EAI Endorsed Transactions
on Tourism, Technology and Intelligence
Research Article EAI.EU

Hybrid ViT-Prototypical Reptile Learning for Few-Shot Brain Tumor MRI Classification

Tuyet-Nhi T. Nguyen[1], Tuan Thanh Nguyen[2], Quang Nhat Le[1], and Nhan Duc Le[3,*]

[1]Memorial University, St. John's, NL A1B3X5, Canada
[2]School of Computing and Mathematical Sciences, University of Greenwich, UK
[3]Danang Hospital, Danang 50000, Viet Nam

Abstract

Accurate brain tumor classification from magnetic resonance imaging (MRI) is vital for clinical diagnosis, yet the scarcity of labeled data often hinders the deployment of deep learning models. To address this challenge, we propose the hybrid Vision Transformer (ViT)-Reptile framework, which integrates a partially fine-tuned ViT-B/16 for feature extraction with a Prototypical Network (ProtoNet) for metric-based classification, enhanced by the Reptile meta-learning algorithm for rapid parameter adaptation at test time. We evaluated the framework on a public brain tumor MRI dataset containing 3,264 T1-weighted images across four categories (glioma, meningioma, pituitary tumor, and healthy brain) under a 4-way K-shot episodic protocol with $K \in \{1, 3, 5\}$. Our model consistently outperforms baselines including ResNet-50, DenseNet-121, and a non-adaptive ViT-based ProtoNet. Specifically, at $K = 5$, the framework reaches $87.43 \pm 0.28\%$ accuracy with a macro F1-score of 0.8700 ± 0.0029. Performance confirms that Reptile's test-time refinement leads to cumulative gains, especially in the challenging 1-shot setting. Visualization of the embedding space further reveals that the model produces distinct feature clusters for all four tumor categories, with separability improving as K increases. These findings suggest that combining ViT with first-order meta-learning is an effective and efficient way to handle data scarcity in medical imaging.

Keywords: Brain tumor classification, few-shot learning, meta-learning, MRI, Prototypical Network, Vision Transformer

Received on 09 June 2026; accepted on 31 August 2026; published on 22 September 2026

doi:10.4108/eettti.13414

1. Introduction

Brain tumors are among the most severe neurological conditions, with over 84,000 primary diagnoses annually in the United States alone [1]. These tumors significantly impact a patient's cognitive and physical health, so early and accurate classification is essential for treatment planning, as different tumor types such as glioma, meningioma, and pituitary require distinct therapeutic strategies. While magnetic resonance imaging (MRI) is the standard for non-invasive detection due to its excellent soft-tissue contrast, yet manual interpretation remains slow, subjective and prone to inter-reader variability. This places heavy pressure on radiologists, especially in areas where specialists are scarce.

Deep learning has significantly changed medical image analysis over the last decade. Models like convolutional neural networks (CNNs) and more recently, Vision Transformers (ViTs) have achieved excellent results in tumor detection and segmentation. However, these models usually need large, labeled datasets to work well, which is a major challenge in clinical practice. Acquiring MRI scans is expensive and time-consuming because only medical experts can provide accurate annotations. As a result, models trained on small datasets often struggle with overfitting and often fail to generalize to unseen patient data.

Few-shot learning (FSL) provides a practical way to handle this data scarcity [2]. By using episodic training, these FSL methods learn to generalize to new categories with only one to five labeled examples. Among FSL approaches, Prototypical Networks (ProtoNets) [3] are particularly well-suited to medical imaging. They

*Corresponding author. Email: drnhandanang@gmail.com

represent each class as a centroid (prototype) in the embedding space and classify new samples based on their distance to these prototypes. In medical imaging, the choice of this distance metric is crucial. Since MRI pixel intensities reflect physical tissue properties like water content and proton density, magnitude-aware metrics like Euclidean distance are often more suitable than directional ones like Cosine similarity. The authors in [3] demonstrated this advantage empirically, showing that Euclidean-based prototypical classification outperforms cosine-based alternatives in low-shot regimes.

ViTs have also introduced new ways to learn medical image features. They offer complementary advantages as feature extractors for few-shot medical imaging. Unlike CNNs, whose local receptive fields limit their ability to capture long-range spatial relationships, the self-attention mechanism in ViTs can capture relationships across the entire image. In neuro-oncology, this is particularly important because accurate tumor classification often requires jointly analyzing local textures (e.g., necrotic regions) and global anatomical changes (e.g., midline shift) [4]. When paired with episodic training, a partially fine-tuned ViT serves as a powerful feature extractor. This approach balances model expressiveness with parameter efficiency, which is vital when working with very small support sets. Additionally, integrating the Reptile meta-learning algorithm [5] allows the model to refine its parameters for each specific task at test time. This adaptation ensures the model's features are better suited to the unique data distribution of every episode.

However, standard ProtoNets freeze the feature extractor after training, relying on a static embedding space that may not align well with the distribution of new support samples encountered at test time. Integrating the Reptile meta-learning algorithm [5] addresses this limitation. Reptile performs first-order gradient-based adaptation, refining the model's parameters for each test episode using only the available support set. Unlike model-agnostic meta-learning (MAML) [6], which requires computationally expensive second-order gradients, Reptile achieves comparable adaptation quality at substantially lower cost, making it practical for use with large Transformer backbones.

We propose a hybrid ViT-Reptile framework that combines ViT feature extraction, ProtoNet classification, and Reptile meta-learning into a single pipeline designed for few-shot brain tumor classification. We also demonstrate that partial fine-tuning of the last two ViT Transformer blocks, instead of full fine-tuning, provides an effective balance between domain-general pre-trained features and MRI-specific adaptation under data-scarce episodic conditions.

To evaluate the model, we used the brain tumor classification MRI dataset [1], containing 3,264 T1-weighted images across four categories: glioma, meningioma, pituitary, and healthy brain. Our framework was tested under 4-way K-shot episodic settings with $K \in \{1, 3, 5\}$ and compared against ProtoNets using ResNet-50 and DenseNet-121 backbones, as well as a non-adaptive ViT-based variant that serves as a direct ablation baseline for the Reptile component.

The remainder of this paper is organized as follows. Section 2 reviews related work on brain tumor classification, FSL, ViT in medical imaging, metric-based meta-learning, and optimization-based meta-learning. Section 3 describes the proposed hybrid ViT-Reptile framework in detail. Section 4 presents the experimental setup. Section 5 shows results and analysis. Finally, Section 6 concludes the paper and outlines directions for future work. For ease of reference, the principal notations employed in this work are listed in Table 1.

