DIABETIC RETINOPATHY DETECTION

A PREPRINT

Yu Zhang
Electromobility
University of Stuttgart
70569 Stuttgart
st176111@stud.uni-stuttgart.de

Zening Du
Electromobility
University of Stuttgart
70569 Stuttgart
st175322@stud.uni-stuttgart.de

July 19, 2022

ABSTRACT

Diabetic retinopathy detection is a challenging task. In this paper, a VGG-like model was employed to accomplish the binary classification between non-referable (NRDR) and referable diabetic retinopathy (RDR). A test accuracy of 80.58% was obtained. Deep visualization of the model shows some useful features to distinguish between different grades of diabetic retinopathy. Some issues are worth thinking about, e.g. a tuned model with high validation accuracy shows relatively lower test accuracy.

1 Introduction

Diabetic retinopathy is a diabetes complication that damages eyes. It is a leading cause of vision loss. Distinguishing between different grades of diabetic retinopathy is important but also challenging.

The Indian Diabetic Retinopathy Image Dataset (IDRID) is a dataset of retinal fundus images that is publicly available [1]. In this paper, we use deep learning methods to learn the small but important features of fundus images in IDRID dataset, like microaneurysms, soft exudates, hemorrhages and hard exudates of the fundus. And we classify the images into non-referable (NRDR) or referable diabetic retinopathy (RDR).

The workflow can be divided into input pipeline, model, metrics, training and evaluation, hyperparameter optimization, and deep visualization. In the following sections they are explained in details.

2 Input pipeline [Yu Zhang]

2.1 Preprocessing [Yu Zhang]

The given IDRID dataset is relatively small. The test set has 103 images, while the train set has 413 images. For a binary classification task, the dataset was relabeled. Images with original labels 0-1 were relabeled to 0 (NRDR), while images with original labels 2-4 were relabeled to label 1 (RDR).

The raw training set was further split into a validation and training set (ratio: 20/80). We noticed that the new training set is highly imbalanced. After applying oversampling on it, a balanced training set was created. The distribution of train/val/test split and dataset balancing are summarized in Table 2.1.

Each original image in the dataset contains uninformative black areas on the left and right side of the retina. After cropping the black edges each image was resized to 256×256, as shown in Figure 2.1.

Table 2.1: Train/val/test split and dataset balancing

Train/val/test split	Label 0	Label 1
Test set	39	64
Val set	30	53
Imbalanced training set before oversampling	124	206
Balanced training set after oversampling	206	206

(a) IDRiD_196 of size 2848×4288 (b) Cropping and resizing to 256×256

Figure 2.1: Image preprocessing

2.2 Data augmentation [Yu Zhang]

After trying different data augmentation operations, we found that the following data augmentations had more significant influence on the improvement of accuracy: flipping horizontally/vertically; random brightness. Some examples are presented in Figure 2.2.

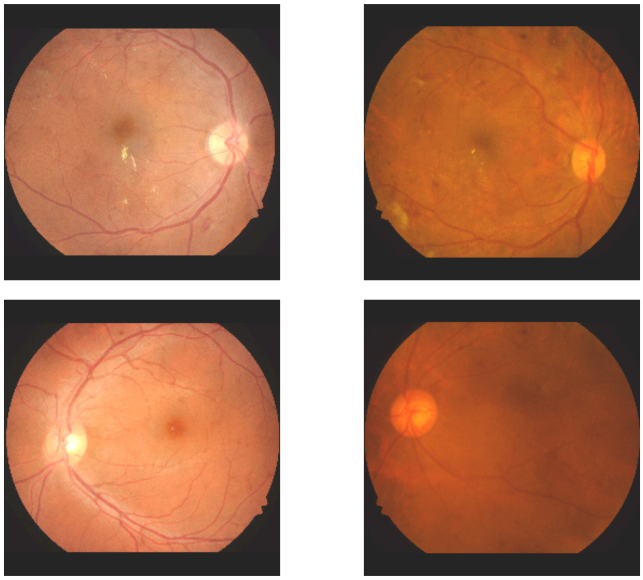

Figure 2.2: Data augmentation

3 Model [Yu Zhang and Zening Du]

A VGG-like model [2] shown in Figure 3.1 was adapted in this project. The main component of this model is the "VGG-block", which contains two convolutional layers with variable number of filters and kernel size, and one Max-pooling layer with pool size 2×2. After stacking several VGG-blocks, a global average pooling layer, a dense layer with variable number of dense units, a dropout layer and another dense layer with a fixed number of dense units are added, in order to get the binary classes outputs.

