KNN ALGORITHM IN A FRAMEWORK OF SCALE-SPACE THEORY FOR RETINAL IMAGE ANALYSIS

Krit Inthajak*, Cattleya Duanggate, Bunyarit Uyyanovara, and Stanislav Makhanov

Received: December 01, 2012; Revised: April 17, 2013; Accepted: April 19, 2013

Abstract

This topic presents a framework in the uses of the K-nearest neighbor algorithm in evaluating an object detection method of scale-space theory with feature stability. A scale-space tree is constructed based on the blobs that were created from a series of images after the blurring process. Features and spatial information provide the role within the scale-space tree construction. After the process of blob extraction, users determine each type of the blob that was detected within the image by distinguishing classes to create ground truth image data. Within the same process, the KNN algorithm is applied to distinguish classes of the image's blobs in determining the performance.

Keywords: Feature stability, K-nearest neighbor, object detection: scale-space

Introduction

A human eye is one of the most important organ structures that the vision sense relies on. Eye diseases are a common problem occurring not only to senior citizens but infants as well. The retinopathy of prematurity (ROP) is a disease in which symptoms show abnormal blood vessels and scar tissue over the retina of the eye. Most ROP cases come from a lower gestational age and birth weight and are increasing at a considerable rate. In addition, most patients recognize this disease only when their retinal insights are degrading, resulting in a blurry vision for which treatment is complicated and nearly helpless.

Programs of pattern recognition and classifications have been one of the crucial subjects in computer vision analysis. Successful implementations would provide beneficial outcomes toward the field of the study's reliability and convenience for the user. The purpose of this task is to apply a segmentation of objects in locating the object of interest within the retinal image. Consecutive tasks such as object recognition, classification, and determination are used after the object of interest has been segmented. Performance of the output in object recognition and classification usually depends on the quality and accuracy

Department of Information, Computer and Communication Technology, Sirindhorn International Institute of Technology (SIIT), Thammasat University, 131 Moo 5, Tiwanont Road, Bangkadi, Muang, Pathumthani 12000, Thailand. Tel.: 0-2501-3505 Ext. 2005; Fax.: 0-2501-3505; E-mail: krit.inthajak@gmail.com

* *Corresponding author*

of object distinguishing after segmentation is applied to the image.

Two algorithms were used in the image segmentation; region-based (Wang, 2007) and edge-based segmentations (Forbes and Draper, 2000; Bowyer *et al.*, 2001; Konishi *et al.*, 2003). Regions of interest such as color, size, or shape are required for prior knowledge in order to provide formidable results. The following features will then be used to specify other parameters in order to improve the image's segmentation such as intensity thresholds, entropy, etc.

Proposed by Witkin (1983), a scale-space technique is a framework on a multi-scale representation which has been developed for computer vision in order to handle image structures at different scales. Applications for detecting image features such as blobs, edges, ridges, and corners are widely used along with scale-space techniques with a representation based on the lifetime of each application (Lindeberg, 1994, 1998a, 1998b). Jalba *et al.* (2006) presented a method in multiscaling for shape recognition which is based on 2 morphological scale-space representations and hat-transforms.Carvalho *et al.* (2003) proposed the method to segment yeast cells based on watershed and scale-space analysis.

The K-nearest neighbor algorithm (KNN) is a simple yet efficient method of classifying objects based on the closest training sample within the feature space (Tekmono, 2006; Thomas, 2008). The simplicity and simultaneous processing within the KNN algorithm is a considerable advantage. Plaku and Kavrati (2007) proposed a distributed computation of the KNN graph for large dimensional point sets. Performance in object detection is evaluated by demonstrating if blobs can only be linked when their features are stable over different scales. The evolution of linked blobs over different scales using feature stability shows how stable structures are in scale-space. Classes are analyzed and distinguished by the process of the KNN algorithm and accuracy is determined to evaluate the algorithm's efficiency (Su, 2011). Such processes could lead to promising and reliable image segmentation and processing procedures.

