AI-Based Mammographic
Breast Cancer Detection Using Deep Learning:
Model Development and Validation
Using Kaggle's RSNA Dataset

Project overview

Build an end-to-end system for screening mammography breast cancer detection using RSNA data, aligned to the Microsoft ML Projects outline.

Full project description

Project goal: Develop a comprehensive AI-powered breast cancer detection system using deep learning techniques on mammography images from the RSNA Breast Cancer Detection Challenge dataset. This graduation project for the Digital Egypt Pioneers Initiative (DEPI) aims to create a production-ready solution that can assist radiologists in early breast cancer screening.

Project scope: The system will implement both classification (predicting cancer presence) and optional detection (localizing suspicious regions) capabilities. The project follows a structured five-milestone approach covering the complete machine learning lifecycle from data acquisition to production deployment.

Milestone 1: Data Collection, Preprocessing, and Exploration

Objective: Establish a clean, well-structured dataset ready for model training.

Key activities:

- Download and organize the RSNA mammography dataset from Kaggle, ensuring diverse cases across different views and patient demographics
- Implement a robust preprocessing pipeline including image resizing, normalization, and standardization to handle variations in mammogram quality and format
- Design and apply data augmentation strategies (rotation, flipping, brightness adjustments) to increase training data diversity and model robustness
- Create stratified train/validation/test splits to handle class imbalance inherent in medical imaging datasets
- Conduct exploratory data analysis to visualize class distributions, identify dataset biases, and document any annotation quality issues

Team assignments:

- Data acquisition: Member 1
- Preprocessing pipeline: Ahmed, Jamal, Omnia (Members 2-3)
- Data augmentation: Member 3
- Stratified splitting: Member 4
- EDA and documentation: Tasnim, Dr. Mohammed (Member 5)

Deliverables: Cleaned dataset, preprocessing documentation, augmentation functions, EDA report with visualizations

Milestone 2: Model Development and Optimization

Objective: Build and optimize classification and detection models for breast cancer screening.

Key activities:

- Develop baseline CNN architectures from scratch using TensorFlow/Keras or PyTorch to establish performance benchmarks
- Implement transfer learning using pre-trained models (ResNet50, EfficientNet, InceptionV3) fine-tuned for mammography classification
- Build object detection models (YOLOv5, YOLOv8, or Faster R-CNN) to localize suspicious regions if bounding box annotations are available
- Evaluate models using comprehensive metrics: accuracy, precision, recall, F1-score for classification; mAP and IoU for detection
- Perform hyperparameter tuning using GridSearch, Optuna, or manual optimization to maximize performance

Team assignments distributed across:

- Baseline CNN development
- Pre-trained model implementation
- Object detection model
- Evaluation metrics and visualization
- Hyperparameter tuning and comparative analysis

Deliverables: Trained classification and detection models, evaluation reports with confusion matrices, model comparison notebook

Milestone 3: Advanced Techniques, Transfer Learning, and Cloud Integration

Objective: Enhance model performance through advanced techniques and prepare for scalable cloud deployment.

Key activities:

- Apply transfer learning from ImageNet and domain-specific backbones to leverage pre-trained feature representations
- Fine-tune models specifically for mammography data characteristics
- Integrate with Azure ML or Azure Custom Vision for managed training, experiment tracking, and model registry
- Package models in Docker containers for consistent deployment across environments
- Develop REST APIs using FastAPI or Flask to expose model predictions as web services
- Create a unified system interface supporting both classification and detection workflows

Team assignments:

- Transfer learning for classification
- Transfer learning for detection
- Model optimization (early stopping, learning rate scheduling, quantization/pruning)
- Azure deployment infrastructure
- API development and integration testing

Deliverables: Fine-tuned models, Azure-deployed endpoints, integrated system demo, API documentation

Milestone 4: MLOps, Monitoring, and Web Interface

Objective: Establish production-grade MLOps infrastructure and user-facing application.

Key activities:

- Set up experiment tracking using MLflow or Azure ML to version models, log metrics, and manage the model registry
- Build CI/CD pipelines for automated training, testing, and deployment workflows
- Develop a web application (Streamlit, Flask, or FastAPI) allowing users to upload mammogram images and receive classification/detection results

- Implement monitoring solutions (Azure Monitor, Prometheus) to track model performance, latency, error rates, and accuracy drift
- Design retraining strategies with scheduled updates and triggers based on performance degradation or new data availability
- Configure alerting for threshold breaches in key performance indicators

Team assignments:

- Experiment tracking setup
- CI/CD pipeline development
- Web UI implementation
- Monitoring infrastructure
- Retraining pipeline and documentation

Deliverables: Production web application, MLOps pipeline documentation, monitoring dashboards, model lifecycle documentation

Milestone 5: Final Documentation and Presentation

Objective: Compile comprehensive project documentation and deliver final presentation.

Key activities:

- Write final project report covering problem statement, methodology, experiments, results, and clinical impact
- Document all model architectures, training procedures, and performance benchmarks
- Summarize the complete deployment pipeline from data preprocessing to production monitoring
- Create presentation slides with clear visualizations, key findings, and live demo
- Demonstrate the deployed application with real mammogram predictions
- Identify future improvements including edge deployment, federated learning, model speed optimizations, and multi-modal integration

Team assignments:

- Introduction and background
- Model development documentation
- Deployment and MLOps documentation

- Presentation design
- Live demo and future work recommendations

Deliverables: Final project report, presentation deck, live demo, future roadmap

Technical stack

- **Deep learning frameworks:** TensorFlow, Keras, PyTorch
- **Cloud platform:** Azure ML, Azure Custom Vision
- **MLOps tools:** MLflow, DVC (Data Version Control), Azure ML pipelines
- **Model architectures:** ResNet, EfficientNet, InceptionV3, YOLOv5/v8, Faster R-CNN, SSD
- **Deployment:** Docker, FastAPI, Flask, Streamlit
- **Monitoring:** Azure Monitor, Prometheus
- **Model export:** TorchScript, ONNX for production optimization

Expected outcomes

A fully functional, production-ready breast cancer detection system that can process mammography images, provide cancer risk predictions, optionally localize suspicious regions, and operate at scale with comprehensive monitoring and retraining capabilities. The system will demonstrate end-to-end MLOps best practices suitable for healthcare AI applications.

Resources

- Competition page and data: Kaggle link
- Dataset mirror and description: AWS Registry entry
- RSNA challenge overview and rules

Team Members: 5

Project Toolset:

TensorFlow / Keras / Azure ML / MLflow