Unsupervised quantum machine learning and computer vision transformer methods for brain tumor classification

by

Dylan Couch

B.S., Mississippi State University, 2022

AN ABSTRACT OF A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Mike Wiegers Department of Electrical and Computer Engineering
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2026

Abstract

The early detection and diagnosis of brain tumors is linked with improved patient quality of life, treatment, and health outcomes. Identifying the presence of brain tumors is typically done using magnetic resonance imaging (MRI) scans. Creating a computerized system to determine whether a patient has a brain tumor, and what kind, remains challenging due to the complexity and visual similarity of brain MRI data. This challenge motivates the creation of methods and systems that can classify patient MRI data automatically without requiring supervised training.

We propose a quantum machine learning and computer vision transformer (ViT) framework for unsupervised brain MRI classification. The Quantum Transformer for Clustering MRI (QTCM) framework combines classical computing methods with a quantum ViT to better identify and classify brain tumors in MRI scan data. We demonstrate that a quantum transformer-based system can improve classification rates when compared to similar classical systems.

This dissertation also evaluates the role of clustering strategy selection in brain MRI tumor classification. In this study, we introduce *DQC1-projected*, a method inspired by deterministic quantum computation with one qubit, and compare its performance against several classical clustering algorithms across two public brain MRI benchmarks. The results show that *DQC1-projected* achieves the strongest performance among the evaluated methods.

Finally, we investigate the feasibility of QTCM on quantum hardware by extending the study from simulation to deployment on a present-day quantum device. These experiments evaluate the effects of hardware noise and resource limitations on the proposed method. Together, these results demonstrate the potential of quantum machine learning and computer ViT methods for advancing brain MRI analysis.

Unsupervised quantum machine learning and computer vision transformer methods for brain tumor classification

by

Dylan Couch

B.S., Mississippi State University, 2022

A DISSERTATION

submitted in partial fulfillment of the
requirements for the degree

DOCTOR OF PHILOSOPHY

Mike Wiegers Department of Electrical and Computer Engineering
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2026

Approved by:

Major Professor
Dr. Samee U. Khan

Copyright

Abstract

The early detection and diagnosis of brain tumors is linked with improved patient quality of life, treatment, and health outcomes. Identifying the presence of brain tumors is typically done using magnetic resonance imaging (MRI) scans. Creating a computerized system to determine whether a patient has a brain tumor, and what kind, remains challenging due to the complexity and visual similarity of brain MRI data. This challenge motivates the creation of methods and systems that can classify patient MRI data automatically without requiring supervised training.

We propose a quantum machine learning and computer vision transformer (ViT) framework for unsupervised brain MRI classification. The Quantum Transformer for Clustering MRI (QTCM) framework combines classical computing methods with a quantum ViT to better identify and classify brain tumors in MRI scan data. We demonstrate that a quantum transformer-based system can improve classification rates when compared to similar classical systems.

This dissertation also evaluates the role of clustering strategy selection in brain MRI tumor classification. In this study, we introduce *DQC1-projected*, a method inspired by deterministic quantum computation with one qubit, and compare its performance against several classical clustering algorithms across two public brain MRI benchmarks. The results show that *DQC1-projected* achieves the strongest performance among the evaluated methods.

Finally, we investigate the feasibility of QTCM on quantum hardware by extending the study from simulation to deployment on a present-day quantum device. These experiments evaluate the effects of hardware noise and resource limitations on the proposed method. Together, these results demonstrate the potential of quantum machine learning and computer ViT methods for advancing brain MRI analysis.

Table of Contents

List of Figures

List of Tables

Acknowledgments

The completion of this dissertation was made possible through the guidance, encouragement, and support of many individuals.

I would first like to thank my advisor, Dr. Samee U. Khan, for his mentorship throughout my graduate studies at both Mississippi State University and Kansas State University. His leadership and guidance helped me learn and experience a great deal about scientific discovery and research, and this work would not have been possible without his support.

I would also like to express my sincere gratitude to Dr. Asad W. Malik for his guidance, time, and thoughtful feedback throughout this study. His mentorship has been invaluable during my graduate studies.

I extend my appreciation to my committee members, Dr. Daniel Andresen, Dr. Xiaolong Guo, and Dr. Don Gruenbacher, for their time, valuable comments, and suggestions, which greatly improved the quality and direction of this work.

I would also like to acknowledge my fellow student researchers, Dr. Xuan-Bac Nguyen, Hoang-Quan Nguyen, and Ramin Salehi, for their support and collaboration throughout this journey.

This work was supported by the National Science Foundation under Grant Number 2211841.

Dedication

To my family, with love and gratitude.

Chapter 1

Introduction

Medical image analysis is integral in the diagnosis, monitoring, and treatment of neurological disorders. Among the available imaging technologies, magnetic resonance imaging (MRI) is especially important because it provides detailed, non-invasive visualization of brain anatomy and pathology. Brain MRI scans support the identification of tumors, structural abnormalities, disease progression, and other clinically relevant patterns. Accurate interpretation of these images can improve diagnosis, treatment, and patient outcomes[1].

Healthcare systems generate a substantial number of MRI scans to identify and diagnose patients, while interpreting scans requires time, expertise, and monetary investment by these healthcare systems. Estimates suggest that more than 150 million MRI scans are performed each year[2]. The need to classify tumor types in an increasing volume of MRI data creates a significant challenge. Automated analysis methods can assist this process by extracting visual features, identifying diagnostic patterns, and supporting clinical decision-making. However, brain MRI analysis remains challenging as MRI imaging data is high-dimensional, spatially complex, and contain variable tumor types.

This work studies unsupervised quantum machine learning and computer vision transformer (ViT) methods for brain tumor classification. The work is centered on MRI-based classification, with particular emphasis on clustering-based analysis, transformer-based representation learning, and quantum processing. The primary goal is to investigate whether these methods can improve unsupervised classification and interpretation of MRI scans for simulation-based evaluation and quantum hardware.

Brain tumor classification is a particularly important application of automated MRI analysis. Tumor categories such as glioma, meningioma, pituitary tumor, and no tumor may differ in location, texture, structure, and intensity. In some cases, these differences are visually distinct. In others, tumor types can appear similar in appearance on an MRI scan, making classification difficult. These challenges motivate models that can capture local image structure, broader relationships among image regions, and similarity relationships among image-derived features.

Deep learning has improved medical image classification by learning hierarchical image representations from data. Convolutional Neural Networks (CNNs) have been widely used for medical imaging tasks because they extract strong spatial features from image data[3;4]. However, convolutional models are often biased toward localized patterns. This can limit their ability to model complex relationships among image regions or among related samples. Computer ViT methods address this limitation through self-attention, which allows the model to evaluate relationships among different parts of an input representation[5–7].

At the same time, quantum machine learning has emerged as an area for exploring how quantum computation can support data representation, similarity evaluation, and hybrid learning[8–11]. Parameterized quantum circuits (PQCs), quantum kernels, and hybrid quantum-classical models are especially relevant in noisy intermediate-scale quantum (NISQ) settings, where current devices remain limited by noise, circuit depth, runtime, and available qubits[12;13]. These limitations prevent current quantum devices from replacing classical medical imaging systems, but they provide a realistic setting for studying selected quantum components within hybrid quantum-classical models.

1.1 Background and Motivation

Automated brain MRI analysis requires methods that can distinguish meaningful diagnostic structure from noise, artifacts, and anatomical variation. This task is difficult because tumor appearance may vary across patients, imaging conditions, and anatomical location. A classification method must therefore learn features that are useful across different images while remaining sensitive to the visual characteristics that separate tumor classes.

Machine learning and deep learning methods have been widely studied for brain tumor classification and related medical imaging tasks[7;14–17]. These methods can learn feature representations that are difficult to design manually. However, several challenges remain. Labeled medical data can be expensive and time-consuming to obtain. Class distributions can be uneven, with some tumor categories represented by fewer samples than others. Visually similar classes can also produce overlapping feature representations, increasing the likelihood of misclassification.

These issues motivate methods that can organize MRI feature representations, evaluate local sample relationships, and identify difficult or inconsistent samples. Visual clustering provides one way to address this problem. Instead of relying only on direct class prediction, clustering-based analysis evaluates how samples group together in feature space. When clustering is paired with transformer-based modeling, the system can evaluate relationships among samples and their local neighborhoods. When quantum-enhanced processing is introduced, the model can also study whether quantum representations and quantum-inspired similarity measures provide useful structure for medical imaging tasks.

The motivation for this dissertation is therefore threefold. First, this work aims to improve clustering-based brain MRI classification using hybrid quantum-classical modeling. Second, it evaluates whether quantum-enhanced components can improve the organization and interpretation of MRI feature clusters. Third, it studies whether selected stages of the proposed method can be executed on contemporary quantum hardware under realistic device constraints.

1.2 Brain Tumor Classification of MRI Data

Brain tumor classification refers to the task of assigning MRI images to clinically meaningful categories. In this dissertation, the primary classification setting involves four classes:

glioma, meningioma, pituitary tumor, and "no tumor". These categories are commonly used in publicly available brain tumor MRI datasets and provide useful benchmarks for evaluating automated medical image analysis methods[18–20].

Automated classification methods usually begin by converting raw images into numerical representations. These representations may be extracted through handcrafted features, convolutional models, transformer-based models, or hybrid methods. Once image features are extracted, a model can use them for class prediction, clustering, similarity evaluation, or downstream decision-making. The quality of the feature representation strongly affects the quality of the final classification result.

Brain MRI classification is challenging because tumor classes do not always form clearly separated visual categories. Glioma and meningioma images, for example, may share overlapping texture and boundary characteristics. Pituitary tumors may vary in size, location, and intensity. "No tumor" images may be easier to separate when the extracted features capture strong structural differences between normal and abnormal scans. These class-dependent differences make it useful to study not only final classification accuracy, but also the organization of samples in feature space.

Medical image analysis also requires caution in interpreting model performance. High accuracy on one dataset does not guarantee generalization to another dataset or clinical setting. Differences in imaging protocols, class balance, data quality, and patient population can affect model behavior. For this reason, studies typically evaluate methods using multiple datasets to identify more generalized solutions.

1.3 Unsupervised Visual Clustering for Medical Image Analysis

Unsupervised learning is important in medical image analysis because labeled medical datasets can be difficult and expensive to produce. Expert annotation requires time and domain knowledge, and class labels may be unevenly distributed across available datasets. In this setting, clustering-based methods are useful because they can organize samples according to feature similarity and reveal structure that may not be immediately visible from raw images alone.

Clustering is a machine learning technique that groups samples according to similarity. In visual clustering, the samples are images or image-derived feature vectors. For brain MRI analysis, clustering can organize MRI feature representations into local neighborhoods. These neighborhoods help reveal whether samples from the same diagnostic class occupy similar regions of feature space.

Many popular methods use clustering to perform classification tasks[16;17;21]. Clustering can be supervised or unsupervised. Supervised clustering uses known labels to guide the organization of samples. Unsupervised clustering seeks structure in the data without relying on predefined labels. In this dissertation, unsupervised visual clustering is used to organize feature representations and evaluate whether clustering structure can support brain tumor classification. This design is consistent with the broader goal of studying methods that can extract useful structure from MRI data while reducing reliance on direct fully supervised classification alone.

1.3.1 Clustering Algorithms

This dissertation evaluates several clustering algorithms popular in the literature. The clustering algorithms examined in this work include the following:

- ***k*-means**[22]. This centroid-based method partitions data into k groups by minimizing distance to cluster centers. It is simple and widely used, but it can be sensitive to initialization and may perform poorly when clusters are non-spherical or overlapping.

- **Fuzzy c-means**[23]. This method assigns soft membership values rather than hard cluster labels. It can be useful when samples lie near class boundaries, but it may still depend strongly on centroid structure.

- ***k*-nearest neighbors (*k*-NN)**[24]. This graph-based method uses local neighborhood relationships to identify similar samples. Popular for classification, k-NN is used to form local clusters around selected feature centroids.

- **Shared Nearest Neighbor (SNN)**[25]. This method evaluates similarity by comparing shared neighbors between samples. SNN can be useful when direct distances are less reliable or when local neighborhood structure is important.

- **DBSCAN (Density-Based Spatial Clustering of Applications with Noise)**[26]. This density-based method identifies clusters by grouping points in dense regions of the feature space. It can identify outliers, but its performance depends on density parameters and may degrade when clusters have different densities.

- **HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise)**[27]. This method extends density-based clustering by building a hierarchy of density-connected clusters. It can handle more complex structure than DBSCAN, but still depends on the density organization of the data.

- **Spectral clustering**[28]. This graph-based method uses eigenstructure to cluster samples based on similarity relationships. It can capture non-linear structure but may be computationally expensive for large datasets.

1.4 Computer Vision Transformers and Quantum Machine Learning

Computer ViT methods use attention mechanisms to evaluate relationships among elements of an image-derived representation[5]. Unlike convolutional models, which primarily emphasize local spatial features, ViT models can learn relationships across different regions or feature elements. This property is useful for large datasets, like those used in brain MRI classification tasks, as tumor-relevant information may not be confined to a single local region. Broader relationships among image features can help the model distinguish between visually similar classes.

In this dissertation, transformer-based modeling is used to evaluate structured representations derived from clustered MRI features. The model uses this structure to identify whether samples within a cluster are consistent with the centroid class. This makes transformer-based modeling useful not only for final classification, but also for evaluating local feature relationships in the clustering stage.

Classical feature extraction remains an important part of this process. Raw MRI images must be converted into numerical feature vectors before they can be clustered, encoded, or processed by quantum circuit components. In this dissertation, ResNet-50 is used as the classical feature extractor. ResNet-50 is a deep convolutional neural network introduced by He et al.[3]. It uses residual connections to improve the training of deep networks by allowing information to pass through shortcut paths.

Quantum machine learning studies how quantum computation can support learning tasks such as classification, clustering, similarity evaluation, and feature representation[8–10]. In near-term settings, quantum machine learning often uses hybrid quantum-classical systems. These systems combine classical computation with quantum circuit components that operate on encoded data.

Parameterized quantum circuits (PQCs) are central to many hybrid quantum-classical methods. A PQC contains trainable parameters that are optimized during training. Classical data are encoded into a quantum state, transformed by parameterized quantum operations, and measured to produce outputs that can be used by a classical optimizer.

Fig. 1.1 shows the basic structure of a PQC. The encoding layer maps classical data into a quantum state. The parameterized layer applies trainable quantum operations. The measurement layer converts the quantum state into output values that can be used for model training or prediction. This structure makes PQCs suitable for hybrid models in which classical feature extraction is combined with quantum-enhanced processing.

Deterministic quantum computation with one qubit (DQC1) is also relevant to this dissertation. DQC1 is a restricted quantum computing model introduced by Knill and Laflamme[29]. The model uses one clean qubit and a register of maximally mixed qubits. Despite this restricted structure, DQC1 can estimate properties of unitary operators that are useful for

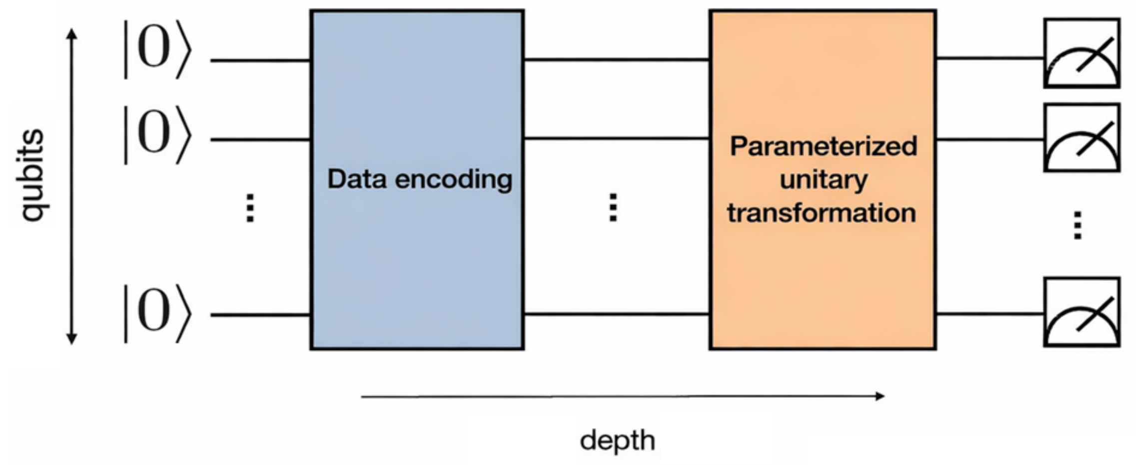


Figure 1.1: *Basic structure of a parameterized quantum circuit, showing the encoding of classical input data into a quantum state, a parameterized unitary transformation layer, and a quantum measurement layer.*

trace estimation and quantum kernel methods.

The standard DQC1 circuit prepares the clean qubit, applies a Hadamard gate, performs a controlled unitary operation, and measures the clean qubit. Measurements in the X and Y bases provide estimates of the real and imaginary components of the normalized trace of the unitary operator. This trace-estimation property makes DQC1 useful for quantum kernel methods and similarity evaluation[30;31].

DQC1 is used in Chapter 4 to develop *DQC1-projected*, a clustering method inspired by DQC1 trace estimation. The method maps feature vectors into unitary form, estimates similarity through a normalized-trace kernel, and uses an entropy-based criterion to guide cluster assignment. This provides a quantum-inspired alternative to classical clustering methods.

DQC1 is also relevant in the context of NISQ hardware. DQC1 circuits can generate quantum discord, which may be more resilient to noise than entanglement-dependent quantum correlations[32]. This does not remove the limitations of current quantum devices, but it makes DQC1 a useful model for studying constrained quantum computation in hybrid learning systems.

Current quantum devices remain limited by noise, circuit depth, runtime budgets, and number of available qubits[12;13]. These constraints are important, as the experiments in

Chapter 5 evaluate a system on quantum hardware. The hardware study does not claim that current quantum devices can replace classical medical imaging systems. Instead, it evaluates whether selected quantum stages of the proposed method can execute under current device constraints.

1.5 Problem Statement and Contributions

Automated brain tumor classification requires models that can learn meaningful structure from complex MRI data while remaining reliable across datasets and execution settings. Existing classical methods have achieved strong performance, but several limitations remain. CNN-based models can extract useful image features, but they may not fully capture broader relationships among samples or image regions. Transformer-based models improve relationship modeling, but they are not designed by default to evaluate quantum-enhanced feature representations. Classical clustering methods can organize feature vectors, but a fixed clustering algorithm may not perform consistently across datasets with different class balance, density structure, or visual overlap.

Quantum machine learning introduces additional possibilities for feature representation and similarity evaluation, but current quantum hardware imposes major constraints. NISQ systems are affected by noise, limited circuit depth, restricted runtime, and finite qubit availability[12;13]. As a result, quantum medical imaging methods must be studied as hybrid systems rather than as replacements for classical computation.

The problem addressed in this dissertation is the need for a clustering-based brain MRI classification method that combines classical feature extraction, transformer-based relationship modeling, and quantum-enhanced processing while remaining suitable for both simulation-based evaluation and constrained hardware-based execution. This problem requires a method with four capabilities. It must form meaningful MRI feature clusters, evaluate difficult samples within those clusters, adapt clustering behavior to dataset characteristics, and support selected quantum stages on available quantum devices.

The main contributions of this dissertation are as follows:

1. **Development of Proposed Solution.** We propose Quantum Transformer for Clustering MRI (QTCM), a hybrid quantum-classical framework for clustering-based brain MRI classification.

2. **Quantum-enhanced cluster evaluation.** The proposed method uses quantum circuit components to evaluate relationships within clustered MRI feature representations and identify hard or noisy samples, removing them from model training, improving classification outcomes.

3. **DQC1-projected clustering.** We introduce *DQC1-projected*, a quantum method that uses normalized-trace similarity to improve classification performance. We also compare the performance of DQC1-projected against other methods in the literature.

4. **Comparative evaluation across MRI datasets.** We evaluate our initial proposed solution across multiple datasets using Pairwise F-score, B-Cubed F-score, and normalized mutual information as evaluation metrics.

5. **Hardware execution study.** We deploy our system on IBM quantum hardware and identify how runtime limits, device noise, and other constraints affect performance of the system.

Collectively, these contributions describe our study of unsupervised quantum machine learning and computer ViT methods for brain tumor classification. They also provide a practical assessment of how such methods perform under both simulation-based and hardware-based settings.

1.6 Dissertation Organization

The organization of this dissertation is as follows.

Chapter 1 introduces the motivation, background, technical concepts, and contributions of the dissertation. It also introduces the major components used throughout the work, including brain MRI classification, unsupervised visual clustering, computer vision transformer methods, quantum machine learning, PQCs, DQC1, and QTCM.

Chapter 2 presents the related work. It reviews classical and quantum approaches to brain tumor classification, transformer-based medical image analysis, quantum machine learning, and quantum-enhanced clustering methods.

Chapter 3 introduces the QTCM framework for visual clustering of brain MRI data. It describes the model assumptions, cluster construction process, cosine similarity encoding, quantum state feature encoding, objective function, datasets, evaluation metrics, and experimental results.

Chapter 4 introduces the *DQC1-projected* method. It also compares *DQC1-projected* against several classical clustering algorithms and evaluates the proposed method on the *Brain Tumor Dataset* and the *Sartaj Dataset*.

Chapter 5 evaluates QTCM on IBM Quantum hardware. It describes the hardware execution method, IBM Quantum Heron backend, batching strategy, experimental settings, hardware results, and limitations of the device-based study.

Chapter 6 summarizes the dissertation findings, discusses limitations, and identifies future research directions for quantum-enhanced medical image analysis.

Chapter 7 lists the journal manuscripts submitted from the research presented in this dissertation.

Chapter 2

Literature Review

This chapter reviews prior work related to unsupervised quantum machine learning and computer vision transformer methods for brain tumor classification. The literature is organized around the major technical areas that support this dissertation: deep learning for brain tumor MRI classification, computer vision transformers, unsupervised visual clustering, quantum machine learning, quantum-enhanced clustering, and hybrid quantum-classical medical imaging systems. The following sections summarize prior studies and identify the research gaps that motivate the methods developed in later chapters.

Brain tumor classification from magnetic resonance imaging (MRI) has been studied using a wide range of computational methods. Classical machine learning methods, convolutional neural networks (CNNs), transformer-based models, and hybrid deep learning methods have all been applied to MRI classification and segmentation tasks. More recently, quantum machine learning methods have been investigated for medical image classification, quantum kernel estimation, and quantum-assisted clustering. These studies demonstrate the growing interest in combining classical image analysis with quantum or quantum-inspired computation. However, existing methods remain limited due to a few factors. Some prior works use supervised learning methods, which is not always viable for environments where labeled data is limited or too costly to obtain. Other studies rely on a single clustering strategy that can struggle when classifying input data that differ in characteristics from those used to train the model. Tables 2.1 and 2.2 summarize the contributions and limitations of studies discussed in this chapter.

