MIJST
MIST International Journal of Science and Technology

Brain Tumor Detection and Classification from MRI Images with Systematic Screening of a CNN–Transformer Hybrid Framework

Mohammad Shahjahan Majib*[1], Faysal Ahmed[1], Shahriar Anwar[1], Md. Mamun Or Rashid[2] and Md. Ashraful Haque[3]

[1] Department of Computer Science and Engineering, Military Institute of Science and Technology (MIST), Dhaka, Bangladesh
[2] Department of Computer Science and Engineering, University of Dhaka, Dhaka, Bangladesh
[3] Department of Neurosurgery, National Institute of Neurosciences & Hospital, Dhaka, Bangladesh

***Corresponding Email:** *smajib@yahoo.com*

ARTICLE INFO

Article History:

Received: 06th November 2025
Revised: 04th March 2026
Accepted: 16th March 2026
Published: 30th June 2026

Keywords:
Brain Tumor MRI Classification
Hybrid CNN–Transformer
Magnetic Resonance Imaging (MRI)
Transfer Learning
Cross-Validation

ABSTRACT

Brain tumors are the most fatal diseases of the central nervous system. Accurate, early MRI classification can support timely clinical decisions and treatment planning. This study systematically identifies an effective and efficient hybrid architecture combining Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) for multi-class brain tumor classification from 2D MRI images through a systematic screening evaluation. In this study, 12 CNN–Transformer pairs were evaluated under an identical training procedure using the original Kaggle train/test split. A late-concatenation fusion of DenseNet-121 and Swin-Tiny yielded the highest Kaggle testing accuracy of 94.38% during the screening stage. To provide a more reliable and fair comparison, a 10-fold stratified Cross-Validation (CV) was conducted on the Kaggle Training set only. The best-ranked pair was EfficientNet-B0 + Swin-Tiny, achieving a mean CV accuracy of 98.29% ± 0.55% with selective fine-tuning. The final model was trained on the Kaggle Training set, evaluated on the Kaggle Testing set, and tested on an independent external dataset (TEST2). The final model achieved 93.94% accuracy on the Kaggle Test Set and 87.21% accuracy on TEST2. The result indicates a notable domain shift under external validation. In addition, paired statistical significance tests among top model pairs, ablation studies on the fusion strategy and trainable ratio, and computational cost analysis (training time, inference latency, GPU memory, and parameter counts) are reported.

This work is licensed under a *Creative Commons Attribution-Non-commercial 4.0 International License.*

1. INTRODUCTION

Magnetic Resonance Imaging (MRI) is the standard non-invasive modality for locating and classifying brain tumors in clinical routine. Manual review of MRI scans by radiologists and neurosurgeons is time-consuming and can lead to fatigue, especially in busy government hospitals where many scans arrive every day (Krishnan et al., 2024). Automated image classifiers can help in fast screening. They can flag high-risk scans. They can help prioritize urgent cases for neurosurgical review. These tools do not replace doctors. Instead, they aim to reduce delays in expert review and support clinical prioritization in resource-limited settings (Krishnan et al., 2024).

Deep learning has become the main approach for tumor detection and classification from MRI. Early work focused on Convolutional Neural Networks (CNNs). Transfer learning is often used from ImageNet. CNN backbones such as EfficientNet (Tan & Le, 2019), DenseNet121 (Huang et al., 2017), ResNet50, and ConvNeXt (Liu et al., 2022) can capture local texture, edges, and contrast changes in lesions. CNN-based pipelines with minimal pre-processing already achieve high multi-class accuracy on brain tumor datasets (Amin et al., 2023; Pacal, 2024).

More recently, Vision Transformers (ViTs) and their variants such as the Swin Transformer (Liu et al., 2021) have improved performance on medical images by modelling long-range spatial relations. Transformers also let the model attend to global context (Liu et al., 2021). In 2024, Krishnan et al. introduced a rotation-invariant ViT (RViT) that is robust to MRI slices being acquired or stored with

arbitrary orientations. They showed that this type of model can reduce false negatives and act as a front-line screening tool in clinics (Krishnan et al., 2024). Pacal (2024) proposed a Swin Transformer model with a residual multi-layer perceptron head for multiple brain tumor types in T1-weighted MRI. This trend shows that the community is moving toward transformer-based backbones. To improve classification accuracy, recent work has begun combining CNNs with Transformer-based vision models (Al Batainehet al., 2024; Benzorgat et al., 2024; Goceri, 2025; Panigrahi et al., 2025; Sharma & Ameta, 2025; Zahoor & Khan, 2025).

Table 1: Hybrid CNN–Transformer methods for MRI brain tumor classification (2024–2025)

Study	Task	Hybrid idea
Al Batainehet al. (2024)	classification	Swin Transformer + ResNet50V2 with transfer learning
Benzorgat et al. (2024)	classification	Ensemble of deep models with transformer integration
Goceri (2025)	Glioma grading & classification	Efficient CNN + Transformer blocks
Panigrahi et al. (2025)	classification	Transfer learning (DenseNet201) + Transformer self-attention
Sharma & Ameta (2025)	Early detection/classification, multi-modal MRI	3D-CNN + Swin Transformer fusion
Zahoor & Khan (2025).	MRI tumor analysis/classification	CE-RS-SBCIT: channel-enhanced CNN + ViT

However, it is still uncommon to systematically screen many CNN–Transformer pairs under the same training procedure and dataset split, then fairly compare them (Pacal, 2024; Krishnan et al., 2024). Based on this idea, this study addresses these gaps through a two-stage evaluation framework. First, twelve CNN–Transformer hybrid pairs are screened under identical settings on the original Kaggle train/test split. Second, to obtain a more reliable estimate of performance and enable fair model comparison, 10-fold stratified cross-validation (CV) is conducted using only the Kaggle Training set. Here, *cross-validation (CV)* refers to repeatedly training and validating the model across multiple stratified folds so that performance is measured across different subsets of the training data. In addition, the final selected model is evaluated on the Kaggle Testing set and on an independent external dataset (TEST2) to assess cross-domain generalization. This study also reports practical considerations for deployment. These include computational cost (training time, inference latency, GPU memory use, and parameter counts), as well as ablation studies on the fusion strategy and the trainable-ratio setting.

In the screening phase, DenseNet121 + Swin-Tiny gave the best accuracy on the Kaggle Testing set. Based on 10-fold cross-validation, EfficientNet-B0 + Swin-Tiny ranked highest, with a mean accuracy of 98.29% ± 0.55%**.** The final variant was then tested on the Kaggle Testing set and an external dataset. It reached 93.94% accuracy on Kaggle Testing and 87.21% on TEST2. This drop shows a clear domain shift during external validation. The full workflow is shown in Figure 1.

his study used the public Brain Tumor MRI Dataset from Kaggle (Nickparvar, 2021). It contains 2D brain MRI slices labeled into four classes: glioma, meningioma, pituitary and no_tumor. The data are provided in two-level directories, Training and Testing, with one subfolder per class, and the images are in JPEG/PNG format.

