2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Learning Effective Gait Features Using LSTM

Yang Feng, Yuncheng Li and Jiebo Luo
Department of Computer Science, University of Rochester, Rochester, NY 14627
{yfeng23, yli, jluo}@cs.rochester.edu

***Abstract*—Human gait is an important biometric feature for person identification in surveillance videos because it can be collected at a distance without subject cooperation. Most existing gait recognition methods are based on Gait Energy Image (GEI). Although the spatial information in one gait sequence can be well represented by GEI, the temporal information is lost. To solve this problem, we propose a new feature learning method for gait recognition. Not only can the learned feature preserve temporal information in a gait sequence, but it can also be applied to cross-view gait recognition. Heatmaps extracted by a convolutional neutral network (CNN) based pose estimate method are used to describe the gait information in one frame. To model a gait sequence, the LSTM recurrent neural network is naturally adopted. Our LSTM model can be trained with unlabeled data, where the identity of the subject in a gait sequence is unknown. When labeled data are available, our LSTM works as a frame to frame view transformation model (VTM). Experiments on a gait benchmark demonstrate the efficacy of our method.**

I. Introduction

Gait recognition focuses on the problem of identifying people by the way they walk. Compared to face and iris, gait can be obtained from a distance without cooperation. Such an advantage makes it quite suitable for video surveillance. While applying gait recognition to real applications, we are facing several problems. The gait of a person can be affected by his shoes and weight-carrying conditions. And viewing at different angles often makes the gait look very different.

Many methods have been proposed to solve the cross-view gait recognition problem. Bodor et al. [1] proposed to use the image-based rendering technique to adapt the input to the view matching the training view, and thus solved the view angle changing problem. This 3D method is not suitable for the public surveillance scenario because it requires several calibrated cameras. Goffredo et al. [2] proposed to develop view-invariant gait features using angular measurements and trunk spatial displacement. Zheng et al. [3] developed view transformation model (VTM) for gait recognition, which transforms gait features from different views to the same view. Gait Energy Image (GEI) [4] is used as gait feature in this method. GEI is calculated by averaging the binary silhouette of human body in one gait cycle. The shape of the human body and the spatial information is well retained in the GEI template, while the temporal information is lost because of averaging. To preserve the temporal information, Wang et al. [5] proposed Chrono-Gait Image (CGI), which encodes gait contour images with a multichannel mapping. Castro et al. [6] proposed a local motion based gait descriptor.

In this paper, we propose a new feature learning method for gait recognition. Deep neural networks are used in the feature learning process. Instead of using binary silhouette to describe each frame, we use the human body joint heatmap extracted by a recent pose estimation method [7]. After obtaining the joint heatmap, we feed the joint heatmap of consecutive frames to Long Short Term Memory (LSTM) [8] to model the gait sequence. The hidden activation values at the last timestep is used as our gait feature. The gait feature extraction is illustrated in Figure 1. Compared to GEI, our feature has several advantages. Binary silhouette may be affected by the covariate factors such as wearing a coat and carrying a bag. However body joint heatmap is not affected by these covariate factors when the pose estimation method is robust enough. The silhouettes in one gait cycle are averaged to represent the gait in GEI. Some dynamic information is lost during the averaging process. To keep the dynamic information, we use LSTM to model a gait sequence. LSTM is good at modeling sequences and has achieved many promising results in sequence modeling tasks such as automatic speech recognition (ASR) [9] and machine translation [10]. However,

