Lappeenranta University of Technology
School of Engineering Science
Master's Programme in Computational Engineering and Technical Physics
Intelligent Computing Major

Master's Thesis

Nikita Ashikhmin

CONTOUR SEGMENT GROUPING FOR OVERLAPPING CONVEX OBJECT SEGMENTATION

Examiners: Professor Heikki Kälviäinen
Associate Professor, Cand. Sci Gleb Radchenko

Supervisors: M.Sc. (Eng.) Sahar Zafari
Adjunct Professor, Dr. Tuomas Eerola
Dr. Jouni Sampo
Prof. Heikki Kälviäinen

ABSTRACT

Lappeenranta University of Technology
School of Engineering Science
Master's Programme in Computational Engineering and Technical Physics
Intelligent Computing Major

Nikita Ashikhmin

Contour segment grouping for overlapping convex object segmentation

Master's Thesis

2017

32 pages, 14 figures, 2 table.

Examiners: Professor Heikki Kälviäinen
Associate Professor, Cand. Sci Gleb Radchenko

Keywords: segment grouping, convex object segmentation, concavity analysis, seed point, fast radial symmetry, image processing, image analysis

Segmentation of convex object has many real-world applications, as including material analysis and morphological analysis of biological cells. The methods for convex object segmentation usually consist of the following stages: 1) contour evidence, 2) edge segmentation, 3) segment grouping and 4) estimating the full contours of the objects. This thesis presents an overview of methods and approaches that are used for convex object segmentation. These work describe a new segmentation framework which key feature is a novel segment grouping method. The proposed segment grouping method was compared with Branch and Boundaries segment grouping method on two types of data: synthetic data and real data. The synthetic data consists of images with different triangle, quadrilateral, ellipse particles, which was generated by a script, the real data consist of nanoparticles images captured using transmission electron microscopy and marked manually. The results of experiments shown that the proposed segment grouping method shows better results on images with shapes of different types.

CONTENTS

LIST OF ABBREVIATIONS

ADD	Average Distance Deviation Criteria
ARF	Adaptive Ring Filter
BB	Branch and Boundaries
BE-FRS	Bounded-Erosion Fast Radial Symmetry
CF	Coin Filter
CSS	Curvature Scale Space
DT	Distance Transform
FRS	Fast Radial Symmetry
IF	IRIS Filter
LCF	Local Convergence Filter
SBF	Sliding Band Filter
UE	Ultimate Erosion
UECS	Ultimate Erosion for Convex Sets

1 INTRODUCTION

1.1 Background

Segmentation or contour estimation of overlapping objects is one of the most important tasks of the image analysis area. This task is connected to the problem of analyzing 2D projections of 3D objects. It is widely used in the industry and biology. Usually, it is difficult to estimate inner contours of the overlapped object so the segmentation methods must relay just on visible parts of particles (see Figure 1). To solve such problems one must estimate the full contour based on visible edge fragments and prior knowledge about the object shape [1].

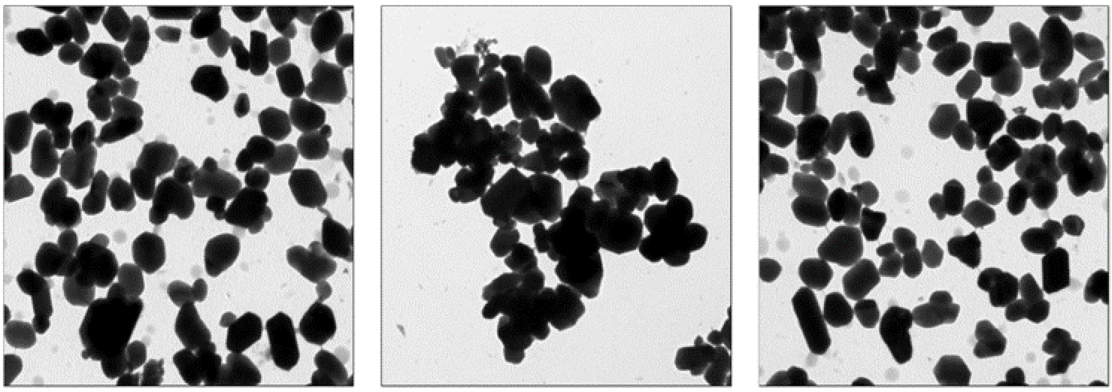

Figure 1. Examples of images from transmission electron microscopy.

This work focuses on segmentation of convex objects. The work continues earlier research where a framework to segment (estimate contours) of partially overlapping nanoparticles was developed [2]. The framework consists of three steps: 1) detecting of concave edge points, 2) grouping of the resulting edge segments to form contour evidence, and 3) estimating the full contours of the objects (see Fig. 2) [3, 4].

In order to be able to estimate full contours of objects with partially observed edges, all edge points or edge segments belonging to the same object need to be grouped. To do this, shape analysis of the resulting object is needed. This can be done by employing a grouping method that defines how likely two edge segments belong to the same object, that is how well the resulting object fits the prior information about the object shapes or contour model.

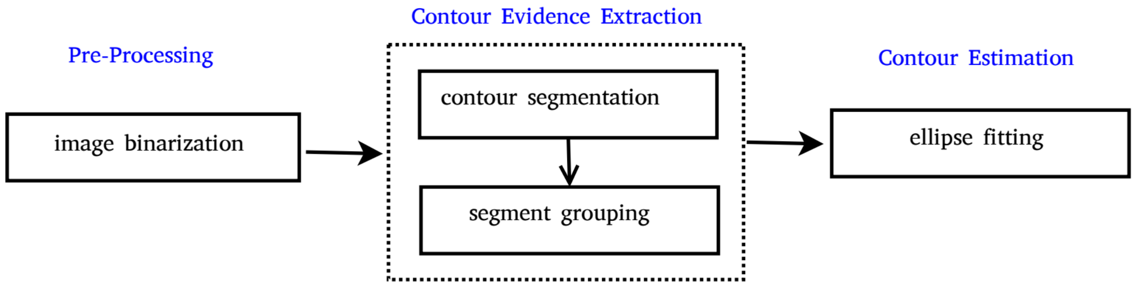

Figure 2. Concave points based framework. [4]

1.2 Objectives and delimitations

The aim of this master's thesis is to develop an efficient grouping strategy that grouping the contour segments which belong to the same object.

The objectives are as follows:

1. Find and develop different grouping methods using the contour model/shape criteria/metrics (e.g. convexity) for objects.
2. Evaluate the grouping methods using the existing framework.
3. Perform parameter optimization and parameter sensitivity analysis for the developed method.

1.3 Structure of the thesis

The rest of thesis has the following structure. Chapter 2 gives a brief overview of an existing contour segmentation methods. This chapter contains a description of seed point extraction methods, concave point-based methods, and edge segment grouping methods. Chapter 3 presents a segmentation framework with a new proposal of a segment grouping method. Chapter 4 contains the information about experiments and validation of the proposed segment grouping method. Chapter 5 discusses the findings and describes goals of the further research. Chapter 6 concludes the thesis and give a brief overview of the problem, the solution, and the results.

2 SEGMENTATION OF OVERLAPPING CONVEX OBJECTS

There are two typical approaches to solve the segmentation of overlapping convex object segmentation problem. The first approach is based on the extraction of seed points [1], special points, that are geometrical centers of overlapping objects. The second approach is based on concave point detection [3]. This method extracts visible boundaries of overlapping objects and detects the special concave points that are corners between overlapping objects.

2.1 Seed point-based methods

A typical framework for segmentation of overlapping object usually consists of the next specific steps [1]:

1. Seed region/point extraction. In this step, there is extracting points or regions of each overlapping object. The seed points usually refer to a geometrical center point of anticipated overlapping objects. The goal of this step is to recognize the number of the individual overlapping objects. The count and position of the detected points in this step are not final and can be improved during the next steps.

