The Brain Watching Itself: Identifying Brain Tumors with Spiking Neural Networks

Namita Achyuthan
Dept. of CSE(AIML)
PES University
Bengaluru, India
namitaachyuthan@gmail.com

Bhaskarjyoti Das
Dept. of CSE(AIML)
PES University
Bengaluru, India
bhaskarjyoti01@gmail.com

***Abstract*—This study looks at the application of spiking neural networks (SNNs) in MRI classification, in order to show their potential as an efficient, biologically inspired alternative to traditional convolutional models. Specifically, we evaluate the performance of leaky integrate-and-fire (LIF) neurons within a convolutional spiking neural network (CSNN). Additionally, we compare two encoding strategies- rate and temporal encoding- to assess their effectiveness in capturing MRI features. The findings aim to demonstrate the viability of SNNs for MRI analysis, offering insights into their computational efficiency and biological plausibility.**

***Index Terms*—Spiking Neural Networks (SNNs), Brain Tumor Detection, MRI Classification, Neuromorphic Computing, Energy-Efficient AI**

I. Introduction

Magnetic resonance imaging plays an important role in tumor detection. Conventional convolutional neural networks do achieve high accuracy, but are computationally expensive and lack biological plausibility. Thus, spiking neural networks offer an alternative to traditional architecture, by mimicking the event-driven nature of biological neurons, thereby identifying patterns efficiently and dynamically, while using reduced compute.

Unlike traditional artificial neural networks (ANNs), SNNs process information through discrete spikes, closely resembling neuronal communication in the human brain. This allows them to encode spatial and temporal dependencies, making them suitable for MRI classification [1] [2] [3]. Additionally, their sparse and asynchronous processing reduces computational overhead [4].

This study investigates the effectiveness of SNNs for MRI classification by evaluating the Leaky Integrate-and-Fire (LIF) neuron within Convolutional Spiking Neural Networks (CSNNs). Furthermore, the impact of rate and temporal encoding on classification performance is analyzed. By demonstrating the viability of SNN-based approaches, this work highlights their potential advantages in computational efficiency and medical diagnostics.

II. Spiking Neural Networks

SNNs are a class of biologically inspired neural networks. These process information through discrete events that are called spikes. Since they leverage the spikes for computation, they are more energy-efficient than their traditional deep learning counterparts. The main idea behind SNNs is to be able to encode and transmit information using spike trains, exactly the way biological neurons communicate. One of the most common neuron models in SNNs is the Leaky-Integrate-and-Fire Model (LIF), which captures the dynamics of a neuron's membrane potential over time and determines the spiking behavior. This membrane potential is modeled by the differential equation:

$$\tau_m \frac{dV(t)}{dt} = -(V(t) - V_{\text{rest}}) + RI(t)$$

Where, τ_m is the membrane time constant, V_{rest} is the resting potential of the neuron, R is the membrane resistance, and $I(t)$ is the input current.

Here, if the membrane potential $V(t)$ reaches a threshold V_{th}, the neuron will emit a spike, and then the potential is reset to a lower value V_{reset}. Mathematically, this is expressed as:

$$V(t) \geq V_{\text{th}} \Rightarrow \text{Spike occurs, and } V(t) \rightarrow V_{\text{reset}}$$

The LIF model also introduces a leak term $-(V(t) - V_{\text{rest}})$, which ensures that the neuron returns to its resting potential when there is no input.

Convolutional Spiking Neural Networks (CSNNs) build on Convolutional Neural Networks (CNNs) by using spiking neurons. In the CSNN, convolutions are applied followed by spiking neuron activations, which makes them efficient for image and video processing. In a CSNN, the convolutional layer extracts features by computing:

$$h_{ij} = \sum_{m,n} W_{mn} \cdot x_{i-m,j-n}$$

where $x_{i,j}$ represents the input feature map, W_{mn} is the convolutional kernel, and h_{ij} is the convolved output. The then convolved features are passed through an SNN layer and converted into spike trains.

