

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

39 pages, 19 figures, 2 tables, 2 appendixes.

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 overlapping convex object has many real-world applications, including material analysis and morphological analysis of biological cells. The methods for convex object segmentation usually consist of the following stages: 1) edge detection, 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. This thesis is focused on the segment grouping part. The novel proposed method is based on estimating a shell, which covers all segments of the group and has minimal area. The proposed segment grouping method was compared with current state-of-art 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. 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 improved results on images with shapes of different types.

# CONTENTS

<b>1</b>	<b>INTRODUCTION</b>	<b>6</b>
1.1	Background . . . . .	6
1.2	Objectives and delimitations . . . . .	7
1.3	Structure of the thesis . . . . .	7
<b>2</b>	<b>SEGMENTATION OF OVERLAPPING CONVEX OBJECTS</b>	<b>8</b>
2.1	Seed point-based methods . . . . .	8
2.1.1	Local Convergence Filters . . . . .	9
2.1.2	Ultimate Erosion for Convex Sets . . . . .	10
2.1.3	Distance Transform . . . . .	11
2.1.4	Fast radial symmetry . . . . .	12
2.1.5	Edge to seed point watching . . . . .	13
2.2	Concave point-based methods . . . . .	13
2.2.1	Edge segmentation . . . . .	14
2.2.2	Segment grouping . . . . .	15
2.3	Contour Estimation . . . . .	17
<b>3</b>	<b>PROPOSED FRAMEWORK</b>	<b>19</b>
3.1	Image binarization and edge extraction . . . . .	20
3.1.1	Image binarization . . . . .	20
3.1.2	Smoothing and eroding the binary image . . . . .	20
3.1.3	Edge extraction . . . . .	20
3.2	Concave points detection and segmentation . . . . .	21
3.3	Segment grouping . . . . .	21
3.3.1	Preprocessing . . . . .	21
3.3.2	Branch and Bound algorithm . . . . .	22
3.3.3	Cost function for BB algorithm . . . . .	25
<b>4</b>	<b>EXPERIMENTS</b>	<b>27</b>
4.1	Data . . . . .	27
4.2	Evaluation criteria . . . . .	28
4.3	Results . . . . .	29
<b>5</b>	<b>DISCUSSION</b>	<b>31</b>
5.1	Current study . . . . .	31
5.2	Future work . . . . .	32
<b>6</b>	<b>CONCLUSION</b>	<b>33</b>

**REFERENCES**

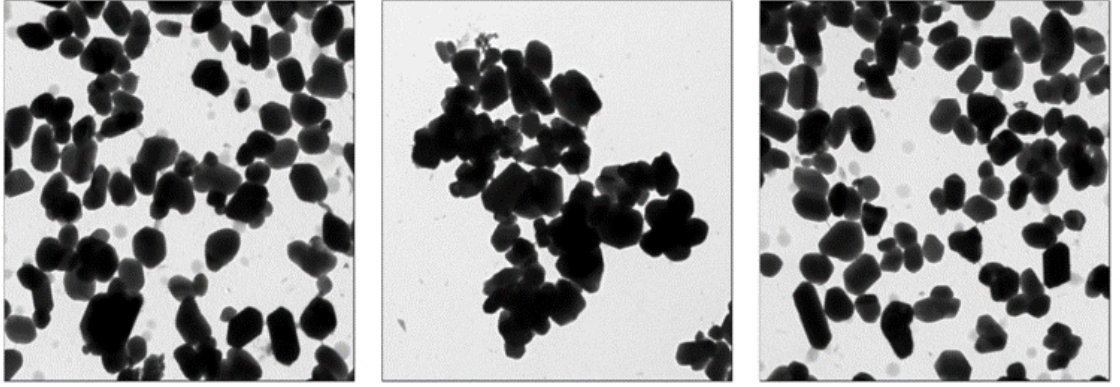
## LIST OF ABBREVIATIONS

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
MMM	Maximum-Match-Measure
SBF	Sliding Band Filter
UE	Ultimate Erosion
UECS	Ultimate Erosion for Convex Sets

# 1 INTRODUCTION

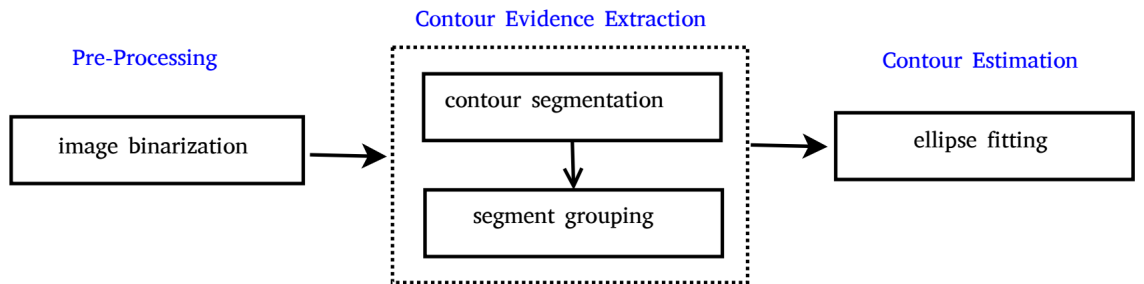
## 1.1 Background

Segmentation or contour estimation of overlapping objects is an image analysis task. 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 rely 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].



**Figure 2.** Concave points based framework. [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.

## 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. Make an overview of existing methods and frameworks for overlapping object segmentation.
2. Propose the new segment grouping method, that improves the performance of segment grouping on images with shapes of different types.
3. Compare the proposed method with existing state-of-art segment grouping method on real and generated data.

## 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

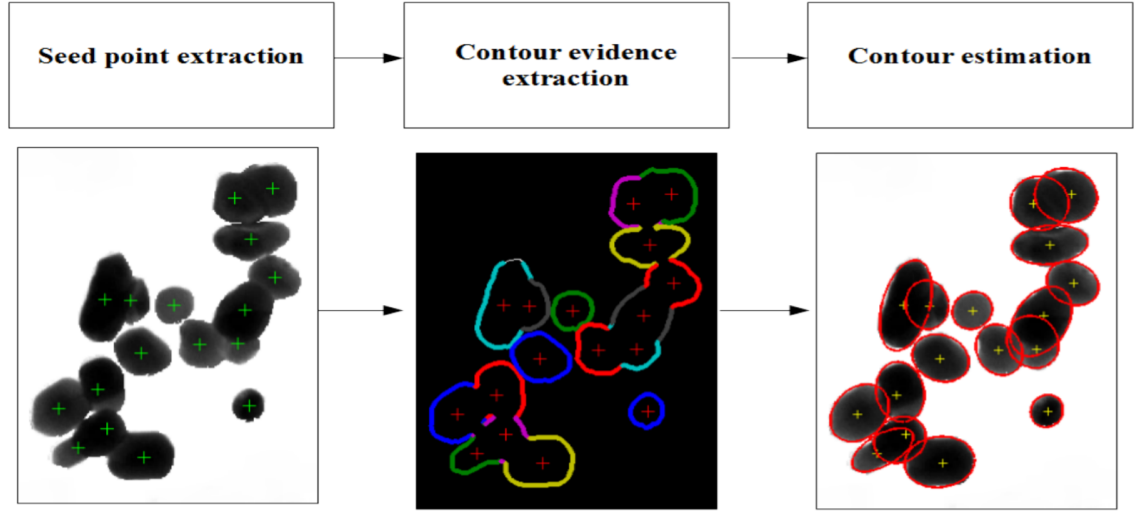
The seed point extraction is one of the most important steps. It has a big impact on the accuracy of the final segmentation result. Seed point extraction produces a priori information that is used in counter evidence extraction and contour estimation. The goal of this step is to recognize the number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

A seed point-based framework for segmentation of overlapping object 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 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. 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.





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

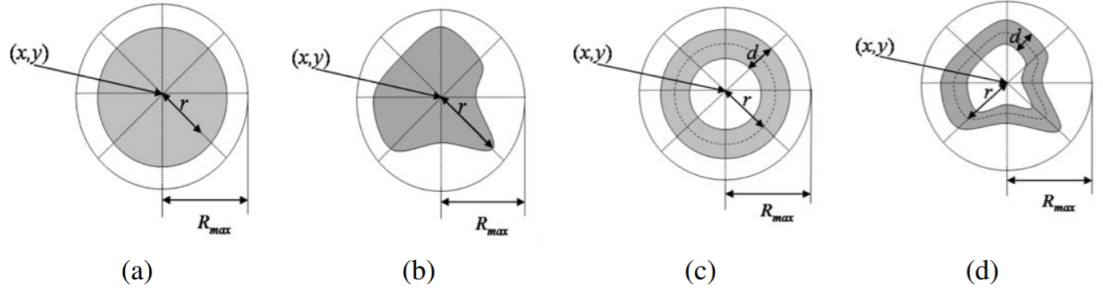
There is a lot of different approaches for seed point extractions, such as Local Convergence Filters [5], Ultimate Erosion for Convex Sets [6], Distant Transform [7], Fast radial symmetry [8].

