TELKOMNIKA Telecommunication, Computing, Electronics and Control
Vol. 19, No. 1, February 2021, pp. 163~172
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v19i1.16398

Half Gaussian-based wavelet transform for pooling layer for convolution neural network

Aqeel M. Hamad Alhussainy, Ammar D. Jasim
Department of Information Engineering, Al-Nahrain University, Iraq

Article Info

Article history:

Received Apr 15, 2020
Revised Jul 3, 2020
Accepted Aug 29, 2020

Keywords:

Convolution neural network
Gaussian
HGT1
HGT2
HGT3

ABSTRACT

Pooling methods are used to select most significant features to be aggregated to small region. In this paper, anew pooling method is proposed based on probability function. Depending on the fact that, most information is concentrated from mean of the signal to its maximum values, upper half of Gaussian function is used to determine weights of the basic signal statistics, which is used to determine the transform of the original signal into more concise formula, which can represent signal features, this method named half Gaussian transform (HGT). Based on strategy of transform computation, Three methods are proposed, the first method (HGT1) is used basic statistics after normalized it as weights to be multiplied by original signal, second method (HGT2) is used determined statistics as features of the original signal and multiply it with constant weights based on half Gaussian, while the third method (HGT3) is worked in similar to (HGT1) except, it depend on entire signal. The proposed methods are applied on three databases, which are (MNIST, CIFAR10 and MIT-BIH ECG) database. The experimental results show that, our methods are achieved good improvement, which is outperformed standard pooling methods such as max pooling and average pooling.

Corresponding Author:

Aqeel M. Hamad Alhussainy
Department of Information Engineering
Al-Nahrain University
Al Jadriyah Bridge, Baghdad 64074, Iraq
Email: aqeel_alhussainy@utq.edu.iq

1. INTRODUCTION

The recent development is used neural network in spired system such as convolutional neural network (CNN), De-noising auto encoder and other deep learning neural network has derived significant development from building more important and complicated network structure, which lead to more non-linear activations [1-3]. Despite the development progress in CNNs, there are still several challenges encountered by this network such as problem of high capacity because of huge processing data, which may result in over fitting problem due to high capacity of CNN [4-6]. In order to solve these problems, different regularization methods were proposed such as weight decay, weight tying and pooling techniques. The central role for CNN network is the features pooling operation, however, pooling have been little revised beyond standard methods of average and max pooling [7-9]. In this paper, anew pooling of features method is proposed based on probability function, the proposed method is replaced the output of convolutional layer with determinists features by using pooling operation, which is evaluated based on distribution statistics for each pooling window, the weight of these statistics are computed depending on normal distribution of statistics [10-13]. the main contributions of this work is that, the basic properties of the signal are filtered by select the most significant information, while the detail of the signal will have little effect, so the elimination of the signal will be satisfied by discard less significant information through the CNNs and this is eliminated shortcoming of max and average pooling methods [14, 15].

2. LITERATURE SURVEY

Tavis William and Roberli is introduced a pooling method based on wavelet transform, this method was based on decompose the original image into second level transform of wavelet, then delete all the sub-band details of first level depending on the fact that , approximation coefficients represent the basic information of the original data, this can reduce the features of the original signal by discarding less significant information [16]. Chen-Yu Lee *et al.*, they are studied the performance of combining average pooling with max pooling and the strategy of tree structured fusion of filters. The basic idea of this work is used learning process of mixed rate between max and average pooling method, they are referred to this method as mixed method, while the second used method in this work was based on gated mask, which is used to find mix of max and average pooling, they are refered to this method as gated max-average method pooling [17].

Dingjun Yu *et al.* they are proposed a method for feature pooling based on replacing determinists with stochastic operation, this is accomplished by chose random value to select the max or average pooling method, the basic benefit from this method is to avoid over fitting problem. They are applied mixed pooling by three different approaches, which are apply pooling for all features in a layer, or by using mixed within layer, or by using mixing between regions within layer. The proposed methods are applied on different types of database [18]. M. D. Zeiler and Rob Fegus are proposed to select activation, that is driven from a multidimensional distribution by activation in the region of pooling(pool size), this is performed by first computing the probability for each region, then this probability is normalized, then sample from these distribution based on the probability is selected to be the pooling features, different methods are used to select these probabilities, which means that, the selected elements for pooling feature may not be the largest value [19]. Takumi Kobayashi is proposed feature pooling layer based on distribution of probabilities over activation, this is performed by determine the statistics of standard deviation and mean depending on distribution of Gaussian function, the basic idea of this work is to summarize the distribution of Gaussian and aggregate the activation into two basic values, which are standard deviation and average, this method is applied later to stochastic pooling method [13].

