

UAV Management System for the Smart City

Peter Šulaj¹, Renát Haluška², Luboš Ovseník³, Stanislav Marchevský⁴, Petri Pulli⁵, Vadim Kramar⁶

^{1,2,3,4}*Dept. of Electronics and Multimedia Communications, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Slovak Republic*

^{5,6}*Dept. of Information Processing Science, Faculty of Information Technology and Electrical Engineering, University of Oulu, Oulu, Finland.*

⁶*Information Technology Department, Oulu University of Applied Sciences, Oulu, Finland*

¹Peter.Sulaj@tuke.sk; ²Renat.Haluska@tuke.sk; ³Lubos.Ovsenik@tuke.sk;

⁴Stanislav.Marchevsky@tuke.sk; ⁵Petri.Pulli@oulu.fi; ⁶Vadim.Kramar@oamk.fi

Abstract— In this paper a collection of technical solutions and results of simulation, supporting the concept of management a small group of unmanned aerial vehicles such as drones, are presented. The focus is given on collision avoiding and communications. The anti-collision system utilises the drone camera and is built on the Robot Operating System using the Histogram of Oriented Gradients and the Support Vector Machine. The proposed inter-drone communication employs ZigBee for system management and telecommunication networks for data exchange and teleoperations.

Keywords— *anti-collision, drone, simulation, telecommunication, uav, zigbee*

I. INTRODUCTION

In recent years, unmanned aerial vehicles (UAVs) became to be an essential mean to improve the quality of human life. The application area of UAVs ranges far beyond recreation. Consumer- and industry-grade UAVs are used for monitoring and surveying, science and discovery, sensing and warning, as news reporting tools, in business and marketing, farming and agriculture, for emergency services and healthcare, security, military, education, gaming and sporting, delivery, and many other.

Due to mobility, range of sensors and communication means, UAVs, particularly drones, appeared in a large variety of smart city cases and scenarios. Environment sensing and construction scanning, observations and monitoring, first responder activities, emergency and security are just some of those. Still, urban environment brings certain challenges to drone operations. Particularly, at low altitude, challenges are relevant to complex building and engineering infrastructures, a big quantity of moving objects, dense geo-fencing, and blind spots of communications.

By increasing a quality of drone's implementation, it is possible to achieve reasonably reliable operations of a drone. But in many cases, it is beneficial to operate a group of drones. That is not only conditioned by a reliability demand. A group of drones may be more responsive in a dynamically-changing environment, ensure a better sensing/monitoring experience, and be able avoid communication problems by forming a communication chain.

In the work presented in this paper, operations of a small group of drones was considered as main approach

for operations in smart city. Mini computers such as Raspberry Pi have sufficient computing power for required computational intelligence. Robot Operating System (ROS) is used in multi-robot environment. The development of UAV management system is still at a very early stage. Currently, the focus is given on collision-avoiding system, inter-drone, and drone-ground communications. The collision-avoiding system was built using the Histogram of Oriented Gradients (*HOG*) and the Support Vector Machine (*SVM*). Inter-drone communications were developed based on ZigBee technologies, and drone-ground communications were considered to be telecommunication-based and implemented using 4G LTE.

The rest of this paper has the following structure. An overview of the concept of UAV group management is presented in Section II. Details about the anti-collision system are given in Section III. In Section IV, simulation of network communications are described. A short overview of future work is given in Section V. In Conclusion, a summary of the paper is given.

II. THE CONCEPT OF UAV GROUP MANAGEMENT

The proposed concept implies a group of five UAVs communicating with each other using ZigBee technologies and utilising telecommunication networks for ground communications. Using Zigbee communications, UAVs share location information, battery condition, maintenance data and control commands. 4G telecommunication networks are currently considered due to their availability. Nevertheless, the future mobile networks such as 5G and 6G are better suitable for low altitude UAVs due to their higher throughput and lower latency. Those properties are important for better experiences with ground communications. The last includes bi-directional image exchange, video and audio streaming, Traffic Management (UTM) data, including geo-fencing, and other command and control data. ZigBee is suitable for a big number of applications, including industrial monitoring and control, family, and building automation, sensor networks, medical and automotive programs. ZigBee wireless technology is built on the IEEE 802.15.4 standard, designed to provide economic, standardised, and

flexible wireless network to provide low-power, low to moderate data rate control and monitoring applications, reliability, work together, and security support. The ZigBee Alliance also provides interoperability and conformance testing specifications. The ZigBee network layer is responsible for device discovery and network configuration, and supports three network topologies (star, tree and mesh) [1]. ZigBee is utilised for inter-drone communications, such as exchange of location data and control commands.

