Temporal Activity Detection in Untrimmed Videos with Recurrent Neural Networks

Alberto Montes
ETSETB TelecomBCN
Universitat Politècnica de Catalunya
Barcelona, Catalonia/Spain
malberto@student.ethz.ch

Amaia Salvador
Image Processing Group
Universitat Politècnica de Catalunya
Barcelona, Catalonia/Spain
amaia.salvador@upc.edu

Santiago Pascual
TALP Research Center
Universitat Politècnica de Catalunya
Barcelona, Catalonia/Spain
santiago.pascual@tsc.upc.edu

Xavier Giro-i-Nieto
Image Processing Group
Universitat Politècnica de Catalunya
Barcelona, Catalonia/Spain
xavier.giro@upc.edu

Abstract

This work proposes a simple pipeline to classify and temporally localize activities in untrimmed videos. Our system uses features from a 3D Convolutional Neural Network (C3D) as input to train a a recurrent neural network (RNN) that learns to classify video clips of 16 frames. After clip prediction, we post-process the output of the RNN to assign a single activity label to each video, and determine the temporal boundaries of the activity within the video. We show how our system can achieve competitive results in both tasks with a simple architecture. We evaluate our method in the ActivityNet Challenge 2016, achieving a 0.5874 mAP and a 0.2237 mAP in the classification and detection tasks, respectively. Our code and models are publicly available at at: https://github.com/imatge-upc/activitynet-2016-cvprw

1 Introduction

Recognizing activities in videos has become a hot topic over the last years due to the continuous increase of video capturing devices and online repositories. This large amount of data requires an automatic indexing to be accessed after capture. The recent advances in video coding, storage and computational resources have boosted research in the field towards new and more efficient solutions for organizing and retrieving video content.

Impressive progress has been reported in the recent literature for video classification [7–10], which requires to assign a label for the input video. While this task is already challenging, it has typically been explored with videos to be trimmed beforehand. However, a video classification system should be able to recognize activities in untrimmed videos, and find the temporal segments in which they appear. This second challenge has been recently proposed in the ActivityNet Challenge 2016 [3], in which participants are asked to both provide a single activity for each video, as well as the temporal segment where the activity happened in the video. In order to face both these challenges at the same time, we propose a simple pipeline composed of a 3D-CNN that exploits spatial and short temporal correlations, followed by a recurrent neural network which exploits long temporal correlations.

1st NIPS Workshop on Large Scale Computer Vision Systems (LSCVS 2016), Barcelona, Catalonia/Spain.

2 Related work

Several works in the literature have used 2D-CNNs to exploit the spatial correlations between frames of a video by combining their outputs using different strategies [4, 12, 1]. Others have tried using the optical flow as an additional input to the 2D-CNNN [9], which provides information of the temporal correlations.

Later on, 3D-CNNs were proposed in [7] (known as C3D), which were able to exploit short temporal correlations between frames and have demonstrated to work remarkably well for video classification [7, 8]. C3D have also been used for temporal detection in [6], where multi-stage C3D architecture is used to classify video segment proposals.

For temporal activity detection, recent works have proposed the usage of Long Short-Term Memory units (LSTM) [5]. LSTMs are a type of RNNs that are able to better exploit long and short temporal correlations in sequences, which makes them suitable for video applications. LSTMs have been used alongside CNNs for video classification [10] and activity localization in videos [11].

In this paper, we combine the capabilities of both 3D-CNNs and RNNs into a single framework. This way, we design a simple network that takes a sequence of video features from the C3D model [7] as input to a RNN and is able to classify each one of them into an activity category.

3 Proposed Architecture

We use the C3D model proposed in [7] to extract features for all videos in the database. We split the videos in 16-frames clips and resize them to 171×128 to fit the input of the C3D model. Features from the second fully connected layer (fc6) are extracted for each video clip.

3.1 Architecture

We design a network that processes a sequence of C3D-f6 features from a video, and returns a sequence of class probabilities for each 16-frames clip. We use LSTM layers, trained with dropout with probability $p = 0.5$ and a fully connected layer with a softmax activation. Figure 1 shows the proposed architecture. Different configurations of the number of LSTM layers N and the number of cells c have been tested and are compared in Section 4.3. Our proposed system has the following architecture: `input(4096) - dropout(0.5) - N` $\times$ `lstm(c) - dropout(.5) - softmax(K+1)` where K is the number of activity classes at the dataset.

