

# Deep Learning for Effective Visualization and Classification of Recyclable Material Labels

V.O. Kuzevanov<sup>1</sup>, D.V. Tikhomirova<sup>2</sup>

National Research Nuclear University “MEPhI”, Moscow, Russia

<sup>1</sup> ORCID: 0009-0005-7093-7709, [slava.kuzevanov@mail.ru](mailto:slava.kuzevanov@mail.ru)

<sup>2</sup> ORCID: 0000-0002-0812-2331, [dvsulim@mail.ru](mailto:dvsulim@mail.ru)

## **Abstract**

This paper presents an example of a system to improve the process of sorting recyclables by using deep learning techniques to automatically detect, classify and visualize recycling codes on product packages. In this paper, the authors discuss various approaches to optical character recognition and object detection in a video stream or image. The authors have developed and proposed a combination of neural networks for detection and classification of recycling codes. The proposed neural network system is designed to facilitate efficient recycling processes by automating the identification of recycling symbols, thereby facilitating the sorting and processing of recyclables.

**Keywords:** Deep learning, neural networks, recycling codes, automated sorting of recyclables, computer vision, image recognition.

## **1. Introduction**

As global attention to sustainable recycling practices increases, there is a growing need for advanced technologies to support recycling management processes for recyclables. Recycling codes, represented by symbols such as a recycling triangle with a numeric identifier (e.g. PETE 1, HDPE 2), play a crucial role in determining the recyclability of packaging materials. Such conventional symbols are used to indicate the material from which the item is made. Recycling codes may vary from country to country, but the states of the European Union have adopted Commission Decision 97/129/EC [1] to create a system for identifying packaging materials. Subsequently, in the countries of the Eurasian Economic Union, a Russian-language list of materials [2] with corresponding codes was approved, based on a document adopted by the EU. Therefore, the symbols indicating the codes of recycled materials in these countries are the same. This paper presents a solution to the problem based on deep learning methods to automate the recognition and classification of recycling codes on product packaging.

In recent years, the desire to solve recycling efficiency problems has stimulated extensive research and technological development in the field of recyclable materials handling. In this context, deep learning approaches have become a powerful tool for automating and optimizing various aspects of recyclables management and processing. However, a critical aspect that has yet to be comprehensively addressed is the accurate recognition and classification of recyclable material codes indicated on product packages. To solve this challenging problem, our approach requires the development of a system consisting of two different types of neural networks.

The difficulty in classifying recyclable material arises from the visual similarity of the recycling codes applied to product packages. For more accurate classification, a multi-stage approach becomes necessary. The system proposed by the authors consists of the stepwise use of two neural networks. The first neural network is designed to detect and determine the bounding box of recycled material codes in the video stream and in the image. Realizing that

the visual similarity of the recycle codes presents a major challenge for accurate classification, the next step we propose to use a second neural network specialized in optical character recognition (OCR). This network works on the detected frames by decoding the digits and signatures of the recycling codes.

By strategically dividing the task between two specialized neural networks, the authors of the paper aim to improve the accuracy and reliability of the whole system.

### **1.1. Review of research in the application of machine learning for object detection**

In the context of recycling management, the importance of object detection and optical character recognition becomes paramount. While many existing studies have laid the foundation for the application of machine learning in waste sorting, recent advances in object detection and optical character recognition have opened up new possibilities for the accuracy and efficiency of this method.

Object detection systems and optical character recognition (OCR) systems have been developed (and continue to improve) to address a wide range of applications in different domains, demonstrating their versatility and impact on various industries. In object detection, these systems excel in applications such as autonomous vehicle navigation [3] [4], where they identify pedestrians, vehicles, and road signs, contributing to improved road safety. In retail, object detection is used for inventory management and for customer analytics, automating inventory monitoring and optimizing the shopping experience [5]. At the same time, OCR systems play a key role in converting handwritten or printed text into digital formats [6], facilitating the digitization of documents and improving their accessibility for visually impaired people. Moreover, OCR is widely used in finance to automate data extraction from bills and receipts [7]. The adaptability of object detection and OCR technologies extends to the healthcare industry, helping healthcare professionals to analyze various images and process documents [8].

In addition to the above applications, object detection and optical character recognition systems play an important role in security and surveillance. In the security domain, object detection is used for real-time monitoring of public places [9], recognizing and tracking potentially suspicious activities or objects. On the other hand, OCR is crucial for the identification of vehicle license plates, extending the capabilities of Automatic License Plate Recognition (ALPR) systems [10], which find applications in law enforcement and parking management.

This wide range of applications emphasizes the potential of these systems, demonstrating their ability to solve a wide range of complex problems with sufficient accuracy and efficiency.

A significant amount of research has been devoted to exploring the integration of machine learning techniques into the recycling industry, reflecting the importance of this area to industry and society. Research has explored ways to apply machine learning algorithms to sort recyclables, analyze their composition, and identify materials in recycling streams.

Most papers on this topic tend to use machine learning to classify recyclables into broad categories such as plastic, paper, and glass, primarily based on visual features extracted from images on packaging materials. For example, the authors of the paper "Comparing deep learning and support vector machines for autonomous waste sorting" [11] compared the performance of convolutional neural networks (CNNs) and support vector machine (SVM) based method for autonomous waste sorting in three main categories: plastic, paper and glass.

While these efforts have contributed significantly to the automation of recyclable sorting processes, there is a notable gap in the literature with respect to detailed analysis of recycling codes on product packaging. Most existing works focus on broader material classification, overlooking the specific identification and interpretation of recycling symbols, a crucial aspect in the recycling ecosystem.

## 1.2. Computer vision and optical character recognition

With the development of technology and the emergence of the first electronic computers, it became necessary to digitize printed documents and texts into information "understood" by computers. For this purpose, work was carried out on the development of optical character recognition systems.

Optical Character Recognition (OCR) is a piece of software that converts printed text and images into a digitized form. In this form, this data can then be manipulated by a machine. Unlike the human brain, which can recognize text/symbols applied to images quite easily, machines are not "intelligent" enough to perceive all the information available in images. In the process of optical character recognition, a number of problems arise due to the fact that the text on the image may be written in different languages, and the processing code itself may be differently designed (the choice of text headset and lettering depends on the manufacturer's corporate identity). Consequently, methods from different disciplines of computer science (e.g., image processing, image classification, and natural language processing) are used to address different aspects of the problem.

It is because of its complexity that the task of optical character recognition belongs to the field of computer vision in machine learning. Machine learning allows computers to learn, make predictions, and make decisions based on a set of data, rather than producing a rigidly programmed and unambiguous result.

Computer vision [12] is a field of machine learning that retrieves meaningful information from image files, i.e. with images and video streams, processes it and provides results. Computer vision is used in tasks such as:

1. Identification
2. Object detection
3. Object segmentation
4. Assessment of the position
5. Text recognition
6. Object generation
7. Video analysis

Let us consider the main steps that are necessary to develop an optical character recognition system [13].

