ADVANCES IN CYBER-PHYSICAL SYSTEMS

Vol. 5, No. 1, 2020

A SOFTWARE SERVICE FOR THE GARBAGE TYPE RECOGNITION BASED ON THE MOBILE COMPUTING DEVICES WITH GRAPHICAL DATA INPUT

Ruslan Bachynskyy, Oleksii Chaku, Nataliia Huzynets

Lviv Polytechnic National University, 12, Bandera Str, Lviv, 79013, Ukraine.
Authors' e-mail*: bac_ruslan@ukr.net, chakualex4@gmail.com, natdemchuk@ukr.net*

Submitted on 01.08.2020

***Abstract*: The article describes problems of determining the type and automatic sorting of household waste using mobile computing devices. All of the required hardware and partially software, required for implementation of this service, are already present in modern smartphones. iOS and Apple products were selected as the base for the service, due to such advantages over competitors: dual or triple depth camera (TDCS), powerful GPU, Neural Engine coprocessor, high autonomy (2750mAh battery size), sensors that allow for user positioning and navigation in space (GPS, Glonass, Gyroscope) and most important feature is possibility of cross-platform designing, suitable for iOS and macOS (Project Catalina). The recognition process consists of several phases, including capturing of graphic image and detecting the object shape, shape analysis, computing the results, and saving new associations to the database. The analysis itself is implemented using a neural network that is able to learn during its operation. Initially, the algorithm is driven by the selection of photographs with a certain type for the base set of associations, each subsequent scan improves accuracy. Cross-platforming plays a very important role - it allows us to develop a single software service that is initially run on a macOS-based computer for faster learning and then can be easily used on an iOS mobile device. After identifying a particular type of garbage, the route to the nearest recycling point of such type of garbage will be proposed for user or user's clarification will be requested. User can also manually browse categories and related items, manually search by name of item, and view locations for sorting and recycling in appropriate city. When a completely unknown object arrives, it is possible to refine the information in order to help further learning of the network.**

***Index Terms*: neural network, cross-platform (iOS/macOS), image analysis, depth sensor, GPS, accelerometer.**

I. INTRODUCTION

In modern world, mobile computing devices have reached a very high level in its computational and photographic capabilities. High quality of output images allows to accurately capture the color and detail of an object, including the smallest text on it. Also, some manufacturers are adding separate sensors to track different optical effects on photos, whether it's focusing of a laser or RGB sensors for more accurate color rendering, or different attempts to get portraits with a blurred background like on SLR cameras.

The computing power, the amount of RAM and the amount of storage on smartphones are growing rapidly too. Because of the great interest of end users in capturing photos and editing them before posting on social networks, playing mobile games with high frame rate, big manufacturers are paying even more efforts and engineering time into developing and improving the graphics core of phone processors.

Although the development of mobile devices is such promising and rapid, amount of waste in the world is increasing rapidly as well. Garbage isn't only taking excessive space but also is poisoning our planet.

Therefore, it is important to use the technologies with benefits for our habitats.

II. RELEVANCE OF RESEARCH

The criticality of the problem of garbage disposal is very different depending on the country – in some countries it's totally centralized and the process is controlled by government, in others – it's only activities of social organizations or responsible individuals. Unfortunately, our country – Ukraine – in the second group and such issues with garbage sorting are solved by activists and volunteers.

There are a few small private recycling plants for a particular type of garbage. For a regular person, not involved in different eco trends, it's relatively hard to sort trash by appropriate types and then find a place where it can be recycled. Basically, there is no centralized place where users can find information about types, some help with sorting and map with locations for all of the recycling facilities. So, as a result of poor information accessibility, no motivation from the government, a lot of people don't really care about the environment and recycling of resources.

III. FORMULATION OF THE TASK

Mobile application for Apple smartphones should be built, it uses smartphone's hardware (camera for capturing objects and then using neural network for determining appropriate type of garbage) for navigating user to the nearest recycling plants for different types of household waste. German classification of garbage types, which contains seven different categories of household waste, could be used as a base.

Let's take a closer look at German classification:
plastic (orange)
electronic devices (red)
organic, etc.(black)
paper (blue)
glass (green)
metal (yellow)
hazardous garbage (purple) – medicines, batteries, mercury thermometers, etc.

