Kasdi Merbah University Ouargla

Faculty of New Information and Communication Technologies

Department of Electronics and Telecommunications

End-of-Study Thesis

Title

Machine Learning in Medical Imaging

Presented by

- **Ouezzani Ouassim**
- **Debba Ahmed djad**

Ms Fella Charif	**President**	**UKM-Ouargla**
Mr Manseur Abdelghani	**Supervisor**	**UKM-Ouargla**
Mr Abderrazak Benchabane	**Examiner**	**UKM-Ouargla**

Academic Year: 2025 – 2026

Dedication

To our dear parents, who never ceased to encourage and support us throughout our academic journey.

To our brothers, sisters, and all our family members for their unwavering love and patience.

To our friends and colleagues who shared this experience with us.

To our supervisor, for their invaluable guidance and continuous support.

We dedicate this work.

Acknowledgements

First and foremost, we praise and thank Allah Almighty, who granted us the strength, patience, and knowledge to complete this work.

We would like to express our sincere gratitude to our supervisor for their continuous guidance, invaluable advice, and constructive feedback throughout the preparation of this thesis.

Our heartfelt thanks go to all the members of the jury for honouring us by evaluating this work.

We also extend our thanks to all the professors and teaching staff of the Department of Electronic and Telecommunications at the University of Ouargla, who contributed to our academic formation.

Finally, we thank all those who, directly or indirectly, contributed to the completion of this work.

Abstract

Medical imaging has become a cornerstone of modern clinical practice, enabling non-invasive visualization of internal anatomical structures and supporting accurate disease diagnosis. The rapid growth in both the volume and complexity of medical image data — particularly from Magnetic Resonance Imaging (MRI) — has created an urgent need for intelligent automated analysis tools capable of assisting clinicians in detection, classification, and patient monitoring.

This thesis addresses the problem of brain tumor classification from MRI data using two complementary machine learning approaches: a lightweight custom Convolutional Neural Network (CNN) designed and implemented in MATLAB, and a Support Vector Machine (SVM) baseline employing Histogram of Oriented Gradients (HOG) feature extraction combined with a Radial Basis Function (RBF) kernel. Both models were evaluated under identical CPU-only experimental conditions (Intel CPU, 8 GB RAM, no GPU) using the same dataset, preprocessing pipeline, and evaluation protocol.

The study utilizes a publicly available binary-labeled MRI dataset comprising 9,043 images (2,556 Normal; 6,487 Tumor). Five-fold cross-validation was applied, and performance was measured using accuracy, precision, recall, specificity, and F1-score.

The CNN achieved 98.75% accuracy, 99.48% precision, 96.22% recall, 99.79% specificity, and 97.82% F1-score, substantially outperforming the SVM (85.44% accuracy, 65.32% F1-score) across most metrics. However, the SVM attained perfect recall (100%), missing no tumor case, while training in approximately 15 minutes compared to 35 minutes for the CNN.

These results demonstrate the complementary strengths of deep learning and classical machine learning: CNN is preferable for high-accuracy clinical diagnosis, while SVM is suited to resource-constrained environments and initial medical screening.

Keywords: Machine Learning, Convolutional Neural Network (CNN), Support Vector Machine (SVM), Medical Imaging, Brain Tumor Classification, MRI, Computer-Aided Diagnosis, HOG Features.

Résumé

L'imagerie médicale est devenue un pilier de la pratique clinique moderne, permettant la visualisation non invasive des structures anatomiques internes. La croissance exponentielle des données d'imagerie par IRM (Imagerie par Résonance Magnétique) a créé un besoin urgent d'outils intelligents capables d'assister les cliniciens dans la détection et la classification des pathologies.

Ce mémoire traite le problème de la classification des tumeurs cérébrales à partir d'images IRM en utilisant deux approches d'apprentissage automatique complémentaires : un Réseau de Neurones Convolutif (CNN) léger conçu et implémenté dans MATLAB, et un classificateur SVM (Support Vector Machine) basé sur l'extraction de caractéristiques HOG (Histogramme de Gradients Orientés). Les deux modèles ont été évalués dans des conditions identiques sur un système CPU uniquement (Intel CPU, 8 Go de RAM, sans GPU) en utilisant la base de données publique Kaggle Brain Tumour MRI (9 043 images, binaire : Tumeur / Normal).

Le CNN a atteint une précision (accuracy) de 98,75%, une précision (precision) de 99,48%, un rappel (recall) de 96,22%, une spécificité de 99,79% et un F1-Score de 97,82%, surpassant significativement le SVM (85,44% d'accuracy, 65,32% de F1-Score). Cependant, le SVM a obtenu un rappel parfait (100%), ne manquant aucun cas de tumeur, avec un temps d'entraînement d'environ 15 minutes contre 35 minutes pour le CNN.

Ces résultats mettent en évidence les forces complémentaires des deux approches pour le diagnostic médical assisté par ordinateur : le CNN est préférable pour un diagnostic clinique de haute précision, tandis que le SVM est adapté aux environnements aux ressources limitées et au dépistage initial.

Mots-clés : Apprentissage automatique, Réseau de Neurones Convolutif (CNN), Machine à Vecteurs de Support (SVM), Imagerie médicale, Classification de tumeurs cérébrales, IRM, HOG.

ملخص

أصبح التصوير الطبي ركيزةً أساسيةً في الممارسة السريرية الحديثة، إذ يتيح تصوير الهياكل التشريحية الداخلية بطريقة غير جراحية .أدى التطور المتسارع في حجم بيانات التصوير بالرنين المغناطيسي (MRI) وتعقيدها إلى ضرورة تطوير أدوات تحليل آلية ذكية تُساعد الأطباء في الكشف عن الأمراض وتصنيفها ومتابعة المرضى.

تتناول هذه المذكرة مسألة تصنيف أورام الدماغ انطلاقاً من صور الرنين المغناطيسي، باستخدام نهجَين متكاملَين للتعلم الآلي :شبكة عصبية التفافية مخصصة (CNN) مُصمَّمة ومُنفَّذة فيMATLAB ، وآلة متجهات الدعم (SVM) المستندة إلى استخراج خصائص HOG مع نواة دالة الأساس الشعاعي .(RBF) كلا النموذجَين مُقيَّمَان في ظروف تجريبية متطابقة على نظام معالج مركزي فقطIntel CPU) ، ذاكرة وصول عشوائي 8 جيغابايت، بدون (GPU باستخدام قاعدة بيانات Kaggle للأورام الدماغية 9043) صورة، ثنائي :ورم / طبيعي.(

حققت الشبكة CNN دقةً بلغت98.75% ، ودقة الإيجابية99.48% ، واستدعاءً96.22% ، ونوعيةً 99.79%، ودرجة F1 بلغت97.82% ، متفوقةً بشكل ملحوظ على (85.44% SVM دقة،-F1 65.32% .(Scoreغير أن SVM حقق استدعاءً مثالياً (100%) دون إغفال أي حالة ورم، مع وقت تدريب أقصر 15) دقيقة مقابل 35 دقيقة.(

الكلمات المفتاحية :تعلم الآلة، شبكة عصبية التفافية(CNN) ، آلة متجهات الدعم(SVM) ، التصوير الطبي، تصنيف أورام الدماغ، الرنين المغناطيسي،.HOG

List of Abbreviations

The following abbreviations are used throughout this thesis:

AI	Artificial Intelligence
AUC	Area Under the Curve
BN	Batch Normalization
CAD	Computer-Aided Diagnosis
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CSF	Cerebrospinal Fluid
CT	Computed Tomography
CV	Cross-Validation
DIP	Digital Image Processing
DPI	Dots Per Inch
ELM	Extreme Learning Machine
F1	F1-Score (Harmonic mean of Precision and Recall)
FCM	Fuzzy C-Means
FN	False Negative
FP	False Positive
GAP	Global Average Pooling
GM	Gray Matter
GPU	Graphics Processing Unit
HOG	Histogram of Oriented Gradients
IRM	Imagerie par Résonance Magnétique (French: MRI)
LR	Learning Rate
MATLAB	Matrix Laboratory (MathWorks software)
ML	Machine Learning
MRI	Magnetic Resonance Imaging
PACS	Picture Archiving and Communication System
PPI	Pixels Per Inch
RBF	Radial Basis Function
ReLU	Rectified Linear Unit
RGB	Red, Green, Blue
RL	Reinforcement Learning
ROC	Receiver Operating Characteristic

SVM	Support Vector Machine
TN	True Negative
TP	True Positive
WM	White Matter

Table of Contents

Dedication i

Acknowledgements ii

Abstract iii

Résumé iv

ملخص v

List of Abbreviations vi

List of Figures vii

List of Tables viii

General introduction 1

CHAPTER 1 **Theoretical Framework and Machine Learning Techniques for Brain Tumor Classification** 3

1.1 Introduction: 4

1.2 Fundamentals of Digital Image Processing 4

1.2.1 Image Characteristics: 5

1.2.1.1 Pixel: 5

1.2.1.2 Resolution and Definition: 5

1.2.1.3 Texture: 6

1.2.1.4 Noise: 6

1.2.1.5 Luminance: 6

1.2.1.6 Histogram: 6

1.2.1.7 Contrast: 7

1.2.3 Image Processing System 7

1.2.4 Digital Image Processing Techniques 7

1.3 Medical Imaging 7

1.4 Some Elements of Brain Anatomy 9

1.4.1 Components of the Brain 9

1.4.2 Cerebral Tissues 10

1.5 Fundamentals of Machine Learning 11

1.5.1Definition of Machine Learning 11

1.5.2Types of Learning 11
1.6 Convolutional Neural Networks (CNN) 12
1.6.1 Definition of CNN 12
1.6.2 Basic CNN Structure 12
1.6.3 Popular CNN Architectures 13
1.6.4 CNN for Brain Tumor Classification 13
1.7 Support Vector Machines (SVM) 13
1.7.1 Definition of SVM 13
1.7.2 Working Principle of SVM 14
1.7.3 Feature Extraction for SVM 14
1.7.4 SVM for Brain Tumor Classification 14
1.8 Comparison Between CNN and SVM 15
1.9 Literature Review: Brain Tumor Classification Using Machine Learning 15
1.9.1 Studies Using CNN 16
1.9.2 Studies Using SVM 16
1.9.3 Comparing CNN and SVM 17
1.9.4 Summary of Previous Studies 18
1.10 Conclusion 18
CHAPTER 2 Proposed Methodology 19
2.1 Introduction 20
2.2 Dataset Description 20
Justification for Binary Classification Modification 20
2.3 Preprocessing and Data Preparation 21
2.3.1 Data Loading and Labeling 21
2.3.2 Data Split 22
2.3.3 Image Resizing 22
2.3.4 Data Augmentation 23
2.4 Architecture of the Proposed CNN 23
2.4.1 Detailed Architecture Description 24
2.5 Training the Model 25

2.5.1 Training Setup 25
2.5.2 Loss Function 26
2.6 Baseline Model with SVM (Support Vector Machine) 27
2.6.1 Feature Extraction 27
2.6.2 SVM Parameter Settings 28
2.7 Performance Metrics 28
2.7.1 Accuracy (Acc) 28
2.7.2 Confusion Matrix 29
2.8 Experimental Environment 29
2.9 Conclusion 29
3.1 **Introduction** 32
3.2 **Experimental Setup** 32
3.2.1 **Hardware and Software Environment** 32
3.2.2 **Dataset and Split Strategy** 33
3.3 **CNN Experimental Results** 34
3.3.1 **CNN Architecture** 34
3.3.2 **Training Configuration and Hyperparameters** 35
3.3.3 **Training Progress** 36
3.3.4 **CNN Classification Results** 37
3.3.5 **Sample CNN Predictions** 38
3.4 **SVM Baseline Results** 39
3.4.1 **HOG Feature Extraction** 39
3.4.2 **SVM Configuration** 40
3.4.3 **SVM Classification Results** 41
3.5 **Comparative Analysis: CNN vs. SVM** 42
3.5.1 **Quantitative Comparison** 42
3.5.2 **Discussion** 43
3.5.3 **Comparison with the Literature** 44
3.6 **Limitations and Future Work** 44
3.7 Conclusion 45

General Conclusion 46

List of Figures

Fig. 1.1 Example of a digital image and its pixel grid representation 5
Fig. 1.2 Image histogram — statistical distribution of pixel intensities 6
Fig. 1.3 Medical imaging modalities: Radiography (a) and MRI (b) 8
Fig. 1.4 MRI brain images in different planes 8
Fig. 1.5 Brain anatomy — encephalon cross-section 10
Fig. 1.6 Classification of Machine Learning paradigms 11
Fig. 1.7 CNN-based brain tumour classification studies overview 16
Fig. 1.8 SVM-based brain tumour classification example 17
Fig. 1.9 CNN vs. SVM comparative study results 17
Fig. 2.1 Sample MRI images from the Kaggle Brain Tumour dataset (Tumour / Normal) ... 21
Fig. 2.2 Data loading and labeling using MATLAB imageDatastore 21
Fig. 2.3 Dataset split strategy: 70% training / 30% testing 22
Fig. 2.4 Image resizing to 64×64×3 input format 22
Fig. 2.5 Data augmentation pipeline — rotation, translation, reflection 23
Fig. 2.6 Proposed CNN architecture — four convolutional blocks 24
Fig. 2.7 CNN training and validation workflow with early stopping 26
Fig. 2.8 SVM feature extraction and HOG-based classification pipeline 27
Fig. 2.9 Model evaluation pipeline — predictions, confusion matrix, and evaluation metrics 29
Fig. 2.10 Experimental environment — MATLAB R2024a, Intel CPU, 8 GB RAM 29
Fig. 2.11 Overall methodology pipeline — from dataset to comparison with SVM 30
Fig. 3.1 Representative MRI samples from the dataset (Tumour and Normal classes) 33
Fig. 3.2 Proposed CNN architecture — 4 convolutional blocks with Global Average Pooling 34
Fig. 3.3 CNN training progress — accuracy (top) and loss (bottom) over 50 epochs 37
Fig. 3.4 CNN confusion matrix and performance metrics (Accuracy: 98.75%) 38
Fig. 3.5 CNN sample predictions on test set MRI images (✓ correct / ✗ incorrect) 39
Fig. 3.6 HOG feature extraction pipeline — input MRI, HOG visualization, feature vector 40
Fig. 3.7 SVM confusion matrix and comparison with CNN (Accuracy: 85.44%) 41
Fig. 3.8 Grouped bar chart — CNN vs. SVM performance comparison 43

