

# 깊이 정보를 이용한 줌 움직임 추정 방법

권순각<sup>†</sup>, 박유현<sup>\*\*</sup>, 권기룡<sup>\*\*\*</sup>

## 요 약

동영상의 줌 움직임 추정은 구현이 아주 복잡하다. 본 논문에서는 줌 움직임 추정을 구현하기 위하여 깊이 카메라와 색상 카메라를 동시에 이용하는 방법을 제안한다. 깊이 카메라로부터 현재블록과 참조블록 사이의 거리 정보를 얻고, 이 거리 정보로부터 두 블록사이의 줌 비율을 계산한다. 줌 비율에 맞게 참조블록을 확대 또는 축소시켜 줌으로서 움직임 추정 차신호를 줄일 수 있다. 따라서, 제안된 방법은 줌 움직임 추정을 위한 복잡도가 크지 않으면서 움직임 추정 정확도를 높이는 것이 가능하다. 모의실험을 바탕으로 제안된 방법의 움직임 추정 정확도를 측정하였으며, 기존 블록정합 방법에 비하여 움직임 추정 오차값이 크게 감소함을 확인하였다.

## Zoom Motion Estimation Method by Using Depth Information

Soon-Kak Kwon<sup>†</sup>, Yoo-Hyun Park<sup>\*\*</sup>, Ki-Ryong Kwon<sup>\*\*\*</sup>

## ABSTRACT

Zoom motion estimation of video sequence is very complicated for implementation. In this paper, we propose a method to implement the zoom motion estimation using together the depth camera and color camera. Depth camera obtains the distance information between current block and reference block, then zoom ratio between both blocks is calculated from this distance information. As the reference block is appropriately zoomed by the zoom ratio, the motion estimated difference signal can be reduced. Therefore, the proposed method is possible to increase the accuracy of motion estimation with keeping zoom motion estimation complexity not greater. Simulation was to measure the motion estimation accuracy of the proposed method, we can see the motion estimation error was decreased significantly compared to conventional block matching method.

**Key words:** Zoom Motion Estimation(줌 움직임 추정), Depth Information(깊이 정보), Zoom Ratio(줌 비율)

## 1. 서 론

동영상 부호화 표준으로 H.264 및 MPEG-4 Part 10 부호화 기법[1,2]이 제정되었으며, 현재에는 HEVC 표준[3-5]이 완료 단계에 있다. 동영상 부호화에서 움직임 추정 및 보상 방법은 시간방향의 중복

성을 제거하기 위해 필수적으로 사용된다. H.264, HEVC 등에서는 움직임 추정을 위해서 현재 화면내의 블록(공간방향으로 이웃한 화소들의 묶음)과 시간방향으로 이웃한 참조 화면내에서 가장 가까운 블록을 찾는 블록정합 방법을 사용하고 있다.

블록정합 움직임 추정 과정에는 블록을 찾는 평가

※ 교신저자(Corresponding Author): 권순각, 주소: 부산광역시 부산진구 엄광로 995번지 동의대학교 컴퓨터소프트웨어공학과(614-714), 전화: 051) 890-1727, FAX: 051) 890-2629, E-mail: skkwon@deu.ac.kr  
접수일: 2012년 9월 24일, 수정일: 2012년 11월 20일  
완료일: 2012년 11월 27일

<sup>†</sup> 종신회원, 동의대학교 컴퓨터소프트웨어공학과

<sup>\*\*</sup> 정회원, 동의대학교 컴퓨터소프트웨어공학과  
(E-mail: yhpark@deu.ac.kr)

<sup>\*\*\*</sup> 종신회원, 부경대학교 IT융합응용공학과  
(E-mail: krkwon@pknu.ac.kr)

척도, 참조 화면내 탐색영역의 크기, 현재 블록의 크기 등이 고려되어야 한다[6]. 움직임 추정 은 동영상 부호화에서 구현 복잡도의 70% 이상을 차지하고 있으므로, 동영상 부호화 초창기부터 복잡도를 줄이는 고속 움직임 추정방법에 대한 연구가 진행되어져 왔다. 이러한 블록정합 움직임 추정은 카메라의 좌우 이동 및 영상 속 객체의 공간 이동 등에는 추정의 정확도가 높지만, 영상 줌에는 정확도가 떨어질 수밖에 없다.

