ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

UDC 004.94
DOI: https://doi.org/10.30837/2522-9818.2019.8.130

E. YASHINA, R. ARTIUKH, N. PAN, A. ZELENSKY

INFORMATION TECHNOLOGY FOR RECOGNITION OF ROAD SIGNS USING A NEURAL NETWORK

The **subject** of the study is the methods and tools for automation of recognition of road signs at the level of software implementation. Detection of road signs is associated with the processing of a significant amount of video data in real time, which requires significant computing power. Therefore, the **purpose** of the work is to automate the process of recognition of road signs for filling the databases of navigators, which will allow operatively provide drivers with up-to-date information on established road signs. The following **tasks** are **solved**: analysis of methods and software for image recognition; development of the search algorithm for characters in the video frame; implementation of the definition of the contour of the sign; realization of a convolutional neural network for recognition of a sign; testing of applied information technology work. **Methods** are used: convolutional neural networks; Viola-Jones's method for recognizing objects in an image, the Bousting method as a way to accelerate the recognition process with a large amount of information. **Results**: Different approaches to the identification of symbols on images, various software tools for object recognition, image transformation for optimal fragment are considered. An algorithm for detecting and recognizing the sign is developed. Using the Viola-Jones method, a fast way to calculate the values of attributes using the integral representation of an image is implemented. The recognition process takes place by constructing a convolutional neural network. Features of the layers of the roller network are considered. Schematically illustrated script recognition. The process of interaction of the system with different data sources is represented by a diagram of precedents. The main result is the creation of information technology for the automated recognition of road signs. The algorithm of its work is presented in the form of a sequence diagram. **Conclusions.** Using the applied application information technology, recognition of road signs is made with an average probability of 88%, which allows automating the process of filling the database of navigators to a large extent, to increase the reliability and productivity of the given process.

Keywords: image recognition; neural network; computer vision; information technology.

Introduction

In our time, a simple and commonly used tool for drivers to make it easier to control a car is navigators. They plan a route, report traffic congestion, inform about any changes in the road situation. It provides both driver comfort and road safety for all road users. One of the most important parts of many programs, including navigators, is a constant update to meet the expectations of users. Thanks for upgrading the recommendations for drivers are really relevant and reflect the real situation. At present, the updating of the database, for the most part, is still provided by people who see the road signs, markup, analyze the situation on their own. But with the development of technologies and software, there is an opportunity to automate some of these processes, and especially – the stage of recognition of road signs. Transferring the process to a software basis will enable data updates to be faster, cheaper, and better than large staffs do. Thus, the need for human presence to verify the work of the program will remain, as it is necessary to achieve a high level of quality that will provide complete safety, but the volume of work will be reduced several times.

Analysis of publications and problem statement

The task of recognizing road signs is quite known. This is due to the fact that, firstly, road signs play a crucial role in regulating traffic, ensuring the safety of drivers and pedestrians, as well as in navigation and route search tasks; and secondly, road signs are designed in such a way as to facilitate their identification.

Signs recognition is usually performed in two stages: the detection of a sign and its own recognition. Road signs have specific colors and shapes, and are usually visible from afar on the road. As a rule, known algorithms and software detecting signs take these features into account. The input (most often video from the camera installed on it) passes a series of transformations in order to select areas that are in the correct shape and contain character-specific colors. [1].

Detection of road signs is associated with the processing of a significant amount of video data in real time, which requires significant computing power. Pre-processing an image can greatly ease the task. One way to preprocess the video is to analyze the colors used to transfer to the HSV (Hue, Saturation and Value) space. The preliminary color analysis allows to increase the reliability and the speed of software [2].

In addition, for the classification of road signs, methods are used based on the analysis of signs [3, 4]. For example, for constructing and analyzing a system of attributes, the method of reference vectors and histogram oriented gradients can be used [5].

A promising area of research is the use of neural networks for problems of pattern recognition, in particular, road signs [6, 7]. When learning a neural network or other classifiers it should be borne in mind that the image of characters on video is not always perfect. The quality of the image depends on lighting, weather conditions, the angle of view, the presence of contamination or damage and other noise factors. In order to take into account these features, it is necessary to have a significant amount of video captured in different conditions. An alternative can be the use of synthetic data obtained by generating different noise and transformations [8].

The purpose of the work is to automate the process of recognizing the road signs. To achieve the goal, the following tasks should be solved:

- analysis of image recognition methods and software;

- development of the search algorithm for characters in the video frame;
- realization of the definition of the contour of the sign;
- realization of a convolutional neural network for recognition of a sign;
- testing of applied information technology.

Materials and methods of research

1. Description of the existing process of recognizing the road signs

Safety on the road is provided primarily by traffic rules. They have their foundation - road signs and markup. Signs are divided into:

- prohibitive;
- giving orders;
- warning;
- priority signs;
- information;
- service signs.

Categories of signs are different, but within the category the signs are very similar. Signs, preferably, have the same colors and shape, so that drivers can understand from the distance what they mean. Similarity of forms and shades on signs (shades according to standards should be used the same on all signs) helps not only the drivers but also those who replenish the base of signs for navigators.

The complexity of recognizing signs will be that they can be accompanied by signs bearing different characteristics (the time of the sign, the length of the sign, etc.). When recognizing characters, it is difficult to take into account such plates, because they are usually smaller than the sign and have a small font. However, their presence allows us to understand that the insertion of signs in the database can not be unambiguous and has certain rules. Without taking into account additional information, some signs can still be entered into the database. However, you need to lay down a number of rules that must be respected when entering the recognized sign.

First of all, when implementing the recognition of road signs, you need to pay attention to the signs "pedestrian crossing", "speed limit", "ban on turning", "ban on overtaking" "curvature of the road", "caution, children" and "steep ascent/descent". Recognition of these signs is of primary importance. If these signs are clearly recognized, further addition of signs to the database will not require the significant effort.

Currently, the process of filling the database of navigators is due to the staff of the workers. From drivers of special cars there are video data on which the real road situation is shot - markup, signs, traffic lights, entrances, intersections - everything that the average driver sees. With this information, it is possible to estimate many different factors, for example: what path is indicated by the sign, to what extent it acts, as well as whether it is duplicated and whether it should be introduced into the database. With the help of human skills, these factors manage to take into account and realize the maximum similarity of the map with the road. Mostly, thanks to this quality, human labor remains indispensable in a field that does not tolerate mistakes.

But software tools are fast developing and at least signs can be automated, supplemented by defining the position in shape and size on the images and integrated into the process of filling the database with a number of advantages. Since the input, we have video data that can be decomposed into frames and each scrupulously checked (automated), then the work is reduced to determining the presence of a character in the frame and determining its content..

To determine the presence, you can use such a feature as the monotony of shades. Each sign uses the standard colors - white and blue; red and white; red, white and black. Signs stand out against the background of the environment. Even if they are not so clearly visible by the human eye, you can automatically recognize red with white between green and blue on a grey background. Of course, these colors are widespread. But looking for specific shades, combinations of colors and their accumulation can at least cut off frames that do not contain them and, accordingly, do not contain signs. It should be noted that most frames can only contain gray road, white marking and sky. These frames are intended to be cut off. Next, you need to find out where there is a sign, and where is the blue car. For example, through the shade of windshield inside or in the form. This will reduce the number of frames and facilitate the work of the neural network, which would require a lot more time.

