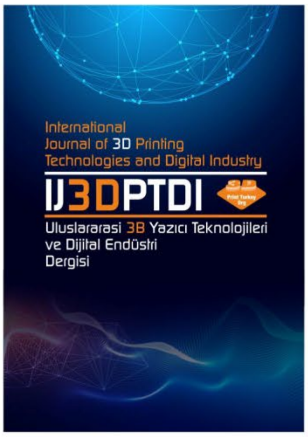

ULUSLARARASI 3B YAZICI TEKNOLOJİLERİ
VE DİJİTAL ENDÜSTRİ DERGİSİ

INTERNATIONAL JOURNAL OF 3D PRINTING
TECHNOLOGIES AND DIGITAL INDUSTRY

ISSN:2602-3350 (Online)

URL: https://dergipark.org.tr/ij3dptdi

ANALYSIS OF DIFFERENT POOLING FUNCTIONS ON A CONVOLUTION NEURAL NETWORK BASED MODEL

Yazarlar (Authors): Halit Çetiner*, Sedat Metlek

Bu makaleye şu şekilde atıfta bulunabilirsiniz (To cite to this article): *Çetiner H., Metlek S., "Analysis of Different Pooling Functions on A Convolution Neural Network Based Model" Int. J. of 3D Printing Tech. Dig. Ind., 8(2): 266-276, (2024).*

DOI: 10.46519/ij3dptdi.1484354

Araştırma Makale/ Research Article

Erişim Linki: (To link to this article): https://dergipark.org.tr/en/pub/ij3dptdi/archive

ANALYSIS OF DIFFERENT POOLING FUNCTIONS ON A CONVOLUTION NEURAL NETWORK BASED MODEL

Halit Çetiner[a]*, Sedat Metlek[b]

[a]Isparta Applied Sciences of University, Vocational School of Technical Sciences, Computer Technology, TÜRKİYE
[b]Burdur Mehmet Akif Ersoy University, Vocational School of Technical Sciences, Mechatronic Program, TÜRKİYE

* *Corresponding Author:* halitcetiner@isparta.edu.tr

(**Received:** 15.05.2024; **Revised:** 13.08.2024; **Accepted:** 14.08.2024)

ABSTRACT

The common denominator of deep learning models used in many different fields today is the pooling functions used in their internal architecture. These functions not only directly affect the performance of the study, but also directly affect the training time. For this reason, it is extremely important to measure the performance of different pooling functions and share their success values. In this study, the performances of commonly used soft pooling, max pooling, spatial pyramid pooling and average pooling functions were measured on a dataset used as benchmarking in the literature. For this purpose, a new CNN based architecture was developed. Accuracy, F1 score, precision, recall and categorical cross entropy metrics used in many studies in the literature were used to measure the performance of the developed architecture. As a result of the performance metrics obtained, 97.79, 92.50, 91.60 and 89.09 values from best to worst for accuracy were obtained from soft pooling, max pooling, spatial pyramid pooling and average pooling functions, respectively. In the light of these results, the pooling functions used in this study have provided a better conceptual and comparative understanding of the impact of a CNN-based model.

Keywords: Pooling, Artificial Intelligence, Convolution Neural Network, Classification.

1. INTRODUCTION

Convolutional neural networks (CNN) are used in many artificial intelligence algorithms, especially image classification and segmentation [1-3]. Image classification applications using CNN architectures are one of today's important research topics [4-7]. The underlying problem of this research topic is that CNN architectures are high-cost algorithms. For this reason, it is aimed that newly developed CNN architectures will be advantageous in terms of time, cost and complexity, especially in image classification applications. At the same time, it is critical that these algorithms are competitive with their competitors in terms of performance. For this reason, many researchers working in the field of artificial intelligence are trying to develop new algorithms for image classification applications. They often aim to solve these problems by using CNN layers in different combinations. However, new solutions to these problems can be brought from a different perspective by focusing on layer structures that are commonly used in many architectures. For this reason, the study focused on the pooling layer. Generally, two types of pooling are used in CNN architectures: local and global. In the local pooling method, feature maps are obtained from local regions in window size with the help of windows hovering over the images. The second type of pooling, the global pooling type, is a pooling that creates a scalar value for each feature in the feature map. Pooling, which has a non-linear process, collects the outputs in layers by reducing them [8].One of the most important features that distinguishes the pooling layer from other layers is that it reduces input sizes to minimize memory consumption in order to maintain statistical performance [9-11]. Pooling layer is used to obtain semantic information and reduce the spatial resolution of feature maps, known as subsampling [12]. When performing subsampling, maximum pooling preserves the most distinctive, distinct features in the feature map, while average pooling creates a smooth transition effect [12]. Pooling also partially solves the overfitting problem, which is a significant disadvantage in deep learning models.

CNN-based architectures generally consist of multiple convolutional layers to extract distinctive features and subsequent layers such as Batch Normalization, Pooling, and Fully connected. In the Pool-SH model proposed in this article, while the layers other than the pooling layer remain constant, the max pooling, average pooling, soft pooling and spatial pyramid pooling methods are used separately and compared in the pooling layer. The main purposes of pooling layers that form the architectures in deep learning are to learn features despite changes such as subsampling feature maps and scaling and rotation [4]. Pooling reduces computational complexity and memory requirements by reducing the feature map size while preserving important features. In the Pool-SH model proposed in this article, it has been proven on a benchmark dataset which of the max pooling, average pooling, soft pooling, spatial pyramid pooling methods will give better performance values. For this purpose, the natural image data set, which is frequently used in the literature, was used.

