IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 43, No. 2, April, 2026, pp. 955-969
Journal homepage: http://iieta.org/journals/ts

A Novel Deep Learning Framework for Accurate Classification and Survival Rate Prediction of Brain Tumors

Kalaiselvi S.[1*], Jesudas Thangaraju[2]

[1] Computer Science and Information Engineering, Anna University, Chennai 600025, India
[2] Department of Mechatronics Engineering, Mahendra Engineering College (Autonomous), Namakkal 637503, India

Corresponding Author Email: kalaiselvi080484@gmail.com

https://doi.org/10.18280/ts.430232

Received: 13 November 2025
Revised: 15 April 2026
Accepted: 23 April 2026
Available online: 30 April 2026

Keywords:
brain tumor, Convolutional Long Short-Term Memory, Deep Neural Network, Secretary Bird Optimization, survival rate prediction, U-Net, image segmentation

ABSTRACT

One of the major health issues facing the world today is brain tumors, which have a higher death rate. Making appropriate treatment options for patients, which enhance their health and longevity, depends on an accurate identification of brain tumors. The existing works focused mainly on binary classification, lacking enough information for medical professionals to begin the treatment process. This study works on developing an automatic survival rate prediction mechanism for brain tumor patients by leveraging the efficiency of deep learning (DL) and meta-heuristic optimization algorithms. Unlike the conventional approaches, the proposed methodology uses multi-faceted patient data, including demographic details and disease features (type and stage of tumor) for survival rate prediction. Initially, the multifaceted patient data was collected and pre-processed to enhance its quality for further processing. The region of interest (ROI) was then separated from the pre-processed images using the U-Net technique for image segmentation. Consequently, a Deep Neural Network (DNN) is used for extracting significant features related to tumor types. The DNN results and the pre-processed demographic details of the patients are fed into the proposed Co-LSTM-SBO for survival rate prediction. The proposed Co-LSTM-SBO integrates the efficiency of Convolutional Long Short-Term Memory (Co-LSTM) and Secretary Bird Optimization (SBO), where the Co-LSTM component learns the hierarchical feature representations and patterns within the data and makes survival rate prediction. On the other hand, the SBO fine-tunes the hyperparameters of Co-LSTM efficiently, thereby improving model training and enhancing its computational efficiency. Python was used to implement the presented work, and its prediction accuracy, F1 score, precision, and sensitivity were compared to those of previous studies.

1. INTRODUCTION

Cells across the body proliferate abnormally and uncontrollably in cancer, a medical disorder. There are over 200 types of cancer, such as blood cancer, lung cancer, skin cancer, breast cancer, and lymphoma [1]. According to the World Health Organization (WHO), around 9.6 million people globally died from cancer in 2018. Brain tumors represent a common and perilous ailment that diminishes life expectancy across all demographics [2]. Their identification can be made based on the existence of an unusual aggregation of brain cells, which typically originates from the rapid and erratic multiplication of brain tissues. Depending on the nature, size, location, and availability of appropriate therapies, brain tumors can have a substantial impact on a person's life expectancy [3]. Being knowledgeable about risk factors and early screening methods is essential to fighting cancer effectively. To lower the incidence of cancer and enhance general health outcomes, adopting a healthy lifestyle, getting frequent checkups, and adhering to suggested screening protocols can be extremely important [4]. For individualized information and advice on cancer prevention and management, it is always advisable to speak with medical professionals. The body's abnormal cells don't stay the same; they proliferate quickly and spread, which causes tumors to form [5]. Brain tumors are categorized as either benign (noncancerous) or malignant (cancerous). If cancerous tumors grow quickly into other regions of the brain, the patient's condition may worsen. In the brain, new cells naturally replace healthy cells that have aged or been damaged [6]. However, if old and damaged cells are not immediately removed, a tumor could develop, which would be quite alarming. Early detection and treatment are essential for increasing survival and therapeutic success because tumors typically spread to other tissues [7].

Brain cancers can be detected with the help of medical imaging, which is essential for this purpose. Magnetic resonance imaging (MRI) and computed tomography (CT) scans are both valuable methods for delivering essential insights into the existence and dimensions of atypical brain tissues, aiding in prompt follow-up actions, and deepening our comprehension of the tumor's status [8]. MRI employs radio waves and a strong magnetic field to provide accurate details about the soft tissues and internal structure of the human body.

It generates various kinds of images, such as FLAIR, T1, T1 contrast-enhanced, Proton Density (PD), and T2-weighted high contrast grayscale images [9]. These images assist greatly in the identification and classification of brain tumors based on their size, shape, and location. Brain tumors are generally classified into two types: primary and secondary. Primary tumors develop from brain cells, while secondary tumors metastasize from other body parts to the brain [10].

Brain tumors are a serious health risk and can have detrimental effects on patients. Brain tumors must be accurately and quickly diagnosed to support successful treatment plans [11]. The precision and effectiveness of the traditional methods for dividing, categorizing, and forecasting the hazards related to brain tumors have been limited. Recently, DL-based models have become increasingly effective instruments for analyzing medical imaging [12]. These models have the potential to greatly increase brain tumor diagnosis efficiency and accuracy. However, several obstacles prevent them from being used effectively in therapeutic settings. Data availability and effectiveness, computational difficulty, intramodality modifications, simulation generalization, overfitting, accessibility, temporal dynamics, annotations, labeling problems, compatibility with clinical processes, and ethical issues such as biases and data privacy are some of these difficulties [13, 14]. In this context, a sophisticated DL model that can effectively and accurately manage brain tumor segmentation, classification, and risk prediction is desperately needed. The model needs to handle high-dimensional data, be able to capture complex tumor shapes, and be resilient to changes in the data [15]. It should also be easily scalable, interpretable, and incorporated into healthcare workflows. For people who have brain tumors to receive prompt and efficient care, certain needs must be met.

There are several reasons for conducting this study. First, since brain tumors can greatly improve the patient's outlook and quality of life, it is vital to improve their early diagnosis and precise categorization. Second, to treat a multiclass variety of cancers, more advanced models must be developed. Furthermore, to make these models more practical and accessible in clinical settings, their computational costs must be decreased. Last but not least, forecasting patient results and survival rates can be quite helpful for individualized treatment programs. A DL architecture specifically designed for brain tumor diagnosis is shown in this article. This method not only detects and segments cancers, but it also classifies them and predicts patient survival rates. This comprehensive approach could be crucial to increasing the potential of automated diagnostic tools and ensuring that medical professionals obtain accurate and timely information that is necessary for patient care. The objective of this research is to create a new survival rate prediction model for brain tumor patients that will be more effective than the binary classification. The key contributions of this work are as follows:

Unlike existing brain tumor analysis methods that primarily focus on image-based classification, this study presents a prognostic-oriented survival prediction framework. The main contributions are as follows:

•The integration of patient demographic information, tumor-specific clinical attributes, and deep imaging features for brain tumor survival rate prediction.

•To employ the U-Net architecture for accurate tumor region-of-interest segmentation, ensuring reliable spatial representation for downstream prognostic modeling.

•The extraction of discriminative and high-level tumor feature representations using a Deep Neural Network (DNN) trained on segmented medical images.

•To develop a Co-LSTM–SBO–based survival prediction framework, where Co-LSTM captures hierarchical and sequential feature dependencies and Secretary Bird Optimization (SBO) optimizes hyperparameters for improved training efficiency and generalization.

•For validating the effectiveness of the proposed framework through a comprehensive evaluation using standard prognostic performance metrics.

•The ultimate goal is to help the medical field with an automatic, highly accurate survival rate prediction helper tool that aids in treatment planning and leads to improved patient outcomes and survival.

The ultimate goal is to help the medical field with an automatic, highly accurate survival rate prediction helper tool that aids in treatment planning and leads to improved patient outcomes and survival.

The literature review is covered in detail in section 2, the issue description in section 3, the suggested approach in section 4, the results and discussion in section 5, and a conclusion in section 6.

2. RELATED WORKS

A brand-new DL system created specifically for the diagnosis of brain tumors is presented by Zaitoon and Syed [16]. The proposed method begins with the gathering of data, which is subsequently pre-processed to improve the data using a Convolutional Normalized Mean Filter (CNMF). This readies the data for multi-class classification using the new Deep Belief–Transfer Convolutional Neural Network (DBT-CNN) classifier method. The RU-Net2+ model, which generates segmented zones to which a Cox model is applied to extract attributes, is employed for precise tumor delineation. These collected traits are essential for predicting the patients' survival rate in the final stage using a logistic regression approach.

Babu Vimala et al. [17] categorized three types of brain tumors—meningioma, glioma, and pituitary tumors—using EfficientNet models and a transfer learning–based fine-tuning technique. The pre-trained EfficientNet system is refined using a two-step procedure. First, use the ImageNet dataset's weights to initialize the framework. Then, add additional layers like top layers and a fully connected layer to facilitate tumor classification. Utilize Grad-CAM visualization to investigate the attention maps produced by the top model that accurately identify tumor regions in brain images, in order to gain a deeper understanding of the model's decision-making process.

Othman et al. [18] created a hybrid DL framework that uses two distinct classifiers and allows for decision-making using data from several sources. Utilizing various omics data from the Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) dataset is anticipated to enhance the precision of patient survival estimations in comparison to relying on a single data modality. A convolutional neural network (CNN) model extracts features. As classifiers, GRU and LSTM are utilized.

A potent new hybrid approach is put out by Celik and Inik [19] to accurately classify brain cancers in numerous ways. This approach employs a specialized CNN framework for feature extraction and utilizes machine learning methods for

feature classification. Moreover, nine advanced CNN models are employed to evaluate CNN performance. The optimal hyperparameter values for machine learning algorithms are determined using the Bayesian optimization procedure. As per the outcomes of the experimental investigations, the proposed hybrid model achieved a mean classification accuracy of 97.15% and recall, precision, and F1-score values of 97%.