Table 2.1: *Summary of related work in brain tumor MRI classification and transformer-based medical image analysis.*

Reference	**Contribution**	**Limitations**
Mohan and Ramkumar[21]	Employed MobileNet for brain tumor MRI classification, achieving an F-score of 93% while optimizing performance for computationally constrained devices.	Focused on efficiency under constraint rather than achieving the highest reported performance.
Mohan and Ramkumar[33]	Implemented an Inception V3-based CNN for glioma brain tumor detection, achieving an F-score of 98%.	Evaluated on a single, relatively small dataset, which may limit generalizability.
Padmakala and Maheswari[15]	Compared ResNet-101 and DenseNet-121 architectures for MRI-based brain tumor detection, reporting accuracies of 85% and 95%, respectively.	Evaluation was limited to smaller datasets, which can affect reliability across broader imaging conditions.
Montalbo[7]	Proposed *TUMbRAIN*, a transformer-based method using ViT, MobileViT, and Swin Transformer components, achieving 97.94% accuracy.	Transformer-based methods can increase computational cost and often require large training datasets.
Albalawi et al.[16]	Introduced a federated learning method using VGG16 to support brain tumor classification while preserving patient data privacy, achieving 98% accuracy.	Added complexity due to the privacy-preserving training structure.
Balasubramanian et al.[17]	Developed a Residual Fused Shepherd CNN (RF-ShCNN) by combining CNN and residual network components, achieving 93.7% accuracy.	High training cost and limited interpretability remain concerns.

Table 2.2: *Summary of related work in quantum classification and quantum-enhanced clustering.*

Reference	Contribution	Limitations
Amin et al.[14]	Integrated a quantum variational classifier (QVC) with classical feature extraction for brain tumor detection.	Practical use remains constrained by quantum hardware limitations.
Pandi and Thiruvenkadam[34]	Proposed a hybrid quantum neural network using a parameterized quantum circuit, achieving 96.97% accuracy.	Restricted to shallow circuits and limited qubit counts.
Akinrotimi et al.[35]	Applied quantum support vector machines to multimodal brain imaging data, including MRI, PET, and fMRI.	Focused on supervised classification rather than unsupervised clustering.
Poggiali et al.[36]	Proposed a quantum-assisted k-means clustering method using quantum circuits for distance computation while retaining classical preprocessing.	Limited by assumptions associated with k-means clustering and dataset structure.

2.1 Deep Learning Methods for Brain Tumor Classification

Deep learning methods have become central to automated brain tumor MRI analysis because they can learn image features directly from data. CNNs are especially important in this area because they are designed to capture spatial patterns, edges, textures, and hierarchical image features. These capabilities make CNNs suitable for brain MRI classification, where tumor categories may differ in shape, location, boundary structure, and intensity.

Traditional CNN architectures such as VGG16[37], ResNet[38], and Inception[39] have been widely applied to MRI classification. These models have demonstrated strong performance in multiclass classification tasks involving common categories such as glioma, meningioma, pituitary tumor, and no tumor. Their strength lies in extracting visual features from MRI images and transforming them into representations that can be used for classification.

Mohan and Ramkumar[21] employed MobileNet for brain tumor MRI classification and achieved a 93% F-score. Their method emphasized efficiency for computationally constrained devices. In later work, Mohan and Ramkumar[33] used an Inception V3 architecture and reported an F-score of 98%. This second study shifted attention from computational efficiency toward classification performance. These studies show that CNN-based models remain competitive for brain tumor classification, but their results also depend on dataset size, model configuration, and evaluation conditions.

ResNet-based methods are also important in medical imaging because residual connections allow deeper models to learn stable feature representations. Padmakala and Maheswari[15] compared ResNet-101 and DenseNet-121 architectures on MRI brain tumor datasets and reported accuracies of 85% and 95%, respectively. Their study supports the usefulness of deep residual and densely connected models for extracting MRI features, while also illustrating that model performance can vary substantially across datasets.

Several deep learning methods have combined CNN components with additional models. Balasubramanian et al.[17] introduced a Residual Fused Shepherd CNN (RF-ShCNN), combining shallow CNN components with a deep residual network for brain MRI classification. Their method achieved 93.7% accuracy across two MRI datasets. Albalawi et al.[16] proposed a federated learning method using VGG16 to support brain tumor classification while preserving patient privacy, reporting 98% accuracy. These studies show that deep learning methods can be adapted for performance, robustness, and privacy, but they also introduce tradeoffs in training cost, system complexity, and interpretability.

Although CNN-based methods remain important, they primarily emphasize localized spatial structure. Brain MRI classification may require a model to evaluate broader relationships among image regions or among extracted feature vectors. This limitation motivates the use of transformer-based methods and clustering-based analysis, both of which are important to this dissertation.

2.2 Computer Vision Transformers for Medical Image Analysis

Transformer models were introduced to model relationships among elements of an input sequence using attention mechanisms[5]. In computer vision, transformer-based methods extend this idea to image-derived representations. Instead of relying only on local convolutional operations, computer vision transformers can evaluate relationships across image patches, feature elements, or structured visual representations.

This property is important for medical image analysis because clinically meaningful information may not be confined to one local region. Brain tumors may be characterized by global anatomical context, intensity patterns, shape, boundary irregularity, or relationships among multiple image regions. Transformer-based models can therefore provide a complementary modeling strategy to CNNs by emphasizing relationships across broader parts of the image representation.

Montalbo[7] introduced *TUMbRAIN*, a transformer-based method for brain tumor classification using ViT, MobileViT, and Swin Transformer components. The study reported 97.94% accuracy across multiple MRI datasets and compared transformer-based methods against convolutional alternatives. The results support the value of transformer-based medical image analysis, while also highlighting an important limitation: transformer methods often require greater computational resources and larger training datasets than simpler CNN-based models.

Other transformer-based medical imaging studies have explored transfer learning, ensemble structures, and explainable artificial intelligence. Hossain et al.[6] investigated vision transformers, ensemble modeling, transfer learning, and explainable AI for brain tumor detection and classification. These methods reflect the broader trend of adapting computer vision transformers to medical imaging tasks, where accurate classification must be balanced with interpretability and computational feasibility.

For this dissertation, transformer-based modeling is important because the proposed method does not use attention only for direct image classification. Instead, transformer-based modeling is applied to structured representations derived from clustered MRI features. This distinction is important because the dissertation studies how transformer methods can support clustering-based brain MRI classification, particularly when combined with quantum-enhanced components.

2.3 Unsupervised Learning and Visual Clustering

Unsupervised learning is important in medical image analysis because labeled medical data can be difficult to obtain. Expert annotation requires time, clinical knowledge, and consistency. In addition, available datasets may have uneven class distributions or may not capture the full variability of clinical MRI data. These constraints motivate methods that can identify structure in image data without relying entirely on direct supervised classification.

Visual clustering applies clustering techniques to image data or image-derived feature vectors. In many systems, raw images are first transformed into numerical feature representations using CNNs or related feature extractors. Clustering algorithms then group the extracted features according to similarity. The resulting clusters can reveal relationships among visually similar samples, identify ambiguous cases, and expose structure that may be difficult to observe from raw images alone.

Visual clustering can be supervised or unsupervised. Supervised visual clustering uses known labels to guide representation learning or cluster formation. This can improve cluster separation when high-quality labels are available. However, supervised clustering becomes less practical when labeled data are scarce, uneven, or expensive to obtain. Unsupervised visual clustering instead seeks structure from the feature space itself. This is useful in medical imaging because it can support analysis of unlabeled or partially labeled data.

In the context of brain MRI analysis, unsupervised visual clustering is useful because tumor images may contain noise, artifacts, and class overlap. Glioma and meningioma samples, for example, may occupy nearby regions of feature space because of similarities in texture or boundary appearance. Clustering-based methods can help identify such difficult regions by organizing samples into local neighborhoods. These neighborhoods can then be evaluated for class consistency, ambiguity, or noise.

Clustering-based analysis is also relevant to quantum machine learning because quantum kernels, quantum distance measures, and quantum-inspired similarity measures can be used to evaluate relationships among feature vectors. This connection makes unsupervised visual clustering a natural setting for studying hybrid quantum-classical medical imaging methods.

2.3.1 Classical Clustering Methods

Classical clustering algorithms provide the basis for many visual clustering methods. Common approaches include k-nearest neighbors (k-NN)[40;41], hierarchical clustering[42;43], spectral clustering[40;41;44], partial clustering[45], probabilistic clustering[46], and density-based clustering[47]. Each method groups samples according to a particular definition of similarity, density, neighborhood structure, or graph connectivity.

The k-means algorithm is one of the most widely used centroid-based clustering methods[22]. It partitions data into k clusters by minimizing the distance between samples and their assigned centroids. Although k-means is computationally efficient, it assumes that clusters can be represented effectively by centroids. It can therefore struggle when clusters are non-spherical, overlapping, or unevenly distributed.

Fuzzy c-means extends centroid-based clustering by assigning soft membership values rather than hard cluster labels[23]. This can be useful when class boundaries are ambiguous, as is often the case in medical imaging. However, fuzzy c-means still depends on the structure of centroid-based clustering and may be sensitive to initialization or overlapping clusters.

Graph-based clustering methods use relationships among samples rather than only distances to centroids. The k-NN algorithm identifies local neighborhoods based on the closest samples in feature space[24]. Shared Nearest Neighbor (SNN) clustering evaluates similarity using the number of shared neighbors between data points[25]. These methods can be useful when local neighborhood relationships are more informative than global centroid structure.

Density-based methods such as DBSCAN and HDBSCAN identify clusters by locating dense regions of the feature space[26;27]. DBSCAN can identify outliers and does not require the number of clusters to be specified in advance. HDBSCAN extends this idea by building a hierarchy of density-connected clusters. These methods can be useful when clusters have irregular shapes, but their performance can depend strongly on density structure and parameter choices.

Spectral clustering uses graph eigenstructure to cluster data according to similarity relationships[28]. It can identify non-linear cluster structure, but it can also become computationally expensive for large datasets. This limitation is relevant to high-dimensional MRI feature spaces, where both clustering quality and computational feasibility must be considered.

Recent work has also investigated k-NN applications in quantum and classical systems. Related studies have used graph-based methods[48] and transformer-based clustering methods[49;50]. Other studies have examined distance choices for k-NN, including Euclidean, Hamming, polar, and Mahalanobis distances[51–54]. These studies show that clustering performance can depend not only on the algorithm, but also on the distance or similarity measure used to construct neighborhoods.

The diversity of classical clustering algorithms motivates the need for adaptable clustering selection. A method that relies on only one clustering algorithm may perform well on one dataset but poorly on another. This issue is especially important in MRI analysis, where datasets can differ in size, class balance, image quality, acquisition source, and feature-space structure.

2.4 Quantum Machine Learning for Classification

Quantum machine learning studies how quantum computation can be used for learning tasks such as classification, clustering, feature representation, and similarity evaluation[8–10]. In medical imaging, quantum machine learning has been investigated primarily through hybrid quantum-classical systems. These methods typically use classical computation for image preprocessing or feature extraction and quantum circuits for classification, kernel evaluation, or related operations.

Quantum classifiers are often motivated by the ability of quantum systems to represent data in high-dimensional Hilbert spaces. Quantum feature maps and quantum kernels can transform classical data into quantum representations, allowing inner products or similarity values to be evaluated in a quantum feature space[9;55]. In principle, this structure can support classification tasks with complex decision boundaries. In practice, performance depends on encoding choice, circuit depth, measurement strategy, optimizer behavior, and hardware constraints.

Amin et al.[14] integrated a quantum variational classifier as the decision layer in a hybrid model for brain tumor detection. Their method used classical feature extraction followed by a quantum variational classifier for four-class tumor classification. This study is important because it demonstrates one way quantum circuits can be incorporated into brain MRI analysis. However, the approach remains limited by scalability and quantum hardware constraints.

Pandi and Thiruvenkadam[34] proposed a hybrid quantum neural network using a shallow parameterized quantum circuit for brain tumor diagnosis. Their method reported 96.97% test accuracy. This study shows that parameterized quantum circuits can be applied to MRI classification, but it also illustrates a common limitation in quantum machine learning: shallow circuits and limited qubit counts restrict the size and complexity of the models that can be implemented.

Quantum support vector machines have also been explored for medical imaging tasks. Rebentrost et al.[55] formalized quantum support vector machines through quantum kernel methods. Akinrotimi et al.[35] applied quantum support vector machines to multimodal brain imaging data, including MRI, PET, and fMRI, for early detection of neurodegenerative disorders. Their results showed improved performance relative to a classical SVM baseline. However, this work focuses on supervised classification and does not address unsupervised visual clustering.

These studies indicate that quantum classifiers are a promising direction for medical image analysis, but they also reveal an important gap. Much of the existing quantum medical imaging literature emphasizes supervised classification. Fewer studies examine how quantum or quantum-inspired methods can improve clustering structure, sample organization, or feature-space similarity in unsupervised settings.

2.5 Quantum-Enhanced Clustering and Quantum Kernels

Quantum-enhanced clustering studies how quantum computation or quantum-inspired representations can support the grouping of data samples. This area is relevant to medical imaging because clustering quality depends heavily on the similarity measure used to compare samples. If quantum kernels or quantum-inspired similarity measures provide useful structure, they may help organize complex feature spaces such as those produced from MRI data.

Quantum clustering methods often use quantum circuits to compute distances, evaluate similarities, or construct kernel matrices. Poggiali et al.[36] proposed a quantum-assisted k-means method in which quantum circuits are used for distance computation while classical procedures handle preprocessing and centroid updates. This type of hybrid structure reflects the current state of near-term quantum machine learning: quantum components are used for selected operations, while classical systems handle the parts of the computation that remain more practical on conventional hardware.

Quantum kernels provide another route for clustering and classification. In quantum kernel methods, classical data are encoded into quantum states or quantum circuits, and similarities are computed using inner products or related measures in a quantum feature space[9]. These similarities can then be used for downstream learning tasks. In clustering, the quality of the kernel is important because it determines how the method represents the relationship between data points.

Blank et al.[56] developed a distance-based quantum classifier with a tailored quantum kernel based on quantum state fidelity between training and test data. Their method evaluates weighted powers of fidelity values, allowing the kernel to be adjusted through both the exponent applied to the fidelity and the weights assigned to training samples. The authors showed that the classifier is related to measuring the expectation value of a Helstrom operator, connecting quantum kernel classification with quantum states. More recently, Zhao et al.[57] proposed a Quantum Kernel Self-Attention Network (QKSAN) that combines quantum kernel methods with self-attention. In this model, classical data are mapped into a quantum Hilbert space through a quantum feature map, and the resulting quantum kernel self-attention score is used to represent relationships between data samples. These studies show that quantum kernels can be used not only for classification, but also as adaptable similarity structures for attention-based learning and clustering problems.

Deterministic quantum computation with one qubit (DQC1) is a method relevant to quantum kernels, as DQC1 can be used to estimate normalized traces of unitary operators[29]. Trace estimation can be interpreted as a similarity calculation when feature vectors are encoded into unitary representations. Ghobadi et al.[30] and Karimi et al.[31] investigated the role of DQC1 in machine learning and supervised learning settings. These studies support the use of DQC1 for trace estimation as a mechanism for constructing similarity values.

Entropy-based clustering is another relevant direction. Jenssen et al.[58] studied clustering using Rényi entropy, where cluster assignments are guided by entropy-based criteria. More recent DQC1-related clustering work uses similar information-theoretic ideas with quantum-inspired similarity measures[59]. This line of work is important because it connects quantum trace estimation, kernel construction, and clustering objectives.

Despite these developments, quantum clustering remains less studied than quantum classification in medical imaging. Existing methods often focus on general clustering tasks or on quantum-assisted versions of classical algorithms. Fewer works address MRI-specific visual clustering, class overlap, hard samples, and dataset-dependent clustering behavior. This gap motivates the DQC1-projected clustering method developed in Chapter 4.

2.6 Hybrid Quantum-Classical Systems for Medical Imaging

Hybrid quantum-classical systems are central to near-term quantum machine learning. Current quantum devices are limited by noise, circuit depth, runtime, and the number of available qubits[12;13]. As a result, fully quantum medical imaging systems are not practical for large MRI datasets. Hybrid systems address this limitation by assigning different parts of the computation to classical and quantum components.

In medical imaging, classical components are commonly used for image preprocessing, resizing, feature extraction, and data organization. Quantum components are then applied to selected tasks such as classification, similarity evaluation, or kernel construction. This design allows quantum methods to be studied without requiring the entire image analysis task to be executed on quantum hardware.

Amin et al.[14] used a hybrid approach in which InceptionV3 extracted classical image features and a quantum variational classifier performed the final classification. This structure shows how quantum layers can be added to classical feature extraction. Pandi and Thiruvenkadam[34] similarly used a hybrid quantum neural network with a shallow parameterized quantum circuit. These methods demonstrate the potential of quantum components in medical imaging, but they also depend on limited circuit sizes.

Hybrid quantum-classical methods are also relevant to clustering. Poggiali et al.[36] used classical preprocessing with quantum distance computation for k-means clustering. This type of design is important because it shows how quantum operations can be inserted into a larger classical learning process. However, when the clustering method remains tied to a single algorithm such as k-means, the resulting method may still inherit the limitations of that algorithm.

The dissertation builds on the hybrid quantum-classical direction by studying clustering-based brain MRI classification. Instead of using quantum circuits only as final classifiers, the proposed approach uses quantum-enhanced and quantum-inspired components to evaluate relationships within clustered MRI feature representations. This shift from direct supervised classification to clustering-based analysis is one of the main distinctions between this dissertation and much of the existing quantum medical imaging literature.

2.7 Research Gaps Addressed by This Dissertation

The reviewed literature shows substantial progress in automated brain tumor MRI classification, computer vision transformers, quantum machine learning, and hybrid quantum-classical systems. However, several gaps remain.

First, many brain tumor MRI classification methods are supervised or semi-supervised. Supervised methods can perform well when labeled data are available, but medical labels are costly and time-consuming to obtain. In addition, high performance on one labeled dataset does not guarantee generalization to other datasets or clinical settings. This motivates the study of unsupervised and clustering-based methods that can organize MRI feature representations with less dependence on direct supervised classification.

Second, many clustering-based methods rely on a fixed clustering algorithm. A single algorithm may not be suitable for all datasets, especially when MRI datasets differ in class balance, density structure, size, image quality, and feature-space organization. This limitation motivates the need for methods that can support multiple clustering choices and adapt clustering behavior to dataset characteristics.

Third, quantum medical imaging studies have focused more heavily on supervised classification than unsupervised clustering. Quantum variational classifiers, quantum neural networks, and quantum support vector machines have been studied for classification, but fewer works investigate how quantum or quantum-inspired methods can improve clustering structure, local neighborhood evaluation, or hard-sample identification in MRI data.

Fourth, the relationship between transformer-based visual modeling and quantum-enhanced clustering remains underdeveloped. Computer vision transformers provide a strong mechanism for modeling relationships among feature elements, while quantum machine learning provides tools for representation and similarity evaluation. Existing work has not sufficiently explored how these ideas can be combined for clustering-based brain MRI classification.

Fifth, many quantum machine learning studies remain simulation-focused. Simulation is important for method development, but it does not fully capture the constraints of quantum hardware. Current devices are affected by gate noise, limited circuit depth, finite runtime, restricted qubit availability, and queue-based access. These constraints must be considered when evaluating whether a quantum-enhanced medical imaging method can move beyond simulation.

This dissertation addresses these gaps by developing and evaluating a hybrid quantum-classical approach for clustering-based brain MRI classification. Chapter 3 introduces an initial proposed solution for visual clustering of brain MRI data. Chapter 4 introduces *DQC1-projected*, a quantum-inspired clustering method based on normalized-trace similarity. Chapter 5 evaluates the method on IBM Quantum hardware, providing an implementation study under current quantum device constraints.

Chapter 3

Visual Clustering Framework Using a Quantum Transformer

Portions of this chapter are adapted from the manuscript titled *QTCM: A Quantum Transformer Framework for MRI Brain Image Clustering*, currently submitted to *Quantum Machine Intelligence* for review.

D. Couch, X.-B. Nguyen, H.-Q. Nguyen, K. Luu, A. W. Malik, and S. U. Khan, "QTCM: A Quantum Transformer Framework for MRI Brain Image Clustering," submitted to *Quantum Machine Intelligence*, (2025). **Under Review**.

3.1 Introduction

Visual clustering refers to the application of clustering techniques to visual data, such as images or video frames. These techniques group visually related samples and often support interpretable visualizations of the resulting structure[45]. Typically, features are extracted from these input images or video frames using convolutional neural networks (CNNs)[3] or other similar methods. The extracted features are then partitioned into groups so that relationships between similar and dissimilar samples can be analyzed.

Visual clustering is a well-studied problem[48;60]. However, the expansion of feasible quantum computing systems has created new opportunities to study clustering through quantum and hybrid quantum-classical methods. Clustering tasks have been applied in areas such as facial recognition[44] and medical imaging[61]. These tasks are also increasingly integrated into systems used in everyday environments. One example is the Star Alliance Biometrics touchless facial recognition boarding system, which uses clustering techniques as part of its passenger identification process[62].

The objective of clustering is to partition an input dataset, $\mathcal{D}$, into a set of clusters. Each data point x_i in $\mathcal{D}$ is assigned to a cluster, and samples within the same cluster are expected to be more similar to one another than to samples in other clusters. The number of clusters, k, may be determined algorithmically or specified by the user, depending on the application. The desired output is a partition of the input dataset that maximizes intra-cluster similarity while minimizing inter-cluster similarity.

This optimization process is commonly expressed through an objective function. Classical approaches to clustering often optimize this objective through iterative methods, including spectral clustering[44], k-means clustering[63], hierarchical clustering[42], and density clustering[64]. These methods seek a partition of the input dataset that minimizes the defined clustering objective.

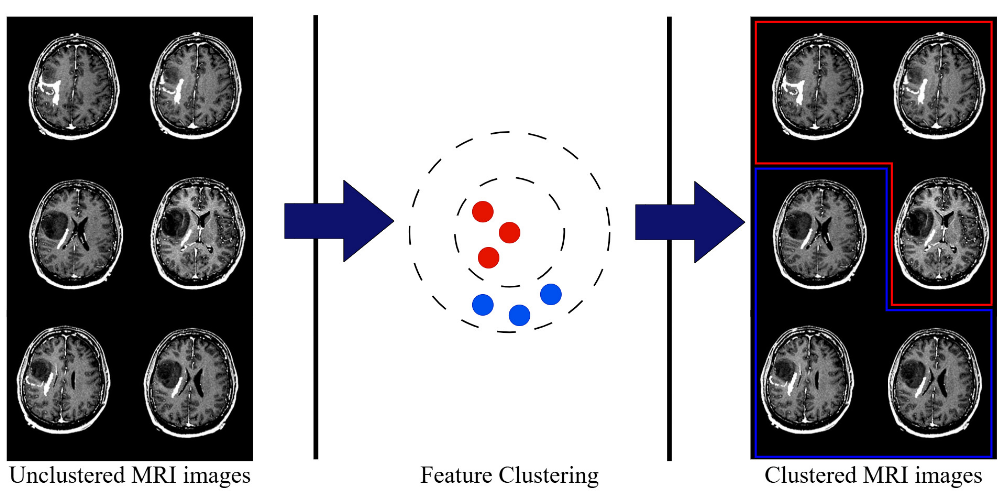


Figure 3.1: *Given a set of input images, clustering algorithms can assign proximity between similar images, given a feature encoding to group these images into classes. Our output shows 2 classes of 3 images each, shown in red and blue.*

3.2 Clustering Choice Based on Input Data Conditions

Visual clustering can be formulated as either a supervised or unsupervised learning problem. In supervised clustering, each input sample is associated with a known label. In unsupervised clustering, the input samples are unlabeled, and the model must infer group structure from the data itself. An example of clustered input features is shown in Fig. 3.1. The two problem formulations are described below.