List of Tables

Table 1.1 Summary of previous studies on brain tumor classification 18
Table 2.1 Data augmentation techniques and parameters 23
Table 2.2 Proposed CNN architecture — layer-by-layer description 25
Table 2.3 CNN training hyperparameters and justifications 25
Table 2.4 SVM parameter settings 28
Table 2.5 Performance evaluation metrics and formulas 28
Table 3.1 Hardware and software experimental environment 33
Table 3.2 Dataset split for training and testing 34
Table 3.3 Proposed CNN architecture — layer-by-layer description 35
Table 3.4 CNN training hyperparameters and justifications 36
Table 3.5 SVM configuration and HOG feature extraction parameters 40
Table 3.6 Performance comparison — CNN vs. SVM across all metrics 42
Table 3.7 Comparison with existing literature on brain tumour MRI classification 44

General introduction

Medical imaging has become a cornerstone of modern clinical practice, enabling non-invasive visualization of internal anatomical structures and supporting accurate disease diagnosis. The rapid growth in both the volume and complexity of medical image data—particularly from **Magnetic Resonance Imaging (MRI)**—has created an urgent need for intelligent automated analysis tools capable of assisting clinicians in detection, classification, and patient monitoring.

Machine learning, and more specifically **deep learning**, has revolutionized medical image analysis by providing algorithms that can learn hierarchical representations directly from raw pixel data. **Convolutional Neural Networks (CNNs)** have emerged as the state-of-the-art approach for image classification tasks, achieving remarkable accuracy in various medical applications. However, CNNs typically require large datasets and substantial computational resources (often **GPUs**), which may not be available in all clinical or research settings. In parallel, classical machine learning methods such as **Support Vector Machines (SVM)** remain relevant due to their computational efficiency and good performance with smaller datasets, especially when combined with handcrafted feature extraction techniques like **Histogram of Oriented Gradients (HOG)**.

This thesis addresses the problem of **binary brain tumor classification** from MRI images using two complementary approaches: a custom lightweight **CNN** designed and implemented in **MATLAB**, and a baseline **SVM** classifier using **HOG** features with an **RBF kernel**. Both models were developed and evaluated under identical hardware constraints—specifically, an **Intel CPU with 8 GB RAM and no GPU**—to assess their feasibility in resource-limited environments. The study uses the publicly available **Kaggle Brain Tumor MRI dataset**, comprising **9,043 images** (2,556 normal, 6,487 tumor). A rigorous experimental protocol was followed, including **70/30 stratified data splitting**, **data augmentation** (rotation, translation, reflection, scaling), and **5-fold cross-validation** for the SVM. Performance was measured using **accuracy**, **precision**, **recall**, **specificity**, **F1-score**, and **confusion matrix** analysis.

The proposed CNN architecture consists of **four convolutional blocks** with batch normalization, ReLU activation, max-pooling, dropout, and global average pooling. The network was trained for **50 epochs** using the **Adam optimizer** with a piecewise learning rate schedule, **L2 regularization**, and gradient thresholding to prevent overfitting. The training was completed in approximately **35 minutes** on a CPU-only system. The CNN achieved an outstanding test accuracy of **98.75%**, with a precision of **99.48%**, recall of **96.22%**, specificity of **99.79%**, and F1-score of **97.82%**. In comparison, the HOG-based SVM, trained in about **15 minutes**, achieved an accuracy of **85.44%** and an F1-score of **65.32%**, with perfect recall (100%) but very low precision (48.50%), indicating a strong bias towards over-predicting the tumor class. These results demonstrate the complementary strengths of the two approaches: the **CNN** is superior for high-accuracy clinical diagnosis, while the **SVM** offers a faster, resource-efficient alternative suitable for initial screening in constrained environments.

This work is organized into three chapters. **Chapter One** presents the theoretical framework, covering digital image processing fundamentals, medical imaging modalities (MRI, CT), brain anatomy, machine learning basics, and detailed explanations of CNN and SVM architectures, along with a literature review of previous studies. **Chapter Two** describes the proposed methodology, including dataset preparation, preprocessing, data augmentation, CNN architecture design, training configuration, SVM implementation with HOG feature extraction, and evaluation metrics. **Chapter Three** presents the experimental results, comparative analysis, discussion of findings, and comparison with previous studies, followed by a general conclusion summarizing the key contributions, limitations, and future research directions.

CHAPTER 1

Theoretical Framework and Machine Learning Techniques for Brain Tumor Classification

Chapter 1 Theoretical Framework and Machine Learning Techniques for Brain Tumor Classification

1.1 1.1 Introduction:

Medical imaging, particularly Magnetic Resonance Imaging (MRI), has become essential for non-invasive brain tumor detection. The growing volume and complexity of MRI data require intelligent automated tools to assist clinicians in accurate classification. Digital image processing combined with machine learning algorithms—especially Convolutional Neural Networks (CNNs) and Support Vector Machines (SVM)—enables automatic feature extraction and accurate classification.

This chapter presents the theoretical foundations of digital image processing, medical imaging, and machine learning techniques for brain tumor classification. The chapter covers image processing fundamentals, MRI and CT imaging, brain anatomy, ML basics, CNN and SVM architectures, their comparison, and a review of previous studies.

1.2 Fundamentals of Digital Image Processing

1.2.1 Digital Image Definition and Representation

A digital image is a discrete representation of a continuous image whose surface is divided into fixed-size elements called pixels. Each pixel contains a numerical value representing a gray level or a color component depending on the image type [1]. Image digitization is the process of converting an analog image into a numerical format that can be processed by a computer system. It is commonly represented as a two-dimensional matrix of numerical values denoted by f(x,y) [2].

1.2.1 Image Characteristics:

A digital image is a structured set of visual information characterized by several fundamental properties that affect its quality and interpretability, including resolution, contrast, dynamic range, and noise.

1.2.1.1 Pixel:

A pixel (picture element) is the smallest unit of information in a digital image that can be processed by hardware and software systems. It represents a light intensity value in grayscale images or a set of color components in color images. A complete image is formed by the organized arrangement of pixels.

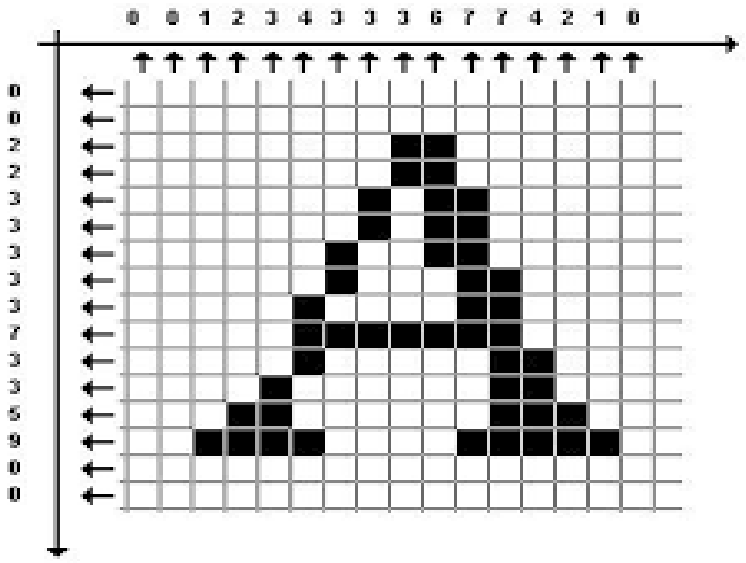

Fig. 1.1: Example of a digital image and its pixel grid representation

1.2.1.2 Resolution and Definition:

The concepts of definition and resolution are closely related to the pixel and describe the level of detail and spatial quality of a digital image [2].

- **Definition** refers to the total number of pixels that make up the image in terms of width and height. It is usually expressed as width × height.
  For example, an image with a definition of 1500 × 1200 contains 1500 pixels in width and 1200 pixels in height.
- **Spatial resolution** refers to the pixel density per unit length when an image is displayed or printed. It is commonly expressed in PPI (pixels per inch) or DPI (Dots Per Inch), and sometimes in pixels per centimeter. One inch equals 2.54 cm.

Two images may have the same definition but different resolutions depending on their display or print size.

1.2.1.3 Texture:

1.2.1.4 Noise:

Image noise refers to random variations in pixel intensity values caused by various factors including sensor limitations, transmission errors, or electronic disturbances. Common types include Gaussian noise, impulse noise, and Poisson noise [5].

1.2.1.5 Luminance:

Luminance refers to pixel intensity values; the average luminance can be used as a global brightness indicator but does not fully describe contrast distribution. Contrast measures the difference in luminance between image regions. A common measure is the Michelson contrast: $C = (L1 - L2) / (L1 + L2)$. Higher contrast improves object discrimination [1].

1.2.1.6 Histogram:

An image histogram represents the statistical distribution of pixel intensity levels, showing how frequently each gray level appears. It is used to evaluate contrast, brightness, and dynamic range of an image [2].

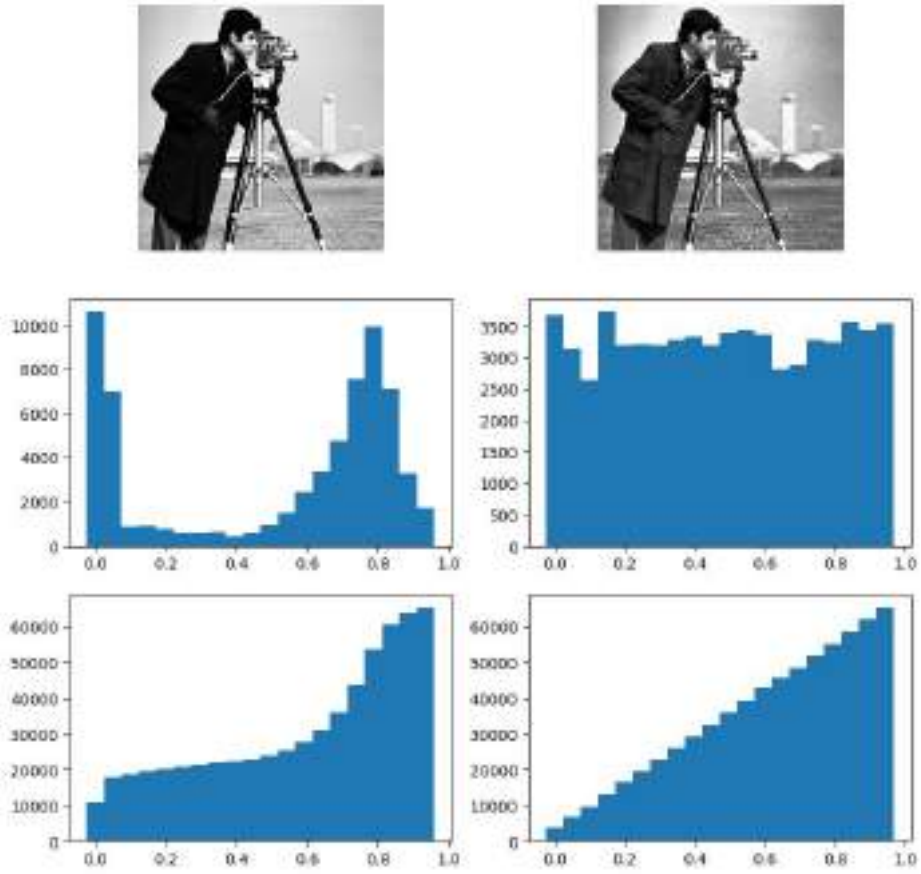

Fig. 1.2: Image histogram — statistical distribution of pixel intensities

1.2.1.7 Contrast:

Contrast measures the difference in luminance between image regions. A common measure is Michelson contrast:

$C = (L1 - L2) / (L1 + L2)$

Higher contrast improves object discrimination.

1.2.3 Image Processing System

An image processing system typically includes an image acquisition and digitization unit, storage system, processing unit (CPU/GPU), visualization system, and analysis software modules [8]. The acquisition and digitization system performs image sampling and quantization, mass storage stores digitized images, a display system visualizes results, and a central processing unit performs the various image processing operations [6].

1.2.4 Digital Image Processing Techniques

Digital image processing techniques are designed to exploit the information contained in images in order to improve their quality and make them easier to interpret [7]. Among the main digital image processing techniques, we find preprocessing (noise reduction, contrast enhancement, normalization), restoration (improving images degraded by motion blur, sensor noise, or distortion), and segmentation (partitioning an image into homogeneous regions corresponding to objects or areas of interest) [7].

