tvst

Artificial Intelligence

Automated Detection of Retinal Detachment Using Deep Learning-Based Segmentation on Ocular Ultrasonography Images

Onur Caki[1,2], Umit Yasar Guleser[3], Dilek Ozkan[2,4], Mehmet Harmanli[1,2], Selahattin Cansiz[1,2], Cem Kesim[5], Rustu Emre Akcan[5], Ivan Merdzo[6], Murat Hasanreisoglu[5,7], and Cigdem Gunduz-Demir[1,2,7]

[1] Department of Computer Engineering, Koç University, Istanbul, Turkey
[2] KUIS AI Center, Koç University, Istanbul, Turkey
[3] Department of Ophthalmology, Acıbadem Maslak Hospital, Istanbul, Turkey
[4] Biomedical Sciences and Engineering Program, Koç University, Istanbul, Turkey
[5] Department of Ophthalmology, School of Medicine, Koç University, Istanbul, Turkey
[6] Department of Ophthalmology, University Hospital Mostar, Mostar, Bosnia and Herzegovina
[7] Koç University Research Center for Translational Medicine, Istanbul, Turkey

Correspondence: Cigdem Gunduz-Demir, Department of Computer Engineering, Koç University, Istanbul 34450, Turkey. e-mail: cgunduz@ku.edu.tr

Received: September 10, 2024
Accepted: January 16, 2025
Published: February 27, 2025

Keywords: deep learning; ocular ultrasound; retinal detachment; automated detection; automated segmentation

Citation: Caki O, Guleser UY, Ozkan D, Harmanli M, Cansiz S, Kesim C, Akcan RE, Merdzo I, Hasanreisoglu M, Gunduz-Demir C. Automated detection of retinal detachment using deep learning-based segmentation on ocular ultrasonography images. Transl Vis Sci Technol. 2025;14(2):26, https://doi.org/10.1167/tvst.14.2.26

Purpose: This study aims to develop an automated pipeline to detect retinal detachment from B-scan ocular ultrasonography (USG) images by using deep learning-based segmentation.

Methods: A computational pipeline consisting of an encoder-decoder segmentation network and a machine learning classifier was developed, trained, and validated using 279 B-scan ocular USG images from 204 patients, including 66 retinal detachment (RD) images, 36 posterior vitreous detachment images, and 177 healthy control images. Performance metrics, including the precision, recall, and F-scores, were calculated for both segmentation and RD detection.

Results: The overall pipeline achieved 96.3% F-score for RD detection, outperforming end-to-end deep learning classification models (ResNet-50 and MobileNetV3) with 94.3% and 95.0% F-scores. This improvement was also validated on an independent test set, where the proposed pipeline led to 96.5% F-score, but the classification models yielded only 62.1% and 84.9% F-scores, respectively. Besides, the segmentation model of this pipeline led to high performances across multiple ocular structures, with 84.7%, 78.3%, and 88.2% F-scores for retina/choroid, sclera, and optic nerve sheath segmentation, respectively. The segmentation model outperforms the standard UNet, particularly in challenging RD cases, where it effectively segmented detached retina regions.

Conclusions: The proposed automated segmentation and classification method improves RD detection in B-scan ocular USG images compared to end-to-end classification models, offering potential clinical benefits in resource-limited settings.

Translational Relevance: We have developed a novel deep/machine learning based pipeline that has the potential to significantly improve diagnostic accuracy and accessibility for ocular USG.

Introduction

Ocular ultrasonography (USG) is a noninvasive imaging modality that plays an important role in the diagnosis and monitoring of posterior segment diseases. This modality provides rapid, real-time imaging, enabling the identification and localization of abnormalities in the eye.[1] Because it is portable and relatively inexpensive, it can be used by trained

emergency physicians at the bedside in patients with visual changes to diagnose sight-threatening emergency conditions when direct or indirect ophthalmoscopic examination is not available.

Retinal detachment (RD) is one of such emergency conditions, which occurs when the neurosensory retina separates from the underlying retinal pigment epithelium layer. Its early diagnosis is very important because delay in treatment can lead to irreversible vision loss.[2] Although dilated fundoscopic examination is usually used to clinically diagnose RD,[3] ocular USG performed by trained operators may be a rapid way to detect RD in selected cases in emergency services, especially when an ophthalmologist is not available. Additionally, media opacities that prevent light from penetrating the eye, such as mature cataracts, vitreous hemorrhages, and inflammation, may prevent fundoscopy, and ocular USG offers an effective alternative to detect RD also in ophthalmology practice. Ocular USG performed by a trained observer has been shown in both the ophthalmology and emergency medicine literature to have high sensitivity and specificity for detecting RD.[4] On the other hand, despite its usefulness, interpretation of ultrasound images requires experience and expertise and can be time consuming. Differentiating between different pathologies, such as RD and posterior vitreous detachment (PVD), can be challenging and subject to observer variability. Computational tools that can quantitatively process ocular USG images provide invaluable support tools for rapid and objective analysis.

In the last decade, several deep learning-based models have been proposed for the analysis of medical images, especially focusing on the problems of medical image classification and segmentation. Although these problems have been widely studied on images of different modalities in ophthalmology (e.g., optical coherence tomography images),[5,6] there are only a few studies for B-scan ocular USG images.[7–10] Furthermore, most of these studies focused on classification. They typically used pretrained models to extract features from either an entire image[7,8] or the eyeball region[9,10] and then trained a classifier on the top of these features for ocular USG classification. Although these previous studies used segmentation networks or object detectors to locate the regions of interests in an image,[9–11] they still fed the bounding boxes of these regions to a pretrained neural network and achieved classification based on the features extracted by this pretrained network. On the other hand, because of the black-box nature of deep learning models in general, the features extracted by a neural network are difficult to explain, and thus the decisions based on these features are not easily interpretable. In the literature, there exists only one study that segmented the optic nerve sheath, besides the eyeball region, in an ocular USG image and extracted an easy-to-explain feature (its diameter) to detect intracranial pressure caused by head trauma.[12] Additionally, the previously cited studies did not focus on multiple structure segmentation, and their classifier was based on the features extracted by pretrained networks, exhibiting the black-box nature.[9–11]

To address this gap, this article aims to develop a computational pipeline that performs automatic segmentation of posterior segment structures in B-scan ocular USG images, enabling extraction of easy-to-interpret features reflecting an ophthalmologist's opinion of an eye pathology. Although these segmented structures can be used to define features related to various eye pathologies, this study specifically focuses on RD detection.

Material and Methods

Ethical Approval and Participant Consent

This study was approved by the Institutional Review Board of Koç University (protocol number: 2024.284.IRB2.119) and each conduct in the study adhered to the tenets of the Declaration of Helsinki. Written and verbal informed consent was obtained from each participant.

Data Collection and Preparation

All computational models were trained and tested on 279 B-scan ocular USG images of 204 participants who visited the Ophthalmology Department of Koç University Hospital between 2019 and 2024. All participants underwent full ophthalmological examination including autorefractometry, best-corrected visual acuity, slit-lamp biomicroscopy, non-contact tonometry, and dilated fundoscopy. B-scan ocular USG was performed to all participants (Ellex Eye Cubed i3 Ultrasound System, Adelaide, Australia) using a 10 MHz high-frequency transducer with the patient in an upright or supine position by experienced ophthalmologists.

Study Groups and Image Selection

The RD group included 66 USG images from 30 participants whose diagnosis of retinal detachment was confirmed by fundoscopy or perioperatively. Thirty-six USG images of 18 PVD cases and

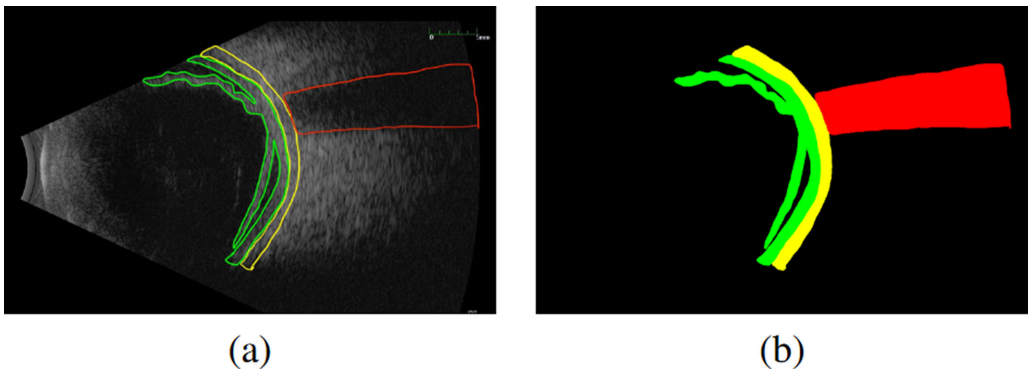

Figure 1. Examples of annotations in different forms. (**a**) B-scan ocular USG image with manually delineated boundaries for retina/choroid, sclera, and optic nerve sheath, and (**b**) the corresponding segmentation map generated from the annotations in (**a**).

