Real-Time Upper-Body Human Pose Estimation Using a Depth Camera

Himanshu Prakash Jain[1], Anbumani Subramanian[2], Sukhendu Das[1], and Anurag Mittal[1]

[1] Indian Institute of Technology Madras, India
[2] HP Labs India, Bangalore

Abstract. Automatic detection and pose estimation of humans is an important task in Human-Computer Interaction (HCI), user interaction and event analysis. This paper presents a model based approach for detecting and estimating human pose by fusing depth and RGB color data from monocular view. The proposed system uses Haar cascade based detection and template matching to perform tracking of the most reliably detectable parts namely, head and torso. A stick figure model is used to represent the detected body parts. The fitting is then performed independently for each limb, using the weighted distance transform map. The fact that each limb is fitted independently speeds-up the fitting process and makes it robust, avoiding the combinatorial complexity problems that are common with these types of methods. The output is a stick figure model consistent with the pose of the person in the given input image. The algorithm works in real-time and is fully automatic and can detect multiple non-intersecting people.

1 Introduction

Motion capture for humans is an active research topic in the areas of computer vision and multimedia. It has many applications ranging from computer animation and virtual reality to human motion analysis and human-computer interaction (HCI) [2] [18]. The skeleton fitting process may be performed automatically or manually, as well as intrusively or non-intrusively. Intrusive manners include, for example, imposing optical markers on the subject [11] while non-automatic method could involve manual interaction to set the joints on the image, such as in [4]. These methods are usually expensive, obtrusive, and not suitable for surveillance or HCI purposes. Recently, due to the advances on imaging hardware and computer vision algorithms, markerless motion capture using a camera system has attracted the attention of many researchers. One of the commercial solutions for markerless motion capture includes Microsoft's Kinect system [17] for console systems. Kolb et al. [14] gives an account of recent developments in Time-of-Flight (ToF) technology and discusses the current state of the integration of this technology into various vision and graphics-related applications.

Since the application domain is less restrictive with only a monocular view, human pose estimation from monocular image captures has become an emerging

A. Gagalowicz and W. Philips (Eds.): MIRAGE 2011, LNCS 6930, pp. 227–238, 2011.

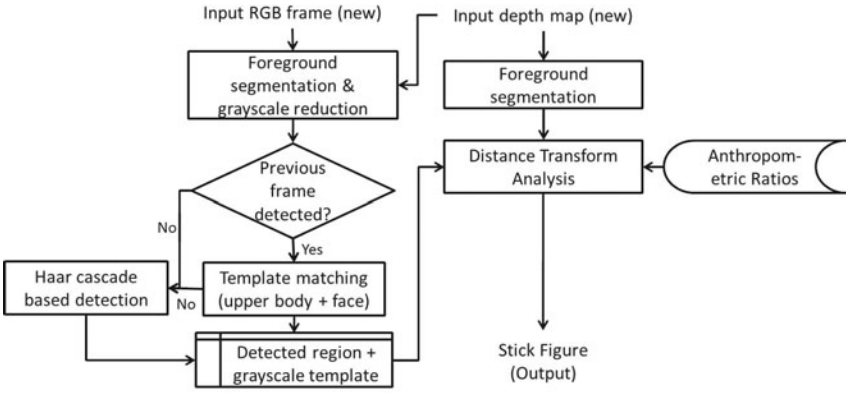

Fig. 1. Flowchart of the proposed system for upper-body human pose estimation

issue to be properly addressed. Haritaoglu et al. [10] tries to find the pose of a human subject in an automatic and non-intrusive manner. It uses geometrical features to divide the blob and determine the different extremities (head, hands and feet). Similarly, Fujiyoshi and Lipton [8] have no model but rather determine the extremities of the blob with respect to the centroid and assume that these points represent the head, hands and feet. Guo et al. [9] attempts to find the exact positions of all body joints (like the neck, shoulder, elbow, etc.) by minimizing the distance based criterion function on the skeletonized foreground object to fit the stick model. Neural networks [19] and genetic algorithms [22] have also been used to obtain the complete position of all the joints of the person. Jensen et al. [12] tries to estimate the pose based on an articulated model, for gait analysis using calibrated ToF camera.

The simplest representation of a human body is the stick figure, which consists of line segments linked by joints. The motion of joints provides the key to motion estimation and recognition of the whole figure. This concept was initially considered by Johansson [13], who marked joints as moving light displays (MLD). Along this vein, Rashid [20] attempted to recover a connected human structure with projected MLD by assuming that points belonging to the same object have higher correlations in projected positions and velocities.

