Implementation of an intelligent decision support system to accompany the manufacturing process

Vladimir Golovko
Alexsander Kroshchanka
Brest State Technical University
Brest, Belarus
gva@bstu.by
kroschenko@gmail.com

Mikhail Kovalev
Belarusian State University of Informatics an Raddioelectronics
Minsk, Belarus
michail.kovalev7@gmail.com

Valery Taberko
Dzmitry Ivaniuk
JSC "Savushkin product"
Brest, Belarus
id@pda.savushkin.by

***Abstract*—The article reviews the implementation of an intelligent decision support system in case of solving real manufacturing problems of JSC "Savushkin Product" for quality control of marking products. The proposed system integrates neural network and semantic models. Also, article proposes original method of localization and recognition of images from a high-speed video stream, as well as a method of intelligent processing of recognition results for developing decision to manufacturing problems.**

***Keywords*—deep neural networks, object detection, knowledge base, integration, inference, IDSS**

INTRODUCTION

The modern direction of industrial development is moving towards increasing automation and robotization of manufacturing. Tasks are set to build autonomous work not only of individual parts of manufacturing, but also of entire plants. Setting and solving such problems in the modern world is called the fourth industrial revolution [1], the development of which is associated with the concept of Industry 4.0 and the Internet of things(IoT) [2].

Under these conditions, the complexity of managing manufacturing [3] increases. There is a need to make decisions not only in the manufacturing process, but also in the process of its adjustment and debugging. Building intelligent systems that can make or offer decisions in the manufacturing process is a promising applied area of artificial intelligence.

The use of intelligent decision support systems(IDSS) can significantly simplify the decision-making process in the workplace for a person [4]. Such systems also allow to make decisions independently and, if necessary, to justify the reasons for the decision. However, implementation of this feature requires integration of different problems solving models [5].

In this article, we consider the modification and implementation of the intelligent decision support system based on the integration of neural network and semantic models described in [6]. The application of this system is considered on the example of the problem of JSC "Savushkin product": quality control of marking on products.

Figure 1. Example of a frame from a video stream

I. FORMULATION OF THE PROBLEM

We were given the task of checking the correctness of the marking printed on products manufactured by JSC "Savushkin product".

The source data is a video stream obtained by an RGB camera mounted above the manufacturing tape. In our example, this stream is characterized by a speed of 76 frames per second. An example of a frame from a stream is shown in Fig. 1.

The solution to this problem includes:

- marking detection and recognition;
- the identification of marking issues;
- search for the causes of these problems;
- search decision to these problems;
- applying a decision independently, if possible.

Problems with marking can be as follows:

- **no paint.** If empty bottles start to go, then the printer has run out of paint. The system can also refer to it (since the printer is connected to the network) and check the availability of paint.
- **camera shift** the system knows that the batch has started filling, but there are no positive recognition results from the camera.
- **incorrect marking.** The marking is recognized, passed to the system, but it does not match the standard – this means that an error occurred when setting the marking text and it is necessary to stop the filling process and notify the operator.
- **unreadable marking.** Marking is not recognized - several digits are not recognized, so the printer nozzles are clogged - it is needed to stop the filling process and notify the operator to clean the printer nozzles. In this case, it is needed to remove bottles with incorrect marking from the conveyor.

In general, we set the following requirements for the system that will solve the problem:

- **High-speed operation.** Manufacturing processes are very fast, so the search for incorrectly marked products and its rejection also must be very fast.
- **Ability to explain.** It is necessary to identify not only incorrect marking, but also to explain the reasons for this situation.
- **Autonomy.** The system should minimize human involvement in the quality control process. Of course, there may be situations when system can't handle problem without operator help, but even in such situations, the system must be able to instruct the operator and be able to explain the reasons for certain actions.
- **Adaptability.** It is necessary to have ability to adapt the system to recognize markings on any other products.

II. Existing approaches

The task of controlling the quality of marking at enterprises similar to the enterprise of JSC "Savushkin Product" is often solved manually. A human periodically selectively checks a part of the product. This approach has the following disadvantages:

- only part of products are checked;
- there is a possibility that a human will not notice a slight discrepancy between the checked marking and the standard one;
- the use of monotonous manual labor.

