《연구논문》 Hankook Kwanghak Hoeji, Volume 20, Number 2, April 2009 DOI: 10.3807/HKH.2009.20.2.094

영상 분할을 이용한 객체 기반 집적영상 깊이 추출

강진모 · 정재현 · 이병호†

서울대학교 전기공학부
㉮ 151-744 서울특별시 관악구 관악로 599

박재형

충북대학교 전기전자컴퓨터공학부
㉮ 361-763 충청북도 청주시 흥덕구 성봉로 410

(2008년 12월 29일 받음, 2009년 2월 12일 수정본 받음, 2009년 3월 3일 게재 확정)

본 논문에서는 집적영상에서 깊이 추출을 할 때 영상 분할 방법을 이용하여 각각의 물체에 대해 삼각형 메쉬(mesh) 모델을 구성하는 방법을 제안하였다. 집적영상에서 렌즈 어레이와 카메라를 이용하여 실제 물체를 픽업하면 요소영상(Elemental image) 집합을 얻을 수 있다. 요소영상 집합은 3차원 물체의 정보를 가지고 있으므로 대응점 분석을 통해 깊이 추출을 할 수 있다. 우선, 각 요소영상 중심점의 대응점 분석을 통해 시차를 구하고 이를 이용하여 깊이를 구한다. 요소영상의 중심점에 해당하는 물체의 X, Y 공간좌표는 각 점들이 사각형 격자 형태를 이룬다. 이 격자 형태의 점들 중에서 가까운 점 3개를 연결하여 삼각형 메쉬를 만들면 물체의 삼각형 메쉬 모델을 구할 수 있다. 이 때 각 물체에 대해 삼각형 메쉬 모델을 구하기 위해서 요소영상의 중심점들로 구성된 가운데 방향별 영상을 영상 분할하고 각각의 분할된 영역에 대해서만 삼각형 메쉬 모델을 구성하였다. 영상 분할 방법은 normalized cut 방법을 이용하였다. 제안된 방법의 검증을 위해 실제 물체를 픽업하고 각 물체의 삼각형 메쉬 모델을 구성하였다.

주제어: Three-dimensional display, Integral imaging, Depth extraction, Normalized cut

I. 서 론

3차원 디스플레이 기술이 발달함에 따라 실제 물체의 3차원 공간정보를 획득하는 것이 중요한 이슈가 되고 있다. 실제 물체의 공간 정보를 알 수 있다면, 사용자가 물체의 공간상 위치를 자유자재로 바꿀 수 있어 인터랙티브한(interactive) 사용자 중심의 3차원 디스플레이를 만들 수 있다. 또한 3차원 가상 물체와 실제 물체를 결합하는 3차원 컨텐츠를 제작할 수도 있다. 그밖에 의학, 로봇, 감시 시스템 등에 응용될 수 있다.

3차원 디스플레이 기술 중 집적영상 기술(integral imaging)은 1908년 Lippmann에 의해 처음 제안된 3차원 디스플레이 기술로 최근 많은 각광을 받고 있다.[1] 집적영상 기술은 여러 개의 작은 렌즈들로 이루어진 렌즈 어레이와 카메라 및 디스플레이 소자를 이용하며, 픽업과 디스플레이의 두 가지 과정을 거쳐 물체를 3차원으로 표시한다. 집적영상 기술은 특수한 안경이 필요 없고, 물체를 관찰할 수 있는 시야각 내에서 상하 방향뿐 아니라 좌우 방향으로 연속적인 시점을 가진다. 또한, 최근 발전된 디스플레이 소자들을 이용하여 실시간 천연색의 동영상을 재생할 수 있다는 등 여러 가지 장점들을 가지고 있다.

집적영상 기술이 발달함에 따라 집적영상 픽업 과정에서 얻을 수 있는 요소영상을 이용하여 깊이 추출을 하는 많은 방법이 제안되었다.[2-8] 가장 최근에 제안된 방법은 요소영상을 이용하여 3차원 공간 정보에 대한 삼각형 메쉬(mesh) 모델을 얻는 방법이다.[8] 삼각형 메쉬 모델은 요소영상에서 깊이 추출을 한 점들을 이용하여 구성되는데, 이 때 어떤 점들을 연결하여 삼각형 메쉬 모델을 만들지가 중요한 문제이다. 기존의 방법을 그대로 적용하면 그림 1과 같이 서로 다른 물체의 불연속적인 깊이 점들을 연결하는 잘못된 삼각형 메쉬(phantom surface)를 구성하게 된다.

본 논문에서는 가운데 방향별 영상을 영상 분할(segmentation)하여 각각의 분할된 영역에 대해서만 삼각형 메쉬 모델을 구성하는 방법을 제안한다. 영상 분할 방법은 normalized cut 알고리즘을 이용하였으며 영상 분할의 속도를 빠르게 하기 위해서 watershed 알고리즘을 먼저 적용한 후, normalized cut 알고리즘을 이용하였다. 영상 분할을 수행한 후 같은 물체에 포함된 점들만을 이용하여 삼각형 메쉬 모델을 구성함

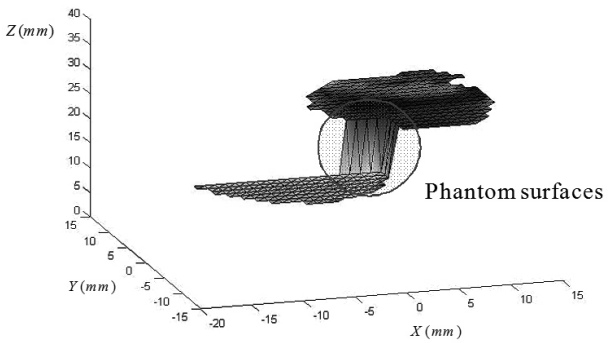


그림 1. 잘못된 삼각형 메쉬(Phantom surface)의 예.