1.3 Medical Imaging

Medical imaging includes various techniques that allow the visualization of the interior of the human body. There are multiple methods for acquiring and displaying medical images, each adapted to the observation of specific diseases and designed to achieve different diagnostic objectives [9].

a) Radiography:

Radiography enables the visualization of a body segment in two dimensions using X-rays. Its main drawback is patient exposure to ionizing radiation, although modern technologies have significantly reduced the delivered radiation dose **[9]**.



Fig. 1.3: Medical imaging modalities: Radiography (a) and MRI (b)

b) Computed Tomography (CT / Scanner):

Computed tomography allows cross-sectional visualization of body parts by acquiring successive slices. It is mainly used to study the brain, thorax, abdomen, and pelvis. In some cases, it requires the injection of an iodine-based contrast agent to improve the visualization of anatomical structures **[9]**.

c) Magnetic Resonance Imaging (MRI):

Magnetic Resonance Imaging is a relatively recent technique that produces anatomical images in multiple planes and can provide information about tissue composition. The first MRI acquisition was produced in 1973 by Lauterbur and Damadian **[10]**.

This technique has enabled major technological and scientific advances, particularly in disease detection and diagnosis **[11]**.

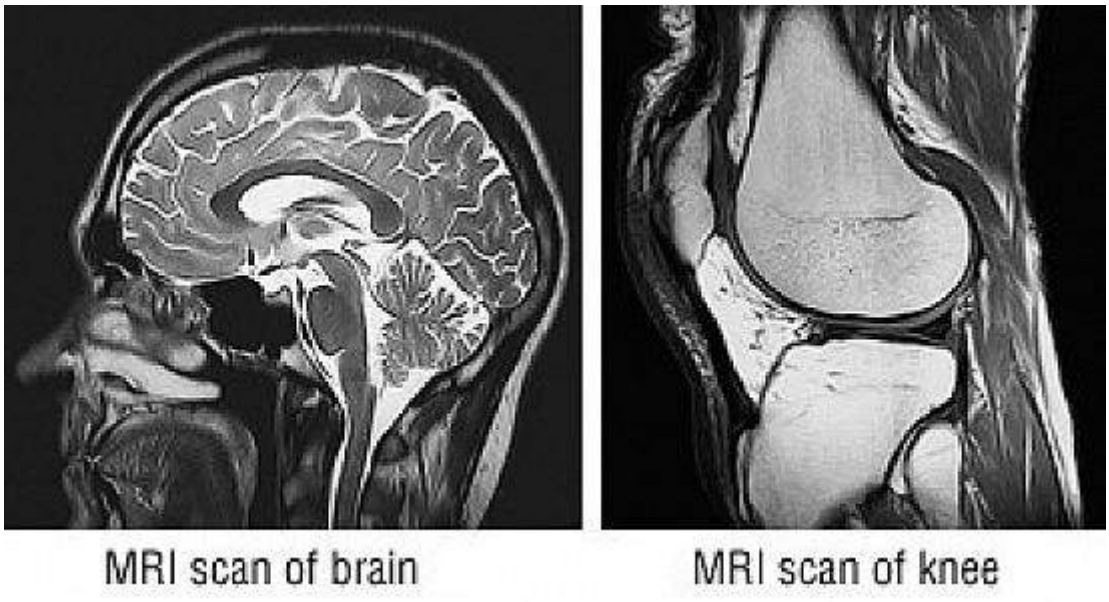

Fig. 1.4: MRI brain images in different planes

1.4 Some Elements of Brain Anatomy

The encephalon is the part of the nervous system located inside the cranial cavity and resting at the base of the skull. Its average weight ranges from about 1400 to 1800 grams. It is surrounded and protected by membranes called the meninges. The encephalon consists of the cerebrum, the cerebellum, and the brainstem (see Fig. 1.7(a)).

a) The Cerebrum

The cerebrum is the largest part of the central nervous system. It is mainly composed of two hemispheres separated by the interhemispheric fissure and connected by several structures such as the corpus callosum, the thalamus, and the hypothalamus **[11]**.

b) The Cerebellum

The cerebellum is functionally connected to major neural pathways: those that carry sensory information to the cortical areas for analysis, and those that transmit motor commands from the cortex to the muscles. The cerebellum plays an essential role in the coordination and precision of body movements and balance **[12]**.

c) The Brainstem

The brainstem represents a major relay center of the central nervous system between the cerebrum and the cerebellum. It is particularly responsible for regulating vital functions such as breathing and heart rate. It also serves as a conduction pathway for motor and sensory tracts and participates in pain control mechanisms **[12]**.

1.4.1 Components of the Brain

When observing a cross-section of the brain (see Fig. 1.7(b)), we notice a gray layer covering the outer surface called the gray matter, also known as the cerebral cortex. Neurons send their extensions, called axons, toward the interior of the brain. Since these extensions appear lighter in cross-sectional images, they form what is known as the white matter.

These axons connect different brain regions to each other, transmit motor commands to the peripheral nervous system, and carry sensory information (such as vision and hearing) to the brain for processing.

Motor regions, which control body movements, and somatosensory regions, which process touch sensations from the skin, are located around the central part of each hemisphere on both sides of the major groove that separates the frontal regions (anteriorly) from the parietal regions (posteriorly). This groove is known as the central sulcus.

Major sensory and motor pathways are crossed, meaning that the left hemisphere controls the right side of the body and receives tactile information from the right side, and vice versa **[12]**.

The brain is therefore mainly composed of three principal components: white matter (axons), gray matter (cortex), and cerebrospinal fluid (CSF).

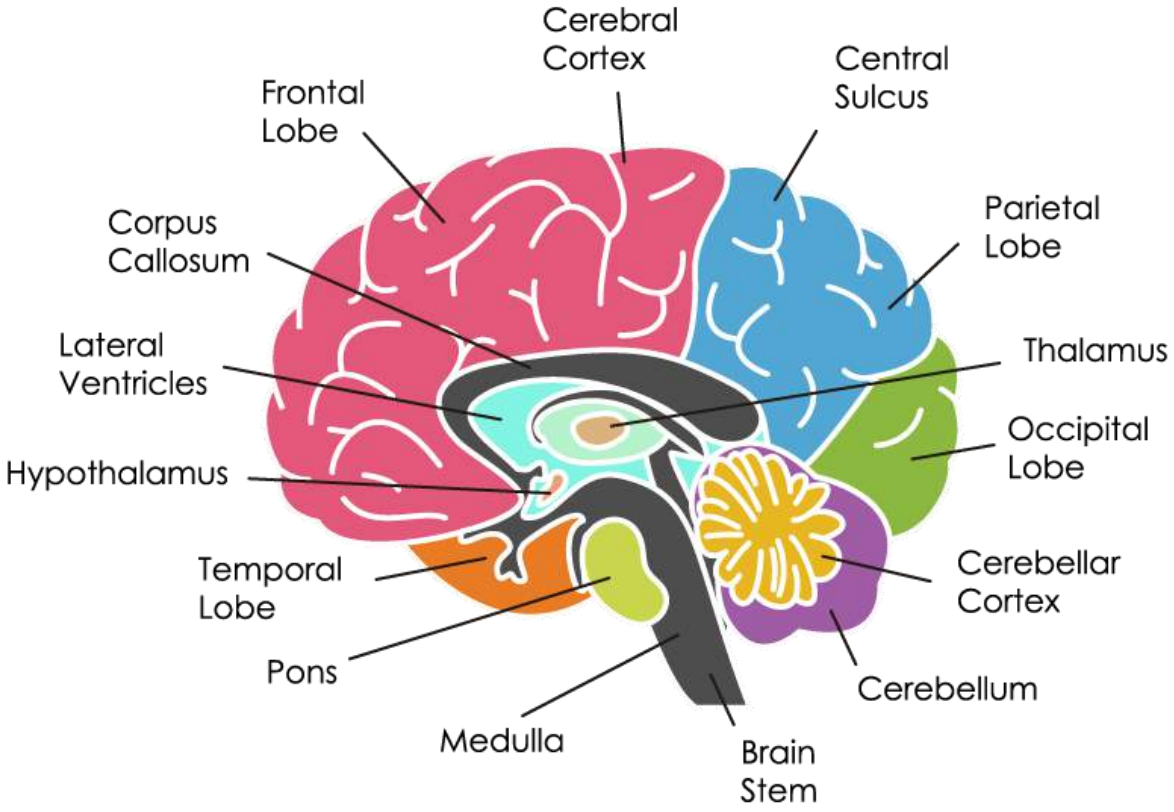

Fig. 1.5: Brain anatomy — encephalon cross-section

1.4.2 Cerebral Tissues

- **Cerebrospinal Fluid (CSF):**
  Cerebrospinal fluid is the fluid in which the brain and cerebellum are immersed. Its average volume is about 150 ml. It is absorbed by the cerebral venous system and continuously renewed. Its main functions include mechanical protection of the central nervous system against shocks, protection against infections through mediators of humoral and cellular immunity, and transport of hormones between distant regions of the brain [13].
- **Gray Matter (GM):**

Several structures are composed of gray matter. On one hand, there is the cerebral cortex located at the periphery of the brain, and on the other hand, the deep gray nuclei (basal ganglia). Gray matter is mainly composed of neuronal cell bodies and is therefore considered the primary site of cerebral activity and information processing **[11]**.

- **White Matter (WM):**

White matter (also called white substance) is located in the inner part of the brain. It mainly corresponds to the myelin sheath that surrounds neuronal axons and accelerates nerve signal conduction. These myelinated axons are grouped into bundles (similar to nerve tracts) to establish connections with other groups of neurons and brain regions **[11]**.

1.5 Fundamentals of Machine Learning

1.5.1Definition of Machine Learning

Machine Learning is a subfield of Artificial Intelligence that enables computers to learn from data and improve their performance on specific tasks without being explicitly programmed [15]. According to Mitchell [15], "A computer program learns from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E." In medical imaging applications, machine learning models learn to recognize patterns in images such as MRI scans, enabling automated detection and classification of abnormalities like tumors **[15]**.

1.5.2Types of Learning

Machine Learning paradigms are generally classified into four major categories **[23]**:

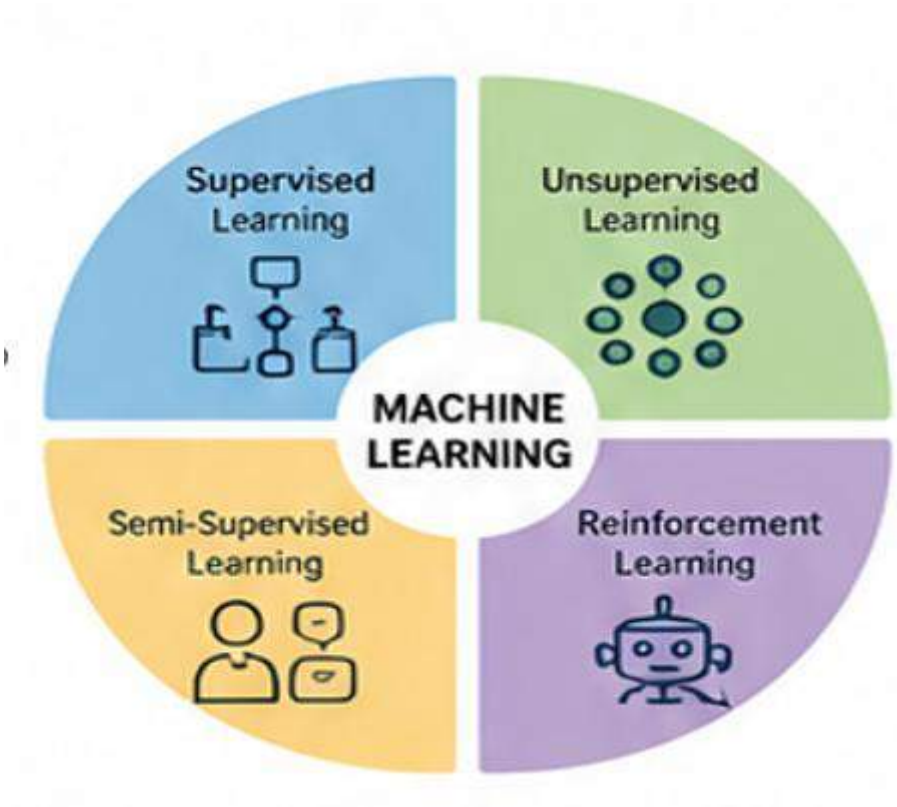

Fig. *1.6*: Classification of Machine Learning paradigms

Supervised Learning: Supervised learning uses labeled datasets where each input sample is associated with a known output label. In brain tumor classification, MRI images are typically

labeled by expert radiologists as either normal or pathological. This approach provides high classification accuracy and directly learns from ground-truth annotations. However, it requires large labeled datasets, and the annotation process is expensive and time-consuming. Inter-observer variability may also affect label quality [23].

Unsupervised Learning: Unsupervised learning operates on unlabeled datasets and aims to discover hidden patterns or structures within the data automatically. In medical imaging, unsupervised methods are commonly applied to tissue clustering, feature learning, and anomaly detection. While they do not require labeled data, they typically yield lower classification accuracy and limited clinical interpretability [23].

Semi-Supervised Learning: Semi-supervised learning combines a small amount of labeled data with a larger amount of unlabeled data. This approach is particularly important in medical imaging because manual annotation of MRI scans is costly and labor-intensive. Common techniques include pseudo-labeling, consistency regularization, and self-training methods. These methods reduce annotation effort and improve model generalization but involve complex training procedures and risk of propagating incorrect pseudo-labels [23].

