CHAPTER 5

Dynamic Gait Energy Image

Although the *GAEI*-based gait recognition approach improves over the accuracy of the Pose-based *BEI* and other existing pose-based approaches, its response time is high due to which it is not very suitable for application in real-life surveillance scenarios. Another possible way to improve the shortcomings of the existing pose-based features is relaxing the constraints associated with the state transition model used in the previous chapters (refer to Fig. 3.7), while using a single key pose set instead of a dictionary of key pose sets. While mapping the frames of a sequence to a key pose set, instead of making transitions from a state to only the next state (as shown in Fig. 3.7), we propose to allow transitions to the next state as well as to the next to next state. This modification in the state transition rule appears to be beneficial in effectively handling sequences with varying walking speeds and due to making use of a single key pose set, it is expected to work in a more time-efficient manner compared to the method used in Chapter 2. Additionally, we develop an automated approach using Generative Adversarial Networks (GANs) to perform gait recognition satisfactorily even in the presence of co-variate conditions. First, we determine a set of generic unique poses in a gait cycle, following which we compute gait features corresponding to these poses, which we term as the *Dynamic Gait Energy Image (DGEI)*. Next, a Generative Adversarial Network (GAN) model is employed to predict the corresponding *DGEI* images without

the co-variate objects. These final gait features are readily comparable with the gallery sequences, and hence, recognition is performed using the GAN-generated *DGEI* images. The main contributions of the work are as follows:

- We develop a new pose-based gait recognition approach that can handle co-variate conditions in gait sequences effectively.
- Binary silhouettes of an input sequence are mapped to a set of unique walking poses following the transition rules of a state transition model with relaxed constraints.
- Computation of GEI features corresponding to each unique pose and GAN-based elimination of co-variate conditions from the derived gait features at the granularity of unique walking poses.

5.1 Overview

The gait recognition algorithm proposed in this chapter consists of the following major steps: (a) maximal unique pose set construction, (b) gait feature extraction, and (c) co-variate condition removal, and (d) comparison with the gallery set for predicting the class of an input test subject. A flowchart of the sequence of steps involved in this work has been shown in Fig. 5.1. With reference to the block

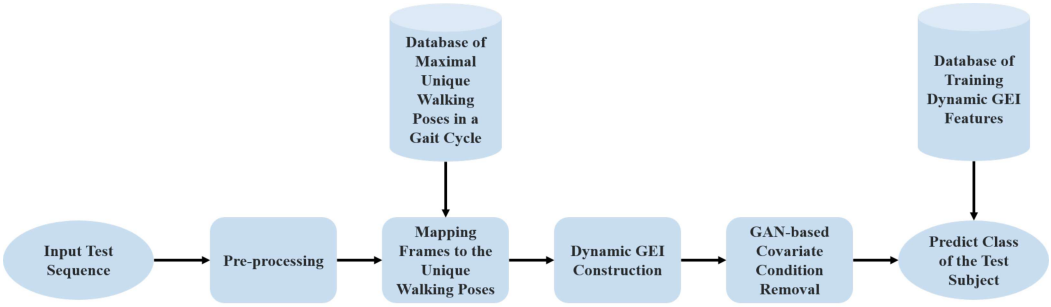


Figure 5.1: A flow chart of the proposed approach

diagram, given an RGB sequence, standard pre-processing operations, as discussed in the previous chapters, are applied to each frame of the sequence to generate the

corresponding cropped and normalized frame. The other individual blocks in the block diagram are explained in further detail in the subsequent sub-sections.

5.1.1 Determination of Maximal Unique Walking Poses

We determine the maximal unique walking poses in a gait cycle from the fronto-parallel view. From a given gait data set, we first select all cycles with the longest length (i.e., with the maximum number of frames). The selection of sequences with maximum length guarantees the availability of maximum possible unique key poses in a gait cycle. The difference of our approach of maximal unique pose extraction from that of the key pose extraction method used in our previous chapters is that in the previous methods the objective was to determine an optimal number of non-overlapping poses termed as *key poses* in a gait cycle by applying constrained K-Means clustering technique. The limitations of K-Means clustering are that it requires specification of the value of K, and it can result in spherical clusters only. In contrast, our approach is capable of obtaining the maximum possible walking poses in a gait cycle that can preserve the dynamics of gait at a high resolution. Once the set of maximal unique walking poses is extracted, we compute the gait features by mapping the frames of an input sequence to the appropriate walking poses using relaxed constraints, as discussed next.

5.1.2 Mapping of Frames to the Unique Poses and *DGEI* Feature Construction

Given an input sequence, we construct gait features at the granularity of the already computed key pose set. First, each frame of the sequence is mapped to the appropriate key pose and next silhouettes mapped to the same key pose are averaged to generate the final *DGEI* feature. The relaxed temporal constraints imposed using a state transition model are shown in Fig. 5.2. As already discussed in the previous chapters, the number of states in this state transitional model must be equal to the number of frames present in the set of maximal unique walking poses. But for ease of presentation, we have shown the diagram with six states only.

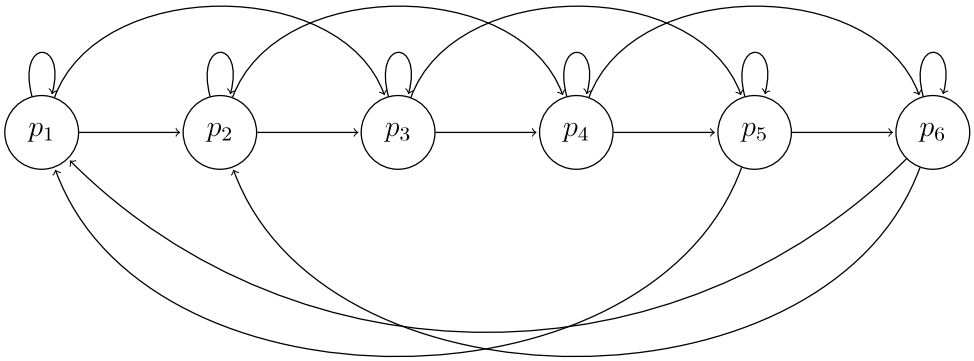


Figure 5.2: State transition model with six states representing six unique poses in a gait cycle

With reference to the figure, each state (i.e., p_1, p_2, ... p_6) represents an unique pose and the arrows show the direction of transition from one state to the other. If K is used to denote the number of states, the transition from any state p_i can be made to either the states p_i, p_j, or p_k, where

