A Hierarchical Quasi-Recurrent approach to Video Captioning

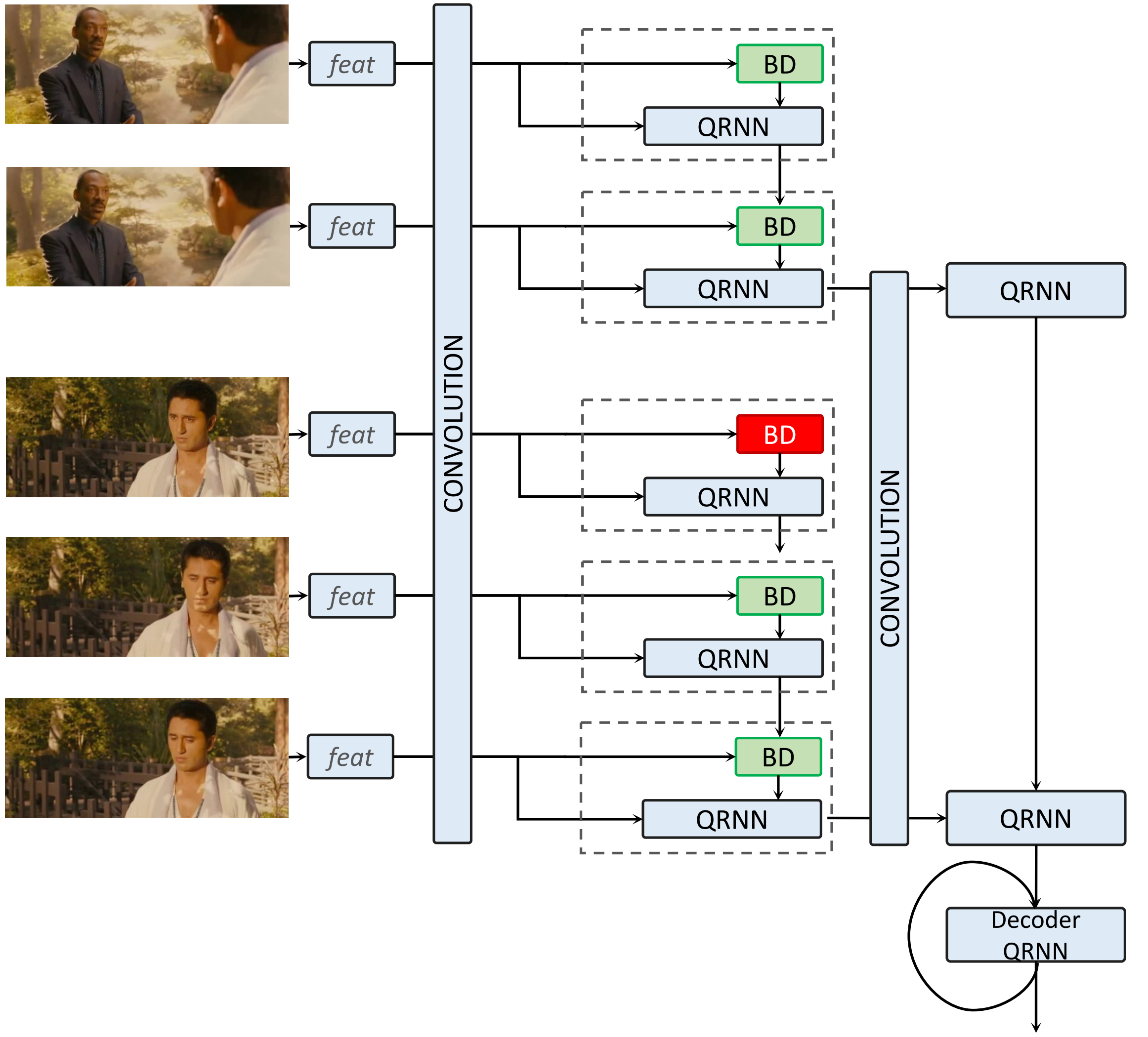
F. Bolelli, L. Baraldi, C. Grana

Università degli Studi di Modena e Reggio Emilia, DIEF, *Italy*

Introduction

- Video captioning has picked up a considerable attention in the last decade;
- Recurrent networks are a popular choice as video encoders for captioning, however ..
  - they require a significantly long training time;
  - they can not optimally deal with long video sequences;
- **The memory of the LSTM (Long Short-Term Memory) mixes representations computed while attending at different actions and appearances.**