Demonstrations of feature stability and the KNN algorithm process is introduced in this research. Scale-space and feature stability provide an automatic detection of drusen and are evaluated by the KNN's performance, comparing the results with ground truth data to determine the accuracy.

Proposed Methods

Scale-Space Tree

Within the structures of image classification, the scale of a particular object as well as the intensity may be important in evaluating image processes. Scale-space theory is a multi-scale representation in a set of output images constructed by the convolution of an original image and a Gaussian filter with different values based on a scale parameter σ. The 2-dimension image, $f(x, y)$, defined by a convolution along with the Gaussian kernel $g(x, y, \sigma)$ is a set of output images in various scales generated by a successive smoothing process. In addition, parameter σ moderately changes in order to create a series of smooth images. Details of the images were suppressed but noticeable structures are still in existence during the blurring process.

The purpose of scale-space construction is to analyze the characteristics of the image's structure from different aspects. Light blobs are created due to the scale parameter results during the process of suppressing the image structures. As a scale-space tree constructs to its limit, we can determine the areas of the object of interest depending on the blob's lifetime. After the tree is formed, each blob is represented as a node and will be evaluated for its significance. Assuming that our object of interest is clearly visible towards our object detection, the object of interest should stay longer over different scales.

Feature Stability

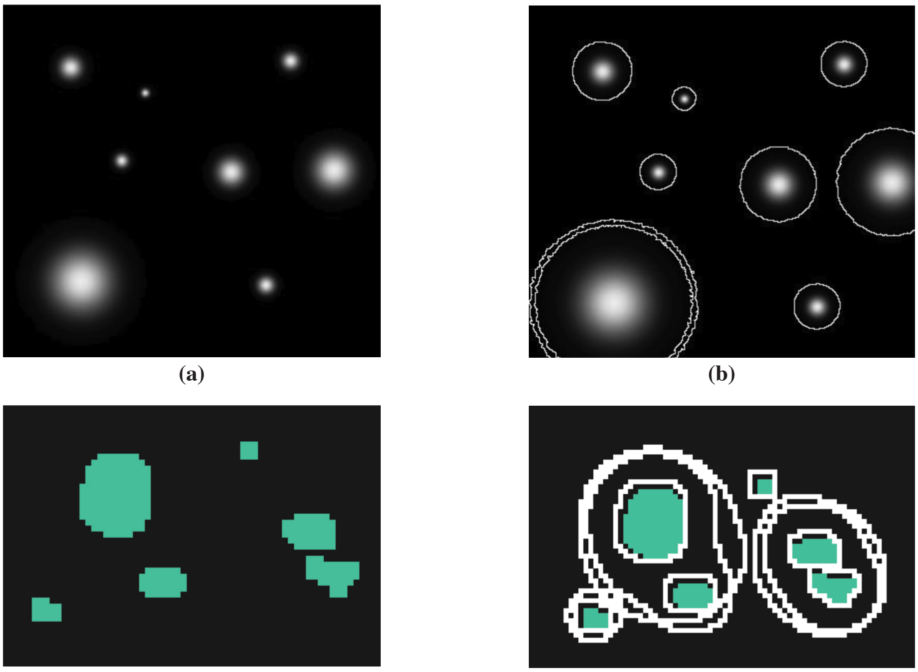
Feature stability provides a conventional blob linking process based on additional information such as color or texture. Blobs from adjacent scales that have spatial intersection will be considered as a pair candidate. Feature stability can be characterized by the following Equation:

$$S = \frac{1}{dist\,(f,\, g)} \tag{1}$$

where $f = [f_1, f_2, f_3..., f_n]$ and $g = [g_1, g_2, g_3,... g_n]$ and are the feature vectors corresponding to 2 candidate blobs simplified with the Euclidean distance, *dist*. Additionally, the significance of the blob features may vary depending on the blob's lifetime.