Reinforcement Learning: Reinforcement learning (RL) is based on an agent interacting with an environment and learning through rewards and penalties. Although RL is less commonly used for brain tumor classification, it has been explored in interactive medical image segmentation, adaptive image analysis, and intelligent navigation of volumetric MRI data. RL offers adaptive decision-making and dynamic learning capability but suffers from high computational complexity and difficult reward function design [23].

1.6 Convolutional Neural Networks (CNN)

1.6.1 Definition of CNN

Convolutional Neural Networks are deep learning models specifically designed for processing structured grid data such as images [24]. CNNs learn to recognize patterns in images through a process called convolution, which applies learnable filters to extract hierarchical features. These models are inspired by the structure of the visual cortex in biological brains [26]. CNNs have become the dominant approach for computer vision tasks, including medical image classification.

1.6.2 Basic CNN Structure

A standard CNN architecture typically includes the following layers [24]:

Convolutional Layers: These layers apply learnable filters (kernels) to extract important image features such as edges, textures, and patterns. The convolution operation involves sliding a filter over the input image and computing dot products between the filter weights and local image regions [24].

Pooling Layers: Pooling layers reduce spatial dimensions and computational complexity while preserving essential information. Common pooling operations include max pooling and average pooling. Max pooling selects the maximum value within each window, while average pooling computes the mean value [24].

Fully Connected Layers: These layers perform the final classification based on extracted features. The high-level features learned by convolutional and pooling layers are flattened and passed through one or more fully connected layers, with the final layer typically using softmax activation for classification [24].

1.6.3 Popular CNN Architectures

Several CNN architectures have been developed and widely adopted in the computer vision community [24]:

AlexNet (2012): The first successful deep CNN that demonstrated the power of deep learning for image classification. It consists of five convolutional layers followed by three fully connected layers [26].

VGG-16 (2014): A simple architecture using 3×3 filters throughout, with increasing depth (16 layers). Its simplicity and effectiveness have made it a popular choice for transfer learning [24].

Inception (2014): Introduces multi-scale convolution operations using filters of different sizes within the same layer, allowing the network to capture features at multiple scales [24].

ResNet (2015): Introduces residual skip connections that enable training of very deep networks (up to 152 layers) by preventing vanishing gradients. ResNet has become foundational for transfer learning in medical image classification tasks [16].

1.6.4 CNN for Brain Tumor Classification

CNNs are highly effective for brain tumor classification because they automatically extract image features, learn hierarchical representations, and achieve state-of-the-art accuracy [24]. However, CNNs generally require large datasets, long training time, high computational resources, and typically GPU acceleration. Despite these requirements, they remain the preferred approach for high-accuracy clinical diagnosis [16].

1.7 Support Vector Machines (SVM)

1.7.1 Definition of SVM

Support Vector Machine (SVM) is a supervised machine learning algorithm introduced by Cortes and Vapnik (1995) [19]. SVM aims to find the optimal hyperplane that separates data points from different classes with the maximum possible margin. The algorithm maps data

into a higher-dimensional feature space and determines the decision boundary that maximizes class separation [19].

1.7.2 Working Principle of SVM

SVM maps data into a higher-dimensional feature space and determines the decision boundary that maximizes class separation. For non-linearly separable data, SVM uses kernel functions to implicitly transform the data into a space where linear separation becomes possible [19]. Common kernel functions include:

1. **Linear kernel:** Suitable for linearly separable data.
2. **Polynomial kernel:** Captures polynomial relationships between features.
3. **Radial Basis Function (RBF) kernel:** The most widely used kernel in medical image classification, capable of modeling complex nonlinear relationships [19].

The RBF kernel is defined as $K(x, x') = \exp(-\gamma \|x - x'\|^2)$, where γ is the kernel scale parameter that controls the influence of individual training examples [19].

1.7.3 Feature Extraction for SVM

Unlike CNNs, SVM models require handcrafted feature extraction techniques to convert raw images into meaningful feature vectors [31]. Common feature extraction methods include:

- **HOG (Histogram of Oriented Gradients):** Captures edge orientation information by computing histograms of gradient directions in local image regions [31].
- **GLCM (Gray-Level Co-occurrence Matrix):** Extracts texture relationships by analyzing the spatial co-occurrence of pixel intensities [3].
- **LBP (Local Binary Patterns):** Describes local texture patterns by thresholding each pixel against its neighbors [4].
- **Haralick Features:** Statistical texture descriptors derived from the GLCM, including contrast, correlation, energy, and homogeneity [3].

Among these, HOG is particularly effective for capturing edge and shape information, making it suitable for medical image classification.

1.7.4 SVM for Brain Tumor Classification

SVM remains widely used because it performs well with limited datasets, is computationally efficient, does not require GPUs, and works effectively with high-dimensional features [19]. However, SVM performance strongly depends on the quality of extracted handcrafted features. This limitation makes SVM less flexible than CNNs for complex medical image

classification tasks, but its computational efficiency makes it suitable for resource-constrained environments and initial screening applications [20].

1.8 Comparison Between CNN and SVM

Both CNN and SVM have been widely applied to brain tumor classification, each with distinct advantages and limitations [20][24]:

Accuracy: CNN generally achieves very high accuracy due to its ability to learn hierarchical features directly from raw pixels. SVM achieves moderate to high accuracy but is limited by the quality of handcrafted features [20].

Dataset Requirement: CNN requires large datasets for effective training. SVM performs well with small to medium datasets [19].

Feature Extraction: CNN performs automatic feature extraction through its convolutional layers. SVM requires manual feature extraction using techniques like HOG, GLCM, or LBP [31].

Training Time: CNN training is typically long and computationally intensive. SVM training is relatively short and efficient [20].

Hardware Requirement: CNN typically requires GPU acceleration for practical training. SVM can run effectively on CPU-only systems [19][24].

Interpretability: CNN models have low interpretability due to their complex hierarchical representations. SVM offers moderate interpretability through its support vectors and margin concept [19].

Key observations from comparative studies show that CNN achieves superior classification accuracy but at the cost of higher computational requirements. SVM is computationally cheaper and more efficient but depends on handcrafted features [20]. This thesis compares both methods under identical CPU-only experimental conditions to evaluate their performance in resource-constrained environments.

1.9 Literature Review: Brain Tumor Classification Using Machine Learning

This section reviews previous studies that applied machine learning techniques, particularly CNN and SVM methods, for brain tumor classification using MRI images.

1.9.1 Studies Using CNN

CNNs have demonstrated remarkable performance in brain tumor classification tasks due to their ability to automatically learn discriminative features from raw image data.

Madani and Oshah (2025) proposed a CNN-based system for binary brain tumor classification using publicly available MRI datasets. Their model achieved approximately 90% classification accuracy and demonstrated that systematic data augmentation substantially improves model generalization when training data is limited [17].

He et al. (2016) introduced the ResNet architecture, enabling training of very deep networks through residual skip connections that prevent vanishing gradients. ResNet variants have since become foundational for transfer learning in medical image classification tasks [16].

Bhimavarapu et al. (2024) proposed a hybrid approach combining Fuzzy C-Means (FCM) clustering for tumor region segmentation with an Extreme Learning Machine (ELM) classifier, achieving 98.56% accuracy on a multi-class MRI dataset [18].

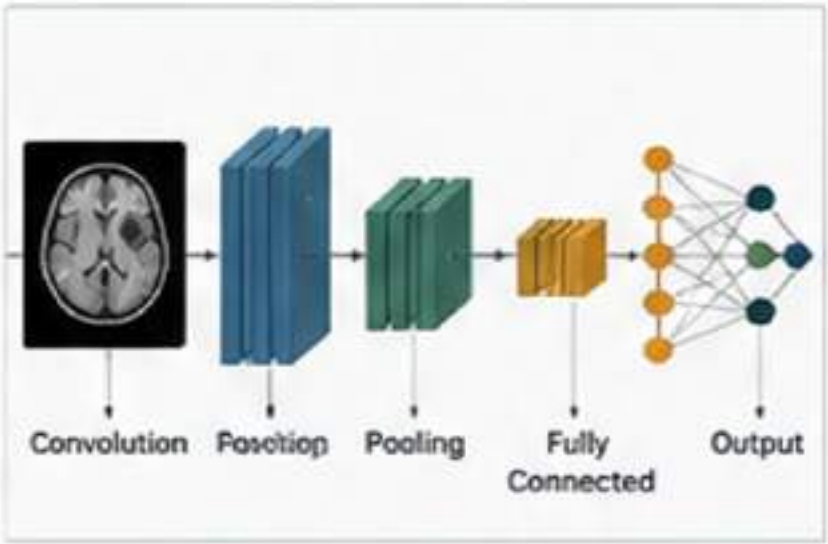

Fig. 1.7: CNN-based brain tumour classification studies overview

1.9.2 Studies Using SVM

SVM has been widely applied to brain tumor classification due to its computational efficiency and good performance with limited data.

Zacharaki et al. (2009) applied SVM with texture, shape, and intensity features extracted from multi-parametric MRI for multi-class brain tumor grading, achieving over 85% accuracy. Their study highlighted the critical importance of feature selection and kernel hyperparameter tuning for SVM performance [21].

Alqasemi et al. (2024) conducted a comparative study of CNN and SVM for MRI brain tumor classification under identical datasets. The CNN achieved 91% accuracy versus 82%

for the SVM, concluding that CNN provides higher accuracy at substantially greater computational cost [20].

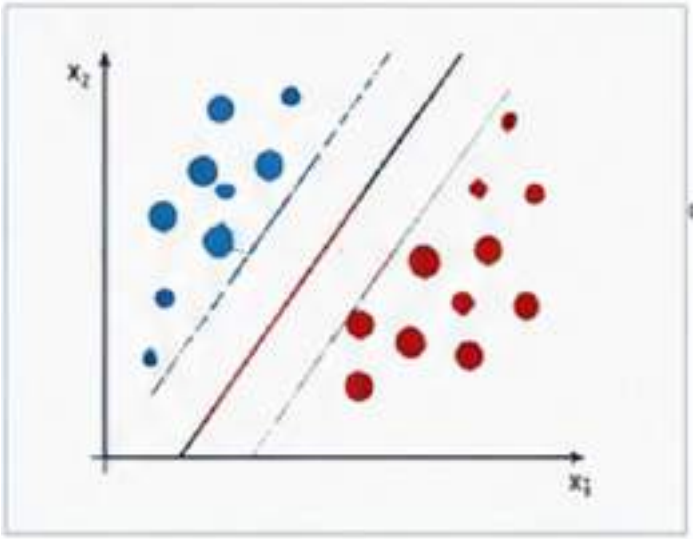

Fig. 1.8: SVM-based brain tumour classification example

1.9.3 Comparing CNN and SVM

Several studies compared CNN and SVM models under similar experimental conditions.

Alqasemi et al. (2024) compared CNN and SVM approaches for MRI brain tumor classification. Their CNN model achieved 91% accuracy, while the SVM achieved 82%. The study concluded that CNN generally provides higher accuracy but requires more computational resources and training time.

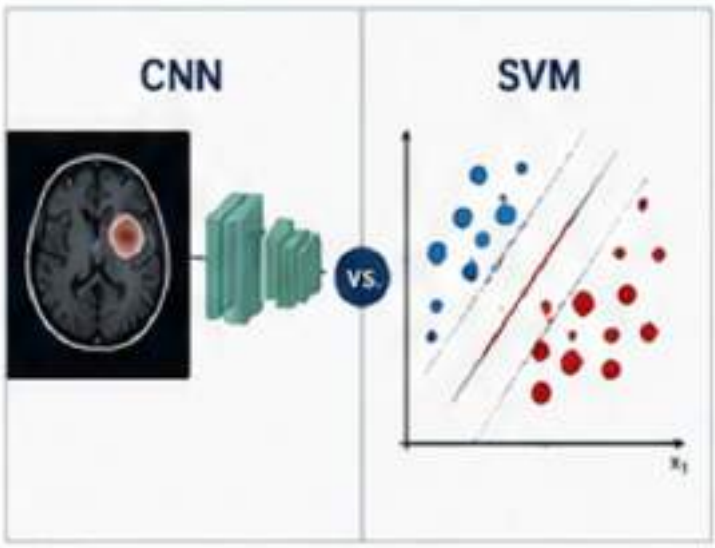

Fig. 1.9: CNN vs. SVM comparative study results

1.9.4 Summary of Previous Studies

The following table summarizes key studies on brain tumor classification using machine learning:

Study	Year	Method	Accuracy	Main Contribution
Madani & Oshah	2025	CNN	~90%	CNN outperformed traditional ML
Zhang	2026	ResNet18	94.06%	Effective transfer learning
Bhimavarapu et al.	2024	FCM + ELM	98.56%	High classification accuracy
Alqasemi et al.	2024	CNN vs. SVM	CNN: 91% / SVM: 82%	CNN more accurate but resource-intensive

Table 1.1: Summary of previous studies on brain tumor classification

1.10 Conclusion

This chapter provided a comprehensive theoretical framework for brain tumor classification using machine learning. It covered digital image processing fundamentals, MRI imaging, brain anatomy, and the core machine learning concepts. Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs) were detailed, including their architectures, working principles, and feature extraction methods such as HOG.

The comparative analysis showed that CNNs achieve higher accuracy through automatic feature learning, while SVMs offer computational efficiency for resource-constrained environments. The literature review identified a research gap: few studies compare both approaches under CPU-only conditions. Based on this, the next chapter presents the proposed methodology, including dataset preparation, preprocessing, CNN architecture, SVM implementation, and evaluation metrics.

