ADVANCES IN CYBER-PHYSICAL SYSTEMS
Vol. 8 No. 2, 2023

DEVELOPMENT OF AN ALGORITHM AND SOFTWARE SYSTEM FOR FACING PANELS ACCOUNTING ON PRODUCTION LINES

Yurii Ivanov, Petro Bilous, Viacheslav Botvinnikov, Maksym Holovatyi

Lviv Polytechnic National University, 12, Bandera Str, Lviv, 79013, Ukraine.
Authors' e-mail: *viacheslav.botvinnikov.mkiks.2022@lpnu.ua*

https://doi.org/10.23939/acps2023.02.089

Submitted on 10.10.2023

Abstract: **This paper aims to develop and implement an algorithm and an automated software system for the automatic accounting process of external facing panels during transportation on line conveyors. The method described in this paper is designed to simplify the process of production and accounting of wall-facing panels. This method can also serve as a model for implementing other manufacturers. The developed algorithm consists of the following steps: obtaining a video stream in real-time or from a file and its targeted processing and determining the number of moving objects of interest. The software accounting system created based on the developed algorithm analyzes the video data and stores all the necessary results and settings in the database. The software system can adapt to the accounting requirements of other types of similar products in other areas.**

***Index Terms*: Algorithm, computer vision, moving object, production line, video stream.**

I. INTRODUCTION

Many production enterprises use modern digital technologies in their production processes, constantly modernizing their production lines at various stages. Firstly, it involves installing new modern equipment, which in most cases is connected to a computer and has software products to facilitate production and improve end-product quality. Secondly, another digitalization factor, which often increases the production speed and quality of the end products, is the minimization of human factors.

During the recent research publications analysis on this work, materials were considered based on the available technologies for implementing a software and hardware complex designed for monitoring moving objects in real-time. Among the materials reviewed, the central aspect was understanding machine vision compared to human vision using neural networks. In many studies, the technology of capturing the image of a moving object has been studied using the example of human movement, and technologies that allow reading, processing, and computing video information in real-time are described.

Based on a review of scientific research and existing examples of implementation, a strategy for improving the operation of existing surveillance systems has been developed.

The system's performance can be improved using the methods described in the article, such as segmentation, image processing, and extracting factors that affect image processing accuracy, such as shadows of moving objects, micromotion, and partial illumination.

In the course of the work, the materials of the OpenCV library, which is a convenient tool for solving machine vision problems, were analyzed. The advantages and disadvantages of the library, existing analogs, and possibilities of improving the system through joint work of the library with a neural network are analyzed.

In analyzing existing publications, we identified some criteria of low accuracy of existing systems that affected software performance and technical implementation. We decided to pay attention to the accuracy and reliability of image reading to improve system performance. The central aspect of the work is to enhance the characteristics of the software system and its maximum adaptability for integration into the production of any industry.

II. RESEARCH PAPERS AND TECHNOLOGIES OVERVIEW

A. OPENCV LIBRARY

OpenCV (Opensource Computer Vision Library) is an open-source machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and the adaptive use of machine perception in commercial products.

The library includes a set of classical and modern computer vision and machine learning algorithms. OpenCV consists of more than 2500 such optimized algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in video, track camera movements, track moving objects, extract 3D object models, obtain 3D cloud points from stereo cameras, stitch images together to get a high-resolution image of an entire scene, search for similar images from image databases, correct red-eye in flash images, track eye movements, recognize landscapes, and more.

The library is used in many industries and companies, such as Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, and Toyota. OpenCV technology with advanced functionality detects intrusions in Israel's video surveillance system, checks for debris on runways in Turkey, and faces recognition in China and Japan. The technology is also used in the art industry in Spain and the United States. OpenCV has interfaces for working with many programming languages (C++ et al.) and supports many operating systems. This technology mainly works with a real-time video stream with MMX and SSE instructions. OpenCV is written in C++, which makes it easy to integrate into software implementations with any programming language [1, 2].

B. TENSORFLOW FRAMEWORK

TensorFlow is a free, open-source software library for data flow and differentiated programming for various tasks. The library can train and run deep neural networks for analyzing and classifying numbers, embedded words, image recognition, and natural language processing.

