SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Input-Level Multi-Branch Feature-Fusion CNN (ABCNN) Optimized by Hybrid Harris Hawks–Whale Algorithm for Brain Tumor Classification from MRI

Kavita Ghuge[1*], Sandeep Musale[2], Supriya Mangale[3]

[1*]Department of Electronics and Telecommunication Engineering, MKSSS Cummins College of Engineering for Women, Maharashtra, India. E-mail: kavitamunde02@gmail.com
[2,3]Department of Electronics and Telecommunication Engineering, MKSSS Cummins College of Engineering for Women, Maharashtra, India.

ABSTRACT
The successful categorization of brain tumors through MRIs prior to starting neurologic treatment is very important, but challenging due to heterogeneity among tumor classes and unavailability of labeled datasets. In this paper, a novel approach is proposed which incorporates fusion of parallel convolutional, max-pooling, and average-pooling branches during the input phase – kept as ABCNN for the sake of consistency with the used structure, tables, and figures although this block is a pre-designed multi-branch fusion module instead of a learned one, and Hybrid Harris Hawks-Whale Optimization (HHWO) for automatic hyperparameter optimization (ABCNN+HHWO). The HHWO approach uses both exploitation and exploration capabilities of the Harris Hawks Optimization (HHO) and Whale Optimization Algorithm (WOA) algorithms together with adaptive escape energy and Lévy flight for optimizing learning rate, weight decay, dropout rate, and dense layer size of the ABCNN. The framework was tested on the BraTS 2021 (binary classification of HGG/LGG gliomas) and Sartaj (three-class glioma/meningioma/no-tumor classification) datasets by following a TRAIN / HHWO validation / TEST procedure where the TEST subset is withheld from the HHWO search procedure and tested only once. Using the TEST subset, ABCNN+HHWO attained an accuracy of 97.69% on BraTS 2021 and 98.26% on Sartaj, while the baseline ABCNN reached 94.66% and 95.05%, respectively (McNemar's test: BraTS $p < 0.0001$, Sartaj $p = 0.0003$); note that the other baselines CNN, VGG19+TL, and CNN-SVM were evaluated under another procedure and are shown here only for comparison purposes. The critical issue is that BraTS data split is performed at the image level, not at the patient level: patient IDs are recorded, but no check is performed when splitting data, hence almost all patients are present in more than one subset, and the 97.69% value on BraTS cannot be considered a generalization result.

Keywords: brain tumor classification; multi-branch feature fusion; ABCNN; Harris Hawks Optimization; Whale Optimization Algorithm; HHWO; hyperparameter optimization; MRI; deep learning; BraTS; Sartaj

INTRODUCTION

One of the most lethal types of cancer is brain tumors, as per the data provided by the Global Cancer Observatory, about 308,000 people develop primary tumors of the central nervous system annually all around the globe [1]. Patients affected by Grade IV glioblastoma multiforme, as classified by the WHO classification system, have about 14-16 months of median survival time despite trimodal therapy [2]. It is thus crucial to differentiate between tumor grades and types before surgical operations, radiation treatment planning, and chemotherapy. MRI is a gold-standard technology for the non-invasive assessment of brain tumors; however, it remains subject to about 20% interobserver variability among neuroradiologists during manual interpretation [3]. A deep Learning-based Computer-Aided Diagnosis (CAD) system may become a valuable tool for assessing MRI images; the proposed framework is developed and tested as a research/benchmark classification model, not a diagnostic system intended for clinical application.

Standard CNN models ignore the relative significance of different spatial locations in the image during training. In brain MRI scans, the percentage of regions that contain tumor borders, necrosis, and edema is very small compared to healthy tissues, and this problem can be addressed by feature-fusion modules inspired by the attention mechanism that highlight important regions before feature extraction. Hyperparameter optimization is another factor that significantly affects a model's generalization capabilities.

The proposed framework employs an attention-based feature fusion module at the CNN input layer and the HHWO algorithm to perform hyperparameter tuning. This is done using two neuroimaging datasets having different class distributions, sizes, and data collection protocols.

The principal contributions of the paper are:
• **Input-Level Feature-Fusion Mechanism:** An input-level multi-branch feature fusion approach combines convolution, max-pooling, and average-pooling paths to produce complementary representations of local structures and contexts in MRI images.
• **HHWO Optimizer:** An HHWO model that combines Harris Hawks optimization and whale optimization algorithms is designed to optimize the learning rate, weight decay, dropout rate, and dense layer size.
• **Enhanced ABCNN Architecture:** This study evaluates the performance of an advanced ABCNN variant that incorporates Batch Normalization, Global Average Pooling, and label smoothing.
• **Component-Wise Ablation:** Component-wise ablation is performed using a framework (Table 6) to assess the influence of the fusion branches, batch normalization, GAP, label smoothing, and HHWO, and to determine the contribution of HHWO optimization and top-K ensembling separately.
• **Experimental Comparison:** In the comparison of ABCNN+HHWO against CNN, transfer learning using VGG19, CNN-SVM and ABCNN, the results are based on the use of ABCNN+HHWO against CNN/VGG19/CNN-SVM/ABCNN because of differences in resolution of input data, optimizer, schedule of learning rate and number of epochs used for training (Sections 3.3-3.5); hence, the gain is not because of HHWO only. It becomes evident that a comparison of ABCNN using default parameters, Random Search, Bayesian Optimization, HHO only and WOA only needs to be done in a comparable search space with a similar budget in the future (Section 4.8); in the meantime, Section 4.7 puts HHWO in relation with its constituent algorithms and other hybrid algorithms.
• **Statistical Analysis:** Statistics for assessing the significance of differences between performance measures (McNemar's test and Cohen's d) are used to establish the accuracy of the results; the interpretability shortcomings of the feature fusion technique presented here are highlighted for further investigation using Grad-CAM (Section 4.8).
This paper is structured as follows. In Section 2, related works on brain tumor classification using CNNs, attention mechanisms in medical imaging, and hyperparameter optimization using metaheuristics is presented. In Section 3, the dataset used, the preprocessing step, the proposed architectures ABCNN and HHWO, and the baselines for comparison are described. Section 4 presents the classification performance on the BraTS and Sartaj datasets, an ablation study, statistical tests to demonstrate statistical significance, comparisons with state-of-the-art techniques, and, finally, a discussion of the limitations of this study. Section 5 concludes the paper.

2. Related Work

In this part, the related work will be reviewed based on the proposed approach, which can be divided into three categories: convolutional networks for brain tumor classification from MRI images (Section 2.1), attention approaches in the medical imaging area (Section 2.2), and hyperparameter optimization using meta-heuristic algorithms (Section 2.3).

2.1CNN-Based Brain Tumor Classification

Deep convolutional neural networks continue to be the leading solution for brain tumor classification using MRI, and recent research is dedicated to improving performance on standard benchmark problems involving three- and four-class tasks defined on Kaggle and Figshare. The early work on the application of CNNs to brain tumor classification is attributed to Pereira et al. [4], who used a CNN for brain tumor segmentation in MRI, and to Sultan et al. [5], who proposed a deep neural network architecture for multi-class brain tumor classification. Missaoui et al. [6] conducted an extensive review of 107 recent studies that use machine learning and deep learning techniques for brain tumor detection, segmentation, and classification based on MRI, and found that customized and transfer-learning CNNs dominate. At the same time, the split of the patients' dataset is poorly reported in the literature. A ResNet101 architecture with Global Average Pooling and progressive dropout for four class Kaggle MRI dataset is cited to have obtained a test accuracy of 98.73% [7] and demonstrates that improvements in existing architectures still provide tangible benefits without deviating from the transfer-learning methodology used in previous VGG19 [8], InceptionV3 [9], ResNet [10], and EfficientNet [11] works, including the VGG19 transfer-learning method directly applied for brain MRI classifications [12]. As is common in most of the above literature, the architectures treat all locations equally, without suppressing non-diagnostic tissues. According to Missaoui et al., most of the literature reports image-level accuracy without validating that there is no single individual in which the slices of the image are split between training and test sets – an aspect also present in the current study as discussed in Section 4.8.

2.2Attention Mechanisms in Medical Imaging

Spatial and channel attention mechanisms have now become a common method of directing the focus of the CNNs onto relevant tumor regions, instead of the background tissue, and include Attention U-Net with soft attention gates [13], residual attention networks [14], CBAM [15], its application as ResNet50+CBAM by Oladimeji and Ibitoye [16], and squeeze and excitation networks [17]. More recent developments in this line of research involve Zarenia et al. [18] who integrated hierarchical multiscale deformable attention module with

deformable convolutions and achieved greater than 96.5% accuracy on 14-tumor MRI benchmark by adaptive receptive fields based on the tumor geometry, Aiya et al. [19] who used VGG16 backbone along with attention and Grad-CAM explainability and got 99% accuracy on four-class MRI benchmark, and Srinivas and Parvathi [20] who achieved 95.86% accuracy using an explainable attention-based CNN. In all of these papers, the mechanism is applied after several convolution layers. In contrast, in this paper, the multi-branch feature-fusion block is applied at the very beginning, before any learned feature extraction via convolutions, which recent surveys [6] note is comparatively less explored than other methods.

2.3Metaheuristic Hyperparameter Optimization

HHO [21] is inspired from the cooperative hunting of hawks through escape energy, whereas WOA [22] is inspired from the hunting behavior of humpback whales through spiraling. It has been proved [23] that hybrid metaheuristics which incorporate both of these behaviors are better than the individual metaheuristics since they counteract the shortcomings of the other one - HHO is prone to converge in an early stage because it tends to stagnate in the exploitation process of high-dimensional space. At the same time, WOA suffers from early convergence due to the rapid decay of escape energy. Beyond the problem of classifying brain tumors, these two techniques have been independently used to optimize neural networks in general: Thaher et al. [24] used binary HHO for high-dimensional, low-sample-size feature selection, while Aljarah et al. [25] used WOA to optimize neural network connection weights. This approach has recently garnered significant interest precisely in terms of CNN tuning for brain tumors: in particular, Kumar et al. [26] combined Aquila Optimizer and HHO to tune the CNN parameters for four-class brain tumor classification and achieved 87.34% accuracy on a 7,023-image benchmark, and Dehkordi et al. [27] utilized a Nonlinear Lévy Chaotic Moth Flame Optimizer (also using Lévy perturbation like HHWO), achieving 97.40% accuracy on a 2,314-image benchmark. This reinforces the hypothesis that hybrid metaheuristics with Lévy perturbation offer benefits over single-heuristic optimization for such problems. As can be seen from the cited works, accuracy largely depends on the number of images and classes in the dataset; hence, the need for a comparison using the same protocol described in Section 4.8.

This work's positioning: with respect to (i) standard and transfer learning CNNs (Section 2.1), which do not weight spatial regions and whose performance is typically assessed in the absence of patient-level validation; (ii) attention modules added to the network after initial convolutional layers (Section 2.2); and (iii) single metaheuristic methods that may get stuck or converge prematurely in the ABCNN hyperparameter search space (Section 2.3) — this research merges an input-level attention-motivated feature fusion strategy with a composite Harris Hawks and Whale optimization approach and clearly describes the constraints of its own technique in Section 4.8.

