**AI-powered medical diagnosis system: Brain Tumor Detection**

A Project Report

submitted in partial fulfillment of the requirements

of

AICTE Internship on AI: Transformative Learning

with

TechSaksham – A joint CSR initiative of Microsoft & SAP

by

**Shreyas Meru, shreyasmeru@gmail.com**

Under the Guidance of

**Saomya Chaudhury**

**ACKNOWLEDGEMENT**

We would like to take this opportunity to express our deep sense of gratitude to all individuals who helped us directly or indirectly during this thesis work.

Firstly, we would like to thank my supervisor, Saomya Chaudhury, for being a great mentor and the best adviser I could ever have. His advice, encouragement and the critics are a source of innovative ideas, inspiration and causes behind the successful completion of this project. The confidence shown in me by him was the biggest source of inspiration for me. It has been a privilege working with him for the last one year. He always helped me during my project and many other aspects related to the program. His talks and lessons not only help in project work and other activities of the program but also make me a good and responsible professional.

#### **ABSTRACT**

**Problem Statement**

Brain tumors are life-threatening conditions that require early and accurate detection for effective treatment. Traditional diagnosis methods, such as MRI scans and manual radiological assessments, are time-consuming and prone to human error. Automating the detection process using deep learning can enhance accuracy and speed in tumor identification.

**Objectives**

This project aims to develop a deep learning-based system for detecting brain tumors in medical images using the YOLO (You Only Look Once) object detection model. The system provides an easy-to-use **Streamlit web interface** for users to upload images or provide image URLs for tumor detection.

**Methodology**

1. **Data Collection & Preprocessing:** A dataset of brain tumor images was used for training. Images were annotated with bounding boxes for tumor regions.
2. **Model Training:** The YOLO model was trained on the dataset to classify and localize brain tumors with high accuracy.
3. **Implementation:** The trained YOLO model was integrated into a Streamlit application, enabling real-time tumor detection. Users can upload images or enter URLs, and the model generates annotated images highlighting potential tumor regions.
4. **Output Interpretation:** The system displays detected tumors, confidence scores, and provides clinical guidance for further action.

**Key Results**

* The YOLO model successfully detected brain tumors with high accuracy.
* The **Streamlit interface** ensures accessibility and ease of use for medical professionals and researchers.
* The detection results are displayed with bounding boxes, confidence scores, and detailed explanations to assist in clinical decision-making.

**Conclusion**

The project demonstrates the effectiveness of deep learning in medical imaging for **automated brain tumor detection**. The developed system can assist radiologists by providing a **quick, preliminary diagnosis**, reducing manual workload, and enhancing diagnostic accuracy. Future work includes improving model robustness with larger datasets and integrating additional imaging modalities like MRI scans.

Top of Form

Bottom of Form

**TABLE OF CONTENT**

**Abstract I**

**Chapter 1.**  **Introduction 1**

1.1 Problem Statement 1

1.2 Motivation 1

1.3 Objectives 2

1.4 Scope of the Project 2

**Chapter 2.**  **Literature Survey 3**

2.1 Review of Relevant Literature 3

2.2 Existing Models and Techniques 4

2.3 Gaps and Limitations in Existing Solutions 4

**Chapter 3.**  **Proposed Methodology 5**

3.1 System Design 5

3.2 Requirement Specification 6

**Chapter 4.**  **Implementation and Results 7**

4.1 Snap Shots of Result 7

4.2 GitHub Link for Code 9

**Chapter 5. Discussion and Conclusion 10**

5.1 Future Work 10

5.2 Conclusion 11

**References** 12

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure Caption** | **Page No.** |
| **3.1.1** | System Design | **5** |
| **4.1.1** | Streamlit Application Interface | **7** |
| **4.1.2** | Uploading an Image for Detection | **8** |
| **4.1.3** | Tumor Detection Output | **8** |
| **4.1.4** | Detection Results Summary | **9** |

**CHAPTER 1**

**Introduction**

Brain tumors are a serious medical condition that can be life-threatening if not detected early. Traditional diagnostic methods, such as MRI scans and manual assessments by radiologists, are often time-consuming and prone to human error. Delayed detection can lead to reduced survival rates and limited treatment options.

