Object segmentation within microscope images of palynofacies ☆

J.J. Charles[a],*, L.I. Kuncheva[a], B. Wells[b], I.S. Lim[a]

[a]*School of Informatics, University of Wales, Bangor LL57 1UT, UK*
[b]*Conwy Valley Systems, UK*

Abstract

Identification of fossil material under a microscope is the basis of micropalentology. Our task is to locate and count the pieces of inertinite and vitrinite in images of sieve sampled rock. The classical watershed algorithm oversegments the objects because of their irregular shapes. In this paper we propose a method for locating multiple objects in a black and white image while accounting for possible overlapping or touching. The method, called Centre Supported Segmentation (CSS), eliminates oversegmentation and is robust against differences in size and shape of the objects.
Crown Copyright © 2008 Published by Elsevier Ltd. All rights reserved.

1. Introduction

Accurate recognition of biological remains found in palaeosediments is the basis of palynology and micropaleontology and this underpins the interpretation of palaeoenvironment, chronostratigraphy and much more. The fundamental importance of accurate recognition of fossil material under the microscope has spurred considerable effort into automating the task. In the last 25 years significant developments have been made in recognising specific types of fossil material under ideal conditions (England et al., 1979; France et al., 2000; Weller et al., 2005).

Nevertheless, obtaining statistically significant, unbiased and reproducible results from automated analysis of microscope images is still regarded as a challenge. Even in the unrealistically simple case of individual, whole, undeformed specimens, difficulties arise from the diversity of the species to be recognised, the variability in the image acquisition techniques as well as the subjectivity of the visual analysis (Bollmann et al., 2004).

Starting with a mixture of objects arranged randomly in an image, we seek to create a collection of individual specimens. Counting objects in an image is straightforward for disconnected objects or objects of a particular known shape. However, counting connected or overlapping objects of arbitrary shape can prove difficult. The standard approach to this task consists of two steps. First, the image is segmented into background and foreground so that the objects of interest appear as a black foreground. Second, the foreground is further segmented to identify separate objects.

☆ Code available from server at http://www.iamg.org/CGEditor/index.htm.

*Corresponding author. School of Computer Science, Bangor University, Bangor LL57 1UT, UK. Tel.: +44 1248 383661; fax: +44 1248 361429.

E-mail address: j.j.charles@bangor.ac.uk (J.J. Charles).

In this paper we expose some deficiencies of the current object-segmentation methods (e.g., their tendency to oversegment and the need for a subjective placement of markers in the image). We propose a new segmentation method whose pre-set parameter corresponds to the human perception of *overlap* and does not depend on the image resolution.

The rest of the paper is organised as follows. Section 2 gives more details about the type of palynological images considered in this study and the objects within such images. Section 3 briefly explains the pre-processing of the image to segment out the background. Section 4 presents an overview of existing object-segmentation methods and highlights the problems arising. Section 5 introduces our algorithm for finding object centres, called Centre Support Segmentation (CSS). The results of an experiment are presented in Section 6. Section 7 offers conclusions and outlines future research directions.

2. Challenges of object segmentation in images of palymnomorphs

Finding roughly elliptical cells or pollen spores in an image is considerably easier than finding objects in palynomorph images. The material being analysed has arisen from biological remains. These remains are subjected to initial distress at time of deposition and subsequently altered and deformed by burial stresses and tectonic deformation. Furthermore the remains are then retrieved from their current position deep in the Earth by techniques which were not designed primarily for optimum sample preservation. Finally, the processed material is arranged haphazardly on a slide, with both material of interest and other materials overlapping and partially hiding each other.

Fig. 1 shows a typical image of an assemblage of objects retrieved by sieve analysis from a sediment. The material consists of light or transparent objects (palynomorphs and amorphous organic matter) and opaque humic kerogen which can be subdivided into inertinite and vitrinite. The dark objects have irregular shapes and different sizes. Sometimes small dark objects appear within light objects. The light objects, on the other hand, vary in texture, intensity and transparency. Up to now, human intervention has been assumed in detecting the objects in the image. At the next stage, automatic classification of the cropped objects can be
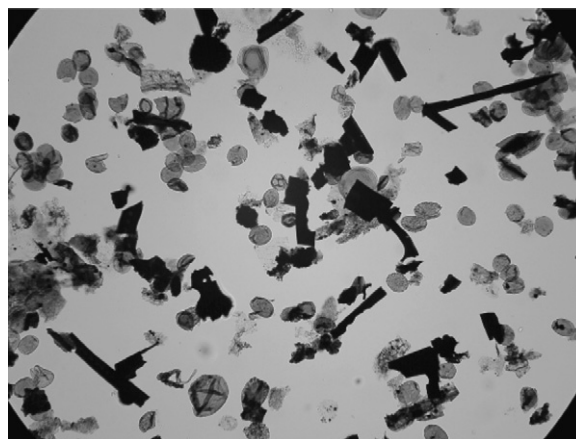

Fig. 1. Microscope image of palynofacies. Dark objects are referred to as "kerogen".

attempted based on expert knowledge, extraction of salient features and various machine learning and pattern recognition methods (Weller et al., 2005). Our goal in this study is to devise an algorithm for automatic identification of dark objects in the image. Such an algorithm will be a step towards a completely automatic classification system for palynological images.