†E-mail: byoungho@snu.ac.kr

으로써 서로 다른 물체를 연결하는 잘못된 삼각형 메쉬를 만들지 않을 수 있었다. 제안된 방법을 검증하기 위해 렌즈 어레이와 카메라를 이용하여 실제 물체를 픽업하고, 각 물체에 대해 깊이를 추출하여 3차원 삼각형 메쉬 모델을 구성하였다.

II. 깊이 추출 시스템 및 방법

그림 2는 $N_x \times N_y$ 렌즈 어레이와 CCD로 구성된 깊이 추출 시스템을 보여주고 있다. 편의상 YZ 평면만 나타내었다. f_a는 렌즈의 초점 길이, s는 CCD의 픽셀 크기, ϕ는 기초 렌즈의 크기이다. 그림 2와 같이 렌즈 어레이의 중심을 원점으로 하는 좌표계에서 (X, Y, Z)에 위치하는 물체의 한 점 P는 렌즈 어레이 각각의 렌즈를 통과하여 픽업 장치에 렌즈 어레이 수만큼의 상을 맺게 된다.[2] (q_x, q_y)번째 렌즈를 통과해서 생긴 점 P의 상은 요소영상 각각의 중심을 원점으로 두면 기하광학을 이용해 식 (1)과 같이 좌표를 구할 수 있다.

$$(x_{q_x}, y_{q_y}) = \left(\frac{f_a}{sZ}(q_x\phi - X), \frac{f_a}{sZ}(q_y\phi - Y) \right) \tag{1}$$

식 (1)로부터 (q_{x1}, q_{y1})번째 요소영상과 (q_{x2}, q_{y2})번째 요소영상의 수평 및 수직시차를 구하면 다음과 같다.

$$(d_x, d_y) = \left(\frac{\phi f_a}{sZ}(q_{x1} - q_{x2}), \frac{\phi f_a}{sZ}(q_{y1} - q_{y2}) \right) \tag{2}$$

수평 및 수직시차를 구할 수 있다면 다른 변수는 알고 있는 값이므로 식 (2)를 이용하여 깊이를 구할 수 있다. 시차를 구하기 위해서는 한 물체의 점에 대한 각 요소영상의 대응점을 찾아야 한다. 대응점 분석은 SSD(Sum of Squared Difference) 기법을 이용하여 구하였다. 각 요소영상이 $I(q_x, q_y)$, 각 요소영상의 좌표가 (x_r, y_r)이라고 가정하고 $I(q_{x1}, q_{y1})$, $I(q_{x2}, q_{y2})$ 두 요소영상에 SSD 기법을 적용하면 아래 식과 같다.

$$e_{(q_{x1},q_{y1}),(q_{x2},q_{y2})}(x_r, y_r, Z) = \sum_{i,j \in W_p} \{ I_{(q_{x1},q_{y1})}(x_r + i, y_r + j) - I_{(q_{x2},q_{y2})}(x_r + i + d_x, y_r + j + d_y) \}^2 \tag{3}$$

여기서 W_p는 요소영상에서 깊이를 구하는 점의 주변 윈도우 크기를 의미한다. 만약 이 윈도우가 요소영상의 경계와 만나게 되면, 오직 요소영상 내부에 있는 픽셀만을 이용하였다. 식 (3)을 최소로 하는 깊이 Z값이 요소영상 $I(q_{x1}, q_{y1})$의 (x_r, y_r)점에 대응하는 물체 점의 깊이 값이 된다. 이용 가능한 요소영상의 수가 많으므로 그림 3과 같이 이웃 영상 윈도우 W_N 내부에 있는 요소영상을 이용하여 S-SSD(Sum of the Sum of Squared Difference) 기법을 적용하면 다음과 같이 식을 구할 수 있다. 이 때 식 (4)를 최소로 하는 Z값이 물체의 점의 깊이 값이 된다.

$$E(x_r, y_r, Z) = \sum_{q_{x2}, q_{y2} \in W_N} e_{(q_{x1},q_{y1}),(q_{x2},q_{y2})}(x_r, y_r, Z) \tag{4}$$

식 (4)를 이용하여 물체의 점의 깊이를 구하면 식 (1)을 변형하여 다음과 같이 해당 점의 X, Y 좌표를 구할 수 있다.

$$(X, Y) = \left(q_x\phi - x_{q_x}\frac{sZ}{f_a}, q_y\phi - y_{q_y}\frac{sZ}{f_a} \right) \tag{5}$$

III. 제안된 3차원 삼각형 메쉬 모델링 방법

제안된 3차원 삼각형 메쉬 모델링 방법은 그림 4와 같다. 우선 그림 4(a)와 같이 식 (4)와 (5)를 이용하여 각 요소영상의 가운데 픽셀에 대해 깊이 추출을 수행하고 공간 좌표를 구한다. 각 요소영상의 중심점에 해당하는 물체의 공간 좌표는 각각의 대응하는 렌즈의 중심점과 같은 X, Y 좌표를 갖는다. 따라서 이 점들은 렌즈 어레이와 같이 사각형 격자 형태를 이룬다. 그 다음 각 요소영상의 위, 오른쪽, 대각선 오른쪽에 위치한 요소영상의 가운데 픽셀에 해당하는 물체 점들

