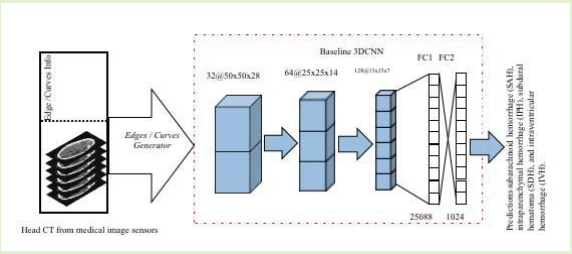
Shallow 3D CNN for Detecting Acute Brain Hemorrhage from Medical Imaging Sensors

Satya P. Singh, Lipo Wang, Sukrit Gupta, Balázs Gulyás, and Parasuraman Padmanabhan*

***Abstract*—Successive layers in convolutional neural networks (CNN) extract different features from input images. Applications of CNNs to detect abnormalities in the 2D images or 3D volumes of body organs have recently become popular. However, computer-aided detection of diseases using deep CNN is challenging due to the absence of a large set of training medical images/scans and the relatively small and hard to detect abnormalities. In this paper, we propose a method for normalizing 3D volumetric scans using the intensity profile of the training samples. This aids the CNN by creating a higher contrast around the abnormal region of interest in the scan. We use the CQ500 head CT dataset to demonstrate the validity of our method for detecting different acute brain hemorrhages such as subarachnoid hemorrhage (SAH), intraparenchymal hemorrhage (IPH), subdural hematoma (SDH), and intraventricular hemorrhage (IVH). We compare the proposed method with a baseline, two variants of the 3D VGGNet architectures, Resnet, and show that the proposed method achieves significant improvement in classification performance. For binary classification, we achieved the best F1 score of 0.96 (normal vs SAH), 0.93 (normal vs IPH), 0.98 (normal vs SDH), and 0.99 (normal vs IVH), and for four-class classification, we obtained an average F1 score of 0.77. Finally, we show a limitation of the proposed method while detecting varied abnormalities. The proposed method has applications for abnormality detection for different organs.**



***Index Terms*—3D CNN, medical image sensors, deconvolution visualization, computational complexity.**

I. Introduction

CONVOLUTIONAL neural networks (CNN) have shown remarkable performance for applications in scene recognition [1], object detection [2], medical image sensors [3], [4], face recognition sensors [5] and wearable sensors [6], [7]. CNNs can automatically extract features from input images using convolution operations without explicit feature handcrafting [8]. Successive convolutional layers in the CNN learn precise features before the spatial scrambling performed by the fully connected layers. The initial convolutional layers learn low-level information. These generally include object edges, boundaries, curves, et cetera. As we move deeper into the CNN, the network learns more abstract features (or higher-level information). The deeper layers in the CNN use the low-level features learned by the initial layers to extract object shapes and specific object parts in the image [9]–[11]. With the advent of CNN models, initial efforts were made to perform classification in the 2D domain, by decomposing the 3D volume into 2D slices in the sagittal, coronal and axial planes [12]–[14]. However, the conversion of 3D volumetric scans to 2D images leads to the loss of information on the spatial context, organ appearance, and texture that are important in medical images. This prompted the adoption of 3D CNN based approaches that were earlier limited by their associated computational costs. The advances in graphical processing hardware capabilities have led to 3D CNN based approaches for segmentation [4], [15], [16] and detection of abnormalities on volumetric scans [17]–[20]. While most of the previous studies analyzing volumetric magnetic resonance imaging (MRI) or CT data have used varied machine learning approaches (including 2D CNNs), multiple recent studies using 3D CNNs have been published. Traditional methods have involved selecting slices manually, manually locating the abnormalities, and extensive feature handcrafting before the samples are fed to the model [21]–[23], which introduces significant cost in terms of computation and manual effort. Traditional methods used to detect hemorrhage from volumetric CT and MRI data used support vector machines [24], Hopfield networks [25], and

Authors acknowledge the support from Lee Kong Chian School of Medicine and Data Science and AI Research (DSAIR) Center of Nanyang Technological University Singapore (Project Number ADH-11/2017-DSAIR). PP and BG also acknowledges the support from the Cognitive Neuro Imaging Centre (CONIC) at Nanyang Technological University Singapore.

Satya P. Singh is with Lee Kong Chian School of Medicine, Nanyang Technological University, Singapore, 636921, Singapore,

Lipo Wang is with School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, 639798, Singapore

Sukrit Gupta is with School of Computer Science and Engineering, Nanyang Technological University, Singapore, 639798, Singapore

*Parasuraman Padmanabhan is with Cognitive Neuroimaging Centre and Lee Kong Chian School of Medicine, Nanyang Technological University, Singapore, 636921, Singapore. corresponding author (email: ppadmanabhan@ntu.edu.sg)

Balázs Gulyás is with Cognitive Neuroimaging Centre and with Lee Kong Chian School of Medicine, Nanyang Technological University, Singapore, 636921 Singapore. Balázs Gulyás also with the Department of Clinical Neuroscience, Karolinska Institute, 17176 Stockholm, Sweden.

logistic classifiers [26]. These were usually applied to manually selected single image slices with known pathology. Therefore, although these classifiers often gave a satisfactory performance, they had little practical application in deployment.

Over the years, CNN variants have been used to detect and classify types of bleeding and CT heads. Standvoss et al. [27] use 3D CNN to detect cerebral microbleed in traumatic brain injury. In this work, the authors achieved 87% accuracy with 8 layers VggNet like CNN architecture. Jnawali et al. [20] use 3D variants of GoogleNet and VggNet for the classification of CT brain hemorrhage. The study was carried out using a large 40,000 3D CT heads. By ensemble of three different 3D CNN architectures, the authors show the area under the curve (AUC) of 0.87. In [28], a masked R-CNN architecture with 3D-contracting and 2D expanding fully convolutional feature pyramid network was used to evaluate brain CT heads. Patel et al. [29] combined CNN and long short-term memory (LSTM) for intracranial hemorrhage (ICH) detection and obtained an AUC of 0.96 for the binary classification task. Cho et al. [30] constructed a cascaded CNN network. One network was used to detect if there is any bleeding and if there is any bleeding then the second network was used to the classification of hemorrhage for their types. Sato et al. [31] use a 3D autoencoder for detecting anomalies in CT heads in an unsupervised manner. 11 and 6 layers were used for convolutional and deconvolutional blocks respectively. 3D patches were extracted from the CT data for training and testing the network. Ker et al. [17] present a 3 layer 3D CNN network for brain hemorrhage detection. Authors use a thresholding technique for creating sharp edges and curves around the region of interest (ROI). They use a single thresholding point for all the images present in the dataset. Since there are large variations in the intensity profile of CT scans, using a single threshold point for all CT scans can result in the loss of important information. Our current work is the extension of this work. We select different thresholding points for each CT scans. The detail about the procedure will be discussed in the materials and methods section.