III. Related Work

A. Convolutional Neural Networks for MRI Classification

Convolutional neural networks are commonly used for MRI-based tumor detection and classification. This is due to their

ability to extract varied features from images [5]. Traditional CNNs, such as AlexNet, VGG, ResNet, and InceptionNet have been used for various medical imaging applications which includes brain tumor detection [6] [7] [8].

These models, however, are very computationally intensive [9] and require significant compute for training and inference. Moreover, traditional CNNs lack biological plausibility, which can be a drawback for researchers interested in understanding brain-like computation. Some approaches may also require manual segmentation of tumor regions [10], which is not fully automated.

B. Spiking Neural Networks in Medical Imaging

SNNs offer a more biologically inspired alternative to traditional convolutional models by mimicking the event-driven nature of biological neurons [19]. Their sparse spike-based communication allows for significant energy efficiency, with studies indicating 10-100 times lower energy consumption compared to conventional neural networks [14]. This efficiency arises from asynchronous activation patterns, which only require computations when membrane potentials reach spiking thresholds [12].

In medical imaging, SNNs have shown promising results [11]:

1) **Brain Tumor Analysis**: Hybrid SNN architectures combining spiking layers with CNNs achieved up to 94.8% accuracy in MRI tumor classification [1], while specialized models improved segmentation by 12% over traditional methods [3]. Evolutionary approaches have enabled precise 3D quantification of tumors in MRI scans [20].
2) **Oncology Imaging**: For breast cancer detection, SNNs utilizing multi-scale saliency fusion reached 96.3% accuracy and reduced computational costs by 38% [13]. In COVID-19 diagnoses, spiking architectures achieved 98.4% sensitivity in detecting lung abnormalities from CT scans [18].
3) **Neurological Signal Processing**: MRI-structured SNNs demonstrated a 20-30% improvement in EEG signal prediction accuracy compared to LSTM models [2]. Real-time implementations have enabled sub-100ms latency in processing functional MRI data streams [16].

Recent advancements address implementation challenges, such as overcoming layer synchronization limitations to improve training convergence by 40% [4]. Adaptive synaptic weight algorithms have enhanced segmentation accuracy on diffusion-weighted MRI scans [17]. Despite these advancements, balancing biological plausibility with computational efficiency remains a challenge, particularly with complex neuron models increasing spiking operations significantly [15].

Continuous learning mechanisms enable real-time model refinement from streaming patient data, positioning SNNs as a transformative technology for medical imaging in resource-constrained environments [19].

C. Neuron Models in SNNs

The choice of neuron model is an important aspect of the performance of an SNN. It has a direct relation to the biological realism and the computational efficiency of the network. Different neuron models vary in their complexity and level of detail [21] [22].

The most commonly used neuron models are:

1) **Leaky Integrate-and-Fire (LIF) Neurons**: LIF neurons are simple and computationally efficient, making them a common choice for large-scale neural networks. They integrate incoming spikes over time and fire when a threshold is reached, resetting afterward. This simplicity allows for fast simulation but may lack the detailed biological accuracy of more complex models [23]
2) **Hodgkin-Huxley (HH) Neurons**: HH neurons are the most biologically realistic since they model the ion channel dynamics of real neurons. However, this comes at the cost of very high computational complexity, limiting their use in large networks [24] [25]
3) **Izhikevich Neurons**: These neurons strike a balance between LIF and HH models by offering both biological realism and computational efficiency. They reproduce firing patterns observed in real neurons, making them versatile for different applications [26]

