A two-step methodology for human pose estimation increasing the accuracy and reducing the amount of learning samples dramatically

Samir Azrour, Sébastien Piérard, Pierre Geurts, and Marc Van Droogenbroeck

INTELSIG Laboratory, Department of Electrical Engineering and Computer Science,
University of Liège, Belgium

Abstract. In this paper, we present a two-step methodology to improve existing human pose estimation methods from a single depth image. Instead of learning the direct mapping from the depth image to the 3D pose, we first estimate the orientation of the standing person seen by the camera and then use this information to dynamically select a pose estimation model suited for this particular orientation. We evaluated our method on a public dataset of realistic depth images with precise ground truth joints location. Our experiments show that our method decreases the error of a state-of-the-art pose estimation method by 30%, or reduces the size of the needed learning set by a factor larger than 10.

1 Introduction

Markerless human pose estimation is essential in a large range of applications including human-computer interaction, video surveillance, video games, virtual reality, gait analysis, rehabilitation, and intelligent houses. The vast majority of markerless human pose estimation systems are camera-based. The preference goes towards depth cameras rather than color cameras. Indeed, the latter are sensitive to illumination conditions and textures, and we need more than one color camera to avoid the problem of scale ambiguity. Since the release of the *Microsoft Kinect* camera and the associated pose estimation method [21], depth cameras have reached an affordable price and, at the same time, pose estimation has reached a new level of accuracy and robustness using only one camera.

However, current methods require a huge set of learning samples to train the pose estimation model. The reason is that pose estimation from a single depth map in an unconstrained environment (*i.e.* where the person can take any orientation with respect to the camera) is very complex. Indeed, a pose estimation method has to implicitly determine the orientation of the person while predicting the 3D location of the body joints. In previous works [14,15,21], a single machine learning model was learned to handle this task all at once which explains the need of an enormous amount of learning data to obtain a model invariant to the orientation. For instance, in [21], Shotton *et al.* used 2 billion data samples to learn their model. Despite this, their results showed that there was still a significant potential for accuracy improvement but they were limited by the size of their model. Furthermore, the noticeable difference in a depth image of a person seen from the back or facing the camera can be very small and the simple features used in [21] to describe the surroundings of a pixel seems unlikely to be efficient to disambiguate the orientation. Some global features would clearly be more appropriate for the orientation estimation task.

Following this observation, the current paper extends a previous work by Azrour *et al.* [2], that introduced the idea of using an orientation estimate to improve the accuracy of the pose estimation and validated it on synthetic data. Here, we go further by evaluating, on the public dataset UBC3V, the complete pipeline composed of an orientation estimation method followed by a multiple model pose estimation method, with each model designed for a different range of orientations. Splitting the process in two different steps allows to use adapted features and methods for each task and our results show that it requires significantly less samples to reach the same or better pose estimation accuracy.

2 Related work

Markerless human pose estimation systems estimate the pose of a human subject without the need of any device or marker attached to his body. At the expense of a loss of accuracy compared to marker-based systems, the markerless systems are simpler and not invasive which is mandatory, for instance, in sport and medical applications where the movement must be natural and not altered by some markers. Indeed, there is a risk that people could be disturbed and distracted by the markers they wear, leading to a different motion.

Markerless pose estimation systems can be classified in different categories depending on the type (color or depth), the number of cameras used, and the process used to recover the pose. The pose estimation process can be based on a body model tracking method (*generative* method), on a more direct prediction of the pose based on the input image (*discriminative* method), or on a combination of the two (*hybrid* method).

The most challenging, in terms of the accuracy of the estimated pose, is to use a unique color camera. Previous works addressing this task include poselets [4] or mixture-of-parts model [26]. More recently, convolutional networks have also been used to perform this task [23,24].

Thanks to a set of calibrated color cameras, we can avoid the problems of scale ambiguity inherent to the use of a single color camera. A great contribution was made by Corazza *et al.* [6,7]. In [6], they built upon the work of Anguelov *et al.* [1] to design a subject-specific human body model. This model was used to track the reconstructed 3D volume of the subject (or visual hull [16]) using an articulated iterative closest point (ICP) algorithm [7]. Thanks to their body model that can fit the body shape of any subject, their method is able to reach a very high accuracy. With their work, the problem of markerless human pose estimation using multiple color cameras is considered as mostly solved by some experts of the domain [22].