Recently CNN based architectures have been proposed for detecting abnormalities in medical images derived from multiple imaging sensors [4], [12], [13], [18]–[20], [25], [32]–[34]. Computed tomography (CT) is one of these medical imaging modalities to be the most commonly used device in emergencies for patients with a traumatic head injury, stroke, or increased intracranial pressure. This is due to their affordability which makes them widely available and their low acquisition time that is crucial for immediate diagnosis and neurosurgical intervention. Since timely diagnosis is detrimental to patient treatment outcomes, it becomes beneficial to deploy deep learning solutions for detecting and locating anomalous ROI in these settings. However, unlike the usual images that CNNs are applied on, the detection of anomalous ROI in head CT scans is not simply based on sharp edges or curves but texture and small intensity variations [35]. In fact, in most cases, the anomalous ROI and the background are almost indistinguishable because of the small variation in their intensities [36], [37]. Fig. 1 shows axial views of CT scan examples used in the experiments. The uniform white outer rim is the bone of the skull surrounded by dark grey tissue. The hemorrhage is depicted with grey patches

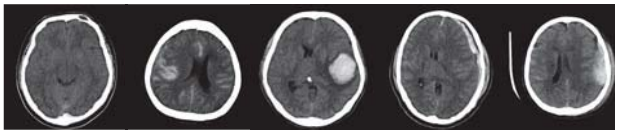


Fig. 1. Axial slice view of the exemplary CT scan used in experiments. From left, Normal, subarachnoid hemorrhage (SAH), intraparenchymal hemorrhage (IPH), subdural hematoma (SDH), and intraventricular hemorrhage (IVH).

that are irregular in shape and of varying sizes. This contrasts with the wide range of pixel intensities in even the head CT of a normal subject, whereby the skull bone represents the highest and the ventricles represent the lowest intensities, respectively. As a result, the application of successive convolutional layers leads to the dispersion of the small intensity variations around the ROI in the image and CNN starts focusing on the edges and the curves instead. Therefore, the CNN extracts features from areas where there are sharp intensity variations such as occipital bone, fornix, ventricle, et cetera that do not help in detecting the anomalous ROI. While this behavior is desirable in cat-dog images, but the same behavior is detrimental for head CT scans, since the anomalous ROI start disappearing during the convolution process [17].

In this paper, we develop a method to create sharp edges and curves around the anomalous ROI to facilitate the extraction of features by CNN. Using this method, we classify the head CT scans in their appropriate classes such as Subarachnoid hemorrhage (SAH), Intraparenchymal hemorrhage (IPH), subdural hematoma (SDH), and intraventricular (IVH) hemorrhage. We made the following contributions in this paper:

1) We propose a feature enhancing method to create edges, curves, and shapes around abnormalities in CT head. We then present a shallow 3D CNN for the classification of Brain Haemorrhages.

2) Using a backtracking algorithm to visualize feature maps, we optimize the architecture of 3D VGGNet for the classification of head CT images using visual information through filters.

3) We compared the proposed shallow 3D CNN with traditional optimized 3D VGGNet and 3D ResNet for the classification of Brain Haemorrhages. The proposed shallow 3D CNN requires fewer training epochs, fewer trainable parameters, and higher accuracy compared to 3D VGGNet and 3D ResNet.

The rest of the paper is organized as follows: In Section II, we discuss the data collection and pre-processing, the proposed framework with a mathematical background, parameter settings, and the filter visualization method used in experiments. Experimental results, discussion, and comparative analysis are given in Section III. Finally, the paper concludes in section IV.

II. MATERIALS AND METHODS

A. Data Collection and preprocessing

The dataset contains non-contrast CT brain images, downloaded from the CQ500 dataset [18]. The CQ500 dataset

is a publicly available dataset collected from various imaging centers in India. There was no overlap between these centers, therefore, there was no chance for a duplicate dataset. We have taken all care to avoid any multiple visits of the same patient in the training, validation, and test dataset. Each scan carries a clinical report (provided by three radiologists) with them and we use this report as the gold standard. The dataset contains 491 patient CT brain scans representing approximately 193,317 slices. Each scan consists of 16–128 slices, depending on the CT scanner. Each scan was anonymized and manually checked by three radiologists to ensure ground truth. We downloaded 222 normal, 134 IPH, 28 IVH, 53 SDH, and 60 SAH. The decision to label the abnormal samples was taken on a majority basis i.e. if two radiologists give the same label then the abnormal sample with the respective label was included. However, for a sample to be included as normal, all the radiologists had to label it as normal. These scans have a different number of image slices and slice thicknesses due to the variability in CT scan models and sensors scanning protocols. The dataset is in DICOM format and contains a lot of metadata. Since scans are usually collected from different scanners, these scans differ in terms of pixel size/thickness (i.e. slice spacing difference between different CT scans collected from different scanners). We use isomorphic resampling to make the slice spacing uniform. The pixel intensity outside the House field Unit (HU) bounds are set to be equal to zero. We use OpenCV [38] for image resizing in the XY direction using linear interpolation. Before feeding the scan to the 3D CNN, we make 28 slices per CT scan for the entire dataset. For resampling in the Z direction (to make 28 slices in each scan), we use a Scipy interpolation function with order 2 and 'nearest' mode. We use a nested cross-validation scheme with five-fold cross-validation. In each fold, the entire dataset is divided into training, validation, and test sets in the ratio of 70:15:15. Since the dataset is highly imbalanced, we balanced the dataset by augmentation. After splitting the data, the entire dataset is augmented by flipping the data around the vertical axis and rotating through angles $\pm 10, \pm 20, \pm 30$, and ± 40 degrees. We took care that the augmented samples from the same subject were either in the train, validation or the test set and that there was no overlap.

B. Intensity Normalization and the Shallow 3D CNN

We hypothesized that by creating sharp edges, curves, and shapes around the anomalous ROI, the 3D CNN can start extracting features related to ROI instead of other regions of sharp intensity variations. Therefore, we propose a method to create some sharp edges and shape around the anomalous region of CT scans before feeding to the 3D CNN.