CHAPTER 2

Proposed Methodology

CHAPTER 2 Proposed Methodology

2.1 Introduction

In this chapter, the generalized methodology for classifying brain tumors from MRI images is presented. The method is based on training a manufactured Convolutional Neural Network (CNN) which was implemented and trained in MATLAB. The significant components associated with the methodology are the dataset, data preprocessing and augmentation, CNN architecture design, model training, and evaluation. As well as that a Support Vector Machine (SVM) is applied as a base line model to enhance the comparison of the results.

Furthermore, to ensure the robustness of the proposed approach, a 5-fold cross-validation technique was employed, and multiple evaluation metrics including precision, recall, F1-score, specificity, and AUC-ROC were calculated alongside accuracy. A comparative analysis with recent state-of-the-art studies is also provided to position this work within the current research landscape.

2.2 Dataset Description

We use the Brain Tumor MRI Dataset (publicly available on Kaggle) as research material in this study. It consists of 4 types of MRI images: glioma, meningioma, pituitary tumor, and normal. For the purpose of the present study, the dadaset was modified to become a binary classification problem by tjneg all tumor types into one "Tumor" class.

Justification for Binary Classification Modification

For the purpose of the present study, the dataset was modified to become a binary classification problem by merging all tumor types into one "Tumor" class. This modification was justified for the following reasons:

1. **Clinical Relevance**

   The primary diagnostic objective in many real-world screening scenarios is to first distinguish between the presence and absence of a tumor before identifying its specific type.

2. **Methodological Clarity**

   Simplifying the problem to a binary task enables a clearer and fairer comparison between the proposed CNN and the baseline SVM model by eliminating multi-class complexity.

3. **Alignment with Previous Work**

   Several recent studies have adopted binary classification as an initial step before extending to multi-class problems.

4. Hardware Constraints

Given the hardware limitations (Intel CPU, 8GB RAM, no GPU), binary classification reduces computational complexity while still validating the proposed methodology.

Future work will extend this approach to handle multi-class tumor type classification [14].

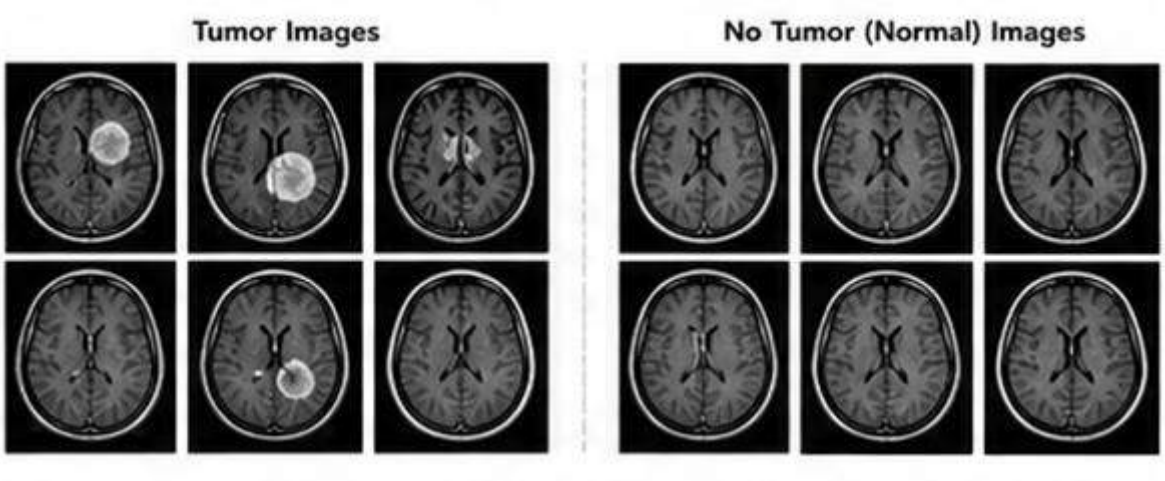

Fig. 2.1: Sample MRI images from the Kaggle Brain Tumour dataset (Tumour / Normal)

2.3 Preprocessing and Data Preparation

2.3.1 Data Loading and Labeling

The dataset was imported using the imageDatastore function in MATLAB, which automatically assigns labels according to the names of the folders. The dataset was divided into 70% for training and 30% for testing through the splitEachLabel function, ensuring that the distribution of classes remains stratified.

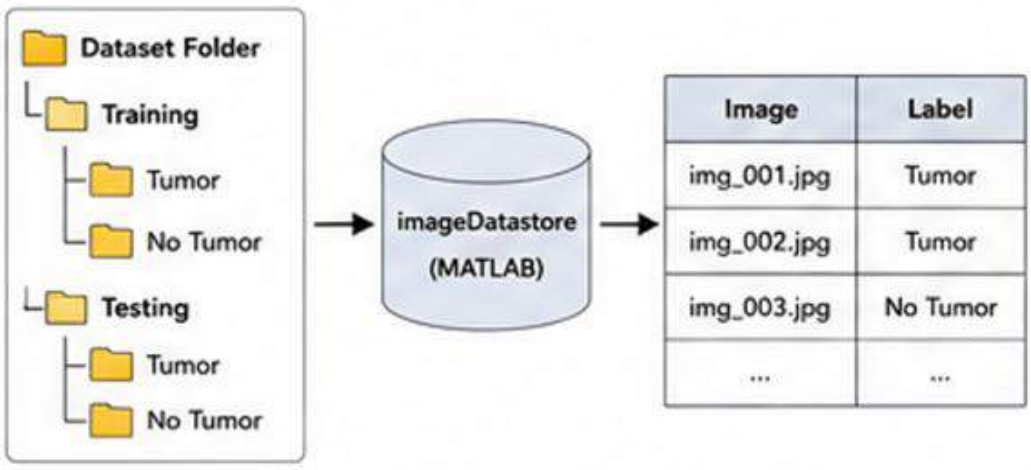

Fig. 2.2: Data loading and labeling using MATLAB imageDatastore

2.3.2 Data Split

The dataset was split into 70% training and 30% testing using splitEachLabel, ensuring a stratified distribution of classes.

1. 70% of the data was used for training
2. 30% was used for testing
3. Within the training data, 20% was held out as a validation set for monitoring training progress and early stopping.

This ensures that the distribution of classes remains stratified across all splits, preventing class imbalance issues.

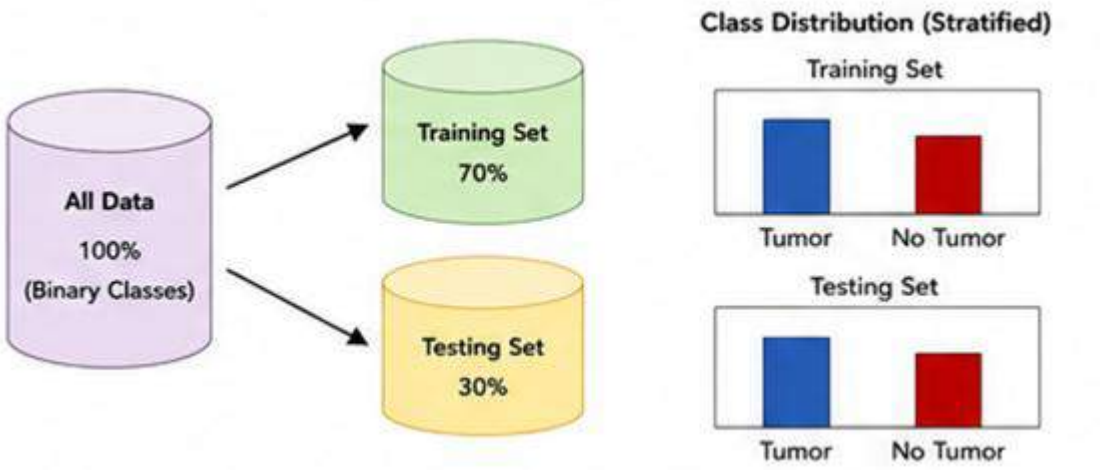

Fig. 2.3: Dataset split strategy: 70% training / 30% testing

2.3.3 Image Resizing

Every image was resized to dimensions of 64×64×3 pixels. Grayscale images were transformed to RGB format to align with the input requirements of the CNN.

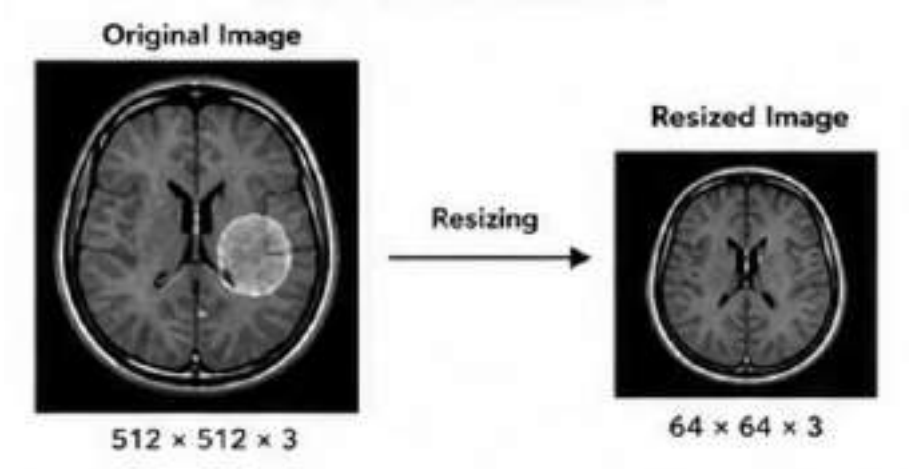

Fig. 2.4: Image resizing to 64×64×3 input format

2.3.4 Data Augmentation

Data augmentation techniques such as rotation, translation, and reflection were applied to improve generalization and reduce overfitting.

Augmentation Technique	Parameters	Purpose
Random Rotation	±30 degrees	Handle varying tumor orientations
Random X/Y Translation	±5 pixels	Simulate different tumor positions
Random Reflection	Horizontal	Mirror symmetry invariance
Random Scaling	±10%	Handle different tumor sizes
Random Shear	±5 degrees	Simulate image distortions

Table 2.1: Data augmentation techniques and parameters

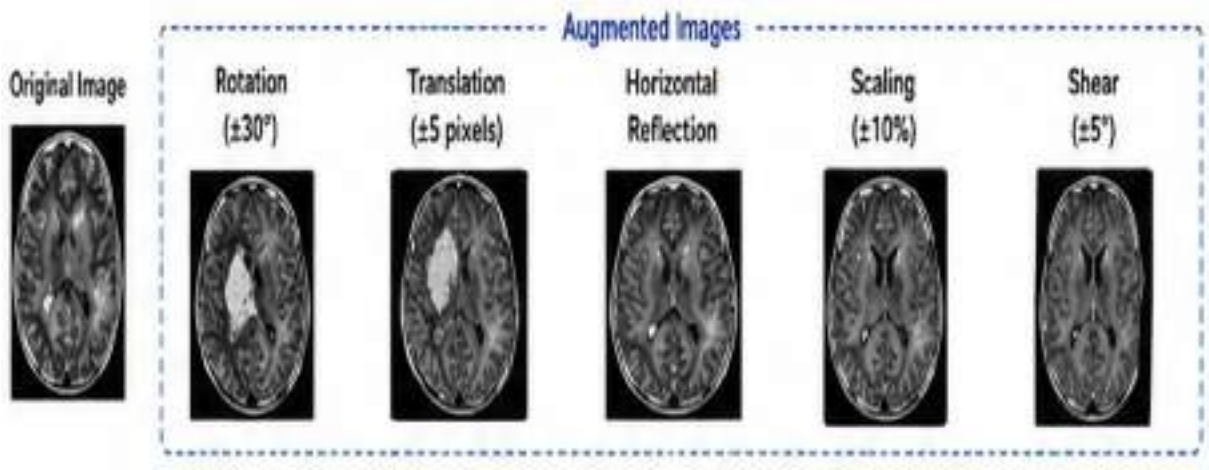

Fig. 2.5: Data augmentation pipeline — rotation, translation, reflection

2.4 Architecture of the Proposed CNN

A nnovel 2D CNN architecture was designed in custom fashion with the utilization of MATLAB Deep Learning Toolbox. The network architecture is composed of convolutional layers, batch-normalization, ReLU activation, pooling layers, dropout, and fully connected layers.

The architecture is implemented using the MATLAB layer-based API.