3. METHODOLOGY

This section explains the dataset and pre-processing pipeline, the novel architecture called ABCNN, the HHWO hyper-parameter optimization method, and the baselines against which comparison is made.

3.1Datasets and Class Distribution

Two benchmark MR brain tumor datasets, each having different class levels of classification, are utilized in the current research. Class-level statistics for the two datasets are presented in Table 1 below, while the subsequent parts describe the significance, imaging features, and class imbalance of the two datasets. Table 1 presents the newly adjusted TRAIN/HHWO_Val/TEST partitioning scheme adopted for the ABCNN and ABCNN+HHWO results reported in this paper; the TEST set is not included in the HHWO hyperparameter tuning (Section 4.8).

In this work, both datasets are considered as already pre-extracted sets of 2D images. In particular, the BraTS 2021 dataset includes four co-registered MRI sequences (T1, T1Gd/T1ce, T2, and T2-FLAIR) for each patient. The total number of 2D axial slices used for experiments, which equals 14,734 samples as shown in Table 1, is not specified; in other words, it remains unknown whether they correspond to certain modality(ies) and which slice selection strategy was used (e.g., including slices with tumors only or all slices with a certain number of slices per patient). This information should be provided for reproducing results (see Section 4.8). Images from the BraTS 2021 dataset used in this paper were downloaded from the derivative dataset ranbeerreddy/bratss hosted on Kaggle. Therefore, the pipeline of skull-stripping, registration, normalization, and slicing used before our preprocessing was inherited from this derivative dataset, which makes it more difficult to compare this work with those papers that use the BraTS 2021 dataset from the official challenge website. The Sartaj (Kaggle) dataset contains pre-extracted T1-weighted contrast-enhanced (T1ce) MRI images in 2D format.

Table 1. Benchmark Dataset Summary with Class-Level Distribution

Dataset	Class	Total	Train	HHWO_Val	Test
BraTS	HGG	8,956	6,269	1,343	1,344

2021	LGG	5,778	4,045	867	866
Sartaj	Glioma	1,621	1,145	242	234
	Meningioma	1,645	1,149	246	250
	No Tumor	2,000	1,428	308	264

3.2Preprocessing

Preprocessing is performed identically across both datasets to maintain spatial resolution, intensity values, and augmentation techniques, so that the architecture and optimization techniques can be evaluated exclusively. All images are read using three color channels, irrespective of the number of channels in the original file (tf.io.decode_image(..., channels=3) for BraTS, tf.image.decode_jpeg(..., channels=3) for Sartaj); this results in the replication of a single channel/grayscale MRI slice into all three channels to produce the required 3-channel input as described in Table 1. The images are resized to the target resolution using bilinear interpolation and normalized to [0, 1]:

$$x = x255 \quad (1)$$

Here, x represents the actual pixel intensity while x255 represents the normalized pixel intensity in [0,1]. Balanced class weight handling for the class imbalance problem is done as follows:

$$wc = NK \cdot Nc \quad (2)$$

Where N stands for the number of training examples, K stands for the number of classes, and N_c stands for the count of class c; w_c stands for the resultant balanced weight of class c. Data augmentation has been done using Keras pipeline (random horizontal flipping p=0.5, rotation ±12°, translation ±8%, zooming ±15%, and contrast adjustment ±10%) on a GPU that is limited to the training split and not the validation split, and has been performed identically in all five models (CNN, VGG19+TL, CNN-SVM, ABCNN, and ABCNN+HHWO) so that any variation due to augmentation will not cloud the comparison.

3.3Proposed ABCNN Architecture

In the case of the ABCNN structure, the feature fusion operation motivated by the attention operation takes place before the operation of convolutional feature extraction, such that the subsequent capacity for representation is dedicated only to the meaningful MRI areas.

The main difference between the current design and conventional multi-branch feature extraction (such as Inception-style modules) is twofold. First, conventional multi-branch blocks are typically added after several learned convolutional layers to enrich features across several levels of the receptive field. The current block, on the other hand, is applied immediately after the input stage and contains no learned convolutional layers. Second, conventional multi-branch layers typically combine several learned convolutional pathways (for example, convolutions with kernels of various sizes). The current block, on the other hand, combines only one learned pathway and two fixed, non-learned statistical features (max- and average pooling) that provide a cheap estimate of locally important and locally homogeneous areas right after the first layer, even before learning any weights. The goal here is to bias early representation toward diagnostically meaningful parts rather than enriching already learned features, which is the goal of conventional multi-branch modules.

3.3.1Input-Stage Attention-Inspired Feature-Fusion Block

The attention-driven feature fusion layer is the main innovation in the ABCNN architecture, in which features are extracted from the input image via three parallel paths to form an attention-driven fusion feature representation, which is then passed to the remaining layers of the network, as presented formally below.

For the input tensor X ∈ R^(H×W×C) (Height H, Width W, Channel count C), three parallel paths are calculated as: X ∈ R^(H×W×C) (H=W=160 or 128, C=3)

• Pathway 1 — Convolutional Branch: 3 × 3 Conv2D with 3 filters and ReLU activation function, followed by 2 × 2 max-pooling, resulting in feature map F_conv:

$$Fconv = MaxPool2Conv3\times3X,\ 3,\ ReLU \in \mathbb{R}H2\times W2\times 3 \quad (3)$$

• Pathway 2 — Max-Pooling Branch: Highlights sparsely distributed, highly active boundaries of structure to produce feature map F_max:

$$Fmax = MaxPool2X \in \mathbb{R}H2\times W2\times C \quad (4)$$

• Pathway 3 — Average-Pooling Branch: Encodes diffuse low frequency and tissue homogeneity, yielding

feature map F_avg:

$$Favg = AvgPool2X \in \mathbb{R}H2 \times W2 \times C \quad (5)$$

Concatenating F_conv, F_max, and F_avg channelwise gives the attention output A:

$$A = ConcatFmax, Favg, Fconv \in \mathbb{R}H2 \times W2 \times 2C+3 \quad (6)$$

While max pooling helps retain high activations for structural break detection, average pooling helps retain spatial context; together with convolution, descriptor A facilitates feature extraction.
In the design of the proposed input-level feature-fusion block, F_conv is paired with two fixed maximum and average-pooling descriptors, F_max and F_avg, respectively. While traditional gated attention blocks use explicit computations of a normalized spatial weight map, such as sigmoid/softmax gating multiplied by the input (as done in CBAM [15], for example), the proposed block does not compute a normalized attention map, nor does it compute any explicit attention scores (normalized or otherwise). However, the max-pooling branch F_max serves as a fixed, untrained descriptor of local high-activation structures (tumor-margin type). In contrast, the average-pooling F_avg branch provides a fixed, untrained descriptor of locally uniform, low-frequency content (diffuse tissue and edema), and channel-wise concatenation with F_conv (eq. (6)) allows the use of all three descriptors in subsequent layers. As such, the expression attention-based feature fusion has been adopted in this document to characterize the spatial focus of the block without any indication of a trainable attention mask; Section 4.8 explains that visualization techniques such as Grad-CAM or its equivalent need to be applied to verify that the fusion focuses on the tumor borders, necrosis, and edema.

3.3.2Improved ABCNN+HHWO Backbone

These improvements have been made to counteract the four main drawbacks associated with the traditional ABCNN backbone, which include: internal covariate shift, over-parameterization, confidence in the results of the softmax layer, and sensitivity to hyperparameter selection. The improvements and their mathematical representations are as follows: Batch normalization after each convoluted layer:

$$BNx = \gamma \cdot x - \mu B \sigma B2 + \varepsilon + \beta \quad (7)$$

where mu_B and sigma_B^2 are the mini-batch mean and variance, gamma and beta are affine parameters that are learned, and epsilon is the small value for numerical stability (note that this is not related to epsilon in Eq. (9)). Global Average Pooling is used in place of complete flattening to compute the mean in the channel dimension: $\mu B \sigma B2$

$$GAPxc = 1H \cdot W \cdot \Sigma_i \Sigma_j xi,j,c \quad (8)$$

Label smoothing alters cross-entropy loss:

$$Lsmooth = 1 - \varepsilon \cdot LCE + \varepsilon K, \quad \varepsilon = 0.05 \quad (9)$$

L_smooth is the label-smoothed loss function, L_CE is the cross-entropy loss function, K is the number of classes, and epsilon = 0.05 is the smoothing coefficient (and does not correlate with the Batch Normalization coefficient in Eq. (7)), while label smoothing mitigates the effect of overconfident output from the softmax layer. Both the dimensions du of the dense layer and the dropout rate dr are optimized by HHWO. du*dr*
A summary of the ABCNN+HHWO Layer Architecture is presented in Table 2 below.

Table 2. ABCNN+HHWO Architecture Specification

Stage	Operation	Config.	Output
Input	MRI slice	128x128x3 (BraTS)/ 160x160x3 (Sartaj); norm. to [0,1]	X (Eq. 1)
Attn. Pathway 1	Conv2D 3x3,3 filters, ReLU, 2x2 max-pool	learned	F_conv (Eq. 3)
Attn. Pathway 2	2x2 max-pool	fixed	F_max (Eq. 4)
Attn. Pathway 3	2x2 avg-pool	fixed	F_avg (Eq. 5)

Fusion	Channel-wise concat.	-	A (Eq. 6)
Conv. tower	Conv. layers, each + BN (Eq. 7)	layer count/filters beyond attn. block not specified (Sec. 4.8)	feature maps
Global pooling	GAP	-	GAP(x_c) (Eq. 8)
Head	Dense, dropout, softmax	du, dr HHWO-tuned (Eqs. 10-13); label smoothing eps=0.05 (Eq. 9)	K-class probs.
Output	Softmax	K=2 (BraTS) or K=3 (Sartaj)	predicted class

3.4Hybrid Harris Hawks–Whale Optimization (HHWO)

HHWO searches for four ABCNN hyperparameters, namely, learning rate, weight decay, dropout, and dense layer size, for Sartaj; for BraTS, HHWO additionally searches in a fifth dimension, that is, a squeeze-excitation ratio, because there exists a squeeze-excitation parameter within the BraTS model architecture that is not mentioned either in the model architecture independent of any data set as described in Section 3.3 (Table 2), or within Sartaj's search space (Table 9, Section 4.7). HHWO members search the aforementioned four- or five-dimensional space via position adaptation using an adaptive energy parameter.

Four of these hyperparameters were chosen for HHWO optimization instead of architectural ones, like the size of kernel, number of convolutional layers or the activation function since the learning rate and weight decay control the optimization and generalization dynamics, while dropout and dense layer width affect the balance between the capacity and regularization of the classifier part; altogether, these hyperparameters have been widely reported to influence the most the CNN generalization for a given architecture. Architectural hyperparameters, however, were kept unchanged (Section 3.3, Table 2) to make the optimization space manageable with a population-based metaheuristic approach and to dissociate hyperparameter tuning from the architecture search; this aspect is addressed in Section 5.

3.4.1Search Space and Fitness Function

HHWO uses a normalized four-dimensional agent for each possible hyperparameter combination, based on the validation loss obtained via a mini-training process that assesses the agent's quality. The learning rate and weight decay have been encoded in a log-uniform manner, thus allowing for the same level of exploration in all orders of magnitude.

