



Classification of Brain Tumor by Combination of Pre-Trained VGG16 CNN

Ouiza Nait Belaid*

*Corresponding author, Laboratoire de la Communication dans les Systèmes Informatiques, Ecole Nationale Supérieure d'Informatique, BP 68M, 16309, Oued-Smar, Alger, Algérie. <http://www.esi.dz>. ORCID: 0000-0002-2948-7284. E-mail: o_naitbelaid@esi.dz

Malik Loudini

Laboratoire de la Communication dans les Systèmes Informatiques (LCSI), École Nationale Supérieure d'Informatique (ESI), BP 68M, 16309, Oued-Smar, Alger, Algérie. <http://www.esi.dz>. ORCID : 0000-0003-4936-4019. E-mail: m_loudini@esi.dz

Abstract

In recent years, brain tumors become the leading cause of death in the world. Detection and rapid classification of this tumor are very important and may indicate the likely diagnosis and treatment strategy. In this paper, we propose deep learning techniques based on the combinations of pre-trained VGG-16 CNNs to classify three types of brain tumors (i.e., meningioma, glioma, and pituitary tumor). The scope of this research is the use of gray level of co-occurrence matrix (GLCM) features images and the original images as inputs to CNNs. Two GLCM features images are used (contrast and energy image). Our experiments show that the original image with energy image as input has better distinguishing features than other input combinations; accuracy can achieve average of 96.5% which is higher than accuracy in state-of-the-art classifiers.

Keywords: Brain tumor, Deep learning, VGG16 CNN, GLCM features.

Introduction

Magnetic Resonance Imaging (MRI) is one of the essential tools used in medical imaging by doctors and radiologists for detection and diagnosis of abnormalities in the brain. Brain tumor is one of these abnormalities. It's detected by radiologists during analysis of different MRI slices. If tumor is identified at an early stage, this allows to treat disease and increase the chance of survival for several patients. But, often large amounts of MRI data and different specific tumors are presented to radiologists, which leads to a long treatment time and a great risk of error, mainly when small slices are affected. However, detection of brain tumor and discrimination between different types of brain tumors are more complex and challenging tasks, automatic methods are desirable to address these problems. The purpose of this research is to develop an automatic method to classify brain tumor images into three types of the most common tumor (glioma, meningioma, and pituitary tumor) in T1-weighted contrast enhanced MRI (CE-MRI) images.

There are several studies on the brain tumors MRI, in which important methods are developed to classify them. This classification involves several steps. The most used are: preprocessing, features extraction and classification. The preprocessing step consist to improve the quality of images for example by filtering, normalizing and removing reflections that can go to segmentation. The step of features extraction is a crucial step because these features must allow classifiers at classification step to improve accuracy. In previous studies, intensity features, features first-order statistics, GLCM, Gabor filters, bag-of-words (BoW), fisher vector (FV), scale-invariant feature transformation (SIFT), and wavelet transform are the most frequently used features to describe brain tumor images. Javed et al. (2013) used texture features, fuzzy weighting, and SVM to multiclass classification, John (2012) developed a method based wavelet and co-occurrence matrix for feature extraction and Probabilistic Neural Network for classification. Jiang et al. (2013) proposed 3D voxel classification by AdaBoost using Gabor features and graph cut. Selvaraj et al. (2007) used first-order and second-order statistics features to binary classification (normal and abnormal) with least squares support vector machine (SVM) technique. Cheng et al. (2015) proposed three feature extraction methods, namely, intensity histogram, gray level co-occurrence matrix (GLCM), and bag-of-words (BoW) model and using augmented tumor region of interest (ROI), image dilation and ring-form partition achieves the accuracy of 91.28%. Cheng et al. (2016) used FV for brain tumor retrieval.