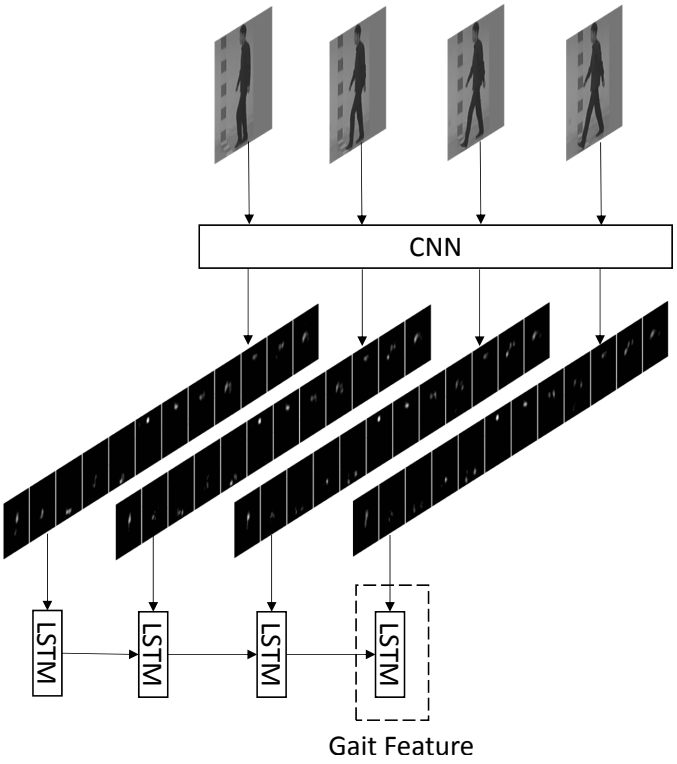

Fig. 1. Illustration of our framework. Each frame is transformed to a joint heatmap using a CNN. Then joint heatmaps are fed into a LSTM. The hidden values at the last timestep is regarded as gait feature.

as far as we know, it has not been used in gait recognition before. Our method demonstrates the effectiveness of applying LSTM to gait recognition.

When solving the cross-view gait recognition problem, many existing methods [3, 11] try to transform the gait feature of the whole sequence in one view to another view. GEI is usually used as the gait feature in those methods, and thus they suffer from the aforementioned problem of losing temporal information. Our method is different from them in that we try to transform the joint heatmap across views frame by frame. This is a much more accurate modeling method because it can preserve both the temporal and spatial information among the frames. When training the model, we have the LSTM encode all the joint heatmap of several frames in one view into a vector and use this vector to decode the joint heatmap in another view. This vector contains the information in the whole sequence and is used as the gait feature.

Unlabeled gait sequences can be used for training in our method. There are a large number of surveillance cameras gathering data all the time. A huge amount of unlabeled gait sequence can be obtained by these cameras. These data cannot be used in the existing cross-view gait recognition methods because those methods rely on corresponding data in both views for training. Our gait feature learning model can be trained in the autoencoder fashion. The input and output of our model can be the same sequence. When training the LSTM model to reconstruct gait sequences under two views, it can learn the common aspects between the two views and thus improve the cross-view gait recognition performance.

Our contributions can be summarized as follows:

1. We propose a new data driven feature learning method for gait recognition.
2. We can make use of unlabeled data for training.
3. Our frame to frame matching method achieves superior performance in most cases on a gait benchmark.

II. Related Work

Gait recognition has been extensively studied in the last 30 years. Generally speaking, cross-view gait recognition methods can be roughly divided into three categories. The first category of methods try to construct 3D gait information. Bodor et al. [1] proposed to use the image-based rendering technique to adapt the input to the proper view matching the training view, and thus solved the view angle changing problem. Zhao et al. [12] used video sequences captured under multiple views to construct 3D human model. This kind of methods are not suitable for public surveillance scenario because they require several calibrated cameras. The second category of methods is to develop view-invariant gait feature. Goffredo et al. [2] first estimated the lower part of body limb pose and then projected the gait parameters to the lateral plane. Angular measurements and trunk spatial displacement were used as a view-invariant gait feature. One limitation to this method is that it cannot be applied to frontal view because their method cannot estimate limb pose under that view. Kale et al. [13] used perspective projection model and optical flow based structure from motion equations to generate side-view gait from other views. Camera calibration is also needed in their method. The third category learns a transformation which transforms gait sequences from one view to another view. Sample pairs of the same subject from both views are needed for training the transformation model. A lot of methods belonging to this category have been proposed in the past few years. Kusakunniran et al. [14] proposed to optimize the GEI feature vectors using Linear Discriminant Analysis (LDA) and build view transformation model (VTM) based on the optimized GEI. VTM construction is reformulated using support vector regression (SVR) in [15] to predict the local motion in one view using local region of interest (ROI) in another view.