The main contributions of the study to the literature are presented below.

- The proposed Pool-SH model provides a structure that can compare popular pooling functions that operate in very critical tasks such as computational cost, complexity and data size reduction.
- The proposed Pool-SH model tries to compete with its unique structure consisting of 16 layers instead of the high-weight structures of transfer learning-based architectures.
- The soft pooling function outperforms the other pooling functions on the proposed Pool-SH model.
- As a result of the performance results obtained on the proposed Pool-SH model, values were obtained from soft pooling, max pooling, spatial pyramid pooling and average pooling functions, from highest to lowest.

The rest of the study consists of 4 sections. Section 2 presents the related work in the literature. Section 3 introduces the materials and methods used. Section 4 presents the performance results obtained from the experimental studies. In the last section, the findings obtained are evaluated in general.

2. RELATED WORKS

In this chapter, A literature review was conducted to cover maximum pooling and average pooling, as well as soft pooling and spatial pyramid pooling methods, which are frequently mentioned in the literature. At the same time, studies using natural images [13], which is the benchmarking dataset used in this article, are also analyzed in this section.

The pooling layer is a layer that reduces the feature map from the previous convolution layer to smaller sizes, which is often used in CNN and transfer learning based architectures. In CNN and transfer learning based architectures, the pooling layers used are among the important factors affecting the performance of the model. Pooling layers greatly reduce the computational cost and learning process of the model by reducing the spatial dimension of the model in transfer learning and CNN models. Among the most widely used pooling layers in the literature are maximum and average pooling layers [14]. The main shortcoming of the maximum pooling layer is that it only takes the largest value in the area where it is used, and therefore ignores other values. The main shortcoming of the average pooling method is that it takes the average of the values in the area where it is applied. Thus, the minimum and maximum values, which are extreme values, are ignored. Due to these disadvantages, there are many studies in the literature where these methods are used and tested [4,8]. A brief analysis of the applications that use the Natural images dataset as a dataset in their work is also shared below.

Dogo et al. [15] compared Adamax, AdaDelta, Nadam, SGD, vSGD, Adam, SGDm, RMSProp and SGDm+n methods, which are stochastic gradient-based optimization techniques frequently used in CNN-based architecture setup. They obtained performance graphs by training the architectural models used on the benchmark dataset used in the article with the relevant optimization technique. As a result of the performance results, it is stated that the

Nadam optimization technique gives better results than other optimization techniques.

Sikandar et al. [16] developed a hybrid machine learning method consisting of ResNet50, VGG16 and KNN algorithms. In the method they developed, first the features from the ResNet50 and VGG16 methods are given as input to the GlobalMaxPooling2D layer and converted into a one-dimensional array. Secondly, the features converted into a one-dimensional array are clustered by determining the Euclidean distances with the KNN method. After the clustering process, the distance between image clusters was determined. Prabavathi and Sakthi [17], carry out a new study, different from the studies in the literature, to obtain a higher compression ratio. First, the noise of the image is removed. It then performed image compression to achieve storage efficiency and transmission.

Praveenkumar and Nagaraj developed a new model consisting of many layers of nodes in deep neural networks. They aimed to increase classification performance and reduce training time with the model they developed [18]. A brief summary of studies using maximum and average pooling methods commonly used in the literature is presented below.

Özdemir et al. [14] takes the average of the K number of highest pixels inspired by the maximum and average pooling layers. It is stated that the Avg-TopK method, which is the pooling method they developed, gives better results in transfer learning-based models. Within the scope of the study, the effects of maximum, average and Avg-TopK methods were tried to be measured not only on color images but also on gray images. The Avg-TopK method is reported to be better than other classical pooling methods in terms of computational cost, speed and performance.

Muhammed et al. [19] implemented designs for a block called vector pooling block for the pooling layer, which is not widely studied in the literature. The developed pooling method consists of two data paths focusing on the extraction of features on vertical and horizontal paths. Here, instead of collecting features using a fixed square filter, CNN architectures can collect both local and global features by using long and narrow filters.

Vigneron et al. [20] developed a new pooling method based on Zeckendorf's number series. It is stated that their newly developed Z pooling layer is better adapted to partitioning tasks than other pooling methods.

Sharma et al. [21], tried to improve the performance of CNN architecture by trying a hybrid pooling method for image classification. The hybrid pooling method they developed is a method that can be thought of as a mixture of fuzzy and maximum pooling with the help of using pixel intensity values such as maximum and average pooling. The fuzzy and maximum pooling layer is combined with the learning parameter α. The study tries to prove that the hybrid pooling method they developed is better than the traditional pooling method in their performance.

Bhattacharjee et al. [22] designs a trainable pooling process that determines the instance-to-bag relationship based on the genetic algorithm. It is reported that the initialization of random weights is achieved by optimizing the attention weights thanks to the genetic algorithm.

