RT&A, No 4(80)
Volume 19, December, 2024

IMPROVING THE RELIABILITY OF RECOGNIZING POTENTIALLY HAZARD UNDERWATER OBJECTS

Artyukhin Valerii, Vyalyshev Alexander, Zinoviev Sergey, Tuzov Fedor

•

All-Russian Research Institute for Civil Defence of the EMERCOM of Russia (the Federal Science and High Technology Center), Moscow, Russia
ikshot@mail.ru, vialyshev@rambler.ru, golf1972@yandex.ru
Corresponding author: Tuzov Fedor fedor-tuz@mail.ru

Abstract

The process of recognizing an underwater object and detecting potentially hazardous underwater object is very important in underwater operations. To facilitate the work of the side scan sonar operator, this paper proposes to increase the reliability of recognizing hydroacoustic images of potentially hazardous underwater objects in automatic mode. Based on the analysis of sonar images received from the side scan sonar, an image of an object is formed, which is then recognized (classified) as belonging to a certain class of objects. Five classes of recognized objects are defined. A convolutional neural network used to determine whether an underwater potentially dangerous object belongs to one of the classes is described. Filters for initial sonar images for acceleration of neural network operation are defined. Algorithms and software for forming an image of the object and making a decision on its belonging to one or another class are developed. It is shown that the use of convolutional neural network allows to determine the correct class of the object with an accuracy of 91%.

Keywords: reliability, side scan sonar, potentially hazardous underwater object, neural networks, classification, classification features, information

I. Introduction

The reliability of underwater object type detection from side scan sonar image processing is an important task for determining the hazard of a submerged object. The usual method of recognizing objects on the basis of classification features and operator's work requires a long time and with a large number of detected objects has a low probability of correctly determining the type of object. On the bottom of the world ocean there are a large number of objects that may pose a potential danger to the population and the environment -potentially hazardous underwater objects (PHUO) [1]. Such objects primarily include:

- sunken nuclear submarines and their structures with used nuclear fuel;
- sunken and sunken ships and vessels with ammunition on board;
- sunken containers and barrels with hazardous substances;
- ammunition (mines, shells, etc.).

The most difficult problem has been and remains the search for small-sized objects, as well objects partially submerged in the ground and silt. These include bottom and near-bottom mines, emergency containers with toxic and radioactive substances, lost ammunition (bombs, torpedoes, warheads, etc.), qualified by hydroacoustic specialists as “small-sized objects”. The “black boxes” of aircrafts that have crashed over the sea fall into the same category.

In recent years, various neural network architectures have been actively implemented in the field of object recognition in side scan sonar images. The most common approaches include:

convolutional neural networks (CNN). Converged networks are the basis for many computer vision tasks, including object recognition. They perform well in object identification and classification tasks due to their ability to extract spatial hierarchies of features [2].

YOLO (You Only Look Once). This neural network architecture enables real-time object detection. YOLO divides the image into a grid and predicts object boundaries and class membership probabilities for each cell [3].

SSD (Single Shot MultiBox Detector). Similar to YOLO, SSD is another architecture for real-time object detection. It utilizes a variety of sizes of predicted rectangles to better handle objects of different scales [4].

Faster R-CNN. This model combines the strengths of CNN and regression algorithms for detection. Faster R-CNN utilizes Region Proposed Networks (RPNs), which achieves high accuracy, albeit at a higher resource cost [5].

The universality of the approach proposed in the paper allows the developed system to be used for classification of any objects on the bottom of the water area.

II Methods

For detection of underwater potentially dangerous objects by hydroacoustic means, as a rule, side-scan sonars (SSS) with a range of up to hundreds of meters and designed for operation in shallow water conditions in coastal and shelf zones of seas are used. As a rule, they have relatively high operating frequencies and good range resolution.

Side scan sonar provides a 3D image of the bottom relief with objects located on it. An example of an image with a submerged vessel is shown in figure 1.

