



# Transfer Learning Using Convolutional Neural Network Architectures for Brain Tumor Classification from MRI Images

Rayene Chelghoum<sup>1</sup> (✉) , Ameer Ikhlef<sup>1</sup> , Amina Hameurlaine<sup>1</sup> ,  
and Sabir Jacquir<sup>2</sup> 

<sup>1</sup> Frères Mentouri University, LARC, Laboratory of Automatic and Robotic,  
Constantine, Algeria

rayene.chelghoum@umc.edu.dz, ameikhlef@yahoo.fr,  
am.hameurlaine@gmail.com

<sup>2</sup> Université Paris-Saclay, CNRS, Institut des Neurosciences Paris Saclay,  
Gif-sur-Yvette, France

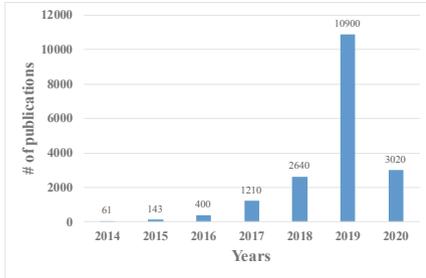
sabir.jacquir@u-psud.fr

**Abstract.** Brain tumor classification is very important in medical applications to develop an effective treatment. In this paper, we use brain contrast-enhanced magnetic resonance images (CE-MRI) benchmark dataset to classify three types of brain tumor (glioma, meningioma and pituitary). Due to the small number of training dataset, our classification systems evaluate deep transfer learning for feature extraction using nine deep pre-trained convolutional Neural Networks (CNNs) architectures. The objective of this study is to increase the classification accuracy, speed the training time and avoid the overfitting. In this work, we trained our architectures involved minimal pre-processing for three different epoch number in order to study its impact on classification performance and consuming time. In addition, the paper benefits acceptable results with small number of epoch in limited time. Our interpretations confirm that transfer learning provides reliable results in the case of small dataset. The proposed system outperforms the state-of-the-art methods and achieve 98.71% classification accuracy.

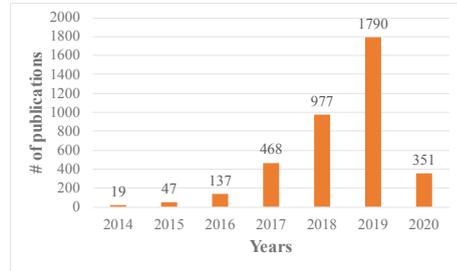
**Keywords:** Convolutional Neural Network · Brain tumor · Classification · Deep learning · Magnetic resonance images · Transfer learning

## 1 Introduction

Brain tumor diagnosis is very important in order to develop an effective plan of treatment. There are more than 120 types of brain and Central Nervous System (CNS) tumors. Neurologists classify manually the brain MR images using the World Health Organization (WHO) classification [1]. The Automation of the classification procedure, in particular brain MR images classification help radiologist in their diagnosis and reduce enormously their interventions.



**Fig. 1.** Number of publications per year in google scholar containing “Convolutional Neural Network” and “medical imaging” keywords from 2014 to 2020. (Queried: February 25<sup>th</sup>, 2020)



**Fig. 2.** Number of publications per year in google scholar containing “Convolutional Neural Network” and “brain tumor” keywords from 2014 to 2020. (Queried: February 25<sup>th</sup>, 2020)

The first automatic classification methods are the machine learning ones. These methods take a long time because they need pre-processing and handcrafted features by experts. The classification accuracy depends on the extracted features which depend on the expert competences. Despite the limitation of machine learning methods, some works [2] achieved between 79% and 85% classification accuracy with their proposed method used tumor extracted features such as shape, rotation invariant texture, intensity characteristics and MR images for brain tumor classification.

To avoid handcrafted features extraction, deep learning (DL) methods involving deep neural networks to classify images in self-learning without the need of handcrafted features extraction are used by Benjio [3] and Litjens *et al.* [4]. Among several DL methods, CNNs are one of the most useful that have been used to solve complex problems in various applications such as detection [5], localization [6], segmentation [7] and classification [8]. They have also yielded good results in medical image application [9–11].

