AI 410
Computer Vision
Group 5

Marcelino Magdy	200013236
Lobna Hegazy	200013237
Mario Ayad	200013216
Mohamed Nasr	200018186
Mostafa Nasr	200017717

Final Report: Brain Tumor MRI Classification Using GAN-Augmented Data and Deep Learning

Introduction

Medical imaging, particularly **Magnetic Resonance Imaging (MRI)**, is a cornerstone in diagnosing **brain tumors**. However, developing robust deep learning models for tumor detection faces a critical challenge: **limited availability of high-quality, annotated datasets**. The original study by **Safdar et al. (2020)** explored traditional data augmentation techniques (e.g., rotation, flipping, noise injection) but was constrained by the **scarcity of diverse tumor samples**.

To overcome this limitation, our project:

1. **Synthesized realistic brain tumor MRI scans** using **Generative Adversarial Networks (GANs)** to expand dataset diversity.
2. **Applied classification models** to improve tumor localization and diagnostic accuracy.

This report details our **methodology, implementation, results, and performance analysis**, demonstrating significant improvements over traditional augmentation approaches.

Problem Description

1. Challenges in Brain Tumor MRI Analysis

- **Small datasets**: Original datasets (e.g., 155 tumor images, 98 non-tumor images) are insufficient for training deep learning models effectively.
- **Limited diversity**: Traditional augmentation (e.g., flipping, rotation) does not generate **structurally diverse** tumor representations.
- **Class imbalance**: Non-tumor cases are underrepresented, leading to biased models.

2. Our Solution

We addressed these challenges by:

- **Generating synthetic MRI scans** using **Deep Convolutional GANs (DCGANs).**
- **Augmenting the dataset** with **4,000 synthetic images (2,000 tumor + 2,000 non-tumor)**.
- **Training a CNN classifier** on the augmented dataset for high-accuracy tumor detection.

Methodology

1. GAN-Based Data Augmentation

We implemented a **DCGAN** to generate synthetic MRI images:

Generator Architecture

- **Input**: 100-dimensional noise vector.
- **Layers**:
  - `Dense(32*32*256) → LeakyReLU → Reshape(32,32,256)`
  - `Conv2DTranspose(128, strides=2) → LeakyReLU`
  - `Conv2DTranspose(128, strides=2) → LeakyReLU`
  - `Conv2D(1, activation='tanh')` (Output: 128×128 grayscale MRI)

Discriminator Architecture

- **Input**: 128×128 MRI image.
- **Layers**:
  - `Conv2D(64 → 128 → 128 → 256, strides=2) → LeakyReLU`
  - `Flatten → Dropout(0.4) → Dense(1, 'sigmoid')`

Training Process

- **Loss**: Binary cross-entropy.
- **Optimizer**: Adam (learning rate = 0.0002).
- **Epochs**: 10 (1,750 steps/epoch).
- **Batch Size:** 4.

2. Traditional Data Augmentation

We further enhanced the dataset using:

- **Horizontal/Vertical flipping**
- **90° and 180° rotation**
- **Total Augmented Image**s: **16,000** (from 4,000 originals).

3. CNN Classification Model

We trained a **deep CNN** for tumor classification:

Architecture

- **Convolutional Blocks**:
  - `Conv2D(32 → 64 → 128) + BatchNorm + MaxPooling + Dropout`
- **Fully Connected Layers**:
  - `Flatten → Dense(256) → Dropout(0.5) → Sigmoid`

Training Parameters

- **Optimizer**: Adam (LR = 0.0001).
- **Batch Size:** 32**.**
- **Epochs**: 20.
- **Callbacks**: Early stopping, model checkpointing, learning rate reduction.

Results & Performance Analysis

1. GAN Performance

- **Generated Images**: 2,000 tumor + 2,000 non-tumor scans.
- **Visual Quality**: Realistic tumor structures.
- **Distribution Matching**: Synthetic images closely followed real data distribution (Kernel Density Estimation plots).

2. Classification Performance

Metric	Training	Validation	Test
Accuracy	99.14%	83.18%	99.10%
Precision	99.21%	100%	100%
Recall	99.07%	66.43%	98.20%
AUC-ROC	99.97%	98.27%	100%

Confusion Matrix (Test Set)

	Predicted No Tumor	Predicted Tumor
Actual No Tumor	1000	0
Actual Tumor	18	982

- **F1-Score: 0.99 (balanced performance).**
- **ROC-AUC: 1.0 (perfect separability).**

3. Key Findings

1. **GANs effectively expanded the dataset**, improving model generalization.

2. **CNN achieved near-perfect test accuracy (99.1%)**, demonstrating robustness.

3. **Precision** = **100%** means **no false positives** in tumor detection.

Conclusion & Future Work

1. Summary

Our approach successfully addressed the **data scarcity problem** in brain tumor MRI analysis by:

1. **Generating high-quality synthetic images** using GANs.

2. **Augmenting the dataset** with traditional techniques.

3. **Training a highly accurate CNN classifier** (99.1% test accuracy).

References

1. Safdar, M. F., Alkobaisi, S. S., & Zahra, F. T. (2020). A Comparative Analysis of Data Augmentation Approaches for Magnetic Resonance Imaging (MRI) Scan Images of Brain Tumor. *Acta Informatica Medica, 28*(1), 29–36. [DOI: 10.5455/aim.2020.28.29-36](https://doi.org/10.5455/aim.2020.28.29-36)

2. Goodfellow, I., et al. (2014). Generative Adversarial Networks. *NeurIPS*.

3. Ronneberger, O., et al. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *MICCAI*.

For the 2nd Try with VGG16

We trained a **transfer learning CNN** for brain tumor classification:

Architecture

- **Base Model**:
  - Pre-trained VGG16 (ImageNet weights)
  - Frozen layers (non-trainable)
- **Input Adaptation**:
  - Lambda layer to convert grayscale to RGB (1→3 channels)
- **Classification Head**:
  - Flatten → Dense(512) → BatchNorm → Dropout(0.5) → Dense(256) → BatchNorm → Dropout(0.5) → Sigmoid

Training Parameters

- **Optimizer**: Adam (LR = 0.0001)
- **Loss Function**: Binary Cross-Entropy
- **Metrics**: Accuracy, Precision, Recall, AUC

Results & Performance Analysis

Evaluating the model with test data

```
Test Loss: 1.3270
Test Accuracy: 0.6863
Test Precision: 0.6744
Test Recall: 0.9355
Test AUC: 0.7758
```

Classification Report

```
Classification Report:
              precision    recall  f1-score   support

    No Tumor       0.75      0.30      0.43        20
       Tumor       0.67      0.94      0.78        31

    accuracy                           0.69        51
   macro avg       0.71      0.62      0.61        51
weighted avg       0.70      0.69      0.64        51
```

Confusion Matrix