1. Further implementation details

1.1. Details of training of 3D CNN

For the training of I3D network, we use the AMSGrad optimizer \(^{[1]}\) with \(\beta_1 = 0.9, \beta_2 = 0.999\). We use a weight decay of \(5.0 \times 10^{-4}\). In each training iteration, we use a batch of 32 clips belonging to 8 person identities with 4 instances of each identity i.e. \(P = 8\) and \(K = 4\). The RGB input values are scaled and shifted to be in the range \([-1.0, 1.0]\). For data augmentation, each input clip is first resized up to \(144 \times 288 (H \times W)\) and then a random crop of size \(128 \times 256\) is taken. Input clips are also randomly flipped horizontally with a probability of 0.5. For training on MARS dataset, we train the network for 1200 epochs with an initial learning rate of \(3.0 \times 10^{-4}\). We reduce the learning rate by a factor of 10 after every 400 epochs. The margin \(m\) in the triplet loss expression is set to 0.3.

1.2. Details of training of Clip-Similarity Aggregation Module

For the training of Clip-Similarity Aggregation module, we again use the AMSGrad optimizer with \(\beta_1 = 0.9, \beta_2 = 0.999\) and a weight decay of \(5.0 \times 10^{-4}\). We use a batch size of 48 with \(P = 12\) and \(K = 4\). We use the same input transformations and data augmentation techniques as described for the training of the I3D network. We train the aggregation module for 12 epochs with an initial learning rate of \(3.0 \times 10^{-5}\). We reduce the learning rate by a factor of 10 after 8 epochs. We set margin \(m = 1\) in the triplet loss.

2. Further experiments

2.1. Ablation experiment for choice of \(M_{\text{test}}\) and \(L_{\text{test}}\)

Tab. \(\dagger\) shows the re-identification performance (mAP) with averaging I3D features of multiple clips as we vary the number of clips \((M)\) and the clip-length \((L)\). We can observe that while \(L = 16\) has a better performance than \(L = 4, 8\) when using a single clip \(M = 1\), it performs lower when number of clips averaged is larger. For \(M = 8, L = 4\) and \(L = 8\) have similar performances, i.e. 74.9 vs. 75.0. Considering the higher computational cost with \(L = 8\), we have used \(L = 4\), with higher \(M\), for the experiments in the paper.

References