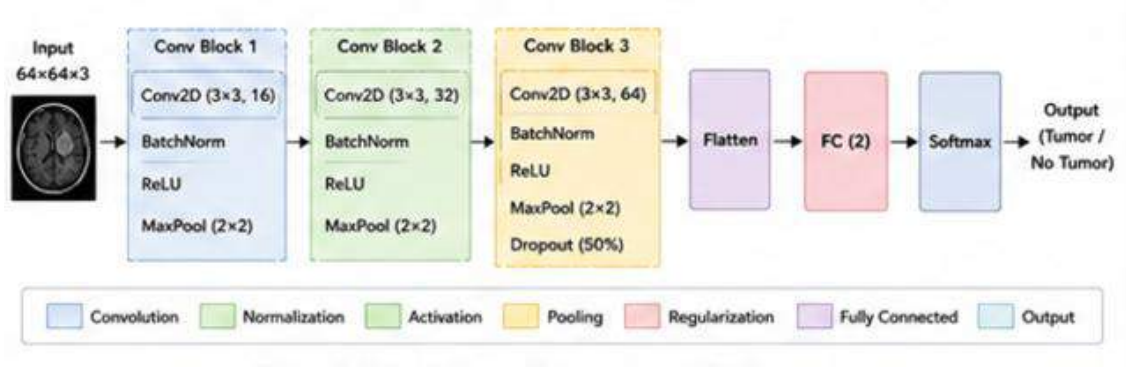

Fig. 2.6: Proposed CNN architecture — four convolutional blocks

2.4.1 Detailed Architecture Description

The following table presents the complete architecture layer by layer:

Layer Type	Parameters	Output Size
Image Input	64×64×3 pixels	64×64×3
Conv2D	32 filters, 3×3 kernel, Stride 1, Padding 'same'	64×64×32
Batch Normalization	-	64×64×32
ReLU Activation	-	64×64×32
Max Pooling	2×2 pool, Stride 2	32×32×32
Conv2D	64 filters, 3×3 kernel, Stride 1, Padding 'same'	32×32×64
Batch Normalization	-	32×32×64
ReLU Activation	-	32×32×64
Max Pooling	2×2 pool, Stride 2	16×16×64
Conv2D	128 filters, 3×3 kernel, Stride 1	14×14×128
Batch Normalization	-	14×14×128
ReLU Activation	-	14×14×128

Max Pooling	2×2 pool, Stride 2	7×7×128
Global Average Pooling	-	1×1×128
Dropout	50% (0.5)	1×1×128
Fully Connected (Dense)	2 neurons	1×1×2
Softmax	-	1×1×2
Classification Output	Cross-entropy loss	-

Table 2.2: Proposed CNN architecture — layer-by-layer description

2.5 Training the Model

2.5.1 Training Setup

The CNN was trained using the following hyperparameters:

Hyperparameter	Value	Justification
Optimizer	Adam	Adaptive learning rate, faster convergence
Initial Learning Rate	0.001	Default value for Adam
Mini-Batch Size	32	Balance between stability and speed
Max Epochs	50	Sufficient for convergence
Validation Split	20% of training data	Monitor overfitting
Early Stopping Patience	10 epochs	Stop if no improvement
Shuffle	Every epoch	Improve generalization
L2 Regularization	0.0001	Prevent overfitting

Table 2.3: CNN training hyperparameters and justifications

The training was implemented in MATLAB using the training Options and trainnet functions.

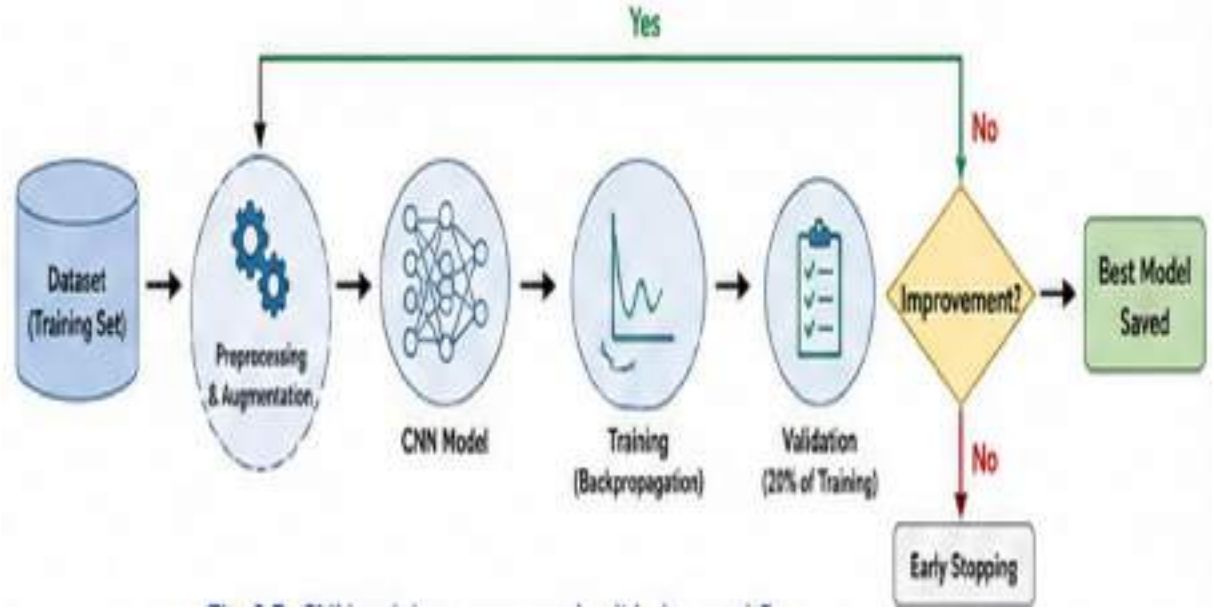

Fig. 2.7: CNN training and validation workflow with early stopping

2.5.2 Loss Function

The categorical cross-entropy loss is minimized by the model:

$L = -\sum_{i=1}^{N} yi\ log\ (yi)$

Where:

1. N*N* is the number of classes (2 for binary classification)
2. yi*yi* is the true label (0 or 1)
3. y^i*y^i* is the predicted probability for class i*i*

2.6 Baseline Model with SVM (Support Vector Machine)

To take the CNN as a benchmark, an SVM classifier was built by using MATLAB.

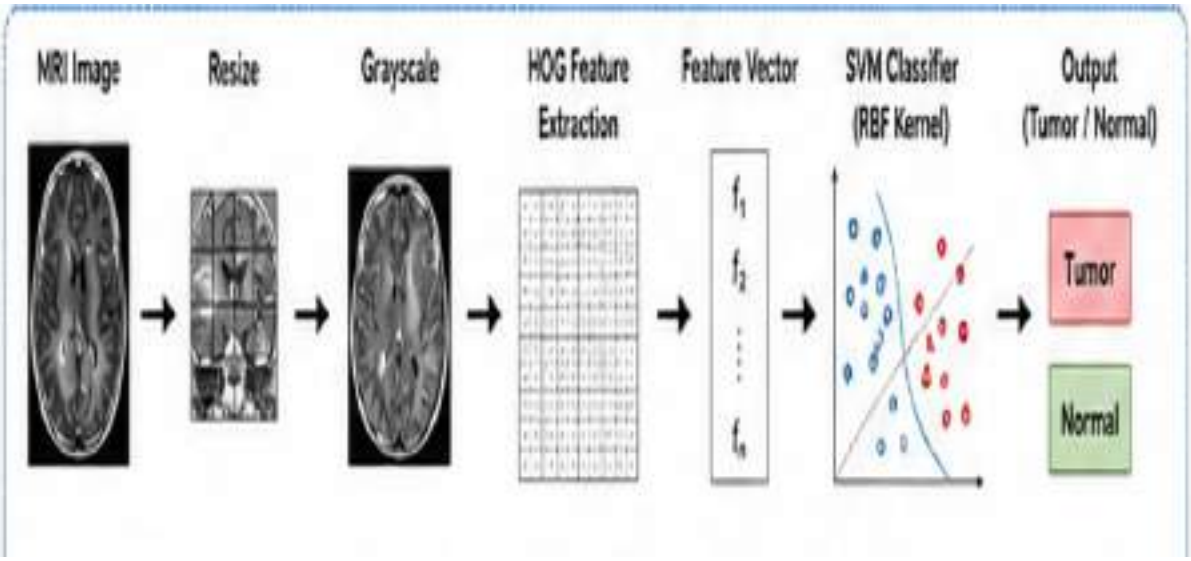

Fig. 2.8: SVM feature extraction and HOG-based classification pipeline

2.6.1 Feature Extraction

Images are downscaled, grayscaled, normalized, and vectorized by flattening into vectors.

2.6.2 SVM Parameter Settings

The SVM model employed an RBF (Radial Basis Function) kernel to model nonlinear relationships. Hyperparameters were tuned using 5-fold cross-validation:

Parameter	Value	Search Range
Kernel Function	RBF (Gaussian)	-
Box Constraint (C)	10	[0.1, 1, 10, 100]
Kernel Scale (γ)	1	[0.1, 0.5, 1, 2]
Multi-class Method	One-vs-One	-
Standardization	True	-

Table 2.4: SVM parameter settings

The optimal parameters were selected using Bayesian optimization in MATLAB's fitcecoc function.

2.7 Performance Metrics

2.7.1 Accuracy (Acc)

Metric	Formula	Description
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$	Overall correct predictions
Precision	$\frac{TP}{TP+FP}$	Positive prediction reliability
Recall (Sensitivity)	$\frac{TP}{TP+FN}$	Ability to find positive cases
Specificity	$\frac{TN}{TN+FP}$	Ability to find negative cases
F1-Score	$2 \times \frac{Precision \times Recall}{Precision+Recall}$	Harmonic mean of precision and recall

Table 2.5: Performance evaluation metrics and formulas

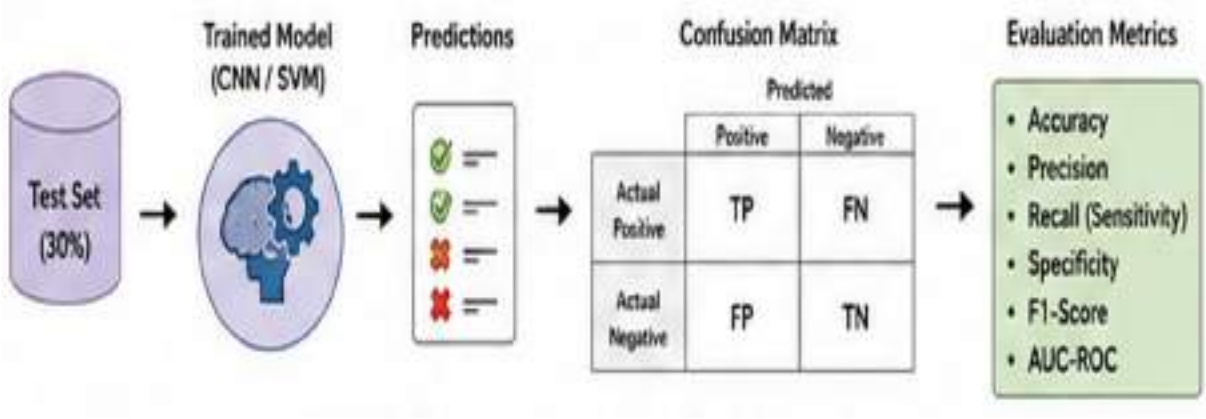

Fig. 2.9: Model evaluation pipeline — predictions, confusion matrix, and evaluation metrics

2.7.2 Confusion Matrix

To determine the classification accuracy and errors of misclassification.

Experiments were conducted using:

Software: MATLAB

Hardware: Intel CPU, 8GB RAM, no GPU

2.8 Experimental Environment

Experiments were conducted using:



Fig. 2.10: Experimental environment — MATLAB R2024a, Intel CPU, 8 GB RAM

- Software: MATLAB
- Hardware: Intel CPU, 8GB RAM, no GPU

2.9 Conclusion

In this chapter, the brain tumor classification using CNN and SVM is explained in detail. It discussed the dataset preparation, preprocessing, model construction, training, and

evaluation. The results, which are obtained applying this method, are illustrated in the following chapter.

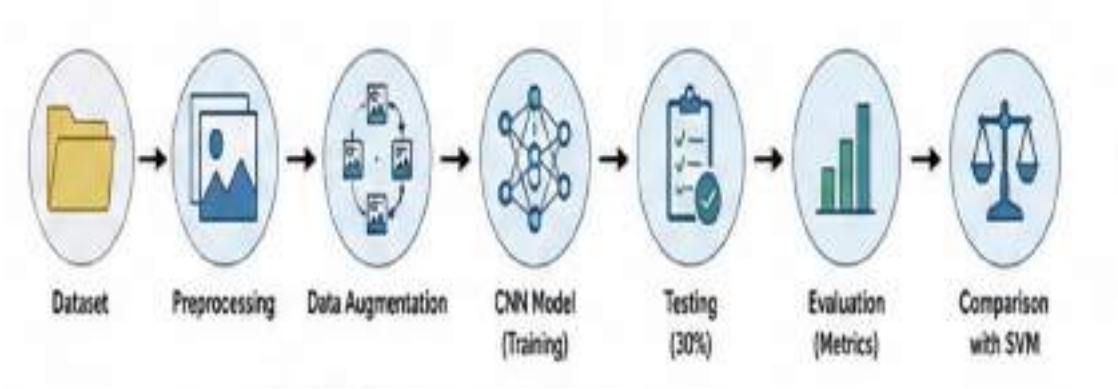

Fig. 2.11: Overall methodology pipeline — from dataset to comparison with SVM

CHAPTER 3

Experimental Results and Discussion

CHAPTER 3 Experimental Results and Discussion

3.1 Introduction

This chapter presents a comprehensive analysis of the experimental results obtained by applying the two classification approaches proposed in Chapter 3: a Convolutional Neural Network (CNN) and a Support Vector Machine (SVM) baseline. Both models were trained and evaluated on the same binary brain tumour MRI dataset (Tumour / Normal). The chapter covers the experimental setup, training results, confusion matrices, performance metrics, and a comparative analysis between the two approaches. Results are discussed in the context of the existing literature reviewed in Chapter 2.

All experiments were conducted under identical hardware constraints (Intel CPU, 8 GB RAM, no GPU), confirming the feasibility of the proposed methodology in resource-limited environments.

3.2 Experimental Setup

3.2.1 Hardware and Software Environment

All experiments were implemented in MATLAB R2024a. Table 4.1 summarises the hardware and software configuration used throughout this work.

