Chapter 4

Temporal Attention Features for Video-Based Person Re-Identification

In the previous chapter, we present a re-identification approach that extracts effective spatial features from images of individuals for person re-identification. Even if walking videos are captured by the cameras, temporal information was completely ignored while computing the features for re-identification. However, the use of temporal features along with spatial features is expected to improve the effectiveness of video-based person re-identification. The spatio-temporal information from a sequence of image frames can be effectively computed using Recurrent Neural Networks (RNN)-based neural networks. As already explained in Chapter 2, to date, only a few previous approaches make use of such spatio-temporal information through use of RNN models, or temporal attention-based models such as [47–50]. Since sequence data captured by a camera has to be used to train an RNN model to re-identify a subject in the field of view of another camera, and only limited information about an individual can be obtained from the video captured by one camera, the RNN model may not get trained well with the limited data which can affect its re-identification performance during the test case. An ensemble of multiple spatio-temporal features captured by different RNN models is expected to perform more accurately than a single RNN model that we study in this chapter.

A similar surveillance setup as discussed using Figure 3.1 in Chapter 3 has also been considered here. With reference to this figure, as a person walks out through an exit gate, we propose capturing spatio-temporal features from the video data instead of spatial features only, as done in Chapter 3. Recurrent Neural Network (RNN)-based models [51, 135–137] have been extensively used in the past to derive spatio-temporal features from video data. Here, we also employ different variations of RNN networks to predict the probabilities of the target subject to belong to the different classes present in the gallery set and next fuse these predictions to perform re-identification. Specifically, predictions from three different Deep Neural Networks are fused to estimate the class of a test subject. The three sub-networks used in the ensemble model are (i) a *Full-Body Pose Attention Network* (*FPAN*), (ii) a *Motion Pooling Network* (*MPN*), and (iii) a Convolutional *Long-Short Term Memory Network* (*LSTM*). These three models are based on the popular *ResNet-101* architecture [138] and capture different but useful information related to human motion. While the *FPAN* captures mainly appearance-related information of an individual, the *MPN* captures dominant motion features, and the *LSTM* derives dynamic information from the spatial correlation between frames in a captured sequence. The use of *ResNet-101* as the base network is justified since its effectiveness in object detection and recognition has already been well-established [139, 140]. The pre-trained version of the *ResNet-101* architecture [141] has been used here to generate the frame-level feature descriptors from the input image sequences. The complete re-identification approach and detailed discussion on the above-mentioned sub-networks are given next.

The overall pipeline of the proposed re-identification approach consists mainly of three