With advancements in deep learning, automated tumor detection using machine learning models has emerged as a promising approach. This project leverages the **YOLO (You Only Look Once)** object detection algorithm to identify brain tumors in medical images efficiently. By integrating the trained model into a **Streamlit-based web application**, the system provides a user-friendly platform where users can upload medical images for real-time tumor detection.

The project aims to assist radiologists by offering **fast, preliminary assessments**, reducing diagnostic time and enhancing accuracy. Potential applications include **telemedicine, AI-assisted medical imaging, and remote diagnostics**, making it valuable for regions with limited medical expertise.

While this system serves as a **preliminary screening tool**, it does not replace clinical diagnosis. Its accuracy depends on the quality of training data and images provided. However, it lays the groundwork for future enhancements, contributing to AI-driven **medical imaging and early detection systems**.

* 1. **Problem Statement:**

Brain tumors pose a significant health risk, affecting millions of people worldwide. Traditional diagnostic methods, such as MRI scans and manual assessment by radiologists, are often time-consuming and subject to human error. The lack of early detection can lead to delayed treatment, reducing survival rates. Automating the tumor detection process using **deep learning-based object detection** can help identify abnormalities efficiently and accurately, aiding medical professionals in early diagnosis and intervention.

* 1. **Motivation:**

This project was chosen to address the need for **faster and more accurate** brain tumor detection methods. By utilizing **YOLO**, a real-time object detection algorithm, the system can provide quick assessments, reducing the burden on radiologists and improving diagnostic accuracy. The potential applications of this system include:

* Assisting radiologists in detecting tumors with high confidence.
* Providing a **preliminary screening tool** in medical facilities with limited expertise.
* Enabling **telemedicine and remote diagnostics**, allowing patients to receive quick feedback.
* Enhancing research in AI-driven medical imaging and diagnosis.
  1. **Objective:**

The primary objectives of this project are:

* To **develop a deep learning-based model** for detecting brain tumors in medical images.
* To train and fine-tune the **YOLO algorithm** for high-accuracy detection.
* To build a **user-friendly web interface** using **Streamlit** for easy image uploads and result visualization.
* To provide confidence scores and **detailed explanations** to assist medical professionals in decision-making.
* To ensure the model achieves a **high detection accuracy** to minimize false positives and negatives.
  1. **Scope of the Project:**

The project is designed as a **proof-of-concept** to demonstrate the feasibility of **automated brain tumor detection** using deep learning. However, it has certain limitations:

* The model is trained on **limited datasets** and may require further enhancement for **real-world clinical use**.
* The system focuses on **image-based detection** and does not replace **advanced medical examinations** such as MRI or biopsy.
* The detection accuracy depends on **image quality** and **model training data**.
* The current system is designed for **preliminary screening** and requires further validation before **clinical implementation**.

**CHAPTER 2**

**Literature Survey**

The field of **automated brain tumor detection** has been extensively studied in medical imaging and deep learning. Machine learning techniques, particularly **Convolutional Neural Networks (CNNs)**, have shown great potential in tumor classification and segmentation. Architectures like **ResNet, VGGNet, and U-Net** have been widely used for analyzing brain scans. However, most of these models primarily focus on **tumor classification** rather than **precise localization**, requiring additional segmentation methods for detailed analysis.

This project builds on these advancements by leveraging **YOLO (You Only Look Once)**, a state-of-the-art real-time object detection model. Unlike conventional CNNs, which perform **patch-based analysis**, YOLO detects tumors in a **single pass**, making it more efficient for real-time applications. By integrating YOLO into a **Streamlit-based web interface**, this project enhances accessibility for users, allowing quick and interactive tumor detection. This approach provides an **end-to-end solution** for both **localization and classification**, improving diagnostic accuracy while reducing processing time in medical imaging.

* 1. **Review of Relevant Literature**

Several studies have demonstrated the effectiveness of **deep learning in brain tumor detection**. CNN-based models have been widely used for **tumor classification**, achieving high accuracy in differentiating between benign and malignant tumors. **Ghaffari et al. (2020)** implemented a **U-Net model** for tumor segmentation, effectively detecting tumor boundaries. Similarly, **Cheng et al. (2017)** explored **Support Vector Machines (SVMs)** for tumor classification, showing promising results but requiring extensive feature extraction.

