

## Article

# Various Approaches to the Quantitative Evaluation of Biological and Medical Data Using Mathematical Models

Mária Ždímalová <sup>1,\*</sup>, Anuprava Chatterjee <sup>2</sup>, Helena Kosnáčová <sup>3,4</sup> , Mridul Ghosh <sup>5</sup>, Sk Md Obaidullah <sup>6</sup> , Martin Kopáni <sup>3,7</sup> and Daniel Kosnáč <sup>3,7</sup>

- <sup>1</sup> Department of Mathematics and Descriptive Geometry, Faculty of Civil Engineering, Slovak University of Technology in Bratislava, 81005 Bratislava, Slovakia
  - <sup>2</sup> Department of Electrical Engineering, Indian Institute of Technology Kharagpur, Kharagpur 721302, India; anuprava.livetowin@iitkgp.ac.in
  - <sup>3</sup> Department of Simulation and Virtual Medical education, Faculty of Medicine, Comenius University in Bratislava, 81372 Bratislava, Slovakia; helena.svobodova@fmed.uniba.sk (H.K.); martin.kopani@fmed.uniba.sk (M.K.); daniel.kosnac@fmed.uniba.sk (D.K.)
  - <sup>4</sup> Department of Genetics, Cancer Research Institute, Biomedical Research Center, Slovak Academy of Sciences, Dúbravská Cesta 9, 84505 Bratislava, Slovakia
  - <sup>5</sup> Department of Computer Science, Shyampur Siddheswari Mahavidyalaya, Howrah 711312, India; mridul@ssmahavidyalaya.edu.in or mridulxy@gmail.com
  - <sup>6</sup> Department of Computer Science and Engineering, Aliah University, Kolkata 700160, India; sk.obaidullah@aliah.ac.in
  - <sup>7</sup> Institute of Medical Physics, Biophysics, Informatics and Telemedicine, Faculty of Medicine, Comenius University in Bratislava, 81372 Bratislava, Slovakia
- \* Correspondence: maria.zdimalova@stuba.sk or zdimalova@math.sk



**Citation:** Ždímalová, M.; Chatterjee, A.; Kosnáčová, H.; Ghosh, M.; Obaidullah, S.M.; Kopáni, M.; Kosnáč, D. Various Approaches to the Quantitative Evaluation of Biological and Medical Data Using Mathematical Models. *Symmetry* **2022**, *14*, 7. <https://doi.org/10.3390/sym14010007>

Academic Editor: Camelia M. Pinte

Received: 24 November 2021

Accepted: 9 December 2021

Published: 22 December 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Abstract:** Biomedical data (structured and unstructured) has grown dramatically in strength and volume over the last few years. Innovative, intelligent, and autonomous scientific approaches are needed to examine the large data sets that are gradually becoming widely available. In order to predict unique symmetric and asymmetric patterns, there is also an increasing demand for designing, analyzing, and understanding such complicated data sets. In this paper, we focused on a different way of processing biological and medical data. We provide an overview of known methods as well as a look at optimized mathematical approaches in the field of biological data analysis. We deal with the RGB threshold algorithm, new filtering based on the histogram and on the RGB model, the Image J program, and the structural similarity index method (SSIM) approaches. Finally, we compared the results with the open-source software. We can confirm that our own software based on new mathematical models is an extremely suitable tool for processing biological images and is important in research areas such as the detection of iron in biological samples. We study even symmetric and asymmetric properties of the iron existence as a design analysis of the biological real data. Unique approaches for clinical information gathering, organizing, analysis, information retrieval, and inventive implementation of contemporary computing approaches are all part of this research project, which has much potential in biomedical research. These cutting-edge multidisciplinary techniques will enable the detection and retrieval of important symmetric and asymmetric patterns, as well as the faster finding of pertinent data and the opening of novel learning pathways.

**Keywords:** mathematical modeling; mathematical biology; segmentation; biological data; data analysis; image analyses; segmentation; filtering

## 1. Introduction

Massive and complicated data sets have become very widely accessible, necessitating the use of highly modern, intelligent, and computerized methodologies to study them. To predict unique symmetric and asymmetric trends, cutting-edge approaches for saving, analyzing, and understanding such complicated data sets are becoming increasingly important.

In the previous decade, the amount of biomedical data (both organized and unstructured) has grown at an accelerating rate. Considering the enormous growth in data gathering and interpretation, the medical and healthcare scientific community has only now begun to realize the revolutionary potential of this data. This study incorporates creative ways for acquiring, structuring, analyzing, and extracting information from medical data, as well as innovative applications of current mathematical strategies that have enormous prospects for biomedical investigation. Such cutting-edge multidisciplinary solutions will make it easier to identify and obtain valuable symmetric and asymmetric structures, as well as speed up the finding of useful data. Biomedical data of scientific programs, for instance, is interpreted using quantitative and computer frameworks. Modeling of complicated biological systems is possible because of the use of powerful computer frameworks, enabling the hypotheses formulation and analyses of symmetric and asymmetric patterns in the biological data. Experiments can be designed to question or confirm the hypothesis. Regarding better accessibility to and transfer of ideas, systems that are properly connected to biological databases are required [1–3]. Biotech and information systems breakthroughs have generated massive volumes of data and are hastening the process of biological component information extraction. These advancements are altering the manner biomedical study is conducted.

It is not enough to recognize and classify isolated molecules in complicated biological structures, including cells, tissues, or even the entire human body, in order to comprehend them. We also need to thoroughly understand the interaction between molecules, minerals, and biological processes [4]. This is even truer for understanding complex diseases such as cancer, Alzheimer's disease, and others [5–8]. Researchers can now track intricate cellular operations at the molecular scale thanks to modern technology breakthroughs. Scientists can use statistical formulas to look at how highly regulated systems are linked and also how disturbances in such systems might lead to illness. One part of understanding biological objects is the processing of the data represented in the form of images from a microscope. An important subpart of data processing is data segmentation [1,2,9]. It involves image subdivision, pixel classification into objects and background and thus simplifies the representation of the real microscopy data.

Quantification of data using mathematical models, which speeds up data analysis, is very helpful and necessary in research. In this article, we analyze models and modeling processes specific to biology and medicine. We focus on the segmentation process of images acquired by microscopy using known methods. In addition, we present two new approaches to the segmentation of microscopic data. We describe these approaches by mathematical models that have been implemented as computer software, and we present their outputs with specific results. As a model, we decided to use animal tissues and look for the presence of iron in them. The occurrence and existence of iron in the brain are associated with Alzheimer's disease, and we present an alternative method for finding iron in such samples. In recent years, it has been found that iron levels in the brain fluctuate during Alzheimer's disease and can play an important role in the development of the disease [5,6,10,11]. These findings can be well applied in the early diagnosis of the disease and early initiation of treatment, which can stop or slow the progression of the disease. Therefore, it is important to find out how these levels change and how they are related to the disease. Even the study of symmetry and asymmetry design in our research data set is necessary for the next analyses and getting additional information to biologists.

Image segmentation is a method that turns data into useful information. During processing, an image is divided into pixels or segments, each of which is assigned a label according to its properties [12]. Digital filters are used to enhance digital images [2,13]. The following is a segmentation of the areas we want to analyze. There are many areas where image processing can be used, e.g., as traffic control at crossroads where each car and person will be detected as a separate object. Other current hot topics include autonomous vehicles [14,15], facial recognition, satellite imaging, video monitoring systems, medical imaging (CT, MRI), cancer cell detection, etc. [13,16–22]. In our work, we focused on

the medical field, specifically biological images of samples, histologically stained tissue. Nowadays, it is very attractive to use machine learning and neural networks for image processing analysis [13,16,23].

Our motivation for writing the article was to work with biologists, and we found out that there is no uniform mathematical approach to image processing in the field of biology. Quantitative analysis of histological samples is an important part of various studies, but everyone does the analysis differently. That is why, in our article, we compare different methodologies and at the same time contribute with optimizations that can improve the analysis in question and provide clearer results. The most accurate quantitative analysis can help steer research in the right direction.

This paper is organized as follows: In Section 1 the introduction provides a mini review of technologies in bioresearch. In Section 2, we present basic methods for image segmentation, detection, and finally, iron segmentation in biological samples. In Section 3, we describe our two new results and present some new and optimized algorithms for image processing of biological data. The use of Image J open-source software and the results of all three methods is discussed in Section 4. Section 5 is a discussion of obtained results and a comparison of used techniques. The conclusion is in Section 6.

## 2. Trivial Methods and an Overview of the Known Methods

The segmentation process aims to distinguish and comprehend what is in the image at the pixel level [24]. Every pixel in an image belongs to a single class, as opposed to object detection, where the bounding boxes of objects overlap. Image segmentation is often employed when high accuracy computer vision is required for a given application. Typical use of image segmentation includes autonomous vehicles, medical and retail applications. We focus mainly on image segmentation. Many image segmentation methods exist, and we present some of them.