In addition to the task facing the geologist, the image analysis has to address the reality that each slide is seen under different conditions, e.g., variations in light intensity, colour balance, background, etc. These variations need to be accounted for in order to create conditions which are sufficiently similar or standard for the algorithms to be usable and the results to be statistically valid. Only then can the large body of work on recognition of individual specimens be made commercially useful.

3. Image pre-processing to segment the kerogen material as foreground

The first stage of our long-term project is to extract "dark" objects that correspond to kerogen material. These objects will be later classified into inertinite and vitrinite. Hence the background/foreground segmentation must leave only the kerogen material as the foreground while labelling all the light or transparent fossils and amorphous material in the image as background. Below we include some details of our background segmentation algorithm. While this part is not directly related to the proposed Centre Support Segmentation

(CSS) method, we include it here to enable reproducibility of the results.

The common procedure for background removal is to apply an intensity threshold on the grey-scale image (Gonzalez et al., 2004; Bollmann et al., 2004). Global thresholding may produce a false foreground containing some of the less illuminated parts of the image together with all their content. To avoid this, background illumination should be equalised. An empty slide can be stored and used for background correction (Weller et al., 2005). The problem with the empty-slide method is that it is specific for the current settings of the microscope and will not produce accurate results in images of a different origin. Further methods include blurring and normalisation (Zawada, 2003, SYSTAT[1]) as well as fitting a two-dimensional quadratic function.

Here we use a recently proposed approach for illumination correction in microscope images (Charles et al., 2008) based upon fitting parabolas in horizontal and vertical "stripes" of the image. The grey level histogram of the corrected image has two distinct peaks, the left peak corresponding to the inertinite and vitrinite material and the right peak representing everything else. A segmentation threshold is then applied, chosen to be the intensity corresponding to the minimum between the peaks. This thresholding will result in a black and white image (binary image), black regions representing the pieces of interest to be segmented into objects that will be further labelled as inertinite, vitrinite or other.

4. Object-segmentation methods

Extracting objects from the image is not a trivial segmentation task. Typically, centre-based segmentation algorithms such as centroidal Voronoi tessellation (Du et al., 2006) and mean-shift algorithms (Comaniciu and Meer, 1997) seek to partition the image into homogenous regions. In our case, the image is already segmented into black and white, and the only guide to splitting the connected components into "objects" is the shape of the silhouette.

4.1. The classical watershed algorithm

The watershed transform (Vincent and Soille, 1991) can be used to identify separate objects on the

black and white image. First, a distance transform is applied (Borgefors, 1986) to the image (a faster approximation, called chamfer distance transform, can be used instead Butt and Maragos, 1998). The distance function $D(p)$ for a pixel p gives the Euclidean distance to the nearest white pixel. One can consider it as a surface with $D(p)$ being the height for pixel p. The watershed transform is applied to $-D(p)$. Its elegant interpretation is that the troughs are filled with "water" in order to find the watershed ridge lines. These lines partition the image into regions so that one object is contained within each region. This process is very effective for segmenting touching objects with circular shapes. However, rectangular objects cause difficulties and oversegmentation occurs due to the increased number of regional minima in the distance transform function. The problem in using the watershed method for detecting dark objects in palynomorph images is illustrated in Fig. 2. The number of regions the watershed method finds in the image is 56 while we are looking for just three objects.

4.2. Marker-based methods to prevent oversegmentaion

The oversegmentation of the watershed algorithm can be reduced by finding markers for each object and only allowing water to fill from the markers position. This is known as marker-controlled watershed segmentation (Vincent, 1993; Beucher, 1992; Landini and Othman, 2003). Provided each object only contains one marker, the segmentation will be near perfect. However, too few or two many markers will result in under or over-segmentation, respectively. One possible solution is to place the markers manually. However, this defeats the purpose of an automatic system for object extraction and classification. An alternative to the manual approach is to identify specific features found inside individual objects (Clocksin, 2003; Lindblad
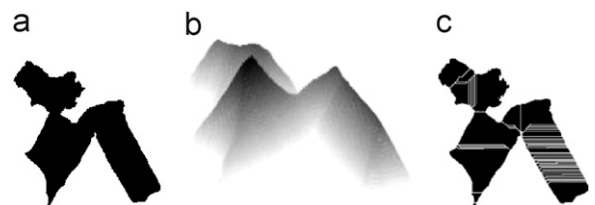

Fig. 2. (a) Cropped subimage from Fig. 1, (b) distance function as a surface and (c) result of watershed algorithm applied to negative distance function (56 regions identified).

---

et al., 2003). For example, when segmenting images of overlapping cells an ideal marker would be a single feature that presents itself in the centre of each cell. Locating the position of a cell nucleus would provide us with a perfect marker system.