Recently, to avoid the step of features extractions and include it at classification step directly, deep learning techniques are appeared and have rapidly become powerful tools in medical diagnosis predominantly in brain tumor classification. Among works that used the same dataset as our work, we can cite: Paul et al. (2017) which used two types of neural networks in classification: fully connected, convolutional neural networks and achieved the

classification accuracy to 91.43%. Abiwinanda et al. (2019) proposed an automatic brain tumor segmentation/classification method based on Convolutional Neural Networks. CNN architecture consists of one convolution, max-pooling, and flattening layers, followed by a full connection layer, trained from scratch and achieved the classification accuracy of 84.19%. Khan Swati et al. (2019) used pre-trained deep CNN model and proposed a block-wise fine-tuning strategy based on transfer learning, achieved average accuracy of 94.82% under five-fold cross-validation.

In this paper, we propose an automatic method based on a combination of Convolutional Neural Network (CNN) with different inputs. We adopt the idea of transfer learning applied by Khan Swati et al. (2019) to avoid overfitting (Krizhevsky, Sutskever, & Hinton, 2012) like in works of Paul et al. (2017) and Abiwinanda et al. (2019) caused by using small dataset MRI (Cheng, 2017). We use pre-trained CNN based on VGG16 (Simonyan & Zisserman, 2014). The main contributions are:

- We propose a model pre-trained VGG16 in which all blocks are trainable.
- We extract GLCM matrix for the original images to form GLCM feature images as another input images.
- We propose to use GLCM feature images like input for CNN on one hand, and use the original images like input for other CNN, then we concatenate the outputs of each CNN to classify the images using the advantage of CNN and GLCM features to achieve enhanced accuracy.

This paper is organized as the following: In next section, we will present our methods. Section 3 we report the experimental details, in Section 4 we present the results analysis and comparison, provide brief discussion and finally, in Section 5, we conclude the paper with our future research direction.

Methods

Architecture of our model

The global approach is based on three main components; the first is a CNN model, the second is a GLCM feature extraction images module, while the third is a classification module. Figure 1 shows a general architecture of the proposed method where the model 1 of CNN, on the one hand, extracts features from the input images. On the other hand, others, CNN extracts features from GLCM images from the input images. The number of this CNN depends on the number texture images used as inputs. The features generated by each CNN module are concatenated to be used as inputs to the classification module.

The CNN used in our work is based on VGG16 which is a convolutional neural network model proposed by K. Simonyan and A. Zisserman (2014) at the University of Oxford. This

model achieved 92.7% top-5 test accuracy in ImageNet. Architecture of VGG-16 is shown in Table 1. It uses 5 convolutional layers and 5 maxpooling layers. The convolutional layers are all 3×3 convolutional layers with a stride size of 1 and the same padding. The pooling layers are all 2×2 pooling layers with a stride size of 2. The default input image size of VGG-16 is 224×224 . After each pooling layer, the size of the feature map is reduced by half.

