

(12) 특허협력조약에 의하여 공개된 국제출원

(19) 세계지식재산권기구
국제사무국

(43) 국제공개일
2013년 4월 18일 (18.04.2013)



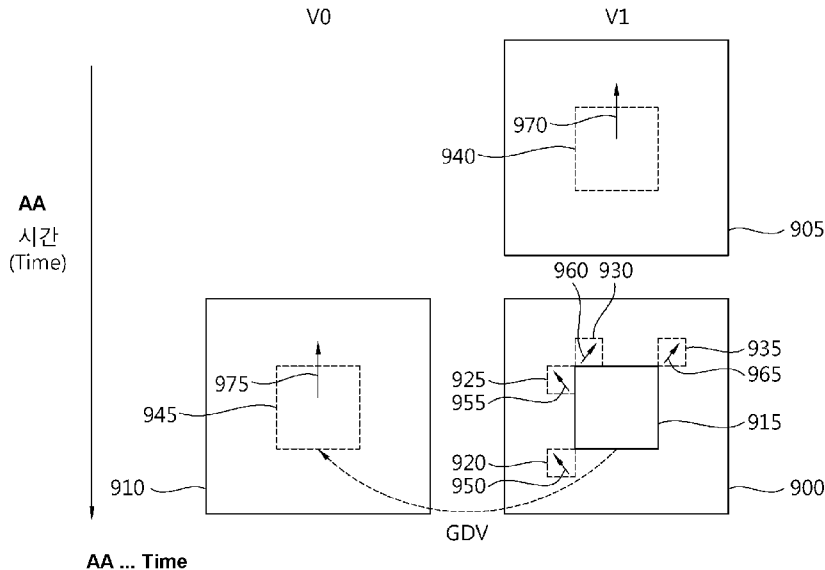
(10) 국제공개번호
WO 2013/055148 A2

- (51) 국제특허분류: H04N 13/00 (2006.01) H04N 7/26 (2006.01)
- (21) 국제출원번호: PCT/KR2012/008311
- (22) 국제출원일: 2012년 10월 12일 (12.10.2012)
- (25) 출원언어: 한국어
- (26) 공개언어: 한국어
- (30) 우선권정보: 61/546,066 2011년 10월 12일 (12.10.2011) US
- (71) 출원인: 엘지전자 주식회사 (LG ELECTRONICS INC.) [KR/KR]; 150-721 서울 영등포구 여의도동 20, Seoul (KR).
- (72) 발명자: 성재원 (SUNG, Jaewon); 137-130 서울시 서초구 양재동 221 엘지전자, Convergence R&D 연구소, Seoul (KR). 예세훈 (YEA, Sehoon); 137-130 서울시 서초구 양재동 221 엘지전자, Convergence R&D 연구소, Seoul (KR). 손은용 (SON, Eunyong); 137-130 서울시 서초구 양재동 221 엘지전자, Convergence R&D 연구소, Seoul (KR). 정지욱 (JUNG, Jiwook); 137-130 서울시 서초구 양재동 221 엘지전자, Convergence R&D 연구소, Seoul (KR).
- (74) 대리인: 에스앤아이피 특허법인 (S&IP PATENT & LAW FIRM); 135-080 서울 강남구 테헤란로 14길 5(역삼동 삼흥역삼빌딩 2층), 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, BN, 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, PA, 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.
- (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, RU, TJ, TM), 유럽 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, 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).

[다음 쪽 계속]

(54) Title: IMAGE ENCODING METHOD AND IMAGE DECODING METHOD

(54) 발명의 명칭 : 영상 인코딩 방법 및 디코딩 방법



(57) Abstract: The present invention relates to an image encoding method and an image decoding method. An image encoding method according to the present invention comprises: a step of determining motion information of a current block; and a step of transmitting information for inducing the motion information, wherein the step of determining motion information of the current block determines the motion information of the current block by reusing motion information of a reference block.

(57) 요약서: 본 발명은 영상 인코딩 방법 및 디코딩 방법에 관한 것으로서, 본 발명에 따른 영상 인코딩 방법은 현재 블록의 움직임 정보를 결정하는 단계 및 상기 움직임 정보를 유도하기 위한 정보를 전송하는 단계를 포함하며, 상기 현재 블록의 움직임 정보를 결정하는 단계에서는 상기 현재 블록의 움직임 정보를 참조 블록의 움직임 정보를 재활용하여 결정할 수 있다.

WO 2013/055148 A2

공개:

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

명세서

발명의 명칭: 영상 인코딩 방법 및 디코딩 방법

기술분야

[0001] 본 발명은 영상 정보 처리 기술에 관한 것으로서, 더 구체적으로는 3D 영상의 인코딩 방법 및 디코딩 방법과 이를 이용하는 장치에 관한 것이다.

배경기술

[0002] 최근 고해상도, 고품질의 영상에 대한 요구가 다양한 응용 분야에서 증가하고 있다. 하지만, 영상이 고해상도를 가지고 고품질이 될수록 해당 영상에 관한 정보량도 함께 증가한다.

[0003] 따라서 기존의 유무선 광대역 회선과 같은 매체를 이용하여 영상 정보를 전송하거나 기존의 저장 매체를 이용해 영상 정보를 저장하는 경우에는, 정보의 전송 비용과 저장 비용이 증가하게 된다. 고해상도, 고품질 영상의 정보를 효과적으로 전송하거나 저장하고, 재생하기 위해 고효율의 영상 압축 기술을 이용할 수 있다.

[0004] 한편, 고해상/대용량의 영상을 처리할 수 있게 됨에 따라서, 3D 비디오를 이용한 디지털 방송 서비스가 차세대 방송 서비스의 하나로 주목 받고 있다. 3D 비디오는 복수의 시점(view) 채널을 이용하여 현장감과 몰입감을 제공할 수 있다.

[0005] 3D 비디오는 FVV(free viewpoint video), FTV(free viewpoint TV), 3DTV, surveillance 및 홈 엔터테인먼트와 같은 다양한 영역에 사용될 수 있다.

[0006] 싱글 뷰 비디오(single view video)와 달리 멀티 뷰를 이용한 3D 비디오는 동일한 POC(picture order count)의 뷰들 사이에 높은 상관도(correlation)를 가진다. 멀티 뷰 영상은 인접한 여러 카메라 즉, 여러 시점(view)를 이용하여 동일한 장면을 동시에 촬영하기 때문에, 시차와 약간의 조명 차이를 제외하면 거의 같은 정보를 담고 있으므로 서로 다른 뷰 간의 상관도가 높다.

[0007] 따라서, 멀티 뷰 비디오의 인코딩/디코딩에서는 서로 다른 뷰 사이의 상관도를 고려할 수 있다. 예를 들어, 현재 뷰의 디코딩 대상 블록을 다른 뷰의 블록을 참조하여 예측하거나 디코딩할 수 있다. 이 경우에, 서로 다른 뷰 간의 관계를 산출하여 예측에 이용할 수 있다.

발명의 요약

기술적 과제

[0008] 본 발명은 3D 영상의 인코딩 및 디코딩에 있어서, 움직임 정보(예컨대, 움직임 벡터)를 재사용하는 방법 및 장치를 제공하는 것을 목적으로 한다.

[0009] 본 발명은 3D 영상의 인코딩 및 디코딩에 있어서, 다른 뷰 영상의 움직임 정보를 재활용하여 현재 뷰 영상의 움직임 정보를 결정 또는 유도하는 방법 및 장치를 제공하는 것을 목적으로 한다.

- [0010] 본 발명은 3D 영상의 인코딩 및 디코딩에 있어서, 텍스처 영상의 움직임 정보를 재활용하여 깊이 영상의 움직임 정보를 결정 또는 유도하는 방법 및 장치를 제공하는 것을 목적으로 한다.
- [0011] 본 발명은 다른 뷰 영상의 움직임 벡터를 재활용하여 얻은 움직임 정보를 기반으로 현재 뷰 영상을 인코딩/디코딩 하는 방법 및 장치를 제공하는 것을 목적으로 한다.
- [0012] 본 발명은 텍스처 영상의 움직임 정보를 재활용하여 얻은 움직임 정보를 기반으로 깊이 영상을 인코딩/디코딩 하는 방법 및 장치를 제공하는 것을 목적으로 한다.

과제 해결 수단

- [0013] (1)본 발명의 일 실시형태는 영상 인코딩 방법으로서, 현재 블록의 움직임 정보를 결정하는 단계 및 상기 움직임 정보를 유도하기 위한 정보를 전송하는 단계를 포함하며, 상기 현재 블록의 움직임 정보를 결정하는 단계에서는 상기 현재 블록의 움직임 정보를 참조 블록의 움직임 정보를 재활용하여 결정할 수 있다.
- [0014] (2) (1)에 있어서, 상기 현재 블록은 텍스처 픽처의 블록이고, 상기 참조 블록은 참조 뷰 내 블록일 수 있다.
- [0015] (3) (2)에 있어서, 상기 움직임 정보를 유도하기 위한 정보는, 상기 참조 블록의 움직임 정보를 재활용할 것을 지시하는 정보, 상기 참조 뷰의 방향을 지시하는 정보, 상기 참조 블록에 대한 GDV(Global Disparity Vector)를 지시하는 정보 중 적어도 하나를 포함할 수 있다.
- [0016] (4) (2)에 있어서, 상기 참조 블록에 대한 GDV는 상기 현재 블록이 속하는 픽처 내에서 영역별로 결정될 수 있으며, 상기 GDV가 결정되는 영역은 쿼드 트리 구조를 가질 수 있다.
- [0017] (5) (4)에 있어서, 상기 움직임 정보를 유도하기 위한 정보는, 상기 쿼드 트리 구조의 최대 깊이 정보 및 최대 크기 정보를 포함할 수 있으며, 상기 GDV가 결정되는 영역이 최대 깊이의 영역인 경우에는 GDV 값을 지정하는 정보를 포함할 수 있다.
- [0018] (6) (4)에 있어서, 상기 움직임 정보를 유도하기 위한 정보는, 상기 쿼드 트리 구조에 대한 분할 여부를 지시하는 정보를 포함할 수 있으며, 상기 GDV가 결정되는 영역이 분할되지 않는 경우에는 GDV 값을 지정하는 정보를 포함할 수 있다.
- [0019] (7) (1)에 있어서, 상기 현재 블록은 깊이 픽처의 블록이고, 상기 참조 블록은 동일 뷰의 텍스처 픽처 내 블록일 수 있다.
- [0020] (8) (7)에 있어서, 상기 현재 블록이 상기 참조 블록보다 큰 경우에, 상기 현재 블록의 움직임 정보는 상기 참조 블록의 움직임 정보 및 상기 참조 블록의 주변 블록들의 움직임 정보를 기반으로 결정될 수 있다.

- [0021] (9) (7)에 있어서, 상기 현재 블록이 상기 참조 블록보다 작은 경우에, 상기 현재 블록의 움직임 정보는 상기 참조 블록의 서브 블록들의 움직임 정보를 기반으로 결정될 수 있다.
- [0022] (10) (7)에 있어서, 상기 움직임 정보를 유도하기 위한 정보는, 상기 참조 블록의 움직임 정보를 기반을 복원된 현재 블록의 깊이 값을 보상하는 오프셋 정보를 포함할 수 있다.
- [0023] (11) 본 발명의 다른 실시형태는 영상 디코딩 방법으로서, 현재 블록의 움직임 정보를 유도하기 위한 정보를 수신하는 단계 및 상기 움직임 정보를 유도하기 위한 정보를 기반으로 상기 현재 블록의 움직임 정보를 유도하는 단계를 포함하며, 상기 현재 블록의 움직임 정보를 유도하는 단계에서는 상기 현재 블록의 움직임 정보를 참조 블록의 움직임 정보를 재활용하여 결정할 수 있다.
- [0024] (12) (11)에 있어서, 상기 현재 블록은 텍스처 픽처의 블록이고, 상기 움직임 정보를 유도하기 위한 정보가 다른 뷰 내 블록의 움직임 정보를 재활용하여 상기 현재 블록의 움직임 정보를 유도할 것을 지시하는 정보를 포함하는 경우에, 참조 뷰 내 블록을 상기 참조 블록으로 설정할 수 있다.
- [0025] (13) (12)에 있어서, 상기 움직임 정보를 유도하기 위한 정보는, 상기 참조 뷰의 방향을 지시하는 정보 및/또는 상기 참조 블록에 대한 GDV(Global Disparity Vector)를 지시하는 정보를 포함할 수 있다.
- [0026] (14) (12)에 있어서, 상기 움직임 정보를 유도하기 위한 정보는 상기 참조 블록에 대한 GDV를 포함할 수 있으며, 상기 GDV의 값은 상기 현재 블록이 속하는 픽처 내에서 영역별로 설정될 수 있다.
- [0027] (15) (14)에 있어서, 상기 GDV 값이 특정되는 영역은 쿼드 트리 구조를 가질 수 있으며, 상기 움직임 정보를 유도하기 위한 정보는, 상기 쿼드 트리 구조의 최대 깊이를 지시하는 정보; 및
- [0028] 상기 현재 블록이 최대 깊이의 GDV 설정 영역인 경우에 GDV 값을 지시하는 정보를 포함할 수 있다.
- [0029] (16) (14)에 있어서, 상기 GDV 값이 특정되는 영역은 쿼드 트리 구조를 가질 수 있으며, 상기 움직임 정보를 유도하기 위한 정보는, 상기 GDV 설정 영역의 분할 여부를 지시하는 정보 및 상기 현재 블록이 속하는 GDV 설정 영역이 분할되지 않는 경우에 GDV 값을 지시하는 정보를 포함할 수 있다.
- [0030] (17) (11)에 있어서, 상기 현재 블록은 깊이 픽처의 블록이고, 상기 움직임 정보를 유도하기 위한 정보가 텍스처 픽처 내 블록의 움직임 정보를 재활용하여 상기 현재 블록의 움직임 정보를 유도할 것을 지시하는 정보를 포함하는 경우에, 상기 깊이 픽처와 동일 뷰 내의 텍스처 픽처 내 블록을 상기 참조 블록으로 설정할 수 있다.
- [0031] (18) (17)에 있어서, 상기 현재 블록이 상기 참조 블록보다 큰 경우에, 상기 현재 블록의 움직임 정보는 상기 참조 블록의 움직임 정보 및 상기 참조 블록의 주변 블록들의 움직임 정보를 기반으로 유도될 수 있다.

[0032] (19) (17)에 있어서, 상기 현재 블록이 상기 참조 블록보다 작은 경우에, 상기 현재 블록의 움직임 정보는 상기 참조 블록의 서브 블록들의 움직임 정보를 기반으로 유도될 수 있다.

[0033] (20) (17)에 있어서, 상기 움직임 정보를 유도하기 위한 정보는, 상기 참조 블록의 움직임 정보를 기반을 복원된 현재 블록의 깊이 값을 보상하는 오프셋 정보를 포함할 수 있다.

발명의 효과

[0034] 본 발명에 의하면, 3D 영상의 인코딩 및 디코딩에 있어서, 움직임 정보(예컨대, 움직임 벡터)를 재사용함으로써, 정보의 전송량을 줄이고 정보 처리의 복잡도를 저감할 수 있다.

[0035] 본 발명에 의하면, 다른 뷰 영상의 움직임 정보를 재활용하여 현재 뷰 영상의 움직임 정보를 결정 또는 유도할 수 있다.

[0036] 본 발명에 의하면, 텍스처 영상의 움직임 정보를 재활용하여 깊이 영상의 움직임 정보를 결정 또는 유도할 수 있다.

[0037] 본 발명은 다른 뷰 영상의 움직임 벡터를 현재 뷰 영상에 재활용하거나, 텍스처 영상의 움직임 벡터를 깊이 영상에 재활용함으로써 코딩 효율을 향상 시키고 오버 헤드를 줄일 수 있다.

도면의 간단한 설명

[0038] 도 1은 3차원 영상을 처리하기 위한 영상 정보의 일 예를 개략적으로 설명하는 도면이다.

[0039] 도 2는 본 발명의 일 실시예에 따른 인코딩 장치를 개략적으로 도시한 블록도이다.

[0040] 도 3은 본 발명의 일 실시예에 따른 영상 디코딩 장치를 개략적으로 나타낸 블록도이다.

[0041] 도 4는 3D 비디오 인코딩/디코딩 과정에서 GDV를 이용하여 인터-뷰 예측이 수행되는 방법을 개략적으로 설명하는 도면이다.

[0042] 도 5는 멀티 뷰의 비디오를 디코딩 하는 경우의 일 예를 개략적으로 설명하는 도면이다.

[0043] 도 6은 인터-뷰 예측의 일 예로서, 스킵 모드가 적용되는 예를 개략적으로 설명하는 도면이다.

[0044] 도 7은 퍼지 모드를 이용한 인터 예측에서 현재 블록의 예측에 이용 가능한 주변 블록의 일 예를 개략적으로 설명하는 도면이다.

[0045] 도 8은 인터 예측에서 현재 블록의 예측에 이용 가능한 주변 블록의 다른 예를 개략적으로 설명하는 도면이다.

[0046] 도 9는 인터-뷰 예측에서 사용될 수 있는 움직임 벡터 예측자의 일 예를 개략적으로 설명하는 도면이다.

[0047] 도 10은 인터-뷰 예측에서 사용될 수 있는 움직임 벡터 예측자의 다른 예를

개략적으로 설명하는 도면이다.

- [0048] 도 11은 참조 뷰의 방향을 개략적으로 설명하는 도면이다.
- [0049] 도 12는 현재 픽처의 각 영역별로 GDV를 정의하여 인터-뷰 예측에 적용하는 방법의 일 예를 개략적으로 설명하는 도면이다.
- [0050] 도 13은 동일한 장면을 텍스처 영상과 깊이 영상으로 나타낸 일 예를 개략적으로 설명하는 도면이다.
- [0051] 도 14는 현재 뷰의 깊이 영상을 텍스처 영상의 움직임 정보를 이용하여 예측하는 방법을 개략적으로 설명하는 도면이다.
- [0052] 도 15는 대응하는 텍스처 블록의 움직임 벡터로부터 현재 깊이 블록의 움직임 벡터를 유도하는 방법을 개략적으로 설명하는 도면이다.
- [0053] 도 16은 본 발명에 따라서 움직임 정보를 재활용하여 3D 영상을 인코딩 하는 방법을 개략적으로 설명하는 순서도이다.
- [0054] 도 17은 본 발명에 따라서 인코딩 장치에서 텍스처 영상의 움직임 정보를 결정하고 이를 이용하여 예측을 수행하는 방법의 일 예를 개략적으로 설명하는 순서도이다.
- [0055] 도 18은 본 발명에 따라서 텍스처 영상의 움직임 정보를 이용하여 깊이 영상의 예측을 수행하는 방법의 일 예를 개략적으로 설명하는 순서도이다.
- [0056] 도 19는 본 발명에 따라서 움직임 정보를 재활용하여 3D 영상을 디코딩 하는 방법을 개략적으로 설명하는 순서도이다.
- [0057] 도 20은 본 발명에 따라서 디코딩 장치가 텍스처 영상의 움직임 정보를 유도하는 방법의 일 예를 개략적으로 설명하는 도면이다.
- [0058] 도 21은 본 발명에 따라서 디코딩 장치가 깊이 영상의 움직임 정보를 유도하는 방법의 일 예를 개략적으로 설명하는 도면이다.

발명의 실시를 위한 형태

- [0059] 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.
- [0060] 한편, 본 발명에서 설명되는 도면상의 각 구성들은 영상 인코딩 장치/디코딩 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 실시에도 본 발명의 본질에서 벗어나지 않는 한 본 발명의 권리범위에 포함된다.

- [0061] 3차원 입체 영상을 디스플레이 장치에 재현하기 위한 인코딩/디코딩에 있어서, 인코딩 장치에 입력되는 영상은 텍스처 영상(texture image)과 깊이 맵(depth map)일 수 있다. 깊이 맵은 시점(viewpoint)로부터 영상 내 물체의 표면(surface)까지의 거리를 나타내는 영상을 의미한다. 여기서, 시점은, 예컨대 해당 영상을 촬영하는 카메라일 수 있다. 깊이 맵(깊이 영상)은 깊이를 촬영하는 카메라를 통해 생성될 수 있다.
- [0062] 텍스처 영상은 3차원 영상을 구성하는 영상으로서 깊이 정보 이외의 정보(예컨대, 색상, 명암 등)를 포함하는 영상을 의미하며, 멀티 뷰(multi-view)의 영상들로 구성될 수도 있다.
- [0063] 3차원 영상을 처리하기 위해, 후술하는 인코딩/디코딩 과정에서는 깊이 맵(깊이 영상)과 텍스처 영상이 각각 처리될 수 있으며, 텍스처 영상의 경우도 각 뷰(view) 별로 처리될 수 있다. 이때, 깊이 맵의 처리를 위해 텍스처 영상이 참조될 수 있고, 텍스처 영상의 처리를 위해 깊이 맵이 참조될 수도 있다. 또한, 텍스처 영상의 경우 다른 뷰의 영상을 참조하여 처리될 수도 있다.
- [0064] 도 1은 3차원 영상을 처리하기 위한 영상 정보의 일 예를 개략적으로 설명하는 도면이다.
- [0065] 3차원 영상을 처리하기 위해서 상술한 바와 같이, 복수 뷰(view)의 영상이 처리된다. 도 1에서는 설명의 편의를 위해 두 뷰(V1, V2)를 이용하는 영상 처리를 예로서 설명한다.
- [0066] 도 1을 참조하면, 동일한 POC(Picture Order Count)에서 뷰 1(V1)의 텍스처 픽처 VT1과 깊이 맵(depth map) DV1 그리고 뷰 2(V2)의 텍스처 영상 VT2와 깊이 맵 DV2가 존재할 수 있다. POC는 영상 출력 순서를 나타내며, POC가 동일한 영상은 출력되는 순서가 동일하다고 할 수 있다.
- [0067] V1의 깊이 맵 DV1을 기반으로 카메라 파라미터와 깊이 정보 등 추가적인 정보를 이용한 와핑(warping)을 통해 뷰 1과 다른 뷰에서의 깊이 맵 VD1'을 생성할 수도 있다.
- [0068] 도 2는 본 발명의 일 실시예에 따른 인코딩 장치를 개략적으로 도시한 블록도이다. 도 2를 참조하면, 인코딩 장치(200)는 픽처 분할부(205), 예측부(210), 변환부(215), 양자화부(220), 재정렬부(225), 엔트로피 인코딩부(230), 역양자화부(235), 역변환부(240), 필터부(245) 및 메모리(250)를 구비한다.
- [0069] 픽처 분할부(205)는 입력된 픽처를 적어도 하나의 처리 단위 블록으로 분할할 수 있다. 이때, 처리 단위로서의 블록은 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.
- [0070] 예측부(210)는 픽처 분할부(205)에서 픽처의 처리 단위에 대하여 예측을 수행하여 예측 블록을 생성한다. 예측부(210)에서 픽처의 처리 단위는 CU일 수도 있고, TU일 수도 있고, PU일 수도 있다. 또한, 예측부(210)는 해당 처리 단위에 대하여 적용되는 예측 방법을 결정하고, 각 예측 방법의 구체적인 내용(예컨대, 예측 모드 등)을 정할 수 있다.
- [0071] 예측부(210)는 예측 방법으로서 인트라 예측(intra prediction), 인터 예측(inter prediction), 인터 뷰(inter-view) 예측 중 어느 하나를 적용할 수 있다.
- [0072] 예측부(210)는 인터 예측을 통해서 현재 픽처의 이전 픽처 및/또는 이후 픽처 중 적어도 한 픽처의 정보를 기초로 예측을 수행하여 예측 블록을 생성할 수 있다. 예측부(210)는 인트라 예측을 통해서 현재 픽처 내의 픽셀 정보를 기초로 예측을 수행하여 예측 블록을 생성할 수 있다. 예측부(210)는 인터-뷰 예측을 통해서 서로 다른 뷰의 픽처를 참조하여 예측 블록을 생성할 수 있다.
- [0073] 인터 예측의 방법으로서, 스킵(skip) 모드, 머지(merge) 모드, MVP(Motion Vector Prediction) 등을 이용할 수 있다. 인터 예측에서는 PU에 대하여, 참조 픽처를 선택하고 PU와 동일한 크기의 참조 블록을 선택할 수 있다. 예측부(210)는 현재 블록에 대한 레지듀얼(residual) 신호와 움직임 벡터를 최소화하는 예측 블록을 생성한다.
- [0074] 인터 예측을 통해 선택된 참조 픽처의 인덱스, 움직임 벡터(ex. Motion Vector Predictor), 레지듀얼 신호 등의 정보는 엔트로피 인코딩되어 디코딩 장치에 전달된다. 스킵 모드가 적용되는 경우에는 레지듀얼을 예측 블록을 복원 블록으로 할 수 있으므로, 레지듀얼을 생성, 변환, 양자화, 전송하지 않을 수 있다.
- [0075] 인트라 예측을 수행하는 경우에는, PU 단위로 예측 모드가 정해져서 PU 단위로 예측이 수행될 수 있다. 또한, PU 단위로 예측 모드가 정해지고 TU 단위로 인트라 예측이 수행될 수도 있다.
- [0076] 인트라 예측에서 예측 모드는 33개의 방향성 예측 모드와 적어도 2개 이상의 비방향성 모드를 가질 수 있다. 비방향성 모드는 DC 예측 모드 및 플래이너 모드(Planar 모드)을 포함할 수 있다.
- [0077] 인트라 예측에서는 참조 샘플에 필터를 적용한 후 예측 블록을 생성할 수도 있다. 이때, 참조 샘플에 필터를 적용할 것인지는 현재 블록의 인트라 예측 모드 및/또는 사이즈에 따라 결정될 수 있다.
- [0078] 인터-뷰 예측에서 예측부(210)는 참조 뷰에서 현재 뷰 내 현재 블록의 예측에 참조 가능한 대응 블록의 위치를 특징하는 글로벌 디스패리티 벡터(Global Disparity Vector: GDV)를 이용하여 현재 블록에 대한 예측을 수행하고, 예측 블록을 생성할 수 있다. 예측부(210)는 대응 블록의 움직임 정보를 기반으로, 인터 예측과 동일하게, 스킵 모드를 적용하거나, 스킵 모드를 적용하거나,

MVP를 이용하여 현재 블록에 대한 인터-뷰 예측을 수행할 수 있다.

