

AUTOMATIC DESIGN OF APERTURE FILTERS USING NEURAL NETWORKS APPLIED TO OCULAR IMAGE SEGMENTATION

Marco E. Benalcázar^{1,2,3}, Marcel Brun³, and Virginia L. Ballarín³

¹ Secretaría Nacional de Educación Superior, Ciencia Tecnología e Innovación (SENESCYT) – Ecuador

² Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET) – Argentina

³ Grupo de Procesamiento Digital de Imágenes, Facultad de Ingeniería, Universidad Nacional de Mar del Plata – Argentina
Email: marco_benalcazar@hotmail.com, marcelbrun@gmail.com, vballari@gmail.com

ABSTRACT

Aperture filters are image operators which combine mathematical morphology and pattern recognition theory to design windowed classifiers. Previous works propose designing and representing such operators using large decision tables and classic linear pattern classifiers. These approaches demand an enormous computational cost in order to solve real image problems. The current work presents a new method to automatically design Aperture filters for color and grayscale image processing. This approach consists of designing a family of Aperture filters using artificial feed-forward neural networks. The resulting Aperture filters are combined into a single one using an ensemble method. The performance of the proposed approach was evaluated by segmenting blood vessels in ocular images of the DRIVE database. The results show the suitability of this approach: It outperforms window operators designed using neural networks and logistic regression as well as Aperture filters designed using logistic regression and support vector machines.

Index Terms— Image processing, pattern recognition, mathematical morphology, neural networks, Aperture filters, ensemble of classifiers.

1. INTRODUCTION

The W-operators, or Window operators, are a class of non-linear operators that belong to computational mathematical morphology [1,2]. Most of the practical applications of W-operators deal with binary images, including image segmentation, texture identification, character recognition, and noise filtering [3]. In the case of grayscale image processing, such operators have been used for texture identification and deblurring in only few applications [1,2]. The main characteristic of W-operators is to assign an output value for a given pixel that depends only on the values observed within a given window neighborhood (i.e., domain constraint). The pixel being processed is usually the center pixel of the window employed [3]. The automatic design of W-operators consists of a statistical optimization using training examples. Each training example is composed of an

observed image (i.e., an image with the problem to solve) and an ideal image (i.e., image desired after processing) [1].

In practice, the statistical optimization process used to automatically design W-operators is based on the estimation of either joint probabilities [1] or conditional probabilities [3] for each pattern (i.e., observation) observed via a given window. Based on these probabilities, W-operators are designed using a learning algorithm and characterized by a function (i.e., classifier) that minimizes some given cost function or error measure. The complexity of this optimization process depends on the size of the window used and the number of channels and gray levels of both the observed and the ideal images. The size of the space of possible W-operators from which to choose the optimal operator is $m^{|W|}$, where m is the number of gray levels of the ideal image, l is the number of gray levels of the observed image, and $|W|$ is the size of the given window W . Therefore, increasing the size of W causes an exponential increase in the complexity of the optimization process. The size and shape of W are usually chosen *a priori* by the designer based on the characteristics of the problem to solve (e.g., size of the objects to segment and size and type of the noise artifacts to filter).

W-operators designed using large windows (i.e., windows bigger than 7x7) possess good discrimination capabilities because they can capture the shape, texture, and grayscale information better than those designed using smaller windows. However, they lead to poor generalization because the space of operators used (i.e., search space) to find the optimal operator is huge, causing overfitting. On the other hand, W-operators designed using small windows (e.g., 3x3 or 5x5) have poor discrimination capabilities. Therefore, the automatic design of W-operators involves a tradeoff between the bias caused by the size of the search space and the variance caused by the generalization error.

The automatic design of W-operators for binary image processing using small and medium-sized windows has low complexity. In such cases, the search space has $2^{|W|}$ hypothesis. This is perhaps the most important reason why the majority of practical applications of W-operators deal only with binary images. For grayscale image processing, the complexity of design of W-operators is a major issue because the search space increases significantly, even for

small windows and images with only few gray levels. For example, for binary segmentation (i.e., 2 classes) of images containing 256 gray levels, using W-operators designed based on a window W of size $|W|$, the search space for the optimal operator has $2^{256^{|W|}}$ hypothesis. To overcome this issue, Aperture filters were proposed [2]. Unlike W-operators which constrain only the domain of the images being processed, Aperture filters constrain both the domain using a window W and the range using a window K , thus reducing (i.e., constraining) the search space for the optimal operator. Therefore, Aperture filters are suboptimal in comparison with W-operators; however, they are expected to have better performance in a practical sample-size context because of their lower cost of estimation [2].

