Journal of Computer and Communications, 2024, 12, 116-141
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

Line Patterns Segmentation in Blurred Images Using Contrast Enhancement and Local Entropy Thresholding

Marios Vlachos[1], Evangelos Dermatas[2]

[1]Institute of Communication and Computer Systems, Athens, Greece
[2]Department of Computer Engineering and Informatics, University of Patras, Patras, Greece
Email: marios.vlachos@iccs.gr, dermatas@ceid.upatras.gr

How to cite this paper: Vlachos, M. and Dermatas, E. (2024) Line Patterns Segmentation in Blurred Images Using Contrast Enhancement and Local Entropy Thresholding. *Journal of Computer and Communications*, **12**, 116-141.
https://doi.org/10.4236/jcc.2024.122008

Received: November 13, 2023
Accepted: February 24, 2024
Published: February 27, 2024

Abstract

Finger vein extraction and recognition hold significance in various applications due to the unique and reliable nature of finger vein patterns. While recently finger vein recognition has gained popularity, there are still challenges associated with extracting and processing finger vein patterns related to image quality, positioning and alignment, skin conditions, security concerns and processing techniques applied. In this paper, a method for robust segmentation of line patterns in strongly blurred images is presented and evaluated in vessel network extraction from infrared images of human fingers. In a four-step process: local normalization of brightness, image enhancement, segmentation and cleaning were involved. A novel image enhancement method was used to re-establish the line patterns from the brightness sum of the independent close-form solutions of the adopted optimization criterion derived in small windows. In the proposed method, the computational resources were reduced significantly compared to the solution derived when the whole image was processed. In the enhanced image, where the concave structures have been sufficiently emphasized, accurate detection of line patterns was obtained by local entropy thresholding. Typical segmentation errors appearing in the binary image were removed using morphological dilation with a line structuring element and morphological filtering with a majority filter to eliminate isolated blobs. The proposed method performs accurate detection of the vessel network in human finger infrared images, as the experimental results show, applied both in real and artificial images and can readily be applied in many image enhancement and segmentation applications.

Keywords

Finger Vein, Vessel Enhancement, Vessel Network Extraction, Non-Uniform

Images, Binarization, Morphological Post-Processing

1. Introduction

The problem of finger vein extraction by processing infrared images arises mainly for biometrics purposes, but it is also very important for the biomedical research community. The problem of line pattern extraction in blurred images arises mainly when images of high-speed moving objects are acquired in low-light conditions, diffusion, fog, light scattering effects, low-quality optics or inexpensive photo-electric transducers are used for image acquisition.

The general structure of a biometric system based on finger veins has five main stages: 1) acquisition of the infrared images exploiting the absorption of light in near-infrared and infrared wavelengths by the different human tissues, 2) preprocessing of the acquired images, which include ROI (region of interest) extraction, image intensity normalization (in this type of images intensity is usually uneven due to the image acquisition system and may suffer from shading artifacts) and noise reduction, 3) segmentation or classification stage in which the preprocessed image is divided into two (or more) regions associated with veins and surrounding tissues, 4) post-processing of the binary images, which deliver the final segmentation result, free of outliers and misclassifications, and finally, 5) matching of the extracted veins in order to perform the desired person verification procedure. Matching procedure can be applied either directly in the extracted finger vein patterns or in their skeletons, depending on the matching algorithm that has to be used. This general structure described so far involves all the stages that may have such a system, but it is worth mentioning that these stages are independent and some of them can be skipped in some applications depending on their specific requirements.

Several methods that adopt this general architecture have already been proposed starting from the pioneering work of Park *et al.* [1]. In this important research work, an application-specific processor for vein pattern extraction and its application to a biometric identification system is proposed. The conventional vein-pattern-recognition algorithm consists of a preprocessing part, applying sequentially an iterative Gaussian low-pass, a high-pass, and a modified median filter and a recognition part, which includes the extraction of the binary veins via local thresholding and finally, the matching between the individual patterns. Consequently, the conventional algorithm [1] [2] [3] consists of low-pass spatial filtering for noise removal, and high-pass spatial filtering for emphasizing vascular patterns, thresholding and matching.

An improved vein pattern extraction algorithm is proposed in [4], which compensates for the loss of vein patterns in the edge area, gives more enhanced and stabilized vein pattern information, and shows better performance than the existing algorithm.

The problem with conventional hand vascular technology mentioned above is that the vascular pattern is extracted without taking into account its direction. So, there is a loss of vascular connectivity, which leads to a degradation of the performance of the verification procedure. An attempt to improve this problem can be found in [5], where a direction-based vascular pattern extraction algorithm based on the directional information of vascular patterns is presented for biometric applications.

Although, the above algorithm considers the directionality of veins, assumes that the veins are oriented in only two principal directions. In [6] [7], a method for personal identification based on finger-vein patterns is presented and evaluated using line tracking starting from various positions. This method allows vein patterns to have an arbitrary direction.

Typically, the infrared images of finger veins are low-contrast images, due to the light scattering effect. An algorithm for finger vein pattern extraction in infrared images is proposed in [8]. This algorithm embeds all the above issues and proposes novel preprocessing and post-processing algorithms.

In [9], new issues are considered and a certification system that compares vein images for low-cost, high speed and high-precision certification is proposed. Several noise reduction filters, sharpness filters and histogram manipulations were tested for best effort. As a result, a high certification ratio in this system was obtained.

In [10], the theoretical foundation and difficulties of hand vein recognition are introduced at first. Then, the optimum threshold of the segmentation process and the vein-lines thinning problem of infrared hand images are deeply studied, followed by the presentation of a novel estimator for the segmentation threshold and an improved conditional thinning method. The method of hand vein image feature extraction based on end points and crossing points is studied initially, and the matching method based on a distance measure is used to match vein images. The matching experiments indicated that this method is efficient in terms of biometric verification.

However, the finger vein technology, as mentioned above, has important applications in the biomedical field from which it originated from. An initial work for localizing surface veins via near-infrared (NIR) imaging and structured light ranging is presented in [11]. The eventual goal of the system is to serve as the guidance for a fully automatic (*i.e.* robotic) catheterization device. The proposed system is based upon near-infrared (NIR) imaging, which has previously been shown effective in enhancing the visibility of surface veins.

Also, in [12] [13], a vein contrast enhancer (VCE) has been constructed to make vein access easier by capturing an infrared image of veins, enhancing the contrast using software, and projecting the vein image back onto the skin.

Although these methods achieve to segment the infrared images, the finger vein pattern extraction task is still challenging mainly due to the fact that infrared images suffer from strong noise presence, uneven illumination and shading, factors that complicate the application of automatic image segmentation techniques. So, another way to segment this kind of image is to assume that veins are located in thin and concave regions (a reasonable assumption obtained by a careful inspection of the image intensity across the image) of infrared images and based on this concept to extract them by optimizing a mathematical model. This can be done by using the Mumford-Shah model, which has well-known capabilities in the image processing applications such as image segmentation, restoration and image inpainting [14] [15]. So, in this paper, we derive an analytical solution to a modified Mumford-Shah model minimization problem and we propose a local application of its results in order to perform fast and accurate finger vein extraction.

In this paper, a novel method to segment finger vessel networks and extract the corresponding pattern is presented. The proposed algorithm is composed of several steps. A finger vein enhancement procedure (second step) is performed in order to facilitate the application of the third step which involves segmentation using an adaptive threshold derived from a local estimation of the brightness entropy. The adaptive threshold method is adopted due to its simplicity and robustness to segment the image of the finger into two regions: vein and tissue. The two additional steps (first and fourth), a preprocessing step and a post-processing step, are applied only when required in order to ease the whole procedure and to give a more robust result.

The paper is organized as follows. In Section 2, a detailed presentation of the line pattern extraction method is given. The experimental results and a discussion are included in Section 3. Finally, the most significant conclusions and some directions for future work are presented in the last section of this paper.

2. Materials and Method

2.1. Image Preprocessing

In **Figure 1**, a flowchart of the proposed line-patterns extraction method is given.

ROI extraction. Depending on the application, one or several, manual or automatic ROIs are extracted in the original image. Among the automatic *ROI* extraction techniques [16] [17] [18], the histogram-based statistical methods are very popular due to their computational efficiency. In applications where the background can be detected in the very low or very high brightness areas, simple histogram rules can be used for fast *ROI* extraction. A typical *ROI* in finger infrared images is shown in **Figure 2(a)**.

Brightness normalization based on local statistical measures. In general, even in cases where leds' brightness is adjusted to satisfy several statistical properties in the acquired infrared image, in few areas unsatisfactory illumination or strong noise distortions are met. This effect appears due to the complicated structure of human tissue. Therefore, an image normalization procedure is applied to restore partially the desirable statistical properties.

