Using object recognition benefits in a system for fire detection in the nature

Maria Pavlova[1]

Abstract – **The subject of fire recognition is quite popular and many studies discus it. This paper presents a research on a fire recognition field. A camera mounted on the unmanned aerial vehicles (UAV) records videos of the surrounding environment. The method of recognising fire captured on the videos, part of this research, aims to reduce time needed to detect fire. The method assisted by OpenCV library capabilities recognises fire event. The results are described in this paper.**

Keywords – **Object recognition, OpenCV, Fire recognition, Machine learning.**

I. Introduction

Fire in a building or in the nature is a problem of very high concern. To find a way to prevent a fire is a task of significant importance. For the matters of the indoor fires there are already different fire preventions systems. These systems proof to be highly efficient in buildings and another indoor facilities.

The problem of detecting fire in the outdoors is quite complex. An efficient fire detection requires more advanced technologies such as neuro networks, object recognition and, in addition, unmanned aerial vehicles (UAV).

This paper presents the results of a research in the area of fire recognition in inaccessible locations. For the purposes of the research is used UAV with a small lightweight digital camera mounted on it. The camera records video clips of the surrounding environment and then transfers them to a remote workstation, running fire recognition software.

Nowadays, different object recognition algorithms are available for any purpose. One of them is Machine learning with feature object recognition used in the current research. For the purpose of the research, Python wrapper is used to encapsulate the C/C++ code, thus increasing overall performance. Additional benefit is that the wrapper can be used in any Python module and OpenCV library as such, and OpenCV opens a large spectrum of possibilities in the Machine Learning.

Video cameras and computer vision methods are used for fire detection in chemical factories and other high fire risk industrial. The Deep machine learning will increase the accuracy and reduces the costs for high level of security [8].

This paper has four contributions with following contents: the first contribution is introduction; second contribution of this paper is the software implementation; the third is about results and the fourth contribution is the conclusion and the References. In each part of this paper the subject will be explained accordingly.

[1]Maria Pavlova is with the Faculty of Telecommunications at Technical University of Sofia, 8 Kl. Ohridski Blvd, Sofia 1000, Bulgaria, E-mail: mariapavlova@tu-sofia.bg.

II. Software description

Fire detection using infrared camera [1], [2] is a popular decision among the building's fire detection systems. There are also researches, using this same camera for outdoor uses.

The solution using infrared cameras is unquestionably reasonable, but it gives its best results in short distances. When talking about open spaces, that's almost never the case – often the areas of fire are inaccessible and thus the results are usually far from the necessary precision.

Therefor this research is aiming to offer a fire recognition method in the outdoors with a camera, capturing the surface. The camera is shown on fig. 1

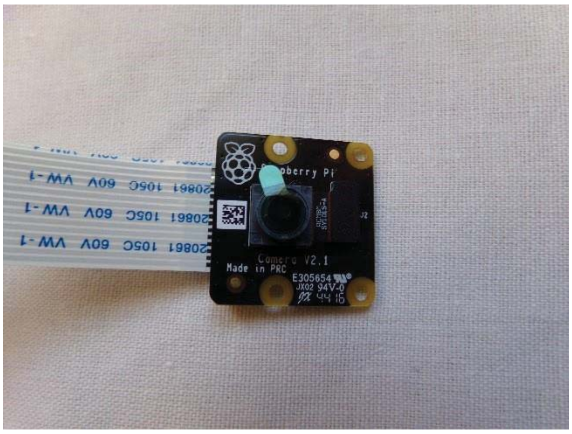

Fig. 1. Camera mounted on UAV

The application has been written and tested on Linux operation system. Captured video is processed by deep learning algorithm for detecting the fire in captured video.

There are many systems based on pixel object recognition like Pixel-to-pixel machine [4] and they have their application in their own fields. This approach is based on Paul Viola and Michael Jones algorithm for rapid and reliable object detection [3]. The algorithm which is proposed by Paul Viola is improved by Rainer Lienhart [5]. In this research the object detection classifier is based on features technique, which is faster than pixel system. Furthermore the complicated fire image (color, texture) will be recognized better with feature classifier. The algorithm allows for additional own classifiers to be created.

