일반논문 (Regular Paper)
방송공학회논문지 제21권 제5호, 2016년 9월 (JBE Vol. 21, No. 5, September 2016)
http://dx.doi.org/10.5909/JBE.2016.21.5.782
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

영상 특성과 스켈레톤 분석을 이용한 실시간 인간 객체 추출

김 민 준[a)], 이 주 철[b)], 김 원 하[a)‡]

Realtime Human Object Segmentation Using Image and Skeleton Characteristics

Minjoon Kim[a)], Zucheul Lee[b)], and Wonha Kim[a)‡]

요 약

영상에서 배경으로부터 객체를 추출하는 영상 segmentation 알고리즘은 물체 인식 및 추적 등 다양한 응용분야에서 활용될 수 있다. 본 논문에서는 고정된 카메라에서 다수의 초기 프레임을 참조하여 실시간 객체 segmentation 방법을 제안한다. 먼저 객체와 배경을 분류하는 확률모델을 제안하였으며 초기 프레임 동안에 카메라의 color consistency와 focus 특성을 분석하여 안정적인 segmentation 성능을 증가시켰다. 또한 분류된 객체에서 human의 skeleton 특성을 이용하여 추출 결과를 보정하는 방법을 제안한다. 마지막으로 제안된 알고리즘은 객체 segmentation 실시간 처리를 위하여 복잡도를 최소화하므로 다양한 mobile 단말에 확대 적용 가능하다.

Abstract

The object segmentation algorithm from the background could be used for object recognition and tracking, and many applications. To segment objects, this paper proposes a method that refer to several initial frames with real-time processing at fixed camera. First we suggest the probability model to segment object and background and we enhance the performance of algorithm analyzing the color consistency and focus characteristic of camera for several initial frames. We compensate the segmentation result by using human skeleton characteristic among extracted objects. Last the proposed method has the applicability for various mobile application as we minimize computing complexity for real-time video processing.

Keyword : segmentation, camera, human, composite

a) 경희대학교 전자정보대학 전자전파공학과(Kyung Hee University)
b) KT 융합기술원(KT Fusion Technology Institute)
‡ Corresponding Author : 김원하(Wonha Kim)
E-mail: wonha@khu.ac.kr
Tel: +82-31-201-2030
ORCID: http://orcid.org/0000-0002-4797-3526
※ 본 연구는 한국연구재단 기초연구 사업(NRF-2015R1D1A1A01059722)의 지원받아 수행됨.
· Manuscript received June 3, 2016; Revised Setember 20, 2016; Accepted Setember 20, 2016.

Ⅰ. 서 론

영상을 객체 별로 분류하거나 관심 있는 영역으로 나누는 작업을 영상 Segmentation이라고 한다. 일반적으로 영상에서 물체를 추출하거나 영상의 전경/배경을 나누는 영상처리 기법은 여러 응용분야에서 사용되고 있다. 대표적으로 intensity 정보만을 사용하는 영상 Segmentation 방법이

있다. [1],[2],[3]에서는 프레임간 비교를 통해 화소 단위로 Mixture of Gaussian 모델을 생성해서 배경을 분리하며 [9],[10]에서는 각각 Bayesian decision 모델, Kernel density 모델을 이용하여 영상 Segmentation하는 방법을 제시하였다. 하지만 이러한 방법들은 intensity 정보만을 사용하기 때문에 충분한 성능을 제공하지 못하는 단점이 있다. 이에 알고리즘의 성능을 개선시키기 위해 [4],[5],[6],[7],[8]에서는 texture 정보를 추가적인 feature 정보로 사용하였다. 이를 통해 알고리즘의 성능은 향상되었으나 feature들을 사용함에 따라 알고리즘 복잡도가 높아졌다. [11]에서는 복잡도가 높은 알고리즘에서 하드웨어적인 향상을 구현하였지만 추가적인 장치가 요구된다. 위와 같이 단일 영상에서 소프트웨어적인 연산만으로 물체를 분류하는 방식과 다르게 영상의 깊이 정보를 이용하는 연구도 이루어지고 있다. [12],[13],[14],[15],[16]에서는 깊이 센서나 stereo 영상으로 깊이 정보를 얻음으로써 간단하게 객체와 배경을 분리하는 방법을 제시한다. 그러나 깊이 센서라는 추가적인 하드웨어가 요구되기 때문에 장치 가격은 증가하게 된다. 또한 깊이 센서의 기술적인 한계로 저해상도의 깊이 영상을 제공함으로써 실제 색 영상과의 합성이 불가피하게 되었다.

