

Empirical Assessment of Transfer Learning Techniques for Surgical Tools Classification

Shweta Bali, Shyam Sunder Tyagi

Abstract -Automated surgical tool classification in the medical images is a real-time computerized assistance for the surgeons in performing different operations. Deep learning has evolved in every facet of life due to availability of large datasets and emergence of Convolutional Neural Networks (CNN) that have paved the way for development of different image related processes. In the medical field there are number of challenges such as non-availability of datasets, image annotation requires extensive time, imbalanced data. Transfer learning is the process of applying existing pretrained models to the new problem. It is useful in those scenarios where the large datasets are not available, or the new dataset shares visual features with the existing dataset on which the model is pretrained. Most of the pretrained models are trained on ImageNet which is a largescale dataset (1.2 million labelled training images). In this paper we evaluated and explored two different CNN architectures namely VGG16 and MobileNet-v1-1.0-224 on subset of surgical toolset. This paper presents comparative analysis of the techniques using learning curves and different performance metrics.

Keywords-Convolutional neural networks, Data Augmentation, Deep learning, Transfer learning

I. INTRODUCTION

In the past few years, artificial intelligence (AI) has shown promising future in the areas such as intelligent chatbots, self-driving vehicles, assistants etc. The remarkable success of deep learning can be attributed to a number of key factors such as generation of a lot of computer accessible data, benchmarked datasets (like ImageNet,), increased computing power due to the GPU's (Graphical Processing Units) as they have higher capacity than CPU and capability of parallel processing of threads. The main benefit that deep learning [1] has over the traditional machine learning algorithms is ability of the network to automatically decide the characteristics of the dataset that will be used as features for labelling it. Deep learning algorithms utilize Convolutional neural network (CNN) [3] as the backbone for the architecture. Deep learning is used in each and every sector of the society. There are a number of applications such as pedestrian detection [2], medical image analysis [4], object recognition, image classification etc. Automation has touched each and every part of the medical fields. The robotic arm can test the existence of all the tools for diagnosis. It reduces the preparation time of the nurses. The main advantages of AI-based systems are making the task performed by humans convenient, efficient, error-free, fast.

Revised Manuscript Received on December 12, 2019.

Shweta Bali, Research Scholar, Department of Computer Science & Engineering (CSE), FET Manav Rachna International Institute of Research and Studies (MRIIRS), Faridabad.

Dr. S. S. Tyagi, Professor, Computer Engineering and Dean, Manav Rachna International Institute of Research and Studies (MRIIRS), Faridabad.

Medical Imaging [5] is one of the key areas where the researchers are doing research to assist the doctors in diagnosis as well as prediction of the diseases. Machine learning conventional techniques[6] used handcrafted features and different classification methods such as Support Vector Machine (SVM), Neural Networks(NN) for images related tasks. In contrast to the earlier developed techniques, deep learning techniques are based on Convolutional neural networks which are used for detecting the objects in the images or videos. CNNs have already proved their performance in different computer vision related tasks. In [7] CNNs are employed for performing automated tool segmentation in videos using the transfer learning on the real as well as artificial data. They concluded that the deep learning algorithms can further be explored for developing computer aided system(CAS) even if the dataset size is small. Different CNN-based techniques are discussed for computer aided diagnosis (CAD) applications [8] and the results showed that deep architectures with 8 and 22 layers is helpful in case of limited datasets and transfer learning is also useful to solve such problems.

The rest of the paper is structured as: In Section II related literature is reviewed. Section III discussed the methodology used for transfer learning. Section IV outlines the two network architectures that are used for classification of surgical tools in images. Section V explains the results of the experimentation work done and the results which helps in analyzing the models. Section VI concludes the paper with future scope.

