

COMPUTER SCIENCES AND INFORMATION TECHNOLOGIES

UDC 004.932.2(045)

DOI:10.18372/1990-5548.73.17007

¹M. P. Vasylenko,²M. V. Haida**VISUAL OBJECT RECOGNITION SYSTEM**

Aviation Computer-Integrated Complexes Department, Faculty of Air Navigation Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine

E-mails: ¹m.p.vasylenko@nau.edu.ua ORCID 0000-0003-4937-8082, ²gaidamax2000@gmail.com

Abstract—This article introduces the problem of object detection and recognition. The potential mobility of this solution, ease of installation and ease of initial setup, as well as the absence of expensive, resource-intensive and complex image collection and processing systems are presented. Solutions to the problem are demonstrated, along with the advantages and disadvantages of each. The selection of contours by a filter based on the Prewitt operator and a detector of characteristic points is an algorithm of the system, developed within the framework of object recognition techniques. The reader can follow the interim and final demonstrations of the system algorithm in this article to learn about its advantages over traditional video surveillance systems, as well as some of its disadvantages. A webcam with a video frame rate of 25 frames per second, a mobile phone and a PC with the Matlab2020 programming environment installed (due to its convenience and built-in image processing functions) are required to illustrate how the system works.

Index Terms—Video surveillance; automation; keypoint; feature detection; contour; contour detection.

I. INTRODUCTION

There is a significant need for establishing visual object recognition systems at a time when robberies, thefts, and other crimes are occurring increasingly frequently. These systems have already acquired popularity and are present in all supermarkets and controlled structures.

However, since video quality and optical sensor quality in general continue to improve year over year, it is no longer sufficient to merely capture the video stream on a storage medium. To make the most of the storage memory and ensure that the system operates automatically without the operator's input, the video must be pre-processed.

The solution to the issues mentioned above is the intelligent video monitoring system of the license plate contour proposed in this study.

A system that automatically locates the coordinates of the license plate, highlights it, and recognizes the characters on it was developed using a combination of techniques for object detection in the image such as template matching, contour detection, and feature detection. This system has good accuracy and speed, can be installed statically or be mobile, and does not require expensive equipment or computing resources to function.

Additionally, during the course of the article, it is possible to learn about the importance of solving

such a challenge, currently used algorithms for work realization, theoretical information, and experimental system work results.

II. OBJECT'S DETECTION METHODS USED IN THE SYSTEM

The program's target of observation was decided to be an automobile. The headlights, windshield, rearview mirrors, wheels, and registration plate are common elements on all automobiles. Nevertheless, practically all of these qualities vary in size, form, and even position for each of the present automobiles, with a big number of models.

Only neural networks could complete the task of object identification while taking into consideration all the features mentioned above. But, they need a lot of computational power and thousands, if not hundreds of thousands, of photos from the training sample to do this. So it was decided to create a custom program to recognize the automobile in the video feed by its registration number. The choice of the registration plate as the monitoring object was made since it is the component that has consistent dimensions, is found on all automobiles, and is the most effective tracking object.

Here were used such approaches as feature points detection, image correlation, contour selecting in order to solve object detection problem.

One of the earliest heuristic approaches for locating exceptional points, FAST was initially

suggested in 2005 in the work and quickly popularized due to its high computing efficiency [10]. This approach takes into account the intensity of pixels in a circle with a point C as its center and a radius of three to determine whether to treat a specific point C as exceptional or not:

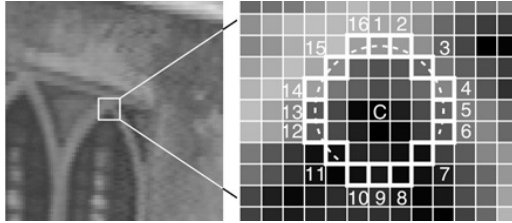


Fig. 1. FAST detector's consideration of pixels

For each of the three potential results, we compare the brightness of the circle's pixels to the center C brightness (lighter, darker it seems):

$$I_p > I_c + t,$$

$$I_p < I_c - t,$$

$$I_c - t < I_c < I_c + t.$$

Here, I represent the pixel brightness, while t represents a specified brightness threshold.

If there are $n = 12$ pixels on the circle that are brighter or darker than the center, or if there are 12 pixels that fall within either category, a point is designated as exceptional (Fig. 2).

The FAST method was selected for usage after tests on photos showed that it completely satisfies the program's requirements.