3. MATERIAL AND METHODS

Deep learning methods are current algorithms developed especially for problems that cannot be solved with a certain formulation. One of the basic building blocks of these algorithms is pooling methods. Pooling methods are special methods developed to use a certain fraction of many features. Therefore, pooling methods improve the learning performance of CNN-based models by reducing computational complexity [4]. Pooling methods are used for different purposes such as reducing overfitting, capturing high-level information between features, and increasing the impact of the most important features. For this reason, soft pooling, SPM, max pooling, average pooling and max pooling methods, which are frequently used in the literature, are used in this study. SPM method is preferred because it extends the fully connected layer feed with multi-level pooling to alleviate the shortcomings of traditional pooling methods.

The main reason for choosing the soft pooling method is that it aims to smooth the maximum activation values in the kernel values used in deep learning methods. Max pooling method is chosen because it is simpler and more understandable than many pooling functions. The average pooling method is generally similar to the max pooling method. The difference is that if the values in the region where the pooling method is applied are zero or close to zero, it presents values close to zero in the output. As a result, dominant features may be lost. In this study, the average pooling method was preferred to examine the effect of this disadvantage on the situation.

3.1. Materials

The natural image dataset used within the scope of the article consists of 727 airplanes, 968 cars, 885 cats, 702 dogs, 843 flowers, 1000 fruits, 788 motorcycles, and 986 person images. The dataset consists of a total of 6899 images belonging to 8 different classes: airplane, dog, flower, fruit, car, cat, motorbike, person [13].

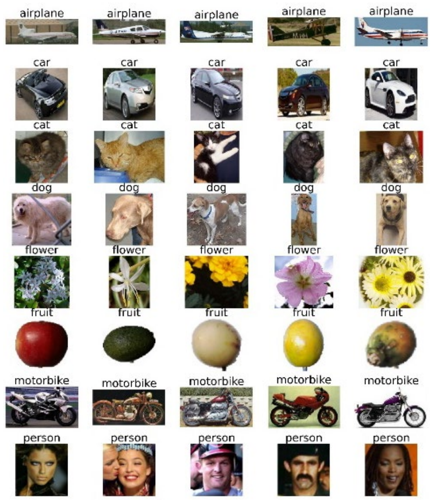

Figure 1. Sample dataset

Figure 1 shows image examples of each class in the dataset.

3.2. Max Pooling Method

In many CNN models in the literature, the maximum pooling method is preferred. The main reason for this is that its structure is simpler and more understandable than many pooling functions. Max pooling is based on the largest value within the kxk neighborhood when optimizing the spatial size of a feature map [23-25]. The general structure of max pooling is presented in Eq. 1.

$$f_{max}(x) = max\{x_i\}_{i=0}^{N} \quad (1)$$

The expression x in Eq. 1 refers to the pooling region in the input image. Given sparse codes and simple linear classifiers, max pooling performs better [14]. The disadvantage of the max pooling function is that it takes the largest value in the relevant region and ignores other values. For this reason, in some cases, distinctive features may be lost. As a result, the performance of applications may be negatively affected.

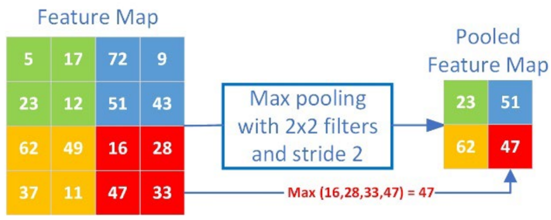

Figure 2. Max pooling

The situation in question can be seen in Figure 2. In the feature map in Figure2, the stride value was selected as 1 and a 2x2 pooling function was applied. As can be seen from here, only the largest values in the relevant area were taken. This may cause unacceptable results in some applications.

3.3. Average Pooling Method

As can be seen in Figure 3, the average pooling method takes the average of the values in the pooling region. This situation is mathematically illustrated by Eq. 2.

$$f_{avg}(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \quad (2)$$

While the x_i value in Eq. 2 shows the data in the area where the pooling process is applied, the N value represents the total number of these data. In addition, the general working principle of the Average pooling method is similar to the max pooling method. The disadvantage is that if the values in the region where the pooling method is applied are zero or close to zero, it presents zero or close to zero values in parallel with these values at the output. As a result, dominant features may be lost.

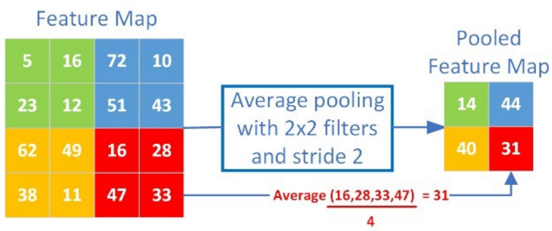

Figure 3. Average pooling

3.4. Soft Pooling Method

The general logic of the soft pooling method developed by Riesenhuber and Poggio [26] is based on the natural exponent (e), which ensures that larger activation values have a greater effect on the output. This can be done with a gradient value proportional to at least the minimum value during backpropagation to all activations within the local core neighborhood [27]. As a result, this process is the opposite of the methods presented in Sections 3.2 and 3.3. The soft pooling method aims to soften maximum activation approaches within the kernel region. For this, Eq. 3 is used first.

