Evaluation of the accuracy of the neural network algorithm for object recognition in security systems

Andrii Sahun[1,†], Vladyslav Khaidurov[2,†] and Valerii Lakhno[1,*,†]

[1] National University of Life and Environmental Sciences of Ukraine, 15 Heroyiv Oborony str., 03041 Kyiv, Ukraine
[2] National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37 Beresteyskiy ave., 03056 Kyiv, Ukraine

Abstract

The study presents the results of applying the main known metrics used to evaluate the performance and accuracy of algorithms and neural network models on different classes for the task of graphic content recognition in security systems. For the analysis, different classes of images processed by the neural network algorithm were compared. To evaluates the quality of the algorithm's training based on the results of graphical pattern recognition, nine different metrics for the five conducted correct classification computational experiments were used. The sample used in research, the CamVid benchmark video dataset for training the neural network model, shows different training results for different recognition classes, with this indicator ranging from 38.15% to 97.07% when using the VGG-16 function. At the same time, the highest standard deviation of accuracy, with a value of 0.030351419, was recorded only for the "Pavement" class. This indicates the imperfection of the CamVid training dataset. It should be modified to improve recognition quality by increasing the size and number of test images.

Keywords

distance metrics, neural network, classifier, algorithm's quality evaluation, image recognition

1. Introduction

Machine learning and neural networks are closely related, as neural networks are one of the primary technologies in the field of machine learning [1–3]. These algorithms are particularly widely used in security systems. In machine learning, several key metrics are used to evaluate model performance. These metrics help to understand how well the model is performing the given task and to identify areas where it can be improved. There are several metrics for evaluating different neural network algorithms [4]. All of them are used to analyze the recognition of various properties and characteristics of neural network recognition algorithms [5]. These are useful for creating an optimal model of a graphic information recognition system. The most important ones are the metrics for evaluating the quality of learning [6].

Therefore, it is of particular interest to understand whether there is a correlation between the weight coefficient of the presence of a particular classification object in graphic object recognition and the accuracy of such recognition. For example, in the works [7–10], the use of metrics such as Distance metrics is considered, while in the research [2] the use of Euclidean Distance. However, the formulation of the task differs from the identification of graphical objects. At the same time, [2] emphasizes that the accuracy of identification (recognition) was 96.38% as the maximum value. In another research related to practical tasks of recognition and identification of graphical images, the average recognition (identification) accuracy is reported at 76.78% [11].

Therefore, it is important to assess how accurately graphical patterns are recognized in a specific practical task [5]. The same systems are used in specific tasks, such as security systems. In particular, the corresponding modules are part of intelligent access control systems [12].

2. Main part

Now mostly part of more complex practical application systems, which are known as Image Identification and Recognition Systems (IIRS). IIRS are often used both for detecting defects on parts within quality control systems according to ISO-9000 standards and for detecting and recognizing the values of vehicle license plates. Based on the results of the IIRS module, the intelligent system can automatically make decisions about granting or denying access to a secured area for a specific object. Another application of such systems is machine vision systems. The common principle of construction for all such systems is:

1) The technical part of acquiring and initial processing of the image.

CPITS-II 2024: Workshop on Cybersecurity Providing in Information and Telecommunication Systems II, October 26, 2024, Kyiv, Ukraine
* Corresponding author.
† These authors contributed equally.
avd29@ukr.net (A. Sahun);
allif0111@gmail.com (V. Khaidurov);
lva964@nubip.edu.ua (V. Lakhno)
0000-0002-5151-9203 (A. Sahun);
0000-0002-4805-8880 (V. Khaidurov);
0000-0001-9695-4543 (V. Lakhno)

2) The technical or software part for analyzing and classifying image elements.
3) The subsystem for registration/identification and summarization of recognition data.

In all similar IIRS systems, this intelligent module with a neural network-based algorithm plays a central role. The accuracy of this module determines the overall performance of the entire system.

For those practical tasks where IIRS is now mostly used, a mathematical apparatus based on neural networks with different types of training is applied [13–17]. The choice of the type of neural network training model is not the subject of this study. And the aspects related to this choice are described, in particular [2, 11, 13–17].

