DOI: https://doi.org/10.15407/kvt199.01.005

UDC 519.1

KYYKO V.M., PhD (Engineering),
Senior Researcher of Pattern Recognition Department
e-mail: vkiiko@gmail.com
International Research and Training Center for Information Technologies
and Systems of the National Academy of Sciences of Ukraine
and Ministry of Education and Science of Ukraine,
40, Acad. Glushkov av., Kyiv, 03187, Ukraine

MATCHING BASED MULTISTYLE LICENSE PLATE RECOGNITION

Introduction. *A State-of-the-Art of license plate (LP) recognition from images is observed. Despite the fact that License Plate Recognition (LPR) is often regarded as a solved task, country-specific systems are mostly designed that limits their application. Pay attention to the increasing mobility, effective LPR systems must handle multistyle LP including multinational ones that have different fonts and syntax. Another bottleneck of LPR is that accuracy of recognition at varying environmental conditions as well as of low resolution or degraded LP usually is rather low.*

The purpose *of the article is to develop algorithms for multistyle single line LP learning and recognition from images as well as for comparatively low resolution LP processing.*

Methods. *Randomized Hough transform is used for detecting horizontal frame lines and subsequent LP localization in image. Structural feature matching approach is used for character recognition. Correction of recognition results is based on calculation of modified Levenstein distance (MGED) between LP description and templates.*

Results. *New algorithms for multinational license plate learning and recognition from images are proposed. Localization of LP in images is based on LP frame detection using a randomized Hough transform to detect horizontal contour frame line segments. Recognition of segmented characters inside LP is based on searching key points in skeletonized character images and matching these points with etalons. Correction of recognition LP output is carried out by matching and defining MGED between LP input description and templates. Online active learning for recognition of new LP symbols and templates is also proposed. Results of testing developed algorithms and software are described.*

Conclusions. *Algorithms for multistyle LP localization and recognition from images are proposed. Control and correction of recognition results is based on calculation of MGED between input LP description and templates which are more general in comparison conventional text lines. As future work, it is planned to increase accuracy by learning feature etalon weights, as well as to consider other LP types for recognition and to test developed means on more representative data samples.*

Keywords: *license plate localization and recognition, key points matching, Levenstein distance.*

INTRODUCTION

The task of automatic search and recognition of license plates of cars (LP) in images (LPR — license plate recognition) is a special case of recognition of text information in the world around us. This task is considered more complicated in comparison with the recognition of texts on paper, since it must be solved in uncontrolled conditions and with limited possibilities for using reference data to correct recognition results. LPR systems [1–6] are actively used to automate payment for travel and parking, fix accidents, track vehicles and solve other important problems. The development of new, more effective LPR tools remains relevant for the following main reasons:

1. Reliability of LP recognition in uncontrolled conditions is relatively low (60 % – 80 % according to [4]) due to the influence of interfering factors (speed, weather conditions, changes in illumination, various angles of vision). Recognition accuracy also depends on the camera-to-car distance: the well-known LPR suggest that the size of the character in the image should be at least 25–30 pixels, and the size of the license plate should be 100–130 pixels [1–2].
2. The number of license plate styles (types) is constantly growing, but most of the well-known LPR systems are designed to recognize national LP of their countries and have limited options for online learning and recognition of new LP types or styles.

Most LPRs consist of three main parts: 1) LP localization in the image, 2) segmentation of the LP image into separate characters, 3) segmented character recognition using optical character recognition (OCR). The OCRs used are usually developed on the basis of learning (neural networks, SVM, AdaBoost) [1–6] using a large number of positive and negative examples of symbol images. As a result, learning requires a lot of time for the preparation of input data and subsequent implementation. So, e.g., in [6], for the deep learning of a neural network to recognition of two types Brazilian license plates was used about 4000 allocated and marked by operators LP images (40 % of the total number, recognition accuracy on the control part of the set — 85 %). To search for LP in images, another neural network was used, previously learned on another training set, and the authors consider the learning to recognize other types of characters and symbol fonts [6] as a rather complicated task that will need to be solved in the future.

When performing the work, two main goals were pursued: 1) the development of online learning tools for recognizing various types of single-line LPs and 2) the recognition of relatively small size noisy LPs. The work is going on of [7] — new algorithms are considered: 1) preprocessing and segmentation of LP into symbols; 2) character recognition based on matching of so-called "key points" on the skeletonized image representation [8]; 3) correction of recognition results based on the modified Levenshtein edit distance (ED) technique [9].

LP recognition accuracy can be significantly increased on the base of calculation the ED between the input text line and each of the reference LP lines that can be in view of the camera. This method can be used only in some especial cases — creating and maintaining a database of all cars is almost impossible, given the huge number of cars used. In this regard, another correction is considered, based on comparison of the input LP description with LP type templates.

