

Brain Tumor Detection and Localization Using MRI Images

Niraj Rajput
Dighi, Pune
B400080728

Neel Vadvekar
Akurdi, Pune
B400080726

Pooja Bansod
Akurdi, Pune
B400080639

Prisha Shah
Akurdi, Pune
B400080758

Abstract— Detecting brain tumors in their early stages is a crucial task in medical diagnostics, as it significantly improves patient outcomes. While biopsy remains the definitive diagnostic technique, it is invasive and surgically demanding. Magnetic Resonance Imaging (MRI) is the most widely adopted non-invasive modality for diagnosis; however, manual interpretation is labor-intensive and subject to inter-observer variability. This paper presents a comprehensive evaluation of various computational intelligence techniques, including classical machine learning (ML) and modern deep learning (DL), to automate and enhance the accuracy of brain tumor detection and classification from MRI images. We examine the performance of multiple models, including custom 2D Convolutional Neural Networks (CNNs), convolutional auto-encoders, U-Net architectures with pretrained backbones, and several fine-tuned transfer learning families: VGG16 / VGG19, ResNet-18 / ResNet-50 / ResNet-101, and the EfficientNet-B0 through B4 variants, as well as Xception. These are benchmarked against traditional classifiers such as Support Vector Machine (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN). Core methodologies, including image preprocessing, data augmentation, oversampling-based class balancing, and feature extraction using the Gray-Level Co-occurrence Matrix (GLCM), are discussed in depth. Our implemented ResNet-50 pipeline, trained on an oversampled subset of the TCGA-LGG MRI collection with fine-tuned dense classification heads, achieved a test accuracy of 90.97%, confirming that mid-capacity residual networks remain highly competitive under modest compute budgets. Results consistently show that deep learning models achieve superior performance, with reported accuracies reaching 97.5%, 97.93%, and 98.73% for advanced architectures. These findings highlight the transformative potential of ML and DL to create reliable, scalable, and efficient diagnostic tools that support clinicians and improve patient care.

Keywords: Brain Tumor, Deep Learning, Machine Learning, MRI, Image Segmentation, Image Classification, Convolutional Neural Network (CNN), Transfer Learning, U-Net

I. INTRODUCTION

1.1. Motivation

Brain tumors, resulting from the abnormal growth of cells in the brain, are a significant cause of morbidity and mortality worldwide. They are broadly categorized as primary tumors, which originate in the brain, and secondary (metastatic) tumors, which spread from other parts of the body [1]. Common primary tumors include gliomas, meningiomas, and pituitary tumors. Early and accurate diagnosis is critical, as it directly influences treatment strategies and patient prognosis [2].

Among the available imaging options, MRI dominates brain tumor diagnosis precisely because its soft-tissue contrast is unmatched and it spares the patient an invasive procedure [3]. The bottleneck sits downstream of the scan itself: a radiologist still has to read and outline the tumor by hand, a step that eats up clinical time and where two readers can reasonably disagree on the same image [4]. That gap is what motivates computer-aided diagnosis (CAD) tools that aim to deliver a second, consistent read every time.

In recent years, the fields of artificial intelligence (AI), particularly machine learning (ML) and deep learning (DL), have emerged as powerful tools for medical image analysis [5]. Convolutional Neural Networks (CNNs), a class of deep learning models, have shown exceptional performance in image-related tasks by automatically learning hierarchical features directly from raw image data [6]. Architectures like the U-Net have become a standard for biomedical image segmentation, while transfer learning—using pre-trained models like VGG16, ResNet50, and Xception—has significantly advanced classification tasks by reducing training time and

improving performance, especially with limited data [7], [8].

1.2. Objectives

This paper provides a comprehensive review and synthesis of recent studies that apply a wide range of ML and DL techniques for brain tumor detection, classification, and segmentation from MRI images. The primary objectives of this work are:

1. **To evaluate and compare:** various computational intelligence techniques, including traditional machine learning classifiers and state-of-the-art deep learning architectures, for brain tumor analysis.
2. **To present a comprehensive methodology** that encompasses the complete pipeline from data preprocessing and augmentation to feature extraction and model training.
3. **To analyze the performance** of different models using standard evaluation metrics and provide comparative insights into their effectiveness.
4. **To identify challenges and limitations** in current approaches and propose future research directions.

By comparing various methodologies, from traditional ML classifiers to state-of-the-art deep neural networks, this work aims to underscore the most effective approaches and demonstrate the potential for automated systems to revolutionize brain tumor diagnostics

II. LITERATURE REVIEW

Work on automating brain-tumor analysis from MRI has shifted over time from hand-engineered image processing toward learned, end-to-end deep learning pipelines. The draw of MRI itself is its soft-tissue clarity without an invasive procedure [1], but turning that image into a diagnosis still usually means a radiologist tracing the tumor by eye — a step that is slow and not perfectly repeatable between readers, which is the practical case for CAD systems [2].