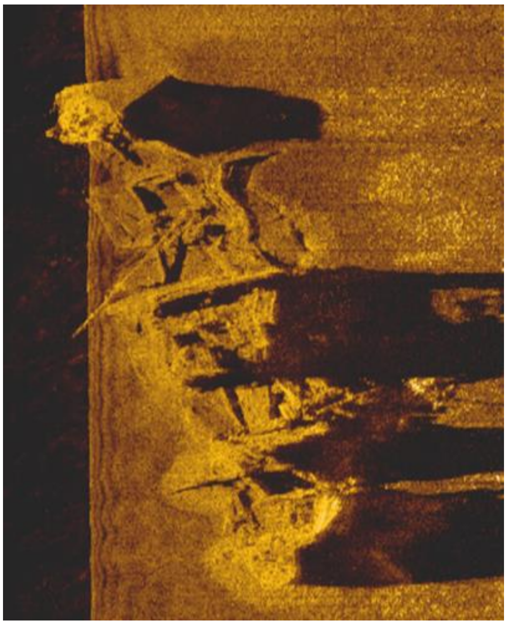


Figure 1: *Submerged vessel on side scan sonar image*

The assessment of the SSS image is usually given visually by the sonar operator, who determines the class of the object. For a number of tasks with a large amount of information it is expedient to recognize images in automatic mode, however, despite the wide application of pattern recognition methods in various fields, the issue with underwater potentially dangerous objects has not been solved.

The problem of reliability of classification or recognition of PHUO is complicated by a variety of types and sizes of objects on the bottom, as well as their location relative to the line of motion of the ship-tugboat SSS, i.e. the same object at different distances from the ship-tugboat of SSS and at different angles of sighting can have a completely different image (picture) on the SSS screen [6]. It is especially noticeable on extended objects (boats, logs, explosive ordnance, etc.). Therefore, the task has to be solved in two stages.

At the first stage, the method of selecting a single object from the data stream from the side-scan sonar output is used, or in other words, the detection of "candidates" for PHUO and their initial selection by object size ("item" for PCOR should be at least 0.5 meters and not more than 100 meters in size). The image at the SSS output allows to do this, since the resolution of sensors (image pixel sizes) is currently not more than 0.2 meters. The results of the detection algorithm are transmitted to classification in the form of image fragments containing "items" for PHUO. We will call them strobes containing images of smaller volume than the original image from the SSS output.

In the second stage, classification of objects in each strobe obtained in the algorithm for detecting "items" in PHUO is performed.

Thus, we can term primary processing as object detection and strobe selection, and secondary processing as object classification in a strobe.

The SSS output produces an image of the underwater situation to the right and left of the SSS tugboat line of motion, which is processed in the pre-processing block. At the output of the pre-processing block two images are obtained (to the right and to the left of the line of motion of the ship-tugboat), in which linear distortions are eliminated and the overlapping part of the frame, which is under the ship-tugboat, is removed. Further the right and left parts of the frame are processed independently.

All lines are processed according to the same algorithm in order to detect the boundaries of a potential object and its acoustic shadow in each line. For this purpose, the signal in the string is "passed" through a linear matched filter, and the filtered signal is compared with two thresholds: exceeding the upper threshold determines the left and right boundaries of the object in this string, and the value of the filtered signal less than the lower threshold determines the left and right boundaries of the acoustic shadow of the object in this string.

After processing of all frame lines, "stitching" of potential object boundaries between the nearest lines is performed according to the "nearest neighbor" principle. The "stitching" of boundaries continues until there is no (will not be found) continuation of the potential object boundary in two neighboring lines. After that, the lower left and upper right boundaries of the potential object are calculated, and these parameters are used to calculate the strobe coordinates (for further selection of a strobe containing signals from the potential object from the frame, taking into account its potential boundaries in the frame).

The selected strobe is then passed to the classification algorithm.

Two approaches can be used to classify the selected underwater potentially hazardous object:

- Measurement and selection of classification features with subsequent analysis and decision making;

- recognizing the image of an underwater potentially hazardous object as given classification type, formed on the basis of accumulated previous data.

In the first approach, the source of initial information is sonar, which, using spatial and temporal processing, generates a signal space belonging to all detected targets. A priori, it is not known which targets and at what distances are located in the processing space. From the output of the system of spatial and temporal processing all signals arrive in the block of measurement of classification features, in which the classification features are selected and converted into a useful form for further analysis. It is necessary to ensure measurement accuracy and resolution in time and space for this process.

