
Segmentation of Vessels in Retinal Images

Shiyu Wang

M.S. in Computational Biology
Carnegie Mellon University
shiyuwa2@andrew.cmu.edu

Enhao He

M.S. in Computational Biology
Carnegie Mellon University
enhaoh@andrew.cmu.edu

Jiacheng Ma

M.S. in Computational Biology
Carnegie Mellon University
jiachenm@andrew.cmu.edu

Joe Wang

M.S. in Computational Biology
Carnegie Mellon University
joewang@andrew.cmu.edu

1 Introduction

Problem Statement. Retinal vessel segmentation is a critical task in medical image analysis that involves isolating the blood vessels from retinal images. Despite its importance in diagnosing ocular diseases, such as diabetic retinopathy and glaucoma [Li et al., 2022], accurate segmentation remains challenging for machine learning methods due to the complex vessel structures, low contrast between the vessels and the surrounding tissue, and varying illumination conditions.

Motivation. The motivation for this work stems from the need to enhance early diagnosis and treatment of retinal disorders. Precise vessel segmentation provides key geometric information—such as blood vessel width and tortuosity—that can be correlated with disease progression [Khandouzi et al., 2022]. Automating this process not only accelerates clinical assessments but also reduces the variability and subjectivity inherent in manual segmentation methods, thereby potentially improving patient outcomes.

Overview of Methods. In this work, we evaluate four machine learning methods for retinal vessel segmentation. Our approach includes a **K-means clustering** algorithm that groups pixels based on intensity and color features with contrast enhancement; a **Naive Bayes Classifier** that uses intensity and gradient information to probabilistically differentiate vessel from non-vessel pixels; a **Support Vector Machine (SVM)** that finds an optimal hyperplane using pixel-wise features and both linear and non-linear kernels; and a **Random Forest Classifier** that leverages texture and edge features through an ensemble of decision trees. Together, these methods provide complementary strengths to effectively tackle the segmentation challenge.

Dataset. The experimental analysis is performed on the DRIVE (Digital Retinal Images for Vessel Extraction) dataset, which is widely recognized in the research community for retinal image analysis. The dataset includes high-resolution retinal images along with manually annotated vessel masks, offering a challenging benchmark due to issues like uneven illumination and the subtle contrast between vessels and the background.

2 Data

The DRIVE database comprises a total of 40 high-resolution retinal photographs originally obtained from a diabetic retinopathy screening program in The Netherlands. Each image was acquired with a 45° field of view and is stored as an 8-bit per color plane JPEG file at a resolution of 768 by 584 pixels, with a circular field of view of approximately 540 pixels in diameter. The dataset is evenly divided into a training set and a test set, each containing 20 images.

Since expert ophthalmologists have provided manual segmentations of the vasculature for the **training set**, we only use these **20 images** for training and evaluation. Based on the professional manual segmentations, each pixel’s intensity value can be used not only for segmentation but also for extracting key morphological features such as vessel length, width, tortuosity, branching patterns,

and angles. These continuous features offer detailed insight into the anatomical and pathological characteristics of the retinal vessels, enabling comparative studies on segmentation performance as well as applications in disease diagnosis, screening, and even biometric identification.

3 Methods

The overall **methodology** is implemented in python and consists of multiple stages including data loading, preprocessing, classification, optimization, and evaluation.

As for the **model** used, we mainly focus on the machine learning model, including K-means Clustering, Naive Bayes Classifier, Support Vector Machine and Random Forest Classifier.

To comprehensively **evaluate** our methods, we will use several metrics including accuracy, sensitivity, specificity, F1-score, and the area under the ROC curve (AUC).

3.1 Data Loading and Preprocessing

The DRIVE dataset is loaded by reading the retinal images, manually segmented vessel masks, and background mask data using libraries such as `tifffile`, `PIL`, and `numpy`.

Preprocessing part mainly focuses on the **Image Augmentation**. Our multi-stage enhancement pipeline first amplifies color saturation to emphasize blood vessel structures, then adjusts brightness to improve visibility of subtle features, and finally enhances contrast to sharpen the boundaries between vessels and surrounding tissues.

Besides, we also remove the background from the original image to focus on the non-background pixels by using the background mask in the dataset.