The organization of the paper is as follows: Section 2 discusses the proposed approach with subsections giving details about each module used in the system. Section 3 extends the discussion towards the implementation details about the proposed prototype. Finally, Section 4 concludes the paper and delineates possible directions for future research.

2 Overview of the Entire System

In this work, we assume that a depth-camera is static and is positioned at human height. It is also assumed that users' interaction spaces are non-intersecting and

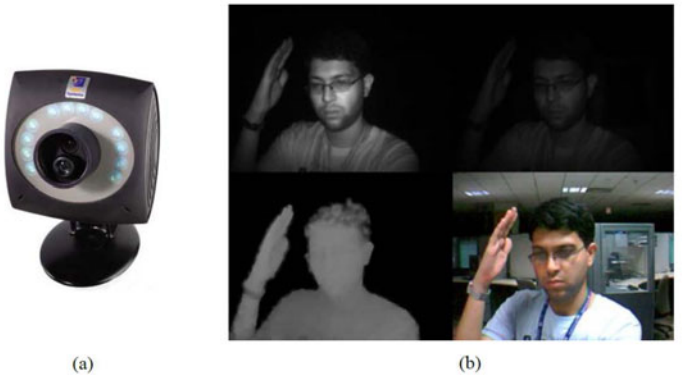

Fig. 2. (a) ZCam from 3DV Systems; (b) Data output from ZCam - top row: primary and secondary infrared images, bottom row: a depthmap and a RGB color image

upper-body and face are visible without any occlusion. A block diagram of the human detection and pose estimation approach used in our work is shown in Fig. 1. The following subsections provide details of each module incorporated in the system.

2.1 Depth Camera

We use ZCam from 3DV Systems [1] (shown in Fig. 2(a)) in our work done in mid-2010. The technology of this device is similar to Kinect systems currently available in the market. This camera uses active illumination for depth sensing - it emits modulated infra-red (IR) light and based on the time-of-flight principle, the reflected light is used to calculate depth (distance from camera) in a scene. This camera provides both RGB (640 x 480 resolution, VGA size) image and a grayscale depthmap (320 x 240 resolution, QVGA size) image at 30 frames per second (fps). Figure 2(b) shows a sample of four images obtained from the camera. The top row shows active brightness (left) and passive brightness (right) IR images and the bottom row shows the depthmap (left) and the RGB (right) image respectively. It can be observed in the depthmap, that the depth values of objects near the camera appear bright while those of objects that are farther appear darker.

2.2 Foreground Segmentation

We use the RGB image and the depthmap as inputs to the system (see Fig. 3). A threshold is used to remove noise (with low values) from the raw depth map information, obtained from ZCam without any calibration. These foreground pixels are then segmented into regions by a linear-time component labeling algorithm [6]. The extracted connected components or blobs, obtained from the depth map using 8-connectivity of pixels, are thresholded based on area. The

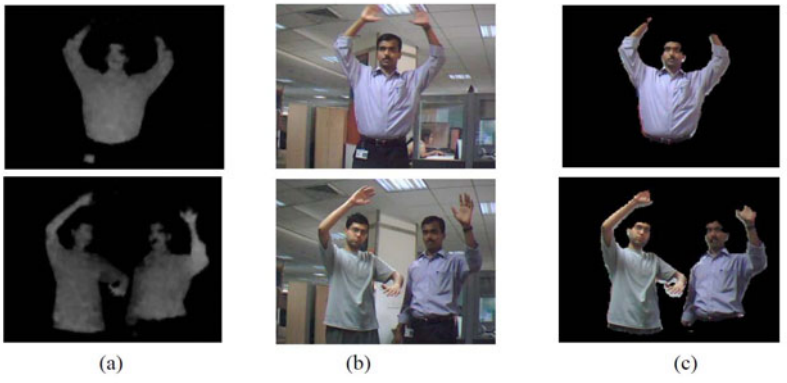

Fig. 3. (a) Input depthmaps; (b) Input RGB images; (c) Foreground segmented RGB images obtained using (a) and (b)