3. METHODOLOGY

In this paper, we have proposed a new pooling methods based on probability function, Figure 1 describes the block diagram of the pooling layer, the basic component of this layer is feature computation, which is extracted depending on algorithm (1) by calculating the basic statistics, which can be used to compute the weights of each element according to (1) and (2),which are represented average and standard deviation respectively [13, 20].

$$\mu_x = \frac{1}{|R|}\sum_{(p \epsilon R)} X_p \quad (1)$$

$$\sigma_x = \frac{1}{|R|}\sum_{(p \epsilon R\,)} (X_p - \mu_x)^2 \quad (2)$$

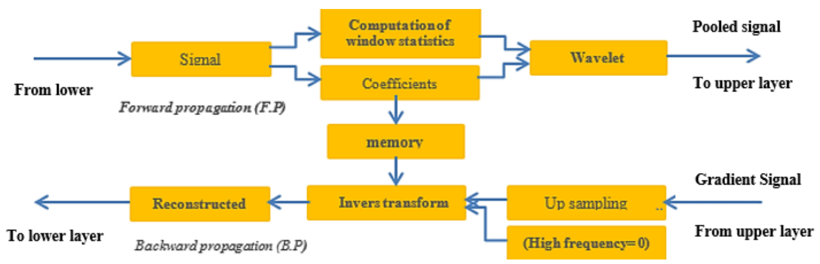

Figure 1. The proposed pooling layer block diagram

The second half of Gaussian function represents the statistics between mean and maximum value, which represents the most important characteristics of the signal. So, the Gaussian is reconstructed for upper half of its function as shown in Figure 2, then most significant statistics are calculated. These statistics will be used later to determine the features and their weights according to the significant of each of them. The selected features will be determined as shown in (3)

$$Y = \sum_{i=0}^{4}(\mu_x + \frac{i*\sigma_x}{2}) * w(i) \qquad (3)$$

where $w(i)$ represent the weight of each element, while μ_x , σ_x are mean and standard deviation of the signal respectively [21, 22].

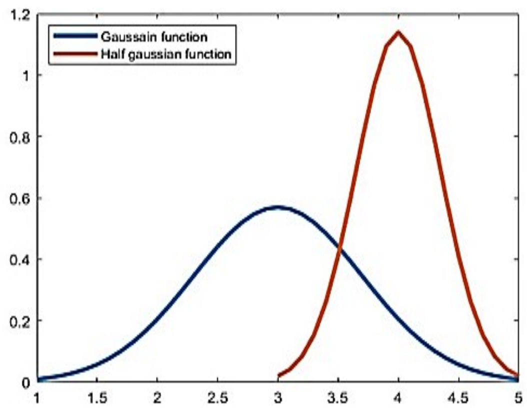

Figure 2. Gaussian and half Gaussian function

3.1. Proposed algorithm

Based on the strategy of transform computation, we have proposed three algorithms. These methods are half Gaussian transform1 (HGT1), half Gaussian transform2 (HGT2) and half Gaussian transform3 (HGT3). The algorithms are differenced in the strategy of transform computation. Also, the statistics are determined in different window size. The details are describedin next sections.

3.1.1. HGT1

This algorithm is used the basic features of the signal (mean and standard deviation), which are determined as shown in (1) and (2) respectively. At first, the size and stride for each pool size window and others parameters are initialized, then mean and standard deviation are determined for each pool window, then the upper half of the Gaussian distribution function is determined, depending on these statistics, basic elements of this function are computed, which are $(\mu_x - \sigma_x\ ,\ \mu_x - \frac{1}{2} * \sigma_x,\ \ \mu_x\ , \mu_x + \frac{1}{2} * \ \sigma_x\ \ and\ \ \ \mu_x + \sigma_x)$. The weights of these elements are determined according to Gaussian function shown in (4), these weights are multiplied by original signal to compute the basics features (pooled signal). The details description of this method shown in Figure 3, which shows algorithm HGT1.

$$f(x) = \frac{1}{\sqrt{2\pi_{*}\sigma x^2}}\ e^{-\frac{(x-\mu_x)^2}{2\sigma^2}} \qquad (4)$$

3.1.2. HGT2

In this algorithm, the weights are determined for Gaussian function, then for each pool size window, the mean and standard deviation are determined. These statistics are used to determine the basic elements of half Gaussian function, which are $((\mu_x - \sigma_x\ , \mu_x\ - \frac{1}{2} * \sigma_x, \mu_x, \mu_x + \frac{1}{2} * \ \sigma_x\ and\ \ \mu_x + \sigma_x))$ **.** Then, the determined elements are multiplied by the constant weights, which are determined at first step. The details description of this algorithm are shown in Figure 4.

3.1.3. HGT3

This algorithm is similar to algorithm II, accept that, it is determined the mean and standard deviation for entire signal instead of determined it for each pool size. At first, these statistics are calculated for entire signal, then the basic elements of the new Gaussian function are determine, which are $((\mu_x - \sigma_x\ , \mu_x - \frac{1}{2} * \sigma_x, \mu_x,\ \ \mu_x + \frac{1}{2} * \ \sigma_x\ and\ \ \mu_x + \ \sigma_x))$. These values are used as inputs to Gaussian function to determine the features, which are multiplied by the original signal to compute the pooled signal. The details are shown in Figure 5.