The following scenario is possible using the proposed concept. In the case of a traffic accident inside a tunnel where is a weak 4G signal, the group of drones departures as first responders. Before the police arrive, the drones discover the accident place by flying around the site and sending to the controls station a bunch of images out of which it is possible to build a 3d model of the site for future investigation. The drone group operator can see the accident site and inform the rescue team in advance of the situation. In case of any other sensors available in the tunnel, the involved in the accident vehicles, or people, drones will be able to get data within their access rights and deliver it to the operator and the rescue team. That kind of data may include medical information of people, such as possible allergies, blood group, age, and other information, type of vehicle propulsion (gasoline, diesel, electric, LPG, CNG, hydrogen), status of damage, speed before the accident, and many other.

III. ANTI COLLISION SYSTEM ON DRONES

The purpose of the anti-collision system is to prevent collision with another UAV or near-located mobile or fixed living or artificial object in order to prevent damage of those and/or self. Preventing a collision with a piloted vehicle may be ensured by dividing airspace, but it is not always achievable. For example, an emergency helicopter may appear in a drone-allowed area. Even more, when it comes to operations in densely populated areas such as cities, a quantity of collision targets grow enormously. With expected policies as for geo-fencing and specified flight corridors for the UAVs, space for their operation will be further reduced and thus the risk of collision with another UAVs is increasing. Therefore, preventing collision is a very challenging functionality, especially speaking of a group or swarm of drones [2].

Like all the other aviation vehicles, UAVs have a limited payload, and many are not able carry heavy on-board equipment, which brings certain constraints to design of anti-collision systems. One solution is to operate them directly by pilot/operator control within the visual range. This form of operation may be reasonably safe, but it depends entirely on abilities of the operator and is prone to be highly affected by unexpected factors. Given limitations make the operator-based solution impractical. Therefore, it is important to have a system that is capable of monitoring the movement of UAVs in the airspace and solving their eventual collision situations in real time [3]. Principally, 5G is the name of a class of communications

technologies. The only 5G technology currently standardized by 3GPP, a cooperative responsible for the development of the 3G UMTS and 4G LTE standards, is 5G NR (New Radio) [4]. 6G networks will embed planning into the network itself, meaning that the network will become aware of the way it is being used [5]. This application will use the design of IEEE 802.15.4 compliance SoC silicon devices that will be used to communicate between UAVs. SoC solutions that integrate all functions (such as a wireless transceiver, data processing unit, memory and user application function) to single silicon chip, are functional, cost-effective and significantly faster superiority [6].

A. Simulation of Camera System Architecture

After testing different simulators, it was decided to use *tum_simulator*, which is a package containing the AR.Drone 2.0 implementation in the Gazebo simulator. This software is often used with ROS driver *ardrone_autonomy* for Parrot AR-Drone 1.0 or 2.0. This solution facilitates implementation on a real device. The data transmission from drone depends on *navdata_demo* parameter. The default setting is *50Hz* for non-real time updating and the drone transmission rate is *200Hz* [7].

The test setup includes a real drone communicating with host computer via Wi-Fi. The user controls the quadcopter using a joystick connected to the host computer running ROS. For simulation, the user can operate the drone in entirely modelled VR environment or partially simulated AR using the physical control (Fig.1).