Previously, Aperture filters have been designed using various methods such as decision trees [2], multi-mask filters [4], and pyramidal multiresolution [5]. These 3 approaches were applied to filtering Gaussian and salt-and-pepper noise in one-dimensional synthetic signals [4]. Another application consisted of deblurring small synthetic images (256x256 pixels) with only few gray levels (16 levels) [5]. The characteristics of these problems require employing only small apertures, making the computational cost of these approaches low and potentially attractive for solving real problems. However, going from synthetic to real problems usually requires using larger apertures than those used in the previously mentioned works. Additionally, the images to process usually have 256 gray levels, and the databases are composed of many pairs of training images. In this scenario, the high computational cost required both to store large tables of patterns and to compute a large number of probabilities when implementing any of the previous approaches makes their application to real problems a daunting or even impossible task. To overcome this problem, we propose to design Aperture filters using artificial feed-forward neural networks (NN).

We model the conditional probability of a pixel as part of the object of interest by using the response of a three-layer (i.e., input, hidden, and output layers) NN [6]. For most practical applications, including the one presented in this work, the number of patterns that can be extracted by placing the aperture in all the pixels of all the available pairs of training images is enormous. Thus, the computational cost of training a single NN is very expensive, making the application of Aperture filters to real problems almost impossible. In this paper, we circumvent this issue by using an ensemble method [6]. Here, we train as many NNs as pairs of training images available. Then, we combine the NN responses in a weighted fashion by using the Shannon entropy to measure the uncertainty of each response [6,9].

To test the performance of the proposed approach in a real image processing problem, we designed and applied Aperture filters to segment blood vessels in fundus ocular images from the DRIVE public database [7]. We compared the results obtained by using the proposed approach with those obtained from applying (a) W-operators designed

using NN and logistic regression [8], (b) Aperture filters designed using logistic regression, and (c) support vector machines (SVM) with linear kernel [8]. We chose this problem because segmentation of ocular images yields important information about the morphological features of ocular blood vessels such as length, width, and tortuosity [7]. These features can be used for medical applications or biometric control of access. For medical applications, medical professionals and computer-aided methods use these features to diagnose and evaluate certain diseases, e.g., hypertension, atherosclerosis, and diabetic retinopathy. Some systems of biometric control of access base their work on the fact that the retinal vasculature morphology is unique in each person.

The rest of this paper is organized as follows. In Section 2, we review some concepts and definitions with respect to Aperture filters applied to image segmentation and describe the proposed approach. We present an example of application and discuss the results in Section 3, and in Section 4, we summarize the main contributions of this work and outline possible future developments.

2. METHODOLOGY

In this section, we present some theoretical definitions that provide background and describe the proposed approach.

2.1. Aperture filters for binary segmentation

Let the function $O: E \rightarrow L$ be a digital *grayscale image*, where E is a nonempty subset of \mathbb{Z}^2 and $L = \{0, 1, \dots, l-1\}$ is a grayscale set, with $l \in \mathbb{Z}^+$. Usually, for practical applications we consider $l = 256$. The set of all possible grayscale images from E to L is denoted by L^E . A *pixel* t of the grayscale image O is a coordinate in E of the form $t = (x, y)$ that takes an element from the set L . In a similar way, we define a *binary image* as a function $I: E \rightarrow \{0, 1\}$. The set of all possible binary images is denoted by $\{0, 1\}^E$. We also define a *multivariate image* as a function $Q: E \rightarrow (L_1 \times L_2 \times \dots \times L_m)$ such that $Q = (O_1, O_2, \dots, O_m)$, where $O_i \in L^E$ are the channels of Q . We denote the set of all multivariate images from E to $(L_1 \times L_2 \times \dots \times L_m)$ by $(L_1 \times L_2 \times \dots \times L_m)^E$. A pixel t of the multivariate image Q is a coordinate in E that takes the vector $(O_1(t), O_2(t), \dots, O_m(t))$. *Color images* are a particular case of multivariate images composed of $m = 3$ channels. For the RGB color space [9,10], the images O_1 , O_2 , and O_3 represent the red, green and blue channels, respectively. Color and grayscale images are particular cases of multivariate images having a single channel.

