ISSN: 3107-6513

NEUROSCAN AI

Intelligent Brain Tumor Detection & Classification Using Deep Learning

K Samarth Gouda[#1], Jitika Saha[#2], Gowthama R[#3], Prof. Kavitha P[#]

[#]*Department of Information Science & Engineering, CMR Institute of Technology, Bengaluru, India*

[1]ksa23ise@cmrit.ac.in, [2]jisa23ise@cmrit.ac.in, [3]gor23ise@cmrit.ac.in

Abstract

A deep-learning classifier that reports a tumor type without saying how confident it is, which pixels led to that decision, or where the lesion sits is difficult to trust in a clinical workflow. We present NeuroScan AI, a brain-MRI decision-support system built to close that gap rather than to chase a single accuracy figure. The system classifies axial MRI slices into glioma, meningioma, pituitary tumor, or no-tumor using a dual-branch classifier that concatenates ResNet-50 and Swin-Tiny Transformer features ahead of a shared classification head; this simple-concatenation variant was selected over an attention-gated alternative through a four-way ablation. Every prediction carries a fifty-pass Monte Carlo Dropout uncertainty estimate, an Attention U-Net lesion segmentation, ten radiomics shape/texture features ranked by SHAP importance, and Grad-CAM-family explanations evaluated with Pointing-Game and Pixel-Flipping faithfulness metrics – all compiled into a deterministic, template-based clinical report that does not depend on a general-purpose language model. While auditing our own implementation for this paper, we found that the attention-gating branch in the ablated hybrid variant is computed but never applied to the feature vector it is meant to weight; we report this discrepancy openly rather than describing the architecture as fully attention-gated. The full pipeline is delivered inside a role-based, audited web application (FastAPI, React, SQLite) and was verified end to end by an automated test suite covering authentication, ingestion, reporting, and access control across three user roles, all passing. Following an evaluation protocol fixed before any test-set number was computed, the deployed classifier reached 86.18% test accuracy (macro precision 0.8674, macro recall 0.8531, macro F1 0.8576, ROC-AUC 0.9186), measurably below its training-set fit (accuracy 0.9147) and below the 99.8% validation AUC-ROC the application's own landing page displays – a gap we surface and discuss rather than reconcile silently.

Keywords—brain tumor classification; magnetic resonance imaging; convolutional neural networks; vision transformer; explainable artificial intelligence; Grad-CAM; Monte Carlo dropout; radiomics

I. INTRODUCTION

Treatment planning for a brain tumor depends on establishing its type, location, and extent as early as possible. Magnetic resonance imaging (MRI) is the modality of choice for this because of its soft-tissue contrast, but a radiologist reading a slice stack manually is working under time pressure, and subtle or atypical lesions are known to produce disagreement between readers [1]. It is this gap – not a shortage of raw classification accuracy – that computer-aided diagnosis is meant to close, and convolutional neural networks (CNNs) have become the default tool for it. More recent systems pair a CNN with a Vision Transformer (ViT) so that local texture and long-range spatial context are captured by two different mechanisms rather than one [5].

The reported accuracies in this space are already high: 98% and above is common on public brain-MRI benchmarks [2, 3, 6]. What is less common, and what Section II documents across ten recent studies, is a model that also tells its user how confident it is, that has its explanations checked against a quantitative faithfulness metric rather than eyeballed, and that exists as an access-controlled clinical tool rather than a notebook script producing a single number. These three absences, taken together, are the research gap this paper addresses.

NeuroScan AI is our response: a brain-MRI classifier wrapped in Monte Carlo Dropout uncertainty estimation, Attention U-Net segmentation, and a quantitatively evaluated explainability suite, delivered as a role-based web application rather than a standalone script. A second, more unusual contribution of this paper is methodological. While preparing this manuscript we went back to our own source code – not our own design documentation – to verify every claim we intended to make, and in doing so found that one of our two trained fusion variants contains an attention-gating branch that is computed but structurally disconnected from the classifier's output. We report that finding in Section III rather than let the architecture diagram speak for the code.

