2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA)
Galgotias University, Greater Noida, UP, India. Oct 30-31, 2020

Explainable Deep Learning Methods for Medical Imaging Applications

Sunil Kumar Vuppala*, Madhumita Behera, He Jack[$], Nagaraju Bussa
* Senior Member, IEEE, Philips Research, Philips India Ltd, Bangalore, India
[$] Philips USA Cambridge
*sunil.vuppala@ieee.org, nagaraju.bussa@philips.com

Abstract— **This paper discusses explainable deep learning approaches for medical imaging applications. This includes evaluation of three different feature visualization approaches for visualizing deep neural network models, developed with medical domain images. The idea behind using these approaches is to unbox the internals of deep neural network, specifically convolutional neural networks (CNN) by providing a step by step learning of these models. These approaches could help clinicians to have confidence in complex deep learning models, as opposed to treating those as black boxes as these approaches can generate a meaningful view of the model layers and their feature maps. Also, these approaches can be adopted by data scientists to improve their model performances by looking at the model behavior in each layer. We have implemented three approaches namely Activation Map, Deconvolution and Grad-CAM localization in Keras with Tensorflow background and have validated the results with CNN models developed with natural images. We have also been able to generate these visualizations for models with even filter sizes with activation map and deconvolution approaches. These methods play crucial role in getting approval from regulatory authorities.**

Keywords— ***activation map, CNN, deconvolution, features, visualization, activation, grad-cam, Keras, tensorflow, interpretable deep learning, explainable deep learning***

I. INTRODUCTION

Deep neural networks have revolutionized computer vision and speech recognition in the last few years and have surpassed the state-of-the-art results of many machine learning algorithms. The capabilities of deep neural networks have been tremendous including CNNs, which have made significant breakthroughs [1]. Despite such success, it has always been a great challenge so far to understand what these networks are learning internally to arrive at a decision.

There lies the problem for adoption of deep learning model-based technology in healthcare setting as deep learning models are often perceived as a "black box". It is difficult for clinicians to understand **how it works and why it works.** Because the mechanisms and reasoning processes are often hidden from technology users, when it fails, there is no way to examine how it happens. The explainability plays an important role in getting the approval from regulatory authorities such as FDA (Food and Drug Administration) of US. Computer aided detection has not produced the gains that might be expected also in part due to poor interaction between clinician and artificial intelligence (AI). Because in most of the cases in screening settings are negative, even a good AI will typically produce many more false positives than true positives. Consequently, the positive predictive value of the information is low. This reduces clinicians' trust and inhibits clinical adoption of AI that could save lives.

Opening the black box to add interpretability of the technology is crucial for users to build trust with the technology, thereby improving adoption of deep learning model-based technologies in healthcare. In order to understand deep neural networks, in our research we reviewed, implemented, and validated a few approaches that are prevalent among deep learning experts. These approaches have proven to be great tools to dig deeper into deep neural networks.

Three different model layer visualization has been explored to understand how the deep learning models are breaking the data internally and learning to come up with a solution for a deep neural network. The purpose this paper is to capture all the functional details of the approaches and discuss the implementation and verification details in Keras with Tensorflow. In this paper, we discuss the three different approaches for deep neural network layer visualization. Namely:

1. Activation Map
2. Deconvolution
3. Localization

The main contribution is to share the ***implementation and evaluation of these methods in medical imaging domain*** as part of Explainable AI both for data scientists and regulators. Our scope of layer visualization in this paper is limited to these three approaches only. However, there are a few such approaches, tried and tested by deep learning enthusiasts, which we briefly describe in literature. We highlight the implementation challenges and how to apply them in healthcare domain problems. Each neuron/filter in a CNN is responsible for detecting a specific feature. In initial layers, they are responsible for detecting features such as edges, colours etc. Complex features such as object locations and specific shapes can be learned in deeper layers. The features detected by the higher layers are more discriminative in nature making separation into various classes. This is important for both the data scientists and the clinicians.

In healthcare / medical imaging domain, these approaches may build trust between the clinicians and the technology and make the reasoning process more transparent to the users. Instead of looking only at the result, with explainable models a clinician can make a more informed decision whether to accept or reject the result. This will increase clinical adoption of deep learning-based technologies. This also play role in satisfying regulatory requirements.

