IRSTI 28.23.37 https://doi.org/10.26577/JPCSIT.2023.v1.i2.01

I.A. Tereikovskyi[1,*], O.I. Tereikovskyi[2]

[1]National Technical University of Ukraine, Ukraine, Kyiv
[2]National Aviation University, Ukraine, Kyiv
*e-mail: terejkowski@ukr.net

NEURAL NETWORK SYSTEM FOR SELECTING INDIVIDUAL OBJECTS ON RAST IMAGES

Abstract. The article is devoted to solving the problem of increasing the efficiency of neural network tools for semantic segmentation of images. Based on the results of the analysis, it is shown that one of the areas of improvement of such tools is the development of the architecture of the neural network system for the selection of individual objects on raster images. As a result of the conducted research, the architecture of the neural network system for the selection of objects on raster images has been developed, which, due to the adaptation of architectural parameters to the features of the construction and use of modern neural network models intended for the semantic segmentation of images, ensures sufficient accuracy with the permissible amount of use of computing resources. The difference of the developed architecture is the use of functional blocks that are related to the formation of training databases, training of a neural network and selection of an object in the image with the help of a trained neural network. The results of the conducted experiments showed that the application of the proposed architectural solutions allows to develop tools that ensure the achievement of image segmentation accuracy of about 0.8, which corresponds to the accuracy of the best known systems of similar purpose. It is shown that the further increase in accuracy, which can be realized by modifying the parameters of convolutional neural networks on which the encoder and decoder are based, requires additional theoretical research. In addition, the perspective of research related to the improvement of neural network models in the direction of their adaptation to the selection of objects in the video stream is shown.

Key words: raster image, object selection, semantic segmentation, convolutional neural network, system architecture.

Introduction

Nowadays, image recognition tools are widely used in both general and special purpose computer systems. For example, these recognition tools are successfully used in vehicle tracking systems, in information protection systems for biometric authentication of users, and in medical diagnostic systems for automated diagnosis. At the same time, one of the main tasks for effective recognition is the selection of one or more target objects, for example, a car, a human face, or an internal organ, on the input raster image. This task is complicated by possible overlapping and blurring of contours of target objects, changes in their size and location. Also, selection tools should effectively level the variability of image registration conditions, which can lead to changes in the main parameters of the raster image under test – size, resolution, and number of color channels. In addition, noise filtering should be implemented and the angle of video recording should be taken into account. As a result of the described difficulties, the means of selecting objects on images based on traditional computer vision methods have a rather narrow scope of application and require modification even with minor changes in operating conditions, which is confirmed by the results of works [1, 2], which consider tracking methods images of objects using various types of integrated filters. At the same time, well-known examples of successful use and the results of many scientific and practical works testify to significant progress in the field of neural network means of selection, which, although they have proven their high efficiency, still need further improvement.

Literature review

Using neural networks (NM) for selecting objects in images is described in scientific and practical works [3, 4, 10, 12]. The approach used in neural network technologies consists in the fact that a neural network model, which consists of

© 2023 Al-Farabi Kazakh National University

two blocks – an encoder and a decoder – is used to highlight objects on a raster image. The task of the encoder is to determine the multidimensional array of features of the initial image, and the task of the decoder is to obtain a processed image in which each of the pixels receives a marker of relation to one of the selected target objects or the background. That is, on such a NN, the image is first fed along the so-called narrowing path, which actually implements the selection of significant features of the image, and then along the expansion path, which ensures the labeling of image pixels. At the same time, the encoder and decoder are slightly modified convolutional neural networks (CNNs). The modification consists in the fact that not full-size CNN are used, but only blocks of CNN, which are intended for determining significant features. That is, only the convolution and scaling layers are included in the composition of such CNNs, and the fully connected layers are removed. Another feature of the scheme of the neural network model (NNM) is the symmetry of the convolution layers and scaling of the encoder and decoder. In the general case, such symmetry may be absent. The input of the neural network model is an image in RGB format, and the output is a segmented image.

In addition, the works [6, 7, 9, 13] were analyzed to determine the features of adaptation of neural network systems for object selection to the specifics of the applied field. Thus, the article [7] is devoted to the tracking of objects registered by cameras of automated video surveillance systems. In this paper, in addition to neural network extraction methods, traditional object tracking models, including mean displacement, Kalman filter, and particle filter, are considered. A combination of methods such as the mean-shift-based Kalman filter and the mean-shift-based particle filter are also considered. Based on the experimental results, it was determined that the combination of a deep learning network with traditional methods can significantly improve the performance of the object tracking model.