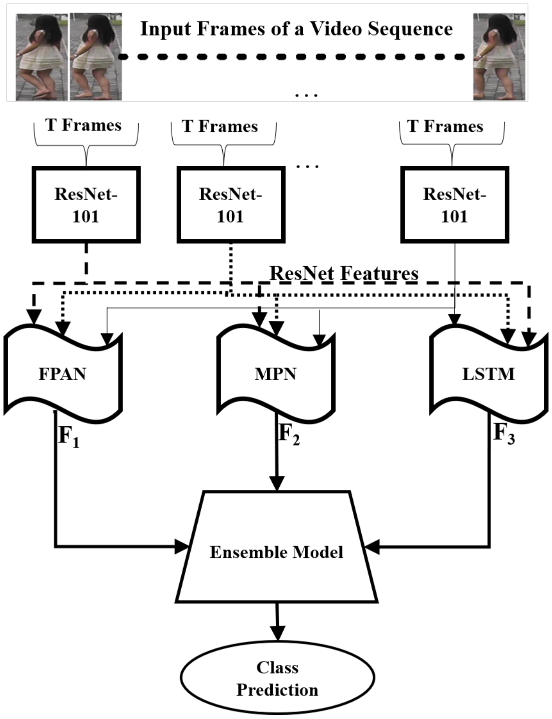


Figure 4.1: A block diagram of the re-identification approach

modules: (i) training of the individual sub-networks, namely, *FPAN*, *MPN*, and *LSTM*, (ii) aggregation of features from the trained models, and (iii) predicting the class of a test subject, as shown in the block diagram of Figure 4.1. With reference to the figure, initially, the entire video is segmented into non-overlapping clips of T frames. Each set of T frames present in a clip is next passed one at a time through a pre-trained *ResNet-101* model to derive T features, one for each set. These T *ResNet* features are next input to each of the three sub-networks, i.e., *FPAN*, *MPN*, and *LSTM*, to compute clip-level motion features. Similarly, clip-level features are computed from each of the other clips at the three sub-networks, and finally, these features from each sub-network are aggregated to obtain three different features from an input video sequence. Each of these features captures different aspects of human walking and preserve important kinematic characteristics about each individual. Figure 4.1 shows separate *ResNet-101* blocks to make it easier for the readers to understand the flow of the work. During implementation, a single such *ResNet* block has been considered, and each frame is passed separately through that network to obtain the deep features corresponding to that frame. Let the dimensions of each feature map at the final convolution layer of the *ResNet-101* model be $w \times h$. Since, this layer contains 2048 feature maps, the feature vector size at this layer can thus be represented by [w,h,2048]. Let us denote the feature vector corresponding to the t^{th} frame of clip c by f_c^t, where t = 1,2,..., T and c= 1,2,...$\mathcal{C}$. If there are $\mathcal{C}$ total clips, each of *FPAN*, *MPN*, and *LSTM* fuses information from the $\mathcal{C}$ clips to compute features F_1, F_2, and F_3, which are finally fused to predict the class of the subject.

4.1 Sub-Network Architecture Details

4.1.1 Full-Body Pose Attention Network

In Full-Body Pose Attention Network (*FPAN*), the average of attention scores corresponding to the different clips (i.e., fragments of an input sequence) is computed. This is done by employing a temporal attention layer after the final convolution layer of the *ResNet-101*, as explained next. To compute the attention score for clip c, we consider the attention feature vector b_c^t as the average of the T ResNet-101 generated feature vectors $\{f_c^1, f_c^2, ..., f_c^T\}$ as shown in (4.1):

$$b_c^t = \frac{1}{T}\sum_{t=1}^{T} f_c^t. \tag{4.1}$$

The size of the feature vector b_c^t can be represented as [w,h,2048]. Next, we multiply each of the T *ResNet-101* feature vectors, f_c^1, f_c^2, ..., f_c^T with the attention feature vector b_c^t and sum up all the resultant feature vectors to obtain a single attention-infused feature vector for clip c denoted by m_c. Mathematically,

$$m_c = \sum_{t=1}^{T} b_c^t \otimes f_c^t, \tag{4.2}$$

where $\otimes$ denotes the product operator. The size of the feature vector m_c is also [w,h,2048]. The feature maps obtained after the multiplication operation are now passed through a convolution layer with 256 kernels, each of dimensions $w \times h$ to reduce the feature vector size to [w,h,256], following which a fully connected layer with a single node is considered that computes the attention score for clip c. If this score is denoted by S_c, then the final attention score (S) provided by the *FPAN* sub-network is obtained by summing up the attention scores from each of the $\mathcal{C}$ clips as shown in (4.3):

$$S = \sum_{c=1}^{\mathcal{C}} S_c. \tag{4.3}$$

Now, if there are N subjects in the gallery set, then we compute N such scores following (4.3). Let us denote these N scores by S^1, S^2, ..., S^N. If the attention score of an input test subject is denoted by S^t, then *FPAN* provides the final feature F_1 in which each attribute represents the probability of the test subject to belong to a particular class. Thus, we can write:

$$F_1 = \{F_1^1 \; F_1^2 \; ... \; F_1^N\}, \tag{4.4}$$

where,

$$F_1^j = \frac{|S^j - S^t|}{\sum_{j=1}^{N} |S^j - S^t|}, \forall \, j = 1, 2, ..., N. \tag{4.5}$$

4.1.2 Motion Pooling Network

A Motion Pooling Network (*MPN*), with average pooling layers, has been employed as the second network, which preserves important information about the shape of a subject in each clip c by averaging the feature vectors f_c^t obtained corresponding to each frame t in the clip. Average pooling enables the preservation of useful dynamic information by aggregating clip-level temporal feature descriptors. The feature vector at the penultimate layer of this network is of size [w,h,2048], which is followed by a fully connected classification layer with a number of nodes equal to the number of classes in the data set. This classification layer outputs a vector F_2 in which each attribute represents the probability of a test subject to belong to the corresponding class.