A simple, fast and efficient local normalization procedure is used to unify the

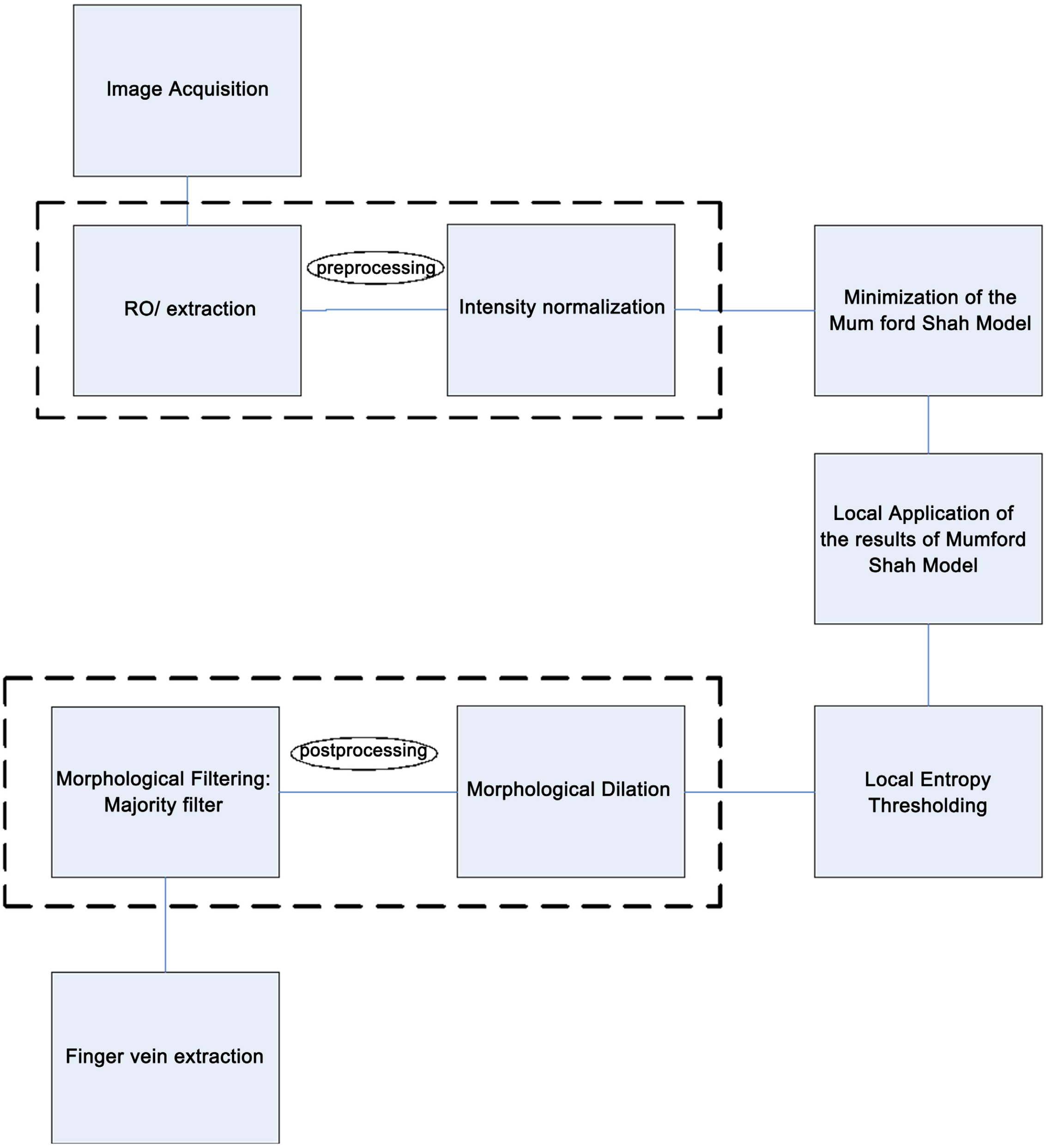


Image Acquisition
RO/ extraction
preprocessing
Intensity normalization
Minimization of the Mum ford Shah Model
Local Application of the results of Mumford Shah Model
Local Entropy Thresholding
Morphological Dilation
postprocessing
Morphological Filtering: Majority filter
Finger vein extraction

Figure 1. Flowchart of the proposed line patterns segmentation method.

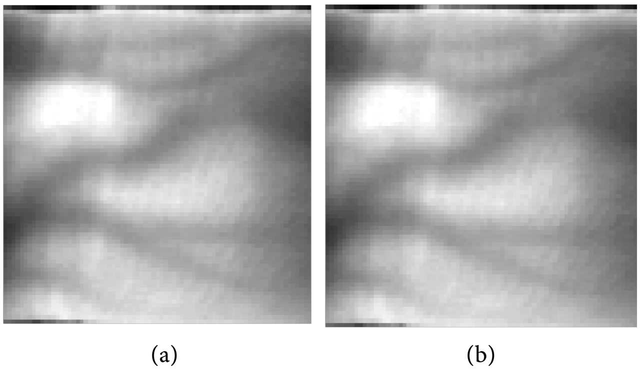


Figure 2. A region of interest (*ROI*), before (a) and after brightness normalization (b).

local mean and variance of the *ROI*, an especially useful technique for correcting non-uniform illumination or shading artifacts, using a linear brightness transformation scheme applied on pixels' brightness:

$$I_{0n}(x,y)=\frac{I_0(x,y)-m_{I_0}(x,y)}{\sigma_{I_0}(x,y)}, \quad (1)$$

where $I_0(x,y)$ is the brightness of the original image at pixel (*x*, *y*), $m_{I_0}(x,y)$ is the brightness local mean and, $\sigma_{I_0}(x,y)$ is the corresponding local standard deviation. The estimation of the local mean and standard deviation is performed using the brightness values of neighbor pixels. **Figure 2(b)** shows the *ROI* image after the application of the brightness normalization method.

2.2. Enhancement of Concave Regions

The finger veins are significantly thinner than the darker structures met in typical infrared images as shown in **Figure 2(a)**. Multiple scattering of the propagated photons significantly reduces the contrast, eliminates the tiny veins and increases the transition regions between the main veins and the surrounding tissue. The "fog" effect hides the vein structures in concave regions of the *ROI*. This assumption could be verified by observing the cross-section profile of the veins which is Gaussian like as claimed in [19]. The aim of the proposed vein enhancement method is to amplify concave regions using several connectivity properties.

In typical images, concave regions are detected by examining the positiveness of the second order spatial derivative. Direct estimation of the derivatives in digital images is an ill posed problem due to noise presence and the non-uniform illumination. Instead of seeking regions which have positive second order derivatives, in this paper the minimization of Equation (2), similar to the Mumford-Shah objective function [20], is proposed. The objective is to estimate a smooth image *I*(., .) similar to the normalized image I_{0n}(., .), *i.e.* decreasing the absolute value of the first derivative. The equivalent mathematical expression leads to the following minimization problem:

$$J_c(I)=\frac{1}{2}\cdot\int_\Omega|\nabla I|^2\,\mathrm{d}x+\frac{\lambda}{2}\cdot\int_\Omega|I-I_{0n}|^2\,\mathrm{d}x, \quad (2)$$

where Ω is the image domain, *λ* is the smoothness weight, and ∇I is the Laplacian gradient of the image *I*(., .).

The minimization of this function is computationally intensive and can be performed by the method proposed by Chan *et al.* [14]. This method belongs to the category of segmentation methods which use partial differential equations (PDEs) and it is iterative. In this paper, a direct solution to the adopted optimization criterion is reached. The proof of the direct-form solution is given in **Appendix**.

The smoothed image *I*(., .) at the minimum of Equation (2) is estimated using Algorithm 1.

***Image-based solution. Algorithm* 1.**

Assuming an image I_{0n} of N rows and M columns, the function $J_c(I)$ in the discrete space becomes:

$$J_c\left(I\right)=\frac{1}{2}\cdot\sum_{i=1}^{N}\sum_{j=1}^{M}\left|\nabla I\right|^2+\frac{\lambda}{2}\cdot\sum_{i=1}^{N}\sum_{j=1}^{M}\left|I-I_{0n}\right|^2. \tag{3}$$

If the partial derivatives, involved in the computation of the Laplacian gradient, are approximated using local differences and substituted back in Equation (3), the following objective function obtained:

$$\begin{aligned}J_c\left(I\right)=&\frac{1}{2}\cdot\sum_{x=1}^{N-1}\sum_{y=1}^{M-1}\left[\left(I\left(x+1,y\right)-I\left(x,y\right)\right)^2+\left(I\left(x,y+1\right)-I\left(x,y\right)\right)^2\right]\\&+\frac{\lambda}{2}\cdot\sum_{x=1}^{N-1}\sum_{y=1}^{M-1}\left[I\left(x,y\right)-I_{0n}\left(x,y\right)\right]^2\end{aligned} \tag{4}$$

The minimum of Equation (4) regarding to $I(.,.)$ can be derived in a close-form by differencing the second-order, positively defined function. The above minimization results in a system of $N\times M$ linear equations from which the unknown image $I(.,.)$ can be readily derived:

$$G\cdot\boldsymbol{I}=\lambda\cdot\boldsymbol{I}_{0n}, \tag{5}$$

where G is a sparse Hermitian matrix depends only on parameter λ, $\boldsymbol{I}$ is the vector form of the unknown image $I(.,.)$ and $\boldsymbol{I}_{0n}$ is the vector form of the normalized image $I_{0n}(.,.)$. If matrix G is invertible, the brightness of the unknown image is derived from the solution of the system of linear equations:

$$\boldsymbol{I}=G^{-1}\cdot\left(\lambda\cdot\boldsymbol{I}_{0n}\right). \tag{6}$$

The enhanced image I_e is created as the difference between the smoothed image $I(.,.)$ and the normalized image $I_{0n}(.,.)$:

$$I_e=I-I_{0n}. \tag{7}$$

In practice, even in the case of small images the inversion of a sparse matrix G is highly computationally prohibitive. Fortunately, the off-line estimation of the inverse of the matrix G reduces significantly the computations, *i.e.* for an arbitrary image of $N\times M$ pixels, the estimation of pixel brightness of the enhanced image requires $N\times M$ multiplications and $N\times M$ additions.

***Sub-image solution. Algorithm* 2.**

Instead of deriving the global minimum, in this paper a closed-form solution of Equation (2) is proposed by processing small rectangular areas. The proposed solution accelerates significantly the processing time required for each pixel and outperforms the classical approach [14]. As a result, fast and accurate extraction of the finger veins is obtained.

An effective reduction of the matrix G dimensionality can be achieved using sub-images, *i.e.* multiple solutions of Equation (6) can be achieved using only a small number of neighbor pixels. In this case the number of linear equations depends on the size of the chosen window.

In order to obtain the image $I_e = I - I_{0n}$ the following process is applied. Initially a window of size $Z\times Z$ is selected and the corresponding matrix G ($Z^2\times$

Z^2) and its inverse $X(Z^2 \times Z^2)$ are computed. Then, the sliding window slides along each pixel of the normalized image and the difference between the pixel values of the image $I(.,.)$ obtained by Equation (6) and the pixel values of the normalized image I0n inside the window is computed. In the next step all these overlapped window differences are added and the image $I_e = I - I_{0n}$ is obtained by keeping only the central $N \times M$ part of the result (where $N \times M$ is the size of the original image).