In each country, a relatively small number of templates (types) of national license plates are used, which specify most of the license plates used and facilitate the task of recognizing them. The description of the template can be presented in the form of a list of minimal rectangles, such that each of them bounds LPs character and has an attribute defining one or more valid codes for this character. In the simplest case, this attribute takes two values — '0' if character is a digit, and '1' otherwise. If we neglect size and relative positions of the characters, we get a description in the form of a text string consisting of 0 and 1. The correction of the output text line can be performed based on the calculation of the modified ED (MGED) between the LP input description and the template descriptions taking into account the sizes and relative arrangement of the constituent characters. The weights of editing operations (deleting, replacing and inserting characters) when calculating MGED, in contrast to [9], are not constant values, but depend on characters, the reliability of their recognition and position in the line. The MGED features will be considered in more detail below in Section 3. The LP localization in image and recognition of the LP characters are performed uniformly for all recognized types of license plates, the structure of which is taken into account only at the stage of MGED calculation between the recognized string and license plate templates.

Learning to LP recognition from images can be performed in offline, online or online-offline modes. In the offline mode, the learning process is close to [1–3, 5–6] — the operator generates training samples and defines LPs templates followed by adjusting of etalons feature weights in the best way to classify training images. In the online mode, the formation of symbol etalons and LP templates is performed during the operation of the system. In the combined online-offline mode, feature weights of etalons are further adjusted.

The rest of the paper is organized as follows. Section 1 discusses the LP localization algorithm in images, and in Section 2 algorithms for preprocessing LP images, segmenting them into individual characters, and recognizing these characters are considered. Section 3 contains descriptions of LP templates and the MGED between LP descriptions. Section 4 describes LPR learning and Section 5 presents the results of preliminary testing of the developed tools. In conclusion, a discussion of results and directions for further research is given.

LP LOCALIZATION IN IMAGE

The main stages of the localization and recognition of license plates in images are presented in Fig. 1.

At the initial stage, LP localization in image is carried out using the randomized Hough transform for horizontal contour lines detection of the LP frame [10, 11]. To avoid detection of these lines in LP characters area, preliminary image erosion operation is performed in the horizontal direction (to erode areas with low brightness). The vertical lines of the frame are detected by tracing vertical edges in the image. The result of the LP localization is a certain number of quadrangles (FS), ordered by the conformity assessment estimates of the LP frame. The value of this estimate depends on the editing operations of the contour line segments during the formation of the FS, the difference in the angles of inclination of the opposite sides, as well as the ratio of the lengths of the horizontal and vertical sides.

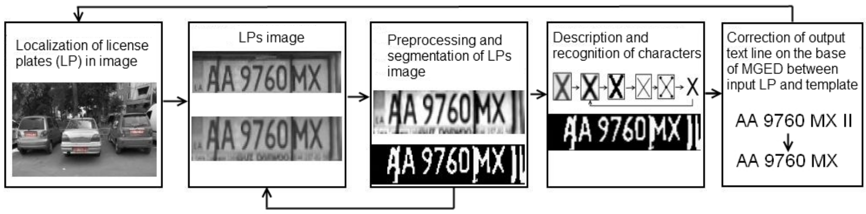

Fig. 1. Localization and recognition of license plates in images.

Hough transform is a relatively time-consuming operation. Therefore, in order to speed up the search, the contrasty image areas, which may include car numbers, are preliminarily found, and only then, within these areas, more accurate LP localization based on the Hough transform is performed. The main feature of such areas is a large number of vertical edges of the contour. The search for these areas is performed using the so-called "integral" image representation [12] — in each pixel of the integral image the total number of contour points located to the left and above this pixel is stored. The integral contour image lets to calculate the number of contour points in any rectangle using several operations.

PREPROCESSING AND RECOGNITION OF LP IMAGE

At subsequent stages, processing and recognition of the selected image parts with the highest values P is performed. First, the image is processed to transform quadrangle to a rectangle with the given dimensions. After that, normalization of the image by brightness, adaptive image binarization and character segmentation are performed [7]. Segmentation of LP image into separate areas is performed by selecting connected objects in a binary image and leaving as candidates for characters those ones whose sizes and additional parameters satisfy certain restrictions. If the size of the area significantly exceeds the average size of the characters, the operation of dividing this area into several parts is performed based on the use of horizontal projections of the binarized and brightness-normalized image of this area. An example of segmentation and binarization of characters in a license plate image is shown in Fig. 2, and examples of processing an image of a single character are shown in Fig. 3.