II. RELATED WORK

Deep learning has achieved a great success in object detection problem due to existence of Convolutional Neural Networks (CNNs). In 2012, the challenge saw a big improvement in performance, when the authors[9] trained a convolutional neural network (CNN) based deep network architecture for classification of large dataset of high-resolution images into 1000 different classes. They also proposed GPU efficient implementation of 2D convolution operation. As there are 60 million parameters so it is difficult for the network to learn so many parameters without overfitting. Data augmentation and dropout techniques were used to reduce overfitting on the images and thus generalizing the model. The network architecture consists of five convolutional layers and three fully connected layers. The experiments brought great amount of reduction in Top 5 classification accuracy and localization error.

The study suggested use of larger and deeper convolutional nets to the videos to extract more information. They also suggested using unsupervised pre-training to simplify the experiments.

In [18] the authors introduced a visualization technique that was used to find problems to obtain better results provides detail of the intermediate layers and the classifier. The visualization technique used multi-layered deconvolutional network (deconvnet). ImageNet trained model performed well on datasets such as Caltech-101, Caltech-256 whereas for trained models built using smaller datasets such as PASCAL, these models generalize less but the performance can be enhanced by using various different types of loss functions that are used to detect multiple objects per image. In [6] work dealing with visual surgical tool detection have been reviewed. They authors concluded that computer vision techniques are promising in the direction of detection, tracking, localization. In [19] the authors have automated the process of segmentation and categorization of different cataract surgery tasks. In [20] automated recognition from the cataract videos is discussed.

III. NETWORK ARCHITECTURES

Convolutional neural networks are the networks comprising of three major players namely convolutional layer, max-pooling layer and fully connected layer. convolutional layer filters the images with the help of different filters resulting in discovery of patterns in the images. Then ReLU activation is performed on those patterns and dimensionality reduction is done using the max-pooling layer. Classification is performed using the fully connected layers.

A. Transfer Learning Process

In transfer learning the features learnt on the large dataset ImageNet[11] as a feature extractor where the last fully connected layer or the fully connected layers of the pretrained network is/are removed and replaced by another layer that represents classes of new problems. It is advantageous in case we have limited training data on the new problem. In this only the weights of the new layers are updated. The weights of original network are locked which prevents overfitting of new data or by fine tuning the whole network, which is trained completely, and all the network weights are updated. It is useful in the scenario where we have large amount of the training data. Fig.1. outlines the general transfer learning process which is used in this paper.

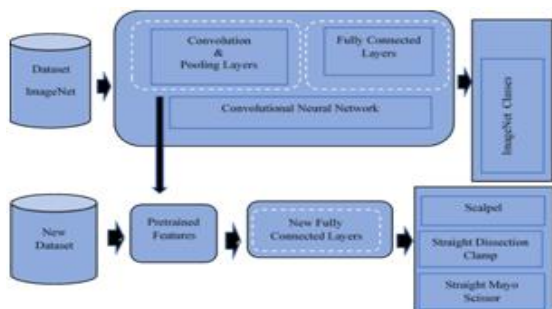


Fig.1. Transfer Learning process

In this paper we compared and analyzed two pretrained models namely VGG (Visual Geometric Group)16 and MobileNet-v1-1.0-224 present in Keras [10].

B. VGG16 Network Architecture

In 2014, the authors came up with the VGG network architecture [13]. The architecture consists of 16 convolutional layers with ReLU [9] activation function and some of them are followed by max-pooling layers. The input is a collection of 224x224 RGB images which are passed through 3x3 receptive field with stride of 1. Spatial pooling is performed by max pooling that uses a receptive field of 2x2 with stride of 2. The stack of 13 convolutional layers is followed by three fully connected layers. The first two have 4096 nodes whereas the third layer performs 1000 classes classification. ReLU activation function is used to handle non-linearity. The final layer uses SoftMax function.

