

(12) 특허협력조약에 의하여 공개된 국제출원

(19) 세계지식재산권기구  
국제사무국

(43) 국제공개일  
2012년 8월 16일 (16.08.2012)



(10) 국제공개번호  
WO 2012/108701 A2

- (51) 국제특허분류:  
H04N 7/32 (2006.01)
- (21) 국제출원번호: PCT/KR2012/000965
- (22) 국제출원일: 2012년 2월 9일 (09.02.2012)
- (25) 출원언어: 한국어
- (26) 공개언어: 한국어
- (30) 우선권정보:  
61/440,872 2011년 2월 9일 (09.02.2011) US  
61/443,261 2011년 2월 16일 (16.02.2011) US  
61/445,539 2011년 2월 23일 (23.02.2011) US  
61/448,634 2011년 3월 2일 (02.03.2011) US  
61/451,124 2011년 3월 10일 (10.03.2011) US  
61/473,153 2011년 4월 8일 (08.04.2011) US  
61/478,055 2011년 4월 21일 (21.04.2011) US  
61/478,914 2011년 4월 25일 (25.04.2011) US
- (71) 출원인 (US 을(를) 제외한 모든 지정국에 대하여): **엘지 전자 주식회사 (LG ELECTRONICS INC.)** [KR/KR]; 서울시 영등포구 여의도동 20, 150-721 Seoul (KR).
- (72) 발명자: **경**
- (75) 발명자/출원인 (US 에 한하여): **박준영 (PARK, Joonyoung)** [KR/KR]; 서울시 서초구 양재동 221 엘지 전자, Convergence R&D 연구소, 137-130 Seoul (KR). **박승욱 (PARK, Seungwook)** [KR/KR]; 서울시 서초구 양

재동 221 엘지전자, Convergence R&D 연구소, 137-130 Seoul (KR). **임재현 (LIM, Jaehyun)** [KR/KR]; 서울시 서초구 양재동 221 엘지전자, Convergence R&D 연구소, 137-130 Seoul (KR). **김정선 (KIM, Jungsun)** [KR/KR]; 서울시 서초구 양재동 221 엘지전자, Convergence R&D 연구소, 137-130 Seoul (KR). **최영희 (CHOI, Younghee)** [KR/KR]; 서울시 서초구 양재동 221 엘지전자, Convergence R&D 연구소, 137-130 Seoul (KR). **성재원 (SUNG, Jaewon)** [KR/KR]; 서울시 서초구 양재동 221 엘지전자, Convergence R&D 연구소, 137-130 Seoul (KR). **전병문 (JEON, Byeongmoon)** [KR/KR]; 서울시 서초구 양재동 221 엘지전자, Convergence R&D 연구소, 137-130 Seoul (KR). **전용준 (JEON, Yongjoon)** [KR/KR]; 서울시 서초구 양재동 221 엘지전자, Convergence R&D 연구소, 137-130 Seoul (KR).

(74) 대리인: **양문옥 (YANG, Moon Ock)**; 서울시 강남구 역삼동 735-10 삼흥역삼빌딩 2층 에센특허법률사무소, 135-080 Seoul (KR).

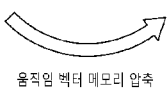
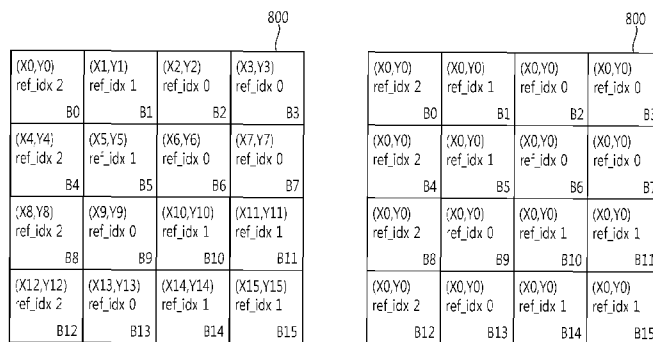
(81) 지정국 (별도의 표시가 없는 한, 가능한 모든 종류의 국내 권리의 보호를 위하여): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM,

[다음 쪽 계속]

(54) Title: METHOD FOR STORING MOTION INFORMATION AND METHOD FOR INDUCING TEMPORAL MOTION VECTOR PREDICTOR USING SAME

(54) 발명의 명칭: 움직임 정보 저장 방법 및 이를 이용한 시간적 움직임 벡터 예측자 유도 방법

[Fig. 8]

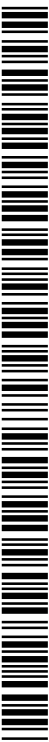


AA ... motion vector memory compression

(57) Abstract: The present invention relates to a method for storing motion information and a method for inducing a temporal motion vector predictor (TMVP) using the method, and the method for storing the motion information according to the present invention comprises the steps of: deciding a representative block from the blocks within a motion information storage unit in a picture having an identical location as a current picture; and storing the motion information of the representative block as the motion information for the motion information storage unit, wherein the motion information of the representative block can be used as the motion information for the blocks included in the motion information storage unit.

(57) 요약서:

[다음 쪽 계속]



WO 2012/108701 A2



PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

(84) **지정국** (별도의 표시가 없는 한, 가능한 모든 종류의 역내 권리의 보호를 위하여): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), 유라시아 (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), 유럽 (AL, AT, BE, BG, CH, CY, CZ, DE, DK,

**공개:**

— 국제조사보고서 없이 공개하며 보고서 접수 후 이를 별도 공개함 (규칙 48.2(g))

본 발명은 움직임 정보 저장 방법 및 이를 이용한 TMVP 유도 방법에 관한 것으로서, 본 발명에 따른 움직임 정보 저장 방법은 현재 픽처에 대한 동일 위치 픽처에서 움직임 정보 저장 단위 내 블록들 중 대표 블록을 결정하는 단계 및 상기 대표 블록의 움직임 정보를 상기 움직임 정보 저장 단위에 대한 움직임 정보로서 저장하는 단계를 포함하며, 상기 움직임 정보 저장 단위에 포함되는 블록들의 움직임 정보로서 상기 대표 블록의 움직임 정보를 이용할 수 있다.

## 명세서

### 발명의 명칭: 움직임 정보 저장 방법 및 이를 이용한 시간적 움직임 벡터 예측자 유도 방법

#### 기술분야

[0001] 본 발명은 비디오 데이터 압축 기술에 관한 것으로서 더 구체적으로는 화면 간 예측(인터 예측)에 있어서 시간적 움직임 벡터 예측자를 유도하는 방법과 시간적 움직임 벡터 예측자를 유도하는데 필요한 정보를 저장하는 방법에 관한 것이다.

#### 배경기술

[0002] 최근, 고해상도, 고품질의 영상에 대한 요구가 다양한 응용 분야에서 증가하고 있다. 하지만, 영상의 고해상도, 고품질이 될수록 해당 영상에 관한 정보량도 함께 증가한다. 따라서 기존의 유무선 광대역 회선과 같은 매체를 이용하여 영상 정보를 전송하거나 기존의 저장 매체를 이용해 영상 정보를 저장하는 경우, 정보의 전송 비용과 저장 비용이 증가하게 된다. 따라서, 고해상도, 고품질 영상의 정보를 효과적으로 전송하거나 저장하고, 재생하기 위해 고효율의 영상 압축 기술을 이용할 수 있다.

[0003] 영상 압축의 효율을 높이기 위해, 화면 간 예측과 화면 내 예측을 이용할 수 있다.

[0004] 인터(inter) 예측에서는 시간적으로 이전 및/또는 이후의 픽처로부터 현재 픽처에 포함된 화소값을 예측하며, 인트라(intra) 예측에서는 현재 픽처 내의 화소 정보를 이용하여 현재 픽처에 포함된 화소값을 예측한다.

[0005] 인터 예측 및 인트라 예측 외에도, 조명 변화 등에 의한 화질의 열화를 방지하기 위한 가중치 예측 기술, 출현 빈도가 높은 심볼(symbol)에 짧은 부호를 할당하고 출현 빈도가 낮은 심볼에 긴 부호를 할당하는 엔트로피 부호화 기술 등이 영상 정

[0006] 보의 효율적인 처리를 위해 이용되고 있다.

#### 발명의 요약

##### 기술적 과제

[0007] 본 발명의 기술적 목적은 효과적인 예측을 수행하기 위한 방법 및 장치를 제공하는 데 있다.

[0008] 본 발명의 다른 기술적 목적은 인터 예측에 있어서, 시간적 움직임 벡터 예측자를 유도하는 경우에 왜곡을 방지하고 압축 효율을 높이는 방법 및 장치를 제공하는데 있다.

[0009] 본 발명의 또 다른 기술적 목적은 시간적 움직임 벡터 예측자를 유도하기 위한 정보를 저장함에 있어서 메모리의 용량을 절감하는 방법 및 장치를 제공하는데 있다.

[0010] 본 발명의 또 다른 기술적 목적은 시간적 움직임 벡터 예측자를 유도하기 위한

정보를 저장함에 있어서 계산량을 줄이고 복잡도를 경감하는 방법 및 장치를 제공하는데 있다.

### 과제 해결 수단

- [0011] (1) 본 발명의 일 실시형태는 움직임 정보 저장 방법으로서 현재 픽처에 대한 동일 위치 픽처에서 움직임 정보 저장 단위 내 블록들 중 대표 블록을 결정하는 단계 및 상기 대표 블록의 움직임 정보를 상기 움직임 정보 저장 단위에 대한 움직임 정보로서 저장하는 단계를 포함하며, 상기 움직임 정보 저장 단위에 포함되는 블록들의 움직임 정보로서 상기 대표 블록의 움직임 정보를 이용할 수 있다.
- [0012] (2) (1)에서, 상기 대표 블록의 움직임 정보는 움직임 벡터와 참조 인덱스를 포함할 수 있으며, 상기 움직임 정보 저장 단위에 포함되는 블록들의 움직임 벡터와 참조 인덱스로서 상기 대표 블록의 움직임 벡터와 참조 인덱스를 이용할 수 있다.
- [0013] (3) (2)에서, 상기 움직임 정보 저장 단위에 포함되는 블록들 중 인트라 예측 모드로 부호화된 블록들에 대해서는 인트라 블록임을 지시할 수 있다.
- [0014] (4) (3)에서, 상기 움직임 정보 저장 단위에 포함되는 블록들 중 인트라 예측 모드로 부호화된 블록들에 대해서는 상기 대표 블록의 참조 인덱스 대신 인트라 블록임을 지시하는 참조 인덱스를 할당할 수 있다.
- [0015] (5) (3)에서, 상기 움직임 정보 저장 단위에 포함되는 블록들 중 인트라 예측 모드로 부호화된 블록들은 코딩 블록의 예측 타입에 의해 인트라 블록임을 지시될 수 있다.
- [0016] (6) (1)에서, 상기 대표 블록의 움직임 정보는 움직임 벡터, 참조 인덱스 및 예측 모드를 포함할 수 있으며, 상기 움직임 정보 저장 단위에 포함되는 블록들의 움직임 벡터, 참조 인덱스 및 예측 모드로서 상기 대표 블록의 움직임 벡터, 참조 인덱스 및 예측 모드를 이용할 수 있다.
- [0017] (7) (1)에서, 상기 대표 블록의 움직임 정보는 움직임 벡터, 참조 인덱스 및 예측 모드를 포함할 수 있으며, 상기 움직임 정보 저장 단위에 포함되는 블록들의 움직임 벡터, 참조 인덱스 및 예측 모드로서 상기 대표 블록의 움직임 벡터, 참조 인덱스 및 예측 모드를 이용하되, 상기 움직임 정보 저장 단위에 포함되는 블록들 중 인트라 예측 모드로 부호화된 블록에 대해서는 인트라 예측 모드임을 지시할 수 있다.
- [0018] (8) (1)에서, 상기 대표 블록은 상기 움직임 정보 저장 단위에 속하는 블록들 중 좌상단의 블록일 수 있다.
- [0019] (9) (1)에서, 상기 대표 블록은, 상기 움직임 정보 저장 블록 내의 블록들을 소정의 검색 순서에 따라서 검색하여 가장 먼저 검색한 인트라 블록일 수 있다.
- [0020] (10) (9)에서, 상기 검색 순서는 상기 움직임 정보 저장 단위 내 좌상단 블록에서부터 시작되는 래스터 스캔 순서일 수 있다.

- [0021] (11) (9)에서, 상기 검색 순서는 상기 움직임 정보 저장 단위 내 중앙의 블록에서부터 나선 방향으로 검색해 나가는 순서일 수 있다.
- [0022] (12) (1)에서, 상기 대표 블록은, 상기 움직임 정보 저장 단위에 속하는 블록들을 소정의 검색 순서에 따라서 검색하여 가장 먼저 검색되는 인터 블록으로서 참조 인덱스의 값이 0인 블록일 수 있다.
- [0023] (13) (1)에서, 상기 움직임 정보 저장 단위 내의 모든 블록들이 인트라 예측 모드로 부호화된 블록들인 경우에는, 상기 움직임 정보 저장 단위에 인접한 블록들 중에서 인트라 예측 모드로 부호화된 블록을 상기 대표 블록으로 선택할 수 있다.
- [0024] (14) (1)에서, 상기 대표 블록은, 상기 움직임 정보 저장 단위의 중앙에 위치하는 네 블록 중에서 우하(Right-Bottom)에 위치하는 블록일 수 있다.
- [0025] (15) (1)에서, 상기 대표 블록은, 상기 움직임 정보 저장 단위의 인접 블록 중 상기 움직임 정보 저장 단위의 우하측 코너에 인접하는 블록일 수 있다.
- [0026] (16) (1)에서, 상기 움직임 정보 저장 단위의 사이즈는 부호화기에서 결정되어 복호화기로 시그널링될 수 있다.
- [0027] (17) (1)에서, 픽처 경계에 위치하는 경계 움직임 정보 저장 단위의 크기가 상기 픽처 내부에 위치하는 내부 움직임 정보 저장 단위보다 작고, 상기 내부 움직임 정보 저장 단위에 대한 대표 블록의 위치와 동일한 위치의 블록을 상기 경계 움직임 정보 저장 단위의 대표 블록으로 이용할 수 없는 경우에는, 상기 경계 움직임 정보 저장 단위 내의 블록들 중에서 좌상단에 위치하는 블록을 상기 경계 움직임 정보 저장 단위에 대한 대표 블록으로 결정할 수 있다.
- [0028] (18) 본 발명의 다른 실시형태는 움직임 정보 저장 방법을 이용한 시간적 움직임 벡터 예측자 유도 방법으로서, 현재 픽처의 동일 위치 픽처를 결정하는 단계, 상기 동일 위치 픽처에서 현재 블록에 대한 동일 위치 예측 유닛을 결정하는 단계, 상기 동일 위치 예측 유닛의 움직임 정보를 상기 현재 블록에 대한 시간적 움직임 벡터 예측자로 설정하는 단계를 포함하며, 상기 동일 위치 예측 유닛은, 상기 동일 위치 예측 유닛에 대한 위치 결정 화소가 속하는 움직임 정보 저장 단위 내 좌상단에 위치하는 예측 유닛이고, 상기 위치 결정 화소는 상기 동일 위치 픽처에서 현재 블록에 대한 동일 위치 블록의 우하단 코너에 인접하는 화소일 수 있다.
- [0029] (19) (18)에 있어서, 상기 동일 위치 블록의 우하단 코너에 인접하는 화소가 인터 예측에 이용할 수 없는 화소인 경우에는, 상기 동일 위치 블록의 중앙에 위치하는 네 화소 중 우하단의 화소를 위치 결정 화소로 선택할 수 있다.
- [0030] (20) (18)에 있어서, 동일 위치 예측 유닛은 상기 움직임 정보 저장 단위의 중앙에 위치하는 예측 유닛 또는 상기 움직임 정보 저장 단위의 우하단 코너에 인접하는 예측 유닛 중 어느 하나일 수 있다.

### 발명의 효과

- [0031] 본 발명에 의하면, 영상 데이터의 압축 처리에 있어서 예측 효율을 높이고 부호화/복호화의 성능을 향상시킬 수 있다.
- [0032] 본 발명에 의하면, 인터 예측을 위해 시간적 움직임 벡터 예측자를 유도함에 있어서 왜곡을 방지하고 압축 효율을 높일 수 있다.
- [0033] 본 발명에 의하면, 시간적 움직임 벡터 예측자를 유도하기 위한 정보를 저장함에 있어서 메모리의 용량을 절감할 수 있다.
- [0034] 본 발명에 의하면, 시간적 움직임 벡터 예측자를 유도하기 위한 정보를 저장함에 있어서 계산량을 줄이고 복잡도를 경감할 수 있다.

### 도면의 간단한 설명

- [0035] 도 1은 본 발명의 일 실시예에 따른 영상 부호화 장치(부호화기)를 개략적으로 도시한 블록도이다.
- [0036] 도 2는 본 발명의 일 실시예에 따른 영상 복호화기를 개략적으로 나타낸 블록도이다.
- [0037] 도 3은 인터 예측 모드에서 AMVP가 적용되는 경우, 움직임 벡터를 도출하는 방법에 관한 일 실시예를 개략적으로 설명하는 순서도이다.
- [0038] 도 4는 MVP 후보 리스트 생성 방법에 관한 일 실시예를 개략적으로 설명하는 개념도이다.
- [0039] 도 5는 복호화기에서 인터 예측을 위해 TMVP를 유도하는 방법의 일 예를 개략적으로 설명하는 순서도이다.
- [0040] 도 6은 colPu를 특징하는 방법을 개략적으로 설명하는 도면이다.
- [0041] 도 7은 움직임 정보를 저장하는 방법에 대하여 개략적으로 설명하는 도면이다.
- [0042] 도 8은 참조 인덱스를 고려하지 않고 움직임 벡터 메모리 압축(움직임 벡터 저장 감축)을 수행하는 예를 개략적으로 설명하는 도면이다.
- [0043] 도 9는 본 발명이 적용되는 시스템에서 4x4 개의 움직임 정보 블록을 하나의 움직임 정보 저장 단위로 하는 경우에 각 움직임 정보 블록의 움직임 정보를 개략적으로 표시한 것이다.
- [0044] 도 10은 본 발명에 따라서 움직임 정보 저장 단위 내에서 움직임 정보 블록이 어떤 예측 모드로 코딩되었는지를 검색하는 순서를 개략적으로 설명하는 도면이다.
- [0045] 도 11은 본 발명에 따라서 현재 움직임 정보 저장 단위의 주변 블록들을 이용하여 현재 움직임 정보 저장 단위에 대한 움직임 정보를 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0046] 도 12는 움직임 벡터 메모리 압축이 수행될 때, 움직임 벡터와 함께 참조 인덱스에 대해서도 움직임 정보 메모리 압축이 수행되는 것을 개략적으로 설명하는 도면이다.
- [0047] 도 13은 움직임 벡터 메모리 압축이 수행될 때, 본 발명에 따라서 인터 블록에 대하여만 움직임 벡터와 참조 인덱스의 움직임 정보 메모리 압축이 수행되는

것을 개략적으로 설명하는 도면이다.

- [0048] 도 14는 움직임 벡터 메모리 압축이 수행될 때, 본 발명에 따라서 움직임 벡터와 참조 인덱스의 움직임 정보 메모리 압축이 수행되되, 인트라 블록을 추정하는 방법을 개략적으로 설명하는 도면이다.
- [0049] 도 15는 움직임 벡터 메모리 압축이 수행될 때, 본 발명에 따라서 예측 모드에 관한 정보가 압축되는 것을 개략적으로 설명하는 도면이다.
- [0050] 도 16은 움직임 정보를 압축 저장할 때, 본 발명에 따라서 현재 움직임 정보 저장 단위에 인접한 블록의 정보를 이용하는 방법을 개략적으로 설명하는 도면이다.
- [0051] 도 17은 움직임 정보의 압축 저장을 수행할 때, 본 발명에 따라서 움직임 정보의 압축 저장 단위보다 작은 화소 블록의 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0052] 도 18은 대표 움직임 정보 블록이 움직임 정보 저장 단위의 좌상단 블록인 경우에, 본 발명에 따라서 잔여 블록에 대한 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0053] 도 19는 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 우하단 블록인 경우에, 본 발명에 따라서 잔여 블록에 대한 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0054] 도 20은 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 우하단 블록이지만, 잔여 블록 내에 이용할 수 없는 영역이 존재하는 경우에 잔여 블록의 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0055] 도 21은 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 중앙 좌상단의 블록인 경우에, 본 발명에 따라서 잔여 블록에 대한 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0056] 도 22는 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 중앙 좌상단 블록이지만, 잔여 블록 내에 이용할 수 없는 영역이 존재하는 경우에 잔여 블록의 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0057] 도 23은 현재 프레임에 이웃하는 프레임에서 현재 블록과 동일한 위치에 있는 블록의 좌상단에 위치하는 움직임 정보가 아니라, 중앙에 위치하는 움직임 정보를 TMVP로 이용하는 예를 개략적으로 설명하는 도면이다.
- [0058] 도 24는 TMVP를 유도할 때 중앙에 위치하는 인터 블록의 움직임 정보를 처리하는 방법을 개략적으로 설명하는 도면이다.
- [0059] 도 25는 본 발명에 따라서 참조 픽처에서 col 블록의 중앙에 위치하는 블록의 움직임 정보를 기반으로 TMVP를 유도하는 방법을 개략적으로 설명하는 방법이다.
- [0060] 도 26은 본 발명에 따라서 인트라 블록의 유무를 고려하여 TMVP를 유도하는 방법을 개략적으로 설명하는 도면이다.

## 발명의 실시를 위한 형태

- [0061] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세하게 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니다. 본 명세서에서 사용하는 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명의 기술적 사상을 한정하려는 의도로 사용되는 것은 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성 요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성 요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0062] 한편, 본 발명에서 설명되는 도면상의 각 구성들은 영상 부호화/복호화 장치에서 서로 다른 특징적인 기능들에 관한 설명의 편의를 위해 독립적으로 도시된 것으로서, 각 구성들이 서로 별개의 하드웨어나 별개의 소프트웨어로 구현된다는 것을 의미하지는 않는다. 예컨대, 각 구성 중 두 개 이상의 구성이 합쳐져 하나의 구성을 이룰 수도 있고, 하나의 구성이 복수의 구성으로 나뉘어질 수도 있다. 각 구성이 통합 및/또는 분리된 실시예도 본 발명의 본질에서 벗어나지 않는 한 본 발명의 권리범위에 포함된다.
- [0063] 이하, 첨부한 도면들을 참조하여, 본 발명의 바람직한 실시예를 보다 상세하게 설명하고자 한다. 이하, 도면상의 동일한 구성 요소에 대해서는 동일한 참조부호를 사용하고 동일한 구성 요소에 대해서 중복된 설명은 생략한다.
- [0064] 도 1은 본 발명의 일 실시예에 따른 영상 부호화 장치(부호화기)를 개략적으로 도시한 블록도이다. 도 1을 참조하면, 영상 부호화 장치(100)는 픽처 분할부(105), 예측부(110), 변환부(115), 양자화부(120), 재정렬부(125), 엔트로피 부호화부(130), 역양자화부(135), 역변환부(140), 필터부(145) 및 메모리(150)를 구비한다.
- [0065] 픽처 분할부(105)는 입력된 픽처를 적어도 하나의 처리 단위로 분할할 수 있다. 이때, 처리 단위는 예측 유닛(Prediction Unit, 이하 'PU'라 함)일 수도 있고, 변환 유닛(Transform Unit, 이하 'TU'라 함)일 수도 있으며, 코딩 유닛(Coding Unit, 이하 'CU'라 함)일 수도 있다. 다만, 본 명세서에서는 설명의 편의를 위해, 예측 유닛을 예측 블록, 변환 유닛을 변환 블록, 부호화 유닛을 부호화 블록으로 표현할 수 있다.
- [0066] 예측부(110)는 후술하는 바와 같이, 화면 간 예측을 수행하는 화면 간 예측부와 화면 내 예측을 수행하는 화면 내 예측부를 포함한다. 예측부(110)는, 픽처 분할부(105)에서 픽처의 처리 단위에 대하여 예측을 수행하여 예측 블록을 생성한다. 예측부(110)에서 픽처의 처리 단위는 CU일 수도 있고, TU일 수도

있고, PU일 수도 있다. 또한, 해당 처리 단위에 대하여 실시되는 예측이 화면 간 예측인지 화면 내 예측인지를 결정하고, 각 예측 방법의 구체적인 내용(예컨대, 예측 모드 등)를 정할 수 있다. 이때, 예측이 수행되는 처리 단위와 예측 방법 및 구체적인 내용이 정해지는 처리 단위는 다를 수 있다. 예컨대, 예측의 방법과 예측 모드 등은 PU 단위로 결정되고, 예측의 수행은 TU 단위로 수행될 수도 있다.

