ActionSpotter: Deep Reinforcement Learning Framework for Temporal Action Spotting in Videos

Guillaume VAUDAUX-RUTH[1,2], Adrien CHAN-HON-TONG[1,3], Catherine[2] ACHARD

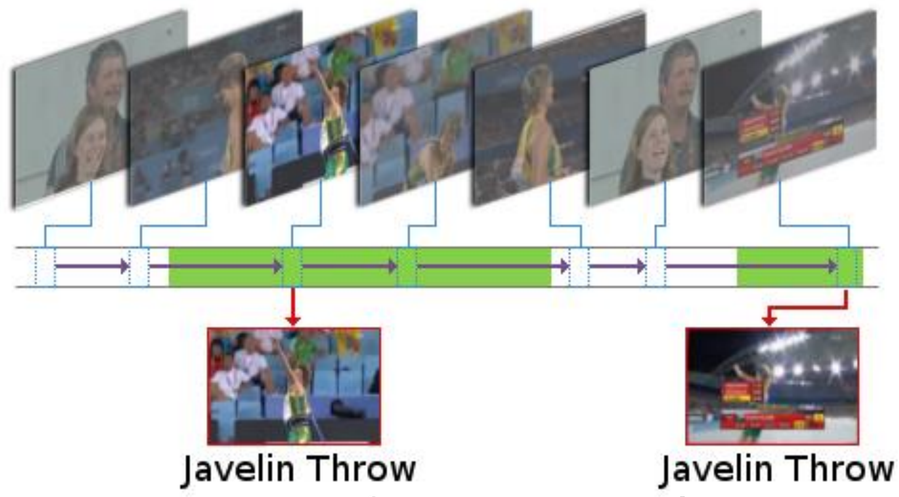
[1]ONERA [2]Sorbonne Université [3]Université Paris-Saclay

Context

Action spotting has recently been proposed as an alternative to action detection [1] and key frame extraction. However, the current state-of-the-art method of action spotting requires an expensive ground truth.

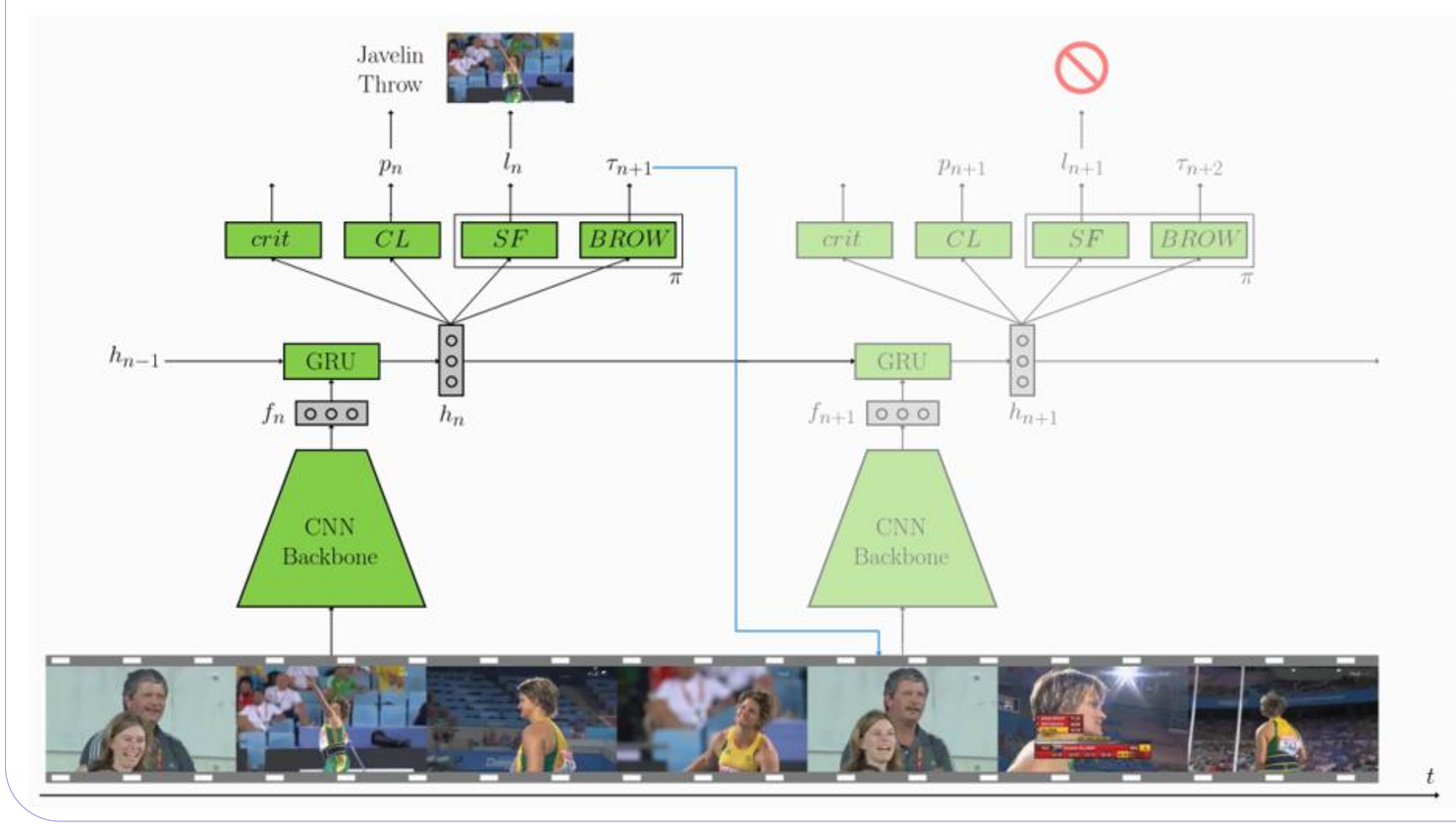
In this work, we propose to use a reinforcement learning algorithm to perform efficient action spotting using only the temporal segments from action detection annotations.