The test model chosen is the neural network model described in [2]. This model has several layers of neurons (Fig. 1).

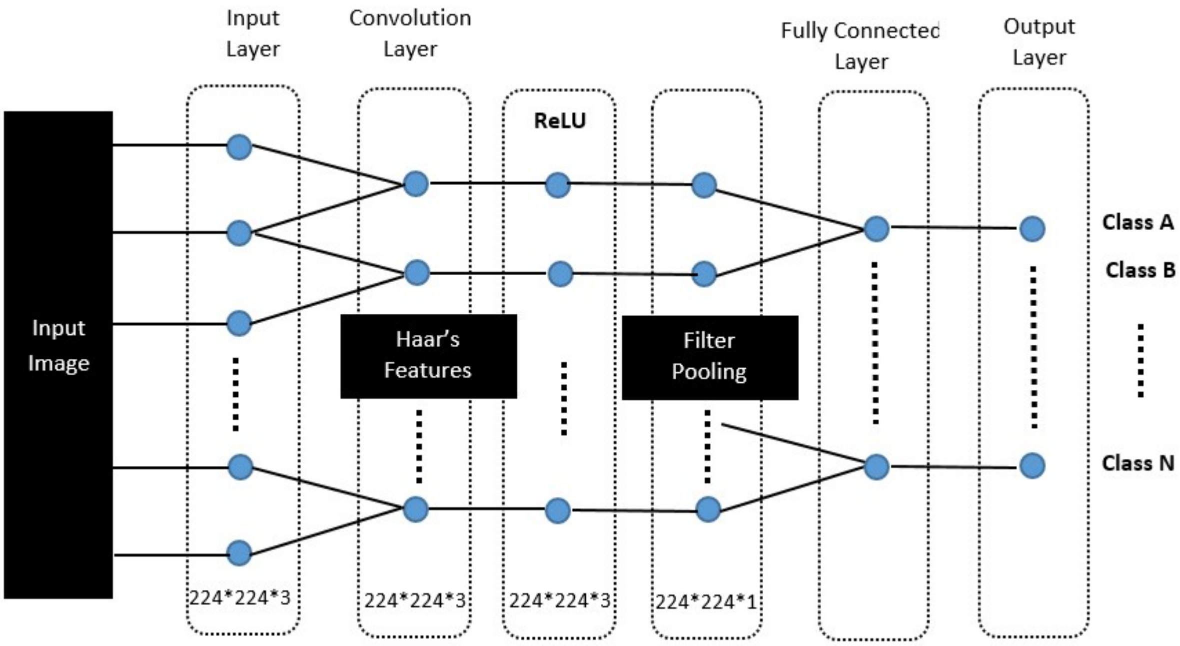

Figure 1: Layered neural network architecture of the IIRS model with Haar feature

Given the practice of using neural network-based algorithms in recognition and identification systems, a deep-learning neural network model was chosen. This is due to several existing advantages of such models for graphic identification/recognition tasks [1, 3].

The main goal of the study is the evaluation of the accuracy of a neural network algorithm in the task of recognizing graphic content.

The neural network diagram of the IIRS shown in Fig. 1 operates with the Haar feature. This approach is most effective when using a deep-learning neural network.

In the basic model described in [2], the input layer of neurons receives initial data, such as the intensity of each pixel and Haar features for various graphical objects to be identified (bushes, trees, cars, roads, sky, sidewalk elements, fences, pedestrians, etc.).

3. Applying distance metrics for neural networks

In the Matlab environment, there is a built-in function vgg16() which implements the architecture of a deep neural network. There is also a function analogous to it, vgg19(). The first function operates with 16 convolutional and fully connected layers of neurons, including 13 convolutional and 3 fully connected layers. This function is used for image classification in the process of pattern recognition. The vgg16() function in MATLAB returns a neural network object but does not contain a specific method for computing distances (metrics) between feature vectors for processed images.

The vgg19() function also implements the architecture of a deep neural network and has an input size of 224×224×3. Unlike vgg16, the neural network in the vgg19 network is trained and fine-tuned on a dataset of graphical data containing over 1,000,000 images and 1000 classes. This allows this neural network to have more powerful capabilities for feature extraction in images. To define metrics based on VGG19 in MATLAB, we first need to load and prepare the VGG19 model, and extract image features from a specific layer of the neural network. After this, both vgg16 and vgg19 functions must use different metrics to compare these features. That is, neither function has built-in distance metric determination.

