

밴드 매칭, 경계제거, 영역분할을 이용한 영상 인페인팅

이수빈[†], 서용덕^{**}

요 약

본 논문은 밴드 매칭과 경계제거 두 부분으로 나누어진 새로운 인페인팅 방법을 제안한다. 밴드 매칭은 삭제하고자하는 영역(인페인팅 영역)을 둘러싸는 밴드와 영상의 나머지 영역을 비교하여 그 차이가 가장 작은 영역을 인페인팅 영역에 채워 넣는 것이고, 경계제거는 밴드매칭으로 채워진 영역과 주변영역 사이에 나타나는 경계를 제거하는 것이다. 제안하는 방법은 인페인팅 영역으로 연속된 선이 지나는 경우 좋지 못한 결과를 얻는 경우가 있는데, 이러한 경우 영역분할이라는 과정을 더한다. 영역분할 방법은 인페인팅 영역을 작게 나누고, 분할된 각각의 영역에 대해 밴드 매칭과 경계제거를 수행하는 것이다. 영역분할을 이용하는 경우 분할 개수에 따라 여러 개의 다른 결과를 얻을 수 있고, 사용자는 그 결과들 중에서 가장 좋은 결과를 선택할 수 있다.

Image Inpainting by Band Matching, Seamless Cloning and Area Sub-Division

Subin Lee[†], Yongduek Seo^{**}

ABSTRACT

We propose a novel image inpainting method composed of two parts: band matching and seamless cloning. In band matching, a band enclosing the boundary of a missing region is compared to those from the other parts of the image. The inner area of the minimum difference band is then copied to the missing region. Even though this band matching results in successful inpainting in many practical applications, brightness discontinuity (a seam) may appear between the filled missing region and its neighborhood. We apply seamless cloning to remove such discontinuity between the two regions. However, since this basic method using one patch may not deal with cases where there are abrupt changes of color or brightness along the boundary, we furthermore devise one more step: target sub-division. The target area is subdivided into small sub-areas, and the band matching and seamless cloning are applied to each of them. The multiple results from the sub-division are then ordered according to inpainting quality, which is measured based on the edge map or discontinuity map along the boundary band.

Key words: Image Inpainting(영상 인페인팅), Seamless Cloning(경계제거), Area Sub-Division(영역분할)

1. 서 론

최근 영화들 중에 특수효과가 사용되지 않은 영화

가 없을 정도로 특수효과는 영화에서 광범위하게 사용되고 있다. 많은 특수 효과 기술 중 디지털 인페인팅(digital inpainting)은 실사 영상에서 제거된 특정

※ 교신저자(Corresponding Author): 이수빈, 주소: 서울시 마포구 신수동 1번지(121-742), 전화: 02)706-5986, FAX: 02)706-5908, E-mail: adrift@sogang.ac.kr

접수일: 2007년 5월 14일, 완료일: 2007년 12월 12일

[†] 정회원, 서강대학교 영상대학원 미디어공학과

^{**} 서강대학교 영상대학원 미디어공학과

(E-mail: yndk@sogang.ac.kr)

※ 이 논문은 문화관광부 및 한국문화콘텐츠진흥원의 문화콘텐츠기술연구소(CT)육성사업의 연구결과로 수행되었고, 2007년도 정부(과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. R01-2006-000-11374-0).

영역을 주변 배경과 어울리게 채워 넣는 기술로 다양한 분야에서 이 기술이 사용되고 있다. 한 예로 영화 촬영 후 영화 속 배경에 시대에 맞지 않은 물체나 사람이 있을 경우 영화를 재촬영할 수도 있지만, 이런 경우 비용과 시간이 많이 든다. 이때 디지털 인페인팅 기술을 사용하면 쉽게 불필요한 물체나 사람을 지울 수 있다.

영상 인페인팅 방법에 관한 연구는 크게 두 가지로 나누어 볼 수 있다. 첫 번째는 작은 영역을 채우는 방법에 대한 연구이다[1,2]. 이것은 편미분 방정식을 이용하여 보간하는 것으로, 그림 1과 같이 비슷한 색상을 가지는 작은 영역에 대해서 효과적이다. 하지만 텍스처가 있는 부분은 주변 색상이 서로 다르기 때문에 흐림 현상이 발생하여 좋지 않은 결과를 보인다.

두 번째로 텍스처 합성을 이용하여 큰 영역을 채우는 방법에 대한 연구가 있다[3-8]. 텍스처 합성은 작은 크기의 입력 영상으로부터 패치들을 복사하여 큰 크기의 출력 영상을 생성하는 것이다[9,10]. 텍스처 합성을 이용한 방법은 그림 2와 같이 구조적 연속성을 갖는 영역에 대해서는 좋은 결과를 보이지만, 적절한 텍스처가 없거나 부드러운 영역에 대해서는 구조적 연속성에 대한 특징을 정의할 수 없기 때문에 효과적이지 않다.

본 논문은 텍스처 합성을 이용하여 큰 영역을 인페인팅하는 방법을 제안한다. 제안하는 방법은 [3]의 방법과 비교할 수 있다. 두 방법의 차이점은 다음과 같다.



