

일반논문-04-09-3-08

3차원 중간영상의 합성을 위한 쿼드트리기반 변이추정 방법

성준호^{a),b)†}, 이성주^{b)}, 김성식^{b)}, 하태현^{a),b)}, 김재석^{a)}

A Quadtree-based Disparity Estimation for 3D Intermediate View Synthesis

Junho Sung^{a),b)†}, Seongjoo Lee^{b)}, Sungsik Kim^{b)}, Taehyun Ha^{a),b)} and Jaeseok Kim^{a)}

요 약

스테레오 방식 및 다시점 3차원 영상시스템에서, 관찰자에게 록어라운드 기능 및 연속적인 운동시차를 부여하여 보다 안락한 3차원 현실감을 제공하기 위해서 중간영상의 합성이 요구된다. 쿼드트리기반의 변이추정은 그 알고리즘과 하드웨어 구현의 간편성 때문에 중간영상의 합성을 위한 유력한 방법 중 하나이다. 본 논문에서는 쿼드트리기반의 변이추정을 이용하여 합성된 중간영상의 물체 간 경계에서 발생하는 플리커링을 제거하기 위하여 두가지의 아이디어를 제안한다. 첫째, 새로운 분할방법은 변이추정 동안 보다 일관된 쿼드트리 분할을 부여한다. 둘째, 이전프레임과 현재프레임의 상관성을 이용하는 적응적 시간평활화는 변이추정의 오류를 감소시켜준다. 두가지의 제안된 아이디어는 여러 스테레오영상을 이용해 검증되었으며, 플리커링이 상당히 감소됨을 확인하였다.

Abstract

In stereoscopic or multi-view three dimensional display systems, the synthesis of intermediate sequences is inevitably needed to assure look-around capability and continuous motion parallax so that it could enhance comfortable 3D perception. The quadtree-based disparity estimation is one of the most remarkable methods for synthesis of intermediate sequences due to the simplicity of its algorithm and hardware implementation. In this paper, we propose two ideas in order to reduce the annoying flicker at the object boundaries of synthesized intermediate sequences by quadtree-based disparity estimation. First, new split-scheme provides more consistent quadtree-splitting during the disparity estimation. Secondly, adaptive temporal smoothing using correlation between present frame and previous one relieves error of disparity estimation. Two proposed ideas are tested by using several stereoscopic sequences, and the annoying flickering is remarkably reduced by them.

Keywords : Flicker, intermediate view, multi-view, quadtree, stereoscopic

I. 서론

인간의 시각인식시스템은 좌안 및 우안으로 얻은 영상의

미세한 차이를 뇌에서 처리하여 깊이정보를 추출함으로써 입체감을 느낄 수 있다. 두개의 시점으로부터의 영상정보만을 이용하는 스테레오방식의 영상시스템은 오직 관찰자에게 깊이 정보만이 부여되므로 특정 시점에서의 입체영상만을 볼 수 있을 뿐이다. 따라서 보다 향상된 3차원 영상을 보기 위해서는 록어라운드(look-around) 기능 및 연속된 운동시차(continuous motion parallax)가 부여되어야 한다^[1].

다시점 3차원 영상시스템은 관찰자에게 많은 시점(view

a) 연세대학교 전기전자공학과
Department of Electrical and Electronic Engineering, Yonsei University
b) 삼성전자(주) DM총괄
Digital Media Business, Samsung Electronics Co.

※ 본 연구는 삼성전자(주) DM연구소의 연구비지원에 의해 실시되었으며, 3D 디스플레이 개발팀의 적극적인 지원에 감사드립니다.

points) 제공하여 보다 고품질의 현실감과 편안한 관측이 가능하도록 하지만, 이를 위해서는 너무 많은 영상 데이터를 필요로 한다. 따라서 영상 데이터의 큰 증가 없이 많은 시점을 생성하는 방법으로 중간영상합성(intermediate view synthesis)이 주로 이용되고 있다. 품질 높은 중간영상을 합성하기 위해서 정밀한 변이추정(disparity estimation)이 요구되는데, 중간영상합성은 좌우 영상으로부터 얻어진 변이값(disparity value)를 이용하기 때문이다.

지난 몇 년간 정확한 변이추정을 위한 여러 가지 접근방법이 제시되었다. 물체기반(object-based)의 변이추정방식은 컴퓨터 합성영상(computer-generated image)과 같은 특정한 스테레오영상에 적합한 방식이지만, 임의의 스테레오영상에 대해 적용할 경우에는 복잡한 알고리즘 때문에 하드웨어 구현이 거의 불가능한 상황이다. 영역기반(block-based)의 변이추정방식은 효율성에 있어서는 물체기반 방식에 비해 떨어지지만, 3D-TV와 같은 임의의 스테레오 영상에 아주 적합한 방식이다^[2].

고정블럭크기(fixed block-size)의 영역기반 방식은 하드웨어의 구현이 매우 쉽지만, 변이추정의 경직성(rigidity)이 있어 이를 이용하여 합성된 중간영상의 물체간 경계에서 인공적인 결함(artifact)을 증가시킨다^{[2][3]}. 반면에 변동블럭크기(variable block-size)의 영역기반 방식은 보다 품질

높은 중간영상을 얻을 수 있지만, 임의의 영상에 대한 명확한 분할(segmentation)이 어렵기 때문에 현수준에서 구현이 거의 불가능하다. 최근 적절한 수준의 중간영상과 하드웨어 구현을 고려한 쿼드트리기반(quadtree-based)방식의 변이추정이 제안되어 연구되고 있다. 하지만 쿼드트리기반 변이추정방식으로 얻은 중간영상 프레임(frame)에서 영상내 물체간 경계(object-boundary)에서 심각한 플리커링(flickering)을 발생시키는데, 이는 부적절한 분할방법(split-scheme) 및 잘못된 변이추정에 원인이 있다^[3].

본 논문에서는 쿼드트리기반의 변이추정 방법을 이용한 중간영상합성에 있어 새로운 분할방법(new split-scheme) 및 적응적 시간평활화(adaptive temporal smoothing)를 제안하여 플리커링이 획기적으로 감소된 높은 품질의 중간영상 프레임을 얻을 수 있었다.

II. 중간영상의 합성

본 논문에서는 쿼드트리기반의 변이추정에 의해 합성된 중간영상 프레임의 화면에서 물체간 경계의 플리커링을 줄이기 위하여 새로운 분할방법 및 적응적 시간평활화를 제시한다. 중간영상합성을 위한 시스템을 그림 1에 나타내었

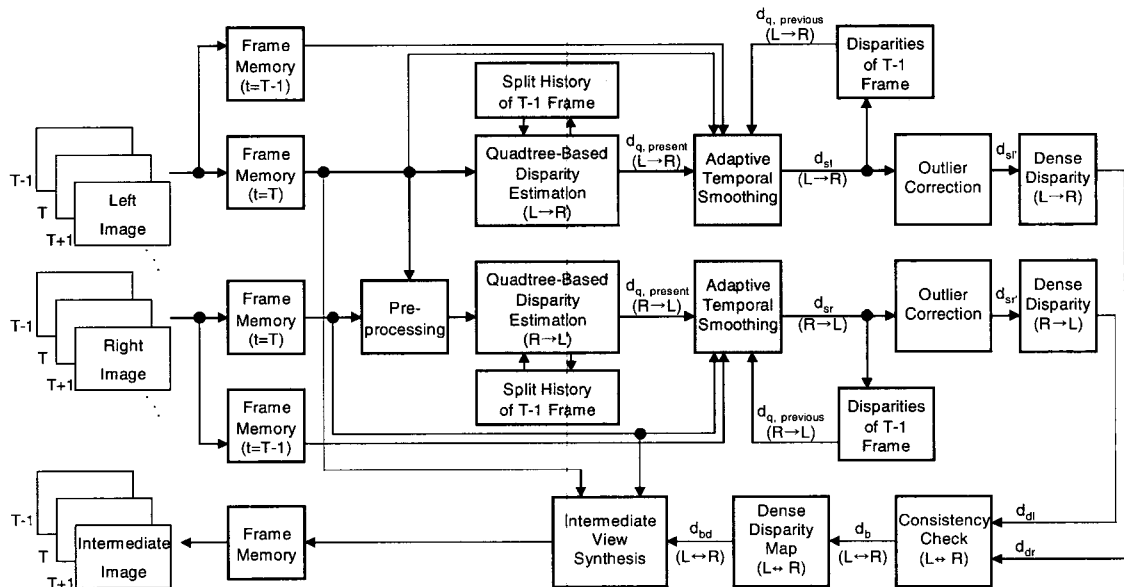


그림 1. 제안된 아이디어를 이용한 중간영상합성의 시스템도

Fig. 1. System diagram for intermediate view synthesis with proposed two ideas, new split-scheme and adaptive temporal smoothing

으며, 처리과정은 ①전처리(preprocessing), ②쿼드트리기반의 변이추정을 위한 블록매칭, ③불합리한 블록의 분할방법, ④적응적 시간평활화, ⑤양방향 미세 변이맵(dense disparity map)의 생성 그리고 ⑥중간 영상의 합성등의 6단계로 설명될 수 있다.

1. 전처리

스테레오방식(stereoscopic)의 비디오는 일반적으로 두 대의 분리된 카메라를 이용하여 촬영되는데, 두 대 카메라의 특성 차이에 의해 좌우 영상의 휘도 및 색상의 불일치가 발생할 수 있다. 이 불일치는 변이추정 시 매칭오차(matching error)를 초래하여 합성된 중간영상의 품질을 떨어뜨리게 된다. 따라서 우영상(right view)의 휘도 및 색상을 좌영상(left view)을 기준으로 보정함으로써 매칭오차를 줄일 수 있다^[4]. 본 논문에서는 MPEG-2에서 제안된 간단한 전처리 알고리즘을 사용하여 변이추정시 매칭오차를 줄이고자 하였다. $I_L(x,y)$ 이 좌영상 (x,y) 좌표에서의 휘도 및 색상을 대표하고, $I_R(x,y)$ 이 우영상 (x,y) 좌표에서의 휘도 및 색상을 대표할 때, 보정된 우영상인 $I'R(x,y)$ 은 식(1)과 같이 표현된다.

$$I'_R(x,y) = \sigma_L / \sigma_R \cdot (I_R(x,y) - \mu_R) + \mu_L \quad (1)$$

여기서 σ 와 μ 는 각각 분산 및 평균이다.

