색체 및 깊이정보를 이용한 실시간 다중얼굴 추적 방법

*장수진 *김윤환 *김혜은 *이우인 *김동환 *윤선아 *유희용 *김우열 *서영호 *김동욱

*광운대학교

*wykim@kw.ac.kr

Real-time Multi-face Tracking Method using Color and Depth Information

*Jang, Su-Jin *Kim, Yoon-Hwan *Kim, Hye-Eun *Lee, Woo-In *Kim, Dong-Hwan *Yoon, Sun-Ah *Yu, Hee-Yong *Kim, Woo-Youl *Seo, Young-Ho *Kim, Dong-Wook

*Kwangwoon University

요약

본 논문에서는 키넥트 센서의 RGB영상을 이용하여 얼굴을 검출하고 검출된 영역의 깊이정보를 템플릿으로 사용하여 다수개의 얼굴을 추적하는 방법을 제안한다. 이 논문은 [1]의 단일 얼굴 추적방법을 다수의 얼굴을 추적하도록 확장한 것이다. 다수의 얼굴추적을 실시간으로 처리하기 위하여 영상을 down sampling 하여 사용한다. 얼굴 검출은 기본적으로 기존의 Adaboost 방법을 사용하나, 피부색만을 이용, 탐색영역을 최대한 축소하여 수행 시간 및 오검출율을 줄인다. 얼굴추적은 깊이정보를 템플릿으로 하며, 깊이값에 따라 크기, 탐색영역을 조정하고, 또한 일정 프레임마다 얼굴을 검출하며 겹침, 새로 나타남, 영상 밖으로 사라짐 등의 얼굴추적 시 발생하는 문제를 해결한다.

1. 서론

물체나 인간의 생체 일부를 검출하고 추적하는 것은 컴퓨터 비전 분야를 비롯한 다양한 분야에서 오래전부터 연구되어 왔으며, 현재 보안시스템, 화상회의, 로봇 비전, HCI(human-computer interface)에 의한 대화형시스템, 스마트 홈 등에 널리 사용되고 있다. 이 중 얼굴에 대한 연구가 가장 활발히 연구되어 왔으며, 그 목적은 빠르고 정확한 얼굴의 검출과 추적이다.

기 제안된 얼굴검출 방법은 연구자의 시각에 따라 여러 방법으로 분류되고 있는데, 주로 지식-기반 방법, 특징-기반 방법, 템플릿 매칭(template matching) 방법 등으로 분류한다. 지식-기반 방법은 많은 수의 사람얼굴을 미리 학습하여 사람얼굴의 형태에 무관하게 검출 및 추적이 가능하도록 하는 방법이다[2]. 특징-기반 방법은 얼굴의 특징 성분인 얼굴요소, 질감 정보, 피부색, 또는 이들을 복합적으로 사용하여 얼굴을 검출한다[3]. 템플릿 매칭 방법은 수동적으로 미리 대상이 되는 모든 얼굴에 대한 표준 얼굴패턴을 만들고 이를 입력영상과 비교하여 얼굴을 검출하는 방법이다[4].

본 논문에서는 팸플릿 매칭 방법을 사용한다. 기본적으로 [1]의 방법을 사용하며, [1]에서 단일 얼굴을 추적하는 방법을 다수의 얼굴을 추적하도록 확장하며, 이 때 실시간 추적을 최대한 고려한다. 본 논문에서 사용하는 영상은 컬러의 texture 영상과 그에 해당하는 깊이영상이며, 이 두 종류의 영상은 Microsoft사의 Kinect를 사용한다. 다수의 얼굴을 실시간으로 추적하기 위하여 원 영상들을 down-sampling 하여 사용하는데, 본 논문에서는 가로, 세로 모두 1/2로 down-sampling 하여 사용한다. Down-sampling 된 영상에서 얼굴검출 또는 추적 오차는 원영상으로 up-sampling하여 보상한다.

