Analysis of the Impact of Preprocessing Techniques on the Performance of Diabetic Retinopathy Image Classification

Nhi Ngoc-Yen Nguyen[1,2,3], Anh Duc Nguyen[1,2,3],Hien The Liu[1,2,3], Hop Trong Do[1,2,4]

[1] University of Information Technology, Ho Chi Minh City, Vietnam
[2] Vietnam National University, Ho Chi Minh City, Vietnam
[3] {21521231, 21520140, 21522062}@gm.uit.edu.vn
[4] Correspondence: hopdt@uit.edu.vn

Abstract. Diabetic retinopathy, a retinal disorder, affects up to 85% of diabetic patients, making its accurate diagnosis and early treatment crucial. Currently, various classification models yield promising evaluation results. However, the effectiveness of preprocessing techniques in diabetic retinopathy image classification remains unverified. This research addresses a gap in existing knowledge by investigating the impact of preprocessing on classification accuracy, potentially leading to the streamlining of current methods and eliminating unnecessary steps. We employ Analysis of Variance (ANOVA) to statistically evaluate the influence of preprocessing techniques. The novelty of this research lies in applying hypothesis testing to quantify the effectiveness of preprocessing, leading to more robust and verifiable results. This approach can improve the certainty of findings compared to traditional methods. By investigating this topic, we contribute to a more nuanced understanding of how preprocessing impacts diabetic retinopathy image classification. This knowledge can inform the development of more efficient and accurate diagnostic tools.

Keywords: Computer Vision · Diabetic Retinopathy · Machine Learning · Deep Learning.

1 Introduction

Preprocessing involves eliminating noise and unnecessary information that may reduce the performance of a model, allowing the model to focus more on important information and achieve better performance. This is a critical step in image processing problems. However, its effectiveness varies across different models, image types, and processing tasks, leading to ongoing debates. Traditional machine learning and statistical methods often incorporate preprocessing techniques to enhance performance. In recent studies, advanced deep learning methods are applied, but with fewer reports. The effectiveness of preprocessing is quite good and depends on the model architecture, image type, and processing task. However, even though preprocessing is a fundamental step in image

processing, many people tend to apply all preprocessing techniques to every image processing problem without checking their effectiveness. This can sometimes be time-consuming and ineffecient. In previous studies, the effectiveness of the image preprocessing techniques was not considered properly, especially on this task, which led to trail of preprocessing images and unnecessary steps. By using statistical methods, the goal of our research group is to analyze the effectiveness of preprocessing techniques in the task of diabetic retinopathy detection and diagnosis. Diabetic retinopathy is a significant cause of visual impairment and blindness in adults, especially those with diabetes. Detecting and preventing this condition is crucial to protect vision and ensure comprehensive visual development for individuals. By leveraging the power of machine learning, the model aims to enhance diagnostic capabilities and extend its applications to eye-care experts in the healthcare community and beyond. To analyze the impact of applying preprocessing techniques in the diabetic retinopathy detection and diagnosis task, our group used two preprocessing techniques in conjunction with six models, including traditional machine learning, advanced deep learning, and two datasets. We employed statistical hypothesis testing with one-way and two-way variance analysis with repeated measures and Tukey's post hoc test.

The contribution of this paper is to demonstrate the limited benefits of applying inappropriate preprocessing techniques in the task of diabetic retinopathy detection and diagnosis. Most preprocessing techniques resulted in insignificant performance improvements. Meanwhile, the statistical hypothesis testing showed no significant differences among the image preprocessing techniques. Additionally, the deep learning methods were less affected by preprocessing techniques compared to the machine learning methods.