3.2 K-means Clustering

1. Determining the Optimal Cluster Number

According to the structure of normal retinal photographs, the optimal number of clusters should be **4**, including: macula (dark shadow in the center), optic disc (bright shadow on the side), blood vessels, and others. But in computer's language, more detailed exploration is still needed.

2. Optimized K-means Algorithm

To address the challenge of correctly identifying vessel clusters, an optimized version of K-means is used. It merges the clusters with higher AUC values and improves the segmentation performance. To match the biological explanation at the beginning of this section, we merge the top 3 clusters with the highest AUC values to form the final vessel segmentation mask (since $10/3 \sim 4$).

3. Image Augmentation with Optimized K-means

Image augmentation was further applied to enhance the quality of vessel segmentation. By pre-enhancing the input images, the clustering algorithm produces improved segmentation results.

4. Direction-enhanced K-means Clustering

To further improve segmentation, we try to utilize the fact that blood vessels are long and thin by incorporating the **gradient-based features** into the clustering process. In this approach, the original RGB features are augmented with gradient magnitude and gradient direction (obtained via the Sobel filter), forming an enhanced feature vector. Such directional information helps to better delineate vessel boundaries.

3.3 Random Forest Classification

The random forest algorithm, a robust ensemble learning method, can be effectively applied to retinal image segmentation by leveraging its ability to classify pixels into anatomical or pathological regions through supervised learning. In this context, features such as pixel intensity, texture gradients, local binary patterns, and responses to edge-detection filters (e.g., Sobel or Gabor) are extracted from retinal images to capture structural details of blood vessels, lesions, or the optic disc.

Training Process: The gist of Random Forest is to build multiple decision trees and aggregate their predictions. The training process involves the following steps:

1. Feature Extraction: Extract features from the training images, in our cases, the features are simply the RGB values of the pixels.

2. Random Sampling: Randomly select subsets of the training data and features to build each decision tree, ensuring diversity among trees.

3. **Tree Construction:** For each subset, construct a decision tree by recursively splitting the data based on feature thresholds that maximize information gain or minimize Gini impurity.
4. **Prediction Aggregation:** At inference, aggregate predictions from all trees using majority voting to assign the final class label to each pixel.

3.4 Support Vector Machine (SVM)

We implemented a soft-margin Support Vector Machine (SVM) using the Radial Basis Function (RBF) kernel to handle non-linear classification. The dual optimization problem was solved using the `cvxopt` quadratic programming solver, enabling efficient training under soft-margin constraints. To address class imbalance, we manually adjusted sample weights so that both classes contributed more equally to the decision boundary. Support vectors were selected based on non-zero Lagrange multipliers ($\alpha > 10^{-5}$). To convert the SVM's decision scores into probabilities, we applied Platt scaling by fitting a logistic regression model to the outputs. This enabled a probabilistic interpretation of predictions, which is useful for downstream tasks. The final implementation supported end-to-end training, prediction, and probability estimation.

3.5 Naive Bayes Classifier

Preprocessing: CLAHE was applied in the LAB color space to enhance vessel contrast. Pixels outside the Field of View were masked using binary masks. For each valid pixel, RGB intensities were extracted as features. Labels were assigned as 1 (vessel) or 0 (background) using manual annotations. To handle class imbalance, all vessel pixels were retained, and an equal number of background pixels were randomly sampled.

Model: A custom **Gaussian Naive Bayes (GNB)** was implemented. For each class, mean, variance, and prior were computed assuming pixel independence and normal distribution. During prediction, posterior probabilities were calculated, and the class with the highest posterior was assigned. 80% of the balanced data was used for training, 20% for testing (random seed fixed).

4 Results

4.1 K-means Clustering

4.1.1 Optimal Cluster Number Determination

After evaluating different numbers of clusters (ranging from 2 to 10) by computing the AUC value for each image and taking the mean value, we found that the AUC value and the details of the vessels are positively correlated with the number of clusters, as shown in the Supplementary Figures S1, S2, S3, S4, and S5.

However, when we set the `n_clusters = 10`, the vessel pixels are fragmented into multiple clusters. Thus, a single cluster in each image is not sufficient to evaluate the segmentation performance. Therefore, we decided to set `n_clusters = 10` as an optimal compromise with an **optimized K-means algorithm**.

4.1.2 Optimized K-means Results

Figure 1 depict the best segmentation examples using this approach, including the original image, the ground truth, the segmentation result with AUC values, and the segmentation overlay. The second and third best examples are shown in the Supplementary Figures S6a, and S6b.