HHWO is run using the four ABCNN hyperparameters, which include the learning rate lr, weight decay wd, dropout rate dr, and dense layer du as: $\theta = (lr, wd, dr, du)\ x \in [0,1]^4$

$$lr = 10=\log10\ 5\times10-5\ x1\cdot\log10\ 3\times10-3-\log10\ 5\times10-5\} \quad (10)$$

$$wd = 10=\log10\ 10-5\ +x2\cdot\log10\ 10-3\ -\log10\ 10-5 \quad (11)$$

$$dr = 0.2 + x3\cdot 0.3 \quad (12)$$

$$du = 256 + x_4 \cdot 768 \quad (13)$$

Fitness F(x) is the lowest validation loss L_val achieved for hyperparameters theta across E_fit mini-epochs and is bounded by F_cap = 5.0:Efit.

$$F(x) = \min\_\{e \in \{1,\ldots,E\_fit\}\}\ L\_val(\theta; e),\ \text{clipped to } [0, F\_cap] \quad (14)$$

Training to convergence for every candidate configuration would be computationally prohibitive for a population-based approach iterated multiple times; instead, a low-fidelity, computationally cheap alternative is to use the minimum validation loss obtained after a limited number of mini-epochs of training (E_fit). To limit the effect of divergent candidate configurations, which could skew the overall fitness evaluation, we limit the fitness value to F_cap = 5.0. With regards to computational cost, HHWO evaluates on the order of population_size * iterations * E_fit mini-epochs of training during its offline search, while each baseline requires a single full training run. The computational effort spent in the initial search phase is independent of the inference cost because HHWO performs hyper-parameter tuning but no architectural tuning. Consequently, the ABCNN+HHWO architecture is identical to that of the baseline ABCNN network during inference. Population size, iterations, number of fitness evaluations, stop condition, random seeds, and the exact search cost are not

provided in this manuscript and must be included in the implementation to achieve full reproducibility (Section 4.8).

3.4.2Adaptive Escape Energy

The escape energy function E(t) is the main variable responsible for the HHWO phase transitions, since its magnitude decreases monotonically throughout the training process from 2 to 0, moving the algorithm from exploration to exploitation – just like hawks go from exploratory flight to a concentrated attack.
The value of the escape energy function at the t-th iteration out of T iterations is defined by:

$$E0 \sim \text{Uniform}-1, 1 \quad (15)$$

$$Et = 2 \cdot E0 \cdot 1 - tT \quad (16)$$

The value of |E(t)| goes down from ≈2 to 0, governing the balance between exploration and exploitation; E0 in Equation (15) is the first drawn from Uniform(-1,1).

3.4.3Exploration Phase (|E| ≥ 1)

If the escape energy exceeds 1, hyperparameters are explored through exploratory repositioning using two equally likely approaches inspired by the behavior of Harris hawks: random perching and repositioning in response to the presence of rabbits.
Let q be an equally likely random variable in the range:q ∈ [0,1]

$$\text{If } q \geq 0.5: \; x_i \leftarrow x_r - r1 \cdot x_r - 2r2 \cdot x_i \quad \text{random perch} \quad (17)$$

$$\text{If } q < 0.5: \; x_i \leftarrow x* - x - r3 \cdot LB + r4 \cdot UB - LB \quad \text{rabbit} - \text{mean} \quad (18)$$

Where x* denotes the global best position of the agents, $\bar{x}$ (x-bar) is the mean position of the agents, x_r is a randomly chosen agent's position, x_i is the position of agent i, and r1-r4 are random numbers between [0,1].LB and UB stand for Lower Bound and Upper Bound, respectively.LB=0,UB=1

3.4.4Exploitation Phase (|E| < 1)

If the escape energy value becomes less than 1, the agents move into the exploitation stage, searching for a better position near the best-so-far solution using one of the three approaches randomly selected by the probability value p (Equation 2): the HHO soft and hard besiege (Eqs. 19-20, where J is a random jump-strength factor) and the WOA logarithmic spiral (Eq. 21). If p is uniformly distributed from [0,1]:

$$\text{If } p < 0.5, E \geq 0.5 \text{ Soft Besiege: } \; x_i \leftarrow x* - x_i - E \cdot J \quad (19)$$

$$\text{If } p < 0.5, E < 0.5 \text{ Hard Besiege: } \; x_i \leftarrow x* - E \cdot x* - x_i \quad (20)$$

$$\text{If } p \geq 0.5 \text{ WOA Spiral: } \; x_i \leftarrow x* - x_i \cdot ebl \cdot \cos 2\pi l + x* \quad (21)$$

where b is the shape parameter of the logarithmic spiral that is independent of b which represents the discordant pairs of McNemar test and is given in Eq. (24) below, and l is a random variable in the range of [-1,1].J = 2(1-r),b=1l ~ Uniform(-1,

3.4.5Lévy Flight Perturbation

The Lévy flight component improves the HHWO search mechanism through heavy-tailed noise, thereby allowing agents to escape the traps of local optima characteristic of a normal random walk. Step sizes are power-law distributed with parameter β = 1.5 (here the β has nothing to do with the Batch Normalization scale/shifting parameters of Eq. (7))

$$\sigma = \left[\Gamma(1+\beta) \cdot \frac{\sin\left(\frac{\pi\beta}{2}\right)}{\left(\Gamma\left(\frac{1+\beta}{2}\right) \cdot \beta \cdot 2^{\left\{\frac{\beta-1}{2}\right\}}\right)} \right]^{\left\{\frac{1}{\beta}\right\}}, \beta = 1.5 \quad (22)$$

$$Lv = \frac{\left(u \sim N(0,\sigma^2)\right)}{|v \sim N(0,1)|^{\left\{\frac{1}{\beta}\right\}}} \quad (23)$$

In Eq. (22), sigma denotes the Lévy step-size scale parameter while Gamma(.) is the gamma function; in Eq. (23), u and v are auxiliary standard-normal random variables, where u ~ N(0, sigma^2) and v ~ N(0,1).

The Lévy flights produce heavy-tailed step sizes which facilitate occasional large jumps such that premature convergence to local optima is avoided. Elitism ensures that one agent is always at x* in every iteration: $x_1 \leftarrow x^*$. Figure 1 shows the system block diagram together with the HHWO tuning procedure.

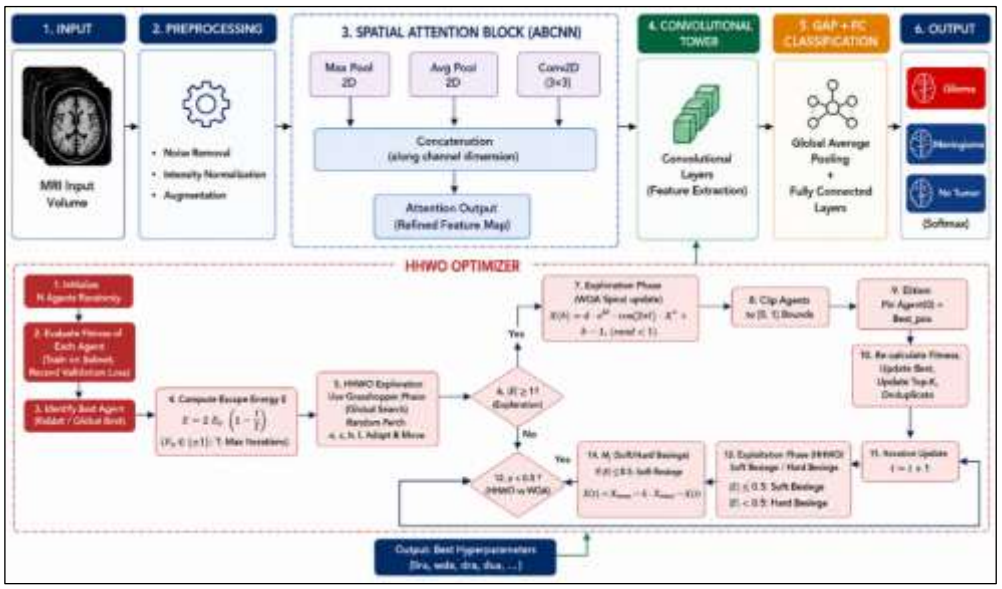


Figure 1 ABCNN+HHWO system block diagram. HHWO optimizes lr*, wd*, dr*, du* using fitness evaluated on a stratified training subsample

HHWO optimizer performs an offline search to find the optimal hyperparameters by minimizing validation loss on a stratified training set, and the Softmax layer makes the final prediction for each tumor class.

3.5Baseline Models

For thorough benchmarking, four baselines across various model architectures are trained using the same preprocessing and experimental setup.

Standard CNN: 3 conv layers (16/32/64 filters), Dense(128), softmax; RMSprop (lr=1e-3), 50 epochs, batch=32.

VGG19 + Transfer Learning (TL): ImageNet-pretrained VGG19, first 5 layers frozen, plus Conv2D(32,64), Dense(256), Dropout, Softmax; input 150×150, RMSprop (lr=1e-5), 20 epochs.

CNN-SVM: Same as CNN but with squared hinge loss and L2 regularization (λ = 0.001) instead of softmax classifier, i.e., SVM classification head on CNN features; RMSprop (lr = 1e-3), 50 epochs.

ABCNN (baseline): Full attention block, Flatten + Dense(4096) × 2, cross entropy loss function, Adam (learning rate = 1e-4), fixed hyperparameters, 50 epochs.

ABCNN+HHWO (proposed): full ABCNN backbone + GAP, BN, label smoothing, and HHWO-tuned hyperparameters (lr*, wd*, dr*, du*). Under the original, uncorrected procedure, the key results from Tables 4, 5, and 8 (as well as ablation variant V7 in Table 6) applied test-time augmentation (TTA) and top-K averaging at test time. In contrast, ablation variant V6 used the same trained model without TTA/top-K averaging (i.e., a single forward pass), thereby attributing the performance difference to the test-time process. Under the new, leakage-tested TRAIN/HHWO_VAL/TEST procedure (see Section 4.8), however, it was observed that TTA significantly reduced independent-TEST-set accuracy on both datasets — especially so on Sartaj, where 8-round TTA decreased accuracy from 98.26% (a single forward pass) down to 59.63%. Thus, the new key results for ABCNN+HHWO in Tables 4, 5, and 8 are based on the single-forward-pass setting. Fine-tuning is optional and was not used to obtain the results presented in this paper.

4.RESULTS AND DISCUSSION

This section contains quantitative results, along with visualizations of the training process, confusion matrices, and classification reports for all five models across both datasets, using a Google Colab T4/P100 Graphics Processing Unit (GPU) running TensorFlow 2.x with mixed-precision training and XLA JIT compilation.

4.1HHWO Convergence and Optimized Hyperparameters

Before conducting the classification task analysis, this section examines how the HHWO optimizer's adjusted parameters differ across the two benchmark data sets.

Hyperparameters optimized for Sartaj via HHWO in the corrected search (see Table 3) are lr* = $4.94\text{x}10^{-3}$, wd* = $6.7\text{x}10^{-6}$, dr* = 0.208, and du* = 160; all of these differ from the default values. The BraTS line in Table 3

represents the results of the original search before correction. The search was re-executed in the new TRAIN/HHWO_VAL/TEST split for this update. However, because the console output for this search is not recorded, the updated hyperparameter values for BraTS remain unavailable.