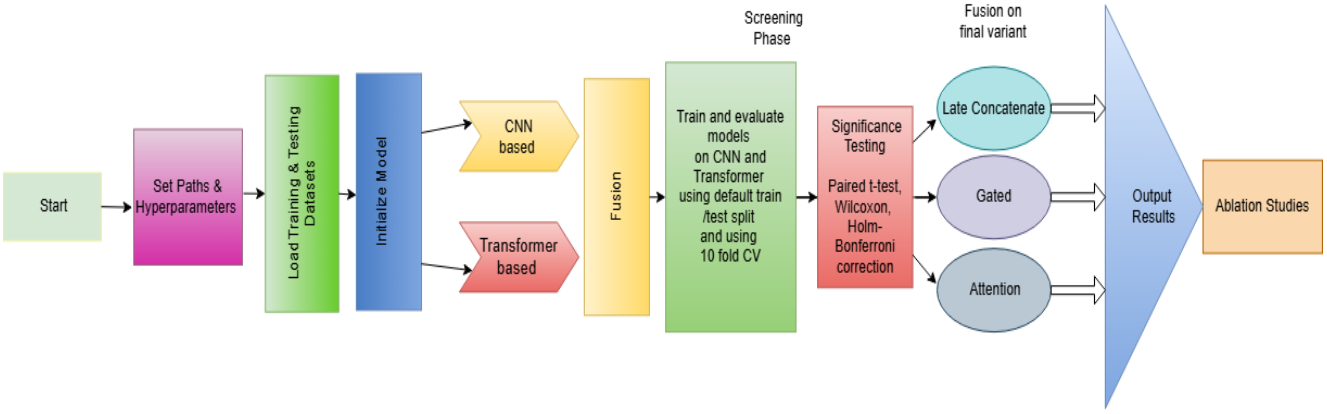

Figure 1: Overall workflow of the proposed two-phase hybrid CNN–Transformer pipeline

2. MATERIALS AND METHODS

2.1 Dataset and Pre-processing

This dataset has been widely used for MRI-based brain-tumor classification in recent peer-reviewed work (Khan & Park, 2024; Wang et al., 2025). Two evaluation protocols were used:

Original Split Protocol: The experiment is conducted by keeping the Training and Testing sets unchanged. From the

Training set, an internal 80/20 stratified split was created for training and validation. After training with early stopping, evaluation was performed on the Testing set.

10-fold cross-validation (CV) on training only: To reduce split bias and to allow fair model comparison, stratified 10-fold cross-validation (CV) was performed using the Kaggle Training set only. The same folds were used for all model pairs.

External Validation (TEST2): An additional dataset (TEST2) was used only for external evaluation. It was not used for training, and it was not part of the splitting process. This set contains 682 MRI images collected from the National Institute of Neurosciences & Hospital, Dhaka, Bangladesh [34]. The class distribution is glioma=65, meningioma=271, no-tumor=218, and pituitary= 128. Md labeled all images. Ashraful Haque, a neurosurgeon at the same hospital, with institutional approval.

The class distribution in the Kaggle dataset is shown in Table 2.

Table 2: Dataset summary and class distribution

Class	Train (n)	Test (n)	Total (n)	Share of Total (%)
Glioma	1,400	400	1,800	25.0
Meningioma	1,400	400	1,800	25.0
No-tumor	1,400	400	1,800	25.0
Pituitary	1,400	400	1,800	25.0
Total	5,600	1,600	7,200	100.0

All MRI slices are resized to 224×224 pixels, normalized with ImageNet mean and standard deviation, and then passed to the model pipeline to make it compatible with the Pretrained models, which is a common practice recommended by PyTorch.

2.2 Model Choice Rationale for Hybrid CNN–Transformer Architecture

To improve classification accuracy without making the model too heavy for practical use, only lightweight ("tiny") backbones were selected from both CNN and transformer families. The shortlisted models were chosen mainly based on model size (parameters), approximate compute cost (FLOPs/GMACs), and their reported ImageNet-1K top-1 accuracy from original papers or official sources. This selection supports a balanced comparison between performance and efficiency.

Three transformer backbones and four CNN backbones were used (Tables 3 and 4).

Table 3: Shortlisted transformer-based architectures

Rank	Variant	Params (M)	Compute	ImageNet-1K Top-1 (%)	ref
1	DeiT-Tiny (patch16-224)	~5.0	~1.2 GFLOPs	72.2	Touvron et al., 2021
2	EVA02-Tiny (patch14)	~5.8	~4.7 GMACs	80.66	Fang et al., 2023
3	Swin-Tiny	~28	~4.5 GFLOPs	81.2	Liu et al., 2021

Table 4: Shortlisted CNN-based architectures

Rank	Variant	Params (M)	Compute	ImageNet-1K Top-1 (%)	ref
1	EfficientNet-B0	~5.3	~0.39B FLOPs	77.1	Tan & Le, 2019
2	DenseNet-121	~8.4	~2.9 GFLOPs	~75.0	Huang et al., 2017
3	ResNet-50	~25.6	~3.8–4.1 GFLOPs	~75–76	He et al., 2016
4	ConvNeXt-Tiny	~28	~4.5 GFLOPs	82.1	Liu et al., 2022

The same MRI slice is passed through two parallel branches, and their features are fused before classification.

CNN branch: Four CNN backbones were evaluated: EfficientNet-B0, DenseNet-121, ResNet-50, and ConvNeXt-Tiny (Liu et al., 2022). Each backbone was initialized with ImageNet-pretrained weights (Deng et al., 2009). The original classification head was removed, and a global pooled feature vector was extracted.

Transformer branch: Three transformer backbones were evaluated: DeiT-Tiny, EVA02-Tiny, and Swin-Tiny. As in the CNN branch, the classification head was removed, and a pooled feature embedding was extracted.

Fusion and classifier: In the baseline setting, CNN and transformer feature vectors were concatenated (late

concatenation). The fused vector was passed to a lightweight classifier head consisting of (i) a fully connected layer that reduces the feature size to 256, (ii) a ReLU activation, and (iii) a final fully connected layer that outputs the four tumor classes.