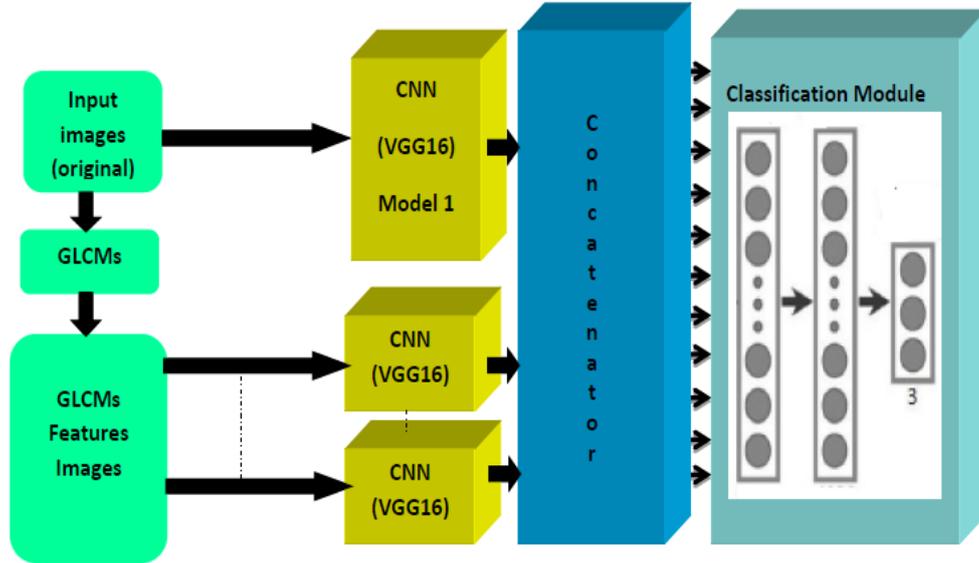


Figure 1. General architecture of our model

Gray-level co-occurrence matrix (GLCM) (Haralick, Shanmugam, & al., 1973) is the statistical method providing texture features information of an image that considers the spatial relationship between two pixels at specific distance and orientation angle. Each element $P(i, j)$ of the GLCM matrix P is simply the number of times that the pixel i having for neighbor a graylevel pixel j located at a certain distance D and orientation angle θ ($\theta = 0, 45, 90, 135$). The features extracted from GLCM used in this research are as following (Beyer, 2007):

The contrast measures the amount of local variations present in an image:

$$CONT = \sum_i \sum_j (i - j)^2 \cdot P(i, j) \quad (1)$$

Energy, also called the "uniformity", is a measure of the gray level concentration of intensity in GLCM:

$$E = \sum_{i,j} [P(i,j)]^2 \quad (2)$$

The classification module receives concatenated features extracted in CNNs modules. Dropout is used after to overcome the problem, followed by fully connected layers with 1024 neurons, softmax layer with three neurons.

Table 1. Architecture of VGG-16

Layer (type)	Output Shape	Param #
input_5 (InputLayer)	(None, 224, 224, 3)	
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	235980
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0

Data set

In this paper, we used dataset provided by Jun Cheng (Cheng, 2017). This dataset is 2-D slices with a large slice gape contains 3064 T-1 weighted CE-MRI of brain tumor images, in which 708 images with glioma, 1426 images with meningioma, and 930 images with pituitary tumors. The dataset (Cheng, 2017) was originally provided in Matlab.mat format where the image data in 512×512 uint16 formats with pixel size $0.49 \times 0.49 \text{ mm}^2$. Figure 2 shows example of the dataset from the three typical brain tumors.

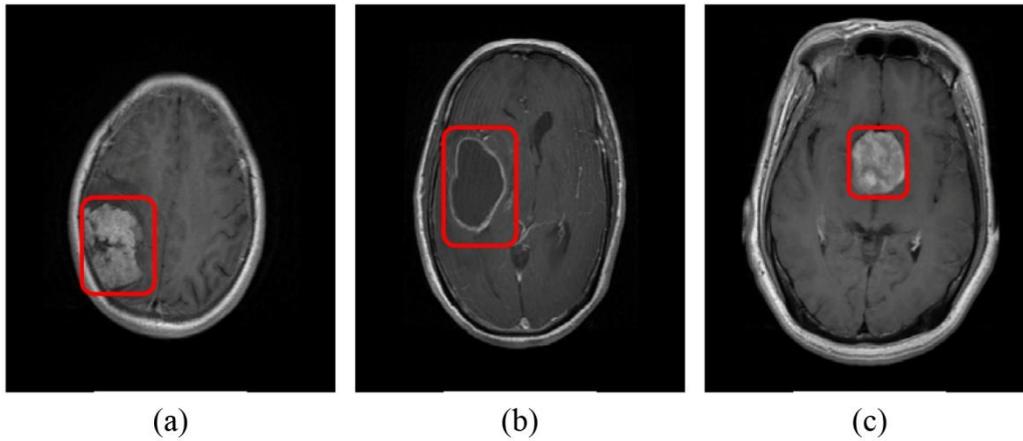


Figure 2. Examples of the three typical brain tumors:(a) glioma; (b) meningioma; and (c)pituitary tumor. Red lines indicate the tumor region.

Preprocessing

In this work, we have different inputs to CNN models. The first inputs are the original images of our dataset, these images are normalized and resized. We resize the images to 224×224 in order to speed time training and to reduce memory requirement. The other inputs are formed from the images of our dataset as follows.