The method proposed in this paper belongs to the third category. Similar to [2], we extract the pose of body. Besides lower part body joints, 12 body joints among the whole body are detected using the pose estimation method [7]. We use LSTM as our view transform model and learn the gait feature in a data driven manner. Leveraging the method in [16], the sequence of estimated joint heatmap from one view is either transformed to another view or used to reconstruct itself. The LSTM model encode the whole sequence into one vector and this vector is used as a feature for gait recognition.

Recently, several CNN based methods have been applied to gait recognition. Yan et al. [17] fed the GEI to a CNN to predict multiple attributes and thus learn a gait feature. However, this is also a GEI based method and still suffers from losing temporal information problem. Wu et al. [18] proposed to learn CNN features from image set and used silhouettes in a sequence to learn the gait feature. Feature vectors of several frames are added up in their method. Castro et al. [19] used CNN to learn high-level descriptors for gait from optical flow components. The temporal information kept in their method is based on low-level motion features, while our LSTM model can capture high-level motion features from the joint heatmap.

III. Human Pose Estimation

GEI has been used in gait recognition for a long time. It is generated by averaging the aligned binary silhouettes of human body in one gait cycle. The binary silhouette is able to describe the body state in one frame well and the computation cost of extracting silhouette is very low. However, using silhouettes also has some disadvantages. Silhouette is sensitive to changing clothes and changing weight-carrying conditions. These variations decrease the gait recognition accuracy. To maintain the ability of describing body state and keep from unwanted variations, we decide to use a heatmap of body joints generated by a pose estimation method instead of using binary silhouettes. Fig. 2 shows some examples of body joint heatmap. The body joint heatmap is used to estimate the position of body. The position of the brightest point on one heatmap is regarded as the position of the corresponding joint. Around the brightest point, there is a bright spot area, which means that area is of high response and likely to be the position of that joint. The pose of the

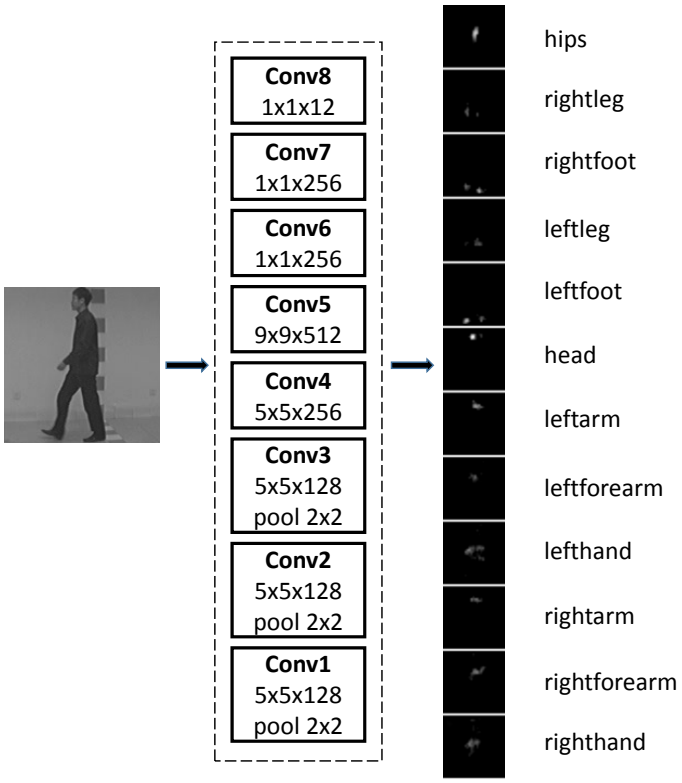

Fig. 2. The human pose estimation model. This is a modified version of the model in [7].

whole body can be clearly represented using the joint heatmap. Moreover, if the pose estimation method is robust enough, the body joint heatmap is invariant to changing clothing and changing carrying conditions, making it very suitable for gait recognition.

Among the pose estimation methods, we adopt [7]. To reduce the computation in the next step, we reduce the size of the output heatmap of the model by adding a pooling layer after the third convolution layer. Our whole model is shown in Fig. 2. Details of training this model are described in Section V.

