i-VisionGroup@Tsinghua

Localized Multi-kernel Discriminative Canonical Correlation Analysis for Video-based Person Re-Identification

Guangyi Chen, Jiwen Lu, Jianjiang Feng, Jie Zhou

Tsinghua University, China

Personal Introduction

Guangyi Chen

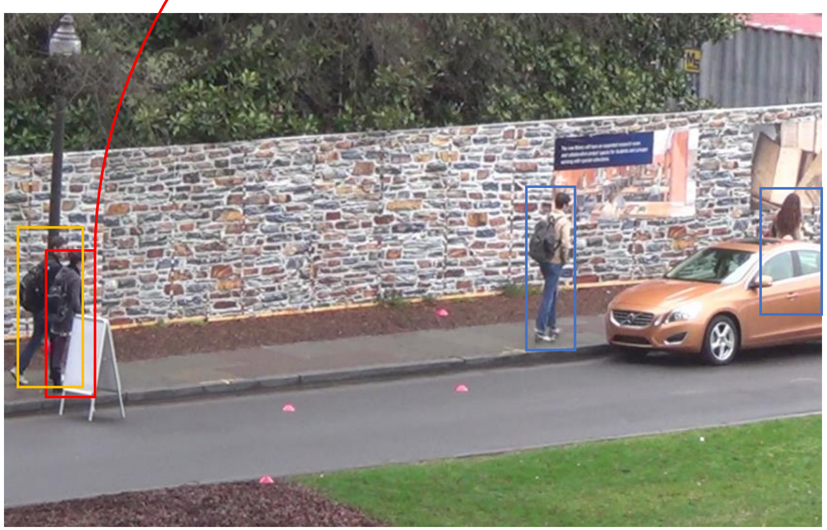
- ✓ Pattern Recognition and Intelligent Systems, Department of Automation, Tsinghua University
- ✓ 1^{st} Year of Ph.D. Candidate
- ✓ Supervised by Professor *Jie Zhou* and Associate Professor *Jiwen Lu*
- ✓ Research Interests: Person Re-identification, Metric Learning.

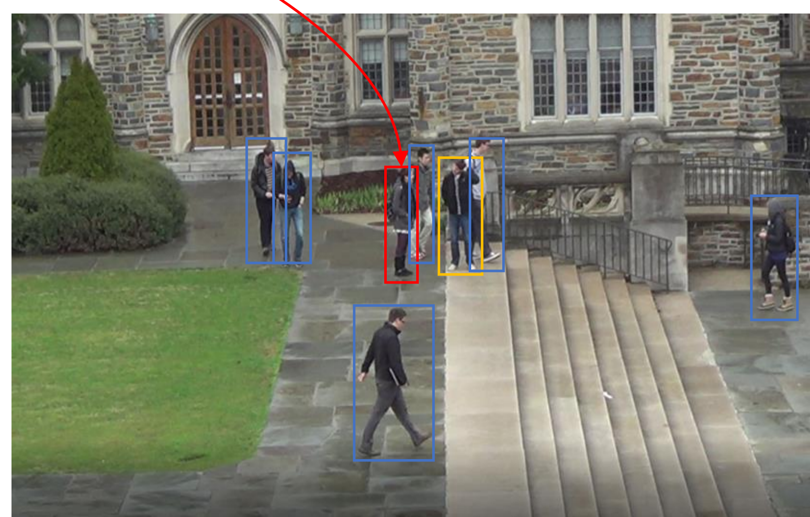
Person Re-Identification

Person re-identification (person re-id) aims to matching the same individuals across multi-cameras without overlapping.