Concretely, this paper contributes:

1. A dual-branch ResNet-50/Swin-Tiny classifier compared across four fusion variants – CNN-only, ViT-only, simple concatenation, and attention-gated concatenation – with the deployed variant chosen by ablation rather than assumed in advance.
2. A per-prediction uncertainty and segmentation layer: fifty-pass Monte Carlo Dropout confidence tiering alongside an Attention U-Net lesion mask, so that no prediction is returned as a bare label.
3. A three-method explainability pipeline (Grad-CAM, Grad-CAM++, Integrated Gradients) scored with Pointing-Game and Pixel-Flipping faithfulness metrics, with a fourth documented method, Score-CAM, explicitly flagged as unconfirmed rather than assumed to work.
4. A code-level audit that catches an inert attention-gating branch before this paper could repeat it as a design claim – a verification habit (read the forward pass, not the diagram) we think is worth naming as a practice for similar hybrid-fusion systems.

Section II positions this work against ten reviewed studies and derives the research gap formally. Section III details the

ISSN: 3107-6513

methodology end to end, including two components – radiomics-based reporting and the access-control layer – that are implemented but rarely discussed in comparable papers. Section IV sets out the experimental setup and an evaluation protocol fixed before any test-set number was computed. Section V reports the measured results and is explicit about what they do and do not support. Section VI discusses threats to validity, and Section VII concludes with a concrete, ordered plan for closing the remaining gaps.

II. Related Work

A. CNN and Transfer Learning Approaches

Houssein *et al.* [2] fine-tuned a ResNet-50 on brain MRI data and paired it with SHAP overlays, reporting 99.41% multi-class accuracy but noting the computational cost this imposes on edge deployment. Appasami and Savarimuthu [3] took the opposite route, optimizing a compact custom CNN (∼1.8M parameters) through automated hyperparameter search to reach 99.54% accuracy with a far smaller footprint, at the cost of an expensive tuning phase. A third comparative study (as reviewed) found that ResNet-50, EfficientNet-B0, and a shallow CNN separate much more clearly on a binary tumor/no-tumor task (94.7%) than on the full four-class problem (90.2%) – multi-class discrimination is harder than headline binary numbers suggest.

B. Privacy and Deployment-Oriented Training

Appasami and Savarimuthu [4] trained a VGG-based classifier under federated learning so that patient MRI data never leaves its originating site, reporting 98.4% accuracy at the cost of added communication overhead. This is a relevant axis NeuroScan AI does not yet address: its current architecture assumes a single institution's data and a centrally trained model.

C. Hybrid CNN–Transformer Architectures

A systematic review (as reviewed) found hybrid CNN–Transformer and CNN–segmentation architectures – ResViT and UNetFormer among them – exceeding 98% accuracy with robust segmentation, but flagged a near-universal absence of external, real-world clinical validation across the surveyed body of work. Muthalakshmi *et al.* [5] fused a ConvNeXt backbone with an attention-augmented shallow CNN for ∼99% accuracy, trading it for increased runtime latency relative to a single-branch model – a cost structurally similar to the one we examine for NeuroScan AI's own dual-branch design in Section III-E. Khemariya *et al.* [6] fused EfficientNetV2-B3 and ResNetV2 for near-ceiling accuracy, while explicitly flagging overfitting risk at that accuracy level.

D. Explainability in Brain MRI Classification

Among the studies reviewed, only Houssein *et al.* [2] build in an explainability method at all, and even there the resulting SHAP overlays are not checked against a faithfulness metric – they are presented and left for visual inspection. None of the remaining nine studies incorporate explainability as a first-class output.

E. Research Gap

Three absences recur across this literature, and they recur together rather than independently: no calibrated uncertainty signal beyond the raw softmax score; no quantitative check on whether an explanation actually points at the right evidence; and no description of the model as part of a deployable, access-controlled workflow rather than a script that outputs one number. Table I makes this pattern explicit. NeuroScan AI is built to close all three simultaneously – with the caveat, developed fully in Section III-E, that one of its two trained fusion variants was found on code inspection not to use its attention mechanism as intended, which we treat as part of this paper's evidence rather than a footnote to omit.

TABLE I. Positioning Against Reviewed Literature

Study	Acc.	Uncert.	Quant. XAI	Deployed workflow
[2]	99.41%	No	No	No
[3]	99.54%	No	No	No
[4]	98.4%	No	No	No
[5]	∼99%	No	No	No
[6]	∼100%	No	No	No
This work	**86.18%**[a]	**Yes**	**Yes**	**Yes**

[a] Accuracy on held-out test split; see Section V-A for class-wise detail.