Google created the library to meet its needs for systems capable of building and training neural networks, so the library is closely related to neural networks and their machine training.

A neural network is a machine learning type modeled after the human brain. A neural network uses an algorithm to allow a computer to learn and incorporate new data. TensorFlow can run on multiple CPUs and GPUs and is available for 64-bit computing platforms, including mobile. With the help of TensorFlow technology and its machine-learning neural networks, the planet Kepler-90i was discovered, making the Kepler-90 system the only known system with eight planets near one star [3, 4].

C. HOUGH TRANSFORM

Hough transform is used in image analysis, computer vision, and digital image processing. The main focus of this technique is to find imperfect instances of objects of a specific class of shapes using a voting procedure [5].

The Hough transform was applied in the classical version by drawing lines on the analyzed image. Later, the functionality was extended to arbitrary shapes, primarily circles and ellipses.

The main task and functionality of the Hough transform is to search for simple geometric shapes (lines, circles, ellipses) in the image analysis area. In many cases, the image could be better and allow us to determine the points of the shapes with maximum accuracy, so it is not trivial to group these shapes.

The easiest way to search using this technology is to search for lines. It is carried out using the Hessian standard form:

$$r = x \cdot \cos(\theta) + y \cdot \sin(\theta). \tag{1}$$

An adjustment algorithm should be applied to detect circular shapes instead of straight lines. The algorithm consists of the following steps:

Step 1. Create a space to accumulate the cell of each pixel.

Step 2. Zoom in on all cells for contour points in the image using the expression:

$$(i-a)^2 + (j-b)^2 = r^2. \tag{2}$$

Step 3. For each possible value of a, find all possible values of b.

Step 4. Find local maxima in the space whose cells represent a circle.

D. RADON TRANSFORM

The mathematical Radon transform is an integral transform that converts a function f defined on a plane into a function Rf defined on a two-dimensional space of lines on the plane, the value of which in a specific line is equal to the curvilinear integral of this function over this line [6].

This transformation was introduced by Johann Radon, who also presented the world with the inverse transformation formula. Radon transform data is often defined as a sinogram since the transform at the center of a point source is a sinusoid. To summarize, the transformation for small objects is a certain number of blurred sinusoids with different amplitudes and phases. The initial function of this transform is one in the white area and zero in the dark area (Fig. 1).

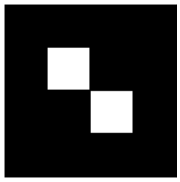

Fig. 1. The initial function

X-ray images are the most prominent application area of this transformation and are used in image processing.

E. NEURAL NETWORK

A neural network is one of the latest developments that help solve computer vision implementation issues [7]. It teaches a machine to perform a sequence of actions necessary to recognize an object from a video.

Computer vision technology began to develop in the 1980s in most enterprises. Still, a breakthrough in this technology occurred in 2012 when Jeff Heaton and Alex Krzyzewski won an object recognition competition with their algorithm. They presented their entire neural network with a recognition accuracy of 84%, a record at the time.

They created a convolutional neural network that gave rise to many modern popular technologies and algorithms with an accuracy close to 100%.

Their development was based on a model of living nerve cells (neurons). They imitate human vision when neurons do not receive signals from all receptors but only from a specific area for greater accuracy. Certain neurons start to activate when they notice vertical stripes. At the same time, others begin to activate when they see horizontal stripes, which makes it possible to cover the entire spectrum of an object.

A convolutional neural network works on the same principle but uses functions instead of living neurons. The name convolutional is derived from the mathematical function of the same name (the combination of two functions), which makes it possible to form a third function at the intersection of the previous two. Based on this principle, the neural network analyzes and processes images and finds the desired objects.

F. MOVING OBJECT TRACKING IN THE SCOPE OF REAL-TIME

Algorithms for identifying, tracking, and accounting for moving objects are now widely used in real-time video surveillance systems [8].

