
Water Shed 알고리즘을 이용한 공 찾기

유지철* · 김영길*

Finding the Ball with WaterShed Algorithm

Ji-Chul Yu* · Young-kil Kim*

요 약

본 논문은 Water Shed 알고리즘을 이용하여 여러 공들 중에 숨어있는 공을 찾는 것으로서 실제 적인 목적은 Water Shed 알고리즘의 한계를 짚어보고 분석해 보고자 함이다. 이로써 세포 분석이나 농산물 관련된 분야에서 사용되던 기존의 Water Shed의 문제점을 보완하는 계기가 될 것이다. 본 연구는 Visual Studio C++을 사용하여 구현해 보았으며, Distance Transform 및 Labelling 등의 알고리즘이 추가로 사용되었다.

ABSTRACT

This is the paper for finding the hidden ball with the Water Shed Algorithm, and for analyzing the weakness. So, With this paper, Other institutions or organizations can improve their research skill. For this research, Visual Studio C++ is used, and some kinds of Algorithm is added to the software such as Distance Transform, Labelling which we make.

키워드

Water-Shed, Distance Transform, Labelling, Distance Map, Minima

I. 서 론

본 논문은 단순히 공을 찾는 것이 목적이 아니라 그림 1에서 보는 것처럼 38개의 공들 중에 34개의 공은 찾았지만 2개의 공은 결국 찾지 못한 것을 알 수 있는데, 이렇게 찾지 못한 2개의 공을 찾지 못한 원인을 분석하고 찾지 못한 공을 찾기 위해, 그 대안을 모색하고 더 나은 영상 분석을 하기 위해서 이다.

본 논문의 Water-Shed 알고리즘에 대하여 간단히 서술하면, Water-Shed는 영상의 gray level 값을 높기로 생각하여 최소 영역에서부터 차츰 물을 채워 나가듯이 영역을 합쳐나가는데 최종적으로 하나의 테두리로 둘러

싸인 부분을 균일 영역으로 판단한다.[1] 이때 물이 차오름을 등고선 적으로 나타내기 위해 사용된 알고리즘을 Distance Transform 알고리즘이라고 하는데, 그림 2에서 보는 것과 같이 각 화소로부터 가장 가운데 '0'이 아닌 화소의 거리로 변환하여 꼭 지점의 거리나 위치를 계산하는 것이다. 이 외에 또 다른 방법으론, 각각의 '0'이 아닌 물체들 사이의 거리를 구하는 방법인데, 본 논문에선 전자의 방법으로 Distance Transform 알고리즘을 사용하였다.[2] 그리고 그것에 대한 간단한 예를 그림 2에서 표현해 보았다.

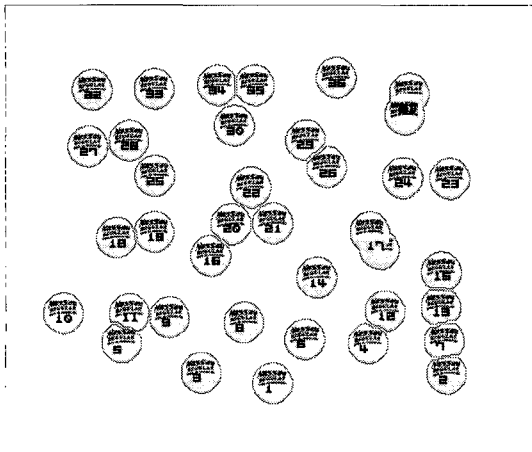


그림 1. 공 찾기를 위한 영상
Fig. 1. Image for finding the ball

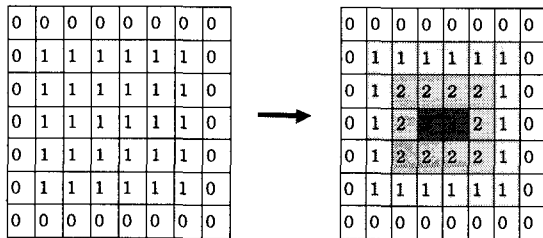


그림 2. Distance Transform
Fig. 2. Distance Transform

이렇게 구해진 Distance Map을 이용하여 두 물체 사이의 유역능선을 찾아낸다. 유역능선이란 그림3에서 보는 것처럼 봉우리 사이의 들어간 부분을 나타낸다.

