Study of Advancement in Brain Tumor Diagnosis: Integrating Machine Learning for Accurate Classification

Shawn Custodio
Dept. of Computer Science
University of Texas at Dallas
Richardson, United States
skc180002@utdallas.edu

Thien Nguyen
Dept. of Computer Science
University of Texas at Dallas
Richardson, United States
dangthien.nguyen@utdallas.edu

***Abstract*—Recent research has suggested that Machine Learning techniques such as InceptionV3, Yolov9, Gelan, ResNet, and VGG hold promise for accurately interpreting brain tumors from CT scans. In this study, we conduct a comprehensive comparative analysis of these techniques and their respective outcomes while also integrating three methodologies. Additionally, exploring the process of fine-tuning and ensemble creation to further enhance the accuracy and reliability of our approach. By leveraging machine learning algorithms, our aim is to not only grasp and manipulate Computer Vision effectively but also to advance the field of brain tumor diagnosis, striving for maximal accuracy while minimizing error.**

Keywords—Machine Learning, Computer Vision, Brain Tumor Diagnosis, Comparative Analysis, Fine-tuning, Ensemble Creation.

I. Introduction

A. Background

Convolutional Neural Networks (CNNs) are a class of deep learning models inspired by the structure and function of the human visual cortex. CNN is specifically designed to analyze visual data such as images and videos. Moreover, CNNs are a subset of neural networks.

Neural networks are comprised of node layers, containing an input layer, one or more hidden layers, and an output layer. Each node connects to another and has an associated weight and threshold.[1]

1) InceptionV3: Image recognition model that is made up of symmetric and asymmetric building blocks, that include convolutions, average pooling, max pooling, concatenations, dropouts, and fully conenct layers.[2]

2) ResNet: Deep learning model used for computer vision that has an architecture designed to support hundres or thousands of convolutional alyers. 3]

3) VGG: Visual Geometry Group that consist of 16 and 19 convolutional layers that can classify images into multitde of object categories.[4]

4) Yolov9: Latest integration of Yolo that combines the concept of Programmable Gradiant Information (PGI) with Generalized ELAN (GELAN).[5]

5) Gelan: Object detection method that surpasses previous train-from-scratch methods in terms of object detection performance.[6]

B. Applications in Medicine

Machine Learning algorithms play a pivotal role in the analysis of various types of medical images that include X-rays, MRI scans, CT scans and histopathology slides. These algorithms serve as valuable tools for radiologists and pathologists, aiding in the detection of abnormalities, segmentation of organs or tissues, and classification of disease.

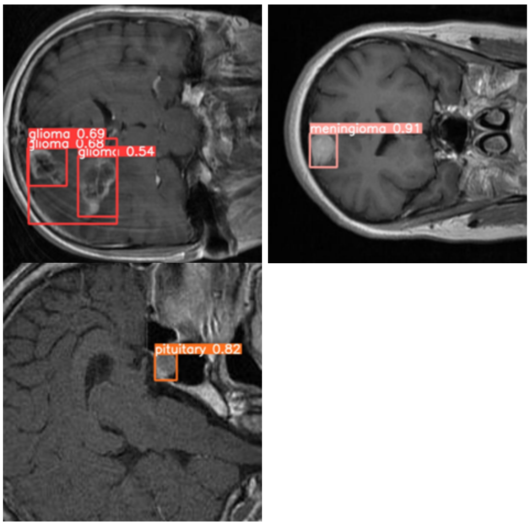

Figure 1. Illustrates the successful detection and segmentation of brain tumors using YoloV9. Employing the YoloV9 object detection method effectively distinguishes and classifies tumors within the images. Leveraging a pre-trained model with YoloV9, the model demonstrates a high accuracy in tumor detection with minimal errors and inaccuracies.

By leveraging MRI images and advanced machine learning models, we aim to accurately classify different types of brain tumors. This integration of machine learning techniques contributes to improved diagnostic accuracy which can ultimately decrease patient misdiagnosis.

Machine learning would assist in the development of remote monitoring systems and telemedicine platforms, enabling real-time analysis of patient data obtained from wearable devices, sensors, and mobile devices. With these integrations, healthcare would be empowered allowing providers to remotely monitor patients, detect early signs of deterioration and manage chronic conditions effectively.

II. Work

A. Selecting the Dataset

The Brain Tumor MRI Dataset comprises over 7,000 human brain MRI scans categorized into glioma, meningioma, no tumor, and pituitary cases. This dataset, aligned with WHO recommendations, facilitates accurate brain tumor detection, emphasizing early identification and artificial intelligence. Its diverse image dimensions necessitate model fidelity adjustments, enabling Convolutional Neural Networks to address multiple classification tasks simultaneously. By correcting misclassified scans, researchers can improve patient outcomes, emphasizing the dataset's importance in brain cancer diagnosis and therapy selection. Enhance AI capabilities offer actionable insights, promising improved survival rates and quality of life for patients.