To use distance metrics with feature vectors extracted from the VGG16 model in MATLAB, we have to follow these steps:

1) Loading and preparing the VGG16 Model (use the pre-trained VGG16 model to extract feature vectors from images.

2) Extracting Feature Vectors (feed your images through the VGG16 model to get the feature vectors).
3) Computing Distance Metrics (use different distance metrics to compare the feature vectors).

Below are the main known metrics used to evaluate the performance of algorithms and neural network models on different classes of graphic content recognition. These metrics are used in machine learning [2].

Accuracy metric in machine learning. Accuracy shows the proportion of correctly classified objects among all objects. This metric is well suited for tasks where classes are balanced. The expression below provides an example of obtaining the accuracy metric in machine learning algorithms [18]:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (1)$$

where TP (True Positive) is the number of correct positive classifications, TN (True Negative) is the number of correct negative classifications, FP (False Positive) is the number of incorrect positive classifications, and FN (False Negative) is the number of incorrect negative classifications.

Precision metric in machine learning. Precision measures the proportion of correctly classified positive objects among all objects classified as positive. This metric is important when the cost of false positive results is high. In (2) we present the expression for computing the accuracy metric in machine learning.

$$Precision = \frac{TP}{TP + FP}. \quad (2)$$

Recall metric in machine learning. Recall measures the proportion of correctly classified positive objects among all actual positive objects. This metric is important when the cost of false negative results is high. The following expression is used to compute this metric (2):

$$Recall = \frac{TP}{TP + FN}. \quad (3)$$

The F1-score metric of recall. The F1-score is the harmonic mean between precision and recall. It is useful when balancing these two metrics is necessary. It is calculated according to the expression provided below:

$$F1 - score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}. \quad (4)$$

Intersection over Union metric. IoU is used to evaluate the quality of segmentation and object detection by measuring the ratio of the intersection area of predicted and ground truth objects to their union area. It is calculated according to the expression provided below:

$$IoU = \frac{Area\ of\ Intersection}{Area\ of\ Union}. \quad (5)$$

Mean Average Precision metric. Average precision is calculated for each category and then averaged across all categories. This metric is often used for object detection tasks. Such a metric is particularly relevant for evaluating the training quality of this neural network-based model. The metric value can be determined using the expression provided below:

$$mAP = \frac{1}{N}\sum_{i}^{N} AP_i, \quad (6)$$

where N is the number of categories.

Confusion matrix. This matrix shows the number of correct and incorrect classifications for each class. It includes TP, FP, TN, and FN for each category.

Area under the ROC curve. The ROC curve shows the relationship between TPR and FPR at different thresholds. The area under the curve (AUC) measures the model's ability to distinguish between classes (7).

$$AUC = \int_0^1 TPR(t)\, dFPR(t). \quad (7)$$

False Positive Rate (FPR). The FPR measures the proportion of false positive results among all negative examples during training.

$$FPR = \frac{FP}{FP + TN}. \quad (8)$$

False Negative Rate (FNR). The FNR measures the proportion of false negative results among all positive examples during training.

$$FNR = \frac{FN}{FN + TP}. \quad (9)$$

The above-mentioned metrics help objectively assess the quality and effectiveness of the model for identifying graphical objects in a video surveillance system based on neural networks, as well as choosing the most efficient algorithm for specific conditions and tasks.

In this research, all the evaluation metrics (1)–(9) listed above were used to assess the quality of model training.

Table 1 shows the quality metric values of the algorithm training obtained in 5 computational experiments (calculated result of correct classification of objects for all classes).

