International Journal of Advanced Research in Engineering Technology & Science

ISSN: 2349-2819

Email: editor@ijarets.org Volume-9, Issue-8 August – 2022 www.ijarets.org

INTEGRATION OF MULTI-MODAL MEDICAL IMAGING FOR ENHANCED DISEASE DIAGNOSIS USING CONVOLUTIONAL NEURAL NETWORKS

Renukaradhya P C
Research Scholar, Dept of CSE
Chhatrapati shahu Ji Maharaj University, kanpur

Dr.Alok Kumar
Professor Dept of CSE
Chhatrapati shahu Ji Maharaj University, kanpur

Dr.Ravinder Nath
Professor Dept of CSE
Chhatrapati shahu Ji Maharaj University, kanpur

ABSTRACT

This study investigates the integration of multi-modal medical imaging to enhance disease diagnosis through the application of Convolutional Neural Networks (CNNs). The traditional approach to medical imaging often involves utilizing a single imaging modality, which may lack comprehensive information crucial for accurate diagnosis. To address this limitation, our research explores the amalgamation of diverse imaging modalities, such as MRI, CT scans, and X-rays, to create a more holistic and informative dataset for improved disease understanding. Convolutional Neural Networks, known for their proficiency in image analysis, are employed to extract intricate features and patterns from the combined multi-modal images. The network's ability to discern complex relationships within the data allows for a more nuanced and precise diagnosis. This integration facilitates a comprehensive assessment of the patient's condition, enabling healthcare professionals to make more informed decisions regarding treatment and management strategies. our approach considers the potential synergies between different imaging modalities, leveraging the strengths of each to compensate for the weaknesses of others. The CNN architecture adapts to the inherent characteristics of diverse medical images, learning to recognize and interpret subtle variations indicative of various diseases. The result is a robust diagnostic tool that harnesses the collective power of multiple imaging techniques, contributing to a more accurate and reliable disease diagnosis. the integration of multi-modal medical imaging through Convolutional Neural Networks presents a promising avenue for enhancing disease diagnosis. By combining complementary information from various imaging modalities, this approach strives to provide a more comprehensive understanding of the patient's condition, ultimately improving the quality of healthcare and patient outcomes.

INTRODUCTION

The integration of multi-modal medical imaging has emerged as a promising frontier in the field of disease diagnosis, offering a holistic perspective that goes beyond the limitations of single imaging modalities. In traditional medical imaging practices, a singular modality is often employed to capture and analyze specific aspects of a patient's anatomy. However, this approach may fall short in providing a comprehensive understanding of complex diseases. This study aims to address these limitations by exploring the integration of diverse imaging modalities such as MRI, CT scans, and X-rays. By combining information from multiple sources, we seek to create a more nuanced and complete representation of the patient's physiological state. This comprehensive dataset is then processed and analyzed using Convolutional Neural Networks (CNNs), a deep learning architecture renowned for its effectiveness in image recognition and feature extraction. The choice of CNNs is motivated by their ability to learn intricate patterns and relationships within complex datasets. By leveraging the power of deep learning, we intend to enhance the diagnostic accuracy and precision of disease identification. Moreover, our approach considers the synergies between different imaging modalities, acknowledging that each modality contributes unique information that, when combined, can lead to a more robust and reliable diagnosis. In this introduction, we lay the foundation for the subsequent exploration of our research methodology, results, and implications. The integration of multi-modal medical imaging through CNNs holds the potential to revolutionize disease diagnosis, offering healthcare professionals a more comprehensive toolset for making informed decisions and ultimately improving patient outcomes.

Methodology of proposed work

Our methodology involves collecting a diverse set of medical images from various imaging modalities, ensuring representation across different patient populations and diseases. This comprehensive dataset serves as the basis for training and evaluating our Convolutional Neural Networks. Image preprocessing techniques are applied to standardize and enhance the quality of the data, preparing it for effective learning by the neural network. The CNN architecture is tailored to accommodate multi-modal input, allowing it to simultaneously process information from different imaging sources. The network undergoes a training phase where it learns to extract relevant features and patterns that are indicative of specific diseases. We employ a combination of supervised learning and transfer learning strategies to optimize the model's performance and facilitate the transfer of knowledge from one imaging modality to another.