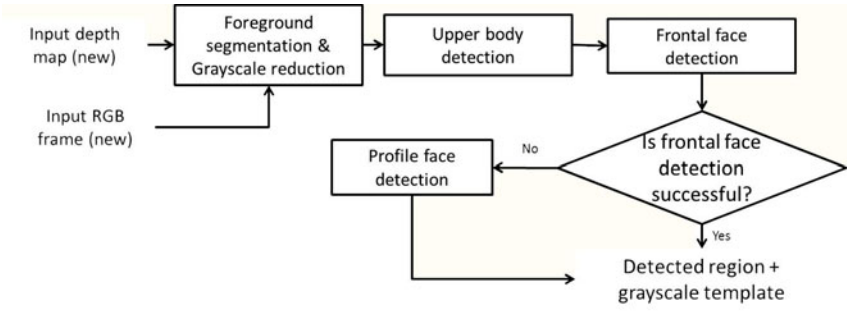

Fig. 4. Haar cascade based detection logic

above blob analysis helps in pruning out small blobs and background noises in the input image. The processed depthmap is then used as a binary mask to obtain the foreground object in the RGB image.

2.3 Haar Cascade Based Detection

The object detector [16] based on Haar classifiers is used for detecting humans in the foreground segmented RGB images. Grayscale based object detector is used instead of an RGB based object detector, since it reduces the time complexity of the system by operating on a single channel. Human detector helps in differentiating humans from non-human objects present in the segmented foreground grayscale image (non-trivial using depth mask detection). For upper body detection, the classifier trained for upper-body (head + torso) [15] is used. The detected regions are then passed on to frontal face detector classifier (see Fig. 4). In case, the frontal face detection fails, a profile face detector [5] is used to detect faces. If either upper body detector or the profile face detector fails to produce any positive results then the frame is completely rejected and the next

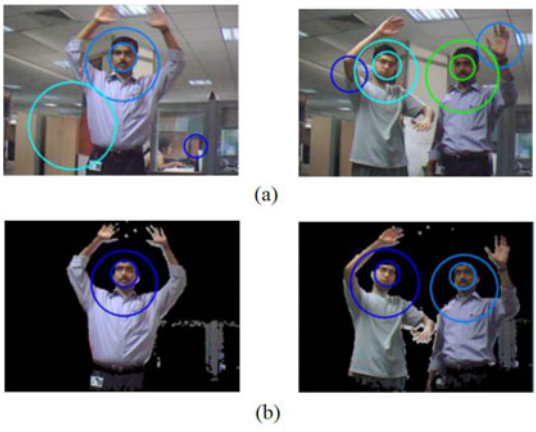

Fig. 5. Haar cascade based detection for upper-body and face. Circumscribed circles denote successful face (inner circle) and upper body detection (outer circle), whereas a single circle denotes a successful upper-body (either false positive or true positive) detection along-with unsuccessful face detection (either false negative or true negative). (a) Haar cascade based detection on original grayscaled RGB images. (b) Haar cascade detection on foreground segmented grayscaled RGB images.

frame is analyzed for any possible upper-body detection. If no face is detected in the identified upper body region, then it is assumed to be a false positive and the detection is rejected for further analysis. This successive detection logic helps in reliably determining the positive detections and pruning out the false positive detections. In order to reduce the computation time as well as the false positives, Haar detection is done on the foreground segmented image (see Fig. 5).

2.4 Template Matching Based Tracking

The template-based approach determines the best location by matching an actual image patch against an input image, by "sliding" the patch over the input search image using normalized cross-correlation, defined as:

$$R(x,y) = \frac{\sum_{x',y'}(T^{G'}_{RGB}(x',y') \cdot I^{G'}_{RGB}(x+x',y+y'))}{\sqrt{\sum_{x',y'} T^{G'}_{RGB}(x',y')^2 \cdot \sum_{x',y'} I^{G'}_{RGB}(x+x',y+y')^2}} \tag{1}$$

$$\text{where,} \qquad T^{G'}_{RGB}(x,y) = T^{G}_{RGB}(x,y) - \overline{T^{G}_{RGB}}$$

$$I^{G'}_{RGB}(x,y) = I^{G}_{RGB}(x,y) - \overline{I^{G}_{RGB}}$$

T^{G}_{RGB} is the grayscaled RGB template image and I^{G}_{RGB} is the grayscaled RGB input image. Since template-based matching requires sampling of a large number of points, we can reduce the number of sampling points by reducing the

