Automated Detection and Segmentation of Parotid Gland Tumors in CT Images Using YOLOv8

February 26, 2025

Abstract

This report introduces a deep learning approach leveraging the YOLOv8 architecture for the automated detection and segmentation of parotid gland tumors in computed tomography (CT) images. Parotid gland tumors are rare, representing only 0.3 percent of all human tumors and 6 percent of head and neck tumors. Consequently, accurate classification and detection systems are essential. The developed solution employs YOLOv8 for two core tasks: (1) detecting the parotid gland by distinguishing left and right normal regions, and (2) segmenting tumor-affected areas, specifically identifying Cancer, Mix, and Warthin tumor types. The dataset comprises 300 CT images annotated by experts with bounding boxes and segmentation masks. Performance evaluation was conducted using Intersection over Union (IoU), accuracy, precision, recall, and Dice coefficient metrics. Results demonstrate high accuracy in tumor localization and segmentation, highlighting the YOLOv8 model's suitability for computer-aided diagnostic systems. The methodology ensures efficient processing with reliable segmentation outcomes, offering significant potential in clinical applications for assisting radiologists in tumor analysis.

1 Functionalities of the PyQt UI for YOLOv8-Based Parotid Gland Tumor Detection and Segmentation

This section briefly explains the functionalities of the PyQt-based user interface (UI) developed for the YOLOv8 model, which detects and segments parotid gland tumors in CT images. The UI facilitates user interaction with the detection and segmentation models and provides relevant performance metrics.

1.1 Model Initialization and Utilities

The UI initializes the following components:

- **Detection Model (DetModel):** Handles tumor detection in CT images.
- **Segmentation Model (SegmModel):** Segments tumor-affected areas, identifying Cancer, Mix, and Warthin tumor types.
- **Ground Truth Utility (GtGenerator):** Generates ground truth data for performance evaluation.
- **Metrics Utility (MetricsRaiza):** Calculates performance metrics such as IoU, accuracy, precision, recall, and Dice coefficient.

1.2 Image Loading and Navigation

- **Load Folder Button:** Allows users to load a folder containing CT images for processing.
- **Image Navigation:** "Pre" and "Next" buttons enable navigation through loaded images. The current image's file name is displayed for reference.

1.3 Detection Functionality

- **Detection Button:** Runs the YOLOv8 detection model to identify parotid gland regions and tumors.
- **Performance Metrics Display:** Shows key performance metrics after detection:
  - IoU (Intersection over Union)
  - Accuracy
  - Precision

1.4 Segmentation Functionality

- **Segmentation Button:** Activates the YOLOv8 segmentation model, segmenting tumor regions into specified tumor types.
- **Dice Coefficient Display:** Shows the Dice coefficient to indicate segmentation accuracy.

1.5 User Interface Design

- The UI uses PyQt5 with a clean and intuitive design.
- **Image Visualization:** A group box to display CT images.
- **Results Section:** Separate sections for displaying detection and segmentation results, including performance metrics.
- **Background and Style:** Customized background for a user-friendly appearance.

1.6 Overall Functionality

The UI provides an efficient and interactive way to:

- Load and browse CT images.
- Perform automated detection and segmentation of parotid gland tumors.
- Visualize and interpret detection and segmentation results with corresponding performance metrics.

This comprehensive interface enhances usability, ensuring that users can easily process CT images and interpret the results, thus supporting radiologists in clinical tumor analysis applications.

2 User Interface Visualization

This section presents the visual representation of the user interface (UI) developed in PyQt5 for the YOLOv8-based system. The images demonstrate different stages of the application's execution, including loading images, detection results, performance metrics, and segmentation outputs.

2.1 Main Interface View

The main window provides a clean design with key buttons: **Load Folder**, **Pre**, and **Next**, facilitating image navigation and model execution.

2.2 Detection Results View

This image shows the detection stage where the YOLOv8 model marks regions such as *right normal* with a confidence score of 0.84. The visual feedback helps assess the detection performance.

2.3 Performance Metrics View

The UI displays real-time performance metrics:
- **IoU:** 0.303659
- **Accuracy:** 0.903728
- **Precision:** 0.5
- **Recall:** 0.363373
- **Dice Coefficient:** 0.954036

These metrics provide critical insights into the model's performance regarding both detection and segmentation tasks.

2.4 Segmentation Results View

The segmentation stage identifies tumor regions, with each tumor type highlighted in specific colors. The example shows a *mix* tumor type, accurately segmented and labeled with its corresponding confidence value.

2.5 Summary of UI Workflow

The UI provides a complete workflow from image loading to performance evaluation:

- Load and navigate CT images.
- Run YOLOv8 detection for region identification.
- Review performance metrics for detection accuracy.
- Execute segmentation for tumor classification and visualization.

Overall, the PyQt5 interface ensures a seamless user experience, allowing efficient processing and visualization of critical results for clinical applications.

Figure 1: Main UI layout showing the initial view after execution. The interface includes options for loading folders, navigating images, and running detection and segmentation models.

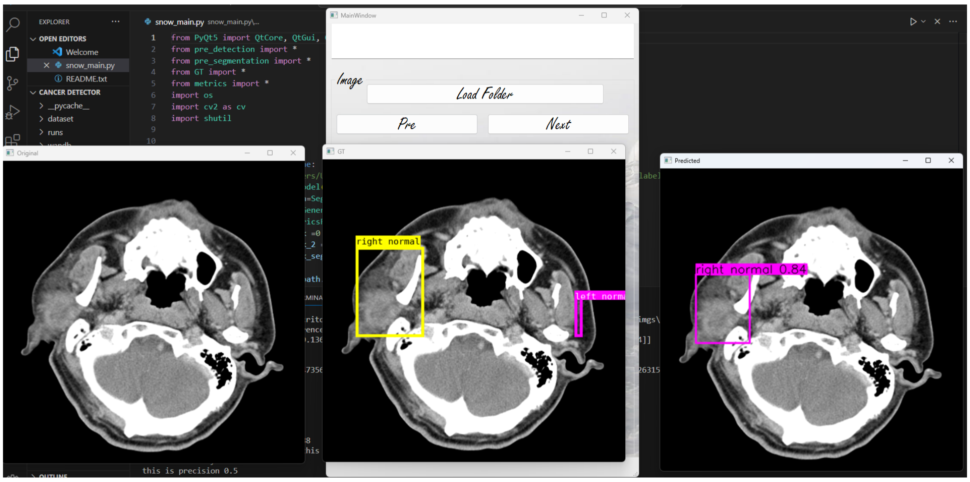

Figure 2: Detection output displaying bounding boxes for the parotid gland regions. The model identifies left and right normal regions with corresponding confidence scores.

Figure 3: Metrics display after detection and segmentation. It shows Intersection over Union (IoU), accuracy, precision, recall, and Dice coefficient.

Figure 4: Segmentation output showing identified tumor areas with class labels and confidence scores. In this example, a *mix* tumor type is detected with 0.84 confidence.