

# 움직임 추정을 이용한 애니메이션 영상의 장면전환 검출

## A Scene Change Detection using Motion Estimation in Animation Sequence

곽성근(Sung-Keun Kwak)<sup>1)</sup>

### 요 약

애니메이션 영상에서 현재 블록의 움직임 벡터와 이전 블록의 움직임 벡터는 시간적 상관성을 갖고 있다. 본 논문에서는 영상의 시간적인 특성과 움직임 벡터의 중심 분포 특성을 이용하는 장면전환 검출 알고리즘을 제안한다. 제안된 알고리즘은 이전 프레임 블록으로부터 예측된 움직임 벡터와 분할된 탐색 구간에 속하는 후보 벡터 중에서 가장 작은 SAD 값을 갖는 점을 정확한 움직임 벡터를 찾기 위해서 초기 탐색점 위치로 결정한다.

실험 결과 제안된 방식은 기존의 대표적인 장면전환 검출 방식들 보다 재요청 측면에서 더 좋은 성능을 나타내었으며, 제안된 방법은 빠르고 정확하며 저장 공간을 적게 사용하는 장점을 가진다.

### Abstract

There is the temporal correlation of a animation sequence between the motion vector of current block and the motion vector of previous block. In this paper, we propose the scene change detection algorithm for block matching using the temporal correlation of the animation sequence and the center-biased property of motion vectors.

The proposed algorithm determines the location of a better starting point for the search of an exact motion vector using the point of the smallest SAD(sum of absolute difference) value by the predicted motion vector from the same block of the previous frame and the predictor candidate point on each search region.

Simulation results show that the proposed algorithm has better detection performance, such as recall rate, then the existing method. The algorithm has the advantage of speed, simplicity and accuracy. In addition, it requires less amount of storage.

논문 접수 : 2008. 10. 1.

심사 완료 : 2008. 10. 9.

---

2) 정회원 : 시립인천전문대학 디지털콘텐츠계열(애니메이션) 교수

※ 본 논문은 2007학년도 시립인천전문대학 학술연구지원비에 의한 것임.

## 1. 서론

디지털 영상 데이터에서 유용한 정보를 자동으로 추출하고 정지 영상이나 비디오 형태로 효율적으로 검색할 수 있는 영상 검색 시스템은 매우 중요하다. 지금까지 정지 영상 부분에서 내용을 기반으로 한 영상 검색 기술에 관한 연구가 활발하게 이루어졌고, 비디오의 경우는 연속된 장면들을 대표하는 키 프레임(Key Frame)을 추출하는 방법에 관한 연구가 진행되고 있다. 이러한 비디오에 있어서 검색 시스템 구축을 위한 첫 번째 단계는 비디오를 여러 개의 샷(Shot)으로 분할하는 것이다.

이와 같이 키 프레임으로 분할된 비디오는 비디오 검색을 위한 수단으로 첫째, 비디오로부터 중요한 정보를 쉽고 빠르게 찾아낼 수 있는 장면검색이 가능하고 둘째, 여러 비디오가 갖고 있는 샷의 특성들을 분석함으로써 같은 종류의 비디오별로 자동 분류가 가능하며 셋째, 검출된 키 프레임들을 이용하여 내용 기반 검색 시스템 등을 제공할 수 있다.[8]

장면전환(Scene Change)은 연속적으로 기록된 일련의 장면인 하나의 샷에서 다른 샷으로 바뀌는 것을 의미하며, 장면전환이 이루어지는 샷의 경계점을 찾는 것을 장면전환 검출(Scene Change Detection)이라고 한다.

애니메이션 영상의 현재 블록의 움직임 벡터(Motion Vector)와 이전 블록의 움직임 벡터는 시간적 상관성을 갖고 있으므로 이를 이용하여 장면전환 검출이 가능하다. 애니메이션에서 인접 프레임간의 시간 간격은 매우 짧기 때문에 단위 프레임의 시간당 움직임 크기 변화량은 일반적으로 적은 범위로 제한된다고 볼 수 있다. 즉, 연속하는 두 프레임간의 움직임에 많은 시간적 중복성을 가지고 있으므로 참조 프레임의 움직임 정보를 현재 프레임의 동일한 위치 매크로 블록의 탐색 시작점으로 사용함으로써 적은 탐색점들을 사용하여 움직임 벡터를 구할 수 있고, 양호한 장면전환 검출 결과를 얻을 수 있다.