At the moment, developments to automate these activities are underway and are being implemented, but in most cases we are only talking about identifying problems with marking, but not at all about finding the causes and solutions to these problems.

For example, Omron [7] sensors are often used. These sensors are equipped with a camera and are able to recognize product markings on high-speed tape. Using these sensors allows to automate the work of a person for quality control, but there are the following disadvantages:

- Recognition quality isn't high enough. Sensors are based on Optical Character Recognition (OCR), an approach used for document recognition that is highly dependent on the image quality of the text. Due to the high speed of the manufacturing tape, it is not always possible to get a high-quality photo of the marking.
- Need to buy specialized software for system configuration.
- No built-in system for finding and fixing marking problems.

III. The proposed approach

From the formulation of the problem, it follows that the system that will solve this problem must be able to search for and propose decisions to emerging problems, as well as justify these decisions. Therefore, the proposed approach is based on the experience of building IDSS during designing this system [4].

Classic DSS can be defined as an interactive automated system that helps the decision maker (DSS) use data and models to identify and solve problems and make decisions [8]. Intelligent DSS is included to DSS concept that is characterized by using of models that are traditionally considered intelligent.

The proposed system goes beyond the definition of IDSS, since the system can not only offer, but also apply decisions. For this purpose, system uses the integration of neural network and semantic models.The proposed system goes beyond the definition of IDSS, since the system can not only offer, but also apply decisions. For this purpose, system uses the integration of neural network and semantic models.

Neural network models are responsible for localization and recognition of markings, which is a non-trivial task because the manufacturing tape has high speed.

Semantic models, represented as the knowledge base(KB) based on ontologies, are responsible for search and decision-making. Based on this, the proposed system can be defined as a knowledge-driven DSS [9].

The proposed system is based on OSTIS technology and its principles. OSTIS technology uses knowledge representation and processing models focused on unification and working with knowledge at the semantic level. The main principles and models used in the approach include:

- knowledge integration model and unified semantic-based knowledge representation model [5], based on the SC code [10];
- principles of situational control theory [11];
- ontological model of events and phenomena in knowledge processing processes [12];

- multi-agent approach [13];
- hybrid knowledge processing models [14].

The main components of the system are:

- **Machine vision module.** The task of the module is localization and recognition product markings on the image and transfer to the results of this recognition to the decision-making module. Also, this module stores all trained artificial neural networks (ANNs), the description of which is stored in the KB. In the future, the module should be able to switch between trained ANNs if the engineer sets the appropriate configuration of the KB.
- **Decision-making module.**. Consists of the KB and the problem solver. The KB contains all the necessary knowledge for making and implementing decisions, such as logical rules, statements, current markings, device states, and so on. The problem solver contains a set of internal and external agents that work with the KB. These agents are used for inference for finding decisions, calling external programs for implementing decisions, and preparing decision for the engineer's terminal.
- **Robotic subsystem control module** This module has access to the subsystem that directly carries out the marking of products. The task of this module is implementation of system decisions that can be taken without the involvement of the engineer, such as marking updates, switching the subsystem on and off, etc.
- **Engineer's terminal.** The user interface module that can be used to track decisions made by the system, including decisions that require the attention of an engineer. The terminal provides full access to the KB, so the engineer has the ability to manually configure the system for certain tasks. For example, he can indicate to the system the fallacy of its reasoning and correct the rule by which it made an wrong decision.

The figure 2 shows the general diagram of interaction of the marking quality control system modules.