Three very successful machine learning classifiers with deep convolutional learned features are used by Khan et al. [20] to classify medical images. The MRI images of the three types of brain tumors, which are utilized to assess the efficacy of the automated approach, are available as an open dataset on Figshare. Brain MRI data are analyzed using the DenseNet169 framework to extract features. Three multiclass machine learning classifiers (Random Forest (RF), Support Vector Machine (SVM), and XGBoost) are fed extracted features to improve performance. The suggested model outperformed the cutting-edge approach with an overall classification accuracy of 95.10%.

To improve early detection, Srinivasan et al. [21] created a deep CNN and offered three different CNN models made for various categorization tasks. The first CNN algorithm has a remarkable 99.53% accuracy rate in identifying brain tumors. With an accuracy of 93.81%, the second CNN model successfully divides brain tumors into five categories. Furthermore, the third CNN model has a 98.56% accuracy rate in classifying brain tumors into different classifications. To ensure optimal performance, all relevant CNN model hyperparameters are automatically adjusted using a grid search optimization technique.

An improved AI-based design for better brain tumor categorization is presented by Dixon et al. [22]. Furthermore, it offers a hybrid approach that creates an ensemble classifier by combining DNN with vision transformers (ViT), resulting in a more dependable framework for brain tumor classification. After preprocessing and data normalization, three different kinds of information-rich features produced from MRIs are extracted as part of the analytical pipeline. The final classification of brain MRIs is achieved by weighting, fusing, and feeding the retrieved characteristics into a machine-learning classifier. Using a variety of measures, the suggested weighted ensemble design has been assessed on locally obtained and publicly accessible brain MRIs of four classes. ShallowMRI, a simple CNN based on the classification of types of brain tumors in MRI, was developed by Khan et al. [23]. The model has a novel attention system that enhances its capacity to attend to pertinent characteristics in proper classification. The best contours are determined by erosion and dilation to take off the unnecessary areas. Also, Local Binary Patterns (LBP) complement image preprocessing by lowering the computational expenses and maximizing the feature extraction. Using three benchmark datasets, Kaggle Multiclass, BR35H Binary, and Histogram Binary, the effectiveness of ShallowMRI is evaluated against pre-trained models and current techniques.

Hekmatet et al. [24] presented an approach, which is the Attention-Fused MobileNet-LSTM model, which is designed to enhance the quality of the classification. Through the combination of the functionality of two highly established pre-trained CNN models called MobileNetV1 and MobileNetV2, we have created a dependable model. We solve the CNN training problems, such as the need to make extensive computations, the inability to identify finer details in the picture, and reduced performance because of noise or useless data. The use of a transformer to assist in feature extraction and the use of gamma correction, subsequently, then feature fusion, helps to address long-range relationships between consecutive features. But, the suggested model, based on LSTM recurrent neural networks (RNN) to directly predict class labels, may not be easy to comprehend the intermediate characteristics of the feature fusion, and this restricts the transparency of the model.

The existing techniques' advantages and disadvantages are detailed in Table 1.

Table 1. Summary of the existing techniques

Author(s)	Year	Technique	Advantage	Disadvantage
Zaitoon and Syed [16]	2023	DBT-CNN classifier with RU-Net2+ and logistic regression	High classification accuracy (HGG: 99.51%, LGG: 99.28%), detailed segmentation, and survival rate prediction	Complex architecture, high computational cost
Babu Vimala et al. [17]	2023	Transfer learning with EfficientNet (B0-B4) and Grad-CAM	High accuracy (EfficientNetB2: 99.06%), interpretability with Grad-CAM	Limited analysis of the EfficientNets family, generalizability concerns
Othman et al. [18]	2023	Hybrid model with multi-omics data, CNN for feature extraction	High prediction accuracy with decision fusion (98.0%)	Limited to breast cancer, potential overfitting due to multi-source data
Celik and Inik [19]	2024	Custom CNN for feature extraction with optimized ML classifiers	High accuracy (97.15%), effective feature extraction	Time-intensive (e.g., InceptionV3 took 370 min), higher computational load
Khan et al. [20]	2024	DenseNet169 for feature extraction, ML classifiers for diagnosis	Robust classification (95.10%) with limited training data	Lower accuracy compared to recent DL-only models
Srinivasan et al. [21]	2024	Custom CNN models with grid search for hyperparameter tuning	High detection accuracy (99.53%), multi-class capability	Moderate performance on multi-class (93.81%) compared to detection
Dixon et al. [22]	2024	Vision Transformer (ViT) with CNN ensemble for feature fusion	Enhanced robustness, improved classification with feature diversity	Higher complexity, sensitivity to hyperparameter selection

3. PROBLEM STATEMENT AND MOTIVATION

Early and correct diagnosis of brain tumors is still a major problem in medical imaging. The current diagnostic techniques include histopathological analysis and manual analysis of MRI images, which are slow, invasive, and prone to errors. Such limitations underscore the importance of developing computer-aided diagnostic (CAD) systems that are

capable of accurately classifying, segmenting, and predicting the survival rate of brain tumors [23]. Even though many DL and hybrid models have been suggested, some issues remain concerning the trade-off between the performance and complexity of the method and its ability to work with a small number of samples. It is crucial to establish an effective, accurate, and high-quality automated system based on DL algorithms to increase the diagnostic accuracy, provide proper treatment, and, as a result, increase the patient's quality of life [24].

The reason for this study is rooted in the fact that brain tumors are among the most critical health challenges in the world. The accurate and timely diagnosis of the disease is vital in treatment since it determines the outcomes of the patients. Today's diagnostic procedures are mostly done through physical examination, which raises the question of inter-observer variability and time. Combining DL and machine learning in diagnosis offers a good opportunity to meet these challenges. These technologies can help medical professionals by automating tasks like tumor classification, segmentation, and survival prediction to minimize diagnostic errors and facilitate decision-making. Moreover, the dynamic nature of sophisticated architectures such as CNNs and ViT, and the integration of both in hybrid models, present a chance to utilize the current and latest techniques to improve diagnostic performance [25]. This motivation leads to the search for a model that is accurate, efficient, and computationally efficient in aiding the design of effective CAD systems that could transform the treatment of brain tumors.

4. PROPOSED METHODOLOGY

This study works on developing an automatic survival rate prediction mechanism for brain tumor patients by leveraging the efficiency of DL and meta-heuristic optimization algorithms. Unlike the conventional approaches, the proposed methodology uses multi-faceted patient data, including demographic details and disease features (type and stage of tumor) for survival rate prediction. Initially, the multifaceted patient data was collected and pre-processed to enhance its quality for further processing. Then, the U-Net algorithm was applied for image segmentation, which isolates the region of interest (ROI) from the pre-processed images. Consequently, DNN is used for extracting deep features for classifying tumor types. The DNN results and the pre-processed demographic details of the patients are fed into the proposed Co-LSTM-SBO for survival rate prediction. The proposed Co-LSTM-SBO integrates the efficiency of Co-LSTM and SBO, where the Co-LSTM component learns the hierarchical feature representations and patterns within the data and makes survival rate predictions. Co-LSTM and SBO were chosen based on the characteristics of the brain tumor survival prediction problem, which is characterized by heterogeneous data with complicated spatial and contextual relationships. The traditional CNN-based models are useful to capture spatial tumor features but fail to take into consideration inter-feature and sequential dependencies, whereas traditional LSTM models can capture dependencies at the expense of missing the spatial information. Co-LSTM combines convolutional algorithms in the framework of recurrence, which allows both maintaining spatial locality and capturing hierarchical dependencies between features, which is appropriate in prognostic analysis.

Co-LSTM performance is also very sensitive to the choice of hyperparameters, and manual tuning can result in poor convergence. SBO was thus used as an adaptive meta-heuristic optimizer to effectively search the high-dimensional space of hyperparameters and enhance training stability and generalization. The combination of Co-LSTM and SBO provides the high efficiency of spatial-temporal feature learning and the optimal work of the model in predicting the survival rate. On the other hand, the SBO fine-tunes the hyperparameters of Co-LSTM efficiently, thereby improving model training and enhancing its computational efficiency. The overall process of the developed model is shown in Figure 1.

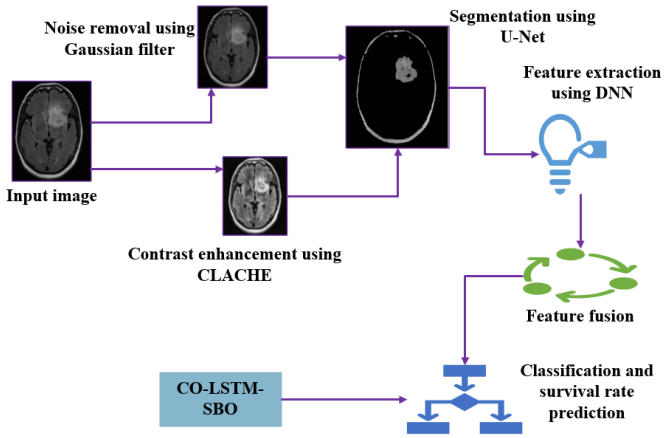

Figure 1. Process of the developed technique

4.1 Dataset description

The research utilized the publicly accessible brain tumor classification dataset [26] found on Kaggle, consisting of 3,264 MRI images categorized into four groups: pituitary (930 images), meningioma (708 images), glioma (1,426 images), and no tumor (200 images). The data will be sorted into training and testing folders.

It used stratified splitting to split the data into 80% training, and 20% test sections to have balance and reproducibility by maintaining the proportion of classes in each section. All images were preprocessed and augmented as a standard to balance out inputs and overcome class imbalance.

Although the dataset is large enough to train deep-learning-based models, the few samples assigned to each of the classes could result in bias and narrow generalization. The sampled images taken for the training and testing process are detailed in Table 2.

Table 2. Dataset details

Class	Testing (20%)	Training (80%)	Total
Glioma	285	1141	1426
Meningioma	142	566	708
Pituitary	186	744	930
No Tumor	40	160	200
Total	**653**	**2611**	**3264**

4.2 Preprocessing

Before segmentation of the tumors using U-Net, the brain MRI images obtained were subjected to a sequence of preprocessing procedures that enhance the quality of the images and guarantee credible extraction of the features. To

normalize spatial dimensions, first, all the images were resized to a standard size. To eliminate random intensity and sensor artifact variations, a median filtering technique was used to reduce noise.