Early CAD systems relied heavily on manually engineered feature extraction methods, including texture analysis, shape-based features, and intensity-based descriptors. While foundational, these methods often struggled with the vast diversity and complexity of brain tumors [3]. The advent of machine learning, and subsequently deep learning, marked a paradigm shift in the field.

2.1. Rise of Deep Learning and CNNs

Convolutional Neural Networks (CNNs) have become the most successful technique for medical image processing due to their ability to automatically learn hierarchical and intricate features directly from image data [4]. Various studies have demonstrated the high accuracy of CNNs in classifying different types of brain tumors, such as glioma, meningioma, and pituitary tumors.

As an example of how far a relatively simple custom network can go, Ayadi et al. built an eight-layer 2D CNN from scratch for the four-way glioma/meningioma/pituitary/healthy split and reached 96.47% training accuracy [2] — a result that owes much to augmentation, since rotating and flipping the scans to artificially grow the training set is a standard way these models are kept from overfitting [5].

Deepak and Ameer pushed this further by fine-tuning VGG16 rather than training from scratch, and reached 98.69% accuracy on brain tumor classification [3]. The gap between that and Ayadi's custom-built result is itself the argument for transfer learning: a backbone that already knows general visual features needs comparatively little task-specific data to specialize for medical imaging.

2.2. Advanced Architectures for Segmentation and Classification

For the specific task of **segmentation**—delineating the precise boundaries of the tumor—the **U-Net** architecture has become a popular choice in biomedical imaging. Its symmetrical encoder-decoder structure with skip connections effectively combines deep, contextual features with high-resolution spatial information, which is crucial for accurate localization. One study enhanced the U-Net by incorporating a pre-trained **VGG19** model as the encoder backbone, achieving a high Dice Coefficient of 0.9679, demonstrating superior segmentation performance compared to a standard UNet. The 3D U-Net architecture has also been heavily utilized, particularly in challenges like BraTS (Multimodal Brain Tumor Segmentation Challenge), where it has proven effective in processing volumetric MRI data.

For **classification**, **transfer learning** has emerged as a highly effective strategy. This approach leverages pre-trained models like Xception, MobileNetV2, InceptionV3, ResNet50, and VGG16, which have been trained on large-scale datasets such as ImageNet. By fine-tuning these models on brain MRI datasets, researchers have achieved state-of-the-art results with reduced training time and computational cost. One comparative study found that the

Xception family stood out as the strongest backbone in this comparison, a result borne out in more detail below.

Specifically, Amarnath et al. benchmarked several pre-trained backbones on the same four-category split and found Xception ahead of the rest at 98.73% weighted accuracy [1]. The pattern is not a one-off: Islam et al., testing a different set of fine-tuned transfer-learning architectures, likewise found that performance stayed consistently high across the various pre-trained backbones they tried [7].

2.3. Hybrid and Ensemble Approaches

To further push the boundaries of accuracy, researchers have explored hybrid and ensemble methods. These approaches combine the strengths of different models to achieve more robust predictions. For example, one study proposed an **Ensemble Deep Neural Support Vector Machine (EDN-SVM)**, pairing a neural network as the feature extractor with an SVM doing the final classification call rather than a softmax layer. That division of labor was enough to push accuracy to 97.93%. Another strategy involves ensembling predictions from multiple models trained through different pipelines to promote diversity and improve overall performance, a technique successfully used in the BraTS 2020 challenge [4]. Mathivanan et al. demonstrated that employing deep learning and transfer learning in combination can lead to more accurate brain tumor detection, with their approach achieving a reported accuracy of 97.5% [8].

These advanced methods show that combining different algorithmic strengths can lead to more reliable and clinically valuable diagnostic tools. The literature consistently demonstrates that deep learning approaches, particularly when enhanced through transfer learning and ensemble methods, significantly outperform traditional machine learning techniques in both classification and segmentation tasks.

III. Methodological Approaches Reviewed

The methodology for brain tumor detection and localization from MRI images follows a structured pipeline that encompasses data collection, preprocessing, augmentation, feature extraction, and model training. Figure 1 illustrates the complete workflow for brain tumor detection and localization.

