Automated Intelligent UAS-based Surveillance System for Urban Security Needs

Antti Perttula, Pekka Pöyry, Esa Kujansuu
Tampere University of Applied Sciences
Tampere, Finland
antti.perttula, pekka.poyry, esa.kujansuu@tuni.fi

Abstract— **In order to improve the usage of data from UAV mounted surveillance cameras we created an autonomous intelligent picture analyzing and forwarding system. UAV sends automatically real time pictures down to server to be analyzed. After analyzing the pictures can be be shared with relevant recipients like Police and Search & Rescue. We compared several analyzing algorithms in terms of accuracy, speed, and overall functionality in our environment. We found that Yolo v4 was the most suitable compromise. We were able to demonstrate that after training the algorithm with over thousand pictures system achieves over 90% accuracy in categorizing and counting objects right.**

I. INTRODUCTION

The aim of City of Tampere 's SURE (Smart urban security and event resilience) project is to improve safety and security of citizens' daily lives [1]. Several cameras located in the city center form the foundation of security information gathering system of the SURE project. Despite of large number of cameras there isn't full coverage of imaging in all parts of the city at all time. Because of this we made it possible to enhance the network of fixed camera installation by cameras installed on UAVs. In SURE project one of the main focuses is to understand possible risk factors where people and different kinds of vehicles are involved. Thus, also our focus was on counting numbers of interesting objects: person, car, bus, van, and truck with indication of their locations.

II. IMAGES FROM UAV

UAV's imaging doesn't have the limitations what fixed cameras installation have but they can fly and take pictures virtually anywhere. Normal way to apply pictures taken by a UAV is to store them first in its memory while device is flying still and just later, after flight, to download them to server for further processing. However, this is not a vital option in security monitoring applications because we need to get real time understanding what is happening. One of the UAVs we used in this study was DJI Mavic Air as shown on the Fig. 1.

The Lightbrigde technology of drones makes it possible with a smart phone to see live video from the UAV and store video and still pictures directly from UAV. DJI GO 4 is an application which supports both Android and iOS operating systems. With this application we can control UAV, change its settings, and see real time video from UAV's camera. Fig. 2 shows DJI GO 4 image when UAV is flying.

Fig. 1. DJI Mavic Air with the controller [2]

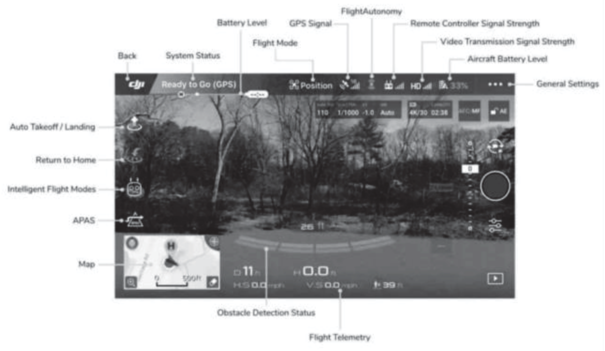

Fig. 2. DJI GO 4 image when UAV is flying [2]

Android is an open source mobile platform and one of the most popular operation systems developed for devices with touch screen. Applications can be programmed using Java and Kotlin. FileObserver is internal function of Android, which monitors and manages changes of files [3]. These changes can trigger desired operations in the code like sending a picture to server. To enable real-time pictures sharing, we made an Android-based application and demonstrated the functionality with DJI's Mavic Air UAV.

III. INTELLIGENT IMAGE ANALYZING

With the new image obtaining solutions, which have plenty of high-resolution cameras, it is easy to get access to huge amount of imaging data. However, data itself has only little value if not processed proper way. Luckily, there are several intelligent processing techniques for image raw data resulting easy use in different applications. These include common face recognition applications of smart phones and road sign detection for vehicles.

We found 41 suitable object detection models of which 17 were selected for evaluation. Test data consisted of over 1000 pictures taken by drones. In these photographs 1754 objects were annotated. Results of object detection tools and actual annotated results were compared. We found that in terms of accuracy the best three tools were EfficientDet D7 1536x1536, EfficientDet D6 1280x1280 and EfficientDet D5 1280x1280. However, as we tried to create as fast system as possible, image processing speed is also extremely important. The fastest models for our purposes were CenterNet Resnet50 V1 FPN Keypoints 512x512, CenterNet Resnet50 V2 Keypoints 512x512 and Faster RCNN ResNet101 V1 800x1333 [4]. Because we wanted to optimize both accuracy and speed in our own environment none of the previous ones were very suitable. Thus, we selected another one better optimized for our own use, Yolo v4.

