Recurrent Neural Networks for Person Re-identification Revisited

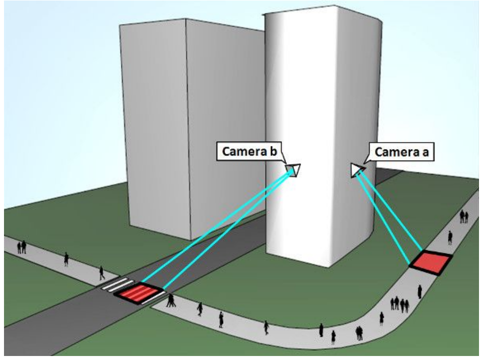
Jean-Baptiste Boin
Stanford University
jbboin@stanford.edu

André Araujo
Google AI
andrearaujo@google.com

Bernd Girod
Stanford University
bgirod@stanford.edu

Person video re-identification

- Goal: associate person video tracks from different cameras
- Applications:
  - Video surveillance
  - Home automation
  - Crowd dynamics understanding



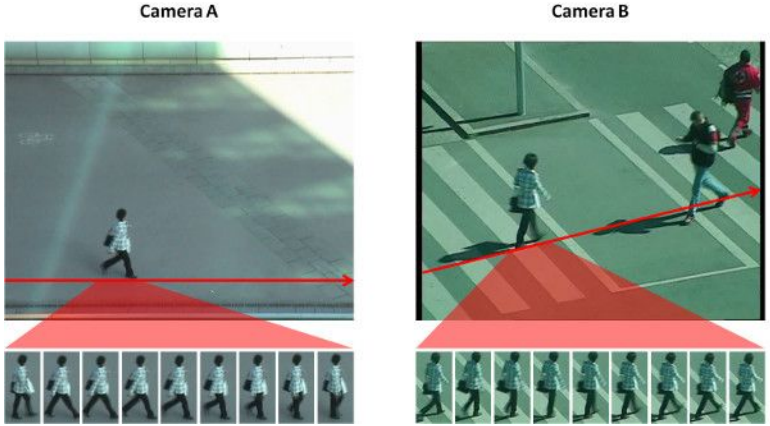


Image credit: PRID2011 dataset [Hirzer *et al.*, 2011]

