International Journal of Intelligent Information Technologies
Volume 18 • Issue 1

Deep Neural Network Regularization (DNNR) on Denoised Image

Richa Singh, Amity Institute of Information Technology, Amity University, Noida, India
Ashwani Kumar Dubey, Amity School of Engineering and Technology, Amity University, Noida, India*
Rajiv Kapoor, Delhi Technological University, India

ABSTRACT

Image dehazing in supervised learning models suffers from overfitting and underfitting problems. To avoid overfitting, the authors use regularization techniques like dropout and L2 norm. Dropout helps in reducing overfitting and batch normalization reduces the training time. In this paper, they have conducted experiments to analyze combination of various hyperparameters to have better network performance using deep neural network (DNN) on cifar10 dataset. The qualitative and quantitative study is performed by estimating the accuracy of the model on training and test images using with and without batch normalization. The proposed model performs better and is more stable. The results shows that dropout regularization technique is better than L2 technique containing hidden layers with large neurons. The paper assesses performance of DNN for any denoised model with the techniques like batch normalization and dropout, feature map, and adding more layers to the network. The authors quantitatively identify the value model loss and accuracy with the absence and presence of these parameters.

KEYWORDS

Batch Normalization, Convolutional Neural Network, Deep Neural Network, Dropout, Multi-Layer Perceptron, Neural Network, Overfitting, Rectified Linear Unit, Regularization

1. INTRODUCTION

Image dehazing in supervised learning models suffers from overfitting and underfitting problems and is a complex job in computer vision. In deep learning due to enormous number of parameters overfitting and training time is really a challenging task. Though there exists batch normalization and dropout to regularize and handle but they do overlap and have their respective strengths and limitations for the network. Overfitting is a fundamental problem which slows down the model performance, dropout technique usually used for this problem. To avoid overfitting, we need to control complex parameters present in the model using regularization techniques. The techniques which are majorly used for regularizing the complex models are dropout and L2 norm. Many researchers try to perform

DOI: 10.4018/IJIIT.309584
*Corresponding Author

these two techniques in majority to reduce overfitting in models. This reduces overfitting and provide advancements around other regularization methods. Dropout enhances the act of deep neural networks having tasks of supervised learning in computer visualization, medical, recognition of speech and other progressive results, though their practice plans are available, unfortunately no well-defined set of rules or comprehensive revisions to explore them regarding network configurations and learning efficacy is defined. It is not obvious when should consider dropout or batch normalization, can they be combined or not. Dropout helps in reducing overfitting as we drop some units in the network and batch normalization reduces training time but to combine these parameters such as Dropout and Batch Normalization does improve accuracy but to have some empirical results, we have conducted some experiments and, in this paper, we try to analyze combination of various hyperparameters to have better network performance by doing some empirical study using deep neural network using images from cifar10 dataset. In this paper we did an empirical study to identify the various parameter effects like using Batch Normalization and dropout effects separately and in combination, further we also identify the effect of adding more layers to the network and adding more feature maps to the network. The results are shown on a visual plot specifying the accuracy. The qualitative and quantitative study is performed by estimating the accuracy of the model on training and test images using with and without batch normalization and dropout. The comparative result of the model proves that the model performs better with which combination of hyperparameters and is more stable. The experiment outcome shows that dropout regularization technique is better than L2 technique containing hidden layers with large neurons. The paper assesses the denoised functioning of DnCNN model with the techniques like batch normalization and dropout, feature map and adding more layers to the CNN network. We quantitatively identify the value model loss and accuracy with the absence and presence of these parameters. Finally, we conclude by specifying that which parameters are necessary to have better accuracy in our network model.

To improve the image quality and eliminate noise many denoising algorithms have been researched in literature. Many important denoising methods such as dark channel prior (He et al., 2010), using transmission map of natural images (Singh et al., 2021), histogram equalization (Stark & J, 2000), convolutional neural network (Ren et al.,2016), video denoising (Buades & Lisani .,2017), recently deep neural networks (Rahangdale & Raut,2019), denoising using autoencoders (Wen & Zhang,2018), generative adversarial network (GAN) (Goodfellow et al.,2020), dehazing using deep neural network (Hodges et al.,2019) are projected for image denoising. To build any model, we should be able to generalize so that it can predict well on unseen data. In the process of generalizing a model, one should consider a mapping function having input and output function such that the parameters used in model should be enough to train the model otherwise model would either suffer from underfitting or overfitting.

The main challenge occurs in training of the multi layered network is the overfitting and the time taken for training the network. Batch Normalization and dropout techniques are there to overcome these problems with limitation of its designing. These approaches have their own advantages, but when to use them independently or in combination is an issue (Li et al.,2018) (Luo et al.,2018). Many researches performed methods based on unsupervised, semi supervised and supervised learning.(Zhu X,2005) and methods learning network using mathematical models without having specific rules. ((Goodfellow et al.,2016).Conventional methods of machine learning trains data such as multilayer perceptron(RumelHart et al.,1995) and decision trees(Loh,2014).Recent deep learning approaches such as recurrent Neural Network(Lipton et al.,2015) and convolutional neural network(Bengio & Lecun .,2015) gain performance improvement due to its various features helps in image classification problems (Perez & Wang,2017)and others.