- [0079] PU는 다양한 사이즈/형태의 블록일 수 있다. 예컨대 PU는 $2N \times 2N$ 블록, $2N \times N$ 블록, $N \times 2N$ 블록, 또는 $N \times N$ 블록 (N 은 정수) 동일 수 있다. 또한, 상술한 크기의 PU 외에, $N \times mN$ 블록, $mN \times N$ 블록, $2N \times mN$ 블록 또는 $mN \times 2N$ 블록 ($m < 1$) 등의 PU를 더 정의하여 사용할 수도 있다.
- [0080] 생성된 예측 블록과 원본 블록 사이의 레지듀얼 값(레지듀얼 블록 또는 레지듀얼 신호)은 변환부(215)로 입력된다. 또한, 예측을 위해 사용한 예측 모드 정보, 움직임 벡터 정보, 디스패리티 벡터 등은 레지듀얼 값과 함께 엔트로피 인코딩부(230)에서 인코딩되어 디코딩 장치에 전달된다.
- [0081] 변환부(215)는 변환 단위별로 레지듀얼 블록에 대한 변환을 수행하고 변환 계수를 생성한다.
- [0082] 변환부(215)는 텍스처 영상과 깊이 맵에 대한 다운 샘플링을 수행한 후에 변환을 수행할 수도 있다. 다운 샘플링은 텍스처 영상과 깊이 맵에서, 저주파 영역에 대해 수행될 수도 있고, 디테일(detail)한 특성이 중요시되지 않는 영역에 대해서 수행될 수도 있다. 다운 샘플링을 통해 복잡도를 낮추고 코딩 효율을 높일 수 있다.
- [0083] 변환부(215)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(215)는 레지듀얼 블록을 DCT(Discrete Cosine Transform) 및/또는 DST(Discrete Sine Transform)를 이용하여 변환할 수 있다.
- [0084] 양자화부(220)는 변환부(215)에서 변환된 레지듀얼 값들을 양자화하여 양자화 계수를 생성할 수 있다. 양자화부(220)에서 산출된 값은 역양자화부(235)와 재정렬부(225)에 제공된다.
- [0085] 재정렬부(225)는 양자화부(220)로부터 제공된 양자화 계수를 재정렬한다. 양자화 계수를 재정렬함으로써 엔트로피 인코딩부(230)에서의 인코딩 효율을 높일 수 있다. 재정렬부(225)는 계수 스캐닝(Coefficient Scanning) 방법을 통해 2차원 블록 형태의 양자화 계수들을 1차원의 벡터 형태로 재정렬할 수 있다. 재정렬부(225)에서는 양자화부에서 전송된 계수들의 확률적인 통계를 기반으로 계수 스캐닝의 순서를 변경함으로써 엔트로피 인코딩부(230)에서의 엔트로피 인코딩 효율을 높일 수도 있다.
- [0086] 엔트로피 인코딩부(230)는 재정렬부(225)에 의해 재정렬된 양자화 계수들에 대한 엔트로피 인코딩을 수행할 수 있다. 엔트로피 인코딩에는 예를 들어, 지수 골롬(Exponential Golomb), CAVLC(Context-Adaptive Variable Length Coding), CABAC(Context-Adaptive Binary Arithmetic Coding) 등과 같은 인코딩 방법을 사용할 수 있다. 엔트로피 인코딩부(230)는 재정렬부(225) 및 예측부(210)로부터 전달받은 양자화 계수 정보 및 블록 타입 정보, 예측 모드 정보, 분할 단위 정보, 전송 단위 정보, 움직임 벡터 정보, 참조 픽처 정보, 블록의 보간 정보, 필터링 정보 등 다양한 정보를 인코딩할 수 있다.

- [0087] 엔트로피 인코딩부(230)는 멀티 뷰에 대한 영상 정보와 깊이 맵에 대한 영상 정보를 다중화하여 비트스트림으로 전송할 수도 있다.
- [0088] 역양자화부(235)는 양자화부(220)에서 양자화된 값들을 역양자화하고, 역변환부(240)는 역양자화부(235)에서 역양자화된 값들을 역변환한다. 역변환부(240)는 변환부(215)에서 다운 샘플링이 수행된 경우에, 역변환된 레지듀얼 블록에 업 샘플링을 수행할 수 있다. 업 샘플링의 샘플링 비율은 변환부(215)에서 수행된 다운 샘플링의 샘플링 비율에 대응하여 결정될 수 있다.
- [0089] 역양자화부(235) 및 역변환부(240)에서 생성된 레지듀얼 값과 예측부(210)에서 예측된 예측 블록이 합쳐져 복원 블록(Reconstructed Block)이 생성될 수 있다.
- [0090] 도 2에서는 가산기를 통해서, 레지듀얼 블록과 예측 블록이 합쳐져 복원 블록이 생성되는 것으로 설명하고 있다. 이때, 가산기를 복원 블록을 생성하는 별도의 유닛(복원 블록 생성부)로 볼 수도 있다.
- [0091] 필터부(245)는 필요에 따라 더블록킹 필터, ALF(Adaptive Loop Filter), SAO(Sample Adaptive Offset) 중 적어도 하나를 복원된 픽처에 적용할 수 있다.
- [0092] 더블록킹 필터는 복원된 픽처에서 블록 간의 경계에 생긴 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 경우에 수행될 수도 있다. SAO는 더블록킹 필터가 적용된 레지듀얼 블록에 대하여, 픽셀 단위로 원본 영상과의 오프셋 차이를 복원하며, 밴드 오프셋(Band Offset), 에지 오프셋(Edge Offset) 등의 형태로 적용된다.
- [0093] 메모리(250)는 필터부(245)를 통해 산출된 복원 블록 또는 픽처를 저장할 수 있다. 메모리(250)에 저장된 복원 블록 또는 픽처는 인터 예측을 수행하는 예측부(210)에 제공될 수 있다.
- [0094] 도 3은 본 발명의 일 실시예에 따른 영상 디코딩 장치를 개략적으로 나타낸 블록도이다. 도 3을 참조하면, 영상 디코딩 장치(300)는 엔트로피 디코딩부(310), 재정렬부(315), 역양자화부(320), 역변환부(325), 예측부(330), 필터부(335) 메모리(340)를 포함할 수 있다.
- [0095] 영상 인코딩 장치에서 영상 비트스트림이 입력된 경우, 입력된 비트스트림은 영상 인코딩 장치에서 영상 정보가 처리된 절차에 따라서 디코딩될 수 있다.
- [0096] 예컨대, 영상 인코딩 장치에서 엔트로피 인코딩을 수행하기 위해 CAVLC 등의 가변 길이 부호화(Variable Length Coding: VLC, 이하 'VLC' 라 함)가 사용된 경우에, 엔트로피 디코딩부(310)도 인코딩 장치에서 사용한 VLC 테이블과 동일한 VLC 테이블로 구현하여 엔트로피 디코딩을 수행할 수 있다. 또한, 영상 인코딩 장치에서 엔트로피 인코딩을 수행하기 위해 CABAC을 이용한 경우에, 엔트로피 디코딩부(310)는 이에 대응하여 CABAC을 이용한 엔트로피 디코딩을 수행할 수 있다.
- [0097] 인코딩 장치로부터 수신한 비트스트림이 멀티 뷰에 대한 영상 정보와 깊이

- 맵에 대한 영상 정보가 다중화된 것이라면, 엔트로피 디코딩부(310)는 수신한 비트스트림을 역다중화한 후 엔트로피 디코딩을 수행할 수도 있다.
- [0098] 엔트로피 디코딩부(310)에서 디코딩된 정보 중 예측 블록을 생성하기 위한 정보는 예측부(330)로 제공되고, 엔트로피 디코딩부(310)에서 엔트로피 디코딩이 수행된 레지듀얼 값은 재정렬부(315)로 입력될 수 있다.
- [0099] 재정렬부(315)는 엔트로피 디코딩부(310)에서 엔트로피 디코딩된 비트스트림을 영상 인코딩 장치에서 재정렬한 방법을 기초로 재정렬할 수 있다. 재정렬부(315)는 1차원 벡터 형태로 표현된 계수들을 다시 2차원의 블록 형태의 계수로 복원하여 재정렬할 수 있다. 재정렬부(315)는 인코딩 장치에서 수행된 계수 스캐닝에 관련된 정보를 제공받고 인코딩 장치에서 수행된 스캐닝 순서에 기초하여 역으로 스캐닝하는 방법을 통해 재정렬을 수행할 수 있다.
- [0100] 역양자화부(320)는 인코딩 장치에서 제공된 양자화 파라미터와 재정렬된 블록의 계수값을 기초로 역양자화를 수행할 수 있다.
- [0101] 역변환부(325)는 영상 인코딩 장치에서 수행된 양자화 결과에 대해, 인코딩 장치의 변환부가 수행한 DCT 및 DST에 대해 역DCT 및/또는 역DST를 수행할 수 있다. 역변환은 인코딩 장치에서 결정된 전송 단위 또는 영상의 분할 단위를 기초로 수행될 수 있다. 인코딩 장치의 변환부에서 DCT 및/또는 DST는 예측 방법, 현재 블록의 크기 및 예측 방향 등 복수의 정보에 따라 선택적으로 수행될 수 있고, 디코딩 장치의 역변환부(325)는 인코딩 장치의 변환부에서 수행된 변환 정보를 기초로 역변환을 수행할 수 있다.
- [0102] 인코딩 장치에서 레지듀얼 블록에 다운 샘플링이 수행된 후에 변환이 이루어졌다면, 역변환부(325)는 인코딩 장치에서 수행된 다운 샘플링에 대응하여 역변환된 레지듀얼 블록에 대한 업 샘플링을 수행할 수 있다.
- [0103] 예측부(330)는 엔트로피 디코딩부(310)에서 제공된 예측 블록 생성 관련 정보와 메모리(340)에서 제공된 이전에 디코딩된 블록 및/또는 픽처 정보를 기초로 예측 블록을 생성할 수 있다.
- [0104] 현재 블록에 대한 예측 모드가 인트라 예측(intra prediction) 모드인 경우에, 예측부(340)는 현재 픽처 내의 픽셀 정보를 기초로 예측 블록을 생성하는 인트라 예측을 수행하고 예측 블록을 생성할 수 있다.
- [0105] 현재 블록에 대한 예측 모드가 인터 예측(inter prediction) 모드인 경우에, 예측부(340)는 현재 픽처의 이전 픽처 또는 이후 픽처 중 적어도 하나의 픽처에 포함된 정보를 기초로 현재 블록에 대한 인터 예측을 수행하고 예측 블록을 생성할 수 있다. 이때, 인코딩 장치에서 제공된 현재 블록의 인터 예측에 필요한 움직임 정보, 예컨대 움직임 벡터, 참조 픽처 인덱스 등에 관한 정보는 인코딩 장치로부터 수신한 스킵 플래그, 머지 플래그 등을 확인하고 이에 대응하여 유도될 수 있다.
- [0106] 현재 블록에 대해 인터-뷰 예측이 적용된 경우에, 예측부(340)는 다른 뷰 내의 참조 픽처를 이용하여 현재 블록에 대한 예측을 수행하고 예측 블록을 생성할 수

있다.

[0107] 복원 블록은 예측부(330)에서 생성된 예측 블록과 역변환부(325)에서 제공된 레지듀얼 블록을 이용해 생성될 수 있다. 도 3에서는 가산기에서 예측 블록과 레지듀얼 블록이 합쳐져 복원 블록이 생성되는 것으로 설명하고 있다. 이때, 가산기를 복원 블록을 생성하는 별도의 유닛(복원 블록 생성부)로 볼 수 있다.

[0108] 스킵 모드가 적용되는 경우에는 레지듀얼이 전송되지 않으며 예측 블록을 복원 블록으로 할 수 있다.

[0109] 복원된 블록 및/또는 픽처는 필터부(335)로 제공될 수 있다. 필터부(335)는 필요에 따라서 복원된 블록 및/또는 픽처에 디블록킹 필터링, SAO(Sample Adaptive Offset) 및/또는 ALF 등을 적용할 수 있다.

[0110] 메모리(340)는 복원된 픽처 또는 블록을 저장하여 참조 픽처 또는 참조 블록으로 사용할 수 있도록 할 수 있고 또한 복원된 픽처를 출력부로 제공할 수 있다. 도시되지는 않았으나, 출력부는 복원된 멀티 뷰의 픽처들을 이용하여 3DV 영상을 제공할 수 있다.

[0111] 3D 비디오 인코딩/디코딩에서는 복수의 카메라에서 촬영된 멀티 뷰 비디오 시퀀스가 이용된다. 서로 다른 뷰(view, 視點)에서 촬영된 영상들 사이에서는 글로벌 디스패리티(global disparity)라고 하는 차이(disparity)가 존재한다. 글로벌 디스패리티는 현재 뷰(View)에서 특정 시간의 픽처와 다른 뷰에서 동일한 시간의 픽처 사이에 존재하는 전체적인(global) 변이의 차이라고 할 수 있다. 이때, 두 뷰 사이의 전체적인 변이의 차이는 글로벌 디스패리티 벡터(Global Disparity Vector: GDV, 이하 'GDV' 라 함)를 통해 나타낼 수 있다.

[0112] 도 4는 3D 비디오 인코딩/디코딩 과정에서 GDV를 이용하여 인터-뷰 예측이 수행되는 방법을 개략적으로 설명하는 도면이다. 도 4에서는 설명의 편의를 위해, 각 뷰 내의 텍스처 영상들에 대한 경우를 예로서 설명한다.

[0113] 도 4에서는 멀티 뷰들 중에서 n번째 뷰(Vn)의 픽처(400) 내 현재 블록(410)에 대하여 인터-뷰 예측을 수행하는 경우를 예로서 설명한다.

[0114] 도 4를 참조하면, 현재 블록(410)의 인터-뷰 예측을 위해, 멀티-뷰 중 m번째 뷰(Vm)의 픽처(420) 내 참조 블록(430)의 움직임 정보를 참조할 수 있다. n번째 뷰의 픽처(400)와 m 번째 뷰의 픽처(420)은 동일한 시간, 즉 동일한 POC(Picture Order Count)의 픽처이다. POC는 픽처의 출력 순서를 나타내는 정보이다.

[0115] 서로 다른 뷰에 속하는 현재 블록(410)과 참조 블록(430)의 관계는 GDV_{nm} (450)에 의해 정의될 수 있다. 현재 블록(410)을 m 번째 뷰에 투영한 블록(440)과 참조 블록(430)의 관계를 생각하면, n 번째 뷰에서 현재 블록(410)의 좌상단 픽셀 (x, y) 와 m 번째 뷰에서 참조 블록(430)의 좌상단 픽셀 (x', y') 의 관계는 수식 1과 같이 나타낼 수 있다.

[0116] <수식 1>

[0117] $(x', y') = GDV_{nm} + (x, y)$

[0118] $GDV_{nm} = (dvx_{nm}, dvy_{nm})$

- [0119] n 번째 뷰의 현재 블록(410)은 수식 1과 같이 유도된 GDV에 의해 특정되는 m 번째 뷰의 참조 블록(430)를 참조 블록으로 해서 예측(460)될 수 있다.
- [0120] 인코딩 장치는 동일한 POC에서 서로 다른 뷰의 두 픽처에 대한 오차를 최소화하는 디스패리티를 해당 뷰 간의 글로벌 디스패리티로 선택할 수 있다. 예컨대, 두 뷰 사이에서 최적의 SAD(Sum of Absolute Differences) 값을 가지는 디스패리티를 블록 단위로 결정할 수 있다. 이때, SAD 대신 MSE(Mean Square Error), MAD(Mean Absolute Difference) 등이 이용될 수도 있다.
- [0121] GDV(Global Disparity Vector, 전역 변이 벡터)는 선택된 글로벌 디스패리티를 나타내는 벡터 표현이 된다.
- [0122] 인코딩 장치는 산출한 GDV 정보를 비트스트림으로 전송하여 디코딩 장치에서도 동일한 GDV가 사용되도록 할 수 있다.
- [0123] GDV를 이용하여 인터-뷰 예측을 하는 경우에는, GDV에 의해 현재 뷰의 현재 블록에 대응하는 참조 뷰(reference view)의 참조 픽처 내 블록(참조 블록)에서 움직임 정보를 가져와 현재 블록의 예측에 이용할 수 있다. 예컨대, 참조 블록의 움직임 정보(예컨대, 움직임 벡터)를 복사하여 현재 블록의 시간적 움직임 벡터(temporal motion vector)로 이용하거나, 현재 블록의 움직임 벡터 예측자(motion vector predictor)로 이용할 수 있다.
- [0124] 도 4에서는 텍스처 영상들에 대한 경우를 예로서 설명하였으나, 본 발명은 이에 한정하지 않는다. 도 4의 예에서 설명한 내용은 깊이 맵들에 대한 경우 또는 깊이 맵과 텍스처 영상 간의 경우에도 동일하게 적용될 수 있다.
- [0125] 도 5는 멀티 뷰의 비디오를 디코딩 하는 경우의 일 예를 개략적으로 설명하는 도면이다. 도 5의 예에서는 세 뷰(V0, V1, V2)의 비디오를 디코딩하는 경우로서, 각 뷰의 픽처들이 POC에 따라서 시퀀스를 이루고 있으며, POC 순서상 연속된 8개의 픽처가 하나의 그룹(Group Of Picture: GOP)를 이루는 경우를 일 예로서 설명하고 있다. 또한 도 5에서도 설명의 편의를 위해 텍스처 영상에 대한 경우를 예로서 설명한다.
- [0126] 도 5에 있어서, GDV_{10} 은 뷰 V0를 참조 뷰(reference view)로 해서 뷰 V1의 현재 블록을 예측하는데 이용되는 GDV이며, GDV_{12} 는 뷰 V2를 참조 뷰로 해서 뷰 V1의 현재 블록을 예측하는데 이용되는 GDV이고, GDV_{20} 은 뷰 V0을 참조 뷰로 해서 뷰 V2의 현재 블록을 예측하는데 이용되는 GDV이다. GDV는 서로 다른 두 뷰(view)의 동일한 시점(time, 즉 POC)에 있는 픽처들 사이에서, 두 블록의 대응 관계를 지시한다.
- [0127] 각 GDV는 상술한 바와 같이 인코딩 장치로부터 비트스트림으로 전송되거나 되며, 디코딩 장치는 비트스트림으로부터 GDV를 디코딩하여 이용할 수 있다.
- [0128] 도 5를 참조하면, 뷰 V2에서 현재 픽처(500)의 현재 블록(520)에 대한 예측을 뷰 V0의 참조 픽처(510) 내 블록(530, 참조 블록)을 참조하여 수행한다. 현재 픽처(500)와 참조 픽처(510)는 동일한 POC (POC=2)를 가진다. $GDV_{20} = (dvx_{20}, dvy_{20})$ 는 상술한 바와 같이, 현재 뷰(V2)를 디코딩할 때, 참조 뷰(V0)에서 참조

블록의 위치를 특정하기 위한 GDV(Global Disparity Vector)이다.

[0129] 수식 1을 참조하면, V0에서 현재 픽처(500) 내 현재 블록의 좌상단 위치가 (x, y)인 경우에, V2(참조 뷰)에서 참조 픽처(510) 내 대응 블록(430, 참조 블록)의 좌상단 위치 (x', y')는 수식 2와 같이 특정될 수 있다.

[0130] <수식 2>

[0131] $(x', y') = (x, y) + \mathbf{GDV}_{20}$

[0132] 인코딩 장치 및 디코딩 장치는 인터-뷰 예측의 경우에 GDV에 의해 결정되는 참조 뷰 내 블록의 움직임 정보를 참조하여 현재 블록에 대한 예측을 수행한다. 예컨대, 도 5의 예에서, 인코딩 장치 및 디코딩 장치는 \mathbf{GDV}_{20} 에 의해 특정되는 위치 (x', y')의 참조 블록(530)에서 움직임 정보를 복사하여 현재 블록(520)의 시간적 움직임 벡터로 이용하거나 현재 블록(520)의 움직임 벡터 예측자로 이용할 수 있다.

