

논문 2011-48SP-2-2

# 몰입형 화상 회의를 위한 강건한 객체 추출 방법

( A Robust Object Extraction Method for Immersive Video Conferencing )

안 일 구\*, 오 대 영\*, 김 재 광\*, 김 창 익\*\*

( Ilkoo Ahn, Daeyoung Oh, Jaekwang Kim, and Changick Kim )

## 요 약

본 논문에서 우리는 실시간 성능이 요구되는 비디오 화상회의 시스템을 위해 사전정보 없이 정확하게도 완전히 자동으로 비디오 객체를 추출하는 방법을 제안한다. 제안하는 방법은 두 단계로 이루어진다: 1) 초기 프레임에서의 정확한 객체 추출, 2) 객체 추출 결과를 이용한 그 이후 프레임에서의 실시간 객체 추출. 초기 프레임에서의 객체 추출은 초기 프레임들의 차영상으로부터 구한 에지들을 누적시킨 누적 에지맵 생성으로부터 시작된다. 즉, 객체의 초기 움직임의 누적으로부터 객체의 형상을 추측하고자 하는 것이다. 이 추측된 형상은 그래프 컷(Graph-Cut) 영상 분할을 위한 객체 씨드(seeds)와 배경 씨드를 할당하는데 이용된다. 그래프 컷 기반 객체 추출 이후 프레임부터는 객체 추출 결과와 연속된 프레임의 차영상의 에지맵을 이용하여 실시간 객체 추출이 수행된다. 실험결과를 통해 제안하는 방법이 이전 연구들과 달리 VGA 크기의 비디오에 대해서도 실시간으로 동작함을 보이고, 따라서 몰입적인 비디오 화상회의 시스템의 개발을 위한 유용한 도구임을 보이고자 한다.

## Abstract

In this paper, an accurate and fully automatic video object segmentation method is proposed for video conferencing systems in which the real-time performance is required. The proposed method consists of two steps: 1) accurate object extraction on the initial frame, 2) real-time object extraction from the next frame using the result of the first step. Object extraction on the initial frame starts with generating a cumulative edge map obtained from frame differences in the beginning. This is because we can estimate the initial shape of the foreground object from the cumulative motion. This estimated shape is used to assign the seeds for both object and background, which are needed for Graph-Cut segmentation. Once the foreground object is extracted by Graph-Cut segmentation, real-time object extraction is conducted using the extracted object and the double edge map obtained from the difference between two successive frames. Experimental results show that the proposed method is suitable for real-time processing even in VGA resolution videos contrary to previous methods, being a useful tool for immersive video conferencing systems.

**Keywords :** video conference, telepresence, object extraction, image segmentation, graph-cut

## I. 서 론

화상 회의 시스템은 물리적으로 원거리에 위치해 있는 사용자들이 회의를 할 수 있는 환경을 제공한다. 즉, 각자의 위치에서 저비용으로 상대방과 정면으로 마주보고 회의를 할 수 있고 물리적으로 이동할 때 드는 시간

과 비용을 절약해 준다. 특히, 글로벌 기업과 같이 회의 참여자들이 전 세계에 퍼져 있는 경우에는 더욱 유용한 시스템일 뿐 아니라 재택근무자에게도 매우 유용하다.

일반 대중들에게 화상 회의 시스템이 보급되기 시작한 것은 1990년대인데, 그 원인으로는 인터넷 프로토콜의 기술적 발전과 비디오 압축기술의 진보를 들 수 있다. 일반 대중도 화상 회의 시스템을 쉽고 편리하게 이용할 수 있다는 장점으로 인하여 화상회의는 교육, 의료와 건강, 사업, 법률, 대중매체와 같은 다양한 분야에서 널리 이용되어 왔다<sup>[1]</sup>.

2000년대 초반, 전 세계적으로 합리적인 가격의 초고속

\* 학생회원, \*\* 정회원, 한국과학기술원 전기및전자공학과 (Dept. of Electrical Engineering, KAIST)

※ “본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음”(NIPA-2011-(C1090-1011-0003))

접수일자: 2010년7월9일, 수정완료일: 2010년10월4일



그림 1. 화상회의 시스템의 예<sup>[3]</sup>

Fig. 1. An example of video conference system.

인터넷 환경이 갖춰졌다. 동시에, 비디오 촬영 및 디스플레이 기술이 비약적으로 발전하였다. 이러한 요인들은 화상회의를 기존의 단순히 상대방의 얼굴을 보며 음성통신을 통해 의사소통하는 수준에서 벗어나, 회의자가 물리적으로 다른 공간에 존재하는 것이 아닌 동일한 공간에 함께 존재하는 듯한 느낌을 주고 회의에 몰입할 수 있는 새로운 형태의 화상회의 시스템을 가능하게 하였다. 이를 텔레프레즌스(Telepresence)라 부른다. Steuer는 텔레프레즌스를 ‘주위 환경(environment)에 대한 중간적 인지(mediated perception)’이라고 정의하였고<sup>[2]</sup>, 위키피디아는 텔레프레즌스를 ‘한 개인으로 하여금 마치 그들이 (같은 곳에) 존재하는 것처럼 느끼게 하고, (같은 곳에) 존재하는 것 같은 외관(appearance)을 주거나 그들이 실제로 존재하는 곳이 아닌 어느 한 곳에 존재하는 것 같은 효과를 주는 기술들의 집합’이라고 정의하고 있다<sup>[3]</sup>. 텔레프레즌스 시스템은 온라인 상에서 고차원의 몰입을 제공하고 사용자들이 그곳에 존재하는 효과를 주어 의심 없이 완전히 회의에 몰입할 수 있는 환경을 제공한다. 텔레프레즌스 시스템이 제공하는 몰입의 요소들로는 실물 크기의 영상, 시선 교차, 그리고 공간적 실감형 사운드 등이 있다.

완전한 몰입형 화상회의 영상을 위해서는 최소한 두 가지 사항을 만족해야 한다. 첫째, 화상회의 영상은 실물 크기의 영상이어야 한다. 구체적으로 화상회의 영상은 1920×1080 고화질(HD) 또는 최소한 640×480 VGA 급의 해상도를 유지해야 한다. 두번째, 화상회의 솔루션은 상대방 참여자들이 마치 동일한 공간의 테이블 건너편에 있는 듯한 느낌을 주어야 한다. 동일한 공간에 존재한다는 느낌을 주기 위한 가장 중요한 요소 중 하나는 화상회의 참여자들의 배경을 모두 유사한 배경으로

보여주는 것이다. 회의자들의 배경이 각각 다를 경우, 참여자들은 상대방과 같은 곳에 존재하는 것이 아니라 다른 곳에 존재한다고 느낄 것이다. 또한, 상대방의 배경에 존재하는 물체는 회의에 집중하는 데 방해 요소가 될 수 있다. 객체가 정확히 추출되고 배경이 미리 잘 준비된 배경으로 대체된다면, 회의 상대방과 동일한 공간에 존재한다는 느낌을 주고 회의에 몰입할 수 있는 환경을 제공할 것이다. 이를 위해 강건하면서도 실시간으로 객체를 추출할 필요가 있다.