Assuming that we plot the intensity profile for normal and SAH scans [39] in Fig. 2. Given a slice, x, in the SAH scan, the pixel $x(l,m)$ can have L number of the grey levels such as $[0, 1, 2, \ldots . L-1]$ and the SAH image contains total N number of pixels such as each j^{th} grey level contains $n(j)$ number of pixels. Let's assume that the intensity profile of the SAH slice is divided into only two classes C_0 and C_1. We decide these classes such that class C_1 contains the grey levels related to abnormality and all other higher intensity grey levels while C_0 contains all those grey levels which are also common in the

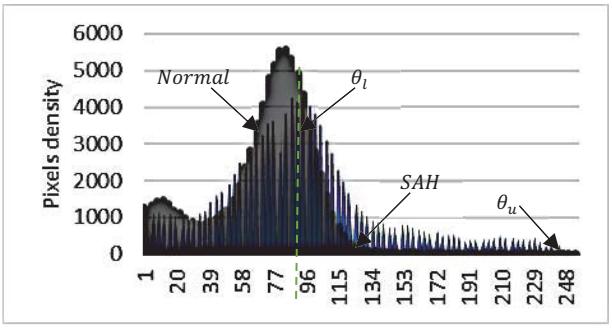


Fig. 2 Histogram plot of normal and abnormal CT scans. The intensity profile is plotted by gray and blue shade for normal and abnormal CT scan respectively.

normal images. Let the allocated grey levels for these two classes be $C_0 = [0, 1, 2, \ldots . k-1]$ and $C_1 = [k, k+1, , \ldots . L-1]$. Now the corresponding probabilities of these classes separated by a parameter θ_l will be given by: $p_0(k) = \sum_{j=0}^{k-1} p(j)$; $p_1(k) = \sum_{j=k}^{L-1} p(j)$. Here, $\mathrm{p}(j) = n(j)/N$ is the probability of the appearance of grey level j in the image. The envelope of normal image histogram is almost equal or greater than the envelope of SAH histogram till a point θ_l. After this point, both histograms start crossing each other. The class C_0 discussed above, we assume that it belongs to all grey levels below this θ_l point. Now if we observe the histograms of both normal and SAH images, we can observe that the class C_0 is a subset of the histogram of normal images till some value of θ_l. That means there is a strong relationship between the lower bounds of the normal image and the SAH image. We start increasing the lower bound in both normal and SAH images and set a point t where these bounds start crossing each other as shown in Fig. 2. We repeat this process for all the normal and SAH images and select a minimum value of θ_l for all images. To get rid of unnecessary edges and information, we set another point called θ_u for upper level. Finally, we processed both normal and SAH images as follows:

$$x(l,m) = \begin{cases} constant & if\ x(l,m) < \theta_l\ ) \\ x(l,m) & if\ \theta_l \le x(l,m) \le \theta_u \\ constant & if\ x(l,m) > \theta_u) \end{cases} \quad (1)$$

We use a 3D convolutional visualization technique for searching optimal CNN architecture [40]. After an exhaustive search for the number of convolutional layers, number of filters in each layer (Detailed procedure will be discussed in the next section), we define a 3D CNN architecture as shown in Fig. 3 and Table I. The definition of the 3D CNN architecture and procedure is summarized as follows:

Step 1) We manually search θ_l and θ_u for entire train data and we use these values as a threshold and set all pixel intensities below θ_l and above θ_u equal to some constant value.

Step 2) In Fig. 3, we define two 3D CNN model such as 1) We called baseline 3D CNN (B3DCNN) if the input data is without threshold (model within a red rectangular box in Fig. 3), and 2) We called shallow 3D CNN (S3DCNN) if the input data is with threshold by θ_l and θ_u (entire model within a blue rectangular box in Fig. 3.

Step 3) We also use θ_l and θ_u to train a YOLO like network

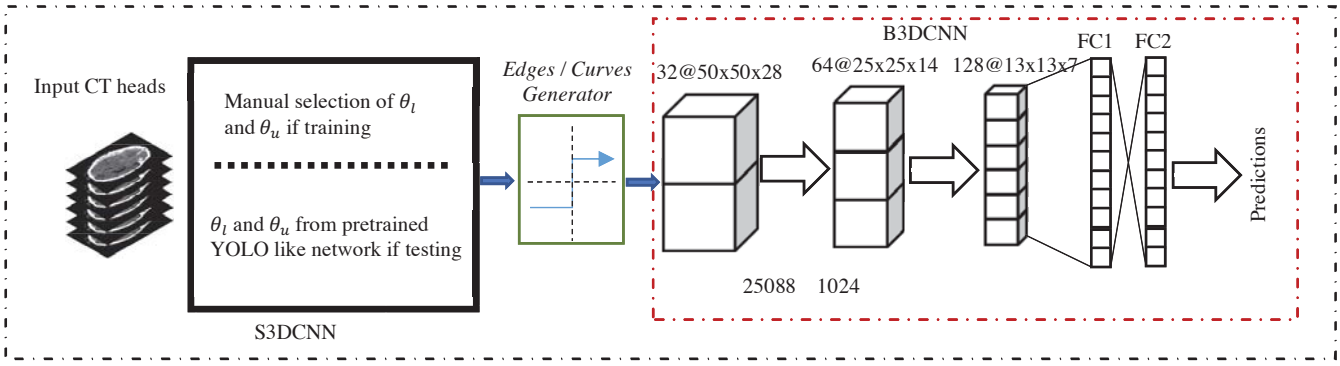


Fig. 3. Proposed 3D CNN architecture for hemorrhage classification. The network inside the red box is baseline 3DCNN (B3DCNN) while the entire framework inside the blue box is shallow 3DCNN (S3DCNN).

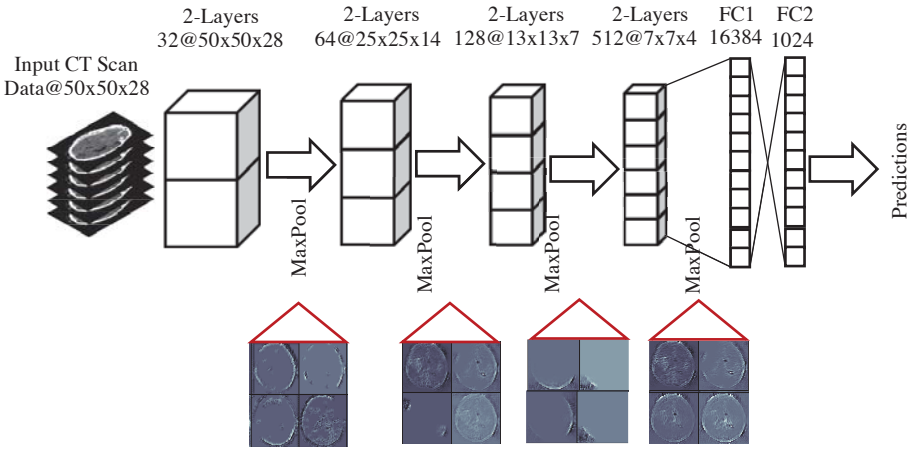


Fig. 4. 3D CNN architecture with filter visualization. The features are visualized using 3D deconvolution visualization methods at each Relu activation. It can be seen that the third stage convolutional layers mostly extract features based on the intensity variations which usually belong to our region of interest.

(same architecture as B3DCNN). We use θ_l and θ_u as targeted output and originally processed (without thresholding) as the input data.

Step 4) For the testing phase, the values of θ_l and θ_u was predicted using a pre-trained YOLO like network from step 3. We use these predicted θ_l and θ_u for intensity normalization of testing data.