978-1-7281-6324-6/20/$31.00 ©2020 IEEE

The rest of the paper is organized as per the following sections. Section II outlines an overview of existing feature visualization approaches. The approaches are discussed in section III. Various implementation details and results are discussed in section IV. The verification of results is analyzed in section V. Section VII contains conclusion and future directions.

II. LITERATURE SURVEY

A. List of Methods of feature visualization

There are many approaches in literature for feature visualization. We shall describe couple of those approaches in this section.

- Layer Activations [2]: During the forward-pass of a Convnet, the simplest way to visualization is to look at the activations of each network layer. Activations of the model layer for initial layers look blobby and dense and then as we go deeper into the network, it becomes sparse and localized. Dead filters (due to high learning rates) also possible- if the activation maps are zero for many different inputs.
- Conv/FC filters [1, 2]: Directly visualize the weights, as they are the most interpretable on the first conv layer. The weights are useful because well-trained networks display smooth filters without any noisy patterns (indicator of a less-trained network or overfitting).
- Retrieving images that maximally activate a neuron [3]: Takes large datasets, feeds to the network, and keeps track of which images maximally activate some neurons. We can visualize the images to get an understanding of what the neuron is looking for in its receptive field.
- Embedding the codes with t-SNE [4]: ConvNets (Convolutional Neural Networks) are interpreted as gradually transforming the images into a representation in which the classes are separable by a linear classifier. Embedding images into two dimensions so that their low-dimensional representation has approximately equal distances than their high-dimensional representation.
- Occluding parts of the image: Plotting the probability of class of interest as a function of the position of an occlude object.
- Deconvolution [5]: It interprets the feature activities, by mapping the activities back to the input pixel space. The reconstruction obtained from a single activation resembles a small piece of the original input image, with structured weighted according to their contribution toward to the feature activation. Since the model is trained discriminatively, they implicitly show which parts of the input image are discriminative. The shortlisted approaches are discussed in next section in detail. We practically implemented these approaches to explain our deep learning models to clinicians and regulatory authorities.

III. VISUALIZATION APPROCHES

A. Activation Map

Each feature map receives some activation during the forward pass of a network. Thus, the most straight-forward visualization technique is to show the activations of the network during the forward pass. The activations usually start out looking relatively blobby and dense, but as the training progresses the activations usually become sparser and more localized. One pitfall that can be easily noticed with this visualization is that some activation maps may be all zero for many different inputs, which can indicate dead filters.

In each layer, the feature maps learn differently. The beginning layers respond to corners and other edge/color conjunctions. As we project further into deeper layers, it has more complex invariances, capturing similar textures. Last layer shows significant variation, mostly object localization and is more class specific.

B. Deconvolution

To understand a convent, requires interpreting the feature activity in intermediate layers. This can be mapped back to the input pixel space to see what input pattern has originally caused a given activation in the feature maps. This mapping can be achieved by deconvolution.

Deconvolution is a unique technique to map each pixel learning to the end of the input layer of the network. However, we found this approach to be limited to few layers as support of Keras with Tensorflow package is concerned. Not all intermediated layers are handled for transpose. So far, it can support convolution and maxpooling layers. If the CNN model contains layers other than these two layers, then the deconvnet (deconvolution network) construction is not considered to be proper. The deconvnet thus constructed for such a model with transposed convolution layer and unpooling layers only, gets a wrong pixel mapping even though some visualization is being generated. This visualization may not be correct. However, this can be corrected if deconvolution for all the layers are possible. In our current results, we have shown results with convolution and max pooling layers only.

C. Localization

Saliency mapping [6], also called attention mapping or localization techniques [7, 8], highlights areas of interest on the original medical images that are most relevant for a diagnosis decision (e.g., with cancer), which is like human attention heatmap. By visualizing the key areas that most relevant to such a decision, a clinician can explore how a deep learning model "sees" on the original image, and understands the rationale of a particular decision (e.g., a particular pattern) even when the decision is not correct, therefore building trust with the model.

