

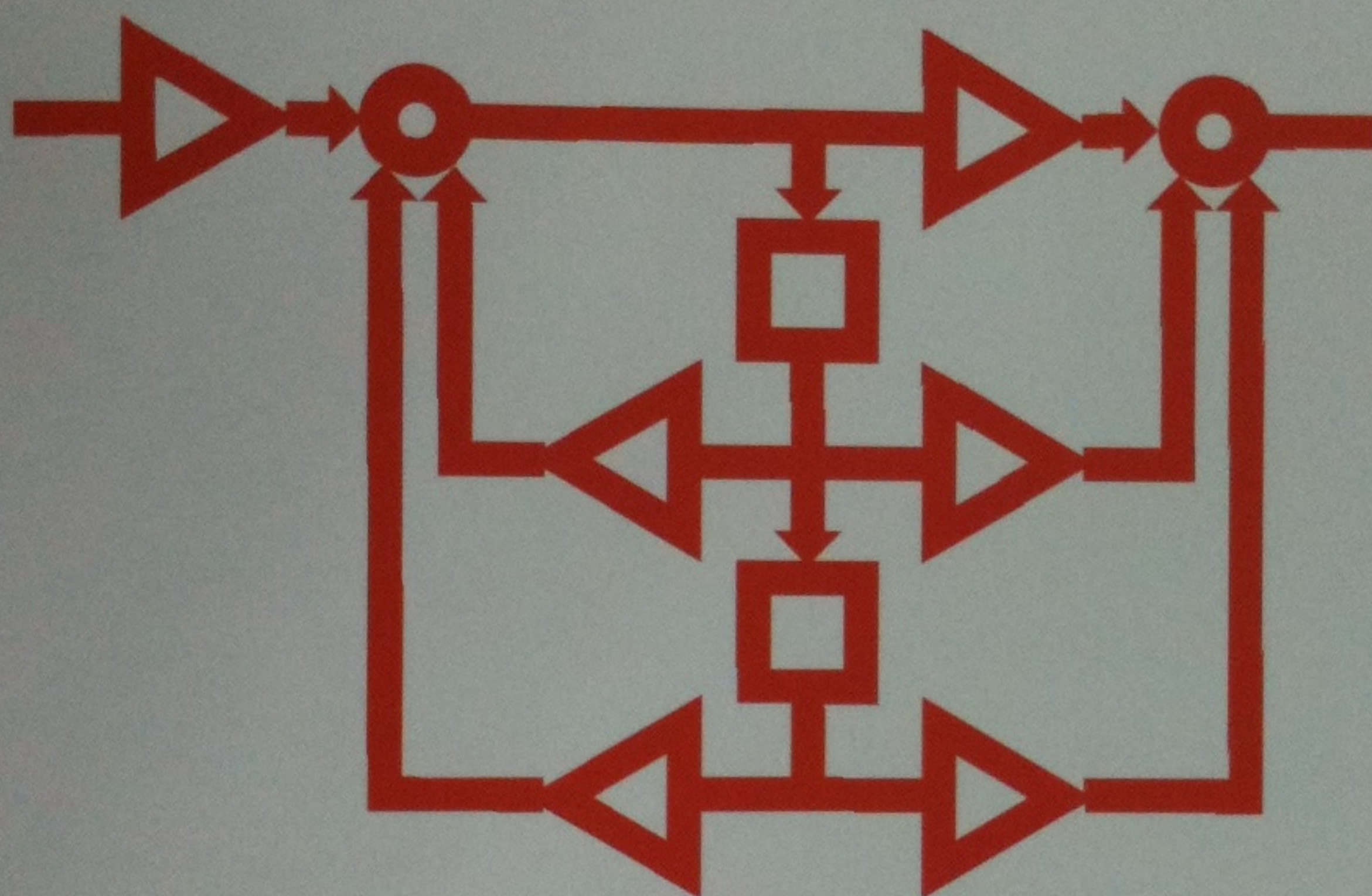
2010년도

제23회 신호처리합동학술대회 논문집

Proceeding of the 2010 Korea Signal Processing Conference

일 시 : 2010년 10월 2일 (토)

장 소 : 중앙대학교



- 주최 : 사단법인 한국방송공학회 3DTV 연구회
사단법인 대한전자공학회 음향 및 신호처리연구회, 화상처리 및 텔레비전연구회
사단법인 한국통신학회 신호처리연구회, 영상통신연구회
사단법인 한국음향학회 신호처리연구회, 음성처리연구회
- 주관 : 사단법인 한국방송공학회
- 후원 : 중앙대학교 (문화콘텐츠기술연구원, BK21 한국형디지털영상사업단, 서울미래형 콘텐츠컨버전스클러스터, 미래형영상콘텐츠기술연구단, 시각 및 지능시스템연구실), IEEE Korea SP Chapter, IEEE Korea CE Chapter
- 협찬 : (주)네오시드넷, (주)스타넥스, (주)에디텍, 인피니티박스, (주)하이트론시스템즈

구두 논문 발표 (1)

발표장소 : 제2공학관 310호

영상 압축

• 좌장 : 호요성 (GIST)

• 시간 : 10:00 ~ 11:30

- [O1-01] 분산 동영상 부호화 시스템을 위한 효과적인 상관 잡음 모델링 137
 문학수, 이창우 (가톨릭대)
- [O1-02] MCTF를 이용한 임베디드 SVC 141
 서재담, 허재성, 류철 (동국대)
- [O1-03] 확장된 매크로블록 위한 고속 움직임 예측 방법 145
 문지희, 호요성 (GIST)
- [O1-04] 색평면 간 예측값과 공간적 예측값의 적응적 가중치 평균을 이용한 인트라 예측법 149
 류재웅, 이상헌, 조남익 (서울대)

음향 처리

• 좌장 : 배명진 (숭실대)

• 시간 : 13:30 ~ 15:00

- [O1-05] 이중채널 음성인식을 위한 공간정보와 에너지를 이용한 음성구간 검출 153
 신민화, 박지훈, 김홍국 (GIST)
- [O1-06] 보컬 피치 추출 정확도 향상을 위한 보컬 강화 기술 155
 이세원*, 서병학**, 이석필**, 박호종* (*광운대, **전자부품연구원)
- [O1-07] 반복적 펄스 추출 방법을 이용한 G.718-SWB 성능 개선 157
 홍기봉*, 임종하*, 이인성*, 정규혁** (*충북대, **LG전자)
- [O1-08] 허밍 피치의 지역적 최대, 최소값 및 분포 기반 음원 검색 연구 160
 남현하*, 남기표*, 박강령** (*동국대, ***교신저자)

회로 및 시스템

• 좌장 : 백광현 (중앙대)

• 시간 : 15:30 ~ 17:00

- [O1-09] 페이지 단위별 부분 어레이 Self Refresh 방법 163
 최인수*, 강용훈**, 황주영**, 이용석* (*연세대, **삼성전자)
- [O1-10] 피크 매칭을 이용한 향상된 부채널 신호 정렬 방법 167
 이유리, 김완진, 김형남 (부산대)
- [O1-11] 높은 Slew-Rate를 갖는 모바일 TFT-LCD 구동회로를 위한 버퍼회로설계 171
 임창우, 한상욱, 최병덕 (한양대)
- [O1-12] 협대역 전력선통신을 위한 주파수 가변 대역통과 필터 설계 175
 박지영, 유영규, 최석우 (전북대)

확장된 매크로블록 위한 고속 움직임 예측 방법

문지희, 호요성
광주과학기술원 정보통신공학부

Fast Motion Estimation for Extended Block Sizes

Ji-Hee Moon, Yo-Sung Ho

Gwangju Institute of Science and Technology (GIST)

{jhmoon, hoyo}@gist.ac.kr

Abstract: H.264/AVC 표준은 16×16 크기부터 4×4 의 다양한 크기의 블록을 이용한 움직임 예측을 수행한다. 움직임을 세밀하게 예측하므로 전송해야 할 정보의 양을 효과적으로 줄일 수 있었다. 그러나 영상의 크기가 커지면서 기존의 16×16 크기의 매크로블록보다 확장된 크기의 매크로블록을 이용하여 움직임을 예측하는 것이 영상 부호화에 더 효율적이다. 하지만 확장된 크기의 매크로블록을 이용한 움직임 예측은 복잡도가 높다는 단점이 있다. 본 논문에서는 움직임 벡터의 상관도를 이용하여 탐색 영역의 크기를 적응적으로 줄이고, 움직임 예측의 정확도를 향상시키기 위해 적응적인 초기 시작점 결정방법을 제안한다. 제안한 방법을 통해 평균 55.80%의 움직임 예측 시간이 감소되었다.

