applied sciences

MDPI

Article

Non-Local Spatial and Temporal Attention Network for Video-Based Person Re-Identification

Zheng Liu [1,2,3], Feixiang Du [1,2,3], Wang Li [1,2,3], Xu Liu [1,2,3] and Qiang Zou [1,2,3,4,*]

[1] School of Microelectronics, Tianjin University, Tianjin 300350, China; liuzheng_2020@tju.edu.cn (Z.L.); feixiangdu@tju.edu.cn (F.D.); wangli2020@tju.edu.cn (W.L.); liuxu_@tju.edu.cn (X.L.)

[2] Tianjin International Joint Research Center for Internet of Things, Tianjin 300350, China

[3] Tianjin Key Laboratory of Imaging and Sensing Microelectronic Technology, Tianjin 300350, China

[4] Haier Internet of Clothing Intelligent Ecosystem Research Institute, Tianjin 300350, China

[*] Correspondence: zouqiang@tju.edu.cn

Received: 13 July 2020; Accepted: 24 July 2020; Published: 4 August 2020

check for updates

Abstract: Given a video containing a person, the video-based person re-identification (Re-ID) task aims to identify the same person from videos captured under different cameras. How to embed spatial-temporal information of a video into its feature representation is a crucial challenge. Most existing methods have failed to make full use of the relationship between frames during feature extraction. In this work, we propose a plug-and-play non-local attention module (NLAM) for frame-level feature extraction. NLAM, based on global spatial attention and channel attention, helps the network to determine the location of the person in each frame. Besides, we propose a non-local temporal pooling (NLTP) method used for temporal features' aggregation, which can effectively capture long-range and global dependencies among the frames of the video. Our model obtained impressive results on different datasets compared to the state-of-the-art methods. In particular, it achieved the rank-1 accuracy of 86.3% on the MARS (Motion Analysis and Re-identification Set) dataset without re-ranking, which is 1.4% higher than the state-of-the-art way. On the DukeMTMC-VideoReID (Duke Multi-Target Multi-Camera Video Reidentification) dataset, our method also had an excellent performance of 95% rank-1 accuracy and 94.5% mAP (mean Average Precision).

Keywords: person Re-ID; video; non-local; spatial-temporal attention

1. Introduction

Person re-identification (Re-ID) aims to use computer vision algorithms for cross-camera tracking, which means finding the same person under different cameras. Person Re-ID intends to identify a probe person in a camera by matching his/her images or videos and has many practical applications, including intelligent surveillance and criminal investigation. Person Re-ID can be divided into image-based and video-based person Re-ID. Image-based person Re-ID has made significant progress in terms of both solutions [1,2] and the construction of large benchmark datasets [3,4]. Recently, more work [5–7] has begun to focus on video-based person Re-ID because of the richer information contained in video data as compared to image data. By extracting more spatial and temporal cues from video data, video person Re-ID has the potential to solve some of the challenges faced in image person Re-ID, e.g., the visual blocking of pedestrians as they walk.

In the video-based person Re-ID task, the video-based dataset is composed of many consequent sequences of images rather than static images. Here, we need to declare that the video is composed of several sequences, and a sequence includes several frames of images in this article. The critical challenge is to make use of the temporal clues embedded in the sequences. Some previous work [5–7] has typically divided this task into two steps. In the first step, image-based convolutional neural networks (CNNs) are used to extract features from each frame in the video to obtain frame-level features. The second step is to aggregate the features based on each frame to form a video-level feature that can represent the entire video. The distance between the video-level features of the input video is generally used to indicate whether the video contains the same target person or not. Ideally, the probability of including the same person in a video is higher if the distance between two video-level features is smaller and lower for two videos that are further apart.

The main problem with video-based person Re-ID now lies in the second step, which is temporal feature aggregation. Depending on their ways of temporal feature learning, existing work can be roughly divided into the following three categories:

1. Extraction of dynamic features from other CNN inputs, e.g., by optical flux [8].
2. Extracting spatial and temporal features by regarding the video as 3-D data, e.g., via 3-D CNN [9,10].
3. Learning robust person representations by temporally aggregating frame-level features, e.g., through recurrent neural networks (RNNs) [5,8,11] and temporal pooling or weights [7,12,13].

The third category to which our work belongs is currently dominant in video-based person Re-ID tasks. Most existing methods represent the frame of the video as a feature map and then use an average or maximum pooling across frames to obtain a representation of the input video. However, this approach tends to fail when occlusions are frequent in the video because it processes all images in the video with equal importance. In order to distill the relevant information from a video and weaken the influence of noisy samples, some works have learned the temporal attention score of each frame in a given video by using recurrent neural networks (RNNs) to solve this problem. The limitation of the RNN method is that it requires sequential calculations to be performed. As a result, it is difficult to compute in parallel and make full use of the graphics processing unit (GPU) hardware. Additionally, a single recurrent operation could only calculate the dependency between the current and the latest frame. In general, it is difficult for RNNs to capture long-range dependencies.

In this paper, we propose a non-local spatial and temporal attention network for video-based person Re-ID. We improve the non-local neural network [14] and apply it to the video-based person Re-ID task with excellent results. The novelty of our approach is that we use non-local neural networks to compute spatial and temporal dependencies over long ranges among video frames. The way each attention score is calculated depends on all the frames in the video, as shown in Figure 1, not just on the adjacent frames. This method gives a better video-level feature representation, making the video look more like a whole rather than merely a few images. Additionally, we apply the improved non-local neural network to CNN networks at different levels, so that the features at different levels obtain a better performance. We performed both frame-level feature extraction, and temporal aggregation using the non-local attention mechanism. Our main contributions can be summarized in three-fold:

1. We propose a plug-and-play non-local attention module (NLAM). It can be inserted into CNN networks for frame-level feature extraction. In the video-based person Re-ID task, the spatial position of the target person in the image can be determined more accurately.
2. We propose a non-local temporal pooling (NLTP) method for temporal feature aggregation. We use it to replace the single average or maximum pooling, which could not order the video frames.
3. We verified the effectiveness of our two methods on different datasets.