Action Spotting Overview



It extracts one spot frame per action occurrence (in green in the figure) by sparsely browsing the video.

Action Spotting Framework



In a first stage, the CNN backbone encodes the frame into a feature vector which is then forwarded to a GRU layer. The resulting hidden state vector is then individually processed by (SF), (CL), (BROW) and (*crit*). The (SF) stage deals with the decision of turning the current frame into a spot frame or skip it. The (CL) stage predicts the action class related to the spot frame and the (BROW) stage outputs the next video frame to look at. The (*crit*) stage is only used to ensure better convergence in the reinforcement learning framework.

Action Spotter Optimization

The global policy has to maximize the final mAP of the video being processed. Thus our local reward at step *n* is the difference of AP between step n and n-1 plus an entropy term.

$$r_{\pi,n} = \text{mAP}(\mathcal{V}_n) - \text{mAP}(\mathcal{V}_{n-1}) + \rho\mathcal{H}(\pi(n))$$

and, the cumulative discounted reward is:

$$R_{\pi,n} = \sum_{k=0}^{N-n-1} \gamma^k r_{\pi,k+n}$$

Then, the global loss is:

$$\mathcal{L}_{global} = \mathcal{L}_{cls} + \lambda_1 \mathcal{L}_{critic} - \lambda_2 J_{actor}$$

with L_{cls} the loss of the classification network (CL) wich is a cross-entropy loss, L_{critic} is the MSE between the estimation of the value function of (*crit*) and the real one.

As J_{actor} objective is non differentiable, we use REINFORCE to derive the expected gradient:

$$\nabla J_{actor} = \nabla \mathbb{E}\left[\sum_{n=1}^{N} \log(\pi(n))(R_{\pi,n} - \mathbb{E}[R_{\pi,n}|h_n])\right]$$

Results

THUMOS'14						
Approach	Detection mAP@					Spotting mAP
	0.1	0.2	0.3	0.4	0.5	
Glimpses [5]	48.9	44.0	36.0	26.4	17.1	-
SMS [30]	51.0	45.2	36.5	27.8	17.8	-
M-CNN [31]	47.7	43.5	36.3	28.7	19.0	41.2
CDC [32]	-	-	41.3	30.7	24.7	31.5
TURN [33]	54.0	50.9	44.1	34.9	25.6	44.8
R-C3D [34]	54.5	51.5	44.8	35.6	28.9	52.2
SSN [35]	*66.0*	*59.4*	51.9	41.0	29.8	-
A-Search [14]	-	-	51.8	42.4	30.8	-
CBR [36]	60.1	56.7	50.1	41.3	31.0	50.1
BSN + UNet [37]	-	-	53.5	45.0	36.9	-
Re-thinking F-RCNN [38]	59.8	57.1	53.2	48.5	42.5	-
D-SSAD [39]	-	-	*60.2*	*54.1*	*44.2*	*59.7*
Ours (TSN backbone)	-	-	-	-	-	62.4
Ours (I3D backbone)	-	-	-	-	-	**65.6**

ActivityNet v1.2					
Approach	Detection mAP@				Spotting mAP
	0.5	0.75	0.95	Avg	
W-TALC [45]	37.0	14.6	-	18.0	-
SSN-SW [35]	-	-	-	18.1	-
3C-Net [46]	37.2	23.7	9.2	21.7	-
FPTADC [47]	37.6	21.8	2.4	21.9	-
SSN-TAG [35]	39.2	25.3	5.4	25.9	55.4
BSN [48]	46.5	30.0	8.0	30.0	49.6
BMN [49]	50.1	34.8	8.3	33.85	55.3
Ours (TSN backbone)	-	-	-	-	58.1
Ours (I3D backbone)	-	-	-	-	**60.2**

Comparison with methods designed to detect actions.

References

[1] H. Alwassel, F. Caba Heilbron, and B. Ghanem, "Action search: Spotting actions in videos and its application to temporal action localization," in Proceedings of the European Conference on Computer Vision (ECCV), 2018.