From the description of the existing recognition process and its shortcomings, we can distinguish the key fragments that should improve the applied information technology (AIT), its main functional tasks and advantages.

The main task of the AIT is to identify places where there is a road sign and recognize which sign is present at least of those that should be manifested in the first place – "pedestrian crossing", "speed limit", "ban on turning", "ban on overtaking" "curvature of the road", "caution, children" and "steep ascent/descent". Thus, a number of images will be created from the files – frames that the program will process. Since it has no information how large the series will be, it is necessary to provide for the acceleration of work with a large amount of data. As a solution to the problem, you can split the video file into smaller fragments during operation.

The AIT should use free software, open source resources and ready-made libraries, modules, or functions since the development of object recognition in images is already well-developed.

When recognizing a road sign, it should be entered into the base of the navigator, although it is desirable to also this stage to automate. But for this, it is necessary to find the distance to the sign, its relation to a certain road, and to a certain part of it, which will greatly increase the complexity of the project. Therefore, there is a need for an intermediate stage of data storage - the result of the process of recognition process should be inserted into a separate database, which will store the data on the sign, the frame on which it is located and additional data. If necessary, it will be possible to integrate the data thus prepared with the results into the navigator base.

2. Research methods

The Viola-Jones method is used to detect objects in images [9]. Although the method can recognize different classes of images, the main task when creating it was the facial recognition. the algorithm of this method is able to detect objects very reliably and quickly enough to work in a real time.

The Viola-Jones detector is a cascade of classifiers. It combines the following concepts:

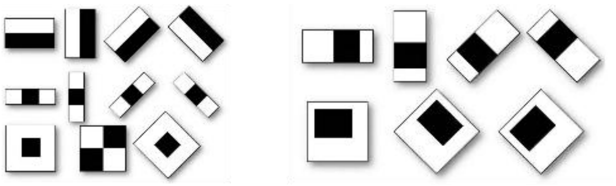
- features of Haar are used;
- images are presented in an integral form;
- boosting is used.

The feature of Haar is a rectangular primitive. From the beginning, the authors proposed the main four primitives (fig. 1a), but the OpenCV library uses additional primitives (fig. 1b) that enhance the recognition quality by additional tests from non-standard points of view.

The value of the attribute is calculated by the formula:

$$F = X - Y,$$

where X – the sum of the pixels values filled with a light part of the sign, Y – the sum of the pixels values filled with the dark part of the sign.



a) *b*)

Fig. 1. Haar's primatives are: a) standard, b) additional

Each feature works in pairs with the threshold, and the decision of the sign is determined by comparing its value with the threshold. The Viola-Jones method implements a fast way to calculate the values of the attributes, which uses the integral representation of the image. The integral representation of an image is a matrix that has dimensions as in the original image, the value of the elements which is defined as the sum of intensity peaks of the ridge, which are left and above. In such a matrix, the sum of the pixels values in an arbitrary rectangle is calculated at a constant time.

Consider a set of methods that help increase the accuracy of analytical models. "Weak" models do not allow to classify objects, make a large number of errors. Therefore, Boosting (enhancement, amplification, improvement) is intended to amplify weak models by sequentially constructing their compositions so that each subsequent model corrects the errors of the previous one..

AdaBoost (adaptive boosting) is a machine learning algorithm proposed by Yoav Freund and Robert Schapire that combines a number of weak classifiers into the strong one. This is done on the Haar features – signs of digital images used in pattern recognition. They owe their name to the intuitive similarity with Haar wavelets. Haar features were used in the first real-time face detector.

AdaBoost assigns a weight to attributes based on their quality, and as a result, a strong classifier is a linear combination of weak classifiers with corresponding weights. The Viola-Jones method combines several strong classifiers built by the AdaBoost method in a cascade. To train the cascade, a positive and false sample is built. The classifier in the first stage is selected in such a way that with a small number of primitives to cast a large number of erroneous objects while preserving almost all positive objects of the training sample.

For each next step, the number of primitives increases, false-positive actions of the previous stage are denoted as negative elements of the sample and the training continues. Consequently, the following stages are trained to correct the errors of the previous, while maintaining high accuracy on the true sample items. Thus, using a cascade allows you to quickly reject most of the false objects in the early stages, which greatly reduces the number of computations. Among the benefits of boosting: good generalizing ability; the simplicity of implementation; own overhead boosting costs are small; the ability to identify objects that are noise emissions.

The detection process is performed by sliding the detection window throughout the image. For each window, the cascade solution is calculated. In the case of a positive answer, it is assumed that the desired object is inside the window. After completing one pass of the image, the window size is increased. The window size is increased until a predetermined size is reached. A smaller percentage of increase improves the detection rate but increases the total processing time.

The recognition process takes place by constructing a convolutional neural network [10, 11].

Rolling Neural Networks include the three main paradigms:

1. Local perception. At the entrance of each neuron, not all images (or outputs of the previous layer) are fed, but only some of its area and each individual neuron is separate. This significantly reduces the amount of computing and at the same time allows you to save the topology of the image from the layer to the layer. With this approach, each neuron of the next layer receives information not from the entire previous layer, but only from the part that it describes. The selected part may be described by a matrix (usually it is chosen in the size of 5 × 5). "Scanning" by a neuron only a part of the image allows you to capture its features, which greatly increases the accuracy of recognition.

2. Joint scales. This concept allows for a large number of bonds to use a relatively small number of parameters to be studied, by selecting weight coefficients. This is achieved due to the fact that all neurons of the same card have the same weight coefficients (matrix of weights). Therefore, when learning the network for an entire map, only the weight matrix is adjusted. As a result, the training of the network runs much faster than the usual perceptron, which, during training, is configured to each link between the neurons of neighboring layers. For example, consider the number of parameters that can be

configured between two 14 × 14 and 10 × 10 cards with a 5 × 5 weight matrix. In the conglomerate it will be the number of elements of the matrix of weights, that is, 25 parameters, in the perceptron, this will be the number of all the links between the cards, that is, 19600 parameters. At the same time, from this assumption, the accuracy of recognition does not decrease.

3. Sub-sampling. The essence of this idea is to reduce the spatial dimension of the image. There are layers of the same name that do not analyze the image but only reduce its size, without losing the already highlighted features. This approach allows achieving partial invariance to scale.

The structure of convolutional neural networks involves the alternation of convolutional and sub-sampling (sub-level) layers initially and the presence of several fully-connected layers at the output.

Consider the features of the layers of the convolutional network.

1) Convolutional layer.

The essence of this layer is as follows: a limited matrix of weights moves along the treated layer, is multiplied by element in a fragment of the layer, the result is summed up and applied to the input of the activation function of the corresponding neuron of the convergent layer. The matrix of weights, it is a set of weights or a kernel of convolution, as if "graphically encodes" any sign, for example, the presence of a corner or a line. By the principle of shared weights, each core of the convolution forms a separate instance of the map of the signs, and since usually several nuclei are used, the neural network is multidimensional (on one layer, many independent maps). When processing the matrix of the weight of the layer, it is displaced every time not in full step (the size of the matrix), but on the small one. For example, with the dimension of a 5 × 5 weighing matrix to not "skip" the sought-after attribute, it tends to move one or two neurons instead of five.