Algorithm 1 (HGT1)
Initialization Z: size of signal, P pool size; S: stride
Read the input signal
While (Pool_size in signal)
Do{
Get window= pool_size; x= pool signal;
Determine mean and standard deviation of the window by:

$$\mu_x = \frac{1}{|R|}\sum_{(p\epsilon R)} X_p ,\sigma_x^{\square} = \frac{1}{|R|}\sum_{(p\epsilon window)} (X_p - \mu_x)^2,$$

Determine the values :

$$\mu_x - \sigma_x ,\mu_x - \frac{1}{2} * \sigma_x, \mu_x, \quad \mu_x + \frac{1}{2} * \sigma_x \, and \; \mu_x + \sigma_x$$

Determine weight of this points by:

$f(x) = \frac{1}{\sqrt{2\pi_{\cdot\sigma x^2}}} \; e^{-\frac{(x-\mu_x)^2}{2\sigma^2}}$ **for all values to be y1,y2,y3.y4 and y5**

Determine the transform by applying:

$Y = \sum_{i=1}^{pool\_size}(x) * y(i);$

End while
End of algorithm

Figure 3. HGT1 algorithm

Algorithm (2) HGT2
Initialization Z: size of signal, P pool size; S: stride
Read the input signal
Give x initial value x = [-2:1:2]
Determine constant weight based on Gaussian function for all x values by:

$f(x) = \frac{1}{\sqrt{2\pi_{\cdot\sigma x^2}}} \; e^{-\frac{(x-\mu_x)^2}{2\sigma^2}}$ **to be: w1,w2,w3,w4,w5**

While (Pool_size in signal)
Do{
Get window= pool_size;
Determine mean and standard deviation of the window by:

$$\mu_x = \frac{1}{|R|}\sum_{(p\epsilon R)} X_p ,\sigma_x^2 = \frac{1}{|R|}\sum_{(p\epsilon window)} (X_p - \mu_x)^2,$$

Determine the basic half Gaussian weight to be:

$(\mu_x, \mu_x + \frac{\sigma_x}{2}, \mu_x + \sigma_x, \mu_x + 3\frac{\sigma_x}{2}, \mu_x + max_x$

Determine the transform by applying:

$Y = \sum_{i=-2}^{2}(\mu_x + \frac{i*\sigma_x}{2}) * w(i);$

End while
End of algorithm

Figure 4. HGT2 algorithm

Algorithm 3 (HGT3)
Initialization Z: size of signal, P pool size; S: stride
Read the input signal
Determine mean and standard deviation of the entire signal by:

$$\mu_x = \frac{1}{|R|}\sum_{(p\epsilon R)} X_p ,\sigma_x^{\square} = \frac{1}{|R|}\sum_{(p\epsilon window)} (X_p - \mu_x)^2$$

While (Pool_size in signal)
Do{
Determine the values:

$$\mu_x - \sigma_x ,\mu_x - \frac{1}{2} * \sigma_x, \mu_x, \quad \mu_x + \frac{1}{2} * \sigma_x \, and \; \mu_x + \sigma_x$$

Determine weight of this points by:

$f(x) = \frac{1}{\sqrt{2\pi_{\cdot\sigma x^2}}} \; e^{-\frac{(x-\mu_x)^2}{2\sigma^2}}$ **for all values to be:**

y1, y2, y3. y4 and y5

Determine the transform by applying:

$Y = \sum_{i=1}^{pool\_size}(x) * y(i);$

End while
End of algorithm

Figure 5. HGT3 algorithm

4. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed pooling methods are used CNN, and applied on different types of database to test the performance of the proposed methods as compared with other methods. These databases are MNIST and CIFAR10, which are two-dimension signal (image) with size (28*28) and (32*32) respectively. The other database was MIT-BIH ECG database, which is one-dimension signal. The experiments are executed by Intel®core ™i7-4500CPU@2.40GHz processor, with8GB of RAM, 64-bit windows seven operating system, on Matlab (2019a). The results are compared with results of standard methods.

4.1. MNIST database results

This database is contained 60000 image of gray scale image with size (28*28), it is divided into (50000) image, which are used for training, while the remaining 10000 images are used for test the proposed model [23]. The CNN is trained with initial learning rate 0.01,10 epochs and 58 iteration per epoch. Table 1 describes the results as compared with standard max and average pooling methods ,it is clear that the proposed method are outperformed these method, the best accuracy is satisfied with (HGT1+average) method, which is achieved accuracy (99.72%) verses (99.48%) and (99.42%) for max. and average pooling methods respectively, also this method is achieved lowest FPR (0.28%) compared with (0.34%) for Max method as shown in Table 2, which shows the different performance metrics for (HGT1) methods.