화상회의 영상과 같은 머리-어깨 영상에서의 객체 추출 연구는 그동안 여러 방법들이 연구 되어 왔다. Luo 등<sup>[4]</sup>은 QCIF크기의 머리-어깨 영상에 대해 통계적 모델 기반의 객체 추출을 수행하였다. 머리-어깨 영상이 머리와 어깨, 배경 영역으로 이루어지고 배경영역은 객체를 포함하지 않는다고 가정하고 배경영역에 대한 모델링을 수행하여 각 영역의 컬러와 공간적 분포를 이용해 객체를 분할하였다. 이 논문에서는 객체 영역을 추출하기 위해 영역 성장법(region growing)을 사용하는데, 객체가 배경과 같은 화소값을 가져 영역이 새는 경우(leaking) 이 방법은 부정확한 추출 결과를 얻게 되며, 배경 영역이 복잡할 경우 배경의 확률 모델링이 쉽지 않게 된다. Gaobo 등<sup>[5]</sup>은 배경영상이 사진에 취득가능하다고 가정하고 컬러 에지 기반의 변화 검출 방법을 이용한 셀방식의 신경망 (cellular neural networks)을 구현하여 머리-어깨 영상에서의 객체를 추출하였다. 하지만, 배경 영상을 화상회의 때마다 미리 취득해야 한다는 단점이 있다. Kolmogorov 등<sup>[6]</sup>과 Criminisi 등<sup>[7]</sup>은 스트레오기반의 비디오 객체 추출을 제안하였다. 이 기법은 스테레오 기반의 깊이맵 정보를 이용하여 강건한 객체 추출 결과를 보이고 있다. 하지만, 화상회의 시스템에 스테레오 카메라를 설치하는 것은 추가 비용이 들게 되고, 민무늬 배경이나 동일한 텍스처의 전경영역에서는 스테레오 깊이맵 추출 성능이 떨어지는 단점이 있다. Wang 등<sup>[8]</sup>은 HOG(Histogram of Oriented Gradients)를 이용해 그래프 컷 영상분할을 수행한다. 그래프 컷 영상분할에 요구되는 많은 연산량과 더불어 HOG 내부의 기울기(gradient) 연산자 사용시 연산량이 커져 실시간 성능을 요구하는 화상회의 시스템에는 부적절하다.

본 논문에서는 사전의 배경정보 없이 현대의 카메라로 몰입형 화상회의 경험을 제공하기 위해 CIF 크기 이하의 비디오를 주 타겟으로 하였던 과거의 방법들과 달리 VGA 급의 비디오에 대해 강건하면서도 실시간으로

동작하는 객체 추출 방법을 제안하고자 한다. 제안하는 알고리즘은 연속된 두 프레임의 차영상에서의 에지 맵(edge map)을 이용한 Kim 등<sup>[9]</sup>의 방법을 기초로 하고 있다. 현재 프레임의 에지 중에서 차영상의 에지 포인트들과 공간적으로 일정 거리 내에 존재하는 에지 포인트들의 집합과 현재 프레임의 에지 중 이전 프레임의 움직임 에지와 공간적으로 일정 거리 내에 존재하는 에지 포인트들의 집합인 움직임 에지(moving edge)를 이용하여 객체를 추출한다. Kim<sup>[9]</sup>의 방법은 일단 첫 프레임에서의 움직임 에지 맵이 구해지면, 그 다음 프레임부터는 빠르면서도 정확하게 객체를 추출한다.

제안된 알고리즘에서는 영상에서의 객체를 추출하기 위한 첫 단계로 초기 수십 프레임동안 차영상의 에지맵을 누적하여 누적 에지맵을 생성한다. 일반적으로 화상회의 영상에서의 회의 참석자들은 수십 프레임의 짧은 시간이라 할지라도 미세한 움직임을 보이기 때문에, 차영상의 에지맵을 수십 프레임동안 누적한 누적 에지맵은 객체의 형상정보를 얻기 위한 단서를 제공한다. 누적 에지맵으로부터 얻어진 초기 머리-어깨 형상은 그래프 컷 영상 분할<sup>[10]</sup>을 위한 객체 씨드와 배경 씨드를 주는 데 사용된다. 그래프 컷 영상 분할을 이용하여 픽셀 단위의 객체 추출을 수행한다. 실험 결과를 통해 제안된 방법이 정확한 분할 결과를 제공할 뿐 아니라 실시간 처리에도 적합한 방법임을 보이고자 한다.

본 논문은 다음과 같이 구성되어 있다. II장에서는 초기 수십 프레임 동안의 차영상을 이용한 객체 추출 과정을 설명하고, III장에서는 II장에서의 결과를 이용하여 이어지는 프레임에서 객체 추출을 실시간으로 추출하는 방법을 설명한다. IV장에서는 실험결과를 제시하며, 이에 대한 결론을 V장에 기술한다.

## II. 차영상을 이용한 자동 객체 추출 방법

이 장에서는 영상의 초기 수십 프레임의 차영상을 이용해 자동으로 객체를 추출하는 방법을 알아본다. 영상의 정확한 객체를 추출하기 위해 II장 1절에서 객체의 대략적인 형상을 파악한 후 II장 2절에서 그래프 컷을 이용한 정확한 객체 추출을 수행한다.

화상회의 영상과 같은 머리-어깨 영상은 머리가 위에 존재하고 몸통이 그 아래 존재하는 오메가( $\Omega$ ) 형상을 띄고 있다. 대부분의 머리-어깨 영상에서 머리와 어깨 형상은 비슷한 형태를 갖지만 팔 부분은 사람마다



(a)



(b)

그림 2. (a) (b) 각각 식 (1)과 식 (2)에 의한 움직이는 에지 검출 결과  
 Fig. 2. (a) (b) The moving edge maps of eq. (1) and eq. (2) respectively.

다른 자세를 취하고 있어 그 형상을 예측하기가 쉽지 않다. 팔을 포함한 전체적인 형상을 정확하게 추출하기 위한 방법을 II장 1절에서 알아본다.

그래프 컷(Graph-Cut) 영상분할 알고리즘은 픽셀 단위의 정확한 객체 추출을 수행하는데, 객체 씨드(seed)와 배경 씨드(seed)를 사전 정보로 입력해 주어야 한다. 이 씨드들은 II장 1절에서 구한 머리-어깨의 형상을 이용해 구할 수 있다. 이 과정은 II장 2절에 기술하도록 한다.

### 1. 머리-어깨 영상에서의 초기 객체 형상 추출

앞서 말한 바와 같이, 화상회의 참석자들은 미세한 움직임을 보여 이웃한 프레임에서조차 차영상 에지가 존재한다. 이러한 차영상의 에지는 객체의 형상을 추출하는 단서를 제공한다. 머리-어깨 영상에서 초기 수십 프레임에서의 객체 형상 추출을 위해 차영상의 에지를 이용한다. 영상의 에지를 이용한 객체 추출은 영상처리에서 기본적으로 사용된 방법이며 특히 Kim<sup>[9]</sup>의 방법은 에지를 이용해 빠르고 우수한 객체 추출 성능을 보



그림 3. (a) 첫번째 프레임 영상 (b)  $CDE_{30}$   
Fig. 3. (a) The first frame. (b)  $CDE_{30}$ .

이고 있다. 그러나 초기화면부터 객체가 존재하는 경우, 사용자에게 의한 배경에지의 추출과정이 필요한 단점이 있다. 우리는 이 절에서 Kim<sup>[9]</sup>의 알고리즘을 바탕으로 하되, 사용자에게 의한 간섭이 필요 없는 완전 자동화된 알고리즘에 대해 자세히 설명하고자 한다.