Histogram equalization was used to improve the contrast of the image to help in enhancing the visibility of tumor areas, especially in low-intensity or low-illumination images. Normalization of intensity was then done to bring pixel values to a normalized range so that there is consistency in the dataset, and this makes model training stable. Lastly, rotation, flipping, and scaling were used to augment the data to improve the diversity of the dataset and to enhance the strength of the segmentation and survival prediction models.

Such preprocessing steps guarantee that the U-Net model is fed with good-quality and standardized inputs, thus enhancing the segmentation of the tumor region-of-interest and feature extraction to perform downstream prognostic analysis.

•Noise removed using a Gaussian filter

When noise is modeled as additive white Gaussian noise (AWGN), each pixel in the image differs from its initial values according to the Gaussian curve. In other words, for every pixel in an image that has an intensity value $\vec{o}_{i,j}(1 \leq i \leq m, 1 \leq j \leq n)$ for an $m \times n$ image, the noisy image's matching pixel $\vec{z}_{i,j}$is provided by Eq. (1),

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - \vec{z}_{i,j} = \vec{o}_{i,j} + \vec{g}_{i,j} 4ac}}{2a} \quad (1)$$

where, each value of noise $\vec{g}$ is taken from a Gaussian distribution with zero mean. To determine thresholds, filter window size, and other parameters for many Gaussian noise removal approaches, it is necessary to understand the standard deviation, which is a measure of the degree of corruption. The conventional standard deviation formula for estimating Gaussian noise (δ_{gn}) is provided by Eq. (2)

$$\delta_{gn} = \sqrt{\frac{\pi}{2}} \frac{1}{6m \times n} \sum_{i,j=1}^{m,n} \left| (\vec{z} * ms)_{i,j} \right| \quad (2)$$

Let, $*$ be the convolution operation, and ms be the mask image.

•Contrast enhancement using CLACHE

To accomplish equalization, the function is modified using the Contrast-Limited Adaptive Histogram (CLACHE) [26]. By applying CLACHE to the values, it improves the grayscale image's contrast. It works on small data sections (tiles) instead of the full image, in contrast to histogram equalization. To make the histogram of every final region roughly resemble the given histogram (uniform distribution by default), the contrast of all tiles is increased. Limiting the contrast enhancement can help prevent the image's potential noise from being amplified. The intensity of the output $i'(a, b)$ according to Eq. (3) for a pixel in the improved image.

$$i'(a, b) = int\, p\left(R[l-1] \times cdf_{i,j}^{nm}\left(i(a, b)\right)\right) \quad (3)$$

Let, $int\, p$ () be the Tile boundaries are blended using a bilinear interpolation technique, $cdf_{i,j}^{nm}$ be the CDF normalized for the corresponding tile $t_{i,j}$, $l$be thehighest value of intensity, and Rbe the round. The range's pixel intensity frequency is shown by the histogram $[l-1]$.

•Resizing and normalization

Resizing: Resizing is the process of making an image a specific size. It is essential for normalizing input data so that images that are input into DL models have the same size. Reducing an image from 1024 × 1024 to 256 × 256 is necessary to make all the inputs to a neural network of the same size, which makes the model and training process easier.

Min-max normalization: In order to improve the speed and stability of neural network training, the process of normalization is used to adjust the pixel values in an image to a common scale. Min-max normalization standardizes the pixel intensities within a fixed range, which could be [0,1] or [-1, 1]. The transformation is given by Eq. (4).

$$\vec{x}_{nrm} = \frac{\vec{x} - min(\vec{x})}{max(\vec{x}) - min(\vec{x})} \quad (4)$$

Let, $\vec{x}_{nrm}$ be the pixel-normalized value $\vec{x}$, $\vec{x}$ be the image's original pixel value, $min(\vec{x})$ be the minimum pixel value throughout the whole image, and $max(\vec{x})$ be the maximum value of each pixel in the whole image. The pre-processed image is shown in Figure 2.

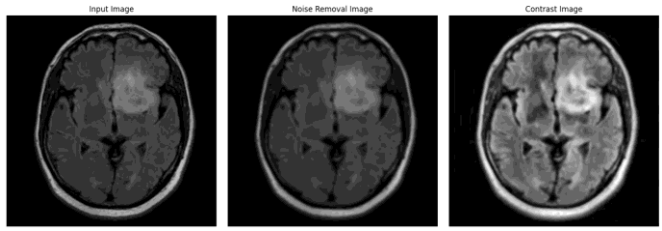

Figure 2. Pre-processed image

4.3 Segmentation

The ROI is separated from the pre-processed data using the U-Net technique [27] for image segmentation. There are two components to the U-net network: The expanding path, which employs a standard CNN design, is the first. A max-pooling layer and a ReLU activation unit come after two consecutive 3 × 3 convolutions in each block of the contracting path. This arrangement is repeated multiple times. The expanding path, the second component of U-net, is distinctive because it employs 2 × 2 up-convolution to sample the feature map at each level. Subsequently, the feature map that has been up-sampled is minimized and then concatenated with the feature map from the matching layer located in the expanding path. ReLU activation and two consecutive 3 × 3 convolutions come next. To create the segmented image and limit the feature map to the necessary number of channels, a final 1 × 1 convolution is done. The complete U-net architecture is shown in Figure 3.

Eq. (5) provides the network's energy function.

$$e_f = \sum W(s)\, log\left(\bar{P}e_{(s)}(s)\right) \quad (5)$$

Let, $\bar{P}e$ be the pixel-wise SoftMax function that is applied over the finished feature map, defined as Eq. (6).

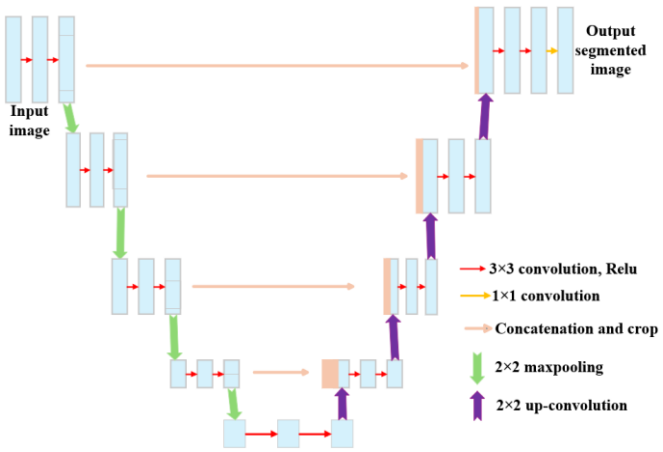

Figure 3. U-Net architecture

$$\bar{P}e = Exp(x_e(s))/\sum_{e'=1}^{E} Exp(x_e(s)') \quad (6)$$

Let x_e be the channel activation in E. Since pixel characteristics near the margins provide the least contextual information, they must be eliminated, necessitating cropping. This creates a U-shaped network and, more significantly, spreads contextual information throughout the network, enabling it to use information from a larger overlapping area to segment items in a given area. The segmented image is shown in Figure 4.

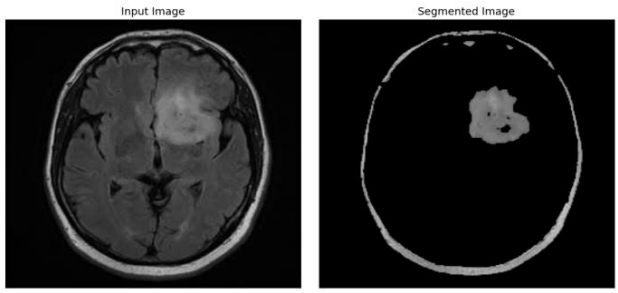

Figure 4. Segmented image

4.4 Feature extraction

The researchers found that the DNN [28] is the best method for extracting important features used for identifying the type of tumor when working with the image dataset. The CNN operates in that it uses photos from various datasets as input, and then its many layers extract features. Since the human brain's virtual cortex (VC) is responsible for learning, the VC has an impact on DL. The VC, a part of the human brain, handles all queries from the external environment. Since each layer pulls data gathered from the layers before it and they require each other to be blended, the final layers are combined, and the photos are categorized. Similar to this, CNN contains numerous filters that each extract data about the image, such as various shapes (round, vertical, and horizontal), edges, and measurements. The DNN network uses numerous filters and layers to gather features with data from images, as seen in Figure 3. After that, it advances to the classification stage of the network, where they are categorized according to the parameters. DNN Layers are covered in detail in the items that follow:

Input layer: Grayscale or RGB input images may be placed beneath this layer. Pixels ranging from 0 to 255 make up each image. Before being passed to the following layer of the model, the range between 0 and 1 must be transformed.

Convolution Layer: The input images must first undergo filtering to obtain or detect their features under this layer. The second is that to produce a feature map that helps with input image classification, a filter must be applied to the images several times.

Pooling Layer: The constant pooling layer is applied after the convolutional layer to downsize the feature maps while aiding in the preservation of essential details or features of the input images. Moreover, it diminishes the time needed for processing. The two predominant forms of pooling are Max Pooling and Average Pooling. Eqs. (7)-(10) specify the calculations for convolutional layers, max-pooling layers, and fully connected layers, respectively.

$$z_j^L = f\left(\sum_{i \in m_j} y_i^{L-1} k_{ij}^L + B_j^L\right) \quad (7)$$

$$f(z) = max(0, z) \quad (8)$$

$$z_j^L = F\left(\beta_j^L . DW_s\left(z_j^L\right) + D_j^L\right) \quad (9)$$

$$H_{w,B(z)} = f(w_t + B) \quad (10)$$

Generally, L is the L^{th} layer, $L = 1$ be the first layer, $L = l$be the last layer, and z be the dimension input. Filter y is the dimension k and B bias 1 through $j$as the iterators. When Pooling is applied, the size of the feature map is reduced. The design of the DNN technique is shown in Figure 5.