앞서 소개한 논문과 비교하여 본 논문에서는 추가적인 장치 없이 단일 카메라로 실시간 영상 segmentation 알고리즘을 제안한다. 제안하는 알고리즘은 다수의 초기 프레임으로 카메라 color consistency와 focus 특성을 분석하여 참조 프레임과 현재 프레임의 intensity, 색 그리고 texture정보 비교를 통해 영상 segmentation을 수행한다. 또한, 기존의 이진분류 방식과 다르게 전경과 배경을 확률적으로 분류함으로써 이진분류에서 나타나는 under/over 분류 문제를 감소시킨다. 본 논문에서는 실시간 처리를 위해 block 단위로 영상 segmentation을 수행하며 낮은 복잡도로 색, texture를 분석하는 방법을 제안한다. 마지막으로 human face 인식과 skeleton 알고리즘을 통해 영상 segmentation의 성능을 향상시켰다. 본 논문은 다음과 같이 구성되어 있다. 두 번째 장은 영상 특성과 카메라 특성을 활용한 영상 segmentation 방법을 제안한다. 세 번째 장은 human skeleton 알고리즘을 활용하여 추출된 객체 결과를 보정하는 방법을 소개한다. 네 번째 장은 예시 영상을 통한 시뮬레이션 결과 비교 및 실제 알고리즘의 복잡도를 제시하고, 다섯 번째 장에서 결과 분석을 통해 결론을 내린다. 그림 1은 전체 흐름도를 나타낸다.

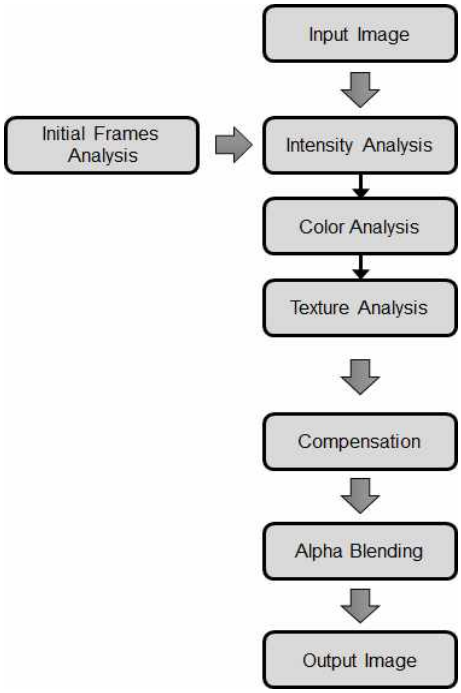


그림 1. 전체 흐름도
Fig 1. Overall Flowchart

Ⅱ. 영상 및 카메라 color/focus 특성 분석 기반 영상 segmentation 방법

공간영역에서 영상의 객체를 추출하기 위하여 초기 프로세스로 1초 동안의 영상 프레임을 분석, 통계화하여 더욱 정확한 참조 프레임 정보를 확보할 수 있다. 특히 여러 프레임을 분석하면 1장의 프레임을 이용한 방법과 다르게 카메라의 color consistency와 focus 특성을 고려하여 안정적인 segmentation이 가능하다. 이렇듯 새로운 물체가 유입되지 않은 참조 프레임의 정보와 현재 프레임의 정보를 비교함으로써 현재 프레임에 새롭게 출현한 물체를 추출 할 수 있다. 본 논문에서는 3단계(영상의 Intensity, 색, Texture) 객체 추

출방법을 제안한다. 1단계 영상 Intensity 차이를 통해 객체로 확실하게 분류되지 못한 영역은 2단계인 색 차이를 통해 다시 한 번 검증되고, 그래도 객체로 분류되지 못한 영역은 최종 3단계로 넘어가는 cascade 알고리즘 구조를 갖는다.

1. Intensity 차이를 이용한 객체 추출 방법

색상과 Texture의 차이를 분석하기에 앞서 가장 먼저 참조 프레임과 현재 프레임의 intensity 차이를 분석한다. 다만 참조 프레임은 초기 30장 프레임의 각 화소별 평균 intensity 값을 포함하며 이때 참조 프레임의 평균 intensity와 현재 프레임의 intensity 차이를 계산하게 된다. 식(1)은 참조 프레임과 현재 프레임의 $N\times N$ block 내부에서 화소별 intensity 차이의 평균을 측정하는 식으로 ∇I는 임의 block의 대표 intensity 차이를 나타낸다. 예를 들어 k번째 block의 대표 intensity 차이의 값은 ∇I_k가 된다. I^{ref}는 참조 프레임의 $N\times N$ intensity block 이고 I^{cur}는 현재 프레임의 $N\times N$ intensity block 이다.

$$\nabla I = \frac{1}{N^2}\sum_{i=0}^{N}\sum_{j=0}^{N}|I^{ref}_{(i,j)} - I^{cur}_{(i,j)}| \tag{1}$$

참조 프레임과 현재 프레임의 차이가 경험적으로 설정된 큰 상수 임계값(Thr_0)을 초과하면 Foreground로 분류하고 더 이상 2단계, 3단계 재검증 절차를 진행하지 않는다. Foreground에 속하지 않은 영역에 대해서는 3가지 영역(Foreground, Background, Fuzzy)으로 분류하여 기존의 이진분류 방법의 under/over 오류 estimation을 최소화한다. 위의 처리를 확률로 표현하면 k번째 block이 Foreground일 경우 $P_{Foreground,\,I_k}=1$ 로 설정되고 Background일 경우 $P_{Foreground,\,I_k}=0$ 로 설정된다. 3번째 애매한 영역인 Fuzzy region은 $P_{Foreground,\,I_k}=[0,1]$의 값을 갖게 된다. 확률식으로 나타내면 다음 식(2)와 같다.