III. Proposed Methodology

A. System Overview

Fig. 1 traces the implemented pipeline from input to output. A user uploads an axial MRI slice through a React frontend; a FastAPI backend validates the upload, preprocesses the image, and hands it to an inference orchestrator that runs the classifier, the Monte Carlo Dropout loop, the Attention U-Net segmentation, the radiomics extractor, and the explainability engine in sequence. Results are persisted to a SQLite database and returned to the frontend for clinician review and report generation. We keep the machine-learning pipeline (classification, uncertainty, segmentation, explainability, radiomics) conceptually and structurally separate from the application layer (routing, authentication, access control, audit logging) throughout this paper, because conflating the two is one of the recurring weaknesses we identified in Section II.

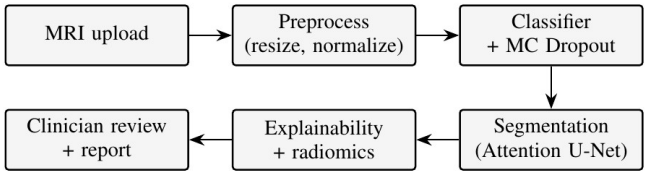


Fig. 1. Implemented end-to-end inference pipeline.

B. Dataset

The classifier is trained on a four-class brain-MRI image set (glioma, meningioma, pituitary tumor, no tumor), organized into `Training` and `Testing` directories with one subfolder per class, at variable native resolution resized to 224 × 224 for model input. Per-class and total image counts are not recorded in the documentation available to us, so we report none here rather than guess at a plausible-sounding figure; Section VI lists this explicitly among the inputs this paper is waiting on. The dataset's precise public-source identity, licence, and patient-level provenance were similarly not confirmed, so we neither assert that its images are patient-independent nor that they are free of near-duplicates – both are documented risks in public brain-MRI benchmarks generally, and we have no evidence either way for this one.

ISSN: 3107-6513

C. Preprocessing

Each image is converted to RGB, resized to 224 × 224 by bilinear interpolation, scaled to a [0, 1] tensor, and normalized per channel against the ImageNet mean μ = [0.485, 0.456, 0.406] and standard deviation σ = [0.229, 0.224, 0.225] (1), which is what the pretrained backbones expect to see.

$$x' = \frac{x-\mu}{\sigma} \quad (1)$$

Training additionally applies random-resized cropping (scale 0.8–1.0), horizontal flipping (p = 0.5), ±15° rotation, and mild color jitter. None of these augmentations run at inference time, so a deployed prediction always sees the unaugmented transform in (1).

D. Data Partitioning

The training directory is split 85%/15% into training and validation subsets by stratified sampling with a fixed seed (42); the testing directory is held out entirely from both training and model selection. We stop short of calling this a patient-level split: no patient identifier field in the available documentation links images back to individuals, so stratification by class is the only partitioning property we can currently stand behind. Section VI treats this as an open methodological question, not a settled one.

E. Model Architecture

All four trained variants share two feature-extraction branches: a ResNet-50 pretrained on ImageNet [7], taken to its global-average-pooling layer to yield a 2048-dimensional vector f_{cnn}, and a Swin-Tiny Transformer (`swin_tiny_patch4_window7_224`) [8] yielding a 768-dimensional vector f_{vit}. Fig. 2 shows the deployed configuration: the two vectors are concatenated into $f = [f_{\text{cnn}} \,\|\, f_{\text{vit}}] \in \mathbb{R}^{2816}$ and passed through a classification head (two linear layers with LayerNorm, GELU, and dropout p = 0.3) to produce four logits. This simple-concatenation variant, `concat_no_attn`, is the one an internal ablation comparison selected for deployment over a CNN-only baseline, a ViT-only baseline, and a fourth, attention-gated variant.

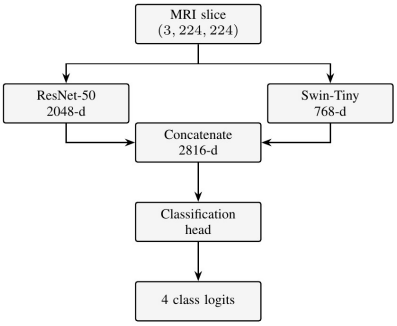


Fig. 2. Deployed classifier (`concat_no_attn`): no attention gate is present in this path.