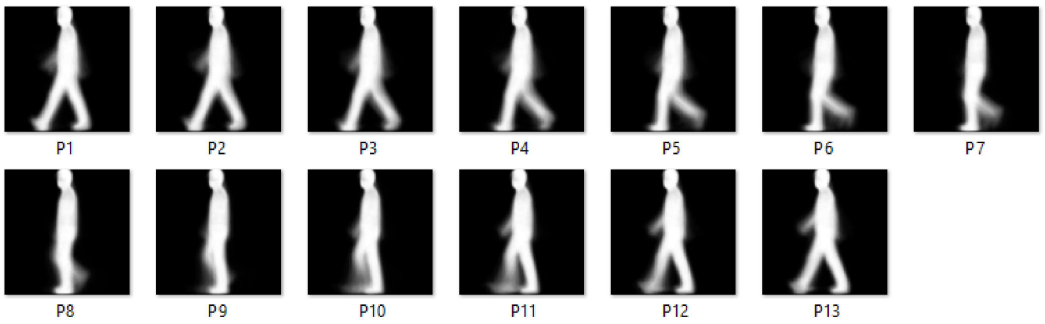
$$p_j = \begin{cases} p_{i+1}, & \text{if } i < K \\ p_1, & \text{if } i = K, \end{cases} \tag{5.1}$$

and

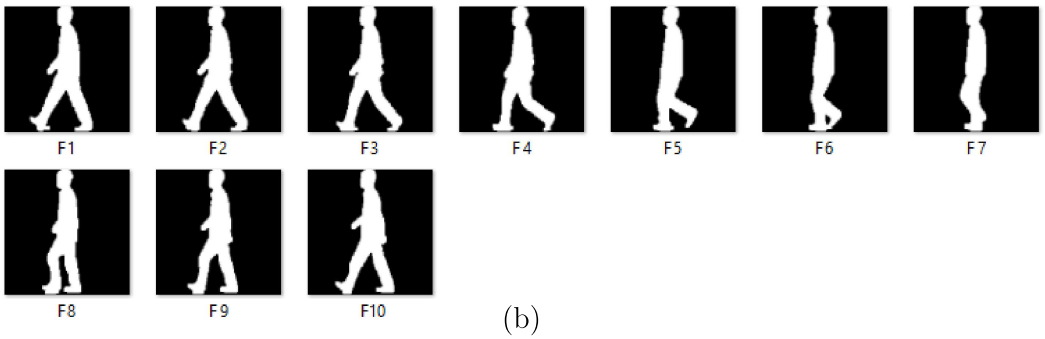
$$p_k = \begin{cases} p_{j+1}, & \text{if } j < K \\ p_1, & \text{if } j = K, \end{cases} \tag{5.2}$$

Our previous two chapters also consider a similar state transition model for mapping the frames to the appropriate key poses. However, the temporal constraints imposed on that model are too strict since it allows consecutive frames to either get mapped to the same state or consecutive states. Hence, this method would work only in situations where the length of an incoming gait sequence is greater than the number of frames in the set of key poses. In contrast, here we relax the constraint by allowing an additional transition from each state to its next-to-next state. This would allow better mapping of frames to key poses in the event of minor changes in walking speed, frame rate, etc. Also, our work will perform the mapping much better in case the number of frames in the input sequence is less

than the number of frames present in the set of unique walking poses. The above fact has been explained with the help of Fig. 5.3(b) and Table 5.1. Fig. 5.3(b) shows an input sequence $\{F_1, F_2, ..., F_{10}\}$ with 10 frames which is less than 13, i.e., the number of frames in the set of unique walking poses.



(a)



(b)

Figure 5.3: (a) Set of maximal unique walking poses derived using the approach given in Section 5.1.1, and (b) an input gait sequence with the number of frames less than the number of frames in the set of maximal unique walking poses

Visually, we can observe that frames F_1 and F_2 are similar to pose p_1, frame F_3 is similar to pose p_3, and so on. It may also be noted that all the poses in the set of maximal unique walking poses are not present in the input gait sequence. This ideal mapping of frames to the set of maximal unique walking poses has been shown in Table 5.1. However, the application of the approach as described in our previous chapters and also in [1] would have failed to perform this mapping correctly since it constrains the frames to get mapped to only consecutive key poses.

Table 5.1: Ideal mapping of the frames of the sequence in Fig. 5.3(b) to the set of maximal unique walking poses

Frame Index	Ideal Mapped Pose	Frame Index	Ideal Mapped Pose
F_1	p_2	F_6	p_7
F_2	p_2	F_7	p_9
F_3	p_3	F_8	p_{11}
F_4	p_4	F_9	p_{12}
F_5	p_6	F_{10}	p_{13}

5.1.3 Construction of Gait Features

Once each frame of a sequence is mapped to the frames in the set of maximal unique walking poses, we proceed to compute the gait features. Frames of a sequence that are mapped to the same pose in the set of maximal unique walking poses are averaged to form the gait feature corresponding to that pose. We term this feature as *Dynamic Gait Energy Image* (*DGEI*) since it preserves the dynamics of gait at a higher resolution than GEI. If t frames F_{p_j}, $F_{p_{j+1}}$, ..., $F_{p_{j+t-1}}$ of an input sequence are mapped to pose p_i in the set of maximal unique walking poses, then lest us denote the *DGEI* feature for pose p_i as $\mathcal{D}_i$. If no frames of an input sequence are mapped to a certain pose in the set of maximal unique walking poses, then the *DGEI* feature for that pose is simply a vector of *0*s. The *DGEI* features (or, images) computed using the above expression for three sequences with different co-variate conditions, namely, carrying bag, wearing coat, and normal walking, are shown respectively in the first, second, and third rows of Fig. 5.4. It can be verified from the figure that for each sequence, the *DGEI* features for certain poses are not available. This is since no frames of the input sequence have been mapped into the respective poses as per the constraints imposed on the state transition model (refer to Fig. 5.2). It can also be observed from the figure that the *DGEI* images are influenced by co-variate factors, which might affect the gait recognition performance. To eliminate the co-variate objects automatically and facilitate a fair comparison of test gait features with those in the gallery set, we make use of a Generative Adversarial Network that translates any *DGEI* image (with/without co-variate objects) into the corresponding *DGEI* image without

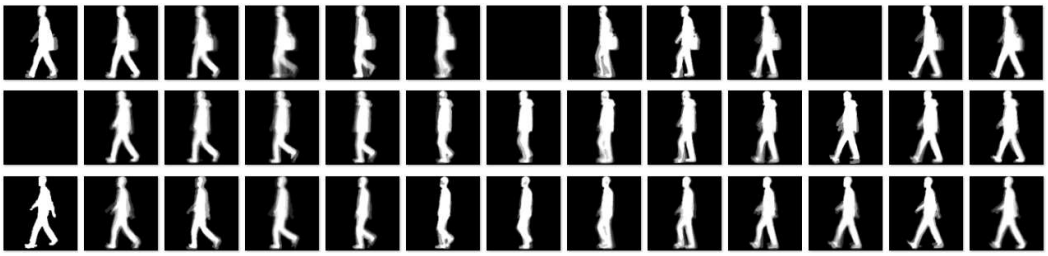


Figure 5.4: *DGEI* features corresponding to the 13 unique walking poses as shown in Fig. 5.3(a) for three different co-variate conditions, namely, (i) carrying bag, (ii) wearing coat, and (iii) normal sequences without any co-variate objects

any co-variate objects.

5.1.4 GAN-based Removal of Co-variate Objects

Conditional Generative Adversarial Networks (cGANs) [117, 118] are Deep Learning models capable of performing pixel-level domain transfer from an input image to a target image. A type of GAN that has been widely used in the recent past for carrying out various types of image translation tasks is the Pix2Pix GAN [117]. As in any GAN architecture, the Pix2Pix GAN also consists of two parts: (i) a generator which is essentially a Convolutional Autoencoder that generates the desired output images, and (ii) a discriminator that predicts if a generated image looks realistic or fake. In this work, we also employ the Pix2Pix GAN to remove the co-variate objects from the *DGEI* images. The GAN takes as input a random noise vector z and an image x as a condition to generate output y. The detailed architecture of the encoder of the Pix2Pix GAN is shown in Table 5.2. The decoder configuration is simply the reverse of that used for the encoder. The architecture detail of the discriminator with four convolution layers is shown in Table 5.3.