The result of this process is the non-smooth image I_e. In this image, the veins are located in concave regions and thus a local entropy thresholding technique is applied in order to segment the non-smooth image in concave (veins) and non-concave (other tissue parts) regions.

2.3. Local Entropy Thresholding

Among the various methods to define automatically the threshold for segmentation local entropy thresholding is selected, which has been successfully used in [21], because in typical images the brightness of pixel neighbors is related and therefore efficient entropy-based thresholding takes into account the spatial brightness distribution. Taking into account that the co-occurrence matrix of the image I_e is a measure of the brightness transition between adjacent pixels, a local entropy thresholding technique described in [22] is implemented, which can preserve the structure details of an image. Two images with identical histograms but different spatial distribution will result in different entropy and consequently different threshold values.

The co-occurrence matrix of the image $I_e = I - I_{0n}$ is a PxQ dimensional matrix $T = \left[t_{ij}\right]_{P\times Q}$ giving quantitative measures of the spatial structural information of an image. Depending upon the ways in which the gray level i follows gray level j different definitions of co-occurrence matrix are possible. Here, the co-occurrence matrix is asymmetric by considering the horizontally right and vertically lower transitions. Thus, t_{ij} is defined as follows:

$$\begin{aligned} t_{ij} = \sum_{l=1}^{P}\sum_{k=1}^{Q} u\Big(\delta\big(I_e(l,k)-i\big)\delta\big(I_e(l,k+1)-j\big) \\ +\delta\big(I_e(l,k)-i\big)\delta\big(I_e(l+1,k)-j\big)-1\Big). \end{aligned} \tag{8}$$

The probability of co-occurrence p_{ij}, of gray levels i and j can be written as:

$$p_{ij} = \frac{t_{ij}}{\sum_i \sum_j t_{ij}}. \tag{9}$$

If s, $0 \le s \le L-1$, is a threshold, then s can partition the co-occurrence matrix into 4 quadrants, namely A, B, C and D (**Figure 3**).

Let us define the following quantities:

$$P_A = \sum_{i=0}^{s}\sum_{j=0}^{s} p_{ij}, \tag{10}$$

$$P_C = \sum_{i=s+1}^{L-1}\sum_{j=s+1}^{L-1} p_{ij}. \tag{11}$$

From the occurrence-matrix, the corresponding sum of probabilities within

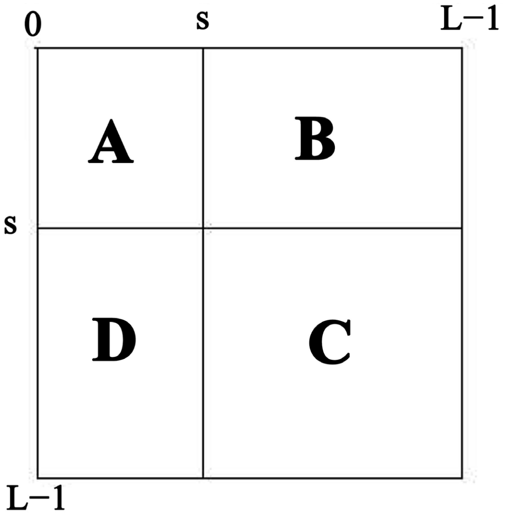


Figure 3. Quadrants of co-occurrence matrix.

each individual quadrant is equal to one. Thus, the following cell probabilities for different quadrants are obtained:

$$P_{ij}^{A}=\frac{p_{ij}}{P_A}=\frac{t_{ij}}{\sum_{i=0}^{s}\sum_{j=0}^{s}t_{ij}},\ \text{for } 0\le i\le s,0\le j\le s, \tag{12}$$

$$P_{ij}^{C}=\frac{p_{ij}}{P_C}=\frac{t_{ij}}{\sum_{i=s+1}^{L-1}\sum_{j=s+1}^{L-1}t_{ij}},\ \text{for } s+1\le i\le L-1, s+1\le j\le L-1. \tag{13}$$

The second-order entropy of the object can be defined as:

$$H_{A}^{(2)}(s)=-\frac{1}{2}\cdot\sum_{i=0}^{s}\sum_{j=0}^{s}P_{ij}^{A}\cdot\log_2 P_{ij}^{A}. \tag{14}$$

Similarly, the second-order entropy of the background can be written as:

$$H_{C}^{(2)}(s)=-\frac{1}{2}\cdot\sum_{i=s+1}^{L-1}\sum_{j=s+1}^{L-1}P_{ij}^{C}\cdot\log_2 P_{ij}^{C}. \tag{15}$$

Hence, the total second-order local entropy of the object and the background can be written as:

$$H_{T}^{(2)}(s)=H_{A}^{(2)}(s)+H_{C}^{(2)}(s). \tag{16}$$

The brightness gray level corresponding to the maximum of $H_{T}^{(2)}(s)$ gives the optimal threshold for object-background classification.

In a correction note, Chanwimaluang and Fanpropose two modifications to improve the results of blood vessel extraction that is essential to the performance of image registration. These modifications were adopted also in our study because they experimentally proved superior to [21].

In the first modification a different definition of the co-occurrence matrix is adopted, increasing the local entropy values in vein structures. As mentioned, the co-occurrence matrix of an image shows the intensity transitions between adjacent pixels. The original co-occurrence matrix is asymmetric by considering the horizontally right and vertically lower transitions. They added some jittering effect to the co-occurrence matrix that tends to keep the similar spatial structure but with much less variations, *i.e.* $T = \left[t_{ij}\right]_{P\times Q}$ is computed as follows:

For every pixel (*l*, *k*) in an image I_e:

$$\begin{aligned} i &= I_e(l,k), \\ j &= I_e(l,k+1), \\ d &= I_e(l+1,k+1), \\ t_{ij} &= t_{id}+1. \end{aligned} \tag{17}$$

One may wonder whether the modified co-occurrence matrix still describes the original spatial structure. Actually, considering a smooth area in an image where j and d should be very close or identical, Equation (17) implicitly introduces a certain smoothing effect and adds some structured noise to the co-occurrence matrix. The two matrices still share a similar structure that is important for the valid thresholding result. Also, the latter one has larger entropy with a much smaller standard deviation, which is more desirable for local entropy thresholding.

Moreover, the sparse foreground in selecting the optimal threshold is studied. The original threshold selection criterion aims to maximize the local entropy of foreground and background in a gray-scale image without considering the small proportion of foreground. Therefore, they proposed to select the optimal threshold that maximizes the local entropy of the binarized image that indicates the foreground/background ratio. The larger the local entropy, the more balanced the ratio between foreground and background in the binary image.

2.4. Post-Processing

The resulting binary image tends to suffer from some misclassifications, *i.e.* outliers.

Morphological dilation. The outliers can be drastically reduced using morphological dilation [23] with a line structuring element oriented along the x-axis and elongated Y pixels. The output of this step is an image with fewer outliers but several misclassifications remain.

Morphological filtering. In this substep, the binary image is processed by applying iteratively a morphological filter called majority [23]. This filter sets a pixel to 1 if five or more pixels in its 3-by-3 neighborhood have the value 1, otherwise, it sets the pixel to 0. Majority filter is applied iteratively until the output image remains unchanged. This application clears the image from small misclassified regions which appear in image due the presence of noise and smoothes the contours.

3. Experimental Results

3.1. Detection of Line-Patterns Vein Network

In a typical infrared image of a human finger (**Figure 4**), several physical phenomena appear during light propagation through human tissue, *i.e.* absorption,

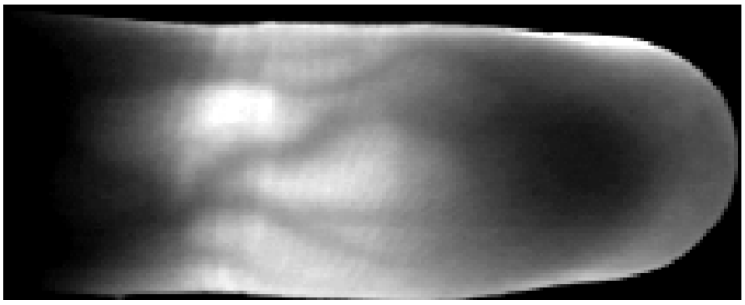


Figure 4. Original image [6] [7].

diffusion, and scattering [24]. Moreover, the great number of substances contained in the human body, the blood dynamics and the mass transfer phenomena complicate significantly the light transformation effects. Therefore, the solution of the inverse light propagation problem, *i.e.* the derivation of the tissue structure arterial from the image data becomes unrealistic. A totally different and popular approach, adopted also in the proposed vein-network detection method, use several image enhancements, feature extraction and path reconstruction methods to derive the vein network, based on the fact that hemoglobin cells have strong absorption coefficient in the infrared light, and therefore the veins appear in the image darker than the other human tissues (**Figure 4**).

A typical hardware used to acquire infrared images consist of a finger probe, an array of infrared leds with adjustable illumination, and a video camera focus on frame, as shown in **Figure 5**. The finger *ROI* is placed inside the probe, between the open frame and the array of infrared leds light source, which consists of a number of leds with adjustable illumination. The finger probe eliminates the influence of the external light sources.

The original image was acquired under infrared light using an inexpensive *CCD* camera. The finger was placed between the camera and the light source which consists of a row of infrared leds (five elements) with adjustable illumination. The intensity of the leds adjusted as far as the illumination of the image was good enough. In the designed hardware each infrared led has adjustable intensity, giving very fine image quality, minimizing also the variance of the automatic exposure times of the image acquisition system.

An excellent image illumination is not a strict requirement because the good performance of our algorithm remains also under adverse illumination conditions. Due to the fact that hemoglobin has strong absorption in the infrared light the veins are shown in the image darker than the other human tissues. So, the goal of our study is to extract these dark regions, corresponding to veins, from the background, corresponding to the other human parts (tissue). The original image which was acquired as described here is shown in **Figure 4**.