2. Contour evidence extraction. The goal of this step is the determination of the counter evidence, the visible parts of the object boundaries, that are used to determinate the hide parts of overlapped objects. The aim of this step to group edge points that belong to each object using information about seed points or seed region from the previous step.

3. Contour estimation. During this step, the full contour of all overlapping objects is estimated based on results of the previous steps.

The schema of the framework is shown in Figure 3.

The seed point extraction is one of the most important steps in overlapping object segmentation. It has a big impact on the accuracy of the final segmentation result. Seed point extraction produces a priori information that is used in counter evidence extraction and contour estimation. The goal of this step is to recognize the number of the individual

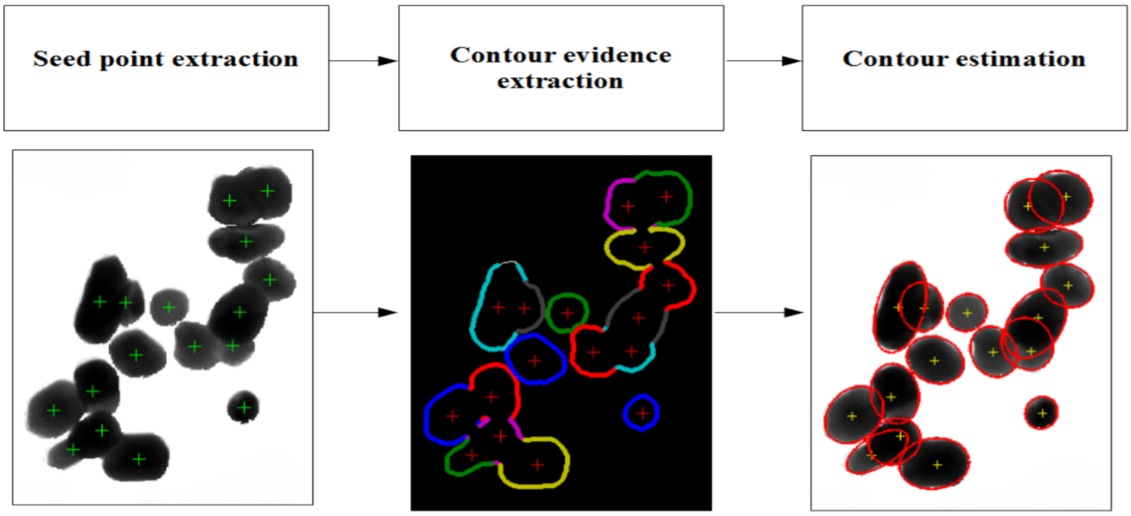

Figure 3. Framework based on seed points. [1]

overlapping objects in the image and correspond them with seed points. The description of the methods is based on [1].

2.1.1 Local Convergence Filters

The description of the local coverage filters is based on [5]. Local Convergence Filters (LCF) [5] are seed points extraction methods, based in gradient convergence and tolerant to noise, illumination variations, and low contrast. This group of filters is one of the most used algorithms to detect seed points. The idea of these algorithms to detect regions in the image with converges of a gradient, what is proper for searching convex shapes. Local Convergence Filters determinate a scale for shape detection and perform detection based on the response within an area relating to such scale, so-called support region. Moreover, LCF have the good robustness to noise due to the essentially added shape prior. The differ between variations of LCF is a shape of a support region. The examples of such filters are following: Coin Filter [6], IRIS filter [6], Adaptive Ring Filter [7], Sliding Band Filter [8]. They are shown in Figure 4.

Coint Filter (CF) [6] is the basic LCF method. The support region in this method has a circular shape. The value of the radius is varied in search for the one which corresponds to maximum convergence, limited by a maximum radius.

IRIS filter (IF) [6] is a modification of the CF filter. The differs from the previous one in that the radius of a support region for every direction changing independently. As a result, this filter is not restricted to circular shapes and can handle more complex objects.

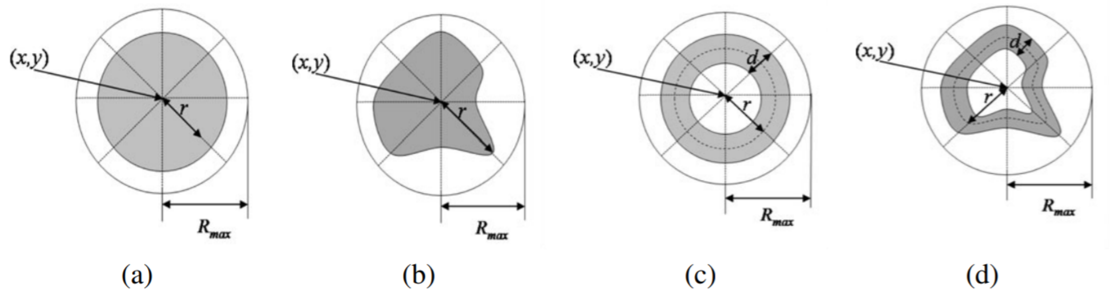

Figure 4. LCF filters: (a) Coin filter; (b) IRIS filter; (c) Adaptive ring filter; (d) Slide band filter. [5]

Adaptive Ring Filter (ARF) [7] define a ring-shaped convergence region. The motivation for that support region is the idea that the convergence of a convex objects is mostly originated at edges, which makes the method more noise-resistant.

Sliding Band Filter (SBF) [8] unites both limited band search of ARF and the shape flexibility of IF. The result of work of SBF looks like the result of IRIS but provides better separating of overlapping objects.

Local Convergence Filters have one common shortcoming. These methods are based just on gradient orientation and ignore gradient magnitude that can lead to big segmentation errors when gradient magnitude is low [5].

2.1.2 Ultimate Erosion for Convex Sets

Ultimate Erosion for Convex Sets (UECS) [9] is an iterative morphological algorithm that extracts the seed regions from overlapping object. UECS is an extension of the Ultimate Erosion (UE) method with a modified stopping criteria [9]. The algorithms consist of two stages. In the first stage, the image is decomposed into disjoint convex sets. This stage is based on the Ultimate Erosion algorithm with a modified stooping criteria that make possible to avoid over-segmentation. The result of this stage is set of disconnected objects. In the second stage, the input segments transform by the Gaussian mixture model on B-splines to fully connected particles. The algorithm contains one key parameter, the threshold for concavity measurement [1]. This parameter depends on data and should be estimated manually. The detailed visualization of the steps of the algorithm is shown in Figure 5.

