

피사체 인식을 이용한 비디오 카메라용 자동 초점 및 자동 노출 알고리즘

*이광현 *이상훈

연세대학교

*kwangsabu@yonsei.ac.kr

Advanced Auto-Focus and Auto-Exposure Algorithm Detecting Object for Video Camera

*Kwanghyun Lee *Sanghoon Lee

Yonsei University

요약

현재의 비디오 카메라는 인간의 눈으로 볼 수 있는 것과 동일한 화질의 비디오 영상을 얻을 수 없다. 인간의 눈은 움직이는 물체에 중점을 두고 초점을 맞추는 시각적 특성을 가진다. 하지만 현재의 비디오 카메라 시스템은 영상에 대한 인간의 눈의 시각적 특성에 대한 고려 없이 조리개와 렌즈의 조절 작업이 이루어진다. 본 논문은 이러한 비디오 카메라 시스템의 개선을 위해 인간의 눈의 시각적 특성에 기반을 둔 실시간 피사체 적응 비디오 카메라 시스템을 제안한다. 제안하는 알고리즘은 영상 내에서 움직임을 가진 물체를 찾아낸 뒤, 이것에 가중치를 두고 조리개와 초점제어 작업의 순서로 이루어진다. 움직이는 물체는 영상 코덱의 움직임 정보를 파악하여 찾아내었다. 제안된 알고리즘은 프로그래밍언어로 구현되었으며, 초점거리와 조리개 제어가 가능한 카메라와 PC를 사용하여 영상을 분석, 검증하였다.

I. 서론

최근 들어 디지털 카메라가 보편화되고 있지만 이러한 카메라로 획득할 수 있는 동영상은 인간의 눈으로 볼 수 있는 영상에 비해 저급의 화질을 가진다. 인간의 눈은 움직이는 물체에 초점을 맞추는 시각적 특성을 가지는데, 현재의 비디오 카메라 시스템에서는 이러한 인간의 시각적 특성에 대한 고려 없이 조리개와 렌즈의 조절 작업이 이루어지기 때문이다.

영상의 초점을 찾기 위한 대표적인 알고리즘으로 Tenengrad method[1], SML (Sum-Modified Laplacian) method[2], SMD (Sum-Modulus Difference) method[3], DoM (Difference of Median) method[4] 등이 있는데, 이러한 알고리즘은 인간의 눈의 시각적 특성에 대한 고려가 없다는 단점이 있다.

본 논문은 현재의 비디오 카메라 시스템의 개선을 위해 인간의 눈의 시각적 특성에 기반을 둔 실시간 피사체 적응 비디오 카메라 시스템을 제안한다. 제안하는 비디오 카메라 시스템은 크게 피사체 인식 단계와 자동 초점 제어 및 노출 제어 단계로 이루어져 있다. 먼저, 피사체 인식 단계에서는 비디오 영상 코덱의 모션 정보를 파악함으로써 움직이는 피사체 인식이 가능케 하였다. 다음으로, 자동 초점 제어 및 노출 제어 단계에서는 이 피사체에 대한 정보를 이용하여 초점 제어와 노출 제어 작업이 이루어진다. 초점 제어 작업을 위한 알고리즘은 기존의 정지 영상 촬영을 위한 자동 초점 제어 알고리즘의 특성을 파악하여 비교적 정확한 초점값을 찾을 수 있는 Tenengrad 알고리즘과 연산량이 적은 DoM 알고리즘을 결합하여 비디오 영상에 적용할 수 있도록 하였다. 이 알고리즘은 연산량이 적은 DoM 알고리즘으로 대략적인 초점을 찾은 뒤, Tenengrad 알고리즘을 이용하여 정확한 초점을 찾는 방법이다.

본 논문은 다음과 같이 구성되어 있다. II장에서는 기존 정지 영상

촬영 시 사용되는 자동 초점·자동 노출 알고리즘에 대해 개괄적으로 설명하고, III에서는 비디오 카메라에 적용할 수 있는 자동 초점·자동 노출 알고리즘을 제안한다. 그리고 IV장에서는 제안하는 알고리즘을 이용한 시뮬레이션의 결과와 그 결과에 대하여 논의하겠다. 마지막으로 V장에서는 결론을 내린다.

