Retinal area detector from Scanning Laser Ophthalmoscope (SLO) images for diagnosing retinal diseases

Citation: HALEEM, M.S. ... et al, 2015. Retinal area detector from Scanning Laser Ophthalmoscope (SLO) images for diagnosing retinal diseases. IEEE Journal of Biomedical and Health Informatics, 19 (4), pp.1472-1482

Additional Information:

- This is the accepted manuscript version of the paper. © 2014 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Metadata Record: [https://dspace.lboro.ac.uk/2134/20260](https://dspace.lboro.ac.uk/2134/20260)

Version: Accepted for publication

Publisher: © IEEE

Rights: This work is made available according to the conditions of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) licence. Full details of this licence are available at: https://creativecommons.org/licenses/by-nc-nd/4.0/

Please cite the published version.
Retinal Area Detector from Scanning Laser Ophthalmoscope (SLO) Images for Diagnosing Retinal Diseases

Muhammad Salman Haleem*, Liangxiu Han, Jano van Hemert, Baihua Li and Alan Fleming

Abstract—Scanning Laser Ophthalmoscopes (SLOs) can be used for early detection of retinal diseases. With the advent of latest screening technology, the advantage of using SLO is its wide Field of View (FOV), which can image a large part of the retina for better diagnosis of the retinal diseases. On the other hand, during the imaging process, artefacts such as eyelashes and eyelids are also imaged along with the retinal area. This brings a big challenge on how to exclude these artefacts. In this paper, we propose a novel approach to automatically extract out true retinal area from an SLO image based on image processing and machine learning approaches. To reduce the complexity of image processing tasks and provide a convenient primitive image pattern, we have grouped pixels into different regions based on the regional size and compactness, called superpixels. The framework then calculates image based features reflecting textural and structural information and classifies between retinal area and artefacts. The experimental evaluation results have shown good performance with an overall accuracy of 92%.

Keywords—Scanning Laser Ophthalmoscope, retinal image analysis, feature selection, retinal artefacts extraction.

I. INTRODUCTION

EARLY detection and treatment of retinal eye diseases is critical to avoid preventable vision loss. Conventionally, retinal disease identification techniques are based on manual observations. Optometrists and ophthalmologists often rely on image operations such as change of contrast and zooming to interpret these images and diagnose results based on their own experience and domain knowledge. These diagnostic techniques are time consuming. Automated analysis of retinal images has the potential to reduce the time which clinicians need to look at the images which can expect more patients to be screened and more consistent diagnoses can be given in a time efficient manner [12].

The 2-dimensional retinal scans obtained from imaging instruments (e.g. fundus camera, SLO) may contain structures other than the retinal area; collectively regarded as artefacts. Exclusion of artefacts is important as a pre-processing step before automated detection of features of retinal diseases. In a retinal scan, extraneous objects such as the eyelashes, eyelids and dust on optical surfaces may appear bright and in focus. Therefore, automatic segmentation of these artefacts from an imaged retina is not a trivial task. The purpose of performing this study is to develop a method that can exclude artefacts from retinal scans so as to improve automatic detection of disease features from the retinal scans.

To the best of our knowledge, there is no existing work related to differentiation between the true retinal area and the artefacts for retinal area detection in an SLO image. The SLO manufactured by Optos [2] produces images of the retina with a width of up to 200 degrees (measured from the centre of the eye). This compares to 45-60 degrees achievable in a single fundus photograph. Examples of retinal imaging using fundus camera and SLO are shown in Fig. 1. Due to the wide FOV of SLO images, structures such as eyelashes, eyelids are also imaged along with the retina. If these structures are removed, this will not only facilitate the effective analysis of retinal area, but also enable to register multi-view images into a montage, resulting in a completely visible retina for disease diagnosis.

Fig. 1. An example of (a) a fundus image and (b) an SLO image annotated with true retinal area and Optic Nerve Head

In this work, we have constructed a novel framework for the extraction of retinal area in SLO images. The three main steps for constructing our framework include:

- Determination of features that can be used to distinguish between the retinal area and the artefacts.
- Selection of features which are most relevant to the classification.
- Construction of the classifier which can classify out the retinal area from SLO images.

For differentiating between the retinal area and the artefacts, we have determined different image-based features which reflect grayscale, textural and structural information at multiple
resolutions. Then we have selected the features among the large feature set which are relevant to the classification. The feature selection process improves the classifier performance in terms of computational time. Finally, we have constructed the classifier for discriminating between the retinal area and the artefacts. Our prototype has achieved average classification accuracy of 92\% on the dataset having healthy as well as diseased retinal images.

The rest of this paper is organised as follows: Section II introduces the previous work for feature determination and classification. Section III discusses our proposed method. Section IV provides the quantitative and visual results of our proposed method. Section V summarizes and concludes the method with future work.