In this work, we transform the RGB color images into grayscale images. For this purpose, given an RGB image $Q = (O_R, O_G, O_B)$ and an arbitrary point $t \in Q$, the function $\Omega: (L_1 \times L_2 \times L_3)^E \rightarrow L^E$ is a transformation of RGB into grayscale images, such that

$$G(t) = \Omega(Q(t)) = \text{round}(\alpha_1 O_R(t) + \alpha_2 O_G(t) + \alpha_3 O_B(t)). \quad (1)$$

In (1), the weights α_i must satisfy $\alpha_1 + \alpha_2 + \alpha_3 = 1$, and $\text{round}(a)$ rounds a to nearest integer. Let us define a *spatial window* $W = \{w_1, w_2, \dots, w_n\}$, with $w_i \in \mathbb{Z}^2$, to be a finite small subset of \mathbb{Z}^2 centered at the origin of \mathbf{E} : $(0,0) \in W$. We denote the number of points (i.e., cardinality) of W by $n = |W|$. Let a *range window* $K = \{-k, -k+1, \dots, 0, \dots, k\}$ be a set that contains the input gray levels for filtering. K is composed of $2k+1$ values, with $k \in \mathbb{Z}^+$ and $k \ll l$. Finally, let an *aperture* $W \times K$ be the Cartesian product of W and K .

Given the grayscale image O and the aperture $W \times K$, an *aperture configuration*, also called *observation*, is a function $\mathbf{X}: W \rightarrow K$ so that $\mathbf{X} = (X_1, \dots, X_n)$, with $n = |W|$. We obtain aperture configurations from the image O by applying the following 4 steps: (a) applying the *spatial translation* of O by t : $O_{-t} = O(t'-t)$, where $t, t' \in \mathbf{E}$; (b) registering the pixel values of O_{-t} within the spatial window W , which returns the vector $\mathbf{u}_{O(t)} = (O(w_1 - t), \dots, O(w_n - t))$ (for simplicity of notation, we refer to vector $\mathbf{u}_{O(t)}$ with \mathbf{u} from now on); (c) applying the *range translation* of \mathbf{u} by a scalar $z \in \mathbb{Z}^+$: $\mathbf{u}_z = (O(w_1 - t) - z, \dots, O(w_n - t) - z)$; and (d) the projection of points of \mathbf{u}_z outside K to either $+k$ or $-k$. Formally and denoting the projection operation by $(\cdot)^*$, each point $X_i \in \mathbf{X}$, with $i = 1, 2, \dots, n$, is obtained from a given observation $\mathbf{u} = (u_1, \dots, u_n)$ as follows:

$$X_i = (u_i)^* = \min(\max(-k, u_i - z), k). \quad (2)$$

Given the definitions above, let an *Aperture filter* $\Psi: L^E \rightarrow \{0,1\}^E$ be a function that maps grayscale images into binary images [2]. Aperture filters are characterized by functions of the form $\psi: K^W \rightarrow \{0,1\}$, called *characteristic functions*, so that for each $t \in \mathbf{E}$:

$$I(t) = \Psi(O)(t) = \psi((O(w_1 - t) - z)^*, \dots, (O(w_n - t) - z)^*). \quad (3)$$

In (3), $z: L^W \rightarrow L$ is a function that allows us to place the aperture $W \times K$ vertically at the pixel $t \in O$. We use $z = \text{median}(u_1, \dots, u_n)$. In the domain of pattern recognition, the function ψ can be assumed to be a *classifier* that maps observations $\mathbf{X} \in \{\mathbf{X}_1, \dots, \mathbf{X}_N\}$, with $\mathbf{X}_i = (X_{i1}, \dots, X_{in})$, to labels from the set $\{0,1\}$ [6,8,11]. Based on this assumption, processing the image O using Aperture filters is reduced to a classification of its pixels within the aperture $W \times K$. Finally, Aperture filters fulfill 3 important properties: (a) *spatial translation invariance*: $\Psi(O_i) = \Psi(O)$, for each pixel $t \in O$; (b) *range translation invariance*: $\Psi(O(t) + z) = \Psi(O(t)) + z$ for any $z \in \mathbb{Z}^+$; (c) *local definition* in $W \times K$: $\Psi(O)(t) = \Psi(O \cap (W \times K))(t)$ [1-5].

