RESEARCH ARTICLE OPEN ACCESS

Residual Network with LBP for Image Classification

Y. Mani Kumar[1] N. Udaya Kumar[2] B. Tapasvi[3]
P.G Student Professor Assistant Professor
Department of E.C.E, SRKR ENGINEERING COLLEGE (Autonomous), Bhimavaram, A.P-India.

ABSTRACT
Deep neural networks work with artificial intelligence makes system more efficient with existence of different architectures. In this work a unique deep-learning based methodology named Residual Network with LBP, is proposed to efficiently extract and compare hierarchical data over complete features in multilayer perceptron (MLP). The LBP Net retains the same topology of Residual Network (ResNet). This method is briefly discussed, which mainly analyses the edge operation, the uniform pattern and rotation invariant pattern of learnable linear weights. This enables the LBP Net to realize a high recognition accuracy without requiring any costly model learning approach on massive data in texture analysis, classification, face analysis and recognition, and other detection applications are reviewed.

In this scenario, as a base content most popular convolution neural networks for object detection and object category classification from images are Alex Nets, Google Net, and ResNet50. A variety of image data sets are available to test the performance of different types of CNN's. The commonly found benchmark datasets for evaluating the performance of a convolutional neural network are an ImageNet dataset, and CIFAR10, CIFAR100, and MNIST image data sets. This analysis shows that Google Net and ResNet50 are able to recognize objects with better precision compared to Alex Net. Moreover, the performance of trained CNN's varies substantially across different categories of objects.

Keywords: Convolutional Neural Network (CNN), Local Binary Pattern (LBP), MLP, ResNet.

Date of Submission: 29-08-2020 Date of Acceptance: 14-09-2020

I. INTRODUCTION

Machine learning has been very successful in different type of applications, such as image recognition, data sensing, object recognition, machine translation, speech processing and various other classifications. Deep learning is well-known as a technique to extract hierarchical representations of knowledge [1]. The designed architecture can perform efficiently through the internal connections laid between multiple layers without any complexity. The later layers can extract the high-level information from the lower level layers which are also called earlier layers. That means low-level features helps to extract high-level information from the later layers. Convolutional Neural Network (CNN) is most commonly studied deep learning architectures. Deep Convolution Neural Network has been very successful in pattern recognition and classification with various improvements in the architectures. However, training these networks end to end is very expensive and also complicated which results in large size in memory usage and disk space which results in problems like overfitting due to various number of parameters included in it for a successful architectural design. The solution for the above can be achieved by using GPU's for maximum performance. Such are set to diminish by several binary comparisons of CNNs [6,10,15]. Through computing binary values with CNNs leads to some loss in performance when compared to the real values of the neural network weights.

In this paper, a simple Residual Network with the help of Local Binary pattern (LBP) for Image Classification is proposed. In this work "Residual Network with LBP for Image Classification" the weights of the neural network, the filter size and pixels within the image are considered. Deep learning reduces to optimize the linear weights with optimizing convolutional filters. Where the image filter size is set to 3×3 for extracting accurate LBP code. Here in this LBP network thus follows 3 parameters such as center find values, radius and neighborhood value pixel sizes. Features are extracted from the dense layers thus the earlier layers extract high-level features from the initial layers as said before. Here using unsupervised learning algorithms where the LBP Net is capable of performing unsupervised learning on data. Since LBP Net comprises of all the characteristics of the machine learning architectures it is further simplified with the deep neural network. The proposed Network has the key features of the ResNet and CNN but it simplifies the expensive training approach which thus also makes the system more vulnerable when compared to other approaches and it can also outperform them.