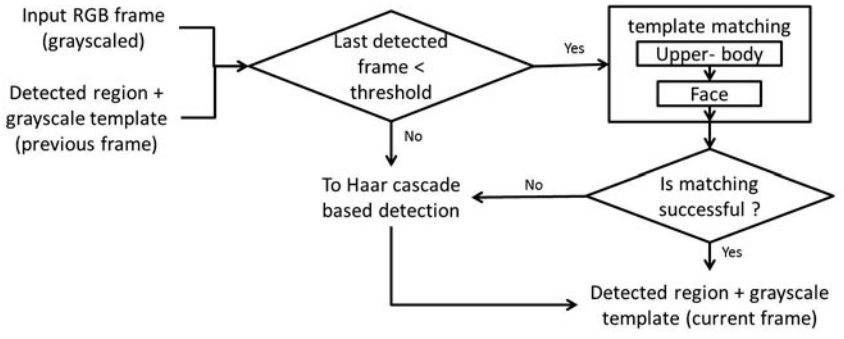

Fig. 6. Template Matching based tracking logic

Fig. 7. Results for template matching based tracking. Templates are grayscaled and down-sampled to QVGA to reduce computation time. Similarly, input RGB image is also grayscaled and down-sampled to QVGA: (a) upper-body template identified in previous frame; (b) face templates identified in previous frames; (c) input image grayscaled and down-sampled with marked rectangular regions denoting successful template based tracking.

resolution of the search and template images by the same factor (in our case, down-sampled by a factor of 2) and performing the operation on the resultant downsized images. The template images/patches are obtained from the successful detection in the previous frame; either by Haar cascade based detection or by template based matching (see Fig. 6). Advantages of using template matching, over Haar cascades, is reduced computation time and higher true positives, since a Haar cascade misses variations in object orientation and pose. Template matching is successful in handling large pose variances of the object, if the inter-frame variance is low, since consecutive frames are used for matching. The system may fail for humans not facing (non-frontal pose) the camera. Haar cascade based detection is used only when there are no templates to perform matching or when

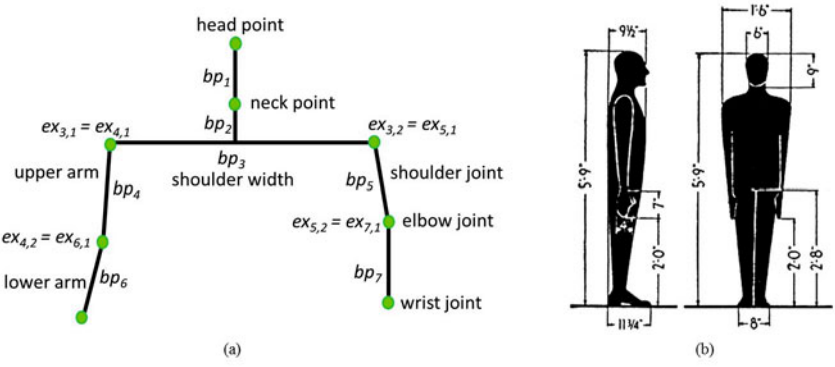

Fig. 8. (a) The stick model used for human upper-body skeleton fitting; (b) Anthropometric ratios of typical human body [3]

the template matching fails to track the template in the input image. Haar cascade based detection is forced after certain empirically chosen time-lapse/frames, to handle drifting errors and appearance of new person into the scene. Figure 7 shows examples of the template matching on the input images.

2.5 Stick Human Body Model

The skeleton model is represented by a vector of 7 body parts (bp_1 to bp_7) as shown in Fig. 8(a). The proportions between the different parts are fixed and were determined based on NASA Anthropometric Source Book [7] and [3](see Fig. 8). Each body part (bp_i) is composed of two extremities ($ex_{i,1}, ex_{i,2}$), representing the coordinates of the body part in the image plane:

$$bp_i = \{ex_{i,1}, ex_{i,2}\} \tag{2}$$

where, $ex_{i,j} = (x_{i,j}, y_{i,j})$. $x_{i,j}$ is the x coordinate of extremity j of the body part i and $y_{i,j}$ is the coordinate of the extremity j of the body part i.

The head, neck and shoulder (both left and right) joints are estimated based on detected upper-body and head region. The centroid of the detected head template is taken as head point. The shoulder joints are taken as the lower extremities of the detected upper body region in the input image. Based on the anthropometric ratios, the neck point is estimated to be at 2/3 of the vertical distance from head to shoulder points. Similarly, length of upper arms is taken as 2/3 of shoulder width and 5/9 of shoulder width in case of lower arms. This helps to detect head, neck and shoulder points of the detected humans from the foreground segments of the input grayscale image.