- [0067] 화면 간 예측을 통해서는 현재 픽처의 이전 픽처 및/또는 이후 픽처 중 적어도 하나의 픽처의 정보를 기초로 예측을 수행하여 예측 블록을 생성할 수 있다. 또한, 화면 내 예측을 통해서는 현재 픽처 내의 화소 정보를 기초로 예측을 수행하여 예측 블록을 생성할 수 있다.
- [0068] 화면 간 예측에서는 PU에 대하여, 참조 픽처를 선택하고 PU와 동일한 크기의 참조 블록을 정수 화소 샘플 단위로 선택할 수 있다. 이어서, 현재 PU와의 레지듀얼(residual) 신호가 최소화되며 움직임 벡터 크기 역시 최소가 되는 예측 블록을 생성한다. 화면 내 예측의 방법으로서, 스킵(skip) 모드, 머지(merge) 모드, MVP(Motion Vector Prediction) 등을 이용할 수 있다. 예측 블록은 1/2 화소 샘플 단위와 1/4 화소 샘플 단위와 같이 정수 이하 샘플 단위로 생성될 수도 있다. 이때, 움직임 벡터 역시 정수 화소 이하의 단위로 표현될 수 있다. 예컨대 휘도 화소에 대해서는 1/4 화소 단위로, 색차 화소에 대해서는 1/8 화소 단위로 표현될 수 있다.
- [0069] 화면 간 예측을 통해 선택된 참조 픽처의 인덱스, 움직임 벡터(ex. Motion Vector Predictor), 레지듀얼 신호 등의 정보는 엔트로피 부호화되어 복호화기에 전달된다.
- [0070] 화면 내 예측을 수행하는 경우에는, PU 단위로 예측 모드가 정해져서 PU 단위로 예측이 수행될 수 있다. 또한, PU 단위로 예측 모드가 정해지고 TU 단위로 화면 내 예측이 수행될 수도 있다.
- [0071] 화면 내 예측에서 예측 모드는 33개의 방향성 예측 모드와 적어도 2개 이상의 비방향성 모드를 가질 수 있다. 비방향성 모드는 DC 예측 모드 및 플레너 모드(Planar 모드)을 포함할 수 있다.
- [0072] 화면 내 예측에서는 참조 샘플에 필터를 적용한 후 예측 블록을 생성할 수 있다. 이때, 참조 샘플에 필터를 적용할 것인지는 현재 블록의 인트라 예측 모드 및/또는 사이즈에 따라 결정될 수 있다. 이때, 현재 블록은 예측이 수행되는 변환 유닛일 수 있다. 한편, 본 명세서에서 픽셀을 이용한다는 것은 해당 픽셀의 정보, 예컨대 픽셀값 등을 이용한다는 것을 의미한다. 다만, 설명의 편의를 위해, '픽셀의 정보를 이용한다' 혹은 '픽셀 값을 이용한다'는 표현을 '픽셀을 이용한다'고 간단하게 나타낼 수도 있음에 유의한다. 인트라 예측의 구체적인 방법은 후술하는 바와 같다.
- [0073] PU는 다양한 사이즈/형태를 가질 수 있으며, 예컨대 화면 간 예측의 경우에 PU는  $2N \times 2N$ ,  $2N \times N$ ,  $N \times 2N$ , 또는  $N \times N$  등의 크기를 가질 수 있다. 화면 내 예측의

경우에 PU는  $2N \times 2N$  또는  $N \times N$  ( $N$ 은 정수) 등의 크기를 가질 수 있다. 이때,  $N \times N$  크기의 PU는 특정한 경우에만 적용하도록 설정할 수 있다. 예컨대 최소 크기 코딩 유닛에 대해서만  $N \times N$ 의 PU를 이용하도록 정하거나 화면 내 예측에 대해서만 이용하도록 정할 수도 있다. 또한, 상술한 크기의 PU 외에,  $N \times mN$ ,  $mN \times N$ ,  $2N \times mN$  또는  $mN \times 2N$  ( $m < 1$ ) 등의 크기를 가지는 PU를 더 정의하여 사용할 수도 있다.

- [0074] 생성된 예측 블록과 원본 블록 사이의 레지듀얼 값(레지듀얼 블록 또는 레지듀얼 신호)은 변환부(115)로 입력된다. 또한, 예측을 위해 사용한 예측 모드 정보, 움직임 벡터 정보 등은 레지듀얼 값과 함께 엔트로피 부호화부(130)에서 부호화되어 복호화기에 전달된다.
- [0075] 변환부(115)는 변환 단위로 레지듀얼 블록에 대한 변환을 수행하고 변환 계수를 생성한다. 변환부(115)에서의 변환 단위는 TU일 수 있으며, 쿼드 트리(quad tree) 구조를 가질 수 있다. 이때, 변환 단위의 크기는 소정의 최대 및 최소 크기의 범위 내에서 정해질 수 있다. 변환부(115)는 레지듀얼 블록을 DCT(Discrete Cosine Transform) 및/또는 DST(Discrete Sine Transform)를 이용하여 변환할 수 있다.
- [0076] 양자화부(120)는 변환부(115)에서 변환된 레지듀얼 값들을 양자화하여 양자화 계수를 생성할 수 있다. 양자화부(120)에서 산출된 값은 역양자화부(135)와 재정렬부(125)에 제공된다.
- [0077] 재정렬부(125)는 양자화부(120)로부터 제공된 양자화 계수를 재정렬한다. 양자화 계수를 재정렬함으로써 엔트로피 부호화부(130)에서의 부호화의 효율을 높일 수 있다. 재정렬부(125)는 계수 스캐닝(Coefficient Scanning) 방법을 통해 2차원 블록 형태의 양자화 계수들을 1차원의 벡터 형태로 재정렬할 수 있다. 재정렬부(125)에서는 양자화부에서 전송된 계수들의 확률적인 통계를 기반으로 계수 스캐닝의 순서를 변경함으로써 엔트로피 부호화부(130)에서의 엔트로피 부호화 효율을 높일 수도 있다.
- [0078] 엔트로피 부호화부(130)는 재정렬부(125)에 의해 재정렬된 양자화 계수들에 대한 엔트로피 부호화를 수행할 수 있다. 엔트로피 부호화에는 예를 들어, 지수 골롬(Exponential Golomb), CAVLC(Context-Adaptive Variable Length Coding), CABAC(Context-Adaptive Binary Arithmetic Coding)과 같은 부호화 방법을 사용할 수 있다. 엔트로피 부호화부(130)는 재정렬부(125) 및 예측부(110)로부터 전달받은 CU의 양자화 계수 정보 및 블록 타입 정보, 예측 모드 정보, 분할 단위 정보, PU 정보 및 전송 단위 정보, 움직임 벡터 정보, 참조 픽처 정보, 블록의 보간 정보, 필터링 정보 등 다양한 정보를 부호화할 수 있다.
- [0079] 또한, 엔트로피 부호화부(130)는 필요한 경우에, 전송하는 파라미터 셋 또는 신택스에 일정한 변경을 가할 수도 있다.
- [0080] 역양자화부(135)는 양자화부(120)에서 양자화된 값들을 역양자화하고, 역변환부(140)는 역양자화부(135)에서 역양자화된 값들을 역변환한다.

- 역양자화부(135) 및 역변환부(140)에서 생성된 레지듀얼 값은 예측부(110)에서 예측된 예측 블록과 합쳐져 복원 블록(Reconstructed Block)이 생성될 수 있다.
- [0081] 필터부(145)는 더블록킹 필터, ALF(Adaptive Loop Filter), SAO(Sample Adaptive Offset)를 복원된 픽처에 적용할 수 있다.
- [0082] 더블록킹 필터는 복원된 픽처에서 블록 간의 경계에 생긴 블록 왜곡을 제거할 수 있다. ALF(Adaptive Loop Filter)는 더블록킹 필터를 통해 블록이 필터링된 후 복원된 영상과 원래의 영상을 비교한 값을 기초로 필터링을 수행할 수 있다. ALF는 고효율을 적용하는 경우에만 수행될 수도 있다. SAO는 더블록킹 필터가 적용된 레지듀얼 블록에 대하여, 화소 단위로 원본 영상과의 오프셋 차이를 복원하며, 밴드 오프셋(Band Offset), 에지 오프셋(Edge Offset) 등의 형태로 적용된다.
- [0083] 한편, 화면 간 예측에 사용되는 복원 블록에 대해서 필터부(145)는 필터링을 적용하지 않을 수 있다.
- [0084] 메모리(150)는 필터부(145)를 통해 산출된 복원 블록 또는 픽처를 저장할 수 있다. 메모리(150)에 저장된 복원 블록 또는 픽처는 화면 간 예측을 수행하는 예측부(110)에 제공될 수 있다.
- [0085]
- [0086] 도 2는 본 발명의 일 실시예에 따른 영상 복호화기를 개략적으로 나타낸 블록도이다. 도 2를 참조하면, 영상 복호화기(200)는 엔트로피 복호화부(210), 재정렬부(215), 역양자화부(220), 역변환부(225), 예측부(230), 필터부(235) 메모리(240)를 포함할 수 있다.
- [0087] 영상 부호화기에서 영상 비트 스트림이 입력된 경우, 입력된 비트 스트림은 영상 부호화기에서 영상 정보가 처리된 절차에 따라서 복호화될 수 있다.
- [0088] 예컨대, 영상 부호화기에서 엔트로피 부호화를 수행하기 위해 CAVLC 등의 가변 길이 부호화(Variable Length Coding: VLC, 이하 'VLC' 라 함)가 사용된 경우에, 엔트로피 복호화부(210)도 부호화기에서 사용한 VLC 테이블과 동일한 VLC 테이블로 구현하여 엔트로피 복호화를 수행할 수 있다. 또한, 영상 부호화기에서 엔트로피 부호화를 수행하기 위해 CABAC을 이용한 경우에, 엔트로피 복호화부(210)는 이에 대응하여 CABAC을 이용한 엔트로피 복호화를 수행할 수 있다.
- [0089] 엔트로피 복호화부(210)에서 복호화된 정보 중 예측 블록을 생성하기 위한 정보는 예측부(230)로 제공되고 엔트로피 복호화부에서 엔트로피 복호화가 수행된 레지듀얼 값은 재정렬부(215)로 입력될 수 있다.
- [0090] 재정렬부(215)는 엔트로피 복호화부(210)에서 엔트로피 복호화된 비트 스트림을 영상 부호화기에서 재정렬한 방법을 기초로 재정렬할 수 있다. 재정렬부(215)는 1차원 벡터 형태로 표현된 계수들을 다시 2차원의 블록 형태의 계수로 복원하여 재정렬할 수 있다. 재정렬부(215)는 부호화기에서 수행된 계수 스캐닝에 관련된 정보를 제공받고 해당 부호화부에서 수행된 스캐닝 순서에

기초하여 역으로 스캐닝하는 방법을 통해 재정렬을 수행할 수 있다.

- [0091] 역양자화부(220)는 부호화기에서 제공된 양자화 파라미터와 재정렬된 블록의 계수값을 기초로 역양자화를 수행할 수 있다.
- [0092] 역변환부(225)는 영상 부호화기에서 수행된 양자화 결과에 대해, 부호화기의 변환부가 수행한 DCT 및 DST에 대해 역DCT 및/또는 역DST를 수행할 수 있다. 역변환은 부호화기에서 결정된 전송 단위 또는 영상의 분할 단위를 기초로 수행될 수 있다. 부호화기의 변환부에서 DCT 및/또는 DST는 예측 방법, 현재 블록의 크기 및 예측 방향 등 복수의 정보에 따라 선택적으로 수행될 수 있고, 복호화기의 역변환부(225)는 부호화기의 변환부에서 수행된 변환 정보를 기초로 역변환을 수행할 수 있다.
- [0093] 예측부(230)는 엔트로피 복호화부(210)에서 제공된 예측 블록 생성 관련 정보와 메모리(240)에서 제공된 이전에 복호화된 블록 및/또는 픽처 정보를 기초로 예측 블록을 생성할 수 있다. 복원 블록은 예측부(230)에서 생성된 예측 블록과 역변환부(225)에서 제공된 레지듀얼 블록을 이용해 생성될 수 있다. 현재 PU에 대한 예측 모드가 인트라 예측(intra prediction) 모드(화면 내 예측 모드)인 경우에, 현재 픽처 내의 화소 정보를 기초로 예측 블록을 생성하는 화면 내 예측을 수행할 수 있다.
- [0094] 현재 PU에 대한 예측 모드가 인터 예측(inter prediction) 모드(화면 간 예측 모드)인 경우에, 현재 픽처의 이전 픽처 또는 이후 픽처 중 적어도 하나의 픽처에 포함된 정보를 기초로 현재 PU에 대한 화면 간 예측을 수행할 수 있다. 이때, 영상 부호화기에서 제공된 현재 PU의 화면 간 예측에 필요한 움직임 정보, 에컨대 움직임 벡터, 참조 픽처 인덱스 등에 관한 정보는 부호화기로부터 수신한 스킵 플래그, 머지 플래그 등을 확인하고 이에 대응하여 유도될 수 있다.
- [0095] 복원된 블록 및/또는 픽처는 필터부(235)로 제공될 수 있다. 필터부(235)는 복원된 블록 및/또는 픽처에 디블록킹 필터링, SAO(Sample Adaptive Offset) 및/또는 적응적 루프 필터링 등을 적용한다.
- [0096] 메모리(240)는 복원된 픽처 또는 블록을 저장하여 참조 픽처 또는 참조 블록으로 사용할 수 있도록 할 수 있고 또한 복원된 픽처를 출력부로 제공할 수 있다.
- [0097]
- [0098] 한편, 인터 예측 모드의 경우에 부호화기 및 복호화기는 현재 블록의 움직임 정보를 도출하고, 도출된 움직임 정보에 기반하여 현재 블록에 대한 인터 예측을 수행할 수 있다.
- [0099] 도 3은 인터 예측 모드에서 AMVP(Advanced Motion Vector Predictor)가 적용되는 경우, 움직임 벡터를 도출하는 방법에 관한 일 실시예를 개략적으로 설명하는 순서도이다.
- [0100] 도 3을 참조하면, 부호화기 및 복호화기는 현재 블록에 대한 움직임 벡터 예측자(Motion Vector Predictor: MVP, 이하 'MVP'라 함) 후보 리스트를 생성할

- 수 있다(S310). MVP는 현재 블록의 움직임 벡터에 대한 예측값을 나타낼 수 있다.
- [0101] 부호화기 및 복호화기는 현재 블록에 인접하고 이용 가능한(available) 주변 블록 및/또는 현재 블록과 동일 위치에 있는(co-located) 각 참조 픽처의 블록들 중에서 이용 가능한(available) 블록(이하, 설명의 편의를 위해 ‘CoI 블록’(co-located block)이라 함)의 움직임 벡터를 이용하여 MVP 후보 리스트를 생성할 수 있다.
- [0102] MVP 후보 리스트 생성 방법의 구체적인 방법은 후술한다.
- [0103] 또한, 본 명세서에서는 현재 블록에 인접한 블록을 설명의 편의를 위해 ‘주변 블록’이라 한다.
- [0104] 부호화기 및 복호화기는 MVP 후보 리스트에 포함된 MVP 후보들 중 현재 블록에 대한 MVP를 선택할 수 있다(S320).
- [0105] 부호화기는 MVP 후보 리스트에 포함된 MVP 후보들에 대해 움직임 벡터 경쟁(Motion Vector Competition: MVC)을 적용하여, 현재 블록에 대한 최적의 MVP를 선택할 수 있다. MVP가 선택되면, 부호화기는 비트 스트림을 통해 MVP 인덱스를 복호화기로 전송할 수 있다. MVP 인덱스는, 현재 블록의 MVP를 지시하는 인덱스로서 MVP 후보 리스트에 포함된 MVP 후보들 중에서 선택될 수 있다.
- [0106] 복호화기는 부호화기로부터 MVP 인덱스를 수신할 수 있다. 수신한 MVP 인덱스를 이용하여, 복호화기는 MVP 후보 리스트에 포함된 MVP 후보들 중 현재 블록에 대한 MVP를 선택할 수 있다.
- [0107] 복호화기는 선택된 MVP를 이용하여 현재 블록의 움직임 벡터를 도출할 수 있다(S330).
- [0108] 현재 블록에 대한 MVP가 선택되면, 부호화기는 현재 블록의 움직임 벡터와 MVP의 차분을 구할 수 있다. 움직임 벡터와 MVP의 차분을 움직임 벡터 차분(Motion Vector Difference: MVD, 이하 ‘MVD’라 함)이라 한다.
- [0109] 부호화기는 움직임 벡터 자체가 아닌, MVD에 대한 정보를 복호화기로 전송할 수 있다. 이 때, MVD가 작을수록 부호화기에서 복호화기로 전송되는 정보량이 감소될 수 있다.
- [0110] 복호화기는 부호화기로부터 MVD에 대한 정보를 수신할 수 있다. 현재 블록에 대한 MVP가 선택되면, 복호화기는 선택된 MVP와 MVD를 더하여 현재 블록의 움직임 벡터를 구할 수 있다.
- [0111]
- [0112] 도 4는 MVP 후보 리스트 생성 방법에 관한 일 실시예를 개략적으로 설명하는 개념도이다.
- [0113] 본 명세서에서는 설명의 편의를 위해, 현재 블록(400) 좌하측 코너의 주변 블록인 좌하측 주변 블록  $A_0$ (410) 및 현재 블록(400)의 좌측 주변 블록들 중 최하단에 위치하는 좌측 블록  $A_1$ (420)을 포함하는 후보 그룹을 좌측 후보

그룹이라 한다. 또한 현재 블록(400) 우상측 코너의 주변 블록인 우상측 주변 블록  $B_0$ (430), 현재 블록(400)의 상측 주변 블록들 중 최우측에 위치한 블록 우측 주변 블록  $B_1$ (440) 및 현재 블록(400) 좌상측 코너의 주변 블록인 좌상측 블록  $B_2$ (450)을 포함하는 후보 그룹을 상측 후보 그룹이라 한다.

- [0114] 부호화기 및 복호화기는 좌측 후보 그룹( $A_0, A_1$ )에서 하나의 MVP 후보를 선택할 수 있다. 좌측 후보 그룹에서 선택된 MVP 후보를  $MV_A$ 라 한다. 예컨대, 부호화기 및 복호화기는 좌측 후보 그룹에 포함된 블록들을  $A_0 \rightarrow A_1$ 의 순서로 스캔하면서, 현재 블록과 참조 픽처 인덱스가 동일하고 이용 가능한 첫 번째 블록의 움직임 벡터를 현재 블록의 MVP 후보( $MV_A$ )로 선택할 수 있다.
- [0115] 부호화기 및 복호화기는 또한, 상측 후보 그룹( $B_0, B_1, B_2$ )에서 하나의 MVP 후보를 선택할 수 있다. 상측 후보 그룹에서 선택된 MVP 후보를  $MV_B$ 라 한다. 예를 들어, 부호화기 및 복호화기는 상측 후보 그룹에 포함된 블록들을  $B_0 \rightarrow B_1 \rightarrow B_2$ 의 순서로 스캔하면서, 현재 블록과 참조 픽처 인덱스가 동일하고 이용 가능한 첫 번째 블록의 움직임 벡터를 현재 블록의 MVP 후보( $MV_B$ )로 선택할 수 있다.
- [0116] 또한, 부호화기 및 복호화기는 참조 픽처 내의 Col 블록 COL(460)의 움직임 벡터를 현재 블록의 MVP 후보로 선택할 수도 있다. 여기서, Col 블록(660)의 움직임 벡터는  $mvCol$ 로 표시할 수 있다.
- [0117] 상술한 실시예에서, 좌측 후보 그룹에서 선택된 MVP 후보 및 상측 후보 그룹에서 선택된 MVP 후보는 공간적 MVP(Spatial Motion Vector Predictor: SMVP, 이하 ‘SMVP’라 함)라고 할 수 있다. 또한, 참조 픽처 내의 Col 블록을 이용하여 선택된 MVP 후보는 시간적 MVP(Temporal Motion Vector Predictor: TMVP, 이하 ‘TMVP’라 함)라고 할 수 있다. 따라서, MVP 후보 리스트는 SMVP 및/또는 TMVP를 포함할 수 있다.
- [0118] 부호화기 및 복호화기는 상술한 방법에 의해 선택된 MVP 후보들 중 중복되는 후보가 있다면, 중복되는 후보들 중에서 가장 높은 순위의 후보를 제외한 나머지 후보를 제거할 수 있다.
- [0119]
- [0120] 상술한 바와 같이, 현재 블록에 대한 인터 예측을 위하여 SMVP 뿐만 아니라, TMVP가 이용될 수 있다.
- [0121] 도 5는 복호화기에서 인터 예측을 위해 TMVP를 유도하는 방법의 일 예를 개략적으로 설명하는 순서도이다. 도 5에서는 설명의 편의를 위해 복호화기의 동작을 설명하지만, 부호화기도 복호화기와 동일한 TMVP를 후보로 이용하여 인터 예측을 수행하기 위해 복호화기와 동일한 동작을 수행한다. 따라서, 이 하 도 5에 관하여 설명하는 복호화기의 동작은 부호화기에도 동일하게 적용될 수 있다.
- [0122] 도 5를 참조하면, 복호화기는 우선 col 블록(동일 위치(co-located) 블록) 또는 col 파티션(co-located partition)을 포함하는 동일 위치 픽처(co-located picture: colPic,

이하 ‘colPic’이라 함)를 특정한다(S510).

- [0123] 복호화기는 슬라이스 타입에 관한 정보와 어떤 참조 픽처 리스트로부터 colPic이 선택되는지에 관한 정보에 기반하여 colPic을 특정할 수 있다.
- [0124] 예컨대, 현재 블록에 대한 슬라이스 타입이 B이고 L0으로부터 참조 픽처가 선택되지 않는다고 하는 경우에, colPic은 L1에서 특정된다.
- [0125] 영상 부호화, 복호화에 사용되는 픽처는 I 픽처(I picture), P 픽처(P picture), B 픽처(B picture)로 분류할 수 있다.
- [0126] I 픽처는 전후의 화면과는 관계없이 그 화면 내에서 독립적으로 부호화되는 픽처로서, 시간 방향의 예측이 적용되지 않으며, 화면 내 정보만이 부호화 처리에 사용된다. P 픽처는 하나의 참조 픽처를 이용한 단방향의 인터 예측에 의해 부호화될 수 있는 픽처다. P 픽처에서는 한 개의 참조 픽처 리스트를 필요로 하며, 이를 참조 픽처 리스트 0(reference picture list 0: L0)이라 지칭한다. L0으로부터 선택된 참조 픽처를 사용하는 인터 예측을 L0 예측이라고도 한다. L0 예측은 주로 순방향 예측에 사용된다. P 픽처에서는 인트라 예측 또는 L0 예측이 수행될 수 있다.
- [0127] B 픽처는 하나 이상, 예를 들어 2 개의 참조 픽처를 이용하여 순방향, 역방향 또는 양 방향 인터 예측에 의해 부호화될 수 있는 픽처다. B 픽처는 두 개의 참조 픽처 리스트를 필요로 하며, 두 개의 참조 픽처 리스트는 각각 참조 픽처 리스트 0(reference picture list 0: L0), 참조 픽처 리스트 1(reference picture list 1: L1)이라 지칭한다. 상술한 바와 같이, L0으로부터 선택된 참조 픽처를 사용하는 인터 예측을 L0 예측이라 하며, L0 예측은 주로 순방향 예측에 사용된다. L1으로부터 선택된 참조 픽처를 사용하는 인터 예측을 L1 예측이라 하며, L1 예측은 주로 역방향 예측에 사용된다. 또한 L0과 L1으로부터 각각 선택된 두 개의 참조 픽처를 사용하는 인터 예측을 쌍 예측(bi prediction)이라고도 한다.
- [0128] B 픽처에서는 인트라 예측, L0 예측, L1 예측 또는 쌍 예측(Bi prediction)이 수행될 수 있다. B 픽처에 대해서는 두 개의 움직임 정보가 사용될 수 있다. 따라서, B 픽처에 대해서 과거의 두 참조 픽처를 이용하여 순방향 예측을 수행할 수도 있고, 미래의 두 참조 픽처를 이용하여 역방향 예측을 수행할 수도 있으며, 과거의 참조 픽처와 미래의 참조 픽처를 하나씩 이용하여 쌍 예측을 수행할 수도 있다. 쌍 예측을 수행할 수 있는 B 픽처에 대해서는 L0, L1 두 개의 참조 픽처 리스트가 필요할 수 있다.
- [0129] 상기 I 픽처, P 픽처, B 픽처의 특징은 픽처 단위가 아닌 슬라이스 단위로도 정의될 수 있다. 예컨대, 슬라이스 단위에서 I 픽처의 특징을 갖는 I 슬라이스, P 픽처의 특징을 갖는 P 슬라이스, B 픽처의 특징을 갖는 B 슬라이스가 정의될 수 있다.
- [0130] 한편, 현재 블록에 대한 슬라이스 타입이 B이고 L0으로부터 colPic이 선택되는 경우 또는 현재 블록에 대한 슬라이스 타입이 P인 경우에 colPic은 L0에서 선택될 수 있다.