Table 1. Results of (HGT1) method for MNIST classification

Method	Max	average	HGT1	HGT1+Max	HGT1+average
Accuracy (%)	99.48	99.42	99.68	99.72	99.96

Table 2. Performance metrics of (HGT1) methods for handwrite digit classification

Method	HGT1	HGT1+Max	HGT1+average
Accuracy (%)	99.68	99.72	99.96
Sensitivity (SN%)	99.66	99.68	99.72
False positive Rate FPR (%)	0.34	0.28	0.28
Specificity (%)	99.66	99.68	99.72
ERR (%)	0.32	0.28	0.04

The results of second method is described in Table 3, which gives the best results by (HGT2+Max), and other metrics of performance are explained in Table 4, from Table 4 it is clear that (HGT2+Max) gives lowest FPR (0.28) with the highest accuracy (99.72). The tables are described the improvements of our methods in terms of accuracy, sensitivity and precision with minimum false positive rate (FPR). The accuracy and loss training progress for (HGT2+Max) method are shown in Figures 6 and 7 respectively, it is clear that, the accuracy is reached to more than 98.5 with less than 2 epochs, this is due to extracting basic features of the image with less elimination as compared with max and average pooling methods, also the loss is attenuated to less than 0.15. The confusion matrix details for this method is described in Figure 8, which is described the high matching between the predicted and actual values, since most of the classes are matched perfectly. Table 5 shows the result of third method (HGT3), which is achieved less results as compared with HGT1 and HGT2 methods, the detail description of these methods for all performance metrics are described in Table 6.

Table 3. Results of (HGT2) method for MNIST classification

Method	Max	average	HGT2	HGT2+Max	HGT2+average
Accuracy (%)	99.48	99.42	99.52	99.72	99.52

Table 4. Performance metrics of (HGT2) methods for handwrite digit classification

Method	HGT2	HGT2+ Max	HGT2+ Average
Accuracy (%)	99.52	99.72	99.52
Sensitivity (SN%)	99.52	99.72	99.58
False Error Rate FER (%)	0.48	0.28	0.42
Specificity (%)	99.52	99.72	99.56
ERR (%)	0.48	0.28	0.48

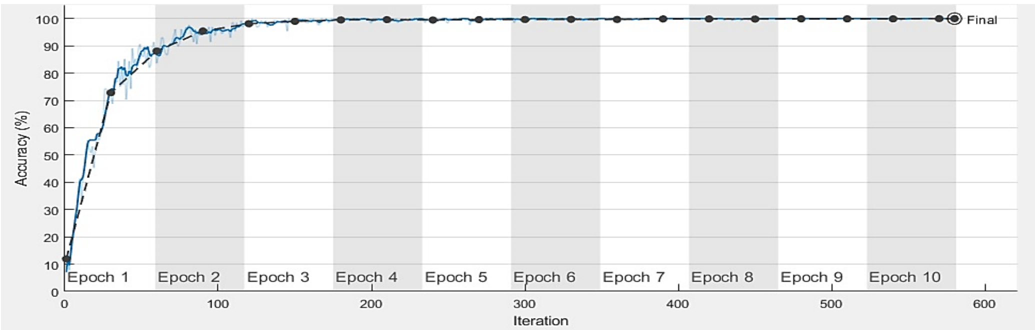

Figure 6. Accuracy training progress for (HGT2+Max) method

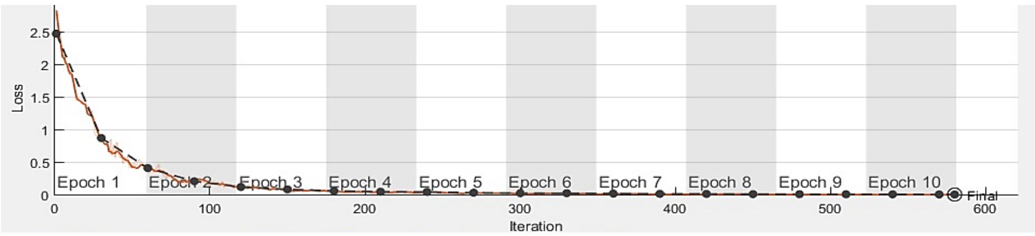

Figure 7. Loss training progress for (HGT2+Max) method

Table 5. Results of (HGT3) method for MNIST classification

Method	Max	average	HGT3	HGT3+Max	HGT3+average
Accuracy (%)	99.48	99.42	99.04	99.52	99.96

Table 6. Performance measures for HGT3 for handwrite digit classification

Method	HGT3	HGT3+ Max	HGT3+ Average
Accuracy (%)	99.04	99.52	99.96
Sensitivity (SN%)	99.30	99.52	99.96
False Error Rate FER (%)	0.7	0.48	0.04
Specificity (%)	99.12	99.52	99.96
ERR (%)	0.96	0.48	0.04