II. LITERATURE SURVEY

Our literature survey is initiated with the methods for detection and segmentation of eyelids and eyelashes applied on images of the front of the eye, which contains the pupil, eyelids and eyelashes. On such an image, the eyelashes are usually in the form of lines or bunch of lines grouped together. Therefore the first step of detecting them was the application of edge detection techniques such as Sobel, Prewit, Canny, Hough Transform [11] and Wavelet transform [5]. The eyelashes on the iris were then removed by applying non-linear filtering on the suspected eyelash areas [29]. Since eyelashes can be in either separable form or in the form of multiple eyelashes grouped together, Gaussian filter and Variance filter were applied in order to distinguish among both forms of eyelashes [20]. The experiment showed that separable forms of eyelashes were most likely detected by applying Gaussian filter whereas Variance filters are more preferable for multiple eyelash segmentation [17]. Initially, the eyelash candidates were localised using Active Shape Modeling(ASM) and then 8-directional filter bank was applied on the possible eyelash candidates. Kang et al. [16] used Focus Score in order to vary the size of convolution kernels for eyelash detection. The size variation of the convolution kernels also differentiated between separable eyelashes and multiple eyelashes. Min et al. [19] determined the features based on intensity and local standard variation in order to determine eyelashes. They were thresholded using Otsu’s method which is an automatic threshold selection method based on particular assumptions about intensity distribution. All of these methods have been applied on CASIA database [1] which is an online database of Iris images. In an image obtained from SLO, the eyelashes show as either dark or bright region compared to retinal area depending upon how laser beam is focused as it passes the eyelashes. The eyelids show as reflectance region with greater reflectance response compared to retinal area. Therefore, we need to find out features which can differentiate among true retinal area and the artefacts in SLO retinal scans. After visual observation in Fig. 1(b), the features reflecting the textural and structural difference could have been the suggested choice. These features have been calculated for different regions in fundus images; mostly for quality analysis.

The characterisation of retinal images were performed in terms of image features such as intensity, skewness, textural analysis, histogram analysis, sharpness etc [8], [12], [28]. Dias et al. [9] determined four different classifiers using four types of features. They were analyzed for the retinal area including colour, focus, contrast and illumination. The output of these classifiers were concatenated for quality classification. For classification, the classifiers such as Partial Least Square (PLS) [6] and Support Vector Machines (SVM) [22] were used. PLS selects the most relevant features required for classification. Apart from calculating image features for whole image, grid analysis containing small patches of the image has also been proposed for reducing computational complexity [8]. For determining image quality, the features of Region of Interest (ROI) of anatomical structures such as Optic Nerve Head (ONH) and Fovea have also been analyzed [23]. The features included structural similarity index, area and visual descriptor etc. Some of the above mentioned techniques suggest the use of grid analysis which can be an time effective method to generate features of particular region rather than each pixel. But grid analysis might not be an accurate way to represent irregular regions in the image. Therefore, we decided the use of superpixels [4], [21], [26], [27] which group pixels into different regions depending upon their regional size and compactness.

Our methodology is based on analyzing the SLO image based features which are calculated for a small region in the retinal image called superpixels. The determination of feature vector for each superpixel is computationally efficient as compared to feature vector determination for each pixel. But grid analysis might not be an accurate way to represent irregular regions in the image. Therefore, we decided the use of superpixels [4], [21], [26], [27] which group pixels into different regions depending upon their regional size and compactness.

Our methodology is based on analyzing the SLO image based features which are calculated for a small region in the retinal image called superpixels. The determination of feature vector for each superpixel is computationally efficient as compared to feature vector determination for each pixel. But grid analysis might not be an accurate way to represent irregular regions in the image. Therefore, we decided the use of superpixels [4], [21], [26], [27] which group pixels into different regions depending upon their regional size and compactness.

Our methodology is based on analyzing the SLO image based features which are calculated for a small region in the retinal image called superpixels. The determination of feature vector for each superpixel is computationally efficient as compared to feature vector determination for each pixel.
superpixels. The generation of the feature vector for each superpixel makes the process computationally efficient as compared to feature vector generation for each pixel.

- **Feature Generation**: We generate image-based features which are used to distinguish between the retinal area and the artefacts. The image-based features reflect textural, grayscale or regional information and they were calculated for each superpixel of the image present in the training set. In testing stage, only those features will be generated which are selected by Feature Selection process.

- **Feature Selection**: Due to a large number of features, the feature array needs to be reduced before classifier construction. This involves features selection of the most significant features for classification.