For each image in the dataset, we calculate GLCM matrix and with this matrix we will generate texture images like GLCM-contrast and GLCM-energy. So, we form two other datasets and each one of them will make an input to other CNN.

Transfer learning

In medical image analysis, datasets are very rare and difficult to obtain, then when use CNN to train these datasets, the network will suffer from overfitting (Krizhevsky, Sutskever, & Hinton, 2012). Therefore, transfer learning is used. In practice, transfer learning is a reuse of pre-trained model on new model (Pan & Yang, 2010). However, for our CNNs, we initialized weights from the pretrained VGG16 model.

Experimental

In our experiment, we applied our proposed model to classify brain tumor on T-1 weighted CE-MRI dataset. We adopted the same experimental setup as in (Abiwinanda et al., 2019). The amount of images used to train CNNs is equalized for each type of tumors (class). For each CNN, we only used 700 images for each class where 500 used for training phase et 200 images used for validation phase and it's the same for the two others datasets. As mentioned above, we have two GLCM features images extracted from GLCM matrix (energy and contrast). Then, we have considered several cases for the inputs CNN which give us two

different architectures of our model. The difference is in the inputs used so the number of CNNs as show in table 2.

Table 2. Architecture inputs and number of CNN

	Number of CNN (VGG16)	Inputs
Architecture 1	2	CNN1: CE-MRI dataset images CNN2: GLCM Energy Images
		CNN1: CE-MRI dataset images CNN2: GLCM Contrast Images
Architecture 2	3	CNN1: CE-MRI dataset images CNN2: GLCM Energy Images CNN3: GLCM Contrast Images

Based on table 1, each architecture has CE-MRI dataset images as input to a CNN; the difference is in the input of other CNN. For architecture 1, the second CNN receives as input one of the features. Two cases are tested, namely: case 1 using GLCM energy images and case 2 used GLCM contrast images. For architecture 2, there are 2 second CNNs, each CNN receives as input respectively: contrast images and energy images.

The proposed model was implemented using Keras (Chollet, et al 2015) library in Python on a Ubuntu X86-64 machine with Processor intel® Xeon(R) Silver 4112 CPU @ 2.60GHz × 16, Graphic card llvmpipe (LLVM 7.0, 256 bits) and NVIDIA-SMI 418.67. We will use Adam optimizer in the training phase which updates the weight parameters and learning coefficients in each batch, and uses the stochastic gradient descent principle.

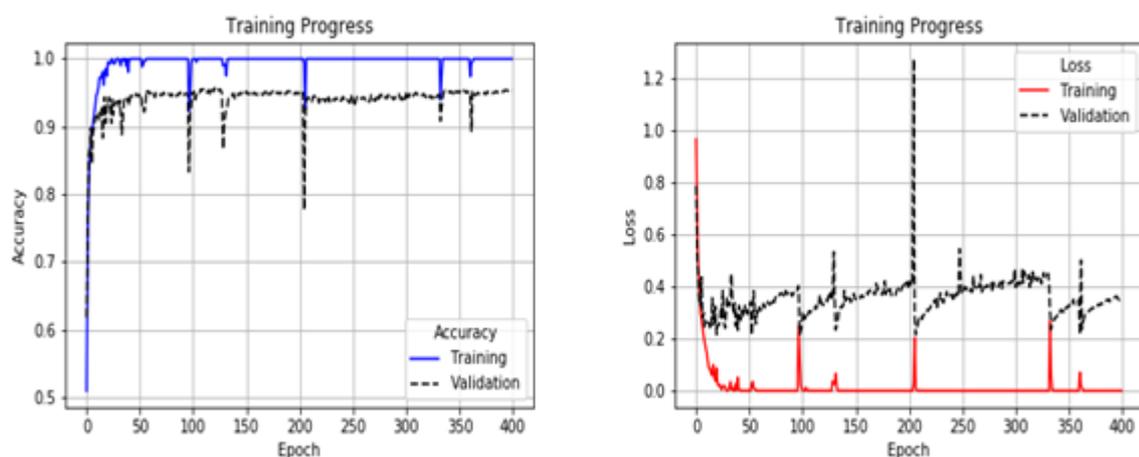


Figure 3. Progress training: the accuracy and loss history for the best architecture