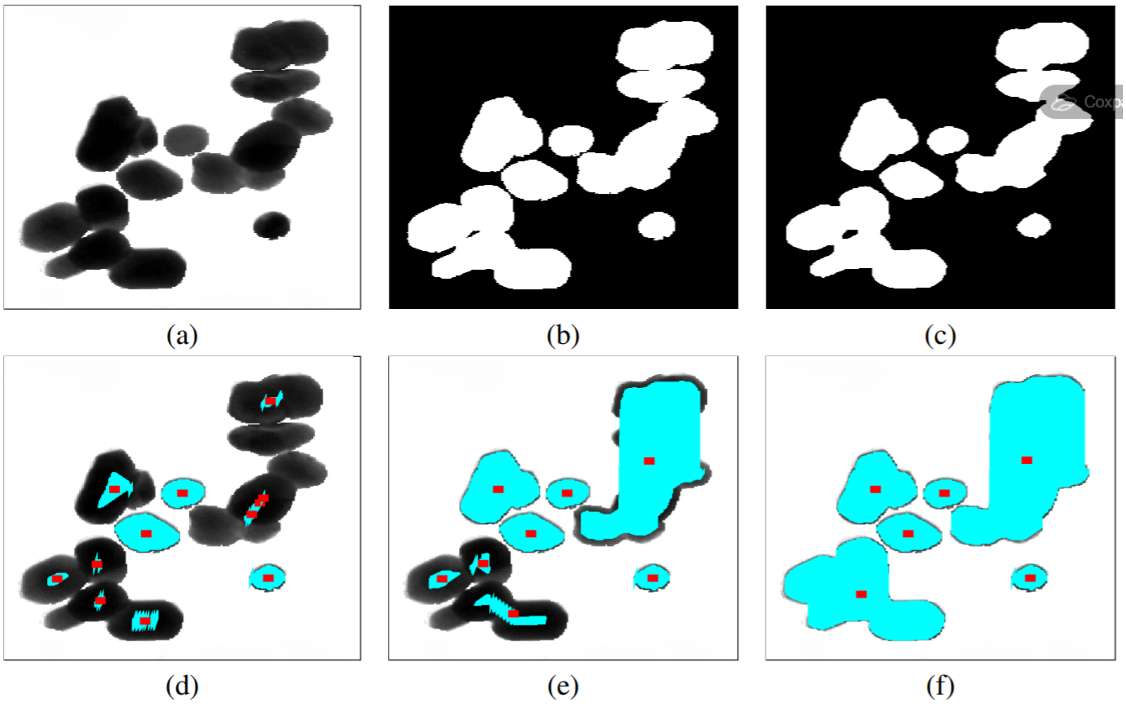

Figure 5. UECS seed point extraction: (a) Original image; (b) Binary image; (c) Binary image after morphological opening; (d) Seed points identified with the threshold 0.1; (e) Seed points identified with the threshold 0.2; (f) Seed points identified with the threshold 0.3. [1]

2.1.3 Distance Transform

Distance Transform (DT) [10] is a simple operator applied to binary images. An output of the transform is a gray level image the same size with an original image, where each original white pixel contains distance to the closest boundary.t. The method is visualized in Figure 6. Algorithm 1 shows the main steps of the method.

Algorithm 1: Distance Transform filter [11].

1. Binarization of the image;
2. Morphological opening of the image;
3. Interactively mapping of the value each pixel to the minimum value of the pixel area by DT;
4. The seed point and region estimation by the predefined threshold;

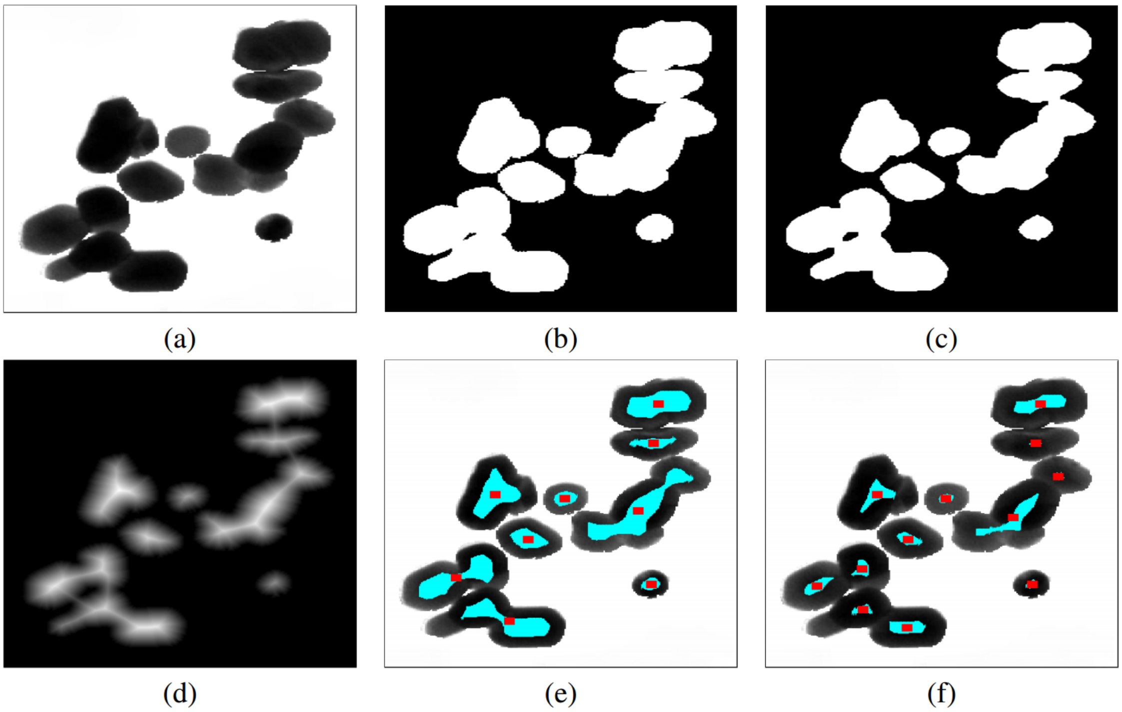

Figure 6. Distance Transform seed point extraction: (a) Original image; (b) Binary image; (c) Binary image after morphological opening; (d) Result of DT; (e) Seed points identified with the threshold 0.6; (f) Seed points identified with the threshold 0.7 [1].

2.1.4 BE-FRS

Fast radial symmetry (FRS) [12] is a wide-used method that transforms the original image to a new representation, highlighting the local radial symmetry of the image gradient. The main idea pf FRS that every edge pixel in the image giving a vote for the plausible radial symmetry at some specific distance from that point. This step relies on two parameters R and T. $R = [R_{min}, R_{max}]$ represents the range of the vote of each point, and T indicates the threshold for the distance between two adjacent seed points. The center of gravity of each vote region is taken as its seed point which is used to mark each object in the image.

The current state-of-art technique in seed point extraction of elliptical objects is Bounded Erosion Fast Radial Symmetry method (BE-FRS) [13], which is a modification of original FRS method. This technique uses two common properties of elliptical objects: convexity and symmetry. It uses a hybrid model consisting of morphological erosion and FRS in a silhouette image to obtain the seed points of each object, while each seed point is used to mark an individual object. Applying bounded erosion [14] before FRS improves the quality of seed point extraction by smoothing the shapes of overlapped objects.

2.2 Concave point-based methods

Another approach for segmentation of overlapping objects is the concave point-based method [3, 15, 16]. The idea of this methods is dividing the contour of the group of overlapping objects to set of segments separated by special points, so-called concave points. The main steps of the methods are shown in Algorithm 2. The visualization of the method is shown in Figure 7.

Algorithm 2: Concave point-based method [3].

1. Image binarization by the Otsu method [17];
2. Dividing the image into the separate Regions of Interest (ROI) , that contains particles or groups of overlapping particles.
3. Extraction of the contour of the whole ROI [18];
4. Splitting the contour to the edges and extracting the concave points;
5. Grouping of segments that are parts of one particle;
6. Ellipse fitting;

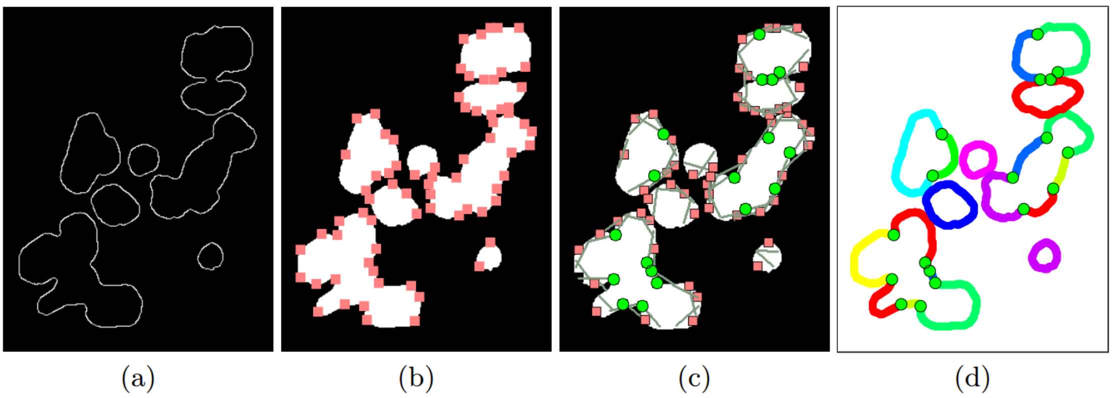

Figure 7. : Contour segmentation: (a) Counter map; (b) Corner detection; (c) Concavity points extraction; (d) Contour segmentation. [3]

2.2.1 Edge segmentation

The following methods perform the segments detection [15]:

1. *Curvature based methods*. These methods relate to an each edge pixel (x_i, y_i) a special value, so-called *curvature value* [3].