While these methods perform well in **tumor classification**, they often struggle with **precise localization**, requiring additional segmentation techniques. This project addresses these limitations by utilizing **YOLO (You Only Look Once)**, which detects tumors in a **single pass**, enabling **real-time localization and classification**. By integrating this model into a **Streamlit-based interface**, the project enhances accessibility, providing a **fast, interactive, and efficient** solution for preliminary tumor detection in medical imaging.

* 1. **Existing Models and Techniques**

Traditional approaches for tumor detection include **thresholding, edge detection, and morphological operations** applied to MRI images. However, these methods are highly sensitive to noise and variations in image quality. **Machine learning models**, such as Random Forests and SVMs, have been used to classify tumor images based on extracted features, but they require **manual feature engineering**, making them less efficient for large-scale applications.

Deep learning models such as **CNNs, U-Net, and ResNet** have improved performance by learning hierarchical features directly from images. The **YOLO (You Only Look Once) model** stands out as a real-time object detection method, enabling both **classification and localization** of tumors in a single step. This makes YOLO an efficient alternative to traditional CNNs, which require **region-based processing (e.g., Faster R-CNN)**, leading to slower inference times.

* 1. **Limitations in Existing Solutions**

Despite advancements in brain tumor detection, existing models face several limitations:

1. **Lack of Real-Time Detection:** Many models focus on classification rather than fast, real-time detection, making them less suitable for immediate analysis.
2. **High Computational Costs:** Techniques such as **Faster R-CNN and U-Net** require significant computational resources, limiting their use in real-time applications.
3. **Limited Generalization:** Models trained on small datasets may not perform well on new, unseen data due to **overfitting**.
4. **Manual Feature Engineering:** Traditional machine learning methods depend on **manual feature extraction**, making them labor-intensive and less adaptable.

**CHAPTER 3**

**Proposed Methodology**

The proposed system utilizes **YOLO (You Only Look Once)**, a state-of-the-art deep learning-based object detection model, to efficiently detect brain tumors in medical images. Unlike traditional machine learning models that require **manual feature extraction and classification**, YOLO performs **end-to-end detection**, allowing simultaneous **localization and classification** of tumors in a single forward pass. This capability makes it highly suitable for real-time applications, ensuring **fast and accurate diagnosis**.

The system is deployed as a **Streamlit-based web application**, providing an intuitive and interactive interface where users can upload brain scans or provide image URLs for analysis. The primary goal is to make the detection process **accessible to medical professionals, researchers, and even non-experts**, allowing them to quickly analyze brain scans without the need for specialized software. The web interface displays both the **original and detected images**, along with tumor confidence scores and detection explanations.

The methodology consists of **three main phases**:

1. **Data Preprocessing:** The input images are resized, normalized, and converted into a format compatible with YOLO for efficient processing.
2. **Model Training:** The YOLO model is trained on a **brain tumor dataset** to recognize tumor regions with high accuracy.
3. **Deployment & Inference:** The trained model is integrated into the Streamlit application to enable real-time detection and visualization.

This system ensures **speed, accuracy, and accessibility**, making it an effective **AI-powered medical diagnostic tool** for early tumor detection, potentially aiding in timely treatment and improved patient outcomes.

* 1. **System Design**

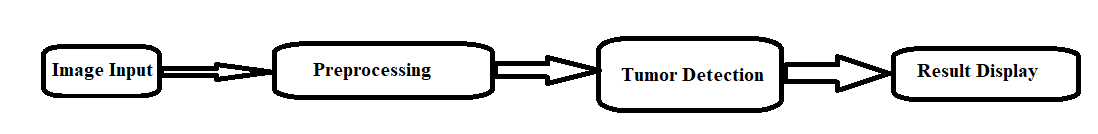


Fig. 3.1.1 System Design

The system follows a structured pipeline:

1. **Image Input:** Users upload a brain scan or provide an image URL via the Streamlit interface.
2. **Preprocessing:** The image is resized and normalized for compatibility with the YOLO model.
3. **Tumor Detection:** The trained YOLO model detects tumor regions, assigns confidence scores, and marks the detected areas with bounding boxes.
4. **Result Display:** The processed image, along with detection confidence and recommendations, is displayed in the web interface.
   1. **Requirement Specification**

To develop and deploy the brain tumor detection system, several **tools and technologies** are required. These include **deep learning frameworks, web development tools, and image processing libraries** to ensure accurate and efficient detection.