C. Optimizing 3D CNN architectures by visualizing layerwise filters

We start with a simple 3 layer CNN network with 32, 64, and 128 filters at each layer followed by pooling layers and two fully connected layers. We look at the visual representation of the filters obtained using deconvolution visualization [40] in different layers of the VGGNet architecture (Fig. 4). As expected, the filters in the first convolutional layer extract edges from the scan, the second layer extracts some curves and the third layer extracts some background information and edges. Keeping the VGGNet architecture in mind, we deepen the network by adding more layers to the architecture. We start increasing the number of layers from 2 layers with 32 filters each (with pooling) by adding 2 layers of 64 filters each with pooling and 2 layers of 128 filters with pooling and 2 layers of 256 filters with pooling. By visual inspection of the last convolutional layer (with 256 filters), we observed that the filters can capture the features of the ROI. Therefore, we add more filters to this layer and increased the number of filters from 256 to 512. Upon adding further layers, we see that the ROI was not visible. We, therefore, obtained the two variants of 3D VGGNet CNN (VggNet1 and VGGNet2 architecture shown in Fig. 4 and Table I. The depth of VggNet1 and VggNet2 is same. The only difference is the number of filters in layer 1,2, and 3 (As shown in Table I). We also prepared a 3D version of 18 layer ResNet (ResNet18) as shown in Table I. We use the 18 layer architecture from [41] with slight modification in kernel size at the first layer.

D. Parameter Settings

It has been observed that the performance of 3D CNN highly depends on the weight initialization methods, kernel size, pooling layer type, learning rate optimization, etc. The weights and biases for each layer are initialized using the normal distribution with mean 0.0 and standard deviation 1.0. In 3D-CNN, the Maxpooling layer is added after each individual (for B3DCNN and S3DCNN) or after two convolution layers (two 3D variants of VGGNet). Max-pooling reduces the dimensionality of the image by reducing the number of pixels in the output of the previous convolution layer. We use the max-pooling layer with a 2x2 filter, and strides of 2 with no padding. Details are given in Table I. The loss function attempts to minimize by continuously updating the weights and the model during the training. We use SoftMax cross-entropy for calculating loss function and then the loss was optimized using Adam optimizer for VggNet1, VggNet2, B3DCNN, S3DCNN. We use learning rate 0.001, 0.002, 0.0001, and 0.0001 for training Resnet18, we use stochastic gradient descent (SGD)

Table I Network architectures details used in this study.

3D VggNet1				3D VggNet2				B3DCNN/S3DCNN (Fig. 2)				Resnet18	
Layer	Filters	Kernel	Strides	Layer	Filters	Kernel	Strides	Layer	Filters	Kernel	Strides	Layer	Architecture
Conv	64	3	1	Conv	32	3	1	Conv	32	3	1	conv1	3×3 $\times 3,64, strides\ 1$
Conv	64	3	1	Conv	32	3	1	Pool		2	2	Pool1	$3 \times 3 \times 3$ max pool
Pool		2	2	Pool		2	2	Conv	64	3	1	Conv2_x	$\begin{Bmatrix} 3 \times 3 \times 3,64 \\ 3 \times 3 \times 3,64 \end{Bmatrix} \times 2$
Conv	128	3	1	Conv	64	3	1	Pool		2	2		
Conv	128	3	1	Conv	64	3	1	Conv	128	3	1	Conv3_x	$\begin{Bmatrix} 3 \times 3 \times 3,128 \\ 3 \times 3 \times 3,128 \end{Bmatrix} \times 2$
Pool		2	2	Pool		2	2	Pool		2	2		
Conv	256	3	1	Conv	128	3	1	FC1	1024			Conv4_x	$\begin{Bmatrix} 3 \times 3 \times 3,256 \\ 3 \times 3 \times 3,256 \end{Bmatrix} \times 2$
Conv	256	3	1	Conv	128	3	1	Fc2	2				
Pool		2	2	Pool		2	2					Conv5_x	$\begin{Bmatrix} 3 \times 3 \times 3,512 \\ 3 \times 3 \times 3,512 \end{Bmatrix} \times 2$
Conv	512	3	1	Conv	512	3	1						
Conv	512	3	1	Conv	512	3	1					Pool	Average Pooling
Pool		2	2	Pool		2	2					FC1	1024
FC1	1024			FC1	1024							FC2	2
FC2	2			Fc2	2								

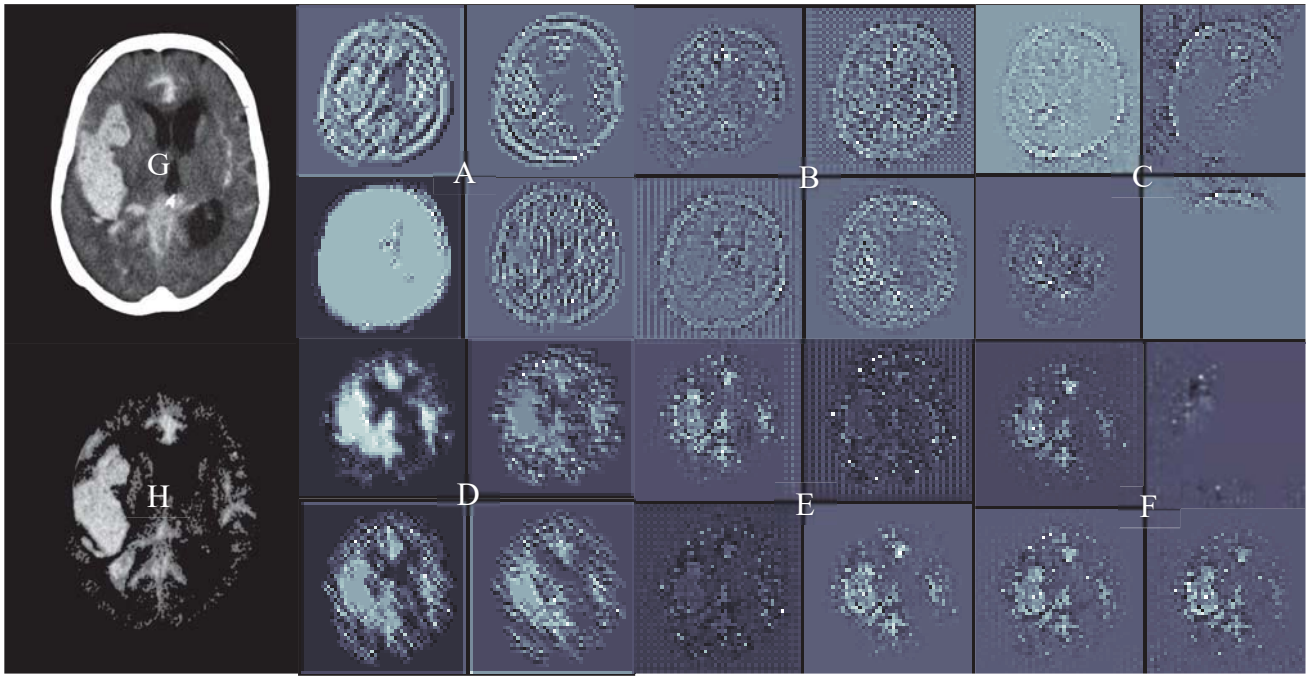


Fig. 5. A, B, C are the visual samples of Relu1, Relu2, and Relu3 of B3DCNN. Similarly, D, E, and F are the visual samples of Relu1, Relu2, and Relu3 of S3DCNN as described in Fig. 3. G is the middle slice of CT scan example (SAH in this case) without thresholding as input to B3DCNN and H is the middle slice of CT scan example with thresholding as input to S3DCNN used in visualization.

and we set the learning rate 0.001.