### 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 [9], IRIS filter [9], Adaptive Ring Filter [10], Sliding Band Filter [11]. They are shown in Figure 4.

**Coint Filter (CF)** [9] 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)** [9] 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,



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

this filter is not restricted to circular shapes and can handle more complex objects.

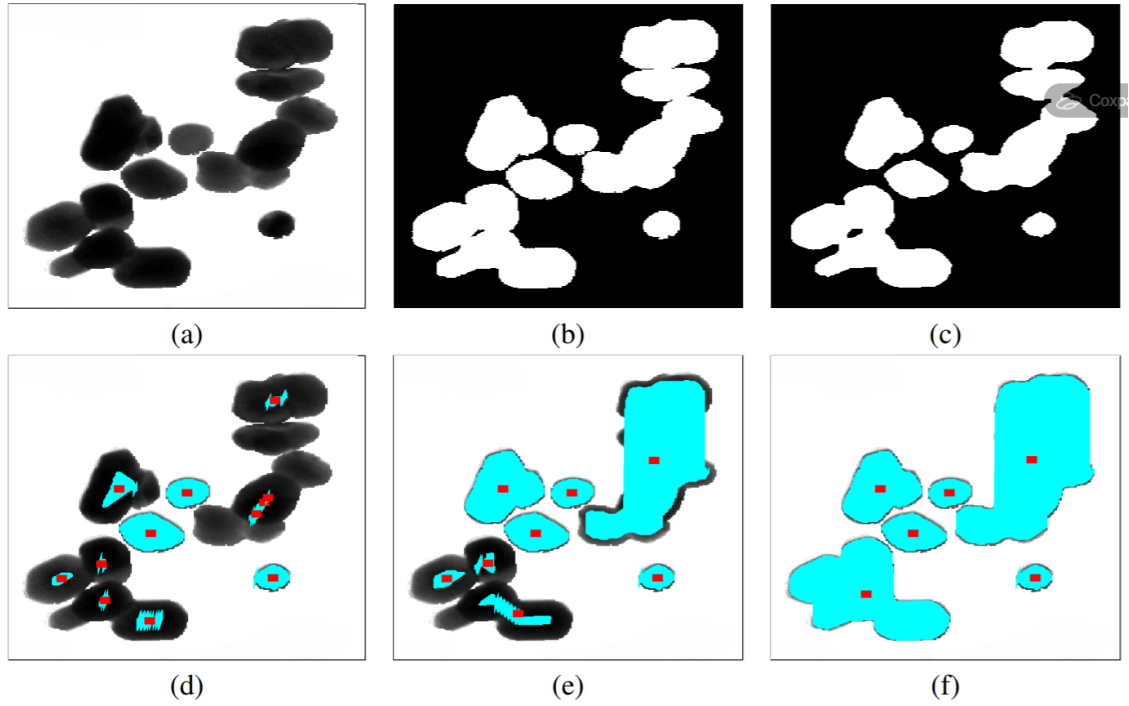
**Adaptive Ring Filter (ARF)** [10] define a ring-shaped convergence region. The motivation for that support region is the idea that the convergence of a convex objects is mostly originated at edges, which makes the method more noise-resistant.

**Sliding Band Filter (SBF)** [11] 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) [6] 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 [6]. 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. 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

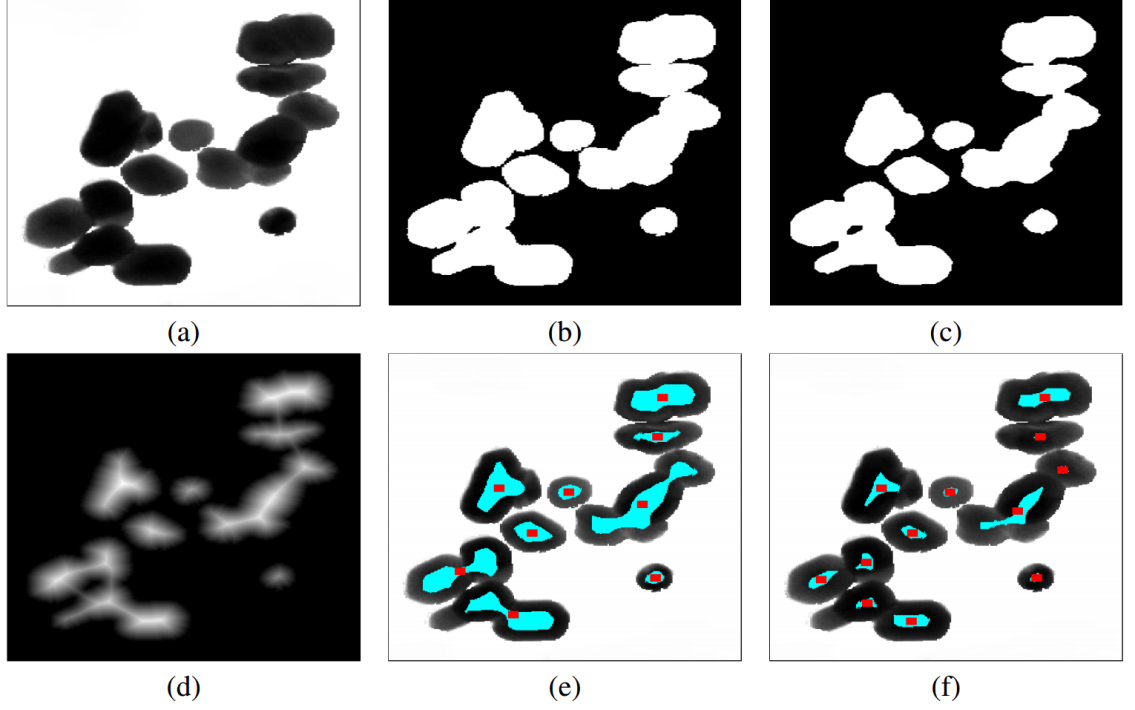
Distance Transform (DT) [7] is a simple operator applied to binary images. An output of the transform is a gray level image the same size with an original image, where each original white pixel contains distance to the closest boundary. The method is visualized in Figure 6. Algorithm 1 shows the main steps of the method.

---

**Algorithm 1:** Distance Transform filter [12].

---

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 Fast radial symmetry

Fast radial symmetry (FRS) [13] 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 of FRS is that every edge pixel in the image gives 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.

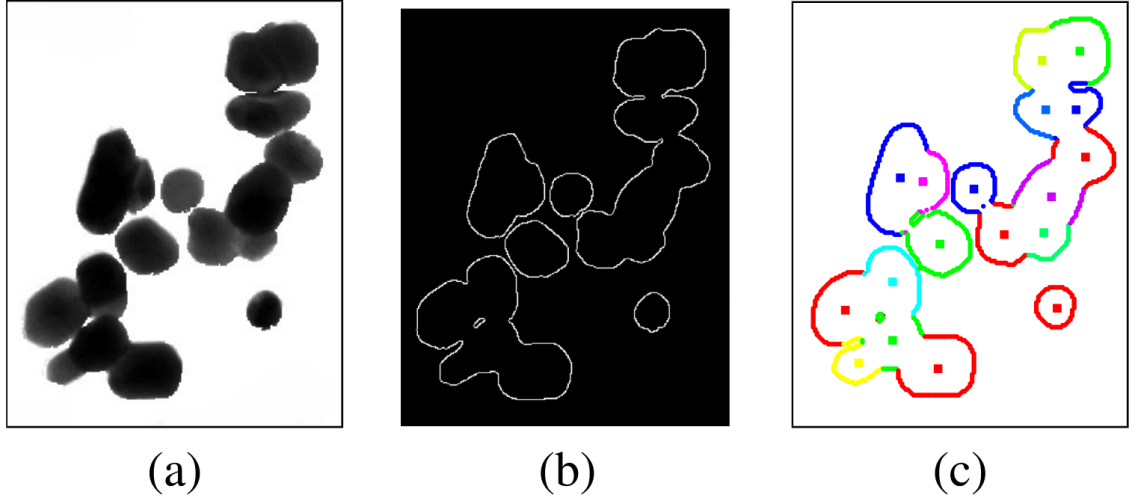
The current state-of-art technique in seed point extraction of elliptical objects is Bounded Erosion Fast Radial Symmetry method (BE-FRS) [8], which is a modification of original FRS method. This technique uses two common properties of elliptical objects: convexity and symmetry. It uses a hybrid model 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.

