Periodicals of Engineering and Natural Sciences ISSN 2303-4521
Vol. 9, No. 3, August 2021, pp.652-661

A novel pooling method for CNN model based on discrete cosine transform

Aqeel Mohsin Hamad[1], Naofal Mohamad Hassein[2]
[1,2]University of Thi-Qar, College of computer and mathematics, Computer Department,Iraq.

ABSTRACT

Deep learning can be used to learn huge volume of data, which will be processed through hidden layers and according to the number of hidden layers ,filter size and numbers and the required computation cost is increased because of the size of raw data, this problem can be avoided by using pooling techniques, different method s are proposed to extract the basic features of the signal instead of all signal, but unfortunately this operation may introduce some noise or omission because of elimination important data from the signal. In this paper, A novel pooling method are proposed based on discrete cosine transform , this method is utilized DCT technique to reduce spatial redundancy of image by transform the spatial domain into frequency domain , which can preserve the most significant image information from the other coefficients, which represents the other details information of the image, so discard these less important coefficients. Its effect will be slight and this can reduce the eliminated information as compared with other methods. After applying DCT, we crop the most significant coefficients to be used in the reconstructed data by applying inverse DCT . then the result is combined in different methods with Max pooling and average pooling methods, this new structure can reduce the effect of discarding most important information and reduce the drawbacks of average and Max. pooling method. The results are proved that our proposed methods are outperformed some standard methods and can be used in more application.

Keywords: *Deep learning, CNN, DCT, PDCTM, SDCTM*

Corresponding Author:

Aqeel Mohsin Hamad
Computer Department, College of computer and mathematics, University of Thi-Qar
Thi-Qar, Iraq
Email: aqeelhamad@utq.edu.iq

1. Introduction

Deep learning represents very important tool for analysis of different application due to its capability for training and learning different representations of data. Data learning techniques require different number of hidden layers, which are depended on types of application and input data as well as available capabilities to complete the required work[1,2,3]. D.L learn huge volume of data from input toward output for raw input data, so according to the size of the input data, there is high complexity in computation through hidden layers , which may result in high cost in computation and may reduce the accuracy of the model analysis and this problem is increased as the input data increased , which is necessary for learning of deep learning algorithms such as CNNs [4,5,6].

The motivation to avoid this problem is to decrease the input data dimensions, different methods are performed data reduction such as Max. pooling method, and average pooling, However, these methods are simple and efficient but it may results in omission and discarding of important information , other method such as stochastic and mixed are used but also may results in artifact and aliasing due to eliminate some basic information [7].

In this paper, Anew pooling method is proposed based on DCT, this method is utilized DCT technique to reduce spatial redundancy of image by transform the spatial domain into frequency domain , which can preserve the most significant image information from the other coefficients, which represents the other details information of the image, so discard these less important coefficients [9,10]. Its effect will be slight and this can reduce the eliminated information as compared with other methods. After applying DCT, we crop the most significant coefficients to be used in the reconstructed data by applying inverse DCT ., then the result is combined in different methods with Max and average pooling methods, this new structure can reduce the effect of discarding most important information and reduce the drawbacks of max and average pooling method [11,12,13].