IV. Gait Sequence Representation

Previously, gait sequence was usually represented by GEI. Part of temporal information is lost during the averaging step. To solve the problem, we propose to directly feed the joint headmaps of consecutive frames to LSTM and use the hidden value at the last timestep as the gait sequence representation. The LSTM model is shown in Fig. 3. To train the model, we follow the method in [16], which is an unsupervised method to learn video representations. In the encoding stage, a sequence of inputs are fed into LSTM. In the decoding stage, the objective is to reconstruct the input sequence in the reversed order. Different from [16], we add the dropout regularization [20] after the encoding stage to reduce overfitting.

Our model is able to work in two scenarios: 1) matching scenario; 2) unsupervised scenario. The matching scenario setting is similar to the existing view transformation model

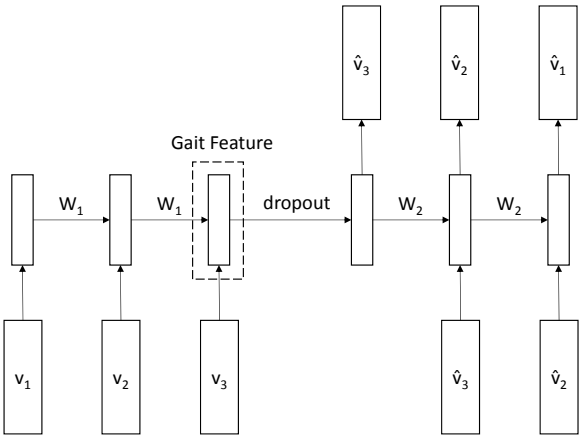

Fig. 3. The gait feature extraction model. The output gait sequence $\hat{v}_1$, $\hat{v}_2$, $\hat{v}_3$ is either the same as the input sequence v_1, v_2, v_3 or the input sequence captured under a different view.

(VTM) setting. In this scenario, two views capturing the same gait sequence are available. One view is regarded as the source view and the other is regarded as the target view. Our model tries to transform gait sequences in the source view to the target view. If the input sequence is from the source view, the output is the corresponding sequence from the target view. If the input sequence is from the target view, the output is itself. The training process can automatically learn the correspondence between the two view of gait sequences without human expert knowledge in gait. In order to guide the model to capture the right transformaiton, we constrain that the sequence pair is synchronized (e.g. in Fig. 3, v_1 and $\hat{v_1}$ are roughly from the same timestamp). This matching scheme works better than the GEI template based VTM because the frame to frame temporal information are fully kept in our method.

In the unsupervised scenario, we simply mix the sequences in both views to train LSTM, and we do not have corresponding gait sequence pairs in both views. By letting the input and output to be the same sequence, the LSTM model can learn how to reconstruct the gait sequence in both views. The LSTM model can learn the common aspects between the two views in this way and thus improve the cross-view gait recognition performance.

V. Experiments

In this section, we first describe how to train the human pose estimation model and how to train the feature extraction model. We then test the performance of the proposed gait feature on the CASIA-B multi-view gait database [21]. The CASIA-B dataset contains the videos of 124 subjects walking under 11 views (0°, 18°, 36°, 54°, 72°, 90°, 108°, 126°, 144°, 162°, 180°). For each view of a subject, there are 6 normal walking sequences, 2 walking with a coat sequences, and 2 walking with a bag sequences. Only the normal walking sequences are used in our experiment. Typically in the literature, this dataset is divided into two groups. The first group containing 24 subjects is used for training the gait recognition

TABLE I
AVERAGED DISTANCE (PX) BETWEEN PREDICATED JOINTS POSITION AND GROUND TRUTH.

Joints	hips	rightleg	rightfoot	leftleg	leftfoot	head	leftarm	leftforearm	lefthand	rightarm	rightforearm	righthand
Distance	6.43	5.76	4.39	7.69	5.08	0.76	0.19	0.27	8.58	3.23	1.48	8.87

model and the remaining 100 subjects are used for evaluating the performance of the model.