$$P_{Foreground,\,I_k} = \begin{cases} 1 & \nabla I \geq Thr_1 \\ \dfrac{1}{Thr_2 - Thr_1}(\nabla I - Thr_1) & Thr_1 > \nabla I \geq Thr_2 \\ 0 & otherwise \end{cases} \tag{2}$$

식 (3)과 같이 Foreground, Background, Fuzzy region을 구분하는 2개의 변수 임계값을 참조 프레임의 intensity에 따라 적응적으로 결정한다. 참조 프레임의 intensity 크기가 클수록 큰 변수 임계값을 갖도록 한다. 이렇게 함으로써 단순한 상수 임계값을 사용하였을 때 보다 물체의 그림자 부분에서 적절한 영상 분리 결과를 얻을 수 있다.

$$\begin{aligned} l_1 &: Thr_1 = a_1 I^{ref} + b_1 \\ l_2 &: Thr_2 = a_2 I^{ref} + b_2 \end{aligned} \tag{3}$$

여기서 a_1, a_2, b_1, b_2는 임계값 계수로 그림 2에서 l_1, l_2의 기울기와 y절편을 나타낸다. 이때 그림 2는 참조 프레임의 intensity에 따른 임계값(l_1, l_2)과 Foreground, Background, Fuzzy region의 영역 분류를 나타낸다.

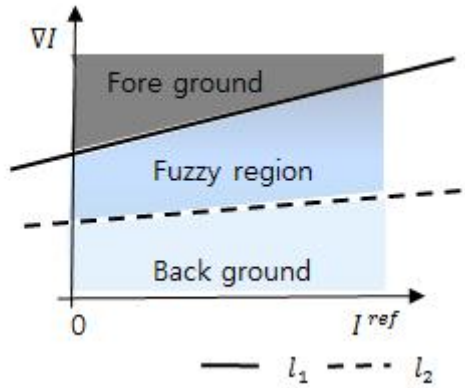


그림 2. 적응적인 임계값에 따른 영역 구분
Fig 2. Region depend on Adaptive Threshold

2. 색 차이를 이용한 객체 추출 방법

1단계 Intensity 차이를 이용한 객체 분류에서 Foreground로 분류되지 않은 영역에 대하여 2단계 색 차이 객체 추출 방법을 적용한다. 일반적으로 color 영상은 RGB 3개의 채널을 가지기 때문에 식(4)처럼 3개의 성분을 가진 벡터로 나타낼 수 있다. block의 색 차이는 두 block의 color 벡터가 이루는 각도의 차이를 계산하여 나타낼 수 있다. 하지만 식(5)와 같이 두 벡터의 내적을 이용하여 두 벡터의 사잇각을 구하는 방식은 $\cos^{-1}$연산이 필수적이므로 연산의

복잡도를 높여 실시간 처리 시스템에서는 비효율적이다.

$$\overrightarrow{v^{ref}} = < R^{ref}, G^{ref}, B^{ref} > \\ \overrightarrow{v^{cur}} = < R^{cur}, G^{cur}, B^{cur} > \quad (4)$$

$$\theta = Cos^{-1}\left(\frac{\overrightarrow{v^{ref}} \bullet \overrightarrow{v^{cur}}}{|v^{ref}||v^{cur}|}\right) \quad (5)$$

식 (5)와 같이 두 벡터의 정확한 사잇각을 구하는 대신 식(6)과 같이 두 벡터의 단위 벡터의 차를 이용하여 두 block의 색 차이정도를 근사시킬 수 있다. 색 차이를 구하는 두 방식은 그림 3과 같이 동일한 분포 결과를 나타냄을 확인할 수 있다. R^{ref}, G^{ref}, B^{ref}는 참조 프레임에서 각 채널의 block 평균 화소값(Average pixel value)이고 R^{cur}, G^{cur}, B^{cur}는 현재 프레임의 값이다.

$$\nabla C = \left|\frac{R^{ref}}{I^{ref}} - \frac{R^{cur}}{I^{cur}}\right| + \left|\frac{G^{ref}}{I^{ref}} - \frac{G^{cur}}{I^{cur}}\right| + \left|\frac{B^{ref}}{I^{ref}} + \frac{B^{cur}}{I^{cur}}\right| \quad (6)$$

(a) (b)

그림 3. 색 차이 비교 (a) 벡터의 사이각 (b) 단위벡터의 차
Fig 3. Comparison of Color Difference (a) Angle between Vectors (b) Subtraction between Unit Vector

참조 프레임과 현재 프레임 사이의 색 차이를 이용하여 영상을 분류하기 위해 임계값을 구하는 방법은 아래와 같다.

$$l_1 : Thr_1 = a_1 \overline{\nabla C^{ref}} + b_1 \\ l_2 : Thr_2 = a_2 \overline{\nabla C^{ref}} + b_2 \quad (7)$$

$\overline{\nabla C^{ref}}$는 n장의 초기 프레임에서 연속된 프레임 간의 block별 평균 색 차이를 의미한다. 그 식은 다음과 같다.