### 2.1. Thresholding Methods

The most basic approach of segmentation is thresholding. Thresholding could be used to generate binary images from the grayscale [1,3,4,24]. Thresholding methods are categorized into six groups based on the information manipulated by the algorithm, and we focus on different clustering-based thresholding methods in this paper.

### 2.2. Level-Set Methods

Level-set methods (LSM) are a theoretical paradigm for analyzing areas and geometries numerically using level sets [3,24]. The level-set technique has the benefit of allowing quantitative computations comprising shapes to be performed on a predefined Cartesian matrix without the need to parameterize such components (called the Eulerian approach) [1,9,25–28].

### 2.3. Graph-Cut Methods

In a segmentation using the graph-cut method, each image is represented as a graph of nodes. Each node corresponds to an image pixel, and links that connect the nodes are called edges. A pathway is constructed connecting all the edges to travel across the graph [1–3,9,29–32].

### 2.4. Neural Network Methods

Image segmentation using a convolutional neural network involves the automatic identification of regions in an image and labeling them to different classes. Automatic image segmentation is currently one of the major research areas in computer vision.

The technique of segmenting data involves dividing it into discrete categories. A well-defined section has individuals who are comparable to one another while yet being distinct from those in neighboring parts.

We need to mention open-source software for image processing: Image J [33,34], MetaMorph (Molecular Devices, San Jose, CA, USA), Cell profiler [35], Illastik [36], and other methods for segmentation and data analysis [4,24,37–44].

### 3. Materials and Methods—Optimized Approaches

#### 3.1. Animal Brain Samples

We used animal models, specifically APP/PS1 mice, to compare mathematical approaches [8]. This model is very well researched and commonly used in Alzheimer's disease research [45–47]. The animals were kept in standard conditions with unrestricted access to food and water. The brains were taken at different ages of animals. After, they were washed with 4% formaldehyde solution to prevent the dye from binding to the iron contained in the blood. Brains were cut to a thickness of 35  $\mu\text{m}$  using a microtome (Leica SM 2000F, Wetzlar, Germany). Brain samples were sourced from an animal research laboratory.

#### 3.2. Histological Staining

Histological staining was performed according to Perl's blue iron staining protocol with DAB (3,3'-diaminobenzidine) visualization. As a result, the iron was colored dark brown (or gold in darkfield) [48].

#### 3.3. Microscopy

We used a light microscope (Zeiss, Gottingen, Germany) with an AxioCam MRc 5 camera (Zeiss, Gottingen, Germany) to photograph the samples. We focused on the hippocampal region, which is directly related to Alzheimer's disease, and thus changes in iron levels are visible here. An objective lens with 4 $\times$  magnification was used.

#### 3.4. Mathematical Approaches

##### 3.4.1. RGB Thresholding Algorithm—The First Method

We present a completely new optimization for data processing based on mathematical modeling. Our main objective is to detect iron in the images from the provided mouse hippocampus samples. The process to accomplish this comprises of a basic image pre-processing step to filter out the content to be analyzed, an image processing algorithm to detect iron in the samples, and a machine learning-based clustering technique to optimize that content, and a final script to record the pixel percentage.

- The RGB thresholding algorithm:

The RGB-based multispectral thresholding approach [49] was devised by us. Each pixel in a color image has three RGB values assigned to it. Every pixel's values are extracted, and strict thresholds are applied to each of the three streams individually, i.e., the red, blue, and green streams. Analyzing the learning batch of images and identifying the RGB intensities of all the pixels that are confirmed to be our targeted sample yields the hard thresholds. The optimum amounts of all three colors in that array are considered as boundaries. For experimentation objectives, a ten percent inaccuracy is permitted on both sides of the limits.

These various requirements are then merged with a logical AND operation, whereby all pixels that do not fulfill several of the three thresholds are assigned to the background, while the remaining pixels are assigned to the foreground.

Formulas (1)–(4) depict the theoretical counterpart of the procedure.

$$g1(x,y) = \begin{cases} 1, & Lr < r(x,y) < Hr, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

$$g2(x,y) = \begin{cases} 1, & Lg < g(x,y) < Hg, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

$$g3(x, y) = \begin{cases} 1, & Lb < b(x, y) < Hb, \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

$$h(x, y) = g1(x, y) \text{ AND } g2(x, y) \text{ AND } g3(x, y) \quad (4)$$

Here,  $Lr$ ,  $Lg$ , and  $Lb$  depict the target lower limits of the red, green, and blue channels, respectively, while  $Hr$ ,  $Hg$ , and  $Hb$  signify the upper limits. Then  $r(x, y)$ ,  $b(x, y)$ , and  $g(x, y)$  signify the red, blue, and green values on the RGB scale of the pixel with  $x$  and  $y$  as coordinates. In addition,  $h(x, y)$  gives the coordinates of the pixels that match the target RGB thresholding and are potentially containing iron in the brain samples.

The results were cross-validated by performing multiple levels of thresholding and finding the optimal values of  $L$ ,  $H$ , and  $S$ , which gives the best results. Equations (1)–(3) represent the values of  $L$ ,  $H$ , and  $S$ .

$$Li = \begin{cases} 200, & i = r, \\ 200, & i = g, \\ 0, & i = b. \end{cases} \quad (5)$$

$$Hi = \begin{cases} 256, & i = r, \\ 256, & i = g, \\ 256, & i = b. \end{cases} \quad (6)$$

- The OPTICS clustering algorithm (or the Ankerst-Breunig-Kriegel-Sander algorithm):

The problem with the current iron detected samples we have is that significant noise is present in the background apart of the samples. In addition, the noise is of varying density because of the varying density, due to the nature of the hippocampus, in different image samples. Hence, we use a density-based spatial clustering algorithm called the OPTICS algorithm (ordering points to identify the clustering structure) [50–52].

OPTICS, such as DBSCAN (density-based spatial clustering of applications with noise) [53], has two factors: that specifies the maximal range (radius) to examine, and MinPts, that specifies the minimum amount of points needed to create a group. If at least MinPts points are discovered inside its  $\epsilon$ -neighborhood  $N(p)$  or  $N(p)$ , the point  $p$  is considered a central point (including point  $p$  itself). Unlike DBSCAN, OPTICS considers points that really are members of a tightly populated group; hence, every point is given a base length that represents the length to the MinPts th nearest point:

$$\text{core} - \text{dist}_{\epsilon, \text{MinPts}}(p) = \begin{cases} \text{UNDEFINED} & \text{if } [N(p) < \text{MinPts}] \\ \text{MinPts} - \text{th smallest distance in } N_{\epsilon}(p) & \text{otherwise} \end{cases} \quad (7)$$

The reachability-distance of another point  $o$  from a point  $p$  is either the distance between  $o$  and  $p$  or the core distance of  $p$ , whichever is bigger:

$$\text{reachability} - \text{dist}_{\epsilon, \text{MinPts}}(p) = \begin{cases} \text{UNDEFINED} & \text{if } [N(p) < \text{MinPts}] \\ \max(\text{core} - \text{dist}(p), \text{dist}(p, o)) & \text{otherwise} \end{cases} \quad (8)$$

We implemented the algorithm in the programming language C and C++. Therefore, we present the core and basic steps of Algorithm 1 and updated Algorithm 2.

**Algorithm 1:** OPTICS (DB, eps, MinPts)**Input:** DB, eps= maximum radius, MinPts- number of points required to form a clusterOutput:  $p$  to the ordered list**Step1:** for each point  $p$  of DB do $p$ .reachability-distance = UNDEFINED**for each** unprocessed point  $p$  of DB **do** $N$  = getNeighbors( $p$ , eps)mark  $p$  as processed**Step2:** if core-distance( $p$ , eps, MinPts) != UNDEFINED **then**

Seeds = empty priority queue

update( $N$ ,  $p$ , Seeds, eps, MinPts)**Step3:** **for each** next  $q$  in Seeds **do** $N'$  = getNeighbors( $q$ , eps)mark  $q$  as processedoutput  $q$  to the ordered list**Step4:** if core-distance( $q$ , eps, MinPts) != UNDEFINED **do**update( $N'$ ,  $q$ , Seeds, eps, MinPts)

In update(), the priority queue seeds is updated with the  $\varepsilon\varepsilon' < \varepsilon$   $\varepsilon$ -neighborhood of  $p$  and  $q$ , respectively.