4.1.3 Long-Short Term Memory Network

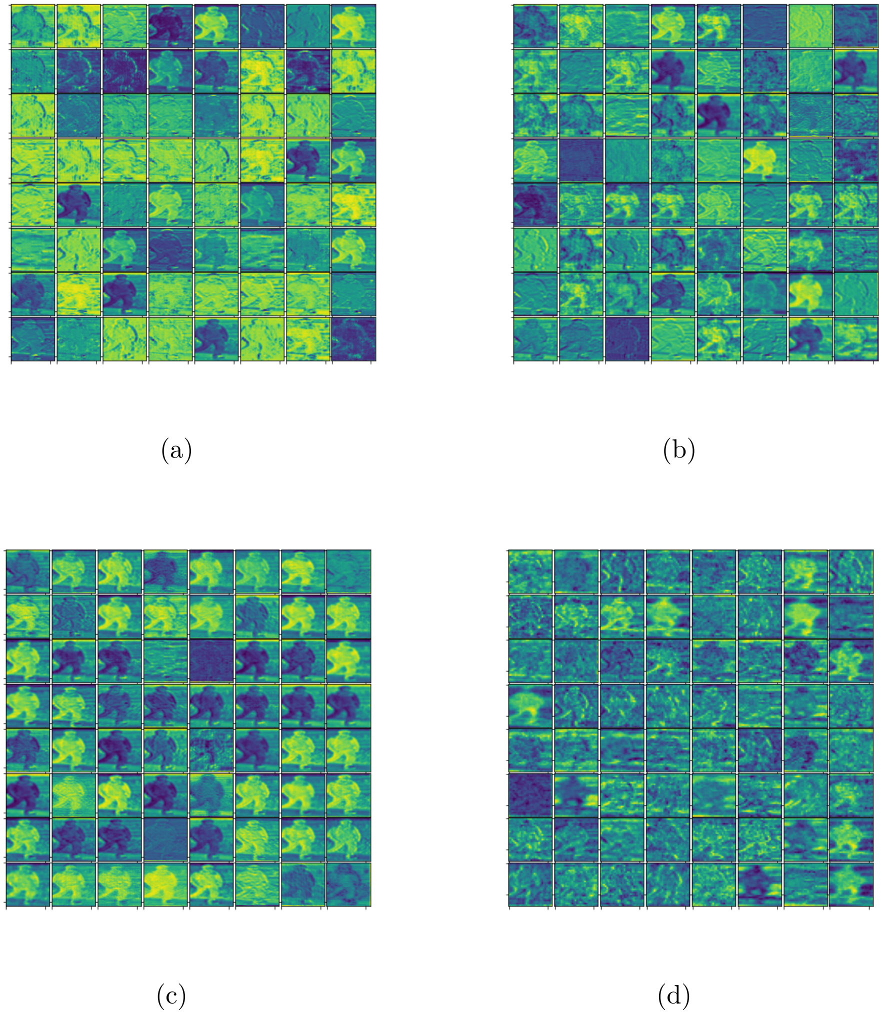
This network is used to capture recurrent information from a walking sequence. It is well-known that an *LSTM* network can represent any time-series data effectively. Since a walking sequence can also be looked upon as time-series data, the features provided by the *LSTM* network are expected to preserve unique motion features for each subject. Specifically, we use two *LSTM* cells on top of the feature descriptor from the *ResNet-101* to generate correspondences among the frames in an input video sequence. The inputs to the second *LSTM* cell are the original image frames along with the hidden layer features from the first *LSTM* cell. If the feature vector corresponding to clip c, as obtained from this second *LSTM* cell, is denoted by o_c, then

$$o_c = \sigma(W_o.[x_i, h_{i-1}] + b_o). \tag{4.6}$$

where W_o represents the weight matrix associated with the network, and x_i and h_{i-1} respectively represent the inputs at the current cell and output from the previous cell, and b_0 denotes the bias. Similar to *MPN*, here also we compute the averaged *LSTM* feature from all the clips and obtain the feature vector F_3 from the final layer whose attributes represent the individual class probabilities. We follow an ensemble-based approach by averaging the class probabilities F_1, F_2, and F_3 provided by the different deep models, namely, *FPAN*, *MPN*, and *LSTM* to make the final prediction about the class of a test subject. Due to fusing information from multiple models, our ensemble approach results in accurate and reliable predictions. This stacked ensemble model has been named as Temporal Motion Aware Network (*T-MAN*) since it accumulates the prediction of several temporal motion models.