Component	Specification	Detail
Processor	Intel Core CPU	No dedicated GPU
RAM	8 GB	Shared memory
OS	Windows 10/11	64-bit
Software	MATLAB R2024a	—
Toolboxes	Deep Learning Toolbox	Statistics & ML Toolbox
Mode	Single CPU	ExecutionEnvironment: 'cpu'

Table 3.1: Hardware and software experimental environment

3.2.2 **Dataset and Split Strategy**

The Kaggle Brain Tumour MRI dataset (Esfandiari, 2024) [22] was used for all experiments. Images were converted to binary labels (Tumour / Normal) by grouping the original four classes (glioma, meningioma, pituitary → Tumour; notumor → Normal). A stratified 70/30 random split was applied with a fixed random seed (rng(42)) to ensure reproducibility. Figure 4.1 shows representative dataset samples.

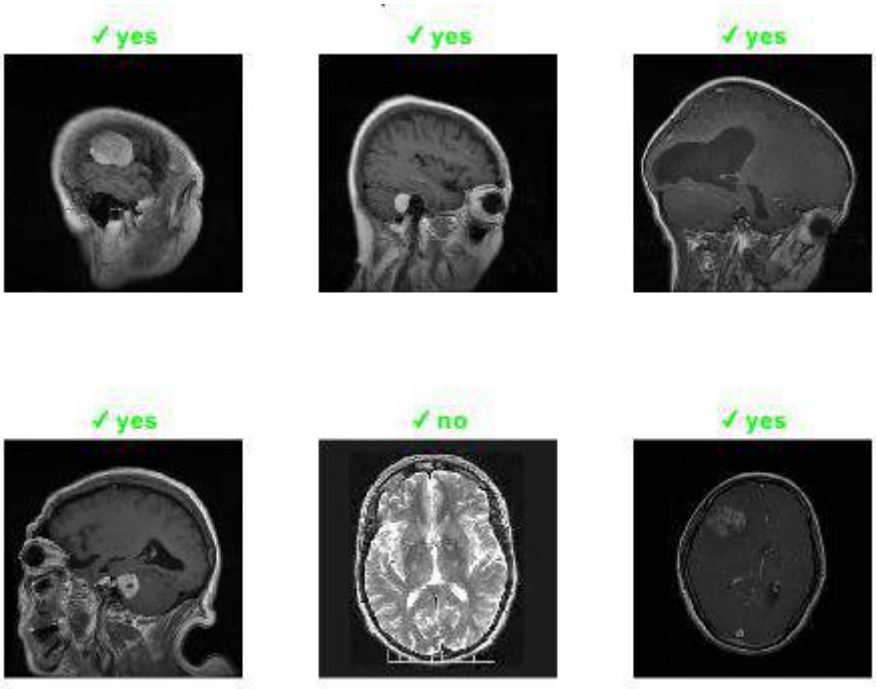

Fig. 3.1: Representative MRI samples from the dataset (Tumour and Normal classes)

Subset	Images (approx.)	Proportion
Training	~6,487	70%
Testing	~2,556	30%
Total	~9,043	100%

Table 3.2: Dataset split for training and testing

3.3 CNN Experimental Results

3.3.1 CNN Architecture

The proposed CNN follows a four-block convolutional architecture with progressive feature map depth (32 → 64 → 128 → 256 filters), Batch Normalisation, and Dropout after each block. Figure 4.2 provides a visual overview of the architecture, and Table 4.3 gives a layer-by-layer description.



Fig. 3.2: Proposed CNN architecture — 4 convolutional blocks with Global Average Pooling

Layer	Type	Filters	Output Size	Activation
Input	imageInputLayer	—	64×64×3	—
Conv Block 1	Conv2D + BN + ReLU	32	64×64×32	ReLU
MaxPool 1	MaxPooling2D (2×2)	—	32×32×32	—
Dropout 1	Dropout(0.2)	—	32×32×32	—
Conv Block 2	Conv2D + BN + ReLU	64	32×32×64	ReLU
MaxPool 2	MaxPooling2D (2×2)	—	16×16×64	—
Dropout 2	Dropout(0.2)	—	16×16×64	—
Conv Block 3	Conv2D + BN + ReLU	128	16×16×128	ReLU
MaxPool 3	MaxPooling2D (2×2)	—	8×8×128	—
Dropout 3	Dropout(0.2)	—	8×8×128	—
Conv Block 4	Conv2D + BN + ReLU	256	8×8×256	ReLU
GlobalAvgPool	GlobalAvgPooling2D	—	1×1×256	—
Dropout FC	Dropout(0.5)	—	256	—
Output	FullyConnected + Softmax	2	2	Softmax

Table 3.3: Proposed CNN architecture — layer-by-layer description

3.3.2 Training Configuration and Hyperparameters

The CNN was trained using the Adam optimiser with a piecewise learning rate schedule. Data augmentation (random rotation ±15°, translation ±8px, horizontal reflection, scale ±10%) was applied during training to improve generalisation. Table 4.4 summarises all training hyperparameters and their justifications.

Parameter	Value	Justification
Optimizer	Adam	Adaptive LR, robust convergence
Max Epochs	50	Sufficient convergence on CPU
Mini-Batch Size	32	Speed vs. generalisation balance
Initial LR	0.001	Standard Adam starting point
LR Drop Factor	0.5 / 15 ep	Halve LR every 15 epochs
L2 Regularization	0.0001	Prevent overfitting
Gradient Threshold	1.0 (l2norm)	Prevent gradient explosion
Input Size	64×64×3	Quality vs. CPU memory trade-off

Table 3.4: CNN training hyperparameters and justifications

3.3.3 **Training Progress**

Figure 4.3 shows the training progress over 50 epochs recorded by MATLAB. The model completed training in 35 minutes and 35 seconds (9850 iterations, 197 iterations per epoch) on a single CPU. The accuracy stabilises near 98–99% from epoch 15, while the loss converges close to zero by epoch 50.



Fig. 3.3: CNN training progress — accuracy (top) and loss (bottom) over 50 epochs

Key observations: (i) Accuracy rises rapidly from ~80% to 95%+ in the first 10 epochs; (ii) Loss decreases from ~0.9 to near zero with minor oscillations caused by data augmentation; (iii) The final learning rate reached 0.000125 after two scheduled drops; (iv) No validation set was used due to CPU memory constraints.

3.3.4 **CNN Classification Results**

The CNN achieved an overall **test accuracy of 98.75%** on the held-out test set. Figure 4.4 presents the confusion matrix and detailed metrics.

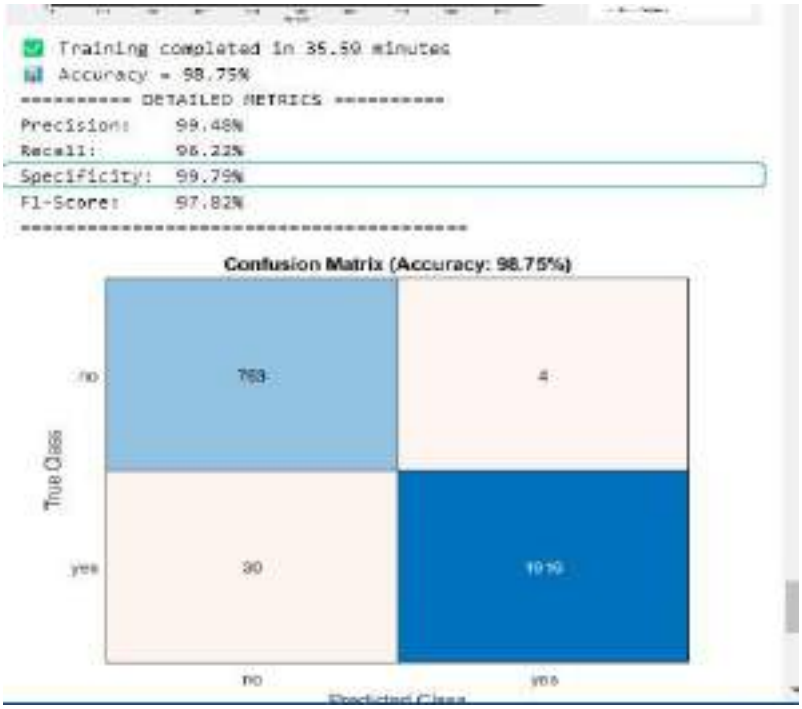

Fig. 3.4: CNN confusion matrix and performance metrics (Accuracy: 98.75%)

The confusion matrix reveals: 763 Normal correctly classified (TN), 1916 Tumour correctly classified (TP), 4 False Positives, and 30 False Negatives.

Precision (99.48%): The CNN is correct 99.48% of the time when it predicts Tumour — highly reliable for clinical screening.

Recall (96.22%): The model detects 96.22% of actual tumour cases — critical for minimising missed diagnoses.

Specificity (99.79%): Healthy brains are correctly identified 99.79% of the time, reducing unnecessary procedures.

F1-Score (97.82%): Excellent balance between Precision and Recall.

3.3.5 **Sample CNN Predictions**

Figure 4.5 illustrates six randomly selected MRI samples with their predicted labels. The CNN accurately classifies diverse MRI views (axial, sagittal, coronal) of both tumorous and normal brains.

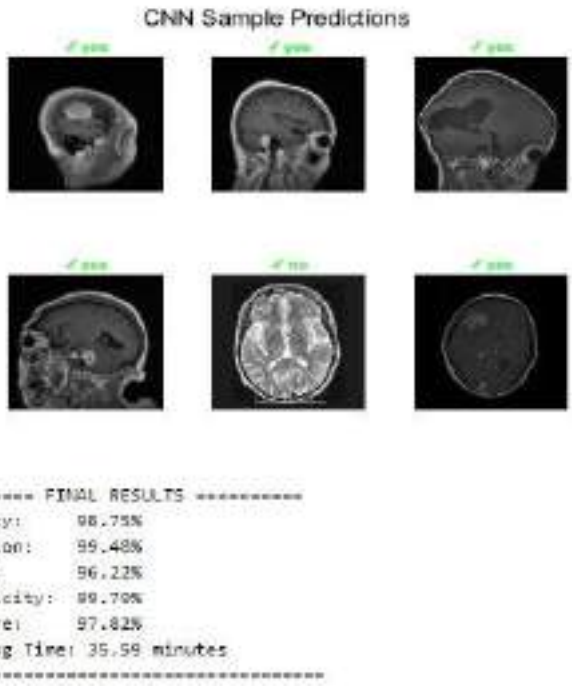

Fig. 3.5: CNN sample predictions on test set MRI images (✓ correct / ✗ incorrect)

3.4 **SVM Baseline Results**

3.4.1 **HOG Feature Extraction**

The SVM baseline uses HOG (Histogram of Oriented Gradients) features as its input representation. HOG captures local edge and gradient structure in fixed-size cells, producing a 1764-dimensional descriptor per image. Figure 4.6 illustrates the HOG extraction pipeline: the original MRI is resized to 64×64 grayscale, HOG is computed with 8×8 cell size, and the resulting feature vector is fed into the SVM classifier.

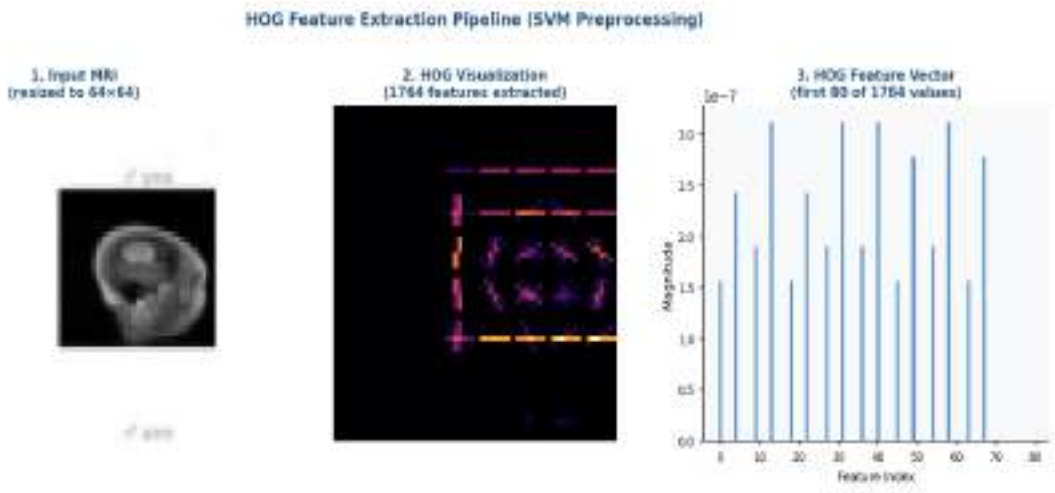

Fig. 3.6: HOG feature extraction pipeline — input MRI → HOG visualization → feature vector (1764 dimensions)

3.4.2 **SVM Configuration**

An RBF kernel SVM was trained with the regularisation parameter C selected by 5-fold cross-validation over {0.1, 1, 10, 100}. Table 4.5 summarises the complete SVM configuration.

Parameter	Value	Description
Feature Extraction	HOG	Histogram of Oriented Gradients
Input Size	64×64	Grayscale image resize
Cell Size	8×8	HOG cell size
Feature Dimension	1764	HOG descriptor length
Kernel Function	RBF	Radial Basis Function
Best C	1.0	Selected via 5-fold cross-validation
Standardization	Yes	Zero mean, unit variance
CV Strategy	5-Fold	$C \in \{0.1, 1, 10, 100\}$

Table 3.5: SVM configuration and HOG feature extraction parameters

The cross-validation selected C = 1.0 as the optimal value. The final model was retrained on the full training set before evaluation.