In the article [9], a multi-type vehicle identification scheme from a real-time traffic database is proposed. The basis of the scheme is a convolutional neural network of the FRCNN type. It has been experimentally determined that the success rate of detecting various vehicles is up to 95%, which may vary depending on environmental conditions. It is indicated that the term of determining the type of car takes from 20 to 40 seconds. It is declared that NN FRCNN offers more accurate results compared to RCNN and YOLO.

Article [6] is devoted to the development of a neural network system designed to recognize Indian sign language. The basis of the system is the LeNet-5 type CNN, which is used for neural network analysis of two-dimensional images recorded in RGB format. It is shown that the recognition accuracy of the developed system reaches 96%, however, when the image quality decreases, the accuracy decreases significantly.

In the article [13], the task of developing a multi-modal CNN intended for face recognition is considered. A feature of the proposed model is the use of several CNNs that are trained taking into account the batch normalization of training examples and equipped with a SoftMax classifier. ORL and Yale datasets are used for training. The average accuracy achieved is 94.74% for the ORL and 96.60% for the Yale datasets. It is declared that the application of the developed multi-modal CNN allowed to increase the recognition performance. As a result of the analysis, it can be noted that the accuracy and computational resource-intensiveness of means of distinguishing different types of objects in images, which are based on neural network approaches, significantly outweighs other selection technologies, however, in the vast majority of analyzed works, there is no description of architectural solutions for the recognition system.

Thus, the purpose of this study is to develop the architecture of a neural network system for object selection on bitmap images, which provides sufficient accuracy under variable application conditions.

The architecture of the selection system

Based on the generally accepted methodology of designing the architecture of the information system of general purpose, at the first stage of the design, the mathematical support of the selection system should be formed. Taking into account the known results of research in the field of methods of neural network analysis of raster images, the author's mathematical apparatus was used, given in [5, 14], which provides a description of the procedure for selecting individual objects on raster images using expressions (1-5).

$$Im_{in} \xrightarrow{Pr} Im_{pr} \xrightarrow{NC} Fm \xrightarrow{ND} Im_{out}, \qquad (1)$$

where:

Im_{in}, Im_{pr} – input and processed image;

$\overset{Pr}{\rightarrow}$ – image preprocessing operator;

$\overset{NC}{\rightarrow}$ – neural network image coding operator;

Fm – a tuple of feature matrices obtained as a result of the operation of neural network coding of the image;

$\overset{ND}{\rightarrow}$ – neural network image decoding operator;

Im_{out} – segmented image.

$$E \rightarrow max, \tag{2}$$

$$E = \sum_{i=1}^{I} \alpha_i w_i, \quad \alpha_i \in \{\alpha\}, w_i \in \{w\}, \tag{3}$$

where:

E – function of efficiency of segmentation tools;

I – the number of efficiency parameters;

k_i – value of the i-th efficiency parameter;

α_i – the weight coefficient of the i-th efficiency parameter;

$\{\alpha\}$ – a set of weight coefficients of efficiency parameters;

$\{w\}$ – set of efficiency parameters.

$$\langle \{u_{req}\}, \{u_{con}\}, \{NN_d\}, \{u_{CNN}\}, \{d\}, \{\alpha\}, \{w\}, \rangle \rightarrow$$
$$\rightarrow \langle CNN_{type}^{enc}, CNN_{type}^{dec}, \{CNN^{enc}\}, \{CNN^{dec}\} \rangle, \tag{4}$$

where:

$\{u_{req}\}$ – a set of registration parameters;

$\{u_{con}\}$ – a set of requirements for recognition results;

$\{u_{obj}\}$ – a set containing a description of the selected objects;

$\{NN_d\}$ – a set of available types of CNN;

$\{u_{cnn}\}$ – a set of parameters of available types of CNN;

$\{d\}$ – a set of expert data for building the encoder and decoder model;

$\{\alpha\}$ – the coefficients of the efficiency parameters used in the expression (6);

$\{w\}$ – a set of performance parameters used in the expression (3);

$CNN_{type}^{enc}, CNN_{type}^{dec}$ – a type of CNN encoder and decoder;

$\{CNN^{enc}\}, \{CNN^{dec}\}$ – CNN parameters of $CNN_{type}^{enc}, CNN_{type}^{dec}$ type respectively.

$$J = \frac{\sum_{i=1}^{I} n_i m_i}{\sum_{i=1}^{I} n_i + \sum_{i=1}^{I} m_i - \sum_{i=1}^{I} (n_i - m_i)}, \tag{5}$$

where:

I – the number of points describing the expected output signal of the neural network model;

n_i – value that characterizes the i-th pixel of the segmented image;

m_i- value that characterizes the i-th pixel of the expected output signal.