Two additional tech types of garbage are added, for development purposes and ability to enhance its functionality:

complex object (blue) – should be separated into small part or there is more than one object in the shot;

unknown object (gray) – a neural network was unable to determine garbage type and manual clarification from user is needed.

IV. TASK IMPLEMENTATION

The software system consists of several main modules which are responsible for the various stages of processing the input data and subsequent actions. The structure and interconnection of modules are shown in (Fig. 1). The system includes a module for processing device's camera data, analytics module, as well as modules that are responsible for user interface and navigation.

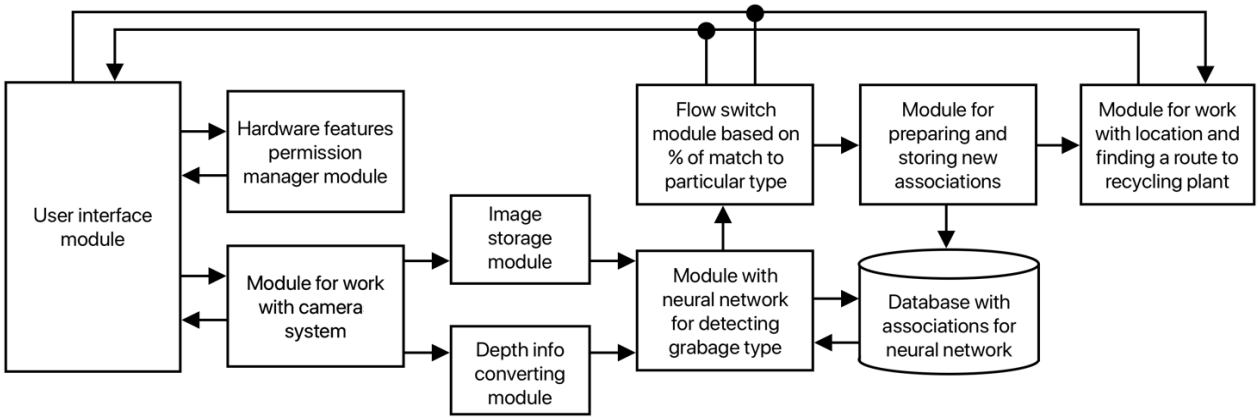

Fig. 1. Structural diagram of application's modules

Technically, the application is written using Swift 5.0 programming language and Xcode [7,10] development environment. MVP (Model – View - Presenter) was used as the architectural template which allows to separate the user interface from the core logic. A coordinator template was also used for better controlling the current state and transitions between different interface screens. Apple's smartphones and iOS operating system were chosen due to such advantages:

Dual or Triple Camera System (TDCS) – A graphical environmental sensing system found in the latest iPhone models, starting from 2017. Developers are able not only to represent data itself, but this system can capture pixel's depth also. From this depth data, we can get an object shape with 95% of accuracy.

GPU and Neural Engine – for our system it's very important to have graphics processor powerful enough in order to greatly accelerate the process of capturing and processing graphical images. And not less important role is played by the coprocessor "Neural Engine" inside main CPU. It is specialized on the processing of big amount of data and processing some math algorithms, often used in neural networks.

Autonomy – the latest devices have batteries ranging in size from 2750 mAh to 3500 mAh. This feature together with an efficient use of resources and proper implementation of application, will allow final users to use smartphones as a sorting tool for a complete 8-hour day.

Navigation (GPS) – to maximize the benefits of our service, it's needed to build a route to the nearest recycling plant and constantly update pointers on the application UI, regarding to the user's current location.

Cross-platform (Project Catalina) – one of Apple's latest innovations that were announced in 2018 and now is available for developers. It allows to develop and launch the same application on both iOS-enabled mobile devices and on powerful macOS computers [4,5].

Let's take a closer look at the neural network that was used and all its technical specifications.

In order to make development process a bit easier and using full hardware acceleration, the native Core ML framework [8,9] was used. Implementing a neural network can be classified like that:

By class: network with straight spread

By way to input data: sending signals to the output of input neurons

By way to output data: additive capture from synapses of output neurons

By education type: network with teacher (supervised neural network)

By education organization process: output training (includes input vector and output vector corresponding to input)

The algorithm of the graphical data analytics module uses the output of the digital data processing module, which includes a graphical image of an object cut from the background and a mathematical form of the object shape with its hash sum for quick comparison [1, 2, 3]. In the analysis phase, the neural network, is using stored associative rules from the database and tries to determine, by means of primitive receptors, which of the types of debris has the most in common and which can be discarded at once. There are 9 different types with color coded description. The next step is run again to clarify the correctness of the selected category in smaller pull of input data. [6] (Fig. 2).

