Appendix of the Paper
(Unsupervised Cross-dataset Person Re-identification by Transfer Learning of Spatial-Temporal Patterns)

Jianming Lv[1], Weihang Chen[1], Qing Li[2], and Can Yang[1]

[1]South China University of Technology
[2]City University of Hongkongy
{jmlv,cscyang}@scut.edu.cn,csscut@mail.scut.edu.cn,qing.li@cityu.edu.hk

1. Architecture of the Visual Classifier $\mathcal{C}$

(Extension of Section 4.2)

We select the recently proposed convolutional siamese network [4] as $\mathcal{C}$, which makes better use of the label information and has good performance in the large-scale datasets such as Market1501[3]. As shown in Fig. 1, the network adopts a siamese scheme including two ImageNet pre-trained CNN modules, which share the same weight parameters and extract visual features from the input images S_i and S_j. The CNN module is achieved from the ResNet-50 network [1] by removing its final fully-connected (FC) layer. The outputs of the two CNN modules are flattened into two one-dimensional vectors: $\vec{v_i}$ and $\vec{v_j}$, which act as the embedding visual feature vectors of the input images.

To measure the matching degree of the input images, their feature vectors $\vec{v_i}$ and $\vec{v_j}$ are fed into the following square layer to conduct subtracting and squaring element-wisely: $\vec{v_s} = (\vec{v_i} - \vec{v_j})^2$. Finally, a convolutional layer is used to transform $\vec{v_s}$ into the similarity score as:

$$\hat{q} = sigmoid(\theta_s \circ \vec{v_s}) \tag{1}$$

.Here θ_s denotes the parameters in the convolutional layer, $\circ$ denotes the convolutional operation, and $sigmoid$ indicates the $sigmoid$ activation function. By comparing the predicted similarity score with the ground-truth matching result of S_i and S_j, we can achieve the **variation loss** as a cross entropy form:

$$LOSS_v = -q \cdot log(\hat{q}) - (1-q) \cdot log(1-\hat{q}) \tag{2}$$

.Here $q = 1$ when S_i and S_j contain the same person. otherwise, $q = 0$.

Besides predicting the similarity score, the model also predicts the identity of each image in the following steps. Each visual feature vector ($\vec{v_x}(x = i, j)$) is fed into one convolutional layer to be mapped into an one-dimensional vector with the size K, where K is equal to the total number of the pedestrians in the dataset. Then the following softmax unit is applied to normalize the output as follows:

$$\hat{P}^{(x)} = softmax(\theta_x \circ \vec{v_x})(x = i, j) \tag{3}$$

Here θ_x is the parameter in the convolutional layer and $\circ$ denotes the convolutional operation. The output $\hat{P}^{(x)}$ is used to predict the identity of the person contained in the input image $S_x(x = i, j)$. By comparing $\hat{P}^{(x)}$ with the ground-truth identify label, we can achieve the **identification loss** as the cross-entropy form:

$$LOSS_{id} = \sum_{k=1}^{K}(-log\hat{P}_k^{(i)} \cdot P_k^{(i)}) + \sum_{k=1}^{K}(-log\hat{P}_k^{(j)} \cdot P_k^{(j)}) \tag{4}$$

Here $P^{(x)}(x = i, j)$ is the identity vector of the input image S_x. $P_k^{(x)} = 0$ for all k except $P_t^{(x)} = 1$, where t is ID of the person in the image S_x.

The final loss function of the model is defined as:

$$LOSS_{all} = LOSS_v + LOSS_{id} \tag{5}$$

According to [2], this kind of composite loss makes the classifier more efficient to extract the view invariant visual features for Re-ID than the single loss function.

While deploying this classifier to perform Re-ID, given two images S_i and S_j as input, the CNN modules extract their visual feature vectors $\vec{v_i}$ and $\vec{v_j}$ as shown in Fig. 1. The matching probability of S_i and S_j is measured as the cosine similarity of the two feature vectors:

$$Pr(S_i \Vdash_{\mathcal{C}} S_j | \vec{v_i}, \vec{v_j}) = \frac{\vec{v_i} \cdot \vec{v_j}}{\| \vec{v_i} \|_2 \| \vec{v_j} \|_2} \tag{6}$$

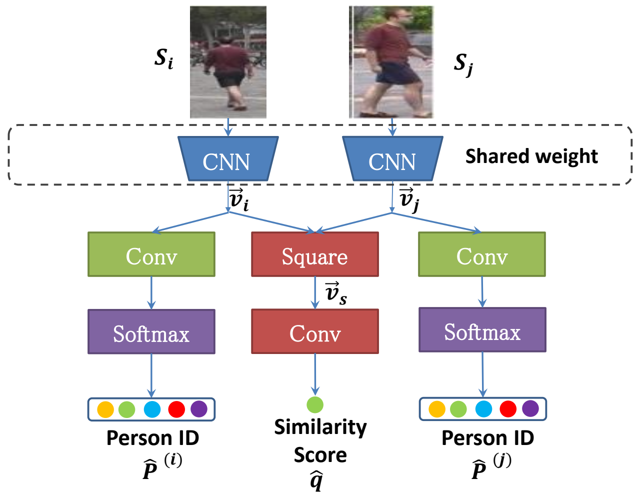


Figure 1: Visual classifier based on CNN.