There is a potential to greatly improve the technology interpretability by using saliency mapping based visualization. Clinicians' training and domain knowledge makes their visual attention very selective—typically focus on certain areas with specific patterns, e.g., large nodules on a lung CT image, and this visual scanning pattern can be visualized as a heatmap.

The rectified output then is normalized and resized. One thing worth of noting is that the last convolution layer is an open question, and based on model structure, it can be a convolution layer, pooling layer, and merge layer, which may be best decided by model developers and/or clinicians.

IV. IMPLEMENTATION AND RESULTS

The implementation of all these approaches are done in python with Keras 2.x and Tensorflow 1.x. Each of these approaches use Keras function for deep learning related operation. After developing these approaches, we have used three CNN models developed with medical domain images in order to generate the visualizations. The CNN models are listed as follows:

1. **Mammography** model prepared by us with 22 layers. 600s of training images from Digital Database for Screening Mammography (DDSM) [9]
2. **Lung CT** is developed by Philips Research team with 318 layers for cancer identification with 1000s of training data.
3. **MRI** segmentation – Prepared from open source with 500 layers [10]

We have also tested our approaches with CNN models with VGG16, VGG19 [11] to verify the results. In below sub sections, the implementation techniques are explained for each of the approaches. We could see the visualization results for both natural and clinical image CNN models are very much similar in terms of learning realization.

A. *Implementation of Activation map:*

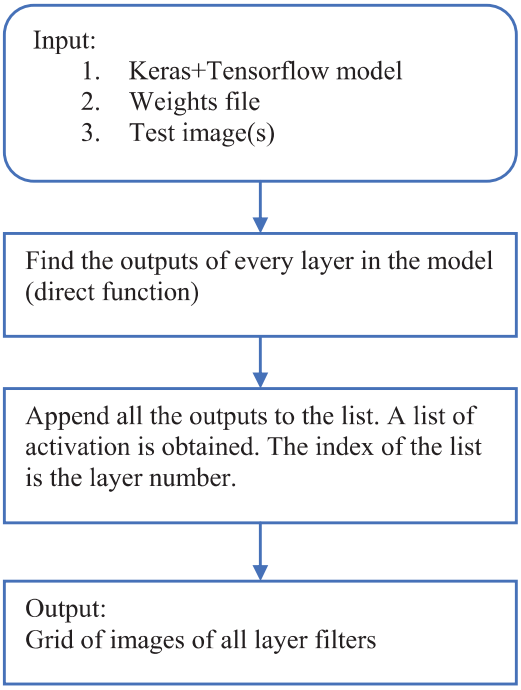

Fig. 1. Activation map implantation

First the CNN model is read into Keras along with a test image where the test image is a preprocessed image like the training images for the model. For a particular layer the feature maps are identified. If we are interested in a particular feature map or for all feature maps of a layer, the activations are saved as an image file. These saved activations saved as an image file show the learning of the feature maps. In section IV we have shown these results. Fig.1 shows a flow chart of the implementation of activation map.

Visualization Results:

The Fig.2 shows the grid of activation map of all the filters in the layer 0 and below Fig.3 shows the grid of activation map for all the filters in the layer 7. Each entry in the grid shows the activation map of a filter in that layer.