### 2.1.5 Edge to seed point watching

The edge-to-marker association method [15] attach found seed points with visible edges of overlapped objects. This method is based on calculating a *relevance metric* that combines the divergence metrics and the distance from an edge pixel to a seed point. If  $S = s_1, s_2, \dots, s_n$  is a set of seed points and  $E = e_1, e_2, \dots, e_n$  is a set of edge points, a relevance metric for edge point  $e_i$  and seed point  $s_j$  can be calculated by the following formula:

$$rel(e_i, s_j) = \frac{1 - \lambda}{1 + dist(e_i, s_j)} + \lambda \frac{div(e_i, s_j) + 1}{2}, \quad (1)$$

where  $dist(.,.)$  is Euclidean distance and  $div(.,.)$  is divergence functions respectively. Each function normalized to  $(0,1]$  and summed by  $\lambda$  weight coefficient. In this way, the edge point  $e_i$  is assigned to seed point  $s_j$  with the highest relevance value (see Figure 7).



**Figure 7.** Edge to Seed points association: (a) Original image; (b) Visible contours of particles; (c) Associations between seed points and contours. [8]

## 2.2 Concave point-based methods

Another approach for segmentation of overlapping objects is the concave point-based method [3, 16, 17]. 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

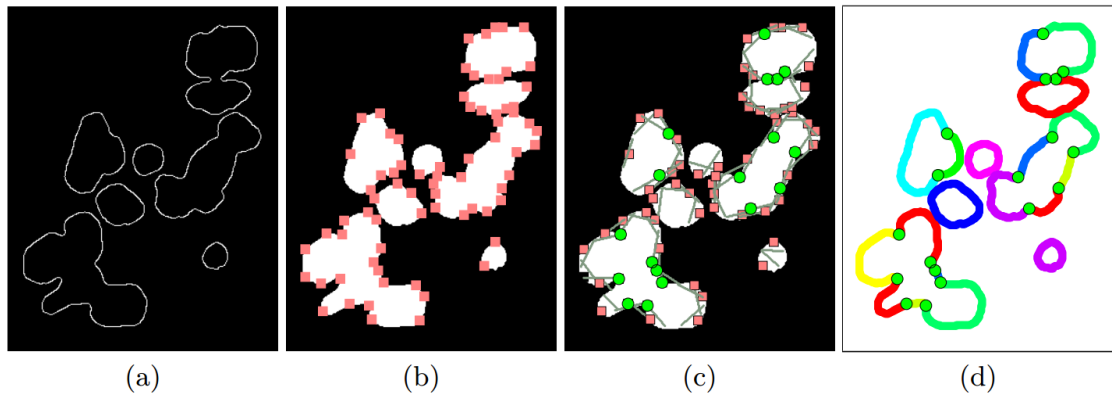
is shown in Figure 8.

---

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

---

1. Image binarization, for example, using Otsu method [18].
  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 [19].
  4. Splitting the contour to the edges by extracting the concave points.
  5. Grouping of segments that are parts of one particle.
  6. Estimating the full contour of the object using, for example, ellipse fitting.
- 



**Figure 8.** : Contour segmentation: (a) Counter map; (b) Corner detection; (c) Concavity points extraction; (d) Contour segmentation. [3]

### 2.2.1 Edge segmentation

Edge segmentation is typically performed by detecting concave points. The following methods exist for concave point detection [16]:

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.* [20] 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.* [21] 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 proposed by Samma *et al.* [22] is the identification of concave points by computing the intersection between the skeleton and contour points. The method proposed by Wang *et al.* [23] 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 [24,25].
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 [26]. Zhang *et al.* [27] define a dominant point as a concave if scalar product between lines, that connect the dominant point with neighbor left and right points, is positive. Sahar *et al.* [16] unite two methods [27] and [26] to avoid any predefined parameters of the concave point detector.

### 2.2.2 Segment grouping

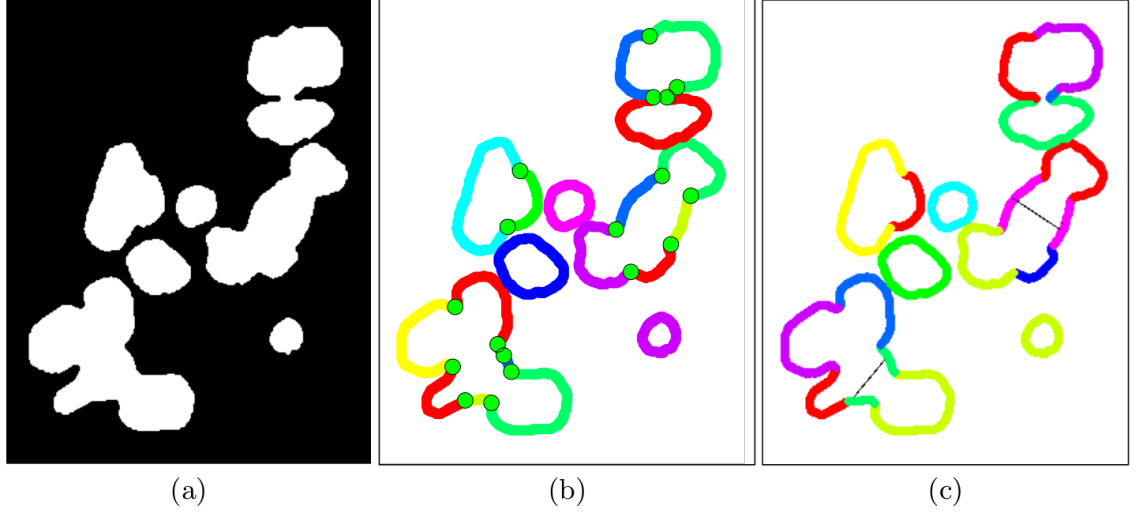
The next step is grouping such segments which belong to one object, as shown in Figure 9. The naive edge segment grouping algorithm [27] iterates over each pair of contour segment, checking if they can be united in one ellipse. However, the number of ways to partition a set of  $n$  segments into  $p$  groups is very big according to the Stirling number formula:

$$P(n, p) = \frac{1}{p!} \sum_{n_1+n_2+\dots+n_p=n} \frac{n!}{n_1! \dots n_p!}, \quad (2)$$

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

Zhang *et al.* [27] 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.



**Figure 9.** : Contour segmentation: Segment grouping: (a) Original image; (b) Contour segmentation; (c) Segment grouping. [3]

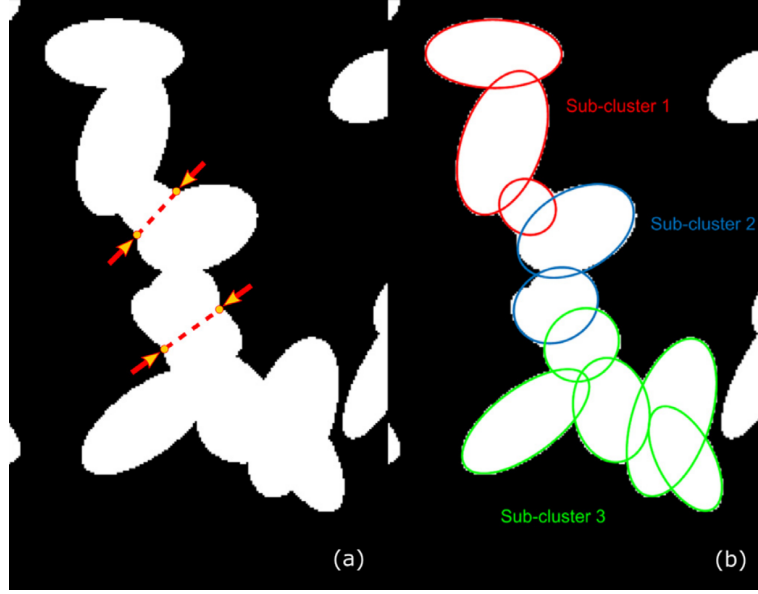
To deal with the problem of a big amount of permutation researchers use different heuristics. Langlard *et al.* [28] proposed the method that divides the cluster of overlapped elliptical objects to several sub-clusters. To do this, the authors search special split lines with the algorithm that was purposed by Farhan *et al.* [29]. The visualization of the method is shown in Figure 10.

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 is a combination of two parts: 1) general part, which represents a convexity of the particle, and 2) specif part that represent the properties of objects with fixed elliptical shape. the specific part consist of two properties: symmetry and ellipticity. The full cost function formula [4] is defined as:

$$J = \alpha J_{convexity} + \beta J_{ellipticity} + \gamma J_{symmetry}, \quad (3)$$

, where  $\alpha, \beta, \gamma$  are weight coefficients.





**Figure 10.** Illustration of cluster decomposition and ellipse fitting: (a) Initial cluster and detected split; (b) Resulting sub-clusters and ellipse fitting. [28].