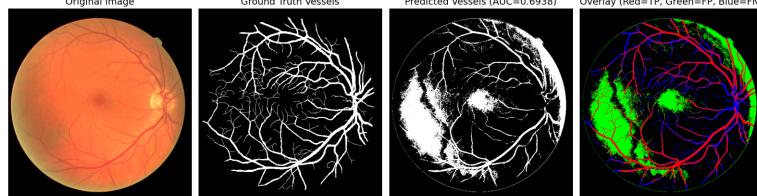


Figure 1: Optimized K-means: the first best example (AUC=0.6938)

As we can see, the biggest problem is that there are a lot of false positive pixels (FP, Green) in the segmentation results, which means that their colors are difficult to distinguish from blood vessels.

4.1.3 Image Augmentation with Optimized K-means Results

The results corresponding to the best examples above after applying image augmentation are shown in Figure 2 below. The other two examples are shown in the Supplementary Figures S7a, and S7b.

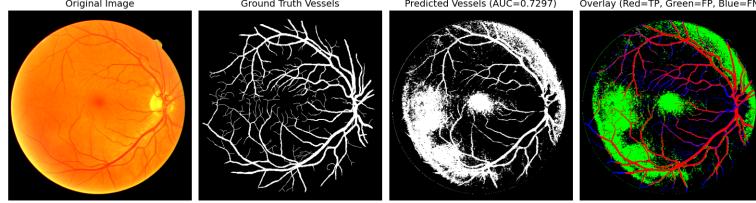


Figure 2: Optimized K-means with augmentation: the first best example ($AUC=0.6938 \rightarrow 0.7297$)

Apparently, the results are much better now. The false positive pixels (FP, Green) are significantly reduced, and the segmentation results are more accurate.

4.1.4 Direction-enhanced K-means Clustering Results

Such directional information helps to better delineate vessel boundaries, as illustrated in the following figures (Figure 3).

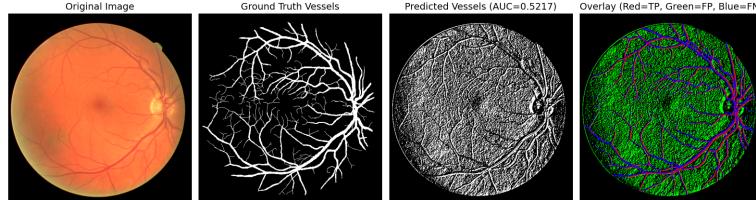


Figure 3: Direction-enhanced K-means ($AUC=0.5217$)

The overall performance of the direction-enhanced K-means is not as good as previous methods, but it still provides a different perspective on the segmentation task by showing the potential of incorporating gradient features. Also, the results are in a different style, which is quite interesting. Besides, we can also see the small improvement of the AUC value after applying image augmentation (see Supplementary Figure S8), which further verify the effectiveness of this technique.

4.1.5 Performance Evaluation and Comparison

Table 1 summarizes the performance comparison between the classic K-means and the optimized K-means methods, both before and after applying image augmentation. (Not include the direction-enhanced K-means is because it is just a try and not the main focus of this project.)

Table 1: Performance Comparison: Before (Black) and After (Red) Image Augmentation

Metric	Classic K-means	Optimized K-means
Accuracy	0.8062 → 0.8404	0.6193 → 0.6938
Sensitivity	0.2228 → 0.2805	0.5862 → 0.6138
Specificity	0.8898 → 0.9200	0.6247 → 0.7057
F1 Score	0.2270 → 0.3115	0.2802 → 0.3393
AUC value	0.5563 → 0.6003	0.6054 → 0.6598

Through comparison of evaluation metrics, it is evident that image augmentation has a positive impact on both K-means algorithms.

Although classic K-means shows better performance in accuracy and specificity, the optimized K-means algorithm demonstrates significant advantages in key metrics such as sensitivity and AUC value, especially when combined with image augmentation techniques, providing more effective discrimination between vessel and non-vessel regions.

This suggests that the optimized K-means algorithm has stronger recognition capabilities in vessel segmentation applications, making it particularly suitable for medical image analysis tasks that require higher detection rates.