- [0131] 복호화기는 선택된 colPic에서 현재 블록의 예측에 이용할 동일 위치 예측 블록 colPu를 특정할 수 있다(S520). 복호화기는 현재 블록에 대한 인터 예측에 있어서, colPic 내에서 특정되는 colPu의 정보를 TMVP로서 이용할 수 있다.
- [0132] 도 6은 colPu를 특정하는 방법을 개략적으로 설명하는 도면이다. 도 6을 참조하면, colPic(미도시) 내에서 현재 블록에 대응하는 col 블록(600)의 좌상단(top-left)에 위치하는 화소 샘플  $S_{TL}$ 의 위치를 (xP, yP)라고 특정할 수 있다. 이때, 현재 블록과 col 블록은 PU일 수 있다.
- [0133] 현재 블록의 폭을 nPSW라고 하고, 현재 블록의 높이를 nPSH라고 할 때, col 블록(600)의 우하단(right-bottom)에 위치하는 화소 샘플  $S_{RB}$ 의 위치를 (xPRb, yPRb)로 특정할 수 있다. 따라서, 현재 블록(col 블록)의 크기는  $S_{RB}$ 에 의해 특정된다고 할 수 있으며,  $S_{RB}$ 는 후술하는 바와 같이 위치 결정 화소로서 역할 할 수 있다.
- [0134]  $S_{TL}$ 과  $S_{RB}$ 가 동일한 LCU(Largest CU)에 속하는 경우에는 colPic 내에서  $S_{RB}$ 가 속하는 소정의 움직임 정보 저장 단위의 좌상단을 커버하는 예측 블록을 colPu(동일 위치 예측 유닛)으로 특정할 수 있다. 예컨대, 상술한 바와 같이  $S_{RB}$ 는 colPu의 위치를 결정하는 화소로서의 역할을 수행한다고 할 수 있다.
- [0135] 움직임 정보 저장 단위라 함은, 인터 예측에 있어서, 예측에 이용되기 위한 움직임 정보를 저장하는 단위이다. 소정 크기의 블록을 움직임 정보 저장 단위로 설정하고 움직임 정보 저장 단위별로 하나의 움직임 정보를 지정할 수 있다. 예컨대, colPic에서 NxN 샘플들(N은 정수)마다 하나의 값을 지정하여 저장하고, colPic 내 특정 블록에 대한 움직임 정보를 인터 예측을 위해 이용하는 경우에는 해당 블록을 커버하는 움직임 정보 저장 단위에 대한 움직임 정보를 이용할 수 있다. 움직임 정보 저장 단위의 크기는 4x4 샘플들, 6x6 샘플들, 8x8 샘플들, 12x12 샘플들, 16x16 샘플들, 32x32 샘플들, 64x64 샘플들과 같이 특정한 크기의 샘플 블록으로 설정될 수 있다. 움직임 정보는 설정된 크기의 샘플 블록 단위로 저장될 수 있다.
- [0136] 예컨대, 움직임 정보 저장 단위가 16x16 샘플 블록이라면, ((xPRb>>4)<<4, (yPRb>>4)<<4)의 위치가,  $S_{RB}$ 가 속하는 움직임 정보 저장 단위의 좌상단을 커버하는 colPu의 위치로서 특정될 수 있다.
- [0137]  $S_{TL}$ 과  $S_{RB}$ 가 동일한 LCU에 포함되지 않는 경우에는 col 블록의 중앙(구체적으로는 col 블록 크기의 1/2에 의해 특정되는 중앙 우하(center right-bottom))에 위치하는 샘플  $S_{CTR}$ 을 기준으로 colPu를 특정할 수 있다. 따라서,  $S_{CTR}$  역시 colPu의 위치를 결정하는 위치 결정 화소로서의 역할을 수행한다.  $S_{CTR}$ 의 위치 (xPCtr, yPCtr)는 (xP+(nPSW>>1), yP+(nPSH>>1))으로 특정될 수 있다. 이 경우에는  $S_{CTR}$ 이 속하는 소정의 정보 저장 단위의 좌상단을 커버하는 예측 블록을 colPu로서 특정할 수 있다.
- [0138] 이 경우에도, 정보 저장 단위가 16x16 샘플 블록이라면, ((xPCtr>>4)<<4, (yCtr>>4)<<4)의 위치가 colPu의 위치로서 특정될 수 있다.

- [0139] 한편, colPic 내에서 colPu의 위치 (xPCol, yPCol)은 colPic의 좌상단(top-left) 샘플의 위치에 대한 colPu의 좌상단 샘플의 위치로 특정될 수 있다.
- [0140] 복호화기는 colPu의 정보를 기반으로 현재 블록의 예측에 이용할 시간적 움직임 벡터(Temporal Motion Vector)와 참조 리스트의 가용성에 관한 정보를 유도할 수 있다(S530).
- [0141] 예컨대, colPu가 인트라 예측 모드로 코딩된 경우, colPu가 가용하지 않은(unavailable) 경우, colPic이 TMVP(Temporal Motion Vector Prediction)에 이용되지 않는 것으로 설정된 경우 등에는 시간적 움직임 벡터 mvCol을 0으로 설정하고, 참조 리스트가 가용하지 않은 것으로 설정할 수 있다.
- [0142] 이때, 참조 리스트의 가용성을 지시하는 정보로서 availableFlagLXCol(X는 0 또는 1일 수 있음)을 이용할 수 있다. availableFlagLXCol의 값이 0이면 해당 참조 픽처 리스트가 가용한 것으로 판단하고, availableFlagLXCol의 값이 1이면 해당 참조 픽처 리스트는 가용하지 않은 것으로 판단할 수 있다.
- [0143] 상술한 colPu가 인트라 예측 모드로 코딩된 경우, colPu가 가용하지 않은(unavailable) 경우, colPic이 TMVP(Temporal Motion Vector Prediction)에 이용되지 않는 것으로 설정된 경우 등에 해당하지 않는다면, 즉 참조 픽처 리스트가 가용한 경우에 시간적 움직임 정보로서 시간적 움직임 벡터 mvCol과 참조 리스트 refIdxCol, 참조 리스트 식별자 listCol은 아래와 같이 유도할 수 있다.
- [0144] colPu에 대하여 L0 유용성(utilization)을 판단하여 L0가 유용하지 않다고 판단된 경우에는 L1을 참조 리스트로서 하고 colPu의 정보를 이용하여 시간적 움직임 벡터 mvCol의 값, 참조 인덱스 refIdxCol의 값을 특정할 수 있다. 예컨대, mvCol의 값은  $MvL1[xPCol][yPCol]$ 로 설정되고, 참조 인덱스 refIdxCol의 값은  $refIdxL1[xPCol][yPCol]$ 로 설정될 수 있다.
- [0145] L0의 유용성까지 판단하여 L0이 유용하다고 판단되고 L1이 유용하지 않다고 판단된 경우에는, L0을 참조 리스트로 하고 colPu의 정보를 이용하여 시간적 움직임 벡터 mvCol의 값, 참조 인덱스 refIdxCol의 값을 특정할 수 있다. 예컨대, mvCol의 값은  $MvL0[xPCol][yPCol]$ 로 설정되고, 참조 인덱스 refIdxCol의 값은  $refIdxL0[xPCol][yPCol]$ 로 설정될 수 있다.
- [0146] L0과 L1이 모두 유용하다고 판단된 경우에는, 현재 픽처의 픽처 오더 카운트와 각 참조 픽처 리스트의 픽처들에 대한 픽처 오더 카운트를 비교하여, 시간적 움직임 정보를 결정할 수 있다. 예컨대, 현재 픽처의 픽처 오더 카운트가 참조 픽처 리스트들 내의 픽처들보다 크거나 같은 경우에는 시간적 움직임 정보를 유도하는 절차의 대상이 된 참조 픽처 리스트에서 colPu의 위치 (xPCol, yPCol)에 의해 지시되는 움직임 벡터와 참조 인덱스가 mvCol, refIdxCol의 값으로 각각 설정될 수 있다.
- [0147] 참조 픽처 리스트들 내의 픽처들 중에서 적어도 하나의 픽처 오더 카운트가 현재 픽처의 픽처 오더 카운트보다 큰 경우에는, 부호화기로부터 시그널링에 의해 지시되는 참조 픽처 리스트를 이용하여, 해당 참조 픽처 리스트에서

colPu의 위치 (xPCol, yPCol)에 의해 지시되는 움직임 벡터와 참조 인덱스가 mvCol, refIdxCol의 값으로 각각 설정될 수 있다.

- [0148] 복호화기는 상술한 바와 같이 유도된 시간적 움직임 벡터와 참조 픽처 인덱스를 움직임 정보로서 가지는 TMVP를 MVP 후보로서 이용할 수 있다. 이때, colPic와 colPic에서 이용하는 참조 픽처 사이의 거리가 현재 픽처와 현재 픽처에서 이용하는 참조 픽처 사이의 거리가 동일하다면 상술한 바와 같이 유도한 mvCol을 그대로 시간적 움직임 벡터로 이용할 수 있다. colPic와 colPic에서 이용하는 참조 픽처 사이의 거리가 현재 픽처와 현재 픽처에서 이용하는 참조 픽처 사이의 거리가 동일하지 않다면 상술한 바와 같이 유도한 mvCol을 스케일링하여 시간적 움직임 벡터로 이용할 수 있다.
- [0149] 한편, 상술한 바와 같이, TMVP로서 이용되는 colPu의 움직임 정보는 소정의 움직임 정보 저장 단위별로 저장될 수 있다. 즉, 움직임 정보 저장 단위별로 colPu를 설정하고, 해당 움직임 정보 저장 단위에 속하는 블록의 움직임 정보로서 colPu의 움직임을 이용할 수 있다.
- [0150] 이는 움직임을 저장하기 위한 메모리를 줄이기 위한 것으로서, 움직임 정보 저장 단위가 되는  $N_1 \times N_2$  ( $N_1, N_2$ 는 정수) 블록의 사이즈는 미리 설정되어 있을 수도 있고, 부호화기로부터 시그널링될 수도 있다.
- [0151] 도 7은 움직임을 저장하는 방법에 대하여 개략적으로 설명하는 도면이다. 도 7에서는 움직임 정보가 설정되는 최소 단위인 블록 B0 ~ B15를 하나의 움직임 정보 저장 단위(Motion Data Storage Unit)로 하는 경우를 예로서 설명하고 있다.
- [0152] 본 명세서에서는 움직임 정보가 설정되는 최소 단위를 설명의 편의를 위해 움직임 정보 블록이라고 표현한다. 움직임 정보 블록은 CU, PU 또는 TU일 수도 있고, CU, TU 및 PU와는 별도의 단위일 수도 있다. 움직임 정보 블록은 도 6에서 설명한  $S_{TL}$ ,  $S_{RB}$ ,  $S_{CTR}$ 에 의해 특정되는 블록에 대응할 수 있다. 움직임 정보 블록이 4x4 샘플의 블록이라고 하면, 움직임 정보 저장 단위(700)는 16x16 샘플의 블록이 된다.
- [0153] 도 7에서는 각 움직임 정보 블록 BI ( $I=0, \dots, 15$ )의 움직임을 벡터를 (XI, YI)로 특정하고 있다.
- [0154] 이때, 움직임 정보를 저장 단위별로 압축하여 저장할 것인지 어떤 저장 단위를 이용할 것인지는 미리 정해져 있을 수도 있고, 부호화기로부터 시그널링될 수도 있다. 부호화기로부터 시그널링되는 경우에, 움직임을 저장 단위별로 압축하여 저장할 것인지를 나타내는 플래그(motion\_vector\_buffer\_comp\_flag)와 몇 개의 움직임 정보 블록(또는 몇 개의 화소)을 하나의 저장 단위로 할 것인지에 관한 압축 비율 정보(motion\_vector\_buffer\_comp\_ratio)를 이용할 수 있다.
- [0155] 예컨대, motion\_vector\_buffer\_comp\_flag의 값이 1이면 움직임을 저장 단위별로 압축하여 저장할 수 있고, motion\_vector\_buffer\_comp\_ratio의 값이 4이면 도 7의 예와 같이 4x4의 움직임 정보 블록을 하나의 움직임 정보 저장 단위로 이용할 수 있다.

- [0156] 복호화기는 적응적 루프 필터를 적용한 후에, 도 7에 도시된 움직임 정보 저장 단위 내에서 가장 왼쪽 위에 있는 움직임 정보 블록 B0의 움직임 벡터 (X0, Y0)를 움직임 정보 저장 단위(700)에 대한 대표값으로 이용할 수 있다. 즉, 다른 15 개의 움직임 정보 블록 B1 ~ B15에 대해서도 (X0, Y0)의 움직임 벡터값이 할당된다.
- [0157] 이하 본 명세서에는 설명의 편의를 위해, 움직임 정보 저장 단위에 대한 대표 움직임 벡터로서 저장되는 움직임 벡터를 갖는 움직임 정보 블록을 대표 움직임 정보 블록이라고도 표현한다. 움직임 정보 블록이 PU인 경우에, 대표 움직임 정보 블록은 도 5에서 설명한 colPu에 대응할 수 있다.
- [0158] 대표 움직임 정보 블록인 B0 블록이 인터 예측에 이용할 수 없는 블록인 경우, 예컨대 인트라 예측 모드로 부호화된 블록(이하, 설명의 편의를 위해 ‘인트라 블록’이라 함)인 경우라면, 16 개의 움직임 정보 블록에 제로 움직임 벡터 (0, 0)이 할당된다.
- [0159] 다시 말하면, 움직임 정보 저장 단위(700) 내의 첫 번째 움직임 정보 블록인 B0이 인트라 코딩되지 않은 경우에는, 움직임 정보 블록 B0의 움직임 벡터 (X0, Y0)이 16 개의 움직임 정보 블록(B0 ~ B15)에 대한 대표 움직임 벡터로서, 시간적 움직임 벡터(TMVP)를 도출하기 위한 움직임 벡터 버퍼에 저장된다. 움직임 정보 블록 B0이 인트라 코딩된 경우에는, 움직임 벡터 (0, 0)이 움직임 정보 저장 단위(700)를 대표하는 움직임 벡터 값으로서 TMVP를 도출하기 위한 움직임 벡터 버퍼에 저장된다.
- [0160] 상술한 바와 같이 움직임 정보를 압축하여 저장하는 경우에는, 움직임 정보 저장 단위 내의 움직임 정보 블록들에 가장 왼쪽 위의 블록 B0의 움직임 정보가 할당되게 된다.
- [0161] 이렇게 움직임 정보 블록 B0의 움직임 정보가 움직임 정보 저장 단위 내 다른 움직임 정보 블록에도 일괄적으로 할당됨으로써 부호화 효율이 저하되는 문제가 생길 수 있다. 예컨대, B0 블록이 인트라 블록인 경우에는, 움직임 정보 저장 단위 내의 다른 블록들 중에 인터 예측 모드로 코딩되어 있는 블록(이하, 설명의 편의를 위해 ‘인터 블록’이라 함)이 있더라도, 움직임 정보 저장 단위에 대한 대표 움직임 벡터 값으로서 (0,0)이 저장된다.
- [0162] 또한, 움직임 정보 저장 단위 내 각 움직임 정보 블록의 참조 인덱스를 고려하지 않고, 모두 동일한 움직임 벡터가 할당됨으로써 왜곡이 발생할 수 있다.
- [0163] 도 8은 참조 인덱스를 고려하지 않고 움직임 벡터 메모리 압축(움직임 벡터 저장 감축)을 수행하는 예를 개략적으로 설명하는 도면이다. 여기서 움직임 벡터 저장 감축은 상술한 바와 같이, 움직임 정보 저장 단위 내의 움직임 정보 블록들에 대하여 하나의 움직임 벡터(대표 움직임 벡터) 또는 하나의 움직임 정보(대표 움직임 정보)만을 할당하여 저장/이용하는 것을 의미한다.
- [0164] 도 8을 참조하면, 움직임 정보 저장 단위(800) 내 각 움직임 정보 블록 B0 ~ B15의 움직임 정보, 즉 움직임 벡터 (XI, YI)와 ref\_idxI (I=0,..., 15)에 대하여, 움직임 벡터 메모리 압축에 의해 ref\_idxI의 값에 상관없이 모두 동일한 움직임

벡터 (X0, Y0)이 할당되고 있다.

[0165] 따라서, 상술한 문제들을 해결하면서, TMVP를 유도하는데 이용되는 움직임 정보를 저장하는 방법을 고려할 필요가 있다.

[0166] 이하, 본 발명에 따라서 TMVP를 유도하는데 이용되는 움직임 정보를 저장하는 방법에 대하여 도면을 참조한 실시예를 통해 구체적으로 설명한다.

[0167]

[0168] <실시예 1>

[0169] 본 실시예에서는 NxN 개(N은 정수)의 움직임 정보 블록들을 하나의 움직임 정보 저장 단위로 하는 경우에, 움직임 정보 저장 단위 내의 좌상단(left-top)에 있는 움직임 정보 블록에 대한 움직임 벡터 대신 다른 특정 위치의 움직임 정보 블록에 대한 움직임 벡터를 움직임 정보 저장 단위를 대표하는 움직임 벡터로서 저장하고 이용한다.

[0170] 이때, 대표 움직임 벡터로서 이용될 움직임 벡터는, 움직임 정보 저장 단위 내에서 (1) 우상단(Right Top: RT)에 위치하는 움직임 정보 블록의 움직임 벡터, (2) 좌하단(Bottom Left: BL)에 위치하는 움직임 정보 블록의 움직임 벡터, (3) 우하단(Bottom Right: BR)에 위치하는 움직임 정보 블록의 움직임 벡터, (4) 가운데에 위치하는 움직임 정보 블록들 중에서 좌상단(Center\_LT) 움직임 정보 블록의 움직임 벡터, (5) 가운데 위치하는 움직임 정보 블록들 중에서 우상단(Center\_RT) 움직임 정보 블록의 움직임 벡터, (6) 가운데 위치하는 움직임 정보 블록들 중에서 좌하단(Center\_BL) 움직임 정보 블록의 움직임 벡터, (7) 가운데 위치하는 움직임 정보 블록들 중에서 우하단(Center\_BR) 움직임 정보 블록의 움직임 벡터 중 어느 하나 일 수 있다.

[0171] 복호화기는 움직임 정보 저장 단위 내 좌상단에 위치하는 움직임 정보 블록의 움직임 벡터 대신에 상기 (1) 내지 (7) 중 특정된 한 움직임 정보 블록의 움직임 벡터를 각 움직임 정보 블록에 할당하여 TMVP를 유도하는데 이용할 수 있다.

[0172] 도 9는 본 발명이 적용되는 시스템에서 4x4 개의 움직임 정보 블록을 하나의 움직임 정보 저장 단위로 하는 경우에 각 움직임 정보 블록의 움직임 정보를 개략적으로 표시한 것이다.

[0173] 도 9를 참조하면, 각 움직임 정보 블록 BI (I=0, ..., 15)에 대한 움직임 벡터 (XI, YI)가 도시되어 있다. 이 경우, 움직임 정보 블록 B0 외에 다른 움직임 정보 블록을 대표 움직임 정보 블록으로 지정하고, 지정된 대표 움직임 정보 블록의 움직임 벡터를 움직임 정보 저장 단위(900)에 대한 움직임 벡터로 이용할 수 있다.

[0174] 이때, 어떤 움직임 정보 블록을 대표 움직임 정보 블록으로 지정할 것인지는 미리 결정되어 있을 수도 있고, 부호화기로부터 시그널링될 수도 있다.

[0175] 예컨대, 움직임 정보 저장 단위(900) 내의 우상단(RT)에 위치하는 움직임 정보 블록 B3을 대표 움직임 정보 블록으로 지정하면, 움직임 정보 저장 단위(900)에 대한 움직임 벡터는 (X3, Y3)이 된다. 움직임 정보 저장 단위(900) 내의

좌하단(BL)에 위치하는 움직임 정보 블록 B12를 대표 움직임 정보 블록으로 지정하면, 움직임 정보 저장 단위(900)에 대한 움직임 벡터는 (X12, Y12)이 된다. 움직임 정보 저장 단위(900) 내의 우하단(BR)에 위치하는 움직임 정보 블록 B15를 대표 움직임 정보 블록으로 지정하면, 움직임 정보 저장 단위(900)에 대한 움직임 벡터는 (X15, Y15)이 된다. 움직임 정보 저장 단위(900) 내 중앙의 좌상단(Center\_LT)에 위치하는 움직임 정보 블록 B5을 대표 움직임 정보 블록으로 지정하면, 움직임 정보 저장 단위(900)에 대한 움직임 벡터는 (X5, Y5)이 된다. 움직임 정보 저장 단위(900) 내 중앙의 우상단(Center\_RT)에 위치하는 움직임 정보 블록 B6을 대표 움직임 정보 블록으로 지정하면, 움직임 정보 저장 단위(900)에 대한 움직임 벡터는 (X6, Y6)이 된다. 움직임 정보 저장 단위(900) 내 중앙의 좌하단(BL)에 위치하는 움직임 정보 블록 B9을 대표 움직임 정보 블록으로 지정하면, 움직임 정보 저장 단위(900)에 대한 움직임 벡터는 (X9, Y9)이 된다. 움직임 정보 저장 단위(900) 내 중앙의 우하단(Center\_BR)에 위치하는 움직임 정보 블록 B10을 대표 움직임 정보 블록으로 지정하면, 움직임 정보 저장 단위(900)에 대한 움직임 벡터는 (X10, Y10)이 된다.

[0176] 한편, 압축하는 단위, 움직임 정보 저장 단위의 크기에 따라서 움직임 정보 압축 저장에 있어서 대표값으로 사용되는 움직임 정보를 가지는 움직임 정보 블록의 위치가 적응적으로 결정될 수도 있다. 예컨대, 압축하는 단위를 부호화기에서 결정하여 전송하는 경우에, `motion_vector_buffer_comp_ratio`의 값에 따라서 압축에 사용되는 움직임 벡터(움직임 정보)를 가지는 블록(움직임 정보 블록)의 위치가 결정될 수 있다.

[0177] 구체적으로, 압축하는 단위를 부호화기에서 결정하여 복호화기로 전송하는 경우에, `motion_vector_buffer_comp_ratio`의 값이 4라면, 우하단(bottom-right) 위치에 있는 움직임 정보 블록의 움직임 벡터를 해당 움직임 정보 저장 단위의 움직임 벡터로 사용할 수 있다(우하단 위치의 움직임 정보 블록을 대표 움직임 정보 블록으로 이용). 또한, `motion_vector_buffer_comp_ratio`의 값이 8이나 16인 경우엔, 중앙 우하단(Center\_BR) 위치에 있는 움직임 정보 블록의 움직임 벡터를 해당 움직임 정보 저장 단위의 움직임 벡터로 사용할 수 있다(중앙 우하단 위치의 움직임 정보 블록을 대표 움직임 블록으로 이용).