In this section, the results of the execution of our algorithm in the ROI image shown in **Figure 2** are presented. For this image a window neighborhood of size 9 × 9 is used which results in an 81 × 81 matrix *M* which can be inverted very quickly. The selection of the parameter λ does not affect the performance of our

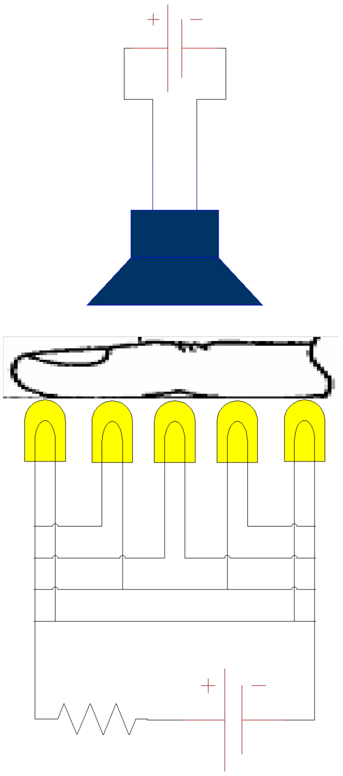


Figure 5. A typical low-cost device for digital image acquisition of finger infrared images.

algorithm, so is arbitrary select $\lambda = 1$. For the threshold computation the modified local entropy thresholding technique described in the correction note is used. In post-processing, a line structuring element with $Y = 3$ pixels length and oriented in the x-axis is selected.

In Figure 6, the results of the execution of our algorithm in the *ROI* image of Figure 2. Figure 6(a) shows the non-smooth image obtained after the application of the minimization results of the local Mumford-Shah model. Figure 6(b) shows the detection, with the help of the modified local entropy thresholding, of the concave regions of image, in which the veins tend to locate. In this binary image concave regions (candidate pixels to be detected as veins) shown in black while other tissue parts shown in white. Figure 6(c) shows the binary image after the application of the morphological dilation substep and the final image in Figure 6(d) shows the extracted finger vein pattern obtained after the final morphological filtering substep.

Finally, the results of the execution of our algorithm in another *ROI* image are presented. The same parameters as in the previous case are used. Figure 7 shows the results.

3.2. Creation of Artificial Infrared Images

A quantitative evaluation of the proposed algorithm in real infrared images is difficult to obtain due to the absence of manual segmentation data. The extremely low-contrast images increase the disagreement of human annotation.

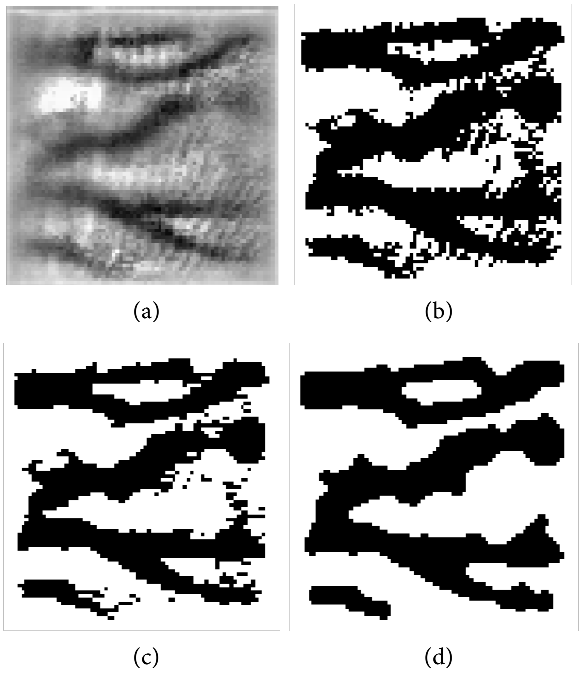


Figure 6. (a) Non-smooth image, (b) Concave (black) and non-concave regions (white), (c) Morphological dilation, and (d) Majority filtering, extracted finger vein pattern.

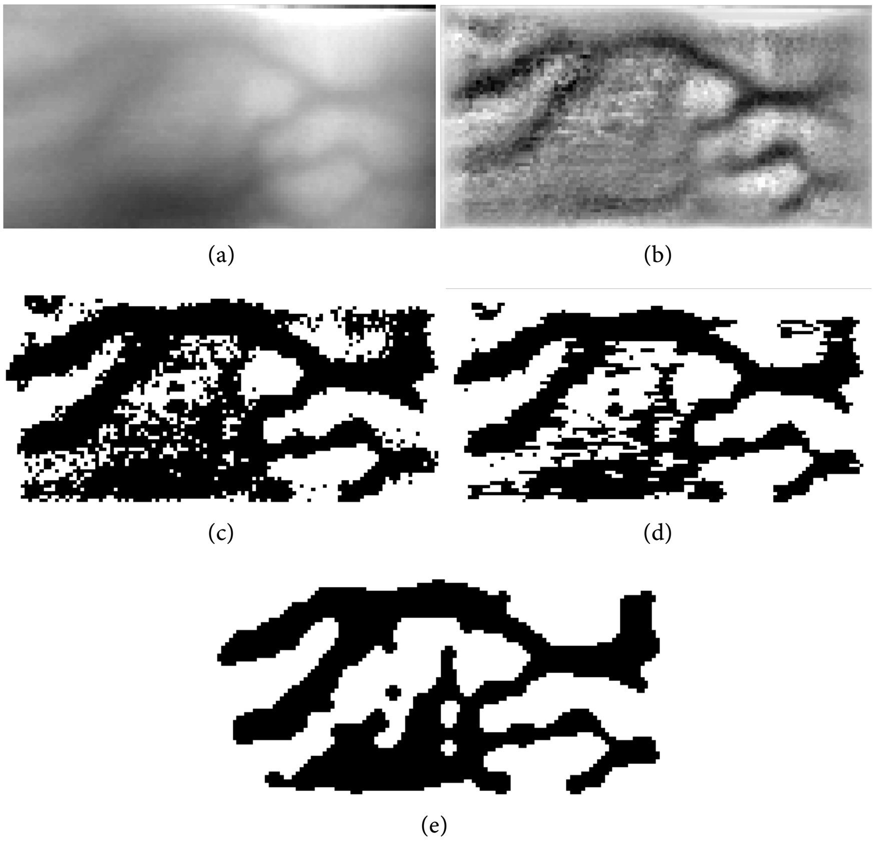


Figure 7. (a) *ROI* image, (b) Non-smooth image, (c) Concave (black) and non-concave regions (white), (d) Morphological dilation, and (e) Morphological (majority) filtering, extracted finger vein pattern.

Therefore, the proposed method is evaluated using a small set of images each one created according to the following procedure. This construction involves the multiplication of two different layers. The first layer is a vein-like pattern. This pattern consists of connected lines of different widths with junctions and bifurcations which are drawn by hand. The second layer is the non-uniform image background which is created by applying an iterative spatial low-pass Gaussian filter with large window size to the original infrared image. The multiplication of the two layers gives the artificial infrared image used in the experiments.

Evaluation rates. In the finger vessel segmentation process, each pixel is classified as tissue of the vessel. Consequently, there are four events, true positive (TP) and true negative (TN) when a pixel is correctly segmented as a vessel or non-vessel, and two misclassifications, a false negative (FN) appears when a pixel in a vessel is segmented in the non-vessel area, and a false positive (FP) when a non-vessel pixel is segmented as a vessel-pixel.

Two widely known statistical measures are used for algorithm evaluation: sensitivity and specificity, which are used to evaluate the performance of the binary segmentation outcome. The sensitivity is a normalized measure of true positives, while specificity measures the proportion of true negatives:

$$\text{sensitivity} = \frac{TP}{TP + FN}, \tag{18}$$

$$\text{specificity} = \frac{TN}{TN + FP}. \tag{19}$$

The tradeoff between the two measures is graphically represented with the receiver operating characteristic curve (ROC), which is a plot of the sensitivity versus 1-specificity. Equivalently, the ROC curve can be represented by plotting the true positive rate (TPR) versus the false positive rate (FPR). These rates are the fractions of TPs and FPs:

$$TPR = \frac{TP}{TP + FN} = \text{sensitivity}, \tag{20}$$

$$FPR = \frac{FP}{FP + TN} = 1 - \frac{TN}{TN + FP} = 1 - \text{specificity}. \tag{21}$$

The accuracy of the binary classification is defined by:

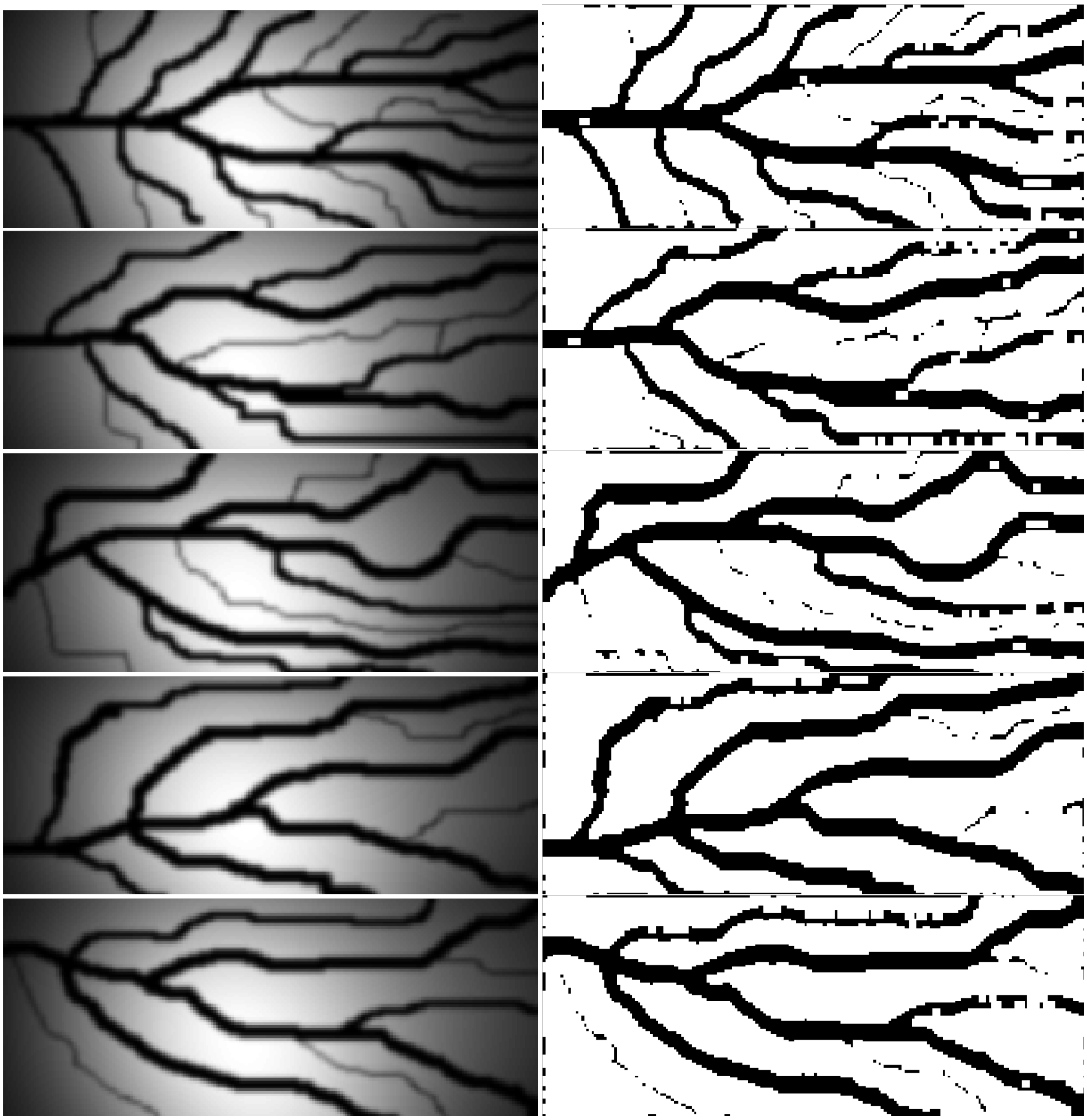
$$\text{accuracy} = \frac{TP + TN}{P + N}, \tag{22}$$

where P and N represent the total number of positives (vessel) and negatives (non-vessel) pixels in the segmentation process and is the degree of conformity of the estimated binary classification to the ground truth according to a manual segmentation. Thus, the accuracy is strongly related to the segmentation quality and for this reason is used to evaluate and compare different methods.

Table 1 shows the evaluation rates in terms of sensitivity, specificity and accuracy for ten artificial images of the image database while **Figure 8** shows these images and the corresponding finger vein patterns extracted using the proposed algorithm.

Table 1. Evaluation rates (sensitivity, specificity, accuracy) for ten artificial images.

Image name	*Sensitivity*	*Specificity*	*Accuracy*
Fv01.bmp	0.877	0.927	0.914
Fv02.bmp	0.872	0.913	0.904
Fv03.bmp	0.897	0.918	0.913
Fv04.bmp	0.947	0.913	0.921
Fv05.bmp	0.910	0.921	0.919
Fv06.bmp	0.690	0.867	0.830
Fv07.bmp	0.927	0.931	0.930
Fv08.bmp	0.950	0.915	0.922
Fv09.bmp	0.922	0.889	0.896
Fv10.bmp	0.944	0.908	0.916
Average	0.894	0.910	0.907
Standard deviation	0.077	0.019	0.029



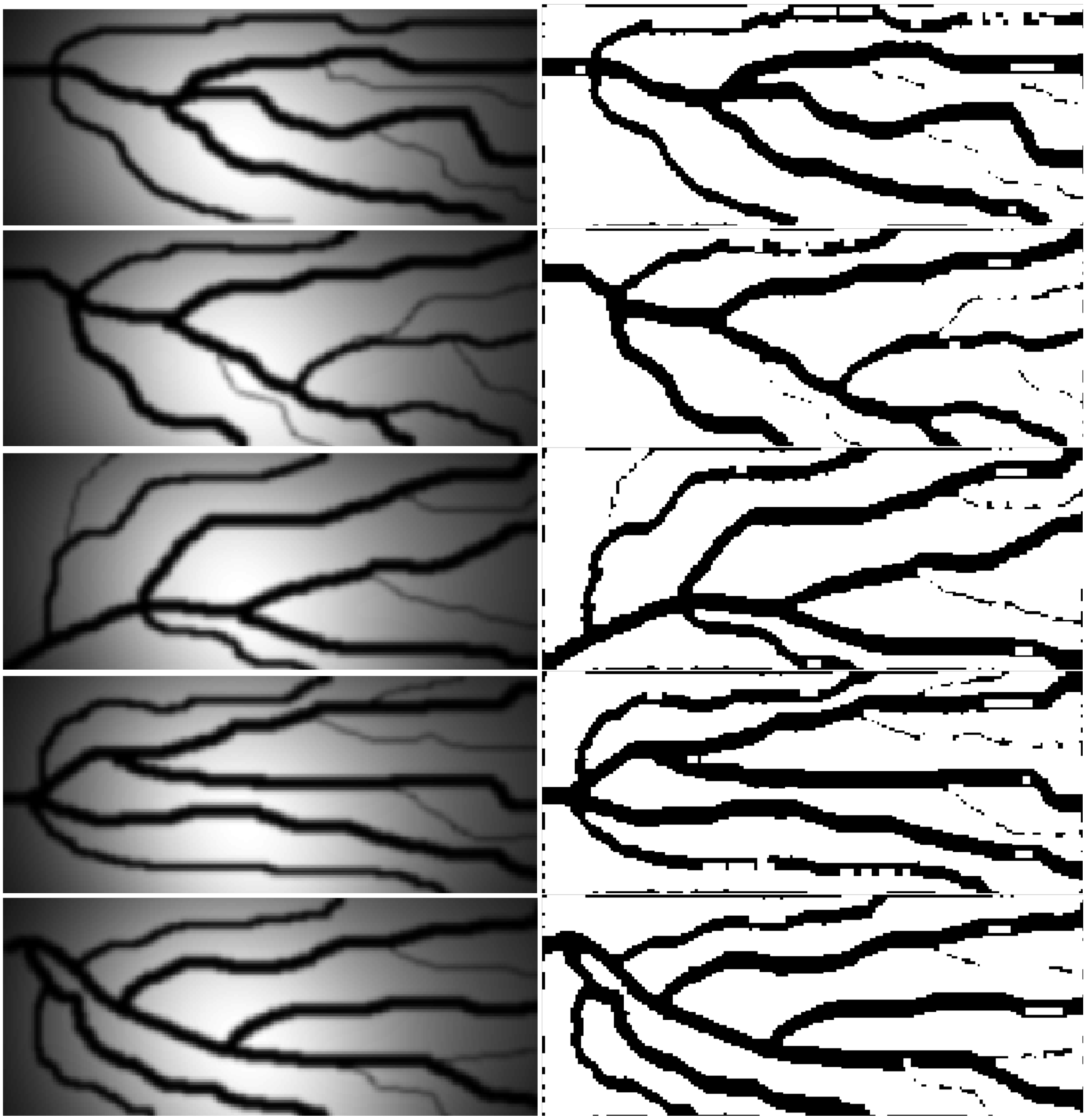


Figure 8. Ten artificial images (left) and the corresponding finger vein patterns (right) extracted using the proposed algorithm.

Figure 9 shows one artificially created infrared image, the non-smooth image estimated using Algorithm 2 (sub-image solution), the binary image obtained after local entropy thresholding and the binary image after morphological post-processing, which is the extracted finger vein pattern. Figure 10 shows the *ROC* curve obtained after varying the threshold, estimating the evaluation rates and plotting the results.

In Table 2, the evaluation rates in terms of sensitivity, specificity and accuracy for artificial images with different degrees of blurriness are shown in order to indicate the robustness of the proposed algorithm under different smoothness while Figure 11 shows the artificial images and the extracted finger vein patterns.

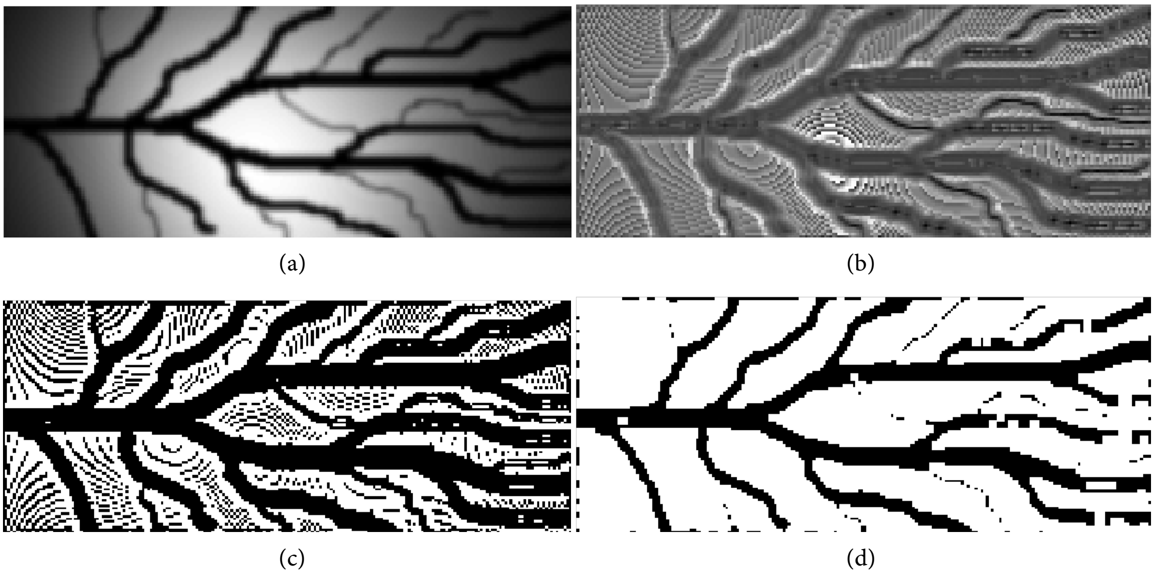


Figure 9. (a) Artificial image, (b) Non-smooth image, (c) Binary image obtained after local entropy thresholding, and (d) Binary image after morphological post-processing (finger vein pattern).