D. Encoding Schemes

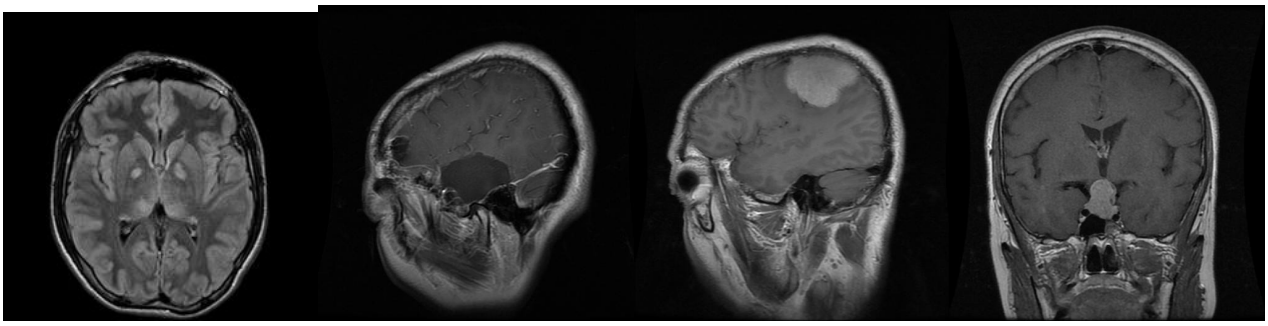
The way that information is encoded in an SNN determines their efficiency and performance in classification. Different methods of encoding process information in different ways, affecting the SNN's ability to capture the spatial and temporal features of the MRI images [27] [30].

1) **Rate Encoding**: Information is encoded by the frequency of spikes over a given time window [28]. It is straightforward in its implementation and is used commonly due to its stability. However, it may not make use of the temporal precision that is innate to spiking computation.
2) **Temporal Encoding**: Unlike rate encoding, this method relies on the timing of the spikes to encode information. This allows more biologically plausible processing and can enhance classification performance. However, this method requires more precise spike timing control, which makes it more complex in its implementation [29].

IV. Methodology

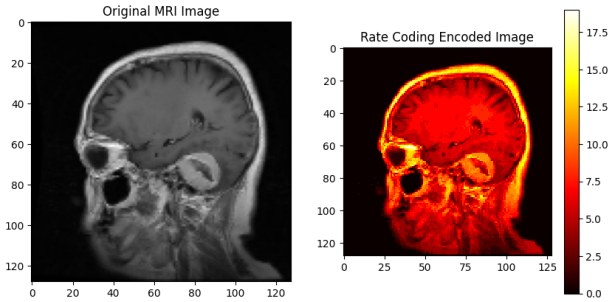
A. Data Acquisition and Preprocessing

For this study, the Brain Tumor MRI dataset sourced from Sartaj Bhuvaji's repository was utilized [31], which comprises labeled MRI scans categorized into four classes. The dataset was structured into distinct directories, enabling a streamlined workflow for classification tasks. To these images, preprocessing techniques were applied, including normalizations by rescaling the pixel values in the [0,1] range. Given the complexity of the MRI images, data augmentation was initially

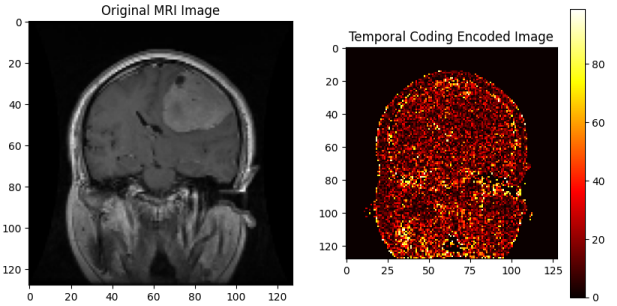


(a) No Tumor (b) Glioma (c) Meningioma (d) Pituitary Tumor

Fig. 1: Example MRI scans from four different classes in the dataset.



(a) Rate Encoding: Spike frequency corresponds to pixel intensity.



(b) Temporal Encoding: Brighter pixels spike earlier in the time window.

Fig. 2: Comparison of encoding methods: rate encoding vs. temporal encoding.

considered but omitted to keep the integrity of the features of the tumors. The dataset was partitioned into training and testing subsets, with the original class distribution maintained to remove any potential bias. Samples from each of the four classes are displayed in Figure 1.

