Fast Gait Recognition from Kinect Skeletons

Tanwi Mallick, Ankit Khedia, Partha Pratim Das and Arun Kumar Majumdar
Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur 721302, India

Keywords: Gait Recognition, Kinect Skeleton Stream.

Abstract: Recognizing persons from gait has attracted attention in computer vision research for over a decade and a half. To extract the motion information in gait, researchers have either used wearable markers or RGB videos. Markers naturally offer good accuracy and reliability but has the disadvantage of being intrusive and expensive. RGB images, on the other hand, need high processing time to achieve good accuracy. Advent of low-cost depth data from Kinect 1.0 and its human-detection and skeleton-tracking abilities have opened new opportunities in gait recognition. Using skeleton data it gets cheaper and easier to get the body-joint information that can provide critical clue to gait-related motions. In this paper, we attempt to use the skeleton stream from Kinect 1.0 for gait recognition. Various types of gait features are extracted from the joint-points in the stream and the appropriate classifiers are used to compute effective matching scores. To test our system and compare performance, we create a benchmark data set of 5 walks each for 29 subjects and implement a state-of-the-art gait recognizer for RGB videos. Tests show a moderate accuracy of 65% for our system. This is low compared to the accuracy of RGB-based method (which achieved 83% on the same data set) but high compared to similar skeleton-based approaches (usually below 50%). Further we compare execution time of various parts of our system to highlight efficiency advantages of our method and its potential as a real-time recogniser if an optimized implementation can be done.

1 INTRODUCTION

Human gait is an important indicator of health and serves as an identification mark for an individual. It was first studied by the biologists because it can provide great information about health, with applications ranging from diagnosis, monitoring, and rehabilitation. However, now it is also accepted as unique identifier for an individual and so can be considered for identification and authentication of an individual.

In this paper, we try to use Kinect[1] 1.0 for detecting the gait of an individual. There are various systems available for gait analysis like wearable sensors, marker-based systems and Kinect is the latest technique in this race. However each has got its own pros and cons and their usage can be judged according to the context. The marker-based systems are the most accurate system used for gait analysis but they are generally very costly and can be used only in laboratory or controlled environments. Then comes the wearable sensors which are cheap, small, lightweight, mobile but they are intrusive, that is, the subject has to wear those sensors. Also they must account for signal drift and noise and must be placed correctly.

The latest sensor used for gait analysis is Kinect. It is cheaper compared to the above two and non-intrusive and can measure a wide range of gait parameters using the sensor and Software Development Kit (SDK).

However the problem is that the joint points are approximated by the Kinect and hence are not very accurate. But, almost all the gait detection systems first try to locate the joint points and then extract features using it so in spite of less accuracy we would still try to exploit this feature of Kinect in this paper so as to obtain maximum possible accuracy out of it as the overhead of joint extraction is removed.

Our objective is to identify an individual on the basis of her gait with maximum use of joint information in extracting various features, use of depth data information for increasing the accuracy, determining which of the features are more crucial over other and looking into different classification algorithms for different types of features.

The paper is organized as follows. Section 2 discusses the prior work in this area. We define the fea-

[1] Kinect for XBox One. has been released a while after this work was completed. This is called Kinect 2.0 now.

Mallick, T., Khedia, A., Das, P. and Majumdar, A.
Fast Gait Recognition from Kinect Skeletons.
DOI: 10.5220/0005713903400347
In *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2016) - Volume 3: VISAPP*, pages 342-349
ISBN: 978-989-758-175-5

tures of gait and their extraction in Section 3. Section 4 discusses the classifiers. Experiments and Results are explained in Section 5. Finally, we conclude in Section 6.

2 RELATED WORK

In vision research, there has been a lot of experiments to recognize people from gait. The gait detection problem also gives information about the well-being of an individual as well as can be used for recognition. Different works propose different approaches to the problem of gait detection. The gait recognition approach can broadly be of two types – *Marker-based* or *Marker-less*. Marker-based approaches make use of wearable sensors for gait detection and the marker-less approaches consist of using video cameras or Kinect for gait detection (Stone and Skubic, 2011), (Gabel et al., 2012), (Preis et al., 2012), (Sinha et al., 2013), (Wang et al., 2003), (Isa et al., 2005), (Ball et al., 2012).