Another possibility to identify centres is by using grey-scale morphology. The extended *h*-maxima transform (Soille, 2003) is an example of this approach. It is widely used in applications for separating touching objects of similar size in grey-scale images (Malpica et al., 1997; Wahlby et al., 2004). The transform can be applied to the distance function of the binary image or to the original grey-scale image itself. The method will filter out all maxima whose heights are smaller than the pre-defined threshold *h*. A low value of *h* will result in many markers and a high value will produce only a few markers. This transform is dependent upon the scale of the binary image. For example, if an image is rescaled, different numbers of objects may be found for a fixed value of *h*, even though the same number of objects are present in the image. The best value for *h* is usually determined by evaluating by eye the segmentation on a small sample of the images of interest. This value is then fixed when segmenting the other images.

While the extended *h*-maxima transform has been very successful for separating touching objects of similar size in grey-scale images, this may not be the case in images containing objects of various sizes and shapes. Fig. 3 shows an example of an image containing three objects altogether: a long wavy object and two overlapping circles. Suppose that the extended *h*-maxima transform is applied to the distance function of the image. The possible range of *h* is $[0, 72.1]$. To detect the *three* objects at the given image resolution, *h* must be between 3.8 and 7.6. For all values $11.4 < h \leqslant 45.5$, only two objects will be detected. The overlapping circles will not be marked for values $49.3 < h \leqslant 72.1$ resulting in the segmentation of one object.

Since palynomorph images will contain both large and small objects of different shapes, a more accurate segmentation method is required.

5. Centre Supported Segmentation (CSS)

We propose an alternative segmentation method based on an intuitive, scale-independent *overlap* parameter. This method will eliminate oversegmentation and successfully segment both circular and elongated objects. Centre Supported Segmentation
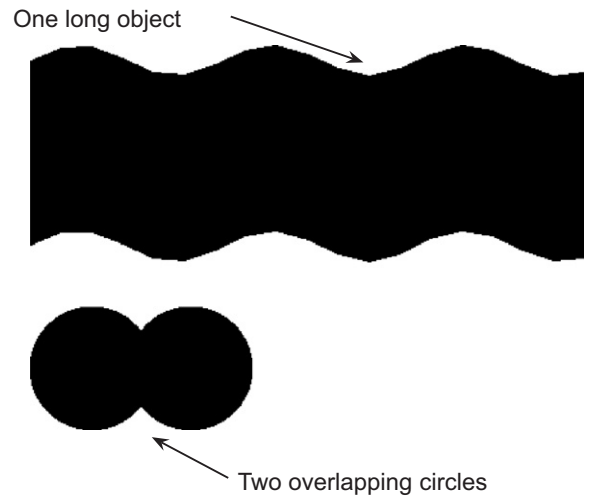

Fig. 3. Image showing three objects, one long wave and two overlapping circles.

(CSS)[2] is based on automatic identification of a centre point for each object. CSS is applied on the black and white image where the black foreground is the object to be segmented. The result from CSS is a set *C* of object centres. The centre of an object is needed for several reasons: (1) counting the number of objects, (2) viewing an object by moving the scanning camera to the centre and (3) cropping the object for further analysis and classification.

Definition. Any pixel *p* with the largest distance $D(p)$ within the object is called a *centre* of this object.

For example, the centre of a filled-in circle will be its geometrical centre. However, a doughnut-shaped object will have infinitely many centres, none of which will be the geometrical centre of the figure.

5.1. Finding the centres

A pixel *p* with coordinates (x, y) has an *eight-neighbourhood* consisting of the set $N_8(p) = \{(x+1, y), (x-1, y), (x, y+1), (x, y-1), (x+1, y-1), (x+1, y+1), (x-1, y-1), (x-1, y+1)\}$. Two pixels *p* and *q* are *eight-connected* if there exists a path of pixels between *p* and *q* where each pixel in the path is of the same intensity and in the eight-neighbourhood of the next pixel. A set of pixels that are all connected to one another is called a *connected component*. CSS is applied to the distance function

---

[2]The images and the Matlab code are available on the IAMG server www.iamg.org.

of the binary image. It first identifies the centres of all possible objects and then filters out the centres which are likely to be noise. The first stage of the algorithm is explained in Table 1.

Two lists are created: the list C of centres and the list V of their *merging heights*. List C is constructed in the following way. Initially C is empty. Suppose that $m = D(q_1) = \max_p D(p)$ is the unique maximum of D across the whole image. Pixel q_1 is taken to be the first centre in C. This thresholding will define a black and white image B, containing one black point at q_1. Consider as an example the image in Fig. 4(a). The boxes delineated by the grid are the

pixels in the image. Fig. 4(b) displays the distance function for the image. All white pixels have $D(p) = 0$. For this example, the maximum of the distance function is 3, found at pixel $q_1 = (6, 4)$. Then C is updated by $C \leftarrow C \cup \{(6, 4)\}$.

By thresholding D at m, all pixels p where $D(p) \geqslant m$ are set to black and the rest are set to white. For $m = D(q_1)$, there will be a single black dot at q_1. Fig. 5(a) shows this first step. The black pixel is q_1.