Let us use x to denote the *DGEI* image with co-variate objects which provides the GAN with concrete information about the shape of a person, z to denote the random noise vector, and y to denote the desired output (i.e., the image of the

Table 5.2: Encoder parameters of Pix2Pix GAN

Layers	No. of filters	Kernel Size	Stride	Batch Norm	Activation
Conv.1	16	2x2	2	No	LeakyReLU
Conv.2	32	2x2	2	Yes	LeakyReLU
Conv.3	64	2x2	2	Yes	LeakyReLU
Conv.4	128	2x2	2	Yes	LeakyReLU
Conv.5	256	2x2	2	Yes	LeakyReLU
Conv.6	512	2x2	2	Yes	LeakyReLU

Table 5.3: Parameters of real/fake-discriminator of Pix2Pix GAN

Layers	No. of filters	Kernel Size	Stride	Batch Norm	Activation
Conv.1	64	4x4	2	Yes	LeakyReLU
Conv.2	128	4x4	2	Yes	LeakyReLU
Conv.3	256	4x4	2	Yes	LeakyReLU
Conv.4	512	4x4	2	Yes	LeakyReLU

same person without co-variate objects). The function G learned by the generator can thus be represented as $G : \{x, z\} \rightarrow y$. The generator improves its prediction in multiple epochs and finally learns to generate visually good quality images without co-variate objects. The standard loss function for training the cGAN is the min-max loss function and is given by:

$$\mathcal{L}_{cGAN}(G, D) = \mathcal{E}_{x,y}[logD(x, y)] + \mathcal{E}_{x,z}[log(1 - D(x, G(x, z)))]. \tag{5.3}$$

The generator tries to minimize the above function, i.e., $\mathcal{L}_{cGAN}(G, D)$, while the discriminator attempts to maximize it. While the discriminator is being trained, in every epoch it improves its capability to differentiate between the real data and the fake data, i.e., that produced by the generator. A penalty is given to the discriminator for misclassifying any real or fake instance using the expression in (5.3). Here, $log(D(x, y))$ refers to the probability that the discriminator correctly classifies the real image and maximizing $log(1 - D(G(z)))$ helps the discriminator in properly labeling the fake image produced from the generator. The generator,

on the other hand, gets rewarded if the discriminator prediction goes wrong and penalized if the discriminator prediction is correct. This is done by minimizing the second term in (5.3.

In addition to $\mathcal{L}_{cGAN}$, we also consider the $L1$ loss between the generator predicted image $G(x, z)$ and the ground-truth image y. A smaller value of the $L1$ loss indicates a higher pixel-wise similarity between the generated image $G(x, z)$ and the ground-truth image y. It may be noted that the binary silhouette images extracted from the RGB frames may contain some noisy pixels due to shadows, or similarity between the foreground and background appearances. Hence, instead of $L2$ loss, we use the $L1$ loss here to put less penalty on pixel-wise mis-predictions. The expression for this loss is shown in (5.4).

$$\mathcal{L}_{L1}(G) = \mathcal{E}_{x,y,z}(|y - G(x, z)|). \tag{5.4}$$

The training is terminated when the generator starts producing images closely similar to the ground-truth. The complete loss function G^* to train the generator is obtained by adding the above-mentioned two loss functions as shown in (5.5).

$$\mathcal{G}^* = arg\ _{min_G}\ (_{max_D}\ \mathcal{L}_{cGAN} + \lambda \mathcal{L}_{L1}(G)). \tag{5.5}$$

In the above equation, λ is a constant term that controls the effect of the $L1$ loss function on the final objective. While $\mathcal{L}_{cGAN}$ trains the generator in outputting realistic images, $\mathcal{L}_{L1}$ helps in refining the output from the generator further by minimizing the pixel-level differences.

We train the GAN to predict *DGEI* for normal walking from a *DGEI* with any other co-variate conditions. For this, we have taken 30 subjects to make the pairs for different translation conditions, namely normal to normal, carry-bag to normal, and wearing-coat to normal. Corresponding to each condition, we have 26 input-output pairs to train the GAN for each subject, resulting in a total of 780 input-output pairs for the 30 subjects for all the different translation conditions. The training of the Pix2Pix GAN with the above input-output gallery is explained with the help of Fig. 5.5. In our experiments, we have used Adam optimizer with a learning rate 0.0002, and a decay rate of 0.5 to train the above model.

With reference to the figure, the input to the GAN can be either *DGEI* without

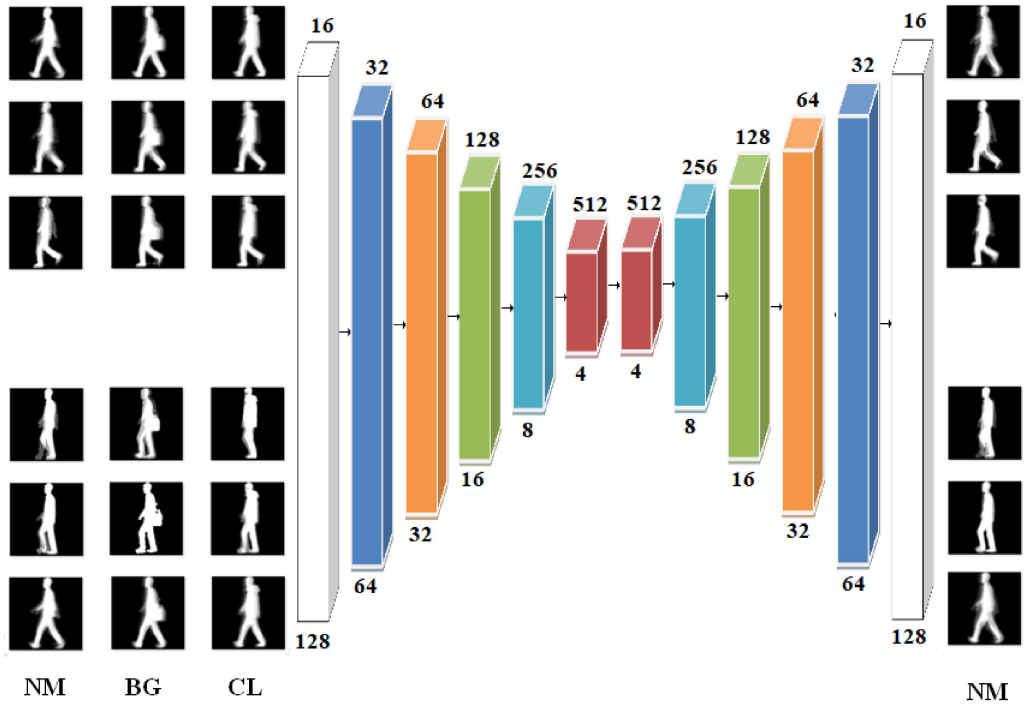


Figure 5.5: Different possible input-output combinations of the Pix2Pix GAN used for Co-variate Object Removal from *DGEI*s while working with the CASIA B Data

co-variate objects (shown as Column NM), or with co-variate objects such as carrying bag (shown as Column BG), or wearing coat (shown as column CL), and the output is the corresponding *DGEI* without co-variate objects (shown as NM).