It may be noted that the first two sub-networks, namely, the *FPAN* and the *MPN* require setting the value of T, i.e., the clip length. If this value is set to *1*, then no temporal information can be captured. On the other hand, for very high values of T, the network will fail to capture the kinematic information of a person's movement at a high resolution. Thus, determining the optimal value of T is very important for re-identification. Similarly, for the third sub-network, namely, the *LSTM*, the optimal size of the hidden state (s) must be determined. Additionally, two other hyper-parameters of the models, namely, the learning rate (η) and the weight decay (γ) have to be determined. This is done by training the ensemble model multiple times from scratch for 10 different combinations of T, s, η, and γ, and observing which among these 10 combinations provide the maximum cross-validation accuracy on the validation set. For training the ensemble, each time we consider a value of T in the range *2* to *8*, and a value of s among *256*, *512*, and *1024*, and η and γ values are randomly generated between *0* and *0.2*. The extensive *MARS* has been used for this experiment. It consists of 1191003 images from 1262 identities (refer to Table 2.3 in Chapter 2). As training set, we consider the first 80% frames from each subject, whereas the validation set is formed with the next 10% frames from each subject. We observe that the combination of T=4, s=512, η=*3e-4*, and γ=*5e-4* provide the best cross-validation accuracy among all the other configurations. Training of each of the sub-networks has been done for a maximum of 1000 epochs, or till the loss value does not alter significantly in successive epochs. The same configuration has been used to report the results in this chapter.

For visualization, we also present the feature maps generated at the final convolution layer of each sub-network as well as the ensemble model *T-MAN* in Figures 4.2(a)-(d). Although, 256 different feature maps are computed at the penultimate layer of each sub-network, here we present randomly chosen 64 feature maps among these for visualization. The soft-max cross-entropy loss function has been used to train each of the individual networks, i.e., *FPAN*, *MPN*, and *LSTM*. If $p_{i,a}$ and $g_{i,a}$, represent the



(a)

(b)

(c)

(d)

Figure 4.2: Feature maps generated at the intermediate layers of (a) *FPAN*, (b) *MPN*, (c) *RNN*, (d) *T-MAN*

predicted and actual ground truth for the a^{th} clip of the i^{th} subject, then

$$Loss_{(Softmax)} = -\frac{1}{N\mathcal{C}} \sum_{i=1}^{N} \sum_{a=1}^{C} g_{i,a} \; log \; p_{i,a}, \tag{4.7}$$

where N and $\mathcal{C}$ respectively represent the number of subjects and the number of clips for each subject.

4.2 Experiments and Results

Here, we present the results obtained from the evaluation of the proposed *T-MAN* on different data sets, namely, *PRID2011* [112], *iLIDS-VID* [111], *MARS* [113], and *IIT (BHU) Re-Identification data* (refer to Section 2.4.2 for further details of the data sets), and compare it with existing video-based re-identification techniques that include [48, 54, 108, 111, 124, 142–151]. For each data set, we consider the sequence formed by the first 80% frames from each subject as the gallery set to train the *T-MAN*, and the sequence formed by the remaining 20% frames for testing. The same train-test split has been used for all of the experiments reported in this chapter.

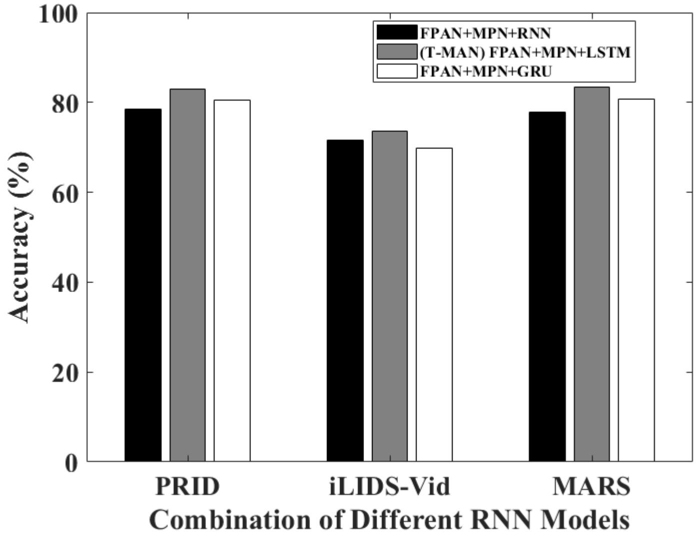


Figure 4.3: Rank 1 accuracy for different combinations of RNN Models (i.e., *Simple RNN*, *LSTM*, and *GRU*)