Probe person

Match it in the gallery





Challenge

- Low resolution
- Occlusion
- Background
- Illumination
- View point
- Pose

Video Based VS Image Based

Image

Video

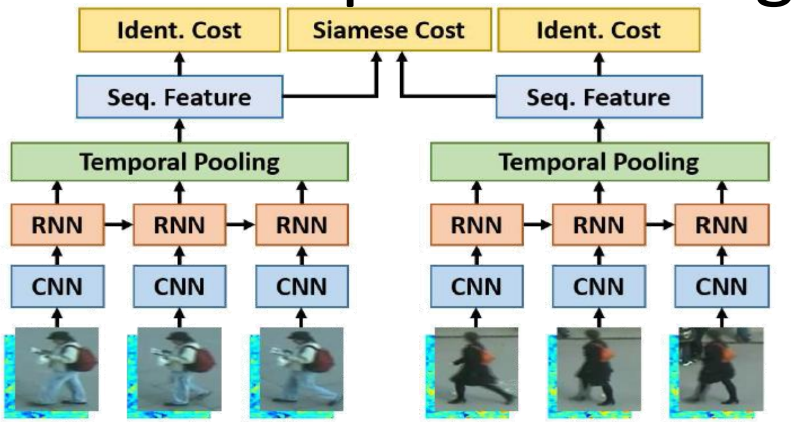
- Temporal Information
- Complementary cues
- Eliminate noise

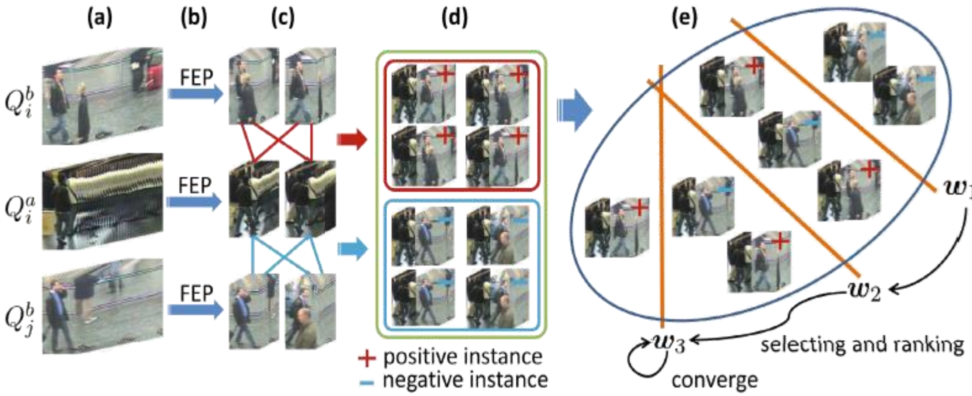
Pedestrian Video Representation

Temporal Pooling:
Recurrent Convolutional Network for Video-based Person Re-Identification[1]

Segment Selecting:
Person Re-identification by Video Ranking[2]