An important point to note here is that during testing, an input sequence may appear either with or without co-variate objects. In case, the input sequence is devoid of co-variate objects, the *DGEI* features computed using the method discussed in Section 5.1.3 must be used for recognition, while the subsequent GAN-based co-variate object removal stage is not necessary. To eliminate the need for detecting if co-variate objects are present in a sequence, and effectively handle any type of walking sequences, training the GAN with NM-NM pair (i.e., the *DGEI* of normal walking both at the input and at the output) is required along with

BG-NM (i.e., *DGEI* with carrying bag to *DGEI* with normal walking) and *CL-NM* (*DGEI* with wearing coat to *DGEI* with normal walking). The discriminator is trained to predict whether an input image is real or fake. If the input *DGEI* to the discriminator is from real gait data corresponding to normal walking, the discriminator will output *1*, otherwise, it will output *0*.

5.1.5 Classification of GAN-Generated Features

Once co-variate objects are eliminated using the Pix2Pix GAN, we obtain clean *DGEI* features that can then be used in the subsequent comparison phase. Consider a gallery of M subjects, and further let the *DGEI* features (after GAN-based co-variate object removal) computed for these gallery subjects be represented by $\mathcal{D}^1$, $\mathcal{D}^2$, ..., $\mathcal{D}^M$, respectively. Here, $\mathcal{D}^i = \{\mathcal{D}^i_1, \mathcal{D}^i_2, ..., \mathcal{D}^i_K\}$, where K is the total number of frames present in the set of maximal unique walking poses. Since the dimensionality of the *DGEI* features is high, we apply PCA to reduce the feature dimension by retaining 95% variance. Let these reduced features for the M subjects be denoted by $\mathcal{D}^{1_R}$, $\mathcal{D}^{2_R}$, ..., $\mathcal{D}^{M_R}$, respectively. Next, consider an input test sequence, and the *DGEI* features obtained from this sequence after GAN-based processing and PCA-based dimensionality reduction by given by $\mathcal{D}^{T_R}$, where $\mathcal{D}^{T_R} = \{\mathcal{D}^{T_R}_1, \mathcal{D}^{T_R}_2, ..., \mathcal{D}^{T_R}_K\}$. The feature $\mathcal{D}^{T_R}$ is compared with each of the features derived from the M subjects, i.e., with $\mathcal{D}^{1_R}$, $\mathcal{D}^{2_R}$, ..., $\mathcal{D}^{M_R}$, and next the class of the input sequence is predicted using a Linear Discriminant Analysis (LDA) classifier.

We carry out pose-level prediction using an LDA corresponding to each pose in the set of maximal unique walking poses, and next aggregate the responses given by the different classifiers to predict the final class of a test subject. Let $\mathcal{P}^m_j$ denotes the probability that the *DGEI* feature of the test subject for the j^{th} key pose is mapped to the m^{th} subject. Similarly, we compute the probabilities for all the poses into which the frames of the test sequence got mapped, and next average these probabilities to obtain a metric $\mathcal{P}^m$ representing the similarity of the test subject with respect to the m^{th} gallery subject. The test subject is assigned to class $\mathcal{C}$ if

$$P^{\mathcal{C}} > P^m \forall m = 1, 2, ..., M, m \neq \mathcal{C}. \tag{5.6}$$

5.2 Experiments and Analysis

Our experiments have been carried out using the same system as described in Section 3.1.3 of Chapter 3. Since in this work we focus on gait recognition under varying co-variate conditions, we choose three gait data sets containing walking sequences of subjects under different co-variate conditions to evaluate the proposed approach, namely, the CASIA B [2], TUM-GAID [3], and OU-ISIR TreadMill Dataset B [103]. We consider 13 unique poses in a gait cycle for each of the CASIA B and TUM-GAID data sets, and 37 poses for the OU-ISIR TreadMill B data set.

5.2.1 Analysis using CASIA B Data Set

To evaluate the effectiveness of the proposed gait recognition approach, we consider three different *gallery-probe* combinations, denoted as C_1, C_4, C_5, as shown in Table 2.1. It may also be noted from the table that the training set consists of four gait cycles for each person, and hence we can construct four different training samples for each subject which can be used to train the LDA model (as discussed in Section 5.1.5).

To remove the effect of co-variate objects from the DGEI features, the GAN model (discussed in Section 5.1.4) is trained using an extensive dataset for which the input data corresponds to the pose-based *DGEI* features with co-variate objects and the ground-truth data corresponds to the corresponding *DGEI* features without co-variate objects. From the training set of the CASIA-B data, we consider 30 subjects and compute the *DGEI* features from each of these subjects for each of the different conditions: i.e., carrying bag, wearing coat, normal walking. Each of these features is paired with a similar feature extracted from a normal walking sequence of the same person and a collection of the above set of features forms the feature set for training the GAN model. The GAN is trained in multiple epochs with a batch size of 4 till 200 epochs where the trained models have been saved from 101 to 200 epochs. To select a suitable one from the saved GAN models, we have considered the next 10 subjects and passed each frame of *DGEI* feature from each trained GAN model and compute the dice score and correlation coefficients by comparing with their normal walking *DGEI* frame. Figure 5.6 shows a plot

of the above dice score and correlation scores obtained for the different saved models. It can be seen from the figure that the trained model with index 146 corresponds to the highest dice score with a sharp peak, and closer to this point, indices 144 and 149 also show quite high correlation scores. To select a suitable model, we have generated the images from each model saved after 144, 146, and 149 epochs and obtained the average accuracy by running the recognition model after removing the co-variates. These values are found to be 87.97%, 88.18%, and 87.60%, respectively. Since the model with index 146 provides a better accuracy than the others, it has been chosen as the optimal GAN model.