$$\overline{\nabla C^{ref}} = \frac{1}{n}\sum_{i=1}^{n}\left(\left|\frac{R_i}{I_i} - \frac{R_{i-1}}{I_{i-1}}\right| + \left|\frac{G_i}{I_i} - \frac{G_{i-1}}{I_{i-1}}\right| + \left|\frac{B_i}{I_i} + \frac{B_{i-1}}{I_{i-1}}\right|\right) \quad (8)$$

이렇게 함으로써 물체의 유입이 없는 상태에서 미묘하게 색을 다르게 인지하는 카메라 color consistency 특성을 반영하여 적응적인 임계값을 설정할 수 있다. 따라서 색 차이를 이용하여 전경일 확률은 $P_{Foreground,C}$로 식(9)와 같이 구할 수 있다.

$$P_{Foreground,C} = \begin{cases} 1 & \nabla C \geq Thr_1 \\ \frac{1}{Thr_2 - Thr_1}(\nabla C - Thr_1) & Thr_1 > \nabla C \geq Thr_2 \\ 0 & otherwise \end{cases} \quad (9)$$

3. Texture 차이를 이용한 객체 추출 방법

전 단계인 색 차이를 이용한 객체 추출방법에서 여전히 Foreground 영역으로 분류되지 못한 영역에 대하여 3단계인 Texture 차이 객체 추출 방법을 적용한다. Texture 정도는 참조 프레임의 intensity와 현재 프레임의 intensity를 이용하여 구하게 된다. 각 프레임의 $N \times N$ intensity block의 평균값과 block 내 각 화소별 차이를 합함으로써 block 내의 texture 정도를 식(10)과 같이 나타낼 수 있다.

$$T = \sum_{i=0}^{N}\sum_{j=0}^{N}\left|(I_{(i,j)} - \mu)\right| \quad (10)$$

여기서 $I_{(i,j)}$는 block 내 (i,j) 위치에 해당하는 화소값을 나타내며 μ는 해당 block의 평균값을 나타낸다. 따라서 참조 프레임과 현재 프레임의 texture 차이는 다음과 같다.

$$\nabla T = \left|T^{ref} - T^{cur}\right| \quad (11)$$

T^{ref}와 T^{cur}는 각 참조 프레임과 현재 프레임에서의 block texture를 의미한다. 참조 프레임과 현재 프레임 사이의 texture 차이를 이용하여 영상을 분류하기 위한 임계값은 식(12)을 통해 구한다.

$$\begin{aligned} l_1 : Thr_1 &= a_1 Var(\overline{\mu^{ref}}) + b_1 \\ l_2 : Thr_2 &= a_2 Var(\overline{\mu^{ref}}) + b_2 \\ \mu^{ref} &= \{\mu_1^{ref}, \mu_2^{ref}, \mu_3^{ref}, \cdots, \mu_n^{ref}\} \end{aligned} \quad (12)$$

$\overline{\mu^{ref}}$는 n장의 초기 프레임에서 동일한 위치에 해당하는 intensity 값의 block 평균값이다. 초기 프레임의 $Var(\overline{\mu^{ref}})$를 구함으로써 카메라의 auto focus 특성에 의한 오검출을 감소시킬 수 있다.

위와 같이 3단계(Intensity, 색, texture) 차이를 이용한 영상 분류 결과는 그림 4와 같다.

(a) (b)

그림 4. 영상분류 결과 (a) 원본 영상 (b) 확률 분류 영상
Fig 4. Segmentation Result (a) Original Image (b) Probability Segmentation Image

Intensity, 색, texture 정보를 바탕으로 새롭게 유입된 물체에 대하여 올바르게 객체를 추출 할 수 있지만 그림 4.(b)와 같이 under/over 오류 estimation 영역이 여전히 존재한다. 그래서 다음 장에서는 새로 유입된 물체가 사람인 경우에 대하여 under/over 오류 estimation 영역을 보정하는 방법을 제시한다.

Ⅲ. Human face 인식과 skeleton 특성 기반 영상 분류 보정기법

이 절에서는 현재 프레임에서 새로 유입된 물체가 사람인 경우 전 단계에서 얻은 분류 결과를 보정하기 위해 Face 검출[17] 및 Skeleton 추출[18]방법을 활용한 보정기법을 제안한다.

1. Face 검출을 활용한 Human 객체 인지

본 장에서는 Intensity, 색, texture 차이를 이용한 객체 추출 결과에서 객체로 분류된 영역이 human 객체인지를 분석한다. Human의 여부를 판단하기 위하여 classification and regression tree analysis(CART)에 기반을 둔 face 검출[17] 알고리즘을 사용한다. 이 알고리즘을 통해 human으로 판단된 물체가 존재하면 객체 분류 결과에 labeling을 적용하여 human을 제외한 나머지 객체는 결과에서 제외시킴으로서 human 객체 추출의 정확도를 향상시킨다. 그림 5는 객체 분류 결과 및 human 검출을 적용시킨 후의 결과를 보

(a) (b) (c)

그림 5. Face 검출을 이용한 영상분류 보정 (a) 확률분류 영상 (b) Face 검출 영상 (c) 보정 결과 영상
Fig 5. Compensation Using Face Detection (a) Probability Segmentation Image (b) Face Detection Image (c) Compensation Result Image

여준다. Human 이외에 객체로 분류된 영역들이 face 검출 방법을 통해 human 객체에서 제외되는 것을 관찰할 수 있다.