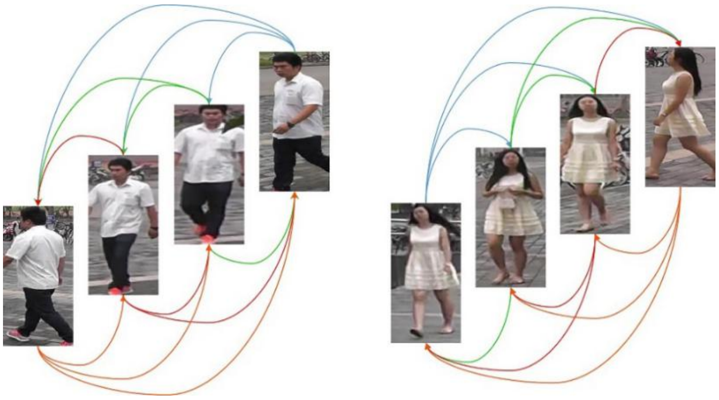

Figure 1. Some examples of the application of non-local attention in our network. The starting point of the arrow represents one frame in a video, and the picture pointed by the arrow represents another frame of the video. For brevity, we only selected four frames in the figure to show how our model directly finds relevant clues in frames to support its prediction. The 'blue' arrow represents the similarity between the first frame and the remaining frames, and the 'green', 'red', and 'orange' respectively represent the similarity of the second, third, and fourth frames with the remaining frames.

2. Related Works

Person Re-ID has always been a hot field in computer vision. In this section, we review the development of video-based person Re-ID, from image-based person Re-ID to video-based person Re-ID. Additionally, we introduced a video-based person Re-ID pipeline.

2.1. Image-Based Person Re-ID

The purpose of image-based person Re-ID is to match a given probe image to the same person in a set of images (gallery images) captured by another non-overlapping camera. Existing methods are usually divided into two steps to complete this work: (1) Extract special vectors, and (2) calculate the similarity of the two feature vectors. With the continuous development of the CNN network [15–19], learning image features from the CNN network has replaced hand-made features [3,20–22] to represent person images. After extracting features from the image, the metric distance is used to calculate the similarity/dissimilarity between the features of the two images. Ideally, if two images contain the same person, the distance should be smaller than two images that do not contain the same person. As suggested by Zheng et al. [23], the calculation of feature vectors can be used for discriminant learning and metric learning. Discriminant learning uses cross-entropy loss [17,19] to learn the deep features used for identity classification. Metric learning uses triple loss to increase the distance among classes and reduce the distance within classes. In our work, we use both loss functions to train our network.

2.2. Video-Based Person Re-ID

Video-based person Re-ID can be seen as an extension of image-based person Re-ID efforts. Compared to static images, video can provide richer information for person Re-ID tasks because it contains both spatial and temporal information. Video-based person Re-ID is also closer to the real world for better application. So, in recent years, video-based person Re-ID has also attracted the attention of more researchers. Some early work [5,24,25] considered frame-level similarities to identify the person. Recently, deep learning methods have been applied to gain more discriminative video-level features. They first trained the CNN network to extract image features and then aggregated them into video features through average or maximum pooling. Mc Laughlin et al. [5] proposed a method for extracting time information using RNN and the temporal pooling layer. Following [5], Xu et al. [7] proposed a spatial and temporal attention pooling network (STAPN), which extracts more robust features by calculating attention in the spatial and temporal dimensions. Li et al. [13] proposed a new spatial-temporal attention model to distinguish different body parts automatically.

2.3. A Video-Based Person Re-ID Pipeline

In our article, we follow the state-of-the-art structure that has been summarized by previous researchers and is the most commonly used base structure for video-based person Re-ID works. It mainly consists of two parts: (1) Feature extraction: This part can extract meaningful abstract spatial representations from video frames through pre-trained ImageNet Models [26,27], such as residual network (ResNet50 [26]) and squeeze-and-excitation residual network (SE-ResNet50 [27]). (2) Temporal feature aggregation: In this part, the frame-level features extracted in the previous step are aggregated into video-level features. Gao et al. [28] summarized that the feature aggregation method could be roughly divided into three types: Average temporal pooling (TP_{avg}) operation, temporal attention (TA), and the RNN layer. Subramaniam et al. [29] compared different feature extraction methods and temporal feature aggregation methods. The comparison results are shown in Table 1. Two main conclusions can be drawn from the comparison results: First, the choice of the backbone network will affect the overall performance of the system, and SE-ResNet50 has a better performance than ResNet50. Second, TP_{avg} is superior to attention/RNN. Therefore, we chose SE-Resnet50 + TP_{avg} as the baseline of our work.

Table 1. Comparison of different basic frameworks on MARS (Motion Analysis and Re-identification Set) [30] and DukeMTMC-VideoReID (Duke Multi-Target Multi-Camera Video Reidentification) [31]. TP_{avg}, TA, RNN stand for average temporal pooling, temporal attention, and recurrent convolution network, respectively. The best results are shown in bold.