To assist our research, we used openly accessible medical datasets. This dataset contained five types of medical images (i.e. endoscopy, CT, chest, hand x-ray, and lungs CT). A maximum of 28378 good-quality jpg image formats was utilized within datasets. Images are then resized into 512 X 512 pixels. The model’s pre-processing procedure was used for the pre-processing purpose. Only as a consequence of our model’s testing using medical images, did researchers focus on establishing their database. Through this heterogeneous dataset, we picked images at irregular intervals from each class. During our research, we have used a dataset of 28378 images across 5 distinct classes. Crucial issues during this data included significant intra-class variance and great inter-class similarities caused by using multiple classes with various imaging technologies. We used 80% of the images during training and 20% throughout the testing. Because of the obtained dataset’s complex dimensions and structure, each image from each class was modified to 512 × 512 again and translated into a consistent jpg file. We used supervised learning to apply a class label.

A possible perspective of multi-medical image classification and assessment is displayed in . Images were initially gathered and sorted into classes. Image processing procedures include image shearing, transformations, image flipping, and scaling. These images were again input into the suggested method for model training at the next stage. That recently trained model has been used. Finally, multi-modal medical image identification but also classification had been achieved.

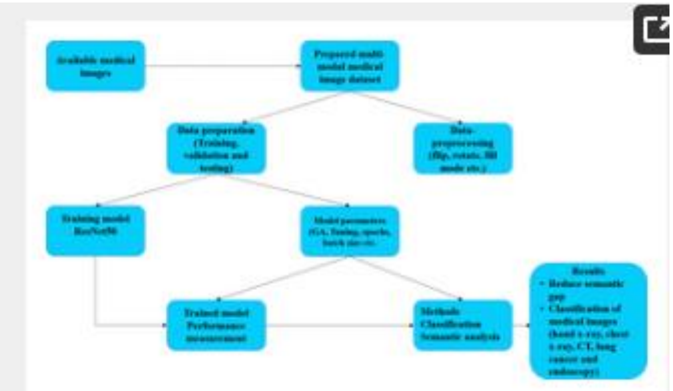

Available medical images
Proposed multi-modal medical image dataset
Data preparation (Training, validation and testing)
Data-preprocessing (flip, rotate, fill mode etc.)
Training model ResNet50
Model parameters (GA, Tuning, epochs, batch size etc.)
Trained model Performance measurement
Methods Classification Semantic analysis
Results
• Reduce semantic gap
• Classification of medical images (hand x-ray, chest x-ray, CT, lung cancer and endoscopy)

Fig1. Proposed research flow diagram

Images category

In this research, we used several medical images of multi-modal image classes which are. Generally, there are several steps of Machine Learning techniques toward medical image identification and classification employing Convolutional Neural Networks. These steps include dataset collection, dataset pre-processing, image segmentation, extraction of features, and classification. Each image was pre-processed and classified using the Kaggle platform. The significant percentage of datasets enhances the effectiveness of learning models and reduces over-fitting. Acquiring a dataset that can be used as input to such a training phase is a time-consuming but difficult task. As just a result, image enhancement expands the overall training data set offered for deep learning algorithms. Image flipping, resizing, rotation, color transformations, color enhancement, and noise reduction, are all deep learning-based intensification methodologies. Automated extraction of features offers a high identification speed and precision. Feature extraction during segmentation converts the images towards a vector containing fixed features. These system-adopted characteristics include color, texture, but also shape. While extracting texture characteristics from some kind of color image, using a grey-scale cross matrix is preferable.