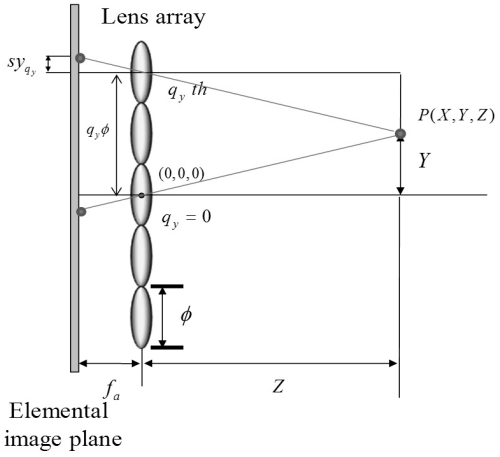


그림 2. 깊이 추출 시스템.

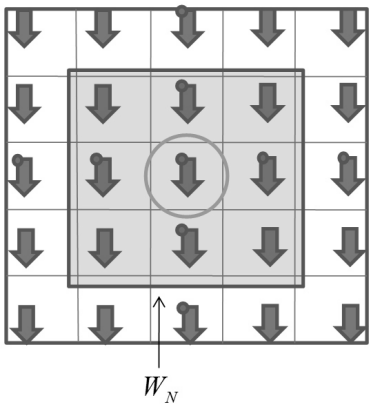


그림 3. 요소영상과 이웃 영상 윈도우 W_N.

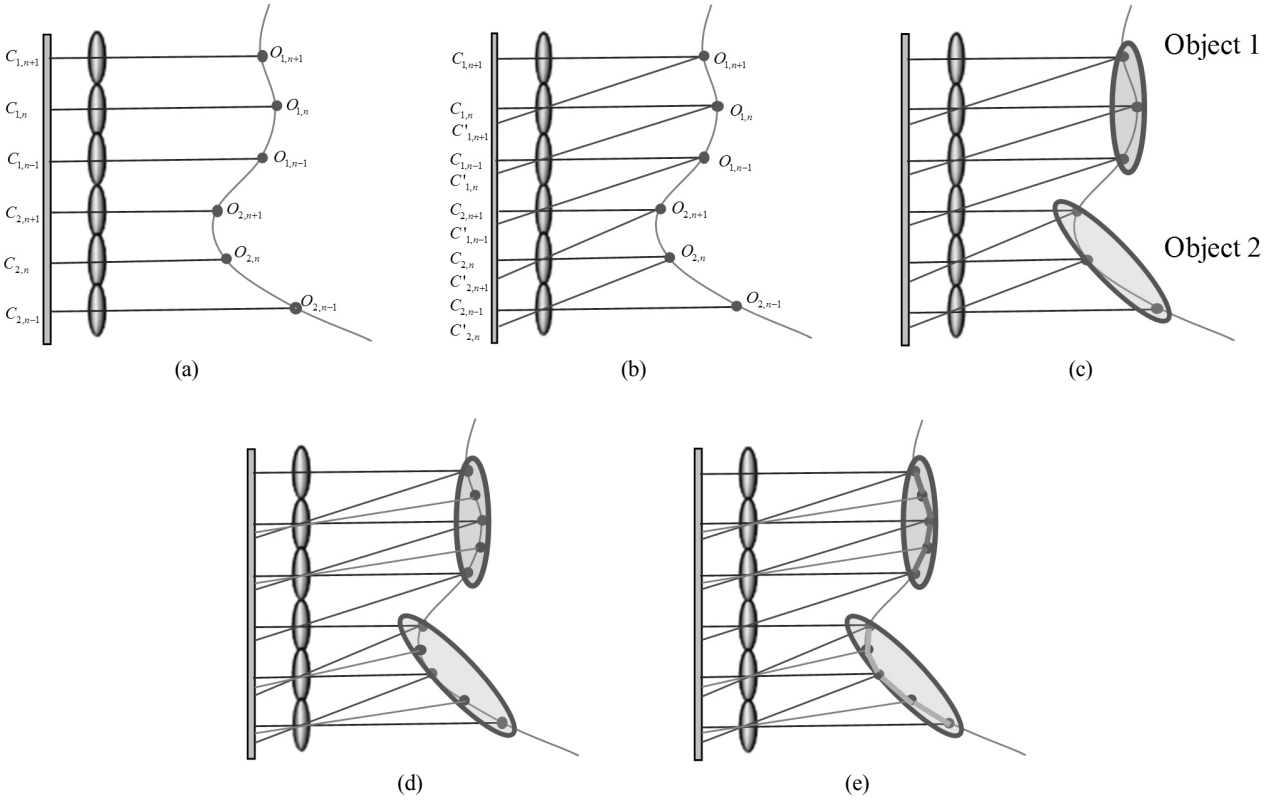


그림 4. 제안된 방법 (a)요소영상 중심점 깊이 추출 (b) 위, 오른쪽, 대각선 오른쪽 요소영상 중심점 해당 물체 점의 대응점 찾기 (c) 가운데 방향별 영상의 영상 분할 및 물체 분류 (d) 세부 깊이 추출 수행 (e) 삼각형 메쉬 모델링 및 interpolation.