Keywords: motion estimation, extended macroblock

I. 서론

H.264/AVC는 우수한 성능의 비디오 부호화 표준으로 ISO/IEC SC29WG11 산하의 동화상 전문가 그룹(Moving Picture Experts Group, MPEG)과 ITU-T SG16/Q6 동영상 부호화 전문가 그룹(Video Coding Experts Group, VCEG)이 설립한 연합 비디오 팀(Joint Video Team, JVT)에 의해 2003년 5월 제정되었다 [1]. H.264/AVC 표준은 압축 효율을 높이기 위해 기존의 표준과 다른 새로운 기술들을 사용하였다. 그 중에서 블록의 크기를 4×4 단위까지 줄여 움직임 보상을 수행하며 $1/4$ 화소 단위의 움직임 예측을 사용하여 이미 부호화된 이전 화면으로부터 움직임을 세밀하게 추정하였다.

최근 HD 및 UHD와 같은 고해상도, 고품질 영상에 대한 수요가 다양한 응용분야에서 급속히 증가하고 있다. 이러한 고해상도, 고품질 비디오는 데이터의 양이 방대하기 때문에 H.264/AVC의 압축 표준으로도 전송 또는 저장 비용이 증가하는 문제점이 있다. 이러한 문제를 해결하기 위해 MPEG과 VCEG은 2010년 1월에 협력팀(Joint Collaborative Team on Video Coding, JCT-VC)을 결성하고 공동으로 차세대 비디오 부호화 표준화(High Efficiency Video Coding, HEVC)를 시작하기로 합의하였다. HEVC는 H.264/AVC 표준보다 2배 이상의 압축률을 목표로 하고 있다 [2].

높은 압축률 향상을 위해 여러 가지 기술들이 제안되었다. 그 중에서 H.264/AVC에서 사용한 16×16 크기의 매크로블록보다 확장된 크기의 매크로블록을 사용

한 부호화 방법이 제안되었다. 일반적으로 고해상도의 영상에서는 화소 간의 상관도가 저해상도의 영상보다 높기 때문에 기존의 매크로블록의 사이즈보다 큰 블록을 이용하는 것이 영상 부호화에 더 효율적이다.

그러나 확장된 크기의 블록을 이용한 움직임 예측 방법은 부호화 효율을 향상시키지만, 계산량이 높다는 단점이 있다. 영상의 크기와 블록의 크기가 증가하면서 최적의 움직임 벡터를 찾기 위해 계산해야 할 화소의 수가 증가하기 때문이다. 그러므로, 본 논문에서는 불필요한 계산을 줄이기 위해 인접하는 블록들의 움직임 벡터의 상관도를 이용하여 확장된 크기의 매크로블록을 위한 고속 움직임 예측 방법을 제안한다. 또한, 적응적인 초기 시작점 결정 방법을 제안하여 움직임 예측의 정확도를 향상시킨다.

II. 기존의 방법

1. H.264/AVC 전 영역 움직임 탐색 방법

H.264/AVC에서 제공하는 전 영역 움직임 탐색 방법은 그림 1과 같이 이루어진다. 현재 부호화하고자 하는 매크로블록의 움직임 벡터를 예측하기 위해 움직임 벡터 간의 공간적 상관도를 이용한다. 인접하는 블록에서 정해진 움직임 벡터들의 중간 값을 수평 방향과 수직 방향으로 독립적으로 구한다 [3].

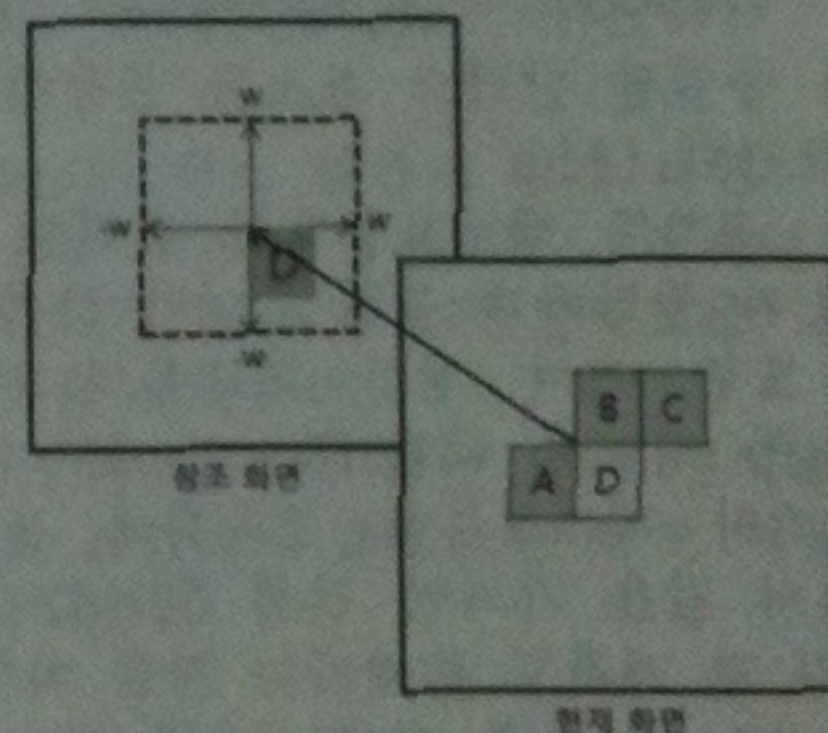


그림 1. H.264/AVC 전 영역 탐색 방법

예측된 움직임 벡터는 참조 화면에서 움직임 예측 시 초기 시작점으로 사용된다. 탐색 영역을 w 로 정했다면 예측된 초기 시작점을 기준으로 하여 상하좌우로 w 길이만큼을 탐색 영역이 정해진다. 정화소 단위로 탐색 영역의 크기는 $(2w+1)^2$ 이 된다. 정화소 단위에서 움직임 예측 과정이 진행되어 결정된 최적의 정화소를 중심으로 부화소 단위로 움직임 예측 과정을 수행한다.

2 확장된 크기의 매크로블록

기존의 H.264/AVC는 정확한 움직임 예측을 위해 5개의 인터 모드를 지원한다. SKIP, 16×16 , 16×8 , 8×16 과 $P8 \times 8$ 모드가 매크로블록 부호화에 이용된다. $P8 \times 8$ 모드는 각각의 8×8 블록 안에서 8×8 , 8×4 , 4×8 , 4×4 중의 하나로 선택된다. Key technology area (KTA)에서는 기존의 16×16 크기의 매크로블록보다 확장된 크기의 매크로블록을 이용하여 움직임 예측을 수행하고 부호화한다. 그림 2는 KTA 소프트웨어에 구현된 확장된 크기의 움직임 예측 블록 단위를 나타낸다.