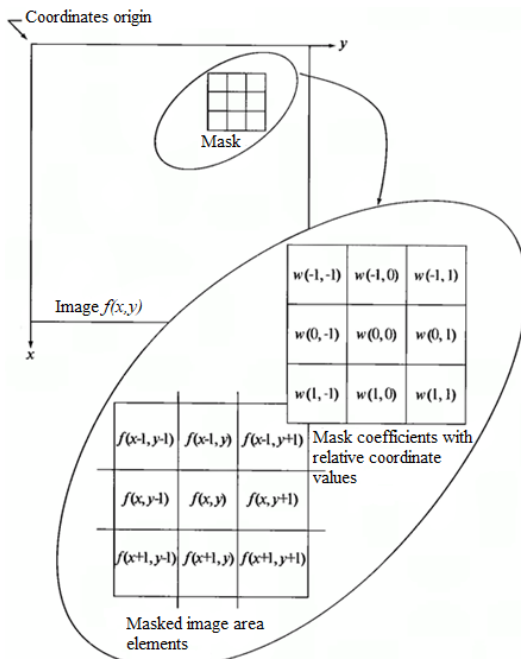


Fig. 2. Algorithm for spatial filtering

For selection of contours the Prewitt operator was used. All currently used techniques are founded on discontinuity, one of the brightness signal's fundamental characteristics. Processing a picture with a sliding mask, also known as a filter, pattern, window, or kernel, which is a form of square matrix matching to a designated set of pixels in the original image, is the most popular method for locating gaps. Coefficients are the common name for matrix elements. Filtering, often known as spatial filtering, is the process of employing a similar matrix in any local changes.

The Prewitt operator detects image edges by convolution with two filter masks. One for horizontal and one for vertical direction. At this way we can extract:

- horizontal edges;
- vertical edges;
- both directions.

The Prewitt operator, which is utilized to execute these calculations, is represented by the masks in Fig. 3.

| | | |
|----|----|----|
| -1 | -1 | -1 |
| 0 | 0 | 0 |
| 1 | 1 | 1 |

| | | |
|----|---|---|
| -1 | 0 | 1 |
| -1 | 0 | 1 |
| -1 | 0 | 1 |

Fig. 3. Masks for Prewitt operator

Also, the template matching approach, which compares the input picture with the template, was used to identify whether characters are present on the number plate.

An image's "similarity" is determined using a certain measure. In other words, the picture is "superimposed" with the pattern, and the difference between the two is taken into account. The location of the target object will be shown by the position of the template where this disparity will be minimized.

You can choose from a variety of metrics, such as cross-correlation (CCORR) or the sum of squared differences (SSD). The method might not be effective when resizing. This is because the function counts on the item being shrunk by the same amount horizontally and vertically. But that may not be the case. The distortion brought on with the log-to-polar conversion causes the search to become unstable when the size is adjusted too much.

III. PROBLEM STATEMENT

There is a huge need to create systems with algorithms for tracking the contour of the target in today's world where video surveillance systems are established at practically all regulated accesses to

private areas (for example, parking lots, territories of government agencies, etc.). This entails keeping track of and identifying the license plate of a vehicle that enters or exits a zone under control, recording the time of entry and exit, the vehicle's coordinates, and identifying and recording the symbols on the license plate.

The deployment of the neural network deep learning technique is the first solution that occurs to you while fixing this issue. But it takes hours to train a neural network to recognize an object since it needs a lot of photos out of a training sample (thousands of examples). The method of its calculation on the video adapter processor, which already places limitations on the application of hardware, is the approach's greatest drawback. The foregoing information indicates that such a system will be similarly costly.

A second method of resolving the issue is using automatic number plate recognition systems. They operate quickly and don't need a lot of computational power. The license plate must be the largest rectangle in the screen and the car must fill the full frame for this type of system to function effectively. These algorithms are extremely sensitive to shifts in the subject's location and other disturbances in the image, and recognition turns unreliable if the backdrop scenes of the frame take up more than 20% of the entire frame.

These days, proprietary, pricey, and with limited availability of code are intelligent video surveillance systems with the capability to track the object. The pricing of the system demonstrated in this work is solely based on the cost of the camera and controller and contains an open source software.

Therefore, it is necessary to develop a quick, low-cost, and simple-to-install system for monitoring the shape of the license plate and identifying license plates. What is going to be discussed in this paper.

IV. PREPARATION OF THE IMAGE

Images from the sensing device are acquired in RGB color model with a resolution of 1280 960 dots, but they also contain a lot of noise and extraneous data. It is necessary to prepare the data stored in the photo in advance so that it may be processed and subjected to various mathematical processes. Subsections after that will explain this.

a) Uploading a picture on the Matlab software environment.

Importing the image from the disk into the Matlab2020 development platform, which includes

the necessary built-in capabilities and tools, is the initial step in image processing.

Using the `imread` function, the picture is imported and then saved to a variable as a three-dimensional data array. This data array shows the value of each pixel on the three channels (red, green, and blue), which combine to determine the color of the pixel.

Now, computations may be made in the program code thanks to these arrays.

b) Convert the picture to grayscale.

The `uint8` data type, a 1 byte-sized integer with a range of 0 to 255, will be used for further computations.

Pixel values for the *R*, *G*, and *B* channels range from 0 to 255. Those three channels should be merged into one grayscale since the value of their aggregate may be more than 255. This issue is resolved using the function `rgb2gray()`.

The `rgb2gray` function weights the *R*, *G*, and *B* elements to generate a grayscale value from an RGB value:

$$I_{gray} = 0.2989 \times R + 0.587 \times G + 0.114 \times B$$

A grayscale picture (Fig. 4), which was captured as a one-dimensional grid of dots, was obtained as the output.