These algorithms allow us to predict binary-large object positions and determine their areas, contours, and other geometrical characteristics through all of the frames and build tracks of objects by using centroid positions on all previous frames where objects were visible.

The software implementation of these algorithms shows that they effectively overcome such factors as temporary partial occlusions, micro-movements, and speed changes of moving objects. The functioning of the software system is entirely autonomous, and at the same time, the accuracy of accounting is ensured to be precise for the overall number of objects. Also, the system allows to monitor conveyors' idle time, making it possible to optimize production. Accounting results are stored in the database and can be presented as histograms or data tables in the viewing module.

III. PROBLEM STATEMENT

The technology of processing statistics from video streams is a reasonably modern development and is beginning to find its application in many areas. Any production has certain production lines where quality and quantity control are carried out, and these data are stored in a database for further processing. Thanks to the technology of recognizing the necessary objects and further software processing of the received data, it opens up many opportunities for analyzing and controlling the quality of production, as well as quickly measuring the necessary characteristics without human intervention, and the possibility of online monitoring and correction of the required factors during the production process.

During the automated video surveillance systems operation, the main problem factors being improved are micromovements, different speeds of production lines, instability in geometric shapes of objects, and lighting conditions.

On the other hand, one of the complications may be the number of objects that need to be analyzed simultaneously. This factor affects the system's ability to explore all the necessary objects that fall within the system's observation area in real time.

Depending on the task set for the system and the required accuracy in the analysis process, the system may require mathematical and software iterations to select the desired areas, increase image quality detail, and analyze the search area for the desired data.

IV. RESEARCH OBJECTIVES

The main goal of this work is to develop and implement algorithms for detecting, tracking, and accounting for wall-facing panels on a conveyor line during production. Software implementation of the algorithm should consider distortions caused by production factors, such as unstable lighting conditions, micro-movements, and differences in geometric shapes. The algorithm should be implemented as a software module integrated into the existing video surveillance system as an additional video analysis module. The system should operate autonomously and process accounting with an accuracy of more than 97% of the total number of objects. Video streams should be analyzed by receiving video from cameras installed on the conveyor line and processed in real time. The system can be remotely adjusted to account for production factors and errors. After analysis, the system results should be stored in a database for quick access and processing of information during production.

V. ALGORITHM IMPLEMENTATION STAGES

The solution to the problem of moving object tracking consists of two main subtasks: moving object representation and its localization on video frames. The representation process, in turn, consists of two components: object segmentation and calculation of its geometric characteristics.

Segmentation is an image separation into several areas with common characteristics. The segmentation process requires creating and updating a background model. All subsequent stages of the algorithm and the amount of computing resources needed depend on the correctness and accuracy of the segmentation.

Segmentation methods can be split into four groups: background extraction, probabilistic methods, time difference methods, and optical flow methods.

The most effective methods for tracking moving objects are those based on contour extraction. In our case, contour extraction of the panel is the most effective method.

With the help of the contour, the desired object is localized on the video frame, and it is possible to determine the perforation lines used to connect the panels on the boundaries of the objects and take them as a basis for accounting for objects moving along the conveyor line.

Deep neural network training is also often used in such tasks; these methods allow tracking several objects at once and objects that move at a certain speed. High accuracy in detail at these stages will minimize the algorithm's incorrect operation and ensure greater accuracy in analyzing and accounting for the required objects. For software based on the OpenCV library to work, we must first upload video material and prepare it for further processing. The next step is recognizing the object and processing the data using the database. The image preparation stage includes noise removal procedures. There are many types of noise: Poisson (dependent on brightness) and Gaussian (independent of signal intensity).

The noise removal process is essential for further work with images and video materials. In the process, it is necessary to use approaches to remove noise and preserve the whole picture of the video material.

For cleaning with TV regularization, the following formula is used:

$$\min\left(\alpha\int_{\Omega}\left|\nabla_u\right|dx+\frac{1}{2}\int_{\Omega}\left|u-z\right|^2 dx\right), \quad (3)$$

where α - positive parameter that controls the level of noise removal.