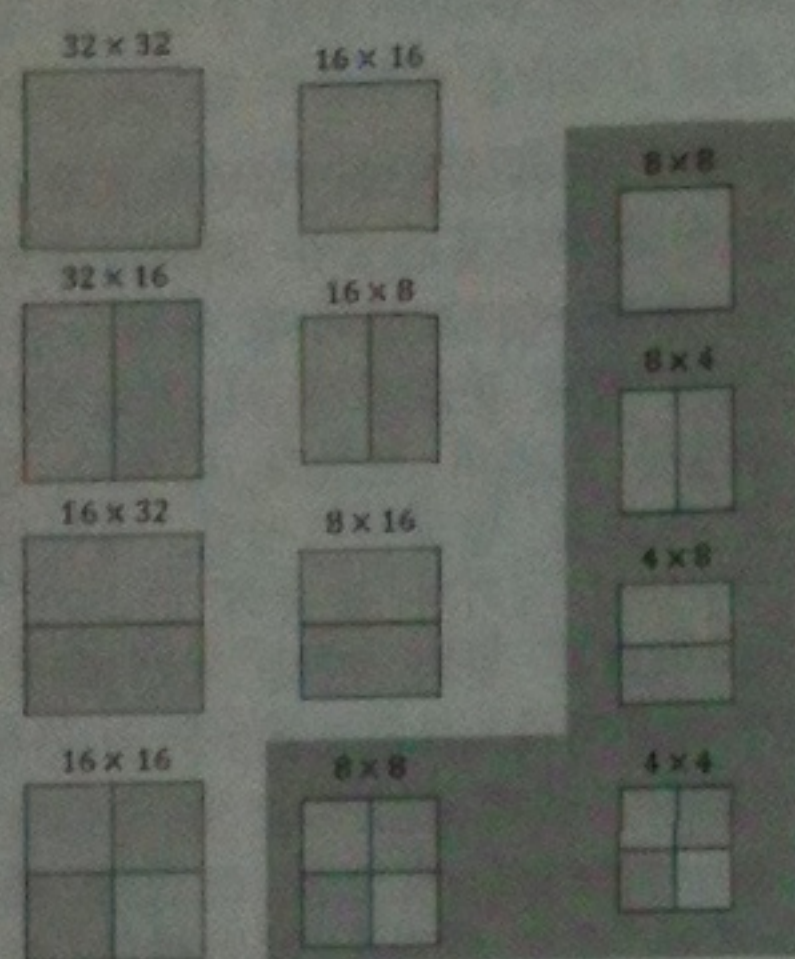


그림 2. KTA의 확장된 매크로블록 인터 모드

각각의 32×32 블록은 H.264/AVC의 매크로블록 부호화와 유사한 방법으로 부호된다. mb32_skip_flag는 H.264/AVC의 SKIP 또는 Direct 모드와 동일한 방법으로 부호된다. H.264/AVC의 움직임 예측 단위인 8×16 또는 16×8 의 mb_type을 부호하는 방법은 32×16 , 16×32 의 mb_type을 부호하는 방법에 적용된다. 예를 들어, H.264/AVC에서 mb_type이 3인 값을 가진 매크로블록은 인터 모드 8×16 를 나타낸다. 그러나 KTA에서는 mb_type이 3인 값을 가진 매크로블록은 인터 모드 16×32 를 나타낸다. mb32_type은 H.264/AVC의 mb_type과 동일한 신택스 요소를 공유하며 유사한 방법으로 부호된다 [4].

확장된 크기의 매크로블록의 움직임 벡터는 기존 H.264/AVC와 동일하게 움직임 예측 블록 당 결정되고 부호된다. 또한 잔여 영상 계수들의 Coded Block Pattern(CBP) 정보를 나타내는 cbp32 신택스 요소도 확장된 크기의 매크로블록 부호화에 포함된다. 만약 cbp32가 1의 값을 가지면 전체 32×32 블록에서 0이 아닌 계수가 한 개라도 존재하는 것을 의미하고 0의 값을 가지면 32×32 블록 안에 0보다 큰 계수가 존재하지 않음을 의미한다.

만약 32×32 블록이 4개의 16×16 블록으로 나누어지면, 4개의 16×16 블록은 raster scan 순서로 부호화되고 각각의 신택스 요소는 기존의 H.264/AVC JM reference software를 따른다. 그러나 delta QP는 전체 32×32 블록에 대해 한 번 전송된다. 즉, 32×32 블록은 모두 같은 QP를 사용한다. 각각의 16×16 블록은 쿼드 트리 방법으로 16×16 블록부터 4×4 블록까지 분할된다. cbp와 잔여 영상의 계수는 H.264/AVC의 신택스

요소와 동일하다.

32×32 블록보다 더 큰 사이즈의 64×64 의 매크로블록을 부호화하는 방법은 위에서 설명한 32×32 블록 부호화 방법과 동일하다.

III. 제안하는 방법

본 논문에서 제안하는 알고리즘은 크게 두 가지 방법이다. 움직임 벡터의 공간적 상관도와 시간적 상관도를 이용하여 탐색 영역을 적응적으로 결정하는 방법과 예측 움직임 벡터와 이미 결정된 움직임 벡터를 이용하여 움직임 예측의 정확도 향상을 위한 적응적인 초기 시작점 결정 방법이다. 본 논문에서는 제안하는 알고리즘에 대해 설명한다.

1. 공간적 상관도를 이용하는 방법

한 화면에 위치하는 인접한 화소 사이에는 높은 상관도가 존재하는데, 이는 움직임 벡터에서도 마찬가지이다. 그러므로 인접하는 매크로블록의 움직임 크기도 공간적 상관도가 높다. 기존의 움직임 예측 방식은 고정된 정사각형 모양의 탐색 영역 내에서 수행되므로 불필요한 위치를 탐색하여 계산량이 많다. 이러한 불필요한 연산을 줄이기 위해 인접하는 블록 사이의 움직임 벡터의 통계적 특성을 이용하여 수평, 수직 방향으로 각각의 움직임 예측 탐색 영역을 가변적으로 결정한다.

움직임 벡터의 통계적 특성을 이용하고자 인접하는 블록의 움직임 벡터들의 표준편차를 계산한다. 표준편차는 움직임 벡터의 수직방향과 수평방향의 움직임의 크기 정도를 의미한다. 예를 들어 수평방향의 표준편차가 수직방향의 표준편차보다 크면 현재 부호화하고자 하는 매크로블록 주변에 수평방향의 움직임이 크다는 것을 알 수 있다. 이를 이용하여 수평방향, 수직방향에 대해 각각 서로 다른 크기의 탐색 영역을 결정하게 된다. 수직방향과 수평방향 움직임 벡터의 표준편차를 이용하여 적절한 영역 내에서 움직임 예측을 수행함으로써 계산량 절감이 가능하다. 인접하는 블록들의 움직임 벡터의 표준편차를 이용하여 결정된 탐색 영역의 개념을 그림 3에 나타낸다.