We note that expression (1) describes the functional of the neural network selection system, expressions (2, 3) describe the procedure for determining the parameters of the CNN, which are the basis of the encoder and decoder, expression (4) describes the procedure for constructing the NNM for the selection of individual objects on a raster image, and expression (5) is used to estimate the accuracy of NNM training.

Taking into account the results of the conducted research, it was determined that the software complex of neural network selection of objects on raster images should provide the following services:

– Determination of a set of parameters that characterize the image.

– Registration of image parameters and saving them to the.

– Registration of the parameters of the mask of the target object, that is, the parameters of the image corresponding to the target object.

– Forming a training sample from parameter matrices of relevant images and object masks.

– Implementation of NNM used in experiments.

– Determination of training parameters, implementation of training and saving of parameters of trained models.

– Implementation of objects selection in the image using NN and demonstration of selection results.

Taking into account the need to provide the specified services and according to the results of [16], it was determined that the following UML diagrams should be built to describe the architecture: precedents, packages, deployment and classes. The development was implemented using the freely available version of the Rational Rose software complex, created by IBM. The precedence diagram, built with the help of the indicated software complex, is shown in fig. 1.

This diagram includes the actors “Client”, “Supervisor” and “Determinant”.

– “Client” – an actor representing a system user responsible for loading the experimental raster image.

– “Supervisor” – an actor corresponding to a system administrator responsible for defining the settings of the feature extraction system on raster images.

– “Determinant” – an actor associated with the system user responsible for defining the target objects to be selected in the raster image.

– Functionality of the selection system:

– Register parameters of image – conducting the registration of raster image parameters.

– Define parameters – defining a set of raster image parameters to be registered.

– Save parameters – saving a set of values of registered parameters.

– Display – displaying selection results.

– Define architecte – specifying the type and parameters of the neural network model.

– Train model – calculation of the set of values of weight coefficients of synaptic connections in the process of training a neural network.

– Save weights – saving the parameters of the values of the weight coefficients obtained as a result of training.

– Realize neural network analysis – implementation of neural network analysis of the experimental raster image using a convolutional neural network.

– Select object – definition of a set of objects to be selected.