그림 1. 편미분 방정식을 이용하여 작은 영역(붉은 글자)을 인페인팅한 결과



그림 2. 텍스처 합성을 이용하여 큰 영역(흰색)을 인페인팅한 결과

첫째로 패치의 크기이다. [3]의 방법은 인페인팅 영역의 윤곽을 사용자가 정의한 작은 크기의 패치들로 나누고, 영상의 색상과 그래디언트(gradient)로 계산된 우선순위에 따라 패치들을 채워 넣는다. [3]의 방법으로 좋은 결과를 얻기 위해서 사용자는 적절한 패치크기를 찾아야한다. 하지만 제안하는 방법은 하나의 큰 패치를 이용하여 인페인팅 영역을 채워 넣기 때문에 사용자가 좋은 결과를 얻기 위한 패치크기를 찾을 필요가 없다. 또한 [3]의 경우 각 패치마다 영상 내에서 인페인팅 영역을 채우기 위한 적절한 영역을 찾는 계산과정이 필요하기 때문에 계산시간이 오래 걸리는 반면, 제안하는 방법은 하나의 패치만 찾으면 되기 때문에 계산시간이 빠르다.

둘째로 [3]의 경우 인페인팅 영역과 이웃한 영역에 대한 경계제거 과정이 없지만, 제안하는 방법은 경계제거 과정을 통해 두 영역사이의 경계를 부드럽게 변화시켜 더 나은 결과를 얻을 수 있다.

마지막으로 인페인팅 영역이 영상 내에서 큰 부분을 차지하면 제안하는 방법을 수행할 수 없다. 따라서 인페인팅 영역을 분할한 다음 제안하는 인페인팅 과정을 수행할 수 있다. 예를 들어 인페인팅 영역을 두 부분으로 나누면 두 개의 영역이 생기고, 두 영역에 대해 각각 제안하는 인페인팅 과정을 수행한다. 이 경우 인페인팅 영역을 작게 분할할수록 [3]의 방법과 유사해지지만, 기본적으로 제안하는 방법은 하향식 접근 방식인 반면에 [3]은 상향식 접근 방식이다.

2장에서는 밴드 매칭과 경계제거로 이루어진 제안하는 영상 인페인팅 방법에 대해, 3장에서는 영역 분할을 이용한 인페인팅 방법에 대해 설명하고, 각각의 결과를 보인다. 4장에서 제안하는 방법과 기존 방법의 실험결과를 비교하고, 5장에서 결론을 맺는다.

2. 제안하는 영상 인페인팅

그림 3은 제안하는 영상 인페인팅의 과정을 보여준다. 제안하는 영상 인페인팅은 크게 밴드 매칭과 경계제거로 나눈다. 밴드 매칭에서 밴드는 삭제된 영역 즉, 인페인팅 영역을 둘러싸는 일정한 두께를 갖는 영역을 말한다. 밴드는 인페인팅 영역을 채워 넣기 위해 인페인팅 영역을 제외한 영상의 나머지 부분과 밴드를 비교하는데 사용된다. 영상의 나머지 부분에서 밴드와 가장 유사한 부분을 찾으면 그 부분의

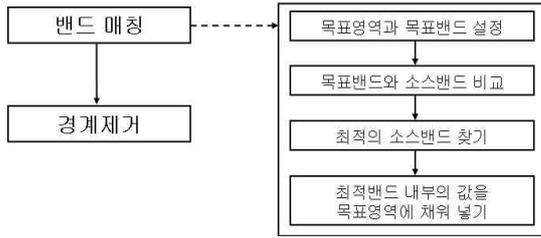


그림 3. 제안하는 영상 인페인팅 과정

내부를 인페인팅 영역에 채워 넣는 것이 밴드 매칭의 과정이다. 밴드 매칭 후 채워진 인페인팅 영역과 주변영상 사이에 경계(seam)가 나타날 수 있다. 이 경우 경계제거 과정을 거쳐 채워진 인페인팅 영역과 주변영상의 경계를 제거한다. 2.1에서 밴드 매칭에 대해 설명하고, 2.2에서 경계제거에 대해 설명한다.

2.1 밴드 매칭

밴드 매칭의 첫 번째 단계는 사용자가 영상에서 지우고자 하는 영역 즉, 인페인팅 영역을 수동으로 정하는 것이다. 그림 4에서 흰색영역은 목표영역이고, 붉은색 선은 목표영역의 경계이다. 이 논문에서는 인페인팅 영역을 목표영역이라 한다.

두 번째 단계는 목표영역의 둘러싸는 일정한 두께

를 갖는 목표밴드를 정의하는 것이다. 그림 4에서 파란색영역은 목표밴드로, 영상 내의 여러 후보 소스밴드들과의 비교를 위해 사용된다. 여기서 소스밴드란 입력영상에서 목표영역을 제외한 나머지 부분을 목표밴드와 같은 모양과 두께로 나눈 것을 말한다. 그림 5(a)에서 녹색 부분들은 목표밴드와 비교하기 위해 사용된 후보 소스밴드의 일부를 표시한 것이다.

