

Article

Metaheuristic Algorithms Applied to Color Image Segmentation on HSV Space

Donatella Giuliani

School of Economics, Management and Statistics, University of Bologna, 40126 Bologna, Italy;
donatella.giuliani@unibo.it

Abstract: In this research, we propose an unsupervised method for segmentation and edge extraction of color images on the HSV space. This approach is composed of two different phases in which are applied two metaheuristic algorithms, respectively the Firefly (FA) and the Artificial Bee Colony (ABC) algorithms. In the first phase, we performed a pixel-based segmentation on each color channel, applying the FA algorithm and the Gaussian Mixture Model. The FA algorithm automatically detects the number of clusters, given by histogram maxima of each single-band image. The detected maxima define the initial means for the parameter estimation of the GMM. Applying the Bayes' rule, the posterior probabilities of the GMM can be used for assigning pixels to clusters. After processing each color channel, we recombined the segmented components in the final multichannel image. A further reduction in the resultant cluster colors is obtained using the inner product as a similarity index. In the second phase, once we have assigned all pixels to the corresponding classes of the HSV space, we carry out the second step with a region-based segmentation applied to the corresponding grayscale image. For this purpose, the bioinspired Artificial Bee Colony algorithm is performed for edge extraction.

Keywords: color image segmentation; color image processing; metaheuristic algorithms; firefly algorithm; artificial bee colony algorithm



Citation: Giuliani, D. Metaheuristic Algorithms Applied to Color Image Segmentation on HSV Space. *J. Imaging* **2022**, *8*, 6. <https://doi.org/10.3390/jimaging8010006>

Academic Editor: Edoardo Provenzi

Received: 11 December 2021

Accepted: 31 December 2021

Published: 5 January 2022

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



Copyright: © 2022 by the author. 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/>).

1. Introduction

Image segmentation is the decomposition of an image into meaningful structures, which is a key step in image processing, with the main purpose of facilitating tasks at higher levels, such as object detection, recognition and classification, in passing from image processing to image analysis and image understanding [1]. The basic goal of any image segmentation process is to subdivide an image into components belonging to different objects or to different parts of an object. Theoretically, pixels derived by the same component should have similar properties, forming a connected region [2]. During recent decades, many color segmentation methods have been proposed in the literature; for an in-depth overview, refer to [3–5]. A general and broad classification of segmentation image techniques is reported in [6], where a low-level taxonomy is introduced based on distinguishing segmentation methods into spatially guided and spatially blind, where the former perform the spatial arrangement of pixels, unlike the second ones.

Color segmentation techniques can also be divided into three main categories: feature-space-based methods, image-domain-based methods and physics-based methods [7]. In the first class, cluster segments are generated in such a way that they are homogeneous with respect to the characteristics of the feature space, such as intensity level, color or texture. After mapping pixels into a color space, they are allocated in each cluster based on their features, recurring to predefined similarity criteria. Generally, feature-space techniques are spatially blind; they ignore the spatial distribution of pixel colors. Histogram thresholding techniques can be ascribed to this category. In thresholding techniques, pixels are partitioned according to their intensity or color levels, recurring to a global thresholding

or multiple thresholds [8,9]. Histograms are composed of relatively separated parts of intensities, each representing an object of the image. In partitional clustering, segmentation is carried out by partitioning the data into a defined number of clusters; K-means and Fuzzy C-Means algorithms belong to this class [10,11]. Image data can be grouped in a one-dimensional or three-dimensional space, depending on whether it is a grayscale or a color image [12]. In our approach, it is not necessary to predefine the number of clusters, as they will be automatically determined.

Beside homogeneity, a second basic property of segmented regions must be the spatial compactness. Generally, feature-space-based techniques do not take into account the spatial locations of pixels, as they provide a global description of the image. On the contrary, image-domain-based techniques are spatially guided, therefore pixels are clustered based on their spatial relationships, assuming that points of the same object are usually nearby. Region-based, energy-based and edge-based algorithms are included in this class, with the aim to form homogenous and compact groups of pixels from the geometrical point of view. The hierarchical clustering methods are included into this category as well, since they produce image subdivision as nested series of partitions based on a criterion for splitting or merging clusters attempting to group pixels into homogenous regions [13,14]. In fact, superpixel algorithms strongly reduce the juxtaposition between these two categories. Recently, Ren and Malik [15] introduced a novel technique based on superpixels that segments an image into regions by considering their proximity and similarity measures defined using image features. Superpixels replace the rigid pixel structure by delineating regions formed by groups of pixels that look similar, making other processing tasks simpler than using single image pixels.

Principally, color segmentation techniques belonging to the two aforementioned categories are based on monochrome segmentation approaches, operating in a specific three-dimensional color space. In most cases, they are a sort of dimensional extension of grayscale image methods. On the other hand, the methodologies which are part of the third category are diversified according to the physical models applied for describing light interaction with materials. Consequently, they do not correspond to any monochromatic segmentation methods. In color image processing, physical models aim at eliminating the effect of highlights and shadowing, [16]. The Healey's reflection model was one of the first attempts at analyzing the geometrical scene and the physical nature of materials [17].

In the present work, we apply a feature-space-based method in which pixels are identified by the three components of a preselected color space [18]. Consequently, it is very important to define which color space is going to be used because the similarity measure will be defined within it. The quality of segmentation may depend considerably on the employed color space [19]. By using a suitable color space, some segmentation techniques for monochrome images can be extended to segment color images ensuring reliable results. Furthermore, we need to keep in mind that segmentation of color images frequently is viewed as an ill-defined problem, meaning that there are multiple acceptable solutions, due to its intrinsically subjective nature.

This paper is structured as follows: Section 2 is composed of two parts, the introduction of the feature-space-based color segmentation method in Section 2.1 and the implemented region-based method for edge extraction in Section 2.2; Section 3 includes the results achieved applying the two proposed methodologies, respectively, in Sections 3.1 and 3.2. Finally, we present the conclusion in Section 4.

2. Materials and Methods

2.1. Color Image Segmentation Method

In this work, a monochromatic-based method was implemented, starting with image decomposition into three different components. After that, each component is processed separately, and finally, the individually achieved results are recombined together [20]. Regarding the selection of color space, we opted for the HSV. In fact, RGB color space is adequate for displaying images but not for image processing, because intensity is not decou-

pled by chromaticity, hence RGB color space does not produce satisfactory segmentation results [21].

As for the HSV space, we need to clarify that hue is the chromatic feature describing a pure color (red, yellow, etc.), saturation quantifies the amount of gray in a particular color, and generally if saturation appears as a range from 0 to 1, 0 represents gray color whereas 1 is a primary color. Value is the intensity or brightness of the color, where 0 is completely black, and 1 is the brightest. The hue and saturation, or alternatively, intensity, components emulate the human perception of color. More precisely, hue is the dominant wavelength, whereas saturation is its purity, or more specifically, the inverse of the amount of white light contained in the color. The value component is apart from chromaticity, so it is decoupled from hue and saturation.