A. Training a Human Pose Estimation Model

We use the Human3.6M dataset [22] to train the human pose estimation model. Human3.6M is currently the largest video pose dataset, which contains 15 activity scenarios performed by seven different professional actors. All the activities are captured using four static cameras and Vicon motion capture system. The 2D body joint locations and actor segmentation masks are provided.

Because the video resolution in CASIA-B dataset is relatively low, we cannot estimate the position of small body joints accurately. So we only use 12 body joints out of 32 provided by Human3.6M dataset. The body joints we are using include "hips", "rightleg", "rightfoot", "leftleg", "leftfoot", "head", "leftarm", "leftforearm", "lefthand", "rightarm", "rightforearm" and "righthand". An example heatmap example can be found in Fig. 2. We first train the model described in Section III using the data in Human3.6M dataset directly. After finishing training, we observe that the performance on validation set is good. When testing the model on the CASIA-B dataset, the performance of that model is quite poor. The declination of performance is because the two dataset look very different. To solve this problem, we decided to generate training data ourself. Each generated training sample contains two parts: the foreground and the background. The foreground part is segmented human body from Human3.6M dataset and the background is obtained from CASIA-B dataset. To make the foreground looks more similar to CASIA-B dataset, we change the color image to gray scale and adjust each foreground part by adding a constant so that the mean value of the foreground part is equal to the mean value of human body from CASIA-B dataset. Some examples of the generated training images are showed in Fig. 4. We use 6 subjects for training and 1 subjects for validating. Because Human3.6M is a huge video dataset, we do not use all the frames in it. We crop one frame in every 10 frames from the videos in Human3.6M. There are $187,924$ images for training and $23,141$ images for validating. The size of an input of the CNN is $1 \times 256 \times 256$ and the size of an output of the CNN is $12 \times 32 \times 32$. The averaged pixel distance between predicated joints position and ground truth on validation set is showed in Table I.

B. Training LSTM Model

Before training the LSTM model, we first need to align the CASIA-B dataset because our view matching method requires synchronized data from two views. We first modify the gait period estimation method in [14] to align the videos automatically. The autocorrelation is replaced with convolution to find the phase difference between two gait sequences. When the phase difference is grater than a quarter of the gait period, the automatic method will fail. So we manually check the result and correct the errors. The system is implemented using Theano [23]. We empirically set the number of hidden values in LSTM to be 128 and the number of timesteps to be 16.

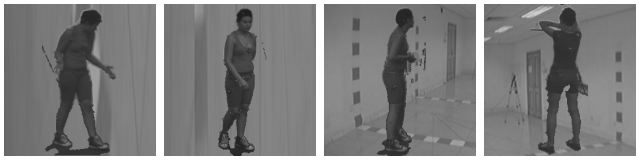

Fig. 4. Examples of generated training images. The foreground is from Human3.6M dataset [22] and background is from CASIA-B dataset [21]

C. Cross-View Gait Recognition

Following [24], one of $0°$, $54°$, $90°$, $126°$ is selected as probe view. For each probe view, one of the remaining 10 views are used as the gallery view. So there are 600 sequences in probe set and 600 sequences in gallery in each testing setup. The identity of a gait sequence in gallery view is known. For a gait sequence in probe view, we find its nearest neighbor in gallery view and regard the nearest neighbor's identity as its identity.

The result of our method under three settings are reported. 1) In the no training setting, we random initialize the weights in LSTM and use the random initialized model to extract gait features. The distance between two gait sequences is calculated based on the extracted features. To reduce the noise, we calculate the distance for five times using different random initialized model and average the obtained distances. The averaged distance is used in nearest neighbor classification. 2) In the unsupervised setting, we train a LSTM model for each pair of gallery view and probe view combination. The training gait sequences from both the gallery and probe views are used to reconstruct the itself. 3) In the matching setting, one of the gallery view and probe view is regarded as the source view and the other is regarded as the target view. The training gait sequences in the source view are transformed to the target view and the training gait sequences in the target view are also used to reconstruct itself. Because gait can be viewed more clear under $90°$, the gallery view or the probe view which is closer to $90°$ is chosen as the target view. We compare our method with the following four methods: 1) the baseline method [21]; 2) Motion co-clustering method in [24]; 3) Appearance Conversion Machine (ACM) [11] and 4) C3A method [25].