세 번째 단계로, 목표밴드와 후보 소스밴드들 사이의 색상과 그래디언트의 차의 합을 계산한다. k 는 후보 소스밴드들의 인덱스를 나타내고, C_k 는 세 색상 채널의 차이의 합, D_k 는 그래디언트의 차이를 나타낸다. 목표밴드와 소스밴드의 C_k 와 G_k 는 식 (1)과 식 (2)의 식으로 계산한다.

$$C_k = \sum_{i=1}^N |R_i^{s_k} - R_i^t| + |G_i^{s_k} - G_i^t| + |B_i^{s_k} - B_i^t| \quad (1)$$

$$G_k = \sqrt{\left(\sum_c \|\nabla I_c\|\right)}, \quad c = \{R, G, B\} \quad (2)$$

식 (1)과 식 (2)에서 t 는 목표밴드, s 는 소스밴드를 나타낸다. N 은 밴드의 총 화소수를 나타내고, R, G, B 는 빨강, 녹색, 파랑의 색상채널이다. $\|\nabla I\| = \|(\nabla t_x - \nabla s_x, \nabla t_y - \nabla s_y)\|$ 로 $\nabla t_x - \nabla s_x$ 는 목표밴드와 소스밴드사이의 수평방향 그래디언트의 차이를

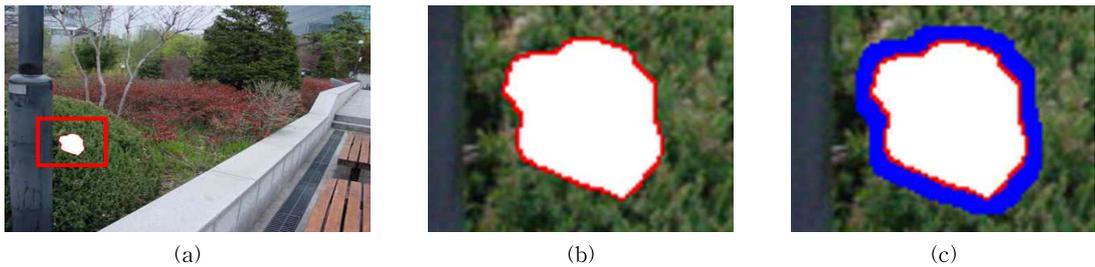


그림 4. (a) 입력영상 (b) 입력영상의 확대 : 흰색영역은 목표영역이고, 붉은색 선은 목표영역의 경계이다. (c) 목표밴드 설정 : 파랑색영역이 목표밴드이다.



그림 5. 밴드 매칭의 과정 : (a) 목표밴드(파란색)와 후보 소스밴드들(녹색)의 비교 (b) 목표밴드와 가장 유사한 소스밴드의 위치, D_k 가 최소인 소스밴드 (c) 목표영역에 최적밴드의 내부 색상정보 채워 넣기 (d) 결과영상

나타내고, $\nabla t_y - \nabla s_y$ 는 목표밴드와 소스밴드사이의 수직방향 그래디언트의 차이를 나타낸다.

목표밴드와 k 번째 소스밴드사이의 D_k 는 C_k 와 G_k 의 합으로 다음과 같이 계산한다.

$$D_k = \lambda_1 C_k + \lambda_2 G_k \quad (3)$$

식 (3)의 식에서 λ_1 과 λ_2 는 C_k 와 G_k 의 가중치를 나타내기 위한 상수이다.

마지막 단계로, D_k 가 최소가 되는 소스밴드를 선택하고, 그 소스밴드의 내부 색상정보를 목표영역에 채워준다. 그림 5(b)에서 녹색 부분이 D_k 가 최소가 되는 소스밴드이고, 여기서도 그 소스밴드를 최적밴드라 한다. 그림 5(c)는 목표영역에 최적밴드의 내부 색상정보를 채워 넣은 것이고, 그림 5(d)는 결과영상이다.

그림 6은 제안하는 방법과 [3]의 방법의 결과를 비교한 것이다. 그림 6(a)는 제안하는 방법으로 실험한 결과로 아랫줄은 인페인팅 영역을 확대한 것이다. 제안하는 방법은 목표밴드와 후보 소스밴드들을 한 번씩만 비교하면 되지만, [3]의 방법은 각 패치마다 소스패치들과 반복비교를 해야 하기 때문에 계산시간이 오래 걸린다. 제안하는 방법의 계산시간은 밴드가 812화소일 때 약 14초이고, [3]의 방법의 계산시간은