Selective fine-tuning (trainable ratio): Selective fine-tuning (trainable ratio) was used to reduce overfitting and make the training more efficient. Most pretrained backbone weights were kept frozen, and only the last part of the network was updated. The default setting used a trainable ratio of 0.2, meaning that about 80% of the parameters were frozen and the last 20% were fine-tuned. To check how sensitive the model was to this choice, additional experiments were run with trainable ratios of 0.0, 0.1, 0.2, and 0.4.

Fusion alternatives: To examine whether late concatenation is the best fusion choice, two additional fusion strategies were tested on the selected best pair: gated fusion and attention-based fusion.

2.3 Phase-1 Screening on the Original Train/Test Split

Phase-1 screening compared all 12 CNN–Transformer pairs (4×3) under the same training pipeline and the original Kaggle split.

Inputs: The Kaggle dataset was used with separate Training and Testing folders (four classes). The Training folder was split into 80% training and 20% validation using stratified sampling. All models were trained with the same settings: image size 224×224, batch size 128, Adam optimizer, learning rate 1×10^{-4}, cross-entropy loss, maximum epochs = 40, early-stopping patience = 6, and random seed = 42. The baseline screening fusion method was late concatenation with a trainable ratio of 0.2.

External dataset (TEST2): An independent dataset (TEST2) was used for external validation.

2.4 Phase-2: 10-fold cross-validation (CV), significance testing, ablations, and final evaluation

Cross-validation (CV) is used to estimate performance stability across multiple splits. In this study, stratified 10-fold cross-validation was applied.

2.4.1 10-fold CV for all 12 pairs

To avoid test leakage and allow fair paired comparisons, a 10-fold CV was performed using only the Kaggle Training set. The same folds were used for every model pair. For each fold, the model was trained on 9 folds and validated on 1 fold with early stopping. Fold-wise validation accuracy was recorded and summarized as mean ± standard deviation.

2.4.2 Statistical significance tests To compare top-performing model pairs, fold-wise accuracies from the same cross-validation folds were compared using paired t-tests and the Wilcoxon signed-rank test. Holm–Bonferroni correction was applied to the paired t-test p-values. Paired Cohen's *d* was reported as an effect size.

2.4.3 Ablation studies

After selecting the best-performing pair, this study carried out two ablation studies using 10-fold cross-validation. First, it evaluated different fusion strategies by comparing late concatenation, gated fusion, and attention fusion, with the trainable ratio fixed at 0.2. Second, it examined the effect of different trainable ratios by comparing 0.0, 0.1, 0.2, and 0.4, with the fusion method fixed to late concatenation.

2.4.4 Final training and evaluation

After model selection, a final model was trained on the Kaggle Training set using a small internal validation split for early stopping. The final trained model was evaluated on Kaggle Testing for internal evaluation and TEST2 for external evaluation.

2.4.5 Computational cost reporting

For each model, total parameters and trainable parameters were reported. Training time, peak GPU memory usage, and inference time per image were also measured.

2.5 Implementation and training setup

All experiments were implemented in PyTorch using the timm library for pretrained CNN and Transformer backbones. Training and evaluation were performed on an NVIDIA RTX 5090 GPU. To ensure reproducibility, we used a fixed random seed (SEED = 42) and enabled deterministic cuDNN behavior.

Hyperparameters and training setup: All models were trained using the same settings for fair comparison. Images were resized to 224 × 224 and normalized using ImageNet mean and standard deviation. We used the Adam optimizer with a learning rate of 1×10^{-4}, batch size 128, and maximum 40 epochs. Early stopping was applied with a patience of 6 epochs, and the best checkpoint was selected based on validation accuracy. Mixed precision (AMP) was enabled during GPU training to speed up training.

3. Results and Discussion

3.1 Phase-1 Screening Results on the Original Kaggle Train/Test Split. The results of Phase-1 screening are shown in Table 5.

Phase-1 performed a systematic screening of 12 CNN–Transformer pairs (4 CNN backbones × 3 transformer backbones) using late concatenation fusion and selective fine-tuning (trainable_ratio = 0.2, i.e., the last ≈20% of backbone parameters were unfrozen). Models were trained on the Kaggle Training folder using an internal 80/20 stratified split for training/validation with early stopping, and then evaluated on the Kaggle Testing folder.

For external validation, the same trained model was evaluated on an independent dataset (TEST2) that was never used for training or model selection. The best Kaggle Testing accuracy in this screening phase was 94.38%, achieved by DenseNet-121 + Swin-Tiny. Parameter counts reported for the hybrid models were measured from the

Table 5 reports the screening results ranked by Kaggle Testing accuracy.

Rank	CNN	Transformer	Kaggle Testing Acc (%)	TEST2 Acc (%)	Infer (ms/img)	Train time (s)	Peak GPU mem (MB)	Params (M)
1	DenseNet-121	Swin-Tiny	**94.38**	85.31	0.704	700.65	2310.77	34.93
2	DenseNet-121	DeiT-Tiny	94.06	84.87	0.470	576.87	2104.50	12.79
3	ConvNeXt-Tiny	DeiT-Tiny	94.00	**90.59**	0.458	436.10	2064.64	33.59
4	ResNet-50	Swin-Tiny	93.81	87.95	0.655	753.77	2493.10	51.75
5	DenseNet-121	EVA02-Tiny	93.75	85.01	0.504	726.66	2104.48	12.77
6	EfficientNet-B0	Swin-Tiny	93.44	85.31	0.504	481.59	2940.85	32.05
7	ConvNeXt-Tiny	EVA02-Tiny	93.19	81.94	0.515	370.61	2064.62	33.57
8	EfficientNet-B0	DeiT-Tiny	93.19	87.07	**0.272**	628.66	2321.00	9.91
9	ConvNeXt-Tiny	Swin-Tiny	93.06	89.71	0.677	383.71	2536.77	55.73
10	ResNet-50	DeiT-Tiny	92.94	88.68	0.455	588.98	1814.14	29.61
11	EfficientNet-B0	EVA02-Tiny	92.69	83.84	0.352	566.46	2320.98	9.89
12	ResNet-50	EVA02-Tiny	92.06	85.45	0.500	818.23	1814.43	29.58

implemented architectures after removing the original classification heads and adding the fusion classifier.

On the Kaggle Testing split, the highest accuracy was achieved by DenseNet-121 + Swin-Tiny (94.38%). On TEST2, the best-performing model was ConvNeXt-Tiny + DeiT-Tiny (90.59%), suggesting a domain shift between the Kaggle and TEST2 datasets. The following figures are presented for the selected screening variant (DenseNet-121 + Swin-Tiny). The confusion matrix of the best screening model is presented in Figure 2.