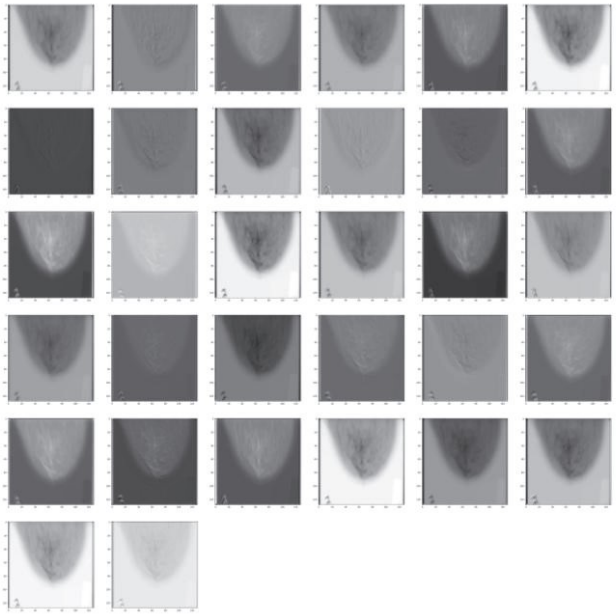

Fig. 2. Activation map of layer 0 for mammographic model

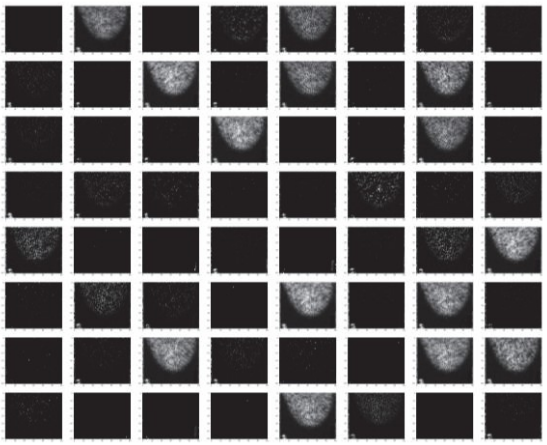

Fig. 3. Visualization of layer 7 for mammographic model

B. *Implementation of Deconvolution:*

First, the activations of desired layer of visualization is saved. Then a deconvnet for this layer is constructed by adding transposed convolution layer for this convolution layer and unpooling layer for max pooling layer. Thus, the deconvnet will have first layer as a transposed convolution 2d layer of the desired layer. Subsequent layers are the transpose of all previous layer to this layer and the last layer as the transpose of the first convolution layer. Thus, a

deconvnet for the desired layer is constructed using the steps mentioned in Fig.4.

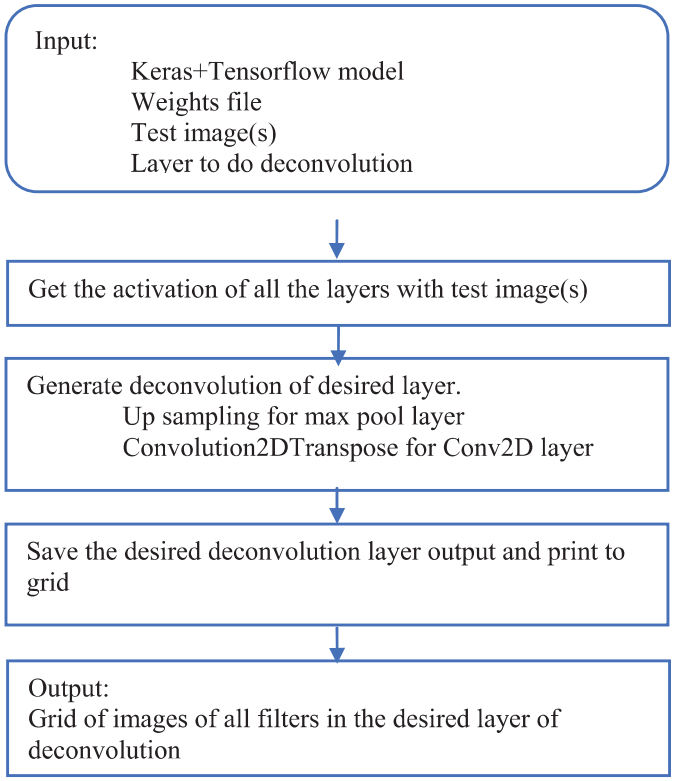

Fig. 4. Implantation of Deconvolution approach

Once the deconvnet is constructed which is basically a transposed model of the original model, a forward pass of the deconvnet is done with saved activation of desired layer. We set all other activations in the layer activation to zero and do this forward pass. This way different activation for the deconvnet is generated, which basically provided the visualization of desired layer of the original model mapping the learning to the input pixels due to the transposed operations with deconvolution.