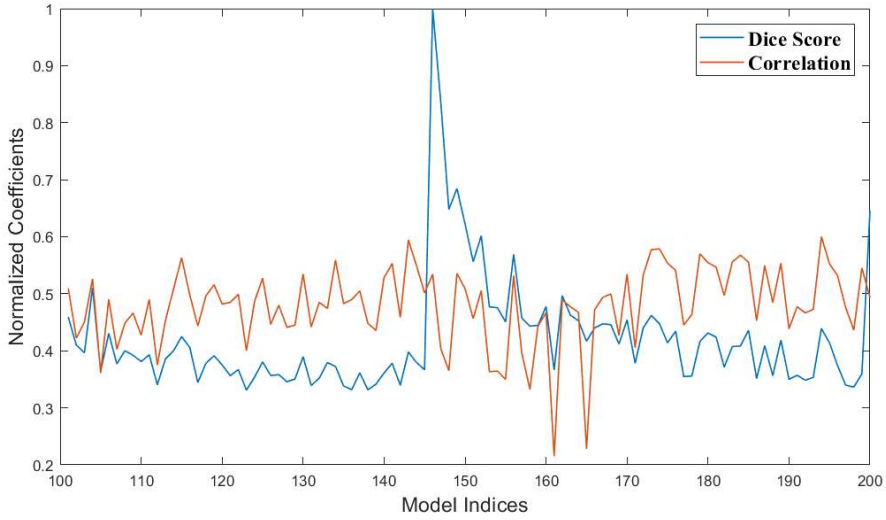
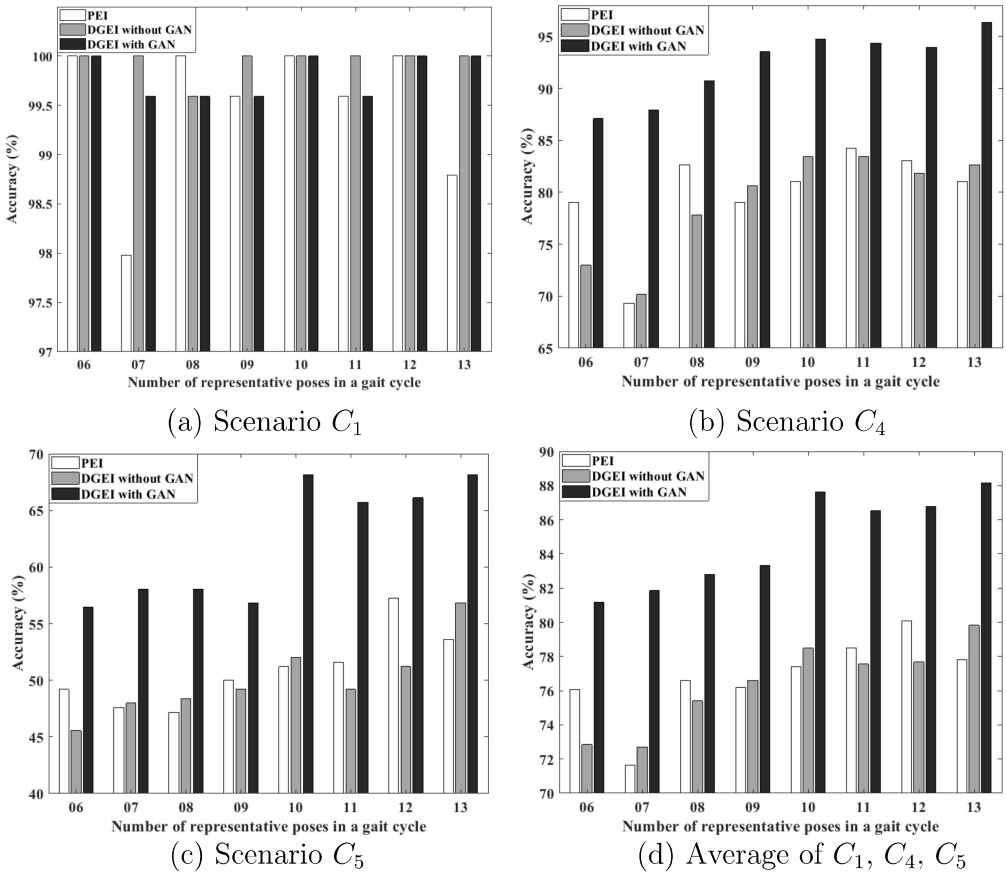


Figure 5.6: Normalized coefficients for pose frame comparison analysis with generated and actual images to select the GAN model on CASIA B dataset

In our first experiment, we evaluate the effectiveness of our approach in the presence of the different co-variate conditions present in the CASIA B data, as discussed in Table 2.1. Since the work in [1] is also a popular pose-based gait recognition approach, we also compare the results of our approach in the presence of different co-variate scenarios with that obtained using [1]. Results are shown in Figs. 5.7(a)-(c) in terms of grouped bar charts for the different numbers of representative poses in a gait cycle (i.e., 6 to 13). In each plot, groups of bars are shown for the different number of representative poses where the first bar corresponds to

the accuracy obtained using [1], while the second and third bars correspond to the accuracy obtained using our approach without the GAN and with the GAN-based co-variate object removal module, and the height of each bar corresponds to the gait recognition accuracy.

It can be seen from Fig. 5.7 that *DGEI* usually has a higher gait recognition accuracy as compared to PEI [1] for any number of representative poses in a gait cycle. This is due to computing gait features from the maximal number of unique poses in a gait cycle that preserves the dynamics of gait at a higher resolution compared to the key pose-based gait feature extraction method given in [1]. The effectiveness of GAN-based co-variate object removal can also be visualized from Figs. 5.7(b) and (c). In each case, GAN-based *DGEI* image refinement by removing co-variate conditions like carrying bags and wearing coats leads to a significant improvement in the gait recognition accuracy. Only if the input test sequence corresponds to normal walking, use of GAN-based *DGEI* image processing sometimes reduces the recognition accuracy by a very small extent. However, for any number of representative poses in a gait cycle, the gait recognition accuracy for normal walking is greater than or equal to 99.5% using *DGEI* with GAN. Because an input gait sequence may come in any form (i.e., with or without co-variate objects), the use of the proposed GAN-based co-variate object removal technique from *DGEI* is highly recommended. Fig. 5.7(d) shows the average accuracy obtained from all the co-variate conditions, namely C_1, C_4, and C_5 (refer to Table 2.1). Once again it can be observed that, on average, the proposed method performs with a higher level of accuracy than [1], which is due to relaxing the constraints of the state transition model for better and more realistic mapping of binary silhouette frames to key poses. The performance of the present approach without GAN-based processing stage. Also, using 13 unique poses in a gait cycle, a higher recognition rate is observed, in general, compared to any other number of unique poses in a gait cycle.
We next perform a comparative performance analysis of our approach with other popular state-of-the-art gait recognition techniques, namely, [1, 31, 50, 51, 69, 88, 91, 94, 95, 119] and the approaches described in the previous two chapters, in terms of *Rank 1* recognition accuracy, and present the results in Table 5.4. In



(a) Scenario C_1

(b) Scenario C_4

(c) Scenario C_5

(d) Average of C_1, C_4, C_5

Figure 5.7: Comparative performance analysis in terms of overall percentage accuracy of [1], *DGEI* and *DGEI* after GAN-based co-variate object removal for (a) scenario C_1, (b) scenario C_4, (c) scenario C_5, and (d) average performance corresponding to the three scenarios using CASIA B data

the same table, we also present results showing the efficacy of the proposed GAN-based co-variate object removal phase (as discussed in Sections 5.1.4) when applied to [31] and [1], as well as the accuracy of our approach without the GAN-based processing stage. The gallery and test set combinations for gait recognition are already specified in Table 2.1, and the Pix2Pix GAN model for co-variate object removal is trained using the gallery set (corresponding to scenarios C_1, C_4, and C_5) for the first 30 subjects only. It can be seen from the table that our proposed

Table 5.4: Table showing a comparative study of our approach with state-of-the-art gait recognition techniques using CASIA B Data. (*In each column, the bold number represents the maximum accuracy for the corresponding scenario*)