C. MobileNet Network Architecture

In 2017, the authors [14] came up with a class of lightweight models for mobile and embedded applications. They used depth-wise separable convolutions that factorizes convolution into a depth-wise convolution and a 1x1 point-wise convolution for developing the models. All the layers are followed by Batch Normalization layer and ReLU nonlinearity except the last fully connected layer. Down sampling operation is managed with the help of stride convolution in the depth-wise convolutions as well as in the first layer. The final average pooling operation causes reduction in spatial resolution to reduce to a value of 1 before the fully connected layer.

IV. METHODOLOGY FOR TRANSFER LEARNING

The methodology for Image Classification for transfer learning using pretrained models consists of various steps from loading the dataset to partitioning to data augmentation to model construction followed by training, validation and testing steps and then prediction of the results and performing the analysis as outlined in Fig.2.

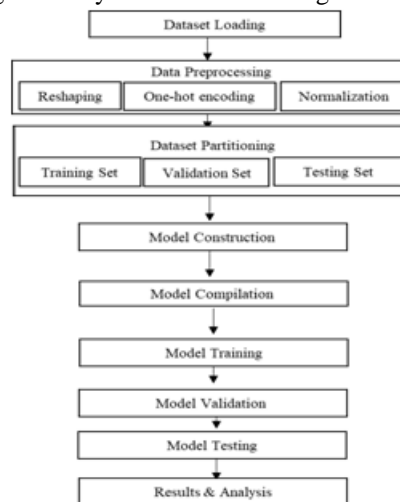


Fig.2. Methodology for Transfer Learning

The above-mentioned steps are explained below:

A. Dataset Loading: The first step is loading the desired dataset. We have taken a subset of surgical dataset [12] classify the instrument images with 640*480 resolution in three categories. The dataset consists of 1069 training dataset which are further divided 748 training images and 321 validation images and 397 testing images. Fig. 3. shows the sample images from the dataset.

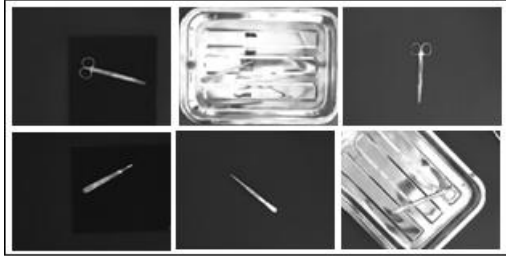


Fig.3. Sample Images of the dataset

B. Data Preprocessing: For the learning algorithm to achieve good performance it needs to be fed with good quality data and also useful information. In this step the images are reshaped into image size of 224*224. Data is normalized using feature scaling which scales all inputs in the range of [0,1]. Also, the categorical variables are hot encoded for feeding into the desired model.

C. Dataset Partitioning: The training set is used for training and fitting different models. For the process of model selection which deals with the selection of optimal values of hyperparameters some portion of training set is cast-off as validation set. Testing set is used for judging the generalization performance.

D. Model Construction: In transfer learning the representations learnt by the previous network are used to extract the meaningful features called as bottleneck features of the new dataset images. The original network before the fully connected network is frozen which means that the weights of these frozen layers are not updated. The activation map generated from the last convolutional layer are fed to the newly constructed fully connected network which acts as the classifier. The final layer uses SoftMax function. The layers of the new network were initialized with weights pretrained on the ImageNet [5] dataset. The reason behind not training the complete model is that the base convolutional layers extract the generic features whereas the classification part deals with the specific features related to the problem. Dropout technique is used to further improve the accuracy of the networks.

E. Model Compilation: The model needs to be compiled before training the same. Since there are three classes, we use categorical cross-entropy loss.

F. Model Training: The next step is training the different models. Two models VGG-16 and MobileNet are trained one after the other on 748 training images.

G. Model Validation: In this step the model is validated on 321 validation images belonging to three different classes of tools. The overfitting and underfitting of the model are judged on the basis of the learning curves for accuracy and loss.