본 연구에서는 애니메이션 검색 시스템 구축의 첫 단계인 장면전환 검출을 위한 새로운 알고리즘을 제안한다. 제안된 알고리즘은 압축영역에서 이루어지게 되는데, 인접한 DC 영상간의 움직임 벡터를 추정된 후에 두 영상간의 움직임 보상된 상관관계를 이용하여 장면전환을 검출한다. 제안된 알고리즘은 여러 종류의 장면전환 중 급격한 장면전환을 검출하기 위해 적용되며, 이를 통하여 압축영역에서의 데이터가 어떻게 영상의 전체 정보를 나타낼 수 있는지, 그리고 장면전환을 위해 어떠한 방법으로 적용될 수 있는지를 보여준다.

## 2. 장면전환 검출

장면전환이 이루어지는 방법은 하나의 프레임을 경계로 장면전환이 이루어지는 급격한-장면전환(Abrupt Scene Change)과 여러 프레임에 걸쳐 장면전환이 이루어지는 점진적 장면전환(Gradual Scene Change)으로 분류될 수 있다.

검출 알고리즘도 급진적 장면전환과 점진적 장면전환에 따라 각각 다른 방법으로 접근되어야 한다. 점진적 장면전환의 경우에는 여러 프레임에 걸쳐 서서히 일어나는 특징이 있기 때문에 상대적으로 많은 요소들이 고려되어야 하며 급진적 장면전환보다 검출 성능이 떨어진 다. 본 연구에서 제안하는 알고리즘은 급진적 장면전환 검출에 적용된다.

기존의 장면전환 검출을 위한 알고리즘은 비압축 영역에서는 화소 단위로, 압축 영역에서는 블록 단위로 이루어지는 것이 일반적이다.

### 2.1 히스토그램 차를 이용한 컷 검출

히스토그램 차의 임계값은 검사되는 후보 컷의 매크로블록 개수  $\theta$ 에 따라 달라진다. 식 (1)은 임계값 적용 조건을 보여준다.[2]

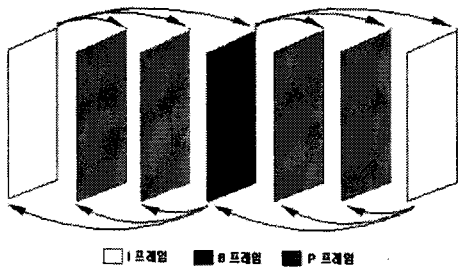
$$\begin{aligned} Th_{\max} &: \text{if } MB_{\max} \leq \theta \\ Th_{\min} &: \text{if } MB_{\min} \leq \theta < MB_{\max} \end{aligned} \quad (1)$$

임계값  $MB_{min}$ 는 후보 컷의 매크로블록의 40% 정도의 개수를 가지며,  $MB_{max}$ 는 후보 컷의 매크로블록의 70% 정도의 개수를 가진다. 여기서  $\theta_{max}$ 는 후보 컷의 매크로블록 개수  $\theta$ 가  $MB_{max}$  이상일 때 적용되는 임계값이며, 이전 장면과 상당히 다른 장면검출을 할 수 있도록 작은 임계값을 주었다.

히스토그램 차를 이용하는 컷 검출은 일반적인 장면전환에는 좋은 결과를 보여주지만, 순간적인 밝기 변화나 점진적인 장면전환에서는 같은 장면임에도 불구하고 오류의 컷으로 검출되기도 한다. 이러한 오류를 방지하기 위하여 후보 컷간의 픽셀 특징값의 평균값과 분산값을 연속적으로 비교한다. 커튼 효과나 페이딩 효과가 발생하는 점진적인 장면전환에서는 샷의 프레임들이 점차적으로 단일색 프레임으로 전환된 후에 새로운 샷의 프레임이 점진적으로 나타나는 특징을 가진다.

2.2 매크로블록 정보를 이용한 후보 컷 검출

애니메이션 시퀀스 데이터는 매크로블록 단위로 부호화가 이루어지며, 각 프레임의 상황에 따라 매크로블록을 적절히 부호화 한다.