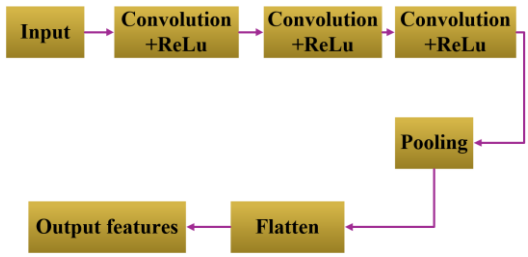

Figure 5. Architecture of the Deep Neural Network (DNN) model

The pooling layers follow to downsample the feature maps in order to maintain important information and also alleviate computational complexity. The two usual methods are max pooling, where the algorithm takes the maximum value from each section in the feature map, and average pooling, where the algorithm takes the average of each section of the feature map.

4.5 Feature fusion

The numerical data and the features extracted by the DNN are merged by concatenation of the two into a single feature vector. This step helps in incorporating both lower-level and higher-level representations into the learning process. The fused feature set x_{fus} is obtained by concatenating x_{org} and x_{dnn} along the feature axis, which is determined in Eq. (11).

$$x_{fus} = [x_{org} | x_{dnn}] \quad (11)$$

The [|] symbol refers to the process of joining two matrices side by side horizontally. This fused representation can then be utilized in the subsequent layers of the DL model employed for classification problems.

The methodology proposed has been done keeping in mind domain knowledge in medicine and tumor biology to make sure it is clinical. The choice of tumor-specific features, such as the type and stage of a tumor, is informed by the oncology criteria that have been proven to affect the progression and survival of the patient. These clinical variables are combined with imaging characteristics to represent the actual diagnostic and prognostic decision-making procedures.

Biologically, U-Net-based tumor segmentation targets isolating pathological regions, which are associated with biologically significant structures, like tumor borders and regions of infiltration, which are essential to prognosis accuracy. The obtained deep features are morphological features related to tumor aggressiveness and progression, and converge with feature learning as understood in known biological variations in brain tumors.

The proposed solution will help address the gap between technical modeling and medical and biological facts by including clinical parameters and biologically relevant imaging data into the learning process, thus increasing the significance and applicability of the results in terms of survival prediction.

4.6 Classification and survival rate prediction using Convolutional Long Short-Term Memory and Secretary Bird Optimization

The survival prediction of brain tumor patients is a multifaceted learning task where there are heterogeneous data sources, such as high-dimensional spatial data in the form of medical images and structured demographic and clinical data. These data have great spatial correlations, cross-feature relationships, and also non-linear connections that cannot be well represented using traditional feedforward or shallow recurrent architectures.

Co-LSTM networks are an extension of the standard LSTM formulation with convolutional operations in the input-to-state and state-to-state transitions. This architectural structure allows Co-LSTM to maintain the spatial locality and, at the same time, long-range dependencies on feature sequences. Within the suggested framework, the deep representations of segmented tumor areas and clinical features constitute a structured sequence representation where the spatial tumor properties and interactions of features are highly essential in the estimation of survival. Thus, Co-LSTM is theoretically best suited to learn hierarchical feature representation and time-based correlations of multi-modal survival prediction tasks.

Although Co-LSTM has the representational power, its performance is very sensitive to a good choice of the hyperparameters, especially in a high-dimensional search space like the kernel size, the number of filters, learning rates, and gating parameters. In this regard, a meta-heuristic algorithm called SBO, which is a nature-inspired algorithm, is used to tune hyperparameters. The adaptive exploration-exploitation balance, which allows SBO to search efficiently, without converging prematurely to local optima, is a characteristic of SBO. Contrary to gradient-based tuning methods and more widespread meta-heuristics, SBO actively modulates the search directions through the diversity of the population, and is therefore appropriate to optimize highly recurrent architectures that have complicated loss landscapes.

Through Co-LSTM plus SBO, the model proposed has the advantage of strong feature learning and training dynamics optimization. The Co-LSTM part is used to successfully learn spatial-temporal feature representations that are useful to tumor progression, patient survival, and SBO is able to improve the convergence behavior, generalization ability, and computational efficiency by optimizing the hyperparameter setting. Such a theoretically motivated combination will provide a robust and scalable structure for the precise prediction of the survival rate.

The DNN results and the pre-processed demographic details of the patients are fed into the proposed Co-LSTM-SBO for survival rate prediction. The proposed Co-LSTM-SBO integrates the efficiency of Co-LSTM [29] and SBO [30], where the Co-LSTM component learns the hierarchical feature representations and patterns within the data and makes survival rate predictions. Among its many uses are object detection, image categorization, and tracking.

4.6.1 Convolutional neural network

A multi-layer perceptron neural network makes up the CNN. The CNN extracts features from the images and classifies them through a fully connected layer. The structure of a CNN is made up of three layers: the convolution layer, the pooling layer, and the fully connected layer. Below is an explanation of how each layer operates:

Convolution layer: The convolution layer is essential to CNN. It produces a feature map and is used to compress the image. This layer contains a set of kernels used to create a matrix of extracted features. These kernels utilize "stride(s)" for interpolation of the whole input, resulting in decimal values for the output volume's dimensions. As a result of the striding process, the input volume dimensions of the convolutional layer diminish. The convolution operation is determined in Eq. (12).

$$\vec{c}(i,j) = \left(I_{img} * f_t\right)(i,j) = \sum\sum I_{img}(i+x, j+y)\, f_t(x,y) \tag{12}$$

Let, I_{img} be the input image, f_t be the 2D filter of size, $\vec{c}(i,j)$ be the 2D feature map output. The nonlinearity in the space of features is enhanced by the rectified linear unit (ReLU) layer. ReLU keeps the threshold input at zero to compute excitation. Eq. (13) details the calculated appearance.

$$f(z) = max(0, z) \tag{13}$$

Pooling layer: A complicated image must be reduced while retaining its essential components because it will be very large. The majority of studies used maximum polling and the maximum pooling layer, which produces the highest possible number in an input area.

Fully connected layer: This layer is used in the categorization procedure. The FC layer receives the resulting feature map and produces the result.

4.6.2 Long Short-Term Memory

To refresh and control the system's data performance, the Long Short-Term Memory (LSTM) employs a gate structure made up of input, forget, and output gates [31]. The LSTM addresses the gradient elimination problem, using memory

storage cells and gate processes to meet long-term information needs, thus overcoming RNN limitations. The numerical equation for each gate is explained below;

Considering the previously hidden structure, the forget gate $\check{f}_t$ aids the LSTM in identifying the information that must be provided and removed via the cell state using Eq. (14).

$$\check{f}_t = \check{\sigma}\left(\check{w}_{\check{f}}.\left[\check{h}_{t-1}, z_t\right] + \check{b}_{\check{f}}\right) \tag{14}$$

Let $\check{h}_{t-1}$ denotes the previous gate, z_t be the input data, $\check{\sigma}(\,)$ be the sigmoid activation function, $\check{w}_{\check{f}}$ be the weight measurements, and $\check{b}_{\check{f}}$ be the bias vector. The information will be moved to the cell state $\check{c}_t$ of the next candidate, which is determined by the input gate $\check{\imath}_t$ using Eq. (15) and (16).

$$c_t = tanh(\,)\left(\check{w}_{\check{c}}.\left[\check{h}_{t-1}, z_t\right] + \check{b}_c\right) \tag{15}$$

$$\check{\imath}_t = \check{\sigma}\left(\check{w}_{\check{\imath}}.\left[\check{h}_{t-1}, z_t\right] + \check{b}_{\check{\imath}}\right) \tag{16}$$

Let $tanh(\,)$ be the hyperbolic tangent function. The updated new cell state $\check{c}_t$ is produced by connecting the previous cell state $\check{c}_{t-1}$, and the new candidate's cell state c_t is created using Eq. (17).

$$\check{c}_t = \check{f}_t * \check{c}_{t-1} + \check{\imath}_t * c_t \tag{17}$$

The output gate $\check{o}_t$ is then developed to regulate the result of the LSTM cell using Eq. (18). The desired result $\check{h}$ is indicated by the following Eq. (19), which is the combination of $\check{o}_t$ and the cell state $\check{c}_t$ activated by the tanh function.

$$\check{o}_t = \check{\sigma}\left(\check{w}_{\check{o}}.\left[\check{h}_{t-1}, z_t\right] + \check{b}_{\check{o}}\right) \tag{18}$$

$$\check{h}_t = \check{o}_t.tanh(\check{c}_t) \tag{19}$$

The use of the LSTM classifier is because of its exceptional performance in handling long-term dependencies in data and its ability to rapidly extract sequential properties using time. The SBO technique is used to regain efficacy by carefully selecting the model parameters for the LSTM.

4.6.3 Parameter hyper-tuning using the Secretary Bird Optimization

In the population-based metaheuristic approach known as the SBO technique, each Secretary Bird (SB) is considered a member of the process's population. The values of the decision variables hinge on the location of each SB within the search space. The locations of SBs in the SBOA technique represent potential solutions to the existing problem. Eq. (20) is used in the SBOA's initial implementation to randomly initialize the SBs' locations throughout the search space.

$$\begin{matrix} Y_{a,b} = La_b + R_n \times (Ua_b - La_b) \\ a = 1,2,\ldots.,n, b = 1,2,\ldots..,dim \end{matrix} \tag{20}$$

Let Y_a be the position of the a^{th} SB La_b and Ua_b be the lower and upper bounds, and R_n be the random value between 0 and 1. A population of potential solutions serves as the starting point for optimization in the population-based SBO, as illustrated in Eq. (21).

These potential solutions Y are produced at random while adhering to the problem's upper Ua_b and lower La_b bound constraints. In each iteration, the best solution found so far is roughly regarded as the ideal answer.

$$Y = \begin{bmatrix} y_{1,1} & y_{1,2} & \cdots & y_{1,b} & \cdots & y_{1,dim} \\ y_{2,1} & y_{2,2} & \cdots & y_{2,b} & \cdots & y_{2,dim} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ y_{a,1} & y_{a,2} & \cdots & y_{a,b} & \cdots & y_{a,dim} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ y_{n,1} & y_{n,2} & \cdots & y_{n,b} & \cdots & y_{n,dim}[]_{n\times dim} \end{bmatrix} \tag{21}$$

Let, Y be the group of the SB, Y_a be the a^{th} SB, $Y_{a,b}$ be the a^{th} SB b^{th} question the value of a variable, n be the secretary group members, and dim be the problem of variable dimension.

