

Early Detection of Melanoma Using Deep Learning

Course : DATS 6303 - Deep Learning

Team:

Amrutha Jayachandradhara

Rasika Nilatkar

George Washington University

Under the guidance of,

Professor Amir Jafari

TA: Parv Bhargava

Introduction:

Melanoma, a severe form of skin cancer, has seen a significant rise in incidence and mortality over the past decade. Early diagnosis is crucial, as the 5-year survival rate can drop from 99% to 27% when the cancer spreads. Traditional diagnostic methods, including manual examination and visual inspection of dermoscopy images, are prone to human error. Therefore, an automated, robust, and accurate Computer-Aided Diagnosis (CAD) system is essential for improving early detection.

Objective:

This proposal aims to develop an advanced deep learning-based system for melanoma detection that integrates lesion classification. The approach leverages different pre-trained CNN architectures to classify segmented lesions efficiently.

Methodology:

Datasets: ISIC 2017 , ISIC 2018 and ISIC 2019 for classification.

The classification stage will involve the use of Convolutional Neural Network (CNN) architectures. **ResNet** and **InceptionNet** will be specifically utilized due to their effectiveness in medical image analysis. These models will be adapted using transfer learning to leverage pre-trained weights from large datasets like ImageNet.

Customization will be performed to tailor the models for multi-label skin lesion classification.

- A pooling layer will be added to reduce the spatial dimensions of the feature maps.
- Batch Normalization will be used to stabilize training and improve convergence speed.
- ReLU activation will be applied to introduce non-linearity to the network.
- Sigmoid activation will be used in the output layer to support multi-label classification.

Implementation:

The proposed network will be implemented using **PyTorch** within PyCharm. This setup is particularly well-suited for building complex architectures such as ResNet and InceptionNet, applying transfer learning, and performing multi-label classification tasks

The **Adam optimizer** will be used for training, along with **adaptive learning rate strategies** and **early stopping** to ensure efficient convergence.

Expected Outcome:

The combination of well-established evaluation metrics: AUC score, precision, and recall are used. These metrics are crucial given the clinical importance of accurately distinguishing melanoma from other benign skin lesions, which often appear visually similar. The **AUC** (Area Under the Curve) score will serve as a primary performance indicator, reflecting the model's

overall ability to distinguish between malignant and non-malignant classes across thresholds.

Precision will be used to evaluate the proportion of true positives among all predicted positives, reducing the risk of false alarms, while recall (sensitivity) will measure the model's ability to correctly identify all true melanoma cases, minimizing false negatives.

References:

1. "Deep-learning-based, computer-aided classifier developed with a small dataset of clinical images surpasses board-certified dermatologists in skin tumor diagnosis." *British Journal of Dermatology*: <https://pubmed.ncbi.nlm.nih.gov/29953582/>
2. The datasets used (ISIC 2017, 2018, and 2019) are part of the International Skin Imaging Collaboration (ISIC). Documentation and challenge descriptions from the ISIC website will help understand dataset structure, labeling conventions, and evaluation metrics.
3. Public GitHub repositories and TensorFlow Model Garden will be explored for implementation references and community-driven enhancements to networks like ResNet, InceptionNet, and U-Net.
https://github.com/tensorflow/models?utm_source=chatgpt.com

Schedule :

Week 1 (April 8 – April 14): Setup, Research & Preprocessing

- Finalize project objectives and tools (Pycharm, Pytorch).
- Download and clean ISIC 2017, 2018, and 2019 datasets.
- Normalize, resize and set up data augmentation.

Week 2 (April 15 – April 21): Segmentation & Classification Pipeline

- Prepare images for classification.
- Start transfer learning setup for CNNs (DenseNet201, etc.).
- Train at least 2 models (e.g., ResNet152V2 and InceptionV3) on data.

Week 3 (April 22 – April 28): Model Training & Evaluation

- Train remaining CNN models (DenseNet201, InceptionResNetV2).
- Apply k-fold cross-validation and early stopping.
- Collect evaluation metrics: AUC, precision, recall.

Week 4 (April 29 – May 7): Optimization, Reporting & Finalization

- Fine-tune hyperparameters if needed.
- Run final tests on unseen data for generalization and finalize the application