II. RELATED WORK

Here, in this work we explain about the Local Binary patterns (LBP) with deep learning architectures and state-of-art in recognition and classification of images. In this work we use local binary patterns on of the most significant use in image recognition and classification. It can also be applied to various other works. Motivated by different architectures and various methods to classify and detect objects and images we took LBP with deep learning architectures for classification purpose [3,4]. The LBP works by thresholding the intensity values of the neighboring pixel. This computing is done by the center pixel with every neighboring pixel. A similar neighboring pixel is developed by this, which are not exactly at the center of a pixel. In this work LBP, the center-find values of every pixel, its radius and neighboring pixel grids are calculated. In deep Learning Several methods in which CNNs are most generally used for classification purposes, these CNNs are well trained with large amount of testing and training labeled datasets. Generally, it starts with RGB images with fixed values given to a set of convolutions followed by normalization and pooling techniques [5]. It normally ends with several fully connected layers which is used for extraction purposes. There are several methods to further modify the deep learning architectures which includes increasing of depth and addition of more convolutions [8,9]. CNNs are widely used in deep learning architectures for classification. As we stated above LBP is very keen in image classification with some changes using ResNet, deep learning architectures. This work describes a deep layered network with Multilayer Perceptron (MLP) with the use of Local Binary Pattern and convolutional layers. Where, LBP makes the system more efficient despite with the efficiency of the deep network architectures in many applications for image classification and recognition. This can be tested with state-of-art datasets like Cifar-10, Cifar-100 and ImageNet by giving RGB images as source. To outperform the bench mark error rate and efficiency we first apply the LBP method to the given datasets which contain the RGB images. Then the data sets with the RGB images are converted into the LBP coded images which are further given as datasets to the training and testing processes. In this conversion of RGB images to the LBP coded images the weights are further adjusted, and the pixel values changes accordingly as stated above i.e., it works by thresholding the intensity values of the neighboring pixel with the center pixel.

III. RESNET WITH LBP

The section in this work refers to the deep learning architecture that is accumulated with the LBP features to enhance the images for classification. Here, ResNet with LBP can be classified into two parts: the feature extraction using the LBP code generated and network classification with the deep learning architectures.

III.I. RESIDUAL BLOCK

Let us take a simple residual network with addition to the convolutional neural networks, which gives more efficiency in training "deep learning" models [11]. This can be taken further as different variants of the architecture are being tested. These can be performed using the neural networks and are compared with past and later experiments. These are preceded by normalization operations, ReLu activation functions and are achieving better results with the bench mark datasets like cifar-10, cifar-100 [13]. This model where LBP is being implemented with the residual blocks where the normal training is done using the weight layer that is, convolutional layer and feed forward to the activation functions. The weights are added up with the LBP code generated and a new base weight is formed with in the given image and the pixel values change simultaneously. The connectivity between the input layers to the processing layer and directly to the output layer is easily accomplished. The weight layer can negotiate the undesirable activations that are being transformed during the process as shown