3.2.1 Supervised Visual Clustering

Supervised visual clustering operates on labeled datasets, where each image or input sample is annotated with a known class or identity. The objective is to learn representations that map visually similar inputs into related clusters based on ground truth labels[65]. This process typically uses a loss function that increases intra-cluster similarity while reducing inter-cluster similarity.

Supervised clustering has been applied to tasks such as species identification[66] and handwriting detection[67]. Classical supervised clustering techniques often rely on convolutional architectures and metric learning. However, these approaches become less effective or unusable when labeled data are scarce, unevenly distributed, or unavailable.

3.2.2 Unsupervised Visual Clustering

Unsupervised visual clustering uses unlabeled input data. This formulation is important because labeled medical imaging data can be expensive, time-consuming, and difficult to obtain at scale. The clustering problem can be stated as the task of creating k clusters among D data points, where $D > k$, subject to a set of constraints. These constraints require each feature vector to belong to at most one cluster and require samples within a cluster to be more similar to one another than to samples in other clusters.

In this setting, clustering groups unlabeled data into class cohorts based on feature similarity. Improving clustering performance has practical implications for pattern recognition,

Table 3.1: *Notations and their meaning*

Symbols	**Meaning**
$\mathcal{D}$	Input dataset
N	Number of images in dataset
x_i	ith image in dataset
s_i	Feature vector of ith image
$\mathcal{F}$	Set of extracted features
n	Total number of qubits
$\lvert\psi\rangle$	Quantum state
$\mathcal{C}$	Set of clusters
C_i	ith cluster C
$\mathbf{c}_i$	Centroid of cluster C_i
S_i	Sequential representation of cluster C_i
$\boldsymbol{S}$	Sequential dataset
$\mathcal{K}$	k-nearest neighbors (k-NN) algorithm
k	Size of samples within cluster C_i
H	Predefined observable
$V_L(\theta)$	Parameterized layer with L operators
T	Transformer block
y_i	Binary sequence output of Transformer
y_i^h	hth sample of binary sequence y_i
$\hat{y} * i^{,h}$	Ground truth of Transformer output
$W_k, , W_q, , W_v$	Weighted key, query, and value matrices
$k_i, , q_i, , v_i$	Key, query, and value attention layer mappings
$a * i, j$	Self-attention coefficient
ρ	Density matrix

language processing, and computer vision. Traditional clustering methods, including k-means[63] and DBSCAN[64], can be limited by scalability. Large-scale datasets may become computationally expensive to process using classical systems alone. Quantum-enhanced clustering provides a potential strategy for exploring these large feature spaces by using quantum computation within a larger framework.

3.3 Model Assumptions and Problem Description

Magnetic Resonance Imaging (MRI) datasets often contain degraded samples that make model training more difficult and reduce classification performance. These degraded samples may include defects caused by noise, collection artifacts, patient movement, or environmental factors. The proposed system is designed to evaluate MRI data under the assumptions listed below. The most frequently used notations in this section and throughout the dissertation are listed in Table 3.1.

1. There is exactly one input dataset $\mathcal{D}$ that contains a specified number of visual input images x.
2. Each visual input image is assigned a unique image identifier. There are a total of N visual images in the dataset, and x_i $(1 \leqslant i \leqslant N)$ denotes a visual image identifier.
3. Each visual cluster is given a unique identifier. There are a total of C_n clusters for a given value of k.
4. The number of samples per cluster k is set by the user.
5. The number of qubits, n, and transformer blocks, T, in the system is known.
6. For visual clustering, we use a transformer model[49]. The system creates a sequential copy of the input dataset $\mathcal{D}$, denoted as $\boldsymbol{S}$. Subsequent processing steps access this sequential copy rather than the original dataset. This prevents the original input images from being overwritten, mismatched, or contaminated during processing.

The assumptions above define the data objects and model constraints used throughout the chapter. The next step is to describe how the input images are converted into clustered representations that can be evaluated by the quantum transformer.

3.3.1 Visual Cluster Dataset

To create a visual cluster dataset, each input image is converted into a classical feature vector using a feature extractor. These feature vectors are then organized into local clusters using a centroid-based nearest-neighbor procedure. Each cluster is converted into a similarity-based sequence that allows the transformer to evaluate relationships between the centroid and neighboring samples. The resulting classical representation is then encoded into a quantum state and processed by a parameterized quantum circuit (PQC). Finally, the model is trained using a binary loss function that penalizes incorrect predictions of whether each cluster member belongs to the same class as the centroid. The equations presented in this section define each stage of the proposed visual clustering framework in order.

The quantum transformer requires a structured clustered dataset rather than a collection of independent input images. Feature vectors form the feature set $\mathcal{F}$, which provides the representation used for neighborhood construction.

Next, the framework organizes individual features into local neighborhoods before they are processed by the quantum transformer. Each local neighborhood is defined relative to a centroid, $\mathbf{c}_i$. The k-nearest neighbors (k-NN) algorithm, denoted by $\mathcal{K}$, selects the k feature vectors most similar to this centroid. This produces a cluster C_i, where each row corresponds to one neighboring feature vector and each column corresponds to one feature dimension:

$$C_i = \mathcal{K}(\mathbf{c}_i, \mathcal{F}, k) \in \mathbb{R}^{k \times D}, \tag{3.1}$$

where k is the number of feature vectors within a single cluster, and D is the feature dimension.

This formulation allows the framework to evaluate smaller groups of visually similar samples centered around representative feature vectors. These clusters are combined into a visual cluster dataset, which serves as the structured input to the quantum transformer. The method used to construct clusters from the input dataset, including the use of the k-NN algorithm, is described in Algorithm 1.

Algorithm 1 Constructing Cluster Dataset from Input Dataset

Input: Input image dataset $\mathcal{D} = \{x_1, x_2, \ldots, x_n\}$, classical feature extractor $\Phi(\cdot)$, cluster size k, number of cluster centers M

Output: Clustered dataset $C = \{C_1, C_2, \ldots, C_M\}$

1: Extract feature set: $\mathcal{F} = \{f_i = \Phi(x_i) \mid x_i \in \mathcal{D}\}$
2: Sample M feature vectors from $\mathcal{F}$ as cluster centers: $\{\mathbf{c}_1, \ldots, \mathbf{c}_M\}$
3: **for** $i = 1$ to M **do**
4: Compute cosine similarity $\mathbf{e}_i$ between center $\mathbf{c}_i$ and all features $f_j \in \mathcal{F}$
5: Select the top-k most similar features (including $\mathbf{c}_i$)
6: Form cluster $C_i = \{f_{i_1}, f_{i_2}, \ldots, f_{i_k}\}$
7: **end for**
8: **return** cluster dataset $C = \{C_1, C_2, \ldots, C_M\}$

3.3.2 Cosine Similarity Encoding

Once each visual cluster has been constructed, the framework requires a representation that captures the relationship between the centroid and its neighboring samples. Cosine similarity is used because it measures the similarity between high-dimensional feature vectors. This is useful for image embeddings because visually related samples may have similar feature directions even when their feature magnitudes differ.

For each cluster C_i, the similarity between the centroid $\mathbf{c}_i$ and each neighboring feature vector $\mathbf{f}_i^{(j)}$ is computed to form a sequence of similarity scores. Following[49], let h be the position of an element in the sequential input, $\mathbf{e}_i^{(h)} \in \mathbb{R}^k$, and define the cosine similarity encoding as follows:

$$\mathbf{e}_i^{(h)} = \{\text{similarity}(\mathbf{c}_i, \mathbf{f}_i^{(j)})\}_{j=1}^{k}, \tag{3.2}$$

where k is the total number of neighboring samples in the cluster, j is the position of a feature vector within the cluster, $\mathbf{f}_i^{(j)}$ is the jth neighboring feature vector, and $\mathbf{c}_i$ is the cluster centroid.

The resulting sequence provides the transformer with a compact representation of the local cluster structure. Samples with high similarity values are expected to be more closely related to the centroid, while lower similarity values correspond to boundary cases, noisy samples, or visually similar samples from a different class. Cosine similarity encoding transforms each cluster from a set of feature vectors into a relational sequence that can be interpreted and processed by the transformer.

3.3.3 Quantum State Feature Encoding

Because the proposed framework includes a quantum circuit component, the classical feature or similarity representation must be mapped into a quantum-readable form. In this framework, amplitude encoding is used to encode classical vectors into quantum states. This encoding method represents a normalized classical vector using the amplitudes of a quantum state, allowing the quantum circuit to process information derived from the clustered MRI features.

The number of available qubits is a limiting factor, therefore the encoding method should represent the classical input using as few qubits as possible while preserving the information required by the model. To encode a D-dimensional feature vector into a quantum state, the minimum number of required qubits is $n = \lceil \log_2(D) \rceil$. Following[5], given n qubits, a general quantum state can be written as:

$$|\psi\rangle = \sum_{(q_1,\ldots,q_n)\in\{0,1\}^n} c_{q_1,\ldots,q_n} |q_1\rangle \otimes \cdots \otimes |q_n\rangle, \tag{3.3}$$

where $c_{q_1,\ldots,q_n} \in \mathbb{C}$ is the amplitude associated with each computational basis state.

In this expression, the coefficients $c_{q_1,\ldots,q_n}$ encode the classical information into the amplitudes of the quantum basis states. This step provides the interface between the classical feature extraction and clustering stages and the quantum circuit. Once encoded, the resulting quantum state can be transformed by the PQC.

After the classical representation has been encoded into a quantum state, the PQC applies trainable quantum operations to the encoded input. The PQC functions as a learnable quantum layer by transforming the encoded input according to trainable parameters θ. Its output is obtained by measuring a predefined observable H. Following[50], the expectation value of this observable is defined as:

$$\langle H \rangle = \langle 0|U^\dagger(\mathbf{z})V^\dagger(\theta)HV(\theta)U(\mathbf{z})|0\rangle. \tag{3.4}$$

Here, $U(\mathbf{z})$ encodes the input data into the quantum circuit, while $V(\theta)$ represents the trainable parameterized circuit layer. Measuring the observable H produces an expectation value that serves as the quantum circuit output. During training, the parameters θ are updated so that this output contributes to identifying consistent and inconsistent samples within each cluster.

3.3.4 Objective and Loss Functions

The quantum transformer evaluates the internal consistency of each constructed cluster by predicting whether each neighboring sample belongs to the same class as the cluster centroid. Given a cluster C_i with centroid $\mathbf{c}_i$, an ideal cluster would contain neighboring samples that share the same class label as the centroid. In practice, MRI datasets often include noisy, ambiguous, or visually similar samples that complicate this ideal partitioning. These samples are treated as hard samples because they may be incorrectly grouped with a centroid from another class.

For each cluster C_i, the quantum transformer produces a binary sequence y_i, where each element corresponds to one sample in the cluster. A value of $y_i^{(h)} = 1$ indicates that the hth sample is predicted to share the same class label as the centroid $\mathbf{c}_i$, whereas $y_i^{(h)} = 0$ indicates that the sample is predicted to belong to a different class. Because each cluster member is evaluated through a binary class-consistency decision, binary cross-entropy is used as the training objective. With the ground truth output denoted as $\hat{y}_i^h$, the loss for cluster C_i is defined as:

$$L_i(y_i, \hat{y}_i) = -\sum_{h=1}^{k} \left[\hat{y}_i^{(h)} \log(y_i^{(h)}) + (1 - \hat{y}_i^{(h)}) \log(1 - y_i^{(h)}) \right]. \tag{3.5}$$

Minimizing this loss encourages the model to assign higher probabilities to samples that share the centroid label and lower probabilities to samples that do not. The loss function aligns with the framework objective of identifying hard, noisy, or incorrectly grouped samples within similar clusters.

Collectively, these equations formalize the proposed framework. The cluster construction equation defines local neighborhoods of visually similar features, while the cosine similarity encoding converts each neighborhood into a sequence centered on the centroid. The quantum state equation describes the representation of this classical sequence in a quantum-readable form, and the PQC expectation equation defines the trainable quantum transformation applied to the encoded input. The binary cross-entropy loss then trains the model to determine whether each cluster member is consistent with the centroid label. This method ultimately identifies which samples are either hard or noisy within each cluster, allowing clustering performance to improve over similar methods.

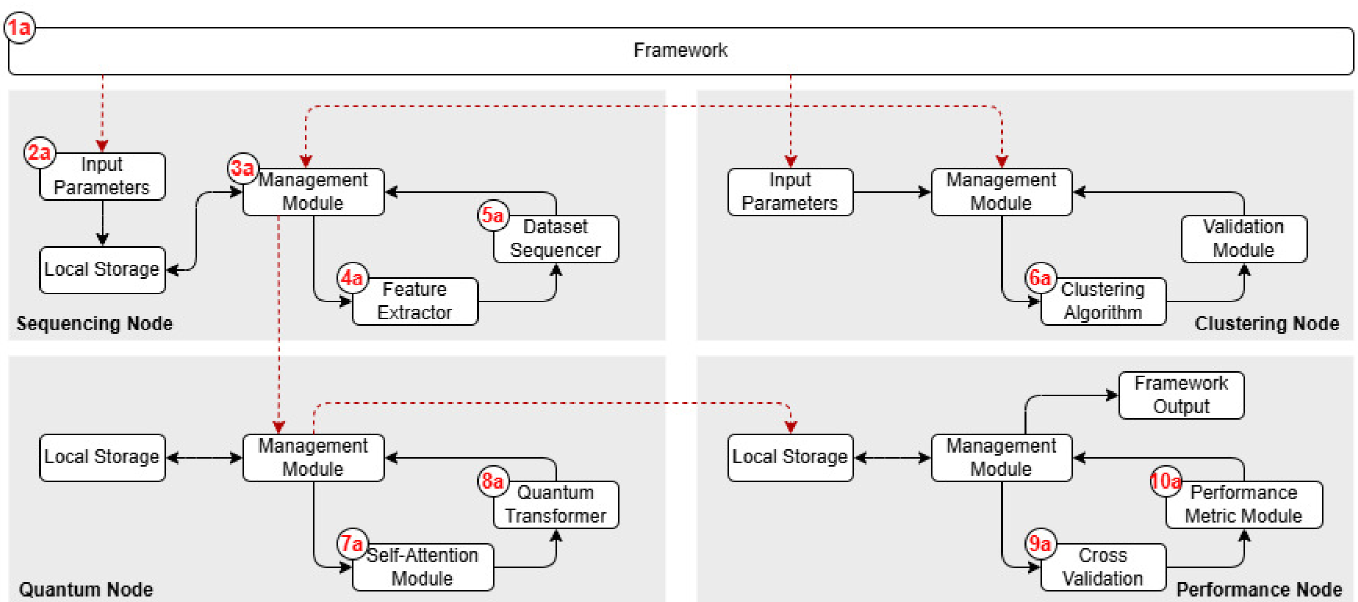


Figure 3.2: *Framework workflow order: Sequencing Node, Clustering Node, Quantum Node, Performance Node. Each node performs a specific set of tasks while communicating with other nodes via a Management Module and Local Storage*

3.4 Framework Architecture

Fig. 3.2 presents the node-based organization of QTCM. This figure shows how the major components are organized during execution. The Sequencing Node, Clustering Node, Quantum Node, and Performance Node each perform a distinct processing function while exchanging intermediate outputs through the Management Module and Local Storage.

The operating procedure of the framework is initialized by the user as shown in 1a. The user is prompted for a set of input parameters in 2a that are saved to local storage, as well as transferred to the various management modules in each of the four nodes, as shown in 3a. These input parameters are used throughout the workflow of the framework, and set performance characteristics including dataset name, feature extractor type, transformer type (quantum or non-quantum transformer), specified number of training epochs, testing and training image split, and value of k for k-nearest neighbor clustering. This framework is modular in design, where each node and corresponding controls can be modified via user-specified configuration changes for a given application context (e.g., differing resolution images, color channels, image size).

The first node to perform tasks in the workflow after user initialization is the sequencing node. After receiving input parameters from the user, the management module initiates feature extraction, shown in 4a, on a designated dataset using input parameters for the extraction method and dataset name. After feature extraction is completed, these features are sent, alongside corresponding class definitions, to the dataset sequencer, shown in 5a, to create a sequential dataset, which is the input format the quantum transformer requires. This sequenced dataset is sent back to the management module, which saves the dataset and, via local storage, it sends the dataset to the next node in the workflow, the clustering node.

The clustering node performs visual clustering on the newly sequenced dataset. After reading input parameters, this node applies a clustering algorithm, shown in 6a, to the sequenced dataset. In this proposed framework, we utilize a k-nearest neighbors (k-NN) clustering algorithm on our sequential dataset. This clustered dataset is sent to a validation module to verify if the value for k yielded a valid number of clusters, and thus a valid clustered dataset, or if the value for k must be changed to continue. The next node in our architecture, the quantum node, utilizes a quantum transformer to remove noisy and invalid entries in the clustered dataset created in the previous node. The management module reads the value for the run type input parameter, and if valid, sends the clustered dataset to the self-attention and quantum transformer, seen in 7a and 8a. The self-attention module performs feature encoding and encodes each classical feature vector into the same number of quantum states. These quantum states are then sent from the self-attention module to the quantum transformer, where the quantum states are transformed via a Parameterized Quantum Circuit (PQC), and sent back to the management module.

The last node in the workflow is the performance node, where the denoised clusters are validated and measured, with three performance metric outputs used to gauge the relative quality of the clustering method based on the set of input parameters. The first of the two modules in the performance node deals with cross-validation, as shown in 9a. This section corrects for overfitting in the framework, and relays information about the current framework operation to the management module, including k-fold information and testing and training split values as defined by the user as an input parameter. Next, the clustered dataset is sent to the performance metric module, seen in 10a, for the final step of the workflow. The performance metric module applies three performance metric techniques to the dataset to obtain Pairwise F-score (F_P), B-Cubed F-score (F_B), and normalized mutual information (NMI) values. These values are saved locally, along with the clustered dataset, as framework outputs and the workflow resets, ready to classify another set of input images.

3.5 Datasets

We used five brain MRI datasets in this section, each containing a varying number of images and subjects. Each dataset contains four class labels: glioma, meningioma, pituitary tumor, and "no tumor". The datasets used in this work were selected because they are relatively large, high-quality, and diverse collections created from images collected during real patient

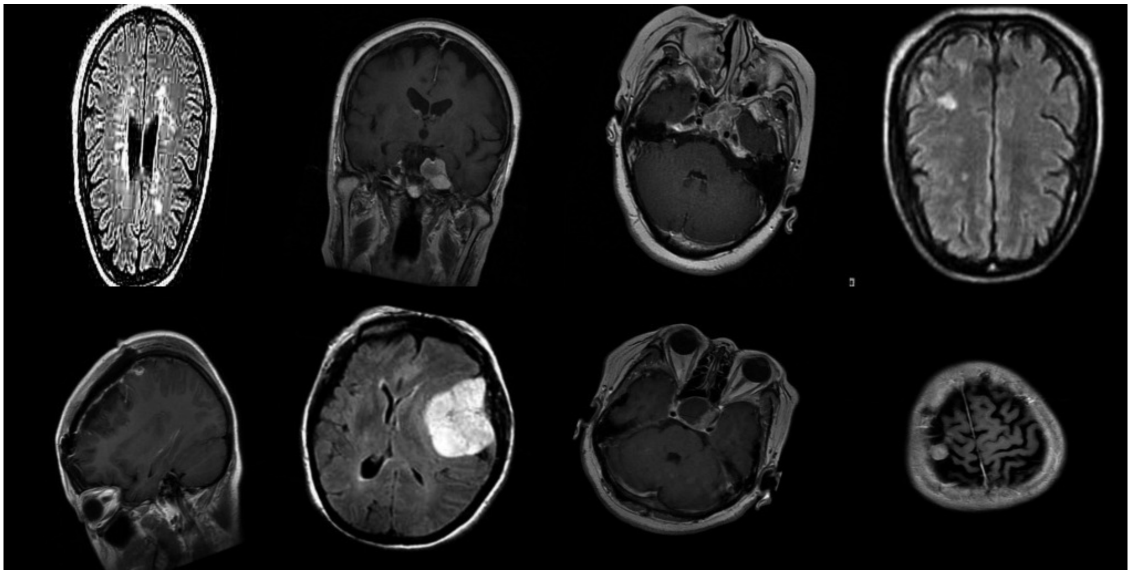


Figure 3.3: *Sample MRI input images from Brain MRI ND-5 (Dataset 1). Each of these input images belongs to one of the four brain MRI class identities including, no tumor (0), meningioma (1), glioma (2), and pituitary tumor (3)*

visits in hospital settings. They are also commonly used in the literature. Using five datasets creates a diverse experimental setting for evaluating the robustness of the proposed method. Example input images, along with accompanying class name and number from one of the datasets, are shown in Fig. 3.3.

- **Brain MRI ND-5**[68]: Brain MRI ND-5 (Dataset 1) contains 17,889 images, with 13,928 training images and 3,961 testing images. This dataset was created for use in machine learning systems performing tasks such as brain tumor classification and detection. It is the largest of the five datasets used in these experiments.

- **Brain Tumor MRI Dataset**[69]: Brain Tumor MRI Dataset (Dataset 2) contains 7,023 images, with a 5,712 training image and 1,311 testing image split. This dataset contains the typical four brain MRI classes, and is a combination dataset created from the accumulation of MRI images from three popular, smaller datasets.