2.1 Marker-based Approach

The marker-based approaches use of some sensors placed on the subject. *Moving Light Display* (MLD) is a light pattern corresponding to the moving subjects. Johansson (Johansson, 1973), (Johansson, 1975) showed that humans can quickly identify a moving light display (MLD), corresponds to a walking human but when presented with a static image from the MLD, humans are unable to recognize any structure. This was the basis of the marker-based systems for gait recognition. Tanawongsuwan & Bobick (Tanawongsuwan and Bobick, 2001) use joint-angle trajectories measured using a magnetic-marker motion-capture system. There is also relevant work in the computer animation field, including that of recovering underlying human skeletons from motion capture data (O'Brien et al., 2000),(Silaghi et al., 1998) and analysing and adjusting characteristics of joint-angle signals (Sudarsky and House, 2000), (Bruderlin et al., 1996), (Bruderlin and Williams, 1995) and (Brand and Hertzmann, 2000).

2.2 Marker-less Approach

This approach uses RGB video cameras or Kinect for data acquisition, extracts each frame from the video and then performs image processing to extract the relevant features for recognition. They can be broadly divided into two categories – RGB and RGB-D[2].

[2]RGB images with depth information.

RGB

The spatial and temporal features are mainly extracted from the RGB frames in various ways. Ran et. al. (Ran et al., 2007) use Hough Transform to extract the main leg angle and use Bayesian Classifier for gait detection. Jean et. al. (Jean et al., 2009) proposed the use of trajectories of significant body points like the head and feet for gait detection.

Model-free Human body silhouette is the most frequently used initial feature, which can be easily obtained from background subtraction. Boulgouris & Chi (Boulgouris and Chi, 2007) use Radon transform on silhouette to extract the feature of each frame, and employ *Linear Discriminant Analysis* (LDA) for dimensionality reduction. A similar method has been used in (Wang et al., 2003) where Wang et. al. detect gait patterns in a video sequence and develop an eigen gait or gait signature for the particular video.

Ben-Abdelkader et. al. (BenAbdelkader et al., 2001) exploit the self similarity to create a representation of gait sequences that is useful for gait recognition. Quasi gait methods rely on various static features like build of the body. One advantage to quasi gait approaches is that they may be less sensitive to variation in a gait. For example, the gait of a person may vary for various reasons, but their skeletal dimensions will remain constant. Bobick & Johnson (Bobick and Johnson, 2001) measured a set of four parameters that describe a static pose extracted from a gait sequence.

Kellokumpu et. al. (Kellokumpu et al., 2009) assume time as the third dimension other than XY axes in the image plane, so that consider the accumulation of gait sequence as XYT three-dimensional space. Davis & Bobick (Davis and Bobick, 1997) describe a *Motion Energy Image* (MEI) and a *Motion History Image* (MHI), both derived from temporal image sequences.

RGB-D

With the development of depth imaging, researchers has also tried using them for Gait analysis. The depth information can be obtained by using multiple cameras, stereo cameras or Kinect.

Non-Kinect

There have been several research in this field using non-Kinect based techniques. Igual et. al. (Igual et al., 2013) presented an approach for gait-based gender recognition using depth cameras. The main contribution of this study was a new fast feature extraction strategy that uses the 3D point cloud obtained from the frames in a gait-cycle. Ioanaddis et. al. (Ioannidis et al., 2007) proposed the use of innovative gait identification and authentication method based on 2-D and 3-D features. The data was captured using stereo camera which can be used to extract the depth information.

