УДК 004.9:629.734/735

COMPARATIVE ANALYSIS OF BUDGET COMPUTING PLATFORMS FOR A PORTABLE MICROMODULE OF ON-BOARD IMAGE

V.A. KOVALEV, PhD
Head of the laboratory of Biomedical Images Analysis, United Institute of Informatics Problems (UIIP) of the National Academy of Sciences of Belarus

D.A. PAULENKA
Software Engineer, United Institute of Informatics Problems (UIIP) of the National Academy of Sciences of Belarus

E.V. SNEZHKO, PhD
Leading Researcher, United Institute of Informatics Problems (UIIP) of the National Academy of Sciences of Belarus

V.A. LIAUCHUK
Researcher, United Institute of Informatics Problems (UIIP) of the National Academy of Sciences of Belarus

The United Institute of Informatics Problems of the National Academyof Sciences of Belarus, Republic of Belarus
E-mail: vassili.kovalev@gmail.com, dmitri.pavlenko@gmail.com, eduard.snezhko@gmail.com, vitali.liauchuk@gmail.com

Abstract. This paper is devoted to the analysis of basic hardware and software of recent cheap and commercially available computing microplatforms for selecting an appropriate solution for development of an onboard micromodule for preliminary classification and selection of images of underlying surface of given types. It is assumed that the corresponding versions of the micromodule can be installed on board of small spacecraft or light unmanned aerial vehicles (drones). In this paper we consider a variant of a micromodule for drones. When choosing a microplatform, the main limitations were its low weight (no more than 300 grams, including camera and interface equipment) and its relatively high performance (time for frame processing of a color image 320×240 pixels is no more than 300 milliseconds). Another important limitation was the low price and commercial availability of micro-platform on the Belarusian market. The information provided in this paper could be useful for engineers and researchers who develop compact budget mobile systems for processing, analyzing and classification of images.

Key words: microcomputer, mobile system, image classification, drone, convolutional neural network

Introduction. In the last few years, significant progress has been made in development and usage of small spacecrafts and unmanned aerial vehicles (drones). This creates prerequisites for development of various systems for automatic video data collection, including mobile systems for search, recognition and selection of necessary video data directly on board of the flying vehicle. Such an option saves the operator from long and tedious viewing of captured photos and videos in order to select sites and objects of interest, because the data collection system captures only those image types (classes), whose samples were included in the flight task before departure. Furthermore, in a number of cases there is an additional possibility of significant energy saving on board, because power-consuming surveillance and photography components can only be used if necessary.

The article's purpose is to analyze basic hardware and software of inexpensive, commercially available computing microplatforms and carry out computational experiments to select a suitable solution for the development of a micromodule for preliminary recognition and selection of images of underlying surfaces on board of light drones. The main features and key technical solutions of the micromodule are listed below.

1. It is assumed that the micromodule is an autonomous system for automatic recognition images of underlying surfaces of various types. Examples of these underlying surfaces of various types can be the following: forests of various types, industrial structures, reservoirs, roads, fields, bushes, country houses, farmland of various types with vegetation at different stages of growth, etc.

2. Specific target classes are set in the flight task, either in the form of corresponding image samples, in case of traditional "image-characteristics-classifier" type recognition scheme, or in the form of pre-trained convolutional neural network, in case of Deep Learning technology.

3. Imbedded video camera of micromodule is used only to solve the task of recognizing the surface type. If necessary, frames of the video sequence (these are low-resolution images of the target classes of the flight task) can be stored on the external micromodule carrier for auxiliary purposes.

4. Interaction between micromodule and drone should be minimal. It should be reduced to real time transferring from micromodule to UAV the number of the current class of underlying surface. Class "0" means that the current image does not belong to any of the target classes specified by the flight task. Additional parameter is the probability (certainty degree) of classifying. The final decision to shoot and record or transmit image to the ground, as well as high-quality shooting itself is done outside of the micromodule.

5. It is assumed that the selected microcomputer for micromodule is a complete computer system provided with all necessary interface equipment, operating system, software development tools, etc. The implementation of the micromodule from individual specialized components such as microcontrollers, digital signal processors (DSPs), chipsets, systems on a chip, and other parts with the help of appropriate circuit solutions is not considered.

The comparative analysis is carried out using method of strategic planning SWOT [1], which reveals the factors of the internal and external environment of the device and divides them into four categories: strengths, weaknesses, opportunities and threats (risks) of using analyzed device.

It should be noted, that the evolution of electronic computing systems has led to the fact that computers have become not only significantly more powerful, cheaper and affordable for the consumer, but also decreased many times in size. There is a whole market of single-board microcomputers capable of solving a variety of tasks [2].

Currently a microcomputer is an electronic computing device corresponding to a bank card by size and comparable with personal computers from previous generations by computing power. For example, a total performance for the Raspberry Pi Zero Wireless model is comparable to the Pentium 2 with a processor frequency of 300 MHz, but with much better graphics [3]. A microcomputer or a single-board computer is a self-contained computer assembled on a single printed circuit board with microprocessor, RAM, I / O systems and other necessary modules. Single-board computers are small in size and cheap. The cheapest of them can be purchased for $ 5-10 [4].