Kim<sup>[9]</sup>의 자동 비디오 객체 추출 알고리즘은 Marr<sup>[11]</sup> 등에 의해 알려진 것과 같이 인간의 시각 인지 과정의 첫 단계이자 가장 중요한 단계인 에지 검출로 시작한다. 에지 정보는 실제 장면에서 표면의 물리적 변화를 추출하는데 중요한 역할을 하지만, 연속된 비디오 영상에서의 움직이는 객체의 형상(shape) 정보를 추출하기 위해 단순히 에지들의 차이를 구하는 것은 정적인 배경을 가진 영상이라 할지라도 그림 2-(a)에서와 같이 많은 잡음을 생성한다. 이는 한 영상에서의 랜덤(random) 잡음은 다른 영상에서의 랜덤 잡음과 다르기 때문이고 결과적으로 연속된 영상이라 할지라도 에지의 작은 변화를 만들어 낸다. 에지의 차영상은 다음과 같이 정의된다.

$$\begin{aligned} & |\Phi(I_{n-1}) - \Phi(I_n)| \\ & = |\theta(\nabla G * I_{n-1}) - \theta(\nabla G * I_n)| \end{aligned} \quad (1)$$

여기서  $\Phi(I)$ 는 가우시안 콘볼루션 된 영상  $G * I$ 에 기울기 연산  $\nabla$ 을 수행 후 가느다란 에지를 만들기 위해 기울기 크기(gradient magnitude)에 비극대점 역제를 수행한 후 에지 검출과 에지 연결을 위해 이중 문턱치(double threshold) 처리를 수행하는 Canny 에지 검출기를 이용해 구한다. 반면, 연속된 프레임의 차영상의 에지를 추출하면 잡음에 강건한 차영상 에지맵  $DE_n$ 을 얻게 된다. 그 이유는 Canny 연산 속에 포함되어 있는 가우시안 콘볼루션 연산이 차영상의 잡음을 억제하기 때문이다.

$$DE_n = \Phi(|I_{n-1} - I_n|) = \theta(\nabla G * |I_{n-1} - I_n|) \quad (2)$$

화상회의 참여자들은 회의 시 많은 움직임을 보이지만 않더라도 미세한 움직임을 보인다. 이러한 미세한 움직임은 몸 전체에서 나타나므로 이 미세한 움직임을 몸의 형상을 알아낼 수 있는 단서가 된다. 몸의 형상을 알아내기 위해 차영상 에지맵  $DE_n$ 을 누적한 누적 차영상 에지맵  $CDE_n$ 을 다음과 같이 정의한다.

$$CDE_n = CDE_{n-1} + DE_n, n \geq 2 \quad (3)$$

여기서  $CDE_1$ 은 아무런 에지도 포함하지 않는다.

그림 3-(b)는 누적 차영상 에지맵  $CDE_{30}$ 을 보인 것으로 객체의 형상이 나타남을 알 수 있다. 본 논문에서는  $CDE_{30}$ 을 이용하여 객체의 형상을 추출하기로 한다(이 후  $CDE$ 는  $CDE_{30}$ 를 나타냄). 객체 외x부의 노이즈는 에지맵의 침식 결과를 마커(marker)로 하는 복구에 의한 열기(Opening By Reconstruction)을 통해 제거된다.

누적 차영상 에지맵  $CDE$ 은 머리-어깨 영상의 대략적인 외부 형상 정보를 제공하지만 내부 영역 정보는 포함하지 않아  $CDE$ 의 내부를 채울 필요가 있다. 실제 머리-어깨 형상과 유사한 형상을 얻기 위해  $CDE$ 의 내부 영역을 채울 때, 턱 위 영역과 턱 아래 영역으로 나뉘어 내부 영역을 채울 필요가 있다. 그 이유는 일반적으로 화상회의 참여자는 머리 영역에서는 많은 움직임을 보여 머리 영역에서는 비교적 많은 누적 에지가 분포하지만 몸통영역에서의 에지는 상대적으로 적기 때문이다.

머리-어깨 영상의 턱 위 영역과 턱 아래 영역을 나누기 위해 얼굴 검출을 수행한다. 화상회의 영상은 상대방의 얼굴을 보면서 회의를 하는 것이 목적이므로 영상 내에 반드시 얼굴이 존재한다고 가정할 수 있다. 객체의 대략적인 위치와 크기를 가능하기 위해 본 알고리즘



그림 4. Viola-Jones 얼굴 검출기<sup>[12]</sup>로 검출한 얼굴 영역의 예

Fig. 4. The detected face regions using Viola-Jones face detector<sup>[12]</sup>.

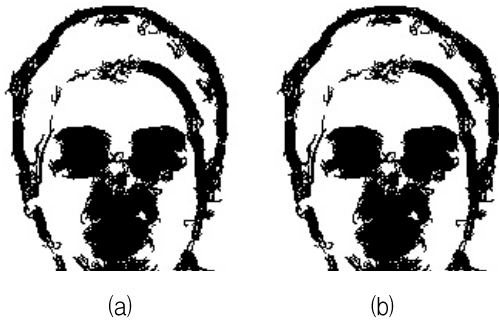


그림 5. 에지 화소가 머리영역 경계와 맞닿을 경우 경계 화소를 추가한다 (편의상 영상의 밝기에 부정(negative) 연산을 취하여 표현함). (a) 그림 3-(b)의 머리 영역. (b) (a)의 에지가 가장자리와 맞닿는 경계(아래 경계) 화소를 채움

Fig. 5. Adding object boundary when edge pixels touch image boundary points (see bottom line on (b)). (the image is represented by negative image) (a) The head region of fig. 3-(b). (b) Adding object boundary when the edge of (a) touch image boundary.

은 강건하면서도 빠른 알고리즘으로 알려진Viola-Jones<sup>[12]</sup>가 제안한 아다부스트(Adaboost)를 이용한 얼굴 검출 알고리즘을 이용하였다.

그림 4는 Viola-Jones 얼굴 검출기로 검출된 얼굴 창(face window)을 나타낸 그림이다. 이 얼굴 창의 가장 아래 행을 기준으로 머리영역과 몸통영역으로 나누기로 한다. 그림 5-(a)는 그림 3-(b)의 머리영역을 나타낸 것이다. 머리영역의 내부를 채우기 전, 머리영역의 하단 경계를 그림 5-(b)와 같이 연결해 준다.

객체의 수평적 그리고 수직적 후보군은 영상 각 행 그리고 열의 첫 번째 포인트와 마지막 포인트의 내부 영역으로 정의한다. 수평과 수직 방향의 객체 후보군을 찾은 후, 두 후보 영역을 논리적 AND 연산하여 교차 영역을 계산하고 모폴로지 연산을 적용하여 내부가 채워진 객체를 얻을 수 있다. 이를 그림 6에 나타내었다.

Kim<sup>[9]</sup>은 수평적·수직적 객체 후보군을 추출한 뒤 경계처리 후 다시 수평적·수직적 객체 후보군을 추출하지만 본 논문에서는 경계 처리를 먼저 수행한 후 한번만 수평적·수직적 객체 후보군을 추출하도록 한다.