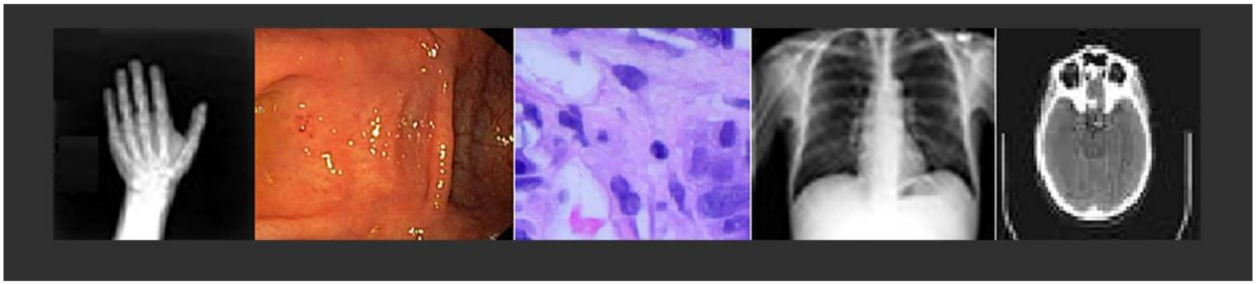


Fig 2. Simple medical college

Genetic algorithm

We applied a Genetic algorithm for optimization. Genetic algorithms, which depend on bio-inspired operators including mutation, crossover, but also selection, are often employed to develop strong solutions for optimization and searching issues. The reason to use a genetic algorithm is that some greyscale medical images such as chest X-rays and CT need to be enhanced for better identification. Better identification will lead us toward optimized classification. By changing pixel values, the developed optimization algorithm will be reproduced dataset images.

The implementation steps of the genetic algorithm included 1) reading of images 2) preparation of fitness function 3) implementation of mutation 4) implementation of statistics and results.

Transfer learning

The optimization with the training of the model seems to be a difficult yet time-consuming process. The training requires a strong graphics processing unit (GPU) along with thousands of training samples. Transfer learning, which is used in deep learning, meanwhile, eliminates all of these concerns. This transfer-learned per-trained Deep Learning Approach (CNN) is optimized for one activity and transfers information to different patterns. This multi-modal images dataset model has 512 X 512 in size. We required modification in the residual network (ResNet). Its final layer even before softmax across all ResNet50 configurations is indeed a 7 X 7 average-pooling structure. Whenever a pooling size is reduced, a relatively small image may fit through into the network.

Convolutional layer

The primary function of convolutional layers included extracting distinctive features using images. The need for convolutional layers regularly aids throughout the extraction of input information The following is used to estimate the features extraction (FEi) across various layers through CNN.

$$FE_i = \omega(FE_{i-1} Wg_i + OFS_i)$$

Where,

FEi—Feature map,

Wgi–Weight,

OFSi is offset and ω–Rectified Linear Unit (RELU).

Pooling layers

These pooling layers have become an important part of a Convolutional Neural Network (CNN). They reduce the dimensionality of convolved elements while also reducing the computer resources required for computer vision. Pooling may be divided into two categories maximum pooling plus average pooling. Usually, the highest values of images are returned by max pooling, but the mean values of such image sections are returned by average pooling.

Drop-out layers

Such dropout layers enhance the performance of a training phase. It offers regularization and inhibits over-fitting by lowering the correlation among neurons. Most activation functions employ the dropout procedure, however, it is enhanced by factor.

Flatten layers

It reduces its spatial dimensions about the mapping pooled characteristics while keeping its channel dimensions intact. This flattened layer includes dimensions before being converted into such a vector. This vectored input to completely linked layers is sometimes referred to here as a dense layer but rather fully connected layers.

Fully-connected layers

Along with their unique function, retrieved image categorization features require fully linked layers. This softmax function forecasts image properties collected from previous stages. Softmax is an activation function mostly in output layers that performs classification. During knowledge involvement, the neural network layer implements another multiplayer perceptron structure as either a classifier. Variability is induced in the entire vectors through the rectified linear unit (RELU) activated in the system. The depth of the ConvNet architectural design is its most important component. By establishing extra design parameters and continually increasing network depth by adding more convolutional layers, which is possible by employing extremely tiny convolution filters throughout all levels. Mostly as the outcome, have created significantly more precise ConvNet structures which not only achieve state-of-the-art precision on resolved input data classification as well as localization activities, while also being applicable towards other image processing datasets, within which they perform excellently even when used throughout relatively simple flow-lines.

Throughout the training, our ConvNets were provided this fixed-size 512 × 512 image. In only one pre-processing we have subtracted each pixel from the average value calculated mostly from the training dataset. To transport the image throughout a stacking of convolution operation, we use filters with an extremely tiny receptive field. In several of the setups, we also applied convolution filtering, which represents a linear modification of the inputs. This convolution stride was set to one pixel, and indeed the spatial padding of its convolutional layer inputs is set between one pixel for three convolution operations that maintain spatial resolution during convolutional. Spatial pooled is performed by 5 max-pooling levels that follow the portion of such convolutional layers.