Person video re-identification: challenges

Lighting variations

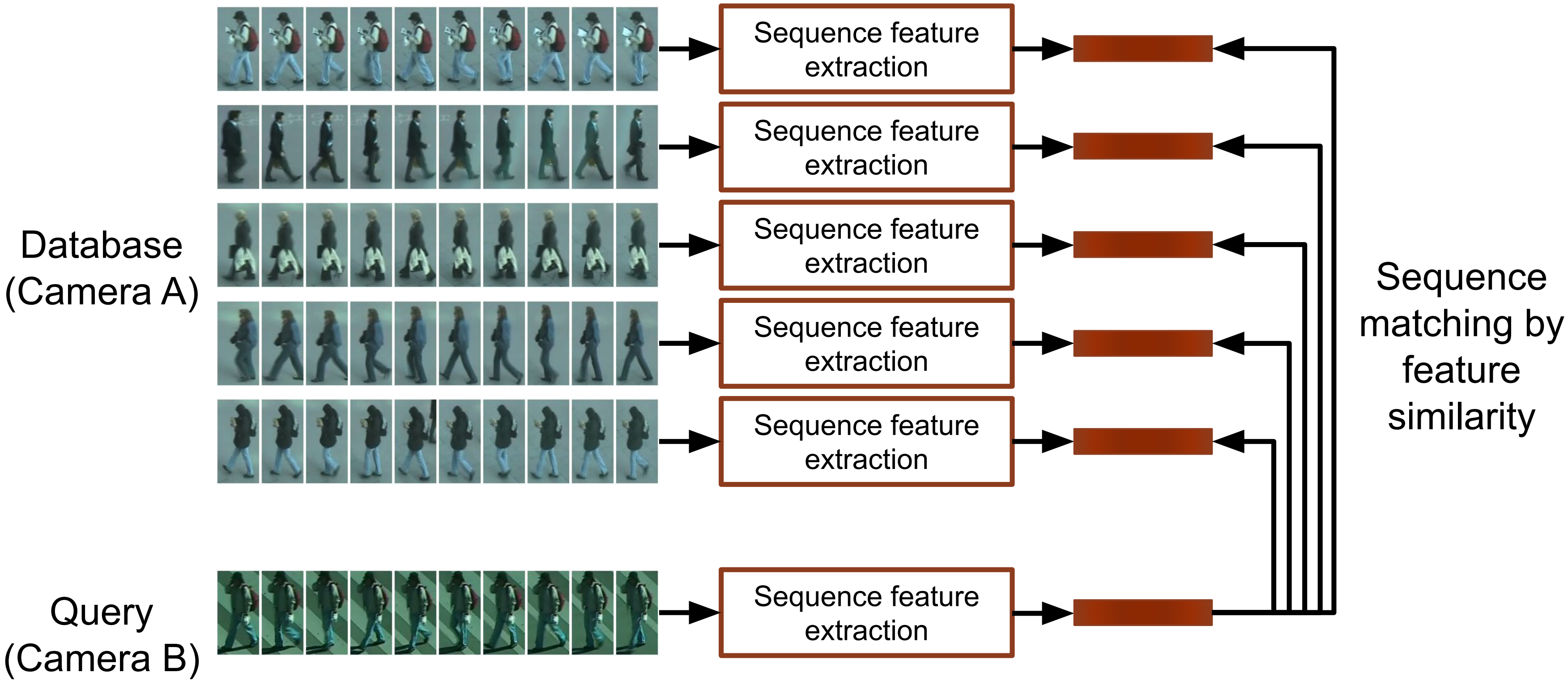
Viewpoint changes

Clothing similarity

Background clutter and occlusions

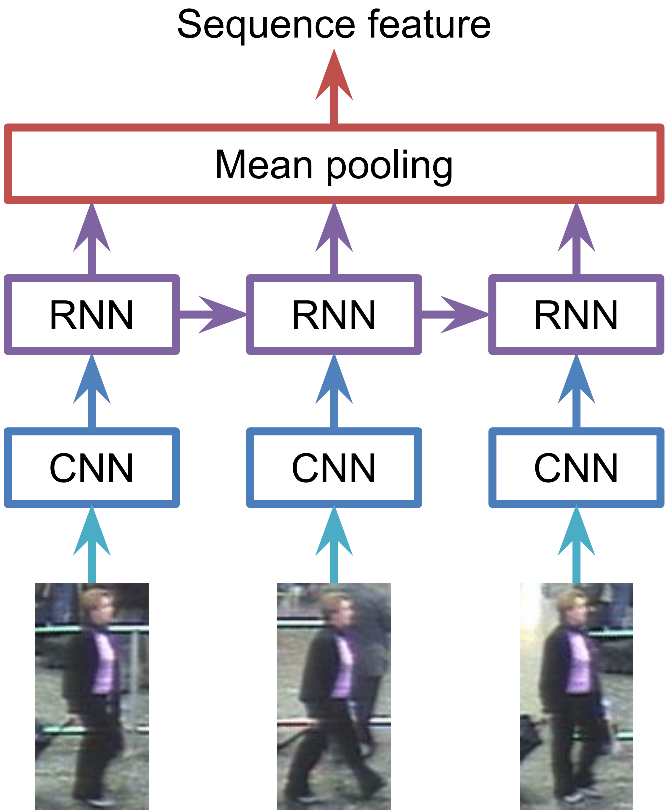
Credit: iLIDS-VID dataset [Wang *et al.*, 2014]