Special mention should be made about usage of smartphones as universal microcomputers. Indicative in this respect is the NASA PhoneSat project [5], carried out as part of the Small Spacecraft Technology Program to create nanosatellites using unmodified commercial smartphones and launching such satellites into low Earth orbit. The project was started in 2009 at the NASA Ames Research Center (Moffett Field, California). The cost of the components used to create one nanosatellite was from $ 3500 for version 1.0 and less than $ 7000 for version 2.0. The nanosatellite is built according to the CubeSat standard, has a size of 1U – it is 10×10×10 cm and weighs approximately 1 kg. In comparison, in 2004, before the launch of the PhoneSat project, the cost of the components for creating the classic CubeSat was estimated at between $ 20 000 and $ 40 000. At the same time, the commercial launch of such 1U nanosatellite into low orbit is about $ 85 000. Another project called STRaND-1 (Surrey Training, Research and Nanosatellite Demonstrator 1) is a nanosatellite "3U" CubeSat. Weight is 4.3 kg. Developed by the Surrey Space Center (SSC) and Surrey Satellite Technology Ltd (SSTL). The satellite was launched into orbit aboard of the PSLV C-20 Rocket from India on February 25, 2013. STRaND-1 became the first satellite running by the management of a smartphone.

1. Basic requirements for micromodule being developed. Requirements and constraints determined by the specification for the development considered here are presented below.

1. Micromodule weight (including camera and interface equipment) is no more than 300 g.
2. Size (length, width, height) is no more than 12×6×4 cm.
3. Power consumption in the active mode is no more than 5 W.
4. Recognition time of one RGB color frame image with 320×240 pixels is no more than 300 ms.
5. Number of the image types of the underlying surface (classes) recognized by the software is not less than 10, including classes which recognized simultaneously is not less than 3.
6. Recognition quality of a given image type of the underlying surface (depending on the type) is not less than 85-95%.
7. In a single production, the price of one micromodule should not exceed $ 500.

On the first stage, the authors investigated the following types of computing microplatforms: several single-board microcomputers of various manufacturers; nettops; smartphones; products with the Mobile-ITX form factor of the motherboards for x86-compatible processors; PC/104 units (or PC104); NVIDIA Jetson TX2 Developer Kit; SMARC-modules of SMARC standard version 2.0; high-performance embedded computer Manifold DJI; various types of so-called computers on the module (computer-on-module, COM).

As a result of preliminary analysis, it was found that most of the above-mentioned models of microcomputers exceed the maximum permissible weight and size. Also according to the technical specifications above, the power consumption of many models exceed the 5 watts' limit in the loaded-state operating. In addition, the price of some microcomputers is too high and/or their acquisition seems problematic in the Belarusian market. So, for example, SMARC-modules [6] are not suitable for both price and size. The size of the carrier board for SMARC-module (20×21 cm) considerably exceeds the maximum permissible size (12×6×4 cm). The price of the cheapest SMARC-modules starts from $ 100 and this is without taking into account the price for necessary carrier board. As a result, four computing platforms were selected for testing which are considered in the following sections.

2. Tested computing microplatforms and their technical specifications. Of all the diversity of microplatforms, for conducting comparative analysis four modern, inexpensive, commercially available microplatforms were selected and purchased, which are typical representatives of their classes (Table 1):

– Intel Compute Stick STK1AW32SC [7] with the operating system Windows 10 Home Edition 32-bit (Fig. 1);
– Raspberry Pi 3 Model B [8] with the operating system Raspbian Stretch 4.9 (Fig. 2);
– Raspberry Pi Zero Wireless [9] with the operating system Raspbian Stretch 4.9 (Fig. 3);
– Xiaomi Redmi Note 4 [10] with the operating system MIUI 9.2 Global based on Android 7.0 Nougat (Fig. 4).

