

# 6

## VORONOI BASED FACE SEGMENTATION

Abbas Cheddad  
Dzulkifli Mohamad

### 1 INTRODUCTION

Image segmentation, decomposition of a gray level or color image into homogeneous tiles, is arguably the most important low-level vision task. Homogeneity is usually defined as similarity in pixel values. Segmentation is a fundamental process in digital image processing which has found extensive applications in areas such as content-based image retrieval, medical image processing, and remote sensing image processing. Its purpose is to extract labeled regions or boundaries for targeted objects for subsequent processing such as surface description and object recognition. A segmentation procedure usually consists of two steps. The first step is to choose a proper set of features which can identify the same-content regions and meanwhile differentiate different-content regions; the second step is to apply a segmentation method to the chosen features to achieve a segmentation map” (Deng H. and Clausi D., 2004). Segmentation is partition of an image  $I$  into a set of regions  $S$  satisfying:

■ Partition covers the whole image:  $\cup S_i = S$  (1)

■ No regions intersect:  $S_i \cap S_j = \phi, i \neq j$  (2)

- Homogeneity predicate is satisfied by each region:  $\forall S_i,$   
 $P(S_i) = \text{true}$  (3)
- Union of adjacent regions does not satisfy it:  $P(S_i \cup S_j) =$   
false,  $i \neq j,$   $S_i$  adjacent  $S_j$  (4)

Applications of image segmentation can be found in a wide variety of areas such as remote sensing, vehicle and robot navigation, medical imaging, surveillance, target identification and tracking, scene analysis, product inspection/ quality control, etc. Image segmentation remains a long standing problem in computer vision and it is found difficult and challenging for two main reasons (Tu Z. and Zhu S., 2002). The first challenge is the fundamental complexity of modeling a vast amount of visual data that appear in the image. The second challenge is the intrinsic ambiguity in image perception, especially when it concerns the so called unsupervised segmentation (e.g.: A decision whereby a region cut is not a trivial task).

In the process of information visualization, one usually will be in front of *dot pattern* rather than color images or gray level ones. The need for this is ultimately to have a system that can tolerate the situations where there is sensitivity of analysis to changes in lighting conditions, scale orientation of the camera, sensor characteristics, geometric distortion...etc. Such invariant and noise tolerant systems are proven to be useful in various industrial and military applications (Dowdall J. *et al.*, 2003). Hence, extracting faces and facial features accurately and efficiently forms a wonderful problem. In addition to the above mentioned factors, the main problem is that facial features vary from individual to another in terms of shape and are highly deformable for one individual with multiple shots.

One of the main motivations behind adopting face features method is due to the fact of representing the face image in a very compact way which for sure lowers the memory needs. Its importance is noticed clearly when dealing with a huge database of faces. Features based methods basically work on a set of a unique dot patterns where trying to represent them in an efficient way.

This makes choosing suitable points extremely critical for the performance of a given segmentation system.

The proposed method will accept a gray scaled image as an input, however the work is mostly on its binary version, which corresponds, to the simplest and yet one of the most useful image types. In general; a binary image is a representation of scenes with only two possible gray values for each pixel, typically 0 or 1. In many cases, an object shape boundary is usually used for pattern recognition and can be represented as follows:

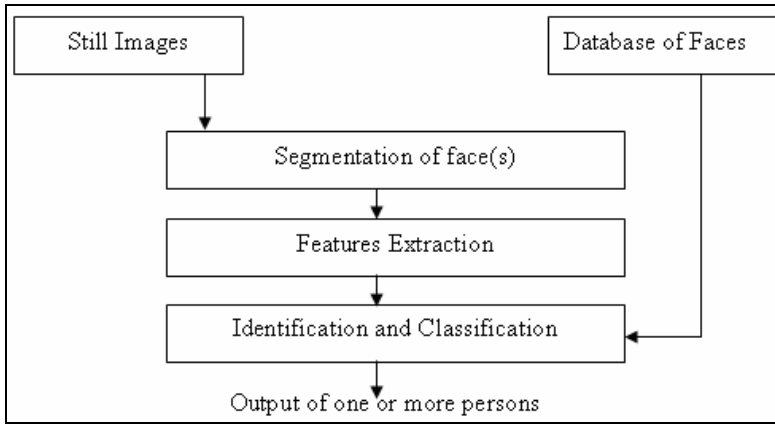
$$f(x, y)=1, \forall (x, y) \in \eta \quad (5)$$

$$f(x, y)=0, \forall (x, y) \notin \eta \quad (6)$$

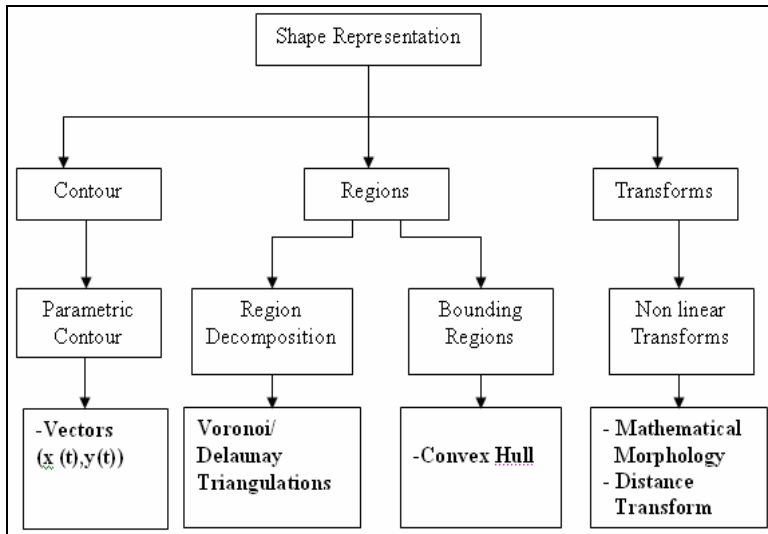
Where

$\eta$  denotes the region of the object in the image plane and  $x$  and  $y$  are the coordinates in the original image  $f$ . The binary image is useful in this study to extract easily labeled blobs and to feed them as input parameters to construct the Voronoi Diagram.

In this study a bottom-up approach is followed, that is, face is located first and fine features are extracted next. It is found that the accuracy of the result is affected by two main factors: the features segmentation and the flexibility of the classifier. A brief illustration of how a machine face recognition system looks like is depicted in Figure 1, while some techniques, which are going to be adopted for this study, are depicted in Figure 2.



**Figure 1** A block diagram for machine face recognition system



**Figure 2** A block diagram of shape representation techniques being used

## 2 SYSTEM OVERVIEW

The proposed method is benefiting from Voronoi properties to detect and segment faces and the block diagram is shown in Figure 3. The first step in every machine vision task is the pre-processing phase, which includes (herein) and not limited to: *Histogram Equalization*, *Mathematical Morphology*, *Median* and *low pass filtering*. The dual *processing* is adopted to reduce as much as possible the mal effect of the intensive uneven illumination. The latter might introduce two problems, the first is creating holes in the face region which in its turn disturb the algorithm efficiency, and the second is merging certain parts of the face/head with the background that leads to lose of the face shape. When a rough face region is presented to the system, a distance transform will be applied to determine the size of the window in which each possible feature (represented as a blob) is segmented. It should be noticed that the sequences of the process depicted in Figure 3 are highly interrelated whereby the success of each process is very much dependable on the predecessor components. As an example, if the face is not segmented well (including background portions or/and head hair), the process of locating face features is highly challenging task because of the unexpected interference of an unwanted components which make the system confused.

A room is left for the possibility of not finding all the pair of Eyes, the system will automatically generate a notice "*Feature not Found*" and hence the process will break. The next sections will discuss in details each step of the system and what it hosts.

Lastly the CPU is Pentium 4 / 2.80 GHz with 2.56 MB of RAM and the development environment for the system is done using MATLAB Version 6.5, figure 4 illustrates the Graphical User Interface (GUI) of the system. The Operating System is Microsoft Windows XP Home Edition Version 2002.