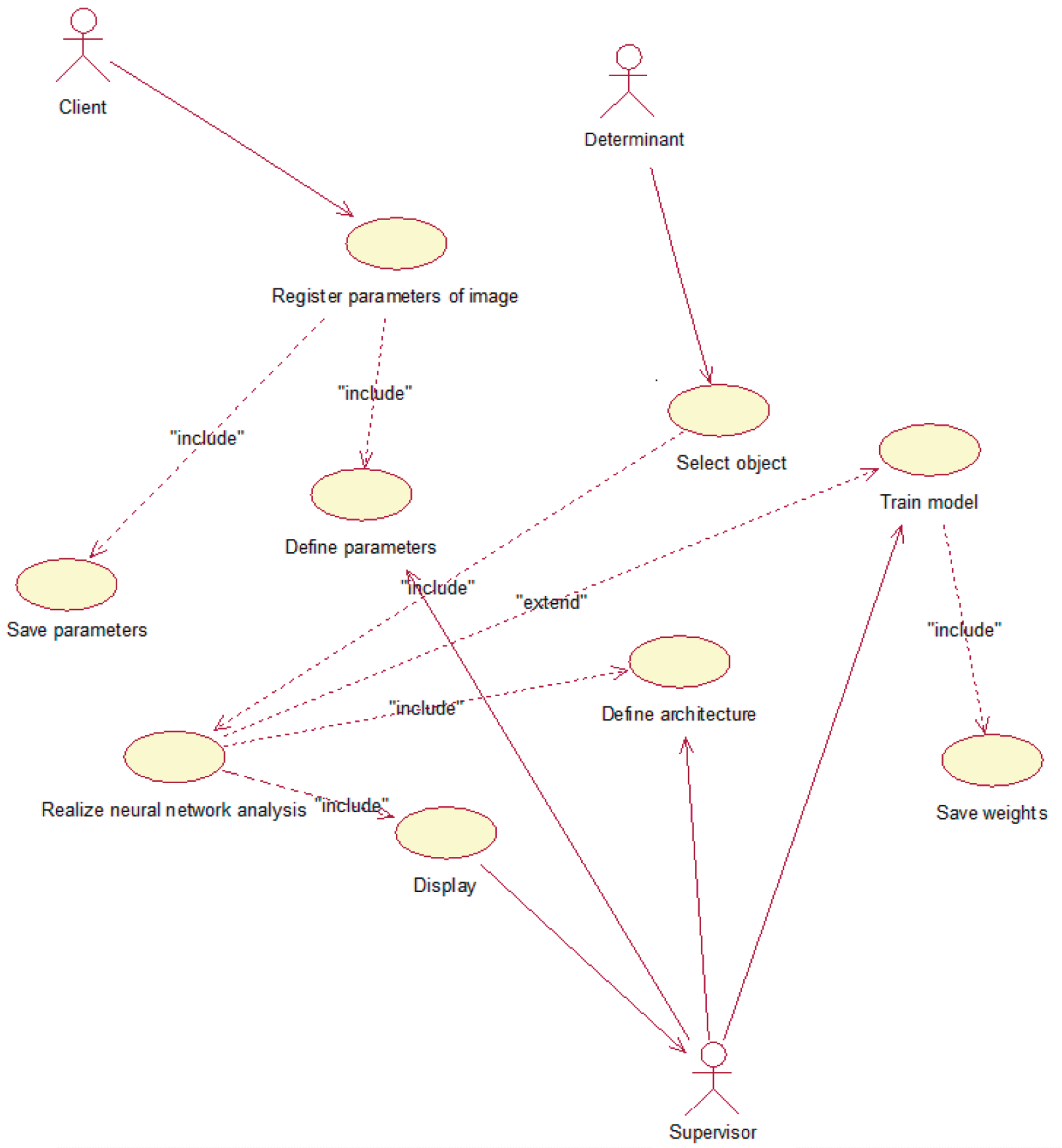

Figure 1 – Selection system precedence diagram

The component diagram and the deployment diagram of the neural network system for selecting objects on raster images are shown in fig. 2 and in fig. 3, respectively. As shown in fig. 2, the neural network selection system includes SystemController, ParamHandler, AISelector and Visualizator modules, which correspond to the interfaces: system management, registration of image parameters, neural network selection of objects, display of selection results.

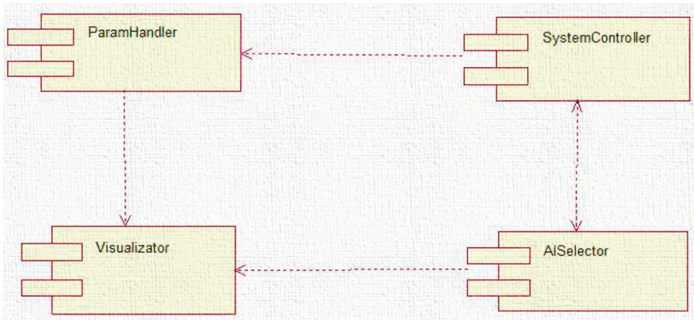

Figure 2 – Component diagram

As shown in fig. 3, the following computing nodes are part of the neural network selection system:

– "Client" – the device on which the user of the system of neural network selection of objects in images works.

– "Server" – the device on which the server software for the system of neural network selection of objects in images is located.

– "Client CPU" – the central processor on the workstation of the system user. This processor can be quite low in terms of speed. All calculations are performed on the system server, only a stable connection to the system server is required from the client.

– "Server CPU" – the central processor on the server system.

– "Server GPU" – graphics processor on the system server.

– "Interconnection Interface" – the communication interface between the system user's computer and the system server.

– "Client Program" – software on a system user's computer that enables the system user to interact with system functionality.

– "ImageProcessor" – software on the system server, which implements the functionality of color image preprocessing.

– "TrainNN" – software on the system server, which implements the functionality of building a neural network, training it and saving the parameters of the neural network to the database.

– "Recognizer" – software on the system server that identifies objects in images using a trained neural network.