2 Literature review

Many existing studies are attempting to automate the diagnosis process of diabetic retinopathy for various reasons, such as providing rapid diagnosis, remote diagnosis, and cost-effectiveness [1]. These studies utilize machine learning and deep learning techniques along with image processing methods to classify images of diabetic retinopathy. The classification tasks in these works can be categorized based on the number of classification types performed, for example, binary classification (normal images and images with diabetic retinopathy), multi-class classification with 3 classes (normal, non-proliferative diabetic retinopathy, and proliferative diabetic retinopathy), and recently used 5-class classification. Some notable studies include the use of a customized DL ResNet model for diabetic retinopathy detection [2]. In this model, the authors added 3 sets of output layers alongside intermediate layers and employed an 11-layer ResNet. Although the overall accuracy reached 81%, the model faced difficulties in distinguishing mild and normal diabetic retinopathy images. In another study [3],], the authors discussed the stages of designing a DR classification model. The overall process was segmented and conducted in a sequential manner to understand the importance of each technique. Another related work, a research paper [4] em-

phasized the importance of studying appropriate image preprocessing techniques for DR classification. In this study, the group focused on analyzing the impact of preprocessing techniques on the algorithm's performance.

3 Analysis of the Impact of Preprocessing Techniques in Diabetic Retinopathy Classification

3.1 Procedure of analysis

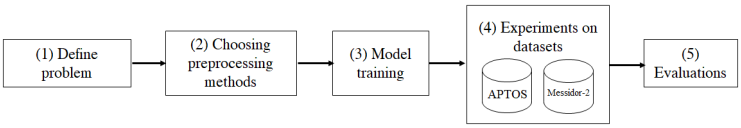

Fig. 1: Experimental analysis procedure

The experiments in this paper involve two main factors: image preprocessing techniques and classification models. To achieve reliable results, each processing method is run on different models and applied to two datasets. Analysis of variance (ANOVA) is applied to the experimental results to assess the effects of image preprocessing techniques, classification machine learning models, and the interaction between these two factors on the obtained results. Tukey's Honestly Significant Difference (HSD) test is also used to identify which groups show significant differences and which do not for the experimental outcomes. The analysis procedure is illustrated in Figure 1.

3.2 Proposed Preprocessing Method

In image classification tasks, including computer vision tasks in general, preprocessing plays a crucial role as it helps enhance available images, highlight image details and features, and can also improve model performance. Based on the diabetic retinopathy competition on Kaggle, the group decides to use three specific preprocessing methods, illustrated in Figure 2, as follows:

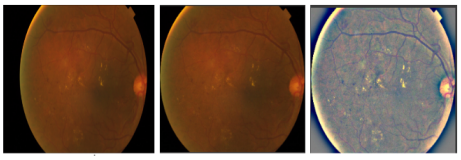

Fig. 2: Images from left to right describe the preprocessing techniques: Resize, Resize combined with Circle Crop, and Circle Crop combined with Ben Graham.

Without preprocessing: We did not use any specific preprocessing methods to avoid the risk of these techniques leading to worse results than normal.

However, all images were resized to a fixed size of 224x224, which was used as the default size for all methods.

Circle crop: We removed black regions around the images that had little impact on the conclusions. Specifically, the method involved removing rows and columns that only contained pixel values less than 7 (the highest value to remove black regions), contributing to avoiding the model's conclusions based on black regions in the images.

Ben Graham: We reused a preprocessing technique proposed by Ben Graham, who won the APTOS competition on Kaggle. The method involved blending images between the original image and the original image processed with GaussianBlur. The parameters included (1) Image1 is the original image; (2) Image2 is the original image processed with GaussianBlur with a kernel size of $(0, 0)$; (3) Alpha and beta are custom hyperparameters with values set by the author as 4 and -4, respectively; (4) Y is another hyperparameter used to bring the data to the desired value range, and reprssented as the following mathematical function:

$$Image = alpha \times Image1 + beta \times Image2 + y \tag{1}$$

3.3 Models and Performance Evaluation Metrics

We conducted experiments using both Deep Learning and Machine Learning models to provide the most objective conclusions:

KNN (K-Nearest Neighbors) is a supervised machine learning algorithm used for classification and prediction based on the principle of 'nearest neighbors.' When predicting, KNN finds k nearest data points in the training set and decides the label for the new data point based on the majority label of the nearest points.