2. 참고논문의 얼굴검색 및 추적

본 논문의 방법은 기본적으로 [1]의 얼굴검색 및 추적방법을 그대로 사용한다. [1]의 방법은 단일 얼굴을 검출하고 추적하는 방법을 제안하였으며, 그 방법을 그림 1과 같다.

2.1 얼굴 검출 방법

얼굴검출 방법은 [5]의 Adaboost 방법을 기본적으로 사용한다. 그러나 이 방법에서 얼굴을 검출하는 영역을 줄여 수행시간을 감소시키기 위해 텍스처 영상에서 얼굴의 피부색 영역을 검출하고 깊이영상에서 움직임이 있는 영역을 검출하여, 두 정보의 유무에 따라 얼굴이 있을 가능성이 있는 영역을 추출한다. 아 영역을 대상으로 Adaboost 알고리듬을 적용하여 얼굴을 검출한다.

검출된 얼굴영역에 해당하는 깊이영상을 찾고, 그 깊이영상을 얼굴 추적의 템플릿으로 사용한다.

2.2 얼굴 추적 방법

얼굴추적은 깊이영상만을 사용한다. 초기에는 얼굴검출에서 받은 깊이영상의 템플릿을 시용하며, 추적결과도 깊이영상 템플릿을 출력한다. 추적은 현재위치 주변을 검색하여 현재의 템플릿과 매칭되는 영역을 찾는 것이다. 여기에는 사람의 깊이방향 움직임을 고려하여 현재 템플릿 영역에 해당하는 추적대상 영역의 깊이를 탐색하여 깊이에 따라 템플릿 크기를 보정하는 과정이 포함되어 있다. 또한 현재 템플릿 영역에서 나선형 검색을 시행하고, 정해진 비용함수의 문턱 값보다 비용함수의 값이 작을 때 조기종료(early termination)하는 방법을 사용한다.

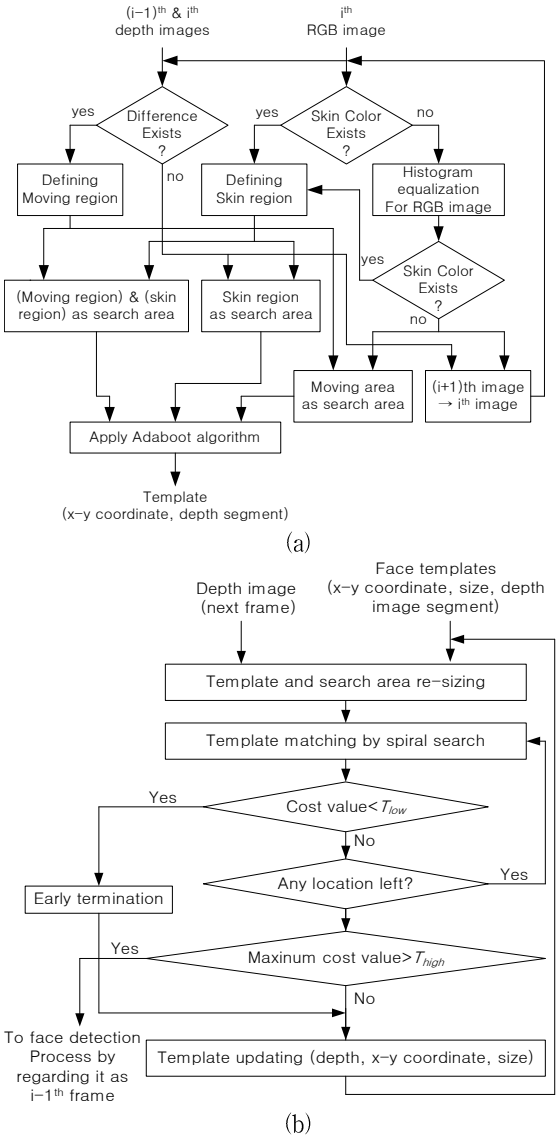


그림 1. 참고문헌 [1]의 얼굴검출 및 추적방법; (a) 얼굴검출, (b) 얼굴추적.