Batch size, number of filters, kernel size, number of VGG-blocks, number of dense units and dropout rate are optimized during hyperparameter tuning.

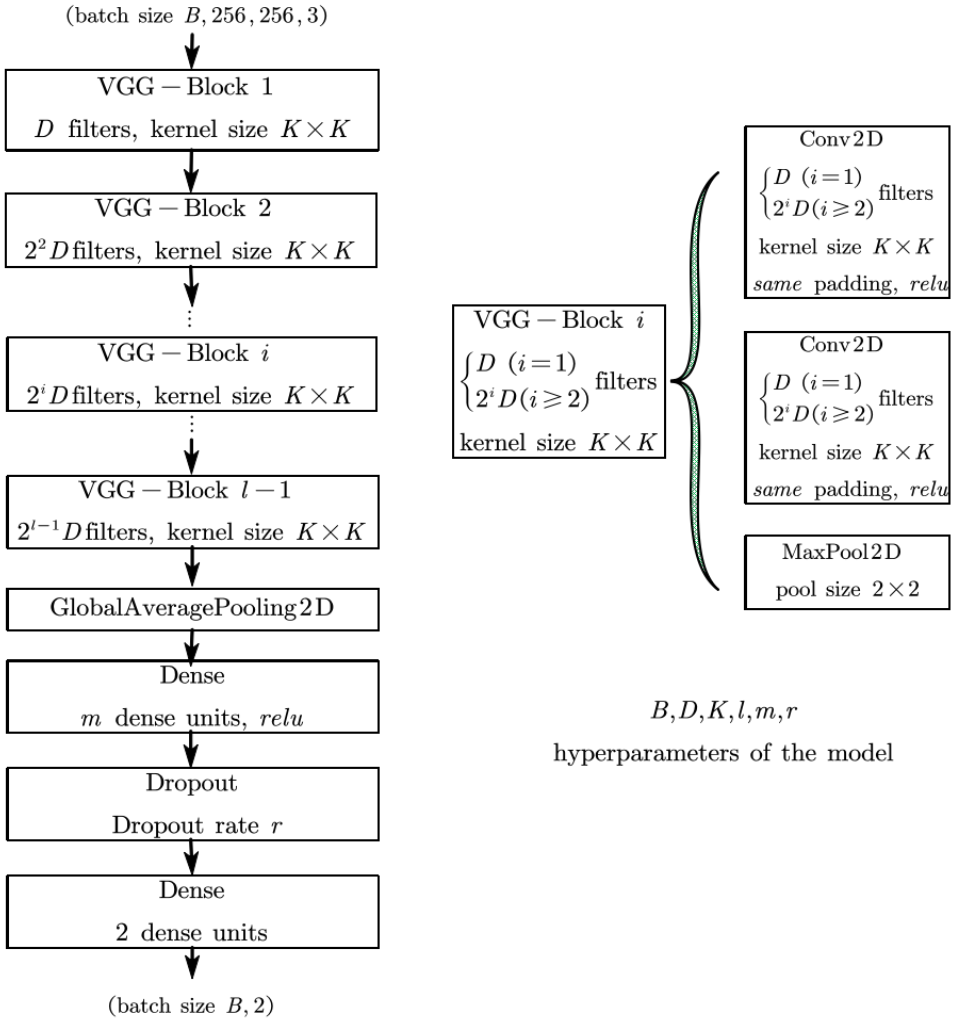

Figure 3.1: Model architecture

4 Metrics [Zening Du]

The metric in this project is mainly confusion matrix. For the binary classification problem, we also consider precision, F1-score and sensitivity.

5 Training and evaluation [Yu Zhang]

The loss function is sparse categorical cross-entropy, while the optimizer is Adam. The learning rate of Adam is also a hyperparameter which remains to be tuned.