Feature	Temporal Aggregation	MARS				DukeMTMC-VideoReID			
Extractor		mAP	R1	R5	R20	mAP	R1	R5	R20
ResNet50	TP_{avg}	75.8	83.1	92.8	96.8	92.9	93.6	**99.0**	**99.7**
ResNet50	TA	76.7	83.3	93.8	**97.4**	93.2	93.9	98.9	99.5
ResNet50	RNN	73.8	81.6	92.8	96.7	88.1	88.7	97.6	99.3
SE-ResNet50	TP_{avg}	**78.1**	84.0	**95.2**	97.1	**93.5**	93.7	**99.0**	**99.7**
SE-ResNet50	TA	77.7	**84.2**	94.7	**97.4**	93.1	**94.2**	**99.0**	**99.7**
SE-ResNet50	RNN	75.7	83.1	93.6	96.0	92.4	94.0	98.4	99.1

3. Our Approach

In this part, we accurately describe our network structure and the innovations we propose in the network. In Section 3.1, we explain the role of our proposed NLAM in the frame-level feature extraction process. In Section 3.2, we take one sequence of a video as an example to analyze our proposed NLTP method. In Section 3.3, we explain the loss function we adopted. The structure of the entire network is shown in Figure 2:

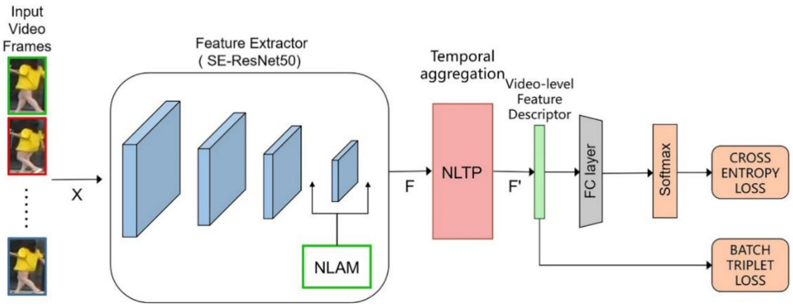

Figure 2. Our overall network architecture. X means the input video; F represents the frame-level features after passing through the feature extractor; F' refers to the resulting video-level representation. NLAM = Non-Local Attention Module; NLTP = Non-Local Temporal Pooling; FC = Fully Connected layer.

Our overall network architecture is shown in Figure 2. Similar to a standard video-based Re-ID framework, it mainly includes two parts: Feature extractor and temporal feature aggregation. The difference is that we insert the non-local attention module (NLAM) we proposed in the feature extractor, and we adopt the non-local temporal pooling (NLTP) method we proposed in the temporal feature aggregation part. NLAM is used to insert between CNN blocks for spatial attention and channel attention extraction. It helps CNN to determine the location of the target person in each frame and reduce the interference caused by occlusion in the image. NLTP improves on the previous temporal pooling method by acquiring temporal attention in a non-local way in the first step and embedding temporal features into video-level features through pooling in the second step. The primary purpose of NLTP is to give a higher weight to frames that are more representative of the entire sequence, thereby obtaining a more robust video-level representation.

3.1. Frame-Level Feature Extraction

In the process of frame-level feature extraction, we use the most modern image recognition network architecture SE-ResNet50 as a feature extractor in video-based Re-ID. SE-ResNet50 contains five consecutive CNN blocks (one initial convolution block, followed by four successive squeeze-and-excitation (SE) residual blocks). We argue that a single CNN is not sufficient for feature extraction of an image. With the addition of an attentional mechanism, CNN can extract more critical spatial information from the image, similar to human visual attention. We add NLAM between CNN blocks to obtain a better frame-level feature representation. The overall structure of NLAM is shown in Figure 3.

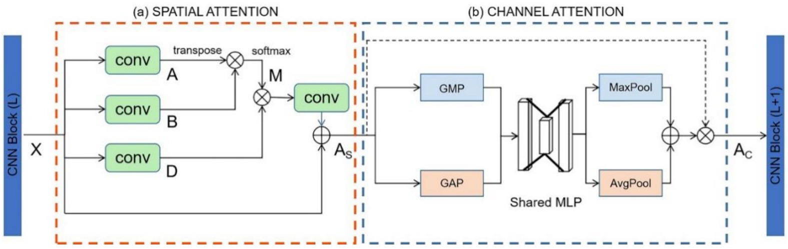

Figure 3. Non-local attention module (NLAM) inserted between the L^{th} and $L+1^{th}$ blocks of CNN. NLAM mainly includes two parts: (**a**) NLAM spatial attention; (**b**) NLAM channel attention. X represents the feature maps output by the L^{th} CNN block; A_S expresses the feature maps after NLAM spatial attention; A_C indicates the feature maps after NLAM channel attention as the input of the $L+1^{th}$ block. '⊕' denotes the addition operation based on elementwise. '⊕' denotes matrix multiplication. The green box represents 1×1 convolution. GMP = Global Maximum Pooling, GAP = Global Average Pooling, MLP = Multi-Layer Perceptron.

In the NLAM spatial attention part, we aim to perform spatial attention calculation on the feature maps output by the CNN network in the previous layer. Given an input feature tensor $\mathrm{X} \in \mathbb{R}^{N\times C\times H\times W}$, it is obtained from a sequence of N feature maps of size $\mathrm{C}\times\mathrm{H}\times\mathrm{W}$. We aim to exchange their spatial information between all frames in the sequence to determine a better position of the target person in the image and reduce the interference of occlusion between frames. Let x_i be sampled from X. First, we reduce the dimension of the input feature channel to C' through three 1×1 convolution blocks (a, b, d) to obtain $\mathrm{A},\ \mathrm{B},\ \mathrm{D} \in \mathbb{R}^{C'\times NHW}$. The transposition of A is multiplied by B to obtain the attention score of all positions x_j at position x_i by using embedded Gaussian instantiation. Then, the weighted average $\mathrm{M} \in \mathbb{R}^{NHW\times NHW}$ of the attention scores of all positions x_j is used to calculate the response y_i of each position x_i. Finally, Y is recovered to the same size as the input X by 1×1 convolution, and the recovered result is added to the original feature tensor X to obtain the final result A_S. NLAM spatial attention can be formulated as follow:

$$y_i = \frac{1}{\sum_{\forall j} e^{a(x_i)^T b(x_j)}} \sum_{\forall j} e^{a(x_i)^T b(x_j)} d\left(x_j\right), \tag{1}$$

$$A_S = W_0(\mathrm{Y}) + \mathrm{X}. \tag{2}$$

Equation (1) represents the process of non-local operations, and the overall spatial attention is formulated as Equation (2). Here, $\mathrm{i,j} = [1,\ NHW]$ refers to all locations of each feature map and in all frames. The convolution operation is expressed by a, b, and c. W_0 recovers Y to the same size as the input tensor X. The idea contained in the non-local operation is that when extracting features at a specific location in a particular time, the network should consider the spatial and temporal dependencies within the sequence by attending on the non-local context.