Also, to clean up images from noise, we need to use the Euler-Lagrange equation for the ROF model:

$$u-z+\alpha\partial J\left(u\right). \quad (4)$$

This formula could demonstrate a staircase effect, thus limiting the image quality. Another stage of image preparation for processing is deblurring. The deblurring process consists of restoring an accurate image with no blurring. Reconstructing the actual image and restoring features after removing noise is called deconvolution.

The processing considers the overlap of one feature when it is superimposed on another feature or neighboring pixels. For the deblurring process, let us denote two functions: f(t) and w(t). The product of f(x) and w(x), denoted as g(x), is a function from $t \geq 0$ given by:

$$g(x)=w(x)\cdot f(x)=\int w(s)\cdot f(x-s)ds. \quad (5)$$

The blurred image can be modeled with the following equation:

$$z(x)=\left[h\cdot u\right](x)+\eta(x). \quad (6)$$

There are three main aspects to check in the blur removal process: non-blind deconvolution, semi-blind deconvolution, and blind image deconvolution.

VI. THE PROCESS OF MOTION DETECTION IN VIDEO FRAMES

During the analysis, the algorithm creates a pixel-by-pixel model and classifies each pixel using the expression:

$$S\sim\sum_{p=1}^{k}w_p\cdot\frac{1}{\sqrt{2\cdot\pi}\cdot\delta}\cdot e^{-\frac{(x-\mu)^2}{2\cdot\delta^2}}. \quad (7)$$

Each element in the sum (7) corresponds to a process in pixels characterized by normal distribution parameters and a weighting factor (an indicator representing how often this process is in the field of view). Additionally, it is necessary to calculate the components of the YUV color space to eliminate moving shadows.

The above expression can successfully process and analyze images, taking into account such aspects as micromovements, object speed, and proximity. These aspects are fundamental to machine vision and are necessary to consider and improve the system and its quantitative indicators. Thanks to algorithms for processing and removing factors that interfere with the system's operation, the speed and accuracy of video data processing by the system directly increases. In the analysis process, the software implementation recognizes grayscale images (Fig. 2).

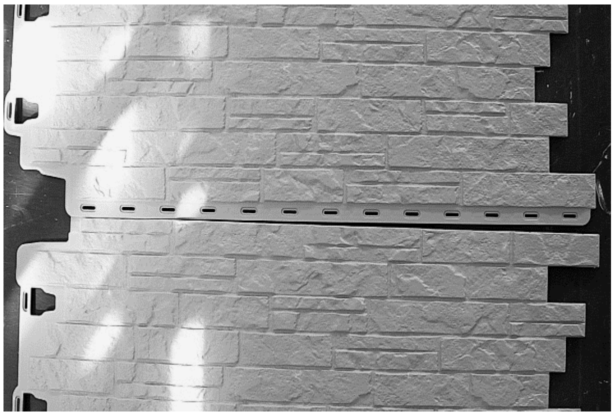

Fig. 2. Raw frame from a video surveillance camera

Image registration during its processing is a challenging task. For image registration, several methods can be divided into parametric and non-parametric. Parametric methods are rigid and depend on several parameters. Linear transformations can be defined as follows:

$$\varphi(x)=\begin{bmatrix}a_1 & a_2\\ a_4 & a_5\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\begin{bmatrix}a_3 & a_6\end{bmatrix}. \quad (8)$$

This formula corresponds to a problem with six parameters for which a reasonable choice of parameter a control will be determined:

$$R(a)=\frac{1}{2}\left(\left(1-a_1\right)^2+a_2^2+a_3^2+a_4^2+\left(1-a_5\right)^2+a_6^2\right). \quad (9)$$

The next step is to separate the desired object to be processed. For this purpose, existing libraries and frameworks facilitate this task. The main thing during the analysis is to decide on the demarcation line along which the frame will be tracked to find the desired object. After separating the desired object, we must remove the shadows and filter the image based on morphological operations.