*Table 1– Main stages of OCR system development*

Stage	Description	Approaches
Data collection	The process of acquiring image information	Digitization, binarization, compression
Data preprocessing	Enhancing image quality	Noise removal, skew removal, thinning, morphological operations
Segmentation	Dividing an image into its component parts	Segmentation techniques
Classification	Categorizing a character or object to its true class	Machine learning methods (Bayesian classifier, decision trees, neural networks, etc. [6])
Postprocessing	Improving the quality of OCR system results	Contextual approaches, use of multiple classifiers

In the book "Optical character recognition: An Illustrated Guide to the Frontier" [14], the authors describe the main problems faced by optical character recognition systems. Even if the text is printed with an easy-to-read headset and in a standard format, many errors are caused by image defects. Such defects can occur at different stages, such as during the printing of an image or during its digitization. Typically, defects consist of heavy, smudged, contiguous marks (Fig. 1.a), light and broken marks (Fig. 1.b), or combinations thereof. Wander-

ing marks (Fig. 1.c) and curved baselines (Fig. 1.d) also distort the system results. In addition, the similarity of basic processing codes and defects from surveying tools also complicate the classification task.

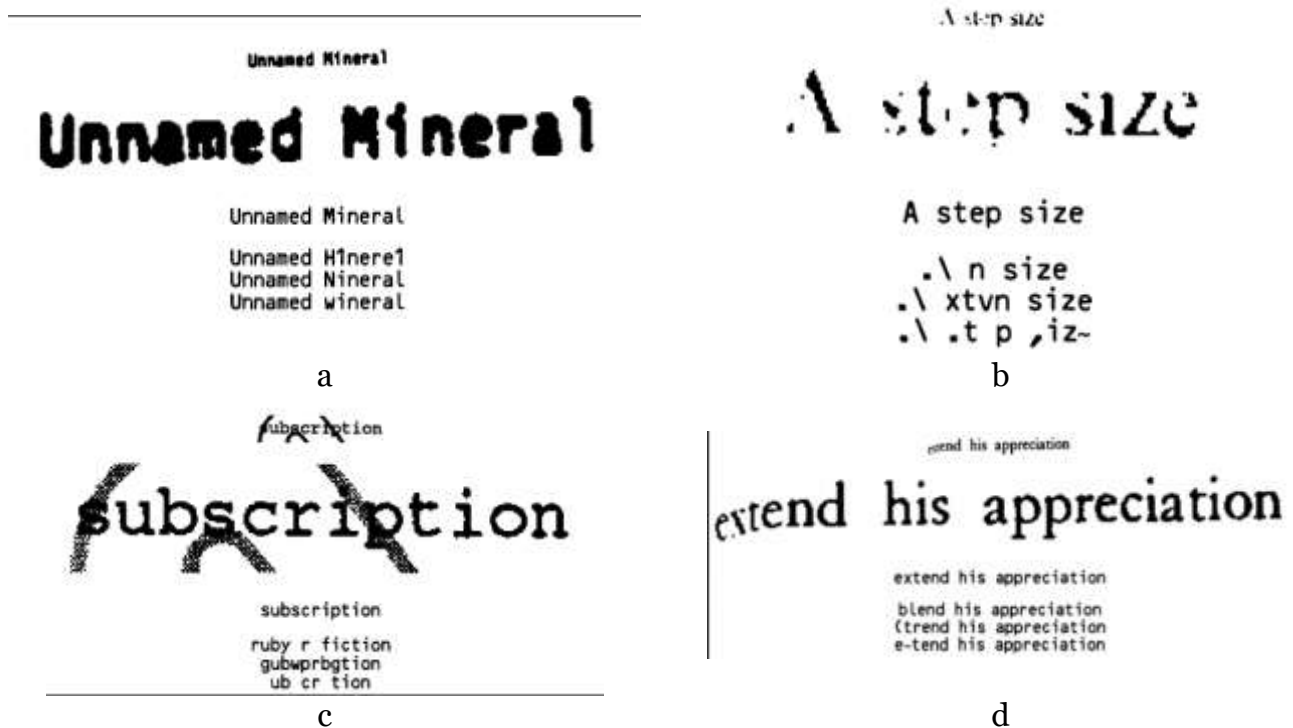


Fig 1. Examples of image defects: a) blurred marks b) light marks c) wandering marks d) curved baselines

One of the goals of this paper is to build a system that recognizes recycling symbols on packages with sufficient accuracy even in the presence of the defects noted by the authors.

### 1.3. Main ways to classify an object in an image

Currently, there are three main ways to recognize and classify objects in an image and video stream:

1. Template matching
2. Image segmentation and BLOB object analysis
3. Neural networks

Pattern matching is one of the easiest to understand ways to recognize an object in an image (a video stream is a sequence of images). This method is based on finding the place in the image that is most similar to the template. The similarity of the image to the template is given by a certain metric: the template is "superimposed" on the image, and the divergence between the image and the template is considered. The position of the template, at which this discrepancy will be minimal, and will mean the place of the desired object.

As a metric we can use different variants, for example, the sum of squared differences (SSD) between the template and the image (formula 1), or use cross-correlation (CCORR) (formula 2). Let  $f$  and  $g$  be an image and a template of sizes  $(k, l)$  and  $(m, n)$  respectively (we will ignore color channels for now);  $i, j$  are positions in the image to which the template is applied.

$$SSD_{i,j} = \sum_{a=0..m, b=0..n} (f_{i+a, j+b} - g_{a,b})^2 \quad (1)$$

$$CCORR_{i,j} = \sum_{a=0..m, b=0..n} (f_{i+a, j+b} * g_{a,b})^2 \quad (2)$$

It can be seen that these metrics require a pixel-by-pixel match of the template in the image being searched. Any deviation in gamma, light, or size will cause the methods to fail. This is exactly what happens with recycling codes, as the images of recycling codes on packages are not standardized (as we mentioned, they can be different sizes, colors, and shapes depending on the manufacturer's corporate identity). For example, one of the PET 1 recycling codes may look different on different packages (see Figures 2.a and 2.b).



a



b

*Fig. 2. a – Sample of PET 1 code marking on packaging; b – Sample of PET 1 code marking on packaging*

Image segmentation usually uses object properties such as size, color or shape. Therefore, it seems possible to correctly classify data by knowing the main characteristics of the object, provided that they are similar in the image. However, the image of processing codes can be strikingly different from the established pattern, which makes it necessary to look for other ways to solve the problem.

A BLOB object (Binary Large Object) is a sequence of binary data that can represent any form of unstructured information such as images, audio, video, or other files. In the context of image processing, BLOB objects represent connected regions in a binarized image that correspond to different objects or parts of interest in the image. BLOB objects help to highlight and identify individual cohesive regions in an image that represent potential objects of interest. Each BLOB object is characterized by a set of features such as area, shape, perimeter, moments of inertia, and other geometric properties. These features can be used to describe and distinguish between different objects.

However, the image of the recycling codes may be strikingly different from the established template, which may suggest that they will not have unambiguously similar features.

Therefore, for classification and recognition of recycling codes it is reasonable to use the third approach - neural networks, especially since the use of neural networks in the field of garbage sorting from photos has already been described in the book "Waste Segregation System Using Artificial Neural Networks" [15]. Moreover, the problem of material sorting using artificial intelligence has also been investigated by R.S.Sandhya Devi, Vijaykumar VR,

M.Muthumeena in their research paper [16]. This once again confirms the effectiveness of such systems in computer vision tasks.