E. Metrics

Due to an imbalance in the number of normal and abnormal data samples, we use both accuracy and F1 score to evaluate and test the models [42]. We also use three other indices such as sensitivity, specificity, and recall given by [43]:

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn} \quad (2)$$

$$sensitivity = \frac{tp}{tp + fn} \quad (3)$$

$$precision = \frac{tp}{tp + fp} \quad (4)$$

$$F1\ score = (2 * senstivity * precision)/(senstivity + precision) \quad (5)$$

Where, tp, tn, fp and fn represent the *true positive*, *true negative*, *false positive*, and *false negative* respectively.

III. Results And Discussion

In this section, we present experimental results for the validation of the proposed approach. The proposed S3DCNN is validated with binary classification and four class classification. We also tested S3DCNN for normal vs all abnormal. For each experiment, we compare the results with the baseline method B3DCNN. For better understanding and comparative analysis, we have also designed two 3D variants of VGGNet.

In the first set of experiments, we test and validate the proposed S3DCNN for binary class classifications i.e. normal vs SAH, IPH, SDH, and IVH. The results for both S3DCNN and B3DCNN are shown in Table II. We conduct all the experiments on 50x50x28 image sizes. For the proposed S3DCNN, we achieve the best F1 score of 0.96 (p-value = 10^{-2}), 0.93 (p-value = 10^{-4}), 0.98(p-value = 10^{-4}), and 0.99(p-value 10^{-3}) for Normal vs SAH, IPH, SDH, and IVH respectively. In all cases, the proposed S3DCNN outperforms the B3DCNN. Let's take a closer look at Fig. 5 to understand the reason behind the high performance of the proposed architecture. In Fig. 5, for a middle slice 'G' of the sample CT scan. 'A', 'B', and 'C', are the samples of the visualized filter (ReLU activation) for layer 1, layer 2, and layer 3 respectively of the B3DCNN. 'D', 'E', and 'F', are the samples of the visualized filter (ReLU activation) for layer 1, layer 2, and layer 3 respectively of the S3DCNN. We use 3D deconvolution method for visualization of the different filters in the S3DCNN and B3DCNN. Using the proposed intensity normalization approach as discussed in Section II.B, S3DCNN generates edges, curves, and shapes around our ROI. We can observe that S3DCNN starts extracting features that are mostly associated with our ROI. On the other hand, in the baseline architecture, B3DCNN extracts feature from areas where there are sharp intensity variations such as occipital bone, fornix, ventricle, et cetera that do not help in detecting the ROI. Therefore, the proposed method S3DCNN facilitates the convolutional filters in extracting features from anomalous ROI. Since the proposed approach avoids extracting features from non-anomalous regions, it takes fewer epochs compared to the baseline approach. Therefore, the proposed 3D CNN not only outperforms the baseline in terms of classification but also converges in fewer epochs.

Since direct performance comparison with the related work is not possible due to differences in datasets, pre-processing, and augmentation techniques, we compared our proposed approach with two variants of 3D VGGNet and 18 layer ResNet (Resnet18) using our processed data. The architectural details of the proposed 3D variants of VGGNet (3D VGGNet1 and 3D VGGNet2) and Resnet18 are as shown in Table I. We didn't add more convolutional layers to the VGGNet variants, because further addition of layers in the 3D VGGNet led to a decline in the classification performance. We varied the number of filters in each layer and obtained the final architecture of 3D VGGNet by using accuracy and visual inspection of convolutional filters as discussed in Section II. We trained the 3D VGGNet1, 3D VGGNet2, and Resnet18 for normal vs SAH, normal vs ICH, normal vs SDH, and normal vs IVH. The results are shown in Table II. For SAH and IVH cases, the proposed 3D VGGNet2 outperforms the 3D VGGNet1. For IPH, 3D VGGNet1 shows very low sensitivity but high specificity in comparison to 3D VGGNet2 and also a low F1 score. For SDH, 3D VGGNet2 gives a lower F1 score (by 17%,) and other indices (e.g. accuracy by 8%). The depth of both VGGNet and proposed 3D VGGNet2 is almost half. If we compare the results of 3D VGGNet1 and VGGNet2 with our proposed approach (Table I), we conclude that the proposed shallow 3D CNN giving a much higher performance in all cases. In terms of AUC, Rsnet18 performs better than B3DCNN, VggNet1, and VggNet2 for all cases. We also plotted the ROC curve for binary class classification for all four approaches i.e. B3DCNN, S3DCNN, VGGNet1, and VGGNet2 as shown in Fig. 6.

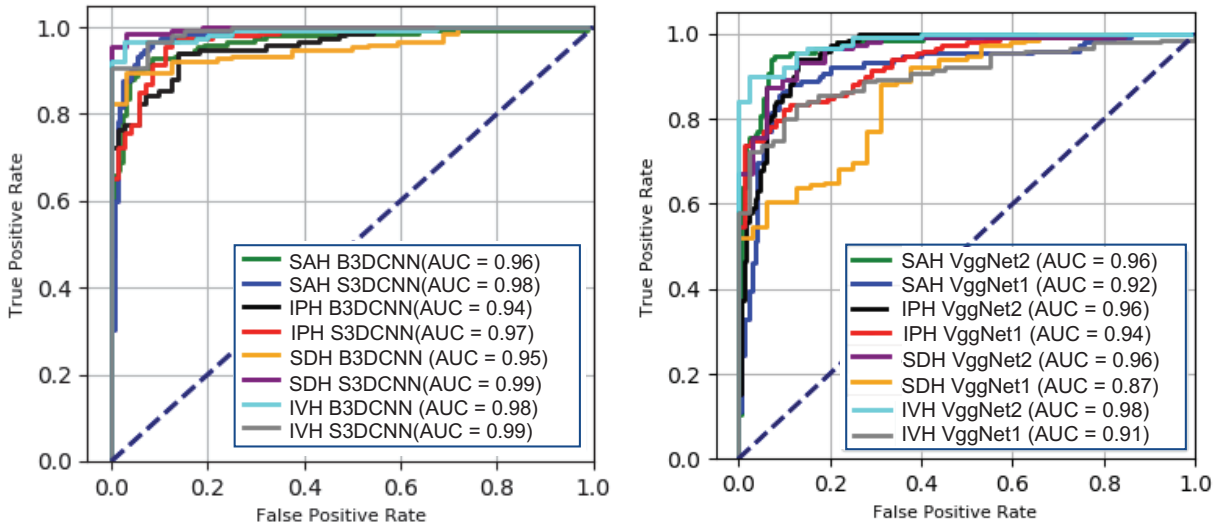
We also test the performance of S3DCNN for the multi-class environment. We experimented for five class classifications i.e.