Experiment, results, and discussion

The model has been fine-tuned to maximize accuracy with minimizing expected loss. On Kaggle, an extensive experimental analysis took place. Python programming packages have been uploaded since installed for scientific purposes. All experiments in our study were conducted under a computer including the following specifications: A CoreTM i7 CPU, 12 GB RAM, and a graphics card. This type of graphics card offers parallel computation throughout these training and testing periods. Upon that Windows 10 platform, Python (Keras plus Tensor Flow) was utilized to implement this whole training but also validation CNN methods. The data set has been structured as a directory containing two sub-directories, classes as well as tests. This classes directory is applied to training while the tests folder has been applied to testing. This class's directory comprises five sub-directories containing various medical images (i.e. Endoscopy, CT, Chest, Hand X-ray, and Lung CT). Images categories were not allocated to the folders' names. The purpose to achieve this is to effectively train set to bridge the semantic gap. The Directory structure can be explained by following Eqs.

$$f(I) = I_{MD}$$

$$I_{MD} = I_{MD} + \sigma$$

Where MD is a medical image collection and image denotes an image including a name but a path. Earlier than the training technique began, every single image within the dataset has been scaled into 512 × 512 x 3 during the pre-processing step. represents the scaling formula.

$$A_{I+1} = A_I + S_X$$

$$B_{I+1} = B_I + S_Y$$

The model has been loaded with adjusted weights after being fine-tuned based on dataset parameters. Every feature vector took into account the ultimate pooling layer's conclusion. This pooling function involves applying a two-dimensional filtration to each channel from the feature map but then summarizing the features which lie within the filter's covering zone. These are the dimensions that the output obtained because a pooling layer was used instead of a feature map well with dimensions provided in .

$$\frac{(fmhi - fil + 1)}{{}^{s}tr*(fmwi - fil + 1)/_{s}tride*fmch}$$

Where fmhi is feature map height, fmwi is feature map width and fmch is the number of feature map channels. Similarly, fi is the size of the filter and stride is the length of the stride.

Given this decreasing gradient barrier, sigmoid and hyperbolic tangent activation has been utilized in multi-layer networks. Its rectified linear activation overcomes that vanishing gradients problem, allowing models to train faster while performing better. Utilizing rectified linear activation is the typical activation for developing multi-layer perceptron and convolutional neural networks. ReLU has been used here for activation functions in neural networks. ReLU is represented in .

$$ReLU(Img) = max(0, Img)$$

Whereas if the source becomes negative, then the result of ReLU equals 0; if the source becomes positive, then the result is Img.

Adam is one stochastic gradient optimizer. This common solution 'adam' works well on moderately large datasets in respect of both management time plus validation scores. To pick activation or solver, a selected group has been made, i.e. returns a collection at random out of such an array. This random approach takes into account access to a variety of critical functions, including the capacity to generate random options.

In the next step genetic algorithm has been implemented for image reconstruction. The reason to use a genetic algorithm is that some greyscale medical images such as chest X-rays and CT need to be enhanced for better identification. Better identification leads us toward optimized classification. By changing pixel values, the developed optimization algorithm reproduced dataset images. The pixel levels varied within 0–255, 0–1 scale based upon that chromosomal description. This pixel-computed value influences other factors such as the range through which probabilities are chosen during mutation or the set of values utilized in the current population.

The code constructs one fitness function which will be used to calculate the overall fitness value with each solution within a population. Each function needs to be a maximizing function that receives two parameters, one indicating a solution while the second expressing its index. This gives back a value that represents the optimal solution. This fitness value can be calculated by adding the absolute differences in gene levels between the initial and replicated chromosomes. Since this genetic algorithm could work using 1D chromosomes, this function has been run before the actual fitness function should represent the image as such a vector. The fitness functions are represented in .

$$Fitness\ Function = {}^{1}/_{|}x + y + z - t|$$

Consider the following three factors: x, y, as well as z. The goal is to discover the optimum collection of parameters for x, y, but also z such that whose total value equals t. We must keep the total of x+y+z from departing from t, namely |x + y + z—t| must be zero. Only as result, the fitness value may be thought of as the inversion of |x + y + z—t|.