H. Model Testing: After the two models are validated, they are tested on 397 images and the different performance metrics are used for evaluating the models and for analysis. Confusion Matrix is a square matrix where the rows (x-axis) represents the testing examples that belong to predicted category whereas each column (y-axis) represents the testing examples that belong to actual category. The diagonal represents the number of samples in the testing data for which the predicted label is same as the true label. Higher the value in the diagonals indicates the large number of correct predictions and better the model. Confusion matrix can be used to derive different metrics [15]. The performance of deep learning model on training and validating set can be judged with the learning curve [16] in terms of accuracy and loss w.r.t number of epochs. As shown in Figure 3 and Figure 4, the accuracy increases while the loss decreases through the training process until convergence. Accuracy (Eq.1) and Error rate (Eq. 2) are the most frequently used measures when evaluating the classification results. Accuracy is the proportion of the observations predicted correctly.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Error Rate = 1 - Accuracy \quad (2)$$

In Eq.1 TP (True Positives) and FN (False Negatives) denotes the number of observations that fit positive class and are projected correctly and incorrectly respectively, TN (True Negatives), FP (False Positives) is equal to the number of observations that belong to negative class and are forecasted correctly and incorrectly respectively. Both the above-mentioned measures are frequently used for evaluating the classification results. In case there is imbalance in classes as in that case these will fail so the following counter measures are being used. Precision (Eq. 3) is the percentage of positively labeled examples that are actually positive. Precision does not give information on the samples from the positive class that are mislabeled as positive. Recall (Eq. 4) is the percentage of positively labeled examples among total number of observations that really belong to the positive class. Recall does not give information on the samples from the negative class that are mislabeled as positive.

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

F1 Score (Eq. 5) which is the weighted average of two values namely Precision and Recall. Higher the value of F1 Score better the model.

$$F1 \text{ Score} = 2 \times \frac{\text{Precision}}{\text{Precision} + \text{Recall}} \quad (5)$$

V. EXPERIMENTS AND RESULTS

Experimentation is performed on COLAB tool from Google with Tesla K80 GPU, pci bus id: 0000:00:04.0, compute capability: 3.7. Both the models VGG16 and MobileNet-v1-1.0-224 are present in Keras library. The optimizer Adam is used with a learning rate of (.0001). Fig.4. & Fig.5. show learning curves for training and validation accuracy and loss for VGG16 model over time period of 20 epochs respectively. Keras Library is used for implementing the network architectures.

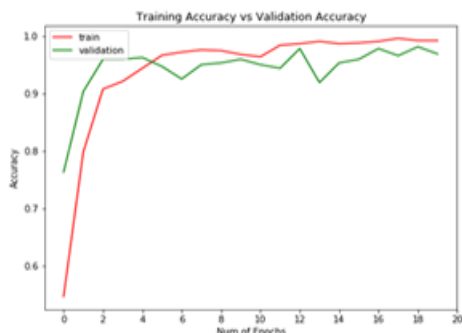


Fig.4. Learning Curves of Training Accuracy and Validation Accuracy for VGG16 Model

The VGG16 model was trained and achieved average training accuracy of 99.20% and average training Loss of 2.24%. The average validation accuracy for the model was 96.88% with an average validation loss of 10.07%. Table-I shows epochs summary for VGG16 model. The model is good as the training and the validation accuracy curves in Fig.4. are close to each other. The validation loss of the model is also similar to that of the training loss, so the model has fitted the data properly.

Table- I: Epochs Summary for VGG16 Model

Epochs	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
1	0.9452	0.5468	0.6145	0.7632
2	0.5052	0.7981	0.3235	0.9034
3	0.2819	0.9078	0.1697	0.9595
4	0.2186	0.9211	0.1595	0.9595
5	0.1654	0.9439	0.1309	0.9626
6	0.1122	0.9666	0.1435	0.9470
7	0.0994	0.9719	0.2098	0.9252
8	0.0763	0.9759	0.1380	0.9502
9	0.0672	0.9746	0.1300	0.9533
10	0.0748	0.9679	0.1265	0.9595
11	0.0944	0.9639	0.1225	0.9502
12	0.0537	0.9840	0.1701	0.9439
13	0.0397	0.9866	0.0823	0.9782
14	0.0325	0.9906	0.2981	0.9190
15	0.0481	0.9866	0.1322	0.9533
16	0.0393	0.9880	0.1198	0.9595
17	0.0294	0.9906	0.0704	0.9782
18	0.0193	0.9960	0.1022	0.9657
19	0.0246	0.9920	0.0659	0.9813
20	0.0224	0.9920	0.1007	0.9688