- **Classifier Construction**: In conjunction with manual annotations, the selected features are then used to construct the binary classifier. The result of such a classifier is the superpixel representing either the ‘true retinal area’ or the ‘artefacts’.

- **Image Postprocessing**: Image postprocessing is performed by Morphological filtering so as to determine the retinal area boundary using superpixels classified by the classification model.

The elements of our detection framework are discussed as follows:

### A. Image Preprocessing

Images were normalized by applying a Gamma (\(\gamma\)) adjustment to bring the mean image intensity to a target value. \(\gamma\) was calculated using

\[
\gamma = \frac{\log_{10}(\mu_{\text{target}}) - \log_{10}(255)}{\log_{10}(\mu_{\text{orig}}) - \log_{10}(255)}
\]  

(1)

where \(\mu_{\text{orig}}\) is mean intensity of the original image and \(\mu_{\text{target}}\) is mean intensity of the target image. For image visualization, \(\mu_{\text{target}}\) is set to 80. Finally, the Gamma adjustment of the image is given by Equation 2.

\[
I_{\text{norm}} = \left( \frac{I}{255} \right)^\gamma.
\]  

(2)

### B. Generation of Superpixels

The superpixel algorithm groups pixels into different regions, which can be used to calculate image features while reducing the complexity of subsequent image processing tasks. Superpixels capture image redundancy and provide a convenient primitive image pattern. As far as fundus retinal images are concerned, the superpixels have been generated for analyzing anatomical structures [7] and retinal hemorrhage detection [25]. For retinal hemorrhage detection, the superpixels were generated using Watershed approach but the number of superpixels generated in our case need to be controlled. The watershed approach sometimes generates number of superpixels of the artefacts more than desired. The superpixel generation method used in our retina detector framework is Simple Linear Iterative Clustering (SLIC) [4], which was shown to be efficient not only in terms of computational time, but also in terms of region compactness and adherence. The algorithm is initialized by defining number of superpixels to be generated. The value was set to 5000 as a compromise between computational stability and prediction accuracy.

### C. Feature Generation

After the generation of superpixels, the next step is to determine their features. We intend to differentiate between the retinal area and artefacts using textural, grayscale-gradient and regional based features. Textural and gradient based features are calculated from red and green channels on different Gaussian blurring scales; also known as smoothing scales [3]. In SLO images, the blue channel is set to zero therefore no feature was calculated for the blue channel. The regional features are determined for the image irrespective of the colour channel. The details of these features are described as follows:

1) **Textural Features**: Texture can be analyzed using Haralick features [14] by Grey Level Co-occurrence Matrix (GLCM) analysis. GLCM determines how often a pixel of a grey scale value \(i\) occurs adjacent to a pixel of the value \(j\). Four angles for observing the pixel adjacency i.e. \(\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ\) are used. These directions are shown in Fig. 3(a). GLCM also needs an offset value \(D\) which defines pixel adjacency by certain distance. In our case, offset value is set to 1. Fig. 3(b) illustrates the process of creating GLCM using the image \(I\). The features, which are calculated using GLCM matrix are summarised in Table I. The mean value in each direction was calculated for the blue channel. The regional features are determined using GLCM matrix.

2) **Gradient Features**: The reason for including gradient features was illumination non-uniformity of the artefacts. In order to calculate these features, the response from Gaussian filter bank [3] is calculated. The Gaussian filter bank includes...
Fig. 3. (a) GLCM directions and offset (b) GLCM process using image I [13]

Gaussian $\mathcal{N}(\sigma)$, its two first order derivatives $\mathcal{N}_x(\sigma)$ and $\mathcal{N}_y(\sigma)$ and three second order derivatives $\mathcal{N}_{xx}(\sigma)$, $\mathcal{N}_{xy}(\sigma)$ and $\mathcal{N}_{yy}(\sigma)$ in horizontal($x$) and vertical($y$) directions. After convolving the image with the filter bank at a particular channel, the mean value is taken over of each filter response over all pixels of each superpixel.

3) Regional Features: The features used to define regional attributes were included because superpixels belonging artefacts have irregular shape compared to those belonging the retinal area in an SLO image. Table II represents the features describing regional attributes.