[0133] 도 5에서는 텍스처 영상들에 대한 경우를 예로서 설명하였으나, 본 발명은 이에 한정하지 않는다. 도 5의 예에서 설명한 내용은 깊이 맵들에 대한 경우 또는 깊이 맵과 텍스처 영상 간의 경우에도 동일하게 적용될 수 있다.

[0134] 한편, 멀티-뷰 비디오 영상(video)을 구성하는 각 뷰의 영상은 동일한 장면을 복수의 카메라(예컨대, 나란히 배치된 카메라들)을 통해 촬영하여 얻어진다. 따라서, 이웃하는 뷰의 영상들 간에는 텍스처 상관도(texture correlation)와 움직임 상관도(motion correlation)가 높다.

[0135] 멀티-뷰를 이용하는 비디오 코딩에 있어서, 인터-뷰 예측(inter-view prediction)을 수행할 때는 이웃하는 뷰들 간의 높은 텍스처 상관도를 활용할 수 있다. 예컨대, 인터-뷰 예측에서는 이웃하는 뷰의 픽처를 참조 픽처로 이용하여 인터 예측을 수행할 수 있다.

[0136] 인터-뷰 예측을 할 때, 이웃 뷰(참조 뷰)와의 텍스처 상관도가 높다면, 참조 뷰의 대응 블록에서 인터 예측에 사용된 움직임 정보를 현재 뷰의 현재 블록에 그대로 적용할 수 있다. 예컨대, 인터-뷰 예측에 있어서 스킵 모드를 적용할 수 있다.

[0137] 도 6은 인터-뷰 예측의 일 예로서, 스킵 모드가 적용되는 예를 개략적으로 설명하는 도면이다.

[0138] 도 6에서는 설명의 편의를 위해 인터-뷰 예측이 수행되는 두 뷰(참조 뷰와 현재 뷰) 간에 스킵 모드가 적용되는 예를 설명한다.

[0139] 인터-뷰 예측에서 스킵 모드는 이웃하는 뷰 사이의 높은 움직임 상관도를 활용한다. 참조 뷰의 인터 예측 모드에 사용된 움직임 정보를 현재 뷰의 스킵 모드 코딩에 적용할 수 있다.

[0140] 도 6을 참조하면, 앵커(anchor) 픽처의 POC_A 에서 참조 뷰의 움직임 정보를 기반으로 현재 뷰의 움직임 정보가 예측된다. 논 앵커(non-anchor) 픽처의 현재 $\text{POC}(\text{POC}_{\text{cur}})$ 에서도 현재 블록(610)은 참조 뷰 내 대응 블록(620)에 대한 GDV를 기반으로 인터-뷰 예측된다. GDV는 참조 하는 이웃 뷰의 참조 픽처에서 현재 블록에 대응하는 블록(대응 블록)의 위치를 산출하기 위해 사용된다. 산출된

대응 블록에서 사용된 움직임 정보는 스킵 모드가 적용되는 경우에 현재 블록의 코딩에 그대로 이용된다.

- [0141] 앵커 픽처는 동일한 액세스 유닛 내에서 모든 슬라이스가 참조 슬라이스들이 픽처들이다. 따라서, 앵커 픽처에 인터 예측(inter prediction)은 이용되지 않으며 인터-뷰 예측(inter-view)이 이용된다. 또한, POC 순서상 앵커 픽처를 뒤따르는 픽처들도 앞서 코딩된 픽처들의 정보를 이용한 인터 예측을 사용하지 않고 인터-뷰 예측을 사용하여 예측될 수 있다.
- [0142] 한편, 인터 예측의 경우에는 주변 블록의 움직임 정보를 재활용하여 현재 블록에 대한 예측을 수행할 수 있다. 이때, 주변 블록은 현재 픽처 내에서 현재 블록의 공간적 주변 블록과 다른 픽처 내에서 현재 블록에 대응하는 블록(collocated block)을 포함한다.
- [0143] 예컨대, 머지 모드와 스킵 모드의 경우에는 주변 블록의 움직임 정보들 중 어느 하나를 현재 블록의 움직임 정보로 이용한다. 스킵 모드의 경우에 인코딩 장치는 레지듀얼 신호를 전송하지 않고, 디코딩 장치는 주변 블록의 움직임 정보를 이용하여 생성한 예측 신호를 현재 블록의 복원 신호로서 이용할 수 있다. MVP를 이용하는 경우에도, 인코딩 장치는 주변 블록들의 움직임 벡터 중 어느 하나를 현재 블록의 움직임 벡터 예측자로 사용하여 움직임 벡터의 차이값을 전송할 수 있고, 디코딩 장치는 수신한 움직임 벡터의 차이값을 기반으로 현재 블록에 대한 움직임 벡터를 복원할 수 있다.
- [0144] 상술한 바와 같이, 인터 예측의 경우에는 주변 블록의 움직임 정보를 재활용하여 현재 블록의 움직임 정보를 예측할 수 있다.
- [0145] 도 7은 머지 모드를 이용한 인터 예측에서 현재 블록의 예측에 이용 가능한 주변 블록의 일 예를 개략적으로 설명하는 도면이다.
- [0146] 도 7을 참조하면, 현재 블록(750) 주변 블록들의 움직임 벡터 정보 중 상측 블록(above block, 700)의 움직임 벡터 정보, 좌측 블록(left block, 710)의 움직임 벡터 정보, Col 블록(co-located block, 720)의 움직임 벡터 정보, 좌하단(bottom left corner) 블록(730)의 움직임 벡터 정보, 우상측(top right corner) 블록(740)의 움직임 벡터 정보들을 현재 블록의 움직임 벡터 예측자(motion vector predictor) 후보로 이용할 수 있다.
- [0147] 움직임 벡터 예측자는 해당 블록에서 사용된 움직임 벡터 정보, 참조 픽처 인덱스 정보, 예측 방향 정보 등을 포함할 수 있다. 움직임 벡터 예측자는 또한, 해당 블록에서 사용된 움직임 벡터 정보를 포함하고, 현재 픽처의 예측에 이용될 참조 픽처 인덱스는 인코딩 장치에서 결정되어 디코딩 장치로 전송될 수도 있다. 한편, 머지 모드 또는 스킵 모드에서 시간적 머지 후보에 대한 참조 픽처 인덱스(예컨대, Col 블록에 대한 참조 픽처 인덱스)는 Col 블록이 참조하는 픽처와 무관하게 참조 픽처 리스트 내의 후보 참조 픽처들 중에서 특정 참조 픽처가 사용될 수도 있다.
- [0148] 인코딩 장치는 움직임 벡터 예측자 후보들 중에서 현재 블록의 움직임 벡터

예측자를 선택할 수 있다. 예컨대, 인코딩 장치는 RDO(Rate Distortion Optimization), 압축 효율 등을 고려하여 현재 블록의 움직임 정보 예측자를 결정할 수 있다. 디코딩 장치는 인코딩 장치로부터 어떤 움직임 정보 예측자를 사용할 것인지를 지시하는 정보를 수신하고, 해당 움직임 정보 예측자를 이용하여 현재 블록에 대한 예측을 수행할 수 있다.

- [0149] 한편, 스킵 모드와 MVP를 이용하는 예측 모드에서도 도 7에 도시된 바와 같은 움직임 벡터 예측자 후보를 머지 모드와 동일하게 이용할 수 있다. MVP를 이용하는 경우에는 참조 픽처 인덱스가 별도로 인코딩 장치로부터 디코딩 장치로 전송될 수도 있다.
- [0150] 도 8은 인터 예측에서 현재 블록의 예측에 이용 가능한 주변 블록의 다른 예를 개략적으로 설명하는 도면이다.
- [0151] 인코딩 장치와 디코딩 장치는 현재 블록(860) 주변 블록들의 움직임 정보를 이용하여 현재 블록에 대한 예측을 수행할 수 있다. 도 8을 참조하면, 현재 블록(860)의 좌상측 블록(800), 현재 블록(860)의 상측 블록(810), 현재 블록의 우상측 블록(820), 현재 블록(860)의 좌하단 블록(830), 현재 블록(860)의 좌측 블록(840)과 다른 픽처에서 현재 블록(860)에 대응하는 Col 블록(collocated block, 850)의 움직임 정보 중 어느 하나를 현재 블록에 대한 움직임 정보 예측자(predictor)로 사용할 수 있다.
- [0152] 인코딩 장치는 움직임 벡터 예측자 후보들 중에서 현재 블록의 움직임 벡터 예측자를 선택할 수 있다. 예컨대, 인코딩 장치는 RDO(Rate Distortion Optimization), 압축 효율 등을 고려하여 현재 블록의 움직임 정보 예측자를 결정할 수 있다. 디코딩 장치는 인코딩 장치로부터 어떤 움직임 정보 예측자를 사용할 것인지를 지시하는 정보를 수신하고, 해당 움직임 정보 예측자를 이용하여 현재 블록에 대한 예측을 수행할 수 있다.
- [0153] 상술한 바와 같이, 움직임 벡터 예측자는 대응하는 블록에 적용된 움직임 벡터, 참조 픽처 인덱스 등을 포함할 수 있다. 예컨대, 머지 모드 또는 스킵 모드의 경우에, 선택된 움직임 벡터 예측자의 정보로서 주변 블록에 대한 움직임 벡터, 참조 픽처 인덱스를 현재 블록에 적용할 수 있다. MVP를 이용하는 경우에는 선택된 움직임 벡터 예측자의 정보로서 주변 블록에 대한 움직임 벡터를 현재 블록에 적용할 수 있다. MVP를 이용하는 경우에, 참조 픽처 인덱스는 인코딩 장치로부터 디코딩 장치로 전송될 수 있다. 한편, 머지 모드 또는 스킵 모드에서 시간적 머지 후보에 대한 참조 픽처 인덱스(예컨대, Col 블록에 대한 참조 픽처 인덱스)는 Col 블록이 참조하는 픽처와 무관하게 참조 픽처 리스트 내의 후보 참조 픽처들 중에서 특정 참조 픽처가 사용될 수도 있다.
- [0154] 인터-뷰 예측의 경우에도, 도 7 및 도 8의 예와 같이, 주변 블록의 움직임 정보를 이용하여 현재 블록에 대한 예측을 수행할 수 있다. 인터-뷰 예측에서 주변 블록의 정보를 이용하는 경우에, 주변 블록은 다른 뷰에서 현재 블록에 대응하는 블록을 포함한다.

- [0155] 도 9는 인터-뷰 예측에서 사용될 수 있는 움직임 벡터 예측자의 일 예를 개략적으로 설명하는 도면이다.
- [0156] 도 9에서는 현재 뷰(V1)의 예측에 있어서 참조 뷰(V0)의 정보를 참조할 수 있는 경우를 예로서 설명하고 있다. 또한, 도 9에서는 도 7의 예에서와 같이 현재 픽처(900) 내 공간적 인접 블록들인 현재 블록(915)의 상측 블록(930), 현재 블록(915)의 우상측 블록(935), 현재 블록(915)의 좌측 블록(925), 현재 블록(915)의 좌하단 블록(920)의 움직임 정보(예측자)를 이용하여 현재 블록(915)에 대한 인터 예측을 수행할 수 있다. 즉, 인코딩 장치/디코딩 장치는 현재 블록(915)의 상측 블록(930)의 움직임 정보(960), 현재 블록(915)의 우상측 블록(935)의 움직임 정보(965), 현재 블록(915)의 좌측 블록(925)의 움직임 정보(955), 현재 블록(915)의 좌하단 블록(920)의 움직임 정보(950)를 움직임 벡터 예측자 후보로 이용하여 현재 블록(915)에 대한 인터 예측을 수행할 수 있다.
- [0157] 또한, 현재 블록(915)의 주변 블록으로서 현재 뷰(V1)의 다른 픽처(905)에서 현재 블록(915)에 대응하는 Col 블록(940)을 이용할 수도 있다. 인코딩 장치/디코딩 장치는 Col 블록(940)의 움직임 정보(970)를 움직임 벡터 예측자 후보로 이용하여 현재 블록(915)에 대한 인터 예측을 수행할 수 있다.
- [0158] 아울러, 현재 블록의 주변 블록으로서 다른 뷰(참조 뷰, V0)의 픽처들 중 현재 픽처(900)와 동일한 POC의 픽처(참조 픽처, 910)에서 현재 블록(915)에 대응하는 참조 블록(945)을 이용할 수도 있다. 인코딩 장치/디코딩 장치는 다른 뷰(V0)의 참조 블록(945)의 움직임 정보(975)를 움직임 벡터 예측자 후보로 이용하여 현재 블록(915)에 대한 인터-뷰 예측을 수행할 수 있다.
- [0159] 인코딩 장치는 상기 움직임 벡터 예측자 후보들 중 현재 블록(915)의 예측에 어떤 움직임 벡터 예측자를 이용할 것인지를 결정하고 결정된 움직임 벡터 예측자에 관한 정보를 디코딩 장치에 전송할 수 있다. 디코딩 장치는 인코딩 장치로부터 전송된 정보를 기반으로 현재 블록(915)의 예측에 이용할 움직임 벡터 예측자를 유도할 수 있다.
- [0160] 상술한 바와 같이, 도 9의 예에서는 현재 블록(915)에 대한 움직임 정보 예측자 후보에 인터-뷰 예측을 위한 예측자(predictor)를 추가하여 현재 블록에 대한 예측을 수행할 수 있다. 예컨대, 도 7과 같은 움직임 벡터 예측자 후보들에 인터-뷰 예측을 위한 움직임 벡터 예측자(인터-뷰 움직임 벡터 예측자)를 추가하여, 스킵 모드와 같은 인터 예측 모드들에 적용하는 방식으로 주변 블록들의 움직임 정보를 재활용할 수 있다.
- [0161] 인터-뷰 예측의 경우에 사용되는 움직임 벡터 예측자(975)는 도시된 바와 같이 GDV를 이용하여 찾은 대응 블록(945)의 움직임 정보로서, 움직임 벡터, 참조 픽처 인덱스(refIdx), 인터 예측 방향 등의 정보 중 적어도 하나의 정보를 포함할 수 있다.
- [0162] 움직임 벡터 예측자(975)는 인터-뷰 예측과 인터 예측의 각 예측 모드(스킵 모드, 머지 모드, MVP 사용 모드)에 적용될 수 있으며, GDV를 이용하여

대응하는 블록으로 결정된 참조 뷰(V0)의 블록(945)이 인터 예측이 적용된 블록이 아닌 경우에는 인터-뷰 예측을 위한 움직임 벡터 예측자는 정의되지 않거나 사용되지 않을 수 있다.

- [0163] 도 10은 인터-뷰 예측에서 사용될 수 있는 움직임 벡터 예측자의 다른 예를 개략적으로 설명하는 도면이다.
- [0164] 도 10에서는 현재 뷰(V1)의 예측에 있어서 참조 뷰(V0)의 정보를 참조할 수 있는 경우를 예로서 설명하고 있다. 또한, 도 10에서는 도 8의 예에서와 같이 현재 픽처(1000) 내 공간적 인접 블록들인 현재 블록(1015)의 위상단 블록(1020), 현재 블록(1015)의 상측 블록(1025), 현재 블록(1015)의 좌상단 블록(1030), 현재 블록(1015)의 좌하단 블록(1035), 현재 블록(1015)의 좌측 블록(1040)의 움직임 정보(예측자)를 이용하여 현재 블록(1015)에 대한 인터 예측을 수행할 수 있다. 즉, 인코딩 장치/디코딩 장치는 현재 블록(1015)의 위상단 블록(1020)의 움직임 정보(1055), 현재 블록(1015)의 상측 블록(1025)의 움직임 정보(1060), 현재 블록(1015)의 좌상단 블록(1030)의 움직임 정보(1065), 현재 블록(1015)의 좌하단 블록(1035)의 움직임 정보(1070), 현재 블록(1015)의 좌측 블록(1040)의 움직임 정보(1075)를 움직임 벡터 예측자 후보로 이용하여 현재 블록(1015)에 대한 인터 예측을 수행할 수 있다.
- [0165] 또한, 현재 블록(1015)의 주변 블록으로서 현재 뷰(V1)의 다른 픽처(1005)에서 현재 블록(1015)에 대응하는 Col 블록(1045)을 이용할 수도 있다. 인코딩 장치/디코딩 장치는 Col 블록(1045)의 움직임 정보(1080)를 움직임 벡터 예측자 후보로 이용하여 현재 블록(1015)에 대한 인터 예측을 수행할 수 있다.
- [0166] 아울러, 현재 블록의 주변 블록으로서 다른 뷰(참조 뷰, V0)의 픽처들 중 현재 픽처(1000)와 동일한 POC의 픽처(참조 픽처, 1010)에서 현재 블록(1015)에 대응하는 참조 블록(1050)을 이용할 수도 있다. 인코딩 장치/디코딩 장치는 다른 뷰(V0)의 참조 블록(1050)의 움직임 정보(1085)를 움직임 벡터 예측자 후보로 이용하여 현재 블록(1015)에 대한 인터-뷰 예측을 수행할 수 있다.
- [0167] 인코딩 장치는 상기 움직임 벡터 예측자 후보들 중 현재 블록(1015)의 예측에 어떤 움직임 벡터 예측자를 이용할 것인지를 결정하고 결정된 움직임 벡터 예측자에 관한 정보를 디코딩 장치에 전송할 수 있다. 디코딩 장치는 인코딩 장치로부터 전송된 정보를 기반으로 현재 블록(1015)의 예측에 이용할 움직임 벡터 예측자를 유도할 수 있다.
- [0168] 상술한 바와 같이, 도 10의 예에서는 현재 블록(1015)에 대한 움직임 정보 예측자 후보에 인터-뷰 예측을 위한 예측자(predictor)를 추가하여 현재 블록에 대한 예측을 수행할 수 있다. 예컨대, 도 8과 같은 움직임 벡터 예측자 후보들에 인터-뷰 예측을 위한 움직임 벡터 예측자(인터-뷰 움직임 벡터 예측자)를 추가하여, 스킵 모드와 같은 인터 예측 모드들에 적용하는 방식으로 주변 블록들의 움직임 정보를 재활용할 수 있다.
- [0169] 인터-뷰 예측의 경우에 사용되는 움직임 벡터 예측자(1085)는 도시된 바와 같이

GDV를 이용하여 찾은 대응 블록(1050)의 움직임 정보로서, 움직임 벡터, 참조 픽처 인덱스(refIdx), 인터 예측 방향 등의 정보 중 적어도 하나의 정보를 포함할 수 있다.

[0170] 움직임 벡터 예측자(1085)는 인터-뷰 예측과 인터 예측의 각 예측 모드(스킵 모드, 머지 모드, MVP 사용 모드)에 적용될 수 있으며, GDV를 이용하여 대응하는 블록으로 결정된 참조 뷰(V0)의 블록(1050)이 인터 예측이 적용된 블록이 아닌 경우에는 인터-뷰 예측을 위한 움직임 벡터 예측자는 정의되지 않거나 사용되지 않을 수 있다.

[0171] 스킵 모드 또는 머지 모드가 적용되는 경우에, 움직임 벡터 예측자들(1055, 1060, 1065, 1070, 1075, 1080, 1085)은 움직임 벡터 외에 현재 블록의 예측에 이용될 참조 픽처를 지시하는 정보를 포함할 수도 있다. 혹은 MVP를 이용하는 경우에, 움직임 벡터 예측자들(1055, 1060, 1065, 1070, 1075, 1080, 1085)은 움직임 벡터에 관한 정보를 포함하며, 현재 블록에 대한 참조 픽처를 지시하는 정보는 별도로 디코딩 장치에 전송되도록 할 수도 있다.

[0172] 상술한 바와 같이, 인터-뷰 예측에서도 현재 블록에 대응하는 다른 뷰(참조 뷰) 내 블록(대응 블록 혹은 참조 블록)의 움직임 정보를 재활용하여 현재 뷰 내 대상 블록(현재 블록)의 예측을 수행할 수 있다.

[0173] 인코딩 장치는 이를 위해 슬라이스 헤더에서 필요한 정보를 전송할 수 있다. 디코딩 장치는 수신한 정보를 기반으로 현재 블록에 대한 인터-뷰 예측을 수행할 수 있다.

[0174] 표 1은 슬라이스 헤더에서 전송되는 정보의 일 예를 개략적으로 나타낸다.

[0175] <표 1>

[0176]

| |
|--|
| slice header () { |
| inter view motion reuse info flag |
| if(inter view motion reuse info flag) { |
| inter view motion reuse dir |
| global disparity vector x |
| global disparity vector y |
| } |
| } |

[0177] 표 1을 참조하면, 인코딩 장치는 슬라이스 헤더에서 인터-뷰 예측을 위해 필요한 정보를 전송한다. 디코딩 장치는 수신한 비트스트림을 파싱 또는 엔트로피 디코딩하여 슬라이스 헤더에서 인터-뷰 예측을 위해 필요한 정보를 획득하고 이를 기반으로 인터-뷰 예측을 수행할 수 있다.

[0178] **inter_view_motion_reuse_info_flag**는 인터-뷰 예측에 있어서 해당 슬라이스 헤더에 대응하는 현재 슬라이스에 움직임 정보(예컨대 움직임 벡터)를 재사용하는 것에 관한 정보가 전송되는지를 지시한다. 예컨대, 움직임 정보(예컨대 움직임 벡터)를 재사용하는 것에 관한 정보는 움직임 정보를 재사용하는지를 지시하는 정보를 포함한다. 움직임 정보의 재사용은 현재

블록의 주변 블록의 움직임 정보를 이용하여 현재 블록에 대한 예측을 수행하는 것을 포함하며, 현재 블록의 주변 블록은 다른 뷰에서 현재 블록에 대응하는 블록을 포함한다.