Every SB represents a possible solution to optimize the problem. Consequently, the suggested values for the problem variables from every SB can be employed to assess the objective function. Subsequently, Eq. (22) is employed to assemble the resulting variables of the objective function into a vector.

$$f = \begin{bmatrix} f(1) \\ \vdots \\ f(i) \\ \vdots \\ f(n) \end{bmatrix}_{n\times 1} = \begin{bmatrix} f(Y_1) \\ \vdots \\ f(Y_i) \\ \vdots \\ f(Y_n) \end{bmatrix}_{n\times 1} \tag{22}$$

Let f be the vector of values for the objective function, and $f(i)$ be the objective function value gained by the a^{th} SB. Two different SB natural behaviors were used to update the SBOA employees. These two types of behavior consist of: (a) the hunting method of the SB and (b) the evasion method of the SB. Every individual of the SB colony is upgraded in two stages on every iteration.

•The secretary bird's hunting strategy (period of exploration)

When hunting snakes, SBs typically exhibit three phases in their hunting behavior: locating prey, consuming prey, and striking at prey.

Stage 1 (searching for prey): SBs typically begin their hunting by searching for potential prey, especially snakes. SBs can easily detect snakes concealed in the savannah's tall grass, thanks to their remarkable vision. They meticulously watch their surroundings and delicately sweep the ground with their long legs, looking for any indications of snakes. Consequently, Eqs. (23) and (24) are appropriate for quantitatively simulating the process of updating the SB's position during the Searching for Prey phase.

$$whileT < \frac{1}{3}t, y_{a,b}^{newq1} = y_{a,b} + (y_{rn1} - y_{rn2}) \times r_1 \tag{23}$$

$$Y_a = \begin{Bmatrix} y_a^{newq1}, & if\left(f_a^{new,q1} < f_a\right) \\ Y_a, & else \end{Bmatrix} \tag{24}$$

Let, T be the current iteration number, t be the maximum iteration number, y_a^{newq1} be the new state of the a^{th} SB in the first stage, y_{rn1} and y_{rn2} are the first stage iteration's random candidate solutions, r_1 symbolizes a dimension array that is produced at random $1 \times dim$ from the interval [0,1], dim be the solution space dimensionality. $y_{a,b}^{newq1}$ be the b^{th} dimension value, and $f_a^{new,q1}$ be the objective function fitness

value.

Stage 2 (consuming prey): An SB uses a unique hunting technique after finding a snake. Unlike other raptors that swiftly swoop in to fight, the SB utilizes its agile footwork to navigate around the snake. Here, we utilize Brownian motion (RB) to simulate the SB's unpredictable movements. Eq. (25) provides a mathematical framework for Brownian motion.

$$rb = R_n(1, dim()) \tag{25}$$

Eqs. (26) and (27) provide a mathematical framework for updating the position of the SB in the Consuming Prey phase.

$$while\frac{1}{3}t < T < \frac{2}{3}t, y_{a,b}^{newq1} = y_{best} + exp\big((T/t) \wedge 4\big) \times (rb - 0.5) \times \big(y_{best} - y_{a,b}\big) \tag{26}$$

$$Y_a = \begin{Bmatrix} y_a^{newq1}, & if\big(f_a^{new,q1} < f_a\big) \\ Y_a, & else \end{Bmatrix} \tag{27}$$

Let, y_{best} be the individual historical best position, $R_n(1, dim())$ symbolizes a dimension array that is produced at random $1 \times dim$ from a standard normal distribution and y_{best} be the current best value.

Stage 3 (attacking prey): The SB identifies the chance when the snake is fatigued and promptly strikes with its powerful leg muscles. At this point, the SB typically employs its leg-kicking approach. This involves swiftly lifting its leg to deliver a precise kick at the snake with its sharp talons, often targeting the snake's head.

As a result, Eqs. (28) and (29) can serve to statistically simulate the updating of the SB's location during the Attacking Prey phase.

$$whileT > \frac{2}{3}t, y_{a,b}^{newq1} = y_{best} + \left(\left(1 - \frac{T}{t}\right) \wedge \left(2 \times \frac{T}{t}\right)\right) \times y_{a,b} \times rl \tag{28}$$

$$Y_a = \begin{Bmatrix} y_a^{newq1}, & if\big(f_a^{new,q1} < f_a\big) \\ Y_a, & else \end{Bmatrix} \tag{29}$$

To improve the method's optimization accuracy, we employ the weighted Levy battle, which is represented as 'rl' in Eq. (30):

$$rl = 0.5 \times lv(dim()) \tag{30}$$

where, $lv(dim())$ symbolizes the distribution function of the Levy battle. This is computed using Eq. (31).

$$lv\left(d() \frac{U \times \sigma}{|V|^{\frac{1}{\varepsilon}}}\right) \tag{31}$$

Let S be the fixed constant of 0.01 and ε be the fixed constant of 1.5. U and V are random values inside the range [0,1].

•Escape strategy of SB (exploitation stage)

SBs are primarily targeted by large predators like eagles, hawks, foxes, and leopards, as they can attack them or take their food. SBs usually employ various evasive tactics to safeguard one another or their food source when faced with these dangers. These tactics can be roughly divided into two main categories. The first tactic involves a battle or fast running. In the first method, upon sensing a predator, SBs first look for an appropriate place to conceal. They will either fight or run quickly if there is no adequate and secure camouflage setting in the vicinity.

Both of the evasion strategies used by SBs can be represented mathematically through Eq. (32), and Eq. (33) can be used to establish this revised condition.

$$y_{a,b}^{newq2} = \begin{Bmatrix} c1: y_{best} + (2 \times rb - 1) \times \left(1 - \frac{T}{t}\right) \times y_{a,b} & if(Rand < R_a) \\ c2: y_{a,b} + r2 \times \left(y_{rn} - k \times y_{a,b}\right) & else \end{Bmatrix} \tag{32}$$

$$Y_a = \begin{Bmatrix} y_a^{newq2}, & if\big(f_a^{new,q2} < f_a\big) \\ Y_a, & else \end{Bmatrix} \tag{33}$$

Let $R = 0.5$, $r2$ be the random creation of a dimension array $(1 \times dim())$ from the normal distribution, y_{rn} be the arbitrary potential solution for the current iteration, and $k$be the random selection of integer 1 or 2.

4.6.4 Hybrid Convolutional Long Short-Term Memory and Secretary Bird Optimization model

The proposed Co-LSTM-SBO model, combining DL and nature-inspired optimization, performs well in classification and survival rate prediction. This approach begins with the fusion of DNN-extracted features and pre-processed demographic patient data into the Co-LSTM-SBO model. Then, the Co-LSTM component, recognized for learning hierarchical features and analyzing time-series data, is utilized to predict survival rates while being capable of detecting long dependencies and patterns within the data. The paper employs a model that utilizes feature extraction via a CNN comprising convolution, pooling, and fully connected layers. The convolutional layer uses kernels to generate feature maps through a stride operation. The pooling layer decreases the size of feature maps while retaining important features and sending them to the fully connected layer for classification.

In the Co-LSTM model, the SBO has the most important function of improving the model's performance by adjusting its parameters. SBO, which relies on hunting and avoiding SBs, is used to adjust the hyperparameters of the LSTM network for recognizing long-term dependencies in the data. The SBO algorithm operates based on the behavior observed in SBs, including foraging, consuming, and attacking to look for and find better-search solutions. While working in the exploration phase, SBO optimizes the exploration of the parameter space and, while operating in the exploitation phase, employs a finer search for the best solutions. The optimization process slashes the compromise for Co-LSTM, where there is an appreciation of enhanced impact for higher survival rate prediction on the incoming data. By applying SBO for parameter tuning, the Co-LSTM model can escape local optima and obtain higher accuracy of survival prediction from the hierarchical feature representations learned by LSTM.

This research develops a novel automatic approach for MRI tumor image detection. In creating the hybrid method applied in this research, CNN and LSTM classifiers are utilized. The features are extracted using CNN, while LSTM is utilized for classification. The SBO technique is used to select the weight values for the LSTM classifier, which represent the optimal

solution. The proposed mixed CN-LSTM-SBO tumor detection framework is illustrated in Figure 6. It is suggested that the hybrid structure be made up of twenty levels. Every convolutional unit consists of a dropout network with a 25% failure probability, along with one pooling layer and a final pooling layer. The dimensions of the kernel utilized for feature extraction are triggered by the ReLU function. Several kernels are used in the max-pooling layer, which decreases the distance between image pixels. The function map is ultimately sent to the LSTM layer in the final design phase. This layer produces the final output.

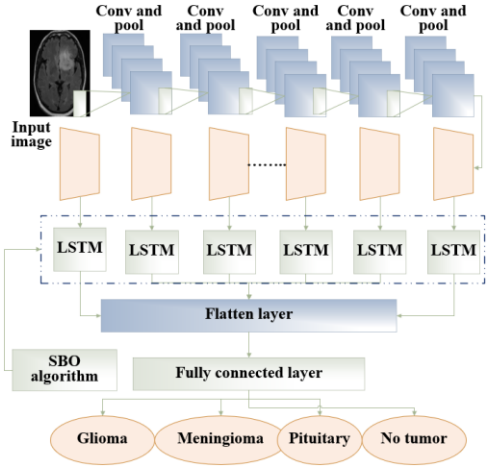

Figure 6. Design of the proposed CN-LSTM-SBO framework

5. RESULTS AND DISCUSSION

The designed technique performance metrics are validated with existing DL models in terms of accuracy, sensitivity, precision, and F1 score. The designed technique accurately segment classify, and predicts survival rates with high efficiency. All experiments were conducted under controlled conditions to ensure reproducibility and reliable evaluation. The following variables were maintained constant: cBatch size: 32, Learning rate: 0.001, Optimizer: SBO, Network architecture: Co-LSTM with 3 convolutional layers and 128 hidden units. Preprocessing pipeline: Images resized to 256 × 256 pixels, denoised using median filtering, contrast-enhanced with histogram equalization, and intensity-normalized to [0,1]. Dataset splits: stratified division of 70% training, 15% validation, and 15% test data. Random seed: 42 for dataset shuffling and weight initialization.