Kinect

Stone et. al. (Stone and Skubic, 2011) has tried to find any anomality of the subject on the basis of the walking speed and stride length using depth information returned by Kinect. Preis et. al. (Preis et al., 2012) proposed to directly calculate the static features(length of the bones) from the actual 3-D coordinates of the joints returned by the Kinect and presented some results using different classification algorithms and compared the performance of different algorithms. Sinha et. al. (Sinha et al., 2013) has presented the use of static, distance and area features with Neural Network learning for gait recognition using Kinect. Ball et. al. (Ball et al., 2012) has used Kinect for gait recognition. It has taken a very small dataset of 4 subjects and tried to do unsupervised clustering using angular features and K-means clustering algorithm. Some works has basically tried to calculate some gait features like stride length, speed using the marker based techniques and the Kinect based techniques and tried to find the accuracy of the Kinect based systems considering the other one as the standard (Gabel et al., 2012), (Stone and Skubic, 2011). Chattopadhyay et. al. (Chattopadhyay et al., 2014) explored the applicability of Kinect RGB-D streams in recognizing gait patterns of individuals. *Gait Energy Volume* (GEV) is a feature that performs gait recognition in frontal view using only depth image frames from Kinect.

In this work we judiciously select and combine the features, through a set of detailed experiments, to get maximum skeleton based recognition in optimal time.

3 FEATURE EXTRACTION

Gait is a continuous yet periodic process. Hence it is usually studied and analysed in terms of the half-gait-cycle. We define various features (usually over a half-gait-cycle) and discuss how they are extracted and what their characteristics are. The half-gait-cycle and the features are defined in terms of the 3D joint-points of the 20-joints' skeletal model (Figure 1) returned by Kinect in every frame. The skeleton stream is first cleaned up using the moving-average filter (of window size 8) to reduce noise due to sudden spikes.

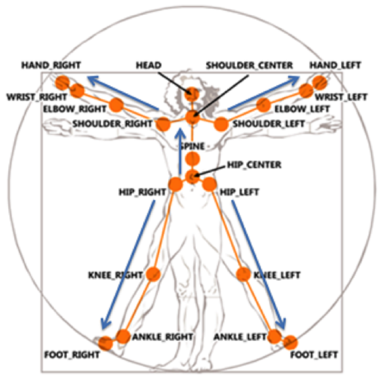


Figure 1: 20-joints' skeletal model tracked by Kinect. We refer to RIGHT as 'R', LEFT as 'L', and 'CENTER' as 'C'.

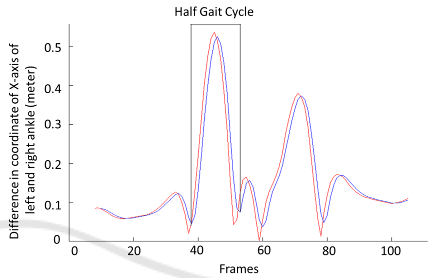


Figure 2: An example of half-gait-cycle extraction.

3.1 Half-Gait-Cycle Detection

Consider the plot (Figure 2) of the absolute difference of X-coordinates D_k between left and right ankle joint-points over consecutive frames. Formally, $D_k = |ANKLE\_L(k).x - ANKLE\_R(k).x|$ for $1 \leq k \leq N$, where N = total number of frames for an individual side-walk ($N > 1$). The plot is first cleaned up using the moving-average filter (of window size 3) to reduce noise. The half-gait-cycle is then defined as the frames between two consecutive local minima in this plot.

We use six types of features, namely, *Static*, *Area*, *Distance*, *Dynamic*, *Angular*, and *Contour-based* features here. The first 5 features are extracted from the skeleton stream while the contour-based features are extracted from the depth stream as detailed in the next few sections.

3.2 Static Features (10-tuple)

The static features estimate the physique of the user. They are invariant over movements. We define 10 static features (Table 1) in terms of the Euclidean distance, $d(.,.)$ between the adjacent joint-points. To estimate these features we consider the median of these values over the entire video to annul the effects of intermittent spikes.