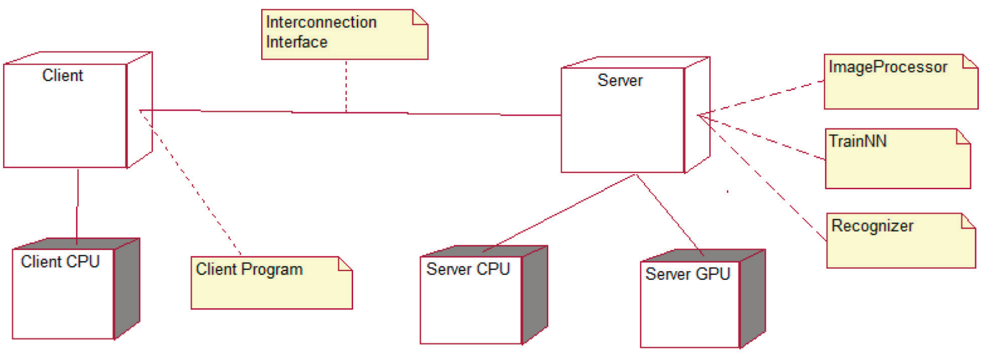

Figure 3 – Deployment diagram

According to those shown in fig. 1-3 UML diagrams, taking into account the feasibility of software development based on an object-oriented approach, it is determined that the software should consist of four classes: ImageProcessor, DatasetProcess, NetworkTrain, NetworkRecognizer. The constructed class diagram is shown in fig. 4. Purpose of classes:

– Class ImageProcessor – processing of images that are input to a neural network that highlights objects. Also, this class is intended for image processing during their preparation for inclusion in the training sample.

– Class DatasetProcess – working with a database that contains images and object masks for neural network training.

– Class NetworkTrain – training of neural network.

– Class NetworkRecognizer – selection of the object in the image using a trained neural network.

The ImageProcessor class contains tools for obtaining data from an image, such as the pixel matrix of the image, the number of channels for displaying pixel color, and the color representation format.