If the number of reacting elements is greater than 1 in the last step or none of them have great percentage advantage over the others, then the neural network can't fully guarantee the exact result and asks the user to select one of the most relevant ones. The system then saves the new association to the database and will use it for further comparisons.

The following diagram shows a block diagram of the functioning of the software system. The current state and transitions between screens depend on the results of the analysis algorithms and the user's decisions (where required). (Fig. 3)

Let's look how user interface of application is implemented. User starts from home screen in Fig. 4, user can transit to screen with auto-recognition functionality in Fig. 5 or go inside of any specific category. After successful recognition, full screen result is opened in Fig. 6. User can also go to the screen with the map shown in Fig. 7.

Below there are some more screenshots for other types of garbage (Fig. 8, Fig. 9 and garbage in Fig. 10 and Fig. 11).

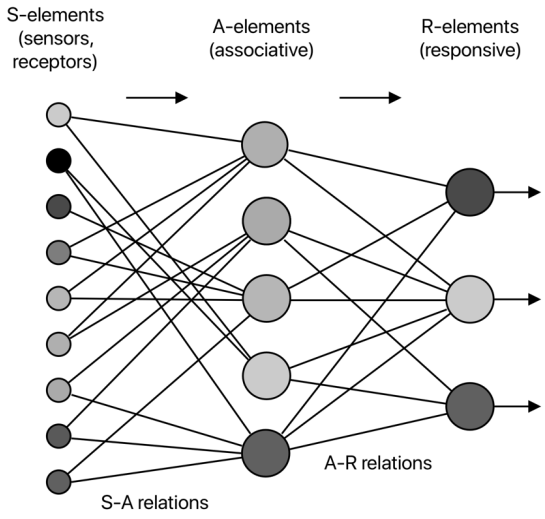

Fig. 2. A neural network algorithm using primitive receptors and more complex associative ones to determine the end result