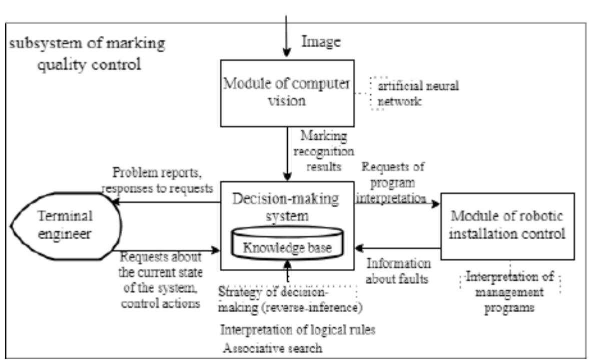

Figure 2. The general diagram of interaction of the marking quality control system modules [6]

Figure 3. Controller RFC 4072S with connected camera DFK 22AUC03

IV. Description of system hardware part

PLCnext. Testing of given program solution was produced based on controller PLCnext RFC 4072S (Fig. 3). This hardware platform is open solution of modern automation systems, which satisfy requirements of IIoT [15].

Controller PLCnext RFC 4072S has next technical parameters: processor Intel® Core™i5-6300U 2x2.4 GHz, USB-interface, 4 Ethernet ports with different values of data transmitting speed. This model has degree of protection IP20. Additionally can be acquired software, fan module and SD-cards 2/8 GB for store programs and configurations [16].

During testing of the this hardware platform, we faced with next problems:

- small amount of storage available to work in this model of controller (only 2 GB of free space on original SD card);
- limited version of OS;
- limitations in current firmware.

First and second problem has been solved by installing a larger non-original SD card with a pre-installed Debian OS.

V. Description of machine vision module module

A. Marking detection problem statement

Detection of objects is possible in real time, but appears problem of fast frame processing by neural network. In-time completition of processing is not possible in case of each frame processing. In our task time

Figure 4. Main marking defects

window for processing one frame from video stream is only 0.013 s or 13 milliseconds. At present, neural network models, which capable of objects detection for a such time interval using a mobile device or a specialized controller, are not exist. Therefore, it is necessary to evaluate the importance of separate frames for performing detection.

On the other hand, bottles move on the manufacturing tape with certain frequency (about 3 pieces per second), which means that the neural network can process not every frame of the video stream, but only some and ignore the rest. This circumstance increases the time interval during which the processing should be performed, to a value of 250-300 milliseconds.

The process of marking recognition includes additional tasks such as test of marking printing. The first task is to determine the presence of marking on the cap. The second task is to determine the presence of marking distortions, which arise during the printing process, the absence of marking parts, etc. And, finally, the third task is the actual detection of the numbers in the marking, the formation of output information (the date of manufacturing of the goods, numbers in the consignment, etc.) and the determination of the correctness of this data in accordance with a predefined pattern. The main marking defects are shown in Fig. 4.

The presence of several subtasks involves the use of a group of neural networks, each of which performs its

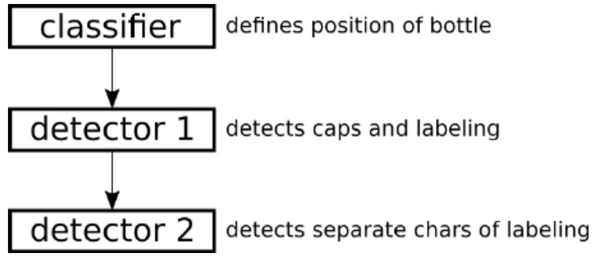

Figure 5. Structure of the marking recognition module

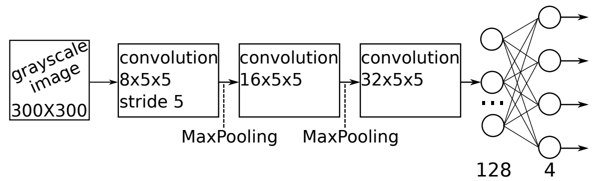

Figure 6. Structure of classifier for evaluating bottle position

own part of the work.

B. Architecture of the recognition module

Our recognition module is an extension of the idea described in [17]. The developed marking recognition module consists of three neural networks (Fig. 5).

The first network is a simple classifier based on a convolutional neural network, which determines the position of the bottle in the frame. We have selected four classes of position by distance from the center of the frame. Class 1 describes the minimal distance from the center of the frame. Only frames of this class are transferred for further analysis to other models. Class 2 and 3 describe the average and maximal distance. Finally, class 4 is needed for the case when the cap with the marking is not in the frame (for example, an empty line is viewed).