1.1. Literature review

Arthia Ghosh and Rama Chellappa in 2016, proposed to use DCT in image classification , they are applied DCT on the output of the convolution layer , this can extract basic features before the pooling layer, then they used threshold to extract the most important features . The suggested work is proved to reduce the convergence cost with little training epochs [14].
MatejUlicny and Rozenn Dahyot in 2017 used convolutional neural network (CNN) with DCT to classify image, which is accomplished by applying DCT to original image, then the output will be set of coefficients with different frequencies , which represents the features of the image, then these features are used as input for CNN model to be classified ,the designed filter in this work is equivalent to two times of the used window size in each dimension[15].
Xiaym et.al. proposed to use DCT as compression method for image data, which can reduce the amount of data of image through the intermediate hidden layer, they are used small set of DCT coefficients to be fed into two layer auto encoder instead of all image pixels , they proved that combining DCT with deep learning can increase the accuracy of the classification, by exploit the DCT properties, their work is accomplished by applying DCT, then auto encoder is used to generate another vector ,which is reconstructed by decoder , and the best vector is obtained by minimize the error between the original and reconstructed vector, so the auto encoder will give best result by using the best features, which are extracted by DCT [16].
Michael Mathieu and Mikael Henaff in 2014 proposed to use fast Fourier transform (FFT) to accelerate the training and inference by determine the convolution in point wise manner in Fourier transform domain and this is achieved improvement in performance of training compared with existing methods. This work is accomplished by determine the FFT for kernels and input in parallel manner , then multiplying the results of FFTs to determine features , which will be inversed by IFFT [17].
Yahao Xu. And Hideki Nakayama in 2019, proposed a pooling layer method based on using DCT for processing information from convolution layer. They used DCT method to isolate the basic features of the input signal because it have isolation property and compaction .It can preserve better information than DFT method. They proved that their work is better than exiting work depending on their results with negligible time consumption 18].
Matej Ulicay et al. in 2020 proposed to use CNN to learn features extracted from DCT, this work is achieved by first apply DCT, which have good compaction property , as used in image compression .They are demonstrate that, the harmonic networks can be compressed efficiently by discarding high frequency information because of redundancy in spatial domain, also they proved from their results, that this model can enable compression of data in the performance[19].

2. Materials and Methods

In this paper, A novel pooling methods are proposed based on Discrete cosine transform (DCT), which is used to select the most important features of the output of convolution layer, this method can decrease the amount of the discarded information during pooling method as in standard max pooling or average method, which are eliminated some of details information from the signal. The proposed methods are shown in Figure 1 [19,20].

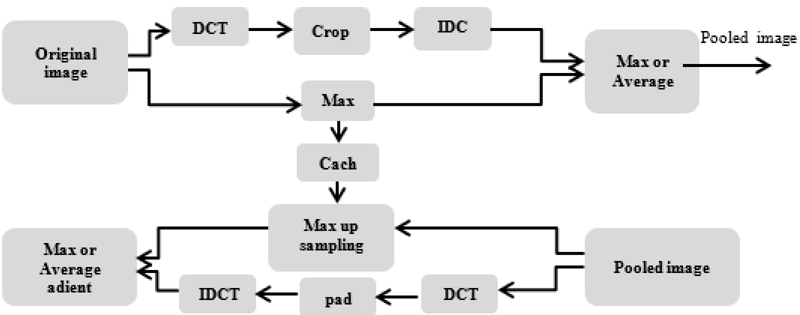

Figure 1. Proposed pooling layer

The original signal is applied to DCT, then the basic frequencies, which are represented basics information of the signal are selected, while the details coefficients, which are used to represent less significant information are eliminated , the cropping is depended on the pool size, then inverse DCT transform is used to reconstruct the signal with down sampling size , the results are combined with Max. and average in different procedures as described in the following sections.

2.1. Discrete cosine transform (DCT)

Discrete Cosine Transform (DCT) is used to separate the images into different frequencies parts, so the information of the images is separated according to their important, the most important information is treated separately, while the details are processed in lossy manner as required in some application such as compression, so DCT can be used in image processing application, it is used cosine function to determine the different frequencies as shown in eq.(1) :[20,21,22,23]

$$\mathrm{D(i,j)} = \frac{1}{2\sqrt{N}} C(i)C(j) \sum_{x=0}^{N-1} \sum_{x=0}^{N-1} (P(x,y)) \cos[\frac{(2x+1)\, i\pi}{2N}] \cos[\frac{(2y+1)\, i\pi}{2N}] --(1)$$

Where C(i) is determined by eq.(2):

$$\mathrm{C(i)} = \begin{cases} \frac{1}{\sqrt{2}} & if \;\; i = 0 \\ 1 & if \;\; i > 0 \end{cases} \quad ----------------------(2)$$

N is the size of the image block.

3-1 Proposed algorithms

Three different pooling methods are proposed based on DCT, which are used to select most significant features of the signal, by extract the most important frequencies after applying DCT, which represents most information of the signal , then IDCT is applied to reconstruct the signal, these signal is combined in different methods with average and Max. methods as described in the following sections.