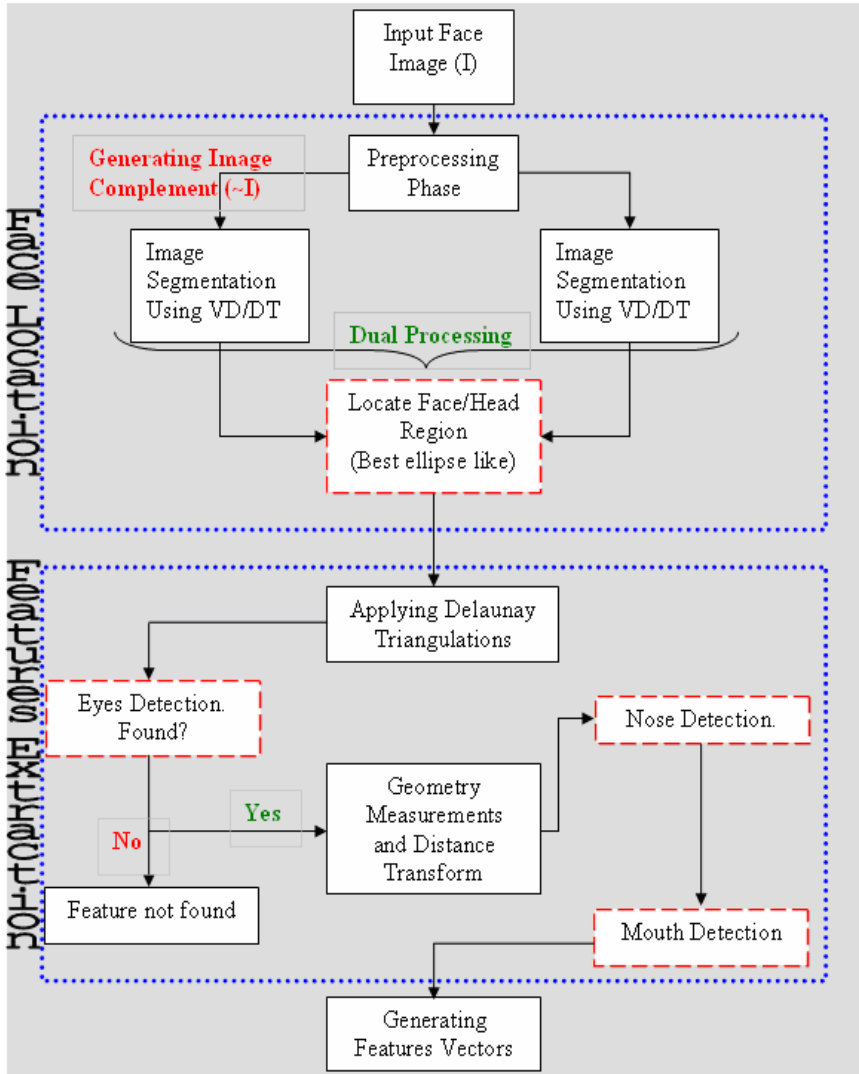
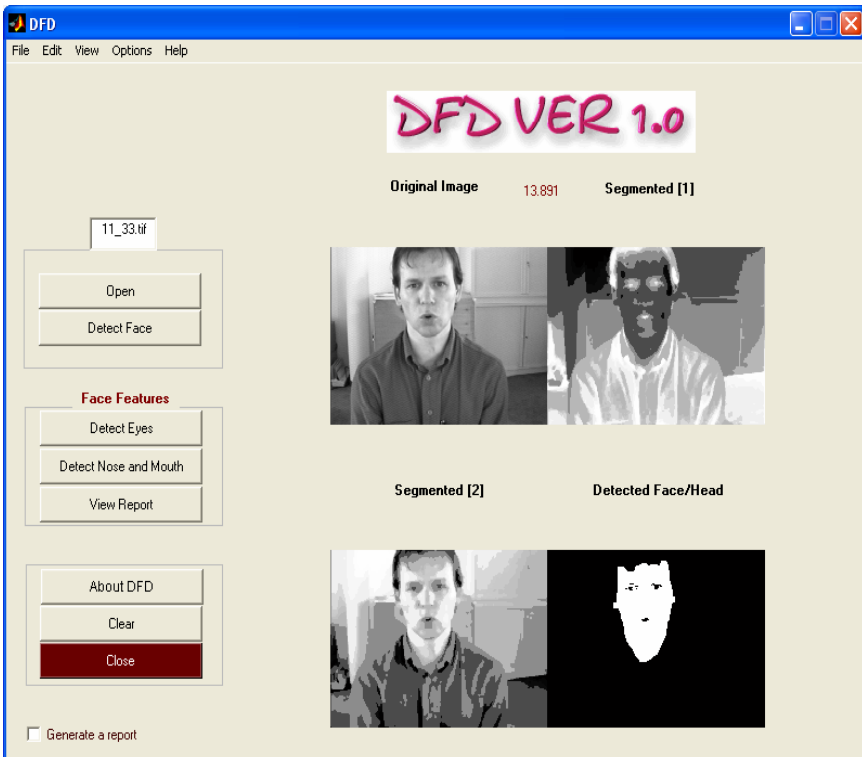


Figure 3 A general block diagram of the proposed system



**Figure 4** The GUI of the proposed system (1) Current image (2) The process with the information will be recorded in an external file (HTML) (3) Time consumed for face detection in seconds (4) Image complement segmentation (5) Direct Segmentation (6) Segmented face displayed as a blob and (7) Setting option

### 3 IMAGE ACQUISITION

The developed face database (indoor at the laboratory) are acquired through a single monochrome camera and embedded frame grabber see figure 5. The system uses PULNiX TM-200 camera that offers a high resolution interline transfer 1/2" CCD imager in a very tiny package. Designed to meet a variety of application requirements, this camera has many standard and optional features. It is available in both EIA (TM-200) formats. The most commonly needed adjustments-manual gain, gamma, AGC, and field/frame selection-are easily accessible on the rear camera panel. Eight shutter speeds, ranging from 1/60 sec. to 1/10,000 sec., can be selected externally using the rotary dial on the rear panel. Table 1 shows some specifications of the device.

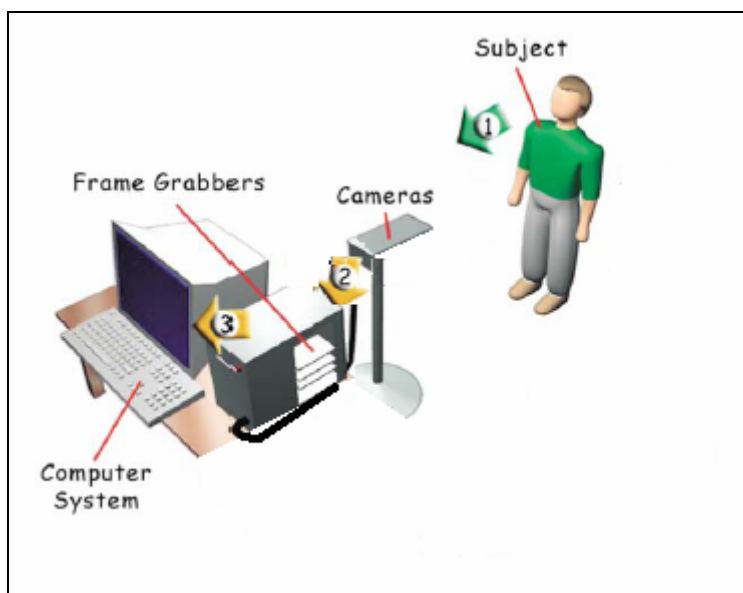
**Table 1** TM-200 specifications

	TM-200 (EIA)
<b>Imager</b>	1/2" Interline transfer CCD, HAD type
<b>Pixel</b>	768 (H) x 494 (V)
<b>Cell size</b>	8.4 $\mu\text{m}$ x 9.8 $\mu\text{m}$
<b>Power req.</b>	190 mA. 11-15V
<b>Size (W x H x L)</b>	45.8mm x 39.4mm x 66.3mm
<b>Weight</b>	157g (5.6 oz.)