4.2 Random Forest Classification

Through the whole training process, totally 10 images, $10 \times 584 \times 565 = 3299600$ pixels are trained upon. The pixels are separated into 3 classes: 0 for masked boundary, 1 for background and 2 for blood vessels. A typical prediction result given by the classifier is shown below in Figure 4 and Table 2, with an AUC of approximately **0.6176**.

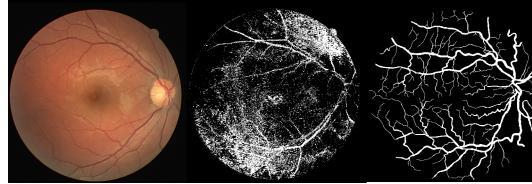


Figure 4: One example (1st image is the original image, 2nd image is the manually labeled image for training, 3rd image is the predicted image by the random forest algorithm).

Table 2: Metrics for the segmentation Task

	precision	recall	f1-score	support
background	0.91	0.80 (specificity)	0.85	198755
retinal	0.24	0.44 (sensitivity)	0.31	29073
accuracy			0.75	227828
weighted average	0.82	0.75	0.78	227828

4.3 Support Vector Machine (SVM)

To evaluate the performance of our custom SVM, we also trained a Support Vector Classifier (SVC) using Scikit-learn as a baseline for comparison. The result is shown below in Table 3.

Table 3: Comparison between Custom SVM and Scikit-learn SVC

Metric	Custom SVM (from scratch)	SVC (Scikit-learn)
Precision	0.22	0.21
Recall	0.11	0.62
F1 Score	0.14	0.31
Accuracy	0.84	0.65
AUC	0.60	0.69

Despite achieving higher accuracy, the custom SVM significantly underperformed in recall, F1 score, and AUC compared to Scikit-learn's SVC. This can be attributed to several factors. First, our model was trained on only 15% of the dataset due to the computational complexity of the SVM algorithm and the high cost of solving the dual optimization problem, especially with the RBF kernel. This limited training set size reduced the model's ability to generalize, particularly on the minority class. Second, our implementation did not include solver optimizations such as SMO, kernel caching, or shrinking heuristics, all of which are integrated into Scikit-learn's efficient SVC. Lastly, we used fixed hyperparameters without tuning (e.g., C and γ), which likely constrained the model's classification performance.

4.4 Naive Bayes Classifier

This study demonstrates the effectiveness of a simple Gaussian Naive Bayes model for retinal vessel segmentation. **Key Observations:** CLAHE improved contrast, and class balancing prevented bias. GNB achieved 67% accuracy and **0.72 ROC AUC**.

Limitations: Struggles with thin vessels and ignores spatial context. Gaussian assumption may not fit real distributions. The total results and an example of the segmentation are shown below in Table 4 and Figure 5 respectively. These results indicate balanced performance across both classes.

Table 4: Classification Report

Class	Precision	Recall	F1-Score	Support
0	0.67	0.66	0.66	113,484
1	0.67	0.67	0.67	114,282
Accuracy			0.67	
Macro Avg	0.67	0.67	0.67	227,766
Weighted Avg	0.67	0.67	0.67	227,766

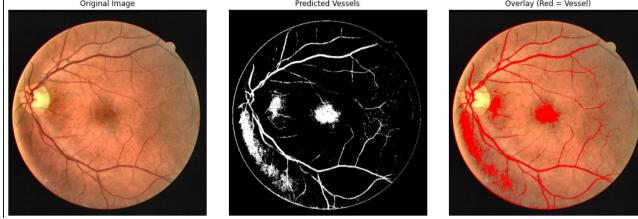


Figure 5: One example (1st is the original image, 2nd is the predicted image, 3rd is the segmentation overlay).

5 Discussions

In this work, our results demonstrate that image augmentation and designed feature enhancements can significantly improve retinal vessel segmentation. In particular, the **optimized K-means clustering** approach yielded notable improvements in sensitivity and AUC compared to the classic K-means variant, even **reaching the level of supervised learning methods**. The integration of directional features offered additional insights into vessel boundary delineation, although its overall performance was comparatively modest.

Furthermore, while methods such as **Random Forests**, **SVM**, and **Gaussian Naive Bayes** each exhibited unique strengths in pixel classification, our findings indicate that careful handling of feature selection, model optimization, and training data balance is essential for robust segmentation performance, which **has a chance to make the performance of the unsupervised learning model better than that of the supervised learning model**.