Any simple feedforward network has layers or neurons per layer starting from input layer till multiple hidden layers called Multilayer perceptron (MLP), these neurons help in maintaining complexity of network. These MLPs are used to solve many vision problems. A network complexity is a major issue in neural network which leads to overfitting problem. To avoid overfitting some mechanism is required to avoid making a network complex, for this reason many researchers tried to solve the problem of overfitting using known method of regularization. Regularization is the technique which can be generalize and deep learning with various additional features as compared to traditional or conventional methods for facing such issues in a network. A recent approach to deep neural network regularization is known as dropout (Srivastava N et al.,2014) and Batch Normalization (Ioffe & Szegedy.,2015). As the name depicts dropout tend to drop certain neurons from the layers and train the model by reducing the complexity and controlling overfitting. There is another regularization technique known as L2 norm, in our study we try to analyze both regularization method. In this paper, an experimental analysis is done to see the performance of L2 and dropout regularization. Usual regularization technique is to add up a regularization word and the objective function. The word confirms that our model does not overfit and leads to generalization. We can define it as:

Objective Function=Loss Function (which is an error term) + Regularization term

As we know the bias is the amount of error in the model whereas variance will tell model changes when trained on various datasets.

$$O = L\left(F\left[Xi\right],\theta\right) + \lambda f\left(\theta\right) \quad (1)$$

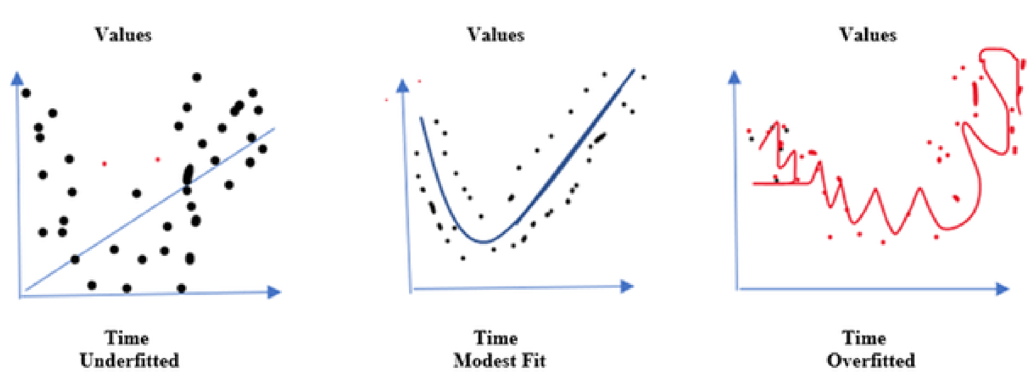
Equation (1), Where O is the objective function, $\left(F\left[Xi\right],\theta\right)$ is the loss function and $\lambda \, is \, the \, regularization \, parameter \, and \, f\left(\theta\right)$ is parameter function. Two types of norm function L1 and L2 but L2 norm is preferred due to the sum of the squares term which are easily differentiable during backpropagation.

Figure 1 depicts the problem of overfitting and underfitting, graph shows the robust model. The deep learning architecture models use features of images and do learning process during training to produce satisfactory results as compared to the traditional methods described.

1.1. Batch Normalization, L2 and Dropout Regularization

Neural networks have wide range of applications in image detection, automated speech identification (ASR), vision captioning, and video evaluation. Artificial Neural Networks (ANNs) are powerful technique for learning models in machine learning and similarly deep neural networks performs with great accuracy called deep learning models. A neural network with deep convolutional learning architecture is most known architecture for images and videos. CNN is a sequence of input layer, various hidden layers, and an output layer. Feature extraction is done at the hidden layers by doing

Figure 1. Underfitting and Overfitting in machine learning



convolution operation. The key CNN architectures are AlexNet (Krizhevsky et al.,2017), VGGNet (Simonyan & Zisserman,2014), LeNet (Lesun et al.,1998), GoogleNet (Szegedy et al.,2015) ResNet (He et al.,2016), etc. Visual Geometry Group (VGGNet) model was launched in the ImageNet LSVRC-2014 (Russakovsky et al.,2015). There are several VGG architectures having convolutional layers from 8 to 16 beside three fully linked layers following in VGG-19, VGG-16, VGG-13, VGG-11 models. Amongst them, VGG-16 model surpasses all other patterns on the categorization task in terms of top-1 fault (%) and top-5 fault (%). The step function, we can make use of 1 and -1 as well in its place of 1 and 0:

$$y = 1 if\, x > 0 and\, y = 0 ifx \leq 0 \tag{2}$$

Equation (2) defines step function where a perceptron utilizes a step function to execute binary class which will be able to identify applying the subsequent diagram. A group of perceptron can act as a universal function approximators using AND gate which helps in understanding Neural Network and is considered as a universal function approximator. A perceptron and a group of such perceptron networks can be used to execute multiclass classification: An artificial neuron is exceptionally like a perceptron, apart from that the function of activation is not a step function.

Activation Function:

$$F\left(a1w1 + a2w2 + a3w3 + a4w4\right) \tag{3}$$

Perceptron which is the input is the inputs of the weighted sum. The activation function equation (3) is the output which is been given as input. The activation function could be a function such that it should be smooth and make the inputs and outputs non-linear. An artificial neural network is a network of such neurons which are arranged in different layers. The layers are called the input and output layers mentioned as first and last. The number of attributes is the neurons in the input layer of data set and number of classes in the output layer has as many as the number of neurons is the target variable. The six main things that defines a neural network entirely: 1. Topology of Network 2. The Input Layer 3. The Output Layer 4. Weights 5. Biases6. Activation functions

The data input for images are needed in the form of arrays known as pixels having each pixel some intensity value having different RGB channels. Figure2(a) depicts a grey image and 2(b) RGB