To fully use all the information in a sequence, we calculate the distance of two gait sequences in the following way.

Each sequence in the CASIA-B dataset is usually several times longer than 16, so we cut a whole gait sequence into several short sequences of length 16. The offset between two adjacent sequences is 1 frame. A feature vector is extracted for each short sequence using the LSTM model. Let $G = \{\boldsymbol{g}_i | i = 1, \ldots, n_g\}$ be the feature vectors extracted from a gait sequence in gallery set and $P = \{\boldsymbol{p}_i | i = 1, \ldots, n_p\}$ be the features vectors extracted from a gait sequence in the probe set. The distance between these two sequences is calculated by

$$DIST(P, G) = \sum_{i=1}^{n_p} \min_j d(\boldsymbol{p}_i, \boldsymbol{g}_j), \quad (1)$$

where $d(\cdot, \cdot)$ is cosine distance.

Recognition results are shown in Fig. 5. Detailed result values can be found in the supplementary material. To make the results more appreciable, we also use Table II to compare the results. We divide views into front side (i.e. 0°-72°) and back side (i.e. 108°-180°). The results within one side are averaged in Table II. From these results, we can observe that

1. Our LSTM model with random initialized weights works quite well when the gallery view and probe view are similar, i.e. the angle difference is 18°. But it is worse than Motion co-clustering and ACM when one of gallery and probe set is from the front side and the other is from the back side. For example, the accuracy of ACM and Motion co-clustering is much higher than our method when the gallery is 180° and probe is 0°. The good results of GEI based methods are because GEI under these two views are similar. But there is quite large difference between the heatmap sequences from the front side and back side. So the random initialized model works poorly.
2. For most of times, the recognition accuracy increases after unsupervised training, which shows that unlabeled data can be leveraged by our method.
3. Our matching method achieves the best results when the probe view is 90°. It also achieves the best results when the gallery view and probe view are both front side or back side in other three probe views. Because the heatmap of synchronized frames under a font side view and a back side view are quite different, our model needs more training pairs to learn the transformation. The performance of the matching model is much better than random initialized model when one of the gallery view and probe is from front size and the other one is from back side, but it cannot get the best accuracy only trained with gait sequences of 24 subjects. We firmly believe our model can achieve better results when more data are available.

VI. Conclusion

In this paper, we present a novel feature learning method for gait recognition. A CNN based pose estimation method is used to extract body joint heatmap for each frame. LSTM is then used to model the high level motion feature in the heatmap sequence. Our model has the advantage that it can be trained with unlabeled data and it performs view transformation at frame level. Comprehensive experiments on the CASIA-B datasets demonstrate the effectiveness of our method for cross-view gait recognition.

Currently, our gait feature is invariant across two views. We will try to make it handle more views in the future.

TABLE II
Averaged accuracy of cross-view gait recognition using different methods. "-" means that the corresponding value is not reported in [25].

Probe view	0°		54°	
Gallery view	Front	Back	Front	Back
Baseline [21]	7.8	12	16.3	15.8
Motion co-clustering [24]	45.8	50.4	70.3	36.0
ACM [11]	58.8	**57.6**	75.3	**53.2**
C3A [25]	-	-	71.8	40.6
No training	57.5	17.5	68.3	24.3
Unsupervised training	57.0	29.7	72.2	39.4
Matching	**63.6**	40.8	**83.8**	43.9

Probe view	90°		126°	
Gallery view	Front	Back	Front	Back
Baseline [21]	22.6	22.8	14	21.5
Motion co-clustering [24]	49	47.6	39.6	73.3
ACM [11]	56.4	57.2	**50.4**	78.5
C3A [25]	55.8	53.6	42.4	72.3
No training	43.6	40.8	33.8	65.0
Unsupervised training	49.9	46.7	44.7	67.7
Matching	**60.0**	**60.0**	41.8	**81.9**

Acknowledgement

This work was supported in part by New York State through the Goergen Institute for Data Science at the University of Rochester. We thank VisualDX for discussions related to this work.