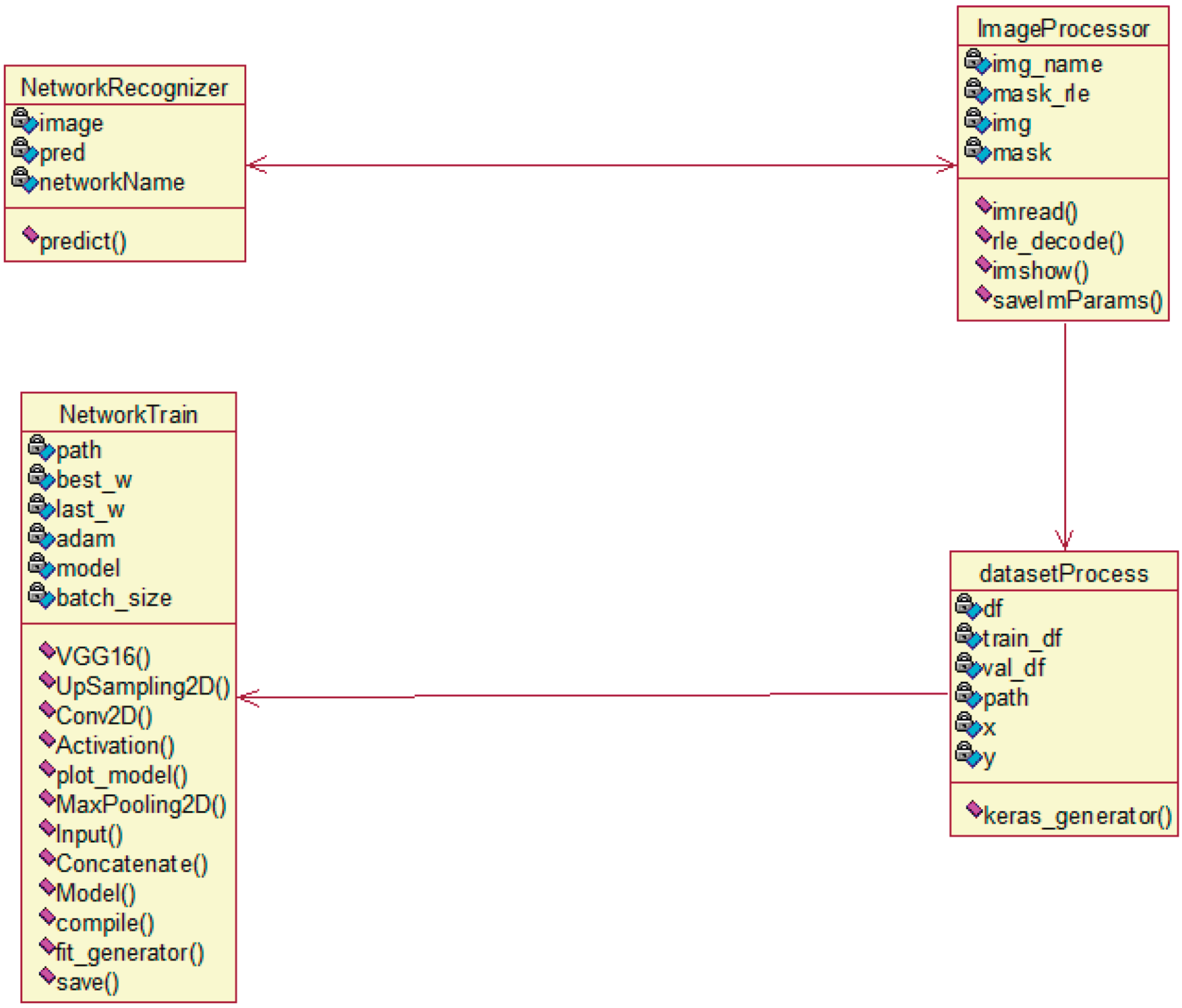

Figure 4 – Class diagram of the neural network selection system

Also, this class provides functionality for processing the mask of objects on images, namely, decoding the mask from the type of coded sequence into a raster format. Attributes of the ImageProcessor class: img_name, mask_rle, img, mask. ImageProcessor class methods: imread, rle_decode, imshow, saveImParams.

The purposes of ImageProcessor class attributes:

– img_name – a variable for storing the path to the image on which the object needs to be selected;

– mask_rle – an array for storing the mask in the form of an encoded sequence;

– img – an array that stores the decoded image;

– mask – an array containing the mask of the object in the image in a decoded form.

– Purpose of ImageProcessor class methods:

– imread – reading image pixels of a certain format into an array;

– rle_decode – decoding of the mask of the object on the image from the view of the sequence to the view in raster format;

– imshow – image demonstration;

– saveImParams – storing data about the name of the image file, its format, the number of image channels, the mask of the object in the image and an array with the representation of image pixels in a certain format in a json format file.

The DatasetProcess class provides a toolkit for generating a training sample for training a neural network. DatasetProcess class attributes: df, train_df, val_df, path, x, y. DatasetProcess class methods: keras_generator.

Purpose of ImageProcessor class attributes:

– df – a structure into which a file containing information about a database with images for neural network training is read;

– train_df – a structure that contains information about the training sample;

– val_df – a structure that contains information about the validation sample;

– path – a variable that stores the path to the database with images for neural network training;

– x – an array containing one batch from the training data sample;

– y – an array containing one batch from the validation data sample.

– Purpose of the ImageProcessor class method:

– keras_generator – to form a database with images for neural network training.

The NetworkTrain class implements the process of building and training a neural network to select objects on raster images. Attributes of the NetworkTrain class: path, best_w, last_w, adam, model, batch_size. NerworkTrain class methods: VGG16, UpSampling2D, Conv2D, Activation, plot_model, MaxPooling2D, Input, Concatenate, Model, compile, fit_generator, save.

Purpose of NetworkTrain class attributes:

– path – variable with the path along which the trained model or its checkpoints are stored.

– best_w – a variable that contains information about the checkpoint in which the state of the best-trained version of the neural network is stored.

– last_w – a variable that contains information about the checkpoint where the latest version of the neural network is stored.

– adam – a variable that represents an optimizer of the Adam type.

– model – a variable that contains a neural network.

– batch_size – variable for setting the batch size.

– Purpose of NetworkTrain class methods:

– VGG16 – loading the pre-trained VGG16 neural network.

– UpSampling2D – implementation of the upsampling layer.

– Conv2d – convolutional layer implementation.

– Activation – setting the activation function for a certain layer of the neural network.

– plot_model – demonstration of neural network architecture in the form of a drawing.

– MaxPooling2D – implementation of maxpooling layer.