Fig.5. Learning Curve of Training Loss and Validation Loss for VGG16 Model

Fig. 6. and Fig.7. show the learning curve for training and validation accuracy and loss of MobileNet-v1-1.0-224 model over 20 epochs respectively. The model was trained and achieved average training accuracy of 99.87% and average training loss of 1.05%. The average validation accuracy for the model was 99.07% with an average validation loss of 2.33%. Table-II shows epochs summary for MobileNet-v1-1.0-224 model. The model is better than the VGG16 model as the training and validation accuracy curves in Fig.6. are closer to each other than for VGG16. The validation loss of the model is overlapping at the later epochs that shows that the model has performed better than VGG16 model and the model has fitted data properly.



Fig. 6. Learning Curves of Training Accuracy and Validation Accuracy for MobileNet-v1-1.0-224 Model

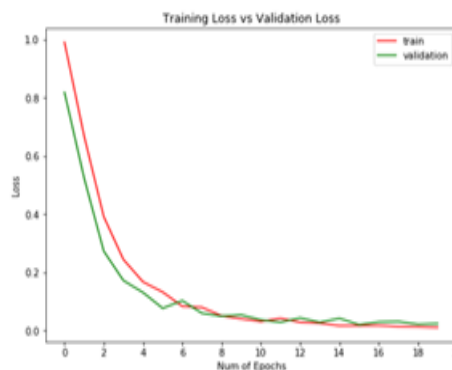


Fig. 7. Learning Curves for Training Loss and Validation Loss for MobileNet-v1-1.0-224 Model

Table- II: Epochs Summary for MobileNet-v1-1.0-224 Model

Epochs	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
1	0.9911	0.4652	0.8188	0.6262
2	0.6690	0.7139	0.5244	0.7445
3	0.3913	0.8489	0.2726	0.9097
4	0.2440	0.9265	0.1717	0.9502
5	0.1670	0.9465	0.1304	0.9626
6	0.1320	0.9626	0.0764	0.9938
7	0.0829	0.9866	0.1023	0.9502
8	0.0795	0.9746	0.0592	0.9626
9	0.0501	0.9853	0.0492	0.9938
10	0.0398	0.9920	0.0534	0.9502
11	0.0307	0.9947	0.0361	0.9813
12	0.0418	0.9880	0.0281	0.9751
13	0.0279	0.9947	0.0430	0.9875
14	0.0251	0.9880	0.0278	0.9907
15	0.0164	0.9947	0.0414	0.9844
16	0.0168	0.9880	0.0199	0.938
17	0.0174	0.9947	0.0290	0.9907
18	0.0135	0.9973	0.0304	0.9875
19	0.0124	0.9973	0.0213	0.9938
20	0.0105	0.9987	0.0233	0.9907

Fig.8. and Fig. 9. shows the confusion matrix for VGG16 and MobileNet-v1-1.0-224model respectively for the different classes of the surgical toolset. From the above figures it can be seen VGG16 and MobileNet-v1-1.0-224model correctly identified 134 Straight Mayo Scissor tool. MobileNet-v1-1.0-224model identified 129 Straight Dissection Clamp correctly and 1 as incorrect whereas VGG16 could only identify 120 correctly and 10 incorrect. A total of 132 Scalpel were identified correctly by the MobileNet-v1-1.0-224model whereas 1 was incorrect. On the other hand, VGG16 identified 130 as correct and 3 incorrect. The performance of the two models is close to each other.