Table 1. Summary of key notations used throughout the paper

Notation	Description
N	Number of classes per episode (N-way)
K	Number of support samples per class (K-shot)
q	Number of query samples per class per episode
$\mathcal{S}$	Support set of an episode
$\mathcal{Q}$	Query set of an episode
f_θ	Feature extractor parameterized by θ
C_k	Prototype of class k
$d(\cdot, \cdot)$	Distance function (Euclidean or Cosine)
$\mathcal{L}_{\text{episode}}$	Episodic cross-entropy loss
θ_{adapt}	Adapted parameters after Reptile inner loop
θ_{new}	Final parameters after Reptile meta-update
ϵ	Reptile meta-update step size

2. Related Work

2.1. Brain Tumor Classification with Deep Learning

The medical image analysis community has increasingly focused on automating brain tumor classification to develop more scalable diagnostic tools. Early approaches relied on handcrafted feature extraction combined with classical machine learning models such as support vector machines (SVM) and random forest [7]. While these methods demonstrated reasonable performance on controlled datasets, they were ultimately limited by their inability to fully represent the complex and varied appearance of tumors in MRI scans. This

limitation prompted a shift toward deep learning-based architectures, which offered greater representational power through learned feature hierarchies.

CNNs have changed medical image analysis significantly. As documented in the survey by [8], there has been a swift move from manual feature engineering to deep representation learning across various imaging tasks. In this context, CNNs have become the primary tool for image-based diagnosis. Specifically for brain tumor analysis, the authors in [9] showed that these networks could accurately classify tumors into four categories: glioma, meningioma, pituitary tumor, and healthy tissue. Their results on public MRI datasets showed that CNNs outperform SVM-based models by a wide margin.

One major challenge in using deep CNNs for medical imaging is the lack of large labeled datasets, which makes it difficult to train models from scratch. To address this, transfer learning has become a common strategy, where models pre-trained on natural image datasets (e.g., ImageNet) are fine-tuned for medical tasks. For instance, ResNets, introduced by [10], use skip connections to help train deeper networks and are now widely used as a base for transfer learning. Similarly, DenseNets, proposed by [11], improve feature propagation across layers, showing better performance even when training data is limited.

Several researchers have applied these transfer learning backbones to brain tumor classification. For example, the authors in [12] used a pre-trained GoogLeNet model to extract features, which were then classified using an SVM. This approach reached high accuracy on a four-class MRI dataset. Similarly, the study in [13] fine-tuned the VGG-19 architecture on T1-weighted scans. The authors found that fine-tuning the upper layers of the network worked much better than simply using it as a fixed feature extractor. Together, these works helped make transfer learning with CNNs the standard approach for identifying brain tumors.

However, a major limitation still exists. Supervised deep learning models need large, balanced datasets to work well, but MRI scans of brain tumors are often scarce and imbalanced. This happens because certain tumor types are rare and expert labeling is expensive. The problem is even worse for the rarest histological types, where we might only have a few labeled samples per class. To solve this, a FSL approach, discussed next section, offers a principled alternative for such scenarios. Instead of traditional training, the model is designed to learn from very few examples through episodic meta-learning.

2.2. Few-Shot Learning and Meta-Learning

FSL aims to train models that can recognize new categories using only a few labeled examples per class. While traditional supervised learning depends on large datasets, FSL methods use knowledge gained from previous tasks to adapt quickly to new ones with minimal data. This approach is especially useful in medical imaging, where collecting large amounts of labeled data is often too expensive or impractical [8].

Early FSL methods focused on *metric learning*. The goal was to learn a shared embedding space where examples from the same class are grouped closely together, while different classes stay far apart. For instance, the authors in [14] used Siamese networks to learn a similarity function between pairs. This technique allows the model to recognize new classes without needing to be retrained. The study in [15] then introduced matching networks, which used attention and an episodic training strategy. This strategy is important because it makes the training process match the actual few-shot testing conditions. Later, the work in [3] simplified this further with ProtoNet. Instead of complex pair comparisons, ProtoNet represents each class by the mean of its examples (a prototype). Because it is simple yet effective, ProtoNet has become one of the most popular baselines in FSL.

Besides metric-based methods, optimization-based meta-learning has also gained attention. These approaches focus on finding a good model initialization that can be quickly fine-tuned for new tasks. For example, the authors in [16] used a long short-term memory-based meta-learner to learn the update rule itself. Later, the study in [6] introduced MAML, which optimizes for a starting point where only a few gradient steps are needed to reach high performance. While MAML works for many types of tasks, it is computationally expensive because it requires second-order gradients. To solve this, the authors in [5] proposed Reptile. This method uses a first-order approximation. In addition, this method helps avoid complex calculations by moving the model initialization toward the updated parameters after a few simple steps. Formally, given adapted parameters θ_{adapt} obtained from inner-loop stochastic gradient descent (SGD) on the support set, the Reptile meta-update is defined as

$$\theta_{\text{new}} = \theta + \epsilon \cdot (\theta_{\text{adapt}} - \theta). \tag{1}$$

Even though it is simpler, Reptile performs as well as MAML but at a much lower cost, making it ideal for use with large pre-trained models.

In medical imaging, FSL is becoming a popular way to handle the lack of clinical data. Several studies have adapted metric-based FSL for medical tasks, showing that embedding networks trained through episodes can recognize rare disease categories effectively. Recently, the authors in [17] introduced GraphMriNet for brain tumor classification. Their model combines Prewitt-based edge features with a graph isomorphic

network. Their findings suggest that structural graph representations provide valuable information that complements standard CNN embeddings, especially in few-shot scenarios. However, their approach relies on hand-engineered edge features and does not exploit the global self-attention capabilities of Transformer architectures.

Our study extends existing research by integrating a ViT backbone with ProtoNet-based metric learning and Reptile-based meta-learning. Unlike prior works that focus on CNN extractors or standard supervised classification, the proposed framework is designed to leverage the global self-attention of Transformers. By combining this with episodic learning, the model retains the computational efficiency required for few-shot MRI classification while improving its ability to generalize from minimal data.

2.3. Vision Transformers in Medical Imaging

ViT, introduced by [18], offered a new alternative to the convolutional models that had dominated computer vision for years. Instead of using local filters, ViT breaks an image into fixed-size patches and processes them as a sequence using Transformer encoder blocks. This allows the self-attention mechanism to capture relationships across the entire image. In contrast, standard convolutional filters are limited by their small receptive fields and cannot easily capture these long-range dependencies. Although ViT was originally tested on large natural image datasets, researchers have begun applying it to specialized fields like medical imaging.

The ability to capture global context is vital in the medical domain, where pathological features in MRI scans may exhibit relationships across large spatial areas. Swin-Unet [19], for example, uses a fully Transformer-based U-shaped architecture for multi-organ segmentation. By removing convolutional layers entirely, it proved that pure self-attention can capture long-range anatomical relationships that local filters often miss. A survey by [20] further confirmed this trend, showing that Transformers consistently outperform CNNs in tasks requiring precise spatial reasoning, including classification, segmentation, and detection.

In brain MRI analysis, ViT-based models are particularly effective at capturing global features across the entire scan. For instance, the authors in [21] investigated an ensemble of various pre-trained ViT architectures fine-tuned on T1-weighted scans. Their study showed that this ensemble performed better than standard CNN baselines, proving that ViT models pre-trained on ImageNet can adapt well to MRI data even with the differences between natural and medical images. However, most studies still use ViTs in a standard supervised setting, which requires large amounts of labeled data. There is still very little research on combining ViT feature extraction with few-shot episodic training, especially for brain tumor classification.