[그림 1] 프레임 분류법에서의 동작 분석  
Fig 1. Motion Analysis of frame classification

프레임 분류법은 [그림 1]과 같이 연속되는 이미지 프레임의 동작 분석을 위해서 시작되는 프레임과 끝나는 프레임 또는 장면의 변화가 두드러진 프레임을 I 프레임(Intra Frame)으로

분류하고 I 프레임들의 사이를 몇 단계로 나누어서 예측 프레임인 P 프레임(Forward Predicted Frame)으로 분류하고 I 프레임과 P 프레임 또는 P 프레임과 P 프레임의 사이에 있는 프레임을 B 프레임(Bidirectional Predicted Frame)으로 분류하여 I 프레임과 P 프레임은 프레임 자체를 분석하고, B 프레임은 앞의 두 프레임과 관계를 보간하여 동작을 분석하게 된다.[1][7]

이때 각 프레임의 상황에 따라 매크로블록을 적절히 부호화하여 장면전환을 검출한다. 그리고 움직임 보상이 없는 경우나 해당 프레임 내에서 움직임 보상을 한 결과보다 인트라 모드로 부호화하는 것이 우수한 경우는 프레임 종류에 관계없이 매크로블록을 인트라 모드로 부호화한다.[3]

인트라 모드로 부호화된 매크로블록의 개수  $\theta$ 가 많은 프레임은 장면전환이 발생했을 가능성이 많으므로 해당 프레임을 후보 컷으로 검출한다. 검출된 후보 컷의  $\theta$ 는 히스토그램 차의 임계값 선택에 중요한 정보가 된다. 식 (2)는 후보 컷을 검출하기 위한 조건이다.

$$MB_{min} \leq \theta \tag{2}$$

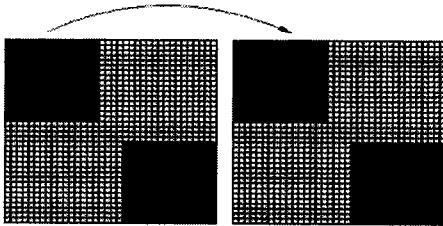
임계값  $MB_{min}$ 는 인트라 모드로 부호화된 매크로블록의 최소 개수로 총 매크로블록 개수의 40%를 가정한다.

3. 제안한 장면전환 검출

제안된 알고리즘은 압축영역에서 이루어지게 되는데, 인접한 DC 영상간의 움직임 벡터를 추정한 프레임 정보의 상관관계를 이용하여 급진적 장면전환을 검출한다.

프레임에서 장면전환이 일어날 때 칼라나 밝기가 이전 프레임과 매우 큰 차이가 나타나게 되며, 갑작스런 변화를 인지하는데 움직임 벡터는 비중 있는 요소가 된다. 또한 인트라로 부호화된 프레임에서 8x8 DCT(Discrete

Cosine Transform) 블록의 계수들은 공간 영역에서 8×8 블록의 휘도 및 색차 성분과 관련이 있다.



■ 움직임 벡터를 구성하기 위한 비교 블록

[그림 2] 블록 분류법에서의 동작 분석

Fig 2. Motion Analysis of block classification

본 연구에서는 움직임 추정 방법은  $p=\pm 2$ 의 중앙 5×5 화소 내의 영역을 움직임 존재 가능 위치[5]로 설정하여 이를 중심으로 움직임 벡터가 존재할 탐색 구간을 <표 1>과 같이 중심에서 45도 간격으로 9개 구간을 설정한다. 이렇게 얻어진 각 구간 내에 정확한 움직임 벡터가 있다고 가정한다.

<표 1> 이전 프레임이 불일치일 때 일치영역에 포함될 범위

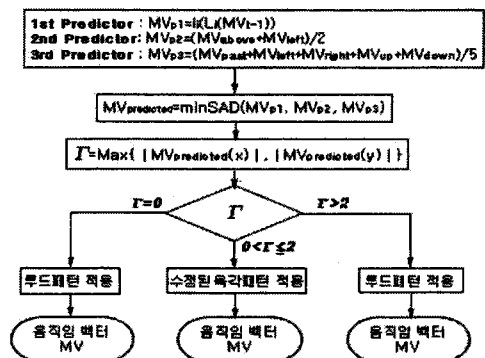
탐색구간	포함영역	수정되는 탐색 구간 설정 범위
0	1,2,3,4,5,6,7,8	$-2 \leq x \leq 2, -2 \leq y \leq 2$
1	0,2,3,7,8	$3 \leq x \leq 7, -2 \leq y \leq 2$
2	0,1,3	$3 \leq x \leq 7, -7 \leq y \leq -3$
3	0,1,2,4,5	$-2 \leq x \leq 2, -7 \leq y \leq -3$
4	0,3,5	$-7 \leq x \leq -3, -7 \leq y \leq -3$
5	0,3,4,6,7	$-7 \leq x \leq -3, -2 \leq y \leq 2$
6	0,5,7	$-7 \leq x \leq -3, 3 \leq y \leq 7$
7	0,1,5,6,8	$-2 \leq x \leq 2, 3 \leq y \leq 7$
8	0,1,7	$3 \leq x \leq 7, 3 \leq y \leq 7$

