Chapter 1

Conclusion - Future work

1.1 Conclusion

In this thesis we explore the problem of action recognition and localization. We design a network base on [**?**] combined with some elements from [**?**], [**?**], [**?**], [**?**] and [**?**].

We write a pytorch implementation expanding code only from [**?**]. Futhermore, we wrote our own code using some CUDA functions designed by us (like calculating connection scores, modifying tubes etc).

We tried to design a design a Tube Proposal Network for proposing action tubes in given video segments, inspired by Faster R-CNN's RPN. We designed it using general anchors and not dataset specific anchors in order to try to generalize our approach for several datasets, on the contrary with the approach proposed by [**?**], in which it uses the most frequently appearing anchors as the general anchors.

On top of that, we designed a naive connection algorithm for connecting our proposed action tubes based on the one proposed by [**?**]. In our approach, we use the same scoring policy, which is a combination between actioness and overlaping scores. The main difference is that we avoid to calculate all the possible combinations using an updating threshold. We, also, tried another connection algorithm inspired by [**?**]. However, our implementation wasn't very good so, we didn't explore all of its pontentials.

Finally, we explored several classifiers for the classification stage of our network, which are a RNN, a SVM and a MLP. We used an implementation taken from Fast RCNN for the SVM classifier, which included hard negatives mining trainig procedure. Futhermore, we explore some training techniques for best classification performance and 2 training approaches, the classic one and using pre-extracted features.

1.2 Future work

There is a lot of room for improvement for our network, in order to achieve state-of-the-art results. The most important are described in next paragraphs.

Improving TPN proposals We implemented 2 networks for proposing action tubes in a video segment. We managed to achieve about 63% recall score for sample duration = 16 and about 80% recall for sample duration = 8. Theses scores show that there is plenty room for improvement especially for sample duration = 16. Even though a lot of networks' architectures have been explored for regression, a good idea would be to try other networks, not necessarily inspired by object detection networks like we did. On top of that, adding a λ factor in training loss would be a good idea and exploring which is the best approach. So training loss could be defined as:

$$L = \sum_i L_{cls}(p_i, p_i^*) + \lambda_1 \sum_i p_i^* L_{reg}(t_i, t_i^*) + \lambda_2 \sum_i q_i^* L_{reg}(c_i, c_i^*) \tag{1.1}$$

Furthermore, it would be a good idea to use SSD's ([**?**]) proposal network instead of RPN, in order to compare result. Finally, we could experiment using Feature Pyramid Networks, which could be extracted in 3 dimensions as another feature extractor or some other type of 3D ResNet.

Changing Connection algorithm In this thesis, another challenge we came was connecting proposed ToIs for proposing action tubes. We implemented a very naive algorithm, which wasn't able to give us very good proposals despite the changes we tried to do. We implemented another connection algorithm which was base in a estimation on temporal progress of an action and their overlap. Although it also didn't give us very good proposals, we believe that we should explore this algorithm's pontelials. That's because it takes advantage of the progress of the action, which the previous algorithm didn't.

Explore other classification techniques For classification stage, we experiment mainly on a SVM classifier for JHMDB dataset and we didn't get involved a lot with UCF dataset. We found the best feature maps from JHDMB and we used the same for UCF. We think that we should explore UCF's feature maps even though we believe that there will be the same. It is essential to confirm our assumption. In addition, we could try other classification techniques like random forest or experiment more with RNN classifier for the UCF dataset. Finally, another classification procedure would be a good idea, like extracting first all the possible action tubes and then using other network's features for classification stage.