The next maximum height, $m' = \max_{p \neq q_1} D(p)$ is identified. The new black and white image B' resulting from the thresholding with m' will contain

Table 1
The Centre Supported Segmentation (CSS) algorithm

CENTRE SUPPORTED SEGMENTATION (CSS) ALGORITHM: C AND V

    (1) Given is a binary image B. Initialise $C = \emptyset$, $V = \emptyset$

    (2) Find the distance image D for B and sort all the distinct distance values in descending order: $m_1 > m_2 > \cdots > m_T$

    (3) For $i = 1 : T$

        (a) Find binary image B_i by thresholding D at m_i

        (b) For $j = 1 : |C|$ (each centre in C)

            (i) find the connected component Z_j

            (ii) if Z_j has no intersection with any of $Z_1, \ldots, Z_{j-1}$, then set $v_j = m_i$

        (c) Remove all connected components $Z_1, \ldots, Z_{|C|}$ from B_i

        (d) Find the remaining connected components in B_i, e.g., $Z_{i,1}, \ldots, Z_{i,k}$. For each component, find $q_{i,t} = \arg\max_{p \in Z_{i,t}} D(p)$ as the centre of this component, $t = 1, \ldots, k$[a]

        (e) Augment C and V

$$C \leftarrow C \cup \{q_{i,1}, \ldots, q_{i,k}\}, \quad V \leftarrow V \cup \underbrace{\{m_i, \ldots, m_i\}}_{k}$$

    (4) Return C and V

Stage 1: obtaining centres C and merging heights V.

[a]*Note*: As each possible distance is checked, the connected components at distance m_i, after removing all connected components at (3c) will all be exactly at height m_i. These may be single points or clusters of points at the same height. Thus any point from $Z_{i,t}$ can serve as the centre of the component.
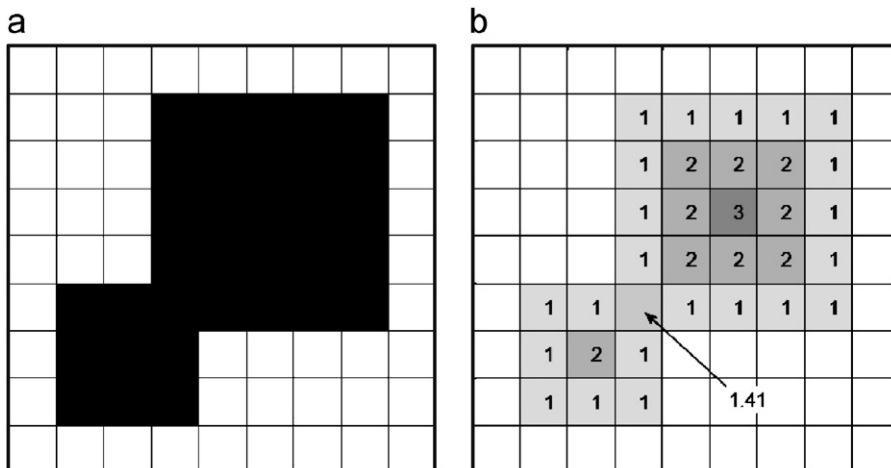

Fig. 4. Example of a 9-by-9 binary image (a) and its distance function (b).

more black points than the image thresholded at $m > m'$, including points around pixel q_1. Fig. 5(b) displays B' found at $m = 2$, the next lower value of the distance function D.

Since the object with centre q_1 is already accounted for, we remove this connected component from B'. The component to be removed from B' is coloured light grey in Fig. 5(b). Let Z be the set of pixels of a connected component in the remaining B'. Denote the centre of the object represented by Z by q_2, where $D(q_2) = \max_p D(p), \forall p \in Z$. The new centres are appended to the current C. In our example $C = \{(6,4),(3,7)\}$. A subsequent threshold $m'' < m'$ is applied to produce image B'' from D, and all connected components with centres in C are removed from B'' in the order they were stored in C. The remaining connected components are used to find new centres, and so on. Fig. 5(c) shows the third step where B'' is obtained by thresholding at $m'' = 1.41$. There is one connected component in this image, which will be removed because it contains the first entry in the set of centres C, q_1. Finally, Fig. 5(d) shows B''' with $m''' = 1$. Again, only one connected component is found and subsequently removed.

If we apply this process to the image in Fig. 2(a), 56 centres will be found, each one located in its own separate region defined by the watershed algorithm (subplot (c)). As with the watershed method, small shape irregularities on the periphery of the object will result in a jagged peak of the distance function with many local maxima of similar heights. Each little spike will generate a centre.

To eliminate oversegmentation we propose the following addition to the algorithm. A parameter called *merging height* is attached to each centre. The merging height of centre q_i, denoted v_i, $(v_i \leqslant D(q_i))$, is the lowest height at which q_i defines a connected component disjoined from any connected components of q_j such that $D(q_j) > D(q_i)$. For any value lower than v_i, q_i and another centre at a larger D will share a connected component. Figuratively speaking, the object of smaller size (smaller peak $D(q_i)$) is eclipsed by an object of a bigger size $(D(q_j) > D(q_i))$. For the example in Figs. 4 and 5, the merging height of q_1 is $v_1 = 0$, and the merging height of q_2 is $v_2 = 2$ because this is the lowest height where the component of q_2 is separate from the component containing q_1.