Figure 2. (a) Grey Scale Image (b) RGB channel image

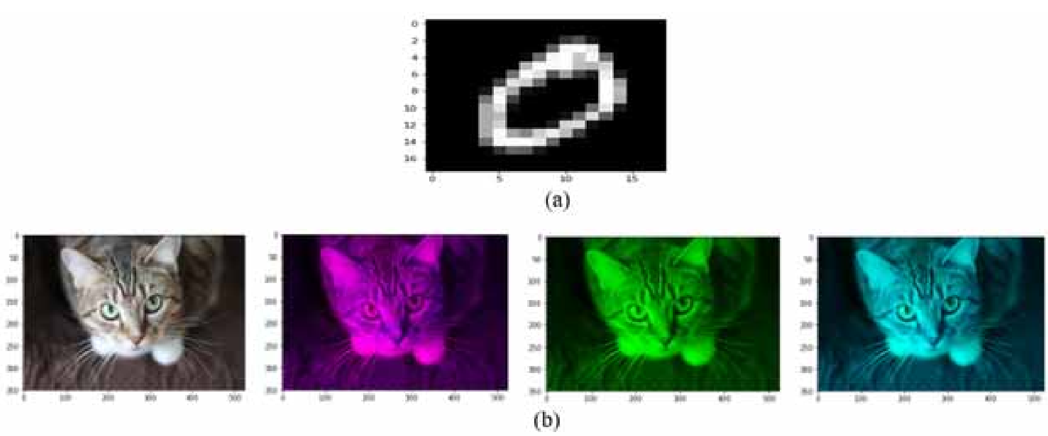


image. Neural networks can potentially have extremely dense structures, some assumptions are there to simplify network model. These are defined as

1. Neurons are arranged in layers sequentially.
2. Neurons inside the same layer do not relate with each one.
3. Each input entering the network all through the input layer and output layer move out of the network all through the output layer.
4. Neurons in successive layers are tightly tied.
5. Every interconnectedness in the neural network takes a weight and bias linked with it.
6. Every one of the neurons in a particular layer use the related activation function.

To minimize the overfitting in the model we do regularization. Two major types are L2 norm and dropout. Recently many researchers proposed only dropout as it gives better results such as variance shift (Li et al.,2019), gaussian denoiser (Zhang et al.,2017) and noise label (Frenay & Verleysen 2014). Neural networks are trained on the small datasets which leads to overfit the model as we are taking too many parameters to train, so we need ideally to train on substantial number without overfitting. Dropout does the approximation and train substantial number of neural networks having different architecture in parallel. Each layer is implemented with dropout in the network. Dropout doubles the process of iteration to converge though the training time for each epoch taken will be less.

To understand the batch normalization, we try to understand by using weights and activation function using equation 4,5, and 6. In any feedforward network:

$$H1 = \sigma\left(w1.x + b1\right) \tag{4}$$

$$H2 = \sigma\left(w2.H1 + b2\right) = \sigma\left(w2\left(\sigma(w1.x + b1)\right) + b2\right) \tag{5}$$

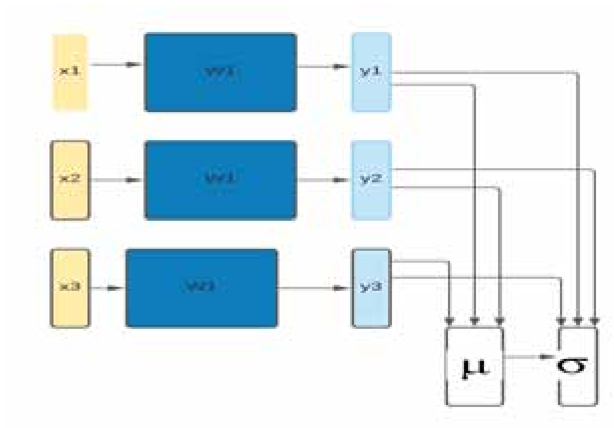
$$H3 = \sigma\left(w3.H2 + b3\right) = \sigma\left(w3\left(\sigma\left(w2\left(\sigma\left(w1.x + b1\right)\right) + b2\right)\right)b2)) + b2\Big)\Big) + b3) \tag{6}$$

Where, σ is the activation function. We can see that H3 is the composite function of weights and biases from earlier layers and interaction between weight matrix is nonlinear, though during backpropagation weights from the layers are independently updated. We can observe that if we change the weights of previous layers, it should affect the output of next layer which in turn affects the gradient. It should not happen, so we try to use batch normalization which helps in solving this issue. Batch normalization is used to apply on output of the layers of the batches we formed, H1. It normalizes H1 and all data points by using mean and standard deviation vector over a batch.

In Figure 3, if layer L has supposed m neurons, then H1, equation (7) takes mean of first element is the mean of first neuron for all data points and so on and similarly the standard deviation. In this way batch normalization is done for all layers except the output layer. There are various heuristics as deep learning frameworks like keras are proven techniques.

$$H1 = h1 - \mu^{\wedge}\sigma \tag{7}$$

Figure 3. Batch normalization process for any layer l



After batch normalization, we use to do regularization there are L2 and dropouts major regularized techniques which helped to ensure that model does not overfit and have sufficient parameters to be trained on the network. Dropout among these gives better accuracy as it can be applied to some layers of the network which can be chosen arbitrarily there a is a mask alpha (α) which generated independently for every layer in the process of feedforward and similarly in backpropagation which changes with each iteration, or the small batches taken from Bernoulli p (1) =q. Dropouts generally reduces the complexity of the model. Symmetry can also be broken using dropouts and there exists such communities in the neurons which helps to restrict them to learn independent. Different minibatches in epoch dropouts are applied. Dropout operation can be specified by weight matrix multiplication with a mask vector alpha equation (8).

$$Dropout = Wl.\neg \tag{8}$$

Where Wl is the weight matrix and alpha are the mask. In Dropout arbitrary neurons are ignored in the exercise phase and when the network is transformed from exercise towards testing, then dropout shifts the variance whereas batch normalization has its variance shift leads drop in the presentation of the network when dropout is blend with batch normalization.

We know that values of weights and filters are learnt during training known as neuron. Multiple filters are used to identify different feature. Feature map having multiple neurons having same weights and all neurons tells the same feature known as feature map.

A feature map with input and output layers with two filters or more, CNN aggregates these features after feature extraction using pooling layers. Using pooling we would be able to extract desirable feature from it. Pooling can be done using average or max. Pooling helps in reducing parameters by the network to be learnt though it might lead to losing some valuable information too. CNNs had initially shown their remarkable accomplishment in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). The ILSVRC uses one thousand image categories and has 1.2 million training images.