Table 3. HHWO-Optimized Hyperparameter Configurations per Dataset (Sartaj: revised, leakage-checked search; BraTS: original pre-correction search, not reproduced in this revision — see Section 4.8)

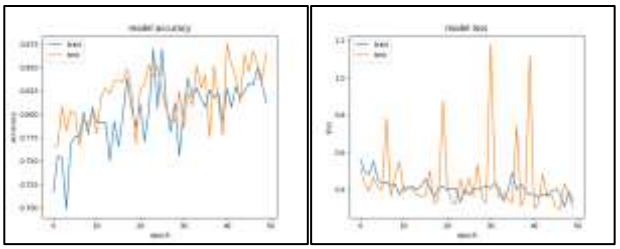
Dataset	Best lr*	Best wd*	Best dr*	Best du*	Best Val Loss
BraTS	1.2×10^{-4}	8.3×10^{-5}	0.31	640	0.041 (orig. run)
Sartaj	4.94×10^{-3}	6.7×10^{-6}	0.208	160	0.7411 (revised)

4.2Results on BraTS 2021 Dataset

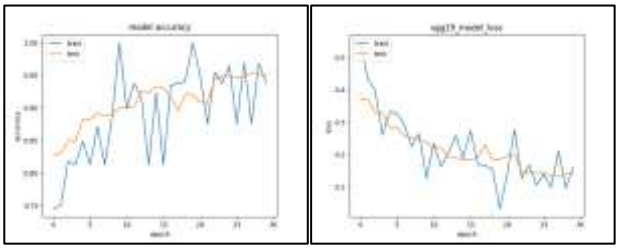
However, the BraTS 2021 binary glioma grading challenge poses difficulty in evaluation, since of the 2,210 independent TEST images, 1,344 correspond to high-grade gliomas (HGGs) and 866 to low-grade gliomas (LGGs). This is because both HGGs and LGGS have different pathological classifications and highly heterogeneous imaging characteristics, and an imbalance between the classes of about 1.55:1 is observed. The performance of ABCNN and ABCNN+HHWO is evaluated using precision, recall, F1 score, and overall accuracy on the TEST dataset (Table 4); the training curves and confusion matrices in Fig. 2(a)–(e) are not regenerated for CNN, VGG19+TL, and CNN-SVM.

Table 4. Performance Comparison on BraTS 2021 Dataset (Best results highlighted; ABCNN/ABCNN+HHWO rows revised, independent-TEST-set protocol; CNN/VGG19+TL/CNN-SVM rows original protocol, not re-evaluated under the revised split — see Section 4.8)

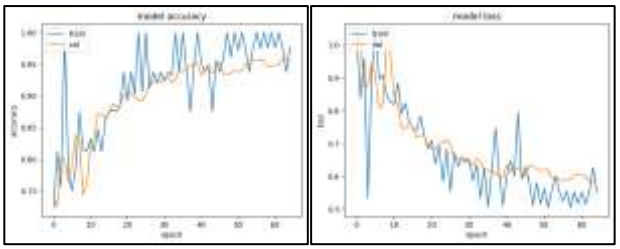
Model	Precision	Recall	F1-Score	Accuracy (%)
CNN	0.8597	0.8602	0.8594	86.02
VGG19+TL	0.9535	0.9511	0.9513	95.11
CNN-SVM	0.9544	0.9544	0.9543	95.44
ABCNN	0.9473	0.9466	0.9468	94.66
Proposed ABCNN + HHWO	0.9772	0.9769	0.9770	97.69



(a)



(b)



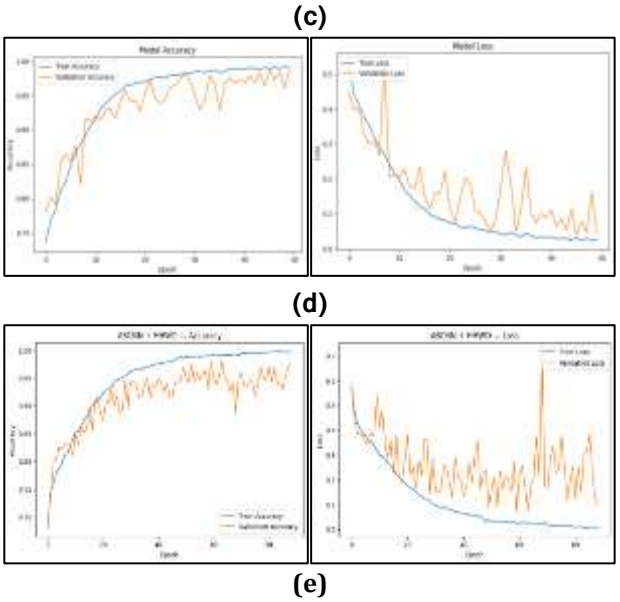


Fig. 2. BraTS — Training accuracy, loss curves, and confusion matrix for (a) CNN, (b) VGG19+TL, (c) CNN-SVM, (d) ABCNN (baseline), (e) Proposed ABCNN+HHWO

The best classification accuracy was obtained from the combination ABCNN+HHWO which was the highest accuracy of 97.69% in the independent BraTS TEST partition dataset compared to CNN (86.02%), VGG19 (95.11%), and CNN-SVM (95.44%), which were under the original protocol but not retested here, as well as the ABCNN baseline (94.66%), which was re-tested in the revised BraTS TEST partition; hyper-parameter tuning using HHWO resulted in a 3.03% improvement of the accuracy over the ABCNN baseline under this protocol. ABCNN+HHWO incorrectly classified 51 out of 2,210 TEST images (2.31% error rate) in that 37 slices of HGG were wrongly identified as LGG, and 14 slices of LGG were wrongly identified as HGG – much better than the conventional CNN, which often misclassifies LGG as HGG (Fig. 2). It is important that this model has a low LGG/HGG misclassification rate because the HGG/LGG grade impacts the treatment plan; we cannot clinically conclude anything about treatment outcomes without further clinical investigation.

4.3Results on Sartaj Dataset

The classification task using the Sartaj data set involves a challenging three-class classification task involving classification of high-grade glioma and low-grade tumors into two classes, as well as a no-tumor class, against each other based on a group-de-duplicated TEST split of 748 images (234 glioma, 250 meningioma, and 264 no-tumor), which provides balanced class representation and reflects the actual triage task faced in clinical practice. The quantitative results for ABCNN and ABCNN+HHWO are presented in Table 5 using this new approach, while those for CNN, VGG19, and CNN-SVM were derived with the original approach and have not been repeated on this split.

Table 5. Performance Comparison on Sartaj Dataset (Best results highlighted; ABCNN/ABCNN+HHWO rows revised, independent-TEST-set protocol; CNN/VGG19/CNN-SVM rows original protocol, not re-evaluated under the revised split — see Section 4.8)

Model	Precision	Recall	F1-Score	Accuracy (%)
CNN	0.9421	0.9404	0.9408	94.04
VGG19+TL	0.8898	0.8825	0.8834	88.25
CNN-SVM	0.8729	0.8685	0.8685	86.85
ABCNN	0.9509	0.9505	0.9505	95.05
Proposed ABCNN+ HHWO	0.9827	0.9826	0.9826	98.26

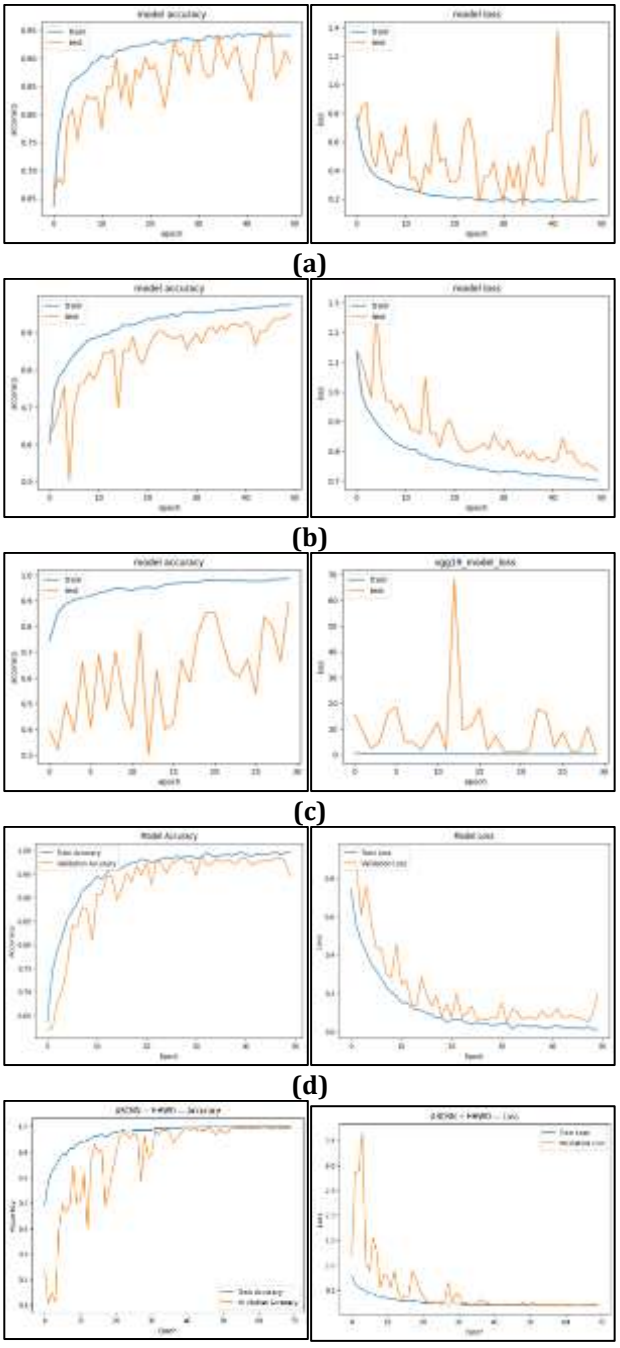


Fig. 3. Sartaj — Training accuracy, loss curves, and confusion matrix for (a) CNN, (b) VGG19+TL, (c) CNN-SVM, (d) ABCNN (baseline), (e) Proposed ABCNN+HHWO

As shown in Table 5, the proposed model achieves 98.26% accuracy on the independent Sartaj TEST set, which is superior to all other models compared. Both VGG19 (88.25%) and CNN-SVM (86.85%) models have a worse performance compared to the simple CNN (94.04%), in the initial setting of the protocol before correction; the performance of these three baseline models was not evaluated in the improved version of the TEST set because, at the 160x160 input size, VGG19 model reduces the spatial feature map to 5x5 due to five pooling layers, failing to capture the necessary morphological information to distinguish between gliomas and meningiomas. On the TEST dataset after modification, ABCNN+HHWO incorrectly labels 13 of 748 images (error rate of 1.74%), with the majority being errors of classification due to overlapping meningioma (5 meningiomas labeled as gliomas, 5 as no-tumors) with a minor overlap of glioma and meningioma (2 images) and no-tumor and meningioma (1 image) – the same two classes which are reported as the major source of error in all classifier confusion matrices of Fig. 3(a-e) due to similar T1ce intensities and morphologies of glioma and meningioma.

