



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2024년11월01일

(11) 등록번호 10-2725370

(24) 등록일자 2024년10월30일

- (51) 국제특허분류(Int. Cl.)
H04N 19/513 (2014.01) *H04N 19/577* (2014.01)
H04N 19/587 (2014.01)
- (52) CPC특허분류
H04N 19/513 (2015.01)
H04N 19/44 (2015.01)
- (21) 출원번호 10-2019-7030837
- (22) 출원일자(국제) 2018년03월22일
 심사청구일자 2021년03월05일
- (85) 번역문제출일자 2019년10월18일
- (65) 공개번호 10-2019-0127884
- (43) 공개일자 2019년11월13일
- (86) 국제출원번호 PCT/US2018/023761
- (87) 국제공개번호 WO 2018/175720
 국제공개일자 2018년09월27일
- (30) 우선권주장
 62/475,177 2017년03월22일 미국(US)
 15/927,854 2018년03월21일 미국(US)
- (56) 선행기술조사문헌
 KR1020140034292 A*
 WO2017036414 A1*
 *는 심사관에 의하여 인용된 문헌

- (73) 특허권자
 쉐캅 인코포레이티드
 미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
- (72) 발명자
 카르체비츠 마르타
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
 천 이-웬
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
 (뒷면에 계속)
- (74) 대리인
 특허법인코리아나

전체 청구항 수 : 총 18 항

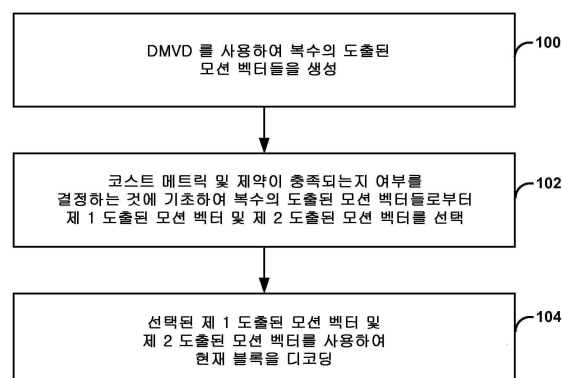
심사관 : 박상철

(54) 발명의 명칭 디코더 측 모션 벡터 도출에 의해 도출된 모션 벡터 정보의 제약

(57) 요약

디코더 측 모션 벡터 도출(DMVD)과 관련된 기법들이 기술된다. 예를 들어, 본 개시는 DMVD에 의해 도출된 모션 벡터(MV), 및/또는 초기 MV와 DMVD에 의해 도출된 MV간의 MV차이와 같은 모션 정보에 하나 이상의 제약들을 적용하는 것과 관련된 기법들을 기술한다. 제약이 DMVD에 적용될 때, 특정 예들에서, 제약을 만족하는 도출된 모션 정보만이 유효한 모션 정보로 간주된다. 조건들이 제약들에 배치될 수도 있다.

대표도 - 도16



(52) CPC특허분류

H04N 19/577 (2015.01)

H04N 19/587 (2015.01)

(72) 발명자

치옌 웨이-정

미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775

추앙 샤오-창

미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775

리 상

미국 92130 캘리포니아주 샌디에고 게일몬트 레인 10574

천 지안레

미국 92130 캘리포니아주 샌디에고 코르테 데 티뷰론 10756

명세서

청구범위

청구항 1

인코딩된 비디오 비트스트림으로부터 비디오 데이터를 디코딩하는 방법으로서,

디코더 측 모션 벡터 도출 (DMVD) 을 사용하여 복수의 도출된 모션 벡터들을 생성하는 단계;

제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 상기 제 1 도출된 모션 벡터와 상기 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다고 결정하는 단계를 포함하는, 코스트 메트릭에 기초하여 상기 복수의 도출된 모션 벡터들로부터 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 결정하는 단계로서,

상기 관계는 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것을 포함하고,

상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 갖는 것으로 결정하는 단계는 다음을 결정하는 단계를 포함하고:

$$L0_MV_X_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_X_Diff * (CURR_POC - L0_POC_Derived) \text{ 또는}$$

$$L0_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived),$$

여기서,

$L0_MV_X_Diff$ 는 제 1 초기 모션 벡터와 상기 제 1 도출된 모션 벡터 간의 차이의 X 성분이고,

$L0_MV_Y_Diff$ 는 상기 제 1 초기 모션 벡터와 상기 제 1 도출된 모션 벡터 간의 차이의 Y 성분이고,

$L1_MV_X_Diff$ 는 제 2 초기 모션 벡터와 상기 제 2 도출된 모션 벡터 간의 차이의 X 성분이고,

$L1_MV_Y_Diff$ 는 상기 제 2 초기 모션 벡터와 상기 제 2 도출된 모션 벡터 간의 차이의 Y 성분이고,

$CURR_POC$ 는 현재 픽처의 현재 픽처 순서 카운트 (POC) 이고,

$L0_POC_Derived$ 는 제 1 리스트의 모션 벡터들에 대한 도출된 POC 이고,

$L1_POC_Derived$ 는 제 2 리스트의 모션 벡터들에 대한 도출된 POC 인, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 단계; 및

결정된 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 디코딩하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 2

제 1 항에 있어서,

상기 복수의 도출된 모션 벡터를 생성하기 위해 DMVD 를 사용하는 것은 양방향 광학 흐름 (BIO) 프로세스, 프레임 레이트 상향 변환 (FRUC) 프로세스, 양방향 매칭 프로세스, FRUC 템플릿 매칭 프로세스, 또는 양방향 템플릿 매칭 프로세스 중 하나를 사용하는 것을 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 3

삭제

청구항 4

삭제

청구항 5

삭제

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

제 1 항에 있어서,

모션 벡터 후보 리스트의 제 1 초기 모션 벡터 및 상기 모션 벡터 후보 리스트의 제 2 초기 모션 벡터가 현재 픽처 앞에 있는지 또는 상기 현재 픽처 뒤에 있는지를 결정하는 단계를 더 포함하고;

상기 코스트 메트릭에 기초하여 상기 복수의 도출된 모션 벡터들로부터 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 결정하는 단계는 상기 결정에 더 기초하는, 비디오 데이터를 디코딩하는 방법.

청구항 10

제 1 항에 있어서,

제 1 모션 벡터 후보리스트와 연관된 제 1 참조 픽처와 현재 픽처 사이의 제 1 픽처 순서 카운트 (POC) 거리가 제로이고 제 2 모션 벡터 후보 리스트와 연관된 제 2 참조 픽처와 상기 현재 픽처 사이의 제 2 POC 거리가 제로 인지 여부를 결정하는 단계를 더 포함하고,

상기 코스트 메트릭에 기초하여 상기 복수의 도출된 모션 벡터들로부터 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 결정하는 단계는 상기 결정에 더 기초하는, 비디오 데이터를 디코딩하는 방법.

청구항 11

제 1 항에 있어서,

제 1 모션 벡터 후보리스트와 연관된 제 1 참조 픽처와 현재 픽처 사이의 제 1 픽처 순서 카운트 (POC) 거리가 제로이고 제 2 모션 벡터 후보 리스트와 연관된 제 2 참조 픽처와 상기 현재 픽처 사이의 제 2 POC 거리가 제로 인지 여부를 결정하는 단계를 더 포함하고,

상기 복수의 도출된 모션 벡터들을 생성하는 상기 코스트 메트릭에 기초하여 상기 복수의 도출된 모션 벡터들로부터 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 결정하는 단계는 상기 제 1 모션 벡터 후보 리스트와 연관된 상기 제 1 참조 픽처와 상기 현재 픽처 사이의 상기 제 1 POC 거리가 제로이고 상기 제 2 모션 벡터 후보 리스트와 연관된 상기 제 2 참조 픽처와 상기 현재 픽처 사이의 상기 제 2 POC 거리가 제로라고 결정하는 단계에 더 기초하는, 비디오 데이터를 디코딩하는 방법.

청구항 12

제 1 항에 있어서,

상기 인코딩된 비디오 비트 스트림으로부터 상기 적어도 하나의 제약을 결정하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 13

제 1 항에 있어서,

상기 코스트 메트릭은 절대 차이들의 합을 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 14

인코딩된 비디오 비트 스트림으로부터 비디오 데이터를 디코딩하도록 구성된 장치로서,

상기 비디오 데이터를 저장하도록 구성되는 메모리; 및

상기 메모리와 통신하는 하나 이상의 프로세서들을 포함하고, 상기 하나 이상의 프로세서들은,

디코더 측 모션 벡터 도출 (DMVD) 을 사용하여 복수의 도출된 모션 벡터들을 생성하고;

제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 상기 제 1 도출된 모션 벡터와 상기 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다는 결정을 포함하는, 코스트 메트릭에 기초하여 상기 복수의 도출된 모션 벡터들로부터 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 결정하는 것으로서,

상기 관계는 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것을 포함하고,

상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 갖는다는 결정은 다음의 결정을 포함하고:

$LO_MV_X_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_X_Diff * (CURR_POC - L0_POC_Derived)$ 또는

$LO_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived)$,

여기서,

$LO_MV_X_Diff$ 는 제 1 초기 모션 벡터와 상기 제 1 도출된 모션 벡터 간의 차이의 X 성분이고,

$LO_MV_Y_Diff$ 는 상기 제 1 초기 모션 벡터와 상기 제 1 도출된 모션 벡터 간의 차이의 Y 성분이고,

$L1_MV_X_Diff$ 는 제 2 초기 모션 벡터와 상기 제 2 도출된 모션 벡터 간의 차이의 X 성분이고,

$L1_MV_Y_Diff$ 는 상기 제 2 초기 모션 벡터와 상기 제 2 도출된 모션 벡터 간의 차이의 Y 성분이고,

$CURR_POC$ 는 현재 픽처의 현재 픽처 순서 카운트 (POC) 이고,

$L0_POC_Derived$ 는 제 1 리스트의 모션 벡터들에 대한 도출된 POC 이고,

$L1_POC_Derived$ 는 제 2 리스트의 모션 벡터들에 대한 도출된 POC 인, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하며;

결정된 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 디코딩하도록 구성된, 비디오 데이터를 디코딩하도록 구성된 장치.

청구항 15

제 14 항에 있어서,

상기 복수의 도출된 모션 벡터를 생성하기 위해 DMVD 를 사용하는 것은 양방향 광학 흐름 (BIO) 프로세스, 프레임 레이트 상향 변환 (FRUC) 프로세스, 양방향 매칭 프로세스, FRUC 템플릿 매칭 프로세스, 또는 양방향 템플릿

매칭 프로세스 중 하나를 사용하는 것을 포함하는, 비디오 데이터를 디코딩하도록 구성된 장치.

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

청구항 22

제 14 항에 있어서,

상기 하나 이상의 프로세서들은 또한, 모션 벡터 후보 리스트의 제 1 초기 모션 벡터 및 상기 모션 벡터 후보 리스트의 제 2 초기 모션 벡터가 현재 픽처 앞에 있는지 또는 상기 현재 픽처 뒤에 있는지를 결정하도록 구성되고;

상기 코스트 메트릭에 기초하여 상기 복수의 도출된 모션 벡터들로부터 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 결정하는 것은 상기 결정에 더 기초하는, 비디오 데이터를 디코딩하도록 구성된 장치.

청구항 23

제 14 항에 있어서,

상기 하나 이상의 프로세서들은 또한, 제 1 모션 벡터 후보리스트와 연관된 제 1 참조 픽처와 현재 픽처 사이의 제 1 픽처 순서 카운트 (POC) 거리가 제로이고 제 2 모션 벡터 후보 리스트와 연관된 제 2 참조 픽처와 상기 현재 픽처 사이의 제 2 POC 거리가 제로인지 여부를 결정하도록 구성되고,

상기 코스트 메트릭에 기초하여 상기 복수의 도출된 모션 벡터들로부터 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 결정하는 것은 상기 결정에 더 기초하는, 비디오 데이터를 디코딩하도록 구성된 장치.

청구항 24

제 14 항에 있어서,

상기 하나 이상의 프로세서들은 또한, 제 1 모션 벡터 후보리스트와 연관된 제 1 참조 픽처와 현재 픽처 사이의 제 1 픽처 순서 카운트 (POC) 거리가 제로이고 제 2 모션 벡터 후보 리스트와 연관된 제 2 참조 픽처와 상기 현재 픽처 사이의 제 2 POC 거리가 제로인지 여부를 결정하도록 구성되고,

상기 복수의 도출된 모션 벡터들을 생성하는 상기 코스트 메트릭에 기초하여 상기 복수의 도출된 모션 벡터들로부터 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 결정하는 것은 상기 제 1 모션 벡터 후보 리스트와 연관된 상기 제 1 참조 픽처와 상기 현재 픽처 사이의 상기 제 1 POC 거리가 제로이고 상기 제 2 모션 벡터 후보 리스트와 연관된 상기 제 2 참조 픽처와 상기 현재 픽처 사이의 상기 제 2 POC 거리가 제로라고 결정하는 것에 더 기초하는, 비디오 데이터를 디코딩하도록 구성된 장치.

청구항 25

제 14 항에 있어서,

상기 하나 이상의 프로세서들은 또한 상기 인코딩된 비디오 비트 스트림으로부터 상기 적어도 하나의 제약을 결정하도록 구성되는, 비디오 데이터를 디코딩하도록 구성된 장치.

청구항 26

제 14 항에 있어서,

상기 코스트 메트릭은 절대 차이들의 합을 포함하는, 비디오 데이터를 디코딩하도록 구성된 장치.

청구항 27

인코딩된 비디오 비트 스트림으로부터 비디오 데이터를 디코딩하도록 구성된 장치로서,

디코더 측 모션 벡터 도출 (DMVD) 을 사용하여 복수의 도출된 모션 벡터들을 생성하는 수단;

제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 상기 제 1 도출된 모션 벡터와 상기 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다고 결정하는 수단을 포함하는, 코스트 메트릭에 기초하여 상기 복수의 도출된 모션 벡터들로부터 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 결정하는 수단으로서,

상기 관계는 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것을 포함하고,

상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 갖는 것으로 결정하는 수단은 다음을 결정하는 수단을 포함하고:

$LO_MV_X_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_X_Diff * (CURR_POC - LO_POC_Derived)$ 또는

$LO_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_Y_Diff * (CURR_POC - LO_POC_Derived)$,

여기서,

$LO_MV_X_Diff$ 는 제 1 초기 모션 벡터와 상기 제 1 도출된 모션 벡터 간의 차이의 X 성분이고,

$LO_MV_Y_Diff$ 는 상기 제 1 초기 모션 벡터와 상기 제 1 도출된 모션 벡터 간의 차이의 Y 성분이고,

$L1_MV_X_Diff$ 는 제 2 초기 모션 벡터와 상기 제 2 도출된 모션 벡터 간의 차이의 X 성분이고,

$L1_MV_Y_Diff$ 는 상기 제 2 초기 모션 벡터와 상기 제 2 도출된 모션 벡터 간의 차이의 Y 성분이고,

$CURR_POC$ 는 현재 픽처의 현재 픽처 순서 카운트 (POC) 이고,

$LO_POC_Derived$ 는 제 1 리스트의 모션 벡터들에 대한 도출된 POC 이고,

$L1_POC_Derived$ 는 제 2 리스트의 모션 벡터들에 대한 도출된 POC 인, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 수단; 및

결정된 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 디코딩하는 수단을 포함하는, 비디오 데이터를 디코딩하도록 구성된 장치.

청구항 28

제 27 항에 있어서,

상기 복수의 도출된 모션 벡터를 생성하기 위해 DMVD 를 사용하는 것은 양방향 광학 흐름 (BIO) 프로세스, 프레임 레이트 상향 변환 (FRUC) 프로세스, 양방향 매칭 프로세스, FRUC 템플릿 매칭 프로세스, 또는 양방향 템플릿

매칭 프로세스 중 하나를 사용하는 것을 포함하는, 비디오 데이터를 디코딩하도록 구성된 장치.

청구항 29

비디오 데이터를 인코딩하도록 구성된 장치로서,

상기 비디오 데이터를 저장하도록 구성되는 메모리; 및

상기 메모리와 통신하는 하나 이상의 프로세서들을 포함하고, 상기 하나 이상의 프로세서들은,

디코더 측 모션 벡터 도출 (DMVD) 을 사용하여 복수의 도출된 모션 벡터들을 생성하고;

제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 상기 제 1 도출된 모션 벡터와 상기 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다는 결정을 포함하는, 코스트 메트릭에 기초하여 상기 복수의 도출된 모션 벡터들로부터 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 결정하는 것으로서,

상기 관계는 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것을 포함하고,

상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 갖는다는 결정은 다음의 결정을 포함하고:

$LO_MV_X_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_X_Diff * (CURR_POC - L0_POC_Derived)$ 또는

$LO_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived)$,

여기서,

$LO_MV_X_Diff$ 는 제 1 초기 모션 벡터와 상기 제 1 도출된 모션 벡터 간의 차이의 X 성분이고,

$LO_MV_Y_Diff$ 는 상기 제 1 초기 모션 벡터와 상기 제 1 도출된 모션 벡터 간의 차이의 Y 성분이고,

$L1_MV_X_Diff$ 는 제 2 초기 모션 벡터와 상기 제 2 도출된 모션 벡터 간의 차이의 X 성분이고,

$L1_MV_Y_Diff$ 는 상기 제 2 초기 모션 벡터와 상기 제 2 도출된 모션 벡터 간의 차이의 Y 성분이고,

$CURR_POC$ 는 현재 픽처의 현재 픽처 순서 카운트 (POC) 이고,

$L0_POC_Derived$ 는 제 1 리스트의 모션 벡터들에 대한 도출된 POC 이고,

$L1_POC_Derived$ 는 제 2 리스트의 모션 벡터들에 대한 도출된 POC 인, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하며;

결정된 상기 제 1 도출된 모션 벡터 및 상기 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 인코딩하도록 구성된, 비디오 데이터를 인코딩하도록 구성된 장치.

청구항 30

제 29 항에 있어서,

상기 복수의 도출된 모션 벡터를 생성하기 위해 DMVD 를 사용하는 것은 양방향 광학 흐름 (BIO) 프로세스, 프레임 레이트 상향 변환 (FRUC) 프로세스, 양방향 매칭 프로세스, FRUC 템플릿 매칭 프로세스, 또는 양방향 템플릿 매칭 프로세스 중 하나를 사용하는 것을 포함하는, 비디오 데이터를 인코딩하도록 구성된 장치.

발명의 설명

기술 분야

[0001] 본원은 2017년 3월 22일자로 출원된 미국 가특허출원 제 62/475,177 호의 이익을 주장하며, 그의 전체 내용은 참조에 의해 본원에 포함된다.

[0002] 본 개시는 비디오 코딩에 관한 것이다.

배경 기술

[0003] 디지털 비디오 능력들은 디지털 텔레비전들, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인용 디지털 보조기들 (PDA들), 랩탑 또는 데스크탑 컴퓨터들, 태블릿 컴퓨터들, e-북 리더들, 디지털 카메라들, 디지털 레코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 소위 "스마트 폰들", 비디오 텔레컨퍼런싱 디바이스들, 비디오 스트리밍 디바이스들 등을 포함한, 광범위한 디바이스들에 통합될 수 있다. 디지털 비디오 디바이스들은 ITU-T H.261, ISO/IEC MPEG-1 Visual, ITU-T H.262 또는 ISO/IEC MPEG-2 Visual, ITU-T H.263, ITU-T H.264/MPEG-4, Part 10, AVC (Advanced Video Coding), ITU-T H.265/HEVC (High Efficiency Video Coding) 에 의해 정의된 표준들, 그리고 스케일러블 비디오 코딩 (SVC) 및 멀티뷰 비디오 코딩 (MVC) 과 같은 그러한 표준들의 확장들에 기재된 것들과 같은 비디오 코딩 기법들을 구현한다. 비디오 디바이스들은 그러한 비디오 코딩 기술들을 구현함으로써 디지털 비디오 정보를 더 효율적으로 송신, 수신, 인코딩, 디코딩, 및/또는 저장할 수도 있다.

[0004] 비디오 코딩 기법들은 비디오 시퀀스들에 내재한 리던던시를 감소 또는 제거하기 위해 공간 (인트라-픽처) 예측 및/또는 시간 (인터-픽처) 예측을 포함한다. 블록 기반 비디오 코딩을 위해, 비디오 슬라이스 (예를 들어, 비

[0005] 디오 화상, 또는 비디오 화상의 일부) 는 비디오 블록들로 파티셔닝될 수도 있고, 이 비디오 블록들은 코딩 트리 유닛 (CTU), 코딩 유닛 (CU) 들, 및/또는 코딩 노드들로도 지칭될 수도 있다. 픽처의 인트라-코딩된 (I) 슬라이스에서의 비디오 블록들은 동일 픽처의 이웃하는 블록들에서의 참조 샘플들에 대한 공간 예측을 이용하여 인코딩된다. 픽처의 인터-코딩된 (P 또는 B) 슬라이스에서의 비디오 블록들은 동일 픽처의 이웃하는 블록들에서의 참조 샘플들에 대한 공간 예측, 또는 다른 참조 픽처들에서의 참조 샘플들에 대한 시간 예측을 이용할 수도 있다. 픽처들은 프레임들로서 지칭될 수도 있으며, 참조 픽처들은 참조 프레임들로 지칭될 수도 있다.

[0006] 공간 또는 시간 예측은 코딩될 블록에 대한 예측성 블록을 발생시킨다. 잔차 데이터는 코딩될 원래의 블록과 예측성 블록 간의 픽셀 차이들을 나타낸다. 인터-코딩된 블록은 예측 블록을 형성하는 참조 샘플들의 블록을 포인팅하는 모션 벡터에 따라 인코딩되고, 잔차 데이터는 코딩된 블록과 예측 블록 간의 차이를 나타낸다. 인트라-코딩된 블록은 인트라-코딩 모드 및 잔차 데이터에 따라 인코딩된다. 추가의 압축을 위하여, 잔차 데이터는 픽셀 도메인으로부터 변환 도메인으로 변환되어 잔차 변환 계수들을 낳을 수도 있고, 그 후 이들은 양자화될 수도 있다. 초기에 2 차원 어레이로 배열된 양자화된 변환 계수들은 변환 계수들의 1 차원 벡터를 생성하기 위하여 스캔될 수도 있고, 엔트로피 코딩이 훨씬 더 많은 압축을 달성하기 위하여 적용될 수도 있다.

발명의 내용

과제의 해결 수단

[0007] 일반적으로, 본 개시는 디코더 측 모션 벡터 도출 (DMVD)과 관련된 기술을 설명한다. 예를 들어, 본 개시는 DMVD에 의해 도출된 모션 벡터 (MV), 및/또는 초기 MV 와 DMVD에 의해 도출된 MV 간의 MV 차이와 같은 모션 정보에 하나 이상의 제약들을 적용하는 것과 관련된 기법들을 설명한다. 이들 기법들은 고효율 비디오 코딩 (High Efficiency Video Coding; HEVC) 과 같은 임의의 기존 비디오 코덱들에 적용될 수 있거나, 임의의 장래 비디오 코딩 표준들에서 효율적인 코딩 툴일 수도 있다. 제약이 DMVD로 도출된 MV 및/또는 MV 차이에 적용될 때, 특정 예들에서, 제약을 만족하는 도출된 모션 정보 만이 유효한 모션 정보로 간주된다. 따라서, 코딩 효율을 증가시키고 및/또는 비트 레이트를 감소시킬 수 있는 제약이 사용되지 않으면 MV 들의 리스트 및 최종 선택된 MV 는 상이할 수 있다. 일부 예에서, 제약은 도출된 MV 가 대칭인지, 초기 MV 와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 대칭인지, 도출된 MV 가 반대칭인지, 초기 MV 와 DMVD 방법으로 도출된 MV 간의 MV 차이가 반대칭인지 여부를 포함한다. 일부 예들에서, 대칭/반대칭 MV/MV 차이 제약의 인에이블링은 명시적으로 시그널링된다. 다른 예들에서, 제약들은 일부 코딩된 정보에 따라 암시 적으로 결정될 수 있다. 다른

예들에서, 제약이 특정 예들에만 적용되도록 조건들이 제약들 상에 배치될 수도 있다.

[0008] 일 예에서, 인코딩 된 비디오 비트 스트림으로부터 비디오 데이터를 디코딩하는 방법으로서, 그 방법은 디코더 측 모션 벡터 도출 (DMVD)을 이용하여 복수의 도출된 모션 벡터들 생성하는 단계; 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 제 1 도출된 모션 벡터와 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다고 결정하는 단계를 포함하는, 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터들로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 단계로서, 상기 관계는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭인 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭 모션 벡터 차이를 가지는 것, 또는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 반대칭 모션 벡터 차이를 가지는 것을 포함하는, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 단계; 및 결정된 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 디코딩하는 단계를 포함한다.

[0009] 다른 예에서, 비디오 데이터를 인코딩하는 방법으로서, 그 방법은 디코더 측 모션 벡터 도출 (DMVD)을 이용하여 복수의 도출된 모션 벡터들을 생성하는 단계; 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 제 1 도출된 모션 벡터와 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다고 결정하는 단계를 포함하는, 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터들로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 단계로서, 상기 관계는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭인 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭 모션 벡터 차이를 가지는 것, 또는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 반대칭 모션 벡터 차이를 가지는 것을 포함하는, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 단계; 및 결정된 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 인코딩하는 단계를 포함한다.

[0010] 다른 예에서, 인코딩된 비디오 비트 스트림으로부터 비디오 데이터를 디코딩하도록 구성된 장치로서, 상기 장치는 비디오 데이터를 저장하도록 구성된 메모리; 및 메모리와 통신하는 하나 이상의 프로세서들을 포함하고, 하나 이상의 프로세서들은, 디코더 측 모션 벡터 도출 (DMVD)을 이용하여 복수의 도출된 모션 벡터들을 생성하고; 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 제 1 도출된 모션 벡터와 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다는 결정을 포함하는, 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터들로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 것으로서, 상기 관계는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭인 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭 모션 벡터 차이를 가지는 것, 또는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 반대칭 모션 벡터 차이를 가지는 것을 포함하는, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하며; 및 결정된 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 디코딩하도록 구성된다.

[0011] 다른 예에서, 비디오 데이터를 인코딩하도록 구성된 장치로서, 그 장치는 비디오 데이터를 저장하도록 구성된 메모리, 및 메모리와 통신하는 하나 이상의 프로세서들을 포함하고, 하나 이상의 프로세서들은, 디코더 측 모션 벡터 도출 (DMVD)을 이용하여, 복수의 도출된 모션 벡터들을 생성하고; 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 제 1 도출된 모션 벡터와 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다는 결정을 포함하는, 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터들로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 것으로서, 상기 관계는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭인 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭 모션 벡터 차이를 가지는 것, 또는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 반대칭 모션 벡터 차이를 가지는 것을 포함하는, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하며; 및 결정된 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 디코딩하도록 구성된다.

[0012] 다른 예에서, 인코딩된 비디오 비트 스트림으로부터 비디오 데이터를 디코딩하도록 구성된 장치로서, 상기 장치는, 디코더 측 모션 벡터 도출 (DMVD)을 이용하여 복수의 도출된 모션 벡터들을 생성하는 수단; 제 1 도출된 모

선 벡터 및 제 2 도출된 모션 벡터가 제 1 도출된 모션 벡터와 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다고 결정하는 수단을 포함하는, 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터들로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 수단으로서, 상기 관계는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭인 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭 모션 벡터 차이를 가지는 것, 또는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 반대칭 모션 벡터 차이를 가지는 것을 포함하는, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 수단; 및 결정된 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 디코딩하는 수단을 포함한다.

[0013] 다른 예에서, 비디오 데이터를 인코딩하도록 구성된 장치로서, 장치는, 디코더 측 모션 벡터 도출 (DMVD)을 이용하여 복수의 도출된 모션 벡터들을 생성하는 수단; 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 제 1 도출된 모션 벡터와 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다고 결정하는 수단을 포함하는, 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터들로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 수단으로서, 상기 관계는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭인 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭 모션 벡터 차이를 가지는 것, 또는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 반대칭 모션 벡터 차이를 가지는 것을 포함하는, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 수단; 및 결정된 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 인코딩하는 수단을 포함한다.