After defining the color space, an illumination equalization was performed, applying a Gaussian blurring to the value channel of the image with standard deviation sigma equal to 1.5; this gives us a local average for the illumination. We have to keep in mind that color representation is sensitive to illumination, so two colors with the same chromaticity can be recognized as different if they have different lighting intensity. This fact makes the clustering processes inefficient, because pixels from the same class but with different illumination can be identified as pixels from separate classes.

After finishing this preprocessing step, we proceeded to a histogram-based segmentation approach applied to each color channel, which uses a metaheuristic algorithm to automatically define the number of clusters and the histogram maxima [22–25]. Metaheuristic algorithms are a class of approximate methods that allow us to discover possible solutions by exploring a search space in order to find near-optimal solutions. They are iterative processes developed to search for a solution that is good enough in a time that is small enough [26]. These algorithms are frequently nature-inspired, and they have the advantages of finding global optima due to the action of multiple search agents which are randomly generated [27]. The solution of an optimization problem with a metaheuristic algorithm implies an initialization step generating one or more random solutions. In each iteration step, the current solution is then changed by a new one, created by search operators, with a global optimization approach composed by two schemes: the exploitation of new solutions with the goal of improving the quality of solutions and the exploration of the entire search space to prevent the selection of local optima.

In searching maxima of histogram distributions, we suggest the use of this class of optimization algorithms because they guarantee an enhancement of convergence into global optima despite the presence of numerous local maxima, thanks to the simultaneous action of multiple agents moving randomly all around the research space, [28,29]. To this end, we applied the Firefly Algorithm that employs fireflies as search agents, making use of their idealized flashing characteristics to locate the most significant peaks of grayscale histogram for each component [30,31].

Subsequently, the detected maxima are used as initial values of cluster means for the parameter estimation of a Gaussian Mixture Model [32]. The probability density function of a Gaussian Mixture Model is expressed as a weighted sum of Gaussian density functions, whose parameters are evaluated applying the Expectation-Maximization (EM) technique [33]. The coefficients of the linear combination of Gaussians can be seen as prior probabilities of each component, while the posterior probabilities, derived by the Bayes rule, can be used for assigning pixels to clusters without recurring to a thresholding process. The Gaussian Mixture Model (GMM) is a parametric probability density function represented as a weighted sum of Gaussian component densities. The GMM parameters are estimated from data using the iterative Expectation-Maximization (EM) algorithm. As we know, a univariate Gaussian density distribution is expressed by:

$$p(x | \mu, \sigma^2) = N(x | \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right) \quad (1)$$

where $N(x | \mu, \sigma^2)$ represents the Gaussian or normal distribution of mean μ and standard deviation σ . A Gaussian mixture model is a weighted sum of K components of Gaussian densities, analytically given by:

$$p(x) = \sum_{k=1}^K \pi_k \cdot N(x | \mu_k, \sigma_k^2) \tag{2}$$

Equation (2) represents a linear superposition of Gaussian probability densities and the mixing coefficients π_k indicates the weight of each distribution. After the initialization phase, the Gaussian parameters μ_k, σ_k^2 and the coefficients π_k of the linear combination are evaluated using the Expectation Maximization (EM) algorithm. At each iteration, the EM algorithm computes the responsibilities γ_k , so defined:

$$\gamma_k(x) = p(k|x) = \frac{\pi_k \cdot N(x | \mu_k, \sigma_k^2)}{\sum_{i=1}^K \pi_i \cdot N(x | \mu_i, \sigma_i^2)} \tag{3}$$

$\gamma_k(x)$ represents the posterior probability of a given intensity x to belong to the k -th cluster, according to the Bayes rule. In short, the responsibilities estimate the grades of membership indicating the degree to which data points belong to each cluster. Consequently, the assignment of pixels of a given gray level x_i to the cluster k is performed by means of the evaluation of the maximum value of responsibilities as k varies, given that, by definition, $\gamma_k(x_i)$ indicates the probability of the k -th GMM's component to have generated the value x_i .

The issues of the segmentation process applied to each color channel are obtained independently, then they are recombined together to compose the final segmented color image. After that, it performs a reduction in the number of distinct colors recurring to the parallelism of vectors represented in HSV space. In order to do so, we would like to reiterate the fundamental laws of colorimetry, that state [34]:

1. Any color can be defined by three values and the combination of the three components is unique;
2. Two colors are equivalent after multiplying or dividing the three components by the same number;
3. The luminance of a mixture of colors is equal to the sum of the luminance of each color.

According to the second statement, we have chosen the cross product to identify parallel vectors in the HSV space. Indeed, if two vectors have the same direction, or equivalently if they are linearly dependent, their cross product is zero. In this context, if the cross product is approximately null it implies that colors are very similar, so they can be considered as belonging to the same class.

At the end of the process, the performance of a clustering algorithm must be estimated. The academic literature in this field has suggested several performance metrics to assess the validity of cluster partitions [35,36]. Basically, three different techniques are implemented for evaluating the efficiency of clustering algorithms: external criteria, internal criteria and relative criteria [37]. The external validity methods evaluate the clustering results based on their comparison to an externally known result, such as manual image segmentation performed by human users. The internal measures estimate the goodness of a clustering process without considering external information but using the data set itself. Finally, relative clustering validation evaluates the clustering structure by varying different parameter values for the same algorithm (for example changing the number of clusters).

In this work, the algorithm efficiency is computed through an internal clustering validation approach referring to the mean-squared error (MSE) [38]. Usually, MSE is used for assessing distortion between the original image and resulted image. For color images, the formula is extended to include the three components:

$$MSE = \frac{1}{n \cdot m \cdot p} \sum_{k=1}^p \sum_{i=1}^n \sum_{j=1}^m (I(i, j, k) - \tilde{I}(i, j, k))^2 \tag{4}$$

where $I(i, j, k)$, $\tilde{I}(i, j, k)$ are, respectively, the original and the segmented image, p is the number of the image components ($p = 3$ for color spaces), and $n \cdot m$ is the size of each component. A low MSE value means that the predicted values are close to the real values, practically, the RMSE will be used, defined by: $\text{RMSE} = \sqrt{\text{MSE}}$. This RMSE depends on the orders of magnitude of the observed values. Therefore, it can vary significantly from one application to the next. To resolve this problem, we could consider the relative absolute error (RAE) defined as follows:

$$\text{RAE} = \frac{1}{n \cdot m \cdot p} \sum_{k=1}^p \sum_{i=1}^n \sum_{j=1}^m \left| \frac{I(i, j, k) - \tilde{I}(i, j, k)}{I(i, j, k)} \right| \quad (5)$$

Small values of validation indices imply that the estimated image is close to the initial one.