In the NLAM channel attention part, we pass the feature maps A_S outputted from Figure 3a through global max pooling (GMP) and global average pooling (GAP) based on the width and height, and then through a multi-layer perceptron (MLP) to get the channel importance of each frame. The obtained channel importance vectors of all N frames are respectively subjected to maximum pooling and average pooling in each dimension to estimate the global channel importance. Then, the features of maximum pooling and average pooling output are subjected to the elementwise addition operation followed by sigmoid activation to obtain the final channel attention feature maps. Then, the features of maximum pooling and average pooling output are subjected to the elementwise addition operation followed by sigmoid activation to obtain the final channel attention feature maps. Finally, the channel attention feature maps and input feature maps are elementwise multiplied to generate the final output A_C of NLAM, which is used as the input of the $L+1^{th}$ layer CNN network. The channel attention map is computed as follows:

$$A_C = \sigma\{Max[W_2\delta(W_1(GMP(A_S)))] + Avg[W_2\delta(W_1(GAP(A_S)))]\}, \tag{3}$$

where σ refers to the sigmoid activation function. A_S stands for NLAM spatial attention output. A_C refers to the final output of NLAM channel attention and the final result of NLAM. GMP and GAP are the same as Figure 3b, which represents global maximum pooling and global average pooling; note that the MLP weights, W_1 and W_2, are shared for both inputs and the δ (Tanh) activation function is followed by W_1.

3.2. *Temporal Aggregation*

In video-based person Re-ID, a key challenge is how to combine frame-level features into video-level features to express the temporal features in the video better. In previous works, researchers generally used temporal pooling to perform temporal feature aggregation. Table 1 makes a detailed comparison of three different temporal aggregation layers (TPavg, TA, RNN), from which we can see that temporal pooling shows the best performance indicators. However, temporal pooling naturally ignores the temporal relationship between frames. The calculation of each attention score in non-local attention depends on all the frames in the sequence, so it can capture the remote dependencies in the deep neural network well.

In this part, we propose a non-local temporal pooling (NLTP) method for temporal feature aggregation. The specific architecture is shown in Figure 4. Our proposed method aims to use the non-local attention mechanism to frame-level feature sequences in the temporal dimension. Our proposed method aims to use the non-local attention mechanism for the extracted frame-level features to perform feature aggregation in the temporal dimension. Enhancing the frame-to-frame relationship allows frames with a closer relevance to obtain a higher weight, resulting in a more reliable person Re-ID model. The NLTP operation we proposed is as follows:

$$F' = \mathrm{Avg}[W(Y) + F], \tag{4}$$

where F represents the frame-level features of a sequence extracted through Section 3.1. Equation (4) represents the entire NLTP process, and the result Y after the non-local operation is restored to the same scale as the input F through W (1×1 Conv) and elementwise addition is performed with F. Finally, the average pooling operation is performed in the temporal dimension to obtain the final output result F' of NLTP.

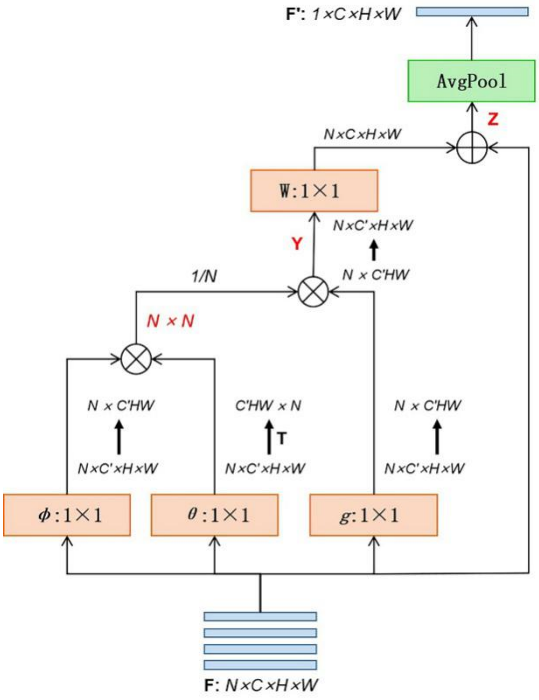

Figure 4. Description of non-local temporal attention (NLTP). The size of the input feature maps F is N × C × H × W (N represents the number of frames included in the input video, C is the number of channels of each frame feature map, and H and W indicate the height and width of the input feature map, respectively). The symbol 'T' stands for the transpose function. '$1/N$' is used for normalization. '⊕' denotes the addition operation based on elementwise. '⊗' denotes matrix multiplication. The yellow box defines four different 1 × 1 convolution blocks. We used the dot product method to calculate the attention score and generate an N × N attention matrix.