Table 1: *Static features in terms of joint-points. The last 4 features are both for right and left limbs (X = R or L).*

Height	=	d(HEAD, SHOULDER_C) + d(SHOULDER_C, SPINE) + d(SPINE, HIP_C) + d(HIP_L, KNEE_L) + d(KNEE_L, ANKLE_L)
Torso	=	d(SHOULDER_C, SPINE) + d(SPINE, HIP_C)
Upper Arm (X)	=	d(ELBOW_X, SHOULDER_X)
Forearm (X)	=	d(ELBOW_X, WRIST_X)
Thigh (X)	=	d(HIP_X, KNEE_X)
Lower Leg (X)	=	d(KNEE_X, ANKLE_X)

3.3 Area Features (2-tuple)

During side-walk the upper (lower) part of the body sweeps a certain area by the swing and spread of the hands (legs). Each such area, usually, is a distinguishing factor for an individual. It is defined as the area of the XY-projection of a closed polygon of N joint-points $\vec{p}_i = (x_i, y_i, z_i), 0 \le i \le N, 3 \le N \le 20$, selected for the side-walk. It is given by $A = \frac{1}{2}\sum_{i=0}^{N}(x_i * y_j - y_i * x_j); j = (i+1)\ mod\ N$.

We consider two discriminating areas defined as:

Upper Body	SHOULDER_C, SHOULDER_R, HIP_R, HIP_C, HIP_L, & SHOULDER_L
Lower Body	HIP_C, HIP_R, KNEE_R, ANKLE_R, ANKLE_L, KNEE_L, & HIP_L

The area feature vector is computed as the mean of these numbers over a half-gait-cycle.

3.4 Distance Features (4-tuple)

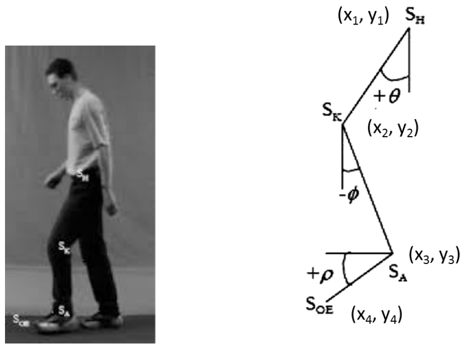
The Euclidean distance between the centroid of a body part and the centroid of the upper body is usually unique for an individual. The body part is contained by a closed polygon of N vertices and the centroid is computed as $\vec{c} = \frac{1}{N}\sum_{i=0}^{N}\vec{p}_i$. We consider four distances – to the centroids of both hands and legs. The corresponding polygons are defined as:

Upper Body	SHOULDER_C, SHOULDER_R, HIP_R, HIP_C, HIP_L, & SHOULDER_L
Right Hand	SHOULDER_R, ELBOW_R, & WRIST_R
Left Hand	WRIST_L, ELBOW_L, & SHOULDER_L
Right Leg	HIP_R, KNEE_R, & ANKLE_R
Left Leg	ANKLE_L, KNEE_L, & HIP_L

The distance feature vector is computed as the mean of these numbers over a half-gait-cycle.

3.5 Dynamic Features (2-tuple)

The *Stride Length* and *Speed* of the subject form the dynamic features. Consider the plot (Figure 2) of the absolute difference of X-coordinates between left and right ankle joint-points over consecutive frames. The gap (in X-coordinate) between the alternate maxima's (or minima's) in this plot gives the *step lengths*. We take the median of step lengths as the stride length. We compute the number of frames in a stride and using the Kinect's frame rate as 30 fps, we compute the speed of the subject as *stride length/stride time*. These dynamic features are situation dependent and can vary abruptly. Yet they often contain some individual-specific information that can improve the overall accuracy.



(a) (b)

Figure 3: Angular Features θ, ϕ, and ρ from (Isa et al., 2005), where $S_H \equiv$ HIP_X, $S_K \equiv$ KNEE_X, $S_A \equiv$ ANKLE_X, $S_{OE} \equiv$ FOOT_X, and X = RIGHT or LEFT.

3.6 Angular Features (6-tuple)

While walking different parts of the leg (side-view) make distinctive angles θ, ϕ, and ρ with the vertical and the horizontal lines. These are depicted in Figure 3. Using the XY-projection of the coordinates of the joint-points these angles can be computed as: $\theta = tan^{-1}\frac{|x_2-x_1|}{|y_2-y_1|}, \phi = tan^{-1}\frac{|x_3-x_2|}{|y_3-y_2|}, \rho = tan^{-1}\frac{|y_4-y_3|}{|x_4-x_3|}$. These angles are considered for the half-gait-cycle.