The first real-word application of CNNs was realized by Yann LeCun in 1998 to recognize hand written digits [12]. Since 2010, ImageNet launched an important visual database project called ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [13]. This challenge runs an annual software contest where research teams evaluate their algorithms on the given dataset and achieved higher accuracy. Moreover, CNNs become more useful when Krizhevsky *et al.* (2012) proposed their CNN architecture called AlexNet [14]. The later competition allowed creating and improving real deep CNNs architectures that have achieved higher accuracy on several visual recognition tasks.

In recent years, CNNs have achieved good results in medical image applications due to the growth of available labelled training data, the increase of powerful graphics processing GPU, the rise of accuracy to solve complicated applications over time and the appearance of numerous techniques to learn features. According to the statistics that we made from google scholar, the number of publications using CNNs in the field of medical image applications in general and brain tumor in particular is increase since 2014. This trend can be observed in Fig. 1. and Fig. 2.

Today, several public brain MR images datasets for classification are available for researchers. This help medical scientists to develop more automated classification methods [15]. However, the CNNs training become more complicated and can lead to over-fitting because of the samples size of medical datasets. Also, applying deep pre-trained CNNs based on transfer learning in medical imaging needs to adjust the hyper-parameters and learning parameters of the models in order to achieve a good result. Training the networks with transfer learning is usually much faster and easier than training the networks with randomly initialized weights [16–18]. In [19], authors used a small CNN architecture and achieved 84.19% classification accuracy. A block-wise fine-tuning has been proposed, this one, based on transfer learning, reached a classification accuracy of 94.82% [20]. Other approach proposed by [21] used pre-trained GoogleNet and transfer learning to classify brain MR images and achieved 97.1% of classification accuracy.

In this paper, we present an automatic classification system designed for three types of brain tumor. We use the brain CE-MRI dataset from figshare [22] which consists of three kind of brain tumors (glioma, meningioma and pituitary tumor) in order to classify only abnormal brain MR images. Based on this dataset, we adopted deep transfer learning for feature extraction from from brain MR images using nine deep CNNs architectures: AlexNet [14], GoogleNet [23], VGG16 [24], VGG19 [24], Residual Networks (ResNet18, ResNet50, ResNet101) [25], Residual Networks and Inception-v2 (ResNet-Inception-v2) [26], Squeeze and Excitation Network (SENet) [27]. This system makes easier the interventions of radiologist, helps them to solve brain tumor classification problem and develop an effective treatment.

We report the overall classification accuracy of the nine pre-trained architectures based on training time and epoch number. We explore the impact of epochs number to minimize the consuming time. We classify the extracted features for three different epochs. We achieve good results compared to related works. Also, with smaller number of epochs, we achieve acceptable results in short time.

The paper is structured as follows. The proposed method and different pre-trained CNNs architectures are given in Sect. 2. The experimental setting, the networks preparation and dataset are shown in Sect. 3. The experimental results with a brief discussion are provided in Sect. 3, the conclusion and an outlook for future work are given in Sect. 4.

## 2 Method

In this work, we applied nine pre-trained deep networks including AlexNet, GoogleNet, VGG16, VGG19, ResNet18, ResNet50, ResNet101, ResNet-Inception-v2 and SENet for brain tumor classification problem using transfer learning.

### 2.1 Pre-trained CNNs Architectures for Image Classification

CNNs architectures have been designed to learn spatial hierarchies of features by building multiple blocks: convolution layers with a set of filters, pooling layers, and fully connected layers (FCLs). The real deep architectures created until 2012 through ILSVRC challenge. The classification error of ILSVRC challenge winners is decreased from

15.3% in AlexNet (2012) [14] to 2.251% in SENet (2017) [27]. Also, the number of layers is increased from 8 layers to 152 layers. Table 1 summarizes the differences between those architectures regarding the classification error, the number of layers, the tasks, the execution environment and the training datasets.

**AlexNet.** AlexNet [14] architecture is deeper and much greater than LeNet architecture [28]. It consists of eight layers, five convolutional layers most of them are followed by max pooling and three fully connected layers. The output is the 1000-way softmax that represents the classes. It is trained on two parallel GTX 580 GPU 3 GB which communicate only in certain layers. This scheme reduces the top-5 error rates. AlexNet is improved with ZFNet architecture [29] which visualizes the AlexNet activities within the layers to debug problems and obtain better results. It allows observing the evolution