2. Skeleton을 이용한 객체 보정

그림 6은 human 내부에 존재하는 오류 영역을 빨간색 원으로 나타낸 것이다. Ⅱ장에서의 영상 신호 분석으로 객체로 분류하지 못한 오류 영역에 대해서 본 절에서는 human 객체의 skeleton 특성을 이용하여 오류 결과를 보정한다.

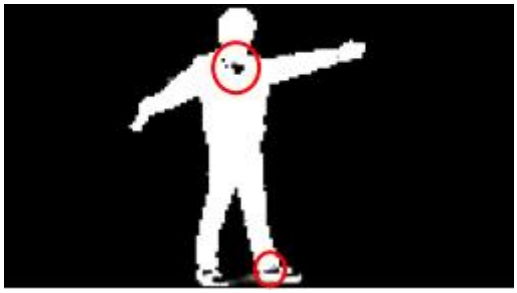


그림 6. 확률분류 영상
Fig 6. Probability Segmentation Image

예시 영상에서 객체로 분류된 영역에 대하여 Morphological 연산[18]을 적용시킴으로서 human 객체에 대한 skeleton을 얻을 수 있다. 얻어진 skeleton은 human의 중심부를 잇는 선들로서 skeleton의 일정 주변은 human 객체로 분류될 수 있을 것이다. Face 검출을 통해 얻은 face의 크기를 이용하여 dilation window를 결정하고 skeleton line을 따라 보정할 객체 영역이 결정된다. 이때 skeleton의 최외각부분은 실제 human의 최외각부분과 일치하므로 Dilation을 적용하면 실제 객체 영역을 벗어나는 오류가 발생한다. 이 오류를 회피하기 위해서 skeleton line 끝부분에서는 dilation window 내 skeleton 화소 개수의 임계값을 지정하여 화소 개수가 작은 경우 dilation을 하지 않는다.

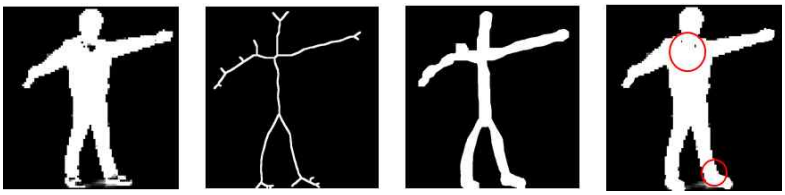
그림 7은 human의 skeleton 특성을 활용하여 객체로 보정할 영역과 이 알고리즘을 통해 human 객체로 보정된 결과를 보여준다. 그림 7.(d)와 그림 7.(a)의 영상 분류결과를 비교하면 human 객체 내부에서 배경으로 잘못 결정된 오류 영역이 객체로 보정되어 그림 7.(d)의 영상 분류결과가 향상됨을 확인 할 수 있다.

위의 과정을 통해 얻은 향상된 객체 분류 결과에 평균 필터를 적용함으로써 [19]의 Alpha Blending의 Alpha 값으로 활용한다. 이 방법은 영상을 새로운 배경에 합성할 경우, block 단위의 영상 분류 및 합성보다 더 자연스러운 효과를 얻을 수 있다.

Ⅳ. 실 험

그림 8은 기존 방법 중에서 texture정보를 분석하는 feature로써 주로 사용하는 LBP(Local Binary Pattern)의 Histogram방법[5]과 DCT(Discrete Cosine Transform)방법을 적용하여 이진 분류한 결과와 제안하는 방법으로 분류한 결과이다. LBP 검사 화소 는 1 화소이며 비트 수는 8비트로 LBP를 설계하였다.

표 1은 제안한 방법의 성능을 검증하기 위해 각 영상의



(a) (b) (c) (d)

그림 7. Skeleton을 이용한 영상분류 보정 (a) 확률분류 영상 (b) Skeleton 영상 (c) 보정영역 영상 (d) 보정 결과 영상
Fig 7. Compensation Using Skeleton (a) Probability Segmentation Image (b) Skeleton Image (c) Compensation Region Image (d) Compensation Result Image