177 USG images of 156 healthy controls were included in the control group. Patients with poor image quality and choroidal and vitreoretinal pathologies, such as choroidal mass, vitreous hemorrhage, endophthalmitis, posterior uveitis, posterior scleritis, and posterior staphyloma, other than retinal detachment and posterior vitreous detachment, were excluded from the study.

Image Preprocessing and Annotation

B-scan USG images were exported in the .jpeg format. In addition to image-level labels, the retina/choroid, sclera, and optic nerve boundaries were manually delineated by two ophthalmologists (U.Y.G and M.H.), as illustrated in Figure 1a. Based on these hand-drawn boundaries, a segmentation map (i.e., pixel-level annotations) was generated for each image. Figure 1b depicts the segmentation map derived from the annotated posterior segment structures shown in Figure 1a. The acquired images were of three different resolutions: 1280 × 800, 1920 × 1200, and 1600 × 1200 pixels. To provide the inputs of the same resolution to the segmentation model (deep neural network), all these images were resized to the dimensions of 1280 × 800. Additionally, after resizing, the first and last 240 pixels were removed from the left and right sides of the images, respectively, to focus onlyon the ocular region, thereby conveying exclusively eye disease–related information to the segmentation model. Figure 2 shows example input images of a healthy eye and an eye with RD condition together with their annotations.

Training and Validation

All segmentation and classification models were evaluated using threefold cross-validation to reduce bias caused by the partitioning of the data into training and test sets. In this technique, all participants were randomly divided into three folds and testing was repeated three times. In each trial, all images of the participants in two folds were used to learn the models (i.e., to train the segmentation and classification models), and all images of the participants in the remaining fold were used as the test set to calculate the performance metrics. This was repeated three times, each time using different configurations of folds for training and testing. At the end, the average metrics were calculated on the test sets of these three different configurations. Note that approximately 20% of the training data in each trial was used as a validation set when a neural network was trained. This validation set was to early-stop the network training to reduce the risk of overfitting. Again, the participants or images of the validation set were not used in the test set. Table 1 presents the number of images in the training, validation, and test sets for each trial. To assess the generalization ability of the proposed pipeline, we validated the classification results on an independent test set consisting of 15 RD and 50 non-RD (32 PVD and 18 healthy control) cases. We tested all networks, which were trained on the first dataset, on this independent test set and reported the classification results.

Development of the Pipeline

This article proposes a computational pipeline for ocular USG images, in which a deep learning model performs segmentation of posterior segment structures in the eye, namely, the retina/choroid, sclera, and optic nerve regions. This pipeline enables us to define easy-to-explain features on the segmented regions, which can then be used for further analysis. In this article, we chose RD detection as a showcase example to demon-

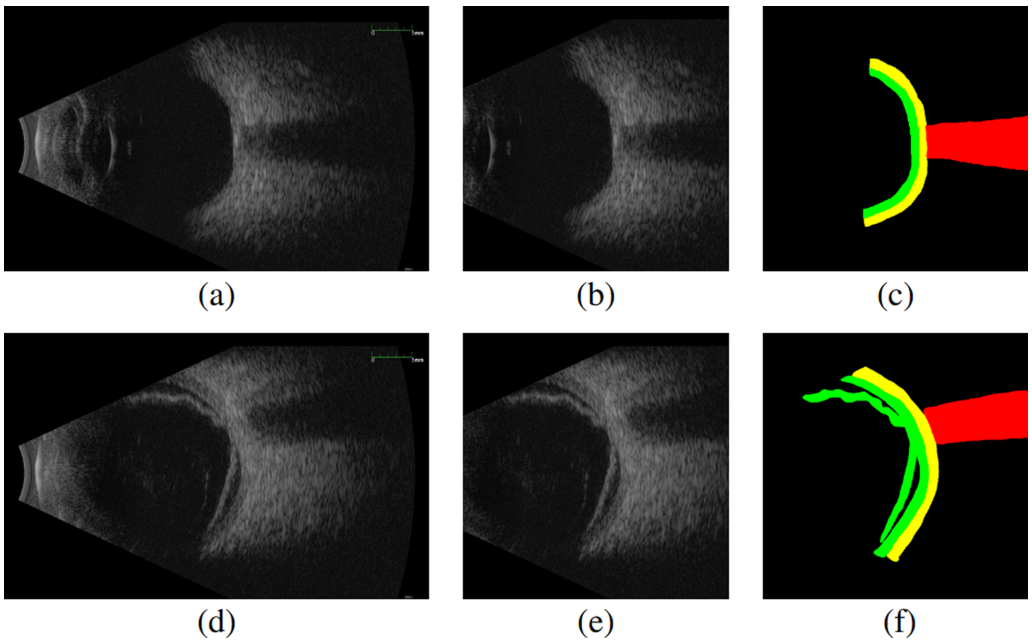

Figure 2. Examples of B-scan ocular USG images together with their manual annotations. (**a**) USG image of a healthy eye, (**b**) the cropped image that will be fed to the segmentation model, and (**c**) its annotated regions. (**d**) USG image of an eye with RD condition, (**e**) the cropped image that will be fed to the segmentation model, and (**f**) its annotated regions. In the annotation maps, the retina/choroid is marked with green, the sclera with yellow, and the optic nerve sheath with red.

Table 1. Number of the Images in the Training, Validation, and Test Sets for Each Trial of Threefold Cross-Validation

	Training			Validation			Test		
	RD	Control	Total	RD	Control	Total	RD	Control	Total
Trial 1	36	115	151	9	30	39	21	68	89
Trial 2	35	111	146	9	28	37	22	74	96
Trial 3	34	113	147	9	29	38	23	71	94

The numbers of images for the RD and control groups were also reported.

strate the use of the entire pipeline. The proposed pipeline has the training and testing phases, which are illustrated in Figures 3 and 4, respectively. The general outline of these phases is given in the next two paragraphs, and the details of the segmentation and classification modules used in these phases are explained in the next two subsections. The training phase, depicted in Figure 3, starts with constructing an encoder-decoder network and learning the network weights by the backpropagation algorithm on the training images and their ground truth maps (i.e., pixel-level annotations of the retina/choroid, sclera, and optic nerve regions in the images). This backpropagation algorithm, which is the fundamental learning technique for neural networks, starts with random weights and iteratively updates them to minimize the difference between pixel labels in the ground truth maps and those predicted by the network. The training phase continues with defining features that reflect an ophthalmologist's view of an eye abnormality of the interest (in our showcase example, features related to RD). These features are extracted on the ground truth maps (pixel-level annotations) of the training images and fed to a classifier as inputs. Finally, the classifier is trained based on the extracted features of the training images and their ground truth labels. This time, ground truths are image-level annotations (in our showcase example, binary class labels to show whether an image exhibits RD or not).

The testing phase inputs an ocular USG image, for which neither the pixel-level nor the image-level ground truth is known (Fig. 4). It first uses the encoder-decoder network, whose weights were learned in the training phase, to obtain the pixel-level predictions. It then eliminates noisy false positives in the predicted map by removing connected components with areas less than a predefined threshold. In our experiments, this threshold was empirically set to be 1500 pixels, considering the image resolutions. At the end, this phase extracts the features from the resulting map, and uses the classifier, trained in the training phase, to estimate the class label of the image. In our showcase example, it labels the input image as being in the RD or the control group.