3.7 Contour-based Features

So far we considered features extracted from the skeleton stream. The contour-based feature, in contrast, is extracted from the depth stream. Recognizing people through gait depends on how the silhouette shape of an individual changes over time. *Procrustes Shape Analysis*[3] is used to obtain the *Gait Signature* (Wang et al., 2003) of the video as follows:

1. The first frame is taken as the static background and is subtracted from all frames to leave only the moving subject in them.

[3]Procrustes analysis is a form of statistical shape analysis used to analyze the distribution of a set of shapes.

2. Each frame is binarized using a threshold. Filter out the largest connected component. This is the shape or silhouette of the subject.
3. Compute the centroid of the silhouette from the points on the contour. Traverse the contour anti-clockwise to transform the points along outer contour in the coordinate system with the centroid $z_c = (x_c, y_c)$ as the origin. Represented each point as a complex number z_i.
4. The shape is represented as the $Z = [z_1, z_2, \cdots, z_{N_b}]$ where N_b is the number of points on the outer contour. Two representations represent same shape if one can be obtained from the other using a combination of *translation*, *rotation*, and *scaling*. Normalized representations for different frames of a video by interpolation such that they contain the same number of points.
5. Compute the principal eigen vector of the matrix $S = \sum(u_i u_i^*)/(u_i^* u_i)$ where u_i represents the configuration of a frame of the video and $*$ operation means the complex conjugate transpose of a matrix. The *Principal Eigen Vector* serves as gait signature for the video.

We use different classifiers for different features.

4 CLASSIFICATION

We use three different classifiers or matching algorithms - Naïve Bayes Classifier for static, area, distance and dynamic features, Dynamic Time Warping for angular features, and Procrustes Distance for contour features.

4.1 Naïve Bayes Classifier

The static, area, distance and dynamic features are mutually independent. Hence they are composed in a 18-dimensional feature vector (10 static, 2 area, 4 distance, and 2 dynamic). Naïve Bayes classifier is used with this feature vector to assign scores to each video in the training set with respect to its similarity to a test video. These scores are stored for later use. The higher the scores, more similar are the gaits.

4.2 Dynamic Time Warping

Angular features are considered as a sequence over a half-gait-cycle. To match such a sequence of a training video with that of a test video, we use *Dynamic Time Warping* (Müller, 2007). DTW works well for non-linear time alignment where one sequence is shifted, stretched, or shrunk in time with respect to the other. Time is normalized over a half-gait-cycle to adjust the sequences to the same length. Also, we perform variance normalisation of these sequences to reduce noise. A test video is matched against each of these sequences in the training set and the resulting DTW scores are stored in the database for later use.

4.3 Procrustes Distance

The contour-based feature is obtained as the eigen gait signature for a video s described in Section 3.7. The Procrustes distance between two gait signatures between a test and a training video is given by $d(u_1, u_2) = 1 - |u_1^* u_2|^2/(|u_1|^2 |u_2|^2)$, where u_1, u_2 are gait signatures and the $*$ operation represents the complex conjugate transpose of a vector. The smaller the distance, more similar are the gaits. The corresponding scores are stored in the database.

4.4 Composite Score

The differences in DTW and Procrustes distances are small compared to the Bayesian scores. Hence these differences are amplified by exponentiation and then the 3 scores are multiplied to obtain the composite score. Finally Nearest-Neighbour classifier is used on this composite score to classify a test video to the class of the training video where the score maximizes.

5 EXPERIMENTS AND RESULTS

The system has been implemented using several libraries. The videos are captured in C++ using Kinect Windows SDK[4] v1.8, the features are extracted using MATLAB 2012b, DTW & Procrustes distances also are computed using MATLAB, and a open-source code[5] for Naïve Bayes Classifier in C#.Net is used. We have carried out several experiment to validate our system as described below.

