

Recent trends in Medical Image Processing Editorial (Preface) for a special issue of Computer Science Journal of Moldova

Hariton Costin

It is said that images bear the greatest density of natural information of all ways of human communication, and biomedical images do not make any exception to this assertion, at least when dealing with morphologic information. The recent rapid advances in medical imaging and automated image analysis will continue and allow us to make significant advances in our understanding of life and disease processes, and our ability to deliver quality healthcare.

Medical imaging and image processing domains mainly manage and process missing, ambiguous, inconsistent, complementary, contradictory, redundant and distorted data, and information has a strong structural character. The processes of human and artificial understanding of any image involve the matching of features extracted from the image with pre-stored models. From the information technology point of view the production of a high-level symbolic model requires the representation of knowledge about the objects to be modeled, their relationships, and how and when to use the information stored within the model.

In general, a distinction is made between (bio)medical imaging and image processing technologies, even if between these fields of knowledge a strong interrelation may be established.

Biomedical imaging concentrates on the capture and display of images for both diagnostic and therapeutic purposes, and modern imaging technology is 100% digital. Snapshots of in vivo physiology and physiological processes can be garnered through advanced sensors and computer technology. Biomedical imaging technologies utilize either x-rays

(CT scans), sound (ultrasound), magnetism (magnetic resonance imaging – MRI), radioactive pharmaceuticals (nuclear medicine: SPECT, PET) or light (endoscopy, OCT) to assess the current condition of an organ or tissue and can monitor a patient over time for diagnostic and treatment evaluation. From the information type point of view, medical imaging can be structural (or morphologic, e.g. CT, MRI, OCT) or functional (PET, SPECT).

Biomedical image processing is similar in concept to biomedical signal processing in multiple dimensions (2D, 3D). It includes the enhancement, analysis and display of images captured via the above mentioned x-ray, ultrasound, MRI, nuclear medicine and optical imaging technologies. Related procedures, like image reconstruction and modeling techniques allow quick processing of 2D signals to create 3D images. Image processing software helps to automatically identify and analyze what might not be apparent to the human eye, even of an expert. Computerized algorithms can provide temporal and spatial analysis to detect patterns and characteristics indicative of tumors and other ailments. Depending on the imaging technique and what diagnosis is being considered, image processing and analysis can be used to determine, for instance, the diameter, volume and vasculature of a tumor or organ, flow parameters of blood or other fluids and microscopic changes that have yet to raise any otherwise discernible flags.

Nowadays some key components of clinical activity are image-guided therapy (IGT) and image-guided surgery (IGS), where localization, targeting, monitoring, and control are main issues. Specifically, in medical imaging and medical image processing we have four key problems: (1) Image Segmentation – dealing with (semi)automated methods that lead to creating patient-specific models of relevant anatomy from images; (2) Image Registration – automated methods that align multiple data sets, eventually coming from different imaging modalities, with each other; (3) Visualization – the technological environment in which image-guided procedures can be displayed; (4) Simulation – software that can be used to rehearse and plan procedures, evaluate access strategies, and simulate planned treatments.

In fact, all traditional and advanced techniques of image processing

and computational vision, analysis and understanding may be used to process medical images, in order to extract useful information for diagnosis and treatment.

A special approach, including that of the editor of this journal issue, directs to the use of Artificial Intelligence (AI), which has proved to yield promising results in medical image processing and analysis. The structural character of information may successfully be approached by using methods of AI such as Knowledge Based Systems, Expert Systems, Decision Support Systems, Neural Networks, Fuzzy Logic and Systems, Neuro-Fuzzy Systems, Evolutionary and Genetic (or bio-inspired) Algorithms, Data Mining, Knowledge Discovery, Semantic Nets, Symbolic Calculus for knowledge representation, etc. The *data fusion* methods successfully solve the aggregation of numerical and linguistic information, and are able to cope with ambiguous, uncertain, conflicting, complementary, imprecise and redundant information, like that occurring in biomedical imaging domain, in order to provide a more accurate and less uncertain interpretation.

One of the main characteristic of the Medical Image Processing domain is its inter- and multidisciplinary nature. In fact, in this field, methodologies of several fundamental and applicative sciences, such as Informatics, Mathematics, Physics, Statistics, Computer science, Medicine, Engineering, Psychology, Artificial Intelligence, (Bio)Mechanics are regularly used. Besides this characteristic, one of the main rationale that promotes the continuous effort being made in this area of human knowledge is the huge number of useful applications in the medical area, some of them being illustrated hereinafter.

This special issue of Computer Science Journal of Moldova contains six invited papers that illustrate new trends and outcomes in medical image processing. It gathers together prominent researches that align to the state-of-the-art of Computational Vision and Medical Image Processing, contributing to the development of both these knowledge areas, and of medical research and clinical activity.

Probably is somehow justified that two of the invited articles deal with retinal images processing. The first one, "Detection of Blood

Vessels in Retinal Fundus Images”, of Oloumi, Dhara, Rangayyan, and Mukhopadhyay approaches automatic detection of blood vessels in retinal fundus images, in order to perform computer-aided diagnosis of some pathologies of the eye such as diabetic retinopathy (DR), retinopathy of prematurity, and maculopathy. The vessels detection techniques include a mix of multiscale and multifeature methods, like multiscale vesselness measures, Gabor filters, line operators, and matched filters. An adaptive threshold selection method is crucial for precise detection of retinal blood vessels. The accuracy of detection is improved by an original postprocessing technique for removal of false-positive pixels around the optic nerve head. The results of detection of blood vessels, evaluated in terms of the area under the receiver operating characteristic curve of up to 0.961, were obtained using the 20 test images of the DRIVE database (which is considered as containing low-resolution retinal images). These results have double meaning: on one hand they outperform other approaches of the chosen topic, and on the other hand they show that a single-scale Gabor filter is capable of detecting blood vessels with accuracy not much different from the best value obtained by means of multifeature and multiscale methods. In this way the authors prove once again the famous Latin saying “non multa, sed multum”.

The second article, “Optic Disc Identification Methods for Retinal Images”, written by Rotaru, Bejinariu, Niță, Luca, and Lazăr, proposes some original methods to identify and model the optic disc in colour retinal images, as well as the blood vessels network, to evaluate different retina diseases such as diabetic macular edema, glaucoma, etc. The paper represents an extension of some early researches of the same authors, in which they heuristically apply certain traditional image processing methods (low-pass filtering, Maximum Difference Method, texture analysis, voting algorithms, morphologic filtering, Otsu binarization) on 40 clinically validated retinal images of high resolution (2592×1728 pixels), 386 images of 720×576 resolution, and more than 100 images from STARE database. The obtained results in terms of accuracy are comparable with the best outcomes in the literature, the proposed techniques are implemented as a Windows application writ-

ten in C++ using Microsoft Visual Studio, and for image manipulation and some processing functions the OpenCV library was used.

The next invited article, “Characterization and Pattern Recognition of Color Images of Dermatological Ulcers: a Pilot Study”, written by L.C. Pereyra, S.M. Pereira, Souza, Frade, Rangayyan, and Azevedo-Marques, approaches content-based image retrieval (CBIR) and computer-aided diagnosis (CAD) applied in dermatological ulcers detection and analysis, which is a very difficult task of color image processing and of tissue composition analysis, respectively. Unsupervised automatic segmentation was performed by using Gaussian mixture modeling, and its performance was assessed by computing the Jacard coefficient between the automatically and manually segmented images. A retrieval engine was implemented using the k -nearest-neighbor method, and classification was made by means of a logistic regression. The performance of CBIR was measured in terms of precision of retrieval, with average values of up to 0.617 obtained with the Chebyshev distance, and the average value of correctly classified instances divided by the total number of instances was 0.738. Results were obtained on a database containing 172 dermatologic images with high geometric and intensity resolution, obtained in a clinical environment and annotated by an expert. Even if the obtained segmentation accuracy is not very high, from clinical and educational utility points of view “objective analysis of color images of skin ulcers using the proposed methods might overcome some of the limitations of visual analysis and lead to the development of improved protocols for the treatment and monitoring of chronic dermatological lesions.”

The last three articles are dedicated to some image processing topics useful in research and clinical practice, but which do not approach image segmentation or classification. They refer to image reconstruction, registration and human genome sequencing, respectively.

“A platform for Image Reconstruction in X-ray Imaging: Medical Applications using CBCT and DTS algorithms”, by Kamarianakis, Buliev, Pallikarakis, presents the architecture of a software platform for the purpose of testing and evaluation of reconstruction algorithms in X-ray imaging. The main elements of the platform are classes, re-

lated together in a logical hierarchy. Real world objects can be defined and implemented as instances of corresponding classes. Different image processing routines (e.g. 3D transformations, loading, saving, filtering of images, etc.) have been incorporated in the software tool as class methods, too. The platform is viewed as a basic tool for future investigations of new reconstruction methods in combination with various scanning configurations. The current tests on Reconstruction Techniques Class Library (RTCL) and the Platform for Image Reconstruction in X-ray Imaging (PIRXI) prove the accuracy and flexibility of this new approach for image reconstruction research and algorithms implementation.

Bejinariu, Costin, Rotaru, Luca, Niță, and Lazăr authored the article “Parallel Processing and Bio-inspired Computing for Biomedical Image Registration”, that deals with image transformations aiming at overlaying one or more image sources to a given model by maximizing a similarity measure. Some bio-inspired metaheuristic optimization algorithms, such as Bacterial Foraging Optimization Algorithm (BFOA), Genetic Algorithms (GAs) and Clonal Selection Algorithm (CSA), are compared in terms of registration accuracy and execution time for area-based and feature-based image registration. Normalized correlation (NCC) and normalized mutual information (NMI) are used as similarity measures. Implementation was made on many images from a publicly available database, mainly using MRI brain images with 256×256 pixels and 8 bits/pixels resolutions, without and with “salt & pepper” noise, respectively. In general, BFOA and GAs yielded comparable results in terms of registration accuracy, GAs performed about three times faster than BFOA, and CSA is too slow for feature-based registration and also with lower registration precision. Even the feature-based image registration performs faster, its use for multimodal images is limited by the procedure’s capability to find common and stable features in the images to be registered.

Last but not least, Voina, Pop, Vaida wrote the article “A New Algorithm for Localized Motif Detection in Long DNA Sequences”, that comes from bioinformatics research domain and approaches human genome sequencing, i.e. the identification of the DNA segments

that have a certain biological significance. The study presents a new algorithm optimized for finding motifs in long DNA sequences and some experiments done to evaluate the performance of the proposed algorithm in comparison with other motifs finding algorithms are described. Some optimizations are introduced, that increase detection accuracy and lower the execution time. Thus, the proposed algorithm proved to have a clear advantage among other similar algorithms due to the detection accuracy of the motifs in long DNA sequences, such as those found in the human genome.

In conclusion we can say that the fundamental, engineering and life sciences are all contributing to a remarkable synergy of efforts to achieve dynamic, quantitative (structural or functional) imaging of the body using minimally invasive, non-invasive or even virtual methods. The structural and functional relationships between the cells, tissues, organs and organ systems of the body are being advanced by molecular imaging, and laboratory imaging techniques. With continuing evolutionary progress in biomedical imaging, visualization and analysis, we can fully expect to benefit from new knowledge about life and disease processes, and from new and efficient methods of diagnosis therapy and prevention.

Iași, 30-th of June, 2014

Prof. Hariton Costin

Brief biography

Prof. dr. eng. Hariton Costin, B.Sc. in Electronics and Telecommunications, Ph.D. in Applied Informatics, is full professor at the University of Medicine and Pharmacy, Faculty of Medical Bioengineering, Iași, Romania, (www.umfiiasi.ro ; www.bioinginerie.ro). Also, he is senior researcher at the Romanian Academy – Iași Branch, Institute of Computer Science, within the Image Processing and Computer Vision Lab, (http://iit.academiaromana-is.ro/personal/h_costin.html). Here his studies are in image processing and analysis by using Artificial Intelligence methods and data fusion.



Competence areas include: medical electronics and instrumentation, biosignal and image processing and analysis, artificial intelligence (soft-computing, bio-inspired algorithms, expert systems), hybrid systems, HCI (human-computer interfaces), assistive technologies, telemedicine/e-health.

Scientific activity can be resumed by about 145 published papers in peer-reviewed journals and conference proceedings, 9 books, 4 book chapters in foreign publishing houses, 3 patents, 2 national awards.

Research activity: 54 research reports; technical manager (within FP5/INES 2001-32316 project) for a telemedicine application (www.euroines.com; „Medcare” project); responsible for the first Romanian pilot telemedical centre in Iași; director for 9 national funded projects in bioengineering and (biomedical) image processing/analysis. He has served as the program committee member of various international conferences and reviewer for various international journals and conferences. Prof. Costin was invited as postdoc researcher at the University of Science and Technology of Lille (France, 2002, in medical imaging), the University of Applied Sciences Jena (Germany, 2013) and had invited talks at international conferences. Prof. Costin is a member of the I.E.E.E./Engineering in Medicine & Biology Society (EMBS) and of other 5 scientific societies.

Detection of Blood Vessels in Retinal Fundus Images*

Invited Article

Faraz Oloumi, Ashis K. Dhara,
Rangaraj M. Rangayyan[®], Sudipta Mukhopadhyay

Abstract

Detection of blood vessels in retinal fundus images is an important initial step in the development of systems for computer-aided diagnosis of pathologies of the eye. In this study, we perform multifeature analysis for the detection of blood vessels in retinal fundus images. The vessel detection techniques implemented include multiscale vesselness measures, Gabor filters, line operators, and matched filters. The selection of an appropriate threshold is crucial for accurate detection of retinal blood vessels. We evaluate an adaptive threshold selection method along with several others for this purpose. We also propose a postprocessing technique for removal of false-positive pixels around the optic nerve head. Values of the area under the receiver operating characteristic curve of up to 0.961 were obtained using the 20 test images of the DRIVE database.

Keywords: Gabor filter, line operator, matched filter, multiscale analysis, retinal fundus image, vessel detection, vesselness measure.

©2014 by Faraz Oloumi, Ashis K. Dhara, Rangaraj M. Rangayyan, and Sudipta Mukhopadhyay

*This work was supported by grants from the Natural Sciences and Engineering Research Council (NSERC) of Canada, the Shastri Indo-Canadian Institute, and the Indian Institute of Technology Kharagpur, India. This article is a revised and expanded version of the conference publication cited as Dhara et al. [1]. [®]Address all correspondence to R. M. Rangayyan, ranga@ucalgary.ca

1 Introduction

Retinal fundus images are used by ophthalmologists for the diagnosis of several disorders, such as diabetic retinopathy (DR), retinopathy of prematurity, and maculopathy [2–4]. Detection of blood vessels is an important initial step in the development of computer-aided diagnostic (CAD) systems and analysis of retinal fundus images. It is possible to detect other anatomical landmarks such as the optic nerve head (ONH) and the macula in the retina with respect to the vascular architecture. The location and certain characteristics of such landmarks can help in the derivation of features for the detection of abnormalities. A variety of methods have been proposed for the detection of blood vessels; some of these methods are reviewed in the following paragraphs.

Chaudhuri et al. [5] proposed an algorithm based on two-dimensional (2D) matched filters for vessel detection. Their method is based on three assumptions: (i) the intensity profile of a vessel can be approximated by a Gaussian function, (ii) vessels can be approximated by piecewise linear segments, and (iii) the width of vessels is relatively constant. Detection of blood vessels was performed by convolving the given image with the matched filter rotated in several orientations. The maximum filter response over all orientations was assigned to each pixel.

Staal et al. [6] extracted the ridges in the images which roughly coincide with the vessel centerlines. In the next step, image primitives were obtained by grouping image ridges into sets that model straight-line elements, which were used to partition the image by assigning each pixel to the closest primitive set. Feature vectors were then computed for every pixel using the characteristics of the partitions and their line elements. The features were used for classification using a k -nearest-neighbor classifier. Staal et al. achieved an area under the receiver operating characteristic (ROC) curve of $A_z = 0.9520$ with 20 images of the test set of the DRIVE database [7].

Soares et al. [8] applied complex Gabor filters for feature extraction and supervised classification for the detection of blood vessels in retinal fundus images. In this method, the magnitude outputs at several scales

obtained from 2D complex Gabor filters were assigned to each pixel as a feature vector. Then, a Bayesian classifier was applied for classification of the results into vessel or nonvessel pixels. Soares et al. reported $A_z = 0.9614$ for the 20 test images of the DRIVE database.

Blood vessels can be considered as dark elongated curvilinear structures of different width and orientation on a brighter background. Several types of vesselness measures have been developed for the detection of blood vessels based on the properties of the eigenvalues of the Hessian matrix computed at each pixel. Because blood vessels are of varying width, different scales are used to calculate the eigenvalues and the maximum response at each pixel over all scales is used for further analysis. Frangi et al. [9] and Salem et al. [10] proposed different vesselness measures to highlight vessel-like structures. Wu et al. [11] applied the vesselness measure of Frangi et al. to the 40 training and testing images of the DRIVE database and reported $A_z = 0.9485$. Salem et al. [10] reported $A_z = 0.9450$ using 20 images of the STARE database [12].

Lupaşcu et al. [13] performed multifeature analysis using previously proposed features [6, 8, 9, 14, 17], combined with new features that represent information about the local intensity, the structure of vessels, spatial properties, and the geometry of the vessels at different scales of length. They used a feature vector containing a total of 41 features obtained at different scales to train a classifier, which was then applied to the test set. They reported $A_z = 0.9561$ using the 20 test images of the DRIVE database.

Rangayyan et al. [15] performed multiscale analysis for the detection of blood vessels using Gabor filters and classified pixels using multilayer perceptron (MLP) neural networks and reported A_z of 0.9597 with the test set of the DRIVE database. Oloumi [16] used multiscale Gabor filter magnitude responses, coherence, and the inverted green channel as features to train an MLP and achieved an A_z value of 0.9611 using the test set of the DRIVE database.

Other available methods in the literature that do not employ a filtering technique for the detection of the blood vessels include, but are not limited to, segmentation using multiconcavity modeling [18]; fractal analysis [19]; mathematical morphology and curvature evaluation [20];

and geometrical models and analysis of topological properties of the blood vessels [21].

In the present work, we perform vessel segmentation by multifeature analysis, using multiscale Gabor filters as proposed by Rangayyan et al. [15], multiscale vesselness measures as proposed by Frangi et al. [9] and Salem et al. [10], matched filters as proposed by Chaudhuri et al. [5], line operators [22], and a gamma-corrected version of the inverted green channel. Thresholding and binarization of the result of vessel detection is a crucial step for further analysis of the characteristics of blood vessels such as thickness and tortuosity [23]; we propose an adaptive thresholding technique by analyzing the intensity values of the boundary pixels of retinal blood vessels and compare the results against several automated thresholding methods. Most of the reported methods for the detection of blood vessels cause false-positive (FP) pixels associated with the boundary of the ONH. We propose a postprocessing technique for removal of FP pixels around the ONH.

2 Database of Retinal Images

In this work, retinal fundus images from the DRIVE database were used to assess the performance of the methods. The images of the DRIVE database [6, 7] were acquired during a screening program for DR in the Netherlands and show signs of mild DR. The images have a size 565×584 pixels and a field of view (FOV) of 45° . The DRIVE images are considered to be low-resolution retinal images; the images have an approximate spatial resolution of $20 \mu\text{m}$ per pixel. The DRIVE database consists of 40 images, which are labeled in two sets of 20 images for training and testing. A manually segmented image (ground-truth) of the vasculature is available for each image in the DRIVE database. Figure 1 shows the original color image 12 of the DRIVE database and its ground-truth image.

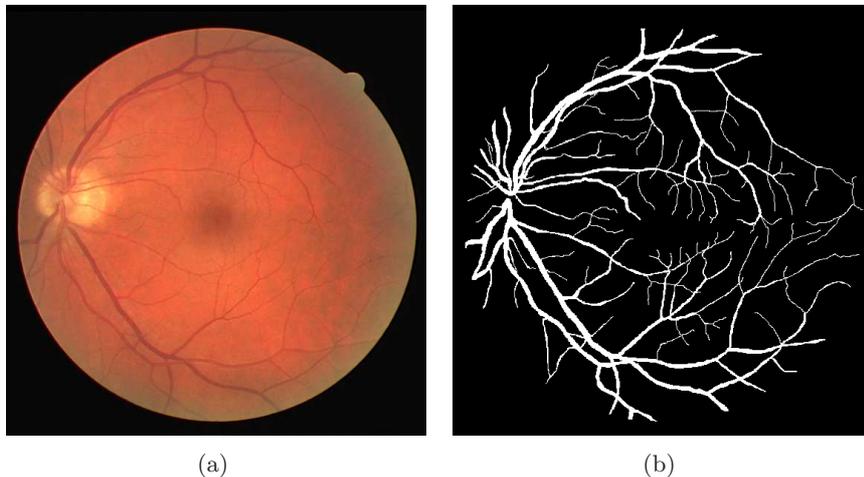


Figure 1. (a) Original color image 12 of the DRIVE database. (b) Ground-truth of the image in part (a), as provided in the database.

3 Detection of Retinal Blood Vessels

In the present study, we review and implement several methods for the detection of blood vessels and investigate their combined application for multifeature analysis.

3.1 Vesselness Measures

Frangi et al. [9] defined a vesselness measure to detect pixels having vessel-like structures based on the properties of the eigenvalues of the Hessian matrix. A numerical estimate of the Hessian matrix, H , at each pixel of the given image, $L(x, y)$, is obtained as

$$H = \begin{bmatrix} \frac{\partial^2 L}{\partial x^2} & \frac{\partial^2 L}{\partial x \partial y} \\ \frac{\partial^2 L}{\partial y \partial x} & \frac{\partial^2 L}{\partial y^2} \end{bmatrix}. \quad (1)$$

The entries of H can be obtained at multiple scales by convolving the

image $L(x, y)$ with the Gaussian kernel $G(x, y; \sigma)$ of different scales σ , defined as

$$G(x, y; \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right). \quad (2)$$

The width of retinal blood vessels varies from 50 μm to 200 μm in retinal fundus images, which translates to the range of about 2 to 10 pixels, given a spatial resolution of 20 μm for the DRIVE images. Gaussian kernels can be used to generate a suitable scale space with an amplitude range of σ related to the range of vessel width. Multiscale derivatives of the image $L(x, y)$ can be obtained by linear convolution of the image with the scale-normalized derivatives of the Gaussian kernel as $\frac{\partial^2 L}{\partial x^2} = L(x, y) * \sigma^2 G_{xx} = L_{xx}$, $\frac{\partial^2 L}{\partial x \partial y} = \frac{\partial^2 L}{\partial y \partial x} = L(x, y) * \sigma^2 G_{xy} = L_{xy} = L_{yx}$, and $\frac{\partial^2 L}{\partial y^2} = L(x, y) * \sigma^2 G_{yy} = L_{yy}$. Here G_{xx} , G_{xy} , and G_{yy} are the second derivatives of the Gaussian kernel G , and the symbol ‘*’ represents the 2D convolution operation.

The Hessian matrix is symmetrical with real eigenvalues. The signs and ratios of the eigenvalues can be used as signatures of a local structure. Let λ_1 and λ_2 represent the eigenvalues of the Hessian matrix, with the condition $|\lambda_2| \geq |\lambda_1|$. The larger eigenvalue, λ_2 , corresponds to the maximum principal curvature at the location (x, y) . A larger value of λ_2 compared to λ_1 represents a vessel-like structure. The eigenvalues and eigenvectors of the Hessian matrix can be computed by solving the following equation:

$$\begin{vmatrix} L_{xx} - \lambda & L_{xy} \\ L_{yx} & L_{yy} - \lambda \end{vmatrix} = 0, \quad (3)$$

where λ represents the two eigenvalues λ_1 and λ_2 . The eigenvalues λ_1 and λ_2 can be obtained as

$$\lambda_1 = \frac{L_{xx} + L_{yy} - \alpha}{2}, \quad (4)$$

and

$$\lambda_2 = \frac{L_{xx} + L_{yy} + \alpha}{2}, \quad (5)$$

where $\alpha = \sqrt{(L_{xx} - L_{yy})^2 + 4L_{xy}^2}$. Based on the property of the eigenvalues of the Hessian matrix, Frangi et al. [9] defined a vesselness measure to highlight pixels belonging to vessel-like structures as

$$V_F = \begin{cases} \exp\left(-\frac{R_\beta^2}{2\beta^2}\right) \left[1 - \exp\left(-\frac{S^2}{2\gamma^2}\right)\right] & \text{if } \lambda_1, \lambda_2 < 0, \\ 0 & \text{otherwise,} \end{cases} \quad (6)$$

where $R_\beta = \frac{\lambda_1}{\lambda_2}$, $S = \sqrt{\lambda_1^2 + \lambda_2^2}$ is the Frobenius norm of the Hessian matrix, $\beta = 0.5$ (as used by Frangi et al. [9]), and γ is equal to one-half of the maximum of all of the Frobenius norms computed for the whole image. The Frobenius norm is expected to be low in background areas where no vessels are present and the eigenvalues are low, because the magnitude of the derivatives of the intensities will be small. On the other hand, in regions with high contrast as compared to the background, the Frobenius norm will become larger, because at least one of the eigenvalues will be large.

The vesselness measure proposed by Salem et al. [10] uses the eigenvalues of the Hessian matrix to detect the orientation of blood vessels. Let \vec{e}_1 and \vec{e}_2 be the eigenvectors corresponding to the eigenvalues λ_1 and λ_2 , respectively, and let θ_1 and θ_2 be the angles of the eigenvectors with respect to the positive x -axis. The orientations of the eigenvectors corresponding to the larger and smaller eigenvalues for every fifth pixel are shown in Figure 2. It can be noted from Figure 2 that the variation of the orientation of the eigenvectors corresponding to the smaller eigenvalues is smaller inside the blood vessels as compared to that outside the blood vessels. The eigenvectors corresponding to the smaller eigenvalues are mainly oriented along the blood vessels; hence, the angle θ_1 is used to analyze the orientation of blood vessels. The orientation of the eigenvector \vec{e}_1 can be represented as

$$\theta_1 = \arctan\left(-\frac{2L_{xy}}{L_{yy} - L_{xx} + \alpha}\right). \quad (7)$$

Detection of blood vessels can be accomplished by assuming that

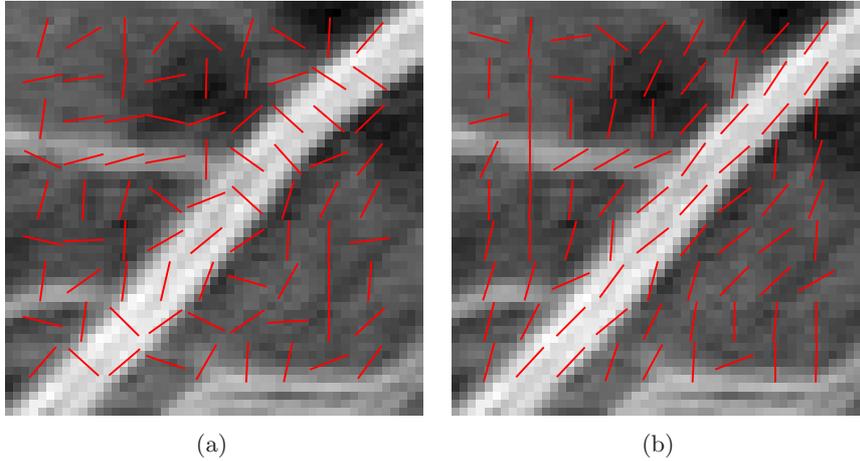


Figure 2. Orientation of the eigenvectors corresponding to (a) the larger eigenvalue and (b) the smaller eigenvalue at each pixel for a part of a retinal fundus image with parts of blood vessels. Straight lines corresponding to the eigenvectors are shown for every fifth pixel. The size of the image is 50×50 pixels.

the maximum value of λ_2 (λ_{\max}) over several scales of σ is at the center of the vessel. Salem et al. [10] defined a vesselness measure as

$$V_S = \frac{\lambda_{\max}}{\theta_{\text{std}} + 1}, \quad (8)$$

where θ_{std} is the standard deviation (STD) of θ_1 over all scales used for the pixel under consideration. The larger the value of V_S for a pixel, the higher the probability that the pixel belongs to a vessel.

In this work, the range of scales $\sigma = [1, 6]$ with steps of 0.05 was determined to be the most suitable range for the vesselness measures of Frangi et al. and Salem et al. using the training set of the DRIVE database, and was used for subsequent analysis. Note that the two vesselness measures implemented in this work perform multiscale analysis by taking the maximum intensity value among all the available scales of σ . The implementation of the method of Frangi et al. used in this

work was provided by Dirk-Jan Kroon of University of Twente [36].

Figure 3 shows the magnitude response images of the result of applying the vesselness measures of Frangi et al. and Salem et al. to the image in Figure 1 (a).

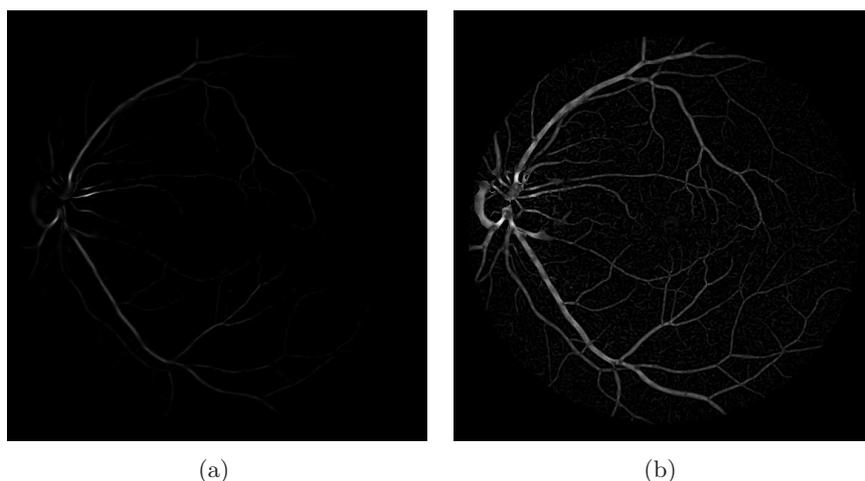


Figure 3. Magnitude response images of the result of filtering the image in Figure 1 (a) obtained using (a) vesselness measure of Frangi et al. and (b) vesselness measure of Salem et al. Note that the result of the method of Frangi et al. provides lower intensity values as compared to the method of Salem et al. and the detected vessels may not be clearly visible in the result.

3.2 Gabor Filters

Rangayyan et al. [15] applied multiscale Gabor filters for the detection of blood vessels by considering the fact that blood vessels are elongated, piecewise-linear, or curvilinear structures with a preferred orientation. Gabor filters are sinusoidally modulated Gaussian functions that are suitable for the analysis of oriented structures because they provide optimal localization in both the frequency and space domains. The real Gabor filter kernel oriented at the angle $\theta = -\pi/2$ can be represented

as [15]

$$g(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp \left[-\frac{1}{2} \left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} \right) \right] \cos(2\pi f_o x). \quad (9)$$

In this equation, the frequency of the modulating sinusoid is given by f_o , and σ_x and σ_y are the STD values in the x and y directions. For simplicity of design, a variable τ is used to represent the average thickness of the vessels to be detected. The value of σ_x is defined based on τ as $\sigma_x = \frac{\tau}{2\sqrt{2\ln 2}}$ and $\sigma_y = l\sigma_x$, where l represents the elongation of blood vessels. A bank of K Gabor filters may be obtained by rotating the main Gabor filter kernel given in Equation 9 over the range $[-\pi/2, \pi/2]$. For a given pixel, the maximum output value over all K filters is saved as the Gabor magnitude response at that particular pixel; the corresponding angle is saved as the Gabor angle response.

Values of $\tau = 8$ pixels, $l = 2.9$, and $K = 180$ were determined to provide the best single-scale results, as determined using the training set of the DRIVE database. Values of $\tau = 4, 8$, and 12 were used to perform multiscale and multifeature analysis as described in Section 3.5. Figure 4 shows the magnitude and angle responses of Gabor filters with $\tau = 8$ pixels, $l = 2.9$, and $K = 180$ as obtained for the image in Figure 1 (a). It is seen that the magnitude response is high at pixels belonging to vessels and that the angle response agrees well with the angle of the vessel at the corresponding pixel.

3.3 Line Operators

Line operators were proposed by Dixon and Taylor [24] and used by Zwiggelaar et al. [25] for the detection of linear structures in mammograms. The main line operator kernel detects horizontal lines. Assume that $N(x, y)$ is the average gray-level of M pixels along a horizontal line centered at (x, y) . Next, assume that $S(x, y)$ is the average gray-level of pixels in a square of width M pixels that is horizontally aligned and centered at (x, y) . The main line operator kernel is defined as $L(x, y) = N(x, y) - S(x, y)$. Detecting lines with various orientations is achieved by rotating the main kernel. Let $L_k(x, y)$ be the line opera-

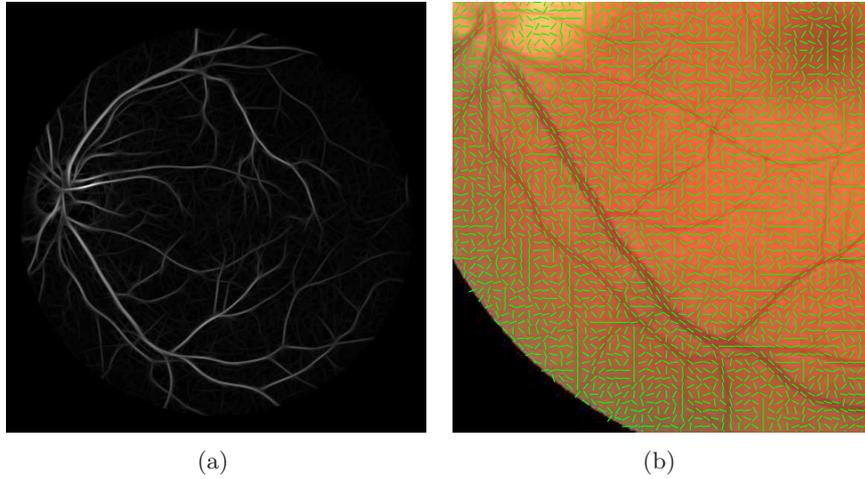


Figure 4. (a) Gabor magnitude and (b) angle responses of the image in Figure 1 (a). The Gabor angle information is shown for every fifth pixel over a portion of the original color image.

tor kernel rotated to the angles $\alpha_k = -\pi/2 + \pi k/K$, $k = 0, 1, \dots, K - 1$. Given $W_k(x, y)$ as the result of filtering the image, $I(x, y)$, with $L_k(x, y)$, the orientation of the detected line is obtained as [22]

$$\theta(x, y) = \alpha_{k_{\max}}, \text{ where } k_{\max} = \arg\{\max[W_k(x, y)]\}. \quad (10)$$

The magnitude response of the result is obtained as $W_{k_{\max}}(x, y)$. The line operator does not provide a specific parameter for scaling; multiscale analysis is performed by applying the line operator to each level of the Gaussian pyramid decomposition of the original image.