The merging heights of centres of large objects will be low even if they overlap with smaller objects. The merging height of large objects will be updated until they are joined to a larger object. On the other hand, centres of smaller objects corresponding to noise at the peak will have high merging heights. The centres with large v_i will be candidates for elimination.

The algorithm for identifying the centres C and their merging heights V is detailed in Table 1.

In Stage 2 of the CSS algorithm redundant centres are eliminated. A threshold s can be applied to account for the minimum allowable size of an object. All objects with centres q, such that $D(q) < s$, are discarded. If the algorithm is run with $s = 0$, it will find all the specs in the image as objects of interest. The value of s can be estimated by eye or can be learned from a sample of training images where the objects of interest have been pre-labelled by hand. Weller et al. (2005) propose an empirical threshold of 14 μm.

5.2. The overlap parameter

The cropped image in Fig. 2(a) looks like three touching objects, however, it may also be a genuine set of 56 tightly packed objects. We introduce a parameter d to determine which centres need to be removed. The degree of overlap is defined using two intersecting circles as demonstrated in Fig. 6(a) and is measured with respect to the smaller circle. If the two circles are of the same size, each of the two can be chosen. The overlap value is defined by
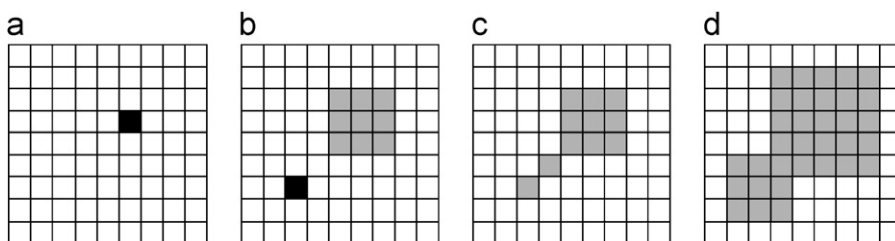

Fig. 5. Finding centres using CSS. Two centres will be stored in C, $q_1 = (6,2)$ at threshold $m = 3$ and $q_2 = (3,7)$ at threshold $m = 2$.

the two intersection points A and B. Denote by t_i the minimum distance from the mid-point of the segment AB to the edge of the smaller circle. The overlap is defined as the ratio of t_i to the radius of the circle, $D(q_i)$. The length t_i is found using v_i, the merging height of the centre q_i. The merging height is half the length of AB and is shown as v_i in the diagram. Since $t_i = D(q_i) - \sqrt{D(q_i)^2 - v_i^2}$, the degree of overlap for a centre q_i is

$$O(q_i) = 1 - \sqrt{1 - (v_i/D(q_i))^2}.$$

By definition $D(q_i) > v_i$ and so $O(q_i) \in [0, 1)$. A centre q_i with overlap $O(q_i) = 0$ means that the object is isolated. As the overlap approaches 1, the object is increasingly covered by a larger item. The overlap of two circles is demonstrated in Fig. 6(b). The small circle is increasingly covered by the larger circle. The overlap value $O(q)$ is also shown. As soon as the two circles merge so that $q_i = t_i$, the CSS algorithm will continue to recognise one object, in this case we have complete overlap.

5.3. Filtering the centres using the overlap

To remove centres from C we set a limit d on the amount of overlap such that if $O(q_i) \leqslant d$ then centre q_i is kept in C and discarded otherwise. If we wish to segment an image to its maximum detail then the threshold is set to $d = 1$. This would return the same number of segments as the watershed algorithm. By adjusting d oversegmentation can be prevented. The

"noisy" centres occur due to small deformations in the shape and this will correspond to relatively large merging heights yielding large overlap values. Hence setting a threshold d not only specifies the connectivity of objects but also eliminates the "noisy" centres. The effect of setting a maximum allowable overlap can be seen in Fig. 7. At $d = 0.5$ we have a "correct" (intuitive) segmentation into three objects and at the limiting case of $d = 1$ we obtain the 56 segments that are produced by the watershed algorithm. In our experiment it would seem a human can most easily separate two overlapping objects provided that their overlap is no greater than 0.5. Hence a chosen value of $d = 0.5$ will best describe this behaviour.