The results of this transformation give an n - dimensional vector, which is the code of the selected class. This code has a random character, because the signs are measured in conditions of interference and errors of the measurement system, caused by the error of its own motion, as well as the randomness of the appearance of a particular representative of a particular class of objects. From the output of the block of measurement of signs the code of the object comes to the block of decision making, which also receives the codes of standards of all classes.

From output of the decision making unit provides quantitative indicators characterizing the proximity of the presented realization with the reference. The result of comparison is, as a rule, a posteriori probability or correlation coefficient. In the decision making block the object is compared according to the specified criterion and attributed to one of the standards. The database is designed to store the attributes of reference classes and the results obtained at various stages of recognition. The attributes of reference classes are put into it on the basis of initial a priori data.

The system of classification attributes should be previously known and a spatio-temporal processing should be developed for it.

It is possible to go another way: using the existing spatio-temporal processing, analyze the results obtained and on their basis formulate a set of classification features. In the first case, when the set of classification features is known, it is possible to form automatic decision making procedures. In the second case, the decision, as a rule, should be made by the operator on the basis of preliminary processing of initial information.

It is the first case when a classification system, in which algorithms of automatic measurement of classification features and algorithms of automatic decision making were implemented, was developed in relation to a small-size environment illumination complex.

The complex was installed on a specific stationary carrier and had the function of ensuring the safety of navigation. The classification system contained a set of classification features that were automatically measured when a target was detected. Based on the developed classification attributes, an automatic decision on the target class was formed for the generated full group of events. The operator made an independent decision based on the type of information displayed and the measured classification features.

In this paper, the classification of potential hazardous underwater object images in software is proposed to be performed using convolutional neuron network algorithms.

The method of neural networks is based on a simple mathematical model of brain functioning. Neural networks can be used to obtain both numerical and binary solutions. The method is computationally expensive, but the structure of the networks allows capturing (accounting for) complex relationships between predictors and values of the dependent variable. On the other hand, the network can be perceived as a "black box" because the identified relationships are not easy to understand and interpret. Thus, the neural network method is a good choice when accurate forecasts are required, but it is unsuccessful when solving descriptive analysis problems or when it is necessary to investigate the nature of relationships between variables. Forecasts obtained with the help of neural networks can be combined with other forecasts to obtain prognostic "ensembles" [7, 8].

The schematic of an artificial neuron (as a mathematical reflection of a physical neuron in the human brain) can be visualized as shown below in figure 2.

As with a biological neuron, an artificial neuron takes input values $x_1, x_2, \dots, x_n$, multiplies them with weights $w_1, w_2, \dots, w_n$, sums the resulting values (producing the "logit" of the neuron, ($z = \sum_i^n w_i x_i$) and passes them to a function f that calculates the neuron's output $y = f(z)$. The value of y can then be passed to other neurons as input.

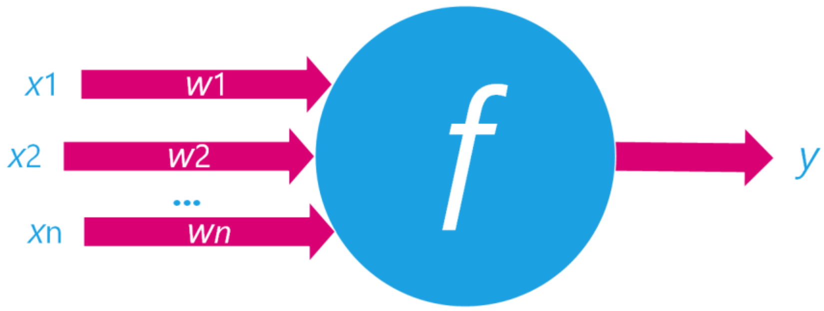


Figure 2: *Scheme of a neuron in an artificial neural network*

In the human brain, neurons are organized into layers. An artificial neural network arises when we connect artificial neurons into layers and link the layers with outputs and inputs, with the first layer receiving as input the raw data of a problem and the last layer producing as output the solution to it (Figure 3).

