

## РОЗДІЛ III. ІНФОРМАЦІЙНО-КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ

УДК 004.4

DOI 10.25140/2411-5363-2017-4(10)-94-100

Володимир Казимир, Денис Юрченко

### МЕТОД ДЕТЕКТУВАННЯ РУХУ ДЛЯ НЕСТАТИЧНИХ КАМЕР АБО ДЛЯ КАМЕР З ФУНКЦІЄЮ PTZ

**Актуальність теми дослідження.** Сучасні тенденції розвитку систем відеоспостереження орієнтовані на створення автоматичних функціональних комплексів, що володіють інтелектом. Виявлення областей руху в кадрі є фундаментальною проблемою цих систем. Тому існує потреба створення програмного забезпечення, яке виявляє зони руху в кадрі.

**Постановка проблеми.** У процесі розроблення таких систем розробникам доводиться вирішувати суперечність між обчислювальною складністю алгоритмів обробки, якістю створюваних зображень і апаратними можливостями сучасної обчислювальної техніки. Багатовимірний характер відеоінформації (координати, колір) накладає значні обмеження на швидкість і якість її обробки. Крім того, виявлення руху на мобільних камерах ускладнюється через постійну зміну фону.

**Аналіз останніх досліджень і публікацій.** Були розглянуті останні публікації у відкритому доступі, включаючи технологію оптичного потоку та алгоритм SURF.

**Виділення не вирішених раніше частин загальної проблеми.** Підвищення швидкодії існуючих методів детектування зон руху по відеоряду низької якості.

**Постановка завдання.** Удосконалення методу детектування руху за допомогою нестатичних камер або камер з функцією PTZ у напрямку прискорення його роботи.

**Виклад основного матеріалу.** Виявлення зон руху в кадрі відбувається за допомогою стандартних бібліотек *opencv*. Для визначення зони руху використовується пара послідовних кадрів: попередній – об'єкт, поточний – сцена. Формування контурів об'єктів відбувається шляхом побудови відповідних гісторам.

**Висновки.** Запропонований метод дозволяє ввічлі прискорити процес виявлення зон руху по даним відеоряду нестатичних камер.

**Ключові слова:** виявлення руху; нестатичні камери; *opencv*; гістограма.

Рис.: 8. Бібл.: 15.

**Постановка проблеми.** Сучасні тенденції розвитку систем відеоспостереження орієнтовані на створення автоматичних функціональних комплексів, що володіють інтелектом. Виявлення областей руху в кадрі є фундаментальною проблемою цих систем, особливо у випадку використання нестатичних камер [1] або камер із функцією PTZ. Вирішення цієї проблеми сприятиме подальшому стеженню за об'єктами: визначанню параметрів їх руху, підрахунку кількості рухомих об'єктів у кадрі, встановленню їх взаємного розташування, а у багатьох випадках, у поєднанні з іншими методами, забезпечить економію обчислювальних ресурсів.

У деяких випадках виявлення руху може виконувати і роль допоміжного механізму. Так, наприклад, розпізнавання автомобільних номерів може відбуватися тільки для рухомих об'єктів у кадрі.

**Аналіз останніх досліджень і публікацій.** Проблема виявлення областей руху детально описана у двох відомих методах [2; 3]. Перший з них для детектування руху використовує технологію «оптичного потоку». Оптичний потік [4] – зображення видимого руху, що являє собою зсув кожної точки між двома зображеннями. По суті, він являє собою поле швидкостей, оскільки зрушення, з точністю до масштабу, еквівалентно миттєвій швидкості. Ця технологія привела до значних результатів в оптимізації алгоритму виявлення рухомих об'єктів, що дозволило використовувати її на мобільних пристроях. Другий метод використовує алгоритм SURF. Цей алгоритм дозволяє знаходити потрібний об'єкт у кадрі навіть тоді, коли він змінює розмір і нахилений під будь-яким кутом. Цей метод успішно використовувався для визначення рухомих областей на відеоряді, знятому за допомогою БПЛА, у тому числі він дає можливість передбачити положення рухомого об'єкта в наступному кадрі відеоряду [5].

Слід зазначити, що в розглянутих роботах як джерело відео виступає відео низької якості, що знижує надійність виявлення об'єктів руху. Крім того, в них пропонується використовувати алгоритм пошуку контурів (`findcontours opencv`), що знижує загальну швидкодію методів розпізнавання.