As for the frame grabber Matrox Meteor-II/Multi-Channel has been used, which has the following key features:

- PCI, CompactPCI® or PC/104-Plus™ form factor
- Captures from interlaced and progressive scan single or dual channel monochrome analog video sources
- Sampling rates up to 30 MHz
- Three 256 x 8-bit LUTs
- 32-bit/33 MHz PCI bus-master
- real-time transfer to system or VGA memory
- extensive on-board buffering for reliable capture



**Figure 5** Hardware architecture of indoor image acquisition

## 4     **PRE-PROCESSING PHASE**

As stated before, the preprocessing step is very critical yet essential task. Due to noise introduced from the electronic signals of the digital camera, face images may experience some resulted pixels that do not reflect the scene. Lighting uneven effect is also another intruder. In addition to these, an image analyzer might want to reduce the impact of the background on the objects (faces), by filling up small holes or make them to vanish, this will of course depend solely on the purpose of the algorithm. Here, the next sections will bring about some famous and strong preprocessing techniques that have been used.

### 4.1     **Histogram Equalization**

Lighting conditions have an enormous effect on the segmentation too. That is why when matching faces; the conventional techniques based on correlation between gray levels are proven to be impractical, because intensity values are prone to change by different factors. Since a face illumination from the left looks different than that from the right, therefore, the image windows must be corrected for illumination. Generally, illumination effects tend to behave as a linear ramp. Thus a solution was needed. A simple way yet powerful to ameliorate the problem is by applying “*Histogram Equalization*”, which is described beneath:

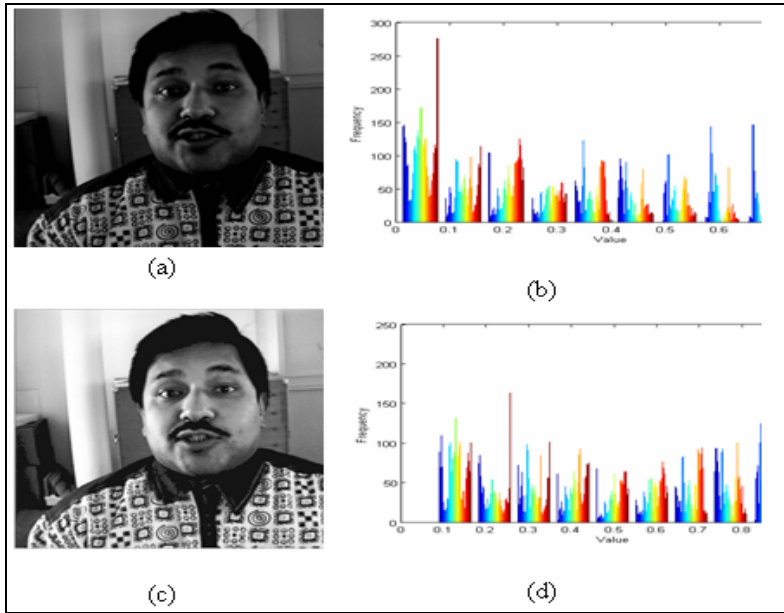
$$f(n) = (N - 1) \sum_{k=0}^n h(k) \quad (7)$$

It mainly enhances the contrast of images by transforming the values in an intensity image, or the values in the color-map of an indexed image, so that the histogram of the output image approximately matches a specified histogram or the one can see it as generating a flat histogram. This transformation transforms the intensity image  $I$  so that the histogram of the output intensity image let say  $J$  with length  $\ell$  bins approximately matches  $\ell$ . The vector  $\ell$  should contain integer counts for equally spaced bins with intensity values in the appropriate range:  $[0, 1]$  for images of class double or  $[0, 255]$  for images of class uint8. In example of Figure 6, a flattening histogram has been created:

$$\ell = \text{ones}(1, n) * \text{prod}(\text{size}(I))/n \quad (8)$$

where  $n$  denotes number of bins.

Beside flattening and equally distributing the histogram, this process will yield a visual enhancement on the host image. By looking at the previous example, the observer can realize the benefit of this method. As shown, the original image was badly affected by the mal illumination, especially when it hits the target object (face), whereas the output was greatly enhanced. The face and its features are clearly distinguishable now, leaving the room to the next process to take place.



**Figure 6** Histogram Equalization (a) Original BioID face image (b) its respective histogram (c) resulted image and (d) its respective histogram. Note the images were in double precision and there intensities values range from 0 to 1

## 4.2 Applying Morphological Operations

Morphological operations were discussed in the previous chapter. Choosing opening operation which is erosion followed by dilation is a wise step as stated by many literatures (G. Feng and P. Yuen, 2001) for it clears small objects in the background (noise removing) and enhances facial features.

Given an image  $I$  and structuring element  $\psi$ . Let  $\psi_x$  denote the translation of the  $\psi$  so that its origin is located at  $x$ . Then the Erosion of  $I$  by  $\psi$  is:

$$I \ominus \psi \triangleq \{x : \psi_x \subset I\} \quad (9)$$

Similarly the dilation of  $I$  by  $\psi$  is defined as:

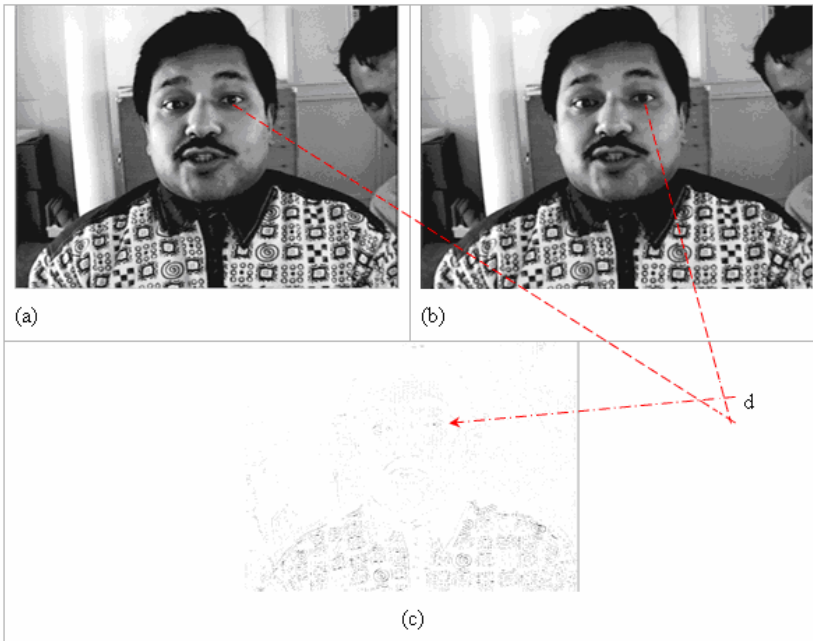
$$I \oplus \psi \triangleq \{x : \psi_x \cap I \neq \emptyset\} \quad (10)$$

Therefore the opening is the combination stated as follows:

$$I \circ \psi = (I \ominus \psi) \oplus \psi \quad (11)$$

A square structuring element with size of 2 has been used, Figure 7 gives a practical example.

Another Morphology based process is *Morphological reconstruction* which processes one image, called the *marker*, based on the characteristics of another image, called the *mask*. The high-points, or peaks, in the marker image specify where processing begins. The processing continues until the image values stop changing figure 8. It is a repeated dilation. To illustrate morphological reconstruction, consider this simple image. It contains two primary regions, the blocks of pixels containing the value 14 and 18. The background is primarily all set to 10, with some pixels set to 11.



