LOCALIZING BODY JOINTS FROM SINGLE DEPTH IMAGES USING GEODETIC DISTANCES AND RANDOM TREE WALK

Sebastian Handrich and Ayoub Al-Hamadi

Otto-von-Guericke-University
Magdeburg, Germany
sebastian.handrich@ovgu.de

ABSTRACT

We address the problem of human pose estimation from single depth images. We extend a previously proposed approach made by Jung and learn the direction toward skeleton joints from both depth and geodetic features. Experimental evaluation shows that our approach achieves a higher precision or a similar precision, but with much smaller regression trees.

Index Terms— Human Pose Estimation, Geodetic Distances, Random Tree Walk

1. INTRODUCTION

Human pose estimation is useful in numerous fields including human-computer interaction, gaming, surveillance, and gait analysis. Despite the constant progress that has been made over the past two decades, it is still a challenging problem since the human body is capable of an enormous range of poses. Frequently occurring self-occlusions make the problem even more difficult. Therefore, marker-based solutions are still the most preferred when accurate and reliable estimates are required.

Earlier works focused on pure 2d image based methods. Typically, silhouettes, skin color or contours were used to detect different body parts. These attempts, however, often lacked the ability to resolve ambiguities, especially in the case of self-occlusions. Therefore and due to the development of low-cost depth sensors, recent approaches focus primarily on the analysis of 3d data. In general, pose estimation methods are divided into generative and discriminative approaches. Generative approaches aim to adapt a model to the observed data. This is done by minimizing some type of energy function and typically requires the detection of corresponding points between the model and the data. An early work, the Articulated ICP, was made by Pellegrini et.al. [9]. These approaches are often prone to misdetected corresponding points and as a result can easily be trapped in local energy minima. To overcome this, a series of extensions were proposed, in which the correspondences were described in a probabilistic manner. For example, by utilizing Random Forests [2], Gaussian Mixture Models [7], Bayesian Networks [8] or in combination with Inverse Kinematics [1]. In another set of approaches, geodetic distances were used to reduce the correspondence problem. For example, in [5] individual body parts were first segmented and Baak et. al. [6] matched geodetic feature points with those in predefined pose databases. In contrast, discriminative approaches aim to derive pose parameters directly from the observed data. In his break-through paper [3], Shotton employed randomly sampled depth differences to classify the body part of each depth pixel. A new state-of-the-art approach was recently proposed by Jung [4], who employed regression trees in order to learn the direction towards a skeleton joint from local depth differences.

In this work, we contribute to this research by extending the work of Jung. Instead of using only depth differences, our approach predicts the direction towards a specific skeleton joint from learned geodetic distances. We thus follow the suggestions made by Shotton [3] to employ more complex features that contain more information per pixel. Experimental evaluation, in which we run both our and the original Random Tree Walk algorithm, shows that our approach achieves a higher precision or a similar precision, but with much smaller regression trees.

2. HUMAN POSE DETECTION

Our goal is to determine the 3d position $\mathbf{p}_j$ of each joint of a skeleton model from a depth image $\mathbf{D} = \{d_i\}_{i=1}^{n_x \times n_y}$. The skeleton consists of $N_J = 15$ joints: the head, two shoulders, elbows and wrists, two hips, knees and ankles and two spine joints. Our approach is an extension to the Random Tree Walk (RTW) approach, proposed in [4]. Similar to [4], we train a regression tree that predicts the direction towards a specific joint $\mathbf{p}_j$ from a a nearby random point. As we cannot describe the complete RTW algorithm here, we would like to refer the reader to [4] for more details, but summarize some important points:

This work was supported by Transregional Collaborative Research Centre SFB/TRR 62 (Companion-Technology for Cognitive Technical Systems) funded by the German Research Foundation (DFG).

(1) In both the original RTW algorithm and our approach, there is one regression tree for each skeleton joint. As a result, the trees are much easier to train.