**Table 1.** Comparison between different CNNs architectures for image analysis.

ILSVRC architectures	Number of layers	Top 5 error rate	Tasks	Training dataset	Execution environment
AlexNet (2012) Ranked 1	8	15.3%	Classification	ImageNet	Two GTX 580 GPUs 3Gg (parallel)
ZFNet (2013) Ranked 1	8	14.8%	Classification	ImageNet Caltech-101 Caltech-256	Single GTX 580 GPU
GoogleNet (2014) Ranked 1	22	6.67%	Classification Detection	ImageNet	CPU
VGGNet (2014) Ranked 2	16–19	6.8%	Classification Localization	ImageNet	Four NVIDIA Titan Black GPU
ResNet (2015) Ranked 1	18–34– 50–101 <b>152</b>	3.57%	Classification Detection Segmentation Localization	ImageNet COCO	Two GPUs
Inception-v4/ResNet-inception (2016) Ranked 1	50–101 <b>152</b>	3.08%	Classification	ImageNet	Twenty replicas with NVIDIA Kepler GPU
SENet (2017) Ranked 1	18–34– 50–101 <b>152</b>	<b>2.251%</b>	Classification Detection	ImageNet –COCO CIFAR-10- CIFAR-100	Eight GPU NVIDIA Titan

of features during training and maps the activities back to the pixel space in intermediate layers.

**GoogleNet.** GoogleNet architecture codenamed Inception-v1 is the improved utilization of computing resources inside the network [23]. The network with the inception architecture is faster than the network with non-inception architecture. The GoogleNet architecture including the inception module uses rectified linear activation function, average pooling layer and not fully connected layer and dropout after removing fully connected layer.

Inception-v1 is improved to Inception-v2 by Ioffe and Szegedy [30] who tried to solve the internal covariate shift. They achieved a top-5 error rate of 4.82%. This result is outperformed to 3.5% by Szegedy *et al.* [31] with their new inception architecture called Inception-v3. Table 2 shows a comparison between the three inceptions.

**Table 2.** Difference between the three inceptions.

Inception-v1	Inception-v2	Inception-v3
Increase the number of units at each stage and shielding the large number of input filters of the last stage to the next layers	Increase the learning rate, remove dropout and local response normalization, shuffle training examples more thoroughly, reduce the L2 weight regularization and the photometric distortions	Trained much faster compared to the other inception and method
Error rate = 6.67%	Error rate = 4.82%	Error rate = 3.5%

**VGGNet.** Karen Simonyan and Andrew Zisserman [24] investigated the effect of the neural convolutional network depth on its accuracy in image recognition. They pushing depth to 11–19 weight layers of the developed VGGNet using very small ( $3 \times 3$ ) convolution filters. The configurations that use 16 and 19 weight layers, called VGG16 and VGG19 perform the best. The classification error decreases with the increased depth and saturated when the depth reached 19 layers. Authors confirm the importance of depth in visual representations.

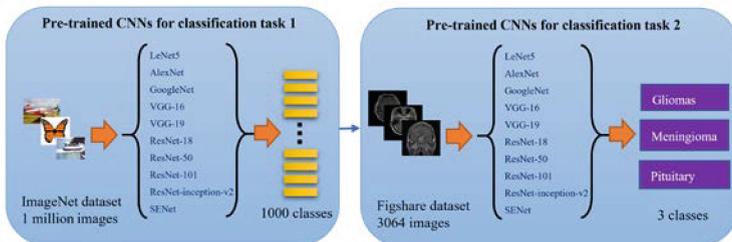
**ResNet/ Inception-v4.** ResNet [25] used residual learning to ease the training of the deeper networks and reduce the errors from increasing depth. This architecture proposed many structures including: 18-layers, 34-layers, 50-layers, 101-layers and 152-layers structure, where the 152-layers structure is better than the other ones. It is less complex and deeper than VGG, and has similar performances to the Inception-v3 network, this is why Szegedy *et al.* [26] combined the inception architecture with residual connections. They evaluated the three ResNet-Inception and the Inception-v4 architectures: The Inception-ResNet-v1 has similar performances to Inception-v3 while the

ResNet-Inception-v2 performs more than ResNet-Inception-v1. The Inception-v4 is simpler and has more inception modules than Inception-v3 but has similar performances to ResNet-Inception-v2.

**SENet.** SENet [27] used Squeeze and Excitation (SE) block which improved the representational power of a network by enabling it to perform dynamic channel-wise feature recalibration. It was applied directly in the Residual Network architecture such as SE- Inception-ResNet-v2, SE-ResNet-101, SE-ResNet-50, SE-ResNet-152 and can be applied to the other existing architectures. It has been performed on ImageNet, COCO, CIFAR-10 and CIFAR-100 datasets across multiple tasks.

