Towards the understanding of object manipulations by means of combining common sense rules and deep networks

Máté Csákvári, András Sárkány

Abstract: Object detection on images and videos improved remarkably recently. However, state-of-the-art methods still have considerable shortcomings: they require training data for each object class, are prone to occlusions and may have high false positive or false negative rates being prohibitive in diverse applications. We study a case that a) has a limited goal and works in a narrow context, b) includes common sense rules on '*objectness*' and c) exploits state-of-the art *deep detectors* of different kinds. Our proposed method works on an image sequence from a stationary camera and detects objects that may be manipulated by actors in a scenario. The object types are not known to the system and we consider two actions: "taking an object from a table" and "putting an object onto the table". We quantitatively evaluate our method on manually annotated video segments and present precision and recall scores.

Keywords: computer vision, class-agnostic object detection, common sense, optical flow, unsupervised

Introduction

Objects appearing and disappearing is a natural part of environment dynamics. It takes place in space and time and for some tasks it is neccessary to keep track of them. For example passing of an object indirectly, when one person places and object onto a surface then another person takes it can be described as a sequence of object appearance and disappearance. The problem is that a general object definition would quickly lead to combinatorical explosion. We must find a more restrictive definition that we can handle. In this case, we consider everything that can be moved by a person with his/her hand, an object. Our base assumption is that an object does not move by itself. If there are no other motions in a scene then no object appearance or disappearance can take place. Therefore it must be a result of an actor acting upon the environment. Taking an object means, that it must be grabbed first, so there must have been some motion before and after that. Finding these points in time is our first task. We restrict ourselves to a fixed environment, namely actors placing and taking objects from a table. We propose an image difference based change detection algorithm. The simplest case is when the only difference in the images is the object. Taking the normalized image difference gives us the object that appeared or disappeared. Of course this ideal case rarely occurs. Our goal is to find images and filter them so that their difference is only the changing object.

Method

In our method's core lies the idea that, unless many interactions are happening at once, an appearing/disappearing object can be found by taking the image difference of two specific frames from a video of a stationary camera. We specify common sense assumptions and derive algorithmic components to:

1. select the two relevant frames
2. process the images to neglect irrelevant differences caused by interfering actions in the scene
3. take the image difference

Image selection

In this step we select two frames, I_{t_1} and I_{t_2}, that will be used at the image differencing step. We restrict ourselves to find objects that were moved by an actor. First, we find a point in time when the object was grabbed or put down, then search backwards and forward in time to find frames where the object is not fully occluded, to find I_{t_1} and I_{t_2}. This is done by simply checking if the bounding boxes of the hand moved significantly since the frame in which the object was grabbed.

We find timestamps of object grabbing by assuming that this action requires that the hand stops for at least an instant. Thus we look for such changes in the speed of the hand which we measure by optical flow. For optical flow estimation we use FlowNet2.0 [3]. If the magnitude of average velocity is at a

local minima, then the timestamp is selected as a candidate for change detection. The local minimas are found by using median filter on the velocity magnitude signal and then using a peak finding algorithm [2]. An example of this can be seen on Figure 1. For each of the candidate position we assume that the appearing/disappearing object is occluded by the hand in the instant of releasing/grabbing and we select a rectangular region of interest (RoI) on the image around the center of the hand. Then we search for frames backward and forward on which the hand moved outside of our selected RoI, so the object is not occluded.

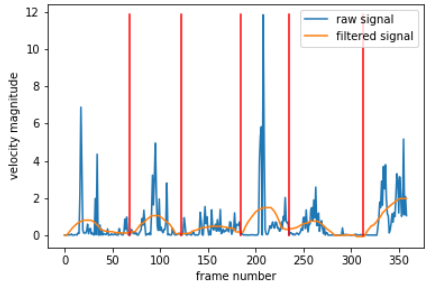


Figure 1: Hand movement segmentation. The plot shows the raw velocity magnitude filtered by median filter. The red lines are the local minimas found by the peak detector. Those timestamps will be investigated further in the next steps.

Interference removal

We found that the selected frames and RoI given by the first step contains other differences than the object we were looking for. These differences come from different sources: effect of actors interfering in the RoI (e.g. body parts, shadow, other manipulated object). To neutralize these effects we create a binary mask on the RoI that neglects pixels that belongs to these phenomena. In fact we estimate multiple binary masks with different strategies and and take the intersection of the relevant pixels found by each method. These are the following:

- Hand occlusion: We used Mask-RCNN [1] for filtering out hand-occluded pixels
- Forward and backward optical flow between I_{t_1} and I_{t_2}
- Optical flow between I_{t_1} and I_{t_1-1}, and also between I_{t_2} and I_{t_2-1}

