Retinal Vessel Segmentation and Classification using Fully Convolutional Neural Networks

Except where reference is made to the work of others, the work described in this project
is my own or was done in collaboration with my advisory committee. Further,
the content of this project is truthful in regards to my own work
and the portrayal of others' work. This project does not
include proprietary or classified information.

Michael DeMauro

Certificate of Approval:

Michael J. Reale
Assistant Professor
Department of Computer and
Information Science

Chris Urban, Chair
Lecturer
Department of Computer and
Information Science

Jorge Novillo
Professor
Department of Computer and
Information Science

Andrew Wolfe
Interim Dean
College of Engineering

Retinal Vessel Segmentation and Classification using Fully Convolutional Neural Networks

Michael DeMauro

A Master's Project

Submitted to the
Graduate Faculty of the
State University of New York Polytechnic Institute
in Partial Fulfillment of the
Requirements for the
Degree of

Master of Science

Utica, New York
May 6, 2017

Retinal Vessel Segmentation and Classification using Fully Convolutional Neural Networks

Michael DeMauro

Signature of Author

Date of Graduation

Master's Project Abstract

Retinal Vessel Segmentation and Classification using Fully Convolutional Neural Networks

Michael DeMauro

Master of Science, May 6, 2017
(B.S., Rochester Institute of Technology, 2009)

41 Typed Pages

Directed by Dr. Michael J. Reale

Automated separation and virtual reconstruction of vasculature from retinal fundus images is a difficult task. While recent studies have shown fully convolutional neural networks to be effective at producing accurate binary vessel segmentations, to the author's knowledge none have regarded their use as classifiers in this context. This project explores variations on a learning model for the classification of retinal vessels as arteries and veins. A key observation is the requirement and challenge of identifying areas of overlapping vessels. The results show a fair ability to correctly classify large areas of contiguous vasculature. The encountered problem of localized discontinuity is discussed.

Acknowledgments

This work is dedicated to Tela.

Thank you to all my professors and educators, past and present. Knowledge and inspiration are the best gifts. Thank you to my friends who have encouraged me. Most of all, thank you to my parents and family who have always given their love and support. My utmost appreciation goes to my project advisor and collaborator, Dr. Michael J. Reale. I would also like to thank Touseef Qureshi and Kevis-Kokitsi Maninis for their help and friendliness.

Style manual or journal used Journal of Approximation Theory (together with the style known as “sunypolyms”). Bibliography follows van Leunen’s *A Handbook for Scholars*.

Computer software used The document preparation package TeX (specifically LaTeX2e) together with the style-file `sunypolyms.sty`.

Table of Contents

List of Figures

List of Tables

Chapter 1

introduction

The segmentation of retinal blood vessels has been a non-trivial computer vision problem for many years. Various motivations for approaching the problem have given rise to slightly different methods. A common goal is to assist in the diagnosis of eye-related pathologies, such as diabetic retinopathy[14]. Many implementations seek to identify the vascular regions from the background of the fundus image as a whole. In addition to this, more recent studies have aimed to further classify the foreground vessels into arteries and veins[2, 3, 5, 11, 12, 16, 17]. This paper describes an experiment to investigate the potential of using deep convolutional neural networks for simultaneous segmentation and classification of retinal vessels from fundus images.

The motivation for this project also stems from the problem of determining and separating the individual vascular structures visible in retinal images. Having only the identification of vein or artery class per pixel does not fully address this. Knowing the small subset of pixels of which one vessel occludes another in addition to the surrounding pixel classification would provide the information necessary to connect the discontinued segments of the trees. A method is devised which can potentially assist in virtual reconstruction of the full directed acyclic graphs of the veins and arteries given sufficiently accurate pixel classification.

Chapter 2

Related Work

2.1 Vessel Classification

Various methods have been applied to identify blood vessel types from retinal images. Some approaches can be described as topological or graph theoretic wherein a set of vessel segments are considered as nodes and the edges are initialized using pixel adjacency of the segments. A commonality among all of these is the necessity to manually resolve ambiguities in the optic nerve area. In the experiments by Hu et al.[3] and Joshi et al.[5], the pixels within this area of the binary segmentation were set to zero intensity. Estrada et al.[2] instead modified the resulting graph to prevent cycles. This topic is a trouble spot and should be noted, but it is not directly examined in this project.

While several ground truth datasets exist for retinal vascular segmentation, there are unfortunately few ground truth datasets for their classification. From the DRIVE[1] images, Hu, Abramoff, and Garvin enlisted experts to establish RITE [2], a dense classification standard including unique identification of overlapping vessel areas[3, 14]. They also designed and implemented an image processing pipeline to estimate per-pixel classifications. They first create a skeleton from their binary segmentation. Spurious skeleton pieces need to be removed via length filtering. The skeleton is then further segmented into nodes of a graph which are connected by edges when endpoints of the skeleton segments are within close proximity of one another.

Traversing the graph, nodes are labeled as either vein or artery to yield multiple directed acyclic graphs each rooted at the optic disc. In a local neighborhood of connected nodes, certain label configurations will be more plausible than others. The probability of

[1] DRIVE dataset available here: `http://www.isi.uu.nl/Research/Databases/DRIVE`

[2] RITE dataset available here: `https://medicine.uiowa.edu/eye/rite-dataset`

each configuration is calculated as a function of vessel end point direction, width, intensity profile, and orientation. These probabilities are then inverted to a cost value and summed. Independently choosing the most likely label configurations for each neighborhood does not guarantee that the global tree structures are valid. For this reason, acyclicity is enforced and global property costs are added to the local cost sum. The total cost sum is then minimized.

Estrada et al.[2] similarly built a graph of segments but instead utilized a tracking algorithm to further segment the binary segmentation into overlapping tracks. During the iterative tracking of a segment, a two dimensional Von-Mises kernel is included to favor straight paths and allow for bifurcations. This also effectively prevents tracks from making 90-degree or greater turns. To seed the tracks, points were chosen in order of highest to lowest value from a LoG-Gabor response filter of the original image until all pixels of the binary segmentation were assigned to at least one track. The unlabeled graph of segments provides the basis for a space of possible labeled trees. Several algorithms are invented to efficiently explore the space and approximate optimal labels with impressive accuracy.

2.2 Fully Convolutional Neural Networks

Long et al. introduce the fully convolutional neural network[6] They extend successful deep learning classification models[13] to dense pixel-wise prediction via deconvolution. Dense prediction alone can yield fine resolution output, but does not fully benefit from the non-local structures of objects in the down-sampled layers. In order to factor in a hierarchy of spatial structure, *skip layers* are created in parallel to the down-sampling. These are respectively up-sampled and eventually concatenated prior to scoring. The scoring layer consists of a 1-pixel-kernel convolution. This series of deconvolution layers combined with per-pixel scoring replaces the typical fully-connected layer and outputs a channel per class. The most likely class per pixel provides the final inference image.

2.3 CNNs for Vessel Segmentation

Maninis et al. developed convolutional neural networks for the purpose of segmenting retinal vessels and optic disc areas from fundus images[7]. Their objective was to score the likelihood of each pixel belonging to the vessel class versus non-vessel class. They fine-tuned multiple models, each initialized from the VGG[13] model weights. Two similar model structures were defined one for the vessels which scored feature maps from depths 1 through 4 and one for the optic disc which scored feature maps from depths 2 through 5. In a similar manner to Long et al.[6], the down-sampled feature maps are respectively up-sampled using deconvolution layers with learned weights. Prior to each pooling step, a split is made to a *skip layer*. The first split is made at input resolution and concatenated with each fully up-sampled set of feature maps. The scoring layer outputs a single class since it is a binary segmentation. This CNN model shows high precision and recall for segmentation in comparison to many other techniques.

Chapter 3

Approach

Classifying foreground pixels into three classes - vein, artery, and overlap - can potentially provide the information required to easily separate continuous vascular tree structures in retinal fundus images. Taking the union of the overlap-classified pixels with either artery or vein sets of pixels yields the expected connected components of the vascular trees. Figure 3.1 illustrates this using the RITE ground truth. If a multi-class learning model for whole retinal images can be developed and provide dense and accurate predictions, the process of separating vascular trees will be significantly simplified.

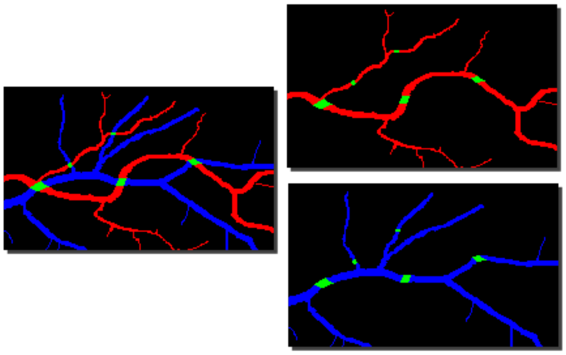


Figure 3.1: Separating vein and artery with union of overlap.