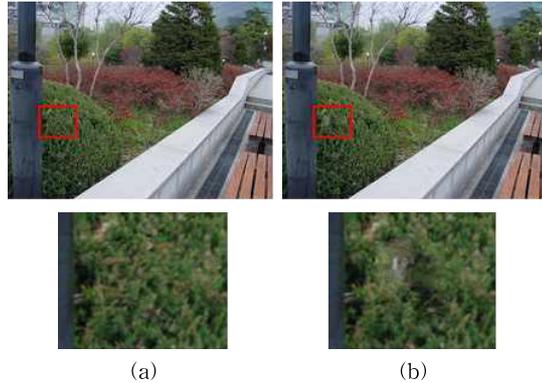


그림 6. 제안하는 방법과 [3]의 방법의 비교 : (a) 제안하는 방법의 결과영상 (계산시간 약 14초) (b) [3]의 방법의 결과영상 (계산시간 - 약 4분)

패치크기가 7×7 (49화소)일 때 약 4분이다. 또한 그림 6(b)에서 보듯이 [3]의 방법의 결과는 여러 개의 패치들로 목표영역을 채우기 때문에 패치사이에 연속성이 없을 경우 채워진 목표영역이 매끄럽지 않을 수 있다. 반면 제안하는 방법은 목표영역을 한 번에 채우기 때문에 채워진 목표영역이 매끄럽다.

그림 7은 제안하는 방법의 여러 결과영상이다. 밴드 매칭은 복잡하거나 반복적인 텍스처를 가진 영상이나 주변의 색상이 비슷한 영상에서 좋은 성능을 보인다.



그림 7. 밴드 매칭의 결과 : 윗줄은 입력영상이고, 아랫줄은 밴드 매칭의 결과이다. 계산시간은 왼쪽부터 각각 약 15, 26, 200초이다. 밴드의 화소 수는 각각 1369, 1576, 2324 화소이다. [3]의 방법의 경우 계산시간은 각각 약 420, 780, 1800초이다.



그림 8. 경계가 나타나는 예 : 왼쪽은 입력영상, 가운데는 밴드 매칭의 결과, 오른쪽은 밴드 매칭 후 경계제거를 한 결과이다.

2.2 경계제거

밴드 매칭으로 인페인팅을 하면 목표영역과 주변 영역사이에 경계가 나타나는 경우가 있다. 이러한 경계는 목표영역과 주변영역사이의 그래디언트 차이로 나타난다. 그림 8은 밴드매칭 후 경계가 나타나는 예를 보여준다. 이 논문에서는 밴드매칭 후 목표영역과 주변영역사이에서 나타나는 경계를 경계제거과정을 통하여 제거한다[11,12]. 이 논문에서는 [11]의 방법을 이용해 경계를 부드럽게 바꾸고자 한다.

경계제거는 목표밴드와 최적밴드의 화소값의 비율을 계산하고, 그 비율을 이용하여 목표영역의 각 화소값을 재계산하는 것이다. 경계제거의 첫 번째 단계는 목표영역과 목표밴드의 초기 $h(x,y)$ 을 계산하는 것이다. $h(x,y)$ 는 목표밴드의 화소값과 최적밴드의 화소값의 비율로 식 (4)로 계산한다.

$$h(x,y) = \frac{f(x,y)}{g(x,y)} \tag{4}$$

식 (4)에서 $f(x,y)$ 는 목표밴드, $g(x,y)$ 는 최적밴드의 색상값이다. 목표영역의 초기 $h(x,y)$ 값은 1로 설정한다.

두 번째 단계는 반복을 통하여 $h(x,y)$ 를 갱신하는 것으로, $h(x,y)$ 에 K 를 곱하여 계산한다.

$$h^n(x,y) = K * h^{n-1}(x,y) \tag{5}$$

식 (5)에서 n 은 반복 횟수를 나타낸다. K 는 3×3의 크기를 가지며 각각의 값은 그림 9과 같다.

0	1	0
1	0	1
0	1	0

그림 9. K

마지막 단계로, 목표영역의 최종 화소값은 반복을 통해 갱신 된 $h^n(x,y)$ 값과 최적밴드 내부의 화소값을 곱하여 계산한다.

$$\hat{f}(x,y) = h^n(x,y)g(x,y) \tag{6}$$

그림 10은 밴드 매칭만 한 결과영상과 밴드 매칭 후 경계제거를 한 결과영상이다. 왼쪽 줄은 밴드 매칭만 한 결과영상으로 채워진 목표영역과 주변영역사이에 뚜렷한 경계가 나타나는 것이 보인다. 오른쪽 줄은 밴드 매칭 후 경계제거를 한 결과영상으로 경계가 제거된 것을 볼 수 있다. 밴드 매칭과 경계제거를 수행한 계산시간은 각각 약 44, 150초이고, [3]의 방법의 계산시간은 각각 약 960, 1260초이다.

3. 목표영역분할과 결과의 순서 정하기

2절에서 제안한 인페인팅 방법은 한 개의 밴드를



그림 10. 밴드 매칭 후 경계제거를 한 결과 : 왼쪽 줄은 밴드 매칭만 한 결과, 오른쪽 줄은 밴드 매칭 후 경계제거를 한 결과이다. 전체 계산시간은 각각 약 44, 150초이고, [3]의 방법의 계산시간은 각각 약 960, 1260초이다.