References

[1] R. Bodor, A. Drenner, D. Fehr, O. Masoud, and N. Papanikolopoulos, "View-independent human motion classification using image-based reconstruction," *Image and Vision Computing*, vol. 27, no. 8, pp. 1194–1206, 2009. 1, 2

[2] M. Goffredo, I. Bouchrika, J. N. Carter, and M. S. Nixon, "Self-calibrating view-invariant gait biometrics," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 40, no. 4, pp. 997–1008, 2010. 1, 2

[3] S. Zheng, J. Zhang, K. Huang, R. He, and T. Tan, "Robust view transformation model for gait recognition," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*. IEEE, 2011, pp. 2073–2076. 1, 2

[4] J. Han and B. Bhanu, "Individual recognition using gait energy image," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 2, pp. 316–322, 2006. 1

[5] C. Wang, J. Zhang, L. Wang, J. Pu, and X. Yuan, "Human identification using temporal information preserving gait template," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 11, pp. 2164–2176, 2012. 1

[6] F. Castro, M. Marín-Jiménez, N. Guil, and R. Muñoz-Salinas, "Fisher motion descriptor for multiview gait recognition," *arXiv preprint arXiv:1601.06931*, 2016. 1

[7] T. Pfister, J. Charles, and A. Zisserman, "Flowing convnets for human pose estimation in videos," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1913–1921. 1, 2, 3