따라서 움직임 벡터를 빠르게 찾아내기 위해서는 주어진 각각의 구간에서 첫 번째 예측 후보점을 설정하여 탐색 속도와 성능을 향상시킨다. 또한 서로 다른 초기점 조건을 이용하여 최소 정합오차 후보에 따라 후보 벡터를 존재 가능한 방향으로 탐색을 수행함으로써 가장 적

합한 움직임 벡터를 찾으므로 매크로블록 개수에 의한 장면전환 검출하는 방법을 연구한다.

현재 블록과 같은 위치에 있는 이전 시점의 프레임 블록의 움직임 벡터가 속하는 탐색 구간에 따른 해당 후보점  $MV_{p1}$ 과 현재 프레임 블록과 같은 위치에 있는 이전 시점의 프레임 블록과 주변 매크로 블록들의 평균으로 예측되는 후보 움직임 벡터  $MV_{p2}$ 와 이전 프레임의 같은 위치에 있는 블록과 상, 하, 좌, 우의 인접 매크로 블록들을 포함한 5개의 움직임 벡터의 평균으로 예측 한다. 후보 움직임 벡터  $MV_{p3}$ 을 구하여 이 3개의 후보점 중에서 최소 SAD인 최적합점을 찾아 최소 SAD인 최적합점의 움직임 벡터의 크기  $\Gamma$ 를 구한다. 최적합점으로 탐색 원점을 이동하여 이동된 탐색 원점을 중심으로  $\Gamma=0$ 일 때는 루드 패턴을,  $0 < \Gamma \leq 2$ 일 때는 수정 육각 패턴을 사용하며,  $\Gamma > 2$ 이면 지역 탐색 패턴을 적용한다.[4]

[그림 3]은 각 움직임 예측 벡터의 SAD를 비교하여 가장 작은 값을 갖는 점을 구한 후, 이 점을 중심으로 임계값  $\Gamma$ 에 따라 적용적인 탐색 패턴에 의해 움직임 벡터를 추정하는 과정을 나타낸다.

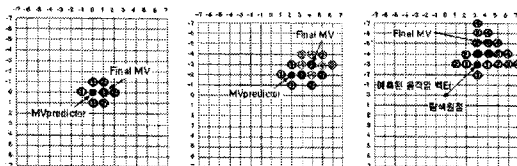


[그림 3] 제안한 움직임 벡터 추정 방식의 흐름도

Fig 3. Flowchart of Proposed Motion Vector Estimation

각 패턴의 탐색 과정은 탐색 원점을 중심으로 5개의 탐색점에 대하여 최소 정합 오차를 계산한다. 이때 중심점이 최소 정합 오차이면 탐색을 중단하고 이 점을 최종적으로 움직임 벡터로 결정한다. 만일 중심점이 최소 정합 오차가 아니라면, 루드 패턴인 경우는 최소 정합 오차를 가지는 점을 중심점으로 탐색을 반복한다. 수정된 육각 패턴인 경우는 최소 정합 오차인 점을 중심으로 큰 납작한 육각 패턴을 구성하여 최소 정합 오차를 계산한다. 새로 계산된 최소 정합 오차 점이 중심점에 위치할 때까지 탐색 과정을 반복하여 최소 정합 오차를 구하며, 최종적으로 최소 정합 오차 점이 중심점일 경우는 반경 1화소에 해당하는 이웃 4점을 포함하는 루드 패턴을 구성하여 최소 정합 오차를 계산하여 이 단계에서 구해진 최소 정합 오차 점이 움직임 벡터로 결정된다. 그리고 매 단계마다 최소 정합 오차는 재정의된다.

[그림 4]는 3개의 후보 벡터중에서 최소 SAD인 최적합점으로 탐색 원점을 이동하여  $\Gamma=0$ 이면 루드 패턴을 적용하고,  $0 < \Gamma \leq 2$  일 때는 수정 육각 패턴을,  $\Gamma > 2$  이면 지역 탐색 패턴을 적용하여 블록 정합을 수행하는 과정을 각각 보인 것이다. 제안된 방식은 탐색 영역이 벗어나는 모든 점을 무시하고 중간에 탐색을 멈추는 방법을 제시하였는데 중간에 멈출 수 있는 조건은 현재 탐색 단계에서 최소 정합 오차를 갖는 점의 위치가 이전 탐색 단계의 최소 정합 오차를 갖는 점의 위치와 같다면 중간에 탐색을 멈추고 최종적인 움직임 벡터로 결정한다.