을 그림 4의 (b)처럼 식 (1)을 이용하여 각 요소영상의 상에 대응시킨다.

그리고 그림 4의 (c)와 같이 가운데 방향별 영상을 영상 분할하여 각각의 물체를 분류한다. 방향별 영상은 각각의 요소영상에서 같은 위치에 있는 픽셀들을 모아 만든 영상으로, 요소영상과는 달리 FOV(Field of view)와 Sampling rate이 깊이에 관계없이 일정한 특징을 갖는다.[4] 방향별 영상의 FOV 크기는 전체 렌즈 어레이의 크기와 같고, sampling rate은 각 렌즈의 크기와 같다. 특히 가운데 방향별 영상의 경우 FOV의 위치가 렌즈 어레이의 위치와 수평방향으로 일치하기 때문에 집적영상에서 픽업되는 모든 물체를 포함하고 물체의 깊이에 관계 없이 sampling rate이 일정하므로 제안된 방법을 위한 영상 분할 이미지에 적합하다.

영상 분할은 디지털 영상을 여러 개의 픽셀 집합으로 나누는 과정을 뜻하며, 분할된 영역 내부는 색, 밝기, 재질 등의 특성이 유사한 특징을 가지고 있다.[9] 영상 분할은 컴퓨터 비전에서 오랫동안 연구되어온 분야로 다양한 방법이 존재하는데 그 중 우리는 normalized cut 알고리즘을 사용하였다. Normalized cut 알고리즘은 그래프 분할 알고리즘의 하나로 영상 분할을 그래프 분할에 대응시켜 고유벡터를 구하는 문제로 일반화하는 알고리즘이다.[10] 전체 영상을 하나의 그래프 G로 나타내면 영상의 픽셀인 노드 V와 각각의 노드 사이를 이어주는 에지 E로 나타낼 수 있다. 이때 각각의 노드 i, j 사이의 에지는 노드 사이의 유사도 w_{ij}로 나타낸다. 그래프 내에서 영역 A와 B 사이의 불일치 정도는 식 (6)과 같이 에지의 가중치의 합인 cut(A, B)로 계산할 수 있으며, 이 값이 가장 작은 경우가 최적의 영역 분할이다.

$$cut(A,B)=\sum_{i\in A,j\in B} w(i,j) \tag{6}$$

하지만 이 방법의 경우 따로 떨어진 노드를 분할하는 경우의 cut이 가장 작게 되어 정확한 분할이 이루어지지 않는 문제점이 있다. 이 점을 보완하기 위해 normalized cut은 영역 A의 노드에서 전체 그래프에 존재하는 노드의 가중치의 합인 식 (7)의 assoc(A, V)를 이용한다.

$$assoc(A,V)=\sum_{i\in A,k\in V} w(i,k) \tag{7}$$

영역 A, B를 나누는 cut(A, B)를 assoc(A, V)와 assoc(B, V)로 정규화하면 식 (8)과 같은 normalized cut을 구할 수 있다.

$$Ncut(A,B)=\frac{cut(A,B)}{assoc(A,V)}+\frac{cut(A,B)}{assoc(B,V)} \tag{8}$$

그래프의 에지 $w_{i,j}$로 표현되는 가중치 행렬 W를 이용하여 위 식을 행렬식으로 변형하고, Rayleigh quotient를 적용하면 고유벡터를 구해 위의 normalized cut을 최소화하는 영역 분

할을 구할 수 있다.[10]

이 방법을 방향별 영상의 분할에 그대로 적용하면 노드의 전체 수가 방향별 영상의 전체 픽셀 수와 같아지므로 연산시간이 오래 걸리고, 부분적인 잡음에 취약하게 된다. 따라서 본 논문에서는 영상 분할의 속도와 정확성을 높이기 위해 normalized cut 알고리즘을 적용하기 전에 영상의 밝기, 색상을 고려하는 watershed 알고리즘을 이용해 초기 영상 분할을 하였다.[11] Watershed 알고리즘은 영상의 gradient 값을 이용하여 영상의 에지를 구분하는 방법으로, 밝기 정보의 변화가 큰 부분에서 영상을 분할해 내는 특징을 가지고 있다. 따라서 watershed 알고리즘을 적용하면 영상의 에지에 해당하는 부분은 어떤 물체 영역에도 속하지 않게 된다.

본 논문에서는 우선 watershed 알고리즘을 사용하여 영상 분할을 하고 각 분할된 영역을 노드로 정의하여 normalized cut 알고리즘을 적용하였다. Normalized cut 알고리즘은 노드 간의 유사도를 잘 표현할 수 있는 가중치의 선정이 중요하다. 가중치는 각 노드의 밝기, 위치, 깊이 값을 이용하여 다음 식과 같이 주었다.

$$W_{ij} = \exp\left(\frac{-w_I \times \left|I_i - I_j\right|}{\sigma_I}\right) \times \exp\left(\frac{-w_t \times \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sigma_t}\right) \times \exp\left(\frac{-w_d \times \left|d_i - d_j\right|}{\sigma_d}\right) \quad (9)$$