* **Deep Learning Framework:** The project uses **YOLO (You Only Look Once)**, implemented with **Ultralytics YOLOv8** in **PyTorch** for high-speed and accurate tumor detection.
* **Web Application Framework:** **Streamlit** is used to build an interactive and user-friendly interface, allowing users to upload and analyze brain scans.
* **Image Processing Tools:** Libraries such as **OpenCV** and **PIL (Pillow)** handle image loading, resizing, and preprocessing before model inference.
* **Development Environment:** Model training is performed using **Google Colab** and **Jupyter Notebook**, leveraging GPU acceleration for faster computation.
* **Backend and Libraries:** The project requires **NumPy** for data handling, **Matplotlib** for visualization, and **requests** for fetching images via URLs.

The combination of **deep learning, real-time processing, and an intuitive web interface** ensures the system is **fast, scalable, and accessible** for tumor detection applications.

* + 1. **Hardware Requirements:**

 **Processor:** Intel Core i5/i7 or AMD equivalent

 **RAM:** Minimum 8GB (16GB recommended)

 **GPU:** NVIDIA GPU with CUDA support for training

 **Storage:** At least 20GB free space

* + 1. **Software Requirements:**

 **Programming Language:** Python

 **Deep Learning Framework:** PyTorch (Ultralytics YOLO)

 **Web Interface:** Streamlit

 **Image Processing:** OpenCV, PIL

 **Development Environment:** Jupyter Notebook, Google Colab, Local system

**CHAPTER 4**

**Implementation and Result**

The implementation of the Brain Tumor Detection System involves training, testing, and deploying a YOLO-based deep learning model for real-time tumor detection in medical images. The project is developed using Python, PyTorch, OpenCV, and Streamlit, providing a fast and interactive web-based interface. The YOLO model is trained on a brain tumor dataset, learning to detect tumor regions with high accuracy. The trained model is then integrated into a Streamlit app, allowing users to upload brain scan images or enter image URLs for instant analysis. During the testing phase, various MRI scan images were used to evaluate the model's performance. The results are displayed with bounding boxes and confidence scores, helping users interpret the findings. The system successfully identifies tumor-affected areas with high precision and distinguishes non-tumor images confidently.

The following sections showcase the snapshots of the detection results, along with explanations of the output. Additionally, the complete source code of the project is available on GitHub, ensuring transparency, reproducibility, and further improvements.