[0014] 다른 예에서, 본 개시는 명령들을 저장하는 컴퓨터 판독가능 저장 매체를 기술하며, 그 명령들은 실행될 때 디바이스의 하나 이상의 프로세서들로 하여금, 디코더 측 모션 벡터 도출 (DMVD)을 이용하여 복수의 도출된 모션 벡터들을 생성하게 하고; 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 제 1 도출된 모션 벡터와 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다는 결정을 포함하는, 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터들로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하게 하는 것으로서, 상기 관계는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭인 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭 모션 벡터 차이를 가지는 것, 또는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 반대칭 모션 벡터 차이를 가지는 것을 포함하는, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하게 하며; 및 결정된 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 디코딩하게 한다.

[0015] 다른 예에서, 본 개시는 명령들을 저장하는 컴퓨터 판독가능 저장 매체를 기술하며, 그 명령들은, 실행될 때 디바이스의 하나 이상의 프로세서들로 하여금, 디코더 측 모션 벡터 도출 (DMVD)을 이용하여, 복수의 도출된 모션 벡터들을 생성하게 하고; 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 제 1 도출된 모션 벡터와 제 2 도출된 모션 벡터 사이의 관계에 대한 적어도 하나의 제약을 만족시킨다는 결정을 포함하는, 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터들로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하게 하는 것으로서, 상기 관계는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 대칭 모션 벡터 차이를 가지는 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭인 것, 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭 모션 벡터 차이를 가지는 것, 또는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 반대칭 모션 벡터 차이를 가지는 것을 포함하는, 상기 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하게 하며; 및 결정된 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 이용하여 현재 블록을 양방향 예측을 이용하여 디코딩하게 한다.

[0016] 하나 이상의 예들의 상세들은 첨부 도면들 및 이하의 설명에서 제시된다. 다른 특징, 목적 및 이점들은 상세한 설명 및 도면, 그리고 청구항들로부터 분명해질 것이다.

도면의 간단한 설명

[0017] 도 1은 본 개시의 디코더 측 모션 벡터 도출 (DMVD)을 수행하기 위한 기법들을 이용할 수도 있는 예시적인 비

디오 인코딩 및 디코딩 시스템을 나타내는 블록도이다.

도 2 은 본 개시의 DMVD 를 수행하기 위한 기법들을 구현할 수도 있는 비디오 인코더의 일 예를 도시한 블록 다이어그램이다.

도 3 은 본 개시의 DMVD 를 수행하기 위한 기법들을 구현할 수도 있는 비디오 디코더의 일 예를 도시한 블록 다이어그램이다.

도 4a 및 도 4b 는 이웃 블록들로부터 도출된 공간 모션 벡터 후보를 나타내는 개념도이다.

도 5a 및 도 5b 는 TMVP (temporal motion vector predictor) 후보에 대한 1 차 블록 위치를 나타내는 개념도이다.

도 6 은 현재 블록의 모션 정보를 도출하기 위한 양방향 매칭에 관한 개념을 나타내는 개념도이다.

도 7 은 현재 블록의 모션 정보를 도출하기 위한 템플릿 매칭에 관한 개념을 나타내는 개념도이다.

도 8 은 예시적인 FRUC (frame-rate up-conversion) 템플릿 매칭 프로세스를 나타내는 흐름도이다.

도 9 는 도 8 의 FRUC 템플릿 매칭 프로세스에 대한 예시의 제안된 변경들을 나타내는 흐름도이다.

도 10 은 양방향 광학 흐름의 예와 관련된 개념을 나타내는 개념도이다.

도 11 은 8x4 블록에 대한 그래디언트 계산의 일 예를 도시한 개념적 다이어그램이다.

도 12 는 양방향 템플릿 매칭에 기초한 DMVD 와 관련된 개념을 나타내는 개념도이다.

도 13a 및 도 13b 는 중첩 블록 모션 보상 (OBMC)과 관련된 개념을 나타내는 개념도이다.

도 14a 내지 14d 는 OBMC 가중화를 나타내는 개념도이다.

도 15 은 현재 블록의 모션 정보를 도출하기 위한 양방향 매칭에 관한 개념을 나타내는 개념도이다.

도 16 은 본 개시에 기술된 기법들에 따른 비디오 데이터를 디코딩하는 예시적인 방법을 나타내는 플로우차트이다.

도 17 은 본 개시에 기술된 기법들에 따른 비디오 데이터를 인코딩하는 예시적인 방법을 나타내는 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

[0018] 본 개시는 디코더 측 모션 벡터 도출 (DMVD)과 관련된 기술을 설명한다. 본 개시의 기법들은 임의의 미래의 비디오 코딩 표준에서 효율적인 코딩 툴로서 사용될 수도 있다.

[0019] 일반적으로, 본 개시는 비디오 데이터의 블록들을 인코딩하기 위해 사용되는 모션 정보 (예를 들어, 하나 이상의 모션 벡터, 모션 벡터 및/또는 모션 벡터 차이 정밀도, 및 하나 이상의 참조 화상 인덱스) 를 도출하기 위한 및 비디오 인코더에 의해 결정된 것과 동일한 모션 정보를 비디오 디코더에 의해 결정하기 위한 기법들을 기술한다. 비디오 인코더에 의해 사용된 동일한 (또는 상호의) 도출 기법들을 사용하여, 비디오 디코더는, 모션 정보를 특정적으로 식별하는 임의의 신택스 엘리먼트를 수신하지 않고, 비디오 데이터를 인코딩하는데 어떤 모션 정보가 사용되었는지를 결정할 수도 있다. 그러나, 일부 코딩 시나리오에서, 비디오 인코더는 블록을 인코딩하는데 사용된 모션 정보를 여전히 명시적으로 시그널링 할 수도 있다.

[0020] 본 개시의 다양한 기법들이 비디오 코더에 관하여 설명될 수도 있고, 이 비디오 코더는 비디오 인코더 또는 비디오 디코더 중 어느 일방을 지칭할 수 있는 일반적인 용어이도록 의도된다. 달리 명시적으로 진술되지 않는 한, 비디오 인코더 또는 비디오 디코더에 관해 설명된 기법들은 비디오 인코더 또는 비디오 디코더의 타방에 의해 수행될 수 없다고 가정되어서는 아니된다. 예를 들어, 많은 경우들에서, 비디오 디코더는 인코딩된 비디오 데이터를 디코딩하기 위해 비디오 인코더와 동일한, 또는 대로는 상호 역의 코딩 기법을 수행한다. 많은 경우들에서, 비디오 인코더는 또한 비디오 디코딩 루프를 포함하고, 따라서 비디오 인코더는 비디오 데이터를 인코딩하는 것의 일부로서 비디오 디코딩을 수행한다. 따라서, 달리 진술되지 않는 한, 비디오 인코더에 관해 이 개시물에서 기술된 기법들은 또한 비디오 인코더에 의해 수행될 수도 있고, 그 역도 가능하다.

[0021] 본 개시는 또한 현재 계층, 현재 블록, 현재 픽처, 현재 슬라이스 등과 같은 용어들을 사용한다. 이 개시물의 맥락에서, "현재 (current)" 라는 용어는, 예를 들어, 이전에 코딩된 계층, 블록, 픽처, 및 슬라이스 또는 아직

코딩되지 않을 블록, 픽처, 및 슬라이스에 대한 반대로서, 현재 코딩되고 있는 계층, 블록, 픽처, 슬라이스 등을 식별하도록 의도된다.

[0022] 그 범위 확장, 멀티뷰 확장 (MV-HEVC) 및 스케일러블 확장 (SHVC) 을 포함하는, 새로운 비디오 코딩 표준, 즉 고효율 비디오 코딩 (High Efficiency Video Coding; HEVC) (ITU-T H.265 로서도 지칭됨) 이 ITU-T 비디오 코딩 전문가 그룹 (VCEG) 및 ISO/IEC 동화상 전문가 그룹 (MPEG) 의 비디오 코딩에 관한 공동 협력 팀 (JCT-VC) 및 3D 비디오 코딩 확장 개발에 관한 공동 협력 팀 (JCT-3V) 에 의해 개발되었다. 이하 HEVC WD로 지칭되는 HEVC 사양은 phenix.int-evry.fr/jct/doc_end_user/documents/14_Vienna/wg11/JCTVC-N1003-v1.zip 으로부터 입수 가능하다.

[0023] ITU-T VCEG (Q6/16) 및 ISO/IEC MPEG (JTC 1/SC 29/WG 11) 는, (스크린 콘텐츠 코딩 및 하이-다이내믹-레인지 코딩을 위한 그것의 현재의 확장들 및 단기 확장들을 포함하는) 현재의 HEVC 표준을 현저하게 초과하는 압축 능력을 갖는 미래의 비디오 코딩 기술의 표준화에 대한 잠재적인 필요성을 지금 연구하고 있다. 그 그룹들은 이 영역에서 그들의 전문가들에 의해 제안된 압축 기술 설계들을 평가하기 위해 공동 비디오 개발 팀 (Joint Video Exploration Team; JVET) 으로서 알려진 공동 협력 노력에서 이 탐구 활동에 대해 함께 작업하고 있다. JVET 는 2015년 10월 19-21 일 동안 처음 만났다. 참조 소프트웨어의 버전, 즉 JEM 5 (Joint Exploration Model 5) 는 jvet.hhi.fraunhofer.de/svn/svn_HMJEMSoftware/tags/HM-16.6-JEM-5.0 에서 이용가능하다. JEM 5 의 알고리즘 디스크립션은 phenix.it-sudparis.eu/jvet/doc_end_user/current_document.php?id=2714 에서 이용가능하다.

[0024] 도 1 은 본 개시의 DMVD 를 수행하기 위한 기법들을 이용할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (10) 을 나타내는 블록도이다. 도 1 에 도시된 바와 같이, 시스템 (10) 은, 목적지 디바이스 (14) 에 의해서 나중 시간에 디코딩될 인코딩된 비디오 데이터를 제공하는 소스 디바이스 (12) 를 포함한다. 특히, 소스 디바이스 (12) 는 비디오 데이터를, 컴퓨터 판독가능 매체 (16) 를 통해 목적지 디바이스 (14) 에 제공한다. 소스 디바이스 (12) 및 목적지 디바이스 (14) 는, 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 태블릿 컴퓨

[0025] 터들, 셋톱 박스들, 전화기 핸드셋 이블테면 소위 "스마트" 폰들, 소위 "스마트" 패드, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 재생기들, 비디오 게이밍 콘솔들, 비디오 스트리밍 디바이스 등을 포함한, 광범위한 디바이스들 중 어느 것을 포함할 수도 있다. 일부 경우에, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 무선 통신을 위해 갖추어질 수도 있다.

[0026] 목적지 디바이스 (14) 는, 컴퓨터 판독 가능 매체 (16) 를 통해 디코딩될 인코딩된 비디오 데이터를 수신할 수도 있다. 컴퓨터 판독 가능 매체 (16) 는, 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 이동시킬 수 있는 임의의 유형의 매체 또는 디바이스를 포함할 수도 있다. 일례에서, 컴퓨터 판독 가능 매체 (16) 는, 소스 디바이스 (12) 로 하여금 실시간으로 목적지 디바이스 (14) 로 직접, 인코딩된 비디오 데이터를 송신할 수 있게 하기 위한 통신 매체를 포함할 수도 있다. 인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라 변조되고, 목적지 디바이스 (14) 로 송신될 수도 있다. 통신 매체는 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적 송신 라인들과 같은 임의의 무선 또는 유선 통신 매체를 포함할 수도 있다. 통신 매체는 로컬 영역 네트워크, 광역 네트워크, 또는 인터넷과 같은 글로벌 네트워크와 같은 패킷 기반 네트워크의 부분을 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로의 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0027] 일부 예들에 있어서, 인코딩된 데이터는 출력 인터페이스 (22) 로부터 저장 디바이스로 출력될 수도 있다. 유사하게, 인코딩된 데이터는 입력 인터페이스에 의해 저장 디바이스로부터 액세스될 수도 있다. 저장 디바이스는, 하드 드라이브, 블루레이 디스크, DVD, CD-ROM, 플래시 메모리, 휘발성 또는 비휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체 등의 다양한 분산형 또는 로컬적으로 액세스되는 데이터 저장 매체 중 어느 것을 포함할 수도 있다. 다른 예에서, 저장 디바이스는, 소스 디바이스 (12) 에 의해 생성되는 인코딩된 비디오를 저장할 수도 있는, 파일 서버 또는 또 다른 중간 저장 디바이스에 대응할 수도 있다. 목적지 디바이스 (14) 는, 스트리밍 또는 다운로드를 통해 저장 디바이스로부터 저장된 비디오 데이터에 액세스할 수도 있다. 파일 서버는, 인코딩된 비디오 데이터를 저장하고 그 인코딩된 비디오 데이터를 목적지 디바이스 (14) 로 송신할 수 있는 임의의 타입의 서버일 수도 있다. 예시적인 파일 서버들은, (예를 들어, 웹사이트용) 웹 서버, FTP 서버, NAS (network attached storage) 디바이스, 또는 로컬 디스크 드라이브를 포함한다. 목적지 디바이스 (14) 는, 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해 인코딩된

비디오 데이터에 액세스할 수도 있다. 이것은, 파일 서버 상에 저장된 인코딩된 비디오 데이터를 액세스하는데 적합한 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀 등), 또는 양자의 조합을 포함할 수도 있다. 저장 디바이스로부터의 인코딩된 비디오 데이터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이들의 조합일 수도 있다.

[0028] 본 개시의 기법들은 무선 애플리케이션들 또는 세팅들에 반드시 한정되는 것은 아니다. 그 기법들은, 공중 경유 (over-the-air) 텔레비전 브로드캐스트, 케이블 텔레비전 송신, 위성 텔레비전 송신, DASH (dynamic adaptive streaming over HTTP) 와 같은 인터넷 스트리밍 비디오 송신, 데이터 저장 매체 상에 인코딩되는 디지털 비디오, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션 등의 다양한 멀티미디어 애플리케이션들 중 어느 것을 지원하는 비디오 코딩에 적용될 수도 있다. 일부 예들에서, 시스템 (10) 은, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅 및/또는 화상 통화 등의 애플리케이션들을 지원하기 위하여 일방향 또는 양방향 비디오 송신을 지원하도록 구성될 수도 있다.

[0029] 도 2 의 예에 있어서, 도 1 의 예에서, 소스 디바이스 (12) 는 비디오 소스 (18), 비디오 인코더 (20), 및 출력 인터페이스 (22) 를 포함한다. 목적지 디바이스 (14) 는, 입력 인터페이스 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32) 를 포함한다. 다른 예들에서, 소스 디바이스와 목적지 디바이스가 다른 컴포넌트들 또는 배열체들을 포함할 수도 있다. 예를 들어, 소스 디바이스 (12) 는 외부 카메라와 같은 외부 비디오 소스 (18) 로부터 비디오 데이터를 수신할 수도 있다. 유사하게, 목적지 디바이스 (14) 는 통합된 디스플레이 디바이스를 포함하는 것보다는 외부 디스플레이 디바이스와 인터페이싱할 수도 있다. 본 개시에 따르면, 소스 디바이스 (12) 의 비디오 인코더 (20) 및/또는 목적지 디바이스 (14) 의 비디오 디코더 (30) 는 본 명세서에 설명된 하나 이상의 DMVD 기법들을 수행하도록 구성될 수도 있다. 예를 들어, 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 는 (1) DMVD 에 의해 도출된 MV 및/또는 (2) 초기 MV 와 DMVD 에 의해 도출된 MV 사이의 MV 차이에 하나 이상의 제약들을 적용하도록 구성될 수도 있다. 제약은 DMVD 프로세스에 의해 유도된 모션 벡터에 대한 제한을 포함할 수도 있다. 본 개시의 예시적인 제약은 두 개의 참조 픽처 리스트 (예를 들어, List0 및 List1) 에서 MV 와 MVD 사이의 일부 미리 정의된 관계를 규정하는 제약들을 포함할 수 있다. 제약이 DMVD 에 적용될 때, 특정 예들에서, 제약을 만족하는 도출된 모션 정보 만이 유효한 모션 정보로 간주된다. 따라서, 제약이 사용되지 않으면 MV 들의 리스트 및 최종 선택된 MV 는 상이할 수도 있다. 제 1 예에서, 제약은 도출된 MV 가 대칭인지 여부를 포함한다. 제 2 예에서, 제약은 초기 MV와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 대칭인지 여부를 포함한다. 제 3 예에서, 제약은 도출된 MV 가 반대칭인지 여부를 포함한다. 제 4 예에서, 제약은 초기 MV 와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 반대칭인지 여부를 포함한다. 일부 예들에서, 대칭/반대칭 MV/MV 차이 제약의 인에이블링은 명시적으로 시그널링된다. 다른 예들에서, 제약들은 일부 코딩된 정보에 따라 암시 적으로 결정될 수 있다.

[0030] 다른 예들에서, 제약이 특정 예들에만 적용되도록 조건들이 제약들 상에 배치될 수도 있다. 조건들은 제약이 인에이블되는시기에 대한 제한을 포함할 수도 있다. 예를 들어 (1) 제약은 초기 MV가 대칭 인 경우에만 가능하다. (2) 제약은 초기 MV 가 의사 대칭일 때만 가능하다; (3) 제약은 초기 MV가 반대칭일 때만 가능하다; (4) 제약은 초기 MV가 의사 반대칭일 때에만 가능하다; (5) 제약은 초기 MV가 대칭이 아닌 경우에만 가능하다. (6) 제약은 초기 MV가 의사 대칭이 아닌 경우에만 가능하다; (7) 제약은 초기 MV가 반대칭이 아닌 경우에만 가능하다; (8) 제약은 초기 MV가 의사 반대칭이 아닌 경우에만 가능하다; (9) 제약은 초기 MV의 List0 및 List1 참조 픽처들이 현재 픽처의 모두 앞에 또는 모두 뒤에 있을 때만 가능하다; (10) 제약은 초기 MV의 List0 및 List1 참조 픽처가 현재 픽처의 모두 앞에 또는 모두 뒤에 없는 경우에만 가능하다; (11) 제약은 초기 MV의 참조 인덱스가 모두 0 일 때만 가능하다. (12) 제약은 초기 MV의 참조 인덱스가 모두 0 이 아닌 경우에만 가능하다; (13) 제약은 리스트 0 참조 픽처와 현재 픽처 사이의 POC 거리와 리스트 1 참조 픽처와 현재 픽처 사이의 POC 거리가 동일할 때만 가능하다; (14) 제약은 리스트 0 참조 픽처와 현재 픽처 사이의 POC 거리와 리스트 1 참조 픽처와 현재 픽처 사이의 POC 거리가 동일하지 않은 경우에만 가능하다; (15) 대칭 MV 차이의 제약은 초기 MV가 대칭이 아닌 경우에 적용된다; 및/또는 (16) 반대칭 MV 차이의 제약은 초기 MV 가 대칭일 때 적용된다.

[0031] 다른 예에서, 대칭 제약의 MV 해상도는 결정론적 방식으로 할당될 수 있다. 일 예에서, 양방향 템플릿 매칭의 정수-픽셀 정제의 경우, 대칭/의사 대칭/비대칭 제약이 부과되지 않아야하며, 전술 한 제약은 단지 하프-픽셀, 1/4-픽셀 또는 고정밀 모션 정제에 대한 것이다. 제약의 수준 (MV 해상도)은 SPS/PPS/슬라이스 헤더를 통해 시그널링될 수 있다. 해상도는 또한 BIO, 서브-픽셀 프레임-레이트 상향 변환 (FRUC) 과 같은 다른 모션 정제 도구가 가능하게 되는지 여부와 함께 결정될 수 있다. 예를 들어, BIO 가 가능하게 되는 경우, 제약은 1/4-픽셀 또는 더 높은-정밀도 모션 정제에 부과되지 않아야 한다.

- [0032] 다른 예에서, 제약의 수준은 List0 (또한 L0으로 지칭 됨) 과 List1 (L1로 지칭 됨) MV 사이의 절대 차, 스케일링된 (상대 POC 거리에 기초하여) List0 및 List1 MV 사이의 절대 차, 또는 List0 및 List1 의 보간된 샘플들 사이의 초기 SAD 값 (즉, 이 예에서는 P0 및 P1) 에 따라 적응적이게 될 수 있다. 대안적으로, 현재 템플릿, P0 과 현재 템플릿, P1 사이의 SAD 값의 비율은 어떤 MV 해상도에서 제약이 부과되어야 하는지를 결정하는데 사용될 수 있다. 이 예에서, SAD 값의 비가 임계 값보다 낮으면, 전술한 대칭 제약이 부과되지 않아야 한다.
- [0033] 또한, 일부 예들에서, 단일 제약 (예를 들어, 대칭 MV, 대칭 MV 차이, 반대칭 MV, 또는 반대칭 MV 차이) 이 조건에 기초하여 도출된 MV 에 적용된다. 다른 예에서, 다수의 제약은 다수의 조건에 기초하여 도출된 MV 에 적용된다.
- [0034] 도 1 의 예시된 시스템 (10) 은 단지 일 예이다. 본 개시의 DMVD 를 수행하기 위한 기법들은 임의의 디지털 비디오 인코딩 및/또는 디코딩 디바이스에 의해 수행될 수도 있다. 예를 들어, (1) DMVD에 의해 유도된 MV 및/또는 (2) 초기 MV와 DMVD에 의해 유도된 MV 간의 MV 차이를 제한하는 기법은 임의의 디지털 비디오 인코딩 및/또는 디코딩 디바이스에 의해 수행될 수도 있다. 일반적으로 본 개시의 기법들은 비디오 인코딩 디바이스에 의해 수행되지만, 그 기법들은 또한 "코덱 (CODEC)" 으로서 통상적으로 지칭되는, 비디오 인코더/디코더에 의해 수행될 수도 있다. 더욱이, 본 개시의 기법들은 또한 비디오 프리프로세서에 의해 수행될 수도 있다. 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 소스 디바이스 (12) 가 목적지 디바이스 (14) 로의 송신을 위한 코딩된 비디오 데이터를 생성하는 이러한 코딩 디바이스들의 단지 예들에 불과하다. 일부 예들에 있어서, 디바이스들 (12, 14) 은, 디바이스들 (12, 14) 각각이 비디오 인코딩 및 디코딩 컴포넌트들을 포함하도록 실질적으로 대칭적인 방식으로 동작할 수도 있다. 따라서, 시스템 (10) 은 예를 들어, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 또는 비디오 전화를 위해, 비디오 디바이스들 (12, 14) 간의 일방향 또는 양방향 비디오 송신을 지원할 수도 있다.
- [0035] 소스 디바이스 (12) 의 비디오 소스 (18) 는 비디오 카메라와 같은 비디오 캡처 디바이스, 이전에 캡처된 비디오를 포함하는 비디오 아카이브 (video archive), 및/또는 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 피드 인터페이스 (video feed interface) 를 포함할 수도 있다. 추가적인 대안으로서, 비디오 소스 (18) 는 컴퓨터 그래픽 기반 데이터를 소스 비디오로서, 또는 라이브 비디오, 아카이브된 비디오, 및 컴퓨터 생성된 비디오의 조합으로서 생성할 수도 있다. 일부 경우들에서, 비디오 소스 (18) 가 비디오 카메라이면, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 소위 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 하지만, 위에서 언급된 바처럼, 본 개시에 설명된 기법들은, 일반적으로 비디오 코딩에 적용가능할 수도 있고, 무선 및/또는 유선 애플리케이션들에 적용될 수도 있다. 각각의 경우에 있어서, 캡처된, 사전-캡처된 또는 컴퓨터 생성된 비디오는 비디오 인코더 (20) 에 의해 인코딩될 수도 있다. 다음으로, 인딩된 비디오 정보는 컴퓨터 판독가능 매체 (16) 상으로 출력 인터페이스 (22) 에 의해 출력될 수도 있다.
- [0036] 컴퓨터 판독 가능 매체 (16) 는, 무선 브로드캐스트 또는 유선 네트워크 송신 등의 일시적 매체, 또는 하드 디스크, 플래시 드라이브, 콤팩트 디스크, 디지털 비디오 디스크, 블루레이 디스크 또는 다른 컴퓨터 판독 가능 매체 등의 저장 매체 (즉, 비일시적 저장 매체) 를 포함할 수도 있다. 일부 예들에서, 네트워크 서버 (미도시) 는 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 수신하고, 인코딩된 비디오 데이터를, 예를 들어, 네트워크 송신을 통해 목적지 디바이스 (14) 에 제공할 수도 있다. 유사하게, 디스크 스템핑 설비와 같은 매체 생성 설비의 컴퓨팅 디바이스는 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 수신하고, 인코딩된 비디오 데이터를 포함하는 디스크를 생성할 수도 있다. 따라서, 컴퓨터 판독가능 매체 (16) 는, 다양한 예들에 있어서, 다양한 형태들의 하나 이상의 컴퓨터 판독가능 매체들을 포함하는 것으로 이해될 수도 있다.
- [0037] 목적지 디바이스 (14) 의 입력 인터페이스 (28) 는 컴퓨터 판독 가능 매체 (16) 로부터 정보를 수신한다. 컴퓨터 판독 가능 매체 (16) 의 정보는 비디오 인코더 (20) 에 의해 정의된 선택스 정보를 포함할 수도 있고, 이는 또한 비디오 디코더 (30) 에 의해 사용되고, 블록들 및 다른 코딩된 유닛들의 처리 및/또는 특성들을 기술하는 선택스 엘리먼트들을 포함한다. 디스플레이 디바이스 (32) 는 디코딩된 비디오 데이터를 이용자에게 디스플레이하고, 음극선관 (CRT), 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 다른 유형의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 어느 것도 포함할 수 있다.
- [0038] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 ITU-T H.265 로도 지칭되는, 고 효율 비디오 코딩 (HEVC) 표준과 같은 비디오 코딩 표준에 따라 동작될 수도 있다. 대안적으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 는, 대안적으로 MPEG-4, Part 10, AVC (Advanced Video Coding) 으로도 지칭되는, ITU-T H.264 표준과 같은 다른

독자적 또는 산업 표준들 또는 그러한 표준들의 확장들에 따라 동작할 수도 있다. 하지만, 본 개시의 기법들은 임의의 특정 코딩 표준에 한정되지 않는다. 비디오 코딩 표준들의 다른 예들은 MPEG-2 및 ITU-T H.263 을 포함한다. 도 1 에 도시되지는 않았지만, 일부 양태들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 오디오 인코더 및 디코더와 각각 통합될 수도 있으며, 오디오 및 비디오 양자의 인코딩을 공통 데이터 스트림 또는 별개의 데이터 스트림들로 처리하기 위해 적절한 MUX-DEMUX 유닛들 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능하다면, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜 또는 다른 프로토콜들 이를테면 사용자 데이터그램 프로토콜 (UDP) 을 따를 수도 있다.

[0039] 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 마이크로프로세서들, (고정 기능 회로 및/또는 프로그램가능 프로세싱 회로를 포함하는) 프로세싱 회로, 디지털 신호 프로세서(DSP)들, 주문형 집적회로(ASIC)들, 필드 프로그래머블 게이트 어레이(FPGA)들, 별도의 로직, 소프트웨어, 하드웨어, 펌웨어 또는 이들의 임의의 조합들과 같은 다양한 적절한 인코더 및 디코더 회로 중 임의의 회로로서 구현될 수도 있다. 그 기법들이 부분적으로 소프트웨어로 구현될 때, 디바이스는 적합한 비일시적 컴퓨터 판독가능 매체에 그 소프트웨어를 위한 명령들을 저장하고 본 개시의 기법들을 수행하기 위하여 하나 이상의 프로세서들을 이용하여 하드웨어에서 그 명령들을 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 의 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있는데, 이들 중 어느 일방은 각각의 디바이스에서 결합된 인코더/디코더 (CODEC) 의 부분으로서 통합될 수도 있다. 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 를 포함하는 디바이스는 집적 회로, 마이크로프로세서, 및/또는 무선 통신 디바이스, 이를테면 셀룰러 전화기를 포함할 수도 있다.