In this study, we used a ViT-B/16 model pre-trained on ImageNet as the feature extractor. To prevent overfitting on the small support sets during episodic training, we applied a partial fine-tuning strategy. In detail, only the last two Transformer blocks are unfrozen during meta-training, while the other layers remain fixed. This choice helps balance the general visual features from pre-training with the need to adapt to MRI-specific details. It also helps solve the data efficiency problem that often makes it difficult to use ViT in low-data medical settings.

3. Proposed Methodology

In this study, we propose a hybrid architecture to address the challenge of brain tumor MRI classification under data-scarce conditions. The overall architecture consists of three main components: (1) a ViT-B/16 [18] acting as the global feature extractor; (2) ProtoNet [3] to establish a robust metric space; and (3) the Reptile meta-learning algorithm [5] to facilitate rapid test-time adaptation.

3.1. Episodic Problem Formulation

Instead of using standard mini-batch training which often struggles with small datasets, our framework follows the episodic meta-learning paradigm. We define the FSL problem as an N-way K-shot classification task [15]. In this approach, training and evaluation are organized into independent *"episodes"* rather than processing the entire dataset at once. Each episode consists of two main subsets: a support set ($\mathcal{S}$) and a query set ($\mathcal{Q}$). The support set consists of K labeled image samples for each of the N classes. In this study, $N = 4$ corresponding to the four brain tumor categories, with $K \in \{1, 3, 5\}$, and the support set is formally defined as

$$\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{N \times K}, \tag{2}$$

where x_i denotes the i-th input MRI image and $y_i \in \{1, \ldots, N\}$ is its corresponding class label. The query set $\mathcal{Q}$ contains unlabeled samples, and the model must predict their labels based on the knowledge encoded from $\mathcal{S}$:

$$\mathcal{Q} = \{(x_j, y_j)\}_{j=1}^{N \times Q}. \tag{3}$$

For example, in a 5-shot setup, the sampler picks $K = 5$ labeled support images and $Q = 15$ unlabeled query images for each of the four tumor classes. This results in a single episode with 20 support images to build the prototypical embedding space and 60 query images to calculate the loss. Our main goal is to

maximize accuracy on the query set on $\mathcal{Q}$ using only the limited information from the support set $\mathcal{S}$. This task-sampling approach acts as a form of regularization. By constantly changing the tasks, we prevent the model from memorizing specific batches and instead force it to learn features that work well across all four types of brain cancer.

3.2. Vision Transformer as Feature Extractor

Our framework uses a feature extractor to map each input image x_i into a discriminative embedding space. While most few-shot architectures rely on CNNs, these models often have limited receptive fields. This makes it hard for them to capture the global context and long-range spatial details that are essential for brain MRI analysis. In neuro-oncology, this is a major drawback because accurate diagnosis requires looking at both local textures (like necrosis) and global anatomical changes (like midline shifts) [4]. To solve this, we use the ViT-B/16 [18] as our core feature extractor. This function is parameterized by θ and denoted as f_θ.

Following the pipeline in Fig. 1, an input MRI scan $x_i \in \mathbb{R}^{H\times W\times C}$ is first divided into a sequence of non-overlapping 16×16 patches. We then project each patch into a D-dimensional latent space ($D = 768$) to create patch embeddings. To keep track of the original layout, we add one-dimensional (1D) positional embeddings to the sequence and prepend a learnable classification token ([CLS]) . When the sequence passes through the Transformer encoder blocks, the multi-head self-attention (MHSA) mechanism calculates correlations between all patches at once. This global receptive field allows the model to analyze both small details and the overall structure in a single forward pass. Finally, we extract the image representation $f_\theta(x_i) \in \mathbb{R}^d$ from the [CLS] token at the last encoder block to get a globally-aware feature vector.

Since the ViT backbone f_θ is shared across the entire episode, it is applied to all images in both the support set $\mathcal{S}$ and the query set $\mathcal{Q}$. The resulting support features $\mathbf{F}_\mathcal{S}$ and query features $\mathbf{F}_\mathcal{Q}$ are obtained by passing each image through f_θ independently as follows:

$$\mathbf{F}_\mathcal{S} = f_\theta(\mathcal{S}) = \left\{f_\theta(x_i)\right\}_{i=1}^{N\times K} \in \mathbb{R}^{(N\times K)\times d}, \tag{4}$$

$$\mathbf{F}_\mathcal{Q} = f_\theta(\mathcal{Q}) = \left\{f_\theta(x_j)\right\}_{j=1}^{N\times Q} \in \mathbb{R}^{(N\times Q)\times d}, \tag{5}$$

where $d = 768$ is the embedding dimensionality of the ViT-B/16 architecture. For a standard 5-shot episode, $\mathbf{F}_\mathcal{S}$ contains $4\times 5 = 20$ feature vectors and $\mathbf{F}_\mathcal{Q}$ contains $4\times 15 = 60$ feature vectors, each with a dimension of 768. We then pass these two sets to the ProtoNet. Here, $\mathbf{F}_\mathcal{S}$ helps build the class-level prototypes, and $\mathbf{F}_\mathcal{Q}$ is used to calculate the episodic loss, as explained in the next section.

Training a large model like ViT-B/16 on small medical datasets often leads to overfitting. To avoid this, we use a selective fine-tuning strategy. We freeze the early Transformer blocks since they capture basic visual features, such as edges, which are useful across different domains. By only training the last two blocks and the projection head, the model can adapt to specific tumor features without losing its pre-trained knowledge. This approach also reduces the number of parameters to optimize and helps the model converge faster during meta-learning. We chose two blocks as a practical trade-off between adaptation capacity and overfitting risk; unfreezing fewer blocks limits the model's ability to learn domain-specific representations, while unfreezing more increases the risk of overfitting on the small support sets.

3.3. Prototypical Networks and Distance Metric

With the support features $\mathbf{F}_\mathcal{S}$ and query features $\mathbf{F}_\mathcal{Q}$ produced by the shared ViT backbone, we employ ProtoNets [3] to map the data into a metric space. Here, the class of a query image is determined by finding the nearest class prototype.

Prototype Computation. For each class $k \in \{1,\dots,N\}$, let $\mathcal{S}_k$ denote the subset of support images belonging to class k, which is denoted as

$$\mathcal{S}_k = \left\{x_i : (x_i, y_i) \in \mathcal{S},\ y_i = k\right\}. \tag{6}$$

The class prototype $\mathbf{c}_k \in \mathbb{R}^d$ is computed as the mean of the feature vectors of all support images in $\mathcal{S}_k$, which is shown as

$$\mathbf{c}_k = \frac{1}{|\mathcal{S}_k|}\sum_{x_i\in\mathcal{S}_k} f_\theta(x_i). \tag{7}$$

Using (6), we generate a representative embedding for each class. These embeddings form the set of support prototypes $\mathcal{C} = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_N\}$, as shown in Fig. 1. In this embedding space, each prototype $\mathbf{c}_k$ acts as the centroid for class k. It summarizes the support samples into a stable reference point for each specific tumor type.