Ablation Study

To analyze how performance improvements can be attributed to specific components, an ablation study with seven variants is conducted on both benchmark datasets by removing one component at a time from the entire ABCNN+HHWO system (V7). These results are presented in Table 6. The ablation variants (V1–V7) shown in Table 6 correspond to the experimental setup before correction (image-level Sartaj split and validation set evaluation), which was not done using the corrected TRAIN/HHWO_VAL/TEST protocol used to obtain the main results for ABCNN and ABCNN+HHWO in Tables 4–5 and Section 4.8; hence, the accuracies reported below for V7 (98.00% for BraTS, 99.51% for Sartaj) are different from the corrected accuracies in the TEST set (97.69% for BraTS, 98.26% for Sartaj).

The notation for the variations of Table 6 follows Section 3.5 in the following way. The variants 'ABCNN' and 'ABCNN (baseline)' denote the fixed-hyperparameters baseline used in Tables 4 and 5 (full-attention block, Flatten + Dense(4096) x2, Adam optimizer, no BN, no GAP, no label smoothing, and no HHWO tuning). 'ABCNN + HHWO (no ensemble)' (V6) is the HHWO-tuned model with BN, GAP, and label smoothing evaluated in a single forward pass at inference. 'ABCNN + HHWO (full)' (V7) includes test-time augmentation and top-K averaging at inference in addition to V6, and is the version reported as 'Proposed ABCNN+HHWO' in Tables 4, 5 and 8 according to the original protocol; however, under the revised protocol, test-time augmentation was shown to reduce accuracy on independent-TEST set significantly (Section 4.8), and therefore the revised 'Proposed ABCNN+HHWO' in Tables 4, 5 and 8 uses the V6 version evaluated on TEST partition. Label smoothing is introduced in conjunction with BN and GAP at V5 and, therefore, not varied independently in this ablation study; consequently, its effect cannot be distinguished from BN/GAP in the results below. This is an open item in Section 4.8.

Table 6. Ablation Study: Contribution of Each ABCNN+HHWO Component (original, pre-correction protocol; not re-run in this revision — see Section 4.8)

Variant	BraTS (%)	Sartaj (%)
V1: Baseline CNN	86.02	94.04
V2: ABCNN (Max)	91.63	95.97
V3: ABCNN (Max+Avg)	95.40	97.12
V4:ABCNN(Conv+Max+Avg)	95.41	94.46
V5: ABCNN + BN +GAP	96.80	96.20
V6:ABCNN+HHWO(no ens.)	97.52	99.21
V7: ABCNN + HHWO (full)	98.00	99.51

The results for BraTS in Table 6 show that accuracy increases monotonically from V1 to V7. In Sartaj's case, there is an increasing trend from V1 (94.04%) to V7 (99.51%), but it is not monotonic. The maximum accuracy is 97.12% for V3 (Max+Avg branches). After that, accuracy drops to 94.46% in V4 because only the convolutional branch combination was tested at this stage. After V4, accuracy increases beyond V3's value from V5 onwards. However, the maximum contribution in BraTS comes from the average-pooling branch (V2→V3), as it can extract complementary contextual information. On the other hand, the convolutional branch (V3→V4) yielded only a small improvement in BraTS for spatial feature extraction. The use of BN+GAP in V5 increases the model's accuracy on the BraTS and Sartaj datasets by 1.39% and 1.74%, respectively. The last step of HHWO-based hyperparameter tuning (V5→V6) yields the largest performance gain on the Sartaj dataset (+3.01 percentage points).

4.5Statistical Significance Testing

Statistical significance of any performance gain achieved by improving accuracy must be demonstrated to be independent of random chance in either initialization or data partitioning to validate medical AI models. In the current subsection, two distinct experimental units are used. McNemar's test [28] has been calculated at the image-prediction level. In one run, correct/incorrect predictions made by ABCNN+HHWO and each baseline are paired at the image level in the same validation set (the original one Valid; see Table 1 for new TRAIN/HHWO_Val/TEST splits), and counts of discordant pairs (Eq. (24)) are made for all images in this split. Cohen's d [29] has been calculated at the run level (ABCNN+HHWO), and each baseline is trained three times with different random seeds, yielding three pairs of accuracy values for each comparison. The headline accuracies presented in Tables 4, 5, and 8 are based on results of a single run; per-run accuracies and a mean ± standard deviation or 95% confidence interval over the three runs are not presented in this manuscript (Section 4.8). Such restriction holds for the six experiments with CNN, VGG19+TL, and CNN-SVM as baselines according to the original protocol; for the comparison of ABCNN vs. ABCNN+HHWO in both cases, this paper utilizes the TEST set independently, which is not used for fitness estimation of HHWO (Section 3.4) and is tested exactly once. The same TEST set is subjected to a non-parametric assessment (2,000-resample bootstrap 95%

CI for the ABCNN+HHWO accuracy), giving [97.06%, 98.28%] for BraTS and [97.33%, 99.20%] for Sartaj. The test statistic for the McNemar test, given by Eq. (24), follows a χ^2 distribution with 1 degree of freedom under H0.

$$\chi 2=b-c-12b+c \quad (24)$$

with χ^2 being McNemar's test statistic, while b and c are the discordant pair counts for ABCNN+HHWO vs. the benchmark model (this b is distinct from the WOA logarithmic-spiral constant b in Eq. (21)). A large value of χ^2 (and small p-value) means that the difference is statistically significant; the practical significance is determined using Cohen's d [29] and three runs with different seeds. The full dataset is shown in Table 7. The discordant-pair counts (b, c), per-run accuracies, and their confidence intervals are not given in Table 7, and multiple comparison adjustment is not used in any of the 12 comparisons. This is noted as a limitation in Section 4.8.

Table 7. Statistical Significance Tests (ABCNN+HHWO-vs-ABCNN rows recomputed under the revised, independent-TEST-set protocol; all other rows, and all Cohen's d values, reflect the original protocol and were not reproduced in this revision — see Section 4.8)

hComparison	Dataset	McNemar χ^2	p-value	Cohen's d
ABCNN + HHWO vs. CNN	BraTS	272.17	< 0.001	4.79
ABCNN + HHWO vs. VGG19 + TL	BraTS	35.83	< 0.001	1.16
ABCNN + HHWO vs. CNN-SVM	BraTS	29.42	< 0.001	1.02
ABCNN+HHWO vs. ABCNN	BraTS	36.61	< 0.0001	1.04 (orig. run)
ABCNN+HHWO vs. CNN	Sartaj	45.63	< 0.001	2.19
ABCNN+HHWO vs. VGG19+TL	Sartaj	103.87	< 0.001	4.50
ABCNN+HHWO vs. CNN-SVM	Sartaj	118.06	< 0.001	5.06
ABCNN+HHWO vs. ABCNN	Sartaj	13.23	0.0003	2.02 (orig. run)

All eight pairwise comparisons have p<0.0001. Comparisons against the three non-attention-based baselines (CNN, VGG19+TL, CNN-SVM) are significant at p<0.0001, having high Cohen's d values (d=1.02-5.06) on both datasets, according to the pre-correction protocol (which is not presented in this revision). On the other hand, the comparison between ABCNN+HHWO and ABCNN was calculated again using McNemar's test with the new, independent TEST partition: χ^2=36.61, p<0.0001 on BraTS, χ^2=13.23, p=0.0003 on Sartaj — both significant, yet more significant than the original protocol statistics (χ^2=29.99 and χ^2=41.47), which can be found in Table 7 below. The values of Cohen's d (d=1.04 for BraTS and d=2.02 for Sartaj) in this comparison remain the same as

4.6Comparison with State-of-the-Art

The proposed ABCNN+HHWO algorithm is compared with seven cutting-edge algorithms, three of which were introduced between 2025 and 2026, on the same benchmark datasets (see Table 8).

Table 8. Comparison with Previously Reported Methods (Proposed ABCNN+HHWO values reflect the revised, independent-TEST-set protocol — see Section 4.8)

Method	Dataset	Accuracy (%)
Afshar et al. – CapsNet [30]	BraTS	86.56
Badza & Barjaktarovic – CNN[31]	BraTS	96.56
Harshavardhan et al. – QuantumMedDx [32]	BraTS	96.42
Oladimeji & Ibitoye – ResNet50+CBAM [16]	Sartaj	96.24
Khan et al. EfficientNet-B3 [33]	Sartaj	97.85

Sankari et al. – Hierarchical Multi-Scale ViT [34]	Kaggle (4-class)	98.70
Ke et al. – MCACNN-SVM [35]	Kaggle (4-class)	96.90
Proposed ABCNN+HHWO	BraTS	97.69
Proposed ABCNN+HHWO	Sartaj	98.26

Reported values are derived from the corresponding publications and are not necessarily obtained using identical dataset partitions, preprocessing procedures, class definitions, or evaluation protocols. Therefore, these values provide contextual rather than strictly controlled comparisons.

In BraTS, the modified accuracy of ABCNN+HHWO (97.69%) is comparable to the accuracy values reported for literature solutions to the same problem, such as CapsNet [30] (86.56%), Badza & Barjaktarovic [31] (96.56%), and a more recent (2026) quantum/classical hybrid approach tested against the same BraTS 2021 HGG-vs-LGG problem, Harshavardhan et al.'s QuantumMedDx [32] (96.42%) – nominally the highest value in Table 8, although from an uncontrolled reproduction. In Sartaj, the modified accuracy of ABCNN+HHWO (98.26%) is also comparable to that of EfficientNet-B3 [33] (97.85%), ResNet50+CBAM [16] (96.24%), and Ke et al.'s multi-scale channel attention convolutional neural network with an SVM classifier [35] (96.90%), also evaluated on the same underlying Kaggle brain tumor MRIs collection but in its full four-class version (glioma, meningioma, pituitary, no tumor), not the three-class glioma/meningioma/no tumor subset tested here, and also not quite reaching Sankari et al.'s hierarchical multi-scale vision transformer [34] (98.70%). This is literature-reported data, not a reimplementation of the same protocol, and can only be considered indicative information, as it does not prove that ABCNN+HHWO performs better than the state of the art, since these experiments differ in data releases, preprocessing, classes, and splits (Section 4.8). The actual comparative experiments that would prove ABCNN+HHWO being the state-of-the-art method require reimplementation of at least random search, Bayesian optimization, HHO, WOA, and one attention-based architecture from recent literature on BraTS 2021 and Sartaj splits; otherwise, ABCNN+HHWO's performance can only be compared to, but not surpassed.