Temporal Pooling & Segment Selecting

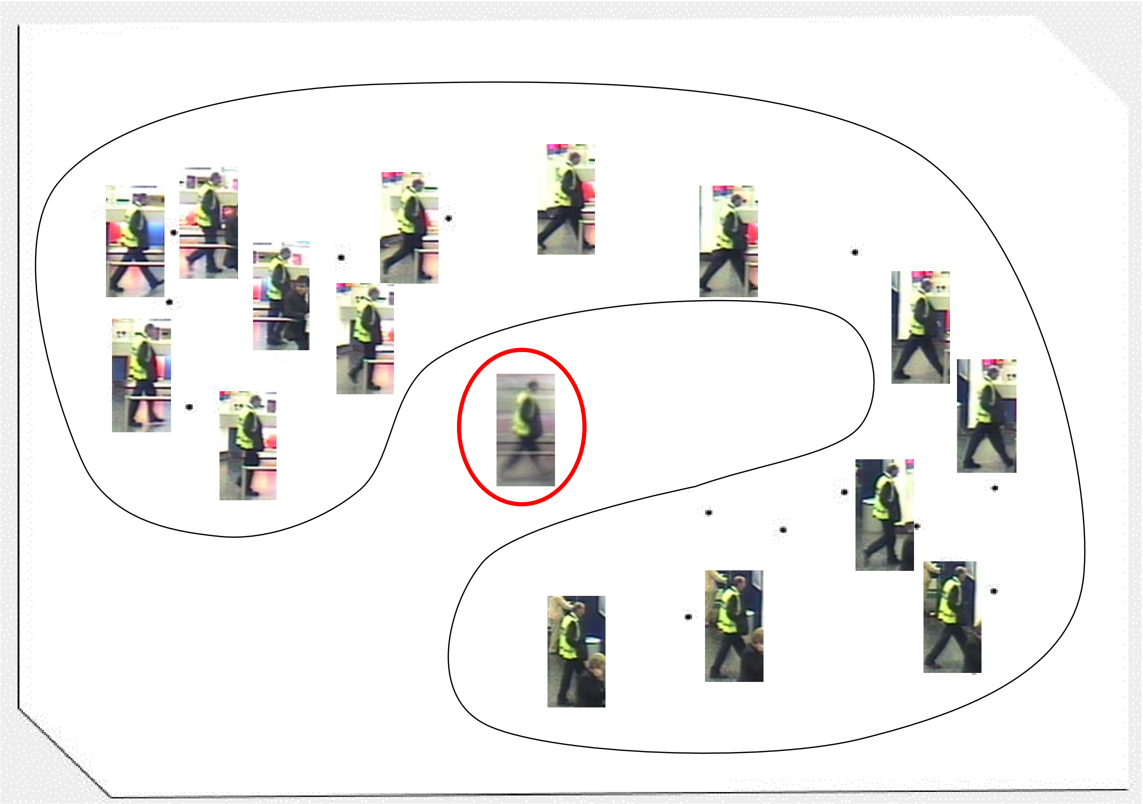




[1]McLaughlin N et.al, Recurrent convolutional network for video-based person re-identification. CVPR, 2016
[2] Wang T, et al. Person re-identification by video ranking. ECCV, 2014

Person Video Manifold

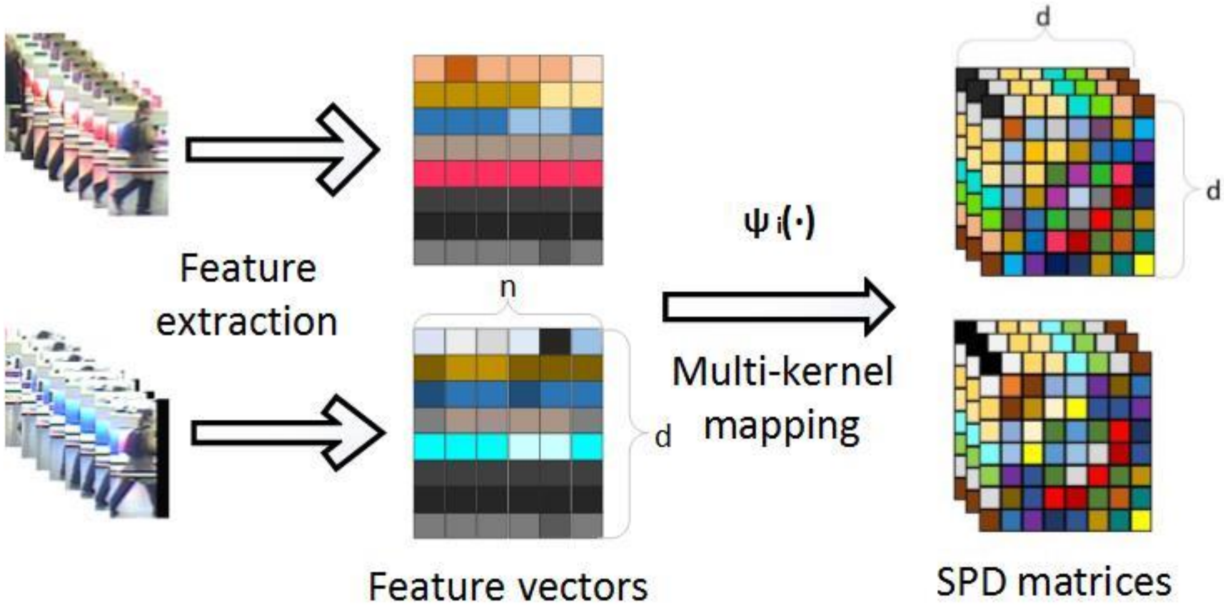
Person video lies on a manifold which can't be represented by an average pooling.