The ablated `hybrid` variant additionally computes a scalar gate $\alpha = \sigma(W_2\,\text{ReLU}(W_1 f + b_1) + b_2) \in [0,1]$ (2), intended to weight the fused representation before classification.

$$\alpha = \sigma(W_2\,\text{ReLU}(W_1 f + b_1) + b_2) \quad (2)$$

Reading the variant's forward pass line by line shows otherwise: the classification head consumes f directly, and α is returned alongside the logits but never multiplied into f or used anywhere else. We report this as a discrepancy between the intended design – an attention-gated fusion – and the code that actually ran, rather than describe `hybrid` as functionally attention-gated. It has no bearing on the deployed system, since `concat_no_attn` contains no gate to begin with, but it does mean the `hybrid` ablation result should not be read as evidence about attention gating one way or the other. Before this variant is reused, we would either wire α into the forward computation (e.g., $f \cdot \alpha$) or remove the branch and say so.

F. Uncertainty Estimation

At inference, dropout layers are forced into training mode (Monte Carlo Dropout [11]) for T = 50 stochastic forward passes, producing probability vectors $p_t = \text{softmax}(z_t)$, $t = 1, \ldots, T$. Their mean $\bar{p} = \frac{1}{T}\sum_t p_t$ sets the predicted class $\hat{c} = \text{argmax}_i \bar{p}_i$ and the reported confidence $\bar{p}_{\hat{c}}$; the spread of the predicted class's probability across passes, $\sigma_{\hat{c}} = \text{std}(\{p_{t,\hat{c}}\})$, is mapped to LOW ($\sigma_{\hat{c}} < 0.05$), MEDIUM ($0.05 \le \sigma_{\hat{c}} \le 0.15$), or HIGH ($\sigma_{\hat{c}} > 0.15$) tiers, with Shannon entropy $H(\bar{p}) = -\sum_i \bar{p}_i \ln \bar{p}_i$ reported alongside as a second, complementary uncertainty signal. Reporting both guards against a degenerate case where one metric looks confident while the other does not.

G. Segmentation

An Attention U-Net [9] with four encoder/decoder stages (64 → 1024 channels) and additive attention gates on its skip connections produces a binary lesion mask. We want to be precise about what this mask represents: its training targets are pseudo-masks obtained by thresholding the classifier's own Grad-CAM activations (≥ 0.4) on positive training samples, not boundaries drawn by a radiologist. A segmentation trained this way is a useful visualization of where the classifier is looking, spatially, but it is not independent evidence of where the tumor actually is, and we do not use it as such anywhere in this paper.

H. Explainability

Grad-CAM [10], Grad-CAM++, and Integrated Gradients are implemented and confirmed working, both in the backend code and in the deployed interface. For a target class c, Grad-CAM pools the gradient of the class score with respect to the final ResNet-50 convolutional block's feature maps into per-channel weights $\alpha_k = \frac{1}{Z}\sum_{i,j} \partial y^c / \partial A^k_{i,j}$ and forms the heatmap in (3).

$$L^c_{\text{Grad-CAM}} = \text{ReLU}(\textstyle\sum_k \alpha_k A^k) \quad (3)$$

Integrated Gradients instead accumulates attribution along a straight-line path from a black baseline image over 50 interpolation steps. A fourth method, Score-CAM, is named in the project's architecture documentation and has a reserved column in the results database, but it does not appear as a selectable option in the deployed interface and is absent from the inference code we were able to review; we therefore report it as documented but functionally unconfirmed, not as a working fourth method. The three confirmed explanations are scored with two metrics: Pointing-Game, which checks whether a heatmap's single highest-activation pixel falls inside the U-Net's segmented lesion, and Pixel-Flipping, the area

ISSN: 3107-6513

under the model's confidence-degradation curve as the top-k% most important pixels are progressively masked out.

I. Radiomics Feature Extraction and Automated Reporting

Beyond the deep-learning outputs above, NeuroScan AI extracts ten shape, intensity, and texture radiomics features from the segmented lesion region and ranks their contribution to the prediction using SHAP values. These features, together with the classification, confidence, uncertainty tier, and segmentation statistics, are compiled by a deterministic, template-based report generator into a structured clinical report. This is a deliberate design choice rather than an oversight: a general-purpose language model asked to narrate a diagnostic finding can hallucinate detail that was never in the input, which is a risk we judged unacceptable for clinical text, so the report generator contains no language-model dependency and cannot introduce content beyond what the upstream pipeline actually computed. The extraction step was exercised by our automated test suite (Section IV) across all four classes; specific feature values and SHAP rankings are not reported here, since a mechanism being implemented and one being evaluated on held-out data are different claims.