Performance comparison by AUC: Optimized K-means (0.66), Random Forest (0.62), SVM (0.60), and Naive Bayes (0.72), with Naive Bayes achieving the best performance.

Future Work:

- 1. Dataset Expansion and Diversification:** Evaluating the proposed approaches on larger and more diverse datasets to enhance the generalizability and robustness of the segmentation models.
- 2. Advanced Feature Engineering:** Integrating modern deep learning-based feature extraction techniques and multi-scale analysis to better capture the intricate morphology of retinal vessels.
- 3. Hyperparameter and Solver Optimization:** Conducting systematic hyperparameter tuning (e.g., grid search, Bayesian optimization) and deploying more efficient solver strategies (such as SMO or kernel caching for SVMs) to improve overall model performance.
- 4. Post-Processing Enhancements:** Investigating post-processing approaches such as morphological filtering or conditional random fields (CRF) to further reduce false positives and refine vessel continuity.

6 Author Contributions

Shiyu Wang: Topic Selection, Proposal, K-means Clustering, Image Augmentation, Final Report.

Enhao He: Proposal (methods part), Support Vector Machine, Final Report (SVM related part).

Jiacheng Ma: Random Forest Classifier, Final Report (RF related part).

Joe Wang: Naive Bayes Classifier, Final Report (NBC related part).

The contributions of each author to slides are equal.

7 Supplementary Material

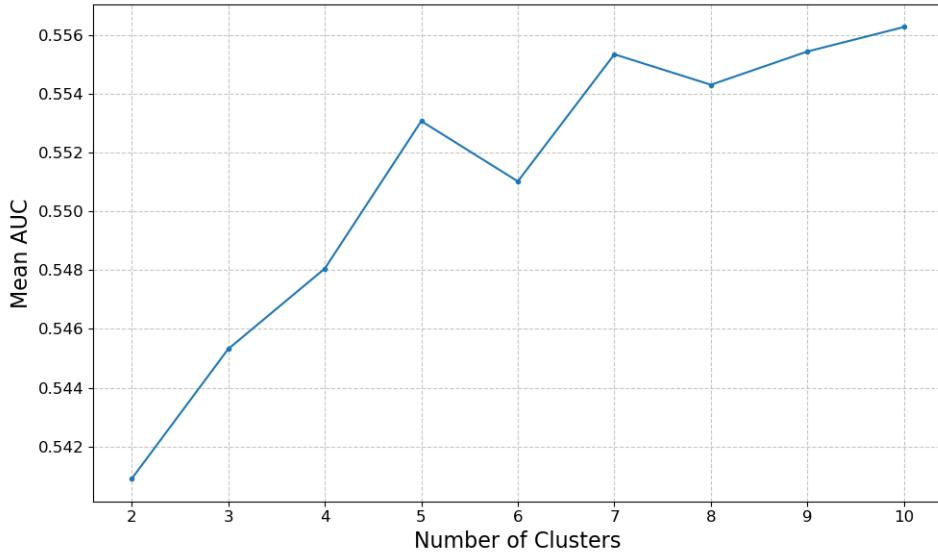


Figure S1: Mean AUC values for different number of clusters.