$$w_i = \frac{e^{a_i}}{\sum_{j \in \mathbf{R}} e^{a_j}} \tag{3}$$

While the weight applied to the ith index in Eq. 3 is expressed with w_i, the applied activation is expressed with a_i. Nonlinear transformations can be performed with weights corresponding to the activation values. Thus, higher activations are made more dominant than lower ones. Since most pooling operations are performed in high-dimensional feature spaces, highlighting activations with higher impact is a more logical approach than simply selecting the mean or maximum value [27]. While in the max pooling and average pooling approaches, discarding some information means discarding important features, in this approach, the equal contribution of activations may correspond to local density reductions by taking the overall regional feature density equally into account. In the soft pooling method, the output value is generated via a standard sum of all weighted activations within the R core neighborhood as in Eq. 4.

$$\tilde{a} = \sum_{i \in \mathbf{R}} w_i * a_i \tag{4}$$

Compared to traditional maximum and average pooling methods, using softmax of regions of interest allows each activation to be normalized relative to neighboring activations for the core region with a probability distribution proportional to their values. This is in contrast to the popular choice of maximum or average value, or averaging all activations over the core region, where the output activations are not regularized [27].

3.5. Spatial Pyramid Matching Method

Spatial pyramid matching (SPM) method is a new pooling method that eliminates the need for fixed-size images in CNNs. This method is applied as fixed-size constraints to fully connected layers instead of convolution layers. In general, before pyramid pooling functions, it was necessary to crop and warp images in order to fit the images into the dimensions in the CNN layers. However, operations such as cropping and warping could lead to content loss and geometric distortions [28,29]. SPM, a popular pooling method today, is designed to match the size of feature maps. The sizes of the contents may vary. For example, let's say you have an image of size 128x128. If the four container number is used under this image, a patch of 32x32 dimensions can be created. In this way, a total of 16 boxes (thousand) are formed. The highest value in each box (bin) is considered the activation value of the next level of the pyramid. As a result, the SPM technique can produce a fixed-length output without taking into account the size of the input. Moreover, it allows adaptation to input image scales during the testing and training phases of SPM, which strengthens the scale invariance feature and eliminates the problem of overfitting in the network [30]. The SPM method is primarily designed to deal with images of variable size. It also has a more complex learning procedure. As a result, it is sometimes less efficient. For example, in the CIFAR10 dataset, it caused an error of 16:89 percent [31].

3.6. Proposed Model (Pool-SH)

Pool-SH, a new CNN-based model, was proposed to analyze the pooling functions used in the study. The Pool-SH model aims to use layers similar to those commonly used in CNN-based architectures.

Thus, a more realistic comparison with the models in the literature was enabled. In the input layer of the model developed in the study, first thresholding is applied to the image, and in the cabin the images are resized to 240x240. These dimensions are again very close to the dimensions used in many transfers learning models in the literature. In the layers following the entry layer, convolution and pooling operations are performed sequentially.

Although ReLU activation functions were used in the convolution layers used in the study, the dimensions of the tested pooling layers were determined as 2x2. This situation is shown in detail in Figure4. Flatten and Fully Connected Layer, which have become standards in many deep learning models in the literature, were added to the model proposed in the study. In the last stage of the proposed model, the classification layer was added. Since there are 8 outputs in the added classification layer, the softmax activation function was used. The content of the softmax activation function is also presented in Eq. 5.