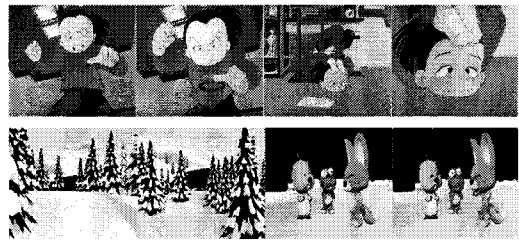
(a)  $\Gamma=0$       (b)  $0 < \Gamma \leq 2$       (c)  $\Gamma > 2$

[그림 4] 탐색 패턴의 적용과 탐색 경로  
Fig 4. Search Path and Search Pattern

따라서 제안된 기법은 예측된 움직임 벡터를 초기점으로 하여 매 탐색 단계마다 지역 전체를 할당하지 않고 해당 방향의 탐색 후보점만 탐색함으로써, 이전 매크로블록(Forward Macroblock) 오류 신호의 평균이 어떤 임계치 이상일 때 P 프레임에서 장면전환이 일어났다고 보면 B 프레임에서는 이전 프레임과 현재 프레임간 차를 계산하여 샷 경계를 판단한다.

#### 4. 실험 결과

제안된 장면전환 검출 방법의 성능을 평가하기 위하여 일부 추출한 4개의 애니메이션 영상을 실험 영상으로 이용하였다.



[그림 5] 실험 영상에서 장면전환의 예  
Fig 5. Sample of Experimental Sequence

먼저 실험 영상의 움직임 추정은 블록 정합을 이용하였다. 그리고 블록 정합의 정도를 평가하기 위해 대표적인 정합 기준인 평가 함수(Cost Function)로 영상 화질의 품질을 평가하기 위한 식 (3)의 평균 제곱 오차(MSE: Mean Squared Error)와 정합 오차 측정 함수로는 식 (4)의 절대값 오차의 합(SAD: Sum of Absolute Difference)을 이용하였다.

$$MSE(i,j) = \left(\frac{1}{N^2}\right) \sum_{k=1}^N \sum_{l=1}^N |I_k(k,l) - I_{k-1}(k+i,l+j)|^2 \quad (3)$$

$$SAD(i,j) = \sum_{k=1}^N \sum_{l=1}^N |I_k(k,l) - I_{k-1}(k+i,l+j)| \quad (4)$$

여기서 N은 영상의 가로와 세로의 각각의 크기이고,  $I_k(k,l)$ 은 원영상의 화면을 나타내고,

$I_{t-1}(k+i, l+j)$ 은 움직임 추정 화면을 나타낸다. 이들 정합 기준에서 최소값을 가지는 위치를 움직임 벡터로 결정하여 구해진 움직임 크기가 40% 이하를 무시한 각 프레임의 움직임 정보로부터 샷의 움직임 정보를 계산하였다. 이때 제안한 장면전환 검출 알고리즘의 성능을 평가하기 위해서 일반적으로 사용되는 식 (5)의 Recall Rate와 식 (6)의 Precision Rate를 사용 평가 척도로 사용하였다.

$$Recall\ Rate = \frac{D}{D+M} \times 100[\%] \quad (5)$$

$$Precision\ Rate = \frac{D}{D+F} \times 100[\%] \quad (6)$$

<표2>는 적용 방식에 따른 실험 결과를 보여준다. 여기서 N은 바람직한 장면의 수이며, D(Detected)는 검출된 장면의 수이다. 그리고 M(Missies)은 장면 경계로 검출되지 않은 회수를 나타내며, F(False Alarms)는 검출된 샷 중 잘못 검출된 샷을 의미하고, Recall Rate인 R과 Precision Rate인 P는 백분율[%]로 나타내었다.

<표3>은 샷으로 구성된 시퀀스 기호열을 장면 전환 그래프를 사용하여 모델링한 시각 정보의 유사성과 시간축 상의 원도를 이용한 이산적인 방식[6]과 비교한 결과이다.