In order to determine if fully convolutional neural networks can be well-suited to classify vessel types in fundus images, an existing model designed for retinal vessel segmentation is fine-tuned and tested. The base model is adapted to output multiple vessel class scores. This provides an idea of how well the model can segment each individual class from the background. Additionally, the vessel scores are masked by the ground truth binary segmentation (i.e. union of all vessels and overlap) to determine the efficacy of the model for classification alone. Various scenarios regarding ground truth dataset usage and preprocessing techniques are also explored.

Multiple instances of two-class models are trained to address several concerns. It would be simpler to allow the model to make a prediction for both vessel channels at an overlap instead of creating this prediction as a separate channel. The overlap ground truth is unioned with each artery and vein classes for these models. The occluding vessels of the overlaps are, on the visible aspect, actually either vein or artery. A control model is devised to measure if this loss of knowledge has any significant effect on training the classifier. The additional preprocessing techniques of rotation and mirroring are applied to extend the training data.

3.1 Model Architecture

The model architecture used in the experiments retains the first four of five sets of convolution and activation layers from the DRIU model; the fourth max pooling layer is excluded. Figure 3.2 illustrates the adapted DRIU architecture. The convolutions, activations, and deconvolution that follow the fourth max pooling are thus discarded as well. For all of these intermediate layers, the number of output channels and kernel sizes are unchanged. The number of scoring output channels are modified according to the specific classification purpose of each model.

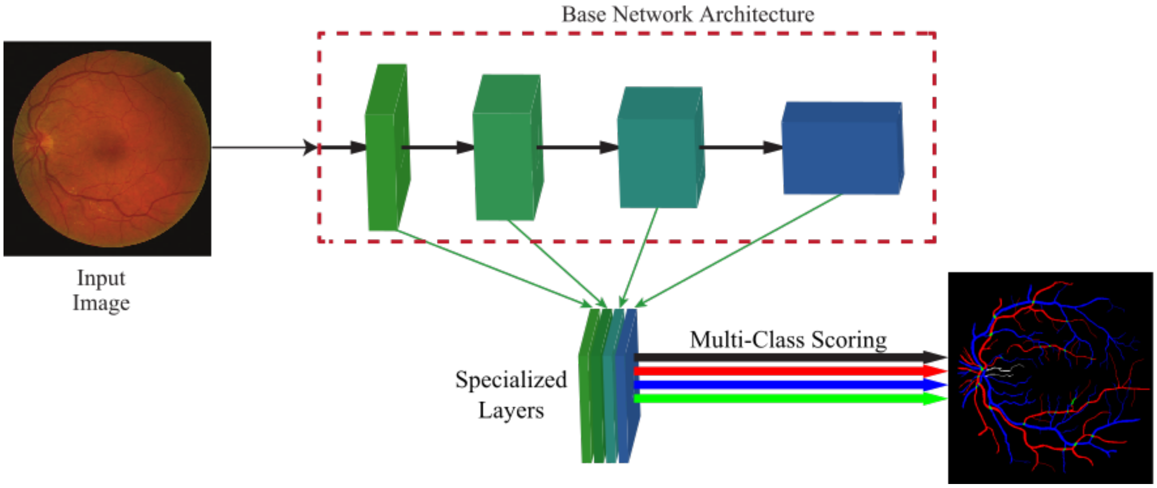


Figure 3.2: The general model architecture from DRIU[7] adapted for multi-class segmentation. The first 4 side layers are utilized for vessel segmentation.

3.2 Data

The models are trained and tested using the original DRIVE color images for inputs[14]. This collection of images was initially created with the direct purpose of comparing segmentation algorithms for retinal vessels. The image set was sourced from a screening program for diabetic retinopathy. The field of view for each image is 45 degrees. There are 40 images in total which are split into 20 training images and 20 test images. Additionally, the original team along with expert ophthalmologists manually created ground truth binary segmentations for the foreground vessels.

The RITE dataset images are used for the ground truth data[3]. These images are created by another team along with experts. They took the original binary segmentations and manually classified each foreground pixel into the set of labels: artery (red), vein (blue), overlap (green), unknown (white). See Example A.7 for an example of a RITE ground truth image. Pixels with unknown label are not included during training or in calculation of classification statistics.

One model (referred to as *2-Class Model - Lincoln*) is trained using a similar ground truth dataset from Qureshi et al. at the University of Lincoln along with experts from the Sunderland Eye Infirmary[10]. This team also classified the ground truth binary segmentation, but instead classified the overlapping areas as either artery or vein. See Example A.11 for an example of these ground truth images. Yellow and white pixels are considered indistinguishable and are not included during training or in calculation of classification statistics.

Due to a limited number of training images, the same 20 are repeatedly cycled through. A single image at a time is input per batch then the model is stepped forward through inference, calculates the cross entropy loss, and then backward propagation is run.

3.2.1 Preprocessing

Input images are preprocessed using the same techniques in DRIU[7]. A MATLAB script pads the field-of-view border by iteratively convolving a linear filter which is a diamond-shaped structuring element. The result reduces filter response scores during vessel segmentation which reduces the visual artifact created at the circular border of the fundus image. A visible difference in results is shown in Figure 3.3.

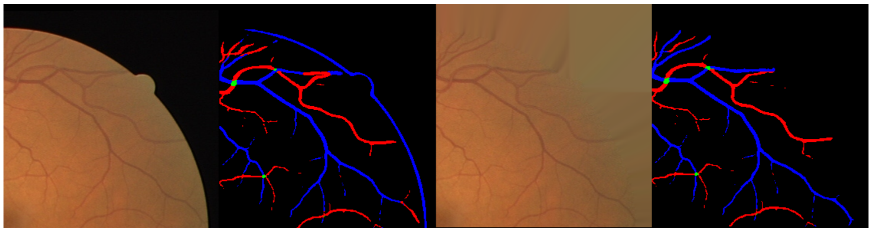


Figure 3.3: From left to right: Original image; Prediction using original image; Padded image; Prediction using padded image.

Subtraction of the mean value across the population of training images is also applied for each color channel. The three color channels (red, green, and blue) are then transposed from the fourth to the second dimension and set as input to the model. Width and height are thus moved to the third and fourth dimensions.

Chapter 4

Experiments and Results

4.1 Software Architecture

To train each model, the Caffe tool was executed from a Slurm[1] script and run on a single shared node[2] with an NVIDIA Tesla K80 GPU, utilizing CUDA[8]. To streamline the process of testing models and generating results, a framework was designed and implemented in Python 2.7. This allowed for easier deployment when training on a multi-user clustered server with virtual environments and a common Caffe installation. The overall process flow is illustrated in Figure 4.1.

4.1.1 Training

The models are implemented and run using the Caffe[3] API [4]. In order to accommodate classification of each pixel, a DenseLabelInputLayer class is implemented which inherits from Caffe's Layer class. It loads each input and label image, performs mean subtraction, transposes, and copies the data into the network input layer during training. Importantly, each model can subclass this layer to provide alternative means to loading images, interpreting labels, and transforming data specific to the requirements of the model. Each model's protocol buffer file specifies the DenseLabelInputLayer class.

[1] More information on Slurm available here: `https://slurm.schedmd.com`

[2] This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number OCI-1053575. Specifically, it used the Bridges system, which is supported by NSF award number ACI-1445606, at the Pittsburgh Supercomputing Center (PSC)[9, 15]

[3] Release: rc5. More information available at: `http://caffe.berkeleyvision.org`