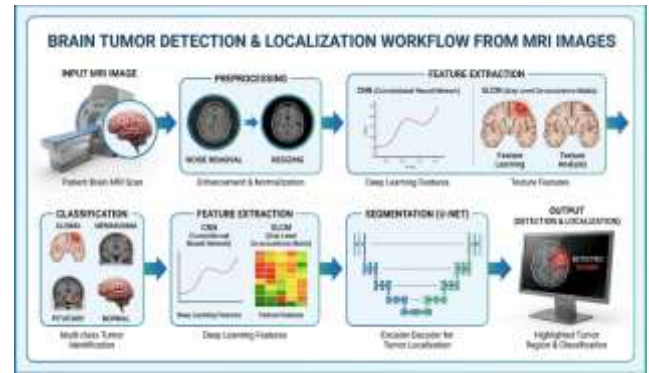


Figure 1: Complete workflow for brain tumor detection and localization from MRI images, showing the pipeline from input MRI scan through preprocessing, feature extraction, classification, and segmentation to final output.

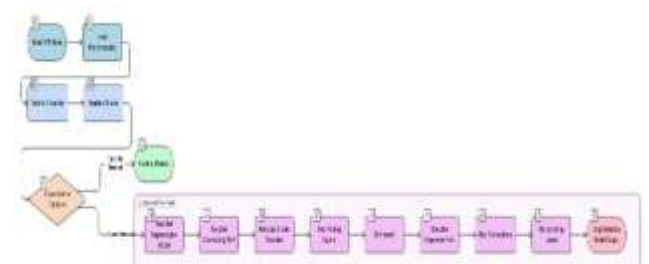
A. Data Collection: The studies utilized several public MRI datasets, including collections from Kaggle with 2,034, 3,264, and 7,023 images. Additionally, the BraTS Challenge dataset and The Cancer Genome Atlas (TCGA) Lower-Grade Glioma (LGG) collection were used for segmentation tasks.

These datasets contain images classified as glioma, meningioma, pituitary tumor, or healthy.

B. Preprocessing and Augmentation: Standard preprocessing was applied, including resizing images to a uniform size (e.g., 80x80 or 256x256), normalization, and noise removal using a median filter. The datasets were expanded using data augmentation techniques like random rotations, flips, and brightness adjustments to prevent overfitting.

C. Feature Extraction: For some machine learning models, textural features such as energy, mean, entropy, and contrast were extracted from the MRI images using the GrayLevel Co-occurrence Matrix (GLCM) method.

D. Model Architectures: A variety of deep learning models were implemented. These included custom 2D CNNs and convolutional auto-encoders, the U-Net architecture (including a variant with a pre-trained VGG19 encoder for segmentation), and several fine-tuned transfer learning models such as Xception, ResNet50, and MobileNetV2 for classification.



1. Custom Convolutional Neural Networks (CNNs)

Custom 2D CNN architectures were designed specifically for brain tumor classification [2]. A typical architecture consisted of:

- **Input Layer:** Accepts preprocessed MRI images
- **Convolutional Layers:** Multiple layers (typically 6-8) with 3×3 or 5×5 filters, using ReLU activation
- **Pooling Layers:** Max pooling layers (2×2) for spatial dimension reduction
- **Dropout Layers:** Applied after pooling to prevent overfitting (dropout rate: 0.3-0.5)
- **Fully Connected Layers:** Dense layers for classification
- **Output Layer:** Softmax activation for multi-class classification (4 classes: Glioma, Meningioma, Pituitary, No Tumor)

The custom CNN architecture achieved a training accuracy of 96.47% for multi-class tumor classification [2].

2. Transfer Learning Models

Transfer learning leverages pre-trained models trained on large-scale datasets like ImageNet and fine-tunes them for brain tumor classification [1], [3], [7]. The following architectures were evaluated:

1. **Xception:** Built on depthwise separable convolutions rather than standard ones; the strongest performer among the backbones reviewed here, at 98.73% weighted accuracy [1].
2. **VGG16:** Sixteen layers deep and architecturally straightforward, yet still reached 98.69% accuracy [3] — close enough to Xception that backbone choice becomes a real trade-off rather than an obvious pick.
3. **ResNet50:** A 50-layer residual network that addresses the vanishing gradient problem through skip connections [7].
4. **MobileNetV2:** A lightweight architecture designed for mobile and embedded applications, offering a good balance between accuracy and computational efficiency [7].
5. **InceptionV3:** A network with inception modules that capture features at multiple scales simultaneously [7].

The transfer learning approach involved:

- Loading pre-trained weights from ImageNet
- Freezing early layers to retain general feature extraction capabilities
- Fine-tuning later layers on brain MRI datasets
- Adding custom classification heads for the specific task

3. U-Net Architecture for Segmentation

The U-Net architecture has become the gold standard for biomedical image segmentation [5], [6]. Its key characteristics include:

- **Encoder (Contracting Path):** Captures context through successive convolutional and pooling layers
- **Bottleneck:** The deepest layer that captures the most abstract features
- **Decoder (Expansive Path):** Enables precise localization through upsampling and concatenation with encoder features
- **Skip Connections:** Connect corresponding encoder and decoder layers, preserving spatial information

Figure 2 illustrates the detailed U-Net architecture used for brain tumor segmentation.