All architectures are trained for 400 epochs, batch sizes of 16 and learning rate of 0.0001. After every epoch, we validate the training process with validation test. Cross-entropy is used for calculation loss (DiPietro, 2016). The process is stopped automatically if there is no improvement on validation test. The accuracy and loss of training-validation process for the best performing architectures are shown in figure 3, where we observe that accuracy training converge to 100% in most of epochs despite that during some epochs it goes down.

Results and discussions

Metric Performance

The performance of a system classification is based on four fundamental notions that are: A true positive (TP) is a result where the model correctly predicts the positive class. Similarly, a true negative (TN) is a result where the model correctly predicts the negative class. A false positive (FP) is a result where the model incorrectly predicts the positive class. A false negative (FN) is a result where the model incorrectly predicts the negative class. These notions are used to calculate metrics to evaluate the prediction performances of our model, which are: accuracy (ACC), specificity (SP), sensitivity (SN) and precision (PPR).

The correct classification rate is defined by:

$$ACC = \frac{(TP + TN)}{(TP + TN + FP + FN)} \times 100 \quad (3)$$

- Sensitivity (SN): describes how good the classifier is at classifying the correct tumors types (true positive rate).

$$SN = \frac{TP}{(TP + FN)} \times 100 \quad (4)$$

- Specificity: shows how good the classifier is at predicting the negative condition (true negative rate)

$$SP = \frac{TN}{(TN + FP)} * 100 \quad (5)$$

- Precision is the positive predictive rate (PPR)

$$PPR = \frac{TP}{(TP + FP)} \times 100 \quad (6)$$

Results analysis and discussion

The performance of the proposed model is evaluated in each architecture. Table 3 shows the classification accuracy of the two architectures mentioned above with their inputs.

Table 3. Classification performance of each architecture with their inputs

	Inputs CNNs	Accuracy
Architecture 1 (2 CNN (VGG16) (Two cases)	CNN1: CE-MRI dataset images CNN2: GLCM Energy Images	96.5 %
	CNN1: CE-MRI dataset images CNN2: GLCM Contrast Images	93.67%
Architecture 2 (3 CNN (VGG16)	CNN1: CE-MRI dataset images CNN2: GLCM Energy Images CNN3: GLCM Contrast Images	95.33%

From the result of table 3, we notice that the accuracies for all architectures are interesting, but the highest accuracy is obtained for architecture 1, it has an accuracy of 96.5 % in cases where we used energy images as input to the second CNN (also shown in figure 3),this indicates that energy features procures more information for CNN that helps model to classify types of tumors, whereas the low accuracy is obtained for architecture 2.

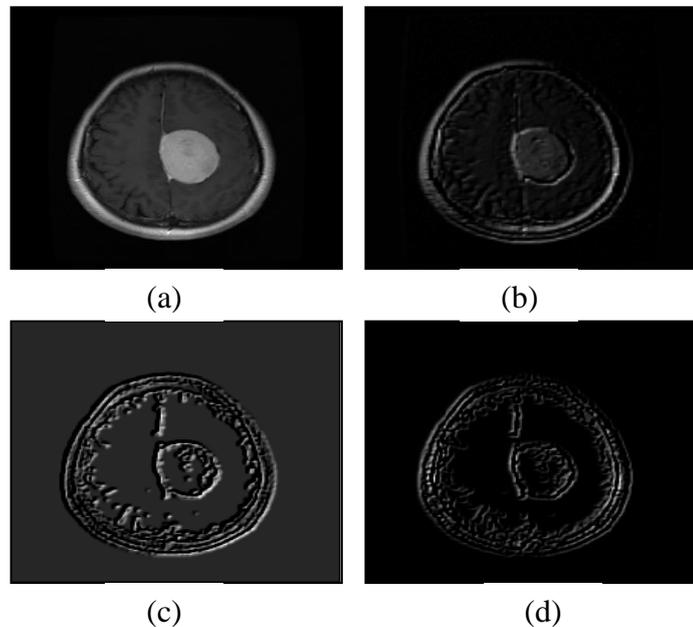


Figure 4. An example of features learned in convolutional layer (block1_conv2) of VGG16