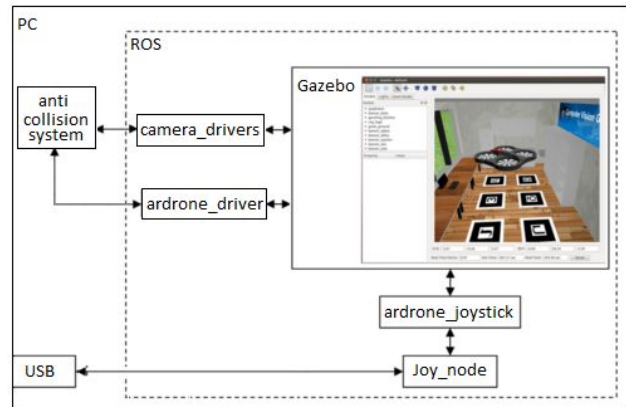


Fig. 1 Simulation architecture

The proposed object detection architecture (Fig. 2) is scalable, easily adjustable and compatible with mini computers such as Raspberry Pi, Banana Pi, and similar. Modern implementations of mini PCs provide relatively high computational power while keeping a low power consumption, which is suitable for outdoor testing of simulated AR scenarios.

Because the Galileo, the European Union's Global Satellite Navigation System (GNSS), is more precise than GPS by design, it was decided to equip drones with

location module utilising both, GPS and Galileo. Using only satellite-based location data for preventing collision is not appropriate due to low precision of positions measurements. For UAVs operating in close proximity, relying on a GPS does not help to avoid collisions. As a sensor for anti-collision system, drone cameras were used. Cameras capture 720p video at 30 fps. ROS allows access to the front and bottom drone cameras via the *ardrone/front* and *ardrone/bottom*.

In many applications utilising the OpenCV library, images are processed in conjunction with ROS [8]. To convert ROS images to *cv::Mat* format for OpenCV it is convenient to use the CvBridge library. CvBridge allows also converting OpenCV images to ROS image message [9].

The very effective tool, that was used to create histogram-based object detectors is the *dlib* library. The *dlib* library contains a large number of machine learning algorithms, one of which is *simple_object_detector*, which helps create and tune the object detector for *SVM* solver [10].

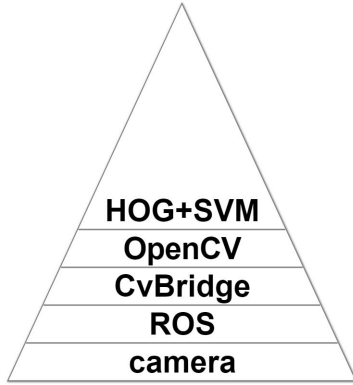


Fig. 2 Object detection architecture

For conflict detection and resolution, it is essential to compare the UAV trajectory and the trajectory of the subject. Trajectory can be produced by the trajectory calculation unit from raw data obtained at input. The most appropriate approach to conflict detection for UAVs operating in autonomous mode is the *Flight Planning Method*. This method generates the most probability of keeping a predefined flight plan. UAVs exchange parts of their flight plans, so the exact trajectory is known [11].

The anti-collision solution system (Fig.3) calculates the height difference alt_diff between the current flight objects from their altitude alt_1 and alt_2 . By comparing the remaining battery capacity bat_cap of both UAVs, it decides which UAV is to make a change in flight height. Subsequently, the system changes the height based on whether the height difference is negative or positive [12]. The new height of the system will determine that the difference between the new heights of both UAVs is 5 m.

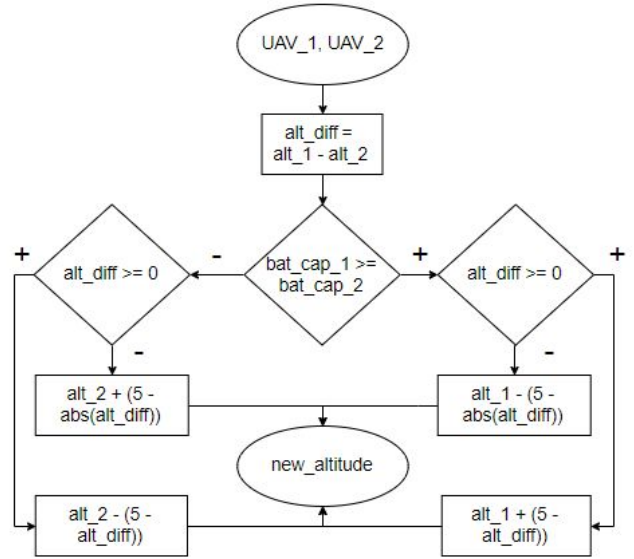


Fig. 3 Altitude change algorithm

B. Simulation of UAVs Detection with Camera

There are many approaches for handling object detection in dynamic environment. The most popular are *Haar cascades* and *HOG+SVM* for their performance [13]. In given case, an object detector was built using *HOG+SVM* model. Structural cues of UAVs give more robust representation such as shape. Gradients of specific directions captures some notion of shape.