영상의 줌 움직임은 현재 화면이 참조 화면에서 확대 또는 축소되는 것을 의미하며, 확대 및 축소에 해당되는 줌 비율을 정확하게 추출하기 위해서는 가능한 모든 줌비율을 적용할 수 밖에 없다. 가능한 줌 비율의 수가 너무 많기 때문에 모든 경우를 적용하기에는 불가능하다. 따라서, 구현의 복잡도를 줄이기 위한 방법에 대한 연구가 진행되어 왔다. 줌 움직임 추정에서 첫 번째 단계에는 선택된 몇 개의 화소들에 대하여 간략하게 움직임을 추정하고, 두 번째 단계에서 모든 화소에 대하여 정확도를 높이는 방법[7], 계산 효율, 정확도 등을 고려한 반복적 최소-자승 추정에 대한 다양한 개선방법[8], 보간된 참조 화면을 이용하는 방법[9], 3-D 다이아몬드 탐색 패턴을 사용하여 움직임 추정을 단순화하는 방법[10]이 제안되었다.

그러나, 기존 대부분의 연구는 카메라의 색상 정보만을 사용하기 때문에 정확한 줌 움직임 벡터를 추정하는 것에 어려움에 있고, 추정의 복잡도를 줄이는 데에도 한계가 있을 수 밖에 없다. 따라서, 본 논문에서는 색상 카메라와 깊이 카메라를 동시에 이용하여 간단하게 줌 움직임을 추정하는 방법을 제안한다.

제안된 방법은 깊이 카메라로부터 카메라와 객체 간의 거리 정보를 얻고, 현재화면내 움직임 추정 블

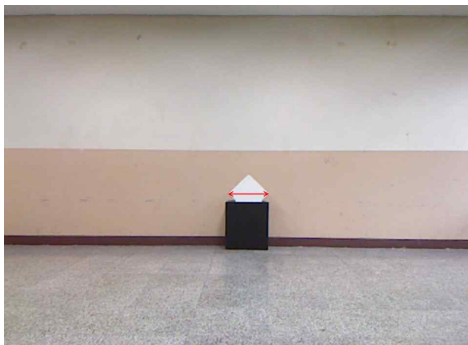
록과 참조화면내 탐색 블록의 평균 거리 값을 비교하며, 거리 값의 비율에 따라 줌 비율을 결정한다. 줌 비율에 따라 참조 블록을 확대 또는 축소하고, 현재 블록에 대한 움직임 추정치 신호를 구한다. 이 신호 값은 움직임 보상 부호화기에 입력된다.

본 논문의 구성은 2장에서는 깊이 및 색상 카메라를 이용하여 카메라와 객체 간의 거리정보를 측정하고, 거리별 색상 정보의 줌 비율의 관계식을 조사한다. 3장에서는 거리정보에 따라 객체의 줌 움직임을 추정하는 방법을 제안한다. 4장에서는 제안한 줌 움직임 추정 방법에 대한 움직임 추정 정확도를 모의실험을 통하여 살펴본다. 마지막으로 5장에서 결론에 대해 기술한다.

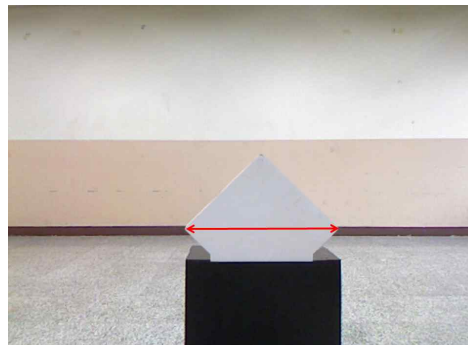
## 2. 깊이 정보와 색상 정보의 관계

객체가 카메라로부터 가까이 있으면 해당되는 색상정보는 확대가 되고, 반대로 카메라로부터 멀어지면 색상정보는 축소가 될 것이다. 객체와 카메라 사이의 거리 정보를 고려하여 색상정보의 줌 비율을 결정할 수 있을 것이다. 이를 위해서는 사용되는 깊이 및 색상 카메라의 특징을 파악해야 한다. 본 논문에서는 Kinect 깊이 및 색상 카메라를 이용하며, 실험을 통하여 두 카메라 사이의 관계를 조사한다. 깊이 카메라는 카메라와 객체와의 거리를 측정할 수 있으며, 측정된 피사체의 거리 정보는 화면으로 구성되며, 화면은 각 화소(pixel) 별로 구성된다[6].

