COMPARISON BETWEEN NETWORKS FOR DIABETIC RETINOPATHY CLASSIFICATION

Lakshay Choudhary and Swarnendu Sengupta

Institute of Signal Processing and System Theory, University of Stuttgart, Germany

Universität Stuttgart

1. Introduction

Diabetic retinopathy is an illness which causes damages to retina. The goal of this project is to compare existing pre-trained architectures to classify diabetic and non-diabetic retinopathy with deep learning algorithms.

2. Methodology

2.1 Dataset used

Indian Diabetic Retinopathy Image Dataset (IDRID) with various degrees of diabetic retinopathy

2.2 Input pipeline

1. Preprocessing -
   - Redefine the labels to make it suitable for binary classification
   - Creation of a bounding box to crop the image and then reshaping it to 256×256
   - Upsampling to balance the dataset (optional)
2. CLAHE equalization -
   Improvement in the contrast of the image by redistribution of pixel intensities for reduction of noise over-amplification
3. Data augmentation -
   - Method: Rotation, Flipping, Brightness, Contrast changes
   - To prevent overfitting

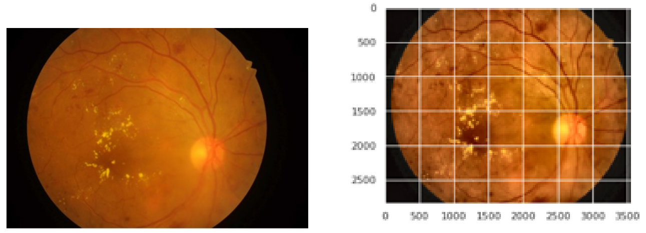


Figure 1: *Before and after CLAHE and bounding box*

2.3 Models

1. Baseline - CNN-DNN
   A small model which would serve as the baseline for other models to be used and trained on

Table 1: *Baseline architecture*

Conv 3×3, ch = 16^*, ReLU
MaxPooling 2×2 (stride - None)
Conv 3×3, ch = $16^* \times 2$, ReLU
MaxPooling 2×2 (stride - None)
Conv 3×3, ch = $16^* \times 4$, ReLU
MaxPooling 2×2 (stride - None)
Dropout rate 0.2^*
Flatten
DNN - $16^* \times 8$, ReLU
Output - 1, sigmoid

2. InceptionV3, ResNet-50, EfficientNetV2 - pretrained models were used; pretrained on ImageNet)

Table 2: *Architecture*

InceptionV3 / ResNet-50 / EfficientNetV2 pretrained model
Global Avg Pooling 2D
DNN - 512, ReLU
Dropout - 0.2^*
Output - 1, sigmoid

* indicates possibility of tuning.

Binary cross-entropy was used as the loss function. **F1-score** was used as the evaluation criteria.

2.4 Training and evaluation

(a) Adam optimiser was used
(b) Early stopping was used (patience parameter along with epochs can be given)
(c) Hyperparameter tuning - Grid search done
   i. Baseline
      - No. of units in Conv layers - 16, 32
      - Learning rate - 1e-1, 1e-3
      - Dropout rate - 0.1, 0.5
   ii. Pretrained model architecture
      - Learning rate - 1e-1, 1e-3
      - Dropout rate - 0.1, 0.5
(d) Model saving - can be done after training with and without hyperparameter optimisation, fine tuning; saved after the whole operation is completed
(e) Bounding box - Removal of redundant pixels
(f) Configuration with **gin**

2.5 Deep visualization

- Method used - Grad-CAM
- The derivatives of prediction with respect to convolution output show network's learning.

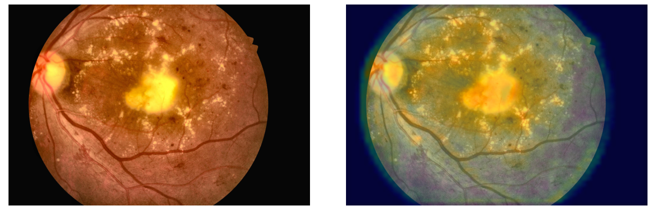
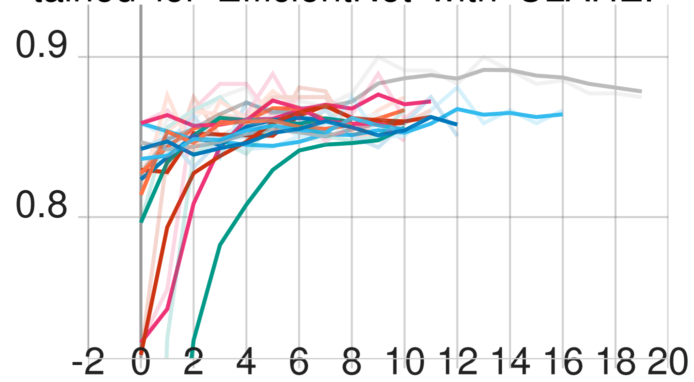


Figure 2: *Original and on application of GradCAM. The network mainly focuses on hard exudates on retinal fundus, which is reasonable and was expected*

3. Results

Model	CLAHE	F1-score	Accuracy
Baseline	No	72.57	64.1
Baseline	Yes	73.42	65.2
EfficientNet	No	82.76	79.12
EfficientNet	**Yes**	**83.47**	**82.5**
ResNet	No	73.4	64.9
ResNet	Yes	75.1	63.1
Inception V3	No	73.87	69.43
Inception V3	Yes	75.3	74.8

The graph below shows the F1-score of the testing output obtained for EfficientNet with CLAHE.



F1-score in EfficientNet with CLAHE