There are two approaches to object recognition using deep learning. The first is to train the model from scratch. To train a deep network from scratch, you need to collect a very large labeled dataset and develop a network architecture that will learn object features and build a model. The results can be impressive, but this approach requires a large amount of training data. In addition, levels and weights in the CNN need to be customized. The second method is to use a pre-trained deep learning model. Most deep learning applications use a transfer learning approach, a process that involves fine-tuning a pre-trained model. It starts by taking an existing network, such as AlexNet [17] or GoogLeNet [18], and introducing new data containing previously unknown classes. This method requires less time and can provide faster results since the model has already been trained on thousands or millions of images.

Deep learning offers a high level of accuracy but requires a large amount of data for accurate predictions.

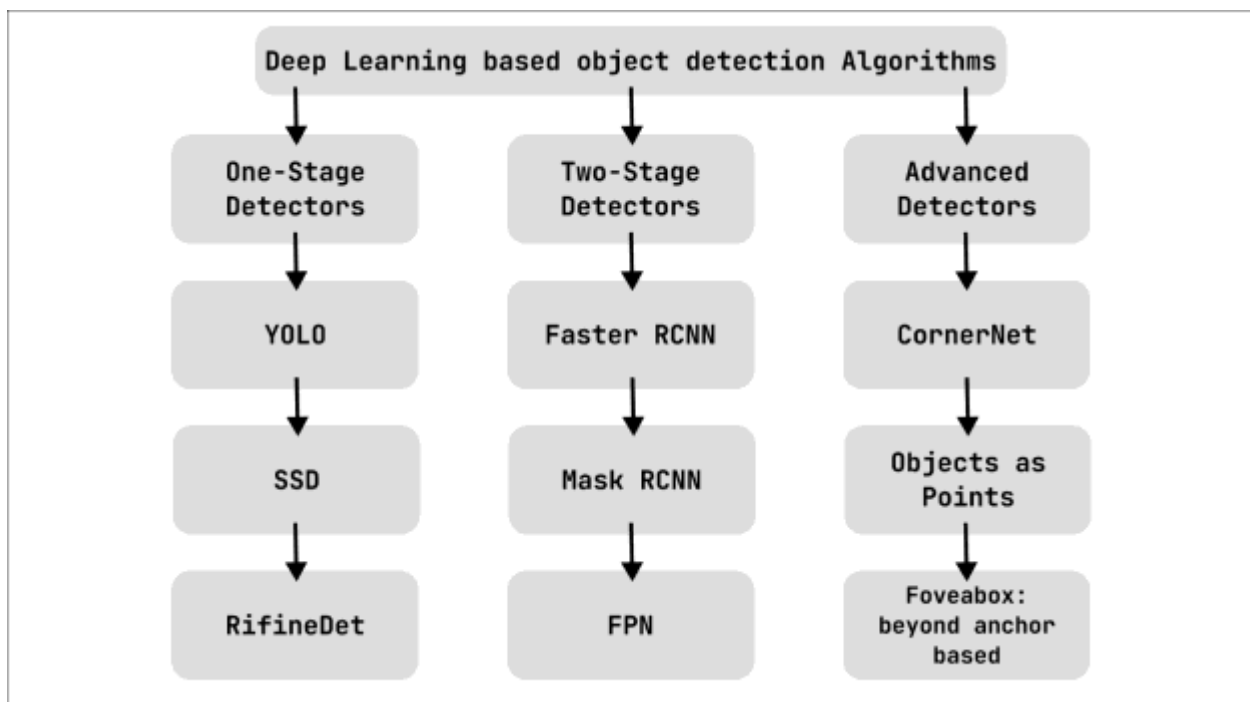
## 2. Object detection and OCR

### 2.1. Neural networks – detectors

The application area of computer vision is very extensive and allows solving a number of problems related to image and video processing. Many methods and neural network architectures have been developed to recognize objects in real time. The authors of the article «Deep-learning based object detection in low altitude UAV dataset: A Survey» [19] propose to divide neural network architectures into three types (see Fig. 3): Two Stage Detectors, One Stage Detectors and Points-based Detectors / Advanced Detectors.

Frame predictions in two-stage detectors are realized in two stages. During the first stage, "regions of interest" (box proposals) are generated and brought to the same size. The second stage involves predicting the coordinates of frames and their class membership.

Single-stage neural networks are characterized by a monolithic neural network architecture without a separate generation algorithm (Selective Search or Region-proposal network). The most popular and frequently used single-stage neural network architectures for detection at the moment are SSD (Single-Shot-Detector) [20] and YOLO (You-Only-Look-Once) [21].



*Fig. 3 – Taxonomy of object detection methods based on neural networks*

Sahel, Alsahafi, Alghamadi and Alsubait [22] worked on a logo detection model and compared RCNN, FRCNN, Retina Detection models to determine which one is the most accurate. The dataset used in the paper is taken from FlickrLogos32, which has 32 logos and also has explicit annotations from the makeense.ai website. The authors stated that they achieved an accuracy of 99.8% for the RCNN model and 95.8% for the Retina Net. The research paper further concludes that CNN models are the most preferred when it comes to accuracy.

Brahim Jabir, Nouredine Falih, Khalid Rahmani [23] worked on comparing different object detection models namely Detectron2, EfficientDet, YOLO and Faster R-CNN. The aim of their study is to create a model that detects weeds in crops in real time using computer vision. Their dataset was created by manually capturing images of weeds from fields using a professional camera under different lighting conditions. In the final section, they stated that YOLO v5 is the fastest model for real-time detection of weeds in crops compared to Detectron2, EfficientDet, and Faster R-CNN. They also stated that Faster R-CNN is better than YOLOv5 in terms of accuracy.

Varad, Rohit, Rushal and Sohan [24] compared two models YOLO and SSD for real-time object detection. The authors collected their dataset from coco.names which has 80 object classes on which the models will be trained to perform detection. The researchers trained the models to detect objects in real-time using an observer drone. The authors of the paper concluded their study by stating that the SSD model is faster but less accurate, whereas YOLO has both speed and accuracy and is considered an efficient model.

Based on the comparative analyses of neural networks for object detection given in the studies above, the authors of the paper conclude that the YOLO architecture is currently the most suitable for solving the problem of real-time detection of recycling codes on packages. In their system, the authors of the paper will utilize the neural network based on YOLO architecture to detect the bounding box of recycling codes.

## **2.2. OCR algorithms**

The best-known models for building OCR systems are Tesseract Model [25] and EAST (An Efficient and Accurate Scene Text Detector) [26]. EAST neural network, known for its efficiency and accuracy, is designed specifically for text detection in natural scene. It uses a unique architecture that divides the text region into quadrilaterals, which allows for more accurate localization and recognition of text in natural scenes. This makes EAST a particularly robust mechanism when processing irregularly shaped text and different orientations. This makes the EAST neural network well suited for tasks such as license plate recognition and scene text extraction (the area that falls within the camera's field of view). On the other hand, Tesseract OCR, an open-source optical character recognition engine developed by Google, is known for its versatility and broad language support. Originally developed at Hewlett-Packard in the 1980s, Tesseract has undergone significant improvements and is now based on deep learning techniques. Tesseract excels at recognizing text in images and documents, providing accurate results even with complex layouts and fonts. Its flexibility and adaptability make it a popular choice for a variety of OCR applications, from digitizing printed documents to extracting text from images for various purposes such as content analysis and data mining.