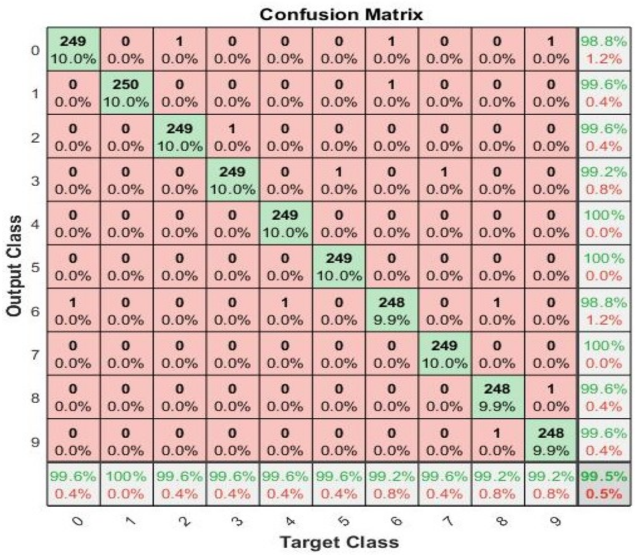

Figure 8. confusion matrix of (HGT3+Max) method

4.2.Results CIFAR 10 dataset

This dataset is constructed from 60000 image, each image with size (32*32) RGB color image, the model is trained on (50000), while the test dataset was 10000 images [24]. In this experiment, the same parameters are used for all pooling methods(the proposed and standard), which are 10 epoch, 128 batch size with 0.01 learning rate .The results of HGT1 method are described in Table 7, it clear that , our method (HGT) gives the best results, because combining this method with max and average can eliminate some significant information from the image, the performance of the methods are shown in Table 8, the lowest FPE is satisfied in our proposed method (HGT1), which is achieved (26.3%). The confusion matrix of this method is shown in Figure 9, which shows good matching between predicted and actual classes.

The progress of the accuracy and loss training are shown in Figures 10 and 11 respectively, the accuracy is reached to (60%) in less than 2 epochs, then increased gradually, while the loss is attenuated to less than 1 in 2 epochs and then, it is decreased slowly. The results of HGT2 are presented in Tables 9 and 10 respectively, there is small improvement compared with max and average pooling methods, because this method is depended on feature of the image instead of the image itself for extraction the pooled signal. Tables 11 and 12 represent results of HGT3 method, which is less in most performance metrics (acc 72.42%) and (FPE27.58 %), this due to that, this method is depended on the statistics of entire signal instead of each pool size window from the signal, which not gives the method high dynamic in dealing with the signal, and this is happened in the first method (HGW1).

Table 7. Results of different proposed pooling method for CIFAR10 classification

Method	Max	average	HGT1	HGT1+Max	HGT1+average
Accuracy (%)	72.59	72.41	73.67	**72.2**	72.7

Table 8. Performance measures of HGT method for CIFAR10 database.

Method	HGT1	HGT1+Max	HGT1+average
Accuracy (%)	73.67	72.2	72.7
Sensitivity (SN%)	73.67	72.27	72.7
False positive Rate FPR (%)	26.3	26.6	27.27
Specificity (%)	73.3	72.29	72.73
ERR (%)	26.33	27.8	27.3

