BBM 418 - Computer Vision Laboratory

HACETTEPE UNIVERSITY

Department of Computer Engineering

Programming Assignment-4
Spring 2020-2021
Dr. Nazli Ikizler Cinbis

Due Date: 30.05.2021, 23:59pm

Single Object Tracking with Regression Networks

In this assignment, you will implement a basic single object tracker by training the network on the given videos. You will use a two-frame architecture so the network will use two features of frames at the same time. You will utilize VOT2017 dataset[1] for training and testing the tracker by considering the ground truth bounding box annotations.

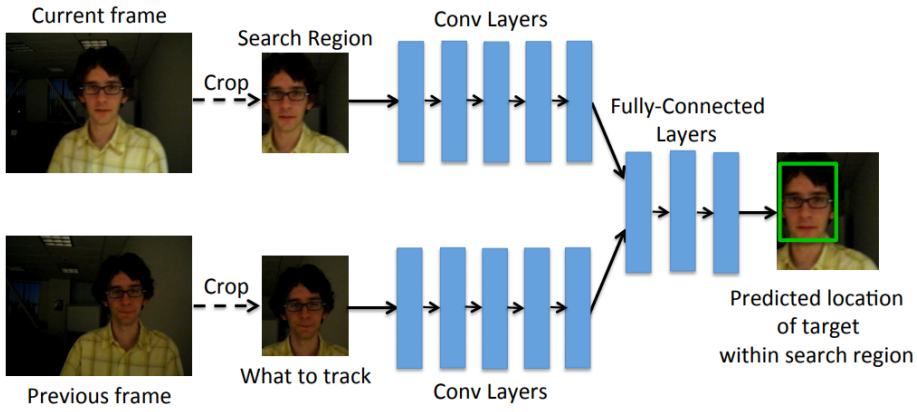

Figure 1: Overview of the tracker.

Background

What you are going to do is basically, trying to predict the position of the object at the current frame by leveraging from the position of it at the previous step. For this purpose, your network will learn to predict a bounding box fromthe given combination of features. This combination includes the feature map of previous frame's object position,and the feature map of current frame's possible object position which is a search region such as the enlarged version of the previous bounding box, with the assumption of: object should have been moved a little, so it should be closeto previous position. The network basically learns: given this object and a search region, object should move to this location (the ground-truth bounding box.) Divide your dataset into train and test set and give details.

1 Training

1. Prepare your training loader with consecutive frame pairs from training set of the dataset such as (t_0, t_1),(t_1, t_2),(t_2, t_3) etc. Shuffle the loader so that the network get random pairs of frames from random videos.
2. Crop the first frame, t_0, of the video by using the padded ground-truth bounding box which is the 2 times enlarged version of it from the center.
3. Extract features from the pretrained ResNet-18.
4. Crop the second frame, t_1, by using the search region which is the 2 times enlarged version of the previous frame's (t_0) bounding box from the center.
5. Repeat the Step 3.
6. Concatenate those two features so that the input dimension will be increased(twice the size).

7. Use fully connected layers such as FC-ReLU-FC-ReLU-FC and a residual connection between the first FC and the last FC layer that maps input features to ground-truth bounding box of the second frame, being $1x4$ vector. Your loss function will be MSEbetween predicted bounding box coordinates and the ground-truth.

2 Testing

1. For each test video, you will initialize the bounding box of the first frame from the ground-truth.
2. Then, your network will predict the bounding boxes of the rest. In other words, for other frame pairs, you will use the previous frame's **predicted** bounding box locations both for padded bounding box from Section 1 Step 2 and the search region from Section 1 Step 4.

3 Evaluation

- Give MSE loss graphics for the training process (training and validation) as well as the test set. This will be the average MSE of all pairs of frames from test set.
- You should give a visualized GIFs of 3 test videos (the best, average and bad result) where you put the predicted bounding boxes on each frame in your report.

The Implementation Details

1. You should pay attention to code readability such as comments, function/variable names and your code quality: 1) no hard-coding 2) no repeated code 3) cleanly separate and organize your code 4) use consistent style, indentation 5) write deterministic algorithms
2. Implement your code with Python 3 and use libraries from Anaconda. You can install any library that is not in Anaconda as well, such as OpenCV.
3. You should use the latest PyTorch as the deep learning framework. You can use Google Colab to run your experiments.

What should you write in the report?

- Give explanations for each step.
- Give experimental/visual results, used parameters and comments on the results in detail.
- Give your model's loss plot both for training and validation set during training.
- Put the results of different hyper-parameters (learning rate, batch size), the effect of them, with the loss plots.
- A basic structure might be: 1) Introduction (what is the problem, how do you approach to this problem, what is the content of your report) 2) Implementation Details (the method you followed and details of your solution) 3) Experimental Results (all results for separate parts with different parameters and your comments on the results) 4) Conclusion (what are the results and what are the weaknesses of your implementation, in which parts you have failed and why, possible future solutions)
- You should write your report in LaTeX
- You should give visual results by using a table structure.

What to Hand In

Your submission format will be:

- README.txt *(give a text file containing the details about your implementation, how to run your code, the organization of your code, functions etc.)*
- code/ *(directory containing all your code)*
- report.pdf

Archieve this folder as **b<studentNumber>.zip** and submit to `https://submit.cs.hacettepe.edu.tr`.

Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.

References

1. https://www.votchallenge.net/vot2017/dataset.html