The confusion matrix of the developed technique is shown in Figure 7.

5.1 Performance analysis

The existing models used to validate the developed model performance are EfficientNetB2 [17], CNN [18], DenseNet169 [20], DBT-CNN [16], and the proposed model. The performance is validated with two splits of the data, such as 70 and 80. The comparison table of the two splitting data, 80 and 70, is detailed in Tables 3 and 4.

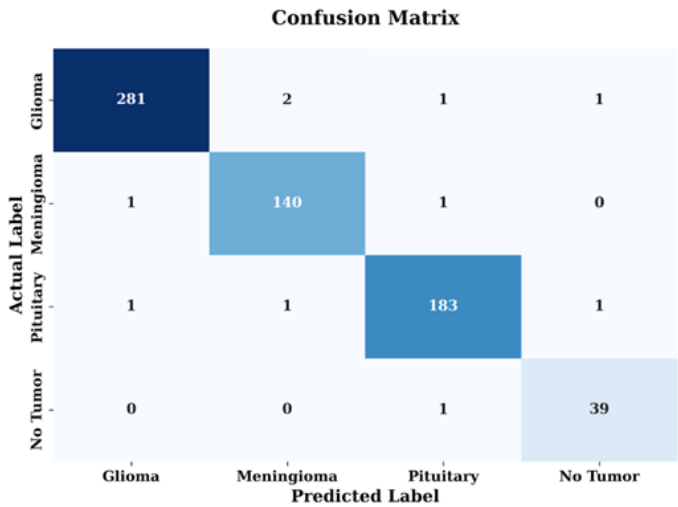

Figure 7. Confusion matrix

Table 3. Comparison data split in 80/20

Performance Metrics	EfficientNetB2	CNN	DenseNet169	DBT-CNN	Proposed
Accuracy	0.90876	0.92925	0.92458	0.94862	0.98712
Precision	0.90864	0.92914	0.92444	0.94856	0.98701
Sensitivity	0.90896	0.92936	0.92472	0.94889	0.98793
F1-Score	0.90902	0.92972	0.92512	0.94907	0.98805

Table 4. Comparison data split in 70/30

Performance Metrics	EfficientNetB2	CNN	DenseNet169	DBT-CNN	Proposed
Accuracy	0.81256	0.84926	0.85329	0.86867	0.92476
Precision	0.81242	0.84918	0.85314	0.86858	0.92468
Sensitivity	0.81268	0.84936	0.85346	0.86885	0.92482
F1-Score	0.81291	0.84962	0.85387	0.86897	0.92509

Tables 3 and 4 show the performance of several models, including EfficientNetB2, CNN, DenseNet169, DBT-CNN, and the proposed model for two splits, 80/20 and 70/30. Four metrics are used to evaluate these models: The measures consist of Accuracy, Precision, Sensitivity, and F1-Score. The results from the 80/20 data split, in which 80% of the data is allocated for training and 20% for testing, show that the Proposed model demonstrates the best performance across all metrics. The suggested method attains an accuracy of 98.71%, precision of 98.70%, sensitivity of 98.79%, and F1-Score of 98.81%. These results highly outcompete the other models, with DBT-CNN ranking the second best with accuracy (94.86%), precision (94.86%), sensitivity (94.89%), and F1-Score (94.91%). While using 70% for training and 30% for the test set, all models' performance is lower than in the case of the 80/20 data split. The Proposed model consistently outperforms the others, with an accuracy of 92.48%, precision of 92.47%, sensitivity of 92.48%, and an F1-Score of 92.51%. While it is not as high as the 80/20 split, it is still the highest in all categories. The other models, EfficientNetB2, CNN, DenseNet169, and DBT-CNN, had similar trends in their accuracy, ranging from 81–92%, depending on the model and accuracy metric.

In general, the proposed model performs better in both data splits, and this proves that it is more efficient in classification tasks than the other models. The findings underscore the significance of model architecture, given that the Proposed model excelled compared to other models regarding accuracy, precision, sensitivity, and F1-Score—particularly with the augmented training data in the 80/20 split.

•Accuracy

Among the Proposed, Baseline, and RF models, the Proposed model yields the highest accuracy of 98.71% in analyzing the 80/20 data split, suggesting its increased generality in the right categorization of most cases (Figure 8). It performs much better than all other models, and DBT-CNN comes second with 94.86% accuracy. The Total accuracy of all the models reduces in the 70/30 split, while in the proposed model, it's 92.48%, which is the highest. All the models, including EfficientNetB2, CNN, DenseNet169, and DBT-CNN, have a relatively low accuracy of 81–85%, while DBT-CNN has an accuracy of 86.87%.

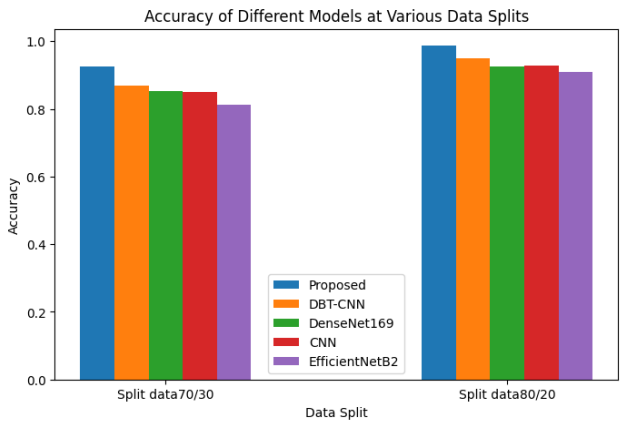

Figure 8. Accuracy comparison

•Precision

For the 80/20 split, the proposed model has a precision of 98.70%, which is a little less than the accuracy. This means that out of all the positive predictions made, 98.70% were accurate. Once more, DBT-CNN comes second with a precision of 94.86% (Figure 9). In the 70/30 split, precision reduces for all models, although the Proposed model outperforms all other models with 92.47% precision. Other models include EfficientNetB2, CNN, and DenseNet169 with a precision of 81–85%, respectively, while DBT-CNN yielded a slightly higher precision of 86.86%.

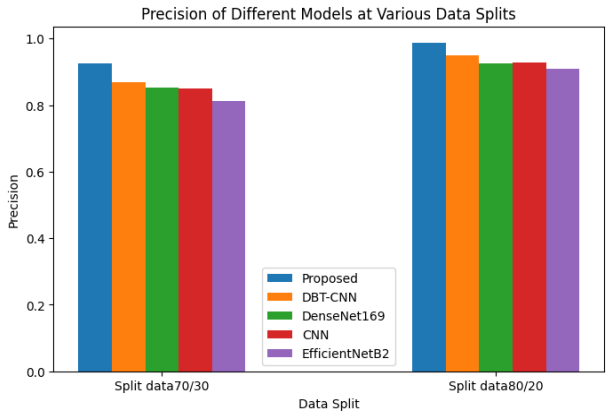

Figure 9. Precision comparison

•Sensitivity

Sensitivity estimates the extent to which all actual positives are predicted as positive by the model. In the 80/20 split, therefore, the proposed model has a high sensitivity of 98.79 percent, which shows that it performs well in identifying the positive cases (Figure 10). DBT-CNN takes the second position with a sensitivity of 94.89%. In the 70/30 split, the proposed model also emerges as the best performer with a sensitivity of 92.48%. The sensitivity of the other models successively fluctuated between 81–85%, and DBT-CNN retains the highest sensitivity value compared to the other at 86.88%.

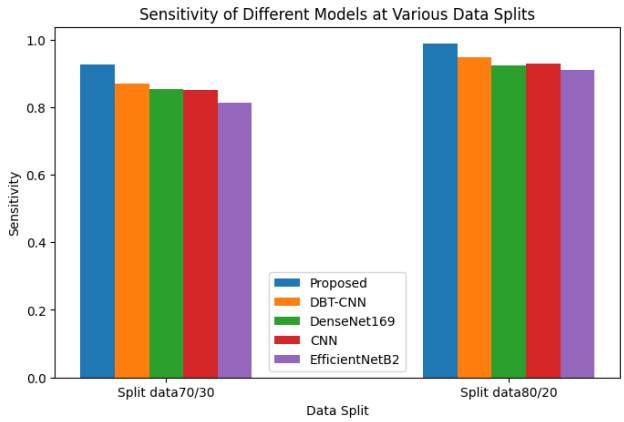

Figure 10. Sensitivity comparison

•F1-Score

The F1-Score is another key metric for assessing the model's performance, considering both precision and sensitivity. The Proposed method achieved the highest F1-Score of 98.81% in the 80/20 split, confirming its overall effectiveness (Figure 11). With an F1-Score of 94.91%, DBT-CNN is in close pursuit. With the 70/30 split, all models exhibit a lower F1-Score compared to the previous split, with the Proposed method achieving the highest score of 92.51%.

The other models also experience a similar decline in performance, and the F1 scores vary between 81% and 86%, while the DBT-CNN is slightly lower than the proposed model at 86.90%.