SVC (Support Vector Machine) is a supervised machine learning model used in classification tasks. It creates the best hyperplane to separate two classes of data so that the distance between the hyperplane and the nearest data points is maximized.

Random Forest is a supervised machine learning algorithm used for classification and prediction. It is an ensemble of multiple decision trees. Each decision tree is built on a subset of the data and makes predictions. The results of the individual trees are then combined to make the final prediction.

CNN (Convolutional Neural Network) is a type of neural network used in image processing and analysis. It uses convolutional layers to automatically learn features from images and pooling layers to reduce the input size. CNN can discover structural and meaningful features in images and is commonly used in classification, recognition, and embedding tasks.

ResNet-18 is a CNN architecture proposed by Kaiming He. ResNet uses residual blocks to address the problem of accuracy degradation in deep networks. ResNet-18 is effective for handling classification and image recognition tasks on large and complex datasets.

VGG-16 is a CNN architecture proposed by Karen Simonyan and Andrew Zisserman. VGG-16 is known for its simple architecture with small convolutional

layers, yet it achieves excellent performance in image classification and recognition tasks. However, due to its large number of parameters, VGG-16 requires significant computational resources during training.

Evaluation Metric For the classification task, the group decided to use the F1-score$_{\text{micro}}$ as the evaluation metric. It is calculated specifically using True Positive (TP), False Positive (FP), False Negative (FN) as the following formula:

$$F1_{\text{micro}} = \frac{TP}{TP + \frac{1}{2} \times (FP + FN)} \tag{2}$$

4 Experiments and Analysis

4.1 Experimental datasets

The datasets used in the experiments are APTOS, collected through the Asian Pacific Tele-Ophthalmology Society (APTOS) Diabetic Retinopathy Detection competition, and Messidor-2[5] available on the Messidor-2 website. These two datasets were chosen because they are popular datasets for diabetic retinopathy detection in recent years. Both datasets displayed in Figure 3, have the same five labels (ranging from 0 to 4, representing increasing levels of disease severity). Despite these similarities, the main difference between the two datasets lies in their sizes.

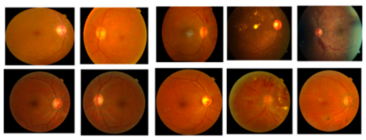

Fig. 3: Examples of datasets

4.2 Collecting experimental results

In the experiments, each experimental unit applied a method consisting of 6 models and 3 preprocessing techniques. Based on the experiments, the classification performance of the models is reported in Table 1, Figure 4. For each type of model, the detection performance is reported in the corresponding first row, and the percentage change compared to the baseline of the same type is reported in the second row. The group observed that the classification performance of the models varied unevenly. Specifically, for deep learning models, all classification performances tended to increase or remain unchanged when applying preprocessing techniques, except for VGG-16 (decreased by 1.53%) and CNN Models (decreased by 1.35%), but these changes were not significant. The highest performance was achieved by VGG-16 with 76% (an increase of 5.56% compared to the baseline). For traditional machine learning models, the classification performance also showed varying changes, with more instances of performance reduction. The most significant performance decrease was seen in the KNN model

[5] https://www.adcis.net/en/third-party/messidor2/

with a decrease of 8.62% when tested on the Messidor-2 dataset. Therefore, it can be concluded that the preprocessing techniques have a positive impact on deep learning models compared to traditional machine learning models.