5.1 Data Sets and Processing

No benchmark gait dataset for skeleton and depth data from Kinect 1.0 is available. Hence we have created a dataset of 29 subjects (20 male and 9 female) for training as well as testing. For this 5 composite Kinect videos (comprising RGB, depth and skeleton streams)

[4] http://www.microsoft.com/en-in/download/details.aspx?id=40278

[5] http://www.codeproject.com/Articles/318126/Naive-Bayes-Classifier

Figure 4: 6 Frames depicting the half-gait-cycle of the subject alternating with the silhouettes from respective frames.

of the side-walk of each subject was recorded using the Kinect 1.0. In every video the subject moves in a straight-line without occlusion against a fixed background that is separately recorded (For sample RGB frames of a video see Figure 4). From the composite video we extract the individual streams[6].

The skeleton stream is first filtered using a moving average filter (of size 8) to reduce jitter. The joint-points are then used to extract the half-gait-cycle (Figure 4) and the static, area, distance, dynamic, and angular features as discussed in Section 3.

Frames from the depth stream are binarised after background subtraction. For the largest connected component in every binary frame (silhouette of the human – Figure 4) the contour is calculated. The contours are used to compute the gait signature of the depth video.

The same processing is done for the training as well as test videos to extract the features. For a test video, however, we need to compute the classification scores and the composite score against every training video (Section 4). These are fed to the Nearest Neighbour classifier for final recognition.

5.2 Results

We use 5 videos each for 29 subjects (20 male and 9 female). The system is trained with 4 of these videos for every subject and the 5th video is used for testing. The performance of the system is measured by the accuracy – the ratio of the number of videos correctly labeled to the total number of test videos (29 here).

To understand the effectiveness and discriminating power of various features, we have performed the recognition using various sets of features (and corresponding classifiers). The results are given in Table 2. The results show that the static and angular features are the most dominating. The dynamic features (speed and stride length), though situation dependent, help to increase the accuracy while area and distance have hardly any impact. The increase in accuracy after incorporating contour based features is marginal because the contour based features are already been taken care of by other features like the distance and area features.

Table 2: Accuracy with different feature sets.

Features used	Accuracy
Static features	48.25%
Distance features	34.48%
Angular features	37.93%
Static, distance features	44.82%
Static, distance, area & dynamic features	55.17%
Static, distance, area & dynamic & angular features	**65.57%**
Static, distance, area & dynamic, angular & contour based features	68.96%

Using the features extracted from the skeleton stream, we get accuracy of around 65%. The lack of accuracy is due to the inaccuracy of the coordinates of the joint-points. The skeletons often are erroneous and any error in this leads to significant loss of feature information.

In Table 3, we compare our results with a number of previous papers using the accuracy data as reported in each. We find that methods working on RGB have better accuracy at the cost of efficiency. Only one Kinect skeleton-based approach (Preis et al., 2012) achieved accuracy comparable to RGB methods. However, its results are reported on a small data set. Otherwise, our method achieves a better accuracy compared to other skeleton-data methods.

For an apple-to-apple comparison we have implemented an RGB-based method by mixing the approaches from (Roy et al., 2012) and (Wang et al., 2003). We test this method with the same data set as our system (only RGB frames are used) and the results are given in Table 4. We find that this achieves a much better accuracy of 83% (in comparison to our 65%) albeit at the cost of efficiency.

[6]While the skeleton stream is used for most features, the depth stream is used only for contours, and the RGB stream is used just for visualization. It has no contribution to the recognition tasks.

Table 3: Comparison with reported results from prior work.

Description	Remarks
16 Wearable Sensor, Angular features, 2 data sets: 73%: 1st. 42%: 2nd (Tanawongsuwan and Bobick, 2001)	Good accuracy but intrusive and costly
RGB sensor, Contour based features, 71% (Wang et al., 2003)	High processing time for each RGB frame
RGB Sensor, Pose Kinematics & Pose energy images, 83% (Roy et al., 2012)	Good accuracy on large dataset. Heavy computation
Kinect, Skeleton, Static features, 85% (Preis et al., 2012)	High accuracy; Small dataset (9 subjects); Frontal view (stationary subjects)
Kinect, Skeleton, Angular features, Avg.: 44% sub1: 35%, sub2: 74%, sub3: 39%, sub4: 33% (Ball et al., 2012)	Accuracy low for even small dataset
Kinect, Skeleton, Static, distance & area features, 25% (Sinha et al., 2013)	Very low accuracy as angular features not considered

Table 4: Comparison on same data set (RGB only).