Using depth instead of color cameras offers many advantages for the pose estimation. Indeed, the use of depth images solves the scale ambiguity of the RGB domain and they are much more invariant to the lightening condition and the texture. Therefore, it becomes possible to create robust and sufficiently accurate systems using only one depth camera. A major contribution was proposed by Shotton, Girschick *et al.* [13,20,21] where they accurately predict in real-time the 3D positions of body joints from a single depth image, without using any temporal information. To achieve this, they used a random forest model learned from a large, realistic, and highly varied synthetic set of training images.

Body model tracking methods were also used with a single depth camera [9,10]. However, these methods are less robust (they are sensitive to local minima) and they are generally slower than discriminative methods (as [21]). Combining a body model tracking method with a discriminative method allows to recover from local minima and to benefit from the time coherence of the tracking which lead to robust and accurate systems [25].

In the specific domain of human pose estimation from a single depth image (*i.e.* where no temporal information is used) two recent contributions were proposed by Jung *et al.* [14,15]. These two methods can be seen as variants of the method proposed by Girschick *et al.* [13]. In [14], Jung *et al.* estimate the joints locations by "walking" through the depth image and predicting the direction toward a given joint at each step . The joints are recovered sequentially by taking advantage of the human skeleton structure. In this way, they significantly increased the speed compared to [13] ($1,000$ frame per second (fps) on a single CPU against 200 fps on a parallelized implementation) while being even more accurate. In [15], they proposed a two-step method where they first estimate the 3D locations of the joints and then label them. They claim that this procedure leads to a significant increase of the accuracy compared to the state-of-the-art methods. However, the dataset used to evaluate these two methods, the EVAL dataset [10], is only 10 cm accurate (mentioned by the authors and confirmed by a visual inspection of the data). Moreover, this dataset contains depth images of subjects performing the same sequence of movements where they are facing the camera most of the time. Therefore, further experimentations with a dataset containing more precise groundtruths and more diversified poses should be used to assess more reliably the accuracy of these algorithms.

Recently, a new dataset (UBC3V) was released by Shafaie *et al.* [19]. This dataset was built in a similar way as the one used by Shotton *et al.* [21]. It can be used to design pose estimation methods using up to 3 depth cameras. The markerless pose estimation domain was really lacking such a publicly available dataset. Therefore, it is an important contribution that allows to compare new methods on a same reliable basis. In [19], Shafaie *et al.* also proposed a multiple depth cameras pose estimation method. In each camera, they used a convolutional network to predict a body part label for each pixel following [20]. Then, they merged the results from all the cameras in a common 3D space and estimated the joints locations based on the 3D point cloud. Their method outperforms state-of-the-art pose estimation methods using multiple depth cameras.

3 Our method

3.1 Leveraging an orientation estimate to improve pose estimation

The orientation θ of a subject is defined as the orientation of his pelvis. Here, we arbitrarily selected $\theta = 0°$ when the subject is seen from his right side and $\theta = 90°$ when the subject is seen from his back, as shown in Figure 1.

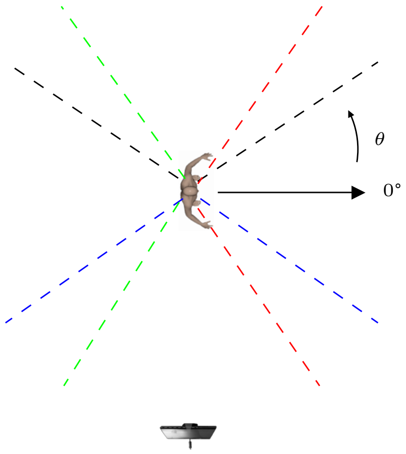

Fig. 1. Orientation convention and ranges of orientations used for the four-model method in the experiments. Each model is learned using poses with orientations included in a sector of 110° and consecutive sectors have an overlap of 20°.