We assume that a continuous image sequence contains N frames, and $f_i(i \in \{1, 2, \ldots, N\})$ is one frame of N frames of the video. We use the function $\mathrm{h}\big(x_i, x_j\big) \in \mathbb{R}^{N \times N}$ to calculate the scale value between the current frame x_i and each frame $x_j(j \in \{1, 2, \ldots, N\})$. There are many different choices for the pairwise function $\mathrm{h}\big(x_i, x_j\big)$ [14]. In our work, we use the "dot product" pairwise function to compute the correlation between frames. Recall that each frame is represented as a C × H × W tensor (see Section 3.1). We apply a 1 × 1 convolution on f_i to reduce its channel dimension to $C' = C/2$ as a way to reduce computation. Then, f_i is reshaped to a vector. We use $\theta(f_i)$, $\varphi(f_j)$, and $g(f_j)$ to indicate three such vectors. The pairwise function is then defined as the point product between $\theta(x_i)$ and $\varphi(x_j)$. The pairwise function is defined as:

$$h_{(f_i,f_j)} = \theta(x_i)^T \varphi(x_j). \tag{5}$$

Then, we multiply the output of the pairwise function by $1/N$ as the normalization operation. We call the normalized result the "attention score" to indicate the influence of all frames f_j on f_i. We then compute a "weighted frame feature" y_i using the attention scores and frames $g(f_j)$. The weighted frame feature y_i is as follows:

$$y_i = \frac{1}{N} \sum_{\forall j} h(f_i, f_j) g(f_j). \tag{6}$$

Note that since y_i is computed based on all frames in the video, y_i implicitly contains information of the frame f_i and all the other frames f_j in the video. To obtain the video-level feature, we simply perform temporal pooling over these weighted frame features and original frame features. Since the weighted frame features already capture long-range dependencies in the video, the output (e.g., video-level feature) of the temporal pooling will implicitly capture rich long-range dependencies in the video.

3.3. Loss Function

We use Softmax cross-entropy loss and batch triplet loss as a loss function for our work. On the one hand, these two loss functions are used for a fair comparison with the baseline. On the other hand, because these two loss functions are proved to be our work is very suitable. We randomly select P identity samples and randomly select K sequences (each sequence contains N frames) from each identity sample to form a batch. Therefore, a batch contains P × K sequences. The overall loss function can be described as:

$$\mathrm{L} = L_{softmax} + L_{triplet}, \tag{7}$$

where $L_{softmax}$ and $L_{triplet}$ refer to the cross-entropy loss and batch triplet loss, respectively.

The cross-entropy loss function encourages the network to classify the P × K sequences to the correct identities. The cross-entropy loss function is defined as follows:

$$\mathrm{L}_{softmax} = -\frac{1}{B} \sum_{i=1}^{B} p_i \log q_i, \tag{8}$$

where p_i and q_i are the groundtruth identity and the prediction of sample i. B represents all sequences in a batch.

Batch triplet loss is generally used to reduce the intra-class distance between each sequence and to increase the inter-class distance. The training instances contain an anchor, a positive instance, and a negative instance. The positive instance belongs to the same class as the anchor, and the negative instance belongs to a different class than the anchor. Let $\{f_{I_A}, f_{I_P}, f_{I_N}\}$ be the video-level descriptors of three different sequences, where I_A, I_P, and I_N are the anchor, positive, and negative examples, respectively. The triplet loss function is defined as:

$$L_{triplet} = \sum_{i=1}^{B} \left[m + D(f_{I_A}, f_{I_P}) - D(f_{I_A}, f_{I_N}) \right]_+, \tag{9}$$

where m is the margin between positive and negative samples, and $\mathrm{D}(i, j)$ indicates the distance function between two video-level descriptors i, j. B represents all sequences in a batch, and I represents the I^{th} sequence in a batch.

The Softmax cross-entropy loss function follows the fully connected (FC) layer for probabilities obtained for the identities. The batch triplet loss is applied to the video-level descriptors to backpropagate the gradients.

4. Experiment

In this part, we introduce the datasets used in the training process, the evaluation method used after the training is completed, and some parameter settings throughout the experiment. Finally, our experimental results are listed and explained.

4.1. Datasets and Evaluation

We evaluated the proposed model on two commonly used video-based person Re-ID datasets: MARS, DukeMTMC-VideoReID.

MARS: The MARS [30] dataset is an extended version of the Market1501 [3] dataset and is also the first large-scale video-based dataset. Since all bounding boxes and tracks are automatically generated, it contains disruption terms, and each identification may contain multiple tracks. It is the largest video-based person Re-ID dataset with 1261 identities and 20,478 videos, with multiple frames per person captured across six non-overlapping camera views. Among the total identities, about half of the identities are used for training, and the other half are used for testing. Additionally, the MARS dataset includes 3248 identities (disjoint with the train and test set) that are used as distractors.

DukeMTMC-VideoReID: The DukeMTMCVideoReID [31] is a subset of the DukeMTMC multicamera dataset [32], which was collected on an outdoor scenario with varying viewpoints, illuminations, backgrounds, and occlusions using eight synchronized cameras. The dataset contains 1404 identities for training and testing and 408 identities as distractors. In total, there are 2196 videos for training and 2636 videos for testing. Each video contains person images sampled every 12 frames. During testing, a video for each ID is used as the query, and the remaining videos are placed in the gallery.

In Table 2, a detailed comparison of the two datasets MARS and DukeMTMC-VideoReID is shown from the following aspects:

- The total number of people included;
- The number of people used for training;
- The number of people used for testing;
- The number of people used as distractors;
- The total number of videos contained in the dataset; and
- The number of cameras used in data collection.

Table 2. Comparison of MARS and DukeMTMC-VideoReID, Duke = DukeMTMC-VideoReID dataset.