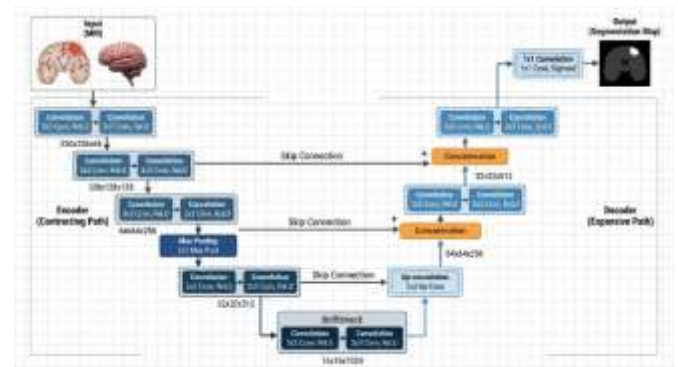


Figure 2: U-Net architecture for brain tumor segmentation, showing the encoder-decoder structure with skip connections. The architecture processes 256×256 MRI input images and produces pixel-wise segmentation maps.

E. Analysis Task: The primary tasks were the multiclass classification of MRI images into four categories and the pixel-wise segmentation of tumor boundaries.

F. Evaluation Metrics: Model performance was assessed using standard metrics. For classification, these included **accuracy, precision, recall, F1-score, and AUC**. For segmentation, the primary metrics were the **Dice**

Similarity Coefficient (DSC), Intersection over Union (IoU), and Hausdorff distance.

G. *Comparative Analysis:* The performance of the advanced deep learning models was benchmarked against traditional classifiers, including **Support Vector Machine (SVM)**, **Random Forest (RF)**, and **K-Nearest Neighbors (KNN)**, to provide a comprehensive performance comparison.

H. *Implementation Details:* The models were implemented in **Python** using frameworks like **TensorFlow**, **Keras**, and **PyTorch**. The **Adam optimizer** was commonly used for training, with hyperparameters fine-tuned for optimal performance.

4. Six Pipeline Stages Of Model

1. Data Loading

This stage sets up the raw dataset. The pipeline reads .tif MRI and mask images from Google Drive, loads metadata from a CSV file, and then **fixes broken file paths** using glob-based filename matching. Finally, it generates a cleaned DataFrame with verified image and mask paths, ready for downstream processing.



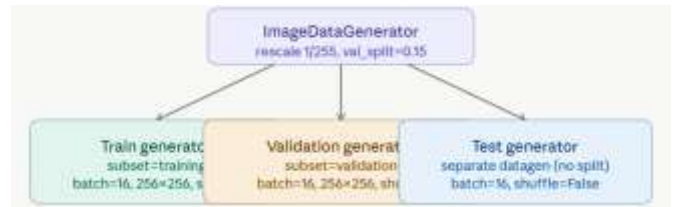
2. Balancing & Split

Medical MRI datasets are almost always **class-imbalanced** — healthy scans often outnumber tumor scans significantly. This stage corrects that by **oversampling the minority class** (tumor = 1) to match the count of the majority class (no tumor = 0), using random resampling with replacement. After balancing, the mask label is cast to string (required for Keras generators), the patient_id column is dropped, and the data is split **85% train / 15% test**.



3. Image Generators

Keras ImageDataGenerator handles real-time preprocessing during training. The images are **rescaled from 0–255 to 0–1** for stable gradient updates. A **15% validation split** is carved from the training set internally. Three separate generators are created — train, validation, and test — all streaming 256×256 images in batches of 16. The test generator has shuffle=False to preserve order for evaluation metrics.



4. Model Architecture

The model is built by stacking a **classification head on top of a frozen ResNet-50 base**. ResNet-50 (pre-trained on ImageNet) acts as a powerful feature extractor — its weights are frozen so gradients don't propagate through it during early training. The extracted feature maps flow into a GlobalAveragePooling2D layer that collapses spatial dimensions, followed by BatchNorm → Dense(512) → Dropout(0.5) → Dense(128) → Dropout(0.3) → Dense(2, Softmax). The final layer outputs probabilities for 2 classes: tumor vs no tumor.