**Метою статті** є опис удосконаленого методу детектування руху за допомогою нестатичної камери, який базується на пошуку ключових точок та дозволяє усунути недоліки існуючих підходів. Особлива увага приділяється зниженню обчислювальних витрат з метою подальшого використання цього методу у вбудованих системах комп'ютерного зору.

**Метод детектування руху.** Роботу методу можна продемонструвати на прикладі двох кадрів з відеоряду, які представлені на рис. 1.



Рис. 1. Приклади кадрів з відеоряду:  
а – сцена; б – об'єкт

Для виявлення руху використовуються два кадри з відеоряду: попередній кадр називається «об'єктом», а поточний – «сценою».

Особливу роль відіграють ключові точки – це точки зображення, що задовольняють певним властивостям:

- визначеність – особливість повинна виділятися на тлі сусідніх точок;
- стійкість – зміна яскравості, контрасту і колірної гами не повинні впливати на місце особливої точки на об'єкті або сцені;
- стабільність – зашумленість зображення, що не перевищує певний поріг, не повинна впливати на роботу детектора щодо цієї точки;
- інтерпретованість – особливі точки повинні бути представлені у форматі, придатному для подальшої обробки.

У загальному вигляді запропонований метод детектування областей руху на кадрі може бути представлений у вигляді алгоритму, наведеному на рис. 2.

Розглянемо більш детально окремі кроки цього алгоритму.

**Пошук ключових точок.** На цьому кроці використовується дескриптор Speeded up Robust Features (SURF) [6]. Він відноситься до числа тих дескрипторів, які одночасно виконують пошук особливих точок і будують їх опис, інваріантний до зміни масштабу та обертання. Крім того, пошук ключових точок є інваріантним також у тому сенсі, що нахилений об'єкт має той же набір особливих точок, що і сцена [7]. З огляду на те, що камери можуть бути встановлені на різних засобах пересування, таких як вертольоти і БПЛА, цей дескриптор підходить для вирішення цього завдання якнайкраще. У `opencv` вже є інструмент, який здатний знаходити ключові точки методом SURF – це метод `SurfFeatureDetector`[8]. Цьому методу передаються як параметри саме зображення і посилення на масив точок, в який буде зроблений запис знайдених ключових точок.