**Algorithm 2:** update**Input:**  $N$ —neighbor,  $p$ —core points, Seeds-priority queue, eps, MinPts

Output: updated seeds

**Step1:** coredist = core-distance ( $p$ , eps, MinPts)**Step2:** **for each**  $o$  in  $N$ **if**  $o$  is not processed **then**new-reach-dist = max(coredist, dist( $p$ ,  $o$ ))**Step3:** if  $o$ .reachability-distance == UNDEFINED **then** //  $o$  is not in Seeds $o$ .reachability-distance = new-reach-distSeeds.insert( $o$ , new-reach-dist)**Step4:** **else** //  $o$  in Seeds, check for improvement**if** new-reach-dist <  $o$ .reachability-distance **then** $o$ .reachability-distance = new-reach-distSeeds.move-up ( $o$ , new-reach-dist)

For our given result, the *MinPoints* = 150, and  $\varepsilon$  = 67 pixels. We use this to find out the total number of pixels that are noise and do not belong to any of the clusters.

- Extracting the pixel percentage:

This is performed through simple algebraic manipulations. We extract the number of dots detected from each segment after the filtration techniques. We find the percentage by dividing the total pixel area.

Now we explain shortly what we get from the algorithms and how we evaluate and count pixel percentage of iron for given picture samples: Pixel percentage of iron for given picture sample = (# of iron pixels detected after RGB thresholding algorithm—# of pixels detected as noise from OPTICS algorithm)/(Total # of pixels in the image), where # is a numerical value.

As an result we can observe one important fact: with this proposed method, we obtain the percentage evaluation of the iron present in the samples.

More, we have compared our method with existing methods such as Otsu's binarization and Kullback–Leibler adaptive thresholding techniques Otsu [54,55].

**Otsu Binarization**

The technique developed by Otsu [54,56] transforms a grayscale level matrix (image) into a binary/logical matrix. This technique can be considered as the fundamental

technique for adapting a regional threshold value of an image. This approach aims for a threshold value that can reduce intra-class variability. The weighted total of the variances of the two classes is the intra-class variance. The intra-class variance is calculated using the equation below

$$\alpha_{v\tau}^2 = \beta_{1(\tau)} * \alpha_{1\tau}^2 + \beta_{2(\tau)} * \alpha_{2\tau}^2 \quad (9)$$

Here,  $\alpha_1$  and  $\alpha_2$  are the variances, and  $v$  represent weights, which can be expressed as the probabilities of the classes using the threshold value  $\tau$ . The preceding calculation explains how minimizing and maximizing intra-class variation are essentially identical in Otsu's approach.

$$\alpha_{W\tau}^2 = \alpha^2 - \alpha_{v\tau}^2 = \beta_{1(\tau)} * \beta_{2(\tau)} * [\mu_{1(\tau)} - \mu_{2(\tau)}]^2 \quad (10)$$

Here,  $\mu$  refers to class mean corresponding to the probability  $P$  to  $y$  values for  $i$ -th bin of the center histogram, which can be expressed as

$$\mu_i(\tau) = \frac{\sum_0^{\tau} P(i) * y(i)}{\beta_i} \quad (11)$$

- Kullback–Leibler adaptive thresholding:

The comparative entropy had to be reduced at the gray level for this strategy to work. The Kullback–Leibler adaptive thresholding approach [55] is unusual because it emphasizes the entropy of a single image's co-occurrence vector. The relative entropy between two probability distributions  $d_1$  and  $d_2$  can be expressed as

$$D(P_1||P_2) = \int P_1(y) \log \frac{P_1(y)}{P_2(y)} dx \quad (12)$$

It distinguishes statistical procedures by expressing how distinct the probability distribution of two classes,  $P_1$  is from  $P_2$ , when the real data abide via maximum-likelihood assumption. For identical probability distribution, Kullback–Leibler adaptive thresholding approach returns a null value.

### 3.4.2. New Filtration Method—Second Method

In this approach, we present and suggest a new method for processing biological data and a new filter for processing image data and segmentation. Here we describe a filter that was used and present our approach to create a special filter. The first step in data analysis is preparing the data to be usable for segmentation. We will transform the image containing color data (three RGB values) to an image that contains a concentration of the selected tissue (one value). One of the most important parts of image processing is using histogram and filtration.

- Filtration based on histogram (black-white images):

The image histogram represents the intensity distribution in the digital image. Mathematically, it can be described as a function that assigns to every value of  $q$  from the set  $\{0, \dots, Q - 1\}$ , where  $Q$  is the maximum possible intensity (in our case 255), the number of pixels having intensity  $q$ . The histogram provides important insights into the distribution of image intensity. It can determine whether the image is underexposed, overexposed, whether it uses a full range of intensities, and the like.

The abbreviation “FSHS” stands for “full-scale histogram stretches”, and this operation is used to achieve the maximum pixel intensity range in black-and-white images. Adjusting the intensity of the original pixel (denoted by  $p(x)$ ) to the new intensity.

$$p_{corr}(x) = \{Q - 1\} \setminus \{B - A\} \cdot (p(x) - A) \quad (13)$$

The  $Q$  is the maximum possible intensity (in our case 255),  $A$  respectively  $B$  denotes minimum and maximum pixel intensities respectively of the original image. By this process, the image contrast for poorly distributed gray values dramatically improves and enables greater precise contour recognition.

Filtration is, in general, the separation of useful information from not useful (if we consider background or noise). By handling digital images, we consider reducing the noise or the background (non-useful information) and marking selected objects (useful information).

- Filtration based on RGB model (colored images):

Therefore, we define the RGB model as an additive-colored model. They belong to the basic colors: red, green, blue. Linear combination of these colors arises the whole scale, e.g.,  $(0, 0, 0)$  is black,  $(1, 1, 1)$  is white. The RGB model is used in image processing. In our case, the RGB information is the one in which we want to catch exactly a certain kind of color connected with the concentration of iron in the selected tissue.

By image processing of color images, filters are used on this basic RGB model with the aim of color processing. In our case, the needed information is the information that we want to catch, certain kinds of colors connected with the concentration of the selected item. By image processing, we use filtering on the RGB base to modify colors. In general, we can define this kind of filtration as follows: each pixel can be defined as a function of the color of the original pixel,  $f_R, f_G, f_B$ , from which we can obtain a new and modified coloring:

$$q_{R(x)} = f_R(p_{R(x)}, p_{G(x)}, p_{B(x)}) \quad (14)$$

$$q_{G(x)} = f_G(p_{R(x)}, p_{G(x)}, p_{B(x)}) \quad (15)$$

$$q_{B(x)} = f_B(p_{R(x)}, p_{G(x)}, p_{B(x)}) \quad (16)$$

where  $p_R, p_G$  and  $q_R, q_G$  are the only values of the original or the newly transformed intensities in the  $RGB$  channels. In the easiest case, we consider a linear combination of the original color values, which we present by the matrices. The color of the pixel will be presented as a vector in three folders:

$$q^{*(x)} = Mp^{*(x)}, \quad (17)$$

where  $p^{*(x)}$  and  $q^{*(x)}$  are the column vectors of the origin and the new color of the pixel,  $M$  is the matrix of the size  $3 \times 3$ , where the coefficients are the elements of the weighted filter. These coefficients show the measure of the new coefficient of colors versus the original coefficients of colors ( $R, G, B$ ).

For our purposes, we use a similar function, which transforms the value of three-channel RGB into the new value intensity of pixel:

$$q(x) = w_R \cdot p_{R(x)} + w_G \cdot p_{G(x)} + w_B \cdot p_{B(x)}. \quad (18)$$

Coefficients  $w_R, w_G, w_B$  are the values between 0 and 1, and they are given by the settings of the filter. These coefficients decide which colored channel will be marked by or suppressed by the filter.

Application of the mentioned approach, the colored picture will be reduced into a black-and-white image. It means we first modify and transform the colored images into black-and-white images, and consequently, we are doing analyses and image processing with black-and-white images. This way, we reduce the amount of data to one-third. The goal of the filter is to assign each pixel a value between 0 and 1 (respectively, 0 and 255), depending on the RGB input values. In doing so, we require that the value of the output pixel 1, if its color perfectly matches the color of the object to be searched and drops reasonably to 0 in the case of another coloring.

The filter sensitivity to the color ratio is ensured by assigning weights (marked as  $w_R, w_G, w_B$ ) to the components  $R, G, B$  of the access pixel. The filter can be written as a function of the “hp”, color,  $R, G, B$  color of the input image labeled.

$$(p_R, p_G, p_B) : q(x) = w_R \cdot p_R(x) + w_G \cdot p_G(x) + w_B \cdot p_B(x). \quad (19)$$

when processing diverse color data, a different color than the desired color can be affected by the filtering result. To suppress the effect of pixels other than of the tuned color, we introduce a penalty function to divide the filtering result, when

$$q(x) = \{w_R \cdot p_R(x) + w_G \cdot p_G(x) + w_B \cdot p_B(x)\} \setminus \{d_R \cdot p_R(x) + d_G \cdot p_G(x) + d_B \cdot p_B(x)\} \quad (20)$$

Penalization values  $d_R, d_G, d_B$  are values between 0 and 1 and determine the rate of suppression of a given color channel. When using this filter alone, the coefficients  $d_R, d_G, d_B$  must be set to  $d_R + d_G + d_B = 1$ .