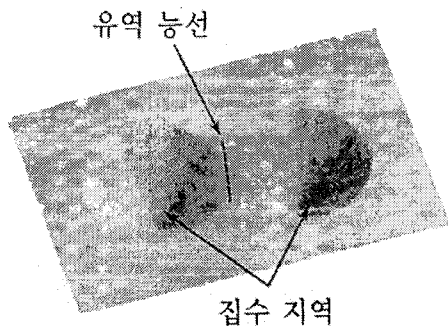


그림 3. 유역 능선[3]
Fig. 3. Watershed ridge line

II. 본 론

위에서 서술된 이론을 가지고 영상처리 알고리즘을 구현 하여 공의 개수를 개산하여 볼 경우 거의 완벽할 정도의 정확도를 보였다. 하지만 몇 층으로 싸여 있는 공의 무더기일 경우엔 정확도가 많이 떨어짐을 그림1에서 확인할 수 있었다. 그럼 왜 정확도가 떨어지는지에 대하여 분석해보고, 그 결과를 바탕으로 문제해결 방법을 제시 하도록 하겠다.

우선, 그림 1에서처럼 인식하지 못한 공과 같이 몇 가지의 경우를 포토샵을 통하여 구현 한 다음 영상처리 알고리즘을 사용하여 그 결과를 확인해 보았다. 그림 4에서 알 수 있듯이 두 물체가 거의 포개어져 있을 경우, 그림 1에서 구분되지 못한 경우와 같이 2개의 공을 하나의 공으로 인식 하고 있음을 알 수 있다. 이에 대한 원인은 다음과 같다.