Dataset	Identity	Train	Test	Distractor	Video	Camera
MARS [30]	1261	625	636	3248	20478	6
Duke [31]	1404	702	702	408	4832	8

We used the same evaluation indicators as those used in the literature [12,13,30,33]: CMC (cumulative matching characteristics) and mAP (mean average precision). CMC refers to the probability of finding the correct identity among the first k matches based on the retrieval ability of the algorithm. We chose to use CMC when only one gallery instance exists for every identity. We tested the probability of rank-1, rank-5, and rank-20. The mAP metric is used when there are multiple instances of the same identity in the gallery.

4.2. Implementation Details

The proposed method was implemented using the PyTorch framework [34]. During training, each sequence consists of N = 8 frames, which is somewhat different from the baseline, and the video frames are resized to 256 × 128. It should be noted that during training, we used a random approach to obtain N frames from the video to form a sequence as input. In testing, we split the video into several sequences of length N in temporal order. The network was trained using the Adam optimizer. The batch size was set to 32, and if the total memory usage was over the GPU memory limit, the batch size was reduced accordingly to the maximum possible extent. The learning rate was initialized to 0.0001, while the learning rate decreased as the number of epochs increased with the parameter $\gamma = 0.1$. The margin of triple loss was $m = 0.3$. We trained the network for 800 epochs, and the learning rate was multiplied by 0.1 after every 200 epochs.

4.3. Result

In our experiments, first, every video of the person was divided into multiple sequences containing N frames, then each sequence was passed through the network to obtain sequence-level features, and finally, the sequence-level features were averaged to obtain a video-level descriptor. We used the L_2 distance to calculate CMC and mAP. The following is a comparative analysis of some experimental data:

Location of the NLAM within the network: In the first step, we explored the effect of NLAM at different locations within the network. We inserted an NLAM layer after one or more feature extraction CNN blocks to compare their capability. In order to ensure the uniqueness of variables, we used NLTP as the temporal aggregation layer. The network was trained and tested on two datasets, MARS and DukeMTMC-VideoReID. The results are shown in Table 3. It can be derived from Table 3 that when we added a single NLAM layer to the network, the network performed better when inserted into deeper blocks (e.g., block3, block4, block5). So, we tested the insertion of multiple NLAM layers for the deeper blocks in Table 4. We found that inserting the NLAM layer after block4 and block5 achieved the best results. The results are as follows:

Table 3. Evaluation of the effect of inserting a single NLAM layer after different CNN blocks. The number in the location column represents adding the NLAM layer after this CNN block.

Location	**MARS**				**Duke**			
	mAP	**R1**	**R5**	**R20**	**mAP**	**R1**	**R5**	**R20**
2	70.6	80.6	92.3	96.3	92.7	94.0	98.8	99.6
3	**79.5**	85.7	94.8	97.5	93.8	94.7	98.9	99.8
4	79.4	**85.8**	95.2	**97.6**	94.0	94.9	**99.1**	**99.9**
5	**79.5**	85.5	**95.3**	97.5	**94.1**	**95.0**	99.0	**99.9**

Table 4. Evaluation of the effect of inserting a single NLAM layer after different CNN blocks simultaneously. The number in the location column indicates that the NLAM layer is added after these CNN blocks. For example, '3,4' represents that an NLAM layer is inserted after the third and fourth CNN blocks, respectively.

Location	**MARS**				**Duke**			
	mAP	**R1**	**R5**	**R20**	**mAP**	**R1**	**R5**	**R20**
3,4	79.3	86.1	94.9	97.6	94.0	94.2	98.9	99.8
3,5	79.5	85.3	95.7	97.7	94.0	94.8	99.1	**99.9**
4,5	79.8	**86.3**	**95.8**	**97.8**	**94.5**	95.0	**99.3**	**99.9**
3,4,5	**80.1**	86.0	95.5	97.4	94.4	**95.1**	99.0	**99.9**

Different temporal aggregation methods: In order to directly compare the superiority of our proposed NLTP feature aggregation method, we compared our proposed method with the temporal pooling method used in the baseline (Table 1). Both experiments were conducted based on using SE-ResNet50 as the feature extractor and adding the NLAM module after the fourth and fifth CNN layers to make sure there was only one variable. Table 5 shows the performance evaluation of the model. Compared with the temporal pooling method, our proposed NLTP method achieved a better performance. Especially on the MARS dataset, NLTP improved mAP by 0.2% and increased the accuracy of rank-1 by 0.5%. The specific data are shown in Table 5.

Table 5. Different time feature extraction methods NLTP and TP are compared on the MARS and DukeMTMC-VideoReID datasets. NLTP = non-local temporal pooling, TP = temporal pooling.

Temporal Aggregation	MARS			Duke		
	mAP	R1	R5	mAP	R1	R5
TP	79.6	85.8	95.7	94.1	94.9	99.1
NLTP (ours)	**79.8**	**86.3**	**95.8**	**94.5**	**95.0**	**99.3**

Effect of different sequence lengths (N): In this step, we studied the impact of the different number of frames in each sequence on our model. N represents the length of the sequence which is the number of frames we captured from a video. We compared the performance of the model on the MARS dataset at N = 2, 4, 8, and the results are shown in Table 6. In order to ensure the uniqueness of the experimental variables, we conducted an experimental comparison based on the SE-ResNet50 with the NLAM module added after the fourth and fifth CNN layers as the feature extractor and NLTP as the temporal aggregation layer. It can be seen from Table 6 that our network achieved a better performance when $N = 8$, which is different from the conclusion in our baseline (the model performs best when $N = 4$). In the MARS dataset, the accuracy of Rank1 was improved by 0.8% relative to $N = 4$, and mAP was improved by 0.9% relative to $N = 4$. Such a result is also expected because for both the NLAM and NLTP, we inserted a non-local mechanism. Hence, a more extended sequence is more helpful for our model to extract long-range dependencies and obtain a more robust video-level feature descriptor.