[0040] 일반적으로, ITU-T H.265 에 따르면, 비디오 화상은, 루마 및 크로마 샘플들 양자 모두를 포함할 수도 있는 코딩 트리 유닛들 (CTU) (또는 최대 코딩 유닛들 (LCU)) 의 시퀀스로 분할될 수도 있다. 대안적으로, CTU 들은 단색 데이터 (즉, 루마 샘플들만) 를 포함할 수도 있다. 비트스트림 내의 선택스 데이터는 CTU 를 위한 크기를 정의할 수도 있으며, 이는 픽셀들의 수의 측면에서 최대 코딩 유닛이다. 슬라이스는 코딩 순서에서의 다수의 연속적인 CTU 들을 포함한다. 비디오 화상은, 하나 이상의 슬라이스들로 파티셔닝될 수도 있다. 각각의 CTU 는 쿼드트리에 따라 코딩 유닛 (CU) 들로 스플리팅될 수도 있다. 일반적으로, 쿼드트리 데이터 구조는, CTU 에 대응하는 루트 노드와, CU 당 하나의 노드를 포함한다. CU 가 4개의 서브-CU들로 분할되면, CU 에 대응하는 노드는 4개의 리프 (leaf) 노드들을 포함하며, 이들 각각은 서브-CU들 중 하나에 대응한다.

[0041] 쿼드트리 데이터 구조의 각각의 노드는 대응하는 CU에 대해 선택스 데이터를 제공할 수도 있다. 예를 들어, 쿼드트리에서의 노드는 그 노드에 대응하는 CU 가 서브-CU들로 분할되는지 여부를 표시하는 분할 플래그를 포함할 수도 있다. CU 에 대한 선택스 엘리먼트들은 재귀적으로 정의될 수도 있으며, CU 가 서브-CU들로 분할되는지 여부에 의존할 수도 있다. CU 가 추가로 분할되지 않으면, 리프-CU 로서 지칭된다. 본 개시에 있어서, 오리지널 리프-CU 의 명시적인 분할이 존재하지 않더라도, 리프-CU 의 4개의 서브-CU들은 또한 리프-CU 들로서 지칭될 것이다. 예를 들어, 16x16 사이즈의 CU 가 더 분할되지 않으면, 16x16 CU 가 결코 분할되지 않았더라도, 4개의 8x8 서브-CU들은 또한 리프-CU들로서 지칭될 것이다.

[0042] CU 가 크기 구분 (size distinction) 을 가지지 않는다는 것을 제외하고는, CU 는 H.264 표준의 매크로블록 (macroblock) 과 유사한 목적을 가진다. 예를 들어, CTU 는 4 개의 자식 노드 (child node) 들 (또한, 서브-CU 들로서 지칭됨) 로 스플리팅될 수도 있고, 각각의 자식 노드는 차례로 부모 노드 (parent node) 일 수도 있고, 또 다른 4 개의 자식 노드들로 스플리팅될 수도 있다. 쿼드트리의 리프 노드로서 지칭되는 최종의 미분할된 자식 노드는, 리프-CU 로서 또한 지칭되는 코딩 노드를 포함한다. 코딩된 비트스트림과 연관된 선택스 데이터는, 최대 CU 깊이로도 지칭되는, CTU 가 스플리팅될 수도 있는 최대 횟수를 정의할 수도 있고, 또한 코딩 노드들의 최소 크기를 정의할 수도 있다. 이에 따라, 비트스트림은 또한 최소 코딩 유닛 (SCU) 을 정의할 수도 있다. 본 개시는, 용어 "블록" 을 사용하여, HEVC 의 콘텍스트에서, CU, 예측 유닛 (PU), 또는 변환 유닛 (TU) 중 어느 것을 지칭하거나, 또는 다른 표준들의 콘텍스트에서 유사한 데이터 구조들 (예를 들어, H.264/AVC 에서 매크로블록들 및 이들의 서브 블록들) 을 지칭한다.

[0043] CU 는 코딩 노드, 그리고 코딩 노드와 연관된 예측 유닛들 (PU들) 및 변환 유닛들 (TU들) 을 포함한다. CU 의 크기는 코딩 노드의 크기에 대응하고 형상이 대체로 정사각형이다. CU 의 크기는 8x8 픽셀들로부터, 최대 크기, 예를 들어, 64x64 픽셀들 이상인 CTU 의 크기에 이르기까지의 범위일 수도 있다. 각각의 CU 는 하나 이상의 PU들 및 하나 이상의 TU들을 포함할 수도 있다. CU에 연관된 선택스 데이터는, 예를 들어, CU의 하나 이상의 PU들로의 파티셔닝을 기술할 수도 있다. 파티셔닝 모드들은, CU 가 스킵 또는 직접 모드 인코딩되는지, 인트라-예측 모드 인코딩되는지, 또는 인터-예측 모드 인코딩되는지 간에 달라질 수도 있다. PU들은 형상이 비-정사각형이 되도록 파티셔닝될 수도 있다. CU 와 연관된 선택스 데이터는 또한, 예를 들어, 쿼드트리에 따라

하나 이상의 TU들로의 CU 의 파티셔닝을 기술할 수도 있다. TU 는 형상이 정방형이거나 비-정방형 (예를 들어, 직방형) 일 수 있다.

[0044] HEVC 표준은 TU들에 따른 변환들에 대해 허용하며, 이는 상이한 CU들에 대해 상이할 수도 있다. TU 들은 통상적으로, 파티셔닝된 CTU에 대해 정의된 주어진 CU 내에 PU 들 (또는 CU 의 파티션들) 의 크기에 기초하여 사이징되지만, 이는 항상 그렇지 않을 수도 있다. TU 는 전형적으로 PU (또는 예를 들어, 인트라 예측의 경우, CU 의 파티션들) 와 동일한 크기이거나 더 작다. 일부 예들에 있어서, CU 에 대응하는 잔차 샘플들은 잔차 쿼드트리 (RQT) 로서 알려진 쿼드트리 구조를 이용하여 더 작은 유닛들로 세분될 수도 있다. RQT 의 리프 노드들은 변환 유닛 (TU) 들로 지칭될 수도 있다. TU 들과 연관된 픽셀 차이 값들이 변환되어 변환 계수들을 생성하고, 이들은 양자화될 수도 있다.

[0045] 리프-CU 는 인트라 예측을 사용하여 예측될 때 하나 이상의 예측 유닛 (PU) 들을 포함할 수도 있다. 일반적으로, PU 는, 대응하는 CU 의 전부 또는 일부에 대응하는 공간 영역을 나타내고, PU 를 위해 참조 샘플을 취출 및/또는 생성하기 위한 데이터를 포함할 수도 있다. 더욱이, PU 는 예측과 관련된 데이터를 포함한다. CU 가 인트라 모드 인코딩 될 때, CU 의 하나 이상의 PU 는 하나 이상의 모션 벡터와 같은 모션 정보를 정의하는 데이터를 포함할 수도 있거나, PU 는 스킵 모드 코딩될 수도 있다. PU 에 대한 모션 벡터를 정의하는 데이터는 예를 들어, 모션 벡터의 수평 컴포넌트, 모션 벡터의 수직 컴포넌트, 모션 벡터에 대한 분해능 (예를 들어, 1/4 픽셀 정밀도 또는 1/8 픽셀 정밀도), 모션 벡터가 포인팅하는 레퍼런스 픽처, 및/또는 모션 벡터에 대한 참조 픽처 리스트 (예를 들어, List0 또는 List1) 를 기술할 수도 있다.

[0046] 리프 CU 는 또한 인트라 모드 예측될 수도 있다. 일반적으로, 인트라 예측은 인트라 모드를 사용하여 리프 CU (또는 그 파티션) 를 예측하는 것을 포함한다. 비디오 코더는 리프 -CU (또는 그 파티션)를 예측하기 위해 사용할 리프-CU 에 대한 이웃하는, 이전에 코딩된 픽셀들의 세트를 선택할 수도 있다.

[0047] 리프-CU 는 또한 하나 이상의 변환 유닛 (TU) 들을 포함할 수도 있다. 변환 유닛들은, 상기 논의된 바와 같이 RQT (TU 쿼드트리 구조로서도 또한 지칭됨) 를 이용하여 명시될 수도 있다. 예를 들어, 분할된 플래그는 리프-CU 가 4개의 변환 유닛들로 분할되는지 여부를 표시할 수도 있다. 그 후, 각각의 TU 는 추가의 서브-TU 들로 더 분할될 수도 있다. TU 가 더 분할되지 않을 경우, 리프-TU 로서 지칭될 수도 있다. 일반적으로, 인트라 코딩에 대해, 리프-CU 에 속하는 리프-TU들 모두는 동일한 인트라 예측 모드를 공유한다. 즉, 동일한 인트라-예측 모드는 일반적으로, 리프-CU 의 모든 TU들에 대한 예측된 값들을 계산하기 위해 적용된다. 인트라 코딩에 대해, 비디오 인코더는 인트라 예측 모드를 이용하여 각각의 리프-TU 에 대한 잔차 값을, TU 에 대응하는 CU 의 부분과 오리지널 블록 간의 차이로서 계산할 수도 있다. TU 가 반드시 PU 의 사이즈로 한정될 필요는 없다. 따라서, TU들은 PU 보다 더 크거나 더 작을 수도 있다. 인트라 코딩에 대해, CU 의 파티션들, 또는 CU 자신은 CU 에 대한 대응하는 리프-TU 와 병치될 수도 있다. 일부 예들에 있어서, 리프-TU 의 최대 사이즈는 대응하는 리프-CU 의 사이즈에 대응할 수도 있다.

[0048] 또한, 리프-CU 들의 TU 들은 또한, 잔차 쿼드트리 (RQT) 들로 지칭되는, 각각의 쿼드트리 데이터 구조들과 연관될 수도 있다. 즉, 리프-CU 는, 리프-CU가 TU 들로 어떻게 파티셔닝되는지를 나타내는 쿼드트리를 포함할 수도 있다. TU 쿼드트리의 루트 노드는 일반적으로 리프-CU 에 대응하는 한편, CU 쿼드트리의 루트 노드는 일반적으로 CTU (또는 LCU) 에 대응한다. 분할되지 않은 RQT 의 TU들은 리프-TU들로서 지칭된다. 일반적으로, 본 개시는, 다르게 언급되지 않는 한, 리프-CU 및 리프-TU 를 지칭하기 위하여 용어 CU 및 TU 를 각각 사용한다.

[0049] 비디오 시퀀스는 일반적으로 랜덤 액세스 포인트 (RAP) 화상으로 시작하는 일련의 비디오 프레임 또는 화상들을 포함한다. 비디오 시퀀스는 비디오 시퀀스의 특성인 시퀀스 파라미터 세트 (SPS) 에 신택스 데이터를 포함할 수도 있다. 픽처의 각각의 슬라이스는 개별 슬라이스에 대한 인코딩 모드를 기술하는 슬라이스 신택스 데이터를 포함할 수도 있다. 비디오 인코더 (20) 는 통상적으로, 비디오 데이터를 인코딩하기 위하여 개별 비디오 슬라이스들 내의 비디오 블록들에 대해 동작한다. 비디오 블록은 CU 내의 코딩 노드에 대응할 수도 있다. 비디오 블록들은 고정되거나 또는 변화하는 크기를 가질 수도 있고, 명시된 코딩 표준에 따라 크기가 다를 수도 있다.

[0050] 예로서, 다양한 크기의 PU 에 대해 예측이 수행될 수도 있다. 특정 CU 의 크기가 $2N \times 2N$ 이라고 가정하면, 인트라-예측은 $2N \times 2N$ 또는 $N \times N$ 의 PU 크기들 상에서 수행될 수도 있고, 인트라-예측은 $2N \times 2N$, $2N \times N$, $N \times 2N$, 또는 $N \times N$ 의 대칭적 PU 크기들 상에서 수행될 수도 있다. 인트라-예측을 위한 비대칭적 파티셔닝은 또한, $2N \times nU$, $2N \times nD$, $nL \times 2N$, 및 $nR \times 2N$ 의 PU 크기에 대해서 수행될 수도 있다. 비대칭 파티셔닝에 있어서, CU 의 일 방향은 파티셔닝되지 않지만 다른 방향은 25% 및 75% 로 파티셔닝된다. 25% 파티션에 대응하는 CU 의 부분은 “n” 다음에

“상 (Up)”, “하 (Down)”, “좌 (Left)”, 또는 “우 (Right)” 의 표시에 의해 표시된다. 따라서, 예를 들어, $2N \times nU$ 는, 상단의 $2N \times 0.5N$ PU 및 하단의 $2N \times 1.5N$ PU 로 수평적으로 파티셔닝되는 $2N \times 2N$ CU 를 지칭한다.

[0051] 본 개시에서, “ $N \times N$ ” 그리고 “N 바이 N” 은, 수직 및 수평 차원 (dimension) 들의 면에서 비디오 블록의 픽셀 차원들, 예를 들면, 16×16 픽셀들 또는 16 바이 16 픽셀들을 지칭하는데 상호교환가능하게 사용될 수도 있다. 일반적으로, 16×16 블록은, 수직 방향에서 16 픽셀들 ($y = 16$) 그리고 수평 방향에서 16 픽셀들 ($x = 16$) 을 가질 것이다. 마찬가지로, $N \times N$ 블록은 일반적으로 수직 방향에서 N 픽셀들 그리고 수평 방향에서 N 픽셀들을 갖고, 여기서 N 은 음이 아닌 정수 값을 나타낸다. 블록에서 픽셀들은 행과 열들로 배열될 수도 있다. 더욱이, 블록들은 수직 방향과 동일한 수의 수평 방향 픽셀들을 반드시 가질 필요는 없다. 예를 들면, 블록들은 $N \times M$ 픽셀들을 포함할 수도 있고, 여기서 M은 N과 반드시 동일한 것은 아니다.

[0052] CU 의 PU 들을 이용한 인트라-예측 또는 인터-예측 코딩 다음에, 비디오 인코더 (20) 는 CU 의 TU 들을 위한 잔차 데이터를 계산할 수도 있다. PU 들은, (픽셀 도메인으로도 지칭되는) 공간 도메인에서 예측 픽셀 데이터를 생성하는 방법 또는 모드를 기술하는 신택스 데이터를 포함할 수도 있고, TU들은 잔차 비디오 데이터에 대한, 변환, 예를 들어, 이산 코사인 변환 (DCT), 정수 변환, 웨이블릿 변환, 또는 개념적으로 유사한 변환의 적용 후에 변환 도메인에서의 계수들을 포함할 수도 있다. 잔차 데이터는 인코딩되지 않은 픽처의 픽셀들과 PU들에 대응하는 예측 값들 간의 픽셀 차이들에 대응할 수도 있다. 비디오 인코더 (20) 는 CU에 대한 잔차 데이터를 나타내는 양자화된 변환 계수를 포함하도록 TU 를 형성할 수도 있다. 즉, 비디오 인코더 (20)는 (잔차 블록의 형태로) 잔차 데이터를 계산하고, 잔차 블록을 변환하여 변환 계수의 블록을 생성한 다음, 변환 계수를 양자화하여 양자화된 변환 계수를 형성할 수도 있다. 비디오 인코더 (20) 는 양자화된 변환 계수 및 다른 신택스 정보 (예를 들어, TU에 대한 스플리팅 정보) 를 포함하는 TU를 형성할 수도 있다.

[0053] 위에 언급된 바처럼, 변환 계수들을 생성하기 위한 임의의 변환 후에, 비디오 인코더 (20) 는 변환 계수들의 양자화를 수행할 수도 있다. 양자화는 일반적으로, 변환 계수들이 그 계수들을 나타내는데 사용되는 데이터의 양을 가능하게는 감소시키도록 양자화되어 추가 압축을 제공하는 프로세스를 지칭한다. 양자화 프로세스는 그 계수들의 일부 또는 전부에 연관된 비트 깊이를 감소시킬 수도 있다. 예를 들어, n-비트 값은 양자화 동안 m-비트 값으로 라운드-다운될 수도 있으며, 여기서, n 은 m 보다 크다.

[0054] 양자화 후에, 비디오 인코더는 변환 계수들을 스캔하여, 양자화된 변환 계수들을 포함하는 2차원 매트릭스로부터 1차원 벡터를 생성할 수도 있다. 그 스캔은 더 높은 에너지 (그리고 따라서 더 낮은 주파수) 계수들을 어레이의 전방에 두고 더 낮은 에너지 (그리고 따라서 더 높은 주파수) 계수들을 어레이의 후방에 두도록 설계될 수도 있다. 일부 예들에 있어서, 비디오 인코더 (20) 는 양자화된 변환 계수들을 스캔하기 위한 미리정의된 스캔 순서를 활용하여, 엔트로피 인코딩될 수 있는 직렬화된 벡터를 생성할 수도 있다. 다른 예들에서, 비디오 인코더 (20) 는 적응적 스캔을 수행할 수도 있다. 1차원 벡터를 형성하기 위해 양자화된 변환 계수들을 스캔한 이후, 비디오 인코더 (20) 는, 예를 들어, 컨텍스트 적응적 가변 길이 코딩 (CAVLC), 컨텍스트 적응적 바이너리 산술 코딩 (CABAC), 신택스 기반 컨텍스트 적응적 바이너리 산술 코딩 (SBAC), 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩, 또는 다른 엔트로피 인코딩 방법에 따라 1차원 벡터를 엔트로피 인코딩할 수도 있다. 비디오 인코더 (20) 는 또한, 비디오 데이터를 디코딩함에 있어서 비디오 디코더 (30) 에 의한 사용을 위해 인코딩된 비디오 데이터와 연관된 신택스 엘리먼트들을 엔트로피 인코딩할 수도 있다.

[0055] CABAC 을 수행하기 위하여, 비디오 인코더 (20) 는, 송신될 심볼에 컨텍스트 모델 내의 컨텍스트를 배정할 수도 있다. 컨텍스트는, 예를 들어, 심볼의 이웃 값들이 제로가 아닌지 여부와 관련될 수도 있다. CAVLC 을 수행하기 위하여, 비디오 인코더 (20) 는, 송신될 심볼을 위해 가변 길이 코드를 선택할 수도 있다. VLC 에서의 코드워드들은, 상대적으로 더 짧은 코드들이 더 높은 확률 심볼들에 대응하는 한편, 더 긴 코드들이 더 적은 확률 심볼들에 대응하도록 구성될 수도 있다. 이런 식으로, VLC 의 사용은, 예를 들어, 송신될 각 심볼에 동일 길이 코드워드들을 이용하는 것에 비해, 비트 절약 (bit savings) 을 달성할 수도 있다. 확률 결정은, 심볼에 할당된 컨텍스트에 기초할 수도 있다.

[0056] 일반적으로, 비디오 디코더 (30) 는 인코딩된 데이터를 디코딩하기 위해 비디오 인코더 (20) 에 의해 수행되는 프로세스와 실질적으로 유사하지만, 상반되는 프로세스를 수행한다. 예를 들어, 비디오 디코더 (30) 는 잔차 블록을 재생하기 위해 수신된 TU 의 계수를 역 양자화 및 역 변환한다. 비디오 디코더 (30) 는 시그널링된 예측 모드 (인트라- 또는 인터-예측) 를 사용하여 예측된 블록을 형성한다. 그 후, 비디오 디코더 (30) 는 예측된 블록과 잔차 블록을 (픽셀 바이 픽셀에 기반해) 결합하여 원래 블록을 재생한다. 블록 경계를 따라 시각적

아티팩트들을 줄이기 위해 디블록킹 프로세스를 수행하는 것과 같은 추가적인 처리가 수행될 수도 있다. 또한, 비디오 디코더 (30) 는 비디오 인코더 (20) 의 CABAC 인코딩 프로세스와는 실질적으로 유사하지만, 상반되는, 방식으로 CABAC 을 사용하여 선택스 엘리먼트를 디코딩할 수도 있다.

[0057] 비디오 인코더 (20) 는 또한, 선택스 데이터, 이를테면 블록 기반 선택스 데이터, 화상 기반 선택스 데이터, 및 시퀀스 기반 선택스 데이터를, 비디오 디코더 (30) 에, 예를 들어, 화상 헤더, 블록 헤더, 슬라이스 헤더, 또는 다른 선택스 데이터, 이를테면 시퀀스 파라미터 세트 (SPS), 화상 파라미터 세트 (PPS), 또는 비디오 파라미터 세트 (VPS) 에서 전송할 수도 있다.

[0058] 도 2 는 본 개시의 하나 이상의 DMVD 기법들을 수행하도록 구성될 수도 있는 비디오 인코더 (20) 의 예를 나타내는 블록도이다. 예를 들어, 비디오 인코더 (20) 는 (1) DMVD 에 의해 도출된 MV 및/또는 (2) 초기 MV 와 DMVD 에 의해 도출된 MV 사이의 MV 차이에 하나 이상의 제약들을 적용하도록 구성될 수도 있다. 제약이 DMVD 에 적용될 때, 특정 예들에서, 제약을 만족하는 도출된 모션 정보 만이 유효한 모션 정보로 간주된다. 따라서, 제약이 사용되지 않으면 MV 들의 리스트 및 최종 선택된 MV 는 상이할 수도 있다.

[0059] 제 1 예에서, 제약은 도출된 MV 가 대칭인지 여부를 포함한다. 제 2 예에서, 제약은 초기 MV와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 대칭인지 여부를 포함한다. 제 3 예에서, 제약은 도출된 MV 가 반대칭인지 여부를 포함한다. 제 4 예에서, 제약은 초기 MV와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 반대칭인지 여부를 포함한다. 일부 예들에서, 대칭/반대칭 MV/MV 차이 제약의 인에이블링은 명시적으로 시그널링된다. 다른 예들에서, 제약들은 일부 코딩된 정보에 따라 암시 적으로 결정될 수 있다. 다른 예들에서, 제약이 일부 예들에만 적용되도록 조건들이 제약들 상에 배치될 수도 있다.

[0060] 특히, 비디오 인코더 (20) 는 역 양자화 유닛 (58), 역 변환 유닛 (60) 및 합산 기 (62)에 의해 수행되는 프로세스들을 포함하는 재구성 루프 동안 본 개시의 DMVD 기법들을 수행할 수도 있다. 또한, 위에서 논의 된 바와 같이, 비디오 인코더 (20)는 DMVD 를 수행함에 있어서 비디오 디코더 (30) 와 같은 비디오 디코더를 도울 수도 있는 특정 값들을 시그널링할 수도 있다.

[0061] 비디오 인코더 (20) 는 비디오 슬라이스들 내에서 비디오 블록들의 인트라 및 인터 코딩을 수행할 수도 있다. 인트라 코딩은 주어진 비디오 프레임 또는 픽처 내의 비디오에서 공간적 리던던시를 감소시키거나 제거하기 위해 공간 예측에 의존한다. 인터 코딩은 비디오 시퀀스의 인접한 프레임들 또는 픽처들 내 비디오에 있어서 시간 리던던시를 감소 또는 제거하기 위해 시간 예측에 의존한다. 인트라-모드 (I 모드) 는 여러 공간 기반 코딩 모드들 중 어느 것을 지칭할 수도 있다. 인터-모드들, 이를테면 단방향 예측 (P 모드) 또는 양방향 예측 (B 모드) 는, 여러 시간 기반 코딩 모드들 중 어느 것을 지칭할 수도 있다.

[0062] 도 2 에 도시된 바와 같이, 비디오 인코더 (20) 는 인코딩될 비디오 프레임 내의 현재 비디오 블록을 수신한다. 도 2 의 예에서, 비디오 인코더 (20) 는, 모드 선택 유닛 (40), (디코딩된 화상 버퍼 (DPB) 로도 지칭될 수도 있는) 참조 화상 메모리 (64), 합산기 (50), 변환 처리 유닛 (52), 양자화 유닛 (54), 및 엔트로피 인코딩 유닛 (56) 을 포함한다. 모드 선택 유닛 (40) 은, 차례로, 모션 보상 유닛 (44), 모션 추정 유닛 (42), 인트라-예측 유닛 (46), 및 파티션 유닛 (48) 을 포함한다. 비디오 블록 복원을 위해, 비디오 인코더 (20) 는 또한 역양자화 유닛 (58), 역변환 유닛 (60), 및 합산기 (62) 를 포함한다. 디블록킹 필터 (도 2 에 도시되지 않음) 는 또한 블록 경계들을 필터링하여 복원된 비디오로부터 블록키니스 아티팩트들을 제거하기 위해 포함될 수도 있다. 원하는 경우, 디블록킹 필터는 통상적으로 합산기 (62) 의 출력을 필터링한다. 추가적인 필터들 (인 루프 또는 포스트 루프) 이 또한, 디블록킹 필터에 추가하여 사용될 수도 있다. 그러한 필터들은 간결성을 위해 나타내지는 않았지만, 원한다면, (인루프 필터로서) 합산기 (50) 의 출력을 필터링할 수도 있다.

[0063] 비디오 데이터 메모리 (66) 는 비디오 인코더 (20) 의 컴포넌트들에 의해 인코딩될 비디오 데이터를 저장하도록 구성될 수도 있다. 비디오 데이터 메모리 (101) 에 저장된 비디오 데이터는, 예를 들어, 비디오 소스 (18) 로부터 획득될 수도 있다. 디코딩된 픽처 버퍼 (116) 는, 예컨대 인트라 또는 인터 예측 모드들에서, 비디오 인코더 (20) 에 의해 비디오 데이터를 인코딩함에 있어서 사용하기 위한 레퍼런스 비디오 데이터를 저장하는 레퍼런스 픽처 메모리일 수도 있다. 비디오 데이터 메모리 (101) 는 다양한 메모리 디바이스들, 이를 테면, SDRAM (synchronous DRAM), MRAM (magnetoresistive RAM), RRAM (resistive RAM) 을 포함하는 DRAM (Dynamic random access memory), 또는 다른 유형들의 메모리 디바이스들의 어느 것에 의해 형성될 수도 있다. 비디오 데이터 메모리 (101) 는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에 있어서, 비디오 데이터 메모리 (101) 는 비디오 인코더 (20) 의 다른 컴포넌트들과 온-칩형이거나 또는 그 컴포넌트들에 대하여 오프-칩형일 수도 있다.