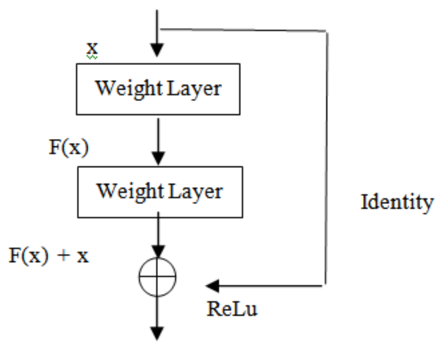

Fig-1: Residual Block

Here, F(x) is the output of the function and the x is the general input taken with the weight function. It performs the normalization, activation and the weight (conv 3×3) operations. The convolutional operations thus take place with the system to effectively generate the weights and also pooling can be taken for up or down sampling the smaller and larger branches as required respectively with convolution operations by thresholding function. Here, a simple network is implemented in an effective way to simplify the complexity problems. The architecture of this system is as follows with a randomly sampled 224×224 crop augmented RGB image applied with a scale to obtain a 32×32 pixel input size, the convolution layer or the weight layer with 3×3 input, followed by pooling layer and further the normalization and activation functions thus again followed by the convolutional layers and it is average pooled at the bottom to retract the original features given at the input, so called the feed forward system design and are directly mapped to the output layers. The overall process of this network is to classify a smaller scale objects effectively because the larger the size more the complexity and higher the parameters involving in it. The resulted values are given to the entropy function and are further compared with the LBP features and add up to the output layers. Thus, by classifying the results with the incorrect classes to the correct classes by minimizing the error rate. Based on the network we make shortcut connections which make the network more efficient. They can be used only when both the input and the output dimensions are same. For these the shortcut connections go across for two sizes and can be adjusted by the strides. To better understand the network the intermediate values are taken with ease and the mean absolute values from the output layers are calculated. These methods gave a good formulation and optimization can be further more simplified. Here, in this model the architecture plays a vital role in the computational processes. In this architectural frame work the ease of training the deeper networks are presented which can perform deeper than the previous analysis.

Network Architecture

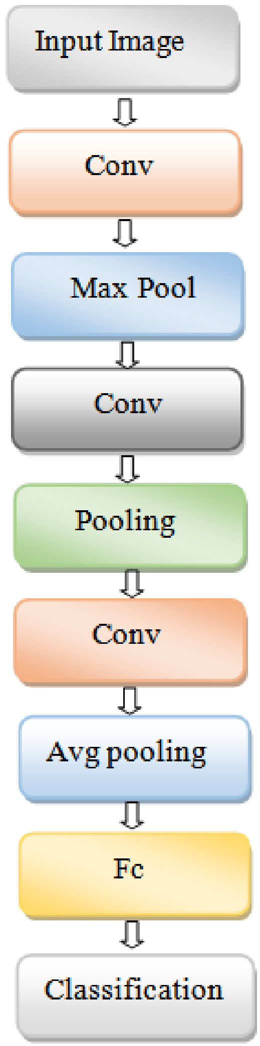

Fig-2: Final Architecture

III.II. LOCAL BINARY PATTERN

LBP features encodes local texture Information. It is used for classification, Detection and Recognition. It partitions the input image into non-overlapping cells. It is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood values of each pixel and consider the result as a binary value. It further calculates the values through image processing with histogram technique. The LBP selects a 3×3 window from the entire image and it will extract an LBP code for the selected 3×3 window, likewise it considers 3×3 windows simultaneously for the entire image and extracts the LBP code for the whole image.

$$\text{LBP} = \sum_{n}^{n-1} S(i_n - i_c)\, 2^n$$

Where i_c is the center find pixel and i_n is the neighbor pixel value and thresholding function can be defined as:

$$S(x) = \begin{cases} 1, & if\ x \geq 0; \\ 0, & otherwise. \end{cases}$$

Here, we see a 3 × 3 window with the pixel values taken from an RGB image, it can be calculated by the above formula and the value obtained after thresholding is replaced with the original value and it does for the entire image and the original values are replaced by the new pixel values. The basic idea for developing this LBP operator is that it can calculate efficiently by two-dimensional surface textures, which are Local spatial Patterns and gray scale contrast of an image.