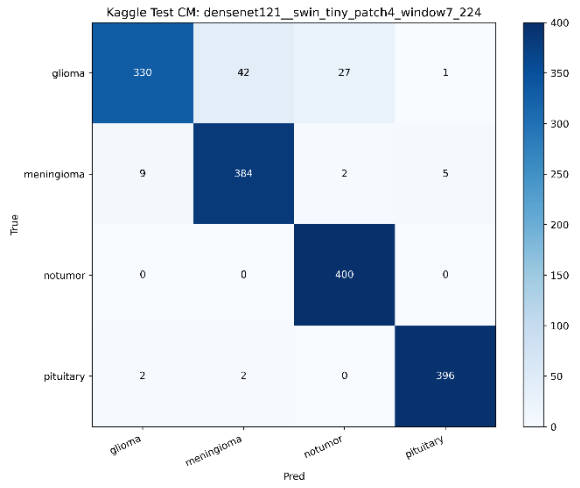

Figure 2: Confusion matrix of the best screening model (DenseNet-121 + Swin-Tiny) on Kaggle Testing

The OvR ROC curves for the best screening model are shown in Figure 3**.**

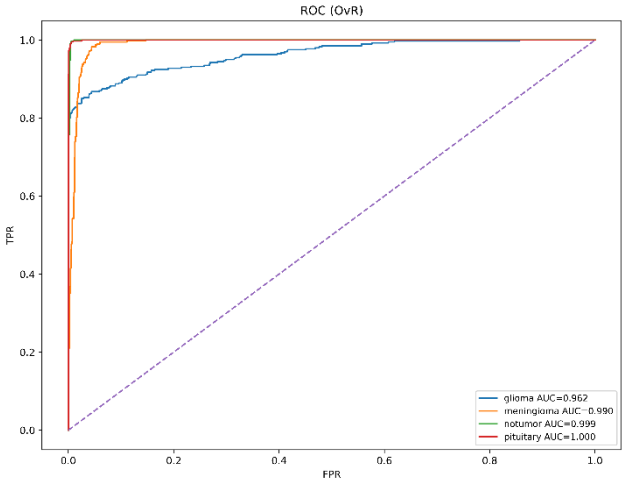

Figure 3: One-vs-rest (OvR) ROC curves on Kaggle Testing for the best screening model (DenseNet-121 + Swin-Tiny)

The OvR precision–recall curves for the best screening model are shown in **Figure 4**.

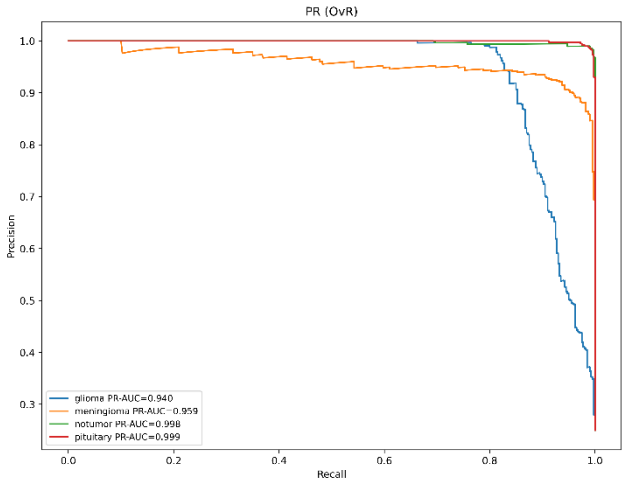

Figure 4: One-vs-rest (OvR) precision–recall curves on Kaggle Testing for the best screening model (DenseNet-121 + Swin-Tiny)

Training/validation accuracy and loss curves are shown in Figure 5 and Figure 6**,** respectively.

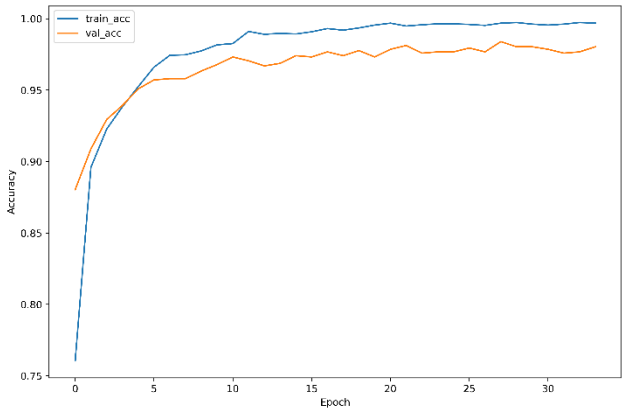

Figure 5: Training vs validation accuracy curve (DenseNet-121 + Swin-Tiny)

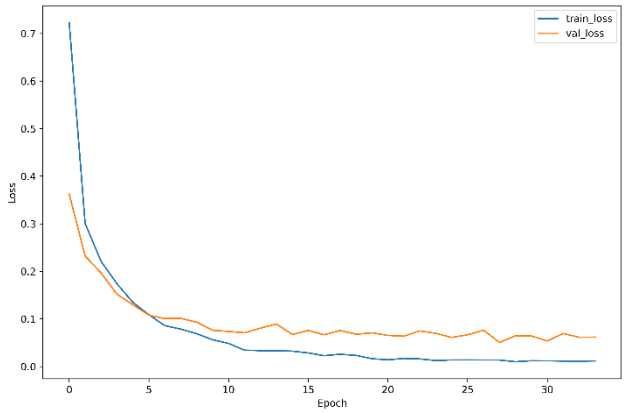

Figure 6: Training vs validation loss curve (DenseNet-121 + Swin-Tiny)

3.2 Ten-fold cross-validation (CV) on Training set for robust ranking: To obtain a more reliable estimate than a single split, 10-fold stratified cross-validation (CV) was performed on the Kaggle Training set (Training-only CV). In each fold, models were trained on 9 folds and validated on 1 fold. The same folds were used across all 12 pairs to enable paired statistical testing. Results are reported as mean ± standard deviation across folds.

The best CV mean accuracy was obtained by EfficientNet-B0 + Swin-Tiny, achieving 98.29% ± 0.55%**.**

Table 6: Ten-fold CV results for all 12 pairs (Training-only CV; late_concat; trainable_ratio = 0.2).