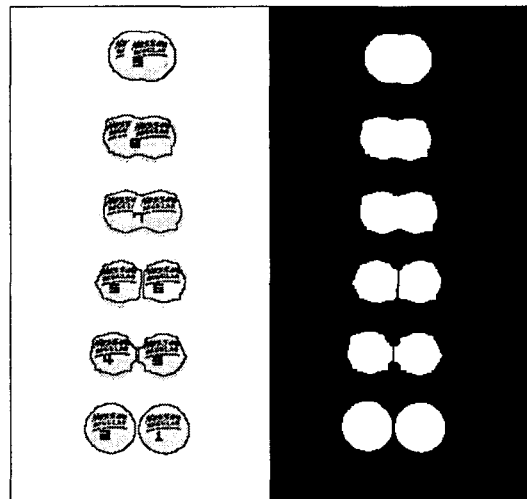


그림 4. 공의 간격에 따른 인식
Fig. 4. Recognition as varying distance

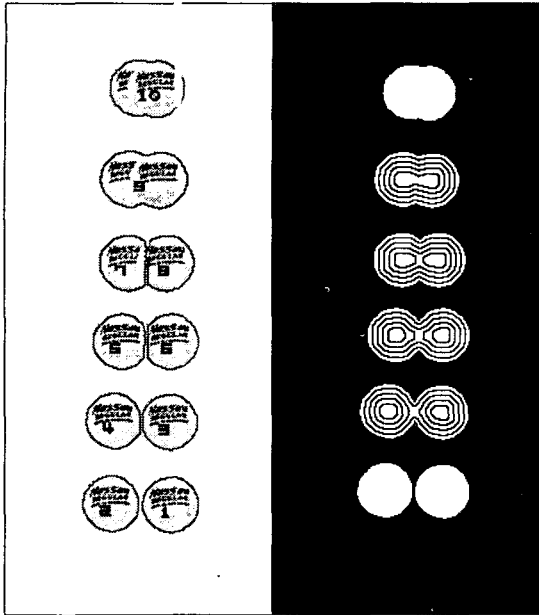


그림 5. Distance Map
Fig. 5. Distance Map

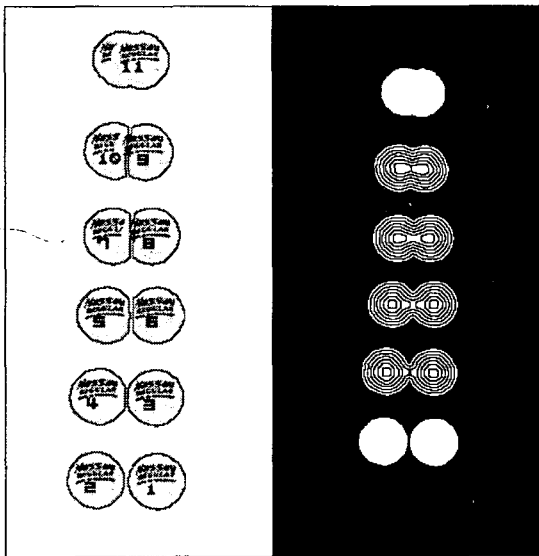


그림 6. Distance Map
Fig. 6 Distance Map

위의 그림 6의 Distance Map의 등고선 간격이 그림 5의 등고선 간격 보다 조밀하다. 결과를 보면 Map의 등고

선 간격이 조밀 한 그림 6의 결과에서 2번째에 있는 공이 서로 분리 되어 나타나는 것을 알 수 있는데, 다시 말해, 그림 3에서 설명된 집수 지역이 그림 5에서는 두 개로 구분 되지 못하여 유역능선을 만들지 못했고, 반면에 그림 6에선 집수지역이 두개로 구별 되어 유역능선을 찾게 된 것이다.

그러면, 그림 6의 첫 번째 공도 Map의 간격을 높이면 찾아 낼 수 있지 않을까? 하지만 Map의 간격을 더 이상 줄이게 된다면 4번째와 5번째의 집수지역에 영향을 준다. 결국에는 그림 7의 결과처럼(오른쪽의 결과는 왼쪽의 결과 보다 Map 간격을 더 좁힌 경우에 해당 된다.) 역행되는 결과가 나오게 된다.

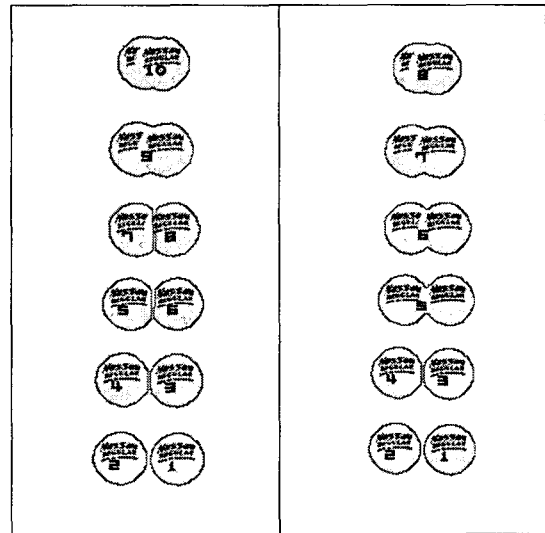


그림 7. Map 간격을 더 좁혔을 때
Fig. 7 Case of narrow distance

이런 결과가 나오게 된 이유를 등고선에 적용하여 설명하면 첫 번째 등고선이 산봉우리의 가장 높은 부분이라고 하면 등고선의 간격이 줄어들에 따라 첫 번째 등고선 영역도 작아지게 된다. 이때 줄어들 정도가 커지게 되면 첫 번째 등고선이 없어지게 되고 결국 두 번째 등고선 부분이 첫 번째 등고선 영역의 역할을 하게 된다. 크기가 있는 물체에 적용할 때에는 어느 정도 효과가 있지만 작은 물체에 적용할 때에는 등고선 사이의 간격이 너무 좁고 약간의 변화에도 크게 반응하게 된다. 예를 들면, 쌀과 같은 작은 물체에 적용할 경우 등고선과 등고선 사이

의 간격은 더 좁기 때문에, 큰 물체에서와 같이 두 물체가 분리되어야 하지만 그림7의 오른 쪽과 같이 붙은 모습이 되게 된다. 다시 말해 물체의 크기가 작을 경우 등고선의 간격을 줄이는 것만으로는 한계가 있다는 것이다.