2.2. Parallel DCT-Max method (PDCTM)

This method is used to extract the pooled signal by applying DCT to the output of the convolution layer , then select the most important features from DCT coefficients according to the required pooling size and eliminate the other coefficients , then applying inverse DCT, at the same time max pooling method is used by select the maximum element for each window of pooling size, then the pooled signal will be determined by combining the two results , there are two procedure to determine pooled signal , the first method which is named as (PDCTM1) is determined the maximum values between the (DCT, cropping and inverse IDCT) and the extracted features from max method, while the second method (PDCTM2) is selected the pooled features based on applying (DCT, cropping and IDCT) on the original and the features ,which is extracted by max pooling with stride on , the detail descriptions for (PDCTM1 and PDCTM2) are shown in Algorithm I and Algorithm II.

Algorithm I (PDCTM)
Input =$X\ Input\ Raw\ signal$;
Output=Y Extracted Features

```
Forward Propagation
Initialization
Sg: signal size ,  P pool size of pool; S : size of stride
Determine size of pooled signal:
HH=
WW=
While (n in range N)                  % loop to number  of data
  While (c in range C)                %loop to the number of channel
      Do { im=X(:,:,n,c)              %extract  image from patch
        While (1 in range HH)         %loop for height of image
           While (j in range WW)      %loop for width of image
             Determine  maximum of window
                                                  Out1=Max
           End
        End
     Determine coef=DCT(im);          % apply DCT
     BF=coef(1: SS);                  %Crop the basic coefficients
     Dc=zero;                         %delete details coefficients
     Rc=BF+Dc
    Out2=IDCT(Rc);                    %determine invers DCT
   Y(n,c)=max(out1,out2)
  End
End
Pooled signal =Y;
```

Algorithm II (PDCTM2)
Input =X $input$ Raw $signal$;
Output=Y extracted Features

```
Forward Propagation
Initialization
Sg: signal size , P pool size of pool; S : size of stride
                                   Determine size of pooled signal:
HH=
WW=
While (n in range N)            % loop to number of data
  While (c in range C)          %loop to the number of channel
    Do { im=X(:,:,n,c)          %extract image from patch
      While (1 in range HH)     %loop for height of image
        While (j in range WW)   %loop for width of image
          Max=maximum of window
          Out1= Determine DCT(Max);
          BF1=Out1(1: SS);               %Crop the basic coefficients
        End
      End
 Determine coef=DCT(im);        % apply DCT
BF2=coef(1: SS);                %Crop the basic coefficients
BF=Max(BF1,BF2)
 Dc=zero;                       %delete details coefficients
Rc=BF+Dc
Out2=IDCT(Rc);                  %determine invers DCT
Pooled signal =Max(out1,out2);
End
```

2.3. Serial DCT-Max method (SDCTM)

This technique is used the (DCT, cropping and IDCT) in serial manner (cascade) with max. pooling method ,so it is named as (SDCTM). At first, the DCT is applied on the signal to transform the signal into frequency domain, then according to priority of significant, the coefficients with less significant is replaced by zero value, while the most important coefficients are cropped according to pool size, then inverse of DCT is applied to reconstruct the signal, then, Max. pooling method is applied to extract the pooled signal , the details are described in Algorithm III .

Algorithm III (SDCTM1)
Input =X input Raw signal ;
Output=Y extracted Features

```
Initialization
Sg: signal size , P pool size of pool; S : size of stride
Determine size of pooled signal:
HH=
WW=
While (n in range N)            % loop to number of data
  While (c in range C)          %loop to the number of channel
    Do { im=X(:,:,n,c)          %extract image from patch
      While (1 in range HH)     %loop for height of image
        While (j in range WW)   %loop for width of image
          Max=maximum of window(stride =1)
           End
      End
      Out1= Determine DCT(Max);
      BF=Out1(1: SS);               %Crop the basic coefficients
      Dc=zero;                      %delete details coefficients
      Rc=BF+Dc
      Out2=IDCT(Rc);                %determine invers DCT
   Y(n,c,:,)=out2;
  End
```

3. Results and discussions

The proposed method are tested on two image with different characteristics(standard Lena and Barbara image), and the results are evaluated based on different quality metrics (SNR and PSNR) and compared with standard max. and average pooling method, then the proposed method are used to design pooling layer, which is used for deep learning CNN model and they are tested by using two different types of datasets, these dataset are two dimensional signal (image), which are MNIST dataset, in it, the input signal is gray scale image with size (28*28), while the second dataset is CIFAR10 dataset with size 32*32.The experiments are performed by matlab2019 with CPU 2.7GHz,intel cori72400with RAM 8GB windows7.