Basically the cascade of boosted classifiers works with haar-like features, from now on called classifiers. It is named cascade, because several simple classifiers (stages) are parts of the whole classifier. The word boosted means that every stage of classifier is a result of another. It is possible to use one of four supported techniques – Adaboost, Real Adaboost, Gentle Adaboost and Logitboos. Adaboost is the technique used in this research. As mentioned before, the classifier has more than one classifier inside. The haar-like features are inputted data of the main classifier and it is deeply explained in [6].

Each feature used in a separate classifier is determined by shape, position, scale. The feature are computed by subtracting the sums of pixels by “integral image” in respected areas under white and black rectangles.[WK1] Black and white rectangles areas, which the image is separated. In their article “Rapid Object Detection using a Boosted Cascade of Simple Features” authors described deeply their method [3]. They apply the results for face detection classifier.

Base on above mentioned and the possibilities to prepare own cascade classifier, this research present the own cascade classifier fire detection. The algorithm use for classifier training is shown in table 1.

TABLE I
ALGORITHM FOR PREPARING THE CASCADE CLASSIFIER

1	Collect the negative images
2	Collect or Create positive images
3	Create a positive vector file by stitching together all positives
4	Train cascade classifier

The process of creating the classifier and all related works are presented in author’s article “Object Recognition System Operating from Different Type Vehicles Using Raspberry and OpenCV”. As the aim of this paper is to present the results of this research, the algorithm steps will be mark shortly.

To create the data base of negatives images is important task and first should be done. The data base should be at least 1000 pictures. Important requirement for the pictures: do not contain the object of recognition. In this case the fire.

Fig. 2. Result from creation of positive vector and stitching together all positives

Second step of this algorithm is creation of the positive data base. It may create from one image or collection of more. OpenCV has a powerful module opencv_createsampels [7]. With this function in terminal under Linux the script is one row:

“opencv_createsamples -img fire5038.jpg -bg bg.txt -info info/info.lst -jpgoutput info -maxxangle 0.5 -maxyangle 0.5 -maxzangle 0.5 -num 1300”

The result is very interesting and is shown in fig.2. We can see that the positive image fire is covered over the background image. The positive object is marked with red circle on the shown figure.

After preparation is possible to do the last step to train the cascade classifier. About cascade classifier is explaned from Viola and Jones in “Rapid Object Detection using a Boosted Cascade of Sample Features” [10]. In OpenCV library this is one row of code, but behind this row has a lot of work for devising. Create simples first and after this is possible to train the stages. The number of stages is important. It depends of number of created samples. One towsend and two hundred positive samples will give opportunity around 10 stages. One stage contain the number of stage, the number of features in each stage and thresholds of each stage. On fig. 3 is shown stage 2 and stage 7 after cascade training.