In our first experiment, we evaluate the effectiveness of fusing three different types of *RNN* models with *FPAN* and *MPN*, namely, (i) simple Recurrent Neural Network (*RNN*), (ii) Long-Short term Memory (*LSTM*) as discussed in the Section 4.1.3, and (iii) Gated Recurrent Unit (*GRU*) using *PRID2011* [112], *iLIDS-VID* [111], and *MARS* [113] data sets. The hidden state size and sequence length for each type of *RNN* cell have been fixed to 512 and 8, respectively. The results are shown in the form of a grouped bar chart in Figure 4.3 in which each color corresponds to a particular ensemble model as indicated in the legend of the plot. Data set names have been specified along the horizontal axis, and Rank 1 accuracy is plotted along the vertical axis. It can be seen from the figure that the proposed ensemble model *T-MAN* with *LSTM* as the recurrent network performs better than any other ensemble model for each of the data sets used in the study. This justifies the use of *LSTM* as a recurrent feature extractor over the other *RNN* models. In our next set of experiments, we compare the performance of the proposed ensemble approach *T-MAN* with existing techniques using the same three data sets as used in the previous experiment, i.e., *PRID2011* [112], *iLIDS-VID* [111], and *MARS* [113]. Results for the first two data sets are shown in Tables 4.1 and 4.2 in terms of Rank 1, Rank 5, and Rank 10 accuracy, whereas results for the third data set are shown in Table 4.3 in terms of Rank 1, Rank 5, and Rank 20 accuracy and the *map* score. In each table, along with the existing approaches, we have also tested the effectiveness of the individual sub-networks used in the proposed *T-MAN* model, i.e., *FPAN*, *MPN*, and *LSTM*.

Table 4.1: Comparative results on *PRID-2011* data set for Ranks 1, 5 and 10

Methods	**Accuracy (%)**		
	Rank 1	**Rank 5**	**Rank 10**
Baseline [47]	70.0	90.0	95.0
STA [142]	64.1	87.4	90.0
TDL [146]	56.7	80.1	87.6
DVR [111]	40.4	71.8	84.6
TAPR [143]	74.0	94.6	94.2
SRID [144]	35.2	59.5	70.0
RFA-Net [48]	58.2	85.8	93.5
AMOC [54]	**83.7**	**98.3**	**99.4**
TAUDL [152]	49.4	78.7	92.6
DGM+XQDA [55]	81.1	95.1	98.9
our *FPAN*	65.0	71.5	83.0
our *MPN*	62.4	72.0	85.5
our *LSTM*	81.5	88.2	93.5
our *T-MAN* (*FPAN+MPN+LSTM*)	83.0	96.4	98.8

Table 4.2: Comparison results on *iLIDS-VID* data set for Ranks 1, 5 and 10

Methods	**Accuracy (%)**		
	Rank 1	**Rank 5**	**Rank 10**
Baseline [47]	58.0	84.0	91.0
STA [142]	44.5	71.7	83.7
TDL [146]	56.5	87.6	95.6
DVR [111]	39.4	61.1	71.8
TAPR [143]	55.1	87.4	93.3
SRID [144]	25.0	44.5	55.6
RFA-Net [48]	49.3	76.7	85.4
AMOC [54]	68.7	**94.3**	**98.3**
TAUDL [152]	26.7	51.3	78.6
DGM+XQDA [55]	42.6	67.7	76.6
our *FPAN*	61.1	69.4	81.0
our *MPN*	59.0	66.5	76.4
our *LSTM*	64.9	77.2	85.0
our *T-MAN* (*FPAN+MPN+LSTM*)	**73.5**	91.4	96.6

From Table 4.1, it can be seen that for the *PRID-2011* data set, we have achieved a quite satisfactory Rank 1 accuracy of 83% with the proposed ensemble model *T-MAN*, and it is only 0.7% less than the best approach [54]. For Rank 10, the accuracy given by our approach is 98.8%, which can be said to be significantly good, given the data set consists of 749 identities. The benefits of choosing an ensemble model can be verified from the final four rows of this table. It can be seen that the ensemble model significantly improves upon the accuracy of the individual sub-networks, namely, *FPAN*, *MPN*, and *LSTM*. It is also observed from the table that although for this data set the re-identification accuracy given by our approach for the different rank values is significantly high, the approach in [54] performs slightly better than ours. Similar observation also follows from Table 4.2. Here, our approach performs better than most of the existing techniques, except [54] for Ranks 5 and 10. We observe that although the Rank 1 accuracy of our method exceeds that of [54] by about 5%, the Rank 5 and Rank 10 accuracy of [54] are slightly better than that of ours. Table 4.3 shows a comparative performance analysis of our work with ensembles of other existing