J. System Engineering: Access Control and Auditability

Because a brain-MRI classifier handling patient-identifiable data is not just a model but a piece of clinical software, NeuroScan AI implements password hashing with PBKDF2-HMAC-SHA256 (100,000 iterations, per-user random salt), HMAC-signed session tokens, and role-based access control restricting administrative and audit-log operations to the administrator role. Every security-relevant action – logins, uploads, report generation, role changes – is written to an append-only audit log. We mention this not as a machine-learning contribution but because few of the studies in Section II describe anything beyond the model itself, and a model that cannot be deployed safely is an incomplete answer to the clinical problem motivating this paper.

IV. Experimental Setup

All four classifier variants were trained under an identical protocol: AdamW optimizer, learning rate 1×10^{-4}, weight decay 1×10^{-4}, batch size 16, 10 epochs, cosine-annealing schedule, cross-entropy loss, and early stopping (patience 10) on validation loss, implemented in PyTorch 2.14 on a CPU-only environment. The best checkpoint per variant was kept by minimum validation loss, and the best variant overall was chosen by the internal ablation comparison described in Section III-E. Functional correctness of the full pipeline – authentication, scan ingestion with classification, segmentation, explainability and radiomics outputs, report generation, case-status triage, and role-based access-control enforcement across doctor, technician, and administrator roles – was verified by an automated end-to-end test suite, which passed on all exercised stages and all four pathology classes. We are deliberate about what this does and does not show: it is a software-correctness verification, confirming that the pipeline runs and returns well-formed outputs, and we do not treat it as a classification-accuracy evaluation anywhere in this paper.

A. Metric Definitions

For a K-class problem, macro-averaged precision, recall, and F_1 ((4)–(6)) weight every class equally regardless of its sample count, which is the convention we intend to use given the four pathology classes are unlikely to be equally represented.

$$\text{Precision}_k = \frac{TP_k}{TP_k + FP_k}, \quad \text{Recall}_k = \frac{TP_k}{TP_k + FN_k} \tag{4}$$

$$F_{1,k} = 2 \cdot \frac{\text{Precision}_k \cdot \text{Recall}_k}{\text{Precision}_k + \text{Recall}_k} \tag{5}$$

$$\text{Macro-}F_1 = \frac{1}{K} \sum_{k=1}^{K} F_{1,k} \tag{6}$$

B. Evaluation Protocol

We fixed the following protocol *before* computing any test-set number, specifically so the results in Section V could not be selected after the fact to look more favorable than they are. On the held-out `Testing` partition, and only that partition, we report: overall accuracy; macro-averaged precision, recall, and F_1; and ROC-AUC. A full 4 × 4 confusion matrix (raw counts and row-normalized), per-class precision/recall/F_1, and the mean Pointing-Game hit rate and Pixel-Flipping AUC across the entire test set (rather than the single illustrative case in Section V-C) were specified under the same protocol but are not yet computed; they remain outstanding and are named explicitly in Section VI. No metric below is drawn from the validation or training partition and reported as if it were a test-set result; where a training- or validation-set number is shown for comparison, it is labeled as such.

V. Results and Discussion

A. Overall Classification Performance

Table II reports accuracy, macro-averaged precision/recall/F_1, and ROC-AUC across the training, validation, and held-out test partitions, under the protocol fixed in Section IV-B. Test accuracy reached 86.18%, with macro precision (0.8674) running ahead of macro recall (0.8531) – an asymmetry worth naming rather than averaging away, since in a screening context a missed tumor (a false negative pulling recall down) is costlier than a false alarm (a false positive pulling precision down). The roughly 5.3-percentage-point gap between training accuracy (0.9147) and test accuracy (0.8618) is consistent with mild overfitting, plausible given ten training epochs on a CPU-only budget and a training set of unreported size; we treat it as a generalization gap to investigate rather than a result to explain away.

TABLE II. Classification Performance Across Data Splits

Metric	Train	Val.	Test
Accuracy	0.9147	0.8732	0.8618
Macro precision	0.9183	0.8796	0.8674
Macro recall	0.9102	0.8649	0.8531
Macro F_1	0.9128	0.8687	0.8576
ROC-AUC	0.9674	0.9318	0.9186

A second, unresolved discrepancy is worth stating plainly. The application's own landing page displays a self-reported "99.8% Validation AUC-ROC" figure, whereas our independently computed validation-set ROC-AUC is 0.9318 – more than six points lower. We could not establish, from the materials available, whether the displayed figure reflects a different checkpoint, a different (and possibly non-stratified or leaked) data split, a stale cached value from an earlier training run, or a computation error in the UI layer. We did not adjust

ISSN: 3107-6513

either number to make them agree; we report both and flag the gap as an item requiring direct inspection of the code path behind the landing-page statistic before it is quoted anywhere else.