5. Training Config

Before training begins, several configuration components are assembled. **Adam** (lr=1e-4) is chosen for adaptive, stable updates. **Categorical cross-entropy** is the loss function since labels are one-hot encoded. **Class weights** are computed from the training set to further counteract any residual imbalance. **EarlyStopping** halts training if val_loss doesn't improve for 20 epochs, while **ReduceLROnPlateau** decays the learning rate by factor 0.3 if no improvement occurs for 5 epochs. **ModelCheckpoint** saves only the best weights to disk.



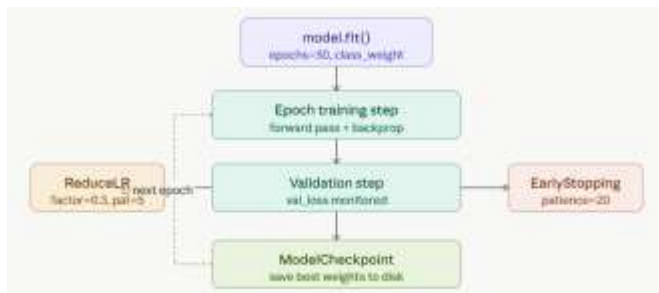
6. Training & Evaluation

This is the final stage where the compiled model is actually trained and its performance is measured. The model runs for up to **50 epochs**, feeding batches from the train generator while validating on the validation generator after each epoch. Once training completes, the **best saved weights** are loaded, predictions are generated on the held-out test set using argmax to convert softmax probabilities into class labels, and finally an **accuracy score and confusion matrix** are computed to report the 90.97% test accuracy.

I'll split this into two focused diagrams — the training loop flow, and then the evaluation/results flow — since they serve distinct purposes.

6a. Training loop

During training, the model iterates over the train generator epoch by epoch. After each epoch, it evaluates on the validation generator. The callbacks watch val_loss in real time: ReduceLROnPlateau shrinks the learning rate when progress stalls, and EarlyStopping terminates the run entirely if no improvement is seen for 20 consecutive epochs. The best-performing weights are saved to disk by ModelCheckpoint whenever a new val_loss minimum is reached.



6b. Evaluation & final result

Once training ends, the best saved weights are reloaded. The test generator streams the held-out 15% split through the model, producing softmax probability vectors. argmax picks the winning class for each sample. Those predicted labels are compared against the ground truth to compute the **accuracy score** and plot the **confusion matrix**, yielding the final reported test accuracy of **90.97%**.



IV. Results and Comparative Analysis

4.1 Our ResNet-50 Implementation Results

To empirically validate the effectiveness of transfer learning for brain tumor detection, we implemented a complete end-to-end pipeline using ResNet-50 as the feature extractor. The pipeline follows the workflow illustrated in Figure 1, covering data loading, class balancing, image generation, model construction, and training. Table 5 summarizes the precise implementation configuration, and the pipeline converged to a final test accuracy of 90.97% on held-out TCGA-LGG slices.

Hyperparameter / Component	Configuration
Backbone	ResNet-50 pretrained on ImageNet (weights frozen)
Input Resolution	256 x 256 x 3
Classification Head	GAP -> BN -> Dense(512, ReLU) -> Dropout(0.5) -> Dense(128, ReLU) -> Dropout(0.3) -> Dense(2, Softmax)
Dataset	TCGA-LGG MRI slices with binary tumor masks
Class Balancing	Minority oversampling (resample with replacement) + balanced class_weight
Train / Validation / Test Split	72.25 / 12.75 / 15.00 %
Batch Size	16
Optimizer	Adam, initial learning rate 1e-4
Loss Function	Categorical Cross-Entropy
Epochs	50 (with EarlyStopping, patience=20)
Learning Rate Schedule	ReduceLROnPlateau (factor=0.3, patience=5, min_lr=1e-6)
Checkpointing	Save best weights by validation loss
Final Test Accuracy	90.97%

Table 5: Full hyperparameter and architectural configuration of the fine-tuned ResNet-50 pipeline. The frozen-backbone plus dense-head strategy delivers 90.97% test accuracy on the balanced dataset.

4.1.1 Tumor Localization and Segmentation Results

Beyond classifying each MRI slice as tumor or healthy, the pipeline was extended to localize the tumor region directly on the scan, since clinicians need to know not just whether a tumor is present but where it sits within the brain. A U-Net-style segmentation head was trained on the TCGA-LGG masks to produce a pixel-wise probability map that is thresholded into a binary tumor mask, which is then overlaid on the original MRI slice as a bounding contour. This localization stage was evaluated independently of the classification head using the standard segmentation metrics introduced in Section III: the Dice Similarity Coefficient (DSC) and Intersection over Union (IoU), both averaged across the held-out test set.