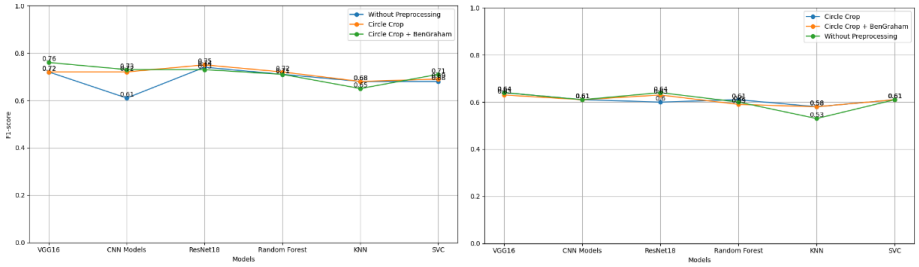

Fig. 4: Performance measures for APTOS dataset and Messidor-2 dataset.

Table 1: Result of the experimental classification models. For each type of model, we report the detection performance on the first line and the rate of decline or change compared to the baseline level of the same type on the second corresponding line.

Model	Dataset	**Without Preprocessing**	**Circle Crop**	**Circle Crop + Ben Graham**
VGG16	APTOS-2019	0.72	0.72 -0.00	0.76 +5.56
	Messidor-2	0.64	**0.63** **-1.56**	0.64 -0.00
CNN	APTOS-2019	0.61	0.72 +18.03	**0.73** **+19.67**
Models	Messidor-2	0.61	0.61 -0.00	0.61 -0.00
ResNet18	APTOS-2019	0.74	0.75 +1.35	0.73 -1.35
	Messidor-2	0.6	0.63 +5	0.64 +6.67
Random	APTOS-2019	0.71	0.72 +1.41	0.71 -0.00
Forest	Messidor-2	0.61	0.59 -3.27	0.6 -1.69
KNN	APTOS-2019	0.68	0.68 -0.00	0.65 -4.41
	Messidor-2	0.58	0.58 -0.00	**0.53** **-8.62**
SVC	APTOS-2019	0.68	0.69 +1.47	**0.71** **+4.41**
	Messidor-2	0.61	0.61 -0.00	0.61 -0.00

4.3 Analysis of experimental results

Hypothesis testing is used to investigate the impact of preprocessing techniques, with both **one-way and two-way ANOVA**. The effectiveness of various pre-

processing methods is assessed using **Tukey's HSD** test as a follow-up to one-way ANOVA with repetitions.

Results of the hypothesis testing for traditional machine learning models We set the hypothesis, with a significance level of 5%, with the null hypothesis being "There is no difference between the F1-scores from the models and the preprocessing techniques," and the alternative hypothesis being "There is at least one difference between the F1-scores from the models and the preprocessing techniques."

Table 2: One-way ANOVA results of machine learning models.

		sum_sq	**df**	**F**	**PR(>F)**
SVM	C(Method)	0.000033	2.0	0.002564	**0.997441**
	Residual	0.019500	3.0	NaN	NaN
KNN	C(Method)	0.002133	2.0	0.186047	**0.839136**
	Residual	0.017200	3.0	NaN	NaN
Random Forest	C(Method)	0.000033	2.0	0.002564	**0.997441**
	Residual	0.019500	3.0	NaN	NaN

Table 2 presents the results of one-way ANOVA analysis on both datasets for each model. Since all the P-values in the tables are greater than 0.05, we cannot reject the null hypothesis: "There is no difference in the average results among the combinations" and reject the alternative hypothesis: "There is a difference in the average results among the combinations." Therefore, the overall conclusion is that there is no significant difference in the average results among the combinations.

