Drinking Detection From Videos in a Home Environment

Carlo M. De Masi
carlo.maria.demasi@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

Mitja Luštrek
mitja.lustrek@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

ABSTRACT

We present a pipeline developed with the aim of helping people with mild cognitive impairment (MCI) in the accomplishment of every-day tasks. Our system adopts a number of computer vision methods to analyze RGB videos collected from cameras, and provides a successful, quasi real-time detection of the targeted activity (drinking) when the latter is at least partially visible to the camera.

KEYWORDS

computer vision, activity recognition, object detection, pose estimation

1 INTRODUCTION

Mild cognitive impairment (MCI) is a common problem among elders, affecting 15–20% of people over 65 in the USA [10]. In order to help people affected by MCI in the accomplishment of every-day tasks, we adopt various kind of detection techniques to predict what users are currently doing, which, combined with a knowledge of their activities schedule, allows our system to provide context-based reminders. Here, we present our attempts to detect one of such activities (i.e. drinking) from videos, by the use of computer vision and deep learning algorithms.

This paper is organized as follows. In the remainder of this section, we give an overview of the current SOTA regarding activity recognition from videos. In Section 2 we describe the computer vision techniques used to trigger the more computationally intensive task of activiry recognition, to obtain a quasi real-time monitoring of the user's activities. Finally, in Sections 3 and 4 we present the results and conclusions of the paper.

1.1 Video Activity Recognition

Differently than what happened for image classification, where in the last years a number of clear front runner architectures and techniques have been established, the topic of activity recognition from videos still presents numerous open issues [1].

An immediate approach to the problem consists in using image classification networks to extract features from each frame of the video; then, predictions for the whole video can either be obtained by pooling over frames (at the cost of losing information about temporal ordering) [5] , or by adopting LSTM layers [2].

A more elaborate way to adapt the concepts used in image classification methods to video recognition consists in using 3DCNN, i.e. convolutional models characterized by an additional third temporal dimension [4, 12, 13]. The increased number of parameters makes 3DCNNs generally harder to train than their 2D counterparts. One way to fix this is to produce 3D models by "inflating" 2D ones, i.e. by adding a temporal dimension to a model pre-trained for image classification. This allows to determine the architecture of the 3D network and to bootstrap its values starting from the corresponding values in the 2D model: convolutional kernels with dimensions $N \times N$ are inflated to a 3D kernel with dimensions $N \times N \times t$, spanning t frames, and each of the t planes in the $N \times N \times t$ kernel is initialized by the pre-trained $N \times N$ weights rescaled by 1/t [1, 9].

Another approach separately analyzes spatial components (i.e. single frames), providing static information about scenes and objects in the picture, and temporal components related to motion and variation between frames [11]. A two-stream network parallelly processes single frames and optical flows, respectively, and then combines their predictions.

Finally, another method worth mentioning is based on the observation that some actions (i.e., clapping hands) are better characterized by high-frequency temporal features, whereas other ones (i.e., dancing) can be better understood when lower frequency variations are observed. As a result, a model characterized by two parallel channels can be used. The first (slow) channel operates at low framerate and analyzes few sparse frames, in order to deduce the semantics of the action, while the second (fast) branch is responsible for capturing fast variations, and so operates at higher framerate [3].

In this work, we adopted a modified version of an inflated 3D network as described in [14], to include non-local blocks. Unlike convolutional and recurrent operations, which are only able to capture spatio-temporal features in a local neighborhood, non-local blocks compute the response at a certain position as a weighted sum of features at all positions in space and time. This allows the model to capture dependencies between pixels that are distant both in space and time, and makes it more accurate for video classification.

2 SYSTEM ARCHITECTURE

The purpose of our system is to provide users context-based reminders related to the activity of drinking. To this aim, a RGB camera is placed in the kitchen of the user's apartment (where the activity is most likely to take place) and the video is sent through a RTSP stream to a remote server, to be analyzed by the activity recognition model during the day. The results are uploaded to a Cloud Firestore Database, which is queried to determine whether the users have been drinking enough, and reminders are provided through an app running on a local device if not.

One problem arising from this scheme is that most action recognition algorithms are computationally expensive, which prevents them from running in real time. For this reason, we decided not to run the model continuously, but to execute it only in moments where it is most likely that the users are about to perform the targeted activity. We employed a combination of

Information Society 2020, 5–9 October, 2020, Ljubljana, Slovenia

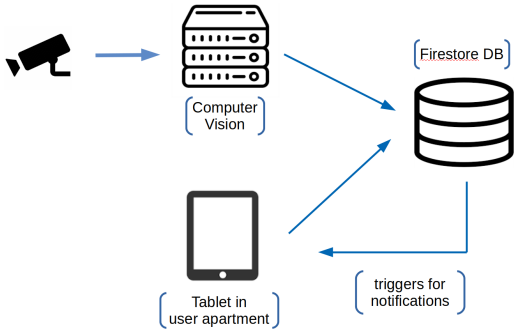

Figure 1: System architecture. Video stream from RGB cameras is sent to a remote server and fed to the activity recognition model. Results are uploaded to a Firestore database, where they are monitored so that notifications can be sent back to an app.

classic and deep-learning-based computer vision techniques to identify some ***triggers*** for the video activity recognition model, such as: (i) user standing in certain areas of the kitchen; (ii) user standing in certain areas of the kitchen, and interacting with some objects (tap, fridge); (iii) a specific object, assumed to be used by the user for drinking, is moved from its current position.