In our implementation, this is handled by a specially modified FSHS method that scales any calculated values into the interval  $[0, 255]$ .

Now we show the filter setting for these samples. In this case of detecting iron in samples, one set of parameters is three images. We compare two procedures for the processing of these images that differ in the order of operations applied:

The average filter is a variant in which we first average the RGB values of three original frames, pixel by pixel. From these averages, we will compile one and then apply a filter to the image. The filter average, on the other hand, consists of filtering on each from three images, then averaging three filtered images.

The filter average consists of filtering on each picture from three given shots. Consequently, we average three filtered images. This variant is much more computationally demanding.

### 3.4.3. Third Method—Using the Image J Program

Image J is a very popular and open-source software for analysis of biological data [33]. In our quantification of the amount of iron, we opened microscope images in this program and adjusted the color threshold so that only the dark brown iron remained marked. This was followed by particle analysis, which calculated the amount of iron in the images. After obtaining quantitative data, we were able to compare the amount of iron in different samples.

User-defined choices and intensity-thresholded entities can have their region and image pixel metrics calculated by Image J. This will calculate distances and slopes. It has the ability to generate densities histograms, and length depends on charts.

### 3.4.4. Fourth Method—The Structural Similarity Index Method (SSIM)

The structural similarity index method (SSIM) is a technique that is centered on perception. Image deterioration is defined in this approach as a modification in the geometric data processing. It also works with various crucial perception-related aspects such as brightness mask, contrasting mask, and so on. In the image field, these highly interdependent pixels relate to a certain additional crucial data regarding the graphic elements. SSIM estimates the perceived quality of the image. It measures the similarity between two images.

$$S(i, j) = F_1(i, j)^{c1} * F_2(i, j)^{c2} * F_3(i, j)^{c3} \quad (21)$$

where  $F_1, F_2, F_3$  and  $F$  signifies contrast, luminance, and structural correlation. The multiplication of these quality measurement metrics represents the SSIM index. Indexes  $c1, c2$ , and  $c3$  are the constants.

$$F_1(ij) = (2 * \alpha_i \alpha_j + a) / (\alpha_i^2 + \alpha_j^2 + a) \quad (22)$$

$$F_2(ij) = (2 * \beta_i \beta_j + b) / (\beta_i^2 + \beta_j^2 + b) \quad (23)$$

$$F_3(ij) = (\beta_{ij} + c) / (\beta_i \beta_j + c) \quad (24)$$

Here,  $\alpha_i$ ,  $\alpha_j$  and  $\beta_i$ ,  $\beta_j$  are the mean and standard deviations, respectively, and  $\beta_{ij}$  is the cross-covariance for  $i$ ,  $j$  images. The similarity index ( $S$ ) can be written as

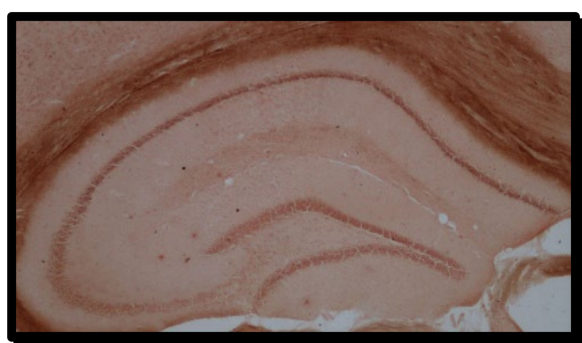
$$S(i,j) = ((2 * \alpha_i \alpha_j + a) (2 * \beta_i \beta_j + b)) / (\alpha_i^2 + \alpha_j^2 + a)(\beta_i^2 + \beta_j^2 + b) \quad (25)$$

#### 4. Results

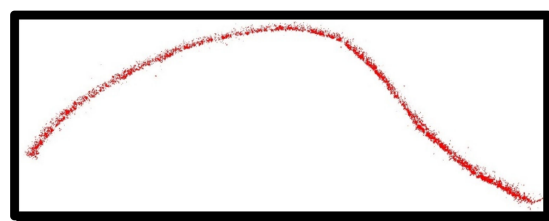
In this section, we present particular image results and applications of the mentioned methods. We apply the results to the biological samples described above. Results are segmented images and a percentual evaluation of the iron in the proceeded samples.

##### 4.1. RGB Thresholding Algorithm—The First Method

In this section, we present results obtained by application of the first method-RGB threshold method. We obtained segmented data—the dots of iron in microscopy images of animal brain hippocampus samples, see Figures 1–5. We see original data and resultant data—segmented iron (Figures 1–3).

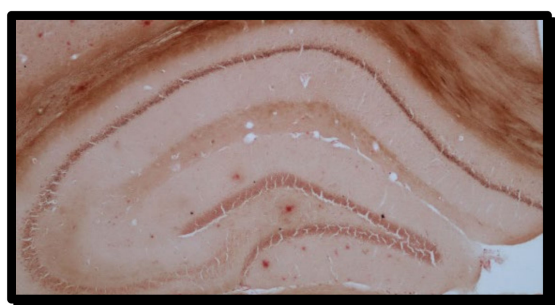


(a)

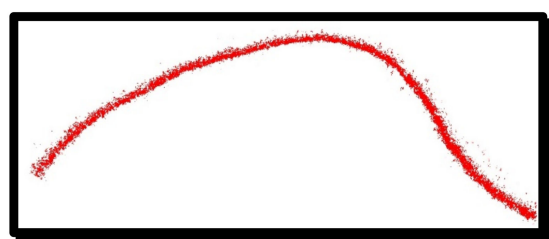


(b)

**Figure 1.** (a) hippocampus of the 8-month-old animal; (b) selected iron from the hippocampus.

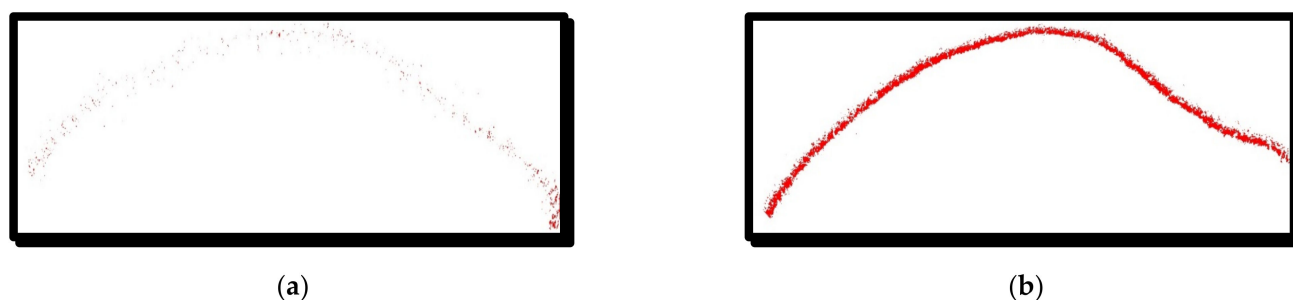


(a)

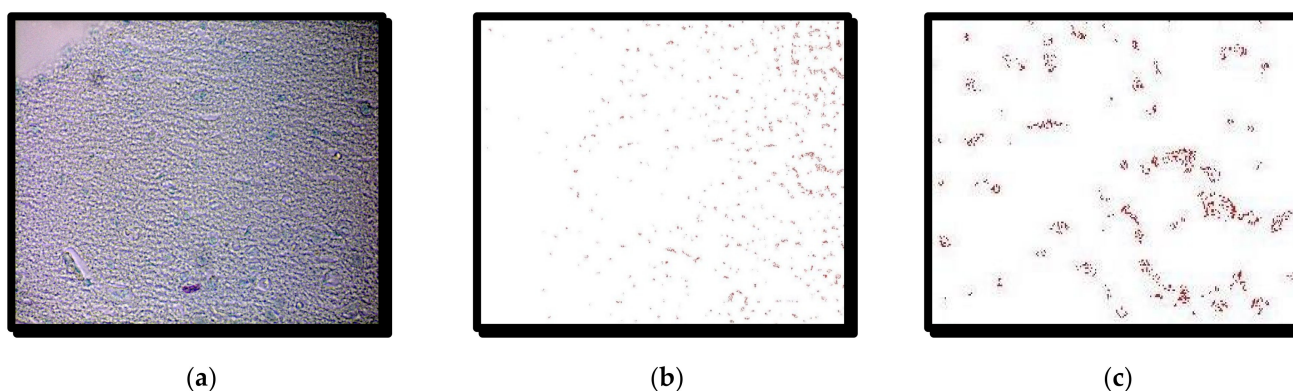


(b)