그림 1과 같이 다이아몬드 형태의 객체를 1m에서 4m 사이에 배치하고, 50cm 간격으로 이동하면서, 깊이 정보와 색상 정보를 조사하였다. 색상 정보는 다이아몬드 수평 대각선 방향의 화소 수를 측정하였다.



(a) 4m



(b) 1m

그림 1. 다이아몬드 객체의 거리별 화소수 측정

그림 2는 거리에 따라 측정된 화소수를 나타내고, Fitting 알고리즘을 바탕으로 근사화될 수 있는 식도 같이 나타낸다.

거리 (R)와 화소수 (P)는 다음 식과 같이 근사화됨을 알 수 있다.

$$P = \frac{\alpha}{R^\beta} + \gamma \quad (1)$$

여기서,  $\alpha$ ,  $\beta$ ,  $\gamma$ 는 상수값이며,  $\alpha=221.65$ ,  $\beta=0.9$ ,  $\gamma=-9.65$ 로 조사되었다. Kinect 카메라가 올바르게 작동하는 거리 범위가 1m~4m 사이임을 고려하여 그 범위내로 국한하여 조사하였다.

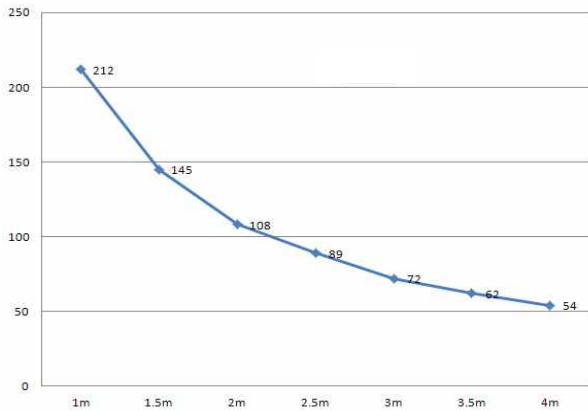


그림 2. 거리와 화소수 관계

### 3. 깊이 정보를 이용한 줌 움직임 추정 방법

현재까지 동영상 표준화 기법은 움직임 추정을 위해서 블록정합 방법이 사용되고 있다. 블록정합은 현재화면을 특정 크기로 나누고 참조화면에서 가장 가까운 블록을 추정하는 과정이다. 움직임 추정을 위한 현재 화면내 블록크기는 작을수록 추정의 정확도가 커지지만 추정에 필요되는 복잡도가 증가되고, 추가적으로 전송되는 움직임벡터가 증가되는 단점이 있다. H.264에서는 16×16, 8×8, 4×4 등으로 블록크기를 가변적으로 선택할 수 있도록 하고 있다.

본 논문에서는 블록정합과 연계되어 사용할 수 있는 줌 움직임 추정방법을 제안한다. 제안되는 줌 움직임 추정방법에 대한 알고리즘의 흐름도는 그림 3과 같다.

색상 카메라로부터는 동영상의 RGB 또는 YUV 색상정보를 획득하며, 깊이 카메라로부터는 카메라와 객체 사이의 깊이정보, 즉, 거리정보를 획득한다.