II. 제안하는 비디오 카메라 시스템

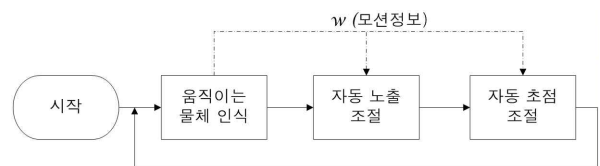


그림 1. 제안하는 비디오 카메라 시스템

제안하는 비디오 카메라 시스템은 그림 1과 같이 크게 세단계로 구성되어 있다. 본 시스템은 먼저, 움직이는 물체 인식 단계로 시작된다. 이 단계를 통해 영상 내의 움직임을 가진 물체를 인식한 뒤, 그 움직임에 대한 정보를 바탕으로 자동 노출·초점 조절이 이루어진다. 각 단계에 대해 자세히 살펴보겠다.

가. 움직이는 물체 인식 단계

영상 내의 움직이는 물체를 인식하는 방법으로 영상 코덱의 모션 정보를 이용한다. 영상 코덱은 두 프레임 사이의 움직임 정도를 파악하여 영상을 압축하게 되는데 그 움직임의 척도가 되는 것이 모션 벡터이다. 모션 벡터의 움직임에 대한 정보는 움직임의 가로축 성분과 세로축 성분으로 표현되어 있다. 움직이는 물체 인식 단계에서는 이 모션 정보를 이용하게 되는데, 모션 벡터의 가로축 성분과 세로축 성분의 절대값

의 크기에 따라, 일정 threshold 값 이상인 범위를 그림 2와 같이 단계별로 지정한다. 단계별로 범위를 지정하고 그 위치에 가중치를 가하는 것은 (1)식을 따라 이루어진다.

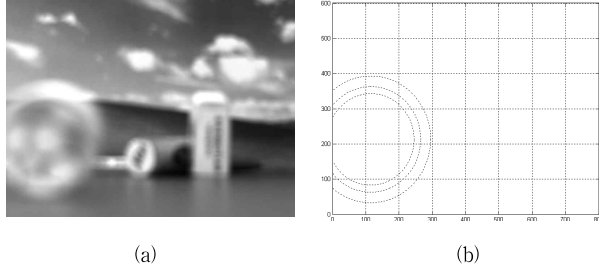


그림 2. 움직임에 따른 가중치

$$w_{ij} = \begin{cases} 10 & 0.7 \times v_{\max} < \sqrt{m_{x_{ij}}^2 + m_{y_{ij}}^2} \\ 5 & 0.5 \times v_{\max} < \sqrt{m_{x_{ij}}^2 + m_{y_{ij}}^2} \leq 0.7 \times v_{\max} \\ 3 & 0.3 \times v_{\max} < \sqrt{m_{x_{ij}}^2 + m_{y_{ij}}^2} \leq 0.5 \times v_{\max} \\ 1 & otherwise \end{cases} \quad (1)$$

나. 자동 노출 조절 단계

두 번째 단계는 자동 노출 조절 단계인데, 앞 단계에서 구한 가중치를 가진 범위에 대한 정보를 이용하여 노출값을 구하게 된다. 이때 노출값이란, 영상의 전 영역에 걸쳐서 휘도 성분의 합을 구한 뒤, 전체 화소의 수를 나눠준 휘도 성분의 평균값이다.

기존의 자동 노출 알고리즘은 노출값이 최적의 값이 되도록 카메라의 조리개를 움직여준다. 조리개를 움직이는 것은 노출값이 최적값이 되는 방향으로 조리개를 움직여가는 방법을 사용한다. 기존의 자동 노출 알고리즘들은 대부분 앞서 설명한 방법에 그림 3과 같이 영상 내의 위치에 따른 가중치를 두는 방법을 사용하였다.[7]

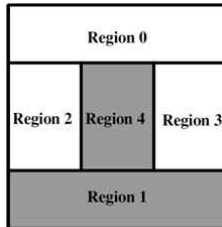


그림 3. 위치에 따른 가중치

제안하는 비디오 카메라 시스템의 자동 노출 조절 단계는 움직이는 물체 인식 단계를 통해 찾은 가중치 정보를 이용하여 (2)식과 같이 노출값을 계산한다. $P(i,j)$ 는 (i,j) 화소의 휘도값이고, w_{ij} 는 움직이는 물체 인식 단계에서 계산한 가중치이다.