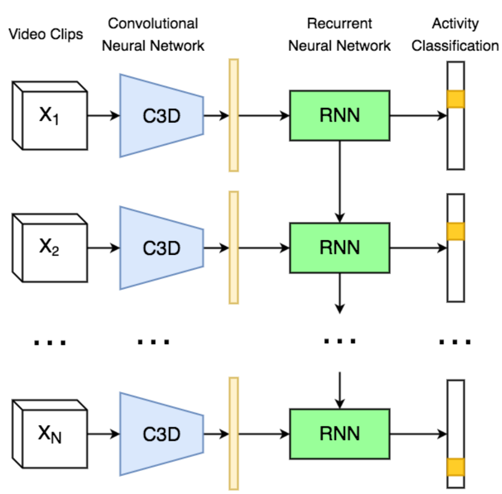

Figure 1: Global architecture of the proposed pipeline.

3.2 Post-Processing

Given a video, the prediction of our model is sequence of class probabilities for each 16-frame video clip. This output is post-processed to predict the activity class and temporally localize it. First, to obtain the activity prediction for the whole video, we compute the average of the class probabilities over all video clips in the video. We consider the class with maximum predicted probability as the predicted class.

To obtain the temporal localization of the predicted activity class, we first apply a mean filter of k samples to the predicted sequence to smooth the values through time (see Equation 1). Then, the probability of *activity* (vs *no activity*) is predicted for each 16-frames clip, being the *activity* probability the sum of all probabilities of activity classes, and the *no activity* probability, the one assigned to the background class. Finally, only those clips with an *activity* probability over a threshold γ are kept and labeled with the previously predicted class. Notice that, for each video, all predicted temporal detections are activity class.

$$\tilde{p}_i(x) = \frac{1}{2k} \sum_{j=i-k}^{i+k} p_i(x) \tag{1}$$

4 Experiments

4.1 Dataset

For all our experiments we use the dataset provided in the ActivityNet Challenge 2016 [3]. This dataset contains 640 hours of video and 64 million frames. The ActivityNet dataset is composed of untrimmed videos, providing temporal annotations for the given ground truth class labels. The dataset is split in 50% for training, 25% for validation and 25% for testing.

4.2 Training

We train the network described in Section 3.1 with the negative log likelihood loss, assigning a lower weight to background samples to deal with dataset imbalance (see Equation 2).

$$L(p,q) = -\sum_x \alpha(x) p(x) \log(q(x)), \text{ where } \alpha(x) = \begin{cases} \rho, & x = \text{background instance} \\ 1, & \text{otherwise} \end{cases} \tag{2}$$

where q is the predicted probability distribution and p the ground truth probability distribution. In our experiments, we set $\rho = 0.3$.

The network was trained for 100 epochs, with a batch size of 256, where each sample in the minibatch is a sequence of 20 16-frame video clips. We use RMSprop [2] with a learning rate set to 10^{-5}.

4.3 Results

We evaluate our models using the metrics proposed in ActivityNet Challenge. For video classification, we use mean average precision (mAP) and Hit@3. For temporal localization, a prediction is marked correct only when it has the correct category and has IoU with ground truth instance larger than 0.5, and mAP is used to evaluate the performance over the entire dataset.

Architecture	**mAP**	**Hit@3**
3 x 1024-LSTM	0.5635	0.7437
2 x 512-LSTM	0.5492	0.7364
1 x 512-LSTM	**0.5938**	**0.7576**

Table 1: Results for classification task comparing different architectures.

γ	$k = 0$	$k = 5$	$k = 10$
0.2	0.20732	**0.22513**	0.22136
0.3	0.19854	0.22077	0.22100
0.5	0.19035	0.21937	0.21302

Table 2: mAP with an IoU threshold of 0.5 comparing between values of k and γ on post-processing.

Table 1 shows the performance of different network architectures. We tested configurations with different number of LSTM layers and different number of cells. These results indicate that all the networks presented high learning capacity over the data, but some over-fitting was observed with the deeper architectures, obtaining the best results with a single layer of 512-LSTM cells.