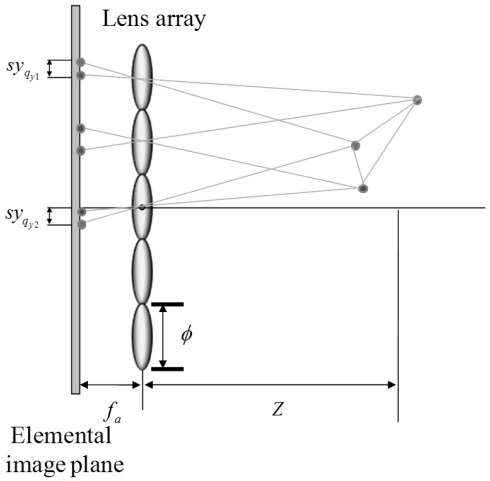
위 식에서 I는 각 노드의 평균 밝기, x와 y는 각 노드의 평균 위치, d는 각 노드의 평균 깊이 값을 의미한다. 그리고 w_I, w_t, w_d는 밝기, 위치, 깊이 값의 가중치를 σ_I, σ_t, σ_d는 각각의 표준편차를 나타낸다. 식 (9)는 밝기와 위치 정보만을 가중치로 이용하던 기존의 normalized cut과는 달리 깊이에 대한 가중치를 함께 포함하고 있다. 본 논문에 제안된 방법에서 영상 분할의 목적은 방향별 영상에서 서로 다른 물체를 구분하는 것이다. 서로 다른 물체의 깊이가 다르고 같은 물체 내부에서 깊이 변화가 크지 않다고 가정하면, 깊이에 대한 가중치를 가장 크게 두는 것이 서로 다른 물체를 정확하게 분할할 가능성이 높아진다. 만약 물체 내부의 깊이 변화가 크다면 주어진 방향별 영상의 특성에 따라 가중치를 다르게 하여 보다 정확하게 영상 분할을 할 수 있다. 또한 normalized cut은 분할 구역의 수를 임의로 지정할 수 있기 때문에 촬영된 물체의 수에 따라 분할 구역의 수를 지정할 수 있다는 장점이 있다.

영상 분할된 결과를 이용하여 그림 4의 (d)처럼 같은 물체 내부 점에 대해서만 세부 깊이 추출을 수행한다. 이 때 watershed 알고리즘을 거치면서 생긴 에지는 물체로 판단되지 않기 때문에, 에지가 아닌 분할된 영역 내부 점에 대해서만 세부 깊이 추출이 수행된다. 이에 따라 기존 방법과는 다르게 물체의 경계선에 대해서는 삼각형 메쉬 구조를 형성하지 않게 된다. 이것은 기존 방법에 비해 단점이지만, 경계선의 경우 어떤 물체에 속하는지 구분하기가 어렵고, 깊이 추출이 정확하게 되지 않을 가능성이 높으므로 물체 별로 정확한 삼각형 메쉬 구조를 얻기 위해서는 이용하지 않는 것이 낫다. 또한 전체 resolution이 크다면 경계선의 resolution 손실은 무시할 수 있을 것이다.

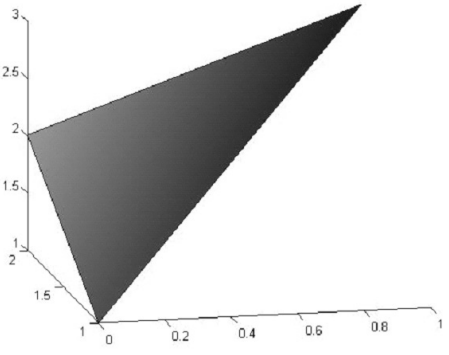
세부 깊이 추출은 subdivision parameter S값에 따라 깊이 추출을 수행하는 점의 수가 달라진다. 세부 깊이 추출을 위한 각 요소영상에서의 점의 위치는 식 (10)을 이용하였다.[8] 아래 식에서 p_{00}는 요소영상의 중심점 p_{10}, p_{01}, p_{11}는 각각 위, 오른쪽, 대각선 오른쪽에 위치한 요소영상의 가운데 픽셀에 해당하는 물체 점의 대응점 픽셀 위치를 의미한다.

$$p' = \frac{(S-i)(S-j)}{S^2} p_{00} + \frac{i(S-j)}{S^2} p_{10} + \frac{(S-i)j}{S^2} p_{01} + \frac{ij}{S^2} p_{11} \quad (1 \le i, j \le S-1) \quad (10)$$

세부 깊이 추출이 끝나면 깊이 추출된 점들에 Median filter와 Gaussian filter를 적용하여 깊이 추출의 오류를 수정한다. 마지막으로 그림 4의 (e)처럼 각 물체의 점들을 연결하여 각 물체 별로 삼각형 메쉬 구조를 형성한다. 형성된 삼각형들의 내부는 그림 5와 같이 각 점의 색상 정보를 이용하여 interpolation을 통해 채운다.



(a) 삼각형 메쉬



(b) interpolation

그림 5. 삼각형 메쉬 모델링.

IV. 실 험

제안된 방법을 검증하기 위하여 실제 물체를 픽업하여 요소영상을 얻고 실험을 수행하였다. 실험 셋업은 그림 6과 같다. 크기 1 mm, 초점 거리 3.3 mm인 렌즈 어레이가 사용되었으며 그림 7의 T와 L 문자가 픽업용 물체로 사용되었다. 렌즈 어레이로부터 T는 15 mm, L은 30 mm 떨어져 있으며 카메라로 촬영한 요소영상의 집합은 그림 8과 같다. 그림 8은 36×27개의 요소영상으로 이루어져 있으며, 각 요소영상의 픽셀 수는 26×26이다.