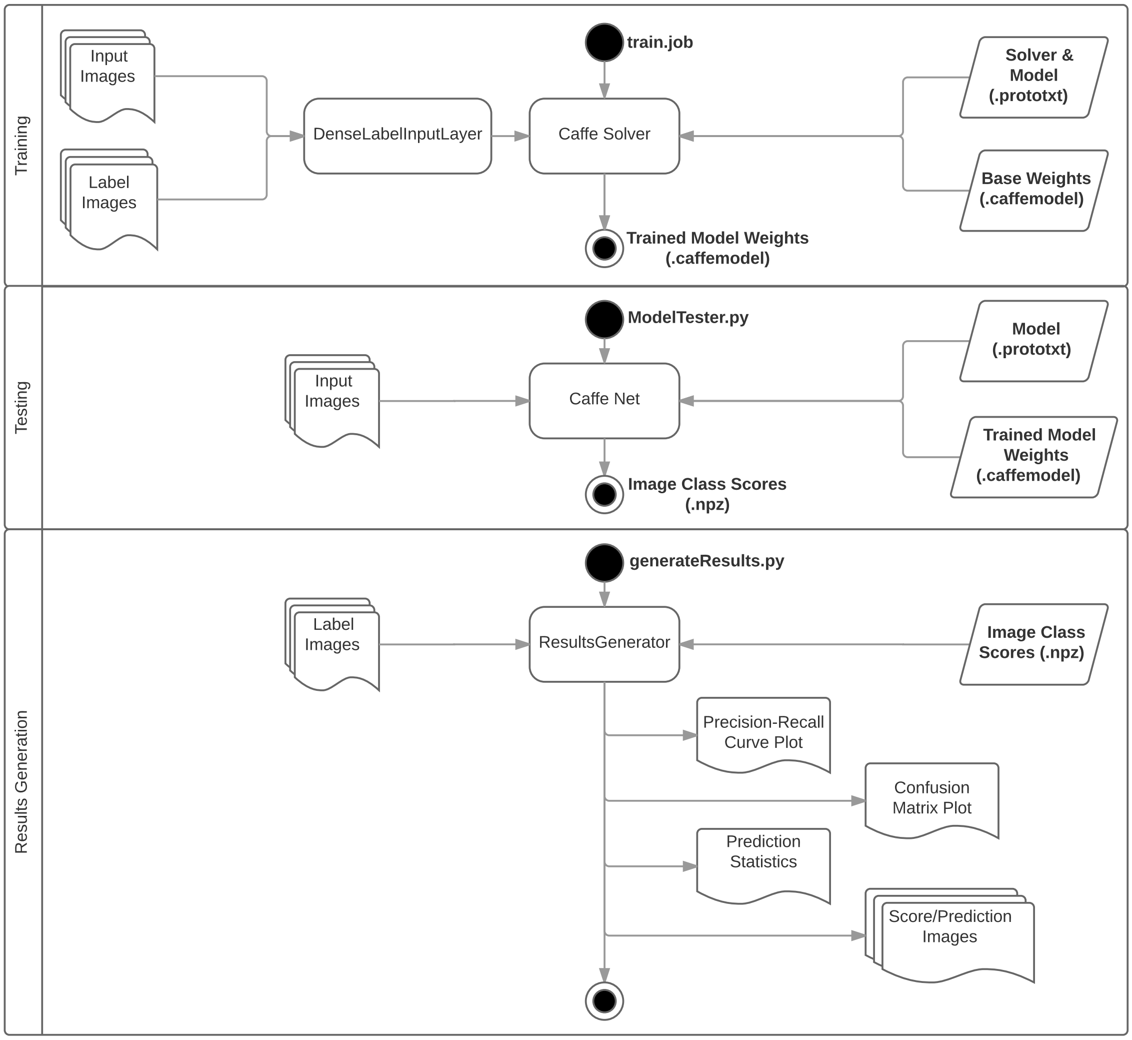


Figure 4.1: The overall process flow for training, testing, and evaluating a model.

4.1.2 Testing

The ModelTester loads the model network and images then runs each through one at a time. A Python dictionary is created to map each image name to its class scores which are copied from the specified network scoring layer. This is then saved to a Numpy file[4]. The initial goal for the testing phase was to reuse DenseLabelInputLayer and create a custom OutputLayer as well. The image names could not be propagated through the network to

[4] More information on the Numpy file format is available here: `https://docs.scipy.org/doc/numpy/neps/npy-format.html`

an OutputLayer, so the ModelTester class was created to wrap the entire process. This unfortunately replicates the logic of DenseLabelInputLayer (and the Caffe executable tool) but also is able to synchronize the image names for output. In hindsight, it would have been simpler to just replicate the loading of image names in the OutputLayer. Refactoring this is a clear next step for simplifying the architecture and increasing re-usability.

4.1.3 Results Generation

The ResultsGenerator class loads the score files and label images and contains methods to output various statistics and renderings. A LabelDataLoader object is required as input and can be specified by each model's invoking script. This allows for modularity of label data loading which may be useful to load datasets other than DRIVE. A DataTransformer object is also required as input and allows for each model to specify the logic for transforming the score and label data to appropriate format for use with the SciKit-Learn[5] metrics package.

4.2 Training Parameters

The learning rate parameters for the model layers and solver remain the same as in DRIU. Model weights are initialized using the base DRIU model and then fine-tuned for classification purposes. Stochastic gradient descent is used to solve for the learned weights.

$$V_{t+1} = \mu V_t - \alpha \nabla L(W_t)$$

$$W_{t+1} = W_t + V_{t+1}$$

This method updates weights using a linear combination of the negative weight gradient ($\nabla L(W)$) and the previous iteration's update (V_t). Momentum (μ) is the coefficient of the previous iteration's weight update and is set to 0.9. The learning rate (α) is the gradient coefficient and is set to 10^{-8}. Each model is trained for 20,000 iterations, decreasing the

[5] More information about SciKit-Learn can be found here: `http://scikit-learn.org`

learning rate every 10,000 iterations using a gamma of 0.1 ($\alpha' = \alpha\gamma$). Because there are 20 training images which are not batched together, an epoch is equivalent to 20 iterations.

The loss is calculated as a sum of binomial cross entropies. Each binomial distribution represents the probability of a pixel belonging to a specific class or not. The sigmoid of each score (Q) is used and the ground truth (P) is a binary value. A score exists for each pixel (x, y) and class (c).

$$Loss = -\sum_{i=(x,y,c)} P_i log(Q_i) + (1 - P_i) log(1 - Q_i)$$

4.3 Experiments

4.3.1 Four-Class Model

The four-class model is based on the DRIU model for vessel segmentation on DRIVE images. Because the original model is a binary segmentation, additional channels are added to the scoring layer, one for each intended class prediction. Four class channels are set for the corresponding pixel colors in the ground truth: background (black), vein (blue), artery (red), and overlap (green). The unknown (white) pixels in the ground truth are ignored.

The weights from the base model's convolution and deconvolution layers are transplanted one-to-one. The scoring layer weights are copied to the vein and artery channels only. The background and overlap channel weights are initialized to random values using a Gaussian distribution with a standard deviation of 0.001.

4.3.2 Three-Class Model

Training the model while including the background class factors into loss calculation. The back-propagation of its differential may also have an effect on the learning of layer weights. The 3-Class model is trained for the purpose of evaluating this impact by excluding the background channel. All other model parameters remain the same as in the 4-Class model.

4.3.3 Two-Class Models

Three 2-class models are trained. Among these, only the input image sets are varied. The *normal* model retains the same input images as the 4-class model with the difference of adding the overlap ground truth to both vein and artery classes. The *overlap control* model is trained using the University of Lincoln dataset[10]. The *rotation and mirroring* model is trained on images which were generated by rotate-and-mirror preprocessing. The base-model weights are copied in the same manner as the 4-class model. There are no background and overlap channel weights in these models to initialize. All learning and layer parameters remain the same as used in the 4-class.

Overlap Control Model

The occluding vessels of the overlaps are, on the visible aspect, actually either vein or artery. A control is needed to determine if the slight ambiguity has any significant effect on training the classifier. Another two-class model using the same exact parameters is trained on a ground truth dataset provided by Qureshi et al. from the University of Lincoln[10]. The manually labeled segmentations of this dataset lack the overlap class and instead have been labeled either vein or artery at these locations. See Example A.11 for an example ground truth image.

Data Augmentation: Rotation and Mirroring

The limited training data available poses the potential problem of over-fitting. Rotating and mirroring the retinal images can potentially increase classification accuracy by providing additional training data. Each image is rotated 90 degrees 3 times to produce 3 additional images. Each original and rotated image is also mirrored horizontally. This produces 8 times the original number of images, making a total of 160 training images.

4.4 Test Results

After testing, the sigmoid for each of the class scores are output. The scores for all DRIVE test images are concatenated for analysis. Example score renderings are available in Appendix A.1. A binary threshold is varied independently across the range of each class score and the precision-recall curve is calculated, shown in Figure 4.2 (the average precisions are listed in the legend). Predictions are made for each pixel by choosing the maximum likelihood class from the output of the score layer. Example prediction renderings are available in Appendix A.2. The confusion matrices for the test predictions are listed in Figure 4.3. The unknown (white) pixels in the ground truth are not included in the confusion matrix. Table 4.1 list the classification statistics for these predictions.

4.4.1 Foreground-Background Segmentation

The base DRIU model was created for classifying the foreground, i.e. the complement of the background class. The optimal F-score from Maninis et al. for the base model was 0.822[7]. To compare the 4-class model to this metric, the F-score is calculated for the maximum likelihood of scores compared to the ground truth. The artery, vein, and overlap classes are summed together as a single class. The precision, recall, and F-score of the foreground in regards to the background are 0.8533, 0.7590, and 0.8033 respectively. This indicates a slight loss of ability to perform foreground-background segmentation in comparison to the base model.

4.4.2 Background Class Presence

In order to evaluate the impact that including the background class in the model has on foreground classification, a second maximum likelihood prediction is created from the 4-Class model scores which excludes the background class (indicated by “BG Excluded” in Table 4.1). The F-score and accuracy between this prediction and the 3-Class model prediction are very close: overlap and vein classes have less than .01 difference and artery has less than .02 difference. The data indicate a very slight gain in vessel classification ability when excluding the background class from training.