Scale-space Blob Linking and Blob Life Time

The normal blob linking process is used to form a scale-space tree using the information of the blobs in consecutive scales. The key concept is to decrease the limitations of parameter tuning by analyzing every blob within each scale. As a result, the scale-space blob does not require any pre-tuned parameters for finding blobs. (Duanggate *et al*., 2011) To obtain the blobs from each scale, the blob seeds are initialized at every local maximum. The region around the seed then grows until it meets a local minimum (Lindeberg, 1994). For example, 2 different blobs with different colors will still have the chance to be merged if they stay close together spatially. The blob lifetime is defined as a discrete value measured from the number of scales at which the structures exist. As the lifetime increases, the blob gains more significance. The lifetime can be defined by the time at which a certain blob at a specific scale appears or disappears. Focusing on 1 group object of interest, such as the green blobs in Figure 1, the white layers of circles present the intermediate results of the scale-space construction and it is assumed that the important structures in the image



(a) **(b)**

(a) **(b)**

Figure 1. Blob classification on the feature of marked ground truth data based on a retinal image where (a) is the original test image while (b) is the result based on blob detection indicated in the white circles

should stay longer over different scales.

K – Nearest Neighbor Algorithm (KNN)

After the process of blob extraction, 2 classifications were processed: by hand and by the KNN algorithm. Users will identify each blob's significance manually to create ground truth data. Once all the blobs have been identified, a training set is created for each class to distinguish each blob's level of significance. Once the ground truth image and the training set is complete, a KNN algorithm will be processed based on the training set's data and the value of *k*.

Given an image and features to be classified, the algorithm searches for the *k* – nearest neighbors among the training data based on similar measures. Training examples are vectors in a multidimensional feature space, each with a class label. Neighbors are taken from the training data to determine the classes of each type. Within the classification phase, *k* is a user-defined constant and a query which is classified by assigning the data which is most frequent among the *k* training samples nearest to the query point.

The data of the KNN algorithm consist of attributes X_i and the output Y. X_i is the proximity of neighboring input observations in the training data while their corresponding output values Y are used to predict the output value of the classes. To demonstrate the algorithm's procedure, assume that the query (q) distance has a value of ($X_1{}^q$, $X_2{}^q$) and a training sample (t) value of ($X_1{}^t$, $X_2{}^t$). The output can be determined by using the Euclidean distance which can be defined by:

$$d^2_{tq} = (X_1^q - X_1^t)^2 (X_2^q - X_2^t) \tag{2}$$

Training sample parameters in the KNN algorithm can also be extended for further development. By assigning more than 2 features, the equation can be calculated as follows:

$$d^2_{qt} = (X_1^q - X_1^t)^2 + (X_2^q - X_2^t)^2 + \ldots + (X_i^q - X_i^t)^2 \tag{3}$$

Given that the KNN method is dependent on distance measurements, the input data has to be standardized before proceeding to the KNN process. Figure 2 illustrates a diagram of a KNN model with 2 classes based on a training set; blue and red dots and a target which is a green dot. The inner dashed circle has a value of K = 1 which is the final result of the target to be considered as the blue class since 2 blue classes existed within the area. The outer dashed circle has a value of K = 3 which results as a red class (6 versus 4).

Experimental Results

Three general features were used as a blob descriptor vector; the blob's entropy, the average value of the Grey (G) channel, and the standard deviation of the G channel. Texture, color, and color distribution represents the 3 values for the blob, respectively. All features are normalized to values ranging from 0 to 1. Other specific features were implemented for particular applications such as compactness. Each blob generates a unique ID at a specific scale alongside the data of the blob's entropy, average G channel, and standard deviation of