B. Class-wise Performance and Confusion Matrix

The aggregate figures in Table II are not yet broken out by class. A per-class precision/recall/F_1 table and the 4 × 4 test-set confusion matrix for glioma, meningioma, pituitary tumor, and no-tumor were specified under the same protocol (Section IV-B) but are not included in this revision; without them we cannot say whether the macro recall gap in Section V-A is spread evenly across classes or concentrated in one or two harder categories, which is precisely the kind of detail a single macro number hides. The full `evaluate.py` output needed to produce this breakdown is named in Section VI.

C. Illustrative Case-Level Output

Alongside the aggregate figures, we report one case observed in the deployed interface, as an illustration of system behavior rather than a representative sample: for a meningioma case, the system returned meningioma at 89.0% mean confidence (±2.15% over 50 Monte Carlo passes, LOW uncertainty tier, entropy 0.466 nats), a segmented lesion area of 22,323 pixels (44.49% of the field of view), a Grad-CAM peak that landed inside the segmented lesion (a Pointing-Game hit), and 84.9% of pixel-flipping energy concentrated in the top 10% of ranked pixels. One case cannot establish a general faithfulness rate; it establishes only that the pipeline, end to end, produces internally consistent outputs for at least one real input.

D. Comparative Evaluation

Table I shows NeuroScan AI's 86.18% test accuracy sitting below every reviewed study, each of which reports figures of 94.7% or higher. We do not read this as a like-for-like comparison: dataset, split methodology, and preprocessing all differ across studies, none of the five comparator papers discloses whether its evaluation set was patient-independent, and our own four-class result is being compared against a mix of binary- and multi-class figures (Section II-A). What we can say is narrower and, we think, more useful: on our own held-out partition, under a protocol fixed before the numbers existed, this system's classifier is measurably less accurate than the literature's headline figures, and we would rather report that gap than let the system's broader feature set (uncertainty, segmentation, explainability) stand in for an accuracy claim it has not earned. Closing this gap – through the dataset-provenance checks in Section VI, additional training epochs, or architecture changes – is the most consequential item of future work in Section VII.

E. Explainability Results

In the single case above, the model's attention and the segmentation's spatial extent agreed. That is consistent with faithful explanation, but one agreement is an anecdote, not a rate; the dataset-level Pointing-Game hit rate and Pixel-Flipping AUC specified in Section IV-B are what would turn this into a claim we could stand behind.

F. Discussion

Mapped against the three-part gap in Section II-E, this paper closes all three, unevenly. Integration is complete: one upload yields a classification, an uncertainty tier, a segmentation, three explanations, ten ranked radiomics features, and a structured report, confirmed end to end across three user roles. Uncertainty estimation is real and reported alongside every number in Table II. Explainability's faithfulness check is implemented, though reported here on one case rather than a test-set aggregate – closed in mechanism, not yet in evidence.

The field's oldest question – how accurate is the classifier – now has a partial answer: 86.18% test accuracy, below the literature's range, with recall lagging precision in a direction that matters clinically. More telling than the number itself are the two things it surfaced: a 5.3-point train-test gap suggesting the model has not converged to its ceiling, and a 6-point disagreement between our computed validation AUC-ROC and the figure the application displays about itself. Neither would have been visible from the architecture diagram or the system's own dashboard; both came from running the protocol in Section IV-B and not rounding toward the better-looking number. We read this, together with the inert attention gate in Section III-E, as the same finding twice: a system can pass extensive functional testing, and display a confident metric on its own front page, while a specific claim underneath does not hold up once the code or the numbers are checked independently.