### 2.3 Contour Estimation

The last step in the segmentation of overlapped object process is contour estimation. The most commonly used method for contour estimation is an ellipse fitting method. This method can be applied to many tasks in a segmentation of overlapped object area [4, 8, 28, 30]. 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:

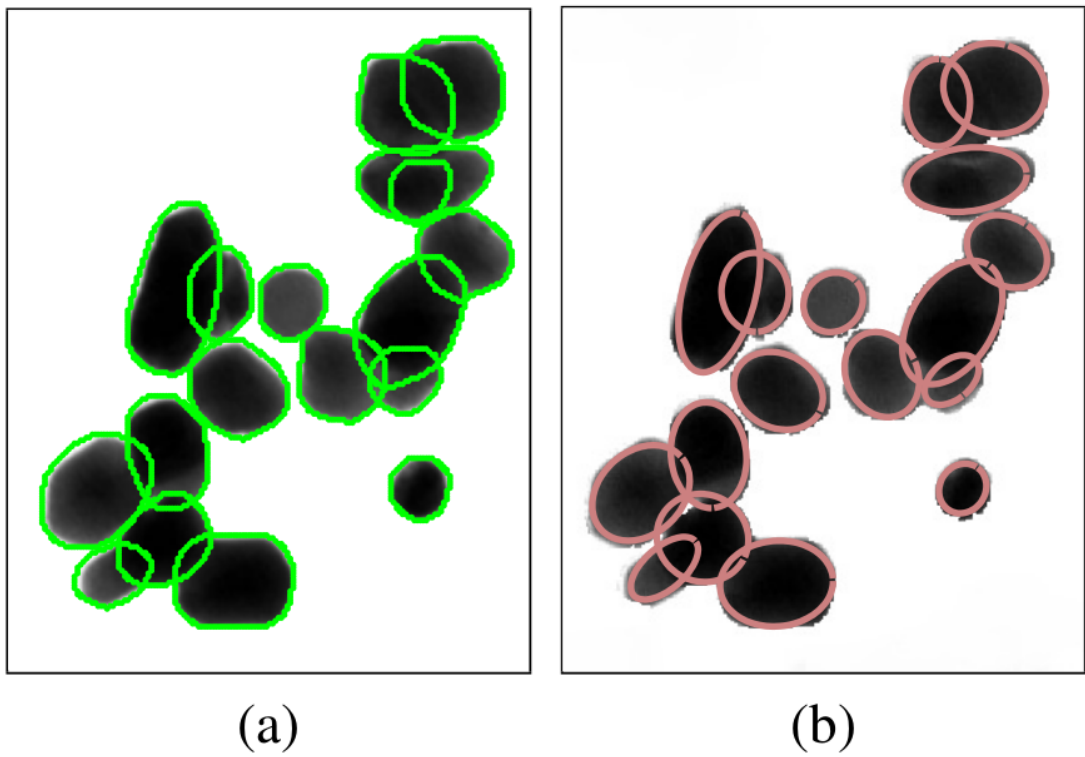
$$F(\mathbf{a}, (x, y)) = a_0x^2 + a_1xy + a_2y^2 + a_3y + a_4y + a_5 = \mathbf{a}^T \mathbf{x} = 0, \quad (4)$$

where  $x$  and  $y$  are the input points,  $a_0 \dots a_5$  are free parameters. Zhang *et al.* [31] formulate the ellipse fitting problem as a direct minimization of the equation

$$E = \sum_{i=1}^n d^2(\mathbf{a}), \quad (5)$$

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 11.

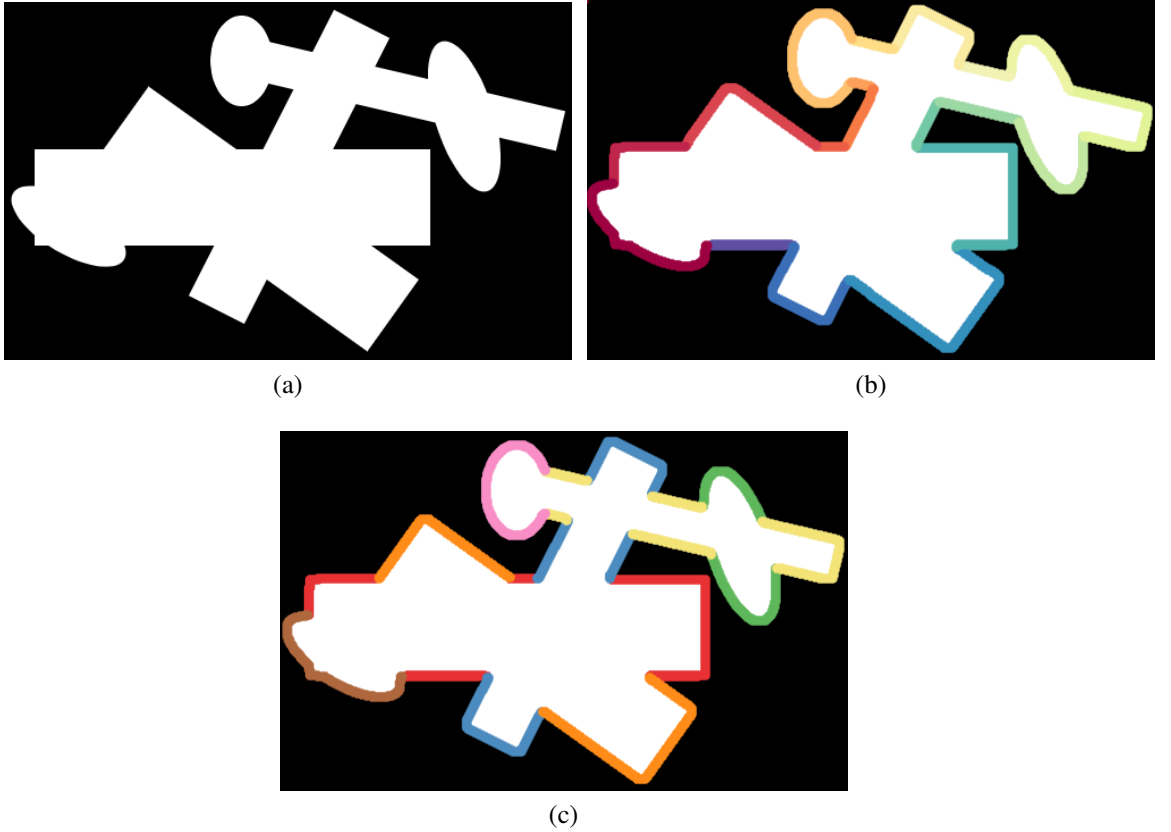


**Figure 11.** Ellipse fitting method: (a) Manual fitting by a researcher; (b) Ellipse fitting. [8].

### 3 PROPOSED FRAMEWORK

This section contains a description of a proposed framework for segmentation of convex overlapped objects. The proposed framework is based on Branch and Boundaries framework [4] with novel segment grouping function. The proposed framework implements the next pipeline (see Figure 12:

1. Image binarization and edge extraction.
2. Concave points detection and contour segmentation.
3. Contour segment grouping.



**Figure 12.** (a) A binary image; (b) a result of contour segmentation; (c) a result of segment grouping.

The novelty of the proposed method is in a segment grouping stage. However, the preceding stages have a significant effect on the success of the grouping.

### **3.1 Image binarization and edge extraction**

The image binarization and edge extraction stage is the most simple, but a very important part of the segmentation process. It consists of three stages: image binarization, smoothing and eroding the binary image, edge extraction.

#### **3.1.1 Image binarization**

Binarization of an original gray-scale image by an Otsu method [18]. This algorithm is a widely used algorithm that divides the image into two classes by an adaptive threshold. It uses the histogram of the image for a threshold searching process. It maximizes a "between class variance" of the segmented classes. Otsu proves that minimizing a "within class variance" is same as maximizing "between class variance" of the segmented classes. Furthermore, maximizing the "between class variance" is computationally less expensive than minimizing "within class variance".

#### **3.1.2 Smoothing and eroding the binary image**

Usually, the original images contain noise and sharp boundaries that must be smoothed. To deal with it the proposed framework applies morphological erosion [6] with a 3-pixel disk structuring element. This smooths the boundaries and reduces a noise that can be detected as small particles.

#### **3.1.3 Edge extraction**

Edge extraction is performed by a Canny detector [32]. The algorithm of this detector has the following steps.

The first step of the algorithm is smoothing. During this step, the image is smoothed and the regions of the picture with high first spatial derivatives are highlighted.

The next step is a searching of gradients. The detector marks the boundaries in the places where the gradient is a local maximum. To find it, the Canny detector uses four filters, which search gradients of different directions. After that, the detector reduces all non-

local maximum gradients.

The final step is an edge tracking. The tracking process is controlled by two thresholds:  $T1$  and  $T2$ , where  $T1 > T2$ . Tracking can only start at a point on an edge greater than  $T1$ . Tracking then proceeds in both ways out from that point until the height of the edge drops below  $T2$ . This hysteresis helps to guarantee that noisy edges are not split up into various edge pieces.