   Furthermore, methods select special dominant points which are local extrema for curvature values. There are different strategies to pick over concave points from the set of dominant points. The method purposed by Wen *et al.* [19] select the points by a preset threshold, Zafari *et al.* [3] approach is based choosing points that neighbors do not reside inside the object. Dai *et al.* [20] present the method that calculates the triangle area of a point with neighbors and if the value of the area is positive the method marks the point as a concave point.

2. *Skeleton methods*. This group of methods uses the skeleton and boundary information. The idea of method purposed by Samma *et al.* [21] is the identification of concave points by computing the intersection between the skeleton and contour points. The method purposed by Wang *et al.* [22] detects concave points if the shortest distance to skeletons is more than a preset threshold.

3. *Chord methods*. The main idea of these methods is the identification of concave points as points that have the maximum distance to the concave area chord. There are several methods to estimate concave point area which are described in [23, 24].

4. *Polygon methods*. These groups of methods are a well-known way to represent the contour of the element as a sequence of dominant points. A dominant point is a contour point that does not lie on the line between its' neighbors. A concave point is a dominant point if the line between neighbors dominant points does not pass throw inside the object and the angle between these points is in the range of predefined thresholds [25]. Zhang *et al.* [26] define a dominant point as a concave if $c_{d,i-1}c_{d,i} \times c_{d,i}c_{d,i+1}$ is positive. Sahar *et al.* [15] unite two methods [26] and [25] to avoid any predefined parameters of the concave point detector.

2.2.2 Segment grouping

The next step is grouping such segments which belong to one object, as shown in Figure 8. Zhang *et al.* [26] proposed an average distance deviation criterion (ADD) as a metric for segment grouping. ADD is based on a heuristic that all particles have elliptical shapes. The method unites two groups of segments if the cost for each group is higher than the cost of merged groups.

The naive edge segment grouping algorithm [26] iterates over each pair of contour seg-

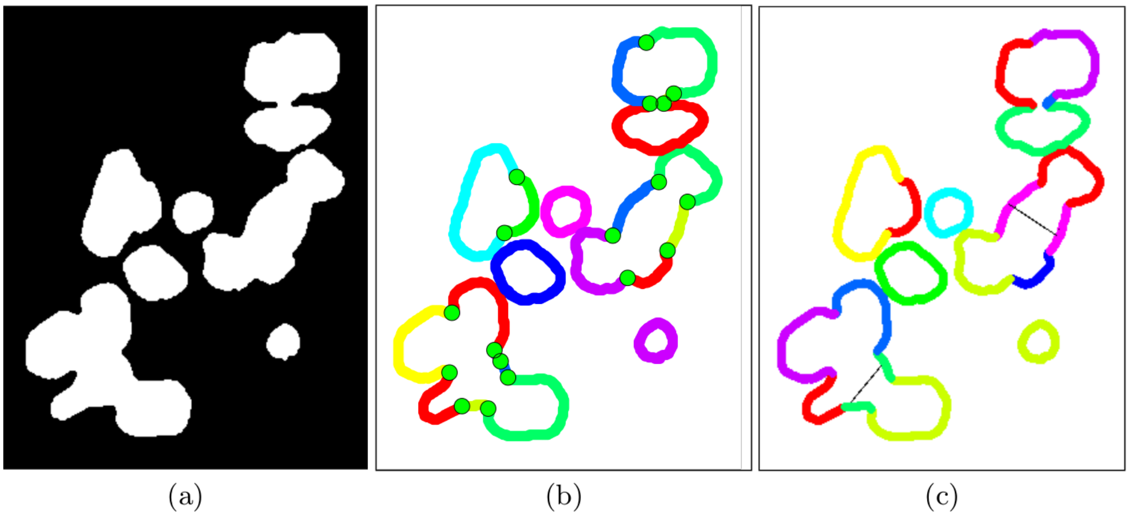

Figure 8. : Contour segmentation: Segment grouping: (a) Original image; (b) Contour segmentation; (c) Segment grouping. [3]

ment, checking if they can be united in one ellipse. However, the number of ways to partition a set of n segments into p groups is very big according to the Stirling number formula:

$$P(n,p) = \frac{1}{p!} \sum_{n_1+n_2+...+n_p=n} \frac{n!}{n_1!...n_p!}, \tag{1}$$

which means that brute-force algorithm is time-consuming.

To deal with the problem of a big amount of permutation researchers use different heuristics. Langlard *et al.* [27] proposed the method that divides the cluster of overlapped elliptical objects to several sub-clusters. To do this, the authors search special split lines with the algorithm that was purposed by Farhan *et al.* [28]. The visualization of the method is shown in Figure 9.

Zafari *et al.* proposed two methods for segment grouping. The first method [4] limits search space by a rule that a distance between centers of segments is less than a predefined threshold value. The second method [4] is based on the branch and bound optimization algorithm. The authors presented a new cost function that uses properties of elliptical objects like ellipticity, convexity, and symmetry.

2.3 Contour Estimation

The last step in the segmentation of overlapped object process is contour estimation. The most use method for contour estimation is an ellipse fitting method. This method can be

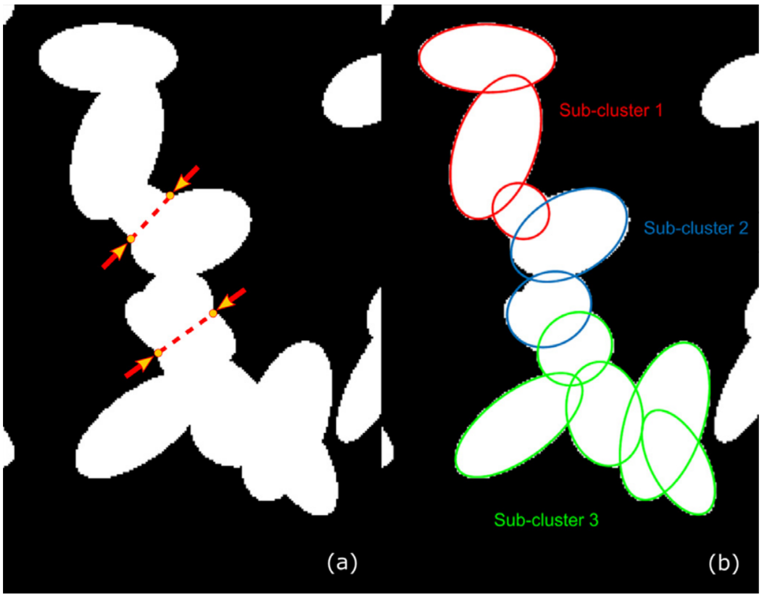

Figure 9. Illustration of cluster decomposition and ellipse fitting: (a) Initial cluster and detected split; (b) Resulting sub-clusters and ellipse fitting. [27].

applied to many tasks in a segmentation of overlapped object area [4, 13, 27, 29]. This method uses the assumption that overlapped objects have an elliptical form. The Ellipse fitting approach is based on minimization the sum of distances between points and an ellipse. The definition of an ellipse equation is:

$$F(\mathbf{a}, (x, y)) = a_0x^2 + a_1xy + a_2y^2 + a_3y + a_4y + a_5 = \mathbf{a}^T\mathbf{x} = 0, \tag{2}$$

where x and y are the input points,$a_0...a_5$ are free parameters. Zhang *et al.* [30] formulate the ellipse fitting problem as a direct minimization of the equation

$$E = \sum_{i=1}^{n} d^2(\mathbf{a}), \tag{3}$$

where $d(\mathbf{a}) = \sum_{i=1}^{N} F(x_i, y_i)$. This equation can be calculated by numerical methods.