- [0064] 인코딩 프로세스 동안, 비디오 인코더 (20) 는 인코딩될 비디오 프레임 또는 슬라이스를 수신한다. 프레임 또는 슬라이스는 다수의 비디오 블록들로 분할될 수도 있다. 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 하나 이상의 참조 프레임들에서의 하나 이상의 블록들에 대해 수신된 비디오 블록의 인터-예측 인코딩을 수행해 시간적 예측을 제공한다. 인트라 예측 유닛 (46) 은 대안적으로, 코딩될 블록과 동일한 프레임 또는 슬라이스에서의 하나 이상의 이웃 블록들에 상대적으로 수신된 비디오 블록의 인트라 예측 인코딩을 수행하여 공간적 예측을 제공할 수도 있다. 비디오 인코더 (20) 는, 예를 들어, 비디오 데이터의 각각의 블록에 대한 적당한 코딩 모드를 선택하기 위해 다중의 코딩 패스들을 수행할 수도 있다.
- [0065] 또한, 파티션 유닛 (48) 은, 이전 코딩 패스들에서 이전 파티셔닝 스킴들의 평가에 기초하여, 비디오 데이터의 블록들을 서브 블록들로 파티셔닝할 수도 있다. 예를 들어, 파티션 유닛 (48) 은, 초기에 프레임 또는 슬라이스를 CTU 들로 파티셔닝할 수도 있고, 레이트 왜곡 분석 (예를 들어, 레이트 왜곡 최적화) 에 기초하여, CTU 들의 각각을 서브 CU들로 파티셔닝할 수도 있다. 모드 선택 유닛 (40) 은 추가로, CTU 의 서브-CU들로의 파티셔닝을 표시하는 쿼트트리 데이터 구조를 생성할 수도 있다. 쿼트트리의 리프-노드 CU 들은 하나 이상의 PU 들 및 하나 이상의 TU 들을 포함할 수도 있다.
- [0066] 모드 선택 유닛 (40) 은 예컨대, 여러 결과들에 기초하여 예측 모드들 중의 하나, 인트라 또는 인터를 선택할 수도 있고, 결과적인 예측된 블록을, 잔차 데이터를 생성하기 위하여 합산기 (50) 에, 그리고 참조 프레임으로서의 이용을 위한 인코딩된 블록을 재구성하기 위하여 합산기 (62) 에 제공한다. 모드 선택 유닛 (40) 은 또한, 모션 벡터들, 인트라 모드 표시자들, 파티션 정보, 및 다른 그러한 신택스 정보와 같은 신택스 엘리먼트들을 엔트로피 인코딩 유닛 (56) 에 제공한다.
- [0067] 모션 추정 유닛(42)과 모션 보상 유닛(44)은 고도로 통합될 수도 있지만 개념적 목적들을 위해 별개로 예시된다. 모션 추정 유닛(42)에 의해 수행되는 모션 추정은 비디오 블록들에 대한 모션을 추정하는 모션 벡터들을 생성하는 프로세스이다. 모션 벡터는, 예를 들어, 현재 프레임 (또는 다른 코딩된 유닛) 내의 코딩되는 현재 블록에 관하여 참조 프레임 (또는 다른 코딩된 유닛) 내의 예측 블록에 관한 현재 비디오 프레임 또는 화상 내의 비디오 블록의 PU 의 변위를 나타낼 수도 있다. 예측 블록은 픽셀 차이의 관점에서 코딩될 블록과 밀접하게 매칭되도록 발견되는 블록이며, 이 픽셀 차이는 절대 차이의 합 (SAD), 제곱 차이의 합 (SSD), 또는 다른 상이한 메트릭들에 의해 결정될 수도 있다. 일부 예들에 있어서, 비디오 인코더 (20) 는 레퍼런스 픽처 메모리 (64) 에 저장된 레퍼런스 픽처들의 서브-정수 픽셀 포지션들에 대한 값들을 계산할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 참조 픽처의 1/4 픽셀 포지션들, 1/8 픽셀 포지션들, 또는 다른 분수의 픽셀 포지션들의 값들을 보간할 수도 있다. 따라서, 모션 추정 유닛 (42) 은 풀 픽셀 포지션들 및 분수 픽셀 포지션들에 관한 모션 탐색을 수행하고, 분수 픽셀 정밀도로 모션 벡터를 출력할 수도 있다.
- [0068] 모션 추정 유닛 (42) 은 인터-코딩된 슬라이스에서의 비디오 블록의 PU 에 대한 모션 벡터를, 그 PU 의 포지션을 참조 픽처의 예측 블록의 포지션과 비교함으로써 계산한다. 참조 픽처는 제 1 참조 픽처 리스트 (List0) 또는 제 2 참조 픽처 리스트 (List1)로부터 선택될 수도 있고, 이들의 각각은 참조 픽처 메모리 (64) 내에 저장된 하나 이상의 참조 픽처들을 식별한다. 모션 추정 유닛(42)은 산출된 모션 벡터를 엔트로피 인코딩 유닛(56)과 모션 보상 유닛(44)으로 전송한다.
- [0069] 모션 보상 유닛 (44) 에 의해 수행된 모션 보상은, 모션 추정 유닛 (42) 에 의해 결정된 모션 벡터에 기초한 예측 블록의 페칭 (fetching) 또는 생성을 수반할 수도 있다. 또, 일부 예에서, 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 기능적으로 통합될 수도 있다. 현재 비디오 블록의 PU를 위한 모션 벡터의 수신시에, 모션 보상 유닛 (44) 은, 모션 벡터가 참조 화상 리스트들 중 하나에서 가리키는 예측 블록을 로케이팅할 수도 있다. 합산기 (50) 는, 하기에서 논의되는 바와 같이, 코딩되는 현재 비디오 블록의 픽셀 값들로부터 예측 블록의 픽셀 값들을 감산하여 픽셀 차이 값들을 형성함으로써 잔차 비디오 블록을 형성한다. 일반적으로, 모션 추정 유닛 (42) 은 루마 컴포넌트들에 대해 모션 추정을 수행하고, 모션 보상 유닛 (44) 은 크로마 컴포넌트들 및 루마 컴포넌트들 양자 모두를 위해 루마 컴포넌트들에 기초하여 계산된 모션 벡터들을 사용한다. 모드 선택 유닛 (40) 은 또한, 비디오 슬라이스의 비디오 블록들을 디코딩함에 있어서 비디오 디코더 (30) 에 의한 사용을 위해 비디오 블록들 및 비디오 슬라이스와 연관된 신택스 엘리먼트들을 생성할 수도 있다.
- [0070] 인트라-예측 유닛 (46) 은, 상술된 바와 같이, 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 수행되는 인터-예측에 대한 대안으로서, 현재 블록을 인트라-예측할 수도 있다. 특히, 인트라-예측 유닛 (46) 은 현재 블록을 인코딩하는데 이용하기 위한 인트라-예측 모드를 결정할 수도 있다. 일부 예들에 있어서, 인트라-예측 유닛 (46) 은 예를 들어 별도의 인코딩 패스들 동안에 다양한 인트라-예측 모드들을 이용하여 현재 블록을 인코딩

할 수도 있으며, 인트라-예측 유닛 (46) (또는 일부 예들에서는 모드 선택 유닛 (40)) 은 테스트된 모드들로부터의 이용을 위해 적당한 인트라-예측 모드를 선택할 수도 있다.

[0071] 예를 들어, 인트라-예측 유닛 (46) 은 다양한 테스트된 인트라-예측 모드들에 대한 레이트-왜곡 분석을 이용하여 레이트-왜곡 값들을 계산하고, 테스트된 모드들 중 최상의 레이트-왜곡 특성들을 갖는 인트라-예측 모드를 선택할 수도 있다. 레이트-왜곡 분석은 일반적으로, 인코딩된 블록과 그 인코딩된 블록을 생성하도록 인코딩되었던 원래의 인코딩되지 않은 블록 간의 왜곡 (또는 에러) 의 양뿐 아니라 인코딩된 블록을 생성하는데 사용된 비트 레이트 (즉, 비트들의 수) 를 결정한다. 인트라-예측 유닛 (46) 은 다양한 인코딩된 블록들에 대한 왜곡들 및 레이트들로부터의 비율들을 계산하여, 어느 인트라-예측 모드가 그 블록에 대한 최상의 레이트-왜곡 값을 나타내는지를 결정할 수도 있다.

[0072] 블록을 위한 인트라-예측 모드를 선택한 후에, 인트라-예측 유닛 (46) 은 엔트로피 인코딩 유닛 (56) 에 블록을 위한 선택된 인트라-예측 모드를 나타내는 정보를 제공할 수도 있다. 엔트로피 인코딩 유닛(56)은 선택된 인트라 예측 모드를 나타내는 정보를 인코딩할 수도 있다. 비디오 인코더 (20) 는 송신된 비트스트림에서 구성 데이터를 포함할 수도 있고, 이는 복수의 인트라 예측 모드 인덱스 테이블들 및 복수의 수정된 인트라 예측 모드 인덱스 테이블들 (코드워드 맵핑 테이블이라고도 한다), 다양한 블록들을 위한 인코딩 컨텍스트들의 정의들, 그리고 컨텍스트들의 각각을 위해 사용할 최고 확률 인트라 예측 모드, 인트라 예측 모드 인덱스 테이블, 및 수정된 인트라 예측 모드 인덱스 테이블의 표시들을 포함할 수도 있다.

[0073] 비디오 인코더 (20) 는 모드 선택 유닛 (40) 으로부터의 예측 데이터를, 코딩되는 오리지널 비디오 블록으로부터 감산함으로써 잔차 비디오 블록을 형성한다. 합산기 (50) 는 이 감산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 변환 처리 유닛 (52) 은 변환, 이를테면 이산 코사인 변환 (DCT) 또는 개념적으로 유사한 변환을 잔차 블록에 적용하며, 변환 계수 값들을 포함하는 비디오 블록을 생성한다. 웨이블릿 변환, 정수 변환, 서브밴드 변환, DST (Discrete Sine Transform) 또는 다른 유형의 변환을 DCT 대신 사용할 수 있다. 어느 경우든, 변환 처리 유닛 (52) 은 변환을 잔차 블록에 적용하며, 변환 계수들의 블록을 생성한다. 변환은 잔차 정보를 픽셀 도메인으로부터 주파수 도메인과 같은 변환 도메인으로 변환할 수도 있다. 변환 프로세싱 유닛 (52)은 결과적인 변환 계수들을 양자화 유닛(54)에 전송할 수도 있다. 양자화 유닛(54)은 변환 계수들을 양자화하여 비트 레이트를 더욱 감소시킨다. 양자화 프로세스는 그 계수들의 일부 또는 전부에 연관된 비트 깊이를 감소시킬 수도 있다. 양자화 정도는 양자화 파라미터를 조정함으로써 수정될 수도 있다.

[0074] 양자화 이후, 엔트로피 인코딩 유닛 (56) 은 양자화된 변환 계수들을 엔트로피 코딩한다. 예를 들어, 엔트로피 인코딩 유닛 (56) 은 컨텍스트 적응적 가변 길이 코딩 (CAVLC), 컨텍스트 적응적 바이너리 산술 코딩 (CABAC), 신택스 기반 컨텍스트 적응적 바이너리 산술 코딩 (SBAC), 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩, 또는 다른 엔트로피 코딩 기술을 수행할 수도 있다. 컨텍스트-기반 엔트로피 코딩의 경우, 컨텍스트는 이웃 블록들에 기초할 수도 있다. 엔트로피 인코딩 유닛 (56) 에 의한 엔트로피 코딩 이후, 인코딩된 비트스트림은 다른 디바이스 (예를 들어, 비디오 디코더 (30)) 로 송신되거나 또는 더 나중의 송신 또는 취출을 위해 아카이브될 수도 있다.

[0075] 역 양자화 유닛 (58) 및 역 변환 유닛 (60) 은 역 양자화 및 역 변환을 각각 적용하여, 픽셀 도메인에서 잔차 블록을 재구성한다. 특히, 합산기 (62) 는 재구성된 잔차 블록을 모션 보상 유닛 (44) 또는 인트라 예측 유닛 (46) 에 의해 더 일찍 생성되는 모션 보상된 예측 블록에 가산하여 참조 화상 메모리 (64) 에 저장하기 위해 재구성된 비디오 블록을 생성한다. 재구성된 비디오 블록은 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 참조 블록으로서 사용되어 후속 비디오 프레임에서 블록을 인터-코딩할 수도 있다.

[0076] 도 3 는 본 개시의 하나 이상의 DMVD 기법들을 수행하도록 구성될 수도 있는 비디오 디코더 (30) 의 예를 나타내는 블록도이다. 예를 들어, 비디오 디코더 (30) 는 (1) DMVD 에 의해 도출된 MV 및/또는 (2) 초기 MV 와 DMVD 에 의해 도출된 MV 사이의 MV 차이에 하나 이상의 제약들을 적용하도록 구성될 수도 있다. 제약이 DMVD 에 적용될 때, 특정 예들에서, 제약을 만족하는 도출된 모션 정보 만이 유효한 모션 정보로 간주된다. 따라서, 제약이 사용되지 않으면 MV 들의 리스트 및 최종 선택된 MV 는 상이할 수도 있다.

[0077] 제 1 예에서, 제약은 도출된 MV 가 대칭인지 여부를 포함한다. 제 2 예에서, 제약은 초기 MV와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 대칭인지 여부를 포함한다. 제 3 예에서, 제약은 도출된 MV 가 반대칭인지 여부를 포함한다. 제 4 예에서, 제약은 초기 MV와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 반대칭인지 여부를 포함한다. 일부 예들에서, 대칭/반대칭 MV/MV 차이 제약의 인에이블링은 명시적으로 시그널링된다. 다른 예들에서, 제약들은 일부 코딩된 정보에 따라 암시 적으로 결정될 수있다. 다른 예들에서, 제약이 일부 예들에

만 적용되도록 조건들이 제약들 상에 배치될 수도 있다.

[0078] 도 3의 예에서, 비디오 디코더 (30)는 비디오 데이터 메모리 (68), 엔트로피 디코딩 유닛 (70), 모션 보상 유닛 (72), 인트라 예측 유닛 (74), 역양자화 유닛 (76), 역변환 유닛 (78), 레퍼런스 픽처 메모리 (82) 및 합산기 (80)를 포함한다. 비디오 디코더 (30)는, 일부 예들에서, 비디오 인코더 (20) (도 2)에 대해 설명된 인코딩 패스에 일반적으로 상반되는 디코딩 패스를 수행할 수도 있다. 모션 보상 유닛 (72)은 엔트로피 디코딩 유닛 (70)으로부터 수신된 모션 벡터들에 기초하여 예측 데이터를 생성할 수도 있지만, 인트라-예측 유닛 (74)은 엔트로피 디코딩 유닛 (70)으로부터 수신된 인트라-예측 모드 표시자들에 기초하여 예측 데이터를 생성할 수도 있다.

[0079] 비디오 데이터 메모리 (68)는 비디오 디코더 (30)의 컴포넌트들에 의해 디코딩될 인코딩된 비디오 데이터, 이를테면 인코딩된 비디오 비트스트림을 저장할 수도 있다. 비디오 데이터 메모리 (68)에 저장된 비디오 데이터는, 예를 들어, 컴퓨터 판독가능 매체 (16)로부터, 예컨대, 카메라와 같은 로컬 비디오 소스로부터, 비디오 데이터의 유선 또는 무선 네트워크 통신을 통해, 또는 물리적 데이터 저장 매체들에 액세스함으로써 획득될 수도 있다. 비디오 데이터 메모리 (68)은, 인코딩된 비디오 비트스트림으로부터 인코딩된 비디오 데이터를 저장하는 코딩된 픽처 버퍼 (CPB)를 형성할 수도 있다. 참조 픽처 메모리 (82)는, 예를 들어, 인트라- 또는 인터-예측 모드들에서 또는 출력에 대하여 비디오 디코더 (30)에 의해 비디오 데이터를 디코딩함에 있어서 사용하기 위한 레퍼런스 비디오 데이터를 저장하는 메모리일 수도 있다. 비디오 데이터 메모리 (68) 및 참조 픽처 메모리 (82)는 SDRAM을 포함하는 DRAM, MRAM, RRAM, 또는 다른 타입들의 메모리 디바이스들과 같은 다양한 메모리 디바이스들의 임의의 것에 의해 형성될 수도 있다. 비디오 데이터 메모리 (68) 및 레퍼런스 픽처 메모리 (82)는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에 있어서, 비디오 데이터 메모리 (68)는 비디오 디코더 (30)의 다른 컴포넌트들과 온-칩형이거나 그 컴포넌트들에 대하여 오프-칩형일 수도 있다.

[0080] 비디오 데이터 메모리 (68)는 비트스트림의 인코딩된 비디오 데이터 (예를 들어, NAL 유닛)를 수신하여 저장한다. 엔트로피 디코딩 유닛 (70)은 비디오 데이터 메모리 (68)로부터 인코딩된 비디오 데이터 (예를 들어, NAL 유닛)를 수신할 수도 있고 선택스 엘리먼트들을 얻기 위해 NAL 유닛을 파싱할 수도 있다. 엔트로피 디코딩 유닛 (70)은 NAL 유닛들에서 엔트로피 인코딩된 선택스 엘리먼트들을 엔트로피 디코딩할 수도 있다. 디코딩 프로세스 동안, 비디오 디코더 (30)는 연관된 선택스 엘리먼트들 및 인코딩된 비디오 슬라이스의 비디오 블록들을 나타내는 인코딩된 비디오 비트스트림을 비디오 인코더 (20)로부터 수신한다. 비디오 디코더 (30)의 엔트로피 디코딩 유닛 (70)은 비트스트림을 엔트로피 디코딩하여, 양자화된 계수들, 모션 벡터들 또는 인트라-예측 모드 표시자들, 및 다른 선택스 엘리먼트들을 생성한다. 엔트로피 디코딩 유닛 (70)은 모션 벡터들 및 다른 선택스 엘리먼트들을 모션 보상 유닛 (72)으로 포워딩한다. 비디오 디코더 (30)는 비디오 슬라이스 레벨 및/또는 비디오 블록 레벨에서 선택스 엘리먼트들을 수신할 수도 있다.

[0081] 비디오 슬라이스가 인트라-코딩된 (I) 슬라이스로서 코딩될 때, 인트라 예측 유닛 (74)은, 현재 프레임 또는 화상의 이전에 디코딩된 블록들로부터 시그널링된 인트라 예측 모드 및 데이터에 기초하여 현재 비디오 슬라이스의 비디오 블록을 위한 예측 데이터를 생성할 수도 있다. 비디오 프레임이 인터-코딩된 (즉, B 또는 P) 슬라이스로서 코딩된 경우, 모션 보상 유닛 (72)은 엔트로피 디코딩 유닛 (70)으로부터 수신된 모션 벡터들 및 다른 선택스 엘리먼트들에 기초하여 현재 비디오 슬라이스의 비디오 블록에 대한 예측 블록들을 생성한다. 예측 블록들은 참조 픽처 리스트들 중 하나의 참조 픽처 리스트 내의 참조 픽처들 중 하나의 참조 픽처로부터 생성될 수도 있다. 비디오 디코더 (30)는 참조 픽처 메모리 (82)에 저장된 참조 픽처들에 기초한 디폴트 구성 기술들을 이용하여 참조 프레임 리스트들, 즉, LList0 및 List1을 구성할 수도 있다. 모션 보상 유닛 (72)은 모션 벡터들 및 다른 선택스 엘리먼트들을 파싱함으로써 현재 비디오 슬라이스의 비디오 블록에 대한 예측 정보를 결정하고, 그 예측 정보를 사용하여, 디코딩되는 현재 비디오 블록에 대한 예측 블록들을 생성한다. 예를 들어, 모션 보상 유닛 (72)은 수신된 선택스 엘리먼트들의 일부를 사용하여 비디오 슬라이스의 비디오 블록들을 코딩하는데 사용된 예측 모드 (예를 들어, 인트라- 또는 인터-예측), 인터-예측 슬라이스 유형 (예를 들어, B 슬라이스, 또는 P 슬라이스), 슬라이스를 위한 참조 화상 리스트들의 하나 이상을 위한 구성 정보, 슬라이스의 각 인트라-인코딩된 비디오 블록을 위한 모션 벡터들, 슬라이스의 각 인터-코딩된 비디오 블록을 위한 인터-예측 상태, 및 현재 비디오 슬라이스에서 비디오 블록들을 디코딩하기 위한 다른 정보를 결정한다.

[0082] 모션 보상 유닛 (72)은 또한, 보간 필터들에 기초하여 보간을 수행할 수도 있다. 모션 보상 유닛 (72)은 비디오 블록들의 인코딩 동안 비디오 인코더 (20)에 의해 이용되는 보간 필터들을 이용하여 참조 블록들의 서브정수 픽셀들을 위한 보간된 값들을 계산할 수도 있다. 이 경우에, 모션 보상 유닛 (72)은 수신된 선택스 엘리

먼트들로부터 비디오 인코더 (20) 에 의해 이용된 보간 필터들을 결정하고, 그 보간 필터들을 이용하여 예측 블록들을 생성할 수도 있다.

[0083] 역 양자화 유닛 (76) 은 비트스트림에서 제공되고 엔트로피 디코딩 유닛 (70) 에 의해 디코딩된 양자화된 변환 계수들을 역 양자화, 즉, 탈양자화한다 (de-quantize). 역양자화 프로세스는 비디오 슬라이스에 있어서의 각각의 비디오 블록에 대해 비디오 디코더 (30) 에 의해 계산된 양자화 파라미터 (QP_i) 의 이용을 포함하여, 적용되어야 하는 양자화의 정도 및 유사하게 역양자화의 정도를 결정할 수도 있다.

[0084] 역 변환 유닛 (78) 은, 픽셀 도메인에서 잔차 블록들을 생성하기 위해 변환 계수들에, 역 변환, 예를 들어, 역 DCT, 역 정수 변환, 또는 개념적으로 유사한 역 변환 프로세스를 적용한다.

[0085] 모션 보상 유닛 (72) 이 모션 벡터들 및 다른 선택스 엘리먼트들에 기초하여 현재 비디오 블록에 대한 예측 블록을 생성한 이후, 비디오 디코더 (30) 는 역변환 유닛 (78) 으로부터의 잔차 블록들을 모션 보상 유닛 (72) 에 의해 생성된 대응하는 예측 블록들과 합산함으로써 디코딩된 비디오 블록을 형성한다. 합산기 (80) 는 이 합산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 원하는 경우, 블로키니스 아티팩트 (blockiness artifact) 들을 제거하기 위하여 디코딩된 블록들을 필터링하기 위한 디블록킹 필터가 또한 적용될 수도 있다. (코딩 루프에 있어서 또는 코딩 루프 이후에) 다른 루프 필터들이 또한 픽셀 천이들을 평활하게 하거나 또는 그렇지 않으면 비디오 품질을 개선하기 위해 이용될 수도 있다. 그 후, 소정의 프레임 또는 픽처에 있어서의 디코딩된 비디오 블록들이 레퍼런스 픽처 메모리 (82) 에 저장되고, 이 레퍼런스 픽처 메모리는 후속적인 모션 보상을 위해 사용되는 레퍼런스 픽처들을 저장한다. 참조 화상 메모리 (82) 는 또한, 도 1 의 디스플레이 디바이스 (32) 와 같은 디스플레이 디바이스 상에 나중에 표출하기 위해 디코딩된 비디오를 저장한다.

[0086] 이 개시물의 기법들에 따르면, 비디오 인코더 (20) 및 비디오 디코더 (30) 와 같은 비디오 코더는 비디오 데이터의 현재 블록에 대한 모션 정보를 도출하기 위해 DMVD 를 수행할 수도 있다. 특히, 이러한 기법들은 다음 기법들 중 임의의 것 또는 모두를 단독으로 또는 임의의 조합으로 포함할 수도 있다.

[0087] 본 개시의 개념 중 하나는 DMVD 를 개선하는 것이다. 이 기법들은 아래에서 논의되는 바와 같이 몇 가지 다른 항목화된 양태들로 정교화된다. DMVD 를 개선하기 위한 다음의 기법들은 개별적으로 적용될 수도 있다. 대안적으로, 본 개시에서 설명된 기법들의 임의의 조합이 함께 적용될 수도 있다. 개별적으로 또는 임의의 조합으로 적용되든, 다음의 DMVD 기법들은 코딩 효율을 증가시키고 및/또는 비트 레이트를 감소시킬 수도 있다. 예를 들어, (1) DMVD 에 의해 유도된 MV 및/또는 (2) 초기 MV 와 DMVD 에 의해 유도된 MV 간의 MV 차이를 제한하는 것은 코딩 효율을 증가시키고 및/또는 비트 레이트를 감소시킬 수도 있다. 전술한 각각의 예에서, 제약은 조건에 기초하여 적용될 수도 있다.

[0088] HEVC 에서 슬라이스의 최대 코딩 유닛은 코딩 트리 블록 (CTB) 또는 코딩 트리 유닛 (CTU) 으로 지칭된다. CTB 는 노드가 코딩 유닛들인 쿼드 트리를 포함한다. CTB 의 사이즈는 (비록 기술적으로 8x8 CTB 사이즈들이 지원될 수 있음에도 불구하고) HEVC 메인 프로파일에서 16x16 에서부터 64x64 까지의 범위에 있을 수 있다. 코딩 유닛 (CU) 은 CTB 의 동일한 사이즈일 수 있고 8x8 만큼 작을 수 있다. 각각의 코딩 유닛은 하나의 모드로 코딩된다. CU 가 인터 코딩될 경우, CU 는 2 또는 4개의 예측 유닛들 (PU들) 로 추가로 파티셔닝될 수도 있거나, 또는 추가의 파티션이 적용되지 않을 경우 단지 하나의 PU 가 된다. 2개의 PU들이 하나의 CU 에 존재할 경우, 그 PU들은 하프 사이즈 직사각형들 또는 CU 의 $\frac{1}{4}$ 또는 $\frac{3}{4}$ 사이즈인 2개의 직사각형들일 수 있다. CU 가 인터 코딩될 때, 모션 정보의 하나의 세트는 각각의 PU 에 대해 존재한다. 부가적으로, 각각의 PU 는 모션 정보의 세트를 도출하기 위해 고유한 인터-예측 모드로 코딩된다.

[0089] HEVC 표준에서, 예측 유닛 (PU) 에 대해 각각 병합 (스킵이 특수 병합의 경우로 간주 됨) 및 AMVP (advanced motion vector prediction) 모드로 명명된 2 개의 인터 예측 모드가 있다. AMVP 모드 또는 병합 모드 중 어느 하나에서, 모션 벡터 (MV) 후보 리스트가 다수의 모션 벡터 예측기들에 대해 유지된다. 현재 PU 의 병합 모드에서의 참조 인덱스들 뿐 아니라 모션 벡터(들)가 MV 후보 리스트로부터 하나의 후보를 취함으로써 생성된다.

[0090] MV 후보 리스트는 병합 모드에 대한 5개까지의 후보들 및 AMVP 모드에 대한 단지 2개의 후보들을 포함한다. 병합 후보는 모션 정보의 세트, 예를 들어, 레퍼런스 픽처 리스트 (List0 및 List1) 양쪽 모두에 대응하는 모션 벡터들 및 레퍼런스 인덱스를 포함할 수도 있다. 병합 후보가 병합 인덱스에 의해 식별되면, 참조 픽처들은 현재 블록들의 예측을 위해 사용될 뿐 아니라 관련 모션 벡터들이 결정된다. 하지만, AMVP 모드 하에서, List0 또는 List1 중 어느 하나로부터의 각각의 잠재적 예측 방향에 대해, AMVP 후보가 오직 모션 벡터만을 포함하기

때문에, MV 후보 인덱스에 대한 MV 예측자 (MVP) 인덱스와 함께, 참조 인덱스가 명시적으로 시그널링될 필요가 있다. AMVP 모드에 있어서, 예측 모션 벡터들은 추가로 정제될 수 있다. 상기에서 볼 수 있는 바와 같이, 병합 후보는 모션 정보의 전체 세트에 대응하는 한편, AMVP 후보는 특정 예측 방향 및 참조 인덱스에 대해 단지 하나의 모션 벡터를 포함한다. 양자 모두의 모드들에 대한 후보들은 동일한 공간 및 시간 이웃 블록들로부터 유사하게 도출된다.

[0091] DMVD 기법들의 개념은 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 가 이전에 디코딩된 정보를 사용하여 모션 벡터 및 예측 방향과 같은 모션 정보를 도출하는 것이다. 본 명세서에서 사용된 바와 같이, 약어 "MV"는 모션 벡터를 나타낸다. List0 및 List1은 디코딩된 픽처 버퍼 (DPB) 또는 참조 픽처 메모리 (82) (도 3) 내의 픽처들의 리스트들로서 구성되는 2 개의 리스트들이다. 이 목록 중 하나에서 특정 픽처를 식별하기 위해 참조 픽처 인덱스라고 불리는 인덱스가 사용된다. 단방향 예측의 경우, 이들 목록들 중 어느 하나에서 픽처가 선택될 수 있다. 양방향 예측의 경우, 각각의 리스트 중 하나씩 (예를 들어, List0 에서 하나 및 List1 에서 하나), 2 개의 픽처들이 선택된다. 현재의 접근법에서, DMVD 에 의해 유도된 List0 및 List1 MV 들은 양방향 템플릿 매칭 및 FRUC (Frame-rate Up-conversion) 템플릿 매칭과 같은 일부 DMVD 기법들에 대해 독립적으로 결정된다. 본 개시에서, 초기 MV 에 기초하여 양방향 예측 MV 들 (예를 들어, List0 MV 및 List1 MV 를 포함하는 MV 들의 쌍) 을 도출할 때, 대칭/반대칭 MV 및/또는 대칭/반대칭 MV 차이의 제약이 적용될 수도 있다. 일부 예들에서, 대칭 MV 제약 및 대칭 MV 차이 제약을 적용하는 것과 같은 2 이상의 제약이 적용될 수도 있다. 제약은 하나 이상의 조건들을 포함할 수도 있다. 일부 예들에서, 제약이 (예를 들어, 제약을 DMVD 로부터 도출된 MV 및/또는 MV 차이에 적용함으로써) DMVD 에 적용되는 경우, 제약을 충족시키는 (예컨대, 통과하는) 도출된 모션 정보만이 유효한 모션 정보로 간주될 수도 있다. 예를 들어, 제약을 만족시키는 (예컨대, 통과하는) DMVD 로부터 도출된 MV 는 유효한 MV 로 간주될 수 있고, 제약을 만족시키지 않는 DMVD 로부터 도출된 MV 는 유효하지 않은 MV 로 간주될 수도 있다. 다른 예에서, 제약을 만족시키는 DMVD 로부터 도출된 MV 차이는 유효한 MV 차이로 간주될 수 있고, 제약을 만족시키지 않는 DMVD 로부터 도출된 MV 차이는 유효하지 않은 MV 차이로 간주될 수도 있다. 일부 예들에서, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 유효한 모션 벡터 정보를 사용하여 비디오 데이터 (예를 들어, 비디오 데이터의 현재 블록) 를 코딩하고, 유효하지 않은 모션 벡터 정보를 사용하여 비디오 데이터 (예를 들어, 비디오 데이터의 현재 블록) 를 코딩하지 않도록 구성될 수도 있다. 일부 예들에서, 유효 모션 벡터 정보는 그 후 다른 유효 모션 벡터 정보와 비교될 수 있고, 최저 코스트 메트릭을 갖는 것으로 결정된 유효 모션 벡터 정보가 선택된다.