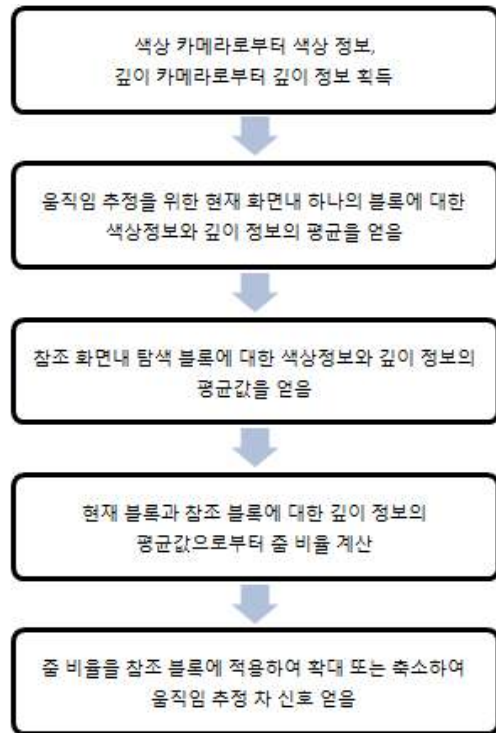


그림 3. 제안된 방법에 대한 알고리즘 흐름도

색상정보는 화소 단위이며, 깊이정보는 mm 단위가 일반적이다. 동영상 움직임 추정을 위해서는 움직임 부호화를 위한 현재화면과 움직임 탐색이 이루어지는 참조화면이 필요하다. 참조화면은 현재화면보다 이전 또는 이후 시간이 모두 가능하다. 현재화면은 특정 크기의 블록단위로 움직임이 추정되며, 해당되는 블록에 대한 색상정보를 가져오고 깊이정보는 블록의 평균값을 구한다. 마찬가지로 참조화면에서도 탐색되는 블록에 대한 색상정보를 가져오고 깊이정보는 블록의 평균값을 구한다. 현재블록과 참조블록의 깊이 정보의 평균값으로부터 참조화면의 줌비율을 계산한다. 계산된 줌비율에 따라 참조화면을 확대 또는 축소하여 현재블록과 참조블록 사이의 움직임 추정 차신호를 구한다.

제안되는 줌 움직임 추정방법은 현재블록내 각 화소의 깊이 정보로부터 블록의 평균 깊이정보  $\bar{R}_c$ 를 구한다. 또한, 참조블록에 대해서도 평균 깊이정보  $\bar{R}_r$ 를 구한다. 식(1)로부터 현재 및 참조 블록에 대한 거리 정보별 화소수를 구하고, 현재블록을 기준으로 기준블록의 줌 비율 (S)을 다음 식과 같이 구한다.

$$S = \frac{P_c}{P_r} = \left( \frac{\alpha}{R_c^\beta} + \gamma \right) / \left( \frac{\alpha}{R_r^\beta} + \gamma \right) \quad (2)$$

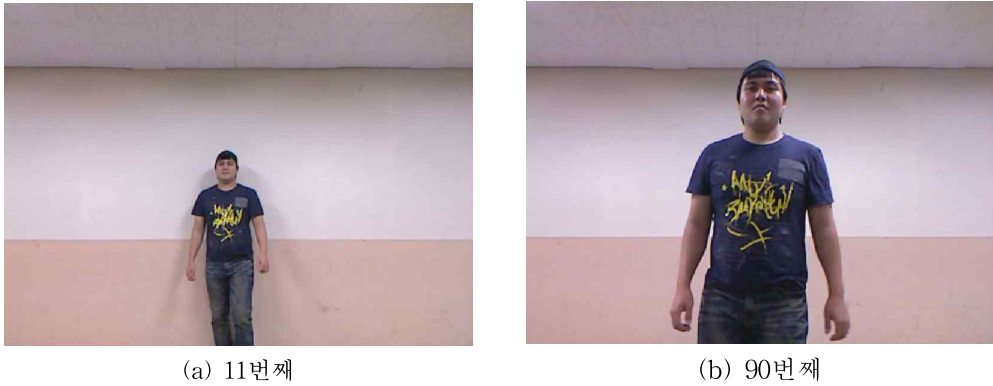


그림 4. 획득한 영상의 두 화면

준비율을 고려하여 참조블록을 확대 또는 축소한다. 확대 또는 축소된 참조블록에 대하여 현재블록과 움직임 추정 차 신호를 구한다.

식(2)를 직접 적용할 수 있으나,  $\gamma = -9.65$ 의 값이 작으므로  $\gamma = 0$ 으로 가정하면, 다음 식과 같이 간단한 관계식이 얻어진다.

$$S = \frac{P_c}{P_r} \cong \left( \frac{\overline{R}_r}{\overline{R}_c} \right)^\beta \quad (3)$$

즉, 줌 비율은 현재블록과 참조블록의 거리 비율에 직접적으로 관계됨을 알 수 있다.  $\overline{R}_r > \overline{R}_c$ 이면,  $S > 1$ 로 되어 참조블록은 확대되고,  $\overline{R}_r < \overline{R}_c$ 이면,  $S < 1$ 로 되어 참조블록은 축소된다.

Kinect 깊이 및 색상 카메라를 사용하여 실제 영상을 획득하고 줌비율을 측정하였다. 영상은 카메라로부터 1m와 4m 거리에서 사람이 자연스럽게 카메라를 기준으로 앞 뒤로 움직이는 특징을 갖는다. 그림 4는 영상의 특정 시간의 두 화면을 나타낸다. 각 화면은 수평방향으로 640화소, 수직방향으로 480화소로 구성되어있으며, 초당 30화면으로 총 120화면을 모의실험에 사용하였다.