Figure 1. Intel Compute Stick STK1AW32SC

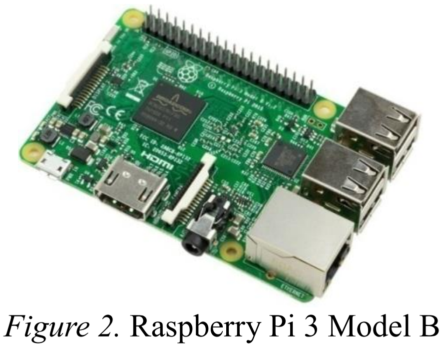

Figure 2. Raspberry Pi 3 Model B

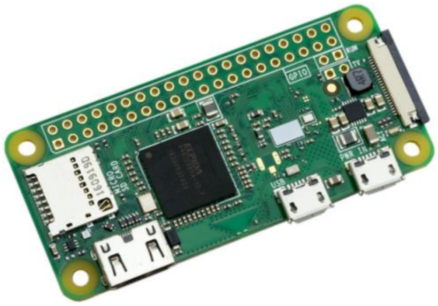

Figure 3. Raspberry Pi Zero Wireless. Latest ultra-light Pi model

Figure 4. Xiaomi Redmi Note 4. Modern budget smartphone

Table 1

Main technical specifications of the tested devices

№	Technical characteristics of microcomputer	Intel Compute Stick	Raspberry Pi 3 Model B	Raspberry Pi Zero Wireless	Xiaomi Redmi Note 4
1	Release date, month/year	03/2016	02/2016	02/2017	01/2017
2	Recommended retail price, USD	129	35	10	250
3	Manufacturer company, country	Intel Corporation, USA	Raspberry Pi Foundation, UK	Raspberry Pi Foundation, UK	Xiaomi Inc., China
4	Size: length width height, cm	12.3 3.8 1.2	8.6 5.7 1.7	6.5 3.0 0.5	15.1 7.6 0.9
5	Weight, g	60	45	9	165 (with battery)
6	Operating temperature range, °C	from 0 upto +35	–	–	from 0 upto +40
7	CPU Model	Intel Atom x5-Z8300	ARM Cortex-A53	ARM1176JZF-S	Snapdragon 625 (MSM8953)
8	CPU frequency, GHz	1,84	1,2	1	2,02
9	Number of processor cores, pcs.	4	4	1	8
10	GPU Model	Intel HD Graphics	Broadcom VideoCore IV 1080p60	Broadcom VideoCore IV 1080p30	Adreno 506 (650 MHz)
11	RAM size, GB	2	1	0,5	4
12	Built-in memory size, GB	32	no	no	64
13	Specialized / built-in camera resolution, Mp	no	5 or 8	5 or 8	13 and 5
14	Autofocus	no	no	no	yes

As can be seen from Table. 1, the price of devices varies from \$ 10 to \$ 250 and correlates with the processing power parameters (number of processor cores, CPU clock frequency, RAM size, built-in memory size, etc.), as well as the presence of external interfaces and the camera. All devices meet a restriction of 300 grams by weight parameter, but Xiaomi Redmi Note 4 does not pass by size (length is 15.1 cm against the maximum allowable 12 cm). However, this excess in a certain sense is “compensated” by its small thickness of 0.9 cm and, consequently, by a small volume.

The value of the “operating temperature range” for Raspberry Pi is not indicated, because there is no data about operating temperature range in the official specifications. In FAQs [11] it is indicated that Raspberry Pi models are assembled from commercial chips of different manufacturers, each has its own operating temperature range. It is indicated that the LAN9514 microchip for USB and Ethernet can operate in the temperature range from 0° C to +70° C, while system on a chip (SoC) is capable of withstanding temperatures from -40° C to +85° C.

It is known that for mobile computing devices, an important group of parameters affecting the choice of the platform are also the power supply and the power consumption parameters. Corresponding values for the four tested devices are presented in Table 2 and on Fig. 5.

Table 2

Electrical parameters of the tested devices

Operating conditions	Intel Compute Stick			Raspberry Pi 3 Model B			Raspberry Pi Zero Wireless			Xiaomi Redmi Note 4		
Electrical parameter	V	A	W	V	A	W	V	A	W	V	A	W
Load-less operating	5.00	0.80 ± 0.10	4.00 ± 0.50	5.00	0.35 ± 0.01	1.75 ± 0.05	5.00	0.12 ± 0.01	0.60 ± 0.05	4.20	0.12 ± 0.02	0.51 ± 0.08
Loaded state operating	5.00	1.70 ± 0.30	8.50 ± 1.50	5.00	1.34 ± 0.10	6.70 ± 0.50	5.00	0.35 ± 0.05	1.75 ± 0.25	4.20	0.7 ± 0.05	2.94 ± 0.21