5.4. Extracting the objects

After filtering the centres the corresponding individual objects can be extracted. CSS belongs in the class of marker-controlled segmentation algorithms. Hence the watershed algorithm is applied to the negative of the distance image $D(p)_{neg} = -D(p)$ modified so that only regional minima occur at each centre. This technique is known as *image imposition* (Soille, 2003). In this way the centres are used as markers and the watershed algorithm will segment according to the position of the centres. The watershed ridge lines will form boundaries for each object. To extract the objects, the watershed boundaries are overlaid in white on top of the black and white image. Thus the new binary image
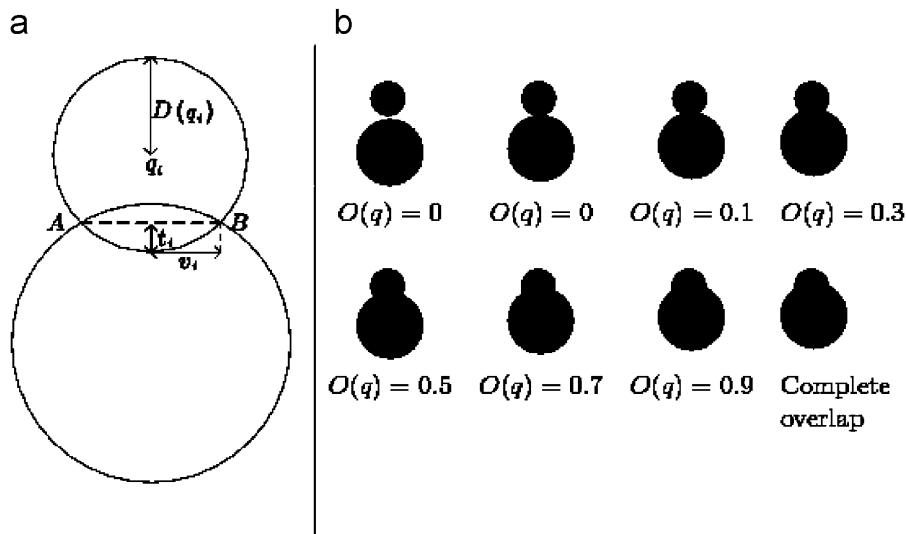

Fig. 6. (a) Defines concept of overlap, (b) illustrates value of overlap $O(q)$.

Fig. 7. Effect of allowable overlap d on image segmentation. Best segmentation (three centres) is obtained for d between 0.3 and 0.5.

will consist of black connected components corresponding to objects, which can be easily extracted for further analysis.[3]

6. Experimental results

6.1. An illustration

Fig. 8 displays the centres of dark objects within the image in Fig. 1. The original image of size 1704×2272 pixels was sampled every 3 pixels. By sampling the image we decrease the processing time for CSS, the sampling number is not sensitive to the output of CSS and is sufficient provided the sampled image can still be segmented into individual objects by human eye. Once the coordinates of centres have been found for the sampled image they were mapped back to the original image so that extraction of objects can proceed. The chosen parameter values were $s = 4$ and $d = 0.5$. The objects were then extracted, the first 30 largest objects are shown in Fig. 9.

6.2. A comparison with the watershed method

To evaluate the quality of the obtained set of centres C against a known set of centres C^* we use a recently proposed measure $S(C, C^*)$ (Charles et al., 2006). The three components of the measure evaluate the under/over segmentation of the objects, the proportion of centres placed in the background rather than in objects, and the distance between the

---

[3]Matlab function *bwlabel* can be used to label all connected components.
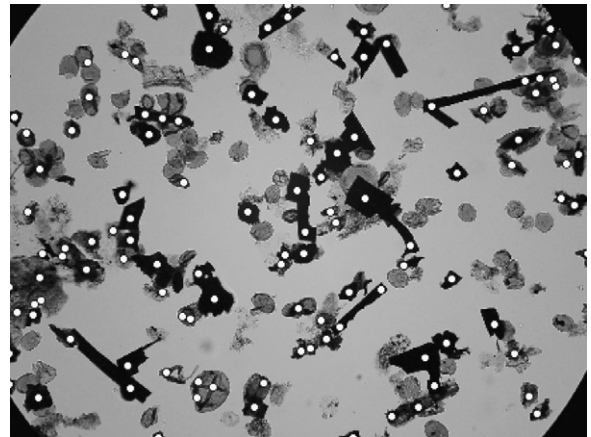

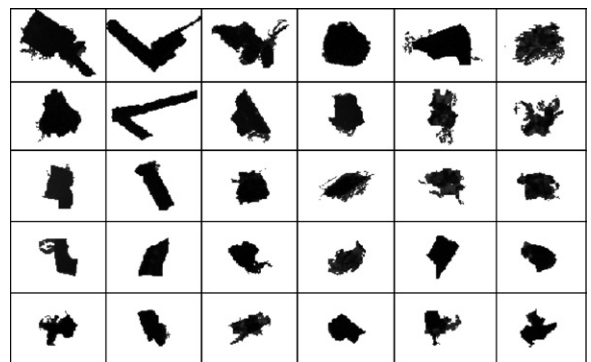Fig. 8. Centres found by CSS.

Fig. 9. Objects extracted using image imposition with centres acting as markers.

guessed and the true centres. The measure varies between 0 and 1 where 0 signifies perfect match while 1 means that set C is useless. Table 2 shows the results on several microscope images of palynomorphs. After the background was removed "ideal centres" C^* were found by first segmenting the black and white image by hand and then calculating the centre of each connected component according with Definition 1. For example, a long and roughly rectangular object would receive a centre somewhere on its ridge, not necessarily in the middle. The values of S and the processing times indicate that CSS is slower but consistently more accurate than the watershed segmentation method.