식(3)으로 계산한 현재블록과 참조블록 사이의 줌비율은 그림 5에 나타낸다. 움직임 추정은 H.264 IBP 구조에서 움직임 추정을 사용하였으며, 참조블록은 참조화면의 탐색영역 내에서 현재블록과 움직임 보상에 사용되는 블록을 의미한다. 연속하는 P-화면에 대한 줌비율만을 나타내는 것으로서 줌비율로부터 확대, 감소, 확대가 이루어지고 있음을 알 수 있으며, 1.02 이상의 확대도 존재하고, 0.95 이하의 감소도 존재한다.

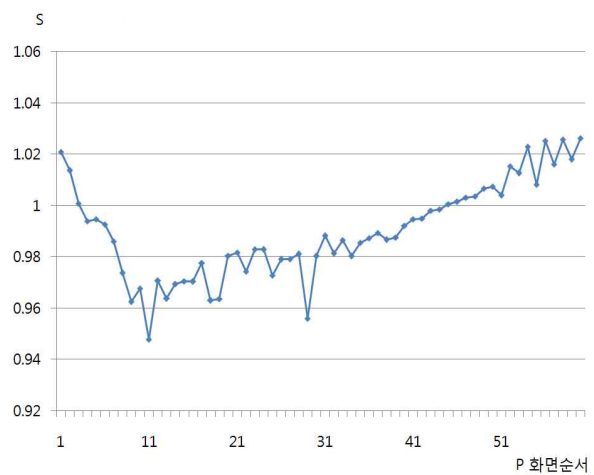


그림 5. P-화면에 대한 줌 비율

#### 4. 모의실험 및 결과

모의실험을 통하여 제안된 줌 움직임 추정방법에 대한 성능을 살펴본다. 실험영상으로는 그림 4와 같이 한 개의 객체와 배경으로 구성된 영상, 그림 6과 같이 두 개의 객체와 배경으로 구성된 영상에 대해 수행하였다.

영상내 화면은 수평방향으로 640화소, 수직방향으로 480화소로 구성되어있으며, 초당 30화면으로 총 120화면을 모의실험에 사용하였다. 움직임 추정은 H.264의 구조를 사용하였으며, 두가지 GOP 구조 (IBPBP., IBBPBBP..)에 대해 살펴보았다. 움직임 추정 정확도를 평가하기 위하여 현재화면과 움직임 보상된 화면 사이의 평균제곱차이 (MSE) 값을 사용하였다.

움직임 추정 블록크기는 수평 및 수직방향으로 각각 16화소이며, 탐색영역 크기는 수평 및 수직방향으

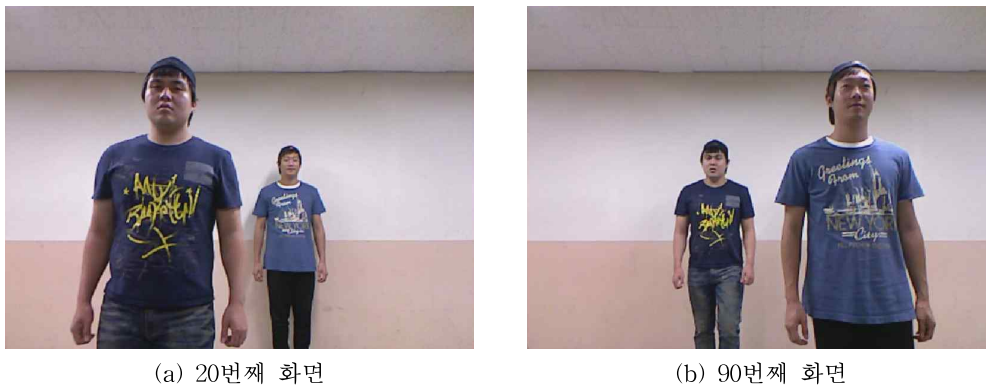


그림 6. 두 개의 객체로 구성된 영상의 화면

표 1. 기존방법과 제안방법의 움직임 추정 성능 비교

객체 수	GOP 구조	전역탐색 MSE	제안방법 MSE
1	IBP	10.65	5.03
	IBBP	11.46	5.49
2	IBP	19.31	9.33
	IBBP	18.77	8.61

로 각각 30화소 경우에 대하여 기존의 전역탐색방법과 제안방법의 성능을 비교하였다. 표 1은 MSE의 평균을 비교한 것으로서 제안 방법이 전역탐색 방법보다 MSE 값이 크게 감소함을 알 수 있다.