2.1 User Localization And Interaction With the Environment

The localization of the user and their interactions with the environment are detected through a combination of object detection and pose estimation techniques. For the object detection, we adopted a Single Shot MultiBox Detector (SSD) [8], pre-trained on the 80 classes of the COCO dataset [7], which also include "person". As for pose estimation, we used a SimpleNet model with a ResNet backbone [15].

During the initial setup, the camera image is shown to the user (Fig. 2a) and regions of interest (ROIs) can be selected (Fig. 2b). These can be of two types, i.e. single or double-zone. The first ones are identified by a single rectangular box, which is activated when the user's feet are within the box, hence providing indications on the user's location (see Fig 2c). Double-zone ROIs are formed by two rectangular boxes; one of them, analogously to the previous case, is activated when the user steps inside of it, while the second box is activated if one of the user's hands (located by the pose estimation model) is within it (Fig. 2d). Overall, a double-zone ROI is considered activated only if both conditions are met. Once the ROI is configured, the user is requested to input:

- the name used to identify the current ROI;
- an observation time t_{obs} (in seconds), i.e. the time after which the ROI is activated, once the requirements (user and hands positions) are met;
- an action to be performed once the ROI is activated. Currently, only one default action - recording and analyzing video clips - is supported, but this will be extended to include further possibilities.

2.2 Drinking Vessel Position Detection

A second trigger for activity recognition is given by the displacement of a particular object (mug, cup, glass). To this regard, in the pilot phase of the project users will be asked to always use one specific drinking vessel when they are drinking, which the model will be trained to recognize.
For this task, we considered two possibilities:

- a classic computer vision approach, where the drinking vessel is located through a color/shape-based detection;
- a deep learning object detection algorithm, re-trained to detect a personalized mug.

In the fist scenario, we applied a series of filters (Gaussian blur, dilation/erosion) to reduce noise, followed by a color mask in the HSV space to select only objects with a certain color. A further selection is then done based on the shape properties of the previously selected areas; a polygonal approximation of their contours is performed, and other shape-related features such as area, circularity and convexity are considered to eliminate shapes different from the expected one.

In the second case, we collected a dataset of about 500 images of the selected mug, and used it to re-train a second SSD model. In order to account for false negatives in the mug detection, that may occur in some frames even if the mug has not been moved, for each frame the current position of the mug is compared to the history of positions in the past few frames. Once a displacement of the mug is detected, the trigger is activated.

2.3 Clip Recording and Activity Recognition

Following the activation of one of the triggers, the next video frames (for a time interval of about 30 seconds) are used to generate short video clips, each of which has a duration of 10 seconds, with an overlapping window of 4 seconds. These values have been selected to have a higher probability to obtain at least one video clip completely capturing the whole drinking process, and to match the length of the videos in the Kinetics400 dataset [6], which has been used for the activity-recognition model training.

3 RESULTS AND DISCUSSION

In this section, we present the results of the various steps involved in the whole drinking-detection pipeline.

3.1 User Localization - Results

We tested the efficiency of the localization module in different scenarios, varying based on how clearly the user was visible (completely visible; legs occluded; head occluded; head and legs occluded, only torso visible) and on which side (front/back/right/left) of the user was visible, and the results showed an average accuracy of over 98%.

3.2 Drinking Vessel Position Detection - Results

As illustrated in Sec. 2.2, for the task of detecting the displacement of the drinking vessel we adopted two approaches, one based on classic computer vision methods and one on deep learning.

