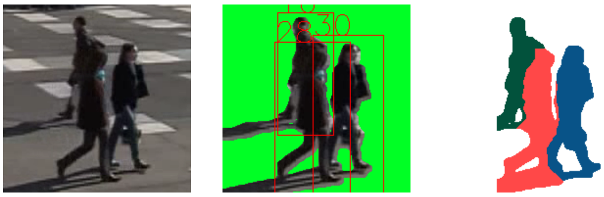
The initial proposal of the research is to work and improve on the existing detection techniques, and attempt to exclude the background subtraction methods from the process, as they prove to be unreliable when working with moving cameras or dynamically changing backgrounds. The primary focus of the research will be analysis of the outdoors scenes and sport activities as they provide a broad spectrum of motions and actions, as well as challenging situations.

The inputs, at least for the initial phase, will be based on the multi-camera setup which is currently being used by the CVLab[15]. This gives us a good comparison point for the detection/tracking algorithms, as well as reliable source of training data. The main benefits of using it as a starting point is the possibility to determine the identity, trajectory (and from that - orientation) of the person in the images.

For the estimation of the pose the focus of the research will be on using the time-domain data to constrain the pose space, which we feel was not used enough in the previous papers. This has been a trend in the recent advances in the pose estimation papers [16]. Combined with recent advances in tracking algorithms [17] we hope to constrain the problem enough to move from the multi-camera to monocular inputs and still retain enough information to process the problems from the natural situations. The immediate idea for the constraints is to detect recognizable parts of the person which such as head, feet, hands combined with typical skeletal model of the person.

Other approach being considered is the generative approach in which we would be a follow-up on the work described in the section *5b*. The idea would be to replace the *library of poses* with a generative model which would combine textural properties of the person in the footage with a SCAPE-like model in the attempt to resolve ambiguities by trying to generate what we see and explain the 3D construction of the scene (mainly the poses, since the position of the persons in the scene can be explained with the tracking information).

At this point we do not have a reason to limit the research purely to model-based or learning-based methods. The approaches will be used as seen fit, but the focus will be to remain as much in the non-controlled environment as possible.



(a) (b) (c)

Fig. 11. Human silhouette example (a) Original problematic scene before segmentation (b) Segmented frame with displayed bounding boxes and pedestrian IDs (c) Color-coded regions of different pedestrians

References

[1] A. Agarwal and B. Triggs, “3D Human Pose from Silhouettes by Relevance Vector Regression,” in *Conference on Computer Vision and Pattern Recognition*, 2004.

[2] R. Urtasun, D. Fleet, and P. Fua, “Monocular 3D Tracking of the Golf Swing,” in *Conference on Computer Vision and Pattern Recognition*, June 2005.

[3] C. K. Liu, A. Hertzmann, and Z. Popovic, “Learning Physics-based Motion Style with Nonlinear Inverse Optimization,” in *SIGGRAPH*, 2005.

[4] J. Shotton, A. Fitzgibbon, M. Cook, and A. Blake, “Real-Time Human Pose Recognition in Parts from a Single Depth Image,” in *Conference on Computer Vision and Pattern Recognition*, 2011.

[5] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, “Scape: Shape Completion and Animation of People,” *ACM SIGGRAPH*, vol. 24, pp. 408–416, 2005.

[6] J. Berclaz, F. Fleuret, and P. Fua, “Multiple Object Tracking Using Flow Linear Programming,” Idiap Research Institute, Tech. Rep., December 2009.

[7] M. Tipping, “Relevance vector machine,” *Journal of Machine Learning Research*, 2001.

[8] J. Shotton, M. Johnson, and P. Cipolla, “Semantic Texton Forests for Image Categorization and Segmentation,” in *Conference on Computer Vision and Pattern Recognition*, 2008.

[9] J. Shotton, J. Winn, C. Rother, and A. Criminisi, “Textonboost for Image Understanding: Multi-Class Object Recognition and Segmentation by Jointly Modeling Texture, Layout, and Context,” *International Journal of Computer Vision*, vol. 81, no. 1, January 2009.

[10] D. Comaniciu and P. Meer, “Mean Shift: a Robust Approach Toward Feature Space Analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[11] J. Davis, S. Marschner, M. Garr, and M. Levoy, “Filling holes in complex surfaces using volumetric diffusion,” in *Symposium on 3D Data Processing*, 2002.

[12] D. Anguelov, D. Koller, H. Pang, P. Srinivasan, and S. Thrun, “Recovering articulated object models from 3d range data,” in *20th converence of Uncertainty in artificial intelligence 18-24*, 2004.

[13] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, “Multi-Camera People Tracking With a Probabilistic Occupancy Map,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 267–282, February 2008.

[14] J. Berclaz, F. Fleuret, E. Türetken, and P. Fua, “Multiple Object Tracking Using K-Shortest Paths Optimization,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

[15] J. Berclaz, F. Fleuret, and P. Fua, “Principled Detection-By-Classification from Multiple Views,” in *International Conference on Computer Vision*, January 2008.

[16] M. Andriluka, S. Roth, and B. Schiele, “Monocular 3d pose estimation and tracking by detection,” in *CVPR*, 2010.

[17] Z. Kalal, K. Mikolajczyk, and J. Matas, “Face-tld: Tracking-learning-detection applied to faces,” in *ICIP*, 2010.