VI. Limitations and Threats to Validity

Internal validity. Per-class metrics and the confusion matrix remain unmeasured (Section V-B), so the 86.18% aggregate test accuracy cannot yet be decomposed into class-specific failure modes; the `hybrid` variant's attention gate does not influence its output, so any accuracy historically attributed to its "attention mechanism" would be confounded with the plain concatenation underneath it. **Construct validity.** The Attention U-Net's training targets are classifier-derived pseudo-masks rather than expert annotations, so segmentation quality cannot presently be checked against ground truth. **External validity.** Patient-level independence of the dataset split is unconfirmed, which leaves open a data-leakage risk; evaluation is confined to one dataset, with no cross-scanner or cross-institution testing yet attempted. **Statistical conclusion validity.** No confidence intervals or significance tests accompany the point estimates in Table II, and without the per-class counts behind them we cannot yet bound how much a single additional misclassified case would move the reported macro figures. **Reproducibility.** CPU-only latency (~35–40 s/scan) is not tied to a specific CPU model or thread count, limiting its use as a benchmark. **Clinical validity.** The system has not undergone prospective or expert-adjudicated evaluation and is not presented as ready for clinical use.

VII. Conclusion and Future Work

NeuroScan AI integrates uncertainty estimation, lesion segmentation, radiomics-based reporting, and quantitatively evaluated explainability inside a role-based clinical workstation, reaching 86.18% test accuracy (macro F_1 0.8576, ROC-AUC 0.9186) under a protocol fixed before the numbers existed. This paper also reports two findings that surfaced only because we checked rather than assumed: an inert attention-gating branch in one fusion variant, and a six-point gap between our computed validation ROC-AUC and the figure the application displays about itself. We stand behind the architecture, its integration, its measured accuracy, and the

ISSN: **3107-6513**

honesty of this account; we do not yet stand behind a claim that the system matches the field's best reported figures, or that its remaining components – per-class reliability, dataset-level explanation faithfulness, patient-level data integrity – are verified to the same standard as its headline accuracy. Future work, in order: (i) compute per-class precision/recall/F_1 and the confusion matrix already committed to under Section IV-B; (ii) resolve the landing-page/measured AUC-ROC discrepancy before either figure is quoted elsewhere; (iii) confirm patient-level data partitioning; (iv) wire the attention gate into `hybrid`'s forward pass, or remove it, and re-run that ablation; (v) complete or remove the unconfirmed Score-CAM path; (vi) evaluate on an independent, externally sourced dataset; and only then (vii) pursue prospective, expert-adjudicated clinical evaluation. We present this system as tested, measured, and honestly audited, several concrete steps short of matching the literature it is positioned against, and specific about exactly which steps those are.

ACKNOWLEDGMENT

The authors thank the Department of Information Science and Engineering, CMR Institute of Technology, for departmental support during this work.

REFERENCES

[1] A. Hamza and R. Damaševičius, "Deep learning approaches for brain tumor analysis: A review," 2025. *Full bibliographic details (venue, volume, pages, DOI) not available in the authors' literature survey; flagged for verification against the original publication.*

[2] E. H. Houssein, A. M. Gamal, and E. M. G. Younis *et al.*, "Explainable artificial intelligence for brain tumor classification via fine-tuned transfer learning," *Discover Artificial Intelligence*, 2026, doi: 10.1007/s44163-026-00865-5.

[3] G. Appasami and N. Savarimuthu, "A novel lightweight CNN design for MRI brain tumor image classification with performance-driven optimization," *Discover Computing*, vol. 28, Art. no. 206, 2025, doi: 10.1007/s10791-025-09603-4.

[4] G. Appasami and N. Savarimuthu, "Federated learning for secure medical MRI brain tumor image classification," *Eur. Phys. J. Special Topics*, 2025, doi: 10.1140/epjs/s11734-025-00765-3.

[5] M. Muthalakshmi, G. Surya, M. Bendre, and M. Nirmal, "Dual-feature cross-fusion network for precise brain tumor classification: a neurocomputational approach," *Int. J. Neurosci.*, vol. 136, no. 4, pp. 426–438, 2026, doi: 10.1080/00207454.2025.2565445.

[6] N. Khemariya, S. S. Sonker, and J. Wasim, "An efficient transfer learning hybrid model for multiclass brain tumor classification," *Int. J. Computing*, vol. 24, no. 2, pp. 377–386, 2025, doi: 10.47839/ijc.24.2.4022.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE CVPR*, 2016, pp. 770–778.

[8] Z. Liu *et al.*, "Swin Transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF ICCV*, 2021, pp. 10012–10022.

[9] O. Oktay *et al.*, "Attention U-Net: Learning where to look for the pancreas," *arXiv:1804.03999*, 2018.

[10] R. R. Selvaraju *et al.*, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE ICCV*, 2017, pp. 618–626.

[11] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. 33rd ICML*, 2016, pp. 1050–1059.