Goals

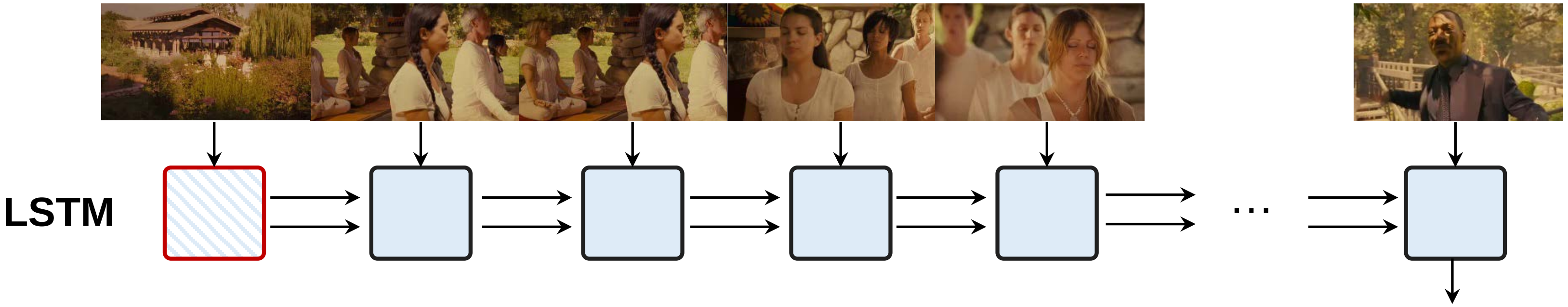


«someone is talking to a man»

- Employing QRNNs (Quasi-Recurrent Neural Networks) to allow parallel computation across both time and minibatch dimensions, enabling:
  - High throughput
  - Good scaling
- Introducing a video encoding architecture capable of identifying temporal boundaries and producing a better video representation.

AImage Lab

Long Short-Term Memory (LSTM)



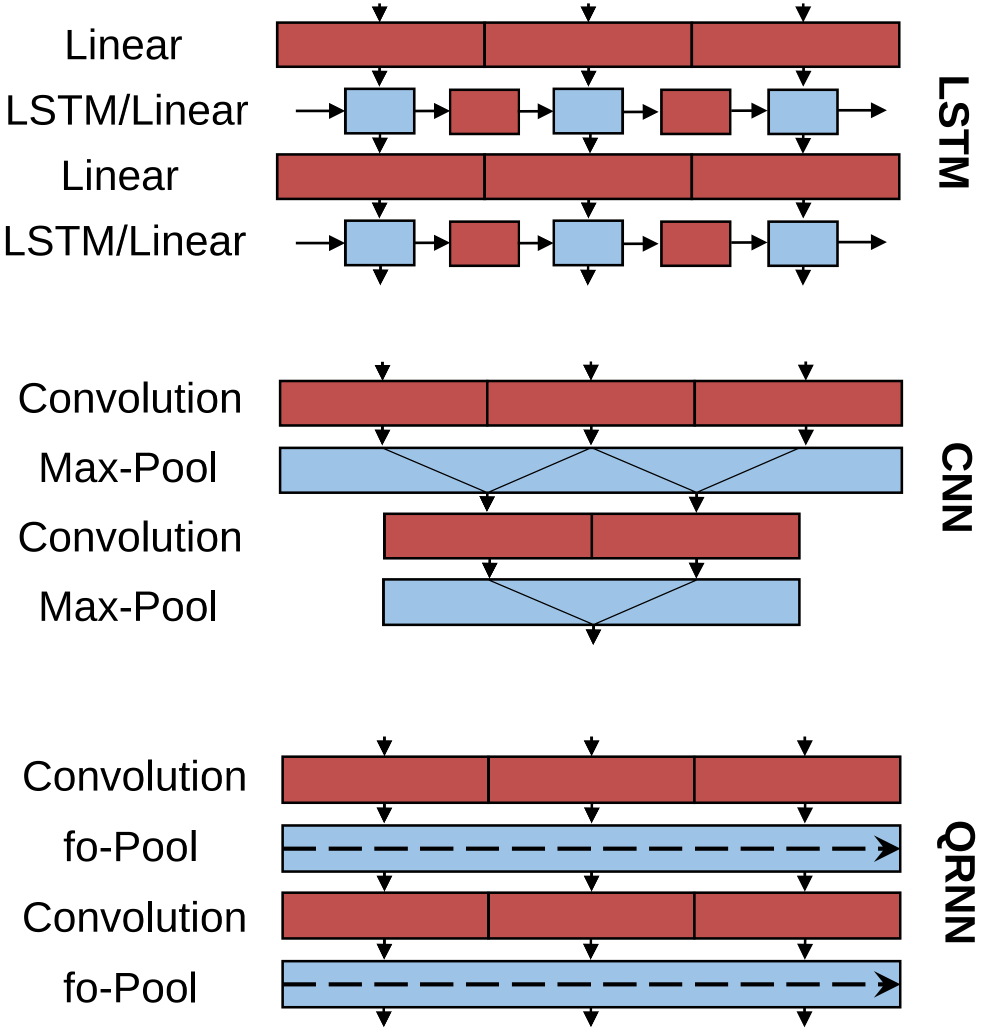
- Dyamic average pooling variant of LSTM:

$$h_t = f_t \odot h_{t-1} + (1 - f_t) \odot z_t$$

where

$$z_t = \tanh(W_{xz} x_t + W_{hz} h_{t-1} + b_z)$$
$$f_t = \sigma(W_{xf} x_t + W_{hf} h_{t-1} + b_f)$$

Quasi-Recurrent Neural Networks [1]



- Convolution on timestamp dimensions:

$$Z = \tanh(Wz * X)$$
$$F = \sigma(W_f \; * \; X)$$
$$O = \sigma(Wo * \; X)$$

where $X \epsilon \mathbb{R}^{T\times n}$

- Pooling subcomponents:

f-pooling $\left\{ h_t = f_t \odot h_{t-1} + (1 - f_t) \odot z_t \right.$

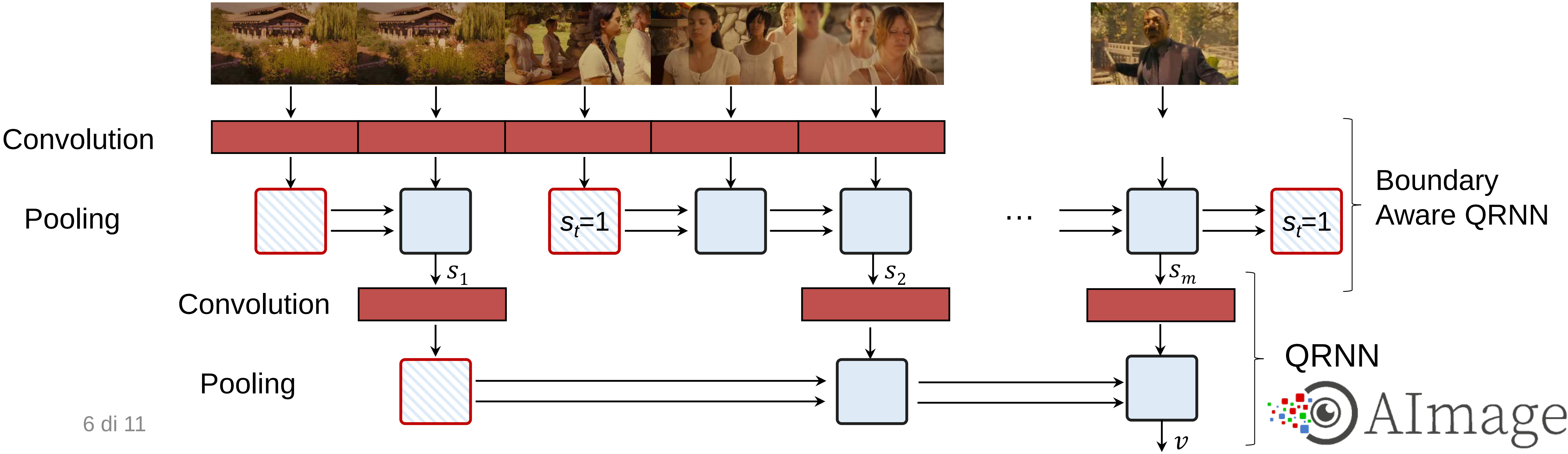
fo-pooling $\left\{ \begin{array}{l} c_t = f_t \odot c_{t-1} + (1 - f_t) \odot z_t \\ h_t = o_t \odot c_t \end{array} \right.$

ifo-pooling $\left\{ \begin{array}{l} c_t = f_t \odot c_{t-1} + i_t \odot z_t \\ h_t = o_t \odot c_t \end{array} \right.$

[1] J. Bradbury, S. Merity, C. Xiong, and R. Socher, "Quasi-recurrent neural networks," in ICLR. Toulon, France: OpenReview.net, 2017.