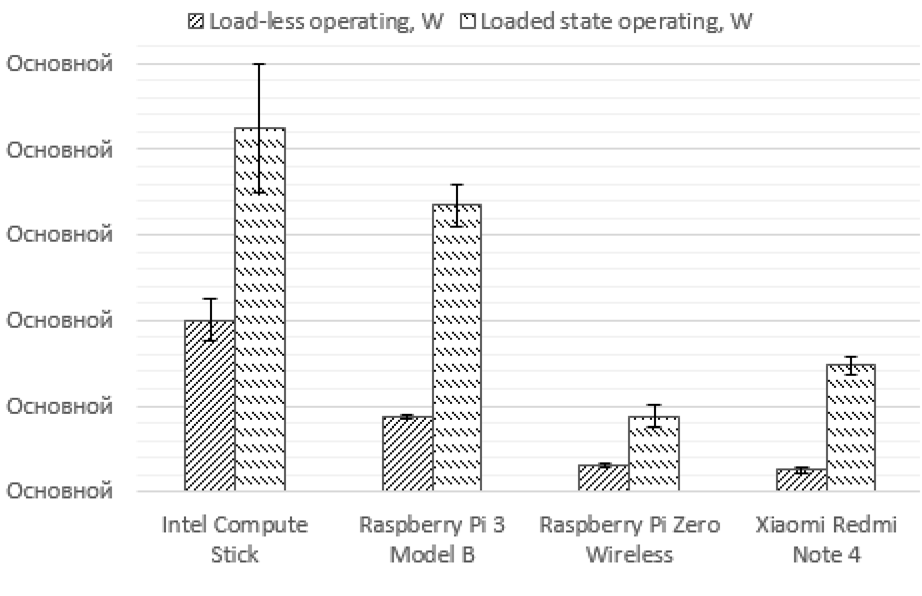

Figure 5. Devices power consumption

Despite the fact that all devices use a 5 volts supply voltage (4.2 volts for the smartphone), the differences in the average current in the loaded state range from 0.35 amperes in Raspberry Zero to 1.70 amperes in the case of the Intel Stick. Respectively, the average power consumption varies from 1.75 to 8.50 watts. In all cases, except Xiaomi Redmi Note 4, it should be noted that when using an external camera such as for Raspberry Pi, an additional 0.25 amperes will be required to ensure its normal functioning. In accordance with the specification requirements (see section 1 above) the power consumption of the micromodule in our case should not exceed 5 watts, it means that Intel Stick and Raspberry Pi 3 shouldn't be used. In addition, it should be noted that due to the relatively large power consumption and compact form factor, the Intel Stick microcomputer in its original branded version is equipped with a cooling fan and has corresponding holes in the case. When used outdoors, this can cause moisture to enter into the device's housing and damage it.

3. Main software and its features. Many modern microcomputers allow to install on them the same operating systems and software as on ordinary desktop personal computers. This allows development on a desktop computer, and then transfer results to the microplatform with minimal changes, which is simplifies and speeds up the work on the project.

Intel Compute Stick STK1AW32SC comes with preinstalled Windows 10 Home Edition 32-bit (paid) operating system. Specialists claim [12] that on this device can also be installed operating system Ubuntu 16.04 LTS 64-bit, but this option has not been tested. It is known that the Windows Home edition is positioned as a "home" operating system, focused on the usage outside of the office or enterprise. By this reason, this operating system does not have many services. For example, you cannot disable automatic system updates, there is no pre-installed Remote Desktop, no group policies, etc. Windows Home operating system is quite resource intensive, it consumes relatively much of the operating memory of the microcomputer and occupies two-thirds of the built-in non-volatile memory.

For the Raspberry Pi products (in this case Raspberry 3 and Raspberry Zero), the recommended operating system is Raspbian [13], which is based on Debian Linux. The latest release is Raspbian Stretch 4.9. This operating system is based on the Linux kernel and is free open source project under the GNU license. Despite the fact that 32-bit versions of Windows and Raspbian are installed on the reviewed microcomputers, installation of 64-bit versions of Windows and Linux is possible too. However, it is not practical in this case, because 64-bit versions require more operational and external memory. As a result, there will be insufficient resources to install and effectively run the image recognition application software.

Operating system MIUI 9.2 Global of the smartphone Xiaomi Redmi Note 4 is based on the Android 7.0 Nougat source code. The user interface of MIUI is different from the interface of the original Android, but the MIUI system itself is compatible with applications developed for the Android operating system. From the Android Market application, which is available on the MIUI 9.2 Global operating system, one can download and install any application for the Android system.

All three operating systems support such common programming languages and their environments as Java, C/C++ and Python. However, Android operating system has weak support of Python. The Android operating system itself is written in C/C++ and Java programming languages, which are optimal for developing applications for Android. There are several projects that develop the Python interpreter for Android [14]: BeeWare, Chaquopy, Kivy, pyqtdeploy, QPython, Termux, etc. However, when choosing such interpreters, there is a sufficiently high risk that over time some of these projects will cease to be supported by their developers. Also, there is every reason to believe that the support of the Python programming language for Android will improve over time. In addition, there is so-called Termux [15] – a freely available Linux emulator for the Android OS, which was well proven in the computational experiments discussed below.

4. Results of microplatforms performance testing. Microplatforms computing power plays a very significant role in solving the problem of preliminary recognition of images of the underlying surface in real time. In our case, the detection rate of one frame of a color RGB image of 320×240 pixels is limited to 300 ms, i.e. the performance of the micromodule should be more than 3 frames per second. Taking into account the experience of solving various problems of color image recognition, an algorithm based on the representation of the content of images with the help of co-occurrence color matrices was adopted as the basic algorithm used for testing [16]. The color space was reduced by using the highest bits of the R, G and B color components according to the scheme 3-3-3 (512 colors). Accordingly, the dimension of the resulting co-occurrence matrix was 512×512. After calculating the matrix for the input image, all of its non-zero elements were written to a vector called an image descriptor.

With any variant of the recognition algorithm construction, the basic operation is to estimate the closeness of the current image to the class samples specified in the flight task. This operation was performed by comparing the image descriptors and calculating a measure of proximity of type L1 in the feature space. Speed of execution of the listed basic operations for all platforms, implemented in Python version 2.7.13, is presented in Table 3. It should be noted that the algorithm based on the co-occurrence color matrices was used here as a test for the speed of the microplatforms being analyzed. However, it does not mean that only this algorithm is used in the developed micromodule to solve the recognition problem.