2.6 Limbs Fitting

In order to estimate the remaining joints (elbow and wrist, both left and right) and limb inclinations (upper and lower arm, both left and right), linear regression

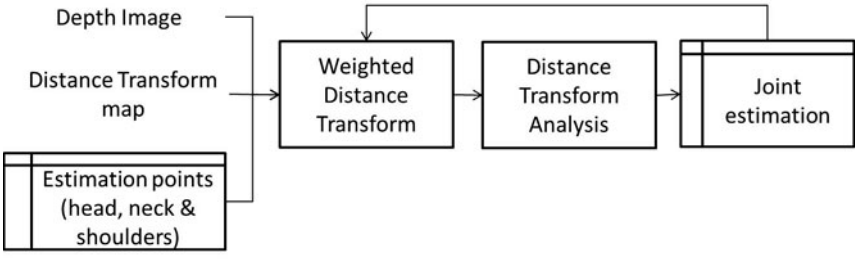

Fig. 9. Flowchat of limbs fitting method, based on linear regression of sampled weighted distance transform map

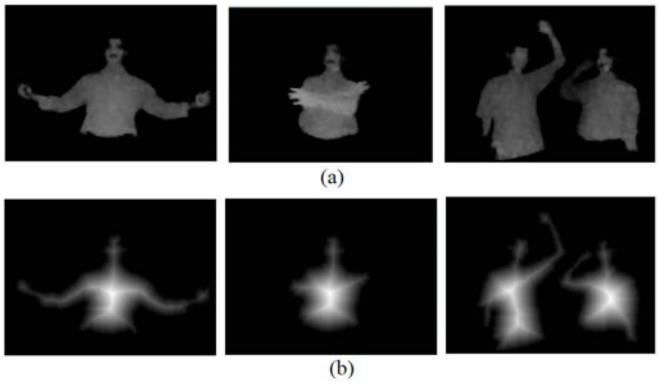

Fig. 10. DT on foreground segmented depthmap normalized from 0 to 255 range for visualization: (a) foreground segmented depthmap; (b) distance transform map

on sampled weighted-distance transform map (distance transform analysis) is performed (see Fig. 9). Once the elbow joints are estimated (as discussed in Sec. 2.5), weighted-distance transform w.r.t. these joints are computed for estimating wrist joints and 2D inclinations for lower arms. The Distance Transform (DT) maps each image pixel into its smallest distance to regions of interest [21]. Figure 10 shows some examples of DT on input images. Limb movements for human body can be out of the image plane, which DT fails to capture in the depthmap. In order to take into account the projected lengths of the limbs weighted-distance transform is calculated. The distance map of the image is multiplied with variance factor representing the variance ratio of the point w.r.t. the reference point (parent joint) in the direction orthogonal to the image plane. This variance can easily be calculated from the input depthmap. The weighted-distance transform $D^w(p, c)$ for point p w.r.t. c in depth image (I_d) is defined as:

$$D^w(p,c) = D(p) \cdot (1 + \frac{|I_d(p) - I_d(c)|}{I_d(c)}) \qquad \forall \quad I_d(c) \neq 0 \tag{3}$$

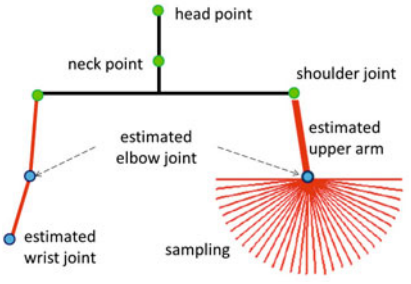

Fig. 11. Sampling of weighted distance transform map for left lower arm pose estimation. The green color points have already been estimated based on upper body and head region detection. The blue colored joints are estimated by sampling followed by linear regression.