The architecture of the classifier is shown in Fig. 6. It consists of 5 layers and has 4 output neurons according to the number of classes that determine the position of the bottle in the frame. All layers use the ReLU activation function except for the 3rd and last layers. They use linear and softmax activation functions, respectively. Also, max pooling is applied after the first and second convolutional layers with stride = 2.

In case the frame was classified as class 1, it's transmitted further to the second neural network.

The second model is a detector and searches for caps and markings in the frame. Here, an SSD network based on the MobileNet v1 [18] classifier was chosen as the architecture.

At the stage of detection the cap and marking, an analysis is made of the presence of a defect associated with the absence of marking. This is made trivial: if object **cap** has been detected without object **marking**, then it has assumed that a defect occurs. If a defect

Figure 7. Distance of the bottle from the center of the frame (4 classes)

has been identified, the module notifies the operator. If no defects were found in the marking, the frame is transferred to the next neural network.

Finally, the third neural network is also SSD-MobileNet v1, which detects individual digits in the marking. After that, the formation of the final recognition result is performed.

The use of two serial networks for the detection of individual objects is explained by the fact that the original image has a high resolution and is scaled to 300X300 pixels before being fed to the detector. This conversion makes the detection of individual digits almost impossible due to their relatively small size. To eliminate this drawback, the image of the marking in the original dimensions with the necessary scaling is fed to the input of the second detector.

C. Training datasets: preparing and main features

To create a training dataset for the classifier, we have used the neural network model Faster R-CNN (based on the pre-trained ResNet50 classifier). This model has better detection efficiency compared to the SSD-MobileNet model, but it is slower [19]. This network was used to detect the caps in the frame. A trained detector was used to split the available dataset to bottle position classes. The Euclidean distance from the center of the cap to the center of the frame was used as a measure of distance. By this way, classes 1-4 were formed for classifier (Fig. 7). The resulting dataset includes 6189 images, 1238 of which make up the testing dataset.

After classifier was trained, the final classification accuracy was about 92%.

Both detectors (for detection of caps / markings and individual digits) were trained based on pre-trained models.

When training the detector of caps and markings, a prepared dataset was used, which includes 637 images in the training part and 157 in the testing part.

The following learning procedure was used to train the digit detector. At the beginning, the SSD model was pre-trained on images from SVHN dataset of house numbers for 80 epochs (several images from this dataset are shown in Fig. 8). For pre-training, a variant of this dataset was used, which includes 33402 images in the training part and 13068 in the test part. After pre-training, we have continued to train the neural network using a dataset of markings. This dataset contains 347 images in the training part and 72 in the test part. Examples of images were used for training are presented in Fig. 9.

Figure 8. Examples of images from SVHN dataset [20]

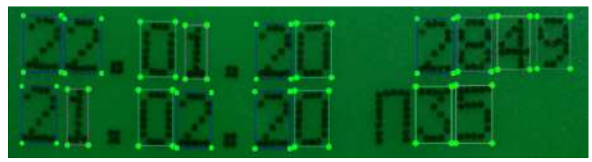

Figure 9. Image from training dataset of detector 2

D. Results

The use of the SSD model allows to achieve a detection efficiency of 99% (mAP = 0.99) for caps and markings detection and 87% (mAP = 0.87) for individual digits. Additionally, the processing speed allows to detect objects in the video stream at a speed of 76 frames per second. Efficiency of the detection of individual digits are presented in the table I.

Table I
DETECTION EFFICIENCY OF INDIVIDUAL CLASSES OF DIGITS

Class label	AP
0	0.8871
1.	0.8766
2.	0.8686
3.	0.8096
4.	0.8874
5.	0.8998
6.	0.8847
7.	0.8933
8.	0.8691
9.	0.8857
mAP	0.8762

The result of detection by the first and second detector is shown in Fig. 10 and 11.

