EE569
Term Paper

Watershed transform

Yue Wu
6733-3385-34
wuyue@usc.edu
Term paper

Task:
Preparation of your term project report
Step 1: Please pick up an image processing topic of interest to you.
Step 2: Please find an algorithm from a journal/conference paper to tackle the problem. You need to digest the material well so that you know how to implement.
Step 3: Please show the application of this algorithm to real world images.
Step 4 (optional): You may show the improvement of the algorithm in Step 2 and discuss reasons for the improvement.

The topic I choose:
Watershed transform. In geography, a watershed is the area of land where all of the water that is under it or drains off of it goes into the same place. In image processing, a watershed is a method to divide an image. It is a segmentation algorithm based on the topology of an image.
The basic idea of watershed is: convert an original image into a gradient-level image. Where the pixel has the lowest gradient-level is seen as a basin. Starting fill up water from the basins, with the increase of the gradient-level, the pixels will be immersed level by level with water. And, at points where water coming from different basins would meet, dams are built. The dams are called as watersheds. When the water level has reached the highest level in the image, the process is stopped. As a result, the image is partitioned into regions or basins separated by watersheds.

Procedure:
Since the watershed concers the changes of the grey-levels of pixels instead of the grey-level itself
1. Turn the original image (N*N) into a gradient image(N*N), gradient can express the extent of changes of the grey-levels.
2. Extend the gradient-level, so it has a scale level from 0 to 255.
3. Set gradient-levels that are small (equal 0,1,2) to 'minimas'.
4. Put each pixel into its minima

In step four, I used 'Algorithmic definition by immersion'.
In the gradient-level image, if the gradient (g) is smaller or equal to two, than set the pixel as a minima, every minima actually present that there will be an area; then
consider the condition that g equals three. Find the minima distance between the pixel (g=3) and the selected minimas. If a pixel has only one minima distance or the equal minima distance come from the same minimas, then the pixel is classified to that minima, they belong to one area. If a pixel has a equal minima distance with two different minimas, then the pixel is set as a 'Watershed', which means that the pixel may belong to none of the areas;
Then consider the condition that pixels have g equals 4 and the pixels that is set as watersheds in the past stage. Use the same method to classify the pixels into their areas. They may belong to one minima, and can be classified to an area, or they may be considered as watershed;
Increase g, until g equals to the highest level. Then I get an image with watersheds that can divide the image to several parts.

![Figure 0. Logical flow chart](image-url)
**Improvement:**
In the watershed transform above, sometimes there occurs the over segmentation. In order to reduce the over segmentation, I can smooth the image, reduce the noise, so that erase the false minimas what actually are noises. So I add a lowpass filter before doing the watershed in a noisy image (pepper_uni.raw). Another way to improve over segmentation is to select a rectangle of minimas manually. So there will not be unnecessary minimas.

**Results:**

![Figure 1](image1.png)
(a). The original image of pepper
(b). The gradient-level of pepper

![Figure 2](image2.png)
(a). The original image of pepper_uni
(b). Gradient-level of pepper_uni
Figure 3.

(a) The denoised image of pepper_uni (b). gradient-level of denoised pepper_uni

Figure 4. watershed of pepper

Figure 5. watershed of pepper_uni
Figure 6. Denoised pepper_uni
(a) with lower minima          (b) with higher minima

Figure 7. Topology method
(a). The chosen minima areas  (b). Watershed begin with the chosen minima areas

Discussion:
In the minima selection, the way to select minima is important. In a relatively smooth image, if value the minimas are very low, then there will be too few minimas and areas, thus, can not clearly divide two different areas. While on the other hand, in a relatively sharp image, if the values of minimas are relatively high, then there will be too many minimas and areas, as a result, it will cause over segmentation.

In the pepper_uni image, we can see from the result. The gradient image of pepper_uni is vague, and it is hard to see the edge lines of the image, which means the gradient-level of the image is very close to each other. In figure 5., even if the value of minima a choose is very high (g=23), it does not have many areas.

In a noisy image, after applying a lowpass filter, the noises have been decreased, and it has a clearer gradient image. While the image has a relatively higher gradient-level, so the image needs a higher value of minimas so separate it. In figure 6(a), set g equals or less then 1 as a minima; In figure 6(b), set g equals or less than 11 as a minima.

I also tried other methods. In figure 7, the algorithm is: first, if a pixel is the smallest of its neighbouring, then it is a minima. Second, if a pixel has at least one neighbour that is higher than it is a watershed, then this pixel is a watershed; if a pixel's neighbours which are higher than it are all belong to one area, then the pixel belongs to them; if a pixel has more than one neighbours and they belong to
different areas, then the pixel is a watershed. However, this method has too many minimas, and thus has a severe over segmentation.

In figure 8., I manually choose some minima areas to reduce the over segmentation. (a) is the areas I choose according to the gradient-level image (Figure.1(b)), and (b) is the result of using chosen minimas. It avoids the over segmentation.

In watershed, there are two ways to choose neighbours: 4-connection and 8-connection. In my term paper, I used 4-connection. I think 8-connection will give a better outcome, and the watersheds will be more clear, because the 8-connection consider 8 neighbours instead of 4 neighbours, more neighbours are considered.