The paper is arranged in following sections: section 2 illustrates Related Work; section 3 displays Proposed Methodology with assumptions and hyperparameters in the network applied and section 4 has Results and Discussion, and section 5 has Conclusion and section 6 displays Future Recommendations.

2. RELATED WORK

During training of complex deep networks various parameters are engaged and to optimize these hyperparameters is a big challenge. Different optimizers are proposed and depend on on the principle of Gradient Descent (Bengio,2012) & (Ruder, 2016).

Standards for tuning and training the parameters specified (Smith, 2018) helps in training the network faster. In (Hinz et al., 2018) proposed to take input data having the low-resolution images or low aspect. Overfitting being a challenge for lower samples some easy and efficient approaches in deep learning are defined as dropout (Srivastava et al., 2014) and to accelerate the performance of the network as Batch Normalization is proposed (Ioffe S & Szegedy, 2015). Combining both dropout and batch normalization is proposed in (Li et al., 2018). Existing approaches are costly and less effective so to find a simple and effective approach is a great challenge.

Dropout regularization is a neural network in which we can add noise to its hidden layers. The adding of randomness to the layers gets applied in the Denoising Autoencoders by (Vincent et al. (2008, 2010)). Our work continues the idea that dropout can be efficiently used in the hidden layers as summing the model. Dropping from 20 percent of the input units to the 50 percent of hidden units found optimal. Dropout is a stochastic regularization technique, (Maaten et al., 2013) also investigated deterministic regularization related to noise distributions. (Green et.al., 2013) suggested a method for speed higher dropout by relegating dropout noise. To explore into downgrade in the perspective of denoising autoencoders. (Globerson and Roweis 2006); (Dekel et al., 2010) discovered a different setting where loss is reduced when an opposition gets to choose which elements to give up.

Deep networks (Liu et al., 2017) used in image denoising in 2015 (Liang & Liu, 2015), deep networks were extensively used in videos (Yuan et al., 2020) and image restoration (Ren et al., 2020). (Mao et al., 2016) to suppress image noise used multiple deconvolutions and convolutions to have high-resolution image. Batch normalization (Ioffe & Szegedy, 2015) and (ReLu) the rectified linear unit (Nair & Hinton, 2010) was proposed.

To summarize, in our empirical study in this paper with some practical recommendations differs from existing works in the following aspects:

1. Train our Deep Neural Network with batch normalization technique creating batches of the dataset and visualize the effect using plotting the graph between model loss and the varying epochs.
2. Using L2 regularization visualization is done.
3. Train our Deep Neural Network with dropout technique using dropout value taken as 0.50 and 0.25 of the datasets and visualize the effect using plotting the graph between model loss and the varying epochs.
4. Combined effect of batch normalization and dropout is visualized similarly.
5. Tried to see the effect of boosting the number of layers in the network.
6. Tried to add feature map in the network.

In literature, the layers of batch normalization have been used to different neural network architectures along with elevated levels of achievement (Noh et al., 2015).

3. PROPOSED METHODOLOGY

In our approach we apply deep learning concept with dropout regularization along with batch normalization. In proposed methodology we are using cifar10 dataset and try to test the effect of batch normalization and dropout applied on to the images. We would see the difference in accuracy having batch normalization and dropout in the images inclusive and non-inclusive. The Cifar-10 dataset (Krizhevsky & Hinton,2009) is a set of small images from dataset (Torralba, et al.,2008). The dataset contains little images divided into ten groups of objects and animals as trucks, automobiles, and horses, The classes are entirely mutually exclusive and there is no similarity between the automobile and truck classes. Currently the best accuracy achieved on CIFAR10 is 96.53% with a convolutional neural network. The Cifar10 dataset is completely stable, sense that the rate of class labels is precisely equal for all the classes.

Cifar-10 having training images of 50,000 and test images of 10,000, have ten categories, image has the of dimension $32 \times 32 \times 3$, where the 3 represents three color channels of the images. The training set is the largest of the three types of datasets i.e., training set, test set and validation set which helps in finding parameters of the model which gives the fundamental analytical association between the data and labels. Most methods that try to fit parameters centered on empirical relations with training set only tend to overfit the data. This inspires to have test set which contains data that is not used while training but observes the same possibility dissemination and predictive relationship. To approximate a model's performance during training, a validation set is used. The validation is generated by dividing the training set in two parts, the smaller is used for hyperparameter optimization and to prevent overfitting out of early stopping. Early stopping is an appropriate effort to find the optimal time to stop the training process and to choose when a model is fully specific with respect to its parameters and hyperparameters.

Figure 4 shows some random images from the cifar10 dataset with output target as ten classes.

3.1. Assumptions in Neural Network

Any neural network architecture has following streamlining assumptions:

1. Neurons are placed in the form of layers and these layers are set serially.
2. Neurons inside the like layer do not relate with each one of another.
3. Each input and output go into and exit in the network across respective layer.
4. Neurons in sequential layers are connected tightly.
5. Every interconnectedness in network has a weight and bias connected with it.
6. All neurons in all the layers make use of the similar activation function.