To take into account the orientation estimate in the pose estimation process, we follow what was done in our previous work [2]. Instead of learning a single model $\mathcal{M}$ to deal with the whole range of orientation $O = [0°, 360°]$, we learn n models $\mathcal{M}_i$ designed for different ordered ranges of orientation O_i, such that:

$$O_1 \cup ... \cup O_n = [0°, 360°]. \tag{1}$$

Consecutive ranges partly overlap to take into account the uncertainty in the orientation estimate:

$$O_i \cap O_j \neq \emptyset, \quad \forall i, j \; : \; i = j \pm 1 \,(\mathrm{mod}\, n), \tag{2}$$

$$O_i \cap O_j = \emptyset, \quad \forall i, j \; : \; i \neq j \pm 1 \,(\mathrm{mod}\, n). \tag{3}$$

Given an overlap of l degrees between all the consecutive orientation ranges, the appropriate pose estimation model is dynamically selected at test time based on an orientation estimate $\hat{\theta}$:

$$\mathcal{M}(\hat{\theta}) = \mathcal{M}_i \;\text{ if }\; \hat{\theta} \subseteq \left[\text{lower bound}(O_i) + \frac{l}{2},\ \text{upper bound}(O_i) - \frac{l}{2}\right]. \tag{4}$$

It follows from Equation (4) that a correct pose estimation model will always be selected if the error on the orientation estimate is bounded by $\frac{l}{2}$ degrees.

Separating the orientation estimation from the pose estimation has two main advantages: (1) it allows to use different and more appropriate features for the orientation estimation, and (2) it greatly facilitates the task of the pose estimation model which allows to reach higher accuracy with less training data.

This general methodology can be used with various orientation and pose estimation methods. For the purpose of the experiments, we selected state-of-the-art methods for these two tasks. These methods are explained below.

3.2 Orientation estimation

The orientation estimation method used in this paper is based on the one proposed by Piérard *et al.* in [17].

To describe each depth image, 5 binary silhouettes are extracted from the depth silhouette of the subject seen by the camera. These binary silhouettes are obtained by thresholding the depthmap. Each of them is described using one shape context descriptor [3] with 5 radial bins and sectors of 30 degrees. The shape context descriptor used is centered on the gravity center of the binary silhouette and is populated by all external and internal contours.

The machine learning algorithm used to predict the orientation is the *ExtRa-Trees* [12]. As this algorithm uses an averaging to produce the outputs, we must avoid to directly predict the orientation angle θ because the poses with $\theta \simeq 0°$ and $\theta \simeq 360°$ are very close which would lead to random orientation estimates. Therefore, we separately estimate $\cos\theta$ and $\sin\theta$ to avoid this discontinuity. Then, the orientation estimate $\hat{\theta}$ can be obtained as follows:

$$\hat{\theta} = \tan^{-1}\left(\frac{\widehat{\sin\theta}}{\widehat{\cos\theta}}\right). \tag{5}$$

3.3 Pose estimation

We used our own implementation of the state-of-the-art pose estimation method proposed by Girshick *et al.* [13] which was thoroughly evaluated in [21] and commercially used with the Microsoft *Kinect* camera. This method uses regression random forests to predict the 3D location of the body joints directly from the raw depth image.

To accurately implement [13], we used the code library *Sherwood* [18] which is a general purpose, object-oriented software framework for solving decision forest inference problems [8]. This framework offers a great flexibility and can be adapted to a wide range of inference problems. This is achieved thanks to interfaces that have to be implemented by the user and that define how the framework interacts with the data.

A forest is an ensemble of decision trees, each composed of split and leaf nodes. In our implementation, we trained one forest for each body joint j separately. The goal of the regression forest, for a given joint j and a given pixel $\mathbf{p}$, is to estimate the offset $\triangle_{p \rightarrow j}$ going from the pixel 3D location $\mathbf{x}(\mathbf{p}) = (x(\mathbf{p}), y(\mathbf{p}), z(\mathbf{p}))$ to the 3D location of the body joint j. At test time, this is performed for all the silhouette's pixels and the predicted offsets are added to the pixel 3D locations to produce a set of estimates for the 3D location of the body joint j. The final proposal is given by the first mode (obtained with the *mean shift* algorithm) of the distribution of these estimates.

The learning data used to train the tree structure include a set of depth images, the 3D locations of the body joints in these images, and a set of pixels $\boldsymbol{S} = \{\mathbf{p}_i\}$ sampled from these images. The features used to describe the surroundings of a pixel are simple depth comparisons [20]. For a given pixel $\mathbf{p} = (x, y)$ and two 2D offsets $\boldsymbol{\Delta} = (\boldsymbol{\delta}_1, \boldsymbol{\delta}_2)$, the feature f is defined as:

$$f(\mathbf{p}, \boldsymbol{\Delta}) = z\left(\mathbf{p} + \frac{\boldsymbol{\delta}_1}{z(\mathbf{p})}\right) - z\left(\mathbf{p} + \frac{\boldsymbol{\delta}_2}{z(\mathbf{p})}\right), \tag{6}$$

where $z(\mathbf{p})$ is the depth at pixel $\mathbf{p}$. The offsets are divided by the depth at the pixel $\mathbf{p}$ to achieve depth translation invariance.