Character image recognition consists of the following basic operations:

1. Normalization by brightness, binarization and scaling of the input image within the symbol area (Fig. 3).

2. Skeletonization of the symbol image [8] and detection the so-called "key points (KP)" (Fig. 4) on symbol skeleton representation. The main KPs are the points of ends or intersections of the symbol lines, as well as the points of changing the bypass direction (clockwise or counterclockwise) of its skeleton (e.g., the point in the middle of the symbol 'S'). Additional KPs are the central points of the inner contours in symbol images such as '0', 'O', 'D' or 'P'. Generating description of each KP: normalized values of its coordinates, type and weight (from 0 to 1) KP , as well as the directions of the skeleton line segments emerging from the main KP. Symbol description includes its KPs descriptions, width and height of the minimum rectangle bounding the symbol, as well as features of fragments (a line or curve indicating its position relative to the KP pair), between some basic KPs.

3. Calculation of the distance between the input description and the set of reference descriptions of symbols used. If this distance is less than the threshold value, the name of the reference corresponding to this distance is assigned to the input symbol. Otherwise, the above operations (items 1–3) are performed once more, but with the other parameters of preprocessing and noise removing on the skeletonized representation of the symbol (feedback between the result of recognition of the symbol and its preprocessing in Fig. 1). In both cases, the recognition confidence (0–100) is calculated based on the obtained distance and the threshold used.

Fig. 2. Position adjustment (b) of the input LP image (a). Brightness normalization (c) and binarization (d) of the image (b).

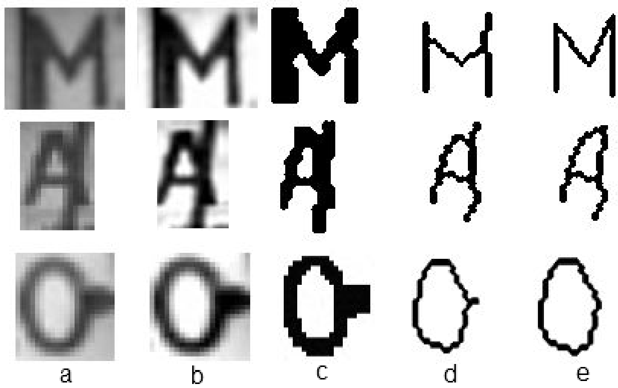

Fig. 3. Examples of brightness normalization (b), binarization (c), skeletonization (d) and noise removing (e) of character images (a).

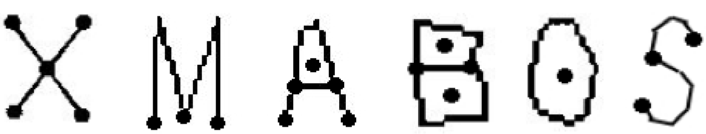

Fig. 4. Key points examples on skeletonized representations of symbols.

In the online learning mode weights of key points depend on their types and conditions used of detection in the symbol image. In two other modes, the point weights are adjusted during learning on a set of image examples. The distance between the two compared characters is defined on the base of matching of their KPs. Let $P_1 = \{p_i^1\}, i = 1,...n_1$ and $P_2 = \{p_j^2\}, j = 1,...n_2$ be sets of key points in two symbols; $R(p_i^1 \in P_1)$ — correspondence function: $R(p_i^1 \in P_1) = p_j^2 \in P_2$, if p_i^1 corresponds to p_j^2, и $R(p_i^1) = 0$ otherwise; matching — $M(P_1, P_2) = \{(p_i^1, p_j^2) \mid R(p_i^1) = p_j^2\}$.

Maximal matching $M_m(P_1, P_2)$ is the matching with size equaled to $n = |M_m(P_1, P_2)| = \min(n_1, n_2)$ value. Distance $D(P_1, P_2) = (D_{del} + D_{cor}) / w$ between two symbols descriptions is defined on the base of finding optimal matching $M^*(P_1, P_2)$ of KPs:

$$M^{\bullet}(P_1, P_2) = \underset{M_m(P_1, P_2)}{\arg\min} D_M = \sum_{k=1}^{n} (w_i^1 + w_j^2) d((p_i^1, p_j^2)_k \in M_m(P_1, P_2)),$$

where $d(p_i^1, p_j^2)$ — distance between two KPs and $(w_i^1 + w_j^2)$ — sum of these points weights; $D_{del} = K_{del} w_{del}$ — cost of deletion $|n_2 - n_1|$ non matched points; w_{del} — sum of weights of deleted KPs; w — total sum of KPs weights; K_{del} — deletion cost of KP with weight, equaled to 1; D_{cor} — value D_M that corresponds to optimal matching $M^*(P_1, P_2)$.