### 3.2 Concave points detection and segmentation

For concave point extraction a modified curvature scale space (CSS) method [33] was selected.

Firstly, the method calculates for each border point, curvature value [33] and selects the points, which are local extrema for curvature values.

The output points, so-called *dominant points*, lie on both convex and concave regions of the object boundaries. To choose just concave points, the method filtrates points with the following criteria. A dominant point is a concave point if the line, connecting left and right neighbors of this point, does not reside inside the object.

Finally, the framework split particles contours into segment edges by concave points.

### 3.3 Segment grouping

Segment grouping method is a part of segmentation framework, that merge segments, which belongs to one object. Segment grouping stage is a modification of the original segment groping stage in Branch and Boundaries framework [4]. Segment grouping could be divided into two steps: 1) a preprocessing step, and 2) a branch and boundaries algorithm step.

#### 3.3.1 Preprocessing

During preprocessing step, the framework creates so-called *grouping matrix*, a special concavity matrix of all segments, that show can two segments be, hypothetically, united

to one group, or not. To estimate this ability, two heuristics are used: adjacency and attainability.

Adjacency is a simple heuristic that is 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 Fig. 13). To determinate that two segments are neighbors, the segment grouping method check that Euclidean distance between ends of the segment is less than 4 pixels.

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 Fig. 14). The segment grouping method performs this test in the way, which is described in Algorithm 3.

---

**Algorithm 3:** Attainability test.

---

**Data:** segments

**Result:** attainability matrix

Represent all segments as sequences of points;

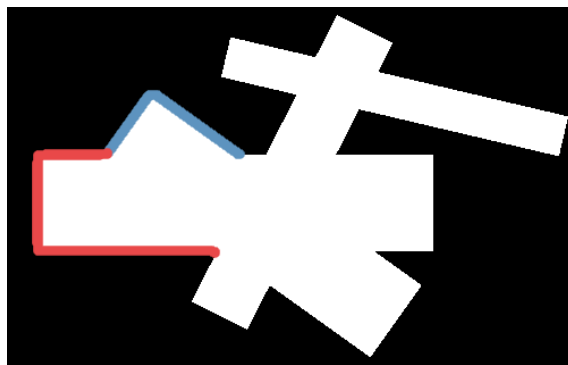
Create shapes for each pair of segments;

**forall** *shape in shapes* **do**

    Test that there are not any points inside shape, except the points which belong  
    to segments that formed the shape;

**end**

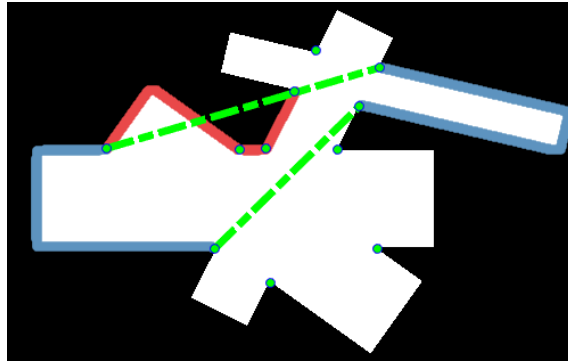
---



**Figure 13.** A pair of neighbour segments (red and blue).

### 3.3.2 Branch and Bound algorithm

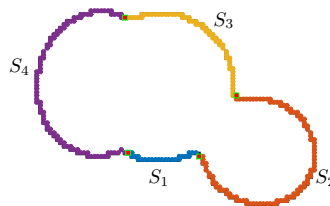
The segment grouping task can be represented as a combination problem of optimal separation of  $N$  objects to  $M$  groups. However, this is an NP-hard problem [27] and cannot



**Figure 14.** A pair of segment, which shape is intersected other segments (blue is a pair of segments, red is intersected segments, green points is concave points, green lines are invisible lines of the pair shape).

be solved by brute force algorithms. One of the most commonly used methods to find a solution for NP-hard problems is a branch and bound (BB) algorithm [27]. BB is efficient for an optimization problem since it avoids exhaustive enumeration using the value of the current optimal solution and defining bounds for the function to be optimized. The BB algorithm is used as a segment grouping algorithm in the original BB framework [4]. In this thesis, a novel cost function for the BB algorithm is proposed.

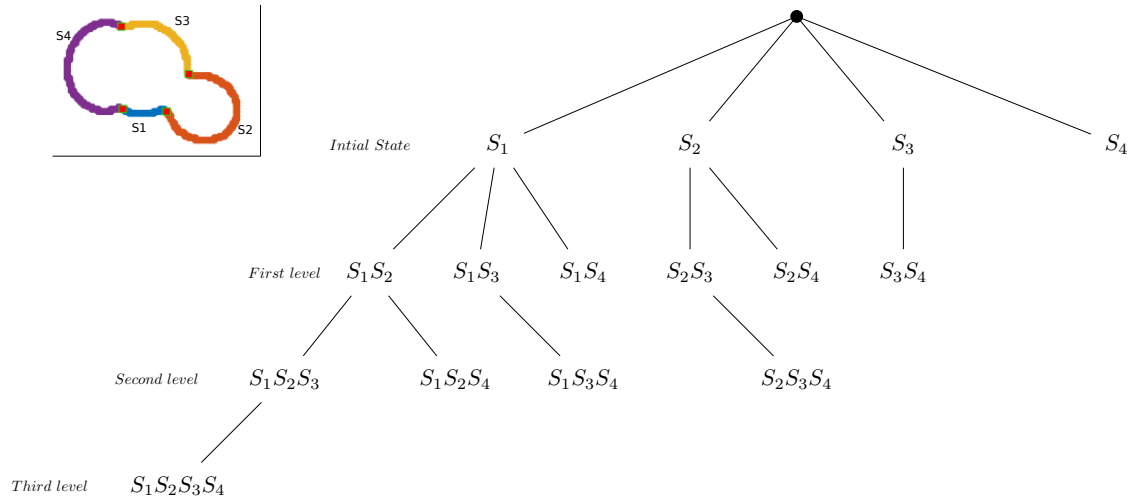
The BB algorithm can be easily demonstrated on the example, that shown in Figure 15. Let  $S = \{S_1, S_2, S_3, S_4\}$  is set of segments and  $J$  is a cost function. To decrease the number of all segment group combinations, let say that grouping candidates  $\{S_1, S_2\}$  and  $\{S_2, S_1\}$  are the candidates because the groups differ if they contain different segments.



**Figure 15.** An instance of contour segments. [4]

In the Figure 16 the search tree with all possible segment groups are shown. The initial state represents grouping candidates which contain just one segment. Each  $n$  level represents all possible grouping candidates that contain  $n$  segments.

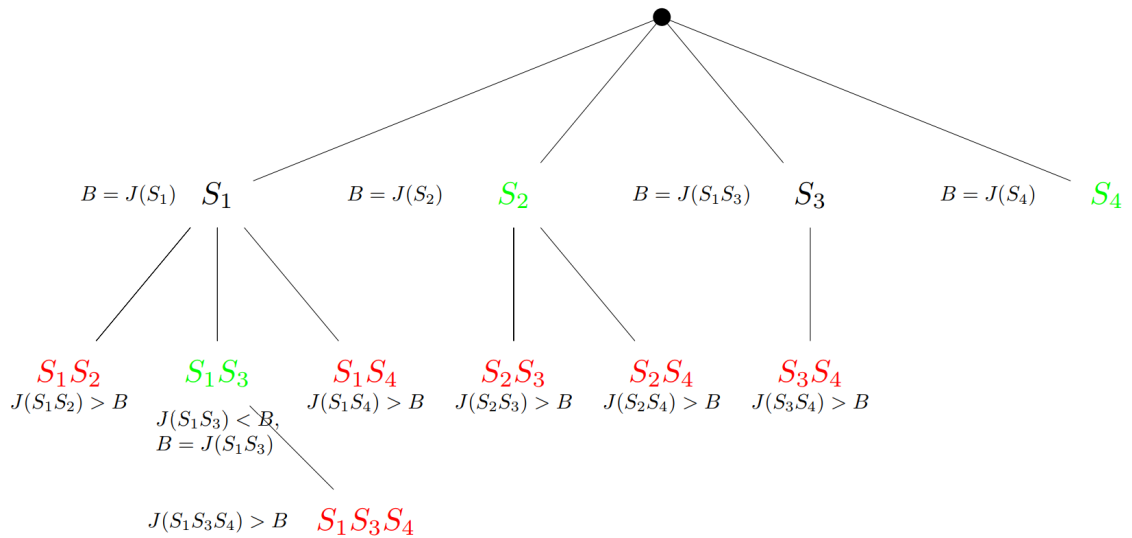
The nodes of the search tree are exploded by depth-first search algorithm [4]. The optimal grouping is a node with the minimal cost function. To reduce the number of grouping



**Figure 16.** The visualization of the full searching tree. [4]

candidates nodes the algorithm uses the following test. Let  $B = J(\{S_1, S_2\})$  is a cost function value for a grouping candidate  $\{s_1\}$ . Let  $\{s_1, s_2\}$  is a children node of  $\{S_1\}$ . If the  $J(\{S_1, S_2\}) > B$  then the BB algorithm will not extend the tree for  $\{s_1, s_2\}$  node, because the child is worse then the parent.