In the present work, values of $M = 15$ and $K = 180$ were determined to provide the best results for detection of vessels using the training set of the DRIVE database, and were employed for further analysis. Figure 5 shows the magnitude response of line operators as applied to the image in Figure 1 (a).

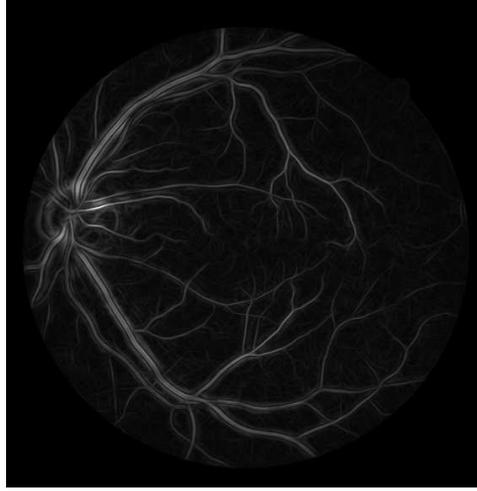


Figure 5. Magnitude response of line operators for the image in Figure 1 (a), obtained using $M = 15$ and $K = 180$.

3.4 Matched Filters

The method of Chaudhuri et al. [5], as explained in Section 1, was implemented in the present work for the detection of blood vessels. The method assumes that blood vessels have a negative contrast with respect to the background, so the Gaussian template will need to be inverted. The main kernel of the matched filter is expressed as

$$M(x, y) = -\exp(-x^2/2\sigma^2), \text{ for } -L/2 \leq y \leq L/2, \quad (11)$$

where L represents the length of the vessel segment that is assumed to have a constant orientation and σ is the STD of the Gaussian. The main kernel of the filter is oriented along the y -axis; in order to detect blood vessels at different orientations, the main kernel is rotated at multiple angles.

In this work, detection of blood vessels using matched filters is performed by taking the maximum filter response of a bank of $K = 180$ filters over the range $[-\pi/2, \pi/2]$ with $L = 15$ and $\sigma = 1$, as determined

using the training set of the DRIVE database. Figure 6 represents the magnitude response of matched filters obtained for the image in Figure 1 (a).



Figure 6. Magnitude response of matched filters obtained using $L = 15$, $\sigma = 1$, and $K = 180$ for the image in Figure 1 (a).

3.5 Multifeature Analysis

In the present work, various features are combined using pattern classification methods [multilayer neural networks (MNN)] in order to distinguish pixels belonging to blood vessels from the background. The features used are:

- the vesselness measure of Frangi et al. [9],
- the vesselness measure of Salem et al. [10],
- the magnitude response of line operators [22],
- the magnitude response of matched filters [5],

- the gamma-corrected [26] inverted green component, and
- the magnitude responses of Gabor filters for $\tau = \{4, 8, 12\}$ [15].

The inverted green (G) component of the RGB color space provides high contrast for blood vessels. Therefore, a gamma-corrected version [26] of the inverted G -component image is also used as a feature in order to improve the result of classification of blood vessels. The value of gamma used for gamma correction in this work is 2.4, with the pixel values normalized to the range $[0, 1]$.

All the MNNs used in this work for multifeature analysis contain two hidden layers with 15 nodes per hidden layer. The number of input layer nodes is equal to the number of features being used and the output layer always contains one node. A tangent sigmoid (tansig) function was used as the training function for each hidden layer and a pure linear function was used at the output layer of the MNN. In each case, the MNN was trained using 10% of the available training data.

Sequential feedforward feature selection was used to determine which combination of the features listed above would provide the best results for multifeature analysis; the feature selection method selected all eight available features.

3.6 Thresholding for Segmentation of Vessels

The histogram of the intensity values of the result of vessel detection is not bimodal with a clear separation of the pixels belonging to blood vessels from the background pixels. Considering the ground-truth data provided for the 20 training images of the DRIVE database within their FOV, only 13% of an image is covered by vessel pixels. As a result, thresholding the gray-scale output images of vessel detection methods with high accuracy is a rather difficult task. Several automated thresholding methods, including Otsu's method [27], a moment-preserving thresholding method [28], the Ridler-Calvard thresholding method [31], the Rutherford-Appleton threshold selection (RATS) method [29], and an entropy-based thresholding method [30] are explored in this work. Additionally, it is possible to use a single fixed-value threshold for each

single feature or the discriminant result of multifeature analysis, obtained as the value of the point on the ROC curve that is closest to the point $[0, 1]$, with the ROC curve obtained by using the training set of images.

Considering that the majority of the pixels in a retinal image are background pixels and possess a low intensity value in the results of vessel detection methods, it could be beneficial to select a binarization threshold by analyzing only the pixels that belong to the boundaries of the vessels. We propose an adaptive thresholding method in which the boundaries of blood vessels are detected using Gabor filters with a low value of $\tau = 3$ pixels. The result is then thresholded at 0.2 of the normalized intensity to obtain the boundaries of blood vessels. Morphological dilation is then applied to the binary image of the vessel boundaries using a disk-shaped structuring element of radius two pixels to identify the adjacent regions of boundaries of blood vessels. The histogram of the pixels (with 25 bins) in the selected regions was observed to have an abrupt change in the values for two adjacent bins. The two adjacent bins with the largest probabilities of values are identified and their corresponding pixel intensity values are noted. An adaptive threshold for each image is obtained as the average of the intensity values corresponding to the two identified bins.

The performance of the proposed and selected thresholding techniques was analyzed in terms of the sensitivity (SE), specificity (SP), and accuracy (Acc) of the segmentation of blood vessels with reference to the ground-truth images provided in the DRIVE database.

3.7 Postprocessing for Removal of FP Pixels Around the ONH

In the results obtained using various vessel detection techniques, the boundary and edges of the ONH are also detected since they represent an abrupt change in intensity, i.e., an edge, which can lead to artifacts (FP pixels) when the gray-scale results are thresholded. In the present work, the FP pixels associated with the boundary of the ONH are identified using an angular difference index (ADI), defined as [1]

$$\text{ADI} = \cos [\theta(i, j) - \gamma(i, j)], \quad (12)$$

where $\theta(i, j)$ is the Gabor angle response and $\gamma(i, j)$ is the radial angle with respect to the center of the ONH, as shown in Figures 7(a) and (b), respectively. The ranges of θ and γ are limited to $[-\pi/2, \pi/2]$. The values of ADI are computed for each pixel within the annular region limited by two circles of radii $0.75r$ and $2r$, where $r = 0.8$ mm is the average radius of the ONH [16]. The center of the ONH was automatically detected using phase portrait analysis of the Gabor angle response [32]. The pixels for which ADI is less than 0.15, i.e., the difference between the Gabor angle and the radial angle is greater than 81° , are removed from the output of the classifier, because they represent artifacts related to the ONH. This step may cause the loss of a few pixels belonging to vessels.

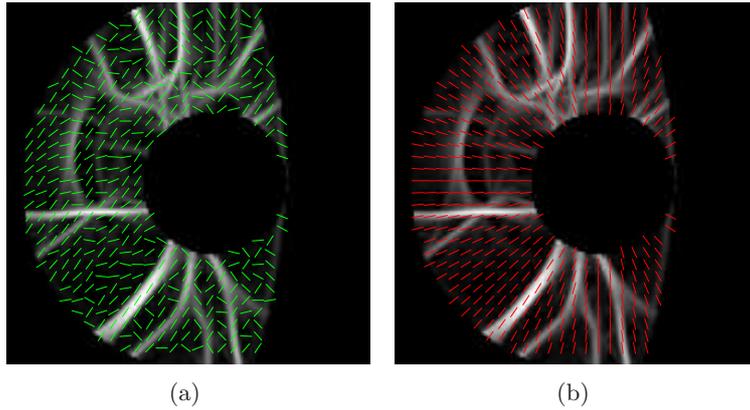


Figure 7. (a) Gabor angle response and (b) radial angle with respect to the center of the ONH for the selected annular region.

4 Results

In order to obtain each feature mentioned in Section 3.5, the luminance component, Y , of the YIQ color space, defined as

$$Y = 0.299R + 0.587G + 0.114B, \quad (13)$$

where R , G , and B represent the red, green, and blue color components in the RGB color space, respectively, was used as the input to the vessel detection methods.

The performance of the proposed methods was tested with the set of 20 test images of the DRIVE database. The training set of 20 images was used to determine the best values for the parameters of the filters (Section 3), to perform the training of the MNNs (Section 3.5), and to determine a suitable threshold for segmentation of vessels (Section 3.6). The ground-truth images of blood vessels were used as reference to perform ROC analysis.

The results of detection of blood vessels were evaluated in terms of the area under the ROC curve (A_z), which are provided in Table 1. For comparative analysis, the result of another previously proposed method [6], as discussed in Section 1, that was not implemented in this work is also presented in Table 1.

Table 1. Comparison of the efficiency of detection of blood vessels in the retina obtained by different methods, as implemented in this work, and another method, for the test set (20 images) of the DRIVE database [7].

Detection method	A_z
Vesselness measure of Salem et al.	0.892
Vesselness measure of Frangi et al.	0.896
Line operators	0.905
Matched filters	0.928
Single-scale Gabor filters	0.950
Ridge-based segmentation [6]	0.952

Table 2 presents the results of performing vessel classification with MNN classifiers using various combinations of the proposed features, as mentioned in Section 3.5. For comparative analysis, the results of

the works of Soares et al. [8] and Lupaşcu et al. [13], who performed multiscale and multifeature analysis, respectively, are also presented in Table 2 (see Section 1 for the details of the methods).

Table 2. Results of detection of blood vessels in terms of A_z with the test set (20 images) of the DRIVE database. For all cases, MNN classifiers were used. Multiscale Gabor filters include the magnitude response images with scales of $\tau = \{4, 8, 12\}$ pixels. In order to keep the table entries short, the following acronyms for different features are used: Gabor filters (GF), vesselness measure of Frangi et al. (VF), vesselness measure of Salem et al. (VS), gamma-corrected green component (GC), matched filters (MF), and line operators (LO).

Detection method	A_z
VF, VS, GC, MF, and LO	0.948
Multiscale GF	0.960
Multiscale GF and LO	0.960
Multiscale GF and MF	0.960
Multiscale GF and VS	0.960
Multiscale GF and VF	0.960
Multiscale GF, VF, and GC	0.961
Multiscale GF, VF, VS, GC, MF, and LO	0.961
Multiscale complex GF [8]	0.961
Multifeature analysis (41 features) [13]	0.956

Figure 8 shows the result of multifeature and multiscale analysis for the image in Figure 1 (a) using four of the combinations given in Table 2.

Table 3 provides the performance of the three methods of entropy-based [30] thresholding, adaptive thresholding, and fixed-value thresholding. The methods of Otsu [27], moment-preserving thresholding method [28], Ridler-Calvard thresholding method [31], and the RATS method [29] were also tested in this work; however, since they did not provide better results than the three methods mentioned above, their

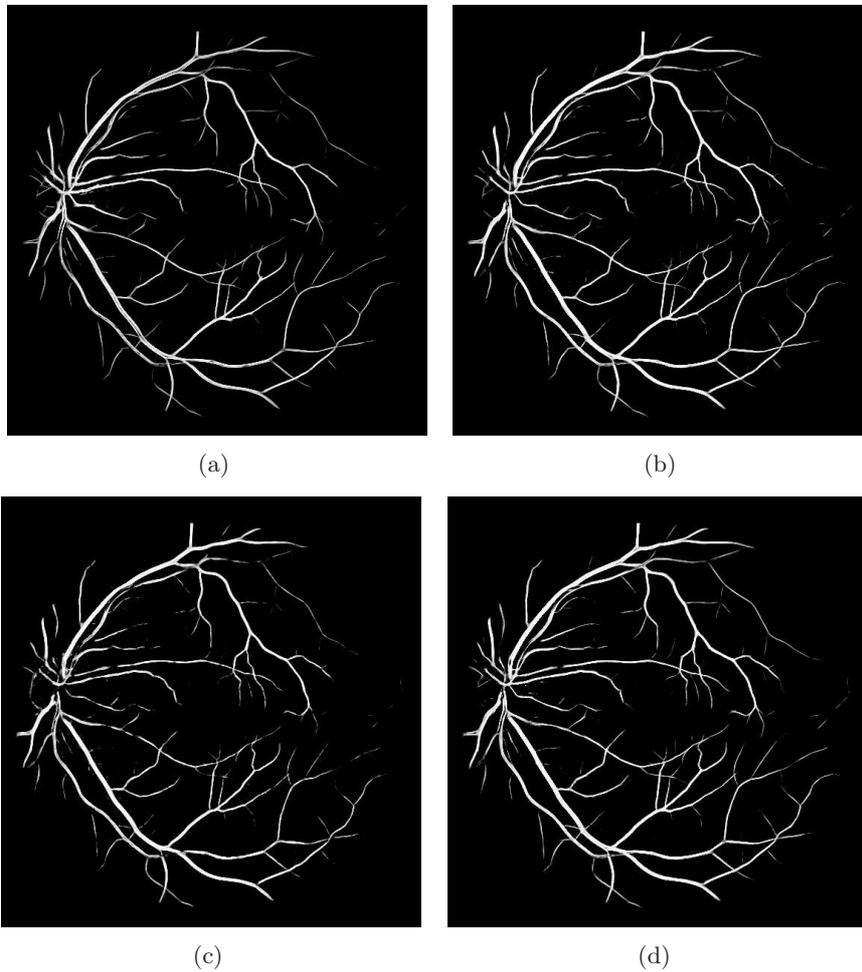


Figure 8. Results of multifeature and multiscale analysis using MNNs: (a) multiscale Gabor filters, (b) multiscale Gabor filters, vesselness measure of Frangi et al., and gamma-corrected inverted G -component, (c) vesselness measure of Frangi et al., vesselness measure of Salem et al., gamma-corrected inverted G -component, matched filters, and line operators, (d) multiscale Gabor filters, vesselness measure of Frangi et al., vesselness measure of Salem et al., gamma-corrected inverted G -component, matched filters, and line operators.

results are not presented in this table.

Figure 9 shows the results of thresholding the magnitude responses of single-scale Gabor filters ($\tau = 8$) [Figure 4 (a)] and the vesselness measure of Frangi et al. [Figure 3 (a)] using the fixed-value threshold, as well as the entropy-based thresholding method. It can be seen that the entropy-based method provides higher specificity [higher number of true-negative (TN) pixels] at the expense of lower sensitivity [lower number of true-positive (TP) pixels]. It can be seen that the boundary of the ONH is not segmented when using the entropy-based method in part (b) of the figure. However, the entropy-based method is incapable of segmenting the majority of the vessel pixels in the case of the vesselness measure of Frangi et al., which is likely due to the low intensity values provided by the method.

Figure 10 shows the results of thresholding the discriminant images in Figures 8 (b) and (d) obtained using multifeature analysis. Both thresholding methods perform well with the results of multifeature analysis, with the fixed-value threshold having a higher SE and the entropy-based method a higher SP. It can be seen that the boundary of the ONH is not segmented when using the entropy-based method. Note that the thresholds for parts (a) and (c) of the figure are negative, because the MNN is trained using a tansig function which maps the discriminant values to the range $[-1, 1]$.

The method for removing the FP pixels around the ONH was evaluated in combination with the methods for the detection of blood vessels based on the vesselness measures and multiscale Gabor filters. For removal of artifacts, the Gabor magnitude response image was thresholded using the fixed threshold as explained in Section 3.6, and the postprocessing technique was applied to the binarized image. The proposed postprocessing technique was applied to the 20 test images of the DRIVE database and was able to remove 224 FP pixels per image on the average, at the cost of losing 22 TP pixels per image on the average. Examples of removal of FP pixels around the ONH are shown in Figure 11.

Table 3. Performance of the proposed adaptive thresholding technique compared with the fixed-value thresholding method and the entropy-based threshold selection method. The fixed threshold was determined by using the point on the ROC curve with the shortest distance to the point $[0, 1]$ considering the 20 training images of the DRIVE database. The highest values of SE, SP, and Acc for each thresholding method are highlighted. In order to keep the table entries short, the following acronyms for different features are used: single-scale Gabor filters (SGF), multiscale Gabor filters (MGF), vesselness measure of Frangi et al. (VF), vesselness measure of Salem et al. (VS), gamma-corrected green component (GC), matched filters (MF), and line operators (LO).

Features	Fixed			Entropy			Adaptive		
	SE	SP	Acc	SE	SP	Acc	SE	SP	Acc
VS	0.816	0.886	0.877	0.056	0.998	0.875	0.863	0.752	0.766
VF	0.810	0.898	0.886	0.026	1.000	0.873	0.482	0.991	0.925
LO	0.810	0.839	0.835	0.145	0.994	0.883	0.556	0.922	0.875
MF	0.832	0.879	0.873	0.125	0.999	0.885	0.520	0.973	0.913
SGF	0.857	0.900	0.894	0.269	0.998	0.902	0.532	0.966	0.910
MGF	0.876	0.909	0.904	0.821	0.944	0.927	0.121	0.988	0.876
MGF and LO	0.876	0.912	0.907	0.827	0.945	0.929	0.150	0.974	0.867
MGF and MF	0.876	0.912	0.907	0.821	0.946	0.930	0.151	0.983	0.875
MGF and VS	0.873	0.914	0.909	0.824	0.947	0.931	0.138	0.987	0.877
MGF and VF	0.872	0.914	0.909	0.818	0.949	0.932	0.147	0.964	0.859
MGF, VF, and GC	0.877	0.914	0.909	0.829	0.948	0.932	0.134	0.993	0.882
VF, VS, GC, MF, and LO	0.853	0.903	0.896	0.806	0.944	0.926	0.128	0.990	0.878
MGF, VF, VS, GC, MF, and LO	0.876	0.918	0.912	0.856	0.934	0.924	0.146	0.983	0.874

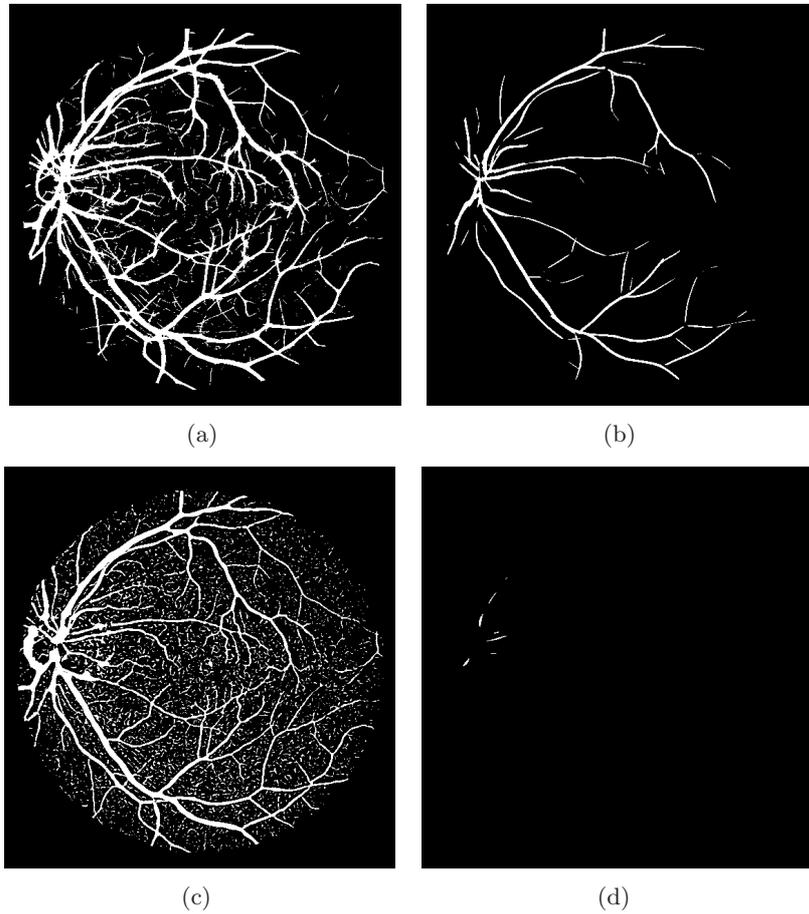


Figure 9. Binarized versions of the image in Figure 4 (a) (single-scale Gabor filter) using: (a) the fixed-value threshold $t = 0.0024$ of the maximum intensity value, with $SE = 0.867$, $SP = 0.908$, and $Acc = 0.903$; and (b) the entropy-based method ($t = 0.196$ of the normalized intensity value), with $SE = 0.397$, $SP = 0.996$, and $Acc = 0.919$. Binarized versions of the image in Figure 3 (a) (vesselness measure of Frangi et al.) using: (c) the fixed-value threshold $t = 3.22 \times 10^{-8}$ of the maximum intensity value, with $SE = 0.810$, $SP = 0.889$, and $Acc = 0.879$; and (d) the entropy-based method ($t = 0.290$ of the normalized intensity value), with $SE = 0.050$, $SP = 1.000$, and $Acc = 0.871$.

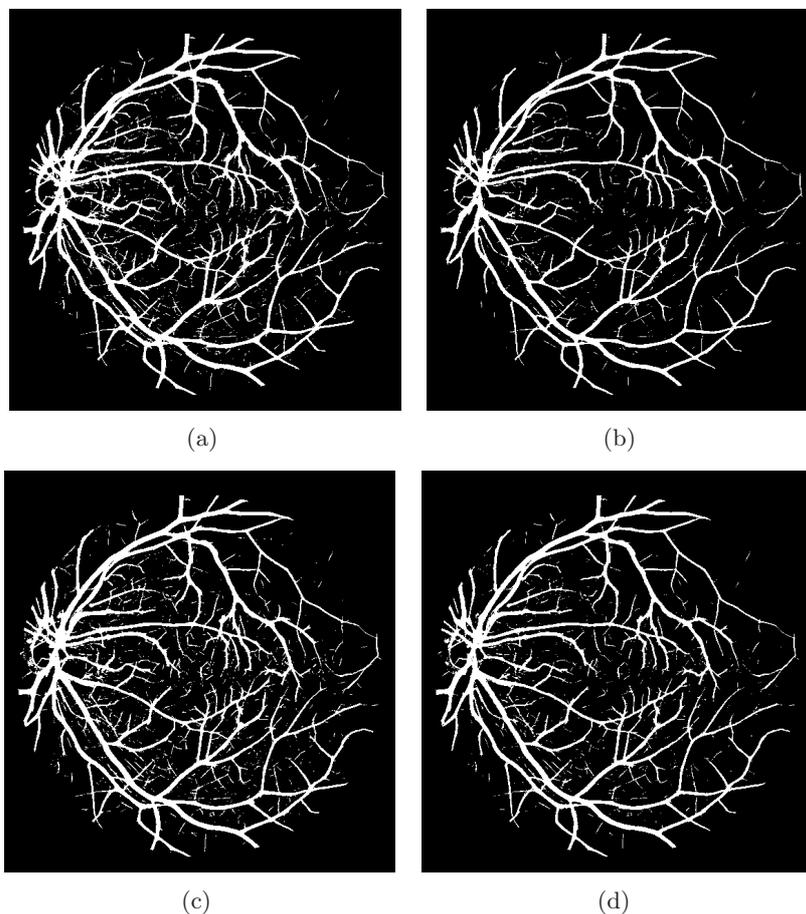


Figure 10. Binarized versions of the image in Figure 8 (b) using: (a) the fixed-value threshold $t = -0.743$ of the maximum intensity value, with $SE = 0.893$, $SP = 0.909$, and $Acc = 0.907$; and (b) the entropy-based method ($t = 0.263$ of the normalized intensity value), with $SE = 0.837$, $SP = 0.950$, and $Acc = 0.936$. Binarized versions of the image in Figure 8 (d) using: (c) the fixed-value threshold $t = -0.740$ of the maximum intensity value, with $SE = 0.895$, $SP = 0.909$, and $Acc = 0.908$; and (d) the entropy-based method ($t = 0.302$ of the normalized intensity value), with $SE = 0.870$, $SP = 0.933$, and $Acc = 0.925$.

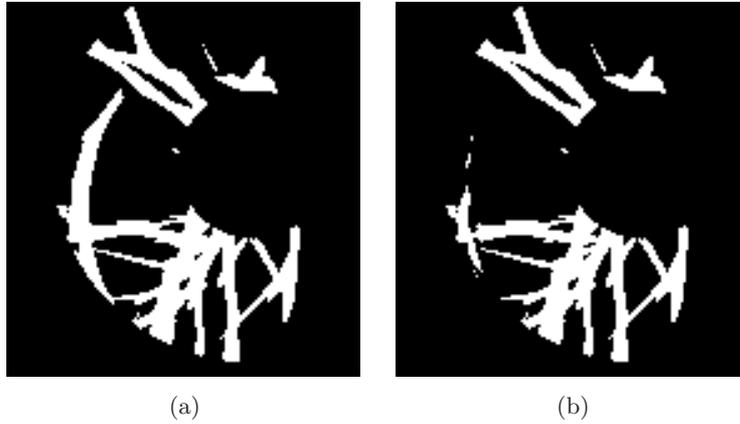


Figure 11. Example of removal of ONH artifacts: (a) thresholded Gabor magnitude response image, and (b) the same region after the removal of ONH artifacts.

5 Discussion

As evident from the results in Table 1, even the use of a combination of large number of features (41) [13], does not lead to substantial increase in the value of A_z . The large number of FP pixels caused by over segmentation of small blood vessels seems to be the limiting factor in achieving higher A_z values. The accuracy of detection of blood vessels could be increased if thin, single-pixel-wide blood vessels are detected accurately. However, thin blood vessels may not be important in the analysis of retinal vasculature as only changes in the major vessels have been observed to be clinically significant [23, 33].

Based on the results obtained in this work, a single-scale Gabor filter is capable of detecting blood vessels with accuracy ($A_z = 0.950$) not substantially different from the highest A_z value obtained with the result of multifeature analysis in this work ($A_z = 0.961$). It would be of interest to determine if the difference between the A_z values given above is statistically significant.

Using a Lenovo Thinkpad T510, equipped with an Intel Core i7

(Hyper-threaded-dual-core) 2.67-GHz processor, 4 MB of level 2 cache, 8 GB of DDR3 RAM, running 64-bit Windows 7 Professional, and using 64-bit Matlab software, the run time for single-scale Gabor filters with $K = 180$, for a single color image from the DRIVE database is approximately 13.5 seconds. The preprocessing step takes about 8.8 seconds to execute.

Although the reduction of FP pixels is visible in the example shown in Figure 11, the result did not lead to a substantial increase in specificity. This is mainly because the total number of FP pixels removed (224 pixels, on the average, per image) by the postprocessing step is small compared to the total number of FP pixels (20,038 pixels, on the average, per image) and the total number of TP pixels (24,888 pixels, on the average, per image). Such a postprocessing step is most beneficial when applied to a skeleton of the vasculature in applications where it is important to process only pixels that belong to vessels, such as tracking the major branches of vessels [34] and measurement of vessel thickness [35].

The problem of segmentation of vessels via thresholding is crucial to applications that deal with measurement and analysis of the statistics of blood vessels. In this work, we have analyzed seven different thresholding methods. Based on the results presented in Table 3, the thresholding method using a fixed value obtained using the ROC curve for the training set of images provides the most consistent results in terms of SE, SP, and accuracy. The entropy-based thresholding method provides a higher SP and similar SE in comparison to the fixed-value method when applied to the results of multifeature analysis. However, the entropy-based method has low sensitivity when applied to single features. The proposed adaptive thresholding method does not perform better than the other two methods. Depending on the desired application, either the fixed-value method, the entropy-based method, or a combination of the two could be employed.

6 Conclusion

In this study, we have analyzed multiscale and multifeature methods for the detection of blood vessels in retinal fundus images, and achieved a maximum A_z value of 0.961 using the 20 test images of the DRIVE database. The results of the present study indicate that the state-of-the-art methods for the detection of blood vessels perform at high levels of efficiency and that combining several features may not yield better results. The result of the fixed-value thresholding or the entropy-based method could be helpful in analyzing the thickness and tortuosity of blood vessels.

References

- [1] A. K. Dhara, R. M. Rangayyan, F. Oloumi, S. Mukhopadhyay. “Methods for the detection of blood vessels in retinal fundus images and reduction of false-positive pixels around the optic nerve head,” in *Proc. 4th IEEE International Conference on E-Health and Bioengineering - EHB 2013*, Iași, Romania, November 2013, pp. 1–6.
- [2] N. Patton, T. M. Aslam, T. MacGillivray, I. J. Deary, B. Dhillon, R. H. Eikelboom, K. Yogesan, I. J. Constable. “Retinal image analysis: Concepts, applications and potential,” *Progress in Retinal and Eye Research*, vol. 25, no. 1, pp. 99–127, 2006.
- [3] American Academy of Pediatrics, American Association for Pediatric Ophthalmology and Strabismus, American Academy of Ophthalmology, “Screening examination of premature infants for retinopathy of prematurity,” *Pediatrics*, vol. 108, pp. 809–811, 2001.
- [4] R. van Leeuwen, U. Chakravarthy, J. R. Vingerling, C. Brussee, A. Hooghart, P. Mudler, P. de Jong. “Grading of age-related maculopathy for epidemiological studies: Is digital imaging as good as 35-mm film?,” *Ophthalmology*, vol. 110, no. 8, pp. 1540–1544, 2003.

- [5] S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, M. Goldbaum. “Detection of blood vessels in retinal images using two-dimensional matched filters,” *IEEE Transactions on Medical Imaging*, vol. 8, no. 3, pp. 263–269, 1989.
- [6] J. Staal, M. D. Abràmoff, M. Niemeijer, M. A. 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, 2004.
- [7] “DRIVE: Digital Retinal Images for Vessel Extraction,” [www.isi.uu.nl/ Research/ Databases/ DRIVE/](http://www.isi.uu.nl/Research/Databases/DRIVE/) , accessed December 2013.
- [8] J. V. B. Soares, J. J. G. Leandro, R. M. Cesar Jr., H. F. Jelinek, M. J. Cree. “Retinal vessel segmentation using the 2-D Gabor wavelet and supervised classification,” *IEEE Transactions on Medical Imaging*, vol. 25, no. 9, pp. 1214–1222, 2006.
- [9] A. F. Frangi, W. J. Niessen, K. L. Vincken, M. A. Viergever. “Multiscale vessel enhancement filtering,” in *Medical Image Computing and Computer-Assisted Intervention - MICCAI98*, vol. 1496 of *Lecture Notes in Computer Science*, pp. 130–137. Springer, Berlin, Germany, 1998.
- [10] M. N. Salem, A. S. Salem, A. K. Nandi. “Segmentation of retinal blood vessels based on analysis of the Hessian matrix and clustering algorithm,” in *15th European Signal Processing Conference (EUSIPCO 2007)*, Poznan, Poland, September 2007, pp. 428–432.
- [11] C.-H. Wu, G. Agam, P. Stanchev. “A hybrid filtering approach to retinal vessel segmentation,” in *Biomedical Imaging: From Nano to Macro, 4th IEEE International Symposium on*, Arlington, VA, April 2007, pp. 604–607.
- [12] “Structured Analysis of the Retina,” [http:// www. parl. clemson. edu/ ahoover/ stare/ index. html](http://www.parl.clemson.edu/ahoover/stare/index.html), accessed December 2013.

- [13] C. A. Lupaşcu, D. Tegolo, E. Trucco. “FABC: Retinal vessel segmentation using AdaBoost,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 5, pp. 1267–1274, September 2010.
- [14] M. Sofka, C. V. Stewart. “Retinal vessel centerline extraction using multiscale matched filters, confidence and edge measures,” *IEEE Transactions on Medical Imaging*, vol. 25, no. 12, pp. 1531–1546, December 2006.
- [15] R. M. Rangayyan, F. J. Ayres, Faraz Oloumi, Foad Oloumi, P. Eshghzadeh-Zanjani. “Detection of blood vessels in the retina with multiscale Gabor filters,” *Journal of Electronic Imaging*, vol. 17, pp. 023018:1–7, April-June 2008.
- [16] F. Oloumi, R. M. Rangayyan, A. L. Ells. *Digital Image Processing for Ophthalmology: Detection and Modeling of the Retinal Vascular Architecture*, Morgan & Claypool, 2014, In press.
- [17] T. Lindeberg. “Edge detection and ridge detection with automatic scale selection,” *International Journal of Computer Vision*, vol. 30, no. 2, pp. 117–154, 1998.
- [18] B. S. Y. Lam, Y. Gao, A.W.-C. Liew. “General retinal vessel segmentation using regularization-based multiconcavity modeling,” *IEEE Transactions on Medical Imaging*, vol. 29, no. 7, pp. 1369–1381, July 2010.
- [19] T. Stösić, B. D. Stösić. “Multifractal analysis of human retinal vessels,” *IEEE Transactions on Medical Imaging*, vol. 25, no. 8, pp. 1101–1107, 2006.
- [20] F. Zana, J. C. Klein. “Segmentation of vessel-like patterns using mathematical morphology and curvature estimation,” *IEEE Transactions on Image Processing*, vol. 10, no. 7, pp. 1010–1019, July 2001.