The distance between two KPs depends on the differences in their coordinates and types, as well as in the directions of the corresponding line segments emanating from these points. To make decision about the input symbol it is necessary to define the distance of its description to the set of reference descriptions. Therewith, the input symbol belongs to a certain class if the distance of the description of the symbol to set of reference samples of this class is the smallest among other classes and less than the specified threshold value.

The distance D_{cor} (therefore, $D(P_1, P_2)$) can be calculated using optimal algorithms [13–15] or algorithm [7], which belongs to type of so-called "greedy" algorithms ($O(n_1 n_2)$). As a result, the correspondence (optimal matching) of the KPs of two compared symbols is defined, which is used to verify features of the traced skeleton segments between the pairs of corresponding main KPs of these symbols.

MODIFIED LEVENSTEIN DISTANCE AND LPs TEMPLATES

The Levenshtein distance (ED — edit distance) $lev_{a,b}$ between two lines of characters a, b is equal to the minimum total cost of deleting, inserting and replacing characters of the first line to convert to the second one [9]. Let C_{del}, C_{ins}, $C_{sub}(a_i \neq b_j)$ be the costs of deleting, inserting, and replacing (0, if $a_i = b_j$, and 1 otherwise) a single character. Then the distance $lev_{a,b}(i, j)$ between the first i characters of the text line a and the first j characters of the text line b can be calculated using the following formulas:

$lev_{a,b}(i, j) = \max(i, j)$, if $\min(i, j) = 0$, and

$lev_{a,b}(i, j) = \min(lev_{a,b}(i-1, j) + C_{del}, lev_{a,b}(i, j-1) + C_{ins}, lev_{a,b}(i-1, j-1) +$
$+ C_{sub}(a_i \neq b_j)$, if not.

The ED between two lines is equal to the minimum number of operations to delete, insert, and replace characters in one line to convert it to the second one and is a measure of the proximity of two lines. This measure is not always effective in solving problems of searching, comparing, and correcting texts;

therefore, in [13–15] more general types of ED were proposed for assessing the proximity of compared strings. In [13] the weight of operations depends on the names of symbols and operations (generalized edit distance — GED), in [14] in addition to [13] an extended list of editing operations is proposed. In [16] a more general case (Markov edit distance — MED) was considered to evaluate the likelihood of each sequence of editing operations and it was shown that ED and GED are particular cases of MED. The book [17] contains a detailed analysis of the most common issues related to the use of ED. In this paper we propose the modified Levenshtein distance (MGED) to evaluate the proximity of the description of the LP recognition result not only to the reference lines, but also, unlike [1, 4, 5, 9, 13–15] to the LP templates.

Let $R=\{r_i\}, i=1,...n$ be the set of minimal rectangles bounding the images v_i of symbols s_i on a license plate; N_S — the number (size of the alphabet) of the characters used; $E_j=\{e_{j,k}\}, j=1,...N_S; k=1,...n_j$ — set of etalons (references) of symbol s_j; $d_{i,j}(v_i,E_j)=\min\limits_k dist(v_i,e_{j,k}\in E_j)$ — distance between image v_i and symbol s_j; $dg(s_j)$ — a feature equal to 1 if s_j is a digit, and 0 otherwise.

Consider the following three types of LP descriptions. The first description is a list of rectangles, to each $r_i \in R$ of which correspond N_S distances between v_i and symbols. The second description is different in that to each $r_i \in R$ two pairs of values correspond: 1) the recognized character s_i and corresponding to it the smallest distance ($d_{i,j1}$); 2) $(s_i^{'}, d_{i,j2}=\min\limits_{j\neq j1} d_{i,j} \mid dg(s_i)\neq dg(s_i^{'}))$. The third description differs from the first one in that it does not contain data on the size and coordinates of the recognized characters.

Consider also the corresponding LP reference descriptions. The first description defines the LP template as a set of rectangles, within each of which there can be one character from the valid lists. The second description differs from the first in that to each rectangle has not a symbol, but a feature (1 or 0) that this symbol should be a number or a letter. The third description represents a text string of characters. The MGED value represents the minimum cost of changing the input description to bring it into correspondence (matching) with the LP reference description. Similarly to GED [13], MGED uses the operations of replacing, inserting and deleting characters, the cost of which depends on the names of the characters being edited and the type of operation, but there are also differences from [9, 13–15].