It can be seen from Section 4.2-4.8 that ABCNN+HHWO performs better than all baseline models (CNN, VGG19+TL, CNN-SVM, and ABCNN) on BraTS and Sartaj, with all differences being statistically significant (see Table 7). On top of that, the results of ablation studies (Table 6) show that pooling-based feature fusion branches provide additional information about spatial location. At the same time, the use of BN, GAP, label smoothing, and HHWO hyperparameter tuning contributes, but it is mostly dataset-specific (see Section 4.4). It aligns with the nature of the proposed ABCNN attention-inspired feature fusion module, which emphasizes spatial regions with local intensity patterns corresponding to tumor boundaries, necrosis, and edema. However, it was not confirmed by attention maps and Grad-CAM visualizations (see Section 4.8) and by the HHWO exploration-exploitation tradeoff, which prevents premature convergence. At the same time, elitism retains the best-found solutions. Nevertheless, one has to consider that the scale of the improvements obtained is contingent on the evaluation procedure itself – the separation of patients, an untouched independent dataset, repeated trials, and external validation are necessary to ensure that the improvements indeed generalize at the patient level. In this respect, the current findings provide evidence of the validity of the proposed approach relative to the current benchmark procedure but do not, by themselves, provide evidence of clinical utility – as explained in Section 4.8.

4.7 HHWO Reproducibility, Novelty Positioning, and Computational Cost

This part summarizes three issues concerning reproducibility and positioning brought up by reviewers:
The details of the HHWO searching process, together with the computational resources required (Table 9),
A qualitative comparison with other published Harris Hawks–Whale hybrid optimization methods (Table 10),
The computational cost of the models that are compared in this research (Table 11).

Table 9. HHWO Search Configuration and Reproducibility Detail

Parameter	BraTS 2021	Sartaj
Population size(agents)	7	12
Max. iterations	8	12
Patience(early-stopiterations)	4	not implemented
Fitness-evaluation epochs per candidate	8	6
Early-reject loss threshold	1.2	not implemented
Search dimensionality	5 (lr, weight decay, dropout, dense units, SE ratio)	4 (lr, weight decay, dropout, dense units)

Learning-rate search range	$[1\times10^{-4}, 3\times10^{-4}]$	$[1\times10^{-5}, 1\times10^{-2}]$ (log-uniform)
Weight-decay search range	$[1\times10^{-6}, 5\times10^{-4}]$	$[1\times10^{-6}, 1\times10^{-2}]$ (log-uniform)
Dropout search range	[0.20, 0.35]	[0.10, 0.50]
Dense-units search range	[256, 4096]	[128, 768]
Nominal search budget (agents × iterations)	agents × iterations = 56	agents × iterations = 144
Fitness evaluations actually consumed	not captured in notebook output (Sec. 4.8)	156 (reported directly in notebook output)
Random seed(s)	42 (GLOBAL_SEED; used for the data split, HHWO search, and augmentations)	42 (GLOBAL_SEED; used for the data split, HHWO search, and augmentations)
HHWO search wall-clock time	not captured in notebook output (Sec. 4.8)	6,002.6 sec (reported directly in notebook output)

All the values in the search configurations used for each dataset using the HHWO algorithm are provided in Table 9 below. While the nominal budget of agents×iterations for the search on the BraTS 2021 dataset is 56, Sartaj's nominal budget is 144, without implementing early stopping based on patience; however, the printed search summary in Sartaj's notebook shows that 156 fitness evaluations have been performed. The number of actual consumed fitness evaluations and wall-clock search time are not present in the notebook's output for BraTS' search; however, for Sartaj's search, these numbers are 156 and 6,002.6 seconds, respectively, as stated in the notebook. A global seed is fixed at GLOBAL_SEED = 42 for all random seeds used for data splitting, data augmentation, and the searches for both BraTS and Sartaj; therefore, no variability estimate is possible for these searches due to the use of a single seed. Evaluation count and search time differences for BraTS are left as open questions in Section 4.8.

Table 10. HHWO Compared with Previously Published Harris Hawks–Whale and Related Hybrid Metaheuristics — Novelty Positioning

Hybrid Method	Constituent Algorithms & Domain	Hybridization Mechanism (as reported)	Relation to Proposed HHWO
BWOAHHO [36]	Binary WOA + HHO; feature selection on 18 UCI benchmark datasets (KNN wrapper)	The transfer function converts continuous WOA/HHO position updates into binary decisions for a discrete search space.	Same two-parent algorithms, but combined for binary/discrete feature selection rather than continuous CNN-hyperparameter tuning; no CNN or medical-imaging application.
AO-HHO [26]	Aquila Optimizer + HHO; CNN hyperparameter tuning for brain-tumor MRI classification (4-class, 7,023 images; 87.34% accuracy reported)	Staged: Aquila Optimizer performs global exploration in early iterations, HHO performs local exploitation in later iterations	Directly comparable task (CNN hyperparameter tuning for brain tumor classification) but hybridizes HHO with a different partner algorithm using a staged, not concurrent, explore/exploit schedule; different dataset, architecture, and search space, so accuracy figures are not directly comparable.
HHO+WOA for IDPS feature selection [37]	HHO + WOA; feature selection for network intrusion detection (Random Forest classifier)	Not specified in the published abstract/available text	Same two-parent algorithms as HHWO, but applied to an unrelated domain with insufficient published mechanism detail for a rigorous comparison.

Proposed HHWO	HHO + WOA; CNN hyperparameter tuning (lr, weight decay, dropout, dense-layer size; BraTS adds an SE ratio) for brain tumor classification on BraTS 2021 and Sartaj	Concurrent, per-iteration blending: HHO's adaptive escape-energy parameter governs the explore/exploit switch, WOA's shrinking-encircling and logarithmic-spiral update supplies the exploitation step, with an added Lévy-flight perturbation	This work.

The HHWO is put into context in Table 10 alongside previous research on hybrids between Harris Hawks Optimization and the Whale Optimization Algorithm, or similar meta-heuristic algorithms identified through a non-systematic literature search rather than a systematic review. The table presents a qualitative comparison of the algorithms based on their methodologies, as reported in each study. It is not a reimplementation of the same protocol benchmark, which would be necessary to show that HHWO provides an actual advantage over these methods, especially BWOAHHO [36] and AO-HHO, for CNN-hyperparameter tuning in a domain-related study [26], which is considered necessary future work in Section 4.8.

Table 11. Computational Cost (GPU: Tesla P100-PCIE-16GB, as recorded in the notebooks' device logs)

Dataset	Model	Total Params	Train. Params	Infer. Time/Batch (s)	Training Time (s)
BraTS	ABCNN baseline	36,448,982	36,448,982	0.077	1,662.6
BraTS	Improved ABCNN (no HHWO)	5,122,206	5,119,006	0.235	n/a (Sec. 4.8)
BraTS	ABCNN + HHWO (final model)	5,096,052	5,092,852	0.185	3,461.8 (search time n/a)
Sartaj	ABCNN baseline	72,108,759	72,108,759	0.259	n/a (Sec. 4.8)
Sartaj	Improved ABCNN (no HHWO)	3,415,447	3,412,247	0.215	n/a (Sec. 4.8)
Sartaj	ABCNN + HHWO (final model)	2,970,139	2,966,939	0.185	n/a (Sec. 4.8)

Table 11 presents the computational cost data found in the notebooks. Not only is the HHWO guided tuning method more precise (Section 4.2), but it is also more compact than the ABCNN baseline on both datasets: the number of parameters is reduced from 36.4M to 5.1M on BraTS (about 86%) and from 72.1M to 3.0M on Sartaj (about 96%). Inference time per batch on BraTS is greater for the tuned version of the model than for the baseline (0.185 s versus 0.077 s), even with the reduction in the number of parameters, probably due to the particular dense layer width and SE ratio chosen by HHWO and not owing to the relation between the number of parameters and the inference time; on Sartaj, inference time is lower than the baseline one (0.185 s versus 0.259 s). Neither FLOPs nor maximum GPU memory usage was recorded in either notebook. For BraTS, the training time of the ABCNN+HHWO model (3,461.8 s) was recorded, but the total wall-clock time of the HHWO search process that selected this model was not recorded separately in the notebook output; similarly, training and search times for Sartaj were not recorded.

4.8 Limitations and Threats to Validity

The following sections explicitly lay out the limitations that the results from Sections 4.2-4.8 must be understood within, including patient-level data split, fairness of comparison, granularity of ablation analysis, HHWO implementation description (which has now been largely covered in Section 4.7), interpretability of the attention mechanism, statistical reporting, and what remains open.

Status of revision, briefly. The current revision addresses four of the major issues mentioned in the review: the ABCNN vs. ABCNN + HHWO headline comparison is now done with an entirely independent TEST set, not seen

by the HHWO during the search (Table 1, Section 3.4); the ablation-block terminology has been redefined as multi-branch feature fusion, and the title, abstract, and keywords have been modified accordingly (Section 2.2); the collapse of Sartaj TTA is no longer an enigma (Section 4.4); and finally, the configuration of HHWO's search, and the relationship of the algorithm to existing HHO-WOA hybrids have been clarified (Section 4.7). Three things are truly open and need additional experimental results outside of the scope of this revision: a re-split and re-analysis of patients for BraTS 2021 (the most important thing to open up, below); a comparative evaluation of optimizers separating the effect of HHWO from Random Search, Bayesian Optimization, HHO, and WOA with the same search budget; and a multiple seeds re-run of the ablation experiment (Table 6), with corrected leakage control, mean ± SD, 95% confidence interval, and Holm-Bonferroni adjusted p-values. The three things above, along with small gaps throughout the text below, are stated in the summary below without any further caveats.

Patient-level partitioning. In this version, a completely independent TEST partition is used for both datasets, is not included in the fitness calculation of HHWO and checkpointing, and is evaluated exactly once for each model (see Table 1); this resolves the confusion between the validation and test partitions mentioned in Section 4.5. For Sartaj, since patient identifiers are not available, patient-level partitioning was performed using perceptual-hash near-duplicate grouping, which has been shown to produce no overlap in groups among TRAIN, HHWO_VAL, and TEST. At the same time, this is only a partial replacement for patient-level independence (as it ensures there are no near duplicates and not whether there are no duplicates of the same patient that are not near-duplicates of one another), for BraTS 2021, patient identifiers were collected but not enforced at the time of splitting, with an audit revealing that most of the patients occur in multiple partitions (271 of 283 TRAIN patients occur in HHWO_VAL, and 271 in TEST). This is by far the most important question about the current manuscript. MRI slices from the same patients may exhibit similarities in anatomical structure, scanner settings, tumor characteristics, and preprocessing errors. This means that an image-level split will considerably boost the reported 97.69% accuracy on BraTS in comparison with a patient-disjoint evaluation of the method. The 97.69% number, thus, should be interpreted in terms of benchmark-level accuracy for image classification only, not generalization across patients and clinics, until the BraTS dataset is split into a patient-disjoint fashion (Patient -> TRAIN/HHWO_VAL/TEST, each patient in only one partition), and the entire experiment is performed again.

Controlled baseline comparison. The baselines (CNN, VGG19+TL, CNN-SVM, ABCNN) and the ABCNN+HHWO did not receive the same training: input size, optimizer used, learning-rate schedule, and number of epochs vary from one to another (Section 3.3-3.5). In addition, the ABCNN+HHWO uses BN, GAP, label smoothing, and HHWO-optimized hyperparameters, compared to its ABCNN baseline. In other words, the comparison done in Tables 4-5 can be considered as CNN/VGG19+TL/CNN-SVM/ABCNN vs. ABCNN+BN+GAP+label-smoothing+HHWO. Thus, the improvement observed cannot be directly attributed to HHWO. Nevertheless, the ablation study presented in Table 6 partly addresses this issue. A fully controlled comparison of the model performance would require training of ABCNN with default hyperparameters, Random Search, Bayesian Optimization, HHO, and WOA under the same search space, data split, training procedure, and computational resources – this goes beyond the current revision and represents the priority for future work, along with patient-level re-partitioning of the dataset (see Revision status, above).