Table 1
The quality metric values of the algorithm training obtained in 5 computational experiments

Class name	Exp #1	Exp #2	Exp #3	Exp #4	Exp #5
Sky	0,9266	0,9320	0,9479	0,9348	0,9818
Building	0,7987	0,8647	0,9181	0,9126	0,8786
Pole	0,8698	0,9397	0,9483	0,9455	0,9541
Road	0,9518	0,9867	0,9551	0,9749	0,9848
Pavement	0,4188	0,4468	0,6394	0,5463	0,5070
Tree	0,4342	0,4347	0,4896	0,4549	0,4465
SignSymb	0,3251	0,3264	0,4621	0,3698	0,4243
Fence	0,4921	0,5825	0,6245	0,5978	0,6582
Car	0,8988	0,9218	0,9542	0,9594	0,9732
Pedestr	0,758	0,8281	0,9104	0,8328	0,8972
Bicyclist	0,8145	0,8172	0,9492	0,8576	0,8207

Fig. 2 shows weights coefficient indicators object recognition for the test video data segment.

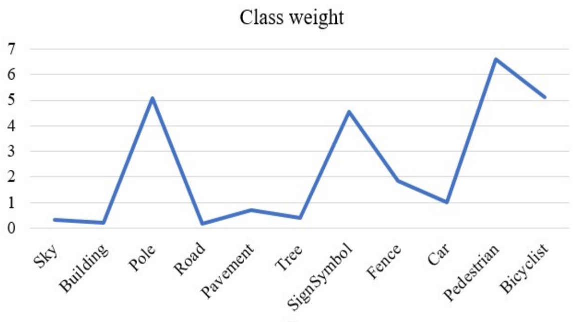

Figure 2: Weight coefficient indicators for each of the recognition classes

The significance of the Intersection over Union (IoU) metric, calculated for each of the semantic classes, lies in its ability to measure the accuracy of the neural network's recognition performance. IoU assesses how well the predicted segmentation overlaps with the ground truth segmentation for each class. Higher IoU values indicate better performance, meaning the predicted areas closely match the actual areas. This metric is crucial for evaluating the effectiveness and reliability of the neural network in accurately recognizing and segmenting different semantic classes within the graphical content. In Fig. 3 we can see values of the IoU Accuracy evaluation metric.

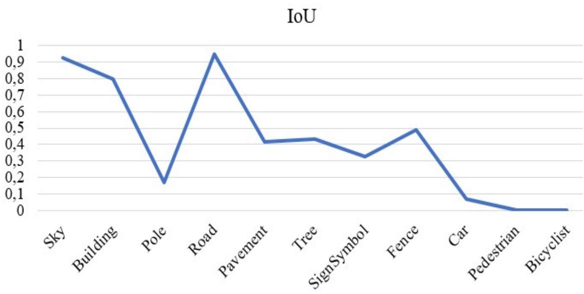

Figure 3: Intersection over Union metric score calculated for each of the semantic classes

As can be understood from above, the most important and resultant indicator of model training quality is the IoU (Intersection over Union) metric. The result of the correct classification of objects for each class in the 5 conducted computational experiments values for different detection classes are presented in Figs. 4–6.

Considering that the model was trained on 421 images, it can be considered that its training level may be sufficient for the graphical identification task at hand. But we see that the training quality even for the same semantic classes varies significantly across the different 5 experiments.

The smallest value of such a deviation will be for objects of the "Bicyclist" class at 0.76%, and the largest will be for objects of the "Fence" class at 25.25%.

Such a difference can be explained by various reasons. For example, the imperfection of the algorithm or the insufficient quality or length of the training data sample.

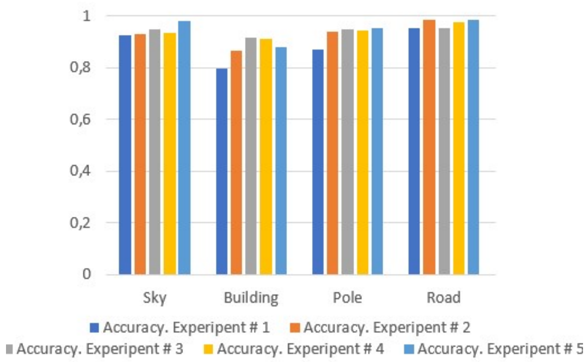

Figure 4: The result of the correct classification of objects for classes "Sky", "Building", "Pole", and "Road" in the 5 conducted computational experiments