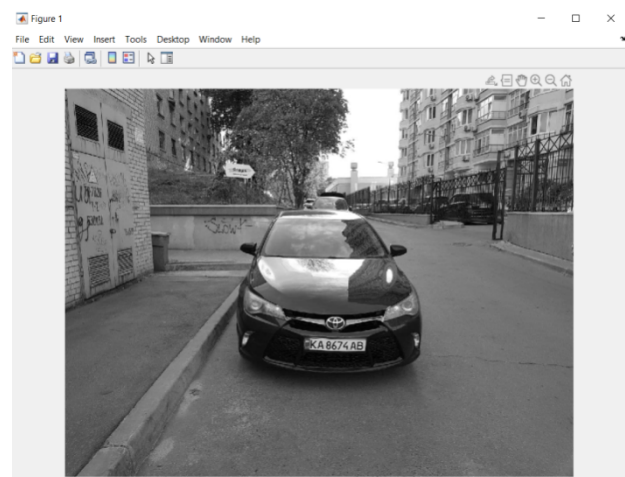


Fig. 4. Image conversion to grayscale

c) Cropping extra picture pixels.

The image above demonstrates that in addition to the car itself, the optical sensor also captured a number of other items. This will make it more difficult to identify characteristic features in the ongoing activity. The picture should be cropped at its boundaries to a specific number of pixels with the function `imcrop` in order to limit the quantity of extraneous information (Fig. 5).

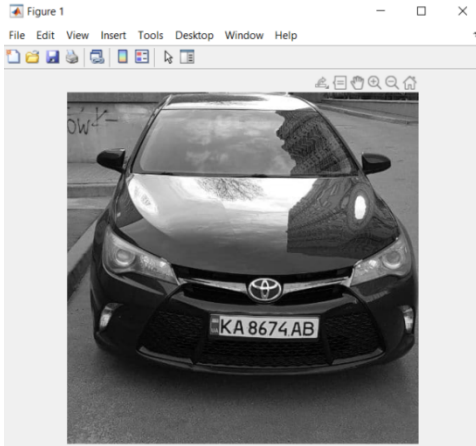


Fig. 5. Cropped photo

The amount of pixels and extraneous data in the image were reduced after cropping, which sped up the program's operation.

d) *Gauss filter used to blur the pixels in an image.*

Once the grayscale picture has been trimmed, 2-D Gaussian filtering must be used to remove any extraneous noise [4].

The sliding window's pixels that are nearer the examined pixel need to have more of an impact on the filtering outcome than the extreme ones. Consequently, a bell-shaped Gaussian function may be used to define the coefficients of the mask weights. A two-dimensional Gaussian filter is utilized to filter pictures:

$$G_{\sigma} = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}} \times \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{y^2}{2\sigma^2}}$$

The blurriness of the image increases with increasing parameter σ size. The usual filter radius equals $r = 3\sigma$. Here, the mask size $2r + 1 \times 2r + 1$ and the matrix size $6\sigma + 1 \times 6\sigma + 1$. The Gaussian function's values will be insignificant outside of this area. The `imgaussfilt()` function in Matlab may be used to apply a Gaussian filter to an image, as seen in Fig. 6.

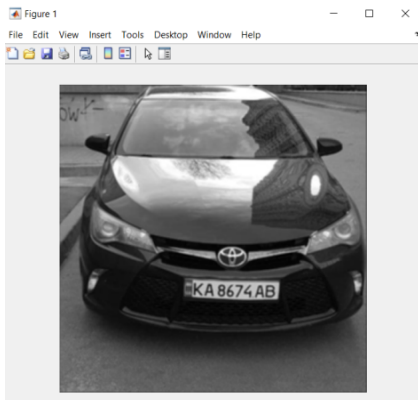


Fig. 6. Gaussian filtering's output

V. RECOGNITION OF ALL FEATURE POINTS IN THE PICTURE

The program's job is to keep track of the target's shape. In this scenario, an automobile serves as the tracking object, and the numberplate is one of the most frequent and recognizable features seen on every standard car. Its exact location will therefore be detected.

Finding all the unique features in the image is necessary before processing them in order to determine where the license plate is located.

The neighborhood of a point in the picture known as a feature point m may be differentiated from the region $o(m)$ of any other image point known as a keypoint $o(n)$ in a different neighborhood of the keypoint $o2(m)$. A detector and a descriptor are used to identify keypoints in a process.

The most well-known feature point detectors are the Harris corner detector, SIFT, and FAST. The FAST detector is optimal at identifying the distinctive spots, according to the research methodology. As its name implies, this algorithm operates more quickly than its rivals while still achieving its intended goal.

These locations can be found: In order to determine if there exists a continuous section of pixels of length t that are K grades darker (or lighter) than the candidate pixel, FAST creates a circle of radius R around it. The pixel is deemed a "key point" if this criterion is satisfied (Fig. 7).

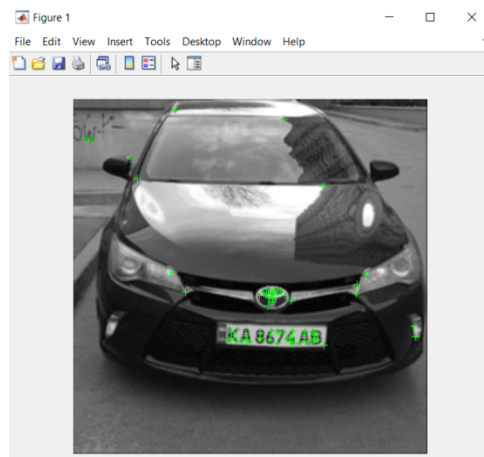


Fig. 7. The FAST algorithm identified feature points

VI. FOUND FEATURE POINTS FILTERING

The graphic above demonstrates that the distinctive points were discovered not solely on the plate number itself, but also in this location, where there is the greatest concentration of them. This implies that they need to be filtered under specific circumstances.