The image features are calculated for each superpixel of the images present in the training set and they form a matrix of the form as shown in equation 3.

$$FM = \begin{bmatrix}
A^{tex}_R & A^{tex}_G & A^g_R & A^g_G & A^{re} \\
B^{tex}_R & B^{tex}_G & B^g_R & B^g_G & B^{re}
\end{bmatrix}$$

(3)

where $A$ and $B$ represent class of true retinal area and class of artefacts, superscripts $tex, re, g$ represent textural features, regional features and gradient based features respectively; and subscript $R$ and $G$ represent the red and green channel respectively. For determining features at different smoothing scales, both red and green channels of images are convolved with the Gaussian [3] at scales $\sigma = 1, 2, 4, 8, 16$. The textural features are calculated at the original scale as well as at five different smoothing scales so as to accommodate their image response in the training set after blurring. In this way the total number of columns in both channels of $A^{tex}$ and $B^{tex}$ will be 114 making it 228 altogether. The gradient features has 6 columns in each scale making 30 columns for each channel of $A^g$ and $B^g$ so 60 columns in total for each superpixel. As far as regional features are concerned, except $I_{\mu}$, they are independent of channel variation. Therefore they are calculated only once for the superpixel so 7 columns for $A^{re}$ and $B^{re}$ ($I_{\mu}$ is calculated for both red and green channels). In this way there are the total number of 295 features in the feature matrix for each superpixel of the image present in the training set. Each column of the feature matrix calculated for the particular image is normalized using z-score normalization [10]. Z-score normalization returns the scores of the column with zero mean and unit variance.

D. Feature Selection

The main purposes for feature selection are reducing execution time, determination of features most relevant to the classification and dimensionality reduction. For feature selection, we have selected Sequential Forward Selection (SFS) Approach.

In the ‘SFS approach’, the interaction among features is taken into account. From the available set of features, the feature with the highest Area Under the Curve (AUC) [24] is selected. The next feature is chosen in such a way that when it is used along with the first selected feature, it will give the highest AUC compared to other non-selected features. The process is repeated until 10 features were selected since a higher number of features resulted in a very small improvement in AUC.

The performance of the SFS has been compared against other feature selection approaches such as ‘Filter Approach’ and ‘Filter and SFS’ approach. In the filter approach, the features are ranked with respect to their effectiveness in classification and higher ranked features are thresholded out. In order to determine most relevant features, an independent evaluation criterion for binary classification is used [18] and AUC is selected as its evaluation measure [24]. The features with higher AUC are ranked higher and the features with AUC greater than 0.9 are selected. In this way, 33 features are selected for classifier construction. The ‘Filter and SFS’ approach is similar to SFS approach except that it is applied on the filtered feature set rather than complete feature set.
<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Equation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autocorrelation</td>
<td>$a_{corr} = \sum_{i,j} ijp(i,j)$</td>
<td>Linear dependence in GLCM between same index</td>
</tr>
<tr>
<td>Cluster Shade</td>
<td>$C_{shade} = \sum_{i,j} (i + j - \mu_x - \mu_y)^3 p(i,j)$</td>
<td>Measure of skewness or non-symmetry</td>
</tr>
<tr>
<td>Cluster Prominence</td>
<td>$C_{prom} = \sum_{i,j} (i + j - \mu_x - \mu_y)^4 p(i,j)$</td>
<td>Show peak in GLCM around the mean for non-symmetry</td>
</tr>
<tr>
<td>Contrast</td>
<td>$con = \sum_{i,j} \left</td>
<td>i - j \right</td>
</tr>
<tr>
<td>Correlation</td>
<td>$corr = \frac{\sigma_x \sigma_y}{\sigma_{xy}}$</td>
<td>Linear dependence in GLCM between different index</td>
</tr>
<tr>
<td>Difference Entropy</td>
<td>$H_{diff} = -\sum_{i=0}^{N_y-1} p_{x,y}(i) \log(p_{x,y}(i))$</td>
<td>Higher weight on higher difference of index entropy value</td>
</tr>
<tr>
<td>Dissimilarity</td>
<td>$diss = \sum_{i,j} \left</td>
<td>i - j \right</td>
</tr>
<tr>
<td>Energy</td>
<td>$E = \sum_{i,j} p(i,j)^2$</td>
<td>Returns the sum of squared elements in the GLCM</td>
</tr>
<tr>
<td>Entropy</td>
<td>$H = -\sum_{i,j} p(i,j) \log(p(i,j))$</td>
<td>Texture randomness producing a low value for an irregular GLCM</td>
</tr>
<tr>
<td>Homogeneity</td>
<td>$homom = \frac{1}{4N_y} \sum_{i,j} \left( \frac{1}{1 + \left</td>
<td>i - j \right</td>
</tr>
<tr>
<td>Information Measures</td>
<td>$IM_1 = \left( 1 - \exp[-2.0(H_{xy} - H)] \right)^{0.5}$</td>
<td>Entropy measures</td>
</tr>
<tr>
<td>Information Measures</td>
<td>$IM_2 = \frac{E_{Entropy} - H_{xy}}{H_{xy}}$</td>
<td>Entropy measures</td>
</tr>
<tr>
<td>Inverse Difference Normalized</td>
<td>$IDN = \sum_{i,j} \left( \frac{p(i,j)}{N_y^2} \right)$</td>
<td>Inverse Contrast Normalized</td>
</tr>
<tr>
<td>Inverse Difference Moment Normalized</td>
<td>$IDMN = \sum_{i,j} \left( \frac{p(i,j)}{N_y^2} \right)^{1/2}$</td>
<td>Homogeneity Normalized</td>
</tr>
<tr>
<td>Maximum Probability</td>
<td>$Pr_{max} = \frac{\max {p(i,j)}}{\sum_{i,j} p(i,j)}$</td>
<td>Maximum value of GLCM</td>
</tr>
<tr>
<td>Sum Entropy</td>
<td>$H_{sum} = -\sum_{i=2}^{N_y} p_{x+y}(i) \log(p_{x+y}(i))$</td>
<td>Higher weight on higher sum of index entropy value</td>
</tr>
<tr>
<td>Sum of Squares: Variance</td>
<td>$\sigma_{sum} = \sum_{i,j} \left( (i - \mu)^2 \right) p(i,j)$</td>
<td>Higher weights that differ from average value of GLCM</td>
</tr>
<tr>
<td>Sum of Variance</td>
<td>$\sigma_{sum} = \sum_{i,j} \left( i - H_{sum} \right) p_{x+y}(i)$</td>
<td>Higher weights that differ from entropy value of marginal GLCM</td>
</tr>
</tbody>
</table>

