Learning Temporal Dimension from Spatial Dimension in Action Recognition Using Capsule Network

Abdullah M. Algamdi[⋆], Victor Sanchez[⋆], and Chang-Tsun Li[†]
[⋆] Dept. of Computer Science, University of Warwick, UK
[†] School of Information Technology, Deakin University, Australia

Overview

We propose a CapsNet architecture that:

- Employs individual video frames for human action recognition without explicitly extracting motion information.
- Propose weight pooling to reduce the computational complexity and improve the classification accuracy by appropriately removing some of the extracted features.

The proposed capsules architecture encodes temporal information by using the spatial features. Compared with a traditional CNN of the same complexity, the proposed CapsNet improves action recognition performance by 19.72% and 32.69% on the KTH and UCF-sports datasets, respectively.

Introduction

CNNs do not encode the spatial relationship among the learned features due to multiple pooling layers. Capsule Networks have been recently introduced to overcome some of the CNNs weaknesses. CapsNets preserve detailed information about an object's location and pose throughout the network. This helps the network to learn:

- All the part-whole relationships.
- Determine the precise location of the extracted features.
- Build a hierarchical representation of objects composed of a hierarchy of parts.

Based on the promising results CapsNets have attained, we propose a 2D architecture based on CapsNet for Human Action Recognition(HAR).

Proposed CapsNet for HAR

The proposed CapsNet architecture for HAR is depicted in Fig. 1.

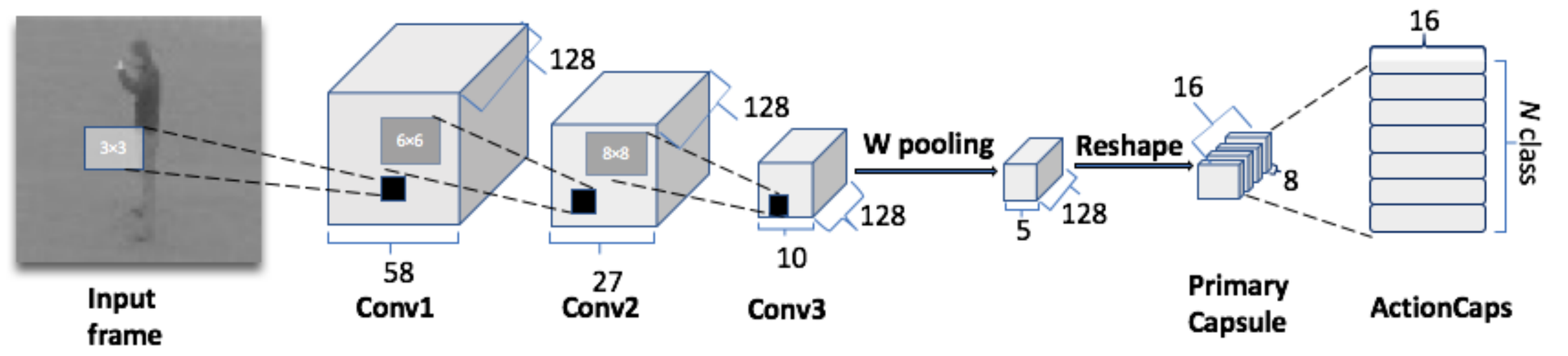

Figure 1: Proposed CapsNet architecture.

The input to the network are gray-level video frames. Three 2D ReLU Conv layers with a kernel size of $\{3 \times 3, 6 \times 6, 8 \times 8\}$ and a stride of $\{1, 2, 2\}$, respectively, first extract 128 feature maps, each. The network uses cross entropy as the loss function for classification:

$$H(p,q) = -\sum_x p(x)\log(q(x)), \qquad (1)$$

Weight pooling: instead of increasing the stride value of the Conv layers or adding several pooling layers, we propose weight pooling, which is inspired by stochastic pooling . Differently from max pooling, which outputs the largest value within a window of values, our weight pooling computes weighted contributions for each value within the window.
Our CapsNet architecture uses a single weight pooling layer with a 2×2 window size and a stride of 2. This guarantees that the pooling operation is only applied once to each 2×2 region.

Experiments

We measure the performance of our CapsNet architecture for HAR on the KTH and UCF-sports datasets
We evaluate five distinct approaches:

- A baseline CNN
- A baseline CapsNet
- Our proposed CapsNet w/ no pooling
- Our proposed CapsNet w/ max pooling
- Our proposed CapsNet w/ weight pooling

The baseline CNN extracts the same amount of features as compared to our proposed CapsNet. Therefore, evaluating the performance of this network allows determining any improvements introduced by the capsules.

Results

Table 1: Frame-level CCR (%) and # of parameters (millions) of various networks on the KTH and UCF-sports datasets

Network	param.	KTH	UCF
Baseline CNN	3.3	61.41%	68.18%
Baseline CapsNet	~21.4 - 31.4	64.34%	84.91%
Proposed CapsNet (no pooling)	~2.8 - 3.8	68.25%	81.42%
Proposed CapsNet (max pooling)	~1.9 - 2.2	71.71%	89.61%
Proposed CapsNet (weight pooling)	~1.9 - 2.2	**73.52%**	**90.47%**

Discussions

- Our proposed CapsNets outperforms the baseline CNN on both the KTH and UCF-sports datasets
- It is still far from the results achieved by some of the state-of-art CNNs which has several pooling layers and very complex architectures.
- Our CapsNet is a very simple architecture with only three Conv layers.
- Results give a notion of the power of capsules and routing-by-agreement for HAR even with a very simple architecture.
- Figure 2 graphically depict what the ActionCaps encode for the KTH dataset in terms of the space of variations in the way an action is instantiated.