The filtering technique will be based on the understanding that the symbols engraved on the plate number and their edges contain the greatest amount of distinctive points. The functioning of this algorithm will then be gradually explained.

First, use the function corners. Location() to record the coordinates x, y of each typical point in their arrays.

Remove feature points that are separated by more than 60 pixels from one another after obtaining their coordinates. This value was determined through experimentation to be the best one. Next, use the formula to get the separation between every point and its neighboring characteristic points:

$$D_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2},$$

where $D_{i,j}$ is the points separation distance; x_i, y_i are coordinates for the initial point; x_j, y_j are next point's coordinates.

Now, set the variable m , that is equal to the point's weight coefficient, to every point whose neighbor is within 60 pixels of it. In other words, the weight of a feature point will increase in proportion to how many neighbors it has at a distance of up to 60 pixels.

It is possible to filter data using this criterion when the weights for each feature point are available. The minimal threshold of weight for feature points, m_{min} , is entered for this purpose. Experimental analysis is used to identify the estimated amount of unique points that are found on the license plate. Adjust the minimum weight threshold accordingly by setting it to $m_{min} - 6$ and comparing it to the weight of every point. If $m \geq m_{min}$, the point passes the test and its coordinates x_{i_i}, y_{i_i} are copied into a new array. Points that failed the check have their coordinates set to 0.

The nonzeros() method is used to delete the locations of the points, which will have a value of 0. This completes the process of filtering points. Now, a collection of points that meet the required standard – those that are located immediately on the license plate – is discovered (Fig. 8).

Additionally, it is possible that smaller groups of characteristic points that are farther away from the license plate but not too far enough to fail the distance check will fulfill the criteria for the minimum weight of the point. The following code won't function properly in this situation when trying to locate the number plate's location. In this scenario, m_{min} will rise by 1 until the weight limit of the point's threshold value is high enough to eliminate parasitic points, allowing the program to run as intended.

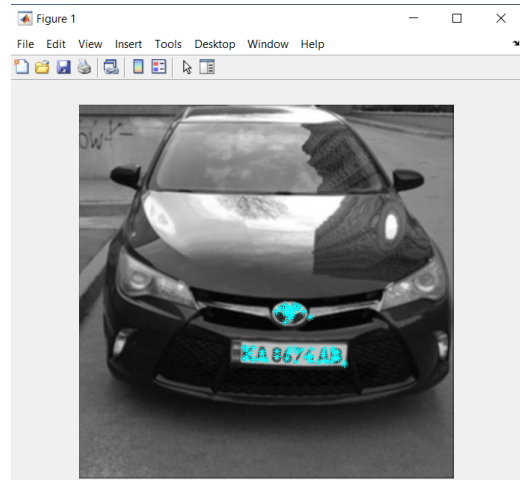


Fig. 8. Visualization of the filtered points

VII. CALCULATING THE LICENSE PLATE'S SIZE AND LOCATION

It is fair to assume that after filtering the feature points, just the license plate remains, and the process may now go on to figuring out its coordinates and size.

It is necessary to obtain information like the coordinates of the upper left corner M_{left}, M_{upper} in order to set the picture cropping boundaries and leave only the image of the plate itself and the width value W and height H of the plate, which has the following formula:

$$W = M_{right} + M_{left},$$

$$H = M_{lower} + M_{upper},$$

where M_{left} is the array of point co-ordinates' lowest value x_{i_i} (cropping to the left); M_{right} is the highest value from the points' coordinates array x_{i_i} (edge of the right crop); M_{lower} is the array of point coordinates' highest value y_{i_i} (lower crop boundary); M_{upper} is the least value in the points' coordinates y_{i_i} (upper crop boundary).

Knowing the aforementioned values, the license plate's picture is cropped as seen in Fig. 9.

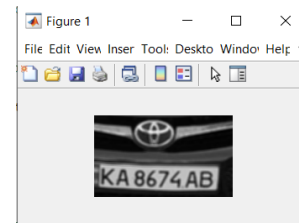


Fig. 9. The license plate image after being cropped based on its distinguishing features

The generated image must be converted from grayscale to binary in order to more clearly illustrate the locations of the number frame.

The goal of image binarization is to make the brightest and most important pixels as bright as possible, converting into white points (the color with the highest intensity or brightness), while all other points that are considered to be background become minimally bright, that is, they are transformed into black points (minimum brightness or intensity, absolute absence of color).

The *imbinarize()* function in the Matlab software environment may binarize an image using the Otsu binarization technique (Fig. 10).

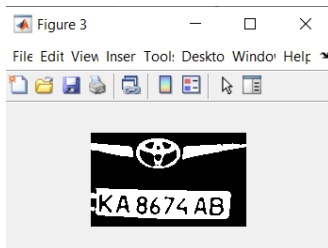


Fig. 10. License plate in binary format

Selecting the contours of the image's objects is necessary to further refine the coordinates (Fig. 11). I decided to use the Prewitt operator to choose contours.