The convolutional layer performs the convolution operation of input signals X (outputs of the previous layer) with the kernel W:

$$\left(X * W\right)(t) = \int_{-\infty}^{\infty} X(u)W(t-u)du\,. \tag{1}$$

In the case of discrete values, the formula for the convolution of the i-th value takes the form:

$$\left(X * W\right)_i = \sum_j x_j w_{i-j}\,. \tag{2}$$

If the input layer of the convolution is an image formed by the outputs of the previous layer X^{L-1}, the convolution with the kernel, which encodes a sign, taking into account the bias B, forms a map of signs:

$$X^L = B + X^{L-1} * W\,. \tag{3}$$

The value of the output of a separate (i, j) – neuron is calculated by the formula:

$$x^L_{i,j} = f\left(b^L + \sum_{s=1}^{K}\sum_{t=1}^{K} w^L_{s,t} x^{L-1}_{(i-1)+s,(j+t)}\right), \tag{4}$$

where L – network layer number, b^L – offset for layer L; K – the size of the receptive field of the neuron; $x^L_{i,j} \in X^L$ – the element of the matrix signal of the layer L; $w^L_{i,j} \in W^L$ – an element of the matrix weights of the layer L; $f(x)$ – activation function.

The sigmoidal function is traditionally used as a function of activation:

$$f_{sigm}(x) = \frac{1}{1+e^{-x}}\,. \tag{5}$$

Recently, however, the most commonly used is the function ReLU (rectified linear unit), which eliminates the negative values, leaving positive unchanged:

$$f_{ReLU}\left(x\right) = \max\left(0, x\right). \tag{6}$$

Using the ReLU function speeds up network performance. Additionally, it does not have a saturation region that slows down the learning by gradient descent methods.

Also, the function Softmax is often used:

$$f_{Softmax}(x_i) = \frac{e^{x_i}}{\sum_i e^{x_i}}\,. \tag{7}$$

The Softmax function outputs the neurons to a range of [0,1], and the sum of all values is equal to 1. This allows you to interpret the value of the function as the probability of belonging to an object in a certain class. Due to this, Softmax is often used as a function of the activation of the source layer of the network.

2) Subsampling layer.

The subsampling operation performs the function of reducing the dimension of the formed charts. The Max Pooling conversion is most often used:

$$x^L_{i,j} = \max_{\substack{s\,\in\,[1,N]\\ t\,\in\,[1,N]}} \left(x^{L-1}_{i+s,\,j+t}\right), \tag{8}$$

where N – window size for subsampling.

From the small square area of the neuron map of the features, the maximum is selected and taken for one neuron map of this layer features. It stores information about the features found during the performance of the convolution. Rarely use the operation of finding the Average Pooling

$$x^L_{i,j} = \frac{1}{N^2}\sum_{s=1}^{N}\sum_{t=1}^{N} x^{L-1}_{i+s,\,j+t}\,. \tag{9}$$

This layer not only accelerates the further calculations but also allows you to design a network that is more invariant to the scale of the input image.

3) Full connection layer.

This layer is a regular perceptron that has proven itself well in recognizing simple objects. After multiple image convolution passages and pushing densities, the system is rebuilt from a specific high-resolution pixel network to more abstract feature cards. As a rule, each subsequent layer increases the number of channels and decreases the dimension of the image in each channel. At the end, there is a large set of channels that store a small amount of data (even one parameter), which are interpreted as the most abstract concepts found from the original image.

These data are merged and transmitted to an ordinary full-connection neural network, which can also consist of several layers. At the same time, full-connection layers already lose the spatial structure of the pixels and have a relatively small dimension (in relation to the number of pixels of the original image).

Research results

3. Designing an algorithm for recognizing signs

In developing the main function of information technology, it was decided to realize the recognition of characters using the neural network, as a sufficiently accurate and reliable method, and to compensate for the time of its work and the complexity of filtration of the personnel that will come to it. Even a small video segment contains a large number of frames. Therefore, before giving frames for inspection, they will pass through a simpler method of recognition, which will only check the frame for the presence of characteristic features of the characters - the color and form.

First of all, the search for shades will allow you to remove more frames. All frames containing only road, markup, cars and homes will be shed as they do not contain the necessary shades and, what is also important, in the right amount. In order for the frame to go further, it must have clearly defined colors in distinct groups..

Before you give a frame for recognition, you need to check which characters are there. To do this, you need to check it for the presence of red and white (in the first place), as well as with black. Such signs should be a certain amount. In addition, so that a red car and white marking are not identified as a sign, they must be searched together. In addition, inside you can find a black color (a specific black hue, so that the windshield is not identified), but maybe the situation that it will not be. After making such settings, the extra frames are filtered out very quickly, and you will need to go further.

However, only a hue will not be enough, since there may still be errors and the frames will be able to go further. That’s why it was decided to take another check by the selected frames - on the shape of the sign.

Another method that removes the rest of the extra frames will be able to determine the shape of the sign. The signs have a clear shape – a triangle, a circle, or a rectangle. By standard, all signs should have a similar shape, so it is a fairly clear indicator for identification. Only one sign has the form of a diamond - a sign of a priority road. This small exception will not be taken into account so far, although in the future it can also be easily recognized by a square pattern rotated 45 degrees.

Unfortunately not all signs are standing strictly front surface to the driver and have a flat shape. Some can be slightly deployed to the side, tilted, or even almost deployed. Therefore, just three form templates will not be enough and you will need to perform certain transformations to recognize the form. The easiest and most obvious way is to find at least one smooth side and turn the template base on this side. If such a face is found (the circle is identified in the process), returning the pattern so that the face converges will reveal a triangle or square. Creating additional templates will only complicate the algorithm, since there are too many options for distorting the shape.

So, with the help of another check, this time a smaller sample of characters, it will be enough to precisely determine exactly which frames are characters. Recognizing a sign by these means will be difficult since it will then be necessary to identify the numbers inside, the drawing arrows, or even the text, so the next turn goes to the convolutional neural network. In addition, the neural network is able to study and improve the data over time, replenishing its own knowledge, which is very useful in this field..

Therefore, you can illustrate the recognition scenario in fig. 2 schematically.

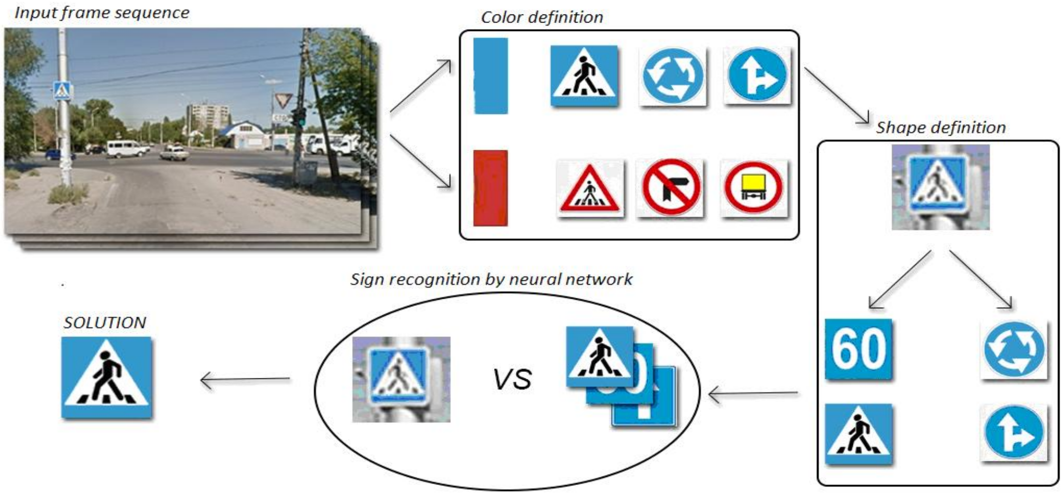


Fig. 2. Scenario of the work of information technology

We illustrate the process of interaction of the system with different data sources using the precedent diagram in fig. 3. The diagram shows:

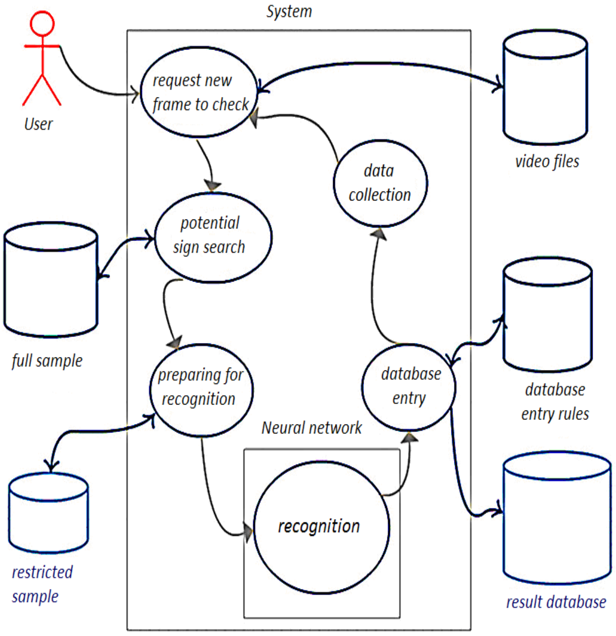


Fig. 3. Diagram of precedents

1. In addition to a sample of characters and video files from which information is to be recognized, there are still different rules for inserting characters into the database, since the characters are different in principle, and may also have different coding options, for example, in different countries. This fact should also be taken into account when recognizing signs. Recognition and entry rules must be specified by the user, or entered in accordance with the technical data from the video. This will allow you to adjust the program in case the rules change.

2. The neural network occupies the key place of recognition of the sign and conducts voluminous work on recognition. Nevertheless, in the scale of AIT in general, much of the technical work in the preparation and collection of data is carried out beyond its borders. Thanks to this, the processing time for image searches should be significantly reduced.

At the output of the program, a database will be formed in which the results of the recognition are stored. These will be the XML files that collect the following data:

- a frame on which the sign is recognized;
- category of the sign;
- sign meaning (speed limit, angle of inclination);
- form of the sign (stored separately for visual display to the user); which should be entered into the navigator in accordance with the rules;
- a note that the sign is not recognized for user verification.

The algorithm of work can also be illustrated using the UML activity diagram in fig. 4, which clearly shows what actions and in which the sequence is executed.

Each main point of the AIT is indicated as a cycle marked on the diagram with an orange diamond. At these stages, the program checks data and decides how to proceed.

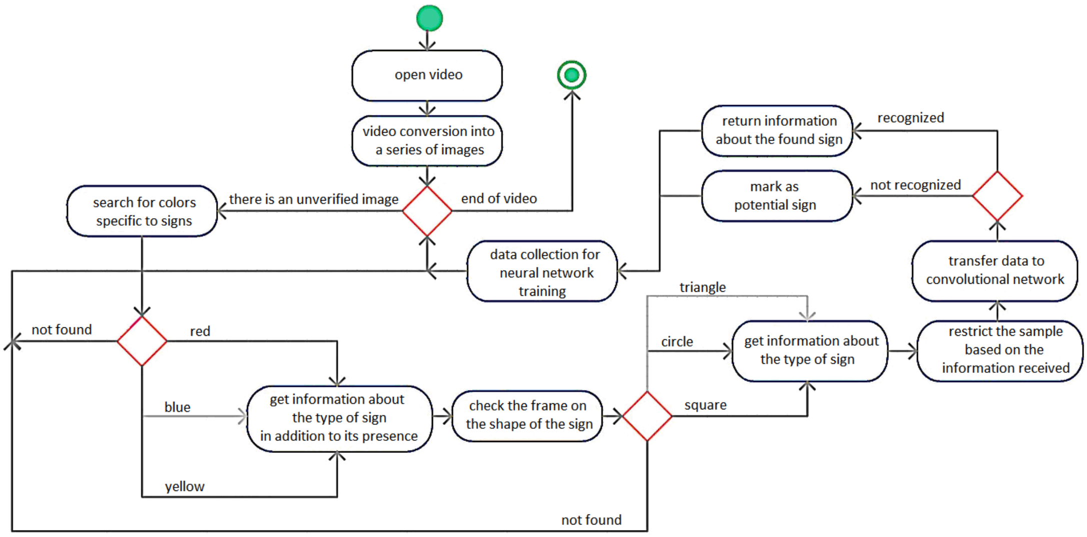


Fig. 4. Diagram of the activity of the technology of recognition of road signs

As you can see in the diagram - the technology implements three major significant actions and one additional – the conversion of video into a series of images. Since the incoming data will come from the customer, then their format may differ from the usual video, so you need to set up the input data to a single type – a group of individual frames in a single format. After that, we have three stages: recognizing the shade in the frame, detecting its shape, and further recognizing it in the neural network.

In addition, for every significant action, we receive additional information:

- when the shades are detected, you can identify the type of sign (prohibitive, warning, or prescriptive) and thereby reduce the subsequent sampling for recognition in the neural network;

- when the shape is recognized, the set of possible variants of the character types will also be reduced;

- after checking the neural network, regardless of whether the mark is recognized or not, the data is collected for the user to be output (or immediately entered into the database) in the first place, as well as for further training of the network in the second place.

Even if the program could not recognize the sign - the data can be passed to the user who either will be able to recognize the sign or will simply be informed about his ability to exist here. In addition, it will allow you to further check the work of the AIT itself and to provide it with data that is not enough.

4. Neural Network Recognition

Using the previous two steps, the neural network work is greatly simplified because:

- A limited number of frames were found from all videos where signs are very likely;

- Reduced sample of characters that can be depicted in a frame using the detected color and form;

- The location of the sign is found in the contour search, so you will not need to check the entire image - only a small part.

From this follows that the work of the neural network itself will take place only with the necessary material and will not use too much time.

The work of AIT is illustrated by a sequence diagram in fig. 5.