Table 3: Two-way ANOVA results with repetitions for machine learning models.

	sum_sq	**df**	**F**	**PR(>F)**
Model	0.00570	2.0	0.541711	0.599592
Preprocessing	0.00040	2.0	0.038015	0.962852
Interaction	0.00200	4.0	0.095037	0.981525
Residual	0.04735	9.0	NaN	NaN

Table 3 describes the results of two-way ANOVA tested on the APTOS and Messidor-2 datasets. Starting with the model factor, as the p-value is 0.599592 $>$ 0.05, the group accepts the null hypothesis that the levels of this factor have the same mean value. Thus, the conclusion is that there is no significant difference among F1-scores of the models. Moving on to the preprocessing technique factor, as the p-value is 0.962852 $>$ 0.05, the group accepts the null hypothesis and concludes that there is no significant difference among F1-scores of the preprocessing techniques. Regarding the interaction aspect between model and preprocessing technique, as the p-value is 0.981525 $>$ 0.05, it can be concluded

that there is no interaction between the two selected main factors. In summary, based on the result table, no single factor or combination has a noticeable impact on the F1-score values in this experiment. The differences in F1-scores observed may come from random factors or other effects.

However, to determine if the average results of any combinations are significantly different from each other, we use the Tukey's HSD method with a 95% confidence level to test the differences between each pair, thus drawing more specific conclusions about the differences in average results among the combinations.

Table 4: Tukey's HSD results of machine learning models.

Method 1	**Method 2**	**meandiff**	**p-adj**	**lower**	**upper**	**reject**
RF_Ben	RF_baseline	0.005	1	-0.2819	0.2919	FALSE
RF_CircleCrop	RF_baseline	0.005	1	-0.2819	0.2919	FALSE
KNN_Ben	KNN_baseline	0.04	0.9995	-0.2469	0.3269	FALSE
KNN_CircleCrop	KNN_baseline	0	1	-0.2869	0.2869	FALSE
SVM_Ben	SVM_baseline	-0.015	1	-0.3019	0.2719	FALSE
SVM_CircleCrop	SVM_baseline	-0.005	1	-0.2919	0.2819	FALSE

The results of the Tukey's HSD method are presented in Table 4. Comparing between the groups under the two factors, model, and preprocessing technique when using machine learning models, we can see that all confidence intervals contain 0. Therefore, we can conclude that there is no significant difference between the groups under the two factors.

Result of the hypothesis test for deep learning models. Similarly, the group sets the hypothesis, with a 5% significance level, where the null hypothesis is "There is no difference in F1-score among the deep learning models and preprocessing techniques," and the alternative hypothesis is "There is at least one difference in F1-score among the deep learning models and preprocessing techniques."

Table 5: One-way ANOVA results of deep learning models

		sum_sq	**df**	**F**	**PR(>F)**
CNN	C(Method)	0.004143	2.0	0.438351	**0.680751**
	Residual	0.014177	3.0	NaN	NaN
VGG-16	C(Method)	0.00070	2.0	0.072664	**0.9315**
	Residual	0.01445	3.0	NaN	NaN
ResNet-18	C(Method)	0.000433	2.0	0.030879	**0.969897**
	Residual	0.021050	3.0	NaN	NaN

Table 5 shows the results of one-way ANOVA analysis on both datasets for each deep learning model. As all the P-values in the tables are greater than 0.05, we can conclude that we cannot reject the null hypothesis: "There is no differ-

ence in the average results among the combinations" and reject the alternative hypothesis: "There is a difference in the average results among the combinations.

Table 6: Two-way ANOVA results for deep learning models.

	sum_sq	**df**	**F**	**PR(>F)**
Method	0.003105	2.0	0.281307	0.761196
Model	0.005314	2.0	0.481350	0.632987
Method:Model	0.002171	4.0	0.098325	0.980343
Residual	0.049677	9.0	NaN	NaN

Table 6 describes the results of the two-way ANOVA analysis conducted on the APTOS and Messidor-2 datasets. From the result table, we observe that first, for the model factor, with a p-value of $0.632987 > 0.05$, the group accepts the null hypothesis and concludes that there is no significant difference in F1-score among the deep learning models. Second, for the preprocessing technique factor, with a p-value of $0.761196 > 0.05$, the group also reaches a similar conclusion, that there is no significant difference in F1-score among the preprocessing techniques. Third, concerning the interaction between the model and preprocessing, with a p-value of $0.980343 > 0.05$, it can be concluded that there is no significant difference in F1-score among the combinations of models and preprocessing techniques. In summary, based on the ANOVA table, no group or combination has a noticeable impact on the F1-score in this experiment. The statistical difference in F1-score may be attributed to random factors or similar effects as the deep learning models.