The *segmentation* of the grayscale image O consists of obtaining a partition $\mathcal{P}_O = \{S_1, S_2, \dots, S_p\}$ of the set of pixels $S = \{t: t \in \mathbf{E}\}$ of O , such that the sets $S_i \in \mathcal{P}_O$ are connected, disjointed, and $S = \bigcup_{i=1}^p S_i$. Therefore, the pixels belonging to each set S_i are spatially connected and have similar properties or features in the image O . As a result of the segmen-

tation process, an image I is obtained containing p segmented objects or regions, including the background. In *binary segmentation*, we extract a single type of object (i.e., $p = 2$), and therefore, I is a binary image.

2.2. Statistical design of Aperture filters

For the statistical design of Aperture filters, we consider each pair of images (O, I) to be drawn from a *jointly random signal process* (\mathbf{O}, \mathbf{I}) . \mathbf{O} generates grayscale images $O \in L^E$ to be observed, and \mathbf{I} generates binary images $I \in \{0,1\}^E$ to be estimated. In this context, the pixels of O and I are considered discrete random variables. Therefore, the *statistical design* is to find an Aperture filter Ψ with characteristic function ψ using a learning algorithm and an error measure between $\Psi(O)$ and I [1-5]. Typically, the error measure to be minimized is the mean squared error (MSE) [6,8,11].

Given a pair of images (O, I) and a filter Ψ , the characteristic function ψ of Ψ is applied to the aperture observation \mathbf{X} obtained at each pixel t of O to form an estimator $\psi(\mathbf{X})$ of $Y = I(t)$. If we assume O and I are jointly stationary, then \mathbf{X} and Y are independent of the position t . Moreover, the jointly random process (\mathbf{O}, \mathbf{I}) induces a probability distribution $\Pr(\mathbf{X}, Y) = \Pr(Y|\mathbf{X})\Pr(\mathbf{X})$ on the vector space $(\mathcal{X}, \mathcal{Y})$, with $(\mathbf{X}, Y) \in (\mathcal{X}, \mathcal{Y})$, where \mathcal{X} is the space of all $|K|^{|W|}$ possible aperture configurations and $\mathcal{Y} = \{0,1\}$. $\Pr(Y|\mathbf{X})$ is the *a posteriori* class conditional distribution, and $\Pr(\mathbf{X})$ is the probability of the occurrence of the observation \mathbf{X} . Thus, in the statistical design, we want to find the filter, Ψ , that minimizes the MSE between $\Psi(O)$ and I defined as follows:

$$\text{MSE}(\Psi) = E[(\Psi(O) - I)^2]. \quad (4)$$

In (4), E denotes the expectation operator relative to the joint distribution $\Pr(\mathbf{X}, Y)$ of (O, I) . Taking into account that Ψ is characterized by $\psi \in \{0,1\}$, then (4) yields the following expression:

$$\text{MSE}(\Psi) = \Pr(\Psi(O)(t) \neq I(t)) = \text{MSE}(\psi) = \Pr(\psi(\mathbf{X}) \neq Y). \quad (5)$$

The probability $\Pr(\psi(\mathbf{X}) \neq Y) = \sum_{\{\mathbf{x}, \psi(\mathbf{x})=1\}} \Pr(Y=0|\mathbf{X})\Pr(\mathbf{X}) + \sum_{\{\mathbf{x}, \psi(\mathbf{x})=0\}} \Pr(Y=1|\mathbf{X})\Pr(\mathbf{X})$ measures the error rate of ψ . $\Pr(Y=1|\mathbf{X})$ and $\Pr(Y=0|\mathbf{X}) = 1 - \Pr(Y=1|\mathbf{X})$ are the *a posteriori* class conditional probabilities that \mathbf{X} takes 1 or 0, respectively. Therefore, the design of an Aperture filter can be reduced to only 2 steps: first estimating $\Pr(Y=1|\mathbf{X})$ and then finding the value of ψ for each possible observation \mathbf{X} by applying the following *decision rule*:

$$\psi = \begin{cases} 1 & \text{if } \Pr(Y=1|\mathbf{X}) \geq \tau \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

If we set the threshold $\tau = 0.5$ in (6), then ψ is the optimal function that minimizes (5). However, for some practical applications, we can select the value of the threshold τ based on a tradeoff between the false positive rate (FPR) and the false negative rate (FNR) of ψ .