- [21] M. Foracchia, E. Grisan, A. Ruggeri. “Detection of optic disc in retinal images by means of a geometrical model of vessel structure,” *IEEE Transactions on Medical Imaging*, vol. 23, no. 10, pp. 1189–1195, 2004.
- [22] F. J. Ayres, R. M. Rangayyan. “Design and performance analysis of oriented feature detectors,” *Journal of Electronic Imaging*, vol. 16, no. 2, pp. 023007:1–12, 2007.
- [23] C. M. Wilson, K. D. Cocker, M. J. Moseley, C. Paterson, S. T. Clay, W. E. Schulenburg, M. D. Mills, A. L. Ells, K. H. Parker, G. E. Quinn, A. R. Fielder, J. Ng. “Computerized analysis of retinal vessel width and tortuosity in premature infants,” *Investigative Ophthalmology and Visual Science*, vol. 49, no. 1, pp. 3577–3585, 2008.
- [24] R. N. Dixon, C. J. Taylor. “Automated asbestos fibre counting,” in *Institute of Physics Conference Series*, 1979, vol. 44, pp. 178–185.
- [25] R. Zwiggelaar, S. M. Astley, C. R. M. Boggis, C. J. Taylor. “Linear structures in mammographic images: detection and classification,” *IEEE Transactions on Medical Imaging*, vol. 23, no. 9, pp. 1077–1086, 2004.
- [26] R. C. Gonzalez, R. E. Woods. *Digital Image Processing*, Prentice Hall, Upper Saddle River, NJ, 2nd edition, 2002.
- [27] N. Otsu. “A threshold selection method from gray-level histograms,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-9, pp. 62–66, 1979.
- [28] W.-H. Tsai. “Moment-preserving thresholding: A new approach,” *Computer Vision, Graphics, and Image Processing*, vol. 29, no. 3, pp. 377–393, 1985.
- [29] J. Kittler, J. Illingworth, J. Föglein. “Threshold selection based on a simple image statistic,” *Computer Vision, Graphics, and Image Processing*, vol. 30, pp. 125–147, 1985.

- [30] J. N. Kapur, P. K. Sahoo, A. K. C. Wong. “A new method for gray-level picture thresholding using the entropy of the histogram,” *Computer Vision, Graphics, and Image Processing*, vol. 29, pp. 273–285, 1985.
- [31] T. W. Ridler, S. Calvard. “Picture thresholding using an iterative selection method,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 8, pp. 630–632, Aug 1978.
- [32] R. M. Rangayyan, X. Zhu, F. J. Ayres, A. L. Ells. “Detection of the optic nerve head in fundus images of the retina with Gabor filters and phase portrait analysis,” *Journal of Digital Imaging*, vol. 23, no. 4, pp. 438–453, August 2010.
- [33] C. M. Wilson, K. Wong, J. Ng, K. D. Cocker, A. L. Ells, A. R. Fielder. “Digital image analysis in retinopathy of prematurity: A comparison of vessel selection methods,” *Journal of American Association for Pediatric Ophthalmology and Strabismus*, vol. 16, no. 3, pp. 223 – 228, 2011.
- [34] F. Oloumi, R. M. Rangayyan, A. L. Ells. “Tracking the major temporal arcade in retinal fundus images,” in *Proc. IEEE Canada 27th Annual Canadian Conference on Electrical and Computer Engineering (CCECE)*, Toronto, ON, Canada, May 2014.
- [35] F. Oloumi, R. M. Rangayyan, A. L. Ells. “Measurement of vessel width in retinal fundus images of preterm infants with plus disease,” in *Proc. IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, Lisbon, Portugal, May 2014, In press.
- [36] “Hessian based Frangi vesselness filter,” <http://www.mathworks.com/matlabcentral/fileexchange/24409-hessian-based-frangi-vesselness-filter>.

Detection of Blood Vessels in Retinal Fundus Images

Faraz Oloumi, Ashis K. Dhara,
Rangaraj M. Rangayyan, and Sudipta Mukhopadhyay

Received June 5, 2014

Faraz Oloumi and Rangaraj M. Rangayyan*
Department of Electrical and Computer Engineering,
Schulich School of Engineering, University of Calgary,
Calgary, Alberta, Canada T2N 1N4.
Phone: +1 (403) 220 – 6745
E-mail: *ranga@ucalgary.ca

Ashis K. Dhara and Sudipta Mukhopadhyay
Department of Electronics and Electrical Communication Engineering,
Indian Institute of Technology,
Kharagpur 721302, India
Phone: +91 3222 283568

Optic Disc Identification Methods for Retinal Images

Invited Article

Florin Rotaru, Silviu Ioan Bejinariu, Cristina Diana Niță,
Ramona Luca, Camelia Lazăr

Abstract

Presented are the methods proposed by authors to identify and model the optic disc in colour retinal images. The first three our approaches localized the optic disc in two steps: a) in the green component of RGB image the optic disc area is detected based on texture indicators and pixel intensity variance analysis; b) on the segmented area the optic disc edges are extracted and the resulted boundary is approximated by a Hough transform. The last implemented method identifies the optic disc area by analysis of blood vessels network extracted in the green channel of the original image. In the segmented area the optic disc edges are obtained by an iterative Canny algorithm and are approximated by a circle Hough transform.

Keywords: optic disc, retinal images, vessel segmentation, Hough transform.

1 Introduction

Proposed in the last years, there is a huge literature on automatic analysis of retinal images, the optic disc evaluation being part of this work. The recognition and assessment of optic disc in retinal images are important tasks to evaluate retina diseases as diabetic macular edema, glaucoma, etc. The uneven quality and diversity of the acquired retinal images and the large variations between individuals made the

automatic analysis a strongly context dependent task. Even so valuable methods were proposed and the authors have reported their results lately using public retinal images databases as DRIVE (40 images), DIARETDB1 (89 images) or STARE (402 images).

Part of the proposed techniques, called bottom-up methods, first locate the optic disc and then starting from that area track the retinal vessels and do the required measurements [1], [2], [3], [8]. There is another approach of retinal image analysis that tracks the retinal vessels and gets the optic disc as the root of the vessels tree. The second one is called top-down approach [4], [5], [7], [17]. Besides these two trends, there are mixed approaches, independently detecting the optic disc centre and retinal vessels, as the ones proposed in [15], [23]. In these ones the blood vessel network analysis is combined with other methods to locate optic disc area.

A bottom-up technique is presented in [1]. The optic disc recognition and modelling were done in two steps. The first one locates the optic disc area using a voting procedure. There were implemented three methods: the maximum difference method that computes the maximum difference between the maximum and minimum grey levels in working windows, the maximum variance method and frequency low pass filter method. The green channel of the RGB input image was used. The first method filters the image using a 21×21 median filter and then for each pixel in the filtered image the difference between the maximum and minimum grey levels in a 21×21 window centred on the current pixel is computed. The pixel with the maximum difference is chosen optic disc centre candidate. Second method calculates the statistical variance for every pixel of the green channel using a 71×71 window. Then, the blue channel image is binarized by Otsu technique. The pixel in the green channel with the maximum statistical variance and having at least 10 white neighbours pixels in a 101×101 area centred on it but in the blue binarized channel is proposed as disc centre. The third voting method transforms the green channel from spatial domain to frequency domain, by a Fourier transform. The magnitude image of the transform is filtered using a Gaussian low-pass filter and the result image is transformed back to the spatial domain. The bright-

est pixel in result image is taken as the third optic disc centre candidate. Finally the voting procedure chooses the estimated disc centre from the three candidates: 1) if all three candidates are close to their centre of mass, the centre of mass is proposed as an approximate disc centre; 2) if only two from three candidates are close to the centre of mass of all three points, the centre of mass of these two candidates is chosen; 3) if all candidates are far apart from their centre of mass, the candidate proposed by the second method is chosen, the most reliable considered by the authors.

Part of this optic disc area segmentation was also implemented in our first system to process retinal images.

In the second step of the whole methodology proposed in [1] a 400×400 window is centred on the estimated disc centre, and extracted from green and red channels of the original image. A morphological filter is employed in [6] to erase the vessels in the new window and a Prewitt edge detector is then applied. Then, by the same Otsu technique, the image is binarized. The result is cleaned by morphological erosion and finally a Hough transform is applied to get the final optic disc boundary. The boundary with the best fitting from the two channels is chosen. The authors report for 1200 retinal images a score of 100% for approximated localisation and a score of 86% for final optic disc localisation.

Another bottom up approach to locate optic disc area was proposed in [8]. The method combines two algorithms: a pyramidal decomposition using Haar wavelet transform and an optic disc contour detection based on Hausdorff distance. Areas, usually white patches that might disturb the right disc area detection are eliminated during the pyramid synthesis. In the end, the low resolution level contains only the useful information. Finally the disc is selected from ten optic disc candidates.

In [3] another automatic optic disc detection was proposed based on majority voting for a set of optic disc detectors. There were employed five methods to detect optic disc centre: pyramidal decomposition [8], edge detection [8], entropy filter [14], fuzzy model [5] and Hough transform [10]. Each of the five methods is applied on the whole working image. A circular template is fit on each pixel in the initial image to

count the outputs of these algorithms that fall within the radius. The circle with the maximum number of optic disc detector outputs in its radius is the chosen area to refine the optic disc detection. An improved version of the voting method was proposed in [2].

From the top down methods the one proposed in [4] detects the retinal vessels convergence using a voting-type algorithm named fuzzy convergence. In another paper [5], in a first step there are identified the four main vessels in the image. Then the four branches are modelled by two parabolas whose common vertex is identified as the optic disc centre.

Another top down approach is proposed in [17]. The blood vessel network is segmented after a sequence of morphological operations:

- a) the bright areas, associated with diabetic lesions, are removed applying a morphological operator to detect regional minima pixels and then the resulted image is reconstructed by dilation;
- b) the result background image is enhanced by a morphological contrast operation and then a Gaussian filter is applied;
- c) the elongated low intensities regions, associated with vascular tree, are extracted with a top-hat by closing operator;
- d) the maximum of openings are retained for a structuring element of 80 pixels long segment and 24 orientations. These are the main branches of the vessels tree;
- e) the vascular tree is then estimated by reconstruction by dilation using the result image from step d) as marker image and the result image from step c) as mask element;
- f) the grey level image resulted in previous step is binarized using a morphological operator to detect regional minima pixels as in step a). The result is complemented;
- g) the skeleton of the vessel tree is obtained in the binary image by morphological operation;
- h) the useless short vessels branches are eliminated by a 20 step pruning operation.

In the resulted vessel tree image a point close to optic disc is calculated: a) the holes of the vessel network are filled; b) the tree branches are thinned; c) a recursive pruning operation is applied until no more reduction is possible, so only the main parabolic branch remains. The mass centre of the parabolic branch is considered the point closest to the optic disc.

In [17] other optic disc detection methods taxonomy is proposed. There is identified a first group of methods, [7], [18 – 21], that localizes the optic disc centre as the convergence point of the main blood vessels. However, these methods can be assimilated to the top-down category. From the second category group identified in [17], M. Niemeijer [23] uses a mixed algorithm combining the vessel network analysis and other segmentation method to locate optic disc area. The rest of the methods proposed in the second group of papers, [13], [22], [24 – 28] can be assimilated to the bottom-up methods. For two of these papers, [27] and [28], the main purpose is the exudate detection, so the optic disc detection and elimination are mandatory. While in [28] the optic disc area is identified using morphological operators, in [27], besides morphological filtering techniques, the watershed transformation is used. Another approach [22] from the second group identifies the optic disc using specialized template matching and segmentation by a deformable contour model. In [25] a genetic algorithm is proposed to localize the optic disc boundary. In [26] the authors utilize texture descriptors and a regression based method to find the most likely circle fitting the optic disc.

Most of the papers mentioned in the taxonomy proposed in [17] report very good results of detecting optic disc area for images from DRIVE or DIARETDB1 database or both.

2 Optic disc area segmentation methods

To locate the optic disc area we started following a similar methodology as the one proposed in [1]. In the first attempt tests have been done on 720×576 RGB retinal images [11], provided by our collaborators from Grigore T. Popa University of Medicine and Pharmacy, Iași, Romania

(UMP). From the three methods of the voting procedure presented in [1] we obtained good optic disc area localisation with a modified Low-Pass Filter Method and the Frequency Low Pass Filter Method.

The first method was implemented as in [1]. The green channel of the input image was transformed in frequency domain and on the image of the magnitude of the FFT transform a Gaussian low-pass filter was applied:

$$H(u, v) = \exp\left(-\frac{D^2(u, v)}{2D_0^2}\right), \quad (1)$$

where $D(u, v)$ is the Euclidean distance from point (u, v) to the origin of frequency domain and D_0 is the cutoff frequency, of 25 Hz. The result was transformed back to the spatial domain and the brightest pixel of the result image was chosen as an optic disc area centre candidate.

For the second voting procedure we tried the Maximum Difference Method proposed in [1]. But good results were obtained with an approach derived from this one. As in [1], a 21×21 median filter was applied on the green channel of the input image to eliminate isolated peaks. Then for each (i, j) pixel of the filtered green channel $I(x, y)$ the difference between the maximum grey value and minimum grey value of the pixels inside a 21×21 window centred on the current (i, j) pixel is calculated:

$$Diff(i, j) = I_W^{\max}(i, j) - I_W^{\min}(i, j). \quad (2)$$

There are stored four pixels with the greatest values $Diff(i, j)$. Then, starting from texture operators:

$$\begin{aligned} L5 &= \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \end{bmatrix}, \\ E5 &= \begin{bmatrix} -1 & -2 & 0 & 2 & 1 \end{bmatrix}, \\ S5 &= \begin{bmatrix} -1 & 0 & 2 & 0 & -1 \end{bmatrix}, \end{aligned} \quad (3)$$

where:

- $L5$ – mask to assess the grey level average;
- $E5$ – edge mask;
- $S5$ – corner mask,

the following masks, as in [8], are synthesized:

$$\begin{aligned}
 L5^t x E5 &= \begin{bmatrix} -1 & -2 & 0 & 2 & 1 \\ -4 & -8 & 0 & 8 & 4 \\ -6 & -12 & 0 & 12 & 6 \\ -4 & -8 & 0 & 8 & 4 \\ -1 & -2 & 0 & 2 & 1 \end{bmatrix}, \\
 L5^t x S5 &= \begin{bmatrix} -1 & 0 & 2 & 0 & 1 \\ -4 & 0 & 8 & 0 & 4 \\ -6 & 0 & 12 & 0 & 6 \\ -4 & 0 & 8 & 0 & 4 \\ -1 & 0 & 2 & 0 & 1 \end{bmatrix}, \\
 E5^t x L5 &= \begin{bmatrix} -1 & -4 & -6 & -4 & -1 \\ -2 & -8 & -12 & -8 & -2 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 8 & 12 & 8 & 2 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}, \\
 S5^t x L5 &= \begin{bmatrix} -1 & -4 & -6 & -4 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 8 & 12 & 8 & 2 \\ 0 & 0 & 0 & 0 & 0 \\ -1 & -4 & -6 & -4 & -1 \end{bmatrix}.
 \end{aligned} \tag{4}$$

For each pixel of the filtered green channel $I(x, y)$ the texture parameter $f(i, j)$ is computed:

$$f(i, j) = \tag{5}$$

$$\sqrt{(f_{L5^t x E5}(i, j))^2 + (f_{L5^t x S5}(i, j))^2 + (f_{E5^t x L5}(i, j))^2 + (f_{S5^t x L5}(i, j))^2}.$$

The value $f(i, j)$ is then normalized:

$$F(i, j) = \frac{f(i, j) - f_{\min}}{f_{\max} - f_{\min}}, \tag{6}$$

where $f_{\max} = \max\{f(i, j)\}$, $f_{\min} = \min\{f(i, j)\}$, $0 \leq i \leq H - 1$, $0 \leq j \leq W - 1$, H is the image height and W is the image width.

From the four pixels with the greatest values $Diff(i, j)$ selected in the first stage it is retained the one with the largest average of $F(i, j)$ computed on the 21×21 window centred on the processed pixel.

From our tests we concluded that on the retinal images of healthy patients or in the early stages of affection this second voting method provides a closer point to the real optic disc centre than the first one. However, on the retinal images strongly affected it fails. Finally, if the two methods to approximate the optic disc centre provide close centres, it is chosen the one computed by the second method. Otherwise the centre computed by the first method is chosen. The results obtained with the two procedures are illustrated in Figure 1, where the little cross is the point found out by maximum difference method and the big cross is the point provided by the second algorithm.

In a second step, we tried to apply the same methodology on images of resolution 2592×1728 [12]. Good optic disc area localization results were obtained only with the Low-Pass Filter Method (1), the third method of the voting procedure in [1].

Results of detecting approximate optic centre position by two voting procedures for low resolution image are illustrated by Figures 2.a and 2.b. A result using Low-Pass Filter Method for high resolution image is depicted in Figure 2.d.

The optic disc zone identification using the same voting procedure as in [12] failed for a third set of retinal images, of resolution 720×576 , provided by our collaborators from a different acquisition system. In the new set, the green channel was not always consistent in term of contrast. For some images the red channel is more suited to locate the optic disc area, for other ones the green channel is desirable.

The right channel selection was done using a square window scanning the whole red and green channels. The pixel intensity variance of the scanning window centre was computed. The window side length is the maximum expected circle diameter, estimated as a fraction of image width. The channel with the greatest maximum variance was chosen as working image to locate the optic disc area.

To identify the optic disc area in the selected image a new method was proposed [13]: the image was transformed in frequency domain

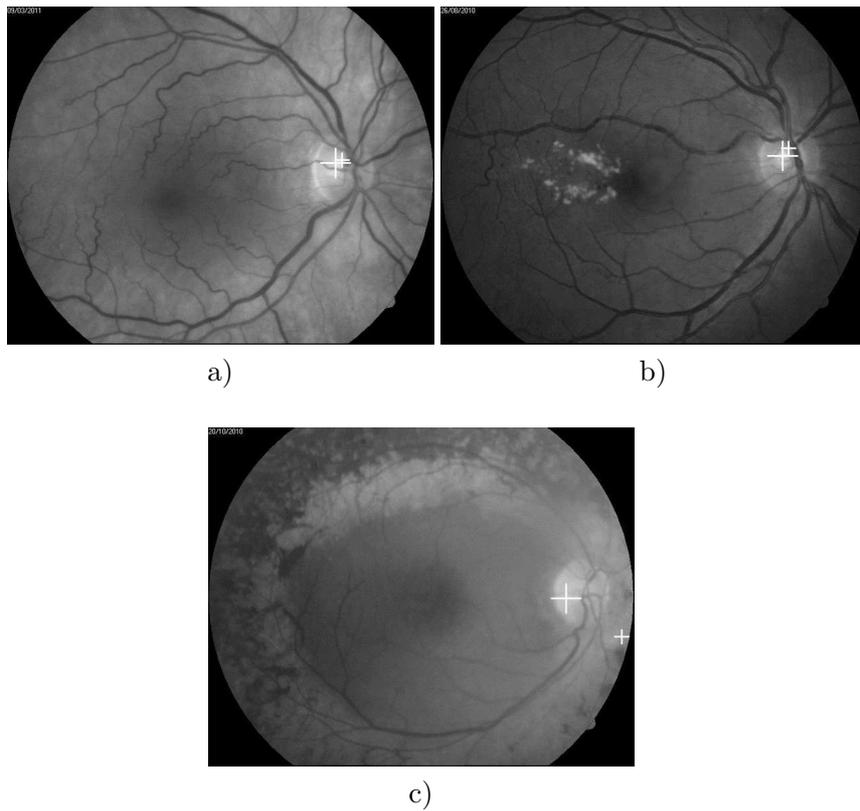


Figure 1. Results of detecting approximate optic centre position by two voting procedures. Point marked with little cross is provided by the first method and the one indicated by large cross is computed by the second voting algorithm. When the two points are far apart, as in the c) image, the centre computed by the first method is chosen – [11].

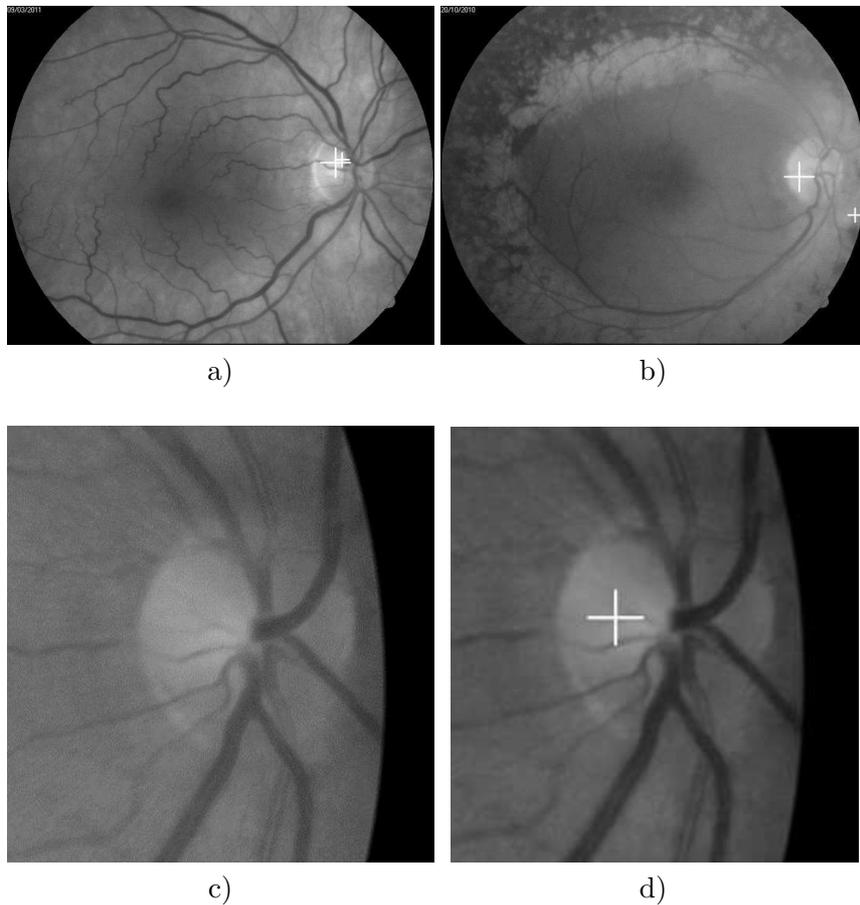


Figure 2. Results of detecting approximate optic centre position by two voting procedures for low resolution image, figures a) and b). Point marked with little cross is provided by the first method and the one indicated by large cross is computed by the second voting algorithm. When the two points are far apart, as in the b) image, the centre computed by the first method is chosen. A result using Low-Pass Filter Method for high resolution image is depicted in figure d). Part of original high resolution image is illustrated in figure c) – [12].

and the magnitude result of the FFT transform was filtered by Gaussian low-pass filter (1). The filtered result was transformed back to the spatial domain. Using the histogram of the new image, noted $I(i, j)$, a binarization threshold was computed. On each “bright” pixel (having a grey value greater than the binarization threshold) a square window of the same dimension as the one used in the channel selection step was centred. Then for every window centred in the “bright” pixels, intensity pixel variance, noted $Var(i, j)$, was calculated. Also, for every pixel $I(i, j)$ a texture measure was computed, using the same technique, Modified Maximum Difference Method, presented at the beginning of paragraph 2. A new image $F(i, j)$, of normalized texture values, was created. Finally, the pixel $O(m, n)$ of image $I(i, j)$ with $F(m, n) > F(i, j)$ and $Var(m, n) > 0.7max(Var(i, j))$ was declared as the centre of a window containing the optic disc.

Results of the new identification optic disc area procedure are depicted in Figure 3. The original image is 3.a. The images $I(i, j)$ and $F(i, j)$ are illustrated by Figures 3.b and 3.c. Black pixels in Figure 3.c are “dark” pixels of $I(i, j)$ not considered as possible optic disc centre candidates. The final result is depicted in Figure 3.d, where the cross indicates the centre of the working window in the selected channel.

Our previous methods to identify and model the optic disc provided very good results on retinal images of patients in early stages of ophthalmic pathologies as diabetic retinopathy or glaucoma. Tests have been made on three databases provided by our collaborators from UMP, Iași. We obtained good results also on images seriously affected by ophthalmic pathologies [12], [13].

The method proposed in [13] was tested on more than 100 images from STARE database of an image selection based on optic disc visibility. The results were good on the majority of these images but on other ones the optic disc was not correctly localized. Another method to segment the optic disc area was implemented based on the main blood vessels convergence point identification in the green channel.

Based on a technique employed from [6], in a first step the vessel tree of the green channel was iteratively segmented. A line of 27 pixels

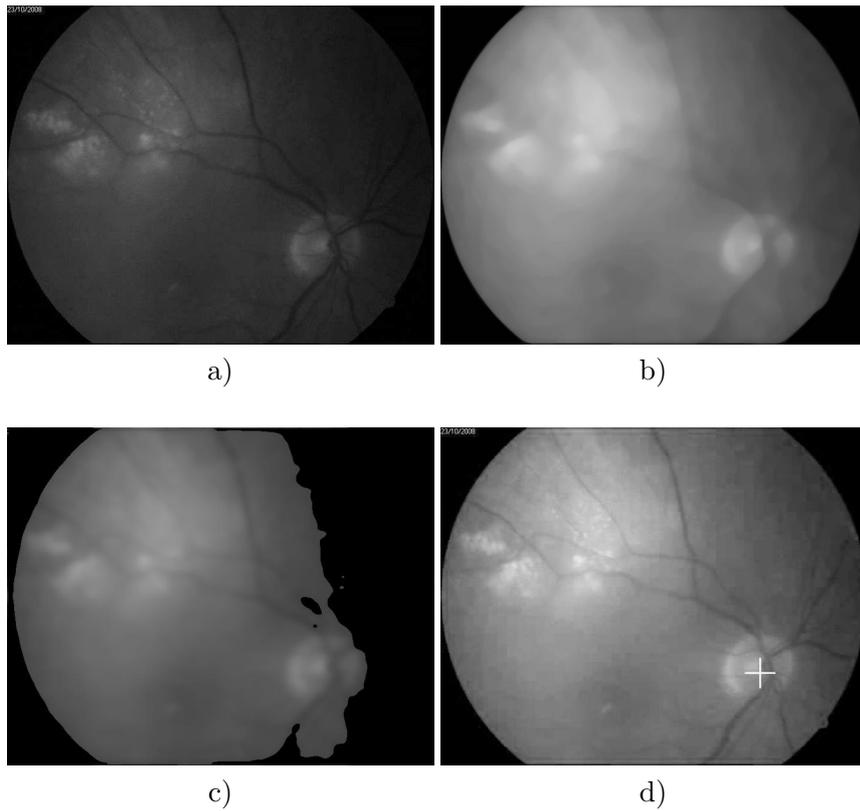


Figure 3. Result of detecting approximate optic centre position. a) Original RGB image; b) Gaussian filtering result in frequency domain $I(i, j)$; c) $F(i, j)$ image, where black pixels are “dark” pixels of $I(i, j)$; d) the cross indicates the working window centre – [13].

length and 1 pixel width was used as structuring element for an opening operation applied on the green channel for 12 different orientations of the element:

$$I_C = \min_{i=1,\dots,12} (\gamma_{B_i}(I)), \quad (7)$$

where I is the input image, B_i is the structuring element and $\gamma_{B_i}(I)$ is the result of the opening for orientation i of the structuring element.

Then, using I_C as marker image and the green channel as mask image a morphological reconstruction was performed:

$$I_C = R_I \left(\min_{i=1,\dots,12} (\gamma_{B_i}(I)) \right). \quad (8)$$

An image containing only background (large homogenous areas) results from:

$$I_B = \max_{i=1,\dots,12} (\gamma_{B_i}(I)). \quad (9)$$

Subtracting I_B from I_C an image containing only blood vessels is generated:

$$I_V = I_C - I_B. \quad (10)$$

Then an Otsu binarization of the I_V image is iteratively applied until one of the vessel configurations is obtained: a) a vessel tree with a big ratio (number of white pixels)/(surrounding tree rectangle area) and with surrounding tree rectangle area at least half of the input image; b) two big vessel branches as illustrated in Figure 4; c) a single large branch with a low ratio (number of white pixels)/(surrounding tree rectangle area) but with surrounding tree rectangle area at least half of the input image.

For cases b) and c) the principal axis of the binarized vessels is computed. A search region is computed considering the distances to principal axis of the endpoints of the branches in two branch case or of the distances to principal axis of the parabola points in case c).

For configuration a) the search area was considered the minimum surrounding rectangle. The search area for case b) is illustrated in Figure 5.

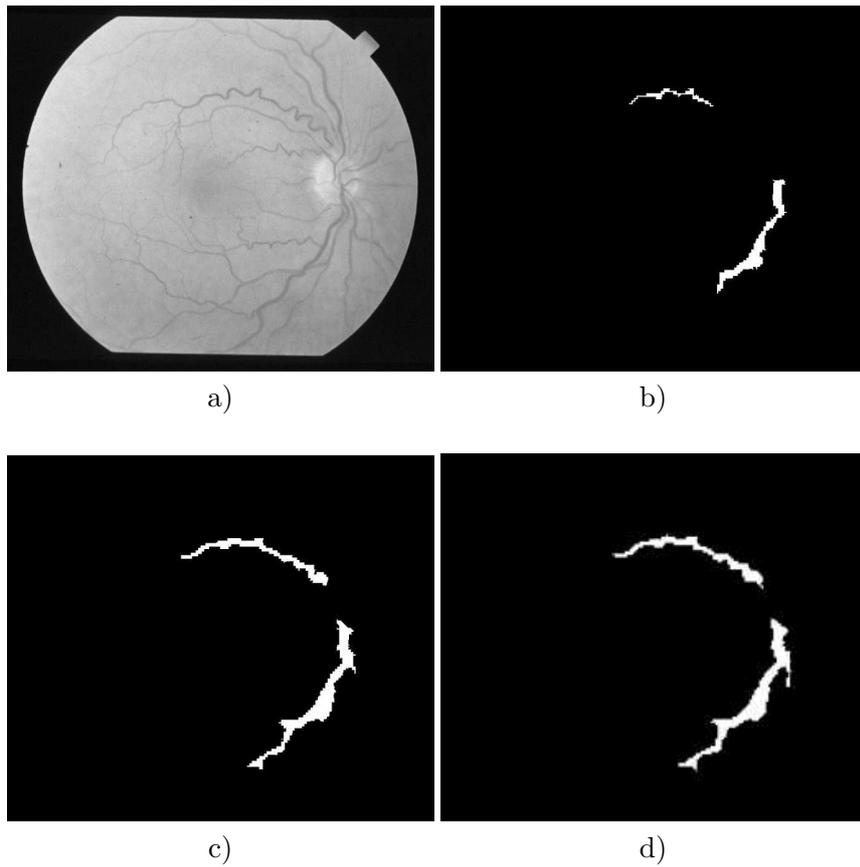


Figure 4. Results of I_V image binarization for two branch case. a) Original image; b) first step binarization; c) second step binarization; d) final step binarization.

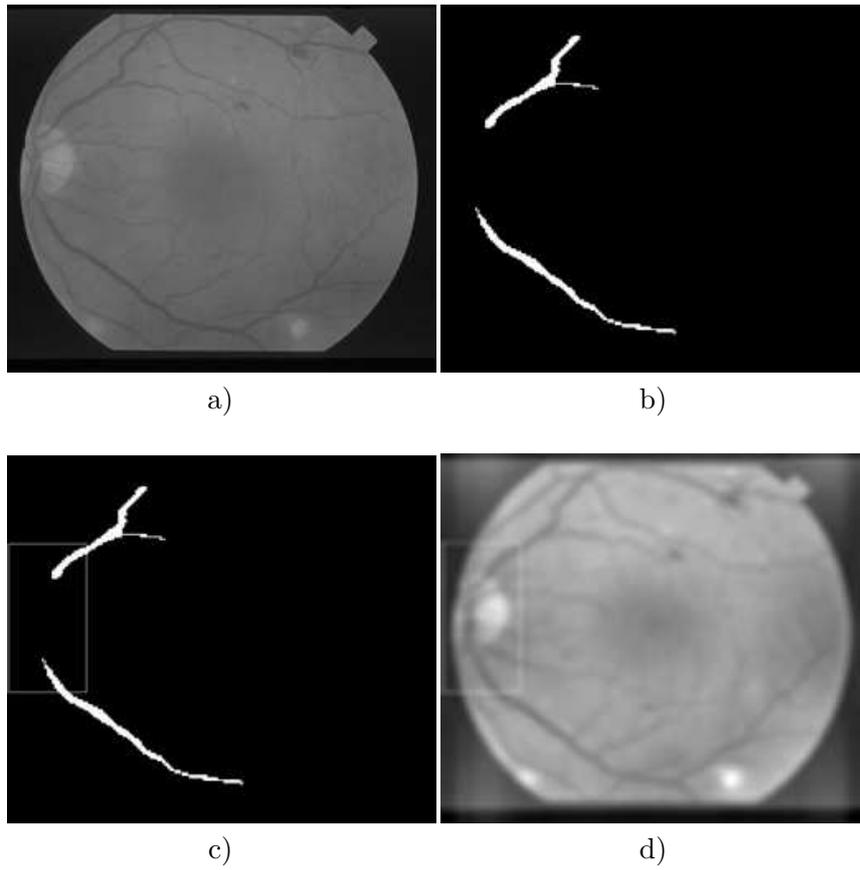


Figure 5. Search region for two branch case. a) original image; b) two final branches; c) search region in the working image; d) search region in the green channel were the next step is to approximately find the disc centre.

On the area established above, except of the FFT transform and Gaussian filtering, the procedure proposed in [13] was applied to identify a point to be declared the centre of a new window containing the optic disc.

Results of the new window centre calculation are depicted in Figure 6.