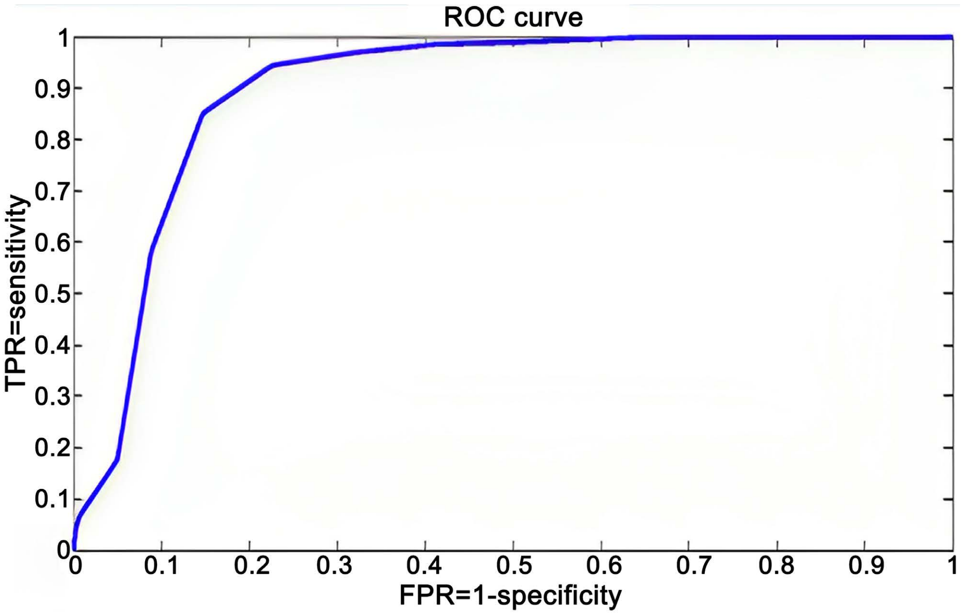
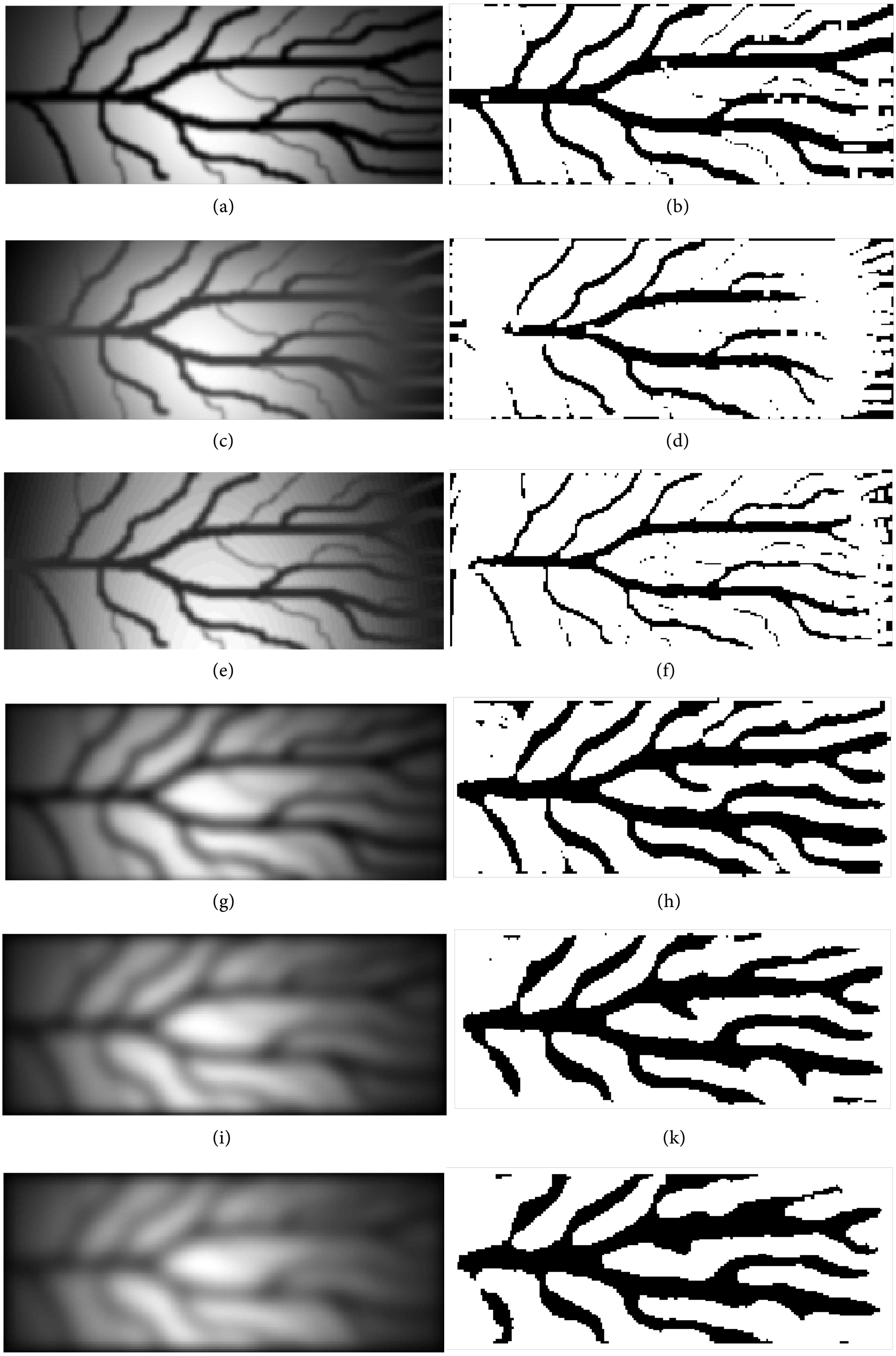


Figure 10. ROC curve.

Table 2. Evaluation rates (accuracy, sensitivity, specificity) for the artificial images.

Image name	*Sensitivity*	*Specificity*	*Accuracy*
ArtificialVein.bmp	0.878	0.927	0.915
ArtificialVein2.bmp	0.523	0.946	0.844
ArtificialVein3.bmp	0.614	0.972	0.886
P1.bmp	0.885	0.843	0.853
P2.bmp	0.810	0.830	0.825
P3.bmp	0.775	0.814	0.805
P4.bmp	0.735	0.809	0.791



(a) (b)

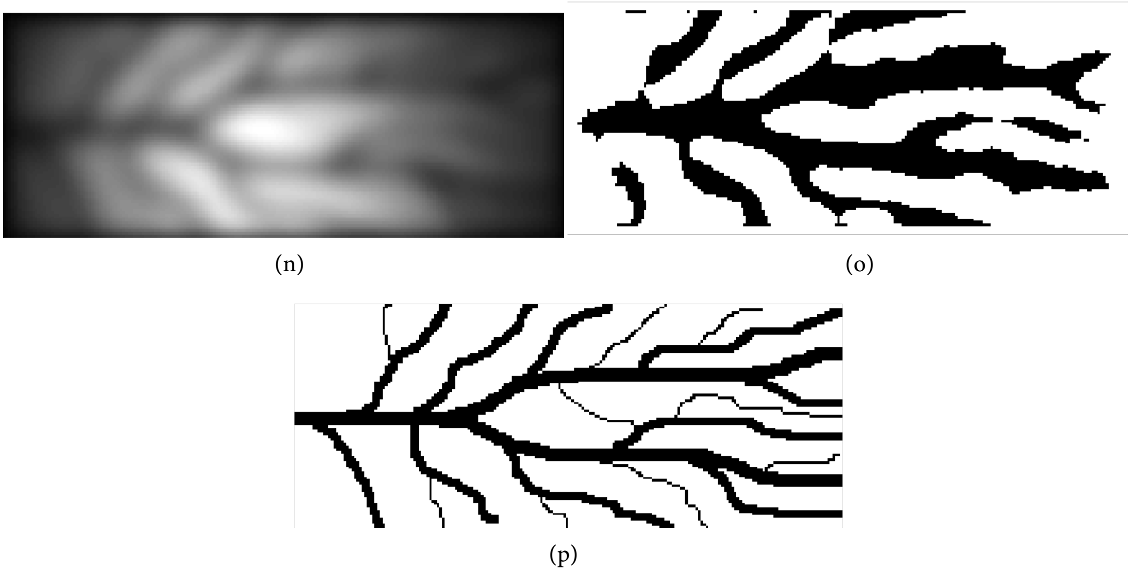
(c) (d)

(e) (f)

(g) (h)

(i) (k)

(l) (m)


(n)

(o)

(p)

Figure 11. (a, c, e, g, i, l, n) Artificial image with different degree of distortion, (b, d, f, h, k, m, o) Corresponding finger vein patterns, and (p) Ground truth.

From the images shown in Figure 11 and from the results presented in Table 2, it is evident that the proposed algorithm is robust against different degrees of blurriness.

4. Conclusions

In this paper, an efficient finger vein pattern extraction algorithm is presented. The algorithm is based on the minimization of the objective function of the Mumford-Shah model and the local application of its results. This application produces a non-smooth image where veins are located in concave regions. Detection of these regions is achieved via a modified local entropy thresholding technique. The preliminary segmentation result was unsatisfactory due to the presence of some outliers (misclassifications).

So, a final morphological post-processing result followed in order to clean the image from the misclassifications and produce a robust finger vein pattern. Future work includes the improvement of our imaging device in order to acquire images with less shading and noise artifacts, something that will guarantee the successful application of our algorithm in the majority of cases. In the case of images with high quality, the preprocessing and/or post-processing step can be skipped.