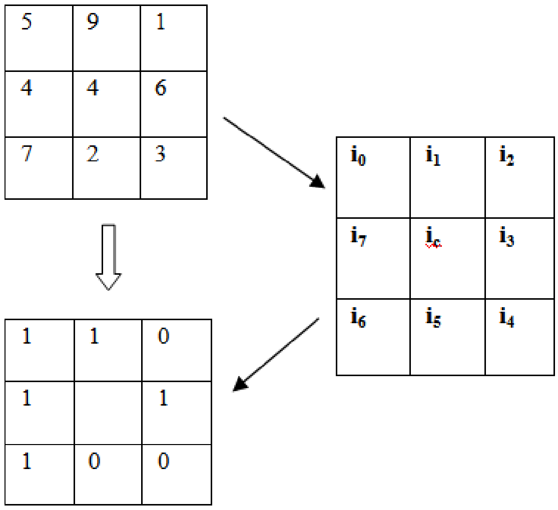

Fig-3: LBP Coding

It thresholds the 3 × 3 neighborhood of each and every pixel with the center value and only considers the result as a binary value. The histogram of these 2^8 = 256 different labels can be used as a texture descriptor. A Local Binary Pattern is called as uniform if the binary pattern consists at most 2 bitwise transitions from 0 to 1 or vice-versa when it is traversed circularly. In this computation process, the uniform patterns are used so that there is a separate label for every uniform pattern and remaining all the non-uniform patterns are labeled as single label [7]. The LBP code for an image can be computed by the center pixel is described in the above figure. The generated LBP code is [1 1 0 1 0 0 1 1], it is multiplied by powers of 2 and summing the results we get,

$1 \times 2^0 + 1 \times 2^1 + 1 \times 2^4 + 1 \times 2^6 + 1 \times 2^7 = 211.$

This process is computed across the whole image to extract the LBP code for an entire image. The generated LBP feature values are generally 0 (lower threshold) or 1 (upper threshold). These are successfully implemented to detect the flat areas, corners, curves, edges and edge ends. These LBP codes are then pooled as histogram representations as a source to the classifier, as we know there is a huge success in the deep learning here these codes are fused with the deep network architectures [2] for efficient classification.

III.III. LBP FILTERS

The LBP filters are formed from the LBP operator [12] it labels the pixel in the image by thresholding its neighbor pixel to the center pixel and takes them into a single binary value. These pixel values are obtained as discussed above can be given by

$$\sum_{n}^{n-1} S(i_n - i_c)\ 2^n$$

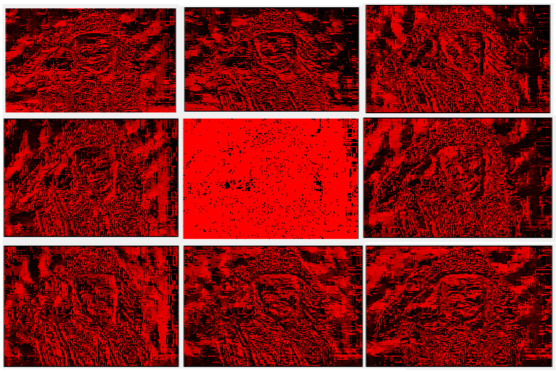

Fig-4: LBP Features

For an LBP feature to be extracted first they are filtered with the convolution operation followed by a set of non-linear operations. Later these features maps are combined to give an LBP feature. This can be taken as input to the other layers for future processing. Neural networks driven with the set of convolutional operations and LBP feature extraction using the ResNet architecture. It consists of a set of convolutional layers with learnable and non-learnable weights. The learnable features are less than the normal convolutional layer with the same size of convolutional operations with number of inputs and outputs. This reduces the dimensions of the image and also improves the decoding abilities. Its computation is based upon the features generated by the LBP of same size with the similar connectivity of the convolutional layer to further simplify the computation level and for an easier classification. In the recognition process generally the change of illumination is absorbed by highly variable images. These can be minimized by increasing the weaker sections and allowing it to circulate uniformly throughout the data.

IV. EXPERIMENTAL ANALYSIS

In this experiment, the performance of the proposed network is trained by using bench mark datasets Cifar-10 and Ciar-100. And the comparison table of the two are given below for future purposes. In this experiment first the performance of the system is analyzed and then the noise factor is compared with the other set of experiments with various deep learning architectures. The data sets consist of a 32×32 color images of about 60,000 images with 50,000 training images and 10,000 test images which include 100 classes. various experiments are conducted on this network to effectively achieve the bench mark results. This LBP can be very easily implemented using any dep learning network. The convolutional weights are fixed and only the LBP weights are updated and thus complexity in deep network is reduced in this scenario.

In the training process the experiments are trained on the training set and are evaluated under test set. Likewise conducting several experiments on this deep network using bench mark data sets like Cifar-10 and Cifar-100 to list out various behaviors of the deep network but not comparing it with other state of art results.