- [0179] **inter_view_motion_reuse_dir**은 인터-뷰 예측에 있어서 움직임 정보를 재사용하는 경우(움직임 정보를 주변블록으로부터 가져오는 경우)에 현재 뷰가 참조하는 뷰를 특정한다. 예를 들어, **inter_view_motion_reuse_dir**의 값이 0이면 제1 방향의 뷰를 참조하고, **inter_view_motion_reuse_dir**의 값이 1이면 제1 방향과 반대 방향인 제2 방향의 뷰를 참조하는 것을 지시하도록 할 수 있다. 예컨대, 멀티-뷰를 구성하는 뷰들을 각 시점(view point)에 따라 배치하였을 때, 오른쪽 뷰로부터 왼쪽 뷰로의 방향을 L0 방향이라고 하고, 왼쪽 뷰로부터 오른쪽 뷰로의 방향을 L1 방향이라고 설정할 수 있다.
- [0180] 도 11은 참조 뷰의 방향을 개략적으로 설명하는 도면이다. 도 11에서는, 동일 시점(POC_i)에서 동일 객체(1110)을 촬영한 5 개의 뷰들(V0~V4)이 존재하는 경우를 예로서 설명한다. 여기서 5 개의 뷰들이 존재하는 경우를 예로서 설명하나, 본 발명은 이에 한정되지 않으며, 뷰들은 5개보다 더 적을 수도 있고, 더 많을 수도 있다.
- [0181] 도 11을 참조하면, 각 뷰들에 대하여 인터-뷰 예측을 적용하는 경우에 상술한 바와 같이, 좌측 방향으로 다른 뷰들을 참조하는 것을 L0 방향이라고 하고, 우측 방향으로 다른 뷰들을 참조하는 것을 L1 방향이라고 할 수 있다.
- [0182] 따라서, 도 11의 예를 참조할 때, 현재 뷰의 뷰 ID가 1이고 L0 방향으로 뷰 ID가 0인 뷰를 참조하고, L1 방향으로 뷰 ID가 2인 뷰를 참조하는 경우, **inter_view_motion_reuse_dir**의 값이 0이면 뷰 ID가 0인 뷰의 움직임 정보를 이용하고, **inter_view_motion_reuse_dir**의 값이 1이면 뷰 ID가 2인 뷰의 움직임 정보를 이용하도록 할 수 있다.
- [0183] **global_disparity_vector_x**, **global_disparity_vector_y**는 현재 블록(예컨대, 디코딩하는 TU, PU 또는 CU)의 위치를 기준으로 참조 뷰 내의 참조 픽처(예컨대, 현재 픽처와 동일 POC의 픽처)에서 움직임 정보를 참조하려는 블록의 위치를 특정한다. 다시 말하면, **global_disparity_vector_x**, **global_disparity_vector_y**는 현재 블록에 대한 GDV의 x 성분과 y 성분을 특정한다. 현재 블록에 대한 GDV는 참조 픽처 내에서 현재 블록의 위치에 대응하는 위치와 참조 블록의 위치 사이의 차이(혹은 현재 픽처에서 현재 블록의 위치와 참조 블록의 위치에 대응하는 위치 사이의 차이)를 나타낸다. **global_disparity_vector_x**, **global_disparity_vector_y**의 값, 즉 GDV의 값은 소정의 픽셀 단위, 예컨대, 4 픽셀, 8 픽셀 또는 16 픽셀 단위로 정의될 수도 있다.
- [0184] 한편, 픽처 내의 배경과 전경이 서로 다른 움직임(motion)을 가지는 경우에는 매 참조 픽처마다 하나의 GDV를 정의하는 것보다 현재 픽처 내의 영역별로 여러 개의 GDV를 정의하는 것이 더 효율적이다.
- [0185] 도 12는 현재 픽처의 각 영역별로 GDV를 정의하여 인터-뷰 예측에 적용하는

방법의 일 예를 개략적으로 설명하는 도면이다.

- [0186] 도 12를 참조하면, 현재 픽처(슬라이스)는 쿼드 트리(Quad-Tree) 방식으로 동일한 움직임 값을 가지는 영역으로 분할된다. 동일한 움직임 값을 가지는 영역마다 각각 최적의 GDV 정보가 할당된다. 현재 블록에 대한 GDV는 픽처 내에서 현재 블록이 속하는 영역의 GDV로 결정될 수 있다.
- [0187] 도 12에서는 현재 픽처(1200)가 깊이 2까지 분할(2번 분할)되는 경우를 예로서 설명한다. 현재 픽처(1200)가 네 개의 영역으로 분할된 1차 분할에서 좌상측 블록, 우상측 블록, 좌하측 블록은 블록 내 움직임이 균일하고 우하측 블록은 다시 움직임이 동일한 네 영역으로 분할된다고 가정한다.
- [0188] 분할의 결과, 7 개의 블록 중 세 블록(1210, 1240, 1270)은 동일한 움직임을 가지며, 가장 큰 GDV인 GDV0을 공통으로 사용한다. 다른 두 블록(1220, 1250) 역시 동일한 움직임으로 가지며 GDV1을 공통으로 사용한다. 나머지 두 블록(1130, 1160)도 동일한 움직임을 가지며 가장 작은 GDV인 GDV2를 공통으로 사용한다.
- [0189] 이때, GDV 정보를 할당하기 위한 쿼드 트리 분할(quad-tree partition)의 최소 단위를 소정의 크기로 한정할 수도 있다. 예컨대, 쿼드 트리 분할의 최소 단위를 LCU, CU, PU 또는 TU로 한정할 수도 있고, 처리 단위가 아닌 소정의 크기(예컨대, 4x4 픽셀, 8x8 픽셀, 16x16 픽셀, 32x32 픽셀 또는 64x64 픽셀)로 한정될 수도 있다.
- [0190] 현재 블록에 대한 GDV는 현재 픽처 내에서 현재 블록이 속하는 영역의 GDV로 결정될 수 있다. 인코딩 장치에서는 현재 블록의 인터-뷰 예측에 사용될 GDV의 정보를 인코딩 장치로 시그널링하며, 이때 GDV는 현재 블록을 기준으로 참조 뷰의 픽처 내 특정 파티션(블록, 영역)의 위치를 지시한다. 디코딩 장치는 현재 블록(CU 또는 LCU)를 디코딩할 때, 인터-뷰 예측을 위하여 GDV를 할당한 쿼드 트리 파티션 정보를 참조하여 어떤 GDV를 사용할 것인지를 판단할 수 있다.
- [0191] 예를 들어, 인코딩 장치는 현재 블록의 인터-뷰 예측을 위한 GDV를 전송하고, 디코딩 장치는 수신한 GDV를 이용하여 현재 블록에 대한 인터-뷰 예측을 수행할 수 있다. 이때, GDV는 쿼드 트리 파티셔닝에 의해 분할된 현재 픽처의 각 영역별로 중 결정될 수 있다.
- [0192] 표 2는 인터-뷰 예측을 위해 시퀀스 파라미터 셋(sequence parameter set)으로 전송되는 GDV에 관한 쿼드 트리 정보의 일 예를 나타낸 것이다.

[0193] <표 2>

| | |
|--------|------------------------------------|
| [0194] | seq_parameter_set_rbsp() { |
| | ... |
| | max_gdv_qtree_depth |
| | max_gdv_unit_size_log_scale |
| | ... |

[0195] 인코딩 장치는 시퀀스 파라미터 셋에서 GDV가 정의되는 최대 깊이 정보와

GDV가 할당되는 최대 블록의 크기에 관한 정보를 전송한다.

[0196] 표 2에서, **max_gdv_qtree_depth**는 GDV에 관한 쿼드-트리 구조의 최대 깊이를 특정한다. 즉, **max_gdv_qtree_depth**는 GDV 정보를 보내는 쿼드-트리 구조에서의 최대 깊이를 나타낸다.

[0197] **max_gdv_unit_size_log_scale**는 GDV 정보에 관한 쿼드-트리 구조에서 최대 영역의 크기를 특정한다. 예컨대, **max_gdv_unit_size_log_scale**는 GDV 정보를 보내는 쿼드-트리 구조에서 최대 영역(예컨대, 최대 유닛)의 크기를 기준 블록의 크기로 나누어 스케일링한 값에 로그(log2)를 취한 값을 나타낸다. 이때, 기준 블록으로서 LCU(Largest CU)를 사용할 수 있다.

[0198] 표 3은 영역별로 GDV를 설정하는 경우에 인터-뷰 예측을 위해 슬라이스 헤더에서 전송되는 정보의 일 예를 나타낸 것이다.

[0199] <표 3>

```
[0200] slice_header ( ) {
    inter_view_motion_info_flag
    if(inter_view_motion_info_flag) {
        num_gdv
        for( i=0 ; i<num_gdv ; i++ ) {
            inter_view_motion_reuse_dir[i]
            global_disparity_vector_x[i]
            global_disparity_vector_y[i]
        }
        if( num_gdv>1 ) {
            gdv_qtree(0,0)
        }
    }
    gdv_qtree(i,depth) {
        if( depth<maxGdvQtreeDepth ) {
            split_flag
            if (depth<maxGdvQtreeDepth && split_flag==1 ) {
                for( i=0 ; i<4 ; i++ ) {
                    gdv_qtree(i,depth+1)
                }
            }
        }
    }
    if(depth == maxGdvQtreeDepth || split_flag==0) {
        gdv_id
    }
}
```

[0201] 표 3을 참조하면, 인코딩 장치는 슬라이스 헤더에서 인터-뷰 예측을 위해 필요한 정보를 전송한다. 디코딩 장치는 수신한 비트스트림을 파싱 또는 엔트로피 디코딩하여 슬라이스 헤더에서 인터-뷰 예측을 위해 필요한 정보를 획득하고 이를 기반으로 인터-뷰 예측을 수행할 수 있다.

[0202] **inter_view_motion_reuse_info_flag**는 인터-뷰 예측에 있어서 해당 슬라이스

헤더에 대응하는 현재 슬라이스에 움직임 정보(예컨대 움직임 벡터)를 재사용하는 것에 관한 정보가 전송되는지를 지시한다. 예컨대, 움직임 정보(예컨대 움직임 벡터)를 재사용하는 것에 관한 정보는 움직임을 재사용하는지를 지시하는 정보를 포함한다. 움직임 정보의 재사용은 현재 블록의 주변 블록의 움직임 정보를 이용하여 현재 블록에 대한 예측을 수행하는 것을 포함하며, 현재 블록의 주변 블록은 다른 뷰에서 현재 블록에 대응하는 블록을 포함한다.

- [0203] **inter_view_motion_reuse_dir**은 인터-뷰 예측에 있어서 움직임을 재사용하는 경우(움직임을 주변블록으로부터 가져오는 경우)에 현재 뷰가 참조하는 뷰를 특정한다. 예를 들어, **inter_view_motion_reuse_dir**의 값이 0이면 제1 방향의 뷰를 참조하고, **inter_view_motion_reuse_dir**의 값이 1이면 제1 방향과 반대 방향인 제2 방향의 뷰를 참조하는 것을 지시하도록 할 수 있다. 가령, 멀티-뷰를 구성하는 뷰들을 각 시점(view point)에 따라 배치하였을 때, 왼쪽 뷰부터 오른쪽 뷰로의 방향을 L0 방향이라고 하고, 오른쪽 뷰로부터 왼쪽 뷰로의 방향을 L1 방향이라고 하자. 현재 뷰의 뷰 ID가 1이고 L0 방향으로 뷰 ID가 0인 뷰를 참조하고, L1 방향으로 뷰 ID가 2인 뷰를 참조하는 경우, **inter_view_motion_reuse_dir**의 값이 0이면 뷰 ID가 0인 뷰의 움직임을 이용하고, **inter_view_motion_reuse_dir**의 값이 1이면 뷰 ID가 2인 뷰의 움직임을 이용하도록 할 수 있다.
- [0204] **global_disparity_vector_x**, **global_disparity_vector_y**는 현재 블록(예컨대, 디코딩하는 TU, PU 또는 CU)의 위치를 기준으로 참조 뷰 내의 참조 픽처(예컨대, 현재 픽처와 동일 POC의 픽처)에서 움직임을 참조하려는 블록의 위치를 특정한다. 다시 말하면, **global_disparity_vector_x**, **global_disparity_vector_y**는 현재 블록에 대한 GDV의 x 성분과 y 성분을 특정한다. 현재 블록에 대한 GDV는 참조 픽처 내에서 현재 블록의 위치에 대응하는 위치와 참조 블록의 위치 사이의 차이(혹은 현재 픽처에서 현재 블록의 위치와 참조 블록의 위치에 대응하는 위치 사이의 차이)를 나타낸다. **global_disparity_vector_x**, **global_disparity_vector_y**의 값, 즉 GDV의 값은 소정의 픽셀 단위, 예컨대, 4 픽셀, 8 픽셀 또는 16 픽셀 단위로 정의될 수도 있다.
- [0205] **num_gdv**는 동일한 GDV를 가지는 영역의 개수를 특정한다. 다시 말하면, **num_gdv**는 인터-뷰 예측에 사용되는 GDV의 개수를 특정한다. 예컨대, 도 11의 예와 같은 경우라면, **num_gdv**의 값으로서 3이 전송될 수 있다.
- [0206] 표 3을 참조하면, **num_gdv**의 값이 0보다 큰 경우에 **gdv**를 할당하기 위한 쿼드-트리가 설정된다. 쿼드-트리 구조에서 분할 여부를 지시하는 플래그(**split_flag**)가 전송된다.
- [0207] GDV는 쿼드-트리 구조에서 분할된 파티션의 파티셔닝 인덱스 *i*와 해당 깊이로 특정된다. 파티셔닝 인덱스는 네 개의 블록으로 분할되는 경우에 예컨대, 좌상측 블록에 0, 우상측 블록에 1, 우하측 블록에 2, 좌하측 블록에 3의 값이 할당될 수

있다.

- [0208] 해당 블록의 깊이가 GDV를 할당하기 위한 쿼드 트리 구조의 최대 깊이에 해당하거나 더 이상 분할되지 않는 경우에, 해당 블록에 대한 GDV값이 전송된다.
- [0209]
- [0210] 지금까지는 멀티-뷰 영상들 중 텍스처 영상에 관해 움직임 정보를 재활용하여 3D 비디오를 코딩/디코딩하는 것에 관하여 설명하였으나, 본 발명은 이에 한정되지 않으며, 깊이 영상에도 적용될 수 있다.
- [0211] 깊이 영상은 일련의 깊이 이미지를 포함하며, 각각의 깊이 이미지는 영상을 촬영하는 카메라의 광학적 중심(optical center)로부터 3D 장면(scene)의 포인트들까지의 거리를 그레이 레벨(gray level)로 나타낸다.
- [0212] 깊이 영상은 장면의 3D 구조를 포착(capture)하며 깊이 영상의 많은 부분은 영상 내 객체의 경계(boundary)를 나타내는 에지에 의해 분리되는 객체의 표면에 따라서 부드럽게 변하는 그레이 값(gray value)들로 구성된다.
- [0213] 도 13은 동일한 장면을 텍스처 영상과 깊이 영상으로 나타낸 일 예를 개략적으로 설명하는 도면이다.
- [0214] 도 13(a)는 인물이 등장하는 장면의 텍스처 영상을 나타낸다. 도 13(b)는 인물이 등장하는 장면의 깊이 영상을 나타내며, 장면 내 객체(object)의 표면에 따라서 그레이 레벨도 표현된다.
- [0215] 깊이 영상은 기하학적 묘사(geometric description)를 나타낸다. 다시 말하면, 대응하는 텍스처 영상(video) 내에서 장면 콘텐츠까지의 거리를 나타낸다. 따라서 텍스처 영상 내 움직임 콘텐츠(motion contents)의 시간적 변화(temporal evolution)은 대응하는 깊이 영상의 영상 내 움직임 콘텐츠의 시간적 변화와 높은 상관성을 가지게 된다. 예컨대, 객체(object) 경계들 주위의 움직임 벡터들은 텍스처 영상과 깊이 영상에서 일치하는 경향을 가지게 된다.
- [0216] 따라서, 깊이 영상과 텍스처 영상 간 움직임 정보의 유사성을 활용함으로써 깊이 영상의 코딩/디코딩을 더 효과적으로 수행할 수 있다.
- [0217] 상술한 바와 같이, 깊이 영상은 대응하는 텍스처 영상과 연관된 특성을 가지며, 특히 객체 경계 영역에서 높은 상관도를 자주 보인다. 따라서, 대응하는 텍스처 영상의 움직임 정보(예컨대, 움직임 벡터)를 깊이 영상의 움직임 정보(예컨대, 움직임 벡터)로서 재활용할 수 있다.
- [0218] 도 14는 현재 뷰의 깊이 영상을 텍스처 영상의 움직임 정보를 이용하여 예측하는 방법을 개략적으로 설명하는 도면이다.
- [0219] 도 14를 참조하면, 뷰 V1의 깊이 영상들(VD1) 중 현재 깊이 픽처(1490)의 현재 깊이 블록(1495)은 뷰 V1의 텍스트 영상들(VT1) 중 현재 깊이 픽처(1490)와 동일한 POC의 텍스트 영상(1400) 내에서 현재 깊이 블록(1495)에 대응하는 텍스처 블록(1415)의 움직임 정보를 이용하여 예측될 수 있다. 예컨대, 현재 깊이 블록(1495)에 대응하는 텍스처 블록(1415)의 움직임 벡터를 현재 깊이

블록(1495)의 움직임 벡터로 이용할 수 있다.

- [0220] 이때, 텍스처 블록(1415)의 움직임 정보는 주변 블록의 움직임 정보를 기반으로 예측될 수 있다. 예컨대, 도 10의 예와 같이, 참조 뷰(V0)의 텍스처 영상들(VT0) 중 현재 픽처(1400)와 동일한 POC를 갖는 참조 텍스처 픽처(1410)의 움직임 정보를 이용하여 현재 뷰(V1)의 현재 텍스처 블록(1415)에 대한 인터-뷰 예측을 수행할 수 있다.
- [0221] 현재 텍스처 블록(1415)이 이용할 수 있는 움직임 벡터 예측자 후보들은 공간적 인접 블록들인 우상단 블록(1420), 상측 블록(1425), 좌상단 블록(1430), 좌하단 블록(1435), 좌측 블록(1440)의 움직임 정보들(1455, 1460, 1465, 1470, 1475)과 현재 뷰(V1)의 다른 텍스처 픽처(1405) 내 Col 블록(1445)의 움직임 정보(1480)를 포함한다. 또한, 다른 뷰의 픽처들 중 현재 텍스처 픽처(1400)와 동일한 POC의 텍스처 픽처(1410)에서 현재 텍스처 블록(1415)에 대응하는 참조 블록(1450)의 움직임 정보(1485)를 움직임 벡터 예측자 후보로 이용할 수도 있다.
- [0222] 인코딩 장치는 상기 움직임 벡터 예측자 후보들 중 현재 텍스처 블록(1415)의 예측에 어떤 움직임 벡터 예측자를 이용할 것인지를 결정하고 결정된 움직임 벡터 예측자에 관한 정보를 디코딩 장치에 전송할 수 있다. 또한, 인코딩 장치는 현재 깊이 블록(1495)의 움직임 정보로서 현재 텍스처 블록(1415)의 움직임 정보(움직임 벡터)를 이용할 것인지를 결정하고, 이에 관한 정보를 전송할 수 있다.
- [0223] 디코딩 장치는 인코딩 장치로부터 전송된 정보를 기반으로 현재 텍스처 블록(1415)의 예측에 이용할 움직임 벡터 예측자를 유도할 수 있다. 또한, 디코딩 장치는 인코딩 장치로부터 전송된 정보를 기반으로 현재 깊이 블록(1495)의 움직임 정보(움직임 벡터)를 현재 텍스처 블록(1415)의 움직임 정보를 이용하여 유도할 수 있다.
- [0224] 도 14에서는 설명의 편의를 위해 현재 텍스처 블록과 현재 깊이 블록, 현재 텍스처 픽처와 현재 깊이 픽처 등의 용어를 사용하였으나, 이는 현재 텍스처 블록과 현재 깊이 블록이 동시에 인코딩 또는 디코딩 대상이 된다는 것을 의미하는 것이 아니며,
- [0225] 현재 텍스처 블록의 움직임 정보를 현재 깊이 블록이 이용할 수 있는 관계를 가진다는 것을 의미한다.
- [0226] 한편, 텍스처 픽처의 움직임 영상의 움직임 정보를 깊이 영상의 인코딩/디코딩에 이용하는 경우에 객체가 카메라 쪽으로 이동하거나 카메라로부터 멀어져서 텍스처 영상이 참조한 참조 픽처의 객체 깊이와 깊이 영상의 객체 깊이 사이에 차이가 발생할 수 있다. 깊이 값의 차이는 텍스처 영상으로부터 획득하는 움직임 벡터와 깊이 영상의 움직임 벡터 간에 큰 차이를 초래할 우려가 있다.
- [0227] 따라서, 대응하는 텍스처 영상의 움직임 정보를 깊이 영상의 움직임 정보로 사용하고, 깊이 값의 차이를 초래할 장면 내 객체의 움직임이 있는 경우에는

- 이를 보상하기 위해 선택적으로 DC 오프셋을 적용하는 방법을 고려할 수 있다.
- [0228] 텍스처 영상의 움직임 정보를 재활용하여 깊이 영상을 처리하면, 깊이 영상을 더 효과적으로 인코딩/디코딩 할 수 있다.
- [0229] 구체적으로, 움직임 벡터의 재활용에 있어서, 대응하는 텍스처 영상 블록(예컨대, 매크로 블록, CU, PU, TU 등)의 움직임 벡터를 깊이 영상의 예측에 어떻게 재사용할 것인지를 결정하기 위해 움직임 벡터 분할/통합 모드를 도입할 수 있다.
- [0230] 또한, 장면 내에서 객체가 카메라로부터 멀어지는 움직임 또는 카메라에 가까워지는 움직임으로 인해 깊이의 차이가 생기는 경우 이를 보상하기 위해 DC 오프셋 값을 검색 또는 결정하여 적용할 수도 있다.
- [0231] 적용되는 DC 오프셋 값은 인코딩 장치가 코스트를 고려하여 결정할 수 있다. 수식 3은 윌-외극 최적화를 기반으로 움직임 벡터 재사용을 결정하는 방법의 일 예를 나타낸 것이다.
- [0232] <수식 3>
- [0233] $J(mb_type, reuse_type | \lambda) = D + \lambda (R_{MV} + R_{mb_type} + R_{reuse_type} + R_{offset})$
- [0234] $D = \sum (I(X) - I_{ref}(X_p) - D_{offset})^2$
- [0235] 수식 3에서 J는 코스트를 나타내며, λ 는 라그랑제 승수(Lagrange multiplier)를 나타낸다.
- [0236] I는 깊이 영상의 픽셀 값에 대응하며, I_{ref} 는 참조 깊이 영상의 픽셀 값에 대응한다.
- [0237] D_{offset} 은 후술하는 reuse_type의 값이 1인 경우에 상기 D를 최소화하는 DC 오프셋 값이다. D_{offset} 값을 이용하여 텍스처 영상의 대응 블록으로부터 깊이 영상 블록에 대한 움직임 벡터(MV)를 유도할 수 있다. D 값의 합(summation)은 대상 블록 내 위치($X \in MB$)들에 대해 수행된다.
- [0238] X_p 는 깊이 영상 블록 내 대상 픽셀의 위치 X와 움직임 벡터(MV)의 합으로서, 움직임 벡터에 의해 지시되는 참조 픽처의 예측 샘플 위치를 나타낸다.
- [0239] R_{MV} 는 참조 픽처 인덱스(refIdx)와 관련된 움직임을 인코딩하기 위한 비트율(bit rate)로서, reuse_type의 값이 1인 경우에는 0의 값을 가진다.
- [0240] R_{offset} 은 reuse_type의 값이 1인 경우에만 제시되는 D_{offset} 을 인코딩하기 위한 비트율이다.
- [0241] R_{mb_type} 은 블록 타입의 정보(mb_type)을 인코딩하기 위한 비트율이며, R_{reuse_type} 은 움직임 벡터 재활용 타입(MV reuse_type)을 인코딩하기 위한 비트율이다.
- [0242] mb_type은 해당 블록(예컨대, 매크로 블록, CU, PU 혹은 TU 등)의 크기 및/또는 예측 모드(예컨대, 스킵 모드, 머지 모드, MVP 이용 모드 등)를 지시한다.
- [0243] 수식 3에서는 해당 블록이 매크로 블록인 경우를 예로서 설명하였으나, 본 발명은 이에 한정되지 않고, 수식 3의 내용은 해당 블록이 CU, PU 혹은 TU 등인 경우에도 동일하게 적용될 수 있다.
- [0244] 인코딩 장치는 수식 3을 기반으로 DC 오프셋 D_{offset} 을 포함하여 필요한 정보를

결정할 수 있다. 인코딩 장치는 DC 오프셋 정보를 이용하여 깊이 영상이 텍스처 영상의 움직임 정보를 재활용하는 경우에, 깊이 영상이 활용하는 텍스처 영상에 대한 보상을 수행할 수 있다. 또한, 인코딩 장치는 DC 오프셋을 포함한 정보를 디코딩 장치로 전송할 수 있다.