$$EV = \frac{\sum_{i=1}^I \sum_{j=1}^J w_{ij} \times P(i,j)}{\sum_{i=1}^I \sum_{j=1}^J w_{ij}} \quad (2)$$

이 단계가 자동 초점 조절 단계보다 앞에 자리한 것은, 카메라의 노출 상태가 초점 조절에 영향을 끼쳐 최적의 노출 상태일 때 가장 초점이 잘 맞게 되지만, 초점 상태는 노출에 영향을 끼치지 않기 때문이다.

다. 자동 초점 조절 단계

세 번째 단계는 자동 초점 조절 단계이다. 자동 초점 조절 단계에서도 첫 번째 단계에서의 움직임에 대한 정보를 바탕으로 초점값을 구하게 된다.

기존 정지 영상 촬영을 위한 카메라의 자동 초점 기능은 크게 두 가지 방식으로 나누어지는데, Active auto-focus 방식과 Passive auto-focus 방식이다. 먼저 Active auto-focus 방식은 카메라로부터 발사된 레이더나 초음파 등의 신호가 피사체에 반사되어 돌아오는 것을 이용하여 거리를 계산하는 방식이다. 이 방식은 피사체와의 정확한 거리를 측정할 수 있지만 추가적인 장치가 필요하다는 단점이 있다. 두 번째 방식인 Passive auto-focus 방식은 추가적인 장치 없이 초점 거리 변화에 따른 영상 신호 자체를 분석하여 정초점일 때의 렌즈 위치를 찾는다. 이 방식은 어두운 곳에서 촬영 시 초점 조절에 어려움이 있지만, 추가적인 장치가 없기 때문에 소비 전력이 적다는 장점이 있다. 이러한 이유로 인해 현재는 Passive auto-focus 방식이 주로 이용되고 있다. Passive auto-focus 방식으로 정초점일 때의 렌즈 위치를 찾는 방법은, 촬영된 영상의 휘도 신호로부터 초점값이라 불리는 초점 평가치를 얻어 그 값이 최대가 되는 방향으로 렌즈를 움직이는 방법이다. 그림 4는 휘도 신호로부터 초점값을 얻어 그 값이 최대치가 되는 곳으로 렌즈를 구동시키는 방법을 나타낸 것이다. 처음의 단계에서부터 시작하여 초점값이 커지는 쪽으로 일정 간격씩 이동하다가 그 값이 감소하면 이전 위치 부근에 최대의 초점값을 가지는 곳이 있다고 판단하고 이전의 위치로 이동한 다음 더 세밀한 간격으로 초점값의 최대치를 찾는다. 초점값을 계산할 수 있는 방법에는 몇 가지가 있는데, 그 중 대표적인 방법이 Tenengrad method와 DoM method이다.

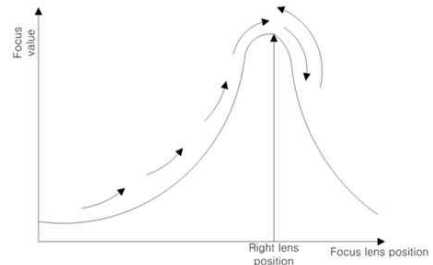


그림 4. 초점값을 이용한 렌즈 구동 방법

Sobel 연산자를 이용하는 Tenengrad 알고리즘은 초점이 맞는 정도에 따라 얻어지는 영상의 선명도가 차이를 보이는 현상을 이용한 방법이다. 수직 및 수평 방향의 에지를 검출하기 위해 두 개의 연산자로 이루어진다.[1]

$$O_{Sobel.x} : \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad O_{Sobel.y} : \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (3)$$

초점값은 (3)식과 이것에 해당하는 화소의 휘도 값, $P(x,y)$, 을 내적을 취한 값, $\nabla g(x,y)$, 를 누적함으로써 얻을 수 있다.

$$FV_{Sobel} = \sum_x \sum_y [\nabla g(x,y)]^2 = \sum_x \sum_y [\{O_{Sobel.x} \cdot P(x,y)\}^2 + \{O_{Sobel.y} \cdot P(x,y)\}^2] \quad (4)$$