Video Modeling by SPD Matrix

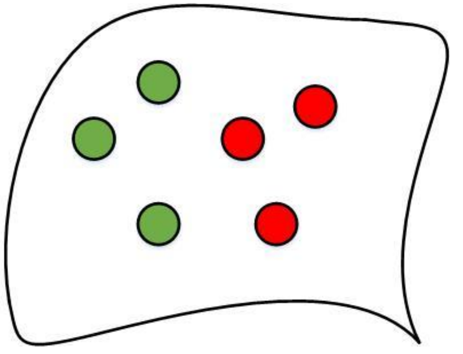
By calculating kernel matrices with features of frames, we represent video as a SPD matrix.

$$S(i,j) = < \phi(f_i), \phi(f_j) > = \kappa(f_i, f_j)$$



Metric of SPDs

How to calculate the distance between two SPD matrices?



Affine-invariant distance $$d_{AID}(S_1, S_2) = \sqrt{\sum_{i=1}^{d} \ln^2 \lambda_i(S_1, S_2)}$$

Log-Euclidean distance $$d_{LED}(S_1, S_2) = \|\log(S_1) - \log(S_2)\|_F$$

Riemannian kernel function $$\kappa_{LOG}(S_1, S_2) = \mathrm{tr}[\log(S_1) \cdot \log(S_2)]$$

Select Appreciate Kernels

Kernel is sensitive to parameters and types

$$Appreciate\ kernel = professional\ knowledge + experiance + parameter\ adjusting$$

Multi-kernel learning

$$\kappa_{RBF}(f_i, f_j) = \exp(-\gamma \| f_i - f_j \|_p^2)$$

$$\kappa_{COV}(f_i, f_j) = \left\langle \frac{\bar{f_i}}{\sqrt{n-1}}, \frac{\bar{f_j}}{\sqrt{n-1}} \right\rangle$$

$$\kappa = \sum_{m=1}^{M} \eta_m \kappa_m$$

$$\kappa_{BHA}(f_i, f_j) = \sqrt{\frac{2\sigma_i \sigma_j}{\sigma_i^2 + \sigma_j^2}}\, exp\left(-\frac{1}{4}\frac{(\mu_i - \mu_j)^2}{\sigma_i^2 + \sigma_j^2}\right)$$

Localized Multi-Kernel CCA

Our framework is an optimization problem as follows:

$$min \sum_{i,j} w_{ij} \left\| f_x(X_i) - g_y(Y_j) \right\|_F^2$$

$$s.t. \sum_i \|f_x(X_i)\|_F^2 = 1 \ , \sum_j \left\| g_y(Y_j) \right\|_F^2 = 1$$

By the Riemannian kernels, we project the SPD from Riemannian manifold to Euclidean space

$$min \sum_{i,j} w_{ij} \left\| \alpha^T K_x^{(i)} - \beta^T K_y^{(j)} \right\|_F^2$$

$$s.t. \ \alpha^T \alpha = I, \beta^T \beta = I$$

Localized Multi-Kernel CCA

With the multi-kernel learning algorithm, we combine multi-SPDs induced by multi-kernels