Tesseract OCR uses deep learning architecture for optical character recognition. Recent versions of Tesseract, including Tesseract 4, have integrated neural networks into the recognition process. Convolutional neural networks (CNNs) are used to analyze image regions and extract features important for character recognition. The Tesseract neural network model works with text string segments extracted from the input image. It incorporates Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, to capture contextual dependencies and improve recognition accuracy, especially in scenarios with complex layouts or different font styles. The architecture of the Tesseract OCR neural network [27] is designed to be flexible and can be fine-tuned for specific recognition tasks, which facilitates its adaptation to a wide range of OCR applications.



The EAST neural network, designed for scene text detection, presents an architecture optimized for efficiency and accuracy. EAST utilizes a U-shaped architecture, often referred to as U-Net architecture, which is a type of convolutional neural network (CNN). This architecture allows EAST to process images at different resolutions, capturing both fine details and coarse contextual information. A unique feature of EAST is an output layer that generates quadrilateral predictions for text regions, including bounding rectangle coordinates and an angle parameter. This choice of architecture allows EAST to efficiently handle text with arbitrary orientation, making it robust in scenarios where text may appear in different angles, such as on signs or in natural scenes.

Thus, both Tesseract OCR and EAST utilize neural network architecture to perform their specific tasks. Tesseract focuses on character recognition using CNNs and LSTMs, while EAST specializes in scene text detection using its U-Net inspired architecture, which provides accurate and efficient text extraction from complex images.

### 3. Proposed approach to improve the sorting process of recyclable materials

The authors propose a new approach to improve the sorting process of recyclable materials based on the use of a combination of two specialized machine learning models. The first neural network, a detector based on the YOLOv7 architecture, is needed to identify the bounding box of recyclable material codes and to pre-classify the symbol (assign it to a certain type of recyclable material). The second neural network EAST is needed to determine the digit and letter inside the bounding box, obtained as a result of the output of the first network. The use of the second network is suggested only if the confidence of the YOLOv7 model in the predicted class is below 0.5. Next, the result of the OCR model on text recognition is compared with text templates that may be on the processing code notation. For example, the recycling code of low density polyethylene can be either 4 LDPE or 4 PE-LD. The architecture of the system proposed by the authors is shown in Figure 4.

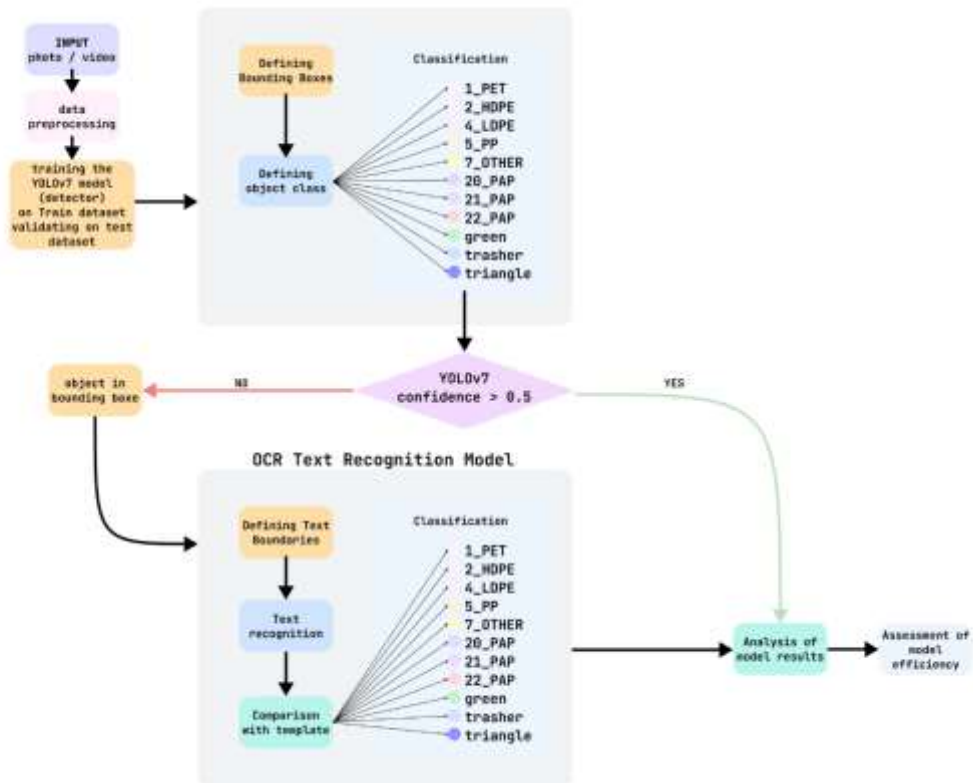


Fig. 4 – Proposed system architecture for recognizing and classifying recyclable material codes



### 3.1. Data collection and data pre-processing

The set of images for training neural networks includes photographs of food and household goods packages freely available on the Internet, as well as images taken by the authors themselves using a smartphone camera. The system is trained on images with 11 symbols that are most often encountered in everyday life.

In the end, 531 images were collected labeling recycling codes on products.

Based on the collected photographic material, the authors of this article determined that very often the processing codes are either "extruded" on the surface of the product and do not have a clear color difference from the background, or are on transparent surfaces. Therefore, it was decided to pre-process the images in such a way to clearly highlight the boundaries of the processing markings.

Several convolution layer kernels for image processing have been considered, such as:

1. Contour kernel (outline)  $\begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$
2. Contour kernel second version (custom)  $\begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$
3. Emboss kernel  $\begin{pmatrix} -2 & -1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix}$
4. Joint kernel of contouring and embossing  $\begin{pmatrix} -3 & -2 & -1 \\ -2 & 9 & 0 \\ -1 & 0 & 1 \end{pmatrix}$

However, after obtaining the results of testing these convolution layer kernels, the authors of the paper came to the conclusion that they do not achieve the required result, and the processing symbols applied to transparent surfaces or made by embossing are not clearly distinguished on the scanned images. For this reason, it was decided to use the operator or Sobel filter. An example of an image with complex variants of recycled material codes application is presented in Figure 5, and the results of Sobel filter application to this image are presented in Figure 6.



Fig. 5 – Original image with examples of hard-to-recognize recyclable material codes



*Fig. 6 – Result of applying the Sobel operator to the image*

The Sobel operator is used to extract boundaries and contours in images. It applies two convolution kernels (masks) to the image to find horizontal and vertical intensity gradients. These convolution kernels, often called filters, compute the first derivative of the image brightness.