픽업된 요소영상의 각 중심점에 대하여 식 (4)와 (5)를 이용하여 공간 좌표를 구하면 그림 9의 (a)와 같다. 이웃 영상으로 상하좌우 4개의 요소영상을 사용하였으며 W_p는 9×9이다. 그림 9의 (a)를 살펴보면 T와 L의 경계부분에서 깊이 추출의 오류가 발생함을 확인할 수 있다. 실제 T와 L의 깊이를 대입하여 최대 깊이 추출 값이 50이라고 가정하고 PSNR(Peak signal-to-noise ratio)을 구해보면 30.73 dB가 나온다. 기존의 방법대로 그림 9의 (a)의 모든 점에 대해 삼각형 메쉬를 구하고 내부를 interpolation 하면 그림 9의 (b)와 같다. T와 L의 경계면에서 잘못된 삼각형 메쉬가 구성됨을 확인할 수 있다.

그림 10은 각각의 요소영상의 중심점을 모아 만든 가운데 방향별 영상이다. 그림 11의 (a)는 가운데 방향별 영상을 water-

그림 6. 실험 셋업.

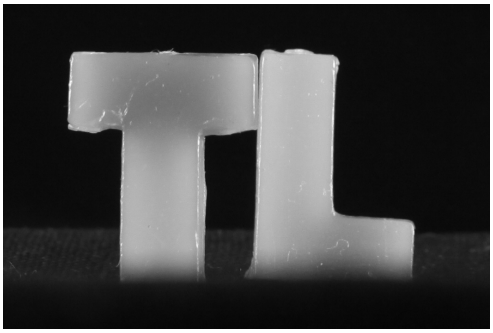


그림 7. 픽업용 물체.

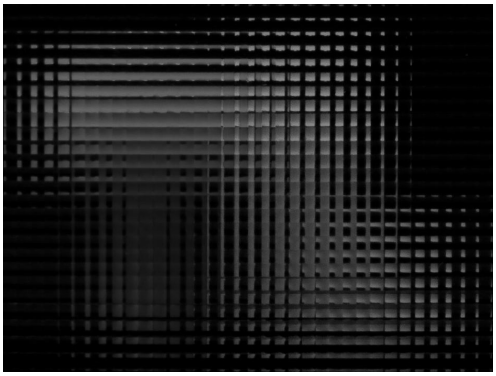
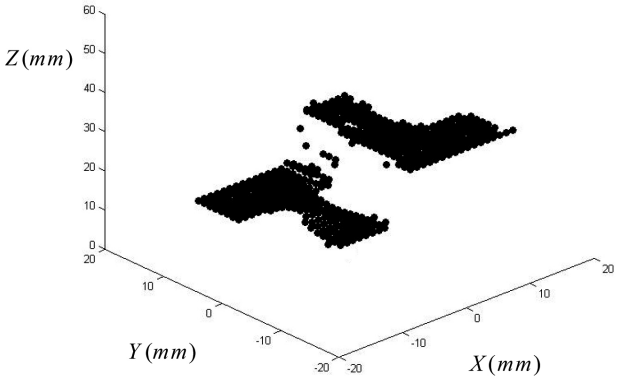
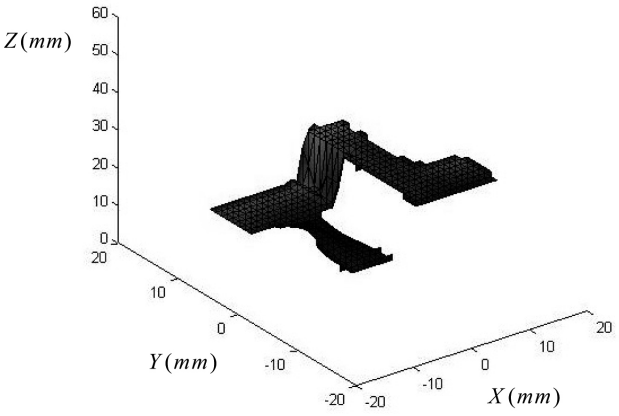


그림 8. 촬영된 요소영상.



(a) 깊이 추출 결과



(b) 삼각형 메쉬 모델링 결과

그림 9. 기존 방법을 이용한 삼각형 메쉬 모델링.

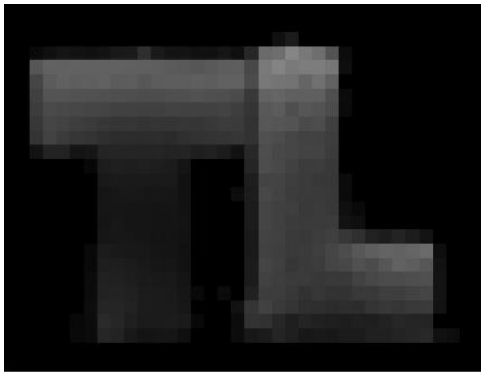
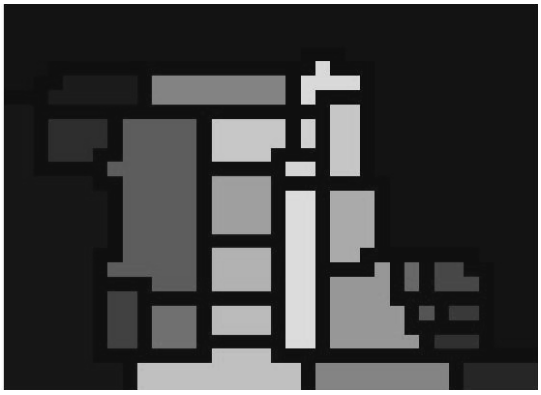
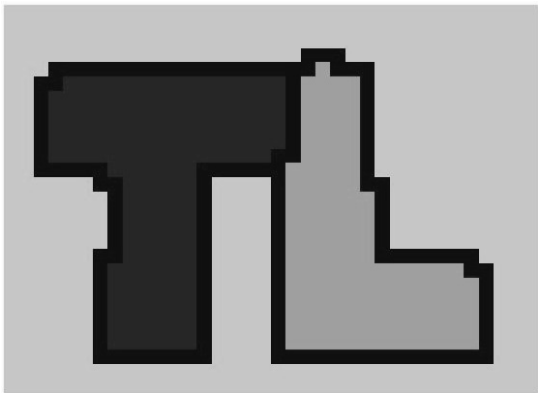