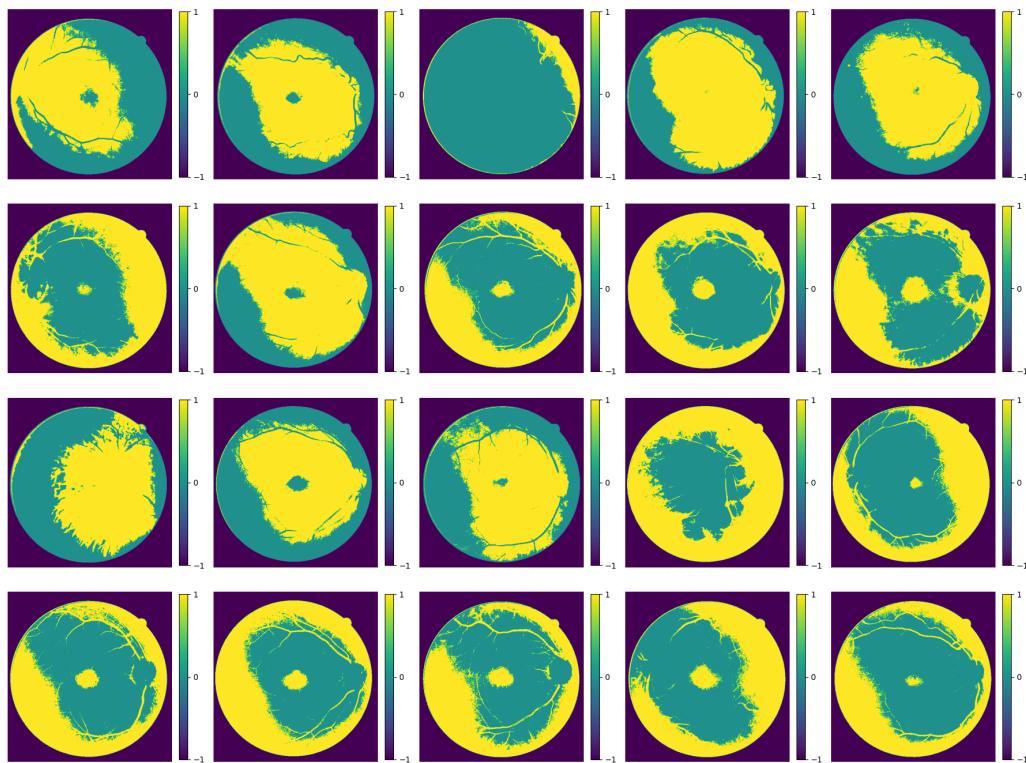


Figure S2: Segmentation results ($n_clusters=2$).

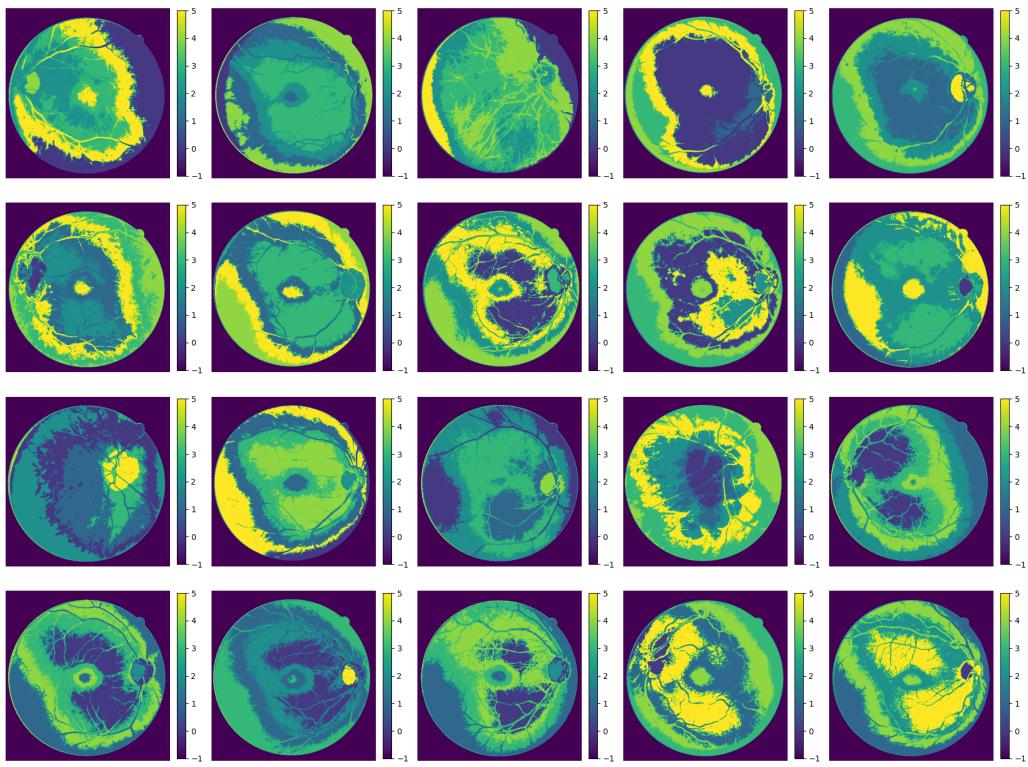


Figure S3: Segmentation results ($n_clusters=6$).