4.4.3 Vessel Classification

To assess each model's ability in regards to vessel classification only, the ground truth is used to mask each prediction. This excludes the true background pixels from each prediction (aside from the first). Because the background class is included for scoring in the 4-Class model, the prediction is inherently both a foreground/background segmentation and vessel classification. To evaluate the vessel classification without regards to the background, the second maximum likelihood prediction is referenced for comparison of the 4-class model.

The artery and vein classes are accurately predicted by the 4-class model 77% of the time and by the 3-class and normal 2-class models 78% of the time. Overall, this leaves much room for improvement but could also still be potentially useful. The overlap class suffers significantly from false negatives in all models.

The 2-class model, when trained on rotated and mirrored images, shows an almost 5% increase in accuracy. The results of the 2-class overlap control model shows nearly no difference when tested on the test images. However, when validating the model by testing on the training images, there is a marked improvement across all metrics compared to the normal 2-class model.

Accuracy alone does not take into consideration the contiguity of the vessels. In the prediction renderings (see Appendix A.2), one can see that mis-classifications are often localized, spanning the width of the vessel and creating discontinuities. This quality, along with poor overlap classification, will likely make the separation directed acyclic graphs problematic. Improvement in these areas will be necessary.

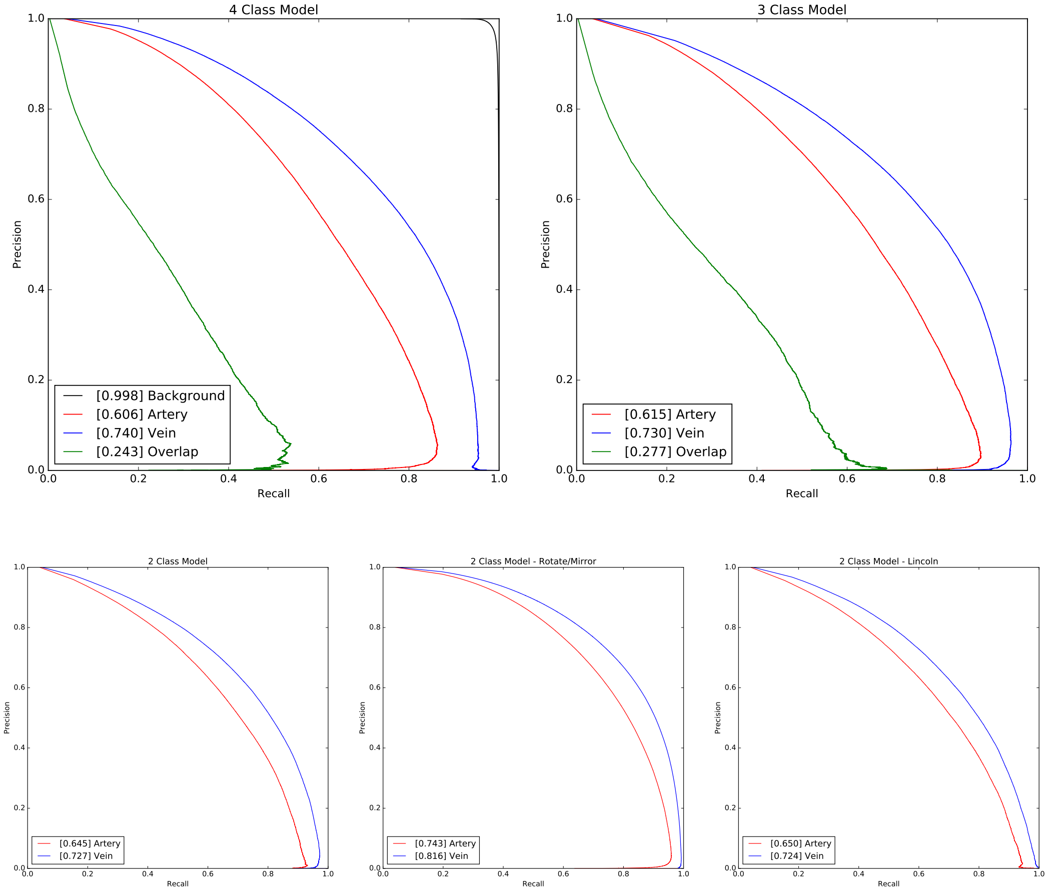
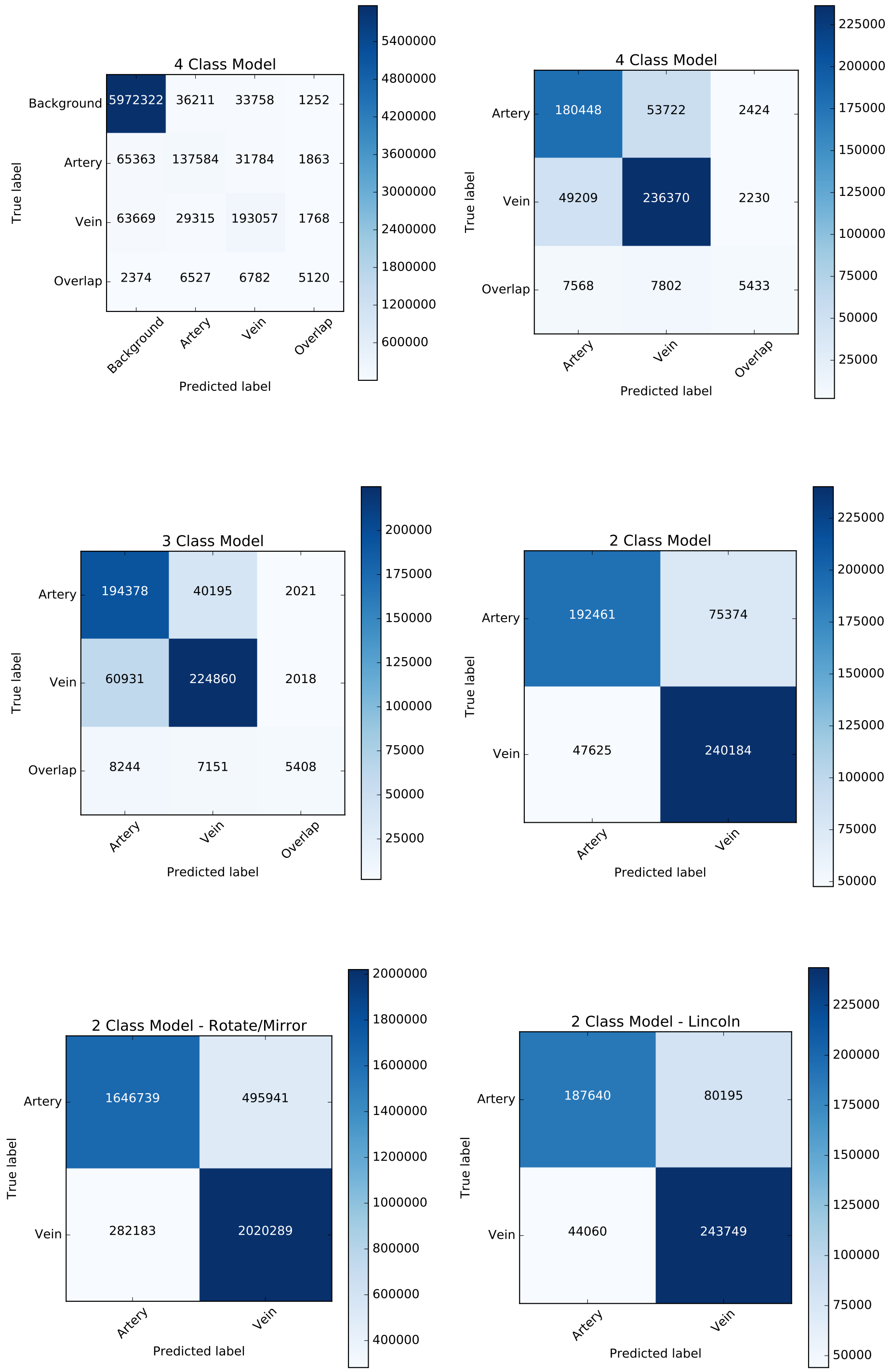


Figure 4.2: The precision-recall curve calculated for each set of class scores are inferred from all DRIVE test images. The legend lists the average precision for each class. This is equivalent to the area under the curve.

Figure 4.3: Confusion matrices for the maximum likelihood predictions.