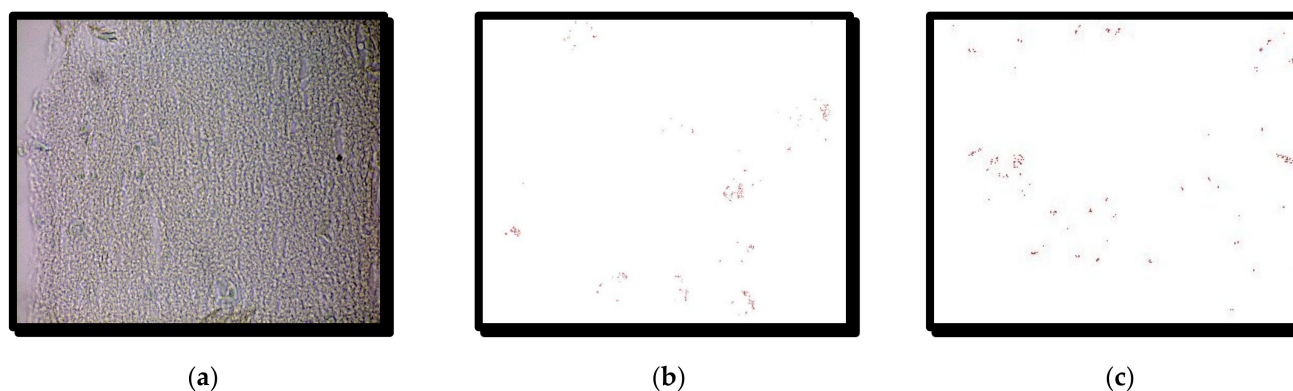
**Figure 2.** (a) left side—hippocampus of the 13-month-old animal; (b) right side—selected iron from the hippocampus.



**Figure 3.** Results of 6-month-old animal (a) original image with filtration subsets in the processing of selection of iron dots, (b) segmented image.



**Figure 4.** (a) 10 months old animal—brain cortex (b); filtered image (c); zoom of a region in a filtered image (b).



**Figure 5.** (a) Second 10-month-old animal—brain cortex (b); filtered image (c); zoom of a region in a filtered image (b).

In Table 1, we obtain a numerical evaluation of discovered iron in our images. We processed data with our own software, where we implemented the required analysis. We note that the calculation and numerical evaluation of the occurrence of iron in the samples were implemented directly in our software. Open-source software does not provide us with this type of special analysis (proof of the existence of iron, but also numerical and percentage evaluation of its occurrence). Therefore, we had to develop special “tailor-made” software and implement the special requirements that biologists need for the analyzes. We extract the image size (length and width of images in pixels) and calculate the whole area of an image and compare it to the amount of iron in that image.

**Table 1.** Percentual expression of the results.

Animal	Length of Image (Pixels)	Width of Image (Pixels)	Total Area (Pixels)	Total Iron (Pixels)	Total Iron (%)
6 months	5102	1961	10,005,022	394,419	0.039
8 months	4776	2281	10,894,056	529,147	0.049
13 months	5352	2497	13,363,944	701,661	0.053

In this section, we present concrete image results and applications of the previously mentioned methods. We apply the results of software for biological data processing. More, we have compared our method with existing methods such as Otsu's binarization (Table 2) and Kullback–Leibler (Table 3) adaptive thresholding techniques. This is performed through simple algebraic manipulations. We extract the number of dots detected from each segment after the filtration techniques and find the percentage by dividing the total pixel area.

**Table 2.** Percentual expression of the results of Otsu's thresholding.

Animal	Length of Image (Pixels)	Width of Image (Pixels)	Total Area (Pixels)	Total Iron (Pixels)	Total Iron (%)
6 months	6889	4495	30,966,055	673,311	0.021
8 months	6786	4271	28,983,006	727,235	0.025
13 months	7305	3431	25,063,455	906,655	0.036

**Table 3.** Percentual expression of the results of adaptive thresholding.

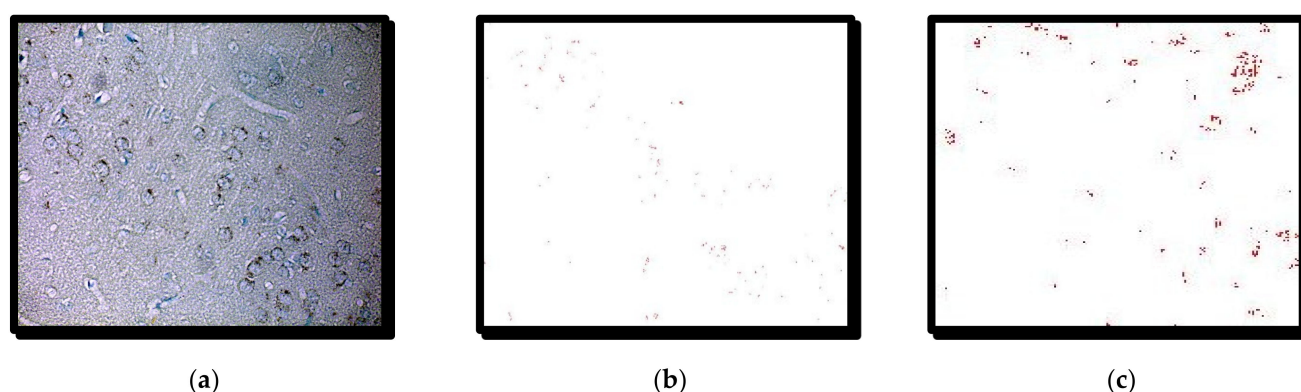
Animal	Length of Image (Pixels)	Width of Image (Pixels)	Total Area (Pixels)	Total Iron (Pixels)	Total Iron (%)
6 months	5880	4593	27,006,840	698,866	0.025
8 months	6001	3281	19,689,281	579,981	0.029
13 months	4800	2597	12,465,600	517,863	0.041

Pixel percentage of iron for given picture sample = (# of iron pixels detected after RGB thresholding algorithm—# of pixels detected as noise from OPTICS algorithm)/(Total # of pixels in the image), where # is a numerical value. With this method, we obtain the percentage evaluation of the iron present in the samples.

Comparing the results as presented in Tables 1–3, we observe that total iron (%) for a 6-month-old animal varies from 0.039% to 0.021% and to 0.025%, while for a 9-month-old animal, it varies from 0.049% to 0.025% and to 0.029% and for a 13-month-old animal it varies from 0.053% to 0.036% and to 0.041% for our own algorithm, Otsu's thresholding and for adaptive thresholding, respectively. Our own algorithm gives roughly a one-fold increase in total iron (%) in comparison to other methods for 6- and 8-month-old animals.

#### 4.2. New Filtration Method—Second Method

We also tried to find different levels of iron in the cortex part of the brain of the mice (Figures 4–6). Dark dots represent segmented iron in images. The goal was to process the original data and obtain the best segmentation of iron. In addition, a numerical evaluation of the segmented iron as a fraction of the whole area was performed. In this part, we offer processing of the brain cortex of animals and its segmentation. We even offer numerical and percentage evaluation of data, segmentation results, which will be used for further biological processing and analysis.



**Figure 6.** (a) Third 10-month-old animal—brain cortex (b); filtered image (c); zoom of a region in a filtered image (b).

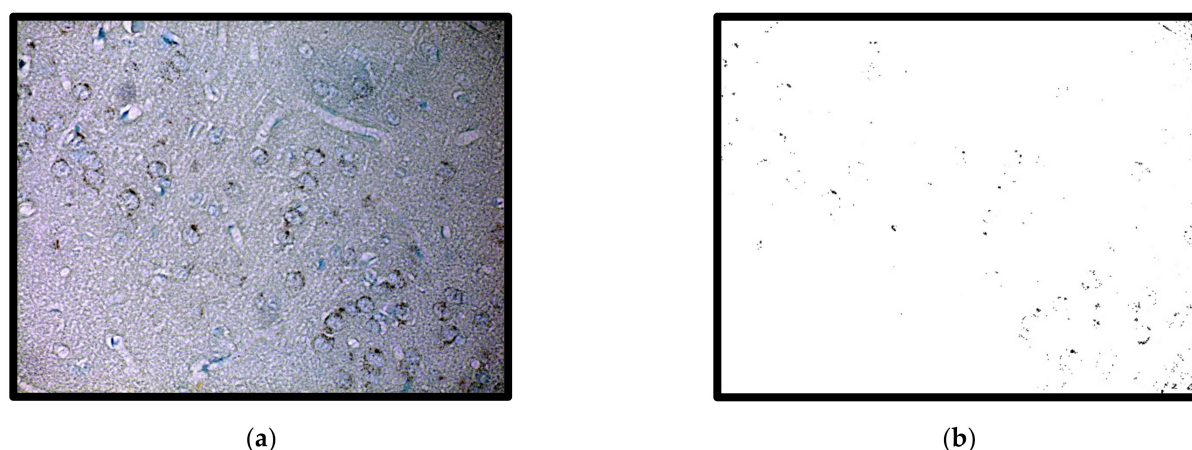
Table 4 shows the percentual amount of iron in the cortex area. The most important contribution of this article is our special tailor-made software, which is able to answer the special questions of biologists, for example: the percentage of pixels of existing iron in the data. Therefore, we have developed special custom software that can answer and quantify the level of histological stained samples (in terms of pixels, percentages, etc.), so it gives us a quantitative analysis.