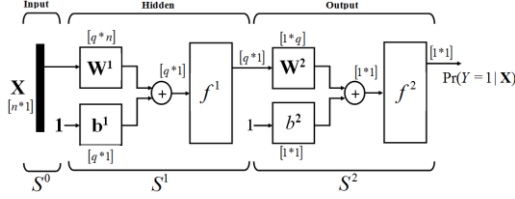


Fig. 1. Three-layer feed-forward NN architecture.

2.3. Automatic design of Aperture filters using NN

Here, we model the distribution $\Pr(Y=1|\mathbf{X})$ by using the response of a three-layer (input S^1 , hidden S^2 and output S^3) feed-forward NN composed of q neurons in S^1 as shown in Figure 1 [3,6]. We chose NNs because of their ability to implement complex decision boundaries. The network architecture we used was selected based on the fact that a feed-forward NN composed of 3 layers with a sufficient number of neurons in S^1 is capable of approximating any Borel measurable function with the desired degree of accuracy [3,6]. Sigmoid transfer functions $f: \mathbb{R} \rightarrow [0,1]$ are used in all the neurons of S^1 and S^2 : $f^1(a) = f^2(a) = 1/(1+\exp(-a))$.

To fit the set $\beta = \{\mathbf{W}^1, \mathbf{W}^2, \mathbf{b}^1, \mathbf{b}^2\}$ of weights and biases of the NN, we maximize the likelihood $\Pr(\mathcal{D}|\beta)$ of the set $\mathcal{D} = \{(\mathbf{X}_1, Y_1, \text{freq}(\mathbf{X}_1, Y_1)), \dots, (\mathbf{X}_N, Y_N, \text{freq}(\mathbf{X}_N, Y_N))\}$ of N training patterns given the parameters β of the NN. The number of times that \mathbf{X}_i has the value Y_i in the training set of images is denoted by $\text{freq}(\mathbf{X}_i, Y_i)$. $\Pr(\mathcal{D}|\beta)$ is computed as follows:

$$\Pr(\mathcal{D}|\beta) = \prod_{i=1}^N \prod_{j=0}^1 \Pr(Y=j|\mathbf{X}_i)^{\text{freq}(\mathbf{X}_i, Y=j)}. \quad (7)$$

The maximization of (7) is equivalent to finding the set β that minimizes $-\ln(\Pr(\mathcal{D}|\beta))$. Since finding an exact solution for this problem is a very difficult task, we use a gradient-descent method to find a local minimum instead [6].

2.3. Ensembles of Aperture filters

The ensemble method tackles the practical problem of dealing with a huge and single training set \mathcal{D} of patterns to design a unique filter using all the pairs of training images available. Storing a single large training table of patterns demands a huge amount of memory, and NN training takes a long time. To overcome these issues, we design a filter for each training pair of images we have available. Then, we combine the predictions of each filter designed $\Psi \in \{\Psi_1, \dots, \Psi_M\}$ by employing the following expression:

$$\Pr(Y=1|\mathbf{X})_{\text{Ensemble}} = \sum_{i=1}^M P e_i \Pr(Y=1|\mathbf{X})_i. \quad (8)$$

In (8), $\Pr(Y=1|\mathbf{X})_i$ is the probability predicted by the filter Ψ_i . The sum of all weights $P e_i$ is 1. The weight $P e_i$ controls the influence of Ψ_i in the final prediction of the ensemble and is computed by using the following equation:

$$P e_i(\mathbf{X}) = (1 - H(\psi(\mathbf{X})_i)) / (M - \sum_{j=1}^M H(\psi(\mathbf{X})_j)). \quad (9)$$

In (9), $H(\psi(\mathbf{X})_i) = -\sum_{j \in \{0,1\}} \Pr(Y=j|\mathbf{X})_i \log_2(\Pr(Y=j|\mathbf{X})_i)$ is the Shannon entropy [6,9] of the prediction of Ψ_i regarding the pattern \mathbf{X} . If the probability value $\Pr(Y=1|\mathbf{X})_i$ is very close to 0.5, then the uncertainty of Ψ_i to assign a label to \mathbf{X} is high; and therefore, its degree of influence on the ensemble prediction $\Pr(Y=1|\mathbf{X})_{\text{Ensemble}}$ is low.