3.1. Results of test images

The results of applying the proposed method on "Lena" and "Barbara" images are shown in Table (1), which is described the improvement of SNR and PSNR metrics , for "Lena" image, (PDCTM1) method is achieved (26.4987) and (32.168dB) in terms of SNR and PSNR respectively compared with (19.013 and 24.28dB) for Max pooling method , also it is achieved good improvement for "Barbara" image, since it is satisfied (19.06 in term of SNR verses 16.06 for max, and in term of PSNR, it is achieved 26.19 dB verses 21.69 dB) in max pooling, which shows that our proposed methods are extracted the most significant information of the image to be the pooled signal. The quality of the pooled signal is shown in figure (2) , which shows the little differences between original and reconstructed image.

Table 1. Performance measures for proposed methods

Images	**Metrics**	**MAX**	**PDCTM1**	**PDCTM1**	**SDCTM**
Lena	SNR	19.013	26.4987	23.7345	20.283
	PSNR in dB	24.28	32.168	29.2189	26.283
Barbara	SNR	16.09	19.7908	18.5145	18.1539
	PSNR in dB	21.69	26.1956	24.09	24.09

Original image (256*256) Pooled image (128*128) Reconstructed image (256*256)

Figure 2. Original, pooled and reconstructed image by PDCTM1method

3.2. Results of MNIST dataset

The first standard dataset is MNIST data set, which have gray scale image with size (28*28)[24], at first, this dataset is divided into training dataset(4800) image and test image 1200,then the CNN model is used to training this dataset. The training is performed based on 10epochs, iteration with hard ware specified in previous section, and the results of our work are compared with standard max and average pooling method in terms of accuracy, sensitivity, recall and error rate as shown in Table 2, it is clear that the proposed method (PDCTM1) gives(99.80%) accuracy better results than Max.(98.88%) and average (98.82%). Method (SDCTM2)gives accuracy (99.52%), because it is depended on determining the maximum value of max method and DCT method with value, because the MNIST dataset is gray scale image with almost black background , so maximum value may provide the most significant information , also by using DCT , we can extract the best feature of the image.

Table 2. Performance of the proposed methods

Measure	Max	Average	PDCTM1	PDCTM2	SDCTM1
Accuracy (Acc.%)	98.88	98.82	99.80	99.52	99.64
Specificity (Sp.%)	98.98	98.82	99.96	99.62	99.68

Measure	Max	Average	PDCTM1	PDCTM2	SDCTM1
Precision (Pr. %)	98.88	98.82	99.80	99.52	99.64
Sensitivity(Sn.%)	98.92	98.82	99.96	99.62	99.68
Fscore (F%)	98.88	98.82	99.80	99.52	99.64

Figure 3 shows the training progress of accuracy for the proposed methods, it is clear from the figure that , (SDCTM2)method is achieved more than 98% accuracy in less than two epochs while other methods are achieved this accuracy at approximately four epochs, this indicates that (SDCTM2)method is provided most effected features, which are increased the fitting of the model early. The confusion matrix for this method is given in Table 3, which is provided the matching between predicted and actual classes of the classification.

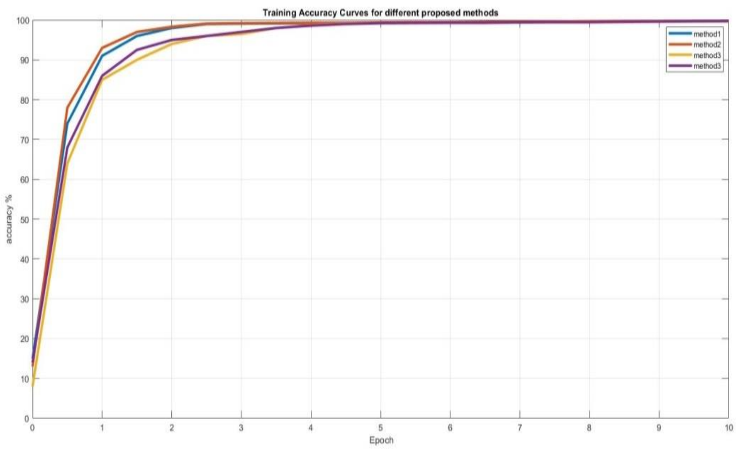

Figure 3. Training progress of accuracy for the proposed methods.