[0178] 여기서, 각각의 압축 단위, 즉 움직임 정보 저장 단위의 다양한 크기에 대응하여 결정될 수 있는 대표 움직임 정보 블록의 위치는 상술한 바에 한정되지 않는다.

[0179]

#### [0180] <실시예 2>

[0181] 본 실시예에서는 대표 움직임 벡터값이 지정되는 대표 움직임 정보 블록이 인트라 블록이어서 제로 움직임 벡터(zero motion vector)가 대표 움직임 벡터로 저장되는 것을 방지하는 방법을 제공한다.

[0182] 본 실시예에서는 움직임 정보 저장 단위 내의 움직임 정보 블록들을 소정의 순서대로 검색하여 인트라 블록인지 인터 블록인지를 판별하고, 인터 블록인

것으로 검색되는 첫 번째 움직임 정보 블록의 움직임 벡터를 움직임 정보 저장 단위에 대한 대표 움직임 벡터로서 이용한다.

- [0183] 도 10은 본 발명에 따라서 움직임 정보 저장 단위 내에서 움직임 정보 블록이 어떤 예측 모드로 부호화되었는지를 검색하는 순서를 개략적으로 설명하는 도면이다.
- [0184] 도 10에서는 4x4 개의 움직임 정보 블록(B0 ~ B15)으로 움직임 정보 저장 단위(1000)가 구성되는 예를 설명하고 있다. 도 10의 각 움직임 정보 블록 BI(I=0, ...15)에서 I는 검색되는 순서를 나타낸다.
- [0185] 도 10(a)의 예에서는 움직임 정보 저장 단위(1000) 내의 좌상단(Top-left)에 위치하는 움직임 정보 블록을 첫 번째 검색할 블록 B0으로 지정하고, 래스터 스캔 오더(raster scan order)에 따라서 인터 블록인지 인트라 블록인지(혹은, 이용 가능(available)한지 즉, L0을 위한 움직임 벡터를 저장할 경우, L0에 저장된 움직임 벡터가 있는지 없는지 등)를 검색한다.
- [0186] 도 10(b)의 예에서는 움직임 정보 저장 단위(1000) 내의 중앙 우하단(Center Bottom-Right)에 위치하는 움직임 정보 블록을 첫 번째 검색할 블록 B0으로 설정하고, 나선형의 시계 방향으로 검색한다.
- [0187] 도 10(c)의 예에서는 움직임 정보 저장 단위(1000) 내의 중앙 좌상단(Center Top-Left)에 위치하는 움직임 정보 블록을 첫 번째 검색할 블록 B0으로 설정하고, 나선형의 시계 방향으로 검색한다.
- [0188] 도 10(d) 내지 도 10(m)에서는 검색에 의한 계산량을 고려하여, 계산량을 줄이기 위한 실시예를 설명하고 있다.
- [0189] 도 10(d)의 예에서는 움직임 정보 저장 단위(1000) 내의 중앙에 위치하는 움직임 정보 블록들만을 검색하되, 중앙 우하단(Center Bottom-Right)에 위치하는 움직임 정보 블록을 첫 번째 검색할 블록 B0으로 설정하고, 시계 방향으로 검색한다.
- [0190] 도 10(e)의 예에서는 움직임 정보 저장 단위(1000) 내의 중앙에 위치하는 움직임 정보 블록들만을 검색하되, 중앙 좌상단(Center Top-Left)에 위치하는 움직임 정보 블록을 첫 번째 검색할 블록 B0으로 설정하고, 시계 방향으로 검색한다.
- [0191] 도 10(f)의 예에서는 움직임 정보 저장 단위(1000) 내의 중앙에 위치하는 움직임 정보 블록들만을 검색하되, 중앙 우하단(Center Bottom Right)의 움직임 정보 블록(B0) → 중앙 좌하단(Center Bottom-Left)의 움직임 정보 블록(B1) → 중앙 우상단(Center Top-Right)의 움직임 정보 블록(B2) → 중앙 좌상단(Center Top-Left)의 움직임 정보 블록(B3)의 순서로 검색한다.
- [0192] 도 10(g)의 예에서는 움직임 정보 저장 단위(1000) 내의 중앙에 위치하는 움직임 정보 블록들만을 검색하되, 중앙 우하단의 움직임 정보 블록(B0) → 중앙 우상단의 움직임 정보 블록(B1) → 중앙 좌하단의 움직임 정보 블록(B2) → 중앙 좌상단의 움직임 정보 블록(B3)의 순서로 검색한다.

- [0193] 도 10(h)의 예에서는 움직임 정보 저장 단위(1000) 내의 중앙에 위치하는 움직임 정보 블록들만을 검색하되, 중앙 우하단의 움직임 정보 블록(B0) → 중앙 좌상단의 움직임 정보 블록(B1) → 중앙 우상단의 움직임 정보 블록(B2) → 중앙 좌하단의 움직임 정보 블록(B3)의 순서로 검색한다.
- [0194] 도 10(i)의 예에서는 움직임 정보 저장 단위(1000) 내의 중앙에 위치하는 움직임 정보 블록들만을 검색하되, 중앙 우하단의 움직임 정보 블록(B0) → 중앙 좌상단의 움직임 정보 블록(B1) → 중앙 좌하단의 움직임 정보 블록(B2) → 중앙 우상단의 움직임 정보 블록(B3)의 순서로 검색한다.
- [0195] 도 10(j)의 예에서는 움직임 정보 저장 단위(1000) 내의 코너에 위치하는 움직임 정보 블록들만을 검색하되, 좌상단(Top-Left) 코너에 위치하는 움직임 정보 블록을 첫 번째 검색할 블록 B0으로 설정하고, 시계 방향으로 검색한다.
- [0196] 도 10(k) 내지 도 10(m)의 예에서는 도 10(a) 내지 도 10(c)에서 수행되는 검색 단위의 2배가 되도록 검색 단위를 설정하여 검색을 수행한다. 예컨대, 도 10(a) 내지 도 10(c)에서 각 움직임 정보 블록이 4x4 샘플의 블록이라면, 도 10(k) 내지 도 10(m)에서는 8x8 샘플 블록 단위로 검색을 수행한다. 따라서, 8x8 샘플 블록 내의 좌상단에 위치하는 첫 번째 움직임 정보 블록이 검색의 대상이 된다.
- [0197] 움직임 정보 저장 단위(1000)가 4x4 개의 움직임 정보 블록으로 구성되는 경우에, 도 10(k)의 예에서는 도시된 바와 같이 좌상단 2x2 개의 움직임 정보 블록들로부터 시계 방향으로 2x2 개의 움직임 정보 블록들 각각에 대하여, 좌상단의 움직임 정보 블록을 검색해 나간다. 예컨대, 검색 순서는 도시된 바와 같이 B0 → B1 → B2 → B3가 된다.
- [0198] 도 10(l)의 예에서는 도시된 바와 같이 좌하단 2x2 개의 움직임 정보 블록들로부터 반시계 방향으로 2x2 개의 움직임 정보 블록들 각각에 대하여, 좌상단의 움직임 정보 블록을 검색해 나간다. 예컨대, 검색 순서는 도시된 바와 같이 B0 → B1 → B2 → B3가 된다.
- [0199] 도 10(m)의 예에서는 도시된 바와 같이 좌상단 2x2 개의 움직임 정보 블록들로부터 지그재그 방향으로 2x2 개의 움직임 정보 블록들 각각에 대하여, 좌상단의 움직임 정보 블록을 검색해 나간다. 예컨대, 검색 순서는 도시된 바와 같이 B0 → B1 → B2 → B3가 된다.
- [0200] 도 10의 실시예에서, 복호화기는 인터 예측 모드로 코딩된 것으로 가장 먼저 검색된 움직임 정보 블록의 움직임 벡터를 해당 움직임 정보 저장 단위(1000)에 대한 대표 움직임 벡터로서 이용한다.
- [0201]
- [0202] <실시예 3>
- [0203] 상술한 실시예 1 및 2의 경우에는, 결과적으로 인트라 블록의 정보를 기반으로 움직임 정보 저장 단위에 대한 대표 움직임 정보를 저장해야 하는 경우가 발생한다. 예컨대, 실시예 1에서 참조하고자 하는 위치의 움직임 정보 블록이 인트라 블록인 경우(혹은 대상 참조 리스트의 움직임 벡터가 이용 불가능인

경우)와 실시예 2에서 움직임 정보 저장 단위 내의 모든 움직임 정보 블록이 인트라 블록인 경우(혹은 대상 참조 리스트의 움직임 벡터가 이용 불가능인 경우)에는 제로 움직임 벡터 (0, 0)를 움직임 정보 저장 단위에 대한 대표 움직임 벡터로서 이용하게 된다.

- [0204] 본 실시예에서는 이런 경우에 현재 블록에 대한 움직임 정보 저장 단위(이하, '현재 움직임 정보 저장 단위'라 함)의 주변 블록들을 참조하여 움직임 정보를 저장하는 방법을 제공한다.
- [0205] 도 11은 본 발명에 따라서 현재 움직임 정보 저장 단위의 주변 블록들을 이용하여 현재 움직임 정보 저장 단위에 대한 움직임 정보를 저장하는 방법을 개략적으로 설명하는 도면이다. 도 11에서는, 움직임 정보 저장 단위가 4x4 개의 움직임 정보 블록으로 구성되는 예를 설명하고 있다.
- [0206] 도 11을 참조하면, 현재 움직임 정보 저장 단위(1100) 내의 모든 움직임 정보 블록(B0 ~ B15)이 인트라 블록인 경우에, 현재 움직임 정보 저장 단위(1100) 주변의 블록(B16 ~ B35)의 움직임 정보를 이용하여 현재 움직임 정보 저장 단위(1100)에 대한 움직임 정보를 저장/이용한다. 이때, 주변 블록 B16 ~ B35는 움직임 정보 블록일 수 있다.
- [0207] 주변 블록 B16 ~ B35 중 하나의 블록만 인트라 블록이라면, 해당 블록의 움직임 정보를 인트라 블록들로 구성된 현재 움직임 정보 저장 단위(1100)의 움직임 정보로 저장 및 사용한다.
- [0208] 이때, 주변 블록 B16 ~ B35 중 하나의 블록만 대상 참조 리스트의 움직임 벡터를 가지고 있는 인트라 블록인 경우에, 해당 블록의 움직임 정보를 인트라 블록들로 구성된 현재 움직임 정보 저장 단위(1100)의 움직임 정보로 저장 및 사용하도록 할 수도 있다
- [0209] 주변 블록 B16 ~ B35 중 두 개 이상의 블록이 인트라 블록이라면, 아래 (1) 내지 (3) 중 어느 한 방법으로 현재 움직임 정보 저장 단위에 대한 대표 움직임 벡터를 결정할 수 있다.
- [0210] (1) 소정의 순서에 따라서 주변 블록을 검색하며, 참조 인덱스(ref\_idx)의 값이 0인 블록으로서 처음 검색되는 블록의 움직임 벡터를 현재 움직임 정보 저장 단위에 대한 움직임 벡터로 이용한다. 도 11의 예에 따르면, 현재 움직임 정보 저장 단위의 좌상측에 위치하는 주변 블록 B16에서부터 시계 방향으로 인트라 예측 모드로 코딩되었고 참조 인덱스의 값이 0인 블록을 검색해서 첫 번째로 검색되는 블록의 움직임 정보를 현재 움직임 정보 저장 단위에 대한 움직임 정보로서 이용한다.
- [0211] 도 11에서는 현재 움직임 벡터 저장 단위의 좌상단 주변 블록에서부터 시계 방향으로 검색하는 것을 예로 설명하였으나, 이에 한정하지 않고, 우상단 주변 블록, 좌하단 주변 블록, 우하단 주변 블록과 같이 다양한 주변 블록에서 출발하여 시계 방향 또는 반시계 방향으로 검색을 수행하도록 할 수도 있다.
- [0212] (2) 현재 움직임 정보 저장 단위의 주변 블록들이 가지는 움직임 벡터들의

미디언(median)을 현재 움직임 정보 저장 단위에 대한 움직임 벡터로 이용한다. 미디언 값을 도출할 때, 주변 블록들 중에서 인트라 블록은 제외하거나 인트라 블록의 움직임 벡터 값을 (0, 0)으로 설정하여 미디언값을 계산하는데 포함시킬 수도 있다.

- [0213] (3) 현재 움직임 정보 저장 단위 주변의 인트라 블록들 중 인접하는 현재 움직임 정보 저장 단위 내의 인트라 블록(혹은 대상 참조 리스트의 움직임 벡터가 이용 불가능 블록)과 화소값 차이(pixel difference)가 가장 적은 블록의 움직임 정보를 현재 움직임 정보 저장 단위에 대한 움직임 정보로 이용한다.
- [0214] 예컨대, 블록 B16의 우측 하단 화소와 인트라 블록 B0의 좌상단 화소 사이의 차이를 구한다. 블록 B17의 하단 8 화소와 인접한 인트라 블록(혹은 대상 참조 리스트의 움직임 벡터가 이용 불가능 블록) B0의 8 화소 사이의 차이를 구한 후 이를 8로 나눠 화소별 평균 차이를 구한다. 이렇게 주변 블록과 현재 움직임 정보 저장 단위 내 인트라 블록(혹은 대상 참조 리스트의 움직임 벡터가 이용 불가능 블록) 사이의 화소값 차이를 산출하여 차이가 가장 작은 주변 블록의 움직임 정보를 현재 움직임 정보 저장 단위에 대한 움직임 정보로서 이용한다. 만약 화소값 차이가 동일한 주변 블록들이 존재한다면, 미리 우선 순위를 부여할 수 있다. 예컨대, 도 11과 같이 주변 블록을 마킹하는 경우, 더 작은 순번을 가지는 주변 블록의 움직임 정보를 현재 움직임 정보 저장 단위에 대한 움직임 정보로 이용할 수 있다.
- [0215] 한편, 현재 움직임 정보 저장 단위 주변의 블록들이 모두 인트라 블록이라면(혹은 모두 이용 불가능이라면), 현재 움직임 정보 저장 단위에 대한 움직임 벡터를 제로 벡터로 설정할 수도 있다.
- [0216]
- [0217] <실시예 4>
- [0218] 본 실시예에서는 움직임 정보 압축 저장(움직임 정보 저장 감축)에 있어서, 움직임 정보 저장 단위로 움직임 벡터뿐만 아니라, 참조 인덱스도 대표 값을 지정하여 저장하는 방법을 제공한다.
- [0219] 예컨대, 실시예 1 내지 3에서는 움직임 정보 저장 단위별로 하나의 움직임 벡터값만을 저장하는 메모리 압축(memory compression) 혹은 저장 감축(storage reduction)에 관하여 설명하였으나, 실시예 4에서는 움직임 정보 저장 단위에 대한 대표 참조 인덱스를 지정하여 움직임 벡터값과 함께 저장하는 방법을 설명한다.
- [0220] 실시예 1 내지 3에서 움직임 벡터를 저장하기 위한 메모리 버퍼를 줄일 수 있었듯이, 실시예 4에 의하면, 참조 인덱스를 저장하기 위한 메모리 버퍼도 줄일 수 있다.
- [0221] 구체적으로 본 실시예에서는 움직임 정보 저장 단위 내의 대표 움직임 정보 블록이 결정되면, 대표 움직임 정보 블록의 움직임 벡터뿐만 아니라 참조 인덱스까지 움직임 정보 저장 단위 내의 다른 움직임 정보 블록에 할당된다.

복호화기는 움직임 정보 저장 단위 내 블록에 대한 움직임 정보로서 대표 움직임 정보(대표 움직임 정보 블록의 움직임 정보)를 이용할 수 있다. 따라서, 움직임 정보 저장 단위 내의 움직임 정보 블록들에 대하여, 움직임 벡터는 대표 움직임 정보 블록의 움직임 벡터 값으로 할당되지만, 참조 인덱스는 그대로 유지됨으로써 왜곡이 발생하던 문제를 해소할 수 있다.

- [0222] 도 12는 움직임 벡터 메모리 압축(움직임 벡터 저장 감축)이 수행될 때, 움직임 벡터와 함께 참조 인덱스에 대해서도 움직임 정보 메모리 압축(움직임 정보 저장 감축)이 수행되는 것을 개략적으로 설명하는 도면이다. 여기서, 움직임 정보 저장 감축은, 움직임 벡터 저장 감축과 유사하게, 상술한 바와 같이 움직임 정보 저장 단위 내의 움직임 정보 블록들에 대하여 하나의 움직임 정보(대표 움직임 정보)만을 할당하여 저장/이용하는 것을 의미한다.
- [0223] 도 12에서는 움직임 정보 저장 단위가  $4 \times 4$  개의 움직임 정보 블록으로 구성되는 경우를 예로서 설명한다.
- [0224] 또한, 도 12에서는 움직임 정보 저장 단위(1200) 내 좌상단에 위치하는 움직임 정보 블록 B0의 움직임 정보(움직임 벡터, 참조 인덱스)를 대표 움직임 정보로 이용하는 경우를 예로서 설명하고 있다.
- [0225] 도 12를 참조하면, 움직임 정보 저장 감축에 의해 움직임 정보 블록 B0의 움직임 벡터와 참조 인덱스가 움직임 정보 저장 단위(1210)에 대한 움직임 정보로서 할당된다. 예컨대, 도 12의 예에서 대표 움직임 정보 블록인 B0의 움직임 벡터와 참조 인덱스가 각각 (X0, Y0) 및 ref\_idx2 이므로, 움직임 정보 저장 단위(1210)가 도 5 및 도 6에 관한 설명에서  $S_{CRT}$  또는  $S_{BR}$ 에 의해 특정되는 블록을 포함하는 경우에는, B0의 움직임 벡터 (X0, Y0)와 참조 인덱스 ref\_idx2가 TMVP를 유도하기 위해 이용될 수 있다. 도 5에 관한 설명에서 colPu의 움직임 벡터와 함께 colPu의 참조 인덱스를 이용하는 것은 실시예 4에서 대표 움직임 정보 블록의 움직임 벡터와 함께 참조 인덱스를 움직임 정보 저장 단위에 대한 움직임 정보로서 저장하고 이를 이용하는 것에 대응한다고 할 수 있다.
- [0226] 본 실시예 4의 방법은 실시예 1 내지 3의 방법과 함께 적용할 수도 있다. 예컨대, 실시예 1 또는 2에 의해 대표 움직임 정보 블록이 특정되는 경우에는 해당 움직임 정보 블록의 움직임 벡터와 참조 인덱스를 함께 움직임 정보 저장 단위에 할당할 수 있다. 또한, 실시예 3에 의해 현재 움직임 정보 저장 단위 주변의 블록들 중에서 움직임 벡터를 이용할 블록이 결정되면, 결정된 주변 블록의 참조 인덱스를 함께 현재 움직임 정보 저장 단위에 할당할 수 있다.
- [0227] 한편, 실시예 4에 의해 움직임 정보를 압축하고자 하는 대상 블록(움직임 정보 저장 단위 내의 움직임 정보 블록)들 중에 인트라 블록이 포함된 경우에는, 인트라 블록의 참조 인덱스는 변경하지 않고 나머지 움직임 정보 블록에만 대표 움직임 정보 블록의 참조 인덱스를 할당할 수도 있다.
- [0228] 즉, 움직임 정보 저장 단위 내의 모든 움직임 정보 블록에 동일한 참조 인덱스를 설정하는 것이 아니라, 움직임 정보 저장 단위 내 인트라 블록의 참조 인덱스는

변경하지 않고, 나머지 인터 블록의 참조 인덱스만을 대표 움직임 정보 블록의 참조 인덱스로 변경하여 저장할 수도 있다.

- [0229] 인터 블록의 참조 인덱스만 대표 움직임 정보 블록의 참조 인덱스로 변경하는 경우, col 블록의 예측 방향(L0 방향, L1 방향, 쌍방향(Bi-Prediction) 또는 인트라 예측)을 알아내기 위해 해당 움직임 정보 블록의 L1, L0에 대한 참조 인덱스 값을 직접 읽어내어 판단할 수도 있다
- [0230] 도 13은 움직임 벡터 메모리 압축이 수행될 때, 본 발명에 따라서 인터 블록에 대하여만 움직임 벡터와 참조 인덱스의 움직임 정보 메모리 압축이 수행되는 것을 개략적으로 설명하는 도면이다. 도 13은 움직임 정보 저장 단위가 4x4 개의 움직임 정보 블록으로 구성되는 경우를 예로서 설명하고 있다.
- [0231] 도 13의 예에서는, 움직임 정보 저장 단위(1300) 내의 움직임 정보 블록들 B0 ~ B15 중 B10, B11, B14, B15가 인트라 블록(참조 인덱스의 값이 -1)인 것으로 가정한다.
- [0232] 따라서, 인트라 블록을 제외한 나머지 인터 블록의 참조 인덱스만을 대표 움직임 정보 블록의 참조 인덱스로 변경하여, 움직임 정보 저장 단위(1310)의 움직임 벡터와 참조 인덱스를 설정할 수 있다. 움직임 정보 저장 단위(1310) 내 인트라 블록 B10, B11, B14, B15의 움직임 벡터는 대표 움직임 정보 블록의 움직임 벡터로 변경되더라도, 참조 인덱스는 변경되지 않고 유지한다.
- [0233] 도 13에서 설명한 방법에 의해, col 블록의 픽처 타입(P 픽처, B 픽처, I 픽처)과 무관하게, 저장된 참조 인덱스의 값을 이용하여 col 블록의 움직임 벡터를 유도할 수 있으며, 인트라 블록에 대해서는 해당 블록으로부터 col 블록의 움직임 벡터가 유도되지 못하도록 할 수 있다.
- [0234] 도 13에서 설명한 방법은 실시예 1 내지 3과 함께 적용할 수도 있다. 예컨대, 실시예 1 또는 2에 의해 대표 움직임 정보 블록이 특정되어 해당 움직임 정보 블록의 움직임 벡터와 참조 인덱스를 함께 움직임 정보 저장 단위에 할당할 때, 인트라 블록의 참조 인덱스는 변경하지 않을 수 있다. 또한, 실시예 3에 의해 현재 움직임 정보 저장 단위 주변의 블록들 중에서 움직임 벡터를 이용할 블록이 결정되어 결정된 주변 블록의 참조 인덱스를 함께 현재 움직임 정보 저장 단위에 할당할 때, 인트라 블록의 참조 인덱스는 변경하지 않을 수 있다.
- [0235] 한편, 도 13의 경우에는 동일한 움직임 정보 저장 단위 내의 움직임 정보 블록들의 경우에도 서로 다른 참조 인덱스가 할당되는 경우가 생길 수 있다. 이 경우, 움직임 벡터만 압축되고, 참조 인덱스는 압축되지 않게 된다.
- [0236] 이를 해결하기 위해, 도 14에서는 참조 인덱스를 도 12의 경우와 같이, 하나의 대표값으로만 저장하되, 인트라 블록에 대해서는 해당 블록이 인트라 블록임을 추정(infer)할 수 있는 방법을 설명한다.
- [0237] 도 14는 움직임 벡터 메모리 압축이 수행될 때, 본 발명에 따라서 움직임 벡터와 참조 인덱스의 움직임 정보 메모리 압축이 수행되되, 인트라 블록을 추정하는 방법을 개략적으로 설명하는 도면이다. 도 14는 움직임 정보 저장 단위가 4x4

개의 움직임 정보 블록으로 구성되는 경우를 예로서 설명하고 있다.

[0238] 도 14의 움직임 정보 저장 단위(1300) 내의 움직임 정보 블록들 B0 ~ B15 중 B10, B11, B14, B15가 인트라 블록(참조 인덱스의 값이 -1)인 것으로 가정한다.

[0239] 도 14를 참조하면, 인트라 블록인 블록 B10, B11, B14, B15에 대하여도 대표 움직임 정보 블록 B0의 참조 인덱스 값이 할당된다. 다만, 블록 B10, B11, B14, B15에 대하여 저장된 참조 인덱스 값이 0보다 크다고 하여도, 해당 블록이 인트라 블록인 것을 복호화기는 추정할 수 있다. 즉, 블록 B10, B11, B14, B15의 움직임 정보가 이용할 수 없다(not available)는 것을 복호화기가 인식할 수 있다.