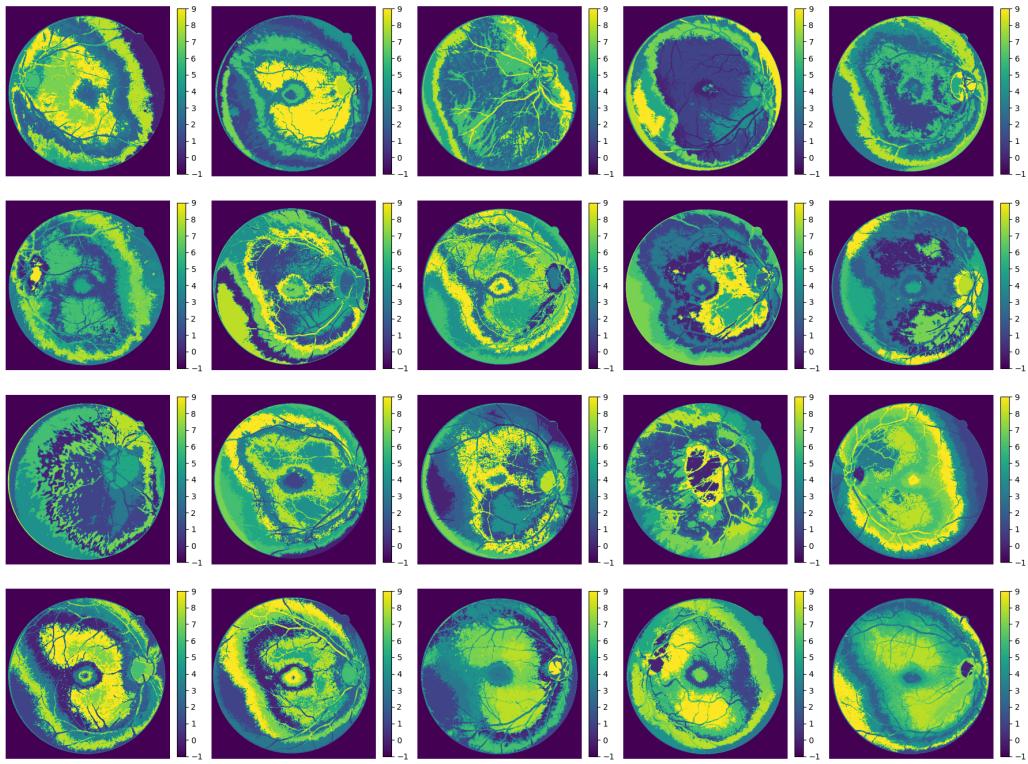


Figure S4: Segmentation results ($n_clusters=10$).

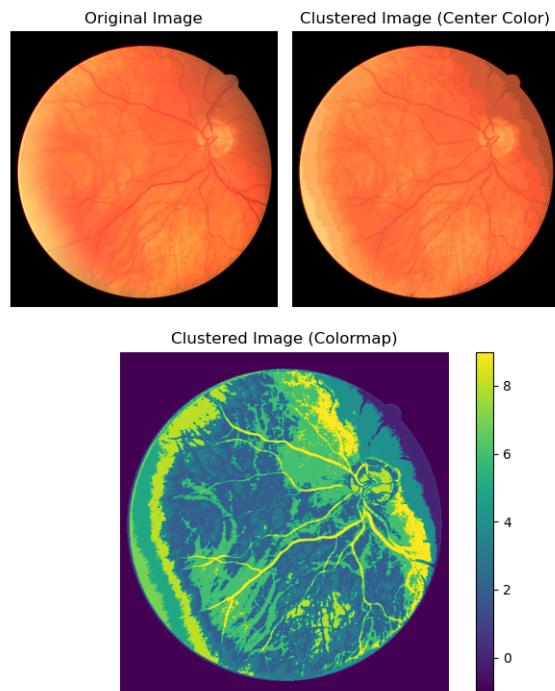
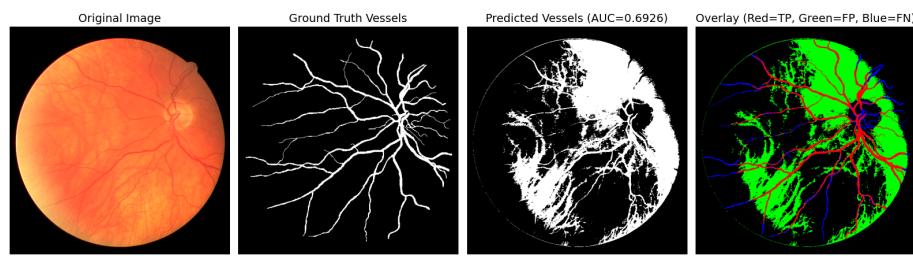
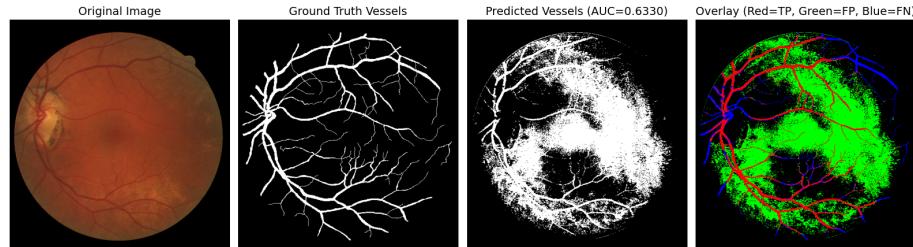