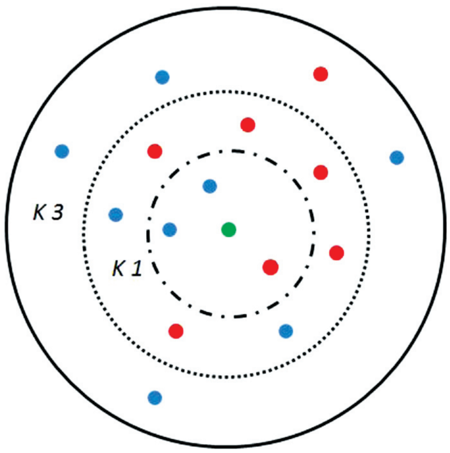


Figure 2. **A representation of a KNN algorithm. The green dot represents the target whereas the red and blue dots represent 2 different classes based on the training set. 2 dashed circles represent the value of *K* in which the inner circle is equal to 1 and the outer dashed circle is 3**

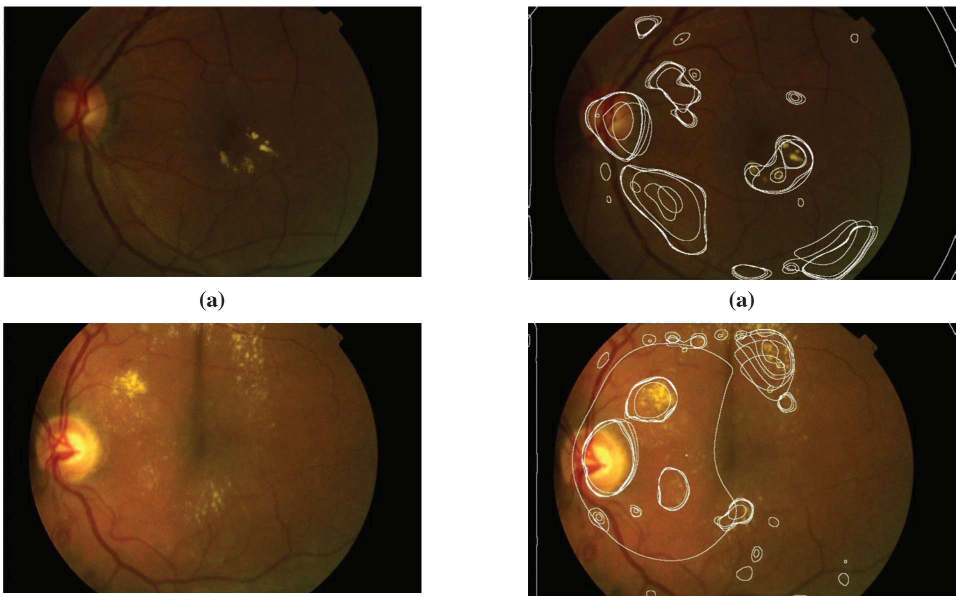
the G channel for further verification in the ground truth data.

For the seeding process of the blob, 3 criteria were used within the blob classification. Regions can be classified as a background or as blobs with the 3 following criteria: A) it is a local maximum point and will be used as a seed of the blob if the region has no neighbours then, or B) if the region is next to other background regions with higher grey levels, then the current region should be assigned as a background as well, or C) if the region has many neighbours within a higher grey level and those neighbouring regions are not parts of the same blob, then the current region must be assigned as a background.

All digital retinal images were obtained from a Kowa-7 non-mydriatic retinal camera (Kowa Co. Ltd., Nagoya, Japan) with a 45^{o} field of view taken at Thammasat University hospital. The images were stored in JPEG image format les with the lowest compression rates. The image size used was 752 × 500 pixels with 24 bit colour. Ten retinal images were used to evaluate the algorithm of which several images were demonstrated in Figures 3 and 4. Many drusen cases in the retinal images were taken into account such as retinas with bright exudates shown in Figure 3 and translucent exudates shown in Figure 4. Blobs were created from the process of feature stability, providing the object of interest. The maximum number of scale-space constructed blobs was set at 50 and 80 in order to determine whether the accuracy of the KNN algorithm is likely to increase or decrease its efficiency.