## 2.2 Transfer Learning Setting

Transfer learning use the gained knowledge that solve one problem and applied them to solve different related problems by using trained model to learn different set of data. The setting for transfer learning used in this work is explained in the following statements. The pre-trained CNNs architectures: AlexNet, GoogleNet, VGG16, VGG19, ResNet-18, ResNet-50, ResNet-101, ResNet-Inception-v2 and SENet consist of 1000 classes, 1.28 million training images, tested on 100 k test images and evaluated on 50 k validation images. They are challenging the accuracy of human with the best given results. The networks take an image as an input and produce the object label in the image as an output as well as the probabilities of the object categories. In this research, we focus on slice by slice classification of brain tumor using CE-MRI dataset into three types of tumors. First, we modified the last three layers of pre-trained networks in order to adapt them to our classification task. Next, we replaced the fully connected layer in the original pre-trained networks by another fully connected layers, in which the output size represents the three kind of tumor. The transfer learning setting and modification are shown in Fig. 3. Finally, we used transfer learned and fine-tuned deep pre-trained CNNs for experiments using MRI data.



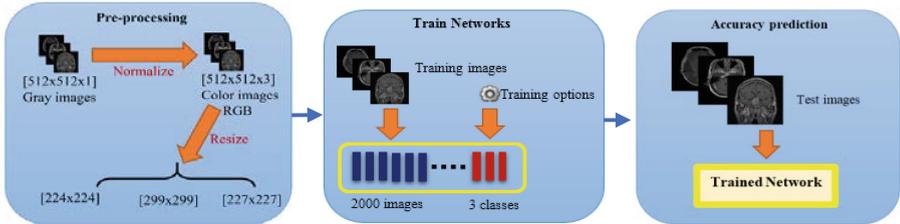
**Fig. 3.** Transfer learning setting and modification

## 3 Experiments and Results

The proposed classification model is implemented in MATLAB 2019b on a computer with the specifications of 16 GB RAM and Intel I9 4.50 GHz CPU.

### 3.1 Experiments

In this section, we describe the dataset used in the experiments, the training parameters and classification accuracy prediction. Figure 4 represents the pre-processing of training dataset and the use of transfer learning networks for brain tumor classification.



**Fig. 4.** Preparation and use the transfer learning network for brain tumor classification

**Dataset and Pre-processing.** The public database used to train and test the networks is available in [22]. It has already used in previous works like in [32, 33]. The dataset is collected from Nanfang Hospital, Guangzhou and General Hospital, Tianjin Medical University, in China during 2005–2010. It contains 3064 abnormal brain CE-MRI from 233 patients with three kinds of brain tumor: meningioma (708 slices), glioma (1426 slices), and pituitary tumor (930 slices). It is based on two dimensional gray images (2D slices). Those data are organized in MATLAB data format (. mat file). The size of images is  $512 \times 512$  pixels and the pixel size is  $49 \text{ mm} \times 49 \text{ mm}$ . In our work, we normalize the gray MRI images in the dataset in intensity values and we convert them into RGB images by corresponding color map to RGB format using Matlab function. We specify the slices as an array of data type where the value 1 corresponds to the first color in the color map. RGB images are returned as an  $m \times n \times 3$  numeric array with values in the range of  $[0, 1]$ . The value 3 corresponds to red, green and blue colors. Then, we resize them according to the used network:  $(227 \times 227)$  in AlexNet and SENet,  $(224 \times 224)$  in GoogleNet, VGGNet and ResNet,  $(299 \times 299)$  in ResNet-Inception-v2 RGB images. The dataset pre-processing is shown on Fig. 4. We divide the data into training and test datasets, where 60% (1836 slices) of the images are used for training and 40% (1228 slices) used for test. The splitting of data into train and test set is performed on a slice basis.

**Training Parameters.** For transfer learning, we train the networks by stochastic gradient descent (SGD) with 0.9 momentum. We use a minibatch size of 128 images and a learning rate of  $10^{-4}$ . To speed up the learning in the new layers, we rise the weight learn rate factor and the bias learn rate factor to 10. Even though, the transferred layers are still slower than the new layers. In order to perform the transfer learning, we train for 25, 50 and 90 epochs where an epoch is a full pass during the dataset training. The networks are validated every 50 iterations during training.