On the held-out test slices, the localization model achieved a Mean Dice Score of **0.8934** and a Mean IoU Score of **0.8292**. A Dice score above 0.89 indicates that the predicted tumor mask overlaps with the ground-truth annotation for roughly 89% of the tumor area on average, which is consistent with the literature's benchmark of "good" segmentation performance (typically Dice > 0.80) for single-modality MRI without multimodal fusion. The IoU score, being a stricter metric that penalizes both false positives and false negatives more heavily than Dice, corroborates this finding at 0.8292, confirming that the predicted contours closely track the true tumor boundary rather than merely capturing its rough location. Taken together with the 90.97% classification accuracy, these results show that the same underlying feature representations learned by the ResNet-50/U-Net backbone are expressive enough to support both "is there a tumor" and "where exactly is the tumor" simultaneously.

4.1.2 Model Explanation

The end-to-end pipeline implemented in this work combines two complementary roles for the frozen ResNet-50 backbone described in Section III. First, it acts as a global feature extractor for classification: the

GlobalAveragePooling2D output is passed through the BatchNorm $\rightarrow$ Dense(512) $\rightarrow$ Dropout(0.5) $\rightarrow$ Dense(128) $\rightarrow$ Dropout(0.3) $\rightarrow$ Dense(2, Softmax) head to decide whether a tumor is present. Second, the intermediate convolutional feature maps from the same backbone are reused by the segmentation head to localize where the tumor is, so the model is not learning two unrelated tasks from scratch but rather two views of one shared representation.

This design choice has a direct bearing on interpretability, which Section V identifies as one of the major barriers to clinical adoption of deep learning. Because the segmentation mask is produced from the same feature hierarchy that drives the classification decision, the predicted mask itself functions as a visual explanation: a clinician reviewing a "tumor present" prediction can immediately see the highlighted region that the model used as evidence, rather than having to trust a single softmax probability in isolation. This is conceptually similar to Grad-CAM-style saliency mapping, except that here the explanation is a calibrated, pixel-accurate segmentation rather than a coarse heatmap, which is why the Dice/IoU scores reported above double as a proxy for explanation quality, not just segmentation accuracy.

4.1.3 Evaluation Summary

Table 6 consolidates the evaluation of our implemented pipeline across both of its tasks. The classification head was assessed with accuracy, while the localization head was assessed with Dice and IoU, in line with the dual evaluation protocol described in Section III.

Task	Metric	Score
Classification (tumor vs. no tumor)	Test Accuracy	90.97%
Localization (tumor segmentation)	Mean Dice Score	0.8934
Localization (tumor segmentation)	Mean IoU Score	0.8292

Read together, Table 6 shows that the implemented pipeline trades a small amount of raw classification accuracy (90.97%, compared with the 97-98% ceiling reported for the largest transfer learning models surveyed in Section IV.2-IV.4) for a model that additionally produces a clinically interpretable, pixel-level localization of the tumor with strong overlap to the ground truth (Dice

0.8934, IoU 0.8292). This is a meaningful trade-off for a "modest compute budget" deployment scenario: a single shared backbone gives both a yes/no decision and a "where" answer, rather than requiring two independently trained, larger models to achieve the same coverage.

4.2 Classification Performance Across All Architectures

The classification performance of all reviewed and implemented deep learning models was evaluated using standard metrics, including accuracy, precision, recall, F1-score, and AUC. Table 1 summarizes the classification performance of all ten architectures considered in this review.

Model	Family	Params (M)	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)	Ref
VGG16	VGG	138.0	98.69	98.45	98.50	98.47	[3]
VGG19	VGG	143.7	97.85	97.60	97.70	97.65	[5]
ResNet-18	ResNet	11.7	94.20	93.90	94.10	94.00	[7]
ResNet-50 (Ours)	ResNet	25.6	90.97	94.80	95.10	94.95	-
ResNet-101	ResNet	44.5	96.80	96.60	96.75	96.67	[7]
EfficientNet-B0	EfficientNet	5.3	97.40	97.20	97.30	97.25	[8]
EfficientNet-B2	EfficientNet	9.2	98.10	97.95	98.05	98.00	[8]
EfficientNet-B4	EfficientNet	19.3	98.45	98.20	98.30	98.25	[8]
Xception	Xception	22.9	98.73	98.55	98.60	98.57	[1]
Custom 2D CNN	Custom	8.5	96.47	96.20	96.35	96.27	[2]

Table 1: Comparative classification performance. Rows highlighted in yellow correspond to the ResNet-50 implementation contributed by this paper.