The Sobel operator uses the following kernels to compute intensity gradients:

Horizontal kernel (highlights horizontal boundaries):

$$G_x = \begin{pmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{pmatrix}$$

Vertical kernel (highlights vertical boundaries):

$$G_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{pmatrix}$$

The image is processed by each of the convolution kernels (horizontal and vertical) separately. This creates two gradient images  $G_x$  and  $G_y$ . For each pixel, the resulting gradient value is calculated, which combines the horizontal and vertical gradients. This is done using the following formula:

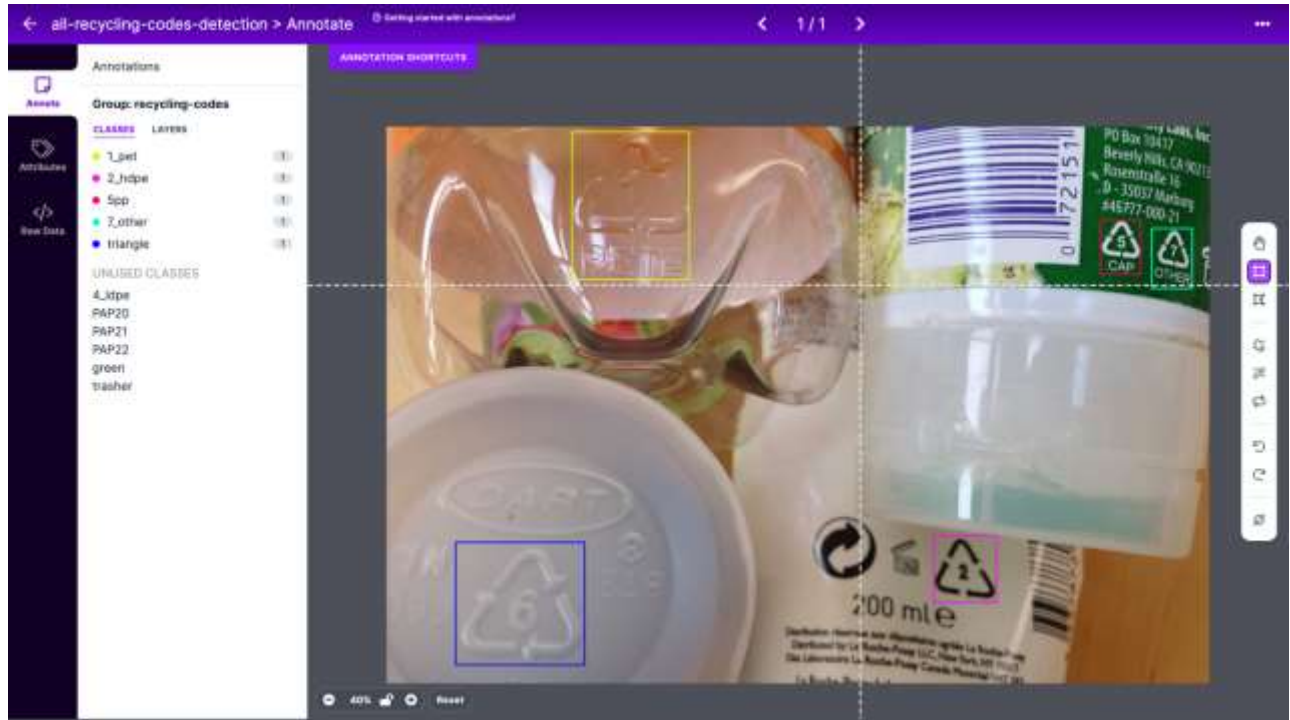
$$G = \sqrt{G_x^2 + G_y^2}$$

In the final image processed by the Sobel operator, contours and boundaries of objects become visible. These boundaries display sharp changes in brightness, which allows to highlight and visualize key structural elements of the image.

As you can see, the Sobel operator significantly improves the selection of the processing code on a transparent surface, making it more visible and clear. On other surfaces, the Sobel operator also helps to emphasize the stamped processing code, improving contrast and making the code more distinguishable from the background.

Next, the authors of the paper wrote an image augmentation class. It assumes that the image is adjusted to a common format (640x640 pixels), random rotation, horizontal reflection and color change occurs.

After creating augmented data, it is necessary to mark up the data. The roboflow service [28] provides convenient tools for this purpose. It is used to create a file necessary for training the YOLOv7 network with the description of classes and coordinates of bounding boxes of objects in the image. The process of data partitioning is presented below in Figure 7.



*Fig. 7 – Image labeling process in the training dataset*

### **3.2. Implementation of YOLOv7 neural network and its training**

The model used by the authors for implementation is a pre-trained YOLOv7 model on MS COCO data [29]. The MS COCO data, in turn, represents images for 80 classes of objects ranging from cars to toothbrushes. Using a pre-trained model reduces the time to train the model on our data, since the convolutional layers that create the feature map are already trained at a reasonably good level. The model trained on MS COCO data is in the public domain [30]. The Python programming language and its standard libraries for data analysis and machine learning: pandas, scikit-learn, Tensorflow and OpenCV are used for training, pre-processing and obtaining predictions.

YOLOv7 consists of 77 layers, which are a sequential combination of convolutional layers looking for feature maps, as well as concatenation and MaxPool layers. The model was trained on 400 epochs. It is worth noting that the training process of the YOLOv7 model takes quite a long time of several hours.

To evaluate the quality of the model, 2 metrics were used: Mean Average Precision (mAP) [31] and Intersection over Union (IoU) [32].

The error value is determined by how far the predicted bounding box is from the true bounding box. If the predicted bounding box is far away, the loss estimate is high. The training graph of the model is shown in Figure 8.

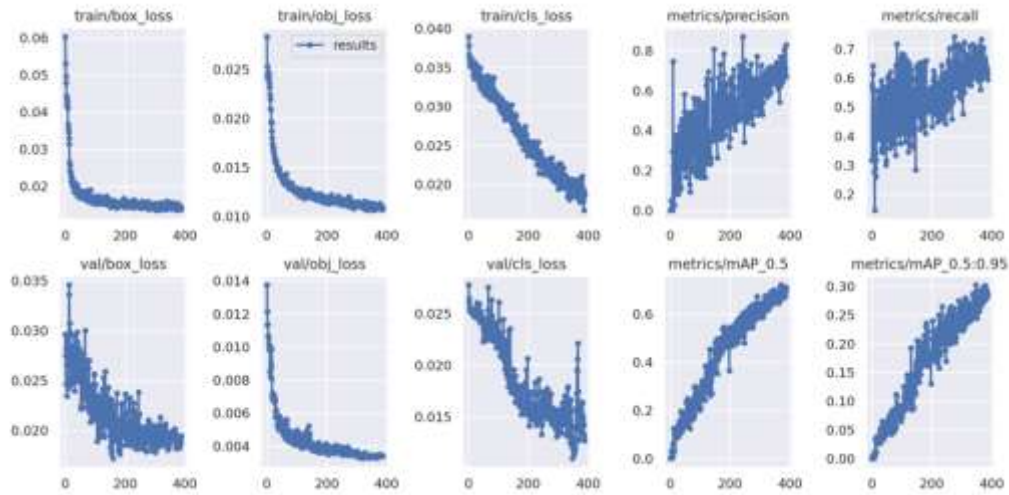


Fig. 8 – YOLOv7 training graphs on training data

After training the first neural network from the system, a mAP quality metric of 79% on training data and 70% on test data was achieved. The charts with average precision values by class (average precision) are shown in Figures 9 and 10.