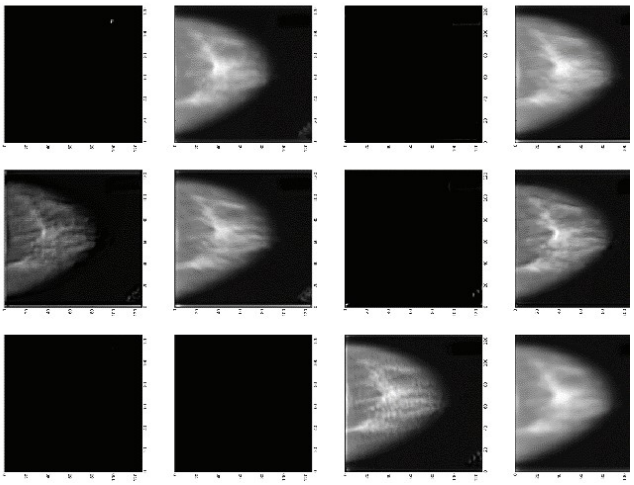

Fig. 5. Snapshot of deconvolution visualization for Layer 0 of mammography image binary classifier

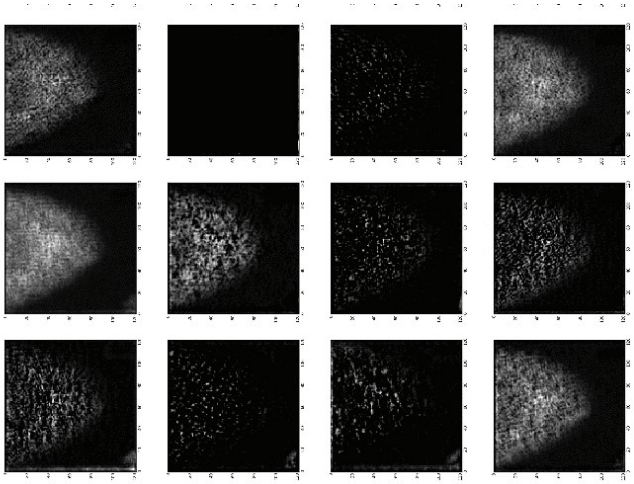

Fig. 6. Snapshot of layer6 of mammographic model deconvolution

Fig.5 and Fig. 6 show the deconvolution results of mammography binary classification model for layer 0 and 6 respectively. From the above figures, the later layers in neural network identify textures and object locations whereas beginning layers identify the low-level features such as edges and colors. Even though these are standard methods from literature, implementation for the 3 different medical models and producing the output is the contribution in this paper.

C. *Implementation of Grad-CAM*

While there are different techniques for saliency mapping, the approach adopted in this project is called Gradient based Class Activation Map, (Grad-CAM), which is class discriminative (i.e., localize the area related to a particular category in the image), and is generalizable to any CNN-based models. The high-level idea of this approach is to calculate the gradient of output class to the last convolutional layer output and identify the most relevant areas in the gradient weighted convolutional layer output, overlay those areas with the original image to highlight the rationale of the class prediction.

The original Class Activation Map works as shown in Fig 7. Grad-CAM [8] improves the method by obtaining weights of each feature activation map from gradients. The weight is calculated by taking the gradients of the score (y) of a particular class (c) with respect to each (k^{th}) activation map (A) and performing global average pooling within each map.

$$\alpha_k^c = \overbrace{\frac{1}{Z}\sum_i\sum_j}^{\text{global average pooling}} \underbrace{\frac{\partial y^c}{\partial A_{ij}^k}}_{\text{gradients via backprop}} \qquad (1)$$

Then the linear combination of feature activation maps weighted by gradients from (1) goes through ReLU.

$$L_{\text{Grad-CAM}}^c = ReLU\underbrace{\left(\sum_k \alpha_k^c A^k\right)}_{\text{linear combination}} \qquad (2)$$