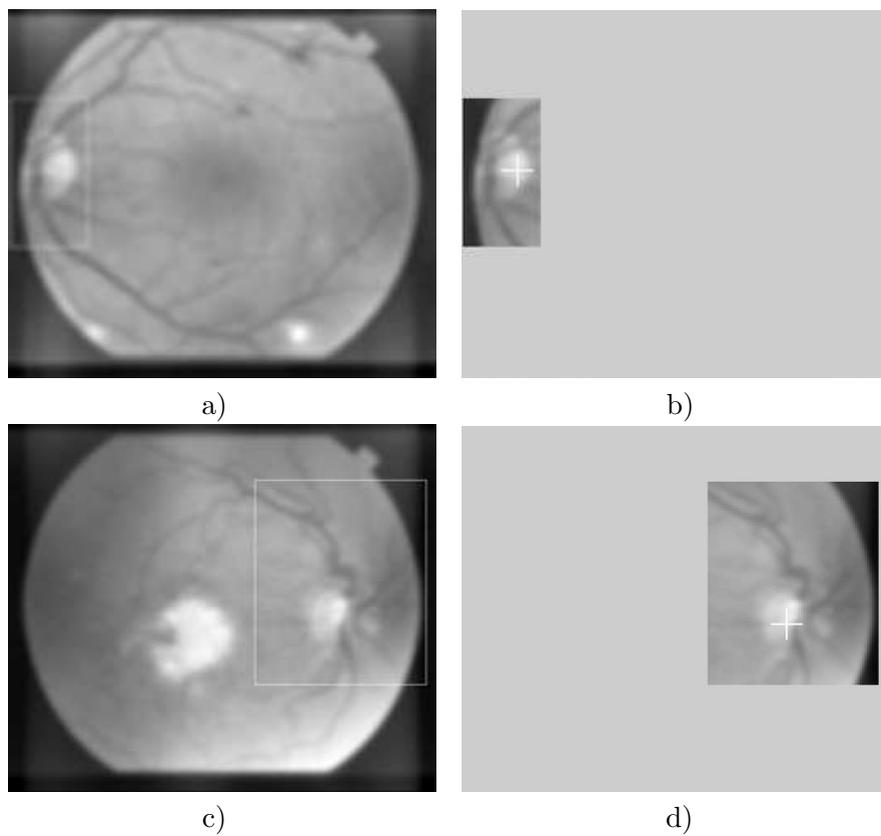


Figure 6. Results of the new window centre calculation. Green channels a), c); new window centre b), d).

3 Optic disc recognition

As in [1], [11], [12] and [13] the further work was done on a square window centred on the optic centre candidate computed previously. The searching window side is a fraction of image height. The tests have been done on the green channel of the first two sets of retinal images (86 retinal images of 720×576 size and 40 images of 2592×1728 resolutions), on chosen channel of the last set (300 of RGB retinal images of 720×576 size) from our collaborators (UMP, Iași) and on 100 images from STARE database where the optic disc is visible.

Following the same technique employed in [6] in the established window I the blood vessels were eliminated (7).

Results of the vessels erasing operation are illustrated by Figure 7, for last set of retinal images received from our collaborators (UMP, Iași).



Figure 7. The result of vessels erasing. a) Selected channel b) Cleaned working window – [13].

In order to perform a circle fitting the disc, edges have to be extracted. This is done by applying on image I_C an iterative Canny filter followed by binarisation. The same technique proposed in [12] and [13]

was employed to do this:

1. Compute a binarization threshold using Otsu method, [9], on image I_C , without performing the binarization.
2. Choose a value close to Otsu threshold as a primary threshold for Canny filtering.
3. Perform Canny filtering.
4. If there are not enough white pixels (less than a predefined threshold) adapt the threshold for Canny filtering and resume process from step 3.
5. Compute r_{\min} and r_{\max} , the minimum and maximum values of circles radius, as fractions of the original image width.
6. For an interval $[r_{\min}, r_{\max}]$ of circle radius compute a circle fitting by Hough transform applied on window pixels with grey level close to the window centre level.
7. Choose the centre radius with the best fitting score and best distribution of fitting points.
8. If the fitting score is not desirable or there are few points to perform the fitting, decrease the Canny threshold by a certain amount (constant in our implementation) and perform Canny filtering on I_C and resume the process from step 6. Do this not more than a predefined number of iterations.
9. If the detected circles have comparable fitting scores and fitting point distributions, choose the circle with the longest radius.

Canny filtering was done using the OpenCV function. Hough transform was performed by implementing our own method in order to get more control on the distribution of the fitting points [12], [13]. The distance between the current fitted circle centre and the mass centre of the fitting points was used to evaluate the point distribution. In

this way some configurations can be rejected even they are generated by an acceptable number of fitting points if the points are not equally distributed around circle centre.

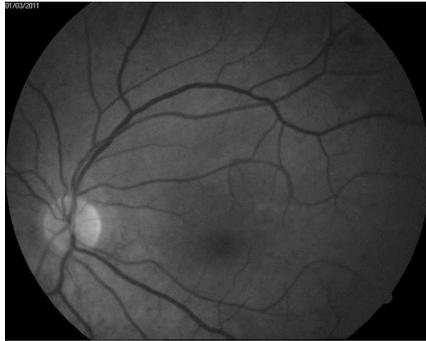
4 Results and conclusions

Tests have been done on two first sets of 86 RGB retinal images of 720×576 resolution and 40 images of 2592×1728 resolution provided by our collaborators (UMP, Iași). The method [13] to detect the optic disc area worked well, with the same results as the one presented in [12]: the rough optic disc localization has been successful on both image sets. The final circle fitting failed on two low resolution images strongly affected. Because the previous method [12] is faster we opted to keep it for the old sets and use the approach presented in [13] only for the last set of 300 retinal images of 720×576 resolution. The optic disc localization has been successful on 280 images of the last set. The final circle fitting failed on 10 images of the 280 images previously mentioned.

The last method based on vessel tree analysis was tested on the set of 300 RGB retinal images of 720×576 size provided lately by our collaborators (UMP, Iași) and on 100 images from STARE database where the optic disc is visible. The new optic disc localization has been successful on 282 images of the first set, a little bit better than the previous method [13]. However, from 100 images chosen from STARE database the method [13] failed to localize to optic disc area on 20 images while the new method was successful on 90 STARE images.

Figure 8 illustrates some final circle localization results for images from the three sets from Grigore T. Popa University of Medicine and Pharmacy Iasi and an image from STARE database.

The optic disk localization and modelling procedure was implemented and tested in an image processing framework developed by authors. It is implemented as a Windows application, in C++ using Microsoft Visual Studio. For image manipulation and some processing functions, the OpenCV library is used.



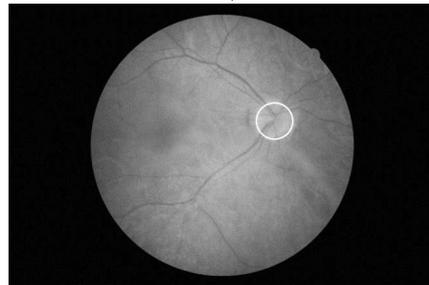
a)



b)



c)



d)



e)



f)

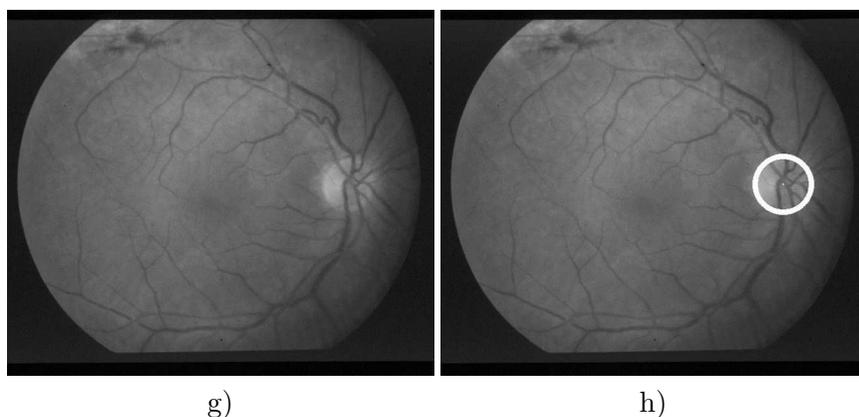


Figure 8. On the left column: original retinal images. On the right: the final optic disc localization results. a), b) image from the first set of 720×576 resolution – [11]; c), d) image from the set of 2592×1728 resolution – [12]; e), f) image from the last set of 720×576 size – [13]; g), h) image from the STARE.

Acknowledgments The work was done as part of research collaboration with Grigore T. Popa University of Medicine and Pharmacy Iasi to analyse retinal images for early prevention of ophthalmic diseases.

References

- [1] A. Aquino, M.E. Gegundez-Arias, D. Marin. *Detecting the optic disc boundary in digital fundus images using morphological, edge detection, and feature extraction techniques*, IEEE Transactions on Medical Imaging, Nov. 2010, Volume 29, Issue 11, pp.1860–1869.
- [2] B. Harangi, A. Hajdu. *Improving the accuracy of optic disc detection by finding maximal weighted clique of multiple candidates of individual detectors*, IEEE 9th International Symposium on

- Biomedical Imaging (ISBI2012), Barcelona, Spain, 2012, pp.602–605.
- [3] B. Harangi, R.J. Qureshi, A. Csutak, T. Peto, A. Hajdu. *Automatic detection of the optic disc using majority voting in a collection of optic disc detectors*, IEEE 7th International Symposium on Biomedical Imaging (ISBI 2010), Rotterdam, The Netherlands, 2010, pp.1329–1332.
- [4] H.Li, O. Chutatape. *Automated feature extraction in color retinal images by a model based approach*, IEEE Transactions on Biomedical Engineering, Vol.51, No.2, February 2004.
- [5] A. Hoover, M. Goldbaum. *Locating the optic nerve in a retinal image using the fuzzy convergence of the blood vessels*, IEEE Trans. Med. Imag., vol. 22, no. 8, pp.951–958, Aug. 2003.
- [6] C. Heneghan, J. Flynn, M. O’Keefe, M. Cahill. *Characterization of changes in blood vessel width and tortuosity in retinopathy of prematurity using image analysis*, Med. Image Anal., vol. 6, pp.407–429, 2002.
- [7] M. Foracchia, E. Grisan, A. Ruggeri. *Detection of optic disc in retinal images by means of a geometrical model of vessel structure*, IEEE Trans. Med. Imag., vol. 23, no. 10, pp. 1189–1195, Oct. 2004.
- [8] M. Lalonde, M. Beaulieu, L. Gagnon. *Fast and robust optic disk detection using pyramidal decomposition and Hausdorff-based template matching*, IEEE Trans. Medical Imaging, Vol. 20, pp. 1193–1200, Nov. 2001.
- [9] N. Otsu. *A threshold selection method from gray-level histograms*, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 9, No. 1, 1979, pp. 62–66.
- [10] S. Ravishankar, A. Jain, A. Mittal. *Automated feature extraction for early detection of diabetic retinopathy in fundus images*, CVPR – IEEE Conference on Computer Vision and Pattern Recognition, pp. 210–217, 2009.

- [11] F. Rotaru, S. Bejinariu, C.D. Niță, M. Costin. *Optic disc localization in retinal images*, 5th IEEE International Workshop on Soft Computing Applications, 23-25 August, 2012, Szeged, Hungary, Soft Computing Applications – Advances in Intelligent Systems and Computing Volume 195, 2013, Springer Verlag.
- [12] F. Rotaru, S. Bejinariu, C.D. Niță, R. Luca. *New optic disc localization method for retinal images*, IEEE International Symposium on Signal, Circuits and Systems, ISSCS 2013, 11-12 July 2013, Iasi, Romania.
- [13] F. Rotaru, S. Bejinariu, C.D. Niță, R. Luca, C. Lazăr. *New optic disc localization approach in retinal images*, The 4th IEEE International Conference on E-Health and Bioengineering, EHB 2013, 21-23 November 2013, Iasi, Romania.
- [14] A. Sopharak, K. Thet Nwe, Y. Aye Moe, M. N. Dailey, B. Uyyanonvara. *Automatic exudate detection with a naive Bayes classifier*, International Conference on Embedded Systems and Intelligent Technology, Grand Mercure Fortune Hotel, Bangkok, Thailand, pp.139–142, 2008.
- [15] G.C. Manikis, V. Sakkalis, X. Zabulis, P. Karamaounas, A. Triantafyllou, S. Douma, Ch. Zamboulis, K. Marias. *An Image Analysis Framework for the Early Assessment of Hypertensive Retinopathy Signs*, Proceedings of the 3rd IEEE International Conference on E-Health and Bioengineering - EHB 2011, 24th-26th November, 2011, Iasi, Romania.
- [16] Yanhui Guo. *Computer-Aided Detection of Breast Cancer Using Ultrasound Images*, PhD Thesis, Utah State University, 2010.
- [17] D. Welfer, J. Scharcanski, C.M. Kitamura, M.M. DalPizzol. *Segmentation of the optic disk in color eye fundus images using an adaptive*, Comput Biol Med, vol. 40, no. 2, pp. 124–137, 2010.

- [18] K.W. Tobin, E. Chaum, V.P. Govindasamy, T.P. Karnowski. *Detection of anatomic structures in human retinal imagery*, IEEE Transactions on Medical Imaging 26 (12) (2007), pp.1729–1739.
- [19] M. Park, J.S. Jin, S. Luo. *Locating the optic disc in retinal images*, Proceedings of the International Conference on Computer Graphics, Imaging, and Visualisation, IEEE, Sydney, Australia, 2006, pp.14–145.
- [20] C. Sinthanayothin, J.F. Boyce, H.L. Cook, T.H. Williamson. *Automated localisation of the optic disc, fovea, and retinal blood vessels from digital colour fundus images*, British Journal of Ophthalmology 83 (1999), pp.902–910.
- [21] A.A.-H.A.-R. Youssif, A.Z. Ghalwash, A.A.S.A.-R. Ghoneim. *Optic disc detection from normalized digital fundus images by means of a vessels' direction matched filter*, IEEE Transactions on Medical Imaging 27 (1) (2008) pp.11–18.
- [22] J. Lowell, A. Hunter, D. Steel, A. Basu, R. Ryder, E. Fletcher, L. Kennedy. *Optic nerve head segmentation*, IEEE Transactions on Medical Imaging 23 (2) (2004), pp.256–264.
- [23] M. Niemeijer, M.D. Abramoff, B.V. Ginneken. *Segmentation of the optic disc macula and vascular arch in fundus photographs*, IEEE Transactions on Medical Imaging 26 (2007), pp.116–127.
- [24] A.D. Fleming, K.A. Goatman, S. Philip, J.A. Olson, P.F. Sharp. *Automatic detection of retinal anatomy to assist diabetic retinopathy screening*, Physics in Medicine and Biology 52 (2) (2007), pp.331–345.
- [25] E.J. Carmona, M. Rincon, J. Garcia-Feijoo, J.M.M.de-la Casa. *Identification of the optic nerve head with genetic algorithms*, Artificial Intelligence in Medicine 43 (3) (2008), pp.243–259.
- [26] C.A. Lupascu, D. Tegolo, L.D. Rosa. *Automated detection of optic disc location in retinal images*, in: 21st IEEE International Sym-

posium on Computer-Based Medical Systems, IEEE, University of Jyvaskyla, Finland, 2008, pp. 17–22.

- [27] T. Walter, J.-C. Klein, P. Massin, A. Erginay. *A contribution of image processing to the diagnosis of diabetic retinopathy—detection of exudates in color fundus images of the human retina*, IEEE Transactions on Medical Imaging, 21 (10) (2002), pp.1236–1243.
- [28] A. Sopharak, B. Uyyanonvara, S. Barman, T.H. Williamson. *Automatic detection of diabetic retinopathy exudates from non-dilated retinal images using mathematical morphology methods*, Computerized Medical Imaging and Graphics, 32 (2008), pp.720–727.

Florin Rotaru, Silviu Ioan Bejinariu,
Cristina Diana Niță, Ramona Luca, Camelia Lazăr

Received June 2, 2014

Institute of Computer Science, Romanian Academy,
Iasi Branch, Romania

E-mails: {*florin.rotaru, silviu.bejinariu, cristina.nita, ramona.luca, camelia.lazar*}@
iit.academiaromana-is.ro

Characterization and pattern recognition of
color images of dermatological ulcers:
a pilot study*

Invited Article

Lucas C. Pereyra, Sílvia M. Pereira, Juliana P. Souza,
Marco A. C. Frade, Rangaraj M. Rangayyan,
Paulo M. Azevedo-Marques

Abstract

We present color image processing methods for the characterization of images of dermatological lesions for the purpose of content-based image retrieval (CBIR) and computer-aided diagnosis. The intended application is to segment the images and perform classification and analysis of the tissue composition of skin lesions or ulcers, in terms of granulation (red), fibrin (yellow), necrotic (black), callous (white), and mixed tissue composition. The images were analyzed and classified by an expert dermatologist following the red-yellow-black-white model. Automatic segmentation was performed by means of clustering using Gaussian mixture modeling, and its performance was evaluated by deriving the Jaccard coefficient between the automatically and manually segmented images. Statistical texture features were derived from cooccurrence matrices of RGB, HSI, $L^*a^*b^*$, and $L^*u^*v^*$ color components. A retrieval engine was implemented using the k-nearest-neighbor classifier and the Euclidean, Manhattan, and Chebyshev distance metrics. Classification was performed by

©2014 by Lucas C. Pereyra, Sílvia M. Pereira, Juliana P. Souza, Marco A. C. Frade, Rangaraj M. Rangayyan, Paulo M. Azevedo-Marques.

*This work was partially supported by The National Council for Scientific and Technological Development (CNPq) — grants 472508/2010-5, 304225/2010-0, and 573714/2008-8 (INCT/INCoD), and the Natural Sciences and Engineering Research Council of Canada.

means of a metaclassifier using logistic regression. The average Jaccard coefficient after the segmentation step between the automatically and manually segmented images was 0.560, with a standard deviation of 0.220. The performance in CBIR was measured in terms of precision of retrieval, with average values of up to 0.617 obtained with the Chebyshev distance. The meta-classifier yielded an average area under the receiver operating characteristic curve of 0.772.

Keywords: Color image processing, color medical images, color texture, content-based image retrieval, computer-aided diagnosis, image segmentation, dermatological ulcers, tissue composition analysis.

1 Introduction

1.1 Computer-Aided Analysis of Medical Images

Recent developments in information and communication technologies have led to the creation and use of huge archives of multimedia data in diverse application areas, such as medical imaging, remote sensing, entertainment, education, and online information services. Traditional database management systems were designed to organize alphanumeric data into interrelated collections so that information storage and retrieval could be performed conveniently and efficiently. However, such methods are not well suited for the organization, management, and efficient use of multimedia information [1]. In many practical applications, the retrieval of a specific image from a database of images could be an important task [2]. Several search engines are available for searching and retrieval of textual and pictorial information [2–5]. However, general-purpose tools for searching and retrieval of data are not suitable for specialized medical applications. For this reason, specific methods and systems for the characterization, searching, and retrieval of image-based data are being developed for particular medical imaging applications [6, 7], such as mammography [8–11], chest and other radiographic imaging [12, 13], dermatological lesions [14–17], and pathology [18].

The traditional approach of indexing images using manual notes (textual annotation) is slow, labor-intensive, and expensive. In addition, textual annotations cannot effectively encode all of the information available in a given image. Furthermore, image features based on huge amounts of pixel data, complex concepts of application-specific patterns, and domain-specific notions may not lend themselves to easy or efficient textual description. Thus, there is a need to develop advanced methods of image processing, feature extraction, quantitative representation, and pattern recognition for effective and efficient indexing and retrieval of images based on their content.

Content-based image retrieval (CBIR) refers to searching, selection, and retrieval of images from a database that are similar to a query image, using measures of information derived from the images themselves, rather than relying on the accompanying text or annotation [6,19]. To facilitate CBIR, the contents of the images need to be characterized by quantitative features. The features of the query image may then be compared with the features of each image in the database, and images having high similarity with respect to the computed features of the query image may be retrieved and displayed [19]. CBIR of medical images is a useful tool, and could provide physicians with assistance in the form of a display of relevant past cases with proven pathology, along with the associated clinical, diagnostic, and other information [19].

The potential use of automated image categorization techniques to assist physicians in diagnosis led to intense research in the field of medical image processing classification [19–21]. This approach usually consists of mapping images into predefined classes and involves the steps of representation (description of image content by feature extraction), adaptation (selection of the most representative subset of features to classify information) and, generalization (the training and evaluation of a classifier) [19,21,22].

The same steps mentioned above may also be used to assist in clinical decision making and lead towards computer-aided diagnosis (CAD) [19,21,23]. The medical domain has specific requirements (high sensitivity and low false-negative as well as false-positive rates) and errors in detection are extremely costly, making the development of a

computerized scheme to aid diagnosis with broad application in clinical practice a challenge [19, 20].

1.2 Analysis of Dermatological Ulcers

Ulcers on the lower limbs can be described as the irregular loss of the epidermis, also possibly involving the dermis and subcutaneous tissue. They affect approximately 1% of the population, causing considerable morbidity [24]. This condition is usually ascribed to a deficiency in blood flow due to venous or arterial insufficiency, and can be caused by ailments such as diabetes mellitus, autoimmune diseases, and local infections. The ulceration may be referred to as a wound, lesion, or ulcer, and medical professionals in dermatology base the diagnosis of skin lesions mainly on visual assessment of pathological regions and the evaluation of macroscopic features. This fact indicates that correct diagnosis is highly dependent on the observer's experience and visual perception [16, 25]. Accurate wound assessment is a critical task in patient care and important for the reduction of costs of care in hospitals. However, this task still relies on manual procedures and tedious practices. Wound shape is measured with rulers and tracing paper, or rarely with alginate castings and serum injection.

Healing is a complex cascade of cellular events operating to reconstruct damaged tissues, and also an individual process that exhibits considerable interpatient variability. As the different tissues may overlap and be difficult to distinguish, wound assessment is not straightforward. The lack of quantitative data affects the coordination of health-care staff and hinders clinical studies focused on healing. Digital cameras, though now widespread in clinical centers, are used mostly for basic patient data recording and not image processing [17].

The appearance of a wound, lesion, or ulcer provides important clues that can help with the diagnosis, determination of severity, and the prognosis of healing [26, 27]. Chronic skin lesions, wounds, or ulcers typically have a nonuniform mixture of red granulation, yellow fibrin (slough), and black necrotic scar tissue. Thus, a red-yellow-black (RYK) model is used by physicians in a descriptive manner [15, 17, 26–29]. In clinical assessment, the category of callous lesions,

composed predominantly of white tissue, is also used. Therefore, an extended version of the RYK model, including white (RYKW) one, may be of interest. The quantification of texture and color distribution in lesions by image processing techniques could assist in the analysis of the dynamics of the pathological process, as well as of healing and response to treatment [17, 27, 29, 30]. Such quantitative analysis, in turn, can be used to design optimized and personalized treatment protocols for each patient.

In a clinical scenario, the possibility of retrieving images from an established database that are similar to the case on hand, based on digital image processing techniques that characterize the color composition of lesions, could facilitate understanding how dermatological lesions are classified and assist in arriving at a diagnostic or therapeutic decision. In this context, efficient systems for content-based indexing and retrieval [6, 27, 31], as well as methods to perform classification of wounds based on the types of tissue present in the wound are useful and could help health professionals assess lesions for any of the aforementioned purposes.

Celebi et al. [32] described a system to retrieve skin lesion images based on shape similarity, using a database with 184 skin ulceration images in cases of melanoma. The agreement between computer assessment and human perception was indicated by values of up to 0.73, when similarity functions were optimized using a genetic algorithm. Dorileo et al. [14] proposed a CBIR system for images of dermatologic ulcers. Features based on histogram and multispectral cooccurrence matrices were used for image retrieval. Performance was evaluated based on precision values with a database of 215 images. The best precision result was 70% for mixed tissue composition images. Rahman et al. [33] presented a CBIR system for dermatoscopic images. Image processing, segmentation, feature extraction (color and texture), and similarity matching were performed on a database comprising 358 images of pigmented skin lesions in three categories (benign, dysplastic nevi and, melanoma). The analysis of precision curves displayed the ability of their system to retrieve visually similar lesions with an average precision value of 60%.

The variability and inhomogeneity of tissues makes color analysis for tissue classification ineffective if the methods are applied directly on the pixels with simple thresholds on separate color components. Spatial continuity needs to be incorporated, which suggests that the classification process should be guided by a segmentation step [34]. Color histograms are commonly utilized as descriptors for statistical data analysis [35], and the use of hybrid tissue classes and limiting classification to a reduced number of tissue types have also been reported, with an average accuracy of 88.7% using a k-nearest-neighbor (kNN) classifier [29].

In previous related work, we evaluated the performance of different distance metrics for CBIR of dermatological images [31]. A database of 172 manually segmented images was used. CBIR using a kNN classifier and the Euclidean, Manhattan, Chebyshev, cosine, and correlation distances was performed. The best results were obtained with the cosine and correlation distances, with average precision of 75%. We also performed automatic segmentation using methods of color clustering and mathematical morphology [36]. The same database of 172 images was used. The average Jaccard coefficient between automatically and manually segmented regions was 0.56, with a standard deviation of 0.22.

The present work, which is an expanded and updated version of recent conference presentations [31, 36], focuses on quantitative assessment of color components in images of ulcers on the lower limbs. We present techniques to perform analysis of color components, texture analysis, and automatic segmentation. Results are presented and discussed in the context of CBIR and computer-aided diagnosis CAD [15–17, 27, 28, 30, 31, 36].

2 Materials and Methods

2.1 Database of Images

A database consisting of 172 dermatologic images has been prepared to date, based on 63 consecutive medical examinations of outpatients

at the University Medical Center at the Ribeirão Preto Medical School of the University of São Paulo, Brazil. Approval was obtained from the Medical Center Ethics Committee for this research. Images were obtained based on a specific protocol that was determined after initial tests [14]. All images were obtained with the same digital camera (Canon EOS 5D0, 2 Megapixels), a 50-mm macro lens, and a polarization filter; see Figure 1 for examples of images of various types of ulcers. The typical size of the color images is 1747×1165 pixels with 24 bits/pixel. The tissue composition of each lesion was classified independently by an expert dermatologist (MACF), based on the color composition, as granulation (red), fibrin (yellow), and mixed tissue. The 172 images in the database include 51 images of lesions predominantly composed of granulation, 31 images of fibrin, three images of callous, three images of necrotic, and 84 images of mixed tissue. The dermatologist also drew the boundaries of the lesions.

A blue cloth was used to create a background in a color not expected within the ulcer or on the part of the body being imaged, as can be seen in Figure 1, parts (c) and (d). Color patches and rulers were included in the images (see Figure 1) to facilitate color normalization and calibration of the images. A suite of color image processing techniques, pattern analysis, classification methods, and graphical user interfaces (GUIs) is being developed to facilitate image analysis, CAD, and CBIR [14, 27, 28, 31, 37].

2.2 Automatic Segmentation of the Ulcer Region

A color clustering process using an expectation maximization (EM) procedure based on a multivariate Gaussian mixture model [37, 38] was applied to each hue-saturation (HS) histogram [39]. The EM algorithm implemented in the present work is an iterative method for finding maximum-likelihood or maximum-a-posteriori (MAP) estimates of parameters in statistical models. The EM iteration alternates between performing an expectation (E) step, which creates a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters, and a maximization (M) step, which computes param-

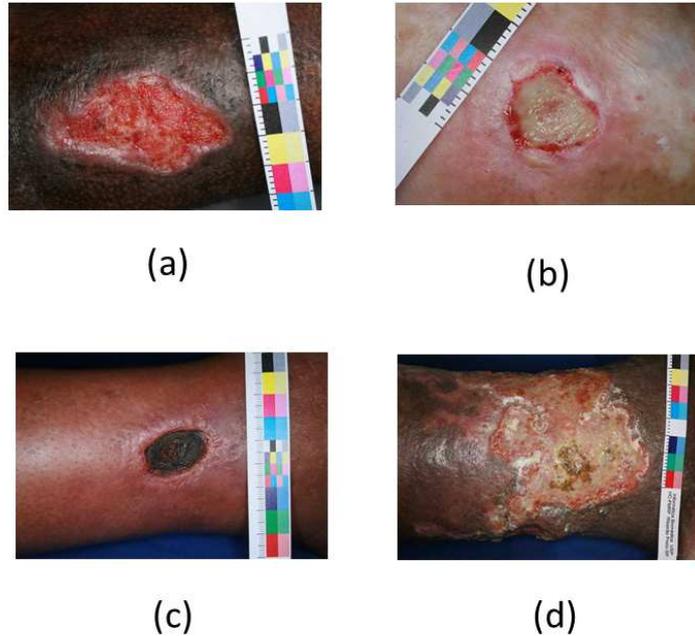


Figure 1. Examples of images of various types of ulcers: (a) predominantly granulation, (b) predominantly fibrin, (c) predominantly necrotic tissue, and (d) mixed tissue composition.

eters maximizing the expected log-likelihood found in the E step. The initialization assumes that the whole set of values in the HS histogram can be clustered into just one Gaussian distribution. This hypothesis is tested based on the values of the mean and standard deviation of a sample of 1,000 values obtained randomly from the HS histogram, considering an estimation error of 0.0001%. If the hypothesis is not true, that is, if the Gaussian adjustment error for the sample is greater than the specified error, a new iteration is performed, assuming the mixture of two Gaussians. The procedure is iterated until every pixel in the sample belongs to a cluster, or the number of iterations reaches a specified limit (100 cycles). The clustering procedure was developed using the Waikato Environment for Knowledge Analysis (WEKA) [41].

After color clustering, the set of clusters representing the tissue composition of the ulcer was manually selected. A GUI was implemented using the Java programming language and the NetBeans¹ environment to facilitate cluster selection. To fill any residual gap, a method based on mathematic morphology and automatic delineation of the convex hull was implemented using ImageJ plugins and applied to determine the final lesion area. The convex hull is defined as the smallest convex polygon that encompasses a set of points [42]. Figure 2 shows an example of an image of an ulcer and the corresponding result of segmentation.

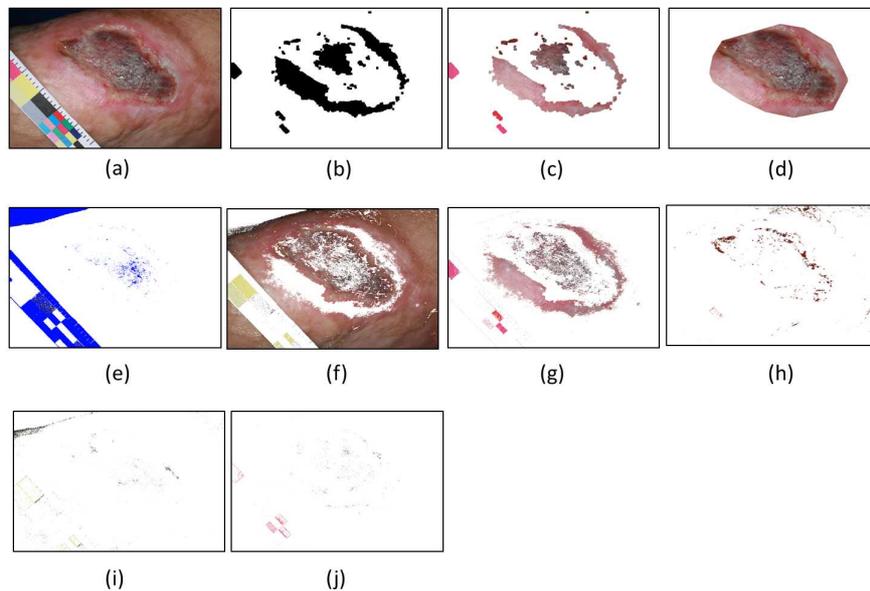


Figure 2. Example of ulcer segmentation. The clustering process resulted in six clusters (e) to (j). (a) shows the original image, (b) the segmentation mask generated by manual cluster selection followed by application of mathematic morphology operations, (c) is the partial segmented area, and (d) is the final lesion area after automatic delineation of the convex hull.

¹<https://netbeans.org/>. Accessed on 01/04/2014.

2.3 Feature Extraction and Indexing of Images

Feature extraction was based on the automatically segmented regions. For each region representing a lesion, in addition to the basic RGB color components, six images were generated. According to the HSI representation, an image (I) was generated based on the hue (H) component and another was generated based on the saturation (S) component [39]:

$$I = \frac{R+G+B}{3},$$

$$S = 1 - \left(\frac{3}{(R+G+B)} \right) * a,$$

where a is the minimum of R , G , and B ,

$$H = \cos^{-1} \frac{(0.5*(R-G)+(R-B))}{\left(((R-G)^2+(R-B)(G-B))^{0.5} \right)}.$$

The next color representations are based on the CIE XYZ color space. The conversion from RGB to XYZ is performed as follows [40]:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.240479 & -1.537150 & -0.498535 \\ -0.969256 & 1.875992 & 0.041556 \\ 0.055648 & -0.204043 & 1.057311 \end{bmatrix} * \begin{bmatrix} X \\ Y \\ Z \end{bmatrix},$$

$$x = \frac{X}{(X+Y+Z)},$$

$$y = \frac{Y}{(X+Y+Z)}.$$

According to the $L^*u^*v^*$ color representation, an image was generated based on the u^* component and another was generated based on the v^* component:

$$u' = \frac{2x}{(6y-x+1.5)},$$

$$v' = \frac{4.5y}{(6y-x+1.5)},$$

$$L^* \begin{cases} 116 \left(\frac{Y}{Y_n} \right)^{\frac{1}{3}} - 16 & \text{if } \frac{Y}{Y_n} > 0.008856, \\ 903.3 \left(\frac{Y}{Y_n} \right) & \text{if } \frac{Y}{Y_n} \leq 0.008856, \end{cases}$$

$$u^* = 13(L^*)(u' - u'_n),$$

$$v^* = 13(L^*)(v' - v'_n).$$

Similarly, according to the L*a*b* color representation, an image was generated based on the a* component and another was generated based on the b* component:

$$L^* \begin{cases} 116 \left(\frac{Y}{Y_n} \right)^{\frac{1}{3}} - 16 & \text{if } \frac{Y}{Y_n} > 0.008856, \\ 903.3 \left(\frac{Y}{Y_n} \right) & \text{if } \frac{Y}{Y_n} \leq 0.008856, \end{cases}$$

$$a^* = 500 * (f(X/X_n) - f(Y/Y_n)),$$

$$b^* = 200 * (f(Y/Y_n) - f(Z/Z_n)),$$

$$\text{where } f(t) = \begin{cases} t^{\frac{1}{3}} & \text{if } t \leq 0.008856, \\ 7.787 * t + 16/116 & \text{if } t > 0.008856. \end{cases}$$

Values of the mean, standard deviation, skewness, and kurtosis were computed from the histogram of each of the R, G, B, H, S, u*, v*, a*,

and b^* components. In addition, the five most discriminative features of the 14 texture features proposed by Haralick et al. [43] were derived from an averaged cooccurrence matrix (CoM) computed from four CoMs for a distance of one pixel at 0, 45, 90, and 135 degrees. The features are homogeneity, contrast, correlation, entropy, and local homogeneity, as suggested by Connors and Harlow [44], and were computed for each of the R, G, B, H, S, u^* , v^* , a^* , and b^* components. Furthermore, the five texture features mentioned above were computed from multispectral or color cooccurrence matrices (CCMs), obtained from the RG, GB, BR, HS, u^*v^* , and a^*b^* components, using the method proposed by Arvis et al. [45]. The method is an extension of the method of Haralick et al. [43], and was developed for application to color images to take into account the correlation existing between the color components, as shown in Figure 3. Thus, a total of 111 features were extracted from the R, G, B, H, S, u^* , v^* , a^* , and b^* components to characterize and index each color image [27, 31, 36].