- **brain-tumor-dataset**[70]: brain-tumor-dataset (Dataset 3) contains 4,292 images, with 3,276 training images and 1,016 testing images. This dataset is considered small for large-scale MRI applications and was chosen to see what effect having fewer training

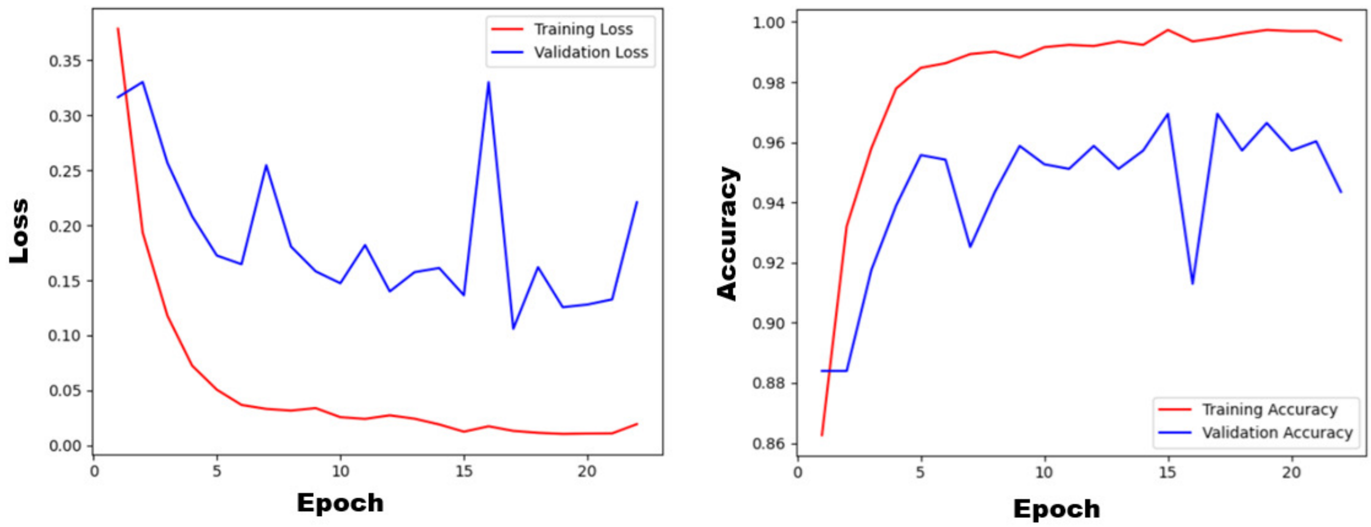


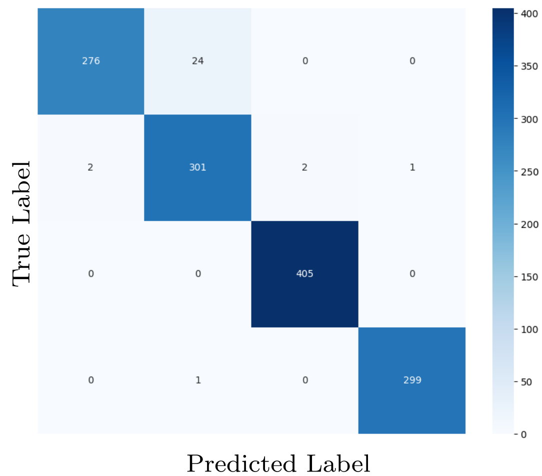
Figure 3.4: *Training and validation performance results of the classical feature extraction method (ResNet-50) on brain-tumor-dataset (Dataset 3) over 25 epochs. (Left) Training and validation loss results over 25 epochs. (Right) Training and validation accuracy results*

images would have on transformer performance due to lack of sufficient testing or training data.

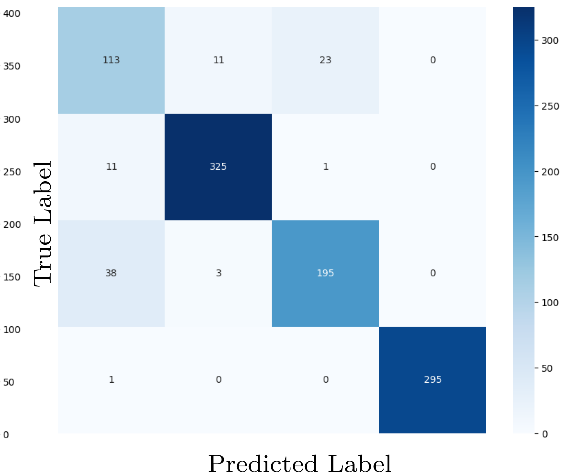
- **Brain Tumors 256x256**[71]: Brain Tumors 256x256 (Dataset 4) contains 3,096 images with no testing and training splits explicitly stated. In order to compare performance with the other datasets, the testing and training images for Dataset 4 were grouped using a traditional eighty-twenty training-testing split. This dataset, along with Dataset 3, was chosen due to its smaller size.

- **Brain Tumor Dataset**[72]: Brain Tumor Dataset (Dataset 5) has 7,023 images with 5,712 training images and 1,311 testing images. This dataset has an identical number of images in its training and testing image splits as Dataset 2, however the images in this dataset are from a singular source, rather than being a combination dataset, like those in Dataset 2.

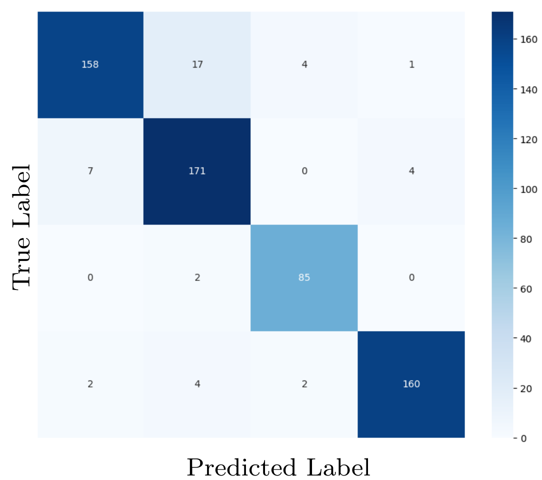
To create the required sequential feature inputs for the quantum transformer architecture, each dataset was preprocessed before clustering. Feature extraction was performed independently for each dataset using ResNet-50[3]. Fig. 3.4 shows the training and validation



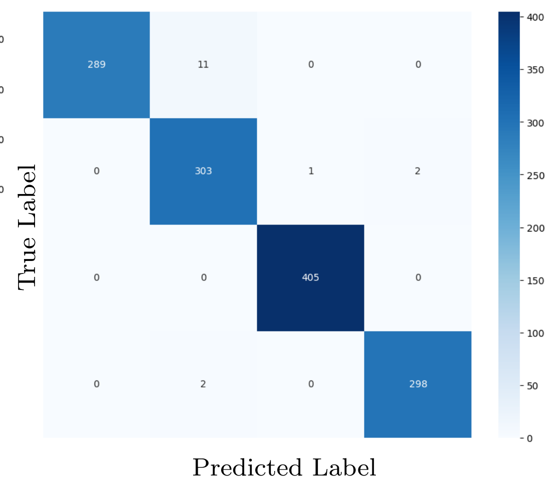
(a) Brain Tumor MRI Dataset (Dataset 2)



(b) brain-tumor-dataset (Dataset 3)



(c) Brain Tumors 256x256 (Dataset 4)



(d) Brain Tumor Data (Dataset 5)

Figure 3.5: *Confusion matrices for Datasets 2–5. For (a)-(d), the y-axis is the true classification label and the x-axis is the predicted classification label. Each column represents one of the four class types (glioma, meningioma, no tumor, and pituitary), respectively, for each dataset*

behavior of the ResNet-50 feature extractor on brain-tumor-dataset (Dataset 3). The loss and accuracy curves provide evidence that the feature extractor learned a stable representation before its outputs were used for clustering. This step is important because the quality of the extracted feature vectors directly affects the quality of the clusters.

The images in each dataset were aligned and cropped to the expected ResNet-50 input image resolution of 224x224. Fig. 3.5 contains the confusion matrices for Datasets 2–5 after feature extraction. These matrices show how well the extracted features separate the four diagnostic categories before the quantum transformer stage. We highlight a low error rate between predicted and true labels for each of the four shown datasets. The most common error across each dataset is with incorrect labels of the meningioma class, which is consistently the most difficult class label for other similar systems. Class-level errors in these matrices indicate where feature overlap remains and where later clustering and denoising steps are most needed.

3.6 Evaluation Metrics

In this section, we deploy our quantum clustering algorithmic approach on five distinct datasets comprised of brain MRI data. Each of the datasets used in this section was chosen based on characteristics such as dataset source, image quality, and total number of images. The number of datasets (five) was selected to provide a variety of input data to better measure the performance of the system on real-world image data and avoid overfitting on a single imaging source.

To evaluate the performance of our approach, we use three popular performance metrics, including Fowlkes–Mallows Score [73], B-Cubed F-score [74], and normalized mutual information (NMI) [75]. Fowlkes–Mallows Score is used to measure the performance of clustering tasks by calculating the distance between two clusters based on a set of points. This score is computed by taking the geometric mean of the precision and recall of the point pairs. Fowlkes–Mallows Score, also called the Pairwise F-score (F_P), is defined as:

$$F_P = \frac{\mathrm{TP}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})}}. \tag{3.6}$$

Here, TP (true positive) is the number of point pairs in the same cluster in both the ground truth and the prediction, FP (false positive) is the number of point pairs in the same cluster in the prediction but not in the ground truth, and FN (false negative) is the number of point pairs in the same cluster in the ground truth but not in the prediction.

B-Cubed F-score is a popular metric for evaluating clustering performance that focuses

Table 3.2: *Performance comparison using F_P, F_B, and NMI metrics across multiple brain MRI datasets.* ***Bold*** *values represent the best performance for each of the three metrics when compared across the five datasets*

Dataset	F_P **Prec.**	F_P **Rec.**	F_P	F_B **Prec.**	F_B **Rec.**	F_B	**NMI**
Brain MRI ND-5 [68]	95.18	93.27	94.21	95.04	94.06	94.55	90.03
Brain Tumor MRI Dataset [69]	98.16	95.41	96.76	96.50	96.11	96.30	93.70
brain-tumor-dataset [70]	98.17	97.17	**97.67**	98.07	96.87	**97.46**	**94.94**
Brain Tumors 256x256 [71]	94.52	80.27	86.81	94.74	75.45	84.00	69.63
Brain Tumor Dataset [72]	91.76	85.90	88.73	91.51	79.31	84.97	71.79

on each individual data point. B-Cubed F-score is similar to Pairwise F-score in that it is derived from precision and recall, but has a different focus, and is defined as:

$$F_B = \frac{1}{N} \sum_x \frac{2\,TP_x}{2\,TP_x + FP_x + FN_x}. \tag{3.7}$$

Normalized mutual information (NMI) is used as an additional clustering quality metric to measure the agreement between the predicted cluster assignments and the ground truth cluster labels. Unlike Pairwise F-score and B-Cubed F-score, which are based on precision and recall, NMI evaluates the amount of shared information between two cluster labelings. It is defined as:

$$\mathrm{NMI}(\Omega, C) = \frac{I(\Omega, C)}{\sqrt{H(\Omega)\,H(C)}}, \tag{3.8}$$

where Ω is the ground truth cluster set, $H(\Omega)$ and $H(C)$ are the entropies for cluster sets Ω and C, and $I(\Omega, C)$ is the mutual information between Ω and C [76].

3.7 Experimental Results and Discussion

This section presents the evaluation of QTCM against published methods and established benchmarks. Performance is measured in terms of clustering accuracy using Pairwise F-score (F_P), B-Cubed F-score (F_B), as well as normalized mutual information (NMI).

Table 3.3: *Clustering results on brain-tumor-dataset (Dataset 3) using the proposed k-NN quantum algorithm, with k=150. Time is given as HH:MM:SS, measured using an NVIDIA A30 PCIe GPU*

Performance Measurements	**Value**
Pairwise F-score	97.67
Precision	98.17
Recall	97.17
NMI	94.94
Nearest Neighbor Computation Time	00:01:17

Implementation Details. The clustering method is implemented using a consistent cluster size $k = 150$ for each dataset observed in this work. For each dataset, we deploy a k-nearest neighbors algorithm to create a corresponding cluster dataset, and retrain the clustering model between each experiment to avoid model overtraining. To create a more realistic and feasible hardware environment, we limited the number of transformers to T = 1 blocks.

Table 3.2 lists the performance of QTCM on five brain MRI datasets. We measured and recorded performance using Pairwise F-score, B-Cubed F-score, and NMI. Our proposed method achieves performance with a Pairwise F-score range from 86.81% to 97.67%, BCubed F-score from 84.00% to 97.46%, and NMI from 69.63% to 94.94%.

Table 3.3 details the recorded values for a complete run of the framework on the brain-tumor-dataset (Dataset 3). The performance metrics have consistent performance values for both Precision and Recall, as well as the derived metric Pairwise F-score. This result shows the framework is both accurate and robust at performing the clustering task on this dataset and its features.

3.7.1 Dimensionality Reduction

Principal Component Analysis (PCA)[77] and t-distributed Stochastic Neighbor Embedding (t-SNE)[78] are used to visualize the structure of the extracted feature space. These methods reduce high-dimensional feature vectors to lower-dimensional representations, making it possible to inspect whether samples from the same class tend to occupy similar regions.

Figs. 3.6 and 3.7 show the reduced feature representations for brain-tumor-dataset (Dataset 3). The PCA visualization in Fig. 3.6 shows clear separation for the no tumor and pituitary classes, while glioma and meningioma remain more intermixed. This pattern suggests that some diagnostic categories are easier to separate using the extracted features than others. The t-SNE visualization in Fig. 3.7 shows stronger class separation, indicating that local neighborhood structure is more distinct when nonlinear relationships are emphasized. Together, these figures support the use of clustering because they show that the extracted feature space contains meaningful class structure rather than random dispersion.

3.7.2 Performance per Epoch

To highlight the performance of the proposed method over a number of training epochs, Fig. 3.8 is a record of the performance metric Pairwise F-score (F_P) per epoch for each of the five MRI datasets used in this experiment. We observe the performance of the framework improving with increasing number of training epochs, but only to a point. For the three top-performing datasets (Datasets 1-3), the performance of the framework plateaus after roughly ten epochs.

Fig. 3.9 reports B-Cubed F-score (F_B) across the same training epochs and datasets. The trends are consistent with Fig. 3.8: Datasets 1–3 stabilize earlier, while Datasets 4 and 5 require a longer training period before reaching their strongest performance. This agreement across the two metrics is important because Pairwise F-score evaluates pair-level clustering consistency, whereas B-Cubed F-score evaluates sample-level consistency. Similar trends in both figures indicate that the observed performance pattern is not specific to only one metric.

Fig. 3.10 compares clustering performance when the quantum transformer module is included against a classical transformer alternative. The two configurations produce similar results for Datasets 1–3, indicating that both models can form reliable clusters when the extracted feature structure is already well separated. The difference becomes more visi-

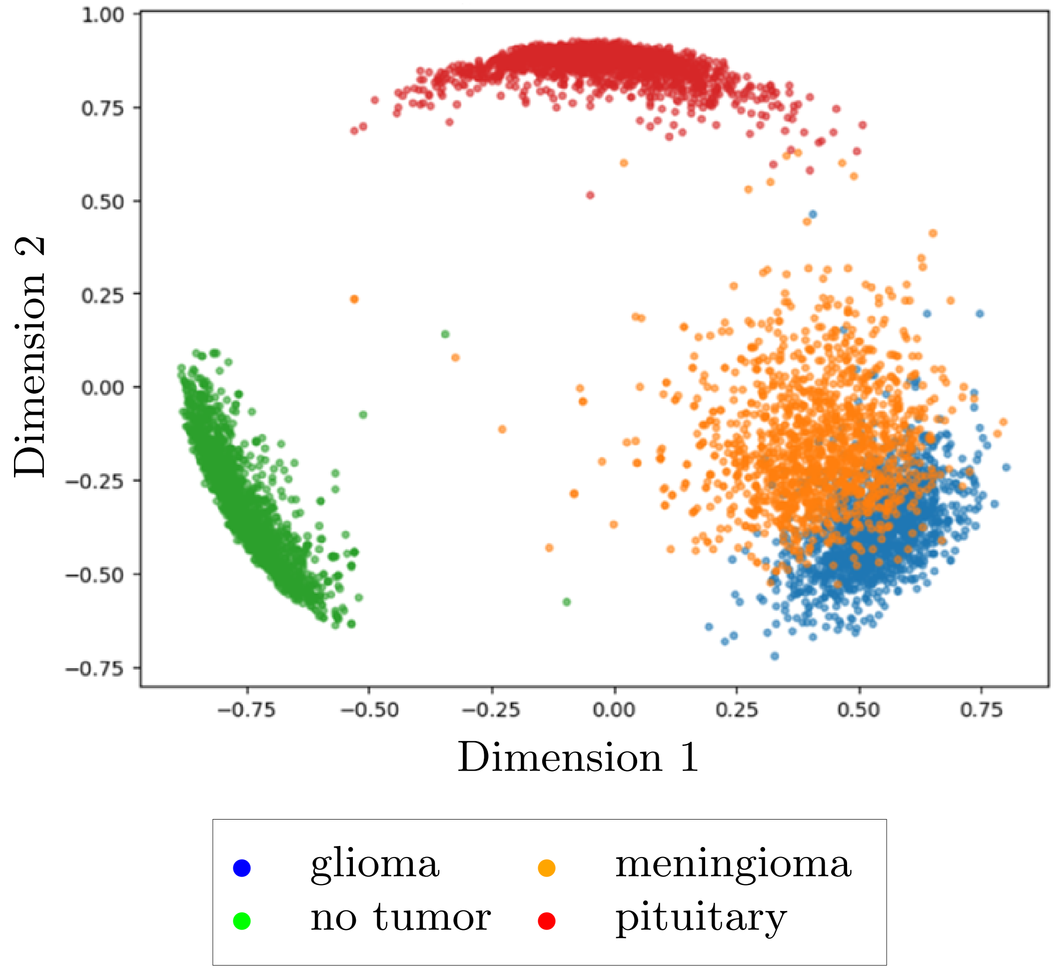


Figure 3.6: *Principal component analysis (PCA) of brain-tumor-dataset (Dataset 3). We observe a clear separation between the no tumor and pituitary classes, whereas the glioma and meningioma classes are intermixed*

ble for Datasets 4 and 5, where the quantum module improves performance by 2.10% and 4.04%, respectively. This result suggests that the quantum module is most beneficial when the dataset presents more difficult clustering conditions, such as weaker class separation or greater feature overlap.

Evaluating the performance of the framework on each of the five datasets used in this experiment, we observe a similar performance level in Pairwise F-score and B-Cubed F-score metrics for some of the datasets, namely Datasets 1, 2, and 3. The most likely cause

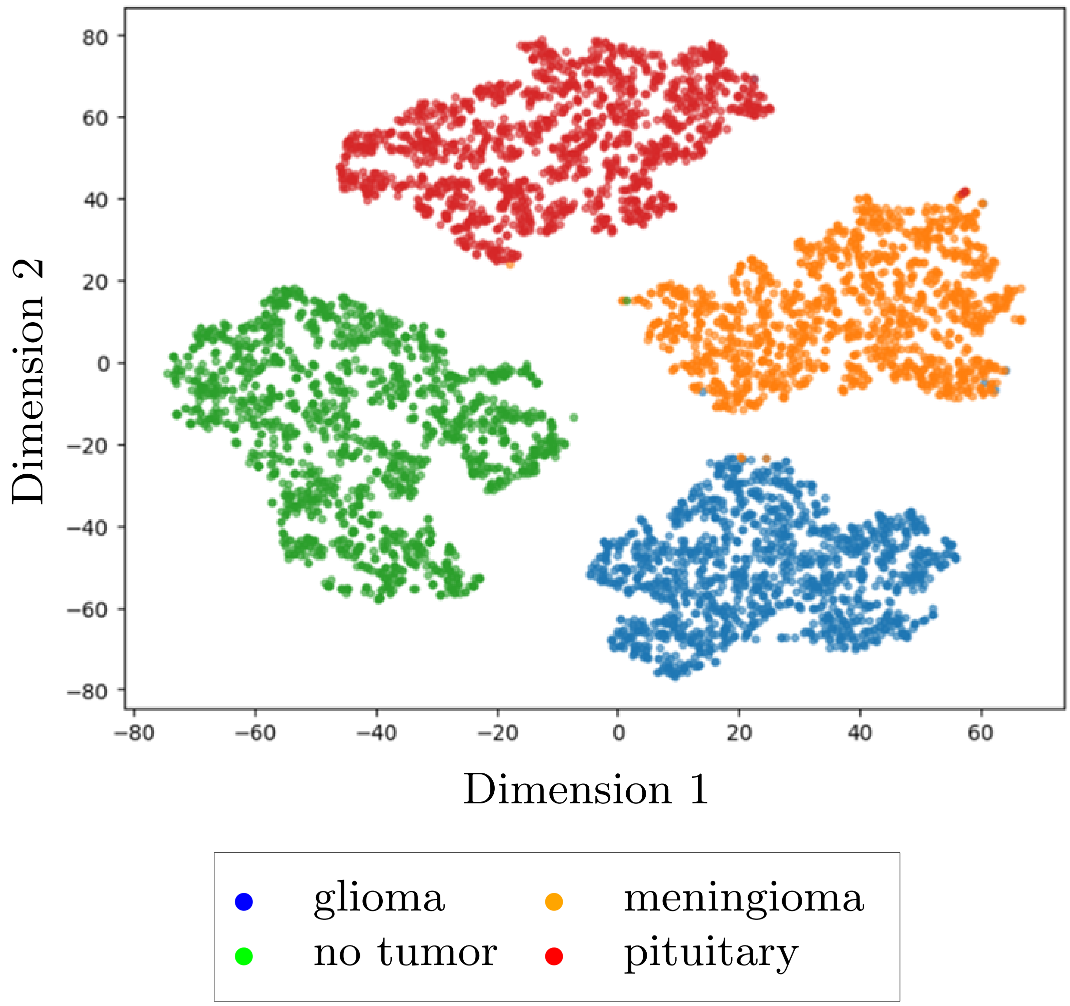


Figure 3.7: *t-distributed stochastic neighbor embedding (t-SNE) of brain-tumor-dataset (Dataset 3). We observe each of the four classes having their own distinct location relative to the other classes with little to no overlap*

of this similar performance level is the relatively low number of classes for this type of application. MRI datasets, especially brain MRI datasets, typically have single-digit class values with variance found in the number of instances per class rather than in the number of class definitions. This similarity between datasets for these characteristics leads to similar performance in clustering tasks across datasets.

These experimental results collectively validate our core hypothesis—a hybrid quantum-classical transformer architecture can offer tangible advantages in clustering image data

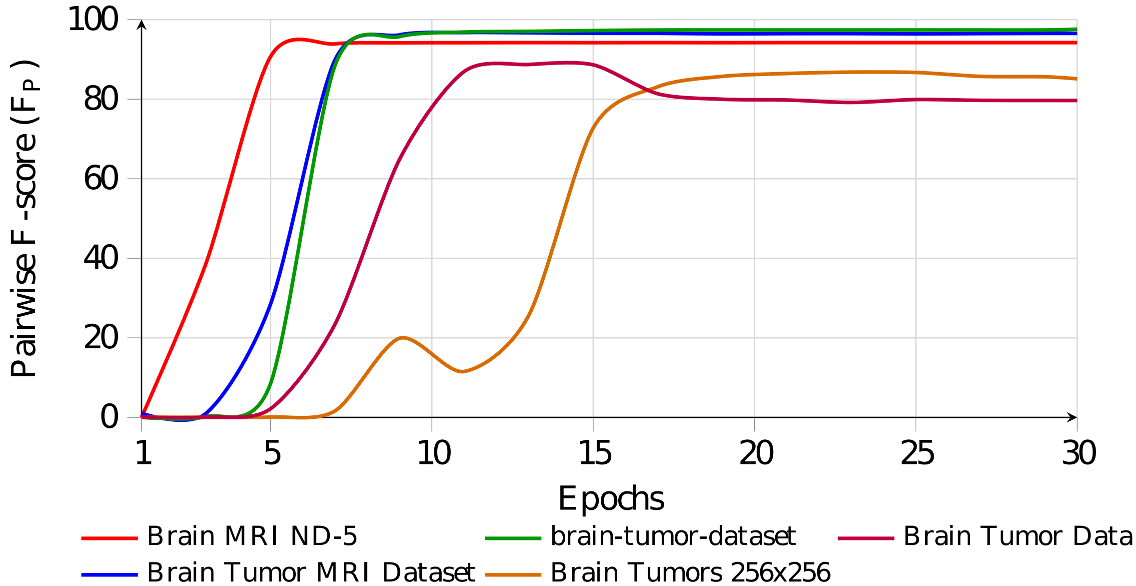


Figure 3.8: *Comparison of Pairwise F-scores (F_P) over training epochs for five MRI datasets. The two lowest-performing datasets, Brain Tumor Data and Brain Tumors 256x256, require more epochs to reach their highest performance levels than the other three datasets.*

performance over purely classical transformer models. Metric evaluations and benchmarks from these experiments confirm the approach's scalability, robustness, and efficiency.