As can be seen from Table 3 and Fig. 6, data processing speed varies greatly depending on the libraries been used. Using extension for Python on a C programming language expectedly increases processing speed on all microplatforms about twice if compare with the usage of the Scikit-image library for images handling and in 3-5 times quicker if compare with the usage of NumPy library for arrays handling.

Table 3

Speed of execution of the test algorithms in milliseconds

No.	Calculation time of co-occurrence matrix and descriptor for image 320×240 pixels and 512 colors in milliseconds, ms	Intel Compute Stick	Raspberry Pi 3	Raspberry Pi Zero Wireless	Xiaomi Redmi Note 4
1	Extension of C language for Python	19 ± 2	44 ± 1	55 ± 3	15 ± 1
2	Scikit-image library for image processing	32 ± 1	87 ± 1	105 ± 4	39 ± 1
3	NumPy library for array processing	63 ± 2	182 ± 1	275 ± 10	51 ± 1
4	Double cycle (cycle in a cycle)	1 888 ± 4	3 918 ± 6	14 131 ± 209	1 560 ± 11

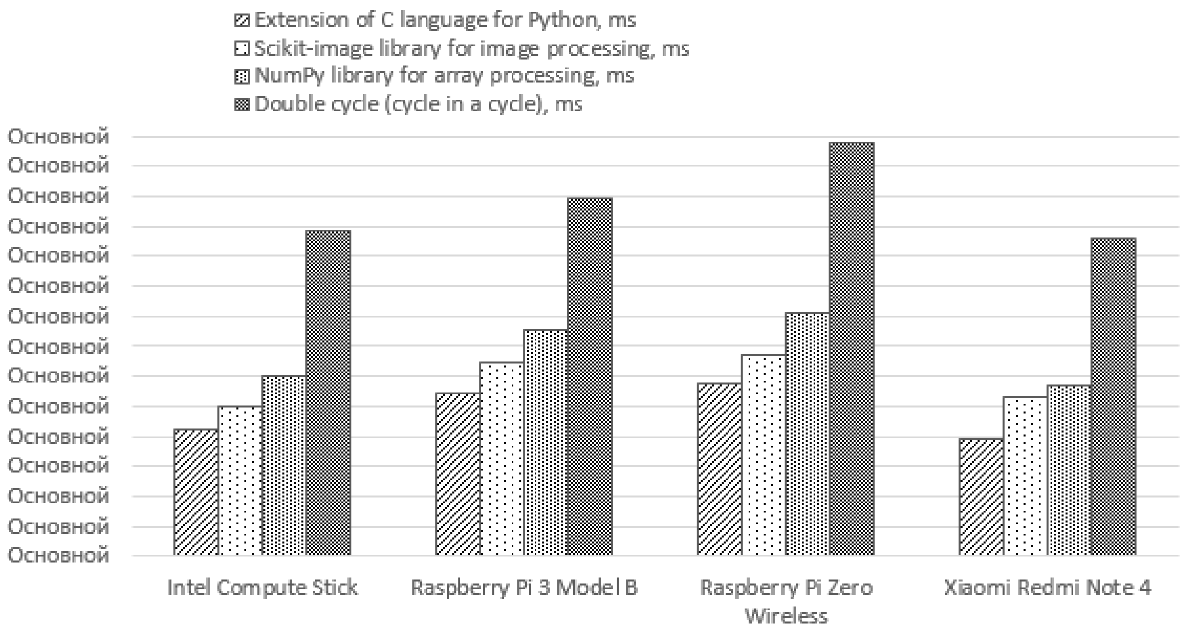

Figure 6. Calculation time of co-occurrence matrix and descriptor for image 320×240 pixels and 512 colors in milliseconds (ms) in logarithmic scale

As for the performance of various microplatforms, it should be noted that run time for tests is slightly differ for Intel Compute Stick and for smartphone Xiaomi Redmi Note 4, despite the fact that the smartphone's processor is more powerful according to the technical specifications. Presumably, it happens because on the smartphone with Android operating system a Python interpreter works via emulator and is not supported by the Android operating system directly. Measurements on the smartphone for the Android operating system were conducted using Termux [15] (Linux emulator for Android). Some of the measurements were repeated on QPython [17] for Android. As measurements have shown, QPython is slightly slower than Termux.

5. Model of micromodule. Data obtained from the comparative analysis of four computing microplatforms allowed to conclude that the best microcomputer for the micromodule of operational image recognition is the Raspberry Pi Zero Wireless model with the Raspbian operating system. This model has minimum power consumption, size, weight, cost and is freely available for purchase. Computational experiments have shown that the main disadvantage of this microplatform is its low processing power, which can be overcame by effective implementation of recognition algorithms in the C/C++ programming language.

In the process of implementation, a case for micromodule was designed on the SolidWorks software package and then printed on a 3D printer (Fig. 7 and 8). The case accommodates: Raspberry Pi Zero Wireless microcomputer with Raspberry Pi Zero V1.3 Mini Camera (13 grams total), Li-Po battery type ROBITON with a capacity of 720 mAh (14 grams), a battery charger with a capacitor, signal LED lights and connecting wires (Fig. 9 and 10).