(2) The leaf nodes of the regression trees store direction vectors $\mathbf{u}_k$ (not offsets) to the true joint position. As such, the RTW algorithm is an iterative approach. In each iteration m, the current joint position $\mathbf{p}_j^m$ is updated by:

$$\mathbf{p}_j^{m+1} \leftarrow \mathbf{p}_j^m + \hat{\mathbf{u}} \cdot dist_s, \text{ with } \hat{\mathbf{u}} = \frac{1}{K} \sum_1^K \mathbf{u}_k, \qquad (1)$$

where $dist_s = 2$ cm is the step size and K the number of direction vectors in the leaf node. This is repeated in each frame N_{RTW} times.

(3) The initial position of a joint $\mathbf{p}_j^{m=0}$ is defined by the hierarchical structure of the skeleton model. This means that the initial position for the left wrist is given by the predicted position of the left elbow and so on. For the spine, the point cloud center was used. A particular joint position is thus predicted from a nearby point which drastically reduces the required amount of training samples.

Whereas in [4] the regression tree training and prediction is solely based on depth image based features, we use both depth differences and geodetic distance based features. Whereas the use of simple depth differences was strongly motivated by their computational efficiency, the use of geodetic distances in our approach is motivated by their much higher information value per pixel that can be used to predict the position of a particular skeleton joint.

Geodetic distances: A geodetic distance, in the context of human pose estimation, is the length of the shortest path between two arbitrary 3d points along the surface of the human body. In order to compute them we first need to transform the depth image $\mathbf{D}$ into a 3d point cloud $\mathbf{W}$. The point cloud is further described as a weighted graph $\mathcal{G} = (\mathbf{W}, E)$. Two 3d points are connected by an edge, if their Euclidean distance is below a threshold $\epsilon_g = 2$ cm and if they correspond to adjacent pixels in the depth image (edge criterion).

This, however, only works when there are no self-occlusions. Otherwise, the graph is fragmented (Fig.1a). Connecting these fragments solely based on their Euclidean distances is very error-prone. Instead, we use a scoring function (Eq.2) as follows: In each fragment, we find the 3d points that are not fully connected, i.e. have no connection in at least one of the four 2d direction top, left, right or bottom. We divide these points into groups $\mathcal{T}$, which we call transitions. A transition contains all 3d points that are connected to each other by the edge criterion described above but have no connections in the same 2d direction. A single 3d point can belong to multiple transitions. We then find all pairs $V = \{(\mathcal{T}_i, \mathcal{T}_k)\}$ of valid transitions. A transition pair is considered valid, if a) the distance between their 3d centers does not exceed a threshold $d_{\mathcal{T}} = 30$ cm, b) they have opposing connectivity (i.e. a transition with no-top connections can only be connected to a transition with no-bottom connections and c) their relative 2d position corresponds to their connectivity (i.e. a no-left transition can only be connected to another one that is located to the left of it). This validity test can be performed very quickly and reduces the number of valid transitions to a few (typically less than 15). For each valid transition pair, the transitions are approximated by two 3d line segments (Fig. 1b) and the transition score $S_{\mathcal{T}}$ is computed as:

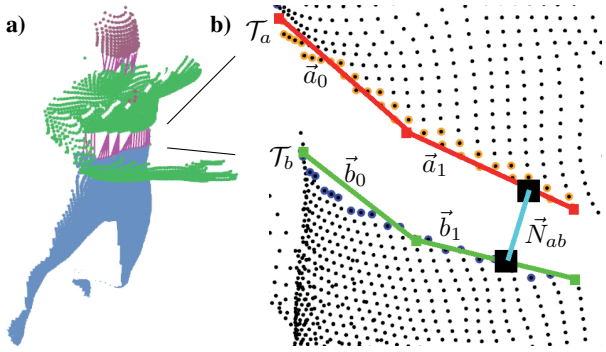

Fig. 1: **a)** Geodetic graph fragmentation in case of self-occlusions. Fragments are connected by adding extra graph edges (purple lines). **b)** Parameters used in the transition score function (Eq. 2). Each fragment border (transition) is approximated by two line segments and offset vector $\vec{N}_{ab}$ between the two closest points is computed.