The example of ellipse fitting is shown in Figure 10.

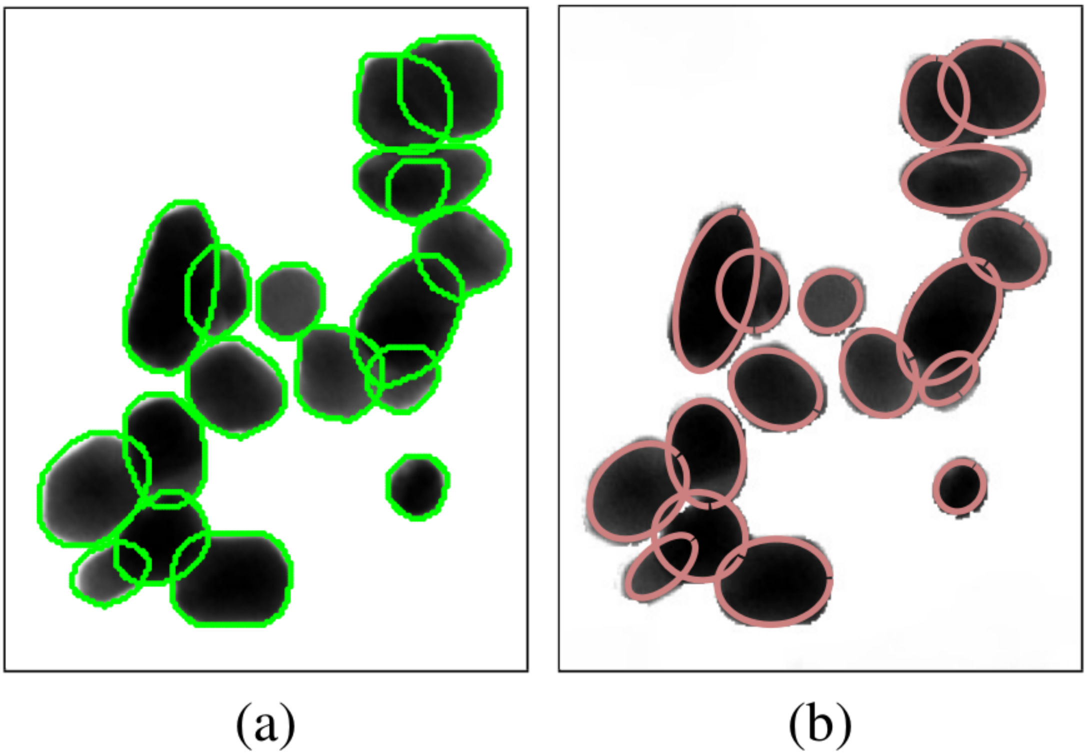

Figure 10. Ellipse fitting method: (a) Manual fitting by a researcher; (b) Ellipse fitting. [13].

3 PROPOSED METHODS

This section contains a description of a proposed method for segmentation of convex overlapped objects. The segmentation method is based on concave points extraction method and has the next pipeline:

1. Image binarization and edge extraction;
2. Concave points detection and segmentation;
3. Segment grouping.

The novelty of the proposed method is in segment grouping stage. However, the other stages are very important for right grouping, so there is a brief explanation of all stages.

3.1 Image binarization and edge extraction

This stage is the most simple but very important part of the segmentation process. It consists of following steps: image binarization, smoothing the binary image, and edge extraction.

1. Binarization of original gray-scale image by Otsu method [17]. This algorithm is a state-of-art algorithm that divides the image into two classes by adaptive threshold. It uses the histogram of the image for threshold searching process. It maximizes "between class variance" of the segmented classes. Otsu proves that Minimizing "within class variance" is same as maximizing "between class variance" of the segmented classes. And maximizing "between class variance" is computationally less expensive than minimizing "within class variance".
2. Smoothing and eroding the binary image. Usually, the original images contain some noise and sharp boundaries, that must be smoothed. To deal with it proposed framework uses morphological erosion [9] with 3-pixel disk structuring element. It smooths the boundaries and reduces a noise that can be detected as small particles.
3. Edge extraction by Canny detector [31]. The Canny operator works in a multistage process. First of all the image is smoothed by Gaussian convolution. Then a simple

2-D first derivative operator (somewhat like the Roberts Cross) is applied to the smoothed image to highlight regions of the image with high first spatial derivatives. Edges give rise to ridges in the gradient magnitude image. The algorithm then tracks along the top of these ridges and sets to zero all pixels that are not actually on the ridge top so as to give a thin line in the output, a process known as non-maximal suppression. The tracking process exhibits hysteresis controlled by two thresholds: $T1$ and $T2$, with $T1 > T2$. Tracking can only begin at a point on a ridge higher than $T1$. Tracking then continues in both directions out from that point until the height of the ridge falls below $T2$. This hysteresis helps to ensure that noisy edges are not broken up into multiple edge fragments.

3.2 Concave points detection and segmentation

For concave point extraction was selected algorithm proposed by X.C. He in [32]. The algorithm was selected because it has some key features. The detector first uses an adaptive local curvature threshold an not the global threshold parameter. Second, the angles of corner candidates are checked in a dynamic region of support that reduces the number of false predicted concave points.

Concave point extraction can produce too many concave points in one small area, especially, in hard cases like an intersection of many objects. The framework uses next filter algorithm, if the distance between concave points is less then some threshold, the points will unite to one point between them (11)

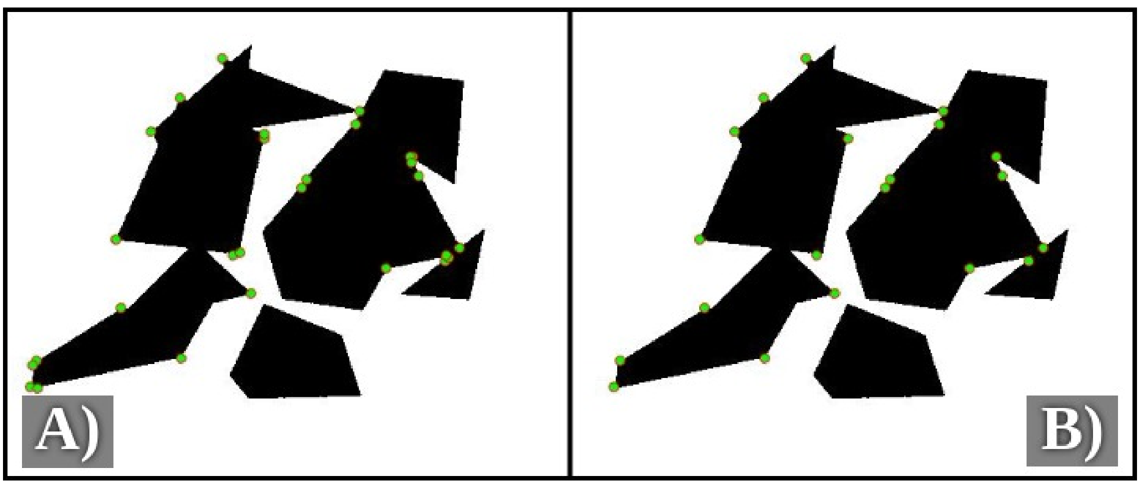

Figure 11. Concave points extraction: a) before reducing, b) after reducing.

After detecting concave points they are projected into edges and divided them into segments, that are bounded by concave points.

3.3 Segment grouping

In this stage, the framework unites segments to one group that belong to one concave object. It consists of two main parts: preprocessing and branch and boundaries algorithm.

3.3.1 Preprocessing

During this step, the framework creates so-called ***grouping matrix***, a special concavity matrix of all segments, that contains 'true' if two segments can be, hypothetically, united to one group, and 'false' if not. To determine the result for two segments are used two heuristics: adjacency and attainability.