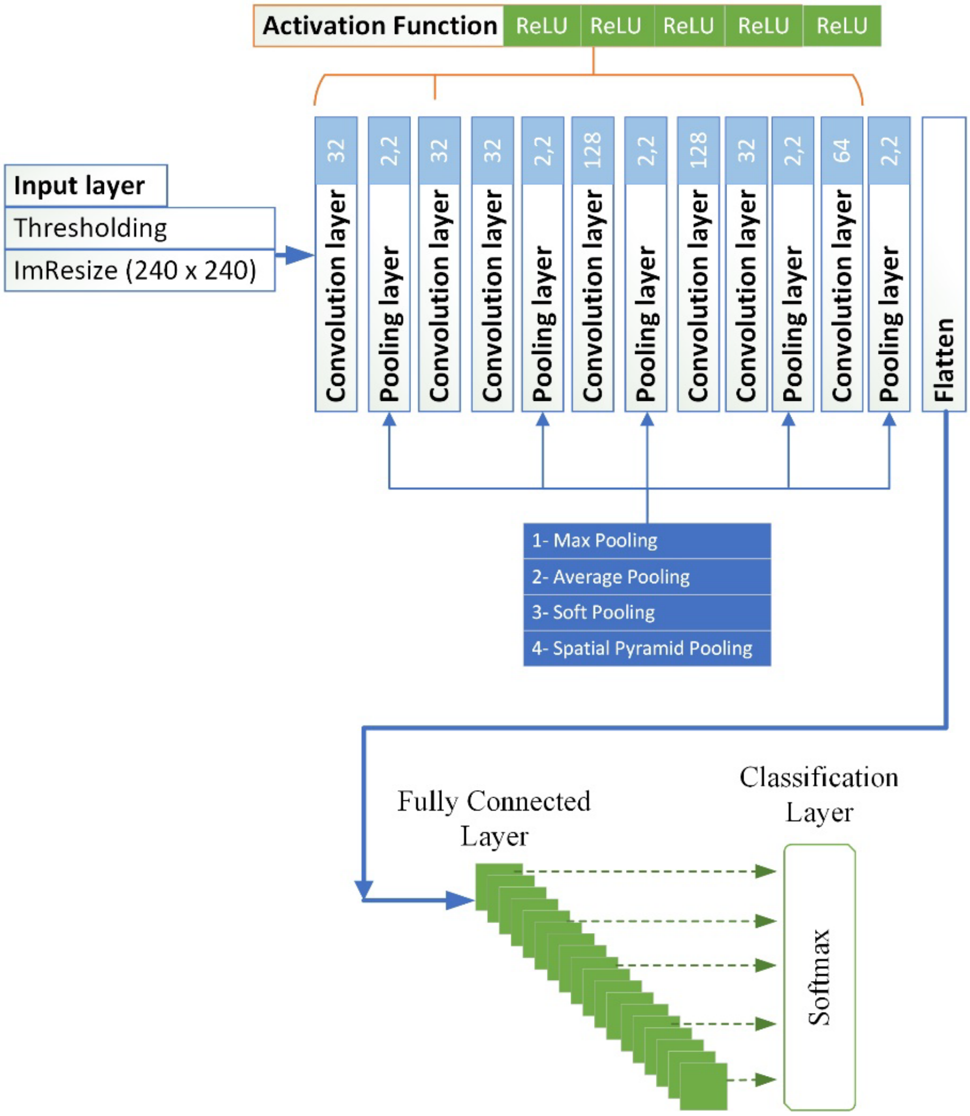

Figure 4. Proposed Pool-SH model.

$$softmax(z_j) = \frac{e^{z_j}}{\sum_{j=1}^{N} e^{z_j}} \quad for\ j = 1, \ldots, N \tag{5}$$

The softmax function is a variant of the sigmoid function. Although the term N used in Eq. 5 refers to N classes, the softmax function allows the calculation of which class each output of these classes belongs to by adding their exponential values. The z_j value in Eq. 5 represents the jth value in the classification layer. The model designed in this way consists of 16 layers in total and includes four pooling layers. Soft pooling, max pooling, spatial pyramid pooling and average pooling functions were used in these pooling layers, respectively. In this case it is shown in detail in Figure 4.

3.7. Execution of the Pooling Methods

In the execution of pooling methods, the max pooling method was preferred first. In this method, using only the largest number in the filter used provides ease of operation and has an effective role in highlighting only the dominant features on the image. However, local features in the data are lost. In the average pooling method, sharp features are lost and more localized images can be obtained. In parallel with the literature, it was found that soft pooling gives higher performance than max pooling and average pooling when the size of the features representing the class is smaller than the image size [8]. SPM is a pooling method that provides multi-level input by removing the fixed constraints associated with the fully connected layer. Due to its inherent computational complexity, it is found to be less efficient than other methods, in accordance with the literature [31].

If it is necessary to evaluate the pooling methods used in terms of computational complexity, SPM, soft pooling, average pooling, and max pooling methods are found to be the highest to the lowest. It is noteworthy that while the computational complexity of average pooling and max pooling methods are close to each other, it is seen that the SPM method offers a considerably higher computational complexity compared to other methods.

4. EXPERIMENTAL RESULTS AND DISCUSSION

The 6899 images in the natural image dataset used in the study were divided into two groups: training and validation, according to the cross-validation value of 5. While the number of images in the test group was determined as 1380, the number of images in the training group was determined as 5519. In the study, experimental studies were carried out on a computer with a 64-bit operating system with NVIDIA RTX 3060 graphics card, AMD Ryzen 7 5800H branded processor with a capacity of 3.2 GHz, 16 GB RAM and hardware features. Training the system takes approximately 48 minutes. For performance measurement in training and testing, we used the commonly used metrics of accuracy, F1 score, precision, recall and categorical cross entropy (CELoss), which are presented in Eqs. 6-10 respectively [32-36,37].

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

$$F1\ Score = \frac{2 * Recall * Precision}{Recall + Precision} \tag{9}$$

$$CE_{Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C}(y_{ic}\log(\widehat{y_{ic}})) \tag{10}$$

The terms TP, TN, FP and FN used in Eqs. 6-8 refer to true-positive, true-negative, false-positive and false-negative respectively. The training results obtained with these metrics are detailed in Table 1. In Eq. 10, $\boldsymbol{\widehat{y_{ic}}}$ denotes the probabilistic prediction result, while $\boldsymbol{y_{ic}}$ denotes the classification result at the end of $\boldsymbol{i}$. training for the $\boldsymbol{c_{th}}$ category

As can be seen from Table 1, the soft pooling function performs the best, followed by max-pooling. These are followed by spatial pyramid-pooling and average pooling. It is noteworthy that the results of max-pooling and spatial pooling are close to each other, while average pooling has the lowest performance.

The main reason behind the high performance of the soft pooling method preferred in the study is the natural upper bound that allows larger activation values to have a greater impact on the output. It can be said that the high performance of the max pooling method after the soft pooling method is due to the high discriminative power of the maximum numbers in the extracted features. In SPM, the multilevel expansion process of feeding the fully connected layer did not have as high performance impact as soft pooling and max pooling. Average pooling, on the other hand, often results in a loss of performance in terms of information in terms of contrast. When calculating the average, all values in the filter are taken into account. As a result, if the values of all activation outputs are low, the average is also low. This situation is obtained in parallel with the literature [38]. Unlike in the literature, if most of the activation results are zero, the performance values decrease even more.

Table 1. Training performance results.