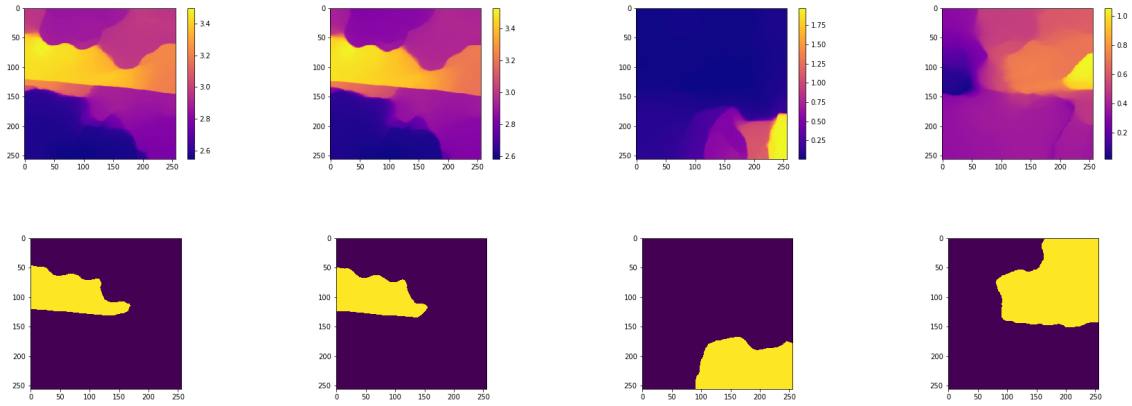
The forward and backward optical flow between I_{t_1} and I_{t_2} accounts for changes that happened over a longer period of time (e.g. edge of the paper moved on which the object was placed). The optical flow between a frame at time instant t and $t-1$ helps in removing any on-going activities in the RoI (e.g. hand is still there but moving). This could be done for any t and $t-k$ time instants, however we found $k=1$ to be sufficient. Optical flow based masks are obtained by thresholding the flow magnitude in each flow field.

Image difference

The final result is obtained by applying the binary masks on the two selected images I_{t_1} and I_{t_2} then taking their difference as follows[4]:

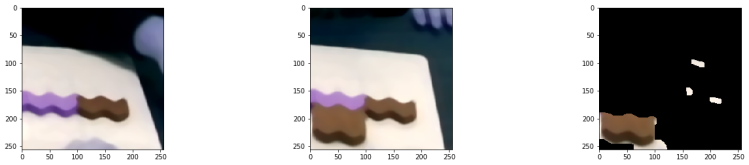
$$I_{final}(x,y) = \|I_{t_1}(x,y) - (\frac{\sigma_1}{\sigma_2}(I_{t_2}(x,y) - \mu_2) + \mu_1)\|$$

where σ_1, μ_1 and σ_2, μ_2 are the mean and standard deviation of I_{t_1} and I_{t_2} respectively. We then threshold I_{final} to detect appearance or disappearance.



(a) Optical flow between I_{t_1} and I_{t_2} (b) Optical flow between I_{t_2} and I_{t_1} (c) Optical flow between I_{t_1} and I_{t_1-1} (d) Optical flow between I_{t_2} and I_{t_2-1}

Figure 2: Optical flow magnitude (upper) and their thresholded binary masks (lower).



(a) Selected image I_{t_1} (b) Selected image I_{t_2} (c) Final mask applied to I_{t_2}

Figure 3: An example from our evaluations. The selected frames I_{t_1} and I_{t_2} during image selection and the objects that appeared between the two time instants as found by our method.

Detector left out	**Precision**	**Recall**
Optical flow between I_{t_1}, I_{t_2}	0.76	0.47
Optical flow between I_{t_1}, I_{t_1-1} and I_{t_2}, I_{t_2-1}	0.71	0.42
Mask-RCNN	0.72	0.52
All detectors active	**0.82**	**0.55**

Table 1: Results of leave-one-out experiments for the different detectors.

Results

We evaluated our method on a dataset provided by Argus Cognitive Inc. It containts sessions where two people are interacting over a table with a static camera above them. We labeled time intervals when the object was grabbed but did not start to move. Finding one point in this interval counts as a true positive detection. We suppressed multiple detections by filtering out points that are closer than the length of the shortest interval in our dataset. The algorithm achieved 82% precision with 55% recall. We did leave-one-out experiments to determine the relevance of each detector based binary mask. Table 1 shows the result of this experiment.

Conclusion

We proposed a method for detecting appearing and disappearing objects without the use of training samples. First by describing the general driving principles of the process, then transforming them into concrete rules. We used high accuracy detectors for each rule which resulted in acceptable overall perfomance.

While this method has acceptable performance, many improvements can be made. The motion segmentation step greatly affects the performance, since it proposes the candidates for detection. Improving this step would be highly beneficial. Another area to investigate further is the treatment of hyperparameters. At almost every step of this method there are thresholding parameters which require careful tuning. These include the bounding box difference threshold in the forward-backward image search for candidate images, the optical flow thresholding parameters and the bounding box size for hand motion. While we can find generally good values for these, treating them in a probabilistic fashion would be of great importance. We could then measure the uncertainty in both the parameters and the detection. This latter topic will be explored in the future.

Acknowledgements

We couldn't have done this work without the great advices from our supervisor Dr. habil András Lőrinc and also from Zoltán Tősér. We would also like to thank Dr. Erzsébet Csuhaj Varjú as professional leader. The project has been supported by the European Union, co-financed by the European Social Fund EFOP-3.6.3-16-2017-00002.

References

[1] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017, October). Mask r-cnn. In Computer Vision (ICCV), 2017 IEEE International Conference on (pp. 2980-2988). IEEE.

[2] Du, P., Kibbe, W. A., & Lin, S. M. (2006). Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching. Bioinformatics, 22(17), 2059-2065.

[3] Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., & Brox, T. (2017, July). Flownet 2.0: Evolution of optical flow estimation with deep networks. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Vol. 2).

[4] İlsever, M., & Ünsalan, C. (2012). Pixel-based change detection methods. In Two-Dimensional Change Detection Methods (pp. 7-21). Springer, London.