Approach Name	C_1(%)	C_4(%)	C_5(%)	Mean (%)
Gait Energy Image (GEI) [31]	99.59	65.44	37.39	67.47
Active Energy Image (AEI) [51]	99.59	88.61	45.93	78.04
Deep-CNN [88] with batch size 16	97.15	35.77	19.51	50.81
GEINet [91]	95.12	57.72	19.91	57.58
DTW [69]	94.71	56.50	27.64	59.61
MGANs [50]	99.59	91.00	48.00	79.53
GaitSet [94] with Large Training(LT)	91.70	81.00	**70.10**	80.93
GaitGAN [95]	98.39	64.52	48.39	70.43
PEI with 13 key poses [1]	98.79	81.04	53.62	77.81
Pose based BEI with 6 poses	**99.59**	92.33	57.25	83.05
***GAEI* [119]**	98.78	90.24	66.66	85.22
***DGEI* without GAN**	**100.00**	82.66	56.85	79.83
Synthesized GEI ([31]+GAN)	**100.00**	72.87	33.51	68.79
Synthesized PEI ([1]+GAN)	98.40	90.95	61.70	83.68
***DGEI* with GAN (Proposed)**	**100.00**	**96.37**	68.14	**88.18**

approach outperforms all other techniques for scenarios C_1 and C_4. Only in the case of scenario C_5, the work in [94] performs better than our approach by 1.96%. However, the average performance of our approach on the CASIA B data is better than all the other techniques by at least 2.96%.

5.2.2 Analysis using TUM-GAID Data Set

In our experiments, as gallery set we consider the fronto-parallel walking sequences from right to left direction. Hence, for test sequences with walking sequences from left to right, we flip the images horizontally to enable proper comparison between the training and test sets. The four sequences of normal walking (*nm*) condition

Table 5.5: Table showing a comparative study of our approach with state-of-the-art gait recognition techniques using TUM-GAID Data. (*In each column, the bold number represents the maximum accuracy for the corresponding scenario*)

Approach Name	T_1(%)	T_4(%)	T_5(%)	Mean (%)
Gait Energy Image (GEI) [31]	**97.03**	20.72	**87.82**	68.52
Active Energy Image (AEI) [51]	91.14	72.78	79.01	80.97
Deep-CNN[88]	90.78	29.60	60.36	60.24
GEINet[91]	90.29	25.49	78.61	64.79
DTW [69]	65.90	12.29	59.67	45.95
MGAN [50]	95.23	66.61	85.19	82.34
GaitSet [94] with Large Training(LT)	86.18	58.55	76.15	73.62
GaitGAN [95]	94.57	54.27	83.22	77.35
PEI with 13 key poses [1]	65.08	47.04	50.65	40.92
Pose based BEI with 6 poses	90.32	62.62	77.86	76.93
***GAEI* [119]**	76.48	51.15	65.13	64.25
***DGEI* without GAN**	89.01	66.88	71.80	75.89
Synthesized GEI ([31]+GAN)	94.36	72.18	84.54	83.69
Synthesized PEI ([1]+GAN)	90.54	66.72	81.81	79.69
***DGEI* with GAN (proposed)**	94.18	**73.81**	86.54	**84.84**

from the right to left direction form the training set in our experiments, while the others with varying co-variate conditions form the test set. The gallery set for training the Pix2Pix GAN is constructed from the first 30 subjects present in the data set. A similar comparative performance evaluation of the different gait recognition methods has been done for the different scenarios of the TUM-GAID data set, and the results are presented in Table 5.5. A similar conclusion about the effectiveness of our proposed approach can also be made from this table. We can observe that for normal walking, our approach performs significantly accurately with 94.18% and 86.54% accuracy for scenarios T_1 and T_4, respectively, but the work in [31] performs better than our approach by 2.85% and 1.28%, respectively.

This is since the training and corresponding test silhouettes for scenarios T_1 and T_4 in the TUM-GAID data are very similar to each other, and [31] performs accurately with similar training and test conditions. However, the accuracy of [31] drops significantly if the training and test conditions differ as seen for scenario T_4. Compared to all other methods, our approach using *DGEI* and GAN performs consistently well for the different scenarios and has the highest average accuracy for the different scenarios.

5.2.3 Analysis using OU-ISIR TreadMill Data Set B

Out of the 68 subjects in this data set, a list of 20 subjects given in the instruction file named as *clothes-training-list* are considered for training the Pix2Pix GAN model to remove the different co-variate conditions effectively. For each of these 20 subjects, there are sequences with the varying co-variate conditions, and the GAN is trained to transform the *DGEI*s for any given co-variate condition into the corresponding *DGEI*s for the target co-variate condition. Similar to the previous two data sets, here also the target co-variate condition is considered to be similar to the condition in which the gallery set for gait recognition has been captured. But unlike the previous two data sets, in the OU-ISIR TreadMill Data Set B, the gait sequences are not grouped with respect to the different co-variate conditions for each person. Rather, a number of different co-variate conditions have been specified, and corresponding to each of these conditions the data set contains gait sequences for a few individuals. Hence, while making a comparative study of the different gait recognition methods using this data set, we could not compute the recognition accuracy separately for the different co-variate (i.e., clothing) conditions. Instead, we consider the complete test data set with varying co-variate conditions, and report the overall gait recognition accuracy after co-variate object removal. The comparative results using the same approaches used in the previous experiments have been presented in Table 5.6 in terms of *Rank 1* accuracy.

From the accuracy values corresponding to the methods: *DGEI without GAN* and *DGEI with GAN* presented in the table, it is clear that the effect of employing the GAN-based co-variate object removal stage is not very prominent for this data set.

Table 5.6: Table showing a comparative study of our approach with state-of-the-art gait recognition techniques using OU-ISIR TreadMill Dataset B. (*In each column, the bold number represents the maximum accuracy for the corresponding scenario*)

Approach Name	Acc (%)	Approach Name	Acc (%)
Gait Energy Image (GEI) [31]	49.57	**GaitGAN [95]**	59.92
Active Energy Image (AEI) [51]	55.16	***GAEI* [119]**	65.83
Deep-CNN[88]	54.82	**PEI with 37 key poses [1]**	59.82
GEINet[91]	56.57	***DGEI* without GAN**	64.04
DTW [69]	40.25	**Synthesized GEI ([31]+GAN)**	63.96
MGAN [50]	66.79	**Synthesized PEI ([1]+GAN)**	**66.84**
GaitSet [94] with large training	65.81	***DGEI* with GAN (proposed)**	65.91

This is since the co-variate conditions present in the OU-ISIR TreadMill Dataset B data have only slight clothing variations, which do not alter the silhouette shape significantly. It is also seen that our approach performs closely similar to that of [50] and [1]+GAN, and has only 0.88% and 0.93% lower accuracy compared to these methods. For this experiment, the gallery set for gait recognition (as tagged in the data set) consists of only one sequence for each subject, and hence the gait recognition accuracy value for each method is lower compared to the other two data sets used in the study (refer to Tables 5.5, and 5.6). As observed from Tables 5.4, 5.5, and 5.6, for the different types of co-variate conditions, our approach shows the most consistent and good performance among all the other competing approaches. In future, the effectiveness of our GAN-based co-variate condition handling proposed approach needs to be studied using datasets which feature a higher number of co-variate conditions. It may also be studied that if the accuracy of our overall approach can be improved by increasing the volume of the training data.