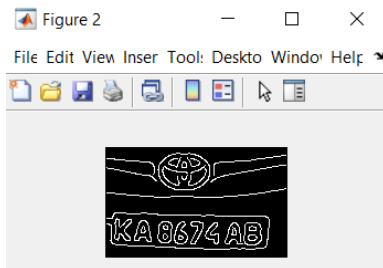


Fig. 11. Image with selected contours

Using a mask, the Prewitt operator selects an object's horizontal outlines. The mask is transposed to choose vertical shapes.

Then, in order to finally cropping of the license plate, use the *regionprops()* method to determine the attributes of the resultant binary picture. The parameters Area, BoundingBox, and Image are among them.

Where Area represents the precise number of pixels within the enclosed area; coordinates and measurements of the smallest rectangle enclosing a closed region, or "bounding box"; Binary image of the same size as the BoundingBox that was returned as a binary array is called Image.

The location, width, and height of the greatest BoundingBox are then discovered by comparing the

Area with the BoundingBox. This is the last cropping of the license plate (Fig. 12).

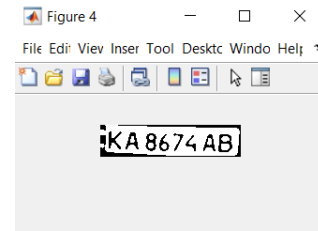


Fig. 12. Cutout of the plate's final appearance

VIII. DEFINING THE SYMBOLS ON A LICENSE PLATE

The position and specifications of the registration number have already been identified, therefore the process of identifying the characters on it may begin.

First, invert the generated binary picture by using the *bwareaopen()* method to eliminate any superfluous pixels whose closed region is less than 50 pixels (Fig. 13).

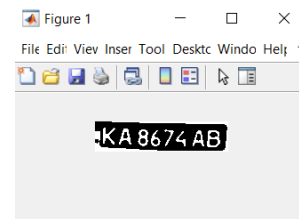


Fig. 13. Picture inverted with noise removed

The process of getting ready to recognize letters and numbers has now been completed.

The approach of template matching will be used for letter recognition. Finding the area of the image that most closely fits a template is the basis of this technique. An image's "similarity" is determined using a certain measure. In other words, the picture is "superimposed" with the pattern, and the difference between the two is taken into account. The requested item will be located at the point of the template where this divergence is least.

As a result, a database of templates for pictures of letters and numbers was initially developed and saved in the folder Alpha in order to apply this technique. Each picture is binary, has a resolution of 24 x 42 pixels, and is saved in the.bmp file format.

Each picture is assigned to its corresponding variable, which is then assigned to the corresponding arrays letter (for letters) and number (for numbers), that are then inserted into the two-dimensional array NewTemplates.

The resolution of the tablet picture that results after storing the array of images of the letters and numbers is altered in order to match it with the templates. From left to right, every template is

contrasted with the final picture of the license plate. The function $corr2()$ is used to get the correlation coefficient for each of them using the following formula:

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2\right) \left(\sum_m \sum_n (B_{mn} - \bar{B})^2\right)}}$$

where \bar{A} is arithmetic mean of A ; \bar{B} is arithmetic mean B ; A is the initial input point array; B is the second input point array; m, n is points in the first and second arrays, respectively.

The more closely the pattern and the input picture match, the higher the correlation coefficient.

The templates each have a unique allocated index. The letter or number associated with this index is shown when the $find()$ method locates the index with the highest correlation coefficient value.

Read characters from the plate are shown as a result.

There is also a check for the amount of characters detected because the algorithm might malfunction occasionally and display more than 8 characters. Additionally, additional characters are dropped if the number of characters is higher.

Additionally, there are instances where inadequate pre-preparation of the image results in the software finding fewer characters than necessary. As a result, until the procedure is successfully carried out, it is repeated if the number of characters obtained is less than 8, then it is repeated once more.

IX. DEFINING THE PLATE'S SHAPE FROM THE SOURCE PICTURE AND PRINTING THE RECOGNIZED CHARACTERS

The number plate has to be highlighted for the program's outcome to be seen. The $rectangle()$ method, which draws a rectangle using the initial coordinates, width, and height provided, will be used to do this (Fig. 14).

It is necessary to carry out inverse calculations – the reverse of what was done earlier – in order to discover these coordinates. Therefore, you can quickly determine the coordinates of the registration plate picture on the original input picture by knowing the number of iterations of the cycle, or how many times the methods for cropping the input image were carried out.

And using the $text()$ function, an inscription that replicates the wording of the characters found on the license plate is shown when these locations have been discovered (Fig. 14).

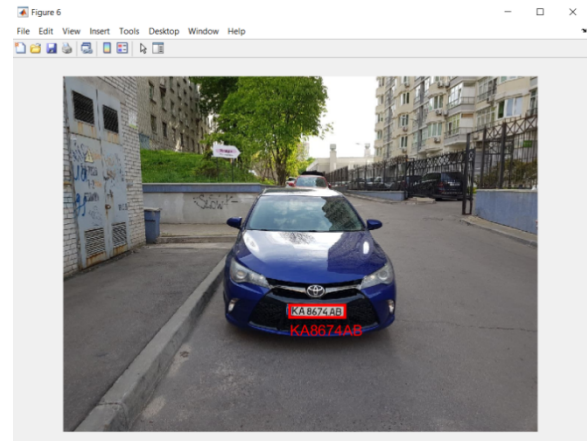


Fig. 14. Displayed letters and a number box are highlighted in the input picture

This makes the software effective at tracking license plates and identifying the symbols on them.