Rank	CNN	Transformer	CV mean acc (%)	CV std (%)	Mean train time (s)	Mean peak mem (MB)	Mean infer (ms/img)
1	EfficientNet-B0	Swin-Tiny	**98.29**	0.55	390.10	2446.92	0.518
2	ConvNeXt-Tiny	Swin-Tiny	98.21	0.47	314.90	2821.05	0.677
3	DenseNet-121	DeiT-Tiny	98.11	0.70	370.55	1508.18	0.445
4	ConvNeXt-Tiny	DeiT-Tiny	98.09	0.51	311.21	2274.25	0.490
5	DenseNet-121	Swin-Tiny	98.07	0.56	337.82	2562.86	0.705
6	EfficientNet-B0	DeiT-Tiny	98.00	0.73	393.50	1666.22	**0.245**
7	DenseNet-121	EVA02-Tiny	97.96	0.61	365.64	1538.63	0.485
8	ConvNeXt-Tiny	EVA02-Tiny	97.95	0.58	324.67	2323.90	0.508
9	ResNet-50	Swin-Tiny	97.89	0.83	405.62	2592.24	0.649
10	ResNet-50	DeiT-Tiny	97.80	0.64	453.99	1885.99	0.491
11	EfficientNet-B0	EVA02-Tiny	97.73	0.64	427.28	1692.81	0.319
12	ResNet-50	EVA02-Tiny	96.86	0.70	405.15	1909.33	0.446

Fold-wise validation accuracy and loss trends for EfficientNet-B0 + Swin-Tiny across the 10 stratified folds are shown in **Figure 7** and **Figure 8**, respectively.

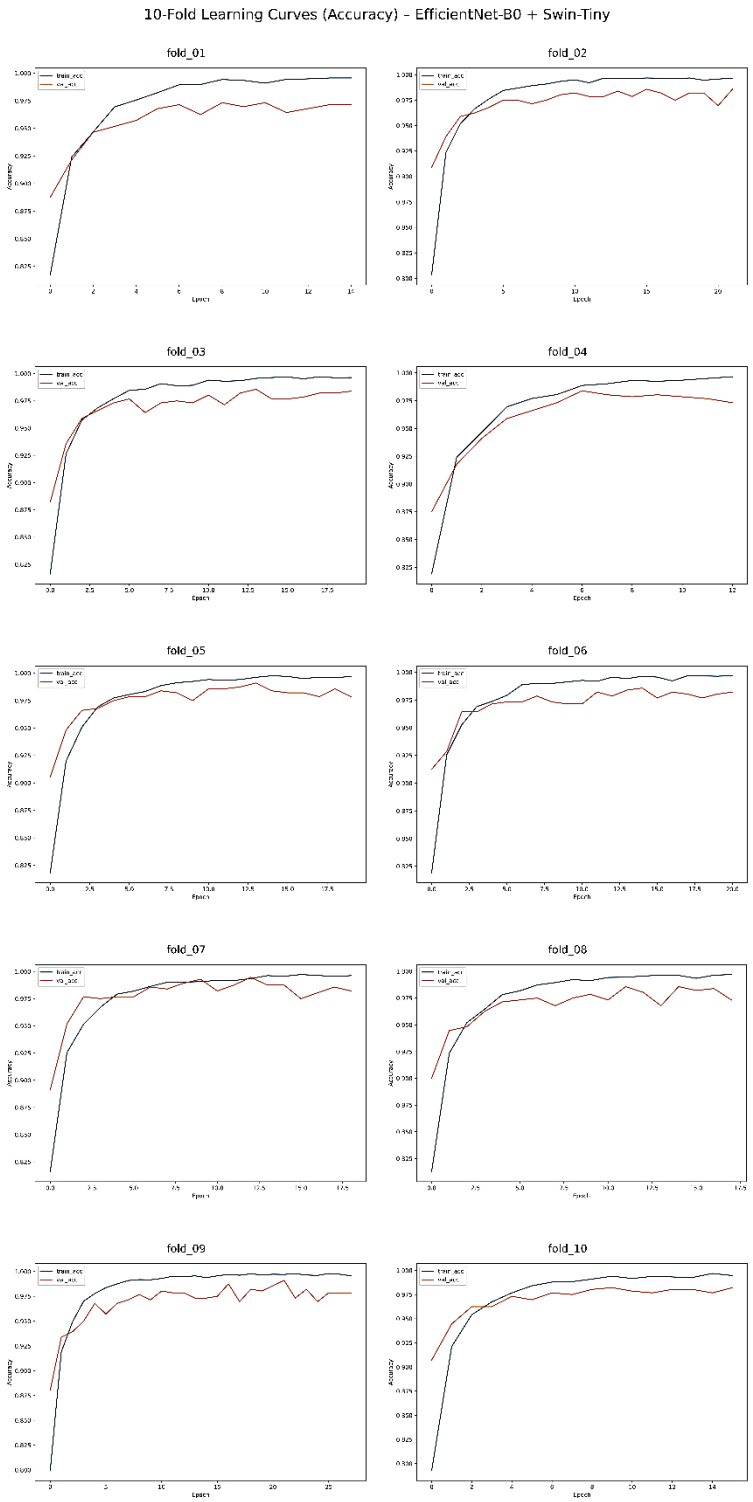

Figure 7: Fold-wise validation accuracy curves across the 10 stratified folds for EfficientNet-B0 + Swin-Tiny