Framework: re-identification by retrieval



Related work

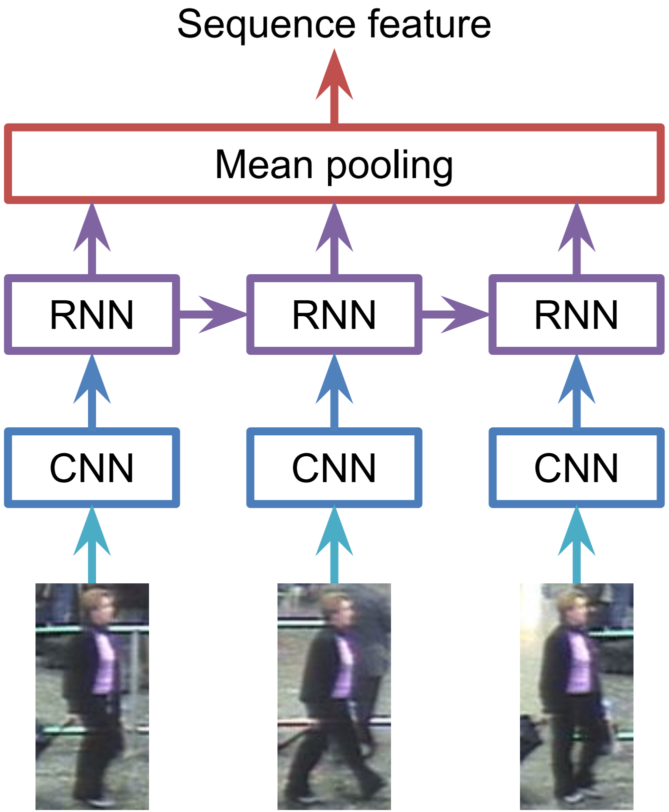
- Most common setup
  - Frame feature extraction: CNN
  - Sequence processing: RNN
  - Temporal pooling: mean pooling
  - *[McLaughlin et al., 2016], [Yan et al., 2016], [Wu et al., 2016]*



Related work

- Most common setup
  - Frame feature extraction: CNN
  - Sequence processing: RNN
  - Temporal pooling: mean pooling
  - *[McLaughlin et al., 2016], [Yan et al., 2016], [Wu et al., 2016]*
- Extensions
  - Bi-directional RNNs *[Zhang et al., 2017]*
  - Multi-scale + attention pooling *[Xu et al., 2017]*
  - Fusion of CNN+RNN features *[Chen et al., 2017]*

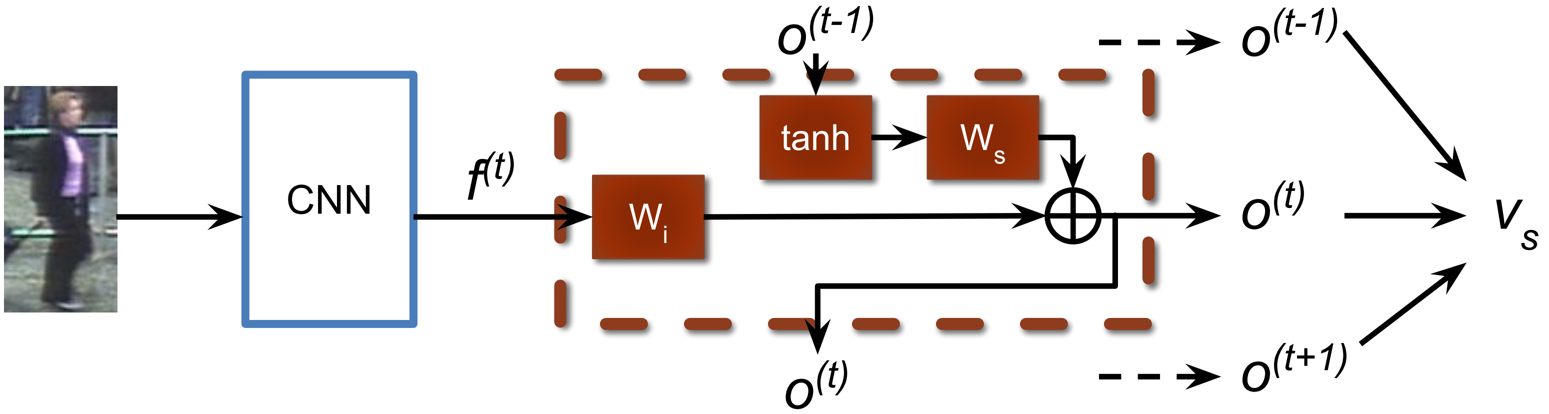
See review paper *[Zheng et al., 2016]*



Outline

- Feed-forward RNN approximation with similar representational power
- New training protocol to leverage multiple video tracks within a mini-batch
- Experimental evaluation
- Conclusions

RNN setup



- $f^{(t)}$: inputs of sequence processing stage (frame descriptors)
- $o^{(t)}$: outputs of sequence processing stage

$$o^{(t)} = W_i f^{(t)} + W_s \tanh\left(o^{(t-1)}\right)$$

- $v_s = \frac{1}{T}\sum_{t=1}^{T} o^{(t)}$: sequence feature (output of temporal pooling stage)