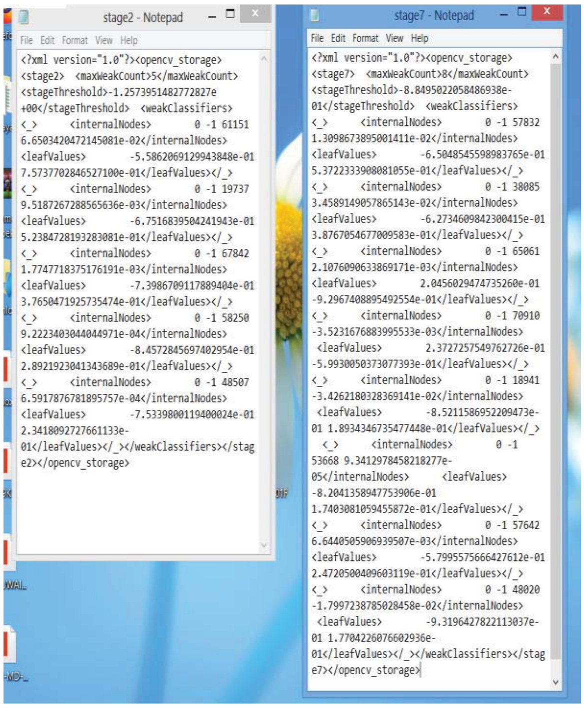

Fig. 3. Stage 2 and stage 7 after Cascade Training

III. RESULTS OF SOFTWARE IMPLEMENTATION

Finally the result of all this steps is the XML file from Classifier Training. This file is argument of function cv2.CascadeClassifier. Implementation of this function give result fire recognition. For this application this XML file is named cascade_training_fire.xml.

In additional one blue rectangle marks the recognized object. For this purpose is used the function cv2.rectangle(). Its arguments are video or image, size and colour of the outline.

On fig. 3 is shown the screenshot of one of the videos, which are used for test. Blue rectangle define very well the fire and the smoke.

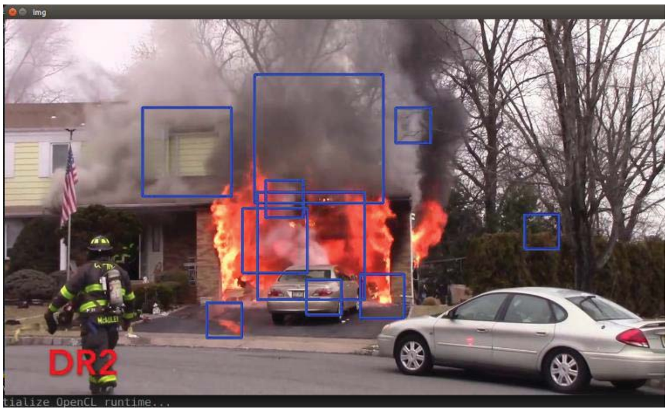

Fig. 3. Result from fire recognition

From another video we have also good results shown in fig. 4 and fig. 5.

Fig. 4. Positive result from fire recognition

On fig. 5 is shown very good result - the recognition of smoke. In this moment of video the classifier works really precise. That means that the training of machine for smoke gives very good results. From all tested pictures the smoke is recognized completely, with only one mistake: if there is the clouds in the picture the result will be wrong. It is shown in fig. 7. If we take into account the application area of the system, namely the observation of the ground surface, we may draw the conclusion that the program recognizes smoke in 98% of the cases.

Fig. 5. Positive result from fire recognition

To recognize the fire precisely is task with high importance, because of people security and prevents of forest fire. It means the accuracy should be very high. But unfortunately in the results we have object recognition different than fire.

On fig. 6 is shown one of this negative results.

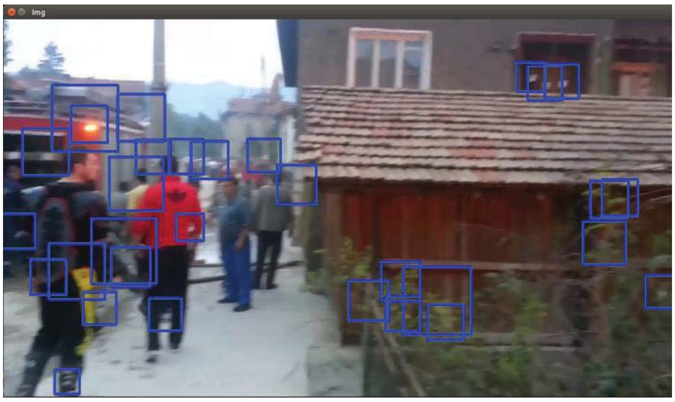

Fig. 6. Recognition of face instead of fire

The problem here is that the classifier recognize the face and the moving of all this people give the negative result. The problem with detection is serious and one of possible way to resolve is present in [9]. The authors have proposed a method, which extract the regions with movement. This regions are the potentials places with fire.

Second problem is that the machine recognise the clouds like smoke. This effect is shown in fig. 7.