The main idea of *HOG* is instead of using every individual gradient direction of every individual pixel of an image, group the pixels into small cells [14]. For each cell, all the gradient directions are computed and grouped into several orientation bins. The gradient magnitude in each sample is summarised. Furthermore, stronger gradients increase more weight to their bins, and reduce the effects of small random orientations due to noise. This histogram gives a picture of the dominant orientation of that cell. Doing this for all cells gives a representation of the structure of the image. The *HOG* features keep the representation of an object distinct but also allow for some variations in shape [15].

To create *HOG* feature descriptor for drone detection, it is calculated based on a 64×128 pattern of an image. The size of the image may be different though. In many cases patterns are analysed at many image locations. There is a constraint for the analyses, the aspect ratio has to be 1:2. It means that the aspect ratio can be, for example 100×200 or 128×256 but not 105×216 . In the given case, the size of the original image from the simulator is 640×360 . The size of the image to calculate the *HOG* feature descriptor after cropping and resizing is 64×128 [16]. After these operations, the *HOG* descriptor for the particular image pattern is calculated.

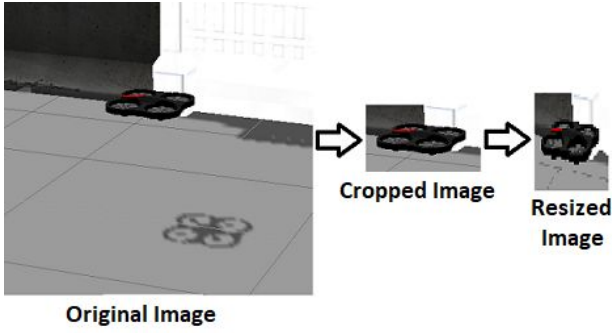


Fig. 4 Pre-processing of the image

For calculating gradient images g_x and g_y from the original image, the original image with the kernels shown in Fig. 4.

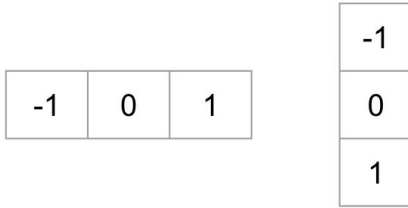


Fig. 5 Kernels for filtering the image [17]

Using the gradient images g_x and g_y , it is possible to calculate the magnitude and orientation of the gradient using the equations (1) and (2).

$$g = \sqrt{g_x^2 + g_y^2} \quad (1)$$

$$\theta = \arctan \frac{g_y}{g_x} \quad (2)$$

The calculated gradient are unsigned and therefore θ is in the range 0 to 180 degrees [18].

After dividing the image into 8×8 cells, it calculates histogram of gradients in these cells. One of the important reasons to use a feature descriptor to describe a patch of an image is that it provides a compact representation. From the sensitivity of the image to the illumination, it is appropriate to normalize the histogram over a bigger sized block of 16×16 , as if the image is darker, the gradient size changes by half. Each 16×16 block is represented by 36×1 vector and there are 105 positions.

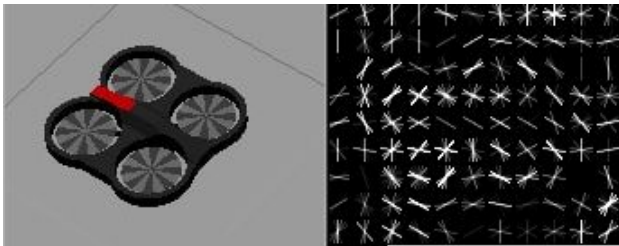


Fig. 6 HOG descriptor

To create and train a linear *SVM* model, extracted *HOG* functions use the descriptor from each area of the object.

When detecting an object from a drone, it is necessary to estimate the average window size and create a picture pyramid by swapping or shrinking images to several layers. After moving the window through each layer in the pyramid image, the *HOG* functions from each location is extracted. The probability of the learned *SVM* model with the current *HOG* features is evaluated, and if it is more than the specified threshold then it contains the object being searched.