2.2. Edge Extraction Applying Artificial Bee Colony Algorithm

After the color segmentation process, we proceed to a region-based segmentation of the corresponding gray image in order to extract edges of homogenous components. The first step in region growing is to select a set of seed points. The region begins to grow from the location of these seeds. In the present work, the selection of initial seed points was achieved through the Artificial Bee Colony algorithm.

The ABC algorithm is a swarm-based metaheuristic algorithm that was introduced by Karaboga in 2005 [39] for optimizing numerical problems. It was inspired by the intelligent foraging behavior of honeybees in nature. The algorithm is specifically based on the model for the foraging behavior of honeybee colonies [40,41]. In the ABC algorithm, the colony of artificial bees consists of three groups: employed bees, onlookers and scouts [25]. Employed bees are those who have discovered a food source. The employed bee whose food source is abandoned becomes a scout bee, starting new random research around the hive. The exchange of information among bees is the most important occurrence in the formation of collective knowledge. Communication among bees related to the quality of food sources takes place in the dancing area of the hive; this dance is called waggle dance. After localizing a source, employed bees share nectar and position information of the food sources with onlooker bees executing the waggle dance. An onlooker bee evaluates nectar information taken from employed bees and decides to employ herself at the most profitable source, with a probability related to the nectar amount [42]. In this context we have adapted the ABC method, considering as food sources the areas of the gray image with pixels not yet assigned to any cluster. Their fruitfulness is greater the greater their extension. The onlooker bees come to the aid of the employed bees in a number proportional to the size of the identified food source and to the number of pixels belonging to it not yet grouped together into any homogeneous region.

3. Results

3.1. Results of the Color Image Segmentation Method

As an initial test image, we have considered the BSD image #295087 extracted by the Berkeley Segmentation Dataset BSDS500. The original image contains 61,258 unique colors (Figure 1).

During the segmentation of the hue component, the FA algorithm identified four different clusters of intensities 16, 35, 124, 147, respectively, and the final outcome is shown in Figure 2. As we can see, in the original image two predominant hues appear, the first ranging from ocher to dark brown and the second one is due to the blue sky of the background. The validation of the grayscale segmentation was performed by using the Root-Mean-Square Error (RMSE) and the Normalized Correlation Coefficient (NK) [43–45]. For the hue component we have obtained $\text{RMSE} = 0.0247$ and $\text{NK} = 0.98$.

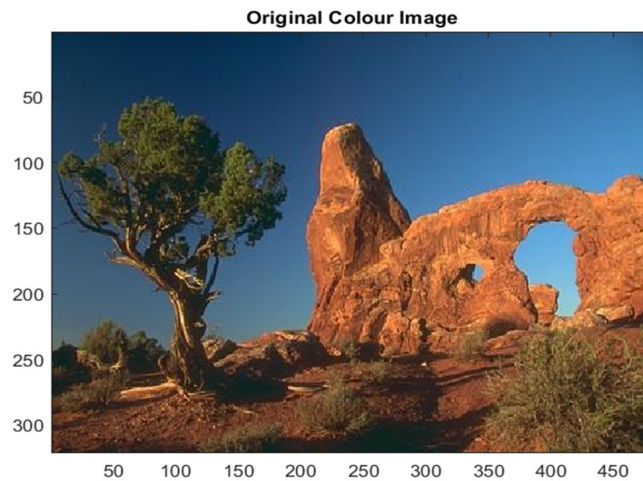


Figure 1. Original image #295087.

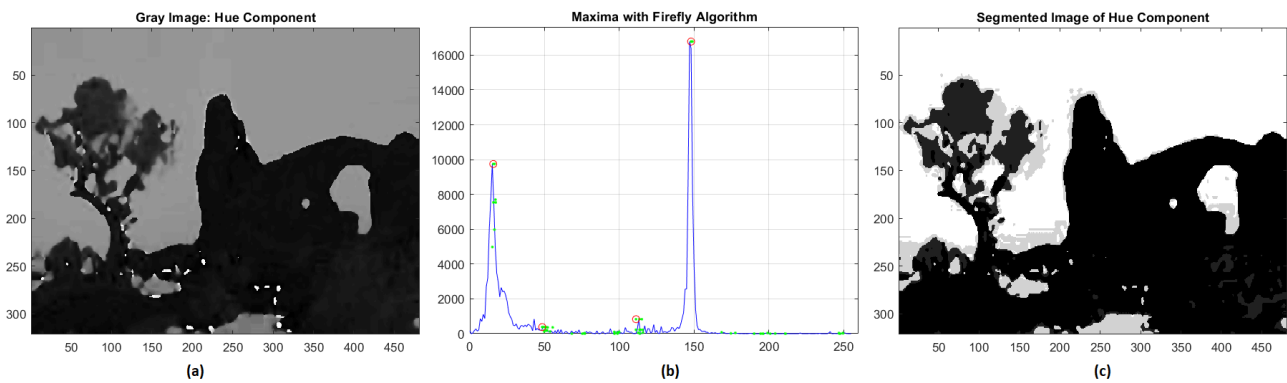


Figure 2. (a) Hue component; (b) global maxima of histogram distribution by FA; (c) segmentation of hue component.

The gray distribution of the saturation component is however more complex. The results of the corresponding segmentation are represented in Figure 3, and the evaluated gray levels of the seven clusters are 72, 101, 122, 145, 169, 204, 228 in increasing order. The validation indices are RMSE = 0.0418 and NK = 0.9947, respectively.

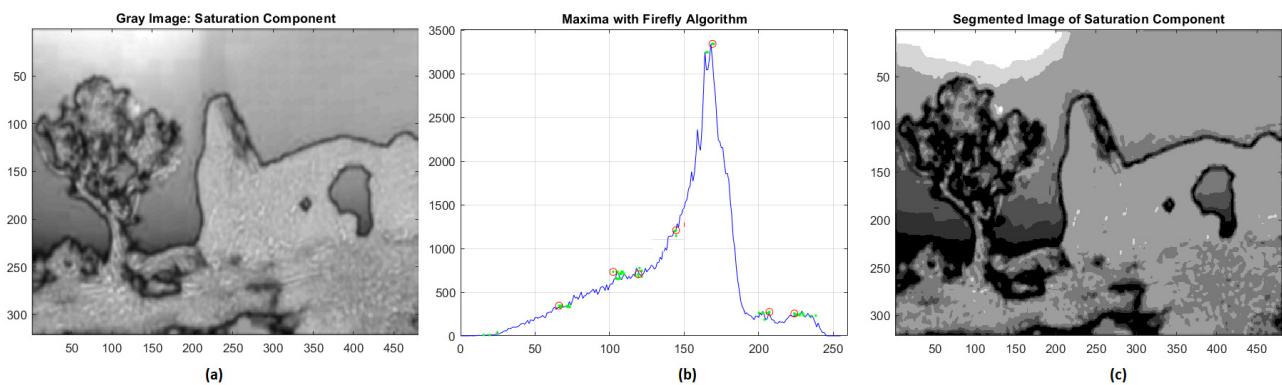


Figure 3. (a) Saturation component; (b) global maxima of histogram distribution by FA; (c) segmentation of saturation component.