At each split node, multiple 2D offsets $\boldsymbol{\Delta}$ and thresholds τ are randomly sampled from uniform distributions. Each pair $\boldsymbol{\phi} = (\boldsymbol{\Delta}_j, \tau_j)$ induces a partition of the set of pixels $\boldsymbol{S}$ at the parent node into left and right subsets $\boldsymbol{S}_L(\boldsymbol{\phi})$ and $\boldsymbol{S}_R(\boldsymbol{\phi})$ such as:

$$\boldsymbol{S}_L(\boldsymbol{\phi}) = \{\mathbf{p}_i \mid f(\mathbf{p}_i, \boldsymbol{\Delta}_j) < \tau_j\}, \tag{7}$$

$$\boldsymbol{S}_R(\boldsymbol{\phi}) = \boldsymbol{S} \setminus \boldsymbol{S}_L(\boldsymbol{\phi}). \tag{8}$$

The best $\boldsymbol{\phi}$ is selected according to :

$$\boldsymbol{\phi}^* = \underset{\boldsymbol{\phi}}{\operatorname{argmin}} \quad \frac{|\boldsymbol{S}_L(\boldsymbol{\phi})|}{|\boldsymbol{S}|}\, I(\boldsymbol{S}_L(\boldsymbol{\phi})) + \frac{|\boldsymbol{S}_R(\boldsymbol{\phi})|}{|\boldsymbol{S}|}\, I(\boldsymbol{S}_R(\boldsymbol{\phi})), \tag{9}$$

where $|.|$ denotes the cardinality, and the objective function $I(\boldsymbol{S})$ is computed as the variance of the offsets going from each pixel in the set $\boldsymbol{S}$ to the body joint j.

In each leaf, based on the pixels that reached this leaf during training, we compute and store the mean offset.

4 Experiments

4.1 Dataset

To learn and test our proposed method, we used the dataset UBC3V introduced by Shafaei *et al.* [19]. This dataset is composed of synthetic depth images of realistic human characters with various poses for training and evaluation of single or multiview depth-based pose estimation methods. The data generation procedure was very close to the one used by Shotton *et al.* [21]. The human characters were generated using the free and open source sofware *MakeHuman* and the poses were taken from the CMU motion capture database [5]. Such a public dataset with realistic and various poses and precise groundtruths was really something missing in the markerless pose estimation community. Hopefully, this will allow to perform a reliable evaluation of the existing and future methods and to compare their performances on a same basis.

Before that, the EVAL dataset [10] was frequently used to evaluate pose estimation methods [14,15,27]. However, in the EVAL dataset, the human models are facing the camera most of the time and performing the same sequence of movements. Moreover, according to the author of the dataset [10,11] and confirmed by visual inspection, the precision of the groundtruth 3D joints locations is about 10 cm. That is a problem to evaluate realiably the accuracy of pose estimation methods. The new synthetic dataset UBC3V fixes these problems by covering the whole orientation range and by providing precise groundtruths.

The UBC3V dataset consists of three sub-datasets with varying complexity. The first one (*easy-pose*) contains one human character in various standing pose. The second one (*inter-pose*) still contains only one human character but with a more varied set of poses, like, for instance, upside-down poses, lying poses, etc. Finally, the third and last dataset (*hard-pose*) includes several human characters with different physical shapes. For each instance of a human character and a pose, three cameras are randomly placed in a virtual scene and depth images are rendered from these viewpoints. Moreover, each of the three mentioned datasets includes *train*, *validation,* and *test* sets with mutually disjoint poses.

In this paper, we only used the first dataset (*easy-pose*) because it is the only one that is compatible with a meaningful definition of the orientation. Indeed, for all the poses where the human character is not in a standing position, it is not possible to reliably define an orientation. Nonetheless, this is not a big limitation for our method because, in most applications, the human subject is in a standing position.

4.2 Orientation estimation

First, we learned and evaluated the orientation estimation method described in Section 3.2 on the new dataset UBC3V. The *easy-pose* dataset was used both to learn and test the two orientation estimation models predicting respectively the cosine and the sine of the orientation angle.