DoM(Difference of Median) 알고리즘은 인접한 화소들 간의 미디언 값을 구한 뒤, 일정 간격 사이의 미디언 값의 차를 누적하여 초점값으로 사용한다.[4]

인접 화소들에 대하여 $2m+1$ 의 윈도우 크기를 가진 미디언 필터

를 적용한 결과는 (5)식과 같다.

$$g(x,y) = \text{median}\{f(x-m,y), f(x,y), f(x+m,y)\} \quad (5)$$

또한 고주파 성분을 검출하기 위하여 일정한 간격 δ 만큼 떨어진 $g(x,y)$ 값의 차를 이용하여 그래디언트 G_x 를 (6)식과 같이 구한다.

$$G_x = |g(x,y) - g(x+\delta,y)| \quad (6)$$

그리고 G_x 값을 전 영상에 걸쳐서 누적하여 (7)식과 같이 초점값 FV_{DoM} 을 얻는다.

$$FV_{DoM} = \sum_x \sum_y |G_x| \quad (7)$$

Tenengrad method와 DoM method를 비교해보면 연산량, 연산시간, noise에 대한 민감도, local maximum의 발생빈도 등의 여러 가지 면에서 DoM method가 뛰어나다.[5][6] DoM방식은 median 필터의 특성에 의해서 잡음에 덜 민감하여 impulse성 noise를 제거하는데 탁월한 효과가 있을 뿐만 아니라, 초점값으로 이용되는 고주파성분은 그대로 유지함으로써 기존 방식보다 local maximum의 발생을 줄일 수 있다. 그리고 초점값을 구하는 계산 과정에서 기존의 알고리즘의 산술 연산들을 비교 연산으로 처리하고 수평 라인으로 입력되는 일차원 데이터만을 처리함으로써 처리 시간을 크게 줄일 수 있다. 또한 계산 과정이 간단하여 하드웨어 구현이 용이하다. 하지만, 초점값 곡선의 예리한 정도에서 DoM method 보다 Tenengrad method가 더 뛰어나다.

그러므로 본 논문에서는 자동 초점 조절 단계에 있어서 DoM method와 Tenengrad method를 함께 이용한다. 먼저, 적은 연산량으로 초점값 계산이 가능한 DoM method로 대략적인 정초점의 위치를 탐색한다. 그런 다음, Tenengrad method를 이용하여 정확한 렌즈의 위치를 찾는다. 제안하는 자동 초점 조절 단계의 블록도는 그림 5와 같으며, Tenengrad method를 통한 초점값은 (8)식을 통해, DoM method를 통한 초점값은 (9)식을 통해 계산할 수 있다.

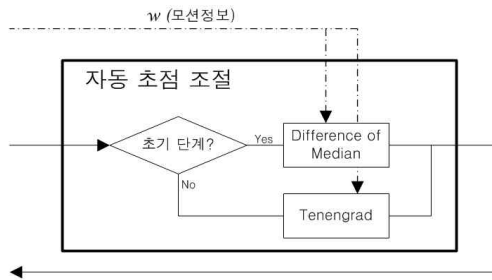


그림 5. 제안하는 자동 초점 조절 단계

$$FV_{Sobel} = \sum_{i=1}^I \sum_{j=1}^J w_{ij} \times [\nabla g(i,j)]^2 \quad (8)$$

$$= \sum_{i=1}^I \sum_{j=1}^J w_{ij} \times [\{O_{Sobel,x} \cdot P(i,j)\}^2 + \{O_{Sobel,y} \cdot P(i,j)\}^2]$$

$$FV_{DoM} = \sum_{i=1}^I \sum_{j=1}^J w_{ij} \times |G_x| \quad (9)$$

또한 그림 1에서 알 수 있듯이 초점 조절 작업이 끝난 후 다시 움직이는 물체를 찾는 작업으로 돌아가게 된다. 이것은 입력으로 들어오는 data가 동영상이므로 각 프레임 마다 세 단계의 작업이 필요하기 때문이다.