The most important and time-consuming task before taking the selected model in use is to train it with real pictures taken in actual environment. There are vast number of sample pictures available for training purposes. However, most of them were not usable for our purpose because of different scaling and altitude of camera. In addition, the backgrounds are usually different than what we have in Finland. Also, there are not too many pictures available taken in wintertime in poor light conditions. Fig. 3 shows typical sample pictures from Common Objects in Context (COCO).

Fig. 3. Sample pictures from COCO [5]

We were not able to apply existing sample pictures but had to take own training pictures from city of Tampere using drones. We trained the algorithm with over thousand pictures during winter and summer months. We annotated the interesting objects on pictures for the algorithm. Fig. 4 shows the numbers of annotated classes.

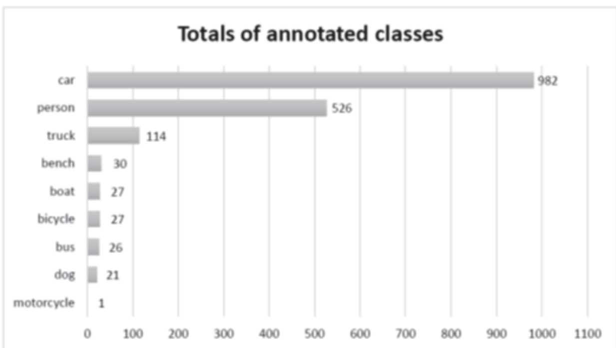

Fig. 4. Total annotated classes [4]

Fig. 5 illustrates examples of performance after the algorithm has been trained. In practice, all categorized objects in different weather conditions were found correctly on the pictures.

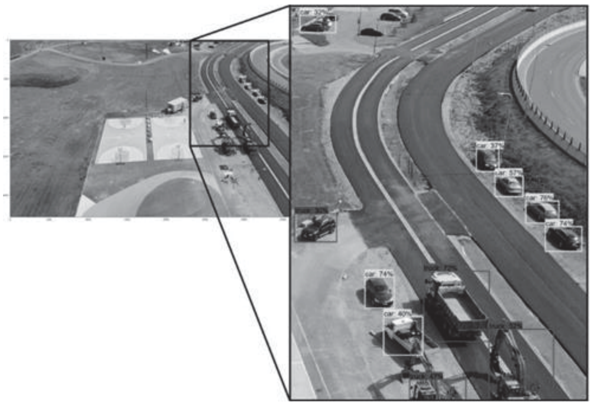

Fig. 5. The algorithm has found all wanted objects on picture taken by a UAV [4]

Fig. 6 shows a picture taken at the university campus area and it shows great performance also in winter conditions.

Fig. 6. Good performance in wither conditions [4].

Table I lists the performance results of the model after training, which takes about 30 hours on a dedicated server. After training the algorithm can analyze any new pictures in terms of seconds.

TABLE I. PERFORMANCE RESULTS [4]

person, AP = 93.64% (TP = 208, FP = 33)

car, AP = 96.53% (TP = 1389, FP = 219)

bus, AP = 100.00% (TP = 20, FP = 2)

van, AP = 93.34% (TP = 192, FP = 72)

truck, AP = 90.01% (TP = 48, FP = 13)

AP (average precision), TP (true positive), FP (false positive)

IV. Conclusion

We created a working system which autonomously take, forward, and analyze pictures taken by a UAV-mounted camera. The Android application sends real time image from a drone to server where another algorithm looks for five categories in the picture: people, cars, vans, buses, and trucks.

After analyzing the picture and highlighting the objects it can be sent to another platform for further use. Training the algorithm is very time consuming. However, after proper training analyzing a new image happens autonomously in seconds. Algorithm has very good accuracy varying between 90% (truck) to 100% (bus).

References

[1] Smart Urban Security and Event Resilience - SURE official website, https://suretampere.fi/events/conferences/publishing/templates.html, cited on Oct 22.2021.

[2] V. Helin, *Sending Images from a Drone to a Server*. Tampere, Tampere University of Applied Sciences, 2021. cited on Oct 22.2021.

[3] Android Developers pages, https://developer.android.com/reference/-android/os/FileObserver, cited on Oct 22.2021.

[4] *G. Laitila, Evaluation of pre-trained object detection models for the use in the SURE project.* Tampere, Tampere University of Applied Sciences, 2021.

[5] COCO official website, https://cocodataset.org/#home, cited 0n Oct 22.2021.