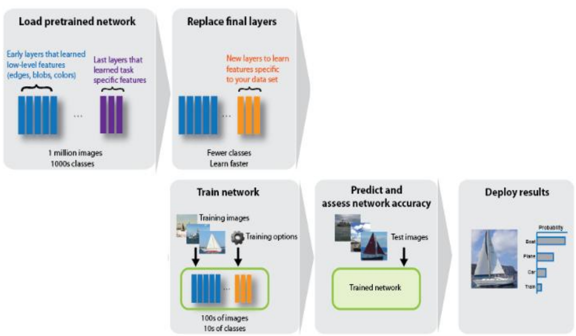

Fig-5: Training Process

By using a simpler network which is efficient enough and also of less complexity with low computational time and without latency. This is done using MATLAB simulation. The training is performed on a system having GPU with 8GB of RAM and CPU with 2GHz clock speed. This network takes a randomly sampled and augmented 224×224 RGB to obtain an image of input size 32×32, with 5×5 convolutional operations taking place. As the LBP uses a 3×3 window to extract its features, further it is pooled to 3×3 convolutions for better results. The error rate, and accuracy during the training process are calculated by performing various experiments with the same network architecture. It consists of different layers in this architecture with convolution layer at the start followed by the normalization and activation layer (ReLu) and pooling are used immediately after the activation layer or before the convolutional layer. Also, strides are used for adjusting the sizes wherever necessary and is ended with an average pooled layer and fed to the output layers. A random image is taken from the outside source which is LBP coded is fed at the output stage to classify the appropriate class from the taken datasets. The below mentioned are some of the basic formula to calculate the responses of the system.

$$\text{Classified Rate} = \frac{\text{Classified Samples}}{\text{Total no of samples}}$$

$$\text{Correct Rate} = \frac{\text{Correctly Classified Samples}}{\text{Classified Samples}}$$

$$\text{Error Rate} = \frac{\text{Incorrectly Classified Samples}}{\text{Classified Samples}}$$

The LBP being a small model with very less learnable parameters makes the training process very easy and can effectively train the learning process and also able to prevent the fitting problems. Very high parameters are used in the deep network training process with huge complexity thus can be achieved using LBP features. Shown below is the training error of the used network. The error (%) can be gradually reduced using normalization methods like drop-out to prevent the indiscriminate neuron activations with the intermediate neurons. Below stated are statistical experiments shown on cifar-10 and cifar-100 datasets shown in fig 6.

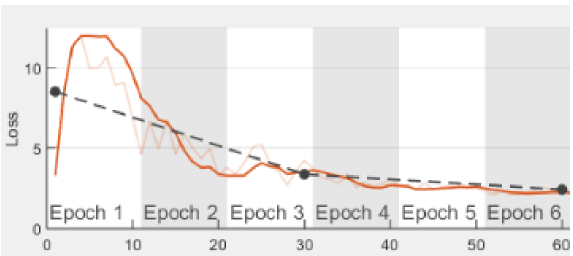

Fig-6: Error rate during training process

This classification has been done with different architectures with various parameters and are acquired significant results using the proposed network architecture. The validation accuracy obtained during the training process is tabulated below and the error rate is figured out. ResNets generally has smaller response with more complexity thus it takes huge time for the training process.

Top-1 error	3 × 3	5 × 5	Validation accuracy
Cifar-10	28.47	27.94	92.3
Cifar-100	29.76	26.39	93.5

Table-1: Classification results during training (%)

Given below is the confusion matrix which is used to describe the performance of a classification model. It shows how accuracte is the classifier. It is the set of features that visualize the dataset taken for both the train and test.