Fixing the architecture, we performed experiments for the temporal detection task using different values of k and γ in the post-processing stage. Table 2 shows results for the temporal activity localization task, where the effect of a mean smoothing filter can be seen, improving the localization performance. Figures 2 and 3 show some examples of classification and temporal localization prediction for some instances of the dataset.

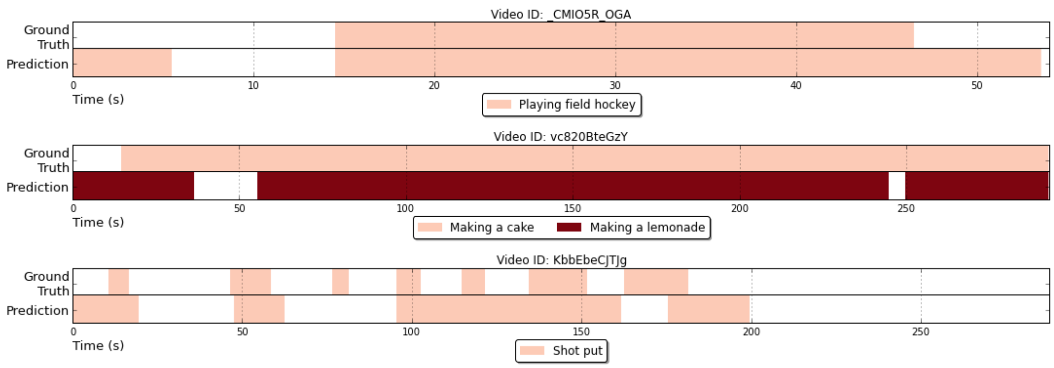

Figure 2: Examples of temporal activity localization predictions.

Video ID: ArzhjEk4j_Y
Ground Truth: Building sandcastles

Prediction:
0.7896 Building sandcastles
0.0073 Doing motocross
0.0049 Beach soccer

Video ID: AimG8xzchfI
Activity: Curling

Prediction:
0.3843 Shoveling snow
0.1181 Ice fishing
0.0633 Waterskiing

Figure 3: Examples of activity classification.

5 Conclusion

In this paper we propose a simple pipeline for both classification and temporal localization of activities in videos. Our system achieves competitive results on both tasks. The sequence to sequence nature of the proposed network offers flexibility to extend it to face more challenging tasks in video processing, e.g. where more than a single activity is present in the video. Future work will address end to end training of the model (3D-CNN + RNN), to learn better feature representations suitable for the dataset.

References

[1] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. Delving deeper into convolutional networks for learning video representations. *arXiv preprint arXiv:1511.06432*, 2015.

[2] Yann N Dauphin, Harm de Vries, Junyoung Chung, and Yoshua Bengio. Rmsprop and equilibrated adaptive learning rates for non-convex optimization. *arXiv preprint arXiv:1502.04390*, 2015.

[3] Bernard Ghanem Fabian Caba Heilbron, Victor Escorcia and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–970, 2015.

[4] Georgia Gkioxari, Ross Girshick, and Jitendra Malik. Contextual action recognition with r* cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1080–1088, 2015.

[5] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[6] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *CVPR*, 2016.

[7] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. *arXiv preprint arXiv:1412.0767*, 2014.

[8] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Deep end2end voxel2voxel prediction. *arXiv preprint arXiv:1511.06681*, 2015.

[9] Limin Wang, Yuanjun Xiong, Zhe Wang, and Yu Qiao. Towards good practices for very deep two-stream convnets. *arXiv preprint arXiv:1507.02159*, 2015.

[10] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. Describing videos by exploiting temporal structure. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4507–4515, 2015.

[11] Serena Yeung, Olga Russakovsky, Ning Jin, Mykhaylo Andriluka, Greg Mori, and Li Fei-Fei. Every moment counts: Dense detailed labeling of actions in complex videos. *arXiv preprint arXiv:1507.05738*, 2015.

[12] Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. End-to-end learning of action detection from frame glimpses in videos. *arXiv preprint arXiv:1511.06984*, 2015.