The error $\varDelta\theta$ is defined as the smallest rotation between the true orientation θ and the estimated orientation $\hat{\theta}$. Hence, $\varDelta\theta$ can be derived from the dot product between $(\cos\theta,\ \sin\theta)$ and $(\cos\hat{\theta},\ \sin\hat{\theta})$:

$$\cos(\varDelta\theta) = (\cos\theta,\ \sin\theta) \bullet (\cos\hat{\theta},\ \sin\hat{\theta}) = \cos(\hat{\theta} - \theta), \tag{10}$$

and therefore

$$\varDelta\theta = \cos^{-1}(\cos(\hat{\theta} - \theta)). \tag{11}$$

We used $100k$ images from the *train* set to learn the *ExtRaTrees* model with 100 trees and default parameters [12]. The learned model was then evaluated on $10k$ images from the *test* set. For each image, the orientation estimate was obtained using Equation (5). Then, Equations (10) and (11) were used to determine the error $\varDelta\theta$ on each image. The mean error $\overline{\varDelta\theta}$ on the $10k$ test images was $6.2°$, which is consistent with the result reported in [17]. It is worth mentioning that, with the learned models, an error of more than 20° is made on less than 5% of the poses in the *test* set. This may lead, in rare cases, to the choice of a wrong pose estimation model if we work on single depth images with no use of temporal information. However, in practice, as mentioned in [17], a light temporal filtering can be used to avoid these rare bigger errors. Such a filtering has not been implemented in this work as we aim at estimating the pose from single images.

4.3 Pose estimation without an orientation knowledge

To have a basis of comparison for our approach, we learned and evaluated the pose estimation method without using an estimation of the orientation of the person seen by the camera. In this case, for each considered body joint, a single model has to deal with the whole range of orientations $[0°, 360°]$. The machine learning model has to implicitly learn the small clues related to the orientation in the depth image while predicting the pose at the same time. It is quite a hard task when the seen person can take any orientation. As we will see in the next section, an orientation knowledge can substantially ease this task.

To build the learning set, given the available computational resources, $20k$ images were picked from the *train* set of the *easy-pose* dataset and $1k$ pixels were randomly sampled from each of them. Then, a forest of three trees (as in [21]) was learned for each body joint separately. Finally, we evaluated the learned models on $10k$ test images.

We analyzed 8 body joints: head, shoulder, elbow, wrist, hip, knee, and ankle. For each considered body joint, the accuracy of the corresponding model is given by the mean Euclidean distance error of the predictions with respect to the groundtruth 3D body joint locations. The results are given in the first column of the Table 1.

Table 1. Mean Euclidean distance errors on the positions of the considered body joints for different numbers of models and learning dataset sizes. We can see that even with ten times less training data (*i.e.* 500 images for each of the four models instead of $20k$ for a single model), the four-model method outperforms the one-model method for almost all the body joints.

	amount of models:	1	4	4
	learning images per model:	$20,000$	$^{20,000}/_{4} = 5,000$	500
	range of each model:	360°	$^{360°}/_{4} + 2 \times 10° = 110°$	$^{360°}/_{4} + 2 \times 10° = 110°$
mean error	head	2.6 cm	2.3 cm (- 11.5 %)	2.6 cm (- 0 %)
	shoulder	3.7 cm	2.9 cm (- 21.6 %)	3.2 cm (- 13.5 %)
	elbow	8.9 cm	6.3 cm (- 29.2 %)	7.8 cm (- 12.4 %)
	wrist	13.9 cm	10.4 cm (- 25.2 %)	13.2 cm (- 5.0 %)
	hip	3.4 cm	2.5 cm (- 26.5 %)	3.1 cm (- 8.8 %)
	knee	6.3 cm	4.6 cm (- 27.0 %)	6.4 cm (+ 1.6 %)
	ankle	11.2 cm	8.1 cm (- 27.7 %)	10.8 cm (- 3.6 %)
	mean	7.1 cm	5.3 cm (- 25.4 %)	6.7 cm (- 5.6 %)

4.4 Pose estimation with an orientation knowledge

To leverage an orientation estimate in the pose estimation process, we used four models instead of a single one with each model designed for orientations included in sectors of 110°. To take into account the small uncertainty in the orientation estimate, the sectors of two consecutive models overlap by 20°. Figure 1 illustrates the four chosen sectors and the orientation convention used in the experiments. The choice of four models was based on the results obtained in [2] which showed that it was a good trade-off between, on the one side, the accuracy improvement that is obtained from the shrinkage of the orientation ranges when we increase the number of models and the accuracy deterioration due to the reduction of the learning set size per model.