* 1. **Snap Shots of Result:**

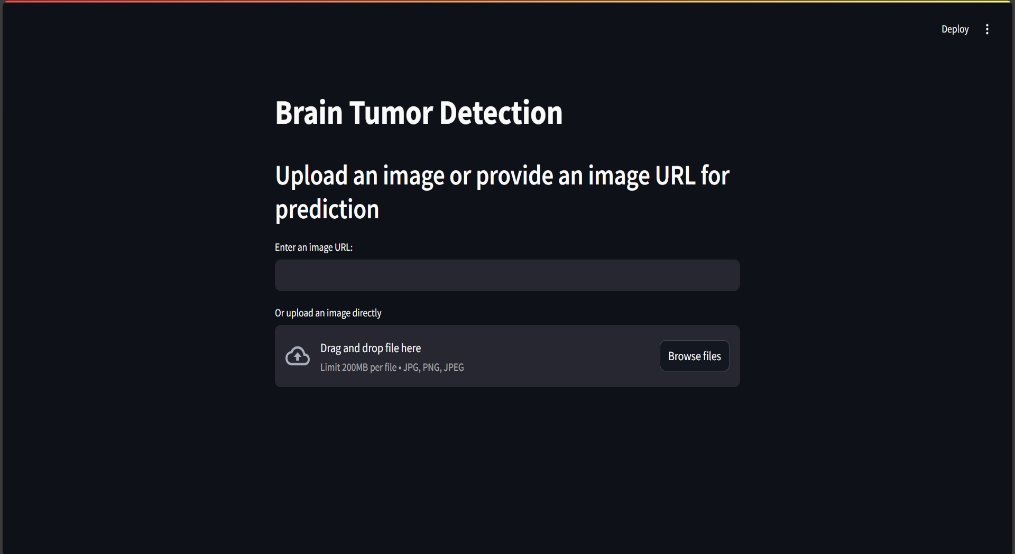
****

Fig. 4.1.1 Streamlit Application Interface

#### **Snapshot 1: Uploading an Image for Detection**

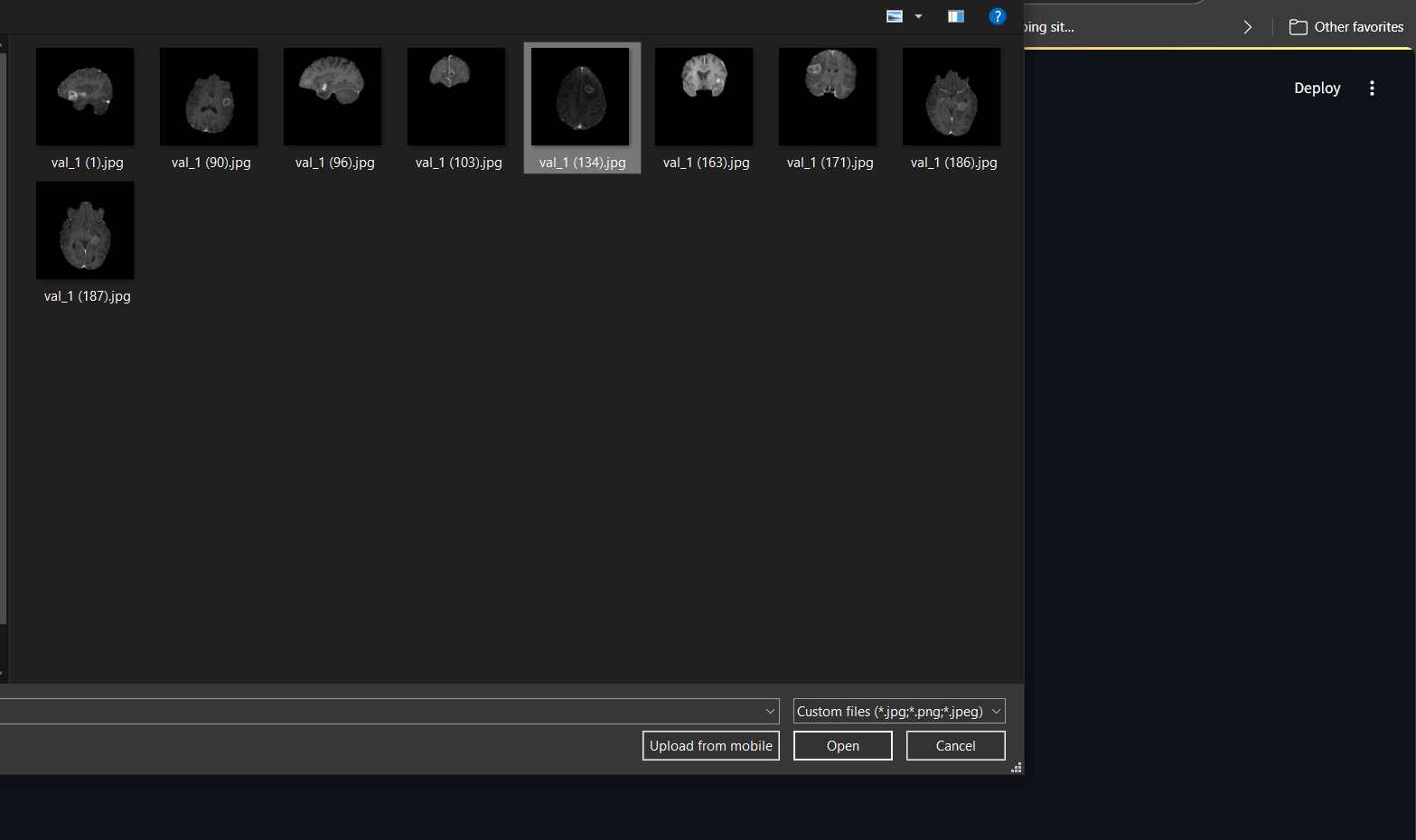


Fig. 4.1.2 **Uploading an Image for Detection**

**Explanation:** This image demonstrates the **user interface** where users can upload an MRI scan or provide an image URL for tumor detection.