Table 3. performance of the (SDCTM1) method.

Confusion matrix	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9
Class 0	**250**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Class 1	**0**	**250**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Class 2	**0**	**0**	**250**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Class 3	**0**	**0**	**0**	**249**	**0**	**1**	**0**	**0**	**0**	**0**
Class 4	**0**	**0**	**0**	**0**	**250**	**0**	**0**	**0**	**0**	**0**
Class 5	**0**	**0**	**0**	**1**	**0**	**249**	**0**	**0**	**0**	**0**
Class 6	**0**	**0**	**0**	**0**	**0**	**0**	**250**	**0**	**0**	**0**
Class 7	**0**	**0**	**2**	**0**	**0**	**0**	**0**	**250**	**0**	**0**
Class 8	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**250**	**0**
Class 9	**0**	**0**	**1**	**0**	**0**	**1**	**0**	**0**	**0**	**250**

3.3. Results of CIFAR10 dataset

This dataset is color image with size 32*32,which is used for training model to classify the objects to ten classes , these images have different characteristics[25] , in this dataset, there are (5000)images are used for training, while the remaining images (1000) are used for testing the model. The model is trained with the same parameters for all methods and it is compared with other methods with this parameters. Table 4 shows the performance of the proposed methods for this dataset. Method (PDCTM1) is achieved the highest results in term of accuracy (73.98%) compared with max (72.2) and average (72.4%), this improvement is obtained due to best feature extraction, which is extracted in both (DCT, cropping the most important frequencies and inverse DCT with max. pooling method).

Table 4. Performance of the proposed methods for CIFAR10 dataset

Measure	Max	Average	PDCTM1	PDCTM2	SDCTM1
Accuracy (Acc.%)	72.2	72.4	73.98	73.75	73.42
Recall (re.%)	72.27	72.4	73.49	73.75	74.42
Precision (Pr. %)	26.6	27.27	73.41	73.87	73.72
Sensitivity(Sn.%)	72.29	72.33	73.98	73.75	73.42
Fscore (F%)	27.8	27.3	73.33	73.54	73.55

The performance of the proposed methods by using different mini-batch size are shown in Table 5 and it is shown in figure (4) , the best results in terms of accuracy, sensitivity and precision are obtained by using batch-size 128, which are (73.98,74.01 and 74.01) for the above metrics respectively, because low batch size may results in oscillation due to repeated tuning , while high batch size may results in over fitting, so best results are satisfied with 128 batch size as shown in figure 3.

Table 5. Performance of the proposed method with different min-batch size

Min-batch size	accuracy	sensitivity	precision
32	71.7	72.19	71.82
64	73.06	73.35	73.22
128	73.98	74.2	74.01
256	73.23	73.43	73.25
512	72.88	73.41	73.05