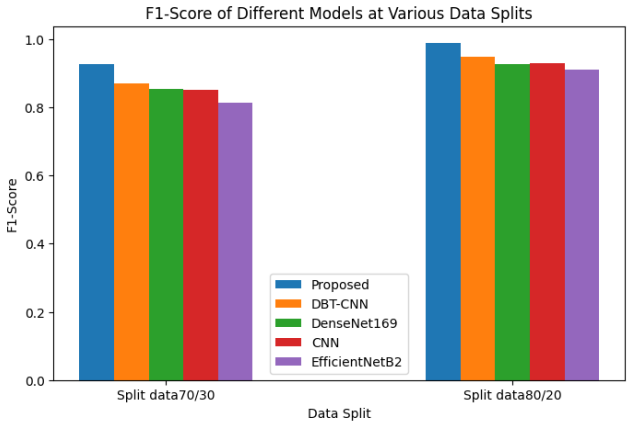

Figure 11. F1-score comparison

5.2 Discussion

The assessment of the suggested model demonstrates its effectiveness compared to all DL models, including EfficientNetB2, CNN, DenseNet169, and DBT-CNN, in the tasks of segmentation, classification, and prediction of survival rates for patients with brain tumors. The proposed model, validated on two data splits (80/20 and 70/30), consistently surpasses the other models in key performance metrics: accuracy, precision, sensitivity, and F1 Score. The proposed model demonstrates exceptional performance in the 80/20 data split (with 80% used for training and 20% for testing), achieving an accuracy of 98.71%, precision of 98.70%, sensitivity of 98.79%, and F1-Score of 98.81%. This is a lot higher than the second-best model, DBT-CNN, which achieved an accuracy of 94.86% and other performance measures slightly below 95%. The 70/30 split, while producing slightly lower results on all models, still shows the dominance of the proposed approach, with an accuracy of 92.48%, precision of 92.47%, sensitivity of 92.48%, and an F1-Score of 92.51%. The EfficientNetB2, CNN, DenseNet169, and DBT-CNN models, on the other hand, had accuracy and other indicators between 81% and 86%. The graphical representation of the ROC curve is shown in Figure 12.

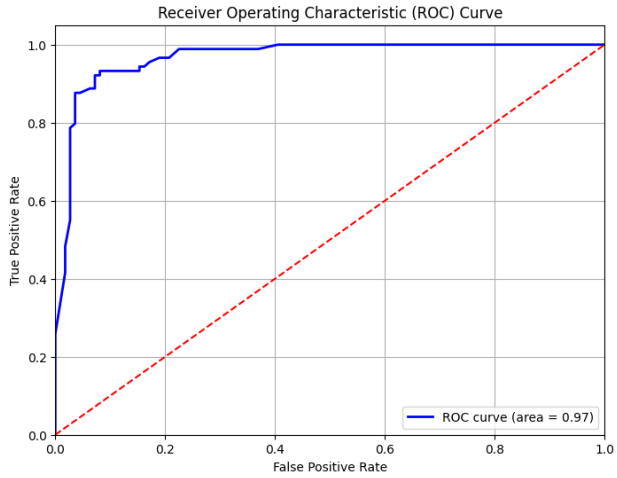

Figure 12. Receiver Operating Characteristic (ROC) curve

The proposed method's consistent performance demonstrates its sound design, allowing accurate predictions of classification and survival. Its superior metrics demonstrate accurate forecasting and management of true positive and negative cases, with a favorable F1-Score reflecting a balance between precision and sensitivity. Figure 13 displays the graph of training and validation accuracy and loss.

This method supports the prognosis of brain tumors and informs timely clinical decisions. This study proposes a novel method for predicting brain tumor survival rates using DL and meta-heuristic optimization algorithms. The method involves preprocessing patient data, using Gaussian filters, CLAHE, and scaling to ensure equal dimensions. U-Net is used for image segmentation, and DNNs are used for feature extraction. The features extracted are combined with patient demographics to predict survival using the Co-LSTM-SBO model. The Co-LSTM component optimizes hierarchical patterns in time-series data, while the SBO algorithm optimizes hyperparameters to increase training speed and computational capability. This integration enhances the credibility of predicted survival rates, advancing medical decision-making and treatment plans.

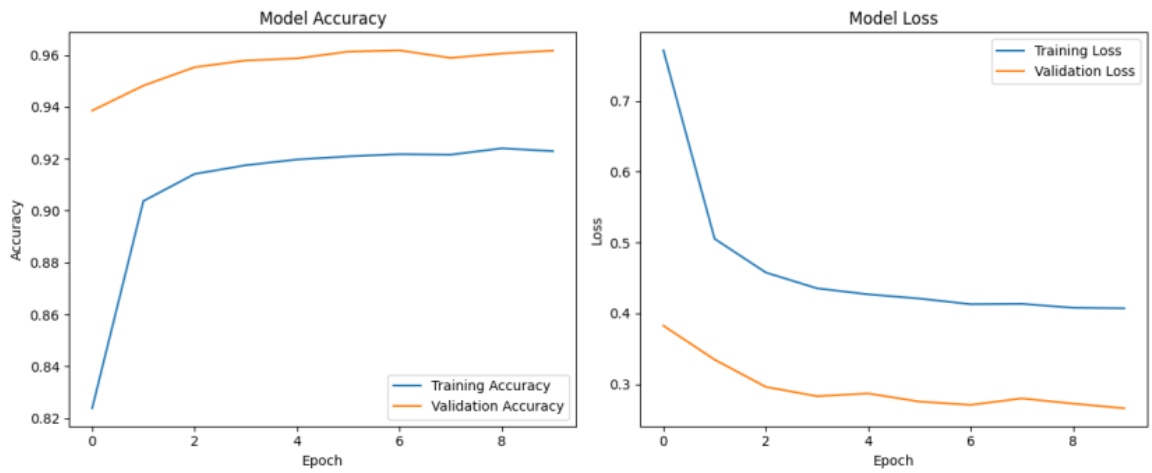

Figure 13. Accuracy and loss graph

The difference in the effectiveness of the evaluated models could be explained by the differences in the ability to represent features, the data usage strategy, and the effectiveness of optimization. Traditional deep learning (DL) architectures that process images at an image level only do not perform as well because they do not take into account patient-specific clinical context, which is very important in the prediction of survival. Due to this reason, these types of models have less generalization as predictors of long-term outcomes.

Standard convolutional architecture models are used in models that demonstrate better performance through the spatial tumor properties, but have no system to learn inter-feature correlations and sequential relations between the heterogeneous inputs. When the survival outcome is influenced by complex relationships between the tumor's morphologic characteristics and clinical properties, such a limitation results in an ineffective survival estimation.

Models Recurrent-based models, such as the conventional LSTM, use sequential dependencies to perform better; however, they are limited by fully connected operations that cannot effectively preserve the spatial locality of tumor representations. Therefore, significant spatial interactions between segmented tumor areas can be diluted when learning features.

The effectiveness of the proposed Co-LSTM-SBO framework is better than the corresponding models because it is capable of simultaneously extracting the spatial and sequential feature-dependence using convolutional recurrent operations. This is made possible by the introduction of convolution into the LSTM gating mechanism that allows the preservation of spatial tumor patterns effectively and modeling the hierarchical features interactions that are important to disease progression. Also, SBO is used to tune hyperparameters, which ensures comparable convergence stability and prevents suboptimal parameterizations as would be experienced in manually tuned or fixed-parameter models. This leads to enhanced predictive accuracy, robustness, and generalization of evaluation metrics.

The overall findings suggest that the improvement in performance is not merely a result of the augmented complexity of the models but the combination of ROI-based segmentation, discriminative feature extraction, spatial-temporal modelling, and adaptive optimisation. All these aspects make the proposed approach superior in predicting survival.

5.1 Unification of theoretical modeling and practical application

The proposed framework, in theory, has been designed with close correlation with the practical requirements in the prognosis of brain tumors. Theoretically, the Co-LSTM architecture is chosen to accommodate hierarchical spatial–temporal feature dependencies, which are critical in meeting complex interrelationships between tumor morphology and patient-specific clinical attributes. This ability in real clinical practice, in terms of predicting the survival results in heterogeneous groups of patients, is manifested in better robustness.

In the same regard, the theoretical rationale of ROI-based feature learning with U-Net segmentation has a direct implication on the practical relevance because it guarantees that prognostic information inferences are made using pertinent tumor tissue as opposed to normal tissue. This makes them less sensitive to noise and increases the accuracy of the predictions of survival in the actual diagnostic process.

The SBO is an optimization mechanism that presents a theoretically based method of the navigation of high-dimensional hyperparameter space. Practically, this allows automated model configuration, which saves the manual tuning and deploys models in resource-constrained clinical conditions. The suggested framework, therefore, fills the gap between the theory modeling and the practice through the translation of the state-of-the-art spatial-temporal learning and optimization concepts into an effective and clinically significant survival prediction system.

5.2 Model validation and robustness

The dataset was divided into independent training, validation, and testing subsets using a stratified sampling method to maintain the distribution of classes regarding tumor types and stages. This was done to assess the applicability of the proposed Co-LSTM-SBO framework. Hyperparameter tuning and early stopping were done on the validation subset, separate from the training data, so that there is no bias in the assessment of model performance.

In spite of the fact that the external independent dataset was not provided, the cross-validation method and stratified sampling reduce the risk of overfitting and give strong performance estimates. To ensure that the improvements are not specific to a particular dataset, the method’s predictive performance was also compared across various metrics, including accuracy, precision, sensitivity, and F1-score.

The testing of the framework on multi-institutional datasets will be conducted in the future to enhance the validity of the generalizability and clinical applicability.

The survival prediction performance of the proposed Co-LSTM–SBO framework was further analyzed using multimodal MRI and clinical data. The framework demonstrated superior capability in predicting patient survival outcomes by effectively learning spatial-temporal tumor progression patterns and integrating demographic and clinical attributes. Compared with conventional DL approaches, the proposed Co-LSTM–SBO model achieved improved survival prediction accuracy, precision, sensitivity, and F1-score, indicating its reliability in prognosis estimation. The incorporation of SBO enabled efficient hyperparameter tuning, which enhanced convergence stability and minimized prediction errors. Furthermore, the integration of imaging and clinical features improved the model’s ability to identify high-risk patients and estimate survival probabilities more accurately, thereby supporting personalized treatment planning and clinical decision-making.