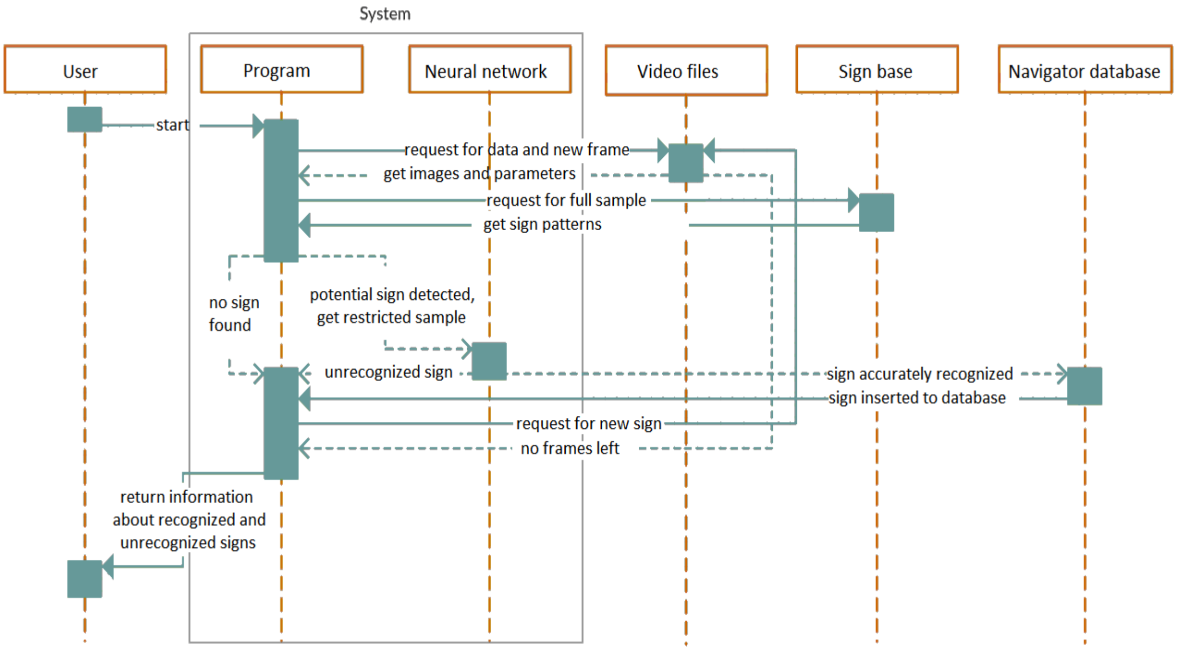

System
User
Program
Neural network
Video files
Sign base
Navigator database
start
request for data and new frame
get images and parameters
request for full sample
get sign patterns
no sign found
potential sign detected, get restricted sample
unrecognized sign
sign accurately recognized
sign inserted to database
request for new sign
no frames left
return information about recognized and unrecognized signs

Fig. 5. A sequence diagram

This diagram shows the sequence and the elements the system interacts with different variants of action, as in the cycles at the activity diagram.

The diagram does not specify a set of rules that are mentioned in the precedent diagram since it is assumed that this is not a separate database, but the configuration within the system and it is arranged in the "Program". The rules are simply taken into account and applied where necessary. In the precedent diagram, they are referred to for completeness of understanding the data that the program works with.

In order to realize the recognition, the library of computer vision OpenCV (open source computer vision library) was thrown [12, 13]. It has an open source code and is available for several programming languages, such as Java. Includes a variety of computer vision algorithms, image recognition and many others working in real time. The library is widely used in research, including the design of machine vision systems for unmanned vehicles [14].

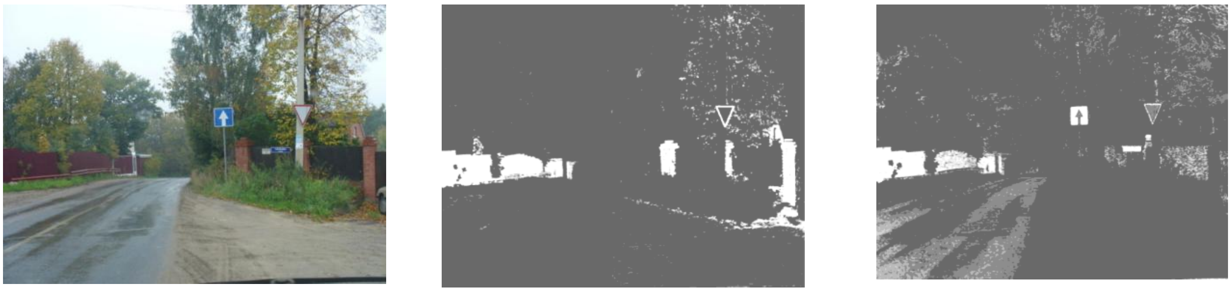
When looking for a color in the frame to restrict the frames to be checked, it was decided to look for a certain color on the signs - one of the standard shades for the signs. It is blue (ordering), red (warning and prohibitive) and white (general). To search, the HSV color model (Hue, Saturation, Value) was used - a color model in which the color coordinates are:

1. Hue - color tone (for example, red, green or blue-blue). It varies in the range of 0-360°, but sometimes it is reduced to a range of 0-100 or 0-1.

2. Saturation. Varying within 0-100 or 0-1. The larger this parameter, the "cleaner" color, so this option is sometimes called purity of color. And the closer this parameter to zero, the closer the color to the neutral gray.

3. Value (color value) or Brightness (Brightness). Also set within 0-100 and 0-1.

An illustration of the work of this function is depicted in fig. 6. Fig. 6. a) depicts an output frame with characters. It has a red sign indicating a low priority at the crossroads, as well as a blue one-way sign. As can be seen in fig. 6. b) and fig. 6. c), the transformation for a given (red or blue) color of a bright white highlights places where there is this shade.



a) input image *b) search for red color* *c) search for blue color*

Fig. 6. Convert an image to search for colors that are specific to road signs

Then, without interference, you can detect a sufficiently white fragment on the converted image by searching through OpenCV and proceed to the form detection. Although at this stage there is some information about the presence of a character in the frame.

To determine the shape of a character on the selected frames using the color of the painted sign, you can understand its boundaries. This will simplify the search by reducing the sample among the options. It will also allow you to remove shots that do not have characters, but there are the necessary shades (billboards, fragments of cars, etc.).

Regarding the work of the network, at the stage where the form was discovered, a number of convolutional and subsampling layers have formed that pass through the entire found area and, in the process of forming the full-connection layers, find the necessary forms by which one can identify the sign. Using the information about the signs that were sent out to check in the previous stages, the forms will continue to appear. First of all, this is a quantitative data - indicators of maximum speed, length of the sign, time of action, and so on. In the next turn (if necessary) - arrows, forms of cars, black fragments of road geometry, geometric shapes, etc. Some of these tags are very similar, and they may have a single template to validate, others will have to be checked separately. The identification of such corporate marks identifies a sign. This will mean that it can be entered into the database, and the network will remember the right decisions. In a situation in which there will be no similarity to any of the fragments, the network will inform that there is a possible sign here. This will give an impetus to the addition of the base, change the principle of work as needed, or provide additional information for verification, even if it is not needed.

Designing a convolutional neural network involves the formation of its architecture [16]. This is done using the freely available Caffee library (Convolutional architecture for fast feature embedding) [17]. Caffee supports CUDA technology, which allows you to perform fast computing on multi-core graphics processors. A network built in Caffee using the JSON format can be imported into the DNN (deep neural network) of the OpenCV library. This allows you to build a single software environment for image preparation and further recognition.

The following classes of the Caffee library were used to build the network:

- Input – the input layer receives a fragment of the image that potentially contains the sign and transmits it to the network;

- Convolution (Conv) – the convolutional layer performs the operations of convolution with kernels of size 5×5;

- ReLU – the activation layer that calculates the activation function of the convolutional neurons by the formula (6);

- Pooling (Pool) – a subsampling layer that performs the max pooling operation using the formula (8) with the cards 2×2.

- InnerProduct (FC, full connection) – a full-connection layer that forms the output of the network;

- Softmax – the activation layer of the output neurons, which calculates the Softmax function by the formula (7).