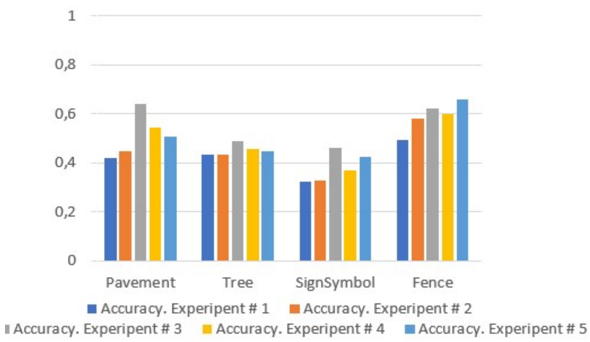

Figure 5: The result of the correct classification of objects for classes "Pavement", "Tree", "SignSymbol", and "Fence" in the 5 conducted computational experiments

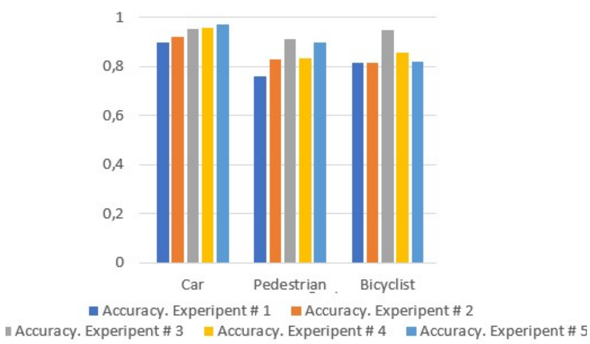

Figure 6: The result of the correct classification of objects for classes "Car", "Pedestrian", and "Bicyclist" in the 5 conducted computational experiments

As shown by the calculations obtained in Table 1, the most accurate results of the learning algorithm NM were obtained for the classes: "Road"—97.06%, "Sky"—94.46%, and "Car"—94.16% accuracy of correct recognitions. At the same time, the recognition quality of images of the type "SignSymbol" was 38.15%, and "Tree" had 45.19% accuracy of correct recognition. The average learning quality of this algorithm on the test fragments was 75.42%.

4. Conclusions

Analyzing the data presented and visualized in Table 1 and Figs. 4–6, it can be said that the quality of the learning algorithm described in [2] significantly depends on the accuracy of the training. The accuracy of image recognition in neural network-based algorithms is highly dependent on the quality of training. Here are some key points of this dependence: Training Data Quality; Training Data Quantity; Preprocessing; Algorithm Complexity; Training Process. The sample used in the study [2] CamVid benchmark video dataset for training the neural network model shows different training results for different recognition classes. This indicator ranges from 38.15% to 97.07% when using the VGG-16 function. It can be noted that all the provided training quality metrics on the same recognition classes yield approximately the same accuracy values. While the variance (standard deviation) indicator is highest only for the "Pavement" class. It amounts to 0.030351419.

The obtained average recognition accuracy of graphical objects at 75.42% is comparable to the recognition rate of 98.7%. This indicates insufficient training quality due to the shortcomings of the training dataset.

It can be assumed that the simplest way to improve recognition accuracy could also be using a more complex neural network algorithm. Such one present in MatLab is called VGG-19 [19–21]. Also, to improve the quality of graphic content recognition, it is necessary to use another, higher-quality training dataset that contains a larger number of relevant sets of graphic datasets. We can also create an improved CamVid benchmark video dataset. As known, benchmark video dataset improvement can also significantly enhance the performance of the deep learning neural network algorithm [22, 23].

References

[1] F. Ahmad, T. Ahmad, Image Mining Based on Deep Belief Neural Network and Feature Matching Approach Using Manhattan Distance, Comput. Assisted Methods Eng. Sci. 28(2) (2021) 139–167. doi: 10.24423/cames.323.

[2] A. Sahun, V. Khaidurov, V. Bobkov, Model of Graphic Object Identification in a Video Surveillance System based on a Neural Network. Cybersecurity Providing in Information and Telecommunication Systems, vol. 3654 (2024) 361–367.

[3] J.-H. Lee, Minimum Euclidean Distance Evaluation using Deep Neural Networks, AEU – Int. J. Electron. Commun. 112 (2019) 152964. doi: 10.1016/j.aeue.2019.152964.