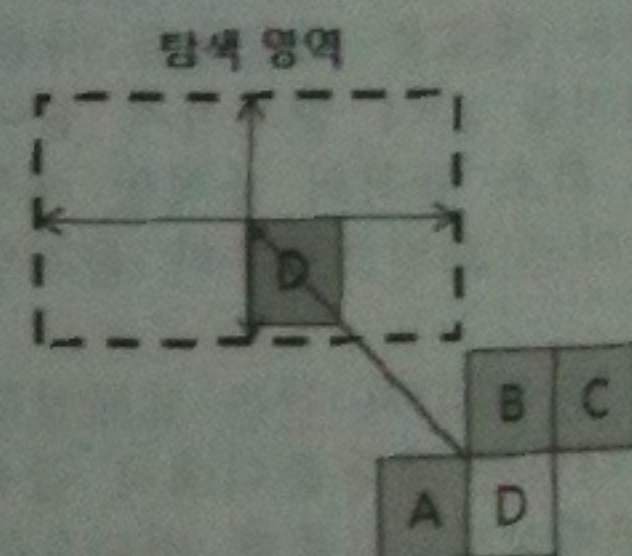


그림 3. 제안된 적응적인 탐색 영역

기존 방법의 성능을 유지하기 위해 결정된 탐색 영역은 최적의 움직임 벡터로 결정될 가능성이 있는 지점을 포함해야 한다. 본 논문에서는 기존의 전 영역 탐색 방법의 성능을 보장하기 위하여 비선형 함수를 이용하여 탐색 영역을 결정한다. 움직임 벡터의 표준편차가 지정된 탐색 영역의 크기를 넘어갔을 때 비선형 함수를 사용하여 최대 탐색 영역의 값을 가지도록 조절할 수 있다. 또한 단순히 표준편차 값을 탐색 영역의 크기

로 결정하는 방법보다 비선형적인 값으로 탐색 영역의 크기를 조정하여 움직임 벡터의 표준편차 값보다 더 큰 값을 탐색 영역의 크기로 결정하여 움직임 예측 성능을 보장할 수 있다. 식 (1)은 표준편차를 이용하여 수평방향과 수직방향의 탐색 영역이 결정되는 식을 나타낸다.

$$R = \max(1, \alpha \sqrt{\text{standard_deviation}(MVs)}) \quad (1)$$

$$\alpha = \text{search_range} / 2\sqrt{\text{search_range}}$$

여기서 α 는 표준편차를 이용하여 결정된 탐색 영역이 사용자가 설정한 최대 탐색 영역을 넘어가는 것을 막기 위한 크기 조정 인자이다.

2. 시간적 상관도를 이용하는 방법

공간적인 특성을 이용하여 결정된 탐색 영역은 모든 참조 화면에서 동일한 크기로 적용된다. 그러나 실제 최적의 참조 화면은 현재 부호화하고자 하는 매크로블록에서 가장 가까운 화면으로 선택될 확률이 높다. 최적의 참조 화면이 선택되는 빈도를 그림 4에 나타낸다.

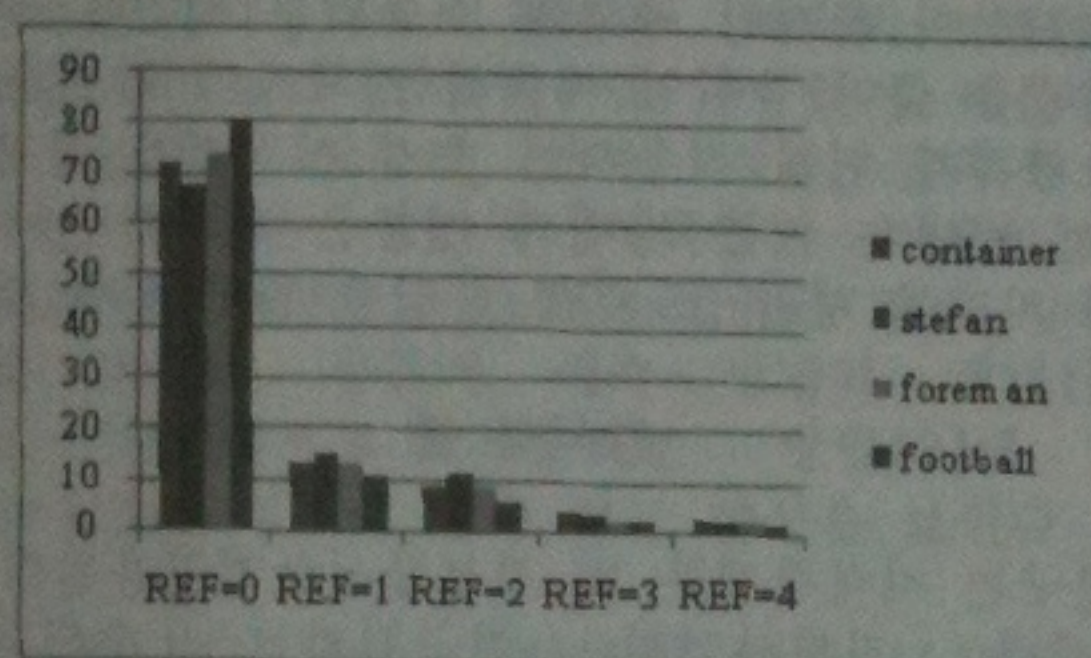


그림 4. 최적의 참조 화면 분포도

그림 4를 보면, 현재 부호화하고자 하는 매크로블록에서 가장 가까운 참조 화면부터 세 번째 참조 화면이 최적의 참조 화면으로 선택될 확률이 95% 이상임을 확인할 수 있다. 이는 부호화하고자 하는 매크로블록에서 가장 멀리 있는 참조 화면에서 동일한 움직임 예측 과정을 수행하여 불필요한 계산량이 증가하는 것을 의미한다. 가장 멀리 있는 참조 화면에서 움직임 예측에 소모되는 시간을 효과적으로 줄이기 위해서 시간적 상관도를 이용한다. 부호화하고자 하는 화면의 시간적 상관도를 이용하여 결정된 탐색 영역을 참조 화면에 따라 적응적으로 줄인다. 이 때, 탐색 영역은 실제 결정된 움직임 벡터와 예측된 움직임 벡터의 차이값을 이용하여 결정된다.

$$MVD_{ref} = |Best_MV_{ref} - Predictive_MV| \quad (2)$$

$$r = \text{median}\{MVD_{ref}\}$$

$$0 \leq ref \leq 2$$

식 (2)는 실제 움직임 벡터와 예측된 움직임 벡터의 차이를 이용하여 각 참조 화면의 탐색 영역을 결정한다. 부호화하고자 하는 매크로블록에서 가장 가까운

첫 번째 참조 화면에서부터 세 번째 참조 화면까지 결정된 움직임 벡터와 예측된 벡터의 차이값들의 중간 값을 이용하여 탐색 영역을 업데이트 한다.

작은 차이값은 현재 부호화하고자 하는 매크로블록의 움직임이 작다는 것과 초기 시작점으로부터 멀리 않은 곳에 최적의 움직임 벡터가 존재한다는 것을 의미한다. 반대로 큰 차이값은 현재 매크로블록의 움직임이 화면 사이에서 크다는 것과 초기 시작점으로부터 멀리 떨어진 곳에서 최적의 움직임 벡터가 존재하는 것을 나타낸다. 그러므로 작은 차이값을 보이는 영역, 즉 움직임 작은 영역에 대해서는 탐색 영역의 크기를 줄인다.

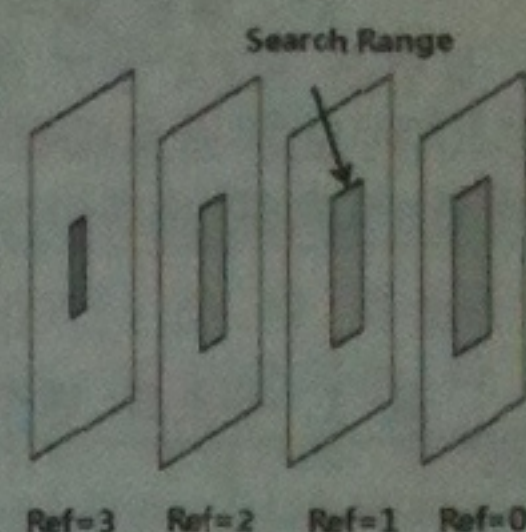


그림 5. 시간적 상관도를 고려한 탐색 영역

그림 5는 시간적 상관도를 이용하여 줄어진 탐색 영역을 보여준다. 현재 부호화하고자 하는 매크로블록에서 멀어질수록 탐색 영역의 크기가 줄어드는 것을 확인할 수 있다.