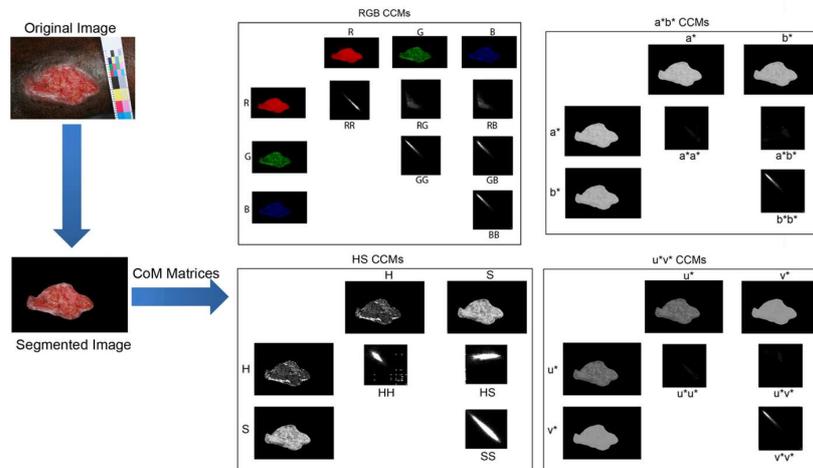


Figure 3. Color cooccurrence matrices (CCMs) obtained from the RGB, HS, u^*v^* , and a^*b^* components

2.4 CBIR Experiment

A retrieval engine was developed using the kNN classifier based on the Euclidean, Manhattan, and Chebyshev metrics between the features of the query image and those of each image in the database. The definitions of the metrics are summarized in Table 1. The value of k was varied from 1 to 7 in steps of 2.

Table 1. Distance metrics used in CBIR experiments. In the equations, x is the feature vector of size n of the query image, y the feature vector of size n of an image being considered for retrieval, and d_{xy} is the distance between the two vectors.

Metric Name	Formula
Euclidean	$d_{xy} = \sqrt{\sum_{j=1}^n (x_j - y_j)^2}$
Manhattan	$d_{xy} = \sum_{j=1}^n x_j - y_j $
Chebyshev	$d_{xy} = \max_{j=1}^n \{ x_j - y_j \}$

For every value of k , all images in the database were used, one by one, as the query image. The mean value of the precision of retrieval was computed for each retrieval experiment and for each distance or metric. Precision values for the categories of necrotic and callous ulcers were not computed due to the small numbers of samples available. The precision of retrieval was computed in each experiment as the ratio of the number of relevant images retrieved to the total number of images retrieved. In computing precision, each retrieved image received a binary weight of unity or zero, representing a relevant image or not, respectively; the sum of such scores for all of the retrieved images was then divided by the total numbers of retrieved images. Ten-fold cross validation was the chosen validation method. An image was considered to be relevant when it belonged to the same class as the query image, according to the classification provided independently by the expert dermatologist (MACF). It should be noted that, in a medical application, a clinician would not be interested in the retrieval of a large number of images or cases. A small number of highly relevant

cases, such as 3 or 5 cases, retrieved from a large database along with the related clinical reports, would serve the purpose of assisting in the diagnosis of the current query case. It would not be of interest to retrieve all relevant cases that exist in the database. Thus, in a clinical application, precision would be, in general, more important than other measures of performance of CBIR, such as recall or area under the precision-versus-recall curve.

2.5 Classification Experiment

Classification of images was performed by means of a metaclassifier for handling multiclass datasets with two-class classifiers, as a proxy to the main classifier which it wraps to provide additional data preprocessing before actually training and/or testing of the wrapped classifier. Logistic regression was used as the wrapped classifier. In order to improve the parameters of the estimates and reduce the prediction error, ridge estimators are used to pool highly correlated covariates and reduce overfitting and colinearity [46]. The classification experiment did not include the categories of necrotic and callous ulcers due to the small number of samples available. Validation was also performed using ten-fold cross validation.

3 Results

3.1 Automatic Segmentation of the Ulcer Region

The Gaussian mixture modeling procedure applied to 172 images resulted in a clustering process with the number of clusters varying from 3 to 9, before the manual selection step using the GUI. Processing times for the clustering process ranged from 3 to 5 minutes per image. After cluster selection, the evaluation of the segmentation step indicated an average Jaccard coefficient of 0.56 with a standard deviation of 0.22 between the lesion area obtained computationally and the same lesion region manually delineated by the dermatologist. The Jaccard coefficient (J) is calculated by computing the ratio of the intersection

to the union of the manually and computationally derived regions and has a value range from zero to one, where zero means a total failure of the computational segmentation process and one indicates a perfect segmentation, based on comparison with the manually segmented regions:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}.$$

3.2 Analysis of Tissue Composition and Classification of Ulcers

To compare the tissue composition within the lesion regions segmented computationally and manually, a measurement approach was applied based on thresholding of HSI values [31,36,39]. White tissue component was defined as the number of pixels within the lesion with $S \leq 0.2$ and $I \geq 0.75$ (independent of H and with the S and I components normalized to the range 0 to 1); black tissue component was defined as the number of pixels within the lesion with $S \leq 0.2$ and $I \leq 0.2$; red tissue component was defined as the number of pixels within the lesion with $S > 0.2$ and $-30^\circ < H < 30^\circ$ (independent of I); and yellow tissue component was defined as the number of pixels within the lesion with $S > 0.2$ and $30^\circ < H < 90^\circ$. A low average root-mean-squared error (RMSE) [36] of 4% with a standard deviation of 5% was obtained between the RYKW tissue composition vectors of computationally and manually segmented lesions.

Classification results with the metaclassifier using logistic regression were analyzed in terms of correctly classified instances and the average area under the receiver operating characteristic curve (AUC). The ratio between the number of correctly classified instances and the total number of instances was 0.638, and the average AUC value was 0.772. The processing time using all the images from the dataset was less than 3 seconds.

3.3 CBIR

Based on the retrieval experiments performed using different distance measures, precision values were calculated for each image class and an average precision value was derived. The Processing time was less than 1 second for all CBIR experiments. The most significant distance regarding precision values was the Chebyshev distance. For $k = 1$, the average precision was 0.617 (as compared to 0.587 and 0.550 using Manhattan and Euclidean distances, respectively), and for $k = 3$, the average precision was 0.609 (as compared to 0.566 and 0.538 using Manhattan and Euclidean distances, respectively). The rates of precision for values of $k = 5$ and 7 were lower than those previously mentioned, for all of the distances considered in this study. Tables 2 and 3 display the confusion matrices for the kNN procedure with $k = 1$ and 3 and using the Chebyshev distance, respectively.

Table 2. Confusion matrix for the CBIR experiment, using a kNN classifier with $k = 1$ and the Chebyshev distance.

Red	Yellow	Mixed	Retrieved for
31	3	17	Red
3	20	8	Yellow
26	8	50	Mixed

Table 3. Confusion matrix for the CBIR experiment, using a kNN classifier with $k = 3$ and the Chebyshev distance.

Red	Yellow	Mixed	Retrieved for
30	5	16	Red
9	15	7	Yellow
24	6	54	Mixed

4 Discussion

Assessment of tissue composition provides crucial information to monitor the effects of treatment in patients with chronic ulcers. Quantitative measures can contribute to objective assessment of the healing process, and may be used for pattern recognition, CAD, and CBIR. Pattern recognition focuses on the classification of ulcers based on tissue composition to help enhance the diagnostic process and follow-up to treatment. CBIR is an approach to information retrieval and is more commonly used for medical decision-making based on previous and proven cases. If the results of a CBIR query bring images with one or more types of tissue composition, the clinician will be able to analyze them, determine which images are relevant, review the clinical reports associated with the retrieved images, and decide on the diagnostic classification of the image on hand. The clinician could also gain an understanding on why some of the retrieved images were classified into certain categories, either by an expert dermatologist or by the CBIR process. In this manner, the CBIR system could assist in arriving at a decision when the image on hand is at the boundaries between different categories. The concept of similarity is important because no two images may be expected to be identical, even when belonging to the same diagnostic category, and a perfect or exact match to a query image is unlikely. The use of vectors of quantitative parameters or features to index images facilitates the application of simple distance measures to select images that are most similar to the query sample, that is, to evaluate the similarity between images. Classification and CBIR applications may share the same set of image processing procedures, such as preprocessing, image segmentation, feature extraction, and categorization [11, 19, 21].

In a previous work based on the same images of manual segmentation of the ulcers by an experienced dermatologist [27], we obtained average AUC values of 0.820, and an average value of correctly classified instances divided by the total number of instances of 0.738. Results using automatic segmentation were observed to be poorer than those using manually segmented images. Therefore, our efforts to perform

automatic segmentation have yielded poorer results in terms of correct classification, which indicates that further improvements are needed. Nonetheless, our results of classification using a metaclassifier (average AUC value of 0.772) are encouraging.

Limitations of this work reside in the difficulty in applying the imaging protocol in a consistent manner. The images were not acquired in laboratory facilities, but in a clinical environment. In some cases, the patient's mobility affected positioning and imaging of the ulcers. Such difficulties also affected the distance and orientation of the camera, the illumination of the ulcer and the composition of the image. Procedures for color correction and normalization of images [39] need to be incorporated.

Another difficulty related to the current database is the variation of the size and position of the ulcers. It was observed that the clustering procedure tends to give better results when the ulcers are small, properly centered in the image, and without significant curvature. In such a situation, the number of Gaussians obtained by the clustering procedure was usually small, in the range of 3 to 5.

The segmentation results obtained in this pilot study have lower accuracy than desired, but are encouraging. Although the results are not very good in terms of the Jaccard coefficient, the small value obtained for the RMSE indicates that the estimation of tissue composition was not substantially affected by the limitations of the segmentation process. To improve the segmentation results, it would be desirable to include procedures for comparative analysis of an ulcer region with reference to the color characteristics of the surrounding normal skin of the patient. Further work is planned with a larger database of images, especially for the callous and necrotic categories of lesions, and also including longitudinal series of images of the same patients under treatment.

Clinical interpretation of images of dermatological ulcers is commonly based on visual analysis of the tissue composition as indicated by their color characteristics. This, however, is a qualitative approach that is affected by interobserver and intraobserver variability. Accurate estimation of the fractional composition of an ulcer in terms of tissue

types, such as granulation and fibrin, is nearly impossible via visual analysis. Quantitative evaluation of tissue composition provides key information for monitoring the response to treatment of patients with chronic ulcers and can assist in the evaluation of the healing process. In the present study, we have proposed the potential use of methods of digital image processing to achieve image segmentation focused on facilitating the characterization of the tissue composition of skin ulcers. We believe that objective analysis of color images of skin ulcers using the proposed methods can overcome some of the limitations of visual analysis and lead to the development of improved protocols for the treatment and monitoring of chronic dermatological lesions. Such advances, in turn, can assist in the design of optimized and personalized therapy for each patient.

References

- [1] T. Chaira, A. K. Ray. *Fuzzy measures for color image retrieval*. Fuzzy Sets and Systems, 150:545–560, 2005.
- [2] P. Welter, T. Deserno, B. Fischer, R. Gunther, C. Spreckelsen. *Towards case-based medical learning in radiological decision making using content-based image retrieval*. BMC Medical Informatics and Decision Making. 11(1):68:1–16, 2011.
- [3] J. Z. Wang, J. Li, G. Wiederhold. *SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries*. IEEE Transactions on Pattern Analysis and Machine Intelligence. 23(9):947–963, 2001.
- [4] Q. Iqbal, J.K. Aggarwal. *CIRES: A system for content-based retrieval in digital image libraries*. International Conference on Control, Automation, Robotics and Vision (ICARCV), Singapore, 1:205-210, December 2–5, 2002.
- [5] J. Ruiz-del-Solar, P. Navarrete. *FACERET: An interactive face retrieval system based on self-organizing maps*. In Image and Video

- Retrieval - Lecture Notes in Computer Science, Springer, Berlin, Germany, 45–64, 2002.
- [6] H. Müller, N. Michoux, D. Bandon, A. Geissbühler. *A review of content-based image retrieval systems in medical applications: clinical benefits and future directions*. International Journal of Medical Informatics, 73:1–23, 2004.
- [7] H. Müller, A. Rosset, J. Vallée, F. Terrier, A. Geissbühler. *A reference data set for the evaluation of medical retrieval systems*. Computerized Medical Imaging and Graphics. 28:295–305, 2004.
- [8] I. El-Naqa, Y. Yang, N. P. Galatsanos, R. M. Nishikawa, M. N. Wernick. *A similarity learning approach to content-based image retrieval: application to digital mammography*. IEEE Transactions on Medical Imaging. 23(10): 1233–1244, 2004.
- [9] S. K. Kinoshita, P. M. Azevedo-Marques, R. R. Pereira Jr., J. A. H. Rodrigues, R. M. Rangayyan. *Content-based retrieval of mammograms using visual features related to breast density patterns*. Journal of Digital Imaging. 20(2): 172–190, 2007.
- [10] D. Tahmoush. *CBIR for mammograms using medical image similarity*. Medical Imaging 2010: Advanced PACS-based Imaging Informatics and Therapeutic Applications. Edited by B. J. Liu, W. W. Boonn. Proceedings of the SPIE. 7628(1):76280A 1–9, 2010.
- [11] H. Alto, R.M. Rangayyan, J. E. L. Desautels. *Content-based retrieval and analysis of mammographic masses*. Journal of Electronic Imaging. 14(2):023016:1–17, 2005.
- [12] C. R. Shyu, C. E. Brodley, A.C. Kak, A. Kosaka, A. M. Aisen, L. S. Broderick. *ASSERT: A physician-in-the-loop content-based image retrieval system for HRCT image databases*. Computer Vision and Image Understanding (Special Issue on Content-based Retrieval from Image Databases). 75(1/2): 111–132, 1999.

- [13] M. C. Oliveira, W. Cirne, P. M. Azevedo-Marques. *Towards applying content-based image retrieval in the clinical routine*. Future Generation Computer Systems. 23:466–474, 2007.
- [14] E. A. G. Dorileo, M. A. C. Frade, A. M. Roselino, R. M. Rangayyan, P. M. Azevedo-Marques. *Color image processing and content-based image retrieval techniques for the analysis of dermatological lesions*. In: Proc. 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Vancouver, BC, Canada. pages 1230–1233, August, 2008.
- [15] L. Ballerini, X. Li, R. B. Fisher, J. Rees. *A query-by-example content-based image retrieval system of non-melanoma skin lesions*. In: Proc. International Conference on Medical Image Computing and Computer Assisted Intervention, London, UK, 5853:31–38, September, 2009.
- [16] H. H. W. J. Bosman, N. Petkov, M. Jonkman. *Comparison of color representations for content-based image retrieval in dermatology*. Skin Research and Technology. 16:1–5, 2009.
- [17] H. Wannous, S. Treuillet, Y. Lucas. *Robust tissue classification for reproducible wound assessment in telemedicine environments*. Journal of Electronic Imaging. 19(2):023002, 2010.
- [18] J. Z. Wang. *Pathfinder: multiresolution region-based searching of pathology images using IRM*. In Proceedings of the AMIA Symposium, Los Angeles, CA, pp. 883887, November 4–8, 2000.
- [19] P. M. Azevedo-Marques, R.M. Rangayyan. *Content-based Retrieval of Medical Images: Landmarking, Indexing and, Relevance Feedback*. Synthesis Lectures on Biomedical Engineering. Lecture #48, Morgan & Claypool, 2013.
- [20] M. Antoine, O. R. Zaiane, A. Coman. *Application of data mining techniques for medical image classification*. Proceedings of the Second International Workshop on Multimedia Data Mining, MDM/KDD, 2001:94–101, San Francisco, CA, August 26, 2001.

- [21] R. M. Rangayyan. *Biomedical Image Analysis*, CRC Press, Boca Raton, FL, 2005.
- [22] T. M. Lehman, M. O. Güld, T. Deselaers, D. Keysers, H. Schubert, K. Spitzer, H. Ney, B. B. Wein. *Automatic categorization of medical images for content-based image retrieval and data mining*. *Computerized Medical Imaging and Graphics*. 29:143–155, 2005.
- [23] K. Doi. *Computer-aided diagnosis in medical images: Historical review, current status and future potential*. *Computerized Medical Imaging and Graphics*. 31:198–211, 2007.
- [24] M. J. Callam, C. V. Ruckley, D. R. Harper, J. J. Dale. *Chronic ulceration of the leg: extent of the problem and provision of care*. *British Medical Journal (Clinical Research Edition)*. 290(6485):1855–1856, 1985.
- [25] I. Maglogiannis, S. Pavlopoulos, D. Koutsouris. *An integrated computer supported acquisition, and characterization system for pigmented skin lesions in dermatological images*. *IEEE Transactions on Information Technology in Biomedicine*. 9:86–98, March, 2005.
- [26] H. Oduncu, A. Hoppe, M. Clarck, R. J. Williams, K. G. Harding. *Analysis of skin wound images using digital color image processing: a preliminary communication*. *The International Journal of Lower Extremity Wounds*. 3(3):151–156, 2004.
- [27] S. M. Pereira, M. A. C. Frade, R. M. Rangayyan, P. M. Azevedo-Marques. *Classification of color images of dermatological ulcers*. *IEEE Journal of Biomedical and Health Informatics*. 17(1):136–142, 2013.
- [28] E. A. G. Dorileo, M. A. C. Frade, R. M. Rangayyan, P. M. Azevedo-Marques. *Segmentation and analysis of the tissue composition of dermatological ulcers*. *Proceedings of the Symposium of the IEEE Canadian Conference on Electrical and Computer Engineering*, 1:1–4, Calgary, AB, Canada, May, 2010.

- [29] H. Zheng, L. Bradley, D. Patterson, M. Galushka, J. Winder. *New protocol for leg ulcer tissue classification from colour images*. Proceedings of the 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, San Francisco, CA, 1:1389-1392, September 1–4, 2004.
- [30] A. S. Tarallo, A. Gonzaga, M. A. C. Frade. *Artificial neural networks applied to the segmentation and classification of digital images of cutaneous ulcers*. Proceedings of the IEEE 7th International Conference on Bioinformatics and Bioengineering, 1:1–1, Boston, MA, October 14-17, 2007.
- [31] S. M. Pereira, M. A. C. Frade, R. M. Rangayyan, P. M. Azevedo-Marques. *Recuperao de imagens baseada em seu contedo: uma avaliao da performance de mtricas de distncia*. Proc. XXIII Congresso Brasileiro em Engenharia Biomdica XXIII CBEB, Porto de Galinhas, PE, Brasil, 1:1040-1044, October 2–5, 2012.
- [32] M. E. Celebi, Y. A. Aslandogan. *Content-based image retrieval incorporating models of human perception*. Proceedings of the International Conference on Information Technology: International Conference on Coding and Computing, Las Vegas, NV, 2:241-245, April 5–7, 2004.
- [33] M. M. Rahman, B. C. Desai, P. Bhattacharya. *Image retrieval-based decision support system for dermatoscopic images*. Proceedings of the 19th IEEE Symposium on Computer-based Medical Systems, Salt Lake City, UT, 285-290, 22–23 June, 2006.
- [34] M. Celebi, Y. Aslandogan, P. Bergstresser, *Unsupervised border detection of skin lesion images*. Proceedings of the International Conference on Information Technology: International Conference on Coding and Computing, Las Vegas, NV, 2:123-128, April 4–6, 2005.
- [35] P. Plassmann, T. Jones. *MAVIS: A non-invasive instrument to measure area and volume of wounds*. Medical Engineering and Physics, 20:332-338, 1998.

- [36] P. M. Azevedo-Marques, S. M. Pereira, M. A. C. Frade, R. M. Rangayyan. *Segmentation of dermatological ulcers using clustering of color components*. In Proc. IEEE Canadian Conference on Electrical and Computer Engineering, Regina, SK, Canada, May 5-8, 2013.
- [37] P. Dempster, N. M Laird, D. B. Rubin. *Maximum likelihood from incomplete data via the EM algorithm*. Journal of the Royal Statistical Society. 39(1):1–38, 1977.
- [38] F. Pernkopf, D. Bouchaffra. *Genetic-based EM algorithm for learning Gaussian mixture models*. IEEE Transactions on Pattern Analysis and Machine Intelligence. 39(8):1344–1348, 2005.
- [39] R. M. Rangayyan, B. Acha, C. Serrano, *Color Image Processing with Biomedical Applications*. SPIE Press, Bellingham, WA, 2011.
- [40] A. Ford, A. Roberts. *Color Space Conversions*. Technical report. Westminster University, London, 1998.
- [41] S. R. Garner. *WEKA: The Waikato Environment for Knowledge Analysis*. In Proc. of the New Zealand Computer Science Research Students' Conference, Hamilton, New Zealand, 1:57-64, April 18–21, 1995.
- [42] R. C. Gonzalez, R. E. Woods. *Digital Image Processing*, 221–248, Addison-Wesley, New York, NY, 1993.
- [43] R. M. Haralick, K. Shanmugam, I. Dinstein. *Textural features for image classification*. IEEE Transactions on Systems, Man, and Cybernetics. 3:610–621, 1973.
- [44] R. W. Conners, C. A. Harlow. *A theoretical comparison of texture algorithms*. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2:204–222, 1980.
- [45] V. Arvis, C. Debain, M. Berducat, A. Benassi. *Generalization of the cooccurrence matrix for colour images: application to colour*

texture classification. Image Analysis and Stereology, 23(1):63–72, 2011.

- [46] S. Le-Cessiet, J.C Van-Howelingen. *Ridge estimators in logistic regression*. Applied Statistics. 41(1):191–201, 1992.

Lucas C. Pereyra, Sílvio M. Pereira, Juliana P. Souza, Received May 19, 2014
Marco A. C. Frade, Rangaraj M. Rangayyan,
Paulo M. Azevedo-Marques

Lucas C. Pereyra
Ribeirão Preto Medical School, University of São Paulo
Av. dos Bandeirantes 3900, Monte Alegre, 14049-900, Ribeirão Preto - SP, Brazil
E-mail: lucascalabrez@gmail.com

Sílvio M. Pereira
Bioengineering Interunits Graduate Course, University of São Paulo
São Carlos - SP, Brazil
E-mail: silviomoreto@usp.br

Juliana P. Souza
Ribeirão Preto Medical School, University of São Paulo
Av. dos Bandeirantes 3900, Monte Alegre, 14049-900, Ribeirão Preto - SP, Brazil
E-mail: jupsouza@usp.br

Marco A. C. Frade
Ribeirão Preto Medical School, University of São Paulo
Av. dos Bandeirantes 3900, Monte Alegre, 14049-900, Ribeirão Preto - SP, Brazil
E-mail: mandrey@fmrp.usp.br

Rangaraj M. Rangayyan
Department of Electrical and Computer Engineering, Schulich School of Engineering,
University of Calgary, Calgary, Alberta, T2N 1N4, Canada
E-mail: ranga@ucalgary.ca

Paulo M. Azevedo-Marques
Ribeirão Preto Medical School, University of São Paulo
Av. dos Bandeirantes 3900, Monte Alegre, 14049-900, Ribeirão Preto - SP, Brazil
E-mail: pmarques@fmrp.usp.br

A platform for Image Reconstruction in X-ray Imaging: Medical Applications using CBCT and DTS algorithms

Invited Article

Zacharias Kamarianakis, Ivan Buliev, Nicolas Pallikarakis

Abstract

This paper presents the architecture of a software platform implemented in C++, for the purpose of testing and evaluation of reconstruction algorithms in X-ray imaging. The fundamental elements of the platform are classes, tightened together in a logical hierarchy. Real world objects as an X-ray source or a flat detector can be defined and implemented as instances of corresponding classes. Various operations (e.g. 3D transformations, loading, saving, filtering of images, creation of planar or curved objects of various dimensions) have been incorporated in the software tool as class methods, as well. The user can easily set up any arrangement of the imaging chain objects in 3D space and experiment with many different trajectories and configurations. Selected 3D volume reconstructions using simulated data acquired in specific scanning trajectories are used as a demonstration of the tool. The platform is considered as a basic tool for future investigations of new reconstruction methods in combination with various scanning configurations.

Keywords: Computed Tomography, Digital Tomosynthesis, CBCT, image reconstruction, class library.

1 Introduction

Virtual instrumentation has become a valuable tool for testing or evaluating different approaches, methods, and techniques in any research field. In the field of Medical Imaging this is especially valid for the

©2014 by Z. Kamarianakis, I. Buliev, N. Pallikarakis

radiography, nuclear imaging, computed tomography, etc., where due to the hazardous nature of the radiation and the limited access to instrumentation, organizing experiments is often difficult. There exist software packages [1-6], which can be used for simulation of X-ray images. For producing series of realistic projections these packages usually assume known, well-studied scanning trajectories, and the development or testing of new approaches is usually related to a significant additional programming work.

The idea to develop a dedicated library to be used in simulations in the field of X-ray imaging originates in the past [7]. Since its earlier implementation showed some drawbacks, the authors have started to develop a new such library – RTCL: Reconstruction Techniques Class Library [12] based on the experience acquired in the field of X-ray imaging algorithms and taking into account the much larger variety of possible applications. During the creation of the library two main application aspects were considered. Firstly, in many cases, known image reconstruction algorithms need to be applied over various projection data. Therefore, the possibility for fast development of such type of applications has been targeted. Secondly, it would be valuable to possess a tool for convenient programming and testing of new reconstruction techniques, using modified or completely new projection acquisition trajectories, etc. and this has been the main driving force to create the RTCL.

This paper describes the current structure of the RTCL library and the way it can be used for developing software applications within the field of the X-ray Computed Tomography (CT). The library is now the core component of an ambitious software application – the Platform for Image Reconstruction in X-ray Imaging (PIRXI). An initial version of the platform has already been developed and is currently used in research projects of the team. The present paper also describes key points in the development of PIRXI and the adopted approach of defining tasks, accessing projection data from different sources. Finally, the paper includes examples of reconstructed images with the help of PIRXI, from simulated data used in medical imaging, after applying Cone-beam CT (CBCT) or Digital Tomosynthesis (DTS) reconstruction algorithms.

2 Materials & Methods

2.1 The RTCL

In RTCL, with respect to its predecessor, assignment, inheritance of geometrical and functional properties, and the use of all library components have been implemented in a different and more practical way. An overview of the different groups of classes and relations is represented by the simplified diagram in Figure 1. An object-oriented programming approach, which perfectly complies with the object-oriented composition and functioning of the CT imaging systems, has been followed.

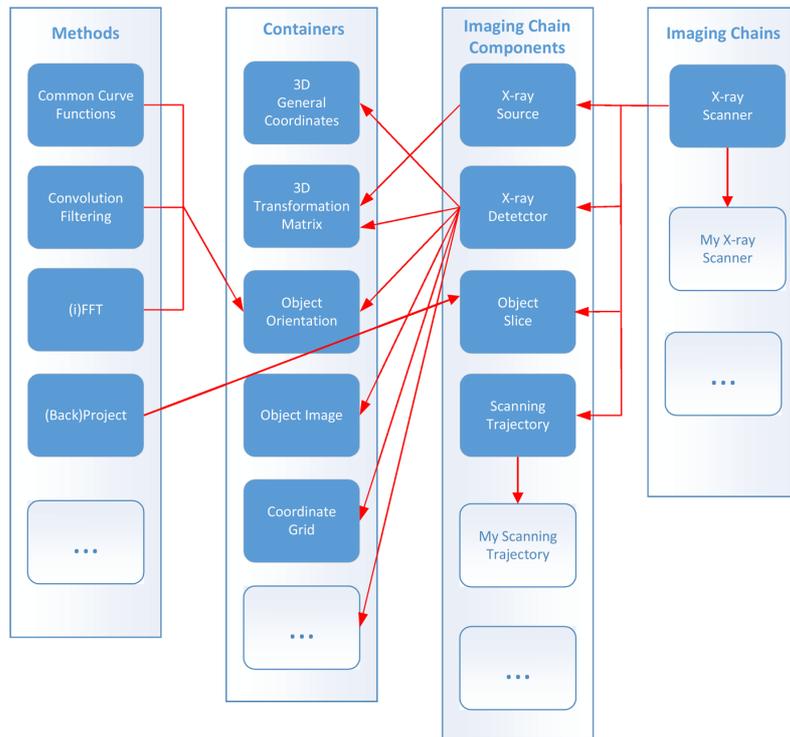


Figure 1. Basic components of the RTCL library

Components in a real CT imaging chain usually include X-ray source(s), X-ray detector(s), mechanical construction able to move

the source-detector pair(s) along a specific trajectory, data processing unit(s), dedicated image reconstruction processor and object motion-correction techniques. RTCL provides the programmer with software equivalents of those components, allowing intuitively and easily to repeat and simulate any imaging setup and eventually use the available projection images to reconstruct tomograms at arbitrary orientations. RTCL offers different types of classes, which combine content and inherit properties in a hierarchical sequence. The Containers group comprises classes for handling 2D/3D coordinates (of pixels, voxels, vectors, etc.) either locally on a surface (e.g. detector plane) or globally in the 3D space. Grid classes carry analytical description of different surfaces and contain arrays of the coordinates of their nodes. The latter can belong to a flat surface, thus are equidistantly distributed over a plane, or they can belong to a curved (cylindrical, spherical) surface, therefore following a different description. Sets of Euler angles describe the orientation of objects in space. Images are separate containers for different types of image data. The Methods group of classes provides mostly functions to facilitate handling and processing projection or reconstruction data. It includes common 1D/2D functions (e.g. signal windows, filter kernels, filter responses), FFT/IFFT routines, convolution/filtering routines, for image processing in the frequency and the spatial domain. Projection/backprojection routines are also implemented in order to help in the validation or development of analytical reconstruction methods or in the production of experimental projection images. Specific classes form the Imaging Chain Components group. Detectors represent the variety of image sensors used in the CT imaging, fluoroscopic or any other direct way of imaging. Within a 2D detector, pixels are uniquely identified by their indices or by their local or global coordinates. It is always possible to convert any type of coordinates to any other type, using the a priori information for the imaging system. Trajectory classes incorporate geometry and motion of different setups of source-detector pairs and simplify their positioning in space. Examples include circular isocentric trajectory, spiral/helical, partial isocentric along a limited arc, linear, etc. Slices are utilized in the representation of tomographic images or 3D

volumes, while the X-ray source class, for example, describes a point source in terms of its coordinates, focal spot size, energy, field distribution, etc. The Imaging Chains group is intended to contain functional aggregations (e.g. scanners, C-arms, etc.) of the above classes. The last two groups are considered quite open (from the user's/developer's aspect) and any new development (e.g. trajectories, geometrical configurations) can be added.

Figure 2 helps to illustrate the example definition of a simple CArm scanner class. As it is shown, the class is defined initially by inheriting most of the functionality from a ready abstract class, the `BaseXrayScanner`. This parent class contains abstract components like a base-source, a base-detector, a base-trajectory as well as virtual methods (e.g the function that is expected to implement the relevant movement – `moveTo(...)`) that the children classes should override accordingly. The constructor of the derived CArm class is written in a forward manner, as it is shown on the right. The input arguments in the definition of the constructor are already specific for the derived class e.g. the source to isocenter distance (SID), the source to detector distance (SDD), the number of nodes along the principal axes of the detector as well as the size of its pixels. In the implementation, these arguments can be used to create and retrieve specific components, like an `XrayPointSource`, and a `FlatXrayDetector`, where the S and D denote pointers to abstract equivalents of these components.

The rotation of the gantry is translated into motion of the source-detector pair. The rules for that could be provided by the `Circular Trajectory Class`. With the help of an overloaded function (e.g `moveTo(...)` in Figure 2) the source-detector pair is instructed to move along a circular arc synchronously, defined exactly by the `Circular Trajectory Class`. The backprojection operation is performed with the help of a universal projector/backprojector function. The arguments of this function include a `Projection Set Description` object, which fully describes the acquisition and geometrical setup of an imaging scenario and a `Slice` object on which the backprojection process will be performed. The compactness of the code proves the robust implementation of the initial idea of RTCL.