#### **Snapshot 2: Tumor Detection Output**

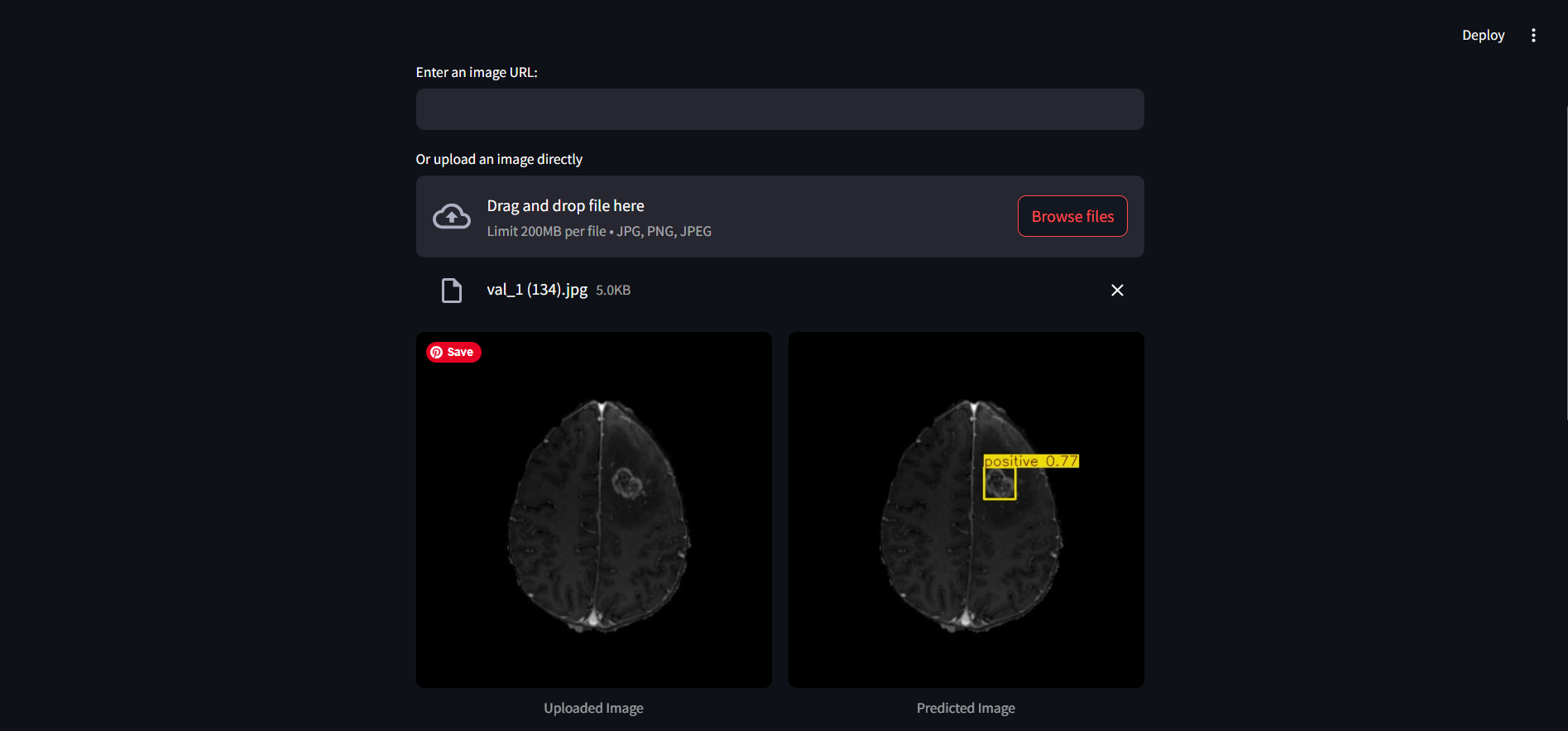


Fig. 4.1.3 **Tumor Detection Output**

**Explanation:** The left side shows the **original brain scan**, while the right side displays the **detected tumor region** with a bounding box. The model assigns **confidence scores** to each detected region, indicating the probability of a tumor being present.