– Input – implementation of the input layer of the neural network.

– Concatenate – implementation of the concatenation layer.

– Model – creation of a neural network model from previously created layers.

– compile – setting parameters of neural network training.

– fit_generator – neural network training on the training sample, which is presented in the form of a generator.

– save – saving the neural network model after completing its training.

The NetworkRecognizer class provides a toolkit to highlight an object in an image using a trained neural network. Attributes of the NetworkRecognizer class: image, pred, networkName. The only method of the NetworkRecognizer class is the predict method.

Purpose of NetworkRecognizer class attributes:
– image – a variable that stores the path to the target image on which the object needs to be selected;
– pred – an array with a mask of the selected object in the image using a trained neural network;
– networkName – a variable that stores the path to the trained neural network.

The purpose of the predict method is to highlight an object in an image using a trained neural network.

The construction of the described UML-diagrams provided the possibility of software development and the transition to conducting experimental studies aimed at verifying the proposed neural network system for the selection of objects on raster images. The Python programming language and the TensorFlow library were used to develop the software.

Computer experiments

In the process of computer experiments, the effectiveness of using the built neural network system for selecting a car-type object on color images was investigated. The Carvana Image Masking Challenge database, which is freely available at the link https://www.kaggle.com/c/carvana-image-masking-challenge, was used as a source for forming the training sample of the neural network. This database contains 5088 high-contrast images recorded in RGB format. The size of a single image is 256×256 pixels. Taking into account the size and color format of the original and segmented images, the expediency of feeding the raw image to the encoder input is determined. Accordingly, the set of registered parameters corresponds to a three-channel image with a size of 256x256 pixels.

With the use of expressions (2, 3), it was determined that the most efficient encoder is based on the VGG-type CNN, and the most efficient type of decoder is the decoder based on one-stage resampling. The input and output parameters of the NNM training examples correspond to the original and segmented image shown in Fig. 6. The design parameters of VGG were adapted to the analysis of halftone images of size 256x256 by increasing the input field of the CNN. The training sample is formed on the basis of the described database of images and is divided into training, validation and test samples. The training sample of NN is supplemented by the augmentation of training examples. Of these, 8,000 examples were used as training data, and 1,088 examples were used for validation and testing. According to the recommendations [8, 15], the volume of the training sample is 8000, and the total volume of the validation and test sample is 1680 examples. The Jacquard coefficient, described by expression (5), was used to assess the accuracy of the NNM.

Graphs of the accuracy of car selection are shown in fig. 5. Comparison of graphs shown in fig. 5 with the results of studies [8, 13] allow us to state that when using the proposed NNM, the segmentation accuracy is approximately 0.8, which, according to the results of the experiments, is more than 2 times higher than the selection accuracy that can be achieved using other NNMs. A further increase in accuracy, which may be implemented by modifying the parameters of the CNN on which the encoder and decoder are based, requires additional theoretical research. In addition, a promising way to develop a neural network model is its adaptation to the selection of objects in a video stream, which should take into account the peculiarities of the sequential analysis of individual frames of such a stream.

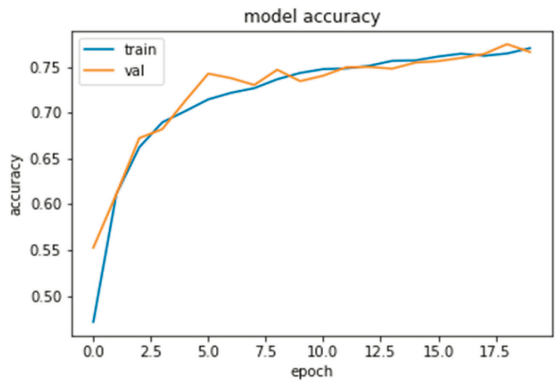

Figure 5 – Graphs of the accuracy of selecting the contour of the car on the raster image

Therefore, the results of the experimental experiments show that the application of the proposed architectural solutions, which are based on the results of [5, 14], allow to develop tools that ensure the achievement of image segmentation accuracy of about 0.8. At the same time, it was possible to avoid conducting complex experiments aimed at determining the effective architecture of the NNM, which makes it possible to minimize the amount of computing resources aimed at the construction of the NNM. Thus, the use of the proposed architecture of the neural network selection system allows, while reducing the computing resources associated with the construction of the NNM, to achieve segmentation accuracy, which is correlated with the accuracy of the best known systems of a similar purpose [11, 13, 15].