Each picture's blobs will be analyzed by hand and be indicated to be from the 3 following classes; optic disc (1), drusen exudate (2), and non-related class (3). During this process, we assume that blobs that cover more than 50% of the exduates are considered as class 2. Features of the blob (entropy, average value, and the standard deviation of G channel) will be used as the main references between the KNN implementation processes.

Once all the blobs have been identified, 3 blobs will be chosen as a training set for each class based on the previously mentioned general features in order to initiate the KNN

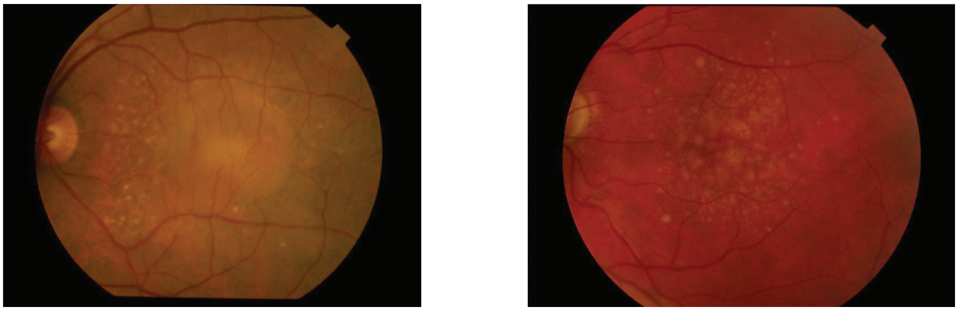


(a) (a)

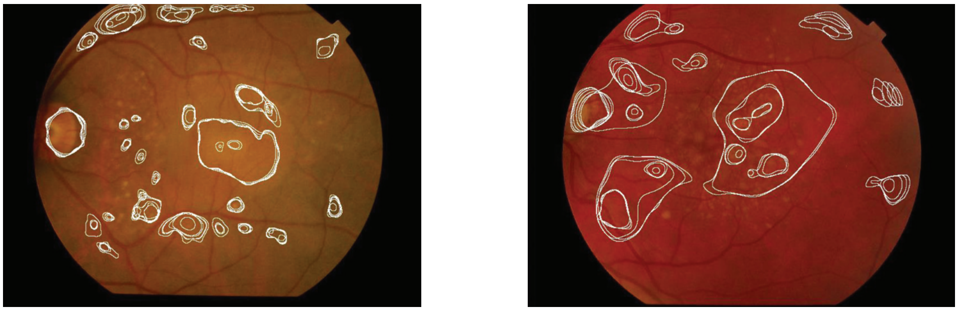
(b) (b)

Figure 3. An example of retinal images including bright exudates shown in (a) Feature stability is processed, creating blobs within the image and general features are recorded, as shown in (b)

program using the value of $K = 1$. The KNN will evaluate the significance of the blob's entropy, average G channel, and standard deviation of the G channel in respect to the selected ground truth's class. Once the process is completed, results of a ground truth image and KNN image will be evaluated to determine the accuracy. The percentage of the accuracy



(a) (b)

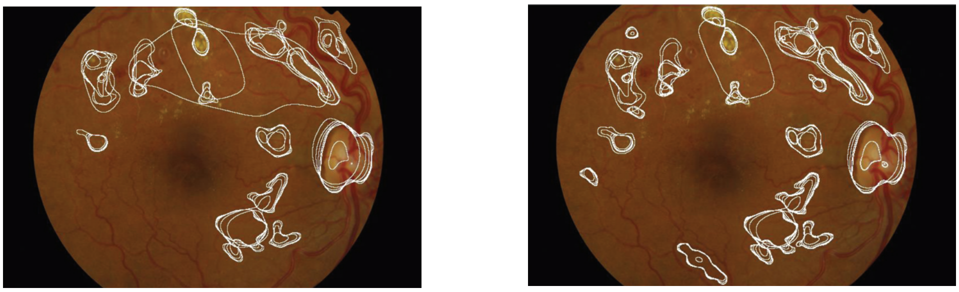


(a) (b)