The first method does not provide a confidence score for detections, nor the coordinates of the object's bounding box, so we took a simpler approach than with normal object detection algorithms in evaluating the results. We collected some videos in a home-like environment, with the object located in different positions, or with a person handling it (moving it, using it to drink...), and analyzed them frame-by-frame to check whether the objects present in each frame were detected or not. The resulting confusion matrix, reported in Table 1, shows that the detection algorithm scored precision and recall values of .93 and .90, respectively. This method proved to be very efficient, when correctly fine-tuned, and the algorithm detected the object in most of the frames where it was at least partially visible. The

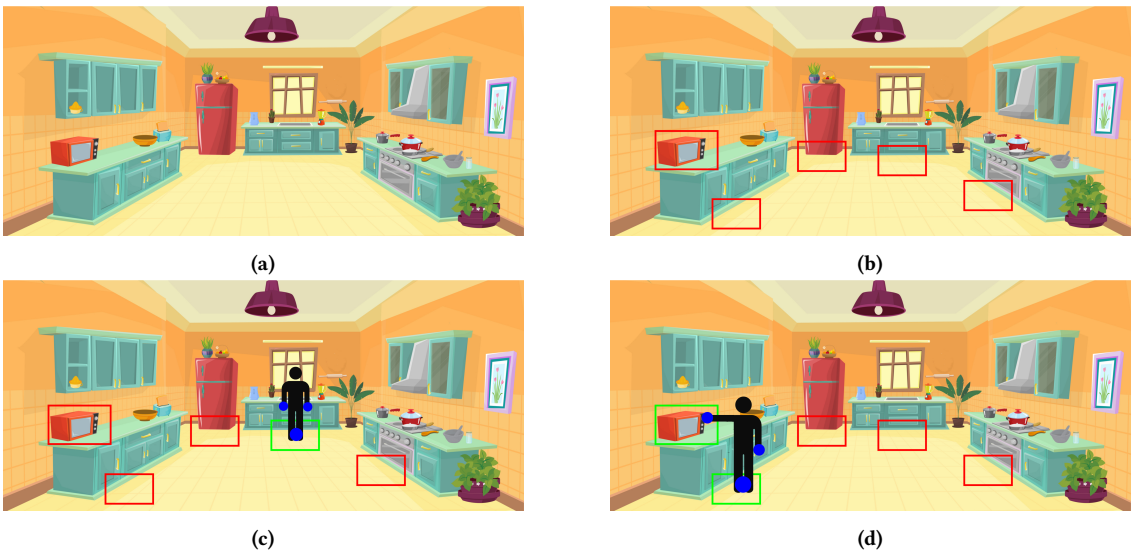

Figure 2: **Triggers based on user's location and their interaction with the environment. Regions of Interest are selected during the setup phase (b), and they are activated either if the user steps inside (c), or if the user steps inside and has their hands next to another object (d).**

Table 1: Confusion matrix for the color/shape-based detection of the mug

		Pred	
		P	N
True	P	133	15
	N	10	1

greatest issue of the method is that it had to be very carefully tuned, especially regarding the color selection part, which is still sensible to lightning variations even after converting the image to the HSV colorspace. False detection can also be a problem. We tested the algorithm in situations where some of the objects present in the scene had colors similar to the object we wanted to detect, and in spite of being able to filter out most of them we still obtained some false positives, especially when the lighting varied, thus rendering the selection of the parameters for the color mask less efficient.

The results of the evaluation of the SSD model are shown in Fig. 3. As evident from the plot, the model immediately reached a very high mAP [7], of the order ≈ 0.9, on our test dataset. It should be noted that, while preparing the training dataset, we followed a somewhat different approach than what is usually done for training object-detection models. In most situations, one wants to make the model as general as possible and avoid overfitting, which is achieved by taking images of the desired object in as many different conditions (size, aspect ratio, point of view angle, rotation, lightning) as possible. In our case, however, the location of the camera will be more or less constant, i.e. attached to the ceiling of the room, in order to provide a good view of the environment. As a result, this will greatly limit the variability in the images of the object the system will analyze, especially regarding the aspect ratio and the orientation of the mug. Moreover, whereas an object detector is usually tasked to identify many different instances of objects in a certain class (i.e., a generic "mug"), in our case the task is greatly simplified by the fact that we are looking to locate one very specific object.

3.3 Activity Recognition - Results

We tested the adopted activity recognition model on a new custom dataset, consisting of roughly 100 videos we recorded ourselves in a variety of environments and conditions. In order to make the clips as similar as possible to real-life situations, the videos contained instances where actions similar to drinking were performed, to increase the recognition difficulty. The clips can be classified as belonging to two difficulty categories, based on the angle the user was facing with respect to the camera; videos were classified as "hard" whenever this angle was greater than 90° (see Fig. 4). The precision-recall curve for the model on this dataset is shown in Fig. 5.

4 CONCLUSIONS

The tests performed on triggers are very encouraging for the one based on the user location and their interaction, and indicate that the deep-learning approach should be preferable for the detection of the drinking vessel and its displacement, especially after increasing the amount of training data. The activity-recognition model based on inflated 3D CNN with the addition of non-local blocks provided the best accuracy in situations were the user is facing the camera at least partially, and the use of triggers allows for a quasi real time usage. A number of improvements will be added to the pipeline in the future. Currently, only one action is triggered, i.e. recording and analysis of video clips, but we plan to include other possibilities, such as using the information on the user location to check whether they need assistance in operating domestic appliances. The object detection model could also be extended, in order to identify interactions with other elements of the environment, and provide corresponding context-based responses. Finally, the only action currently recognized is drinking, but as mentioned in the introduction the aim of the project is to assist users in the accomplishment of various activities. In this sense, the next planned step is to include detection of parts of the morning toilet routines, such as brushing teeth and washing hands.