In our next experiment, we study the rank-wise improvement in recognition accu-

racy of the proposed *DGEI* with GAN on the three data sets, namely CASIA B, TUM-GAID, and OU-ISIR TreadMill B. Results are shown in Fig. 5.8 in terms of the Cumulative Match Characteristic (CMC) Curves as the value of the Rank is increased from 1 to 10. In this plot, the horizontal axis represents the rank, whereas the vertical axis represents the accuracy in percentage for the different rank values. It can be observed from the plot that, as expected, for each data set,

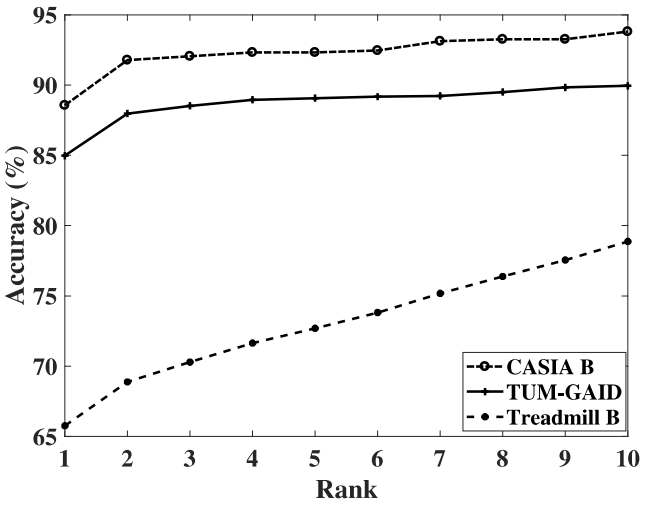


Figure 5.8: Cumulative match characteristic curves showing rank-wise improvement in the accuracy of our approach for the different data sets as the rank is increased from 1 to 10

the recognition accuracy has an improvement trend as the magnitude of the rank is increased. Out of 124 subjects in the CASIA B data set, the Rank 5 and Rank 10 accuracy for this data set are 92.32% and 93.80%, respectively. For TUM-GAID data set, the corresponding numbers are 89.07% and 89.94% respectively out of 305 subjects, while for the OU-ISIR TreadMill B data set, out of 68 subjects, the Rank 5 and Rank 10 accuracy values are 72.69% and 78.87%, respectively. As explained previously, the low accuracy for the OU-ISIR TreadMill B data set is mostly due to the presence of only a single instance of each subject to train the gait recognition model, and also the availability of a fewer number of subjects in the gallery set compared to the other two data sets.

Finally, we make a comparative study among the different gait features proposed

in the thesis, namely *BEI* and Pose-based *BEI* discussed in Chapter 1, *GAEI* discussed in Chapter 2, and *DGEI* discussed in Chapter 3 in terms of the Rank 1 accuracy corresponding to scenarios C_1, C_4 and C_5 for the CASIA B data and T_1, T_4, and T_5 for the TUM-GAID data (refer to Table 2.1), and the mean accuracy corresponding to these three scenarios. The test sequences of the CASIA B and TUM-GAID data sets are considered for this experiment and results are presented in Tables 5.7,5.8 respectively for the two data sets. In both these tables, we have performed additional experiments to study: (i) if the accuracy of the *GAEI*-based recognition (i.e., the approach proposed in Chapter 4) improves after appending the GAN-based co-variate object removal phase, and (ii) the accuracy of *DGEI* with GAN-based co-variate object removal (i.e., the approach proposed in Chapter 5) on sequences with a lower frame rate. Since neither of the CASIA B or the TUM-GAID data sets contains sequences with low frames rates, we skip every alternate frame from each test sequence present in the two data sets to synthetically generate a sequence with almost half frame rate.

From the results shown in both the tables, we observe that our *DGEI* with GAN-based co-variate object removal phase, proposed in Chapter 5, performs the best among the three different approaches proposed in the thesis in terms of the mean accuracy and its classification time is also reasonably less. The average accuracy of this method for the CASIA B data is 88.18% and that for the TUM-GAID data is 84.84%. It has also been seen to perform quite consistently across the different co-variate scenarios considered in the experiment. The *GAEI* with GAN improves over our originally proposed *GAEI* feature by 2.71% for the CASIA B data and by 20.48% for the TUM-GAID data, which once again emphasizes the effectiveness of the use of GAN for co-variate object removal. Although, the performances of the *GAEI* with GAN and the *DGEI* with GAN are comparable in terms of mean accuracy, the former suffers from a significantly high classification time. It may be noted that the classification time reported in the final column of the table does not include the pre-processing time, the feature computation time, or the GAN-based processing time. If these are added, then it can be understood that the total response time of the *GAEI*-based recognition will be even higher. The classification time of *DGEI* with GAN is 4.05 secs while working with the CASIA B data and 8.61 secs while working with the TUM-GAID data with a

larger gallery set. Although the reported classification times for the *DGEI* with GAN are reasonably low, these are slightly higher than those reported for the Pose-based *BEI*. This is due to the fact that the Pose-based *BEI* feature has been derived using six key poses corresponding to a half gait cycle, whereas the *DGEI* feature has been derived using 13 unique poses corresponding to a half gait cycle. Reducing the number of unique poses in a half gait cycle would reduce the feature computation time as well as the classification time for the *DGEI* with GAN, but the accuracy is also expected to reduce, especially if the training and test sequences are captured at varying frame rates. From the above comparative study of the three proposed features, it can be concluded that the *DGEI* feature with GAN performs consistently well for the different co-variate conditions and also with a reasonably low response time.

For carrying out a global evaluation of the proposed approaches in the thesis, we have made a comparative study among all the methods proposed in Chapters 3, 4, and 5 using the CASIA B, TUM-GAID, and CASIA C datasets. Results are shown in Tables 5.7, 5.8, 5.9 respectively for the above three datasets in terms of Rank 1 accuracy for specific training-test scenarios, and the mean accuracy. For CASIA B data, we have considered the scenarios C_1, C_4, and C_5, whereas for the TUM-GAID data, we have considered scenarios T_1, T_4, and T_5, and for the CASIA C data, we have considered scenarios A_1, A_2, and A_3 (refer to Table 2.1).