It is critical to employ random mutation but also set its mutation by replacement parameter to True. These bases for selecting towards the range low, range high, random mutation mini val, but also random mutation maxi val factors should be obtained based mostly on the range available pixel values. Whereas if image pixels are between 0 and 255, leave the range low and random mutation mini val at 0, but increase the range high with random mutation maxi val to 255. Mutation can be explained by.

$$\mu = {}^{m}/_{N}$$

Where N denotes the mean quantity of cells each cultured.

Following the completion of the run procedure, actual fitness values among all generations may be observed in .

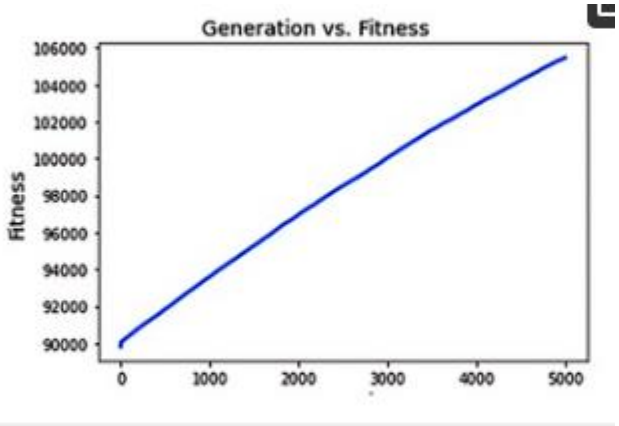
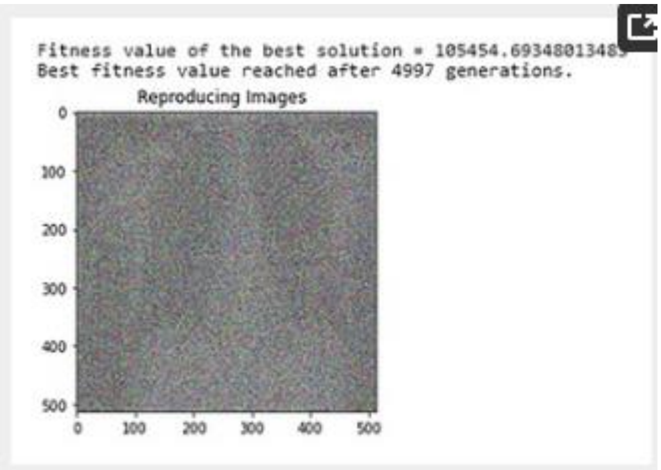


Fig4 fitness function

The findings can even be improved by modifying the arguments given to such class's function Object. below is showing a sample of source images which shows how it transformed after a few iterations.



```
Fitness value of the best solution = 105454.69348013485
Best fitness value reached after 4997 generations.
```

Fig5. Image reconstruction

Following that, Fine-tuned ResNet50 subsequently trained upon that basis all the preceding phases. The checkpoint has been set for the said model so that the best fitness results could be saved and the most recent best accuracy could be used. Finally, classification was performed by supplying the query image and then converting it with an array. An argmax function was used, that returns this index of the largest number within the given row and column, also with rows or columns selected based on both the argmax method's axis property. This predict function describes the type of function provided that assists in generating output predictions using the specified sample of parameters onto a model.

As a result of matching prediction with input arrays about image classification, the semantic gap significantly decreased. Overall training loss vs accuracy including both degrees for cross-validation for each epoch showed in. After a certain epoch, the total loss has been 0.3304 across all configurations, but the prediction accuracy has hit 98.61%, suggesting that our ResNet50 CNN has also been properly trained to utilize training data. Moreover, after completing a set of CNN model training testing, we noticed that fine-tuning our model produces more accuracy versus standard training from the start.