The result of the BB algorithm for  $S$  is shown in Figure 17. Is it can be seen in the Figure the cost value  $J(S_1, S_2) > B$ , so the algorithm is not extend the search tree in this way, but  $J(s_1, s_3) < B$ , so the BB algorithm adds child node  $\{S_1, S_2, S_3\}$  into the searching tree. The result of segment grouping for  $S$  are following groups:  $\{S_1, S_2\}$ ,  $\{S_2\}$ ,  $\{S_4\}$ .



**Figure 17.** An example of a result of a BB algorithm work. [4]



### 3.3.3 Cost function for BB algorithm

The key part of the BB algorithm is a cost function. A cost function represents the quality of grouping and makes possible to compare two segmentation candidates. To estimate the cost function, in [4], it was assumed that particles have an elliptical form and a cost function was created based on symmetry, convexity, and ellipticity.

In this thesis, a novel cost function was developed. The cost function is based on a hypothesis that objects have a convex form and the shape is similar to one of the next type of shapes: ellipse, quadrilateral, or triangle. The grouping cost calculation algorithm consists of several parts and is shown in Algorithm 5.

Firstly, the segments in a grouping candidate must be adjacent to each other in grouping matrix. If at least one pair of segments is not adjacent, the grouping cost is set to infinity.

The second step is a rearrangement of grouping candidate segments. It is a necessary operation because the convexity checking function requires segments sorting by a clockwise order. The algorithm 4 describes the main steps of sorting. The complexity of the sort algorithm is  $O(n^2)$ .

---

**Algorithm 4:** Cost function.

---

**Data:** input segments

**Result:** sorted segments

let  $n$  is a number of segments;

let  $S$  is an empty array with sorted segments;

**for**  $i = 1..n$  **do**

add to  $S$  a segment from the input segments which maximize the area  
of the sorted segments shape;

**end**

**return**  $S$

---

The next step of cost function calculating is 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 ( $A_{segments}$ ) of the segment group polygon. Next, the segments are represented as a cloud of points. For the cloud of points, the algorithm builds an ellipse, a quadrilateral, and a triangle, which cover the points and have a minimal area. To build shapes with the minimal area a suite of minimal bounding [34] objects were used.

To estimate an area of the minimal bounding triangle ( $A_{triangle}$ ) is used the following algorithm: 1) for each edge of the shape was build a triangle that has this edge in the base and covers all points, 2) selected a triangle with minimal area.

The algorithm which determinates a minimal bounding quadrilateral ( $A_{quadrilateral}$ ) the following: 1) sort all edges in counter-clockwise order 2) find the combination of 4 edges, which form a quadrilateral with minimal area.

The minimal area of a covering ellipse ( $A_{ellipse}$ ) is calculated by the way that was described by minimization the sum of distances between points and an ellipse. The paragraph 2.3 has a more detailed explanation.

The similarity ( $S$ ) of the segments group is calculated as:

$$S = \frac{\min(A_{triangle}, A_{ellipse}, A_{quadrilateral})}{A_{segments}}. \quad (6)$$

The final step is a groping cost calculation. The grouping criteria should encourage the big amount of segment in a group. It can be done by simply dividing the result of previous steps by the count of segments ( $N$ ). So the final grouping criteria are:

$$J = \frac{S}{N}. \quad (7)$$

---

**Algorithm 5:** Cost function.

---

**Data:** segments, grouping matrix

**Result:** cost

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

    | **return**  $\infty$

**end**

sort segments clockwise;

check segments group convexity;

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

    | **return**  $\infty$

**end**

calculate similarity by Eq. 6;

calculate grouping cost by Eq. 7;

**return** grouping cost

---

## 4 EXPERIMENTS

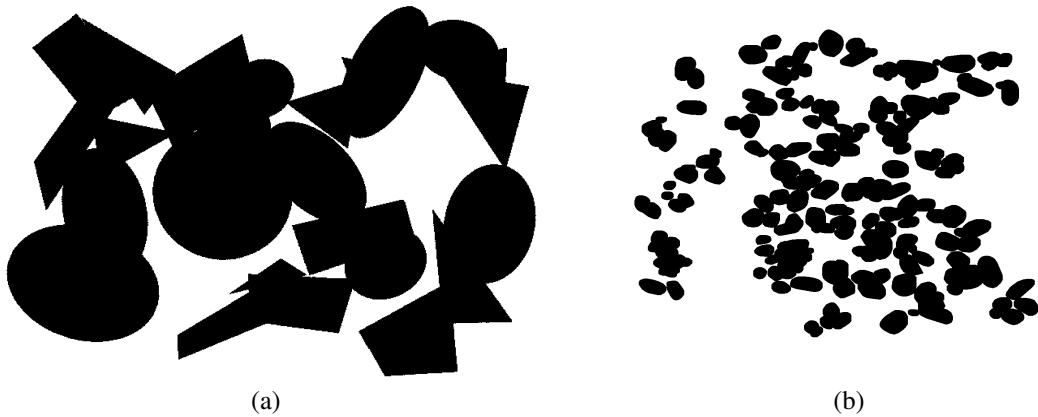
### 4.1 Data

For validation of the proposed method, two types of data were used: real data and synthetic data.

The synthetic dataset (see Figure 18(a)) consists of images with 25 overlapping objects of three different types: ellipses, triangles, and quadrilateral. All objects were uniformly randomly translated, scaled and rotated. All generated images have a size of 700x500 pixels, and a maximum rate of the overlapping area is 40%. The dataset consists of 12 generated images. Every image was generated with a ground truth text file which contains boundaries of the figures.

The real dataset contains images of nanoparticles, captured using transmission electron microscopy (see Figure 18(b)). In result, the dataset contains 11 images of 4008x2672 pixels. Approximately 200 particles were marked manually in each image by a specialist. The ground truth consists of manually drawn contour segments and contour segments groups.

The examples of images is shown in the appendix Figures A1.1 and A2.1.



**Figure 18.** Examples of data: (a) a synthetic image; (b) a real image.

## 4.2 Evaluation criteria

A segment grouping task can be evaluated as a clusterization problem where the count of predicted classes may differ from the ground truth. Ground truth is a result of segment grouping that was gotten from real shapes of particles. One of the best metrics of clusterization is Maximum-Match-Measure (MMM) [35]. This metric was selected for several reasons. Firstly, this metric admits the difference between a number of real groups and predicted. Secondly, this metric is tolerant to the different labels for the same segments in the predicted and ground truth contour segment grouping results. This measure tries to find an optimal matching between the predicted results and the ground truth segment groups.

The algorithm of MMM can be shown by an example. Let the predicted segment groups for the segments in Figure 15 are  $\{S_1, S_3, S_4\}$  and  $\{S_2\}$ , the ground truth segment groups are  $\{S_1, S_3\}, \{S_2\}, \{S_4\}$ .

The similarity matrix are built as:

$$MM_{origin} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}. \quad (8)$$

It means that: the predicted group 1 has 2 elements from the expected group 1, 0 from the expected group 2, and 1 from the expected group 3, and so on. After that, the maximum-match algorithm tries to find the 2x2 matrix with maximum trace (diagonal sum). In the case of the example, it is

$$MM_{reduced} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}. \quad (9)$$

The result of MMM is the sum of diagonal elements divided by the sum of all elements of  $MM_{origin}$ . For this example, it is

$$MMM = \frac{2 + 1}{4} = 0.75. \quad (10)$$

### 4.3 Results

The proposed segment grouping method was compared with original branch and boundaries segment grouping method proposed in [4]. The results for synthetic and real data are shown in Tables 1 and 2.