[0240] 움직임 정보 블록이 인트라 블록인지 인터 블록인지는 해당 움직임 정보 블록이 속하는 CU의 예측 타입(인트라 예측 모드인지 인터 예측 모드인지)을 이용하여 판단할 수 있다.

[0241] 도 14에서 설명한 방법은 실시예 1 내지 3과 함께 적용할 수도 있다. 예컨대, 실시예 1 또는 2에 의해 대표 움직임 정보 블록이 특정되면, 해당 움직임 정보 블록의 움직임 벡터와 참조 인덱스를 함께 움직임 정보 저장 단위에 할당하고, 인트라 블록인지는 해당 블록이 속하는 CU의 예측 타입을 통해 판단할 수 있다. 또한, 실시예 3에 의해 현재 움직임 정보 저장 단위 주변의 블록들 중에서 움직임 벡터를 이용할 블록이 결정되면, 결정된 주변 블록의 참조 인덱스를 함께 현재 움직임 정보 저장 단위에 할당하고, 인트라 블록인지는 해당 블록이 속하는 CU의 예측 타입을 통해 판단할 수 있다.

[0242] 도 14에서 설명한 방법에 의해, 참조 인덱스를 저장하기 위한 메모리를 절약하고 메모리 대역폭을 줄일 수 있다. 또한, 인트라 블록에 대해서는 해당 블록으로부터 움직임 벡터가 유도되지 않도록 할 수 있다.

[0243]

[0244] <실시예 5>

[0245] 본 실시예에서는 움직임 정보 저장 감축의 방법으로서, 상술한 실시예 2와 실시예 4를 결합하고, 이와 함께 참조 인덱스가 0인지를 고려하는 방법을 제공한다.

[0246] 즉, 실시예 2에서 설명한 방법과 같이 소정의 순서에 따라서 움직임 정보 저장 단위 내의 움직임 정보 블록들에 대하여 인터 블록인지 인트라 블록인지 그리고 인터 블록이라면 참조 인덱스의 값이 0인지를 검색한다.

[0247] 실시예 5에서, 복호화기는 소정의 순서에 따라서 움직임 정보 저장 단위 내의 움직임 정보 블록을 검색하면서, 최초로 검색되는 인터 예측 모드로 부호화되었고 참조 인덱스가 0인 움직임 정보 블록을 대표 움직임 정보 블록으로 선택할 수 있다. 복호화기는 선택된 대표 움직임 정보 블록의 움직임 벡터와 참조 인덱스를 움직임 정보 저장 단위 내의 다른 움직임 정보 블록들에 할당하여 이용할 수 있다.

[0248]

[0249] <실시예 6>

- [0250] 본 실시예에서는 움직임 정보 저장 감축(움직임 벡터 메모리 압축)을 통해, 움직임 벡터, 참조 인덱스와 함께 예측 모드도 압축하여 저장하는 방법을 제공한다.
- [0251] 복호화된 영상은 PU별로 예측 모드에 대한 정보를 저장할 수 있다. 이때, 예측 모드에 대한 정보가 저장되는 PU는 최소 예측 단위 또는 움직임 정보 블록일 수 있으며, 최소 예측 단위 또는 움직임 정보 블록은 4x4 크기의 샘플 블록일 수 있다. 예컨대, PU 단위로 예측 모드가 인트라 모드인지, 인터 모드인지, 인터 모드라면 스킵 모드인지 머지 모드인지 AMVP인지 등에 대한 정보를 저장할 수 있다.
- [0252] 이런 예측 모드에 대한 정보는 col 블록의 상태를 확인하기 위해 복호화된 픽처의 버퍼(Decoded Picture Buffer: DPB)에 저장될 수 있다. 이 경우, 움직임 벡터 버퍼 압축(움직임 벡터 메모리 압축)을 사용하면 PU 단위로 예측 모드에 관한 정보를 저장하지 않고 움직임 정보 저장 단위로 예측 모드에 관한 정보를 저장할 수 있다.
- [0253] 따라서, 본 실시예에서는 움직임 벡터 메모리 압축(움직임 정보 저장 감축)이 적용될 때, 움직임 정보 저장 단위 내 대표 움직임 정보 블록의 예측 모드도 함께 압축(저장)한다.
- [0254] 도 15는 움직임 벡터 메모리 압축이 수행될 때, 본 발명에 따라서 예측 모드에 관한 정보가 압축되는 것을 개략적으로 설명하는 도면이다. 도 15는 움직임 정보 저장 단위가 4x4 개의 움직임 정보 블록으로 구성되는 경우를 예로서 설명하고 있다.
- [0255] 도 15를 참조하면, 움직임 정보 저장 단위(1500) 내 16개의 움직임 정보 블록들은 다양한 예측 모드로 부호화되어 있다. 도 15에서도 움직임 정보 저장 단위(1500)에 대한 대표 움직임 정보 블록은 좌상단에 위치하는 움직임 정보 블록 B0인 경우를 설명한다.
- [0256] 움직임 벡터 메모리 압축에 의해 움직임 정보 저장 단위(1510) 내 4x4 개의 움직임 정보 블록들에 대표 움직임 정보 블록 B0와 동일한 움직임 벡터, 참조 인덱스, 예측 모드가 할당된다. 따라서, 움직임 정보 저장 단위(1510) 내 움직임 정보 블록 B0 ~ B15에 대한 움직임 벡터는 (X0, Y0), 참조 인덱스는 ref\_idx 2, 예측 모드는 스킵(skip) 모드인 것으로 저장된다.
- [0257] 예컨대, 움직임 정보 저장 단위(1510)가 도 5 및 도 6에서 S<sub>RB</sub> 또는 S<sub>CRT</sub>에 의해 특정되는 블록을 포함하는 경우, B0의 움직임 벡터인 (X0, Y0), 참조 인덱스인 ref\_idx 2, 예측 모드인 스킵(skip) 모드가 현재 블록에 대한 TMVP를 유도하기 위한 움직임 정보로서 이용될 수 있다.
- [0258]
- [0259] <실시예 7>
- [0260] 본 실시예에서는 움직임 정보 저장 감축 시, 예컨대 움직임 벡터 버퍼 메모리를 압축할 때 더 효과적으로 압축을 수행할 수 있는 방법을 개시한다.

- [0261] 움직임 정보 저장 감축(혹은 움직임 벡터 버퍼 압축(MV buffer compression)) 기술을 적용할 때, 영상의 특성에 따라서 로컬 움직임 정보들이 유사할수록 성능 저하가 줄어들 수 있다. 로컬 움직임의 유사성은 영상마다 다를 수 있는데, 예를 들면, 16:9인 영상을 16x16 화소 블록의 움직임 정보 저장 단위로 압축하는 경우와 4:3인 영상은 16x16 화소 블록의 영상 정보 저장 단위로 압축하는 경우는 압축 성능에 차이가 있을 수 있다.
- [0262] 따라서, 본 실시예에서는 움직임 정보를 저장하는 버퍼 메모리를 더 동적으로 압축할 수 있는 방법을 개시한다.
- [0263] 부호화기는 시퀀스 레벨(sequence level)에서 움직임 정보 혹은 움직임 벡터를 압축하겠다는 정보가 전달되면 압축 비율에 대한 정보를 복호화기에 시그널링을 할 수 있다. 예컨대, NxM(N, M은 정수) 샘플 블록의 움직임 정보를 하나의 움직임 정보로 압축하여 저장하는 경우에, 복호화기는 가로 방향의 압축 비율에 관한 N의 값과 세로 방향의 압축 비율에 관한 M의 값을 시그널링 할 수도 있다.
- [0264] 부호화기는 움직임 정보 저장 감축에 있어서 가로 방향 압축 비율에 대한 정보로서, motion\_vector\_buffer\_comp\_x\_ratio\_log2와 같은 정보를 전송하고, 세로 방향 압축 비율에 대한 정보로서 motion\_vector\_buffer\_comp\_y\_ratio\_log2와 같은 정보를 전송할 수 있다.
- [0265] 예를 들어, 부호화기가 전송한 motion\_vector\_buffer\_comp\_x\_ratio\_log2의 값이 2이고, motion\_vector\_buffer\_comp\_y\_ratio\_log2의 값이 2라면, 4x4 샘플 단위로 움직임 정보를 저장하는 압축을 수행한다는 것을 지시할 수 있다.
- [0266] 또한, 부호화기가 전송한 motion\_vector\_buffer\_comp\_x\_ratio\_log2의 값이 3이고, motion\_vector\_buffer\_comp\_y\_ratio\_log2의 값이 2라면, 8x4 샘플 단위로 움직임 정보를 저장하는 압축을 수행한다는 것을 지시한다.
- [0267] 따라서, 부호화기가 처리하는 영상의 특성을 고려하여 움직임 정보의 압축 저장 비율을 결정하고 이를 복호화기에 전달할 수 있다. 복호화기는 전달된 움직임 정보의 압축 저장 비율에 따라서, TMVP를 유도할 때 움직임 벡터 및/또는 참조 인덱스 정보를 획득할 수 있다.
- [0268] 실시예 7에 의하면, 영상의 특성을 반영하여 동영상 정보를 압축 저장하므로, 시간적 움직임 벡터 버퍼의 압축에 있어서 손실을 최대한 줄일 수 있다.
- [0269] 이 경우에, 움직임 벡터 버퍼 압축에 관한 정보를 이용하여 참조 인덱스 버퍼 압축(메모리 압축)도 동일하게 수행할 수 있다. 또한, 참조 인덱스 버퍼 압축에 대해서는 별도로 정보를 시그널링하여 움직임 벡터 버퍼 압축과는 별도로 버퍼 압축이 수행되게 할 수도 있다.
- [0270] 한편, 움직임 정보를 저장하는 단위가 되는 샘플 블록의 크기 즉, 압축 저장 비율을 부호화기가 결정하여 복호화기에 시그널링하는 방법은 앞선 실시예 1 내지 6에도 동일하게 적용할 수 있다. 이 경우, 부호화기가 결정한 압축 저장의 단위가 되는 샘플 블록의 크기가 NxM 이라면, 앞서 설명한 실시예 1 내지 6에서

움직임 정보 저장 단위는 NxM 샘플 크기의 블록이 된다.

[0271]

[0272] <실시예 8>

[0273] 본 실시예에서는 움직임 정보 압축 저장에 사용되는 대표 움직임 정보 블록으로 움직임 정보 저장 단위 내의 움직임 정보 블록이 아니라 움직임 정보 저장 단위의 주변 블록을 이용하는 방법을 개시한다. 예컨대, 본 실시예에서는 16x16 화소 블록인 현재 움직임 정보 저장 단위 내의 어느 한 움직임 정보 블록을 대표 움직임 정보 블록으로 이용하는 것이 아니라, 현재 움직임 정보 저장 단위에 인접한 우하단의 4x4 화소 블록을 현재 움직임 정보 저장 단위에 대한 움직임 정보를 가지는 대표 움직임 정보 블록으로 사용하여 움직임 정보를 압축 저장할 수 있다.

[0274] 도 16은 움직임 정보를 압축 저장할 때, 본 발명에 따라서 현재 움직임 정보 저장 단위에 인접한 블록의 정보를 이용하는 방법을 개략적으로 설명하는 도면이다. 도 16은 움직임 정보 저장 단위가 4x4 개의 움직임 정보 블록으로 구성되는 경우를 예로서 설명하고 있다.

[0275] 도 16의 예에서 PU(1600)는 네 개의 움직임 정보 저장 단위(1610, 1620, 1630, 1640)를 포함하며, 각 움직임 정보 저장 단위는 4x4 화소 블록인 움직임 정보 블록들로 구성된다.

[0277] 본 실시예에 따라서, 16x16 화소 블록인 현재 움직임 정보 저장 단위(1610)의 움직임 정보(움직임 벡터, 참조 인덱스, 예측 모드 등)의 압축 저장을 위해, 현재 움직임 정보 저장 단위(1610)의 우하단 코너에 인접하여 위치하는 4x4 화소 블록 B1의 움직임 정보('움직임 데이터'라고도 함)를 현재 움직임 정보 저장 단위(1610)에 대한 움직임 정보로서 이용한다. 즉, 현재 움직임 정보 저장 단위(1610)에 대한 대표 움직임 정보 블록으로서 블록 B1을 이용한다.

[0278] 따라서, 현재 움직임 정보 저장 단위(1610) 내의 모든 블록에 대해서는 B1 블록의 움직임 정보가 할당된다.

[0279] PU(1600) 내의 다른 움직임 정보 저장 단위(1620, 1630, 1640)에 대해서도 동일한 방식으로 우하단 코너에 인접하는 주변 블록의 움직임 정보를 각 움직임 정보 저장 단위에 대한 움직임 정보로서 할당한다.

[0280] 따라서, 현재 블록에 대한 예측 과정에서 움직임 정보가 압축 저장된 참조 영상에서 TMVP를 가져오는 경우, 상술한 바와 같이 매번 현재 PU의 우하단에 있는 이웃 PU를 찾아내서 해당 PU가 속하는 움직임 정보 저장 단위의 좌상단 블록의 움직임 정보를 가져오는 추가 과정 없이, 단순히 현재 PU의 중앙에 위치하는 움직임 정보 블록들(중앙의 좌상, 중앙의 좌하, 중앙의 우상, 중앙의 우하에 위치하는 블록들)의 움직임 정보를 그대로 사용하면 된다.

[0281] 현재 PU의 중앙에 위치하는 움직임 정보 블록들은 도 16에서 B1, B2, B3, B4와 같다.

[0282] 이때, B1이 인트라 블록이거나 이용할 수 없는 블록인 경우에는, 현재 움직임

정보 저장 단위(1610)가 속하는 PU(1600)의 중앙에 위치하는 다른 움직임 정보 블록들(B2, B3, B4)를 현재 움직임 정보 저장 단위(1610)에 대한 대표 움직임 정보 블록으로 이용할 수 있다.

[0283] 이 경우에는, 현재 움직임 정보 저장 단위(1610)에 대하여, (1) B1 → B2 → B3 → B4의 순서, (2) B1 → B3 → B4 → B2의 순서, (3) B1 → B4 → B3 → B2의 순서 중 어느 한 순서에 따라서 움직임 정보 블록을 검색하여 첫 번째로 검색되는 인터 블록 또는 가용한 블록을 현재 움직임 정보 저장 단위(1610)에 대한 대표 움직임 정보 블록으로 이용할 수 있다. 어떤 순서로 검색할 것인지에 관해서는 미리 정해져 있을 수도 있고 부호화기로부터 시그널링될 수도 있다.

[0284]

[0285] <실시예 9>

[0286] 영상(픽처)의 폭(width)이나 높이(height)가 움직임 정보의 압축 저장 단위(움직임 정보 저장 단위)의 배수가 아닌 경우에는, 움직임 정보의 압축 저장 단위 보다 작은 크기의 블록에 대하여 움직임 정보를 압축 저장해야 하는 경우가 생기게 된다.

[0287] 본 실시예에서는 움직임 정보의 압축 저장 단위보다 작은 화소 블록에 대하여 움직임 정보를 압축 저장하는 방법을 개시한다.

[0288] 도 17은 움직임 정보의 압축 저장을 수행할 때, 본 발명에 따라서 움직임 정보의 압축 저장 단위(움직임 정보 저장 단위)보다 작은 화소 블록의 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다. 도 17은 영상의 크기가 움직임 정보 저장 단위의 배수가 되지 않는 경우의 일 예로서, 16x16 화소 샘플을 하나의 움직임 정보 저장 단위로 하는 경우를 설명하고 있다.

[0289] 도 17을 참조하면, 픽처(1700)는 다수의 움직임 정보 저장 단위를 포함한다. 도 17의 예에서 움직임 정보 저장 단위는 상술한 바와 같이 블록(1710)과 같은 16x16 화소 블록이다. 이때, 픽처(1700)의 크기와 움직임 정보 저장 단위의 크기가 비례하지 않기 때문에, 블록(1720)과 같이 픽처의 경계에서 움직임 정보 저장 단위보다 작은 크기의 블록(이하, 설명의 편의를 위해 ‘잔여 블록’이라 함)이 존재하게 된다. 도 17의 예에서는 잔여 블록의 크기를 8x16으로 가정한다.

[0290] MVP를 이용하여 인터 예측을 수행하기 위해서는 잔여 블록에 대해서도 움직임 정보를 압축 저장할 필요가 있다. 실시예 9에서는 움직임 정보 저장 단위에 대한 대표 움직임 정보 블록으로서 어떤 움직임 정보 블록을 이용하는지에 따라서 잔여 블록의 움직임 정보를 저장하는 방법을 설명한다. 대표 움직임 정보 블록으로서 이용할 수 있는 움직임 정보 블록의 위치는 실시예 1에서 설명한 바와 같다.

[0291] 이하, 실시예 1에서 특정한 대표 움직임 정보 블록의 위치에 따라서 잔여 블록에 대한 움직임 정보 압축 저장을 어떻게 처리할 것인지를 설명한다.

[0292] (1) 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 좌상단의 움직임 정보 블록인 경우

- [0293] 도 18은 대표 움직임 정보 블록이 움직임 정보 저장 단위의 좌상단 블록인 경우에, 본 발명에 따라서 잔여 블록에 대한 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0294] 도 18에서 픽처(1800) 내 움직임 정보 저장 단위(1810)는 16x16 화소 블록이고, 잔여 블록(1820)은 16x8 화소 블록이라고 가정한다. 이와 다른 사이즈의 움직임 정보 저장 단위 및 잔여 블록에 대해서도 도 18에서 설명하는 방법이 적용될 수 있음은 물론이다.
- [0295] 움직임 정보 저장 단위(1810)에 대해서는 좌상단의 움직임 정보 블록(1830)을 대표 움직임 정보 블록으로 하는 경우에, 움직임 정보를 압축하여 저장하기 위한 블록의 단위가 픽처(1800)의 경계 부근에서 작아져 잔여 블록(1820)과 같이 되더라도, 움직임 정보 저장 단위(1810)와 동일한 방법으로 잔여 블록(1820)에 대한 대표 움직임 정보 블록을 결정할 수 있다.
- [0296] 따라서, 잔여 블록(1820) 내 좌상단에 위치하는 움직임 정보 블록(1840)의 움직임 정보를 잔여 블록(1820)에 대한 움직임 정보로 이용한다.
- [0297] 따라서, 잔여 블록에 대하여 움직임 정보 저장 단위와 동일한 방법으로 대표 움직임 정보 블록을 결정할 수 있다.
- [0298] (2) 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 우하단의 움직임 정보 블록인 경우
- [0299] 도 19는 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 우하단 블록인 경우에, 본 발명에 따라서 잔여 블록에 대한 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0300] 도 19에서 픽처(1900) 내 움직임 정보 저장 단위(1910)는 16x16 화소 블록이고, 잔여 블록(1920)은 16x8 화소 블록이라고 가정한다. 이와 다른 사이즈의 움직임 정보 저장 단위 및 잔여 블록에 대해서도 도 19에서 설명하는 방법이 적용될 수 있음은 물론이다.
- [0301] 움직임 정보 저장 단위(1910)에 대해서는 우하단의 움직임 정보 블록(1930)을 대표 움직임 정보 블록으로 하는 경우에, 잔여 블록(1920)에 대해서도 잔여 블록(1920) 내 이용 가능한 영역 중 우하단에 위치하는 움직임 정보 블록(1940)의 움직임 정보를 잔여 블록(1920)에 대한 움직임 정보로 이용한다.
- [0302] 도 20은 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 우하단 블록이지만, 잔여 블록 내에 이용할 수 없는 영역이 존재하는 경우에 잔여 블록의 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0303] 구체적으로, 잔여 블록(1920) 내에 이용할 수 없는(not available) 영역이 존재하는 경우에는 잔여 블록(1920)의 좌상단에 위치하는 움직임 정보 블록(2000)의 움직임 정보를 잔여 블록(1920)에 대한 움직임 정보로 이용한다. 이 경우, 좌상단의 영역은 항상 이용 가능하기 때문에 이용 가능한 블록의 경계를 찾아내기 위한 연산을 수행할 필요가 없다는 효과도 아울러 거둘 수 있다.
- [0304] (3) 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 중앙에 위치하는 움직임

정보 블록들 중 어느 하나인 경우

- [0305] 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 중앙에 위치하는 움직임 정보 블록들 중 어느 하나인 경우라 함은, 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 중앙의 좌상, 중앙의 좌하, 중앙의 우상, 중앙의 우하에 위치하는 움직임 정보 블록 중 어느 하나로 특정되는 경우를 의미한다. 이하, 설명의 편의를 위해, 중앙의 움직임 정보 블록의 위치를 Center\_XX로 표시한다. XX는 LT(Left-Top), RT(Right-Top), LB(Left-Bottom) 혹은 RB(Right-Bottom) 중 어느 하나를 나타낸다. XX가 LT, RT, LB, RB 중 어느 것에 해당하는지는 미리 정해져 있을 수도 있고, 부호화기에서 정해서 복호화기로 시그널링 될 수도 있다. 이하, 설명의 편의를 위해, XX가 LT인 경우를 예로서 설명한다. 다만, 이하의 설명 내용이 XX가 RT, RB, LB인 경우에도 동일하게 적용될 수 있음은 물론이다.
- [0306] 도 21은 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 중앙 좌상단(Center\_LT)의 블록인 경우에, 본 발명에 따라서 잔여 블록에 대한 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0307] 도 21에서 픽처(2100) 내 움직임 정보 저장 단위(2110)는 16x16 화소 블록이고, 잔여 블록(2120)은 16x8 화소 블록이라고 가정한다. 이와 다른 사이즈의 움직임 정보 저장 단위 및 잔여 블록에 대해서도 도 21에서 설명하는 방법이 동일하게 적용될 수 있다.
- [0308] 움직임 정보 저장 단위(2110)에 대해서는 중앙 좌상단의 움직임 정보 블록(2130)을 대표 움직임 정보 블록으로 하는 경우에, 잔여 블록(2120)에 대해서도 잔여 블록(2120) 내 이용 가능한 영역 중 중앙 좌상단에 위치하는 움직임 정보 블록(2140)의 움직임 정보를 잔여 블록(2120)에 대한 움직임 정보로 이용한다.
- [0309] 도 22는 대표 움직임 정보 블록이 움직임 정보 저장 단위 내 중앙 좌상단 블록이지만, 잔여 블록 내에 이용할 수 없는 영역이 존재하는 경우에 잔여 블록의 움직임 정보를 압축 저장하는 방법을 개략적으로 설명하는 도면이다.
- [0310] 구체적으로, 잔여 블록(2120) 내에 이용할 수 없는(not available) 영역이 존재하는 경우에는 잔여 블록(2120)의 좌상단에 위치하는 움직임 정보 블록(2200)의 움직임 정보를 잔여 블록(2120)에 대한 움직임 정보로 이용할 수 있다.
- [0311] 또한, 잔여 블록(2120) 내에 이용할 수 없는(not available) 영역이 존재하되, 이용할 수 없는 영역이 잔여 블록(2120) 내 중앙 좌상단의 영역인 경우에만 잔여 블록(2120)의 좌상단에 위치하는 움직임 정보 블록(2200)의 움직임 정보를 잔여 블록(2120)에 대한 움직임 정보로 이용하게 할 수도 있다.
- [0312] 도 22에서 설명하는 방법의 경우에는, 좌상단의 영역은 항상 이용 가능하기 때문에 이용 가능한 블록의 경계를 찾아내기 위한 연산을 수행할 필요가 없다는 효과도 아울러 거둘 수 있다.
- [0313]

- [0314] 복호화기와 복호화기는 인터 예측을 위해 도 5에서와 같이 TMVP를 유도하는 과정에서, 현재 블록에 대한 col 블록의 우하단에 인접한 코너에 위치하는 화소가 속하는 움직임 정보 저장 단위의 움직임 정보를 TMVP의 움직임 정보로 이용한다. 부호화기와 복호화기는 움직임 정보 저장 단위 내 PU들의 움직임 정보로서, 움직임 정보 저장 단위 내 움직임 정보 블록들 중 대표 움직임 정보 블록의 움직임 정보를 이용할 수 있다. 움직임 정보 저장 단위 내 대표 움직임 정보 블록의 움직임 정보는, 움직임 벡터 외에도 참조 인덱스, 예측 모드를 포함할 수 있다.
- [0315]
- [0316] 한편, 움직임 벡터 예측(motion vector prediction)에서는 상술한 바와 같이 현재 블록에 이웃하는 블록의 움직임 정보를 MVP 후보로 사용할 뿐만 아니라, 인접 프레임(픽처)의 동일 위치 움직임 벡터도 MVP 후보로 사용할 수 있다. 이를 앞서 설명한 바와 같이 TMVP(Temporal Motion Vector Predictor)라 한다.
- [0317] TMVP를 이용하기 위해서는 복호화한 정보를 저장해 두어야 한다. 프레임(픽처)를 복호화한 후 소정의 블록 단위(예컨대, 4x4 화소 블록)로 움직임 벡터 등의 움직임 정보를 저장해 두어야 현재 블록이 속하는 다음 프레임에서 이를 기반으로 TMVP를 이용할 수 있다.
- [0318] 하지만, 4x4의 최소 단위로 움직임 정보를 모두 저장하려면 메모리를 많이 사용해야 하므로, 더 큰 단위(예컨대, 16x16 화소 블록, 32x32 화소 블록, 64x64 화소 블록 단위 중 어느 하나)로 움직임 정보를 압축하여 저장할 필요가 있다. 이를 앞서 움직임 정보 저장 감축, 움직임 벡터 메모리 압축 또는 움직임 벡터 버퍼 압축이라고 설명하였으며, 움직임 정보를 압축하여 저장하는 단위를 움직임 정보 저장 단위라고 설명한 바 있다. 이때, 어느 단위로 움직임 정보를 압축하여 저장할 것인지는 미리 정해져 있을 수도 있고, 부호화기에서 결정되어 SPS(Sequence Parameter Set) 레벨의 시그널링을 통해서 복호화기에 전달될 수도 있다.
- [0319] 예컨대, 설명의 편의를 위해 16x16 화소 블록 단위로 움직임 정보를 압축 저장하는 경우를 예로 든다면, 16x16 화소 블록의 좌상단에 있는 4x4 블록의 움직임 정보(예컨대, 움직임 벡터 및 참조 인덱스)를 L0, L1 방향에 대해 저장할 수 있다. 이때, 좌상단에 있는 블록 대신에 16x16 화소 블록의 중앙에 있는 4x4 화소 블록의 움직임 정보 정보를 이용할 수도 있다.
- [0320] 도 23은 현재 프레임(또는 픽처, 2300)에 이웃하는 프레임(또는 픽처, 2310)에서 현재 블록(2320)과 동일한 위치에 있는 블록(2330)의 좌상단(2340)에 위치하는 움직임 정보가 아니라, 중앙(2350)에 위치하는 움직임 정보를 TMVP로 이용하는 예를 개략적으로 설명하는 도면이다.
- [0321] 이와 관련한 내용들은 이미 상술한 바 있다.
- [0322] 또한, 앞서 설명한 바와 같이, 사용하고자 하는 TMV(Temporal Motion Vector)가 인트라 블록에 포함된 움직임 벡터라면 현재 블록의 인터 예측에 유용한

정보라고 할 수 없다. 일반적으로 인터 블록으로부터 도출되는 TMV는 (0,0)의 값을 가지는 것으로 처리할 수 있다.