6.3. A comparison with the extended h-maxima transform

A similar comparison was made between the extended h-maxima transform and CSS. The best

Table 2
Segmentation results for the watershed method, Centre Supported Segmentation (CSS) and extended h-maxima on six microscope images of palynofacies

Image no.	Number of objects	Watershed		Extended h-maxima		CSS	
		$S(C, C^*)$	Time (s)	$S(C, C^*)$	Time (s)	$S(C, C^*)$	Time (s)
1	73	0.43	7.37	0.33	19.10	0.14	48.87
2	72	0.58	7.32	0.45	46.65	0.22	107.07
3	82	0.60	7.59	0.48	44.10	0.29	94.21
4	14	0.55	7.31	0.39	33.58	0.13	73.05
5	77	0.53	7.39	0.41	38.89	0.22	87.61
6	25	0.50	7.30	0.38	16.89	0.11	31.90

Note: Small values of $S(C, C^*)$ indicate better match between the obtained (C) the ideal (C^*) centres.

value for h was determined by eye on one of the images. This value was then fixed when segmenting the other images. The segmentation results and times shown in Table 2 indicate that CSS outperforms the extended h-maxima transform. The following arguments shed a light on the reasons for this result:

- The value h depends upon the scale of the input image. The six images used in the experiment were at similar resolutions but not completely identical. Slight changes in resolution might have affected the results. The CSS overlap parameter d is scale independent so it consistently segments objects regardless of the variability in the image resolution.
- The images contained both small objects and large objects. Choosing a value of h that would segment both the large and small objects correctly is difficult, if possible at all. In some cases a large value of h is required to successfully segment large overlapping objects. This results in some smaller objects not being given markers by the transform.
- The shapes of the objects in our images were different, some were long or rectangular and others were circular. The extended h-maxima transform does not support the segmentation of both long and circular objects. The output of the algorithm in these cases is sensitive to h, especially towards the lower end of the scale. Consider as an illustration the image in Fig. 3 (a long wavy object and two overlapping circles). The extended h-maxima transform was applied to the distance function for 20 values of h spanning uniformly the whole range of possible values. Similarly the CSS algorithm was applied with the values of d spanning its range [0, 1]. The

graphs in Fig. 10 depict the number of segmented objects in each case. The long wave should be segmented as one object and the overlapping circles should be separated into two objects. The highlighted region shows the range in which this segmentation occurs. The extend h-maxima transform will only produce the correct results for a very small range of h. The CSS algorithm will yield the correct result for a much larger range of values in d. It appears the extended h-maxima transform is more likely to either correctly segment long objects but under-segment circular ones or oversegment long objects while correctly segmenting circular ones.

- The value of h requires human intervention in order to successfully segment objects of a particular size. On the other hand, the CSS algorithm can be run for any image with the most intuitive results by fixing $d = 0.5$. This again is shown in Fig. 10(b) as 0.5 is contained within the highlighted region.

7. Conclusion

We propose an algorithm for automatic detection of the centres of objects in an image and extraction of the corresponding objects. The algorithm, called Centre Supported Segmentation (CSS), remedies the oversegmentation problem of the watershed method traditionally used for such segmentation. The experiments show that CSS is more accurate although slower than the watershed method. A comparison with a mark-based segmentation method using extended h-maxima transform revealed that CSS is more accurate and robust with respect to its parameter called *overlap*. The algorithm relies on another parameter, s, which is the threshold for
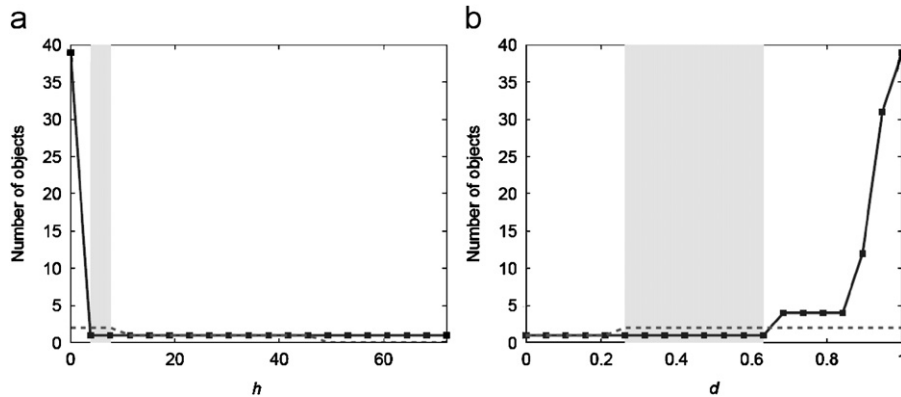
Fig. 10. (a) Segmentation results of extended h-maxima. (b) Segmentation results of CSS. Solid lines show segmentation of long object, dashed lines show segmentation of overlapping circles. Regions of correct segmentation are highlighted.

cleaning the "dust specs" from an image. This parameter can be picked in advance and applied across the whole domain knowing that pieces of kerogen with a radius smaller than t μm are of no interest (not considered objects). Then s can be calculated automatically for each image knowing the resolution at which the image has been captured and the microscope resolution.