**Figure 7** Opening process (a) Histogram equalized image (b) Result of opening with square size 2 (c) The hit places by the opening and (d) The effect on the face feature (bright in the right eye of (a) is filled). *Remark:* Image (c) is not used in the algorithm; it's just placed here to visualize the effect of the opening process for it is with structuring element of (2\*2)

```

A = [ 10   10   10   10   10   10   10   10   10   10;
      10  14  14  14   10   10  11   10  11   10;
      10  14  14  14   10   10  10  11  10  10;
      10  14  14  14   10   10  11  10  11  10;
      10  10  10  10   10   10  10  10  10  10;
      10  11  10  10   10   18  18  18  10  10;
      10  10  10  11   10   18  18  18  10  10;
      10  10  11  10   10   18  18  18  10  10;
      10  11  10  11   10   10  10  10  10  10;
      10  10  10  10   10   10  11  10  10  10];

```

The marker can be constructed easily and straightforward by subtracting a value from the original image.

```

marker =
  8   8   8   8   8   8   8   8   8   8
  8  12  12  12   8   8   9   8   9   8
  8  12  12  12   8   8   8   9   8   8
  8  12  12  12   8   8   9   8   9   8
  8   8   8   8   8   8   8   8   8   8
  8   9   8   8   8  16  16  16   8   8
  8   8   8   9   8  16  16  16   8   8
  8   8   9   8   8  16  16  16   8   8
  8   9   8   9   8   8   8   8   8   8
  8   8   8   8   8   8   9   8   8   8

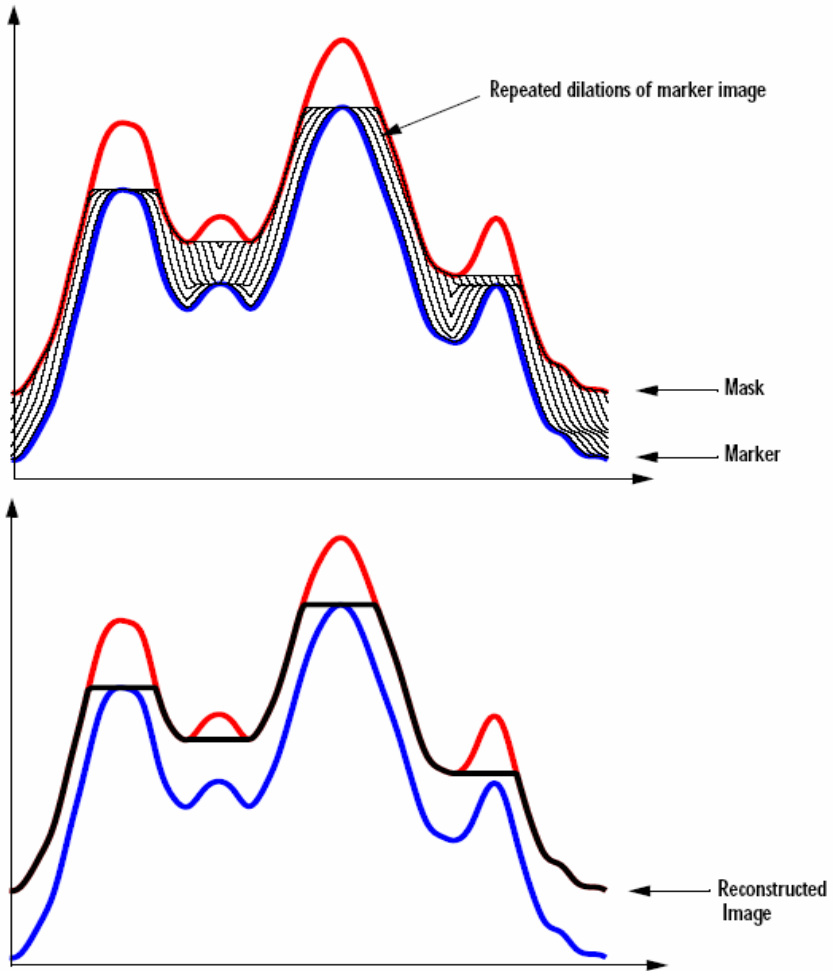
```

Hence the result will be as follows:

```

recon =
  10   10   10   10   10   10   10   10   10   10
  10  12  12  12   10   10  10  10  10  10
  10  12  12  12   10   10  10  10  10  10
  10  12  12  12   10   10  10  10  10  10
  10  10  10  10   10   10  10  10  10  10
  10  10  10  10   10  16  16  16  10  10
  10  10  10  10   10  16  16  16  10  10
  10  10  10  10   10  16  16  16  10  10
  10  10  10  10   10  10  10  10  10  10
  10  10  10  10   10  10  10  10  10  10

```



**Figure 8** Repeated Dilations of Marker Image, Constrained by Mask (MATLAB documentation)



## 5 FACE SEGMENTATION

In what follows, a novel method for locating and extracting face/head boundary is introduced. As it is mentioned earlier the preprocessing stage plays a major role in segmentation process. During this stage, the image illumination must be distributed evenly; edges must be smoothed and linked to ease the extraction of contours.

As it is shown in the system overview section, a dual processing scheme has been followed. That is the Voronoi is applied on feature points of the origin image and its complement as well. Although it doubles the time consumed but on the other hand it's found to be very efficient to ease the task of segmentation. It is especially useful when having the problem of intensity similarities between the face and its background. The author emphasizes here on the fact that each path of this dual scheme will not yield the perfect segmentation always; therefore a rule is based to chose between the outputs of the two.

### 5.1 Definition of Vd And Dt

A *Voronoi diagram* is a set of geometric elements derived from the distance relationship of geometric objects and has useful properties for engineering applications. Given a set of 2D points, the *Voronoi region* for a point  $P_i$  is defined as the set of all the points that are closer to  $P_i$  than to any other points. The intersections of the Voronoi regions for the set of points construct the Voronoi diagram. Voronoi diagrams have several geometric elements, as shown in Figure 9, such as Voronoi edges and Voronoi vertices. A Voronoi edge is a boundary line segment limiting its associated Voronoi region. A point on a Voronoi edge is associated with two input points such that each point on a Voronoi edge is equidistant

from these two input points. A Voronoi vertex is an intersection of Voronoi edges and is associated with three (or more) input points such that each Voronoi vertex is equidistant from these input points as shown in.

More formally, let a set  $S = \{p_1; \dots ; p_n\}$  of  $n$  distinct points in the plane. The Voronoi cell  $V(p_i)$  of a point  $p_i \in S$  is defined as:

$$V(p_i) := \{q \in \mathbb{R}^2 : d(p_i, q) \leq d(p_j, q), i \neq j\} \tag{12}$$

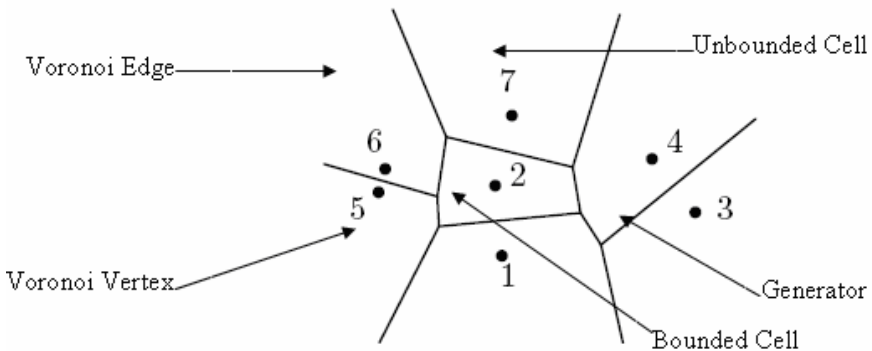
Here,  $d(p; q)$  denotes the ordinary Euclidean distance between  $p$  and  $q$

$$\sqrt{(x_p - x_q)^2 + (y_p - y_q)^2} \tag{13}$$

The Voronoi diagram  $V(S)$  of  $S$  is the family of subsets of  $\mathbb{R}^2$  consisting of the Voronoi cells and all of their intersections. The boundary of a Voronoi cell consists of Voronoi edges and Voronoi vertices. A point  $q \in \mathbb{R}^2$  is on the Voronoi edge  $e(p_i; p_j)$  if

$$d(q; p_i) = d(q; p_j) \text{ and } d(q; p_k) \geq d(q; p_i) \text{ if } k \neq i, j \tag{14}$$

(M. Berg, 2000; A. Okabe *et al.*, 2000).