[0092] 공간 MV 후보는 특정 PU (PU0) 에 대해, 도 4 에 도시된 이웃 블록들로부터 도출되지만, 블록들로부터 후보들을 생성하는 방법들은 병합 및 AMVP 모드들에 대해 상이하다. 병합 모드에서, 번호들을 갖는 도 4a 에 도시된 순서로 최대 4 개의 공간 MV 후보가 도출될 수 있으며, 그 순서는 다음과 같다: 도 4a 에 도시된 바와 같이, 좌측 (0, A1), 상측 (1, B1), 우상측 (2, B0), 좌하측 (3, A0), 좌상측 (4, B2).

[0093] AVMP 모드에 있어서, 이웃 블록들은 2개의 그룹들로 분할된다: 도 4b 에 나타낸 바와 같이, 블록들 0 및 1 로 이루어지는 좌측 그룹, 및 블록들 2, 3, 및 4 로 이루어지는 상측 그룹. 각각의 그룹에 대해, 시그널링된 참조 인덱스에 의해 표시된 것과 동일한 참조 픽처를 참조하는 이웃한 블록에서의 잠재적인 후보가 그룹의 최종 후보를 형성하도록 선택될 최고 우선순위를 갖는다. 모든 이웃한 블록들이 동일한 참조 픽처를 포인팅하는 모션 벡터를 포함하지 않는 것이 가능하다. 따라서, 이러한 후보가 발견될 수 없으면, 최종 후보를 형성하기 위해 제 1 이용가능 후보가 스케일링될 것이며, 따라서, 시간적 거리 차이들이 보상될 수 있다.

[0094] 시간 모션 벡터 예측기 (TMVP) 후보는, 인에이블되고 이용가능한 경우, 공간 모션 벡터 후보들 이후에 MV 후보 리스트로 추가된다. TMVP 후보에 대한 모션 벡터 도출의 프로세스는 병합 및 AMVP 모드 양자 모두에 대해 동일하지만, 병합 모드에서 TMVP 후보에 대한 타겟 참조 인덱스는 항상 0으로 설정될 수도 있다. TMVP 후보 도출을 위한 프라이머리 블록 위치는 공간적 이웃 후보를 생성하는데 사용된 상측 및 좌측 블록들에 대한 바이어스를 보상하기 위해, 도 5a 에서 블록 "T" 로서 도시된 바와 같은 병치된 PU 외부의 우하측 블록이다. 하지만, 그 블록이 현재의 CTB 행의 외부에 로케이팅되거나 또는 모션 정보가 이용가능하지 않으면, 블록은 PU 의 중심 블록으로 치환된다.

[0095] TMVP 후보에 대한 모션 벡터는, 슬라이스 레벨에서 표시된, 병치된 픽처의 병치된 PU 로부터 도출된다. 공동 위치된 PU 에 대한 모션 벡터를 병치된 MV 라고 한다. AVC 에서의 시간적 다이렉트 모드와 유사하게, TMVP 후보 모션 벡터를 도출하기 위해, 공동 위치된 MV 는 도 5 에 도시된 바와 같이 시간적 거리 차이를 보상하도록 스케일링 될 필요가 있다.

- [0096] 병합 모드 및 AMVP 모드의 여러 양태들이 아래에 설명되어 있다.
- [0097] 모션 벡터 스케일링: 모션 벡터들의 값이 그 프리젠테이션 시간에서의 픽처들의 거리에 비례한다고 가정될 수 있다. 모션 벡터는 2개의 픽처들, 즉, 참조 픽처와, 모션 벡터를 포함하는 픽처 (즉, 포함 픽처) 를 연관시킨다. 모션 벡터가 다른 모션 벡터를 예측하는데 사용될 경우, 포함 픽처와 참조 픽처의 거리는 픽처 순서 카운트 (POC) 값들에 기초하여 계산된다.
- [0098] 예측될 모션 벡터에 대해, 그의 연관된 포함 픽처 및 참조 픽처 양자는 상이할 수도 있다. 따라서, (POC 에 기초한) 새로운 거리가 계산된다. 그리고, 모션 벡터는 이들 2개의 POC 거리들에 기초하여 스케일링된다. 공간 이웃 후보에 대해, 2개의 모션 벡터들에 대한 포함 픽처들은 동일한 반면, 참조 픽처들은 상이하다. HEVC 에 있어서, 모션 벡터 스케일링이 공간 및 시간 이웃 후보들에 대해 TMVP 및 AMVP 양자에 적용된다.
- [0099] 인공 모션 벡터 후보 생성: 모션 벡터 후보 리스트가 완료되지 않은 경우 (즉, 미리 결정된 최대 후보 수보다 적은 경우), 인공 모션 벡터 후보가 생성되어 모든 후보를 가질 때까지 리스트의 종단에 삽입된다.
- [0100] 병합 모드에 있어서, 2개의 유형들의 인공적인 MV 후보들: 즉, 오직 B-슬라이스들에 대해서만 도출되는 결합된 후보; 및 제 1 유형이 충분한 인공적인 후보들을 제공하지 않으면 오직 AMVP 에 대해서만 사용되는 제로 후보들이 존재한다.
- [0101] 후보 리스트에 이미 있고 필요한 모션 정보를 갖는 후보들의 각각의 쌍에 대해, 양방향 결합된 모션 벡터 후보들이 List0 에서의 픽처를 참조하는 제 1 후보의 모션 벡터와 List1 에서의 픽처를 참조하는 제 2 후보의 모션 벡터의 조합에 의해 도출된다.
- [0102] 후보 삽입을 위한 프루닝 (Pruning) 프로세스: 상이한 블록들로부터의 후보들은 동일하도록 발생할 수도 있으며, 이는 병합/AMVP 후보 리스트의 효율을 감소시킨다. 이러한 문제를 해결하기 위해 프루닝 프로세스가 적용된다. 그것은 특정 범위에서 동일한 후보를 삽입하는 것을 회피하기 위해 하나의 후보를 현재의 후보 리스트에서의 다른 후보들과 비교한다. 복잡도를 감소시키기 위해, 각각의 잠재적 후보를 모든 다른 현존 후보들과 비교하는 대신, 오직 제한된 수들의 프루닝 프로세스만이 적용된다.
- [0103] JEM 참조 소프트웨어의 일례에서, 디코더 측에서 현재 블록에 대한 모션 벡터 (MV)를 유도하거나 정제하는 몇몇 인터 코딩 툴이 있다. 이러한 DMVD 접근법들은 다음과 같이 정교화된다.
- [0104] 도 6 및 도 7 은 PMMVD (Pattern Matched Motion Vector Derivation)에 대한 개념을 나타내는 개념도이다. PMMVD 모드는 FRUC (Frame-Rate Up Conversion) 기법들을 기반으로하는 특수 병합 모드이다. 이 모드에서, 블록의 모션 정보는 시그널링되지 않고 디코더 측에서 유도된다. 이 기술은 JEM의 한 예에 포함되었다.
- [0105] FRUC 플래그는 병합 플래그가 참인 경우 CU 에 대해 시그널링된다. FRUC 플래그가 거짓일 때, 병합 인덱스가 시그널링되고 정규 병합 모드가 사용된다. FRUC 플래그가 참일 때, 블록에 대한 모션 정보를 유도하기 위해 어떤 방법 (예를 들어, 양방향 매칭 또는 템플릿 매칭) 이 사용될 것인지를 표시하기 위해 추가적인 FRUC 모드 플래그가 시그널링된다.
- [0106] 모션 유도 프로세스 동안, 양방향 매칭 또는 템플릿 매칭에 기초하여 초기 모션 벡터가 먼저 전체 CU 에 대해 유도된다. 먼저, CU 의, 또는 PMMVD 시드들로 불리는 병합 리스트가 체크되고, 최소 매칭 코스트 (예를 들어, 레이트-왜곡 (RD) 코스트) 를 야기하는 후보가 시작 포인트로서 선택된다. 그 후, 시작 포인트 중심으로 양방향 매칭 또는 템플릿 매칭에 기초한 로컬 검색이 수행되고, 최소 매칭 코스트를 초래하는 MV 가 전체 CU 의 MV 로서 취해질 수 있다. 후속적으로, 모션 정보는 유도된 CU 모션 벡터를 시작 포인트로서 서브 블록 레벨에서 추가로 정제된다.
- [0107] 본 개시의 본 기법들에 따르면, 예를 들어, DMVD에 의해 유도된 MV 가 대칭인지 또는 반대칭인지, 및/또는 (2) 초기 MV 와 DMVD 에 의해 유도된 MV 간의 MV 차이가 대칭인지 또는 반대칭인지 여부에 기초하여 도출된 CU 모션 벡터에 제약들이 적용될 수도 있다. 전술한 각각의 예에서, 제약들은 예를 들어 초기 MV 들 또는 도출된 MV 들에 대한 조건에 기초하여 적용될 수도 있다.
- [0108] 도 6 에 도시된 바와 같이, 양방향 매칭은 두 개의 다른 참조 픽처들 (Ref0 및 Ref1) 에서 현재 블록의 모션 궤적을 따라 두 개의 참조 블록들 (R_0 및 R_1) 사이의 최상의 매치를 찾아서 현재 블록 (Cur)의 모션 정보를 도출하는데 사용된다. 모션 궤적은 3 차원 연속 시공간 필드로서 이미지 시퀀스 (예를 들어, 참조 프레임 및 현재 프레임) 를 고려할 때 블록 내의 픽셀이 공간 및 시간을 따라 따르는 경로를 포함할 수도 있다. 연속 운동 궤적을

가정하면, 두 개의 참조 블록 (R_0 그리고 R_1) 을 가리키는 모션 벡터 (MV0 및 MV1) 는 현재 픽처 (Cur) 와 2 개의 참조 픽처들 (Ref0 및 Ref1) 사이의 시간적 거리에 비례한다. 도출된 MV (600 및 602) 는 양방향 매칭을 사용하여 도출되고 각각 참조 블록 (R'_0 그리고 R'_1) 을 가리킨다 특별한 경우에, 현재 픽처 (Cur) 가 2 개의 참조 픽처들 (Ref0 및 Ref1) 사이에 일시적으로 존재하고 현재 픽처로부터 2 개의 참조 픽처들까지의 시간 거리가 동일할 때, 양방향 매칭은 미리 기반 양방향 MV 가 된다. 결과적으로, 도출된 모션 벡터들 (600 및 602) 이 유사하게 미리링된다.

[0109] 도 7 에 도시된 바와 같이, 템플릿 매칭은 현재 픽처에서의 템플릿 (현재 블록의 상측 및/또는 좌측 이웃 블록) 과 참조 픽처 (Ref0 및 Ref1) 에서의 블록 (템플릿과 동일한 크기) 사이에서 최상의 매칭을 발견함으로써 현재 블록 (Cur) 의 모션 정보를 도출하는데 사용된다. 템플릿은 R_0 및 R_1 의 이웃 블록들을 검색함으로써 관심 블록 (Cur) 을 후보 참조들 (MV0 를 갖는 R_0 및 MV1 을 갖는 R_1) 또는 도출된 참조들 (MV 700 를 갖는 R'_0 및 MV 702 를 갖는 R'_1) 과 비교하는 데 사용되는 블록의 이웃 픽셀들을 포함할 수도 있다. 그런 다음 가장 유사한 참조가 예측으로 사용된다.

[0110] 비디오 인코더 (20) 에서, CU 에 대해 FRUC 병합 모드를 사용할지 여부에 대한 결정은 정상 병합 후보에 대해 수행된 바와 같은 RD 코스트 선택에 기초한다. 즉, 두 가지 매칭 모드들 (양방향 매칭 및 템플릿 매칭) 이 RD 코스트 선택을 사용하여 CU 에 대해 체크된다. 최저 코스트를 야기하는 매칭 모드는 다른 CU 모드와 추가로 비교된다. FRUC 매칭 모드가 가장 효율적인 것인 경우, CU 에 대해 FRUC 플래그가 참으로 설정되고 관련 매칭 모드가 사용된다.

[0111] 제 5 차 JVET 회의에서, <http://phenix.it-sudparis.eu/jvet/> 에서 이용가능한 “Enhanced Template Matching in FRUC Mode,” JVET-E0035 는 FRUC 템플릿 매칭을 더욱 향상시키기 위해 제안되었다. 예시적인 FRUC 템플릿 매칭 모드의 흐름도가 도 8 에 도시되어 있다. 제 1 단계에서, 템플릿 T_0 (및 그의 대응하는 모션 정보 MV0) 는 list0 참조 픽처로부터 현재 블록의 현재 템플릿 T_c 와 매칭하는 것으로 발견되었다. 제 2 단계에서, 템플릿 T_1 (및 그의 대응하는 모션 정보 MV1) 는 list1 참조 픽처로부터 발견되었다. 획득된 모션 정보 (MV0 및 MV1) 는 현재 블록의 예측자를 생성하기 위해 양방향 예측을 수행하는데 사용된다.

[0112] FRUC 템플릿 매칭 모드는 단방향 예측과 양방향 예측 사이에서 양방향 템플릿 매칭 및 적응적 선택을 도입함으로써 향상될 수도 있다. 도 8 에 대한 예시적인 변경들이 도 9 에 밑줄로 표시되어 있다.

[0113] 양방향 템플릿 매칭은 단방향 템플릿 매칭에 기초하여 구현될 수도 있다. 도 8 에 도시된 바와 같이, 매칭된 템플릿 T_0 는 List0 참조 픽처들 (800) 로부터의 템플릿 매칭의 제 1 단계에서 처음 발견된다. 여기서 List0 은 단지 예일뿐이다. 실제로, 제 1 단계에서 사용된 List0 또는 List1 이 대응하는 참조 픽처에서의 초기 템플릿과 현재 템플릿 사이의 초기 왜곡 코스트에 적응적인지 여부. 초기 템플릿은 제 1 템플릿 매칭을 수행하기 전에 이용 가능한 현재 블록의 초기 모션 정보로 결정될 수 있다. 최소 초기 템플릿 왜곡 코스트에 대응하는 참조 픽처 리스트는 템플릿 매칭의 제 1 단계에서 사용될 것이다. 예를 들어, list0 에 대응하는 초기 템플릿 왜곡 코스트가 List1에 대응하는 코스트보다 크지 않으면, List0 이 템플릿 매칭의 제 1 단계에서 사용되고 List1 이 제 2 단계에서 사용되며, 그 후 현재 블록의 현재 템플릿 T_c 는 다음과 같이 업데이트된다:

[0114]
$$T'_c = 2 * T_c - T_0$$

[0115] 현재 템플릿 T_c 대신, 업데이트된 현재 템플릿 T'_c 는 제 2 템플릿 매칭 (802) 의 List1 참조 픽처들로부터 다른 매칭된 템플릿 T_1 를 찾는 데 사용된다 (802). 결과적으로, 매칭하는 템플릿 T_1 는 List0 및 List1 참조 픽처들을 함께 사용함으로써 발견된다 (804). 이 매칭 과정을 양방향 템플릿 매칭이라고 한다.

[0116] 모션 보상 예측 (MCP) 에 대한 단방향 예측과 양방향 예측 사이의 선택은 템플릿 매칭 왜곡에 기초할 수도 있다. 도 9 에 도시된 바와 같이, 템플릿 매칭 동안, 템플릿 T_0 와 T_c (현재 템플릿) 사이의 왜곡은 cost0 로서 계산될 수 있고 (900), 현재 템플릿이 업데이트될 수도 있으며 (902), 및 템플릿 T_1 과 T'_c (업데이트된 현재 템플릿) 사이의 왜곡은 cost1 으로서 계산될 수 있다 (904). cost0 이 $0.5 * cost1$ 보다 작은 경우 (906), MV0 에 기초한 단방향 예측은 FRUC 템플릿 매칭 모드에 적용될 수도 있고 (908); 그렇지 않으면, MV0 및 MV1에 기초한 양방향 예측이 적용된다 (910). cost1은 템플릿 T_1 과 T'_c (업데이트된 현재 템플릿) 사이의 차이를 나

타내므로 cost0 은 $0.5 * cost1$ 과 비교되며, 이것은 T_c (현재 템플릿) 와 $0.5 * (T_0 + T_1)$ 의 그의 예측 사이의 차이의 2 배이다. MCP 는 PU 레벨 모션 정제에 적용될 수도 있다. 서브 PU 레벨 모션 정제는 변경되지 않은 상태로 유지될 수도 있다.

[0117] 본 개시의 기법들에 따르면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 (1) 제약이 도출된 MV 가 대칭인지 여부를 포함하는지, (2) 초기 MV 와 DMVD 방법으로 도출된 MV 간의 MV 차이가 대칭인지 여부, (3) 도출된 MV 가 반대칭인지 여부, 및/또는 (4) 초기 MV 와 DMVD 방법에 의해 도출된 MV 간의 MV 차이가 반대칭인지 여부에 기초하여 PU 레벨 또는 서브 PU 레벨에서 FRUC 템플릿 매칭 모드 (단방향 및 양방향) 로부터 도출된 모션 벡터들을 제약할 수도 있다. 제약이 초기 MV 또는 FRUC 템플릿에 기초하여서만 적용되도록 조건들이 제약들에 배치될 수도 있다.

[0118] 도 10 은 JEM 의 일 예에서 양방향 광학 흐름과 관련된 개념을 나타내는 개념도이다. 양방향 광학 흐름 (Bi-directional Optical flow: BIO) 은 양방향 예측의 경우에 블록 단위 모션 보상에 더하여 수행되는 픽셀 단위 모션 정제이다. BIO 가 블록 내부의 미세한 모션을 보상하므로, BIO 를 인에이블링하는 것은 모션 보상을 위한 블록 크기를 확대하는 것을 야기한다. 샘플 레벨의 모션 정제는 철저한 검색이나 시그널링이 필요하지 않으며, 이는 각 샘플에 대해 미세한 모션 벡터를 제공하는 명시적인 방정식이 존재하기 때문이다.

[0119] $I^{(k)}$ 를 보상 블록 모션 후의 참조 k ($k=0, 1$) 로부터의 루미넌스 값으로 두면, $\partial I^{(k)}/\partial x, \partial I^{(k)}/\partial y$ 은 각각 $I^{(k)}$ 그레디언트의 수평 및 수직 컴포넌트들이다. 광학 흐름이 유효하다고 가정하면, 모션 벡터장 (v_x, v_y) 은 다음 식에 의해 주어진다:

$$\partial I^{(k)}/\partial t + v_x \partial I^{(k)}/\partial x + v_y \partial I^{(k)}/\partial y = 0. \quad (1)$$

[0121] 각 샘플의 모션 궤적에 대해 허마이트 보간법과 광학 흐름 방정식을 결합하면, 단부들에서 함수 값들 $I^{(k)}$ 및 도함수들 $\partial I^{(k)}/\partial x, \partial I^{(k)}/\partial y$ 과 매치하는 3차의 고유 다항식을 얻는다. $t=0$ 에서의 이러한 다항식의 값이 BIO 예측이다:

$$pred_{bio} = 1/2 \cdot (I^{(0)} + I^{(1)} + v_x/2 \cdot (\tau_1 \partial I^{(0)}/\partial x - \tau_0 \partial I^{(1)}/\partial x) + v_y/2 \cdot (\tau_1 \partial I^{(0)}/\partial y - \tau_0 \partial I^{(1)}/\partial y)) \quad (2)$$

[0122] 여기서, τ_0 및 τ_1 은 도 10 에 도시된 참조 프레임들까지의 거리를 표시한다. 거리들 τ_0 및 τ_1 은 Ref0 및 Ref1 에 대한 POC 에 기초하여 계산된다:

[0124] $\tau_0 = \text{POC}(\text{current}) - \text{POC}(\text{Ref0}), \tau_1 = \text{POC}(\text{Ref1}) - \text{POC}(\text{current})$. 양 예측들이 동일한 시간 방향으로 (양자 모두 과거로부터 또는 양자 모두 미래로 부터) 오면, 부호들이 상이하다 $\tau_0, \tau_1 < 0$. 이러한 경우, BIO 는 예측이 동일한 시간 순간으로부터 오지 않는 경우에만 ($\tau_0 \neq \tau_1$) 적용되고, 양 참조된 영역들은 년-제로 모션을 갖고 ($MVx_0, MVy_0, MVx_1, MVy_1 \neq 0$), 블록 모션 벡터들은 시간 거리에 비례한다 ($MVx_0/MVx_1 = MVy_0/MVy_1 = -\tau_0/\tau_1$).

[0125] 모션 벡터장 (v_x, v_y) 은 점 A 및 B 에서의 값들 사이의 차이 Δ 를 최소화함으로써 결정된다 (도 10 상의 모션 궤적과 참조 프레임 평면의 교차). 모델은 Δ 에 대한 로컬 테일러 전개식의 제 1 선형 항만을 사용한다:

$$\Delta = (I^{(0)} - I^{(0)}_0 + v_x (\tau_1 \partial I^{(0)}/\partial x + \tau_0 \partial I^{(1)}/\partial x) + v_y (\tau_1 \partial I^{(0)}/\partial y + \tau_0 \partial I^{(1)}/\partial y)) \quad (3)$$

[0126] (1) 에서의 모든 값들은 이제까지 생략되었던 샘플 위치 (i', j') 에 의존한다. 모션이 로컬 환경에서 일관성이 있다고 가정하면, 우리는 현재 예측된 점 (i, j) 에 중심이 맞춰진 $(2M+1) \times (2M+1)$ 정사각형 윈도우 Ω 내의 Δ 를 최소화한다:

$$(v_x, v_y) = \underset{v_x, v_y}{\operatorname{argmin}} \sum_{[i', j'] \in \Omega} \Delta^2[i', j'] \quad (4)$$

이러한 최적화 문제의 경우, 수직 방향에서 및 그 후 수평 방향에서 제 1 최소화를 행하는 단순화된 솔루션이 사용되며, 그것은 다음을 야기한다:

$$v_x = (s_1 + r) > m? \operatorname{clip3}\left(-thBIO, thBIO, -\frac{s_3}{(s_1+r)}\right) : 0 \quad (5)$$

$$v_y = (s_5 + r) > m? \operatorname{clip3}\left(-thBIO, thBIO, -\frac{s_6 - v_x s_2 / 2}{(s_5+r)}\right) : 0 \quad (6)$$

여기서,

$$\begin{aligned} s_1 &= \sum_{[i', j'] \in \Omega} (\tau_1 \partial I^{(0)} / \partial x + \tau_0 \partial I^{(0)} / \partial x)^2; s_3 = \sum_{[i', j'] \in \Omega} (I^{(0)} - I^{(0)}) (\tau_1 \partial I^{(0)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) \\ s_2 &= \sum_{[i', j'] \in \Omega} (\tau_1 \partial I^{(0)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) (\tau_1 \partial I^{(0)} / \partial y + \tau_0 \partial I^{(0)} / \partial y) \\ s_5 &= \sum_{[i', j'] \in \Omega} (\tau_1 \partial I^{(0)} / \partial y + \tau_0 \partial I^{(0)} / \partial y)^2; s_6 = \sum_{[i', j'] \in \Omega} (I^{(0)} - I^{(0)}) (\tau_1 \partial I^{(0)} / \partial y + \tau_0 \partial I^{(0)} / \partial y) \end{aligned} \quad (7)$$

제로 또는 매우 작은 값에 의한 제산을 피하기 위해, 정규화 파라미터들 r 및 m 이 식 (2), (3) 에 도입된다.

$$r = 500 \cdot 4^{d-8} \quad (8)$$

$$m = 700 \cdot 4^{d-8} \quad (9)$$

여기서, d 는 입력 비디오의 내부 비트 깊이이다.

일부 경우들에서, BIO 의 MV 정제는 잡음 또는 불규칙적 모션에 기인하여 신뢰불가능할 것이다. 따라서, BIO 에서, MV 정제의 크기는 특정의 임계값 (thBIO) 으로 클립핑된다. 그 임계값은 현재의 픽처의 모든 참조 픽처들이 모두 한 방향으로부터인지 여부에 기초하여 결정된다. 현재의 픽처의 모든 참조 픽처들이 한 방향으로부터인 경우, 임계값은 $12 \times 2^{14-d}$ 로 설정되고, 그렇지 않으면 그것은 $12 \times 2^{13-d}$ 로 설정된다.

BIO 에 대한 그레디언트들은 HEVC 모션 보상 프로세스 (2D 분리가능 유한 임펄스 응답 (FIR)) 와 일관된 동작들을 사용하여 모션 보상 보간과 동시에 계산된다. 이러한 2D 분리가능 FIR 에 대한 입력은 블록 모션 벡터의 프랙셔널 부분에 따라 모션 보상 프로세스 및 프랙셔널 위치 (fracX, fracY) 에 대해서와 동일한 참조 프레임 샘플이다. 디스케일링 시프트 d-8 을 갖는 프랙셔널 위치 fracY 에 대응하는 BIOfilterS 를 사용하여 수직으로 먼저 보간된 수평 그레디언트 $\partial I / \partial x$ 신호의 경우, 그 후 그레디언트 필터 BIOfilterG 가 18-d 만큼의 디스케일링 시프트를 갖는 프랙셔널 위치 fracX 에 대응하는 수평 방향으로 적용된다. 수직 그레디언트 $\partial I / \partial y$ 의 경우, 먼저 그레디언트 필터가 디스케일링 시프트 d-8 을 갖는 프랙셔널 위치 fracY 에 대응하는 BIOfilterG 를 사용하여 수직으로 적용되고, 그 후 신호 변위가 18-d 만큼의 디스케일링 시프트를 갖는 프랙셔널 위치 fracX 에 대응하여 수평 방향으로 BIOfilterS 를 사용하여 수행된다. 그레디언트 계산 BIOfilterG 및 신호 변위 BIOfilterF 에 대한 보간 필터의 길이는 합리적인 복잡성을 유지하기 위해 더 짧다 (6-탭). 표 1 은 BIO 에서의 블록 모션 벡터의 상이한 프랙셔널 위치들에 대한 그레디언트 계산을 위해 사용되는 필터들을 도시한다. 표 2 는 BIO 에서의 예측 신호 생성에 사용되는 보간 필터들을 도시한다.