이처럼 Distance Map의 경우 어느 정도까지의 사용에는 상당히 합당한 결과를 얻을 수 있지만, 작은 물체 즉, 더욱 정밀함이 요구되는 물체에는 두 개로 나누어 인식하는데 한계가 있었음을 알 수 있었다. 그림 이 알고리즘의 한계는 어떻게 되고 정확도를 높이기 위해선 어떤 방법이 있는지 알아보도록 하겠다.

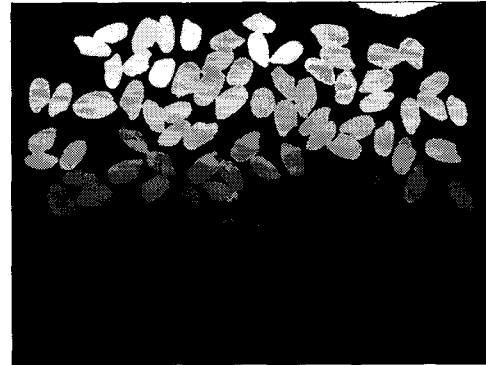


그림 8. 쌀을 Labelling 한 결과
Fig. 8. Labeled image of rice

III. 실험

표 1. 프로그램 블록도
Table. 1 Block diagram of program

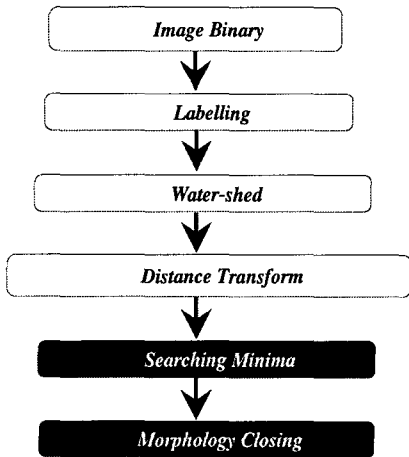


표 1은 실제 실험을 위해 만든 프로그램의 블록도 이다. 맨 처음 영상을 2진화 시킨 다음 Labelling을 하게 된다. Labelling이란 쉽게 말해 사람마다 이름이 있듯이 각각의 객체에 숫자를 부여 하는 것이다. 그림 8은 쌀을 가지고 Labelling한 결과를 나타낸 것이다. 떨어져 있는 숫자 대신 각각의 떨어져있는 객체의 색깔 값을 달리 해주었다.[4]

이렇게 구해진 객체를 Water-Shed와 Distance Transform 알고리즘을 거치면서 유역능선을 추출하는 식으로 경계선을 추출해 낸다.[5][6]

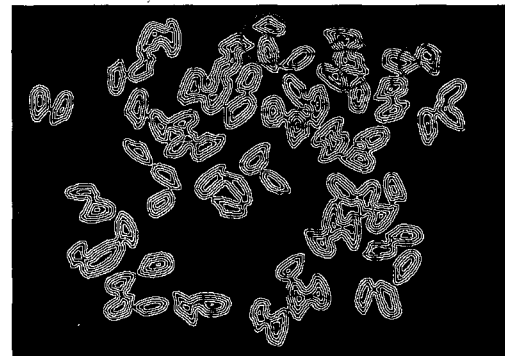


그림 9. Water-Shed와 Distance Transform을 거친 영상
Fig. 9. Water-Shed and Distance Transform