**Classification Accuracy Prediction.** In this part we use the trained networks to classify the test images and calculate the overall classification accuracy. The classification

accuracy is the ability to predict correctly and guess the value of predicted attribute for new data. It is defined as the ratio of sum of true positives (TP) and true negatives (TN) to the total number of trials:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \times 100 \quad (1)$$

Where TP and TN are outcomes produced when the model correctly classifies the positive class and the negative class, respectively. While FP and FN are outcomes produced when the model incorrectly classifies the positive class and the negative class, respectively.

### 3.2 Results

We evaluate the classification performance using the nine pre-trained architectures and summarize our results in the form of tables.

In fact, the purpose of this study is to increase the classification accuracy, speed the training time and avoid the overfitting. This can be assessed through the classification accuracy and the training time of our pre-trained networks. The classification accuracy and the training time using different transfer learning architectures trained for different epochs are respectively shown in Table 3 and Table 4. All our pre-trained networks excepting SENet are reached up to 90% classification accuracy for three different epochs. Despite the use of transfer learning, SENet has an overfitting with epoch equal to 25 and 50, but achieves an acceptable result with epoch equal to 90.

**Table 3.** Classification accuracy using different transfer learning architectures for different epochs.

Architectures	Epoch = 25	Epoch = 50	Epoch = 90
AlexNet	<b>98.14</b>	<b>98.55</b>	<b>98.22</b>
GoogleNet	95.69	97.16	97.24
VGG-16	<b>98.06</b>	<b>98.14</b>	<b>98.71</b>
VGG-19	97.97	<b>98.55</b>	<b>98.47</b>
ResNet-18	96.01	97.86	97.81
ResNet-50	96.67	97.65	96.16
ResNet-101	96.67	96.83	95.99
ResNet-inception-v2	93.67	95.03	95.50
SENet	56.66	56.66	95.18

Another characteristic observed during experiments is the impact of epoch number on the classification accuracy. This effect can be seen in Table 3 and Table 4. The training time is increasing gradually with incremental epochs number, which means that we can consume less time using less epoch. However, the classification accuracy is neither influenced by the epochs number, nor the deep architectures. As shown in Table 3 the

**Table 4.** Training time in minutes using different transfer learning architectures for different epochs.

Architectures	Epoch = 25	Epoch = 50	Epoch = 90
AlexNet	<b>24</b>	<b>48</b>	<b>91</b>
GoogleNet	79	158	281
VGG-16	<b>495</b>	<b>907</b>	<b>1953</b>
VGG-19	532	<b>1174</b>	<b>1979</b>
ResNet-18	71	148	245
ResNet-50	190	374	678
ResNet-101	409	777	1339
ResNet-inception-v2	766	1481	2643
SENet	42	85	160

majority of the pre-trained networks record well with epoch equal to 50 and achieve acceptable results with epoch equal to 25. Even though, the fewer layers of AlexNet, VGG16 and VGG19 perform more than deeper architectures such as ResNet and ResNet-Inception-v2 for the three chosen epochs. They achieved respectively **98.55%**, **98.71%** and **98.55%** classification accuracy. Also, we observed that with an epoch of 50 AlexNet and VGG16 achieved the same accuracy **98.55%** however VGG16 consume a long training time compared to AlexNet.

**Table 5.** Samples images classification prediction using different architectures.

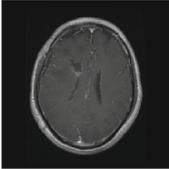
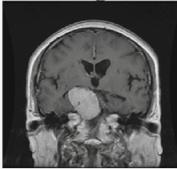
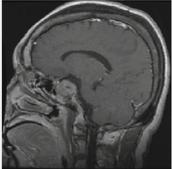
Sample Images	Glioma	Meningioma	Pituitary
			
AlexNet	<b>Glioma 100%</b>	Meningioma 99.9%	<b>Pituitary 100%</b>
GoogleNet	<b>Glioma 100%</b>	Meningioma 97%	Pituitary 99.8%
VGG-16	<b>Glioma 100%</b>	<b>Meningioma 100%</b>	<b>Pituitary 100%</b>
VGG-19	<b>Glioma 100%</b>	Meningioma 99.9%	<b>Pituitary 100%</b>
ResNet-18	<b>Glioma 100%</b>	Meningioma 99.9%	Pituitary 98.3%
ResNet-50	<b>Glioma 100%</b>	Meningioma 98.3%	Pituitary 98.9%
ResNet-101	<b>Glioma 100%</b>	Meningioma 99.2%	Pituitary 99.8%
ResNet-Inception-v2	Glioma 96.9%	Meningioma 93.5%	Meningioma 73.1%
SENet	<b>Glioma 100%</b>	Meningioma 96.1%	Pituitary 91.6%