2. 쿼드트리기반의 변이추정

3차원 영상시스템에 이용할 품질 높은 중간영상을 합성하기 위해서는 좌우 영상간 정확한 변이추정이 요구된다. 좌우 영상간 변이정보는 좌우 영상의 상관성을 추정하는 블록매칭 과정에 의해 얻어진다. 3차원 영상시스템의 카메라 구조는 주로 수평으로 위치하므로 수평성분의 변이정보만이 이용하였으며, 수직성분은 무시하였다. 그림2(a)에서와 같이 두 물체가 서로 다른 깊이(depth planes)에 위치하고, 변이추정되는 블록이 두 물체의 경계에 위치한다면, 그 블록 내 모든 화소는 하나의 변이값으로 대표될 수 없게 된다. 그 때 그 블록은 4개의 서브블록(sub-blocks)으로 분할되어야 하며, 분할 후 각각의 서브블록은 다시 변이추정을 실시하게 된다. 이러한 알고리즘을 '쿼드트리기반의 변이추정'이라 한다. 블록 내 모든 화소가 하나의 변이값으로

대표될 수 있을 때까지 변이추정되는 블록을 분할함으로써 블록내 변이값의 불합리성이 제거되어 합성된 중간영상의 품질이 개선될 수 있게 된다. 따라서 쿼드트리기반의 변이추정은 일종의 가변블록 방식의 변이추정으로서, 중간합성을 위한 주요 알고리즘으로 연구되고 있다.

본 논문에서는 3단계 블록크기를 갖는 쿼드트리기반의 변이추정을 사용하고자 한다. 여기서 LB(large block)는 16×16 화소를 갖는 블록으로 정의하며, MB(medium block)와 SB(small block)는 각각 8×8 화소 및 4×4 화소를 갖는 블록으로 정의한다. 그림2(b)에 3단계 블록크기에 대한 정의를 도시하였다.

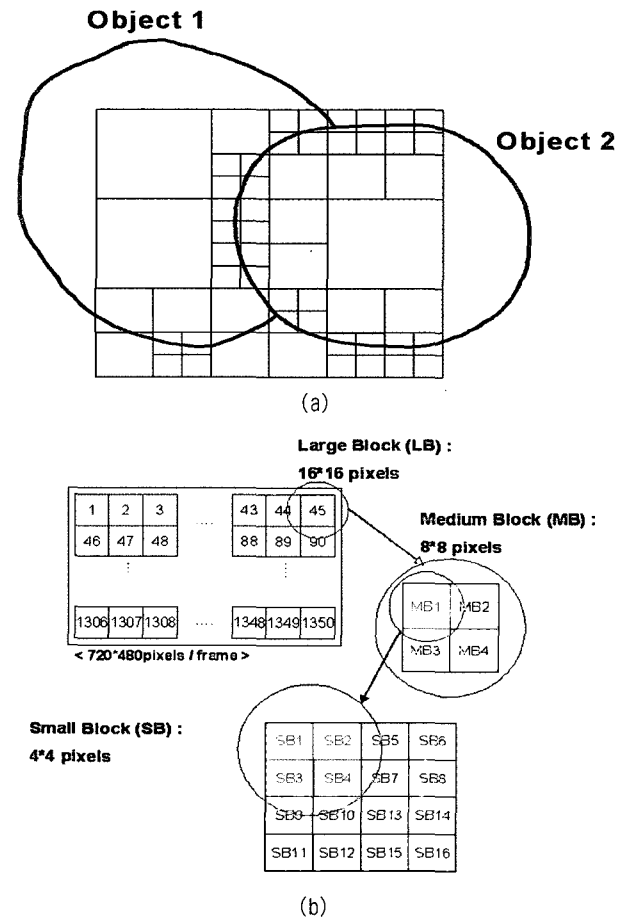


그림 2. (a) 물체간 경계에서의 쿼드트리 분할 (블록들은 경계에서 4개의 서브블록으로 분할됨) (b) 쿼드트리 블록의 정의, LB, MB 그리고 SB
 Fig. 2. (a) The quadtree splitting at object boundaries. The blocks are split into 4 sub-blocks at the boundaries of objects. (b) The definitions of quadtree blocks, LB, MB, and SB.

일반적으로 블록매칭을 실시할 때, 탐색범위 내에서 가장 일치하는 곳을 선택하기 위해 평균절대차이(mean absolute difference, MAD)를 이용한다. LB에 관한 블록매칭 및 불균일성을 나타내는 관계식을 식(2)~(5)에 나타내었다. LB에 관한 블록매칭은 식(2)로 표현할 수 있다. 하지만, 변이추정의 불균일성(inhomogeneity)을 동시에 파악하기 위하여 식(3)과 같이 4개의 서브블록의 블록매칭을 실시하고, 식(4)를 이용하여 그 LB에 대한 블록매칭 결과를 얻을 수 있다. LB의 4개 서브블록중 최대 MAD를 나타내는 것과 최소 MAD를 나타내는 것의 비율, ρ_L 을 식(5)와 같이 정의하며, 변이추정의 불균일성을 나타내는 척도로 이용한다. 식(3)~(5)에서 m 과 n 은 모두 $\{0, 1\}$ 이다.

$$MAD_{LB}[d_k] = \frac{1}{256} \sum_{i=0}^{15} \sum_{j=0}^{15} | I_L(x+j, y+i) - I_R(x+j+d_k, y+i) | \quad (2)$$

$$MAD_{MB(m,n)}[d_k] = \frac{1}{64} \sum_{i=8 \cdot m}^{8 \cdot (1+m) - 1} \sum_{j=8 \cdot n}^{8 \cdot (1+n) - 1} | I_L(x+j, y+i) - I_R(x+j+d_k, y+i) | \quad (3)$$

$$MAD_{LB}[d_k] = \frac{1}{4} \sum_{m=0}^1 \sum_{n=0}^1 MAD_{MB(m,n)}[d_k] \quad (4)$$

$$\rho_L = \frac{\max(MAD_{MB(0,0)}[d_k], MAD_{MB(0,1)}[d_k], \dots, MAD_{MB(1,1)}[d_k])}{\min(MAD_{MB(0,0)}[d_k], MAD_{MB(0,1)}[d_k], \dots, MAD_{MB(1,1)}[d_k])} \quad (5)$$

LB에 대한 변이추정의 불균일성이 커서 그 블록이 분할되어 각각의 MB에 대해 다시 변이추정을 실시해야 하는 경우, 이에 대한 관계식을 식(6)~(9)에 나타내었다. MB에 관한 블록매칭은 식(6)으로 표현할 수 있다. 하지만, 변이추정의 불균일성(inhomogeneity)을 동시에 파악하기 위하여 식(7)과 같이 4개의 서브블록의 블록매칭을 실시하고, 식(8)을 이용하여 그 MB에 대한 블록매칭 결과를 얻을 수 있다. MB의 4개 서브블록중 최대 MAD를 나타내는 것과 최소 MAD를 나타내는 것의 비율, ρ_M 을 식(9)와 같이 정의하며, 변이추정의 불균일성을 나타내는 척도로 이용한다. 식(7)~(9)에서 p 과 q 은 모두 $\{0, 1\}$ 이다.

$$MAD_{MB}[d_k] = \frac{1}{64} \sum_{i=0}^7 \sum_{j=0}^7 | I_L(x+j, y+i) - I_R(x+j+d_k, y+i) | \quad (6)$$

$$MAD_{SB(p,q)}[d_k] = \frac{1}{16} \sum_{i=4 \cdot p}^{4 \cdot (1+p) - 1} \sum_{j=4 \cdot q}^{4 \cdot (1+q) - 1} | I_L(x+j, y+i) - I_R(x+j+d_k, y+i) | \quad (7)$$

$$MAD_{MB}[d_k] = \frac{1}{4} \sum_{p=0}^1 \sum_{q=0}^1 MAD_{SB(p,q)}[d_k] \quad (8)$$

$$\rho_M = \frac{\max(MAD_{SB(0,0)}[d_k], MAD_{SB(0,1)}[d_k], \dots, MAD_{SB(1,1)}[d_k])}{\min(MAD_{SB(0,0)}[d_k], MAD_{SB(0,1)}[d_k], \dots, MAD_{SB(1,1)}[d_k])} \quad (9)$$

MB에 대한 변이추정의 불균일성이 커서 그 블록이 분할되어 각각의 SB에 대해 다시 변이추정을 실시해야 하는 경우, 각각의 SB는 더 이상 분할되지 않으므로 식(10)과 같이 블록매칭을 실시한다.

$$MAD_{SB}[d_k] = \frac{1}{16} \sum_{i=0}^3 \sum_{j=0}^3 | I_L(x+j, y+i) - I_R(x+j+d_k, y+i) | \quad (10)$$

식(2)~(10)에서, I_L 은 좌영상의 휘도 및 색상 정보, I_R 은 보정된 우영상의 휘도 및 색상 정보 그리고 dk 는 탐색범위내 변이값의 후보가 된다. 최종적으로 쿼드트리기반의 블록매칭에 의한 변이값, dq 는 식(11)과 같이 탐색범위내에서 MAD를 최소로 하는 점을 찾음으로써 결정된다.

$$dq = \arg \min_{d_k} (MAD [d_k]) \quad (11)$$

LB의 변이추정 결과, 불균일성이 작아서 하나의 변이값이 그 LB를 대표할 수 있을 경우에는 LB의 매칭 결과는 식(4) 및 식(11)을 이용해 결정한다. 유사하게, MB의 변이추정 결과, 불균일성이 작아서 하나의 변이값이 그 MB를 대표할 수 있을 경우에는 MB의 매칭 결과는 식(8) 및 식(11)을 이용해 결정하며, SB의 매칭 결과는 식(10) 및 식(11)을 이용해 결정한다.