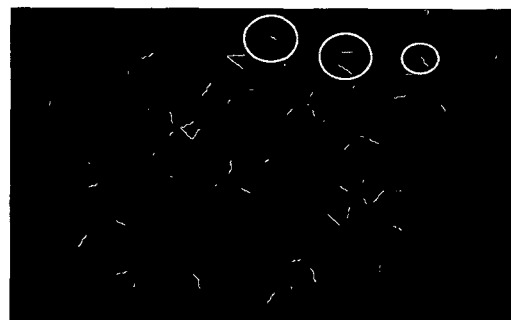


그림 10. 찾아낸 유역 능선
Fig. 10. Watershed ridge lines



그림 11. Binary 과정만 끝낸 영상
Fig. 11. Binary image



그림 12. Distance Transform까지 끝낸 영상[7]
Fig. 12. Distance Transform

그림 8,9,10,11에서 12까지가 객체를 분리하기 위한 기존의 방법이다. 하지만 본 논문에서는 Searching Minima과정과 Closing 과정을 추가로 처리 해 주어 다른 결과를 얻어내었다.

그림 4나 5,7의 그림에서 맨 첫 번째 공을 서서히 서로 떨어뜨려 두 개의 공으로 인식하는 바로 그 순간을 측정 하여 과연 공이, 다시 말해, 두 물체가 몇 %이상 까지 겹치면 각각의 물체로 인식하는 지, 못하는 지를 관찰해 보았다. 그림 13은 두 물체의 간격을 서서히 벌리면서 포개 짐의 비율을 측정 한 그림이다.

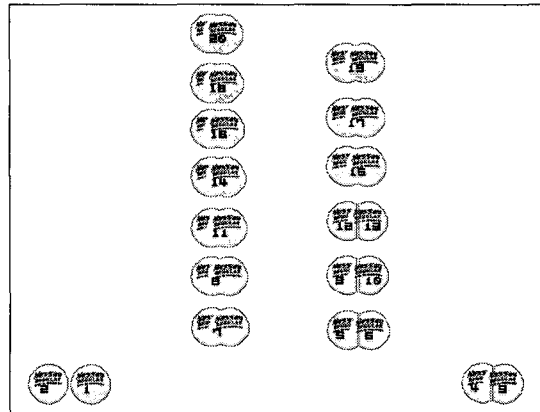


그림 13. 포개어짐 정도에 따른 결과를 알아보기 위한 그림

Fig. 13. Image of Superimposed balls

표 2. 포개어짐 정도 율
Table. 2 superimposed rate

공번호	19	17	15	12,13	9,10	5,6
비율	53%	50.3%	45.5%	44.8%	44.1%	43.4%

위의 표 2가 의미하는 것은 공의 절반을 가릴 경우, 다시 말해, 한 물체가 다른 물체의 반 이상을 가릴 경우 각각의 객체로 인식하지 못한다는 것이다. 이러한 점이 Water-Shed의 한계라 할 수 있다.

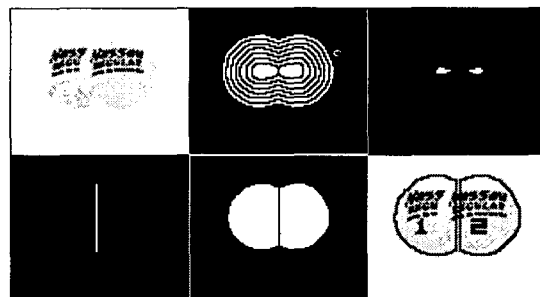


그림 14. 공 찾기까지의 일련의 과정
Fig. 14. Process of Finding the ball

그러므로 Water-Shed를 최대한 잘 활용하기 위해서는 물체가 반 이상 겹치게 되어서는 안 된다는 것이다.