3. 최종 탐색 영역 결정 방법

공간적 상관도와 시간적 상관도를 동시에 고려하여 최종의 탐색 영역을 결정한다. 참조 화면과 현재 부호화하고자 하는 화면 사이의 거리가 멀어질수록 시간적 상관도를 이용하여 결정된 탐색 영역의 가중치를 높인다. 최종 탐색 영역은 다음 식 (3)을 통해 결정된다.

$$\text{Final_search_range} \quad (3)$$

$$= (R + (\text{Distance} \times r)) / (\text{Distance} + 1)$$

여기서 Distance는 참조 화면과 현재 화면 사이의 거리를 나타내고, R은 공간적 상관도를 고려하여 결정된 탐색 영역을 나타내며, r은 시간적 상관도를 고려하여 결정된 탐색 영역을 의미한다.

식 (3)을 통해 결정된 최종 탐색 영역을 적용하면 멀리 있는 참조 화면에서 불필요한 탐색 지점의 움직임 예측을 줄일 수 있다.

4. 적응적인 초기 시작점 결정 방법

제안한 방법은 공간적 상관도와 시간적 상관도를 고려하여 탐색 영역의 크기를 줄여 불필요한 탐색 지점을 제거한다. 그러므로 실제로 최적의 움직임 벡터로 결정될 수 있는 탐색 지점이 제거되어 움직임 예측의 정확도가 감소한다는 단점이 있다. 움직임 예측의 정확도를 향상시키기 위해, 적응적인 초기 시작점 결정 방법을 제안한다. 움직임 예측의 초기 시작점이 실제 움직임 예측을 통해 결정된 움직임 벡터와 유사할수록 움직임 예측의 정확도는 향상된다.

제안된 초기 시작점은 식 (4)를 통해 결정된다. 제안된 초기 시작점은 예측 움직임 벡터와 이미 이전 참조 화면에서 결정된 최적의 움직임 벡터의 평균으로 결정된다. 제안된 초기 시작점은 연속하는 프레임의 움직임을 추적하므로 움직임 예측의 정확도를 향상시킬 수 있다.

$$\text{Determined_pmv}_{ref} = (\text{pmv} + \text{mv}_{ref-1}) / 2 \quad (4)$$

$$1 \leq ref \leq 3$$

IV. 실험 결과

본 논문에서 실험은 KTA 2.7 소프트웨어[5]를 이용하여 실험했다. KTA 2.7 소프트웨어는 JM 11.0을 기반으로 한다. 자세한 부호화 변수는 표 1과 같다.

표 1. 부호화 변수.

실험 변수	값
Profile	100 (high profile)
QP	22, 27, 32, 37
SearchRange	128
SymbolMode	1 (CABAC)
FrameStructure	IPPP...P
UseIntraMDDT	1
UseExtMB	2 (64×64)
Transform8×8mode	1
SearchMode	Full search
NumberReferenceframes	4

제안된 알고리즘의 성능을 평가하기 위해 1280×720 크기의 영상을 이용하여 실험을 진행했다. 표 2는 기존의 방법과 제안된 방법의 움직임 예측 시간 비교를 나타낸다.

표 2. 움직임 예측 시간 비교.

실험영상	QP	Original	Proposed	△Time (%)
Jets	22	26773.16	12362.24	-53.83
	27	18628.08	7857.58	-57.82
	32	12204.86	5144.02	-57.85
	37	8391.66	3896.35	-53.57
Crew	22	29600.69	13037.08	-55.96
	27	26040.99	99736.00	-61.70
	32	21645.74	81258.28	-62.46
	37	16576.37	63817.24	-61.50
City	22	24219.14	13515.19	-44.20
	27	23706.94	11445.61	-51.72
	32	23326.60	9475.93	-59.38
	37	22010.32	7856.75	-64.30
Night	22	22628.35	12321.41	-45.55
	27	21322.94	10527.71	-50.63
	32	19614.12	88564.16	-54.85
	37	17359.79	73733.32	-57.53
평균				-55.80

표 2를 통해 제안된 방법이 기존의 방법에 비해 움직임 예측 시간을 평균 55.80% 감소함을 확인할 수 있다. 그림 6은 비트율 왜곡 곡선으로 제안하는 방법은 기존의 방법의 성능과 유사함을 확인할 수 있다.

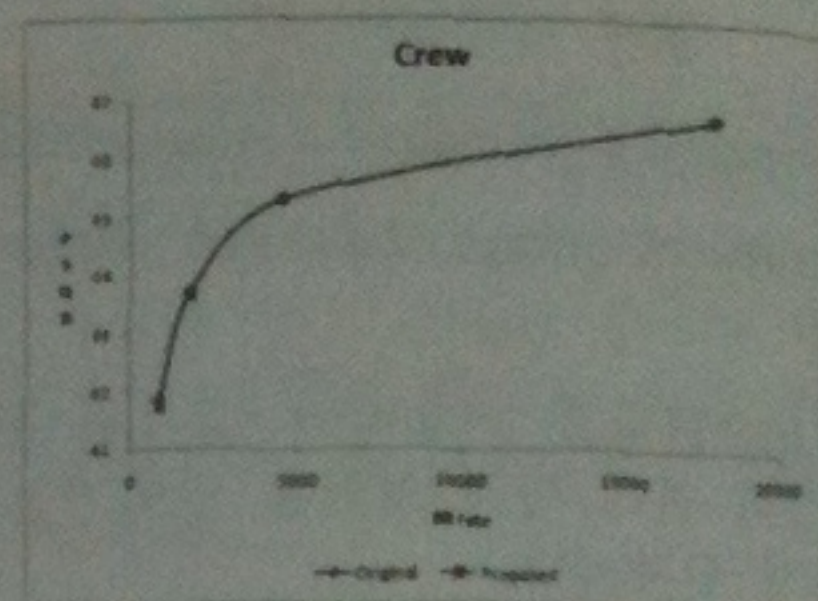


그림 6. 비트율 왜곡 곡선

V. 결론

본 논문에서는 확장된 크기의 매크로블록을 위한 고속 움직임 예측 방법을 제안했다. 부호하고자 하는 매크로블록에 인접하는 블록의 움직임 벡터의 시간적 상관도와 공간적 상관도를 이용하여 불필요한 탐색 지점을 제거하여 움직임 예측에 소요되는 시간을 줄인다. 탐색 지점을 줄여 움직임 예측의 정확도가 감소하는 단점을 보완하기 위해 적응적인 초기 시작점을 결정한다. 제안하는 방법은 기존 방법의 성능은 유지하면서 움직임 예측 시간을 평균 55.80% 줄임을 확인할 수 있다.

감사의 글

본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음. (NIPA-2010-(C1090-1011-0003))

참고문헌

- [1] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 560-576, July 2003.
- [2] Joint Call for Proposals on Video Compression Technology, ISO/IEC MPEG 91st Meeting, Kyoto, Japan, Document N11113, April 2010.
- [3] I. Richardson, H.264 and MPEG-4 Video Compression, Wiley, 2003, pp. 175-177.
- [4] K. Lee, E. Alshina, J. Park, W. Han and J. Min, "Technical consideration for Ad Hoc Group on New Challenges in Video Coding Standardization," ISO/IEC JTC/SC29/WG11 doc.M1558, July 2008, Hannover, Germany
- [5] KTA Software Version 2.7, available at: <http://iphone.hhi.de/suehring/tml/download/KTA/jm11.0kta2.7>.