Training Images		4-Class Model	4-Class Model BG Excluded	3-Class Model	2-Class Model	2-Class Model Rotate/Mirror	2-Class Model Lincoln
Background	Average Cross Entropy	0.0837					
	Precision	0.9908					
	Recall	0.9857					
	F-Score	0.9882					
Artery	Average Cross Entropy	0.0897	0.6060	0.7526	1.3336	1.3066	0.6025
	Precision	0.8131	0.9847	0.9960	0.8704	0.8534	0.9190
	Recall	0.8695	0.9786	0.9781	0.9654	0.9409	0.9922
	F-Score	0.8403	0.9816	0.9870	0.9155	0.8950	0.9542
Vein	Average Cross Entropy	0.0944	0.8828	1.3919	1.3297	1.2831	0.6138
	Precision	0.8386	0.9862	0.9874	0.9701	0.9486	0.9931
	Recall	0.8802	0.9824	0.9958	0.8865	0.8709	0.9275
	F-Score	0.8589	0.9843	0.9916	0.9264	0.9081	0.9592
Overlap	Average Cross Entropy	0.0104	0.0968	0.1105			
	Precision	0.6183	0.6461	0.7351			
	Recall	0.7490	0.7895	0.8469			
	F-Score	0.6774	0.7106	0.7871			
All	Accuracy	0.9772	0.9765	0.9845	0.9213	0.9020	0.9568
Test Images		4-Class Model	4-Class Model BG Excluded	3-Class Model	2-Class Model	2-Class Model Rotate/Mirror	2-Class Model Lincoln
Background	Average Cross Entropy	0.2168					
	Precision	0.9882					
	Recall	0.9785					
	F-Score	0.9833					
Artery	Average Cross Entropy	0.3227	2.0429	2.9115	2.2961	1.5492	2.4277
	Precision	0.5815	0.7627	0.8216	0.7186	0.7685	0.7006
	Recall	0.6563	0.7607	0.7375	0.8016	0.8537	0.8098
	F-Score	0.6166	0.7617	0.7773	0.7578	0.8089	0.7513
Vein	Average Cross Entropy	0.3021	2.7382	4.5497	2.2527	1.5047	2.3203
	Precision	0.6708	0.8213	0.7813	0.8345	0.8774	0.8469
	Recall	0.7275	0.7935	0.8261	0.7611	0.8029	0.7524
	F-Score	0.6980	0.8071	0.8030	0.7961	0.8385	0.7969
Overlap	Average Cross Entropy	0.0514	0.4659	0.6507			
	Precision	0.2461	0.2612	0.2600			
	Recall	0.5118	0.5386	0.5725			
	F-Score	0.3324	0.3518	0.3576			
All	Accuracy	0.9574	0.7745	0.7789	0.7786	0.8249	0.7764

Table 4.1: Table of classification statistics of maximum likelihood predictions. For each model, a prediction is made at each pixel by taking the maximum likelihood class. The average cross entropy, precision, and recall are calculated for each individual class. The classification accuracy is calculated for all classes.

Chapter 5

Conclusion and Future Work

5.1 Improving Vessel Classification

The results of the classification between arteries and veins has much room for improvement. Given the increase in classification when training the model on rotation and mirroring, this technique should be investigated and applied to the 4-class and 3-class models. Perhaps another point of interest is to alter the loss layer. Currently, each classification score is calculated with no direct dependence on the other scores. The cross entropy uses binomial distributions for each pixel and class. Instead of this, the loss and differential passed backward to the sigmoid could be calculated using a probability distribution across the classes (per pixel).

As discussed in Section 2.1, the optic nerve area is significantly complicated by ambiguities. It may be that these areas are impacting the progression of learning in the model's filter weights. An experiment should be carried out which excludes these areas, and perhaps any area containing unknown labels. It would be difficult to simply 'black out' the input image in these locations, as the zero-valued intensities would also have an undesired effect on the learned filters. An alternative may be to utilize pre-defined sub images, or patches, of input and corresponding ground truth in order to provide more optimal training data.

5.2 Patch-Based Model

As an alternative technique, a preliminary patch-based model was also implemented. The input patches are 65x65 pixels in size. They are generated by centering them on points of vessel intersections. The intersection coordinates[1] are adopted from a separate study by

[1] `http://www.cs.rug.nl/~imaging/databases/retina_database/retinalfeatures_database.html`

Azzopardi et al.[1]. Each image's set of patches is batched together and run through the model. The same 20 training images are repeated and cycled for 20,000 iterations. All of the parameters remain the same as the 4-class model. The evaluation has not been performed on the test results yet. Rotation and mirroring preprocessing has not been applied to input images, but should be included in the future.

Post-process image stitching of the scores could be investigated, perhaps through an element-wise sum and normalization. Also, this model design can potentially be integrated as a local learning model for topological-based pipelines such as in [2, 3, 5]. The author has also previously implemented a similar type of program which uses morphological skeletonization and a minimum span tree search.

5.3 Second-Stage Model

The results of testing each yield predictions with sudden, significant discontinuities along large sections of some vessels. These are mostly correctly classified, but occasionally have large contiguous sections of incorrect label. Two preliminary test models were trained to attempt to fill these sections, but showed a large decrease in classification performance. The training inputs for the models were the output scores of the 4-Class model for the first 10 test images. Each model outputs a single channel score and is trained on either artery or vein ground truth. This approach could also be further investigated as a possible corrective model.

5.4 Additional Datasets

A handful other studies have produced unique datasets for classifying retinal vessels [2, 11, 12, 17]. Most of these are sparse data representations. Nonetheless, these can still be tested against for cross-dataset comparison and validation.

5.5 Conclusion

We adapted and fine-tuned a fully convolutional neural network model for the purpose of classifying retinal vessels. We also provided a scientific framework on which to create new experimental models, and test and evaluate them. We elucidated and quantified the potential abilities of the model to separate each class through a variety of experiments. We showed the critical aspect of identifying the overlapping vessel areas. Several exciting ideas have the potential to solve the local discontinuity problem. Further development of the techniques explored in this project can potentially lead to a more complete pipeline and assist in the virtual reconstruction of the vasculature.

Bibliography

[1] George Azzopardi and Nicolai Petkov. Detection of retinal vascular bifurcations by trainable V4-like filters. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6854 LNCS(PART 1):451–459, 2011.

[2] Rolando Estrada, Michael Allingham, Priyatham Mettu, Scott Cousins, Carlo Tomasi, and Sina Farsiu. Retinal artery-vein classification via topology estimation. *IEEE transactions on medical imaging*, 0062(c):1–18, 2015.

[3] Qiao Hu, Michael D Abràmoff, and Mona K Garvin. Automated construction of arterial and venous trees in retinal images. *Journal of medical imaging (Bellingham, Wash.)*, 2(4):044001, 2015.

[4] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[5] Vinayak S. Joshi, Joseph M. Reinhardt, Mona K. Garvin, and Michael D. Abramoff. Automated method for identification and artery-venous classification of vessel trees in retinal vessel networks. *PLoS ONE*, 9(2):1–12, 2014.

[6] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation ppt. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[7] K K Maninis, J Pont-Tuset, P Arbeláez, and L Van Gool. Deep Retinal Image Understanding. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2016.

[8] John Nickolls, Ian Buck, Michael Garland, and Kevin Skadron. Scalable parallel programming with cuda. *Queue*, 6(2):40–53, March 2008.

[9] Nicholas A. Nystrom, Michael J. Levine, Ralph Z. Roskies, and J. Ray Scott. Bridges: A uniquely flexible hpc resource for new communities and data analytics. *In Proceedings of the 2015 Annual Conference on Extreme Science and Engineering Discovery Environment - XSEDE15*, 2015.

[10] Touseef Ahmad Qureshi, Maged Habib, Andrew Hunter, and Bashir Al-diri. A Manually-Labeled , Artery / Vein Classified Benchmark for the DRIVE Dataset Sunderland Eye Infirmary , UK. pages 2–5, 2013.

[11] D Relan, T Macgillivray, L Ballerini, and E Trucco. Automatic Retinal vessel classification using a Least Square- Support Vector Machine in VAMPIRE *. *36th Annual International Conference of the IEEE Engineering in Medicince and Biology Society*, pages 142–145, 2014.

[12] Marc Saez, Sonia Gonzalez-Vazquez, Manuel Gonzalez-Penedo, Maria Antonia Barcelo, Marta Pena-Seijo, Gabriel Coll de Tuero, and Antonio Pose-Reino. Development of an automated system to classify retinal vessels into arteries and veins. *Computer Methods and Programs in Biomedicine*, 108(1):367–376, 2012.

[13] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *International Conference on Learning Representations (ICRL)*, pages 1–14, 2015.

[14] Joes Staal, Michael D. Abràmoff, Meindert Niemeijer, Max A. Viergever, and Bram Van Ginneken. Ridge-based vessel segmentation in color images of the retina. *IEEE Transactions on Medical Imaging*, 23(4):501–509, 2004.

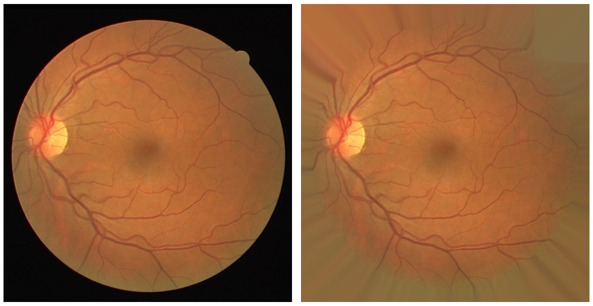
[15] John Towns, Gregory D. Peterson, Ralph Roskies, J. Ray Scott, Nancy Wilkens-Diehr, Timothy Cockerill, Maytal Dahan, Ian Foster, Kelly Gaither, Andrew Grimshaw, and et al. Xsede: Accelerating scientific discovery. *Computing in Science & Engineering*, Sep 2014.