- [0245] 디코딩 장치는 수신한 정보를 기반으로 깊이 영상에 대한 디코딩을 수행할 수 있다. 예컨대, 디코딩 장치는 인코딩 장치로부터 수신한 정보가 텍스처 영상의 움직임 정보를 기반으로 깊이 영상을 디코딩하는 것을 지시하는 경우, 대응하는 텍스처 영상의 움직임 정보(예컨대, 움직임 벡터)와 DC 오프셋 값을 이용하여 현재 깊이 블록을 디코딩 할 수 있다.
- [0246] 이때, 상술한 바와 같이 인코딩 장치는 대응하는 텍스처 블록의 움직임 정보를 이용할 것인지 또한 텍스처 블록의 움직임 정보를 이용하는 경우에는 어떻게 이용할 것인지를 지시하는 정보를 디코딩 장치에 전송할 수 있다.
- [0247] 표 4는 텍스처 영상의 움직임 정보를 깊이 영상의 디코딩에 이용하기 위한 정보를 시그널링하는 신택스의 일 예를 나타낸 것이다.

[0248] <표 4>

| | |
|--------|-------------------------------------|
| [0249] | coding_unit(x0, y0, log2CbSize) { |
| | ... |
| | reuse_type |
| | ... |
| | If (reuse_type){ |
| | ... |
| | DC_offset |
| | ... |

- [0250] 표 4는 현재 깊이 블록이 CU인 경우를 예로서 설명한다. 표 4를 참조하면, **reuse_type**는 현재 깊이 블록의 디코딩에 텍스처 영상의 움직임 정보를 이용할 것인지를 지시한다. **reuse_type**가 텍스처 영상을 이용하는 것을 지시하는 경우에는, 텍스처 영상과 깊이 영상 사이에서 객체의 움직임(깊이의 차이를 만드는 움직임, 예컨대, 카메라 방향의 움직임)을 보상하는 DC 오프셋 값이 전송된다.
- [0251] 구체적으로, **reuse_type**의 값이 0인 경우에는 텍스처 영상의 움직임 정보를 이용하지 않고 현재 깊이 블록을 디코딩한다. 따라서, 현재 깊이 블록은 일반적인 예측 모드를 이용하여 예측될 수 있다.
- [0252] 예컨대, 현재 깊이 블록은 인트라 예측이 적용되는 경우에 현재 깊이 픽처 내 인접하는 픽셀들을 이용하여 인트라 예측될 수 있다. 또한, 인터 예측이 적용되는 경우에, 현재 깊이 블록은 스킵 모드, 머지 모드 또는 MVP 모드를 이용하여 참조 깊이 영상을 기반으로 인터 예측될 수 있다. 깊이 영상의 인터 예측 및 인트라 예측은 텍스처 영상의 인터 예측 및 인트라 예측과 동일하게 수행될 수 있다.

- [0253] **reuse_type**의 값이 0인 경우에, 텍스처 영상과 깊이 영상 사이에서 객체의 움직임을 보상하는 DC 오프셋 값은 전송되지 않는다.
- [0254] **reuse_type**의 값이 1인 경우에는 텍스처 영상의 움직임 정보를 이용하여 현재 깊이 블록을 디코딩한다. 따라서, 현재 깊이 블록에 대한 예측은 대응하는 텍스처 블록의 움직임 정보를 기반으로 수행될 수 있다. 이 경우에, 표 4에서 예시한 바와 같이, 텍스처 영상과 깊이 영상 사이에서 객체의 움직임을 보상하는 DC 오프셋 값이 디코딩 장치로 전송될 수 있다. DC 오프셋 값은 현재 깊이 블록의 복원된 픽셀 값에 추가되어 텍스처 영상과 깊이 영상 사이에서 객체의 움직임을 보상할 수 있다.
- [0255] **reuse_type**의 값이 1인 경우에 현재 깊이 영상의 움직임 정보(예컨대, 움직임 벡터)는 현재 깊이 블록에 대응하는 텍스처 블록의 크기를 고려하여 텍스처 영상의 움직임 정보(예컨대, 움직임 벡터)로부터 유도될 수 있다.
- [0256] 표 4에서는 **reuse_type**의 값이 1인 경우에 **DC_offset** 값이 전송되는 것으로 설명하였으나, 상술한 바와 같이 **DC_offset** 값은 선택적으로 적용될 수도 있다. 예컨대, **reuse_type** 값이 1인 경우로서 텍스처 영상의 움직임 정보를 이용하는 경우에도 참조 대상이 되는 텍스처 영상과 깊이 영상 간의 깊이 값 차이가 없거나 적은 경우에는 **DC_offset** 값을 전송하지 않거나 **DC_offset** 값을 0으로 설정할 수 있다.
- [0257] 도 15는 대응하는 텍스처 블록의 움직임 벡터로부터 현재 깊이 블록의 움직임 벡터를 유도하는 방법을 개략적으로 설명하는 도면이다.
- [0258] 현재 깊이 블록의 사이즈가 대응하는 텍스처 블록(예컨대, 인코딩 장치로부터 지시되는 후보 **mb_type**)의 사이즈보다 크거나 같은 경우에, 현재 깊이 블록의 움직임 벡터는 대응하는 텍스처 블록의 움직임 벡터와 대응하는 텍스처 블록의 주변 블록들의 움직임 벡터를 기반으로 유도될 수 있다. 예컨대, 현재 깊이 블록의 움직임 벡터는 대응하는 텍스처 블록의 움직임 벡터와 대응하는 텍스처 블록의 주변 블록들의 움직임 벡터의 평균으로 유도될 수 있다. 또는 현재 깊이 블록의 움직임 벡터는 대응하는 텍스처 블록의 움직임 벡터와 대응하는 텍스처 블록의 주변 블록들의 움직임 벡터의 미디언으로 유도될 수도 있다. 이때, 대응하는 텍스처 블록은 매크로 블록, CU, PU, TU 등일 수 있다.
- [0259] 도 15(a)는 현재 깊이 블록의 사이즈가 대응하는 텍스처 블록(예컨대, 인코딩 장치로부터 지시되는 후보 **mb_type**)의 사이즈보다 크거나 같은 경우에 현재 깊이 블록의 움직임 벡터를 유도하는 방법을 개략적으로 설명하는 도면이다.
- [0260] 도 15(a)를 참조하면, 현재 깊이 블록(1500)이 대응하는 텍스처 블록(1510)보다 크거나 같은 경우에, 현재 깊이 블록(1500)의 움직임 벡터는 대응하는 텍스처 블록(1510)의 움직임 벡터와 주변 텍스처 블록들(1520, 1530, 1540, 1550, 1560)의 움직임 벡터를 기반으로 결정될 수 있다. 예컨대, 현재 깊이 블록(1500)의 움직임 벡터는 대응하는 텍스처 블록(1510)의 움직임 벡터와 주변 텍스처 블록들(1520, 1530, 1540, 1550, 1560)의 움직임 벡터들의 평균으로서 유도될 수 있다. 혹은

현재 깊이 블록(1500)의 움직임 벡터는 대응하는 텍스처 블록(1510)의 움직임 벡터와 주변 텍스처 블록들(1520, 1530, 1540, 1550, 1560)의 움직임 벡터들의 미디언으로서 유도될 수도 있다.

- [0261] 또한, 현재 깊이 블록(1500)이 대응하는 텍스처 블록(1510)보다 크거나 같은 경우에는, 주변 텍스처 블록들의 움직임 벡터들 중에서 일부만을 이용하여 현재 깊이 블록(1500)의 움직임 벡터를 유도할 수도 있다. 예컨대, 현재 깊이 블록(1500)의 움직임 벡터는 대응하는 텍스처 블록(1510)의 움직임 벡터, 좌측 텍스처 블록(1530)의 움직임 벡터 및 상측 텍스처 블록(1550)의 움직임 벡터의 미디언 또는 평균으로 유도될 수 있다.
- [0262] 현재 깊이 블록의 사이즈가 대응하는 텍스처 블록(예컨대, 인코딩 장치로부터 지시되는 후보 `mb_type`)의 사이즈보다 작은 경우에, 현재 깊이 블록의 움직임 벡터는 대응하는 텍스처 블록 내 서브-블록들의 움직임 벡터들을 기반으로 유도될 수 있다. 예컨대, 현재 깊이 블록의 움직임 벡터는 대응하는 텍스처 블록 내 서브-블록들에 대한 움직임 벡터들의 평균으로 유도될 수 있다. 혹은 현재 깊이 블록의 움직임 벡터는 대응하는 텍스처 블록 내 서브-블록들에 대한 움직임 벡터들의 미디언으로 유도될 수도 있다. 이때, 대응하는 텍스처 블록 및/또는 대응하는 텍스처 블록 내 서브-블록은 매크로 블록, CU, PU, TU 등일 수 있다.
- [0263] 도 15(b)는 현재 깊이 블록의 사이즈가 대응하는 텍스처 블록(예컨대, 인코딩 장치로부터 지시되는 후보 `mb_type`)의 사이즈보다 작은 경우에 현재 깊이 블록의 움직임 벡터를 유도하는 방법을 개략적으로 설명하는 도면이다.
- [0264] 도 15(b)를 참조하면, 현재 깊이 블록(1570)이 대응하는 텍스처 블록(1580)보다 크거나 같은 경우에, 현재 깊이 블록(1570)의 움직임 벡터는 대응하는 텍스처 블록(1580) 내 서브 블록들(`sb0, sb1, ..., sb_i, ..., sb_n`, `n`은 서브 블록의 개수)의 움직임 벡터를 기반으로 결정될 수 있다. 예컨대, 현재 깊이 블록(1570)의 움직임 벡터는 대응하는 텍스처 블록(1580) 내 서브 블록들(`sb0, sb1, ..., sb_i, ..., sb_n`)에 대한 움직임 벡터들의 평균으로 유도될 수 있다. 또한, 현재 깊이 블록(1570)의 움직임 벡터는 대응하는 텍스처 블록(1580) 내 서브 블록들(`sb0, sb1, ..., sb_i, ..., sb_n`)에 대한 움직임 벡터들의 미디언으로 유도될 수도 있다.
- [0265] 도 16은 본 발명에 따라서 움직임 정보를 재활용하여 3D 영상을 인코딩 하는 방법을 개략적으로 설명하는 순서도이다.
- [0266] 도 16을 참조하면, 인코딩 장치는 현재 뷰에서 텍스처 영상의 움직임 정보를 결정한다(S1610). 인코딩 장치는 텍스처 영상의 현재 디코딩 대상 블록(혹은 예측 대상 블록)의 움직임 정보를 주변 블록의 움직임 정보를 이용하여 결정할 수 있다. 이때, 주변 블록은 현재 블록의 공간적 주변 블록과 시간적 주변 블록(동일 뷰 내 다른 픽처의 대응 블록)뿐만 아니라, 현재 픽처와 동일한 POC의 다른 뷰 내 픽처에서 현재 블록에 대응하는 블록을 포함한다. 인코딩 장치는 주변 블록의 움직임 정보를 기반으로 결정한 현재 블록의 움직임 정보를 이용하여 현재 블록에 대한 예측을 수행하고, 현재 블록을 인코딩 할 수 있다.

- 예측과 인코딩에 관한 내용은 도 2에 관한 설명을 포함하여 상술한 바와 같다.
- [0267] 인코딩 장치는 깊이 영상의 움직임 정보를 결정할 수 있다(S1620). 인코딩 장치는 깊이 영상의 움직임 정보를 텍스처 영상의 움직임 정보를 기반으로 결정할 수도 있고, 다른 깊이 영상의 움직임 정보(참조 깊이 영상 내 참조 블록의 움직임 정보)를 기반으로 결정할 수도 있다.
- [0268] 인코딩 장치는 깊이 영상에 대한 오프셋을 결정할 수 있다(S1630). 깊이 영상이 텍스처 영상의 움직임 정보를 이용하는 경우에, 텍스처 영상이 참조하는 블록에서의 객체와 현재 깊이 블록에서의 객체 사이에 깊이의 차이가 발생할 수 있다. 인코딩 장치는 깊이의 차이를 고려한 DC(Direct Current) 오프셋을 통해 복원된 픽셀값을 보상할 수 있다.
- [0269] 인코딩 장치는 움직임 정보를 전송할 수 있다(S1640). 인코딩 장치는 텍스처 영상의 움직임 정보와 깊이 영상의 움직임 정보를 인코딩하여 비트스트림으로 디코딩 장치에 전송할 수 있다. 인코딩 과정에서 수행되는 변환, 양자화, 재정렬 및 엔트로피 인코딩은 도 2에 관한 설명을 포함하여 앞서 설명한 바와 같다.
- [0270] 전송되는 정보는 표 1 내지 표 4에서 설명한 정보들을 포함할 수 있다.
- [0271] 인코딩 장치는 인터-뷰 예측을 수행하는 경우에 다른 뷰에 속하는 참조 픽처의 움직임 정보를 이용할 것인지를 지시하는 정보(예컨대, `inter_view_motion_info_flag`), 다른 뷰에 속하는 참조 픽처의 움직임 정보를 이용하는 경우에 어느 방향의 뷰를 참조할 것인지를 지시하는 정보(예컨대, `inter_view_motion_reuse_dir`), 참조할 블록의 위치를 지시하는 GDV값에 관한 정보(예컨대, `global_disparity_vector_x`, `global_disparity_vector_y`)를 전송할 수 있다. 현재 픽처의 영역별로 GDV가 설정되는 경우에는 현재 픽처에서 정의된 GDV의 개수를 지시하는 정보(예컨대, `num_gdv`)와 GDV가 정의되는 영역에 관한 분할 정보(`split_flag`) 및 현재 영역에서의 GDV 정보(`gdv_id`)가 함께 전송될 수도 있다. 이때, GDV가 정의되는 영역에 관한 분할의 최대 깊이(예컨대, `max_gdv_qtree_depth`)와 영역의 최대 크기에 관한 정보(예컨대, `max_gdv_unit_size_log_scale`)도 함께 또는 별도로 전송될 수 있다.
- [0272] 또한, 인코딩 장치는 깊이 영상의 움직임 정보를 텍스처 영상의 움직임 정보를 기반으로 유도할 것인지를 지시하는 정보(예컨대, `reuse_type`), 깊이 영상의 움직임 정보를 텍스처 영상의 움직임 정보를 기반으로 유도하는 경우에 복원된 픽셀 값을 보상하는 오프셋 값에 관한 정보(예컨대, `DC_offset`)를 전송할 수도 있다.
- [0273] 각 정보가 전송되는 위치는 표 1 내지 표 4에서 설명한 바와 같다.
- [0274] 도 16에서는 설명의 편의를 위해 각 단계가 인코딩 장치에서 수행되는 것으로 설명하였으나, 본 발명은 이에 한정되지 않는다. 예컨대, 움직임 정보와 오프셋의 결정은 인코딩 장치의 예측부에서 수행될 수 있다.
- [0275] 도 17은 본 발명에 따라서 인코딩 장치에서 텍스처 영상의 움직임 정보를 결정하고 이를 이용하여 예측을 수행하는 방법의 일 예를 개략적으로 설명하는

순서도이다.

- [0276] 도 17을 참조하면, 인코딩 장치는 현재 텍스트 블록의 예측에 이용할 참조 블록을 결정한다(S1710). 현재 텍스트 블록의 예측은 동일 뷰 내에서의 인터 예측과 서로 다른 뷰 사이의 인터-뷰 예측을 포함한다. 인코딩 장치는 현재 텍스트 블록의 주변 블록들 중에서 참조할 블록을 결정할 수 있다. 현재 텍스트 블록의 주변 블록으로서 해당 블록의 움직임 정보를 이용할 수 있는 블록들은 도 9 및 도 10에서 설명한 바와 같다.
- [0277] 인코딩 장치는 참조 블록의 움직임 정보를 움직임 정보 예측자로서 결정할 수 있다(S1720). 인코딩 장치는 참조 블록 후보들 중 선택된 참조 블록의 움직임 정보를 움직임 정보 예측자로 이용할 수 있다. 움직임 정보 예측자는 해당 블록에 대한 움직임 벡터에 관한 정보만을 포함할 수도 있고, 움직임 벡터와 참조 픽처 인덱스에 관한 정보를 포함할 수도 있다. 움직임 정보 예측자가 해당 블록에 대한 움직임 벡터에 관한 정보만을 포함하는 경우에, 인코딩 장치는 예측에 필요한 참조 픽처의 정보를 디코딩 장치에 별도로 전송할 수도 있다.
- [0278] 인코딩 장치는 움직임 정보 예측자를 기반으로 현재 텍스트 블록에 대한 레지듀얼을 산출할 수 있다(S1730). 인터 예측 또는 인터-뷰 예측에서 스킵 모드 또는 머지 모드가 적용되는 경우에, 움직임 정보 예측자는 움직임 벡터뿐만 아니라 참조 픽처 인덱스를 포함할 수 있다. 이 경우, 인코딩 장치는 움직임 정보 예측자가 지시하는 블록의 픽셀 값을 기반으로 예측 블록을 생성하고, 현재 텍스트 블록에 대한 레지듀얼을 산출할 수 있다. 스킵 모드의 경우에 디코딩 장치는 레지듀얼을 생성하지 않거나 전송하지 않을 수 있다. 인터 예측 또는 인터-뷰 예측에서 MVP를 이용하는 경우에는 예측에 사용될 움직임 정보 예측자는 움직임 벡터 예측자, 즉 MVP 정보를 포함하며 참조 픽처의 인덱스는 별도로 결정될 수 있다. 이 경우, 인코딩 장치는 움직임 정보 예측자와 참조 픽처 인덱스를 기반으로 현재 텍스트 블록에 대한 예측 블록을 생성하고, 현재 텍스트 블록에 대한 레지듀얼을 산출할 수 있다.
- [0279] 도 18은 본 발명에 따라서 인코딩 장치에서 깊이 영상의 움직임 정보를 결정하고 이를 이용하여 예측을 수행하는 방법으로서, 텍스트 영상의 움직임 정보를 이용하여 깊이 영상의 예측을 수행하는 방법의 일 예를 개략적으로 설명하는 순서도이다.
- [0280] 도 18을 참조하면, 텍스트 영상의 움직임 정보를 이용하는 경우에, 인코딩 장치는 현재 깊이 블록에 대응하는 텍스트 영상의 블록을 결정한다(S1810). 인코딩 장치는 현재 깊이 블록의 움직임 정보를 텍스트 영상의 움직임 정보를 이용하여 결정할 수 있다. 이 경우에, 인코딩 장치는 현재 깊이 블록에서 참조할 텍스트 영상의 블록을 결정한다. 현재 깊이 영상에 대응하는 텍스트 영상의 블록은 동일 뷰 내의 텍스트 영상으로서 현재 깊이 영상과 동일한 POC를 가지는 텍스트 영상 내 블록일 수 있다. 텍스트 영상 내 대응 블록은 깊이 영상과 동일 뷰 및 동일 POC의 텍스트 영상에서 현재 깊이 블록과 동일한 위치의 블록일 수

있다. 또한, 텍스처 영상 내 대응 블록은 깊이 영상과 동일 뷰 및 동일 POC의 텍스처 영상에서 현재 깊이 블록을 특징하는 픽셀에 대응하는 픽셀을 포함하는 블록일 수도 있다.

- [0281] 인코딩 장치는 텍스처 영상의 움직임 정보를 기반으로 현재 깊이 블록의 움직임 정보를 산출할 수 있다(S1820). 현재 깊이 블록의 움직임 정보는 대응하는 텍스처 블록의 움직임 정보를 기반으로 결정될 수 있다. 또한, 현재 깊이 블록의 움직임 정보는 대응하는 텍스처 블록의 움직임 정보 및 대응하는 텍스처 블록의 주변 블록의 움직임 정보를 이용하여 결정될 수 있다. 이때, 텍스처 영상의 움직임 정보를 기반으로 현재 깊이 블록의 움직임 정보를 결정하는 방법은 도 15에서 설명한 바와 동일하다.
- [0282] 인코딩 장치는 현재 깊이 블록에 대한 오프셋을 결정할 수 있다(S1830). 상술한 바와 같이, 깊이 영상이 텍스처 영상의 움직임 정보를 이용하는 경우에, 텍스처 영상이 참조하는 블록에서의 객체와 현재 깊이 블록에서의 객체 사이에 깊이의 차이가 발생할 수 있다. 인코딩 장치는 깊이의 차이를 고려한 DC(Direct Current) 오프셋 값을 결정할 수 있다. 깊이 차이에 의한 복원 픽셀 값의 오류는 DC 오프셋을 통해 보상될 수 있다.
- [0283] 인코딩 장치는 현재 깊이 블록을 복원한다(S1840). 인코딩 장치는 텍스처 영상의 움직임 정보를 이용하여 현재 깊이 블록을 복원할 수 있다. 스킵 모드 또는 머지 모드가 적용되는 경우에, 텍스처 영상의 움직임 정보로부터 얻은 현재 깊이 블록에 대한 움직임 정보는 움직임 벡터뿐만 아니라 참조 픽처(참조 깊이 영상) 인덱스를 포함할 수 있다. 인코딩 장치는 텍스처 영상의 움직임 정보로부터 얻은 현재 깊이 블록에 대한 움직임 정보가 지시하는 깊이 영상 블록의 픽셀 값을 기반으로 예측 블록을 생성하고, 현재 깊이 블록에 대한 레지듀얼을 산출할 수 있다. 인터 예측 또는 인터-뷰 예측에서 MVP를 이용하는 경우에는 움직임 정보는 움직임 벡터 예측자, 즉 MVP 정보를 포함하며 참조 픽처의 인덱스는 별도로 결정될 수 있다. 이 경우, 인코딩 장치는 움직임 정보 예측자와 참조 픽처(참조 깊이 영상) 인덱스를 기반으로 현재 깊이 블록에 대한 예측 블록을 생성하고, 현재 깊이 블록에 대한 레지듀얼을 산출할 수 있다.
- [0284] 인코딩 장치는 레지듀얼과 예측 블록을 기반으로 복원한 깊이 블록(깊이 영상)을 이후에 인코딩되는 깊이 블록의 예측에 이용할 수 있다. 또한, 인코딩 장치는 현재 깊이 블록의 레지듀얼에 관한 정보 및 예측에 관한 정보를 인코딩하여 디코딩 장치에 전송할 수도 있다.
- [0285] 한편, 도 16과 도 18에서는 텍스처 영상의 움직임 정보를 이용하여 현재 깊이 블록을 복원하는 것을 설명하였으나, 현재 깊이 블록은 표 4와 관련하여 설명한 바와 같이, 다른 깊이 영상의 움직임 정보를 이용하여 인코딩될 수도 있다. 인코딩 장치는 표 4에서와 같이 텍스처 영상의 움직임 정보를 이용할 것인지를 지시하는 정보를 디코딩 장치에 전송할 수 있다.
- [0286] 도 19는 본 발명에 따라서 움직임 정보를 재활용하여 3D 영상을 디코딩 하는

방법을 개략적으로 설명하는 순서도이다.