This step is necessary because every moving object casts shadows when illuminated, and this is one of the main problems with the dynamic segmentation of moving objects. That can lead to significant shape distortion or merging of moving objects. These aspects significantly complicate further work with the localized object if they need to be removed. The basis for removing these aspects is to remove shadows by recognizing their RGB values and removing the necessary sectors or by setting the algorithm to ignore them. To do this, we need to capture one pixel with the background color characteristic for further orientation according to it, in comparison with it, to understand which sectors contain pixels that do not correspond to the object but cast shadows.

After the previous steps, the implementation requires an algorithm for tracking binary large objects based on area estimation and center prediction. This algorithm allows us to track the position and area of a moving object in all frames and build its trajectory using the position of the object's center captured in the previous frame.

VII. RESULTS REPRESENTATION AND SOFTWARE SYSTEM IMPLEMENTATION

In the last stage, processing and storing the results is necessary. To accomplish this, we need to customize the interface to manage and process the required parameters of objects conveniently, in our case, counting the required facing panels.

For the convenience of the implementation, it is necessary to design a user-friendly interface for changing the current camera for further system expansion and modernization. It is required to add options for adjusting the aspects used in the program, such as removing shadows, erosion, thresholds, or the position of the transverse line of object localization.

The image (Fig. 3) shows the preliminary window for tracking the workflow and accounting of external facing panels.

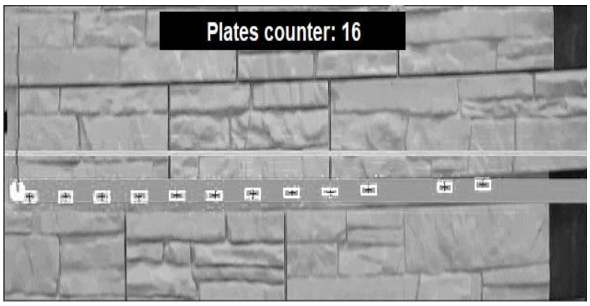

Fig. 3. Results of raw image analysis and objects of interest accounting

The developed algorithm is implemented as a video analysis software module that can be easily integrated into almost any video surveillance system. One distributed video surveillance system was chosen to incorporate the video analysis software module. This system is focused on working with IR cameras, video recorders (DVR, NVR), video compression boards, and network video archives (NAS) of various models and manufacturers. The main tasks performed by the video surveillance system include:

- monitoring video streams in real-time and generating signals to the operator if certain (preset events) are detected;
- automated analysis of video streams (for the presence of object movement, counting, and recognition);
- automatic video stream recording, archiving from local and remote video surveillance devices.

An example of the main window of the CCTV system user interface is shown in Fig. 4.

Fig. 4. Facing panels accounting module integrated into the surveillance system

The video surveillance system also records video streams in the original frame size and low quality, allowing remote viewing of network archives recordings using low-speed channels in an accelerated mode. The transmission of video streams to the network is carried out, taking into account the adaptation of the video stream to the bandwidth of the network. It is also possible to simultaneously record several copies in the local video archive and remote HVS network archives. Thus, viewing a copy of video recordings on other archive servers is possible without access to the video archive.

If the network connection is unstable, the video stream's resolution and the image's quality are automatically reduced. In this way, we can eliminate the problem of delaying the video stream when viewing the archive remotely, recording video information to the HVS disc at maximum image quality and frame size.

A simple SQLite database is used to store the obtained results. This local database works without additional settings, saving the processing results on the user's PC. The database contains two tables for directly storing the results of counting objects and keeping a list of cameras registering certain products. The camera list is used to provide information about video streams and to navigate between them. New records can be entered into it to create new observable processes, for example, when changing products, and vice versa to connect to existing records to provide information on specific products. The structure of the database is shown in Fig. 5.

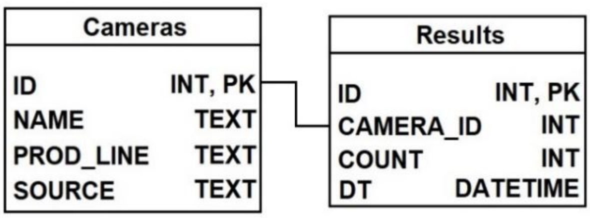

Fig. 5. SQLite database table structure

Several classes are used to manage the database, simplifying interaction with the data. The Qt framework provides tools for developing databases, simplifying their setup and maintenance with the QSqlDataBase and QSqlQuery classes.