[4] K. Khorolska, et al., Application of a Convolutional Neural Network with a Module of Elementary Graphic Primitive Classifiers in the Problems of Recognition of Drawing Documentation and Transformation of 2D to 3D Models, J. Theor. Appl. Inf. Technol. 100(24) (2022) 7426–7437.

[5] V. Dudykevych, et al., Detecting Deepfake Modifications of Biometric Images using Neural Networks, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, CPITS, vol. 3654 (2024) 391–397.

[6] H. Mohammad, M. N. Sulaiman, A Review on Evaluation Metrics for Data Classification Evaluations, Int. J. Data Mining Knowledge Manag. Process 5 (2015) 01–11. doi: 10.5121/ijdkp.2015.5201.

[7] O. V. Herasina, V. I. Korniienko Global and Local Optimization Algorithms in the Problem of Identification of Complex Dynamic Systems, Inf. Process. Syst. (6) (2010) 73–77.

[8] V. Lakhno, et al., Information Security Audit Method Based on the Use of a Neuro-Fuzzy System, LNNS 232 (2021) 171–184.

[9] H. G. Schuster, Deterministic Chaos: Introduction and Recent Results, Springer, Berlin, Heidelberg (1992). doi: 10.1007/978-3-642-95650-8_2.

[10] A. Pérez-Romero, et al., Evaluation of Artificial Intelligence-based Models for Classifying Defective Photovoltaic Cells. Appl. Sci. 11 (2021) 4226. doi: 10.3390/app11094226.

[11] L. Ljung, et al., Deep Learning and System Identification, IFAC-PapersOnLine, 53(2) (2020) 1175–1181. doi: 10.1016/j.ifacol.2020.12.1329.

[12] P. Anakhov, et al., Evaluation Method of the Physical Compatibility of Equipment in a Hybrid Information Transmission Network, J. Theor. Appl. Inf. Technol. 100(22) (2022) 6635–6644.

[13] V. Lakhno, et al., Development Strategy Model of the Informational Management Logistic System of a Commercial Enterprise by Neural Network Apparatus, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 2746 (2020) 87–98.

[14] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, The MIT Press (2016).

[15] D. F. Kandamali, et al., Machine Learning Methods for Identification and Classification of Events in ϕ-OTDR Systems: A Review, Applied Optics, 61(11) (2022) 2975. doi: 10.1364/ao.444811.

[16] S. Bickler, Machine Learning Identification and Classification of Historic Ceramics, Archaeology in New Zealand, 61 (2018) 48–58.

[17] O. Rainio, J. Teuho, R. Klén, Evaluation Metrics and Statistical Tests for Machine Learning, Sci. Rep. 14 (2024) 6086. doi: 10.1038/s41598-024-56706-x.

[18] S. Orozco Arias, et al., Measuring Performance Metrics of Machine Learning Algorithms for Detecting and Classifying Transposable Elements, Processes. 8 (2020) 1–19. doi: 10.3390/pr8060638.

[19] A. V. Ikechukwu, et al., ResNet-50 vs VGG-19 vs Training from Scratch: A Comparative Analysis of the Segmentation and Classification of Pneumonia from Chest X-ray Images, Global Transitions Proceedings, 2(2) (2021) 375–381. doi: 10.1016/j.gltp.2021.08.027.

[20] A. Saleh, R. Sukaik, S. Abu-Naser, Brain Tumor Classification Using Deep Learning (2020) 131–136. doi: 10.1109/iCareTech49914.2020.00032.

[21] L. Ali, et al., Performance Evaluation of Deep CNN-Based Crack Detection and Localization Techniques for Concrete Structures, Sensors, 21 (2021). 1688. doi: 10.3390/s21051688.

[22] S. Richter, et al., Playing for Data: Ground Truth from Computer Games, LNCS 9906 (2016). doi: 10.1007/978-3-319-46475-6_7.

[23] P. Ravishankar, A. Lopez, G. M. Sanchez, Unstructured Road Segmentation using Hypercolumn based Random Forests of Local experts (2022). doi: 10.48550/arXiv.2207.11523.