3. 문제의 블록(problematic block)의 분할방법

쿼드트리 기반 변이추정으로 고품질의 중간영상을 합성하기 위하여 엄격한 분할방법을 사용하는 것이 필수적이

다. 물체의 경계를 포함하는 블록에서 그 블록내 변이추정의 불균일성(inhomogeneity)을 식(5) 및 식(9)를 이용하여 판정한다. 만일 ρ_L 및 ρ_M 이 정해진 임계값을 넘는다 면 그 블록은 하나의 단일한 변이값으로 대표할 수 없는

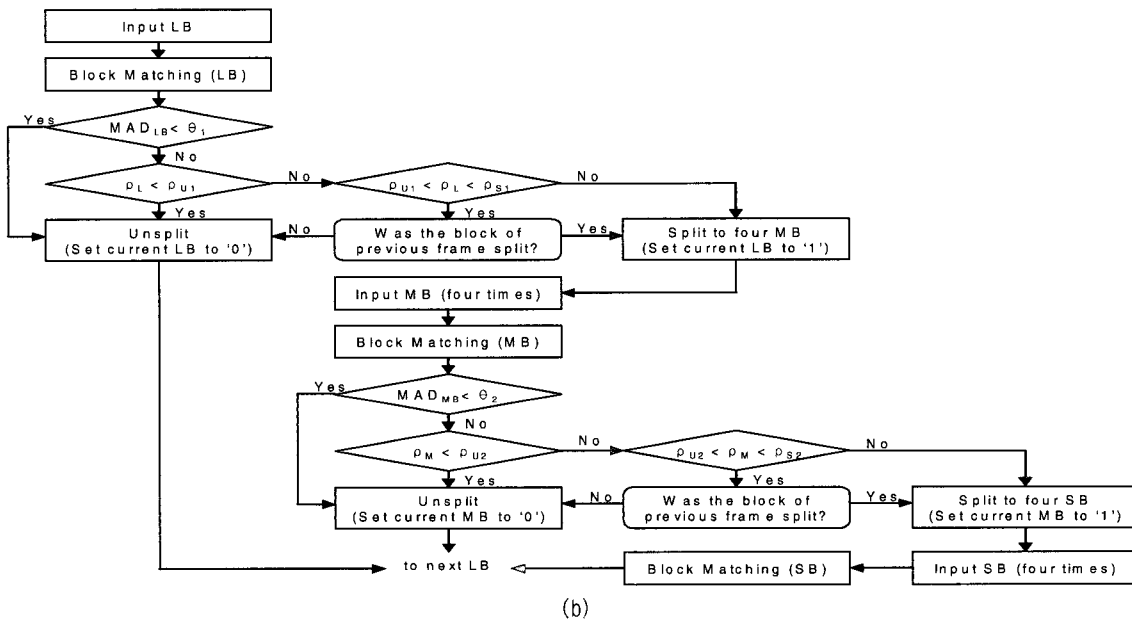
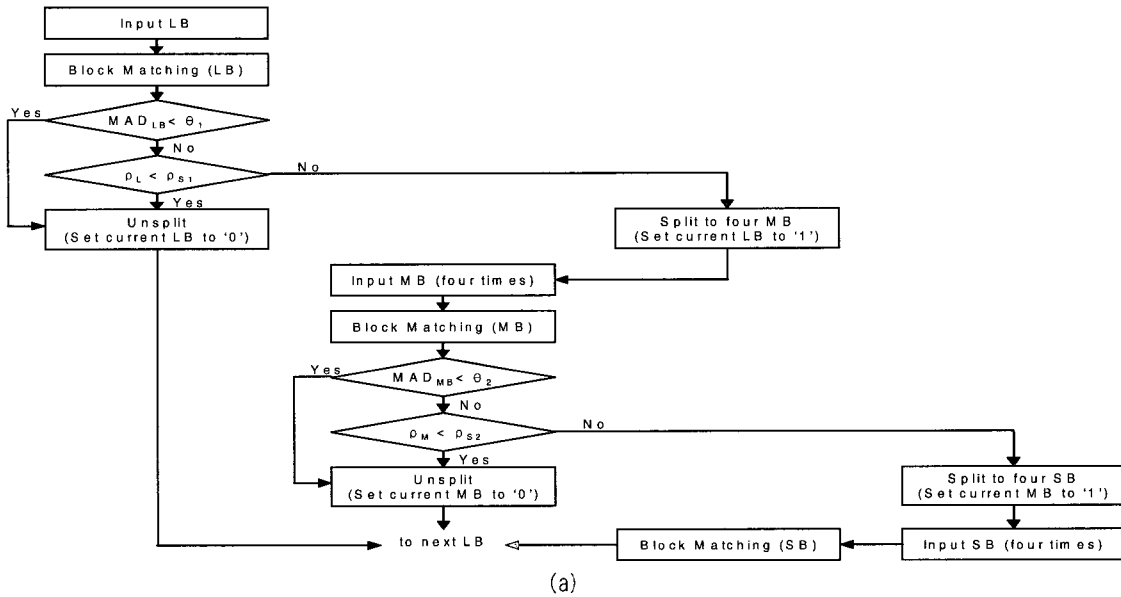


그림 3. 쿼드트리 분할의 흐름도 (a) 단일임계값을 사용하는 일반적인 분할방법, (b) 이중임계값을 사용하는 제안된 새로운 분할방법
 Fig. 3. Flow-charts for quadtree splitting. (a) General split-scheme with single-threshold, (b) Proposed new split-scheme with dual-threshold

문제의 블록으로 간주되며, 4개의 블록으로 분할되어 다시 매칭과정을 실시해야 한다. 일반적으로 단일 임계값을 갖는 분할방법을 그림3(a)에 나타내었으며, 그 과정은 다음과 같다.

LB에 대해,

- Step1: 식(4)의 최소 MADLB가 임계값(θ_1) 보다 작으면 분할하지 않으며, 다음 LB를 검사한다. 그 외에는 Step2로 간다. (이는 어떤 블록의 최소 MAD가 아주 작으면 그 블록을 하나의 변이값으로 대표하는데 무리가 없다는 의미이다.)
- Step2: 식(5)의 p_L 이 임계값($pS1$) 보다 작으면 그 블록은 분할하지 않으며, 다음 LB를 검사한다. p_L 이 임계값($pS1$) 보다 크면 그 LB는 4개의 MB로 분할하고 Step3으로 간다. (이는 블록내 변이추정의 불균일성이 커서 하나의 변이값으로 대표할 수 없다면, 그 블록은 4개의 서브블록으로 분할하여 각각 다시 매칭과정을 수행함으로써 별도의 변이값을 찾아야 함을 의미한다.)

각각의 MB(MB1, MB2, MB3 그리고 MB4)에 대해,

- Step3: 식(8)의 최소 MADMB가 임계값(θ_2) 보다 작으면 분할하지 않는다. 그 외에는 Step4로 간다.
- Step4: 식(9)의 p_M 이 임계값($pS2$)보다 작으면 그 블록은 분할하지 않는다. p_M 이 임계값($pS2$)보다 크면 그 MB는 4개의 SB로 분할한 후, 각각의 SB에 대해 다시 매칭과정을 수행한다.

하지만 몇몇 스테레오 비디오를 이용하여 합성한 중간영상의 물체의 경계부에서 많은 플리커링이 발생됨이 보고되고 있다^[3]. 정지 동영상(static video)의 경우, 매초당 최소 45프레임 동안 동일한 영상을 디스플레이하여야 하는데, 동일한 영상이 매초당 45프레임에 미치지 못하거나, 동일한 영상을 디스플레이하지 못할 경우에 관찰자는 정지된 물체에서 흔들림을 느끼며 이를 플리커링이라 규정한다. 중간영상합성의 경우, 정지된 물체는 그 시간동안 매 프레임에서 일관된 영상을 유지해야 하는데, 잘못된 중간영상합성에 의해 왜곡된 프레임이 삽입되면, 관찰자가 그 정지된 물체를 동영상으로 볼때 흔들림을 느끼게 되며, 이런 현상을 플리커링으로 정의한다.

플리커링이 발생하는 이유로 부적절한 쿼드트리 분할 방법과 부정확한 변이추정을 들 수 있다. 실제로 스테레오 카메라의 미세한 떨림이나 조명의 시간적 변화에 의해 연속되는 프레임 간에 영상정보 면에서의 미세한 차이가 발생할 수 있다. 이러한 영상정보의 미세한 변

화는 쿼드트리 분할방법 및 변이추정에 영향을 주어 특히 물체간 경계부에서 심한 플리커링을 야기할 수 있게 된다. 그러므로 본 논문에서는 이중 임계값을 사용하는 새로운 쿼드트리 분할방법을 제안하여 합성된 중간영상의 물체간 경계부에 발생하기 쉬운 플리커링을 줄이고자 하였다. 제안된 새로운 분할방법을 그림3(b)에 나타내었다.

LB에 대해,

- Step1: 식(4)에서 최소 MADLB $\leq\theta_1$ 이면 분할하지 않으며, 다음 LB를 검사한다. 그 외에는 Step2로 간다.
- Step2: 식(5)에서 $p_L \leq pU1$ 이면 분할하지 않으며, 다음 LB를 검사한다. 이 외에는 Step3으로 간다.
- Step3: 식(5)에서 $p_L \geq pS1$ 이면 그 LB를 4개의 MB로 분할하고, Step5로 간다. 이 외에는 Step4로 간다.
- Step4: 식(5)에서 $pS1 < p_L < pU1$ 이고 이전 프레임의 해당 블록이 분할되지 않았다면 분할하지 않고, 다음 LB를 검사한다. 그 외에는 그 LB를 4개의 MB로 분할하고 Step5로 간다.

각각의 MB(MB1, MB2, MB3 그리고 MB4)에 대해,

- Step5: 식(8)에서 최소 MADMB $\leq\theta_2$ 이면 분할하지 않는다. 그 외에는 Step6으로 간다.
- Step6: 식(9)에서 $p_M \leq pU2$ 이면 분할하지 않는다. 이 외에는 Step7로 간다.
- Step7: 식(9)에서 $p_M \geq pS2$ 이면 그 MB를 4개의 SB로 분할한다. 이 외에는 Step8로 간다.
- Step8: 식(9)에서 $pS2 < p_M < pU2$ 이고 이전 프레임의 해당 블록이 분할되지 않았다면 분할하지 않는다. 그 외에는 그 MB를 4개의 SB로 분할한다.