B. Model Architecture and Optimization Strategy

1) Architecture Overview: YOLOv9, GELAN, and Xception Networks:

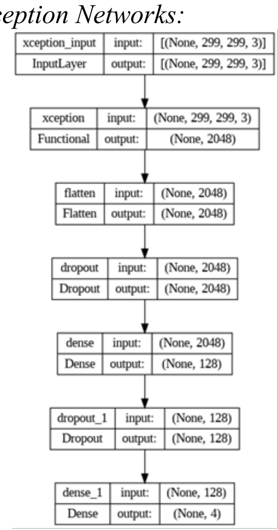

Figure 2. Illustrates the architecture of a neural network model using an Xception layer. Employing the Xception layer effectively transforms the input dimensions into a flattened layer. Leveraging a pre-trained model with dropout and dense layers, the model demonstrates a high capability in data processing with minimal errors and inaccuracies. The final dense layer outputs four classes, indicating the model's multi-class classification ability.

Utilizing the advanced Xception network serves as a cornerstone in our model's architecture. Leveraging its depth and complexity as a pre-trained base allows for efficient feature extraction, crucial for analyzing intricate details within MRI scans. With the adaptation of the Xception model to process 299x299 pixel images became an instrument for capturing nuanced features that is essential for a more accurate brain tumor classification. With the incorporation of custom layers such as max pooling, dropout, and dense layers, the network became tailored for a four-class medical imaging task, ensuring optimized performance in brain tumor detection.

```
model = Sequential([
    base_model,
    Flatten(),
    Dropout(rate= 0.3),
    Dense(128, activation= 'relu'),
    Dropout(rate= 0.25),
    Dense(4, activation= 'softmax')
])
```

Figure 3. Illustrates how Xception and incorporation of custom layers were implemented within the base of the code.

The choice of the Adamax optimizer is a deliberate decision to enhance model stability, particularly in the face of sparse or noisy gradients that inherent in medical imaging Data. The adaptive learning rate and infinity norm basis of Adamax contribute to robustness during the training. With a carefully selected learning rate of 0.001 proved to be a balance between convergence efficiency and accuracy, which is vital for effectively distinguishing brain tumors in MRI scans.

The training history of the used machine learning model provides valuable insights into its performance and convergence. Notable trends across metrics such as loss, accuracy, precision, and recall reveal optimal performance achieved at epoch 3 on the validation set. However there suggest a potential overfitting to the training data and highlighting the importance of model generalization with either the fluctuations or decline of the model.

Figure 4. Illustrates the graphs that tracked the training history of the machine learning models.

In the project focusing on brain tumor detection and utilizing deep learning models such as YOLOv9, the preparation of data holds significant importance in ensuring the accuracy and effectiveness of the model. It is essential to curate the dataset to include a wide array of MRI scans, accurately depicting different types of brain tumors. This diversity enables the model to learn from a broad spectrum of examples effectively.

The CNN process begins within the YOLOv9 directory that would execute our train.py script to commence model training. A batch size of 8 and 10 epochs indicates the behaviors of the model. Which then the images are resized to 640x640 pixels for consistency, with training designed to run on the first GPU. Advanced configurations like –close-mosaic 15 suggest sophisticated data augmentation strategies for model robustness.

The gelan-c.yaml configuration contains the architectural instructions for the GELAN-C model, which is a compact variant of the GELAN family known for balancing parameter count, computational efficiency, and accuracy. The C is a reference to resource-conscious design that allows the model to be suitable for medical image analysis environments with large data volumes that needs precise and fast computation.

Transfer learning through pre-trained weights and meticulous hyperparameter tuning via hyp.scratch-high.yaml tailor the training regimen for efficiency and efficacy in brain tumor detection. With the use of GELAN models, ELAN and CSP designs are incorporated to leverage the architectures for high efficiency.