Regarding the value component, the great variability of the histogram gives rise to seven different clusters with gray intensities equal to 44, 81, 115, 146, 162, 183, 224. Even in this case, we obtained very reliable results with RMSE = 0.0363 and NK = 0.9899 (Figure 4).

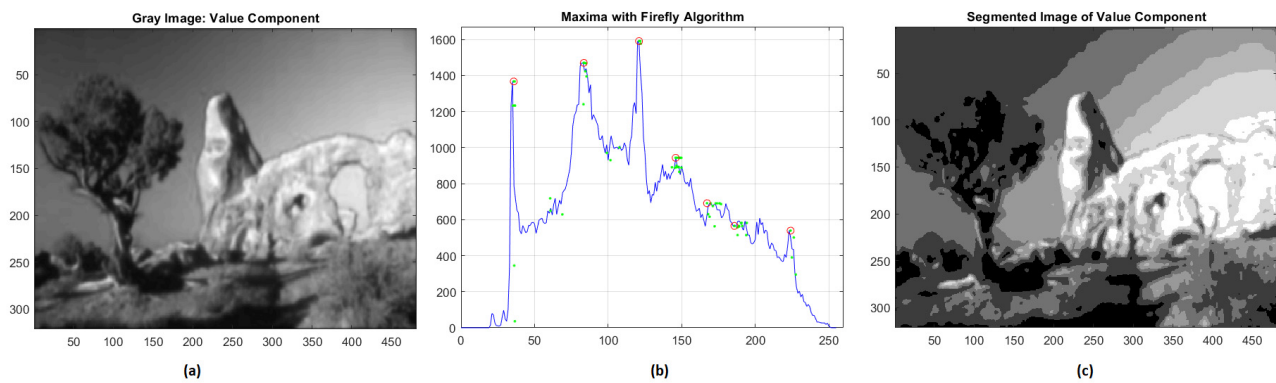


Figure 4. (a) Value component; (b) global maxima of histogram distribution by FA; (c) segmentation of value component.

After having processed each component separately, we proceed to recombine the images in order to compose the final image, in which 132 different colors are present (Figure 5). The procedure performed a color reduction of 99.7%. For the segmented image, the validation indices, computed with Equations (4) and (5), are $RMSE = 0.0618$ and $RAE = 0.9729$, respectively.

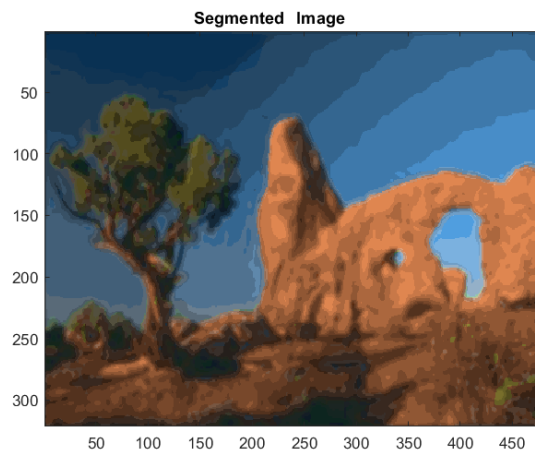


Figure 5. Segmented color image with 132 colors.

As previously asserted, the variability in adequate solutions for image segmentation is an intrinsic and unavoidable feature, primarily due to the differences in the level of attention, the degree of detail perceived by one human observer compared to another, and the type of represented objects in which the user is interested. However, when pixel colors are projected onto three components, color information are widely scattered, and therefore, one of the drawbacks of color image processing is how to employ this great amount of information. To address this, we performed a color reduction relating to the evaluation of the inner product among vectors in the HSV space. Figure 6 allows us to compare the segmented images after doing cluster reduction. Iteratively applying the procedure, at first we obtained a reduction of 34%, passing from an initial 132 clusters to 87, and then of a further 34.4% with respect to the previous one, reducing the colors to 57 and finally to 37 (Figure 6). Table 1 shows the values of RMSE and RAE from the original image (Figure 1) and the successive segmented ones (Figure 6a–c).

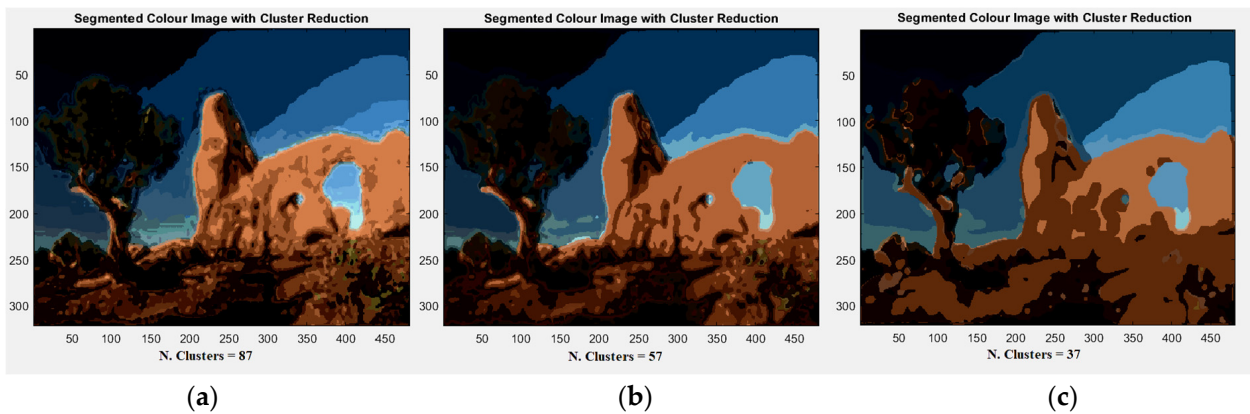


Figure 6. (a) Segmented image with 87 colors; (b) segmented image with 57 colors; (c) segmented image with 37 different colors.

Table 1. Root-mean-squared errors and relative absolute errors.

| Number of Clusters | RSME | RAE |
|--------------------|---------|--------|
| 132 | 0.0618 | 0.9729 |
| 87 | 0.0690 | 1.0087 |
| 57 | 37.0155 | 0.9960 |
| 37 | 37.2132 | 0.9963 |

The BSD image #295087 represents a case with a low color content, essentially blue, brown and green, but with a high texture content, as we can notice by observing the chromatic distribution of the original image shown in Figure 7.