- [0287] 도 19를 참조하면, 디코딩 장치는 인코딩 장치로부터 정보(비트스트림)을 수신한다(S1910). 수신한 정보(비트스트림)는 레지듀얼을 포함한 영상 정보 외에도 디코딩에 필요한 파라미터 등을 포함한다. 디코딩 장치는 엔트로피 디코딩을 통해 필요한 정보를 획득할 수 있다.
- [0288] 수신한 정보는 표 1 내지 표 4에서 설명한 정보들을 포함할 수 있다. 예컨대, 인터-뷰 예측을 수행하는 경우에 다른 뷰에 속하는 참조 픽처의 움직임 정보를 이용할 것인지를 지시하는 정보(예컨대, `inter_view_motion_info_flag`), 다른 뷰에 속하는 참조 픽처의 움직임 정보를 이용하는 경우에 어느 방향의 뷰를 참조할 것인지를 지시하는 정보(예컨대, `inter_view_motion_reuse_dir`), 참조할 블록의 위치를 지시하는 GDV 값에 관한 정보(예컨대, `global_disparity_vector_x`, `global_disparity_vector_y`)가 포함될 수 있다. 현재 픽처의 영역별로 GDV가 설정되는 경우에는 현재 픽처에서 정의된 GDV의 개수를 지시하는 정보(예컨대, `num_gdv`)와 GDV가 정의되는 영역에 관한 분할 정보(`split_flag`) 및 현재 영역에서의 GDV 정보(`gdv_id`)가 포함될 수도 있다. 이때, GDV가 정의되는 영역에 관한 분할의 최대 깊이(예컨대, `max_gdv_qtree_depth`)와 영역의 최대 크기에 관한 정보(예컨대, `max_gdv_unit_size_log_scale`)도 함께 또는 별도로 수신될 수 있다.
- [0289] 또한, 깊이 영상의 움직임 정보를 텍스처 영상의 움직임 정보를 기반으로 유도할 것인지를 지시하는 정보(예컨대, `reuse_type`), 깊이 영상의 움직임 정보를 텍스처 영상의 움직임 정보를 기반으로 유도하는 경우에 복원된 픽셀 값을 보상하는 오프셋 값에 관한 정보(예컨대, `DC_offset`) 등이 수신한 정보에 포함될 수 있다.
- [0290] 각 정보가 전송되는(수신되는) 비트 스트림 상의 위치는 표 1 내지 표 4에서 설명한 바와 같다.
- [0291] 디코딩 장치는 수신한 정보를 기반으로 텍스처 영상의 움직임 정보를 유도할 수 있다(S1920). 디코딩 장치는 수신한 정보가 인터-뷰 예측에 있어서 움직임 정보의 재활용을 지시하는 경우에는 다른 뷰의 대응 블록에 대한 움직임 정보를 기반으로 현재 블록의 움직임 정보를 유도할 수 있다. 예컨대, 인터-뷰 예측에 있어서 머지 모드 또는 스킵 모드가 적용되는 경우에, 디코딩 장치는 다른 뷰의 대응 블록에 대한 움직임 정보를 현재 텍스처 블록의 움직임 정보로 이용할 수 있다. 이때, 움직임 정보는 움직임 벡터와 참조 픽처 인덱스를 포함한다. 인터-뷰 예측에 있어서 MVP를 이용한 경우에, 디코딩 장치는 다른 뷰의 대응 블록에 대한 움직임 벡터를 예측자로 하고 움직임 벡터 차이값을 더하여 현재 블록의 움직임 벡터를 유도할 수 있다. 이 경우, 참조 픽처 인덱스와 움직임 벡터 차이값은 인코딩 장치로부터 전송될 수 있다.
- [0292] 디코딩 장치는 깊이 영상의 움직임 정보를 유도할 수 있다(S1930). 디코딩 장치는 텍스처 영상의 움직임 정보를 재활용하여 현재 깊이 블록의 움직임

정보를 유도할 수도 있고, 다른 텍스처 영상의 움직임 정보를 기반으로 현재 깊이 블록의 움직임 정보를 유도할 수도 있다. 텍스처 영상의 움직임 정보를 재활용할 것인지는 인코딩 장치로부터 전송된 정보에 의해 지시될 수 있다.

[0293] 디코딩 장치는 유도한 움직임 정보를 이용하여 텍스처 영상(텍스처 블록)을 복원할 수 있다. 예컨대, 인터 예측 또는 인터-뷰 예측에서 스킵 모드 또는 머지 모드가 적용되는 경우에, 유도된 움직임 정보는 움직임 벡터뿐만 아니라 참조 픽처 인덱스를 포함할 수 있다. 디코딩 장치는 움직임 정보가 지시하는 블록의 픽셀 값을 기반으로 예측 블록을 생성할 수 있다. 디코딩 장치는 현재 텍스처 블록에 대한 레지듀얼을 예측 블록과 합하여 현재 텍스처 블록을 복원할 수 있다. 스킵 모드의 경우에는 레지듀얼이 전송되지 않으며, 디코딩 장치는 예측 블록을 복원 블록으로 할 수 있다. 인터 예측 또는 인터-뷰 예측에서 MVP를 이용하는 경우에 참조 픽처의 인덱스는 별도로 전송될 수 있다. 디코딩 장치는 움직임 정보와 참조 픽처 인덱스를 기반으로 현재 텍스처 블록에 대한 예측 블록을 생성하고, 레지듀얼과의 합으로 현재 텍스처 블록을 복원할 수 있다.

[0294] 디코딩 장치는 유도한 움직임 정보를 이용하여 깊이 영상(깊이 블록)을 복원할 수 있다. 스킵 모드 또는 머지 모드가 적용되는 경우에, 움직임 정보는 움직임 벡터뿐만 아니라 참조 픽처(참조 깊이 영상) 인덱스를 포함할 수 있다. 디코딩 장치는 움직임 정보가 지시하는 블록의 픽셀 값을 기반으로 현재 깊이 블록에 대한 예측 블록을 생성하고, 레지듀얼과 합하여 복원 블록을 생성할 수 있다. 스킵 모드의 경우에, 디코딩 장치는 예측 블록을 복원 블록으로 할 수 있다. 인터 예측 또는 인터-뷰 예측에서 MVP를 이용하는 경우에 참조 픽처(참조 깊이 영상)의 인덱스는 별도로 전송될 수 있다. 디코딩 장치는 움직임 정보와 참조 픽처(참조 깊이 영상) 인덱스를 기반으로 현재 깊이 블록에 대한 예측 블록을 생성하고, 레지듀얼과 더하여 복원 블록을 생성할 수 있다.

[0295] 여기서는 설명의 편의를 위해 디코딩 장치가 도 19의 각 단계를 수신하는 것으로 설명하였으나, 도 19의 단계들 중 일부 또는 전부는 디코딩 장치 내 소정의 유닛에서 수행될 수도 있다. 예컨대, S1920 및 S1930은 디코딩 장치 내 예측부에서 수행될 수 있다.

[0296] 도 20은 본 발명에 따라서 디코딩 장치가 텍스처 영상의 움직임 정보를 유도하는 방법의 일 예를 개략적으로 설명하는 도면이다.

[0297] 도 20을 참조하면, 디코딩 장치는 인코딩 장치로부터 수신한 정보로부터 예측에 필요한 정보를 유도한다(S2010). 예측에 필요한 정보는 비트스트림을 통해 인코딩 장치로부터 수신될 수 있다. 수신한 정보는 표 1 내지 표 3에서 설명한 정보들을 포함할 수 있다. 예컨대, 인터-뷰 예측을 수행하는 경우에 다른 뷰에 속하는 참조 픽처의 움직임 정보를 이용할 것인지를 지시하는 정보(예컨대, `inter_view_motion_info_flag`), 다른 뷰에 속하는 참조 픽처의 움직임 정보를 이용하는 경우에 어느 방향의 뷰를 참조할 것인지를 지시하는 정보(참조 뷰 방향 정보), 참조할 블록의 위치를 지시하는 GDV값에 관한 정보가 수신한 정보에

포함될 수 있다. 현재 픽처의 영역별로 GDV가 지시되는 경우에는 현재 픽처에서 정의된 GDV의 개수를 지시하는 정보와 GDV가 정의되는 영역에 관한 분할 정보가 수신한 정보에 포함될 수 있다.

- [0298] 디코딩 장치는 텍스처 영상의 움직임 정보를 유도할 수 있다(S2020). 인터-뷰 예측을 수행하는 경우에, 디코딩 장치는 인코딩 장치로부터 전송된 참조 뷰 방향 정보(예컨대, `inter_view_motion_reuse_dir`)가 지시하는 방향의 뷰들 중 현재 뷰가 참조하는 뷰의 동일 POC 픽처를 참조 픽처로 이용할 수 있다. 이때, 참조 픽처 내의 참조 블록은 GDV 정보(예컨대, `global_disparity_vector_x`, `global_disparity_vector_y` 또는 `gdv_id`)에 의해 지시될 수 있다.
- [0299] 스킵 모드 또는 머지 모드가 적용되는 경우에, 참조 블록의 움직임 정보를 현재 텍스처 블록의 움직임 정보로 이용할 수 있다. 이 경우, 움직임 정보는 움직임 벡터와 참조 픽처 인덱스를 포함한다.
- [0300] MVP를 이용하는 경우에, 디코딩 장치는 참조 블록의 움직임 벡터를 현재 텍스처 블록의 움직임 벡터 예측자로 이용하고, 인코딩 장치로부터 수신한 움직임 벡터 차이를 더하여 현재 텍스처 블록의 움직임 벡터를 생성할 수 있다. 이 경우, 참조 픽처 인덱스는 인코딩 장치로부터 전송될 수 있다.
- [0301] 도 21은 본 발명에 따라서 디코딩 장치가 깊이 영상의 움직임 정보를 유도하는 방법의 일 예를 개략적으로 설명하는 도면이다.
- [0302] 도 21을 참조하면, 디코딩 장치는 현재 깊이 블록에 대응하는 텍스처 블록의 움직임 정보를 유도한다(S2110). 텍스처 블록의 움직임 정보를 유도하는 방법은 도 20을 포함하여 앞선 부분에서 설명한 바와 같다.
- [0303] 디코딩 장치는 현재 깊이 블록의 움직임 정보를 유도한다(S2120). 디코딩 장치는 인코딩 장치로부터 수신한 정보가 텍스처 영상의 움직임 정보를 이용할 것을 지시하는 경우(예컨대, `reuse_type=1`인 경우)에 대응하는 텍스처 블록의 움직임 정보를 기반으로 현재 깊이 영상의 움직임 정보를 유도한다. 텍스처 블록의 움직임 정보를 기반으로 현재 깊이 블록의 움직임 정보를 유도하는 방법은 도 15에서 설명한 바와 같다.
- [0304] 인코딩 장치로부터 수신한 정보가 텍스처 영상의 움직임 정보를 이용하지 않을 것을 지시하는 경우(예컨대, `reuse_type=0`)에, 디코딩 장치는 동일한 뷰의 다른 깊이 영상을 참조하여 현재 깊이 블록의 움직임 정보를 유도할 수 있다.
- [0305] 디코딩 장치는 유도한 움직임 정보를 이용하여 깊이 영상(깊이 블록)을 복원할 수 있다. 예컨대, 인터 예측 또는 인터-뷰 예측에서 스킵 모드 또는 머지 모드가 적용되는 경우에, 유도된 움직임 정보는 움직임 벡터뿐만 아니라 참조 픽처(참조 깊이 영상) 인덱스를 포함할 수 있다. 디코딩 장치는 움직임 정보가 지시하는 블록의 픽셀 값을 기반으로 예측 블록을 생성할 수 있다. 디코딩 장치는 현재 깊이 블록에 대한 레지듀얼을 예측 블록과 합하여 현재 텍스처 블록을 복원할 수 있다. 스킵 모드의 경우에는 레지듀얼이 전송되지 않으며, 디코딩 장치는 예측 블록을 복원 블록으로 할 수 있다. 인터 예측 또는 인터-뷰 예측에서 MVP를

이용하는 경우에 참조 픽처의 인덱스는 별도로 전송될 수 있다. 디코딩 장치는 움직임 정보와 참조 픽처 인덱스를 기반으로 현재 깊이 블록에 대한 예측 블록을 생성하고, 레지듀얼과의 합으로 현재 깊이 블록을 복원할 수 있다.

- [0306] 한편, 디코딩 장치는 텍스처 영상의 움직임 정보를 이용하여 현재 깊이 블록을 복원하는 경우(예컨대, reuse_type = 1)로서 DC_offset 값이 인코딩 장치로부터 전송된 경우에는, 오프셋 값을 적용하여 현재 깊이 블록을 복원할 수 있다.
- [0307] 도 16 내지 도 21에서는 앞서 설명한 본 발명의 내용에 대한 이해를 돕기 위해, 본 발명에 따른 방법을 개략적으로 설명하고 있다. 이는 앞서 설명한 발명의 내용 중 도 16 내지 도 21에서 설명되지 않은 내용을 배제하는 것이 아님에 유의한다.
- [0308] 본 명세서에서는 ‘영상’과 ‘픽처’를 혼용하여 기재하였으나, 이는 ‘영상’과 ‘픽처’를 별개의 개념으로 구분하여 사용하고자 하는 것이 아니며, 본 명세서에서 ‘영상’은 ‘픽처’를 포함하는 개념으로서, 기술 내용에 따라서 ‘픽처’를 의미할 수 있음에 유의한다.
- [0309] 또한, 본 명세서에서는 ‘깊이 영상’, ‘깊이 픽처’, ‘깊이 맵’을 혼용하여 기재하였으나, 이는 ‘깊이 영상’, ‘깊이 픽처’, ‘깊이 맵’을 별개의 개념으로 구분하여 사용하고자 하는 것이 아니며, 본 명세서에서 ‘깊이 영상’, ‘깊이 픽처’, ‘깊이 맵’은 기술 내용에 따라서 동일한 개념으로 사용될 수 있음에 유의한다.
- [0310] 본 명세서에서 ‘깊이 영상이 텍스처 영상의 움직임 정보를 이용한다’는 것은 깊이 영상 내 현재 깊이 블록의 움직임 정보를 대응하는 텍스처 블록(들)의 움직임 정보(들)을 기반으로 결정하는 것을 의미하며, 텍스처 블록의 움직임 정보를 깊이 블록의 움직임 정보로 이용하거나 텍스처 블록들의 움직임 정보들을 이용하여 깊이 블록의 움직임 정보를 도출하는 것을 포함한다.
- [0311] 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 이하의 특허청구범위 내에 속하는 모든 다른 교체, 수정 및 변경을 포함한다고 할 것이다.

[0312]

[0313]

청구범위

- [청구항 1] 현재 블록의 움직임 정보를 결정하는 단계; 및
상기 움직임 정보를 유도하기 위한 정보를 전송하는 단계를 포함하며,
상기 현재 블록의 움직임 정보를 결정하는 단계에서는
상기 현재 블록의 움직임 정보를 참조 블록의 움직임 정보를 재활용하여 결정하는 것을 특징으로 하는 영상 인코딩 방법.
- [청구항 2] 제1항에 있어서, 상기 현재 블록은 텍스처 픽처의 블록이고,
상기 참조 블록은 참조 뷰 내 블록인 것을 특징으로 하는 영상 인코딩 방법.
- [청구항 3] 제2항에 있어서, 상기 움직임 정보를 유도하기 위한 정보는,
상기 참조 블록의 움직임 정보를 재활용할 것을 지시하는 정보,
상기 참조 뷰의 방향을 지시하는 정보, 상기 참조 블록에 대한
GDV(Global Disparity Vector)를 지시하는 정보 중 적어도 하나를 포함하는 것을 특징으로 하는 영상 인코딩 방법.
- [청구항 4] 제2항에 있어서, 상기 참조 블록에 대한 GDV는 상기 현재 블록이 속하는 픽처 내에서 영역별로 결정되며,
상기 GDV가 결정되는 영역은 쿼드 트리 구조를 가지는 것을 특징으로 하는 영상 인코딩 방법.
- [청구항 5] 제4항에 있어서, 상기 움직임 정보를 유도하기 위한 정보는,
상기 쿼드 트리 구조의 최대 깊이 정보 및 최대 크기 정보를 포함하며,
상기 GDV가 결정되는 영역이 최대 깊이의 영역인 경우에 GDV 정보를 전송하는 것을 특징으로 하는 영상 인코딩 방법.
- [청구항 6] 제4항에 있어서, 상기 움직임 정보를 유도하기 위한 정보는,
상기 쿼드 트리 구조에 대한 분할 여부를 지시하는 정보를 포함하며,
상기 GDV가 결정되는 영역이 분할되지 않는 경우에 GDV 정보를 전송하는 것을 특징으로 하는 영상 인코딩 방법.
- [청구항 7] 제1항에 있어서, 상기 현재 블록은 깊이 픽처의 블록이고,
상기 참조 블록은 동일 뷰의 텍스처 픽처 내 블록인 것을 특징으로 하는 영상 인코딩 방법.
- [청구항 8] 제7항에 있어서, 상기 현재 블록이 상기 참조 블록보다 큰 경우에,
상기 현재 블록의 움직임 정보는 상기 참조 블록의 움직임 정보 및
상기 참조 블록의 주변 블록들의 움직임 정보를 기반으로 결정되는 것을 특징으로 하는 영상 인코딩 방법.
- [청구항 9] 제7항에 있어서, 상기 현재 블록이 상기 참조 블록보다 작은

- 경우에,
 상기 현재 블록의 움직임 정보는 상기 참조 블록의 서브 블록들의 움직임 정보를 기반으로 결정되는 것을 특징으로 하는 영상 인코딩 방법.
- [청구항 10] 제7항에 있어서, 상기 움직임 정보를 유도하기 위한 정보는, 상기 참조 블록의 움직임 정보를 기반을 복원된 현재 블록의 깊이 값을 보상하는 오프셋 정보를 포함하는 것을 특징으로 하는 영상 인코딩 방법.
- [청구항 11] 현재 블록의 움직임 정보를 유도하기 위한 정보를 수신하는 단계; 및
 상기 움직임 정보를 유도하기 위한 정보를 기반으로 상기 현재 블록의 움직임 정보를 유도하는 단계를 포함하며,
 상기 현재 블록의 움직임 정보를 유도하는 단계에서는
 상기 현재 블록의 움직임 정보를 참조 블록의 움직임 정보를 재활용하여 결정하는 것을 특징으로 하는 영상 디코딩 방법.
- [청구항 12] 제11항에 있어서, 상기 현재 블록은 텍스처 픽처의 블록이고, 상기 움직임 정보를 유도하기 위한 정보가 다른 뷰 내 블록의 움직임 정보를 재활용하여 상기 현재 블록의 움직임 정보를 유도할 것을 지시하는 정보를 포함하는 경우에, 참조 뷰 내 블록을 상기 참조 블록으로 설정하는 것을 특징으로 하는 영상 디코딩 방법.
- [청구항 13] 제12항에 있어서, 상기 움직임 정보를 유도하기 위한 정보는, 상기 참조 뷰의 방향을 지시하는 정보 및/또는 상기 참조 블록에 대한 GDV(Global Disparity Vector)를 지시하는 정보를 포함하는 것을 특징으로 하는 영상 디코딩 방법.
- [청구항 14] 제12항에 있어서, 상기 움직임 정보를 유도하기 위한 정보는 상기 참조 블록에 대한 GDV를 포함하며,
 상기 GDV의 값은 상기 현재 블록이 속하는 픽처 내에서 영역별로 설정되는 것을 특징으로 하는 영상 디코딩 방법.
- [청구항 15] 제14항에 있어서, 상기 GDV 값이 특정되는 영역은 쿼드 트리 구조를 가지며,
 상기 움직임 정보를 유도하기 위한 정보는,
 상기 쿼드 트리 구조의 최대 깊이를 지시하는 정보; 및
 상기 현재 블록이 최대 깊이의 GDV 설정 영역인 경우에 GDV 값을 지시하는 정보를 포함하는 것을 특징으로 하는 영상 디코딩 방법.
- [청구항 16] 제14항에 있어서, 상기 GDV 값이 특정되는 영역은 쿼드 트리 구조를 가지며,

상기 움직임 정보를 유도하기 위한 정보는,
 상기 GDV 설정 영역의 분할 여부를 지시하는 정보; 및
 상기 현재 블록이 속하는 GDV 설정 영역이 분할되지 않는 경우에
 GDV 값을 지시하는 정보를 포함하는 것을 특징으로 하는 영상
 디코딩 방법.

[청구항 17]

제11항에 있어서, 상기 현재 블록은 깊이 픽처의 블록이고,
 상기 움직임 정보를 유도하기 위한 정보가 텍스처 픽처 내 블록의
 움직임 정보를 재활용하여 상기 현재 블록의 움직임 정보를
 유도할 것을 지시하는 정보를 포함하는 경우에, 상기 깊이 픽처와
 동일 뷰 내의 텍스처 픽처 내 블록을 상기 참조 블록으로 설정하는
 것을 특징으로 하는 영상 디코딩 방법.

[청구항 18]

제17항에 있어서, 상기 현재 블록이 상기 참조 블록보다 큰 경우에,
 상기 현재 블록의 움직임 정보는 상기 참조 블록의 움직임 정보 및
 상기 참조 블록의 주변 블록들의 움직임 정보를 기반으로
 유도되는 것을 특징으로 하는 영상 디코딩 방법.

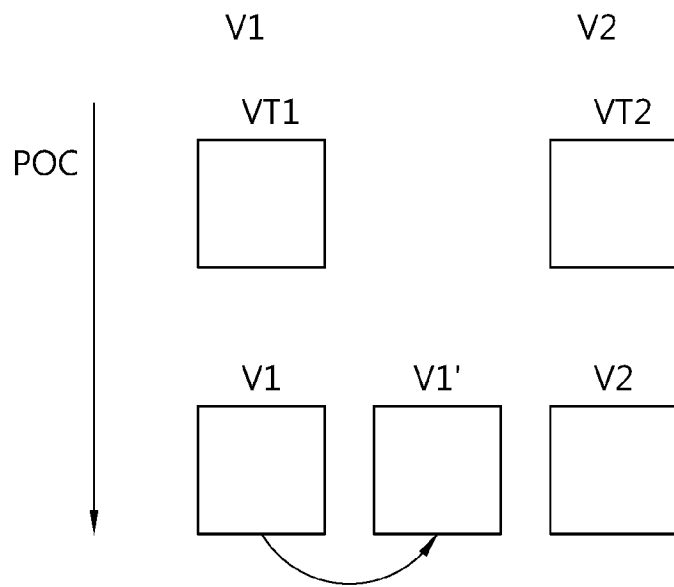
[청구항 19]

제17항에 있어서, 상기 현재 블록이 상기 참조 블록보다 작은
 경우에,
 상기 현재 블록의 움직임 정보는 상기 참조 블록의 서브 블록들의
 움직임 정보를 기반으로 유도되는 것을 특징으로 하는 영상
 디코딩 방법.