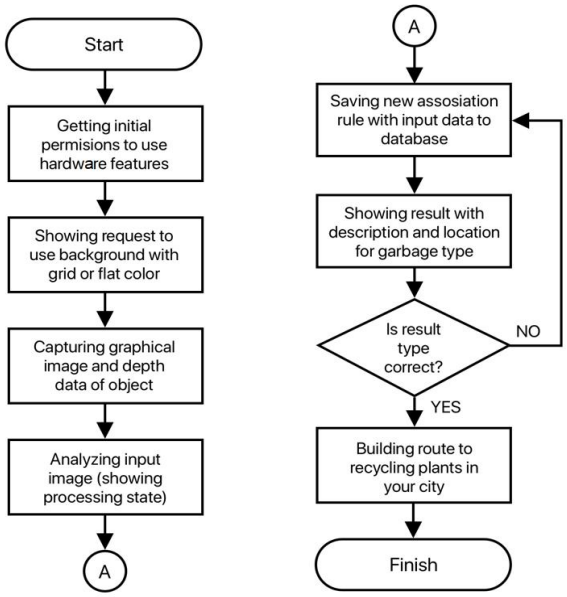

Fig. 3. The algorithm diagram

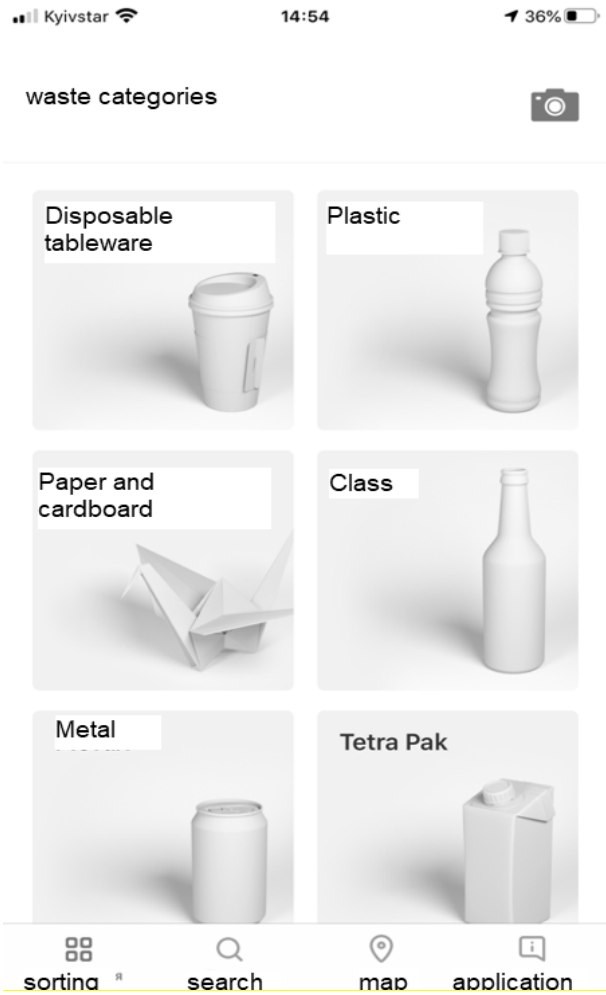

Fig. 4. Home screen of the application

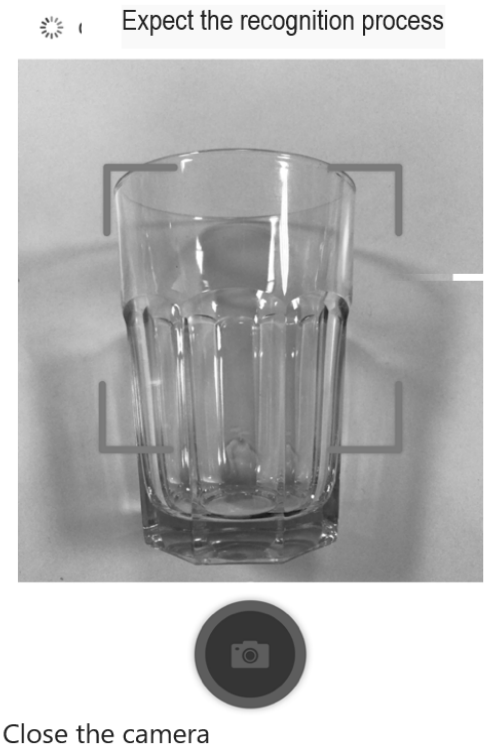

Fig. 5. Auto-Recognition Screen (glass)

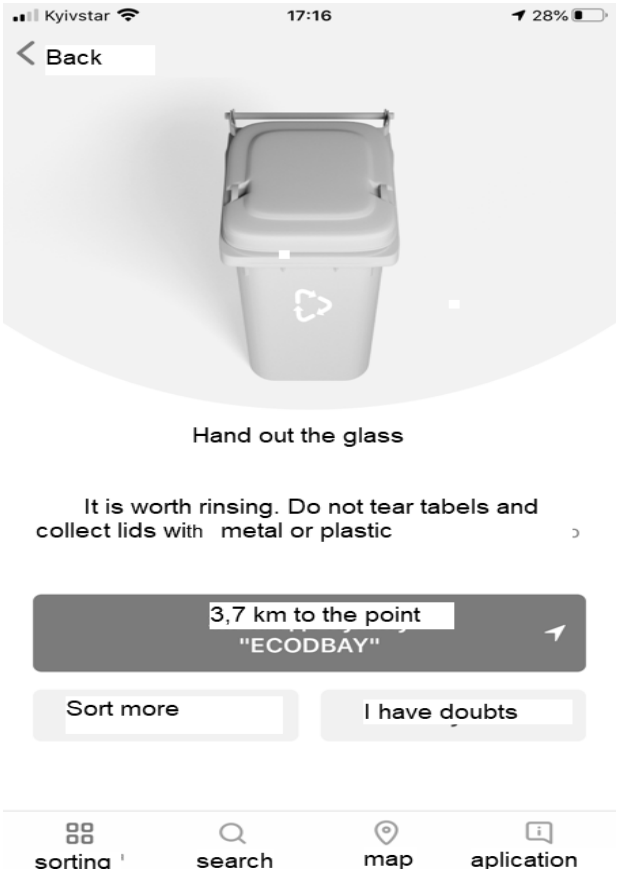

Fig. 6. Result type-specific screen (glass)