[0323] 도 24는 TMVP를 유도할 때 중앙에 위치하는 인터 블록의 움직임 정보를 처리하는 방법을 개략적으로 설명하는 도면이다.

[0324] 도 24에서, 현재 블록(2410)이 TMVP를 위해 이용할 블록이 이웃 프레임(2420)의 동일 위치 블록(2430) 내 중앙의 블록(2440)이라고 가정하자. 이때, 블록(2440)이 인트라 블록인 경우라면, 블록(2440)으로부터 유도되는 움직임 벡터의 값은 (0, 0)으로 처리되며 부정확한 예측이 수행될 수 있다. 이때, 이웃하는 블록(2450)이 인터 블록이고 정상적인 움직임 벡터를 가지고 있는 상황이라면, 굳이 블록(2440)의 부정확한 움직임 벡터를 이용할 필요가 없다.

[0325] 한편, 도 24의 경우와 같이, TMVP를 유도하기 위해 이웃 프레임(colPic)에서 현재 블록과 동일한 위치에 있는 블록(col 블록)의 중앙에 위치하는 블록의 움직임 정보를 이용할 때, 블록의 크기에 따라서 정확히는 동일 위치 블록(col 블록)의 중앙에 있는 블록이 아니라, 중앙을 둘러싼 4 개의 블록 중 어느 하나(예컨대 중앙 좌상의 블록)를 이용할 수 있다.

[0326] 도 23 및 도 24에 관한 내용은 이미 앞서 설명한 바 있으나, 이와 관련하여, TMVP를 유도하기 위한 실시예를 이하 도면과 함께 설명한다.

[0327]

[0328] <실시예 10>

[0329] 본 실시예에서는 다양한 사이즈의 현재 블록에 대하여 TMVP를 유도하고자 할 때, col 블록의 중앙에 위치하는 블록들의 움직임 정보를 이용하는 방법을 설명한다.

[0330] 도 25는 본 발명에 따라서 참조 픽처(참조 프레임)에서 col 블록의 중앙에 위치하는 블록의 움직임 정보를 기반으로 TMVP를 유도하는 방법을 개략적으로 설명하는 방법이다. 도 25에서는 설명의 편의를 위해 움직임 정보(예컨대, 움직임 벡터)의 저장 단위(움직임 정보 저장 단위)가 16x16 화소 블록인 경우를 예로서 설명한다. 상술한 바와 같이, 움직임 정보 저장 단위의 크기는 미리 정해져 있을 수 있고, 부호화기에서 결정되어 SPS 레벨의 시그널링을 통해 복호화기에 전달될 수도 있다.

[0331] 도 25(a)는 현재 블록 및 현재 블록에 대한 col 블록(2500)이 64x64 화소 블록인 경우를 설명하고 있다. 도 25(a)에서 현재 블록의 TMVP를 유도하기 위해 col 블록(2500)의 중앙에 위치하는 움직임 정보 저장 단위인 블록 Bs0, Bs1, Bs2, Bs3을 이용한다.

[0332] 구체적으로, col 블록으로부터 TMVP를 유도할 때, (1) col 블록 중앙의 좌상(left-top)에 위치하는 움직임 정보 저장 단위 Bs0의 움직임 정보, (2) col 블록 중앙의 우상(right-top)에 위치하는 움직임 정보 저장 단위 Bs1의 움직임 정보, (3) col 블록 중앙의 좌하(left-bottom)에 위치하는 움직임 정보 저장 단위 Bs2의 움직임 정보, (4) col 블록 중앙의 우하(right-bottom)에 위치하는 움직임 정보 저장

단위 Bs3의 움직임 정보 중 어느 하나를 TMVP를 유도하는데 이용한다. (1) 내지 (4) 중 어떤 움직임 정보를 이용할 것인지는 미리 정해져 있을 수도 있고, 부호화기에서 결정하여 복호화기로 시그널링될 수도 있다.

[0333] 움직임 정보 저장 단위에 움직임 정보를 저장하는 방법(움직임 정보 저장 감축)에 관해서는 실시예 1 내지 실시예 9를 통해 설명하였으며, 본 실시예에서는 실시예 1 내지 실시예 9 중 어느 하나의 방법에 의해 저장된 움직임 정보를 이용할 수 있다.

[0334] 도 25(b)는 도 25(a)와 달리 현재 블록 및 현재 블록에 대한 col 블록(2510)이 64x32 화소 블록인 경우를 설명하고 있다. 도 25(b)에서도 현재 블록의 TMVP를 유도하기 위해 col 블록(2510)의 중앙에 위치하는 움직임 정보 저장 단위인 블록 Bs0, Bs1, Bs2, Bs3을 이용한다.

[0335] 구체적으로, col 블록(2510)으로부터 TMVP를 유도할 때, (1) col 블록 중앙의 좌상(left-top)에 위치하는 움직임 정보 저장 단위 Bs0의 움직임 정보, (2) col 블록 중앙의 우상(right-top)에 위치하는 움직임 정보 저장 단위 Bs1의 움직임 정보, (3) col 블록 중앙의 좌하(left-bottom)에 위치하는 움직임 정보 저장 단위 Bs2의 움직임 정보, (4) col 블록 중앙의 우하(right-bottom)에 위치하는 움직임 정보 저장 단위 Bs3의 움직임 정보 중 어느 하나를 TMVP를 유도하는데 이용한다. (1) 내지 (4) 중 어떤 움직임 정보를 이용할 것인지는 미리 정해져 있을 수도 있고, 부호화기에서 결정하여 복호화기로 시그널링될 수도 있다.

[0336]

[0337] <실시예 11>

[0338] 본 실시예에서는 현재 블록에 대한 TMVP를 유도할 때, 인트라 블록의 존부를 고려하는 방법을 설명한다.

[0339] 도 26은 본 발명에 따라서 인트라 블록의 유무를 고려하여 TMVP를 유도하는 방법을 개략적으로 설명하는 도면이다. 도 26에서는 현재 블록에 오버랩하여 현재 블록과 동일 위치에 있는 참조 프레임의 TMVP 후보들을 도시하고 있다. 따라서, 블록(2600), 블록(2610), 블록(2620)은 현재 블록으로 볼 수도 있고, 현재 블록의 col 블록으로도 볼 수 있으나, 여기서는 설명의 편의를 위해 현재 블록의 col 블록으로 가정하고 설명하도록 한다.

[0340] 도 26(a)는 현재 블록의 col 블록에서 중앙에 위치하는 움직임 정보 저장 단위를 이용하여 TMVP를 유도하는 것을 설명하는 도면이다. 예컨대 도 26(a)의 예에서 TMVP 후보군은 움직임 정보 저장 단위 Bs0, Bs1, Bs2, Bs3의 움직임 벡터가 된다. 도 26(a)에서는 16x16 화소 블록을 움직임 정보 저장 단위로 하는 경우를 예로서 설명한다.

[0341] 도 26(a)에서는 움직임 정보 저장 단위가 인트라 블록인지 인터 블록인지에 상관없이 해당 움직임 정보 저장 단위의 움직임 벡터를 TMVP로 이용하는 것이 아니라, 인트라 블록인지 인터 블록인지를 판단하여 인터 블록인 움직임 정보 저장 단위의 움직임 정보를 TMVP에 이용한다.

- [0342] 따라서, 복호화기는 col 블록 중앙에 위치하는 4 개의 움직임 정보 저장 단위(Bs0, Bs1, Bs2, Bs3)를 소정의 검색 순서에 따라서 검색하고, 가장 먼저 인터 블록인 것으로 검색되는 움직임 정보 저장 단위의 움직임 정보를 이용하여 TMVP를 유도할 수 있다. 만약 4개의 움직임 정보 저장 단위(Bs0, Bs1, Bs2, Bs3)가 모두 인트라 블록(혹은 이용 불가)이라면, 움직임 정보 저장 단위 Bs0의 움직임 정보를 이용하여 TMVP가 유도되도록 할 수도 있다.
- [0343] 이때, 검색의 순서는 (1) Bs0 → Bs1 → Bs2 → Bs3, (2) Bs2 → Bs3 → Bs0 → Bs1, (3) Bs0 → Bs3 → Bs1 → Bs2, (4) Bs0 → Bs2 → Bs1 → Bs3 등과 같이 다양하게 정해질 수 있다. 검색의 순서는 미리 결정될 수도 있고, 부호화기에서 결정되어 복호화기에 시그널링될 수도 있다.
- [0344] 도 26(b)는 도 26(a)에서 TMVP 후보군의 수를 더 늘린 경우를 설명하고 있다. 도 26(b)에서는 col 블록 내 모든 움직임 벡터 저장 단위의 움직임 정보를 TMVP 후보군으로 하고 있다.
- [0345] 이때, 검색의 순서는 다양하게 결정될 수 있다. 예컨대, 움직임 정보 저장 단위의 인덱스 순서를 따라서, Bs0부터 시계 방향 나선형으로 검색할 수도 있고, 반시계 방향 나선형으로 검색할 수도 있다. 또한, col 블록 중앙의 네 움직임 정보 저장 단위 중 Bs0이 아닌 다른 블록을 시작점으로 하여 검색을 진행할 수도 있다(예컨대, Bs2 → Bs3 → Bs0 → Bs1 → Bs10 → Bs11 → Bs12 → Bs13 → Bs14 → Bs15 → Bs4 → Bs5 → Bs6 → Bs7 → Bs8 → Bs9).
- [0346] 도 26(c)는 현재 블록에 대한 col 블록의 외부에 있는 블록의 움직임 정보를 이용하여 TMVP를 유도하는 방법을 설명하는 도면이다. TMVP 후보가 col 블록(현재 블록)의 경계를 벗어나 있을 수도 있고, col 블록(현재 블록)의 경계를 벗어나 있는 TMVP의 후보가 col 블록(현재 블록) 내의 TMVP 후보보다 더 우선 순위가 높을 수도 있다. 도 26(c)에서도 설명의 편의를 위해 16x16 화소 블록을 움직임 정보 저장 단위로 하는 경우를 예로서 설명한다.
- [0347] 도 26(c)에서는 설명의 편의를 위해 두 개의 TMVP 후보군을 이용한다. 현재 블록에 대한 col 블록(2620) 외부의 블록으로서 col 블록(2620)의 우하단(right-bottom)에 인접하는 블록(Bs0)과 col 블록(2620)의 중앙에 위치하는 블록들 중 중앙 우하단(center right-bottom)에 위치하는 움직임 정보 저장 단위(Bs1)의 움직임 정보를 TMVP 후보로서 이용할 수 있다. 이때, 블록 Bs0은 움직임 정보 저장 단위일 수 있다.
- [0348] 복호화기는 우선 col 블록 외부의 블록 Bs0이 인터 블록인지(혹은 이용 가능한 블록인지) 인트라 블록인지(혹은 이용할 수 없는 블록인지)를 검색한다. 블록 Bs0이 인터 블록으로서 이용 가능한 경우에, 복호화기는 블록 Bs0의 움직임 정보를 이용하여 TMVP를 유도한다. 예컨대, 블록 Bs0의 움직임 정보를 TMVP로 이용할 수 있다.
- [0349] 블록 Bs0이 인트라 블록 혹은 이용할 수 없는 블록인 경우에, 복호화기는 움직임 정보 저장 단위 Bs1이 인터 블록인지(혹은 이용 가능한 블록인지) 인트라

블록인지(혹은 이용할 수 없는 블록인지)를 검색한다. 블록 Bs1이 인터 블록으로서 이용 가능한 경우에, 복호화기는 블록 Bs1의 움직임 정보를 이용하여 TMVP를 유도한다.

[0350] 블록 Bs0과 Bs1이 모두 인트라 블록인 경우에, 복호화기는 Bs0의 움직임 정보를 기반으로 TMVP를 유도할 수 있다.

[0351] 도 26에서 TMVP의 후보군 개수와 위치, 검색의 순서는 설명한 바에 한정되지 않으며, 움직임 정보 저장 단위의 사이즈 역시 16x16에 한정되지 않는다. 또한, 현재 블록과 col 블록의 사이즈도 64x64에 한정되지 않으며, 32x32 등과 같은 다양한 사이즈에도 본 발명이 적용될 수 있다. 예컨대, 현재 블록이 32x32 사이의 블록이고 움직임 정보 저장 단위의 사이즈가 16x16이라면, 현재 블록에 대하여 최대 4개의 TMVP 후보가 이용될 수 있다.

[0352] 현재 블록의 크기가 움직임 벡터 메모리 버퍼, 즉 움직임 정보 저장 단위보다 작은 경우에는 기존의 방법대로 col 블록 내 좌상단의 움직임 정보 등을 이용하여 TMVP를 유도할 수도 있다.

[0353]

[0354] 상술한 설명에서 블록 인덱스(BI 또는 BsI, I=0, 2, 3, ...)는 블록을 특정하기 위한 것으로서, 특별한 설명이 없는 한 검색 순서, 처리 순서 등에서 우선 순위를 나타내기 위한 것이 아니다.

[0355] 또한, 상술한 실시예들은 인터 예측에서 움직임 정보 저장 감축 방법과 저장된 움직임 정보를 이용하여 TMVP를 유도하는 방법에 관한 것으로서, 부호화기와 복호화기에서 동일한 예측이 수행될 수 있도록 부호화기와 복호화기에 동일하게 적용될 수 있다. 따라서, 상술한 실시예들은 설명의 편의를 위해 복호화기의 동작으로 기술되어 있으나, 부호화기에도 동일하게 적용된다. 예컨대, 본 발명은 상술한 실시예들에서 ‘복호화기’를 ‘부호화기’로 대체하는 경우를 포함한다.

[0356] 또한, 상술한 설명에서는 상황에 따라서 설명의 편의를 위해, ‘움직임 정보 저장 감축’, ‘움직임 벡터 메모리 압축’, ‘움직임 벡터 버퍼 압축’, ‘움직임 벡터 압축 저장’, ‘움직임 정보 압축 저장’ 등을 혼용하였으나, 이는 설명의 편의를 위한 것으로서 동일한 의미로 사용될 수 있다. 아울러, 상술한 설명에서 설명의 편의를 위해 사용된 용어들은 사용된 명칭뿐만 아니라 도면 및 관련 설명으로부터 정의될 수 있다.

[0357]

[0358] 상술한 예시적인 시스템에서, 방법들은 일련의 단계 또는 블록으로써 순서도를 기초로 설명되고 있지만, 본 발명은 단계들의 순서에 한정되는 것은 아니며, 어떤 단계는 상술한 바와 다른 단계와 다른 순서로 또는 동시에 발생할 수 있다. 또한, 상술한 실시예들은 다양한 양태의 예시들을 포함한다. 따라서, 본 발명은 상술한 실시예들을 동시에 적용하거나 조합하여 적용하는 실시 형태를 포함한다.

[0359] 지금까지 본 발명에 관한 설명에서 일 구성 요소가 타 구성 요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 상기 일 다른 구성 요소가 상기 타 구성 요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 상기 두 구성 요소 사이에 다른 구성 요소가 존재할 수도 있다고 이해되어야 한다. 반면에, 일 구성 요소가 타 구성 요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 두 구성 요소 사이에 다른 구성요소가 존재하지 않는 것으로 이해되어야 한다. 아울러, 본 발명의 기술적 범위는 각 실시예들을 동시에 적용하거나 조합하여 적용하는 것을 포함한다.

[0360]

[0361]

[0362]

## 청구범위

- [청구항 1] 현재 픽처에 대한 동일 위치 픽처에서 움직임 정보 저장 단위 내 블록들 중 대표 블록을 결정하는 단계; 및  
상기 대표 블록의 움직임 정보를 상기 움직임 정보 저장 단위에 대한 움직임 정보로서 저장하는 단계를 포함하며,  
상기 움직임 정보 저장 단위에 포함되는 블록들의 움직임 정보로서 상기 대표 블록의 움직임 정보를 이용하는 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 2] 제1항에 있어서, 상기 대표 블록의 움직임 정보는 움직임 벡터와 참조 인덱스를 포함하며,  
상기 움직임 정보 저장 단위에 포함되는 블록들의 움직임 벡터와 참조 인덱스로서 상기 대표 블록의 움직임 벡터와 참조 인덱스를 이용하는 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 3] 제2항에 있어서, 상기 움직임 정보 저장 단위에 포함되는 블록들 중 인트라 예측 모드로 부호화된 블록들에 대해서는 인트라 블록임을 지시하는 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 4] 제3항에 있어서, 상기 움직임 정보 저장 단위에 포함되는 블록들 중 인트라 예측 모드로 부호화된 블록들에 대해서는 상기 대표 블록의 참조 인덱스 대신 인트라 블록임을 지시하는 참조 인덱스를 할당하는 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 5] 제3항에 있어서, 상기 움직임 정보 저장 단위에 포함되는 블록들 중 인트라 예측 모드로 부호화된 블록들은 코딩 블록의 예측 타입에 의해 인트라 블록임을 지시되는 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 6] 제1항에 있어서, 상기 대표 블록의 움직임 정보는 움직임 벡터, 참조 인덱스 및 예측 모드를 포함하며,  
상기 움직임 정보 저장 단위에 포함되는 블록들의 움직임 벡터, 참조 인덱스 및 예측 모드로서 상기 대표 블록의 움직임 벡터, 참조 인덱스 및 예측 모드를 이용하는 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 7] 제1항에 있어서, 상기 대표 블록의 움직임 정보는 움직임 벡터, 참조 인덱스 및 예측 모드를 포함하며,  
상기 움직임 정보 저장 단위에 포함되는 블록들의 움직임 벡터, 참조 인덱스 및 예측 모드로서 상기 대표 블록의 움직임 벡터, 참조 인덱스 및 예측 모드를 이용하되,  
상기 움직임 정보 저장 단위에 포함되는 블록들 중 인트라 예측 모드로 부호화된 블록에 대해서는 인트라 예측 모드임을 지시하는

- 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 8] 제1항에 있어서, 상기 대표 블록은 상기 움직임 정보 저장 단위에 속하는 블록들 중 좌상단의 블록인 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 9] 제1항에 있어서, 상기 대표 블록은, 상기 움직임 정보 저장 블록 내의 블록들을 소정의 검색 순서에 따라서 검색하여 가장 먼저 검색한 인터 블록인 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 10] 제9항에 있어서, 상기 검색 순서는 상기 움직임 정보 저장 단위 내 좌상단 블록에서부터 시작되는 래스터 스캔 순서인 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 11] 제9항에 있어서, 상기 검색 순서는 상기 움직임 정보 저장 단위 내 중앙의 블록에서부터 나선 방향으로 검색해 나가는 순서인 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 12] 제1항에 있어서, 상기 대표 블록은, 상기 움직임 정보 저장 단위에 속하는 블록들을 소정의 검색 순서에 따라서 검색하여 가장 먼저 검색되는 인터 블록으로서 참조 인덱스의 값이 0인 블록인 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 13] 제1항에 있어서, 상기 움직임 정보 저장 단위 내의 모든 블록들이 인트라 예측 모드로 부호화된 블록들인 경우에는, 상기 움직임 정보 저장 단위에 인접한 블록들 중에서 인트라 예측 모드로 부호화된 블록을 상기 대표 블록으로 선택하는 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 14] 제1항에 있어서, 상기 대표 블록은, 상기 움직임 정보 저장 단위의 중앙에 위치하는 네 블록 중에서 우하(Right-Bottom)에 위치하는 블록인 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 15] 제1항에 있어서, 상기 대표 블록은, 상기 움직임 정보 저장 단위의 인접 블록 중 상기 움직임 정보 저장 단위의 우하측 코너에 인접하는 블록인 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 16] 제1항에 있어서, 상기 움직임 정보 저장 단위의 사이즈는 부호화기에서 결정되어 복호화기로 시그널링되는 것을 특징으로 하는 움직임 정보 저장 방법.
- [청구항 17] 제1항에 있어서, 픽처 경계에 위치하는 경계 움직임 정보 저장

단위의 크기가 상기 픽처 내부에 위치하는 내부 움직임 정보 저장 단위보다 작고,

상기 내부 움직임 정보 저장 단위에 대한 대표 블록의 위치와 동일한 위치의 블록을 상기 경계 움직임 정보 저장 단위의 대표 블록으로 이용할 수 없는 경우에는,

상기 경계 움직임 정보 저장 단위 내의 블록들 중에서 좌상단에 위치하는 블록을 상기 경계 움직임 정보 저장 단위에 대한 대표 블록으로 결정하는 것을 특징으로 하는 움직임 정보 저장 방법.

[청구항 18]

현재 픽처의 동일 위치 픽처를 결정하는 단계;

상기 동일 위치 픽처에서 현재 블록에 대한 동일 위치 예측 유닛을 결정하는 단계;

상기 동일 위치 예측 유닛의 움직임 정보를 상기 현재 블록에 대한 시간적 움직임 벡터 예측자로 설정하는 단계를 포함하며,

상기 동일 위치 예측 유닛은,

상기 동일 위치 예측 유닛에 대한 위치 결정 화소가 속하는 움직임 정보 저장 단위 내 좌상단에 위치하는 예측 유닛이고,

상기 위치 결정 화소는 상기 동일 위치 픽처에서 현재 블록에 대한 동일 위치 블록의 우하단 코너에 인접하는 화소인 것을 특징으로 하는 시간적 움직임 벡터 예측자 유도 방법.