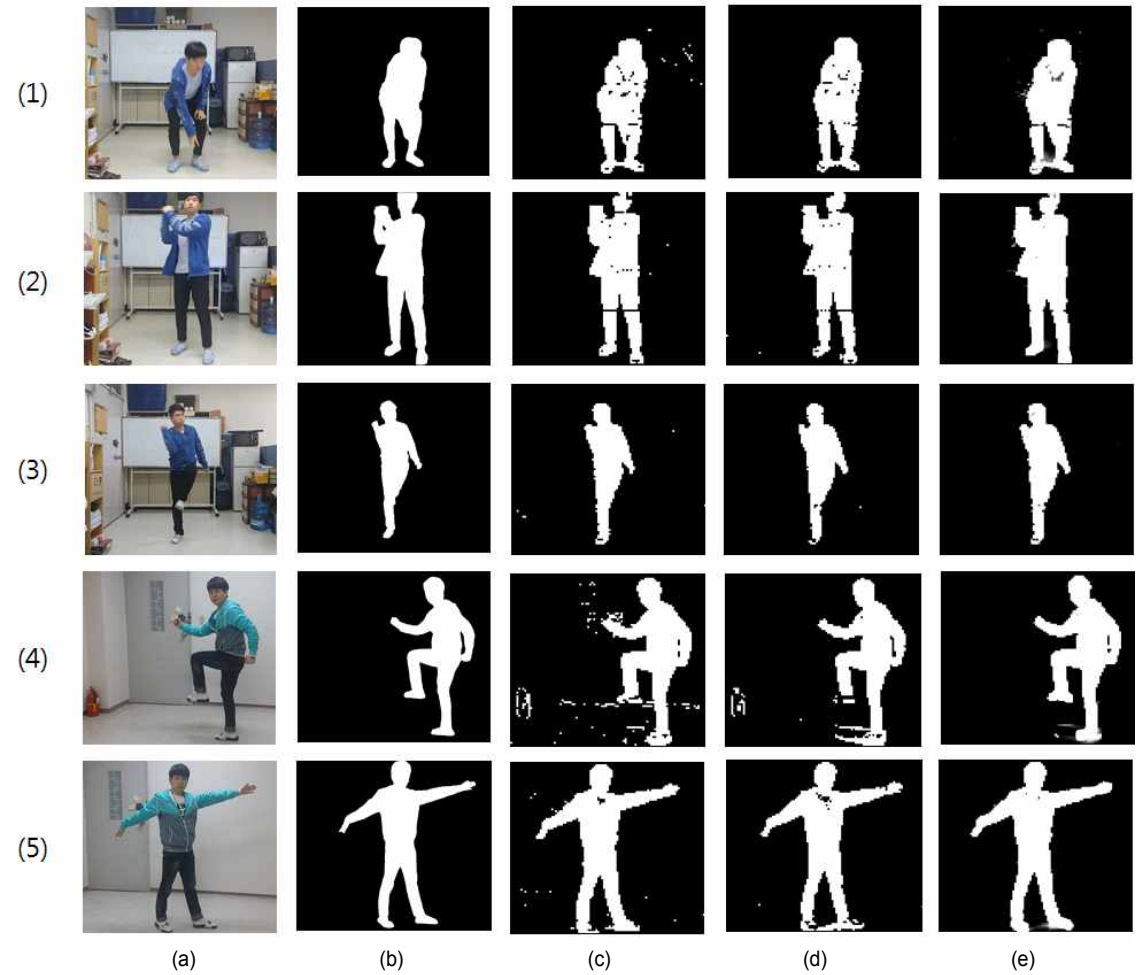


그림 8. 분류 성능 비교 영상 (a)원본 영상 (b)Ground Truth (c)DCT 방법 (d)LBP 방법 (e)제안하는 방법
Fig 8. Comparison of Segmentation Performance (a) Original Image (b) Ground Truth (c) DCT Method (d) LBP Method (e) Proposed Method

표 1. 영상 분류 화소 개수 비교
Table 1. Comparison of Segmentation Pixel Number

Method / Image No.	Ground Truth	DCT	LBP	Propose	|G.T-DCT|	|G.T.-LBP|	|G.T.-Propose|
Image (1)	72883	79424	78400	76038	6541	5517	3155
Image (2)	98742	97472	94528	97634	1270	4214	1108
Image (3)	56986	59264	55360	57413	2278	1626	427
Image (4)	89459	111616	97984	95981	22157	8525	6522
Image (5)	115370	118464	110784	117080	3094	4586	1710

Ground Truth와 이진분류결과의 각각 화소개수를 측정한 것이다. 제안한 방법의 DCT방법, LBP방법과 다르게 확률값을 가지므로 분류결과의 총합함으로써 화소개수를 측정하였다. (영상 크기 1280x720, 총 화소개수 921600)

표와 같이 제안한 방법에서 Ground Truth와 화소 개수의 차이가 상대적으로 작은 것을 관찰할 수 있다. 제안하는 방법의 성능이 더 뛰어난 이유는 그림 8을 통해 오검출/미검출 부분에 대한 보정 때문임을 알 수 있다. DCT방법과 LBP방법 모두 새로 유입된 물체 이외에 대한 오검출 영역과 human내부에 배경으로 분류된 미검출 영역 또한 존재한다.

실험에서 사용한 block의 크기는 8x8이며 경험적으로 설정되었다. block의 크기가 작아지면 global texture 분석의 효과가 감소하며 복잡도가 증가하여 실시간 처리가 불가능하다. 반대로 block의 크기가 커지면 객체 추출 결과에 step 효과가 발생하여 blending 처리 시 자연스러운 결과를 얻기 힘들다. 위와 같은 사항을 고려하여 경험적으로 결정하였다.

앞서 언급하였듯이 새로운 배경에 자연스러운 합성을 위해서는 영상 객체 추출값에 단순 평균 필터를 적용한 값을 Alpha 값으로 지정하여 Blending 처리하는 것이 효율적이다. 그림 9는 현재 프레임을 새로운 배경에 합성한 영상이다.

알고리즘은 OpenCV 기반으로 구현 하였으며 Intel Core i7-4770 cpu 3.4GHz의 컴퓨터 환경에서 HD(1280x720)영상에 대하여 약 28fps의 합성속도를 가지며 이는 일반적인 실시간 프레임속도라 할 수 있다.