Table 5.7: Comparative performance analysis of the different gait recognition methods on CASIA B data [2] in terms of Rank 1 accuracy and mean accuracy

Approach Name	C_1(%)	C_4(%)	C_5(%)	Mean (%)
BEI	98.38	90.32	29.43	72.71
Pose based *BEI*	99.59	92.33	57.25	83.05
***GAEI* without GAN**	98.78	90.24	66.66	85.22
***GAEI* with GAN**	99.59	92.68	**71.54**	87.93
***DGEI* without GAN**	**100.00**	82.66	56.85	79.83
***DGEI* with GAN**	**100.00**	**96.37**	68.14	**88.18**

We observe from the above Tables 5.7, 5.8, that the pose-based features proposed in the thesis based on dictionary of key pose sets or relaxed state transition model (as in Chapters 4 and 5), in general, improve upon the aggregation-based feature *BEI* once co-variate conditions are removed by employing a *GAN*. Out of

Table 5.8: Comparative performance analysis of the different gait recognition methods on TUM-GAID data [3] in terms of Rank 1 accuracy and mean accuracy

Approach Name	T_1(%)	T_4(%)	T_5(%)	Mean (%)
BEI	93.93	68.29	81.47	81.23
Pose based *BEI*	90.32	62.62	77.86	76.93
***GAEI* without GAN**	76.48	51.15	65.13	64.25
***GAEI* with GAN**	85.36	**92.68**	76.15	84.73
***DGEI* without GAN**	89.01	66.88	71.80	75.89
***DGEI* with GAN**	**94.18**	73.81	**86.54**	**84.84**

Table 5.9: Comparative performance analysis of the different gait recognition methods on CASIA C data [2] in terms of Rank 1 accuracy and mean accuracy

Approach Name	A_1(%)	A_2(%)	A_3(%)	Mean (%)
BEI	**97.05**	**96.07**	**97.38**	**96.84**
Pose based *BEI* with 14 poses	88.88	77.77	88.88	85.17
GAEI without GAN	93.13	**96.07**	96.07	95.09
GAEI with GAN	95.13	93.13	93.17	93.82
***DGEI* wrt 16 poses without GAN**	95.09	95.75	96.73	95.85
***DGEI* wrt 16 poses with GAN**	97.17	93.85	94.05	95.02

the datasets used, CASIA-B has a sufficiently large number of frames and the state transition model-based mapping of frames to key poses works better for this dataset compared to that for TUM-GAID. This is apparent from the results presented in Tables 5.7 and 5.8, in which we can see that although the pose-based features show a strikingly high improvement in the average recognition accuracy for the CASIA-B data, the same is not true for the TUM-GAID data in which the average recognition accuracy is for each of *Pose-based BEI*, *GAEI*, and *DGEI* without GAN is less than that of *BEI*. Employing the GAN-based co-variate condition removal phase helps in improving the recognition accuracy for scenarios C_4 and C_5 in case of CASIA-B data and scenarios T_4 and T_5 in case of TUM-GAID data, thereby improving the overall average accuracy. On the other hand, the CASIA-C data does not feature walking sequences with different co-variate conditions, but has sequences with varying walking speeds. Due to this reason, our

trained GAN model is not able to make any improvements to the *GAEI* or *DGEI* features, rather it reduces the quality of the features, resulting in a decrement in the mean accuracy, as can be seen from Table 5.9.

From the extensive experiments conducted across the chapters as well as the global analysis results involving all the approaches proposed in the thesis, it can be concluded that our *Pose-based BEI* feature proposed in Chapter 3 is most effective when both the training and test sets are captured under uniform walking speeds and frame rates. In contrast, the approaches proposed in Chapters 4 and 5 are capable of handling the limitations of the first approach proposed in Chapter 3 to a certain extent, although it requires the gallery for constructing key poses and the gallery for gait recognition to be captured at similar frame rates/walking speeds. Among these, recognition using the *GAEI* feature is time-intensive since it makes the prediction by comparing with all the key poses present in the dictionary of key pose sets. On the other hand, *DGEI with GAN*-based technique proposed in Chapter 5 can simultaneously handle walking speed variation and co-variate differences much effectively than *GAEI*-based approach. Our methods can be conveniently extended to perform view-invariant recognition and recognition in the presence of occlusion by integrating with the suitable view-translation and occlusion reconstruction models. However, a limitation of each of the proposed methods is that these are not tuned to handle drastic differences in the walking speed. Any pose-based gait recognition approach requires division of a gait cycle into a fixed number of intermediate key poses, and significantly different walking speeds between the training and test sequences will result in obtaining very different sets of features corresponding to the key poses, even after relaxing the constraints in the state transition model, as described in Chapter 5. We have also experimentally demonstrated this fact using the *Pose-based BEI* feature with 36 key poses considering the OU-ISIR Treadmill A data, which is specifically constructed for testing cross-speed gait recognition models, in Table 5.10. Specifically, this dataset contains sequences with 10 different walking speeds labeled *1* to *10*. We consider sequences for walking speeds labeled *4*, *5*, and *6* for training our model. The trained model is next used to find the recognition accuracy on different test sets with varying walking speeds, as given in the second column of the same table. As in all other experiments, here also none of the test sequences

has been used to train the model. We observe that higher recognition accuracy

Table 5.10: Accuracy of *Pose-based BEI* for cross-speed scenarios

Training Speeds	Testing Speeds	Accuracy in (%)
4, 5, 6	2 to 10	66.01
4, 5, 6	2	91.17
4, 5, 6	3	97.05
4, 5, 6	4	100.0
4, 5, 6	5	100.0
4, 5, 6	6	97.05
4, 5, 6	7	82.35
4, 5, 6	8	14.70
4, 5, 6	9	5.88
4, 5, 6	10	5.88

is always obtained if the training and test walking speeds are closely similar to each other. When the test speed labels are *4*, *5*, or *6*, the recognition accuracy values are quite high, i.e., (>97%). The accuracy decreases gradually as the difference between the training and test speeds is increased. Very poor results are obtained for the test walking speeds labeled *9* and *10*, resulting in a degradation in the overall average accuracy (reported in the first row of the table). Similar results are observed for each of the other approaches proposed in the thesis and we have skipped these results to avoid redundancy, although it may be noted that the effect of speed variation on the recognition accuracy is less prominent for the approaches in Chapter 4 and 5 than that for the approach given in Chapter 3. The corresponding accuracy values for testing speeds labeled *9* and *10* for the *GAEI* feature (discussed in Chapter 4) are 26.90% and 20.23%, respectively, and that for the *DGEI* feature (discussed in Chapter 5) are 30.30% and 28.97%, respectively.

5.3 Summary

In this chapter, an effective approach has been proposed that can identify an individual from his/her walking sequence even if during the testing phase the silhouette appearance of the individual gets affected due to the presence of covariate factors like carrying bags, wearing coats, or other objects. Initially, a

maximal set of unique walking poses in a half gait cycle are determined, and next an input sequence of a subject (with/without co-variate objects) is mapped to the appropriate unique poses in the set by preserving the temporal order of walking. Next, appearance-based gait features are extracted at the granularity of these pre-determined unique poses. A state transition model with relaxed constraints has been employed in this work to map a sequence of frames to the appropriate unique poses. If the number of frames in the input sequence is less than the number of unique poses in the half cycle, a few states will be skipped during the mapping phase so that the input frame can be correctly matched with the appropriate pose by maintaining the spatio-temporal order of walking. The stricter constraints used in the state transition model in the previous chapters could not handle this situation effectively. This approach helps in handling training and test sequences with slight differences in walking speeds in an efficient manner. But for significantly high differences in the walking speed, this approach also fails to provide good results. Co-variate objects, if any, in a test sequence are automatically eliminated by employing a Pix2Pix Generative Adversarial Network before carrying out the final comparison with the gallery set. Experimental results using the CASIA B, TUM-GAID, and OU-ISIR Treadmill Dataset B as well as a comparative study with the state-of-the-art gait recognition approaches and our previously proposed approaches in the thesis emphasize the superiority of the proposed approach in this chapter over the others in the presence of varying co-variate conditions during training and testing. Our present approach performs with an average Rank 1 accuracy of more than 84% for the different co-variate conditions, which is quite significant.