3.2. Parameters and Hyperparameters of Neural Networks

Neural network learning is accomplished with a static collection of hyperparameters – the structure of the network includes quantity of neurons and layers in the input, output and hidden layers having the weights and the biases, parameters in the network. Activation functions used should be smooth, the inputs and outputs non-linear making network compact. The known activation functions used for neural networks generally are logistic function, hyperbolic tangent function (Jarrett et al., 2009) and

Figure 4. Images from cifar10 dataset

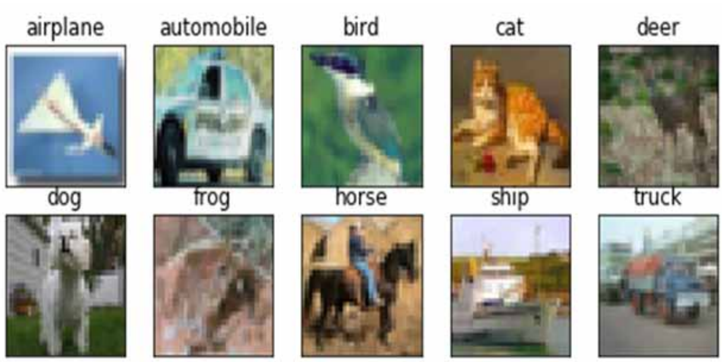
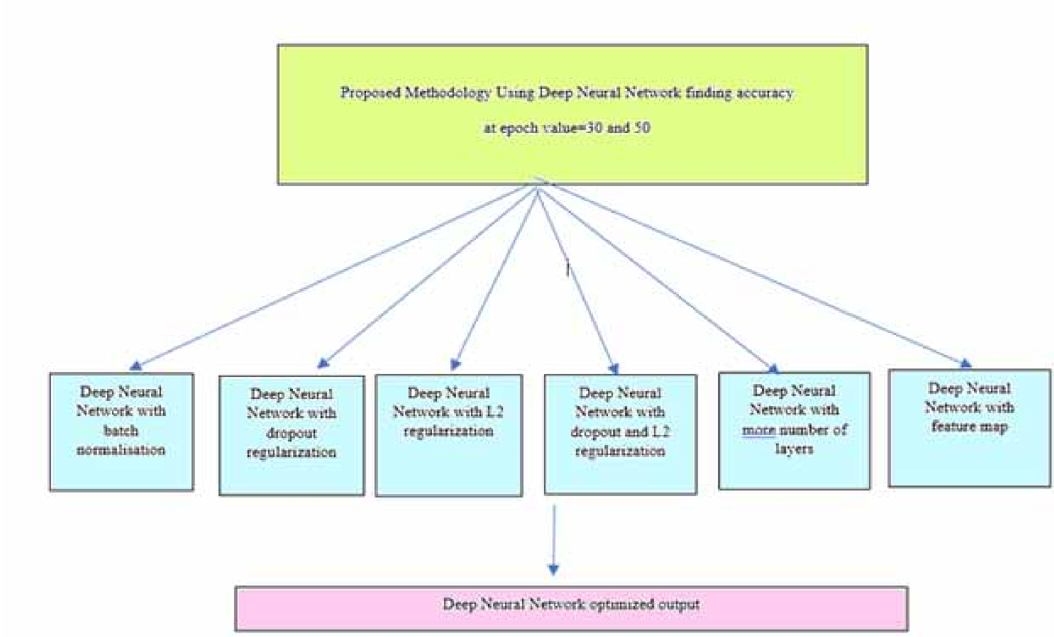


Figure 5. Process flow of proposed methodology



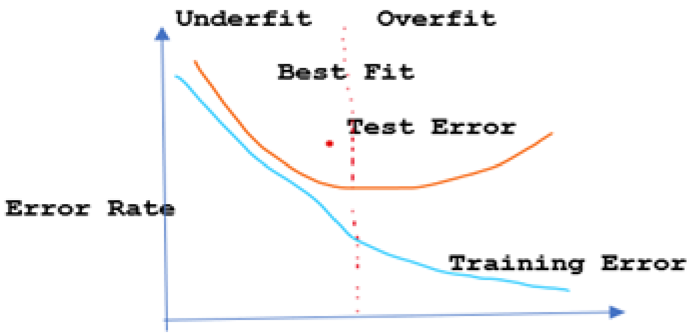
(Yuan et al., 2020) sigmoid function (Marreiros et al., 2008), Rectilinear Unit and other activation functions like Leaky Relu. CNN kernels can be designed and convolves an image and isolates features from each 'patch'. Different features. Multiple features can be extracted using unique features from the image. A typical CNN have multiple filters, observed as non-linearity in the activations, and in a pooling layer. The pooling layer calculates a cumulative statistic.

In our proposed methodology we are trying to show that the parameters responsible for having better accuracy of training and validation set Figure 5. We also try to propose the model to avoid overfitting (Shen &Shafiq, 2018) and underfitting. Our methodology as per the process flow

1. Deep Neural Network with batch normalization
2. With Dropout regularization in deep neural Network.
3. Deep Neural Network with L2 regularization
4. Deep Neural Network having dropout as well L2 regularization
5. Deep Neural Network with a greater number of layers
6. Deep Neural Network with feature map
7. Deep Neural Network optimized output

We try to evaluate performance based on the graph shown in Figure 6. We have used RMS Prop optimizer and epoch value of 30 and 50, batch size of thirty-two has been taken. For convolutional layers we have use 32 and 64 features and for increasing layer option we have used 128 and 64 features. Adding one more convolution layer means adding one level of abstraction whereas adding feature map more features are added at the same level of abstraction.

Figure 6. Model Complexity



4. ANALYSIS OF THE RESULTS TESTED WITH DIFFERENT PARAMETERS

We depict performance of different DnCNN architectures having kernel size = 3 and 5 with and without batch normalization along with and without dropout (Garbin et al., 2020) at 30 and 50 epochs using RMSProp optimizer. We try to evaluate how to avoid overfitting or underfitting by adding parameters.