Fig. 7 Detecting the search object

IV. SIMULATION OF NETWORK COMMUNICATION

For simulation, a small group of five drones was used. The ground communication was arranged via 4G network using existing infrastructure and inter-drone communications were arranged via ZigBee. Every drone was equipped with a Raspberry Pi unit that included 4G and ZigBee communication modules. Every drone, in addition to main camera, was equipped with two secondary cameras located in the front and on the left side of the drone. The collision detection was enhanced with ultrasonic sensors located at the front of every drones. This equipment was utilised in anti-collision system with a sampling algorithm for detecting the possible collision target. The whole principle is that all the video streams from every drone camera was received and processed by the central surveillance system.

The control of a group of drones was organised based on a principle of coordinated flight. In accordance to the principle, the control commands are sent to one drone only. The other drones repeat the behaviour of the controlled drone utilising the collision-avoidance system for preventing accidents.

In case the operator loses the signal with the controlled drone, the system automatically switches to control to the other drone of the group, and the rest of the drones update their behaviour correspondingly. The system is also effective in a case of a sudden discharge of a drone battery. The energy-conscious drone determines the take-off base location using on-board GPS/Galileo module and whenever it is possible in out-of-control or maintenance-is-needed cases, returns to the base carrying along the other drones of the group.

Every drone was equipped with a 5700 mAh battery [19], based on data from DJI for the drones of Matrix 600 Pro type. The approximate flight time is 38 minutes . To calculate the flight time of a drone, the consumption of on-board equipment, such as Raspberry Pi 3, has to be

calculated. The consumption value of Raspberry Pi 3 B+ is 985 mA [20].

Based on consumption of Raspberry Pi and drone consumption it is possible to calculate the flight time of the drone. Based on the above calculation, Raspberry Pi operations consume battery power required for approximately five minutes of flight time. For the given type of drones, if this is required, it is possible to extend their power and functional capacity by adding other modules.

A. ZigBee communication between drones

Every drone sends updates about its GPS/Galileo-identified position to other drones of the group. An example of communication measures is presented in Table 1. Eight-byte (64 bits) floating point variable were used for the coordinates. To test the sixteen-bytes (128 bits) communication channel, twenty bytes were sent using ZigBee technology. The purpose of that simulation was to verify the success of the communication between the drones and the delays that occur during the broadcast data transmission.

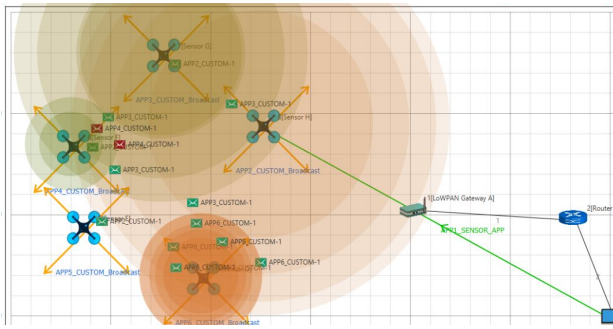


Fig. 8 visualization of the signal and graphical representation of send data

In the Figure 8, the ZigBee-based drone network topology is shown. The drones communicate with each other and for simulation purposes with the ZigBee gateway, which acts as a gateway between Zigbee and IP networks. The gateway was used to evaluate the quality of data transmission. In addition to tracking the data flow, the focus was on the throughput in transmitting data of 20 bytes but also on the network delay. It is very important to have a fast response so that the drones are updated in time of their neighbour's position. The ZigBee-based communications and the camera-based anti-collision system supplement each other.

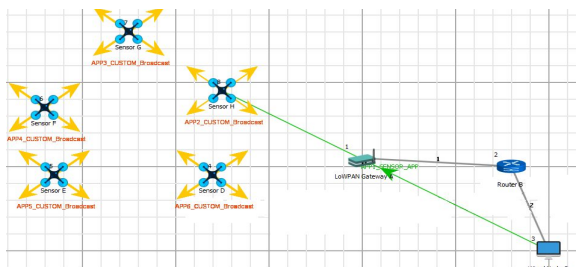


Fig. 9 Concept of drones during simulation Zigbee technology

Within the proposed implementation, every drone knows its position and at the same time can use the camera system to more accurately determine its relative position with respect to the other group of drones. A visual representation of drones is shown on Figure 9.