Figure 4. An example of retinal images including translucent exudates shown in (a) Feature stability is processed, creating blobs within the image and general features are recorded, as shown in (b)

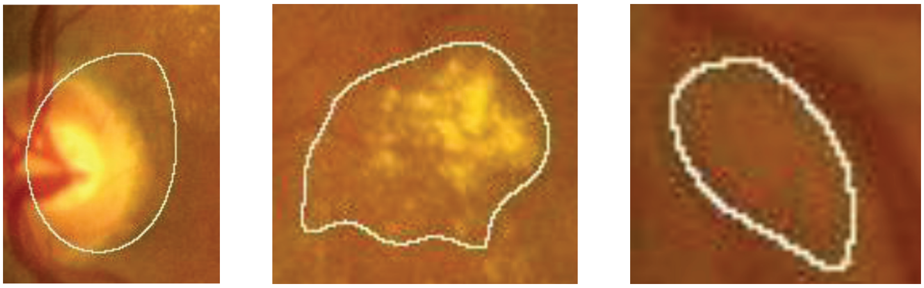
Table 1. Detection rate at maximum blobs = 50

Image No.	Total number of blobs	Ground truth result			KNN result			Accuracy
		Class						
		1	2	3	1	2	3	
#1	40	4	8	28	7	7	26	87.50%
#2	35	0	16	19	0	22	13	65.71%
#3	40	5	1	34	3	3	27	65.71%
#4	43	2	10	31	6	20	27	65.71%
#5	41	5	19	17	12	21	8	57.14%
#6	42	5	25	12	5	27	10	88.57%
#7	25	6	4	15	5	5	15	95.83%
#8	39	10	4	25	10	13	16	77.14%
#9	26	6	7	13	5	8	13	80.00%
#10	42	4	7	31	6	2	34	80.95%



(a) (b)

Figure 5. A demonstration of the maximum number of blobs indicated where (a) is set at 50 and (b) is set at



(a) (b) (c)

Figure 6. Images providing an example of an individual blob being assigned to each of the 3 classes in order to create ground truth data where (a) is an optic disc (1), (b) is a drusen exudate (2), and (c) is a non-related class (3)

Table 2. Detection rate at maximum blobs = 80

Image No.	Total number of blobs	Ground truth result			KNN result			Accuracy
		Class						
		1	2	3	1	2	3	
#1	55	9	10	36	8	17	30	74.28%
#2	69	3	19	47	13	19	47	71.01%
#3	68	6	0	32	6	0	32	100.00%
#4	74	3	16	55	3	27	41	76.81%
#5	53	7	25	21	7	33	13	84.90%
#6	71	5	25	41	6	19	46	84.90%
#7	49	8	5	36	6	6	37	93.88%
#8	46	12	4	30	12	6	28	91.30%
#9	55	8	13	34	7	15	33	84.76%
#10	64	6	16	42	6	8	50	82.61%

is computed by the overall correct class evaluated by the KNN compared with the ground truth data.

Most images resulted in an overall above average accuracy in detecting each class for each blob. In addition, most images with the parameter of a larger number of blobs tend to increase the performance of the KNN's accuracy. However, class 2 results are currently in a fluctuated state as shown in Tables 3 and 4.

Table 1 and Table 2 illustrate the overall results of the detection rates from all 10 images. Table 1 provides the results at the maximum number of blobs set to 50 and Table 2 at 80. The purpose of these Tables is to determine whether a larger number of blobs will improve the detection rate performance and the KNN classifications. Images No.1 and No.4 are based from the retinal image provided in Figure 3 and Images No.2 and No.5 are based from the retinal image provided in Figure 4. The accuracy of both Tables is determined by the KNN on how many results of each blob based on the KNN matches with the ground truth data divided by the total number of blobs.