		Predicted Category		
		Straight Dissection Clamp	Straight Mayo Scissor	Scalpel
Actual Category	Straight Dissection Clamp	120	1	9
	Straight Mayo Scissor	0	134	0
	Straight Mayo Scissor	2	1	130

Fig. 8. Confusion Matrix for VGG16 Model

		Predicted Category		
		Straight Dissection Clamp	Straight Mayo Scissor	Scalpel
Actual Category	Straight Dissection Clamp	129	0	1
	Straight Mayo Scissor	0	134	0
	Straight Mayo	0	1	132

Fig. 9. Confusion Matrix for MobileNet-v1-1.0-224 Model

Table-III and Table-IV further strengthen our claim that the models have not overfitted and the results for the MobileNet-v1-1.0-224 model are better for the dataset used.

Table- III: Performance Metrics for VGG16 Model

Surgical Tool	Precision	Recall	F1 Score
Straight Dissection Clamp	0.98	0.92	1.0
Straight Mayo Scissor	0.98	1.00	0.95
Scalpel	0.93	0.97	0.97

Table- IV: Performance Metrics for MobileNet-v1-1.0-224 Model

Surgical Tool	Precision	Recall	F1 Score
Straight Dissection Clamp	1.00	0.99	1.00
Straight Mayo Scissor	0.99	1.00	1.00
Scalpel	0.99	0.99	0.99

VI. CONCLUSION AND FUTURE SCOPE

With the help of this paper, a comparison between the existing pretrained models VGG16 and MobileNet can be drawn in terms of their performance to detect the surgical instruments in the images. It has been found that MobileNet performed better in classifying instruments from small dataset of surgical instruments. MobileNet on the other hand requires less training time as compared to VGG16 network. Also, the number of the parameters MobileNet are less in comparison to VGG16. Data augmentation and Dropout techniques are also performed to improve the accuracy of the network. This analysis is useful for making a choice between the pretrained models. Also, the results can be used for the purpose of designing automated systems to directly classify the instrument according to its category. The performance of the existing models can be further be checked different surgical problems.

REFERENCES

1. Y. LeCun, Y. Bengio, G. Hinton, "Deep learning," *nature*, May 2015, 521(7553), pp. 436-444.
2. S. Bali and S.S. Tyagi, "A Review of Vision-Based Pedestrian Detection Techniques," *International Journal of Advanced Studies of Scientific Research*, 3(9), 2018.
3. G. Litjens, T. Kooi, B.E. Bejnordi, A.A.A. Setio, F. Ciompi, M. Ghafoorian, J.A. Van Der Laak, B. Van Ginneken and C.I. Sánchez, "A survey on deep learning in medical image analysis," *Medical image analysis*, Dec 2017, 42, pp.60-88.
4. M.I. Razzak, S. Naz, and A. Zaib, "Deep learning for medical image processing: Overview, challenges and the future," *Classification in BioApps*, 2018, pp. 323-350, Springer, Cham.
5. J. Ker, L. Wang, J. Rao, T. Lim, "Deep learning applications in medical image analysis," *IEEE Access* (6), 2018, pp.9375-9389.
6. [6] D. Bouget, M. Allan, D. Stoyanov, and P. Jannin, "Vision-based and marker-less surgical tool detection and tracking: A review of the literature," *Medical Image Analysis*, vol. 35, pp. 633-654, Jan. 2017.