(i,j) represent rows and columns respectively. $N_y$ is number of distinct grey levels in the quantised image, $p(i,j)$ is the element from normalized GLCM matrix $p_x(i)$ and $p_y(j)$ are marginal probabilities of matrix obtained by summing rows and columns of GLCM respectively i.e. $p_x(i) = \sum_{j=1}^{N_y} p(i,j)$, $p_y(j) = \sum_{i=1}^{N_x} p(i,j)$ and $p_{x+y}(k) = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} p(i,j)$, $k = i + j - 1 = 1, 2, 3, \ldots, N_x N_y$ and $p_{x-}\bar{y}(k) = \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} p(i,j), k = |i - j| + 1 = 1, \ldots, N_y$, $H_y$ and $H_x$ and entropies of $p_x$ and $p_y$ respectively, $H_{xy} = -\sum_{i,j} p_x(i)p_y(j) \log(p_x(i)p_y(j))$, $H_{xy2} = -\sum_{i,j} p(i,j) \log(p_x(i)p_y(j))$.

Table I. Textural Features extracted using GLCM

The individual and collective performance of the features selected in the feature sets from the above mentioned approaches are shown in Fig. 4 and Fig. 5. The axis of ‘Feature Index’ in Fig. 4 is ordered according to descending independent evaluation criterion. The axis of ‘Number of Selected Features’ in Fig. 5 represents the order with which the features are selected using the SFS approach. We have not applied SFS on the ‘Filter Approach’, therefore axis of ‘Number of Selected Features’ for ‘Filter Approach’ would be same as that of ‘Feature Index’ in Fig. 4. The features represented by ‘Feature Index’ and ‘Number of Selected Features’ are shown in Table III. SFS is computationally intensive as it required 5 minutes / feature on filtered feature set and 30 minutes/feature on complete feature set. But, the results show that the SFS approach performed better compared to other two approaches despite of the fact that the feature set also consists of those features which ranked low in independent evaluation criterion. The Table IV represents the percentage of different types of features selected in each feature set. The table shows clear dominance of textural features compared to gradient features.
TABLE II. REGIONAL FEATURES

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Equation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Intensity</td>
<td>( I_{\mu} = \frac{\sum \sum I_s(i,j)}{N_s} )</td>
<td>Mean value of Superpixel</td>
</tr>
<tr>
<td>Area</td>
<td>( N_s )</td>
<td>Number of pixels in Superpixel</td>
</tr>
<tr>
<td>Convex Area</td>
<td>( N_{sc} )</td>
<td>Number of pixels in convex area of Superpixel</td>
</tr>
<tr>
<td>Extent</td>
<td>( Ext = \frac{N_s}{N_{sb}} )</td>
<td>Ratio of area to number of pixels in the bounding box</td>
</tr>
<tr>
<td>Orientation</td>
<td>( \theta_s )</td>
<td>Superpixel angle with respect to x-axis</td>
</tr>
<tr>
<td>Solidity</td>
<td>( Sol = \frac{N_s}{N_{sc}} )</td>
<td>Ratio of area to convex area</td>
</tr>
</tbody>
</table>

and regional features.