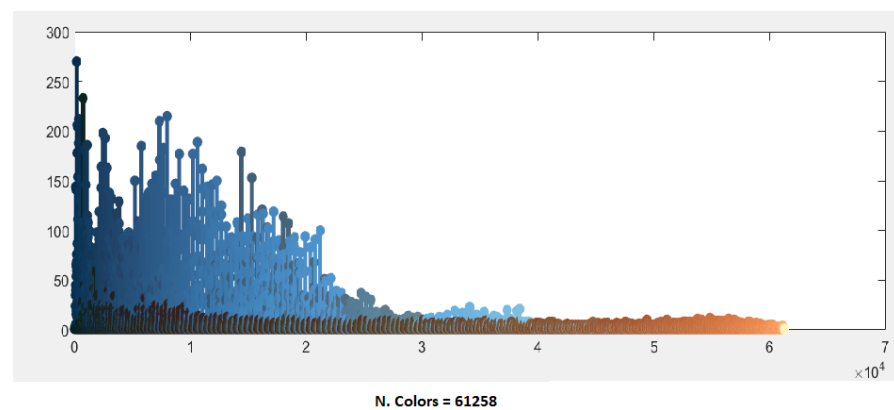


Figure 7. Distribution of the initial 61,258 colors of image #295087.

In Figures 8 and 9, the significant color reduction is highlighted through the three-dimensional scatter diagrams and the bidimensional chromatic distributions related to the segmented images with 132, 87, 57 and 37 distinct colors, respectively. The relevant decrement of colors may avoid an over-segmentation because of merging pixels with similar colors.

This method has also been applied to some other images extracted by the Berkeley Segmentation Dataset BSDS500. In the test image #118035 of BSD, the initial 23,786 unique colors are reduced to 19 (Figure 10).

The training image #35010 of BSD contains 61,267 colors, the final image is represented with only 219 different colors. Nevertheless, the basic chromatic characteristics of the butterfly and the surroundings are preserved (Figure 11).

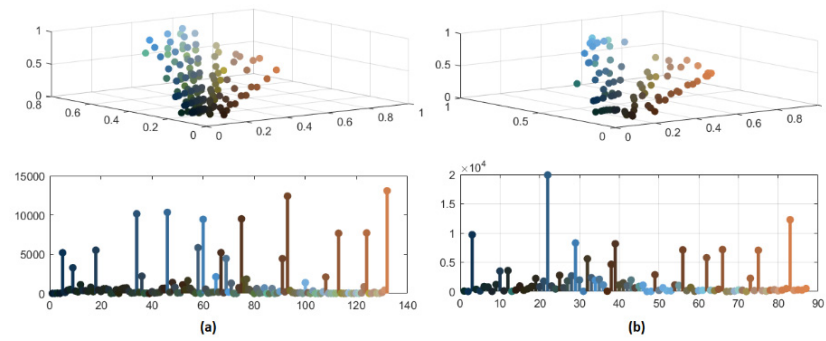


Figure 8. (a) Color distribution of segmented image with 132 colors; (b) color distribution of segmented image with 87 colors.

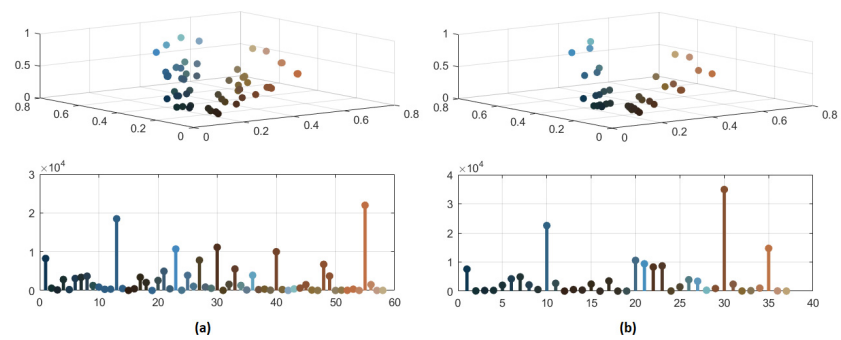


Figure 9. (a) Color distribution of segmented image with 57 colors; (b) color distribution of segmented image with 37 colors.

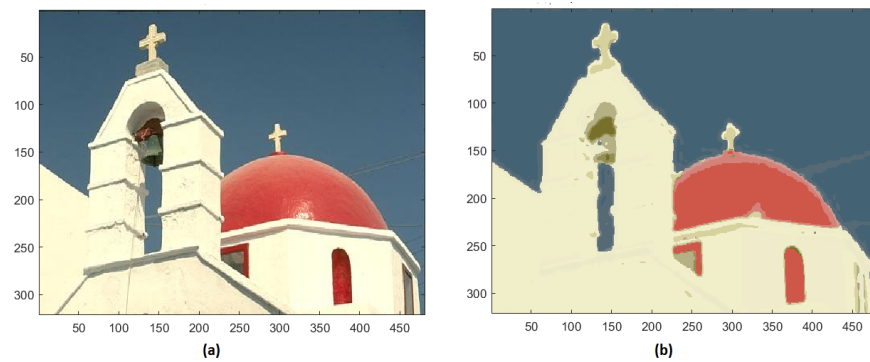


Figure 10. (a) Original image #118035 of BSD; (b) segmented image with 19 colors.

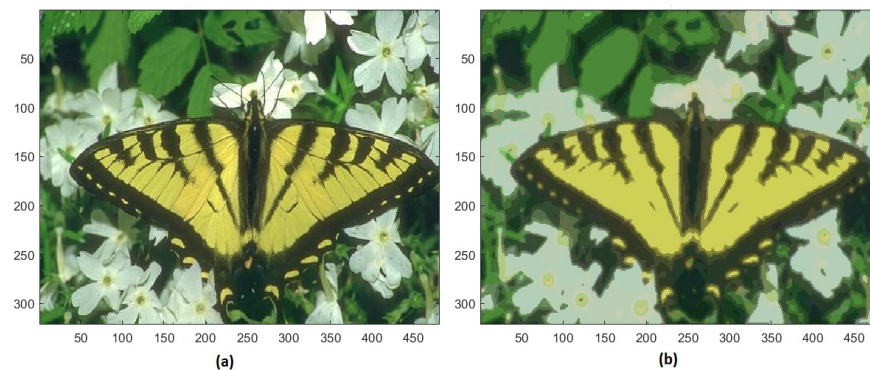


Figure 11. (a) Original image #35010 of BSD; (b) segmented image with 129 colors.

Analyzing the training image #296059, the initial 27,871 colors are reduced to 48 different colors, and the resulting segmented image is shown in Figure 12. The complexity of the ground texture and the elephant skin is strongly simplified, while the tusks are still clearly distinguishable.