그림 10. 가운데 방향별 영상.

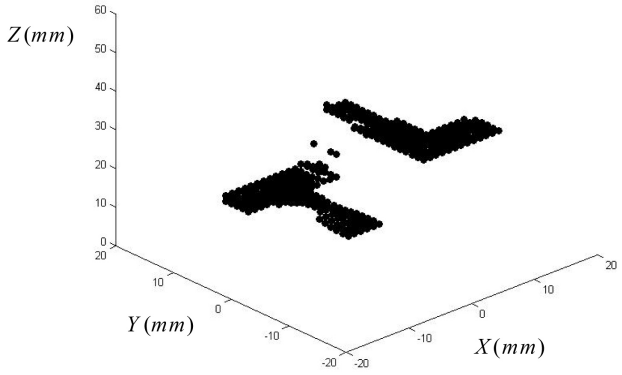


(a) watershed 적용

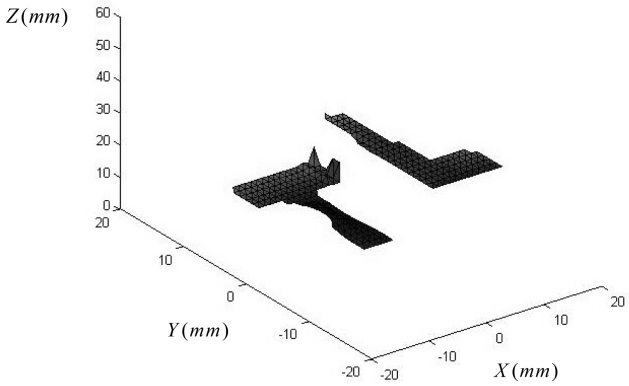


(b) normalized cut 적용

그림 11. 가운데 방향별 영상의 영상 분할.



(a) 깊이 추출 결과



(b) 삼각형 메쉬 모델링 결과

그림 12. 제안된 방법을 이용한 삼각형 메쉬 모델링.

shed를 이용하여 영상 분할한 결과이다. 이 결과에 normalized cut을 적용해 3개의 영역으로 영상 분할하면 그림 11의 (b)를 얻을 수 있다. T와 L 및 배경이 각각 잘 분할된 것을 확인할 수 있다. 가중치 w_I, w_t, w_d는 각각 0.5, 1, 3을 대입하였다. 그림 12의 (a)는 그림 11을 이용하여 같은 영역으로 분할된 점들에 대해서만 깊이 추출을 한 결과이다. 그림 11의 (b)에 존재하는 경계면을 사용하지 않아 resolution이 감소하지만, T와 L 사이의 경계면에서 깊이 추출 오류가 감소함을 확인할 수 있다. 실제 T와 L의 깊이를 대입하여 최대 깊이 추출 값이 50이라고 가정하고 PSNR을 구해보면 32.07 dB가 나온다. 그림 12의 (a) 점들에 대해 삼각형 메쉬를 구성하고 내부를 interpolation하면 그림 12의 (b)의 결과를 얻을 수 있다. T와 L 사이에 잘못된 삼각형 메쉬 없이 T와 L 각각에 대해 삼각형 메쉬 모델을 구한 것을 확인할 수 있다. 그림 12의 결과를 이용하여 임의로 T는 40 mm에 L은 20 mm 위치에 두고 삼각형 메쉬 모델을 구성하면 그림 13과 같다.

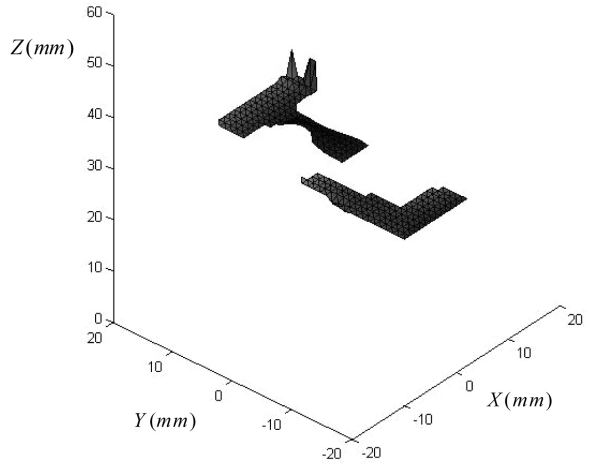


그림 13. 임의로 물체 깊이를 바꾼 삼각형 메쉬 모델링.