2. Proof of Eq. (5)

$$\begin{aligned}
& Pr(\triangle_{ij}, c_i, c_j | S_i \Vdash S_j) \\
= \ & Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) = \Upsilon(S_j)) * \\
& Pr(\Upsilon(S_i) = \Upsilon(S_j) | S_i \Vdash S_j) + \\
& Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) \neq \Upsilon(S_j)) * \\
& Pr(\Upsilon(S_i) \neq \Upsilon(S_j) | S_i \Vdash S_j) \\
= \ & Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) = \Upsilon(S_j)) * (1 - E_p) + \\
& Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) \neq \Upsilon(S_j)) * E_p
\end{aligned} \tag{7}$$

Similarly, we have:

$$\begin{aligned}
& Pr(\triangle_{ij}, c_i, c_j | S_i \nVdash S_j) \\
= \ & Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) = \Upsilon(S_j)) * \\
& Pr(\Upsilon(S_i) = \Upsilon(S_j) | S_i \nVdash S_j) + \\
& Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) \neq \Upsilon(S_j)) * \\
& Pr(\Upsilon(S_i) \neq \Upsilon(S_j) | S_i \nVdash S_j) \\
= \ & Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) = \Upsilon(S_j)) * E_n + \\
& Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) \neq \Upsilon(S_j)) * (1 - E_n)
\end{aligned} \tag{8}$$

From (7) and (8)), we have:

$$\begin{aligned}
& Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) = \Upsilon(S_j)) \\
= & (1 - E_n - E_p)^{-1}((1 - E_n) * Pr(\triangle_{ij}, c_i, c_j | S_i \Vdash_{\mathcal{C}} S_j) \\
& - E_p * Pr(\triangle_{ij}, c_i, c_j | S_i \nVdash_{\mathcal{C}} S_j))
\end{aligned} \tag{9}$$

□

3. Proof of Theorem 1

Proof of Theorem 1: By analyzing the relationship between $Pr(\Upsilon(S_i) = \Upsilon(S_j) | v_i, v_j, \triangle_{ij}, c_i, c_j)$ and $Pr(S_i \Vdash_{\mathcal{F}} S_j | v_i, v_j, \triangle_{ij}, c_i, c_j)$, we have:

$$\begin{aligned}
& Pr(\Upsilon(S_i) = \Upsilon(S_j) | v_i, v_j, \triangle_{ij}, c_i, c_j) \\
= & Pr(\Upsilon(S_i) = \Upsilon(S_j) | S_i \Vdash_{\mathcal{F}} S_j) * Pr(S_i \Vdash_{\mathcal{F}} S_j | v_i, v_j, \triangle_{ij}, c_i, c_j) + \\
& Pr(\Upsilon(S_i) = \Upsilon(S_j) | S_i \nVdash_{\mathcal{F}} S_j) * Pr(S_i \nVdash_{\mathcal{F}} S_j | v_i, v_j, \triangle_{ij}, c_i, c_j) \\
= & (1 - E'_p) * Pr(S_i \Vdash_{\mathcal{F}} S_j | v_i, v_j, \triangle_{ij}, c_i, c_j) \\
& + E'_n * (1 - Pr(S_i \Vdash_{\mathcal{F}} S_j | v_i, v_j, \triangle_{ij}, c_i, c_j)) \\
= & (1 - E'_p - E'_n) * Pr(S_i \Vdash_{\mathcal{F}} S_j | v_i, v_j, \triangle_{ij}, c_i, c_j) + E'_n
\end{aligned} \tag{10}$$

According to the Eq.(11) of the original paper, we have:

$$\begin{aligned}
& Pr(S_i \Vdash_{\mathcal{F}} S_j | v_i, v_j, \triangle_{ij}, c_i, c_j) \\
= & \frac{(M_1 + \frac{\alpha}{1-\alpha-\beta})((1-\alpha) * M_2 - \beta * M_3)}{Pr(\triangle_{ij}, c_i, c_j)} (0 \leq \alpha, \beta \leq 1)
\end{aligned} \tag{11}$$

By substituting Eq.(11) into Eq.(10), we have:

$$\begin{aligned}
& Pr(\Upsilon(S_i) = \Upsilon(S_j) | v_i, v_j, \triangle_{ij}, c_i, c_j) \\
= & (1 - E'_p - E'_n) * \frac{(M_1 + \alpha(1 - \alpha - \beta)^{-1})}{Pr(\triangle_{ij}, c_i, c_j)} \\
& * ((1-\alpha)M_2 - \beta M_3) + E'_n
\end{aligned} \tag{12}$$