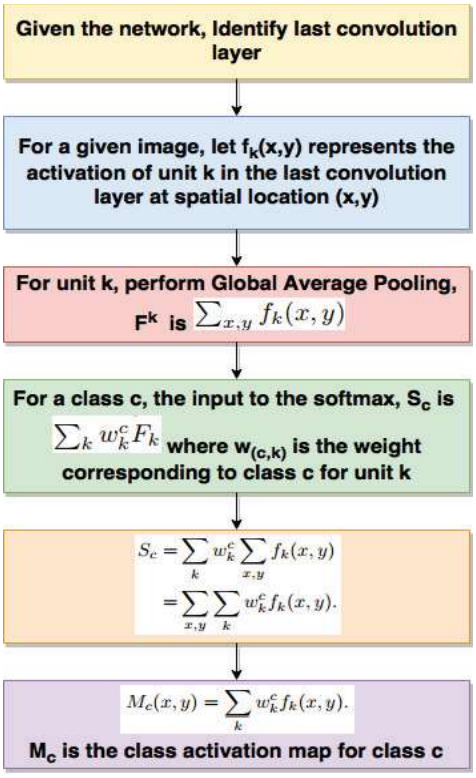

Fig. 7. Flow of class activation map

D. *Localization*

Fig 8 showcases one sample lung CT cancer case from original 3D CT volume. Fig 9 showcases the results when we run Grad-CAM module for class 1 of cancer on these lung CT images. First part of the image is original image, followed by heat map generated from Grad-CAM and the last image is an overlaid heatmap on original image.

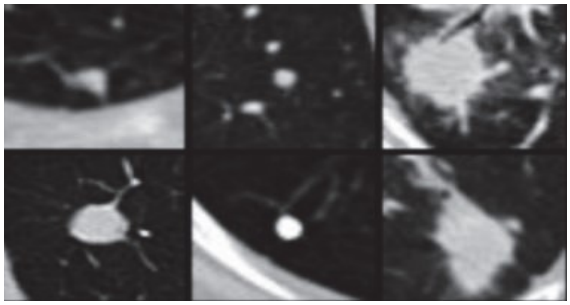

Fig. 8. Sample Lung CT cancer cases

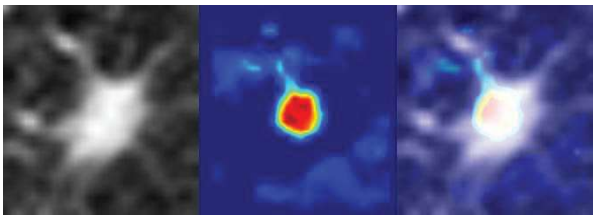

Fig. 9. Grad-CAM visualization for class 1 of cancer in Lung CT (Original image, heatmap and overlaid heatmap on original image)

VI. VERIFICATION OF RESULTS

In order to verify the methods implemented we tried to compare our results with some published and open source work in the same area. We compared our visualization results with these references results keeping dl model and sample images to be same. Thus, we could compare all these three methods with cross platforms like Caffe, PyTorch etc.

As shown in fig 10 we considered VGG16 model from Keras package and used ImageNet weights for transfer learning. We generated the visualizations for each of these approaches. We compared our results with the results from the referenced sources.

Fig 11 showcases the results of our implementation and the reference images captured from reference paper [8]. The identification of tiger cat and boxer dog are matching with the reference results. The heat map can visually represent the features, which helps in decision making process (classification) of the type of animal. This increases the trust of the output from deep learning models for the clinicians.