Acknowledgements

This work is partially supported by grant KARATHEODORIS of the University of Patras.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Park, G.T., Im, S.K. and Choi, H.S. (1997) A Person Identification Algorithm Utilizing Hand Vein Pattern. *Proceedings of Korea Signal Processing Conference*, **10**, 1107-1110.

[2] Hong, D.U., Im, S.K. and Choi, H.S. (1999) Implementation of Real Time System for Personal Identification Algorithm Utilizing Hand Vein Pattern. *Proceedings of IEEK Fall Conference*, **22**, 560-563.

[3] Im, S.K., Park, H.M., Kim, S.W., Chung, C.K. and Choi, H.S. (2000) Improved Vein Pattern Extracting Algorithm and Its Implementation. *Proceedings of IEEE International Conference on Consumer Electronics*, Los Angeles, 13-15 June 2000, 2-3.

[4] Im, S.K., Park, H.M. and Kim, S.W. (2001) A Biometric Identification System by Extracting Hand Vein Patterns. *Journal of the Korean Physical Society*, **38**, 268-272.

[5] Im, S.K., Choi, H.S. and Kim, S.-W. (2003) A Direction-Based Vascular Pattern Extraction Algorithm for Hand Vascular Pattern Verification. *ETRI Journal*, **25**, 101-108. https://doi.org/10.4218/etrij.03.0102.0211

[6] Miura, N., Nagasaka, A. and Miyatake, T. (2004) Feature Extraction of Finger-Vein Patterns Based on Repeated Line Tracking and Its Application to Personal Identification. *Machine Vision and Applications*, **15**, 194-203. https://doi.org/10.1007/s00138-004-0149-2

[7] Miura, N., Nagasaka, A. and Miyatake, T. (2004) Feature Extraction of Finger-Vein Patterns Based on Iterative Line Tracking and Its Application to Personal Identification. *Systems and Computers in Japan*, **35**, 61-71. https://doi.org/10.1002/scj.10596

[8] Vlachos, M. and Dermatas, E. (2006) A Finger Vein Pattern Extraction Algorithm Based on Filtering in Multiple Directions. 5*th European Symposium on Biomedical Engineering*, Patras, 7-9 July 2006.

[9] Tanaka, T. and Kubo, N. (2004) Biometric Authentication by Hand Vein Patterns. *SICE Annual Conference*, Vol. 1, Sapporo, 4-6 August 2004, 249-253.

[10] Ding, Y.H., Zhuang, D.Y. and Wang, K.J. (2005) A Study of Hand Vein Recognition Method. *Proceedings of the IEEE International Conference on Mechatronics & Automation*, Niagara Falls, 29 July-1 August 2005, 2106-2110.

[11] Paquit, V., Price, J.R., Seulin, R., Meriaudeau, F., Farahi, R.H., Tobin, K.W. and Ferrell, T.L. (2006) Near-Infrared Imaging and Structured Light Ranging for Automatic Catheter Insertion. *Proceedings of SPIE*, **6141**, Article No. 61411T.

[12] Zeman, H.D., Lovhoiden, G. and Vrancken, C. (2002) The Clinical Evaluation of Vein Contrast Enhancement. *Proceedings of SPIE*, **4615**, 61-70. https://doi.org/10.1117/12.476117

[13] Zeman, H.D., Lovhoiden, G. and Vrancken, C. (2004) Prototype Vein Contrast Enhancer. *Proceedings of SPIE*, **5318**. https://doi.org/10.1117/12.517813

[14] Chan, T. and Shen, J. (2005) Image Processing and Analysis: Variational, PDE, Wavelet and Stochastic Methods. Society for Industrial and Applied Mathematics, Philadelphia. https://doi.org/10.1137/1.9780898717877

[15] Esedoglu, S. and Shen, J. (2002) Digital Inpainting Based on the Mumford Shah Euler Image Model. *European Journal of Applied Mathematics*, **13**, 353-370. https://doi.org/10.1017/S0956792502004904

[16] Sun, X.D., Foote, J., Kimber, D. and Manjunath, B.S. (2005) Region of Interest Extraction and Virtual Camera Control Based on Panoramic Video Capturing. *IEEE Transactions on Multimedia*, **7**, 981-990. https://doi.org/10.1109/TMM.2005.854388

[17] Lamouri, N., Gharbi, M., Benabdellah, M., Regragui, F. and Bouyakhf, E.H. (2006) Adaptive Still Image Compression Using Semi-Automatic Region of Interest Extraction. *CCSP*2006, *Second International Symposium on Communications*, *Control and Signal Processing*, Marrakech, 13-15 March 2006.

[18] Agili, S., Balasubramanian, V. and Morales, A. (2007) Semi-Automatic Region of Interest Identification Algorithm Using Wavelets. *Optical Engineering*, **46**, Article ID: 035003. https://doi.org/10.1117/1.2713377

[19] Chaudhuri, S., Chatterjee, S., Katz, N., Nelson, M. and Goldbaum, M. (1989) Detection of Blood Vessels in Retinal Images Using Two Dimensional Matched Filters. *IEEE Transactions on Medical Imaging*, **8**, 263-269. https://doi.org/10.1109/42.34715

[20] Bar, L., *et al*. (2011) Mumford and Shah Model and Its Applications to Image Segmentation and Image Restoration. In: Scherzer, O., Ed., *Handbook of Mathematical Methods in Imaging*, Springer, New York, 1095-1157. https://doi.org/10.1007/978-0-387-92920-0_25

[21] Chanwimaluang, T. and Fan, G. (2003) An Efficient Algorithm for Extraction of Anatomical Structures in Retinal Images. *IEEE Proceedings of International Conference on Image Processing*, **1**, I1093-I1096.

[22] Pal, N.R. and Pal, S.K. (1989) Entropic Thresholding. *Signal Processing*, **16**, 97-108. https://doi.org/10.1016/0165-1684(89)90090-X

[23] Gongalez, R., Woods, R. and Eddins, S. (2020) Digital Image Processing Using MATLAB. Prentice Hall, Upper Saddle River.

[24] Vo-Dinh, T. (2019) Biomedical Photonics Handbook. CRC Press, Boca Raton, 889 p.

[25] Ran, R.-S. and Huang, T.-Z. (2006) The Inverses of Block Tridiagonal Matrices. *Applied Mathematics and Computation*, **179**, 243-247. https://doi.org/10.1016/j.amc.2005.11.098

[26] El-Mikkawy, M.E.A. (2004) On the Inverse of a General Tridiagonal Matrix. *Applied Mathematics and Computation*, **150**, 669-679. https://doi.org/10.1016/S0096-3003(03)00298-4

[27] Da Fonseca, C.M. and Petronilho, J. (2001) Explicit Inverses of Some Tridiagonal Matrices. *Linear Algebra and Its Applications*, **325**, 7-21. https://doi.org/10.1016/S0024-3795(00)00289-5

[28] Meurant, G. (1992) A Review on the Inverse of Symmetric Tridiagonal and Block Tridiagonal Matrices. *SIAM Journal on Matrix Analysis and Applications*, **13**, 707-728. https://doi.org/10.1137/0613045

[29] El-Mikkawy, M.E.A. (2003) A Note on a Three-Term Recurrence for a Tridiagonal Matrix. *Applied Mathematics and Computation*, **139**, 503-511. https://doi.org/10.1016/S0096-3003(02)00212-6

[30] Golub, G.H. and Van Loan, C.F. (2001) Matrix Computation. 3rd Edition, The Science Press, Beijing, 27.

[31] Ding, L.J. (1997) Numerical Computing Method. Science and Engineering University Press, Beijing, 26.

Appendix

Assuming an image I_{0n} of N rows and M columns, the function $J_c(I)$ in the discrete space becomes:

$$J_c(I) = \frac{1}{2} \cdot \sum_{i=1}^{N} \sum_{j=1}^{M} |\nabla I|^2 + \frac{\lambda}{2} \cdot \sum_{i=1}^{N} \sum_{j=1}^{M} |I - I_{0n}|^2 , \tag{A1}$$

where,

$$|\nabla I|^2 = \left(\frac{\partial I}{\partial x} \right)^2 + \left(\frac{\partial I}{\partial y} \right)^2 . \tag{A2}$$

If the partial derivatives are approximated using local differences:

$$\frac{\partial I}{\partial x} = I(x+1, y) - I(x, y) \text{ and } \frac{\partial I}{\partial y} = I(x, y+1) - I(x, y), \tag{A3}$$

and Equations (A3) are substituted in Equation (A2) and in the sequel Equation (A2) back to Equation (A1), the following objective function obtained:

$$\begin{aligned} J_c(I) = {} & \frac{1}{2} \cdot \sum_{x=1}^{N-1} \sum_{y=1}^{M-1} \left[\left(I(x+1, y) - I(x, y) \right)^2 + \left(I(x, y+1) - I(x, y) \right)^2 \right] \\ & + \frac{\lambda}{2} \cdot \sum_{x=1}^{N-1} \sum_{y=1}^{M-1} \left[I(x, y) - I_{0n}(x, y) \right]^2 \end{aligned} \tag{A4}$$

The minimum of Equation (A4) regarding to $I(.,.)$ can be derived in a close-form by differencing the second-order, positively defined function:

$$\begin{aligned} \frac{\partial J_c(I)}{\partial I} = {} & \frac{1}{2} \cdot \sum_{x=1}^{N-1} \sum_{y=1}^{M-1} \frac{\partial}{\partial I} \left[\left(I(x+1, y) - I(x, y) \right)^2 + \left(I(x, y+1) - I(x, y) \right)^2 \right] \\ & + \frac{\lambda}{2} \cdot \sum_{x=1}^{N-1} \sum_{y=1}^{M-1} \frac{\partial}{\partial I} \left[I(x, y) - I_{0n}(x, y) \right]^2 = 0, \\ & \Rightarrow (\lambda + 4) \cdot I(x, y) - I(x+1, y) - I(x, y+1) - I(x, y-1) - I(x-1, y) \\ & = \lambda \cdot I_{0n}(x, y), \ \forall x \in [1, N-1], \ y \in [1, M-1]. \end{aligned} \tag{A5}$$

The image I at the minimum of Equation (A1) is a linear system of $N \times M$ equations:

$$G \cdot \boldsymbol{I} = \lambda \cdot \boldsymbol{I}_{0n} . \tag{A6}$$

Matrix G is a sparse Hermitian matrix dependent only on parameter λ, $\boldsymbol{I}$ is the vector form of the unknown image $I(.,.)$ and $\boldsymbol{I}_{0n}$ is the vector form of the normalized image $I_{0n}(.,.)$. If matrix G is invertible, the brightness of the unknown image is derived from the solution of the system of linear equations:

$$\boldsymbol{I} = G^{-1} \cdot \left(\lambda \cdot \boldsymbol{I}_{0n} \right) . \tag{A7}$$

The block tridiagonal matrix G is invertible if the determinant is non-zero. From the optimization criterion, the new image $I(.,.)$ is a smooth image similar to the normalized $I_{0n}(.,.)$.