Pooling Type	Accuracy	F1 score	Precision	Recall	CE_{Loss}
Soft	98.25	98.40	98.25	98.25	0.001
Spatial pyramid	93.37	92.15	94.63	94.40	0.016
Max	95.50	94.72	95.61	95.60	0.015
Average	90.72	90.03	90.25	90.17	0.054

Table 2. Validation performance results.

Pooling Type	Accuracy	F1 score	Precision	Recall	CE_{Loss}
Soft	97.79	97.80	97.79	97.79	0.095
Spatial pyramid	91.60	90.87	91.68	91.50	0.776
Max	92.50	91.82	92.59	92.50	0.628
Average	89.09	88.05	89.82	88.30	0.386

In the study, the same metrics were used to measure the performance of the data allocated for testing. The results obtained here are shared in detail in Table 2.

When the performance results presented in Table 2 are analyzed, it is seen that they are in parallel with the training data. However, the test results were about 1% lower than the training results. The graphs of accuracy, F1 score, precision, recall and CE_{Loss} obtained from four different pooling functions as a result of the test are presented in Figures 5-9, respectively.

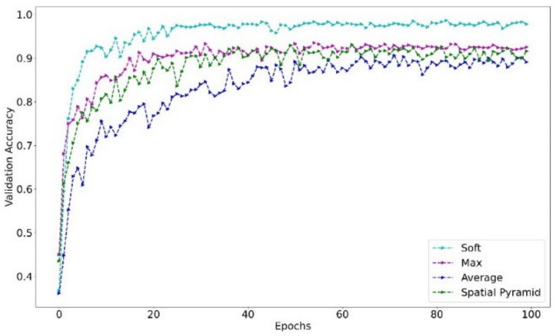

Figure 5. Validation accuracy performance graphics of proposed model for different pooling types

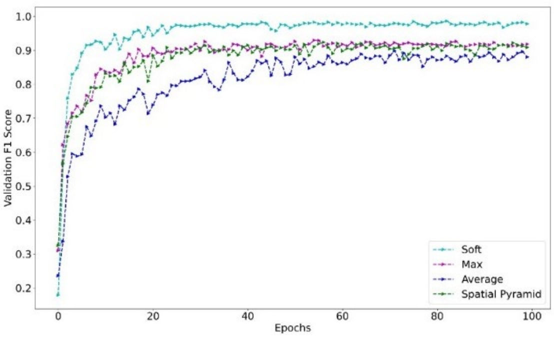

Figure 6. Validation F1 score performance graphics of proposed model for different pooling types

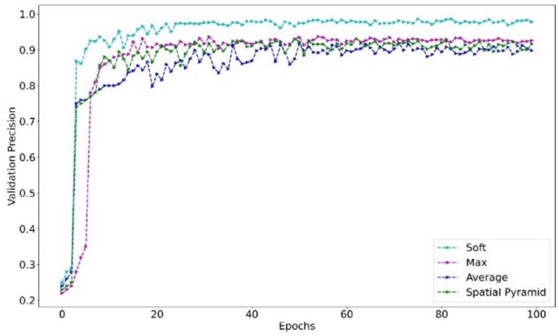

Figure 7. Validation precision performance graphics of proposed model for different pooling types

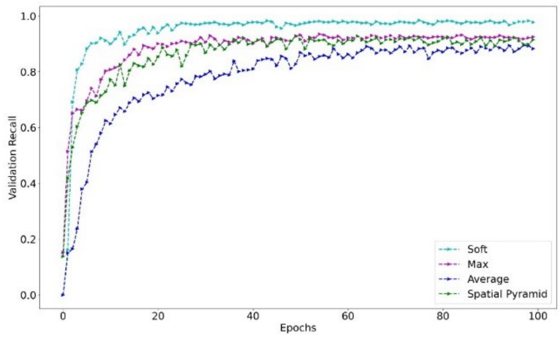

Figure 8. Validation recall performance graphics of proposed model for different pooling types

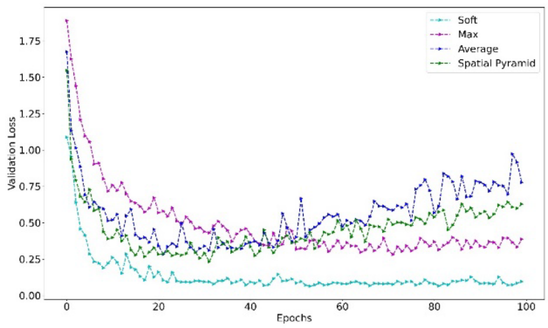

Figure 9. Validation loss performance graphics of proposed model for different pooling types

As can be seen in Figure 6, the lowest CE_{Loss} value is obtained from the soft pooling function, while the highest CE_{Loss} value is obtained from the average pooling function. This also confirms the accuracy values.

5. CONCLUSION

In this study, a study-specific convolutional deep learning model with four different pooling layers is used. The main purpose of the designed model is to measure the performance of soft pooling, max pooling, spatial pyramid pooling, and average pooling functions which are widely used in the literature. For this purpose, the natural images dataset used as benchmarking in the literature is used. On this dataset, accuracy, F1 score, precision, recall and CELoss metrics commonly used in the literature were used. The dataset used in the study was divided into two groups as training and test according to the cross validation 5 value.