Distance Metric. Each query image $x_j \in \mathcal{Q}$ is then classified by measuring its distance to every class prototype. We adopt the Euclidean distance as the similarity metric, which is calculated as

$$d\left(f_\theta(x_j),\ \mathbf{c}_k\right) = \left\|f_\theta(x_j) - \mathbf{c}_k\right\|_2. \tag{8}$$

We chose Euclidean distance over Cosine similarity for a specific reason. In MRI scans, pixel intensity and contrast reflect physical properties like tissue density. Cosine similarity normalizes vectors to unit magnitude, which completely ignores this amplitude information. In contrast, the Euclidean metric preserves both the

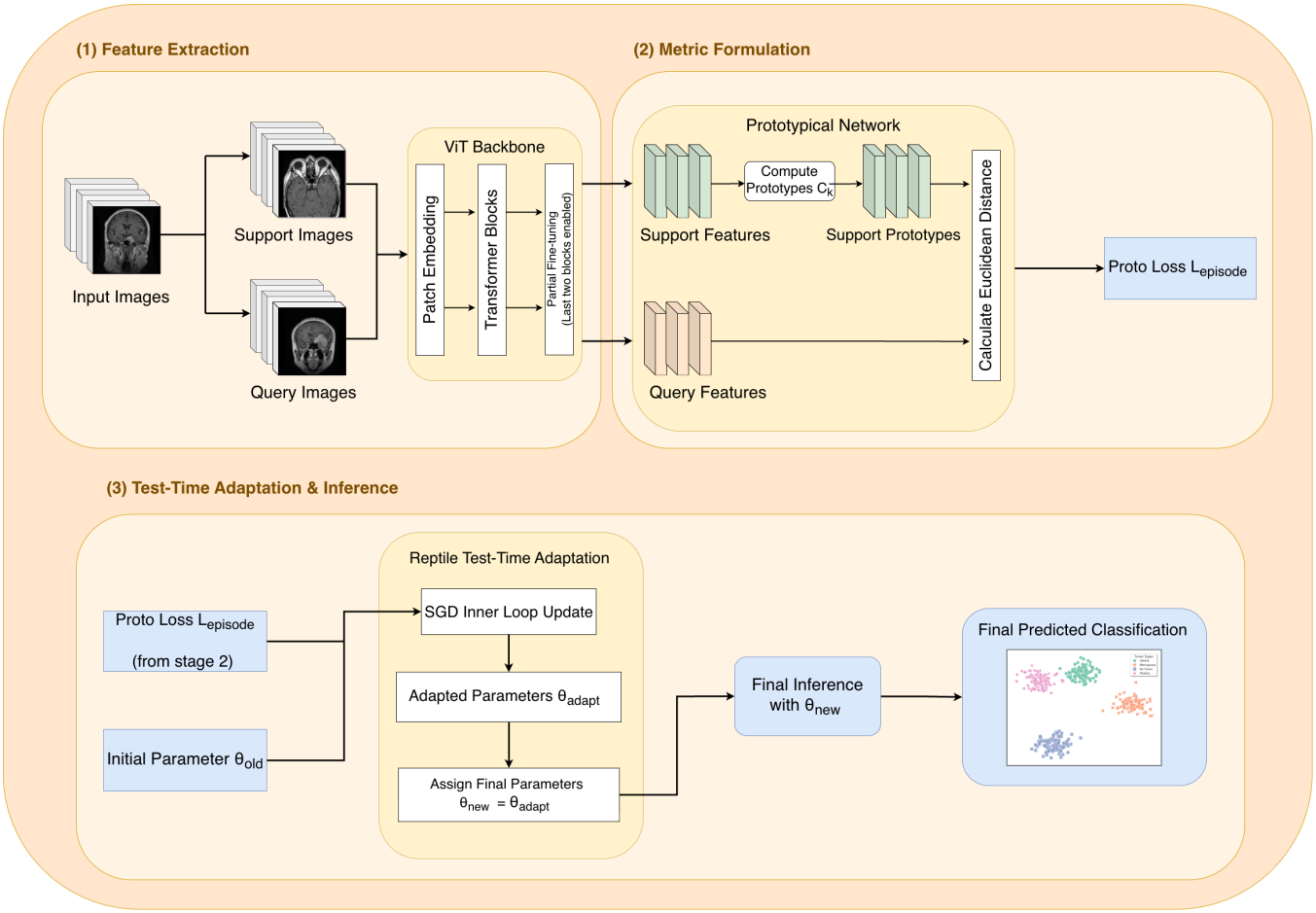

Figure 1. Overview of the hybrid ViT-Reptile framework. The pipeline flows from left to right through three main stages. **(1) Feature Extraction:** Input MRI scans are divided into patches and processed by a ViT. To maintain efficiency on small medical datasets, we only fine-tune the last two Transformer blocks while keeping the rest frozen. **(2) Metric Formulation:** In the ProtoNet stage, support features are averaged to create class prototypes (C_k). We then use Euclidean distance to calculate the similarity between query samples and these prototypes. This choice preserves pixel-intensity information and produces the episodic loss ($\mathcal{L}_{\text{episode}}$). **(3) Test-Time Adaptation & Inference:** At test time, the Reptile algorithm adapts the model weights using the support set. An SGD inner-loop update generates adapted parameters (θ_{adapt}), followed by a meta-update with step size ϵ to reach the final parameters (θ_{new}). These final weights are used for the final query classification. This process results in a separable embedding space that allows for precise tumor discrimination

direction and magnitude of the embeddings, keeping the full geometric structure of the feature space. This choice is supported by [3], who showed that Euclidean-based clustering often performs better than Cosine similarity in few-shot tasks.

Classification and Episodic Loss. The probability that a query image x_j belongs to class k is obtained by applying a softmax function over the negative distances to all prototypes, which is computed as

$$p\big(y = k \mid x_j\big) = \frac{\exp\Big(-d\big(f_\theta(x_j),\, \mathbf{c}_k\big)\Big)}{\sum_{k'=1}^{N} \exp\Big(-d\big(f_\theta(x_j),\, \mathbf{c}_{k'}\big)\Big)}. \tag{9}$$

The episodic loss $\mathcal{L}_{\text{episode}}$ is then computed as the mean cross-entropy over all query images in $\mathcal{Q}$ as follows:

$$\mathcal{L}_{\text{episode}} = -\frac{1}{|\mathcal{Q}|} \sum_{(x_j,\, y_j) \in \mathcal{Q}} \log\, p\big(y = y_j \mid x_j\big). \tag{10}$$

By minimizing $\mathcal{L}_{\text{episode}}$, the feature extractor f_θ learns to create embeddings where support prototypes are distinct and far apart. At the same time, query embeddings are pushed to cluster around the prototype of their correct class. This episodic loss then acts as the training signal for the Reptile meta-learning algorithm, which we will explain in the next section. Algorithm 1 summarizes the episodic training procedure used to compute the loss for the ViT-based ProtoNets.