III. 실험 및 고찰

1. 자동 초점 알고리즘

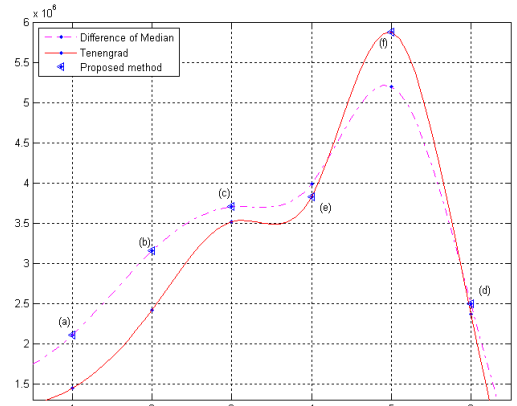


그림 6. 초점값 곡선

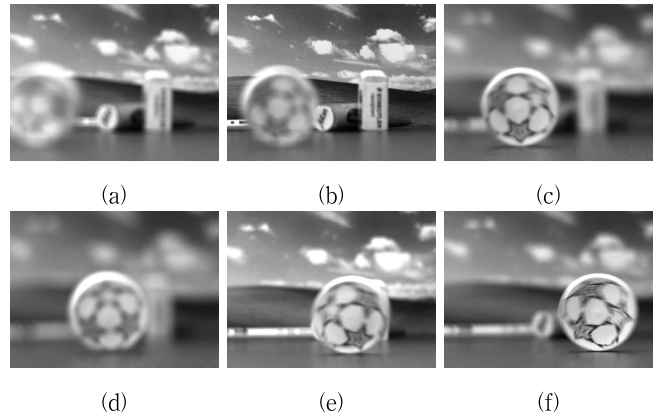


그림 7. 렌즈 위치에 따른 정지영상

	Focus value		
	DoM method	Tenengrad method	Proposed method
(a)	2107568	1443570	2107568
(b)	3151557	2419749	3151557
(c)	3701747	3519870	3701747
(d)	2492031	2361737	2492031
(e)	3984410	3822059	3822059
(f)	5199831	5872402	5872402

표 1. 렌즈 위치에 따른 초점값

실험은 먼저, 영상 내의 움직이는 물체를 찾고, 그 물체의 움직임 정도에 따라 가중치를 두는 작업으로 시작된다. 그런 다음, DoM method를 이용하여 초점값을 구한 다음, Tenengrad method를 이용하여 자세한 초점값을 찾는 과정으로 진행하였다. 그림 6은 움직이는 물체를 가진 영상을 DoM method와 Tenengrad method를 이용하여 초점 조절을 하였을 때의 그래프이다. 점선으로 연결된 DoM method과 실선으로 연결된 Tenengrad method의 몇 지점을 뽑아서 자세히 살펴보면, 그림 7, 표 1과 같다. 이때 (a)에서 (f)까지는 시간 순으로 나열된 것이다. 먼저, 그림 7의 (a)에서 (d)까지는 DoM method를 이용한 것이다. 이때 초점값이 감소하는 지점을 찾을 때까지 초점값이 증가하는 방향으

로 렌즈를 움직여 나가게 되는데, (a)에서 (c)까지 초점값이 증가하다가 (d)에서 초점값이 감소함을 확인할 수 있다. 마찬가지로 그림 7의 (e)에서 (f)까지는 Tenengrad method를 이용한 것이다. DoM method를 통해 찾은 (c)와 (d) 사이의 지점에서 Tenengrad method를 이용하면 정확한 초점값 (f)를 찾을 수 있다.

그림 6에서 알 수 있듯이 Tenengrad method 만을 이용할 경우, local maximum에 빠질 수 있기 때문에 엉뚱한 위치를 정초점으로 판단하는 경우가 발생할 수 있다. 또한 DoM method 만을 이용할 경우, 정 초점일 때의 초점값 곡선이 Tenengrad method 만큼 날카롭지 못하여 정확한 초점값을 찾지 못하는 경우가 발생할 수 있다. 이를 통해 Tenengrad method와 DoM method를 함께 사용하는 것이 적합한 방법이었음을 알 수 있다. 또한 움직이는 물체에 가중치를 가함으로써 움직이는 물체에 맞게 초점을 맞출 수 있다.