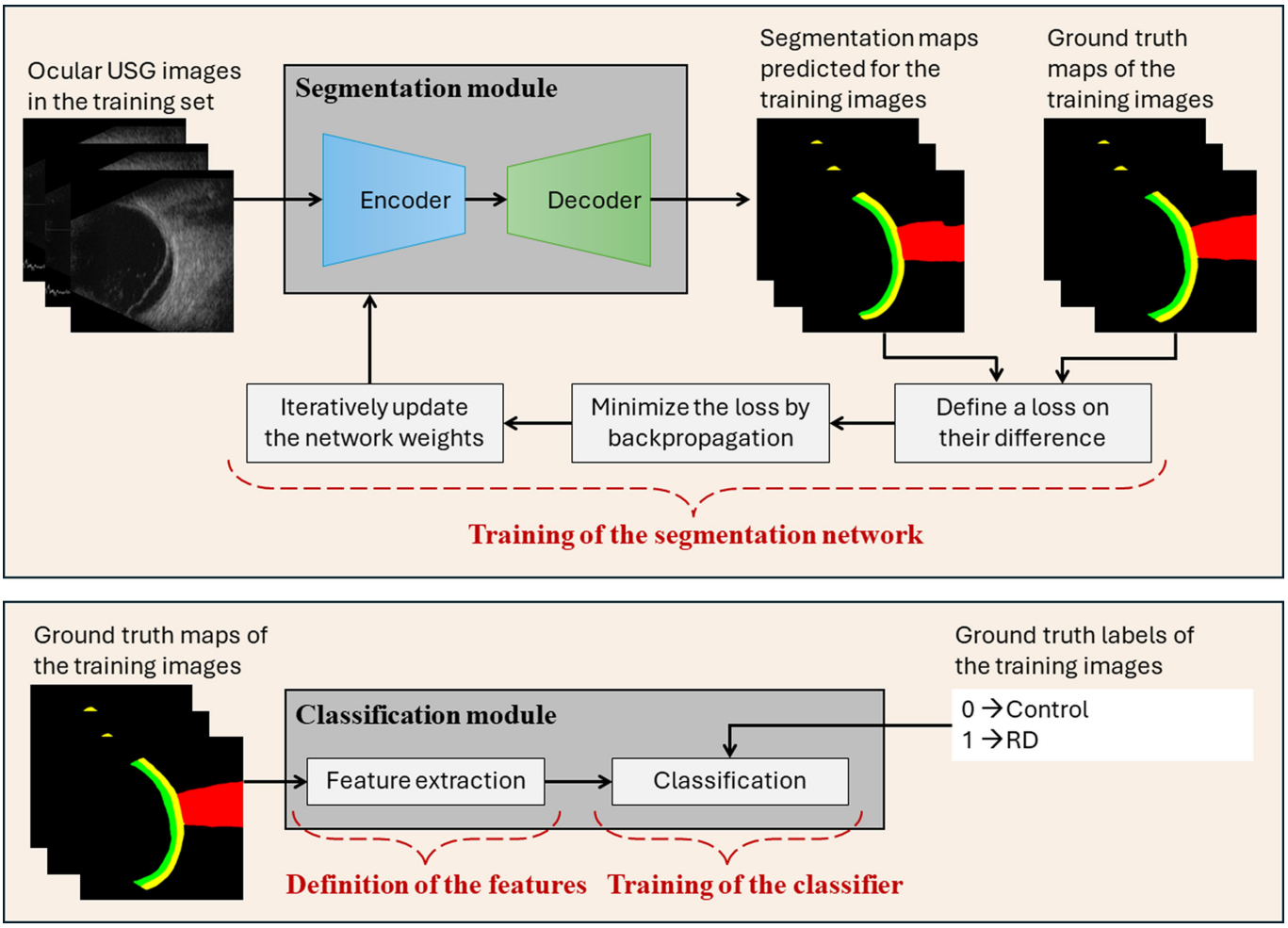

Figure 3. Schematic overview of the training phase of the proposed pipeline.

Segmentation Module

The goal of a segmentation model is to assign a label to each image pixel. In the last decade, deep learning models have been frequently used for this purpose. Since the proposal of UNet[13] in 2015, many variants of encoder-decoder networks have been proposed for medical image segmentation.[14] Our proposed pipeline also uses an encoder-decoder network for ocular USG segmentation. It is TransUNet that has an attention mechanism with a transformer block.[15] The UNet architecture consists of symmetric encoder and decoder paths, along with long skip connections between them. The encoder path is responsible for learning high-level features (downsampled feature maps) to encode an input image whereas the decoder path is responsible for constructing the segmentation map at the same resolution as the input image from these high-level features. To do that, the encoder-path extracts feature by successive use of convolutional layers while it achieves downsampling by max-pooling layers between the convolutional layers. The decoder path constructs the segmentation map by upsampling the downsampled feature maps to the original image size via transpose convolution layers. Long skip connections are used to better recover spatial information in the upsampling process. Because this UNet model may struggle to capture long-range dependencies because of the inherent locality of the convolution operations, the proposed pipeline uses TransUNet that combines convolutional layers with a vision transformer to exploit local features and global context information, respectively. It has an architecture similar to UNet except that transformer layers are embedded in its encoder. Particularly, its encoder path uses the ResNet-50[16] architecture to generate the high-level feature maps, which are then tokenized into 16×16 patches and fed to the vision transformer (ViT) with 12 transformer layers.[17] In the training of this segmentation network, the weights of ResNet-50

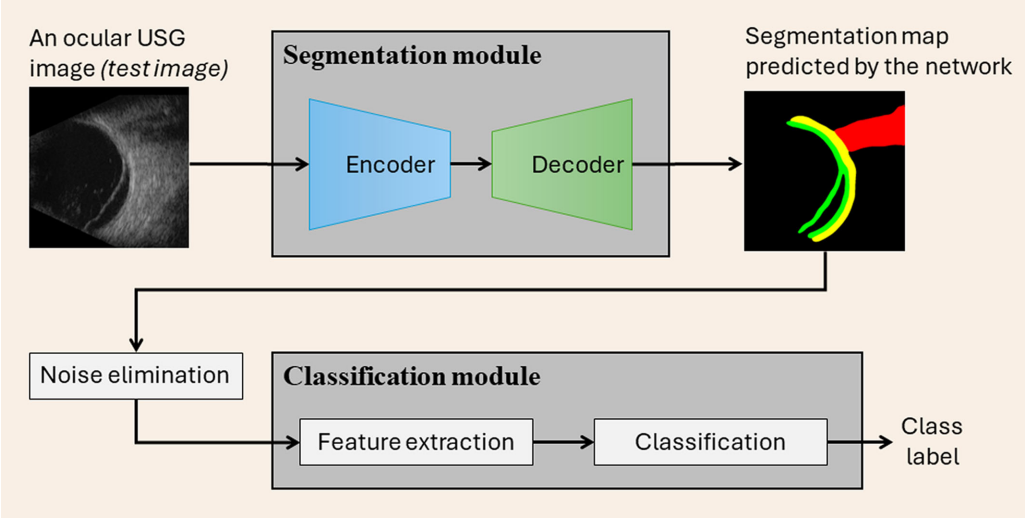

Figure 4. Schematic overview of the testing phase of the proposed pipeline.

and ViT are initialized with those pretrained on the ImageNet dataset.[18] The weights of the decoder path are randomly initialized. Afterward, they are end-to-end trained on our ocular USG data using stochastic gradient descent with momentum as the optimizer. The summation of the cross-entropy and the Dice loss is used as the loss function. The momentum and weight decay parameters are set to 0.9 and 0.0001, respectively, and the batch is set to 2. The early stopping is used; the training is stopped if there is no improvement in the loss function calculated on validation images in the last 25 epochs. All networks are constructed and trained in the Python programming language using PyTorch.

To understand the effectiveness of using the TransUNet network, which had the transformer mechanism to encode long-range dependencies, we compared it with two other networks. First, we used the original architecture of the UNet network proposed by Ronneberger et al.,[13] except that we added batch normalization layers after each convolutional layer to alleviate the overfitting problem. We trained this network from scratch, using the same training settings used to train TransUNet. Additionally, we removed the transformer layers from the original TransUNet design. In particular, in its original design, the features extracted from the ResNet-50, which was pre-trained on the ImageNet dataset, were used as inputs to the transformer layers in the encoder path. In our comparison, we removed these transformer layers from the encoder path and directly upsampled the hidden features to the full resolution through the decoding path.

Retinal Detachment Detection

The classification module in the proposed pipeline relies on extracting easy-to-explain features that will be also related to an eye disease of the interest. In this article, we use the problem of RD detection as a showcase example. For that, we propose to use the segmented retina/choroid and define three simple but easy-to-explain features. All these features are to quantify retinal layer fragments resulting from the detachment of the retina. The selection of these features is motivated by the clinical and anatomical characteristics of retinal detachment. In healthy anatomical structures, the retina and choroid typically appear as a continuous layer, resulting in each column in the binary map of the retinal/choroid structure containing at most two connected components. In contrast, retinal detachment causes the retinal structure to fold or separate, leading to an increase in the number of connected components in certain columns. This behavior is quantified by finding the maximum value of an array where each element represents the number of connected components of a column. Furthermore, this behavior is quantified by finding the maximum value of an array where each element represents the number of connected components of a column. Furthermore, the standard deviation of these connected components across the columns reflects the irregularity of the retinal structure, which is a key indicator of retinal detachment. To quantify these characteristics with the first two features, we obtain a column-wise projection of the estimated retinal regions in the segmentation map. This projection considers each column in the map separately and calculates the number of