#### **Snapshot 3: Detection Results Summary**

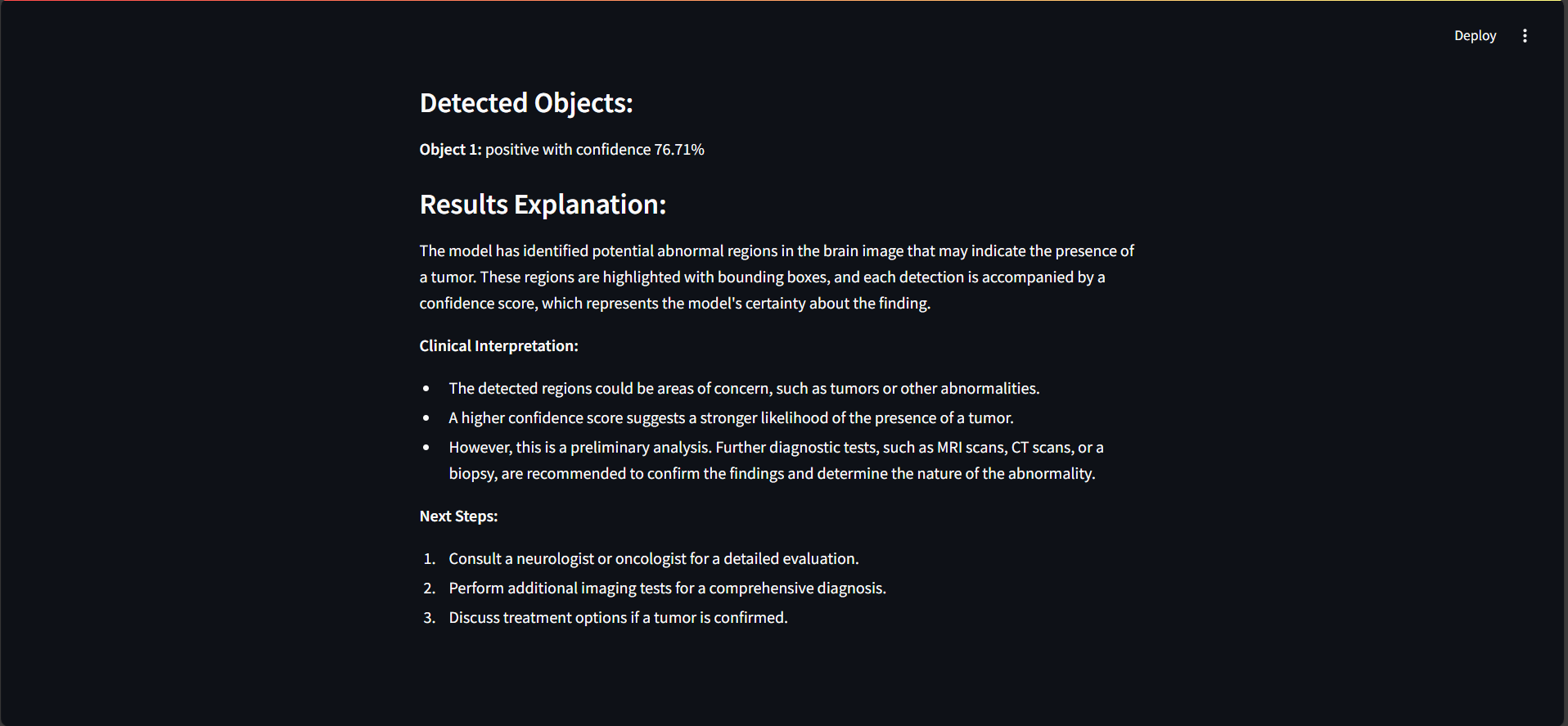


Fig. 4.1.4 **Detection Results Summary**

**Explanation:** This section lists **detection results**, including **tumor presence, class name, and confidence percentage**. If no tumor is detected, the system provides an **explanation and recommended next steps**.

* 1. **GitHub Link for Code:**

The complete source code, including **model training, Streamlit integration, and dataset preprocessing**, is available on GitHub:

🔗 **GitHub Repository:** https://github.com/Shreyas-Meru/Brain-Tumor-Detection

This repository includes **detailed instructions**, allowing researchers, developers, and healthcare professionals to **reproduce, improve, and expand** the project for further advancements.