**Figure 9** The Voronoi diagram of some points 1; : : : ; 7

Here the Voronoi is constructed on a finite set of distinct sites in the Euclidean space; which means that these points are not supposed to be collinear. In addition to this the minimum number of points is set to four points (sites). Voronoi diagram has several properties which are listed beneath:

- The nearest generator for all the points in the Euclidean subspace defined by  $V(\mathbf{g}_i)$  is  $\mathbf{g}_i$  (this is the definition of Voronoi diagrams).
- A Voronoi diagram is a unique tessellation.
- Every Voronoi region  $V(\mathbf{g}_i)$  is a non-empty convex polygon.
- A Voronoi polygon is infinite if and only if its generator is on the convex hull of the generators set.
- All Voronoi edges are straight lines of infinite extend if and only if the generators are collinear.
- The closest generator to  $\mathbf{g}_i$  generates one of the edges of the Voronoi region  $V(\mathbf{g}_i)$  (and hence they share the edge).
- The Voronoi diagram of a set of  $n$  point sites in the plane has at most  $2n-5$  vertices and  $3n-6$  edges.
- A point  $q$  is a vertex of  $V(S)$  iff its largest empty circle  $C(q)$  contains three or more sites on its boundary and there is no site located inside the circle (M. Berg, 2000; A. Okabe *et al.*, 2000)
- The union of all borders of the voronoi cells  $V(P_1)$ ,  $V(P_2)$ , ...,  $V(P_n)$  constitutes the Voronoi diagram of the given set  $\{P_1, P_2, \dots, P_n\}$ .

The properties of Voronoi polygons draw the attention of many researchers; as a result various algorithms have been developed to calculate the Voronoi, the reader is directed to any geometry book. Among these methods are Divide and Conquer Algorithm, Fortune's Algorithm (Seul M. *et al.*, 2000) or what Trigui A. (2002) described as *Fortune Plane Sweep* algorithm and region growing (Ahuja N., 1982).

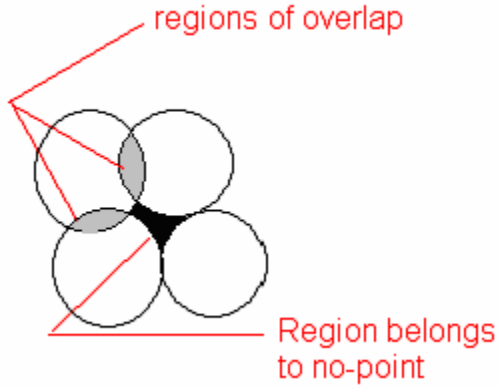
Ahuja N. (1982) was in favor to Voronoi over other techniques that bring about "Neighborhood of a point". His findings are summarized in the following tables (Table 2 and Table 3) for their importance:

**Table 2** Shows the difference between Voronoi and Fixed Radius approaches

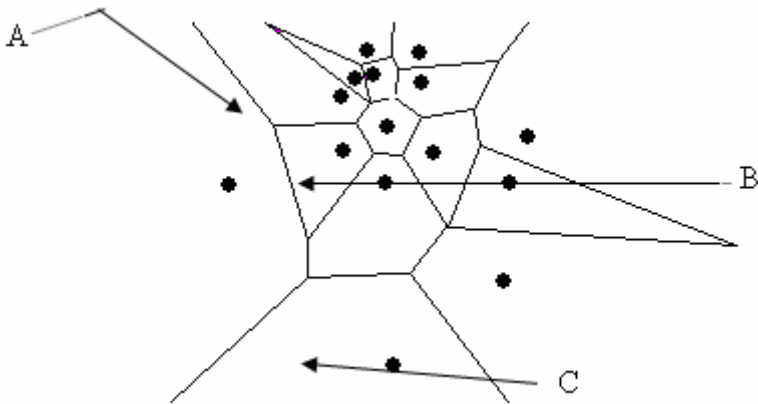
Voronoi Diagram Approach	Fixed Radius Approach
<ul style="list-style-type: none"> <li>The Voronoi tessellation assigns each region of the plan to the neighbor of one and only one point.</li> </ul>	<ul style="list-style-type: none"> <li>The set of circles of radius <math>R</math> centered at a given set of points will, in general, have regions of overlap. There will be also regions surrounded by various neighborhoods but not included in any figure 10</li> </ul>

**Table 3** shows the difference between Voronoi and K.NN approaches

Voronoi Diagram Approach	K-Nearest Neighbor Approach
<p>Some point's neighbors may be far away from it than some other points which are not Voronoi neighbors figure 11.</p>	<ul style="list-style-type: none"> <li>All KNN are assumed to be the nearest neighbors.</li> </ul>

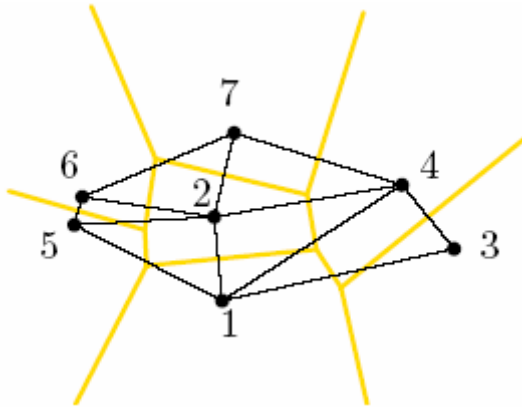


**Figure 10** The fixed radius weaknesses

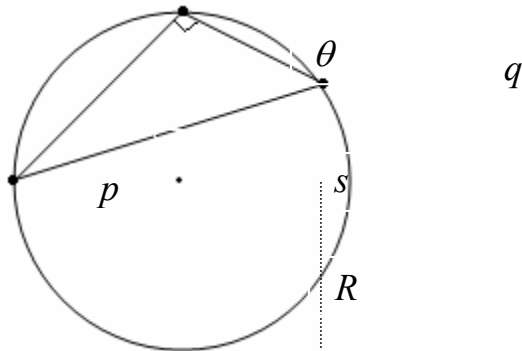


**Figure 11** VD point's neighbors; notice that C is a neighbor point to B but not A although A is nearer to B than does C

The *dual* tessellation of the Voronoi Diagram (VD) is known as Delaunay Triangulations (DT) Figure 12 in the following sense: vertices in the Voronoi diagram correspond to faces in the Delaunay triangulation, while Voronoi cells correspond with vertices of *DT*. Once the Voronoi diagram for a set of points is constructed it is a simple matter to produce Delaunay triangulation by connecting any two sites whose Voronoi polygons share an edge. Or in another way, let  $\mathbf{P}$  be a circle free set. Three points  $P_i, P_j, P_k$  of  $\mathbf{P}$  define a Delaunay triangle iff there is not further point of  $\mathbf{P}$  in the interior of the circle which is circumscribed to the triangle  $P_i, P_j, P_k$  and its center lies on a Voronoi vertex, Figure 13 depicts this notion.



**Figure 12** The Delaunay triangulation of the points introduced in Figure 9



**Figure 13**  $pqr$  define a Delaunay triangle when the center ( $s$ ) of the circle circumscribed to  $pqr$  is a Voronoi vertex

Delaunay Triangulation as far as geometry is concern has some properties which are listed below:

- The external triangle edges of the Delaunay tessellation is the convex hull containing the given set of sites.
- It is the dual tessellation of the Voronoi diagram.
- There is no site located inside the circle passing through the vertices of any Delaunay triangle.