V. 결 론

제안하는 알고리즘은 intensity와 색, 그리고 texture의 변화를 측정하고 카메라의 color consistency와 focus 특성을 분석하여 배경으로부터 객체를 실시간으로 추출하는 방법으로 추출된 객체 영역에 human skeleton 특성을 적용하여 최종 객체 추출 결과를 향상시켰다. 또한 intensity, 색 그리

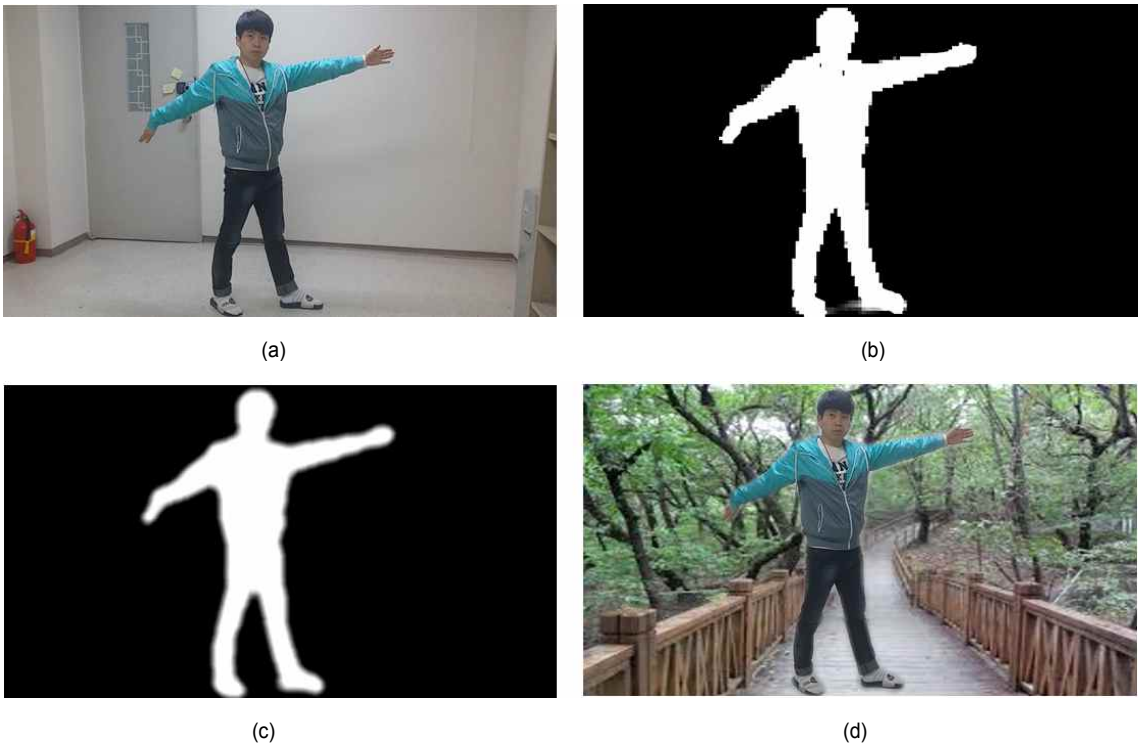


그림 9. 영상 분류의 응용 예시 (a) 원본 영상 (b) 분류 영상 (c) Alpha 영상 (d) 합성 영상
Fig 9. Application Example of Segmentation (a) Original image (b) Segmentation Result (c) Alpha Image (d) Composite Image

고 texture의 분석의 복잡도를 낮추어 알고리즘의 속도를 향상시켜 알고리즘의 범용성을 증가시켰다.

제안하는 알고리즘은 영상 전체에서 Alpha값을 사용하여 객체를 색인화하기 때문에 영상 압축/전송 등 다양한 응용분야에 사용될 수 있다. 또한 human의 skeleton 특징을 이용하여 움직이는 human을 검출하므로 동작인식 및 패턴 분석의 응용분야에서 활용도가 높아질 것이다. 본 논문에서는 움직이는 물체를 안정적이며 실시간으로 추출할 수 있는 새로운 영상 객체 추출 방법을 제시하고 있지만 몇 가지 제한사항이 존재한다. 참조 프레임과 현재 프레임 block의 색 및 texture 특성이 유사한 영역에서 human의 skeleton 특성을 이용하여 결과를 보정하지만 만약 human face를 검출하지 못한다면 실제 객체 일지라도 보정이 어려운 것이 사실이다. 또한 고정된 카메라를 이용하여 초기 프레임 정보를 분석하므로 카메라의 움직임, 조명의 변화에 취약하다. 위의 한계점들은 후속연구에서 다루어 질 것이다.

참 고 문 헌 (References)

[1] T. Yu, C. Zhang, M. Cohen, Y. Rui and Y. Wu, "Monocular Video Foreground/Background Segmentation by Tracking Spatial-Color Gaussian Mixture Models," . In IEEE Workshop on Motion and Video Computing,pages 5 – 5, 2007.