Table 6. Evaluation of the effect of different sequence lengths N on the MARS dataset on our best model (SE-RestNet50 + NLAM (4,5) + NLTP). NLAM (4,5) means to add the NLAM layer after the fourth and fifth CNN blocks.

Sequence Length	MARS		
	mAP	R1	R5
N = 2	77.1	83.6	94.2
N = 4	78.9	85.5	95.3
N = 8	**79.8**	**86.3**	**95.8**

Comparison with state-of-the-art methods: We compared our method with the state-of-the-art method [15,22,28,33,35–38] in the MARS and DukeMTMCVideoReID datasets. The results are shown in Table 7. Our final model selection was tested on the basis of $N = 8$ using SE-ResNet50 with the NLAM module added after the fourth and fifth CNN layers as the feature extractor and NLTP as the temporal aggregation layer. It is observed that our proposed model achieved a good performance. Especially in the MARS dataset, our method improved by 2.3% on CMC Rank-1 and nearly 1.8% on mAP compared to our baseline (Table 1). Compared with the state-of-the-art method [29], our method also improved the CMC Rank-1 by 1.4%. Our model also achieved impressive results in the DukeMTMCVideoReID dataset. Compared to the baseline (Table 1), our network improved by 0.9% and 1.3% on mAP and CMC Rank-1, respectively. We attribute this improvement to NLAM in frame-level feature extraction and NLTP in temporal feature aggregation to better obtain global information, resulting in a more robust feature representation.

Table 7. Comparison of our model with our baseline and a series of state-of-the-art models on the two datasets MARS and DukeMTMCVideoReID. TP_{avg} = average Temporal Pooling. NLAM (4,5) means to add the NLAM layer after the fourth and fifth CNN blocks.

Network	**MARS**		
	mAP	**R1**	**R5**
JST-RNN [6]	50.7	70.6	90.0
Context Aware Parts [36]	56.1	71.8	86.6
Region QEN [37]	71.1	77.8	88.8
TriNet [15]	67.7	79.8	91.4
Comp. Snippet Sim. [38]	69.4	81.2	92.1
Part-Aligned [33]	72.2	83.0	92.8
RevisitTempPool [28]	76.7	83.3	93.8
[28] + SE-ResNet50 + TP_{avg} (Baseline)	78.1	84.0	95.2
SE-ResNet50 + COSAM + TP_{avg} [29]	**79.9**	84.9	95.5
SE-ResNet50 + NLAM (4,5) + NLTP (ours)	79.8	**86.3**	**95.8**
Network	**Duke**		
	mAP	**R1**	**R5**
ETAP-Net [31]	78.3	83.6	94.6
RevisitTempPool [28]	93.2	93.9	98.9
[28] + SE-ResNet50 + TP_{avg} (Baseline)	93.5	93.7	99.0
SE-ResNet50 + COSAM + TP_{avg} [29]	94.1	**95.4**	**99.3**
SE-ResNet50 + NLAM (4,5) + NLTP (ours)	**94.5**	95.0	**99.3**

5. Conclusions

Person Re-ID based on video is an important task that has received much attention in recent years. In this paper, we proposed a non-local attention model (NLAM) that can be added between CNN blocks for frame-level feature extraction and a non-local temporal pooling (NLTP) method for temporal feature aggregation. The experiments showed that the two methods we proposed have shown excellent results on the video-based person Re-ID datasets. Compared with most existing methods, the advantage of our proposed network architecture (SE-ResNet50 +NLAM (4,5) + NLTP) is that it better describes the relationship between frames in the video. It focuses on the spatial and temporal relationships of all frames in a non-local way and gives different weights, thus forming a more accurate representation of the video. The results performed better compared to state-of-the-art methods. Our proposed NLAM and NLTP methods can also be applied to other video-based tasks, such as target tracking and pose estimation.

Author Contributions: Conceptualization, Z.L. and Q.Z.; methodology, Z.L. and W.L.; software, Z.L. and W.L.; validation, Z.L., W.L. and F.D.; formal analysis, Z.L.; investigation, Z.L. and X.L.; resources, Z.L., W.L. and Q.Z.; data curation, Z.L.; writing—original draft preparation, Z.L.; writing—review and editing, Z.L., Q.Z. and F.D.; visualization, Z.L.; supervision, Q.Z.; project administration, Q.Z.; funding acquisition, Q.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Project of Supporting Plan of Tianjin (China) (16-YFZCSY00850).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chi, S.; Li, J.; Zhang, S.; Xing, J.; Qi, T. Pose-Driven Deep Convolutional Model for Person Re-identification. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
2. Li, J.; Zhang, S.; Tian, Q.; Wang, M.; Gao, W. Pose-Guided Representation Learning for Person Re-Identification. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *99*, 1. [CrossRef] [PubMed]
3. Zheng, L.; Shen, L.; Tian, L.; Wang, S.; Tian, Q. Scalable Person Re-identification: A Benchmark. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–14 December 2015.