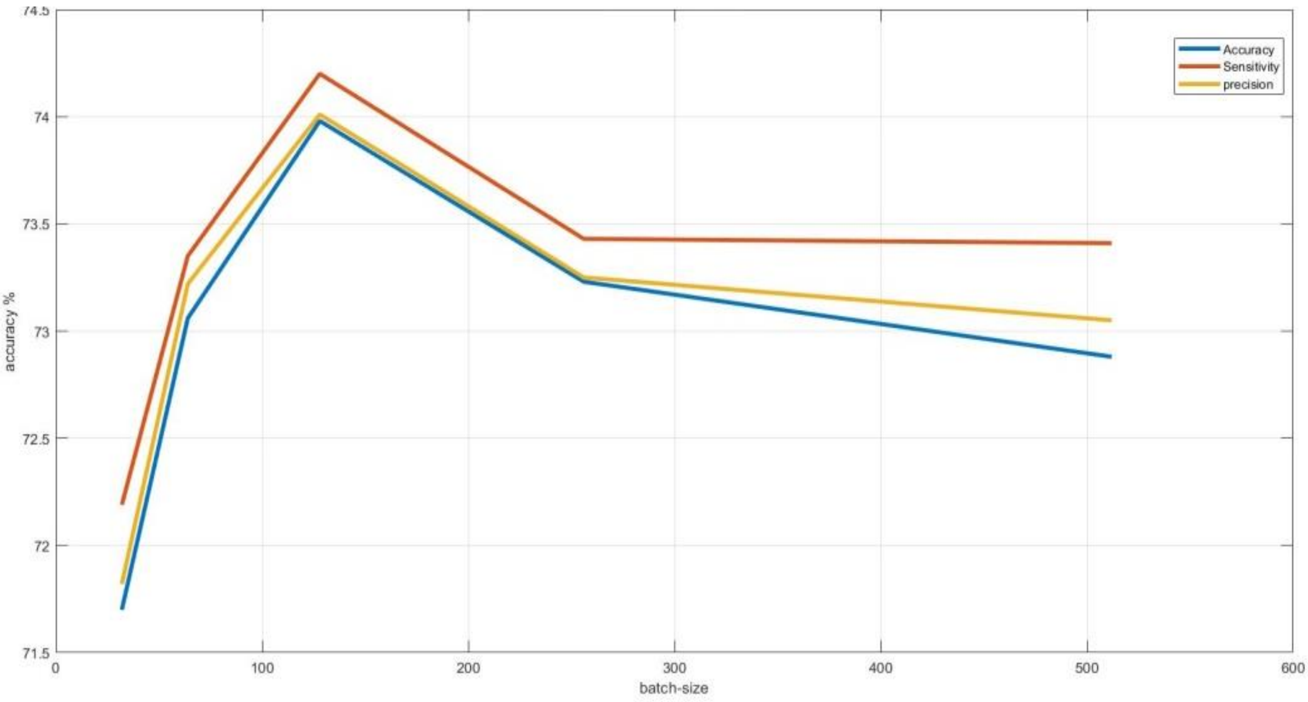

Figure 4. Performance measures for different batch-size for proposed methods.

The confusion matrix is shown in Table 6 , which shows good matching between the predicted and the actual classes in most of the classes, also, it is clear from the table that all category of the table is satisfied matching with approximately equivalent matching ratio, this proves that our proposed method are more efficient . The training curve for accuracy and loss are shown in figure (5), it is clear that the accuracy reached to near 60% in approximately one epochs , then increased slowly until achieved (73.98%), also the loss is decreased to less than one in two epochs then it is decreased gradually until it is reached to less than 0.7.

Table 6. Performance of the proposed methods

Confusion matrix

class	**Airplane**	**Automobile**	**Bird**	**Cat**	**Deer**	**Dog**	**Frog**	**Hors**	**Ship**	**truck**
Airplane	**797**	**29**	**69**	**33**	**25**	**12**	**10**	**23**	**57**	**35**
Automobile	**15**	**826**	**7**	**8**	**4**	**2**	**4**	**3**	**15**	**76**
Bird	**37**	**10**	**599**	**60**	**69**	**42**	**36**	**32**	**13**	**9**
Cat	**17**	**3**	**61**	**533**	**36**	**164**	**57**	**46**	**10**	**12**
Dear	**12**	**3**	**76**	**60**	**707**	**37**	**26**	**55**	**6**	**7**
Dog	**6**	**4**	**108**	**201**	**55**	**694**	**22**	**97**	**5**	**2**
Frog	**9**	**10**	**46**	**66**	**55**	**13**	**839**	**6**	**7**	**12**
Hors	**10**	**1**	**17**	**12**	**37**	**25**	**2**	**724**	**2**	**5**
Ship	**63**	**36**	**9**	**15**	**9**	**6**	**3**	**6**	**866**	**34**
truck	**34**	**75**	**8**	**12**	**3**	**5**	**1**	**8**	**17**	**808**