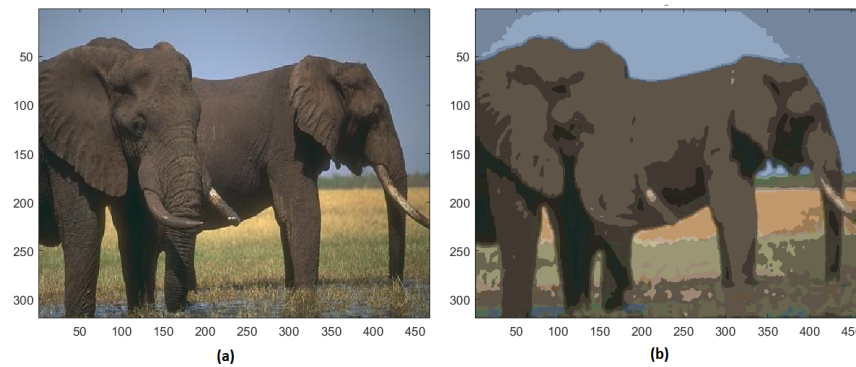


Figure 12. (a) Original image #296059 of BSD; (b) segmented image with 48 colors.

For image #198023 with 31,863 colors, the reduction gives rise to 157 colors. With a further reduction to 124 different colors, the small squares of the grating behind the woman are no longer distinguishable, this makes the foreground more recognizable from the background (Figure 13).

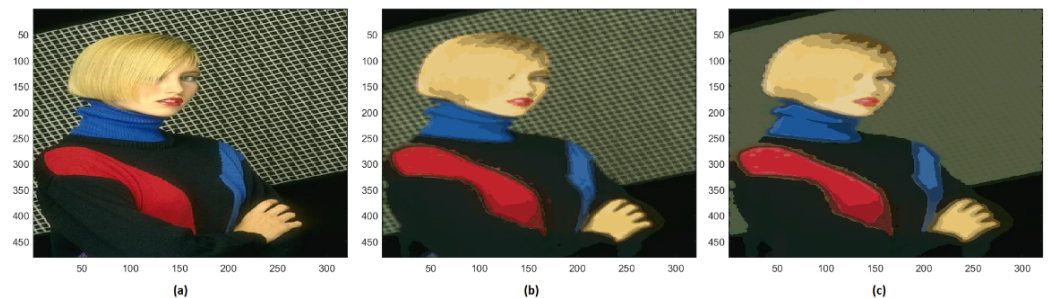


Figure 13. (a) Original image #296059 of BSD; (b) segmented image with 157 colors; (c) segmented image 124 with different colors.

3.2. Results of Edge Extraction Applying Artificial Bee Colony Algorithm

In this paper, initially scout bees move in the search space, which is the gray image, describing random paths, each of which is a piecewise linear curve composed by a connected sequence of M arbitrary line segments. The trajectories of scout bees are defined by the following parametric equations:

$$x_{k+1}(t) = x_k(t) + v_0 \cdot \text{rand}(1) \cdot \cos(\text{rand}(1) \cdot \theta)$$

$$y_{k+1}(t) = y_k(t) + v_0 \cdot \text{rand}(1) \cdot \sin(\text{rand}(1) \cdot \theta)$$

where $k = 1, \dots, M, t \in [0; 1], v_0$ is the initial velocity, $\theta \in [0; 2\pi]$ and $\text{rand}(1)$ is a generator of random numbers uniformly distributed in the interval $[0; 1]$, the end points of each line segment determine the positions of unemployed bees during their flights. Initially, the paths will be confined inside the image space, so as not to go beyond edges. If along the path a scout bee finds a food source, which is a zone with unclassified pixels, the growing process will be activated starting from the actual position, otherwise the bee keeps going undisturbed. Once the region with uniform gray intensity is outlined, its edges are extracted and the bounded box of boundaries is determined (Figure 14).

At this point, employed bees share their food source information with onlooker bees waiting in the hive and then onlooker bees choose their food sources depending on this

information. The scout bees come back to the hive for executing the waggle dance in order to involve onlooker bees in the exploitation phase. In the present application of the ABC algorithm, the fitness values are computed through the percentage of pixels not yet assigned and included inside the bounded box of the extracted regions. Then, onlooker bees give rise to a local search, rushing to scouts' aid proportionally to the number of unclassified pixels and to the size of the rectangle containing the extracted boundary (Figures 15 and 16).

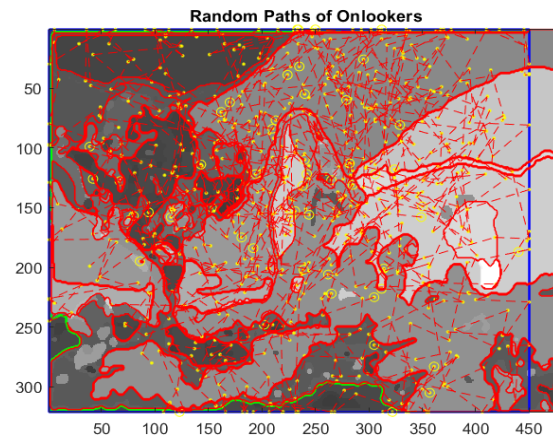


Figure 14. Random paths of onlookers.

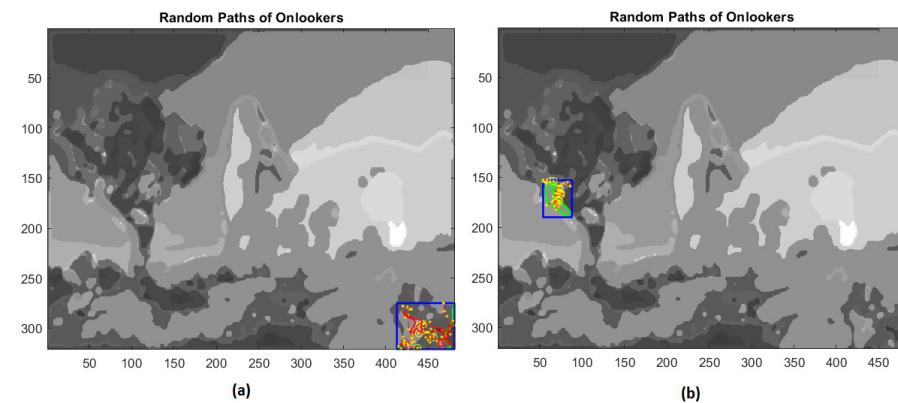


Figure 15. (a) First example of local search by onlookers in a rectangular area (blue line); (b) second example of local search by onlookers on test image.