CDE의 머리영역을 채운 후, 몸통영역을 채운다. CDE의 몸통영역의 내부를 채우는 데 있어서는 수직적 후보를 먼저 구한 뒤, 그 수직적 후보군에 대해 수평적 후보군을 구한다. 그 이유는 어깨 아래 영역의 적은 움직임으로 인해 누적 에지가 존재하지 않는 경우 수직

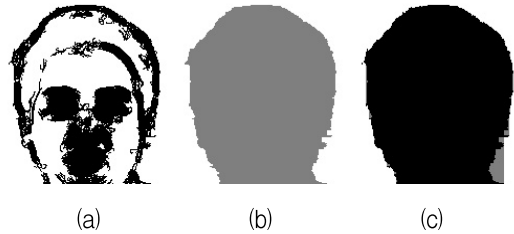


그림 6. 내부 영역 채우기 및 객체 추출 과정. 편의상 영상의 밝기에 부정(negative) 연산을 취하여 표현함. (a) CDE의 머리 영역 (b) 수평적 후보군. (c) 수평적 후보군과 수직적 후보군의 AND 연산 결과(검정색 영역)

Fig. 6. Filling inner region and object extraction process. (the image is represented by negative image). (a) The head region of CDE. (b) horizontal candidates. (c) logical AND(black area) of horizontal and vertical candidates.

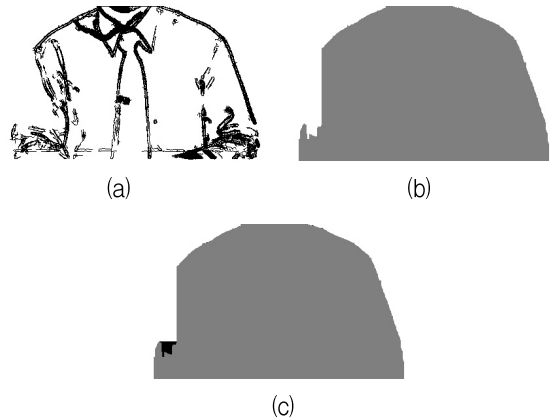


그림 7. CDE의 내부 영역 채우기 및 객체 추출 과정. 편의상 영상의 밝기에 부정(negative) 연산을 취하여 표현함. (a) 의 몸통 영역 (b) 수직적 후보군. (c) 수직적 후보군(회색 영역)에 대한 수평적 후보군(회색 영역과 검정색 영역)

Fig. 7. Filling inner region and object extraction process of CDE. The image is represented by negative image. (a) The body region CDE of . (b) vertical candidates. (c) horizontal candidates(gray and black area) of vertical candidates(gray area).

적 후보군과 수평적 후보군을 동시에 만족하는 영역이 객체의 형태를 제대로 근사시키지 못하기 때문이다. 몸통 영역의 수직적 후보군은 몸통 영역의 각 열의 첫 포인트부터 가장 아래 행까지의 내부로 정의한다. 수평적 후보군은 CDE의 머리영역의 수평적 후보군과 같은 방식으로 구한다. 이 과정을 그림 7에 나타내었다.

객체와 배경의 밝기 값이 비슷하거나 회의자의 움직임이 없어 그림 7-(c)에서와 같이, 불완전한 누적 에지 맵 (즉, 회의 참여자의 오른쪽 팔 부분의 에지맵이 없

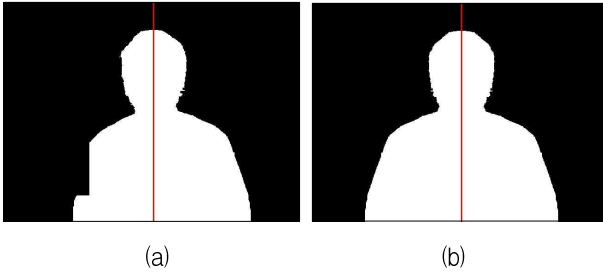


그림 8. (a)  $CDE$ 의 내부가 채워진 객체 형상 (b) (a)를 얼굴의 중심축을 기준으로 대칭시킨 형상  
 Fig. 8. (a) Filled inner regions of  $CDE$ . (b) The horizontally flipped regions against the vertical axis passing the center of face region (vertical red line).

다)이 생성될 경우 이를 보완하기 위해 검출된 얼굴 창의 중심을 세로로 관통하는 수직선을 중심으로 대칭을 취하여 (horizontal flipping), 그림 8-(b)와 같이 새로이 화소를 포함하는 대칭형상을 얻도록 한다. 이와 같이 구해진 최종 객체 형상은 다음 절에서 설명되는 그래프 컷 기반의 영상 분할의 씨드 할당에 이용된다.

2. 그래프 컷 기반 객체 추출

그래프 컷 영상 분할은 영역 특성 항(region properties term)과 경계 특성 항(boundary property term)으로 구성된 에너지 함수를 전역적으로 최소화하는 방법으로, 영역 특성 항은 주어진 사전정보와 각 픽셀간의 유사도를 나타내며, 경계 특성 항은 영상공간에서 이웃하는 픽셀간의 유사도를 나타낸다<sup>[10]</sup>.

$P$ 와  $N$ 을 각각 영상의 픽셀의 집합과  $P$ 내의 이웃하는 픽셀들의 순서가 없는 모든 쌍  $\{p, q\}$ 의 집합이라고 하자. 또  $A = (A_1, \dots, A_p, \dots, A_{|P|})$ 를 그 요소가 객체 "obj" 또는 배경 "bkg" 중 어디에 속하는지를 나타내는 이진 벡터라고 하자. 전체적인 최적의 분할을 위해 경계 특성 항과 영역 특성 항에 의한 분할 비용 함수는 다음과 같이 표현된다.

$$E(A) = \lambda \cdot R(A) + B(A) \tag{4}$$

여기서  $\lambda$ 는 영역 특성 항과 경계 특성 항의 상대적 중요성을 결정하는 상수이다.

그래프 컷 영상분할에서 최적의 영상분할을 위해 가장 중요한 것은 초기 객체 씨드와 배경의 씨드를 어떻게 주느냐 하는 점이다. 초기 객체 씨드와 배경 씨드를 다음과 같이 나타낼 수 있다.

$$\forall p \in O, A_p = "obj" \tag{5}$$

$$\forall p \in B, A_p = "bkg" \tag{6}$$

여기서  $O$ 와  $B$ 는 각각 객체와 배경의 씨드로 표시된 픽셀들의 집합을 말한다.

영상을 분할하기 위해서 이렇게 사용자가 직접 객체와 배경에 해당하는 픽셀을 지정해주는 경성 제약(hard constraints)은 분할 비용 함수 식 (4)와 결합된다. 경성 제약은 영상의 관심 객체에 대한 고급 문맥 정보를 제공한다. 다시 말해, 분할 비용 함수 식 (4)의 전체적 최소값은 경성 제약을 만족하는  $A$  중에 존재한다. 따라서 분할 결과는 사용자가 지정한 경성 제약에 상당히 민감하다. 그래서 가능한 한 객체의 형상에 근사하게 씨드를 주는 것이 보다 정확한 추출 결과를 얻을 수 있다.