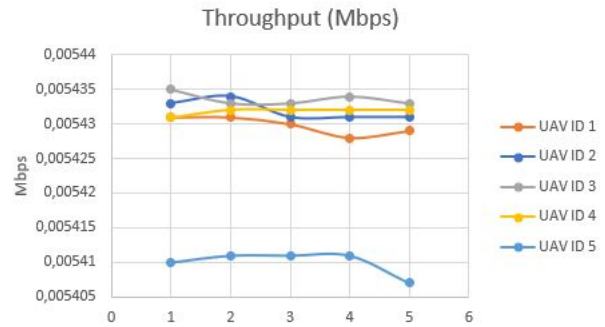


Fig. 10 Graph of throughput between UAVs

The throughput measurements are shown in Figure 10. Each drone sends broadcast data to other drones. The throughput holds at 5.4Kbps, which is well sufficient to transfer 20 bytes of position data. With the increasing distance between the UAV, the speed decreases.

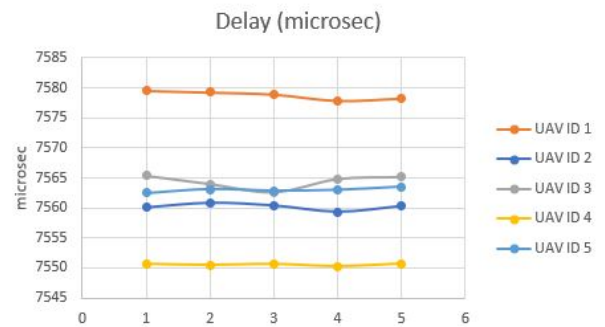


Fig. 11 Graph of Delay between UAVs

The delay are shown in Figure 11, that occurs in this communication is within the order of 7 milliseconds, which is sufficient for the transfer of specified position data block, relative position with respect to the other group of drones. There is a delay in receiving each other from another drone between each drones. If the value is kept at low values, the drone gets the data on time and can respond.

B. Result of 4G communication between system and drone

4G network communications were simulated using the NetSim program for the city of Košice. 4G modems implemented as USB dongles were used as interfaces to 4G telecommunication network. A group of five drones operated in the city at a speed of 10 m/s for simulation purposes. The control commands were always sent to one drone only. The other drones were operated based on position of a controlled one and used the anti-collision system. The drones were equipped with a battery check system that predicted a need for a maintenance. In case of communication loss, the operator could communicate with other drones from the group, but at the same time could

communicate with only one of them in accordance to the principle of coordinated flight. In addition to control commands, drones streamed videos from their main cameras through this network, and the resulting image of the multi-view system was discharged at the headquarters. Video streaming was encoded using the H.265 codec that allows to use bandwidth more effectively and ensures optimal image quality for the operator. Videostream has the following parameters: 30 frames per second, 1920 x 1080 resolution. The average Throughput was 2.75 Mbps, detailed information is described in Tab. 1.

Table 1: Throughput of video streaming from UAVs

UAVs ID	Throughput (Mbps)	Delay (microsec)
1	2,603805	2346,928851
2	2,593514	1800,996612
3	2,594603	2807,288934
4	2,807261	1102,914016
5	3,110538	1778,616882

The communication delays were low because the simulation was in an isolated 4G network.

V. FUTURE WORK

In the future work it is planned to expand the proposed system to a swarm of drones and conduct more experiments about the communication requirements and quality of communications. More experiments are needed to improve the anti-collision system by distinguishing not only UAVs, but also other fixed or mobile objects such as birds. The solution would be to modify the algorithm for detecting different objects using different training sets for SVM.

VI. CONCLUSIONS

This paper presents the UAVs management system operating using the 4G and ZigBee networks. The anti-collision system uses SVM technique of machine learning along with HOG image processing technique. The developed system was tested and simulated in terms of ability to avoid other drones using on-board camera and transmit data within the drone group and the ground station.