The network architecture reflects the sequence of the libraries used:

$$\text{Input} \to \text{Conv} \to \text{ReLU} \to \text{Pool} \to \text{Conv} \to \text{ReLU} \to \text{Pool} \to \text{FC} \to \text{Softmax}$$

At the learning stage, additional layers were used, such as:

- Dropout – a regularization layer that prevents network redesign, that is, memorization of the training sample instead of its generalization.

- Accuracy – calculates the accuracy of the classification as the ratio of the number of objects that are classified correctly to the total number of objects in the training voter.

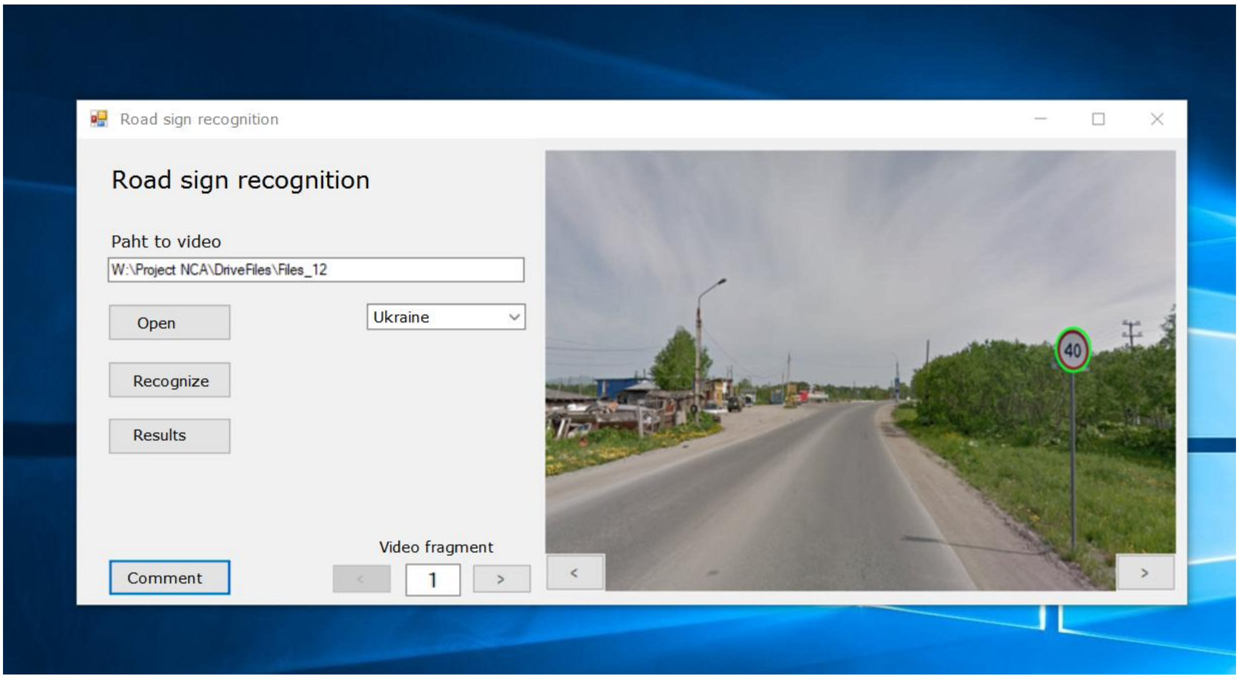
Thus, the software architecture of the neural network is constructed and implemented.

5. Experiments on work with AIT

In the first stage of user work with AIT it is necessary to determine the path to the video file. If the data format is correct - the image from the first frame will appear in the right window. If not, a new window opens with a message that the format is not supported.

When opening the video will be divided into fragments of 1000 frames, for the convenience of viewing very large files. After that, the frame view of all fragments will be realized according to the previously described algorithm and the entry of the data in the new database.

In case the data needs to be checked, this can be done after the recognition. The program will download information about the frame and position of the mark from the newly created database only in those frames that are identified as potentially having a sign. Fig. 7 shows an example where the program recognized the speed limit sign and highlighted it in color.



Paht to video
W:\Project NCA\DriveFiles\Files_12

Ukraine

Fig. 7. An example of speed limit sign recognition

After developing algorithms for recognition and creating a form for input values, the system was tested on test video data. Two video files were taken, with a number of frames over 1000, and provided to the program to recognize the specified types of characters. Some signs have varieties: distortion of the road, the prohibition of turning (right and left), and speed limitations. The test sample contained different characters and the program recognized them separately.

The volume of 1000 frames includes almost 10 kilometers of road outside the city and twice less in the city, but in the city a large number of signs. In total, 1000 frames have 30 to 60 characters, this is enough to determine the quality of recognition. After recognizing the program, the signs were verified manually and data was collected on how many times the program found the mark correctly, and how many times it was mistaken.

Fig. 8 shows a histogram in which the percentage of recognition of a particular sign is given.

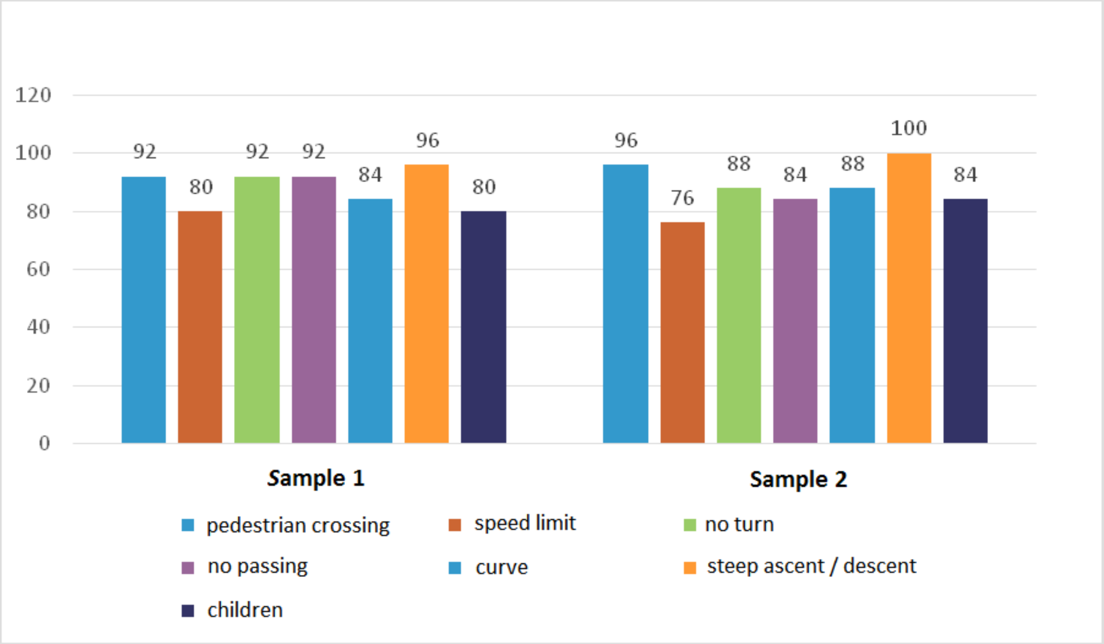


Fig. 8. Histogram of recognition quality

From fig. 8, we can draw some conclusions about the work of the AIT – it recognizes all the prescribed categories of signs and is sufficiently accurate. In both cases, all categories of recognition were successful at 80% or more except for the speed limit in the second sample. The program correctly recognizes the category of the sign, but allows errors in recognizing the speed value.