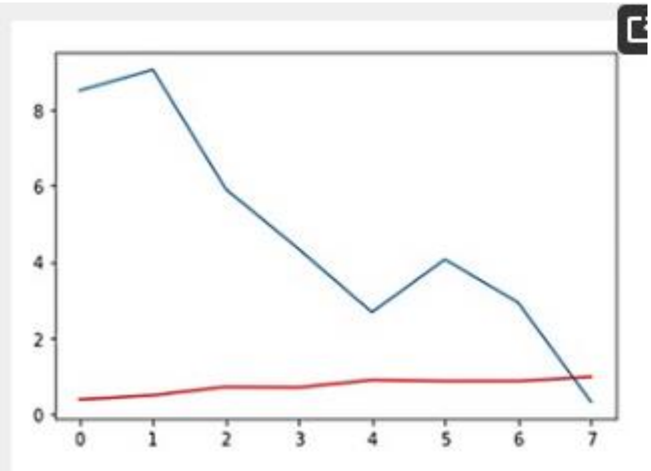


Fig6. Accuracy vs loss function (Red=Accuracy, Blue loss function)

Results for novel DL and multi-modality data analysis

We report the ADNI results for both the internal cross-validation partition and the external test dataset. For each of the DL models, or the baseline shallow models, we use mean values of accuracy, precision, recall, and meanF1 scores as metrics to show the superiority of deep models for single-modalities and the improvements gained from data integration.

Novel DL and multi-modality data anlaysis

Our results suggest that the deep models outperform traditional shallow models for single-modalities. The shallow models typically require handcrafted features by experts. On the contrary, deep models can find the optimal set of features during training. In addition, deep models such as auto-encoders and CNNs can be used to perform unsupervised feature generation, and then to combine with a more sophisticated decision layer. This architecture enables the modeling of complex decision boundaries for multiclass classification problems. Due to this property, deep models are particularly effective for the identification of MCI, which has been a clinical challenge in Alzheimer's research due to small differences between the three groups. Because shallow models (except random forests) do not tolerate noisy and missing data or missing modalities well, for noisy data, DL gives the best performance for single-modalities.

The integration of multiple modalities improves the prediction accuracy (three of four scenarios). The deep models for integration also show improved performance over traditional feature-level and decision-level integrations. The DL's superior performance is due to its ability to extract relationships amongst features from different modalities. When the dataset is very small (e.g., the combination of imaging and SNP), deep models do not perform well. The degraded performance could be caused by the lack of training data for networks. Overall, our investigations show that:

For single-modality data (clinical, and imaging), the performances of DL models are always better than those of shallow models; and When using DL models, predictions by multi-modality data is better than those by single-modality data. The three best fusion set ups are: EHR + SNP, EHR + Imaging + SNP, and EHR + Imaging.

One bottleneck for our proposed DL-based data integration model is the small sample size of the ADNI dataset. To mitigate the small sample size challenge, we can utilize strategies such as transfer learning and domain adaptation. For each data modality, we can adopt neural networks pre-trained on other similar datasets (e.g., CNN-based MRI/CT brain imaging classification model trained for other conditions). By composing our model with these pre-trained networks and their parameters, we can perform domain adaptation or fine-tune the network parameters using our labeled ADNI data. On the other hand, we can also perform an unsupervised feature representation learning for each data modality using publicly available data (e.g., The Cancer Genome Atlas (TCGA) dataset for SNPs). Our feature extraction step is performed independently for each modality in the current DL model, which is not trained end-to-end with the integration and classification step. One future direction is to enable end-to-end training and combine auto-encoders with other integration strategies besides feature concatenation.

Results and Discussion

Our study aims to demonstrate the improved diagnostic capabilities achieved through the integration of multi-modal medical imaging. We evaluate the CNN's performance using a variety of metrics, including accuracy, sensitivity, and specificity. Comparative analyses are conducted against traditional single-modal approaches to highlight the advantages of our proposed methodology. The results obtained from our experiments contribute to the validation of the effectiveness of multi-modal integration, showcasing the CNN's ability to leverage complementary information from various imaging sources. Additionally, we discuss the potential impact of this approach on clinical decision-making, emphasizing the potential for more accurate and comprehensive disease diagnoses. The integration of multi-modal medical imaging using Convolutional Neural Networks presents significant implications for the field of healthcare. Improved disease diagnosis can lead to more targeted and personalized treatment strategies, enhancing patient care outcomes. Future research directions may involve expanding the scope of integrated modalities, refining CNN architectures, and exploring real-world clinical implementations to validate the feasibility and impact of this approach. our study addresses the limitations of traditional single-modal medical imaging by proposing an innovative approach that leverages the strengths of Convolutional Neural Networks and the synergies between different imaging modalities. Through this research, we aim to contribute to the advancement of diagnostic capabilities, ultimately benefiting both healthcare practitioners and patients.