3.8 Summary

This chapter introduced Quantum Transformer for Clustering MRI (QTCM), a hybrid quantum-classical method for unsupervised visual clustering of brain magnetic resonance imaging data. The framework combines classical feature extraction, k-nearest neighbors-based cluster construction, and transformer-based visual processing with quantum state feature encoding and a quantum transformer designed to identify hard or noisy samples within clustered image data. Amplitude encoding is used to map classical feature vectors into quantum states, while cosine similarity encoding preserves sequential and positional structure across clustered inputs for transformer processing. A binary prediction objective with cross-entropy loss is used to train the framework to distinguish samples that are consistent with cluster centroids from those that degrade clustering quality. Experimental studies were

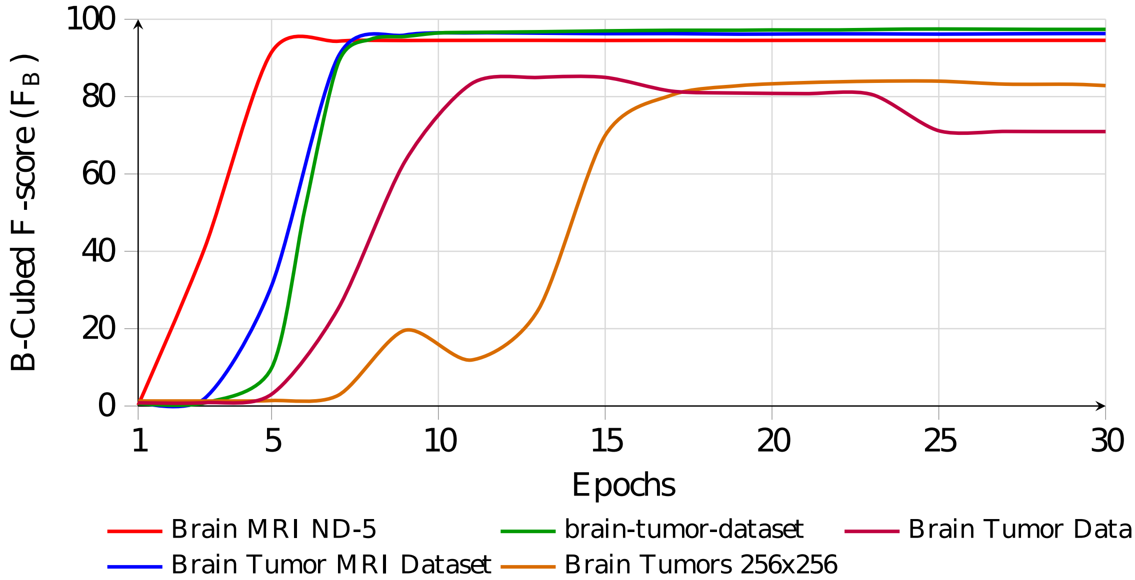


Figure 3.9: *Comparison of B-Cubed F-scores (F_B) over training epochs for five MRI datasets. We observe three datasets plateauing within 10 epochs, whereas Brain Tumor Dataset and Brain Tumors 256x256 take more (more than 20) to plateau*

carried out on five multi-class brain MRI datasets using Pairwise F-score, B-Cubed F-score, and normalized mutual information to evaluate performance. Consistent performance across these datasets, together with improvements over a comparable classical transformer-based approach, demonstrates the benefit of the proposed framework for unsupervised brain MRI clustering.

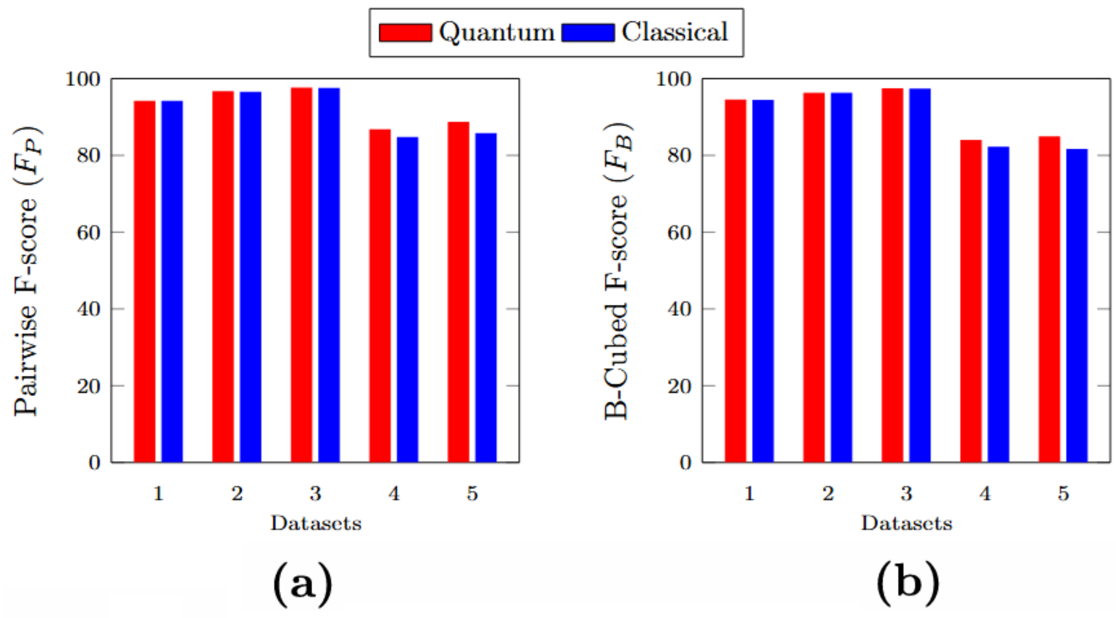


Figure 3.10: *Comparison of clustering performance between quantum and classical algorithms on five brain MRI datasets. Dataset order is consistent across subplots, with Datasets 1, 2, 3, 4, and 5 representing Brain MRI ND-5, Brain Tumor MRI Dataset, brain-tumor-dataset, Brain Tumors 256x256, and Brain Tumor Dataset, respectively. (a) Pairwise F-score (F_P), (b) B-Cubed F-score (F_B)*

Chapter 4

Clustering Using Deterministic Quantum Computation with One Qubit

Portions of this chapter are adapted from the manuscript titled *Impact of Clustering Strategy Selection on Brain MRI Tumor Classification*, submitted to *Quantum Information Processing* for review.

D. Couch, R. Salehi, A. W. Malik, and S. U. Khan, "Impact of Clustering Strategy Selection on Brain MRI Tumor Classification," submitted to *Quantum Information Processing*, (2026). **Under Review**.

4.1 Clustering Strategy Selection

This chapter evaluates the impact of clustering strategy selection on brain MRI tumor classification. MRI datasets may differ in size, class balance, feature distribution, and image quality. Because of these differences, a fixed clustering method, like the k-nearest neighbors system shown in Chapter 3, may perform well on one dataset but poorly on another. This chapter addresses this issue by comparing multiple clustering algorithms under the same feature extraction and evaluation conditions. These algorithms are commonly used in the literature and are detailed in Section 1.3.1.

This chapter introduces *DQC1-projected*, a method inspired by deterministic quantum computation with one qubit (DQC1)[29]. The method is designed to group feature representations using a quantum-based similarity structure. The performance of *DQC1-projected* is compared with several classical clustering algorithms across two public MRI datasets using Pairwise F-score, B-cubed F-score, and normalized mutual information. The experiments presented later in this chapter evaluate whether clustering strategy selection affects classification performance across different MRI dataset conditions.

4.2 DQC1-projected

This section introduces *DQC1-projected*, a clustering method inspired by the Deterministic Quantum Computation with One Qubit (DQC1) model. The method is designed to improve cluster formation by comparing feature vectors through a quantum kernel representation. DQC1 is used to estimate global properties of unitary operators through the normalized trace[29]. These trace estimates are interpreted as entries of a valid kernel matrix[30;31]. Clustering is then performed by minimizing the increase of an entropy criterion, following Rényi-entropy clustering and its DQC1 adaptation[58;59].

4.2.1 Problem Setup

Let $X = \{x_i\}_{i=1}^{N}$ be a set of d-dimensional feature vectors, where $x_i \in \mathbb{R}^d$. These feature vectors are obtained from the MRI images after feature extraction. Dimensionality reduction is then applied so that each feature vector can parameterize an n-qubit feature map, giving $x_i \in \mathbb{R}^n$. The objective is to assign each point to one of K clusters.

Following quantum kernel methods, each feature vector x is encoded into a parameterized unitary operator $U(x)$ acting on an n-qubit register[30;31]. Pairwise similarity is then computed by comparing two unitary encodings. From[9], this comparison is represented using a relative unitary:

$$U_{\text{rel}}(x_i, x_j) \;=\; U(x_i)\,U(x_j)^{\dagger}, \tag{4.1}$$

where $U(x_i)$ and $U(x_j)$ are the unitary encodings of two feature vectors. When $x_i = x_j$, the relative unitary becomes the identity operator. When the two feature vectors differ, the relative unitary captures the transformation between their quantum encodings. This allows the similarity between two feature vectors to be evaluated through the relationship between their corresponding unitary operators.

4.2.2 Trace Estimation as a Kernel

The DQC1 model estimates the normalized trace of a unitary operator using one clean control qubit and an n-qubit register in the maximally mixed state $I/2^n$ [29]. In this setting, the control qubit is prepared, a Hadamard gate is applied, and a controlled-U_{rel} operation is performed. Measuring the control qubit in the X and Y bases provides estimates of the real and imaginary parts of the normalized trace [30;31]:

$$\langle X \rangle = Re\left(\frac{\mathrm{Tr}(U_{\text{rel}})}{2^n}\right), \tag{4.2}$$

$$\langle Y \rangle = Im\left(\frac{\mathrm{Tr}(U_{\text{rel}})}{2^n}\right). \tag{4.3}$$

These two measurements describe how the relative unitary contributes to the normalized trace. In this work, the real component is used to define the DQC1 kernel entry between two feature vectors:

$$K_{ij} \;=\; k(x_i, x_j) = Re\left(\frac{\mathrm{Tr}\big(U_{\text{rel}}(x_i, x_j)\big)}{2^n}\right), \tag{4.4}$$

which corresponds to the normalized-trace kernel used for DQC1 clustering [30]. The value K_{ij} represents the similarity between feature vectors x_i and x_j after both are mapped into unitary form. In this way, the kernel matrix provides the similarity structure used for clustering.

For clustering, it is useful for the similarity values to be nonnegative and normalized. Following the DQC1 formulation in [59], the kernel matrix is transformed using the affine map

$$M = \frac{1}{2}(K + I), \tag{4.5}$$

so that $M_{ii} = 1$. The resulting matrix M is interpreted as a nonnegative similarity matrix. This matrix is then used to evaluate cluster quality through a Rényi entropy criterion.

4.2.3 Rényi Entropy

Rényi-entropy clustering assigns samples to clusters by minimizing the increase in within-cluster entropy when a new point is added [58]. This criterion favors compact clusters whose members are highly similar to one another. The proposed method uses Rényi order-2 entropy, which can be expressed in terms of the purity of a density operator [59].

Given a cluster $C \subset \{1, \ldots, N\}$, let M_C denote the submatrix of M restricted to the samples in cluster C. This submatrix contains the pairwise similarities among the members of the cluster. To evaluate the entropy of this cluster, M_C is normalized to unit trace:

$$\rho_C = \frac{M_C}{\mathrm{Tr}(M_C)}. \tag{4.6}$$

The normalized matrix ρ_C acts as the cluster-level density representation. Its Rényi-2 entropy is then defined as:

$$H_2(C) = -\log(\mathrm{Tr}(\rho_C^2)). \tag{4.7}$$

A lower value of $H_2(C)$ corresponds to a more compact cluster because the cluster members have stronger internal similarity. The term $\mathrm{Tr}(\rho_C^2)$ can be estimated using one-clean-qubit trace estimation on a block-encoding of ρ_C^2. In this implementation, $\mathrm{Tr}(\rho_C^2)$ is computed directly from M_C after the DQC1 kernel matrix has been calculated:

$$\mathrm{Tr}(\rho_C^2) \;=\; \frac{\mathrm{Tr}(M_C^2)}{\mathrm{Tr}(M_C)^2} \;=\; \frac{\|M_C\|_F^2}{\mathrm{Tr}(M_C)^2}, \tag{4.8}$$

where $\|M_C\|_F^2$ is the Frobenius norm. This formulation preserves the Rényi-2 objective used in prior entropy clustering methods[58] while avoiding additional computation beyond the kernel matrix evaluation.

Following Rényi-entropy clustering[58], each point i is assigned to the cluster that produces the smallest increase in within-cluster entropy. For cluster C_k, this increase is defined as

$$\Delta H_k(i) = H_2\big(C_k \cup \{i\}\big) - H_2(C_k), \tag{4.9}$$

and the final assignment is given by

$$\ell_i \;=\; \arg \min_{k\in\{1,\ldots,K\}} \Delta H_k(i). \tag{4.10}$$

This assignment rule places each point into the cluster that best preserves compactness in the kernel space. As a result, *DQC1-projected* uses the normalized-trace kernel to define pairwise similarity and the Rényi-2 entropy criterion to guide cluster membership.

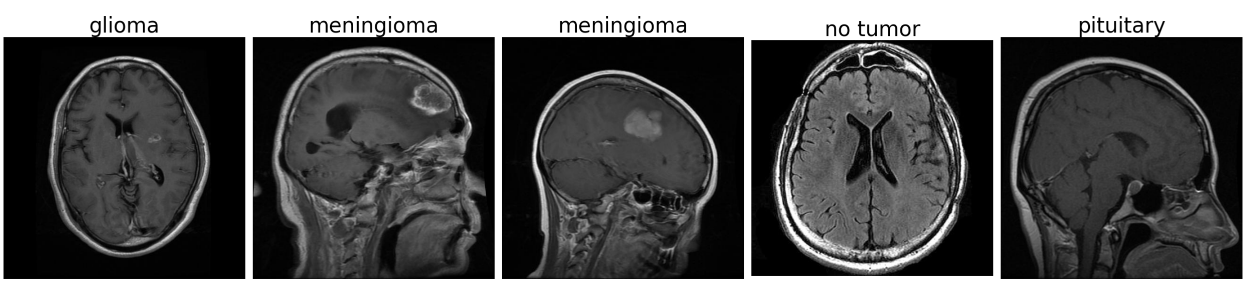


Figure 4.1: *Example axial and sagittal brain MRI images from the Brain Tumor Dataset with corresponding class labels.*

4.3 Datasets

A broader description of the MRI datasets used in this dissertation is provided in Section 3.5. This chapter uses two datasets for evaluation: the *Brain Tumor Dataset*[18] and the *Brain Tu-*

mor Classification (MRI) dataset[19], also referred to as the *Sartaj Dataset*[20]. These datasets are used because they share the same four diagnostic categories while differing in size, class balance, and train-test composition. This allows the proposed *DQC1-projected* method to be evaluated under two related but distinct data conditions.

Fig. 4.1 shows example MRI images with corresponding class labels from the *Brain Tumor Dataset*. These examples provide visual context for the classification task by showing the four diagnostic categories used in the experiments: glioma, meningioma, pituitary, and no tumor. The visual differences among these categories are not uniform. Some classes show clearer structural differences, while others share similar visual characteristics. This overlap is important for the experiments in this chapter because the proposed method is intended to improve separation among visually similar tumor classes.

The *Brain Tumor Dataset* is retained from the Chapter 3 evaluation because it provides a large four-class MRI benchmark with established training and testing divisions. This dataset has a training-to-testing ratio of approximately 81.4:18.6, close to the 80:20 ratio commonly found in medical imaging studies[79;80]. The class distribution is also relatively balanced across the four diagnostic categories, making this dataset useful for comparing the proposed method against classical clustering alternatives under stable evaluation conditions.

The *Sartaj Dataset* provides a complementary evaluation setting. It contains 2,870 training images and 394 testing images across the same four diagnostic categories. The training group contains 826 glioma images, 822 meningioma images, 827 pituitary images, and 395 no tumor images. The testing group includes 100 glioma images, 115 meningioma images, 74 pituitary images, and 105 no tumor images. Compared with the *Brain Tumor Dataset*, this dataset is smaller and less balanced, particularly in the no tumor category. This difference is useful for evaluating whether the proposed method remains effective when class representation differs across datasets.

Figs. 4.4 and 4.5 show the class distributions for the *Brain Tumor Dataset* and *Sartaj Dataset*, respectively. The *Brain Tumor Dataset* shows a relatively even distribution across the four classes, while the *Sartaj Dataset* shows a more uneven distribution. These differences provide context for interpreting the later performance results because class imbal-

Table 4.1: *Deep learning feature extraction summary*

Layer	**Dimension**	**Hyperparameter**
Input Image	$224 \times 224 \times 3$	image_type = RGB
Conv 1	$112 \times 112 \times 64$	kernel = 7×7, activation = `relu`
Max Pooling	$56 \times 56 \times 64$	pool_size = 3×3, stride = 2
Conv 2	$56 \times 56 \times 256$	blocks = 3, filters = [64, 256]
Conv 3	$28 \times 28 \times 512$	blocks = 4, filters = [128, 512]
Conv 4	$14 \times 14 \times 1024$	blocks = 6, filters = [256, 1024]
Conv 5	$7 \times 7 \times 2048$	blocks = 3, filters = [512, 2048]
Average Pooling	$1 \times 1 \times 2048$	pooling = global_average
Feature Vector	2048	weights = `imagenet`, top = False

ance can influence clustering behavior, classification outcomes, and the stability of algorithm comparisons.

The authors of both datasets include information about MRI data collection, image specifications, and file properties. The images are stored in Joint Photographic Experts Group (JPEG) format rather than Digital Imaging and Communications in Medicine (DICOM) format. Although the images are visually grayscale, some are stored with three-channel encodings. This variation in image encoding can occur in real medical imaging datasets and must be handled consistently during preprocessing. To maintain a uniform input representation, all images are treated as three-channel inputs and resized to 224×224 pixels to match the expected ResNet-50 input dimension.

4.3.1 Preprocessing System Configuration

Selecting ResNet-50 to perform deep feature extraction in this work over other models was based on its performance in brain tumor classification and similar tasks[4;81;82]. These works implemented ResNet-50 as part of a greater system and achieved high accuracy and F1-scores using these methods.

Our ResNet-50 method was initialized with ImageNet pre-trained weights and the classification top layer removed (*top* = *False*). Input MRI images were pre-processed through the residual convolutional stages of ResNet-50, consisting of five convolutional blocks. The

Table 4.2: *Performance comparison of our method to state-of-the-art methods on the Brain Tumor Dataset.*

Study	Year	Method	Precision	Recall	F-score
Mohan *et al.*[21]	2024	MobileNet	91%	97%	93%
Mohan *et al.*[33]	2024	Inception V3	100%	97%	98%
Gagliardi *et al.*[83]	2025	k-NN	98%	97%	97%
Ali *et al.*[84]	2025	Fused-feature vector (FFV)	98.23%	96.52%	97.34%
Vickram *et al.*[85]	2025	FFV	98.97%	96%	97.46%
Zamil *et al.*[86]	2025	CNN	94%	94%	94%
Ours	–	**DQC1-projected**	**99.0%**	**98.3%**	**98.7%**

network transforms the input into high-level feature maps, culminating in a tensor of size 7×7×2048 at the output of the final convolutional block. A global average pooling operation is then applied, reducing the spatial dimension to produce a single fixed-length 2048-dimensional feature vector. Table 4.1 provides a layer level detail of the ResNet-50 configuration used to perform deep feature extraction of the two datasets used in this work.

Table 4.3: *Performance comparison of state-of-the-art methods on the Sartaj Dataset.*

Study	Year	Method	Precision	Recall	F-score
Alanazi *et al.*[87]	2022	CNN	95.15%	96.32%	95.71%
Başaran[88]	2022	GLCM	95.27%	99.43%	98.09%
Chitnis *et al.*[89]	2022	LeaSE + DARTS	91.49%	91.5%	91.48%
Ravinder *et al.*[90]	2023	Net-2	90.0%	97.0%	93.37%
Akter *et al.*[91]	2024	CNN and U-Net	78.9%	78.9%	78.9%
Hossain *et al.*[6]	2024	IVX16	80.0%	76.0%	75.0%
Ours	–	**DQC1-projected**	**98.8%**	**98.2%**	**98.5%**

4.4 Experimental Results

This section details the performance of our proposed method relative to other recent methods. The comparison used two experimental scenarios and three performance metrics. The

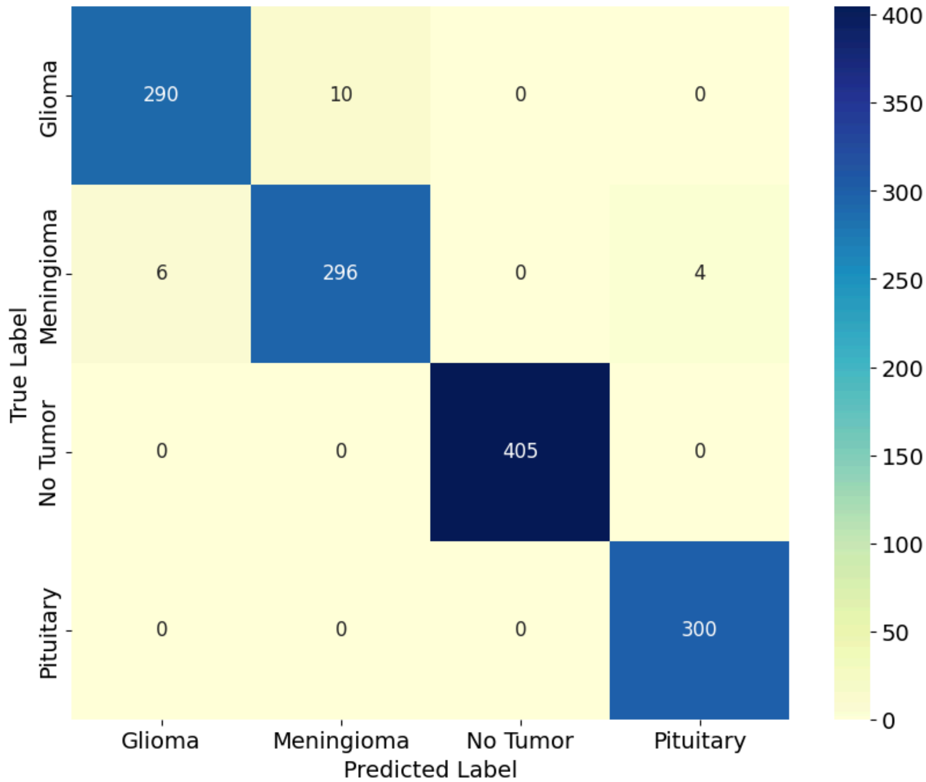


Figure 4.2: *Brain Tumor Dataset confusion matrix. The majority of classification errors occur in the upper-left region, corresponding primarily to confusion between the glioma and meningioma classes.*

proposed solution demonstrated state-of-the-art results in both dataset environments. Our experimental approach examined two challenging, multi-class classification datasets that contain real patient clinical data.