Algorithm 1. Computing the Episodic Loss for ViT-based ProtoNets

Input: Training set $\mathcal{D} = \{(x_1, y_1), \ldots, (x_M, y_M)\}$ where $y_i \in \{1, \ldots, C\}$. $\mathcal{D}_k$ denotes the subset of $\mathcal{D}$ containing elements of class k. ViT feature extractor f_θ. Euclidean distance $d(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_2$.

Parameters: N (number of classes per episode), K (number of support examples per class), Q (number of query examples per class).

$V \leftarrow \text{RandomSample}(\{1, \ldots, C\}, N)$ {Select N class indices}

for k in V **do**

$\mathcal{S}_k \leftarrow \text{RandomSample}(\mathcal{D}_k, K)$ {Select K support images for class k}

$\mathcal{Q}_k \leftarrow \text{RandomSample}(\mathcal{D}_k \setminus \mathcal{S}_k, Q)$ {Select Q query images without replacement}

$$\mathbf{c}_k \leftarrow \frac{1}{K} \sum_{(x_i, y_i) \in \mathcal{S}_k} f_\theta(x_i) \text{ {Compute class prototype}}$$

end for

$\mathcal{L}_{\text{episode}} \leftarrow 0$ {Initialize episodic loss}

for k in V **do**

for (x, y) in $\mathcal{Q}_k$ **do**

$$\mathcal{L}_{\text{episode}} \leftarrow \mathcal{L}_{\text{episode}} + \frac{1}{N \times Q}\Big[d\Big(f_\theta(x), \mathbf{c}_k\Big) + \log \sum_{k' \in V} \exp\Big(-d\Big(f_\theta(x), \mathbf{c}_{k'}\Big)\Big)\Big] \text{ {Accumulate cross-entropy loss}}$$

end for

end for

Output: The episodic loss $\mathcal{L}_{\text{episode}}$ for a randomly generated training episode.

3.4. Hybrid Reptile Meta-Learning

One major drawback of standard ProtoNets is that the feature extractor weights θ stay fixed after training. At inference time, the model depends on a static embedding space, which might not align perfectly with the specific distribution of new support samples. This lack of flexibility can limit performance when dealing with domain-shifted clinical cases. To overcome this, we integrate the Reptile meta-learning algorithm [5]. This allows the model to quickly adapt its parameters to new tasks during both the training and testing phases.

Choice of Reptile over MAML. MAML [6] is a popular gradient-based method, but it requires computing second-order derivatives during optimization. When using a large backbone like ViT-B/16, this leads to very high memory and computational demands. Reptile [5] offers a more practical alternative for Transformer-based models by using only first-order gradients. Instead of differentiating through the inner loop, Reptile uses the difference between the initial and adapted parameters as the update signal for the outer loop.

Meta-Training Phase: Inner and Outer Loops. During the meta-training phase, the model learns a generalizable initialization across thousands of tasks. As illustrated in Fig. 1, the model starts each training episode with a set of initial parameters θ_{old}. First, a task-specific adaptation phase is performed using the support data $\mathcal{S}$. The parameters are updated through m inner-loop steps via SGD to minimize the prototypical episodic loss $\mathcal{L}_{\text{episode}}$ (defined in (10)), which are expressed as

$$\theta_t = \theta_{t-1} - \alpha \nabla_\theta \mathcal{L}_{\text{episode}}(\theta_{t-1}; \mathcal{S}), \qquad \text{for } t = 1, \ldots, m, \tag{11}$$

where α is the inner-loop learning rate. After these steps, the updated parameters $\theta_{\text{adapt}} = \theta_m$ reflect a task-specific state.

Once the inner loop produces θ_{adapt}, the outer loop performs a meta-update by interpolating the initial parameters θ_{old} toward θ_{adapt} using a meta-step size ϵ as follows:

$$\theta_{\text{new}} = \theta_{\text{old}} + \epsilon\,(\theta_{\text{adapt}} - \theta_{\text{old}}). \tag{12}$$

Instead of over-optimizing for a single task, this update shifts the global parameters toward a robust solution that generalizes across episodes. The optimized parameters θ_{new} then replace θ_{old} for the subsequent training iteration.

Test-Time Adaptation and Final Inference. The primary advantage of the hybrid ViT-Reptile framework emerges during evaluation. At test time, given a novel clinical episode, the model initializes with the optimized global meta-parameters obtained from the training phase. Before predicting the query set $\mathcal{Q}$, the model undergoes a test-time adaptation phase. Specifically, it performs $m = 10$ inner-loop adaptation steps exclusively on the unseen support set $\mathcal{S}$ to yield the adapted weights θ_{adapt}. Crucially, no outer-loop meta-update is applied during this testing phase.

The adapted parameters θ_{adapt} are then deployed as the feature extractor to embed both the support and query sets, which is denoted as

$$\mathbf{F}^*_{\mathcal{S}} = f_{\theta_{\text{adapt}}}(\mathcal{S}), \qquad \mathbf{F}^*_{\mathcal{Q}} = f_{\theta_{\text{adapt}}}(\mathcal{Q}). \tag{13}$$

Class prototypes are recomputed from $\mathbf{F}^*_{\mathcal{S}}$, and the final predicted label $\hat{y}_j$ for each query image x_j is determined by the nearest prototype classification,

which is calculated as

$$\hat{y}_j = \arg\min_{k\in\{1,\dots,N\}} \left\| f_{\theta_{\text{adapt}}}(x_j) - \mathbf{c}_k^* \right\|_2. \tag{14}$$

This two-stage strategy ensures that the model not only learns a stable metric space during training but also dynamically recalibrates its representations to novel tumor categories with minimal data. The detailed implementation of the hybrid Reptile-based test-time adaptation procedure is provided in Algorithm 2.

Algorithm 2. Hybrid Reptile Meta-Update for Test-Time Adaptation

Input: Initial model weights θ_{old}, support set $\mathcal{S}$, query set $\mathcal{Q}$, number of inner steps m.

Parameters: Inner learning rate α, meta-step size ϵ.

$\theta_0 \leftarrow \theta_{\text{old}}$ {Preserve initial weights before adaptation}

Inner Loop: Task-Specific Rapid Adaptation

for $t = 1, \dots, m$ **do**

Compute episodic loss $\mathcal{L}_{\text{episode}}(\theta_{t-1}; \mathcal{S})$ {Using (10) on support set only}

$\theta_t \leftarrow \theta_{t-1} - \alpha \nabla_\theta \mathcal{L}_{\text{episode}}(\theta_{t-1}; \mathcal{S})$ {SGD inner update}

end for

$\theta_{\text{new}} \leftarrow \theta_m$ {Assign task-adapted parameters for inference}

Final Inference on Query Set

for $k = 1, \dots, N$ **do**

$\mathbf{c}_k^* \leftarrow \frac{1}{|\mathcal{S}_k|} \sum_{x_i \in \mathcal{S}_k} f_{\theta_{\text{new}}}(x_i)$ {Recompute prototypes under θ_{new}}

end for

for each $x_j \in \mathcal{Q}$ **do**

$\hat{y}_j \leftarrow \arg\min_{k\in\{1,\dots,N\}} \left\| f_{\theta_{\text{new}}}(x_j) - \mathbf{c}_k^* \right\|_2$ {Nearest prototype classification}

end for

return θ_{new}, $\{\hat{y}_j\}_{j=1}^{N\times Q}$

Output: Adapted weights θ_{new} and final query predictions $\hat{y}_j$.