Discussion

In summary, an implementation of blob detection with feature stability and the use of the KNN classification have been proposed in this research. The object of interest can be detected effectively due to the robustness of the algorithm for detecting variable size and variable shape objects. Gaussian filters provide the means of extracting the feature vectors of each blob by convoluting the original image. The blobs, taking account of their feature stability, construct a scale-space tree. While traversing on the tree, the lifetime of each blob can be calculated to notify its significance. All blobs are contained and extracted to analyze their significance in order to create a ground truth image. The KNN classification is processed and compared with the ground truth data for the algorithm's effectiveness. Feature vectors can be extended or modified to suit different applications and images. Blobs within a specified lifetime can be considered chosen as an object of interest, rather than the blob with the longest lifetime which does not indicate the object of interest.

The proposed algorithm could lead to an ideal outcome towards various applications that may require precision such as in medical applications for diagnosing symptoms and pinpointing interests. The processes of pinpointing the feature may still require human analysis in order to fully enhance

Table 3. Detection rate on class 2 (Drusen) at maximum blobs = 50

Image No.	Ground truth result	KNN result	Accuracy
#1	8	7	87.50%
#2	16	22	72.72%
#3	1	3	33.33%
#4	10	20	50.00%
#5	19	21	90.48%
#6	25	27	92.59%
#7	4	5	80.00%
#8	4	13	30.77%
#9	7	8	87.50%
#10	7	2	28.57%

the results of the output, along with the improvements to the algorithm. However, blob indication is recommended for enhancement in order to prevent an excess amount of blob identification, especially when executed on a high number of blobs. Developments in classification accuracy may lead to major benefits in computer analysis such as less human analysis and better performance such as the enhancement in integrating additional classification algorithms like Bayesian classifiers.

Although the current drusen detection rate is at a satisfactory level, this process consumes a certain amount of time. Blob indication requires a hand-drawn analysis over each blob in order to evaluate the blob's class by using the KNN classification. Certain retinal images that provide distinctive areas of optic disc and bright drusen exudates result in high accuracy towards the KNN classification process such as the results in Image No.7. In contrast, images that provide translucent or relatively small drusen exudates will likely result in a lower accuracy such as Image No. 8. However, several images provide a higher accuracy when the number of blobs is set to 80. In order to provide a better detection result for translucent drusen exudates at this current time, increasing the limit of the number of blobs to about 100 is the optimal choice while bright drusen exudate retinal images could be set at a lower number of blobs. However, the user is also required to indicate all those blob identifications before implementing the classification process. As a result, blob indication becomes one of the most time consuming tasks within this process. Class 2 results based on Tables 3 and 4 are still in a highly varied state due to the low number of drusen exudates that existed within several images and the existence of translucent drusen exudates which results in blobs being unable to be detected. In addition, human errors are also an issue in creating ground truth data since some blobs provide a dilemma as to which class should be selected since the blob may not cover the areas entirely or may exceed the areas.

Conclusions

Studies of the implementation of retinal detection based on scale-space theory, feature stability, and an evaluation of the KNN were presented in this paper. In summary, an automatic optic disc, ROP, and drusen detection are investigated for further development. Many experiments have provided promising results within the rate of drusen and optic disc

Table 4. Detection rate on class 2 (Drusen) at maximum blobs = 80

Image No.	Ground truth result	KNN result	Accuracy
#1	10	17	58.82%
#2	19	19	100%
#3	0	0	100%
#4	16	27	59.26%
#5	25	33	75.76%
#6	25	19	76.00%
#7	5	6	83.33%
#8	4	6	66.67%
#9	13	15	86.67%
#10	16	8	50.00%

detection. The procedures of detection include scale-space implementation and blob detection methods. All ground truth retinal images are analyzed by hand. Data analysis and results are presented using the implementation of the KNN classification. Both results are compared to evaluate the performance. Each retinal image is assigned with a specific identification of a blob and is analyzed on its significance accordingly. Overall, the automatic detection of drusen within the patient's retina could provide a major benefit toward the ophthalmologist's role in patient's treatment. Accurate classifications based on the KNN could also help in identifying the abnormal areas of a patient's eye. In addition, further developments in classification techniques could also be implemented in the near future for higher accuracy.