**Table 4.** Amount of iron in the cortex area.

Animal	Detected Area (Pixel)	False Positive Area	Net Detected Area	Total Image Area (Pixel)	%
10 months old animal	14,983	0	14,983	4,958,816	0.3
2nd animal	1653	100	1553	4,958,816	0.11
3rd animal	6262	170	6092	4,958,816	0.12

#### 4.3. Third Method—Using the Image J Program

In this part, we present results obtained by the open-source software Image J (Image 7). Iron was segmented out manually from the images of animal brain samples as in Section 4.2, see Figure 7. In Table 5 are the results from Image J software, also included example in Figure 7. The results are similar in the case of the first animal. However, in the other two samples, where the level of iron was lower, we can see bigger differences. This can be due to noise, which was filtrated better in the previous case.



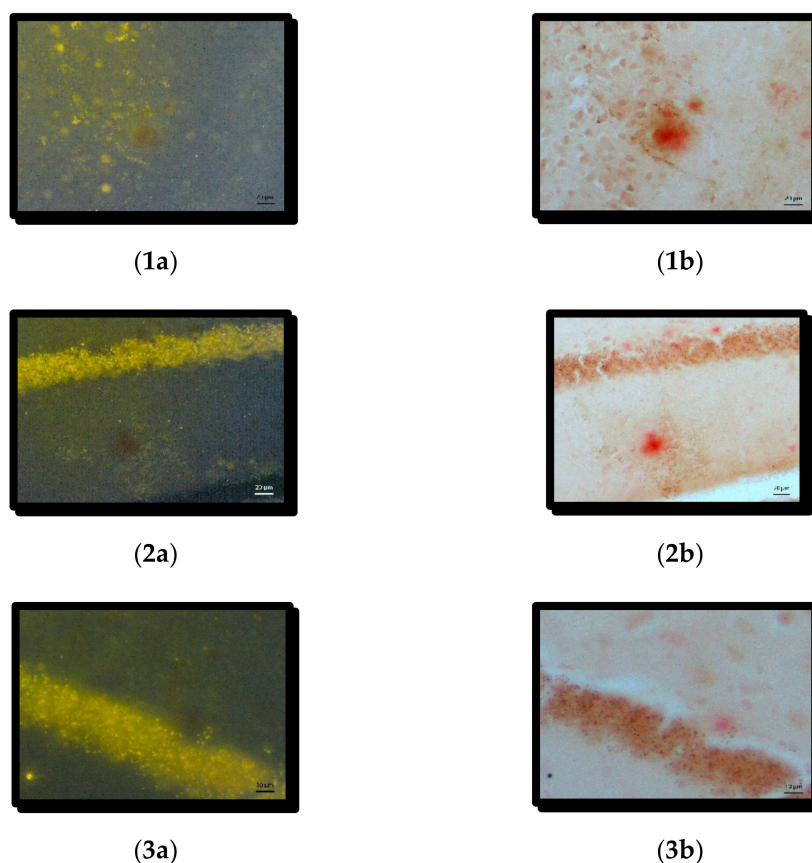
**Figure 7.** Third 10-month-old animal from Figure 6—(a) brain cortex; (b) segmented iron in Image J program.

**Table 5.** Amount of iron in the cortex area by using Image J software.

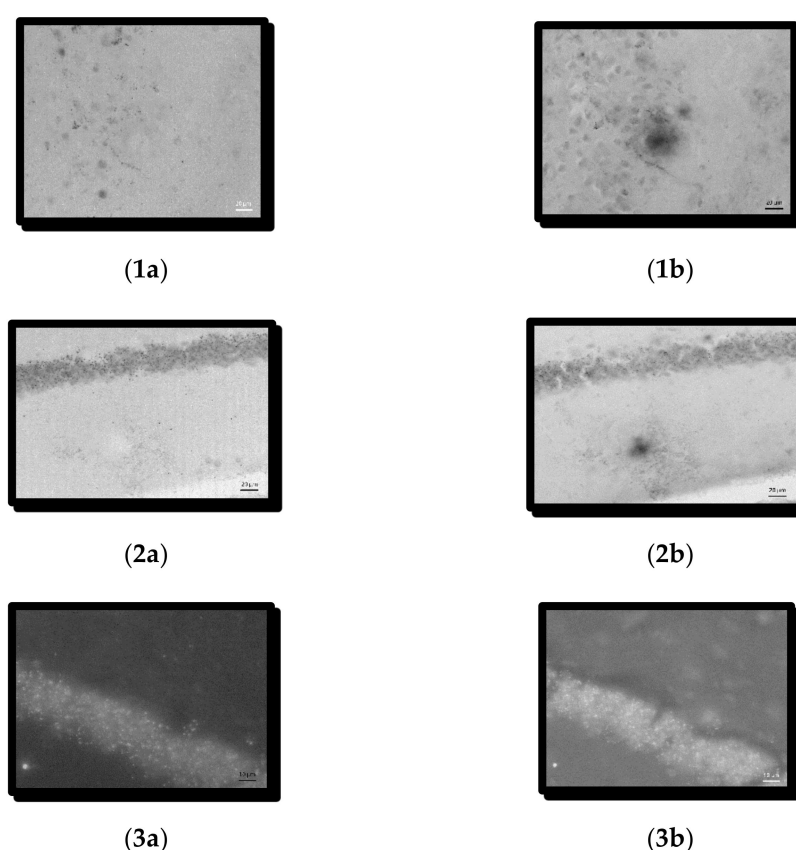
Animal	Total Area	%
10-month-old animal	15,186	0.3
2nd animal	11,786	0.24
3rd animal	13,105	0.26

#### 4.4. Symmetry Analysis of Images Photographed in the Bright and Dark Fields of the Light Microscope

In this part, we used four pairs of images, which are photographed in the bright and dark fields of the microscope, see Figure 8. Iron is brown (not to be confused with red) in the brightfield, golden in the dark field. We wanted to compare the same images and find out the accuracy of the algorithms and whether the result is affected by the selected imaging mode.

**Figure 8.** The same photos from darkfield (1a–3a) and brightfield (1b–3b) microscopy.

From the images of Figure 8 the SSIM values are calculated. The grayscale images obtained after considering the intensity value as (I-130) on the images from Figure 9. The SSIM values are calculated for the images in Figure 9. The results are shown in Table 6. Since the SSIM values are greater than 0.5, we can conclude that these images are similar.



**Figure 9.** The SSIM values are calculated after converting the images of Figure 8 into grayscale images. The grayscale images obtained after considering the intensity value as (I-130) on the images from Figure 8. The same photos from darkfield (1a–3a) and brightfield (1b–3b) microscopy. Figures (1a–3a) represent the dark images after decreasing the individual intensity by 130. Figures (1b–3b) represent the light–bright grayscale images after decreasing the individual intensity by 130, respectively.

**Table 6.** The SSIM values calculated for the images in Figure 9.

Image Set	SSIM
1a–1b	0.6512
2a–2b	0.7708
3a–3b	0.7436

## 5. Discussion

In this article, we present a summary of methods for processing biological data as well as their direct applications. Our used methods aimed to eliminate noise by filter processes. The main benefits of this work are as follows: Using a modified RGB algorithm, a unique threshold algorithm, and a new filtering method to obtain more precise results. The presented methods have a wide application. They may not be useful for the quantification of all biological data, but they can be used mainly for the quantitative analysis of photographs by histological or immunohistochemical staining taken under a microscope. It does not depend on the type of histological or immunohistochemical staining. It does not depend on tissue and organ. Thanks to color thresholding, we can use our optimizations for all these samples.

Research aimed at finding histologically stained protein, even its numerical evaluation, has been appearing in biological research for a long time. We tried to bring new ways of technological perspectives on the detection of stained samples and facilitate the identification and representation of reading and interpretation of data. There is great

interest in providing an objective tool for evaluating the amount of iron in microscope images by using mathematical modeling and technology. We try to transfer the biological problem to mathematical models and speed up and refine the analysis. Then we can solve questions in the language of mathematics. We used two innovations and one open-source software method to determine the amount of iron, which can process data in suitable quality and provide a responsible answer and numerical evaluation. Two methods are based on “image processing methods” and deal with data filtering and segmentation. In this way, we obtained reasonable and objective results for further analysis of biological data. We obtained an objective segmentation of iron and even an objective numerical evaluation of the appearance of iron in the samples. A comparison of the software shows us that our own modified algorithm brings us more detailed results as well as numerical evaluation. Therefore, it is also important to consider the use of mathematical models in biological research.