**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
<i>Mean</i>	0.80	0.65

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

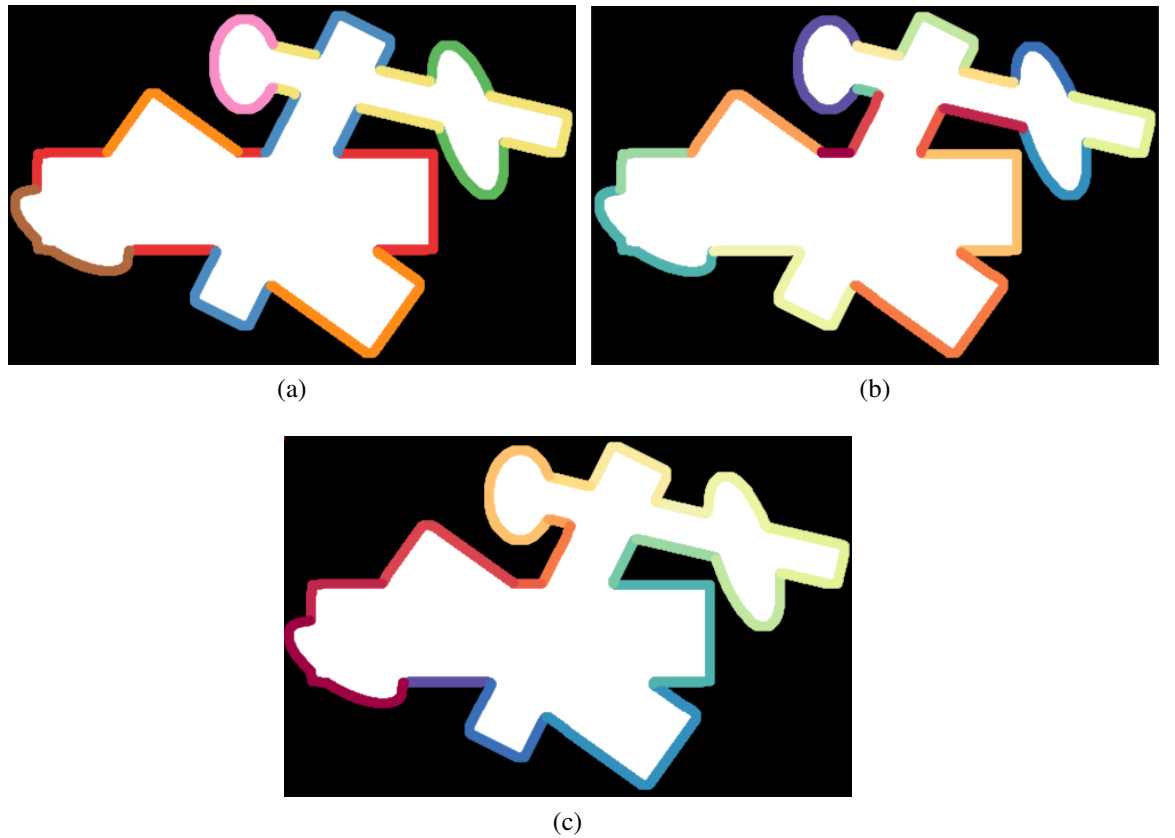
Number	Proposed method	Zafari BB [4]
1	0.62	0.78
2	0.59	0.74
3	0.57	0.79
4	0.53	0.64
5	0.65	0.59
6	0.77	0.74
7	0.70	0.70
8	0.66	0.72
9	0.65	0.80
10	0.56	0.74
11	0.51	0.72
<i>Mean</i>	0.62	0.71

From the result with the synthetic dataset (Table 1) it can be seen that the proposed algorithm shows the best results. This is because the original BB method have an assumption

that objects have approximately elliptical shape. The proposed method is developed to recognize all types of shapes in the synthetic dataset.

The results on the nanoparticles dataset (Table 2) show different results. The proposed algorithm worse than original BB algorithm. It is connected with two reasons. The first reason is that the real data contain a lot of of particles, and some have not convex shape, what is a critical for proposed algorithm. The second reason is that all particles are predominantly symmetrical and have elliptical form. The proposed algorithm does not take into account the symmetry of objects in contrast to the original algorithm.

The result of segment grouping of the proposed method and the original BB algorithm is shown in Figure 19.



**Figure 19.** (a) A result of segmentation of the proposed segment grouping method; (b) a result of segmentation of the original BB method [4]; (c) a result of segmentation.

## 5 DISCUSSION

### 5.1 Current study

In this work, a new segmentation framework with a novel segment grouping method. The segmentation framework consists of the following parts: image preprocessing, concave points detection and segmentation, and segment grouping.

The image preprocessing stage prepares an image for concave points extraction and segmentation. This step includes binarization of an image by an Otsu method [18], smoothing and reducing noise. Finally, there is an extraction of edges on the image by a Canny detector [32].

the concave points detection and segmentation stage are performed in two steps. Firstly, the framework extracts concave points by CSS algorithm [33]. Furthermore, the framework split particles contours into segment edges by concave points.

The segment grouping stage consists of two parts: a preprocessing and a branch and bound optimization. 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 to test the segments can be in one segment group. The main part of this stage is the Branch and Bound algorithm. The grouping task is an NP-hard [4] problem and BB is one of the methods that can reduce the computational time. The implementation of the BB algorithm was based on [4] with the proposed cost function.

A cost function is the most important part of the BB algorithm because it makes possible to estimate the quality of grouping. The cost function is based on the following assumptions: similarity to one of the predefined shapes, convexity.

The synthetic dataset consists of 12 images. Each image contains 25 overlapping objects of three different types: ellipses, triangles, and quadrilateral. The real dataset consists of 11 images of nanoparticles, captured using transmission electron microscopy.

As an evaluating criterion Maximum-Match-Measure criteria were selected. This criteria searches maximal matching between ground truth and predicted segment grouping results and is tolerant to a different number of predicted and ground truth segment groups. The proposed method was compared with the original Branch and Bound [4] algorithm.

The result of experiments showed that proposed method outperforms existed solution on the synthetic data, but shows the worse result with the real data.

## **5.2 Future work**

The segment grouping function proposed in thesis paper can be improved by several ways.

The first way is to add symmetry calculation in the cost function. It is connected with an observation that real particles usually have a symmetrical shape.

The second way is to extend the list of recognized types of shapes. For instance, it can be a good idea, for example, to add a hexagon, a semicircle, etc. An extended list of shapes can improve the performance of the algorithm.



## 6 CONCLUSION

In this work methods for segment grouping were studied and a new segment grouping framework was proposed. A survey of existing segmentation methods was made in order to understand which methods can be used for the tasks of segmentation and segment grouping. There are two main approaches for segmentation. The one is based on seed-points detection and the other is based on concave points extraction. The proposed framework is based on the original BB algorithm [4] with a proposed cost function. This cost function based on shapes fitting and allows users to group segments of convex objects with ellipse, quadrilateral and triangle forms.

The proposed cost function was compared with original BB cost function [4] on the real and the synthetic data. The results of the experiments showed that the proposed cost function outperforms the original cost function on the synthetic data.