Table 5 shows a classification of three sample instances. We find that all of the pre-trained architectures pertain to the class glioma and meningioma. All of them, excepting ResNet-Inception-v2 pertain to the class pituitary. This confirms that the deeper architectures do not result good with small datasets.

Table 6 provides a broad comparison based on classification accuracy with the existing methods on the same CE-MRI dataset. Abiwinanda *et al.* [19] achieved 84.19% accuracy with their proposed CNN. Swati *et al.* [20] propose a block-wise fine-tuning method based on transfer learning and achieved 94.82% accuracy. Deepak and Ameer [21] used a pre-trained GoogleNet to extract features from brain MRI images and achieved 97.1% classification accuracy. Our proposed method using the pre-trained VGG16 achieved 98.71% classification accuracy.

**Table 6.** Related works & classification accuracy comparison using the CE-MRI training dataset.

Methods	Abiwinanda <i>et al.</i> (2019) [19]	Swati <i>et al.</i> (2019) [20]	Deepak and Ameer (2019) [21]	Proposed
Training Data	–	25–50–75%	56%	<b>60%</b>
Classification accuracy (%)	84.19	94.82	97.1	<b>98.71</b>

## 4 Conclusion

This paper presents a fully automatic system for three kind of brain tumor classification using CE-MRI dataset from figshare. The proposed system applied the concept of deep transfer learning using nine pre-trained architectures for brain MRI images classification trained for three epochs. Our system outperforms the classification accuracy compared to related works. It shows a good performance with a small number of training samples and small epochs number, which allows to reduce consuming time. The architectures which have fewer layers perform more than the deeper architectures. In the future work, we will apply our system to classify medical images from different modalities such as X-rays, Positron Emission Tomography (PET) and Computed Tomography (CT) for other body organ. Also, we will address the effect of epochs number to the classification performances.

## References

1. Tustison, N.J., et al.: Optimal symmetric multimodal templates and concatenated random forests for supervised brain tumor segmentation (Simplified) with ANTsR. *Neuroinform* **13**(2), 209–225 (2015). <https://doi.org/10.1007/s12021-014-9245-2>
2. Zacharaki, E.I., et al.: Classification of brain tumor type and grade using MRI texture and shape in a machine learning scheme. *Magn. Reson. Med.* **62**(6), 1609–1618 (2009). <https://doi.org/10.1002/mrm.22147>