In the light of the results obtained, the accuracy values of 0.9779 for soft pooling function, 0.9250 for max pooling function, 0.9160 for spatial pyramid pooling and 0.8909 for average pooling were obtained on the same dataset and deep learning model. As it can be seen from these results, the soft pooling function gave very good performance results compared to the other pooling functions used in the study. The main objective of this study is to measure the performance of the pooling functions comparatively.

It is seen that many CNN-based applications have been developed in autonomous systems and medical imaging systems, and the pooling methods tested in this study are also used in these applications [39-40]. It is seen that the performance results of the proposed method on the test set used in the study are quite good compared to other methods. In parallel with this, it is obvious that it will increase the success performance in CNN-based medical imaging and autonomous systems. When the pooling methods tested comparatively on the model proposed in the study are analysed, it is found that the computational complexity of the soft pooling method is higher than the other methods. However, the computational performance is quite high. If there are no hardware constraints, it is obvious that the soft pooling method will improve the performance of the study. On the other hand, if there are any hardware constraints, max pooling or average pooling can be preferred, respectively. Although the complexity of SPM is higher than max pooling, its performance is lower than max pooling. In the presence of hardware constraints, it would not be a reasonable solution to choose SPM. It is recommended that future research should take this situation into consideration.

At the same time, it is aimed to improve performance by developing different pooling functions with low computational cost with a new study on the dataset used in the study.

REFERENCES

1. Çetiner, H. and Metlek, S., "DenseUNet+: A novel hybrid segmentation approach based on multi-modality images for brain tumor segmentation,", J. King Saud Univ. - Comput. Inf. Sci., Vol. 35, Issue 8, Pages 101663, 2023.

2. Metlek, S., "CellSegUNet: an improved deep segmentation model for the cell segmentation based on UNet++ and residual UNet models,", Neural Comput. Appl., Vol. 36, Issue 11, Pages 5799–5825, 2024.

3. Çetiner, H., "Citrus disease detection and classification using based on convolution deep neural network,", Microprocess. Microsyst., Vol. 95, Issue 104687, Pages 1–10, 2022.

4. Nirthika, R., Manivannan, S., Ramanan, A., and Wang, R., "Pooling in convolutional neural networks for medical image analysis: a survey and an empirical study,", Neural Comput. Appl., Vol. 34, Issue 7, Pages 5321–5347, 2022.

5. Jena, B., Saxena, S., Nayak, G. K., Saba, L., Sharma, N., and Suri, J. S., "Artificial intelligence-based hybrid deep learning models for image classification: The first narrative review,", Comput. Biol. Med., Vol. 137, Pages 104803, 2021.

6. Mai, Z., Li, R., Jeong, J., Quispe, D., Kim, H., and Sanner, S., "Online continual learning in image classification: An empirical survey,", Neurocomputing, Vol. 469, Pages. 28–51, 2022.

7. Schmarje, L., Santarossa, M., Schröder, S.-M., and Koch, R., "A survey on semi-, self-and unsupervised learning for image classification,", IEEE Access, Vol. 9, Pages. 82146–82168, 2021.

8. Zafar, A., Aamir, M., Nawi, N. M., Arshad, A., Riaz, S., Alruban, A., Dutta, A. K., and Almotairi, S. "A Comparison of Pooling Methods for Convolutional Neural Networks,", Applied Sciences, Vol. 12, Issue 17, 2022.

9. Zhao, R., Song, W., Zhang, W., Xing, T., Lin, J, and Srivastava, M., "Accelerating binarized convolutional neural networks with software-programmable FPGAs,", in Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, Pages 15–24, 2017.

10. Yildirim, O., Baloglu, U. B., Tan, R.-S., Ciaccio, E. J. and Acharya, U. R., "A new approach for arrhythmia classification using deep coded features and LSTM networks,", Comput. Methods Programs Biomed., Vol. 176, Pages 121–133, 2019.

11. Cai, H., Gan, C., Wang, T., Zhang, Z., and Han, S., "Once-for-all: Train one network and specialize it for efficient deployment,", arXiv Prepr. arXiv1908.09791, 2019.

12. Murray, N. and Perronnin, F., "Generalized max pooling,", in Proceedings of the IEEE conference on computer vision and pattern recognition, Pages 2473–2480, 2014.

13. Roy, P., Ghosh, S., Bhattacharya, S., and Pal, U., "Effects of degradations on deep neural network architectures," arXiv Prepr. arXiv1807.10108, 2018.

14. Özdemir, C., "Avg-topk: A new pooling method for convolutional neural networks,", Expert Syst. Appl., Vol. 223, Pages 119892, 2023.

15. Dogo, E. M., Afolabi, O. J., Nwulu, N. I., Twala, B. and Aigbavboa, C. O., "A Comparative Analysis of Gradient Descent-Based Optimization Algorithms on Convolutional Neural Networks,", in 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), Pages 92–99, 2018.

16. Sikandar, S., Mahum, R. and Alsalman, A., "A Novel Hybrid Approach for a Content-Based Image Retrieval Using Feature Fusion,", Applied Sciences, Vol. 13, Issue 7, 2023.

17. Prabavathi, M. V and Sakthi, M., "Poisson Wavelet Quantized Piecewise Regressive Distributed Coding for Image Compression and Transmission,", Tuijin Jishu/Journal Propuls. Technol., Vol. 44, Issue 6, 2023.

18. Praveenkumar, G. D. and Nagaraj, R., "Regularized Anisotropic Filtered Tanimoto Indexive Deep Multilayer Perceptive Neural Network learning for effective image classification,", Neurosci. Informatics, Vol. 2, Issue 2, Pages 100063, 2022.