## REFERENCES

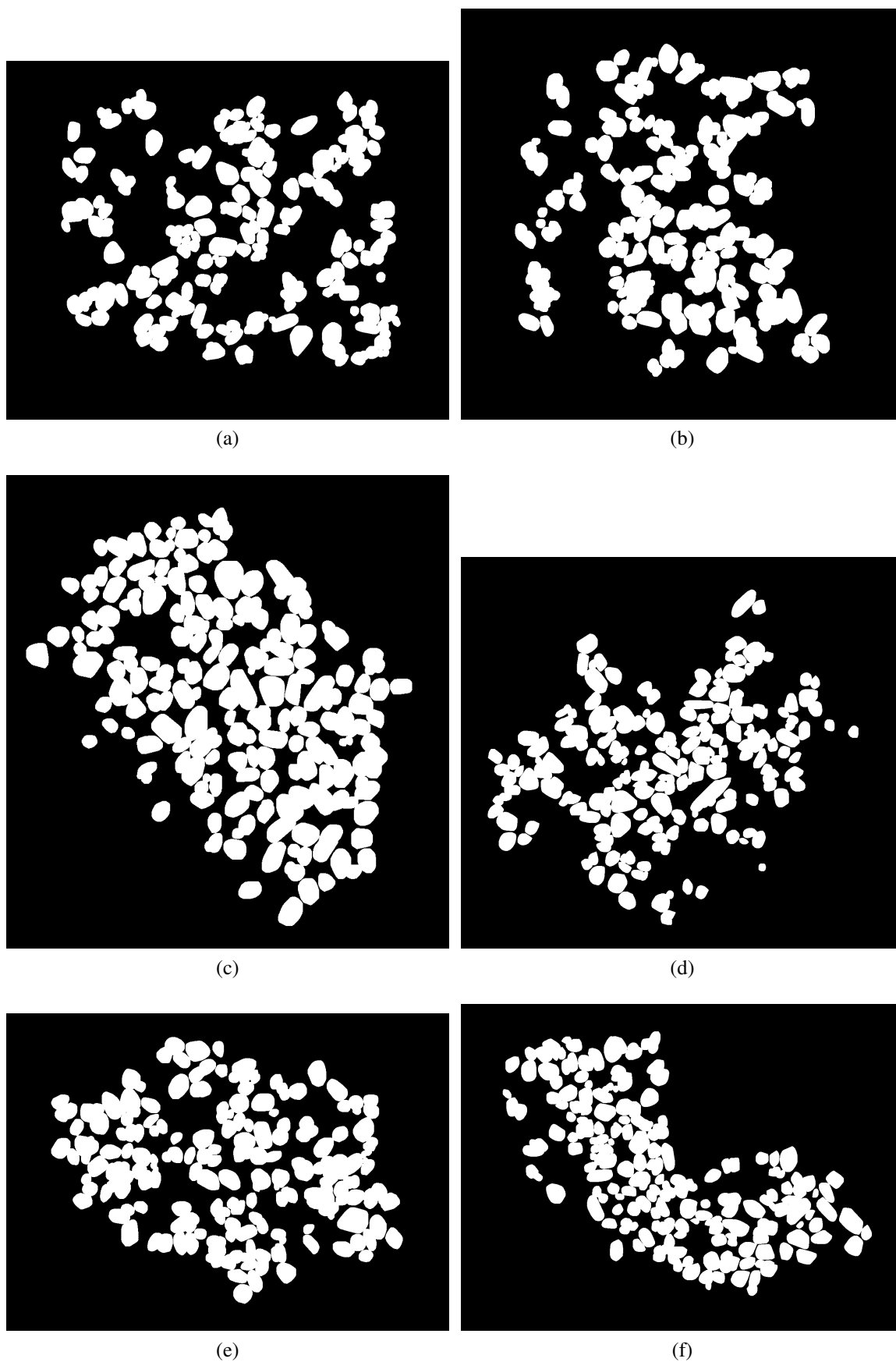
- [1] Sahar Zafari. Segmentation of overlapping convex objects. Master's thesis, Lappeenranta University of Technology, Lappeenranta, Finland, 2014.
- [2] Sahar Zafari, Tuomas Eerola, Jouni Sampo, Heikki Kälviäinen, and Heikki Haario. Comparison of concave point detection methods for overlapping convex 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] 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.
- [7] Azriel Rosenfeld and John Pfaltz. Sequential operations in digital picture processing. *Journal of the ACM*, 13:471–494, 1966.
- [8] 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.
- [9] H. Kobatake and S. Hashimoto. Convergence index filter for vector fields. *IEEE Transactions on Image Processing*, 8(8):1029–1038, Aug 1999.
- [10] Jun Wei, Y. Hagihara, and H. Kobatake. Detection of cancerous tumors on chest x-ray images -candidate detection filter and its evaluation. In *Proceedings International Conference on Image Processing*, volume 3, pages 397–401 vol.3, 1999.
- [11] 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*, pages 170–177, Berlin, Heidelberg, 2007. Springer-Verlag.
- [12] 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.
  - [13] 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.
  - [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] Chiwoo Park, Jianhua Z Huang, Jim X Ji, and Yu Ding. Segmentation, inference and classification of partially overlapping nanoparticles. *IEEE transactions on pattern analysis and machine intelligence*, 35(3):1–1, 2013.
  - [16] Sahar Zafari, Tuomas Eerola, Jouni Sampo, Heikki Kälviäinen, and Heikki Haario. Comparison of concave point detection methods for overlapping convex objects segmentation. In *The 20th Scandinavian Conference on Image Analysis, SCIA 2017, Tromsø, Norway*, pages 245–256, 2017.
  - [17] 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.
  - [18] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, Jan 1979.
  - [19] 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.
  - [20] 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.
  - [21] 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.

- [22] 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 on Computer Graphics, Imaging and Visualization (CGIV)*, pages 113–117, 2010.
- [23] 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.
- [24] 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.
- [25] 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.
- [26] Xiangzhi Bai, Changming Sun, and Fugen Zhou. Splitting touching cells based on concave points and ellipse fitting. *Pattern Recognition*, 42(11):2434 – 2446, 2009.
- [27] 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.
- [28] 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.
- [29] 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.
- [30] Guanghui Zhao, Xingyan Zi, Kaitai Liang, and Junwei Zhou Panyi Yun. A modified segmentation approach for overlapping elliptical objects with various sizes. *12th International Conference on Green, Pervasive, and Cloud Computing, GPC 2017*, 10232:222–236, 2017.
- [31] Qilong Zhang and Robert Pless. Segmenting multiple familiar objects under mutual occlusion. In *IEEE International Conference on Image Processing (ICIP)*, pages 197–200, 2006.

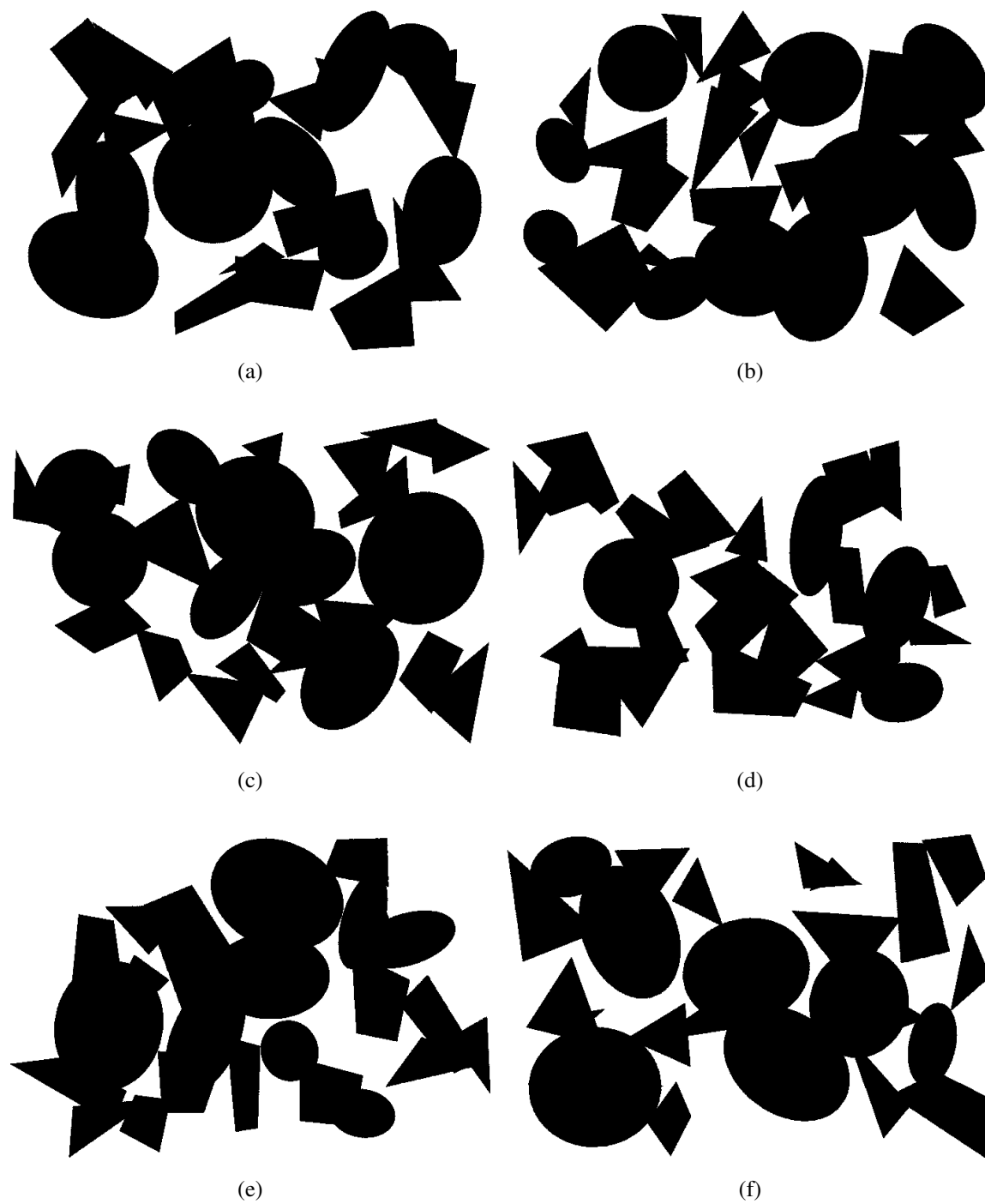
- [32] J Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, June 1986.
- [33] X.C. He and N.H.C. Yung. Curvature scale space corner detector with adaptive threshold and dynamic region of support. In *Proceedings of the 17th International Conference on Pattern Recognition*, pages 791–794, Aug 2004.
- [34] A suite of minimal bounding objects. <https://nl.mathworks.com/matlabcentral/fileexchange/3476-a-suite-of-minimal-bounding-objects>. [Online; accessed January, 29, 2017].
- [35] Silke Wagner and Dorothea Wagner. *Comparing clusterings: an overview*. Universität Karlsruhe, Fakultät für Informatik Karlsruhe, 2007.

## Appendix 1. Examples of data images



**Figure A1.1.** Examples of the real data images.

## Appendix 2. Examples of data images



**Figure A2.1.** Examples of the synthetic data images.