본 논문에서 제안된 새로운 분할방법에 의하면, $pS1 < p_L < pU1$ 또는 $pS2 < p_M < pU2$ 일때 그 블록의 분할은 이전 프레임의 해당 블록이 분할되었었는지의 여부에 따라 결정된다. 따라서 따라서 이전 프레임의 분할정보(split history)를 참조하기 위해서는 그림3(b)의 해상도를 갖는 스테레오 영상신호에 대해 5비트 \times 1350(LBs) \times 2(L \rightarrow R, R \rightarrow L)의 메모리 용량이 요구된다. 각 LB에 대해 LB 그리고 4개의 MB의 분할 여부를 기록하기 위해서 5비트가 필요하기 때문이다. 예를들어 어떤 LB의 이전 프레임의 분할정보가 '10100'이라면 그 LB가 4개의 MB로 분할되었으며, 4개의 MB중에서 MB2는 다시 4개의 SB로 분할되었음을 의미한다. 새로운 분할방법을 적용한 쿼드트리기반 변이추정에 대한 하드웨어 구조를 그

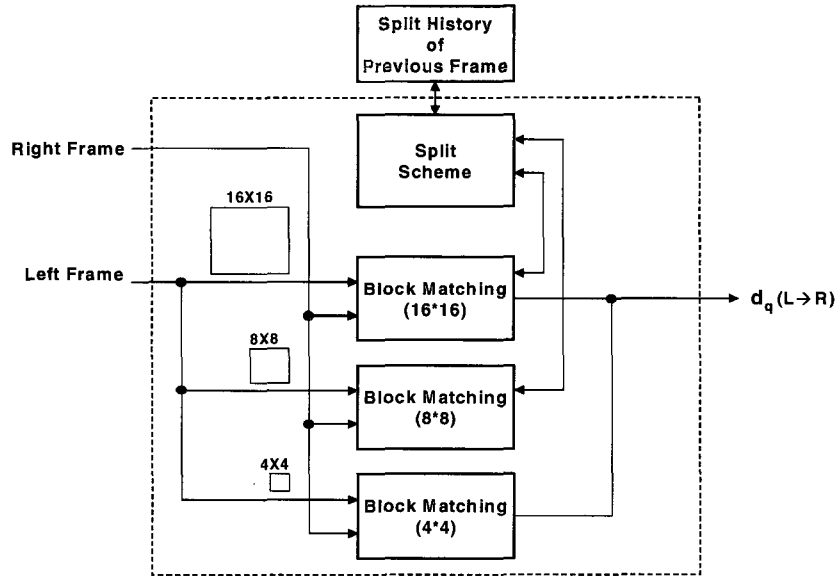


그림 4. 새로운 분할방법을 사용한 쿼드트리 변이추정부의 하드웨어 구조 (일반적인 분할방법을 이용한 쿼드트리 변이추정부를 점선안에 나타냄)
 Fig. 4. Hardware structure of quadtree-based disparity estimation with new split-scheme. Quadtree-based disparity estimation with general split-scheme is shown in the dashed block

림 4에 나타내었다.

4. 적응적 시간평활화

합성된 중간영상에서 플리커링이 발생하는 주된 이유는 부적절한 쿼드트리 분할방법 및 부정확한 변이추정이다. 앞에서 제시한 새로운 쿼드트리 분할방법은 플리커링을 줄이는데 효과적이었지만, 부정확한 변이추정에 의해 발생하는 플리커링을 줄이는데 적합하지 않다. 따라서 부정확한 변이추정에 기인하는 플리커링을 줄이기 위해 추정된 변이값에 '적응적 시간평활화'를 적용을 제안한다. 이는 스테레오 영상의 연속된 2개 프레임간의 시간적인 불일치성(dissimilarity)을 이용하여 추정된 변이값을 적응적으로 평활화하므로써 부정확한 변이추정을 줄이고자 함이다. 그림 5(a)에 적응적 시간평활화에 대한 블록도를 나타내었다.

각 블록(LB)의 불일치계수(dissimilarity coefficient), ϵ 는 식(12)를 이용하는 불일치평가부(dissimilarity estimator)로부터 구할 수 있다. 여기서 ϵ 는 8비트로 양자화된 디지털 신호이며, 최대 255의 값을 갖는다. 시간평활화계수(temporal smoothing coefficient), η 는 그림5(b)의 그래프

를 이용하여 결정되는데, 여기서 η_{max} , η_{min} , ϵ_{low} 그리고 ϵ_{high} 는 모두 영상의 특성에 따라 사용자가 설정해 줄 수 있는 값이다. 그림5(b)로부터 현재 프레임의 해당 블록과 이전 프레임의 해당 블록의 불일치성(dissimilarity)이 작을 경우($\epsilon < \epsilon_{low}$), 시간평활화계수는 커져 η_{max} 가 된다. 반면, 현재 프레임의 해당 블록과 이전 프레임의 해당 블록의 불일치성이 클 경우($\epsilon > \epsilon_{high}$), 시간평활화계수는 작아져 η_{min} 이 된다. $\epsilon_{low} \leq \epsilon \leq \epsilon_{high}$ 일 경우에는 시간평활화계수는 불일치계수에 선형적으로 역비례한다. 하드웨어 구현상 편의성을 위해 함수는 그림5(b)의 점선과 같이 양자화 될 수 있다.

$$\epsilon = \frac{1}{256} \sum_{i=0}^{15} \sum_{j=0}^{15} |I^{(t)}(x+j, y+i) - I^{(t-1)}(x+j, y+i)| \quad (12)$$

각 블록에 대한 시간평활화계수와 식(13)을 이용하여 각 블록에 대한 평활화된 변이값(smoothed disparity value) ds 를 얻을 수 있게 된다. 그림 5(a)에 나타낸 바와 같이 평활화된 변이값 ds 는 다음 프레임을 위해 메모리에 저장되어야 한다.

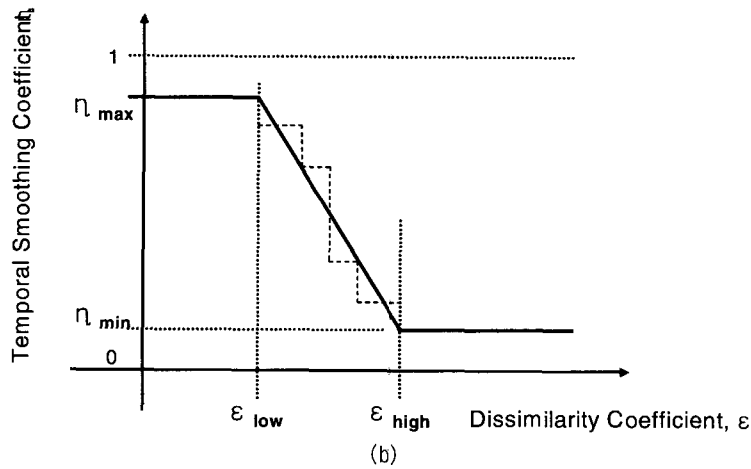
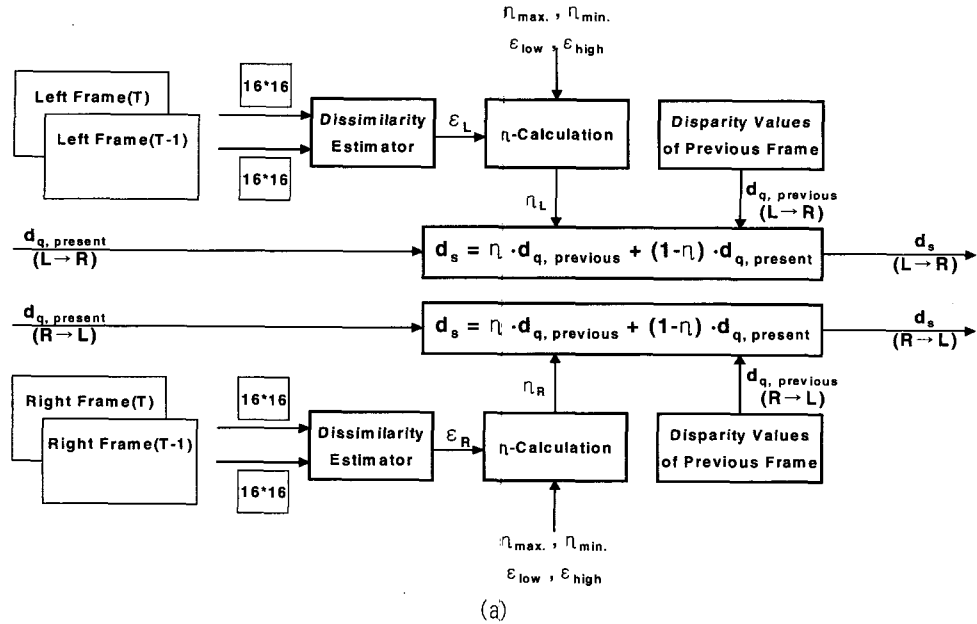


그림 5. 적응적 시간평활화 (a) 블록도, (b) 상관계수와 시간평활화계수와의 관계
 Fig. 5. Adaptive temporal smoothing (a) Block diagram, (b) The relationship between dissimilarity coefficient and temporal smoothing coefficient

$$d_s = \eta \cdot d_{q,previous} + (1-\eta) \cdot d_{q,present} \quad (13)$$

5. 양방향 미세 변이맵(bidirectional dense disparity map)의 생성

양방향 미세 변이맵을 얻기 위한 과정을 그림6에 나타내었다. 쿼드트리기반의 변이추정 및 적응적 시간평활화에 의해 매 4x4 블록 마다 얻은 변이값을 그림6(a) 및 그림6(b)

에 나타내었다. 그림(6)에서 각각의 열은 좌영상 및 상대되는 우영상의 한줄을 각각 나타낸다. 그림6(a)에 좌에서 우로의 변이값(dsl)을 매 4화소마다 화살표로 나타내었으며, 그림6(b)에 위에서 좌로의 변이값(dsr)을 매 4화소마다 화살표로 나타내었다. 스테레오영상으로부터 추정된 모든 변이값은 식(14)를 만족해야 한다. 만일 식(14)를 만족하지 못할 경우 그 변이값은 잘못 추정된 아웃라이어(outlier)로 간주되며, 미세변이맵을 생성하기 전에 반드시