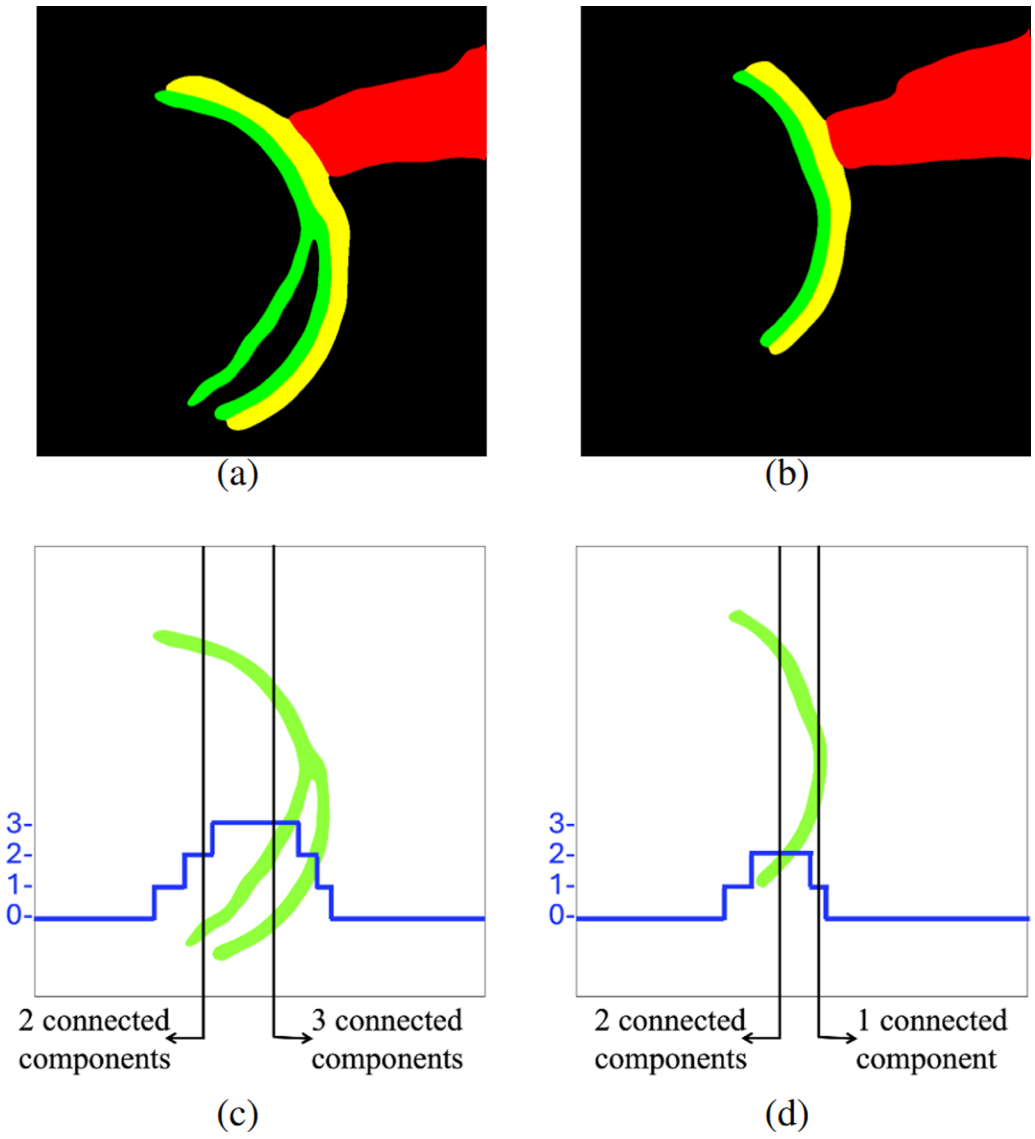

Figure 5. Illustration of feature extraction used for RD detection. Multiple structures estimated by the segmentation module for an example test image (**a**) with and (**c**) without RD. The column-wise projection (*blue*) on the estimated retina/choroid region (green) in the segmentation map (**b**) with and (**d**) without RD. *Black arrows* illustrate the calculation of this projection for the two selected columns in each map. For example, the first *black arrow* in (**c**) is for the column where there are two connected components found on the pixels of this corresponding column.

connected components on the pixels of the corresponding column. This calculation is illustrated in Figure 5. As also seen in this figure, it is expected to have one or two connected components for the cases without any RD. On the other hand, this number can be greater than two in some columns of the map and the projection exhibits more variations for RD cases. For example, for the segmentation maps illustrated in Figure 5, the maximum value is 3 and the standard deviation is 0.83 for the RD sample whereas these values are 2 and 0.49, respectively, for the control sample. Additionally, because of the detached retina, the area of the retinal structure is larger compared to that of a healthy retinal structure. Thus we use the area of the segmented region as the third feature. It is worth noting that although one of these features is sufficient to accurately detect RD in most of the samples, using three enables us to have more robust classification because of possible noise and artifacts in the images or their estimated segmentation maps.

Let x_i be the feature vector of a sample i and $y_i \in \{C_0, C_1\}$ be its class label, where C_0 and C_1 denote the classes for the control and the RD group, respectively. The feature vector x_j of a training sample j is calcu-

lated on its ground truth map, and its class label y_j is already known. On the other hand, the feature vector x_t of a test sample t should be calculated on the map estimated by the segmentation module, since it does not have pixel-label annotations of the regions, and its class label y_t is not known and should be estimated by a classifier. Although the proposed pipeline allows us to use any classification algorithm, we prefer to use a simple binary classifier, the k-nearest neighbors algorithm. For a given test sample, this classifier selects the k-nearest samples from the training set based on the Euclidean distance between their feature vectors and classifies the test sample with the majority class of the selected samples. Here it is essential to select an optimum k value to obtain an accurate classifier. In our study, we select the value of k by using the previously proposed optimization method.[19]

This method[19] identifies an optimal value for a machine learning algorithm based on the asymptotic properties of the nearest neighbor classification rule[20] by comparing a set of labeled data from a class of interest against a set of unlabeled data. In this quasi-supervised learning framework, the objective is to estimate the samples in the unlabeled dataset that overlap with the labeled set of the target class. In this study, we adapt this method to a supervised framework, in which the optimum k is selected on the training set by comparing a set of RD samples against a set of control samples using the previously described optimization method and classifier.[19] The k value selected on the training set is then used for classifying the test samples using the k-nearest neighbors classifier. To find the optimal k value, this optimization method calculates the cost function $L(k)$ for each possible value of k, from 1 to the number of RD samples in the training set and selects the value that minimizes $L(k)$. This cost function is defined as

$$L(k) = 4\,\frac{|D_1|}{|D_0|}\sum_{j\in D_0} P\left(C_0|x_j\right)\cdot P\left(C_1|x_j\right) + 4\sum_{j\in D_1} P\left(C_0|x_j\right)\cdot P\left(C_1|x_j\right) + 2k \qquad (1)$$

where $D_0 = \{(x_j, y_j)\,|\,y_j = C_0\}$ and $D_1 = \{(x_j, y_j)\,|\,y_j = C_1\}$ denote the subsets of the training set that contain only the first and second class samples, respectively, and $|D_0|$ and $|D_1|$ denote the number of samples in these subsets. $P(C_0|x_j)$ and $P(C_1|x_j)$ are the probabilities of classifying the sample j with the first and second class, respectively, using the remaining training samples, assuming that its class label y_j is unknown. The multiplication of these probabilities reaches its minimum with increasing certainty of classification (in this case, one probability will go to 1 and the other to 0, and their product will be close to 0). Thus the first and the second term in Equation 1 penalizes the classification uncertainty for the training samples in D_0 and D_1, respectively. The third term, $2k$, is to penalize selecting large k values, and thus is used for regularization. The probabilities $P(C_0|x_j)$ and $P(C_1|x_j)$ are estimated with the average rates $f_0(j)$ and $f_1(j)$, respectively,[19] using the quasi-supervised learning algorithm.[20] This estimation relies on repeating the following experiment M times, for each training sample j separately, and obtaining the average rate of classifying this sample with the classes C_0 and C_1 over the M trials. In each trial, R_m is created by randomly selecting k samples from D_0 and k samples from D_1, and the sample j is classified with the label of its nearest neighbor in R_m. Although the number of trials M is theoretically a very large number, the quasi-supervised learning algorithm provides a computationally efficient numerical approximation of these rates.[20] Once it is determined on the training samples, the same k value is used for classifying the test samples using the k-nearest neighbors classifier.

To understand the effectiveness of using our easy-to-explain features for RD detection, we compared our results with end-to-end classification networks. To that end, we selected the ResNet-50[16] network with residual connections, which was shown to be effective for various applications by previous studies.[21] We also employed MobileNetV3,[22] which was a lightweight model tuned for mobile applications, to provide insight into the trade-offs between accuracy and computational cost. The weights of these classification networks were initialized with those pretrained on the ImageNet dataset[18] and fine-tuned on our training sets. A weighted cross entropy loss was used to address the class imbalance problem, where the weights were selected inversely proportional to the class frequencies. In training, a batch size was selected as 4, a learning rate is 0.0005, and the optimizer with a weight decay of 0.0002 was used.