3. 다중 얼굴 검출 및 추적

[1]의 방법은 단일 얼굴을 목표로 하기 때문에 다중 얼굴을 검색하고 추적하기에는 시간이 많이 걸려 실시간 추적이 불가능하다. 따라서 본 논문에서는 다중 얼굴을 추적하기 위한 방법을 [1]의 방법을 변형하여 제안한다. 제안하는 방법은 그림 2에 나타내었다.

3.1 다중 얼굴 검출

여러 얼굴을 검출하는 방법은 그림 2(a)와 같다. 앞에서 언급한 바와 같이 여러 얼굴을 검출하는 방법은 먼저 원 영상을 가로, 세로 모두 down-sampling하여 사용한다. Down-sampling 정도는 검출시간과 밀접한 연관이 있으므로, 필요에 따라 그 양을 조정할 수 있다. 본 논문에서는 최고 3개의 얼굴을 검출하는 것으로 하고, down-sampling 을 가로, 세로 모두 1/2(총 1/4 크기)로 하였다.

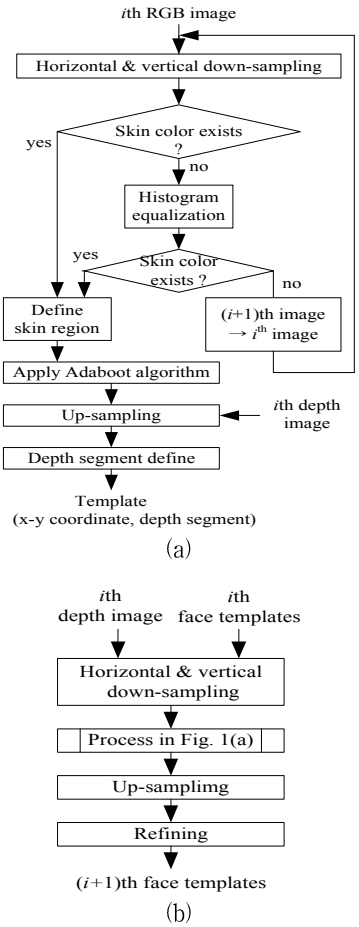


그림 2. 다중 얼굴 검출 및 추적방법; (a) 얼굴검출, (b) 얼굴추적.

그림에서 보듯이, 제안한 방법은 down-sampling 뿐만 아니라 얼굴의 피부색만을 사용한다는 점이 [1]과 다르다. [1]에서는 움직임이 있는 영역을 깊이영상에서 추출하고, 피부색 영역을 텍스처 영상에서 추출하여 두 영역의 공통부분만을 대상으로 Adaboost 알고리듬을 적용하였다. 그러나 여러 개의 얼굴을 검색할 때 움직이지 않은 얼굴과 움직이는 얼굴이 같이 존재할 때 한 얼굴 만 깊이영상으로 감지되기 때문에 움직이지 않는 얼굴은 검출할 수 없다. 따라서 본 논문에서는 피부색 영역만으로 얼굴을 검출한다. 이 경우 깊이영상을 사용할 때보다 더 큰 영역을 검색하게 되어 검출시간이 증가할 수 있다.

3.2 다중 얼굴 추적

본 논문의 얼굴추적은 [1]과 같이 초기에는 얼굴검출에서 얼굴의 템플릿 들을 받아서 사용하며, 추적이 시작되면 추적결과의 템플릿들을 사용하여 추적을 계속한다. 얼굴추적도 얼굴검출과 마찬가지로 추적시간을 단축하여 실시간 추적이 가능하도록 하였다. 이를 위한 방법은 다음과 같다.