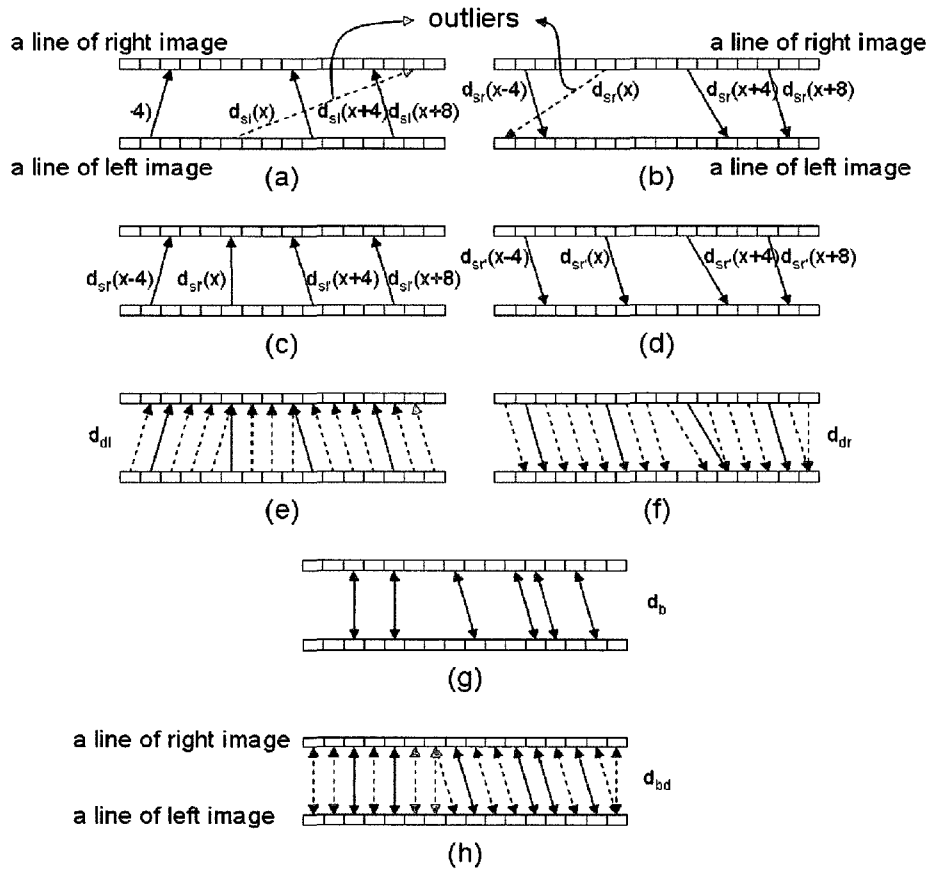


그림 6. 양방향 미세 변이맵을 구성하기 위한 과정 (a) 좌-우 거친 변이맵 dsl, (b) 우-좌 거친 변이맵 dsr, (c) 아웃라이어가 제거된 좌-우 거친 변이맵 dsl', (d) 아웃라이어가 제거된 우-좌 거친 변이맵 dsr', (e) 좌-우 미세 변이맵 ddl, (f) 우-좌 미세 변이맵 ddr, (g) 일관성 검사후 양방향 변이맵 db 그리고 (h) 양방향 미세 변이맵 dbd

Fig. 6. Processes for bidirectional dense disparity map. (a) Left-to-right sparse disparity map, (b) Right-to-left sparse disparity map, (c) Outlier corrected left-to-right sparse disparity map, (d) Outlier corrected right-to-left sparse disparity map, (e) Left-to-right dense disparity map, (f) Right-to-left dense disparity map, (g) Bidirectional disparity map after consistency check, and (h) Bidirectional dense disparity map

수정되어야 한다. 그림6(a)와 그림6(b)에 아웃라이어를 점선 화살표로 나타내었으며, 대부분의 아웃라이어가 식 (15)의 메디안필터에 의해 제거될 수 있다. 다음, 그림6(e) 및 그림6(f)와 같이 선형보간(linear interpolation)에 의해 단방향 미세 변이맵(ddl 및 ddr)을 얻게 된다^{[5][6]}. 일치성 검사(consistency check) 과정을 통해 L→R 및 R→L의 단방향 미세 변이맵에서 양방향으로 일치하는 변이(db)만을 골라낸다. 본 논문에서는 합성된 중간영상의 화질을 떨어뜨리지 않고, 적당한 수의 양방향으로 일치된 변이를 얻기 위해 일치성 검사의 허용오차를 ±1화소로 설정하였다.

이 과정을 그림6(g)에 나타내었다. 마지막으로 양방향으로 일치하는 변이만을 이용하여 다시 보간을 실시하여 그림6(h)와 같이 양방향(L↔R) 미세 변이맵(dbd)을 얻게 된다.

$$d_s(x - 4) - 4 \leq d_s(x) \leq d_s(x + 4) + 4 \tag{14}$$

$$d_{s'}(x) = \frac{d_s(x - 4) + d_s(x + 4)}{2} \tag{15}$$

6. 중간 영상의 합성

중간영상을 합성하는 과정을 그림7에 나타내었다. 합성은 양방향 변이맵의 변이 d_c 를 이용하여 실행된다. 중간영상은 식(14)를 사용해 좌, 우영상으로부터 보간하며, 좌영상 및 우영상의 오클루션 영역(occlusion region)은 각각 우영상 및 좌영상으로부터 외삽에 의해 복원한다^[7].

$$I[x + \alpha \cdot d_{bd}, y] = (1 - \alpha) \cdot I_L(x, y) + \alpha \cdot I_R[x + d_{bd}, y] \quad (16)$$

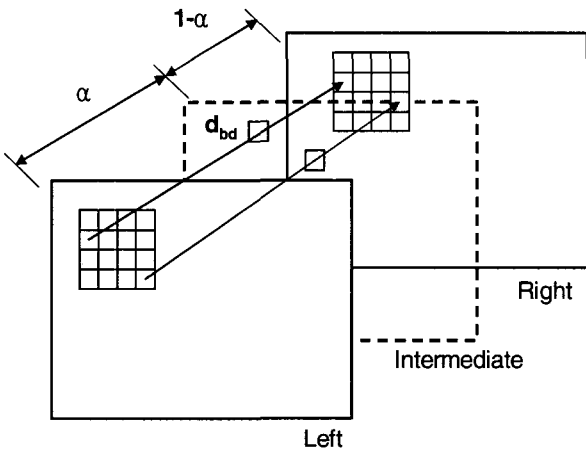


그림 7. 보간 및 외삽에 의한 중간영상의 합성
Fig. 7. Intermediate view synthesis by interpolation or extrapolation

III. 시뮬레이션 결과

제안된 아이디어에 관한 효과를 PUPPY, WORLDCUP-2002 그리고 CROWD 스테레오영상을 이용하여 검증하였다. PUPPY와 WORLDCUP2002는 ETRI(한국, 대전)의 실감방송연구팀에 의해 수렴형(convergence-type) 카메라 구조를 이용해 녹화되었다. 특히, WORLDCUP2002는 2002년 한일월드컵중 스테레오영상으로 방송된 실제 프로그램의 일부이다. 해상도는 모두 720×480화소이며, 좌우 카메라간 거리(baseline)는 각각 115mm(PUPPY)와 250mm(WORLDCUP- 2002)이다. CROWD는 Carnegie Mellon 대학(미국, Pittsburgh)에서 평행형(parallel-type) 카메라 구조를 이용해 녹화되었다. 해상도는 640×240화소, 그리고 좌우 카메라간 거리(baseline)는 50mm이다. 테스트용 스

테레오영상의 특성을 표 1에 요약하였다.

표 1. 테스트용 스테레오영상의 특성
Table 1. Properties of test sequences.

	Puppy (ETRI)	2002worldcup (ETRI)	Crowd (CMU)
No. of test frames	10	10	20
Picture size	720*480	720*480	640*240
Stereo camera type	Convergence	Convergence	Parallel
Camera focus	8.0 mm	7.5 mm	9.5 mm
Baseline of camera	115 mm	250 mm	50 mm
Search range	-8 ~ 60 pixels	-10 ~ 24 pixels	0 ~ 40 pixels
Characteristics of stereoscopic sequence	A moving object with static background	Some moving object with dynamic background	A lot of moving object with nearly static background
Camera motion	None	Abundant	Minimal
Correlation between frames of background area	High	Low	Medium

PUPPY, WORLDCUP2002 그리고 CROWD의 탐색범위는 영상의 크기, 좌우 카메라간 거리 그리고 카메라 초점을 고려하면 수평방향으로 각각 -8~60, -10~24 그리고 0~40화소이다. 변이값의 수직성분은 본 논문에서는 무시하였다.

일반적으로 스테레오영상으로부터 합성된 중간영상의 품질에 대한 평가는 좌우 카메라의 정중앙에서 촬영된 참조중간영상(reference intermediate sequence)과 비교함으로써 얻을 수 있다. 하지만 불행히도 대부분의 스테레오영상은 오직 좌우 2개의 카메라에 의해서 촬영되므로 참조중간영상과의 비교에 의한 정량적인 평가를 할 수 없다. 따라서 본 논문에서는 $\alpha=0.5$ 에서 계산된 편이맵에 근거해 합성된우영상을 참조우영상(reference right sequence, \hat{I}_R)으로 하여 우영상과 비교한다. 합성오차($\hat{I}_R - I_R$)는 PSNR(peak signal to noise ratio)로서 표현되며, 이 결과는 정량적인 해석을 위해 사용될 수 있다. 우영상에는 있으나 좌영상에는 없는 영역(occlusion region)을 제외하면 이와 같은 정량적인 해석 방법은 상당히 정확한 것으로 알려진다^{[8][9]}.

그림 8에 $\rho S1$, $\rho S2$, $\theta 1$ 그리고 $\theta 2$ 에 따른 PSNR을 나타내었다. $\rho S1 = \rho S2 = 1.5$ 그리고 $\theta 1 = \theta 2 = 10$ 일 때, 27.174[dB]

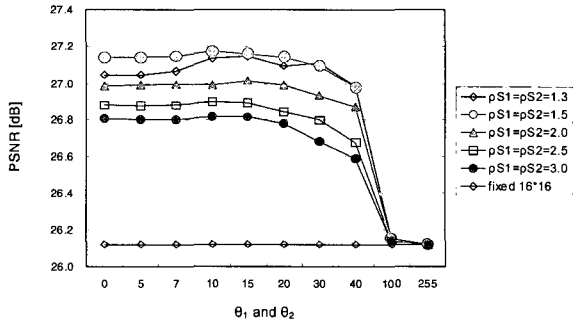


그림 8. $pS1$, $pS2$, θ_1 그리고 θ_2 에 따른 PSNR.
Fig. 8. The PSNR as functions of $pS1$, $pS2$, θ_1 and θ_2 .

로 가장 높았다. PSNR 및 시각적 평가 결과, 파라미터로서 $pS1=pS2=1.5$ 그리고 $\theta_1=\theta_2=10$ 이 본 연구에 사용되었다. 표 2에 $\theta_1=\theta_2=10$ 일 때, $pS1$ 와 $pS2$ 에 따른 PSNR과 연산량을 나타내었다. 고정 16X16 블록 방식에 비해 연산량이 14.7% 증가되었다.