$$S_{\mathcal{T}}(\mathcal{T}_i, \mathcal{T}_k) = \sum_{i=1}^{3} w(i) s_i, \text{ with } w = [1\ 0.5\ 0.5]^T, \qquad (2)$$

$$s_1 = \exp(-||\vec{N}_{ab}||^2/(2\sigma^2)), \qquad (3)$$

$$s_2 = \max(|\vec{a}_0\vec{b}_0|, |\vec{a}_1\vec{b}_1|, |\vec{a}_0\vec{b}_1|, |\vec{a}_1\vec{b}_0|), \qquad (4)$$

$$s_3 = \vec{N}_{ab}/||\vec{N}_{ab}|| \cdot \vec{N}_{con}. \qquad (5)$$

According to Eq.2, we assign a higher score to transition pairs which are close to each other (s_1), parallel to each other (s_2) and are located to each other in the expected direction $\vec{N}_{con}$ based on their connectivity (s_3). Having $N_{\mathcal{F}}$ graph fragments, we determine the $N_{\mathcal{F}} - 1$ best transition pairs and add extra graph edges between their 3d points (Fig. 1a). Having a fully connected graph, Dijkstra's algorithm is used to compute the geodetic distances. This results in a geodetic distance map $\mathbf{G}$ which contains for each 3d point its distance to the point cloud center (Fig. 2b).

Regression tree training: The generation of training samples is similar to [4]. For each joint we sample a set of random offset points $\mathbf{q}_j^i$ which are uniformly distributed around the position of joint $\mathbf{p}_j$ and its parent. In our approach, a single training sample $S = (\mathbf{D}, \mathbf{G}, \mathbf{q}_j^i, \mathbf{u}_j^i)$ consists of the depth map, the geodetic distance map, a random offset point and the direction vector towards the true joint position $\mathbf{u}_j^i =$

Fig. 2: **a)** Qualitative result of our approach for various depth images. The colored lines represent the iterativly determined paths to each skeleton joint. The starting position of a particular joint is initialized with the predicted position of its parent joint in the skeleton hierarchy. **b)** Colored representation of the geodetic distance maps $\mathbf{G}$ for the poses shown above. Red colors denote high distances and blue colors low geodetic distances from the torso center.

$(\mathbf{p}_j - \mathbf{q}_j^i)/||\mathbf{p}_j - \mathbf{q}_j^i||$. During training, the samples are recursively partitioned into left and right child nodes. At each split node, we determine the split parameter ϕ^* that minimizes the variance of the direction vectors in the left (Q_L) and right (Q_R) child nodes (Eq. 6).

$$\phi^* = argmin_\phi \sum_{u \in Q_L} ||u - \bar{u_L}||^2 + \sum_{u \in Q_R} ||u - \bar{u_R}||^2 \quad (6)$$

Two types of features ($f_{\theta 1}$, $f_{\theta 2}$) for partitioning the regression tree are used:

$$f_{\theta 1}(\mathbf{G}, \mathbf{D}, x) = \mathbf{G}\Big(x + \frac{t}{\mathbf{D}(x)}\Big), \quad (7)$$

$$f_{\theta 2}(\mathbf{D}, x) = \mathbf{D}\Big(x + \frac{t}{\mathbf{D}(x)}\Big) - \mathbf{D}(x), \quad (8)$$

where t is a random offset 2d vector to the current depth pixel position x. The first feature is used to learn the geodetic distance at a random offset position and the second feature is used to learn the depth difference between a random offset position and the current pixel position. In contrast, the features used in [4] employ the depth differences between two random offset positions (Eq. 9).

$$f_\theta(\mathbf{D}, x) = \mathbf{D}\Big(x + \frac{t_1}{\mathbf{D}(x)}\Big) - \mathbf{D}\Big(x + \frac{t_2}{\mathbf{D}(x)}\Big) \quad (9)$$

Similar to [4] we limit the minimum number of samples in a leaf node ($|Q_L|$ and $|Q_R|$) in order to avoid overfitting. Additionally, feature bagging was used: Before training, we build a large pool (N=2000) of random offset vectors t. 90% of these vectors are used for the feature $f_{\theta 1}$ and the remaining 10% for feature $f_{\theta 2}$. At each split node, however, only a small random subset ($N_S = 200$) is used for the split test.