A comparison of common procedures is also included. Authors Boykov and Funka Lea [31] conducted an in-depth evaluation of graph cuts and level-set approaches. They conducted a thorough investigation using both simulated and actual data. They spent a long time researching this issue. We also tried to apply the newest other techniques to this data. However, there was a problem that other techniques such as level-set methods, region growing methods, and others, even Image J, were not able to deal with the noise in the data up to our satisfaction. This is the current problem with biological data. The method, which can process the data, is always dependent on the specific data, on their quality, on the noise in images, and on the output from the microscope. Biologists often do not have the mathematical and numerical tools to verify their assumptions. So, we also tried to show the fact that images acquired in the dark and bright fields of the microscope provide us the same results.

At last, we performed a similarity check of images taken in different situations using the structural similarity index (SSIM) method. We compared the segmentation process using Otsu and Kullback–Leibler adaptive thresholding. It was very useful to create your own software that can deal with the special requirements of biologists and their special needs. Even in our case, we needed to try out more methods and develop our own software and optimize it for obtaining optimal results that are appropriate for biologists. We also compare the techniques that we used to approach and process this biological data. We conclude that our own software solution, written in C/C++ programming language, was able to process the data in such a way that segmentation of suitable quality was achieved and thus resulting in a proper numerical evaluation of the segmented elements.

Copulas are special classes of aggregation functions. Copulas are used to describe or model the dependence (or inter-correlation) between random variables. There are many parametric copula families available, which usually have parameters that control the strength of dependence. Copulas have many applications in the area of medicine, e.g., in the segmentation of magnetic resonance images, in a study on schizophrenia, or studies on Alzheimer’s patients. Copulas were also used in brain research based on EEG signals to detect drowsiness during a daytime nap, or, for example, to characterize dependence in oscillatory activity between EEG channels, to capture dependence between pairs of EEG channels using their time-varying envelopes and other neuronal dependencies (spike counts and other). Finally, although we have considered the linearity relationship between images  $i$  and  $j$  and used their cross-covariance in Equation (24). Another interesting situation that we are going to consider occurs when the assumption is beyond linearity, and then some nonlinear measures such as a copula or mutual information can be used in our implementation [22,57].

## 6. Conclusions

Technology continues to yield ever new methods of solutions and answers to many biological questions. The methods mentioned in this paper can help in the detection of iron presence and its quantities particularly, but such methods can also be used on any light

microscopy images of stained samples. New methods and trends can also be opened for data segmentation and analysis. The real problem may arise from the fact that the data acquired by different procedures are not always clear; they are affected by different types of noise. However, various techniques can help in noise-filtering and bring new ways to advance imaging data, thus helping biological and medical research.

**Author Contributions:** Conceptualization: M.Ž., H.K., and M.K.; software, A.C.; M.G., and S.M.O.; validation, M.Ž. and H.K.; methodology, M.Ž., A.C., H.K., M.G., and S.M.O.; formal analyses, investigation, M.Ž., M.K., H.K., M.G., and S.M.O.; resources, M.Ž. and H.K.; writing—original draft preparation, M.Ž., H.K., A.C., M.K., M.G., and S.M.O.; writing—review and editing, M.Ž., H.K., M.K., D.K., M.G., and S.M.O.; project administration, M.Ž. All authors have read and agreed to the published version of the manuscript.

**Funding:** Mária Ždímalová acknowledges the Scientific Slovak Grant Vega 1/006/19. This paper was also supported by the Scientific Slovak Grant 1/0173/20.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Basavaprasad, B.; Hegadi, R. A Survey on Traditional and Graph Theoretical Techniques for Image Segmentation. *Int. J. Comput. Appl.* **2014**, *975*, 8887.
2. Boykov, Y.Y.; Jolly, M.-P. Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images. In Proceedings of the International Conference on Computer Vision, Vancouver, BC, Canada, 7–14 July 2001; pp. 105–112.
3. Boykov, Y.; Veksler, O. Graph Cuts in Vision and Graphics: Theories and Applications BT—Handbook of Mathematical Models in Computer Vision. *Handb. Math. Model. Comput. Vis.* **2006**, *14*, 79–96.
4. Callara, A.L.; Magliaro, C.; Ahluwalia, A.; Vanello, N. A Smart Region-Growing Algorithm for Single-Neuron Segmentation from Confocal and 2-Photon Datasets. *Front. Neuroinform.* **2020**, *14*, 9. [\[CrossRef\]](#)
5. Casadesus, G.; Smith, M.A.; Zhu, X.; Aliev, G.; Cash, A.D.; Honda, K.; Petersen, R.B.; Perry, G. Alzheimer disease: Evidence for a central pathogenic role of iron-mediated reactive oxygen species. *J. Alzheimers. Dis.* **2004**, *6*, 165–169. [\[CrossRef\]](#) [\[PubMed\]](#)
6. Gong, N.; Dibb, R.; Bulk, M.; van der Weerd, L.; Liu, C. Imaging beta amyloid aggregation and iron accumulation in Alzheimer's disease using quantitative susceptibility mapping MRI. *Neuroimage* **2019**, *191*, 176–185. [\[CrossRef\]](#)
7. Caselli, R.J.; Beach, T.G.; Knopman, D.S.; Graff-Radford, N.R. Alzheimer Disease: Scientific Breakthroughs and Translational Challenges. *Mayo Clin. Proc.* **2017**, *92*, 978–994. [\[CrossRef\]](#)
8. Jankowsky, J.L.; Fadale, D.J.; Anderson, J.; Xu, G.M.; Gonzales, V.; Jenkins, N.A.; Copeland, N.G.; Lee, M.K.; Younkin, L.H.; Wagner, S.L.; et al. Mutant presenilins specifically elevate the levels of the 42 residue  $\beta$ -amyloid peptide in vivo: Evidence for augmentation of a 42-specific  $\gamma$  secretase. *Hum. Mol. Genet.* **2004**, *13*, 159–170. [\[CrossRef\]](#)
9. Peng, B.; Zhang, L.; Zhang, D. A Survey of Graph Theoretical Approaches to Image Segmentation. *Pattern Recognit.* **2012**, *46*, 1020–1038. [\[CrossRef\]](#)
10. Singh, N.; Haldar, S.; Tripathi, A.K.; Horback, K.; Wong, J.; Sharma, D.; Beserra, A.; Suda, S.; Anbalagan, C.; Dev, S.; et al. Brain Iron Homeostasis: From Molecular Mechanisms To Clinical Significance and Therapeutic Opportunities. *Antioxid. Redox Signal.* **2014**, *20*, 1324–1363. [\[CrossRef\]](#) [\[PubMed\]](#)
11. Andersen, H.H.; Johnsen, K.B.; Moos, T. Iron deposits in the chronically inflamed central nervous system and contributes to neurodegeneration. *Cell. Mol. Life Sci.* **2014**, *71*, 1607–1622. [\[CrossRef\]](#)
12. Yang, W.; Cai, L.; Wu, F. Image segmentation based on gray level and local relative entropy two dimensional histogram. *PLoS ONE* **2020**, *15*, e0229651. [\[CrossRef\]](#)
13. Bagga, P.J.; Makhesana, M.A.; Patel, K.; Patel, K.M. Tool wear monitoring in turning using image processing techniques. *Mater. Today Proc.* **2021**, *44*, 771–775. [\[CrossRef\]](#)
14. Zontone, P.; Affanni, A.; Bernardini, R.; Del Linz, L.; Piras, A.; Rinaldo, R. Stress evaluation in simulated autonomous and manual driving through the analysis of skin potential response and electrocardiogram signals. *Sensors* **2020**, *20*, 2494. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Sim, R.; Dias, K.; Martins, R. Dynamic Allocation of SDN Controllers in NFV-based MEC for the Internet of Vehicles. *Futur. Internet* **2021**, *13*, 270.
16. Singh, S.P.; Wang, L.; Gupta, S.; Goli, H.; Padmanabhan, P.; Gulyás, B. 3D Deep Learning on Medical Images: A Review. *Sensors* **2020**, *20*, 5097. [\[CrossRef\]](#) [\[PubMed\]](#)