<표 3> 기존 방식과의 비교[%]

Data Sequence	기존 방식		제안한 방식	
	Recall	Precision	Recall	Precision
영상1	76.5	81.3	82.3	77.8
영상2	73.3	84.6	73.3	78.6
영상3	71.4	75.0	76.2	72.7
영상4	68.8	68.8	75.0	75.0

본 논문에서 제안한 방식이 기존 방식보다 정확성이 낮은 경향은 일부 있지만 검출 소요 시간이나 저장 공간 면에서는 우수한 결과가 나왔다. 정확성 면에서는 참조 프레임에 대한 움직임 추정 정보가 장면전환에 반영되기 때문에 움직임이 심한 샷에서는 이를 장면전환으로 잘못 인지하였기 때문이다. 그러나 전체적으로 73% 이상의 검출 정확성을 가지므로 각 샷에 대한 정보는 다양한 검색 및 인덱싱이 가능하게 되었다.

<표 2> 장면전환 검출 결과

Data Sequence	N	D	M	F	R [%]	P [%]
	(Num.of SC)	(Detected)	(Missies)	(False Alarms)	(Recall)	(Precision)
영상1 (20min)	17	14	3	4	82.3	77.8
영상2 (18min)	15	11	4	3	73.3	78.6
영상3 (25min)	21	16	5	6	76.2	72.7
영상4 (20min)	16	12	4	4	75.0	75.0

## 5. 결 론

본 논문의 실험에서 영상의 세기에 근거한 각종 컷 검출법을 비교하였고, 영상은 편집시 전혀 효과를 사용하지 않는 단순 영상과 각종 효과를 가미한 특수효과 영상으로 구분하여 비교 하였다. 그리고 움직임 추정을 이용한 급진적 장면전환 검출 방법을 구현하여 실험용으로 편집된 애니메이션 영상 시퀀스에 적용하였다.

그러나 제안한 장면전환 검출은 움직임이 심한 샷에서는 잘못 검출하는 경우가 많아지는 단점이 있었다. 그러므로 이러한 움직임의 영향을 줄이기 위해서 적절한 움직임 보상이 이루어져야 한다.

또한 움직임 벡터의 예측 과정에서 발생할 수 있는 예외적인 요소를 모두 고려하여 가장 적응적인 현재 프레임의 매크로 블록 내에서 움직임 벡터를 구간별로 탐색하므로 본 연구에서 제안한 탐색 기법을 사용하여 움직임 예측을 한다면 보다 빠르게 움직임 벡터를 찾을 수 있을 것이며, 또한 동영상 압축의 성능을 향상시키고 우수한 보상 결과를 얻을 수 있을 것으로 기대된다.

## 참고문헌

- [1] W. A. C. Fernando, "Sudden Scene Change Detection in Compressed Video using Interpolated Macroblocks in B-frames", *Multimedia Tools and Applications*, Vol. 28, No.3, pp.301-320, May, 2006.
- [2] Goela, N., Wilson, K., Feng Niu, "An SVM Framework for Genre-Independent Scene Change Detection", *Multimedia and Expo, 2007 IEEE International Conference on*, pp.532-535, Jul., 2007
- [3] X Yi, N Ling, "Fast Pixel-Based Video Scene Change Detection", *Circuits and Systems, IEEE International Symposium on*, Vol.4, pp.3443-3446, May, 2005.
- [4] Y. Nie, K. Ma, "Adaptive Rood Pattern Search for Fast Block-Matching Motion Estimation", *IEEE Transactions on Image Processing*, Vol. 11, No. 12, pp.1442-1449, Dec., 2002.
- [5] F. Moschetti, M. Kunt, "A Statistical Block-Matching Motion Estimation", *IEEE Transactions on Circuits & System for Video Tech.*, Vol. 13, No. 4, pp.417-431, Apr., 2003.
- [6] M. Yeung, B. Yeo, B. Liu, "Segmentation of Video by Clustering and Graph Analysis", *Computer Vision and Image Understanding*, Vol. 71, No. 1, pp.94-109, 1998
- [7] Lelescu, D., "Statistical Sequential Analysis for Real-Time Video Scene Change Detection on Compressed Multimedia Bitstream", *Multimedia, IEEE Transactions on*, Vol. 5, No. 1, pp.106- 117, Apl., 2003.
- [8] Saez, E., Benavides, J. I., Guil, N., "Reliable real time scene change detection in MPEG compressed video", *Multimedia and Expo, 2004. IEEE International Conference on*, Vol. 1, pp.567-570, June, 2004.

곽성근



1980.2 연세대학교 대학원 졸업(공학석사)

2004.8 아주대학교 대학원 컴퓨터공학과(공학박사)

1980~현재 시립인천전문대학 디지털콘텐츠계열(애니메이션) 교수

관심분야: 동영상압축, 컴퓨터그래픽스, 애니메이션