Table 4.3: Comparison results on *MARS* data set for Ranks 1, 5 and 20 with Mean Average Precision (map)

Methods	**Accuracy (%)**			**Mean Average Precision** (*map*)
	Rank 1	**Rank 5**	**Rank 20**	
SDALF [145]+DVR [111]	4.1	12.2	25.4	1.8
HOG3D [124]+KISSME [147]	2.7	6.4	12.5	0.8
HistLBP [151]+XQDA [150]	18.2	33.1	46.0	8.0
BoW [108]+KISSME [147]	30.6	46.4	60.1	15.5
LOMO + XQDA [150]	30.8	46.4	60.9	16.5
gBiCov [148]+XQDA [150]	9.2	19.8	33.4	3.7
IDE [149]+XQDA [150]	65.5	82.0	89.0	47.5
AMOC [54]	68.3	81.4	90.6	52.9
TAUDL [152]	43.8	59.9	72.8	29.1
DGM+IDE [55]	48.1	64.7	77.4	29.1
our *FPAN*	60.1	69.9	78.4	49.5
our *MPN*	57.5	66.1	79.0	45.0
our *LSTM*	68.7	79.8	89.2	53.4
our *T-MAN* (*FPAN+MPN+LSTM*)	**83.3**	**93.5**	**95.6**	**76.7**

approaches using the *MARS* data set. Three different metric learning algorithms and seven feature descriptors have been used in this study, as given next. The descriptors include SDALF [145], HOG3D [124], HistLBP [151], gBiCov [148], LOMO [150], BoW [108] and IDE [149], whereas the metric learning methods are DVR [111], KISSME [147], and XQDA [150]. In addition to these, we have also used AMOC [54], TAUDL [152], and the individual sub-networks, namely *FPAN*, *MPN*, and *LSTM*, as well as their ensemble, i.e., *T-MAN* in the comparative study. Although among the existing approaches, the work in [54] was seen to perform slightly better than our *T-MAN* for the *PRID-2011* data at Ranks 1, 5, and 10, and also for the iLIDS-VID data at Ranks 5 and 10, for the *MARS* data, *T-MAN* outperforms the approach in [54] by a substantially large margin for all the different rank values. Additionally, *T-MAN* has achieved a mean average precision (map) score of 76.7%, which is significantly higher than any of the existing methods. This verifies the effectiveness of our ensemble model in handling large data sets over [54] and other existing approaches.

We next make a comparative study of the re-identification accuracy given by the different ensembles of the feature extractors, namely, *FPAN*, *MPN*, and *LSTM*. These are *Ensemble 1* formed by combining *FPAN* and *MPN*, *Ensemble 2* formed by combining *MPN* and *LSTM*, *Ensemble 3* formed by combining *FPAN* and *LSTM*, and finally, T-MAN formed by combining *FPAN*, *MPN*, and *LSTM* on the extensive *MARS* data set in terms of rank-based accuracy for Rank values 1, 5, and 20 and *map* score. Results presented in the table reveal that the proposed ensembling technique (i.e., *Ensemble 4*) performs the best among the different ensembles in terms of both accuracy at Ranks 1, 5, 20, and *map* score. Similar results have also been observed for each of the other video-based re-identification data sets and to avoid repetition of similar results and discussion these are not presented in the chapter. From these results, we conclude that the ensemble of *FPAN*, *MPN*, and *LSTM* is most suitable for video-based person re-identification.