The Hierarchical Approach

- The proposed video encoder process the input video in a hierarchical fashion:
  - $(s_1, s_2, s_3, \dots, s_m)$ is the first level representation based on connectivity schema that varies with both the current input and the hidden state.
  - The second recurrent layer encodes this variable-length representation into a feature vector for the overall video.



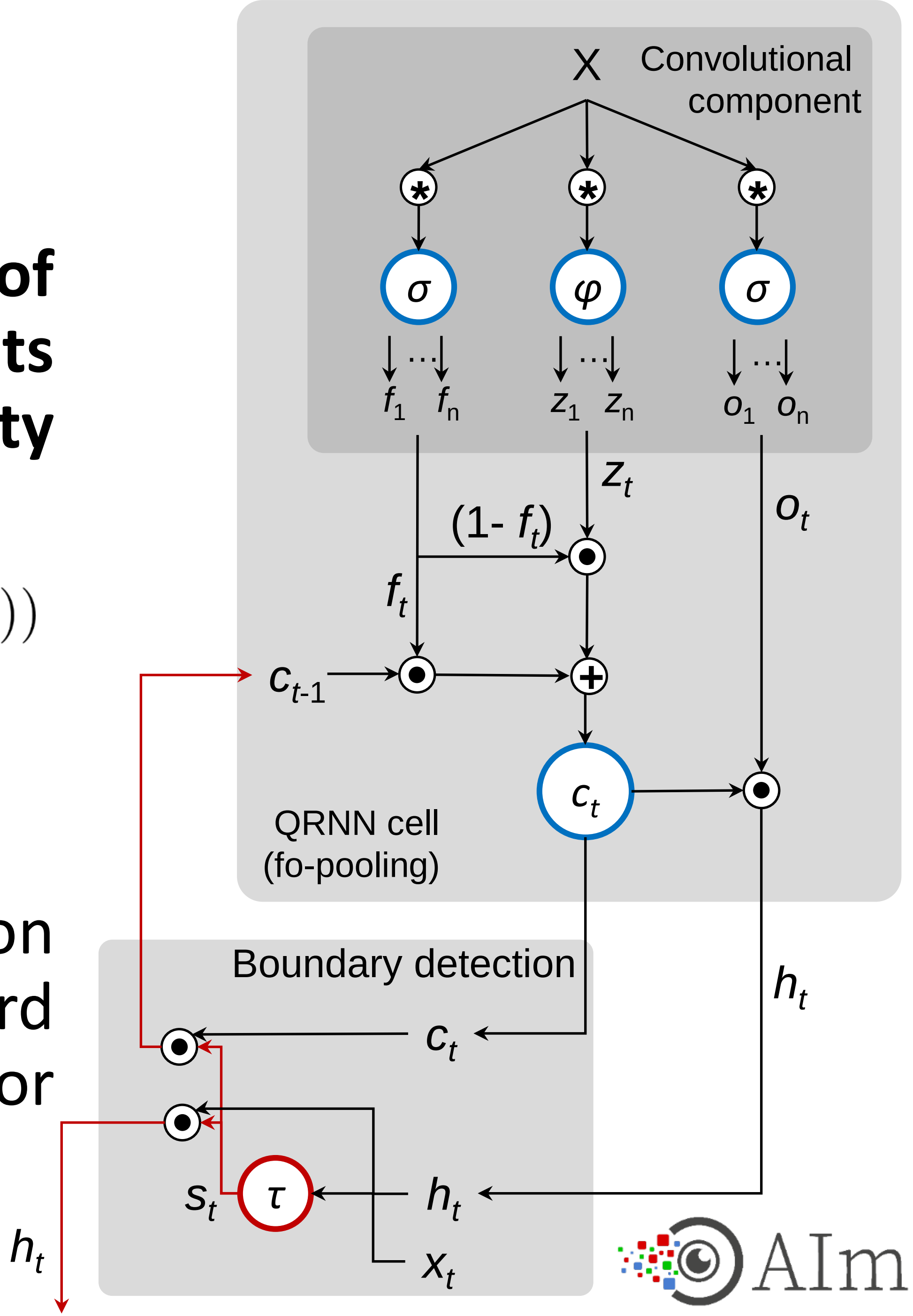
The Boundary Detector

- **A video encoding cell capable of identifying discontinuity points and modify the layer connectivity through time.**

$$s_t = \tau(\mathbf{v}_s^T \cdot (W_{si}\mathbf{x}_t + W_{sh}\mathbf{h}_{t-1} + \mathbf{b}_s))$$

$$\tau(x) = \begin{cases} 1, & \text{if } \sigma(x) > 0.5 \\ 0, & \text{otherwise} \end{cases}$$

- During training: stochastic version of the step function in the forward pass, and a differentiable estimator in the backward pass.