IV. 결 론

지금까지 열거한 것들을 종합해 보면 결국, 물체를 더 정확하게 분리하기 위해서는 Water-Shed의 가장 중요한 부분을 차지하는 Distance Transform에서 집수지역을 최대한 분리해 내는 것이 핵심이고 가장 중요한 부분이라는 것이다. 하지만 물체의 반을 가려서는 안 되기 때문에 정확한 Data 보다는 어느 정도의 결과에 만족해야만 했다. 이를 보완하고 보정하기 위해 몇 가지 영상처리 과정을 더 추가 시켰는데, 그것이 바로 Searching Minima와 Closing 과정이다. 그림 15는 Searching Minima과정과 Closing 과정을 끝낸 영상으로서, 그림 15의 왼쪽 그림은 Minima를 찾아낸 그림이고 오른쪽 그림은 Closing 과정을 끝낸 그림이다. 오른쪽 그림의 점이 두 개로 분리된 것을 볼 수 있다.



그림 15. Closing을 사용하여 Minima를 분리시킨 결과
Fig. 15. Image of seperated minima using the Closing procedure

그림 16은 그림 15에서 얻은 결과의 Minima를 가지고 나타낸 결과 영상이다. 물체가 서로 절반이상을 가린 상황에서 두 물체를 분리해 내었다.

정리하자면, 기존의 Water-Shed 만을 상용했던 경우와는 다르게 Minima를 구하고 그 구해진 Minima를 또 다시 Closing이라는 영상처리를 거침으로써 하나로 잘못 인식한 공을 두 개로 바르게 인식 하도록 한 것이다. 이는 공의 정확한 총 개수를 찾게 하고 나아가서는, 결국 data의 정확도를 높여 주게 된다.



그림 16. 분리시킨 Minima에 의한 영상 결과
Fig. 16. Image of seperated balls

이렇게 개수의 정확성에는 많은 공언을 하게 되겠지만 그림16에서와 같이 동그란 모양 그대로를 분리해 내는 데는 아직까지 어려운 점이 많다. 이를 해결하기 위해서는 새로운 Algorithm의 연구가 이루어져야 하겠다.

참고문헌

- [1] Lee Vincent and Pierre Soille, "Watersheds in digital spaces: An efficient algorithm", IEEE PAMI, 13, 6, p583~598, 1991
- [2] Jain, "Fundamentals of Digital Image Processing", Pearson Prentice Hall, Chap9, 1989
- [3] Rafael C.Gonzalez, Richard E.Woods, Steven L.Eddins, "Digital Image Processing", Pearson Prentice Hall, p418, 1992
- [4] Haralick, R. M. and L. Shapiro. 1992. Computer and Robot Vision. Reading, Mass.: Addison-Wesley Publishing Co.
- [5] Neuman, M., H. D. Sapirstein, E. Shwedyk and W. Bushuk. 1987.Discrimination of wheat class and variety by digital image analysis of whole grain samples. J. Cereal Sci.6:125-132
- [6] Zayas,I., F. S. Lai and Y. Pomeranz. 1986. Discrimination between wheat classes and varieties by image analysis. Cereal Chem. 63:52-56
- [7] N. S. Shashidhar, D. S. Jayas, T. G. Crowe and N. R. Bulley, "Processing of digital images of touching kernels by ellipse fitting", Department of Biosystems Engineering, MB. Canada R3T 5V6, 1997

자기소개



유 지 철(Ji-Chul Yu)

2005년 아주대학교 전자공학과(학사)

2006년 아주대학교 전자공학과
(석사과정)

2005년 ~ 농업 공학 연구소

※ 관심분야: Embedded system, dsp 신호처리, 실시간
시스템, RFID, 운영체제



김 영 길(Young-Kil Kim)

1978년 고려대 전자공학과 (공학사)

1980년 한국과학기술원 전자공학과
(공학석사)

1984년 E.N.S.T(프) 전자공학과 (공학박사)

1984년 ~ 현재 아주대 전자공학과 정교수

※ 관심분야: 멀티미디어 통신, Embedded, RFID, 차세대
무선 네트워크, 초음파 의료기기