Regularizations Techniques for Network Training

Given the limited size of our dataset, we trained our networks using various regularization techniques to alleviate the risk of overfitting. Because the TransUNet and UNet models have relatively complex architectures, we stopped the training through early stopping before the weights became overly large and complex. Apart from that, we applied L2 regularization, also known as *weight decay*, to encourage smaller and smoother weights. We empirically set the coefficients of this regularization to 0.0001 and 0.0002 for the

segmentation and classification networks, respectively. Additionally, in the UNet architecture, we adopted batch normalization layers right after each convolution, providing a regularization effect similar to dropout layers. In the TransUNet network, we used normalization layers within the transformer mechanism, which contributed to generalization in a manner comparable to batch normalization. When fine-tuning the ResNet50 and MobileNetV3 architectures for RD classification, we introduced a batch normalization layer and a dropout layer with a 0.5 probability of neuron masking between the feature extractor and the final classification layer. Furthermore, because we end-to-end trained the entire classification networks on our dataset, there was also an inherent regularization effect of the batch normalization layers following each convolutional layer.

Moreover, we considered data augmentation as another way of regularizing network training and explored its impact by benchmarking the results of networks trained with and without data augmentation. To this end, we implemented data augmentation techniques inspired by the methods proposed for radiology data,[23] adapting them to simulate ocular variations and USG device artifacts. These augmentations included elastic deformations, blurring by a random size Gaussian filter, ghosting effects, anisotropy, bias field artifacts, spike artifacts, gamma correction, and affine transformations such as scaling, rotation, and translation.

Evaluations

We evaluated our segmentation module visually and quantitatively. For the quantitative evaluation, we found the number of true-positive (TP), false-positive (FP), and false-negative (FN) pixels by comparing their ground truth maps with those estimated by the segmentation module. Then, we calculated the precision = TP / (TP + FP), recall = TP / (TP + FN), and F-score = 2 (precision. recall)/(precision + recall) as the performance metrics. Note that there is always a trade-off between precision and recall, and the F-score reflects this trade-off. We used threefold cross-validation in our experiments and reported these performance metrics calculated on the test folds. In addition, because the weights of a neural networks are randomly initialized in its training and because this initialization affects the performance of the network, we repeated the training for each fold 10 times by selecting 10 different sets of initial weights. Thus we had 30 different trials, for each of which we had a test fold. For each test fold, these performance metrics were calculated first on all images (overall) and then separately for the RD and the control groups.

We quantitatively evaluated our classification module for RD detection using precision, recall, and F-score. This time, all these metrics are calculated on image-level annotations. For each of the 30 trials of the segmentation experiments, we calculated the features on the prediction maps of the test fold images and ran the classifier afterward. The average test fold metrics over these 30 runs and their standard deviations are reported. Moreover, since differentiating PVD from RD is crucial, we analyzed the models' performance in this regard. For that, we calculated the percentage of PVD cases correctly classified as non-RD by the proposed pipeline, as well as by the end-to-end classification networks (ResNet50 and MobileNetV3).

Results

Table 2 reports the averages of the performance metrics obtained by the segmentation networks over the 30 test folds and their standard deviations. It shows that the TransUNet model used by the proposed segmentation module led to 84.7%, 78.3%, and 88.2% overall F-scores for the segmentation of the retina/choroid, sclera, and optic nerve sheath structures, respectively. The model was more successful in segmenting these structures in the control group compared to the RD group. This can also be observed in the example predictions shown in Figure 6. Table 2 also presents the test fold results obtained when data augmentation was used to train the TransUNet model. It shows that data augmentation improved performance for optic nerve sheath segmentation but did not significantly improve retina/choroid and sclera segmentation. Because the RD classifier in the proposed pipeline uses features extracted on the segmented retina, this data augmentation only slightly affects the classification results (Table 3).

Moreover, the effectiveness of the transformer mechanism in the TransUNet network was experimentally analyzed in Table 2, which contains the comparative results of TransUNet, both with and without transformer layers, as well as UNet. The comparison with UNet showed that the TransUNet model was more successful in segmentation, especially for the optic nerve sheath, which is also observed in the visual results shown in Figure 6. Moreover, TransUNet demonstrated superior performance compared to UNet in segmenting the regions of detached retina as well as separating such regions from the sclera, which were critical aspects for our classification pipeline.

Table 2. For the Segmentation Module, Test Fold Metrics of the TransUNet Network Trained Without and With Data Augmentation

	TransUNet (W/o Data Augmentation)			TransUNet (With Data Augmentation)			TransUNet (W/o Transformer Layers)			UNet		
	Precision	Recall	F-Score	Precision	Recall	F-score	Precision	Recall	F-score	Precision	Recall	F-Score
Retina/choroid												
Overall	85.7 ± 4.2	85.2 ± 3.1	84.7 ± 1.0	85.2 ± 3.9	85.9 ± 3.2	85.0 ± 0.9	84.5 ± 3.3	83.7 ± 3.4	83.5 ± 1.1	85.7 ± 4.2	83.3 ± 4.1	83.7 ± 1.2
Control	86.7 ± 4.2	86.9 ± 3.2	86.2 ± 1.2	86.5 ± 4.6	87.5 ± 3.3	86.4 ± 1.2	85.4 ± 3.9	86.2 ± 3.5	85.2 ± 1.1	86.1 ± 5.1	86.1 ± 4.2	85.3 ± 1.6
RD	81.2 ± 3.1	79.9 ± 4.3	80.0 ± 1.4	81.4 ± 2.9	80.7 ± 3.9	80.5 ± 1.0	81.4 ± 2.6	75.6 ± 4.2	77.8 ± 1.7	85.5 ± 2.1	72.4 ± 5.1	77.8 ± 2.4
Sclera												
Overall	76.4 ± 3.0	81.3 ± 3.1	78.3 ± 1.4	76.6 ± 3.0	80.7 ± 3.1	78.1 ± 1.6	76.9 ± 2.1	79.1 ± 3.0	77.5 ± 1.4	76.0 ± 4.4	77.8 ± 4.5	76.1 ± 1.9
Control	76.9 ± 3.7	82.6 ± 2.7	79.1 ± 1.9	77.2 ± 4.2	81.6 ± 2.8	78.7 ± 2.2	77.4 ± 2.7	80.9 ± 2.8	78.6 ± 1.7	76.4 ± 5.4	80.5 ± 4.3	77.6 ± 2.9
RD	74.8 ± 2.3	77.2 ± 5.5	75.5 ± 2.9	75.5 ± 1.5	76.9 ± 5.2	75.8 ± 2.8	75.3 ± 2.1	73.2 ± 4.8	73.9 ± 2.8	73.5 ± 4.8	68.6 ± 7.4	70.3 ± 4.9
Optic nerve sheath												
Overall	89.5 ± 3.5	88.4 ± 4.2	88.2 ± 3.7	90.7 ± 2.8	89.9 ± 3.0	89.9 ± 2.8	83.5 ± 7.1	81.3 ± 6.2	81.2 ± 6.2	52.7 ± 8.0	44.9 ± 5.4	47.0 ± 6.2
Control	90.1 ± 3.4	90.0 ± 3.9	89.5 ± 3.3	91.3 ± 2.6	91.5 ± 2.5	91.0 ± 2.4	84.4 ± 6.4	83.7 ± 5.9	83.2 ± 5.9	58.0 ± 9.0	50.4 ± 6.6	52.6 ± 7.5
RD	87.8 ± 7.2	83.2 ± 8.1	83.9 ± 7.9	88.1 ± 6.5	85.2 ± 6.2	85.8 ± 6.2	80.7 ± 10.8	73.6 ± 9.6	74.7 ± 9.8	35.6 ± 10.5	26.7 ± 8.3	28.7 ± 9.0

To understand the effectiveness of the transformer-based attention mechanism in the TransUNet network, these results were compared with the TransUNet model without transformer layers and the UNet model. They are the averages of the 30 runs (three folds and 10 runs for each fold) and their standard deviations. This table reports the metrics for each structure in the eye separately. In addition to presenting these metrics for overall test set images, it also reports them separately for images in the RD and the control groups.

Moreover, the TransUNet model without transformer layers resulted in performance decrease for all ocular structures, particularly for optic nerve sheath segmentation and in cases with retinal detachment. These two comparisons indicate the effectiveness of using an attention mechanism through the transformer layers of the TransUNet model. Thus we will decide to continue with the segmentation results predicted by the TransUNet model in our classification module.