[16] S. G. Vazquez, B. Cancela, N. Barreira, M. G. Penedo, M. Rodriguez-Blanco, M. Pena Seijo, G. Coll De Tuero, M. A. Barcelo, and M. Saez. Improving retinal artery and vein classification by means of a minimal path approach. *Machine Vision and Applications*, 24(5):919–930, 2013.

[17] Andrea Zamperini, Andrea Giachetti, Emanuele Trucco, and Khai Sing Chin. Effective features for artery-vein classification in digital fundus images. *Proceedings - IEEE Symposium on Computer-Based Medical Systems*, 2012.

Appendices

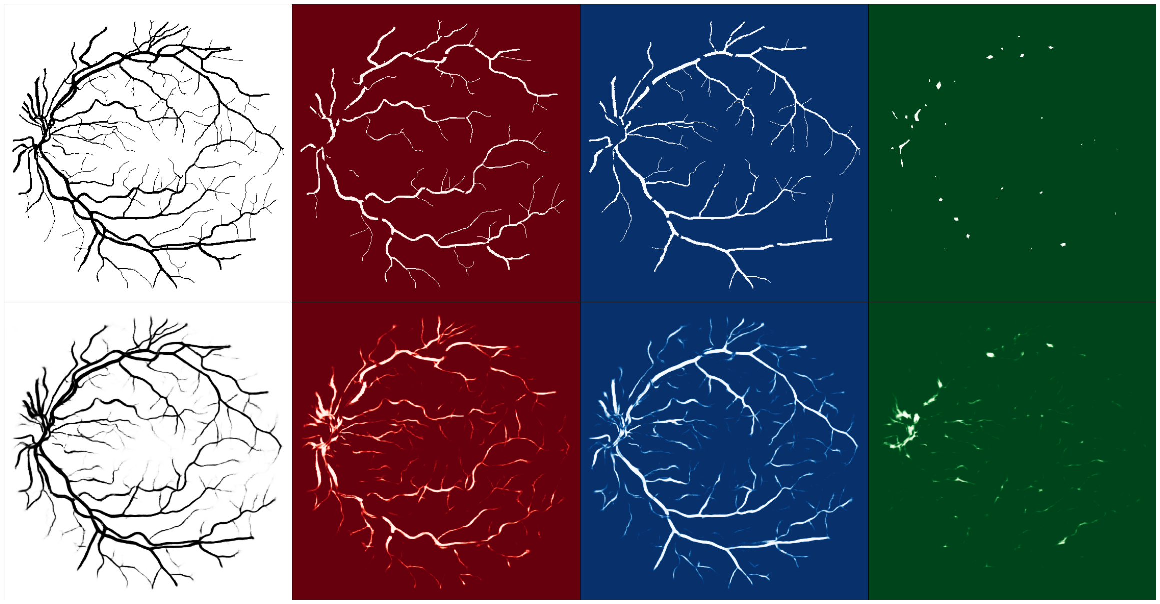
Appendix A
Renderings



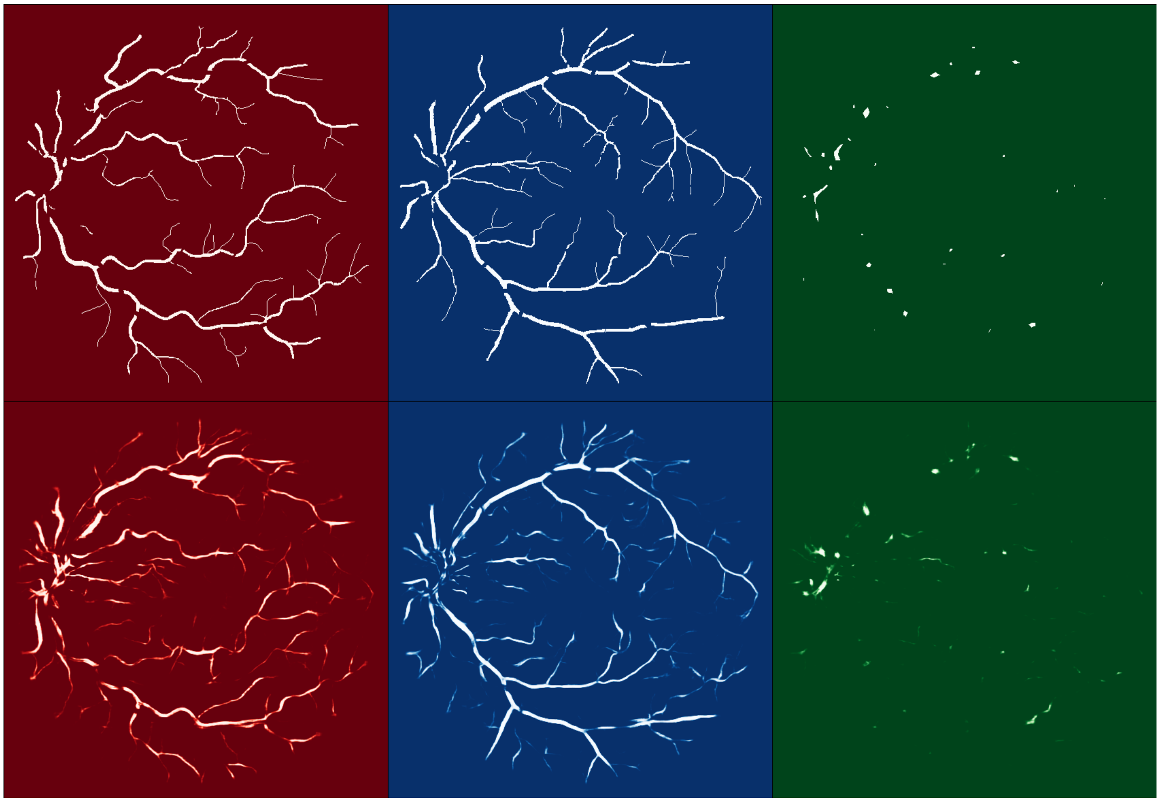
(a) Original Input Image (b) Preprocessed Image

A.1 Scores

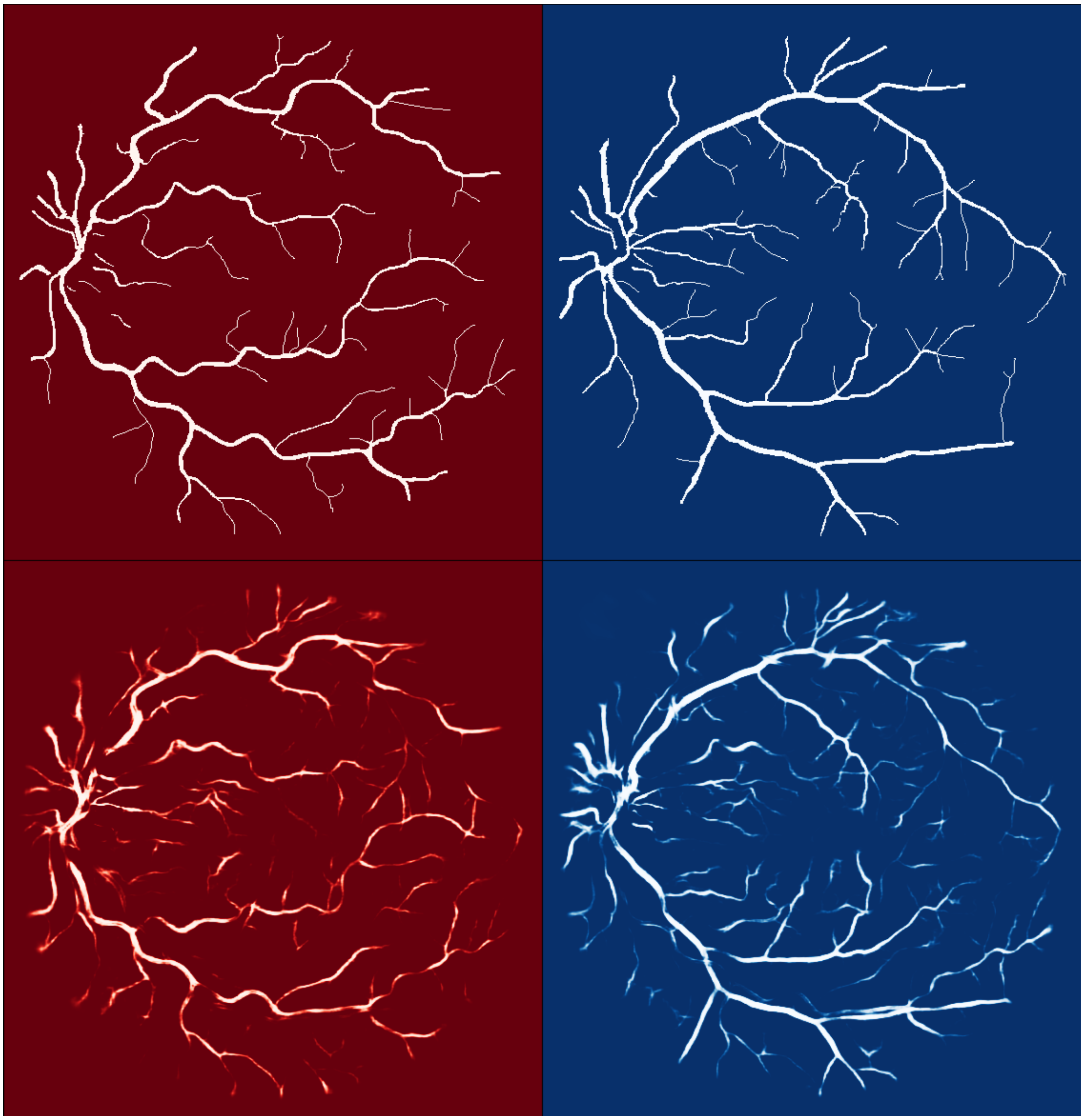
The qualitative results are shown as the individual class scores of the first DRIVE test image for each model. The top row of each example displays the binary ground truth and each the bottom row displays the test scores. A higher pixel intensity corresponds to a higher score.