## 5.2 Application of Vd/Dt

There is a wide variety of geometric problems that can be naturally solved using spatial tessellations which explains why they are so widely used in many different applications. More specifically,

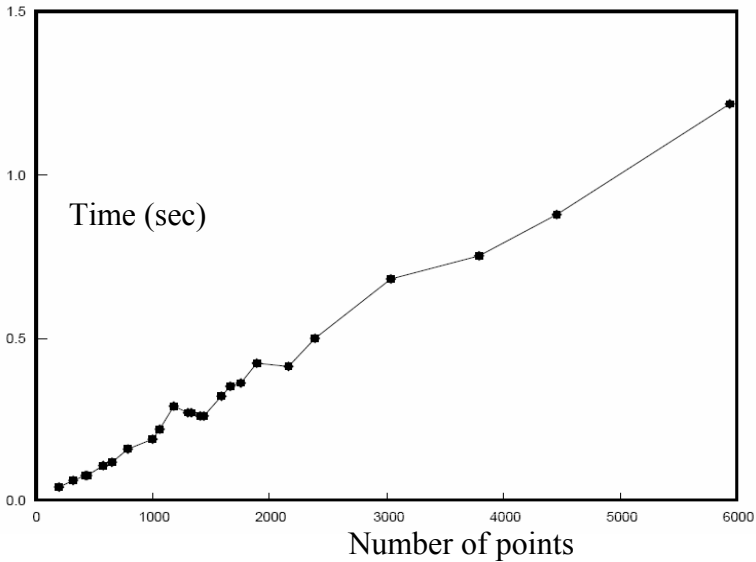


given a set of points, some of the problems that the Voronoi diagrams can solve, efficiently, are:

1. The closest pair problem.
2. The all k nearest neighbors' problem.
3. The all nearest neighbors' problem.
4. The fixed radius near neighbors.
5. The nearest-search problem (*the "Knuth's Post Office problem"*).
6. The largest empty circle problem (*the "Toxic Waste Bag problem"*).

The proposed approach is based ultimately on Voronoi Diagram (VD) technique, which is a versatile geometric structure (Costa L. and Cesar R., 2001; Ahuja N., 1981, Xiao Y. and Yan H., 2002). In spite of its usefulness in various fields ranging from Remote Sensing to Robot Navigation, this method was largely forgotten in applications in image segmentation until Suhail M. et al., (2002) rekindled interest in it. Their approach was based on extracting feature points, which they used to define Voronoi cells' centers for image segmentation. The researcher proposed method although it is inline with theirs in using VD; however the proposed way and target are completely different.

Researchers tend to apply VD on the image itself (after binarizing it and getting its edges). This is usually time consuming, because as stated Voronoi construction has a time complexity of  $O(n \log n)$  which means it has a direct relationship (linear) with the number of generators figure 14. Therefore the idea came to apply VD not on images but on selected few points where ( $n \leq 255$ ), which are but the intensity frequencies.



**Figure 14** CPU times for computing Delaunay graphs on SUN SPARCstation 10/20 (Reinelt G., 1994)

So, VD is constructed from features points (generators) which result from gray intensity values summations (frequency).

$$Val(x) = \sum_{i=1}^m \sum_{j=1}^n f(x,y)_k \quad (15)$$

Where

$k$  is the range of the intensity values,  $k = \{0, 1, 2, \dots, 255\}$ , and  $f(x, y)$  denotes pixel location in the host image.

The method has the spirit of dynamic Thresholding used for segmentation while it differs in terms of *divide and merge decision*. Human faces have a special color distribution that differs significantly from that one of the background. The VD is used to

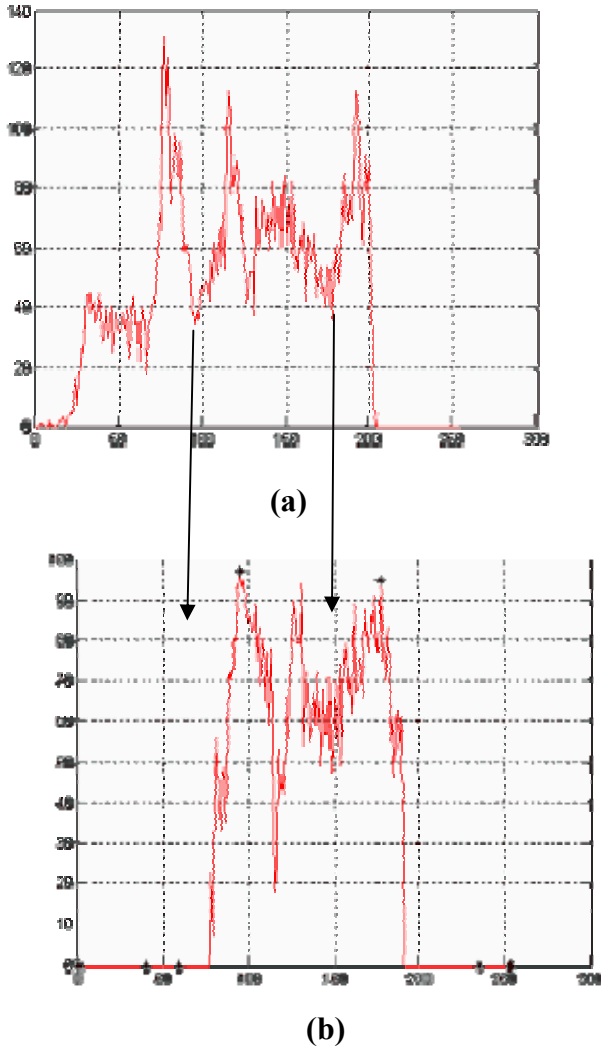
construct the Delaunay Triangulations (DT). The outer boundary of DT is simply the convex hull of the set of the featured points. The convex hull  $CH(S)$  of a set of points is the smallest convex set containing these points. It is a convenient means for representing point sets. If the point set is dense then the convex hull may very well reflect its shape. Large and diverse instances ones usually exhibit several clusters. Building the convex hull of these clusters can result in a concise representation of the whole point set still exhibiting many of its geometric properties. Many algorithms were introduced to build the  $CH(S)$  e.g.: Graham's Scan, Divide and Conquer, Throw-Away Principles and Convex Hulls from Maximal Vectors.

Hence by getting the  $CH(S)$  the global two maxima are obtained. In order to get the minima that fall between these two peaks, a new set of points are generated using the following steps:

- Generate the host image histogram
- Set all the points below the first peak to zeros, and set all the points beyond the second peak to zeros.
- Set all points that are equal to zeros to be equal to the  $\text{argmax}(\text{peak1}, \text{peak2})$ .
- Derive the new points by:

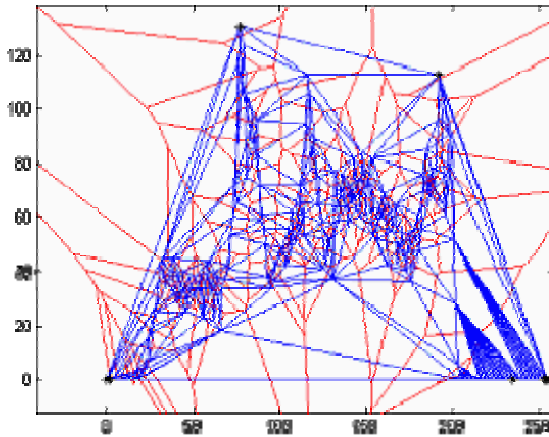
$$Val_{new}(x) = ||(val(x) - \max(val(x)))|| \quad (16)$$

This will expose the minima points to be part of the convex hull constructed by DT Figure 15.



**Figure 15** Histogram manipulation. (a) Gray level frequency (Getting maxima) and (b) showing local minima

It should be born in mind that the outer layer of DT contains vertices which are but some selected Voronoi (VD) generators, in another word the vertices forming the *CH* are what the study is interested in. In this case these vertices give a direct access to their respective gray values Figure 16.