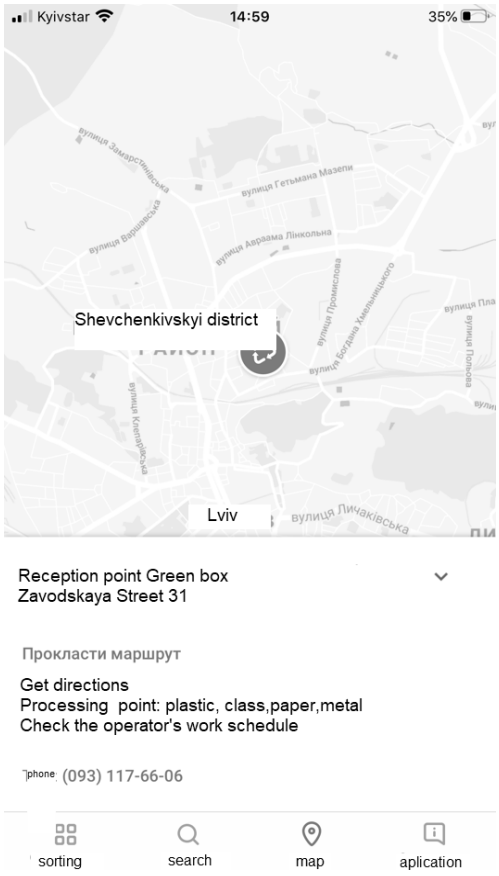

Fig. 7. Map with recycling plants

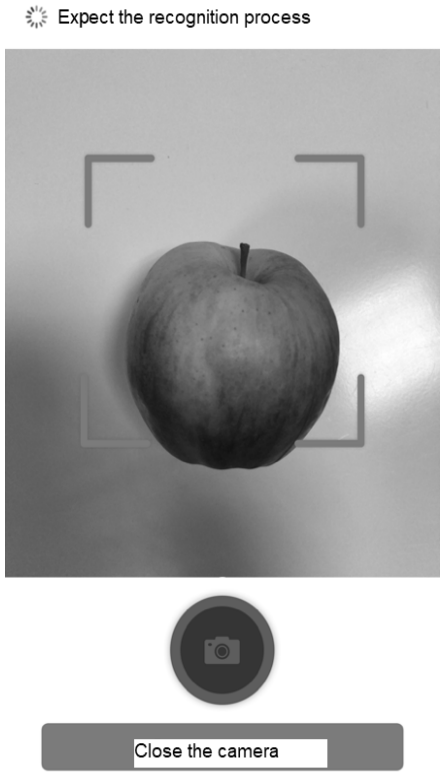

Fig. 8. Auto-Recognition Screen (organics)

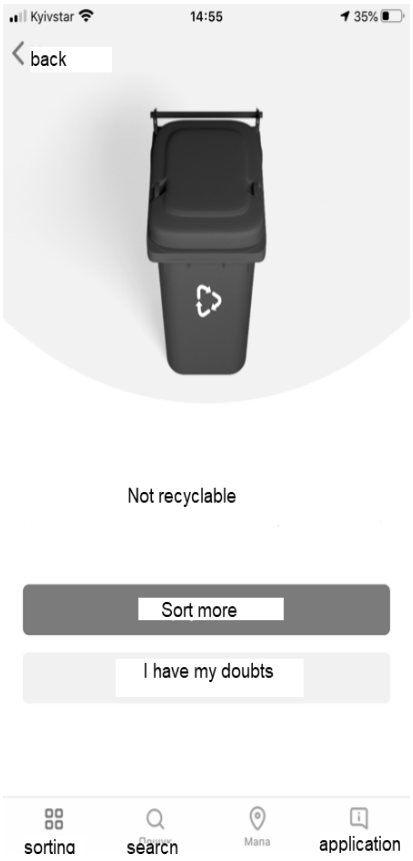

Fig. 9. Result type-specific screen (organics or other)

Fig. 10. Auto-Recognition Screen (metal)