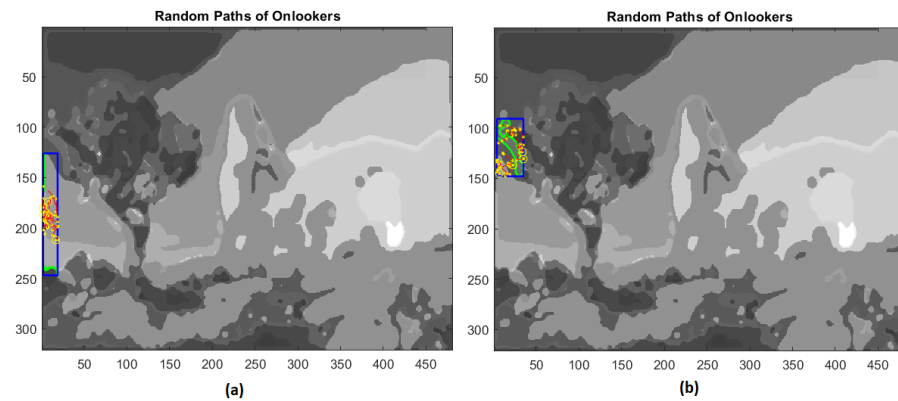


Figure 16. (a) Third example of local search by onlookers in a rectangular area (blue line); (b) fourth example of local search by onlookers on test image.

The extracted edges of the segmented image #295087 of BSD with 37 clusters are displayed in Figure 17. The algorithms, developed with Matlab, are able to detect even low significance regions that eventually could be excluded on the basis of the measure of their area or perimeter length.

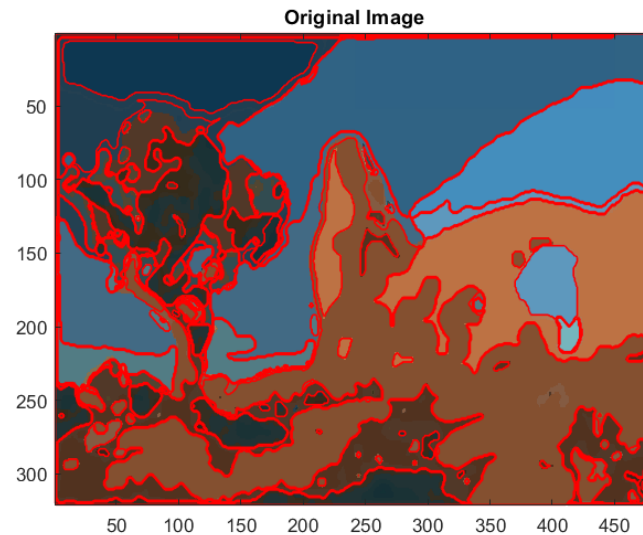


Figure 17. Edge extraction of the segmented image #295087 of BSD with 37 clusters.

4. Discussion

This work performed a color image segmentation, referring to metaheuristic and nature-inspired algorithms. The algorithms are applied to hue, saturation and value components separately. Thus, this method pertains to the category of monochrome segmentation approaches, which can be considered as a dimensional extension of grayscale image methods.

The metaheuristic Firefly Algorithm automatically evaluates the number of clusters and their initial centroids. Subsequently, the outcomes of FA are used as initial means to estimate the Gaussian Mixture Model. The multilevel image is obtained by recombining the three segmented components. A further color reduction is performed through the use of the inner product, as an index of similarity among colors. The validation analysis has been carried out using different standard measures, showing that the method is fairly robust and reliable.

Concerning the spatial segmentation, the application of the probabilistic ABC algorithm carries out boundaries of segmented regions in a fast way. The erratic motion of scout bees makes it possible to detect edges even if the size of the regions is very small. This is due to the local search activated by onlookers rushing to scouts' aid that makes the research more effective and detailed. In this context, the region-based approach is performed on the segmented grayscale image rather than on the color one. This is due to a limitation of the method hereby applied, which will have to be overcome in future research. While Balasubramanian et al. [46] applied the region-growing method on color images with a dynamic color gradient thresholding, in this research, the choice of operating with grayscale images is crucial for the application of the ABC metaheuristic algorithm in image processing, which is also one of the aims of the present work.

Funding: This research received no external funding.

Data Availability Statement: The images analysed in this research were taken from the Berkeley Segmentation Dataset BSDS500. Available at: <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html/dataset/images.html>, accessed on 10 December 2021.