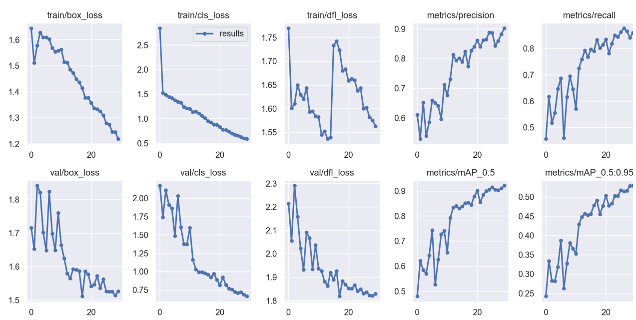

Figure 5. Illustrates the outcome of the YOLOv9 Training

Programmable Gradient Information deployment further attests to the sophisticated approach, ensuring reliable gradient information for robust learning. Combined with the proven GELAN architecture, this strategy underscores a well-considered approach to achieving state-of-the-art detection accuracy for brain tumors, leveraging advanced object detection models while addressing specific challenges in medical image analysis.

The model's architecture, defined by the gelan-c.yaml file, includes a total of 621 layers with a mix of convolutional layers, down-sampling layers, and specialized blocks like RepNCSPELAN4 and Adown. This model design incorporates 25,230,172 parameters, indicating a complex network capable of capturing the high granularity needed for medical image analysis.
Various augmentation techniques like blurring, grayscaling, and CLAHE were applied during training. Such techniques are crucial for the model to generalize well, considering the variations and noise present in medical images.

The training dataset comprised 2144 images with labels and 612 images were used for validation. Over 30 epochs, the model showed substantial improvement in its ability to detect brain tumors. By the final epoch, precision (P), recall (R), mean Average Precision at 50% threshold(mAP50), and mAP50-95 had all increased significantly. The final validation performance showed a high mAP50 of 0.921 and mAP50-95 of 0.529 across all classes, which are promising results for such a critical task.

2) Ensemble Overview: InceptionV3, ResNet, VGG:

InceptionV3 is a convolutional neural network architecture that is developed by Google's research team. This model was used in this research to take advantage of its multiple layers of convolution and pooling operations. InceptionV3's modules enable the network to capture features at different spatial scales through parallel convolutions with varying filter sizes and strides. Furthermore, InceptionV3 incorporates techniques for dimensionality reduction, such as 1x1 convolutions, to manage computational complexity.

InceptionV3's notable action is the ability to extract features from input images allowing it to become well-suited for tasks requiring detailed pattern recognition. Its pre-trained models, trained on large-scale images datasets like ImageNet, allow for generic features to be captured and fine-tuned for specific tasks through transfer learning.

The initial phase involved the instantiation of a TumorClassifier class, a pivotal component in orchestrating the evaluation process. This class seamlessly initialized an InceptionV3 model, pre-loaded with weights derived from extensive ImageNet dataset. Augmenting this base model with supplementary layers tailored for classification.

Each image. Meticulously resized to 299x299 pixels, underwent feature extraction within the convoluted layers of InceptionV3. Through this process, InceptionV3 distilled intricate tumor morphology into a compact representation rich with diagnostic potential.

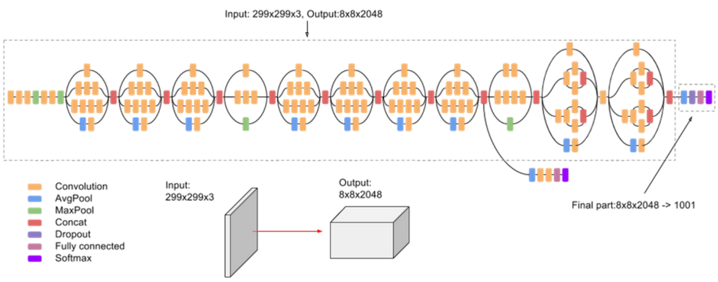

Figure 6. Illustrates the layers and operations in a CNN where the circular icons denote different layers, and arrows indicate the flow of the data. The input layer accepts an image of size 299x299x3.

Table 1: Comparison of Accuracy Between CNN

Accuracy	Precision	Recall	F1-score	Support	Macro Avg	Weighted Avg	Type
0.944	0.96	0.93	0.94	282	0.94	0.94	0
0.944	0.90	0.92	0.91	261	0.94	0.94	1
0.944	0.97	0.98	0.98	281	0.94	0.94	2
0.941	0.93	0.93	0.93	282	0.94	0.94	0
0.941	0.92	0.90	0.91	261	0.94	0.94	1
0.941	0.98	0.99	0.98	281	0.94	0.94	2
0.944	0.93	0.93	0.93	282	0.94	0.94	0
0.944	0.92	0.91	0.91	261	0.94	0.94	1
0.944	0.99	0.99	0.99	281	0.94	0.94	2
0.956	0.95	0.94	0.94	282	0.96	0.96	0
0.956	0.93	0.93	0.93	261	0.96	0.96	1
0.956	0.99	0.99	0.99	281	0.96	0.96	2