In figure 4, we show an example. Features have learned in layer from architecture1 of VGG16, where figure 4(a) shows the original image. Figure 4(b) shows an example output of this image at convolutional layer block1_conv2 of CNN1. Figure 4(c) shows an example output of energy image at convolutional layer block1_conv2 of CNN2. Figure 4(d) shows an example output of a contrast image at convolutional layer block1_conv2 of CNN2. We can observe in the figure 4(b) compared with the input image in figure 4(a) that the bright areas of the image are positively activated while the dark areas are negatively activated. So, figure 4(c) clearly identifies the edges of the tumor compared to figure 4(d) where the edges are incomplete, which explains the best accuracy obtained by energy image as input to second CNN and less accuracy when contrast is added as input.

We analyzed the best performing architecture further in tables 4-7. Table 4 shows the classification performance with the two cases of inputs in the best architecture. Table 5 illustrates the average of sensitivity. Table 6 presents specificity and table 7 shows precision for individual specific tumor type (i.e. Glioma, Meningioma, and Pituitary tumor). From table 4, we can observe that from the two features of GLCM used in the best architecture, energy has the better performance.

In table 5, we can see that gliomas sensitivity is much higher than those of meningiomas and pituitary tumors, while in table 6 and 7 we observe than gliomas specificity and precision are lower which indicates that gliomas are easy to distinguish from the other two kinds of tumors.

Table 4. Classification performance of the best architecture

	Sensitivity	Specificity	Precision
CE-MRI dataset and Energy Images	96.5%	98.25%	96.57%
CE-MRI dataset and contrast Images	93.66%	96.83%	93.89%

Table 5. Classification Sensitivity for specific types of brain tumors of the best architecture

	Glioma	Meningioma	Pituitary tumor
CE-MRI dataset and Energy Images	97.5%	96%	96%
CE-MRI dataset and contrast Images	95%	91%	95%

Table 6. Classification Specificity for specific types of brain tumors of the best architecture

	Glioma	Meningioma	Pituitary tumor
CE-MRI dataset and Energy Images	96.5%	98.75%	99.5%
CE-MRI dataset and contrast Images	93.5%	98%	99%

Table 7. Classification Precision for specific types of brain tumors of the best architecture

	Glioma	Meningioma	Pituitary tumor
CE-MRI dataset and Energy Images	93.30%	97.46%	98.96%
CE-MRI dataset and contrast Images	87.96%	95.78%	97.93%

Comparison

The performance that was achieved by the best architecture where energy images and original image are used as inputs to CNNs was compared, at first, to model based on pre-trained VGG16 CNN with single input which is original image, the accuracy of this model is 94.83% compared with accuracy of our proposed model is less efficient. Second, we compare with state-of-the-art methods on the same CE-MRI dataset as shown in table 8.

Table 8. Accuracy comparison among the proposed and state-of-the-art methods on the CE-MRI dataset

Methods	Cheng, et al (2015)	Paul, et al (2017)	Abiwinanda, et al (2019)	Khan Swati, et al (2019)	Our method
Accuracy	91.28%	90.26%	84.18%	94.82%	96.5%

Table 8 describes the comparison of our proposed method with state-of-the-art methods on the same CE-MRI dataset. Our proposed method achieved the highest classification accuracy of 96.5%, which proves that the use of pre-trained VGG16 and features GLCM together can help to increase model accuracy, more specifically where there is not a large amount of training data available.