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

References

1. Gonzalez, R.C.; Wood, R.E. *Digital Image Processing*, 3rd ed.; Prentice Hall: Hoboken, NJ, USA, 2007.
2. Russ, J.C.; Neal, F.B. *The Image Processing Handbook*, 7th ed.; CRC Press: Boca Raton, FL, USA, 2015.
3. Skarbek, W.; Koschan, A. *Colour Image Segmentation: A Survey*; Technical Report 94–32; Technical University of Berlin: Berlin, Germany, 1994.
4. Lucchese, L.; Mitra, S.K. *Colour Image Segmentation: A State of the Art Survey*; PINSA: Lee's Summit, MO, USA, 2001; Volume 67, pp. 207–221.
5. Fernandez-Maloigne, C. *Advanced Color Image Processing and Analysis*; Springer: Berlin/Heidelberg, Germany, 2012.
6. Vantaram, S.R.; Saber, E. Survey of contemporary trends in color image segmentation. *J. Electron. Imaging* **2012**, *21*, 4. [[CrossRef](#)]
7. Gracia-Lamont, F.; Cervantes, J.; Lopez, A.; Rodriguez, A. Segmentation of images by color features: A survey. *Neurocomputing* **2018**, *292*, 1–27. [[CrossRef](#)]
8. Ridler, T.W.; Calvard, S. Picture thresholding using an iterative selection method. *IEEE Trans. Syst. Man Cybern.* **1978**, *8*, 630–632.
9. Chen, Y.B.; Chen, O.T. Image Segmentation Method Using Thresholds Automatically Determined from Picture Contents. *EURASIP J. Image Video Process.* **2009**, *2009*, 140492. [[CrossRef](#)]
10. Kanungo, T.; Mount, D.M.; Netanyahu, N.S.; Piatko, C.D.; Silverman, R.; Wu, A.Y. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 881–892. [[CrossRef](#)]
11. Chen, J.; Pappas, T.N.; Mojsilovic, A.; Rogowitz, A. Adaptive perceptual color-texture image segmentation. *IEEE Tran. Image Proc.* **2008**, *17*, 1524–1536. [[CrossRef](#)]
12. Frigui, H.; Krishnapuram, R. A Robust Competitive Clustering Algorithm with Applications. *Comput. Vision J. IEEE Trans. Pattern Anal. Mach. Intell.* **1999**, *21*, 450–465.
13. Gu, C.; Lim, J.J.; Arbelaez, P.; Malik, J. Recognition using regions. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009.
14. Gould, S.; Fulton, R.; Koller, D. Decomposing a scene into geometric and semantically consistent regions In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009.
15. Ren, X.; Malik, J. Learning a classification model for segmentation. *Int. Conf. of Comput. Vis.* **2003**, *2*, 10–17.
16. Healey, G. Using color for geometry insensitive segmentation. *Opt. Soc. Am.* **1989**, *22*, 920–937. [[CrossRef](#)]
17. Xin, J.H.; Shen, H.L. Accurate color synthesis of three-dimensional objects in an image. *Opt. Soc. Am.* **2004**, *21*, 5. [[CrossRef](#)]
18. Chaves-Gonzalez, J.M.; Vega-Rodriguez, M.A.; Gomez-Pulido, J.A.; Sanchez-Perez, J.M. Detecting skin in face recognition systems: A color spaces study. *Digit. Signal Process.* **2009**, *20*, 806–823. [[CrossRef](#)]
19. Jurio, A.; Pagola, M.; Galar, M.; Lopez, C.; Paterna, D. A comparison study of different color spaces in clustering based image segmentation. In *Communications on Computer and Information Science*; Springer: Berlin/Heidelberg, Germany, 2010.
20. Koschan, A.; Abidi, M. Detection and classification of edges in color images: A review of vector valued techniques. *IEEE Signal Process. Mag.* **2005**, *22*, 64–73. [[CrossRef](#)]
21. Ruiz-Ruiz, G.; Gomez-Gil, J.; Navas-Gracia, L.M. Testing different color space based on hue for the environmentally adaptive segmentation algorithm (ESA). *Comput. Electron. Agric.* **2009**, *68*, 88–96. [[CrossRef](#)]
22. Alata, O.; Quintard, L. Is there a best color space for color image characterization or representation based on multivariate Gaussian Mixture Model? *Comput. Vis. Image Under-Standing* **2009**, *113*, 867–877. [[CrossRef](#)]
23. Yang, X.S. *Nature-Inspired Metaheuristic Algorithms*; Luniver Press: Beckington, UK, 2008.
24. Parpinelli, R.S.; Lopes, H.S. New inspirations in swarm intelligence: A survey. *Int. J. Bio-Inspired Comput.* **2011**, *3*, 1–16. [[CrossRef](#)]
25. Pham, D.T.; Ghanbarzadeh, A.; Koc, E.; Otri, S.; Rahim, S.; Zaidi, M. The Bees Algorithm-A Novel Tool for Complex Optimisation Problems. In *Intelligent Production Machines and Systems, 2nd I*PROMS Virtual International Conference, 3–14 July 2006*; Elsevier: Amsterdam, The Netherlands, 2006; pp. 454–459.
26. Kennedy, J.; Eberhart, R.; Shi, Y. *Swarm Intelligence*; Academic Press: Cambridge, MA, USA, 2001.
27. Blum, C.; Roli, A. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Comput. Surv.* **2003**, *35*, 268–308. [[CrossRef](#)]
28. Rothlauf, F. *Design of Modern Heuristics Principles and Application*; Springer: Berlin/Heidelberg, Germany, 2011.
29. Senthilnath, J.; Vipul, D.; Omkar, S.N.; Mani, V. Clustering using levy flight cuckoo search. In Proceedings of the 7th International Conference on Bio-Inspired Computing: Theories and Applications, Advances in Intelligent Systems and Computing, Beijing, China, 2–4 November 2018; Lecture Notes in Computer Science (LNCS). Springer: Delhi, India, 2012; pp. 65–75.
30. Senthilnath, J.; Omkar, S.N.; Mani, V. Clustering using firefly algorithm: Performance study. *Swarm Evol. Comput. Elsevier* **2011**, *1*, 164–171. [[CrossRef](#)]
31. Giuliani, D. Colour Image Segmentation based on Principal Component Analysis with application of Firefly Algorithm and Gaussian Mixture Model. *Int. J. Image Process.* **2018**, *12*, 4.
32. Giuliani, D. A Grayscale Segmentation Approach using the Firefly Algorithm and the Gaussian Mixture Model. *Int. J. Swarm Intell. Res.* **2017**, *9*, 1. [[CrossRef](#)]
33. Lindsay, B.G. Mixture Models: Theory, Geometry and Applications. In *NSF-CBMS Regional Conference Series in Probability and Statistics*; Institute of Mathematical Statistics: Hayward, CA, USA, 1995.
34. McLachlan, G.J.; Basford, K.E. *Mixture Models: Inference and Applications to Clustering*; Marcel Dekker: New York, NY, USA, 1988.

35. Chapron, M. A new Chromatic Edge Detector Used for Color Image Segmentation. In Proceedings of the IEEE International Conference on Pattern Recognition, Hague, The Netherlands, 30 August–3 September 1992; pp. 311–314.
36. Halkidi, M.; Batistakis, Y.; Vazirgiannis, M. On Clustering Validation Techniques. *J. Intell. Inf. Syst.* **2001**, *17*, 107–145. [[CrossRef](#)]
37. Saitta, S.; Raphael, B.; Smith, I.F.C. A comprehensive validity index for clustering. *Intell. Data Anal.* **2008**, *12*, 529–548. [[CrossRef](#)]
38. Bolshakova, N.; Azuaje, F. Cluster validation techniques for genome expression data. *Signal Process.* **2003**, *83*, 825–833. [[CrossRef](#)]
39. Celebi, M.E. Improving the performance of K-means for color quantization. *Image Vis. Comput.* **2011**, *29*, 260–271. [[CrossRef](#)]
40. Karaboga, D. An idea based on honey bee swarm for numerical optimization. In *Technical Report of Computer Engineering Department*; Engineering Faculty of Erciyes University: Kayseri, Turkey, 2005.
41. Lučić, P.; Teodorović, D. Computing with bees: Attacking complex transportation engineering problems. *Int. J. Artif. Intell. Tools* **2003**, *12*, 375–394. [[CrossRef](#)]
42. Karaboga, D.; Gorkemli, B.; Orzturk, C.; Karaboga, N. A comprehensive survey: Artificial Bee Colony algorithm and applications. *Artif. Intell. Rev.* **2014**, *42*, 21–57. [[CrossRef](#)]
43. Nikolić, M.; Teodorović, D. Empirical study of the Bee Colony Optimization (BCO) algorithm. *Expert Syst. Appl.* **2013**, *40*, 4609–4620. [[CrossRef](#)]
44. Jaskirat, K.; Sunil, A.; Renu, V. A comparative analysis of thresholding and edge detection segmentation techniques. *Int. J. Comput. Appl.* **2012**, *39*, 29–34.
45. Nihar, R.N.; Bikram, K.M.; Amiya, K.R. A Time Efficient Clustering Algorithm for Gray Scale Image Segmentation. *Int. J. Comput. Vis. Image Process.* **2013**, *3*, 22–32.
46. Balasubramanian, G.P.; Saber, E.; Misic, V.; Peskin, E.; Shaw, M. Unsupervised color image segmentation using a dynamic color gradient thresholding algorithm. In Proceedings of the Human Vision and Electronic Imaging XIII, San Jose, CA, USA, 27 January 2008; Volume 6806.