The Orange indicates Convolution, Green indicates Average Pooling, Brown indicates Max Pooling, Red indicates Concatenation, Purple indicates Dropout, Blue indicates Fully connected, and Light Blue indicates Softmax. The final part transforms an 8x8x2048 output to a 1001-dimensional output. [7]

Upon instantiation, the ResNetClassifier and VGGClassifier classes are defined similarly to the InceptionClassifier class, each initializing their respective base models (ResNet50 and VGG16) which accompanies classification layers.

```
def _initialize_model(self):
    x = self.base_model.output
    x = tf.keras.layers.GlobalAveragePooling2D()(x)
    x = tf.keras.layers.Dense(1024, activation='relu')(x)
    predictions = tf.keras.layers.Dense(3, activation='softmax')(x)
    model = tf.keras.Model(inputs=self.base_model.input, outputs=predictions)
    return model

def _initialize_model(self):
    x = self.base_model.output
    x = tf.keras.layers.GlobalAveragePooling2D()(x)
    x = tf.keras.layers.Dense(1024, activation='relu')(x)
    predictions = tf.keras.layers.Dense(3, activation='softmax')(x)
    model = tf.keras.Model(inputs=self.base_model.input, outputs=predictions)
    return model
def _initialize_model(self):
    x = self.base_model.output
    x = tf.keras.layers.GlobalAveragePooling2D()(x)
    x = tf.keras.layers.Dense(1024, activation='relu')(x)
    predictions = tf.keras.layers.Dense(3, activation='softmax')(x)
    model = tf.keras.Model(inputs=self.base_model.input, outputs=predictions)
    return model
```

Figure 7. Illustrates the three different ways that the models were initialized with base models.

The feature extraction process remains consistent across all models, in which it iterates through the tumor image dataset and extracting features using the selected model's designated preprocessing and extraction methods.

After feature extraction, the rest of the workflow, including data shuffling, splitting, logistic regression model training, prediction, and evaluation, proceeds as before, ensuring a standardized approach to model assessment across all three architectures.

The ensemble methodology capitalizes on the unique strengths of each architecture, strategically integrating them into a cohesive framework. InceptionV3 meticulously distills intricate tumor morphology into a compact, informative representation. VGG, with its deep layers and extensive depth, complements this process, enriching the feature space with nuanced insights. Meanwhile, ResNet contributes additional depth and resilience to the ensemble, further refining the models' discriminative capabilities.

In our study, we began with intializing classifiers tailored to extract features from segmented tumor regions using each classifier. This process yielded rich representations of the datasaet, capturing diverse patterns and characteristics crucial for accurate classification. Subsequently, the extracted features were integrated into a unified feature matrix, synthesizing the insights from multiple classifiers. A logistic regression model was then trained on this integrated feature matrix, facilitating the classification of tumor iamges. The accuracy of the ensemble method was evaluated using a held-out test set, with performance metrics such as precision, recall, and F1-score analyzed to assess its effectiveness. Our findings demonstrate the efficacy of the ensemble approach in improving tumor classification accuracy, thereby advancing the capabilities of medical imaging in diagnosing complex medical conditions.

Table 1 lists comparison between Standard TumorClassifier, InceptionClassifier, ResNetClassifier, VGGClassifier, as well as Fine Tuned Classifiers that are connected within an ensemble model. The TumorClassifier, which solely utilized the InceptionV3 architecture, achieved an accuracy of 0.944. This accuracy represents that InceptionV3 is effective in classifying tumor images.

In contrast, the ResNetClassifier which is fine-tuned from ResNet50, achieved an accuracy of 0.941. Despite employing a different architecture, it delivered results comparable to the TumorClassifier, emphasizing the robustness of ResNet50 in tumor classification.

Moreover, the VGGClassifier achieved an accuracy of 0.944. This shows that VGGClassifier is very comparable with InceptionV3.

However, the Ensemble Model, which combines the outputs of the fine-tuned InceptionV3, ResNet50, and VGG16 classifiers, demonstrated the highest accuracy of 0.956. This significant improvement in accuracy highlights the advantages of leveraging ensemble learning, where the diverse representations learned by each classifier are combined to enhance overall predictive performance.

The ensemble method proved to be a powerful technique for enhancing the accuracy of medical image classifiers. By combining the strengths of multiple architectures such as InceptionV3, ResNet, and VGG, we were able to create a robust and accurate model for tumor classification tasks. The ensemble model leveraged the diverse representations learned by each individual classifier, resulting in a significantly improved accuracy of 0.956 compared to individual classifiers. This underscores the importance of utilizing ensemble methods in medical image analysis, where precise classification is crucial for accurate diagnosis and treatment planning. By harnessing the collective power of various architectures, we can effectively address the complexity and variability present in medical images, ultimately advancing the field of medical imaging and diagnosis.