For the initial hyperparameter settings (checkpoint 0), the corresponding training results are listed in Table 5.1.

Using this trained model to evaluate on the test set, the results are shown in Table 5.2.

Table 5.1: Training results for the checkpoint 0

	checkpoint 0
Learning rate	3×10^{-4}
Total steps	5000
Batch size	32
Number of filters	16
Number of dense units	64
Dropout rate	0.44
Number of VGG-blocks	5
Total parameters of model	1 171 242
Best validation accuracy	86.75
Confusion matrix for the validation set	$\begin{bmatrix} 25 & 5 \\ 6 & 47 \end{bmatrix}$

Table 5.2: Evaluation results for checkpoint 0 in Table 5.1

	checkpoint 0
Test accuracy	80.58%
Confusion matrix for the test set	$\begin{bmatrix} 30 & 9 \\ 11 & 53 \end{bmatrix}$
Precision	$[\ 0.73 \quad 0.85\ ]$
Sensitivity	$[\ 0.77 \quad 0.83\ ]$
F1-score	$[\ 0.75 \quad 0.84\ ]$

6 Hyperparameter optimization [Yu Zhang]

The goal of hyperparameter optimization is to maximize best validation accuracy. Bayesian optimization was employed. Hyperparameters and their search space are shown in Table 6.1.

The best two results (checkpoint 1 and 2) after hyperparameter tuning are listed in Table 6.2.

But when evaluating with these two checkpoints on the test set, we found that the test accuracy is significantly lower than the validation accuracy, as shown in Table 6.3. This may result from overfitting during hyperparameter tuning, or different data distribution of the validation and test set.

A comparison between the initial hyperparameters, tuned hyperparameters 1 and tuned hyperparameters 2 shows: using has relatively better test accuracy, while checkpoint 2 (tuned) has significantly fewer total parameters. One must trade them off.

Table 6.1: Hyperparameters and their search space

Hyperparameters	Search space
Learning rate	Log uniform: $\log(10^{-5}) \sim \log(10^{-3})$
Total steps	2×10^4
Batch size	32
Number of filters	16
Number of dense units	64
Dropout rate	0.44
Number of VGG-blocks	5
Total parameters of model	1 171 242
Best validation accuracy	86.75%
Confusion matrix for the validation set	$\begin{bmatrix} 25 & 5 \\ 6 & 47 \end{bmatrix}$

Table 6.2: The best two results after hyperparameter tuning

	checkpoint 1	checkpoint 2
Learning rate	1.06×10^{-4}	1.27×10^{-4}
Total steps	2×10^4	2×10^4
Batch size	32	16
Number of filters	16	8
Number of dense units	128	128
Dropout rate	0.67	0.79
Number of VGG-blocks	5	5
Total parameters of model	1188818	306026
Best validation accuracy	92.77%	96.39%
Confusion matrix for the validation set	$\begin{bmatrix} 25 & 5 \\ 1 & 52 \end{bmatrix}$	$\begin{bmatrix} 29 & 1 \\ 2 & 51 \end{bmatrix}$

7 Deep visulization [Yu Zhang and Zening Du]

For deep visualization, Gradient-weighted Class Activation Mapping (Grad-CAM) [3] was employed. As shown in the highlighted areas of Figure 7.1, the model really learned some subtle features like microaneurysms, soft exudates, hemorrhages and hard exudates of the fundus. The combination of Grad-CAM and Guided Backpropagation [4] – Guided Grad-CAM is also presented in Figure 7.1.

Table 6.3: Evaluation results for checkpoint 1 & 2

	checkpoint 1	checkpoint 2
Test accuracy	71.84%	73.79%
Confusion matrix for the test set	$\begin{bmatrix} 18 & 21 \\ 8 & 56 \end{bmatrix}$	$\begin{bmatrix} 23 & 16 \\ 11 & 53 \end{bmatrix}$
Precision	[0.69 0.73]	[0.68 0.77]
Sensitivity	[0.46 0.88]	[0.59 0.83]
F1-Score	[0.55 0.79]	[0.63 0.80]