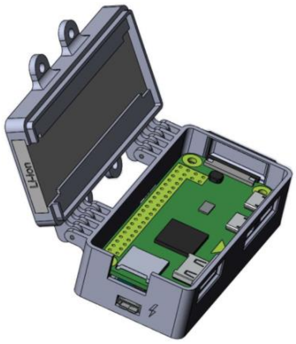

Figure 7. 3D-model of micromodule with the open case

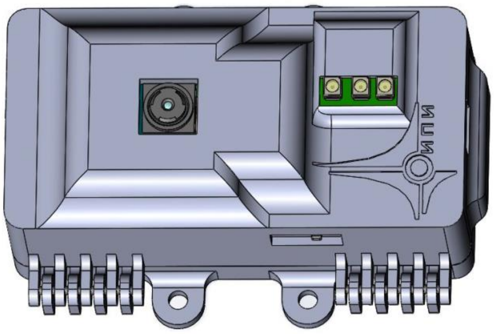

Figure 8. 3D-model of micromodule

Figure 9. Model of micromodule in disassembled state

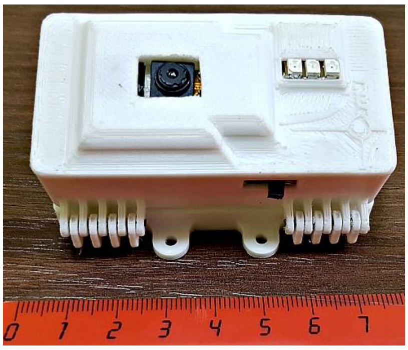

Figure 10. Model of micromodule

Model of micromodule has 7.3×3.6×2.5 cm and a total mass in the assembled state of 57 grams, which is much less than the limit of 300 grams. Such modest weight-and-size characteristics allow installation of micromodule even on budget drones with low load-carrying capacity, which creates additional competitive advantages. It should be emphasized that mentioned 57 grams include an autonomous power source that ensures the functioning of micromodule for entire flight time, charger, as well as the switch to an external outlet in hot mode, which is very convenient for the development and debugging of software. During laboratory tests using drone Parrot AR.Drone 2.0, the current recognized class of image in real time is sent directly to the color LED indicators on the housing of the micromodule layout. In addition, a copy of the current image frame and the results of its recognition are transferred to the laptop via Wi-Fi to simplify the process of monitoring and debugging the software.

6. Composition and general characteristics of the software being developed. It is known that the Python programming language provides fast prototyping and software implementation. In addition, it is widely used in most software frameworks for convolutional neural networks. However, it is not an ideal software development tool for embedded systems, where C/C++ and Java programming languages are commonly used. Therefore, prototyping and development of graphical user interfaces and other auxiliary functions are developed by the authors in Python, but code sections that are critical from the point of view of computational efficiency are implemented in C/C++ or using C/C++ extensions for Python.

Software of the micromodule consists from ground and flight parts. Main task for the ground part is formation of the flight task by selecting samples of the target image classes for the traditional recognition scheme or loading of pre-trained neural network into micromodule.

Main task for the flight part is operative recognition of images of the underlying surface, i.e. analysis of the current frame of the video sequence and its assignment to one of the classes indicated in the flight task. This task is solved by two different methods, described below.

Traditional recognition scheme. The traditional scheme includes identification of characteristic features and usage of the known classifier such as kNN, SVM and others. As characteristics, we use co-occurrence matrices, described in detail in [16]. High recognition speed is provided by storing only significant (non-zero) matrix elements in the form of "key-value" lists, by single-pass algorithm for characteristics comparison and by the kNN classifier with small values of k (from 1 to 10). As a result of the experiments, it was shown that the recognition time of a single 320×240 pixel frame is 14.5 milliseconds when using 256 colors (2-3-3 bit scheme) and 44.5 milliseconds when using 512 colors (3-3-3 bit scheme).

Recognition using convolutional neural network. Rigid limitations of the basic microcomputer on the speed of computations and the size of the RAM prevent to use one of the widely known neural networks such as AlexNet or GoogLeNet. Therefore, it was decided to develop and use its own, compact architecture of the convolutional network. Main solutions and implementation features for image recognition software based on convolutional neural network are briefly described below.

1. The process of solving the recognition problem consists of two stages: training neural network on a samples of images of the target classes of the underlying surface and operative recognition of images from the camera on the flying part of micromodule software. Network training refers to the ground part of micromodule software and is implemented using a powerful graphics processor. Recognition of image frames from the camera does not require a powerful processor, since image convolution with the filters obtained during the learning phase of the network is performed.

2. Taking into account a limited number of classes to recognize, as well as a small computing power of microcomputers, a simplified architecture of convolutional neural network was implemented, the main characteristics of which are presented below.

– The first part of our network consists of two blocks arranged in series and containing convolutional layers. In the first block, the convolutional layer contains 12 convolution kernels 3x3 pixels in size each. In the second block, the convolutional layer contains 24 kernels. After each convolutional layer there is a layer with the activation function ReLU: $f(x) = max\ (0, x)$. Behind the activation layers there are subsampling layers that perform a non-linear compaction of the characteristic map.