B. Convolutional Network Classification

Different CNN architectures were implemented to review their effectiveness in brain tumor classification. The architectures evaluated included EfficientNetB0 [32], TumorDetNet [33], AlexNet [34], and DenseNet201 [35]. Each of these used different structural advantages to enhance feature extraction and accuracy. To optimize model performance, multiple regularization techniques were used, such as dropout layers, batch normalization, and weight decay, to prevent overfitting. Early stopping and learning rate scheduling were used to dynamically adjust training based on the validation performance. Also, initial layers were selectively frozen to finetune deeper layers to adapt to MRI specific features. Each model was trained using categorical cross-entropy loss and the Adam optimizer, with validation accuracy to assess model performance. Computational resources used was measured by CPU Usage and RAM Usage. A comparative analysis of these architectures was conducted to determine the computational requirement and accuracy. The hyperparameters for the different architectures are described in Table I.

C. Spiking Neural Network Classification

In order for the SNNs to process MRI images, we use spike-based encoding schemes that convert static MRI images to spike trains. We utilize two common methods- rate and temporal encoding. Both methods translate pixel intensity values to spike patterns that are then processed by the spiking neurons.

1) **Rate Encoding**: This method encodes information by changing the spike frequency relative to the pixel intensity. A higher intensity would correspond to a greater number of spikes within a fixed time window. Conversely, the lower the intensity, the lower the number of spikes. Given an MRI image $\mathbf{I} \in \mathbb{R}^{H \times W}$, where H and W denote height and width, each pixel intensity $I_{ij} \in [0, 1]$ is mapped to a spike train $S_{ij}(t)$ such that:

$$S_{ij}(t) = \begin{cases} 1, & \text{if } t < \lfloor I_{ij} \cdot M \rfloor \\ 0, & \text{otherwise} \end{cases}$$

where M represents the maximum number of spikes per pixel within the predefined time window T. The spike trains are stored as binary tensors of dimension $H \times W \times T$, where T represents the temporal dimension.

2) **Temporal Encoding**: This method uses the precise timing of the spikes to encode pixel intensity. This means that brighter pixels fire earlier in the time window,

Model	Learning Rate	Epochs	L2 Regularization	Dropout	BatchNorm
EfficientNetB0	0.001	50	1×10^{-5}	0.2	Yes
TumorDetNet	0.0005	100	1×10^{-4}	0.3	Yes
AlexNet	0.01	30	5×10^{-4}	0.5	No
DenseNet201	0.0001	30	1×10^{-5}	0.2	Yes

TABLE I: Hyperparameters for different CNN architectures.

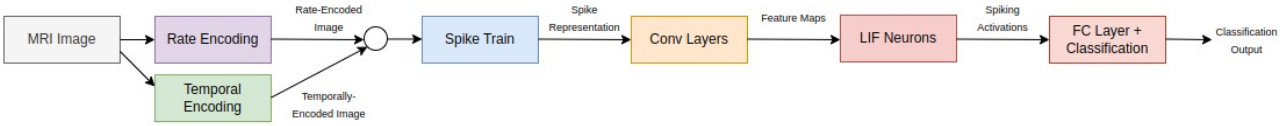


Fig. 3: Architecture of the Convolutional Spiking Neural Network (CSNN).

and darker pixels may fire later or not at all. This helps preserve the temporal dynamics that are important for SNN processing. For a given pixel intensity I_{ij}, spike times t_k are determined as follows:

$$t_k = \lfloor (1 - I_{ij}) \cdot T \rfloor + \epsilon_k$$

where ϵ_k is a small perturbation to introduce biological variability in spike timing.

The resulting spike train consists of sparse and precisely timed events, thereby enabling efficient information representation. To apply temporal encoding to static MRI images, we mapped each pixel intensity to a single spike event occurring at a specific time step within the window . Brighter pixels triggered spikes at earlier timesteps, while darker pixels fired later. If a pixel had zero intensity, no spike was generated. This encoding transformed a static 2D image into a time-series spike representation, allowing the SNN to process spatial features dynamically over time. The encoded spike matrices were finally stored as .npy files. The results of the rate and temporal encoding are presented in Figure 2. Both encoding schemes were applied to the MRI images from the aforementioned dataset, and converted the images to spike trains.