**Figure 16** Delaunay Triangulation being imposed on the histogram of figure 15a

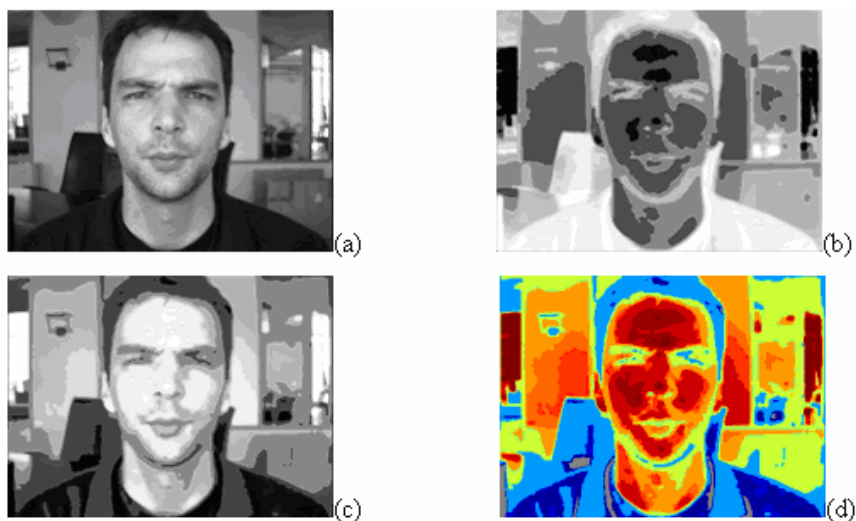
These unique feature points are then sorted in ascending order to form ranges from which decision of merging and splitting regions is based. The following paradigm is applied:

```

Input: Host gray scale image  $\Psi$ 
Input: Vector generated by the previous described procedure  $\Omega$ 
Initialize a new vector  $d = []$ 
Set all pixels in  $\Psi$  smaller than  $\Omega(1,1)$  to black (0)
Set all pixels in  $\Psi$  greater than  $\Omega(\text{length}(\Omega(:)),1)$  to black (0)
for  $i = 1$  to  $\text{length}(\Omega(:)) - 1$  do
    if  $i == 1$  then
        set all ( $\Psi \geq \Omega(i)$  AND  $\Psi \leq \Omega(i + 1)$ ) to  $\Omega(i + 1)$ 
         $d = [d ; \Omega(i + 1)]$ 
    else
        set all ( $\Psi > \Omega(i)$  AND  $\Psi \leq \Omega(i + 1)$ ) to  $\Omega(i + 1)$ 
         $d = [d ; \Omega(i + 1)]$ 
    end if
end for
Output segmented gray scale image  $\Psi_{\text{Segmented}}$ 

```

An example of the application of the above pseudo code is shown in Figure 17.



**Figure 17** Segmentation using the proposed method. (a) Original image (b) Segmentation based on the image complement (c) A direct segmentation and (d) (c)-With colormap

Segmentation carries along various advantages if it is efficiently describing the host image. It is one way of smoothing and enhancing the edge detection because of clustering the image into different regions where the intensity value assigned to each region is dissimilar to other clusters' ones. It is another way to reduce intensity values as it is clear from the experiment made on one of the BioID faces (table 4). The re-sampled result has the minimum gray values while preserving objects shapes in the image. This helps in reducing the time consumed in searching for the best fit ellipse in these objects.

**Table 4** Shows the benefit of segmentation in gray values' reduction (re-sampling)

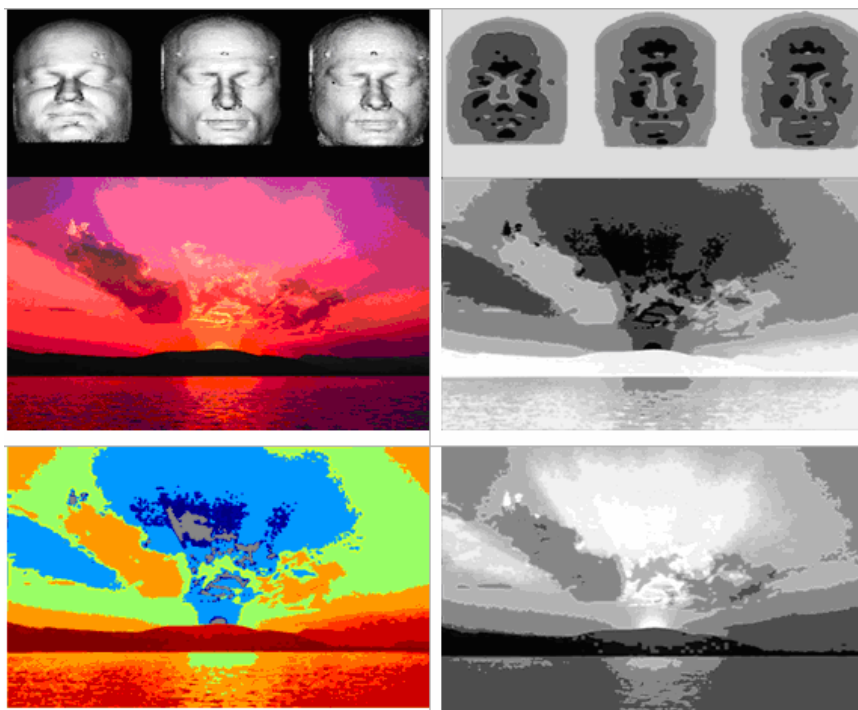
Gray values in host image	After preprocessing	After segmentation	
252 Values	59 Values	Direct	Complement
		20 Values	19 Values

To conclude this section, the author stresses that although the scope of this research is bounded by face database with gray scale images, however the designed system can handle the segmentation of other types of images as it is clear from the next example figure 18.

A second fact is that the Voronoi based image segmentation can be extended to RGB images without converting it into grayscale, that is done by working on the RGB map only and treating each color matrix as a separated input to the segmentation process.

Finally, a dual processing has been used namely for direct image and for its complement because according to the author observation; the complement segmentation in certain cases tends to be bias to bright pixels; while direct segmentation is found in other cases to be bias to dark pixels.





**Figure 18** Application of the algorithm on WWW images. The method can be extended to non face images. Up to this point the algorithm is general

### 5.3 Extraction of Probable Face Region

The input of this stage is the segmented image. The image is represented by the minimum number of values ( $\leq 20$  values). A brute force is applied to each segment to yield labeled connected regions. When filling all regions' holes in the image, a total -or in worst case majority -of head region (face and neck) is formed. Small regions are of no interest to the algorithm e.g.: eliminating of small objects in the background, therefore the study eliminates

any block less than an arbitrary value by disregarding all blocks less than (60\*60); many authors in the literature have chosen this size of window as the minimum size of the face. A labeled image is defined as:

$$\hat{h}(x, y) = \forall f(x, y) \in G_k \quad (17)$$

where

$G_k$  is a labeled region, and  $k$  is the number of segments.

Hence, each binary segment  $\delta_n$  is defined as:

$$\delta_n = \begin{cases} 1, & \text{if } f(x, y) \in G_n \\ 0, & \text{otherwise} \end{cases} \quad (18)$$

It has been found that faces show elliptical shapes with minor variations in eccentricity from one person to another. The ramp transition characterizing the face intensity will give a hand, because in general it forms a contour of *on* pixels surrounding *off* pixels, which means when filling it the exact shape of the face is obtained. Thus, the voting phase can be made based on detection of ellipses.

A major problem researchers are fighting with occurs when the contrast between intensity values of faces and the values at their background is low. This obstacle has been reported clearly by (Kwolek B., 2003; Hsu R. and Abdel-Mottaleb M., 2002) in their skin tone detection in color images. According to Kwolek B. (2003), if an input image contains a background whose color is similar to the color of the true skin region, the labeled segments in the image can be of any arbitrary shape and not necessarily elliptical and therefore the hypothesized face location is not reliable at this stage. This fact was also admitted by (Rizon M. and Kawaguchi T., 2000) when they said "...But, when the intensity difference between background and skin-region of the face is small, it is difficult to correctly extract the face region ..."