그림 9는 2.1절에서 구한 객체의 형상정보를 이용하여 객체와 배경 영역의 씨드를 실제 객체 영역과 근사하게 주는 과정을 보여준다. 그림 9-(a)는 2.1절에서 구한 객체 형상을 나타낸 것이고 그림 9-(b)는 이 형상 영역에 모폴로지 침식과 팽창 연산을 수행하여 각각 객체 씨드 영역을 파란색으로, 배경 씨드 영역을 빨간색으로 표시한 것이다. 그림 9-(c)는 그림 9-(b)를 씨드로 입력하여 그래프 컷 객체 추출 결과를 보인 것이다.

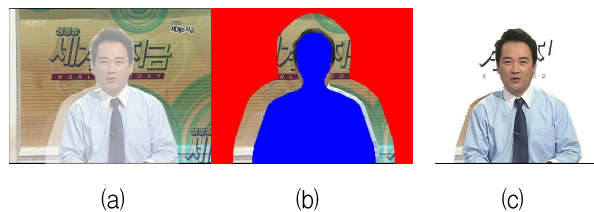


그림 9. 2.1절에서의 결과로 객체 영역과 근사하게 경성 제약을 주는 과정. (a) 2.1절에서 구한 객체 형상이 원본 영상에 겹쳐져 있다. (b) (a)의 형상 영역에 모폴로지 침식 및 팽창 연산을 수행하여 구한 객체와 배경 씨드 (c) 그래프 컷 객체 분할 결과

Fig. 9. Assigning hard constraints closely to object region using the results of section 2.1 (a) the extracted object shape from section 2.1 (semitransparent white area) is overlaid on original image. (b) object and background seeds obtained from taking morphological erosion and dilation to the object shape. (c) extracted objects.

### III. 초기영상에서의 추출 결과를 이용한 실시간 객체 추출

#### 1. 움직임 에지 맵 추출

움직임 에지 맵  $ME_n$ 은 2.1절에서 설명한 차영상의 에지맵  $DE_n = \Phi(|I_{n-1} - I_n|)$ , 현재 프레임에 대한 에지 맵  $E_n = \Phi(I_n)$ , 그리고 배경 에지 맵  $E_b$ 를 조합하여 생성된다.

현재 프레임에서의 Canny 에지 검출기로 구해진 에지 포인트들의 집합을 에지 모델  $E_n = \{e_1, \dots, e_k\}$ 이라 정의하자. 비슷한 방식으로 움직임 에지 포인트들의 집합을  $ME_n = \{m_1, \dots, m_j\}, j \leq k$ ,이고  $ME_n \subseteq E_n$ ,으로 정의한다.  $ME_n$ 내의 에지 포인트들은 움직임 객체의 경계에만 국한되지 않고 객체의 경계 내의 내부에도 존재할 수 있다.  $DE_n$ 이 차영상의 에지 맵에 속한 모든 픽셀들의 집합이라면, 에지 변화에 의한 움직임 에지 맵  $ME_n^{change}$ 은  $E_n$ 의 포인트들의 집합 중에서 각 포인트들로부터  $T_{change}$  거리 내에  $DE_n$ 의 포인트가 존재하는 포인트들의 집합으로 구해진다. 즉, 다음 수식으로 표현된다.

$$ME_n^{change} = \{e \in E_n | \min_{x \in DE_n} \|e - x\| \leq T_{change}\} \quad (7)$$

$ME_n$ 는 다음 단계로 처리되기 전 제거되어야 할 분산된 잡음을 포함할 수도 있다. 또한 시간적으로 정적인 움직임 에지  $ME_n^{still}$ 를 검출하기 위해 이전 프레임에서의 움직임 에지를 참고할 필요가 있다. 즉  $ME_n^{still}$ 은 다음 수식으로 표현된다.

$$ME_n^{still} = \{e \in E_n | e \notin E_b, \min_{x \in ME_{n-1}} \|e - x\| \leq T_{still}\} \quad (8)$$

배경 에지 맵  $E_b$ 는 2장에서 그래프 컷 분할에 의해 구해진 객체와 배경 영역을 이용하여 다음과 같이 구할 수 있다.

$$E_b = \{e \in E_{30} | e \in "bkg"\} \quad (9)$$

즉,  $E_b$ 는 30번째 프레임에서의 에지 맵  $E_{30}$  중 II장 2절에서 추출한 배경 "bkg"에 속하는 에지 포인트들의 집합이다.

두 움직임 에지 맵을 결합한 현재 프레임  $I_n$ 에서의 최종 움직임 에지맵은 다음과 같다.

$$ME_n = ME_n^{change} \cup ME_n^{still} \quad (10)$$

초기 움직임 에지 맵  $ME_{30}$ 는  $ME_{30} = E_{30} - E_b$ 으로 정해지는데, 다시 말하지만  $E_{30}$ 은 30번째 프레임의 에지 맵이고  $E_b$ 는 배경영역의 에지맵으로 II장 2절의 그래프 컷 영상분할에서 구한 배경영역을 이용해 구할 수 있다.

#### 2. 객체 추출

본 절에서의 객체 추출 과정은 2.1절에서 설명한 CDE의 머리영역의 객체 추출과 같다. 이를 그림 10과 그림 11에 나타내었다.

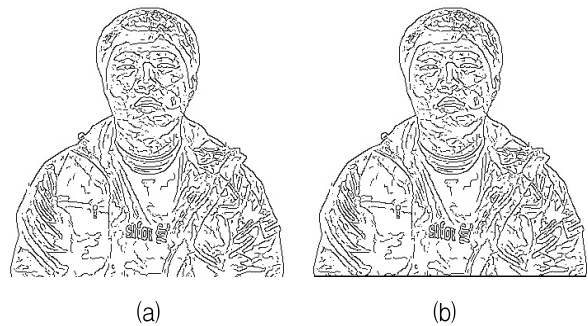


그림 10. 움직임 에지가 경계 픽셀에 존재할 경우 객체의 경계 추가하기 (a) 움직임 에지 맵 (b) (a)에 객체 경계가 추가된 에지맵. 아래 경계에 에지가 추가된 것을 알 수 있다

Fig. 10. Adding object boundary when edge pixels touch image boundary points (see bottom line on (b)) (a) Moving edge map (b) Adding object boundary when the edge of (a) touch image boundary.

#### 3. 첨도(Kurtosis)를 이용한 객체의 재추출 (Re-extract)

움직임 에지 맵을 이용하여 객체 추출 시, 이전 프레임의 에지를 참조하여 현재 프레임의 에지를 검출하기 때문에 움직임 에지가 잘못 선언되는 경우, 시간이 지남에 따라 에러가 누적(propagation)될 확률이 존재한다. 따라서 '적절한 시기'마다 초기화면에 대해 적용했던, 그래프 컷을 이용한 전경/배경의 분류를 실시하여 추출되는 객체 형상을 늘 정확하게 유지해 줄 필요가 있다. 하지만, 그래프 컷의 이용에는 수십 프레임의 지연이 수반되므로, 우리는 위에서 말한 '적절한 시기'를

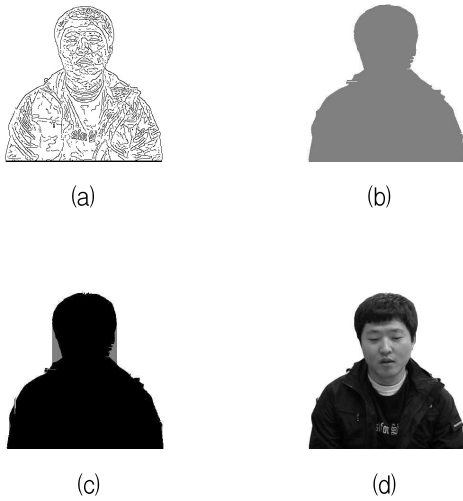


그림 11. 객체 추출 과정. 편의상 영상의 밝기에 부정 (negative) 연산을 취하여 표현함. (a) 움직임 에지 맵 (b) 수평적 후보군. (c) 수평적 후보군과 수직적 후보군(회색 영역과 검정색 영역)의 AND 연산 결과(검정색 영역) (d) 모폴로지 처리 후 추출된 객체

Fig. 11. The object extraction process (a) The moving edge map (b) horizontal candidates (c) logical AND(black area) of horizontal and vertical candidates (d) extracted object after morphological processing.