V. 결 론

본 논문에서는 집적영상에서 깊이 추출을 할 때 영상 분할 방법을 이용하여 각각의 물체에 대해 삼각형 메쉬(mesh) 모델을 구성하는 방법을 제안하였다. 집적영상에서 얻은 요소영상을 이용하여 각 요소영상 중심점의 공간 좌표를 구한다. 그리고 각 물체에 대해 삼각형 모델을 구하기 위해서 요소영상의 중심점들로 구성된 가운데 방향별 영상을 watershed와 normalized cut 방법을 이용해 영상 분할하고 각각의 분할된 영역에 대해서 세부 깊이 추출을 하고 삼각형 메쉬 모델을 구성하였다. 그리고 제안된 방법의 검증을 위해 실제 물체를 픽업하고 각 물체의 3차원 삼각형 메쉬 모델을 구성하였다. 제안된 방법을 이용하여 집적영상에서 실제 물체를 픽업한

후 각각의 물체 별 3차원 컨텐츠를 만들 수 있기 때문에, 인터랙티브한 3차원 컨텐츠 제작에 활용 가능할 것으로 기대된다.

감사의 글

본 연구는 BK21 프로그램(Information Technology)의 지원을 받았습니다.

참고문헌

[1] G. Lippmann, "Épreuves réversibles. Photographies intégrale," *Comptes-Rendus de l'Académie des Sciences*, vol. 146, no. 9, pp. 446-451, 1908.

[2] J.-H. Park, S.-W. Min, S. Jung, and B. Lee, "A new stereovision scheme using a camera and a lens array," Conference on Algorithms and Systems for Optical Information Processing V, SPIE Annual Meeting, San Diego, CA, USA, Proc. SPIE, vol. 4471, pp. 73-80, Jul.-Aug. 2001.

[3] H. Choi, J.-H. Park, J. Hong, and B. Lee, "An improved stereovision scheme using one camera and a composite lens," Stereoscopic Displays and Virtual Reality Systems XI, Electronic Imaging, San Jose, CA, USA, Proc. SPIE, vol. 5291, pp. 196-203, Jan. 2004.

[4] J.-H. Park, S. Jung, H. Choi, Y. Kim, and B. Lee, "Depth extraction by use of a rectangular lens array and one-dimensional elemental image modification," *Appl. Opt*., vol. 43, no. 25, pp. 4882-4895, 2004.

[5] C. Wu, A. aggoun, M. McCormick, and S. Y. Kung, "Depth extraction from unidirectional image using a modified multi-baseline technique," Stereoscopic Displays and Virtual Reality Systems IX, Proc. SPIE, vol. 4660, pp. 135-145, 2002.

[6] C. Wu, A. Aggoun, M. McCormick, and S. Y. Kung, "Depth measurement from integral images through viewpoint image extraction and a modified multibaseline disparity analysis algorithm," *Journal of Electronic Imaging*, vol. 14, no. 2, pp. 023018-1-023018-9, 2005.

[7] C. Wu, M. McCormick, A. Aggoun, and S. Y. Kung, "Depth mapping of integral images through viewpoint image extraction with a hybrid disparity analysis algorithm," *Journal of Display Technology*, vol. 4, no.1, pp. 101-108, 2008.

[8] G. Passalis, N. Sgouros, S. Athineos, and T. Theoharis, "Enhanced reconstruction of three-dimensional shape and texture from integral photography images," *Appl. Opt*., vol. 46, no. 22, pp. 5311-5320, 2007.

[9] L. G. Shapiro and G. C. Stockman, *Computer Vision* (Prentice-Hall, New Jersey, USA, 2001), pp. 279-325.

[10] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888-905, 2000.

[11] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE Transactions of Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583-598, 1991.

Object-Based Integral Imaging Depth Extraction Using Segmentation

Jin-Mo Kang, Jae-Hyun Jung, and Byoungho Lee†

School of Electrical Engineering, Seoul National University, Gwanak-Gu Gwanakro 599, Seoul 151-744, Korea

†*E-mail: byoungho@snu.ac.kr*

Jae-Hyeung Park

School of Electrical & Computer Engineering, Chungbuk National University, 410 SungBong-Ro, Heungduk-Gu, Cheongju-Si, Chungbuk 361-763, Korea

(Received December 29, 2008; Revised manuscript February 12, 2009; Accepted March 3, 2009)

A novel method for the reconstruction of 3D shape and texture from elemental images has been proposed. Using this method, we can estimate a full 3D polygonal model of objects with seamless triangulation. But in the triangulation process, all the objects are stitched . This generates phantom surfaces that bridge depth discontinuities between different objects. To solve this problem we need to connect points only within a single object. We adopt a segmentation process to this end. The entire process of the proposed method is as follows. First, the central pixel of each elemental image is computed to extract spatial position of objects by correspondence analysis. Second, the object points of central pixels from neighboring elemental images are projected onto a specific elemental image. Then, the center sub-image is segmented and each object is labeled. We used the normalized cut algorithm for segmentation of the center sub-image. To enhance the speed of segmentation we applied the watershed algorithm before the normalized cut. Using the segmentation results, the subdivision process is applied to pixels only within the same objects. The refined grid is filtered with median and Gaussian filters to improve reconstruction quality. Finally, each vertex is connected and an object-based triangular mesh is formed. We conducted experiments using real objects and verified our proposed method.

OCIS codes: (100.0100) Image processing; (100.6890) Three-dimensional image processing; (110.6880) Three-dimensional image acquisition