그림 7, 8은 객체 수가 1, 2인 경우에 대한 화면별 MSE의 변화를 나타낸다. 제안방법은 P-화면이 B-화면보다 MSE의 감소 정도가 더 큼을 알 수 있다.

P-화면을 위한 현재화면과 참조화면의 사이의 거리가 B-화면을 위한 현재화면과 참조화면 사이의 거리보다 더 크기 때문에, 현재화면과 참조화면 사이의 줌비율이 더 커져서 제안한 방법의 효율이 더욱더 부각된 것으로 이해할 수 있다. 또한, 객체 수가 증가할수록 제안방법의 성능이 더욱더 커지는 것을 알 수 있는데, 객체 수가 증가할수록 화면 전체에서 객체가 차지하는 비율이 증가하고 이로 인해 줌을 고려한 객체의 움직임 추정이 가능하기 때문인 것으로 이해할 수 있다.

따라서, 제안된 방법은 화면간 줌의 비율이 커지고, B-화면의 개수가 증가하고, 배경에 비하여 객체가 차지하는 비율이 커질수록 더욱더 성능이 증가할 것으로 기대된다.

기존의 줌 움직임 추정 방법은 색상카메라 만을

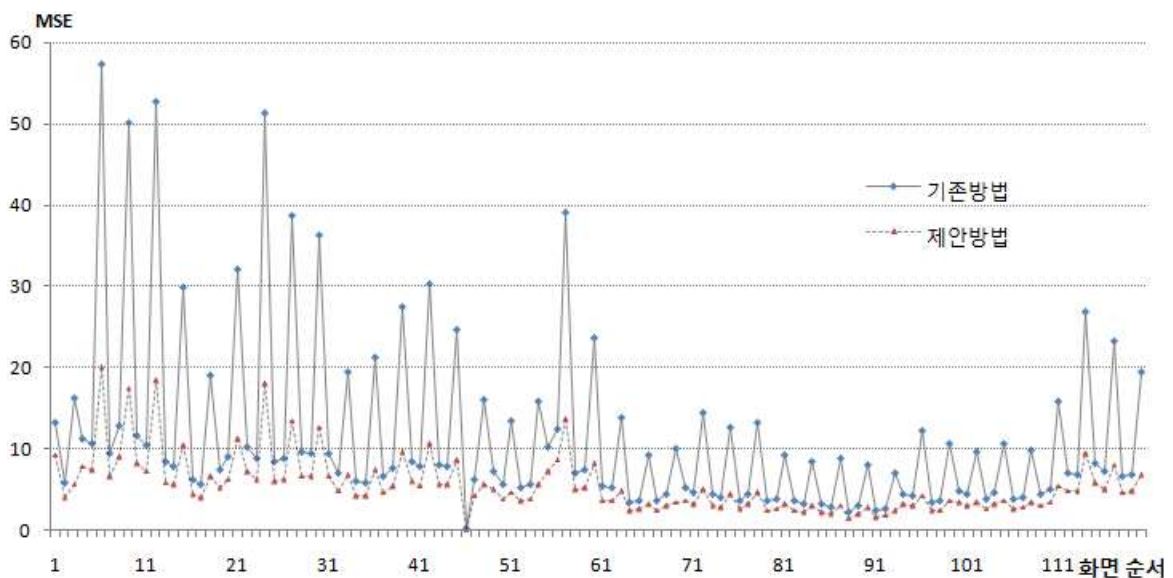


그림 7. 객체 1, IBBP 구조에서 움직임 추정 성능



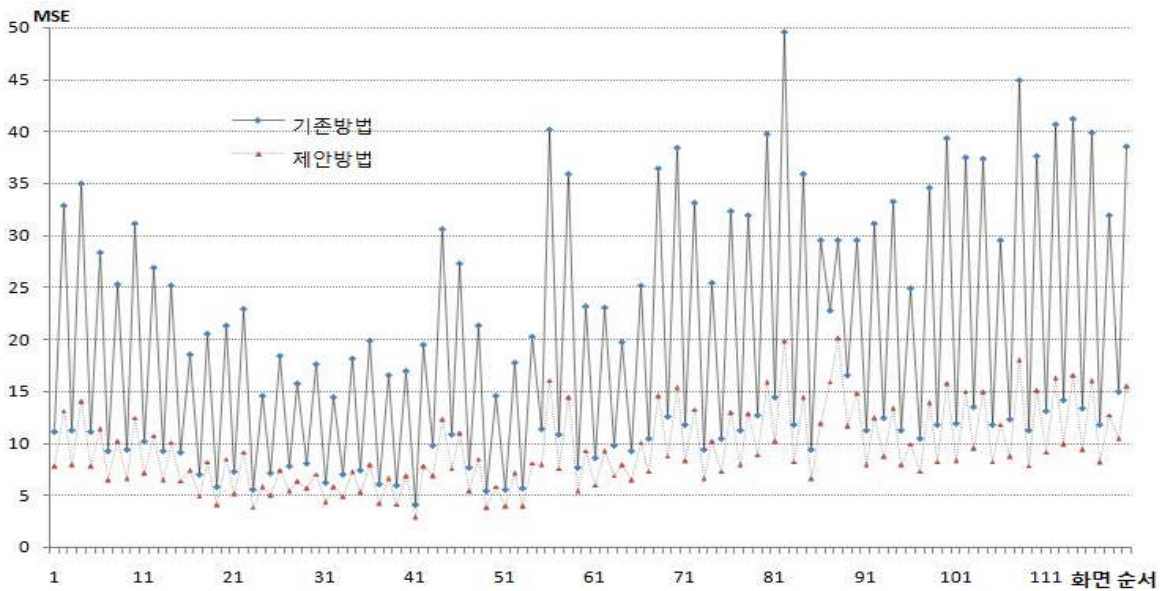


그림 8. 객체 2, IBP 구조에서 움직임 추정 성능

사용하기 때문에 추정의 정확도와 계산량의 상관관계를 고려하여 적절한 타협점을 찾은 것이며, 계산량을 줄이기 위한 방법도 최소 3~5개 이상의 준비율을 적용해야 한다. 따라서, 제안된 방법은 기존 줌 움직임 추정방법보다 계산량을 크게 줄일 수 있다.

### 5. 결 론

블록정합 움직임 추정에는 줌 움직임을 추정하기 어렵다. 본 논문에서는 색상 카메라와 깊이 카메라를 동시에 이용하여 줌 움직임을 간단하게 추정하는 방법을 제안하였다. 깊이 카메라는 카메라와 객체 사이의 거리 정보를 알려주며, 이러한 거리 정보로부터 움직임 추정하는 현재화면내 블록과 추정되는 참조 화면내 블록 사이의 화면의 확대 또는 축소를 나타내는 준비율을 계산한다. 