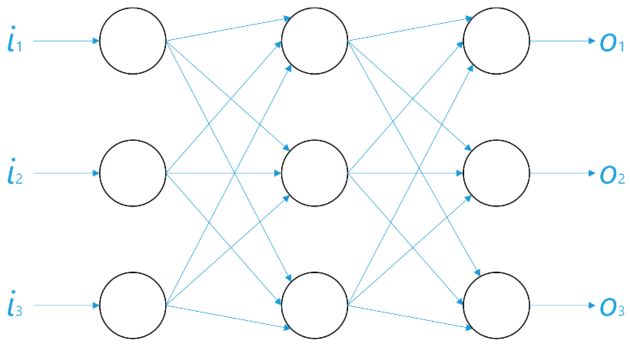


Figure 3: *A simple example of a neural network with three layers*

The logit transformation functions of the neurons in each layer can be different: linear in the simplest case ($f(z) = az + b$), sigmoidal, tangent, or boundedly linear (respectively, $f(z) = \frac{1}{1+e^{-z}}$), $f(z) = \tanh(z)$, and $f(z) = \max(0; z)$ and others. The simplest example of a neural network is a network without hidden layers (there are only input and output layers), equivalent to linear regression. The prediction (solution to the problem) is obtained as a linear combination of inputs. The weights are chosen using a learning algorithm that minimizes a cost function (e.g., MSE - mean squared error). Of course, in such a simple case, there is no sense in a neural network at all, because using a linear regression method directly is much more efficient [9].

In our case, a more complex network will be used; the input of the lower layer of neurons will be image data (pixel intensity values) of underwater objects, and the outputs of the upper layer of neurons will give us probabilities of assigning objects in the images to different classes. The neural network converts image pixel values into a useful representation by extracting high-level image features (such as shapes and edges in the image) that describe complex concepts by combining a large number of small pieces of information [10, 11, 1].

In the task of image classification, it is convenient to obtain the probability distribution of assigning an object in the image to a particular class at the output of a neural network, so in this kind of tasks, a softmax function is used on the output layer of neurons. The radical difference of the function is that its calculation requires not only the logit value of a given neuron, but also the logit values of all other neurons in the layer (1):

$$y_i = \frac{e^{z_i}}{\sum_j e^{z_j}} \tag{1}$$

In the case of a "good", "strong" prediction, the output of one of the neurons will be significantly greater than the outputs of the others, in other words, ideally, the object should be assigned to a particular class with a fair amount of confidence.

Initially, all connection weights in the network are initialized in some random way. During the training phase, images of objects whose classes are known (labeled) are loaded into the model. By comparing the network answers y with the known correct answers t one can obtain the error function E, in particular, in the form of a second-order norm (2):

$$E = \frac{1}{2}\sum_i \left(t^{(i)} - y^{(i)}\right)^2 \qquad (2)$$

When E is closer to zero, the more accurate the predictions of our model are, and if E=0, then our model performs predictions with perfect accuracy.

The accuracy of the model during the training phase is improved by applying optimization techniques to the error function E. More precisely, we need to find the minimum value of E as a function of, among other things, the weights of the inputs to the last layer of neurons. Using gradient descent (or another optimization method), we can adjust the weights for the inputs of the last layer. On the other hand, the value of the error function depends not only on the weights of connections in the last layer of neurons, but also on the values of inputs to it. Those, in turn, depend on the weights on the layer before the last one, etc. By spreading the "responsibility" for the magnitude of the error to all layers, we can adjust all the weights in the network, thus training the model. This approach is called the "error back propagation method" [13, 14, 15].

A significant problem arises in image processing by classical neural networks. Because the input image is divided into pixels, it means that the number of inputs (and thus weights) to each neuron of the lower layer of the network will be:

number of pixels in height * number of pixels in width * number of color components, e.g. 200*200*3=120,000.

This number of connections is not only redundant, consumes a large amount of memory and slows down the learning process, it is also likely to overtrain the network with respect to the training data.

A convolutional neural network takes advantage of the knowledge that it is images that we are processing, which means that we can intelligently constrain the network architecture in a way that significantly reduces the number of incoming connections.

The basis of a convolutional neural network is the use of filter layers and max pooling layers.

The filtering layer processes a three-dimensional array of information (in our case, the dimensions of such an array are determined by the length and width of the image, as well as the number of color components in it) and outputs another three-dimensional array of information, with length and width determined by the parameters of the filters, and depth determined by the number of applied filters (figure. 4).