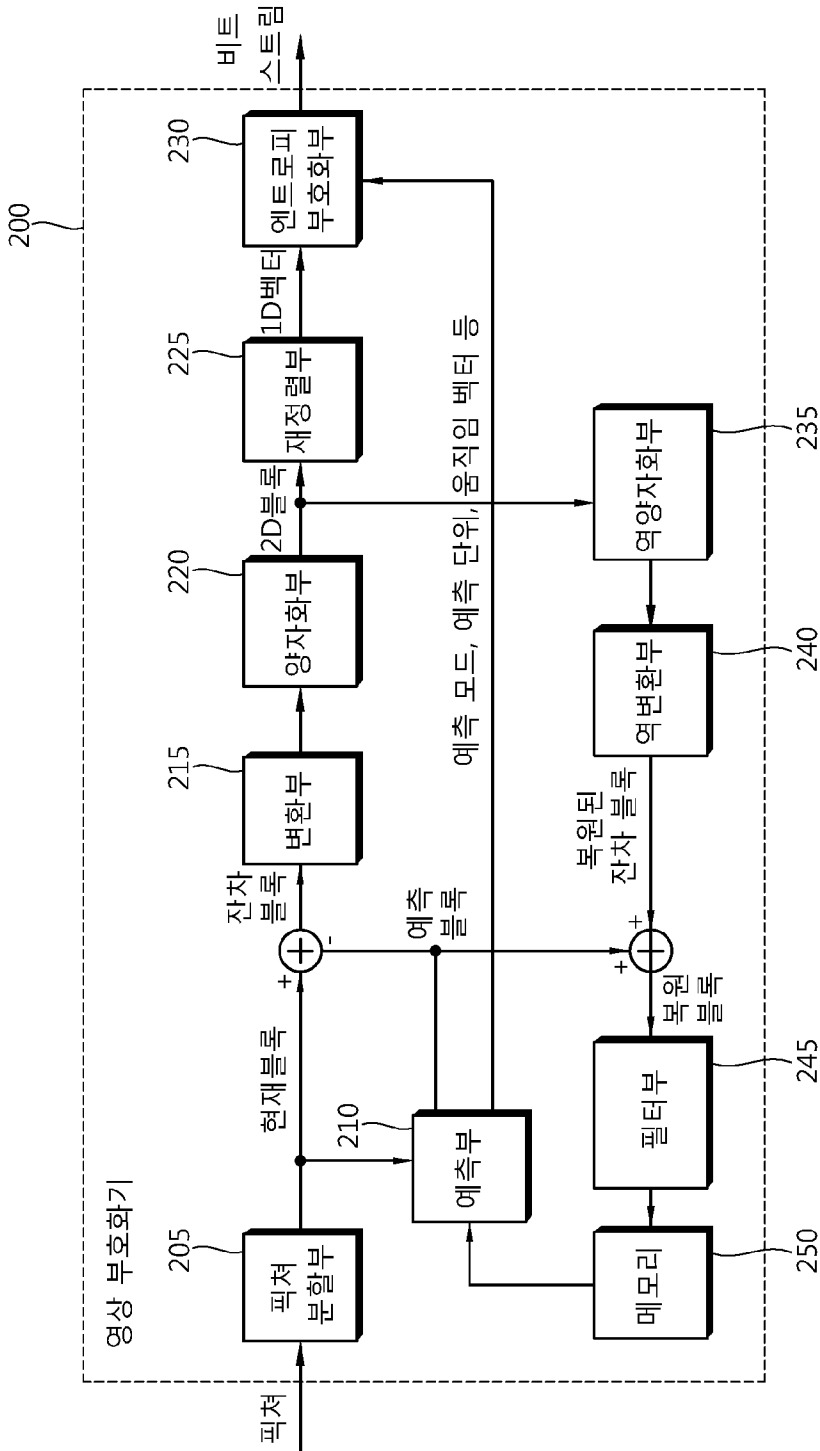
[청구항 20]

제17항에 있어서, 상기 움직임 정보를 유도하기 위한 정보는,
 상기 참조 블록의 움직임 정보를 기반을 복원된 현재 블록의 깊이
 값을 보상하는 오프셋 정보를 포함하는 것을 특징으로 하는 영상
 디코딩 방법.

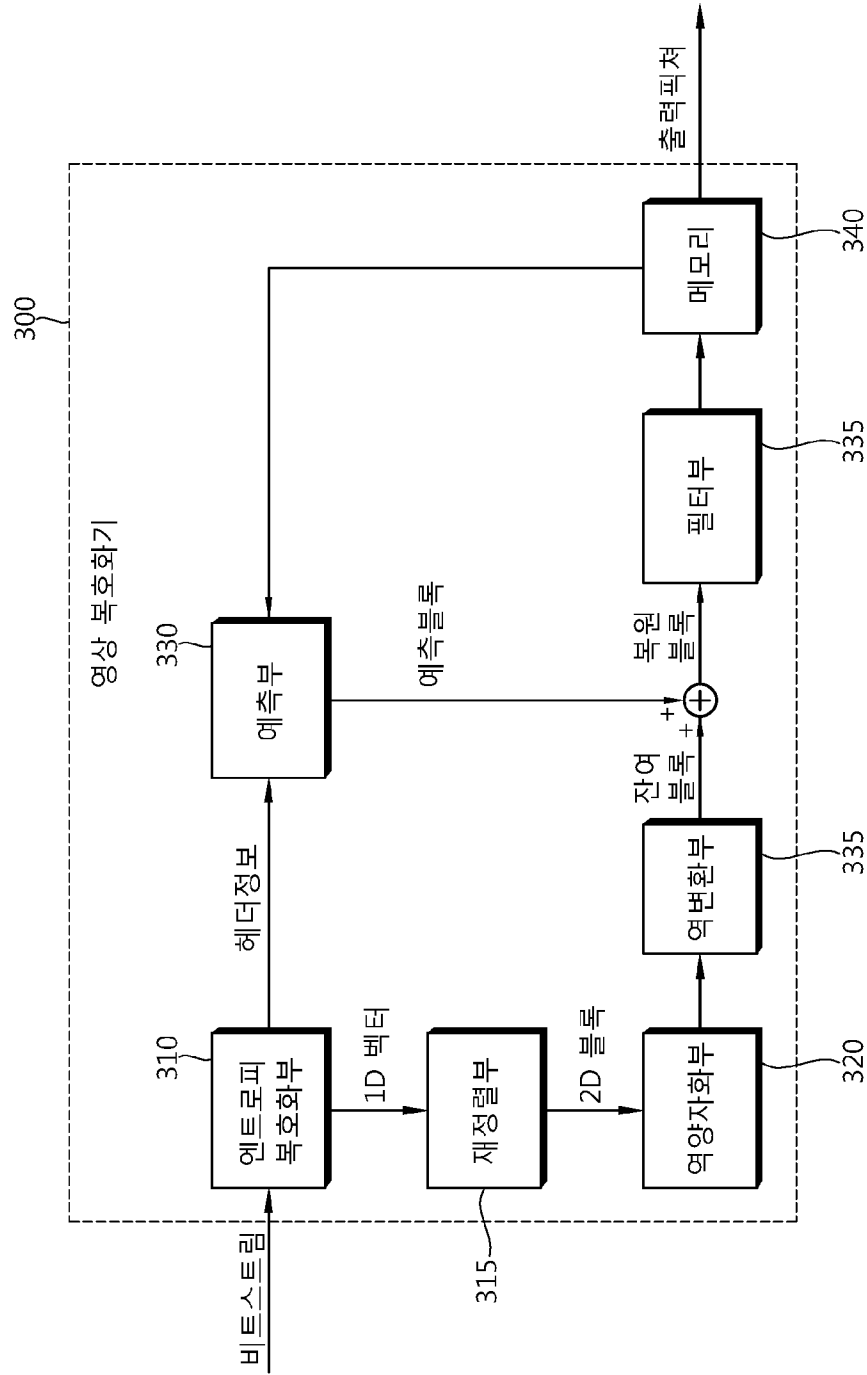
[Fig. 1]



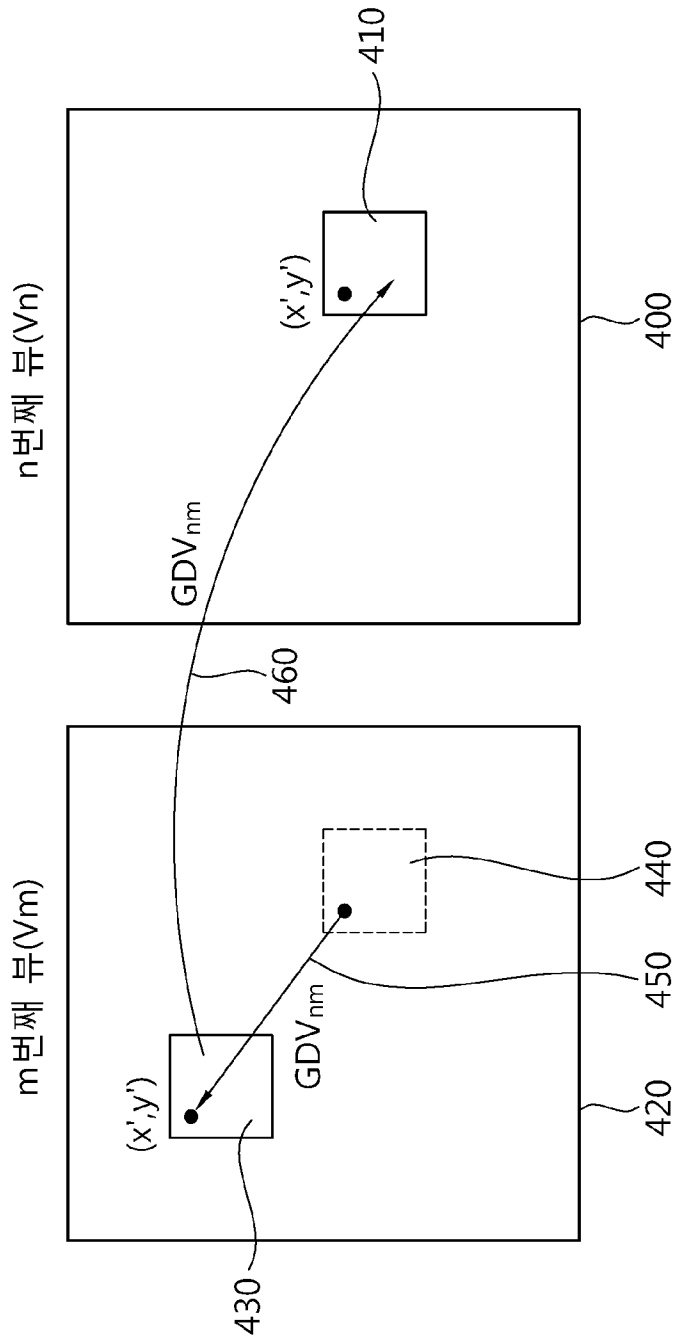
[Fig. 2]



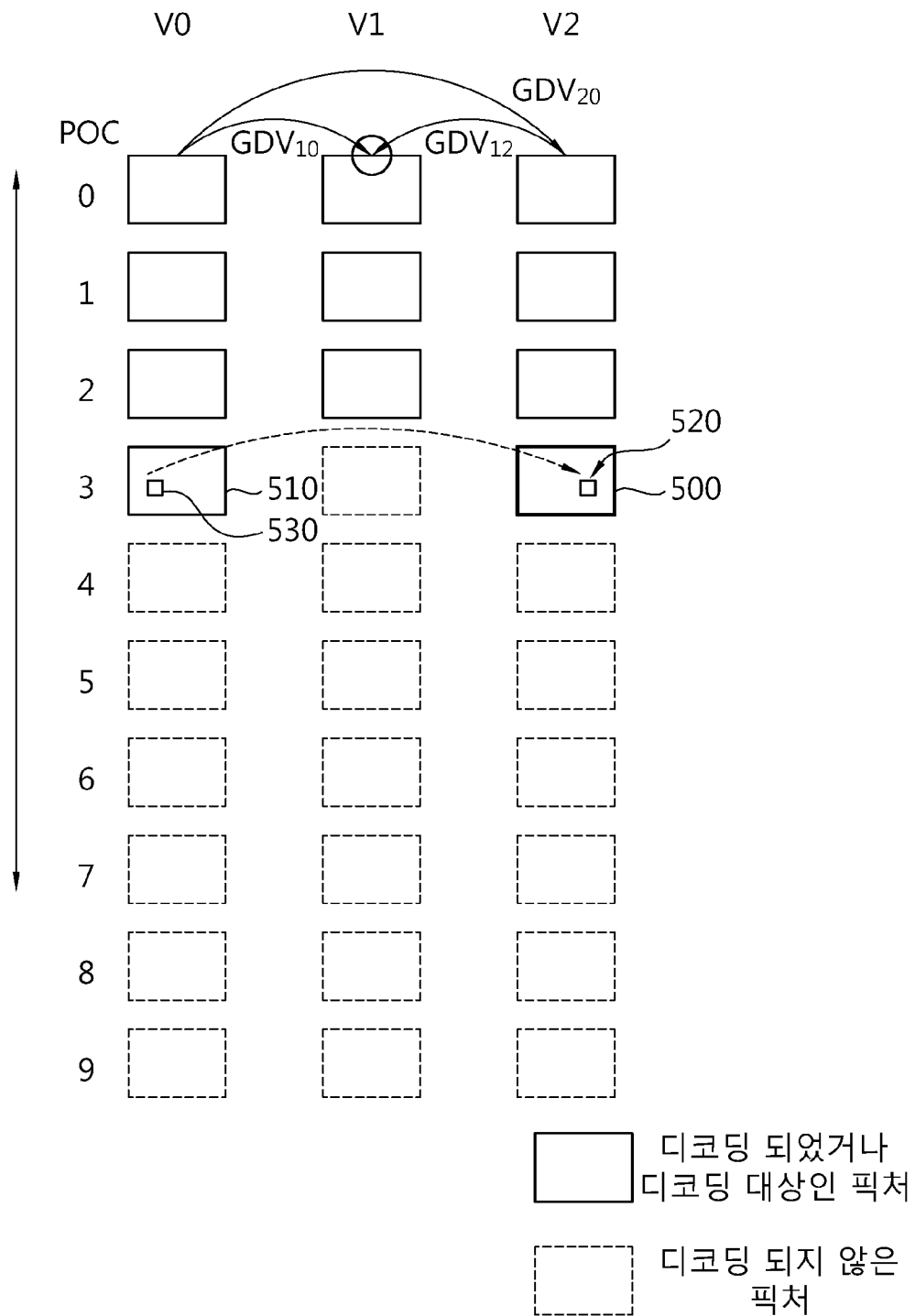
[Fig. 3]



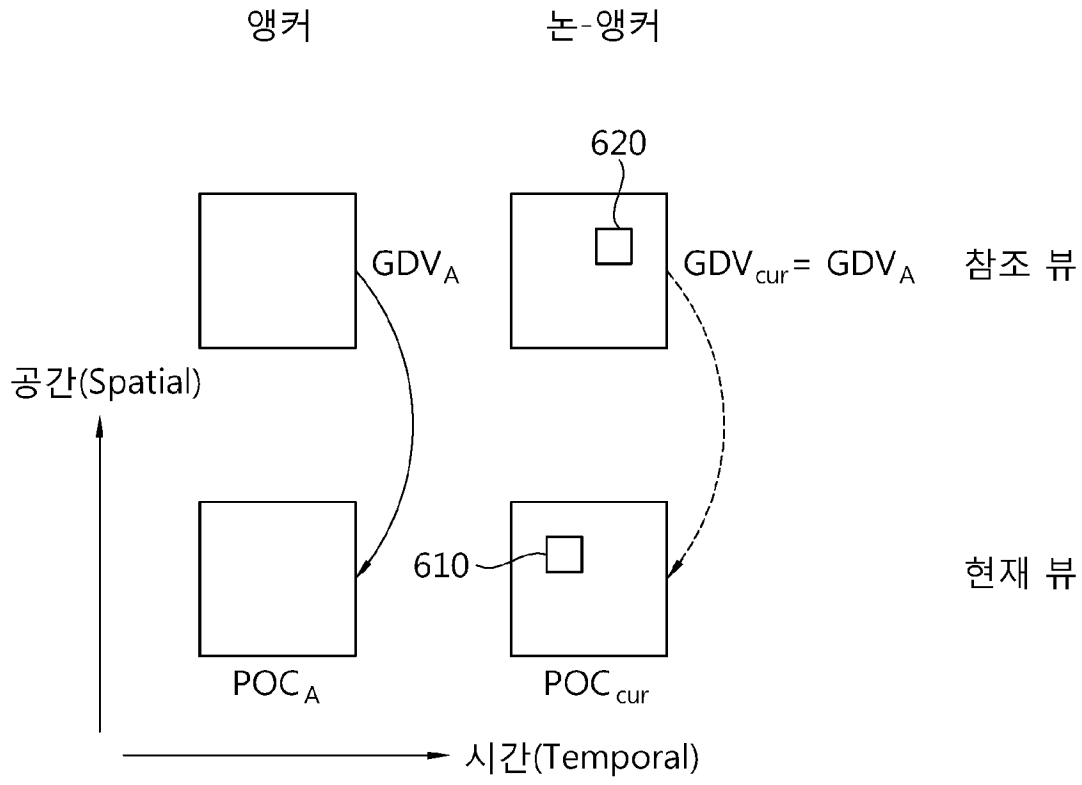
[Fig. 4]



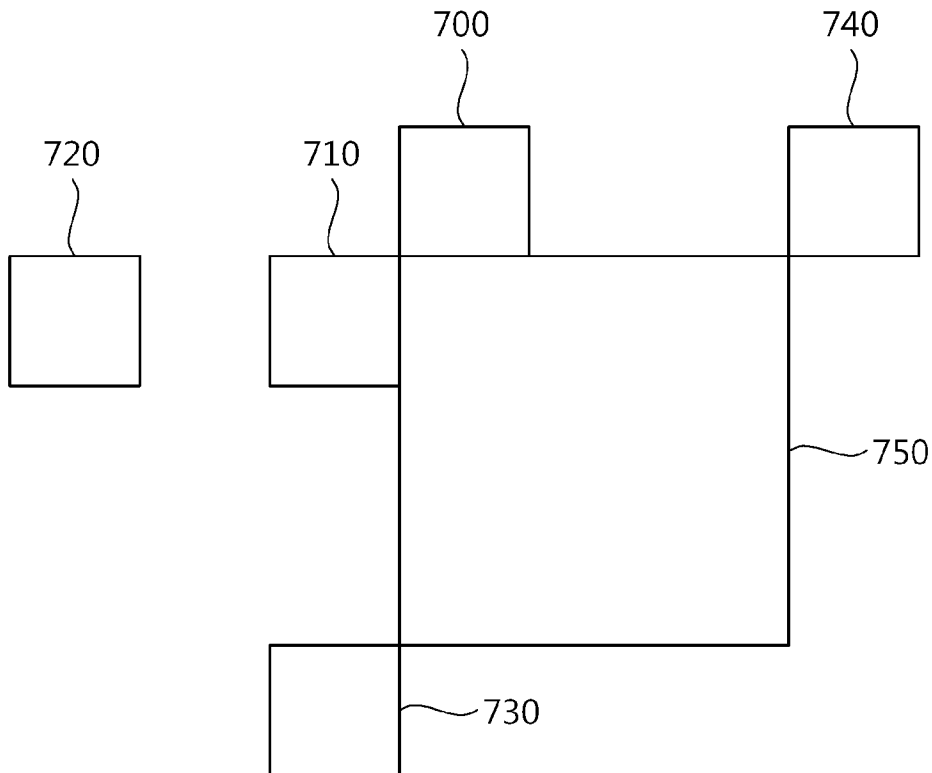
[Fig. 5]



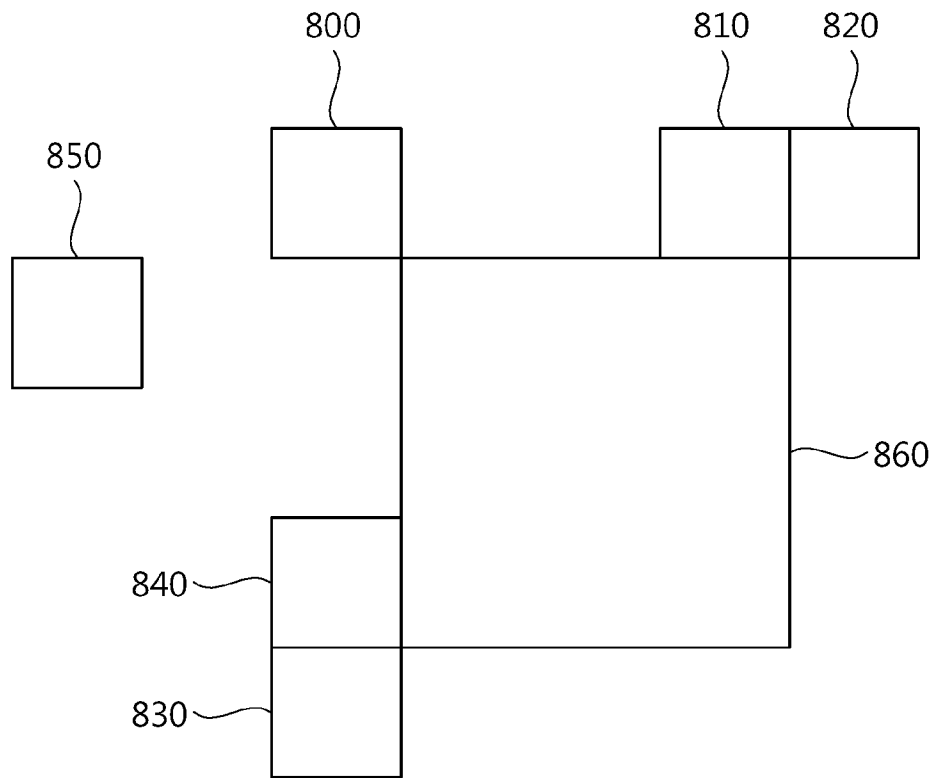
[Fig. 6]