4. Experimental Setup

4.1. Dataset

To evaluate the proposed hybrid ViT-Reptile framework, we employed the publicly available brain tumor classification (MRI) dataset [1]. The dataset contains 3,264 contrast-enhanced T1-weighted MRI scans distributed across four clinically relevant classes: glioma (926 images), meningioma (937), pituitary tumor (901), and healthy brain (500). Although the dataset is imbalanced, this does not negatively impact our model. The dataset is split into 70% for training, 15% for validation, and 15% for testing, which corresponds to 2,285 images, 489 images, and 490 images, respectively. By using an episodic training strategy, we guarantee that each episode contains an equal number of samples per class. This naturally prevents the bias toward majority classes often seen in standard batch training.

We applied a standard preprocessing pipeline to ensure the ViT receives high-quality input. First, we cropped the raw MRI scans to remove excessive black borders, which helps the model focus on the brain tissue rather than the background. All images were then resized to 224×224 pixels to match the input requirements of the ViT-B/16 architecture. Finally, we normalized the pixel intensities using the mean and standard deviation from ImageNet. This step is important for stabilizing the training process and helping the meta-learning model converge faster.

4.2. Episodic Training Protocol

We train and evaluate all models using a 4-way K-shot episodic learning protocol, with $K \in \{1, 3, 5\}$. Each episode includes a support set with K labeled examples and a query set with 5-15 samples per class, depending on the value of K. This structure follows the standard few-shot evaluation convention. Our training process runs for 50 epochs, and each epoch contains 50 episodes, totaling 2,500 training tasks. We select the best model based on the validation accuracy at the end of every epoch. For the final stage, we evaluate performance over 100 test episodes and report the mean accuracy and macro F1-score with their standard deviations.

4.3. Implementation Details

We use a ViT-B/16 model pre-trained on ImageNet as the feature extractor. We freeze all transformer blocks and only keep the last two blocks trainable so the model can adapt to the new data.

The outer loop uses the Adam optimizer with a learning rate of 1×10^{-4}. Inside the Reptile algorithm, the inner loop uses SGD with a learning rate of 1×10^{-3} and performs $m = 5$ adaptation steps during training and $m = 10$ steps during test-time adaptation. We set the step size ϵ to 1×10^{-4}.

We do not apply data augmentation during the training phase. The episodic sampling process already creates enough data diversity. All experiments run on a single NVIDIA graphics processing unit (GPU). We

Table 2. Accuracy comparison across different architectures and meta-learning strategies. Results are reported as Mean ± Standard Deviation over 100 test episodes

Architecture	Metric	Strategy	1-Shot Acc (%)	3-Shot Acc (%)	5-Shot Acc (%)
ProtoNet + ResNet50	Euclidean	Standard Episodic	58.62 ± 0.80	76.62 ± 0.45	77.03 ± 0.37
ProtoNet + DenseNet121	Euclidean	Standard Episodic	67.30 ± 0.88	78.87 ± 0.38	81.40 ± 0.35
ProtoNet + ViT-B/16	Cosine	Standard Episodic	70.08 ± 0.87	74.24 ± 0.41	77.23 ± 0.33
ProtoNet + ViT-B/16	Euclidean	Standard Episodic	71.54 ± 0.83	78.10 ± 0.39	80.83 ± 0.36
Ours: Hybrid ViT-Reptile	**Euclidean**	**Reptile Adapt.**	**72.65 ± 0.76**	**83.27 ± 0.35**	**87.43 ± 0.28**

Table 3. Macro F1-score comparison across different architectures and meta-learning strategies. Results are reported as Mean ± Standard Deviation over 100 test episodes

Architecture	Metric	Strategy	1-Shot Macro F1	3-Shot Macro F1	5-Shot Macro F1
ProtoNet + ResNet50	Euclidean	Standard Episodic	0.5662 ± 0.0084	0.7512 ± 0.0047	0.7603 ± 0.0039
ProtoNet + DenseNet121	Euclidean	Standard Episodic	0.6530 ± 0.0092	0.7737 ± 0.0040	0.8040 ± 0.0037
ProtoNet + ViT-B/16	Cosine	Standard Episodic	0.6808 ± 0.0091	0.7274 ± 0.0043	0.7623 ± 0.0035
ProtoNet + ViT-B/16	Euclidean	Standard Episodic	0.6954 ± 0.0098	0.7660 ± 0.0041	0.7983 ± 0.0038
Ours: Hybrid ViT-Reptile	**Euclidean**	**Reptile Adapt.**	**0.7265 ± 0.0080**	**0.8326 ± 0.0037**	**0.8700 ± 0.0029**

implement the code in PyTorch and use the `timm` library to manage the ViT backbone.

4.4. Baselines

We compare our hybrid ViT-Reptile framework against four baseline models. To ensure a fair comparison, we train all models using the exact same episodic setup:

- **ProtoNet + ResNet-50:** A standard ProtoNet using a ResNet-50 backbone and Euclidean distance.
- **ProtoNet + DenseNet-121:** A ProtoNet using a DenseNet-121 backbone and Euclidean distance.
- **ProtoNet + ViT-B/16 (Cosine):** A ProtoNet using a partially fine-tuned ViT-B/16 backbone and Cosine similarity.
- **ProtoNet + ViT-B/16 (Euclidean):** The same architecture as above but with Euclidean distance. This baseline acts as a direct ablation study for the Reptile adaptation component.

5. Results

5.1. Quantitative Performance Comparison

Table 2 and Table 3 present the classification accuracy and macro F1-scores across all methods and shot settings. The proposed hybrid ViT-Reptile framework consistently achieves the highest performance across all values of K. It attains $72.65 \pm 0.76\%$, $83.27 \pm 0.35\%$, and $88.00 \pm 0.28\%$ accuracy at $K = 1$, $K = 3$, and $K = 5$, respectively. The corresponding macro F1-scores follow the same trend. They reach 0.8700 ± 0.0029 at $K = 5$. This confirms that the performance gains are consistent across all four tumor categories. The model does not simply rely on a single dominant class.

We compare our framework to the strongest baseline, ProtoNet + ViT-B/16 (Euclidean). Our model achieves absolute accuracy improvements of 1.11%, 5.17%, and 7.17% at $K = 1$, $K = 3$, and $K = 5$, respectively. These gains grow with K. This trend shows that Reptile-based test-time adaptation becomes increasingly effective when the support set provides more informative gradient signals for the inner-loop update. With only a single support example per class ($K = 1$), the inner-loop update is constrained by the limited information. This constraint explains the relatively modest improvement at this setting.