Description	Remarks
RGB Sensor, Shape feature, 83% (Wang et al., 2003)	Good accuracy on large dataset.
RGB Sensor, Key Poses (Roy et al., 2012)	Heavy computation.

We extract the shape based feature (Wang et al., 2003) from RGB data then estimate the key poses (Roy et al., 2012) to recognize gait from Kinect RGB data of our gait data set.

6 CONCLUSION

There have been several attempts to recognize gait from RGB video. Many of these offer about 85% accuracy (Tables 3 and 4). Handling RGB data is expensive in terms of processing speed and hence most of these methods cannot work in real-time. In contrast, the present system works mainly with skeleton stream to recognize gait. Skeleton data is less in volume (only 60 floating point numbers per frame corresponding to the 3D coordinates of 20-joints) compared to RGB or depth data (typically 640 X 480 $\approx$ 0.3 million integers). Therefore skeleton-based techniques are more amenable to real-time processing.

The system takes about 1.5 secs (for a test video) to recognize the gait if only static, area, distance & dynamic features are used. This gives over 55% accuracy (Table 2) which is better than similar skeleton-based methods reported earlier (Table 3). Recognition from RGB (Roy et al., 2012), (Wang et al., 2003) on the same data set takes about 12 secs each video while the accuracy improves to 83% (Table 4).

If angular features are added to the set, the execution time of our system increases to about 29 secs / video while the accuracy goes to over 65% (Table 2). This nearly 20-fold increase in time is due to the use of DTW in matching because we use a naïve MATLAB implementation that is quadratic in complexity. Using a linear implementation can drastically reduce this time. Also, reduction of the dimensionality of the angular feature set can substantially improve time.

Adding contour-based features to our set improves the accuracy to 69% (Table 2) while the time shoots to 127 secs. This is due to use of depth data that is inherently heavy. Hence we recommend not to use depth data and contour-based features.

We are, therefore, working further on smarter implementations for skeleton-based features for meeting real-time constraints and at the same time experimenting with better classifiers including HMM and SVM.

ACKNOWLEDGEMENT

We acknowledge TCS Research Scholar Program.

REFERENCES

Ball, A., Rye, D., Ramos, F., and Velonaki, M. (2012). Unsupervised clustering of people from skeleton data. In *Human-Robot Interaction, Proc. of Seventh Annual ACM/IEEE International Conference on*, pages 225–226.

BenAbdelkader, C., Cutler, R., Nanda, H., and Davis, L. (2001). Eigengait: motion-based recognition of people using image self-similarity. In *Audio- and Video-Based Biometric Person Authentication (AVBPA 2001). Lecture Notes in Computer Science. Proc. of 3rd International Conference on*, volume 2091, pages 284–294.

Bobick, A. F. and Johnson, A. Y. (2001). Gait recognition using static activity-specific parameters. In *Computer Vision and Pattern Recognition (CVPR 2001). Proc. of 2001 IEEE Computer Society Conference on*, volume 1, pages 423–430.

Boulgouris, N. V. and Chi, Z. X. (2007). Gait recognition using radon transform and linear discriminant analysis. *Image Processing, IEEE Transactions on*, 16:731–740.

Brand, M. and Hertzmann, A. (2000). Style machines. In *Computer Graphics and Interactive Techniques (SIGGRAPH '00). Proc. of 27th Annual Conference on*, pages 183–192.

Bruderlin, A., Amaya, K., and Calvert, T. (1996). Emotion from motion. In *Graphics Interface (GI '96). Proc. of Conference on*, pages 222–229.

Bruderlin, A. and Williams, L. (1995). Motion signal processing. In *Computer Graphics and Interactive Techniques (SIGGRAPH '95). Proc. of 22nd Annual Conference on*, pages 97–104.

Chattopadhyay, P., Roy, A., Sural, S., and Mukhopadhyay, J. (2014). Pose depth volume extraction from rgb-d streams for frontal gait recognition. *Journal of Visual Communication and Image Representation*, 25:53–63.