Naturally, it has its drawbacks, which are evident in the degree of the angle at which the video frame was captured to trace the object's contour. In an experimental study, it was discovered that the system correctly determines the coordinates of the license plate until the angle of view on the vertical and horizontal axes has exceeded about 55 degrees, and that after an angle of 50 degrees, the accuracy of the license plate symbol detection becomes less instructive.

X. WORKING CONDITIONS EXPERIMENTAL TESTINGS

The quality of work of visual recognition systems is primarily limited by the quality of the camera matrix. It is clear that for the optimal operation of the system, it should be installed at a comfortable distance from the object of recognition. To find out the optimal range of camera installation distance, you need to find the minimum and maximum permissible installation distance.

That is why experimental work was carried out to identify critical distances for the operation of the installation.

For experiments, the camera was installed at the level of the license plate, perpendicular to it. During their implementation, it was found that the minimum permissible distance to the license plate is 0.5 m.

If the distance to the object is less than the minimum permissible distance, then the FAST corner recognition algorithm will not be able to correctly find the characteristic points. Due to the large scale of the license plate frame in the photo, the FAST algorithm will find characteristic points on the license plate, but they will be too far apart. Therefore, they will not pass the process of further filtering, and those that pass will be false (Fig. 15).

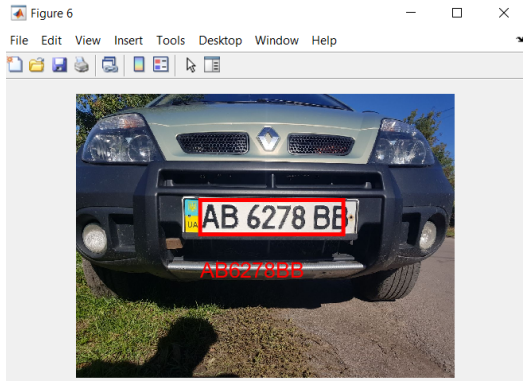


Fig. 15. Recognised number plate at minimum permissible distance

After determining the minimum permissible installation distance, the installation was moved away from the plate with the step of 0.5 meters to determine the maximum permissible installation distance (Fig. 16).



Fig. 16. Incorrectly defined characteristic points at a distance smaller than the minimum permissible

Approaching the 12 m mark, the resolution of the camera began to fall short and the accuracy of plate recognition began to drop. At a distance of 12 m, the recognition accuracy was approximately 80%, and already at a distance of 13 m, it was approximately 50%.

Judging by this experiment, it can be concluded that the maximum permissible distance for installing the camera is 12 m (Fig. 17).

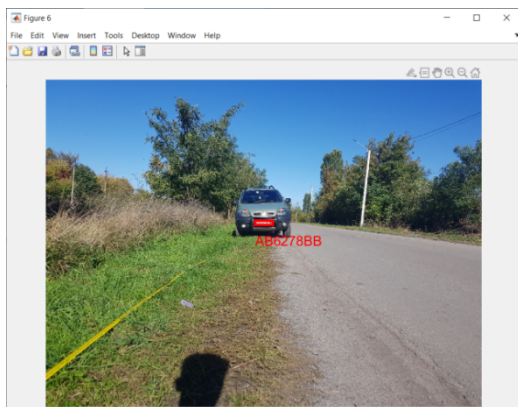


Fig. 17. Recognized license plate at a maximum permissible distance of 12 m

Usually, license plate recognition systems for cars are installed in parking lots, entrances to private or guarded areas. In such areas, it may not be possible to install the system directly in front of the car. That is, it will be located either on the side of the trajectory of its movement, or below or above.

So, for such an installation that it is necessary to know the critical angles for the stable operation of the installation.

To determine the critical vertical angle of installation of the camera, it was installed step by step at a height of 2.5 m and a distance from 1 to 5 meters.

During the experiments, the critical vertical angle of camera installation was determined, which is $\approx 60^\circ$ (Fig. 18).

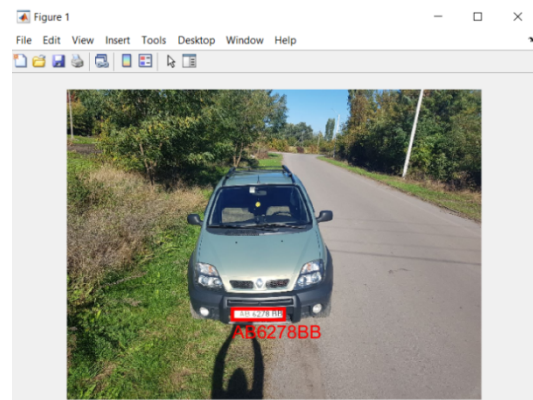


Fig. 18. Recognition result at close to critical vertical observation angle

An experiment similar to the previous one was also conducted to determine the critical horizontal angle of camera installation. At a distance of 2 m from the observation object, the camera was moved step by step in the horizontal plane.