Adjacency - is a simple heuristic that based on an idea that two neighbor segment cannot be grouped, because in another way there could not be a concave point between them (see Image 12(a). *Attainability* is more complicated heuristic which validates that two segments can be united to one polygon, and this polygon does not intersect other segments (see Image 12(b).

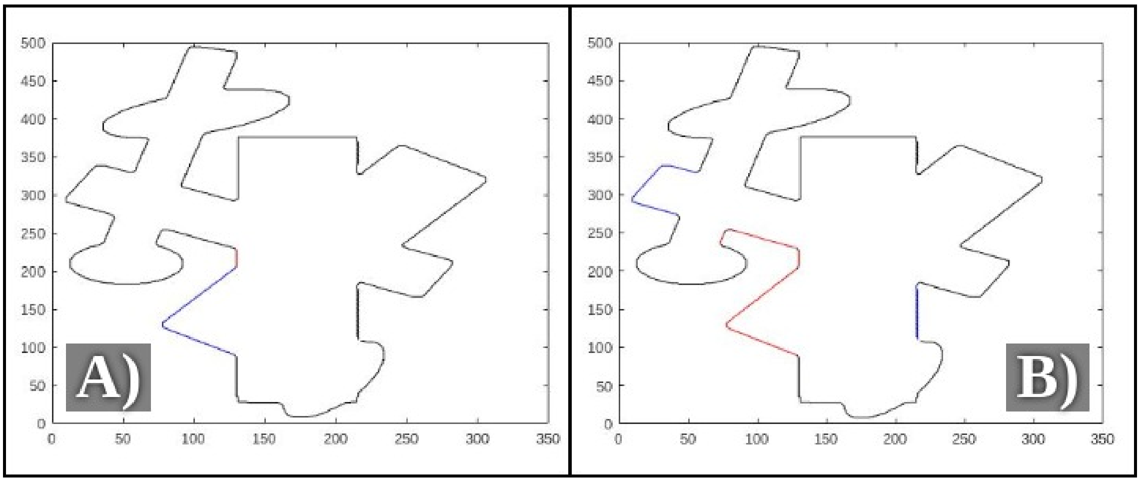

Figure 12. Examples of restricted pairs of segment: a) neighbours, b) a pair that intersects other segments.

3.3.2 Branch and boundaries

The segment grouping task can be represented as a combination problem of optimal separation of N objects to M groups by some criteria. However, it is an NP-hard problem and cannot be solved by brute force algorithms. One of the best and simple methods to solve NP-hard problems is Branch and Boundaries algorithm. BB is efficient for optimization problem since it avoids exhaustive enumeration using the value of the current optimal solution and defining bounds for the function to be optimized.

Firstly, the algorithm creates initial root node with an empty group. Next, the algorithm interactively tries to add new child nodes that extending parent groups with new segments. If the grouping criteria of a child node are bigger then parent node, the algorithm removes that node and do not evaluate parents node for it. That algorithm is well-described in [4]. The groping process of Branch and Boundaries algorithm is shown in Picture 13.

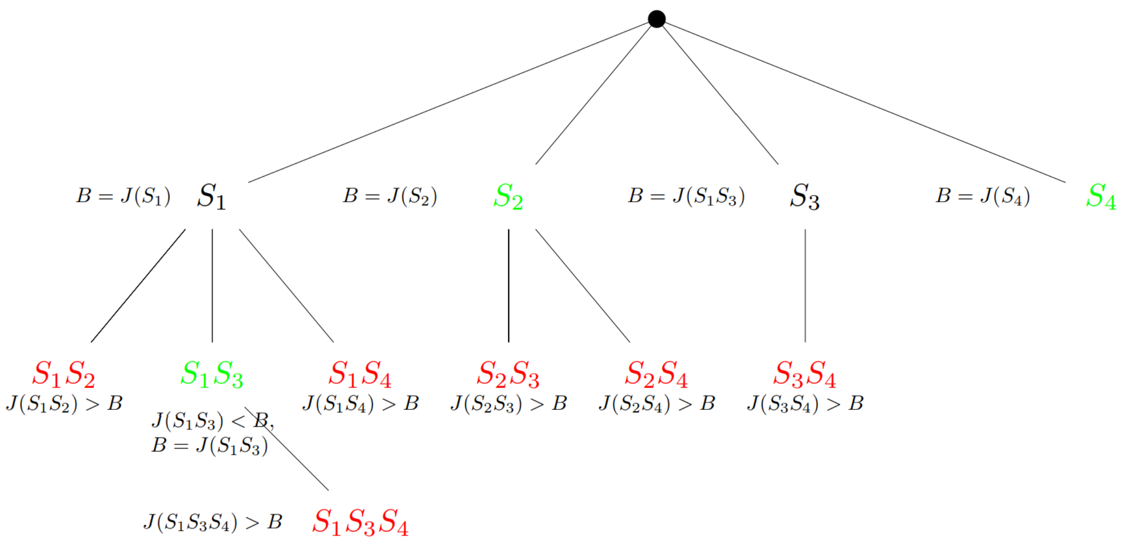

Figure 13. Example of Branch and Boundaries groups tree [4].

3.3.3 Grouping criteria

The key part of BB algorithm is grouping criteria or cost function. This criteria represent the "quality" of grouping and allow the algorithm to compare two segment groups. To estimate the grouping criteria researcher believe that particles have an elliptical form and create cost function based on symmetry, convexity and ellipticity [4].

For this framework was developed another cost function that based on a hypothesis that object has a convex form and are similar to one of the next type of shapes: ellipse, quadrilateral, triangle. The grouping cost calculation algorithm consists of several parts and pseudo-code of it is shown in Algorithm 3.

1. The segments in a group must be adjacent to each other in grouping matrix. If at least one pair of segments is not adjacent, the grouping cost will be infinity.

2. Reordering of segments. It is a necessary operation because the convexity checking function requires segments sorting by clockwise order. The idea of sorting algorithm is following. Each segment from input array must be added to the result array greedy by maximization the total area of a polygon based on result vector. The complexity of the algorithm is $O(n^2)$.

3. Shape fitting. For this framework, the type of figures was limited by ellipse, quadrilateral, and triangle, as the most common types. To determinate the similarity of a segment group to one to the shape are used the following trick. The framework calculates the area of the segment group polygon. Next, the algorithm represents the segments as a cloud of points and build around it the ellipse, quadrilateral, and triangle with minimal area. The similarity of the group is calculated by the next formula:

$$similarity = \frac{min(S_{convexTriangle}, S_{convexEllipse}, S_{convexQuadrilateral})}{S_{segments}} \tag{4}$$

4. Groping cost calculation. The grouping criteria should encourage the big amount of segment in a group. It can be done very simple by dividing the result of previous steps by the count of segments. So the final grouping criteria are:

$$J = \frac{similarity}{size(segments)} \tag{5}$$

Algorithm 3: Cost function.

Data: segments, groupingMatrix

Result: cost

if *one or more pair of segment is not adjacent in grouping matrix* **then**

  return ∞

end

sort segments clockwise;

check segments group convexity;

if *group is not convex* **then**

  return ∞

end

calculate simularity by equation 4;

calculate grouping cost by equation 5;

return *grouping cost*

4 EXPERIMENTS

4.1 Data

For validation of developed framework was used two types of data: real data and synthetic data.

The synthetic dataset (see figure 14-A) consists of images with overlapping objects of different types, as ellipses, triangles, and quadrilateral. All objects uniformly randomly translated, scaled and rotated. All generated images have size 400x500 pixels and maximum rate of overlapping area 40%. There were 40 generated images. Every image was generated with additional information about the shape of figures.

The real dataset contains images of nanoparticles, captured using transmission electron microscopy (see figure 14-B). In result, the dataset includes 11 images of 4008x2672 pixels. Approximately 200 particles were marked manually in each picture by a specialist. The explanations consist of manually drawn contours of the objects.