4.4.1 Experimental Settings

This section contains the experimental settings and evaluation metrics used in this work. We discuss the datasets, experimental settings, and performance metrics in the respective sections. We conducted our series of experiments using the ChameleonCloud service[92]. The experimental setup comprised two Intel(R) Xeon(R) Gold 6126 CPUs at speeds of 2.60GHz, one NVIDIA Quadro RTX 6000 GPU with 24 GB VRAM and 4,608 CUDA Cores. This system utilized a Linux operating system, Python 3.11 and associated machine learning

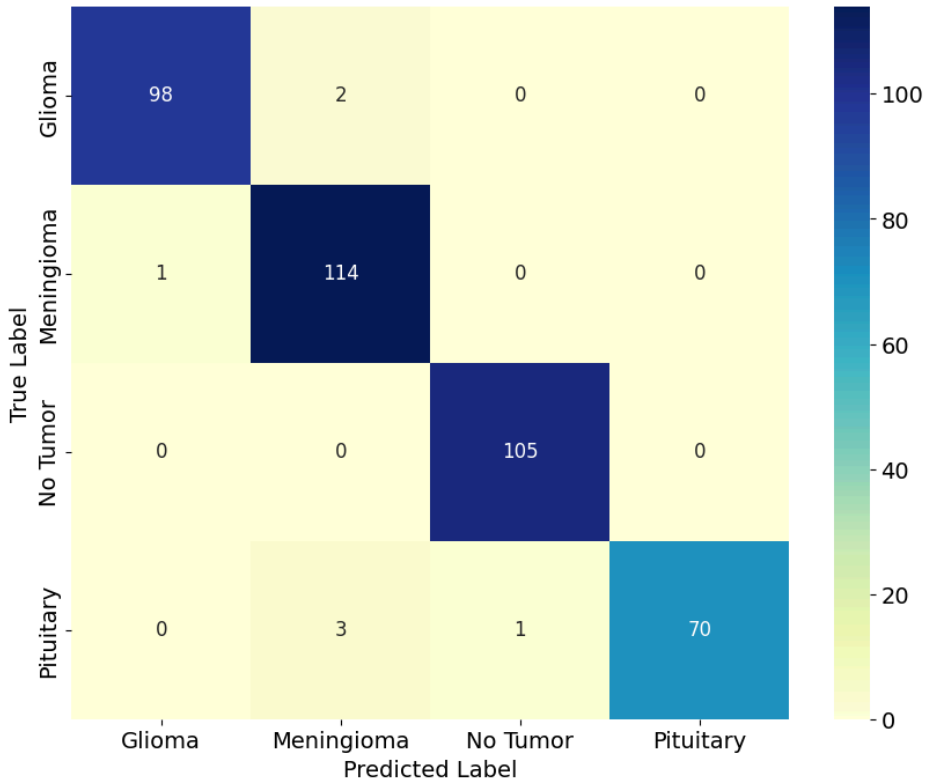


Figure 4.3: *Sartaj Dataset confusion matrix. Although there are few prediction errors shown in this figure, the most difficult class to identify is Meningioma*

libraries including scikit-learn, Qiskit, PyTorch, TorchQuantum, among others.

4.4.2 Performance on the Brain Tumor Dataset

To establish direct comparisons with recent works, we first analyzed performance on the *Brain Tumor Dataset*, which contains four diagnostic categories: glioma, meningioma, pituitary, and no tumor. The proposed *DQC1-projected* model achieves **99% precision**, **98.3% recall**, and a **98.7%** $\boldsymbol{F}$**-score**, shown in Table 4.2. This result exceeds state-of-the-art methods on the same dataset[21;33;83–86]. In particular, prior systems achieve strong performance using CNNs (Inception V3 with 98% F-score), as well as FFVs (with 97.34% and 97.46% F-scores). Our results demonstrate that our proposed method is comparable to similar methods in the literature, supporting the central motivation of improving performance in settings where label availability may be constrained.

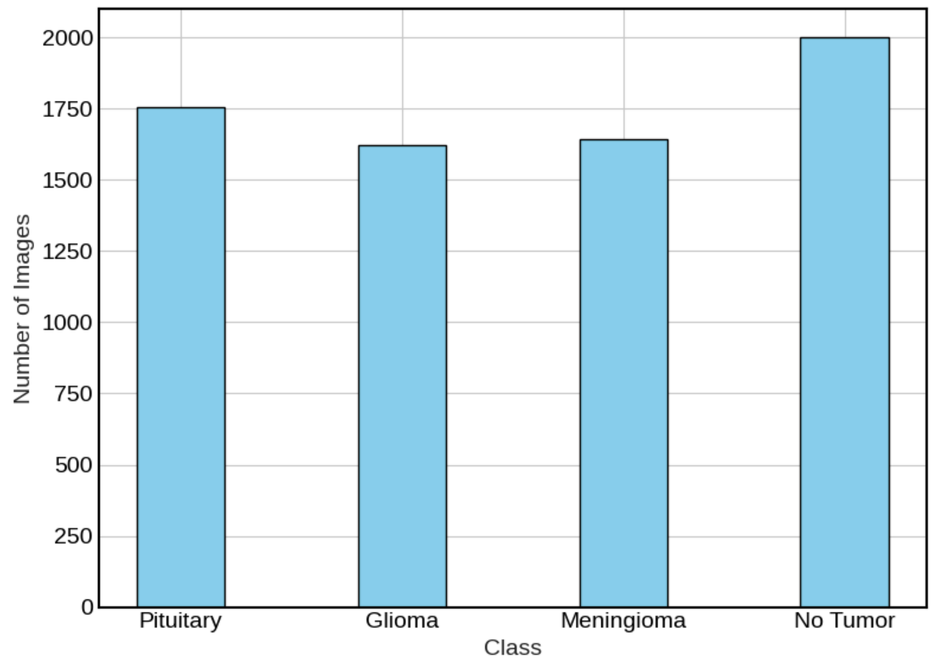


Figure 4.4: *Brain Tumor Dataset class distribution. There is a relatively even distribution, with No Tumor being the most populous and Glioma the least populous*

The confusion matrices in Figs. 4.2 and 4.3 highlight which class definitions were most difficult to discern in either dataset. For the *Brain Tumor Dataset*, there were only a small number of errors in total. Specifically, the dominant number of misclassifications occur between the *glioma* and *meningioma* class labels, with 10 glioma instances incorrectly predicted as meningioma and 6 meningioma instances predicted as glioma. This pattern is a common source of error in four-class brain tumor datasets because glioma and meningioma images can appear more visually similar to each other than to the other classes. A second error dynamic is observed for meningioma, where 4 samples are assigned to the pituitary class. In contrast, the *no tumor* and *pituitary* categories are classified without any error for this dataset, indicating that the ResNet-50 embeddings combined with DQC1-projected clustering yield strong class definitions. This pattern is consistent with the high precision, recall, and F-score values reported in Table 4.2.

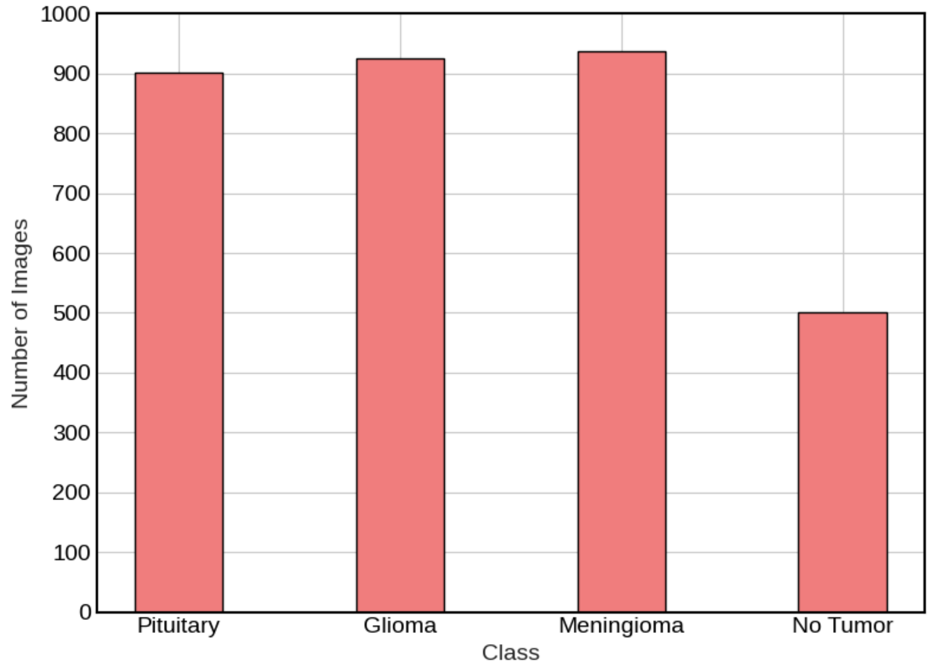


Figure 4.5: *Sartaj Dataset class distribution. While Pituitary, Glioma, and Meningioma classes have a relatively even number of images, No Tumor has roughly half that number which can affect system performance*

4.4.3 Algorithm Analysis on the Brain Tumor Dataset

Building on the dataset-level performance, we next examine how clustering algorithm choice affects performance. Figs. 4.6, 4.7, and 4.8 compare *DQC1-projected* with several classical clustering baselines. These baselines include k-means, fuzzy c-means, k-NN, Shared Nearest Neighbor (SNN), DBSCAN, HDBSCAN, and spectral clustering. Across pairwise consistency, per-sample consistency, and information agreement, *DQC1-projected* produces the strongest result on the *Brain Tumor Dataset*. The next highest score is produced by k-NN.

Two observations are notable. First, the classical methods form a distinctly lower performance band across the three metrics. The graph-based approaches, k-NN and SNN, remain competitive relative to the density- and centroid-based methods. However, they still underperform *DQC1-projected*, suggesting that the quantum-based projection reduces class mismatch. Second, the density-based methods show the largest degradation, which is consistent with the sensitivity of DBSCAN and HDBSCAN to data density and scale.

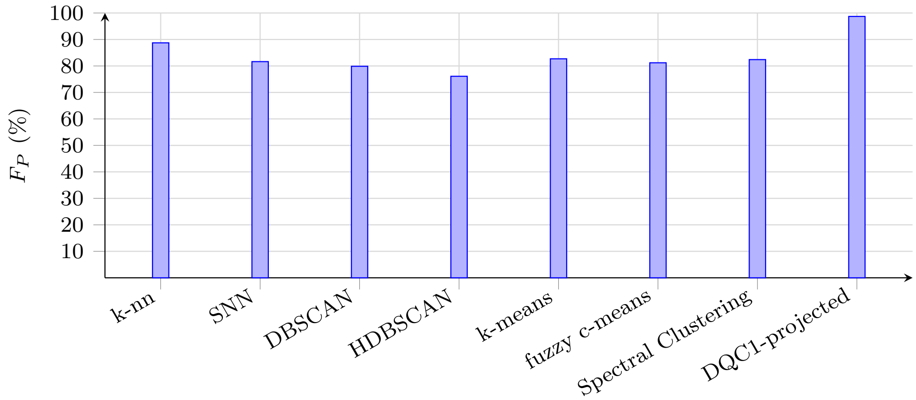


Figure 4.6: *Pairwise F-score (F_P) comparison for different algorithms on the Brain Tumor Dataset. DQC1-projected is the highest performing algorithm while HDBSCAN is the lowest performing algorithm*

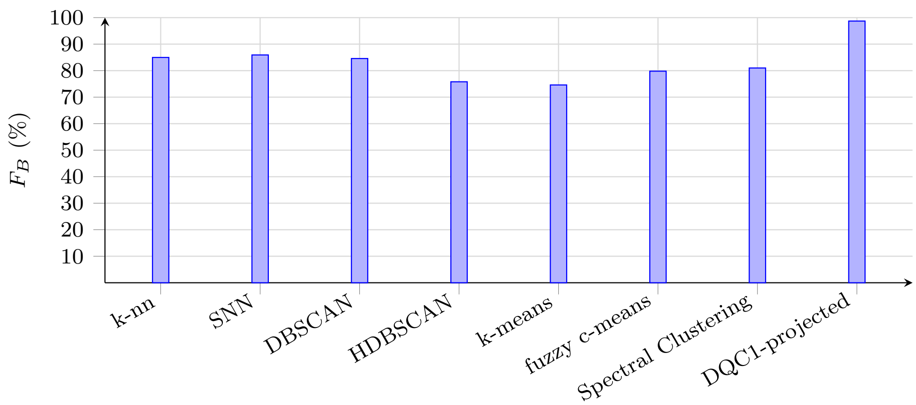


Figure 4.7: *B-cubed F-score (F_B) comparison for different algorithms on the Brain Tumor Dataset. DQC1-projected is the highest performing while k-means is the lowest performing algorithm*

The consistent separation across the three metrics indicates that *DQC1-projected* improves clustering quality in a manner that is not metric-specific.

4.4.4 Performance on the Sartaj Dataset

We next evaluate performance using the *Sartaj Dataset*, which differs in size and class balance and therefore provides a complementary setting for testing the method with different input data. Despite the shift in class representation ratio in the dataset, our method achieves high

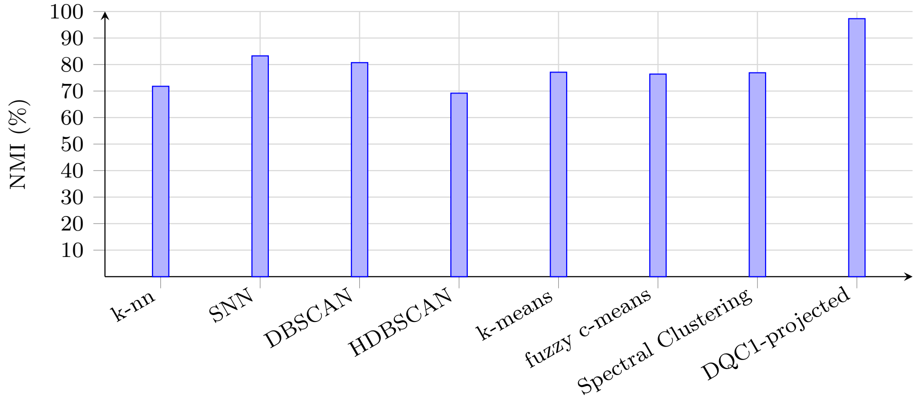


Figure 4.8: *Normalized Mutual Information (NMI) comparison for different algorithms on the Brain Tumor Dataset. DQC1-projected has the highest cluster quality, while HDBSCAN had the lowest cluster quality*

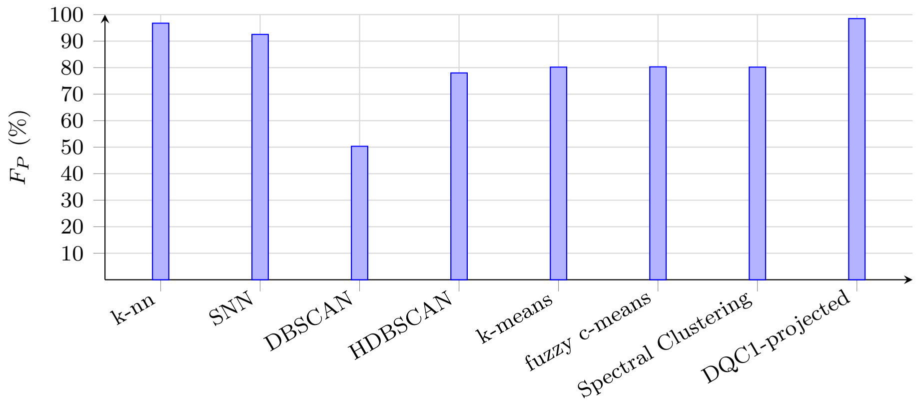


Figure 4.9: *Pairwise F-score (F_P) comparison for different algorithms on the Sartaj Dataset. DQC1-projected was the highest performing method while HDBSCAN was the lowest performing method*

performance, reporting **98.8% precision**, **98.2% recall**, and **98.5% F-score** performing well against the state-of-the-art methods on this dataset[6;87–91], shown in Table 4.3. This result matches the performance for the previous dataset, and highlights a robust solution.

The Sartaj confusion matrix shown in Fig. 4.3 is a strong result with only a few total errors. A small confusion exists between glioma and meningioma, where two gliomas were predicted as meningioma while one meningioma was wrongly predicted to be a glioma. Additionally, four pituitary cases are incorrectly assigned to meningioma (three instances) and no tumor (one instance). The class label with the fewest errors was the *no tumor*

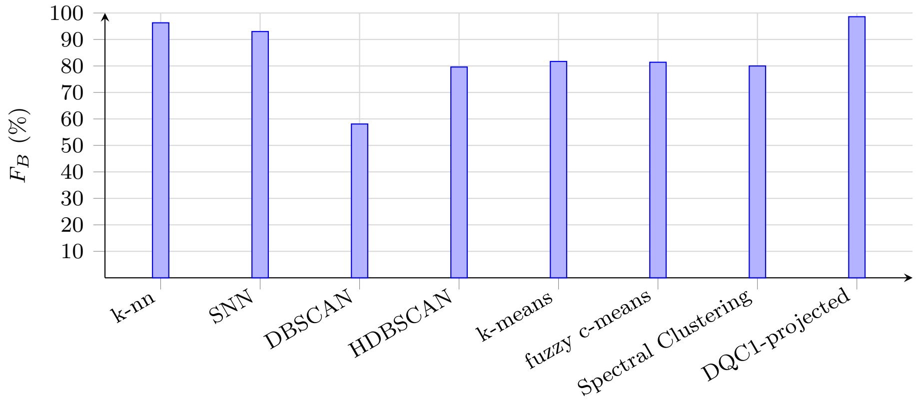


Figure 4.10: *B-cubed F-score (F_B) comparison for different algorithms on the Sartaj Dataset. DQC1-projected and k-NN were the two highest scoring algorithms while DBSCAN was the lowest*

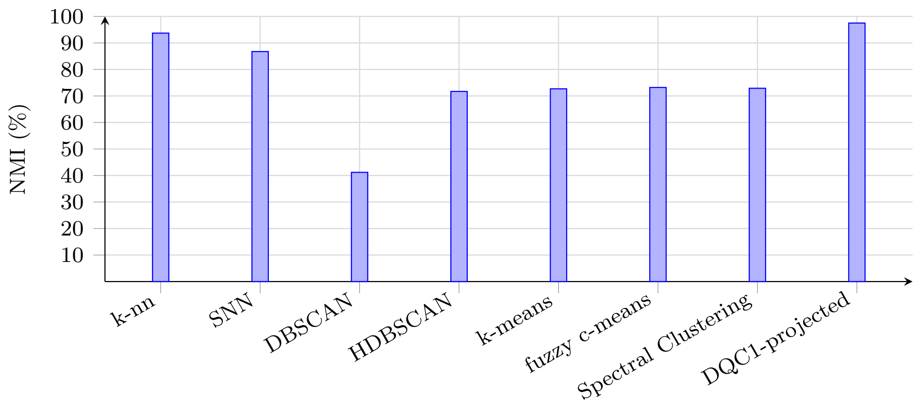


Figure 4.11: *Normalized Mutual Information (NMI) comparison for different algorithms on the Sartaj Dataset. DQC1-projected and k-NN had the highest cluster quality score, while DBSCAN had the lowest cluster quality*

category, which was classified without error in this testing split. These misclassifications align with clinically known ambiguities. Pituitary tumors may present with a wide range of size and intensity, while meningioma and glioma share overlapping texture and boundaries. From a system perspective, the limited number of these mismatch errors and high performance metric values points to a quality solution correctly identifying important image features.

4.4.5 Algorithm Choice Analysis on the Sartaj Dataset

To evaluate whether the observed performance generalized beyond one dataset, we repeated the algorithm comparison experiment on the *Sartaj Dataset*, as shown in Figs. 4.9, 4.10, and 4.11. The performance trends mirror the relationships found in the *Brain Tumor Data* figures. *DQC1-projected* achieves the highest F_P, F_B, and NMI, with the next highest performing method, k-NN only slightly behind. This multiple dataset consistency supports the robustness of the proposed solution.

Comparatively, the more pronounced spread among classical algorithms on Sartaj is consistent with its class imbalance and reduced sample count relative to the *Brain Tumor Dataset*. Under these conditions, centroid-based methods (k-means, fuzzy c-means) may be negatively biased, and density-based methods (DBSCAN, HDBSCAN) may be less stable. The performance of DQC1-projected across these datasets highlights its viability on brain MRI classification tasks.

4.5 Summary

This chapter presented an empirical comparative study of clustering strategy selection for brain MRI tumor classification. The main contribution is the introduction and evaluation of *DQC1-projected*, a DQC1-inspired method, against seven classical clustering alternatives across two public MRI datasets. The study emphasized benchmarking, clustering behavior, robustness, class separability, and consistency across heterogeneous MRI feature spaces.

Across both datasets, *DQC1-projected* achieved the strongest reported F-score among the methods included in the comparison, obtaining an F-score of 98.7% on the *Brain Tumor Dataset* and 98.5% on the *Sartaj Dataset*, with balanced precision and recall. The confusion matrices show that classification errors are limited and concentrated in the most challenging class pairs, while *DQC1-projected* consistently outperformed classical clustering alternatives in Pairwise F-score, B-cubed F-score, and normalized mutual information. These results indicate that *DQC1-projected* improves clustering-based classification performance in a manner that is stable across datasets and evaluation metrics.

Chapter 5

Quantum Hardware Execution

5.1 Introduction

The results presented in Chapters 3 and 4 establish QTCM as a hybrid quantum-classical framework for unsupervised brain MRI classification, and show that DQC1-projected achieves state-of-the-art performance on two widely used brain MRI datasets. Those studies, however, were carried out in simulation and controlled experimental environments. A critical remaining question is whether the same system can be executed on physical quantum processors, where noise, limited runtime, and resource constraints are unavoidable[12;13]. In near-term quantum computing, the practical value of a quantum machine learning method depends not only on its algorithmic design, but also on its ability to operate under these hardware limitations[8;10;93].