7. O. Zisimopoulos, E. Flouty, M. Stacey, S. Muscroft, S.P. Giataganas, J. Nehme, A. Chow, D. Stoyanov, "Can surgical simulation be used to train detection and classification of neural networks?", *Healthcare technology letters*, Nov 2017,4(5), pp.216-222.
8. H.C. Shin, H.R. Roth, M. Gao, L. Lu, L., Z. Xu, Z., I. Nogues, J. Yao, D. Mollura, R.M. Summers, Feb. 2016, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE transactions on medical imaging*, 35(5), pp.1285-1298.
9. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, 2012, pp. 1097-1105.
10. Keras.io. (2019). Keras Documentation. [online] Available at: <https://keras.io/> [Accessed 15 Nov.2019].
11. J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," IEEE conference on computer vision and pattern recognition, Jun. 2009, pp. 248-255.
12. Diana Lavado (2018, September), Labeled Surgical Tools and Images, Version 1. Retrieved November 15, 2019 from <https://www.kaggle.com/dilavado/labeled-surgical-tools>.
13. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", CoRR, abs/1409.1556, 2014.
14. A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M., Andreetto, H. Adam, H., "Mobilenets: Efficient convolutional neural networks for mobile vision applications," Apr. 2017, *arXiv preprint arXiv:1704.04861*.
15. J.M. Johnson, T.M. Khoshgoftaar, "Survey on deep learning with class imbalance," *Journal of Big Data*, Dec 2019,6(1), pp. 27.
16. C. Perlich, "Learning curves in machine learning. Encyclopedia of machine learning," 2010, pp.577-580.
17. J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" Proceedings of the 27th International Conference on Neural Information Processing Systems, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, pp. 3320–3328.
18. M.D. Zeiler, R. Fergus, R., "Visualizing and understanding convolutional networks," European conference on computer vision, September, pp. 818-833, Springer, Cham.
19. G. Quellec, M. Lamard, B. Cochener, and G. Cazuguel, "Real-time segmentation and recognition of surgical tasks in cataract surgery videos," *IEEE Transactions on Medical Imaging*, Dec2014, vol. 33, no. 12, pp. 2352–2360
20. G. Quellec, M. Lamard, B. Cochener, and G. Cazuguel, "Real-time task recognition in cataract surgery videos using adaptive spatiotemporal polynomials," *IEEE Transactions on Medical Imaging*, Apr. 2015, vol. 34, no. 4, pp. 877–887.
21. V. Arora, and S.S. Tyagi, "Performance evaluation of load balancing policies across virtual machines in a data center," International Conference on Reliability Optimization and Information Technology (ICROIT), February 2014, pp. 384-387, IEEE.
22. S.B. KaebbehYaeghoobi, M.K. Soni, and S.S. Tyagi, "Dynamic and real-time sleep schedule protocols for energy efficiency in WSNs," *International Journal of Computer Network and Information Security (IJCNIS)*, 2016, 8(1), pp.9-17.

did his M.E from BITS Pilani and B.Tech in Computer Technology from Nagpur University. He is having an experience of more than 27 years in academics/teaching and research. He has been holding various academic and administrative positions during his career and contributing towards research. He is a senior member of various professional organizations like IEEE, ACM, CSI, QCI, ASQ etc. He is past chair, IEEE Computer Society, Delhi Section. There are more than 70 publications to his credit in National and International Journals. He is associated as an editor/reviewer of various journals. He has guided 06 Ph.Ds. and several M. Tech Thesis and guiding Ph. D scholars in the field of Software Defined Networking, Cloud Computing, Adhoc Networks, Wireless Security etc.

AUTHOR PROFILE



Shweta Bali is a research scholar in department of Computer Science & Engineering (CSE), FET Manav Rachna International Institute of Research and Studies (MRIIRS), Faridabad. She completed her M.E from Punjab Engineering College (Deemed to be University), Chandigarh in the year 2006 and B.Tech in Computer Science from Punjab Technical University, Jalandhar in the year 2003. She has 6 research papers in International Journals and in proceedings of International and National Conferences. Her area of interest in machine learning and computer vision.



Dr. S. S. Tyagi is presently working as a Professor, Computer Engineering and Dean at Manav Rachna International Institute of Research and Studies (MRIIRS), Faridabad. He completed his Ph.D in Computer Science and Engineering from Kurukshetra University, Kurukshetra. He