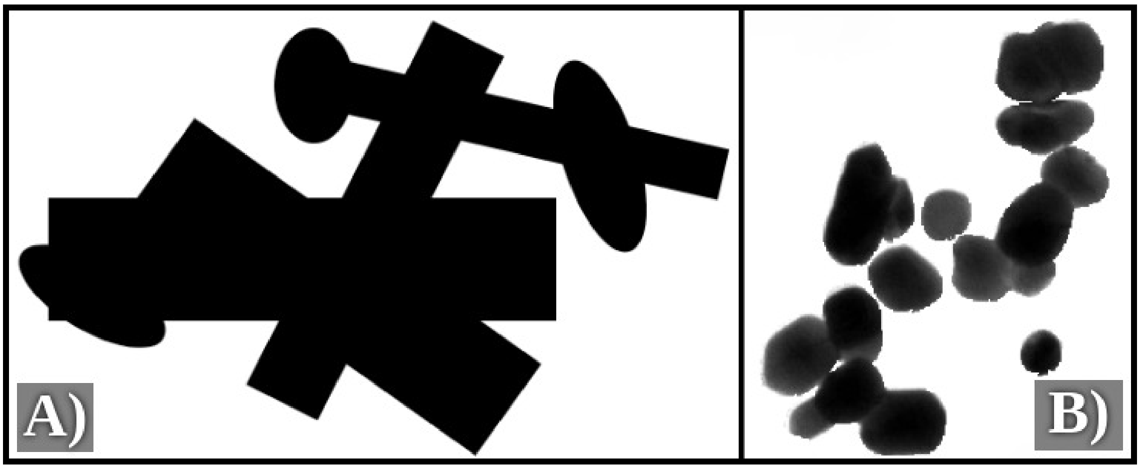

Figure 14. Example of data images: (a) Synthetic data (b) Real data.

4.2 Evaluation criteria

Segment grouping task can be evaluated, as clusterization problem, where the count of predicted classes may differ from ground truth. One of the best metrics of clusterization id Maximum-Match-Measure [33]. For the evaluating criteria were selected next metric.

This measure tries to find an optimal matching between the predicted results and the true classes.

For instance, there is 4 predicted groups and 3 ground truth groups. The matrix of similarity looks like:

$$similarity = \begin{bmatrix} 2 & 7 & 3 \\ 6 & 2 & 3 \\ 3 & 3 & 1 \\ 2 & 1 & 8 \end{bmatrix} \tag{6}$$

Which can be read like this: predicted group 1 has 2 elements from expected group 1, 7 from expected group 2 and 3 from expected group 3, and so on. Maximum-match tries to find the 3x3 matrix with maximum trace (diagonal sum). In this case, it'd be

$$MMmatrix = \begin{bmatrix} 6 & 2 & 3 \\ 2 & 7 & 3 \\ 2 & 1 & 8 \end{bmatrix} \tag{7}$$

The result for this case will be $mm = \frac{6+7+8}{41}$.

4.3 Results

The proposed method was compared with two methods: Branch and Boundaries algorithm based on concave points [4] and the algorithm based on seed points [3]. The result for synthetic and real data was shown in Table 1 and Table 2 respectively.

From the result with the synthetic dataset (Table 1) it can be seen that proposed algorithm shows better result that others it. This is because other methods have assumption that segments can be just elliptical form. The proposed method can is developed to recognize all types of shapes in synthetic dataset.

The result on the nanoparticles dataset (Table 2) show differ results and all methods show low accuracy. It is connected with two reasons. The first reason is that real data contains a lot of of particles, and some have not convex shape. The second reason is that all particles have elliptical form and it the purposed algorithm has not clear advantages over other approaches.

Table 1. The result of validation comparing for synthetic data.

Number	Proposed method	Zafari BB [4]
1	0.85	0.57
2	0.74	0.56
3	0.73	0.5
4	0.75	0.55
5	0.71	0.53
6	0.90	0.52
7	0.81	0.72
8	0.85	0.78
9	0.80	0.55
10	0.71	0.52
11	0.93	0.73
12	0.83	0.65
Mean	0.80	0.65

Table 2. The result of validation comparing for real data.

Number	Proposed method	Zafari BB [4]
1	0.62	0.78
2	0.59	0.74
3	0.57	0.79
4	0.53	0.64
5	0.65	0.59
6	0.77	0.74
7	0.70	0.70
8	0.66	0.72
9	0.65	0.80
10	0.56	0.74
11	0.51	0.72
Mean	0.62	0.71

5 DISCUSSION

5.1 Current study

In this work, the new method for segment grouping was described and proposed new segmentation framework with novel segment grouping technique. This technique outperforms others by 72% with synthetic images and by 34% with real images.

The segmentation framework was implemented with Matlab. The segmentation framework consists of several independent parts that are united in one pipeline. There are following parts: image preprocessing, concave points extraction and segment grouping.

Image preprocessing stage prepares an image for concave points extraction and segmentation. This step includes binarization of an image by Otsu method [17], smoothing and reducing noise. Finally, there is an extraction of edges on the image by Canny detector [31].

Concave point extraction stage searches concave points on the edges of the image. This stage of the framework implements an algorithm that was proposed by X.C. He [32]. This method was selected for several reasons. the first reason is that it is adaptive does not require special global parameters for concave points detection. Secondly this method show good result on images with highly overlapped particles.

Segment grouping stage consists of two parts: preprocessing and branch and boundaries part. During the first part of this stage, the framework for every pair of segments checks two heuristics, if there is at least one segment between these segments, and if the segments are neighbors. If one of heuristics return 'true' result so these segments can be in one segments group. the main part of this stage is Branch and Boundaries algorithm. The grouping task is NP-hard [4] problem and BB one of the methods that can reduce a computational time. The implementation of BB algorithm is based on Zafari proposal in [4] with a new cost function.

The cost function is the most important part of BB algorithm because make possible to estimate a "quality" of grouping. If the cost of a group of segments is less it is better. The cost function is based on some assumptions: similarity to one of the predefined shapes, convexity and that all pairs satisfy the conditions from the heuristic part.

For synthetic data generation was implemented with Matlab a special image generator.

The generator creates images with size 400x500 pixels and maximum 40% shapes overlapping. The generator creates shapes of three types: triangles, quadrilateral, and ellipses. The real data contains 11 images with 4008x1672 pixel size. The images captured by transmission electron microscopy.

As an evaluating criterion was selected Maximum-Match-Measure criteria that search maximal matching between real and predicted results. As a method for comparing with proposed method was selected Branch and Boundaries [4] and Seed Points extractions [3] algorithms.

The result of experiments showed that proposed method outperforms existed solution, especially in cases of images which contains particles of different shapes. On real data, the algorithm shows 72% accuracy and on synthetic data 34%. The low accuracy in real data due the fact that there big a big error of concave points detection.

5.2 Future work

The framework proposed in this paper can be improved by several ways.

The first way is to increase the accuracy of the concave points detection algorithm. It is one of the biggest problems, that significantly reduce the accuracy of the algorithm on the real images because The grouping method is very sensitive for correct concave points detection.

The second way is to extend the list of recognized types of shapes. For instance, it can be a good idea to add hexagon, semicircle, and others. An extended list of shapes can improve the performance of the algorithm on some images.

Finally, it is useful to add some heuristics that reduce the complexity time of the framework. Now, for big ima ges, around 3000x4000, the complexity time can exceed 5 minutes. It can be critical in the real-time application.

6 CONCLUSION

In this work was studied methods for segment grouping and proposed new segment grouping framework. This framework was compared with existed solutions and it was estimated that the new approach is more efficient than others.

A survey of existing segmentation methods was made in order to understand which methods can be used for the task of segmentation and segment grouping. There were estimated that there are two main approaches for segmentation. One approach is based on seed-points detection and another approach is based on concave points extraction. The proposed framework is based on concave points and consists of three stages. It based on heuristics and allows users to group segments of convex objects with ellipse, quadrilateral and triangle form.