To cope with this situation, the distance transformation (DisT) is introduced to separate the face region from its background. The function operates on the Euclidean distance between two given points  $(x_1, y_1)$  and  $(x_2, y_2)$ , where:

$$DisT = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (19)$$

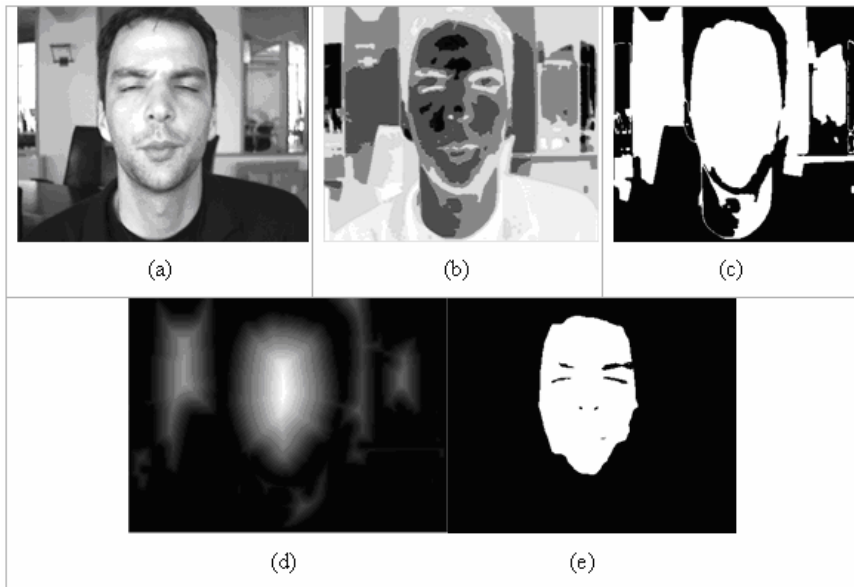
Then a new image is chosen obtained by the following:

$$new\_image = (DisT \geq Th) \quad (20)$$

where

$Th$  denotes a threshold value. The resulted image is dilated with a structuring element of size 3 in order to recover back the shape distorted by the DisT function.

This works fine while the bridge connecting the face to the background is not wide, which is a very rare case because the presence of the head hair will prevent that and the merging if it occurs it will be in the region of face sides Figure 19.



**Figure 19** Applying Distance Transformation to cut the bridge merging the face with the background. (a) Original image (b) Segmented (complement) (c) filled segment (d) Its DisT and (e) Background eliminated

As a vivid example on the DisT in binary images and its concept, let's have a small matrix (5\*5) and name it bw:

```

bw =  0  0  0  0  0
      0  1  0  0  0
      0  0  0  0  0
      0  0  0  1  0
      0  0  0  0  0
    
```

The associated values of the DisT from the *off* pixels to the nearest *on* pixel is denoted by the new matrix DisT:

DisT =	1.4142	1.0000	1.4142	2.2361	3.1623
	1.0000	0	1.0000	2.0000	2.2361
	1.4142	1.0000	1.4142	1.0000	1.4142
	2.2361	2.0000	1.0000	0	1.0000
	3.1623	2.2361	1.4142	1.0000	1.4142

Since horizontal and vertical pixels of 8 neighbor point have the distance of 1 unit, the diagonal direction however has the distance

$$\text{of } \sqrt{1^2 + 1^2} = \sqrt{2} = 1.4142.$$

After stepping out from this probable problem, three main criterions for a possible rule can be used for estimating and evaluating the *Region Of Interest (ROI)*, which are:

- **Rule 1: Ellipse Fitting**

This is a well known strategy for face estimation and segmentation. The connected components given by the above stage will be fit into an ellipse model. An ellipse is ultimately characterized by its center (X0, Y0), Orientation ( $\theta$ ), minor axis (a) and major axis (b). The center is but the center of mass, while  $\theta$  is determined using the least moment of inertia:

$$\theta = 1/2 \arctan(2M_{1,1}(M_{2,0} - M_{2,0})) \quad (21)$$

where

$M_{i,j}$  denotes the central moments of  $\hat{h}(x,y)$ .

$$I_{\min} = \sum_{(x,y) \in C} [(x-X0) \cos \theta - (y-Y0) \sin \theta]^2 \quad (22)$$

$$I_{\max} = \sum_{(x,y) \in C} [(x-X0) \sin \theta - (y-Y0) \cos \theta]^2 \quad (23)$$

$$a = (4/\pi)^{1/4} [(I_{\max})^3 / I_{\min}]^{1/8} \quad (24)$$

$$\text{and } b = (4/\pi)^{1/4} [(I_{\min})^3 / I_{\max}]^{1/8} \quad (25)$$

Assessing how well the  $\hat{h}(x, y)$  is approximated by its constructed ellipse the new distances measure is defined between  $\hat{h}(x, y)$  and the best fit ellipse as follows:

$$\Phi_i = P_{inside} / M_{0,0} \quad (26)$$

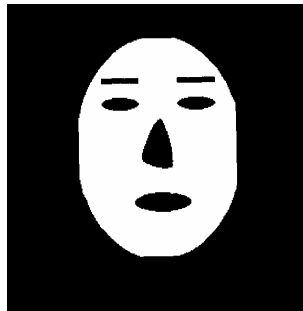
$$P_{outside} / M_{0,0} \quad (27)$$

Where

$P_{inside}$  denotes the sum of background pixels within the ellipse,  $P_{outside}$  is the sum of  $\hat{h}(x, y)$  pixels lying outside the ellipse (Haddadnia J. *et al.*, 2002), and  $M_{0,0}$  is the total area of  $\hat{h}(x, y)$ .

- **Rule 2: Cross Correlation**

This is an acknowledged strong method in detecting objects and reporting the level of much. It is well know that in binary images Cross Correlation has less computational complexity comparing to gray scale. Therefore the study came up with the model depicted in Figure 20.



**Figure 20** Template model to vote for face blocks

- **Rule3:** Euler Number

A binarized face region will expose face features as holes because of their low intensity. This helps in judging a probable face region in terms of its Euler number. Therefore, it is safe to say that a face would consist of at least two holes. To determine that, the following equation is used:

$$E = C - H \quad (28)$$

where

$E$  is the Euler number,  $C$  is the number of connected components and  $H$  is equal to the number of holes. Because only one segment is present at a given time, the value  $C$  is equal to one. Thus,

$$H = 1 - E \quad (30)$$

In this research **Rule2** and **Rule3** are selected to judge any segment whether it contains a face.

## REFERENCE

- COSTA L. AND CESAR R., 2001. Shape Analysis and Classification, CRC Press, USA.
- DOWDALL J., PAVLIDIS I. AND BEBIS G., 2003. Face detection in the near-IR spectrum, *Elsevier Sci. Image Vision Comput.* vol.21, pp. 565–578.
- FENG G.C. AND YUEN P.C., 2001. ‘Multi-cues eye detection on gray intensity image’, *Elsevier Sci. Pattern Recognition* vol.34, no.5, pp. 1033–1046.

- H. Dang and D. Claussi, (2004). Unsupervised image segmentation using a simple MRF model with a new implementation scheme. *Pattern Recognition* 37 (2004) 2323-2335
- HADDADNIA J., AHMADI M. AND FAEZ K., An efficient method for recognition of human faces using higher orders pseudo Zernike moment invariant, *IEEE Proc. Fifth Int. Conf. Autom. Face Gesture Recognition*, pp. 315–320.
- HSU R. AND ABDEL-MOTTALEB M., Face detection in color images, *IEEE Trans. Pattern Anal. Mach. Intell.* Vol.24, no.5, pp. 696–706.
- KWOLEK B., 2003. Face tracking system based on color, stereovision and elliptical shape features, *IEEE Proc. IEEE Conf. Adv. Video Signal Based Surveillance*, pp. 21–26.
- N. Ahuja.(1982). Dot Pattern Processing Using voronoi Neighborhoods. *IEEE Trans on Pattern Recognition and Machine Intelligence.* PAMI-4.3. 336-343
- RIZON M. AND KAWAGUCHI T., Automatic eye detection using intensity and edge information, *Proc. IEEE TENCON* vol.2, pp. 415–420.
- SEUL M., O’GORMAN L. AND SAMMON M., 2000. *Practical Algorithms for Image Processing*, Cambridge University Press, USA.
- XIAO Y. AND YAN H., 2002. ‘Facial Feature Location with Delaunay Triangulation and Voronoi Diagram Calculation’. Australian Computer Society, Inc. The Pan-Sydney Area Workshop on Visual Information Processing (VIP2001).
- Z. Tu and S. Zhu, (2002). Image Segmentation by Data-Driven Markov Chain Monte Carlo. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* Vol.24, No.5 May 2002, pp 657-673