Using the description of the first type, the cost of replacing the s_i character in the input description with the s_j character in the reference is equal to $C_{sub}(s_i,s_j)=\min\limits_{k\in S_J} d_{i,k}(v_i,E_k)+ds(i,j)$, where S_j is the list of numbers of valid characters for replacement and $ds(i,j)$ is the penalty for differences in sizes and relative coordinates of the characters s_i and s_j. Using the description of the second type, $C_{sub}(s_i, s_j) = d_{i,j1}(v_i, E_{j1}) + ds(i, j)$, if $dg(i) = dg(j)$, and $C_{sub}(s_i, s_j) = d_{i,j2}(v_i, E_{j2}) + ds(i, j)$ otherwise. The cost of replacing one character with another is lower, the closer these characters are in their drawing, size and position on the LP relatively to other characters. For example, the cost of replacing '5' with 'S' is much lower than replacing '5' with '8' or '4'. Data on the proximity of individual pairs of characters is either predefined or generated during the operation of the recognition system.

The cost of deleting a symbol depends on its similarity with the nearest reference (increases with increasing similarity), as well as on the position of the symbol in the line (decreases at the beginning and end of the line, since there is a greater chance of interference). The cost of inserting a character is small if there is a gap in the LP image at the place of insertion that is not covered by recognized characters. This symbol cannot be taken from the templates of the first two types — when using them, the missing symbol can be determined only by additional recognition of the license plate in defined part of the input image, and the correction of the results should be performed not outside the LPR, but in the process of recognition.

LP RECOGNITION LEARNING

LP learning can be performed in offline, online or online-offline modes. In the first case, the learning process is close to [1–3, 5, 6] — the operator generates a training set of images, and also defines LP templates. Therewith, due to the use of informative features and essential part of noise removing at the processing stage, smaller training set can be used in comparison with most other works. In subsequent learning weights of character feature weights are adjusted to minimize image recognition errors from the training set.

In the online mode, the formation of symbols references and LP templates is performed during the system operation. In this case, the weights of the symbols features depend on the conditions used of their detection in the image and are not formed on the base of learning. Before learning it is assumed that there is a rectangular frame on the LP image and no data on the characters and type format of recognized license plates. LP search, segmentation and recognition of characters is performed uniformly for all LP types, the structure of which is taken into account only at the stage of MGED calculation between the recognized string and LP templates. If the program cannot recognize the symbol, the image of this symbol is presented to the user for identification and creation of an additional reference in taking an appropriate decision. Each group of recognized characters is checked by the program for matching with previously added LP templates and, in the absence of close matching, is presented to the operator for taking a decision on adding a new LP template. The input LP image and its recognition result can also be presented to the user for verification and correction of these results. The indicated actions can be considered as the process of the so-called "active learning" of the program for recognizing new characters and license plate patterns in images. In the online-offline mode, automatic adjustment of the features weights of the references is performed after learning recognition in online mode. After completing the learning process, the program switches to a mode in which insufficiently reliable results are stored in memory and can be viewed by the user at a convenient time for him.

EXPERIMENTAL TEST RESULTS

During the testing, LP were searched and recognized on the following two sets of images. The first set (B1) consists of 175 images in which cars are photographed at various distances and angles with respect to the camera. This set contains 19 types of license plates, including three types of Ukrainian numbers, seven — Polish, three — Belarusian, in two — Russian, Moldavian and Lithuanian. The second set contains 85 images of two types of Greek numbers from a database [2]. Of these 85 images, 65 are simpler to recognize (sample "day color" — B2) and 20 are most complex of this database (sample "difficult_color_more_than_one" — several cars, different shooting angles, lighting conditions, small size LP — B3). During the testing, the LP image was considered correctly recognized if all characters of the text line on this LP image were correctly recognized. Examples of images from samples B1, B2, B3 (by two images in the upper, middle and lower parts) and the results of their recognition are presented in Fig. 8.

Images from sample B2 in Fig. 8 (middle part) have interference in the form of touching the characters to the frame (left image) and varying lighting. Images from sample B3 in Fig. 8 (lower part) have a horizontal size of less than 65 cells. At the same time, five out of six LPs are recognized correctly, one of the LPs (left image) is detected, but not recognized. Main recognition results:

1. Recognition accuracy on sets B1–B2 is 97 % and on set B3 is 72 %, provided that the horizontal size of the LP in the image is not less than 65 pixels. Character references and descriptions of the LP templates were obtained during online learning.
2. For learning in online mode, it is enough to have a relatively small number of examples of LP types and the characters used. The number of character references after learning is from one to four for different characters.
3. The average processing time for an input image of 800x600 and 1600x1200 is respectively 0.14 sec. and 0.52 sec.

The database [2] was created in 2008 to provide the possibility of comparing different LPRs on the same data, since usually the developer provides recognition results on his data that are not available for testing other LPRs. However, we are not aware of publications containing comparative test results for recognizing systems on this database.

Fig. 8. Examples of localization and recognition of license plates in images from test samples B1, B2 and B3 (by two images in the upper, middle and lower parts).