3. RESULTS AND DISCUSSION

We evaluated the performance of the proposed approach by segmenting fundus ocular images. This segmentation task consists of partitioning the set of pixels of an ocular image into 2 disjoint subsets. One subset contains the thick and thin blood vessels of the retinal vascular tree. The other subset is formed by the pixels belonging to both the optical disc and the retinal contour, and in some cases, the pixels belonging to certain ocular pathologies such as hard and soft exudates and hemorrhages.

The images used in this work come from the DRIVE public database [7]. This database consists of 2 groups of images, one for training and the other for testing. Each group contains 20 pairs of images. Each pair consists of one color image and its respective binary image. Binary images contain the blood vessels manually segmented by experts. Each channel of the color images contains 256 gray levels. Both color and binary images have a size of 565x584 pixels. In this database, 7 images have pathologies (i.e., exudates and hemorrhages). Four of these images are in the training group, and the other 3 are in the testing group.

We transformed the color images into grayscale images by employing (1) with the weights $\alpha_1 = 0.07$, $\alpha_2 = 0.9$, and $\alpha_3 = 0.03$. We chose these weights because the green channel in ocular images has the highest contrast between the blood vessels and the background; whereas, the red and blue channels have lower contrast and some level of noise [7,9].

Then we designed an ensemble of Aperture filters using the proposed approach. The filters that composed the ensemble were designed using the patterns extracted from each of the 20 pairs of training images by using the aperture $W \times K$. W is a 16x16 spatial window, and $K = \{-10, -9, \dots, 10\}$ is the range window employed. This aperture allowed us to capture the structure of the blood vessels well enough for the segmentation task. Each NN used to model $\Pr(Y=1|\mathbf{X})$ was composed of 30 neurons in S^1 . This number of neurons was obtained by applying model selection with crossvalidation in the training set of images.

Next, we applied the ensemble designed in the previous step to each of the 20 images in the testing set. The performance of the ensemble was evaluated by computing the receiver operating characteristic (ROC) curve and the value of the area under the curve (AUC) [11] shown in Figure 2. The ROC curve is based on the true positive rate $\text{TPR} = \text{TP}/(\text{TP} + \text{FN})$, and the false positive rate $\text{FPR} = \text{FP}/(\text{FP} + \text{TN})$ [10]. For each image segmented, TP is the number of pixels properly classified as blood vessels; FN is

the number of pixels incorrectly classified as background; FP is the number of pixels incorrectly classified as blood vessels; and TN is the number of pixels properly classified as background. Each pair (TPR,FPR) of the ROC curves was computed by thresholding $\Pr(Y=1|\mathbf{X})_{Ensemble}$ at each value of $\tau \in \{0,0.05,\dots,1\}$. The TPR and FPR used to plot the ROC curve are the averages of the 20 testing images.

The ROC curves and AUC values presented in Figure 2 show that the proposed approach outperforms W-operators designed using NN and logistic regression. This could be because the total cost of estimating optimal Aperture filters using NN and combining the predictions in the ensemble is lower than the cost of designing ensembles of W-operators.

The results show that the proposed approach also outperforms ensembles of Aperture filters designed using logistic regression and SVM with linear kernel. This could be because NNs are able to implement more complex decision boundaries than hyperplanes implemented by logistic regression and SVM [6,11]. We used linear kernels for the case of SVMs because of the high computational cost of finding the support vectors using SVMs with more sophisticated kernels (e.g., radial basis kernels).

Finally, the examples of images predicted by the ensemble of Aperture filters (using a threshold $\tau = 0.5$) shown in Figure 2 indicate that this method is able to identify both thick and thin blood vessels. We can also see that the ensemble of Aperture filters is robust to changes in color and brightness inside the image as well as between images.