VI. DECISION MAKING MODULE

A. Knowledge base

KB designed using OSTIS technology are divided into subject domains, which in turn have its own ontologies. The proposed system has next subject domains:

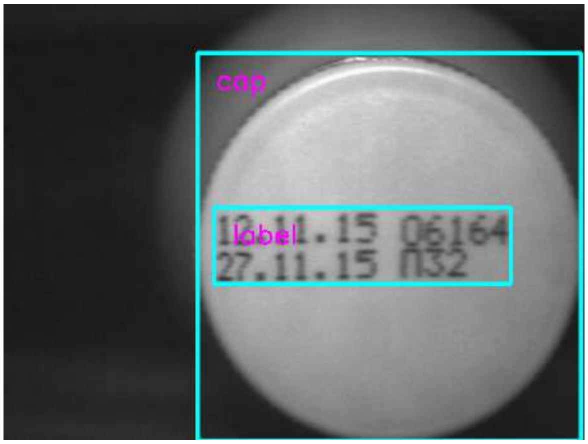

Figure 10. Example of cap/marking detection

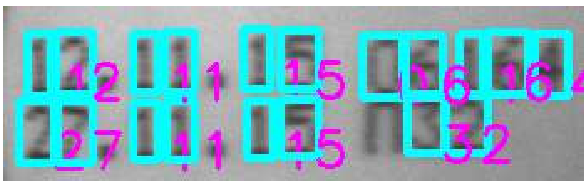

Figure 11. Example of digits detection

- Subject domain of product marking;
- Subject domain of neural network models;

Each subject domain that described in the KB has ontology in KB.

1) The subject domain of product marking: This subject domain contains a description of all necessary information about the product and quality control of its marking, such as:

- product brand;
- the recognized marking;
- the standard marking;
- number of consecutive unmarked products;
- the level of paint in the printer;
- and etc.

The figure 12 shows a fragment from this subject domain that describes the quality control point for marking the bottles of some yogurt. This fragment contains information that machine vision module didn't recognize the marking of bottle, which is currently being controled. Moreover, at this control point, this is the fourth bottle with an unrecognized marking. For this control point, there is a limit on the number of consecutive bottles with the wrong marking, which is equal to three. Also, the control point knows a printer, which prints the marking and the level of paint in it. In this case, this level is zero.

This information will be enough for a human to conclude on the reasons for the missing of marking. Below we will consider the mechanism by which the system will make a similar conclusion.

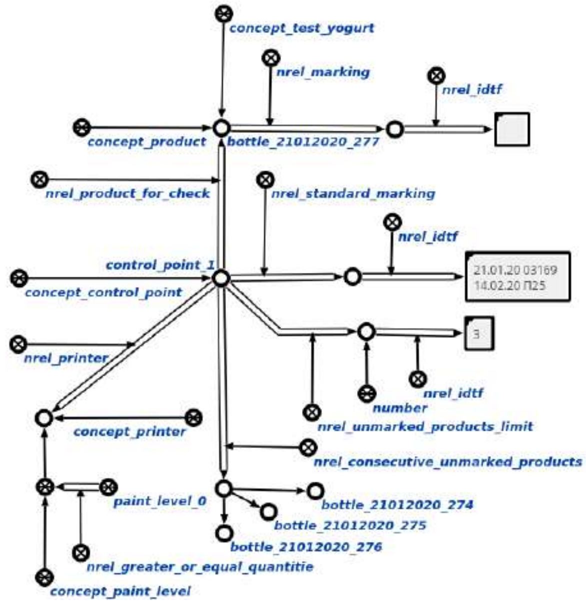

Figure 12. Fragment of subject domain of product marking for some yogurt bottles

This subject domain also contains a set of implicative and equivalence bundles, which we will call logical rules for short. According to these rules, the reverse inference [21] is used to make the decision or a set of decisions. Inference uses the ≪if≫ part as search patterns in the KB. When matching the ≪if≫ part of a statement was found, the system generates the knowledge described in the ≪then≫ part of the implicative bundle of the used logical rule. For logical rules, presented in the form of equivalence tuples, the mechanism of its using is similar, with the only difference that in place of the ≪if-then≫ parts there can be any part of the equivalence tuple.