Table 4.4: Comparative analysis of different combinations of proposed models (*FPAN*, *MPN*, and *T-MAN*) on the *MARS* data set

Methods	**Accuracy (%)**			**Mean Average Precision**
	Rank 1	**Rank 5**	**Rank 20**	(*map*)
Ensemble 1 **(*FPAN+MPN*)**	65.2	76.4	89.0	51.6
Ensemble 2 **(*MPN+LSTM*)**	73.1	89.0	92.6	64.9
Ensemble 3 **(*FPAN+LSTM*)**	79.2	92.1	95.5	74.1
our *T-MAN* **(*FPAN+MPN+LSTM*)**	**83.3**	**93.5**	**95.6**	**76.7**

In our next experiment, we evaluate the robustness of our proposed Temporal Motion Aware Network (*T-MAN*) against various initialization parameters of the three sub-networks. To do this, we first train the individual models three different times and next ensemble these to get three different trained models. We test the performances of each of the above trained models and observe the Rank 1 accuracy. Results are shown in the form of a grouped bar diagram in Figure 4.4. Here, the height of each bar corresponds to the Rank 1 recognition accuracy, and each group of three bars represents the accuracy obtained by running the three different trained models on each of the *PRID-2011*, *iLIDS-VID*, and *MARS* data sets. The data set names and the standard deviation of the recognition accuracy are shown along the horizontal axis. It is observed from the figure that the Rank 1 accuracy for *PRID-2011*, *iLIDS-VID*, and *MARS* data set ranges between [80.2 83.0], [71.0 73.5], and [76.8 83.3], respectively. The standard deviation values for the *PRID-2011* and *iLIDS-VID* data sets are well below 1.5, which implies that the proposed *T-MAN* model is robust against varying initialization parameters of the sub-networks. A slightly higher value of standard deviation (i.e., 2.7) has been observed for the *MARS* data set. This is since the *MARS* data set consists of low-resolution images that pose a significant challenge to the classification algorithms. However, despite its slightly less robustness, the Rank 1 accuracy values obtained from

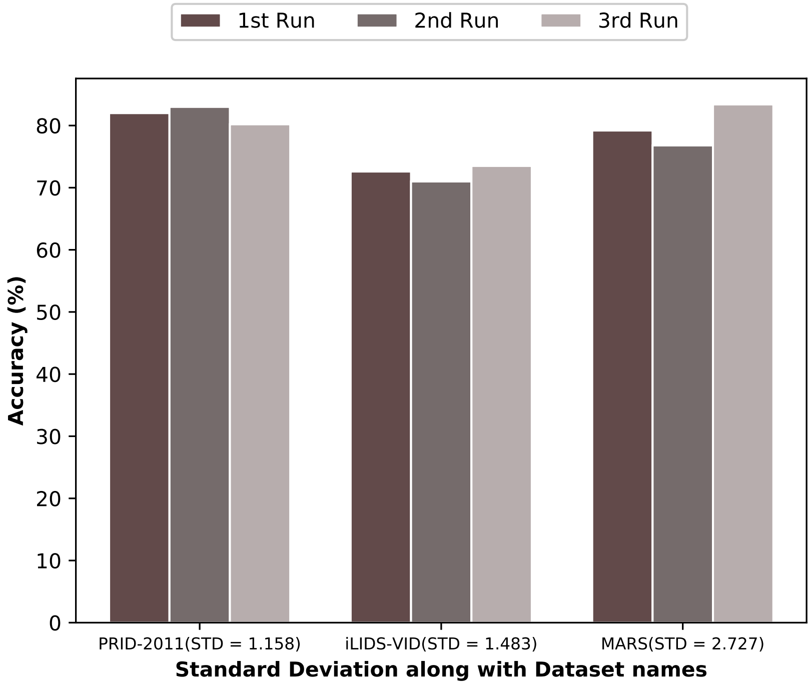


Figure 4.4: Rank 1 accuracy obtained by executing our ensemble *T-MAN* model three times along with the standard deviation

the differently trained models are quite high.

The above experimental results show that the proposed *T-MAN*-based re-identification method is accurate, robust to varying initialization parameters, and, in general, performs better than most existing approaches for the different experimental settings. The effectiveness of the proposed ensemble approach over existing techniques on extensive data sets has also been verified from the results shown in Table 4.3 using the MARS *data* set which consists of low-resolution images from a large number of subjects. Most of the previous approaches working on *MARS* data set have used either a single network model or ignored the important motion-related information from the video sequences. In contrast to the existing techniques, in *T-MAN*, we combine the contextual, motion, and temporal information to carry out person re-identification effectively.