3. Bengio, Y.: Learning deep architectures for AI. *Found. Trends® Mach. Learn.* **2**(1), 1–127 (2009)
4. Litjens, G., et al.: A survey on deep learning in medical image analysis. *Med. Image Anal.* **42**, 60–88 (2017). <https://doi.org/10.1016/j.media.2017.07.005>
5. Dung, C.V., Anh, L.D.: Autonomous concrete crack detection using deep fully convolutional neural network. *Autom. Constr.* **99**, 52–58 (2019). <https://doi.org/10.1016/j.autcon.2018.11.028>
6. Long, C., Basharat, A., Hoogs, A.: A Coarse-to-fine Deep Convolutional Neural Network Framework for Frame Duplication Detection and Localization in Forged Videos, p. 10 (2018)
7. Nogovitsyn, N., et al.: Testing a deep convolutional neural network for automated hippocampus segmentation in a longitudinal sample of healthy participants. *NeuroImage* **197**, 589–597 (2019). <https://doi.org/10.1016/j.neuroimage.2019.05.017>
8. Merdivan, E., et al.: Image-based Text Classification using 2D Convolutional Neural Networks, p. 6 (2019)
9. Dou, Q., et al.: Automatic detection of cerebral microbleeds From MR images via 3D convolutional Neural networks. *IEEE Trans. Med. Imaging* **35**(5), 1182–1195 (2016). <https://doi.org/10.1109/TMI.2016.2528129>
10. Suk, H.-I., Wee, C.-Y., Lee, S.-W., Shen, D.: State-space model with deep learning for functional dynamics estimation in resting-state fMRI. *NeuroImage* **129**, 292–307 (2016). <https://doi.org/10.1016/j.neuroimage.2016.01.005>
11. Zhang, W., et al.: Deep convolutional neural networks for multi-modality iso-intense infant brain image segmentation. *NeuroImage* **108**, 214–224 (2015). <https://doi.org/10.1016/j.neuroimage.2014.12.061>
12. Lecun, Y.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 47 (1998)
13. Russakovsky, O., et al.: ImageNet Large Scale Visual Recognition Challenge. [arXiv:1409.0575](https://arxiv.org/abs/1409.0575) [cs], September 2014
14. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105. Curran Associates, Inc. (2012)
15. Jui, S., et al.: Brain MRI tumor segmentation with 3D intracranial structure deformation features. *IEEE Intell. Syst.* **31**(2), 66–76 (2016). <https://doi.org/10.1109/MIS.2015.93>
16. Toğaçar, M., Ergen, B., Cömert, Z.: BrainMRNet: brain tumor detection using magnetic resonance images with a novel convolutional neural network model. *Med. Hypotheses* **134**, 109531 (2020). <https://doi.org/10.1016/j.mehy.2019.109531>
17. Sharif, M.I., Li, J.P., Khan, M.A., Saleem, M.A.: Active deep neural network features selection for segmentation and recognition of brain tumors using MRI images. *Pattern Recogn. Lett.* **129**, 181–189 (2020). <https://doi.org/10.1016/j.patrec.2019.11.019>
18. Bernal, J., et al.: Deep convolutional neural networks for brain image analysis on magnetic resonance imaging: a review. *Artif. Intell. Med.* **95**, 64–81 (2019). <https://doi.org/10.1016/j.artmed.2018.08.008>
19. Abiwinanda, N., Hanif, M., Hesaputra, S.T., Handayani, A., Mengko, T.R.: Brain tumor classification using convolutional neural network. *World Congress Med. Phys. Biomed. Eng.* **2019**, 183–189 (2018)
20. Swati, Z.N.K., et al.: Brain tumor classification for MR images using transfer learning and fine-tuning. *Comput. Med. Graph.* **75**, 34–46 (2019). <https://doi.org/10.1016/j.compmedimag.2019.05.001>
21. Deepak, S., Ameer, P.M.: Brain tumor classification using deep CNN features via transfer learning. *Comput. Biol. Med.* **111**, 103345 (2019). <https://doi.org/10.1016/j.compbiomed.2019.103345>

22. Brain tumor dataset. [https://figshare.com/articles/brain\\_tumor\\_dataset/1512427](https://figshare.com/articles/brain_tumor_dataset/1512427). Accessed 17 Feb 2020
23. Szegedy, C., et al.: Going Deeper with Convolutions. [arXiv:1409.4842](https://arxiv.org/abs/1409.4842) [cs], September 2014
24. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition, [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) [cs], September 2014
25. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition, [arXiv:1512.03385](https://arxiv.org/abs/1512.03385) [cs], December 2015
26. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.: Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning, [arXiv:1602.07261](https://arxiv.org/abs/1602.07261) [cs], February 2016
27. Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E.: Squeeze-and-Excitation Networks, [arXiv:1709.01507](https://arxiv.org/abs/1709.01507) [cs], September 2017
28. Handwritten digit recognition with a back-propagation network | Advances in neural information processing systems 2. <https://dl.acm.org/doi/10.5555/109230.109279>. Accessed 17 Feb 2020
29. Zeiler, M.D., Fergus, R.: Visualizing and Understanding Convolutional Networks, [arXiv:1311.2901](https://arxiv.org/abs/1311.2901) [cs], November 2013
30. Ioffe, S., Szegedy, C.: Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, [arXiv:1502.03167](https://arxiv.org/abs/1502.03167) [cs], February 2015
31. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the Inception Architecture for Computer Vision, [arXiv:1512.00567](https://arxiv.org/abs/1512.00567) [cs], December 2015
32. Cheng, J., et al.: Enhanced performance of brain tumor classification via tumor region augmentation and partition. *PLoS ONE* **10**(10), e0140381 (2015). <https://doi.org/10.1371/journal.pone.0140381>
33. Retrieval of Brain Tumors by Adaptive Spatial Pooling and Fisher Vector Representation. <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0157112>. Accessed 17 Feb 2020