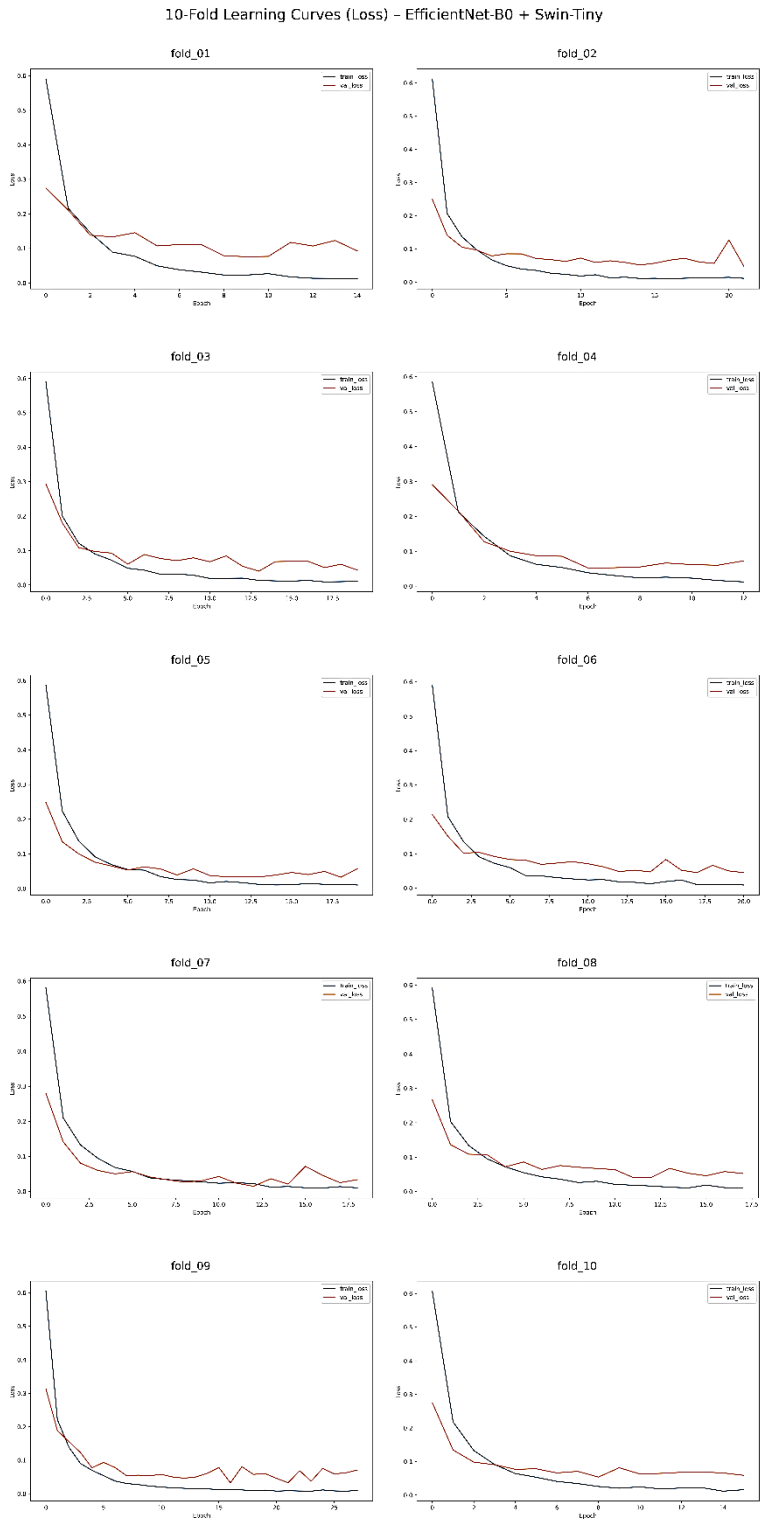

Figure 8: Fold-wise validation loss curves across the 10 stratified folds for EfficientNet-B0 + Swin-Tiny

3.3 Statistical significance testing among top-performing pairs

Paired tests were performed using fold-wise accuracies from identical CV splits. Both paired t-test and Wilcoxon signed-rank test were computed, and Holm correction was applied to paired t-test p-values. The statistical comparison among the top-performing pairs is summarized in Table 7.

Table 7: Statistical comparison among top pairs (paired on the same 10 CV folds).

Test type	Model A (mean)	Model B (mean)	paired t-p	Wilcoxon p	Cohen's d (paired)	Holm-adjusted p
vs_best	Eff-B0 + Swin (0.982857)	ConvNeXt + Swin (0.982143)	0.7527	0.7578	0.1027	0.7527
vs_best	Eff-B0 + Swin (0.982857)	DenseNet + DeiT (0.981071)	0.3330	0.3281	0.3235	0.6660

As shown in Table 7, the performance differences among the top-ranked pairs are small under the tested setting ($p > 0.05$), and therefore computational cost and stability are important secondary selection criteria.

3.4 Ablation studies

3.4.1 Fusion strategy ablation

Fusion strategies were evaluated on the CV-selected pair (EfficientNet-B0 + Swin-Tiny**)** using 10-fold CV. The results are reported in Table 8.

Table 8: Fusion strategy ablation on EfficientNet-B0 + Swin-Tiny (10-fold CV).

Fusion strategy	Mean acc (%)	Std (%)	Mean train time (s)	Mean peak mem (MB)	Mean infer (ms/img)
gated	**98.38**	0.49	430.65	2518.90	0.518
late_concat	98.25	0.66	402.32	2575.80	0.522
attn	98.04	0.68	397.97	**2754.21**	0.525

Gated fusion achieved the highest mean accuracy, while attention fusion increased memory usage without improving accuracy in this configuration.

Table 9: Trainable ratio ablation on EfficientNet-B0 + Swin-Tiny.

Trainable ratio	Mean acc (%)	Std (%)	Mean train time (s)	Mean peak mem (MB)	Mean infer (ms/img)
0.4	**98.59**	0.56	343.14	2778.38	0.517
0.2	98.27	0.62	473.31	3121.97	0.522
0.1	97.66	0.63	427.29	2599.09	0.519
0.0	94.71	0.88	514.95	2630.20	0.518

3.4.2 Selective fine-tuning (trainable ratio) ablation

Sensitivity to the freezing/unfreezing setting was evaluated using trainable ratios of 0.0, 0.1, 0.2, and 0.4. The results are summarized in Table 9.

A clear drop was observed when backbones were fully frozen (0.0), supporting the need for selective fine-tuning in the MRI domain adaptation.

3.5 Final evaluation on Kaggle Testing and external TEST2

After selecting the final architecture, a final training run was performed on the Kaggle Training set with an internal split for early stopping, followed by evaluation on Kaggle Testing and the external dataset (TEST2). Although gated fusion and a trainable ratio of 0.4 achieved slightly higher cross-validation accuracy, late concatenation with a trainable ratio of 0.2 was selected as the final configuration to reduce model complexity, maintain a simpler design, limit overfitting, and preserve competitive performance. The final evaluation results are summarized in Table 10.

Table 10: Final evaluation (EfficientNet-B0 + Swin-Tiny; late_concat; trainable_ratio = 0.2).

Metric	Value
Kaggle Testing accuracy	93.94%
External TEST2 accuracy	87.21%
Best internal validation accuracy	98.21% (epoch 13)
Train time	338.38 s
Peak GPU memory	2499.71 MB
Inference time	0.504 ms/img
Total parameters	32.05 M
Trainable parameters	8.45 M

The gap between Kaggle Testing and TEST2 indicates a dataset shift. It is common in medical imaging due to changes in acquisition protocols, such as differences in scanner characteristics and preprocessing methods. The confusion matrix of the final model on the Kaggle Testing set is shown in Figure 9. The one-vs-rest precision, recall,

and ROC curves are shown in Figure 10 and Figure 11, respectively. Training dynamics (accuracy and loss) for the final run are shown in Figures 12 and 13.