[2] J.J. Verbeek, N. Vlassis and B. Kr ose, "Efficient Greedy Learning of Gaussian Mixture Models," Published in Neural Computation 15(2), pages 469-485, 2003.

[3] Cheol-Jun Jeong, Tae-Ki Ahn, Jong-Hwa Park, and Goo-Man Park, “Real Time Abandoned and Removed Objects Detection System,” JBE, VOL. 16, NO. 3, 462-470, May 2011.

[4] J. Yao and J.M. Odobez, "Multi-Layer Background Subtraction Based on Color and Texture," IN IEEE The CVPR Visual Surveillance Workshop (CVPR-VS), MINNEAPOLIS, June 2007.

[5] X. Sun and F. Chang, "Background Model Combining Gauss Model with Local Binary Pattern Feature," Journal of Convergence Information Technology(JCIT) Volume 7, Number 17, Sep 2012.

[6] M.Heikkila and M.Pietikainen, "A Texture-Based Method for Modeling the Background and Detecting Moving Objects," IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. 28, NO. 4, April 2006.

[7] A. Shimada and R.I Taniguchi, "Hybrid Background Model using Spatial-Temporal LBP," DOI: 10.1109/AVSS.2009.12 Conference: Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2009, 2-4 September 2009.

[8] B. Vishnyakov, V. Gorbatsevich, S. Sidyakin, Y. Vizilter, I. Malin and A. Egorov, "Fast Moving Objects detection Using iLBP Background Model," The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-3, 2014.

[9] L. Jeisung and P. Mignon, "An Adaptive Background Subtraction Method Based on Kernel Density Estimation," Sensors journal, 2012.

[10] L. Li, W. Huang, I. Y.H. Gu, and Q. Tian, "Foreground Object Detection from Videos Containing Complex Background," In Proceedings of the Eleventh ACM International Conference on Multimedia, Berkeley, CA, USA, 2 – 8 November 2003.

[11] R. Rodriguez-Gomez, E. J. Fernandez-Sanchez, J. Diaz and E. Ros, "FPGA Implementation for Real-Time Background Subtraction Based on Horprasert Model," Sensors journal, 2012.

[12] Soon-kak Kwon and Yoohyun Park, "Picture Quality Control Method for Region of Interest by UsingDepth Information", JBE, VOL. 17, NO. 4, 676-683, July 2012.

[13] V. Kolmogorov, A. Criminisi, A. Blake, G. Cross and C. Rother, "Bi-layer segmentation of binocular stereo video," In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20 – 25 pp. 407 – 414. June 2005.

[14] V.Ganapathi, C. Plagemann, D. Koller and S. Thrun, "Real Time Motion Capture Using a Single Time-Of-Flight Camera," In Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13 – 18 pp. 755 – 762. June 2010.

[15] E. J. Fernandez-Sanchez and J.Diaz and E.Ros, "Background Subtraction Based on Color and Depth Using Active Sensors," Sensors journal, 2013.

[16] Saeed Mahmoudpour and Manbae Kim, "Detecting Foreground Objects Under Sudden Illumination Change Using Double Background Models", JBE, VOL. 21, NO. 2, 268-271, March 2016.

[17] R. Lienhart, A. Kuranov, and V. Pisarevsky, “Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection,” Proc. DAGM 25th Pattern Recognition Symp., pp. 297-304, 2003.

[18] R. C. Gonzalez, R. E. Woods and S. L. Eddins, Digital Image Processing Using MATLAB, Gatesmark Publishing, 2009.

[19] T. porter and T.Duff. "Compositing Digital Images," Computer Graphics Volume 18, Number 3 July 1984.

저 자 소 개

김 민 준

- 2014년 8월 : 경희대학교 전자전파공학과 공학사
- 2014년 9월 ~ 현재 : 경희대학교 전자전파공학과 석사과정
- ORCID : http://orcid.org/0000-0002-6743-7958
- 주관심분야: : 사진/동영상 신호처리, 컴퓨터 비전

이 주 철

- 1997년 2월 : 연세대학교 전자공학과 공학사
- 2003년 2월 : 연세대학교 전자공학과 석사
- 2014년 2월 : University of California, San Diego 전자컴퓨터공학 박사
- 1997년 1월 ~ 현재 : KT(Korea Telecom) 연구원
- 주관심분야 : 영상/동영상 신호처리, 컴퓨1터 비전, 동영상 부호화

김 원 하

- 1985년 2월 : 연세대학교 전자공학과 공학사
- 1988년 5월 : University of Wisconsin-Madison 전기공학과 석사
- 1996년 1월 ~ 7월 : (미) Motorola 연구원
- 1997년 5월 : University of Wisconsin-Madison 전기공학과 박사
- 1997년 8월 ~ 2000년 2월 : (미) Los Alamos National Lab. 연구원
- 2000년 3월 ~ 2003년 8월 : 명지대학교 정보통신공학과 조교수
- 2009년 8월 ~ 2010년 8월 : University of California San Diego (UCSD) 교환 교수
- 2003년 9월 ~ 현재 : 경희대학교 전자정보대학 전자전파공학과 교수
- ORCID : http://orcid.org/0000-0002-4797-3526
- 주관심분야 : 영상/동영상 신호처리, 컴퓨터 비전, 동영상 부호화