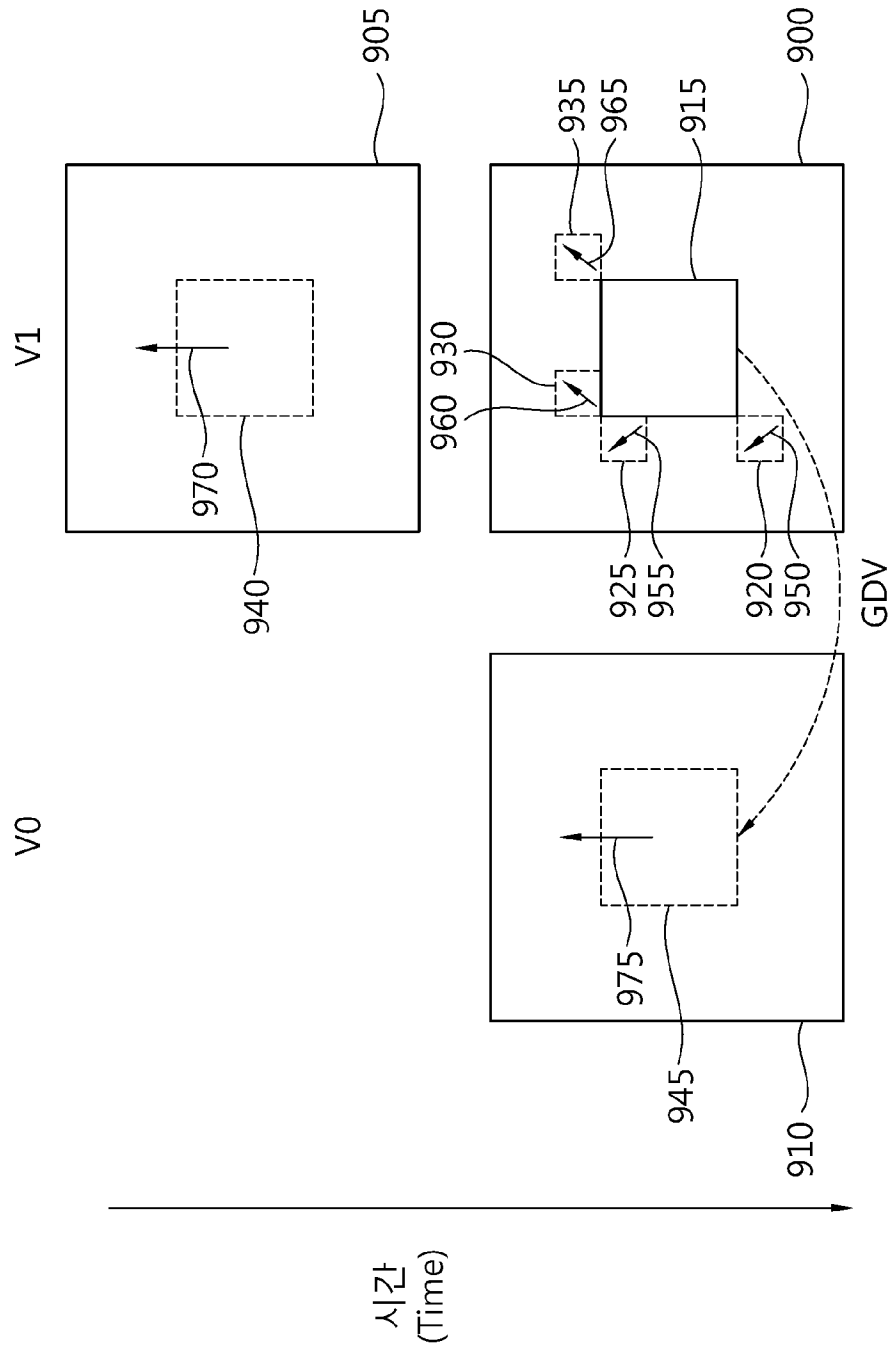
[Fig. 7]



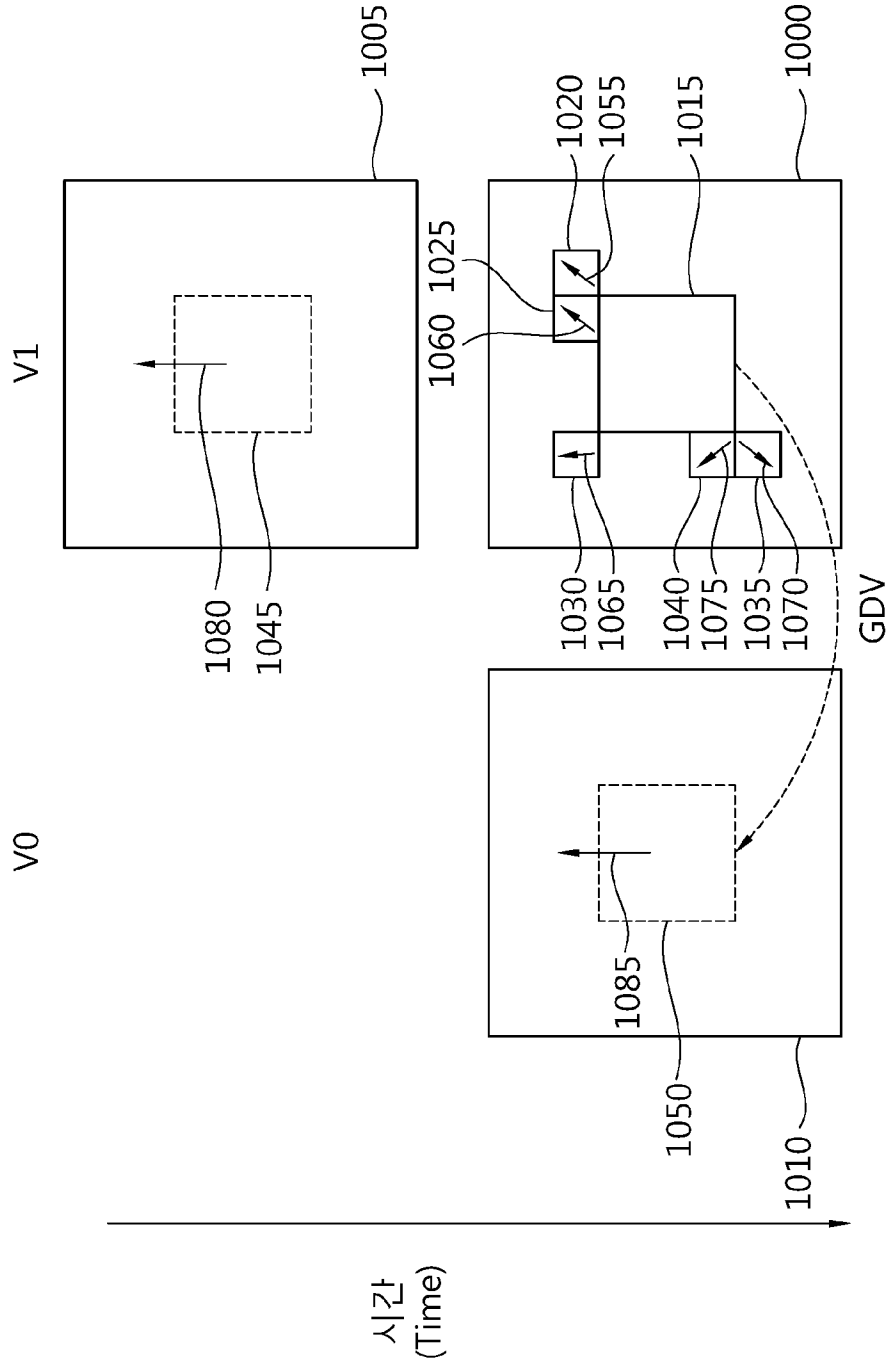
[Fig. 8]



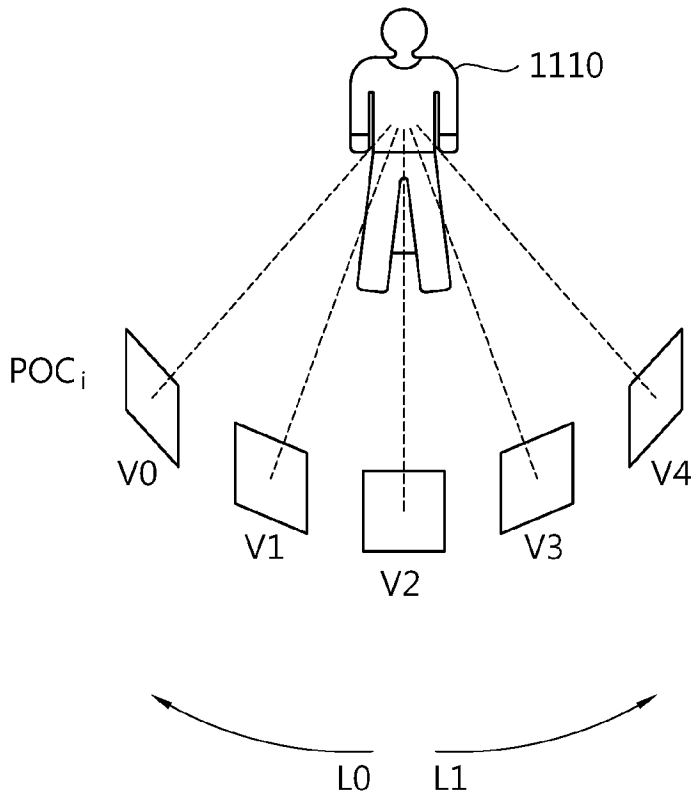
[Fig. 9]



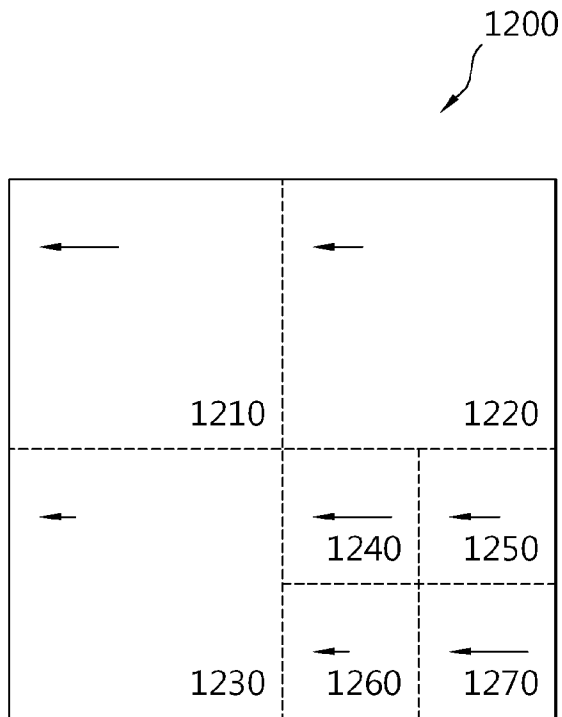
[Fig. 10]



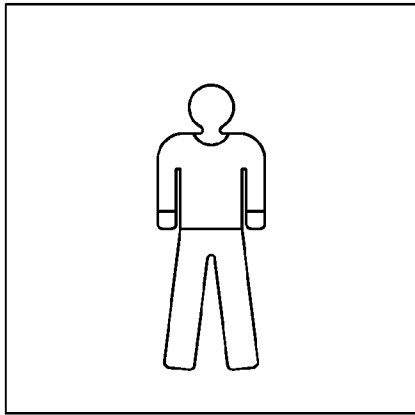
[Fig. 11]



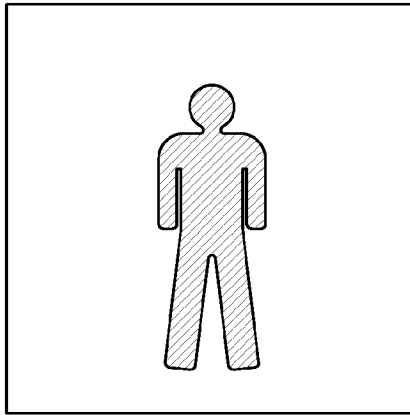
[Fig. 12]



[Fig. 13]

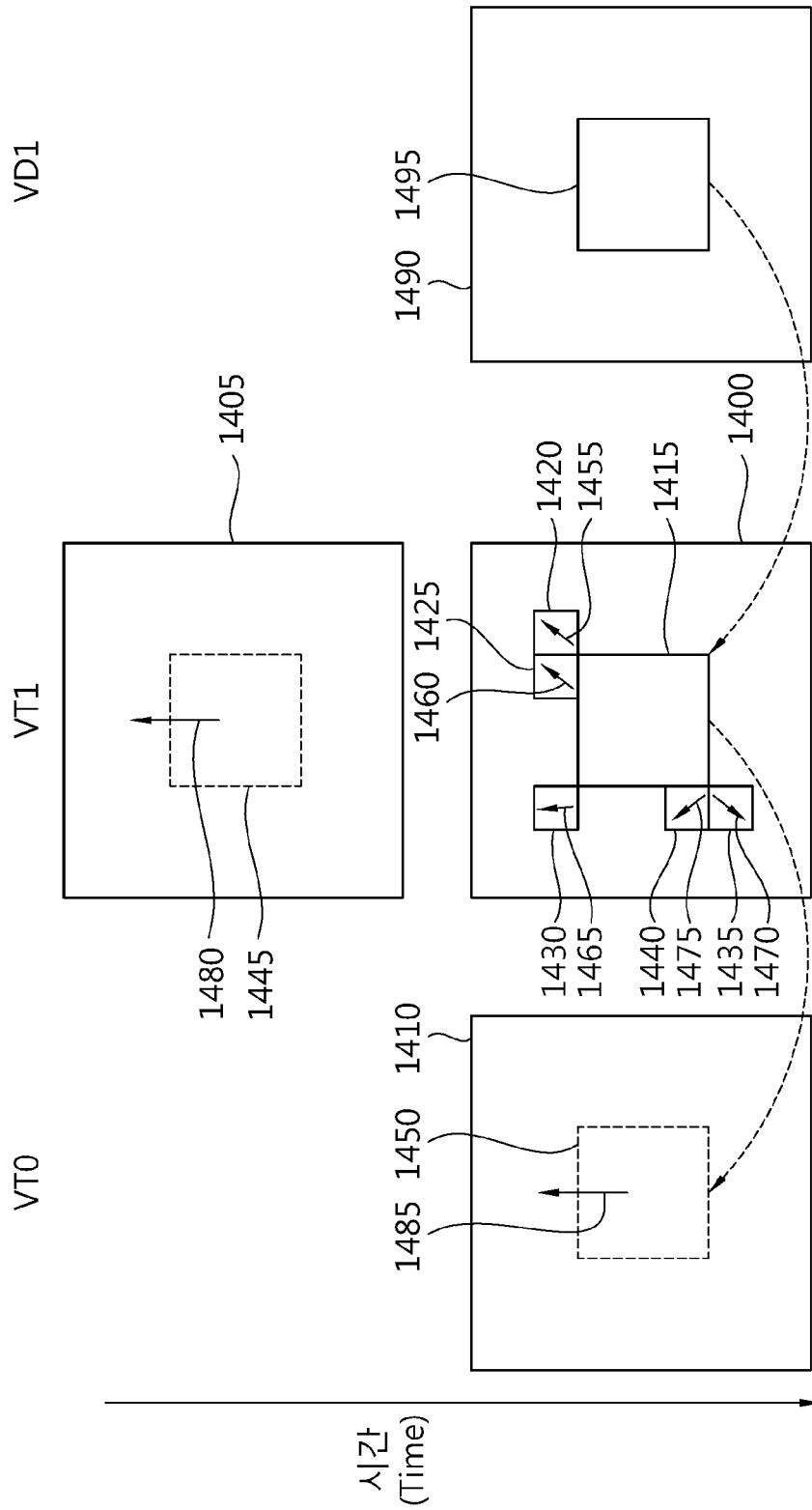


(a)

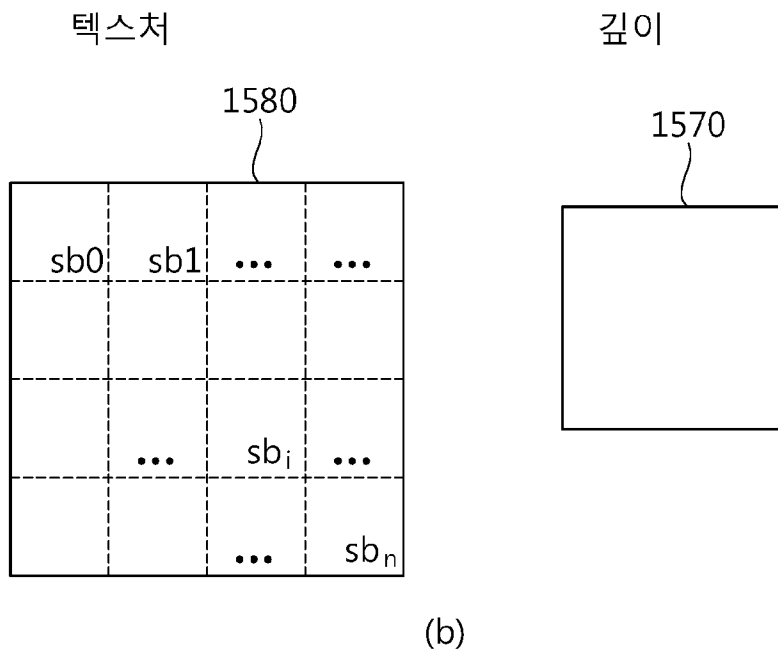
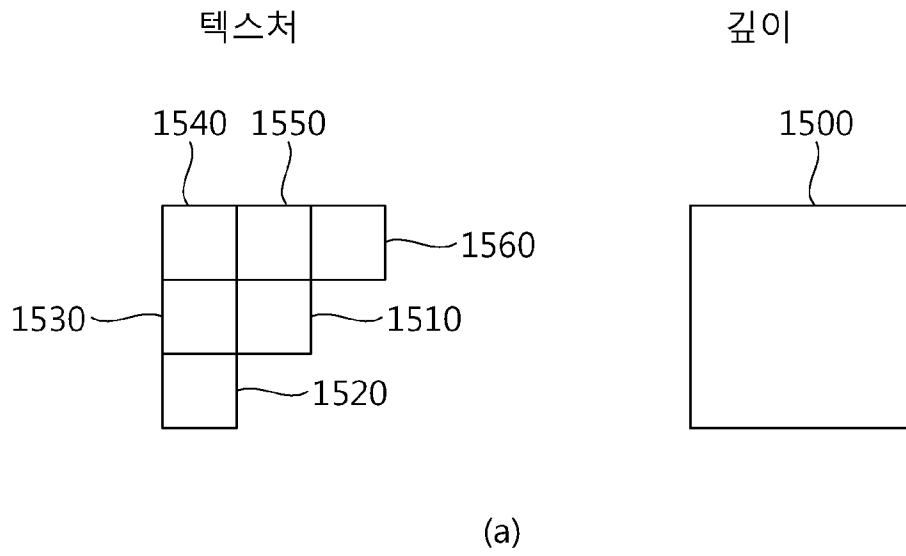


(b)

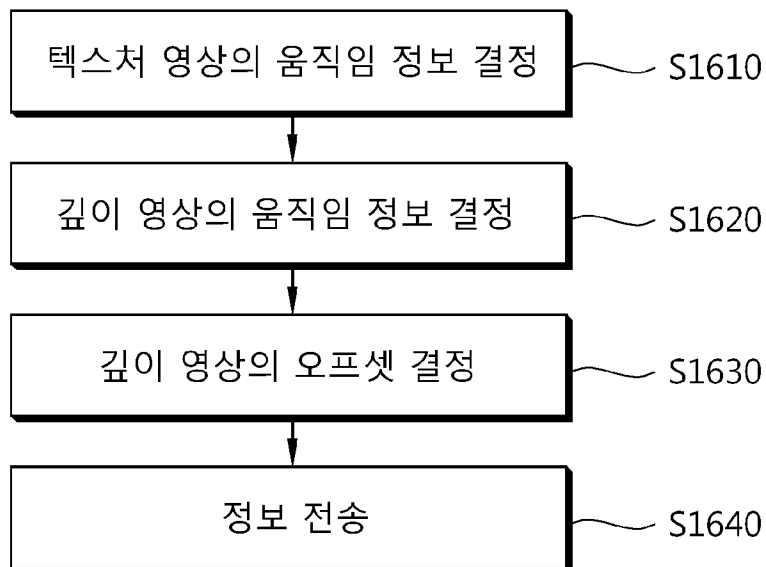
[Fig. 14]



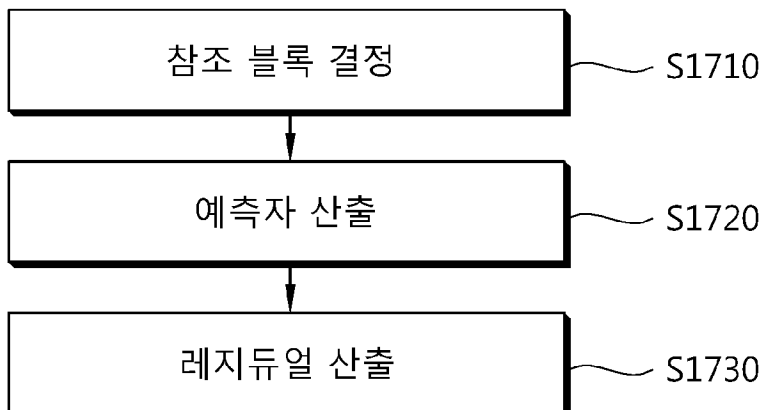
[Fig. 15]



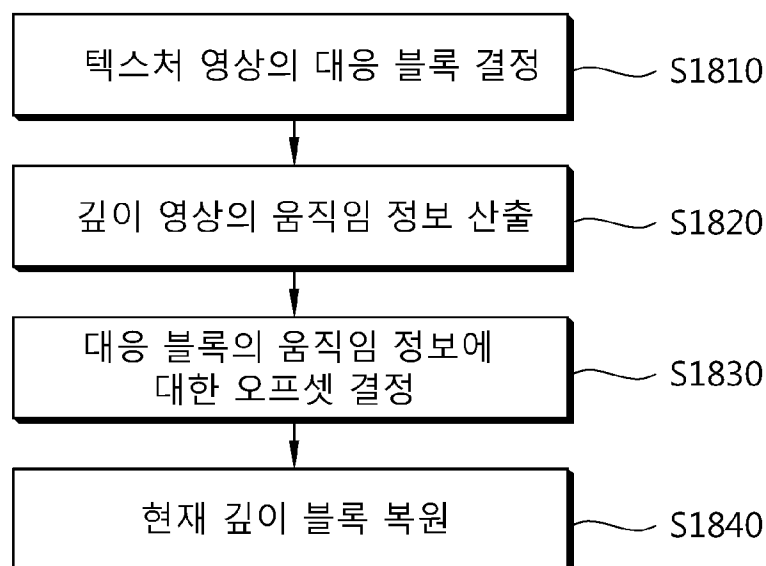
[Fig. 16]



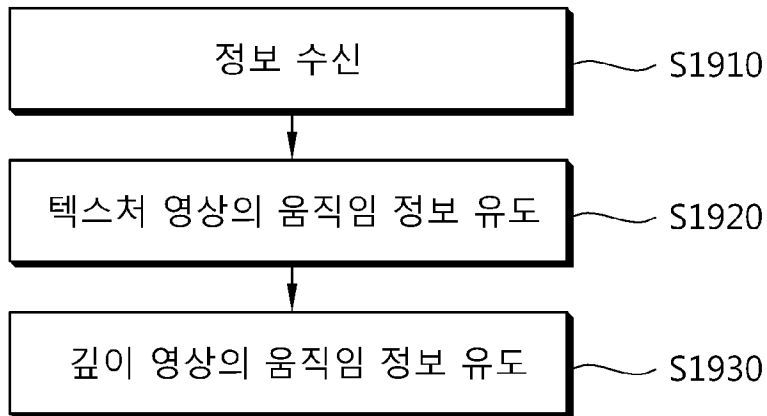
[Fig. 17]



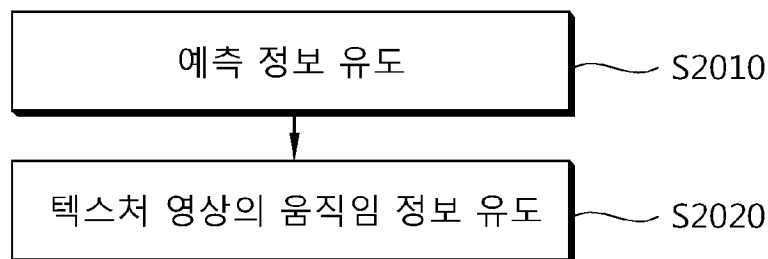
[Fig. 18]



[Fig. 19]



[Fig. 20]



[Fig. 21]