Table II Binary classification results for 1) baseline method B3DCNN, proposed method S3DCN 2), 3D VGGNet, 3D VGGNet2 and Resnet18. The third decimal point is rounded up that (1.00 is rounded value).

Normal vs	Methods	Sensitivity	Specificity	Precision	F1 Score	Acc	AUC
SAH	B3DCNN	0.91	0.93	0.94	0.92	0.92	0.96
	S3DCNN	0.95	1.00	1.00	**0.96**	**0.96**	**0.98**
	VggNet1	0.93	0.94	0.95	0.94	0.93	0.92
	VggNet2	0.93	0.92	0.93	0.94	0.94	0.96
	Resnet18	0.93	0.97	0.98	0.95	0.95	0.97
IPH	B3DCNN	0.82	0.95	0.88	0.85	0.90	0.94
	S3DCNN	0.97	0.94	0.89	**0.93**	**0.95**	**0.97**
	VggNet1	0.42	0.99	0.94	0.58	0.80	0.94
	VggNet2	0.89	0.95	0.90	0.90	0.93	0.96
	Resnet18	0.86	0.98	0.95	0.91	0.95	0.96
SDH	B3DCNN	0.77	0.94	0.68	0.72	0.91	0.95
	S3DCNN	0.98	1.00	0.98	**0.98**	**1.00**	**0.99**
	VggNet1	0.97	0.93	0.74	0.84	0.94	0.87
	VggNet2	0.91	0.93	0.73	0.81	0.92	0.96
	Resnet18	0.94	0.99	0.93	0.93	0.98	0.98
IVH	B3DCNN	0.97	0.87	0.96	0.96	0.95	0.98
	S3DCNN	1.00	0.85	0.97	**0.99**	**0.97**	**0.99**
	VggNet1	0.68	0.93	0.73	0.70	0.88	0.91
	VggNet2	0.89	0.96	0.79	0.84	0.95	0.98
	Resnet18	0.96	0.83	0.95	0.96	0.94	0.97

Table III Five class classification results for baseline method B3DCNN and proposed method S3DCN

	Methods	Sensitivity	Specificity	Precision	F1 Score	Acc
Overall	B3DCNN	0.75	0.90	0.75	0.75	0.75
	S3DCNN	0.76	0.90	0.78	0.77	0.77
SAH	B3DCNN	0.73	0.81	0.71	0.72	0.78
	S3DCNN	0.74	0.82	0.72	0.73	0.79
IPH	B3DCNN	0.42	0.95	0.68	0.52	0.86
	S3DCNN	0.53	0.98	0.81	0.64	0.90
SDH	B3DCNN	0.97	0.98	0.81	0.88	0.98
	S3DCNN	0.97	0.99	0.86	0.92	0.99
IVH	B3DCNN	67.5	93.4	73.0	70.1	88.0
	S3DCNN	88.5	96.1	79.3	83.6	95.0



(a) (b)

Fig. 6. ROC curve for binary class classifications (a) S3DCNN and S3DCNN. (b) 3D VggNet 1 and VggNet2.

Table IV Normal vs All Abnormal (SAH+IPH+SDH+IVH)

Methods	Sensitivity	Specificity	Precision	F1 Score	Acc
B3DCNN	0.97	0.94	0.97	0.97	0.96
S3DCNN	0.98	0.94	0.97	0.98	0.97
VggNet1	0.90	0.89	0.82	0.85	0.89
VggNet2	0.90	0.93	0.93	0.92	0.91
Resnet18	0.97	0.95	0.97	0.96	0.97

normal vs. SAH vs. IPH vs. SDH. The results are shown in Table III. In this experiment, the proposed approach is showing better performance for all cases compared to B3DCNN. The overall 0.77 (p-value = 10^{-2}) F1 score is achieved for both using B3DCNN and S3DCNN. Both networks show poor results for individual IPH.

In the final experiment, we combine all abnormal scans (i.e. SAH+SDH+ICH+IVH) as abnormal scans and perform experiments for normal vs abnormal classification. The results are shown in Table IV. Both B3DCNN and S3DCNN give promising results with accuracies around 0.96 (p-value = 10^{-4}) and 0.97 (p-value = 10^{-5}), whereas the accuracies for 3D VGGNet2 and 3D VGGNet1 are close to 0.91 (p-value = 10^{-3}). In this case, Resnet18 was showing 0.97 acc. (p-value = 10^{-5}). A similar trend is seen for other performance indices whereby both the B3DCNN and S3DCNN give similar performances that are much higher than their VGGNet counterparts. One interesting result is the similar performance of S3DCNN with the baseline B3DCNN which shows that the proposed technique does not improve the classification performance significantly when heterogeneous abnormalities are considered. We attribute this to the different intensities and volumes of the ROI in different types of brain hemorrhages that leads to a high variation in the values of θ_l and θ_u across the scans. Thus, the image normalization in the case of S3DCNN does not work as

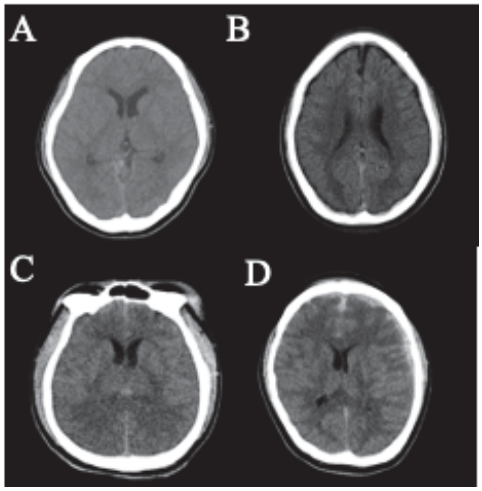


Fig. 7. Some erroneous predictions from the proposed approach. "A" was SAH and was predicted as Normal, "B" was Normal and was predicted as SAH, "C" was SDH but was predicted as Normal, and "D" was IVH and was predicted as IPH.

expected and there is no gain in classification performance. This is a limitation of the proposed method, whereby it needs to be trained on scans where the type of abnormality is similar across scans.