The SQL queries.h file contains ready-made QSqlQuery queries for creating tables and retrieving data. For convenience, all SQL code was placed in this file to create a complete wrapper that could hide all complex operations and simplify further development. All functions are as universal as possible and suitable for any user data.

The DBTableController class is created to manage a separate table. All fields, types, and the table's name are specified through it. They are stored directly in the body of the class, allowing the minimization of all possible errors related to incorrect specification of column names or other data. DBTableController is a wrapper for simplified interaction in a separate database table. In the methods of this class, all simpler operations with any table are performed, and QSqlQuery queries previously prepared in SQL queries.h are executed. At the output, the developer receives ready-to-use data. This class is primarily universal and works with any data.

The DBController class completes the DB control unit. The DBController class connects to the database or creates it if there is none on the user's PC, creates all the necessary tables, and customizes everything to the application's needs. The functions prescribed in this class are required to receive and record object count data, provide a list of cameras, and create a new camera record. All interaction with the database takes place through the controller, and all functionality is encapsulated in this class. The figure illustrates these classes' interactions (Fig. 6).

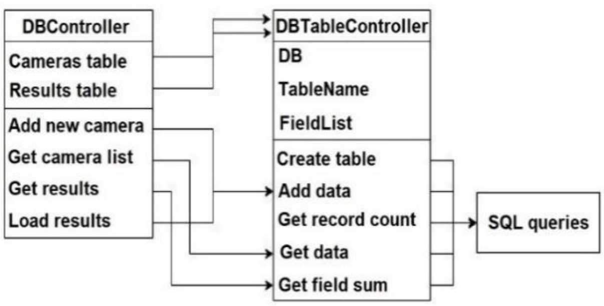

Fig. 6. SQLite database class diagram

During system performance testing, we analyzed the number of pixels in the search area and the accuracy of detecting the desired object. Since the search objects in the system are different, our new machine vision implementation plans to improve the object recognition procedure using existing algorithms to solve the problem of lighting and shadows that reduce object recognition accuracy. We plan to implement new developments in machine vision and updated algorithm models to speed up object reading and localization, which should speed up the work and increase the accuracy of the entire system.

The main idea of the system upgrade is to develop tools to increase the system's adaptability to integrate it into various production facilities with the ability to quickly customize it for a specific object while maintaining the system's performance indicators.

The facing panels accounting process results were obtained based on five different video streams; each had five other facing panels. The results are shown in Table 1.

Table 1

Examples of object accounting results

Video stream	Detection accuracy, %	Amount, %
Facing panel (type 1)	99.54	98.2
Facing panel (type 2)	99.11	98.5
Facing panel (type 3)	99.67	97.5
Facing panel (type 4)	99.88	98.6
Facing panel (type 5)	99.30	98.3

As we can see from Table 1, the accounting results for different types of objects are more than 97 %, which completely meets the research objectives. The accounting results do not depend on the shape of the object, its texture, and geometric parameters.

VIII. CONCLUSIONS

In this paper, we described the technology of computer vision and the factors that link it to real human vision. We presented actual software implementations of computer vision technology, primarily those that work in real-time and those that work with the finished image.

This paper also describes the stages of developing an algorithm for identifying, recording, and tracking moving objects. Examples of software implementations that will be used for the future implementation of the algorithm are given. During the video stream processing and analysis, several negative factors in producing and reading visual information were identified that need to be considered when developing the algorithm, such as temporary micro-movements, changes in speed, and shadow overlap.

We obtained accounting results for different facing panels based on tests of the implemented algorithm on real video streams from production lines. The experimental results show that the accuracy of panel accounting is more than 97%, which completely meets the research objectives.

Considering the analyzed materials, existing developments, and systems, a plan was developed to improve the machine vision system and its performance, quality, and accuracy of video processing using updated technologies and algorithms that would need to be tested in practice and the most optimal ones selected. The main characteristics to be improved are to speed up the system, which will allow processing a more significant number of objects and speed up the company's operations. The second improvement goal is to increase the system's adaptability to integrate into any production and process or line of work, using tools for recognizing different shapes and a user interface for quickly setting up the objects to be read.