확장된 매크로블록 위한 고속 움직임 예측 방법

문지희, 호요성
광주과학기술원 정보통신공학부

Fast Motion Estimation for Extended Block Sizes

Ji-Hee Moon, Yo-Sung Ho
Gwangju Institute of Science and Technology (GIST)
{jhmoon, hoyo}@gist.ac.kr

Abstract: H.264/AVC 표준은 16×16 크기부터 4×4 의 다양한 크기의 블록을 이용한 움직임 예측을 수행한다. 움직임을 세밀하게 예측하므로 전송해야 할 정보의 양을 효과적으로 줄일 수 있었다. 그러나 영상의 크기가 커지면서 기존의 16×16 크기의 매크로블록보다 확장된 크기의 매크로블록을 이용하여 움직임을 예측하는 것이 영상 부호화에 더 효율적이다. 하지만 확장된 크기의 매크로블록을 이용한 움직임 예측은 복잡도가 높다는 단점이 있다. 본 논문에서는 움직임 벡터의 상관도를 이용하여 탐색 영역의 크기를 적응적으로 줄이고, 움직임 예측의 정확도를 향상시키기 위해 적응적인 초기 시작점 결정방법을 제안한다. 제안한 방법을 통해 평균 55.80%의 움직임 예측 시간이 감소되었다.

Keywords: motion estimation, extended macroblock

I. 서론

H.264/AVC는 우수한 성능의 비디오 부호화 표준으로 ISO/IEC SC29WG11 산하의 동화상 전문가 그룹(Moving Picture Experts Group, MPEG)과 ITU-T SG16/Q6 동영상 부호화 전문가 그룹(Video Coding Experts Group, VCEG)이 설립한 연합 비디오 팀(Joint Video Team, JVT)에 의해 2003년 5월 제정되었다 [1]. H.264/AVC 표준은 압축 효율을 높이기 위해 기존의 표준과 다른 새로운 기술들을 사용하였다. 그 중에서 블록의 크기를 4×4 단위까지 줄여 움직임 보상을 수행하며 $1/4$ 화소 단위의 움직임 예측을 사용하여 이미 부호화된 이전 화면으로부터 움직임을 세밀하게 추정하였다.

최근 HD 및 UHD와 같은 고해상도, 고품질 영상에 대한 수요가 다양한 응용분야에서 급속히 증가하고 있다. 이러한 고해상도, 고품질 비디오는 데이터의 양이 방대하기 때문에 H.264/AVC의 압축 표준으로도 전송 또는 저장 비용이 증가하는 문제점이 있다. 이러한 문제를 해결하기 위해 MPEG과 VCEG은 2010년 1월에 협력팀(Joint Collaborative Team on Video Coding, JCT-VC)을 결성하고 공동으로 차세대 비디오 부호화 표준화(High Efficiency Video Coding, HEVC)를 시작하기로 합의하였다. HEVC는 H.264/AVC 표준보다 2배 이상의 압축률을 목표로 하고 있다 [2].

높은 압축률 향상을 위해 여러 가지 기술들이 제안되었다. 그 중에서 H.264/AVC에서 사용한 16×16 크기의 매크로블록보다 확장된 크기의 매크로블록을 사용

한 부호화 방법이 제안되었다. 일반적으로 고해상도의 영상에서는 화소 간의 상관도가 저해상도의 영상보다 높기 때문에 기존의 매크로블록의 사이즈보다 큰 블록을 이용하는 것이 영상 부호화에 더 효율적이다.

그러나 확장된 크기의 블록을 이용한 움직임 예측 방법은 부호화 효율을 향상시키지만, 계산량이 높다는 단점이 있다. 영상의 크기와 블록의 크기가 증가하면서 최적의 움직임 벡터를 찾기 위해 계산해야 할 화소의 수가 증가하기 때문이다. 그러므로, 본 논문에서는 불필요한 계산을 줄이기 위해 인접하는 블록들의 움직임 벡터의 상관도를 이용하여 확장된 크기의 매크로블록을 위한 고속 움직임 예측 방법을 제안한다. 또한, 적응적인 초기 시작점 결정 방법을 제안하여 움직임 예측의 정확도를 향상시킨다.

II. 기존의 방법

1. H.264/AVC 전 영역 움직임 탐색 방법

H.264/AVC에서 제공하는 전 영역 움직임 탐색 방법은 그림 1과 같이 이루어진다. 현재 부호화하고자 하는 매크로블록의 움직임 벡터를 예측하기 위해 움직임 벡터 간의 공간적 상관도를 이용한다. 인접하는 블록에서 정해진 움직임 벡터들의 중간 값을 수평 방향과 수직 방향으로 독립적으로 구한다 [3].

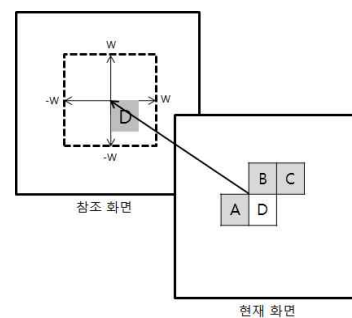


그림 1. H.264/AVC 전 영역 탐색 방법

예측된 움직임 벡터는 참조 화면에서 움직임 예측 시 초기 시작점으로 사용된다. 탐색 영역을 w 로 정했다면 예측된 초기 시작점을 기준으로 하여 상하좌우로 w 길이만큼을 탐색 영역이 정해진다. 정화소 단위로 탐색 영역의 크기는 $(2w+1)^2$ 이 된다. 정화소 단위에서 움직임 예측 과정이 진행되어 결정된 최적의 정화소를 중심으로 부화소 단위로 움직임 예측 과정을 수행한다.

2. 확장된 크기의 매크로블록

기존의 H.264/AVC는 정확한 움직임 예측을 위해 5개의 인터 모드를 지원한다. SKIP, 16×16 , 16×8 , 8×16 과 $P8 \times 8$ 모드가 매크로블록 부호화에 이용된다. $P8 \times 8$ 모드는 각각의 8×8 블록 안에서 8×8 , 8×4 , 4×8 , 4×4 중의 하나로 선택된다. Key technology area (KTA)에서는 기존의 16×16 크기의 매크로블록보다 확장된 크기의 매크로블록을 이용하여 움직임 예측을 수행하고 부호화한다. 그림 2는 KTA 소프트웨어에 구현된 확장된 크기의 움직임 예측 블록 단위를 나타낸다.

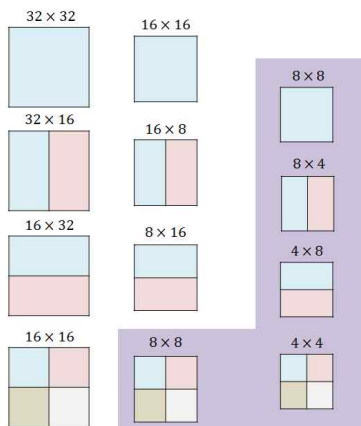


그림 2. KTA의 확장된 매크로블록 인터 모드

각각의 32×32 블록은 H.264/AVC의 매크로블록 부호화와 유사한 방법으로 부호된다. mb32_skip_flag는 H.264/AVC의 SKIP 또는 Direct 모드와 동일한 방법으로 부호된다. H.264/AVC의 움직임 예측 단위인 8×16 또는 16×8 의 mb_type을 부호하는 방법은 32×16 , 16×32 의 mb_type을 부호하는 방법에 적용된다. 예를 들어, H.264/AVC에서 mb_type이 3인 값을 가진 매크로블록은 인터 모드 8×16 를 나타낸다. 그러나 KTA에서는 mb_type이 3인 값을 가진 매크로블록은 인터 모드 16×32 를 나타낸다. mb32_type은 H.264/AVC의 mb_type과 동일한 선택스 요소를 공유하며 유사한 방법으로 부호된다 [4].