사용한다. 한 개의 밴드를 사용하여 인페인팅을 할 때, 그림 11(a)와 같이 목표영역으로 바다의 수평선과 같은 선이 통과하는 경우 선이 불일치될 수 있다. 또한 목표영역이 영상에 비해 클 경우 영상 내에서 최적밴드를 찾을 수도 있다. 따라서 이런 문제를 해결하기 위해 이 논문에서는 그림 11과 같이 목표영역을 일정한 크기로 분할하고, 각 분할 영역에 대해 제안하는 인페인팅 즉, 밴드 매칭과 경계제거를 수행한다. 목표영역분할시 분할 개수에 따라 여러 개의 다른 결과들을 얻을 수 있다. 이 논문에서는 사용자가 여러 결과들 중 좋은 결과를 선택하는데 도움을 주기 위해 결과들의 순서를 정하는 방법을 제안한다.

영역분할개수는 가로방향과 세로방향의 개수로 정해진다. 그림 11은 가로방향으로 두 부분, 세로방향으로 두 부분으로 나누는 경우의 예이다. 이런 경우 최대 네 개의 결과를 얻을 수 있다. 먼저 그림 11(a)와 같이 영역분할을 하지 않은 결과, 두 번째로 그림 11(b), (c)와 같이 목표영역을 각각 가로방향, 세로방향으로 이등분한 결과, 마지막으로 목표영역을 가로방향과 세로방향으로 모두 이등분하여 총 사등분한 결과를 얻을 수 있다. 그림 11(c)와 그림 11(d)에서 보듯이 영역분할에 의해 목표영역과 주변영역사이의 선이 일치되어 자연스럽게 보이는 것을 볼 수 있다. 그림 11의 계산시간은 순서대로 약 10, 19, 22, 37초로 4장의 이미지를 얻는데 총 88초가 소요되었다. 반면 [3]의 방법의 경우 하나의 이미지를 얻는데 패치크기

가 7×7인 경우 1550초, 11×11인 경우 723초가 소요되었다. 사용자 입장에서는 4(1+3)개의 후보들 중에서 가장 적합한 것 하나를 선택할 수 있다. 영역 분할 방법의 기본 목적은 영상의 구조에 의해 초래되는 불연속경계를 제거하기 위한 것이다. 아울러 사용자에게 다양한 인페인팅 결과물을 빠른 시간 내에 보여줌으로써 선택의 폭을 넓히게 한다. 순서정하기는 사용자의 선택 과정에 도움을 주기 위해서이다.

영역분할개수에 따른 여러 결과들의 순서를 정하기 위해 이 논문에서는 두 가지 방법을 제안한다. 첫 번째는 원본영상 즉, 인페인팅 영역을 표시하기 전 영상과 인페인팅한 결과영상들의 에지(edge) 영상을 이용하는 방법이다. 두 번째는 에지영상대신에 세그멘테이션(segmentation) 영상을 이용하는 방법이다. 두 방법은 인페인팅 영역 주위의 에지나 색상이 얼마나 일치하는지를 계산하여 구조적 연속성을 측정하는 것이다. 사용자는 두 방법을 이용하여 정한 결과들의 순서를 토대로 가장 좋은 결과를 선택할 수 있다.

에지영상을 이용하여 결과의 순서를 정하는 방법은 다음과 같다. 첫 번째로, 소벨필터를 통해 원본영상과 결과영상들의 에지영상을 구한다. 에지영상에서 에지는 흰색으로 표시하고, 나머지 부분은 검은색으로 표시한다. 그림 12(b)는 원본영상의 에지영상이고, 그림 13(b)는 인페인팅한 결과영상의 에지영상이다. 두 번째로, 원본영상과 인페인팅한 결과영상들을 비교하기 위한 영역을 정의하고, 그 영역을 비교영역

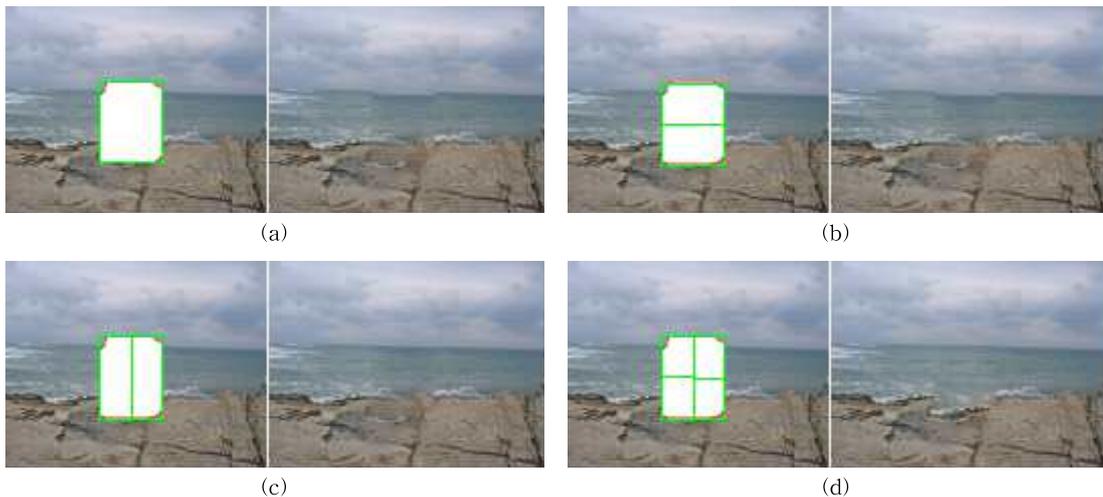


그림 11. 6목표영역분할 방법과 그 결과 : (a) 영역분할 하지 않음 (b) 가로방향으로 이등분할 (c) 세로방향으로 이등분할 (d) 가로, 세로방향으로 이등분하여 사등분할