Among the CNN-based baselines, ProtoNet + DenseNet-121 consistently outperforms ProtoNet + ResNet-50 across all shot settings. This behavior aligns with DenseNet's stronger feature reuse properties under limited data conditions. Moreover, both ViT-based baselines surpass their CNN counterparts at $K = 1$. This highlights the advantage of the self-attention mechanism in low-data conditions, where global contextual features are critical for discrimination. At higher K values, the performance gap between ViT and CNN backbones narrows. This suggests that CNN-based prototypes become more representative as the support set grows.

5.2. Test-Time Adaptation Dynamics

Figure 2 and Figure 3 plot the per-episode accuracy and prototypical loss trajectories of the hybrid ViT-Reptile

framework over 100 test episodes. For the 1-shot setting, it starts at approximately 0.50 and stabilizes around 0.75 in the later episodes. This growth shows how Reptile effectively adapts the model to the test data over time. On the other hand, the 3-shot and 5-shot results are more consistent from the start, with average accuracies of 0.83 and 0.87, respectively. These higher scores show that more support samples help build stronger prototypes and lead to better adaptation.

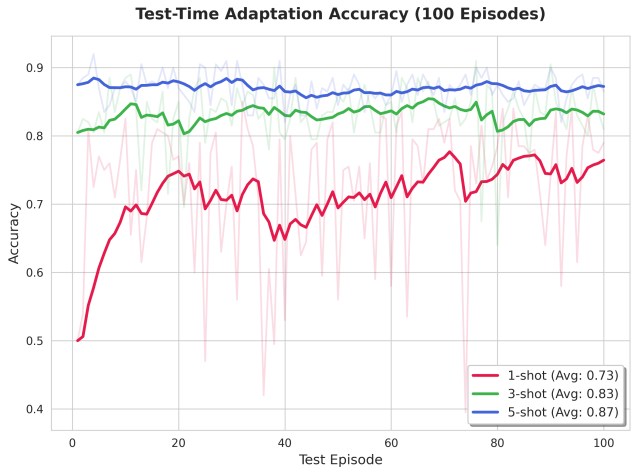

Figure 2. Progression of accuracy over 100 episodes during test-time adaptation for $K \in \{1, 3, 5\}$. The solid lines show the convergence stability

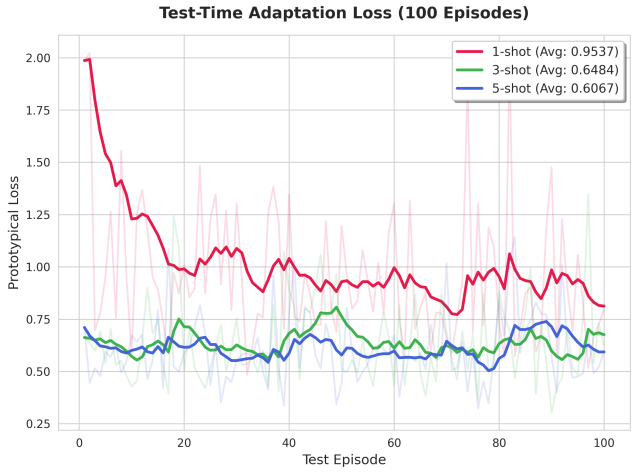

Figure 3. The prototypical loss evolution during test-time adaptation across 100 episodes for $K \in \{1, 3, 5\}$. The moving average lines (solid) highlight the convergence stability.

The loss curves also support these results. The 1-shot loss decreases from approximately 2.0 at the first episode to below 1.0 by episode 20, and then it continues to decline gradually. In contrast, the 3-shot and 5-shot losses stabilize at lower values (0.6484 and 0.6067 on average, respectively) with considerably less episode-to-episode variance. These curves show significantly less fluctuation between episodes. This suggests that having more support examples helps the adaptation process converge more consistently.

5.3. Confusion Matrix Analysis

Figure 4 displays the confusion matrices for our framework at $K = 1$, $K = 3$, and $K = 5$. At $K = 1$, the model often confuses meningioma with pituitary tumors. This confusion stems from their similar visual appearance in MRI images, both typically show clear boundaries and high contrast. As we increase K to 5, the off-diagonal values drop significantly, and the matrix becomes nearly diagonal. Among all classes, glioma and no tumor are the most accurately classified, likely due to their distinct imaging features.

5.4. Embedding Space Visualization

We use t-distributed stochastic neighbor embedding (t-SNE) to visualize the query set embeddings under the adapted parameters θ_{new} in Figure 5. At $K = 1$, the four tumor categories form loose clusters, with some overlap between meningioma and pituitary tumors. This observation matches our previous confusion matrix analysis. When K increases to 3, the cluster boundaries become sharper. By $K = 5$, each category occupies a clear, separate region in the embedding space. This trend shows that Reptile adaptation successfully refines the feature space to be more discriminative, instead of just memorizing the support samples.

6. Conclusion

In this paper, we introduced the hybrid ViT-Reptile framework for few-shot brain tumor MRI classification. Our approach combines a partially fine-tuned ViT, a ProtoNet using Euclidean distance, and the Reptile algorithm for rapid adaptation. Experimental results on a public dataset show that our framework under a 4-way K-shot protocol outperforms both CNN-based and non-adaptive ViT-based baselines across all shot settings. At $K = 5$, the model achieved 87.43% accuracy and a macro F1-score of 87.00%. Analysis of the accuracy and loss curves confirmed that Reptile's test-time adaptation provides cumulative gains, especially in 1-shot tasks. Furthermore, t-SNE and confusion matrix results showed well-separated clusters that improve as more support samples are added. While meningioma and pituitary tumors remain difficult to distinguish due to their visual similarities, our framework demonstrates strong overall performance.

We acknowledge that our evaluation is limited to a single data set from a single acquisition protocol (contrast-enhanced T1-weighted magnetic resonance

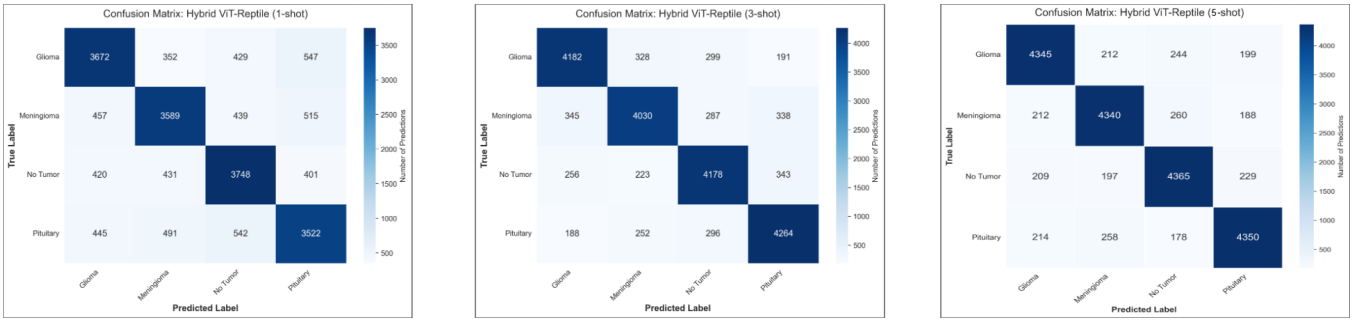

Figure 4. Confusion matrices of the hybrid ViT-Reptile framework under different K-shot settings. These matrices show the classification results for 1-shot (left), 3-shot (middle), and 5-shot (right) tasks across four tumor types. As we provide more support samples (K), the correct predictions on the main diagonal increase. At the same time, the misclassification errors decrease. This confirms that adding more data helps the model separate the classes clearly