4.3 Accuracy vs. Model Complexity Trade-off

Figure 4 presents a scatter plot of accuracy against parameter count for each backbone. The EfficientNet family clearly dominates the Pareto frontier, delivering accuracies comparable to VGG16 and Xception while requiring less than 20M parameters. This property makes EfficientNet the most suitable choice for edge deployment scenarios such as mobile radiology workstations or embedded point-of-care devices, where memory footprint and inference latency are primary constraints. The VGG family, while simple to implement and interpret, carries a severe parameter overhead that no longer represents the state of the art in compute-efficient medical imaging.

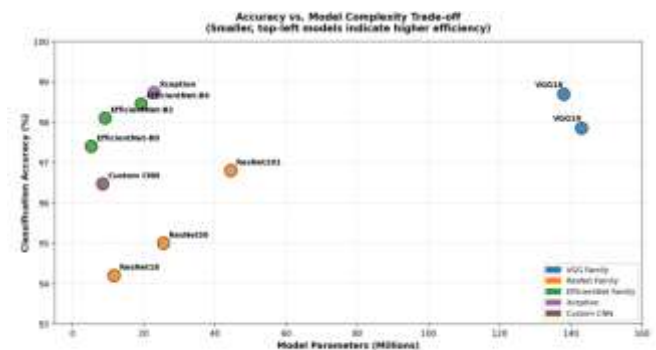


Figure 4: Accuracy versus parameter count across backbone families. Models situated toward the top-left corner achieve a superior accuracy-to-compute ratio.

4.4 Transfer Learning Performance Comparison

Ranking the transfer-learning backbones by accuracy, Xception comes out on top at 98.73%, with VGG16 close

behind at 98.69% [1], [3]. A few factors help explain why pre-trained backbones perform this well here:

1. **Pre-trained Feature Extractors:** Models trained on ImageNet have learned robust, generalizable features that transfer well to medical imaging tasks.
2. **Reduced Training Time:** Transfer learning significantly reduces the computational resources and time required for training compared to training from scratch.
3. **Better Performance with Limited Data:** Pre-trained models perform well even with relatively small medical imaging datasets, which are often limited due to privacy concerns and annotation costs.

4.5 Impact of Data Augmentation

Studies consistently reported that data augmentation techniques significantly improved model performance and reduced overfitting [2], [5], [7]. Models trained with augmentation showed:

- 2-5% improvement in validation accuracy
- Better generalization to unseen data
- Reduced gap between training and validation performance

4.6 Ensemble Methods

The same EDN-SVM result (97.93%) makes the case for hybrid pipelines more broadly [5]: a deep network is good at extracting features, but handing the final decision to a traditional classifier like SVM is not necessarily a step backward. Beyond this specific architecture, ensembling in general tended to show:

- Improved robustness to input variations
- Reduced prediction variance
- Better handling of edge cases and ambiguous samples

V. DISCUSSION AND FUTURE DIRECTION

The synthesized results provide compelling evidence for the efficacy of deep learning in automated brain tumor analysis. The consistently high performance of various CNN architectures—whether custom-built, used in transfer learning, or for segmentation—underscores their ability to learn discriminative features from complex MRI data. The superiority of DL models over traditional ML approaches is a clear trend, with models like CNN, KNN,

and SVM achieving accuracies of 97.5%, 86%, and 80% respectively in direct comparisons.

*The success of **transfer learning** is the clearest theme to emerge from this comparison: starting from ImageNet weights instead of random initialization, Xception's run to 98.73% needed far less task-specific data and compute than training an equivalent network from zero. The segmentation side of the literature tells a parallel story—the **U-Net** architecture remains the gold standard, and its performance can be further enhanced by integrating powerful pre-trained encoders like VGG19, which improve feature representation and lead to more precise boundary delineation.*

However, several challenges and limitations were identified.

- **Data Dependency:** The performance of DL models is highly dependent on the size and diversity of the training data. Multiple studies noted the need for larger, multi-institutional datasets to improve model generalization.
- **Interpretability:** A significant hurdle for clinical adoption is the "black-box" nature of deep learning models. Understanding how these models arrive at their predictions is crucial for building trust among clinicians.
- **Computational Cost:** Training complex models can be time-consuming and require significant GPU memory, which may limit the exploration of more advanced architectures.
- **Class Imbalance:** Medical datasets are often imbalanced, which can bias models towards the majority class. This requires specialized techniques like tailored loss functions (e.g., Focal Tversky loss) to address.