3. EXPERIMENTAL RESULTS AND DISCUSSION

We tested our approach on our pose database which consists of $N_D = 26372$ depth images with a resolution of 320×240 pixels. For each depth image, the true joint positions $\hat{\mathbf{p}}_j$ are known. The database contains many poses with self occlusions (as depicted in Fig. 2) because these are the most challenging for our graph-defragmentation. Since consecutive poses might be similar, we used only every eighth image for training. All remaining images were used for testing. A qualitative result of our approach is shown in Fig. 2. The top row depicts the depth images and the iteratively predicted joint positions (Eq. 1, $N_{RTW} = 64$). The bottom row depicts the computed geodetic distance maps $\mathbf{G}$ for the poses shown above. In each frame, the pose estimation was completely reinitialized and no temporal information was used. Similar to [4], the predicted parent joint positions were used as

starting positions for the child joints. It can be seen how the algorithm *walks* towards the correct joint position and converges once the correct position is reached. This is even true for hidden joints like e.g. the wrists in pose 1 and 5.

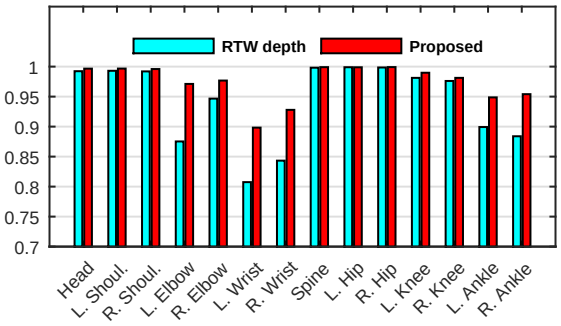

Fig. 3: Mean Average Precision (mAP) per joint for the proposed approach and that in [4]. The mAP for the wrists, elbows and ankle joints is significantly higher in our approach.

In a quantitative analysis, we compared our approach to that in [4]. We trained both theirs and our approach with the exact same training data and identical training parameters. This includes the minimum number of samples per leaf node and the offset vectors. The only difference is that we used Eq. 7 and 8 for partitioning the regression trees, whereas Eq. 9 was used for their approach.

A common precision measure in human pose estimation is the mean average precision (mAP) which is the percentage of correctly estimated joints. Typically, a joint position is considered as correctly estimated, if it is within 10 cm of the ground truth [3, 4, 7]. For trained regression trees with a minimum of 10 samples (directions vectors) per leaf node, the mAP per joint is shown in Fig. 3 for both our and the approach in [4]. We achieve a significantly higher precision for the wrists, ankles and elbows. There are two reasons for this: First, compared to depth differences, geodetic distances are a very strong feature. In fact, the geodetic distance of a particular body part to the body center is almost independent of the human pose. It is not completely independent, since geodetic distances are the length of the shortest possible path and this length may vary, especially for bent limbs. Also, particular body parts can not be detected simply by thresholding the geodetic distances, since different regions have identical distances or the body part may be hidden. A second reason is that we sample the geodetic distance or the depth difference (Eq. 7 and 8) at only one random offset position $(x + t)$, whereas in [4] two offset vectors (t_1, t_2) were used (Eq. 9). Sample values outside of the human body are typically set to a large constant positive value but this value has little information. Using only one offset position, it is less likely that the sample position is outside the human body. This is particularly true for the thinner body parts like the arms and legs and explains our increased precision for the elbow, ankles and wrist joints. Another common but in our opinion better precision measure is shown in Fig. 4. It shows the proportion of test images in which the mean joint distance to the ground truth position is below a particular distance value (averaged over all joints in single a frame). It can be seen that e.g. in about 80% of the test images the mean joint error is less than 3 cm in our approach but only in 60% for the original RTW algorithm.

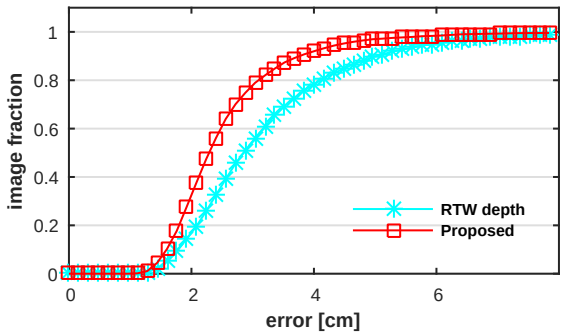

Fig. 4: Proportion (*y-axis*) of test images in which the mean joint error (averaged over all joints) is below a particular threshold (*x-axis*) for the proposed approach and that in [4].