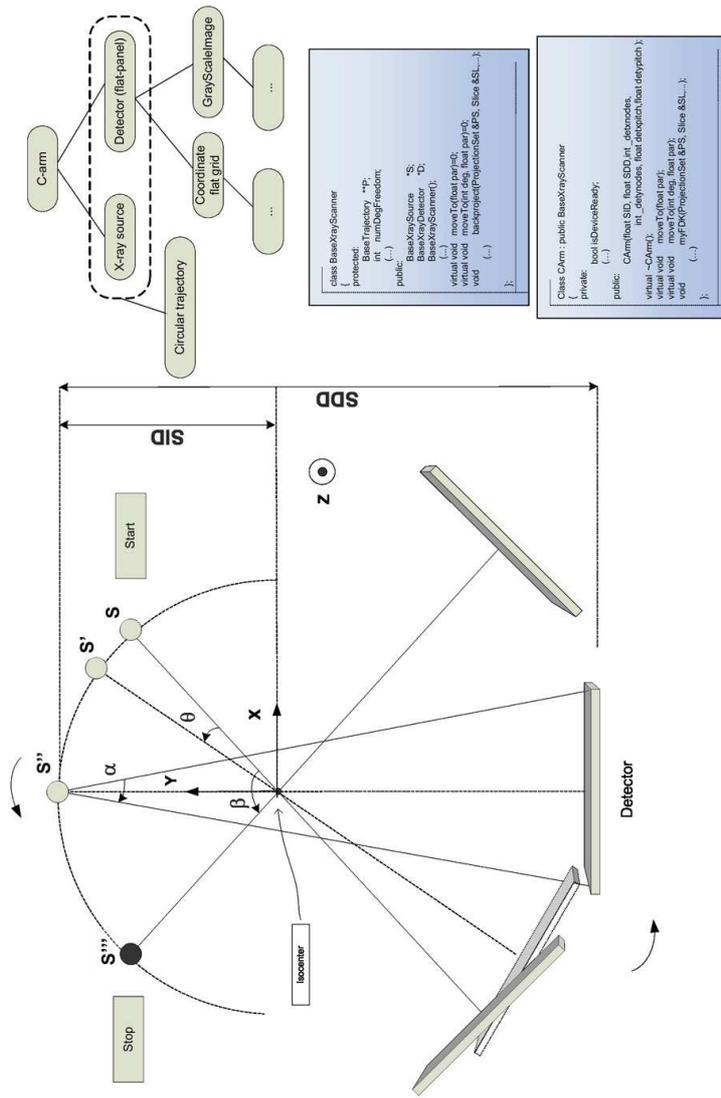


Figure 2. C-arm device configuration and relations to the RTCL components. On the left, the acquisition geometry in limited arc trajectory like a C-arm is shown. On the right, basic building blocks that constitute the construction of a CArm object scanner class. The CArm is derived from its parent, the BaseXrayScanner class

2.2 Integrating RTCL into PIRXI

The idea of integrating the library into a multi-purpose application for image reconstruction is a logical follow-up of the first medical and non-medical (e.g. in the area of NDT) imaging applications based on it. The library is now a core component of a larger software application: the Platform for Image Reconstruction in X-ray Imaging – PIRXI. An initial version of the platform has been already created and is currently being used in the implementation of various image reconstruction tasks in the research activities of the team.

The GUI of PIRXI is developed in C/C++ and Qt. This allows a good portability of the code along different computing platforms and OSs. It has been tested on Windows and Ubuntu Linux.

It is a common situation when a researcher obtains CT projection data from external sources and spends significant amount of time trying to convert them into an appropriate format of the application he/she uses. PIRXI tries to deal with this problem and can provide a solution to reconstruction tasks using a wide range of scanning geometries and projection image formats. Based on the system of the .INI files and with the help of a rich set of keys, a large number of projection file naming, contents and data formats are transparently handled by the platform and the projections can be loaded and further used.

A snapshot of the GUI of the PIRXI platform running under Windows is shown in Figure 3. The GUI currently allows the definition of a whole reconstruction job. The user can interactively describe such a reconstruction task by providing information to the system (e.g. through a Dialogue), where he/she describes the general settings of the task. Common parameters include the description of the projection set (by means of data format, geometric acquisition settings, description of the detector's pixel size and metrics, location of data and destination of the processed files, etc.). The projection processing/preprocessing (e.g. filtering) follows, where again the task is easily performed through buttons on the GUI. Finally, the user can easily perform experiments and gain an understanding of the loaded/processed available data by choosing to reconstruct either central positioned slices or tomograms

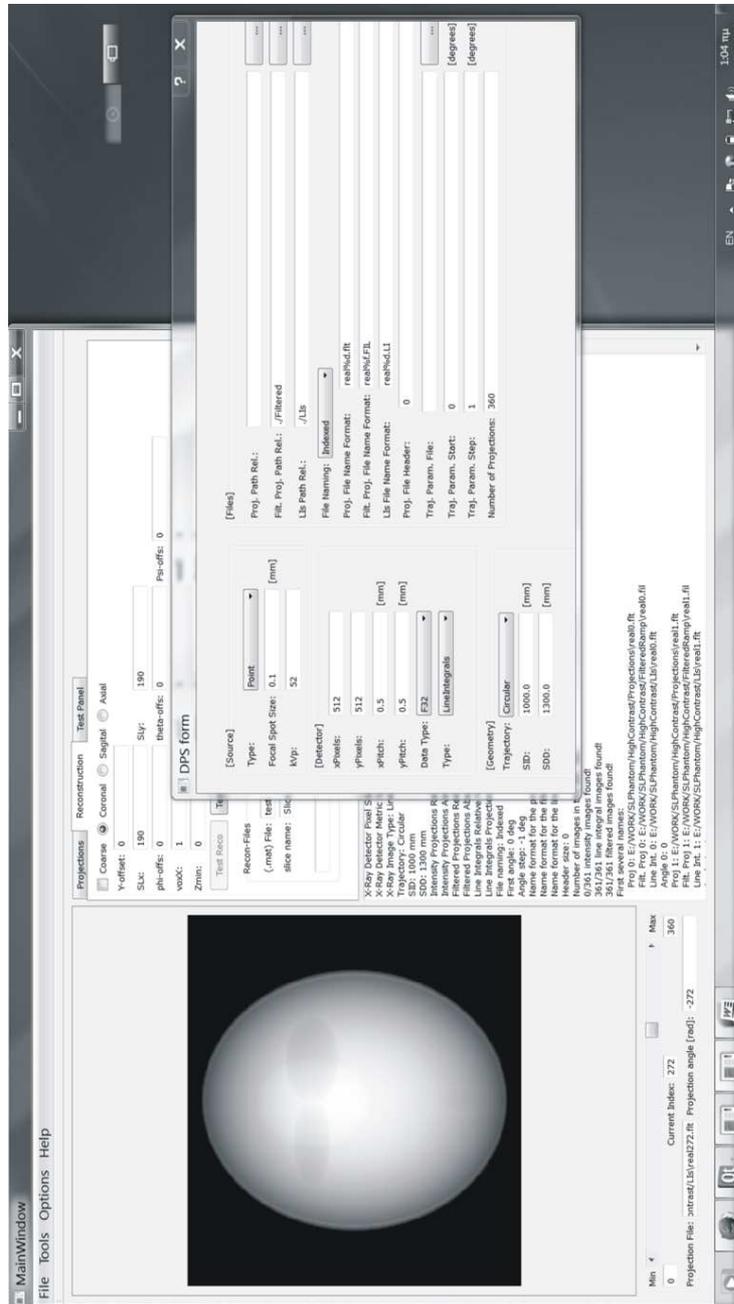


Figure 3. A snapshot of the GUI of PIRXI platform

of arbitrary orientation. Moreover, the possibility for a direct volumetric reconstruction is given. The image reconstruction method can be chosen from the already implemented ones through the GUI, where further options are provided in case the reconstruction algorithm demands additional input from the user (e.g. the arc in DTS or the projection angle spacing used for reconstruction).

2.3 Examples of using PIRXI in medical applications

Selected reconstructed tomograms using simulated data acquired in specific scanning trajectories are used as a demonstration of the tool. The use of simulated data is particularly useful in studying and analyzing reconstruction algorithms in principle, since data are free from distortions and mechanical inaccuracies inherent to radiographic units. Noise-free monoenergetic projection images of three simulated phantoms were acquired using an in-house developed tool, the XRAYImagingSimulator [2].

The first phantom is a modified (high contrast) version of the well known analytical Shepp-Logan phantom that approximates the human head and consists of 12 ellipsoids. The phantom was included in a cubic volume of size 190x190x190 and voxel size 1.

The second phantom is a 3D voxelized model of a metallic cylindrical implant (e.g. intramedullary tibial nail) used in interventional procedures in orthopedic surgery for the treatment of bone fractures. The implant consists of 8 screw holes at various locations and orientations, and is placed in three other cylinders that model a human leg (e.g. red bone marrow, bone and muscle). A detailed study on accurate localization of hole canals using CBCT, is presented in [9].

As a third example, a simulated 8cm complex uncompressed breast phantom [3] approximating the medium-sized breast was used. The model contains four clusters of microcalcifications (μ Cs) delivered in two different groups. In two different layers of the phantom, two clusters of 5 μ Cs in one layer and two clusters of 6 μ Cs, modeled as spheres/ellipsoids of calcium carbonate and of size 0.2 mm and 0.4 mm, are placed. The simulation parameters for image acquisition as well as

the reconstruction settings used for all the phantoms are presented in table [1].

Table 1. Acquisition and reconstruction settings used along with the three simulated phantoms

Parameter/ Settings	Phantoms		
	Shepp-Logan	Metallic im- plant	Uncompressed Breast
Source-isocenter distance (SID)	1000 mm	1000 mm	600 mm
Source-detector distance (SDD)	1300 mm	1300 mm	665 mm
Number of views/Projections	360	121	25
Detector size	512x512 (pixels)	480x480 (pixels)	500x500 (pixels)
Pixel pitch	0.5mm	0.5mm	0.2 mm
Magnification factor	1.3	1.3	1.108
Reconstruction matrix/slice size	190x190x190	370x370	500x500
Source/Detector trajectory	Full 360° circular trajectory, angular step 1°	Circular isocentric rotation, 121° limited arc, angular step 1°	Circular isocentric rotation, 48° limited arc, angular step 2°
Reconstruction algorithm	FDK	FBP	FBP & SAA

2.3.1 CBCT reconstruction using the FDK algorithm

For the case of Shepp-Logan phantom, 360 projections images, acquired every 1° step, were used for volumetric reconstruction using the FDK algorithm [10]. The main advantage of cone beam algorithms is the reduction in data collection time. With a single source, ray integrals are measured through every point in the object, in the time it takes to measure a single slice in a conventional two-dimensional scanner. The projection data can be expressed as a function of the source angle and

the horizontal and vertical positions on the detector plane. The FDK algorithm is an approximate formula and represents a generalization of the 2D fan-beam reconstruction formula to the 3D case. The volume reconstruction is based on initial filtering and subsequent length correction weighting of the projection data followed by backprojecting a single plane within the cone, for each elevation along the z -axis. The final three-dimensional reconstruction is obtained by summing the contribution to the object from all tilted fan beams [8], which involves a final weighting that accounts for this tilt during backprojection. The FDK algorithm can be implemented with moderate computational requirements and delivers a satisfactory reconstruction quality for small cone angles (e.g. up to 10°). The individual steps of this algorithm are already implemented in the RTCL.

2.3.2 DTS reconstruction using Shift and Add & Filtered Backprojection algorithms

For the breast phantom and the metallic implant phantom, DTS image reconstruction techniques were applied. DTS is a limited angle method of image reconstruction, where projection images acquired at regular angular intervals, and during a single acquisition pass, are used for reconstruction of planar sections [11]. In many applications (e.g. mammography) the source trajectory traces a limited circular arc, while the detector usually remains stable. In the current investigation, the acquisition geometry is isocentric. The acquisition geometry of isocentric Digital Tomosynthesis setup resembles that of the C-arm shown in Figure 2, but the angular range of the source-detector pair is usually much smaller. Important parameters for this acquisition geometry are once again the SID and SDD, as well as the acquisition range (denoted with ϕ) and the angular step θ . With the parameter α in Figure 2, the fan-beam angle is indicated. In the present example, 25 images of an uncompressed simulated breast were acquired in the limited arc -24° to 24° , with 2° step (Table 1). Tomograms were further reconstructed using a simple backprojection technique (Shift and Add algorithm – SAA), similar to that of [11] and a Filtered Backprojection (FBP) al-

gorithm. Both of these algorithms are implemented in RTCL. In both cases, the purpose was to bring in focus the planes of interest (e.g. the μ Cs). The Simple Backprojection algorithm was utilized, as it requires a straightforward implementation and minimal computational power and processing time efforts. However, since this technique introduces additional reconstruction artifacts (e.g. out-of-plane structures with high contrast tend to appear as low-contrast replicas in reconstruction planes), the FBP algorithm was also applied in an attempt to recover the loss of contrast especially for the small structures (e.g. the μ Cs). For the case of metallic implant phantom, a larger acquisition arc of 121° was used with step of 1° . In this case, only a FBP approach was followed for the reconstruction.

3 Results

Figure 4 demonstrates the outcome of the FDK reconstruction of the first phantom. Using projection images of the modified Shepp-Logan phantom, central three orthogonal to each other reconstructed slices (axial, coronal, saggital) and a slice at a plane $Z = -24$ mm away from the central plane, are presented.

Accordingly, below each reconstructed slice, line plot profiles (corresponding positions are marked with a white solid line in each reconstruction), are presented as compared to the original (dashed line).

The visual inspection of the reconstructed images and especially the comparison of the selected line profiles (in the reconstructions as compared to the original phantom) validate the correct implementation of the reconstruction algorithm using the RTCL.

In Figure 5, DTS reconstructions of a simulated uncompressed breast are shown. In the upper row, a FBP algorithm implemented with the help of the RTCL library is used for reconstruction, while in the bottom row, results after applying a SAA approach in corresponding locations, are presented. In both reconstruction approaches the μ Cs were found to be in-focus in the planes at $Z = -21$ mm below and $Z = 9$ mm above the central plane, which is considered at the isocenter. In the right part of Figure 5 and from up to down, zoomed recon-

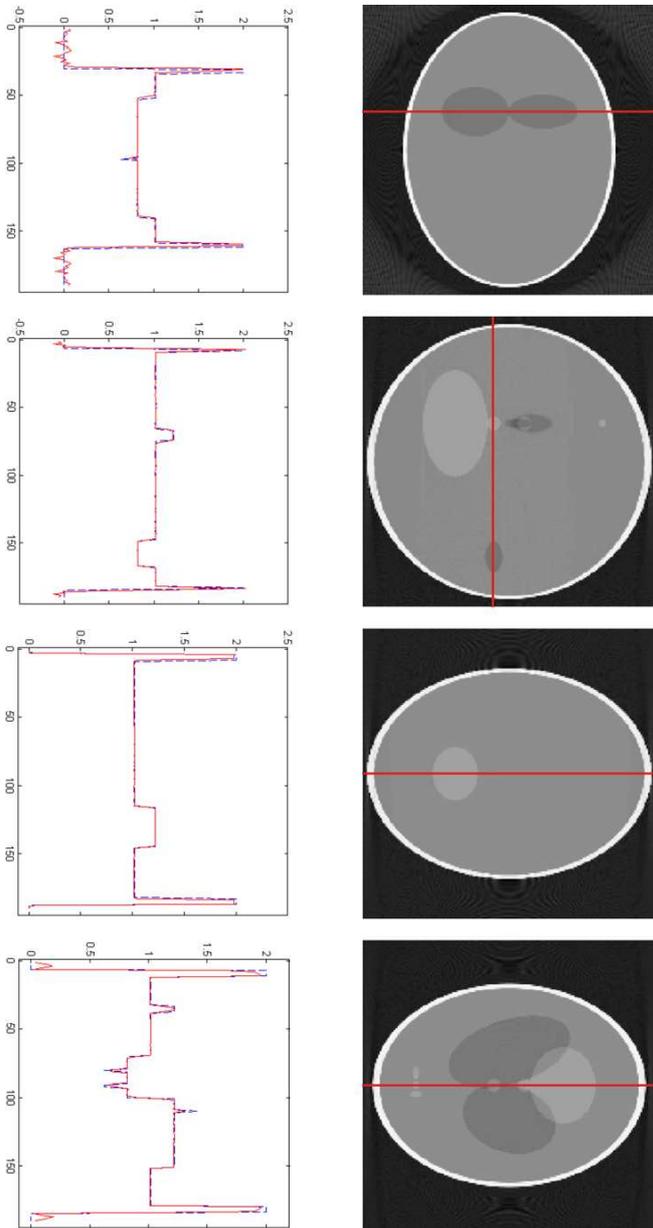


Figure 4. CBCT reconstruction of a modified Shepp-Logan phantom using the FDk algorithm implemented in the RTCL library. From left to right, line profiles (with solid lines) in the reconstructions at $x = 70$, $y = 95$, $x = 105$ and $x = 96$ (for the far right slice) at plane $z = -24$, as compared to the original (with dashed line)

structed regions of interest containing the features under investigation are presented, at $Z = -21\text{mm}$ and $Z = 9\text{mm}$, after applying a FBP algorithm and after using SAA, respectively. Both groups of larger μCs (4mm diameter) are well visualized in the reconstructed planes. Regarding the smaller in diameter μCs (2mm), 2 out of the group of 5 μCs from the plane at $Z = -21\text{mm}$ and 1 out of the group of 6 μCs were not observed at all in any of the reconstructed slices.

This is a common situation in Breast imaging tomography, since μCs with size smaller than 0.25mm are reported to have difficulties in detection. Moreover, tomographic reconstruction further away from the central plane of the isocenter introduces additional artifacts that overlay in some cases the features under investigation.

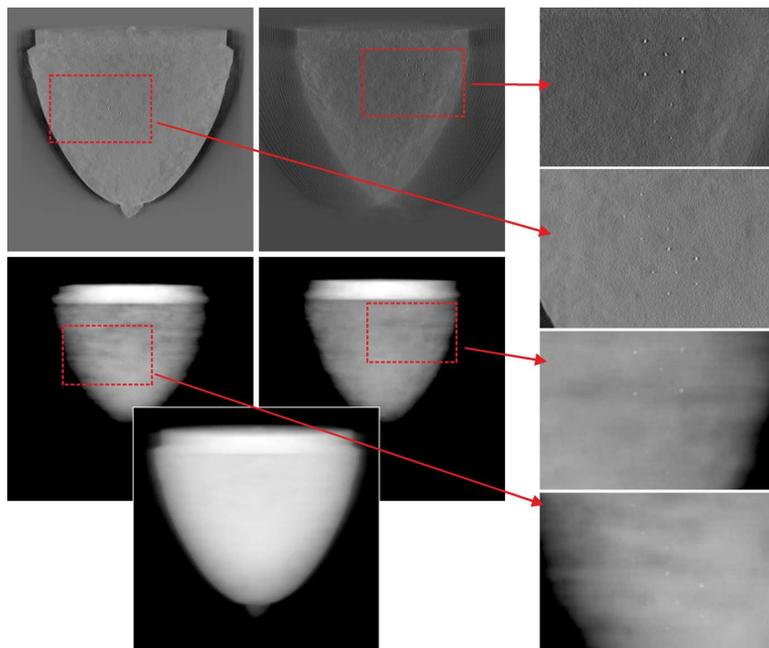


Figure 5. DTS reconstruction of a simulated breast using an implementation of a FBP reconstruction algorithm in RTCL (upper row) and a corresponding Shift and Add technique (bottom)

In Figure 6, DTS reconstructions of a simulated metallic implant are shown. Reconstructions were performed as previously with a FBP reconstruction algorithm using the RTCL library. 2D tomograms were reconstructed at “arbitrary” orientations by feeding the platform with Euler angles close to those that describe the phantom rotation around the principal axes, but not with their exact values.

The results present reconstructed slices almost “parallel” to the main axis of the implant, providing useful information for further image processing and analysis. The openings (e.g. hole canals) are in most cases well visualized.

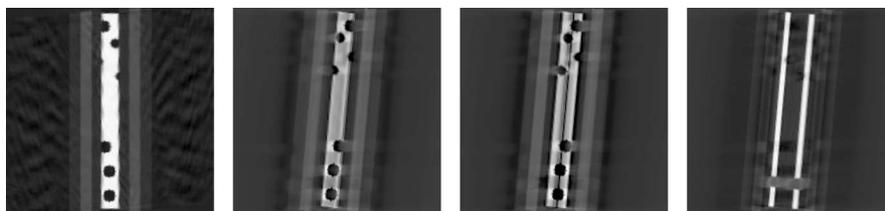


Figure 6. DTS reconstruction of a simulated metallic surgical implant

4 Conclusions

This paper describes the current structure of the RTCL library and the way it can be used for developing software applications within the field of the X-ray Computed Tomography (CT). The library is considered as the core component of an integrated software application – the Platform for Image Reconstruction in X-ray Imaging (PIRXI). Selected reconstructed slices from simulated data used in medical imaging, after applying CBCT or DTS reconstruction algorithms, were used for demonstration purposes. The current tests of both RTCL and PIRXI prove the flexibility of the new approach to image reconstruction research and algorithms implementation. The tools aim to facilitate any experimentation, study and development/implementation of reconstruction algorithms and scanning geometries and configurations in the Computed Tomography field.

References

- [1] A.Glière. *Sindbad:From CAD model to synthetic radiographs* (1998) Review of Progress in Quantitative Nondestructive Evaluation,17:387–394.
- [2] D.Lazos, K.Bliznakova, Z.Kolitsi, N.Pallikarakis. (2003) *An integrated research tool for X-ray imaging simulation*, Comp. Meth. & Prog. in Biomed: 70:241–251.
- [3] K.Bliznakova, R.Speller, J.Horrocks, P.Liaparinos, Z.Kollitsi, N.Pallikarakis. *Experimental validation of a radiographic simulation code using breast phantom for X-ray imaging*, vol 40:2 (2010), pp. 208–214.
- [4] P.Duvauchelle, N.Freud, V.Kaftandjian, D.Babot. *A computer code to simulate X-ray imaging techniques*, Nuclear Instruments and Methods in Physics Research B 170, (2000), pp. 245–258.
- [5] M.Winslow, X.G.Xu, B.Yazici. *Development of a simulator for radiographic image optimization*, Comp. Meth. & Prog. in Biomed, (2005) 78, pp. 179–190.
- [6] D.Lazos, Z.Kolitsi, N.Pallikarakis. *A software data generator for radiographic imaging investigations*, IEEE T. Inf. Technol. B, 4 (2000), pp. 76–79.
- [7] Z.Kamarianakis, I.Buliev, N.Pallikarakis. *CLCT: A C++ Library for Computed Tomography applications*, Proc. ESBME 2008, 6th European Symposium on Biomedical Engineering, 19-21 June, Chania, Greece.
- [8] A.C.Kak, M.Slaney. *Principles of Computerized Tomographic Imaging*, IEEE Press 1988.
- [9] Z.Kamarianakis, I.Buliev, N.Pallikarakis. *Robust identification and localization of intramedullary nail holes for distal locking using CBCT: A simulation study*, Med. Eng. & Physics vol. 33(2011), pp. 479–489.

- [10] L.A.Feldkamp, L.C.Davis, J.W.Kress. *Practical cone-beam algorithm*, J. Opt. Soc. Am. A. 1(6): 612–619, 1984.
- [11] Z.Kolitsi, G.Panayiotakis, V.Anastassopoulos, A.Scodras, N.Pallikarakis. *A multiple projection method for digital tomosynthesis*, Med. Phys. 19(4), 1045–1050, 1992.
- [12] Z.Kamarianakis, I.Buliev, N.Pallikarakis. *A C++ platform for image reconstruction in X-ray imaging*, 5th Panhellenic Conference on Biomedical Technologies, ELEVIT(2013), 4-6 April, Athens, Greece.

Zacharias Kamarianakis, Ivan Buliev,
Nicolas Pallikarakis

Received March 13, 2014

Zacharias Kamarianakis
University of Patras, Dept. of Medical Physics
26504 Rio, Patras, Greece
Phone: (30) 2610 969109
E-mail: zkamar@upatras.gr

Ivan Buliev,
Studentska str.1, Technical University of Varna, Dept. of Electronics
9010, Varna, Bulgaria
Phone: (359) 52 383 572
E-mail: buliev@tu-varna.bg

Nicolas Pallikarakis
University of Patras, Dept. of Medical Physics
26504 Rio, Patras, Greece
Phone: (30) 2610 997702
E-mail: nipa@upatras.gr

Parallel Processing and Bio-inspired Computing for Biomedical Image Registration

Invited Article

Silviu Ioan Bejinariu, Hariton Costin, Florin Rotaru,
Ramona Luca, Cristina Niță, Camelia Lazăr

Abstract

Image Registration (IR) is an optimization problem computing optimal parameters of a geometric transform used to overlay one or more source images to a given model by maximizing a similarity measure. In this paper the use of bio-inspired optimization algorithms in image registration is analyzed. Results obtained by means of three different algorithms are compared: Bacterial Foraging Optimization Algorithm (BFOA), Genetic Algorithm (GA) and Clonal Selection Algorithm (CSA). Depending on the images type, the registration may be: area based, which is slow but more precise, and features based, which is faster. In this paper a feature based approach based on the Scale Invariant Feature Transform (SIFT) is proposed. Finally, results obtained using sequential and parallel implementations on multi-core systems for area based and features based image registration are compared.

Keywords: image registration, clonal selection algorithm, bacterial foraging algorithm, genetic algorithm, parallel computing.

1 Introduction

Image registration is the process of geometric overlaying or alignment of two or more images of the same scene taken at different times, from different viewpoints, and/or by different sensors [1]. Image registration

©2014 by Silviu Ioan Bejinariu, Hariton Costin, Florin Rotaru, Ramona Luca, Cristina Niță, Camelia Lazăr

(IR) is the first step in image fusion procedures, which combine relevant information from one or more images to create a single image with more informational content. Image registration and fusion methods are used in remote sensing applications, geographic information systems, multispectral image analysis, medical image analysis and other domains. Image fusion may be applied at pixel, feature or decision levels. In the first case, when pixel level image fusion have to be applied, the input images must be registered, because they may differ by the view angle, subject position and also some geometric distortions may be added by the capture device.

There are two different approaches in IR: area (pixel intensity) based methods and feature-based methods [1]. The geometric transform that must be computed may be global (for the entire image) or local in case the images are locally deformed. The most frequently used transforms are the shape preserving mappings (rotation, translation, scaling and the affine transform).

In this paper three different optimization methods are used for the geometric transform parameters estimation: Bacterial Foraging Optimization Algorithm (BFOA), Genetic Algorithm (GA) and Clonal Selection Algorithm (CSA). The foraging model is suitable for optimization problems because animals search for nutrients and try to avoid noxious substances in a way that maximize their energy intake per unit time spent foraging [2]. Computational methods can provide decision models for optimal foraging. The Bacterial Foraging Optimization Algorithm (BFOA) proposed by Passino uses the *Escherichia coli* bacteria model because it is the most understood microorganism [2], [3], [4]. BFOA is used in image processing to solve also other optimization problems: edge detection in combination with a probabilistic derivative technique [5]; fuzzy entropy based image segmentation [6]. A modified version of BFOA used for multilevel thresholding segmentation was compared to genetic algorithms and particle swarm optimization algorithm [7]. Image registration BFOA based methods were proposed in [8], [9] and [10] Parallel implementations of BFOA were proposed in [11] and [12]. Genetic Algorithms are search techniques that emulate evolutionary processes to solve optimization problems [13]. Like BFOA,

GAs start with a population of individuals (points) in the problem domain and use these points to approximate the optimal solution. The difference is that instead moving in the problem domain, GAs use the recombination of two or more parents to produce offspring [14]. GAs are often used in biomedical or remote sensing image registration [13]. The Clonal Selection Algorithm (CSA) belongs to the field of Artificial Immune Systems that include computational methods inspired by the mechanisms of the biological immune system [15]. Like GAs, CSA may use binary solution coding and real coding. In [16] a real coded clonal selection algorithm is used in electromagnetic design optimization. It is also suitable for high dimensional optimization problems. CSA is effective, in terms of accuracy, capable of solving large-scale problems [17] and is comparable to other optimization algorithms. A performance comparison of CSA and GA is presented in [18] and conclusion is that each one has better performance depending on the function to optimize.

The paper is organized as follows. In the second section the measures used for area based and features based IR methods are described. In case of features based IR, a short description of SIFT transform and procedure to find the SIFT key points correspondences are also included. In the third section, the optimization algorithms – BFOA, GA and CSA – are shortly presented. In the fourth section the proposed parallel versions of BFOA and GA are presented. In the fifth section the results obtained by applying the optimization procedures for biomedical image registration are shown in both sequential and parallel versions. The last section concludes the paper.

2 Image Registration

There are two different approaches in IR: area (pixel intensity) based methods and feature-based methods [1]. Almost all methods consist of four steps: feature detection, feature matching, transform estimation and image resampling. The feature detection step is specific to feature based registration methods and distinctive and stable features (points, lines, contours, regions) have to be detected. Because the transform es-

timation is performed while looking for the correspondent features, the second and third steps are usually combined. In the image resampling step, different interpolation methods are used: the nearest neighbor function, the bilinear and bicubic functions, quadratic splines, cubic and higher-order B-splines [1].

2.1 Area Based IR

In case of area based IR methods, to evaluate the similarity between images the normalized correlation (NCC), the Fourier representation or normalized mutual information (NMI) are used. In this study, the Mutual Information is used to evaluate the similarity in case of area based registration.

Mutual information is a robust measure used in image registration [1], [10]. It evaluates the relative independence of two images and does not depend on the specific dynamic range or intensity scaling of the images. High values of mutual information indicate high dependence between images. It is defined as

$$MI(A, B) = H(A) + H(B) - H(A, B), \quad (1)$$

where $H(\cdot)$ is the image entropy and $H(A, B)$ is the joint entropy of the two images. Because mutual information based registration methods are sensitive to changes that occur in the distributions as a result of difference in overlapping regions, normalized mutual information can be used:

$$NMI(A, B) = \frac{H(A) + H(B)}{H(A, B)}. \quad (2)$$

Registration of two images A and B requires maximization of mutual information, thus maximization of the entropies $H(A)$ and $H(B)$, and minimization of the joint entropy $H(A, B)$.

Usually, optimization in image registration means to maximize similarity. If the optimization algorithm is oriented on cost function minimization, then the value of $(-1) * MI$ is used to evaluate the cost of the transform for a certain solution.

2.2 Feature based IR

In case of feature based IR methods, spatial relations, invariant descriptors, relaxation methods and multiresolution transforms (pyramids and wavelets) are used. In this paper, the features based IR will use key points determined using the Scale Invariant Feature Transform (SIFT) [19], [20]. SIFT is used to select distinctive features, used in pattern recognition, localization, 3D mapping, tracking and image registration. It allows scale and rotation invariant features detection, with good results for affine distortions. The SIFT algorithm has 4 distinctive stages: extrema detection in the scale space of the image, key points selection and localization, key points orientation assignment and description generation. The identified features have to be distinctive.

a. *Scale-space extrema detection.* Key point candidates selection is performed by finding the extrema of the Difference of Gaussians (DOG) function computed as the difference of two scaled images separated by a multiplicative factor k .

$$\begin{aligned} D(x, y, \sigma) &= L(x, y, k\sigma) - L(x, y, \sigma) = \\ &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y), \end{aligned} \quad (3)$$

where $L(x, y, \sigma)$ is the scale space of the image $I(x, y)$ obtained by convolving it with the Gaussian kernel $G(x, y, \sigma)$. Extrema points depend on the frequency sampling in the scaled space and the initial value of σ .

b. *Key points localization.* Key points are selected from the most stable and accurately localized candidates. Key point candidates having low contrast or strong edge response in one direction only are removed. Because the candidates obtained in higher scales correspond to several pixels in the original image, for an exact localization is performed by computing the extrema points of the Taylor expansion up to quadratic terms of the scale space function $D(x, y, \sigma)$ [19].

c. *Orientation assignment.* To make key point descriptions invariant to rotation, their orientations are computed using the orientation histogram of local gradients of the closest smoothed image $L(x, y, \sigma)$.

The gradient magnitude and orientation are computed using pixel differences:

$$m(x, y) = \sqrt{(L(x-1, y) - L(x+1, y))^2 + (L(x, y-1) - L(x, y+1))^2}, \quad (4)$$

$$\theta(x, y) = \text{arctg} \frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}. \quad (5)$$

Each point is added to the histogram weighted by the gradient magnitude $m(x, y)$ and by a circular Gaussian. To obtain a more accurate orientation, the dominant peaks in the histogram are interpolated with their neighbors.

d. *Key point descriptor computing.* The key point descriptor contains $128 = 4 \times 4 \times 8$ values obtained using 16 orientation histograms computed in a 4×4 grid. Each histogram contains 8 orientation bins. The descriptor is computed in a support window of 16×16 pixels around the key point [19].

To evaluate the similarity between two images the key points correspondences have to be established. The Euclidian distances between SIFT descriptors of each key point from source image and those of the model image are computed. By sorting the computed values for source images key points, a match is established when the minimum computed distance is less than a certain percentage from the second distance. In our experiment a percent of 30% is used [20].

For IR, in the similarity evaluation step, the coordinates of the key points in the source image are transformed accordingly to the values of transform parameters and the sum of Euclidean distances between positions of key points in the model and transformed source image is used as similarity value [12].

3 Bio-inspired computing in IR

In this paper three different bio-inspired optimization methods are used in order to compute the optimal geometric transform that allows the

source image to overlay the model image: Clonal Selection Algorithm, Bacterial Foraging Optimization Algorithm and Genetic Algorithm.

3.1 Clonal Selection Algorithm

The Clonal Selection Algorithm (CSA) belongs to the field of Artificial Immune Systems which includes computational methods inspired by the mechanisms of the biological immune system. A simplified description of the immune system is an organ system intended to protect the host organism from the threats posed to it from pathogens and toxic substances.

CSA is inspired by the Clonal Selection theory of acquired immunity. It is a population based stochastic method with binary representation of variables [16] which may be used for multimodal optimization. In some cases, also real encoding of variables may be used to solve numerical problems.

Clonal Selection Algorithm can be listed as follows [18]:

1. Randomly generate a set of solution candidates: antibodies.
2. Compute the affinity values of each candidate solutions.
3. While the minimum error criterion is not met
 - 3.1 Sort the antibodies starting from the lowest affinity. The lowest affinity means better matching between antibody and antigen.
 - 3.2 Clone the better matching antibodies more with some predefined ratio.
 - 3.3 Mutate the antibodies with some predefined ratio. This ratio is obtained in a way that better matching clones mutated less and weakly matching clones mutated much more in order to reach the optimal solution.
 - 3.4 Compute affinity values of each antibody.