도 11 은 8x4 블록에 대한 그레디언트 계산의 예를 도시하는 개념도이다. 8x4 블록의 경우, 비디오 코더는 모션 보상된 예측자를 가져오고 현재 블록 내의 모든 픽셀들 뿐 아니라 외부의 2 개의 라인의 픽셀들의 수평/수직 (HOR/VER) 그레디언트들을 계산하며, 이는 각 픽셀에 대한 v_x 및 v_y 를 푸는 것은 식 (4) 에 도시된 각 픽셀에 중심이 맞춰진 윈도우 Ω 내의 픽셀들의 모션 보상된 예측자들 및 HOR/VER 그레디언트 값이 필요하기 때문이다. 그리고, JEM 의 하나의 예에서, 이러한 윈도우의 크기는 5x5 로 설정된다. 따라서, 비디오 코더 (예를 들어, 비디오 인코더 (20) 및/또는 비디오 디코더 (30)) 는 모션 보상된 예측자를 가져오고 점들 (A 및 B) 주위의 외부의

2 개의 라인의 픽셀들에 대한 그래디언트들을 계산한다.

표 1

Table 1: Filters for gradients calculation in BIO

Fractional pel position	Interpolation filter for gradient(BIOfilterG)
0	{ 8, -39, -3, 46, -17, 5}
1/16	{ 8, -32, -13, 50, -18, 5}
1/8	{ 7, -27, -20, 54, -19, 5}
3/16	{ 6, -21, -29, 57, -18, 5}
1/4	{ 4, -17, -36, 60, -15, 4}
5/16	{ 3, -9, -44, 61, -15, 4}
3/8	{ 1, -4, -48, 61, -13, 3}
7/16	{ 0, 1, -54, 60, -9, 2}
1/2	{ 1, 4, -57, 57, -4, 1}

[0139]

표 2

Table 2: Interpolation filters for prediction signal generation in BIO

Fractional pel position	Interpolation filter for prediction signal(BIOfilterS)
0	{ 0, 0, 64, 0, 0, 0}
1/16	{ 1, -3, 64, 4, -2, 0}
1/8	{ 1, -6, 62, 9, -3, 1}
3/16	{ 2, -8, 60, 14, -5, 1}
1/4	{ 2, -9, 57, 19, -7, 2}
5/16	{ 3, -10, 53, 24, -8, 2}
3/8	{ 3, -11, 50, 29, -9, 2}
7/16	{ 3, -11, 44, 35, -10, 3}
1/2	{ 1, -7, 38, 38, -7, 1}

[0140]

[0141] JEM 의 예들에서, BIO 는 2 개의 예측들이 상이한 참조 픽처들로부터인 경우 모든 양방향 예측된 블록들에 적용된다. 로컬 조명 보상 (LIC) 이 CU 에 대해 가능하게 되는 경우, BIO 는 불가능하게 된다.

[0142] 본 개시의 기법들에 따르면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 (1) 제약이 도출된 MV 가 대칭인지 여부, (2) 초기 MV 와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 대칭인지 여부, (3) 도출된 MV 가 반대칭인지 여부, 및/또는 (4) 초기 MV 와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 반대칭인지 여부에 기초하여 BIO 로부터 도출된 모션 벡터를 제약할 수도 있다. 조건들은 제약이 초기 MV 에 기초하여서만 적용되도록 제약에 배치될 수도 있다.

[0143] 도 12 는 양방향 템플릿 매칭에 기초하여 제안된 디코더측 모션 벡터 도출 (DMVD) 과 관련된 개념들을 도시하는 개념도이다. 비디오 코더는 도 12 에 도시된 바와 같이 각각 List0 의 초기 MV0 및 List1 의 MV1 으로부터 2 개의 예측 블록들 (R₀ 및 R₁) 의 가중된 결합으로서 양방향 템플릿을 생성할 수도 있다. 하나의 예에서, 양방향 템

플릿은 공식: $(R_0+R_1) \gg 1$ 또는 $(R_0+R_1)/2$ 를 사용하여 List0 및 List1 참조 블록들 (R_0 및 R_1) 을 평균화함으로써 생성될 수도 있다.

[0144] 템플릿 매칭 동작은 생성된 템플릿과 참조 픽처 내의 (초기 예측 블록 주위의) 샘플 영역 사이의 코스트 측정을 계산하는 것을 포함한다. 2 개의 참조 픽처들 (R_0 및 R_1) 각각의 경우, 최소 템플릿 코스트를 산출하는 MV 는 초기 MV (List0 로부터의 MV0 또는 List1 으로부터의 MV1) 를 대체하는 해당 리스트의 업데이트된 MV 로서 고려된다. 마지막으로, 도 12 에 도시된 바와 같은 2 개의 새로운 MV 들, 즉 MV0' 및 MV1' 는 정규의 양방향 예측을 위해 사용된다. 그것은 통상 블록-매칭 모션 추정에서 사용되기 때문에, 절대 차이들의 합 (SAD) 이 코스트 측정으로서 이용된다.

[0145] 디코더측 모션 벡터 도출 (DMVD) 은 추가적인 선택스 엘리먼트의 송신 없이, 과거의 참조 픽처로부터 하나 및 미래의 참조 픽처로부터 다른 것으로 양방향 예측의 병합 모드에 대해 적용될 수도 있다. JEM4.0 에서, LIC, 애포인, 서브 CU 병합 후보, 또는 FRUC 가 하나의 CU 에 대해 선택되는 경우, DMVD 가 적용되지 않는다.

[0146] 본 개시의 기법들에 따르면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 (1) 제약이 도출된 MV 가 대칭인지 여부, (2) 초기 MV 와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 대칭인지 여부, (3) 도출된 MV 가 반대칭인지 여부, 및/또는 (4) 초기 MV 와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 반대칭인지 여부에 기초하여 DMVD 에 기초한 양방향 템플릿 매칭으로부터 도출된 모션 벡터를 제약할 수도 있다. 조건들은 제약이 초기 MV 또는 양방향 템플릿에 기초하여서만 적용되도록 제약에 배치될 수도 있다.

[0147] 도 13a 및 도 13b 는 JEM 의 예들에서의 중첩 블록 모션 보상 (OBMC) 과 관련된 개념들을 도시하는 개념도들이다. OBMC 는 예를 들어 H.263 에서와 같이 비디오 표준들의 초기 세대들에 대해 사용되었다. JEM 의 예들에서, OBMC 는 CU 의 우측 및 하측 경계들을 제외한 모션 보상된 (MC) 블록 경계들에 대해 수행된다. 또한, 그것은 루마 및 크로마 컴포넌트들 모두에 대해 적용된다. JEM 의 하나의 예에서, MC 블록은 코딩 블록에 대응한다. CU 가 (서브 CU 병합, 애포인, 및 FRUC 모드를 포함하는) 서브 CU 모드로 코딩되는 경우, CU 의 각 서브 블록은 MC 블록이다. CU 경계들을 균일한 방식으로 프로세싱하기 위해, OBMC 가 모든 MC 블록 경계들에 대해 서브 블록 레벨에서 수행되며, 여기서 서브 블록 사이즈는 도 13a 및 도 13b 에 도시된 바와 같이 4x4 와 동일하게 설정된다.

[0148] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 OBMC 를 현재의 서브 블록에 적용할 수도 있으며, 현재의 모션 벡터들 외에, 4 개의 연결된 이웃 서브 블록들의 모션 벡터들이, 이용가능하고 현재의 모션 벡터와 동일하지 않은 경우, 현재의 서브 블록에 대한 예측 블록을 도출하기 위해 또한 사용된다. 다수의 모션 벡터들에 기초한 이들 다수의 예측 블록들은 현재의 서브 블록의 최종 예측 신호를 생성하기 위해 결합된다.

[0149] 도 14a 내지 도 14d 는 OBMC 가중화들을 도시하는 개념도들이다. 도 14a 내지 도 14d 는 이웃 서브 블록의 모션 벡터들에 기초한 예측 블록이 P_N 으로서 표시되며, N 은 이웃하는 상측 (도 14a), 하측 (도 14b), 좌측 (도 14c), 및 우측 (도 14d) 서브 블록들에 대한 인덱스를 표시하고, 현재 서브 블록의 모션 벡터들에 기초한 예측 블록은 P_C 로서 표시되는 것을 도시한다. P_N 이 현재 서브 블록과 동일한 모션 정보를 포함하는 이웃 서브 블록의 모션 정보에 기초하는 경우, OBMC 는 P_N 으로부터 수행되지 않는다. 그렇지 않은 경우, P_N 의 모든 픽셀이 P_C 내의 동일한 픽셀에 가산되며, 즉 P_N 의 4 개의 행들/열들이 P_C 에 가산된다. 가중 팩터들 $\{1/4, 1/8, 1/16, 1/32\}$ 이 P_N 에 대해 사용되고, 가중 팩터들 $\{3/4, 7/8, 15/16, 31/32\}$ 이 P_C 에 대해 사용된다. 예외는 작은 MC 블록들이며, (즉, 코딩 블록의 높이 또는 폭이 4 와 동일하거나 CU 가 서브 CU 모드로 코딩되는 경우), 그들의 경우 P_N 의 단지 2 개의 행들/열들이 P_C 에 가산된다. 이러한 경우, 가중 팩터들 $\{1/4, 1/8\}$ 은 P_N 에 대해 사용되고, 가중 팩터들 $\{3/4, 7/8\}$ 은 P_C 에 대해 사용된다. 수직으로 (수평으로) 이웃하는 서브 블록의 모션 벡터들에 기초하여 생성된 P_N 의 경우, P_N 의 동일한 행 (열) 에서의 픽셀들은 동일한 가중 팩터와 함께 P_C 에 가산된다. BIO 는 또한 예측 블록 P_n 의 도출을 위해 적용된다.

[0150] JEM 의 예들에서, 256 개의 루마 샘플들 이하의 크기를 같은 CU 의 경우, CU 레벨 플래그가 OBMC 가 현재 CU 에 대해 적용되는지 여부를 나타내기 위해 시그널링된다. 256 개의 루마 샘플들보다 큰 크기를 같거나 AMVP 모드로 코딩되지 않은 CU 의 경우, OBMC 는 디폴트로 적용된다. 인코더에서, OBMC 가 CU 에 대해 적용되는 경우, 그것의 영향이 모션 추정 단계 동안 고려된다. 상측 이웃 블록 및 좌측 이웃 블록의 모션 정보를 사용하는 것에 의한 예측 신호가 현재의 CU 의 오리지널 신호의 상측 및 좌측 경계들을 보상하기 위해 사용되고, 그 후 통상 모

선 추정 프로세스가 적용된다.

[0151] 본 개시의 기법들에 따르면, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 (1) 제약이 도출된 MV 가 대칭인지 여부, (2) 초기 MV 와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 대칭인지 여부, (3) 도출된 MV 가 반대칭인지 여부, 및/또는 (4) 초기 MV 와 DMVD 방법에 의해 도출된 MV 사이의 MV 차이가 반대칭인지 여부에 기초하여 OBMC 로부터 도출된 모션 벡터를 제약할 수도 있다. 조건들은 제약이 초기 MV 에 기초하여서만 적용되도록 제약에 배치될 수도 있다.

[0152] DMVD 관련 방법들 (예를 들어, BIO, FRUC 양방향 매칭, FRUC 템플릿 매칭, 양방향 템플릿 매칭 등) 은 상당한 비트레이트 감소들을 제공할 수도 있다. 본 개시는 DMVD 를 추가로 개선할 수도 있는 수개의 기법들을 기술하며, 이것은 코딩 효율을 증가시키고 및/또는 비트레이트를 감소시킬 수도 있다. 예를 들어, (1) DMVD 에 의해 도출된 MV, 및/또는 (2) 초기 MV 와 DMVD 에 의해 도출된 MV 사이의 MV 차이를 제약하는 것은 코딩 효율을 증가시키고 및/또는 비트레이트를 감소시킬 수도 있다.

[0153] 아래 표 3 은 아래에 사용된 다양한 용어를 설명한다.

표 3

[0154]	<div> L0_POC_Init, L0_MV_X_Init, L0_MV_Y_Init, L1_POC_Init, L1_MV_X_Init, L1_MV_Y_Init </div>	<div> List0 및 List1 에서의 초기 MV 들의 X, Y 성분들 List0 및 List1 에 대한 초기 픽처 순서 카운트 픽처 순서 카운트는 HEVC에 정의된 POC 일 수도 있다. </div>
	<div> L0_POC_Derived, L0_MV_X_Derived, L0_MV_Y_Derived, L1_POC_Derived, L1_MV_X_Derived, L1_MV_Y_Derived </div>	<div> L0 및 L1에 대한 DMVD 방법으로 도출된 MV 들의 X, Y 성분들 List0 및 List1 에 대한 초기 픽처 순서 카운트 픽처 순서 카운트는 HEVC에 정의된 POC 일 수도 있다. </div>
	<div> L0_MV_X_Diff, L0_MV_Y_Diff, L1_MV_X_Diff, L1_MV_Y_Diff </div>	<div> List0 및 List1 의 DMVD 방법에 의한 초기 MV와 도출된 MV 간의 MV 차이의 X, Y 성분 </div>
	CURR_POC	<div> 현재 슬라이스/픽처의 픽처 순서 카운트 (예를 들어, HEVC 에 정의된 바와 같은 POC) </div>

[0155] 대칭 모션 벡터 — 일부 예에서, DMVD 에 의해 유도된 MV 는 유도된 MV 가 대칭이어야 하도록 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 에 의해 제약될 수도 있다. 일 예에서, 대칭이기 위해, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 도출된 MV 가 서로에 대해 다음의 조건들 중 하나 이상 (예를 들어, 모두) 을 충족시키도록 도출된 MV 를 제약하도록 구성될 수도 있다:

[0156] 1. $L0_MV_X_Derived * (CURR_POC - L1_POC_Derived) = L1_MV_X_Derived * (CURR_POC - L0_POC_Derived)$, 및/또는

[0157] 2. $L0_MV_Y_Derived * (CURR_POC - L1_POC_Derived) = L1_MV_Y_Derived * (CURR_POC - L0_POC_Derived)$.

[0158] 일부 예에서, 상기 조건들 (1) 및 (2) 는 각각 별개의 대칭 MV 제약으로 간주될 수도 있거나, 또는 둘 다 적용될 때 단일 대칭 MV 제약으로 간주될 수도 있다. 이와 관련하여, 본 명세서에서 인에이블되는 제약에 대한 언급은 인에이블되는 조건을 지칭할 수 있으며, 이는 제약으로서 동작한다. 예를 들어, 인에이블되는 대칭 MV 제약에 대한 언급은 (i) 인에이블되는 상기 조건 (1), (ii) 인에이블되는 상기 조건 (2), 또는 (iii) 인에이블되는 조건들 (1) 및 (2) 를 포함할 수도 있다.

[0159] 일부 예들에서, 추가적인 단순화 및/또는 코딩 개선을 달성하기 위해, DMVD 에 의해 유도된 MV 는 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 에 의해 제약되어, 유도된 MV 가 (의사 대칭 조건으로서 지칭될 수도 있는) 아래의 조건들 중 하나 이상 (예를 들어, 모두, 조건 (1) 이 참인 경우 조건들 (a) 및/

또는 (b), 및/또는 조건 (2) 가 참인 경우 조건들 (c) 및/또는 (d)) 을 충족시키도록 요구될 수 있다.

- [0160] 1. List0 및 List1 참조 픽처들이 현재 픽처의 모두 앞에 또는 모두 뒤에 있을 때 (예를 들어, $(CURR_POC - L0_POC_Derived) * (CURR_POC - L1_POC_Derived) > 0$),
- [0161] a. $L0_MV_X_Derived = L1_MV_X_Derived$, 및/또는
- [0162] b. $L0_MV_Y_Derived = L1_MV_Y_Derived$;
- [0163] 및/또는
- [0164] 2. List0 및 List1 참조 픽처들이 현재 픽처의 모두 앞에 또는 모두 뒤에 없을 때 (예를 들어, $(CURR_POC - L0_POC_Derived) * (CURR_POC - L1_POC_Derived) < 0$),
- [0165] c. $L0_MV_X_Derived = -1 * L1_MV_X_Derived$, 및/또는
- [0166] d. $L0_MV_Y_Derived = -1 * L1_MV_Y_Derived$.
- [0167] 일부 예에서, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 예를 들어 초기 MV 의 POC 정보 및/또는 유도된 MV 의 POC 정보를 사용하도록 구성됨으로써 POC 차이를 체크 또는 결정하도록 구성될 수도 있다. 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 예를 들어 초기 MV 의 POC 정보 및/또는 유도된 MV 의 POC 정보에 기초하여 POC 차이를 체크 또는 결정하도록 구성될 수도 있다.
- [0168] 대칭 모션 벡터 차이들— 일부 예에서, 초기 MV 와 DMVD 에 의해 유도된 MV 사이의 MV 차이는 MV 차이가 대칭이어야 하도록 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 에 의해 제약될 수도 있다. MV 차이는 초기 MV 와 DMVD 에 의한 유도된 MV 사이의 차이이다. MV 차이는 X 성분 및 Y 성분에 기초하여 결정될 수도 있다. 일 예에서, 대칭이기 위해, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 도출된 MV 차이가 다음의 조건들 중 하나 이상 (예를 들어, 모두) 을 충족시킬 것을 요구하도록 구성될 수도 있다:
- [0169] 1. $L0_MV_X_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_X_Diff * (CURR_POC - L0_POC_Derived)$, 및/또는
- [0170] 2. $L0_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived)$.
- [0171] 일부 예들에서, 추가적인 단순화 및/또는 코딩 개선을 달성하기 위해, DMVD 에 의해 유도된 MV 차이는 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 에 의해 제약되어, 유도된 MV 차이가 의사 대칭 조건으로서 지칭될 수도 있는) 아래의 조건들 중 하나 이상 (예를 들어, 모두, 조건 (1) 이 참인 경우 조건들 (a) 및/또는 (b), 및/또는 조건 (2) 가 참인 경우 조건들 (c) 및/또는 (d)) 을 충족시키도록 요구될 수 있다:
- [0172] 1. List0 및 List1 참조 픽처들이 현재 픽처의 모두 앞에 또는 모두 뒤에 있을 때 (예를 들어, $(CURR_POC - L0_POC_Init) * (CURR_POC - L1_POC_Init) > 0$),
- [0173] a. $L0_MV_X_Diff = L1_MV_X_Diff$, 및/또는
- [0174] b. $L0_MV_Y_Diff = L1_MV_Y_Diff$;
- [0175] 및/또는
- [0176] 2. List0 및 List1 참조 픽처들이 현재 픽처의 모두 앞에 또는 모두 뒤에 있지 않을 때 (예를 들어, $(CURR_POC - L0_POC_Init) * (CURR_POC - L1_POC_Init) < 0$),
- [0177] c. $L0_MV_X_Diff = -1 * L1_MV_X_Diff$, 및/또는
- [0178] d. $L0_MV_Y_Diff = -1 * L1_MV_Y_Diff$.
- [0179] 일부 예에서, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 예를 들어 초기 MV 의 POC 정보 및/또는 유도된 MV 의 POC 정보를 사용하도록 구성됨으로써 POC 차이를 체크 또는 결정하도록 구성될 수도 있다. 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 예를 들어 초기 MV 의 POC 정보 및/또는 유도된 MV 의 POC 정보에 기초하여 POC 차이를 체크 또는 결정하도록 구성될 수도

있다.

[0180] 반대칭 모션 벡터 — 일부 예에서, DMVD 에 의해 유도된 MV 는 유도된 MV 가 반대칭이어야 하도록 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 에 의해 제약될 수도 있다. 일 예에서, 반대칭이기 위해, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 도출된 MV 가 다음의 조건들 중 하나 이상 (예를 들어, 모두) 을 충족시킬 것을 요구하도록 구성될 수도 있다:

[0181] 1. $L0_MV_X_Derived * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_X_Derived * (CURR_POC - L0_POC_Derived)$, 및/또는

[0182] 2. $L0_MV_Y_Derived * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_Y_Derived * (CURR_POC - L0_POC_Derived)$.

[0183] 일부 예들에서, 추가적인 단순화 및/또는 코딩 개선을 달성하기 위해, DMVD 에 의해 유도된 MV 차이는 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 에 의해 제약되어, 유도된 MV 차이가 (의사 반대칭 조건으로서 지칭될 수도 있는) 아래의 조건들 중 하나 이상 (예를 들어, 모두, 조건 (1) 이 참인 경우 조건들 (a) 및/또는 (b), 및/또는 조건 (2) 가 참인 경우 조건들 (c) 및/또는 (d)) 을 충족시키도록 요구될 수 있다:

[0184] 1. List0 및 List1 참조 픽처가 현재 픽처의 모두 앞에 또는 모두 뒤에 있을 때 (예를 들어, $(CURR_POC - L0_POC_Derived) * (CURR_POC - L1_POC_Derived) > 0$),

[0185] a. $L0_MV_X_Derived = -1 * L1_MV_X_Derived$, 및/또는

[0186] b. $L0_MV_Y_Derived = -1 * L1_MV_Y_Derived$;

[0187] 및/또는

[0188] 2. List0 및 List1 참조 픽처들이 현재 픽처의 모두 앞에 또는 모두 뒤에 있지 않을 때 (예를 들어, $(CURR_POC - L0_POC_Derived) * (CURR_POC - L1_POC_Derived) < 0$),

[0189] c. $L0_MV_X_Derived = L1_MV_X_Derived$, 및/또는

[0190] d. $L0_MV_Y_Derived = L1_MV_Y_Derived$.

[0191] 일부 예에서, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 예를 들어 초기 MV 의 POC 정보 및/또는 유도된 MV 의 POC 정보를 사용하도록 구성됨으로써 POC 차이를 체크 또는 결정하도록 구성될 수도 있다. 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 예를 들어 초기 MV 의 POC 정보 및/또는 유도된 MV 의 POC 정보에 기초하여 POC 차이를 체크 또는 결정하도록 구성될 수도 있다.

[0192] 반대칭 모션 벡터 차이들 — 일부 예에서, 초기 MV 와 DMVD 에 의해 유도된 MV 사이의 MV 차이는 MV 차이가 반대칭이어야 하도록 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 에 의해 제약될 수도 있다. 일 예에서, 반대칭이기 위해, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 도출된 MV 차이가 다음의 조건들 중 하나 이상 (예를 들어, 모두) 을 충족시킬 것을 요구하도록 구성될 수도 있다:

[0193] 1. $L0_MV_X_Diff * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_X_Diff * (CURR_POC - L0_POC_Derived)$, 및/또는

[0194] 2. $L0_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived)$.

[0195] 일부 예들에서, 추가적인 단순화 및/또는 코딩 개선을 달성하기 위해, DMVD 에 의해 유도된 MV 차이는 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 에 의해 제약되어, 유도된 MV 차이가 (의사 반대칭 조건으로서 지칭될 수도 있는) 아래의 조건들 중 하나 이상 (예를 들어, 모두, 조건 (1) 이 참인 경우 조건들 (a) 및/또는 (b), 및/또는 조건 (2) 가 참인 경우 조건들 (c) 및/또는 (d)) 을 충족시키도록 요구될 수 있다:

[0196] 1. List0 및 List1 참조 픽처가 현재 픽처의 모두 앞에 또는 모두 뒤에 있을 때 (예를 들어, $(CURR_POC - L0_POC_Init) * (CURR_POC - L1_POC_Init) > 0$),

- [0197] a. $LO_MV_X_Diff = -1 * L1_MV_X_Diff$, 및/또는
- [0198] b. $LO_MV_Y_Diff = -1 * L1_MV_Y_Diff$;
- [0199] 및/또는
- [0200] 2. List0 및 List1 참조 픽처들이 현재 픽처의 모두 앞에 또는 모두 뒤에 있지 않을 때 (예를 들어, $(CURR_POC - LO_POC_Init) * (CURR_POC - L1_POC_Init) < 0$),
- [0201] c. $LO_MV_X_Diff = L1_MV_X_Diff$, 및/또는
- [0202] d. $LO_MV_Y_Diff = L1_MV_Y_Diff$.
- [0203] 일부 예에서, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 예를 들어 초기 MV 의 POC 정보 및/또는 유도된 MV 의 POC 정보를 사용하도록 구성됨으로써 POC 차이를 체크 또는 결정하도록 구성될 수도 있다. 예를 들어, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 예를 들어 초기 MV 의 POC 정보 및/또는 유도된 MV 의 POC 정보에 기초하여 POC 차이를 체크 또는 결정하도록 구성될 수도 있다.
- [0204] MV/MV 차이 제약들을 적용하기 위한 조건들 — DMVD 모션 정보에 대한 제약들 (예를 들어, 대칭/반대칭 MV/MV 차이 제약들) 의 인에이블먼트는, 일부 예에서, 명시적으로 시그널링될 수도 있거나 일부 코딩된 정보에 따라 암시적으로 결정될 수도 있다. 일부 예들에서, 제약의 인에이블먼트를 시그널링하는 것 또는 제약의 인에이블먼트를 결정하는 것은 각각 제약이 적용되었거나 적용되어야 한다고 시그널링하는 것 또는 제약이 적용되었거나 적용되어야 한다고 결정하는 것으로 기술될 수도 있다.
- [0205] 일부 예들에서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 본원에 설명된 MV 및/또는 MV 차이 제약들 중 하나 이상을 인에이블하도록 구성될 수도 있다. 예를 들어, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 하나 이상의 제약들이 인에이블되는지 여부를 나타내는 하나 이상의 값들을 시그널링하도록 구성될 수도 있다.
- [0206] 일례로서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 제 1 신택스 엘리먼트에 대응하는 비트 스트림에서의 값을 시그널링하도록 구성될 수도 있고, 여기서 제 1 신택스 엘리먼트에 대응하는 값은 하나 이상의 대칭 MV 제약들이 인에이블되는지 여부를 나타낸다. 그 값은 인에이블먼트를 나타내는 참 값 (예를 들어, 1 의 값) 이거나, 디스에이블먼트를 나타내는 거짓 값 (예를 들어, 0 의 값) 일 수도 있거나, 그 값은 인에이블먼트를 나타내는 거짓 값 (예를 들어, 0 의 값) 또는 디스에이블먼트를 나타내는 참 값 (예를 들어, 1 의 값) 일 수도 있다. 일부 예들에서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 하나 이상의 대칭 MV 제약들이 인에이블되는 때만 제 1 신택스 엘리먼트에 대응하는 값을 시그널링하도록 구성될 수도 있다. 다른 예들에서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 하나 이상의 대칭 MV 제약들이 디스에이블되는 때만 제 1 신택스 엘리먼트에 대응하는 값을 시그널링하도록 구성될 수도 있다.
- [0207] 다른 예로서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 제 2 신택스 엘리먼트에 대응하는 비트 스트림에서 값을 시그널링하도록 구성될 수도 있고, 여기서 제 2 신택스 엘리먼트에 대응하는 값은 하나 이상의 대칭 MV 차이 제약들이 인에이블되는지 여부를 나타낸다. 그 값은 인에이블먼트를 나타내는 참 값 (예를 들어, 1 의 값) 이거나, 디스에이블먼트를 나타내는 거짓 값 (예를 들어, 0 의 값) 일 수도 있거나, 그 값은 인에이블먼트를 나타내는 거짓 값 (예를 들어, 0 의 값) 또는 디스에이블먼트를 나타내는 참 값 (예를 들어, 1 의 값) 일 수도 있다. 일부 예들에서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 하나 이상의 대칭 MV 차이 제약들이 인에이블되는 때만 제 2 신택스 엘리먼트에 대응하는 값을 시그널링하도록 구성될 수도 있다. 다른 예들에서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 하나 이상의 대칭 MV 차이 제약들이 디스에이블되는 때만 제 2 신택스 엘리먼트에 대응하는 값을 시그널링하도록 구성될 수도 있다.
- [0208] 다른 예로서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 제 3 신택스 엘리먼트에 대응하는 비트 스트림에서 값을 시그널링하도록 구성될 수도 있고, 여기서 제 3 신택스 엘리먼트에 대응하는 값은 하나 이상의 반대칭 MV 제약들이 인에이블되는지 여부를 나타낸다. 그 값은 인에이블먼트를 나타내는 참 값 (예를 들어, 1 의 값) 이거나, 디스에이블먼트를 나타내는 거짓 값 (예를 들어, 0 의 값) 일 수도 있거나, 그 값은 인에이블먼트를 나타내는 거짓 값 (예를 들어, 0 의 값) 또는 디스에이블먼트를 나타내는 참 값 (예를 들어, 1 의 값) 일 수도 있다. 일부 예들에서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 하나 이상의 반대칭 MV 제약들이 인에이블되는 때만 제 3 신택스 엘리먼트에 대응하는 값을 시그널링하도록 구성될 수도 있다. 다른 예들에

서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 하나 이상의 반대칭 MV 제약들이 디스에이블되는 때만 제 3 선택스 엘리먼트에 대응하는 값을 시그널링하도록 구성될 수도 있다.