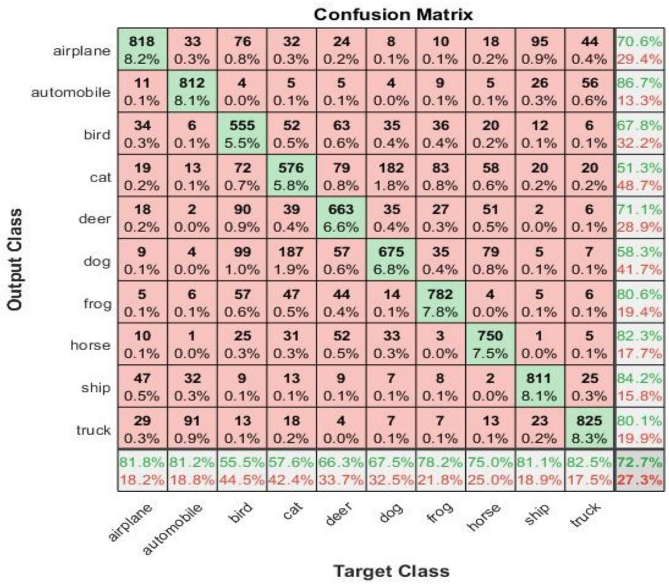

Figure 9. Confusion matrix of HGT1 method

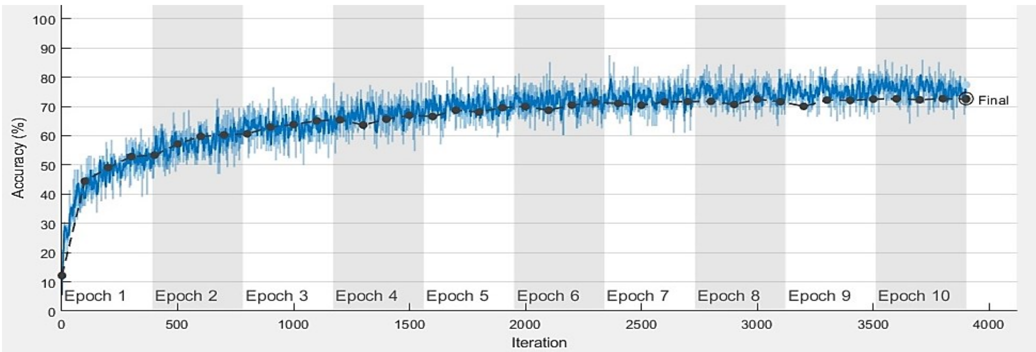

Figure 10. accuracy progress for training HGT1 method

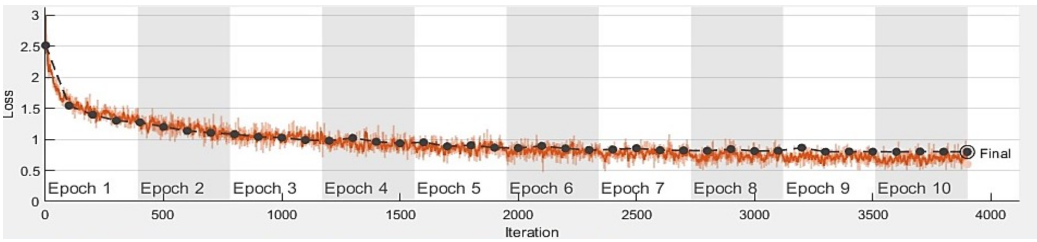

Figure 11. Loss progress for training HGT1 method

Table 9. Results of HGT2 method for CIFAR10 classification

Method	Max	average	HGT2	HGT2+Max	HGT2+average
Accuracy (%)	72.59	72.41	72.21	72.42	72.7

Table 10. Performance measures of HGT2 method for CIFAR10 database

Method	HGT2	HGT2+Max	HGT2+average
Accuracy (%)	72.21	72.42	72.7
Sensitivity (SN%)	72.23	72.42	72.7
False positive Rate FPR (%)	27.77	27.58	27.3
Specificity (%)	72.21	72.38	72.68
ERR (%)	27.79	27.58	27.3

Table 11. Results of HGT3method for CIFAR10 classification

Method	Max	average	HGT3	HGT3+Max	HGT3+average
Accuracy (%)	72.59	72.41	72.41	72.33	72.38

Table 12. Performance measures of HGT 3method for CIFAR10 database

Method	HGT3	HGT3+Max	HGT3+average
Accuracy (%)	72.41	72.33	72.38
Sensitivity (SN%)	72.51	72.36	72.39
False positive Rate FPR (%)	27.49	27.64	27.62
Specificity (%)	72.40	72.30	72.35
ERR (%)	28.59	27.67	27.62

4.3. Result of ECG signal

This dataset is contained data with size (109446*188), which represent (109446) signal, each one with one dimension, with 188 samples, the training set with size (87554), while test size is (21892) [25, 26]. The model is trained with same parameters for all methods of pooling layers, which are 10 epochs, batch size 128 and 0.01 learning rate. Table 13 shows the results of HGT1 compared with other most common methods, the best results are achieved with (HGT1) method (accuracy 94.51%,) with lowest FPE (4.44), while combining this method with max and average are achieved less accuracy, this is happened due to that ECG signal is oscillated signal , Max or average can produce elimination of more significant information, which may reduce the overall accuracy. The results of HGT2 is shown in Table 14, it gives highest accuracy (94.51%).

The results of third proposed method (HGT3) are shown in Table 15. It is clear that, this method gives the lowest result as compared with other proposed methods (HGT1 and HGT2), which satisfied (Acc=92.35 %), the results are dropped because this method is depended on statistics of the entire signal instead of every pool size, which is very different because ECG signal have high differences in their samples. The detail performance metrics for our methods are described in Table 16, which is concluded that, the best results are obtained with (HGT2) method, which is achieved accuracy (94.94%) with ERR (5.09%), and FPR (4.44%) this improvement is achieved because the pooled signal is depended on extraction of the most significant feature of the signal. The progress of training for accuracy and loss for (HGT2) method are shown in Figures 12 and 13 respectively, after one epoch, the training accuracy is reached to approximately (90%) and the loss is decreased to less than (0.4).