Figure7 shows deep neural network with batch normalization having training Accuracy=95% and Validation Accuracy=78% at the epoch value of 30 and similarly at epoch value=50 training accuracy=98% and validation Accuracy=79%, we can infer that using only batch normalization would lead to underfitting of the model. Dropout values without Batch Normalization model loss at different epochs deep neural network with regularization technique dropout having training accuracy of Accuracy=77% and Validation Accuracy=78% at the epoch value of 30 and similarly at epoch value=50 training accuracy=84% and validation Accuracy=79%, we can infer that using only dropout will give less difference between training and validation accuracy, deep neural network with regularization technique L2 having training accuracy of Accuracy=77% and Validation Accuracy=84% at the epoch value of 30 and similarly at epoch value=50 training accuracy=92% and validation Accuracy=78%, we can infer that using L2 regularization does not give better result than dropout, deep neural network with Batch Normalization and Dropout of Accuracy=83% and Validation Accuracy=80% at the epoch value of 30 and similarly at epoch value=50 training accuracy=89% validation Accuracy=82%, we can infer that it gives better approach, deep Neural Network with more layer training accuracy=85% and Validation Accuracy=83% at the epoch value of 30 and similarly at epoch value=50 training accuracy=89% and validation Accuracy=83%, we can infer that it will increase the training time with marginal difference, deep neural network with more feature map of training Accuracy=90% and Validation Accuracy=82% at the epoch value of 30 and similarly at epoch value=50 training accuracy=92% validation Accuracy=83%, we can infer that quality does not get degraded but only increases features at same level of abstraction.

Table1 depicts the inference of applying various parameters on the network which concludes that using batch normalization with dropout is a better choice. Increasing layers or feature map leads to marginal difference in accuracy and quality remains intact with increase in computation complexity.

5. CONCLUSION

Deep Neural network use batch normalization and dropout as regularization technique as we have observed from experiment that in comparison of L2 and dropout, dropout works better and helps in improving the performance of the model. The overlapping of batch normalization and dropout in many applications are quite contradictory and provide less detailing. In this paper so for that reason we did

Figure 7. Batch Normalization

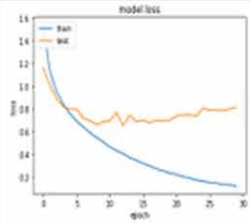
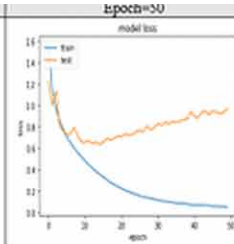
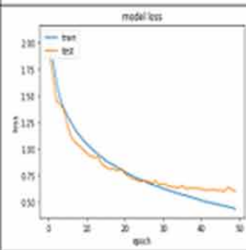
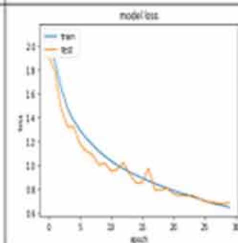
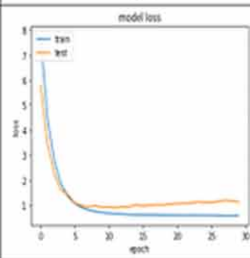
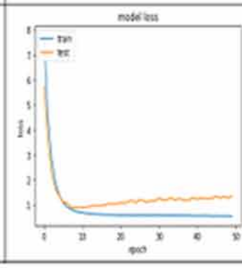
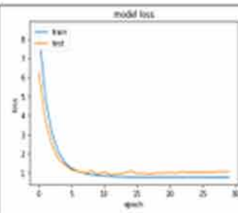
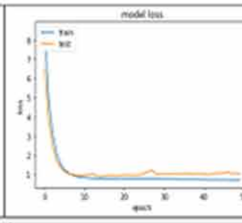
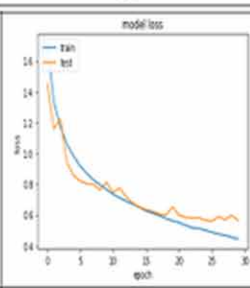
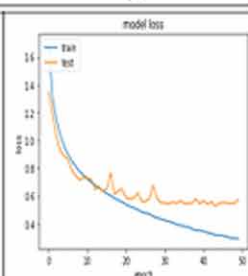
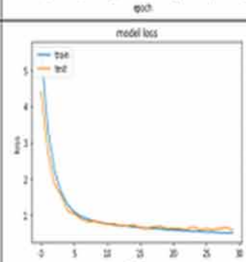
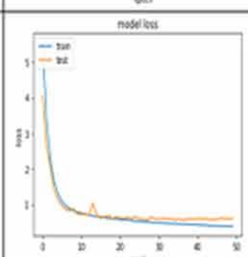
Process	Epoch=30	Epoch=50	Inference
Batch Normalization without dropout model			After using batch normalization there is not much difference in training and testing accuracy, but the model is giving low accuracy and model is overfitting.
Dropout without batch normalization model			This is the regularization technique though the difference between training and testing accuracy is low; but connections are effective
Deep Neural Network with regularization technique L2			This is also the regularization technique but not so effective as dropout.
Deep Neural Network with more feature map			The quality does not get degraded and increasing a feature map at the same level of abstraction increases only the features.
Batch Normalization and dropout values			If we use both dropout and batch normalization is a better approach
Deep Neural Network with more layer values			Adding more layers to the network increases the computational time whereas giving a marginal difference in accuracy

Table 1. Testing Accuracy various parameters at epochs=30 and epochs=50

Epochs	Epoch=30	Epoch=50	Inference
Deep Neural Network with batch normalization	Accuracy=95% Validation Accuracy=78%	Accuracy=98% Validation Accuracy=79%	After using batch normalization there is not much difference in training and testing accuracy, but the model is giving low accuracy and model is overfitting
Deep Neural Network with dropout regularization	Accuracy=77% Validation Accuracy=76%	Accuracy=84% Validation Accuracy=79%	This is the regularization technique though the difference between training and testing accuracy is low, but connections are effective
Deep Neural Network with L2 regularization	Accuracy=92% Validation Accuracy=78%	Accuracy=77% Validation Accuracy=84%	This is also the regularization technique but not so effective as dropout.
Deep Neural Network with batch normalization and drop out	Accuracy=83% Validation Accuracy=80%	Accuracy=89% Validation Accuracy=82%	If we use both dropout and batch normalization is a better approach.
Deep Neural Network with a greater number of layers	Accuracy=85% Validation Accuracy=83%	Accuracy=89% Validation Accuracy=83%	Adding more layers to the network increases the computational time whereas giving a marginal difference in accuracy
Deep Neural Network with feature map	Accuracy=90% Validation Accuracy=82%	Accuracy=92% Validation Accuracy=83%	The quality does not get degraded and increasing a feature map at the same level of abstraction increases only the features.

an empirical study of all the features like batch normalization, dropout together and in isolation how they perform on multilayer perceptron, try to identify the effect of adding feature maps and more layers to the network to differentiate and see the performance based on these features. We also see the role of hyperparameters. Our major finding based on experiments conducted.