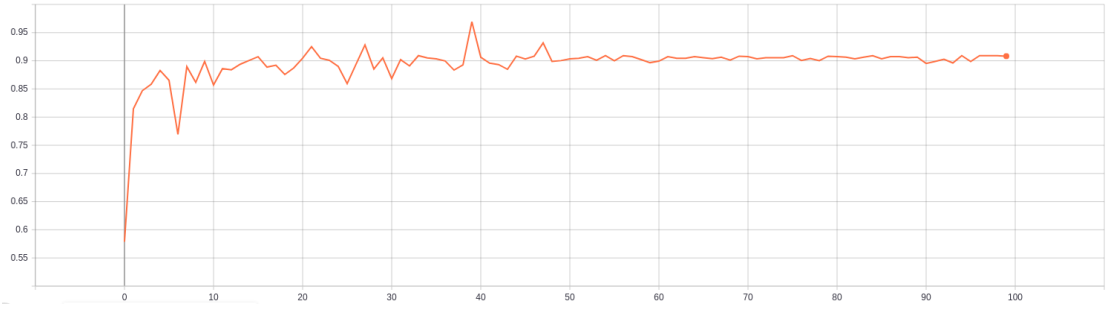

Figure 3: mAP values on the test dataset for the SSD model, re-trained to recognize the project custom mug.

Figure 4: Difficulty classes for the custom dataset we used to test the activity recognition model. Video clips were classified as "hard" whenever the angle between the user front side and the camera was greater than 90°.

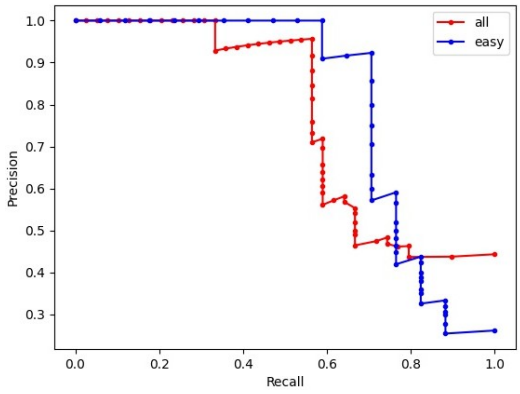

Figure 5: Test results of the activity recognition model on the test dataset.

REFERENCES

[1] Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6299–6308.

[2] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, et al. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2625–2634.

[3] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, et al. 2019. Slowfast networks for video recognition. In *Proceedings of the IEEE international conference on computer vision*, 6202–6211.

[4] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 2012. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35, 1, 221–231.

[5] Andrej Karpathy, George Toderici, Sanketh Shetty, et al. 2014. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 1725–1732.

[6] Will Kay, Joao Carreira, Karen Simonyan, et al. 2017. The kinetics human action video dataset. (2017). arXiv: 1705.06950 [cs.CV].

[7] Tsung-Yi Lin, Michael Maire, Serge Belongie, et al. 2014. Microsoft coco: common objects in context. (2014). arXiv: 1405.0312 [cs.CV].

[8] Wei Liu, Dragomir Anguelov, Dumitru Erhan, et al. 2016. Ssd: single shot multibox detector. *Lecture Notes in Computer Science*, 21–37. ISSN: 1611-3349. DOI: 10.1007/978-3-319-46448-0_2. http://dx.doi.org/10.1007/978-3-319-46448-0_2.

[9] Elman Mansimov, Nitish Srivastava, and Ruslan Salakhutdinov. 2015. Initialization strategies of spatio-temporal convolutional neural networks. *arXiv preprint arXiv:1503.07274*.

[10] Ronald C Petersen, Oscar Lopez, Melissa J Armstrong, et al. 2018. Practice guideline update summary: mild cognitive impairment: report of the guideline development, dissemination, and implementation subcommittee of the american academy of neurology. *Neurology*, 90, 3, 126–135.

[11] Karen Simonyan and Andrew Zisserman. 2014. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, 568–576.

[12] Du Tran, Lubomir Bourdev, Rob Fergus, et al. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 4489–4497.

[13] Gül Varol, Ivan Laptev, and Cordelia Schmid. 2017. Long-term temporal convolutions for action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40, 6, 1510–1517.

[14] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. 2018. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7794–7803.

[15] Bin Xiao, Haiping Wu, and Yichen Wei. 2018. Simple baselines for human pose estimation and tracking. In *Proceedings of the European conference on computer vision (ECCV)*, 466–481.