TABLE III. FEATURE SETS OBTAINED USING DIFFERENT FEATURE SELECTION APPROACHES. ‘FEATURE INDEX’ REPRESENTS THE ORDER OF HIGHEST INDEPENDENT EVALUATION CRITERION MEASURE. ‘NUMBER OF SELECTED FEATURES’ REPRESENT THE SEQUENCE OF FEATURE SELECTION IN THE FEATURE SET. R AND G SUBSCRIPTS REPRESENT RED AND GREEN CHANNEL RESPECTIVELY

<table>
<thead>
<tr>
<th>Feature Selection Method</th>
<th>Feature Symbols</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter Approach (Feature Index and Number of Selected Features)</td>
<td>( \mu_{sumR}(16), \sigma_{sumR}(16), \mu_{sumR}(8), N_R(16), \mu_{sumR}(4), \sigma_{sumR}(8), \mu_{sumR}(2), \mu_{sumR}(1), \mu_{sumR}, \sigma_{sumR}(4), \sigma_{sumR}(2), N_R(8), \sigma_{sumR}(1), acorr_R(16), \sigma_{sosR}(16), \sigma_{sumR}, N_R(4), N_R(2), N_R(1), N_{yyR}(1), I_{\mu R}, N_{xxR}(1), acorr_R(8), \sigma_{sosR}(8), \sigma_{sosR}(4), \sigma_{sosR}(2), acorr_R(1), \sigma_{sosR}(1), acorr_R, \sigma_{sosR} )</td>
</tr>
<tr>
<td>Filter and SFS Approach (Feature Index)</td>
<td>( \mu_{sumR}(16), \sigma_{sumR}(16), \sigma_{sumR}(8), \mu_{sumR}, \sigma_{sumR}(4), \sigma_{sumR}, acorr_R(16), \sigma_{sosR}(16), \sigma_{sosR}(8), \sigma_{sumR}, \sigma_{sosR}(8), \sigma_{sosR}(2), \sigma_{sosR}(16), \sigma_{sosR}(8), N_R(1), acorr_R(1), \sigma_{sumR}(4) )</td>
</tr>
<tr>
<td>Filter and SFS Approach (Number of Selected Features)</td>
<td>( \mu_{sumR}(16), \sigma_{sosR}(1), \sigma_{sumR}(8), \sigma_{sosR}(8), \sigma_{sumR}(16), \sigma_{sumR}, \sigma_{sumR}, acorr_R(8), acorr_R(1), \sigma_{sumR}(4) )</td>
</tr>
<tr>
<td>SFS Approach (Feature Index)</td>
<td>( \mu_{sumR}(16), acorr_R(8), \sigma_{sosR}(8), \sigma_{sumG}, acorr_G, \sigma_{sosG}, H_G(8), N_{yyR}(16), H_G(1), H_{diffG}(1) )</td>
</tr>
<tr>
<td>SFS Approach (Number of Selected Features)</td>
<td>( \mu_{sumR}(16), \sigma_{sosG}, H_G(8), N_{yyR}(16), \sigma_{sosR}(8), H_{diffG}(1), acorr_G, acorr_R(8), \sigma_{sumG}, H_G(1) )</td>
</tr>
</tbody>
</table>

E. Classifier Construction

The classifier is constructed in order to determine the different classes in a test image. In our case, it is a two...
class problem: true retinal area and artefacts. We have applied Artificial Neural Networks (ANN). The ANN is the classification algorithm that is inspired by human and animal brain. It is composed of many interconnected units called artificial neurons. ANN takes training samples as input and determines the model that best fits to the training samples using non-linear regression. Consider the Fig. 6 which shows three basic blocks of ANN i.e. input, hidden layer (used for recoding or providing representation for input) and output layer. More than one hidden layer can be used but in our case, there is only one hidden layer with 10 neurons. The output of each layer is in the form of matrix of floating values which can be obtained by sigmoid function as:

\[ h_W(x) = \frac{1}{1 + \exp(-W^T x + b)} \]  

where \( b \) is the bias value and \( W \) are the weights of input \( x \). These weights can be determined by backpropagation algorithm which tends to minimize mean square error value between desired output and actual output as:

\[ err = \frac{1}{2} (t - y)^2 \]  

where \( t \) and \( y \) represent the target output and actual output of the output layer. The minimization of equation 5 can be represented as:

\[ \partial err / \partial W_i = (y - t)y(1 - y)x_i \]  

Since it is an iterative process therefore weights are updated by delta rule as

\[ \Delta w_i = \alpha (t - y)x_i \]  

\( \alpha \) represents the step size. The weights were updated until 1000 iterations.

F. Image Postprocessing

After classification of the test image, the superpixels are refined using morphological operation [11] so as to remove misclassified isolated superpixels. The morphological closing was performed so as to remove small gaps among superpixels. The size of disk structuring element can be a smaller value; say 10. For better results, we can perform area opening so as to remove one or two misclassified isolated superpixels.