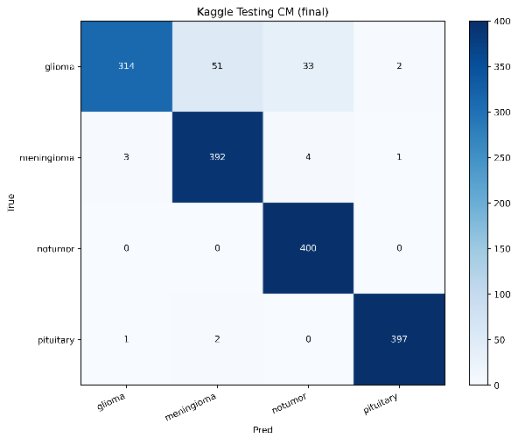

Figure 9: Confusion matrix of the final model (EfficientNet-B0 + Swin-Tiny) on the Kaggle Testing set

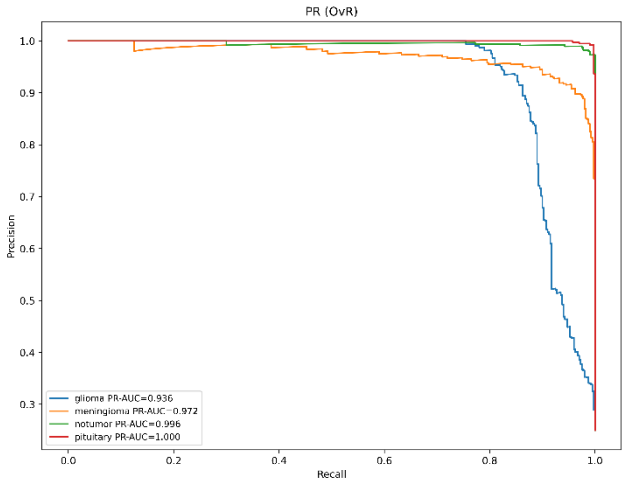

Figure 10: One-vs-rest (OvR) precision–recall curves on the Kaggle Testing set for the final model

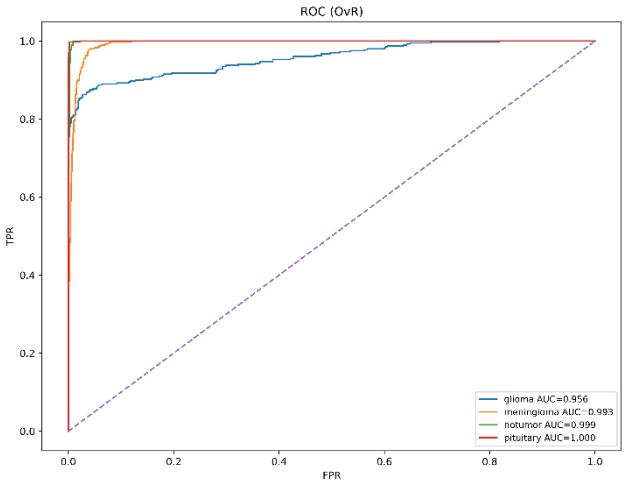

Figure 11: One-vs-rest (OvR) ROC curves on the Kaggle Testing set for the final model

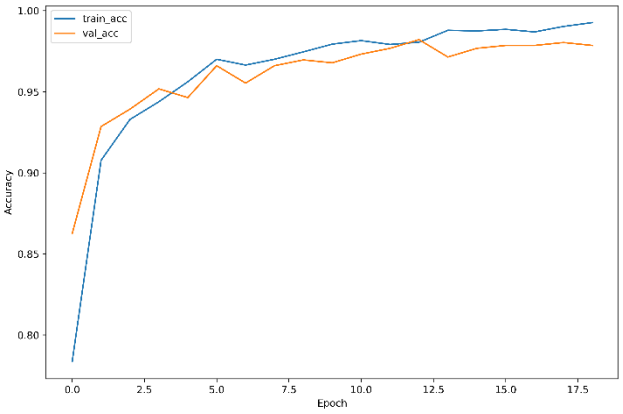

Figure 12: Training and validation ***accuracy*** curves for the final training run

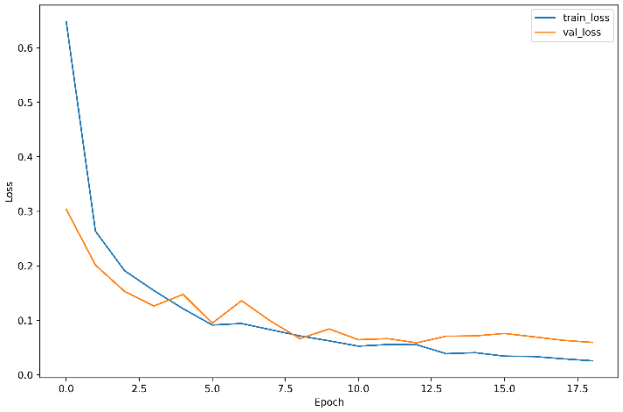

Figure 13: Training and validation ***loss*** curves for the final training run

3.6 Clinical interpretation of misclassified cases:

To understand why these errors occurred, we manually reviewed the misclassified cases together with our neurosurgeon co-supervisor (Dr Md. Ashraful Haque). In real clinical practice, tumor type is finally confirmed after surgery using histopathology and immunohistochemistry. In contrast, our model must make a decision based on a single MRI slice and without any clinical history. Because of this limitation, some images were difficult even for an experienced neurosurgeon to interpret. Several tumors showed very similar radiological appearance, especially between glioma and meningioma, and sometimes between meningioma and pituitary tumors. Based on the one slice alone, these cases can look almost the same. Another major reason was image quality. Some images contained significant noise, motion artefacts, low contrast, or unclear tumor boundaries.

3.7 Discussion

Our model performs well on the Kaggle dataset. It shows high accuracy in both the original split and the cross-validation results. However, the accuracy drops when tested on real hospital data (TEST2). This indicates that Kaggle data alone is not enough to train a model for real-world use.

The reason can be a domain shift. Hospital MRI images are different from Kaggle images. They may have different scanners, protocols, noise levels, contrast settings, and

image quality, since the model was not trained on these real hospital patterns. As a result, it fails when it faces them during real prediction. To make the model suitable for real clinical use, it should be trained on substantial hospital data from multiple sources. If the training data covers almost all real cases and variations, the model will be more likely to learn more reliable tumor features. Also, the model should be retrained regularly with new hospital data to remain robust and generalizable to unseen samples.

3.8 Limitations

This study has several limitations. First, the datasets were collected from publicly available sources, and the ground-truth labels were not clinically re-verified. So, some label noise may still exist. Second, MRI images can vary across scanner vendors, field strengths, and acquisition protocols. These differences can change the image distribution and reduce generalization on new hospitals. Third, patient-level separation cannot be fully guaranteed because patient identifiers are not available in the public datasets.

Finally, our model decides on a single MRI slice. In real clinical practice, neurologists and neurosurgeons review multiple slices (and often multiple MRI sequences) before confirming the tumor type. As a result, single-slice classification can miss important context and may lead to confusion between visually similar tumor types.