The introduced restriction of 65 pixels on the minimal horizontal size of the LP in images is relatively weak. So, for example, the "Nomerok-4" system [18] provides 95 % of the correct recognition of state Ukrainian and Russian LP when the following conditions are met: 1) the speed of the car is not more than 120 km / h; 2) the horizontal size of LP in the frame is at least 130 cells; and 3) LP in frame can be recognized by the operator. The developers of the SecurOS Auto intelligent video analysis system [19] claim to recognize with reliability up to 96 % of state license plates of vehicles moving at speeds up to 180 km / h, without providing data on the fulfillment of conditions for recognition and used restrictions on the size of the number in the frame. With a decrease in the size of the license plate, the task of its localization and recognition in the image becomes more complicated. On the other hand, recognition of LP with small sizes provides the possibility of further video surveillance and control.

CONCLUSION

Algorithms for localizing, processing and recognizing LP in images are considered. Algorithms and software for online learning to recognition new characters and LP templates are developed. Recognition of each character is performed based on the detection of key points on the skeletonized representation of the character image and matching them with the key points on each of the compared reference images. The minimum allowable size of LP in the image is 65 pixels, which provides the possibility of more distant video surveillance compared to other systems.

The recognition results are controlled and corrected based on the calculation of the modified Levenshtein distance between the description of the input LP and the reference descriptions of LP templates. These descriptions are more general than usually used text strings on these LP, which are available only for some applications. In subsequent studies, it is planned recognition also of two-line types of LP and increasing reliability by additionally adjusting the weights of symbol reference features during learning. It is also necessary to implement LP recognition in the video stream and test the developed tools on more representative data samples.

REFERENCES

1. Shan D., Mahmoud I., Shehata M., Badawy W. Automatic License Plate Recognition (ALPR): A State-of-the-Art Review. *IEEE Transactions on circuit and systems for video technology*. 2013, Vol. 23, no. 2, pp. 311–325.
2. Anagnostopoulos C.N.E., Anagnostopoulos I.E., Psoroulas I.D., Loumos V., Kayafas E. License plate recognition from still images and video sequences: A survey. *IEEE Trans. Intell. Transp. Syst.* 2008, Vol. 9, no. 3, pp. 377–391.
3. Anagnostopoulos C.N.E., Anagnostopoulos I.E., Loumos V., Kayafas E. A license plate-recognition algorithm for intelligent transportation system applications. *IEEE Transactions on Intelligent Transportation Systems.* 2006, 7 (3), pp. 377–392.
4. Oliveira-Neto F.M., Han L.D., Jeong M.K. An Online Self-Learning Algorithm for License Plate Matching. *IEEE Transactions on intelligent transportation systems*. 2013, Vol. 14, no. 4, pp. 1806–1816.
5. Thome N., Robinault L. A cognitive and video-based approach for multinational license plate recognition. *Mach. Vision Application*. 2011, Vol. 22, no. 2, pp. 389–407.
6. Laroca R., Severo E., Zanlorensi L.A., Oliveira L.S., Gonçalves G.R., Schwartz W.R., Menotti D. A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector. 2018 International Joint Conference on Neural Networks (IJCNN), (Rio de Janeiro, 8-13th of Jul, 2018). Rio de Janeiro, Brazil, 2018, pp. 1–10.
7. V.M. Kyyko. Localization and recognition of license plates from images. *Upravlâûŝie sistemy i mašiny*. 2017, no. 6, pp. 26–34, 40 (in Russian).
8. Kyyko V.M., Schlesinger M.I. Width — Independent Fast Skeletonization Algorithm for Binary Pictures. *Intern. Journal of Imaging Systems and Technology*. 1991, Vol. 3, no. 3, pp. 222–226.
9. Levenshtein V.I. Binary codes capable of correcting deletions, insertions, and reversals. *Doklady Academy of sciences USSR*. 1965, no. 163 (4), pp. 845–848 (in Russian).
10. Duda R.O., Hart P.E. Use of the Hough Transformation to Detect Lines and Curves. *Pictures. Comm. ACM*. 1972, Vol. 15, pp. 11–15.
11. Kiryati N., Eldar Y., Bruckstein A.M. A Probabilistic Hough transform. *Pattern Recognition*. 1991, no.24, (4), pp. 303–316.
12. Schlesinger M.I. Fast implementation of one class of linear convolutions. *Theoretical and applied issues of image recognition*. 1991, Kyiv, pp. 61–69 (in Russian).