However, to check whether the average results of any combinations are truly different from each other, we use the Tukey's HSD method with a 95% confidence level to test the differences between each pair, thereby drawing more specific conclusions about the differences in average results among the combinations.

Table 7: Tukey's HSD results for deep learning models

Method 1	**Method 2**	**meandiff**	**p-adj**	**lower**	**upper**	**reject**
BenCNN	CNN_baseline	-0.015	1	-0.3089	0.2789	FALSE
BenResNet18	ResNet18_baseline	-0.0585	0.9944	-0.3524	0.2354	FALSE
BenVGG16	VGG16_baseline	-0.02	1	-0.3139	0.2739	FALSE
Circle_CropCNN	CNN_baseline	-0.02	1	-0.3139	0.2739	FALSE
Circle_CropResNet18	ResNet18_baseline	-0.0525	0.9972	-0.3464	0.2414	FALSE
Circle_CropVGG16	VGG16_baseline	0.005	1	-0.2889	0.2989	FALSE

The results of the Tukey's HSD method for deep learning models are presented in Table 7. Comparing between the groups under the two factors, model, and preprocessing technique when using deep learning models, we can see that all confidence intervals contain 0. Therefore, we can conclude that there is no significant difference between the groups under the two factors.

4.4 Discussion and Future works

Based on the conducted research, it is not yet possible to conclude which method truly differs from the other methods. However, with more data, there is a possibility that the p-values will decrease, and statistically significant differences may be observed. Therefore, in the future, the group plans to increase the number of models and the number of experiments. The group will conduct experiments on more datasets or divide the existing datasets into smaller subsets to obtain more reliable results. As the current experiments were only conducted on two datasets, the obtained results may be subject to random variations. Additionally, the group will refer to more research papers to discover better methods for conducting experiments.

5 Conclusion

This study investigated the impact of preprocessing techniques on diabetic retinopathy image classification using traditional and deep learning models. While our results suggest a trend towards preprocessing benefitting deep learning models more, these differences were statistically insignificant. This indicates that the specific preprocessing methods employed, including Ben Graham's, may not significantly affect classification performance compared to conventional processing for diabetic retinopathy detection. However, our findings do not definitively rule out the value of preprocessing. Future studies with larger datasets may reveal more pronounced effects, particularly for deep learning models. Additionally, exploring alternative preprocessing techniques specifically tailored to diabetic retinopathy features could be beneficial. Therefore, while the current impact of preprocessing seems limited, further research with larger datasets and more targeted techniques is warranted to fully understand its potential role in improving diabetic retinopathy classification accuracy.

References

1. D. Das, S. K. Biswas, S. Bandyopadhyay, A critical review on diagnosis of diabetic retinopathy using machine learning and deep learning, Multimedia Tools and Applications (2022) 1–43.
2. D. Zhang, W. Bu, X. Wu, Diabetic retinopathy classification using deeply supervised resnet, in: 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation, IEEE, 2017, pp. 1–6.
3. N. Sikder, M. S. Chowdhury, A. S. M. Arif, A.-A. Nahid, Early blindness detection based on retinal images using ensemble learning, in: 2019 22nd International Conference on Computer and Information Technology (ICCIT), IEEE, 2019, pp. 1–6.
4. M. S. Patil, S. Chickerur, a. C, a. naik, n. kumari, s. maurya, Effective deep learning data augmentation techniques for diabetic retinopathy classification, Procedia Computer Science (2022).
5. Karthik, Maggie, Sohier Dane, APTOS 2019 Blindness Detection, Kaggle, 2019