To process the input spike trains, we designed a Convolutional Spiking Neural Network (CSNN) optimized for feature extraction and classification. The overall architecture of the proposed model is shown in Figure 3. The network consists of the following key components:

1) **Convolutional Layers**: Two convolutional layers, with 32 and 64 filters, extract the spatial features. Each convolutional layer is followed by an LIF neuron activation and average pooling.
2) **Fully Connected Layers**: A FC layer with 128 neurons processes the extracted features, and is then followed by a dropout layer to prevent overfitting. The final output layer classifies the images.
3) **Leaky Integrate-and-Fire Neurons**: Each layer used LIF Neurons. The purpose was to accumulate input currents over a period of time (the time window was set to 100 timesteps) and fire spikes when a threshold is reached.

SnnTorch 0.9.4 [36] was used for implementing Spiking Neural Networks (SNNs), leveraging its support for neuron models and surrogate gradient learning.

The implementation for the convolutional spiking neural network is described in table II.

Parameter	Value
Learning Rate	0.001
Epochs	30
Batch Size	8
Optimizer	Adam
Loss Function	Cross Entropy
Regularization	Implicit L2 via Adam
Encoding Method	Temporal / Rate
Neuron Model	Leaky-Integrate and Fire
Membrane Decay	0.9
Time Window	100

TABLE II: Parameters for the Convolutional Spiking Neural Network

V. Results and Discussion

The outcome of the brain tumor classification task by the various convolutional neural network architectures is summarized in table III. The results of the CSNN with different encoding schemes is summarized in table IV.

From the results of our study, we demonstrate that the CSNN achieved comparable, and in some cases, superior accuracy compared to the traditional CNN counterparts, along with utilizing significantly less memory and computational resources. This highlights the advantages of biologically inspired neural network architectures in resource constrained environments. One possible reason for this is the spiking nature of the CSNN which allows it to efficiently encode and process information. Traditional CNNs operate on continuous valued activations, whereas CSNNs operate using discrete spikes, similar to the way biological neurons communicate.

This ensures that neurons only activate when necessary- an advantage that reduces redundant computations and improves overall efficiency. CSNNs can also learn from sparse and temporally structured data, as in the case of MRIs, thereby contributing to its superior performance. CSNNs also leverage temporal encoding, which enhances feature extraction by capturing patterns in data more effectively than traditional CNNs. This leads to better generalization and robustness as well. Another contributing factor could be the way CSNNs handle noise and redundant information. By using spike-based communication, CSNNs inherently filter out irrelevant information, focusing only on salient features. In contrast, CNNs process all input information continuously, potentially leading to overfitting or inefficient feature selection.

Overall, our findings suggest that biologically inspired architectures such as CSNNs offer promising alternatives to traditional deep learning models. Their ability to match or exceed the accuracy of CNNs while requiring fewer resources demonstrates the viability of neuromorphic computing as a step towards more energy-efficient artificial intelligence systems.

Metric	Value
EfficientNetB0	
Validation Accuracy	68.78%
CPU Usage	2.0%
RAM Usage (MB)	6818.33
TumorDetNet	
Validation Accuracy	67.01%
CPU Usage	40.5%
RAM Usage (MB)	6892.12
AlexNet	
Validation Accuracy	74.62%
CPU Usage	23.6%
RAM Usage (MB)	2287.14
DenseNet201	
Validation Accuracy	77.92%
CPU Usage	48.0%
RAM Usage (MB)	5210.22

TABLE III: Performance metrics of different CNN models.

Metric	Value
Rate Encoding	
Validation Accuracy	53.55%
CPU Usage	0.8%
RAM Usage (MB)	1187.95
Temporal Encoding	
Validation Accuracy	73.10%
CPU Usage	1.4%
RAM Usage (MB)	948.27

TABLE IV: Performance metrics of CSNN with different encodings.