표 3에 $pS1=pS2=1.5$ 일 때의 $pU1$ 와 $pU2$ 에 대한 PSNR 및 연산량을 나타내었다. $pU1=pU2=1.3$ 일 때, 가장 높은 PSNR을 나타내며, 연산량은 일반적인 분할방법(단일임계값)을 사용하는 일반적 쿼드트리기반 방식의 연산량에 비해 약 1.6% 증가하였다. PSNR과 연산량을 고려하여 본 논문에서는 $pU1=pU2=1.3$ 을 적용하였다.

그림 9에 ϵ_{low} 와 ϵ_{high} 의 변화에 따른 PSNR을 나타내었다. $\epsilon_{low}=10$ 그리고 $\epsilon_{high}=20$ 일 때, 가장 높은 PSNR이 얻어졌다. 과평활(over-smoothing)을 방지하기 위하여 불일치계수, ϵ 가 ϵ_{high} (=20)보다 클 때, η_{min} 이 0이 되도록 하였다. 또한 잘못 추정된 변이값이 시간적으로 전파되는

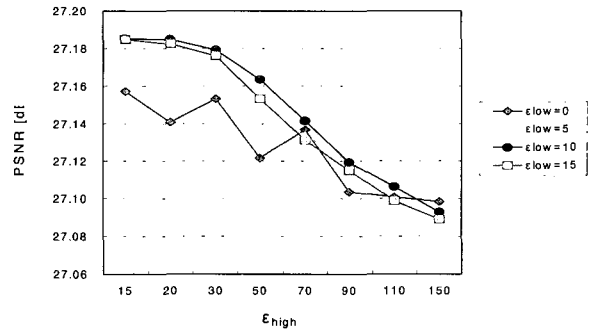


그림 9. ϵ_{low} 그리고 ϵ_{high} 의 변화에 따른 PSNR.
Fig. 9. The PSNR as functions of ϵ_{low} and ϵ_{high} .

것을 완화하기 위해 η_{max} 를 0.9로 제한하였다.

본 논문에서 변이추정을 위해 사용한 파라미터를 표 4에 요약하였으며, 이를 이용하여 적절한 중간영상을 합성할 수 있었다.

표 4. 컴퓨터 시뮬레이션을 위한 파라미터
Table 4. The parameters for computer simulation

Values	Remark	General Split-scheme	New Split-scheme	General Split-scheme + Adaptive Temporal Smoothing	New Split-scheme + Adaptive Temporal Smoothing
θ_1, θ_2	average MAD	10	10	10	10
$pS1, pS2$	ratio	1.5	1.5	1.5	1.5
$pU1, pU2$	ratio	-	1.3	-	1.3
Scaling of disparities	-	Linear	Linear	Linear	Linear
Tolerance of bidirectional consistency check	pixel	± 1	± 1	± 1	± 1
η_{max}	1.0 max.	-	-	0.9	0.9
η_{min}	0.0 min.	-	-	0	0
ϵ_{high}	255 max.	-	-	20	20
ϵ_{low}	0 min.	-	-	10	10

표 2. $pS1$ 와 $pS2$ 에 따른 PSNR 및 연산량 ($\theta_1=\theta_2=10$ 일 때)
Table 2. The PSNR and computational load for the thresholds, $pS1$ and $pS2$ when $\theta_1=\theta_2=10$.

$pS1, pS2$	quadtree					fixed 16x16
	1.3	1.5	2.0	2.5	3.0	∞
PSNR [dB]	27.155	27.174	27.092	26.992	26.928	26.245
Computational Load [%]	118.5	114.7	109.5	106.9	104.9	100.0

표 3. $pU1$ 와 $pU2$ 에 따른 PSNR 및 연산량 ($pS1=pS2=1.5$ 일 때)
Table 3. The PSNR and computational load for the thresholds, $pU1$ and $pU2$ when $pS1=pS2=1.5$.

$pS1, pS2$	1.5			
$pU1, pU2$	1.5	1.4	1.3	1.2
PSNR [dB]	27.174	27.183	27.192	27.191
Computational Load [%]	100	100.7	101.6	102.8

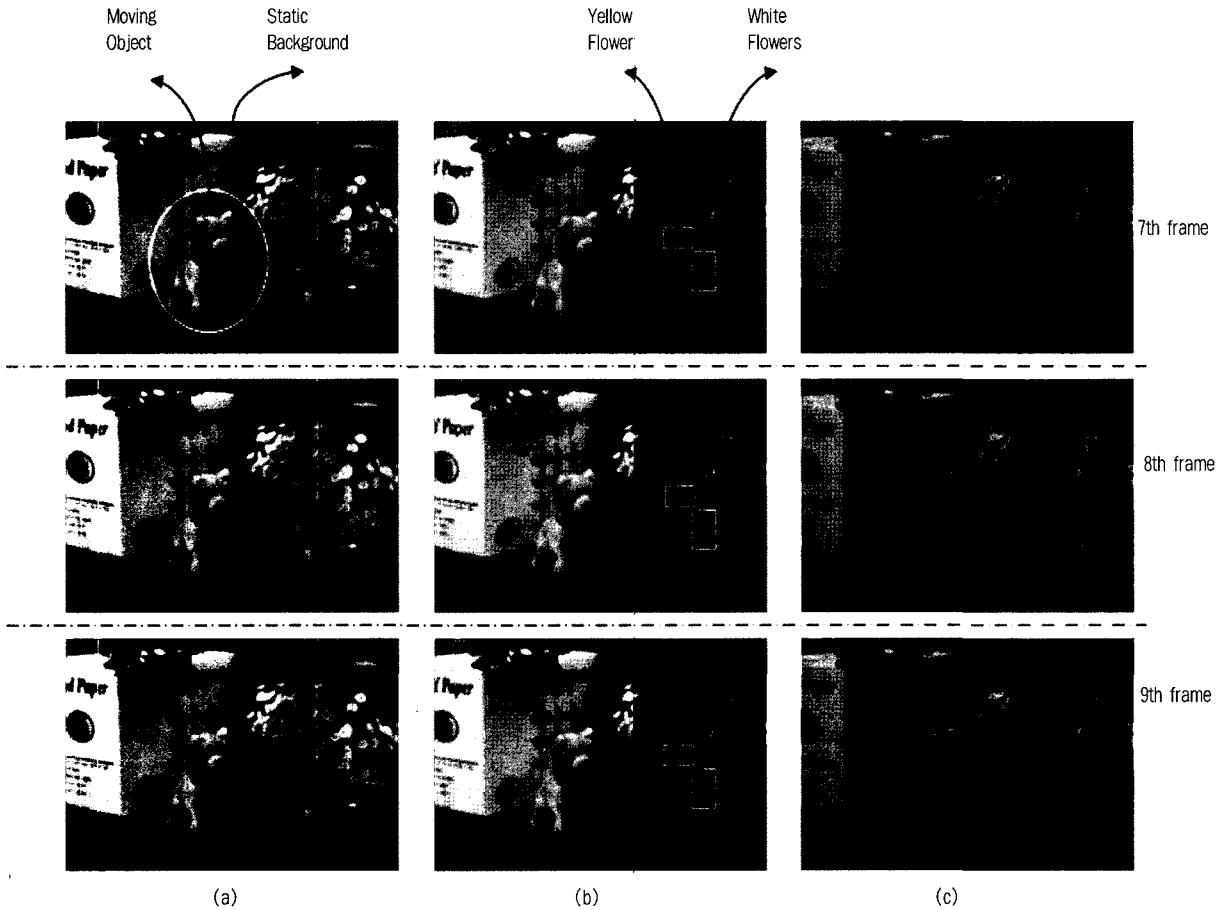


그림 10. PUPPY의 영상신호 (a) 좌영상 신호, (b) 중간영상 신호 그리고 (c) 우영상 신호 (그림에서 정적인 배경을 가지며, 오직 강아지 인형만이 활동적으로 움직임, 합성된 중간영상에서 문제의 플리커링이 발생하여 'Yellow flower'와 'White flowers'로 명명하고 표시함)
 Fig. 10. Image sequences of PUPPY. (a) Left image sequence, (b) Synthesized intermediate sequence, and (c) Right image sequence. In the pictures, only the dog-doll is moving actively with static background. In the synthesized intermediate frames of (b), problematic flickering is observed at the blocks of 'Yellow flower' and 'White flowers'.

PUPPY의 좌우영상 및 합성된 중간영상을 그림10에 나타내었다. 그림10(a)에서 오직 원안에 표시한 강아지인형만이 움직이며, 그 외의 배경은 정적임을 알 수 있다. 정적인 배경(static background)에서의 물체간 경계에서의 플리커링에 관심을 두고 있기 때문에 PUPPY는 본 논문에서 제안하는 아이디어를 검증하기에 매우 적절한 샘플이다. 그림 10(b)에의 네모상자로 표시된 두 곳에서 플리커링이 발견되었다. 이를 각각 'White Flowers'와 'Yellow Flower'로 명명하기로 한다.

그림 11에 식별을 용이하게 하기 위해 확대되고, 콘트라스트를 보강한 'White Flowers'를 나타내었다. 일반적인 분

할방법(단일임계값), 새로운 분할방법(이중임계값), 적응적 시간평활화를 적용한 일반적인 분할방법(단일임계값) 그리고 적응적 시간평활화를 적용한 새로운 분할방법(이중임계값)등의 4가지 방법을 이용하여 합성된 중간영상 프레임들 그림11(a), 그림11(b), 그림11(c) 그리고 그림11(d)에 각각 나타내었다. 그림11(a)의 네모상자 및 원안에 합성된 영상은 프레임에 따라 일관성 있게 동일하지 않다. 이는 동영상으로 볼때, 네모상자 및 원안에서 플리커링이 발생하고 있음을 의미한다. 그림11(c)에서 네모상자 내에서는 플리커링이 관찰되지 않았지만, 원안에서는 여전히 플리커링이 관찰됨을 알 수 있다. 그러나 그림11(b)와 그림11(d)에서는 네