where, $D(p)$ is DT value at point p in the input depth map image I_d. c is the reference point (parent joint) for estimating the angles for upper and lower arms. e.g. for estimating the inclination of upper left arm, reference point (c) is left shoulder joint and similarly for estimating the lower right arm, reference point (c) is right elbow joint. Sampling of the Weighted-Distance Transform map is done upto length l from the reference point (parent joint) c in an angular region varying from 0 to 2π, and with a predefined sampling angle. Temporal information can be incorporated to improve computational efficiency by imposing range constraints on the angular region for sampling the map (see Fig. 11). The length l of arms is estimated based on anthropometric ratios as discussed in Sec. 2.5. The step size for sampling (search for optimal value in 1-D) the orientation angle influences the robustness and speed of the technique. If it's too large, a good solution could be overlooked. However, the whole process might take too long if the step size is chosen small. It then becomes possible to sample points along and for each candidate solution. In estimation of both upper arms and lower arms, a second global maximum is taken as the estimated pose of the limb. In case of upper arms, the global maxima always denotes the angle from left or right shoulder joint towards torso's center region; since weighted-distance transform map value is always maxima along this path (see Fig. 10). Similarly for lower arms, a global maximum denotes the angle connecting the elbow joints to shoulder joints, as the physical structure of human body, upper arms are broader in width compared to lower arms. Due to these reasons second maxima is universally chosen to represent the estimated limb's inclination.

The sampling rate is an adjustable parameter that also influences the robustness and speed of the method. Indeed, the more points there are along a line to validate a solution, the more robust the system is if a part of a limb has been poorly extracted. However, the fitting process becomes more time consuming. A local method such as the one presented above also increases the robustness of the whole system in the following way. If some region of the blob has been poorly extracted, it is likely that only this part will be poorly fitted, while the

Table 1. Computational time for various modules in our system

Modules	**Time/frame (in ms)**
Haar cascade based upper-body & face detection	$\sim 57ms/frame$
Skeleton fitting	$\sim 11ms/frame$
Total time using detection	**$\sim$ 68 ms/frame**
Template matching based tracking	$\sim 3ms/frame$
Skeleton fitting	$\sim 5ms/frame$
Total time using tracking	**$\sim$ 8 ms/frame**
Average Running Time (Threshold = 15 frames/sec)	**$\sim$ 14 ms/frame**

Fig. 12. (a) Foreground segmented grayscaled RGB image; (b) Input depthmap; (c) Estimated upper body human stick figure overlaid upon the grayscaled RGB image

other limbs will be successfully fitted if the upper body detection is successful. In the case of a global method, a small error can lead to the failure of the whole fitting module. However, because of the local fitting method, even if one part is missed, the overall fitting is often acceptable. The fitting process for the right arm is independent from that of the left arm, therefore, the error in the estimation process of the former will not affect the later, and vice-versa. This makes our proposed local approach more robust.

3 Experimental Results

We have developed a working prototype of our human detection and pose estimation logic. The prototype was implemented using C/C++ and OpenCV library, on a windows platform. The prototype works in real-time using live feeds from 3DV camera mounted on top of a personal computer. We have tested the above prototype for single as well as multiple (upto 3) non-intersecting people with appearance and disappearance of people at random and for various different upper body poses. The input RGB stream is of 640 x 480 resolution (VGA) at 30

fps and the depth stream is of 320 x 240 resolution (QVGA) at 30 fps. For foreground segmentation, blob with size less than 400 pixels (empirically chosen) are considered as non-humans. Haar cascade based detection is done on VGA size grayscaled RGB image to increase true positive detections. Template matching based tracking is done on a QVGA size grayscaled RGB image to reduce computation time. Threshold used for enforcing Haar cascade based detection is empirically chosen as 15 frames. Since foreground segmentation is the most critical step in pose estimation, poor foreground segmentation can sometimes lead to incorrect pose estimation. Figure 12 shows a few examples of our analysis done on input frames of humans interacting in various poses. Table 1 gives the time taken (on a machine with Intel Core 2 Extreme processor, 3 GHz and 3 GB RAM) for various processes in the prototype. The average running time of the entire process is less than the total time used for detection (∼68 ms/frame) since Haar cascade based detection is enforced only once in every 15 frames while for the rest of the frames, template matching based tracking (∼8ms/frame) is used. A rigorous performance analysis for measuring the scalability and robustness of our approach can be a possible scope of future work.

4 Conclusions

In this paper, we have presented a viable vision-based human pose estimation technique using RGB and depth streams from a monocular view. An articulated graphical human model is created for pose estimation of upper-body parts for HCI applications. Our technique uses a balance of Haar cascade based detection and template matching based tracking. Haar based detection handles appearance of humans and drifting errors in tracking, while template matching based tracking is able to handle variations in object pose and makes the approach computationally light. Limbs fitting is performed progressively, one limb at a time, instead of globally. This way, the process is faster and robust. We have demonstrated the technique for various real-world input data. Some improvements are possible in this framework. Incorporating skin detection and edge detection would reduce false positive configurations for lower arms. Occlusion handling and comparative studies with published work form nice scope of work in the future.