VI. Conclusion and Future Work

The findings of this study highlight the potential of SNNs as an effective approach for MRI classification, demonstrating their efficiency and accuracy. They show that CSNNs can achieve comparable or even superior performance to traditional Convolutional Neural Networks (CNNs) while requiring significantly less computational power. This underscores the importance of biologically inspired algorithms in medical imaging applications.

Future research will focus on enhancing hybrid encoding techniques and optimizing hardware implementations to improve the real-world applicability of SNNs. Advancements in neuromorphic computing and energy-efficient architectures will further support their integration into clinical workflows, making SNN-based models a viable alternative for high-performance medical imaging tasks.

References

[1] B. Niepceron, A. Nait-Sidi-Moh, and F. Grassia, "Spiking convolutional neural network for brain tumor classification," 2022.

[2] S. A. Saeedinia, M. R. Jahed-Motlagh, A. Tafakhori, and N. Kasabov, "Design of MRI structured spiking neural networks and learning algorithms for personalized modelling, analysis, and prediction of EEG signals," *Scientific Reports*, vol. 11, no. 1, p. 12064, 2021.

[3] Y. Yue et al., "Spiking neural networks fine-tuning for brain image segmentation," *Frontiers in Neuroscience*, vol. 17, p. 1267639, 2023.

[4] R. Koopman, A. Yousefzadeh, M. Shahsavari, G. Tang, and M. Sifalakis, "Overcoming the Limitations of Layer Synchronization in Spiking Neural Networks," *arXiv preprint arXiv:2408.05098*, 2024.

[5] D. Bala et al., "Automated brain tumor classification system using convolutional neural networks from MRI images," in *Proc. 2022 Int. Conf. Eng. Emerging Technol. (ICEET)*, 2022, pp. 1–6.

[6] A. A. Akinyelu et al., "Brain tumor diagnosis using machine learning, convolutional neural networks, capsule neural networks and vision transformers, applied to MRI: a survey," *Journal of Imaging*, vol. 8, no. 8, p. 205, 2022.

[7] M. Talo, O. Yildirim, U. B. Baloglu, G. Aydin, and U. R. Acharya, "Convolutional neural networks for multi-class brain disease detection using MRI images," *Comput. Med. Imaging Graph.*, vol. 78, p. 101673, 2019.

[8] S. D. Ch. and P. Gundagurti, "Deep learning based diagnosis of Parkinson's disease using CNN," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 6, pp. 351–355, 2020.

[9] P. Maji and R. Mullins, "On the reduction of computational complexity of deep convolutional neural networks," *Entropy*, vol. 20, no. 4, p. 305, 2018.

[10] F. Ullah et al., "Brain tumor segmentation from MRI images using handcrafted convolutional neural network," *Diagnostics*, vol. 13, no. 16, p. 2650, 2023.

[11] X. Li et al., "Review of medical data analysis based on spiking neural networks," *Procedia Comput. Sci.*, vol. 221, pp. 1527–1538, 2023.

[12] K. Yamazaki, V. K. Vo-Ho, D. Bulsara, and N. Le, "Spiking neural networks and their applications: A review," *Brain Sciences*, vol. 12, no. 7, p. 863, 2022.

[13] Q. Fu and H. Dong, "Spiking neural network based on multi-scale saliency fusion for breast cancer detection," *Entropy*, vol. 24, no. 11, p. 1543, 2022.

[14] D. Wu, X. Yi, and X. Huang, "A little energy goes a long way: Build an energy-efficient, accurate spiking neural network from convolutional neural network," *Frontiers in Neuroscience*, vol. 16, p. 759900, 2022.

[15] P. Pietrzak *et al.*, "Overview of spiking neural network learning approaches and their computational complexities," *Sensors*, vol. 23, no. 6, p. 3037, 2023.