13. Ocuda T., Tanaka E., Kasai T. A method for correction of garbled words based on the Levenstein metric. *IEEE Transactions on Computers*. 1976, no. 25 (C–2), pp. 172–177.
14. Seni G., Kripasundar V., Srihari R. Generalizing edit distance to incorporate domain information: handwritten text recognition as a case study. *Pattern Recognition*. 1996, no. 29 (3), pp. 405–414.
15. Jonker R., Volgenant A. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing.* 1987, no. 38, pp. 325–340.
16. Wei J. Markov edit distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2004, no. 6(3), pp. 311–320.
17. Schlesinger M., Hlavac V. Ten Lectures on Statistical and Structural Pattern Recognition. Dordrecht / Boston / London, 2002, 520 p.
18. URL: http://viatec.ua/product/nomerok-4
19. URL: http://isscctv.com.ua/?page=SecurOSAuto

Received 11.11.2019

ЛІТЕРАТУРА

1. Shan Du, Mahmoud Ibrahim, Mohamed Shehata and Wael Badawy. Automatic License Plate Recognition (ALPR): A State-of-the-Art Review. *IEEE Transactions on circuit and systems for video technology*. Vol. 23. No. 2. 2013. P. 311–325.
2. Anagnostopoulos, C.N.E., Anagnostopoulos, I.E., Psoroulas, I.D., Loumos, V., Kayafas. License plate recognition from still images and video video sequences: A survey. *IEEE Trans. Intell. Transp. Syst.* Vol. 9. No. 3. 2008. P. 377–391.
3. Anagnostopoulos, C.N.E., Anagnostopoulos, I.E., Loumos, V., Kayafas, E., A license plate-recognition algorithm for intelligent transportation system applications. *IEEE Transactions on Intelligent Transportation Systems.* 7 (3). 2006. P. 377–392.
4. Francisco Moraes Oliveira-Neto, Lee D. Han, and Myong Kee Jeong. An Online Self-Learning Algorithm for License Plate Matching. *IEEE Transactions on intelligent transportation systems*. Vol. 14. No. 4. 2013. P. 1806–1816.
5. N. Thome and L. Robinault. A cognitive and video-based approach for multinational license plate recognition. *Mach. Vision Application*. Vol. 22. No. 2. 2011. P. 389–407.
6. R. Laroca, E. Severo, L.A. Zanlorensi, L.S. Oliveira, G.R. Gonçalves, W.R. Schwartz, and D. Menotti. A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector. 2018 International Joint Conference on Neural Networks (IJCNN), (8th–13th of July, 2018, Rio de Janeiro). Rio de Janeiro, 2018. Brazil. P. 1–10.
7. В.М. Кийко. Локализация и распознавание автомобильных номеров на изображениях. *Управляющие системы и машины*. No 6. 2017. С. 26–34, 40.
8. V.M. Kiyko and M.I. Schlesinger. Width — Independent Fast Skeletonization Algorithm for Binary Pictures. *Intern. Journal of Imaging Systems and Technology*. USA. Vol. 3, N 3. 1991. P. 222–226.
9. В.И. Левенштейн. Двоичные коды с исправлением выпадений, вставок и замещений символов. *Доклады АН СССР*. 163 (4). 1965. С. 845–848.
10. R.O. Duda and P.E. Hart. Use of the Hough Transformation to Detect Lines and Curves. *Pictures. Comm. ACM*. Vol. 15. 1972. P. 11–15.
11. N. Kiryati, Y. Eldar and A.M.Bruckstein. A Probabilistic Hough transform. *Pattern Recognition*. 24 (4).1991. P. 303–316.
12. М.И. Шлезингер. Быстрая реализация одного класса линейных сверток. *Теоретические и прикладные вопросы распознавания зображений.* Киев, 1991. С. 61–69.
13. T. Ocuda, E. Tanaka, T. Kasai. A method for correction of garbled words based on the Levenstein metric. *IEEE Transactions on Computers*. 25 (C–2). 1976. P. 172–177.
14. G. Seni, V. Kripasundar, R. Srihari. Generalizing edit distance to incorporate domain information: handwritten text recognition as a case study. *Pattern Recognition*. №29 (3). 1996. P. 405–414.
15. R. Jonker, A. Volgenant. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*. № 38. 1987. P. 325–340.
16. J. Wei. Markov edit distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 6(3). 2004. P. 311–320.