Best of all, the program coped with the recognition of the sign of a steep climb and descent, in which part of the sign is filled with black, and the sign of the pedestrian crossing. In other cases, recognition took place at 88%. This does not allow you to fully automate the recognition of signs, but allows you to reduce human participation to review the results of the program and correct individual errors. In addition, the filling of the base of navigators substantially accelerates and the complexity of this process decreases.

Conclusions

Within the framework of this work various approaches to the detection of characters in images, various software tools for object recognition, image transformation for optimal fragment were considered. The principles of operation of the convolutional neural network, which can recognize a sign from an image, are broken down into fragments. Also, the principle of the Viola-Jones method is considered, as one of the most common for recognizing objects in the image. Separately, we present a way of Boosting to accelerate the recognition process with a large amount of information prior to recognition. Also reviewed is a library that can detect frame fragments for validation. This greatly simplifies and accelerates the work of the neural network and technology as a whole

The general algorithm of work of AIT is developed, which includes the units of detection of a sign in a frame, definition of its form, and then its contents.

In general, information technology is divided into identification, a bulky and key process, as well as a technical part that helps in the collection and preparation of data, reducing the work time, interaction with external resources and the user.

To identify the sign and its forms, the OpenSV library was used. For the recognition of the content of the characters in the image, a convolutional neural network is developed.

The peculiarities of realization of the system of recognition of road signs in different stages of its work are described - the search for a certain shade in the frame, the recognition of the shape along the sign's mark, and the actual recognition of the image on the sign using a convolutional neural network.

After the description of the technology itself, the results of the recognition of the test sample, which tested the system's performance, are given. According to the results of the AIT testing, it was found that the program recognizes signs with an average probability of 88%, which significantly increases the reliability and efficiency of the process of filling the databases of navigators.

References

1. Shneier, M. (2006), "Road sign detection and recognition", *Unmanned Systems Technology VIII, International Society for Optics and Photonics*, Vol. 6230 (623016). DOI: https://doi.org/10.1117/12.660219

2. Yakimov, P. Yu. (2013), "Preliminary processing of digital images in the systems of localization and recognition of road signs" ["Predvaritel'naya obrabotka tsifrovykh izobrazheniy v sistemakh lokalizatsii i raspoznavaniya dorozhnykh znakov. Komp'yuternaya optika"], *Computer Optics*, Vol. 37, No. 3. P. 401–405.

3. Malyeyeva, O., Nosova, N., Fedorovych, O., Kosenko, V. (2019), "The Semantic Network Creation for an Innovative Project Scope as a Part of Project Knowledge Ontology", *CEUR Workshop Proceedings. Proceedings of the 3rd International Conference on Computational Linguistics and Intelligent Systems (COLINS-2019)*, Vol. I, P. 301–311.

4. Bolohova, N. Ruban, I. (2019), "Image processing models and methods research and ways of improving marker recognition technologies in added reality systems", *Innovative Technologies and Scientific Solutions for Industries*, No. 1 (7), P. 25–33. DOI: https://doi.org/10.30837/2522-9818.2019.7.025

5. Lisitsyn, S. O., Baida, O. A. (2012), "Recognition of road signs using the method of support vectors and histograms of oriented gradients" ["Raspoznavaniye dorozhnykh znakov s pomoshch'yu metoda opornykh vektorov i gistogramm oriyentirovannykh gradiyentov"], *Computer optics*, Vol. 36, No. 2, P. 289–295.

6. Zhu, Y., Zhang, C., Zhou, D., Wang, X., Bai, X., Liu, W. (2016), "Traffic sign detection and recognition using fully convolutional network guided proposals", *Neurocomputing*, Vol. 214, P. 758–766. DOI: http://dx.doi.org/10.1016/j.neucom.2016.07.009

7. Bodyanskiy, Y., Perova, I., Zhernova, P. (2019) "Online fuzzy clustering of high dimension data streams based on neural network ensembles", *Innovative Technologies and Scientific Solutions for Industries*, No. 1 (7), P. 16–24. DOI: https://doi.org/10.30837/2522-9818.2019.7.016

8. Chigorin, A. A., Konushin, A. S. (2013), "Experiments with learning methods for recognizing road signs on synthetic data" ["Eksperimenty s obucheniyem metodov raspoznavaniya dorozhnykh znakov na sinteticheskikh dannykh"], *Science and education: a scientific publication MSTU. N.E. Bauman*, No. 08, P. 315–324. DOI: https://doi.org/10.7463/0813.0603378, available at : https://bmstu.ru/doc/603378.html

9. Viola, P. A., Jones, M. J. (2004), "Robust real-time face detection", *International journal of computer vision*, Vol. 57, No. 2, P. 137–154.

10. LeCun, Y., Bengio, Y. (1994), "Word-level training of a handwritten word recognizer based on convolutional neural networks", *Proceedings of the International Conference on Pattern Recognition*, Jerusalem, Israel, Vol. 2, P. 88–92.

11. Lawrence, S., Giles, C. L., Tsoi, A. C., Back, A. D. (1997), "Face recognition: a convolutional neural network approach", *IEEE Transactions on Neural Networks*, Vol. 8, No. 1, P. 98–113.

12. Open Source Computer Vision, available at : https://docs.opencv.org/3.4/index.html

13. Kaehler, A., Bradski, G. (2016), *Learning OpenCV 3: computer vision in C++ with the OpenCV library*, Reilly Media, Inc., 1024 p.

14. Dergachev, K. Yu., Krasnov, L. A., Pyavka, I. A. (2018), "Analysis of the performance of technical vision algorithms in trajectory measurement problems" ["Analiz raboty algoritmov tekhnicheskogo zreniya v zadachakh trayektornykh izmereniy"], *Electrotechnic and Computer Systems*, No. 27 (103), P. 187–195. DOI: https://doi.org/10.15276/eltecs.27.103.2018.21

15. Yaloveha, V., Hlavcheva, D., Podorozhniak, A. (2019), "Usage of convolutional neural network for multispectral image processing applied to the problem of detecting fire hazardous forest areas", *Advanced Information Systems*, Vol. 3, No. 1, P. 116–120. DOI: https://doi.org/10.20998/2522-9052.2019.1.19

16. Krizhevsky, A., Sutskever, I., Hinton, G. E. (2012), ЄImagenet classification with deep convolutional neural networks", *Advances in neural information processing systems*, No. 25, P. 1097–1105.

17. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T. (2014), "Caffe: Convolutional architecture for fast feature embedding", *Proceedings of the 22nd ACM international conference on Multimedia*, ACM, P. 675–678.

Received 31.05.2019

Відомості про авторів / Сведения об авторах / About the Authors

Яшина Олена Сергіївна – кандидат технічних наук, Національний аерокосмічний університет ім. М. Є. Жуковського "Харківський авіаційний інститут", доцент кафедри комп'ютерних наук та інформаційних технологій, Харків, Україна; e-mail: o.yashina@khai.edu; ORCID: https://orcid.org/0000-0003-2459-1151.

Яшина Елена Сергеевна – кандидат технических наук, Национальный аэрокосмический университет им. Н. Е. Жуковского "Харьковский авиационный институт", доцент кафедры компьютерных наук и информационных технологий, Харьков, Украина.

Yashina Elena – PhD (Engineering Sciences), National Aerospace N. E. Zhukovskiy University "Kharkov Aviation Institute", Associate Professor of the Department of Computer Science and Information Technologies, Kharkov, Ukraine.