References

1. Zcam from 3dv systems (2009), `http://3dvzcam.com`
2. Aggarwal, J., Cai, Q.: Human motion analysis: A review. In: Proceedings of the Nonrigid and Articulated Motion Workshop, pp. 90–102 (1997)
3. Badler, N.I., Phillips, C.B., Webber, B.L.: Simulating Humans: Computer Graphics, Animation, and Control. Oxford University Press, Oxford (1993)
4. Barrón, C., Kakadiaris, I.A.: Estimating anthropometry & pose from a single uncalibrated image. Computer Vision and Image Understanding 81, 269–284 (2001)
5. Bradley, D.: Profile face detection (2003), `http://opencv.willowgarage.com`

6. Chang, F., jen Chen, C., jen Lu, C.: A linear-time component-labeling algorithm using contour tracing technique. Computer Vision and Image Understanding 93, 206–220 (2004)
7. Churchill, E., McConville, J.T., Laubach, L.L., Erskine, P., Downing, K., Churchill, T.: Anthropometric source book. A handbook of anthropometric data, vol. 2. NASA (1978)
8. Fujiyoshi, H., Lipton, A.J.: Real-time human motion analysis by image skeletonization. In: Proceedings of the Fourth IEEE Workshop on Applications of Computer Vision (WACV 1998), pp. 15–21 (1998)
9. Guo, Y., Xu, G., Tsuji, S.: Tracking human body motion based on a stick figure model. Journal of Visual Comm. and Image Representation 5(1), 1–9 (1994)
10. Haritaoglu, I., Harwood, D., Davis, L.: W4: Who? when? where? what? A real time system for detecting and tracking people. In: Proceedings of the Third IEEE Int. Conf. on Automatic Face and Gesture Recog., pp. 222–227 (1998)
11. Herda, L., Fua, P., Plänkers, R., Boulic, R., Thalmann, D.: Skeleton-based motion capture for robust reconstruction of human motion. In: Proceedings of the Computer Animation, pp. 77–83. IEEE Computer Society, Los Alamitos (2000)
12. Jensen, R.R., Paulsen, R.R., Larsen, R.: Analyzing gait using a time-of-flight camera. In: Salberg, A.-B., Hardeberg, J.Y., Jenssen, R. (eds.) SCIA 2009. LNCS, vol. 5575, pp. 21–30. Springer, Heidelberg (2009)
13. Johansson, G.: Visual motion perception. Scientific American 232(6), 76–89 (1975)
14. Kolb, A., Barth, E., Koch, R., Larsen, R.: Time-of-flight cameras in computer graphics. Computer Graphics Forum 29, 141–159 (2010)
15. Kruppa, H., Santana, M.C., Schiele, B.: Fast and robust face finding via local context. In: Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (October 2003)
16. Lienhart, R., Maydt, J.: An extended set of Haar-like features for rapid object detection. In: Proceedings of the International Conference on Image Processing, vol. 1, pp. 900–903 (2002)
17. Microsoft: Kinect for xbox 360 (2010), `http://www.xbox.com/en-US/kinect`
18. Moeslund, T., Hilton, A., Krüger, V.: A survey of advances in vision-based human motion capture and analysis. Computer Vision and Image Understanding 104(2-3), 90–126 (2006)
19. Ohya, J., Kishino, F.: Human posture estimation from multiple images using genetic algorithm. In: Proceedings of the 12th IAPR International Conference on Pattern Recognition, Conference A: Computer Vision & Image Processing. vol. 1, pp. 750–753 (1994)
20. Rashid, R.F.: Towards a system for the interpretation of moving light display. IEEE Transactions on Pattern Analysis and Machine Intelligence 2(6), 574–581 (1980)
21. Rosenfeld, A., Pfaltz, J.: Distance function on digital pictures. Pattern Recognition 1(1), 33–61 (1968)
22. Takahashi, K., Uemura, T., Ohya, J.: Neural-network-based real-time human body posture estimation. In: Proceedings of the IEEE Signal Processing Society Workshop Neural Networks for Signal Processing X, vol. 2, pp. 477–486 (2000)