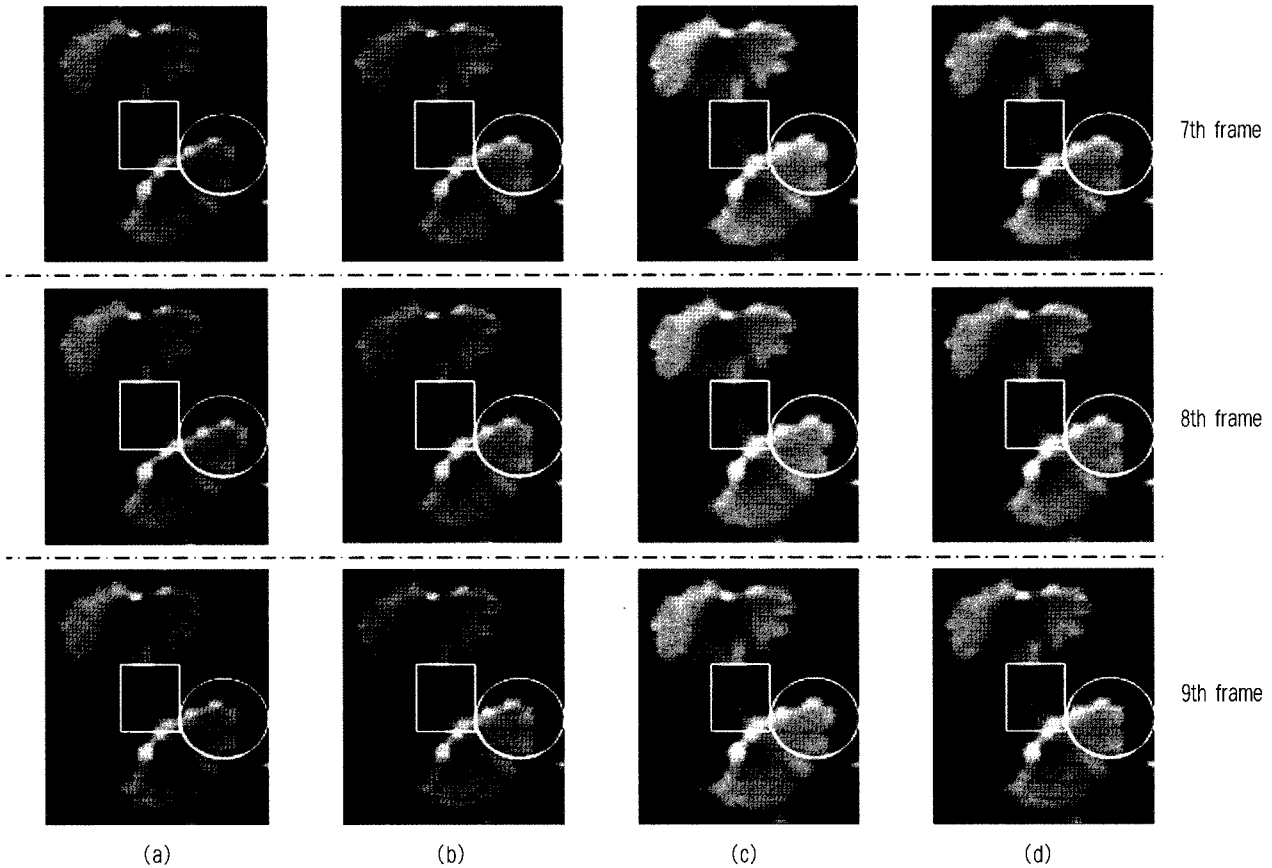


그림 11. 'White flowers'의 합성된 프레임, (a) 일반적인 분할방법에 의한, (b) 새로운 분할방법에 의한, (c) 일반적인 분할방법과 적응적 시간평활화를 적용한 그리고 (d) 새로운 분할방법과 적응적 시간평활화를 적용하여 얻은 합성 영상
 Fig. 11. The synthesized frames of 'White flowers' of Fig. 10(b). (a) By general split-scheme, (b) By new split-scheme, (c) By general split-scheme with adaptive temporal smoothing, and (d) By new split-scheme with adaptive temporal smoothing. The synthesized frames are enlarged and contrast-compensated for the visibility

모상자 및 원안의 합성된 영상들이 프레임간에 거의 동일하여 플리커링이 발생하지 않음을 보여준다. 그림11의 결과로부터 PUPPY의 'White Flowers'에서의 플리커링은 부적절한 쿼드트리 분할에 의해 발생하였으며, 새로운 분할방법을 적용하여 제거되었음을 알 수 있다.

그림 12에 식별을 용이하게 하기 위해 확대되고, 콘트라스트를 보강한 'Yellow Flower'를 나타내었다. 일반적인 분할방법(단일임계값), 새로운 분할방법(이중임계값), 적응적 시간평활화를 적용한 일반적인 분할방법(단일임계값) 그리고 적응적 시간평활화를 적용한 새로운 분할방법(이중임계값) 등의 4가지 방법을 이용하여 합성된 중간영상 프레임들

그림 12(a), 그림 12(b), 그림 12(c) 그리고 그림 12(d)에 각각 나타내었다. 그림12(a)의 네모상자 내에 합성된 영상은 프레임에 따라 일관성 있게 동일하지 않다. 이는 동영상으로 볼때, 네모상자 내에서 플리커링이 발생하고 있음을 의미한다. 그림 12(b)에서 네모상자 내에서는 플리커링이 여전히 관찰되지만, 그림 12(c)와 그림 12(d)에서는 네모상자안에 합성된 영상들이 프레임 마다 거의 동일하여 플리커링이 발생하지 않음을 보여준다. 그림 12의 결과로부터 PUPPY의 'Yellow Flower'에서의 플리커링은 부정확한 변이추정에 의해 발생하였으며, 새로운 분할방법이 아니라 적응적 시간평활화를 적용함으로써 제거되었음을 알 수 있다.

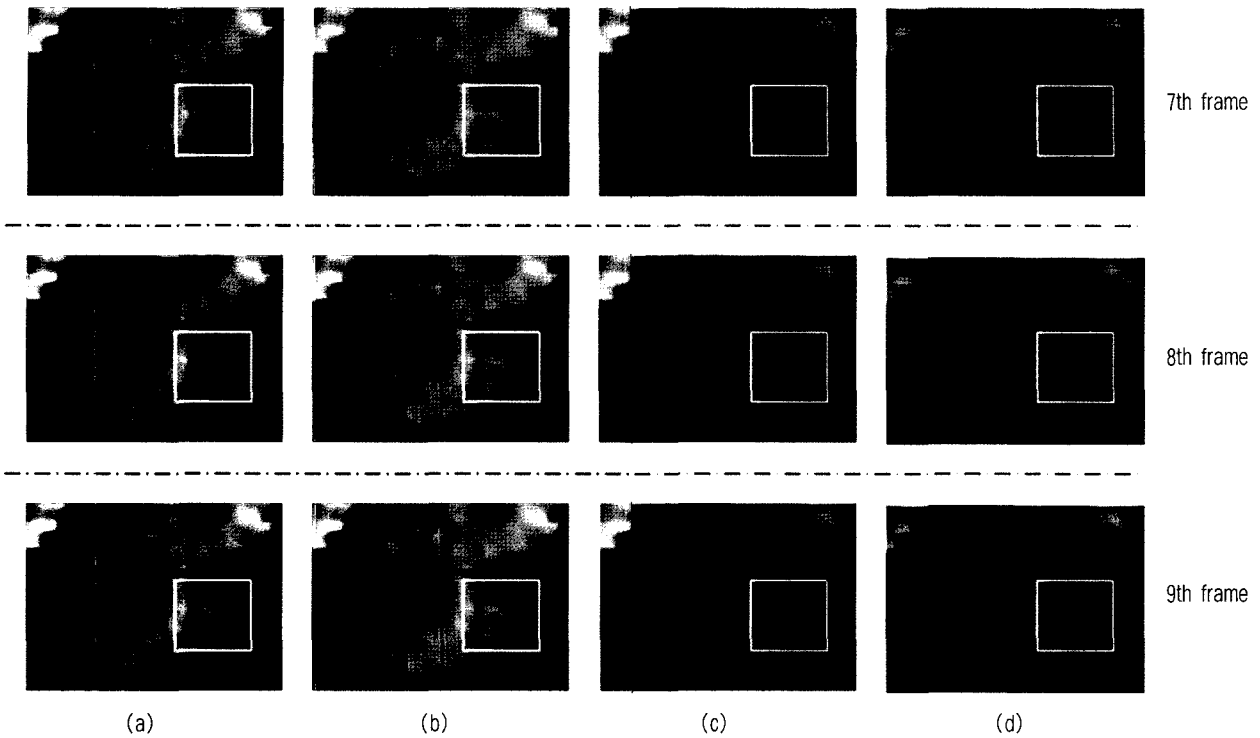


그림 12. 'Yellow flower'의 합성된 프레임, (a) 일반적인 분할방법에 의한, (b) 새로운 분할방법에 의한, (c) 일반적인 분할방법과 적응적 시간평활화를 적용한 그리고 (d) 새로운 분할방법과 적응적 시간평활화를 적용하여 얻은 합성 영상

Fig. 12. The synthesized frames of 'Yellow flower' of Fig. 10(b). (a) By general split-scheme, (b) By new split-scheme, (c) By general split-scheme with adaptive temporal smoothing, and (d) By new split-scheme with adaptive temporal smoothing. The synthesized frames are enlarged and contrast-compensated for the visibility

실제로 그림 11(d) 및 그림 12(d)에서처럼 PUPPY의 정적인 배경에서 발생하는 플리커링이 본 논문에서 제안된 새로운 분할방법 및 적응적 시간평활화를 동시에 적용함으로써 획기적으로 감소됨을 확인되었다.

4가지 다른 방식에 의해 합성된 중간영상의 PSNR을 표 5에 요약하였다. 새로운 분할방법에 의해 PSNR이 약간 증

가하였는데, 이는 물체간 경계에서 플리커링이 일부 제거되었기 때문이다. 하지만 적응적 시간평활화가 적용된 경우에는 PSNR이 약간 감소되는 경향을 보이는데, 이는 물체간 경계에서 플리커링이 줄어 화질이 개선됨에도 불구하고, 전체적인 이미지의 합성오차가 증가되기 때문인 것으로 판단된다.

본 논문에서 제안된 아이디어가 동적인 배경(dynamic background)을 갖는 스테레오 영상의 중간영상을 합성할 때도 합성된 중간영상에 인공적인 결함(artifact)을 유발하지 않음을 확인하기 위하여 CROWD와 WORLDCUP2002를 이용하여 검사하였다. CROWD와 WORLDCUP2002의 합성된 중간영상의 몇 프레임은 그림 13(a) 및 그림 13(b)에 각각 나타내었다. 제안된 아이디어를 이용하여 합성한 CROWD와 WORLDCUP2002의 중간영상은 인공적인 결함(artifact) 없이 품질도 우수함을 확인하였다.

표 5. 4가지 다른 방법에 의한 합성영상의 PSNR
Table 5. The PSNR values by using four different methods.