[16] J. Kirkland *et al.*, "Imaging from temporal data via spiking convolutional neural networks," *SPIE Proceedings on Medical Imaging and Sensing Technologies*, vol. 11540, pp. 66–85, 2020.

[17] R. Zheng *et al.*, "Image segmentation method based on spiking neural networks with adaptive synaptic weights," *IEEE Signal and Image Processing Conference (ICSIP)*, 2019.

[18] A. Garain, A. Basu, F. Giampaolo, J. D. Velasquez, and R. Sarkar, "Detection of COVID-19 from CT scan images: A spiking neural network-based approach," *Neural Computation and Applications*, vol. 33, no. 19, pp. 12591–12604, 2021.

[19] W. Maass, "Networks of spiking neurons: The third generation of neural network models," *Neural Networks*, vol. 10, no. 9, pp. 1659–1671, Sep. 1997.
[20] A. Baladhandapani and D. S. Nachimuthu, "Evolutionary learning of spiking neural networks towards quantification of 3D MRI brain tumor tissues," *Soft Computing*, vol. 19, pp. 1803–1816, 2015.
[21] M. Ahmadi, A. Sharifi, S. Hassantabar, and S. Enayati, "QAIS-DSNN: tumor area segmentation of MRI image with optimized quantum matched-filter technique and deep spiking neural network," *BioMed Research International*, vol. 2021, no. 1, p. 6653879, 2021.
[22] S. Koravuna et al., "Exploring spiking neural networks: a comprehensive analysis of mathematical models and applications," *Frontiers in Computational Neuroscience*, vol. 17, p. 1215824, 2023.
[23] S. Lu and F. Xu, "Linear leaky-integrate-and-fire neuron model based spiking neural networks and its mapping relationship to deep neural networks," *Frontiers in Neuroscience*, vol. 16, p. 857513, 2022.
[24] X. Fang, S. Duan, and L. Wang, "Memristive Hodgkin-Huxley spiking neuron model for reproducing neuron behaviors," *Frontiers in Neuroscience*, vol. 15, p. 730566, 2021.
[25] U. Chandrashekhar, "The Hodgkin-Huxley Model for Neuron Action Potentials: A Computational Study," *Macalester J. Phys. Astronomy*, vol. 12, no. 1, p. 4, 2024.
[26] E. M. Izhikevich, "Simple model of spiking neurons," *IEEE Trans. Neural Netw.*, vol. 14, no. 6, pp. 1569–1572, 2003.
[27] A. Fois and B. Girau, "Enhanced representation learning with temporal coding in sparsely spiking neural networks," *Frontiers in Computational Neuroscience*, vol. 17, p. 1250908, 2023.
[28] A. A. Al-Hamid and H. Kim, "Optimization of spiking neural networks based on binary streamed rate coding," *Electronics*, vol. 9, no. 10, p. 1599, 2020.
[29] I. M. Comsa et al., "Temporal coding in spiking neural networks with alpha synaptic function," in *Proc. ICASSP 2020 IEEE Int. Conf. Acoust. Speech Signal Process.*, 2020, pp. 8529–8533.
[30] C. Wang et al., "Neural encoding with unsupervised spiking convolutional neural network," *Communications Biology*, vol. 6, no. 1, p. 880, 2023.
[31] A. Kadam, S. Bhuvaji, and S. Deshpande, "Brain tumor classification using deep learning algorithms," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 9, pp. 417–426, 2021.
[32] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.
[33] N. Ullah et al., "TumorDetNet: A unified deep learning model for brain tumor detection and classification," *PLoS ONE*, vol. 18, no. 9, p. e0291200, 2023.
[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Adv. Neural Inf. Process. Syst.*, vol. 25, 2012.
[35] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.
[36] J. K. Eshraghian, M. Ward, E. Neftci, X. Wang, G. Lenz, G. Dwivedi, M. Bennamoun, D. S. Jeong, and W. D. Lu, "Training spiking neural networks using lessons from deep learning," *Proceedings of the IEEE*, vol. 111, no. 9, pp. 1016–1054, 2023.