With the selected orientation ranges, when the orientation estimate doesn't fall in the overlap of two sectors, a correct pose estimation model is always selected (using Equation (4)) if the error on the orientation estimate is smaller than 20°. However, when the orientation estimate falls in the overlap of two sectors, the risk of selecting a wrong pose estimation model is higher. Here, we can ensure that a correct pose estimation model will be selected for any orientation estimate if the error on the orientation estimate is bounded by 10° which is equal to half of the overlap.

To make a fair comparison with the one-model method, we used the same total amount of data, *i.e.* a total of $20k$ images with $5k$ images for each of the four models. The results on $10k$ test images are given in the second column of Table 1. We can see a significant accuracy improvement for all the body joints with respect to the one-model method.

Figure 2 illustrates the influence of the learning dataset size on the mean euclidean error for the most challenging joint (wrist). The three curves correspond, respectively from to bottom, to the one-model method, the four-model method when we know the exact orientation, and the four-model method when we use an orientation estimate. As noticed above, the accuracy improvement for a constant learning dataset size is significant when we use an orientation estimate. We also see that the accuracy obtained with the orientation estimate is a little bit worse than the one obtained knowing the exact orientation. This means that a wrong pose estimation model is selected for some poses in the test set. This difference can be reduced by a using bigger overlaps between consecutive sectors at the expense of a loss of accuracy.

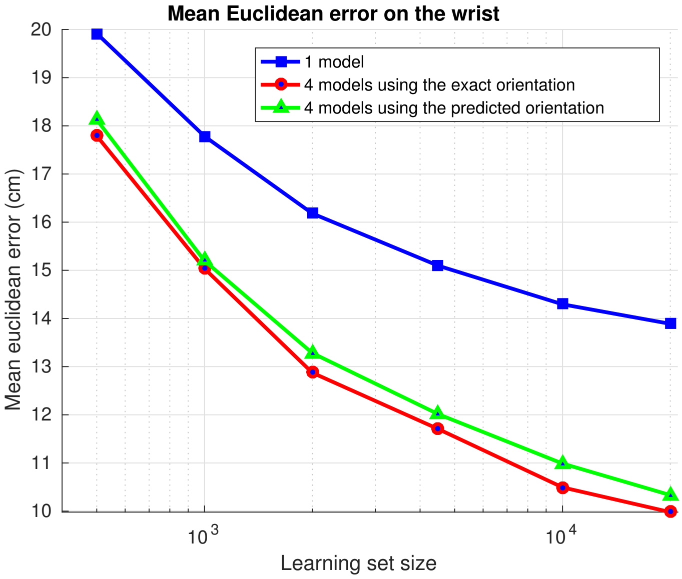

Fig. 2. Mean Euclidean error on the wrist for the one-model method (blue curve), the four-model method using the exact orientation (red curve) and the four-model method using the predicted orientation (green curve).

Furthermore, it is very interesting to note that the four-model method using only $2k$ images (500 images per model) performs better than the one-model method using $20k$ images. The last column of Table 1 gives the results obtained for all the joints using only 500 images to learn each of the four models. We observe that, thanks to an orientation estimate, we can reduce the learning dataset size by a factor larger than 10 and still outperform the one-model method for most of the joints which allows to save a precious amount of time during the learning phase.

5 Conclusion

In this paper, we evaluated a novel two-step markerless pose estimation method on the public pose estimation dataset UBC3V. The first step of this method consists in estimating the orientation of the person seen by the camera. Then, this orientation estimate is used to dynamically choose an appropriate pose estimation model. Our results show that this methodology significantly improves the accuracy when a constant learning set size is used. Moreover, we show that using an orientation estimate allows to reach a better accuracy even when the learning set size is reduced by a factor greater than 10.

Acknowledgments

Samir Azrour is supported by a research fellowship of the Belgian National Fund for Scientific Research (www.frs-fnrs.be).