The method for finding centres is a step towards building an automatic system for identifying palynomorphs and humic kerogen in images of rock samples captured through a microscope. Once the objects are extracted, classification methods can be applied to determine their types. In this study we demonstrated how CSS operates for extracting dark objects, which could be further classed into inertinite and vitrinite. After the dark objects are removed, the remaining image will contain palynomorphs and amorphous objects. These objects can be extracted in turn using again CSS. As CSS is not meant to work on-line, the processing time is not crucial. Nevertheless future effort will be also directed towards speeding up CSS.

Application of CSS may be sought in various domains, e.g., segmenting cell nuclei and setting the initial position of active contours (Clocksin, 2003), separating pollen grains for automated analysis (France et al., 2000) and marker-controlled segmentation (Gonzalez et al., 2004).

Appendix A. Supplementary data

Supplementary data associated with this article can be found in the online version at doi:10.1016/j.cageo.2007.09.014.

References

Beucher, S., 1992. The watershed transformation applied to image segmentation. Scanning Microscopy 6, 299–314.

Bollmann, J., Quinn, P.S., Vela, M., Brabec, B., Brechner, S., Cortés, M.Y., Hilbrecht, H., Schmidt, D.N., Schiebel, R., Thierstein, H.R., 2004. Automated particle analysis: calcareous microfossils. In: Francus, P. (Ed.), Image Analysis, Sediments and Paleoenvironments. Kluwer Academic Publishers, Dordrecht, Netherlands, pp. 229–252.

Borgefors, G., 1986. Distance transformations in digital images. Computer Vision Graphics and Image Processing 34 (3), 344–371.

Butt, M.A., Maragos, P., 1998. Optimum design of chamfer distance transforms. IEEE Transactions on Image Processing 7 (10), 1477–1484.

Charles, J.J., Kuncheva, L.I., Wells, B., Lim, I.S., 2006. An evaluation measure of image segmentation based on object centres. In: Proceedings of the International Conference on Image Analysis and Recognition ICIAR. Portugal, pp. 283–294.

Charles, J.J., Kuncheva, L.I., Wells, B., Lim, I.S., 2008. Background removal in microscopy images. In: Proceedings of the 3rd International Conference on Computer Vision Theory and Applications (VISAPP). Portugal, in press.

Clocksin, W., 2003. Automatic segmentation of overlapping nuclei with high background variation using robust estimation and flexible contour models. In: Proceedings of the 12th International Conference on Image Analysis and Processing. IEEE Computer Society, Los Alamitos, CA, USA, pp. 682–687.

Comaniciu, D., Meer, P., 1997. Robust analysis of feature spaces: color image segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Puerto Rico, pp. 750–755.

Du, Q., Gunzburger, M., Ju, L., Wang, X., 2006. Centroidal Voronoi tessellation algorithms for image compression, segmentation, and multichannel restoration. Journal of Mathematical Imaging and Vision 24 (2), 177–194.

England, B., Mikka, R., Bagnall, E., 1979. Petrographic characterization of coal using automatic image analysis. Journal of Microscopy 116, 329–336.

France, I., Duller, A., Duller, G., Lamb, H., 2000. A new approach to automated pollen analysis. Quaternary Science Reviews 19 (6), 537–546.

Gonzalez, R.C., Woods, R.E., Eddins, S.L., 2004. Digital Image Processing Using Matlab. Pearson Education Inc., USA, 782pp.

Landini, G., Othman, I., 2003. Estimation of tissue layer level by sequential morphological reconstruction. Journal of Microscopy 209 (2), 118–125.

Lindblad, J., Waehlby, C., Bengtsson, E., Zaltsman, A., 2003. Image analysis for automatic segmentation of cytoplasms and classification of Rac1 activation. Cytometry Part A 57, 22–33.

Malpica, N., de Solorzano, C., Vaquero, J., Santos, A., Vallcorba, I., Garcia-Sagredo, J., del Pozo, F., 1997. Applying watershed algorithms to the segmentation of clustered nuclei. Cytometry 28 (4), 289–297.

Soille, P., 2003. Morphological Image Analysis: Principles and Applications. Springer, New York, 407pp.

Vincent, L., 1993. Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. IEEE Transactions on Image Processing 2 (2), 176–201.

Vincent, L., Soille, P., 1991. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (6), 583–598.

Wahlby, C., Sintorn, I., Erlandsson, F., Borgefors, G., Bengtsson, E., 2004. Combining intensity, edge and shape informations for 2d and 3d segmentation of cell nuclei in tissue sections. Journal of Microscopy 215, 67–76.

Weller, A.F., Corcoran, J., Harris, A.J., Ware, J.A., 2005. The semi-automated classification of sedimentary organic matter in palynological preparations. Computers & Geosciences 31 (10), 1213–1223.

Zawada, D.G., 2003. Image processing of underwater multi-spectral imagery. IEEE Journal of Oceanic Engineering 28 (4), 583–594.