This chapter addresses that question by deploying QTCM on an IBM quantum computer. In contrast to the simulation-based studies of the previous chapters, the focus here is not to introduce a new clustering method or to perform broad hyperparameter exploration, but rather to evaluate the feasibility of the proposed framework on a present-day superconducting quantum backend. The deployment uses the same two datasets explored in Chapter 4, the *Brain Tumor Dataset*[18] and the *Sartaj Dataset*[19;20], and preserves the same classical preprocessing pipeline, feature extraction method, and DQC1-based quantum method. In this way, the study isolates the effect of moving the quantum portion of the system from simulation to device-based execution.

Executing QTCM on a physical quantum device required practical adjustments to account for runtime and resource constraints. The central adjustment was to process each dataset in smaller batches, allowing the quantum stage to execute within the available IBM Quantum runtime budget while preserving the structure of the overall model. The detailed batching procedure is described in Section 5.2.3. The hardware used for these experiments was IBM Quantum Heron, a 156-qubit device with fixed-frequency qubits and tunable couplers. The device specifications used in this work include an error per layered gate (EPLG) rate of 2.20×10^{-3} and circuit layer operations per second (CLOPS) of 340,000.

The contribution of this chapter is therefore distinct from those of the previous chapters. Chapter 3 introduced the original QTCM framework for unsupervised visual clustering, while Chapter 4 introduced algorithmic comparisons for the proposed DQC1-projected method. Here, we demonstrate that QTCM can be carried beyond simulation and executed under the limitations of a physical quantum backend. The results presented in this chapter provide an implementation study of QTCM and discuss the practical considerations required to run hybrid quantum machine learning systems for brain tumor classification on available quantum devices.

5.2 Hardware Deployment Methodology

The experimental hardware deployment preserves the hybrid structure of QTCM described in the previous chapters. Classical image preprocessing, ResNet-50 feature extraction, and dataset organization remain classical tasks, while the parameterized quantum circuit stage is executed on the IBM backend. This is consistent with the design principles of hybrid quantum-classical learning systems, where classical resources are used for computationally mature stages and quantum resources are reserved for the circuit-based stage most directly associated with the learned representation[8;11;94]. The deployment therefore does not require altering the structure of the QTCM framework; instead, it alters the environment in which the quantum portion of the model is executed.

5.2.1 Execution Workflow

The hardware execution process consists of four stages. First, the system preprocesses the input MRI images and converts them to feature vectors using the same ResNet-50 feature extractor[3] used in Chapter 4. Second, it groups the extracted features into hardware-feasible batches. Third, it encodes the batched features into the quantum circuit required by QTCM, including the DQC1-inspired stage introduced in Chapter 4. Finally, the system collects the hardware outputs and returns them to the classical processing stages for post-processing and analysis. This process preserves the organization of QTCM while adapting the quantum stage to the operational constraints of the backend.

The motivation for this partitioned design is straightforward. Real quantum hardware remains constrained by noise, variation, and limited runtime availability[12;13]. For this reason, the goal of the present deployment is not to move every component of the framework onto the quantum device, but to execute the quantum portion of QTCM under realistic conditions while keeping the remainder of the pipeline consistent with the previous sections. This makes the experiment an implementation study of the quantum stage rather than a redesign of the framework itself.

5.2.2 IBM Quantum Heron Backend

The experiments in this chapter were carried out on IBM Quantum Heron. This popular backend provides 156 qubits and uses an architecture based on fixed-frequency qubits with tunable couplers. In addition to the total qubit count, two device specifications are especially relevant for this work: an EPLG of 2.20×10^{-3} and a CLOPS value of 340,000. These characteristics make Heron an appropriate target for evaluating whether QTCM can be executed outside simulation while still retaining DQC1-based circuit structure.

Although the available qubit count is high relative to earlier generations of hardware, the full deployment problem is still constrained by factors beyond raw qubit number. These

Table 5.1: *IBM Quantum Heron deployment configuration*

Parameter	**Value**
Quantum backend	IBM Quantum Heron
Total qubits	156
Qubit architecture	Fixed-frequency qubits with tunable couplers
EPLG	2.20×10^{-3}
CLOPS	340,000
Available computation window	10 minutes
Execution mode	Batched QTCM deployment

include the size of the batched dataset, the number of circuit evaluations required, and the limited execution time available for each deployment window. As a result, the principal challenge in this chapter is mapping the circuit within the operational limitations of the quantum hardware. In this sense, the IBM Heron experiment evaluates the practicality of executing QTCM on a noisy intermediate-scale quantum (NISQ) system for a medical imaging task.

5.2.3 Batch Construction for Hardware Execution

Batching is the main mechanism used to make the dataset-level experiments compatible with the available quantum runtime. Let $\mathcal{D}$ denote a dataset containing N MRI images. For hardware execution, we partitioned the dataset into B disjoint subsets,

$$\mathcal{D} = \bigcup_{b=1}^{B} \mathcal{D}^{(b)}, \tag{5.1}$$

subject to the constraint

$$|\mathcal{D}^{(b)}| \leq 50, \qquad b = 1, 2, \ldots, B. \tag{5.2}$$

The number of required batches is therefore

$$B = \left\lceil \frac{N}{50} \right\rceil. \tag{5.3}$$

This partitioning was necessary because the available IBM budget of ten computational minutes did not permit a single end-to-end execution over either dataset in this section.

Using the dataset totals in Chapter 4, this strategy produces 141 batches for the *Brain Tumor Dataset* and 66 batches for the *Sartaj Dataset*. These values are shown in Table 5.2. Batching changes the scale of each execution unit, but not the structure of QTCM. Each batch is processed using the same preprocessing, feature encoding, and quantum execution stages; these stages are applied repeatedly across subsets of the data rather than once across the complete dataset.

Table 5.2: *Dataset batching summary for hardware deployment*

Dataset	Images	Batch size	Batches
Brain Tumor Dataset	7,023	50	141
Sartaj Dataset	3,264	50	66

5.3 Datasets and Preprocessing

This hardware study uses the same two datasets evaluated in Chapter 4 so that the results remain directly comparable to the earlier simulation-based evaluation. The first dataset, *Brain Tumor Dataset*[18], is a four-class MRI dataset containing glioma, meningioma, pituitary, and no tumor classes, and was selected in Chapter 4 as one of the principal benchmark environments for evaluation. The second dataset, *Brain Tumor Classification (MRI)*[19], also called the *Sartaj Dataset*[20], provides a second four-class environment with a different dataset size and class balance. Reusing these two datasets in this section ensures that the real-hardware experiments have a point of reference with the evaluation methodology already established for the previous experiments.

Preprocessing was kept consistent with Chapter 4. Each MRI image was resized to the input dimensions expected by ResNet-50, and deep feature extraction was performed using the same ImageNet-initialized ResNet-50 feature extractor with the classification head removed[3]. This choice preserved consistency between the simulation study and the device-based study. The objective of this chapter is not to reevaluate the classical feature extractor, but to assess whether the quantum stage of the framework can be executed on real hardware when preprocessing remains unchanged.

Maintaining an identical classical front end serves two purposes. First, it makes the real-hardware study directly comparable to the simulation-based framework developed in Chapters 3 and 4. Second, it ensures that any implementation-level differences observed during hardware deployment arise from the execution environment of the quantum stage rather than from changes to the feature extraction method or dataset preprocessing. This is especially important in a hybrid system such as QTCM, where both the classical and quantum stages contribute to overall system outcome.

5.4 Experimental Settings

The experiments in this section were designed as a feasibility study for QTCM under practical quantum device constraints. The same DQC1-based method developed in Chapter 4 was used, but execution on the IBM backend required the batching strategy defined in Section 5.2.3. As a result, this study places greater emphasis on successful execution than on metric performance.

Within each experiment, the system completes classical preprocessing and feature organization before quantum execution, then submits the batched feature sets to the IBM Quantum platform. This approach ensured that the quantum backend was used only for the circuit-based stage that constitutes the quantum portion of QTCM. This is consistent with the literature, where classical resources perform data-intensive operations while circuit-based quantum components are used for the learned quantum transformation or kernel evaluation stage[8;9;11].

The batch size of 50 images represents a compromise between per-job burden and total job count. Smaller batches reduce the execution burden of each job but require more submissions, while larger batches reduce the number of jobs but increase the computational burden of

Table 5.3: *Deployment settings for quantum hardware*

Setting	Value
Framework Section	Vision Transformer (ViT)
Datasets	*Brain Tumor Dataset*, *Sartaj Dataset*
Images per batch	50
Backend	IBM Quantum Heron
Available compute time	10 minutes
Hardware designation	IBM-fez

each execution. The resulting configuration is summarized in Table 5.3.

An important implication of these settings is that these hardware experiments should be interpreted differently from the simulation-based experiments of the previous chapters. In Chapters 3 and 4, the primary emphasis was on algorithmic performance and benchmarking. Here, the primary emphasis is on whether the quantum component of QTCM can be executed faithfully on a real device while preserving the same framework structure. The primary outcome is a successful implementation and validation.

5.5 Experimental Results and Discussion

This section discusses the execution of QTCM on the two brain MRI datasets introduced in Chapter 4. Because the batching strategy has already been defined, the discussion focuses on feasibility, behavior, and dataset-level differences. The principal result of this chapter is that QTCM, originally developed and benchmarked in simulation, executes on an IBM quantum backend while preserving its structure.

5.5.1 Deployment on the Brain Tumor Dataset

The *Brain Tumor Dataset* represents the larger of the two datasets and therefore provides the more demanding test of hardware feasibility. Its 7,023 images produce 141 batches under the configuration in Table 5.2. This makes the dataset useful for evaluating whether repeated execution across many smaller subsets can preserve the preprocessing and quantum execution structure from previous experiments.

The successful execution of the *Brain Tumor Dataset* workload was significant because it showed that QTCM could be mapped to an IBM quantum backend for a multi-class MRI dataset under the batching and runtime constraints used in this study. The classical stages of the framework remained unchanged, and the quantum stage was transferred from simulation to the IBM backend without requiring a redesign of the algorithm. This result indicates that the quantum transformer portion of QTCM is not limited to simulation and can be executed as a hardware-based stage under the constraints considered in this experiment.

The deployment also highlights the practical limitations of real-hardware execution. The larger dataset size required more batch repetitions, which increases runtime budgets. In simulation, the framework can be studied over broad parameter sweeps and repeated trials. On hardware, the available computation window imposes a stricter design constraint. The importance of the deployment on the *Brain Tumor Dataset* therefore lies in establishing that QTCM can be executed on a real backend for a dataset of realistic scale in medical imaging.

5.5.2 Deployment on the Sartaj Dataset

The deployment on the *Sartaj Dataset* provides a second point of validation with a smaller total image count and class distribution. This dataset serves as a complementary test of hardware portability. Whereas the *Brain Tumor Dataset* deployment emphasizes a larger number of batches, the *Sartaj Dataset* study emphasizes whether the same deployment strategy remains effective when applied to a smaller brain MRI dataset without reduced framework performance.

Because the *Sartaj Dataset* contains fewer total images, it requires fewer batches than the *Brain Tumor Dataset*. The same preprocessing, batch size, and DQC1 method are used for both tests. The second dataset therefore evaluates whether the hardware execution procedure remains consistent when the total number of required batches is lower.

The hardware execution on *Sartaj Dataset* is also important because it mirrors the role

Table 5.4: *Hardware deployment performance of QTCM on IBM Quantum Heron using Brain Tumor Dataset and Sartaj Dataset*

Dataset	Pairwise F-score (F_P)	B-Cubed F-score (F_B)	NMI
Brain Tumor Dataset	89.7	90.0	90.9
Sartaj Dataset	90.1	90.5	91.6

this dataset played in Chapter 4. In that chapter, *Sartaj Dataset* served as a complementary benchmark to *Brain Tumor Dataset*, showing that DQC1-projected was a generalized solution beyond a single dataset environment. In this experiment, the successful classification of the *Sartaj Dataset* validates that the real-hardware execution of QTCM is reproducible across more than one clinically relevant brain MRI dataset.

Table 5.4 lists the clustering performance of QTCM when deployed on IBM Quantum Heron. Across both the *Brain Tumor Dataset* and *Sartaj Dataset*, the framework maintains consistent performance under real-hardware constraints, with all three evaluation metrics remaining near 90%.

5.6 Observations on Batching and Hardware Constraints

Several practical observations emerge from the hardware-based experiment. First, the batching strategy defined earlier determines the scale at which QTCM can be executed on the selected backend. The batch size of 50 images keeps each execution unit small enough for the device constraints while still allowing the complete dataset to be evaluated through repeated runs of the system.

Second, the deployment reinforces the importance of the hybrid design of QTCM. The classical stages, including image preprocessing and deep feature extraction, absorb the computational burden associated with large MRI datasets. The quantum hardware is used selectively for the circuit-based stage. This division is consistent with the broader literature on quantum machine learning in the noisy intermediate-scale quantum (NISQ) era, where hybrid methods are often the most realistic strategy for practical applications[10;12;34].

Rather than replacing the full method with a purely quantum model, QTCM uses quantum hardware only where quantum computation is central to the proposed design.

Third, the deployment illustrates the gap between algorithmic success in simulation and operational success on hardware. A method can achieve strong clustering behavior under simulation and still require substantial engineering adaptation for use on a real device. Batched execution, runtime budgeting, and hardware workload management are the primary concerns in these experiments. Each of these factors is also a primary consideration for quantum engineers deploying and testing their systems. These do not diminish the framework; rather, they show the practical steps required to move a hybrid quantum-classical system toward real deployment.

Finally, the IBM Quantum Heron results provide evidence that current hardware can execute a constrained hybrid quantum-classical medical imaging experiment. This result is important, but its scope is limited. The experiment shows that selected quantum stages of QTCM can be executed on a superconducting backend; it does not show that current quantum devices can independently support purely-quantum MRI classification.

5.7 Limitations of the Hardware Execution Study

The hardware-based results in this chapter should be interpreted as an implementation study rather than as evidence of full-scale clinical readiness or quantum advantage. The primary contribution is to show that the quantum stage of QTCM can be executed on IBM Quantum Heron under practical device constraints. The results do not establish that current quantum hardware can replace classical medical image classification systems, nor do they show that the full computational burden of MRI classification can be shifted to a quantum processor.

Several limitations affect the interpretation of the results. First, the hardware experiment relies on batching because the full datasets cannot be executed as single quantum workloads within the available computation window. Although batching makes device-based execution possible, it also changes how the dataset is processed. Each batch is evaluated independently, and the final results reflect aggregated behavior across repeated executions rather than end-to-end execution over the complete datasets, as in the experiments detailed in Chapters 3 and 4.

Second, the hardware execution is constrained by the characteristics of the selected backend, including gate noise, runtime limits, circuit depth restrictions, and queue-based access. These constraints affect the number of circuits that can be executed and limit the extent of repeated trials, hyperparameter variation, and statistical analysis. As a result, the hardware results should be viewed as evidence of executable behavior under one practical configuration rather than as a complete characterization of performance across devices or operating conditions.

Third, the evaluation is limited to the two MRI datasets used in Chapter 4. These datasets provide useful benchmarks because they differ in size and class distribution, but they do not capture the full variability of clinical MRI data. Additional validation on larger, more diverse, and independently collected neuroimaging datasets would be required before stronger claims could be made about a generalized solution.

Finally, the comparison between simulation and hardware execution is affected by differences in runtime, noise, and resource availability. The near-90% hardware metric values reported in Table 5.4 show that the method remains operational under device constraints, but they should not be interpreted as a direct replacement for the higher simulation-based results reported in earlier chapters. Instead, the hardware study demonstrates that selected stages of the proposed hybrid method can be executed beyond simulation, while also highlighting the engineering constraints that must be addressed before larger-scale deployment is practical.

5.8 Summary

This chapter executed the Quantum Transformer for Clustering MRI (QTCM) framework on IBM Quantum hardware. The experiment preserved the hybrid quantum-classical structure developed in the previous chapters, including preprocessing, clustering, and quantum transformer denoising. By applying the batching strategy introduced in this chapter, the quantum stage was executed on a 156-qubit backend without altering the central design of QTCM. These experiments provide an initial demonstration of hardware-based execution for a hybrid quantum machine learning system applied to brain tumor classification.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

We presented and evaluated Quantum Transformer for Clustering MRI (QTCM) as a hybrid quantum-classical method for unsupervised brain tumor classification. The results in Chapter 3 show that QTCM can produce strong clustering performance across multiple brain MRI datasets. Across the five datasets evaluated, QTCM achieved a Pairwise F-score of 97.67%, B-Cubed F-score of 97.46%, and NMI of 94.94%. These findings indicate that the proposed method can identify meaningful structure in extracted MRI feature representations without relying only on supervised class labels.

The results in Chapter 3 also show where the quantum transformer component is most useful. For datasets with strong features, the quantum and classical transformer configurations produced similar results. For more difficult datasets, however, the quantum transformer component improved clustering performance by 4.04%. This suggests that the proposed quantum component is most beneficial when the clustering task includes more hard or noisy samples.

In Chapter 4, we introduced the *DQC1-projected* method. The results show that clustering algorithm choice has a measurable effect on performance. Among the evaluated methods, the *DQC1-projected* method produced the strongest results and outperformed the other tested methods. On the *Brain Tumor Dataset*, the method achieved 99.0% precision, 98.3% recall, and a 98.7% F-score. On the *Sartaj Dataset*, it achieved 98.8% precision, 98.2% recall, and a 98.5% F-score. These findings show that DQC1-projected improves cluster quality and resulting classification for this task.

In Chapter 5, we demonstrated that our method can be executed on IBM Quantum hardware. The hardware experiments produced near-90% clustering metric values on both tested datasets, with Pairwise F-score, B-Cubed F-score, and NMI values of 89.7%, 90.0%, and 90.9% on the *Brain Tumor Dataset* and 90.1%, 90.5%, and 91.6% on the *Sartaj Dataset*. These results show that the quantum stage of the framework can operate on quantum hardware while preserving the overall structure of the proposed method.

The hardware results also clarify the current practical limits of the approach. Real-device execution required batching, and the results were constrained by device noise and runtime limits. Overall, the results show that QTCM can support accurate unsupervised computer vision transformer methods, that *DQC1-projected* clustering improves performance across benchmark datasets, and that the quantum ViT portion of QTCM can be executed on present-day IBM Quantum hardware under practical constraints.

6.2 Future Work

This section summarizes several future research directions that can extend the presented work. Chapter 3 introduces Quantum Transformer for Clustering MRI (QTCM) as a hybrid quantum-classical method for unsupervised visual clustering of brain MRI data. Future work can further study which conditions the quantum transformer provides measurable advantages over a classical transformer. This can include evaluating QTCM on datasets with greater image noise, stronger class overlap, and more uneven class distributions. Additional work can also improve cluster construction by using variable cluster sizes, centroid selection, or methods that identify hard and noisy samples before processing.

Chapter 4 evaluates multiple clustering methods and introduces the *DQC1-projected* method. Future work can examine additional dimensionality reduction strategies to determine which components most strongly affect cluster compactness and class separation. The *DQC1-projected* method should also be evaluated on larger and more diverse MRI datasets to determine whether its performance remains stable outside those studied in the Chapter 4 experiments. Another useful extension would be a dataset-aware clustering selection method that recommends a clustering algorithm based on measurable properties of the extracted feature space.

Chapter 5 demonstrates that the quantum ViT of QTCM can be executed on IBM Quantum hardware. Future work can extend this hardware study by testing across multiple quantum backends with different noise levels, number of qubits, and runtime constraints. Since batching is currently required for real-device execution, future research should also examine class-balanced batches, feature-aware batches, and adaptive batch sizes to improve the stability of the results.

More broadly, future work should validate QTCM on larger, independently collected, and multi-institutional MRI datasets. This would help determine whether the proposed method generalizes across differences in scanner type, imaging protocol, and image resolution. Extending the method to volumetric MRI or multi-sequence MRI data, such as T1, T2, contrast-enhanced T1, and related imaging methods, would also provide a more robust medical imaging evaluation.

Chapter 7

Publications

This chapter lists the journal manuscripts submitted from the research presented in this dissertation.

7.1 Submitted Journal Manuscripts

1. **D. Couch**, X.-B. Nguyen, H.-Q. Nguyen, K. Luu, A. W. Malik, and S. U. Khan, "QTCM: A Quantum Transformer Framework for MRI Brain Image Clustering," submitted to *Quantum Machine Intelligence*, (2025). **Under Review**.

2. **D. Couch**, R. Salehi, A. W. Malik, and S. U. Khan, "Impact of Clustering Strategy Selection on Brain MRI Tumor Classification," submitted to *Quantum Information Processing*, (2026). **Under Review**.

Bibliography

[1] Daniela Pia Rosaria Chieffo, Federica Lino, Daniele Ferrarese, Daniela Belella, Giuseppe Maria Della Pepa, and Francesco Doglietto. Brain tumor at diagnosis: From cognition and behavior to quality of life. *Diagnostics*, 13(3), 2023. ISSN 2075-4418. doi: 10.3390/diagnostics13030541. URL https://www.mdpi.com/2075-4418/13/3/541.

[2] Yilong Liu, Alex T L Leong, Yujiao Zhao, Linfang Xiao, Henry K F Mak, Anderson Chun On Tsang, Gary K K Lau, Gilberto K K Leung, and Ed X Wu. A low-cost and shielding-free ultra-low-field brain MRI scanner. *Nat. Commun.*, 12(1):7238, dec 2021.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[4] Chun-Ling Lin and Kun-Chi Wu. Development of revised resnet-50 for diabetic retinopathy detection. *BMC bioinformatics*, 24(1):157, 2023.

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[6] Shahriar Hossain, Amitabha Chakrabarty, Thippa Reddy Gadekallu, Mamoun Alazab, and Md Jalil Piran. Vision transformers, ensemble model, and transfer learning leveraging explainable AI for brain tumor detection and classification. *IEEE J. Biomed. Health Inform.*, 28(3):1261–1272, March 2024.

[7] Francis Jesmar P. Montalbo. Tumbrain: A transformer with a unified mobile residual attention inverted network for diagnosing brain tumors from magnetic resonance scans. *Neurocomputing*, 611:128583, 2025. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2024.128583. URL https://www.sciencedirect.com/science/article/pii/S0925231224013547.

[8] Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4): 043001, nov 2019. doi: 10.1088/2058-9565/ab4eb5. URL https://dx.doi.org/10.1088/2058-9565/ab4eb5.

[9] Maria Schuld and Nathan Killoran. Quantum machine learning in feature hilbert spaces. *Physical Review Letters*, 122(4), February 2019. ISSN 1079-7114. doi: 10.1103/physrevlett.122.040504. URL http://dx.doi.org/10.1103/PhysRevLett.122.040504.

[10] Danyal Maheshwari, Begonya Garcia-Zapirain, and Daniel Sierra-Sosa. Quantum machine learning applications in the biomedical domain: A systematic review. *Ieee Access*, 10:80463–80484, 2022.