Table 4.5: Comparison of Rank 1 accuracy for Siamese Convolution Box *(SCB)* and Temporal Motion Aware Network *(T-MAN)*

Methods	***Rank 1 accuracy (%)***		
	PRID-2011	***iLIDS-VID***	***IIT (BHU)*** **(*ReID Data*)**
Hierarchical Classification using SCB	72.2	61.8	91.4
T-MAN	83.0	73.5	95.2

In our next experiment, we compare the performance of the *T-MAN*-based re-identification approach described in this chapter with that of the hierarchical classification based re-identification approach using Siamese Convolution Box *(SCB)* as discussed in Chapter 3.2 in terms of Rank 1 accuracy using the *PRID-2011*, *iLIDS-VID*, and *IIT (BHU) Re-identification* data. Table 4.5 shows the corresponding results. The results presented in the table show that *T-MAN* consistently performs better than the hierarchical approach discussed in Chapter 3 corresponding to each of the three data sets. The results also highlight the fact that temporal features are indeed effective to perform video-based person re-identification.

The re-identification performances of *SCB* with and without the clustering step and *T*-

MAN are next studied on the subset of the IIT (BHU) Re-identification data set with similar clothing conditions (refer to Section 2.4.2 of Chapter 2). Results are shown in Table 4.6 in terms of Rank 1 accuracy. It can be observed from the results that here also *T-MAN* performs better than *SCB* without clustering by more than 10% Rank 1 accuracy. It also performs better than *SCB* with the clustering step by 1.8%. The superior performance of *T-MAN* over *SCB* with and without clustering is due to the use of motion-based features instead of spatial features for video-based person re-identification.

Table 4.6: Rank 1 accuracy of Siamese Convolution Box *(SCB)* (with and without clustering) and Temporal Motion Aware Network *(T-MAN)* on a data set with similar clothing conditions

Methods	*Rank 1 accuracy (%)*
	IIT (BHU) ReID Data
SCB without clustering	75.0
SCB with clustering	85.0
T-MAN	86.8

It was observed from Chapter 3 that the hierarchical classification scheme using color-based matching followed by *SCB*-based re-identification improves over single-level *SCB*-based re-identification. It appears that the use of a similar hierarchical classification can improve the re-identification accuracy of *T-MAN* as well, which we are going to study next. Table 4.7 compares the Rank 1 accuracy given by *Hierarchical T-MAN*, i.e., color-based matching followed by *T-MAN*-based re-identification with that of the T-MAN-based re-identification (without the color matching phase) discussed in this chapter on three video data sets, namely, *PRID-2011*, *iLIDS-VID*, and *IIT (BHU) Re-identification* data. As expected, it can be seen from the table that the *Hierarchical T-MAN* usually improves upon the accuracy of the *T-MAN*.

It can be concluded from the above experiments for video-based person re-identification, *T-MAN* is more effective than *SCB* discussed in Chapter 3 since it has the potential to extract useful spatial and temporal features from the sequential frames, which *SCB* cannot capture. However, for image-based re-identification, *T-MAN* should not be applied since the required sequential information is not available in the image frames.

Table 4.7: Comparison of Rank 1 accuracy for simple *T-MAN* and *Hierarchical T-MAN*

	Rank 1 accuracy (%)		
Methods	***PRID-2011***	***iLIDS-VID***	***IIT (BHU)*** **(*ReID Data*)**
Hierarchical T-MAN	83.4	73.5	95.6
T-MAN	83.0	73.5	95.2

4.3 Summary

In this chapter, we focus on video-based person re-identification and present a video-based person re-identification approach capable of exploiting spatial as well as temporal information from the video sequences. Specifically, we propose three different deep models, namely, Full-Body Pose Attention Network (*FPAN*), Motion Pooling Network (*MPN*), and Long-Short Term Memory (*LSTM*) to capture attention features, shape-related features, and temporal features, respectively, and ensemble these by fusing their predictions to determine the final class of a test subject. Extensive experiments show that the T-MAN-based ensembling approach outperforms the existing video-based re-identification techniques in most of the experimental settings. Also, it has been seen to work better than the *SCB*-based re-identification technique discussed in Chapter 3 for video data sets as well as for situations where subjects wear similar colored clothes. However, at any real-life surveillance site, it is quite difficult to capture clean images/videos of individuals due to occlusion caused by other static/dynamic objects. The presence of occlusion corrupts the input image frames providing misleading information about the color and pose of the target subject. The techniques discussed in Chapters 3 and 4 heavily rely on the above factors due to which these methods are not expected to be effective in handling occluded sequences. In the next two chapters, we will specifically focus on handling occlusion by employing Deep Neural Network-based generators to reconstruct the occluded frames effectively.