Table 3 reports the classification results (including PVD accuracies) obtained by the proposed pipeline, as well as the ResNet-50 and MobileNetV3 classifiers. These are the average test fold metrics over the 30 runs and their standard deviations. The first part of this table reports the test fold results obtained through three-fold cross-validation on the initial dataset, which was available during the pipeline implementation. It shows that the proposed pipeline could achieve almost perfect precision, i.e., almost all control samples were classified correctly, and it led to high recall and F-scores. It also led to very high PVD accuracies. All metrics were higher than those of the end-to-end classifiers, indicating the effectiveness of using features that provide insight into an ophthalmologist's perspective when diagnosing an eye pathology. Data augmentation improved the results of the classification networks. It did not have this effect on our proposed pipeline since data augmentation did not improve retina segmentation and the features used in our classifier were extracted on the segmented retinal structure. The images in this dataset were labeled by two board-certified ophthalmologists. We also compared the predictions of the proposed pipeline with labels provided by a less experienced clinician, an ophthalmology resident with three years of experience. Table 3 also presents the performance metrics calculated based on the resident's annotations. We found that while the ophthalmology resident successfully distinguished between RD and healthy control cases, there were some misclassifications in RD and PVD cases. In contrast, the proposed pipeline consistently made accurate predictions.

The second part of Table 3 shows the results obtained by the already trained networks on an independent test set, which was not entirely or partially seen during the implementation. This table shows that although classification networks (ResNet50 and MobileNetV3) achieved good test performance on our initial dataset, their performance dropped significantly on this independent test set, regardless of whether data augmentation was applied. On the other hand, our proposed pipeline maintained similar high performance on the independent test set, highlighting the robustness and advantage of using easy-to-explain

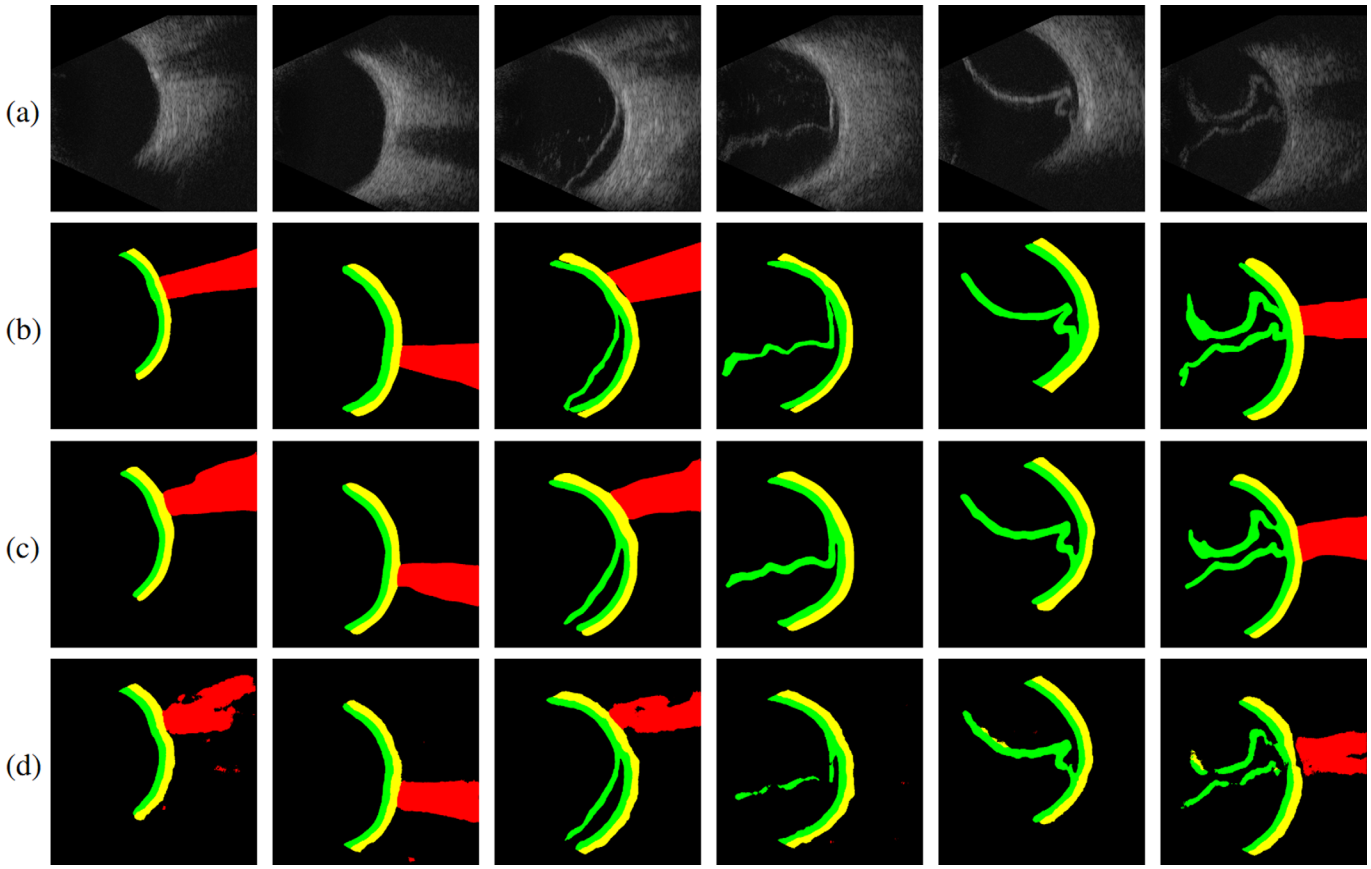

Figure 6. Visual results on exemplary test fold images. (**a**) B-scan ocular USG images and (**b**) ground truth maps. The segmentation maps predicted by the (**c**) TransUNet and (**d**) UNet models. The first two images are from the control group whereas the remaining four are from the RD group. In the maps, *green* correspond to the retina/choroid, *yellow* to the sclera, and *red* to the optic nerve sheath.

features extracted from the maps predicted by the segmentation network.

Last, we compared the training and inference times for the end-to-end classifiers (ResNet50 and MobileNetV3), as well as for the proposed pipeline in Table 4. It shows that training the proposed pipeline took approximately 25% longer, mainly due to the higher number of network parameters in TransUNet. Although the inference time for the proposed pipeline was significantly longer compared to the classification networks, it remained under half a second.

Although end-to-end classification networks have inherently black-box decision-making processes, one can argue that there are techniques to visualize which regions in an image a network classifier pays more attention to when making a decision. For example, the gradient-weighted class activation mapping (Grad-CAM) technique[24] can produce a heatmap of attentions using the gradients of the target class score with respect to the final classification layer. Such heatmaps produced for the ResNet-50 classifier are provided in Figure 7. The first two columns of this figure show the examples of test samples correctly classified by the ResNet-50 classifier. As observed here, the classifier successfully attended the detached retina in Figure 7a. Although this heatmap highlighted the region relevant to the presence of RD, it did not tell anything about the reason of why the classifier attended this region. Moreover, although the classifier correctly classified the sample given in Figure 7b, the highlighted regions were irrelevant to the retinal region; in fact, the least attended (the darkest blue) region in this heatmap corresponded to the retina. On the other hand, segmentation maps along with features defined on them help make relevant and explicit interpretations. Additionally, and more importantly, this may result in defining simpler but more effective classifiers since it allows to reflect the ophthalmologist's insight into the feature definition.

We also investigated the heatmaps of samples correctly classified by our proposed pipeline but incorrectly classified by the ResNet-50 classifier. Figures 7c–e are false-negative examples of the ResNet-50 classifier. Although it attended some parts of the relevant

Table 3. Test Set Performance of Retinal Detachment Detection

	Augmentation	Precision	Recall	F-Score	PVD Accuracy
Test results obtained through threefold cross-validation on the initial dataset					
Easy-to-explain features	False	99.4 ± 1.6	94.0 ± 8.1	96.4 ± 4.6	99.2 ± 0.01
ResNet-50 features	False	94.0 ± 7.4	92.8 ± 6.9	93.1 ± 5.2	96.4 ± 0.03
MobileNetV3	False	93.3 ± 11.1	89.1 ± 10.6	90.3 ± 8.3	94.4 ± 0.07
Easy-to-explain features	True	99.7 ± 1.1	93.4 ± 6.8	96.3 ± 3.8	100.0 ± 0.00
ResNet-50 features	True	96.1 ± 5.8	93.1 ± 7.7	94.3 ± 4.6	97.8 ± 0.02
MobileNetV3	True	97.4 ± 4.1	92.9 ± 4.3	95.0 ± 2.7	97.8 ± 0.03
Manual inference	—	89.6 ± 0.7	78.7 ± 2.8	83.8 ± 1.9	83.3 ± 0.0
Test results obtained on an independent test set					
Easy-to-explain features	False	97.6 ± 4.2	96.0 ± 7.4	96.5 ± 4.4	98.1 ± 3.1
ResNet-50 features	False	92.9 ± 14.8	57.6 ± 23.1	68.6 ± 22.0	99.0 ± 1.7
MobileNetV3	False	94.9 ± 13.8	60.4 ± 23.5	70.4 ± 22.0	98.3 ± 5.3
Easy-to-explain features	True	98.9 ± 2.4	96.9 ± 4.1	97.8 ± 2.4	99.3 ± 1.6
ResNet-50 features	True	88.3 ± 18.4	53.1 ± 29.4	62.1 ± 28.9	98.1 ± 3.5
MobileNetV3	True	97.0 ± 6.5	76.9 ± 14.8	84.9 ± 10.6	99.6 ± 1.3

The last column displays the percentage of PVD cases correctly classified as non-RD. They are the averages of the 30 runs (three folds and 10 runs for each fold) and their standard deviations. The first part of this table reports the test fold results obtained through threefold cross-validation on the initial dataset, which was available during the pipeline implementation. The second part shows the results obtained on an independent test set, which was not entirely or partially seen during the implementation. This independent test set became available after all networks were trained, and it was used to validate the results.