REFERENCES

[1] Sharma A., Pathak J., Prakash M., Singh J. N., (2021). “Object Detection using OpenCV and Python”, *3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 2021,* pp. 501–505, DOI: 10.1109/ ICAC3N53548.2021.9725638.

[2] Archana K., Prasad K., (2022). “Object Detection Using Region-Conventional Neural Network (RCNN) and OpenCV”, *International Journal of Distributed Artificial Intelligence (IJDAI), 14(2), 2022,* pp. 1–9, DOI: 10.4018/ IJDAI.315277.

[3] Pang, B., Nijkamp, E., Wu, Y. N., (2020). “Deep Learning With TensorFlow: A Review”, *Journal of Educational and Behavioral Statistics, 45(2)*, pp. 227–248, DOI: 10.3102/ 1076998619872761.

[4] Shallue, C. J., Vanderburg, A., (2018). “Identifying Exoplanets with Deep Learning: A Five-planet Resonant Chain around Kepler-80 and an Eighth Planet around Kepler-90”, *The Astronomical Journal, Volume 155, Number 2,* pp. 94–117, DOI: 1 10.3847/1538-3881/aa9e09.

[5] Chen D., Zheng P., Chen Z., Lai R., Luo W., Liu H., (2021). “Privacy-Preserving Hough Transform and Line Detection on Encrypted Cloud Images”, *IEEE 20th International Conference on Trust, Security, and Privacy in Computing and Communications (TrustCom), Shenyang, China, 2021,* pp. 486–493, DOI: 10.1109/ TrustCom53373.2021.00078.

[6] Marichal-Hernández J.G., Oliva-García R., Gómez-Cárdenes Ó., Rodríguez-Méndez I., Rodríguez-Ramos J.M., (2021). “Inverse Multiscale Discrete Radon Transform by Filtered Backprojection”. *Applied Sciences. 2021; 11(1):22*, pp. 34–44, DOI: 10.3390 /app11010022.

[7] Kulik S., Shtanko A., (2020). “Using convolutional neural networks for recognition of objects varied in appearance in computer vision for intellectual robots”, *Procedia Computer Science, Volume 169, 2020,* pp. 164–167, DOI: 10.1016/j.procs.2020.02.129.

[8]Ivanov, Y., Sharov, B., Zalevskyi N., Kernytskyi, O., (2022). “Software System for End-Products Accounting in Bakery Production Lines Based on Distributed Video Streams Analysis”, *Advances in Cyber-physical Systems 2022; Volume 7, Number 2,* pp. 101–107, DOI: 10.23939/ acps2022.02.101.

Yurii Ivanov is a Ph.D. and associate professor. In 2004, he graduated from the Institute of Power Engineering and Control Systems, and in 2009, from the Institute of Computer Sciences and Information Technologies of Lviv Polytechnic National University. Yurii has an M.Sc. degree in Electrical Machines and Apparatus and Computer Science. In 2015, he obtained his Ph.D. degree at Lviv Polytechnic National University. His research interests are digital video stream processing, web-oriented applications, and database development

Petro Bilous, in 2022, received his B.Sc. degree at Computer Engineering at Lviv Polytechnic National University. Currently, he is a sixth-year student and is obtaining an M.Sc. degree. His interests are the development of hardware and software solutions based on microcontrollers.

Viacheslav Botvinnikov, in 2022, received his B.S. degree in Computer Engineering at Lviv Polytechnic National University. Currently, he is a sixth-year student and is obtaining an M.Sc. degree. His interests are developing applications using computer vision libraries and frameworks. He is also interested in combining production processes and computer vision technology.

Maksym Holovatyi, in 2022, received his B.Sc. degree in Computer Engineering at Lviv Polytechnic National University. Currently, he is a sixth-year student and is obtaining an M.Sc. degree. His interests include web back-end development using Node.js, MySQL technologies, and front-end development.