Stanford University

Proposed feed-forward approximation (1/2)

- “Short-term dependency” approximation

  Disregard terms from step *(t-2)* in output from step *(t)*

$$
\begin{aligned}
o^{(t)} &= W_i f^{(t)} + W_s \tanh\left(o^{(t-1)}\right) \\
&\approx W_i f^{(t)} + W_s \tanh\left(W_i f^{(t-1)}\right)
\end{aligned}
$$

Proposed feed-forward approximation (2/2)

- "Long sequence" approximation

$$v_s = \frac{1}{T}\sum_{t=1}^{T} o^{(t)}$$

Using approximation from previous slide

$$\approx \frac{1}{T}\sum_{t=1}^{T}\left(W_i f^{(t)} + W_s \tanh\left(W_i f^{(t-1)}\right)\right)$$

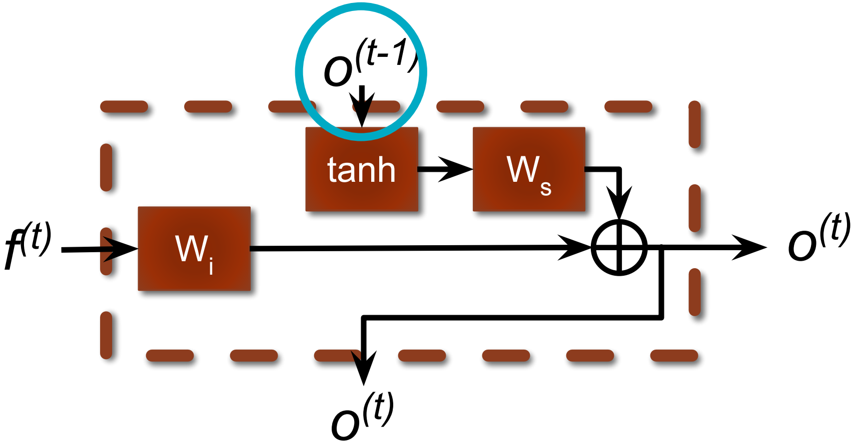
$$= \frac{1}{T}\sum_{t=1}^{T} W_i f^{(t)} + \frac{1}{T}\sum_{t=0}^{T-1} W_s \tanh\left(W_i f^{(t)}\right)$$

Disregard edge cases (first and last frame) since videos are long

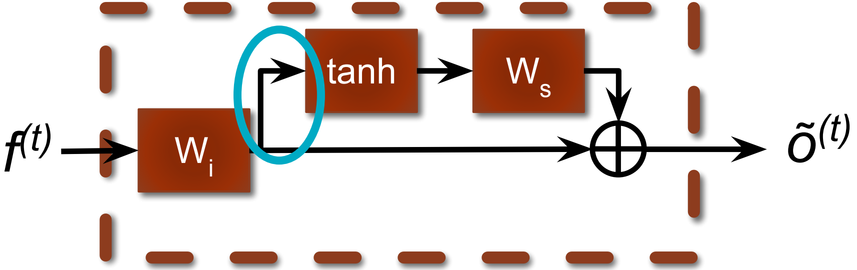
$$\approx \frac{1}{T}\sum_{t=1}^{T}\underbrace{\left(W_i f^{(t)} + W_s \tanh\left(W_i f^{(t)}\right)\right)}_{\tilde{o}^{(t)}}$$