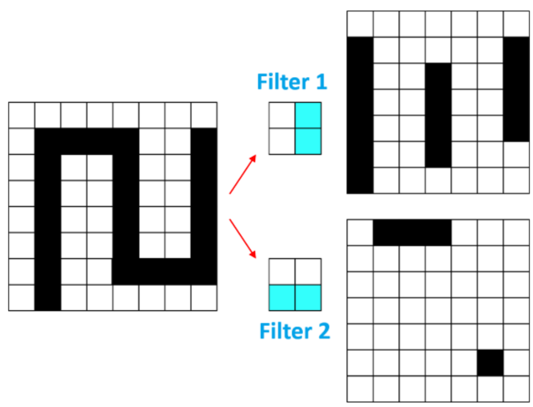


Figure 4: *Application of two filters that select horizontal and vertical lines*

The unifying layer is responsible for reducing the size of the information array (and, as a consequence, the number of inputs to the next layer) by splitting the information array into equal-sized fragments and condensing the values of individual elements in each fragment into a single value (figure 5) [13, 14].

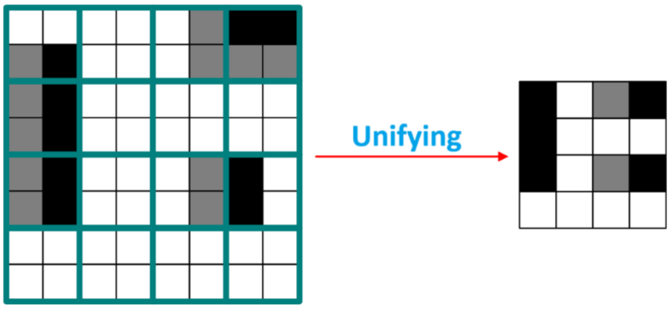


Figure 5: *An example of significant reduction of the number of neural network parameters by applying th When solving the problem of object classification in general terms, a neural algorithm is a computational procedure, the main part of which can be realized on a neural network.*

III. Results and discussion

The basis for the development of a neural algorithm for solving the problem is a systematic approach, in which the process of solving the problem is represented as the functioning in time of some dynamic system, whose input is a set D (initial data), and the output is a set R (objects to be defined and received their values).

Solving the problem with the help of neural algorithm includes the following points:

1) Obtaining a specific neural network structure corresponding to the applied algorithm.

2) Finding the values of weight coefficients, or selecting them from memory, if they were found earlier.

3) Generation of initial approximations of parameters, if it is necessary.

4) Transfer of all numerical values to the neural network and running it.

5) Functioning of the network according to the mode:

- over a single step or a fixed number of steps;

- for a variable number of steps, depending on the required accuracy and/or specific numerical values of the parameters (in it, the process of tuning the input signal takes place).

6) Obtaining the solution.

Points 1 and 2 may be performed once for subsequent repeated application of points 3, 4, 5 and 6.e unification.

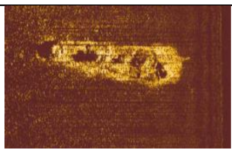
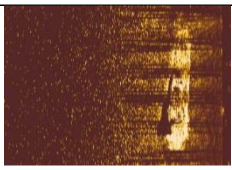
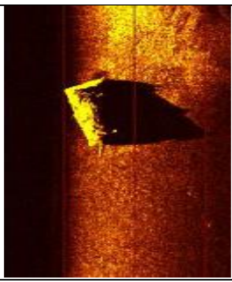
In the case of applying the approach based on a neural network to the task of classification of images of underwater objects, the above means that at the stage of training in the neural network with the selected topology loaded pre-arranged data, that is, data directly images and pre-assigned to these images classes of objects, corrected network parameters, and then at the stage of work in the network loaded one by one already unarranged images, for make a decision on the assigned class at the output of the neural network.

Modern image recognition and classification models may include millions of parameters. Training them from scratch requires large amounts of training marked data (millions of images) and considerable time (hundreds of hours of GPU work - graphics processing units on video cards used for parallel computations, including those associated with neural networks).

Knowledge transfer is a technique that takes a part of a model already trained (trained) on a similar task and uses it in a new model for a different task. This approach does not provide the accuracy that can be obtained by training a full model, but it shows itself to be surprisingly good on multiple tasks, also allowing thousands rather than millions of training images to be used, and models to be trained in half an hour or an hour rather than hundreds of hours. This approach applied in this paper [16, 17, 18, 19].