3.2.1 Down-sampling 조절

얼굴을 검출하는 것보다 추적하는 프레임이 더욱 많다. 따라서 실시간 처리를 위해서는 얼굴추적 시간을 단축하는 것이 필요하다. 다중

얼굴의 경우 각 얼굴을 추적하여야 하기 때문에 얼굴 수에 비례하여 추적시간이 증가한다. 따라서 필요에 따라서는 영상을 상당히 축소하여 추적을 수행하여야 한다. 그림 3은 필요한 수행속도를 얻기 위해 여러 단계의 down-sampling을 수행할 경우를 나타내고 있다. 예를 들어 가로세로 각각 1/2 씩 3차례 down-sampling 을 수행하면 가로, 세로 모드 1/8로 축소되어 영상은 1/64이 된다.

```
Procedure {multiple face detection or tracking}
input: RGB video, and/or;
output: detected or tracked faces;
begin {
    down-sample the images;{
        down-sample the images;{
            down-sample the images;{
                apply face detection{ } or face tracking{ };
                up-sample the images and templates;
                apply refinement{ };}
            up-sample the images and templates;
            apply refinement{ };}
        up-sample the images and templates;
        apply refinement{ };}}
end
```

그림 3. 다중 얼굴검색/추적을 위한 down-sampling 처리

3.2.2 Up-sampling과 보정

Down-sampling 정도를 크게 할수록 추적결과의 오차가 크게 발생한다. 예를 들어 가로/세로를 1/2로 축소한 경우 1 화소의 오차, 1/4로 축소한 경우 2 화소의 오차 등이 발생할 수 있다. 따라서 down-sample된 영상에서 추적을 수행한 후 그 영상을 up-sampling하여 보정(refine)함으로써 그 오차의 가능성을 줄일 수 있다. 이 때 보정을 위해 추가로 검사하여야 하는 화소는 최대 추적결과 화소 주변의 8개의 화소들이다.

그림 4에 보정방법을 나타내고 있는데, 화소 P가 down-sampling된 영상에서 추적한 결과이고, 그 주위의 흰색 화소들은 up-sampling하였을 때, 즉 down-sampling 하기 전의 주변 화소들 중 P와 대각선 방향의 화소들이다. 보정과정은, 다음과 같다.

① 먼저 4개의 흰색 화소들에 대해 추적검색을 실시한다.
② 그 중 비용함수의 값이 가장 작은 화소(그림에서 C) 주변의 두 화소를 다시 검색한다.
③ 최종 추적화소 P_T는 식 (1)로 결정한다.

$$P_T = \{pixel \mid \min cost(P, C, A, B)\} \quad (1)$$

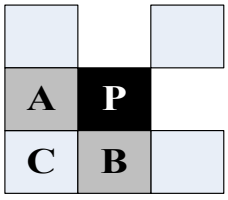


그림 4. 보정방법

3.2.3 정기적 얼굴검출

얼굴을 추적하는 과정에 추적중인 얼굴이 화면 밖으로 사라지거나, 새로운 얼굴이 나타나기도 하고, 한 얼굴이 다른 얼굴 뒤로 가려졌다가 다시 나타나기도 한다. 본 논문의 추적방법은 이전 추적 결과를 다음 추적의 템플릿으로 사용하기 때문에 이런 경우가 발생하면 추적이 실패하거나 오차가 누적되어 큰 오차가 발생할 수 있다.

따라서 본 논문에서는 일정 프레임마다 얼굴검색을 수행하여 위의 경우에 대비하였다. 얼굴검색을 수행하는 주기가 짧을수록 얼굴추적 오차가 줄어드는 반면 얼굴검색 시간이 추적시간보다 길기 때문에 전체적인 얼굴추적시간이 길어진다.

4. 실험 및 결과

본 논문에서 제안하는 방법을 구현하고 여러 테스트 시퀀스를 대상으로 실험을 수행하였다. 구현은 Microsoft window 7 운영체제에서 Microsoft visual studio 2010과 OPEN-CV Library 2.4.5를 이용하였다. 알고리즘의 테스트를 위해 Kinect를 사용하여 자체 제작한 비디오 클립들을 사용하였으며, 최대 3명의 사람이 화면에 나타나도록 하였다. 각 테스트 시퀀스는 200프레임으로 편집하여 사용하였다. 이 실험에서 얼굴검색을 수행하는 주기는 10프레임으로 하였으며, 나머지 파라미터들은 [1]과 동일하게 하였다. 또한 여기서는 가로와 세로 각각 1/2 down-sampling하여 검색 및 추적을 수행하였다.