[11] Syed Muhammad Abuzar Rizvi, Usama Inam Paracha, Uman Khalid, Kyesan Lee, and Hyundong Shin. Quantum machine learning: Towards hybrid quantum-classical vision models. *Mathematics*, 13(16):2645, aug 2025.

[12] John Preskill. Quantum computing in the nisq era and beyond. *Quantum*, 2:79, 2018.

[13] Muhammad AbuGhanem and Hichem Eleuch. Nisq computers: a path to quantum supremacy. *IEEE Access*, 2024.

[14] Javeria Amin, Muhammad Almas Anjum, Muhammad Sharif, Saima Jabeen, Seifedine Kadry, and Pablo Moreno Ger. A new model for brain tumor detection using ensemble transfer learning and quantum variational classifier. *Comput. Intell. Neurosci.*, 2022: 3236305, April 2022.

[15] S Padmakala and S Uma Maheswari. Deep learning-based mri analysis for brain tumor detection: Insights from resnet and densenet models. In *2024 8th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, pages 1620–1626. IEEE, 2024.

[16] Eid Albalawi, Mahesh T R, Arastu Thakur, V Vinoth Kumar, Muskan Gupta, Surbhi Bhatia Khan, and Ahlam Almusharraf. Integrated approach of federated learning with transfer learning for classification and diagnosis of brain tumor. *BMC Med. Imaging*, 24(1):110, May 2024.

[17] Swaminathan Balasubramanian, Jyothi Mandala, Telu Venkata Madhusudhana Rao, and Alok Misra. Rf-shcnn: A combination of two deep models for tumor detection in brain using mri. *Biomedical Signal Processing and Control*, 88:105656, 2024. ISSN 1746-8094. doi: https://doi.org/10.1016/j.bspc.2023.105656. URL https://www.sciencedirect.com/science/article/pii/S1746809423010893.

[18] Ayesha Ghaffar. Brain tumor data, 2024. URL https://doi.org/10.17632/w4sw3s9f59.1.

[19] Sartaj Bhuvaji, Ankita Kadam, Prajakta Bhumkar, Sameer Dedge, and Swati Kanchan. Brain tumor classification (mri), 2025. URL https://www.kaggle.com/dsv/12745533.

[20] Md Saiful Islam Sajol and ASM Jahid Hasan. Benchmarking cnn and cutting-edge transformer models for brain tumor classification through transfer learning. In *2024 IEEE 12th International Conference on Intelligent Systems (IS)*, pages 1–6. IEEE, 2024.

[21] Pattabirama Mohan and G. Ramkumar. Detection and classification of brain tumor using fine tuned mobile net algorithm. In *2024 3rd International Conference for Advancement in Technology (ICONAT)*, pages 1–5, 2024. doi: 10.1109/ICONAT61936.2024.10775053.

[22] Mohiuddin Ahmed, Raihan Seraj, and Syed Mohammed Shamsul Islam. The k-means

algorithm: A comprehensive survey and performance evaluation. *Electronics*, 9(8):1295, 2020.

[23] R Suganya and R Shanthi. Fuzzy c-means algorithm-a review. *International Journal of Scientific and Research Publications*, 2(11):1, 2012.

[24] Daniel T. Larose and Chantal D. Larose. k -nearest neighbor algorithm, jun 2014.

[25] Rui Liu, Hong Wang, and Xiaomei Yu. Shared-nearest-neighbor-based clustering by fast search and find of density peaks. *information sciences*, 450:200–226, 2018.

[26] Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, 42(3):1–21, 2017.

[27] Geoffrey Stewart and Mahmood Al-Khassaweneh. An implementation of the hdbscan* clustering algorithm. *Applied Sciences*, 12(5):2405, 2022.

[28] Kamal Berahmand, Farid Saberi-Movahed, Razieh Sheikhpour, Yuefeng Li, and Mahdi Jalili. A comprehensive survey on spectral clustering with graph structure learning. *arXiv preprint arXiv:2501.13597*, 2025.

[29] E. Knill and R. Laflamme. Power of one bit of quantum information. *Physical Review Letters*, 81(25):5672–5675, December 1998. ISSN 1079-7114. doi: 10.1103/physrevlett.81.5672. URL http://dx.doi.org/10.1103/PhysRevLett.81.5672.

[30] Roohollah Ghobadi, Jaspreet S. Oberoi, and Ehsan Zahedinejhad. The power of one qubit in machine learning, 2019. URL https://arxiv.org/abs/1905.01390.

[31] Mahsa Karimi, Ali Javadi-Abhari, Christoph Simon, and Roohollah Ghobadi. The power of one clean qubit in supervised machine learning. *Scientific Reports*, 13(1), November 2023. ISSN 2045-2322. doi: 10.1038/s41598-023-46497-y. URL http://dx.doi.org/10.1038/s41598-023-46497-y.

[32] Shalin Jose, Akshay Kannan Sairam, and Anil Shaji. Efficient classical simulation of the dqc1 circuit with zero discord. *Quantum*, 9:1895, October 2025. ISSN 2521-327X. doi: 10.22331/q-2025-10-28-1895. URL http://dx.doi.org/10.22331/q-2025-10-28-1895.

[33] Pattabirama Mohan and G. Ramkumar. An inception v3 based glioma brain tumor detection in mri images. In *2024 5th International Conference on Circuits, Control, Communication and Computing (I4C)*, pages 560–564, 2024. doi: 10.1109/I4C62240.2024.10748505.

[34] Veerakumar Pandi and Kalaiselvi Thiruvenkadam. An ablation study on hybrid quantum neural networks for enhancing brain tumour diagnosis. *Philosophical Magazine*, 0(0):1–33, 2025. doi: 10.1080/14786435.2025.2600762. URL https://doi.org/10.1080/14786435.2025.2600762.

[35] Akinyemi Omololu Akinrotimi, Joseph Bamidele Awotunde, and Israel Oluwabusayo Omotosho. Quantum support vector machines for early detection of neurodegenerative disorders using multimodal brain imaging. *Malaysian Journal of Science and Advanced Technology*, pages 210–221, 2025.

[36] Alessandro Poggiali, Alessandro Berti, Anna Bernasconi, Gianna M. Del Corso, and Riccardo Guidotti. Quantum clustering with k-means: A hybrid approach. *Theoretical Computer Science*, 992:114466, April 2024. ISSN 0304-3975. doi: 10.1016/j.tcs.2024.114466. URL http://dx.doi.org/10.1016/j.tcs.2024.114466.

[37] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. URL https://arxiv.org/abs/1409.1556.

[38] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

[39] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper

with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015. doi: 10.1109/CVPR.2015.7298594.

[40] Jeong-Hun Kim, Jong-Hyeok Choi, Young-Ho Park, Carson Kai-Sang Leung, and Aziz Nasridinov. Knn-sc: Novel spectral clustering algorithm using k-nearest neighbors. *IEEE Access*, 9:152616–152627, 2021. doi: 10.1109/ACCESS.2021.3126854.

[41] Krishna Kumar Sharma and Ayan Seal. Spectral embedded generalized mean based k-nearest neighbors clustering with s-distance. *Expert Systems with Applications*, 169: 114326, 2021. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2020.114326. URL https://www.sciencedirect.com/science/article/pii/S0957417420310186.

[42] Gongde Guo, Kai Yu, Hui Wang, Song Lin, Yongzhen Xu, and Xiaofeng Chen. Quantum hierarchical agglomerative clustering based on one dimension discrete quantum walk with single-point phase defects. *Computers, Materials & Continua*, 65(2): 1397–1409, 2020. ISSN 1546-2226. doi: 10.32604/cmc.2020.011399. URL http://www.techscience.com/cmc/v65n2/39883.

[43] Ming Zhao, Yong Wei Teo, Siliang Liu, Tat-Seng Chua, and Ramesh Jain. Automatic person annotation of family photo album. In Hari Sundaram, Milind Naphade, John R. Smith, and Yong Rui, editors, *Image and Video Retrieval*, pages 163–172, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. ISBN 978-3-540-36019-3.

[44] Jingyu Cui, Fang Wen, Rong Xiao, Yuandong Tian, and Xiaoou Tang. Easyalbum: an interactive photo annotation system based on face clustering and re-ranking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, page 367–376, New York, NY, USA, 2007. Association for Computing Machinery. ISBN 9781595935939. doi: 10.1145/1240624.1240684. URL https://doi.org/10.1145/1240624.1240684.

[45] Yuandong Tian, Wei Liu, Rong Xiao, Fang Wen, and Xiaoou Tang. A face annotation framework with partial clustering and interactive labeling. In *2007 IEEE Conference*

on Computer Vision and Pattern Recognition, pages 1–8, 2007. doi: 10.1109/CVPR.2007.383282.

[46] Raúl V Casana-Eslava, Paulo JG Lisboa, Sandra Ortega-Martorell, Ian H Jarman, and José D Martín-Guerrero. Probabilistic quantum clustering. *Knowledge-Based Systems*, 194:105567, 2020.

[47] Lihua Hu, Hongkai Liu, Jifu Zhang, and Aiqin Liu. Kr-dbscan: A density-based clustering algorithm based on reverse nearest neighbor and influence space. *Expert Systems with Applications*, 186:115763, 2021. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2021.115763. URL https://www.sciencedirect.com/science/article/pii/S0957417421011374.

[48] Lei Yang, Dapeng Chen, Xiaohang Zhan, Rui Zhao, Chen Change Loy, and Dahua Lin. Learning to cluster faces via confidence and connectivity estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[49] Xuan-Bac Nguyen, Duc Toan Bui, Chi Nhan Duong, Tien D. Bui, and Khoa Luu. Clusformer: A transformer based clustering approach to unsupervised large-scale face and visual landmark recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10847–10856, June 2021.

[50] Xuan-Bac Nguyen, Hoang-Quan Nguyen, Samuel Yen-Chi Chen, Samee U Khan, Hugh Churchill, and Khoa Luu. Qclusformer: A quantum transformer-based framework for unsupervised visual clustering. In *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, volume 2, pages 347–352. IEEE, 2024.

[51] Jing Li, Song Lin, Kai Yu, and Gongde Guo. Quantum k-nearest neighbor classification algorithm based on hamming distance. *Quantum Information Processing*, 21(1):18, 2022.

[52] Enrico Zardini, Enrico Blanzieri, and Davide Pastorello. A quantum k-nearest neighbors algorithm based on the euclidean distance estimation. *Quantum Machine Intelligence*, 6(1), Apr 2024. doi: 10.1007/s42484-024-00155-2.

[53] Congcong Feng, Bo Zhao, Xin Zhou, Xiaodong Ding, and Zheng Shan. An enhanced quantum k-nearest neighbor classification algorithm based on polar distance. *Entropy*, 25(1), 2023. ISSN 1099-4300. doi: 10.3390/e25010127. URL https://www.mdpi.com/1099-4300/25/1/127.

[54] Li-Zhen Gao, Chun-Yue Lu, Gong-De Guo, Xin Zhang, and Song Lin. Quantum k-nearest neighbors classification algorithm based on mahalanobis distance. *Frontiers in Physics*, 10, Oct 2022. doi: 10.3389/fphy.2022.1047466.

[55] Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. Quantum support vector machine for big data classification. *arXiv preprint arXiv:1307.0471*, 2013.

[56] Carsten Blank, Daniel K. Park, June-Koo Kevin Rhee, and Francesco Petruccione. Quantum classifier with tailored quantum kernel. *npj Quantum Information*, 6(41), 2020. doi: 10.1038/s41534-020-0272-6.

[57] Ren-Xin Zhao, Jinjing Shi, and Xuelong Li. Qksan: A quantum kernel self-attention network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12): 10184–10195, 2024. doi: 10.1109/TPAMI.2024.3434974.

[58] R. Jenssen, K.E. Hild, D. Erdogmus, J.C. Principe, and T. Eltoft. Clustering using renyi's entropy. In *Proceedings of the International Joint Conference on Neural Networks, 2003.*, volume 1, pages 523–528 vol.1, 2003. doi: 10.1109/IJCNN.2003.1223401.

[59] Ramin Salehi, Asad Waqar Malik, Khoa Luu, and Samee U Khan. Unsupervised quantum machine learning using one clean qubit. In *Quantum Computing, Communication, and Simulation VI*, volume 13919, pages 251–263. SPIE, 2026.

[60] Kushal Batra, Kimberley M Zorn, Daniel H Foil, Eni Minerali, Victor O Gawriljuk, Thomas R Lane, and Sean Ekins. Quantum machine learning algorithms for drug

discovery applications. *Journal of chemical information and modeling*, 61(6):2641–2647, 2021.

[61] Rebecca Smith-Bindman, Marilyn L. Kwan, Emily C. Marlow, Mary Kay Theis, Wesley Bolch, Stephanie Y. Cheng, Erin J. Bowles, James R. Duncan, Robert T. Greenlee, Lawrence H. Kushi, and et al. Trends in use of medical imaging in us health care systems and in ontario, canada, 2000-2016. *JAMA*, 322(9):843, Sep 2019. doi: 10.1001/jama.2019.11456.

[62] Theodore Christakis, Karine Bannelier-Christakis, Claude Castelluccia, and Daniel Le Métayer. *Mapping the Use of Facial Recognition in Public Spaces in Europe*. PhD thesis, Université Grenoble Alpes (UGA), 2022.

[63] David Sculley. Web-scale k-means clustering. In *Proceedings of the 19th International Conference on World wide web*, pages 1177–1178, 2010.

[64] Dingsheng Deng. Dbscan clustering algorithm based on density. In *2020 7th International Forum on Electrical Engineering and Automation (IFEEA)*, pages 949–953, 2020.

[65] Alireza Taheri Dehkordi, Mohammad Javad Valadan Zoej, Hani Ghasemi, Ebrahim Ghaderpour, and Quazi K. Hassan. A new clustering method to generate training samples for supervised monitoring of long-term water surface dynamics using landsat data through google earth engine. *Sustainability*, 14(13), 2022. ISSN 2071-1050. doi: 10.3390/su14138046. URL https://www.mdpi.com/2071-1050/14/13/8046.

[66] Yang Wang, Xubing Yang, Li Zhang, Xijian Fan, Qiaolin Ye, and Liyong Fu. Individual tree segmentation and tree-counting using supervised clustering. *Computers and Electronics in Agriculture*, 205:107629, 2023. ISSN 0168-1699. doi: https://doi.org/10.1016/j.compag.2023.107629. URL https://www.sciencedirect.com/science/article/pii/S0168169923000170.

[67] Muhammad Munsarif and Muhammad Saman. Improving the quality of handwritten image segmentation using k-means clustering algorithms with spatial filters. *Computer Science and Information Technologies*, 5(1):38–45, 2024.

[68] Md. Nasif Safwan, Souhardo Rahman, Mahamodul Hasan Mahadi, Taharat Muhammad Jabir, and Iftekharul Mobin. Brain mri nd-5 dataset, 2024. URL https://dx.doi.org/10.21227/q2vt-tf46.

[69] Msoud Nickparvar. Brain tumor mri dataset, 2021. URL https://www.kaggle.com/dsv/2645886.

[70] Mohamed Metwaly Sherif, M. Elmetwaly, and Fares Abdalsamad. brain-tumor-dataset, 2020. URL https://www.kaggle.com/datasets/mohamedmetwalysherif/braintumordataset.

[71] Thomas Dubail. Brain tumors 256x256, 2023. URL https://www.kaggle.com/datasets/thomasdubail/brain-tumors-256x256.

[72] Ayesha Ghaffar. Brain tumor data. https://doi.org/10.17632/w4sw3s9f59.1, 2024.

[73] Edward B Fowlkes and Colin L Mallows. A method for comparing two hierarchical clusterings. *Journal of the American statistical association*, 78(383):553–569, 1983.

[74] Amit Bagga and Breck Baldwin. Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*, volume 1, pages 563–566. Citeseer, 1998.

[75] LN Fred Ana and Anil K Jain. Robust data clustering. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 2, pages II–II. IEEE, 2003.

[76] Aaron F McDaid, Derek Greene, and Neil Hurley. Normalized mutual information to evaluate overlapping community finding algorithms. *arXiv preprint arXiv:1110.2515*, 2011.

[77] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[78] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

[79] K. A. Dishner, B. McRae-Posani, A. Bhowmik, M. S. Jochelson, A. Holodny, K. Pinker, S. Eskreis-Winkler, and J. N. Stember. A survey of publicly available mri datasets for potential use in artificial intelligence research. *Journal of Magnetic Resonance Imaging*, 59(2):450–480, 2024. doi: 10.1002/jmri.29101.

[80] H. Byeon, M. Al-Kubaisi, A. K. Dutta, F. Alghayadh, M. Soni, M. Bhende, V. Chunduri, K. Suresh Babu, and R. Jeet. Brain tumor segmentation using neuro-technology enabled intelligence-cascaded u-net model. *Frontiers in Computational Neuroscience*, 18:1391025, 2024. doi: 10.3389/fncom.2024.1391025.

[81] Madona B Sahaai, G R Jothilakshmi, D Ravikumar, Raghavendra Prasath, and Saurav Singh. ResNet-50 based deep neural network using transfer learning for brain tumor classification. In *INTERNATIONAL CONFERENCE ON RECENT INNOVATIONS IN SCIENCE AND TECHNOLOGY (RIST 2021)*. AIP Publishing, 2022.

[82] Rakesh Kumar Yadav, Abhishek Kumar Mishra, DK Jang Bahadur Saini, Hemlata Pant, RG Biradar, and Pranati Waghodekar. A model for brain tumor detection using a modified convolution layer resnet-50. *Indian Journal of Information Sources and Services*, 14(1):29–38, 2024.

[83] Marco Gagliardi, Danilo Maurmo, Tommaso Ruga, Eugenio Vocaturo, and Ester Zumpano. Brainvision: A hybrid explainable artificial intelligence framework for brain mri analysis. *Image and Vision Computing*, 161:105629, 2025. ISSN 0262-8856. doi: https://doi.org/10.1016/j.imavis.2025.105629. URL https://www.sciencedirect.com/science/article/pii/S0262885625002173.

[84] Syed Masooma Ali, K Vijayakumar, Mathiyazhagan Narayanan, and Illavarason. Deep transfer learning framework for brain tumor detection in MRI slices: A study. In *2025 International Conference on Frontier Technologies and Solutions (ICFTS)*, pages 1–6. IEEE, mar 2025.

[85] Vickram A S and Mathan Muthu C M. Automatic classification of brain mri slices into glioma/meningioma using deep transfer-learning. In *2025 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*, pages 1–5, 2025. doi: 10.1109/IITCEE64140.2025.10915481.

[86] Md Zakir Hossain Zamil, Md Raisul Islam, Md Anisur Rahman, Mumtahina Ahmed, Md Nahid Hasan, and Md. Mohsin Kabir. Optimizing magnetic resonance imaging analysis for brain tumors: A lightweight neural network approach. In *Proceedings of the 3rd International Conference on Computing Advancements*, ICCA '24, page 822–829, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400713828. doi: 10.1145/3723178.3723287. URL https://doi.org/10.1145/3723178.3723287.

[87] Muhannad Faleh Alanazi, Muhammad Umair Ali, Shaik Javeed Hussain, Amad Zafar, Mohammed Mohatram, Muhammad Irfan, Raed AlRuwaili, Mubarak Alruwaili, Naif H Ali, and Anas Mohammad Albarrak. Brain tumor/mass classification framework using magnetic-resonance-imaging-based isolated and developed transfer deep-learning model. *Sensors (Basel)*, 22(1):372, January 2022.

[88] Erdal Başaran. A new brain tumor diagnostic model: Selection of textural feature extraction algorithms and convolution neural network features with optimization algorithms. *Computers in Biology and Medicine*, 148:105857, 2022. ISSN 0010-4825. doi: https://doi.org/10.1016/j.compbiomed.2022.105857. URL https://www.sciencedirect.com/science/article/pii/S0010482522006114.

[89] Shubham Chitnis, Ramtin Hosseini, and Pengtao Xie. Brain tumor classification based on neural architecture search. *Sci. Rep.*, 12(1):19206, November 2022.

[90] M Ravinder, Garima Saluja, Sarah Allabun, Mohammed S Alqahtani, Mohamed Abbas, Manal Othman, and Ben Othman Soufiene. Enhanced brain tumor classification using graph convolutional neural network architecture. *Sci. Rep.*, 13(1):14938, September 2023.

[91] Atika Akter, Nazeela Nosheen, Sabbir Ahmed, Mariom Hossain, Mohammad Abu Yousuf, Mohammad Ali Abdullah Almoyad, Khondokar Fida Hasan, and Mohammad Ali Moni. Robust clinical applicable cnn and u-net based algorithm for mri classification and segmentation for brain tumor. *Expert Systems with Applications*, 238: 122347, 2024. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2023.122347. URL https://www.sciencedirect.com/science/article/pii/S095741742302849X.

[92] Kate Keahey, Jason Anderson, Zhuo Zhen, Pierre Riteau, Paul Ruth, Dan Stanzione, Mert Cevik, Jacob Colleran, Haryadi S. Gunawi, Cody Hammock, Joe Mambretti, Alexander Barnes, François Halbah, Alex Rocha, and Joe Stubbs. Lessons learned from the chameleon testbed. In *2020 USENIX Annual Technical Conference (USENIX ATC 20)*, pages 219–233. USENIX Association, July 2020. ISBN 978-1-939133-14-4. URL https://www.usenix.org/conference/atc20/presentation/keahey.

[93] Javier Gil Vidal and Dirk Oliver Theis. Calculus on parameterized quantum circuits, 2018. URL https://arxiv.org/abs/1812.06323.

[94] Aksultan Mukhanbet and Beimbet Daribayev. A hybrid quantum–classical architecture with data re-uploading and genetic algorithm optimization for enhanced image classification. *Computation*, 13(8), 2025. ISSN 2079-3197. doi: 10.3390/computation13080185. URL https://www.mdpi.com/2079-3197/13/8/185.

Appendix A

Acronyms

Acronyms

CLOPS	circuit layer operations per second
CNN	convolutional neural network
CUDA	Compute Unified Device Architecture
DARTS	differentiable architecture search
DBSCAN	density-based spatial clustering of applications with noise
DICOM	Digital Imaging and Communications in Medicine
DQC1	deterministic quantum computation with one qubit
EPLG	error per layered gate
F_B	B-Cubed F-score
F_P	Pairwise F-score
FFV	fused-feature vector
fMRI	functional magnetic resonance imaging
FN	false negative

FP	false positive
HDBSCAN	hierarchical density-based spatial clustering of applications with noise
***k*-NN**	*k*-nearest neighbors
MRI	magnetic resonance imaging
NISQ	noisy intermediate-scale quantum
NMI	normalized mutual information
PCA	principal component analysis
PQC	parameterized quantum circuit
QTCM	Quantum Transformer for Clustering MRI
QVC	quantum variational classifier
ResNet-50	50-layer residual network
SNN	shared nearest neighbor
SVM	support vector machine
***t*-SNE**	*t*-distributed stochastic neighbor embedding
TP	true positive
VGG16	Visual Geometry Group 16-layer convolutional neural network
ViT	Vision Transformer
VRAM	video random-access memory