[청구항 19]

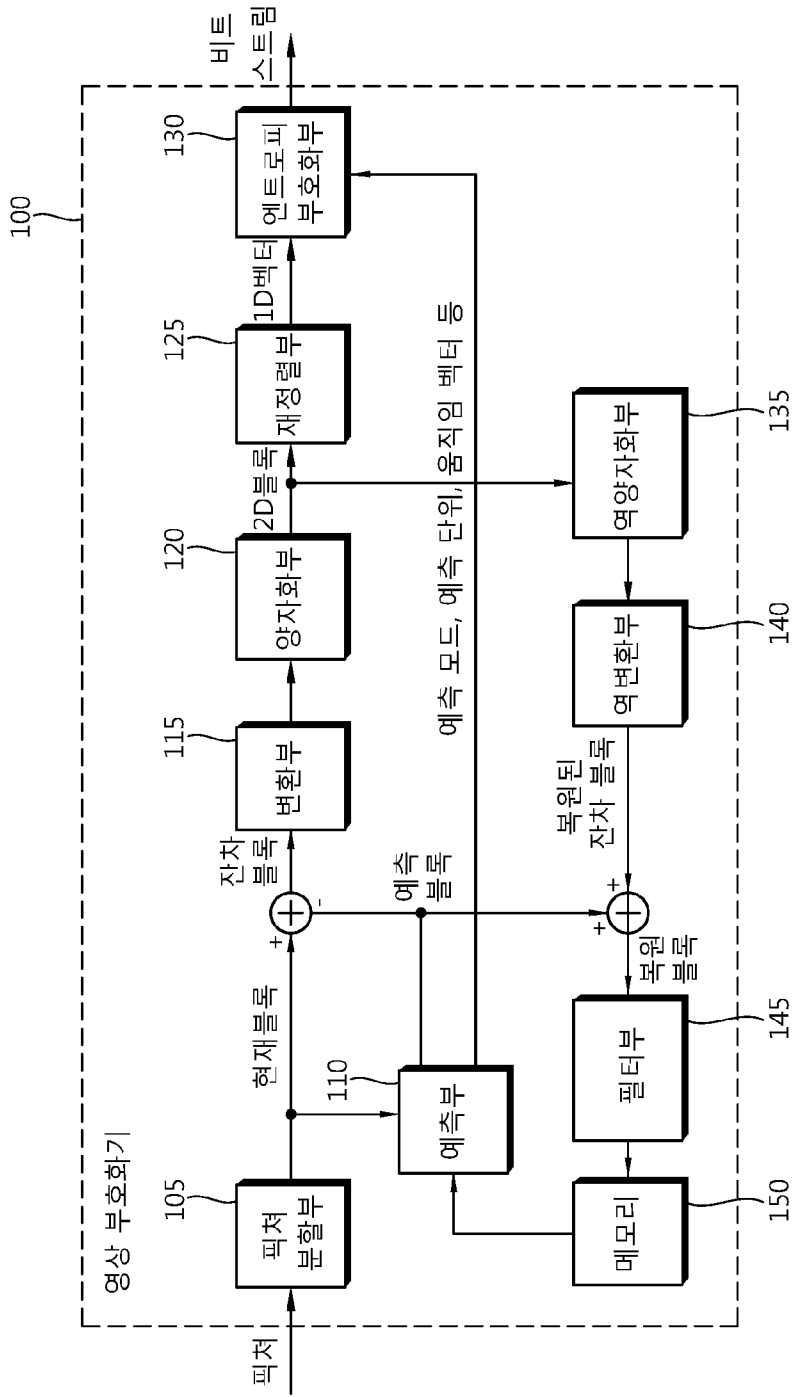
제18항에 있어서, 상기 동일 위치 블록의 우하단 코너에 인접하는 화소가 인터 예측에 이용할 수 없는 화소인 경우에는,

상기 동일 위치 블록의 중앙에 위치하는 네 화소 중 우하단의 화소를 위치 결정 화소로 선택하는 것을 특징으로 하는 시간적 움직임 벡터 예측자 유도 방법.

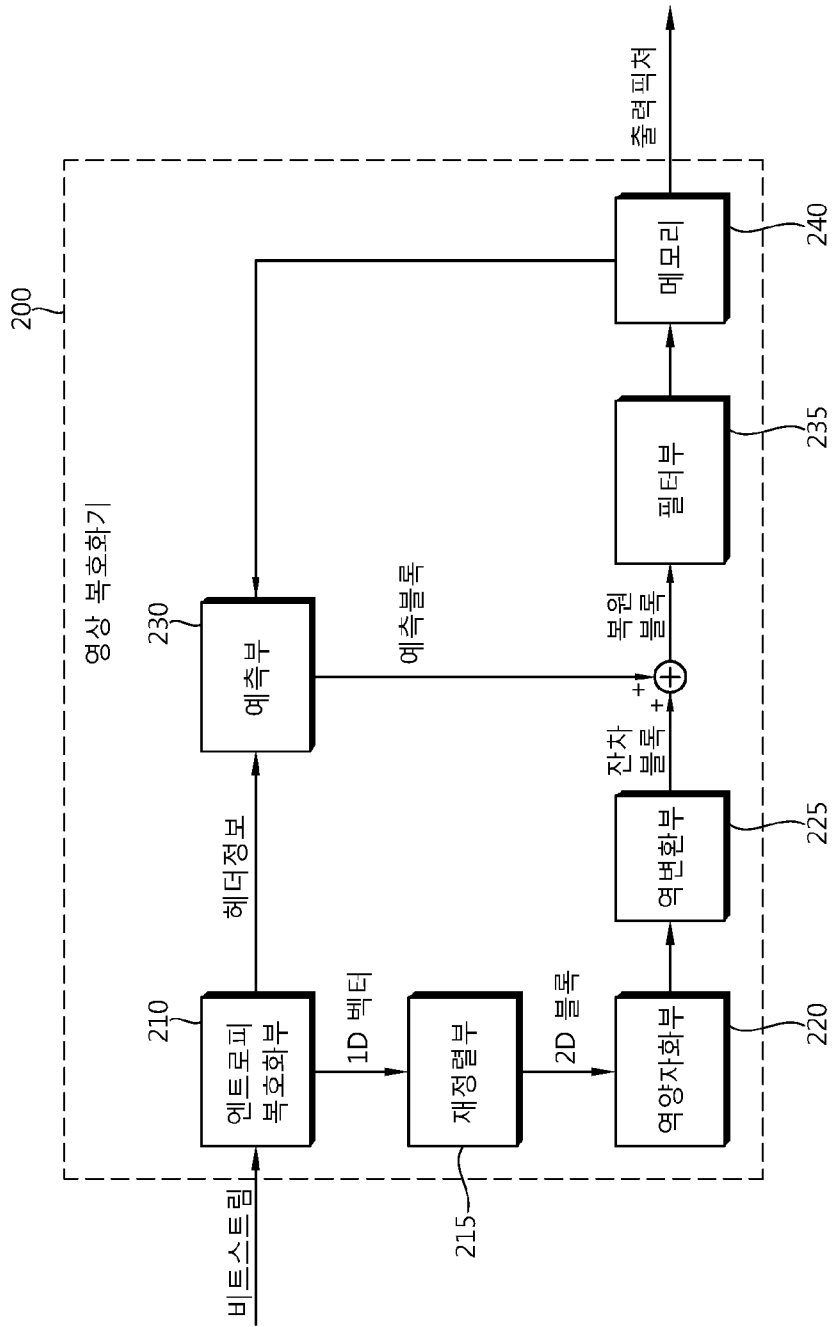
[청구항 20]

제18항에 있어서, 동일 위치 예측 유닛은 상기 움직임 정보 저장 단위의 중앙에 위치하는 예측 유닛 또는 상기 움직임 정보 저장 단위의 우하단 코너에 인접하는 예측 유닛 중 어느 하나인 것을 특징으로 시간적 움직임 벡터 예측자 유도 방법.

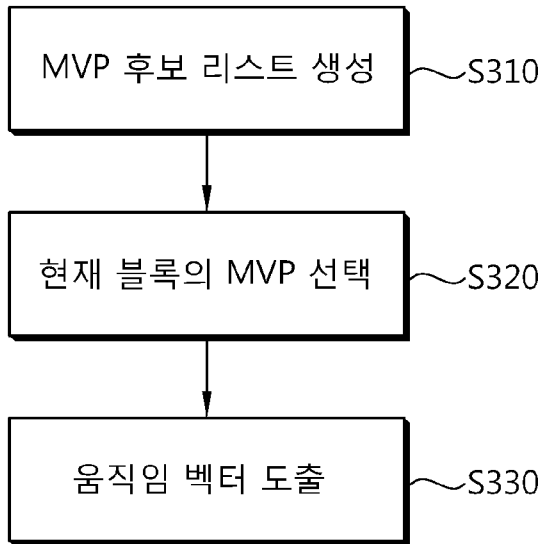
[Fig. 1]



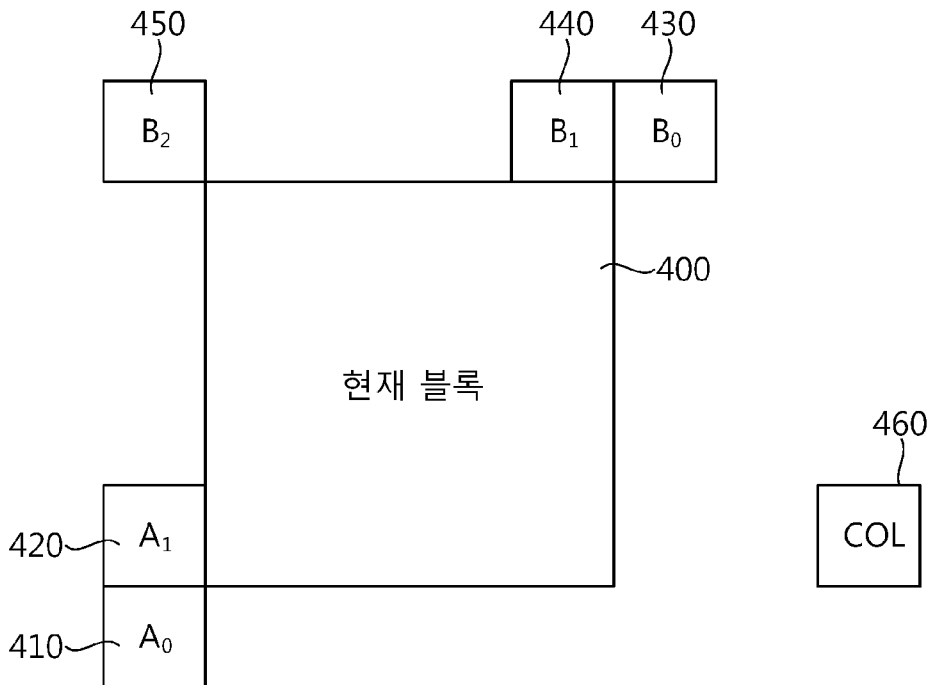
[Fig. 2]



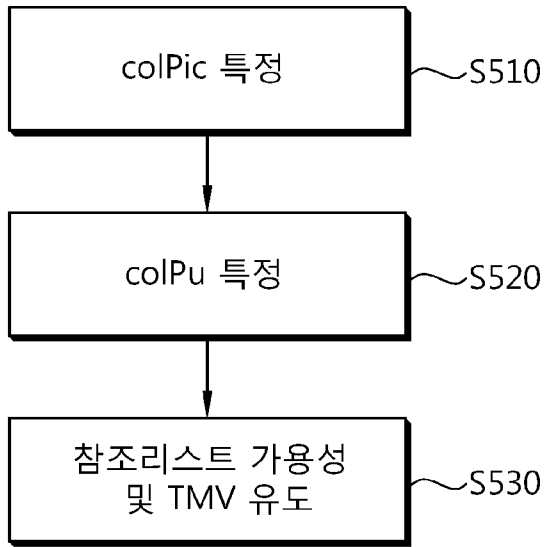
[Fig. 3]



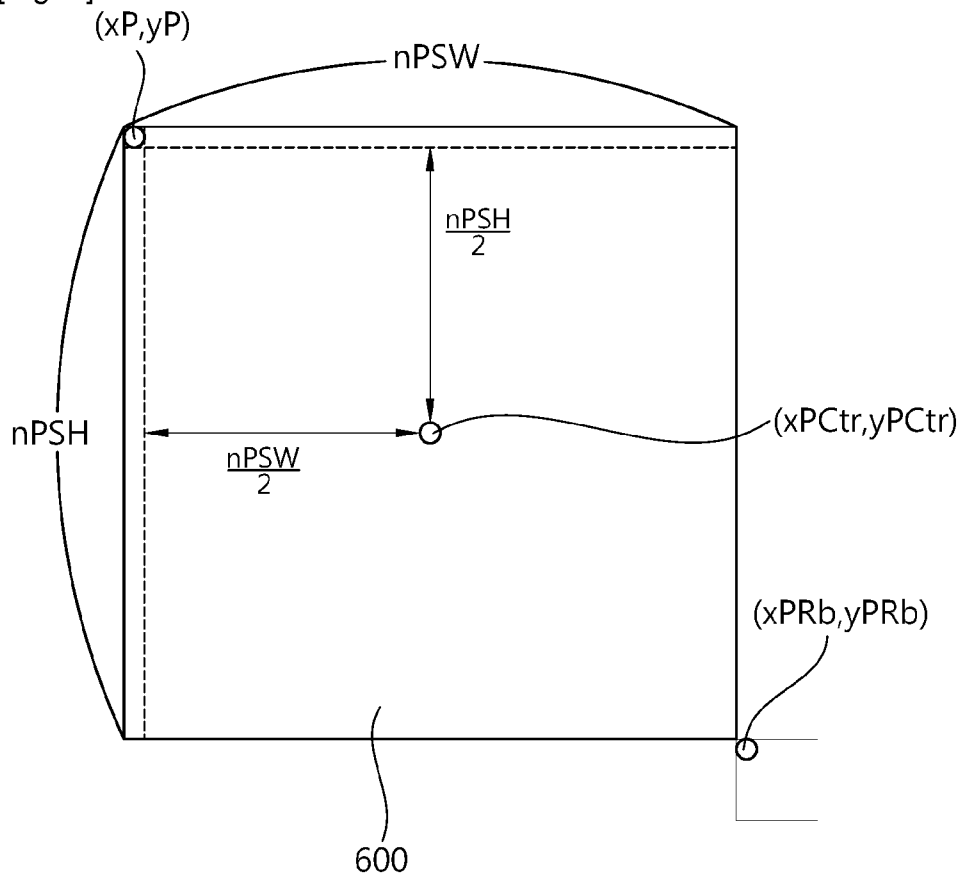
[Fig. 4]



[Fig. 5]



[Fig. 6]



[Fig. 7]

(X0,Y0) B0	(X1,Y1) B1	(X2,Y2) B2	(X3,Y3) B3
(X4,Y4) B4	(X5,Y5) B5	(X6,Y6) B6	(X7,Y7) B7
(X8,Y8) B8	(X9,Y9) B9	(X10,Y10) B10	(X11,Y11) B11
(X12,Y12) B12	(X13,Y13) B13	(X14,Y14) B14	(X15,Y15) B15

700

[Fig. 8]

(X0,Y0) ref_idx 2 B0	(X1,Y1) ref_idx 1 B1	(X2,Y2) ref_idx 0 B2	(X3,Y3) ref_idx 0 B3
(X4,Y4) ref_idx 2 B4	(X5,Y5) ref_idx 1 B5	(X6,Y6) ref_idx 0 B6	(X7,Y7) ref_idx 0 B7
(X8,Y8) ref_idx 2 B8	(X9,Y9) ref_idx 0 B9	(X10,Y10) ref_idx 1 B10	(X11,Y11) ref_idx 1 B11
(X12,Y12) ref_idx 2 B12	(X13,Y13) ref_idx 0 B13	(X14,Y14) ref_idx 1 B14	(X15,Y15) ref_idx 1 B15

(X0,Y0) ref_idx 2 B0	(X0,Y0) ref_idx 1 B1	(X0,Y0) ref_idx 0 B2	(X0,Y0) ref_idx 0 B3
(X0,Y0) ref_idx 2 B4	(X0,Y0) ref_idx 1 B5	(X0,Y0) ref_idx 0 B6	(X0,Y0) ref_idx 0 B7
(X0,Y0) ref_idx 2 B8	(X0,Y0) ref_idx 0 B9	(X0,Y0) ref_idx 1 B10	(X0,Y0) ref_idx 1 B11
(X0,Y0) ref_idx 2 B12	(X0,Y0) ref_idx 0 B13	(X0,Y0) ref_idx 1 B14	(X0,Y0) ref_idx 1 B15



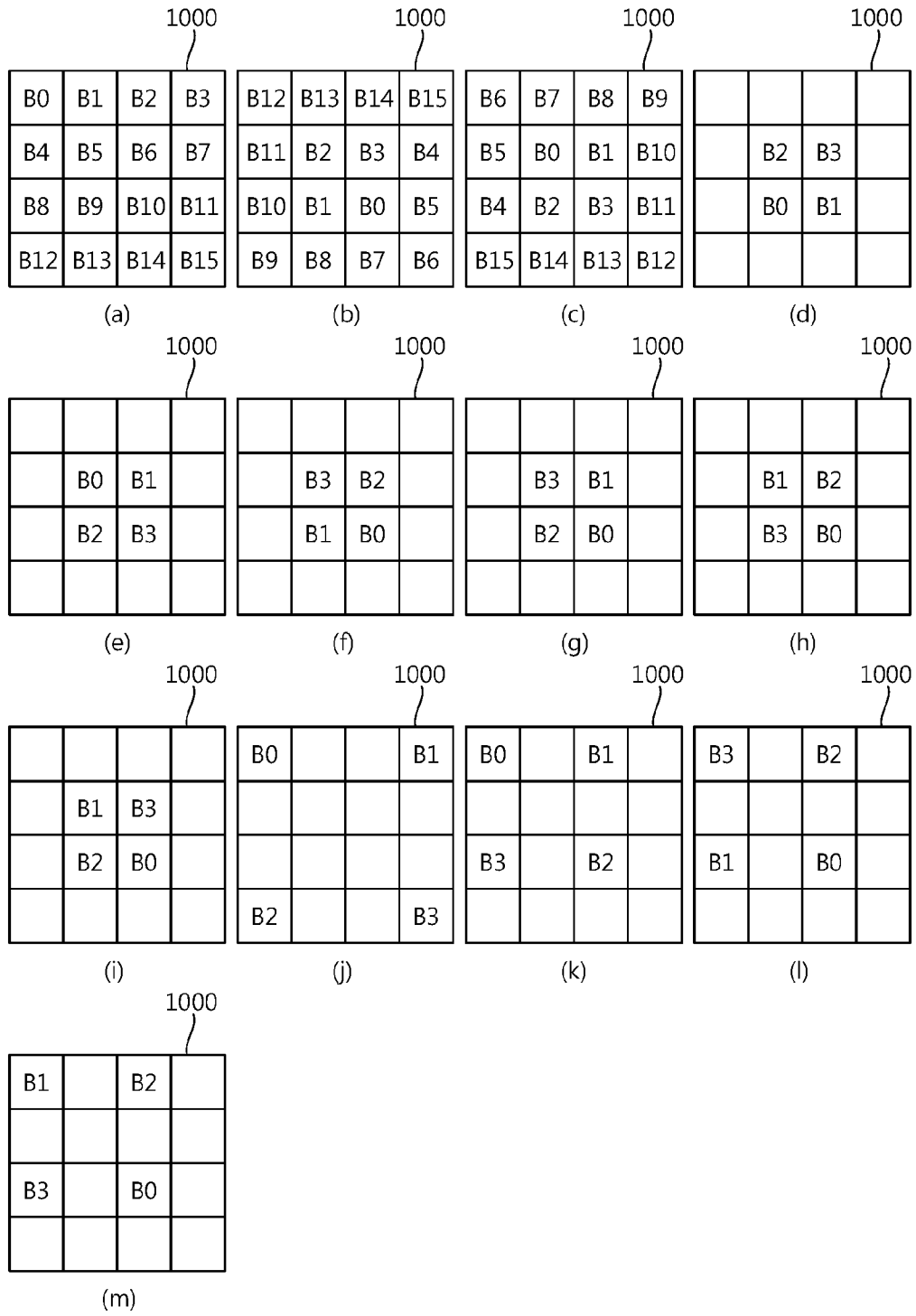
움직임 벡터 메모리 압축

[Fig. 9]

(X0,Y0) B0	(X1,Y1) B1	(X2,Y2) B2	(X3,Y3) B3
(X4,Y4) B4	(X5,Y5) B5	(X6,Y6) B6	(X7,Y7) B7
(X8,Y8) B8	(X9,Y9) B9	(X10,Y10) B10	(X11,Y11) B11
(X12,Y12) B12	(X13,Y13) B13	(X14,Y14) B14	(X15,Y15) B15

900

[Fig. 10]

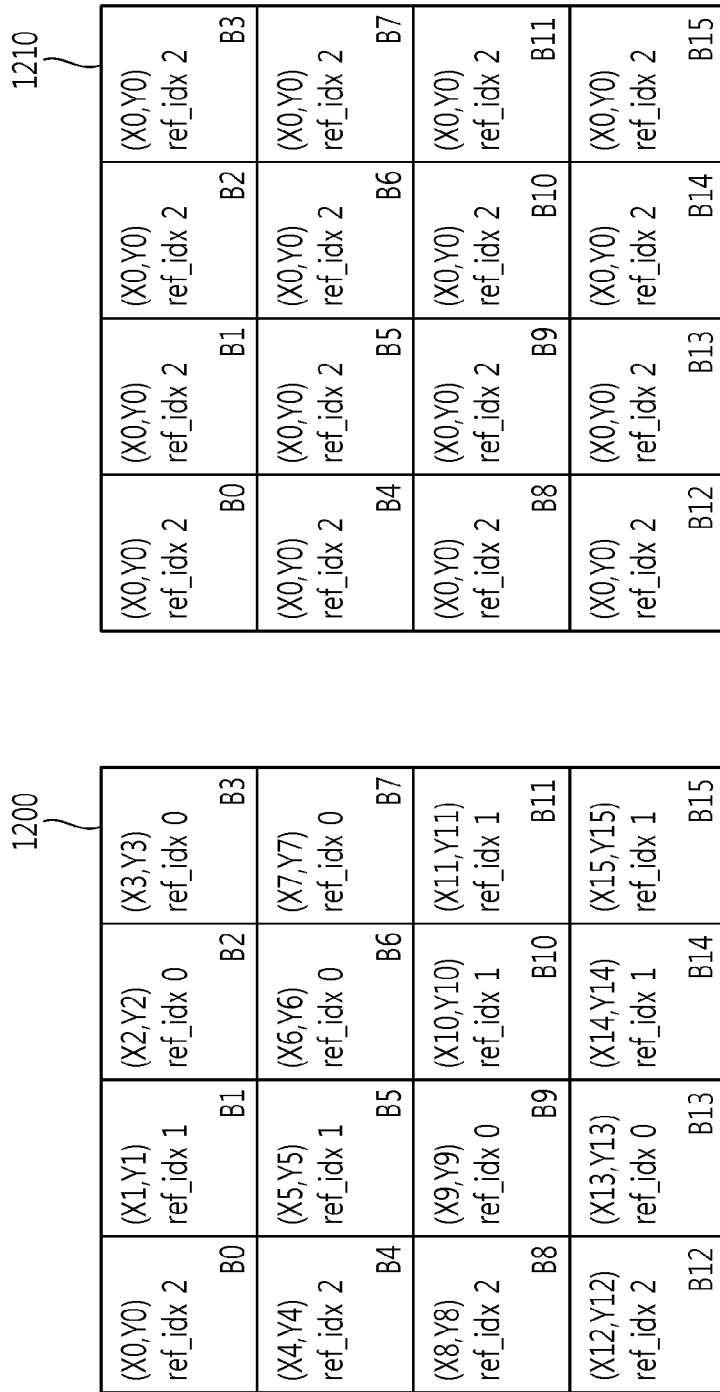


[Fig. 11]