움직임 추정을 위한 차신호는 준비율을 참조블록에 적용하고 현재블록과의 차이 신호를 구하여 사용한다. 객체가 1개 또는 2개인 동영상에 대한 모의실험을 통하여 제안된 방법이 기존 방법에 비하여 움직임 추정 차신호를 크게 줄일 수 있음을 확인하였다. 본 논문에서 제안한 움직임 추정 방법은 줌 움직임 벡터를 전송하는 방법을 추가하면 다양한 국제표준 동영상 부호화기법 표준 등에 적용 가능할 것으로 기대된다. 줌 움직임 벡터의 추가 전송을 위한 비트수와 줄어든 움직임 추정 차신호에 따른 부호화 이득을 동시에 고려하여 최적의 부호화

효율을 얻는 방법에 대한 추가 연구가 필요할 것이다.

### 참 고 문 헌

[1] S.k. Kwon, A. Tamhankar, and K.R. Rao, "Overview of H.264/MPEG-4 Part 10," *Journal of Visual Communications and Image Representation*, Vol. 17, No. 2, pp. 186-216, 2006.

[2] 이호영, 권순각, 이중화, "H.264 동영상 부호화에서 관심영역의 주관적 화질 개선 방법," *멀티미디어학회논문지*, Vol. 12, No. 7, pp. 913-921, 2009.

[3] K. Ugur, K. Andersson, A. Fuldseth, G. Bjøntegaard, L.P. Endresen, J. Lainema, A. Hallapuro, J. Ridge, D. Rusanovskyy, C. Zhang, A. Norkin, C. Priddle, T. Ruser, J. Samuelsson, R. Sjöberg, and Z. Wu, "High Performance, Low Complexity Video Coding and the Emerging HEVC Standard," *IEEE Trans. Circuit Syst. Video Technology*, Vol. 20, No. 12, pp. 1688-1697, 2010.