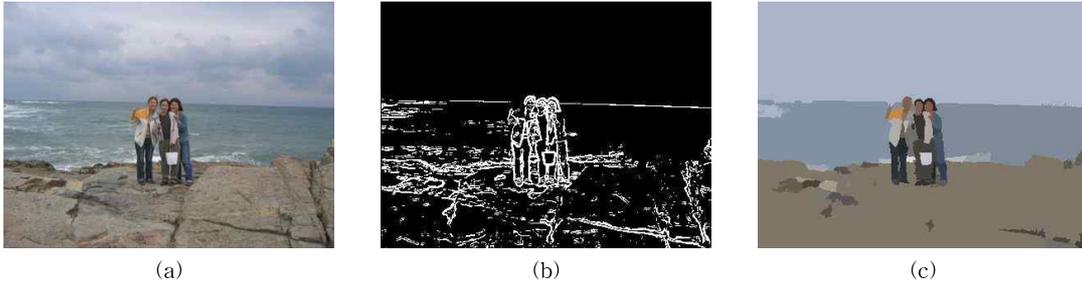


그림 12. (a) 원본영상 (b) 에지영상 (c) 세그먼테이션 영상

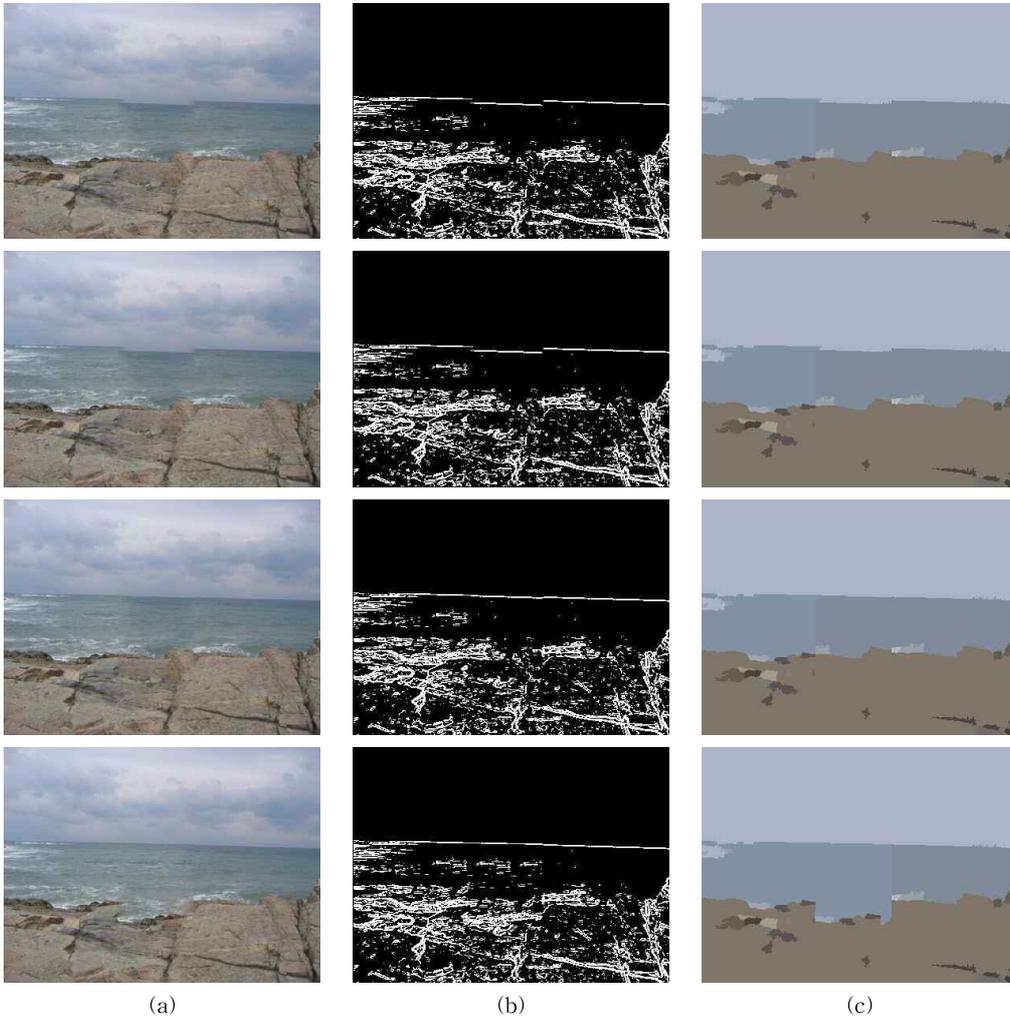


그림 13. (a) 영상분할 후 인페인팅한 결과영상 (b) 에지영상 (c) 세그먼테이션 영상

이라 한다. 비교영역은 목표영역의 경계와 그 경계의 앞뒤 화소들로 정의한다. 세 번째로, 비교영역에 대해 원본영상과 각 인페인팅한 결과영상의 에지영상