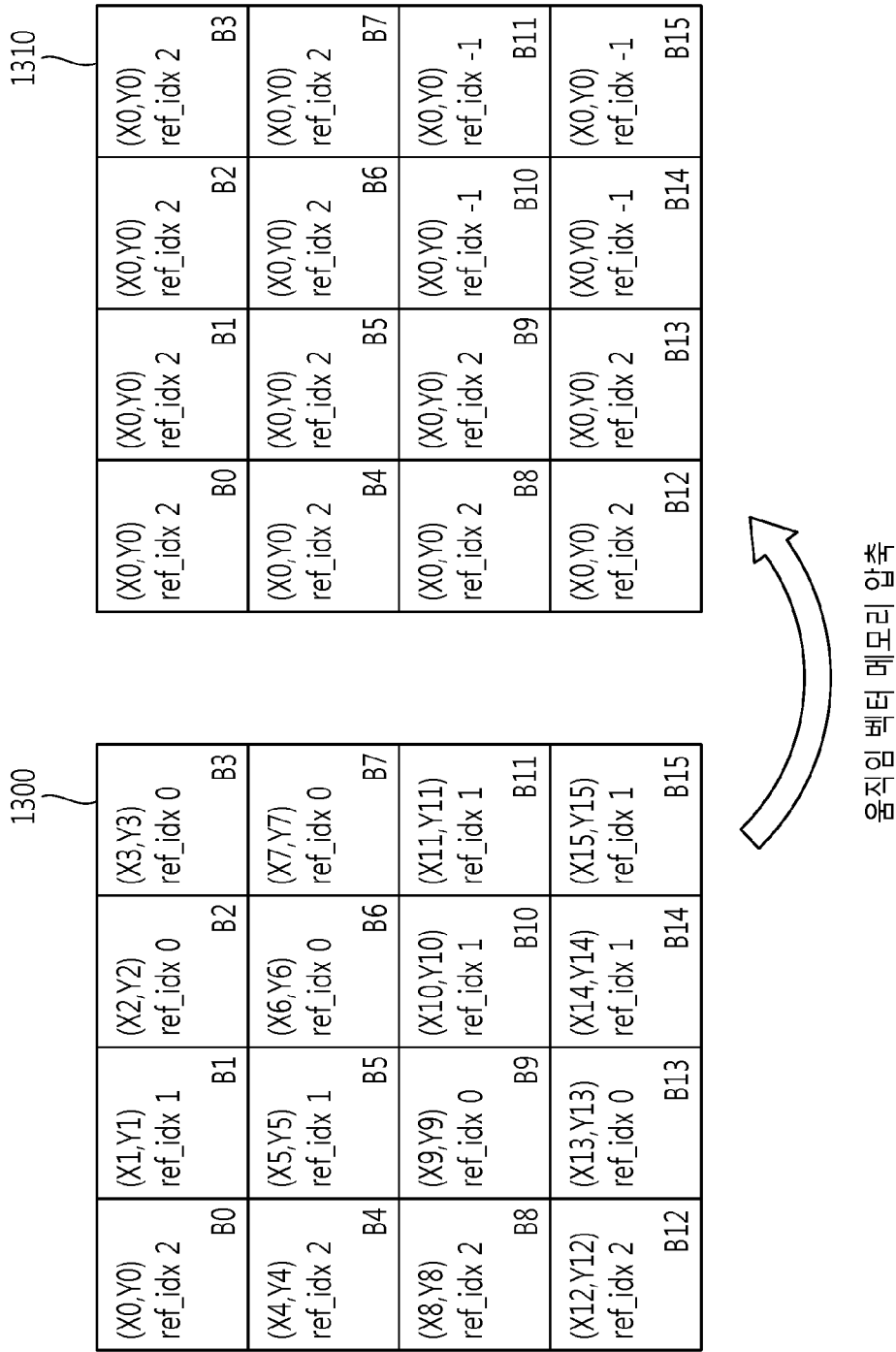
B16	B17	B18	B19	B20	B21
B35	B0	B1	B2	B3	B22
B34	B4	B5	B6	B7	B23
B33	B8	B9	B10	B11	B24
B32	B12	B13	B14	B15	B25
B31	B30	B29	B28	B27	B26

110

[Fig. 12]



[Fig. 13]



[Fig. 14]

1400	(X0,Y0) ref_idx 2 B0	(X1,Y1) ref_idx 1 B1	(X2,Y2) ref_idx 0 B2	(X3,Y3) ref_idx 0 B3
	(X4,Y4) ref_idx 2 B4	(X5,Y5) ref_idx 1 B5	(X6,Y6) ref_idx 0 B6	(X7,Y7) ref_idx 0 B7
	(X8,Y8) ref_idx 2 B8	(X9,Y9) ref_idx 0 B9	(X10,Y10) ref_idx -1 B10	(X11,Y11) ref_idx -1 B11
	(X12,Y12) ref_idx 2 B12	(X13,Y13) ref_idx 0 B13	(X14,Y14) ref_idx -1 B14	(X15,Y15) ref_idx -1 B15

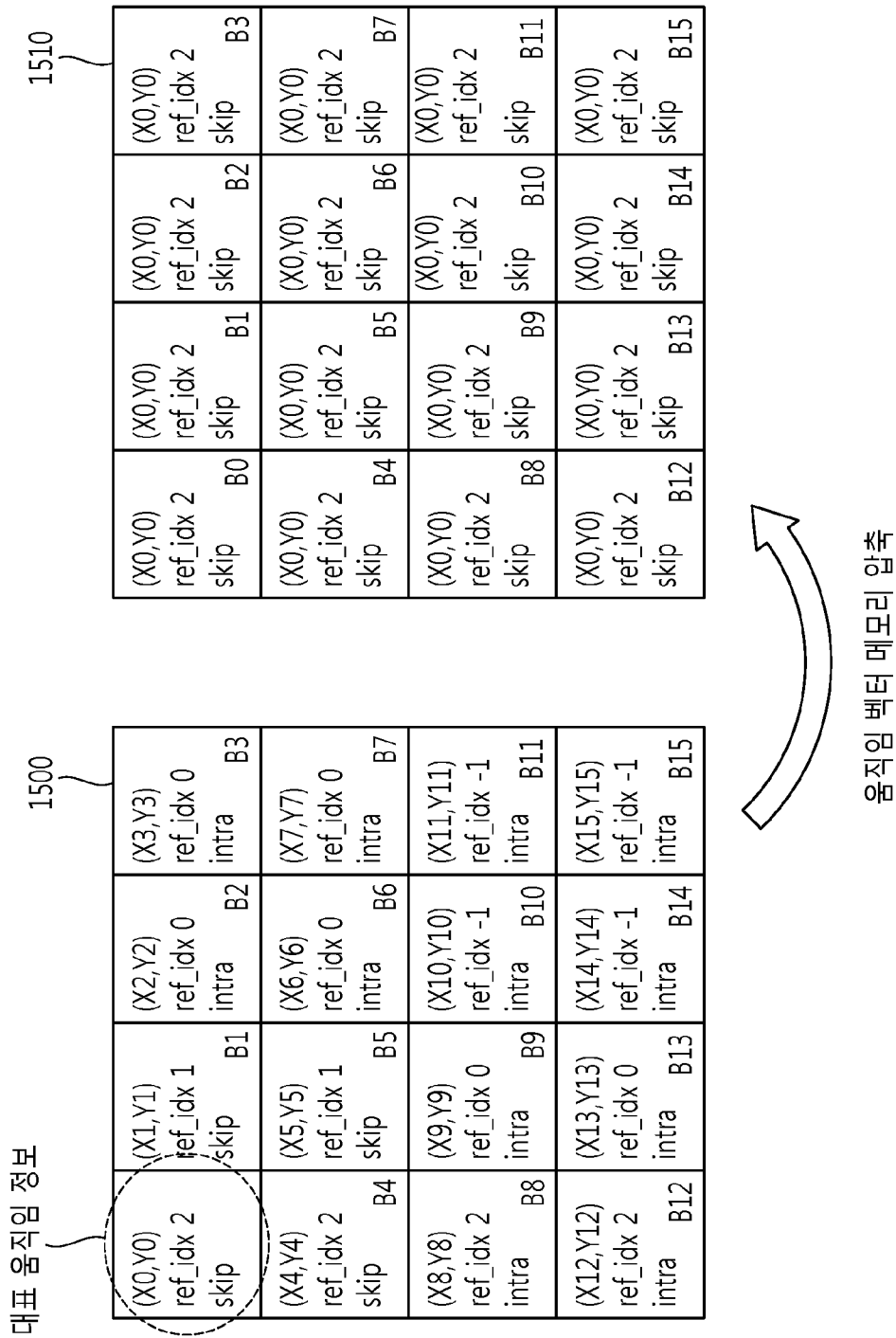
  

1410	(X0,Y0) ref_idx 2 B0	(X0,Y0) ref_idx 2 B1	(X0,Y0) ref_idx 2 B2	(X0,Y0) ref_idx 2 B3
	(X0,Y0) ref_idx 2 B4	(X0,Y0) ref_idx 2 B5	(X0,Y0) ref_idx 2 B6	(X0,Y0) ref_idx 2 B7
	(X0,Y0) ref_idx 2 B8	(X0,Y0) ref_idx 2 B9	(X0,Y0) ref_idx 2 인트라로 추정 B10	(X0,Y0) ref_idx 2 인트라로 추정 B11
	(X0,Y0) ref_idx 2 B12	(X0,Y0) ref_idx 2 B13	(X0,Y0) ref_idx 2 인트라로 추정 B14	(X0,Y0) ref_idx 2 인트라로 추정 B15

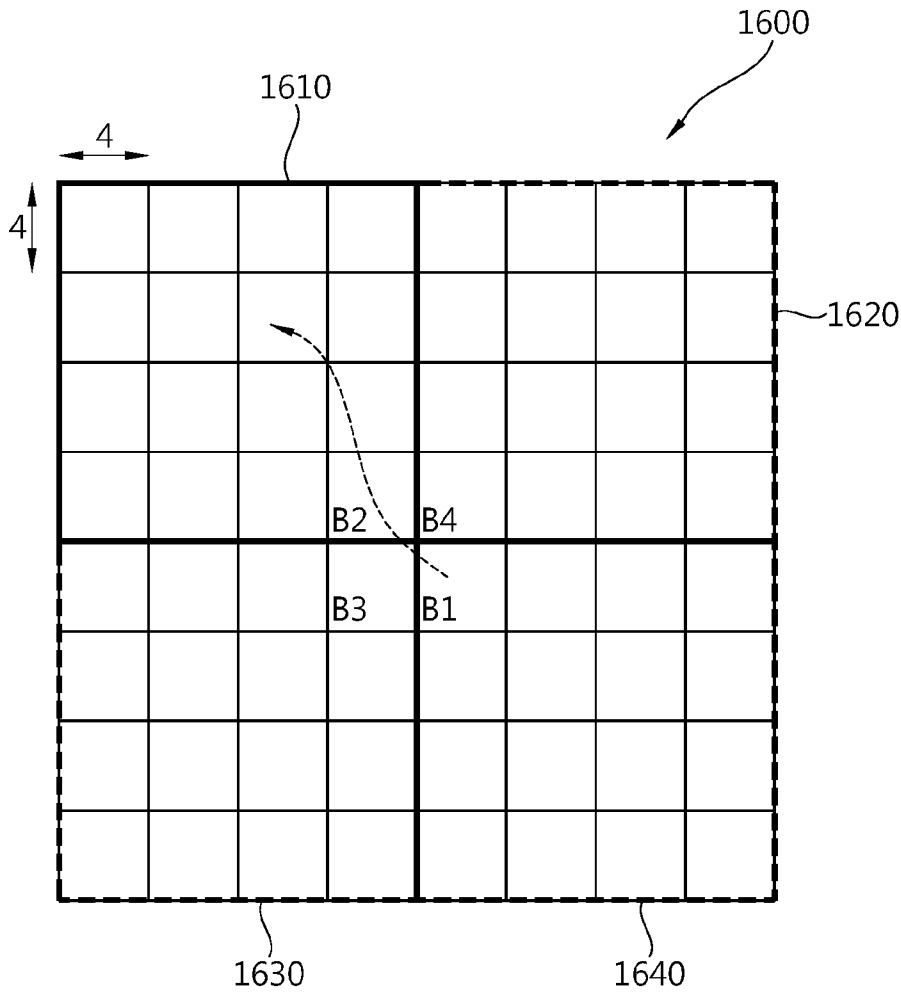


움직임 벡터 메모리 압축

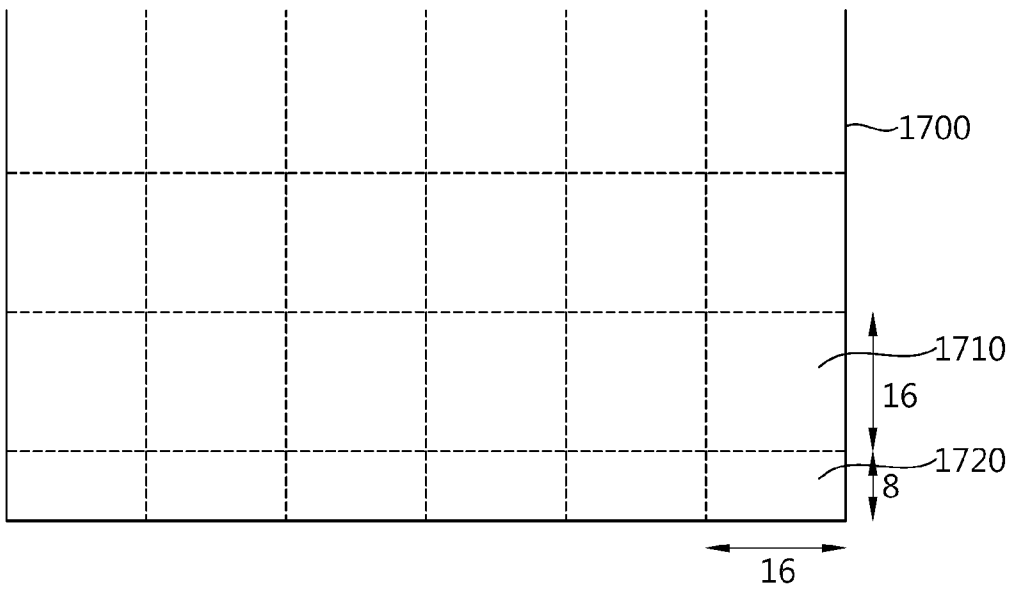
[Fig. 15]



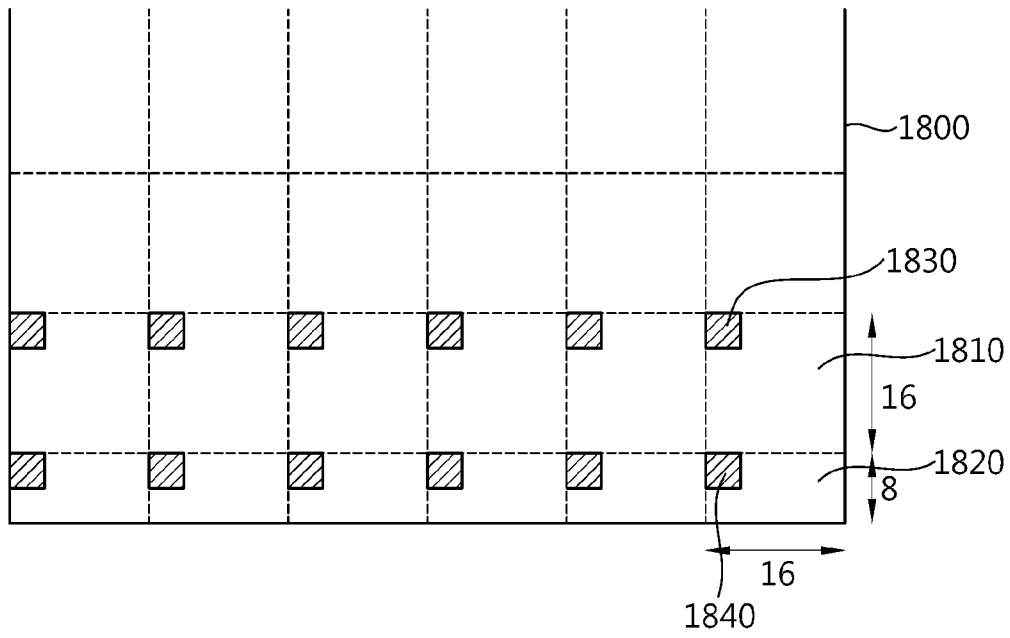
[Fig. 16]



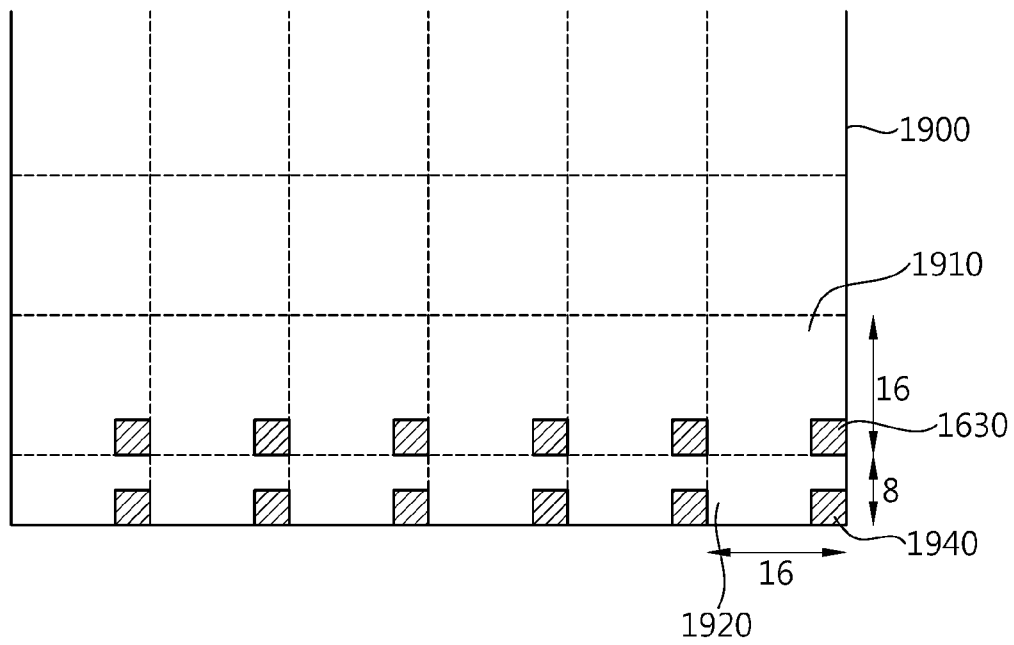
[Fig. 17]



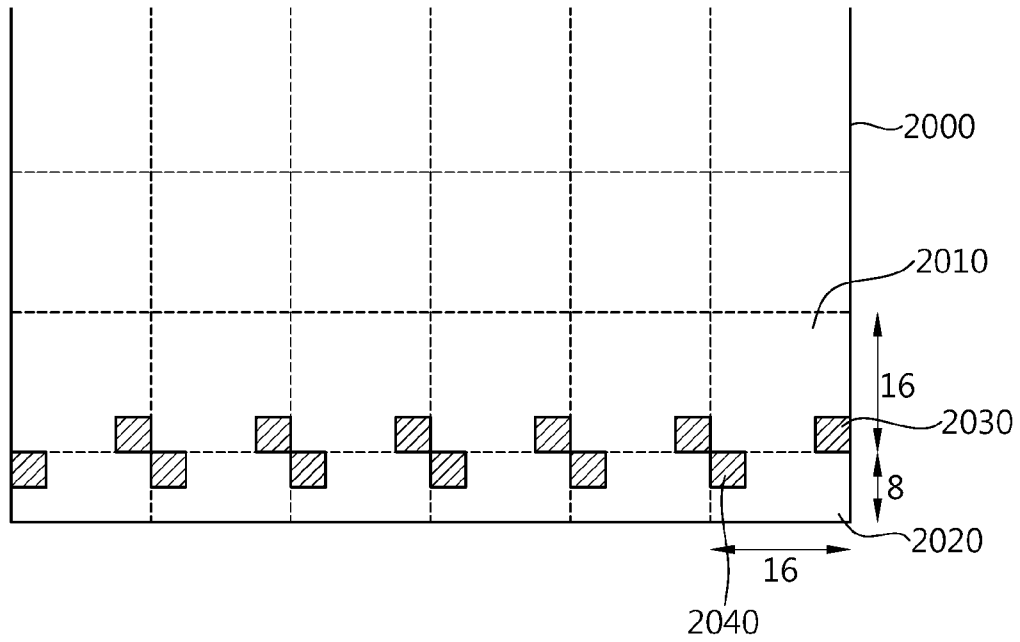
[Fig. 18]



[Fig. 19]

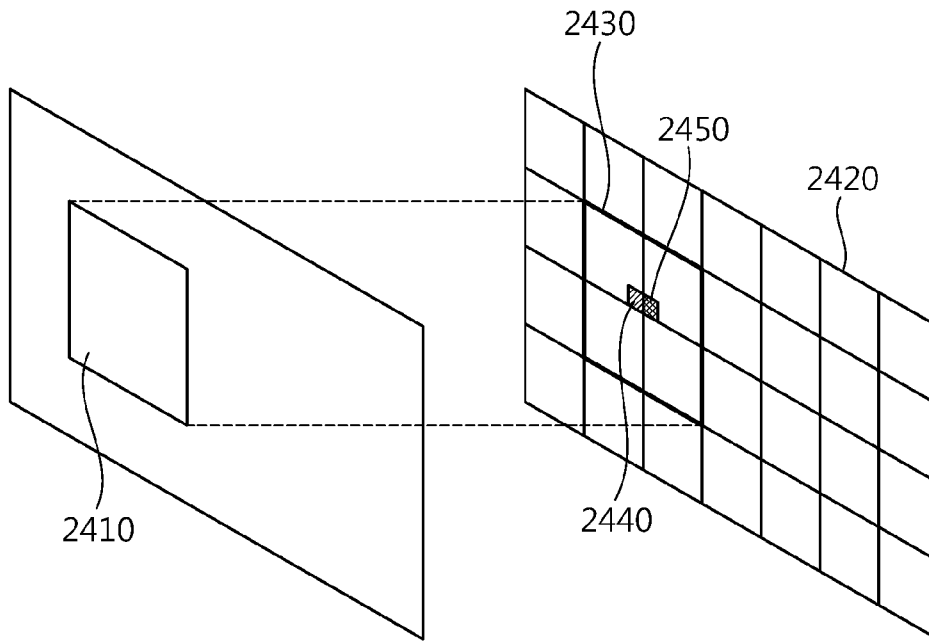


[Fig. 20]

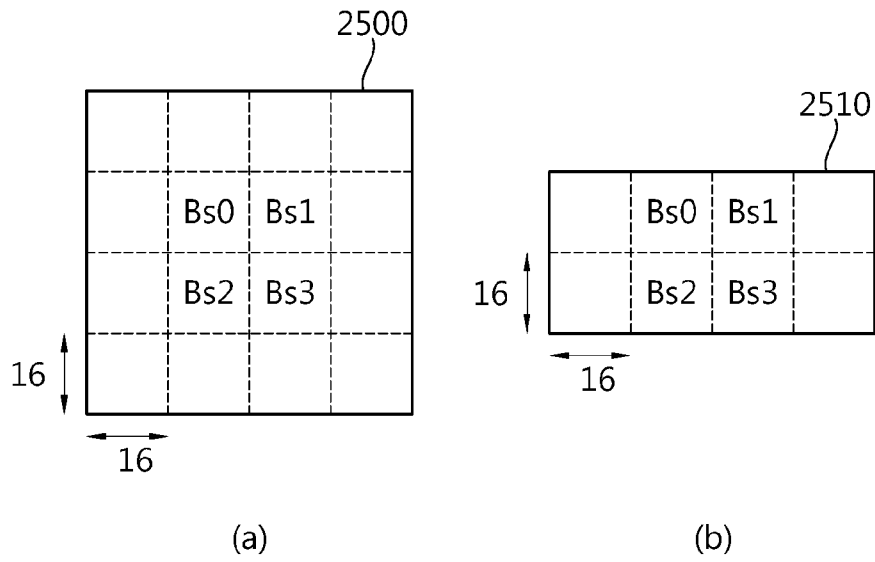




[Fig. 24]



[Fig. 25]



[Fig. 26]