Conclusion

In this paper, we presented a model of classification based on using of pre-trained VGG 16 and GLCM features (contrast, energy). The concatenation of CNN outputs features helps a lot in increasing the performance of classification system. The relevance of these features depends strongly on the image used as input to CNN. Experimental results on our dataset

reveal that using GLCM energy images with original image as input to two CNN is efficient in comparison with contrast. Indeed, the accuracy is evaluated at 96.5%. It can be considered as a good result particularly in computer-assisted diagnostic tools for brain classification tumors compared with accuracy in the methods applied to the same CE-MRI dataset.

As future work, we intend to add a segmentation step to true classified images by our model in order to estimate the grade of tumors.

References

- Abiwinanda, N., Hanif, M., Hesaputra, S.T., Handayani, A., & Mengko, T.R. (2019). Brain tumor classification using convolutional neural network. In: *World Congress on Medical Physics and Biomedical Engineering 2018*. Springer, (pp. 183–189).
- Beyer, M.H. (2007). The GLCM Tutorial Home Page, [Online]. Available: <http://www.fp.ualgary.ca/mhallbey>, February, 2007 [Accessed: May 20,2014].
- Cheng, J. (2017). Brain tumor dataset. Distributed by Figshare. [Online]. Available: https://figshare.com/articles/brain_tumor_dataset/1512427/5.
- Cheng, J., Huang, W., Cao, S., Yang, R., Yang, W., Yun, Z., Wang, Z., &Feng, Q. (2015). Enhanced performance of brain tumor classification via tumor region augmentation and partition. *PLoS One* 10, e0140381. doi: 10.1371/journal.pone.0140381.
- Cheng, J., Yang, W., Huang, M., Huang, W., Jiang, J., Zhou, Y., Yang, R., Zhao, J., Feng, Y., Feng, Q., & Chen, W. (2016). Retrieval of brain tumors by adaptive spatial pooling and Fisher vector representation. *PLoS One* 11, e0157112. doi:10.1371/journal.pone.0157112.
- Chollet, F., & al. (2015). Keras. <https://github.com/keras-team/keras>
- DiPietro, Rob. (2016). A Friendly Introduction to Cross-Entropy Loss. Retrieved from <https://rdipietro.github.io/friendly-intro-to-cross-entropy-loss/>.
- Haralick, R. M., Shanmugam, K.& al. (1973). Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, (6):610–621.
- Javed, U., Riaz, M.M., Ghafoor, A., & Cheema, T.A. (2013). MRI brain classification using texture features, fuzzy weighting and support vector machine. *Prog.Electromagn. Res. B* 53, 73–88.
- Jiang, J., Wu, Y., Huang, M., Yang, W., Chen, W., & Feng, Q. (2013). 3D brain tumor segmentation in multimodal MR images based on learning population-and patient-specific feature sets. *Comput.Med. Imaging Graph*, 37, 512–521.
- John, P. (2012). Brain Tumor Classification Using Wavelet and Texture Based Neural Network. *Int J Sci Eng Res*, 3: 85–90.
- Khan Swati, Z. N., Zhao, Q., Kabir, M., Ali, F., Ali, Z., Ahmed, S., & Lu, J. (2019). Brain tumor classification for MR images using transfer learning and fine-tuning. *Computerized Medical Imaging and Graphics*, 0895-6111/2019, Elsevier
- Krizhevsky, A., Sutskever, I., & Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*. (pp 1097–1105)

- Pan, S.J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* 22, 1345–1359.
- Paul, J.S., Plassard, A.J., Landman, B.A., & Fabbri, D. (2017). Deep learning for brain tumor classification. *medical imaging 2017: biomedical applications in molecular, structural, and functional imaging*. International Society for Optics and Photonics. p.1013710.
- Selvaraj, H., Selvi, S.T., Selvathi, D., & Gewali, L. (2007). Brain MRI Slices Classification Using Least Squares Support Vector Machine. *Int J Intell Comput Med Sci Image Process*, 1: 21–33.
- Simonyan, K., & Zisserman, K. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, vol. abs/1409.1556.

Bibliographic information of this paper for citing:

- Nait Belaid, O., & Loudini, M. (2020). Classification of Brain Tumor by Combination of Pre-Trained VGG16 CNN. *Journal of Information Technology Management*, 12(2), 13-25.