Figure S5: Specific case ($n_{\text{clusters}}=10$).

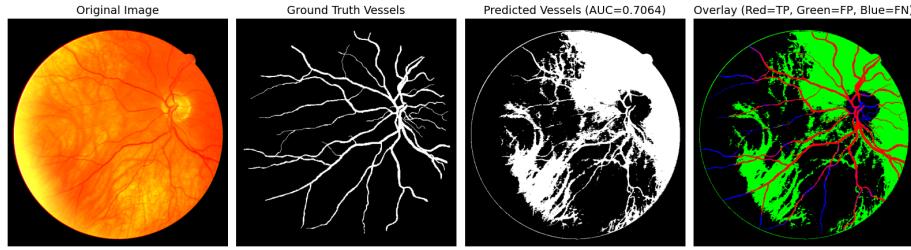


(a) The second best example ($AUC=0.6926$)

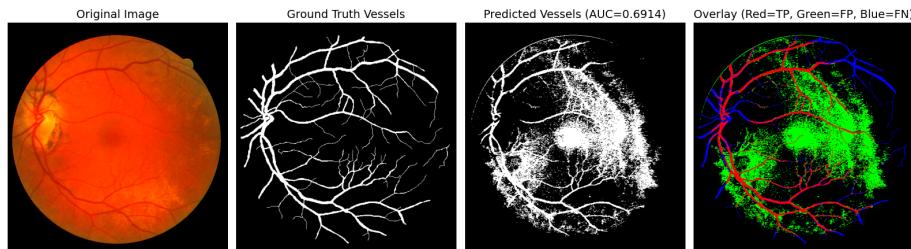


(b) The third best example ($AUC=0.6330$)

Figure S6: Optimized K-means: Top 3 Examples



(a) The second best example ($AUC=0.6926 \rightarrow 0.7064$)



(b) The third best example ($AUC=0.6330 \rightarrow 0.6914$)

Figure S7: Optimized K-means with Augmentation: Corresponding 3 Examples

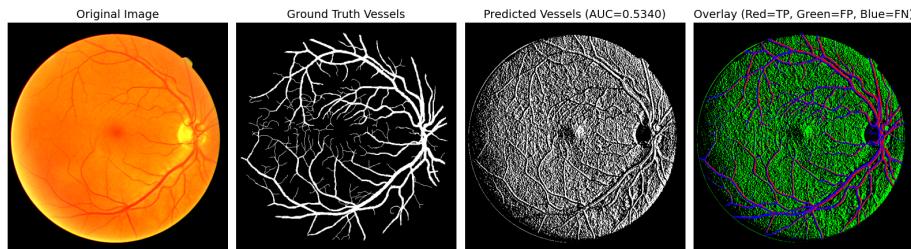


Figure S8: Direction-enhanced K-means with augmentation ($AUC=0.5217 \rightarrow 0.5340$)

References

W. Li, Y. Xiao, H. Hu, et al. Retinal vessel segmentation based on b-cosfire filters in fundus images. *Frontiers in Public Health*, 10:914973, 2022. doi: 10.3389/fpubh.2022.914973.

A. Khandouzi, A. Ariaifar, Z. Mashayekhpour, M. Pazira, and Y. Baleghi. Retinal vessel segmentation, a review of classic and deep methods. *Annals of Biomedical Engineering*, 50(10):1292–1314, 2022. doi: 10.1007/s10439-022-03058-0.