- Using batch normalization there is not much difference in training and testing accuracy, but the model is giving low accuracy and model is overfitting.
- Using Dropout regularization technique though the difference between training and testing accuracy is low, but connections are effective.
- Using L2 regularization technique we find that, but it is not so effective as dropout.
- If we use both dropout and batch normalization is a better approach.
- Adding more layers to the network increases the computational time whereas giving a marginal difference in accuracy.
- The quality does not get degraded and increasing a feature map at the same level of abstraction increases only the features.
- Using in combination dropout and batch normalization we could see training accuracy and validation accuracy as 83% and 80% respectively running at 30 epochs whereas on fifty epochs this accuracy becomes 89% and 82%.

The performance of denoised image with deep convolutional neural network created the DnCNN model and try to estimate the accuracy of the model. Batch normalization and Residual learning are key features of the DnCNN model. We have used cifar10 dataset for applying our proposed methodology on the images and performance is analyzed by evaluating training and validation accuracy by applying different parameters The quantitative analysis of accuracy at various epochs has been evaluated concludes that using batch normalization with dropout is a better choice. Increasing layers or feature map leads to marginal difference in accuracy and quality remains intact with increase in computation complexity. In future work we would be extending the proposed work to video images as well.

6. FUTURE RECOMMENDATIONS

- For increased accuracy we can make use of batch normalization and dropout. though dropout may reduce some accuracy, but we can try different dropout rates.
- Tests can be conducted on deeper networks for finding decrease in dropout accuracy.
- The same can be extended to video images.

CONFLICT OF INTEREST

The authors of this publication declare there is no conflict of interest.

FUNDING AGENCY

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

Bengio. Y. & LeCun. Y. (2007). Scaling learning algorithms towards AI. *Large-Scale Kernel Machines, 34*(5), 1-41.

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade* (pp. 437–478). Springer. doi:10.1007/978-3-642-35289-8_26

Buades, A., & Lisani, L. J. (2017, February). Denoising of Noisy and Compressed Video Sequences. In *VISIGRAPP* (Vol. 4, pp. 150–157). VISAPP. doi:10.5220/0006101501500157

Dekel, O., Shamir, O., & Xiao, L. (2010). Learning to classify with missing and corrupted features. *Machine Learning*, *81*(2), 149–178. doi:10.1007/s10994-009-5124-8

Fan, E. (2000). Extended tanh-function method and its applications to nonlinear equations. *Physics Letters. [Part A]*, *277*(4-5), 212–218. doi:10.1016/S0375-9601(00)00725-8

Frénay, B., & Verleysen, M. (2013). Classification in the presence of label noise: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, *25*(5), 845–869. doi:10.1109/TNNLS.2013.2292894 PMID:24808033

Garbin, C., Zhu, X., & Marques, O. (2020). Dropout vs. batch normalization: An empirical study of their impact to deep learning. *Multimedia Tools and Applications*, *79*(19-20), 12777–12815. doi:10.1007/s11042-019-08453-9

Globerson, A., & Roweis, S. (2006, June). Nightmare at test time: robust learning by feature deletion. In *Proceedings of the 23rd international conference on Machine learning* (pp. 353-360). doi:10.1145/1143844.1143889

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., & Bengio, Y. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27.

Goodfellow, J. I., Bengio, Y., & Courville, C. A. (2016). *Deep learning. Adaptive Computation and Machine Learning*. MIT Press. https://www.deeplearningbook.org/

Green, S., Wang, S. I., Cer, D., & Manning, D. C. (2013, August). Fast and adaptive online training of feature-rich translation models. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (*Volume 1*: Long Papers)* (pp. 311-321). Academic Press.

He, K., Sun, J., & Tang, X. (2010). Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(12), 2341–2353. PMID:20820075

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778). IEEE.

Hinz, T., Navarro-Guerrero, N., Magg, S., & Wermter, S. (2018). Speeding up the hyperparameter optimization of deep convolutional neural networks. *International Journal of Computational Intelligence and Applications*, *17*(02), 1850008. doi:10.1142/S1469026818500086

Hodges, C., Bennamoun, M., & Rahmani, H. (2019). Single image dehazing using deep neural networks. *Pattern Recognition Letters*, *128*, 70–77. doi:10.1016/j.patrec.2019.08.013

Ioffe, S., & Szegedy, C. (2015, June). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448-456). PMLR.

Jarrett, K., Kavukcuoglu, K., Ranzato, M. A., & LeCun, Y. (2009, September). What is the best multi-stage architecture for object recognition? In *2009 IEEE 12th international conference on computer vision* (pp. 2146-2153). IEEE. doi:10.1109/ICCV.2009.5459469

Krizhevsky. A. & Hinton. G. (2009). *Learning multiple layers of features from tiny images*. Academic Press.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324. doi:10.1109/5.726791

Li, X., Chen, S., Hu, X., & Yang, J. (2019). Understanding the disharmony between dropout and batch normalization by variance shift. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2682-2690). IEEE.

Liang, J., & Liu. (2015, October). Stacked denoising autoencoder and dropout together to prevent overfitting in deep neural network. In *2015 8th international congress on image and signal processing (CISP)* (pp. 697-701). IEEE.

Lipton, Z. C., Berkowitz, J., & Elkan, C. (2015). *A critical review of recurrent neural networks for sequence learning.* arXiv preprint arXiv:1506.00019.