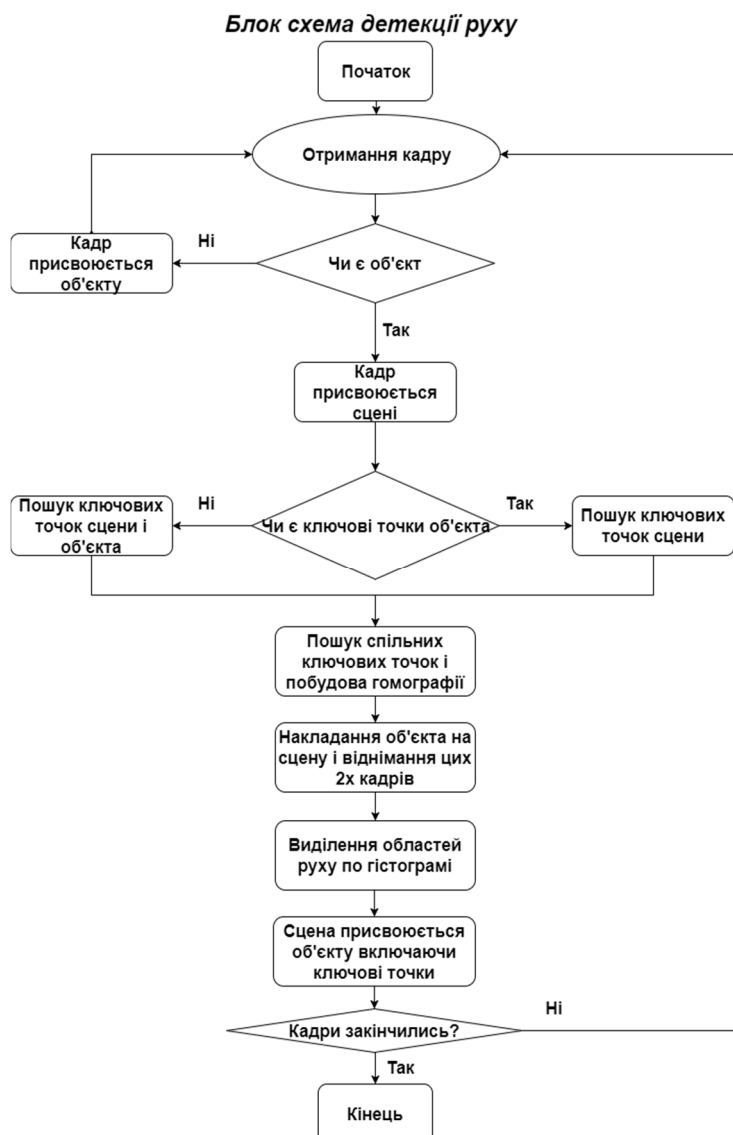


Рис. 2. Блок-схема детектування руху

**Пошук спільних точок.** На цьому кроці стоїть завдання пошуку спільних ключових точок з метою знаходження позиції об'єкта на сцені для подальшого його накладення на поточний кадр. Спільні точки – це ті ключові точки які присутні на сцені і на об'єкті.

Відповідності між дескрипторами знаходяться шляхом порівняння набору дескрипторів з першого зображення з набором дескрипторів з другого зображення. Для такого порівняння пропонується використовувати відомий метод BFMatcher [9]. Результат цього кроку наведено на рис. 3.

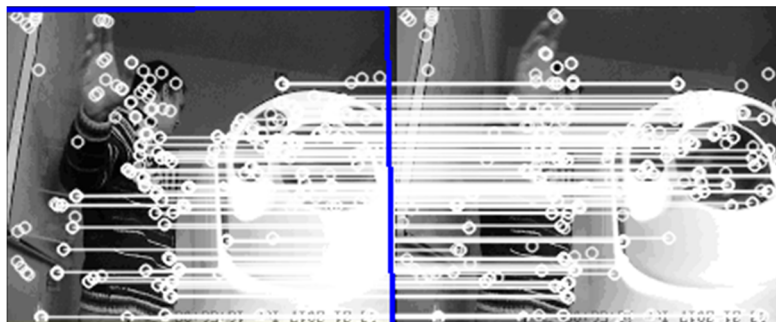


Рис. 3. Знаходження позиції об'єкта відносно сцени

**Накладення попереднього кадру на поточний.** Результати попереднього кроку використовуються для побудови гомографії.

Використовуючи набір ключових точок на цільовому об'єкті і зіставлений йому набір ключових точок у сцені, знаходиться відповідність між ними у вигляді матриці гомографії  $H$ . Знаходження позиції об'єкта на сцені відбувається шляхом порівняння дескрипторів, за допомогою яких ідентифікуються ключові точки. У свою чергу, дескриптори забезпечують інваріантність відповідності між точками щодо перетворень зображень. Чотири точки по краях цільового об'єкта відображаються на зображенні сцени за допомогою знайденого перетворення [9]. У результаті будуть отримані координати вершин об'єкта, на яких він має бути розташованим відносно сцени – об'єкт розміщується саме у ці межі за допомогою методу `warpPerspective` [10]. Оскільки знайдені вершини для об'єкта можуть знаходитися за межами сцени, потрібно спочатку створити тимчасове зображення з розміром, який може його вмістити, а вже потім поміщати в нього перетворений об'єкт. Розміщення об'єкта відносно сцени показано на рис. 4.



Рис. 4. Положення перетвореного об'єкта відносно сцени

**Детектування руху.** Після накладання об'єкта на сцену необхідно визначити абсолютну різницю цих двох кадрів і виявити, чи був рух у цей момент. Для цього використовується метод `absdiff` [11], який приймає на вхід два зображення і видає їх різницю у вигляді, показаному на рис. 5.



Рис. 5. Різниця об'єкта і сцени

Далі проводиться бінаризація зображення [14], а також його ерозія [15] для зменшення шумів. Це досягається шляхом використання, відповідно, методів `threshold` і `erode` [12].

На виході отримуємо результат, представлений на рис. 6.



Рис. 6. Результат бінаризації та ерозії

Ерозія дозволила позбавитися зайвих шумів, але це привело до зменшення реальної області руху. Щоб уникнути цього і відновити реальний розмір області руху був використаний метод dilate [13].

**Знаходження контурів об'єктів.** Виявлення контурів об'єктів відбувається шляхом побудови гістограм. Спочатку зображення розбивається на вертикальні смуги в товщину в 1 піксель. Далі вони перевіряються на наявність білих пікселів у ній. Якщо в смугі є білі пікселі, то призначається початкова координата ( $xS1$ ) майбутньої «обрізки». Далі у разі виявлення відсутності білих пікселів у смугі можна задати кінцеву координату «обрізки» ( $xE1$ ). Після проходження всього кадру проводиться вертикальна нарізка на смуги по отриманим координатам. Далі ці смуги обробляються таким же чином, але по горизонталі. Як результат, будуть отримані координати областей руху, показані на рис. 7.