After training the network on a large database, 100 images with objects were selected/synthesized to test the accuracy of the recognition program. It was not previously encountered by the neural network, i.e. it was not used in its training. Selected test results are given in the table below.

Table 1: *Results of testing the underwater object classification program on side scan sonar images (Letter designations correspond to the following categories: A - ship/submarine, B - containers, C - barrels, D - ammunition, E - other)*

Test number	Image	The probability of categorizing the object in the image, %				
		A	B	C	D	E
13		83,36	3,78	1,43	9,27	2,16
14		94,57	1,26	1,53	1,22	1,41
26		10,95	10,68	0,76	2,12	75,49

According to the results of the final testing, the program showed the correctness of category determination for 100 new images at the level of 91%.

The developed method can be used to improve the reliability of classification of potentially hazardous underwater objects, both directly at the side scan sonar output and during post-processing of the accumulated records.

References

[1] Vjalyshev A.I. (2017) Emercom of Russia and Potentially Hazardous Underwater Objects [MChS Rossii i podvodnye potencial'no opasnye obekty]. Civil security Technology. Vol. 14 № 1(51) p 4-10. (in Russian).

[2] Du, X., Sun, Y., Song, Y., Sun, H., & Yang, L. (2023). A comparative study of different CNN models and transfer learning effect for underwater object classification in side-scan sonar images. Remote Sensing, 15(3), 593.

[3] Yulin, T., Jin, S., Bian, G., & Zhang, Y. (2020). Shipwreck target recognition in side-scan sonar images by improved YOLOv3 model based on transfer learning. IEEE Access, 8, 173450-173460.

[4] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14 (pp. 21-37). Springer International Publishing.

[5] Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE transactions on pattern analysis and machine intelligence, 39(6), 1137-1149.

[6] Merrifield, S. T., Celona, S., McCarthy, R. A., Pietruszka, A., Batchelor, H., Hess, R., ... & Terrill, E. J. (2023). Wide-area debris field and seabed characterization of a deep ocean dump site surveyed by Autonomous Underwater Vehicles. Environmental Science & Technology, 57(46), 18162-18171.

[7] Shmueli Galit, Lichtendahl Jr. Kenneth C. (2016) Practical Time Series Forecasting with R. Axelrod Schnall Publishers.

[8] Woodward Wayne A., Sadler Bivin Philip, Robertson Stephen. (2024) Time Series for Data Science. CRC Press.

[9] Hyndman Rob J., Athanasopoulos George. (2021) Forecasting. Principles and Practice. 3rd ed. OTexts.com.

[10] Hope Tom, Resheff Yehezkel S., Lieder Itay. (2017) Learning Tensor Flow. O'Reilly.

[11] Géron Aurélien. (2022) Hands-On Machine Learning with Scikit-Learn, Keras, and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems. 3rd ed. O'Reilly.

[12] Chollet Francois, Kalinowski Tomasz, Allaire J. J. (2022) Deep Learning with R. 2nd ed. Manning.

[13] Galushkin A.I. (2010) Neural networks: fundamentals of theory [Nejronnye seti: osnovy teorii]. Hotline-Telecom Publishing House. (in Russian).

[14] Hagan Martin T., Demuth Howard B., Beale Mark H., De Jesús Orlando. (2014) Neural Network Design. (Martin hagan) 2nd ed.

[15] Nikhil Buduma. (2017). Fundamentals of Deep Learning: Designing Next-generation Artificial Intelligence Algorithms/c Nikhil Buduma. Beijing, Boston, Farnham, Sebastopol, Tokyo: OReilly.

[16] Jiang, J., Shu, Y., Wang, J., & Long, M. (2022). Transferability in deep learning: A survey. arXiv preprint arXiv:2201.05867.

[17] Gutstein, S., Fuentes, O., & Freudenthal, E. (2008). Knowledge transfer in deep convolutional neural nets. International Journal on Artificial Intelligence Tools, 17(03), 555-567.

[18] Passalis, N., Tzelepi, M., & Tefas, A. (2020). Probabilistic knowledge transfer for lightweight deep representation learning. IEEE Transactions on Neural Networks and Learning Systems, 32(5), 2030-2039.

[19] Chen, T., Goodfellow, I., & Shlens, J. (2015). Net2net: Accelerating learning via knowledge transfer. *arXiv preprint arXiv:1511.05641*.