17. М. Шлезингер, В. Главач. Десять лекций по статистическому и структурному распознаванию. Киев, 2004. С. 545.
18. http://viatec.ua/product/nomerok-4
19. http://issctv.com.ua/?page=SecurOSAuto

Отримано 11.11.2019

Кийко В.М., канд. техн. наук, старш. наук. співроб.
відд. розпізнавання образів
e-mail: vkiiko@gmail.com
Міжнародний науково-навчальний центр інформаційних
технологій та систем НАН України і МОН України,
пр. Акад. Глушкова, 40, м. Київ, 03187, Україна

РОЗПІЗНАВАННЯ НОМЕРНИХ ЗНАКІВ НА ЗОБРАЖЕННЯХ
З КОРЕКЦІЄЮ РЕЗУЛЬТАТІВ

Вступ. Розглянуто сучасний стан розпізнавання номерних знаків (НЗ) автомобілів на зображеннях. Відомі системи розпізнавання номерних знаків (СРНЗ) зазвичай забезпечують порівняно високу надійність у сприятливих умовах тільки найпоширеніших (національних) типів НЗ у кожній країні. Можливості цих систем для оперативного налаштування на розпізнавання нових типів НЗ та символів є доволі обмеженими, що ускладнює їхнє застосування. Беручи до уваги швидке збільшення мобільності пересування транспортних засобів, у багатьох роботах зазначається, що найактуальнішими завданнями для розроблення сучасних СРНЗ є розпізнавання різних типів НЗ, особливо в складних та неконтрольованих умовах змінення зовнішнього освітлення та наявності різних завад як на самих НЗ, так і у навколишньому середовищі.

Метою роботи є розроблення алгоритмів пошуку і розпізнавання різного вигляду однорядкових НЗ автомобілів на зображеннях в умовах наявності завад та порівняно малих розмірів НЗ на зображеннях.

Методи. Рандомізоване трансформування Хафа використовують для пошуку горизонтальних контурних ліній рамки НЗ та наступної локалізації НЗ на зображеннях. Розпізнавання сегментованих символів на зображенні НЗ виконується шляхом пошуку та розмітки так званих «особливих точок» на скелетизованих поданнях цих символів. Корекція результатів розпізнавання реалізують шляхом обчислення модифікованої відстані Левенштейна (МВЛ) між вхідним описом та шаблонами НЗ.

Результати. Розглянуто нові алгоритми пошуку і розпізнавання різного вигляду однорядкових номерних знаків автомобілів на зображеннях. Локалізація НЗ виконується за допомоги пошуку горизонтальних контурних ліній рамки номера за методом Хафа. Розпізнавання сегментованих зображень символів на НЗ реалізовано шляхом виділення «особливих точок» на скелетизованих поданнях символів і пошуку відповідності цих точок еталонним описам. Контроль і корекція результатів розпізнавання НЗ виконують на основі обчислення МВЛ між вхідним описом НЗ і шаблонами типів НЗ. Запропоновано та реалізовано засоби активного навчання під контролем оператора розпізнаванню нових типів НЗ та символів у процесі роботи системи. Наведено результати тестування розроблених алгоритмів у разі розпізнавання НЗ різних країн.

Висновки. Запропоновано нові алгоритми локалізації та розпізнавання різних типів НЗ. Контроль та корекція отриманих результатів базується на обчисленні модифікованої відстані Левенштейна вхідного опису НЗ до множини шаблонів типів НЗ, які мають більш загальний вигляд порівняно з еталонними текстовими рядками. У подальшому виконанні роботи заплановано розгляд алгоритмів розпізнавання НЗ, які містять декілька текстових рядків, а також підвищення надійності розпізнавання шляхом оптимального налаштування за допомогою навчання вагів особливих точок символів.

Ключові слова: *пошук та розпізнавання номерних знаків на зображеннях, відповідність особливих точок, відстань Левенштейна.*

Кийко В.М., канд. техн. наук, старш. науч. сотр.
отд. распознавания образов
e-mail: vkiiko@gmail.com
Международный научно-учебный центр информационных
технологий и систем НАН Украины и МОН Украины,
пр. Акад. Глушкова, 40, г. Киев, 03187, Украина

РАСПОЗНАВАНИЕ НОМЕРНЫХ ЗНАКОВ НА ИЗОБРАЖЕНИЯХ
С КОРРЕКЦИЕЙ РЕЗУЛЬТАТОВ

Рассмотрены алгоритмы поиска и распознавания различного вида однострочных номерных знаков (НЗ) автомобилей на изображениях. Локализация НЗ выполняется с помощью поиска горизонтальных контурных линий рамки номера методом Хафа. Распознавание сегментированных изображений символов на НЗ реализовано путем выделения особых точек на скелетизированных представлениях символов и поиска соответствия этих точек с эталонными описаниями. Контроль и коррекция результатов распознавания НЗ выполняется на основе вычисления модифицированного расстояния Левенштейна между входным описанием НЗ и шаблонами типов НЗ. Приведены результаты тестирования разработанных алгоритмов при распознавании НЗ из различных стран.

Ключевые слова: *поиск и распознавание номерных знаков на изображениях, соответствие особых точек, расстояние Левенштейна.*