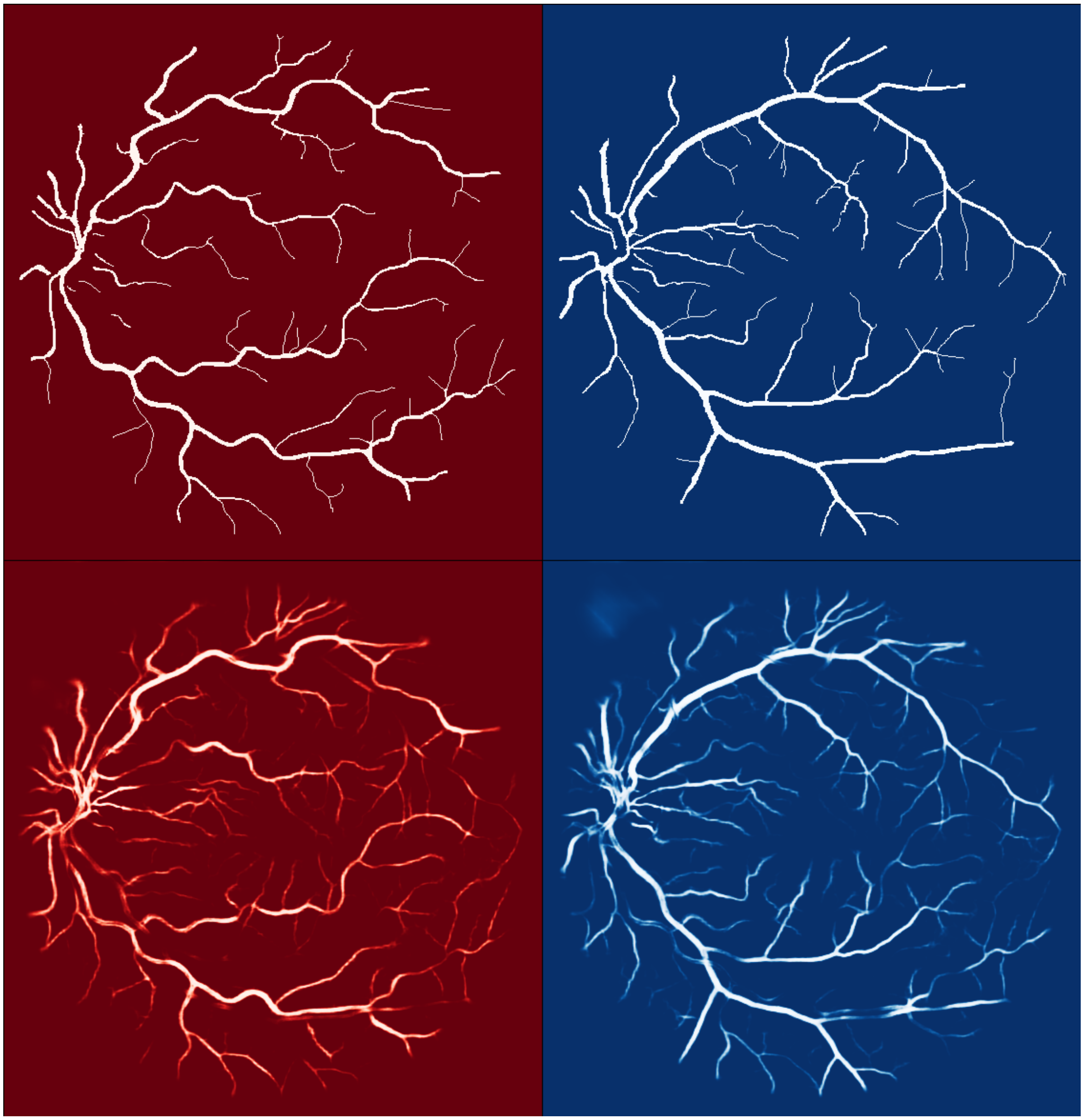
Example A.2: 4-Class Model



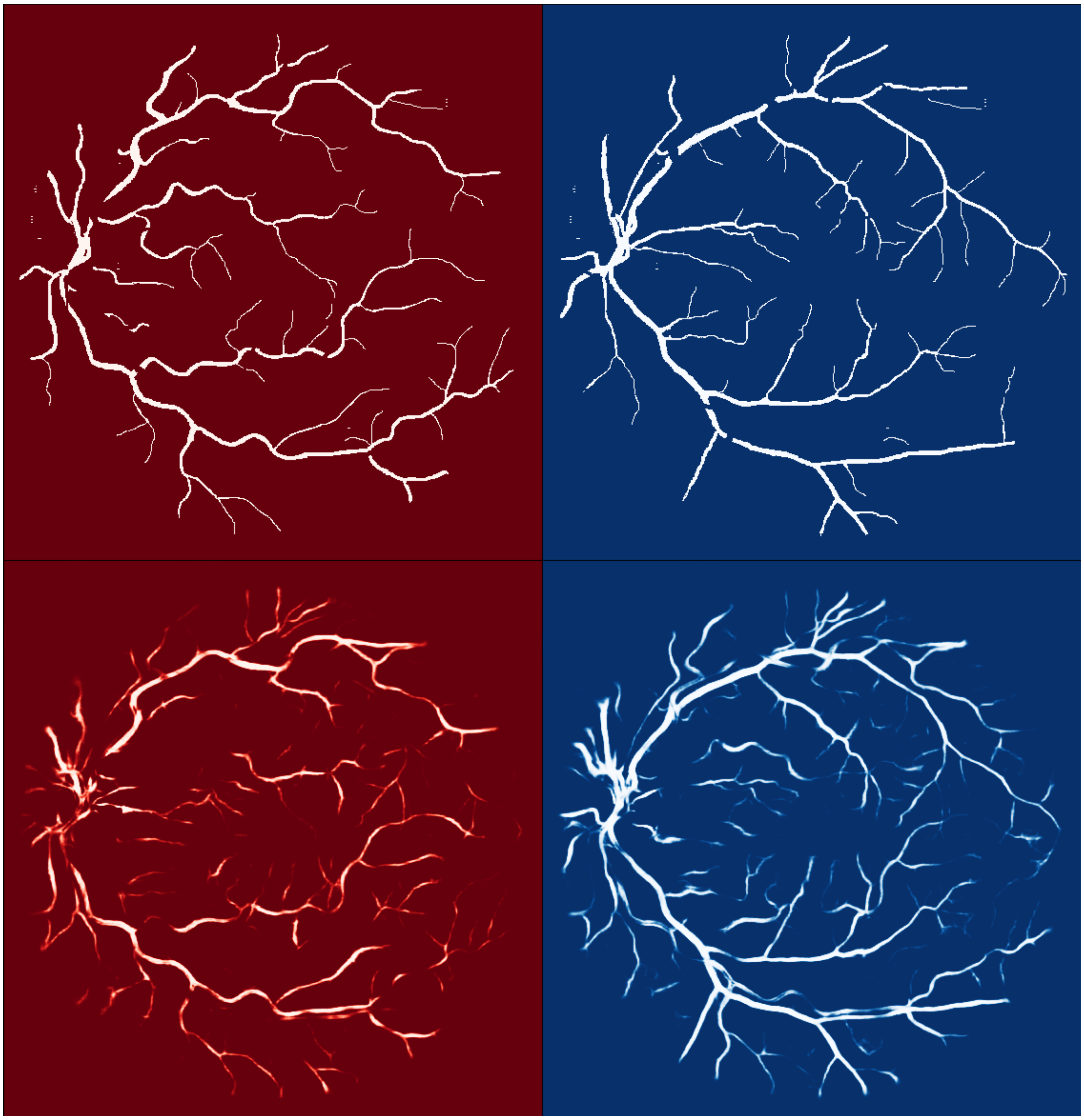
Example A.3: 3-Class Model



Example A.4: 2-Class Model (Normal)