19. Mohamed, E. A., Gaber, T., Karam, O. and Rashed, E. A., "A Novel CNN pooling layer for breast cancer segmentation and classification from thermograms,", PLoS One, Vol. 17, Issue 10, Pages e0276523, 2022.

20. Vigneron, V., Maaref, H. and Syed, T. Q., "A New Pooling Approach Based on Zeckendorf's Theorem for Texture Transfer Information,", Entropy, Vol. 23, Issue 3. 2021.

21. Sharma, T., Verma, N. K. and Masood, S. "Mixed fuzzy pooling in convolutional neural networks for image classification,", Multimed. Tools Appl., Vol. 82, Issue 6, Pages 8405–8421, 2023.

22. Bhattacharjee, K., Pant, M., Zhang, Y.-D. and Satapathy, S. C., "Multiple Instance Learning with Genetic Pooling for medical data analysis,", Pattern Recognit. Lett., Vol. 133, Pages 247–255, 2020.

23. Boureau, Y.-L., Ponce, J., and LeCun, Y., "A theoretical analysis of feature pooling in visual recognition,", in Proceedings of the 27th international conference on machine learning (ICML-10), Pages 111–118, 2010.

24. Singh, P., Chaudhury, S., and Panigrahi, B. K., "Hybrid MPSO-CNN: Multi-level Particle Swarm optimized hyperparameters of Convolutional Neural Network,", Swarm Evol. Comput., Vol. 63, Pages 100863, 2021.

25. He, Z., Shao, H., Zhong, X., and Zhao, X., "Ensemble transfer CNNs driven by multi-channel signals for fault diagnosis of rotating machinery cross working conditions,", Knowledge-Based Syst., Vol. 207, Pages 106396, 2020.

26. Riesenhuber, M. and Poggio, T., "Hierarchical models of object recognition in cortex,", Nat. Neurosci., Vol. 2, Issue 11, Pages 1019–1025, 1999.

27. Stergiou, A., Poppe, R., and Kalliatakis, G., "Refining activation downsampling with SoftPool,", in Proceedings of the IEEE/CVF international conference on computer vision, Pages 10357–10366, 2021.

28. Zeiler, M. D. and Fergus, R., "Visualizing and understanding convolutional networks,", in Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, Springer, Pages 818–833, 2014.

29. Girshick, R., Donahue, J., Darrell, T., and Malik, J., "Rich feature hierarchies for accurate object

detection and semantic segmentation,", in Proceedings of the IEEE conference on computer vision and pattern recognition, Pages 580–587, 2014.

30. Mumuni, A. and Mumuni, F., "CNN Architectures for Geometric Transformation-Invariant Feature Representation in Computer Vision: A Review,", SN Comput. Sci., Vol. 2, Issue 5, Pages 340, 2021.

31. Cao, Z., Xu, X., Hu, B. and Zhou, M., "Rapid Detection of Blind Roads and Crosswalks by Using a Lightweight Semantic Segmentation Network,", IEEE Trans. Intell. Transp. Syst., Vol. 22, Issue 10, Pages 6188–6197, 2021.

32. Ghosh, P., Azam, S., Jonkman, M., Karim, A., Shamrat, F. J. M., Ignatious, E., and De Boer, F. , "Efficient prediction of cardiovascular disease using machine learning algorithms with relief and LASSO feature selection techniques,", IEEE Access, Vol. 9, Pages 19304–19326, 2021.

33. Gupta, A., Kumar, R., Arora, H. S., and Raman, B., "MIFH: A machine intelligence framework for heart disease diagnosis,", IEEE access, Vol. 8, Pages 14659–14674, 2019.

34. Wang, L., Zhou, W., Chang, Q., Chen, J., and Zhou, X., "Deep ensemble detection of congestive heart failure using short-term RR intervals,", IEEE Access, Vol. 7, Pages 69559–69574, 2019.

35. Miao, F., Cai, Y. P., Zhang, Y. X., Fan, X. M., and Li, Y., "Predictive modeling of hospital mortality for patients with heart failure by using an improved random survival forest,", IEEE Access, Vol. 6, Pages 7244–7253, 2018.

36. Mohan, S., Thirumalai, C., and Srivastava, G., "Effective heart disease prediction using hybrid machine learning techniques,", IEEE access, Vol. 7, Pages 81542–81554, 2019.

37. Çelebi, S. B. and Emiroğlu, B. G., "A Novel Deep Dense Block-Based Model for Detecting Alzheimer's Disease,", Applied Sciences, Vol. 13, Issue 15, 2023.

38. Zhang, W., Li, C., Peng, G., Chen, Y., and Zhang, Z., "A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load,", Mech. Syst. Signal Process., Vol. 100, Pages 439–453, 2018.

39. Metlek, S. and Çetiner, H., "ResUNet+: A New Convolutional and Attention Block-Based Approach for Brain Tumor Segmentation," IEEE Access, Vol. 11, Pages. 69884–69902, 2023.

40.Metlek, S. and Çetiner, H., "Inception SH: A New CNN Model Based on Inception Module for Classifying Scene Images,", Mühendislik Bilim. ve Tasarım Dergisi, Vol. 12, Issue 2, Pages 328–344, 2024.