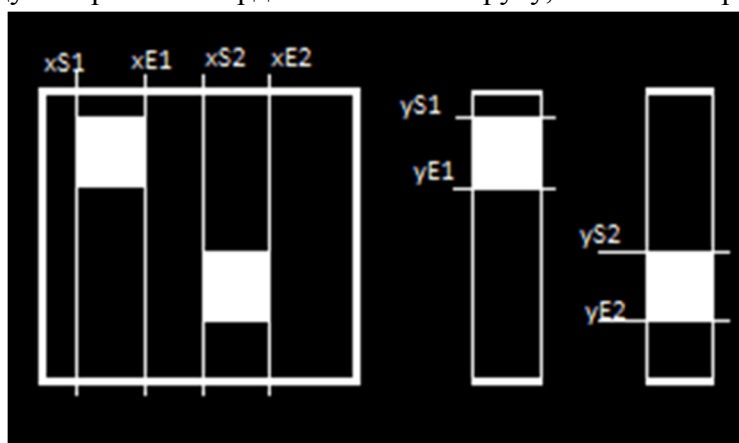


Рис. 7. Виділення областей руху з відповідними координатами

Важливо зазначити, що хоч цей метод і працює вдвічі швидше, ніж стандартний findcontours від opencv, але визначені області досить приблизні, а не точні, оскільки будуть мати контури прямокутної форми, як показано на рис. 8.

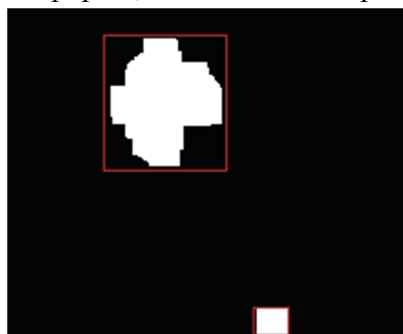


Рис. 8. Формування прямокутних контурів об'єктів

**Висновок.** Запропонований метод успішно справляється зі своїм завданням, а саме виявлення рухомих об'єктів за допомогою нестационарної камери. Використання гістограм для виявлення об'єктів дозволило прискорити його роботу – на обробку одного кадру запропонованим методом витрачається часу на 50 % менше у порівнянні зі стандартним методом, що підкреслює його ефективність, особливо у разі використання для обробки відеоданих від БПЛА.

#### Список використаних джерел

1. *Detection of Moving Objects with Non-Stationary Cameras in 5.8ms: Bringing Motion Detection to your Mobile Device* [Електронний ресурс]. – Режим доступу : <http://f4k.dieei.unict.it/proceedings/CVPR12/data/papers/workshops/4990a027.pdf>.
2. *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops* [Електронний ресурс]. – Режим доступу: <http://ieeexplore.ieee.org/document/6595847/>.