Davis, J. W. and Bobick, A. F. (1997). The representation and recognition of human movement using temporal templates. In *Computer Vision and Pattern Recognition. Proc. of IEEE Computer Society Conference on*, pages 928–934.

Gabel, M., Gilad-Bachrach, R., Renshaw, E., and Schuster, A. (2012). Full body gait analysis with Kinect. In *Engineering in Medicine and Biology Society (EMBC), Proc. of 2012 Annual International Conference of the IEEE*, pages 349–361.

Igual, L., Lapedriza, A., and Borras, R. (2013). Robust Gait-Based Gender Classification using Depth Cameras. *Eurasip Journal On Image And Video Processing*.

Ioannidis, D., Tzovaras, D., Damousis, I. G., Argyropoulos, S., and Moustakas, K. (2007). Gait recognition using compact feature extraction transforms and depth information. *Information Forensics and Security, IEEE Transactions on*, 2:623–630.

Isa, W. N. M., Sudirman, R., and Sh-Salleh, S. H. (2005). Angular features analysis for gait recognition. In *Computers, Communications, & Signal Processing with Special Track on Biomedical Engineering (CCSP 2005). Proc. 1st International Conference on*, pages 236–238.

Jean, F., Albu, A. B., and Bergevin, R. (2009). Towards view-invariant gait modeling: Computing view-normalized body part trajectories. *Pattern Recognition*, 42:2936–2949.

Johansson, G. (1973). Visual perception of biological motion and a model for its analysis. *Perception and Psychophysics*, 14:201–211.

Johansson, G. (1975). Visual motion perception. *Scientific American*, 232:76–88.

Kellokumpu, V., Zhao, G., Li, S. Z., and Pietikinen, M. (2009). Dynamic texture based gait recognition. In *Advances in Biometrics (ICB 2009). Lecture Notes in Computer Science. Proc. of 3rd International Conference on*, volume 5558, pages 1000–1009.

Müller, M. (2007). *Dynamic Time Warping*.

O'Brien, J. F., Jr., R. E. B., Brostow., G. J., and Hodgins, J. K. (2000). Automatic joint parameter estimation from magnetic motion capture data. *Graphics Interface. Proc. of Conference on*, pages 53–60.

Preis, J., Kessel, M., Werner, M., and Linnhoff-Popien, C. (2012). Gait recognition with kinect. In *Kinect in Pervasive Computing, Proc. of the First Workshop on*.

Ran, Y., Weiss, I., Zheng, Q., and Davis, L. S. (2007). Pedestrian detection via periodic motion analysis. *International Journal of Computer Vision*, 71:143–160.

Roy, A., Sural, S., and Mukhopadhyay, J. (2012). Gait recognition using pose kinematics and pose energy image. *Signal Processing*, 92:780–792.

Silaghi, M.-C., Plänkers, R., Boulic, R., Fua, P., and Thalmann, D. (1998). Local and global skeleton fitting techniques for optical motion capture. In *Modelling and Motion Capture Techniques for Virtual Environments (CAPTECH '98). Proc. of International Workshop on*, pages 26–40.

Sinha, A., Chakravarty, K., and Bhowmick, B. (2013). Person identification using skeleton information from kinect. In *Advances in Computer-Human Interaction (ACHI 2013), Proc. 6th International Conference on*, pages 101–108.

Stone, E. and Skubic, M. (2011). Evaluation of an inexpensive depth camera for in-home gait assessment. *Journal of Ambient Intelligence and Smart Environments*, 3:349–361.

Sudarsky, S. and House, D. (2000). An Integrated Approach towards the Representation, Manipulation and Reuse of Pre-recorded Motion. In *Computer Animation (CA '00). Proc. of Conference*, pages 56–61.

Tanawongsuwan, R. and Bobick, A. (2001). Gait Recognition from Time-normalized Joint-angle Trajectories in the Walking Plane. In *Computer Vision and Pattern Recognition (CVPR 2001), Proc. of 2001 IEEE Computer Society Conference on*, pages 726–731.

Wang, L., Tan, T., Hu, W., and Ning, H. (2003). Automatic gait recognition based on statistical shape analysis. *Image Processing, IEEE Transactions on*, 12:1120–1131.