$$\kappa = \sum_{m=1}^{M} \eta_m \kappa_m$$

All samples share same weights

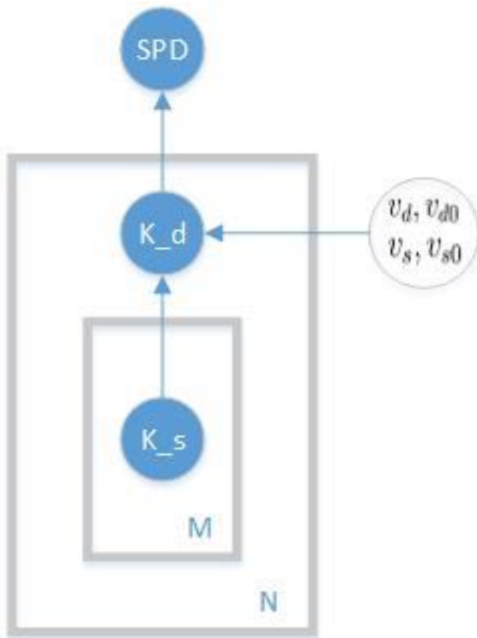
We learn the localized weights with a softmax function

$$\eta_m\left(K_m^{(i)}\right) = \frac{\exp(v_m^T K_m^{(i)} + v_{m0})}{\sum_{m=1}^{M} \exp(v_m^T K_m^{(i)} + v_{m0})}$$

$$K(i,j) = \sum_{m=1}^{M} \eta_m\left(K_m^{(i)}\right) K_m(i,j) \eta_m(K_m^{(j)})$$

Two Layer Localized Multi-Kernel CCA

In addition, we design a two layer framework to learn representation kernel and Riemannian metric kernel simultaneously.



Extend to multiple cameras

We extend our method to multiple cameras by learning view-aware metric.

$$min \sum_{i,j,c_1,c_2(c_1 \neq c_2)} w_{ij} \left\| W_{c_1}^T K_{c_1}^{(i)} - W_{c_2}^T K_{c_2}^{(j)} \right\|_F^2$$

$s.t.\ W_c^T W_c = I$, c=1,2,…,C

Datasets

We evaluate our method on three open dataset.

Datasets	identities	cameras	images	setting	partition
PRID 2011	178	2	40033	Random partition	89 for train 89 for test
iLIDS-VID	300	2	42495	Random partition	150 for train 150 for test
MARS	1261	6	1191003	Fixed partition	625 for train 634 for test

Evaluation on ILIDS-VID

Method	Rank=1	Rank=5	Rank=10	Rank=20
DVDL	25.9	48.2	57.3	68.9
SDALF+DVR	41.3	63.5	72.7	83.1
TDL	56.7	80.0	87.6	93.6
McLaughlin	58.0	84.0	91.0	96.0
STFV3D+KISSME	44.3	71.7	83.7	91.7
DCCA(mean)	60.3	80.6	87.3	90.9
GMKDCCA	70.6	90.1	93.8	97.3
LMKDCCA	**73.3**	**90.5**	**94.7**	**98.1**

Evaluation on PRID 2011

Method	Rank=1	Rank=5	Rank=10	Rank=20
DVDL	40.6	69.7	77.8	85.6
SDALF+DVR	48.3	74.9	87.3	94.4
TDL	56.3	87.6	95.6	98.3
McLaughlin	70.0	90.0	95.0	97.0
STFV3D+KISSME	64.1	87.3	89.9	92.0
DCCA(mean)	76.7	92.8	95.9	98.0
GMKDCCA	83.0	96.1	99.4	99.8
LMKDCCA	**86.4**	**97.5**	**99.6**	**100**

Evaluation on MARS

Method	Rank=1	Rank=5	Rank=20	Map
IDE + Kissme	65.0	81.1	88.9	45.6
IDE + XQDA	65.3	82.0	89.0	47.6
IDE+ LMKDCCA	69.2	84.0	91.2	50.6

Future Works

- Use the deep network to mine relationship of kernels
- Exploit the temporal information more effectively

Thanks!