References

1. D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. SCAPE: shape completion and animation of people. *ACM Trans. on Graph.*, 24(3):408–416, July 2005.
2. S. Azrour, S. Piérard, and M. Van Droogenbroeck. Leveraging orientation knowledge to enhance human pose estimation methods. In *Articulated Motion and Deformable Objects AMDO*, volume 9756 of *Lecture Notes Comp. Sci.*, pages 81–87, Palma de Mallorca, Spain, 2016. Springer.
3. S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.
4. L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3D human pose annotations. In *IEEE Int. Conf. Comput. Vision (ICCV)*, pages 1365–1372, Kyoto, September-October 2009.
5. Carnegie Mellon University. Motion capture database. http://mocap.cs.cmu.edu.
6. S. Corazza, E. Gambaretto, L. Mundermann, and T. Andriacchi. Automatic generation of a subject-specific model for accurate markerless motion capture and biomechanical applications. *IEEE Trans. Biomedical Engineering*, 57(4):806–812, Apr. 2010.
7. S. Corazza, L. Mündermann, E. Gambaretto, G. Ferrigno, and T. Andriacchi. Markerless motion capture through visual hull, articulated ICP and subject specific model generation. *Int. J. Comp. Vision*, 87(1-2):156–169, Mar. 2010.
8. A. Criminisi and J. Shotton, editors. *Decision Forests for Computer Vision and Medical Image Analysis.* Springer, 2013.
9. J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H.-P. Seidel. Motion capture using joint skeleton tracking and surface estimation. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 1746–1753, 2009.
10. V. Ganapathi, C. Plagemann, D. Koller, and S. Thrun. Real-time human pose tracking from range data. In *Eur. Conf. Comput. Vision (ECCV)*, pages 738–751, 2012.

11. V. Ganapathi, C. Plagemann, S. Thrun, and D. Koller. Real time motion capture using a single time-of-flight camera. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 755–762, San Francisco, CA, USA, June 2010.
12. P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Mach. Learn.*, 63(1):3–42, Apr. 2006.
13. R. Girshick, J. Shotton, P. Kohli, A. Criminisi, and A. Fitzgibbon. Efficient regression of general-activity human poses from depth images. In *IEEE Int. Conf. Comput. Vision (ICCV)*, pages 415–422, Barcelona, Spain, Nov. 2011.
14. H. Jung, S. Lee, Y. Heo, and I. Yun. Random tree walk toward instantaneous 3D human pose estimation. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 2467–2474, June 2015.
15. H. Jung, Y. Suh, G. Moon, and K. Lee. A sequential approach to 3D human pose estimation: Separation of localization and identification of body joints. In *Eur. Conf. Comput. Vision (ECCV)*, pages 747–761, Amsterdam, The Netherlands, Oct. 2016.
16. A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16(2):150–162, Feb. 1994.
17. S. Piérard, D. Leroy, J.-F. Hansen, and M. Van Droogenbroeck. Estimation of human orientation in images captured with a range camera. In *Advanced Concepts for Intell. Vision Syst. (ACIVS)*, volume 6915 of *Lecture Notes Comp. Sci.*, pages 519–530. Springer, 2011.
18. D. Roberston, J. Shotton, and T. Sharp. The sherwood software library. In Criminisi and Shotton [8], chapter 22, pages 333–342.
19. A. Shafaei and J. Little. Real-time human motion capture with multiple depth cameras. In *Conference on Computer and Robot Vision (CRV)*, pages 24–31, 2016.
20. J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 1297–1304, Providence, RI, USA, June 2011.
21. J. Shotton, R. Girshick, A. Fitzgibbon, 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.*, 35(12):2821–2840, Dec. 2013.
22. L. Sigal and M. Black. Guest editorial: State of the art in image- and video-based human pose and motion estimation. *Int. J. Comp. Vision*, 87(1):1–3, Mar. 2010.
23. J. Tompson, A. Jain, Y. Lecun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *Adv. in Neural Inform. Process. Syst. (NIPS)*, pages 1799–1807. Curran Associates, Inc., 2014.
24. A. Toshev and C. Szegedy. DeepPose: Human pose estimation via deep neural networks. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 1653–1660, Washington, DC, USA, 2014.
25. X. Wei, P. Zhang, and J. Chai. Accurate realtime full-body motion capture using a single depth camera. *ACM Trans. on Graph.*, 31(6):188.1–188.12, Nov. 2012.
26. Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 1385–1392, Washington, DC, USA, 2011.
27. M. Ye and R. Yang. Real-time simultaneous pose and shape estimation for articulated objects using a single depth camera. In *IEEE Int. Conf. Comput. Vision and Pattern Recogn. (CVPR)*, pages 2345–2352, June 2014.