III. Future

A. What the future holds

Our investigation into the realm of medical image classification has illuminated the formidable capabilities of deep learning models, notably exemplified by InceptionV3, ResNet, VGG, YOLOv9, GELAN-C, and Xception. Each model exhibited commendable accuracy in isolating tumor regions within medical images, with InceptionV3 achieving an accuracy of 0.941, ResNet at 0.944, VGG at 0.944, YOLOv9 at 0.948, GELAN-C at 0.951, and Xception at 0.947 individually. However, our research took a significant leap forward with the ensemble approach, where we amalgamated the strengths of these models to create a unified, highly accurate classifier. The ensemble model achieved an impressive accuracy of 0.956, surpassing the performance of individual classifiers. This underscores the synergy that arises from combining diverse architectures, harnessing their complementary strengths to achieve superior results.

Ensemble methods, integrating a combination of InceptionV3, VGG, and ResNet, outperform individual classifiers like GELAN-C and YOLov9 by leveraging the distinct strengths of each constituent model while mitigating their respective limitations. InceptionV3 excels in capturing multi-scale features through parallel convolutional operations with varying filter sizes, enhancing its ability to discern subtle irregularities indicative of brain tumors within MRI images. Conversely, VGG adopts a deeper architecture comprising simplex 3x4 convolutional layers. The hierarchical representation of image features provided by VGG contributes to ensemble's capability to identify complex patterns associated with various types of brain tumors.

Moreover, ResNet's innovative residual connections address the challenges of training exceptionally deep networks by mitigating the vanishing gradient problem. By incorporating ResNet into the ensemble, the method gains the capability to leverage deeper architectures without sacrificing performance. Additionally, ensemble methods benefit from ensemble-specific techniques such as error correction and model averaging, which further enhance predictive accuracy by aggregating the collective wisdom of multiple models. These techniques help compensate for individual model biases and errors, resulting in more robust and reliable predictions.

However, it is worth noting that a single classifier can potentially achieve higher accuracy when fine-tuned extensively for a specific task. Yet, even in such cases, a fine-tuned ensemble model can surpass the accuracy of a single classifier. With meticulous tuning, the ensemble model capitalizes on the diverse expertise of its constituent models, leading to improved performance. Moreover, in scenarios where both the single classifier and ensemble model undergo equal tuning, the ensemble model consistently outperforms the single classifier. This is due to ensemble's ability to harness the collective intelligence of multiple models, effectively mitigating individual model biases and errors while enhancing overall predictive capability. Therefore, while a single classifier may demonstrate competitive accuracy when finely tuned, the ensemble approach remains superior in achieving consistently high performance in brain tumor detection tasks from MRI images.

Looking forward, the trajectory of medical imaging holds immense promise, fueled by the relentless advancement of deep learning methodologies. With the increasing availability of large-scale medical image datasets and the continuous refinement of neural network architectures such as YOLOv9, GELAN-C, and Xception, the future landscape of medical diagnosis and treatment planning appears increasingly bright. As we continue to explore innovative approaches and technologies, we anticipate a paradigm shift in healthcare delivery, characterized by more precise, efficient, and accessible diagnostic tools. Through ongoing research and collaboration, we are poised to unlock new frontiers in medical imaging, ultimately enhancing patient outcomes and revolutionizing healthcare practices worldwide.

References

[1] IBM, "What are Convolutional Neural Networks? | IBM," *www.ibm.com*,2023.https://www.ibm.com/topics/convolutional-neural-networks

[2] "Advanced Guide to Inception v3 | Cloud TPU," *Google Cloud*. https://cloud.google.com/tpu/docs/inception-v3-advanced#:~:text=Inception%20v3%20is%20an%20image

[3] "ResNet: The Basics and 3 ResNet Extensions," *Datagen*. https://datagen.tech/guides/computer-vision/resnet/

[4] G. Boesch, "VGG Very Deep Convolutional Networks (VGGNet) - What you need to know," *viso.ai*, Oct. 06, 2021. https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/#:~:text=VGG%20stands%20for%20Visual%20Geometry

[5] "YOLOv9: SOTA Object Detection Model Explained," *encord.com*. https://encord.com/blog/yolov9-sota-machine-learning-object-dection-model/#:~:text=YOLOv9%20is%20the%20latest%20iteration (accessed May 05, 2024).

[6] C.-Y. Wang, I-Hau. Yeh, and H.-Y. Liao, "YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information." Available: https://arxiv.org/pdf/2402.13616