이 얼마나 일치하는가를 계산하고, 그 값을 $u(x,y)$ 라 한다. $u(x,y)$ 는 두 영상간의 화소값이 같으면 1, 다르 면 0으로 설정한다. 마지막으로 비교영역의 모든 화



그림 14. 윗줄은 에지영상을 이용한 결과의 순서이고, 아랫줄은 세그먼테이션 영상을 이용한 결과의 순서이다.

소의 $u(x,y)$ 를 더하여 S_k 를 계산한다. S_k 는 목표영역 분할로 생성한 k번째 결과영상의 순서를 정하기 위한 값이다. 결과영상들의 순서는 S_k 의 크기순으로 정한다. 즉, S_k 가 큰 것이 가장 좋은 결과이다.

$$u(x,y) = \begin{cases} 1 & o(x,y) = r(x,y) \\ 0 & o(x,y) \neq r(x,y) \end{cases} \quad (7)$$

$$S_k = \sum_{i=1}^n u_i \quad (8)$$

식 (7)에서 (x,y) 는 비교영역의 화소위치이고, $o(x,y)$ 는 원본영상의 에지영상 값, $r(x,y)$ 는 결과영상의 에지영상 값이다. 식 (8)에서 k 는 목표영역분할 방법에 따른 결과영상들의 인덱스이고, n 은 비교영역의 총 화소수이다.

에지영상 대신 세그먼테이션 영상을 이용하여 결과영상의 순서를 정하는 방법은 에지영상을 이용하는 것과 거의 동일하다. $u(x,y)$ 를 계산할 때 에지영상 값 대신 세그먼테이션 영상 값을 사용한다. 그 외에 나머지 계산은 에지영상을 이용하는 방법과 동일하다. 그림 12(c)는 원본영상의 세그먼테이션 영상이고, 그림 13(c)는 인페인팅한 결과영상의 세그먼테이션 영상이다.

그림 14에서 윗줄은 에지영상을 이용하여 결과영상의 순서를 정한 것이고, 아랫줄은 세그먼테이션 영상을 이용하여 결과영상의 순서를 정한 것이다. 가장 왼쪽에 있는 영상이 가장 좋은 결과이고, 가장 오른쪽에 있는 영상이 가장 좋지 않은 결과이다. 그림 14의 윗줄과 아랫줄의 결과를 보면 에지영상을 이용한 결과영상의 순서와 세그먼테이션 영상을 이용한 결

과영상의 순서에 많은 차이가 없음을 볼 수 있다. 에지영상과 세그먼테이션 영상을 이용하여 결과영상의 순서를 정하는데 걸리는 시간은 약 0.5초이다.

4. 실험

제안하는 인페인팅 방법과 기존 방법을 비교하기 위해 다양한 영상을 가지고 실험을 하였다. 모든 실험은 펜티엄IV 3.0GHz에 2GB RAM을 가진 컴퓨터에서 테스트하였으며, 실험에 사용된 밴드의 두께는 5화소이다. 표 1은 제안하는 방법과 [3]의 방법의 계산시간을 비교한 것이다. 1번은 그림 7의 가운데 영상으로 제안하는 방법 중 밴드 매칭의 수행시간이고, 2번은 그림 10의 위쪽 영상으로 밴드 매칭 후 경계제거까지 하는데 걸린 시간이다. 마지막으로 3번은 그림 11을 영역분할 후 각 영역에 대해 밴드 매칭과 경계제거를 하는데 걸린 시간으로 총 4장의 결과를 얻고 그 결과의 순서를 정하는 것까지 수행하는데 걸린 시간이다. 1번과 2번은 [3]의 방법에서 패치 크기를 7×7로 설정한 것이고, 3번은 11×11로 설정하여 계산한 것이다.

표 1. 제안하는 방법과 [3]의 계산시간 비교

	1번	2번	3번
영상 크기	640 × 480	640 × 480	400 × 300
인페인팅 영역	8,881화소	7,988화소	11,091화소
제안하는 방법의 시간	26초	44초	1분 30초
[3]의 시간	13분	16분	12분



그림 15. 왼쪽은 원본 영상, 가운데는 [3]의 결과, 오른쪽은 목표영역을 6등분하여 제안하는 방법을 수행한 결과이다. 계산시간은 각각 1300초, 90초이다.