It should be noted that the logical rule can also be the specifications of agents or programs. These specifications are represented in the form of the implicative tuple, in which "if" part describes the input data, and in "then" part describes the output data. When making the inference, the problem solver will use these logical rules on a par with the rest, but when using these logical rules, the appropriate agent or program will be called.

Each logical rule has the number of times it is used by inference. This technique will allow the system to self-learn and speed up the inference for trivial situations [6].

The figure 13 shows an example of a logical rule that can be written in natural language like this: *If the product is not marked, but it is critically bad marked, and the paint level in the printer is less than or equal to 0, then it is needed to stop the production tape and send a message that the printer has run out of paint.*

Applying of this logical rule to the KB which containing the fragment shown in the figure 12 will lead to the fact that the system will be able to conclude on the reasons for the missing of marking.

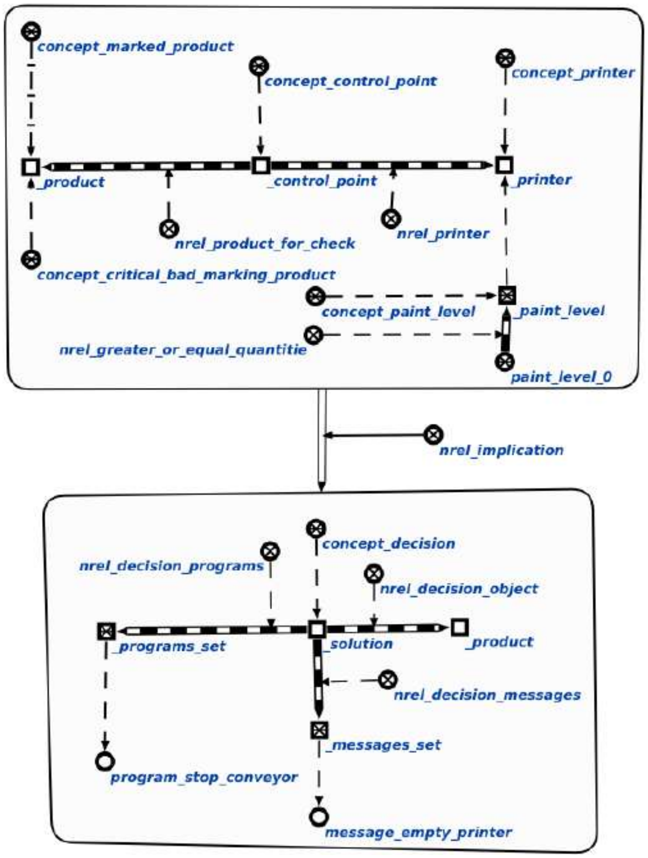

Figure 13. Logical rule for checking the reasons for missing a marking

One of the most important features of the system is the ability to explain made or proposed decisions. For this purpose, the inference makes a decision tree in the course. Decision tree stores the input recognition data, the recognized marking, the chain of applied logical rules and the applied (proposed for application by the engineer) decision.

2) Subject domain of neural network models: The description of this subject domain will help the specialist when setting up the system for a specific task. Based on this knowledge, the system will be able to offer one or another trained ANN available in the machine vision module. In this regard, there is a need to describe the entire hierarchy of subject domain of neural networks models, proposed in [5], in the KB. This subject domain contains describtion of the following information about the trained ANN:

- type of input data;
- set of recognized classes or output data type;
- class of tasks solved by ANN;
- architecture;
- average operating time;
- the quality of recognition.

In addition, this detailed description of the trained ANN in the KB can be used to provide information support to the engineer who will update the architecture or retrain the ANN.

In the future, this part of the KB can be used to expand the system's authority from supporting decision-making on marking control to supporting decision-making on selecting the configuration of machine vision module for a specific hardware and a specific hardware platform.

B. Problem solver

In OSTIS technology, problem solvers are constructed on the basis of the multi-agent approach. According to this approach, the problem solver is implemented as a set of agents called *sc-agents*. All sc-agents interact through common memory, passing data to each other as semantic network structures (sc-texts) [14].

The general tasks of the problem solver of IDSS are:

- access to knowledge in the KB;
- processing (analysis, verification, and generation)of knowledge; item interaction with other modules of the system.