객체의 움직임이 매우 적은 시기로 규정한다. 재추출시 걸리는 시간에 재추출하기 직전의 영상을 계속 보여 주어 불편함을 줄인다.

이 절에서 우리는 차 영상의 첨도를 이용하여 회의자의 움직임량을 추정한다<sup>[13]</sup>. 첨도는 확률 분포의 뾰족함 (Peakedness) 정도를 측정하는 지표로서 분포의 뾰족함 정도를 통해 분산을 예측하고, 분포 형태를 파악할 수 있다. 첨도는 아래와 같이 정의된다.

$$k = \frac{E[(x - \mu)^4]}{\sigma^4} \quad (11)$$

첨도값이 커질수록 분포는 큰 분산값을 가지는 경향이 있다. 또한, 첨도값이 0에 가까울수록 라플라스 분포 (Laplace distribution)에 가까우며, 3이면 표준 정규분포(Normal distribution)를 따르는 경향이 있다. 화상회의 영상은 배경이 정적이고 회의자만 움직이므로, 차영상의 밝기 값은 대부분 작은 값을 갖게 되며 회의자가 움직일 경우에만 차영상의 분산값이 커지게 된다. 그러므로 회의자의 움직임이 많아 차영상의 분산값이 크면

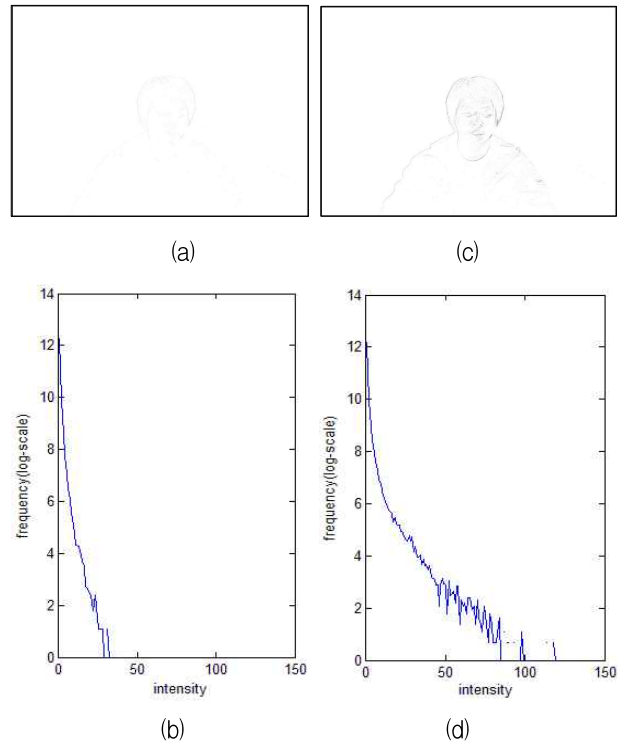


그림 12. 회의자의 움직임량에 따른 첨도값 (a) 움직임이 적은 경우의 차 영상(편의를 위해 밝기의 부정 (Negative) 연산을 취하여 표현) (b) (a)의 확률 분포(첨도값 18.02) (c) 움직임이 많은 경우의 차 영상 (d) (c)의 확률 분포(첨도값 181.84)

Fig. 12. Kurtosis value based on motion of meeting participant (a) difference image in the case of low motion variation(using negative operation of intensity) (b) distribution of (a)(Kurtosis value : 18.02) (c) difference image in the case of high motion variation (d) distribution of (c)(Kurtosis value : 181.84).

첨도값 또한 크다고 볼 수 있다.

그림 12는 회의자의 움직임량에 따른 차 영상과 분포를 보여주고 있다. 그림 12-(a)의 경우 움직임이 거의 존재하지 않기 때문에 차영상에서는 각 픽셀별 변화가 거의 일어나지 않는다. 그림 12-(b)는 그에 대한 분포로서, 밝기(intensity)에 따른 확률 분포를 나타낸다. 편의를 위해 도수들은 log-scale로 표현하였다. 첨도값이 18.02이고 퍼져있는 정도가 작다. 대부분의 픽셀들이 밝기가 작은 부분에 밀집되어 있음을 알 수 있다. 그림 12-(c)는 움직임이 많은 경우이다. 차 영상에서 회의자의 전체적인 윤곽이 나타나는 것을 확인할 수 있다. 그림 12-(d)는 (c)의 밝기에 대한 확률 분포이다. 분포가 상대적으로 더 퍼져있는 것을 알 수 있으며 첨도값이 181.84로서 첨도값 또한 높은 것을 알 수 있다. 이 같은



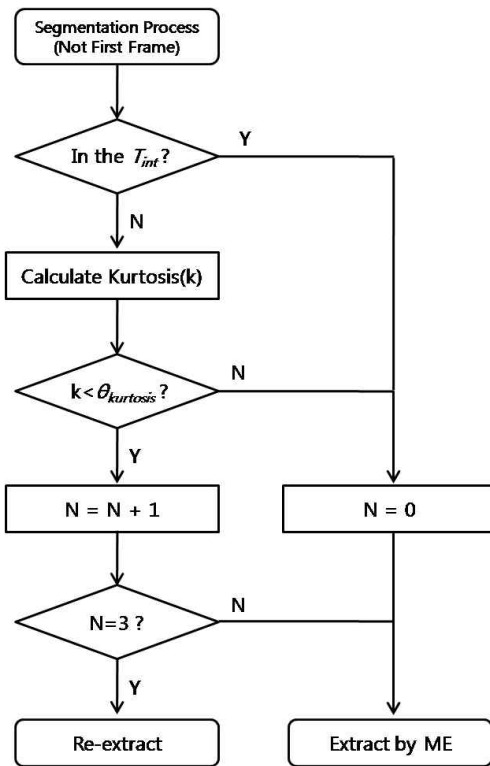


그림 13. 회의자의 움직임 판단 & 재추출 프로세스  
 Fig. 13. Motion analysis of meeting participant & process of re-extraction.

관찰 결과를 기초로 우리는 침도값을 재추출하는 시점에 대한 척도로 사용하였다.