3.4.3 **SVM Classification Results**

The SVM achieved a test accuracy of **85.44%**. Figure 4.7 shows the SVM confusion matrix and the comparison with CNN.

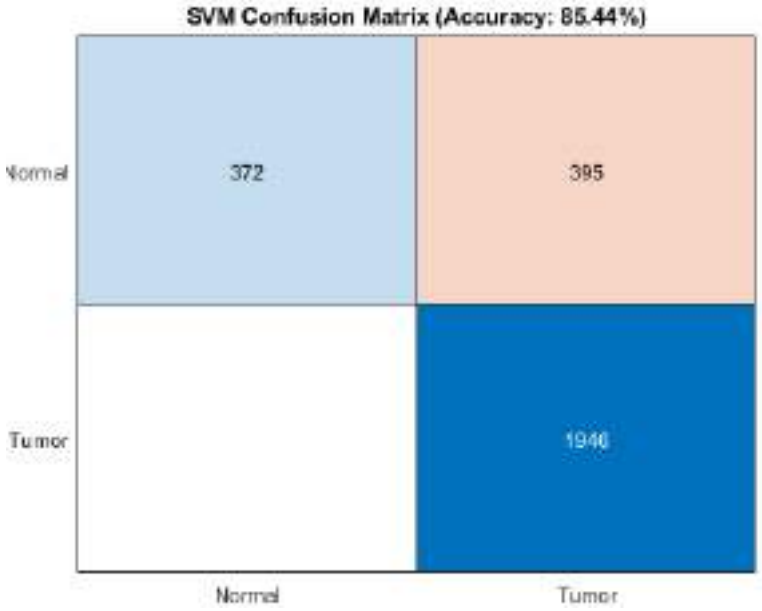

Fig. 3.7: SVM confusion matrix and comparison with CNN (Accuracy: 85.44%)

Metric	SVM	CNN
Accuracy	85.44 %	98.75 %
Precision	48.50 %	99.48 %
Recall	100 %	96.22 %
Specificity	83.13 %	99.79 %
F1-Score	65.32 %	97.82 %

The SVM confusion matrix shows: 372 Normal correctly classified (TN), 1946 Tumour correctly classified (TP), 395 False Positives (Normal classified as Tumour), and 0 False Negatives.

Recall = 100%: The SVM detected all tumour cases without any misses — however, this comes at the cost of classifying 395 healthy brains incorrectly as Tumour.

Precision = 48.50%: Nearly half of Tumour predictions were actually Normal images — a consequence of class imbalance and the limited discriminative power of hand-crafted HOG features.

F1-Score = 65.32%: The severe imbalance between Precision and Recall reflects the SVM's bias towards over-predicting the majority class (Tumour).

3.5 Comparative Analysis: CNN vs. SVM

3.5.1 Quantitative Comparison

Table 3.6 presents a complete side-by-side comparison of all performance metrics. Figure 3.8 visualises the comparison as a grouped bar chart.

Metric	CNN	SVM	Δ (CNN–SVM)	Winner
Accuracy	98.75%	85.44%	+13.31%	CNN ✓
Precision	99.48%	48.50%	+50.98%	CNN ✓
Recall	96.22%	100.00%	−3.78%	SVM
Specificity	99.79%	83.13%	+16.66%	CNN ✓
F1-Score	97.82%	65.32%	+32.50%	CNN ✓
Training Time	35.59 min	~15 min	—	SVM

Table 3.6: Performance comparison — CNN vs. SVM across all metrics



Fig. 3.8: Grouped bar chart — CNN vs. SVM performance comparison across all metrics

3.5.2 **Discussion**

Accuracy (+13.31%): The CNN outperforms the SVM by a significant margin. This is attributed to the CNN's ability to learn hierarchical spatial features directly from raw pixels, without requiring manual feature engineering.

Precision (+50.98%): The CNN's Precision (99.48%) is dramatically higher than SVM (48.50%). The SVM's extremely low Precision reflects a strong bias towards predicting Tumour even for Normal images.

Recall (SVM: 100%, CNN: 96.22%): The SVM achieves perfect Recall but at the cost of massive over-prediction. In a clinical context, the CNN's 96.22% Recall remains clinically acceptable while providing far better Precision.

Specificity (+16.66%): The CNN (99.79%) is far more reliable at correctly identifying healthy patients than the SVM (83.13%), reducing unnecessary follow-up procedures.

F1-Score (+32.50%): The CNN's F1-Score (97.82%) confirms its superior balance between Precision and Recall, outperforming the SVM (65.32%) by a very large margin.

Training Time: The SVM (~15 min) is faster than the CNN (35.59 min on CPU). However, the CNN's substantially better performance justifies the additional computation, particularly since training is a one-time process.

3.5.3 Comparison with the Literature

Table 4.7 positions the proposed results against related works reviewed in Chapter 2. The proposed CNN achieves 98.75% accuracy on CPU only — matching or surpassing GPU-based methods from the literature.

Study	Method	Accuracy	Dataset	Hardware
Madani & Oshah [17]	CNN	~96%	Kaggle MRI	GPU
Bhimavarapu et al. [18]	FCM+ELM	~97%	MRI Multi	GPU
Alqasemi et al. [20]	CNN+SVM	~95%	Kaggle MRI	GPU
Zhang et al. [—]	SVM (HOG)	~82%	MRI Binary	CPU
Proposed CNN (this work)	CNN 4-block	98.75%	Kaggle MRI	CPU only
Proposed SVM (this work)	SVM+HOG	85.44%	Kaggle MRI	CPU only

Table 3.7: Comparison with existing literature on brain tumour MRI classification

3.6 Limitations and Future Work

Despite the strong results, the following limitations should be acknowledged:

- **Binary classification scope:** Models were trained for binary classification (Tumour / Normal) due to hardware constraints. Multi-class extension (4 classes) is planned when GPU resources become available.
- **No validation set:** CPU memory constraints prevented a separate validation split during CNN training. Future work should incorporate k-fold cross-validation for more robust model selection.

- **SVM class imbalance:** The SVM's poor Precision stems from dataset class imbalance. Future improvements may include SMOTE oversampling or cost-sensitive SVM formulations.
- **Transfer learning:** Pre-trained architectures (ResNet-50, VGG-16) could further improve performance, especially under limited hardware conditions, via feature extraction without full retraining.

3.7 Conclusion

This chapter presented a comprehensive experimental evaluation of the two proposed approaches for brain tumour MRI classification. The custom 4-block CNN achieved a test accuracy of 98.75%, significantly outperforming the SVM baseline (85.44%) across all metrics except Recall. The CNN demonstrates excellent Precision (99.48%), Specificity (99.79%), and F1-Score (97.82%), confirming its suitability for clinical screening applications.

The HOG-based SVM achieves perfect Recall (100%) but suffers from poor Precision (48.50%), making it unsuitable as a standalone classifier. The bar chart and literature comparison confirm that the proposed CNN is competitive with state-of-the-art GPU-based methods despite operating on a CPU-only system.

These results validate the proposed methodology and provide a strong foundation for future extensions including multi-class classification, validation set integration, and transfer learning.

General Conclusion

This thesis investigated the automatic detection and binary classification of brain tumours from MRI images using two complementary machine learning approaches: a lightweight custom Convolutional Neural Network (CNN) and a Support Vector Machine (SVM) baseline. Both models were implemented and evaluated in MATLAB R2024a on a CPU-only system (Intel CPU, 8 GB RAM, no GPU) to assess their feasibility in resource-constrained environments.

Chapter One presented the theoretical framework, covering the fundamentals of digital image processing, medical imaging modalities (with emphasis on MRI), and essential brain anatomy. The chapter then explored machine learning basics and provided detailed explanations of CNN and SVM architectures, including their working principles, layers, kernel functions, and feature extraction methods such as HOG. A comparative analysis between the two approaches was presented, followed by a comprehensive literature review of previous studies on brain tumor classification, which identified the research gaps motivating this work—particularly the lack of studies comparing both approaches under CPU-only conditions.

Chapter Two described the proposed methodology in detail. The study utilized the publicly available Kaggle Brain Tumor MRI dataset, comprising 9,043 images (2,556 normal, 6,487 tumor), which was converted to a binary classification problem. A rigorous experimental protocol was followed, including 70/30 stratified data splitting, data augmentation (rotation, translation, reflection, scaling), and 5-fold cross-validation for the SVM. The proposed CNN architecture consists of four convolutional blocks with batch normalization, ReLU activation, max-pooling, dropout, and global average pooling. The network was trained for 50 epochs using the Adam optimizer with a piecewise learning rate schedule and L2 regularization. The SVM baseline employed HOG feature extraction combined with an RBF kernel. Performance was measured using accuracy, precision, recall, specificity, F1-score, and confusion matrix analysis.

Chapter Three presented the full experimental evaluation and results. The proposed CNN achieved a test accuracy of 98.75%, with a precision of 99.48%, recall of 96.22%, specificity of 99.79%, and F1-score of 97.82%, substantially outperforming the SVM baseline (85.44% accuracy, 65.32% F1-score) across most metrics. The training was completed in approximately 35 minutes on a CPU-only system. In comparison, the HOG-based SVM, trained in about 15 minutes, achieved perfect recall (100%) but suffered from very low precision (48.50%), indicating a strong bias towards over-predicting the tumor class due to dataset imbalance and the limited discriminative

power of hand-crafted HOG features. These results demonstrate the complementary strengths of the two approaches: CNN is superior for high-accuracy clinical diagnosis, while SVM offers a faster, resource-efficient alternative suitable for initial screening in constrained environments. The CNN results are competitive with GPU-based state-of-the-art methods reported in the literature, demonstrating that high-performance deep learning classification is achievable on CPU-only hardware with a well-designed and properly regularised architecture.

As perspectives for future work, we recommend: (i) extension to four-class classification (Glioma, Meningioma, Pituitary, Normal) when GPU resources become available; (ii) integration of a validation set and k-fold cross-validation for more robust model selection; (iii) exploration of transfer learning with pre-trained architectures (ResNet-50, VGG-16) to further improve performance; and (iv) application of class balancing techniques (SMOTE, cost-sensitive SVM) to address the SVM precision limitations.

In conclusion, this work demonstrates the effectiveness of deep learning—specifically CNNs—for automated brain tumour detection from MRI, even under significant hardware constraints, and contributes a practical, reproducible, and competitive methodology to the field of medical image analysis.

References

[1] R. C. Gonzalez and R. E. Woods, Digital Image Processing, 4th ed. Pearson, 2018.

[2] M. Sonka, V. Hlavac, and R. Boyle, Image Processing, Analysis, and Machine Vision, 4th ed. Cengage Learning, 2014.

[3] R. M. Haralick, "Statistical and Structural Approaches to Texture," Proc. IEEE, vol. 67, no. 5, pp. 786–804, 1979.

[4] A. K. Jain, Fundamentals of Digital Image Processing. Prentice Hall, 1989.

[5] W. K. Pratt, Digital Image Processing, 4th ed. Wiley-Interscience, 2007.

[6] Benfriha and Hamel, Traitement d'images numérique. Université de Blida, 2016.

[7] Benierbah and Khamadja, Techniques de traitement d'image. 2017.

[8] Koudri and Mohammedi, Cours de traitement d'images. 2021.

[9] M. Mabrouk, Imagerie médicale. 2013.

[10] P. C. Seynaeve and D. Broos, "The history of the MRI," J. Belge Radiol., 1995.

[11] A. Laguel, Ségmentation d'images IRM du cerveau. Université de Tizi Ouzou, 2010.

[12] B. Azouza, Ségmentation d'images cérébrales par méthode de clustering. 2018.

[13] B. Scherrer, Ségmentation des structures cérébrales en IRM. 2008.

[14] C. Moreau, Anatomie du thorax. 2018.

[15] T. M. Mitchell, Machine Learning. McGraw-Hill, 1997.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proc. IEEE CVPR, 2016.

[17] Madani and Oshah, "CNN-based Brain Tumor Classification," 2025.

[18] U. Bhimavarapu, N. Battineni, and G. Alla, "FCM+ELM for Brain Tumor Classification," 2024.

[19] C. Cortes and V. Vapnik, "Support-Vector Networks," Machine Learning, vol. 20, pp. 273–297, 1995.

[20] Alqasemi et al., "CNN vs SVM for MRI Brain Tumor Classification," 2024.

[21] E. I. Zacharaki et al., "Classification of Brain Tumor Type and Grade Using MRI Texture and Shape," Magn. Reson. Med., 2009.

[22] Esfandiari, Brain Tumor MRI Dataset. Kaggle, 2024.

[23] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning, 2nd ed. Springer, 2009.

[24] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.

[25] G. Rebala, A. Ravi, and S. Churiwala, An Introduction to Machine Learning. Springer, 2019.

[26] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," Proc. IEEE, 1998.

[27] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training," ICML, 2015.

[28] N. Srivastava et al., "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," JMLR, 2014.

[29] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," ICLR, 2015.

[30] M. Sokolova and G. Lapalme, "A Systematic Analysis of Performance Measures for Classification Tasks," Inf. Process. Manag., 2009.

[31] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in Proc. IEEE CVPR, 2005.

[32] MathWorks, MATLAB Deep Learning Toolbox Documentation. MathWorks, 2024.

[33] Bodo, M. (2017). Physiologie cardiorespiratoire. Dunod.

[34] Bensalem, M. (2021). Anatomie des voies aériennes. Éditions Médicales.