The main task of the solver is implementation of reverse inference. In this way, the system knows in advance the set of reasons why incorrect markings may be printed.

At the start work, solver creates sets of logical rules(by pattern search), which applying will lead to the appearance of the necessary semantic construction in the KB. Next, it tries to apply the most frequently used logical rule. The logical rule can be applied when the KB contains semantic construction that isomorphic to the construction that was obtained by substituting nodes associated with the processed product into a template from a logical rule. This pattern is the first part of the logical rule, the second part describes the knowledge that will be generated after applying this logical rule.

If the rule can be applied, the system will generate knowledge from second part of rule and will add the rule and the result of its using to the decision tree.

In the case when there is not enough knowledge to apply a logical rule, the solver recursively initiates the work of itself, where it is already trying to find logical rules, the applying of which will lead to the appearance of the missing knowledge in the KB.

If the applying of any logical rule does not lead to the the appearance of the necessary semantic constructions in the KB, the agent reports that it can't find the decision for this problem.

In the future, we consider an option in which the system will be able to generate logical rules itself, based on patterns in the course of the system's operation or on cases of manual correction of problems.

VII. Conclusion

The implementation of the proposed IDSS significantly improves quality control in manufacturing, since such a system is able not only to identify a problem with products, but also to help find the causes of these problems, and even, in some cases, to solve them independently.

The quality of detection of individual digits can be improved with by increase of the training dataset, this is a subject of future work. The second aspect that affects the quality of detection is the location of the marking. It was noticed that when the marking is in a horizontal orientation, the quality of detection is higher than in case of rotated marking. Therefore a study of the possibility of using neural network models to select the correct marking orientation is required.

A perspective development of the system is the addition of software functionality to support the engineer when scaling the solution to any other product or on another hardware platform. The engineer should be able to set the configuration of the machine vision module, as well as receive recommendations on this configuration from the system. Recommendations will be based on the specification of the task and the hardware capabilities of the platform. This requires a description of the ontology of neural network models and hardware platforms.

References

[1] Schwab K. The fourth industrial revolution. Crown Publ., NY, 2016, 198 p.

[2] Tarasov V. B. Strategicheskii injiniring predpriyatii buduschego: massovoe sotrudnichestvo, internet veschei, iniciativa “industriya 4.0”, chto dalshe // Injiniring predpriyatii i upravlenie znaniyami (IP&UZ-2016): collection of scientific papers, XIX Scientific and practical conference, Moscow, April 26-27, 2016 / by scientific ed. Telnova. M.: PLEKHANOV Russian University of Economics, 2016 pp. 57–68., (in Russian).

[3] Schwab K., Davis N. Shaping the Fourth Industrial Revolution. World Economic Forum, Geneva, 2018, 287 p.

[4] Tariq A., Rafi K. Intelligent Decision Support Systems. Information and Knowledge Management, ISSN 2224-5758 (Paper) ISSN 2224-896X (Online) Vol 2, No.6, 2012

[5] Golovko V. A., Golenkov V. V., Ivashenko V. P., Taberko V. V., Ivaniuk D. S., Kroshchanka A. A., Kovalev M. V. Integration of artificial neural networks and knowledge bases. In: Open semantic technologies for designing intelligent systems (OSTIS2018): materials of the International Science and Technology Conf. Minsk, February 15-17, 2018) – Minsk: BSUIR, 2018. – P. 133–145

[6] Golovko V. A., Ivashenko V. P., Taberko V. V., Ivaniuk D. S., Kroshchanka A. A., Kovalev M. V. Principles of decision making building systems based on the integration of neural networks and semantic models. In: Open semantic technologies for designing intelligent systems (OSTIS2019): materials of the International Science and Technology Conf. Minsk, February 21-23, 2019 – Minsk: BSUIR, 2019 – P. 91–102

[7] (2020, Jan.) Omron documentation. [Online]. Available: https://robotics.omron.com/browse-documents

[8] Saraev A.D., Shcherbina O. A. System analysis and modern information technologies. Proceedings of the Crimean Academy of Sciences. - Simferopol: SONAT, 2006. - Pp. 47-59.