A limitation of the present work is the small and imbalanced dataset that can cause overfitting in the model. This is a common problem with medical imaging datasets, and we prevented overfitting by introducing dropouts in the convolutional and early stopping during model training. We also use data augmentation techniques and the obtained performances are close to much bigger datasets. A second limitation of our work is that useful information may also be lost during the threshold if the threshold points are not selected properly. Therefore, great care needs to be taken when choosing the threshing point. Furthermore, for a large dataset, manual

selection of threshold points can be a tedious and time-consuming process. Fig. 7 is showing some examples of failures using our proposed approach. In Fig. 7, "A" was SAH and was predicted as Normal, "B" was Normal and was predicted as SAH, "C" was SDH but was predicted as Normal, and "D" was IVH and was predicted as IPH. While our dataset of around 490 scans is comparable to some previous studies [17], [19], [44], it is much smaller than the ones used in [18], [20]. Jnawali et al. [20] used ensemble learning to detect brain hemorrhage by training three 3D CNNs on more than 40,000 head CT scans. To reduce the class imbalance, the author used rotation, flipping and mirroring, etc. For binary classification, the author achieved an overall 0.78 F1 score. Chilamkurthy et al. [18] obtained more than 313,000 head CT scans for training and validating multiple deep learning algorithms for the detection of brain hemorrhages, skull fractures, midline shifts, and mass effects. On CQ 500 dataset, their algorithm achieved AUC of 0.942, 0.931, 0.973, and 0.957 for ICH, IVH, SDH, and SAH.

IV. Conclusion

In this paper, we presented a shallow 3D CNN for the classification of abnormal medical images. The proposed approach is validated by classifying head CT scans of normal and patients suffering from different types of brain hemorrhage obtained from the CQ500 dataset. We use the proposed shallow B3DCNN network with our intensity normalization technique and variants of the deep 3D VGGNet. We show that the network with the proposed normalizing method S3DCNN outperforms the baseline shallow B3DCNN, and both the B3DCNN and the S3DCNN outperform the deeper 3D VGGNet variants. Our experiments validate our hypothesis that successive convolutional layers lead to the dispersion of the small intensity variations caused by the abnormalities in medical images and thus shallow 3D CNNs can outperform their deeper variants for medical image classification tasks. In the future, we will extend the current work for a deeper model and high-resolution images. Our proposed method fails for very minor hemorrhages.

Acknowledgment

Authors acknowledge the support from the Cognitive Neuroimaging Centre – Nanyang Technological University, Singapore and Lee Kong Chian School of Medicine and Data Science and AI Research (DSAIR) center of Nanyang Technological University, Singapore (Project Number ADH-11/2017-DSAIR). PP and BG also acknowledges the support from the Cognitive Neuro Imaging Centre (CONIC) at Nanyang Technological University Singapore.

References

[1] N. Qin, X. Hu, H. Dai, and X. Hu, "Deep fusion of multi-view and multimodal representation of ALS point cloud for 3D terrain scene recognition," *ISPRS J. Photogramm. Remote Sens.*, vol. 143, pp. 205–212, Sep. 2018, doi: 10.1016/j.isprsjprs.2018.03.011.

[2] J. C. Rangel, J. Martínez-Gómez, C. Romero-González, I. García-Varea, and M. Cazorla, "Semi-supervised 3D object recognition through CNN labeling," *Appl. Soft Comput. J.*, vol. 65, pp. 603–613, Apr. 2018, doi: 10.1016/j.asoc.2018.02.005.

[3] A. M. DSouza, L. Chen, Y. Wu, A. Z. Abidin, C. Xu, and A. Wismüller, "MRI tumor segmentation with densely connected 3D CNN," in *Medical Imaging 2018: Image Processing*, 2018, vol. 10574, p. 50, doi: 10.1117/12.2293394.

[4] K. Kamnitsas *et al.*, "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation," *Med. Image Anal.*, vol. 36, pp. 61–78, Feb. 2017, doi: 10.1016/j.media.2016.10.004.

[5] H. Li, J. Sun, Z. Xu, and L. Chen, "Multimodal 2D+3D Facial Expression Recognition with Deep Fusion Convolutional Neural Network," *IEEE Trans. Multimed.*, vol. 19, no. 12, pp. 2816–2831, Dec. 2017, doi: 10.1109/TMM.2017.2713408.

[6] Q. Teng, K. Wang, L. Zhang, and J. He, "The layer-wise training convolutional neural networks using local loss for sensor based human activity recognition," *IEEE Sens. J.*, pp. 1–1, Mar. 2020, doi: 10.1109/jsen.2020.2978772.

[7] S. Ji, W. Xu, M. Yang, and K. Yu, "3D Convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, 2013, doi: 10.1109/TPAMI.2012.59.

[8] J. Gu *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognit.*, vol. 77, pp. 354–377, May 2018, doi: 10.1016/j.patcog.2017.10.013.

[9] X. Fang, "Understanding deep learning via backtracking and deconvolution," *J. Big Data*, vol. 4, no. 1, pp. 1–14, Dec. 2017, doi: 10.1186/s40537-017-0101-8.

[10] M. Menikdiwela, C. Nguyen, H. Li, and M. Shaw, "CNN-based small object detection and visualization with feature activation mapping," in *International Conference Image and Vision Computing New Zealand*, 2018, vol. 2017-Decem, pp. 1–5, doi: 10.1109/IVCNZ.2017.8402455.

[11] C. C. J. Kuo, "Understanding convolutional neural networks with a mathematical model," *J. Vis. Commun. Image Represent.*, vol. 41, pp. 406–413, Nov. 2016, doi: 10.1016/j.jvcir.2016.11.003.

[12] H. C. Shin *et al.*, "Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning," *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1285–1298, May 2016, doi: 10.1109/TMI.2016.2528162.

[13] H. R. Roth *et al.*, "A new 2.5D representation for lymph node detection using random sets of deep convolutional neural network observations," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2014, vol. 8673 LNCS, no. PART 1, pp. 520–527, doi: 10.1007/978-3-319-10404-1_65.

[14] A. Prasoon, K. Petersen, C. Igel, F. Lauze, E. Dam, and M. Nielsen, "Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2013, vol. 8150 LNCS, no. PART 2, pp. 246–253, doi: 10.1007/978-3-642-40763-5_31.

[15] M. Havaei *et al.*, "Brain tumor segmentation with Deep Neural Networks," *Med. Image Anal.*, vol. 35, pp. 18–31, Jan. 2017, doi: 10.1016/j.media.2016.05.004.

[16] P. Moeskops, M. A. Viergever, A. M. Mendrik, L. S. De Vries, M. J. N. L. Benders, and I. Isgum, "Automatic Segmentation of MR Brain Images with a Convolutional Neural Network," *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1252–1261, May 2016, doi: 10.1109/TMI.2016.2548501.

[17] J. Ker, S. P. Singh, Y. Bai, J. Rao, T. Lim, and L. Wang, "Image Thresholding Improves 3-Dimensional Convolutional Neural Network Diagnosis of Different Acute Brain Hemorrhages on Computed Tomography Scans," *Sensors*, vol. 19, no. 9, p. 2167, May 2019, doi: 10.3390/s19092167.

[18] S. Chilamkurthy *et al.*, "Deep learning algorithms for detection of critical findings in head CT scans: a retrospective study," *Lancet*, vol. 392, no. 10162, pp. 2388–2396, Dec. 2018, doi: 10.1016/S0140-6736(18)31645-3.

[19] Q. Dou *et al.*, "Automatic Detection of Cerebral Microbleeds from MR Images via 3D Convolutional Neural Networks," *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1182–1195, May 2016, doi:

10.1109/TMI.2016.2528129.