Table 13. Results of different proposed pooling method for CIFAR10 classification

Method	Max	average	HGT1	HGT1+Max	HGT1+average
Accuracy (%)	93.27	94.01	94.51	93.92	93.97

Table 14. Results of HGT2 method for CIFAR10 classification

Method	Max	average	HGT2	HGT2+Max	HGT2+average
Accuracy (%)	93.27	94.01	94.94	94.54	94.30

Table 15. Results of HGT2 method for CIFAR10 classification

Method	Max	average	HGT3	HGT3+Max	HGT3+average
Accuracy (%)	93.27	94.01	92.35	92.13	92.24

Table 16. Performance of the proposed methods

Method	HGT1	HGT2	HGT3
Accuracy (%)	94.51	94.94	92.35
Sensitivity (SN%)	94.21	94.56	91.85
False positive Rate FPR (%)	5.79	4.44	8.15
Specificity (%)	95.305	94.56	91.55
ERR (%)	4.49	5.09	7.65

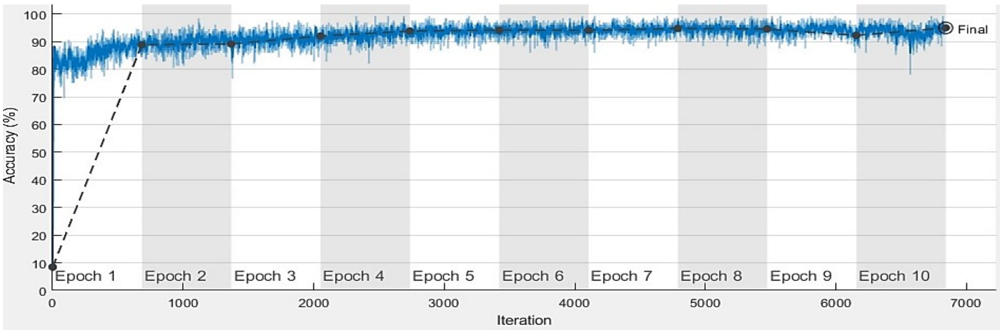

Figure 12. Accuracy progress for HGT2 method

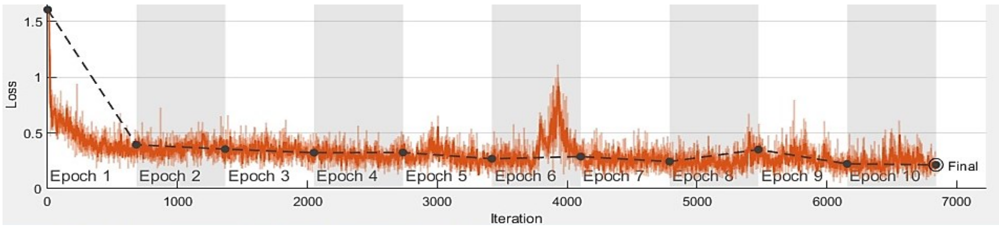

Figure 13. Loss progress for HGT2 method

5. CONCLUSION

The most important layer in CNNs is convolutional layer, but according to the size of inputs, number of used filters and kernel size of each filter in this layer, the output of this layer will be too much and this may reduce the efficiency of the network and increase its complexity. So, different studies and research have been performed to reduce this problem. In this paper, three methods have been proposed based on the principle of Gaussian function, by using the fact that the second half of Gaussian function represents the statistics between mean and maximum value, which represents the most important characteristics of the signal. So, the main concentration of information is from mean to max, and depending on this fact, the Gaussian is reconstructed for upper half of its function, and depending on the most significant features. Depending on the new function (HG), the basic statistics values are calculated to be weights for the original signal to calculate the features (selecting feature). Three method are proposed HGT1,which is used the values of basic statistics after normalized it as weights to be multiplied by original signal, the HGT2 is used the determined statistics as features of the original signal and multiply it with constant weights based on half Gaussian ,while the HGT3 is worked in similar way to (HGT1) except that, it is depended on entire signal instead of every pool size for calculation the basic statistics. The proposed methods are applied to three types of datasets, which are (MNIST and CIFAR10), which are two-dimension signal and MIT-BIH ECG dataset, which is one-dimension signal.

For MNIST dataset, the best results are achieved with HGT1+average, (accuracy 99.96% and FPR 0.28%), while for CIFAR10 dataset, the best result are satisfied with HGT1 method (accuracy 73.67% and FPR 26.3%). For ECG dataset, the HGT1 gives the good results (acc=94.51 %), (sen 94.21%) and (FPE 5.79%), and HGT2 gives approximately better results ,which are (acc=95.91 %), (sen.94.56%) and (FPE4.44 %), while the HGT3 is satisfied the lowest results (acc= 92.35%), (sen.91.85%) and (FPE8.15 %), the result is dropped because this method is depended on the statistics of overall signal instead of statistics of every pool size as in HGT1, which is very different because ECG signal have high differences in their samples. The experimental result show that, our methods are achieved good improvements, which is performed or outperformed standard pooling methods such as max pooling and average pooling, and can be used in classification problem.