The HOG + SVM has proven to be very useful for detecting UAVs using the Flight Planning Method. However, the disadvantage of the current version of the system is that it can recognize only UAVs. The missing functionality is still to be developed.

Throughput and delay are important parameters of the network, which were simulated and tested in field experiments. Based on those values, it is possible to determine whether the UAVs are able to exchange data

using Zigbee. The described experiments have given a positive answer to this question. It has also been confirmed that modern 4G networks provide sufficient meant for ground communication given current development of sensors and camera technologies.

ACKNOWLEDGMENT

This work was supported by Cultural and Educational Grant Agency (KEGA) of the Ministry of Education, Science, Research and Sport of the Slovak Republic under the project No. 062TUKE-4/2017 and the project No. 023TUKE-4/2017.

REFERENCES

- [1] Gislason, Drew. *ZigBee wireless networking*. Newnes, 2008.
- [2] Van Tooren, Joost, et al. "Collision and conflict avoidance system for autonomous unmanned air vehicles (UAVs)." U.S. Patent No. 7,737,878. 15 June. 2010.
- [3] Volčko, Tomáš, et al. "Possibility of usage the latest GSM generations for the purpose of UAV communication." *New Trends in Signal Processing (NTSP)*. IEEE, 2016.
- [4] Sanders, James. "5G Mobile Networks: A Cheat Sheet." *TechRepublic*, 8 Mar. 2018, www.techrepublic.com/article/5g-mobile-networks-a-cheat-sheet/.
- [5] Saracco, Roberto. "Let's Start Talking about 6G!" *IEEE Future Directions*, 25 Jan. 2018, sites.ieee.org/futuredirections/2018/01/25/lets-start-talking-about-6g/.
- [6] Velez, Fernando J., et al. "Wireless sensor and networking technologies for swarms of aquatic surface drones." *Vehicular Technology Conference (VTC Fall), 2015 IEEE 82nd*. IEEE, 2015.
- [7] Banach, Artur. "Visual control of the Parrot drone with OpenCV, Ros and Gazebo Simulator." (2016).
- [8] C. Hudson et al., "A training tool for speech driven human-robot interaction applications" *In: ICETA 2017*. IEEE, pp. 167-172, 2017.
- [9] M. Pleva et al., "English speech interface to Jaguar robot for SWAT training" *In: SAMI 2017*. IEEE, pp. 105-110, 2017.
- [10] King, Davis E. "Dlib-ml: A machine learning toolkit." *Journal of Machine Learning Research* 10 Jul (2009): 1755-1758.
- [11] Angelov, Plamen, ed. *Sense and avoid in UAS: research and applications*. John Wiley & Sons, 2012.
- [12] Volčko, Tomáš. "Monitorovací a Protizrážkový Systém Pre Malé UAV." *Košice: TU*, 2017.
- [13] Sedaghat-Pisheh, Hani, et al. "Collision avoidance algorithms for unmanned aerial vehicles using computer vision." *Journal of Computing Sciences in Colleges* 33.2 (2017): 191-197.
- [14] Singh, Wahengbam Suman, and Sapam Jitu Singh. "Vehicle Detection And Counting Using Hog Feature Extraction For Traffic Signal Control System." (2017).
- [15] Hu, Rui, and John Collomosse. "A performance evaluation of gradient field hog descriptor for sketch based image retrieval." *Computer Vision and Image Understanding* 117.7 (2013): 790-806.
- [16] Mallick, Satya. "Histogram of Oriented Gradients." *Learn OpenCV*, 6 Dec. 2016, www.learnopencv.com/histogram-of-oriented-gradients/.
- [17] Mallick, Satya. "Image Recognition and Object Detection." *Learn OpenCV*, 14 Nov. 2016, www.learnopencv.com/image-recognition-and-object-detection-part1/.
- [18] Gungor, Vehbi C., et al. "Smart grid technologies: Communication technologies and standards." *IEEE transactions on Industrial informatics* 7.4 (2011): 529-539.
- [19] dji. "MATRICE 600 PROSPECS." *MATRICE 600 PRO*, www.dji.com/matrice600-pro/info#specs.
- [20] Pidramble. "Power Consumption Benchmarks." *Raspberry Pi Dramble*, www.pidramble.com/wiki/benchmarks/power-consumption