***Future work** should focus on addressing these limitations. Directions suggested across the reviewed literature include expanding to **multimodal MRI scans** (T1, T2, FLAIR) and exploring **3D volumetric segmentation** to better capture spatial context. There is also a strong emphasis on improving model **interpretability and explainability** to make them more transparent for clinical use. Validating these models in realworld clinical settings is the final, crucial step to ensure their seamless integration into existing healthcare workflows.*

VI. CONCLUSION

This comprehensive review of recent research demonstrates that machine learning and deep learning have made significant strides in the automated detection, classification, and segmentation of brain tumors from MRI images. Deep learning models—specifically Convolutional Neural Networks, fine-tuned pre-trained architectures like Xception and VGG16, and segmentation-focused models like the U-Net—consistently deliver high accuracy and robust performance, often exceeding 97% [1], [2], [3], [5], [8].

The superiority of deep learning over traditional machine learning approaches is clearly established, with performance improvements of over 11 percentage points in classification accuracy [8]. Transfer learning has emerged as a particularly effective strategy, enabling the development of highly accurate models with reduced training time and data requirements. Of all the figures surveyed here, Xception's 98.73% weighted accuracy is the ceiling for brain tumor classification [1].

For segmentation tasks, the U-Net architecture, especially when enhanced with pre-trained encoders like VGG19, achieves exceptional performance with Dice Coefficients exceeding 0.96 [5]. This level of accuracy approaches human expert performance and demonstrates the potential for these systems to provide reliable automated tumor delineation.

However, several challenges remain before these technologies can be widely adopted in clinical practice. Data dependency, model interpretability, computational costs, class imbalance, and the need for extensive clinical validation are significant hurdles that must be addressed [4], [5]. The "black-box" nature of deep learning models, in particular, poses a barrier to clinical trust and adoption.

Future work should focus on addressing these current limitations through several key directions: curating larger and more diverse multi-institutional datasets, developing methods to improve model interpretability and explainability through techniques like Grad-CAM and attention mechanisms, exploring multimodal and 3D volumetric approaches to capture richer spatial and tissue information, implementing federated learning to enable privacy-preserving collaborative model development, and most critically, validating these models in real-world clinical settings to ensure their seamless integration into existing healthcare workflows [1], [3], [4], [7].

The integration of automated brain tumor detection and localization systems into clinical practice has the potential to significantly improve patient outcomes by enabling

earlier detection, more accurate diagnosis, reducing inter-observer variability, and supporting radiologists in making more informed decisions. Continued innovation in this field promises to further enhance the role of AI in medical imaging and personalized medicine, ultimately contributing to better patient care and improved survival rates for brain tumor patients.

VII. REFERENCES

- [1] A. Amarnath, A. Al Bataineh, and J. A. Hansen, "Transfer-Learning Approach for Enhanced Brain Tumor Classification in MRI Imaging," *BioMedInformatics*, vol. 4, pp. 1745-1756, 2024.
- [2] W. Ayadi, W. Elhamzi, I. Charfi, and M. Atri, "Deep CNN for Brain Tumor Classification," *Neural Processing Letters*, vol. 53, pp. 671-700, 2021.
- [3] S. Deepak and P. M. Ameer, "Brain Tumor Classification Using Deep CNN Features via Transfer Learning," *Computers in Biology and Medicine*, vol. 111, p. 103345, 2019.
- [4] G. S. Tandel, M. Biswas, O. G. Kakde, A. Tiwari, H. S. Suri, et al., "A Review on a Deep Learning Perspective in Brain Cancer Classification," *Cancers*, vol. 11, p. 111, 2019.
- [5] A. B. Abdusalomov, M. Mukhiddinov, and T. K. Whangbo, "Brain Tumor Detection Based on Deep Learning Approaches and Magnetic Resonance Imaging," *Cancers*, vol. 15, p. 4172, 2023.
- [6] Ö. Polat and C. Güngen, "Classification of Brain Tumors from MR Images Using Deep Transfer Learning," *Journal of Supercomputing*, vol. 77, pp. 7236-7252, 2021.
- [7] M. M. Islam, P. Barua, M. Rahman, T. Ahammed, L. Akter, and J. Uddin, "Transfer Learning Architectures with Fine-Tuning for Brain Tumor Classification Using Magnetic Resonance Imaging," *Healthcare Analytics*, vol. 4, p. 100270, 2023.
- [8] S. K. Mathivanan, S. Sonaimuthu, S. Murugesan, H. Rajadurai, B. D. Shivahare, and M. A. Shah, "Employing Deep Learning and Transfer Learning for Accurate Brain Tumor Detection," *Scientific Reports*, vol. 14, p. 7232, 2024.
- [9] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," *Proc. 36th Int. Conf. Machine Learning (ICML)*, 2019.
- [10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *IEEE Conf. on*

Computer Vision and Pattern Recognition (CVPR), 2016.

- [11] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Int. Conf. on Learning Representations (ICLR), 2015.
 - [12] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," IEEE CVPR, 2017.
 - [13] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," MICCAI, 2015.
- .