In a final experiment, the mean precision (mAP) for different mean tree sizes was evaluated by varying the minimum number of samples per leaf node (Fig. 5). As expected, the precision decreases for smaller trees, however, not as fast as for the original RTW approach. In fact, in order to achieve a similar precision the regression trees in our approach can be about 25 times smaller! This either allows to reduce the required training time or, when using identical tree sizes, allows for the training of more complex and diverse input data. A drawback of our method is that we first have to compute the geodetic distances. Using a non-optimized implementation, we achieve 30–50 fps, depending on the tree size, the number of iteration steps and the number of graph fragments. This is still real-time capable but significantly slower than the reported 224–2697 fps in [4].

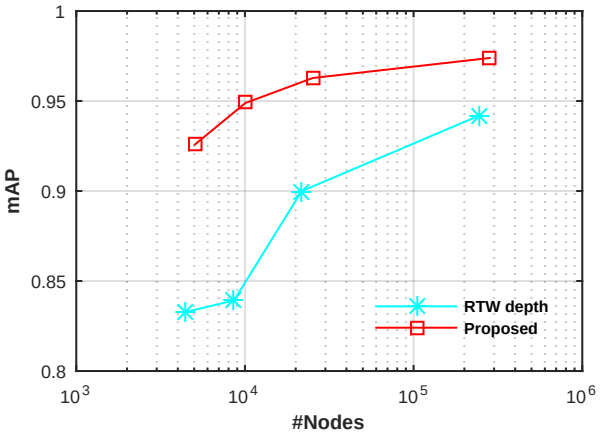

Fig. 5: Mean Average Precision (mAP) for different tree sizes for the proposed approach and that in [4]. For similar precisions, trees can be ca. $25\times$ smaller in the proposed approach.

4. REFERENCES

[1] S. Fleishman, M. Kliger, A. Lerner, G. Kutliroff. ICPIK Inverse Kinematics based Articulated-ICP. CVPR, 2015.

[2] J. Taylor, J. Shotton, T. Sharp and A. Fitzgibbon. The Vitruvian Manifold: Inferring Dense Correspondences for One-shot Human Pose Estimation. CVPR, 103–110, 2012.

[3] J. Shotton, A.F. Girshick, T.Sharp, M.Cook, M.Finocchio, R.Moore, P.Kohli, A.Criminisi, A.Kipman and A.Blake: Efficient human pose estimation from single depth images. IEEE Trans. Pattern Anal Mach Intell, vol. 35, no. 12, pp. 2821–2840, 2013.

[4] H.Y. Jung, S. Lee, Y.S. Heo and I.D.Y: Random Tree Walk toward Instantaneous 3D Human Pose Estimation. CVPR, 2467–2474, 2015

[5] L.A. Schwarz, A. Mkhitaryan, D. Mateus and N. Navab: Human skeleton tracking from depth data using geodesic distances and optical flow. Image Vision Comput., vol. 30, no. 3, pp. 217–226, 2012

[6] A.Baak, M. Mller, G. Bharaj, H.-P. Seidel, C. Theobalt: A Data-Driven Approach for Real-Time Full Body Pose Reconstruction from a Depth Camera. Consumer Depth Cameras for Comp. Vis., Springer London, 71-98, 2013

[7] M. Ye, Y. Shen, C. Du, Z. Pan, R. Yang: Real-time simultaneous pose and shape estimation for articulated objects using a single depth camera. IEEE Trans. Pattern Anal Mach Intell, vol. 38(8), 1517–32, 2014.

[8] V. Ganapathi, C. Plagemann, D. Koller and S. Thrun:Real-Time Human Pose Tracking from Range Data. Proceedings of the European Conference on Computer Vision (ECCV), 2012.

[9] S. Pellegrini, K. Schindler and D. Nardi. A Generalisation of the ICP Algorithm for Articulated Bodies. BMVC, 2008.