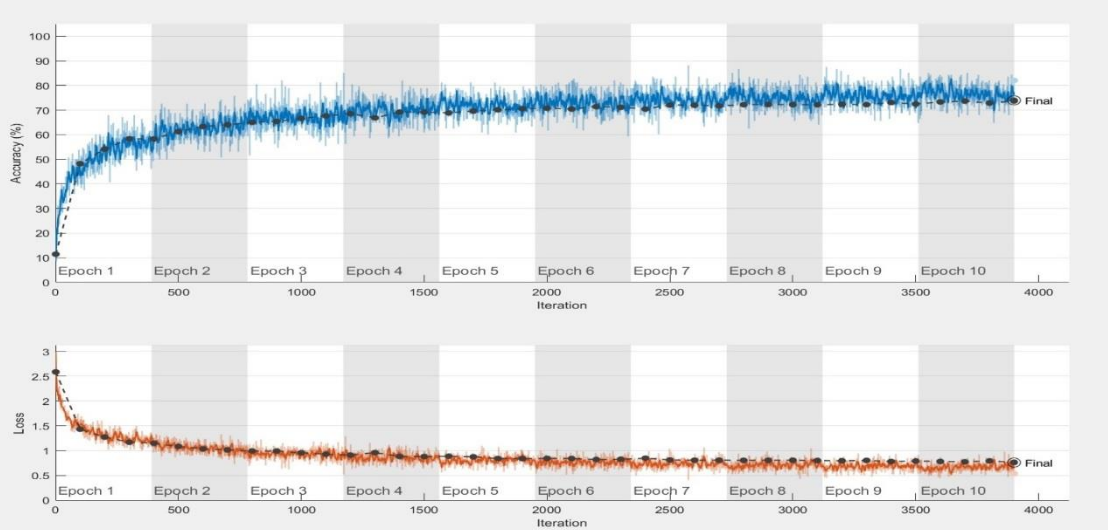

Figure 5. Training progress for accuracy and loss of proposed PDCTM1 method

4. Conclusion

D.L. learn huge volume of data through the CNN layers, so according to the size of the input data, there is high complexity in computation through hidden layers , which may result in high cost in computation and may reduce the accuracy of the model analysis and this problem is increased as the input data increased . One of the most significant solution to this problem is using pooling layer, which can be used to solve the problem of over fitting and reduce the data volume through training. In this paper, We have proposed a new pooling method based on DCT, this method is utilized DCT technique to reduce spatial redundancy of image by transform the spatial domain into frequency domain , which can preserve the most significant image information from the other coefficients, which represents the other details information of the image, so the effect of discarding such information will be slight and can reduce the eliminated information as compared with other methods. After applying DCT, We are cropped the most significant coefficients to be used in the reconstructed data by applying inverse DCT , then the result is combined in different methods with Max and average pooling methods, this new structure can reduce the effect of discarding most important information and reduce the drawbacks of max and average pooling method. We have design three pooling methods , which are PDCTM1,PDCT2 and SDCTM depending on the structure of the of combining of DCT with max pooling method. The proposed methods are tested on different images and compared with results of average and Max. pooling methods, the results show that proposed methods PDCTM1 achieved (PSNR =32.168 dB and SNR=26.49) for standard Lena image verse (PSNR=24.28dB and SNR =19.03)for Max. pooling method, also this pooling method is used in CNN model and performed on two types of datasets and the results shows that (PDCTM1) method is achieved better results in terms of accuracy (99.80%) than standard max (98.88 %). Also from results, it is clear that (PDCTM1) method can be reached to more than (98.5%) with less than (two epochs) .For CIFAR10 dataset, method(PDCTM1) is achieved (73.98%) with sensitivity (73.98%) better than max (72.2%) and (72.9%) for accuracy and sensitivity. The proposed methods are outperformed the standard pooling method and can be used for most of classification application.

References

[1]Christian S, Wei Liu , Chapel H., Pierre S., Dumitru E., Vincent V. , Andrew R., " Going Deeper with Convolutions", IEEE explorer, arXiv preprint arXiv:1409.4842, 2014.

[2]Krizhevsky, Sutskever, and Hinton, "ImageNet classification with deep convolutional neural networks", NIPS, Vol. 1 , 2012.

[3]Kaiming H., Xiangyu Z., Shaoqing R., Jian S. "Deep residual learning for image recognition", CVPR, pp. 770–778, 2016.