REFERENCES

[1] Agostinelli L., *et al.*, "Learning activation functions to improve deep neural networks,". *ICLR*, 2015.
[2] Krizhevsky, S., *et al.*, "ImageNet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, no. 2, January 2012.
[3] Mallat, S., "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 11, no. 7, pp. 674-693, July 1989.
[4] Y. Boureau, P., *et al.*, "A theoretical analysis of feature pooling in visual recognition," *Proceeding of the 27th International Conference on Machine Learning (ICML-10)*, June 2010.

[5] K. He, Zhang, Ren, and J. Sun, “Deep residual learning for image recognition,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, June 2016.
[6] Travis Williams and Robert Li, “Advanced image classification using wavelets and convolutional neural networks,” *15th IEEE International Conference on Machine Learning and Applications*, pp. 233–239, December 2016.
[7] J. Kim, *et al*., “Accurate image super-resolution using very deep convolutional networks,” *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1646–1654, June 2016.
[8] C. Szegedy, *et al*., “Going deeper with convolutions,” *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.
[9] I. Ioffe and Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shif,” *arXiv preprint arXiv:1502.03167*, February 2015.
[10] A. Ferra, *et al*., “Wavelet pooling for CNNs,” *ECCV*, vol. 4, pp. 671–675, 2019.
[11] P. Liu, *et al*., “Multi-level wavelet convolutional neural networks,” *CoRR*, July 2019.
[12] Allison M. Rossetto, *et al.,* “Improving classification with CNNs using Wavelet Pooling with Nesterov-Accelerated Adam, EPiC Series in Computing,” *Proceedings of 11th International Conference on Bioinformatics and Computational Biology*, vol. 60, pp. 84-93, 2019.
[13] Takumi K., “Gaussian-based pooling for convolutional neural networks,” *33rd Conference on Neural Information Processing Systems*,” Advances in Neural Information Processing Systems 32, 2019.
[14] Danilo P., *et al*, “Acoustic impulsive noise based on non-gaussian models: an experimental evaluation,” *Sensors*, vol. 19, no. 12, pp. 2827, June 2019.
[15] Jawad N., *et al*, “Max-pooling convolutional neural networks for vision-based hand gesture recognition,” *2011 IEEE International Conference on Signal and Image Processing Applications*, November 2011.
[16] T. Williams and R. Li., “Wavelet pooling for convolutional neural networks,” *ICLR*, 2018.
[17] C.-Y. Lee, *et al*, “Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree,” *AISTATS*, pp. 464–472, 2016.
[18] D. Yu, *et al*., “Mixed pooling for convolutional neural networks,” *RSKT*, pp. 364–375, October 2014
[19] M. Zeiler, *et al.,* “Stochastic pooling for regularization of deep convolutional neural network,” *ICLR*, January 2013.
[20] Swietojanski and Renals. "Differentiable pooling for unsupervised acoustic model adaptation,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 24, no. 10, pp. 1773–1784, March 2016.
[21] A. Pewsey, “Large-sample inference for the general half-normal distribution,” *Communications in Statistics - Theory and Methods*, vol. 31, no.7, pp.1045–1054, July 2002.
[22] Wenhao G., “A generalization of the slash half normaldistribution: properties and inferences,” *Journal of Statistical Theory and Practice*, vol.8, no. 2, pp. 283–296, March 2014.
[23] Yann Le. C., *et al*, "The MINST database for hand wright digits,” [Online]. Available: http://yann.lecun.com/exdb/mnist/. Accessed: 2005.
[24] Alex K., *et al.*, “The CIFAR-10 dataset” [Online]. Available: https://www.cs.toronto.edu/~kriz/cifar.html.
[25] George B., *et al*, “The impact of the MIT-BIH Arrhythmia database,” *IEEE engineering in medicine and bilology*, vol. 20, no. 3, June 2001.
[26] Goldberger, A., *et al.,* “PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals,” Circulation, vol. 101, no. 23, pp. e215–e220, July 2000.

BIOGRAPHIES OF AUTHORS

Assist. Prof. Aqeel M. Hamad Alhussainy. He obtained his BSc. Tech in Computer and Control Engineering from University of Technology Iraq 2003 and M.Sc. from Basrah University, Iraq in 2010 and he is Ph.D. student now at Al-Nahrain University, Iraq. He is having 10 years of teaching and research experience worked as a head for computer center in ThiQar University for three years. He is presently working as Assistant Professor in ThiQar University, College of Computer and Mathematics.

Assist. Prof. Dr. Ammar D. Jasim. He obtained his BSc.Tech in Computer from University of Al-Nahrain, Iraq 1999 and M.Sc from Al-Nahrain University, Iraq in 2002 and Ph.D from Al-Nahrain University, Iraq. He is having 20 years of teaching and research experience worked as a lecturer in College of Information Engineering, Al-Nahrain University.