확장된 크기의 매크로블록의 움직임 벡터는 기존 H.264/AVC와 동일하게 움직임 예측 블록 당 결정되고 부호된다. 또한 잔여 영상 계수들의 Coded Block Pattern(CBP) 정보를 나타내는 cbp32 선택스 요소도 확장된 크기의 매크로블록 부호화에 포함된다. 만약 cbp32가 1의 값을 가지면 전체 32×32 블록에서 0이 아닌 계수가 한 개라도 존재하는 것을 의미하고 0의 값을 가지면 32×32 블록 안에 0보다 큰 계수가 존재하지 않음을 의미한다.

만약 32×32 블록이 4개의 16×16 블록으로 나누어지면, 4개의 16×16 블록은 raster scan 순서로 부호화되고 각각의 선택스 요소는 기존의 H.264/AVC JM reference software를 따른다. 그러나 delta QP는 전체 32×32 블록에 대해 한 번 전송된다. 즉, 32×32 블록은 모두 같은 QP를 사용한다. 각각의 16×16 블록은 쿼드 트리 방법으로 16×16 블록부터 4×4 블록까지 분할된다. cbp와 잔여 영상의 계수는 H.264/AVC의 선택스

요소와 동일한다.

32×32 블록보다 더 큰 사이즈의 64×64 의 매크로블록을 부호화하는 방법은 위에서 설명한 32×32 블록 부호화 방법과 동일하다.

III. 제안하는 방법

본 논문에서 제안하는 알고리즘은 크게 두 가지 방법이다. 움직임 벡터의 공간적 상관도와 시간적 상관도를 이용하여 탐색 영역을 적응적으로 결정하는 방법과 예측 움직임 벡터와 이미 결정된 움직임 벡터를 이용하여 움직임 예측의 정확도 향상을 위한 적응적인 초기 시작점 결정 방법이다. 본 논문에서는 제안하는 알고리즘에 대해 설명한다.

1. 공간적 상관도를 이용하는 방법

한 화면에 위치하는 인접한 화소 사이에는 높은 상관도가 존재하는데, 이는 움직임 벡터에서도 마찬가지이다. 그러므로 인접하는 매크로블록의 움직임 크기에도 공간적 상관도가 높다. 기존의 움직임 예측 방식은 고정된 정사각형 모양의 탐색 영역 내에서 수행되므로 불필요한 위치를 탐색하여 계산량이 많다. 이러한 불필요한 연산을 줄이기 위해 인접하는 블록 사이의 움직임 벡터의 통계적 특성을 이용하여 수평, 수직 방향으로 각각의 움직임 예측 탐색 영역을 가변적으로 결정한다.

움직임 벡터의 통계적 특성을 이용하고자 인접하는 블록의 움직임 벡터들의 표준편차를 계산한다. 표준편차는 움직임 벡터의 수직방향과 수평방향의 움직임의 크기 정도를 의미한다. 예를 들어 수평방향의 표준편차가 수직방향의 표준편차보다 크면 현재 부호화하고자 하는 매크로블록 주변에 수평방향의 움직임이 크다는 것을 알 수 있다. 이를 이용하여 수평방향, 수직방향에 대해 각각 서로 다른 크기의 탐색 영역을 결정하게 된다. 수직방향과 수평방향 움직임 벡터의 표준편차를 이용하여 적정한 영역 내에서 움직임 예측을 수행함으로써 계산량 절감이 가능하다. 인접하는 블록들의 움직임 벡터의 표준편차를 이용하여 결정된 탐색 영역의 개념을 그림 3에 나타낸다.

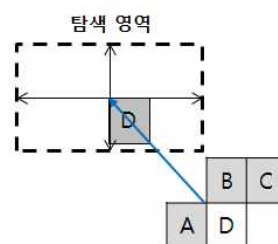


그림 3. 제안된 적응적인 탐색 영역

기존 방법의 성능을 유지하기 위해 결정된 탐색 영역은 최적의 움직임 벡터로 결정될 가능성이 있는 지점을 포함해야 한다. 본 논문에서는 기존의 전 영역 탐색 방법의 성능을 보장하기 위하여 비선형 함수를 이용하여 탐색 영역을 결정한다. 움직임 벡터의 표준편차 값이 지정된 탐색 영역의 크기를 넘어갔을 때 비선형 함수를 사용하여 최대 탐색 영역의 값을 가지도록 조절할 수 있다. 또한 단순히 표준편차 값을 탐색 영역의 크기

로 결정하는 방법보다 비선형적인 값으로 탐색 영역의 크기를 조정하여 움직임 벡터의 표준편차 값보다 더 큰 값을 탐색 영역의 크기로 결정하여 움직임 예측 성능을 보장할 수 있다. 식 (1)은 표준편차를 이용하여 수평방향과 수직방향의 탐색 영역이 결정되는 식을 나타낸다.

$$R = \max(1, \alpha \sqrt{\text{standard_deviation}(MV_s)}) \quad (1)$$

$$\alpha = \text{search_range} / 2\sqrt{\text{search_range}}$$

여기서 α 는 표준편차를 이용하여 결정된 탐색 영역이 사용자가 설정한 최대 탐색 영역을 넘어가는 것을 막기 위한 크기 조정 인자이다.

2. 시간적 상관도를 이용하는 방법

공간적인 특성을 이용하여 결정된 탐색 영역은 모든 참조 화면에서 동일한 크기로 적용된다. 그러나 실제 최적의 참조 화면은 현재 부호화하고자 하는 매크로블록에서 가장 가까운 화면으로 선택될 확률이 높다. 최적의 참조 화면이 선택되는 빈도를 그림 4에 나타낸다.

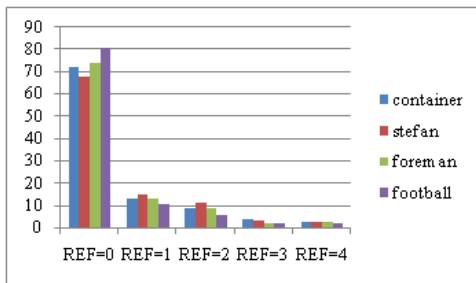


그림 4. 최적의 참조 화면 분포도

그림 4를 보면, 현재 부호화하고자 하는 매크로블록에서 가장 가까운 참조 화면부터 세 번째 참조 화면이 최적의 참조 화면으로 선택될 확률이 95% 이상임을 확인할 수 있다. 이는 부호화하고자 하는 매크로블록에서 가장 멀리 있는 참조 화면에서 동일한 움직임 예측 과정을 수행하여 불필요한 계산량이 증가하는 것을 의미한다. 가장 멀리 있는 참조 화면에서 움직임 예측에 소모되는 시간을 효과적으로 줄이기 위해서 시간적 상관도를 이용한다. 부호화하고자 하는 화면의 시간적 상관도를 이용하여 결정된 탐색 영역을 참조 화면에 따라 적응적으로 줄인다. 이 때, 탐색 영역은 실제 결정된 움직임 벡터와 예측된 움직임 벡터의 차이값을 이용하여 결정된다.

$$MVD_{ref} = |Best_MV_{ref} - Predictive_MV| \quad (2)$$

$$r = \text{median}\{MVD_{ref}\}$$

$$0 \leq ref \leq 2$$

식 (2)는 실제 움직임 벡터와 예측된 움직임 벡터의 차이를 이용하여 각 참조 화면의 탐색 영역을 결정한다. 부호화하고자 하는 매크로블록에서 가장 가까운

첫 번째 참조 화면에서부터 세 번째 참조 화면까지 결정된 움직임 벡터와 예측된 벡터의 차이값들의 중간 값을 이용하여 탐색 영역을 업데이트 한다.