Артюх Роман Володимирович – кандидат технічних наук, Державне підприємство "Південний державний проектно-конструкторський та науково-дослідний інститут авіаційної промисловості", директор, Харків, Україна; e-mail: roman.artyuh77@gmail.com; ORCID: https://orcid.org/0000-0002-5129-2221.

Артюх Роман Владимирович – кандидат технических наук, Государственное предприятие "Южный государственный проектно-конструкторский и научно-исследовательский институт авиационной промышленности", директор, Харьков, Украина.

Artiukh Roman – PhD (Engineering Sciences), State Enterprise "National Design & Research Institute of Aerospace Industries", Director, Kharkiv, Ukraine.

Пан Микола Павлович – кандидат технічних наук, доцент, Харківський національний університет міського господарства імені О. М. Бекетова, доцент кафедри прикладної математики та інформаційних технологій, Харків, Україна; e-mail: pan@kname.edu.ua; ORCID: https://orcid.org/0000-0001-7137-5174.

Пан Николай Павлович – кандидат технических наук, доцент, Харьковский национальный университет городского хозяйства имени А. Н. Бекетова, доцент кафедры прикладной математики и информационных технологий, Харьков, Украина.

Pan Nikolai – PhD (Engineering Sciences), Associate Professor, O. M. Beketov National University of Urban Economy in Kharkiv; Associate Professor of the Department of Applied Mathematics and Information Technology, Kharkiv, Ukraine.

Зеленський Андрій Олександрович – Національний аерокосмічний університет ім. М. Є. Жуковського "Харківський авіаційний інститут", студент кафедри комп'ютерних наук та інформаційних технологій, Харків, Україна; e-mail: zelenman48@gmail.com; ORCID: https://orcid.org/0000-0001-8862-5387.

Зеленский Андрей Александрович – Национальный аэрокосмический университет им. Н. Е. Жуковского "Харьковский авиационный институт", студент кафедры компьютерных наук и информационных технологий, Харьков, Украина.

Zelensky Andrei – National Aerospace N. E. Zhukovskiy University "Kharkov Aviation Institute", Student of Department of Computer Science and Information Technologies, Kharkov, Ukraine.

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ РОЗПІЗНАННЯ ДОРОЖНІХ ЗНАКІВ З ВИКОРИСТАННЯМ НЕЙРОННОЇ МЕРЕЖІ

Предметом дослідження є методи та інструментальні засоби автоматизації розпізнання дорожніх знаків на рівні програмної реалізації. Детектування дорожніх знаків пов'язане з обробкою значного обсягу відеоданих в реальному часі, що потребує значних обчислювальних потужностей. Тому **метою** роботи є автоматизація процесу розпізнання дорожніх знаків для наповнення баз даних навігаторів, що дозволить оперативно надавати водіям актуальну інформацію щодо встановлених дорожніх знаків. Вирішуються наступні **завдання**: аналіз методів та програмних засобів розпізнання зображень; розробка алгоритму пошуку знаків на кадрі відео; реалізація визначення контуру знаку; реалізація згорткової нейронної мережі для розпізнання знаку; тестування роботи прикладної інформаційної технології. Застосовуються **методи**: згорткові нейронні мережі; метод Віоли-Джонса для розпізнання об'єктів на зображенні, метод Бустінгу як спосіб прискорення процесу розпізнання при великому об'ємі інформації. **Результати**: розглянуті різні підходи до виявлення знаків на зображеннях, різноманітні програмні засоби розпізнання об'єктів, перетворення зображень для отримання оптимального фрагменту. Розроблено алгоритм виявлення та розпізнання знаку. Із застосуванням методу Віола-Джонса реалізовано швидкий спосіб обчислення значень ознак, який використовує інтегральне представлення зображення. Процес розпізнавання відбувається шляхом побудови згорткової нейронні мережі. Розглянуто особливості шарів згорткової мережі. Схематично проілюстровано сценарій розпізнання. Процес взаємодії системи з різними джерелами даних представлений за допомогою діаграми прецедентів. Основним результатом є створення інформаційної технології автоматизованого розпізнання дорожніх знаків. Алгоритм її роботи представлено у вигляді діаграми послідовності. **Висновки.** Із застосуванням створеної прикладної інформаційної технології розпізнання дорожніх знаків робиться з середньою вірогідністю 88%, що дозволяє значною мірою автоматизувати процес наповнення бази даних навігаторів, підвищити надійність та продуктивність вказаного процесу.

Ключові слова: розпізнання зображень; нейронна мережа; комп’ютерний зір; інформаційна технологія.

ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ РАСПОЗНАВАНИЯ ДОРОЖНЫХ ЗНАКОВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННОЙ СЕТИ

Предметом исследования являются методы и инструментальные средства автоматизации распознавания дорожных знаков на уровне программной реализации. Детектирования дорожных знаков связано с обработкой большого объема видеоданных в реальном времени, что требует значительных вычислительных мощностей. Поэтому **целью** работы является автоматизация процесса распознавания дорожных знаков для наполнения баз данных навигаторов, что позволит оперативно предоставлять водителям актуальную информацию об установленных дорожных знаках. Решаются следующие **задачи**: анализ методов и программных средств распознавания изображений; разработка алгоритма поиска знаков в кадре видео; реализация определения контура знака; реализация сгруппированной нейронной сети для распознавания знака; тестирование работы прикладной информационной технологии. Применяются **методы**: сгруппированные нейронные сети; метод Виолы-Джонса для распознавания объектов на изображении, метод Бустинга как способ ускорения процесса распознавания при большом объеме информации. **Результаты**: рассмотрены различные подходы к выявлению знаков на изображениях, различные программные средства распознавания объектов, преобразования изображений для получения оптимального фрагмента. Разработан алгоритм обнаружения и распознавания знака. С применением метода Виола-Джонса реализовано быстрый способ вычисления значений признаков, который использует интегральное представление изображения. Процесс распознавания происходит путем построения сгруппированной нейронной сети. Рассмотрены особенности слоев сгруппированной сети. Схематично проиллюстрировано сценарий распознавания. Процесс взаимодействия системы с различными источниками данных представлен с помощью диаграммы прецедентов. Основным результатом является создание информационной технологии автоматизированного распознавания дорожных знаков. Алгоритм ее работы представлен в виде диаграммы последовательности. **Выводы**. С применением созданной прикладной информационной технологии распознавания дорожных знаков производится со средней вероятностью 88%, что позволяет в значительной степени автоматизировать процесс наполнения базы данных навигаторов, повысить надежность и производительность указанного процесса.

Ключевые слова: распознавание изображений; нейронная сеть; компьютерное зрение; информационная технология.

Бібліографічні описи / Bibliographic descriptions

Яшина О. С., Артюх Р. В., Пан М. П., Зеленський А. О. Інформаційна технологія розпізнання дорожніх знаків з використанням нейронної мережі. *Сучасний стан наукових досліджень та технологій в промисловості*. 2019. № 2 (8). С. 130–141. DOI: https://doi.org/10.30837/2522-9818.2019.8.130.

Yashina, E., Artiukh, R., Pan, N., Zelensky, A. (2019), "Information technology for recognition of road signs using a neural network", *Innovative Technologies and Scientific Solutions for Industries*, No. 2 (8), P. 130–141. DOI: https://doi.org/10.30837/2522-9818.2019.8.130.