	General Split-scheme	New Split-scheme	General Split-scheme + Adaptive Temporal Smoothing	New Split-scheme + Adaptive Temporal Smoothing
PSNR [dB]	27.174	27.192	27.167	27.185

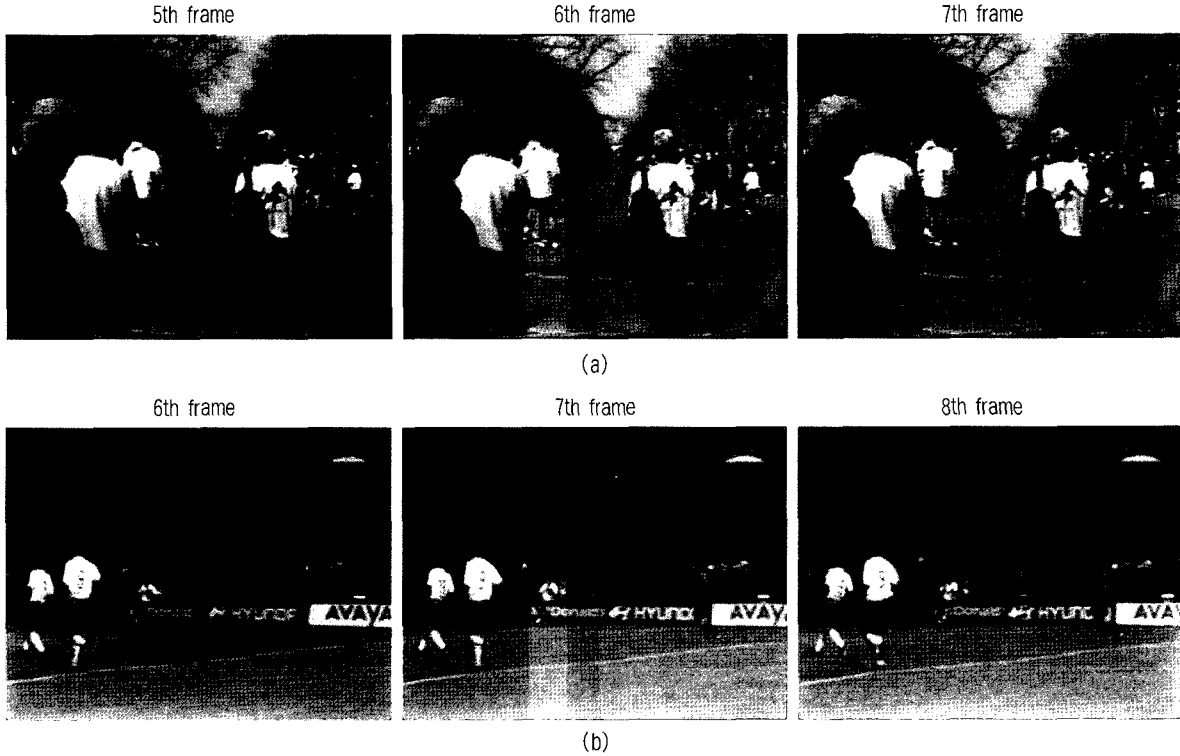


그림 13. 합성된 중간영상 (a) CROWD의 합성된 중간영상, (b) WORLD CUP 2002의 합성된 중간영상
 Fig. 13. Synthesized intermediate sequences. (a) Synthesized intermediate sequence of CROWD, (b) Synthesized intermediate sequence of WORLD CUP 2002.

IV. 결 론

다시점 3차원 디스플레이에 적용 가능한 쿼드트리 기반의 변이추정 및 중간영상합성에 대해 소개하였다. 본 연구에서 제안한 새로운 쿼드트리 분할방법 및 적응적 시간평활화를 이용하여 쿼드트리 기반의 변이추정을 이용한 중간영상합성 시 합성된 중간영상의 물체간 경계에서 발생하는 플리커링을 대폭 감소시킬 수 있었다.

컴퓨터 시뮬레이션을 통해 부적절한 쿼드트리 분할에 기인한 플리커링을 감소시키는데 있어, 본 논문에서 제안한 이중임계값을 갖는 새로운 분할방법이 효과적임을 확인하였다. 새로운 분할방법을 적용할 때, 쿼드트리 분할 확률이 약간 증가되어 계산량 부담(computational burden)이 약간 증가되며, 이전프레임의 분할정보를 기억하기 위한 약간의 메모리가 요구된다.

부정확한 변이추정에 기인한 플리커링은 적응적 시간평활화를 적용함으로써 상당한 수준으로 감소시킬 수 있었다.

이전프레임과 현재프레임의 상관성을 검사하고, 시간평활화 계수를 계산하기 위하여 약간의 하드웨어 복잡도의 증가가 요구된다. 또한, 부정확한 변이값을 추정할 경우, 그것이 시간적으로 전파될 수 있으므로 적응적 시간평활화를 이용하기 위해서는 보다 엄정한 변이추정이 요구된다.

시뮬레이션 결과로부터 새로운 분할방법 및 적응적 시간평활화를 동시에 사용할 때, 가장 우수한 중간영상(intermediate video sequence)을 합성할 수 있었으며, 약간의 계산량 및 하드웨어 복잡도의 증가를 감수할 만큼 개선된 영상을 얻을 수 있었다.

제안된 아이디어는 정적인 배경을 갖는 스테레오 영상으로부터 중간영상을 합성할 때 매우 효과적이었으며, 동적인 배경을 갖는 스테레오 영상으로부터 중간영상을 합성할 때 어떤 인공적인 결함(artifact)도 발생시키지 않음을 확인하였다. 그러므로 제안된 아이디어는 임의의 스테레오 영상으로부터 중간영상을 합성할 때 적용 가능함을 확인하였다.

제안된 아이디어를 적용했을 때, 합성된 중간영상의 물

체간 경계에서 발생하는 플리커링을 완전히 제거하지는 못하였다. 이는 완전하지 못한 변이추정에 의한 것으로 판단되며, 더욱 개선된 중간영상을 얻기 위해 보다 정확한 변이추정등 추가적인 개선 노력을 계속하고자 한다.

참 고 문 헌

- [1] M.H. Kim and K.H. Sohn, "Edge-preserving directional regularization technique for disparity estimation of stereoscopic images," IEEE Trans. Consumer Electronics, vol. 45, no. 3, pp. 804-811, Aug. 1999.
- [2] J. Konrad, "View reconstruction for 3-D video entertainment: issues, algorithms and applications," Proc. of Int. Conf. on Image Processing and its Applications, pp. 8-12, July 1999.
- [3] A. Mancini and J. Konrad, "Robust quadtree-based disparity estimation for the reconstruction of intermediate stereoscopic images," IS&T/SPIE Symposium on Electronic Imaging Stereoscopic Displays and Virtual Reality Systems, vol. 3295, pp. 53-64, Jan. 1998.
- [4] A. Puri, R.V. Kollarits, and B.G. Haskell, "Basics of stereoscopic video, new compression results with MPEG-2 and a proposal for MPEG-4," Signal Processing: Image Communication, no. 10, pp. 201-234, 1997.
- [5] C.L. Pagliari, M.M. Perez, and T.J. Dennis, "Reconstruction of intermediate views from stereoscopic images using a rational filter," Proc. of IEEE Int. Conf. on Image Processing, vol. 2, pp. 627-631, Oct. 1998.
- [6] J.R. Ohm and E. Izquierdo, "An object-based system for stereoscopic viewpoint synthesis," IEEE Trans. Circuits Sys. Video Tech., vol. 7, no. 5, pp. 801-811, Oct. 1997.
- [7] M.M. Perez, C.L. Pagliari, and T.J. Dennis, "Stereo-based intermediate view synthesis with realistic 'look around' capability," Electronic Letters, vol. 34, no. 19, Sep. 1998.
- [8] C. Vazquez, J. Konrad, and E. Dubois, "Wavelet-based reconstruction of irregularly-sampled images: application to stereo imaging," Proc. of Int. Conf. on Image Processing vol. 2 pp. 319 - 322 Sept. 2000.
- [9] M. B. Slima, J. Konrad, and A. Barwicz, "Improvement of stereo disparity estimation through balanced filtering: the sliding-block approach," IEEE Trans. Circuits and Systems, Video Tech., vol. 7, no. 6, pp. 913-920, Dec. 1997.

저 자 소 개



성 준 호

- 1988년 : 광운대학교 전자재료공학과 학사
- 1990년 : 광운대학교 전자재료공학과 석사
- 2001년 ~ 현재 : 연세대학교 전기전자공학과 박사과정
- 1991년 ~ 1996년 : 삼성종합기술원 디스플레이 Lab. 전문연구원
- 1997년 ~ 현재 : 삼성전자(주) VD사업부 책임연구원
- 주관심분야 : 비디오 관련 알고리즘 연구 및 H/W 설계, 3D Video Processing



이 성 주

- 1993년 : 연세대학교 전자공학과 학사
- 1998년 : 연세대학교 전자공학과 석사
- 2002년 : 연세대학교 전기전자공학과 공학박사
- 2003년 ~ 현재 : 삼성전자(주) VD사업부 책임연구원
- 주관심분야 : 통신 및 비디오 관련 알고리즘 연구 및 H/W 설계

저 자 소 개



김 성 식

- 연세대학교 전기전자공학과 공학박사
- 삼성전자(주) VD사업부 책임연구원
- 주관심분야 : 3D Display System



하 태 현

- 1993년 : 부산대학교 전자공학과 학사
- 1995년 : 부산대학교 전자공학과 석사
- 2001년 ~ 현재 : 연세대학교 전기전자공학과 박사과정
- 1995년 ~ 현재 : 삼성전자(주) VD사업부 책임연구원
- 주관심분야 : 통신 및 비디오 관련 알고리즘 연구 및 H/W 설계



김 재 석

- 1977년 : 연세대학교 전자공학과 학사
- 1979년 : 한국과학기술원 전기 및 전자공학과 석사
- 1988년 : Rensselaer Polytechnic Institute 공학박사
- 1979년 ~ 1984년 : 한국전자통신연구원 선임연구원
- 1988년 ~ 1993년 : AT&T Bell Lab. Member of Technical Staff
- 1993년 ~ 1995년 : 한국전자통신연구원 VLSI 연구실장
- 1995년 ~ 현재 : 연세대학교 전기전자공학과 교수
- 주관심분야 : 통신 및 비디오 관련 알고리즘 연구 및 H/W 설계