[4]Y-Lan B.Jean P.,JEAN.P., "A Theoretical Analysis of Feature Pooling in Visual Recognition", Proceedings of the 27th International Conferenceon Machine Learning, ICML, 2010.

[5]Forest.A, Matthew H., Peter S. and Pierre B.,"Learning Activation Functions to Improve Deep Neural Networks", International Conference on Learning Representations (ICLR), 2015.

[6]Andrew G. Howard , Menglon Z., Bo C. and Dmitry K., "Efficient convolutional neural networks for mobile vision applications". arXiv, 2017.

[7]Jawad N., "Max-Pooling Convolutional Neural Networks for Vision-based Hand Gesture Recognition", IEEE International Conference on Signal and Image Processing Applications, ICSIPA 2011, Kuala Lumpur, Malaysia, 2011.

[8]Wan A. M., Haniza Y., Wan K. and Mohd A. G. "Image Enhancement Based on Discrete Cosine Transforms (DCT) and Discrete Wavelet Transform (DWT): A Review", IOP Conferences Series: Materials Science and Engineering , 2019.

[9]Takumi Kobayashi , "Global Feature Guided Local Pooling", Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 3365-3374,2019.

[10] Matej U., Valdimir A.K. and Rozeen D, "Harmonic Convolutional Networks based on Discrete Cosine Transform", arXiv:2001.06570v1 [cs.CV] , 2020.

[11] Tsai S. and Yang S. "A Fast DCT Algorithm for Watermarking in Digital Signal Processor", Hindawi, Mathematical Problems in Engineering, Vol.1, 2017.

[12] Muhammad T. and Abhijit M., "A method for compressed representationand learning in CNNs", . arXiv preprint arXiv, 2019.

[13] Qiang Q., Xiuyuan C., Robert C., and Guillermo S., " Deep neural network with decomposed convolutional filters." 35th International Conference on Machine Learning, Vol. 80, pp. 4198-4207,PMLR, 2018.

[14] Arthita G. and Rama C., "Deep Feature Extraction in the DCT domain", 23rd International Conference on Pattern Recognition (ICPR),2016.

[15] Matej U.& Rozenn D., "On using CNN with DCT based Image Data", Proceedings of the 19th Irish Machine Vision and Image Processing conference, IMVIP, 2017.

[16] Xioayi Z. , Xiangmin X., Chunmei Q. and Xiaofen X. “High speed deep networks based on discrete cosine transformation”, Image Processing (ICIP), 2014 IEEE International Conference on image processing, pp. 5921-5925, 2014.
[17] Michael M, Mikael H. and Yann L., “ Fast Training of Convolutional Networks through FFTs”, arXiv preprint arXiv,2014.
[18] YahaoXu. And Hideki N., “DCT Based Information-Preserving Pooling for Deep Neural Networks”, (ICIP), Taipei, Taiwan ,2019.
[19] Matej U., Vladimir A. Krylov, and Rozenn D. , “Harmonic Convolutional Networks based on Discrete Cosine Transform”, arXiv:2001.06570v1 [cs.CV] 18 Jan 2020.
[20] Naidu V., “Discrete Cosine Transform based Image Fusion Techniques”, Journal of Communication Navigation and Signal Processing , Vol. 1, No. 1, pp. 35-45 ,2012
[21] Vassil D. and Khan W., “Multiplier less DCT Algorithm for Image Compression Applications”, .International Journal on Information Theories and Applications, Vol.11, pp.162-169, 2004.
[22] Raid A., Khedr W and Wesam A., “JPEG Image Compression Using Discrete Cosine Transform : A Survey”, Nternational Journal of Computer Science & Engineering Survey , Vol.5, No.2, 2014.
[23] Liu C.and Hang H., “Acceleration and Implementation of JPEG 2000 Encoder on TI DSP platform”, Image Processing, IEEE International Conference on image processing, Vo1. 3, pp. 329-339, 2005 .
[24] Yann L., Corinna C. and Chirstopher J., “ The MINSTdatabase for hand wright digits”, http://yann.lecun.com/exdb/mnist/, 2005.
[25] Alex K., Vinod N., and Geoffrey H.,” The CIFAR-10 dataset”, https://www.cs.toronto.edu/~kriz/cifar.html