Proposed feed-forward approximation: new block

RNN



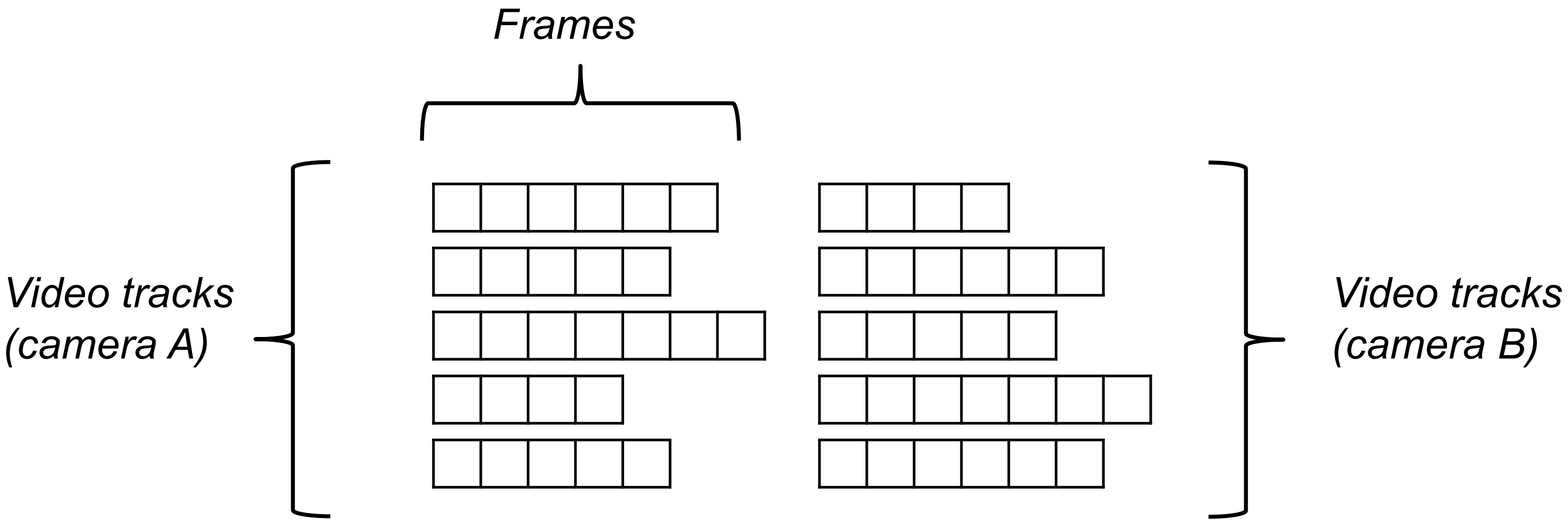
Ours: FNN



- Same memory footprint
- Direct mapping between RNN and FNN parameters

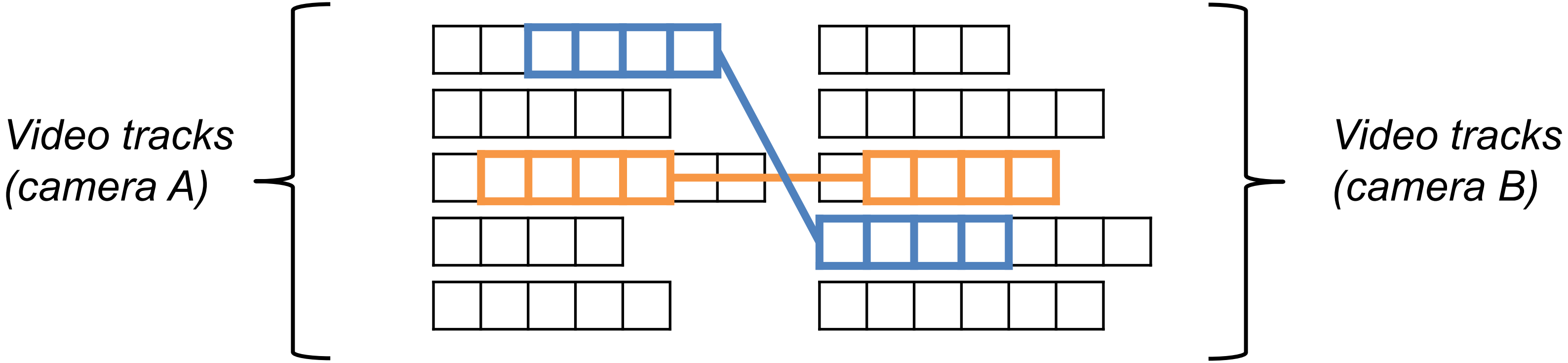
Training pipeline

- Training data



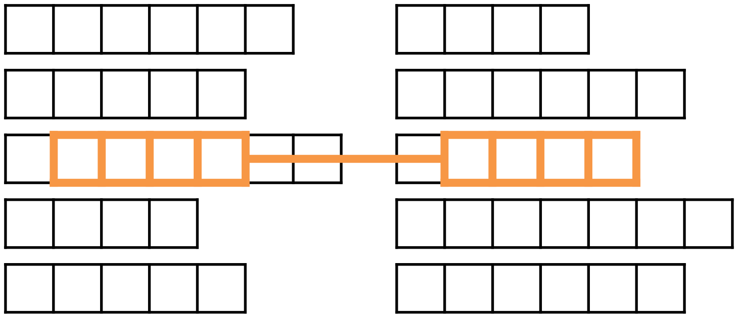
Training pipeline: RNN baseline

- **SEQ**: load sequences of consecutive frames in mini-batch

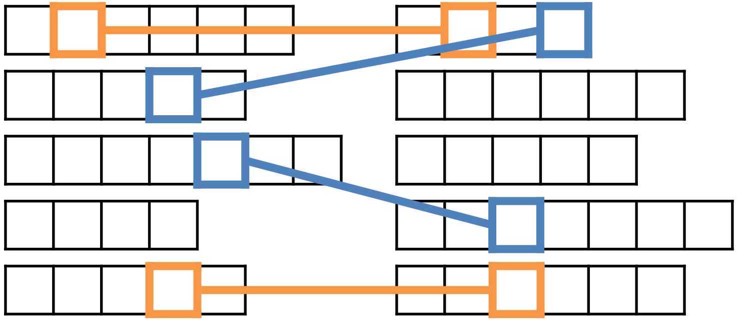


Proposed FNN training pipeline

- **FRM**: load independent frames
- Load images from many more identities in a mini-batch (same memory/computational cost)



SEQ (baseline)



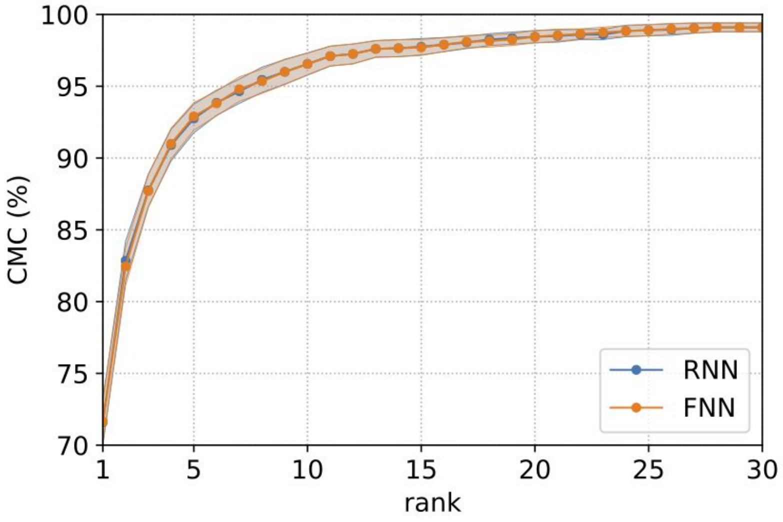
FRM (ours)