Ablation granularity. In Table 6, the seven-variant ablation analysis quantifies gains due to the addition of the attention branch, BN+GAP, and HHWO fine-tuning as joint operations; it does not quantify gains due to variants that isolate label smoothing, BN, GAP, or TTA/top-K ensembling in isolation (the ensembling operation alone, but not any other operation, is isolated in the comparison between V6 and V7). The marginal contribution of label smoothing to performance, in particular, is not quantified. Significantly, in the context of the modified procedure, with independent TEST partitions, eight-round TTA led to substantial degradation of accuracy compared to a single forward pass (59.63% on Sartaj vs. 98.26%, -38.64 pp; 97.33% on BraTS vs. 97.69%, -0.36 pp) – showing that not only is the contribution of ensembling dataset-dependent in size, but it can also be strongly negative; TTA should thus be per-dataset validated, not assumed helpful, and the single forward pass configuration is used throughout all revised results in this manuscript. The rationale behind this is the differences between the TTA procedures for the two datasets. In BraTS, the TTA method averages the softmax outputs of the image and its horizontal flip in a standard inference setting (no dropout, Batch Normalization using its running-trained statistics). This augmentation method can be considered safe and appropriate for HGG and LGG classification, since horizontally flipping an image does not change glioma grade. The performance decrease observed was only a modest 0.36 pp drop, as reported above. On the other hand, in Sartaj, TTA was done by changing the TTA cell to one which makes eight iterations of random forward passes with the model called in training mode, without any transformation to the input images; moreover, this TTA implementation forces Batch Normalization to use per-batch statistics rather than the trained running statistics in inference mode – a known point of instability when the size of the batch does not represent well enough the overall distribution. In neither of these notebooks is there a 'top-K prediction averaging' technique implemented separately from the individual image TTA. The phrase used in the baseline description of the protocol in question probably refers to an unused ensemble on the top-5 HHWO hyperparameter settings

candidates (Section 3.4) for this manuscript.

HHWO implementation reporting. Population size, number of iterations, search space limits, fitness function evaluations, and the random seed are now documented for each problem in Table 9 (Section 4.7). There are two unknowns: First, the number of evaluations and the time taken by the BraTS search were not documented in this update due to a lack of console output capture in the search cell of the notebook (this run), whereas the Sartaj numbers (156 evaluations, 6,002.6 seconds) are available in Table 9. The search needs to be rerun with logging enabled and an open-source implementation made available so the search can be verified independently.

Interpretation through attention mechanisms. The designed feature-fusion block (Section 3.3.1) aims to highlight the saliency and homogeneity of local regions using fixed pooling descriptors and learned convolution-based descriptors, rather than computing an explicit, gated attention map. Thus, it is referred to as a multi-branch input-level feature-fusion mechanism rather than learned attention. The change is accompanied by the addition of an initial Grad-CAM, as well as attention-block saliency computation, on a few representatives correctly and incorrectly classified TEST images from each dataset (Fig. 4 for Sartaj, Fig. 5 for BraTS), only for the ABCNN+HHWO architecture (with no comparison to the ABCNN baseline). The heatmaps are localized to the tumor and its neighboring regions in correctly classified images. In contrast, they are localized to non-tumor regions (e.g., sinus cavity, skull margin) in incorrectly classified images. It is merely an example without performing any comprehensive analysis on the entire dataset. Furthermore, the comparison between ABCNN and ABCNN+HHWO is still missing. Both analyses are necessary for the validation of the claim made in Section 4.6. The lists for both are provided under future work in Section 5.

Fig. 4(a). Sartaj — Grad-CAM and attention-block-saliency maps, correctly classified TEST examples (glioma, meningioma, no-tumor rows; input / Grad-CAM / feature-fusion-block saliency columns)

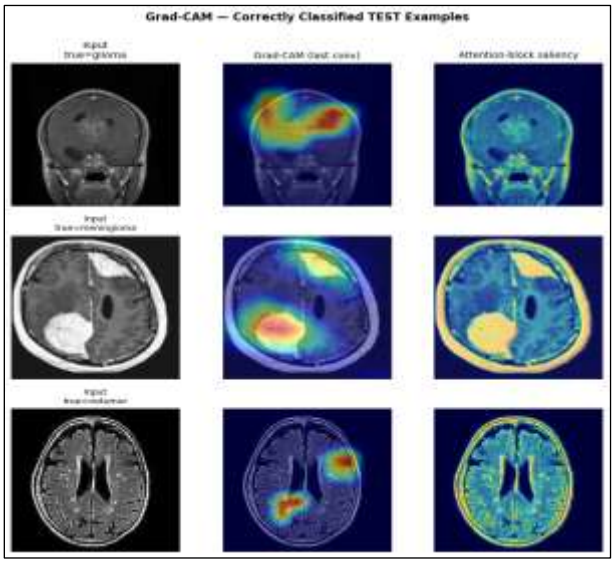


Fig. 4(b). Sartaj — Grad-CAM and attention-block-saliency maps, misclassified TEST examples.

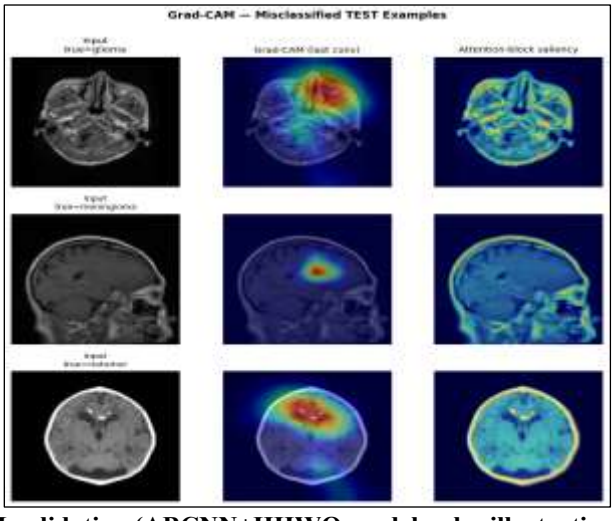


Fig. 4. Sartaj Grad-CAM validation (ABCNN+HHWO model only; illustrative sample, not a systematic evaluation — Section 4.8).

Fig. 5(a). BraTS — Grad-CAM and attention-block-saliency maps, correctly classified TEST examples (HGG, LGG rows; input MRI / Grad-CAM / feature-fusion-block map columns).

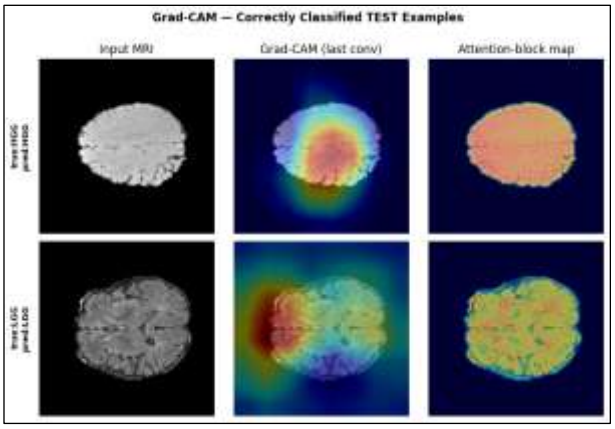


Fig. 5(b). BraTS — Grad-CAM and attention-block-saliency maps, misclassified TEST examples (HGG→LGG and LGG→HGG errors).

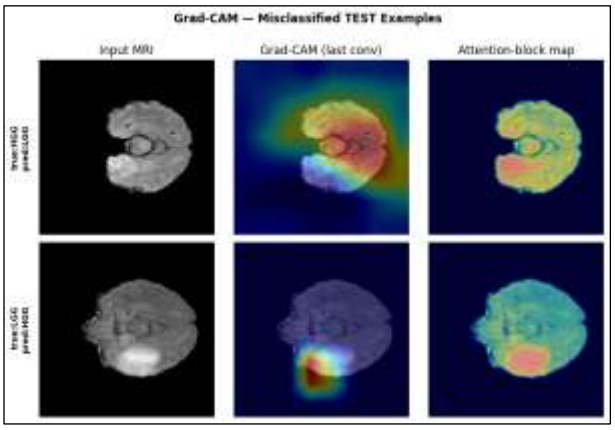


Fig. 5. BraTS Grad-CAM validation (ABCNN+HHWO model only; illustrative sample, not a systematic evaluation — Section 4.8).

State-of-the-art comparison. Table 8 provides a comparison of ABCNN+HHWO with respect to literature values and not the re-implementation; note that the different studies used varying datasets, preprocessing, and splitting methodologies, and even the year of BraTS data in some cases; the numbers are meant to give context but not necessarily to be part of an apples-to-apples comparison. The listed comparisons must be independently verified from their respective sources; a proper head-to-head comparison, re-implementing at least random search, Bayesian optimization, HHO, WOA, and one modern attention or transformer network on our BraTS 2021 and Sartaj splits, is desirable for future work.

Reporting statistics. The McNemar's test and Cohen's d statistics reported in Table 7 lack the number of discordant pairs, accuracy of predictions per run, and confidence intervals for six out of eight comparisons, and Cohen's d is based on only three runs. The experimental unit (image-level vs. per-run predictions) needs to be specified. The application of Holm-Bonferroni correction for two comparisons with exact p-values (ABCNN-vs-ABCNN+HHWO: BraTS $p=1.45\times10^{-9}$, Sartaj $p=0.0003$) would leave both significant at the family-wise level of significance $\alpha=0.05$; a full correction for all 8 comparisons in Table 7 cannot be performed because six of eight p-values in McNemar's test are presented as an upper bound $p<0.001$, not exact values. On the other hand, since BraTS's splits are not patient-disjoint (see Patient-level partitioning, above), then, about BraTS, the assumption of independence in the image-level McNemar test between paired observations is not met (correlated errors within the same patient will increase the value of the resultant χ^2, thus reducing the true p-value). Hence, the McNemar result for BraTS ($\chi^2=36.61$, $p<0.0001$) should be interpreted as an overly optimistic estimate of significance, and a patient-level clustered test (such as a cluster bootstrap or patient-level paired test), calculated using the split specified above, is necessary.

Data validation. The plots shown in Figs. 1-3 would have been better with higher resolution and larger font sizes for the axes and legends, given their print publication. The same applies to Figs. 4-5, introduced in this revision. In future research, there is a need to recompute Tables 4-5 and the BraTS confusion matrix in Fig. 2(e) using archived image-based predictions.

5. CONCLUSION AND FUTURE SCOPE

In this paper, we have proposed ABCNN+HHWO, a new methodology for classifying brain tumors using MRI, where a new input-level attention-inspired fusion technique using max pooling, average pooling, and convolution is combined with hybrid Harris hawks and whale optimization to tune learning rate, weight decay, dropout rate, and dense layer size.