– After the above-mentioned blocks, a fully connected network is located, the output of which is another activation layer with the Softmax function.

– The convolutional network output for the input image is a 5-element vector, which corresponds to the number of recognizable classes of the underlying surface. Each element of the vector represents probability that the input image belongs to the corresponding class. The image belongs to some class, if the specified probability exceeds the threshold of 0.99.

Implementation of the convolution algorithms is performed in C/C++ programming language with the subsequent "manual" code optimization, resulting to a single frame recognition time of 80 milliseconds, i.e. 12 frames per second for Raspberry Pi Zero Wireless. However, experiments showed that due to the limited architecture of the neural network and the influence of various noise factors (variability of illumination, altitude variation of drone flight, camera vibration, large images variability of "non-target" surface classes), the recognition accuracy varied widely from 75% to 95%, which did not meet the requirements.

Solution was found by usage of a cascade of two neural networks. Both networks have the same architecture described above. In this case, the first network was trained to divide an input stream into images related to a mixture of target classes (i.e., to any of target classes) and into "background" images. The second network was trained to divide a mixture of target classes from the first network into separated classes. As a result, it was possible to achieve a recognition quality of 90% or more, although the recognition time doubled, i.e. up to 160 milliseconds per frame for Raspberry Pi Zero Wireless. A more detailed consideration of image recognition quality issue is the subject of a separate publication.

7. A brief SWOT analysis of the tested microplatforms. Below is a brief SWOT-analysis of four tested microplatforms with concise enumeration of strengths, weaknesses, opportunities and threats (risks) for each tested microplatform.

<u>Intel Compute Stick STK1AW32SC</u>

Strengths:

– almost three times faster than Raspberry Pi 3;

– preinstalled operating system Windows 10 Home for a number of users can be more simple and familiar than Linux or Android.

Weaknesses:

– high power consumption and, as a result, need for a more capacious and heavy battery;

– presence of a cooling fan and holes in the housing, which in the case of unmanned aerial vehicles can lead to moisture penetration inside the device and its failure;

– Windows 10 Home Edition operating system has limitations in the settings.

Opportunities:

– powerful processor makes it possible to use more sophisticated data processing algorithms;

– there is a graphics card Intel HD Graphics, which can be used to implement algorithms based on convolutional neural networks.

Risks:

– small amount of built-in non-volatile memory of 32 GB creates a risk of storage overflow;

– it is hard to pick up a small, light-weight, but energy-intensive battery.

<u>Raspberry Pi 3 Model B</u>

Strengths:

– the fastest microcomputer among the Raspberry Pi series;

– a large number of ports for connecting external devices.

Weaknesses:

– sufficiently high power consumption requires power from a more capacious and heavy battery, which adds mass to the mobile device being developed.

Opportunities:

– it is possible to use more sophisticated data processing algorithms in comparison with Raspberry Pi Zero;

– it is possible to use convolutional neural networks to recognize underlying surfaces.

Risks:

– width of 5.7 cm (Table 1, line 4) is closely approaches the maximum size of 6 cm (section 1, requirement 2);

–as practice shows, with high loads on the CPU and/or high ambient temperatures, there is a risk of processor overheating and system shutdown.

Raspberry Pi Zero Wireless

Strengths:

–low price (10-15 USD);

–extremely small weight and size parameters (9 grams and 6.5 cm in maximum dimension);

–low power consumption.

Weaknesses:

–a small number of ports for connecting external devices;

–relatively low productivity.

Opportunities:

–designed specifically for embedded mobile applications.

Risks:

–there is a risk of too slow data processing, especially if use resource-intensive algorithms, and poorly designed software architecture.

Xiaomi Redmi Note 4

Strengths:

–there are all necessary elements in one housing to develop systems for image recognition;

–the most powerful CPU from the four microcomputers tested;

–the lowest power consumption in standby mode;

–relatively low power consumption for large computational loads;

–presence of a metal housing that protects against mechanical and electromagnetic influences;

–sufficiently small size, especially the thickness.

Weaknesses:

–complexity or inability to add or remove hardware components of the device;

–weak support of Python programming language for Android operating system.

–relatively large weight of 165 grams against limit of 300 grams.

Opportunities:

–it is possible to develop software in various programming languages, including C/C++, Java, Kotlin, .NET C #, etc.;

–smartphone is equipped with a large number of built-in sensors that can be used to develop a sophisticated mobile solution.

Risks:

–there is a risk that the software development under the MIUI operating system, based on Android OS, will somehow differ from the “classic” development for Android

–developers have lack of experience in software development for Android operating system.

8. Conclusions. The data obtained from the comparative analysis of computing microplatforms allowed to conclude that the best microcomputer for a portable micromodule of on-board image classification is the Raspberry Pi Zero Wireless model with the Raspbian operating system. This model has minimal energy consumption, size and weight, but it has sufficient processing power to solve the problem of recognition and classification of underlying surfaces.