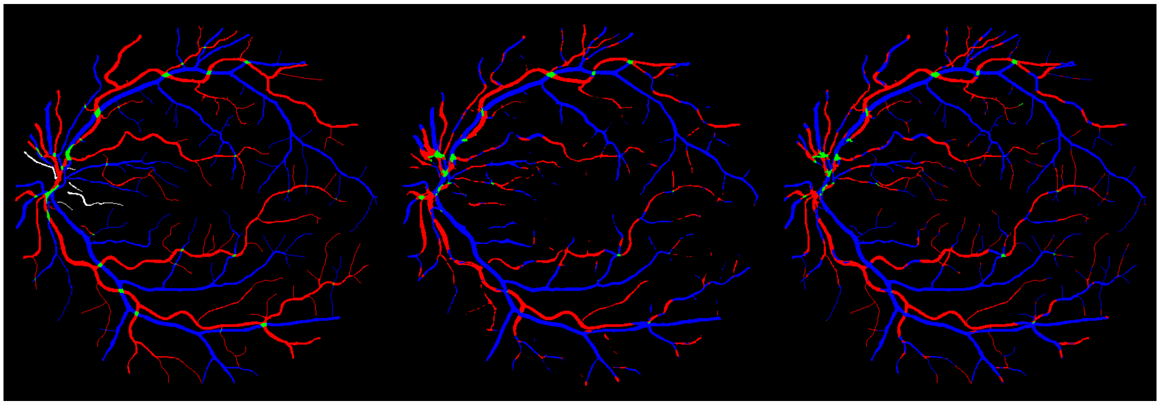
Example A.5: 2-Class Model (Mirroring/Rotation)



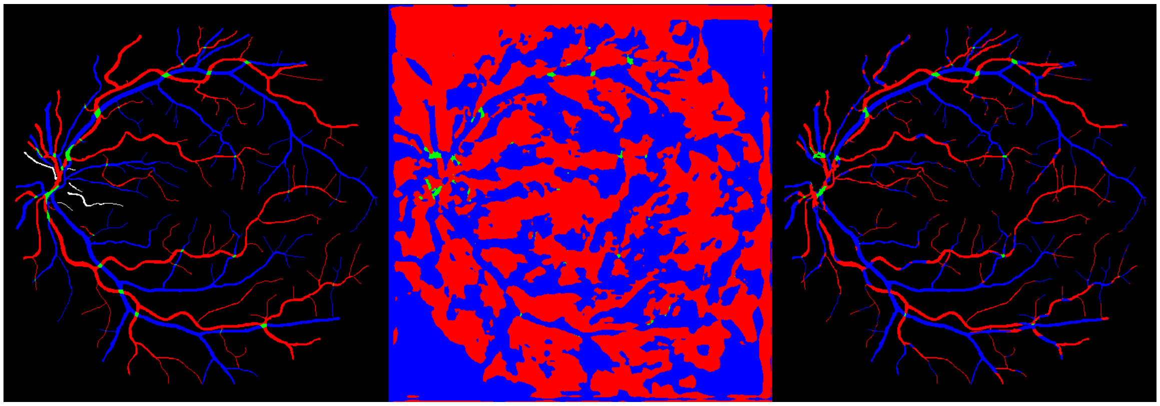
Example A.6: 2-Class Model (Overlap Control)

A.2 Predictions

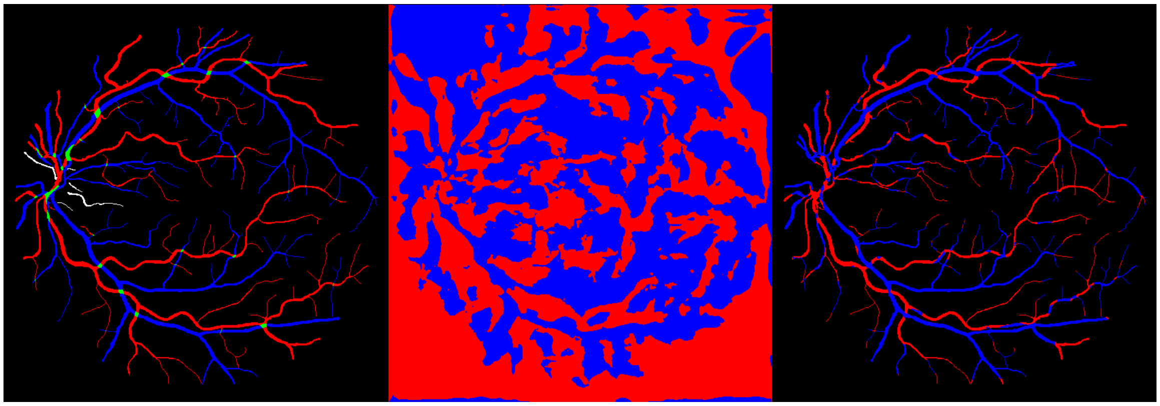
The maximum likelihood prediction results of the first DRIVE test image are shown for the models. Each left image is the ground truth. Each middle image is the unmasked arguments of the maxima. Each right image is the masked arguments of the maxima excluding the background class. The mask is created using the union of ground truth classes excluding the background class.



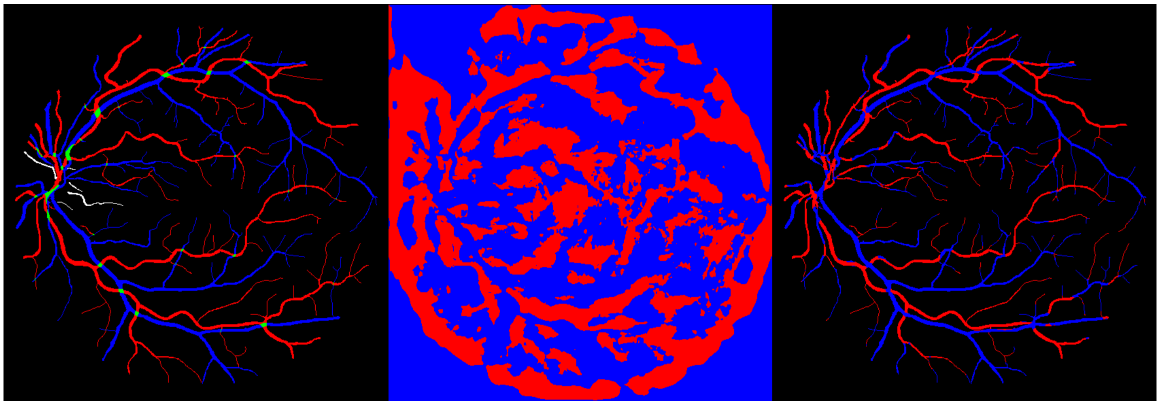
Example A.7: 4-Class Model



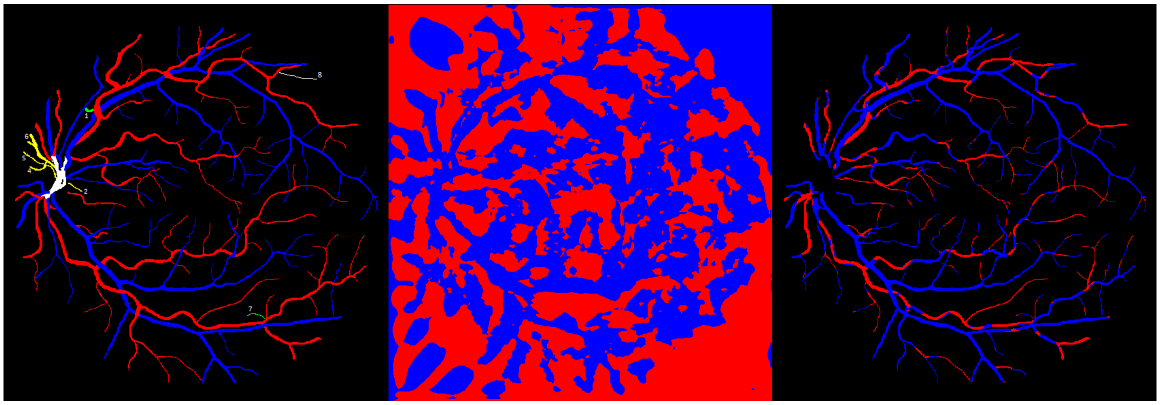
Example A.8: 3-Class Model



Example A.9: 2-Class Model (Normal)



Example A.10: 2-Class Model (Mirroring/Rotation)



Example A.11: 2-Class Model (Overlap Control)