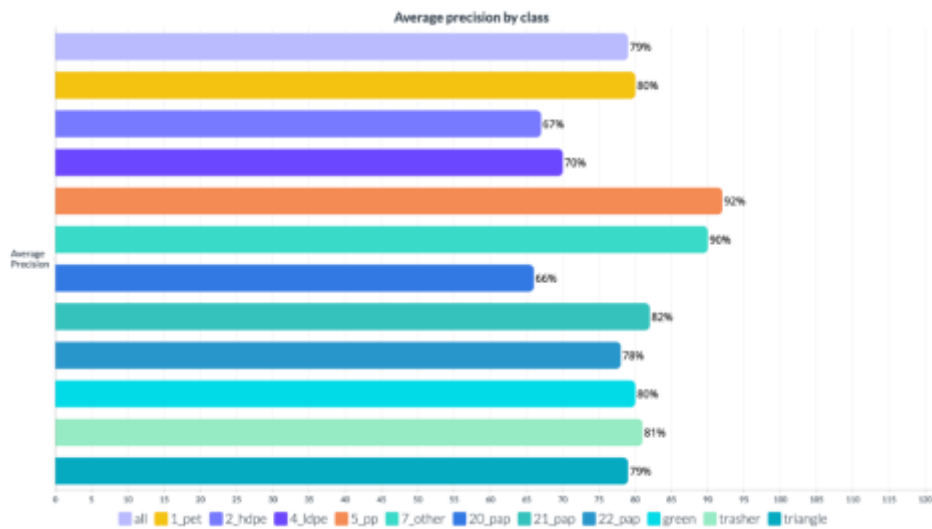


Fig. 9 – Average Precision by class on training data

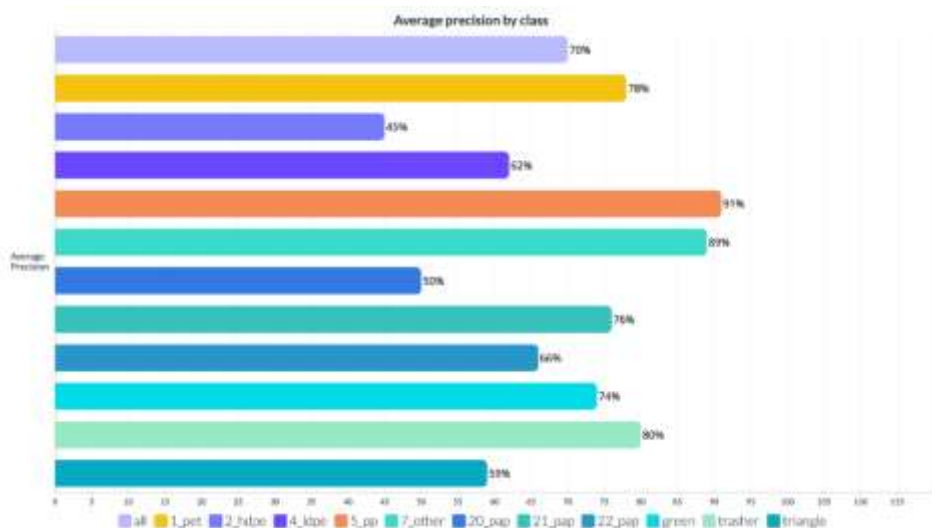


Fig. 10 – Average Precision by class on test data

As a result of demonstrating the performance of the model, we can say that it has learned to generalize knowledge about all classes well, and identifies areas of interest (necessary recycling codes on product packages) quite accurately and with high confidence. However, it can be observed that it is wrong in the classes of its prediction. This may be due to the fact that the recycling class logos have many different representations. In addition, they are all very similar to each other and differ at best by a number inside a triangle or an inscription. Testing of the model's predictions on the image is presented in Figures 11.a – 11.c.

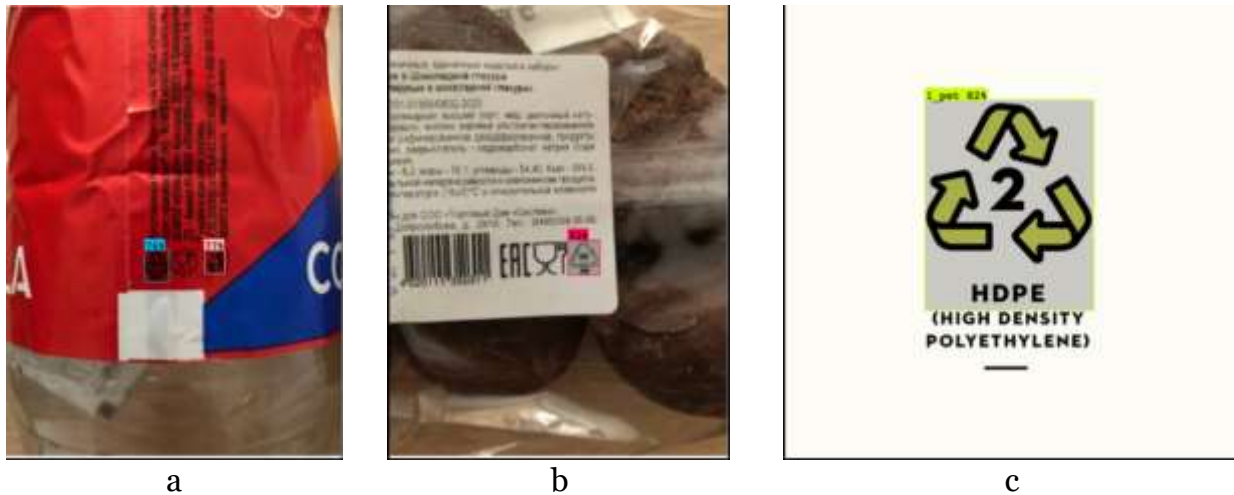


Fig 11. An example of the model's performance on test images: a) test 1 b) test 2 c) test 3

Figures 11.a and 11.b show a visualization of the bounding boxes for recyclable material codes on product labels. The numbers above the bounding boxes are a visualization of the probability that the selected object belongs to the class predicted by the model. Figure 11.c clearly shows an example where the model has qualitatively identified the bounding box of a recyclable material code, but is wrong about the predicted class.

To reduce the share of such false positive classification results, the authors of the paper suggest using OCR models such as EAST in addition to the YOLOv7 model.

### 3.3. Application of EAST to improve system predictions

The Efficient and Accurate Scene Text Detector model was chosen as the implemented model for recognizing text in an image. The OpenCV library of the Python programming language is used for this purpose.

To implement the EAST model for text discovery in Python, the following key steps need to be followed:

Step 1: Install the required dependencies:

Before starting the implementation, make sure that the necessary dependencies are installed. The main libraries include OpenCV, NumPy and argparse. OpenCV is used for image processing, providing various tools and algorithms such as the Sobel operator. NumPy is needed to work with data arrays and perform numerical operations. Argparse is used for processing command line arguments, which allows convenient passing of parameters and settings into a script.

Step 2: Loading a pre-trained EAST model:

The EAST model requires the use of pre-trained weights and a configuration file. These can be obtained from the official EAST repository.

Step 3: Pre-processing the image:

Before passing an image through the EAST model, some preprocessing steps must be performed. This step is necessary to improve the readability of the text areas, as various inaccuracies make it very difficult for OCR models to work. Commonly used are converting the image to grayscale and applying blurring and noise reduction techniques. In addition, it is nec-



essary to bring the width and height of the image to numbers divisible by 32. This is a prerequisite for the EAST model.