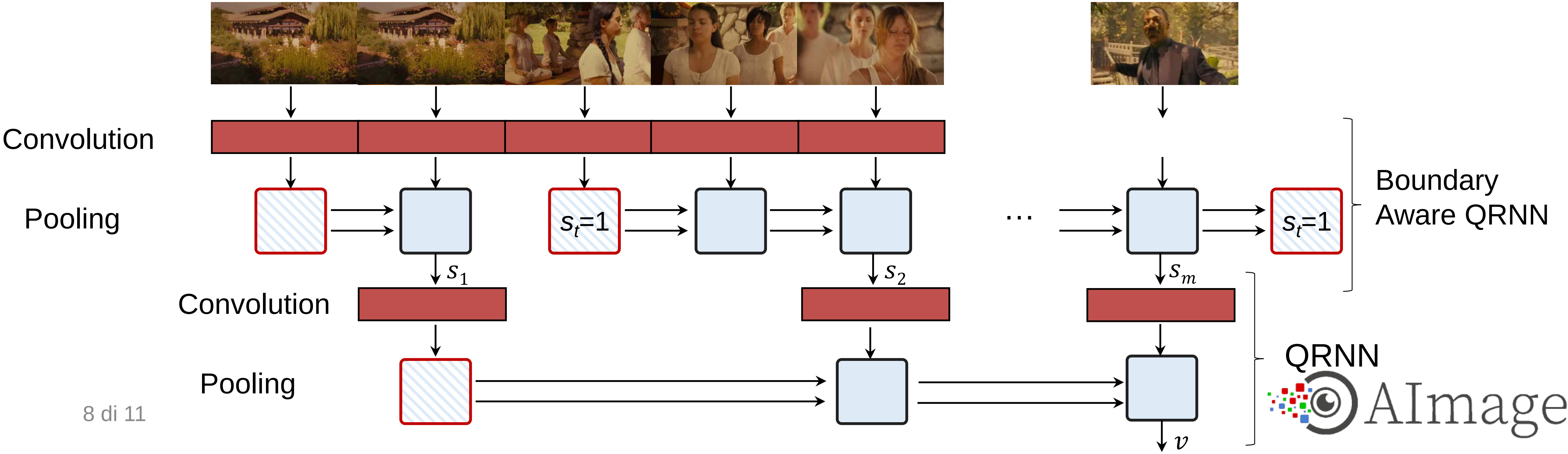
AImage Lab

The Boundary Detector

- When a boundary is estimated, the hidden state and memory cell are reinitialized, and the previous hidden state is given to the output, as a summary of the detected segment:

$$\mathbf{h}_{t-1} \leftarrow \mathbf{h}_{t-1} \cdot (1 - s_t)$$
$$\mathbf{c}_{t-1} \leftarrow \mathbf{c}_{t-1} \cdot (1 - s_t)$$

- **The connectivity schema of the layer is thought as an activation rather than as a non-learnable hyperparameter.**



Training and Sentence Generation

- The boundary detector is treated as a stochastic neuron during forward:

$$\tau(x) = 1_{\sigma(x)>z}\, , with\ z\ \sim U[0,1]$$

where $U[0,1]$ is the uniform probability distribution over [0,1]

- and as a differentiable estimator during backward:

$$\frac{\partial\tau}{\partial x}(x) = \sigma(x)(1 - \sigma(x))$$

- Decoder: optimize the log-likelihood of correct words over the sequence

$$\max_{w} \sum_{t=1}^{T} \log Pr\,(y_t | y_{t-1}, y_{t\,-2}, \ldots, y_0, v)$$

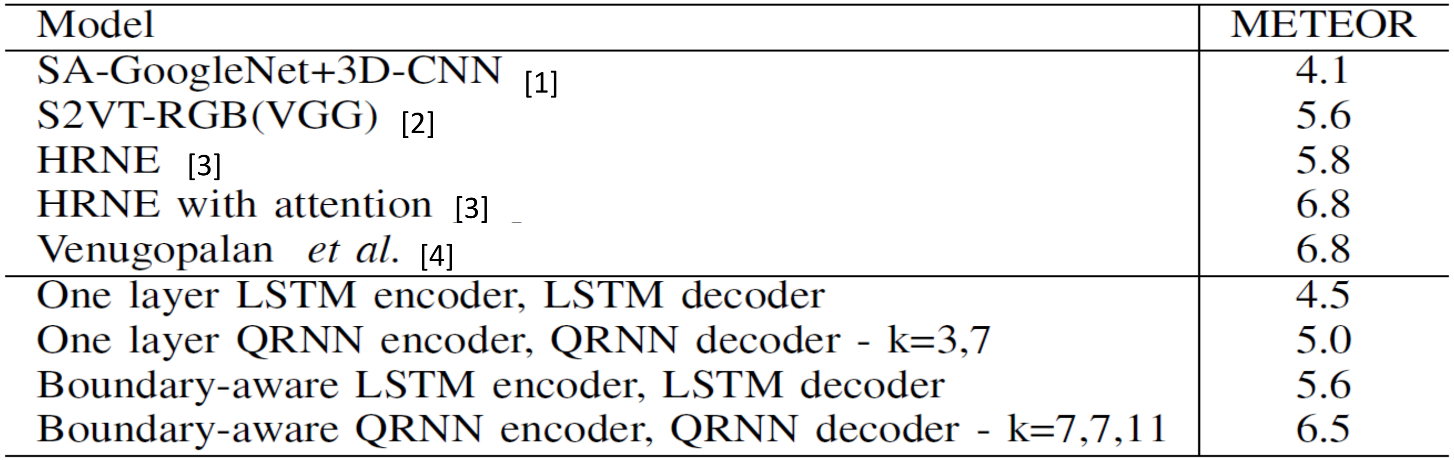
the probability of a word is modeled via a softmax layer applied on the output of the decoder.

v: video vector produced by the encoder

$y_0, y_1, \ldots, y_T$: sentence encoded with one-hot vector

Experimental Results

- Performed on the Montreal Video Annotation Dataset (M-VAD):
  - 36,921 training clips
  - 4,651 validation clips
  - 4,951 test clips
- .. with the Microsoft CoCo evaluation toolkit:



Model	METEOR
SA-GoogleNet+3D-CNN [1]	4.1
S2VT-RGB(VGG) [2]	5.6
HRNE [3]	5.8
HRNE with attention [3]	6.8
Venugopalan *et al.* [4]	6.8
One layer LSTM encoder, LSTM decoder	4.5
One layer QRNN encoder, QRNN decoder - k=3,7	5.0
Boundary-aware LSTM encoder, LSTM decoder	5.6
Boundary-aware QRNN encoder, QRNN decoder - k=7,7,11	6.5

- QRNN and LSTM have a similar epoch time.
- QRNN converges in 1/3 of the epochs required by LSTM.

[1] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville, “Describing videos by exploiting temporal structure,” in ICCV, 2015, pp. 4507–4515.

[2] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, “Sequence to sequence-video to text,” in ICCV, 2015, pp. 4534–4542.

[3] P. Pan, Z. Xu, Y. Yang, F. Wu, and Y. Zhuang, “Hierarchical recurrent neural encoder for video representation with application to captioning,” CVPR, 2016.

[4] S. Venugopalan, L. A. Hendricks, R. Mooney, and K. Saenko, “Improving lstm-based video description with linguistic knowledge mined from text,” in Conference on Empirical Methods in Natural Language Processing (EMNLP), 2016.

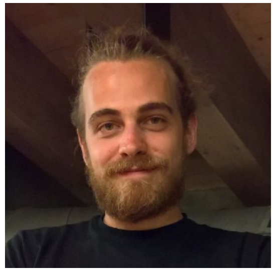
Conclusions

- We introduced a novel video encoding architecture for captioning which combines the effective QRNN in a hierarchical structure.
- The connectivity over time of the QRNN layer is changed when an action discontinuity is detected.
- Experimental results on the M-VAD dataset are comparable with the state-of-the-art on movie description, with a fraction of the required training time.

A Hierarchical Quasi-Recurrent approach to Video Captioning

Thank You!

federico.bolelli@unimore.it



Federico Bolelli

Lorenzo Baraldi

Costantino Grana