Data and experimental protocol

- Dataset 1: PRID2011 *[Hirzer et al., 2011]*
  - 200 identities, average length: 100 frames / track
- Dataset 2: iLIDS-VID *[Wang et al., 2014]*
  - 300 identities, average length: 71 frames / track
- Data splits
  - Train/test set with half of the identities each
  - Performance averaged over 20 splits
- Evaluation metric: CMC (equivalent to mean accuracy at rank k)

Experiment: Influence of the recurrent connection

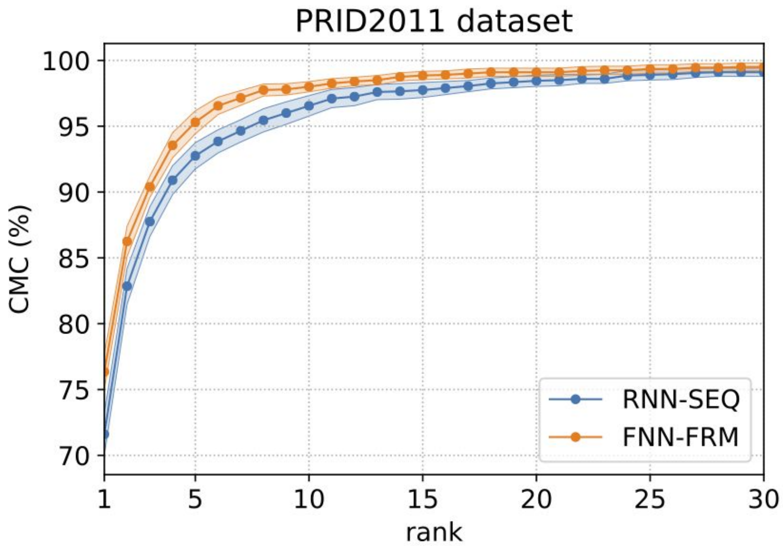
- Train weights on RNN-SEQ (RNN architecture, SEQ training protocol)
- Evaluate on RNN and FNN using the weights directly (**no re-training**)
- Same performance obtained