17. Loucký, J.; Oberhuber, T. Graph cuts in segmentation of a left ventricle from MRI data. In Proceedings of the Czech-Japanese Seminar in Applied Mathematics, Prague and Telč, Czech Republic, 30 August–4 September 2010; pp. 46–54.
18. Manwar, R.; Zafar, M.; Xu, Q. Signal and Image Processing in Biomedical Photoacoustic Imaging: A Review. *Optics* **2020**, *2*, 1. [CrossRef]
19. Mathur, S.; Shantanu, A.; Rana, A. Estimating the imaging in medical science using image processing techniques. *J. Phys. Conf. Ser.* **2021**, *1714*, 012007. [CrossRef]
20. Zhang, J.; Pang, H.; Cai, W.; Yan, Z. Using image processing technology to create a novel fry counting algorithm. *Aquac. Fish* **2021**. Available online: <https://www.sciencedirect.com/science/article/pii/S2468550X20301659> (accessed on 23 November 2021).
21. Sheikhi, A.; Mehrali, Y.; Tata, M. On the exact joint distribution of a linear combination of order statistics and their concomitants in an exchangeable multivariate normal distribution. *Stat. Pap.* **2013**, *54*, 325–332. [CrossRef]
22. Mesiar, R.; Sheikhi, A. Nonlinear random forest classification, a copula-based approach. *Appl. Sci.* **2021**, *11*, 7140. [CrossRef]
23. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. Available online: [https://openaccess.thecvf.com/content\\_cvpr\\_2016/html/He\\_Deep\\_Residual\\_Learning\\_CVPR\\_2016\\_paper.html](https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html) (accessed on 23 November 2021).
24. Stockman, G.; Shapiro, L.G. *Computer Vision*, 1st ed.; Prentice Hall PTR: Hoboken, NJ, USA, 2001.
25. Larbi, M.; Messali, Z.; Hafaifa, A.; Kouzou, A.; Fortaki, T. An image segmentation algorithm based on LSM with stochastic constraint applied to computed tomography images. *EEA—Electroteh. Electron. Autom.* **2019**, *67*, 87–96.
26. Liu, J.; Wei, X.; Li, L.M.R. image segmentation based on level set method. *Multimed. Tools Appl.* **2020**, *79*, 11487–11502. [CrossRef]
27. Jiang, X.; Zhang, R.; Nie, S. Image Segmentation Based on Level Set Method. *Phys. Procedia* **2012**, *33*, 840–845. [CrossRef]
28. Katopodes, N.D. Level Set Method. In *Free-Surface Flow*; Elsevier: Amsterdam, The Netherlands, 2019; pp. 804–828.
29. Ždímalová, M.; Major, J.; Kopáni, M. Graph cutting and its application to biological data. *Open Phys.* **2019**, *17*, 468–479. [CrossRef]
30. Yi, F.; Moon, I. Image segmentation: A survey of graph-cut methods. In Proceedings of the 2012 International Conference on Systems and Informatics (ICSAI2012), Yantai, China, 19–20 May 2012; pp. 1936–1941. [CrossRef]
31. Boykov, Y.; Funka-Lea, G. Graph cuts and efficient N-D image segmentation. *Int. J. Comput. Vis.* **2006**, *70*, 109–131. [CrossRef]
32. Kolmogorov, V.; Zabih, R. What energy functions can be minimized via graph cuts? *Lect. Notes Comput. Sci. (Incl. Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinform.)* **2002**, *2352*, 65–81. [CrossRef]
33. Rueden, C.T.; Schindelin, J.; Hiner, M.C.; DeZonia, B.E.; Walter, A.E.; Arena, E.T.; Eliceiri, K.W. Image J2: Image J for the next generation of scientific image data. *BMC Bioinform.* **2017**, *18*, 529. [CrossRef]
34. Image J. Available online: <https://imagej.nih.gov/ij/index.html> (accessed on 15 October 2021).
35. CellProfiler. Cell Image Analysis Software. Available online: <https://cellprofiler.org/> (accessed on 15 October 2021).
36. Ilastik the Interactive Learning and Segmentation Toolkit. Available online: <https://www.ilastik.org/> (accessed on 15 October 2021).
37. Geman, D. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *J. Appl. Stat.* **1993**, *20*, 25–62. [CrossRef]
38. Maeda, J.; Ishikawa, C.; Novianto, S.; Tadehara, N.; Suzuki, Y. Rough and accurate segmentation of natural color images using fuzzy region-growing algorithm. In *Proceedings of the 15th International Conference on Pattern Recognition. ICPR-2000, Barcelona, Spain, 3–7 September 2000*; IEEE Computer Society: Washington, DC, USA, 2000; Volume 3, pp. 638–641.
39. Magzhan, K.; Jani, H. A Review and Evaluations of Shortest Path Algorithms. *Int. J. Sci. Technol. Res.* **2013**, *2*, 99–104.
40. Moghaddamzadeh, A.; Bourbakis, N. A fuzzy region growing approach for segmentation of color images. *Pattern Recognit.* **1997**, *30*, 867–881. [CrossRef]
41. Horowitz, S.L.; Pavlidis, T. Picture Segmentation by a Directed Split and Merge Procedure. In Proceedings of the 2nd International Joint Conference on Pattern Recognition, Copenhagen, Denmark, 13–15 August 1974; pp. 424–433.
42. Ohlander, R.; Price, K.; Reddy, D.R. Picture segmentation using a recursive region splitting method. *Comput. Graph. Image Process.* **1978**, *8*, 313–333. [CrossRef]
43. Jain, S.; Salau, A.O. An image feature selection approach for dimensionality reduction based on kNN and SVM for AkT proteins. *Cogent Eng.* **2019**, *6*, 1–14. [CrossRef]
44. Ždímalová, M.; Krivá, Z.; Bohumel, T. Graph Cuts in Image Processing. In *Proceedings of the Aplimat 2015: 14th Conference on Applied Mathematics, Bratislava, Slovak Republic, 3–5 February 2015*; Faculty of Mechanical Engineering, Slovak University of Technology: Bratislava, Slovakia, 2015; pp. 774–786.
45. Watanabe, T.; Tan, Z.; Wang, X.; Martinez-Hernandez, A.; Frahm, J. Magnetic resonance imaging of noradrenergic neurons. *Brain Struct. Funct.* **2019**, *224*, 1609–1625. [CrossRef] [PubMed]
46. Meadowcroft, M.D.; Connor, J.R.; Yang, Q.X. Cortical iron regulation and inflammatory response in Alzheimer’s disease and APPSWE/PS1ΔE9 mice: A histological perspective. *Front. Neurosci.* **2015**, *9*, 255. [CrossRef] [PubMed]
47. Svobodová, H.; Kosnáč, D.; Balázsová, Z.; Tanila, H.; Miettinen, P.O.; Sierra, A.; Vitovič, P.; Wagner, A.; Polák, S.; Kopáni, M. Elevated age-related cortical iron, ferritin and amyloid plaques in APPSWE/PS1ΔE9 transgenic mouse model of Alzheimer’s disease. *Physiol. Res.* **2019**, *68*, S445–S451. [CrossRef] [PubMed]
48. Meguro, R.; Asano, Y.; Odagiri, S.; Li, C.; Iwatsuki, H.; Shoumura, K. Nonheme-iron histochemistry for light and electron microscopy: A historical, theoretical and technical review. *Arch. Histol. Cytol.* **2007**, *70*, 1–19. [CrossRef]

49. Zhang, D.; Islam, M.M.; Lu, G. A review on automatic image annotation techniques. *Pattern Recognit.* **2012**, *45*, 346–362. [[CrossRef](#)]
50. Ankerst, M.; Breunig, M.M.; Kriegel, H.; Sander, J. OPTICS: Ordering Points to Identify the Clustering Structure. *ACM SIGMOD Rec.* **1999**, *28*, 49–60. [[CrossRef](#)]
51. Breunig, M.M.; Kriegel, H.P.; Ng, R.T.; Sander, J. OPTICS-OF: Identifying local outliers. *Lect. Notes Comput. Sci. (Incl. Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinform.)* **1999**, *1704*, 262–270. [[CrossRef](#)]
52. Kriegel, H.; Kröger, P.; Sander, J.; Zimek, A. Density-based clustering. *WIREs Data Min. Knowl. Discov.* **2011**, *1*, 231–240. [[CrossRef](#)]
53. Ester, M.; Kriegel, H.P.; Sander, J.; Xiaowei, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, OR, USA, 2–4 August 1996*; AAAI Press: Palo Alto, CA, USA, 1996; pp. 226–231, ISBN 1-57735-004-9.
54. Otsu, N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Trans. Syst. Man. Cybern.* **1979**, *9*, 62–66. [[CrossRef](#)]
55. Kullback, S.; Leibler, R.A. On Information and Sufficiency. *Ann. Math. Stat.* **1951**, *22*, 79–86. [[CrossRef](#)]
56. Goh, T.Y.; Basah, S.N.; Yazid, H.; Aziz Safar, M.J.; Ahmad Saad, F.S. Performance analysis of image thresholding: Otsu technique. *Measurement* **2018**, *114*, 298–307. [[CrossRef](#)]
57. Lapuyade-Lahorgue, J.; Xue, J.-H.; Ruan, S. Segmenting Multi-Source Images Using Hidden Markov Fields With Copula-Based Multivariate Statistical Distributions. *IEEE Trans. Image Process.* **2017**, *26*, 3187–3195. [[CrossRef](#)] [[PubMed](#)]