작은 차이값은 현재 부호화하고자 하는 매크로블록의 움직임이 작다는 것과 초기 시작점으로부터 멀지 않은 곳에 최적의 움직임 벡터가 존재한다는 것을 의미한다. 반대로 큰 차이값은 현재 매크로블록의 움직임이 화면 사이에서 크다는 것과 초기 시작점으로부터 멀리 떨어진 곳에서 최적의 움직임 벡터가 존재하는 것을 나타낸다. 그러므로 작은 차이값을 보이는 영역, 즉 움직임 작은 영역에 대해서는 탐색 영역의 크기를 줄인다.

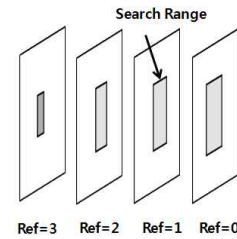


그림 5. 시간적 상관도를 고려한 탐색 영역

그림 5는 시간적 상관도를 이용하여 줄어든 탐색 영역을 보여준다. 현재 부호화하고자 하는 매크로블록에서 멀어질수록 탐색 영역의 크기가 줄어드는 것을 확인할 수 있다.

3. 최종 탐색 영역 결정 방법

공간적 상관도와 시간적 상관도를 동시에 고려하여 최종의 탐색 영역을 결정한다. 참조 화면과 현재 부호화하고자 하는 화면 사이의 거리가 멀어질수록 시간적 상관도를 이용하여 결정된 탐색 영역의 가중치를 높인다. 최종 탐색 영역은 다음 식 (3)을 통해 결정된다.

$$Final_search_range = (R + (Distance \times r)) / (Distance + 1) \quad (3)$$

여기서 Distance는 참조 화면과 현재 화면 사이의 거리를 나타내고, R은 공간적 상관도를 고려하여 결정된 탐색 영역을 나타내며, r은 시간적 상관도를 고려하여 결정된 탐색 영역을 의미한다.

식 (3)을 통해 결정된 최종 탐색 영역을 적용하면 멀리 있는 참조 화면에서 불필요한 탐색 지점의 움직임 예측을 줄일 수 있다.

4. 적응적인 초기 시작점 결정 방법

제안한 방법은 공간적 상관도와 시간적 상관도를 고려하여 탐색 영역의 크기를 줄여 불필요한 탐색 지점을 제거한다. 그러므로 실제로 최적의 움직임 벡터로 결정될 수 있는 탐색 지점이 제거되어 움직임 예측의 정확도가 감소한다는 단점이 있다. 움직임 예측의 정확도를 향상시키기 위해, 적응적인 초기 시작점 결정 방법을 제안한다. 움직임 예측의 초기 시작점이 실제 움직임 예측을 통해 결정된 움직임 벡터와 유사할수록 움직임 예측의 정확도는 향상된다.

제안된 초기 시작점은 식 (4)를 통해 결정된다. 제안된 초기 시작점은 예측 움직임 벡터와 이미 이전 참조 화면에서 결정된 최적의 움직임 벡터의 평균으로 결정된다. 제안된 초기 시작점은 연속하는 프레임의 움직임을 추적하므로 움직임 예측의 정확도를 향상시킬 수 있다.

$$\text{Determined_pmv}_{ref} = (\text{pmv} + \text{mv}_{ref-1}) / 2 \quad (4)$$

$$1 \leq ref \leq 3$$

IV. 실험 결과

본 논문에서 실험은 KTA 2.7 소프트웨어[5]를 이용하여 실험했다. KTA 2.7 소프트웨어는 JM 11.0을 기반으로 한다. 자세한 부호화 변수는 표 1과 같다.

표 1. 부호화 변수.

실험 변수	값
Profile	100 (high profile)
QP	22, 27, 32, 37
SearchRange	128
SymbolMode	1 (CABAC)
FrameStructure	IPPP...P
UseIntraMDDT	1
UseExtMB	2 (64×64)
Transform8×8mode	1
SearchMode	Full search
NumberReferenceframes	4

제안된 알고리즘의 성능을 평가하기 위해 1280×720 크기의 영상을 이용하여 실험을 진행했다. 표 2는 기존의 방법과 제안된 방법의 움직임 예측 시간 비교를 나타낸다.

표 2. 움직임 예측 시간 비교.

실험영상	QP	Original	Proposed	△Time (%)
Jets	22	26773.16	12362.24	-53.83
	27	18628.08	7857.58	-57.82
	32	12204.86	5144.02	-57.85
	37	8391.66	3896.35	-53.57
Crew	22	29600.69	13037.08	-55.96
	27	26040.99	99736.00	-61.70
	32	21645.74	81258.28	-62.46
	37	16576.37	63817.24	-61.50
City	22	24219.14	13515.19	-44.20
	27	23706.94	11445.61	-51.72
	32	23326.60	9475.93	-59.38
	37	22010.32	7856.75	-64.30
Night	22	22628.35	12321.41	-45.55
	27	21322.94	10527.71	-50.63
	32	19614.12	88564.16	-54.85
	37	17359.79	73733.32	-57.53
평균				-55.80

표 2를 통해 제안된 방법이 기존의 방법에 비해 움직임 예측 시간을 평균 55.80% 감소함을 확인할 수 있다. 그림 6은 비트율 왜곡 곡선으로 제안하는 방법은 기존 방법의 성능과 유사함을 확인할 수 있다.

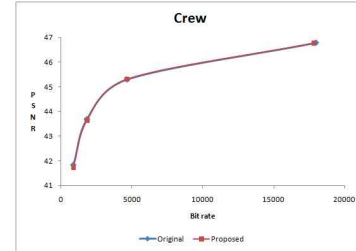


그림 6. 비트율 왜곡 곡선

V. 결론

본 논문에서는 확장된 크기의 매크로블록을 위한 고속 움직임 예측 방법을 제안했다. 부호하고자 하는 매크로블록에 인접하는 블록의 움직임 벡터의 시간적 상관도와 공간적 상관도를 이용하여 불필요한 탐색 지점을 제거하여 움직임 예측에 소요되는 시간을 줄인다. 탐색 지점을 줄여 움직임 예측의 정확도가 감소하는 단점을 보완하기 위해 적응적인 초기 시작점을 결정한다. 제안하는 방법은 기존 방법의 성능은 유지하면서 움직임 예측 시간을 평균 55.80% 줄임을 확인할 수 있다.

감사의 글

본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음. (NIPA-2010-(C1090-1011-0003))

참고문헌

- [1] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 560-576, July 2003.
- [2] Joint Call for Proposals on Video Compression Technology, ISO/IEC MPEG 91st Meeting, Kyoto, Japan, Document N11113, April 2010.
- [3] I. Richardson, H.264 and MPEG-4 Video Compression, Wiley, 2003, pp. 175-177.
- [4] K. Lee, E. Alshina, J. Park, W. Han and J. Min, "Technical consideration for Ad Hoc Group on New Challenges in Video Coding Standardization," ISO/IEC JTC/SC29/WG11 doc.M1558, July 2008, Hannover, Germany
- [5] KTA Software Version 2.7, available at: <http://iphone.hhi.de//suehring/tml/download/KTA/jm11.0kta2.7>.