- [0209] 다른 예로서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 제 4 선택스 엘리먼트에 대응하는 비트 스트림에서 값을 시그널링하도록 구성될 수도 있고, 여기서 제 4 선택스 엘리먼트에 대응하는 값은 하나 이상의 반대칭 MV 차이 제약들이 인에이블되는지 여부를 나타낸다. 그 값은 인에이블먼트를 나타내는 참 값 (예를 들어, 1 의 값) 이거나, 디스에이블먼트를 나타내는 거짓 값 (예를 들어, 0 의 값) 일 수도 있거나, 그 값은 인에이블먼트를 나타내는 거짓 값 (예를 들어, 0 의 값) 또는 디스에이블먼트를 나타내는 참 값 (예를 들어, 1 의 값) 일 수도 있다. 일부 예들에서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 하나 이상의 반대칭 MV 차이 제약들이 인에이블되는 때만 제 4 선택스 엘리먼트에 대응하는 값을 시그널링하도록 구성될 수도 있다. 다른 예들에서, 비디오 인코더 (예를 들어, 비디오 인코더 (20)) 는 하나 이상의 반대칭 MV 차이 제약들이 디스에이블되는 때만 제 4 선택스 엘리먼트에 대응하는 값을 시그널링하도록 구성될 수도 있다.
- [0210] 일부 예들에서, 비디오 코더 (예를 들어, 비디오 디코더 (30)) 는 비트 스트림을 수신하고, 여기에 설명된 MV 및/또는 MV 차이 제약들 중 하나 이상이 하나 이상의 선택스 엘리먼트들 (예를 들어, 본 명세서에 설명된 제 1, 제 2, 제 3 및 제 4 선택스 엘리먼트들 중 하나 이상) 에 대응하는 비트 스트림에 존재하는 하나 이상의 값들에 기초하여 인에이블되는지 여부를 결정하도록 구성될 수도 있다.
- [0211] 다른 예들에서, 비디오 코더 (예를 들어, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 본 명세서에 설명된 MV 및/또는 MV 차이 제약들 중 하나 이상이 다른 코딩된 정보에 기초하여 인에이블되는지 여부를 결정 (예를 들어, 암시 적으로 결정) 하도록 구성될 수도 있다.
- [0212] 본 명세서에 기술된 하나 이상의 제약들 (예를 들어, 대칭/반대칭 MV/MV 차이 제약들 중 하나 이상) 을 적응 적으로 인에이블하는 몇몇 예시적인 조건들이 아래에 나열되어 있다.
- [0213] 1. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 가 대칭인 경우에만 인에이블될 수도 있다.
- [0214] 2. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 가 의사 대칭인 경우에만 인에이블될 수도 있다.
- [0215] 3. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 가 반대칭인 경우에만 인에이블될 수도 있다.
- [0216] 4. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 가 의사 반대칭인 경우에만 인에이블될 수도 있다.
- [0217] 5. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 가 대칭이 아닌 경우에만 인에이블될 수도 있다.
- [0218] 6. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 가 의사 대칭이 아닌 경우에만 인에이블될 수도 있다.
- [0219] 7. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 가 반대칭이 아닌 경우에만 인에이블될 수도 있다.
- [0220] 8. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 가 의사 반대칭이 아닌 경우에만 인에이블될 수도 있다.
- [0221] 9. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 의 List0 및 List1 참조 픽처들이 현재 픽처의 모두 앞에 또는 모두 뒤에 있는 경우에만 인에이블될 수도 있다.
- [0222] 10. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 의 List0 및 List1 참조 픽처들이 현재 픽처의 모두 앞에 또는 모두 뒤에 있지 않은 경우에만 인에이블될 수도 있다.
- [0223] 11. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 들의 참조 인덱스가 모두 제로인 경우에만 인에이블될 수도 있다.

- [0224] 12. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 초기 MV 들의 참조 인덱스가 모두 제로가 아닌 경우에만 인에이블될 수도 있다.
- [0225] 13. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 List0 참조 픽처와 현재 픽처 사이의 POC 거리들 및 List1 참조 픽처와 현재 픽처 사이의 POC 거리들이 동일한 경우에만 인에이블될 수도 있다.
- [0226] 14. 일부 예들에서, 본 명세서에 설명된 하나 이상의 제약들 (예를 들어, 하나 이상의 대칭/반대칭 MV/MV 차이 제약들) 은 List0 참조 픽처와 현재 픽처 사이의 POC 거리들 및 List1 참조 픽처와 현재 픽처 사이의 POC 거리들이 동일하지 않은 경우에만 인에이블될 수도 있다.
- [0227] 15. 일부 예들에서, 하나 이상의 대칭 MV 차이 제약들은 초기 MV 들이 대칭이 아닐 때 적용될 수도 있다,
- [0228] 16. 일부 예들에서, 하나 이상의 반대칭 MV 차이 제약들은 초기 MV 들이 대칭일 때 적용될 수도 있다,
- [0229] 17. 일부 예들에서, 대칭 제약들 중 하나 이상의 MV 해상도는 결정론적 방식으로 할당될 수도 있다. 일 예에서, 양방향 템플릿 매칭의 정수-픽셀 정제의 경우, 대칭/의사 대칭/비대칭 제약이 부과되지 않을 수도 있으며, 전술 한 제약은 단지 하프-픽셀, 1/4-픽셀 또는 고정밀 모션 정제에 대한 것이다. 제약의 수준 (MV 해상도)은 SPS/PPS/슬라이스 헤더를 통해 시그널링될 수도 있다. 해상도는 또한 BIO, 서브-픽셀 FRUC 과 같은 다른 모션 정제 도구가 인에이블 되는지 여부와 함께 결정될 수도 있다. 예를 들어, BIO 가 가능하게 되는 경우, 제약은 1/4-픽셀 또는 더 높은-정밀도 모션 정제에 부과되지 않을 수도 있다.
- [0230] 18. 일부 예들에서, 제약의 수준 (예를 들어, 어떤 제약들이 인에이블되는지) 은 List0 과 List1 MV 들 사이의 절대 차이, List0 과 List1 MV 들 사이의 (그들의 상대 POC 거리에 기초하여) 스케일링된 절대 차이, 또는 List0 및 List1 의 보간된 샘플들 (즉, 이 예에서는 P0 및 P1) 사이의 초기 SAD 값들에 따라 적응적이게 될 수 있다. 다른 예들에서, (현재 템플릿, P0) 과 (현재 템플릿, P1) 사이의 절대 차이의 합 (SAD) 값의 비율은 어떤 MV 해상도에서 제약이 부과되어야 하는지를 결정하는데 사용될 수 있다. 이 예에서, SAD 값의 비가 임계 값보다 낮으면, 전술한 대칭 제약이 부과되지 않아야 한다.
- [0231] 표 4 는 전술한 조건들에 기초한 적용 조건들 및 제약들이 충족되는 예들을 열거한다.
- [0232] 이 예는 표 4 의 제약 예 1로 설명되어 있으며, 다른 제약들의 다른 예들도 포함되어 있다.

표 4

[표 4] 조건과 제약의 예

제약 예	조건	적용되는 제약
1	초기 MV는 대칭이다	제약 없음
	초기 MV는 대칭이 아니다	의사 대칭 MV 차이
2	초기 MV는 대칭이다	제약 없음
	초기 MV는 대칭이 아니다	대칭 MV 차이
3	초기 MV는 의사 대칭이다	제약 없음
	초기 MV는 의사 대칭이 아니다	의사 대칭 MV 차이
4	초기 MV는 의사 대칭이다	제약 없음
	초기 MV는 의사 대칭이 아니다	대칭 MV 차이
5	초기 MV는 대칭이다	비의사 대칭 MV 차이
	초기 MV는 대칭이 아니다	의사 대칭 MV 차이
6	초기 MV는 대칭이다	비대칭 MV 차이
	초기 MV는 대칭이 아니다	대칭 MV 차이
7	초기 MV는 의사 대칭이다	비의사 대칭 MV 차이
	초기 MV는 의사 대칭이 아니다	의사 대칭 MV 차이
8	초기 MV는 의사 대칭이다	비대칭 MV 차이
	초기 MV는 의사 대칭이 아니다	대칭 MV 차이

- [0234] 도 15 는 본 개시에 따른 양방향 템플릿 매칭의 예시적인 사용 사례를 예시한다. 도 15 는 본 명세서에 설명된 하나 이상의 의사 대칭 MV 차이 제약이 초기 MV가 대칭이 아닌 경우에만 양방향 템플릿 매칭에 적용될 수 있는 예를 도시한다. 이러한 예에 따르면, 양측 템플릿 매칭의 경우에, 초기 MV 가 대칭이 아닌 경우, 최종 MV

는 초기 MV 와 최종 MV 사이의 MV 차이가 의사 대칭이 되도록 유도될 수도 있다. 예를 들어, 도 15 에 나타낸 바와 같이, 양방향 템플릿 (1506) 은 먼저 R_0 및 R_1 을 평균화함으로써 List0 ($L0_MV_Init$, 블록 R_0 (1502) 을 가리킴) 및 List1 ($L1_MV_Init$, 블록 R_1 (1504) 을 가리킴) 에서의 초기 MV 들 (예를 들어, 양방향 예측 병합 후보의 MV 들) 를 사용하여 도출될 수도 있고, 그 후 양방향 템플릿 (1506) 은 최상의 List0/List1 MV 들을 발견 하기 위해 List0 및 List1 에서 최상의 매칭 블록들을 검색하는데 이용될 수도 있다. 양방향 템플릿 매칭은 블록들, 예를 들어 R'_0 (1508) 및 R'_1 (1510) 을 검색하는 List0/List1 의 초기 MV 들을 중심으로 미리 정의된 검색 범위 (예를 들어, -1 내지 +1 정수 픽셀) 내에서 수행될 수도 있다. 초기 블록들 (블록 R_0 (1502) 와 블록 R_1 (1504)) 과 검색된 블록들 (R'_0 (1508) 및 R'_1 (1510)) 사이의 차이는 각각의 MV 차이 ($L0_MV_Diff$ 및 $L1_MV_Diff$ 로 표시됨) 로 도시된다. 일부 예들에서, List0 및 List1 에서 ($L0_MV_Diff$ 및 $L1_MV_Diff$ 를 비교하는) 의사 대칭 MV 차이를 갖는 MV 쌍들만이 유효한 MV 들로 간주될 수도 있다. 최상의 매칭 코스트를 갖는 유효한 MV 쌍은 그 후 양방향 템플릿 매칭에 의해 최종 유도된 MV 들로서 간주 될 수도 있다. 예시된 예는 상 기 도시된 표 4 에서 제약 예 1 로 표시된다.

[0235] 도 16 는 본 개시의 기법들에 따른 예시적인 방법을 나타내는 플로우차트이다. 도 16 의 기법들은 목적지 디바 이스 (14) 또는 비디오 디코더 (30) 중 하나 이상에 의해 구현될 수도 있다 (도 1 및 도 3 참조).

[0236] 본 개시의 일 예에서, 비디오 디코더 (30) 는 디코더 측 모션 벡터 도출 (DMVD) 을 이용하여 복수의 도출된 모 션 벡터들을 생성하도록 구성될 수도 있다 (100). 일부 예에서, 복수의 도출된 모션 벡터를 생성하기 위해 DMVD 를 사용하는 것은 BIO 프로세스, FRUC 프로세스, 양방향 매칭 프로세스, FRUC 템플릿 매칭 프로세스, 또는 양방향 템플릿 매칭 프로세스 중 하나를 사용하는 것을 포함한다. 일부 예들에서, 비디오 디코더 (30) 는 제 1 및 제 2 초기 모션 벡터들이 하나 이상의 조건들을 만족하는지의 결정에 기초하여 복수의 도출된 모션 벡터들을 생성할 수도 있다. 비디오 디코더 (30) 는 코스트 메트릭 및 제약이 충족되는지를 결정하는 것에 기초하여 복 수의 도출된 모션 벡터들로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하도록 구성될 수도 있 다 (102). 예를 들어, 코스트 메트릭은 절대 차이들의 합을 포함한다. 제약은 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 갖는지, 의사 대칭 모션 벡터 차이를 갖는지, 반대칭인지, 반대칭 모션 벡터 차를 갖는지, 또는 의사 반대칭 모션 벡터 차이를 갖는지 여부를 포함할 수도 있다. 비디오 디코더 (30) 는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 갖는지, 의사 대칭 모션 벡 터 차이를 갖는지, 반대칭인지, 반대칭 모션 벡터 차를 갖는지, 또는 의사 반대칭 모션 벡터 차이를 갖는지 여 부를 결정하도록 구성될 수도 있다. 제약은 비트 스트림을 통해 시그널링 될 수 있거나, 도출되거나, 미리 프 로그램 될 수도 있다.

[0237] 일 예에서, 비디오 디코더 (30) 는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 갖는 것으로 결정하도록 구성될 수도 있으며, 상기 결정하는 것은 다음을 결정하는 것을 포함한다: $L0_MV_X_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_X_Diff * (CURR_POC - L0_POC_Derived)$ 또는 $L0_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived)$, 여기서, $L0_MV_X_Diff$ 는 제 1 초기 모션 벡터와 제 1 도출된 모션 벡터 간의 차이의 X 성분이고, $L0_MV_Y_Diff$ 는 제 1 초기 모션 벡터와 제 1 도출된 모션 벡터 간의 차이의 Y 성분이고, $L1_MV_X_Diff$ 는 제 2 초기 모션 벡터와 제 2 도출된 모션 벡터 간의 차이의 X 성분이고, $L1_MV_Y_Diff$ 는 제 2 초기 모션 벡터와 제 2 도출된 모션 벡터 간의 차이의 Y 성분이고,

[0238] $CURR_POC$ 는 현재 픽처의 현재 POC 이고, $L0_POC_Derived$ 는 제 1 리스트의 모션 벡터들에 대한 도출된 POC 이고, $L1_POC_Derived$ 는 제 2 리스트의 모션 벡터들에 대한 도출된 POC 이다.

[0239] 다른 예에서, 비디오 디코더 (30) 는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭 모션 벡터 차 이를 갖는 것으로 결정하도록 구성될 수도 있으며, 상기 결정하는 것은 다음을 결정하는 것을 포함한다: $L0_MV_X_Diff * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_X_Diff * (CURR_POC - L0_POC_Derived)$, 또는 $L0_MV_Y_Diff * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_Y_Diff * (CURR_POC - L0_POC_Derived)$, 여기서, $L0_MV_X_Diff$ 는 제 1 초기 모션 벡터와 제 1 도출된 모션 벡터 간의 차이의 X 성분이고, $L0_MV_Y_Diff$ 는 제 1 초기 모션 벡터와 제 1 도출된 모션 벡터 간의 차이의 Y 성분이고, $L1_MV_X_Diff$ 는 제 2 초기 모션 벡터와 제 2 도출된 모션 벡터 간의 차이의 X 성분이고, $L1_MV_Y_Diff$ 는 제 2 초기 모션 벡터와 제 2 도출된 모션 벡터 간의 차이의 Y 성분이고,

[0240] $CURR_POC$ 는 현재 픽처의 현재 POC 이고, $L0_POC_Derived$ 는 제 1 리스트의 모션 벡터들에 대한 도출된 POC

이고, L1_POC_Derived 는 제 2 리스트의 모션 벡터들에 대한 도출된 POC 이다.

- [0241] 다른 예에서, 비디오 디코더 (30) 는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 반대칭이라고 결정하도록 구성될 수도 있으며, 상기 결정하는 것은 다음을 결정하는 것을 포함한다: $LO_MV_X_Derived * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_X_Derived * (CURR_POC - LO_POC_Derived)$, 또는 $LO_MV_Y_Derived * (CURR_POC - L1_POC_Derived) = -1 * L1_MV_Y_Derived * (CURR_POC - LO_POC_Derived)$, 여기서, $LO_MV_X_Derived$ 는 제 1 도출된 모션 벡터의 X 성분이고, $LO_MV_Y_Derived$ 는 제 1 도출된 모션 벡터의 Y 성분이고, $L1_MV_X_Derived$ 는 제 2 도출된 모션 벡터의 X 성분이고, $L1_MV_Y_Derived$ 는 제 2 도출된 모션 벡터의 Y 성분이고,
- [0242] $CURR_POC$ 는 현재 픽처의 현재 POC 이고, $LO_POC_Derived$ 는 제 1 리스트의 모션 벡터들에 대한 도출된 POC 이고, $L1_POC_Derived$ 는 제 2 리스트의 모션 벡터들에 대한 도출된 POC 이다.
- [0243] 다른 예에서, 비디오 디코더 (30) 는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 대칭 모션 벡터 차이를 갖는 것으로 결정하도록 구성될 수도 있으며, 상기 결정하는 것은 다음을 결정하는 것을 포함한다: $LO_MV_X_Diff = L1_MV_X_Diff$, $LO_MV_Y_Diff = L1_MV_Y_Diff$, $LO_MV_X_Diff = -1 * L1_MV_X_Diff$, 또는 $LO_MV_Y_Diff = -1 * L1_MV_Y_Diff$; 여기서 : $LO_MV_X_Diff$ 는 제 1 초기 모션 벡터와 제 1 도출된 모션 벡터 간의 차이의 X 성분이고, $LO_MV_Y_Diff$ 는 제 1 초기 모션 벡터와 제 1 도출된 모션 벡터 간의 차이의 Y 성분이고, $L1_MV_X_Diff$ 는 제 2 초기 모션 벡터와 제 2 도출된 모션 벡터 간의 차이의 X 성분이고, $L1_MV_Y_Diff$ 는 제 2 초기 모션 벡터와 제 2 도출된 모션 벡터 간의 차이의 Y 성분이다.
- [0244] 다른 예에서, 비디오 디코더 (30) 는 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 의사 반대칭 모션 벡터 차이를 갖는 것으로 결정하도록 구성될 수도 있으며, 상기 결정하는 것은 다음의 결정을 포함한다: $LO_MV_X_Diff = L1_MV_X_Diff$, $LO_MV_Y_Diff = L1_MV_Y_Diff$, $LO_MV_X_Diff = -1 * L1_MV_X_Diff$, 또는 $LO_MV_Y_Diff = -1 * L1_MV_Y_Diff$; 여기서 : $LO_MV_X_Diff$ 는 제 1 초기 모션 벡터와 제 1 도출된 모션 벡터 간의 차이의 X 성분이고, $LO_MV_Y_Diff$ 는 제 1 초기 모션 벡터와 제 1 도출된 모션 벡터 간의 차이의 Y 성분이고, $L1_MV_X_Diff$ 는 제 2 초기 모션 벡터와 제 2 도출된 모션 벡터 간의 차이의 X 성분이고, $L1_MV_Y_Diff$ 는 제 2 초기 모션 벡터와 제 2 도출된 모션 벡터 간의 차이의 Y 성분이다.
- [0245] 다른 예에서, 비디오 디코더 (30) 는 모션 벡터 후보리스트의 제 1 초기 모션 벡터 및 모션 벡터 후보리스트의 제 2 초기 모션 벡터가 제 1 초기 모션 벡터와 제 2 초기 모션 벡터 사이의 관계에 대한 적어도 하나의 조건을 만족시키는 것으로 결정하도록 구성될 수도 있고, 상기 관계는 제 1 초기 모션 벡터 및 제 2 초기 모션 벡터가 대칭인 것, 제 1 초기 모션 벡터 및 제 2 초기 모션 벡터가 의사 대칭인 것, 제 1 초기 모션 벡터 및 제 2 초기 모션 벡터가 대칭 모션 벡터 차이를 가지는 것, 제 1 초기 모션 벡터 및 제 2 초기 모션 벡터가 반대칭인 것, 제 1 초기 모션 벡터 및 제 2 초기 모션 벡터가 의사 반대칭인 것, 또는 제 1 초기 모션 벡터 및 제 2 초기 모션 벡터가 반대칭 모션 벡터 차이를 갖는 것을 포함하며, 여기서, 복수의 도출된 모션 벡터를 생성하는 것은 적어도 하나의 조건을 만족시키는 제 1 초기 모션 벡터 및 제 2 초기 모션 벡터에 기초한다.
- [0246] 다른 예에서, 비디오 디코더 (30) 는 모션 벡터 후보 리스트의 제 1 초기 모션 벡터 및 모션 벡터 후보 리스트의 제 2 초기 모션 벡터가 현재 픽처 이전인지 또는 현재 픽처 이후인지를 결정하도록 구성될 수도 있으며, 여기서 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 것은 그 결정에 더 기초한다.
- [0247] 다른 예에서, 비디오 디코더 (30) 는 제 1 모션 벡터 후보 리스트와 연관된 제 1 참조 픽처와 현재 픽처 사이의 제 1 픽처 순서 카운트 (POC) 거리가 제로이고 제 2 모션 벡터 후보 리스트와 연관된 제 2 참조 픽처와 현재 픽처 사이의 제 2 POC 거리가 제로인지 여부를 결정하도록 구성될 수도 있고, 여기서 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 것은 그 결정에 더 기초한다.
- [0248] 다른 예에서, 비디오 디코더 (30) 는 제 1 모션 벡터 후보 리스트와 연관된 제 1 참조 픽처와 현재 픽처 사이의 제 1 픽처 순서 카운트 (POC) 거리가 제로이고 제 2 모션 벡터 후보 리스트와 연관된 제 2 참조 픽처와 현재 픽처 사이의 제 2 POC 거리가 제로인지 여부를 결정하도록 구성될 수도 있고, 여기서 복수의 도출된 모션 벡터들을 생성하는 코스트 메트릭에 기초하여 복수의 도출된 모션 벡터로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하는 것은
- [0249] 제 1 모션 벡터 후보 리스트와 연관된 제 1 참조 픽처와 현재 픽처 사이의 제 1 POC 거리가 제로이고 제 2 모션 벡터 후보 리스트와 연관된 제 2 참조 픽처와 현재 픽처 사이의 제 2 POC 거리가 제로인지 여부를 결정하는 것

에 더 기초한다.

- [0250] 다른 예에서, 비디오 디코더 (30) 는 인코딩된 비디오 비트 스트림으로부터 적어도 하나의 제약을 결정하도록 구성될 수도 있다.
- [0251] 비디오 디코더 (30) 는 결정된 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 이용하여 현재 블록을 디코딩하도록 구성될 수도 있다 (104).
- [0252] 도 17 는 본 개시의 기법들에 따른 예시적인 방법을 나타내는 플로우차트이다. 도 17 의 기법들은 소스 디바이스 (12) 또는 비디오 인코더 (20) 중 하나 이상에 의해 구현될 수도 있다 (도 1 및 도 2 참조).
- [0253] 비디오 인코더 (20) 는 DMVD 를 이용하여 복수의 도출된 모션 벡터들을 생성하도록 구성될 수도 있다 (200). 일부 예들에서, 비디오 인코더 (20) 는 제 1 및 제 2 초기 모션 벡터들이 하나 이상의 조건들을 만족하는지의 결정에 기초하여 복수의 도출된 모션 벡터들을 생성할 수도 있다.
- [0254] 비디오 인코더 (20) 는 코스트 메트릭 및 제약이 충족되는지를 결정하는 것에 기초하여 복수의 도출된 모션 벡터들로부터 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 결정하도록 구성될 수도 있다 (202). 제약은 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터가 대칭 모션 벡터 차이를 갖는지, 의사 대칭 모션 벡터 차이를 갖는지, 반대칭인지, 반대칭 모션 벡터 차를 갖는지, 또는 의사 반대칭 모션 벡터 차이를 갖는지 여부를 포함할 수도 있다. 비디오 인코더 (20) 는 결정된 제 1 도출된 모션 벡터 및 제 2 도출된 모션 벡터를 이용하여 현재 블록을 인코딩하도록 구성될 수도 있다 (204). 비디오 인코더 (20) 는 제약을 코딩하도록 구성될 수도 있다. 비디오 인코더 (20) 는 또한 비트스트림을 통해 제약을 전송하도록 구성될 수도 있다.
- [0255] 예에 의존하여, 본 명세서에서 설명된 기법들의 임의의 특정 행위들 또는 이벤트들은 상이한 시퀀스로 수행될 수 있고, 전체적으로 부가되거나 병합되거나 또는 제거될 수도 있음 (예를 들어, 설명된 모든 행위들 또는 이벤트들이 그 기법들의 실시를 위해 필수적인 것은 아님) 이 인식되어야 한다. 더욱이, 특정 예들에 있어서, 행위들 또는 이벤트들은 순차적인 것보다는, 예를 들어, 다중-스레딩된 프로세싱, 인터럽트 프로세싱, 또는 다중의 프로세서들을 통해 동시에 수행될 수도 있다.
- [0256] 하나 이상의 예들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 그 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현되면, 그 기능들은 컴퓨터 판독가능 매체 상의 하나 이상의 명령 또는 코드로서 저장되거나 송신될 수도 있고 하드웨어 기반 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터 판독가능 매체는, 데이터 저장 매체와 같은 유형의 매체에 대응하는 컴퓨터 판독가능 저장 매체, 또는 예를 들면, 통신 프로토콜에 따라, 일 장소로부터 다른 장소로의 컴퓨터 프로그램의 전송을 가능하게 하는 임의의 매체를 포함하는 통신 매체를 포함할 수도 있다. 이런 방식으로, 컴퓨터 판독가능 매체는 일반적으로, (1) 비일시적인 유형의 컴퓨터 판독가능 저장 매체 또는 (2) 신호 또는 캐리어 파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체들은 본 개시에서 설명된 기법들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 추출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 이용가능한 매체들일 수도 있다. 컴퓨터 프로그램 제품이 컴퓨터 판독가능 매체를 포함할 수도 있다.
- [0257] 제한이 아닌 일 예로, 이러한 컴퓨터 판독가능 저장 매체들은 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 스토리지, 자기 디스크 스토리지, 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 명령들 또는 데이터 구조들의 형태로 원하는 프로그램 코드를 저장하는데 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 커넥션이 컴퓨터 판독가능 매체로 적절히 명명된다. 예를 들어, 동축 케이블, 광섬유 케이블, 꼬임쌍선, 디지털 가입자 라인 (DSL), 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들을 이용하여 웹사이트, 서버, 또는 다른 원격 소스로부터 소프트웨어가 송신된다면, 동축 케이블, 광섬유 케이블, 꼬임쌍선, DSL, 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들은 매체의 정의에 포함된다. 하지만, 컴퓨터 판독가능 저장 매체 및 데이터 저장 매체는 접속, 캐리어 파, 신호 또는 다른 일시적 매체를 포함하는 것이 아니라, 대신에 비일시적, 유형의 저장 매체에 관련된다는 것이 이해되어야 한다. 본 명세서에서 사용된 바와 같은 디스크 (disk) 및 디스크 (disc) 는 콤팩트 디스크 (CD), 레이저 디스크, 광학 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크 및 블루레이 디스크를 포함하며, 여기서, 디스크 (disk) 는 통상적으로 데이터를 자기적으로 재생하지만 디스크 (disc) 는 레이저를 이용하여 데이터를 광학적으로 재생한다. 상기의 조합들이 또한, 컴퓨터 판독가능 매체들의 범위 내에 포함되어야 한다.
- [0258] 명령들은 하나 이상의 프로세서, 이를테면 하나 이상의 DSP (digital signal processor), 범용 마이크로프로세서, (고정 기능 회로 및/또는 프로그램가능 프로세싱 회로를 포함하는) 프로세싱 회로, ASIC (application

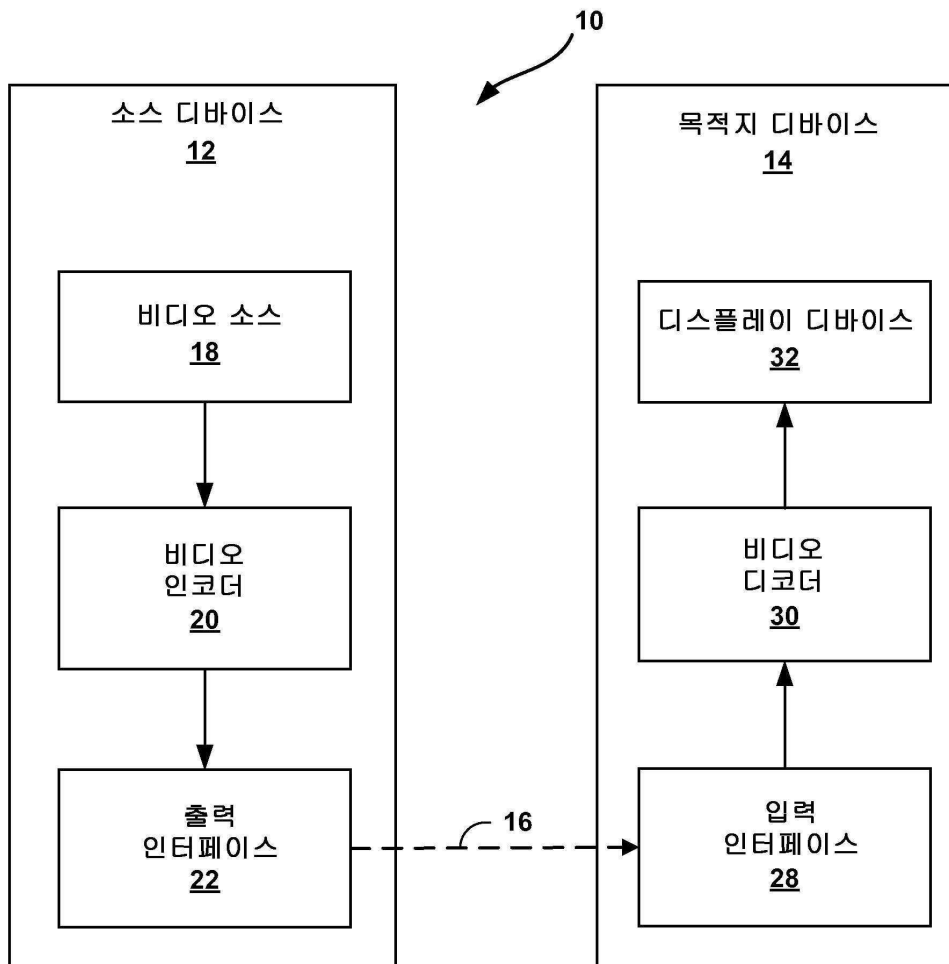
specific integrated circuit), FPGA (field programmable gate array), 또는 다른 등가 집적 또는 이산 로직 회로에 의해 실행될 수도 있다. 이에 따라, 본 명세서에서 사용된 바와 같은 용어 "프로세서"는 본 명세서에서 설명된 기법들의 구현에 적합한 전술한 구조 또는 임의의 다른 구조 중 임의의 구조를 지칭할 수도 있다. 추가로, 일부 양태들에서, 본 명세서에서 설명된 기능성은 인코딩 및 디코딩을 위해 구성되거나, 또는 결합된 코덱에 통합된 전용 하드웨어 및/또는 소프트웨어 모듈들 내에 제공될 수도 있다. 또한, 그 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수 있다.