그림 5는 추적결과의 예를 보이고 있는데, (a)(b)(c)는 640×480의 원영상, (d)(e)(f)는 320×240의 down-sample된 영상이며, 각각 1명, 2명 3명을 추적한 결과이다. 그림에서는 편의상 두 가지 영상의 크기를 동일하도록 320×240 영상을 확대하여 보이고 있다. 두 종류의 추적 결과를 비교하면 약간의 차이가 있지만 거의 유사함을 알 수 있다.

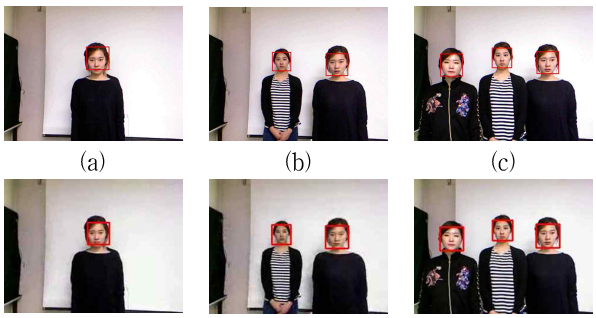
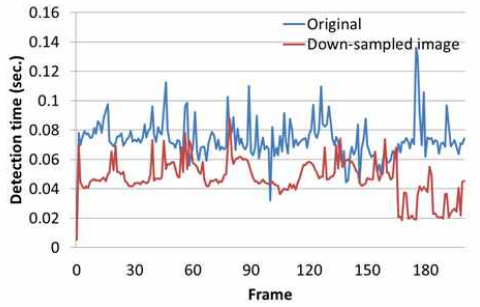


그림 5. 다수의 얼굴추적 예: (a)(b)(c) 640×480 영상, (d)(e)(f) 320×240 영상; (a)(d) 한 얼굴 추적, (b)(e) 두 얼굴 추적, (c)(f) tp 얼굴 추적.

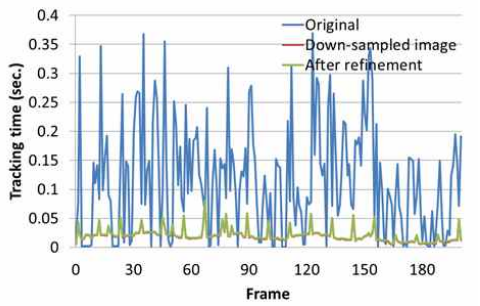
그림 6은 실험한 모든 비디오에 대한 평균 검출 및 추적시간을 프레임에 대해 보이고 있다. 이 두 그래프는 모두 세 개의 얼굴을 검출 또는 추적하는 시간이다. 얼굴검출의 경우 한 얼굴을 검출하는 시간에 비해 세 얼굴을 검출하는 시간이 그 만큼 증가하지 않는데, 이것은 Adaboost를 적용하는 영역이 세배로 커지지 않는다는 것을 보이는 것이다. 얼굴추적의 경우 각 얼굴을 추적하기 때문에 거의 세 배의 시간이 소요되는데, down-sample된 영상의 크기가 1/4이기 때문에 추적시

간이 1/4 정도로 줄어드는 것을 볼 수 있다. 반면, 추적보정에 소요되는 시간은 추적시간에 비해 매우 적음을 알 수 있다.

표 1은 실험한 전체영상에 대해 세 개의 얼굴을 검출/추적한 부분에 대한 평균 시간을 보이고 있다. 640×480 영상의 경우 검출시간은 74[ms]로 크지 않으나, 추적시간은 116[ms]로 나타나 실시간 추적이 불가능하다는 것을 알 수 있다. Down-sample된 320×240 영상으로 검출 및 추적한 결과 검출시간은 47[ms]로 실시간보다 약간 많은 시간이 소요되나, 추적시간은 보정시간을 포함하여 22[ms]로, 충분히 실시간 동작을 수행할 수 있음을 보이고 있다.