[20] K. Jnawali, M. R. Arbabshirani, N. Rao, and A. A. Patel, "Deep 3D convolution neural network for CT brain hemorrhage classification," in *Medical Imaging 2018: Computer-Aided Diagnosis*, 2018, vol. 10575, p. 47, doi: 10.1117/12.2293725.

[21] S. R. S. Barnes, E. M. Haacke, M. Ayaz, A. S. Boikov, W. Kirsch, and D. Kido, "Semiautomated detection of cerebral microbleeds in magnetic resonance images," *Magn. Reson. Imaging*, vol. 29, no. 6, pp. 844–852, Jul. 2011, doi: 10.1016/j.mri.2011.02.028.

[22] T. Chan, "Computer aided detection of small acute intracranial hemorrhage on computer tomography of brain," *Comput. Med. Imaging Graph.*, vol. 31, no. 4–5, pp. 285–298, Jun. 2007, doi: 10.1016/j.compmedimag.2007.02.010.

[23] K. N. Keshavamurthy *et al.*, "Machine learning algorithm for automatic detection of CT-identifiable hyperdense lesions associated with traumatic brain injury," in *Medical Imaging 2017: Computer-Aided Diagnosis*, 2017, vol. 10134, p. 101342G, doi: 10.1117/12.2254227.

[24] R. Liu *et al.*, "Hemorrhage slices detection in brain CT images," in *Proceedings - International Conference on Pattern Recognition*, 2008, doi: 10.1109/icpr.2008.4761745.

[25] S. Schreiner, B. M. Dawant, C. B. Paschal, and R. L. Galloway, "The importance of ray pathlengths when measuring objects in maximum intensity projection images," *IEEE Trans. Med. Imaging*, vol. 15, no. 4, pp. 560–567, 1996, doi: 10.1109/42.511759.

[26] M. Al-Ayyoub, I. Aljarrah, D. Alawad, and K. Al-Darabsah, "Automatic Detection and Classification of Brain Hemorrhages Article in WSEAS Transactions on Computers," *WSEAS Trans. Comput.*, vol. 12, no. 10, pp. 395–405, 2013.

[27] K. Standvoss *et al.*, "Cerebral microbleed detection in traumatic brain injury patients using 3D convolutional neural networks," in *Medical Imaging 2018: Computer-Aided Diagnosis*, 2018, vol. 10575, p. 48, doi: 10.1117/12.2294016.

[28] P. D. Chang *et al.*, "Hybrid 3D/2D convolutional neural network for hemorrhage evaluation on head CT," *Am. J. Neuroradiol.*, vol. 39, no. 9, pp. 1609–1616, 2018, doi: 10.3174/ajnr.A5742.

[29] A. Patel, S. C. Van De Leemput, M. Prokop, B. Van Ginneken, and R. Manniesing, "Image Level Training and Prediction: Intracranial Hemorrhage Identification in 3D Non-Contrast CT," *IEEE Access*, vol. 7, pp. 92355–92364, 2019, doi: 10.1109/ACCESS.2019.2927792.

[30] J. Cho *et al.*, "Improving Sensitivity on Identification and Delineation of Intracranial Hemorrhage Lesion Using Cascaded Deep Learning Models," *J. Digit. Imaging*, vol. 32, no. 3, pp. 450–461, Jun. 2019, doi: 10.1007/s10278-018-00172-1.

[31] D. Sato *et al.*, "A primitive study on unsupervised anomaly detection with an autoencoder in emergency head CT volumes," in *Medical Imaging 2018: Computer-Aided Diagnosis*, 2018, vol. 10575, p. 60, doi: 10.1117/12.2292276.

[32] S. P. Singh, A. Lay-Ekuakille, D. Gangwar, M. K. Sharma, and S. Gupta, "Deep ConvLSTM with self-attention for human activity decoding using wearables," May 2020.

[33] S. P. Singh, L. Wang, S. Gupta, H. Goli, P. Padmanabhan, and B. Gulyás, "3D Deep Learning on Medical Images: A Review," Mar. 2020.

[34] H. Nahata and S. P. Singh, "Deep Learning Solutions for Skin Cancer Detection and Diagnosis," Springer, Cham, 2020, pp. 159–182.

[35] S. Prasad and S. Gupta, "Volumetric tumour detection using improved region grow algorithm," *Int. J. Comput. Syst. Eng.*, vol. 4, no. 2/3, p. 127, 2018, doi: 10.1504/ijcsyse.2018.091388.

[36] G. J. E. Rinkel *et al.*, "Nonaneurysmal perimesencephalic subarachnoid hemorrhage: CT and MR patterns that differ from aneurysmal rupture," *Am. J. Roentgenol.*, vol. 157, no. 6, pp. 1325–1330, Sep. 1991.

[37] W. Zhao and L. Wang, "Research on 3D Reconstruction Algorithm of Medical CT Image Based on Parallel Contour," *IEEE Sens. J.*, pp. 1–1, 2019, doi: 10.1109/jsen.2019.2948579.

[38] G. Bradski and A. Kaehler, *Learning openCV: computer vision with the openCV library; electronic version*. 2008.

[39] T. Zhang *et al.*, "A hybrid surrogate and pattern search optimization method and application to microelectronics," *Struct. Multidiscip. Optim.*, vol. 32, no. 4, pp. 327–345, Oct. 2006, doi: 10.1016/j.imavis.2005.11.010.

[40] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2014, vol. 8689 LNCS, no. PART 1, pp. 818–833, doi: 10.1007/978-3-319-10590-1_53.

[41] K. Hara, H. Kataoka, and Y. Satoh, "Learning spatio-Temporal features with 3D residual networks for action recognition," in *Proceedings - 2017 IEEE International Conference on Computer Vision Workshops, ICCVW 2017*, 2017, vol. 2018-Janua, pp. 3154–3160, doi: 10.1109/ICCVW.2017.373.

[42] V. Bhateja, H. Patel, A. Krishn, A. Sahu, and A. Lay-Ekuakille, "Multimodal Medical Image Sensor Fusion Framework Using Cascade of Wavelet and Contourlet Transform Domains," *IEEE Sens. J.*, vol. 15, no. 12, pp. 3783–3790, Dec. 2015, doi: 10.1109/JSEN.2015.2465935.

[43] S. P. Singh, S. Urooj, and A. Lay-Ekuakille, "Breast Cancer Detection Using PCPCET and ADEWNN: A Geometric Invariant Approach to Medical X-Ray Image Sensors," *IEEE Sens. J.*, vol. 16, no. 12, pp. 4847–4855, Jun. 2016, doi: 10.1109/JSEN.2016.2533440.

[44] M. Grewal, M. M. Srivastava, P. Kumar, and S. Varadarajan, "RADnet: Radiologist level accuracy using deep learning for hemorrhage detection in CT scans," in *Proceedings - International Symposium on Biomedical Imaging*, 2018, vol. 2018-April, pp. 281–284, doi: 10.1109/ISBI.2018.8363574.