Inverse of a block tridiagonal matrix

Tridiagonal matrices often arise in many problems in applied mathematics, physics and engineering and inversion methods are an important topic as referred in [25]. Although a lot of work has been done in the past [26] [27] [28]

[29], the investigation on the inverses of the block tridiagonal matrices is poor. Recently, an inversion method of symmetric block tridiagonal matrices was presented in [28], based mainly on sub-matrix multiplications. With the contemporary development of high-powered computing, the fast inversion algorithm is more appropriate than the basic algorithms met in linear algebra [30]. The inversion of the matrix G is obtained using an iterative algorithm presented in [25].

In the following, the twisted block decompositions of the block tridiagonal matrices are presented, which leads to a computing sequence of the sub-matrix elements of the inverse tridiagonal matrix. These direct-form equations are derived according to the special structure of the matrix G.

The block tridiagonal matrices studied in [25] follows the form:

$$G=\begin{bmatrix} B_1 & -C_1 & O & O & O \\ -A_2 & B_2 & -C_2 & O & O \\ O & \bullet & \bullet & \bullet & O \\ O & O & -A_{k-1} & B_{k-1} & -C_{k-1} \\ O & O & O & -A_k & B_k \end{bmatrix}, \tag{A8}$$

where A_i, B_i and C_i are all matrix sequences of order m.

The LU decomposition of the tridiagonal matrices has been presented in [31]. In [25], the twisted block decompositions of matrix G are presented. According to the special structure of the twisted block decompositions, an iterative formula of computing the column block elements of the inverse matrix can be obtained. The twisted block LU decomposition of G can be obtained by Equation (A9) as stated and proved in the following lemma.

Lemma **1.** Let G be a block tridiagonal matrix in the form of Equation (A8), then for each j ($j=1,2,\cdots,k$), a twisted block LU decomposition of G can be formulated also as:

$$G=\tilde{L}_j\tilde{U}_j=\begin{bmatrix} H_1 & O & O & O & O & O & O \\ -L_2 & H_2 & O & O & O & O & O \\ O & \bullet & \bullet & O & O & O & O \\ O & O & -L_j & H_j & -L_{j+1} & O & O \\ O & O & O & O & \bullet & \bullet & O \\ O & O & O & O & O & H_{k-1} & -L_k \\ O & O & O & O & O & O & H_k \end{bmatrix} \cdot\begin{bmatrix} I & -U_1 & O & O & O & O & O \\ O & \bullet & O & O & O & O & O \\ O & \bullet & I & -U_{j-1} & O & O & O \\ O & O & O & I & O & O & O \\ O & O & O & U_j & I & \bullet & O \\ O & O & O & O & O & \bullet & O \\ O & O & O & O & O & -U_{k-1} & I \end{bmatrix}, \tag{A9}$$

where L_i, H_i, and U_i are matrix sequences estimated by the iterative Algorithm 1.

Thus, Equation (A6) can be rewritten in a matrix form as:

$$\begin{bmatrix} \lambda+4 & -1 & 0 & \bullet & \bullet & -1 & \bullet & 0 \\ -1 & \lambda+4 & -1 & 0 & \bullet & \bullet & -1 & 0 \\ 0 & -1 & \lambda+4 & -1 & \bullet & \bullet & \bullet & -1 \\ \bullet & 0 & -1 & \bullet & \bullet & \bullet & \bullet & \bullet \\ -1 & \bullet & \bullet & \bullet & \bullet & \bullet & -1 & 0 \\ \bullet & -1 & \bullet & \bullet & \bullet & -1 & \lambda+4 & -1 \\ 0 & 0 & -1 & 0 & \bullet & 0 & -1 & \lambda+4 \end{bmatrix} \cdot \begin{bmatrix} I(1,1) \\ I(1,2) \\ \bullet \\ I(1,M) \\ I(2,1) \\ I(2,2) \\ \bullet \\ I(2,M) \\ \bullet \\ I(N,1) \\ \bullet \\ I(N,M) \end{bmatrix} = \lambda \cdot \begin{bmatrix} I_{0n}(1,1) \\ I_{0n}(1,2) \\ \bullet \\ I_{0n}(1,M) \\ I_{0n}(2,1) \\ I_{0n}(2,2) \\ \bullet \\ I_{0n}(2,M) \\ \bullet \\ I_{0n}(N,1) \\ \bullet \\ I_{0n}(N,M) \end{bmatrix}, \quad \text{(A10)}$$

and matrix G can be minimized using Algorithm 1.

In the case of the inversion of matrix G, the corresponding block matrices are simplified to $B_1 = B_2 = \cdots = B_k = B = \begin{bmatrix} \lambda+4 & -1 & 0 \\ -1 & \lambda+4 & -1 \\ 0 & -1 & \lambda+4 \end{bmatrix}$,

$A_1 = A_2 = \cdots = A_k = A = I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and

$C_1 = C_2 = \cdots = C_k = C = I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

In this case, the iterative LU decomposition algorithm for $k = 3$ (the value of k is selected only for presentation simplicity) becomes:

***j* = 1** ($i = 1, 2, 3$):

$$L_1 = L_2 = L_3 = I$$

$$U_1 = \left(B - B^{-1}\right)^{-1}, \quad U_2 = B^{-1}$$

$$H_1 = B - \left(B - B^{-1}\right)^{-1}, \quad H_2 = B - B^{-1}, \quad H_3 = B$$

$$X_{11} = H_1^{-1}, \quad X_{21} = U_1 * X_{11}, \quad X_{31} = U_2 * X_{21}$$

***j* = 2** ($i = 1, 2, 3$):

$$L_1 = L_2 = L_3 = I$$

$$U_1 = B^{-1}, \quad U_2 = B^{-1}$$

$$H_1 = B, \quad H_2 = B - 2 * B^{-1}, \quad H_3 = B$$

$$X_{12} = U_1 * X_{22}, \quad X_{22} = H_2^{-1}, \quad X_{32} = U_2 * X_{22}$$

***j* = 3** ($i = 1, 2, 3$):

$$L_1 = L_2 = L_3 = I$$

$$U_1 = B^{-1},\quad U_2 = \left(B - B^{-1}\right)^{-1}$$

$$H_1 = B,\quad H_2 = B - B^{-1},\quad H_3 = B - \left(B - B^{-1}\right)^{-1}$$

$$X_{13} = U_1 * X_{23},\quad X_{23} = U_2 * X_{33},\quad X_{33} = H_3^{-1}$$

and the inverse of matrix G is:

$$X = G^{-1} = \begin{bmatrix} X_{11} & X_{12} & X_{12} \\ X_{21} & X_{22} & X_{23} \\ X_{31} & X_{32} & X_{33} \end{bmatrix}$$

$$= \begin{bmatrix} \left(B - K_1\right)^{-1} & B^{-1}\cdot K_2 & B^{-1}\cdot K_1\cdot\left(B - K_1\right)^{-1} \\ K_1\cdot\left(B - K_1\right)^{-1} & K_2 & K_1\cdot\left(B - K_1\right)^{-1} \\ B^{-1}\cdot K_1\cdot\left(B - K_1\right)^{-1} & B^{-1}\cdot K_2 & \left(B - K_1\right)^{-1} \end{bmatrix}, \tag{A11}$$

where $K_1 = \left(B - B^{-1}\right)^{-1}$ and $K_2 = \left(B - 2\cdot B^{-1}\right)^{-1}$.

***Algorithm* 1.** LU decomposition of tridiagonal matrices.

```
$H_1 = B_1$
$H_k = B_k$
for j = 1,k
    for i = 1,k
        if (i<j)
            $L_i = A_i$
            if (i>1)
                $H_i = B_i - L_i * U_{i-1}$
            end
            $U_i = H_i^{-1} * A_i$ (*)
        end
        if (i>j)
            $L_i = C_{i-1}$
            if (i> = k-1) and (i< = j+1)
                $U_i = H_{i+1}^{-1} * A_{i+1}$
                $H_i = B_i - L_{i+1} * U_i$
            end
        end
        if (i = j)
            $L_i = A_i$
            $U_i = H_{i+1}^{-1} * A_{i+1}$
            $H_i = B_i - L_i * U_{i-1} - L_{i+1} * U_i$
        end
    end
end
```

The proof is given in [25]. In the same paper, the expression (*) has been incorrectly printed as $U_i = H_i^{-1} * B_i$. After the twisted LU decomposition, the block tridiagonal matrices of the inverse of matrix G can be derived using an iterative algorithm as stated in the following theorem.

***Theorem* 1.** Let G be a block tridiagonal matrix in the form of Equation A(10) and let $G^{-1} = X = \left(X_{ij}\right)$, where $X_{ij}\left(i, j = 1, 2, \cdots, k\right)$ are all square matrices of order *m*, then the *jth* ($j = 1, 2, \cdots, k$) column block elements of X can be derived by the following matrix iterative formulas:

```
for j = 1,k
    $X_{jj} = H_j^{-1}$
    for i = 1,k
        if (i<j)
            $X_{ij} = U_i*X_{i+1j}$
        end
        if (i>j)
            $X_{ij} = U_{i-1}*X_{i-1j}$
        end
    end
end
```

The proof is also given in [25].