C
100x100 double

	1	2	3	4	5	6
1	0	0	0	0	4	0
2	0	0	0	0	1	0
3	0	0	0	0	0	0
4	0	0	0	0	2	0
5	0	0	0	0	5	0
6	0	0	0	0	1	0
7	0	0	0	0	3	0
8	0	0	0	0	1	0
9	0	0	0	0	2	0
10	0	0	0	0	2	0
11	0	0	0	0	3	0

Fig-7: Small part of the Confusion matrix

The Confusion matrix results C in this experiment is shown above. It is actually calculated for 100 classes as per the datasets taken in this experiment. Those results show the accuracy of the confusion matrix is very efficient. This deep learning architecture with effectively reduced parameters which reduces the complexity and thus improving the classification performance.

V. CONCLUSION

By considering various image classification methods in this paper an effective approach using Local Binary Patterns was proposed using Deep learning architecture. The LBP Net retains the similar topology of Residual Network with effective cost. Deep learning reduces to optimize the linear weights with optimizing convolutional filters, where LBP comprises of binary values and randomly generated convolutional weights which are linear learnable weights. Here, it can be described by both theoretically and experimentally that this ResNet with LBP performs as a good descriptor thus by using different set of parameters during the training processes with 3 × 3 and 5 × 5 sized filters respectively. This network is very simple yet very effective due to its low network model suitable for deep learning with low complexity. This proposed ResNet with LBP has an excellent performance when training with different data sets like Cifar-10 and Cifar-100 and outperforms various other datasets across different network architectures.

REFERENCES

[1]. Li Deng and Dong Yu, "*Deep learning: methods and applications*," Foundations and Trends in Signal Processing, *vol. 7, no. 3–4, pp. 197–387, 2014*.

[2]. G. Levi, T. Hassner, *Emotion recognition in the wild via convolutional neural networks and mapped binary patterns*, in: ICMI, *2015*.

[3]. C. Chen, B. Zhang, H. Su, W. Li, L. Wang, Land-*use scene classification using multi-scale completed local binary patterns*, SIVP 4 *(2016)*.

[4]. J. A. dos Santos, O. A. B. Penatti, R. da Silva Torres, *Evaluating the potential of texture and color descriptors for remote sensing image retrieval and classification*, VISAPP, *2010*.

[5]. T. Ojala, M. Pietikainen, D. Harwood, *A comparative study of texture measures with classification based on featured distributions*, PR *29 (1) (1996) 51–59*.

[6]. M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi. XNORNet: *ImageNet Classification Using Binary Convolutional Neural Networks*, 2016.

[7]. Ojala. T, Pietikainen. M, and Maenpaa. T, *Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns*. IEEE Trans. Pattern Analysis and Machine Learning, *2002*.

[8]. K. Simonyan, A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, ICLR, *2015*.

[9]. K. He, X. Zhang, S. Ren, J. Sun, *Deep residual learning for image recognition*, CVPR, *2016*.

[10]. M. Courbariaux, Y. Bengio, and J.P. David. Binary Connect: *Training Deep Neural Networks with binary weights during propagations*. In Advances in Neural Information Processing Systems (NIPS), *2015*.

[11]. K. He, X. Zhang, S. Ren, and J. Sun. *Deep residual learning for image recognition*. CoRR, *abs/1512.03385, 2015*.

[12]. Timo Ahonen, Abdenour Hadid, and Matti Pietikainen, ¨ "*Face recognition with local binary patterns*," Computer vision-eccv *2004, pp. 469–481, 2004*.

[13]. S. Zagoruyko and N. Komodakis. *Wide residual networks*. CoRR, *abs/1605.07146, 2016*.

[14]. Timo Ahonen, Abdenour Hadid, and Matti Pietikainen, ¨ "*Face recognition with local binary pattern*s," Computer vision-eccv *2004, pp. 469–481, 2004.*
[15]. M. Courbariaux and Y. Bengio. Binary Net: *Training Deep Neural Networks with Weights and Activations Constrained to +1 or -1, 2016*.

Y. Mani Kumar, et. al. "Residual Network with LBP for Image Classification." *International Journal of Engineering Research and Applications (IJERA)*, vol.10 (09), 2020, pp 36-42.