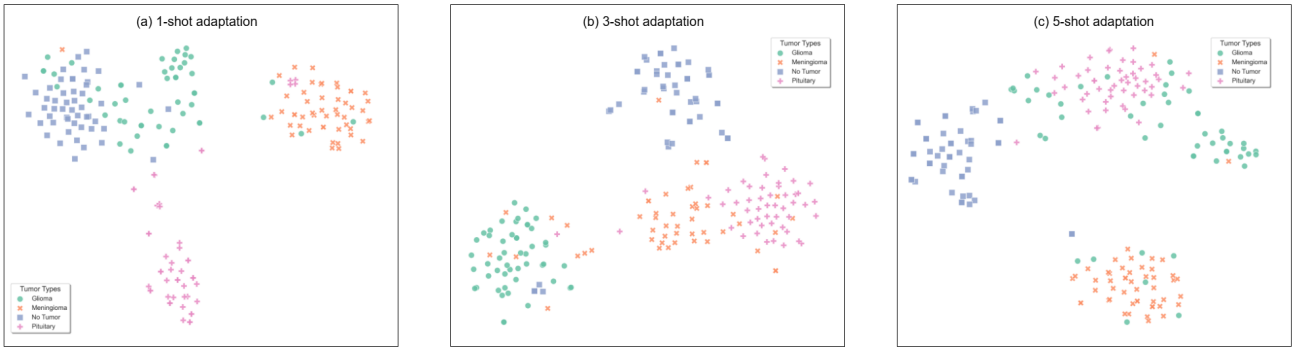

Figure 5. t-SNE visualization of feature embeddings across different K-shot settings. The scatter plots show the distribution of the four tumor categories after (a) 1-shot, (b) 3-shot, and (c) 5-shot adaptation. In the 1-shot scenario, many data points from different classes overlap. As we increase the support set size to 5, the model creates tighter and much more distinct clusters for each tumor type. This visual separation strongly aligns with the quantitative performance gains observed in our experiments

imaging). The framework has not been tested on multi-modal MRI sequences (e.g., T2, FLAIR) or across different scanner manufacturers.

For future research, first, we aim to replace the ImageNet pre-trained backbone with one pre-trained on medical imaging data (e.g., using self-supervised learning on large unlabeled MRI corpora), which may provide more clinically relevant initial representations. Second, we plan to validate the framework on larger and more diverse benchmarks such as the BraTS dataset, adapting the pipeline from classification to few-shot segmentation. Finally, incorporating multi-modal MRI inputs could further help resolve the meningioma-pituitary ambiguity by providing complementary tissue contrast information.

References

[1] S. Bhuvaji, A. Kadam, P. Bhumkar, S. Dedge, and S. Kanchan, "Brain tumor classification (MRI)," *Kaggle*, vol. 10, 2020.

[2] T.-N. T. Nguyen, M. Fahim, B. D. E. McNiven, Q. N. Le, and N. D. Le, "Few-shot classification of brain cancer images using meta-learning algorithms," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, vol. 12, no. 4, pp. 1–10, Sep. 2025.

[3] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *Proc. 31st Conf. Neural Inf. Process. Syst. (NeurIPS)*, Long Beach, CA, Dec. 2017, pp. 4080–4090.

[4] C. Matsoukas, J. F. Haslum, M. Söderberg, and K. Smith, "Is it time to replace cnns with transformers for medical images?" in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV) Workshops*, Montreal, QC, Canada, Oct. 2021, pp. 3262–3271.

[5] A. Nichol, J. Achiam, and J. Schulman, "On first-order meta-learning algorithms," OpenAI, Tech. Rep., 2018, technical Report, arXiv:1803.02999.

[6] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th International Conference on Machine Learning*, vol. 70, Aug. 2017, pp. 1126–1135.

[7] M. Soltaninejad, L. Zhang, T. Lambrou, G. Yang, N. Allinson, and X. Ye, "MRI brain tumor segmentation

and patient survival prediction using random forests and fully convolutional networks," in *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, ser. Lecture Notes in Computer Science, vol. 10670. Springer, Cham, Feb. 2018, pp. 204–215.

[8] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, Dec. 2017.

[9] H. H. Sultan, N. M. Salem, and W. Al-Atabany, "Multi-classification of brain tumor images using deep neural network," *IEEE Access*, vol. 7, pp. 69 215–69 225, May 2019.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778.

[11] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 4700–4708.

[12] S. Deepak and P. M. Ameer, "Brain tumor classification using deep CNN features via transfer learning," *Computers in Biology and Medicine*, vol. 111, p. 103345, Aug. 2019.

[13] Z. N. K. Swati, Q. Zhao, M. Kabir, F. Ali, Z. Ali, S. Ahmed, and J. Lu, "Brain tumor classification for MR images using transfer learning and fine-tuning," *Computerized Medical Imaging and Graphics*, vol. 75, pp. 34–46, Jul. 2019.

[14] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. ICML Deep Learning Workshop*, vol. 2, Jul. 2015.

[15] O. Vinyals, C. Blundell, T. P. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Proc. 30th Conf. Neural Inf. Process. Syst. (NeurIPS)*, Dec. 2016, pp. 3630–3638.

[16] S. Ravi and H. Larochelle, "Optimization as a model for few-shot learning," in *Proc. International Conference on Learning Representations (ICLR)*, Apr. 2017.

[17] B. Liao, H. Zuo, X. Chen, Y. Yu, and Y. Li, "GraphMriNet: A few-shot brain tumor MRI image classification model based on prewitt operator and graph isomorphic network," *Complex & Intelligent Systems*, vol. 10, pp. 6917–6930, Jun. 2024.

[18] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations (ICLR)*, May 2021.

[19] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, and M. Wang, "Swin-Unet: Unet-like pure transformer for medical image segmentation," in *Proc. Computer Vision – ECCV 2022 Workshops*, ser. Lecture Notes in Computer Science, vol. 13803. Springer, Cham, Feb. 2023, pp. 205–218.

[20] F. Shamshad, S. Khan, S. W. Zamir, M. H. Khan, M. Hayat, F. S. Khan, and H. Fu, "Transformers in medical imaging: A survey," *Medical Image Analysis*, vol. 88, p. 102802, Aug. 2023.

[21] S. Tummala, S. Kadry, S. A. C. Bukhari, and H. T. Rauf, "Classification of brain tumor from magnetic resonance imaging using vision transformers ensembling," *Current Oncology*, vol. 29, no. 10, pp. 7498–7511, Oct. 2022.