On the other hand, from the Eq.(9) of the original paper, we have:

$$\begin{aligned}
& Pr(\Upsilon(S_i) = \Upsilon(S_j) | \vec{v_i}, \vec{v_j}, \triangle_{ij}, c_i, c_j) \\
= & \frac{(M_1 + \frac{E_n}{1 - E_n - E_p})((1 - E_n)M_2 - E_p M_3)}{Pr(\triangle_{ij}, c_i, c_j)}
\end{aligned} \tag{13}$$

From (13) and (12) we have:

$$\begin{aligned}
& (M_1 + E_n(1 - E_p - E_n)^{-1}) * ((1 - E_n) * M_2 - E_p * M_3) \\
= \ & (1 - E'_p - E'_n) * (M_1 + \alpha(1 - \alpha - \beta)^{-1}) \\
& *((1-\alpha)M_2 - \beta M_3) + E'_n * Pr(\triangle_{ij}, c_i, c_j)
\end{aligned} \tag{14}$$

Thus, we have:

$$\begin{aligned}
& \sum_{\triangle_{ij}, c_i, c_j} [(M_1 + E_n(1 - E_p - E_n)^{-1}) \\
& * ((1 - E_n) * M_2 - E_p * M_3)] \\
= & \sum_{\triangle_{ij}, c_i, c_j} [((1 - E'_p - E'_n) * (M_1 + \alpha(1 - \alpha - \beta)^{-1}) \\
& * ((1-\alpha)M_2 - \beta M_3) + E'_n * Pr(\triangle_{ij}, c_i, c_j))]
\end{aligned} \tag{15}$$

From (15) have:

$$\begin{aligned}
& (M_1 + E_n(1 - E_p - E_n)^{-1})(1 - E_n - E_p) \\
= \ & (1 - E'_p - E'_n)((1 - \alpha - \beta)M_1 + \alpha)p + E'_n
\end{aligned} \tag{16}$$

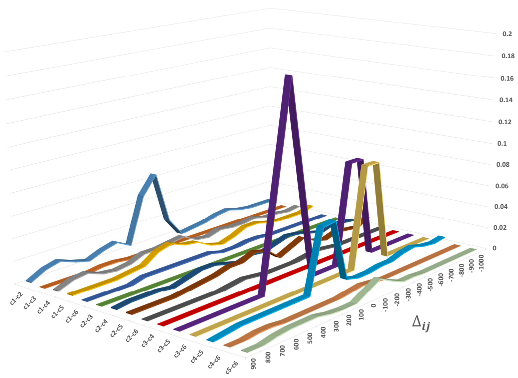
After taking the derivative with respect to M_1 in the both sides of Eq. (16), we can get:

$$1 - E_p - E_n = (1 - \alpha - \beta)(1 - E'_p - E'_n) \tag{17}$$

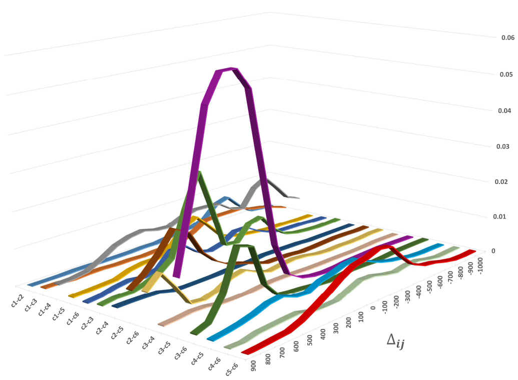
Thus, when $E_p + E_n < 1$ and $\alpha + \beta < 1$, we can infer from Eq.(17) that:

$$E'_p + E'_n < E_p + E_n. \tag{18}$$

□



(a)



(b)

Figure 2: The spatio-temporal distribution $Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) \Vdash_{\mathcal{C}} \Upsilon(S_j))$ learned (a) in the 'GRID' dataset, and (b) in the 'Market1501' dataset.

4. Learned Spatio-temporal Patterns

(Extension of Fig.4)

Fig. 2 shows the spatio-temporal distribution $Pr(\triangle_{ij}, c_i, c_j | \Upsilon(S_i) \Vdash_{\mathcal{C}} \Upsilon(S_j))$ learned in the 'GRID' and 'Market1501' dataset.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015. 1

[2] L. Wei, Z. Xiatian, and G. Shaogang. Person re-identification by deep joint learning of multi-loss classification. In *CVPR*, 2017. 1

[3] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *Computer Vision, IEEE International Conference on*, 2015. 1

[4] Z. Zheng, L. Zheng, and Y. Yang. A discriminatively learned cnn embedding for person re-identification. *TOMM*, 2017. 1