## TECHNICAL SCIENCES AND TECHNOLOGIES

3. *Moving object detection in unmanned aircraft images* [Електронний ресурс]. – Режим доступу : <http://volkansalma.blogspot.com.tr/2015/11/insansz-hava-arac-goruntulerinde.html>.
4. *То, что вы хотели знать про оптический поток, но стеснялись спросить* [Электронный ресурс]. – Режим доступа : <https://habrahabr.ru/post/201306>.
5. *Motion detection on uav videos* [Електронний ресурс]. – Режим доступу : <https://www.youtube.com/watch?v=vMnpD-6PtmE>.
6. C4.5: Programs for Machine Learning Morgan Kaufmann Publishers, San Mateo.
7. *Введение в разработку мультимедийных приложений с использованием библиотек OpenCV и IPP* [Электронный ресурс]. – Режим доступа : <http://www.intuit.ru/studies/courses/10621/1005/lecture/16983?page=2>.
8. *Feature Detection* [Електронний ресурс]. – Режим доступу : [http://docs.opencv.org/2.4/doc/tutorials/features2d/feature\\_detection/feature\\_detection.html](http://docs.opencv.org/2.4/doc/tutorials/features2d/feature_detection/feature_detection.html).
9. *Common Interfaces of Descriptor Matchers* [Електронний ресурс]. – Режим доступу : [http://docs.opencv.org/2.4/modules/features2d/doc/common\\_interfaces\\_of\\_descriptor\\_matchers.html](http://docs.opencv.org/2.4/modules/features2d/doc/common_interfaces_of_descriptor_matchers.html).
10. *Geometric Image Transformations* [Електронний ресурс]. – Режим доступу : [http://docs.opencv.org/2.4/modules/imgproc/doc/geometric\\_transformations.html](http://docs.opencv.org/2.4/modules/imgproc/doc/geometric_transformations.html).
11. *Operations on Arrays* [Електронний ресурс]. – Режим доступу : [http://docs.opencv.org/2.4/modules/core/doc/operations\\_on\\_arrays.html](http://docs.opencv.org/2.4/modules/core/doc/operations_on_arrays.html).
12. *Miscellaneous Image Transformations* [Електронний ресурс]. – Режим доступу : [http://docs.opencv.org/2.4/modules/imgproc/doc/miscellaneous\\_transformations.html?highlight=threshold#threshold](http://docs.opencv.org/2.4/modules/imgproc/doc/miscellaneous_transformations.html?highlight=threshold#threshold).
13. *Image Filtering* [Електронний ресурс]. – Режим доступу : <http://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html?highlight=dilate#cv2.dilate>.
14. *Бинаризация изображений: алгоритм Брэдли* [Электронный ресурс]. – Режим доступа : <https://habrahabr.ru/post/278435>.
15. *Morphological Transformations* [Електронний ресурс]. – Режим доступу : [https://docs.opencv.org/3.0-beta/doc/py\\_tutorials/py\\_imgproc/py\\_morphological\\_ops/py\\_morphological\\_ops.html](https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_morphological_ops/py_morphological_ops.html).

### References

1. *Detection of Moving Objects with Non-Stationary Cameras in 5.8ms: Bringing Motion Detection to your Mobile Device*. Retrieved from <http://f4k.dieei.unict.it/proceedings/CVPR12/data/papers/workshops/4990a027.pdf>.
2. *2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops*. Retrieved from <http://ieeexplore.ieee.org/document/6595847>.
3. *Moving object detection in unmanned aircraft images*. Retrieved from <http://volkansalma.blogspot.com.tr/2015/11/insansz-hava-arac-goruntulerinde.html>.
4. *To, chto vy khoteli znat pro opticheskii potok, no stesnialis sprosit [What you wanted to know about the optical flow, but were shy to ask]*. Retrieved from <https://habrahabr.ru/post/201406>.
5. *Motion detection on uav videos*. Retrieved from <https://www.youtube.com/watch?v=vMnpD-6PtmE>.
6. C4.5: Programs for Machine Learning Morgan Kaufmann Publishers, San Mateo.
7. *Vvedenie v razrabotku multimediynyih prilozheniy s ispolzovaniem bibliotek OpenCV i IPP [Introduction to the development of multimedia applications using libraries]*. Retrieved from <http://www.intuit.ru/studies/courses/10621/1005/lecture/16983?page=2/>.
8. *Feature Detection*. Retrieved from [http://docs.opencv.org/2.4/doc/tutorials/features2d/feature\\_detection/feature\\_detection.html](http://docs.opencv.org/2.4/doc/tutorials/features2d/feature_detection/feature_detection.html).
9. *Common Interfaces of Descriptor Matchers*. Retrieved from [http://docs.opencv.org/2.4/modules/features2d/doc/common\\_interfaces\\_of\\_descriptor\\_matchers.html](http://docs.opencv.org/2.4/modules/features2d/doc/common_interfaces_of_descriptor_matchers.html).
10. *Geometric Image Transformations*. Retrieved from [http://docs.opencv.org/2.4/modules/imgproc/doc/geometric\\_transformations.html](http://docs.opencv.org/2.4/modules/imgproc/doc/geometric_transformations.html).
11. *Operations on Arrays*. Retrieved from [http://docs.opencv.org/2.4/modules/core/doc/operations\\_on\\_arrays.html](http://docs.opencv.org/2.4/modules/core/doc/operations_on_arrays.html).
12. *Miscellaneous Image Transformations*. Retrieved from [http://docs.opencv.org/2.4/modules/imgproc/doc/miscellaneous\\_transformations.html?highlight=threshold#threshold](http://docs.opencv.org/2.4/modules/imgproc/doc/miscellaneous_transformations.html?highlight=threshold#threshold).