4. CONCLUDING REMARKS

We have presented a new method of designing Aperture filters for color and grayscale image processing. This new approach uses feed-forward NNs to design and represent the characteristic functions of Aperture filters. We chose NNs because of their ability to implement complex decision boundaries for classification. To address the high computational cost of training a single NN using a large table of patterns, we proposed designing a family of Aperture filters using the tables of patterns extracted from each pair of training images available and then combining their predictions in a weighted fashion to form an ensemble of filters.

To test the performance of the proposed approach, we segmented blood vessels in ocular images from the DRIVE database. The results show that our method outperforms ensembles of W-operators designed based on NNs and logistic regression. We also compared the results obtained in this work with those obtained by applying an ensemble of linear classifiers based on logistic regression and SVM.

Further work includes testing the proposed method to solve additional image processing problems involving the segmentation of objects. In the future, this procedure could be expanded to include direct processing of color images, which has the potential of improving its performance due to the additional information contained in color images.

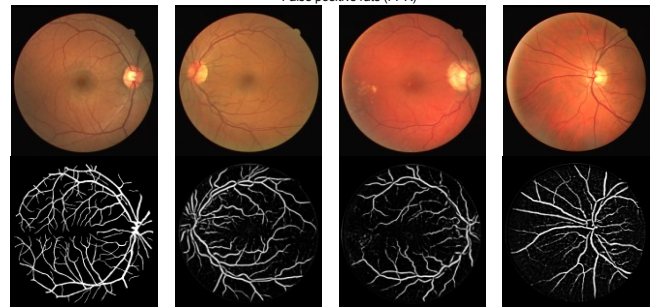
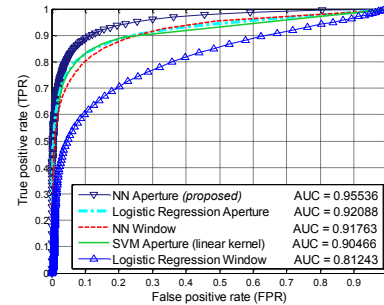


Fig. 2. Results of applying the proposed approach. In the top row, the ROC curves and AUC values are presented. The 2nd row shows the original color images, and the 3rd row shows binary images. The 1st column shows an example of a training pair of images (observed and ideal); whereas, the 2nd, 3rd, and 4th columns show the results of the segmentation of the images from the 2nd row.

5. REFERENCES

- [1] J. Barrera, G. Banon, and E.R. Dougherty, "Automatic design of morphological operators," in *Space Structure and Randomness*, 2005, pp. 257-278.
- [2] R. Hirata, E.R. Dougherty, and J. Barrera, "Aperture Filters," *Signal Processing*, vol. 80, no. 4, pp. 697-721, April 2000.
- [3] M.E. Benalcázar, M. Brun, and V.L. Ballarin, "Artificial neural networks applied to statistical design of window operators," *Pattern Recognition Letters*, vol. 34, no. 9, pp. 970-979, July 2013.
- [4] A.C. Green, S. Marshall, D. Greenhalgh, and E.R. Dougherty, "Design of multi-mask aperture filters," *Signal Processing*, vol. 83, no. 9, pp. 1961-1971, September 2003.
- [5] R. Hirata Jr., M. Brun, J. Barrera, and E.R. Dougherty, "Multiresolution Design of Aperture Operators," *Mathematical Imaging and Vision*, vol. 16, no. 3, pp 199-222, May 2002.
- [6] C.M. Bishop, *Neural Networks for Pattern Recognition*, Oxford, 2005.
- [7] J. Staal, M. Abramoff, M. Niemeijer, M. Viergever, B. Van Ginneken, "Ridge-based vessel segmentation in color images of the retina," *IEEE Transactions on Medical Imaging*, vol. 23, no. 4, pp. 501-509, April 2004.
- [8] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1999.
- [9] R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, Prentice Hall, 2008.
- [10] A. Koschan and M. Abidi, *Digital Color Image Processing*, Wiley, 2008.
- [11] K.P. Murphy, *Machine Learning a Probabilistic Perspective*, MIT, 2012.