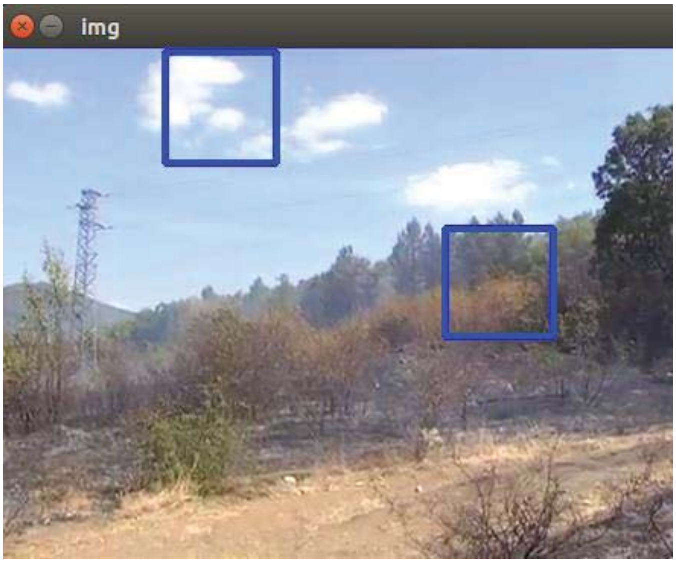

Fig. 7. Recognizes the clouds instead of smoke

There is one more problem on this picture – the blue rectangle on the right side surrounds leaves instead of fire.

The last problem that was registered is that the classifier recognized the car lights instead of fire and this bad result is shown on fig. 8.

Fig. 8. Recognizes the car lights instead of fire

IV. Conclusions

The results of this research show that 40 % of all tested videos give bad results in fire recognition. In the remaining 60% of test videos the classifier recognizes the fire very well. It's clear that the described approach is necessary to be improved. This will be the next aim in this research.

References

[1] https://doi.org/10.3182/20050703-6-CZ-1902.01380

[2] https://www.tandfonline.com/doi/abs/10.1080/014311699212290

[3] P. Viola, M. Jones "Rapid object detection using a Boosted Cascade of Simple Features", ACCVPR, 2001 https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/viola-cvpr-01.pdf

[4] Keysers D., Deselaers T., Ney H. (2004) Pixel-to-Pixel Matching for Image Recognition Using Hungarian Graph Matching. In: Rasmussen C.E., Bülthoff H.H., Schölkopf B., Giese M.A. (eds) Pattern Recognition. DAGM 2004. Lecture Notes in Computer Science, vol 3175. Springer, Berlin, Heidelberg

[5] Rainer Lienhart and Jochen Maydt. An Extended Set of Haar-like Features for Rapid Object Detection. IEEE ICIP 2002, Vol. 1, pp. 900-903, Sep. 2002. This paper, as well as the extended technical report, can be retrieved at http://www.multimedia-computing.de/mediawiki//images/5/52/MRL-TR-May02-revised-Dec02.pdf

[6] https://docs.opencv.org/2.4/modules/objdetect/doc/cascade_classification.html?highlight=cascadeclassifier#lienhart02

[7] https://pythonprogramming.net/haar-cascade-object-detection-python-opencv-tutorial/, 2017

[8] Hao Wu, Deyang Wu, Jinsong Zhao, An intelligent fire detection approach through cameras based on computer vision methods, Process Safety and Environmental Protection, Volume 127, 2019, Pages 245-256,ISSN 0957-5820, https://doi.org/10.1016/j.psep.2019.05.016

[9] Mahdi Hashemzadeh, Alireza Zademehdi, "Fire detection for video surveillance applications using ICA K-medoids-based color model and efficient spatio-temporal visual features, Expert Systems with Applications", Volume 130, 2019, Pages 60-78, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2019.04.019.

[10] Paul Viola, Michael Jones "Rapid Object Detection using a Boosted Cascade of Simple Features", ACCEPTED CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION 2001

.