After obtaining images from the output of the first model (detector), it remains only to bring the width and height of the image of the obtained area with the processing code to numbers that are multiples of 32, since the Sobel operator has already been previously applied to the image.

Step 4: Perform text recognition:

To perform text detection using the EAST model, we need to transmit the preprocessed image through the network and extract the coordinates of the bounding box of the detected text regions.

Figure 12 illustrates the results of the EAST model for recognizing text in an image from the previous prediction of the YOLOv7 model.

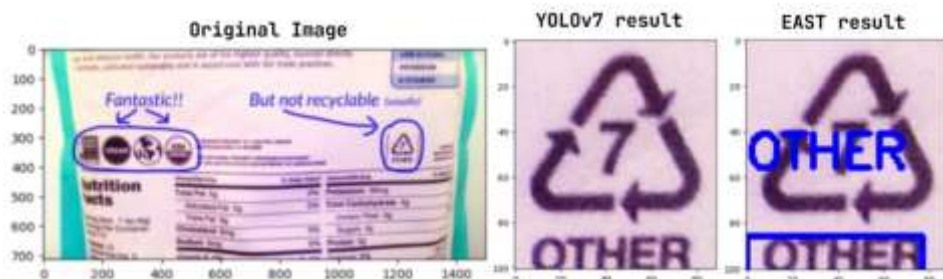


Fig. 12 – Text recognition by EAST model based on resulting bounding boxes from YOLOv7 model

After applying the OCR of the EAST neural network in addition to the results of the YOLOv7 model, the results of the whole system were significantly improved.

The predictions of the system became more accurate, and the mAP quality metric reached a level of 93%. To verify the system performance, predictions are performed on new images and an error matrix is constructed (Figure 13).

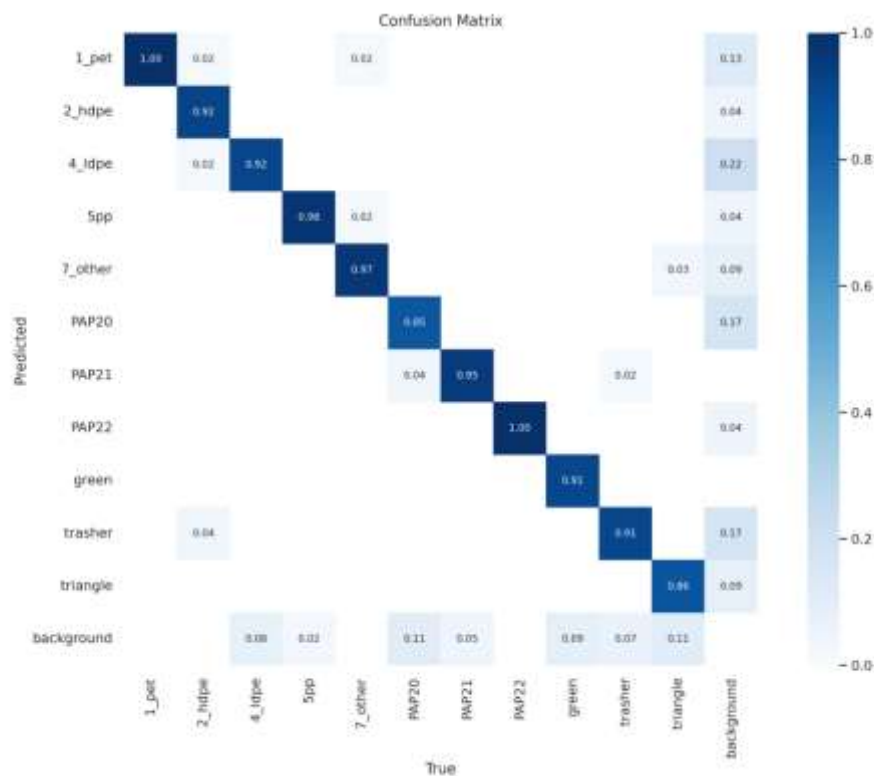


Fig. 13 – Confusion Matrix

## 4. Conclusion

This work reviewed the main approaches to object detection in video streams and real-time images, as well as the main technologies in the field of optical character recognition. The research on the applicability of such object detection and optical character recognition systems to various fields has been studied. The authors proposed a system for detection and recognition of recyclable material codes on packages for more accurate sorting of recyclables.

The proposed system, based on the latest technologies in the field of real-time object detection and optical character recognition, in particular the YOLOv7 and EAST models, has demonstrated high performance in the accurate detection and classification of recycling codes on product packages. The performance of the model indicates its potential for real-world applications in automated recycling systems.

The practical application of a system for recognizing the codes of recyclable materials represents an important step in sustainable waste management. In the context of visual image processing, the system demonstrates the ability to efficiently highlight and interpret symbols on packaging, which is of key importance for building mobile applications and automated receiving points. A mobile application based on this technology can become a reliable ally in the daily practice of waste collection and sorting. Users will be able to easily scan special codes on packages, receiving instant information on how to properly categorize and discard waste. This creates an interactive and educational interaction, promoting public awareness and engagement in sustainable consumption and resource management.

Moreover, automated collection points with recyclable materials sorting systems enhance the efficiency of the recycling process. Citizens can easily turn in separately collected materials knowing that their efforts are aimed at maintaining a green lifestyle and environmental sustainability. Such innovative solutions not only simplify people's lives, but also actively contribute to an environmentally responsible society where resource recycling becomes an integral part of the daily routine.

This system will allow not to change the processes of packaging production on the part of manufacturers, to unify the symbols of codes of recyclable materials. The system proposed by the authors provides high efficiency of classification of recyclable materials, even in the presence of differences in the image of the symbols of recyclable materials. Visual accuracy and reliability of the system emphasize its importance for practical application in the field of recycling and waste management.

## Acknowledgements

The work was carried out under the financial support per the government order for the project "Analytical and numerical methods of research of complex systems and non-linear problems of mathematical physics" (mnemocode FSWU-2023-0031).

## References:

1. 97/129/EC: Commission Decision of 28 January 1997 establishing the identification system for packaging materials pursuant to European Parliament and Council Directive 94/62/EC on packaging and packaging waste. EUR-Lex. 28.01.1997. URL: <https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX%3A31997D0129> (accessed: 02.10.2023).
2. Цифровой код и буквенное обозначение (аббревиатура) материала, из которого изготавливается упаковка (укупорочные средства) [Numeric, Alphabetic (Abbreviation) Designation of the Material from Which the Packaging (Closures) is Made]. Технический регламент Таможенного союза О безопасности упаковки ТР ТС 005/2011 [Technical regulations of the Customs Union "On Safety of Packaging"]. 18.10.2016. ЕЭК [Eurasian Economic Commission]. URL: <https://eec.eaeunion.org/comission/departament/deptexreg/tr/bezopypakovki.php> (accessed 01.10.2023).