4. Wei, L.; Zhang, S.; Gao, W.; Tian, Q. Person Transfer GAN to Bridge Domain Gap for Person Re-Identification. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
5. Mclaughlin, N.; Rincon, J.M.D.; Miller, P. Recurrent Convolutional Network for Video-based Person Re-Identification. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016.
6. Zhen, Z.; Yan, H.; Wei, W.; Liang, W.; Tan, T. See the Forest for the Trees: Joint Spatial and Temporal Recurrent Neural Networks for Video-Based Person Re-identification. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
7. Xu, S.; Cheng, Y.; Gu, K.; Yang, Y.; Chang, S.; Zhou, P. Jointly Attentive Spatial-Temporal Pooling Networks for Video-Based Person Re-identification. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 4743–4752.
8. Chung, D.; Tahboub, K.; Delp, E.J. A Two Stream Siamese Convolutional Neural Network for Person Re-identification. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
9. Strnadl, C.F. Dense 3D-convolutional neural network for person re-identification in videos. *Comput. Rev.* **2019**, *60*, 298.
10. Li, J.; Zhang, S.; Huang, T. Multi-Scale 3D Convolution Network for Video Based Person Re-Identification. *Proc. AAAI Conf. Artif. Intell.* **2019**, *33*, 8618–8625. [CrossRef]
11. Yan, Y.; Ni, B.; Song, Z.; Ma, C.; Yan, Y.; Yang, X. Person Re-Identification via Recurrent Feature Aggregation. In Proceedings of the European Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
12. Liu, H.; Jie, Z.; Jayashree, K.; Qi, M.; Jiang, J.; Yan, S.; Feng, J. Video-based Person Re-identification with Accumulative Motion Context. *IEEE Trans. Circuits Syst. Video Technol.* **2017**, 1. [CrossRef]
13. Li, S.; Bak, S.; Carr, P.; Wang, X. Diversity Regularized Spatiotemporal Attention for Video-Based Person Re-identification. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 369–378.
14. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 7794–7803.
15. Hermans, A.; Beyer, L.; Leibe, B. In Defense of the Triplet Loss for Person Re-Identification. *arXiv preprint* **2017**, arXiv:1703.07737. Available online: http://arxiv.org/abs/1703.07737 (accessed on 21 November 2017).
16. Li, W.; Zhu, X.; Gong, S. Harmonious Attention Network for Person Re-Identification. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
17. Sun, Y.; Zheng, L.; Deng, W.; Wang, S. SVDNet for Pedestrian Retrieval. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
18. Xiao, T.; Li, H.; Ouyang, W.; Wang, X. Learning Deep Feature Representations with Domain Guided Dropout for Person Re-identification. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 26 June–1 July 2016.
19. Zheng, L.; Zhang, H.; Sun, S.; Chandraker, M.; Tian, Q. Person Re-identification in the Wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 384–393.
20. Dong, S.C.; Cristani, M.; Stoppa, M.; Bazzani, L.; Murino, V. Custom Pictorial Structures for Re-identification. *Br. Mach. Vis. Conf. (BMVC)* **2011**, *1*, 6.
21. Gray, D.; Hai, T. Viewpoint Invariant Pedestrian Recognition with an Ensemble of Localized Features. In Proceedings of the Computer Vision—ECCV 2008, 10th European Conference on Computer Vision, Marseille, France, 12–18 October 2008.
22. Liao, S.; Hu, Y.; Zhu, X.; Li, S.Z. Person Re-identification by Local Maximal Occurrence Representation and Metric Learning. *arXiv* **2015**, arXiv:1406.4216. Available online: http://arxiv.org/abs/1406.4216 (accessed on 6 May 2015).
23. Zheng, L.; Yang, Y.; Hauptmann, A.G. Person Re-identification: Past, Present and Future. *arXiv preprint* **2016**, arXiv:1610.02984. Available online: http://arxiv.org/abs/1610.02984 (accessed on 10 October 2016).

24. Karaman, S.; Bagdanov, A.D. Identity Inference: Generalizing Person Re-identification Scenarios. In Proceedings of the International Conference on Computer Vision-volume Part I, Florence, Italy, 7–13 October 2012.
25. Simonnet, D.; Lewandowski, M.; Velastin, S.A.; Orwell, J.; Turkbeyler, E. Re-identification of Pedestrians in Crowds Using Dynamic Time Warping. In Proceedings of the International Conference on Computer Vision-volume Part I, Florence, Italy, 7–13 October 2012.
26. He, K.; Zhang, X.; Ren, S. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
27. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-Excitation Networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
28. Gao, J.; Nevatia, R. Revisiting Temporal Modeling for Video-based Person ReID. *arXiv* **2018**, arXiv:1805.02104.
29. Arulkumar, S.; Athira, N. Anurag Mittal Co-Segmentation Inspired Attention Networks for Video-Based Person Re-Identification. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27–28 October 2019.
30. Zheng, L.; Bie, Z.; Sun, Y.; Wang, J.; Su, C.; Wang, S.; Tian, Q. MARS: A Video Benchmark for Large-Scale Person Re-Identification. In Proceedings of the European Conference on Computer Vision, Durham, NC, USA, 11–14 October 2016; pp. 868–884.
31. Wu, Y.; Lin, Y.; Dong, X.; Yan, Y.; Yang, Y. Exploit the Unknown Gradually: One-Shot Video-Based Person Re-identification by Stepwise Learning. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
32. Ristani, E.; Solera, F.; Zou, R.S. Performance Measures and a Data Set for Multi-Target, Multi-Camera Tracking. In Proceedings of the European Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 17–35.
33. Suh, Y.; Wang, J.; Tang, S.; Mei, T.; Lee, K.M. Part-Aligned Bilinear Representations for Person Re-identification. In Proceedings of the European Conference on Computer Vision (ECCV), Venice, Italy, 22–29 October 2017; pp. 402–419.
34. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in pytorch. In Proceedings of the 31st Conference on Neural Information Processing System, Long Beach, CA, USA, 28 October 2017.
35. Liu, Y.; Yan, J.; Ouyang, W. Quality Aware Network for Set to Set Recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 26 June–1 July 2016.
36. Li, D.; Chen, X.; Zhang, Z.; Huang, K. Learning Deep Context-aware Features over Body and Latent Parts for Person Re-identification. In Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 384–393.
37. Song, G.; Leng, B.; Liu, Y.; Hetang, C.; Cai, S. Region-based Quality Estimation Network for Large-scale Person Re-identification. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
38. Chen, D.; Li, H.; Tong, X.; Shuai, Y.; Wang, X. Video Person Re-identification with Competitive Snippet-Similarity Aggregation and Co-attentive Snippet Embedding. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.