(a)



(b)

그림 6 프레임별 평균 검출 및 추적시간; (a) 얼굴검출, (b) 얼굴추적.

표 1. 얼굴검출 및 얼굴추적 평균시간

		얼굴검출[ms]	얼굴추적[ms]
원영상		74	116
down-sample된 영상	추적	47	20
	추적보정	-	2
	합계	-	22

표 2는 원 영상과 down-sample된 영상의 얼굴 검출률을 비교한 것이다. Down-sample된 영상이 약간 낮은 검출률을 보이고 있으나, 충분한 검출률을 나타내고 있다.

표 2. 평균 얼굴검출률

	원영상	down-sample된 영상
검출률	98.9%	95.2%
오검출률	1.05%	4.79%

표 3은 평균 추적오차를 보이고 있는데, 여기서 down-sample된 영상과 보정 후의 추적 오차는 원영상의 크기로 up-sample한 후 측정한 것이다. 예상한 바와 같이 down-sample된 영상에서의 추적오차가 더 크게 나타났으나, 보정과정을 거치면 오히려 원영상보다 추적오차가 줄어드는 것으로 나타났다.

표 3. 평균 추적 오차 (단위: 화소 수)

	원영상	down-sample된 영상	추적보정 후
단일 얼굴	4.78	4.85	4.76
두 개의 얼굴	6.98	7.04	6.64
세 개의 얼굴	7.60	8.13	7.42
평균	6.45	6.67	6.27

5. 결론

본 논문은 [1]의 얼굴검출 및 추적 방법을 다수의 얼굴로 확장하는 방법을 제안하였다. 그 방법으로는 원영상을 원하는 크기로 down-sampling하여 사용하였고, 추적의 경우 다시 up-sampling하여 주변 화소에 대한 보정과정을 거치도록 하였다.

실험결과 제안한 방법이 실시간 동작을 수행할 수 있음을 보였고, 얼굴검출의 경우는 약간의 검출률 손실을 초래하나, 추적의 경우 보정과정에서 오차를 충분히 보상할 수 있었다. 따라서 제안한 방법은 다수의 얼굴을 추적할 때 그 수에 따라 확장하여 유용하게 사용할 수 있으리라 판단된다.

감사의 글

본 연구는 지식경제부 및 한국산업기술평가관리원의 IT산업원천기술개발사업의 일환으로 수행하였음.[KI002058, 대화형 디지털 홀로그램 통합서비스 시스템의 구현을 위한 신호처리 요소 기술 및 하드웨어 IP 개발]

참고문헌

[1] D-W. Kim, W-Y. Kim, J-S. Yoo, and Y-H. Seo, “A Fast and Accurate Face Tracking Scheme by using Depth Information and in addition to Texture Information,” JEET Vol. 9, No. 1, Jan. 2014.

[2] Y. Lin et al., “Real-time Face Tracking and Pose Estimation with Partitioned Sampling and Relevance Vector Machine,” IEEE Intl. Conf. Robotics and Automation, pp. 453-458, 2009.

[3 M. Lievin and F. Luthon; “Nonlinear Color Space and Spatiotemporal MRF for Hierarchical Segmentation of Face Features in Video,” IEEE Trans. Image Processing, vol. 13, No. 1, Jan. 2004.

[4] A. An and M. Chung, “Robust Real-time 3D Head Tracking based on Online Illumination Modeling and its Application to Face Recognition,” IEEE Intl. Conf. Intelligent Robots and Systems, pp. 1466-1471, 2009.

[5] P. Viola and M. J. Jones, "Robust Real-Time Face Detection," Computer Vision, Vol. 52, No. 2, pp. 137-154, 2004.