3. Feng D., Harakeh A., Waslander S., Dietmayer K. A Review and Comparative Study on Probabilistic Object Detection in Autonomous Driving // IEEE Transactions on Intelligent Transportation Systems (<https://arxiv.org/abs/2011.10671>)
4. Li Y., Hu H., Liu Z., Zhao D. Influence of Camera-LiDAR Configuration on 3D Object Detection for Autonomous Driving (<https://arxiv.org/abs/2310.05245>)
5. Fuchs K., Grundmann T., Fleisch E. Towards Identification of Packaged Products via Computer Vision: Convolutional Neural Networks for Object Detection and Image Classification in Retail Environments // 9th International Conference on the Internet of Things. 2019. Vol. 9. P. 1-3. ([https://www.researchgate.net/publication/337068624\\_Towards\\_Identification\\_of\\_Packaged\\_Products\\_via\\_Computer\\_Vision\\_Convolutional\\_Neural\\_Networks\\_for\\_Object\\_Detection\\_and\\_Image\\_Classification\\_in\\_Retail\\_Environments](https://www.researchgate.net/publication/337068624_Towards_Identification_of_Packaged_Products_via_Computer_Vision_Convolutional_Neural_Networks_for_Object_Detection_and_Image_Classification_in_Retail_Environments))
6. Christy M., Gupta A., Grumbach E., Mandell L., Furuta R., and Gutierrez-Osuna R. Mass Digitization of Early Modern Texts with Optical Character Recognition // ACMJ. Comput. Cult. Herit. 11, 1, Article 6 (December 2017). P. 1-17. (<https://psi.engr.tamu.edu/wp-content/uploads/2018/01/christy2017jccch.pdf>)
7. Yimyam W., Ketcham M., Jensuttiwetchakult T., Hiranchan S. Enhancing and Evaluating an Impact of OCR and Ontology on Financial Document Checking Process // 2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP). 2020
8. Li X., Hu G., Teng X., Xie G. Building structured personal health records from photographs of printed medical records // American Medical Informatics Association Annual Symposium (AMIA). 2016. P. 1-5. ([https://www.researchgate.net/publication/308724882\\_Building\\_structured\\_personal\\_health\\_records\\_from\\_photographs\\_of\\_printed\\_medical\\_records](https://www.researchgate.net/publication/308724882_Building_structured_personal_health_records_from_photographs_of_printed_medical_records))
9. Ko T. A survey on behavior analysis in video surveillance for homeland security applications // Video Surveillance. 2011. P. 279-294.
10. Saluja R., Maheshwari A., Ramakrishnan G., Chaudhuri P., Carman M. OCR On-the-Go: Robust End-to-end Systems for Reading License Plates & Street Signs // International Conference on Document Analysis and Recognition (ICDAR). 2019. P. 1-6. (<https://www.cse.iitb.ac.in/~ganesh/papers/icdar2019b.pdf>)
11. Sakr G., Mokbel M., Darwich A., Mia Nasr Khneisser, Hadi A. Comparing deep learning and support vector machines for autonomous waste sorting // IEEE International Multi-disciplinary Conference on Engineering Technology (IMCET). 2016.
12. Klette, R. Concise Computer Vision: An introduction into Theory and Algorithms // Springer. USA. 2014. P. 339-379.
13. N. Islam, Z. Islam, N. Noor. A Survey on Optical Character Recognition System // Journal of Information & Communication Technology-JICT V. 10. 2016. P. 1-3.
14. Nagy G., Nartker N. A., Rice S. V. Optical character recognition: an illustrated guide to the frontier // Society of Photo-Optical Instrumentation Engineers. 1999. P. 7-57.
15. Singh S., Mamatha K., Ragothaman S., Raj K.D., Anusha N., Susmi Z. Waste Segregation System Using Artificial Neural Networks // HELIX. 2017. № 7. P. 2053-2058.
16. Devi R.S., Vijaykumar V., Muthumeena M. Waste Segregation using Deep Learning Algorithm // International Journal of Innovative Technology and Exploring Engineering (IJITEE). V. 8. 2018. P. 1-3.
17. AlexNet. PyTorch documentation // URL: <https://pytorch.org/vision/main/models/generated/torchvision.models.alexnet.html> (accessed: 11.10.2023)
18. GoogLeNet. PyTorch documentation // URL: [https://docs.openvino.ai/latest/omz\\_models\\_model\\_googlenet\\_v3.html](https://docs.openvino.ai/latest/omz_models_model_googlenet_v3.html) (accessed: 11.10.2023)
19. Mittal P., Singh R. Deep learning-based object detection in low-altitude UAV datasets: A survey // Image and Vision Computing. 2020. P. 4-8.

([https://www.researchgate.net/publication/346201758\\_Deep\\_learning-based\\_object\\_detection\\_in\\_low-altitude\\_UAV\\_datasets\\_A\\_survey](https://www.researchgate.net/publication/346201758_Deep_learning-based_object_detection_in_low-altitude_UAV_datasets_A_survey))

20. J. Hui. SSD THESIS. Dec. 2020 // URL: <https://jonathan-hui.medium.com/ssd-object-detection-single-shot-multibox-detector-for-real-time-processing-9bd8deacoe06>
21. Li C., Li L., Jiang H., Weng K., others. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. 2022. P. 3–10.
22. S. Sahel, M. Alsahafi, M. Alghamdi, and T. Alsubait. Logo detection using deep learning with pretrained CNN models. Engineering, Technology & Applied Science Research, vol. 11, no. 1. 2021. P. 6724–6729.
23. B. Jabir, N. Falih, and K. Rahmani. REF2 THESIS. International Journal of Online & Biomedical Engineering, vol. 17, no. 5, 2021.
24. V. Choudhari, R. Pedram, and S. Vartak. Comparison between YOLO and SSD MobileNet for Object Detection in a Surveillance Drone. Tech. Rep. Oct. 2021.
25. Tesseract documentation // URL: <https://tesseract-ocr.github.io/tessdoc/AddOns.html#tesseract-wrappers> (accessed: 9.04.2023)
26. Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He and Jiajun Liang. EAST: An Efficient and Accurate Scene Text Detector. Computer Vision Foundation. Megvii Technology Inc., Beijing, China // URL: [https://openaccess.thecvf.com/content\\_cvpr\\_2017/papers/Zhou\\_EAST\\_An\\_Efficient\\_CVP\\_R\\_2017\\_paper.pdf](https://openaccess.thecvf.com/content_cvpr_2017/papers/Zhou_EAST_An_Efficient_CVP_R_2017_paper.pdf)
27. Github. Overview of the new neural network system in Tesseract 4.00 // URL: <https://github.com/tesseract-ocr/tessdoc/blob/main/tess4/NeuralNetsInTesseract4.00.md> (accessed: 10.11.2023)
28. Roboflow. E-service // URL: <https://roboflow.com/>
29. MS COCO. Dataset description. // URL: <https://cocodataset.org/#home>
30. Github. Pretrained YOLOv7 model on MS COCO dataset // URL: <https://github.com/WongKinYiu/yolov7> (accessed: 9.11.2023)
31. R. J. Tan. Map thesis. Mar. 2022 // URL: <https://towardsdatascience.com/breaking-down-mean-average-precision-map-ae462f623a52>
32. Zhang H., Xu C., Zhang S. Inner IoU: More Effective Intersection over Union Loss with Auxiliary Bounding Box. 2023. P. 1-3. (<https://arxiv.org/abs/2311.02877>)