그림 15는 [3]에 나온 예제를 비교한 것으로, 영상 크기는 416×316이고, 인페인팅 영역은 28,349화소이다. [3]은 15×15의 패치로 수행하였으며 계산시간은 약 1300초이고, 제안하는 방법은 목표영역을 6등분하여 수행한 결과로 계산시간은 약 95초이다. [3]의 결과는 수평선이 고르지 않지만 제안하는 방법은 수평선이 자연스럽게 나타난다. 그림 15에서 보듯이 제안하는 방법은 빠른 시간 안에 더 나은 결과를 얻을 수 있다.

5. 결 론

이 논문에서는 밴드매칭과 경계제거를 이용한 인페인팅 방법을 제안하였다. 기존의 인페인팅 방법은 목표영역을 작은 여러 개의 영역으로 나누어 각각의 영역들에 대해 영상 내에서 최적의 영역을 찾아 채웠기 때문에 오랜 계산시간이 소요되는 단점이 있었다. 이를 개선하고자 제안하는 인페인팅 방법은 하나의 밴드를 가지고 매칭을 통해 최적의 영역을 찾아 이전 방법보다 수행시간을 줄일 수 있었고, 경계제거를 통해 목표영상과 주변 영상사이에 나타날 수 있는 색상 밝기의 불일치를 제거하여 자연스러운 결과를 얻을 수 있었다. 제안하는 방법은 기본적으로 밴드매칭과 경계제거로 인페인팅을 하였지만, 목표영역이 크거나 구조적 연속성으로 인해 결과가 좋지 못할 경우 목표영역을 분할하여 각각의 분할영역에 대해 제안하는 방법의 인페인팅을 수행하였다. 이 경우 분할영역 개수에 따라 여러 다른 결과를 얻었고, 사용자에 의해 그중 좋은 결과를 선택할 수 있다.

참 고 문 헌

[1] M. Bertalmio, G. Sapiro, V. Caselles, and C.

Ballester, "Image Inpainting," *In Proc. ACM Transactions on Graphics, SIGGRAPH*, pp. 417-424, 2000.

[2] T.F. Chan, and J. Shen, "Mathematical Models for Local Nontexture Inpaintings," *Society for Industrial and Applied Mathematics Journal on Applied Mathematics*, Vol. 62, No. 3, pp. 1019-1043, 2002.

[3] A. Criminisi, P. Perez, and K. Toyama, "Object Removal by Exemplar-Based Inpainting," *In Proc. Conference on Computer Vision and Pattern Recognition*, pp. 721-728, 2003.

[4] I. Drori, D. Cohen-Or, and H. Yeshurun, "Fragment-Based Image Completion," *In Proc. ACM Transactions on Graphics, SIGGRAPH*, pp. 303-312, 2003.

[5] N. Komodakis, and G. Tziritas, "Image Completion using Global Optimization," *In Proc. Conference on Computer Vision and Pattern Recognition*, pp. 442-452, 2006.

[6] J. Shen, X. Jin, C. Zhou, and C.C.L. Wang, "Gradient Based Image Completion by Solving Poisson Equation," *Pacific Rim Conference on Multimedia*, Vol. 3767, pp. 257-268, 2005.

[7] M. Wilczkowiak, G.J. Brostow, B. Tordoff, and R. Cipolla, "Hole Filling Through Photomontage," *British Machine Vision Conference*, pp. 492-501, 2005.

[8] H. Yamauchi, J. Gaber, and H.P. Seidel, "Image Restoration using Multiresolution

Texture Synthesis and Image Inpainting,” *In Proc. Computer Graphics International*, pp. 120-125, 2003.

- [9] A.A. Efros, and T.K. Leung, “Texture Synthesis by Non-Parametric Sampling,” *In Proc. International Conference on Computer Vision*, pp. 1033-1038, 1999.
- [10] V. Kwatra, A. Schodl, I. Essa, Greg Turk, and Aaron Bobick, “Graphcut Textures: Image and Video Synthesis using Graph Cuts,” *In Proc. ACM Transactions on Graphics, SIGGRAPH*, pp. 277-286, 2003.
- [11] T. Georgiev, “Covariant Derivatives and Vision,” *In Proc. European Conference on Computer Vision*, pp. 56-69, 2006.
- [12] P. Perez, M. Gangnet, and A. Blake, “Poisson Image Editing,” *In Proc. ACM Transactions on Graphics, SIGGRAPH*, pp. 313-318, 2003.



이 수 빈

2004년 2월 성신여자대학교 컴퓨터정보학부 멀티미디어학과 학사
 2006년 2월 서강대학교 영상대학원 미디어공학과 석사
 2006년 3월 현재 서강대학교 영상대학원 미디어공학과 박사

사과정

관심분야 : 인페인팅, 영상기반 렌더링, 증강현실



서 용 덕

1992년 2월 경북대학교 전기전자공학과 학사
 1994년 2월 포항공과대학교 대학원 전기전자공학과 석사
 2000년 2월 포항공과대학교 대학원 전기전자공학과 박사
 2003년~현재 서강대학교 영상대학원 미디어공학과 교수

주관심분야 : 컴퓨터 비전, 증강현실, 영상기반 3차원 모델 복원, 카메라 보정, 최적화