It starts with an initial set of adaptive units: the general immune cells. Each cell represents a possible solution of the problem and participates in a competitive selection process. The algorithm involves the selection of antibodies based on affinity against a pattern, computed by a cost function. Selected antibodies are cloned and resulted clones are subject of hypermutation. The hypermutation is inverse proportional to computed clone affinity. The resulted set competes with the already existing antibodies in the next generation of the evolution process. The low-affinity population members are replaced by new randomly generated antibodies.

3.2 Bacterial Foraging Optimization Algorithm

The Bacterial Foraging Optimization Algorithm belongs to the field of Bacteria Optimization Algorithms and Swarm Optimization. There have been many extensions of the approach that attempt to hybridize the algorithm with other Computational Intelligence algorithms and Metaheuristics such as Particle Swarm Optimization, Genetic Algorithm.

The bacterial foraging paradigm [2], [3], [4] is suitable as model for optimization algorithms because animals / bacteria behavior is to search for nutrients and avoid noxious substances to maximize their energy. BFOA is based on a colony of evolving bacteria which are replicated if have a good strategy to find nutrients or die in the other case. Each bacterium is characterized by its position and quantity of accumulated nutrients or healthy status.

In optimization problems, the possible solutions are encoded in the bacteria position and the movement of the colony members tends to approximate the optimal solution. The final solution is specified by the position in which a bacterium is in the best healthy state or the nutrients amount is the highest.

According to BFOA approach, the bacteria colony moves in the n -dimensional space, where n is the optimization problem dimension and the quantity of nutrients / healthy status is described by a cost function defined according to the optimization problem. Dur-

ing its evolution, the bacteria colony proceeds through four foraging steps: chemotaxis, swarming, reproduction and elimination-dispersal. In the following paragraphs, the colony consists of S individuals; $P(j, k, l) = \{\theta^i(j, k, l), i = 1 \dots S\}$ is the position of colony members in the j^{th} chemotactic step, k^{th} – reproduction step and l^{th} – elimination-dispersal step; $J(i, j, k, l)$ – the cost of the i^{th} bacterium in position $\theta^i(j, k, l)$.

Chemotaxis. In the chemotactic step, a bacterium can move in two ways: tumble and swim. First, a tumble is executed in a random direction. The new position of the i^{th} bacterium is:

$$\theta^i(j + 1, k, l) = \theta^i(j, k, l) + C(i)\varphi(i), \quad (6)$$

where $C(i)$ is the size of the chemotactic step and $\varphi(i)$ is a unit length of randomly generated direction [4]. The movement continues in the same direction while the value of the cost function decreases but not more than a maximum number of steps.

Swarming. The bacteria tend to swarm together if they have the possibility to signal to each other the presence of a favorable or poisonous environment (social behavior). The cell to cell attraction or rejection is modeled by modifying the value of the cost function $J(i, j, k, l)$ by a value that depends on the status of all the other bacteria in the colony.

Reproduction. After a number of chemotactic steps, all bacteria accumulate a quantity of nutrients that is usually expressed as the cost function computed in the current position. Those which accumulated a greater quantity of nutrients are in a healthier state and split into two bacteria. Those which accumulated a smaller amount of nutrients die. In BFOA, to keep constant the size of the colony, the number of bacteria which split is equal to the number of bacteria which die. The new bacteria are created without mutation in the same position as the parent bacteria [4].

Elimination and Dispersal. After a number of reproduction steps, with a specified probability P_{ed} , some bacteria are removed from colony

(elimination) regardless their healthy state and new bacteria are created in random positions (dispersal) [4].

The optimization algorithm starts with a colony of S bacteria placed in randomly generated positions. The evolutionary process consists of N_{ed} elimination-dispersal steps, each of these consists of N_{re} reproduction steps and each reproduction step consists of N_C chemotactic steps. In each chemotactic step a bacterium may do at most N_S swarming steps while the cost function value decreases.

Bacterial Foraging Optimization Algorithm can be listed as follows:

```
Initialize bacteria colony
for l = 1 to  $N_{ed}$  (elimination dispersal loop)
  for k = 1 to  $N_{re}$  (reproduction loop)
    for j = 1 to  $N_C$  (chemotaxis loop)
      for i = 1 to  $S$  (each bacterium)
        perform tumble and change bacteria position to  $\theta^i(j+1, k, l)$ 
        compute cost function in new position
         $m = 0$ 
        while cost function value decreases and  $m < N_S$ 
          perform swarm and change bacteria position
          compute cost function in new position
        end while
      end for
    end for
  end for
end for
```

The position in which a bacterium reaches the lowest value of the cost function (greatest healthy status) is the solution of the optimization problem. In case of image registration, the size of the search space is equal to the number of parameters of the geometric transform.

3.3 Genetic Algorithms

Genetic Algorithm is an adaptive strategy used for global optimization problems. Inspired by population genetics, GA is based on a set of individuals in which the possible solutions of the problem are encoded as chromosome strings. The general structure of GAs is: (a) selection of the appropriate encoding method and fitness function, (b) generation of a random initial population and (c) the evolution loop of the algorithm: fitness function evaluations, application of genetic operators and creation of the new generation. After a number of generations, the population is expected to contain chromosomes that approximate the global maximum value of the fitness function. In each generation chromosomes with best fitness values are retained and generate offspring that replaces chromosomes with the lowest values of the fitness function. Genetic operators used for new generation creation are: selection, crossover and mutation.

1. Randomly generate a set of individuals.
2. Compute fitness for all individuals.
3. While the stop criterion and maximum generation number are not met
(Evolution loop)
 - 3.1 Apply reproduction
 - 3.2 Mutation
 - 3.3 Crossover.

In [13] it is proposed an IR procedure using the string encoding of chromosomes. The parameters of the geometric transform are encoded as bit fields in a 32 bit value. In the procedure described below, the real encoding is used and each chromosome is characterized by a number of real values equal to the number of geometric transform parameters. Discrete, average and simplex crossover operators are used depending on user defined probabilities (p_d , p_a and p_s).

4 Parallel approach for bio-inspired IR

Analyzing the optimization procedures execution, it must be noticed that most of the processing time is spent in the cost function evaluations. In case of BFOA based registration, about 99% of the execution time is spent in the cost evaluation function and more detailed, about 83% for mutual information computing and 16% of total time is spent in when the geometric transform is applied to source image. The same, in case of GA optimization: 96% of the execution time is spent in the cost evaluation (81% to compute mutual information and 15% to apply the geometric transform). Because both BFOA and GA procedures were executed using the sequential implementation, only about 25% of the computing power is used in case of a Core i5 processor.

To optimize the IR procedure parallel implementations based on the computing power of multi-core processors were proposed in [12].

A closer look at BFOA reveals that it contains 4 nested loops: elimination/dispersal, reproduction and chemotaxis for each bacterium in the colony. The body of the inner loop is executed $N_{ed} \times N_{re} \times N_C \times S$ times, which may be a fairly large number. In fact, the cost function evaluation is performed more than two times this number due to the fact that each bacterium may perform more swim steps in a single chemotactic step. While the calculations performed for each individual bacterium in the inner loop are independent, the bacteria colony may perform the chemotactic steps simultaneously.

In case of GA optimization, the cost function evaluation is called from two different places. First, it is called from the main evolution loop of the algorithm (about 41% of execution time) only for the new created chromosomes evaluation, and second, in the simplex crossover function (about 53% of execution time). In the first case, the cost function is called for all not already evaluated chromosomes, so this task is easily parallelized. In case of simplex crossover that involves more than one chromosome, the crossover function will be executed in parallel for each group of chromosomes [12].

The IR procedure that uses the Clonal Selection Algorithm was not parallelized because there are few tasks completely independent,

suitable for parallel execution.

The parallel implementation was evaluated on an Intel Core i5 3.10 GHz processor and is detailed in the next section.

5 Experiments

In this section a comparison of results obtained using image registration procedures based on Bacterial Foraging Optimization Algorithm, Clonal Selection Algorithm and Genetic Algorithm is presented. The optimization procedures were applied for area based registration and features based registration. In the second case the SIFT features are used.

The image registration procedure was tested on a large set of DICOM medical images from a database available at <http://www.osirix-viewer.com/datasets/> [21]. Below, only the results obtained using the *Brainix* image as model are described. It is a gray level image (8 bits per pixel) and size 256×256 pixels. *Brainix* is a MR image of a brain tumor.

The image registration procedure was applied also to the image after it was modified by adding “salt & pepper” noise.

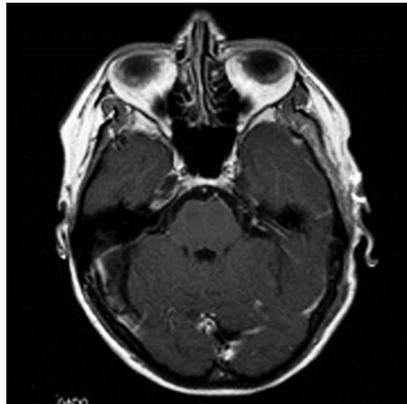


Figure 1. Original image BRAINIX (from [21])

The source images were obtained by applying a rotation (angle $\theta = 10^\circ$) against the rotation center ($c_x = -20$ and $c_y = 20$) followed by an isotropic scaling ($scale = 1.2$). While the transform is defined by 4 parameters, the search space in the optimization problem is R^4 . The actual value of the transform matrix is

$$T = \begin{bmatrix} \alpha & \beta & (1 - \alpha)c_x - \beta c_y \\ -\beta & \alpha & \beta c_x + (1 - \alpha)c_y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1.1818 & 0.2084 & -0.5322 \\ -0.2084 & 1.1818 & -7.8029 \\ 0 & 0 & 1 \end{bmatrix}, (7)$$

where $\alpha = scale \cdot \cos \theta$ and $\beta = scale \cdot \sin \theta$.

The inverse transform matrix is

$$T^{-1} = \begin{bmatrix} 0.8207 & -1.1447 & -0.6924 \\ 1.1447 & 0.8207 & 6.4807 \\ 0 & 0 & 1 \end{bmatrix}$$

that corresponds to an affine transform with the following parameters: $\theta' = -10^\circ$, $c'_x = -20$, $c'_y = 20$ and $scale' = 0.8333$.

To evaluate the similarity between model image and registered source image, the normalized mutual information is used.

The BFO parameters values used in the experiment are: bacteria colony size $S = 400$; number of chemotactic steps $N_c = 20$; maximum number of swim steps $N_s = 10$; number of reproduction steps $N_{re} = 16$; number of elimination / dispersal steps $N_{ed} = 2$; probability of dispersal $P_{ed} = 0.25$; length of the move step $C_i = 0.005$.

In case of GA optimization, the real encoding is used, and each chromosome is characterized by four real values representing the number of geometric transform parameters. Discrete, average and simplex crossover operators are used depending on user defined probabilities (p_d , p_a and p_s). The GA parameters are: number of generations $nGen = 500$ and number of chromosomes $nCr = 1500$. The crossover probabilities are: $p_d = 0.05$, $p_a = 0.15$ and $p_s = 0.2$.

Two different source images were used. The first one was obtained by applying the transform described above to the model image. The

second source image was obtained from the first one by applying “salt & pepper” noise. The signal-to-noise ratio in these images is about -1 dB.

In Figure 2, source images and some samples of registered images are presented.

In Tables 1 and 2 the results of both sequential and parallel area based registration are presented. The column ‘# cost eval’ shows the total number of cost function evaluations; the column ‘best cost eval’ shows the cost function evaluation in which the best value was obtained; column ‘ex_MI’ (expected MI) shows the expected cost value obtained by measuring the similarity between model image and the source image after the computed inverse transform was applied; column ‘c_MI’ (computed MI) shows the cost value obtained by measuring the similarity between model image and source image after the approximated inverse transform was applied. In both tables 1 and 2, ‘Brainix’ denotes the source image and ‘Brainix+SP’ denotes the source image altered by adding the “salt & pepper” noise.

Table 1. Results of area based image registration, sequential version

Mode	Image	Opt.	Time (sec)	# cost eval	best cost eval	ex_MI	c_MI
Seq	Brainix	BFOA	811.3	591686	469054	1.3218	1.3202
		GA	135.8	99148	63176	1.3218	1.3215
	Brainix+SP	BFOA	820.9	563561	546678	1.1394	1.1382
		GA	140.2	95371	60.348	1.1402	1.1387
Parallel	Brainix	BFOA	253.4	595365	249742	1.3218	1.3177
		GA	46.6	98894	79889	1.3218	1.3191
	Brainix+SP	BFOA	254.9	566433	269008	1.1398	1.1390
		GA	47.4	95042	86093	1.1398	1.1388

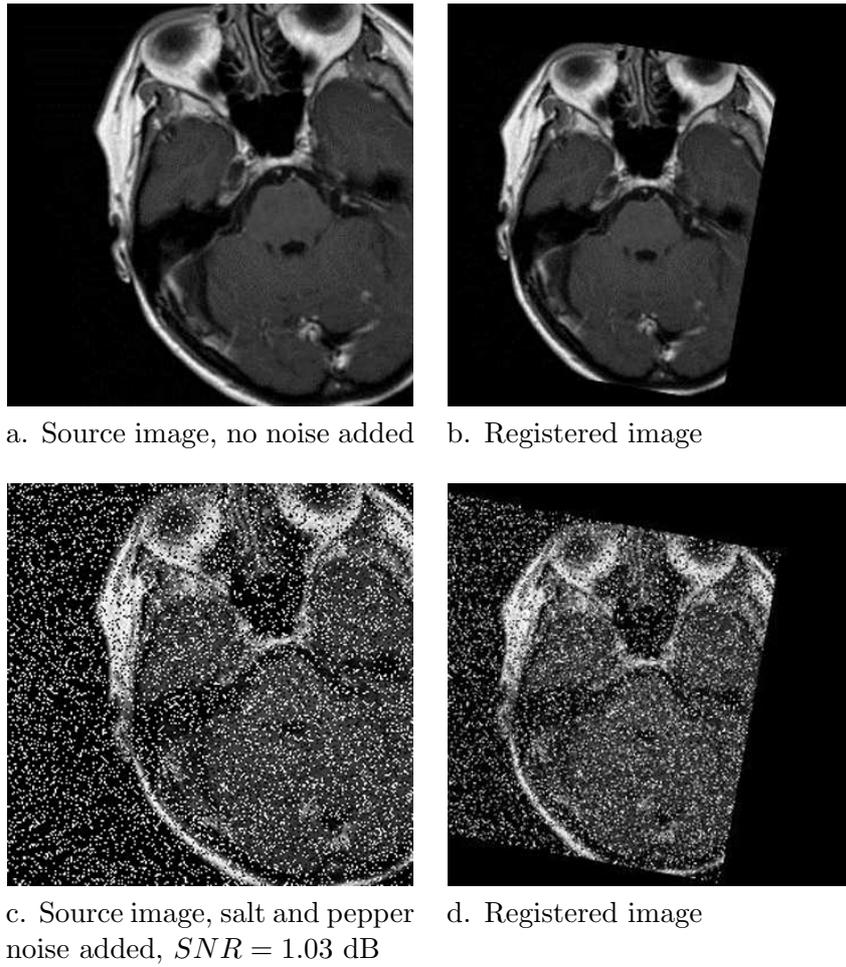


Figure 2. Results of image registration procedure

As noted in Table 1, the GA optimization requires about 6 times less cost function evaluations and this is the reason for which the IR procedure is faster in this case. In case of area based IR, the Clonal Selection Algorithm was not used for optimization because it requires about 100 times more cost function evaluations, consequently the registration is not achieved within a reasonable time interval. In fact, in case CSA is used as optimization method, the duration of features based registration is comparable to duration of area based registration using BFOA / GA as optimization algorithm.

On the Intel Core i5 processor which was used in experiments, the parallel implementations of the optimization procedures are about three times faster than the sequential versions, while the number of cost evaluations is close.

The expected value of similarity measure in Table 2 has different values due to the “salt & pepper” noise randomly added into images. The best values were obtained using sequential GA optimization in case of ‘*Brainix*’ image and parallel BFOA optimization in case of ‘*Brainix+SP*’ image.

In Table 2 the parameters of the approximated geometric transform are presented for both sequential and parallel implementation. The values determined by computing the inverse geometric transform are: $\theta' = -10^\circ$, $c'_x = -20$, $c'_y = 20$ and $scale' = 0.8333$.

In Table 3 the results of features based IR are presented. While in this case the cost function evaluation requires applying the approximated inverse transforms to a small number of pixel coordinates, the registration process is faster. The last column of Table 2 contains the number of correspondent key points between model and source images, i.e. the number of key points for which the geometric transform must be applied. If source image is not altered by noise, there are 188 correspondent key points. In case the source image is randomly altered by noise, the number of correspondences is between 11 and 16. But, features based registration is not a solution for noisy images, by increasing the noise in the source images, it is possible to don't find any correspondent key points pairs. This is the case for images obtained using different types of sensors or acquisition methods.

Table 2. Parameters of inverse affine transform computed using area based registration

Mode	Image	Opt.	c'_x	c'_y	θ'	$scale'$
Seq	Brainix	BFOA	-19.69	20.17	-10.02	0.83
		GA	-20.10	20.12	-9.99	0.83
	Brainix+SP	BFOA	-20.40	18.79	-9.98	0.83
		GA	-19.12	19.35	-10.07	0.83
Parallel	Brainix	BFOA	-20.16	19.90	-9.98	0.83
		GA	-20.26	20.30	-9.98	0.83
	Brainix+SP	BFOA	-19.85	19.41	-9.99	0.83
		GA	-19.10	20.13	-10.06	0.83

In Table 4 the values of the approximated transform parameters are presented. It must be noticed that in two cases the results are not so near to expected values: when the noisy image ‘Brainix+SP’ is used as source and in case CSA is used as optimization method.

The charts presented in Figure 4 show compare the sequential and parallel execution time for all the experiments presented before. It is obvious that for long tasks, as area based IR, the gain obtained by using the parallel versions is greater. For short tasks, as features based IR, the speedup is lower. This happens because the processing time becomes comparable to that of the synchronization tasks required by the parallel implementation and also because shorter time intervals are affected by all other processes and events that occur in the operating system. The results obtained in case of CSA optimization are not included in charts because in this case the duration was too long.

In Figure 4, the parallelization evaluation is presented. The most common evaluation of parallel algorithms is performed using the parallel efficiency $E = \frac{t_s}{t_p \times n}$, where t_s is the time used by the sequential version of the algorithm, t_p is the processing time for the parallel version and n is the number of used processors. As it was already said, better efficiency ($> 74\%$) is obtained for area based IR parallelization

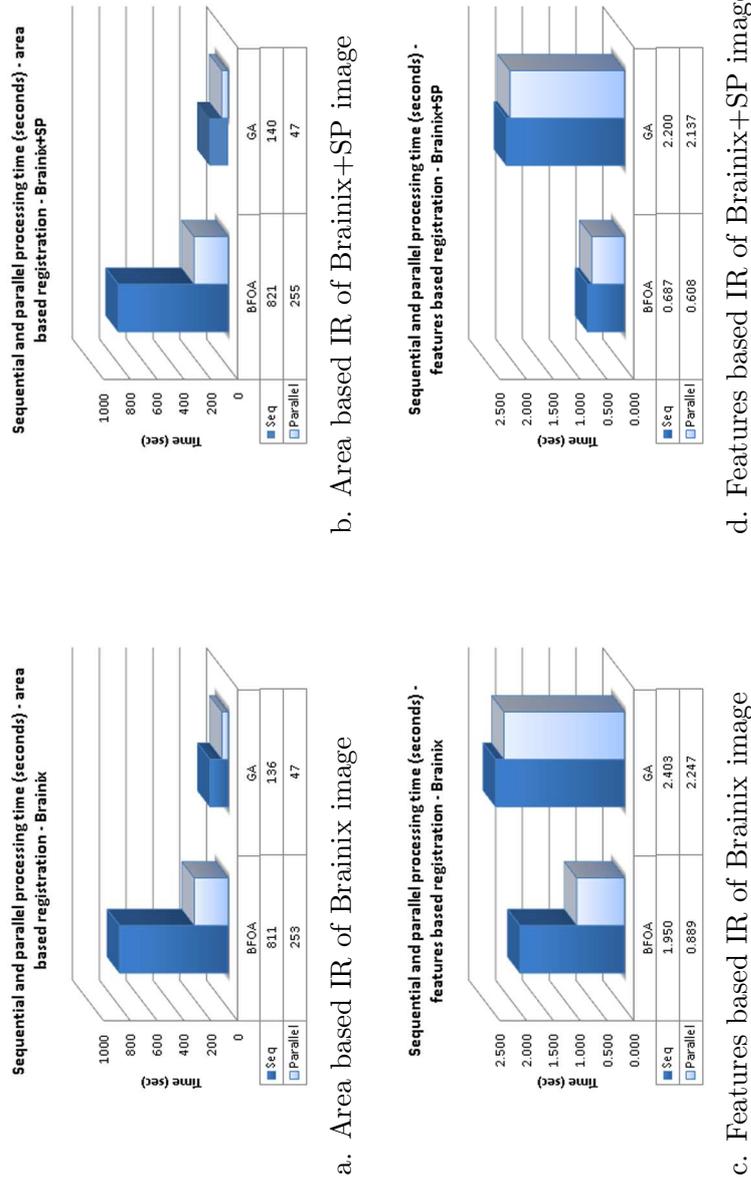


Figure 3. Processing time for different IR approaches

Table 3. Results of features based image registration, sequential version

Mode	Image	Opt.	Time (sec)	# cost eval	best cost eval	ex.MI	c.MI	key-points		
Seq	Brainix	BFOA	1.9	660726	391414	1.3218	1.3145	188		
		GA	2.4	118359	17701	1.3218	1.3173	188		
		CSA	353.3	73240500	-	1.3218	1.2948	188		
	Brainix	BFOA	0.7	662137	378711	1.1400	1.1365	15		
		+SP	GA	2.2	111775	40274	1.1405	1.1240	11	
		CSA	191.5	65687973	-	1.1392	1.1187	12		
Parallel	Brainix	BFOA	0.9	658684	636958	1.3218	1.3171	188		
		GA	2.2	115568	39971	1.3218	1.3168	188		
		CSA	139.9	71315426	-	1.3218	1.2916	188		
		Brainix	BFOA	0.6	640345	528443	1.1398	1.1234	12	
			+SP	GA	2.1	104900	26898	1.1389	1.1174	16
			CSA	102.9	70095946	-	1.1405	1.1305	11	

which has a longer execution time. In case of CSA optimization used for features based IR, the efficiency is between 47% and 68% which may lead to the conclusion that the procedure was not completely parallelized. In all other cases (BFOA and GA for features based IR), the lower parallelization efficiency is not relevant, while the execution time is too short.

The image registration procedures were implemented and tested in an image processing framework developed by authors of this paper. It is implemented in C++ as a Windows application and uses OpenCV library [22] for images manipulation and the parallel programming support available in Microsoft Visual Studio 2010 [23].

6 Conclusions

This paper is focused on the use of some bio-inspired optimization methods for medical images registration. Three different approaches

Table 4. Parameters of inverse affine transform computed using features based registration

Mode	Image	Opt.	c'_x	c'_y	θ'	$scale'$	
Seq	Brainix	BFOA	-20.85	20.14	-9.93	0.83	
		GA	-19.81	20.10	-9.99	0.83	
		CSA	-20.35	19.15	-9.97	0.83	
	Brainix+SP	BFOA	-20.01	20.20	-9.97	0.83	
		GA	-22.48	19.33	-9.96	0.84	
		CSA	-17.99	16.93	-10.22	0.84	
Parallel	Brainix	BFOA	-19.88	19.54	-9.99	0.83	
		GA	-19.82	20.01	-9.99	0.83	
		CSA	-17.33	20.11	-10.16	0.83	
		Brainix+SP	BFOA	-25.43	21.66	-9.45	0.83
			GA	-11.73	19.72	-10.77	0.83
			CSA	-20.31	20.64	-9.85	0.83

are presented: Bacterial foraging optimization algorithm, genetic algorithm and clonal selection algorithm. Since image registration may be a time consuming task, different optimization strategies were applied: the use of scale invariant features transform key points and full usage of computing power of multi-core processors. The obtained results may be summarized as follows:

- BFOA and GA allow to obtain comparable results in terms of registration precision;
- GAs perform faster the image registration about three times faster than BFOA;
- CSA is too slow for features based registration (comparable to area based IR combined with BFOA and GA) and also with lower precision, provided that algorithm's parameters were not enough tuned;

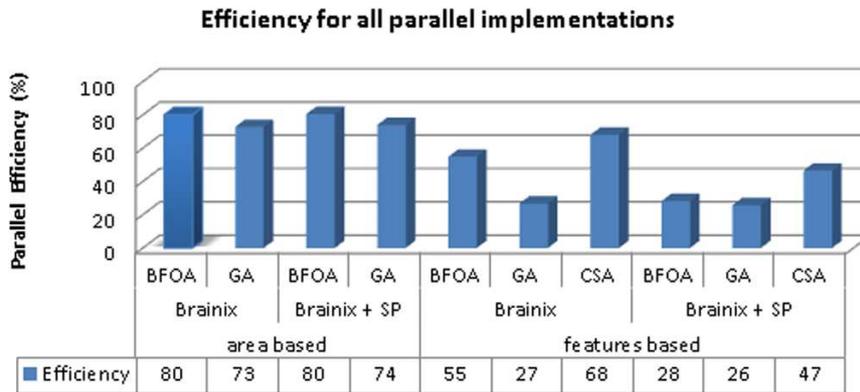


Figure 4. Parallel efficiency obtained in all experiments

- Even the features based IR performs faster, it's usage for multi-modal images is limited by the procedure's capability to find common and stable features in the images to be registered;
- Parallel implementations are suitable in image registration, while cost function evaluations are independent and time consuming tasks.

References

- [1] B. Zitova, J. Flusser. *Image registration methods: a survey*, Image and Vision Computing, 21, Elsevier, 2003, pp. 977–1000.
- [2] Y. Liu, K. M. Passino. *Biomimicry of Social Foraging Bacteria for Distributed Optimization: Models, Principles, and Emergent Behaviors*, Journal of Optimization Theory and Applications, Vol. 115, No. 3, 2002, pp. 603–628.
- [3] K.M. Passino. *Biomimicry of Bacterial Foraging for Distributed Optimization and Control*, IEEE Control Systems Magazine, June 2002, pp. 52–67.

- [4] K.M. Passino. *Biomimicry for Optimization, Control, and Automation*, Chapter 18: Cooperative Foraging and Search, Springer Verlag, 2005.
- [5] O. P. Verma, M. Hanmandlu, P. Kumar, S. Chhabra, A. Jindal. *A novel bacterial foraging technique for edge detection*, Pattern Recognition Letters, 32, Elsevier, 2011, pp. 1187–1196.
- [6] N. Sanyal, A. Chatterjee, S. Munshi. *An adaptive bacterial foraging algorithm for fuzzy entropy based image segmentation*, Expert Systems with Applications, 38, Elsevier, 2011, pp. 15489–15498.
- [7] P.D. Sathya, R. Kayalvizhi. *Modified bacterial foraging algorithm based multilevel thresholding for image segmentation*, Engineering Applications of Artificial Intelligence, 24, Elsevier, 2011, pp. 595–615.
- [8] Z. Yudong, W. Lenan. *Multi-resolution rigid image registration using bacterial multiple colony chemotaxis*, 5th International Conference on Visual Information Engineering, 2008, VIE 2008, pp. 528–532.
- [9] H. Costin, C. Rotariu. *PET and CT images registration by means of soft computing and information fusion*, Proceedings of the 1st WSEAS international conference on Biomedical electronics and biomedical informatics, 2008, pp. 150–161.
- [10] H. Costin, S. Bejinariu. *Medical Image Registration by means of a Bio-Inspired Optimization Strategy*, Computer Science Journal of Moldova, vol. 20, Nr. 2 (59), 2012, pp. 178–202.
- [11] H. Costin, S. Bejinariu. *Medical Signal Processing by Means of Immune Algorithms*, 4th IEEE International Conference E-Health and Bioengineering – "EHB 2013", Iasi, Romania, 21-23 Nov. 2013.
- [12] S. Bejinariu. *Image Registration using Bacterial Foraging Optimization Algorithm on Multi-core Processors*, Electrical and Elec-

- tronics Engineering (ISEEE), 2013 4th International Symposium on, October, 11-13, 2013. Galați, România.
- [13] R. Singhai, J. Singhai. *Registration of Satellite Imagery using Genetic Algorithm*, Proc of the World Congress on Engineering, WCE 2012, London, UK, Vol II.
- [14] S. Tsutsui, M. Yamamura, T. Higuchi. *Multi-parent Recombination with Simplex Crossover in Real Coded Genetic Algorithms*, Proc. of the Genetic and Evolutionary Computation Conference, pp. 657–664. Orlando, Florida, USA, 2000.
- [15] J.Brownlee. *Clever Algorithms. Nature-Inspired Programming Recipes*, lulu.com; 1st edition, 2012.
- [16] F.Campelo, F.G. Guimarães, H.Igarashi, J.A. Ramírez. *A Clonal Selection Algorithm for Optimization in Electromagnetics*, IEEE Transactions on Magnetics, Vol. 41, no. 5, 2005, pp.1736–1739.
- [17] M. Pavone, G. Narzisi, G. Nicosia. *Clonal selection: an immunological algorithm for global optimization over continuous spaces*, Journal of Global Optimization, Springer Science+Business Media, LLC. 2011.
- [18] E.D. Ülker, S. Ülker. *Comparison Study for Clonal Selection Algorithm and Genetic Algorithm*, International Journal of Computer Science & Information Technology (IJCSIT) Vol 4, No 4, 2012.
- [19] D. Lowe. *Distinctive Image Features from Scale-Invariant Keypoints*, International Journal of Computer Vision, 60(2), 2004, pp. 91–110.
- [20] S. Bejinariu, M.Costin, A. Ciobanu, S. Cojocaru. *Similarities Identification in Logo Images*, Proceedings IIS 2013, International Workshop on Intelligent Information Systems, Chisinau, Republic of Moldova, 2013, pp. 53–59.
- [21] *DICOM sample image sets*, <http://www.osirix-viewer.com/data-sets/>, last accessed on 1.06.2014.

- [22] G. Bradski, A. Kaehler. *Learning OpenCV. Computer Vision with the OpenCV Library*, O'Reilly Media, Inc., 2008.
- [23] C. Campbell, A. Miller. *Parallel programming with Microsoft Visual C++*, Microsoft Corporation, 2012.

Silviu Ioan Bejinariu, Hariton Costin, Florin Rotaru,
Ramona Luca, Cristina Niță, Camelia Lazăr

Received June 2, 2014

Institute of Computer Science, Romanian Academy,
Iasi Branch, Romania

E-mails: {*silviu.bejinariu, hariton.costin, florin.rotaru, ramona.luca, cristina.nita, camelia.lazar*}@iit.academiaromana-is.ro

A New Algorithm for Localized Motif Detection in Long DNA Sequences

Invited Article

Alin G. Voina, Petre G. Pop, Mircea F. Vaida

Abstract

The evolution in genome sequencing has known a spectacular growth during the last decade. One of the main challenges for the researchers is to understand the evolution of the genome and in particular to identify the DNA segments that have a biological significance. In this study we present a new algorithm – ADMSL – optimized for finding motifs in long DNA sequences and we emphasize some experiments done in order to evaluate the performance of the proposed algorithm in comparison with other motifs finding algorithms.

Index Terms: motifs search algorithms, motifs identification, transcriptions factor binding site, biological data analysis.

1 Introduction

The identification of novel cis-regulatory motifs in DNA sequences experienced a spectacular development in the recent years. As a consequence, an important number of algorithms have been developed with the scope to detect transcriptional regulatory elements from genes that belong to a specific genome [1].

The main scope of these algorithms is to identify the transcriptional regions and to find the motifs which are repeating most because those are good candidates for functional elements in genome. Phylogenetic footprinting is a particular method that is used to identify transcription factor binding sites in a set of orthologous noncoding DNA sequences.

The algorithms elaborated so far are capable of analyzing multiple DNA sequences and some of them can perform also over an entire gene. The process of regulating gene expression is an important challenge in molecular biology. The main task in this challenge is to identify the DNA binding sites for transcription factors. Computational methods have a special place in researcher's studies as are expected to offer the most promising results.

The problem of motifs detection can be formulated as: having a group of S sequences, search for a pattern M of length l which is spread more often. If the pattern M of length l is present in each sequence from the group of S sequences, then by enumerating the l letters of the pattern we obtain the regulatory element. The mutations of the nucleotides can affect the identification of transcription factor binding sites from a set of DNA sequences.

The identification of sequence motifs is an important step for understanding the process behind gene expression. A DNA motif is a short, well conserved pattern that usually has a biological significance [2]. Some of the motifs are included in complex RNA processes like transcription termination, mRNA processing, ribosome binding [3]. The length of the motif can vary from five base pairs (bp) to twenty (bp) and can be identified within the same gene or in different genes. Motifs can be classified based on their length but can be split also in palindromic motifs and gapped (space dyad) motifs [4]. We classify a motif as palindromic if its complementary read backwards is identical with the motif itself (e.g. 'AGAGCGCTCT' is a palindromic motif). Space dyed (gapped) motifs are usually formed from two sites of relatively short length, well conserved and usually separated by a spacer. The gap is usually located in the middle of the motif due to the fact that transcription factor (TF) usually binds as a dimer. The length of the sites where TF binds to the DNA varies from three to five bp which are usually well conserved.

In the past, binding sites determination was performed with gel-shift and footprinting methods or reported construct assays [3].

In the recent years, for determining motifs in a sequence or a set of sequences, computational methods are used increasingly more.

The development of DNA motifs search algorithms was materialized into more than seventy elaborated methods for motifs identification. A good part of these methods are based on phylogenetic footprinting and/or probabilistic models.

The algorithms dealing with motifs identification can be organized into three main groups:

- algorithms that use promoter sequences from co-regulated genes of a single genome;
- algorithms that use phylogenetic footprinting;
- a combination of the above algorithms.

In this study we present a new algorithm (ADMSL) for motifs identification in long DNA sequences and we compare the results with the ones obtained with six popular tools: MEME, Weeder, AlignACE, YMF, Scope and Improbizer which are presented in the table below (Table 1).