2. 자동 노출 알고리즘

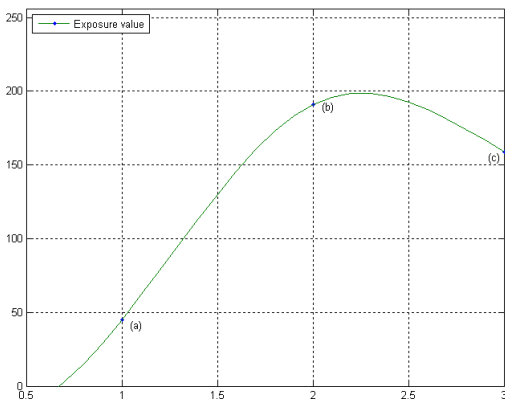


그림 8. 노출값 곡선

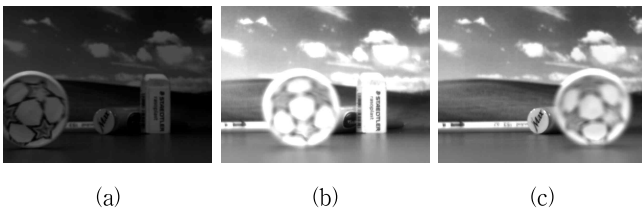


그림 9. 노출 정도에 따른 정지영상

	Exposure value
(a)	45
(b)	191
(c)	159

표 2. 노출 정도에 따른 노출값

자동 노출 실험 역시 자동 초점 실험에서와 같은 방법으로 진행된다. 먼저, 움직이는 물체에 따라 가중치를 적용하고, 그 범위에 따라 노출값을 구한다. 이 과정을 통해 현재 카메라의 노출 상태를 파악하고, 최적값보다 어두울 경우에는 조리개를 열고, 밝을 경우에는 조리개를 닫는 작업을 진행한다. 노출 조절 과정 중 몇 지점을 뽑아서 자세히 살펴보면, 그림 9, 표 2와 같다. 이때 (a)에서 (c)까지는 시간 순으로 나열된 것이다. 그림 9와 표 2에서 확인할 수 있듯이 어두웠던 영상을 움직이는 물체에 맞게 조리개를 맞출 수 있다.

IV. 결론

본 논문에서는 인간의 시각적 특성에 따른, 움직이는 물체에 맞게 초점·노출을 자동으로 조절할 수 있는 비디오 카메라 시스템을 제안하였다. 현재까지의 정지 영상 촬영에 한정되어있던 자동 초점·노출 조절 기능을 비디오 카메라 시스템에 적용함으로써 많은 효율을 가져올 것으로 예상된다. 그뿐 아니라, 움직이는 물체에 맞는 초점·노출 조절을 통해 좀 더 인간의 시각적 특성에 맞는 초점·노출 조절 작업이 가능하게 되었다.

V. 감사의 글

이 논문은 2008년도 정부(과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. R01-2007-000-11708-0). 또한 본 연구는 서울시 산학연 협력사업 중 (과제번호:11136) 보유기술 사업화 지원 사업의 연구 결과로 수행되었음.

참고문헌

- [1] J. M. Tenenbaum, Accommodation in computer vision, Ph.D. thesis, Stanford University, 1970.
- [2] S. K. Nayar and Y. Nakagawa, "Shape from Focus", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, no. 8, pp.824-831, August 1994.
- [3] R. A. Jarvis, "Focus optimization criteria for computer image processing", Microscope, vol. 24, no. 2, pp. 163-180, 1976.
- [4] S. H. Lee et al, "Auto-focusing algorithm for video camera using the difference of medians", KICS vol. 23, no. 1, pp. 41-50, 1998
- [5] KR PARK, "System of the Real-Time Acquisition and Recognition for Iris Images", IEICE trans. Fundamentals, vol. E88 - A, no. 9 September 2005.
- [6] Yoon Hee-Sang, Park Tae-Hyung, "A Fast Focusing Method for CCM Auto-Focusing Handlers", Robotics and Biomimetics, ROBIO IEEE International Conference on pp. 767-772, December 2006.
- [7] June-Sok Lee, You-Young Jung, Byung-Soo Kim, Sung-Jea Ko, "An Advanced Video Camera System with Robust AF, AE, and AWB Control", IEEE Transactions on Consumer Electronics, vol. 47, no. 3, August 2001.