**CHAPTER 5**

**Discussion and Conclusion**

The proposed brain tumor detection system using YOLO and Streamlit offers a fast, accurate, and accessible method for detecting tumors in medical images. Traditional brain tumor detection relies on manual examination of MRI scans, which can be time-consuming and prone to human error. This project automates the process by leveraging deep learning-based object detection, significantly improving diagnostic speed and accuracy. The system enables real-time tumor localization, allowing users to visualize tumor regions with confidence scores. By providing a user-friendly web interface, it enhances accessibility for medical professionals, researchers, and students.

However, some challenges remain. The model’s performance depends on the quality and diversity of training data. If the dataset lacks sufficient variation, it may struggle with complex or rare tumor cases. Additionally, while YOLO is fast, it might not be as precise as segmentation-based approaches like U-Net, which focus on fine-grained tumor boundaries. The system could further benefit from multi-modal analysis, integrating CT scans or patient history for more comprehensive diagnosis.

Despite these limitations, the project demonstrates the practicality of AI in medical imaging, providing a scalable and efficient solution for early tumor detection. Future enhancements can make it even more robust and clinically applicable.

* 1. **Future Work:**

Although the current system provides **real-time tumor detection**, there are several areas for future improvements. One key enhancement is **model refinement** through transfer learning on **larger and more diverse medical datasets**, improving detection accuracy across different tumor types. Additionally, integrating **MRI and CT scans** can help develop a **multi-modal system**, leading to better generalization and improved diagnostic performance.

Another promising direction is **fine-tuning YOLO with segmentation-based models** like **U-Net or Mask R-CNN**, allowing for **precise tumor boundary detection** rather than just bounding boxes. This hybrid approach could help doctors assess the **exact size, shape, and spread** of tumors more effectively.

Further, incorporating **explainable AI techniques** can make the model’s decision-making process more transparent. By visualizing **feature importance maps**, users can better understand why the model predicts certain regions as tumors.

Finally, deploying the system as a **cloud-based application** with support for **batch processing and automated reporting** could enhance its usability in clinical settings. With these improvements, the model could evolve into a **fully-fledged medical diagnostic tool**, assisting healthcare professionals in **early and accurate** brain tumor detection.

* 1. **Conclusion:**

This project successfully demonstrates the **application of deep learning in medical imaging** by developing an **efficient and user-friendly** brain tumor detection system using **YOLO and Streamlit**. The system automates tumor localization, reducing reliance on manual diagnosis and minimizing human error. By offering a **real-time detection** interface, it enables **faster decision-making** for medical professionals, potentially aiding in **early diagnosis and treatment planning**.

The project’s key contributions include **fast detection, ease of use, and accessibility**, making AI-driven diagnostics more approachable. Unlike traditional classification models, which require **segmentation and post-processing**, YOLO enables **direct object detection**, making it suitable for **real-time applications**.

While the system has **achieved high accuracy**, certain **challenges remain**, such as the **need for more diverse datasets** and **better tumor boundary detection**. Future enhancements, such as **segmentation-based improvements, multi-modal analysis, and explainable AI**, could further refine the model's effectiveness.

Overall, this project highlights the **potential of AI in medical diagnostics**, paving the way for **faster, more reliable, and scalable** solutions in brain tumor detection. With further advancements, AI-driven tools like this could become an essential component of **modern healthcare**, assisting doctors in **timely and precise medical decision-making**.

**REFERENCES**

1. Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. arXiv preprint arXiv:1804.02767.
2. This paper provides an in-depth explanation of the YOLO object detection algorithm, which forms the foundation of this project.
3. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
4. This book covers fundamental deep learning concepts, including convolutional neural networks (CNNs), which are essential for medical image analysis.
5. Menze, B. H., et al. (2015). The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). IEEE Transactions on Medical Imaging, 34(10), 1993-2024.
6. Discusses a widely used brain tumor dataset, providing valuable insights into tumor segmentation and classification challenges.
7. Ultralytics YOLO Documentation - https://docs.ultralytics.com/
8. The official documentation for YOLOv8, detailing model training, inference, and implementation.
9. Streamlit Documentation - https://docs.streamlit.io/
10. Guides on building interactive web applications for AI-based projects.
11. • OpenCV Library - https://opencv.org/
12. Provides functionalities for image processing and visualization.
13. • PyTorch Official Documentation - https://pytorch.org/
14. Details on using deep learning frameworks for training and deploying neural networks.
15. • Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation.
16. This paper introduces U-Net, a widely used model in medical image segmentation, highlighting differences between detection and segmentation approaches.