6. CONCLUSIONS

The proposed automatic survival rate prediction system for brain tumor patients fulfills the objective of using DL models and meta-heuristic optimization to improve the prediction performance. Thus, using the U-Net for segmentation and a DNN for feature extraction, as well as demographic characteristics, the current study offers extensive input to the Co-LSTM-SBO model. The use of the Co-LSTM to learn long-term dependencies in the data, combined with the optimization properties of SBO, greatly enhances hyperparameter tuning and hence enhances the training of the

model. Experimental findings suggest that this hybrid method outperforms traditional predictive models. The proposed model achieves an accuracy of 98.71%, precision of 98.70%, sensitivity of 98.79%, and F1-Score of 98.81% when using the 80/20 data split, where 80% is for training, and 20% is for testing. This is a lot higher than the second-best model, DBT-CNN, which achieved 94.86% and very similar performance statistics just below 95%. This consistent effectiveness of the proposed model establishes a strong and sound framework that can learn and improve on the data features with high accuracy in classification and survival rates. The future work is to include more samples with different types of populations and tumor types and to investigate other optimization methods for better performance improvement.

REFERENCES

[1] Baid, U., Rane, S.U., Talbar, S., Gupta, S., Thakur, M.H., Moiyadi, A., Mahajan, A. (2020). Overall survival prediction in glioblastoma with radiomic features using machine learning. Frontiers in Computational Neuroscience, 14: 61. https://doi.org/10.3389/fncom.2020.00061

[2] Zhang, B., Shi, H., Wang, H. (2023). Machine learning and AI in cancer prognosis, prediction, and treatment selection: A critical approach. Journal of Multidisciplinary Healthcare, 16: 1779-1791. https://doi.org/10.2147/JMDH.S410301

[3] Talukder, M.A., Islam, M.M., Uddin, M.A., Akhter, A., Pramanik, M.A.J., Aryal, S., Almoyad, M.A.A., Hasan, K.F., Moni, M.A. (2023). An efficient deep learning model to categorize brain tumor using reconstruction and fine-tuning. Expert Systems with Applications, 230: 120534. https://doi.org/10.1016/j.eswa.2023.120534

[4] Srinivas, C., K.S., N.P., Zakariah, M., Alothaibi, Y.A., Shaukat, K., Partibane, B., Awal, H. (2022). Deep transfer learning approaches in performance analysis of brain tumor classification using MRI images. Journal of Healthcare Engineering, 2022(1): 3264367. https://doi.org/10.1155/2022/3264367

[5] Cè, M., Irmici, G., Foschini, C., Danesini, G.M., Falsitta, L.V., Serio, M.L., Fontana, A., Martinenghi, C., Oliva, G., Cellina, M. (2023). Artificial intelligence in brain tumor imaging: A step toward personalized medicine. Current Oncology, 30(3): 2673-2701. https://doi.org/10.3390/curroncol30030203

[6] Amin, J., Sharif, M., Haldorai, A., Yasmin, M., Nayak, R.S. (2022). Brain tumor detection and classification using machine learning: A comprehensive survey. Complex & Intelligent Systems, 8: 3161-3183. https://doi.org/10.1007/s40747-021-00563-y

[7] Kurdi, S.Z., Ali, M.H., Jaber, M.M., Saba, T., Rehman, A., Damaševičius, R. (2023). Brain tumor classification using meta-heuristic optimized convolutional neural networks. Journal of Personalized Medicine, 13(2): 181. https://doi.org/10.3390/jpm13020181

[8] Adewole, M., Rudie, J.D., Gbadamosi, A., Toyobo, O., et al. (2023). The Brain Tumor Segmentation (BraTS) challenge 2023: Glioma segmentation in sub-saharan africa patient population (BraTS-Africa). arXiv preprint arXiv:2305.19369. https://doi.org/10.48550/arXiv.2305.19369

[9] Putz, F., Beirami, S., Schmidt, M.A., May, M.S., Grigo, J., Weissmann, T., Schubert, P., Höfler, D., Gomaa, A., Hassen, B.T., Lettmaier, S., Frey, B., Gaipl, U.S., Distel, L.V., Semrau, S., Bert, C., Fietkau, R., Huang, Y. (2024). The Segment Anything foundation model achieves favorable brain tumor auto-segmentation accuracy in MRI to support radiotherapy treatment planning. Strahlentherapie und Onkologie, 201(3): 255-265. https://doi.org/10.1007/s00066-024-02313-8

[10] Anjum, S., Hussain, L., Ali, M., Alkinani, M.H., et al. (2022). Detecting brain tumors using deep learning convolutional neural network with transfer learning approach. International Journal of Imaging Systems and Technology, 32(1): 307-323. https://doi.org/10.1002/ima.22641

[11] Mahmud, M.I., Mamun, M., Abdelgawad, A. (2023). A deep analysis of brain tumor detection from MR images using deep learning networks. Algorithms, 16(4): 176. https://doi.org/10.3390/a16040176

[12] Amin, J., Anjum, M.A., Sharif, M., Jabeen, S., Kadry, S., Moreno Ger, P. (2022). A new model for brain tumor detection using ensemble transfer learning and quantum variational classifier. Computational Intelligence and Neuroscience, 2022(1): 3236305. https://doi.org/10.1155/2022/3236305

[13] Soewu, T., Singh, D., Rakhra, M., Chakraborty, G.S., Singh, A. (2022). Convolutional neural networks for MRI-based brain tumor classification. In 2022 3rd International Conference on Computation, Automation and Knowledge Management (ICCAKM), Dubai, United Arab Emirates, pp. 1-7. https://doi.org/10.1109/ICCAKM54721.2022.9990173

[14] Guan, X., Yang, G., Ye, J., Yang, W., Xu, X., Jiang, W., Lai, X. (2022). 3D AGSE-VNet: An automatic brain tumor MRI data segmentation framework. BMC Medical Imaging, 22(1): 6. https://doi.org/10.1186/s12880-021-00728-8

[15] Mijwil, M.M., Doshi, R., Hiran, K.K., Unogwu, O.J., Bala, I. (2023). MobileNetV1-based deep learning model for accurate brain tumor classification. Mesopotamian Journal of Computer Science, 2023: 29-38. https://doi.org/10.58496/MJCSC/2023/005

[16] Zaitoon, R., Syed, H. (2023). RU-Net2+: A deep learning algorithm for accurate brain tumor segmentation and survival rate prediction. IEEE Access, 11: 118105-118123. https://doi.org/10.1109/ACCESS.2023.3325294

[17] Babu Vimala, B., Srinivasan, S., Mathivanan, S.K., Mahalakshmi, Jayagopal, P., Dalu, G.T. (2023). Detection and classification of brain tumor using hybrid deep learning models. Scientific Reports, 13(1): 23029. https://doi.org/10.1038/s41598-023-50505-6

[18] Othman, N.A., Abdel-Fattah, M.A., Ali, A.T. (2023). A hybrid deep learning framework with decision-level fusion for breast cancer survival prediction. Big Data and Cognitive Computing, 7(1): 50. https://doi.org/10.3390/bdcc7010050

[19] Celik, M., Inik, O. (2024). Development of hybrid models based on deep learning and optimized machine learning algorithms for brain tumor multi-classification. Expert Systems with Applications, 238: 122159. https://doi.org/10.1016/j.eswa.2023.122159

[20] Khan, S.U.R., Zhao, M., Asif, S., Chen, X. (2023). HybridNET: A fusion of DenseNet169 and advanced machine learning classifiers for enhanced brain tumor

diagnosis. International Journal of Imaging Systems and Technology, 34(1): e22975. https://doi.org/10.1002/ima.22975

[21] Srinivasan, S., Francis, D., Mathivanan, S.K., Rajadurai, H., Shivahare, B.D., Shah, M.A. (2024). A hybrid deep CNN model for brain tumor image multi-classification. BMC Medical Imaging, 24(1): 21. https://doi.org/10.1186/s12880-024-01195-7

[22] Dixon, J., Akinniyi, O., Abdelhamid, A., Saleh, G.A., Rahman, M.M., Khalifa, F. (2024). A hybrid learning-architecture for improved brain tumor recognition. Algorithms, 17(6): 221. https://doi.org/10.3390/a17060221

[23] Khan, S.U.R., Asif, S., Zhao, M., Zou, W., Li, Y., Xiao, C. (2026). ShallowMRI: A novel lightweight CNN with novel attention mechanism for Multi brain tumor classification in MRI images. Biomedical Signal Processing and Control, 111: 108425. https://doi.org/10.1016/j.bspc.2025.108425

[24] Hekmat, A., Zhang, Z., Khan, S.U.R., Shad, I., Bilal, O. (2025). An attention-fused architecture for brain tumor diagnosis. Biomedical Signal Processing and Control, 101: 107221. https://doi.org/10.1016/j.bspc.2024.107221

[25] Almadhoun, H.R., Abu-Naser, S.S. (2022). Detection of brain tumor using deep learning. International Journal of Academic Engineering Research, 6(3): 29-47.

[26] Rao, C.S., Karunakara, K. (2022). Efficient detection and classification of brain tumor using kernel based SVM for MRI. Multimedia Tools and Applications, 81(5): 7393-7417. https://doi.org/10.1007/s11042-021-11821-z

[27] brain_tumor_classification. https://www.kaggle.com/datasets/prathamgrover/brain-tumor-classification.

[28] Muhammad, L.J., Badi, I., Haruna, A.A., Mohammed, I.A., Dada, O.S. (2022). Deep learning models for classification of brain tumor with magnetic resonance imaging images dataset. In Computational Intelligence in Oncology: Applications in Diagnosis, Prognosis and Therapeutics of Cancers, pp. 159-176. https://doi.org/10.1007/978-981-16-9221-5_9

[29] Agrawal, P., Katal, N., Hooda, N. (2022). Segmentation and classification of brain tumor using 3D-UNet deep neural networks. International Journal of Cognitive Computing in Engineering, 3: 199-210. https://doi.org/10.1016/j.ijcce.2022.11.001

[30] Yuan, Y., Shao, C., Cao, Z., He, Z., Zhu, C., Wang, Y., Jang, V. (2020). Bus dynamic travel time prediction: Using a deep feature extraction framework based on RNN and DNN. Electronics, 9(11): 1876. https://doi.org/10.3390/electronics9111876

[31] Shanthi, S., Saradha, S., Smitha, J. A., Prasath, N., Anandakumar, H. (2022). An efficient automatic brain tumor classification using optimized hybrid deep neural network. International Journal of Intelligent Networks, 3: 188-196. https://doi.org/10.1016/j.ijin.2022.11.003