[8] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997. 1

[9] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6645–6649. 1

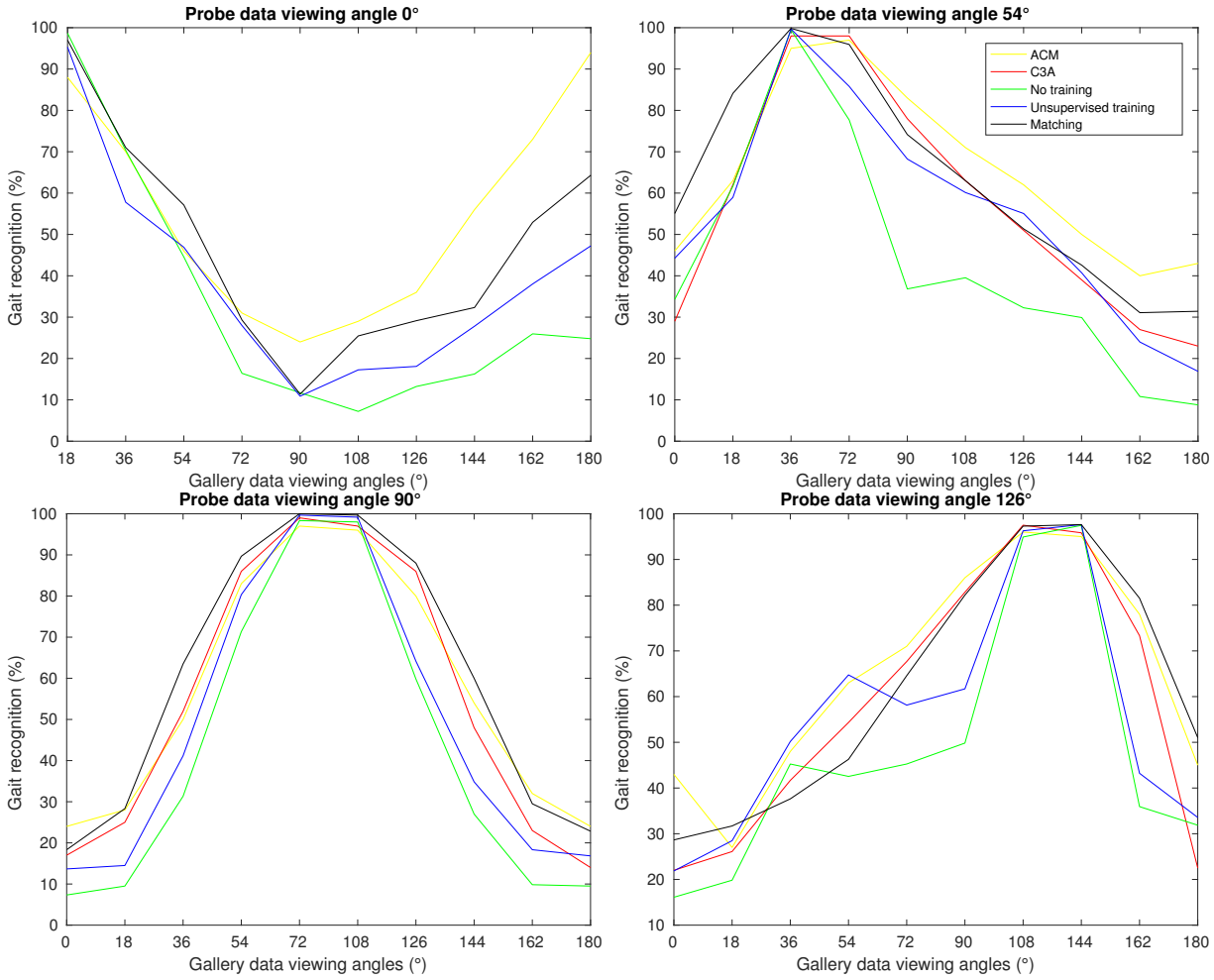

Fig. 5. Accuracy of cross-view gait recognition using different methods. Only the most competitive methods are shown for clarity.

[10] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112. 1

[11] X. Zhao, Y. Jiang, T. Stathaki, and H. Zhang, "Gait recognition method for arbitrary straight walking paths using appearance conversion machine," *Neurocomputing*, vol. 173, pp. 530–540, 2016. 2, 4, 5

[12] G. Zhao, G. Liu, H. Li, and M. Pietikäinen, "3d gait recognition using multiple cameras," in *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*. IEEE, 2006, pp. 529–534. 2

[13] A. Kale, A. K. R. Chowdhury, and R. Chellappa, "Towards a view invariant gait recognition algorithm," in *Advanced Video and Signal Based Surveillance, 2003. Proceedings. IEEE Conference on*. IEEE, 2003, pp. 143–150. 2

[14] W. Kusakunniran, Q. Wu, H. Li, and J. Zhang, "Multiple views gait recognition using view transformation model based on optimized gait energy image," in *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*. IEEE, 2009, pp. 1058–1064. 2, 4

[15] W. Kusakunniran, Q. Wu, J. Zhang, and H. Li, "Support vector regression for multi-view gait recognition based on local motion feature selection," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 974–981. 2

[16] N. Srivastava, E. Mansimov, and R. Salakhutdinov, "Unsupervised learning of video representations using lstms," *arXiv preprint arXiv:1502.04681*, 2015. 2, 3

[17] C. Yan, B. Zhang, and F. Coenen, "Multi-attributes gait identification by convolutional neural networks," in *2015 8th International Congress on Image and Signal Processing (CISP)*. IEEE, 2015, pp. 642–647. 2

[18] Z. Wu, Y. Huang, and L. Wang, "Learning representative deep features for image set analysis," *Multimedia, IEEE Transactions on*, vol. 17, no. 11, pp. 1960–1968, 2015. 2

[19] F. Castro, M. Marin-Jimenez, N. Guil, and N. P. de la Blanca, "Automatic learning of gait signatures for people identification," *arXiv preprint arXiv:1603.01006*, 2016. 2

[20] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," *arXiv preprint arXiv:1409.2329*, 2014. 3

[21] S. Yu, D. Tan, and T. Tan, "A framework for evaluating the effect of view angle, clothing and carrying condition on gait recognition," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, vol. 4. IEEE, 2006, pp. 441–444. 3, 4, 5

[22] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 7, pp. 1325–1339, 2014. 4

[23] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio, "Theano: new features and speed improvements," Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop, 2012. 4

[24] W. Kusakunniran, Q. Wu, J. Zhang, H. Li, and L. Wang, "Recognizing gaits across views through correlated motion co-clustering," *Image Processing, IEEE Transactions on*, vol. 23, no. 2, pp. 696–709, 2014. 4, 5

[25] X. Xing, K. Wang, T. Yan, and Z. Lv, "Complete canonical correlation analysis with application to multi-view gait recognition," *Pattern Recognition*, vol. 50, pp. 107–117, 2016. 4, 5