13. *Image Filtering*. Retrieved from <http://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html?highlight=dilate#cv2.dilate>.

14. *Binarizatsiya izobrazheniy: algoritm Bredli [Binarization of images: the Bradley algorithm]* (2016). Retrieved from <https://habrahabr.ru/post/278435>.

15. *Morphological Transformations*. Retrieved from [https://docs.opencv.org/3.0-beta/doc/py\\_tutorials/py\\_imgproc/py\\_morphological\\_ops/py\\_morphological\\_ops.html](https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_morphological_ops/py_morphological_ops.html).

UDC 004.4

*Volodymyr Kazymyr, Denis Yurchenko*

## THE METHOD OF MOTION DETECTION FOR MOBILE CAMERAS OR FOR CAMERAS WITH PTZ FUNCTION

**Urgency of the research.** Modern trends in the development of video surveillance systems focus on the creation of automatic functional complexes with intelligence. Detection of the areas of motion in a frame is a fundamental problem of these systems. Therefore, there is a need to create software that detects areas of motion in the frame.

**Target setting.** By developing such systems, developers have to solve the contradiction between the computational complexity of processing algorithms, the quality of images created and the hardware capabilities of modern computing. The multidimensional nature of video information (coordinates, color) imposes significant limitations on the speed and quality of its processing. In addition, motion detection on mobile cameras is complicated due to constant background change.

**Actual scientific researches and issues analysis.** The latest open access publications, including optical streaming technology and the SURF algorithm, were considered.

**Uninvestigated parts of general matters defining.** Increasing the speed of existing methods of detecting traffic zones on the video of low quality.

**The research objective.** Improvement of the method of detecting motion using non-static cameras or cameras with PTZ function in the direction of accelerating its operation.

**The statement of basic materials.** Detection zone of motion in the scene comes with standard libraries openCV. To determine the zone of motion, a pair of sequential frames are used: the previous one - the object, the current one - the scene. Forming contours of objects occurs by constructing the corresponding histograms.

**Conclusions.** The proposed method allows twice accelerating the process of detecting traffic zones by the given video of non-static cameras.

**Key words:** motion detection; non-static camera; openCV; histogram.

Fig.: 8. Bibl.: 15.

УДК 004.4

*Владимир Казимир, Денис Юрченко*

## МЕТОД ДЕТЕКЦИИ ДВИЖЕНИЯ ДЛЯ НЕСТАТИЧНЫХ КАМЕР ИЛИ ДЛЯ КАМЕР С ФУНКЦИЕЙ PTZ

В статье описан усовершенствованный метод надежного и быстрого обнаружения движения объектов по данным видеоряда. Для первичной обработки изображений предлагается использовать стандартные библиотеки *opencv*, встроенные в программное обеспечение динамических систем видеонаблюдения. Поиск контуров объектов осуществляется путем построения гистограмм контрастных изображений выделенных областей движения.

**Ключевые слова:** обнаружения движения; нестатические камеры; *opencv*; гистограмма.

Рис.: 8. Библ.: 15.

**Казимир Володимир Вікторович** – доктор технічних наук, професор, проректор з наукової роботи, Чернігівський національний технологічний університет (вул. Шевченка 95, м. Чернігів, 14027, Україна).

**Казимир Владимир Викторович** – доктор технических наук, профессор, проректор по научной работе, Черниговский национальный технологический университет (ул. Шевченко 95, г. Чернигов, 14027, Украина).

**Kazymyr Volodymyr** – Doctor of Technical Sciences, professor, Vice-Rector for scientific work, Chernihiv National University of Technology (95 Shevchenka Str., 14027 Chernihiv, Ukraine).

**E-mail:** [vvkazymyr@gmail.com](mailto:vvkazymyr@gmail.com)

**ORCID:** <http://orcid.org/0000-0001-8163-1119>

**ResearcherID:** Q-2925-2016

**Юрченко Денис Юрійович** – аспірант, асистент кафедри ІКС, Чернігівський національний технологічний університет (вул. Шевченка, 95, м. Чернігів, 14027, Україна).

**Юрченко Денис Юрьевич** – аспирант, ассистент кафедры ИКС, Черниговский национальный технологический университет (ул. Шевченко, 95, г. Чернигов, 14027, Украина).

**Yurchenko Denis** – Phd student, assistant of Department of IKS, Chernihiv National University of Technology (95 Shevchenka Str., 14027 Chernihiv, Ukraine).