Conclusion

As a result of the conducted research, the architecture of the neural network system for the selection of objects on raster images has been developed, which, due to the adaptation of architectural parameters to the features of the construction and use of modern neural network models intended for the semantic segmentation of images, ensures sufficient accuracy with the permissible amount of use of computing resources. The difference of the developed architecture is the use of functional blocks related to: image processing during their preparation for entering the training sample and entering the neural network; using a database with images and object masks used for neural network training; neural network training; selection of the object in the image using a trained neural network. The results of the conducted experiments showed that the application of the proposed architectural solutions allows to develop tools that ensure the achievement of image segmentation accuracy of about 0.8, which corresponds to the accuracy of the best known systems of similar purpose. It is shown that the further increase in accuracy, which can be realized by modifying the parameters of convolutional neural networks on which the encoder and decoder are based, requires additional theoretical research. In addition, a promising way to develop a neural network model is its adaptation to the selection of objects in a video stream, which should take into account the peculiarities of the sequential analysis of individual frames of such a stream.

References

1. Abraham J., Paul V. “An imperceptible spatial domain color image watermarking scheme”. Journal of King Saud University – Computer and Information Sciences. 2019. Vol. 31 (1), pp. 125-133.

2. Adithya U., Nagaraju C., “Object Motion Direction Detection and Tracking for Automatic Video Surveillance”, International Journal of Education and Management Engineering (IJEME), Vol.11, No.2, pp. 32-39, 2021. DOI: 10.5815/ijeme.2021.02.04.

3. Dmitry A. “Segmentation Object Strategy on Digital Image”. Journal of Siberian Federal University. Engineering & Technologies. 2018. № 11(2), pp. 213-220.

4. Cherrat, Rachid Alaoui, Hassane Bouzahir. “Score Fusion of Finger Vein and Face for Human Recognition Based on Convolutional Neural Network Model”. International Journal of Computing, 2020. 19(1), pp. 11-19.

5. Hu Z., Tereykovskiy I., Zorin Y., Tereykovska L., Zhibek A. Optimization of convolutional neural network structure for biometric authentication by face geometry. Advances in Intelligent Systems and Computing. 2019. Vol. 754, pp. 567-577.

6. Jun Shen. “Motion detection in color image sequence and shadow elimination”. Visual Communications and Image Processing. 2014. Vol. 5308, pp. 731-740.

7. Kong T., et al. “FoveaBox: Beyound Anchor-Based Object Detection”, IEEE Trans. Image Process. 29 (2020), pp. 7389–7398.

8. Liu, X.-P., Li, G., Liu, L., Wang, Z. “Improved YOLOV3 target recognition algorithm based on adaptive eged optimization”. Microelectron. Comput. 2019. Vol. 36, pp. 59–64.

9. Prilianti, K. R., Anam, S., Brotosudarmo, T. H. P., Suryanto, A. “Non-destructive Photosynthetic Pigments Prediction using Multispectral Imagery and 2D-CNN”. International Journal of Computing. 2021. 20(3), pp. 391-399.

10. Reja, S. A., Rahman, M. M. “Sports Recognition using Convolutional Neural Network with Optimization Techniques from Images and Live Streams”. International Journal of Computing, 2021. 20(2), pp. 276-285.

11. Ronneberger O., Fischer P., Brox T. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2015. Vol.9351, pp. 234-241.

12. Shkurat O. et al. “Image Segmentation Method Based on Statistical Parameters of Homogeneous Data Set”. Advances in Intelligent Systems and Computing. 2020. Vol. 902, pp. 271–281.

13. Taqi A., Awad A., Al-Azzo F., Milanova M. “The impact of multi-optimizers and data augmentation on TensorFlow convolutional neural network performance”. Proceedings of the 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR). 2018, pp. 140-145.

14. Tereikovskyi I., Hu Z., Chernyshev D., Tereikovska L., Korystin O., Tereikovskyi O. “The Method of Semantic Image Segmentation Using Neural Networks”. International Journal of Image, Graphics and Signal Processing (IJIGSP), Vol.14, No.6, pp. 1-14, 2022

15. Wu C., Wen W., Afzal T., Zhang Y., Chen Y. “A compact DNN: Approaching GoogLeNet-Level accuracy of classification and domain adaptation”. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.