2 Motifs localization in long sequences

The detection of motifs in case of long-range regulatory sequences became a requirement in ChIP experiments [5] – especially when searching for vertebrate promoters. If we refer to long DNA sequences, some recent studies [6] [7], reported that stochastic patterns may behave as real motifs. This can lead to false positive motifs which can eclipse the motifs identified as real. The length of the analyzed DNA sequence has a large influence over memory and time requirements for algorithms that search for motifs.

The binding sites are specifically bound by one or more DNA-binding proteins and are usually localized in specific positions [5]. Most of the Transcription Factor Binding Sites-TFBS are positioned relative to TSS to allow the transcription factors to anchor at specific positions with respect to each other and the TSS [8]. For this particular situation, the detection of the motif can be performed by searching into

Table 1. Analyzed Tools-Operation Principles

Analyzed tool	Principle of functionality	Observations
AlignAce	It uses an iterative masking procedure together with Gibbs sampling.	The detection of motifs is accomplished using an iterative masking procedure [6].
<i>MEME</i>	Uses statistical modeling techniques.	Motif detection consists in performing expectation maximization from starting points derived from each subsequence occurring in the input dataset [15].
<i>Improbizer</i>	Uses Expectation Maximization.	In particular, Improbizer is using a variation of the expectation maximization (EM) algorithm [16].
<i>Weeder</i>	Consensus-based method.	It has options for “post-processing” i.e. analysis of location and significance of the motifs [17].
<i>YMF</i>	Finds motifs with the greatest z-score.	Identifies candidates for binding sites by searching for statistically over-represented motifs.
SCOPE	Uses three programs behind the scenes to identify different kind of motifs.	Utilizes three algorithms to identify sequence motifs: BEAM-finds non degenerate motifs, PRISM-finds degenerate motifs and SPACER – finds bipartite motifs [18].

an appropriate interval after the sequence is aligned relative to an anchor point. In this way, the regions that are not containing any motif are removed and the probability of reporting false positive motifs is decreased.

One solution would be to divide the long sequences into short overlapping sequences of the same length and to analyze each subsequence with a motif finding algorithm. But this approach can lead us into several problems:

- in most of the situations we have no prior information regarding the regulatory region where motifs may be localized;
- it is a big challenge to localize the motifs which are most significant for the whole DNA sequence when a considerable number of motifs were reported over a range of intervals;
- the length of the subsequences has a big influence over the motif identification process – in case of a short length the motif may not be visible and in case of a long length, the motif may be eclipsed;
- the analyzed sequence must be divided automatically; otherwise it will take considerable time and also may be predisposed to errors.

In the proposed algorithm of this research, we've taken the decision to not use subsequences of the original DNA sequence and to make the analysis over the entire sequence as we get it from genome repositories.

The problem of motif detection is well defined in the literature. One of the most common definitions is the one described in [9]. So, the main task is to determine all the instances of the pattern M of length l with d substitutions that occur into the set of analyzed sequences. The pattern M is known as a *motif* and each instance of the motif M represents a *binding site*.

Positional weight matrix (PWM) is another representation that can be used for motif detection, especially for the motifs that have particular instances localized over DNA sequences. For initial motif detection,

the consensus representation (l, d) proved to be more efficient, in particular for the motifs which are not having a consistent instance across the sequences [9].

The definition of motif referenced above is taking into consideration the fact that instances of a motif can be distributed over the entire sequence which is true, in particular for short sequences. For long sequences it is considered that most of the motif instances are found into a specific interval, relative to an anchor point (Figure 1).

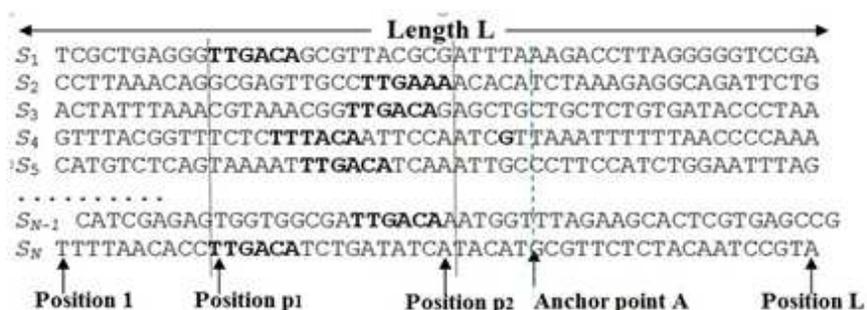


Figure 1. Motifs detection of pattern $(6,1)$ into a set of N sequences, each one of length L . The random pattern TTTAAA hides the real motif TTGACA

The problem of motif detection, in fact, is a variation of the above definition:

- for a set of S sequences $S = S_1, S_2, \dots, S_N$, each of length L , we have to find all instances of pattern M of length l across the interval (p_1, p_2) of the sequences S ;
- known values: S – the set of sequences, l – length of the pattern, d – maximum number of substitutions.

3 ADMSL – Algorithm description

The scope of the ADMSL algorithm is to identify candidate motifs from different intervals of the analyzed sequence and to report the ones which

have the best score. An exhaustive enumeration strategy will require the computation of the score functions for 4^l patterns from all possible intervals of the sequences which gets to an increased complexity. One of the goals of the ADMSL algorithm is to process faster long data sequences.

In order to make judgments if a candidate pattern is a motif or not we've used several score functions. The motifs are expected to be distinct from the general nucleotide composition of the regulatory sequences – known as *background* – since the transcription factors can distinguish them from other neighborhood nucleotide patterns. One score function that we've used in order to measure the difference between the motif M and the background model B is the relative entropy score [10] [11] measured as Kullback-Leibler divergence:

$$D_{norm}(M||B) = \frac{1}{l \ln 4} \sum_{i=1}^L \sum_b f_{b,i} \ln(f_{b,i}) - \frac{1}{\ln 4} \sum_b \bar{f}_b \ln p_b , \quad (1)$$

where

$$\bar{f}_b = \frac{1}{l} \sum_{i=1}^L f_{b,i}. \quad (2)$$

$f_{b,i}$ – represents the average frequency of occurrence on each nucleotide $b \in \{A, C, G, T\}$ at each position $i = 1, 2, \dots, l$. To measure the statistical deviation between the observed and expected occurrences of a motif we've used the Z -score function [12]:

$$Z - score = \frac{\binom{n}{NL} - e}{\delta} , \quad (3)$$

where n is the number of observed instances, N – is the total number of input sequences, L – is the average length of input sequence, e – is the probability to generate a motif instance according to the background model and δ – represents the standard deviation.

In order to make decisions regarding the distribution and localization of motifs into a certain interval (p_1, p_2) we've used the following score function defined by the mathematical definition from relation (4):

$$D(\hat{p}||p_0) = \hat{p} \ln\left(\frac{\hat{p}}{p_0}\right) + (1 - \hat{p}) \ln\left(\frac{1 - \hat{p}}{1 - p_0}\right), \quad (4)$$

where \hat{p} – represents the observed proportion of the motifs that is found in (p_1, p_2) interval, $(1 - \hat{p})$ – the observed proportion of the motifs that lies outside of (p_1, p_2) ; p_0 and $1 - p_0$ are the proportions that correspond to uniform distribution.

The combined score function may be calculated as a sum of the above scoring functions (Hamming measure) or as an Euclidean measure – root mean square of the above score functions.

The algorithm contains several optimizations which are presented in the next paragraphs.

One of the optimizations that we've done is to create a position dictionary. The main role of the dictionary is to optimize the computation of the number of candidate pattern instances from a specific position interval of the sequence. The dictionary is formed from all unique character arrays, with the length l , identified in input sequences. One of the particularities of the dictionary is the fact that the patterns which overlap are excluded: e.g. if the array 'ACACACAC' is found in each input sequence and we are interested to find just the patterns of 4 nucleotides length, then into the dictionary we'll have just two instances for the pattern 'ACAC' instead of three. Another particularity of the proposed dictionary is the fact that the patterns which are having a Hamming distance d or lower than d are interconnected. This interconnection allows a fast enumeration of all instances for each pattern of a specific length.

Another optimization that we've used in the algorithm is to accelerate the calculation of the score functions that we've used. Score functions calculations for each candidate pattern in all positions intervals (p_1, p_2) , where $0 \leq p_1 \leq p_2 \leq L$, will be ideal. In the current algorithm implementation we've taken into consideration the intervals (p_1, p_2) ; $p_1 < p_2$, $p_1, p_2 \in \{0, i, 2i, 3i, \dots, L\}$, where i represents the step size of the search. The score functions are being determined individually for each position interval. The score for a long interval can be directly determined from the scores of the shorter intervals from

which it is formed. The necessary computations are made in two steps: the score functions for all intervals of size i are being computed in the first step, then in the second step, the scores for longer intervals are computed from the scores of the constituent intervals obtained in first step. The most time consuming is the first step; in the second step the time and complexity are significantly reduced due to the fact that it is just a direct computation from the results obtained previously. This is why the proposed computational method is efficient also in case of long sequences.

The filtering of the similar patterns is another optimization that accompanies the proposed algorithm. As the scores of the candidate patterns are being determined for different intervals, the algorithm is maintaining a list with the scores in descending order. The similar patterns which are having a relative low score, and the ones which have position intervals which overlap, are removed from the list of possible motifs. In this way we maintain only the n motifs where n is user defined and represents a percentage from the total number of candidate motifs. These filters are leading to an important reduce of memory requirements for ADMSL algorithm. The similarity between two patterns of length l is evaluated by using the Needleman-Wunsch algorithm for global alignment. The similarity score is evaluated based on length l .

At each run, the ADMSL algorithm finds motifs for specific values of l and d . To combine the results at each run of the algorithm, for different (l, d) values, a post processing algorithm is needed. Since the score functions used in ADMSL algorithm don't depend on l or d , the motifs with different values for length l and substitutions number d , can be compared directly based on their scores. The motifs which are having a similar pattern can be determined using Needleman-Wunsch alignment algorithm. In this way, if we build motif groups with a similarity greater than 65% (relatively measured for the shortest motif), the motif with the lowest score is being removed. If two motifs have a high similarity (greater than 90%) and localization intervals are overlapping, these are combined into a single motif which has as localization interval the union of the two intervals.

4 Experiments and Results

The first test that we've done with the scope to get an overview of the ADMSL algorithm was by generating with [13] a dataset which contains 50 DNA sequences, each of them of 3000 nucleotides length. Randomly, we've inserted the motif GCATG (5,1) in 75% of the sequences at different positions. The obtained sequences were analyzed using ADMSL configured to search for motifs of length $l = 5$ and a maximum of $d = 1$ substitutions. The motif instances have been determined by ADMSL algorithm as localized in [900, 1500] interval.

From the analysis of other researchers [7] [14], the motif (5,1) is a subtle motif and is almost impossible to detect through a sequence of 3000 nucleotides because there actually are like a few thousands possible random motifs. The first ten motifs detected by ADMSL (together with the afferent scores) are presented in Table 2.

Table 2. The first 10 motifs reported by ADMSL algorithm when running over a dataset of 50 DNA sequences of 3000 nucleotides each

Pattern	Interval	SER	SSR	SIS	Score
GCATG	[900, 950]	0.469	0.345	0.432	1.246
CGCGA	[400, 450]	0.471	0.325	0.423	1.219
GTCGA	[900, 950]	0.424	0.342	0.359	1.125
ATCGT	[1200, 1250]	0.425	0.297	0.398	1.12
CTTCG	[2100, 2150]	0.378	0.432	0.295	1.105
TACGC	[2850, 2900]	0.421	0.305	0.292	1.018
CCGAT	[2650, 2700]	0.397	0.297	0.291	0.985
TACCG	[1800, 1850]	0.345	0.348	0.287	0.98
CGTCG	[900, 950]	0.451	0.276	0.251	0.978
CGATC	[950, 1000]	0.411	0.324	0.237	0.972

The pattern (5,1) was correctly identified as the most prominent motif and the localization interval was detected with accuracy.

This first test was performed to get an overview of the ADMSL

performance before getting to more representative tests.

In the next paragraphs we'll present the ADMSL performance in case of short sequences, long sequences and real sequences.

4.1 Short DNA sequences

The tests performed on short DNA sequences have the role to evaluate the detection accuracy of ADMSL algorithm and to emphasize the robustness of the algorithm. Each set of sequences was having N sequences of nucleotides, each of them with a length $L < 1000$, randomly generated using [13]. All of the sequences were artificially implanted with a motif M which has the characteristics $l = 6$ and $d = 1$ along of a randomly position interval (p_1, p_2) . We have generated 10 datasets by varying the number of sequences N and the length of the sequence L . The parameters and their values are presented in Table 3.

Table 3. The value of the parameters used in performance analysis over short DNA sequences

Parameter	N	L	l	d
Value	10..50	200-1000	6	1

In Figure 2 it is presented the detection accuracy of the ADMSL algorithm in case of short DNA sequences (randomly generated) implanted with motif $M = 'CGATGC'$.

The ADMSL algorithm was configured to report the first 50 possible motifs for each DNA sequence. From the reported motifs, we've chosen the motif most closely of the implanted motif M and we had retained – based on the score – the position occupied in the list of reported motifs.

The motifs reported in this case are presented in Table 4.

From Figure 2 we can observe that the detection accuracy is decreasing while the length of the sequence is increasing but the average detection accuracy value was around 83.6%. So, we can observe that the detection accuracy of the ADMSL algorithm is relatively high. This is because the ADMSL algorithm is not dependent upon the length of

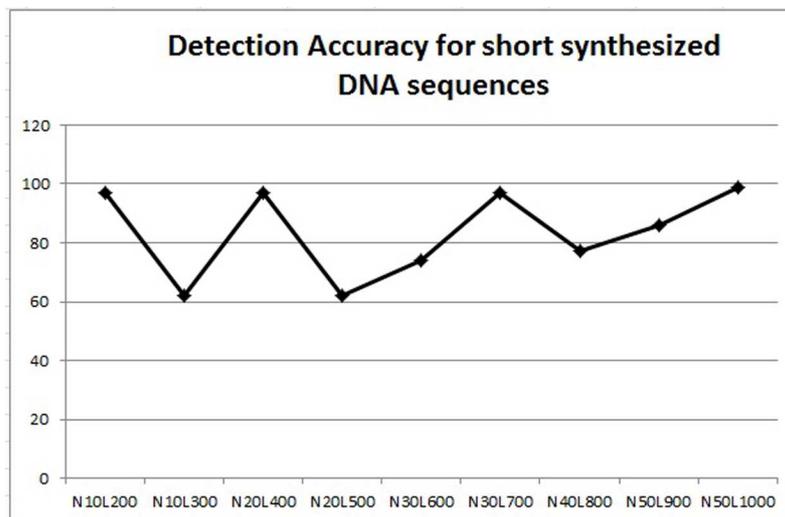


Figure 2. ADMSL detection accuracy in case of short DNA sequences randomly generated

Table 4. Detected motifs by ADMSL algorithm in case of short DNA sequences. Detection accuracy for the considered dataset

DNA Se-quence	Motif	Position (in the list of 50 motifs reported)	Detection accuracy
N10L200	GCATGC	3	97%
N10L300	GCATGC	39	62%
N20L400	TCATGC	4	97%
N20L500	ATGCTT	39	62%
N30L600	CATGCG	27	74%
N30L700	GCATGC	5	97%
N40L800	GTGCTA	24	77%
N50L800	CATGTA	16	85%
N50L900	CCATGC	15	86%
N50L1000	ATGCGT	2	99%

the sequence but rather of the motif localization interval. The localized search reduces the number of concurrent random patterns and increases the possibility of comparing motifs.

4.2 Long DNA sequences

The analysis of detection accuracy in case of ADMSL algorithm for long sequences was performed using data sequences as it follows:

- we've generated using [13] ten data sets of 30 random sequences by varying the length of the sequences from 1000 to 6000 of base pairs;
- in each data set we've randomly inserted, in the interval position [200-800], the motif *CATGCT*.

The ADMSL algorithm was executed directly on the sequences previously obtained, with a maximum length of the interval set to 500 nucleotides. We must specify that the fragmentation of the analyzed sequences was not needed (even if their length hit almost 180000 nucleotides), because the ADMSL algorithm automatically determines the localization interval of the motif.

In case of other motif detection algorithms (like MEME, Weeder) there is necessary a fragmentation of the long sequences and to maintain the accuracy, these fragments need to have an overlapping rate of about 50%.

Each run of the ADMSL algorithm was performed using the parameters specified in Table 5.

Table 5. The value of the parameters used in performance analysis over long DNA sequences

Parameter	N	L	l	d
Value	30	1000-6000	6	1

The detection accuracy of the randomly implanted motif, in case of the long sequences is presented in Figure 3. The detection accuracy was evaluated as the detection sensitivity based on the combined score function of the reported motif.

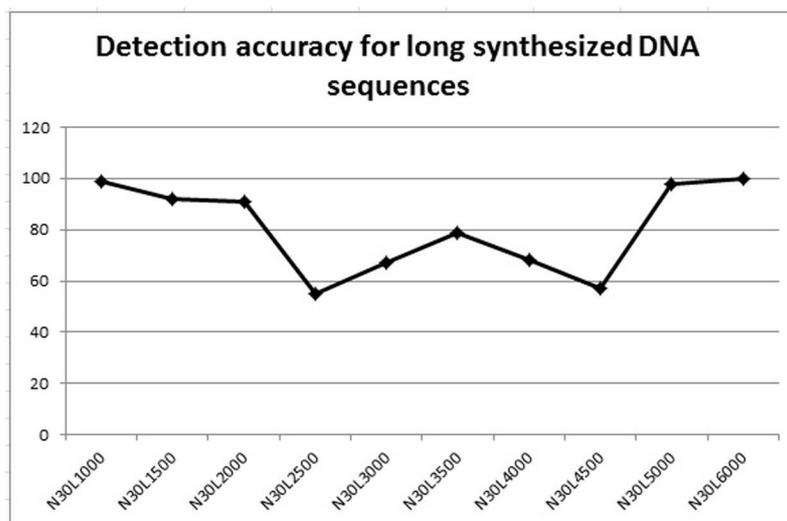


Figure 3. ADMSL detection accuracy in case of long DNA sequences randomly generated

As we can observe, the detection accuracy is maintained over 60% also in case of the long sequences. It is interesting to observe the fluctuation of the accuracy detection once the length of the analyzed sequences grows – we can notice that the accuracy value is increasing and decreasing randomly for the sequences that have a length between 200 and 4500 nucleotides. These fluctuations appeared due to the fact that we had randomly inserted the implanted motif and also because the analyzed sequences were randomly generated.

The motifs detected by ADMSL algorithm, as being the most closest to the implanted motif, are presented in Table 6.

In Table 6, we can observe that motifs similar to the implanted motif were detected and reported in localization intervals where the in-

Table 6. Detected motifs by ADMSL algorithm in case of short DNA sequences. Localization interval

DNA Se- quence	Motif	Position (in the list of 50 motifs reported)	Localization interval
N30L1000	CATGCG	2	[300,550]
N30L1500	GCATGC	9	[250,300]
N30L2000	CATGCG	10	[250,600]
N30L2500	CATGCT	48	[450,500]
N30L3000	ATGCTC	34	[1050,1100]
N30L3500	ATGCTG	22	[2700,2750]
N30L4000	GCATGC	33	[1550,1600]
N30L4500	ACATGC	44	[300,350]
N30L5000	GCATGC	3	[400,450]
N30L6000	CATGCA	1	[250,350]

sersion of the random motif did not occurred – the motif was implanted only in the positions interval [200, 800]. Those reported motifs have been detected as valid motifs because they were present in the initial sequences, randomly generated.

4.3 Real DNA sequences

Motifs detection in long regulatory sequences it is an actual requirement especially in ChIP experiments for determining the promoters for vertebrates [5]. Some recent studies [7] [14] are highlighting that random patterns from DNA sequences may become remarkable as if the real motifs. In this specific case, the algorithms used for motif detection are returning false positives hiding the real motif. For most of the algorithms, the necessary resources – memory requirements and execution time – are proportionally increasing with the size of the analyzed sequence.

In the literature, it is known the fact that the motif instances are

found to be localized at specific positions, relatively to a reference position (anchor point) [5]. Most of the transcription factors are being localized relatively to a transcription start site to allow the transcription factors to be localized in specific positions. In these conditions, the motif detection can be done by searching into a specific interval after the alignment of the sequences relatively to the anchor point.

The localization of motifs has an important advantage by removing the regions which are not containing motifs and by decreasing the possibility of returning false positives.

One possibility is to divide the DNA sequences into short overlapping subsequences of the same size. Some problems may occur:

- in most of the cases we don't have prior information regarding the regions where the motifs are distributed;
- in case of a big number of reported motifs in a range of intervals it is really a challenge to identify and extract those motifs which have the greatest importance for the analyzed sequence;
- depending on the chosen length for the sequences the motifs might not be so obvious if the length is short and might be poorly demarcated if the length is too big;
- the division of the analyzed sequence in subsequences must be done automatically otherwise it will require time and it will be more susceptible to errors.

In the performance evaluation for real data sequences, to not disadvantage any of the algorithms, we've chosen to not split the sequence into subsequences. The analysis was performed on the entire sequence in order to make judgments regarding performance directly over long DNA sequences as we found them in genomic repositories.

A big challenge in this research was to choose the right datasets with the scope to not favor or disfavor any of the algorithms that we've used in comparison with ADMSL. Tompa [12] presents a few solutions for DNA datasets selection but each of them have several drawbacks. In order to pass these drawbacks we've used transcription

factors reported as real in TRANSFAC repository. From the biological database previously mentioned, we had chosen only the transcription factors which were having also a consensus sequence defined.

We've executed tests on different sequences corresponding to the following species: *Saccaromyces Cerevisiae*, *Drosophila Melanogaster* and *Homo Sapiens*. All the algorithms used in this assessment (ADMSL, MEME, AlignAce, YMF, Improbizer, Weeder and SCOPE) have been configured to detect motifs that have a length in the range of six to ten nucleotides. In this performance evaluation we took into account the first ten motifs detected by each of the analyzed algorithm. In order to obtain an overview of each algorithm we've run the applications/algorithms over each dataset. Besides the proposed algorithm – ADMSL – all others have been used without modifying the source code of the applications and the evaluations were performed over their official web sites or by running the application locally.

In the next figures we present the detection accuracy of the considered algorithms.

If we consider the *Drosophila Melanogaster* dataset (Figure 4), a big majority of the reported motifs had a length between 3bp (Improbizer) and 10bp (MEME, YMF).

Most of the motifs reported had a corresponding real transcription factor in TRANSFAC database (the motifs reported by Improbizer were not found in TRANSFAC database – that's why the accuracy is set to 0). From the performance point of view we can confirm that ADMSL had reported the most motifs for which we had found a corresponding transcription factor in TRANSFAC database. Also, we've noticed that YMF and SCOPE had good performances.

For *Homo Sapiens* dataset (Figure 5) we've used sequences with more than 36000 nucleotides. We've observed that the algorithm had reported motifs that we had found as transcription factors in biological database (TRANSFAC). Over 90% of the motifs reported by ADMSL were identified as real transcription factors in TRANSFAC genome repository. Once the length of the analyzed sequence had increased, the number of false positives had also increased.

Also in case of *Saccaromyces Cerevisiae* dataset (Figure 6) the

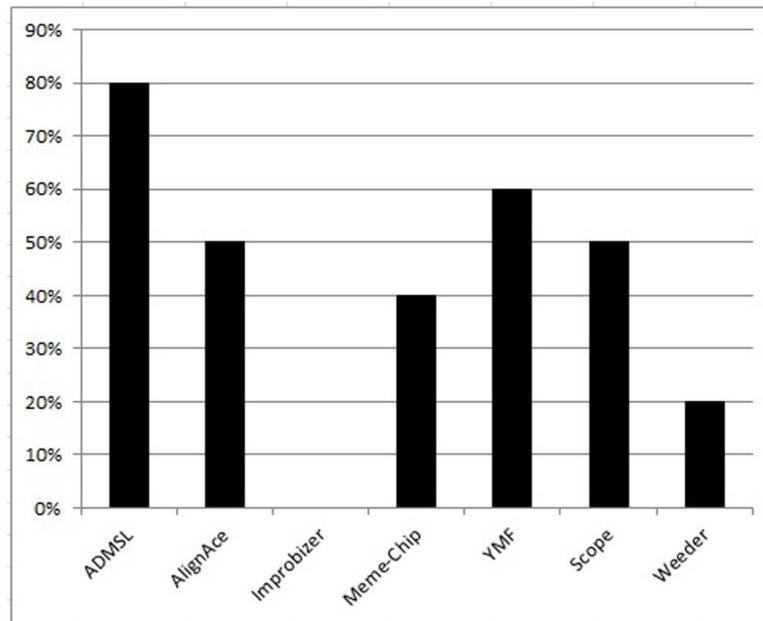


Figure 4. Detection accuracy for Drosophila Melanogaster dataset

ADMSL algorithm had proved to be more accurate than the other analyzed algorithms. MEME and YMF had accuracy close to ADMSL algorithm.

5 Conclusions

The main purpose of this research was to design and develop a new algorithm for detecting DNA motifs especially in long sequences where the performance of existing applications is relatively poor. The algorithm proposes an innovative way for detection and localization of DNA motifs by combining multiple score functions to evaluate the existence of a motif.

ADMSL had been optimized to fast process long DNA sequences. The results obtained on synthetic or real data confirmed us that

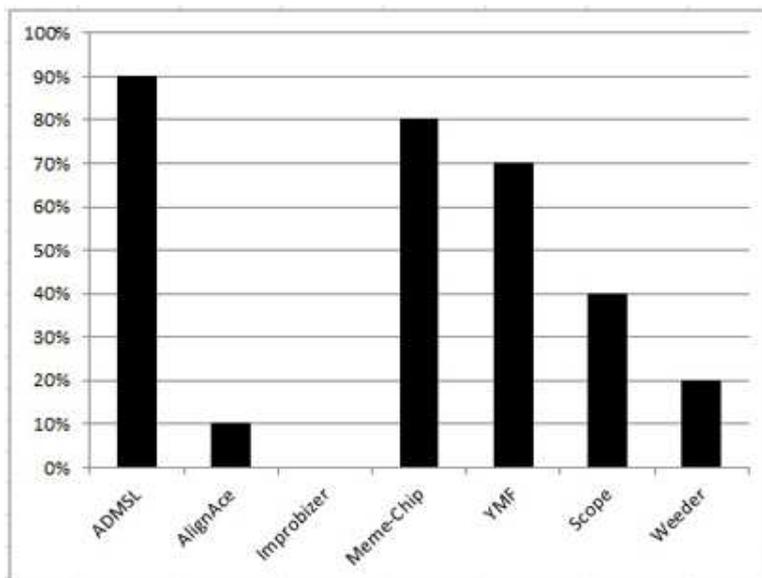


Figure 5. Detection accuracy in case of Homo Sapiens dataset

ADMSL has a definite advantage beside other algorithms due to the detection accuracy of the motifs in long DNA sequences.

In the recent years, considerable efforts were made in elaborating computational methods and more and more species have a complete DNA sequence. Nevertheless, the identification of the elements that are part of the cis-regulatory process continues to be an important challenge for scientists.

At the beginnings, the algorithms focused on motifs searching were combining the phylogenetic data with co-regulated genes in order to find regulatory motifs. In the present, most of the algorithms are oriented to computational methods and researchers are designing new approaches to better identify the motifs from the analyzed DNA sequences.

Due to the big number of algorithms and multitude of the methods designed for motif identification, for a user, it will be helpful a set of

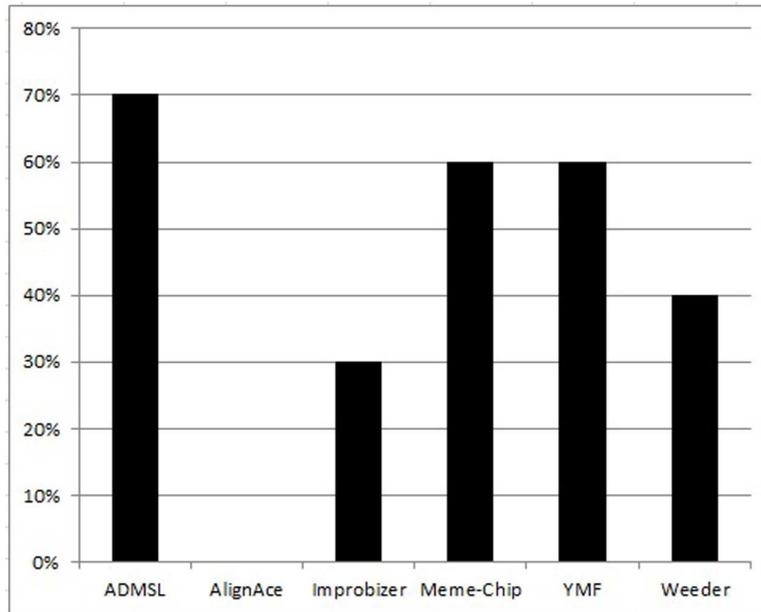


Figure 6. Detection accuracy in case of *Saccaromyces Cerevisiae* dataset

instructions for choosing the best algorithm/method before starting the analysis of the DNA sequence.

One of the drawbacks when providing instructions for deciding to a method or another is the number of settings and parameters which need to be chosen for each algorithm. The main advantage of ADMSL algorithm is the fact that the DNA sequences, even if they have a considerable length, don't need to be divided in order to obtain motif localization information. Another plus of ADMSL algorithm is the fact that needs just a few parameters (e.g. length of the search motif, allowed substitutions, size of interval search) which have also default values set in the application that runs ADMSL algorithm. In this way a user can obtain a first set of results with a minimum effort.

Performance evaluation of a motif search algorithm by comparing with other algorithms is especially problematic. This is because we

don't have yet a complete understanding of the process that regulates gene activity and expression. Also, there is no standardized model against to evaluate the efficiency of an algorithm. In the tests done in this research we must consider the fact that in case of the other algorithms used in comparison with ADMSL, the parameters were set with values to reflect as much as accurate the configurations done for ADMSL algorithm. Because this was done through human interaction – it is susceptible to errors.

Most of the algorithms used in comparison to evaluate the performance of ADMSL algorithm, have good results in case of lower organisms, especially when they are set to report short motifs (of 6-8 nucleotides). The ADMSL algorithm, through the computed score functions, highlights the motifs conservation through different species. The performances of ADMSL algorithm have proved to be much better especially for long DNA sequences like the ones that we've analyzed from human genome.

References

- [1] Martin Tompa Mathieu Blanchette. *Discovery of Regulatory Elements by a Computational Method for Phylogenetic Footprinting*, Genome Res, pp. 739–748, 2002.
- [2] Patrik D'haeseleer. *What are DNA motifs?*, Nature Biotechnology, vol. 24, pp. 423–425, 2006.
- [3] Hubbard TJ. Down TA, *NestedMICA: sensitive inference of over-represented motifs in nucleic acid sequence*, Nucleic Acids Res., pp. 1445–53., 2005.
- [4] Ho-Kwok Dai Modan K Das. *A survey of DNA motif finding algorithms*, BMC Bioinformatics, 2007.
- [5] Ankush Mital, Wing-Kin Sung Vipin Narang. *Localized Motif discovery in gene regulatory sequences*, Bioinformatics, vol. 26, no. 9, 2010.

- [6] F.P. Roth, J. D. Hughes, P. W. Estep, G.M. Church. *Finding DNA Regulatory Motifs within Unaligned Non-Coding Sequences Clustered by Whole-Genome mRNA Quantitation*, Nature Biotechnology, pp. 939–945, 1998.
- [7] Martin Tompa Jeremy Buhler. *Finding Motifs using Random Projections*, Journal of Computational Biology, vol. 9, no. 2, pp. 225–242, 2002.
- [8] S.T. Smale, J.T. Kadonaga. *The RNA polymerase II core promoter*, Annu Rev Biochem, vol. 72, pp. 449–479, 2003.
- [9] P.A. Pevzner, M.Yu. Borodovsky, A.A. Mironov. *Linguistics of nucleotide sequences*, J Biomol Struct Dyn, vol. 6, pp. 1013–1026, 1989.
- [10] G. Thijs, M. Lescot, K. Marchal, S. Rombauts, B. De Moor, P. Rouze, Y. Moreau. *A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling*, Bioinformatics, pp. 1113–1122, 2001.
- [11] Stormo G.D. *DNA binding sites: representation and discovery*, Bioinformatics, vol. 16, no. 23, 2000.
- [12] Tompa M. *An exact method for finding short motifs in sequences, with application to the ribosome binding site problem*, in Proceedings of the Seventh International Conference on Intelligent Systems on Molecular Biology, 1999, pp. 262–271.
- [13] http://www.bioinformatics.org/sms2/random_dna.html
- [14] U. Keich, P.A. Pevzner. *Finding motifs in the twilight zone*, Bioinformatics, vol. 18(10), no. 1382–1390.
- [15] Elkan C Bailey TL. *Unsupervised learning of multiple motifs in biopolymers using expectation*, Machine Learning, pp. 51–80, 1995.
- [16] W.Y. Ao, J. Gaudet, W.J. Kent, S. Muttumu, S.E. Mango. *Environmentally Induced foregut remodelling by PHA-4/FoxA and DAF-12/NHR*, Science 305, pp. 1743–1746, 2004.

- [17] G. Pavessi, P. Mereghetti, G. Mauri, G. Pesole. *Weeder Web: discovery of transcription factor binding sites in a set of sequences from co-regulated genes*, Nucleic Acid Res. 32, pp. 199-203, 2003.
- [18] Jonathan M. Carlson, Arijit Chakravarty, Charles E. DeZiel, Robert H. Gross. *Scope: a web server for practical de novo motif discovery*, Nucl. Acids Res, pp. 259-64, Jul 2007.

Alin G. Voina, Petre G. Pop, Mircea F. Vaida

Received May 19, 2014

Technical University of Cluj-Napoca,
Communications Department,
26-28 George Baritiu St., Room 364
400027 Cluj-Napoca, Romania,
Tel: +40-264-401226, Fax: +40-264-597083
E-mails: alin.voina@com.utcluj.ro
petre.pop@com.utcluj.ro
mircea.vaida@com.utcluj.ro