There are microcomputers with faster CPU and GPU, larger RAM and built-in memory and with different number of standard interfaces which Raspberry Pi Zero Wireless doesn't have, but they do not meet the existing technical requirements for the development of the micromodule. Nevertheless, it should be noted the advantages of smartphones, which, according to the authors view, are the best microcomputers and embody all the latest advances of embedded systems and are commercially available at the same time. The main drawback of smartphones, in our case, is the big weight that does not allow to use them on board of budget drones with low load-carrying capacity.

Gratitude. This work was carried out with the financial support of project No. 3.2.4.1 of the Union State program “Technology-SG”.

References

[1]. SWOT Analysis. Discover New Opportunities, Manage and Eliminate Threats [Electronic resource] // By the Mind Tools Content Team. 2015. URL: https://www.mindtools.com/pages/article/newTMC_05.htm (access date: 15.03.2018).

[2]. Comparison of single-board computers [Electronic resource] // Wikipedia, the free encyclopedia. 2013. 08 February. Update date: 11.03.2018. URL: https://en.wikipedia.org/wiki/Comparison_of_single-board_computers (access date: 15.03.2018).

[3]. FAQs. Performance and cost considerations. How powerful is it? [Electronic resource] // Raspberry Pi Help. 2012-2018. URL: https://www.raspberrypi.org/help/faqs/#performanceSpeed (access date: 15.03.2018).

[4]. What are the best single-board computers? [Electronic resource] // Slant, the product recommendation community. 2016. Update date: 15.03.2018. URL: https://www.slant.co/topics/1629/~single-board-computers (access date: 15.03.2018).

[5]. Salas A.G., Attai W., Oyadomari K.Y., Priscal C., Schimmin R.S., Gazulla O.T. and Wolfe J.L., “PhoneSat in-flight experience results,” in Proceedings of the Small Satellites and Services Symposium. 26-30 May 2014. Porto Petro. Majorca. Spain. 2014. [Electronic resource] System requirements: Adobe Acrobat Reader. URL: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20140008616.pdf (access date: 06.04.2018).

[6]. Eder C., SMARC 2.0 - A New Standard to Bridge the Gap between Qseven and COM Express [Electronic resource] // SMARC 2.0 Whitepaper - congatec AG. 2016. System requirements: Adobe Acrobat Reader. URL: https://www.mouser.com/pdfdocs/congatec_SMARC_Whitepaper.pdf (access date: 09.04.2018).

[7]. Intel® Compute Stick STK1AW32SC STK1A32SC. Technical Product Specification [Electronic resource] // Intel Corporation. 2015. Update date: October 2017. System requirements: Adobe Acrobat Reader. URL: https://www.intel.com/content/dam/support/us/en/documents/boardsandkits/computestick/STK1AW32SC_STK1A32SC_TechProdSpec.pdf (access date: 15.03.2018).

[8]. Raspberry Pi Comparison Chart [Electronic resource] // ModMyPi LTD. 2017. 09 November. System requirements: Adobe Acrobat Reader. URL: https://www.modmypi.com/download/raspberry-pi-comparison-chart.pdf (access date: 15.03.2018).

[9]. Raspberry Pi Documentation [Electronic resource] // Written by the Raspberry Pi Foundation with community contributions. 2012-2018. URL: https://www.raspberrypi.org/documentation (access date: 15.03.2018).

[10]. Mi Phone User Guide [Electronic resource] // Xiaomi Communications Co., Ltd. 2016. 37 p. System requirements: Adobe Acrobat Reader. URL: http://files.xiaomi-mi.com/files/Mi_Phone_User_Guide/Mi_Phone_User_Guide_eng.pdf (access date: 15.03.2018).

[11]. FAQs. Performance and cost considerations. What is its operating temperature? [Electronic resource] // Raspberry Pi Help. 2012-2018. URL: https://www.raspberrypi.org/help/faqs/#performanceOperatingTemperature (access date: 15.03.2018).

[12]. Linuxium. Ubuntu flavoured 16.04 ISOs for Atom based Intel Compute Sticks [Electronic resource] // Customizing Ubuntu ISOs: Documentation and examples of how to use 'isorespin.sh': web-site. 2016. 23 May. URL: https://linuxiumcomau.blogspot.com.by/2016/05/ubuntu-flavoured-1604-isos-for-atom.html (access date: 15.03.2018).

[13]. Harrington W. Learning Raspbian. Packt Publishing Ltd. ISBN 978-1-78439-219-2. 2015. 154 p.

[14]. Android [Electronic resource] // The Python Wiki. Update date: 10.01.2018. URL: https://wiki.python.org/moin/Android (access date: 15.03.2018).

[15]. The Termux Wiki [Electronic resource] Update date: 14.12.2017. URL: https://wiki.termux.com/wiki/Main_Page (access date: 15.03.2018).

[16]. Kovalev V., Volmer S. Color co-occurrence descriptors for querying-by-example, Int. Conf. on Multimedia Modeling, Oct. 12-15, Lausanne, Switzerland, IEEE Comp. Society Press, pp. 32-38, 1998 (best paper award).

[17]. QPython - Python on Android [Electronic resource] // QuseitLab, a Mobile Internet Startup. 2017. URL: http://www.qpython.com (access date: 15.03.2018).