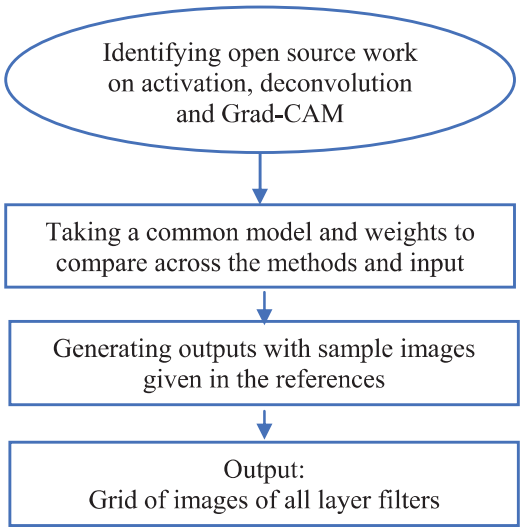

Fig. 10. Flow of verification

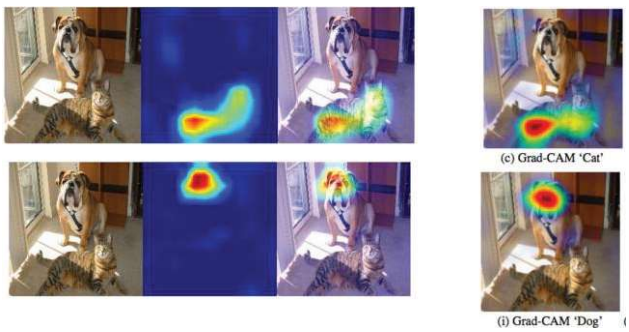

Fig. 11. Grad-CAM verification, Input image and our results (right) reference images from paper [8]

A. *Guided Approach to tune hyper parameters:*

We leveraged some of these capabilities to help with tuning of parameters of a healthcare deep learning model, and therefore, efficiently guiding the training process to fast optimal convergence. Validation of specific parameter tuning could result into improvement of metrics (automated

workflow) – Increase the accuracy and reduce the time of training.

The mentioned approaches help as guided approach of parameter tuning with network and feature visualization for number of filters and filter size, number of Layers, activations, localization and learning rate. For instance, when dead filters are identified, decrease the number of filters and change the filter size. If abstraction is missing, add upper layers in the network. If localization is not working correctly, increase the activation filter and number of filters.

B. *Challenges addressed:*

As part of the verification process, we also addressed couple of challenges.

1) *Even kernel size:*

In general, the kernel for convolution is of odd dimension (3x3, 5x5). We validated the impact of even kernel size on deconvolution. Even though it is common practise to use odd kernel size, we experimented with even kernel size in our model and deconvolution worked fine for even kernel size. These validations are carried out by modifying the mammography image binary classifier model.

2) *Visualization in case of Average pooling layer:*

If the network has pooling layer other than max pooling (Googlenet has average pooling), we cannot visualize the average pooling layers. We tested our modules with Googlenet. We can visualize the activation map and deconvolution for all the supported layers.

VII. CONCLUSIONS

We discussed three approaches of explainable deep learning for feature visualization with medical models. All these approaches demonstrated the internals of the deep learning models. Thus, a thorough understanding of each feature map learning and the variability is also understood. The implemented methods are compared with the results from literature. This is used as a good tool to debug into the model and enhance the learning and tune the parameters and hyper parameters in a visual guided approach for the data scientists. These approaches also helped to build the trust for the clinicians and getting the approvals from regulatory authorities. In future work, we wish to implement new explainable AI methods and compare them with available libraries such as LIME and SHAP.

REFERENCES

[1] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, NIPS 2012

[2] http://cs231n.github.io/understanding-cnn/

[3] Ross G, Jeff D, Trevor D, Jitendra M, Rich feature hierarchies for accurate object detection and semantic segmentation, CoRR, 2013

[4] L.J.P. van der Maaten, G.E. Hinton. Visualizing High-Dimensional Data Using t-SNE. Journal of Machine Learning Research 9(Nov):2579-2605, 2008

[5] M. D. Zeiler, D. Krishnan, G. W. Taylor and R. Fergus, "Deconvolutional networks," 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, 2010, pp. 2528-2535

[6] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps," Proc. the International Conference on Learning Representations, 2014

[7] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Net-works," Proc. the European Conference on Computer Vision, pp. 818–833, 2014.

[8] Selvaraju R, C Michael, Abhishek D, Ramakrishna V, Devi P, Dhruv B, Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization, eprint arXiv:1610.02391 2006

[9] Chris Rose, Daniele Turi, Alan Williams, Katy Wolstencroft and Chris Taylor, Web services for the DDSM and digital mammography research, IWDM,2006.

[10] Akkus Z, Galimzianova A, Hoogi A, Rubin DL, Erickson BJ. Deep Learning for Brain MRI Segmentation: State of the Art and Future Directions. J Digit Imaging. 2017;30(4):449–459. doi:10.1007/s10278-017-9983-4

[11] Karen Simonyan, Andrew Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, ICLR 2015