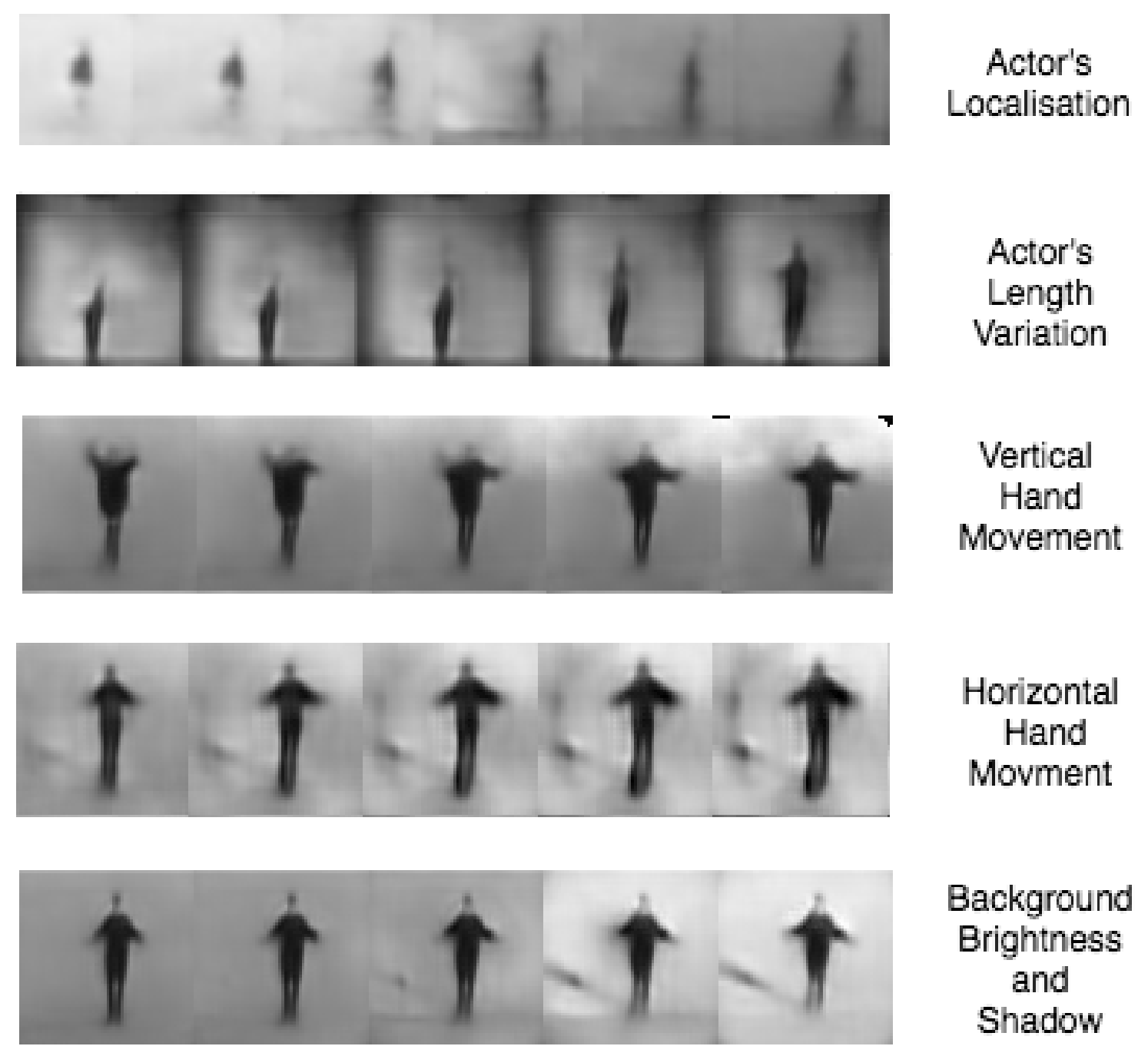

Figure 2:
Manipulation of ActionCaps for the KTH dataset. Each row represents 5 frames reconstructed after modifying one of the 16 dimensions of the output vector by $+/-0.05$.

Conclusions

We investigated the power of CapsNets for HAR by proposing a simple architecture that can accurately classify actions from video sequences without explicitly extracting temporal information. We also introduced weight pooling to improve the classification accuracy compared to no pooling and max pooling. Our proposed CapsNet outperforms a traditional CNN of similar complexity by 19.72% and 32.69% on the KTH and UCF-sports datasets, respectively, when both networks extract the same features.

Contact Information

Email a.algamdi@warwick.ac.uk
Phone +44 (795) 771 3931