[9] Power D. J. Decision support systems: concepts and resources for managers, Westport, Conn., Quorum Books, 2002, 251 p.

[10] V. Golenkov and N. Gulyakina, "Proekt otkrytoi semanticheskoi tekhnologii komponentnogo proektirovaniya intellektual’nykh sistem. Chast’ 2: Unifitsirovannye modeli proektirovaniya [project of open semantic technology of component design of intelligent systems. part 2: Unified design models] Ontologiya proektirovaniya [Ontology of design], no. 4, pp. 34-53, 2014, (in Russian).

[11] D. Pospelov, "Situacionnoe upravlenie. Teoriya i praktika Moscow: Nauka, 228p, 1986, (in Russian).

[12] V. Ivashenko, "Ontologicheskaya model’ prostranstvenno vremennykh otnoshenii sobytii i yavlenii v protsessakh obrabotki znanii Vestnik BrSTU, Brest, No5(107), pp. 13–17, 2017, (in Russian).

[13] V Tarasov, "Ot mnogoagentnih sistem k intellektualnim organizaciyam filosofiya psihologiya informatika Moscow: Editorial URSS, 352 p., 2002, (in Russian).

[14] D. Shunkevich, "Agent-oriented models, method and tools of compatible problem solvers development for intelligent systems,"Otkrytye semanticheskie tekhnologii proektirovaniya intellektual’nykh system [Open semantic technologies for intelligent systems], vol. 2, pp. 119-132, 2018.

[15] PHOENIX CONTACT | Control technology and software – Intelligent automation [Online]. Available: https://www.phoenixcontact.com/assets/2018/interactive_ed/101_86135/index.html#4.

[16] PHOENIX CONTACT | Safety controller – RFC 4072S – 1051328 [Online]. Available: https://www.phoenixcontact.com/online/portal/us?uri=pxc-oc-itemdetail:pid=1051328&library=usen&tab=1.

[17] Golovko, V. Brands and Caps Labeling Recognition in Images Using Deep Learning / V. Golovko, A. Kroshchanka, E. Mikhno // International Conference on Pattern Recognition and Information Processing (PRIP 2019). – Springer, 2019. – P. 35–51.

[18] Howard A. G., Zhu M., Chen B., Kalenichenko D., Wang W., Weyand T., Andreetto M., Adam H., MobileNets: Efficient ConvolutionalNeural Networks for Mobile Vision Applications [Online]. Available: https://arxiv.org/pdf/1704.04861.pdf.

[19] Hui J., Object detection: speed and accuracy comparison (FasterR-CNN, R-FCN, SSD, FPN, RetinaNet and YOLOv3) [Online]. Available: https://medium.com/@jonathan_hui/object-detection-speed-and-accuracy-comparison-faster-r-cnn-r-fcn-ssd-and-yolo-5425656ae359.

[20] Netzer Y., Wang T., Coates A., Bissacco A., Wu B., Ng A.Y. Reading Digits in Natural Images with Unsupervised Feature Learning, NIPS Workshop on Deep Learning and Unsupervised Feature Learning, 2011.

[21] V. Vagin, A. Zagoryanskaya, M. Fomina, "Dostovernii i pravdopodobnii vivod v intellektualnih sistemah Moscow: FIZMATLIT, 2008, 704 p., (in Russian).

Реализация интеллектуальной системы поддержки принятия решений для сопровождения производственного процесса

Головко В. А., Крощенко А. А., Ковалев М. В.
Таберко В. В., Иванюк Д. С.

В работе рассматривается реализация интеллектуальной системы поддержки принятия решений в рамках решения реальных производственных задачах ОАО «Савушкин Продукт» по контролю качества нанесения маркировки на продукцию. Предложенная система интегрирует в себе нейросетевые и семантические модели. Предложен оригинальный метод локализация и распознавания изображений из видеопотока высокой скорости, а также способ интеллектуальной обработки результатов распознавания для выработки решений производственных проблем.

Received 30.01.2020