그림 13에 제안하는 움직임 판단 프로세스가 명시되어 있다. 움직임이 적은 경우는 침도값이 임계치인  $\theta_{Kurtosis}$  보다 작은 경우로 정의한다. 연속되는 세 프레임의 침도값이 임계치보다 작은 경우 재추출하는 시점으로 정의한다.

$$\begin{aligned} & \text{If } (k_{t-2} < \theta_{Kurtosis}) \text{ and } (k_{t-1} < \theta_{Kurtosis}) \\ & \text{and } (k_t < \theta_{Kurtosis}), \\ & \text{then RE-EXTRACT} \end{aligned} \quad (12)$$

움직임이 적은 프레임이 지속될 경우, 연속되는 침도값만을 매 프레임 판단하여 재추출을 지속적으로 하게 되면 회의자는 프로그램의 지나친 업데이트로 인한 불편함을 느끼게 된다. 그러므로 재추출이 이루어진 후, 일정기간 동안은 침도값을 판단하지 않는 기간  $T_{int}$ 가 필요하게 된다.  $T_{int}$ 가 지나치게 길 경우에 에러가 누적되는 것을 막지 못하게 되므로  $T_{int}$ 는 다음과 같은 제약조건을 가지게 된다.

$$T_{user} < T_{int} < T_{error} \quad (13)$$

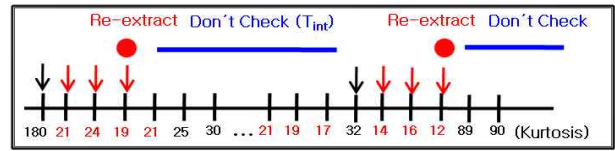


그림 14. 회의자의 움직임 판단 후 재추출 예  
 Fig. 14. Example of re-extraction after motion analysis of meeting participant.

$T_{user}$ 는 회의자가 불편함을 느끼기 시작하는 시간으로  $T_{user}$ 보다 작을 경우 너무 잦은 재추출이 이루어진다고 볼 수 있다.  $T_{error}$ 는 에러가 누적되기 시작하는 시간으로  $T_{error}$  이상의 시간동안 재추출이 이루어지지 않으면 에러가 누적될 가능성이 크다. 그림 14는 회의자의 움직임을 파악하는 과정의 예를 나타낸 것이다.  $\theta_{Kurtosis}$ 를 25로 설정한 경우로 가정하고 표현하였다.  $\theta_{Kurtosis}$ 보다 작은 침도값이 3번 반복되는 시점에서 재추출이 이루어지고, 그 이후의 시간에서는 일정기간 동안 재추출 여부를 판단하지 않는다.

#### IV. 실험 결과

제안하는 시스템은 Win32 환경에서 Visual Studio 2008 (C++)를 이용하여 개발하였으며 OpenCV library를 사용하였다. 텔레프레즌스를 제공하는 것을 목적으로 640×480 해상도의 영상과 그에 준하는 영상에 대해 실험하였다. 모든 실험은 Intel(R) Core(TM)2 Duo CPU E8400 @ 3.00GHz PC에서 실시간으로 수행되었다. 그림 15는 데모 시스템의 스크린 샷이다.

앞서 말한바와 같이 30번째 프레임에서 2장의 알고



그림 15. 데모 시스템의 스크린 샷. 모든 과정은 자동적이고 실시간으로 수행된다. (a) 원 영상 (b) 추출된 객체  
 Fig. 15. Screen shots of our demo system. All process is automatic and operated in real-time. (a) original frame (b) extracted object.



그림 16. 다양한 영상에 대한 객체 추출 결과  
 Fig. 16 Object extraction results of various test videos.

리즘으로 자동 객체 추출을 수행한 후 그 이후 프레임부터는 추출된 결과를 이용하여 실시간으로 객체를 추출한다. 3장의 알고리즘은 이전 프레임의 추출 결과를 이용하므로 에러가 누적될 수 있다. 이를 보완하기 위해 III장 3절의 침도를 이용한 객체의 재추출을 수행한다. 이에 더하여 사용자가 추출결과를 관찰 시 에러가 누적된 경우, 사용자가 직접 그림 15의 'Re-Extract' 버튼을 눌러 2장의 알고리즘을 반복한 후 다시 3장의 객체 추출을 계속한다.

III장 2절에서의 최종 객체 추출 시 오류를 줄이기 위해 사용한 모폴로지 연산은 단기 연산으로서 11×11크기의 구성요소를 이용하였다. 또한 배경영역에서의 유동 잡음에 의한 작은 오검출 영역을 제거하기 위해 7×7 크기의 구성요소를 이용하여 열기연산을 통한 후처리를 수행하였다. 본 실험에서 식 (7)과 (8)의  $T_{change}$  와  $T_{still}$  은 1로 설정하였다.

본 실험에서는 강건한 객체 추출 결과와 실시간 동작 성능을 평가하기 위하여 다양한 영상을 대상으로 실험하였다. 그림 16은 각 실험 영상에 대한 30번째 첫프레임, 2장의 방법에 의한 그의 씨드 영상, 그래프 컷 분할

표 1. 침도 모듈을 제외하고 수행한 객체 추출 속도 (단위: frames/sec)

Table 1. Object extraction processing speed without Kurtosis module(unit: frames/sec).

	30번째 프레임에서의 객체 추출 속도 (얼굴 검출, 씨드 할당, 그래프-컷 영상분할 포함)	31번째 프레임부터의 객체 추출 속도
실험 영상 1	1.32	30.53
실험 영상 2	1.39	29.88
실험 영상 3	1.34	27.33
실험 영상 4	1.41	28.13
평균	1.36	28.96

표 2. 침도 모듈을 포함하여 수행한 객체 추출 속도 (단위: frames/sec)

Table 2. Object extraction processing speed with Kurtosis module(unit: frames/sec).

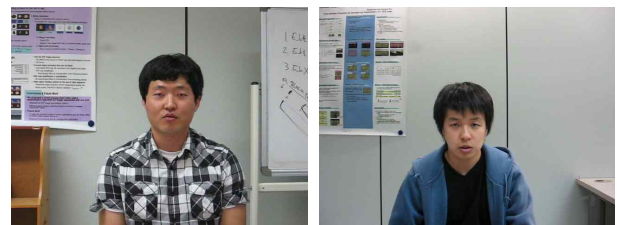
	침도에 의한 재추출 회수	침도에 의한 재추출을 포함한 31번째 프레임부터의 객체 추출 속도
실험 영상 1	0	28.40
실험 영상 2	1	23.62
실험 영상 3	0	25.25
실험 영상 4	0	25.92
평균	-	25.79

로 추출된 객체영상, 배경 에지맵  $E_b$ 을 나타낸 것이다. 그 다음 각 영상의 실시간 추출 결과들을 보였다.

표 1과 표 2는 제안하는 알고리즘의 수행속도를 나타낸 것이다. 수행시간 측정을 위해 300프레임 길이의 640×480 VGA 영상에 대해 실험하였으며 초당 프레임 속도를 나타내었다. 표에서 알 수 있듯이 30번째 프레임에서의 객체 추출 후에는 평균 28.96fps (침도 모듈 제외)의 실시간 속도로 객체가 추출되고 있음을 알 수

표 3. 재현률(Recall)과 정확률(Precision) 측정 결과 Table 3. Recall and precision measurements.