After conducting the experiment, the critical horizontal angle of camera installation was determined to be $\approx 55^\circ$ (Fig. 19).

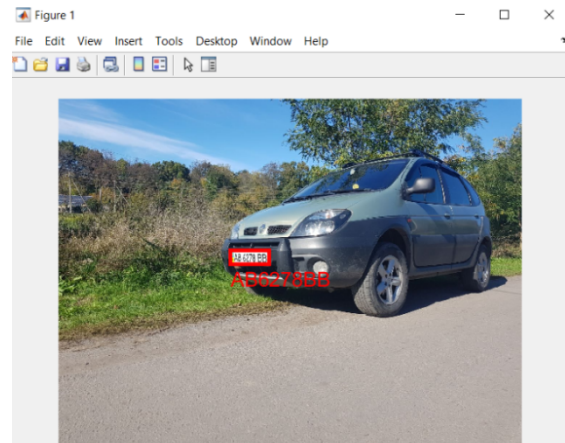


Fig. 19. Recognition result at close to critical horizontal observation angle

Therefore, it follows from the conducted research that this system can be installed within such working angles as $-60^{\circ}/+60^{\circ}$ vertically and $-55^{\circ}/+55^{\circ}$ in the horizontal plane.

Like any optical system, this setup can be sensitive to the illumination level of the observation object. In order to find out the working range of illumination for the system, several experiments were conducted under daylight and artificial illumination.

The light sensor from the Samsung Galaxy S8 phone, the TMD4906 lux Sensor and the Physics Toolbox Suite android application were used to measure the level of illumination.

An illumination sensor (dusk sensor) is a specialized technical device for automatic control of light sources that responds to the degree of illumination of a certain space. Ambient temperature and humidity do not affect the activation of the sensor.

The principle of operation is very simple: the device is equipped with a phototransistor or photoresistor, which can register the level of illumination. Speaking from a technical point of view, a photoresistor changes its resistance depending on the light flux, and a phototransistor when illuminated generates a certain amount of electricity.

Illumination measurements are not affected by the range and angle of observation of the research object, if they are within the permissible operating range.

First, experiments were carried out under daylight. They were conducted outdoors on a bright sunny day, in fog, and at dusk, to assess the impact of illumination and determine its minimum value.

At the maximum level of illumination achieved on a sunny day, which is approximately 11,500 lx, the system was found to be working normally (95% accuracy).

At normal light levels (namely 925 lx) and with fog, the system works with 100% accuracy.

At dusk, it was possible to find the minimum level of illumination for the effective operation of the system, which is 19 lx (Fig. 20).



Fig. 20. System operation at the minimum daytime level of illumination

Research has shown that the lower is the level of daylight, the more the contrast of the image drops. Since the FAST characteristic point detection algorithm reacts to changes in the intensity of neighboring pixels, the lower the contrast level, the fewer characteristic points it finds, as shown in Fig. 21.

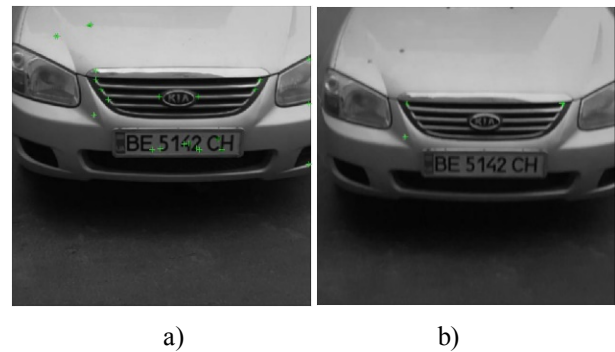


Fig. 21. Results of detection of characteristic points at the minimum permissible level of daylight (a) and beyond it (b)

Now let's move on to experiments with artificial lighting.

Experiments on the effect of artificial lighting were carried out using an LED flashlight and an LED lamp.

As it turned out during experiments, artificial lighting has a much lower light intensity than daylight. However, in some situations, the system recognizes objects better with it. This is due to the greater contrast between the background of the photo and its foreground, as well as the peculiarities of the paint applied to the license plate. Everything described above facilitates the work of the FAST characteristic point detector, so its effectiveness increases precisely under artificial lighting (Fig. 22).

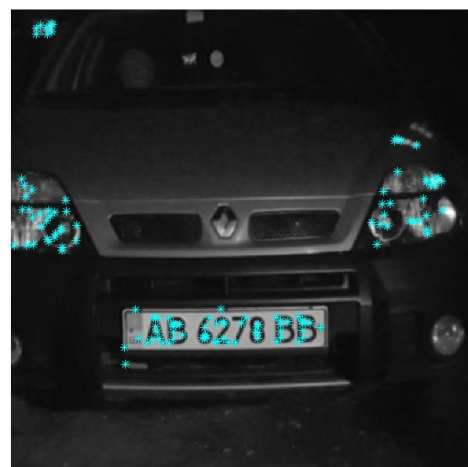


Fig. 22. The results of the detection of characteristic points at close to minimum permissible level of artificial lighting

Thus, the minimum permissible level of illumination was 2 lx.

With the help of the flashlight, it was possible to achieve a maximum illumination level of 200 lx (for comparison, the illumination level of an ordinary room with three LED lamps is 250 lx on average). At this level of light and close to minimum, the system works great, with close to 100% accuracy.