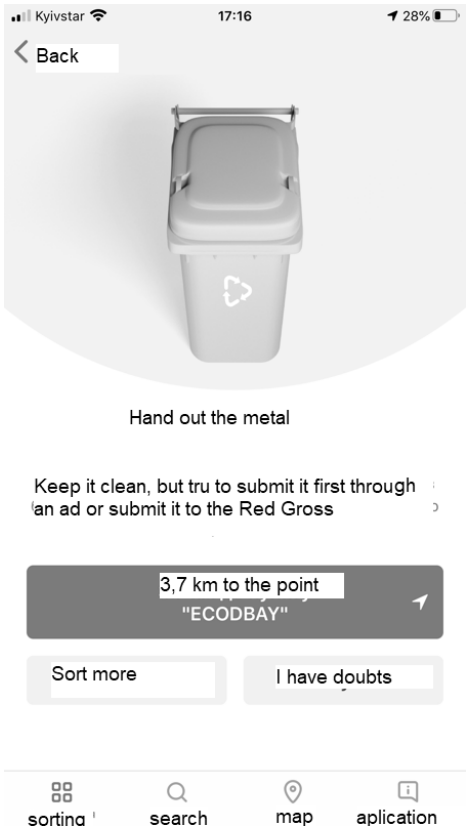

Fig. 11. Result type-specific screen (metal or other)

V. PROSPECTS FOR FURTHER RESEARCH

Even with the current implementation there are drawbacks and space for potential improvements. A short list of possible improvements in the future:

The process of learning model for neural network requires considerable hardware resources. In addition, in the current implementation of the learning it is implemented on a more powerful computer and the mobile application only uses the results of that training. So, it would be better to turn it all into a cloud solution where a single model is distributed to different devices and all devices will train it.

Investigate the need and possibilities of using a flash of the camera. Continuous illumination of the object may help improve the shape recognition of the object through greater brightness and contrast. But despite the potential improvement in shape recognition, this can worsen regular photography for analysis, so it's needed to experiment and investigate that question before implementing it.

A public list of recycling plants. In the current implementation, the list of potential recycling plants is stored in the configuration file locally. So, to update it, it is necessary to update configuration file in application's project and send a new formed one to the user. A public list with a webpage allows to edit this list without updating the entire application.

VI. CONCLUSIONS

This paper proposes an innovative approach to garbage sorting. To use this approach, no special equipment is required. Usual Apple's smartphones with usual iOS-based application is enough to start saving our habitats. Also, the developed algorithm can be implemented for other operating systems, only things that will vary are methods and hardware for image capturing and determining the shape of the object. The implemented software system for iOS successfully accomplishes the task and learns during its operation, that helps in further improving the quality of recognition. Built-in mapping and navigation capabilities allow users to easily and conveniently build a route to a specific recycling site for the appropriate type of garbage.

REFERENCES

[1] Bachynskyy R.V. (2006). Selection of the image compression processor structure for typical algorithms Lviv Polytechnic National University Journal. – P. 3–8.

[2] Morozov Y.V. (2018). Basic principles of PLO Apple Inc.

[3] https://learn.ztu.edu.ua/mod/page/view.php?id=7255

[4] Apple Developer Documentation (2018). Apple Inc. – https://developer.apple.com/documentation.

[5] Map Kit (2018). Apple Developer Documentation Apple Inc. – Access mode: https://developer.apple.com/reference/mapkit.

[6] Encoding and Decoding Custom Types | Apple Developer Documentation (2018). Apple Inc. – Access mode: https://developer.apple.com/documentation/foundation/archives_and_serialization/encoding_and_decoding_custom_types

[7] Generics in Swift 4/Candost Dağdeviren (2017). Access mode: https://theswiftpost.co/generics-swift-4/

[8] Overview Geocoding API Google Developers (2018). Access mode:https://developers.google.com/maps/documentation/geocoding/intro?hl=en

[9] Core ML Apple Developer Documentation (2018) Access mode:https://developer.apple.com/documentation/coreml.

[10] A deep dive into Grand Central Dispatch in Swift / Jonh Sundell – 2017 Access mode: https://www.swiftbysundell.com/posts/a-deep-dive-into-grand-central-dispatch-in-swift

Ruslan Bachynskyy was born in Lviv, Ukraine. He obtained his Ph.D. degree in Computer Sciences at Lviv Polytechnic National University in 2008

Oleksii Chaku, student of Lviv Polytechnic National University. He graduated from Lviv Polytechnic National University in 2019 with master degree.

Nataliia Huzynets, an engineer at Lviv Polytechnic National University. She graduated from Lviv Polytechnic National University in 2002 with specialist degree. Areas of interest: programming, High Performance computing, Digital signal processing.