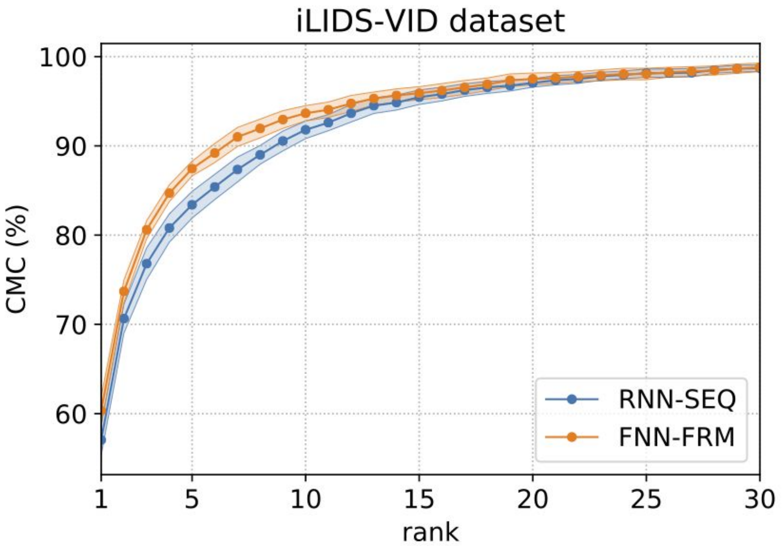


PRID2011 dataset

Experiment: Comparison with baseline

- FNN-FRM (ours) outperforms RNN-SEQ
- More diversity in mini-batches allows for a much better training





Comparison with baseline (comprehensive)

- Our method outperforms the baseline for all ranks in both datasets

Dataset	PRID2011				iLIDS-VID			
Rank	1	5	10	20	1	5	10	20
RNN [12]	70	90	95	97	**58**	84	91	96
– (reproduced)	71.6	92.8	96.6	98.5	57.1	83.4	91.8	97.1
FNN-SEQ (ours)	72.3	92.9	96.4	98.4	**58.0**	84.2	92.0	97.3
FNN-FRM (ours)	**76.4**	**95.3**	**98.0**	**99.1**	**58.0**	**87.5**	**93.7**	**97.5**

CMC values (in %)

Comparison with state-of-the-art RNN methods

- Our method is considerably simpler than the other state-of-the-art RNN methods compared but still achieves comparable performance results

Dataset	PRID2011				iLIDS-VID			
Rank	1	5	10	20	1	5	10	20
RNN [12]	70	90	95	97	58	84	91	96
– (reproduced)	71.6	92.8	96.6	98.5	57.1	83.4	91.8	97.1
RFA-Net [18]	58.2	85.8	93.4	97.9	49.3	76.8	85.3	90.0
Deep RCN [15]	69.0	88.4	93.2	96.4	46.1	76.8	89.7	95.6
Zhou *et al.* [25]	**79.4**	94.4	-	**99.3**	55.2	**86.5**	-	97.0
BRNN [20]	72.8	92.0	95.1	97.6	55.3	85.0	91.7	95.1
ASTPN [17]	**77**	**95**	**99**	99	**62**	86	**94**	**98**
Chen *et al.* [2]	**77**	93	95	98	**61**	85	**94**	97
FNN-FRM (ours)	76.4	**95.3**	**98.0**	**99.1**	58.0	**87.5**	**93.7**	**97.5**

CMC values (in %)

Stanford University

Conclusions

- Simple feed-forward RNN approximation with similar representational power
- New training protocol to leverage multiple video sequences within a mini-batch
- Results significantly and consistently improved compared to baseline
- Results on par or better than other published work based on RNNs, with a much simpler technique
- Faster model training compared to RNN baseline

Questions?