XI. CONCLUSIONS

Proposed algorithm allows to perform real time object recognition on video. It has significant performance and allows to detect necessary type of object automatically without need of human operator presence. It also allows to decrease the necessary storage capacity for video by starting recording only if the required type of object (in my example – car number plate) is detected.

System's accuracy and reliability was improved by combining the algorithms of Contour and Keypoint detection and program coefficients adjustment. It has a great speed of action and doesn't depend of the weather conditions and part of the day. More specifically, the range of the effective distance of work is 0.5–12 m; effective angles: vertically $-60^{\circ}/+60^{\circ}$, horizontally $-55^{\circ}/+55^{\circ}$; minimum illuminance threshold: in daylight 19 lx, in artificial lighting 2 lx.

Proposed software is deployed on a microcomputer RaspberryPi and became the

universal pocket video surveillance system, which can be installed and configured anywhere.

REFERENCES

- [1] A system for detecting and recognizing moving biological objects for unmanned vehicles based on intelligent edge computing / m. A. Al-Sveity et al. *Electrosvyaz* 2021, no. 9(22).
<https://doi.org/10.34832/elsv.2021.22.9.004> (date of access: 18.11.2022).
- [2] Gepard_vvk. Algoritmy vydeleniya konturov izobrazheniy. Habr <https://habr.com/ru/post/114452/> (date of access: 18.11.2022).
- [3] Car Number Plate Detection Using MATLAB and Image Processing. *Circuit Digest – Electronics Engineering News, Latest Products, Articles and Projects*.
URL: <https://circuitdigest.com/tutorial/vehicle-number-plate-detection-using-matlab-and-image-processing> (date of access: 18.11.2022).
- [4] *Repository BNTU – Repository BNTU*.
URL: https://rep.bntu.by/bitstream/handle/data/27161/Vosstanovlenie_kontrasta_polutonovogo_izobrazheniya_v_pakete_MATLAB.pdf?sequence=1&isAllowed=y (date of access: 18.11.2022).
- [5] M. P. Mukhina, "Algorithm of Visual Tracking of Contrast Moving Objects," *Of the Third World Congress "Aviation in The XXI-st Century,"* 2008.

Received September 02, 2022

Vasylenko Mykola. ORCID 0000-0003-4937-8082.

Candidate of Science (Engineering). Senior lecturer.

Aviation Computer-Integrated Complexes Department, Faculty of Air Navigation Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine.

Education: Kyiv National University of Technologies and Design, Kyiv, Ukraine, (2012).

Research interests: renewable energy sources, thermal noise based estimation of materials properties.

Publications: more than 20 papers.

E-mail: m.p.vasylenko@nau.edu.ua

Haida Maksym. Bachelor.

Aviation Computer-Integrated Complexes Department, Faculty of Air Navigation Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine.

Education: National Aviation University, Kyiv, Ukraine, (2020).

Publications: 1

E-mail: gaidamax2000@gmail.com

М. П. Василенко. М. В. Гайда. Система візуального розпізнавання об'єктів

У статті розглянуто проблему виявлення та розпізнавання об'єктів. Представлена потенційна мобільність цього рішення, простота встановлення та легкість початкового налаштування, а також відсутність дорогих, ресурсомістких та складних систем збору та обробки зображень. Продемонстровані варіанти вирішення проблеми разом із перевагами та недоліками кожного. Виділення контурів фільтром на основі оператора Превітта та детектором характерних точок є алгоритмом роботи системи, розробленим в рамках методики розпізнавання об'єктів. У статті надано можливість стежити за проміжною та остаточною демонстраціями системного алгоритму, щоб дізнатися про його переваги перед традиційними системами відеоспостереження, а також про деякі його недоліки. Веб-камера з частотою кадрів відео двадцять п'ять кадрів за секунду, мобільний

телефон і ПК із встановленим середовищем програмування Matlab 2020 (завдяки її зручності та вбудованим функціям обробки зображень) необхідні для того, щоб проілюструвати, як працює система.

Ключові слова: відеоспостереження; автоматизація; ключові точки; виявлення характерних точок; контур; виявлення контуру.

Василенко Микола Павлович. ORCID 0000-0003-4937-8082.

Кандидат технічних наук. Старший викладач.

Кафедра авіаційних комп'ютерно-інтегрованих комплексів, Факультет аеронавігації, електроніки і телекомунікацій, Національний авіаційний університет, Київ, Україна.

Освіта: Київський національний університет технологій та дизайну, Київ, Україна, (2012).

Напрямок наукової діяльності: відновлювальні джерела енергії, оцінка властивостей речовин та матеріалів за власними електромагнітними випромінюваннями.

Кількість публікацій: більше 20 наукових робіт.

E-mail: m.p.vasylenko@nau.edu.ua

Гайда Максим Володимирович. Баклавр.

Кафедра авіаційних комп'ютерно-інтегрованих комплексів, Факультет аеронавігації, електроніки і телекомунікацій, Національний авіаційний університет, Київ, Україна.

Освіта: Національний авіаційний університет, Київ, Україна, (2020).

Кількість публікацій: 1.

E-mail: gaidamax2000@gmail.com