Table 4. Training and Inference Times in Seconds, for the End-to-End Classification Networks and the Whole Proposed Pipeline

	Training	Inference
Easy-to-explain features	76	0.379
ResNet-50 features	61	0.038
MobileNetV3 features	59	0.001

The training time was measured per epoch using a batch size of two, whereas the inference time was the duration required to process a single image.

regions in Figures 7c and 7d, it cannot make correct classification based on these regions. Additionally, it totally focused on irrelevant regions in Figure 7e. For all these images, our proposed pipeline led to correct classification thanks to its effective feature extraction. Here it is worth noting that although retina/choroid layer segmentation was not that accurate in Figures 7c and 7d, our feature definition tolerated this imperfect segmentation (e.g., the maximum number of their column-wise projection was the same in their ground truth and prediction maps) and could be still distinctive for correct classification. The last two images shown in Figures 7f and 7g are false-positive examples of the ResNet-50 classifier. These samples do not exhibit RD but contain mild pathologies of PVD. The ResNet-50 model misclassified these samples with the RD class most probably due to these pathologies; the heatmaps shown in Figures 7f and 7g were also consistent with this explanation. On the other hand, since the vitreous was not annotated as a part of the retina layer, the proposed segmentation module did not identify it as a segmented region, and thus the extracted features did not get affected from these posterior vitreous abnormalities.

Discussion

In this study, we demonstrated the usefulness of the proposed pipeline for RD detection in B-scan ocular USG images. Our results revealed that automated segmentation of posterior segment structures allowed us to define useful features that could significantly improve the detection success. Different than previous studies, our work focused on segmenting multiple structures, including the segmentation of the retina/choroid and sclera for the first time, and demonstrated RD detection based on the easy-to-explain features extracted from the segmented structures.

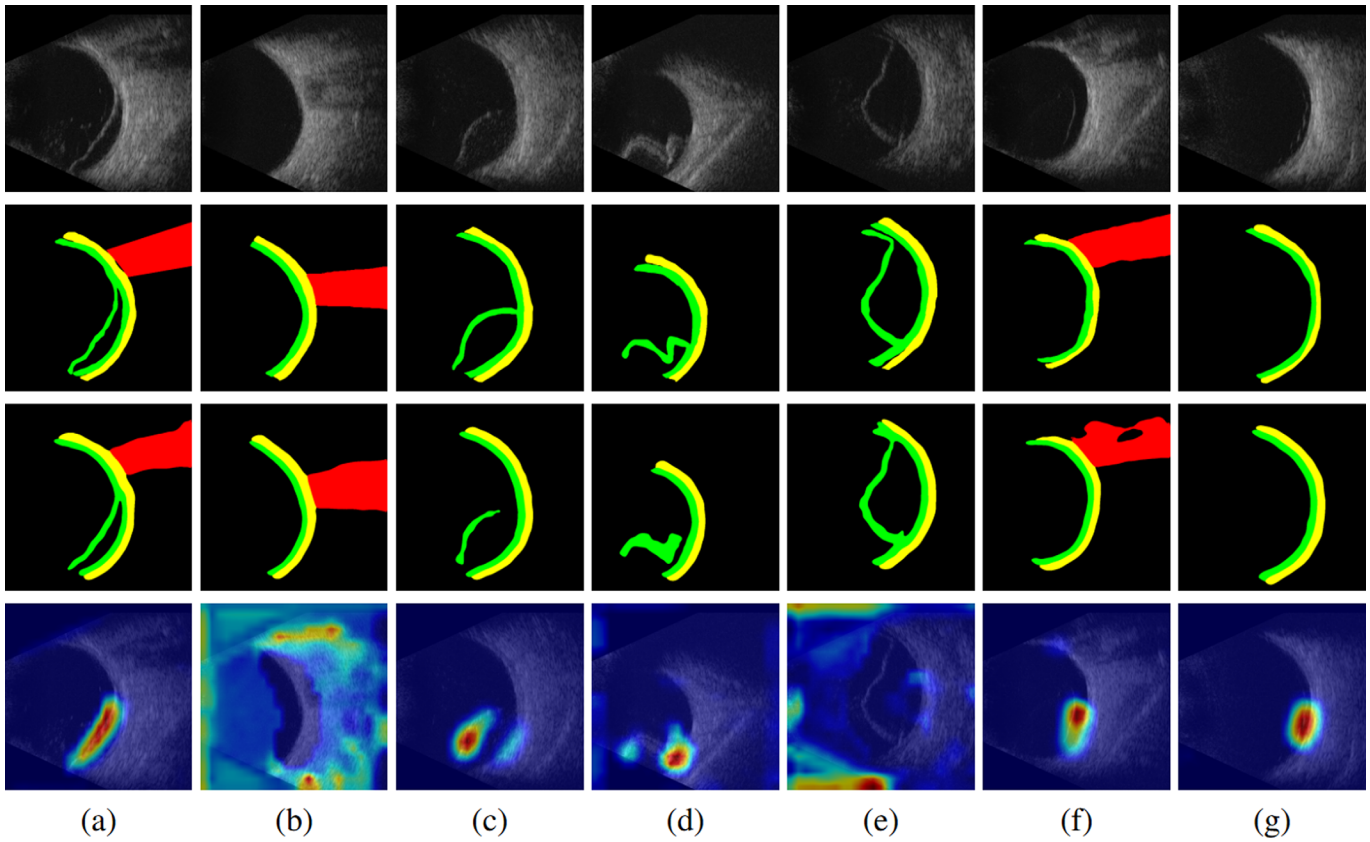

Figure 7. Analysis of the ResNet-50 classifier, which is an end-end classification network. First row: B-scan ocular USG images. Second row: Ground truth maps for the segmentation of multiple structures in the eye. In these maps, the retina/choroid is marked with *green*, the sclera with *yellow*, and the optic nerve sheath with *red*. Third row: Segmentation maps predicted by the TransUNet network in the proposed pipeline. Fourth column: Heatmaps generated by the Grad-CAM technique[19] for the ResNet-50 classifier. These are the example results from the test set folds. (**a**) and (**b**) show true-positive and true-negative examples of the ResNet-50 classifier, respectively. (**c**), (**d**), and (**e**) correspond to false-negative examples of the ResNet-50 classifier whereas (**f**) and (**g**) correspond to false-positive examples. Note that all these samples were correctly classified by the proposed pipeline.

Only a few studies based on deep learning models for B-scan ocular USG images have been reported previously.[7–10,12] Only one of these studies measured the optic nerve sheath diameter based on automated optic nerve sheath segmentation in addition to the eyeball area.[12] Other previous studies centered around classification.[7–10] They trained a classifier on the features extracted from either an entire image[7,8] or the eyeball region[9,10] using pretrained models. The eyeball region was identified by Chen et al.[9] with a UNet-based network and by Zhang et al.[10] with a Yolov3 object detector. Likewise, in a previously reported study, frames containing the optic nerve sheath were detected on a given USG video by the Faster R-CNN algorithm, features were extracted from the bounding boxes detected by the Faster R-CNN also using the pretrained networks, and images were classified based on these features.[11]

Although previous studies used segmentation networks or object detectors to identify regions of interest within an image, they still relied on passing the bounding boxes of these regions to a pretrained neural network, using the features extracted by this network for classification.[9–11] They did not prioritize multistructure segmentation, and their classifiers were dependent on features from pretrained networks, reflecting the black-box nature of these approaches. The features derived from such networks are challenging to interpret, making the decisions based on these features less transparent. The computational pipeline proposed in our study, which performs automatic segmentation of posterior segment structures in B-scan ocular USG images and extracts easily interpretable features that reflect an ophthalmologist's opinion on eye pathology such as RD, has significant potential to address this gap. It constructs an encoder-decoder network and trains it end-to-end to accurately

segment the retina/choroid, sclera, and optic nerve sheath and demonstrates RD detection based on the easy-to-explain features extracted from the segmented structures in an ocular USG image.