[4] W.J. Han, J. Min, I.K. Kim, E. Alshina, A. Alshin, T. Lee, J. Chen, V. Seregin, S. Lee, Y.M. Hong, M.S. Cheon, N. Shlyakhov, K. McCann, T. Davies, and J.H. Park, "Improved

Video Compression Efficiency through Flexible Unit Representation and Corresponding Extension of Coding Tools," *IEEE Trans. Circuit Syst. Video Technology*, Vol. 20, No. 12, pp. 1709-1720, 2010.

- [ 5 ] F. Bossen, V. Drugeon, E. Francois, J. Jung, S. Kanumuri, M. Narroschke, H. Sasai, J. Sole, Y. Suzuki, T.K. Tan, T. Wedi, S. Wittmann, P. Yin, and Y. Zheng, "Video Coding using a Simplified Block Structure and Advanced Coding Techniques," *IEEE Trans. Circuit Syst. Video Technology*, Vol. 20, No. 12, pp. 1667-1675, 2010.
- [ 6 ] 권순각, 김성우, "깊이 카메라를 이용한 움직임 추정 방법," 방송공학회 논문지, 제17권, 제4호, pp. 676-683, 2012.
- [ 7 ] A. Ahmadi and S. Talebi, "Fast Global Motion Estimation in Two Sampling Steps," *Majlesi Journal of Electrical Engineering*, Vol. 5, No. 4, pp. 9-15, 2011.
- [ 8 ] S. Sorwar, M. Murshed, and L. Dooley, "Fast Global Motion Estimation using Iterative Least- Square Estimation Technique," *Proc. ICICS-PCM*, pp. 282-286, 2003.
- [ 9 ] L.M. Po, K.M. Wong, K.W. Cheung, and K.H. Ng, "Subsampled Block-Matching for Zoom Motion Compensated Prediction," *IEEE Trans. Circuit Syst. Video Technology*, Vol. 20, No. 11, pp. 1625-1637, 2010.
- [ 10 ] H.S. Kim, J.H. Lee, C.K.. Kim, and B.G. Kim, "Zoom Motion Estimation Using Block-Based Fast Local Area Scaling," *IEEE Trans. Circuit Syst. Video Technology*, Vol. 22, No. 9, pp. 1280 - 1291, 2012.



권 순 각

1990년 2월 경북대학교 전자공학과 졸업  
 1992년 2월 KAIST 전기및전자공학과 석사  
 1998년 2월 KAIST 전기및전자공학과 박사

1997년 3월~1998년 8월 한국전자통신연구원 연구원  
 1998년 9월~2001년 2월 기술신용보증기금 기술평가센터 팀장  
 2003년 9월~2004년 8월 Univ. of Texas at Arlington 교환 교수  
 2010년 9월~2011년 8월 Massey University 교환 교수  
 2001년 3월~현재 동의대학교 컴퓨터소프트웨어공학과 교수  
 관심분야: 멀티미디어신호처리, 영상통신



박 유 현

1996년 2월 부산대학교 전자계산학과 졸업  
 1998년 2월 부산대학교 전자계산학과 석사  
 2008년 8월 부산대학교 전자계산학과 박사

2000년 1월~12월 한국국방연구원 연구원  
 2001년 1월~ 2009년 2월 한국전자통신연구원 선임연구원  
 2009년 3월~현재 동의대학교 컴퓨터소프트웨어공학과 조교수  
 관심분야: 인터넷 시스템, 클라우드 시스템, 융합IT 서비스



권 기 룡

1986년 경북대학교 전자공학과 학사 졸업(공학사)  
 1990년 경북대학교 전자공학과 석사 졸업(공학석사)  
 1994년 경북대학교 전자공학과 박사 졸업(공학박사)

2000년~2001년 Univ. of Minnesota, Post-Doc.  
 1996년~2005년 부산외국어대학교 디지털정보공학부 부교수  
 2011년~2012년 Colorado State Univ., Visiting Scholar  
 2006년~현재 부경대학교 IT융합응용공학과 교수  
 2012년~현재 한국멀티미디어학회 총무및조직담당부회장  
 관심분야 : 멀티미디어정보보호, 디지털영상처리, 멀티미디어통신및신호처리