	재현률(Recall)	정확률(Precision)
실험 영상 1	0.99	0.98
실험 영상 2	1.0	0.98
실험 영상 3	1.0	0.96
실험 영상 4	1.0	0.93



(a)



(b)

그림 17. 다른 배경을 가진 화상 회의자들에게 같은 배경을 주어 텔레프레즌스 제공하기. (a) 각기 다른 공간에 존재하는 회의 참석자들 (b) (a)의 회의 참석자들의 객체가 미리 준비된 배경안에 합성되어 두 회의자들이 같은 공간에 있는 듯한 느낌을 준다.

Fig. 17. Extracting the foreground and background of video and providing same background to all videoconferencing participants. (a) videoconferencing participants existing in different places. (b) objects of (a) can be easily immersed into prepared background, which allows a person to feel as if they were in same place.

있다. 표 3은 추출된 객체의 정확성을 측정하기 위해 식 (14)와 같이 정의되는 재현율(Recall)과 정확률(Precision)을 나타낸 것이다.

$$\begin{aligned} recall &= \frac{|OBJ_{GT} \cap OBJ_{Test}|}{|OBJ_{GT}|} \\ precision &= \frac{|OBJ_{GT} \cap OBJ_{Test}|}{|OBJ_{Test}|} \end{aligned} \quad (14)$$

$OBJ_{GT}$ 와  $OBJ_{Test}$ 는 각각 수동으로 추출한 영상의 객체 영역(Ground Truth)과 제안된 방법을 통해 얻어진 객체 영역을 뜻하며,  $|\bullet|$ 는 영역의 픽셀 수를 나타낸다. 각 영상의 31번째 프레임부터 10개의 프레임에 대해 계산 후 평균하였다. 실험 영상 3과 4에서 정확률이 저하되는 이유는 영상의 심한 잡음으로 인해 차영상의 에지가 오검출되기 때문이다. 영상의 잡음은 카메라의 영상 취득 과정에서 발생할 수 있으며, 영상의 인코딩 및 디코딩 과정에서도 발생할 수 있다.

그림 17-(a)는 각기 다른 공간에 위치한 화상회의 참석자들의 영상이고 17-(b)는 회의 참석자들이 미리 준비된 배경에 합성되어 두 회의자들이 같은 공간에 있는 느낌을 주어 텔레프레즌스를 제공하는 예이다.

## V. 결론

제안하는 방법에서 우리는 화상회의 시스템에서 상대방 화상 회의자가 마치 자신과 같은 공간에 존재하는 듯한 느낌을 주는 텔레프레즌스를 제공하기 위한 객체 추출 방법을 논의하였다. 텔레프레즌스를 제공하기 위한 화상회의 영상은 객체를 정확하게 추출하면서도 640×480 크기의 영상에 대해 실시간으로 동작할 수 있어야 하기 때문에, 제안하는 알고리즘이 텔레프레즌스를 제공하기 위한 적합한 방법임을 보이기 위해 640×480 크기의 영상들에 대해 실험하였다. 실험 결과에서 보인 것처럼 제안한 시스템은 일반 PC상에서 실시간으로 동작하였으며 양질의 영상에서 정확한 객체 추출 결과를 보이고 있다. 또한 다른 배경을 가진 두 화상회의 영상에 같은 배경을 주어 두 사용자가 같은 공간에 존재하는 느낌을 주는 것을 실험을 통해 알 수 있었다.

향후 과제는 심한 잡음을 가진 영상이나 보다 복잡한 배경을 가진 영상에서도 정확한 추출 결과를 낼 수 있는 보다 강건한 알고리즘을 개발하는 것이다. 또한 640×480 크기의 영상 뿐 아니라 1920×1080 크기의 고화질 영상에

대해서도 실시간으로 동작하면서, 다수의 객체를 정확히 추출하는 것을 들 수 있겠다.

## 참고 문헌

- [1] <http://en.wikipedia.org/wiki/Videoconferencing>
- [2] Steuer, J. "Defining Virtual Reality: Dimensions of Determining Telepresence," *Journal of Communication*, 42(4), 73 - 93. 1992.
- [3] <http://en.wikipedia.org/wiki/Telepresence>
- [4] H. Luo, A. Eleftheriadis, "Model-Based Segmentation and Tracking of Head-and-Shoulder Video Objects for Real Time Multimedia Services," *IEEE Transactions on Multimedia*, vol.5, no.3, pp.379-389, 2003.
- [5] Y. Gaobo, Z. Zhaoyang, "Video object segmentation for head-shoulder sequences in the cellular neural networks architecture," *Real-Time Imaging*, Vol.9, Issue3, pp.171-178, 2003.
- [6] V. Kolmogorov, A. Criminisi, A. Blake, G. Cross, and C. Rother, "Bi-layer segmentation of binocular stereo video," *IEEE International Conference on Computer Vision and Pattern Recognition*, pp.407 - 414, 2005.
- [7] A. Criminisi, J. Shotton, A. Blake, and P. H. S. Torr, "Gaze manipulation for one-to-one teleconferencing," *IEEE International Conference on Computer Vision*, pp. 191 - 198, 2003.
- [8] C. Wang, L. Guan, "Graph Cut Video Object Segmentation using Histogram of Oriented Gradients," *IEEE International Symposium on Circuits and Systems*, pp.2590-2593, 2008.
- [9] C. Kim and J.-N. Hwang, "Fast and Automatic Video Object Segmentation and Tracking for Content-Based Applications," *IEEE Tr. on Circuits and Systems for Video Technology*, vol.12, no.2, pp.122-129, Feb. 2002.
- [10] Y. Boykov and M. P. Jolly, "Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D images," *IEEE International Conference on Computer Vision*, vol. I, pp. 105-112, 2001.
- [11] W. E. Grimson, *From Images to Surfaces*. Cambridge, MA : MIT Press, pp. 3 - 5, 1981.
- [12] P. Viola, M.J. Jones, Robust real-time face detection, *Int.J. Comput. Vis.*, 57 (2) 137 - 154, 2004.
- [13] L. Aihong, "Evaluation of Gray Image Definition Based on Edge Kurtosis In Spatial Domain,"

*in'09. First International Workshop on Education Technology and Computer Science, pp.472-475, 2009.*

— 저 자 소 개 —



안 일 구(학생회원)  
2005년 전북대학교 전자정보  
공학부 학사 졸업.  
2008년 한국정보통신대학교 전자  
공학과 석사 졸업.  
2008년~현재 한국과학기술원  
박사과정.

<주관심분야 : 3D 영상처리, 패턴인식, 지능형 디  
스플레이>



오 대 영(학생회원)  
2009년 한밭대학교 전자공학과  
학사 졸업.  
2009년~현재 한국과학기술원  
석사과정.  
<주관심분야 : 컴퓨터비전, 영상  
이해, 패턴인식>



김 재 광(학생회원)  
2010년 연세대학교 전기전자  
공학과 학사 졸업.  
2010년~현재 한국과학기술원  
석사과정.  
<주관심분야 : 영상이해, 3D 영상  
처리, 패턴인식>



김 창 익(정회원)-교신저자  
1989년 연세대학교 전기공학과  
학사 졸업.  
1991년 포항공과대학교 전기전자  
공학과 석사 졸업.  
2000년 워싱턴주립대학교  
전기공학과 박사 졸업.

2000년~2005년 Epson Palo Alto Lab.  
책임연구원  
2005년~2009년 한국정보통신대학교 부교수.  
2009년~현재 한국과학기술원 부교수  
<주관심분야 : 3D 영상처리, 의료영상, 영상이  
해>