There are possibilities of expanding the process of scale-space theory, feature stability, and classification procedure and of being able to optimize the program for a better result regarding precision and accuracy. Since the process requires a certain amount of time to implement each retinal image, the first and foremost priority is to find the best solution in optimizing the time for all the procedures. Tweaks in the algorithm of scale-space and the feature stability procedure need to be looked at. Providing a solution with complex retinal images (for example, drusen that have a similar shape and intensity with the optic disc) is another challenge that must be undertaken.

References

Bowyer, K., Kranenburg, C., and Dougherty, S. (2001). Edge detector evaluation using empirical roc curves. Comput. Vis. Image Und., 84:77-103.

Carvalho, M.A.G., Lotufo, R.A., and Couprie, M. (2003). Segmentation of images of yeast cells by scale-space analysis. XVI Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'03); 12-15 Oct, 2003; Sao Carlos, Brazil; p. 376-380.

Duanggate, C., Uyyanonvara, B., Makhanov, S.S., Barman, S., and Williamson, T. (2011). Parameter-free optic disc detection. Comput. Med. Imag. Grap., 35(1):51-63.

Forbes, L.A. and Draper, B.A. (2000). Inconsistencies in edge detector evaluation. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition; 13-15 June, 2000; Hilton Head Island, SC, USA; 2:398-404.

Inthajak, K., Duanggate, C., Uyyanonvara, B., Makhanov, S.S., Barman, S., and Williamson, T. (2010). Variable size blob detection with feature stability. Proceedings of the 3rd Biomedical Engineering International Conference on Medical Imaging, Image and Signal Processing; 27-28 Aug, 2010; Kyoto, Japan; p. 133-137.

Jalba, A.C., Wilkinson, M.H.F., and Roerdink, J.B.T.M. (2006). Shape representation and recognition through morphological curvature scale spaces. IEEE T. Image Process., 15:331-341.

Konishi, S., Yuille, A., and Coughlan, J. (2003). Statistical edge detection: Learning and evaluating edge cues. IEEE T. Pattern Anal., 25:57-74.

Lindeberg, T. (1994). Scale-Space Theory in Computer Vision. Kluwer Int Ser. Eng. C. Springer-Verlag, Berlin, Germany, 423p.

Lindeberg, T. (1998a). Edge detection and ridge detection with automatic scale selection. Int. J. Comput. Vision, 30:117-156.

Lindeberg, T. (1998b). Feature detection with automatic scale selection. Int. J. Comput. Vision, 30: 79-116.

Plaku, E. and Kavrati, L.E. (2007). Distributed computation of the knn graph for large high-dimensional point sets. J. Parallel Distr. Com., 67(3): 346-359.

Su, M-Y. (2011). Using clustering to improve the KNN-based classifiers for online anomaly network traffic identification. J. Netw. Comput. Appl., 34(2):722-730.

Tekmono, K. (2006). K-Nearest Neighbors Tutorial. http://people.revoledu.com\kardi\tutorial\KNN\, Accessed date: June 8, 2010.

Thomas, B.F. (2008). K-Nearest Neighbors Algorithm: Prediction and Classification. http://faculty.smu. edu/tfomby/eco5385/lecture/K-NN%20 Method.pdf, Accessed date: April 14, 2012.

Wang, S., Ma, X., Zhang, X., and Jiao, L. (2007). Watershed-based textural image segmentation. Proceedings of International Symposium on Intelligent Signal Processing and Communication Systems; 28 Nov-1 Dec, 2007; Xiamen, China; p. 312-315.

Witkin, A.P. (1983). Scale-space filtering. Proceedings of the 8th International Joint Conference on Artificial Intelligence; Aug 8-12, 1983; Kalsruhe, Germany; p. 1019-1022.