G. A Comparison Study

After the construction of our classifier, we have compared its performance against different classifiers in terms of accuracy and computational time. The classifiers have been applied across different feature sets which are obtained by using different feature selection procedures as mentioned in Section III-D. The classifiers we have selected for comparing the performance of our classifier are Support Vector Machines (SVM) and \( k \) Nearest Neighbours (\( kNN \)) [10].

The idea behind \( kNN \) method is to find out samples whose feature are similar to the classes to be detected. The function, which we are following in order to determine the similarity of the features with true retinal area is ‘Euclidean Distance’. SVM finds a separating hyperplane with the maximal margin in higher dimensional space. In our comparison study, we are using non-linear SVM with Radial Based Function (RBF) kernel with default parameter of \(( \text{number of features} )^{-1} = 0.1 \) [15].

IV. EXPERIMENTAL EVALUATION

The images for training and testing have been obtained from Optos [2] and are acquired using their ultra-wide field SLO. Each image has a FOV of up to 200 degrees of the retina in a resolution of 14 \( \mu \)m. The device captures the retinal image without dilation, through a small pupil of 2 mm. The image has two channels: red and green. The green channel (wavelength: 532nm) provides information about the sensory retina to retinal pigment epithelium whereas the red channel (wavelength: 633nm) shows deeper structures of the retina towards the choroid. Each image has a dimension of 3900 \( \times \) 3072 and each pixel is represented by 8-bit on both red and green channels. The dataset is composed of healthy and diseased retinal images; most of the diseased retinal images are from Diabetic Retinopathy patients. The system has been trained with 28 images and tested against 76 images.

Fig. 7 compares the classification power of different feature sets with the help of Receiver Operating Characteristics (ROC). One of those feature sets include all features calculated. The rest of other feature sets include features selected by the approaches discussed in Section III-D. By using SFS approach, 10 features out of 295 features have been selected and their calculation time is 25 seconds per image whereas calculating the complete feature set can take around 10 minutes per image. The ROC curves and AUC values reveal that if the features are selected using the SFS approach, they can have a classification power almost similar to the complete feature set while reducing the computational time.
The visual results and the accuracies of different classifiers among different feature sets have been presented using Dice Coefficient as evaluation metric. The Dice Coefficient is the degree of overlap between the framework output and the benchmark obtained from the clinician. The Dice Coefficient is defined in equation 8:

\[ D(A, B) = \frac{2|A \cap B|}{|A| + |B|} \] (8)

where \( A \) and \( B \) are the segmented images obtained from the framework and the benchmark respectively, \(|.|\) represents number of samples of the region, and \( \cap \) denotes the intersection. Its value varies between 0 and 1 where a higher value indicates an increased degree of overlap. Let \( RA_1 \) and \( AR_1 \) represent samples from the retinal area and the artefact area obtained from the framework respectively and \( RA_2 \) and \( AR_2 \) be these samples from the benchmark. The class of superpixels in the benchmark was decided based on majority of pixels in the superpixel belonging to particular class. Also, \( |RA_1| + |AR_1| = |RA_2| + |AR_2| = N_{sample} \) i.e. total number of samples (superpixels or pixels) in an image. If we calculate Dice Coefficient for the image, equation 8 can be deduced as:

\[ D_I = \frac{(|RA_1 \cap RA_2| + |AR_1 \cap AR_2|)}{N_{sample}} \] (9)

The Dice Coefficient for the retinal area \( D_R \) and artefacts \( D_A \) will be given as:

\[ D_R = \frac{2|RA_1 \cap RA_2|}{|RA_1| + |RA_2|}, D_A = \frac{2|AR_1 \cap AR_2|}{|AR_1| + |AR_2|} \] (10)

Table V compares the performance of different classifiers across different feature sets. As far as classification accuracy is concerned there is little difference among the outputs of different classifiers. The advantage of using ANN is its high computational efficiency in terms of testing time as shown in Table VI. Although the training time of ANN is longer compared to its other two counterparts, the training time is once in a lifetime process and once the model is deployed, it can process any image. Fig. 9 represents the total time taken by an image to be processed for automatic annotations. The block diagram and the Table VI shows that while using ANN, couple of seconds can be saved per image during automatic annotation process. As shown in Table V, SVM although performed better on SFS feature set compared to ANN and \( kNN \), ANN has the highest classification accuracy in other two feature sets. This shows that classification accuracy is highly dependent on type of features selected.