The major strength of this study is the design of a strong classifier that relied on segmenting multiple structures by a robust network, namely the TransUNet model with transformer layers, and defining simple but effective features on the segmented structures. As also observed in our experiments, TransUNet can produce more accurate results compared to the networks without transformer layers, especially in segmenting the regions of detached retina and separating these regions from the sclera, which are critical for a classifier to detect RD. The results reported in Table 2 suggested that the segmentation model benefited from long-range dependencies when the ocular structures exhibited spatially dispersed patterns across larger regions in an image, as especially observed in cases with retinal detachment. Furthermore, TransUNet dramatically outperformed UNet in optic nerve sheath segmentation, which had very similar regions across B-scan USG images because of the shadow-like structure of the sheath. This further emphasized the importance of integrating information across a wide spatial context, enabling the model to effectively distinguish between similar regions. All these results supported the claim that encoding global context, through the transformer mechanism, was crucial when the segmentation target exhibited significant variability across the image.

It is worth noting that the ability of our pipeline to define explainable features contrasts with the working principle of classification networks, which are inherently black-box in nature and whose decision-making process is often opaque and difficult to interpret. On the other hand, we define the features that focus on column-wise projections of the segmented retinal regions to mimic the clinical process of evaluating retinal fragmentation. This makes the decision-making process of our pipeline more aligned with the visual assessment of an ophthalmologist in clinical practice, where it is critical to understand how a model arrives at a particular diagnosis.

Additionally, the fact that the segmentation models are more successful in the control group compared to the RD group may be attributed to the following. The variation in the RD group is expected to be greater as RD can be observed in different forms. To better learn this variation, more annotated RD images may be necessary. In our dataset, this number is smaller than the number of control images. Nevertheless, our classification module handled these imperfections with its proposed features and yielded high accuracy for RD detection.

Ocular USG provides an effective alternative to ophthalmoscopic examination in cases where direct visualization of the retina is obstructed, such as in the presence of dense cataracts or vitreous hemorrhage.[4] The proposed pipeline can distinguish RD from healthy as well as from PVD, which is one of the most frequently confused case with RD, with high accuracy under media opacity conditions. This distinction is critical because accurate differentiation between RD that needs to be treated urgently and PVD that does not require treatment can significantly impact treatment decisions and patient outcomes under challenging media opacity conditions. In addition, detecting RD by an automated system can greatly enhance patient care in rural or emergency settings where access to ophthalmic expertise is limited. The portability and low cost of ocular USG, together with the integration of such AI-driven analysis, contributes to the democratization of healthcare in ophthalmology.

This study has the following limitations. First, it included a limited amount of data, especially for cases with complex RD presentations. It is important to expand the data to include a wider variety of RD and other vitreoretinal pathologies. Increasing the number of training data may lead to more accurate results for segmentation and a more generalized solution for classification. Second, our proposed pipeline focuses only on distinguishing RD from healthy and PVD. Although it can successfully distinguish RD from PVD, it does not differentiate between PVD and healthy. Additionally, choroidal detachment was not present in our data set, and retinal and choroidal detachment may not be distinguishable with the current segmentation model. Future studies should investigate the applicability of our approach to other vitreoretinal and choroidal pathologies, such as PVD and vitreous hemorrhage, to expand its clinical use. In this case, features should be defined to reflect the pathology of interest. If this feature definition necessitates segmenting other structures in the eye, the annotation should be provided to the segmentation module. For example, if one wants to extend this pipeline for PVD, vitreous regions should also be segmented. These possibilities are considered as future work.

In conclusion, the proposed computational pipeline with deep learning-based segmentation provides an accurate and interpretable solution for the automated detection of RD in B-scan ocular USG images. Its use of explainable features makes it a valuable tool for clinicians, especially in resource-limited settings where expertise is scarce. Future research should aim to expand the variety of data, refine the segmentation and classification models for more complex cases,

and explore its application to detect other ocular pathologies.

Acknowledgments

Disclosure: **O. Caki**, None; **U.Y. Guleser**, None; **D. Ozkan**, None; **M. Harmanli**, None; **S. Cansiz**, None; **C. Kesim**, None; **R.E. Akcan**, None; **I. Merdzo**, None; **M. Hasanreisoglu**, None; **C. Gunduz-Demir**, None

References

1. De La Hoz Polo M, Torramilans Lluís A, Pozuelo Segura O, Anguera Bosque A, Esmerado Appiani C, Caminal Mitjana JM. Ocular ultrasonography focused on the posterior eye segment: what radiologists should know. *Insights Imaging*. 2016;7:351–364.
2. Pastor JC, Fernández I, Rodríguez de la Rúa E, et al. Surgical outcomes for primary rhegmatogenous retinal detachments in phakic and pseudophakic patients: the Retina 1 Project–report 2. *Br J Ophthalmol*. 2008;92:378–382.
3. Steel D. Retinal detachment. *BMJ Clin Evid*. 2014;2014:0710.
4. Kim DJ, Francispragasam M, Docherty G, et al. Test characteristics of point-of-care ultrasound for the diagnosis of retinal detachment in the emergency department. *Acad Emerg Med*. 2019;26:16–22.
5. Pekala M, Joshi N, Liu TYA, Bressler NM, DeBuc DC, Burlina P. Deep learning based retinal OCT segmentation. *Comput Biol Med*. 2019;114:103445.
6. Cansiz S, Kesim C, Bektas SN, Kulali Z, Hasanreisoglu M, Gunduz-Demir C. FourierNet: Shape-Preserving Network for Henle's Fiber Layer Segmentation in Optical Coherence Tomography Images. *IEEE J Biomed Health Inform*. 2023;27:1036–1047.
7. Li Z, Yang J, Wang X, Zhou S. Establishment and evaluation of intelligent diagnostic model for ophthalmic ultrasound images based on deep learning. *Ultrasound Med Biol*. 2023;49:1760–1767.
8. Adithya VK, Baskaran P, Aruna S, et al. Development and validation of an offline deep learning algorithm to detect vitreoretinal abnormalities on ocular ultrasound. *Indian J Ophthalmol*. 2022;70:1145–1149.
9. Chen D, Yu Y, Zhou Y, et al. A deep learning model for screening multiple abnormal findings in ophthalmic ultrasonography. *Transl Vis Sci Technol*. 2021;10(4):22.
10. Zhang X, Lv J, Zheng H, Sang Y. Attention-Based Multi-Model Ensemble for Automatic Cataract Detection in B-Scan Eye Ultrasound Images. *2020 International Joint Conference on Neural Networks (IJCNN)*. 2020:1–10.
11. Singh M, Kumar B, Agrawal D. Good view frames from ultrasonography (USG) video containing ONS diameter using state-of-the-art deep learning architectures. *Med Biol Eng Comput*. 2022;60:3397–3417.
12. Pang M, Liu S, Lin F, et al. Measurement of optic nerve sheath on ocular ultrasound image based on segmentation by CNN. *2019 IEEE International Conference on Signal, Information and Data Processing (ICSIDP)*. 2019:1–5.
13. Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. *Medical image computing and computer-assisted intervention–MICCAI2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Berlin: Springer International Publishing. 2015:234–241.
14. Siddique N, Sidike P, Elkin C, Devabhaktuni V. U-net and its variants for medical image segmentation: A review of theory and applications. *IEEE Access*. 2021;9:82031–82057.
15. Chen J, Lu Y, Yu Q, et al. TransUNet: Transformers make strong encoders for medical image segmentation. arXiv preprint arXiv:2102.04306, 2021.
16. He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016:770–778.
17. Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16 × 16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*. 2020.
18. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei Li. ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA*. 2009:248–255.
19. Çakı O, Karaçalı B. Quasi-supervised strategies for compound-protein interaction prediction. *Mol Inform*. 2022;41(4):e2100118.
20. Karacali B. Quasi-supervised learning for biomedical data analysis. *Pattern Recognit*. 2010;43:3674–3682.
21. Canziani A, Paszke A, Culurciello E. An analysis of deep neural network models for practical applications. *arXiv preprint arXiv:1605.07678*. 2017.

22. Howard A, Sandler M, Chen B, et al. Searching for MobileNetV3. Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019:1314–1324.
23. Perez-García F, Sparks R, Ourselin S. TorchIO: a Python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning. *Comput Methods Programs Biomed*. 2021;208:106236.
24. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *Int J Comput Vis*. 2020;128:336–359.