Liu, D., Wen, B., Liu, X., Wang, Z., & Huang, T. S. (2017). *When image denoising meets high-level vision tasks: A deep learning approach.* arXiv preprint arXiv:1706.04284.

Luo, P., Wang, X., Shao, W., & Peng, Z. (2018). *Towards understanding regularization in batch normalization.* arXiv preprint arXiv:1809.00846.

Maaten, L., Chen, M., Tyree, S., & Weinberger, K. (2013, February). Learning with marginalized corrupted features. In *International Conference on Machine Learning* (pp. 410-418). PMLR.

Mao, X., Shen, C., & Yang, Y. B. (2016). Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. *Advances in Neural Information Processing Systems*, 29.

Marreiros, C. A., Daunizeau, J., Kiebel, S. J., & Friston, K. J. (2008). Population dynamics: Variance and the sigmoid activation function. *NeuroImage*, *42*(1), 147–157. doi:10.1016/j.neuroimage.2008.04.239 PMID:18547818

Nair, V., & Hinton, G. E. (2010, January). Rectified linear units improve restricted boltzmann machines. ICML.

Noh, H., Hong, S., & Han, B. (2015). Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision* (pp. 1520-1528). IEEE.

Perez, L., & Wang, J. (2017). *The effectiveness of data augmentation in image classification using deep learning*. Academic Press.

Rahangdale, A., & Raut, S. (2019). Deep neural network regularization for feature selection in learning-to-rank. *IEEE Access: Practical Innovations, Open Solutions*, *7*, 53988–54006. doi:10.1109/ACCESS.2019.2902640

Ren, D., Shang, W., Zhu, P., Hu, Q., Meng, D., & Zuo, W. (2020). Single image deraining using bilateral recurrent network. *IEEE Transactions on Image Processing*, *29*, 6852–6863. doi:10.1109/TIP.2020.2994443

Ren, W., Liu, S., Zhang, H., Pan, J., Cao, X., & Yang, M. H. (2016, October). Single image dehazing via multi-scale convolutional neural networks. In *European conference on computer vision* (pp. 154-169). Springer. doi:10.1007/978-3-319-46475-6_10

Ruder, S. (2016). *An overview of gradient descent optimization algorithms.* arXiv preprint arXiv:1609.04747.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). *Learning internal representations by error propagation*. California Univ San Diego La Jolla Inst for Cognitive Science. doi:10.21236/ADA164453

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., & Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), 211–252. doi:10.1007/s11263-015-0816-y

Shen, J., & Shafiq, O. M. (2018, December). Deep learning convolutional neural networks with dropout-a parallel approach. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)* (pp. 572-577). IEEE. doi:10.1109/ICMLA.2018.00092

Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition.* arXiv preprint arXiv:1409.1556.

Singh, R., Dubey, K. A., & Kapoor, R. (2021). Improved Transmission Map for Dehazing of Natural Images. In *Emerging Technologies in Data Mining and Information Security* (pp. 339–347). Springer. doi:10.1007/978-981-15-9927-9_34

Smith, N. L. (2018). *A disciplined approach to neural network hyper-parameters: Part 1—learning rate, batch size, momentum, and weight decay.* arXiv preprint arXiv:1803.09820.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(1), 1929–1958.

Stark, J. A. (2000). Adaptive image contrast enhancement using generalizations of histogram equalization. *IEEE Transactions on Image Processing*, *9*(5), 889–896. doi:10.1109/83.841534 PMID:18255459

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9). IEEE.

Torralba, A., Fergus, R., & Freeman, W. T. (2008). 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *30*(11), 1958–1970. doi:10.1109/TPAMI.2008.128 PMID:18787244

Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning* (pp. 1096-1103). doi:10.1145/1390156.1390294

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P. A., & Bottou, L. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, *11*(12).

Wen, T., & Zhang, Z. (2018). Deep convolution neural network and autoencoders-based unsupervised feature learning of EEG signals. *IEEE Access: Practical Innovations, Open Solutions*, *6*, 25399–25410. doi:10.1109/ACCESS.2018.2833746

Yuan, D., Fan, N., & He, Z. (2020). Learning target-focusing convolutional regression model for visual object tracking. *Knowledge-Based Systems*, *194*, 105526. doi:10.1016/j.knosys.2020.105526

Zhang, K., Zuo, W., Chen, Y., Meng, D., & Zhang, L. (2017). Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, *26*(7), 3142–3155. doi:10.1109/TIP.2017.2662206 PMID:28166495

Zhu, X. J. (2005). *Semi-supervised learning literature survey*. Academic Press.

Richa Singh received the degree of Master of Information Technology from Meerut University, UP, India. Presently, she is pursuing PhD degree from Amity University, Noida, UP, India in information Technology. Her research interest includes Image Dehazing, Computer Vision and Computer Graphics.

Ashwani K. Dubey received the M.Tech. degree in Instrumentation and Control Engineering from Maharshi Dayanand University, Rohtak, Haryana, India, in 2007, and Ph. D degree from the Department Electrical Engineering, Faculty of Engineering and Technology, Jamia Millia Islamia (A Central Govt. University), New Delhi, India, in 2014. Currently, he is a Professor in the Department of Electronics and Communication Engineering, Amity School of Engineering, Amity University, Noida, Uttar Pradesh, India. He has chaired many IEEE international conferences and workshops. His research works include wireless communications, energy optimization of sensor networking, computer and machine vision applications.

Rajiv Kapoor received B. Tech, M. Tech, and PhD degree in Electronics and Communication Engineering. Presently he is Professor with the Department of Electronics and Communication Engineering, Delhi Technological University, New Delhi, India. He filed six patents and published more than fifty papers in journal of repute. He has published two books with Pearson Inc., India. He has chaired many IEEE international conferences and workshops. His current areas of interests are Image Processing, Computer Vision, Signal Processing, Power Quality, and Cognitive Radio.