Fig. 8 shows superpixel classification results and final output after post processing of different examples of healthy and diseased retinal images. ANN is able to achieve the average accuracy nearer to that of other two classifiers while saving significant computational time when processing millions of images for automatic annotations.
Fig. 8. Superpixel Classification Result of two examples of SLO images. Columns represent different examples of retinal images. Left column are retinal scans with lesions whereas right column is the retinal scan from healthy subject. Figures (a) to (b) represent the test images divided into superpixels. Figures (c) to (d) represent superpixel classification results and Figures (e) to (f) represent output after post processing.
Although other two classifiers, the training and testing time is quite long. The overall accuracy of SVM was the highest compared to other two classifiers. Although the process and we can compromise on computational time for that selection of features based on their mutual interaction can provide the classification power close to that of feature selection approach surpassed the filter approach and ‘Filter and SFS’ approaches in terms of classification power. The comparison of different feature selection approaches shows that selection of features based on their mutual interaction can provide the classification power close to that of feature set with all features. Feature selection is once in a life time process and we can compromise on computational time for feature selection on account of accuracy.

As far as the classifier is concerned, the testing time of ANN was the lowest compared to other two classifiers. Although the overall accuracy of SVM was the highest compared to other two classifiers, the training and testing time is quite long. Although kNN has the shortest training time, the testing time can be quite high compared to ANN while processing millions of images. Compared to SVM, we can trade-off the overall accuracy of 0.1% on average while saving the testing time of 8 seconds per image. As far as images with lesions are concerned (see Fig. 8 (a),(c),(e)), ANN misclassified 1 or 2 superpixels at the corners but they are corrected using morphological post-processing as shown in Fig. 8 (e).

Our retina detection framework serves as the first step towards the processing of ultra-widefield SLO images. A complete retinal scan is possible if the retina is imaged from different eye-steered angles using an ultra-widefield SLO and then montaging the resulting image. Montaging is possible only if the artefacts are removed before.

V. DISCUSSION AND CONCLUSION

Distinguishing true retinal area from artefacts in SLO images is a challenging task, which is also the first important step towards computer-aided disease diagnosis. In this work, we have proposed a novel framework for automatic detection of true retinal area in SLO images. We have used superpixels to represent different irregular regions in a compact way and reduce the computing cost. Feature selection enables the most significant features to be selected and thus reduces computing cost too. A classifier has been built based on selected features to extract out the retina area. It has been compared with other two classifiers and was compatible while saving the computational time. The experimental evaluation result shows that our proposed framework can achieve an accuracy of 92% in segmentation of the true retinal area from an SLO image.

Feature selection is necessary so as to reduce computational time during training and classification. Among different approaches used for feature selection, the performance of our feature selection approach surpassed the filter approach and ‘Filter and SFS’ approaches in terms of classification power. The comparison of different feature selection approaches shows that selection of features based on their mutual interaction can provide the classification power close to that of feature set with all features. Feature selection is once in a life time process and we can compromise on computational time for feature selection on account of accuracy.

ACKNOWLEDGMENT

This work is supported by EPSRC-DHPA funded project “Automatic Detection of Features in Retinal Imaging to Improve Diagnosis of Eye Diseases” and Optos plc. The authors would also like to thank the anonymous reviewers, who provided detailed and constructive comments on an earlier version of this paper.

TABLE VI. COMPARISON OF FRAMEWORK OUTPUT PERFORMANCE USING DIFFERENT CLASSIFIERS. THE PERFORMANCE IS COMPARED WITH RESPECT TO COMPUTATIONAL TIME TAKEN DURING TRAINING AND TESTING AND AVERAGE ACCURACY. THE TRAINING TIME IS CALCULATED FOR 28 IMAGES. TESTING TIME SHOWS THE AVERAGE TIME TAKEN BY THE FRAMEWORK

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Training Time</th>
<th>Testing Time</th>
<th>$D_I$</th>
<th>$D_R$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN</td>
<td>30 minutes</td>
<td>0.013 seconds</td>
<td>91.93%</td>
<td>91.87%</td>
</tr>
<tr>
<td>SVM</td>
<td>12.5 minutes</td>
<td>8.5 seconds</td>
<td>92.00%</td>
<td>91.94%</td>
</tr>
<tr>
<td>kNN</td>
<td>1.45 seconds</td>
<td>2.05 seconds</td>
<td>91.43%</td>
<td>91.31%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Image Dimension: 3900 x 3072</th>
<th>1.14 seconds</th>
<th>20 seconds</th>
<th>25 seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Pre-processing</td>
<td>Image</td>
<td>Super-pixel</td>
<td>Feature</td>
</tr>
<tr>
<td>Annotation</td>
<td>Generation</td>
<td>Generation</td>
<td></td>
</tr>
<tr>
<td>Total Time ~ 50 seconds</td>
<td>5 seconds</td>
<td>0.013 seconds</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 9. Block diagram of Deployment Stage along with execution time of each block.

REFERENCES