Evaluation of the framework was conducted on two MRI image benchmark datasets: BraTS 2021 (14,734 images) and Sartaj (5,266 images). For the independently generated TEST partitions without leakage, ABCNN+HHWO provided classification accuracies of 97.69% and 98.26% compared to 94.66% and 95.05% by the ABCNN baseline on the same TEST partitions (McNemar's test: BraTS $p < 0.0001$, Sartaj $p = 0.0003$). ABCNN+HHWO also outperformed CNN, VGG19+TL, and CNN-SVM baselines on both datasets using the original, pre-correction procedure reported in Tables 4 and 5. But the baselines were not evaluated again on the updated TEST partitions. The ablation study (Table 6), also presented according to the original procedure and not re-run for this update, revealed that the max- and average-pooling branches and the BN+GAP/HHWO stages improved accuracy on both datasets. In contrast, the convolutional branch improved accuracy only on one dataset.

The results above show independent TEST set classification performance for both ABCNN and ABCNN+HHWO, improvements compared to baselines tested, and statistically significant pair-wise comparison of two benchmark MRI datasets, using the improved leakage-free partitioning and single run protocol as defined in Sections 3 and 4 for these two models (and using the original image level, single run procedure for the CNN, VGG19+TL, and CNN-SVM baselines). However, on their own, the results do not demonstrate clinical reliability or cross-institutional generalizability; patient-level partitioning still needs to be performed for BraTS (Section 4.8), and a multi-site study is needed before the results become clinically relevant.

In future work, there is a need for: (i) re-partitioning of the data at the patient level for BraTS 2021 (patient IDs are known but currently not being enforced during splitting, however, a test set that has been truly independent and has not involved HHWO (in this revision) has been created for both the datasets, and Sartaj uses perceptual hash near-duplicate clustering without any patient information); (ii) comparative study between HHWO and Random Search, Bayesian optimization, HHO alone, and WOA alone using an identical search space and computational budget; (iii) enhancing model interpretability through gradient-weighted class activation mapping (Grad-CAM) and SHapley Additive exPlanations (SHAP) in order to prove instead of assuming the spatial nature of the proposed feature fusion method – a representative Grad-CAM visualization per each dataset has been done in this revision as a preliminary test; (iv) repeated multi-seed training runs, reporting mean, standard deviation, and 95% confidence intervals alongside multiple-comparison-corrected significance tests; (v) calibration analysis has been performed for ABCNN+HHWO according to the new protocol (expected calibration error: 0.0332 BraTS, 0.0145 Sartaj; Brier score: 0.0418 BraTS, 0.0323 Sartaj), yet remains to be done for the other baselines and with more seeds, along with reliability diagrams; (vi) external, multi-institutional and prospective clinical evaluation, along with domain adaptation for multi-scanner MRIs and robustness test for varying image quality and unseen patient cohorts; (vii) extension to multi-modality 3D MRIs (T1, T1ce, T2, FLAIR) and WHO glioma grading (Grades II–IV); (viii) federated learning to train on multi-site datasets without sacrificing patient privacy; and (ix) reduction of computational cost of the HHWO search process itself. These research directions follow directly from the limitations outlined in Section 4.8.

References

1. H. Sung, J. Ferlay, R. L. Siegel, et al., “Global Cancer Statistics 2020: GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries,” CA: A Cancer Journal for Clinicians, vol. 71, no. 3, pp. 209–249, 2021.
2. R. Stupp, W. P. Mason, M. J. van den Bent, et al., “Radiotherapy plus Concomitant and Adjuvant Temozolomide for Glioblastoma,” New England Journal of Medicine, vol. 352, no. 10, pp. 987–996, 2005.
3. S. Bauer, R. Wiest, et al., “A Survey of MRI-Based Medical Image Analysis for Brain Tumor Studies,” Physics in Medicine & Biology, vol. 58, no. 13, pp. R97–R129, 2013.
4. S. Pereira, A. Pinto, V. Alves, and C. A. Silva, “Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images,” IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1240–1251, 2016.
5. H. H. Sultan, N. M. Salem, and W. Al-Atabany, “Multi-Classification of Brain Tumor Images Using Deep Neural Network,” IEEE Access, vol. 7, pp. 69215–69225, 2019.
6. R. Missaoui, W. Hechkel, W. Saadaoui, A. Helali, and M. Leo, "Advanced Deep Learning and Machine Learning Techniques for MRI Brain Tumor Analysis: A Review," Sensors, vol. 25, no. 9, p. 2746, 2025.
7. M. Rasheed, S. Iqbal, A. Jaffar, and S. Akram, "Advanced Deep Learning-Based Brain Tumor Classification Using a Novel Customized CNN and Optimized Residual Network," PLOS ONE, vol. 20, no. 10, 2025.
8. K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,”

in Proc. International Conference on Learning Representations (ICLR), 2015.
9. C. Szegedy, V. Vanhoucke, et al., “Rethinking the Inception Architecture for Computer Vision,” in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2818–2826, 2016.
10. K. He, X. Zhang, et al., “Deep Residual Learning for Image Recognition,” in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.
11. M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” in Proc. International Conference on Machine Learning (ICML), pp. 6105–6114, 2019.
12. J. Amin, M. Sharif, M. Raza, et al., “Brain Tumor Detection: A Long Short-Term Memory (LSTM)-Based Learning Model,” Neural Computing and Applications, vol. 32, pp. 15965–15973, 2020.
13. O. Oktay, J. Schlemper, L. L. Folgoc, et al., “Attention U-Net: Learning Where to Look for the Pancreas,” in Proc. Medical Imaging with Deep Learning (MIDL), 2018.
14. F. Wang, M. Jiang, C. Qian, et al., “Residual Attention Network for Image Classification,” in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3156–3164, 2017.
15. S. Woo, J. Park, et al., “CBAM: Convolutional Block Attention Module,” in Proc. European Conference on Computer Vision (ECCV), pp. 3–19, 2018.
16. O. O. Oladimeji and A. O. J. Ibitoye, “Brain Tumor Classification Using ResNet50-Convolutional Block Attention Module,” Applied Computing and Informatics, vol. 22, no. 3-4, pp. 249–261, 2026.
17. J. Hu, L. Shen, and G. Sun, “Squeeze-and-Excitation Networks,” in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7132–7141, 2018.
18. E. Zarenia, A. Akhlaghi Far, and K. Rezaee, "Automated Multi-Class MRI Brain Tumor Classification and Segmentation Using Deformable Attention and Saliency Mapping," Scientific Reports, vol. 15, 2025.
19. A. J. Aiya, N. Wani, M. Ramani, A. Kumar, S. Pant, K. Kotecha, and A. Kulkarni, "Optimized Deep Learning for Brain Tumor Detection: A Hybrid Approach with Attention Mechanisms and Clinical Explainability," Scientific Reports, vol. 15, 2025.
20. V. R. Srinivas and R. Parvathi, "Explainable AI-Driven MRI-Based Brain Tumor Classification: A Novel Deep Learning Approach," Frontiers in Artificial Intelligence, vol. 8, 2026.
21. A. A. Heidari, S. Mirjalili, H. Faris, et al., “Harris Hawks Optimization: Algorithm and Applications,” Future Generation Computer Systems, vol. 97, pp. 849–872, 2019.
22. S. Mirjalili and A. Lewis, “The Whale Optimization Algorithm,” Advances in Engineering Software, vol. 95, pp. 51–67, 2016.
23. K. Hussain, M. N. Mohd Salleh, et al., “Metaheuristic Research: A Comprehensive Survey,” Artificial Intelligence Review, vol. 52, pp. 2191–2233, 2019.
24. T. Thaher, A. A. Heidari, M. Mafarja, et al., “Binary Harris Hawks Optimizer for High-Dimensional, Low Sample Size Feature Selection,” in Evolutionary Machine Learning Techniques, Springer, Singapore, pp. 251–272, 2020.
25. I. Aljarah, H. Faris, and S. Mirjalili, “Optimizing Connection Weights in Neural Networks Using the Whale Optimization Algorithm,” Soft Computing, vol. 22, no. 1, pp. 1–15, 2018.
26. M. Kumar, N. Mohd, G. Shivam, A. Goyal, D. Parashar, and R. Khan, "Hybrid Aquila Optimizer–Harris Hawks Optimization for CNN Hyperparameter Tuning in Brain Tumor Classification," Scientific Reports, vol. 16, 2026.
27. A. Abdollahi Dehkordi, M. Neshat, A. Khosravian, M. Thilakaratne, A. S. Sadiq, and S. Mirjalili, "Lightweight Convolutional Neural Networks Using Nonlinear Lévy Chaotic Moth Flame Optimisation for Brain Tumour Classification via Efficient Hyperparameter Tuning," Scientific Reports, vol. 15, 2025.
28. Q. McNemar, “Note on the Sampling Error of the Difference Between Correlated Proportions or Percentages,” Psychometrika, vol. 12, no. 2, pp. 153–157, 1947.
29. J. Cohen, Statistical Power Analysis for the Behavioral Sciences, 2nd ed., Lawrence Erlbaum Associates, Hillsdale, NJ, 1988.
30. P. Afshar, K. N. Plataniotis, and A. Mohammadi, “Capsule Networks for Brain Tumor Classification Based on MRI Images and Coarse Tumor Boundaries,” in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1368–1372, 2019.
31. M. M. Badža and M. Č. Barjaktarović, “Classification of Brain Tumors from MRI Images Using a Convolutional Neural Network,” Applied Sciences, vol. 10, no. 6, p. 1999, 2020.
32. A. Harshavardhan, V. C. S. Rao, Y. M. Reddy, S. R. Polamuri, B. Jamalpur, and V. L. Reddy, "Hybrid Quantum–Classical Learning for MRI-Based Brain Tumour Diagnosis," Discover Computing, vol. 29, art. 269, 2026.
33. M. A. Khan, I. Ashraf, M. Alhaisoni, et al., “Multimodal Brain Tumor Classification Using Deep Learning and Robust Feature Selection: A Machine Learning Application for Radiologists,” Diagnostics, vol. 10, no. 8, p. 565, 2020.
34. C. Sankari, V. Jamuna, and A. R. Kavitha, "Hierarchical multi-scale vision transformer model for

accurate detection and classification of brain tumors in MRI-based medical imaging," Scientific Reports, vol. 15, art. 38275, 2025.
35. L. Ke, G. Hu, M. Zhao, Z. Liu, Z. Lv, and Y. Yang, "Brain tumor classification from MRI images using a multi-scale channel attention CNN integrated with SVM," Scientific Reports, vol. 16, art. 6297, 2026.
36. R. Alwajih, S. J. Abdulkadir, H. Al Hussian, N. Aziz, Q. Al-Tashi, S. Mirjalili, and A. Alqushaibi, "Hybrid Binary Whale with Harris Hawks for Feature Selection," Neural Computing and Applications, vol. 34, pp. 19377–19395, 2022.
37. M. M. Abualhaj, S. N. Al-Khatib, M. Al Zyoud, I. Qaddara, and M. Anbar, "Enhancing Intrusion Detection System Performance Using a Hybrid of Harris Hawks and Whale Optimization Algorithms," Engineering, Technology & Applied Science Research, vol. 15, no. 4, pp. 24354–24361, 2025.