4. CONCLUSIONS

This study presented a systematic evaluation of CNN–Transformer hybrid models for MRI-based brain tumor classification. Twelve model pairs were screened under identical conditions, followed by 10-fold cross-validation to obtain a reliable ranking. EfficientNet-B0 + Swin-Tiny showed the best cross-validation accuracy and was selected as the final model. The final configuration was selected not only for peak cross-validation accuracy but also for simplicity, reduced complexity, and robustness against overfitting. The model performed well on the Kaggle dataset but showed a clear drop in accuracy on external hospital data, highlighting a domain shift. Ablation studies and computational cost analysis further supported the design choices. Overall, the proposed hybrid framework demonstrates strong potential, but real-world deployment will require training on a broader range of clinical data.

ACKNOWLEDGMENTS

We sincerely thank Kaggle for providing the open- access brain tumor MRI dataset (Nickparvar, 2021) and the broader open-source community for developing and maintaining powerful pretrained models like Swin Transformer (Liu et al., 2021) and EfficientNet (Tan & Le, 2019). This work was implemented using the PyTorch deep learning framework and the ‘timm‘(PyTorch Image Models) library.

DATA AVAILABILITY STATEMENT

Datasets generated during the current study are available from the corresponding author upon reasonable request.

FUNDING DECLARATION

This research was self-funded.

ETHICS APPROVAL

This study is an engineering experimental investigation. The MIJST Research Ethics Committee has confirmed that formal ethical approval was not required.

ETHICS, CONSENT TO PARTICIPATE, AND CONSENT TO PUBLISH

Not applicable.

COMPETING INTERESTS

The authors declare that they have no competing interests.

AUTHOR CONTRIBUTIONS

Author 1: Writing – original draft, methodology, formal analysis, Supervision, editing, Conceptualization, Resources, Editing, Supervision
Author 2: Data curation, formal analysis
Author 3: Writing—Review & editing
Author 4: Writing-review & editing
Author 5: Resources, editing

ARTIFICIAL INTELLIGENCE ASSISTANCE STATEMENT

Portions of this manuscript were assisted by an artificial intelligence language model (ChatGPT, OpenAI). The tool was used solely for language editing, text refinement, and improving clarity. All content, data interpretation, analysis, conclusions, and final decisions were produced, verified, and approved by the authors. The authors take full responsibility for the accuracy and integrity of the manuscript.

CONFLICT OF INTEREST DECLARATION

The authors declare that they have no conflicts of interest.

REFERENCES

Al Bataineh, A. F., Nahar, K. M. O., Khafajeh, H., Samara, G., Alazaidah, R., Nasayreh, A., Bashkami, A., Gharaibeh, H., & Dawaghreh, W. (2024). Enhanced magnetic resonance imaging-based brain tumor classification with a hybrid Swin Transformer and ResNet50V2 model. *Applied Sciences*, 14(22), 10154. https://doi.org/10.3390/app142210154

Amin, B. M., Elhalawany, M., Sultan, N., & Alwazzan, M. (2023). Brain tumor classification and segmentation using EfficientNet-B1 and U-Net: An approach using a transfer-learning model with contrast enhancement for improved segmentation and classification. *arXiv*. https://doi.org/10.48550/arXiv.2304.10039

Benzorgat, N., Xia, K., & Benzorgat, M. N. E. (2024). Enhancing brain tumor MRI classification with an ensemble of deep learning models and transformer integration. *PeerJ Computer Science*, 10, e2425. https://doi.org/10.7717/peerj-cs.2425

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 248–255). IEEE. https://doi.org/10.1109/CVPR.2009.5206848

Fang, Y., Sun, S., Wang, K., Peng, H., Chen, D., et al. (2023). EVA: Exploring the limits of masked visual representation at scale. *arXiv*. https://doi.org/10.48550/arXiv.2303.11331

Goceri, E. (2025). An efficient network with CNN and transformer blocks for glioma grading and brain tumor classification from MRIs. *Expert Systems with Applications*, 268, 126290. https://doi.org/10.1016/j.eswa.2024.126290

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings *of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778). IEEE.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In Proceedings *of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2261–2269). IEEE.

Khan, M. A., & Park, H. (2024). A convolutional block base architecture for multiclass brain tumor detection using magnetic resonance imaging. Electronics, 13(2), 364. https://doi.org/10.3390/electronics13020364

Krishnan, P. T., Krishnadoss, P., Khandelwal, M., Gupta, D., Nihaal, A., & Kumar, T. S. (2024). Enhancing brain tumor detection in MRI with a rotation invariant Vision Transformer (RViT). *Frontiers in Neuroinformatics*, 18, 1413446. https://doi.org/10.3389/fninf.2024.1413446

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., & Guo, B. (2021). Swin Transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF *International Conference on Computer Vision (ICCV)* (pp. 10012–10022). https://doi.org/10.1109/ICCV48922.2021.00986

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022). A ConvNet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 11976–11986). IEEE.

Nickparvar, M. (2021). Brain tumor MRI dataset [Data set]. *Kaggle*. https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset

Pacal, I. (2024). A novel Swin transformer approach utilizing residual multi-layer perceptron for diagnosing brain tumors in MRI images. *International Journal of Machine Learning and Cybernetics*, 15, 3579–3597. https://doi.org/10.1007/s13042-024-02074-z

Panigrahi, S., Adhikary, D. R. D., & Pattanayak, B. K. (2025). Hybrid transfer learning and self-attention framework for robust MRI-based brain tumor classification. *Scientific Reports*, 15, 21343. https://doi.org/10.1038/s41598-025-09311-5

Sharma, V. K., & Ameta, G. K. (2025). Hybrid 3D CNN–transformer model for early brain tumor detection with multi-modal magnetic resonance imaging. *Bulletin of Electrical Engineering and Informatics*, 14(5), 3923–3934. https://doi.org/10.11591/eei.v14i5.11082

Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning* (pp. 6105–6114). PMLR. https://proceedings.mlr.press/v97/tan19a.html

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., & Jégou, H. (2021). Training data-efficient image transformers & distillation through attention. In *Proceedings of the 38th International Conference on Machine Learning* (pp. 10347–10357). PMLR.

Wang, S., Wang, C., Shao, Y., Chen, B., & Xu, Y. (2025). Enhanced MRI brain tumor detection using deep learning and radiomics features: Implementation details using real datasets. *Scientific Reports*, 15, 21717. https://doi.org/10.1038/s41598-025-13115-y

Zahoor, M. M., & Khan, S. H. (2025). CE-RS-SBCIT: A novel channel enhanced hybrid CNN–Transformer with residual, spatial, and boundary-aware learning for brain tumor MRI analysis. *arXiv*. https://doi.org/10.48550/arXiv.2508.17128