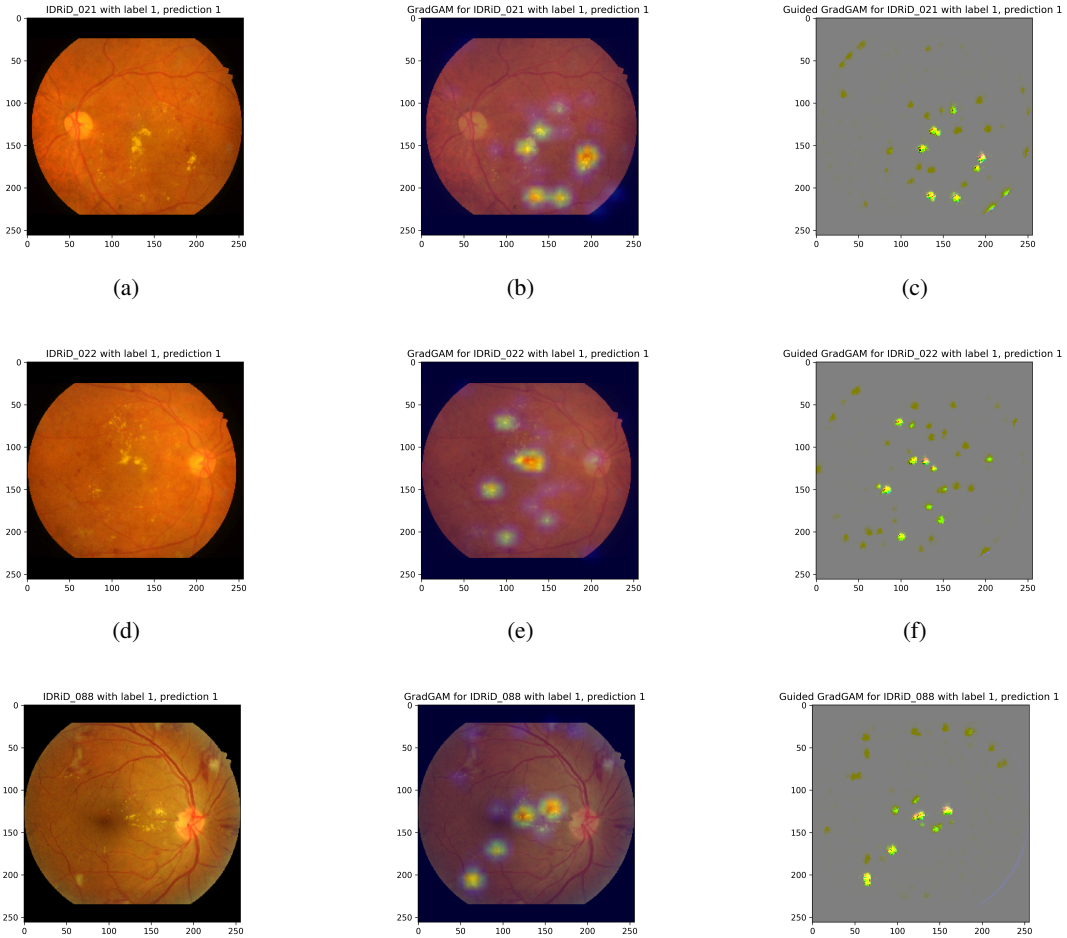

Figure 7.1: Some results of deep visualization

8 Conclusion

The given IDRID dataset is relatively small, impeding the classification task. After balancing the dataset, we get a good test accuracy using a VGG-like model. Deep visualization of the last convolutional layer shows some useful subtle features on the fundus to distinguish between different grades of diabetic retinoathy.

We encounter a problem during hyperparameter optimization, that is, a tuned model with high validation accuracy shows relatively lower test accuracy. This may owe to overfitting during hyperparameter tuning, or different data distribution of the validation and test set.

References

[1] P. Porwal, S. Pachade, R. Kamble, M. Kokare, G. Deshmukh, V. Sahasrabuddhe, and F. Meriaudeau. Indian diabetic retinopathy image dataset (idrid): a database for diabetic retinopathy screening research. In *Data*, vol. 3, no. 3, p. 25, 2018

[2] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *arXiv preprint arXiv:1409.1556*, 2014.

[3] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[4] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. In *arXiv preprint arXiv:1412.6806*, 2014.