The proposed framework was designed and implemented in Matlab. There were performed experiments with real and synthetic data was compared with other segmentation algorithms. The results of the experiments showed that the proposed method is better than other methods and can be used for segmentation of particles with different shapes.

REFERENCES

[1] Sahar Zafari. Segmentation of overlapping convex objects. Master's thesis, Lappeenranta University of Technology, Lappeenranta, Finland, 2014.

[2] Sahar Zafari, Tuomas Eerola, Jouni Sampo, Heikki Kälviäinen, and Heikki Haario. Comparison of concave point detection methods for overlapping convex objects segmentation. In *Scandinavian Conference on Image Analysis (SCIA)*, pages 245–256. Springer, 2017.

[3] Sahar Zafari, Tuomas Eerola, Jouni Sampo, Heikki Kälviäinen, and Heikki Haario. Segmentation of partially overlapping nanoparticles using concave points. In *Advances in Visual Computing*, pages 187–197. Springer, 2015.

[4] Sahar Zafari, Tuomas Eerola, Jouni Sampo, Heikki Kälviäinen, and Heikki Haario. Segmentation of partially overlapping convex objects using branch and bound algorithm. In *ACCV 2016 International Workshops, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part III*, pages 76–90, 2017.

[5] Tiago Esteves, Pedro Quelhas, Ana Maria Mendonça, and Aurélio Campilho. Gradient convergence filters and a phase congruency approach for in vivo cell nuclei detection. *Machine Vision and Applications*, 23(4):623–638, Jul 2012.

[6] H. Kobatake and S. Hashimoto. Convergence index filter for vector fields. *IEEE Transactions on Image Processing*, 8(8):1029–1038, Aug 1999.

[7] Jun Wei, Y. Hagihara, and H. Kobatake. Detection of cancerous tumors on chest x-ray images -candidate detection filter and its evaluation. In *Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348)*, volume 3, pages 397–401 vol.3, 1999.

[8] Carlos S. Pereira, Hugo Fernandes, Ana Maria Mendonça, and Aurélio Campilho. Detection of lung nodule candidates in chest radiographs. In *Proceedings of the 3rd Iberian Conference on Pattern Recognition and Image Analysis, Part II*, IbPRIA '07, pages 170–177, Berlin, Heidelberg, 2007. Springer-Verlag.

[9] C. Park, J. Z. Huang, J. X. Ji, and Y. Ding. Segmentation, inference and classification of partially overlapping nanoparticles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(3):1–1, March 2013.

[10] Azriel Rosenfeld and John Pfaltz. Sequential operations in digital picture processing. *Journal of the ACM*, 13:471–494, 1966.

[11] Dipti Prasad Mukherjee, Amita Pal, S.Eswara Sarma, and D.Dutta Majumder. Bacterial colony counting using distance transform. *International Journal of Bio-Medical Computing*, 38(2):131 – 140, 1995.

[12] G. Loy and A. Zelinsky. Fast radial symmetry for detecting points of interest. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):959–973, Aug 2003.

[13] S. Zafari, T. Eerola, J. Sampo, H. Kälviäinen, and H. Haario. Segmentation of overlapping elliptical objects in silhouette images. *IEEE Transactions on Image Processing*, 24(12):5942–5952, Dec 2015.

[14] R. M. Haralick, X. Zhuang, C. Lin, and J. S. J. Lee. The digital morphological sampling theorem. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(12):2067–2090, Dec 1989.

[15] Sahar Zafari, Tuomas Eerola, Jouni Sampo, Heikki Kälviäinen, and Heikki Haario. Comparison of concave point detection methods for overlapping convex objects segmentation. In *Image Analysis - 20th Scandinavian Conference, SCIA 2017, Tromsø, Norway, June 12-14, 2017, Proceedings, Part II*, pages 245–256, 2017.

[16] Mathieu de Langlard, Hania Al-Saddik, Sophie Charton, Johan Debayle, and Fabrice Lamadie. An efficiency improved recognition algorithm for highly overlapping ellipses: Application to dense bubbly flows. *Pattern Recognition Letters*, 101:88 – 95, 2018.

[17] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, Jan 1979.

[18] H. Samet and M. Tamminen. Efficient component labeling of images of arbitrary dimension represented by linear bintrees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):579–586, Jul 1988.

[19] Quan Wen, Hang Chang, and Bahram Parvin. A delaunay triangulation approach for segmenting clumps of nuclei. In *IEEE Sixth International Conference on Symposium on Biomedical Imaging: From Nano to Macro*, ISBI'09, pages 9–12, Piscataway, NJ, USA, 2009. IEEE Press.

[20] Jinxia Dai, Xiaochun Chen, and Naiqing Chu. Research on the extraction and classification of the concave point from fiber image. In *IEEE 12th International Conference on Signal Processing (ICSP)*, pages 709–712, 2014.

[21] Ali Salem Bin Samma, Abdullah Zawawi Talib, and Rosalina Abdul Salam. Combining boundary and skeleton information for convex and concave points detection. In *IEEE Seventh International Conference onComputer Graphics, Imaging and Visualization (CGIV)*, pages 113–117, 2010.

[22] Weixing Wang and Hao Song. Cell cluster image segmentation on form analysis. In *IEEE Third International Conference on Natural Computation (ICNC)*, volume 4, pages 833–836, 2007.

[23] Muhammad Farhan, Olli Yli-Harja, and Antti Niemistö. A novel method for splitting clumps of convex objects incorporating image intensity and using rectangular window-based concavity point-pair search. *Pattern Recognition*, 46(3):741–751, 2013.

[24] Saravana Kumar, Sim Heng Ong, Surendra Ranganath, Tan Ching Ong, and Fook Tim Chew. A rule-based approach for robust clump splitting. *Pattern Recognition*, 39(6):1088–1098, 2006.

[25] Xiangzhi Bai, Changming Sun, and Fugen Zhou. Splitting touching cells based on concave points and ellipse fitting. *Pattern Recognition*, 42(11):2434 – 2446, 2009.

[26] Wen-Hui Zhang, Xiaoya Jiang, and Yin-Mingzi Liu. A method for recognizing overlapping elliptical bubbles in bubble image. *Pattern Recognition Letters*, 33(12):1543–1548, 2012.

[27] Mathieu de Langlard, Hania Al-Saddik, Sophie Charton, Johan Debayle, and Fabrice Lamadie. An efficiency improved recognition algorithm for highly overlapping ellipses: Application to dense bubbly flows. *Pattern Recognition Letters*, 101:88 – 95, 2018.

[28] Muham Farhan, Olli Yli-Harja, and Antti Niemistö. A novel method for splitting clumps of convex objects incorporating image intensity and using rectangular window-based concavity point-pair search. *Pattern Recognition*, 46(3):741–751, 2013.

[29] Guanghui Zhao, Xingyan Zi, Kaitai Liang, and Junwei Zhou Panyi Yun. A modified segmentation approach for overlapping elliptical objects with various sizes. *Green, Pervasive, and Cloud Computing: 12th International Conference, GPC 2017, Cetara, Italy, May 11-14, 2017, Proceedings*, 10232:222–236, 2017.

[30] Qilong Zhang and Robert Pless. Segmenting multiple familiar objects under mutual occlusion. In *IEEE International Conference on Image Processing (ICIP)*, pages 197–200, 2006.

[31] J Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, June 1986.

[32] X.C. He and N.H.C. Yung. Curvature scale space corner detector with adaptive threshold and dynamic region of support. In *Proceedings of the 17th International Conference on Pattern Recognition*, pages 791–794, Aug 2004.

[33] Silke Wagner and Dorothea Wagner. *Comparing clusterings: an overview*. Universität Karlsruhe, Fakultät für Informatik Karlsruhe, 2007.