[0259] 본 개시의 기법들은 무선 핸드셋, 집적 회로 (IC) 또는 IC 들의 세트 (예를 들면, 칩 세트)를 포함하는, 매우 다양한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들, 모듈들, 또는 유닛들은 개시된 기법들을 수행하도록 구성된 디바이스들의 기능적 양태들을 강조하기 위해 본 개시에 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 요구하는 것은 아니다. 오히려, 상기 설명된 바와 같이, 다양한 유닛들은 코덱 하드웨어 유닛에서 결합되거나 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 상기 설명된 바와 같은 하나 이상의 프로세서들을 포함하는, 상호동작가능한 하드웨어 유닛들의 컬렉션에 의해 제공될 수도 있다.

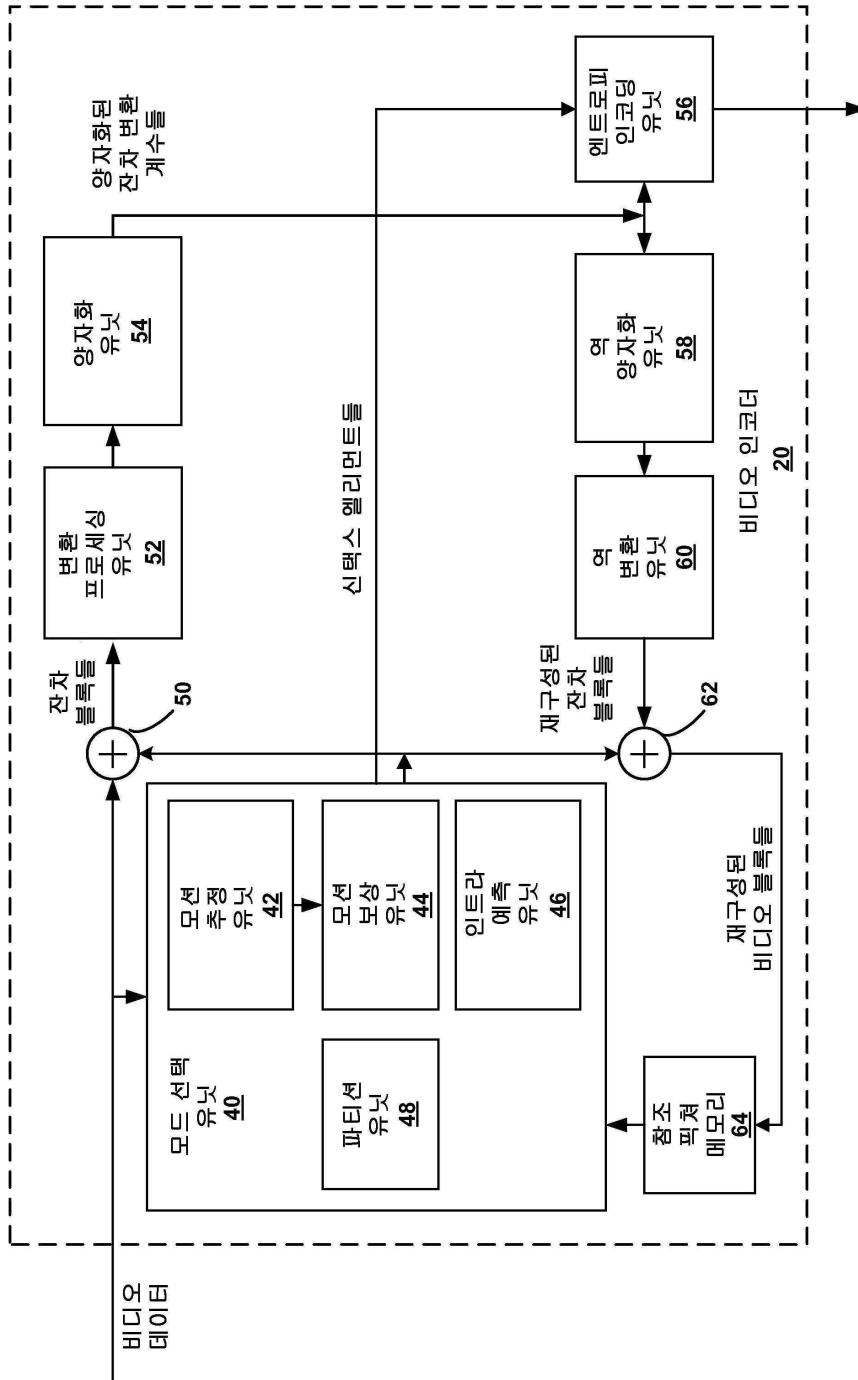
[0260] 다양한 예들이 설명되었다. 이들 및 다른 예들은 다음의 청구항들의 범위 내에 있다.

도면

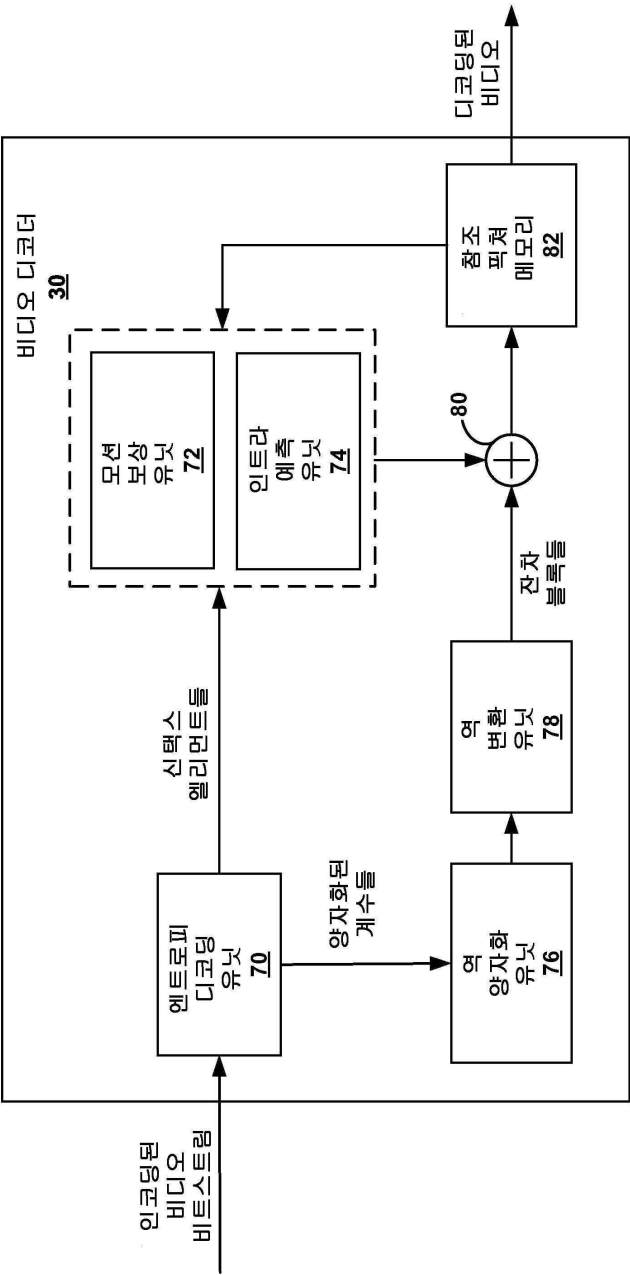
도면1



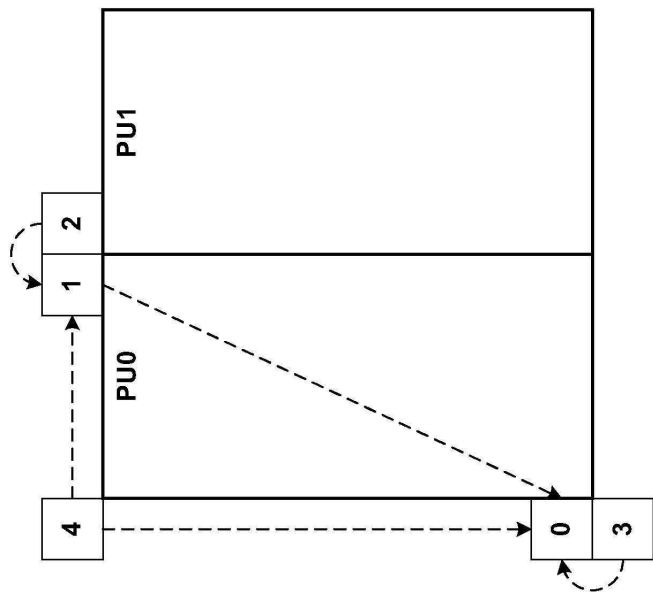
도면2



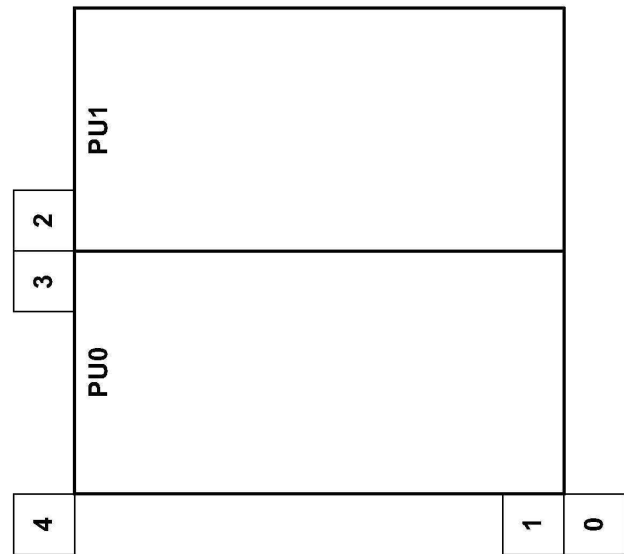
도면3



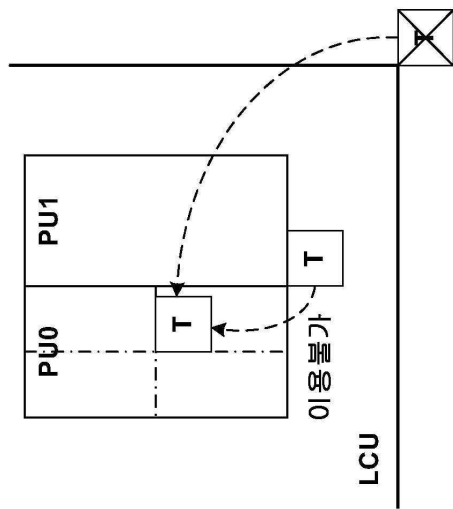
도면4a



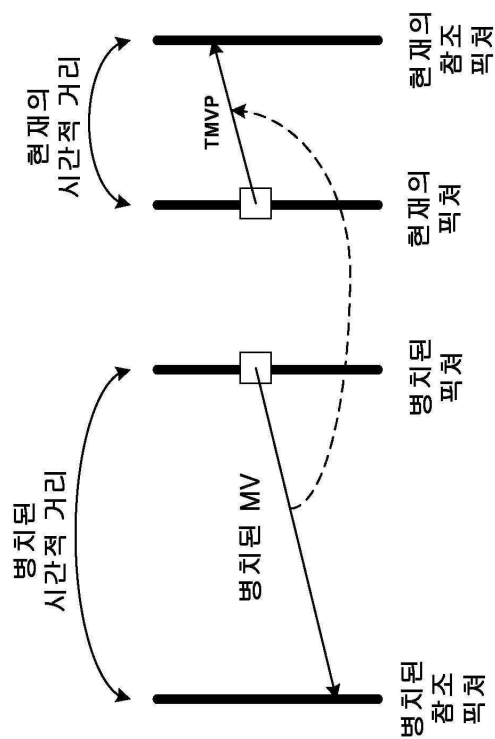
도면4b



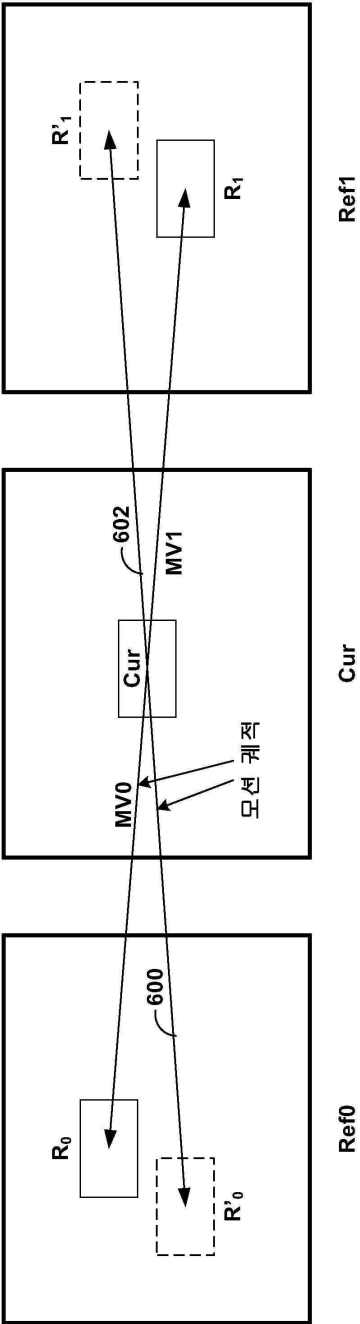
도면5a



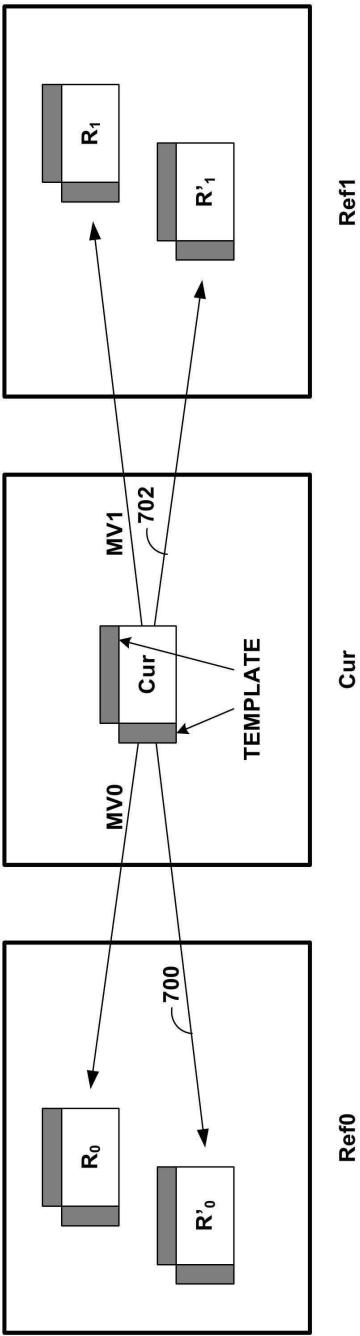
도면5b



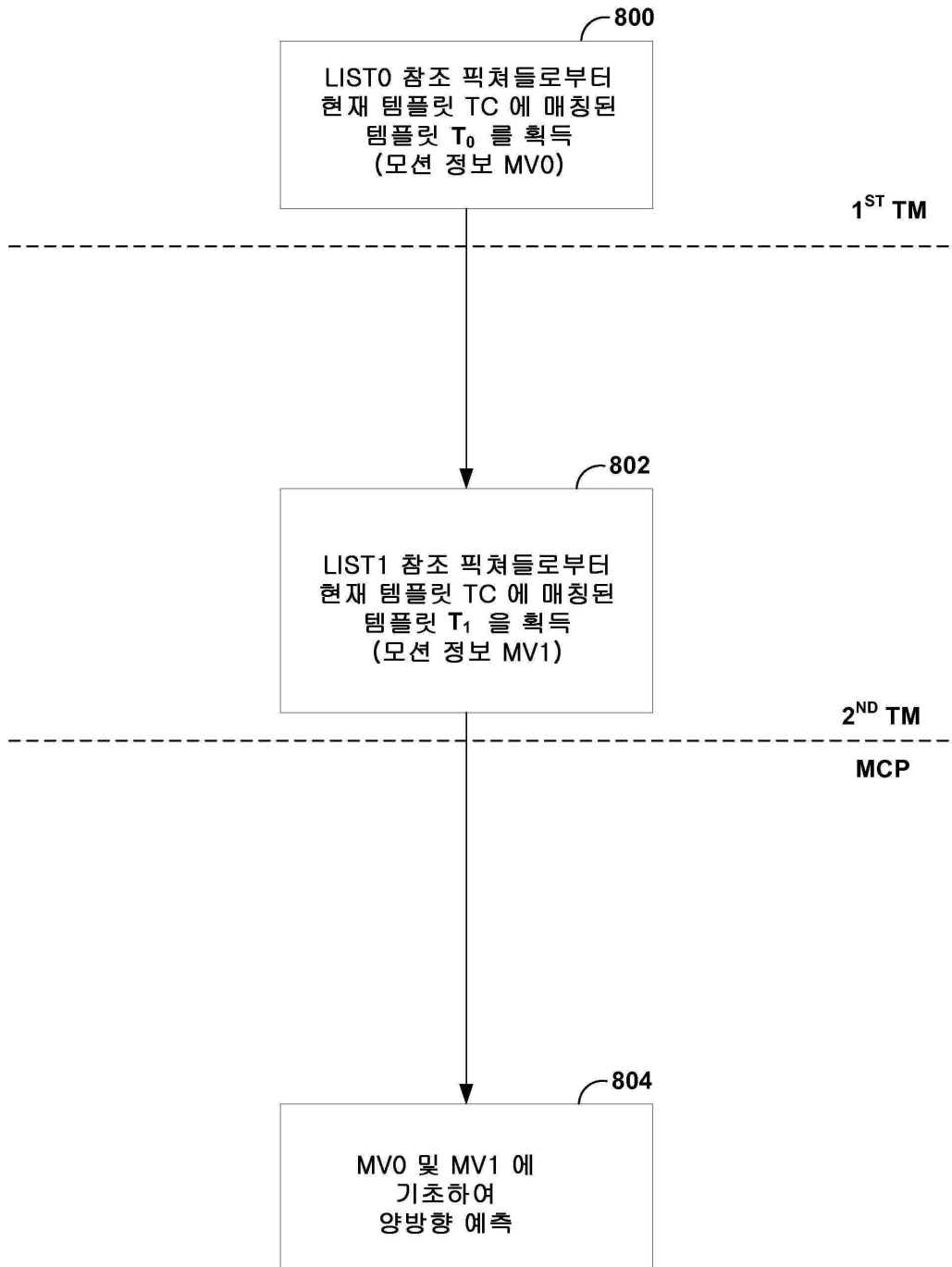
도면6



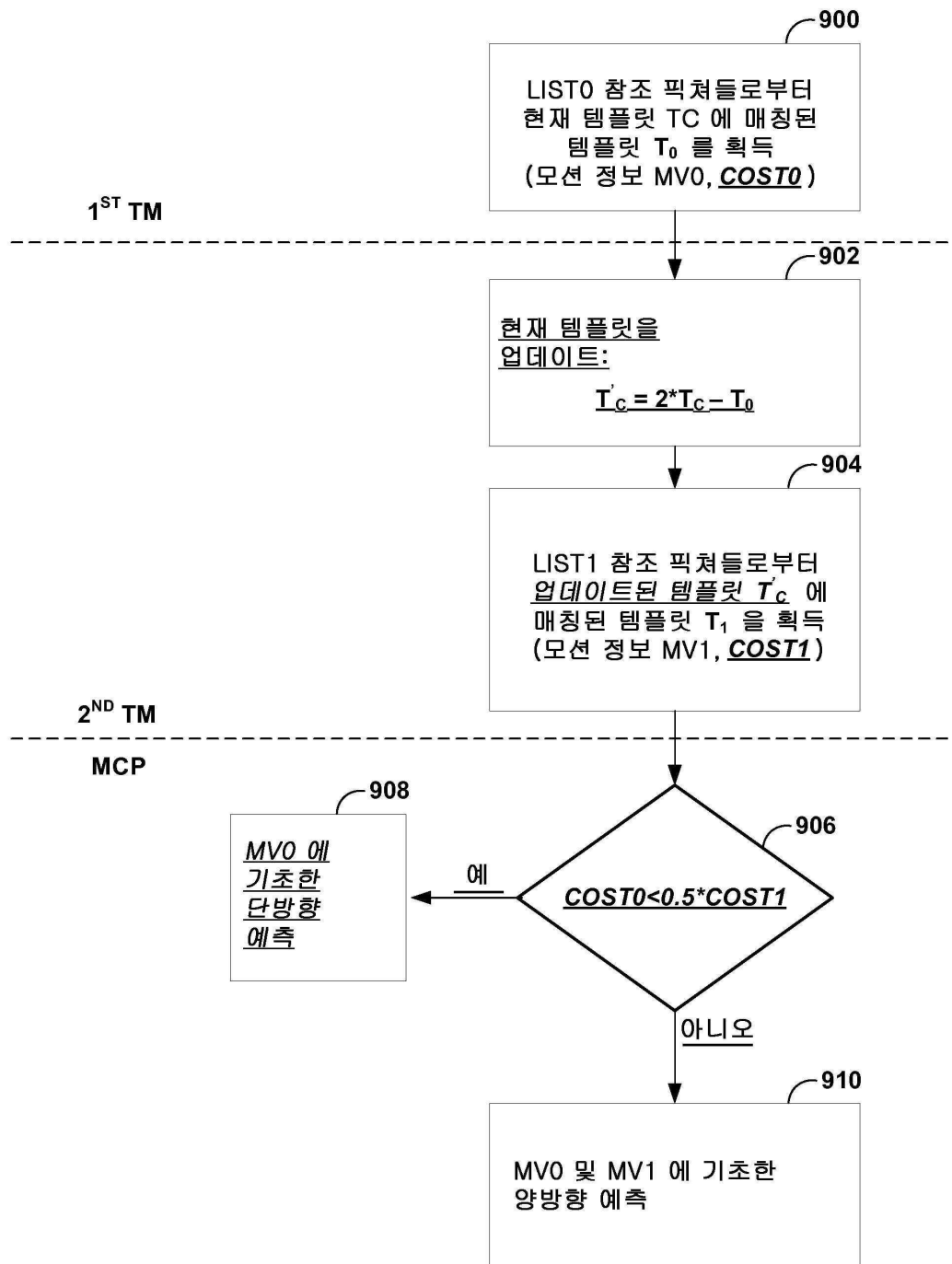
도면7



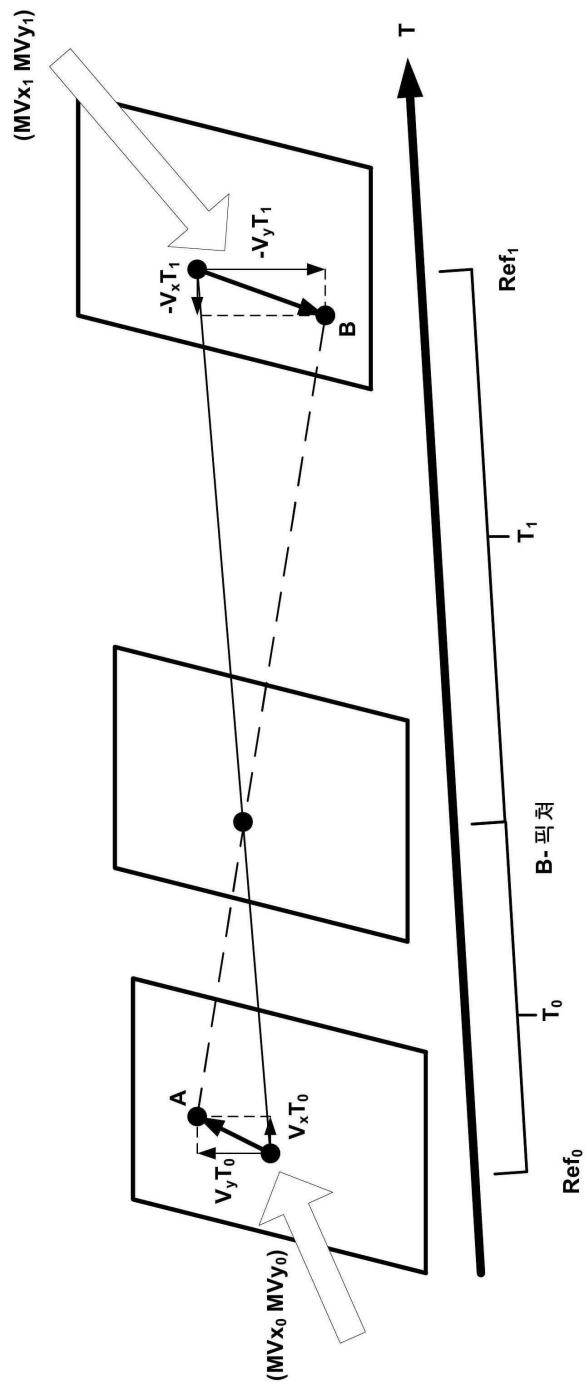
도면8



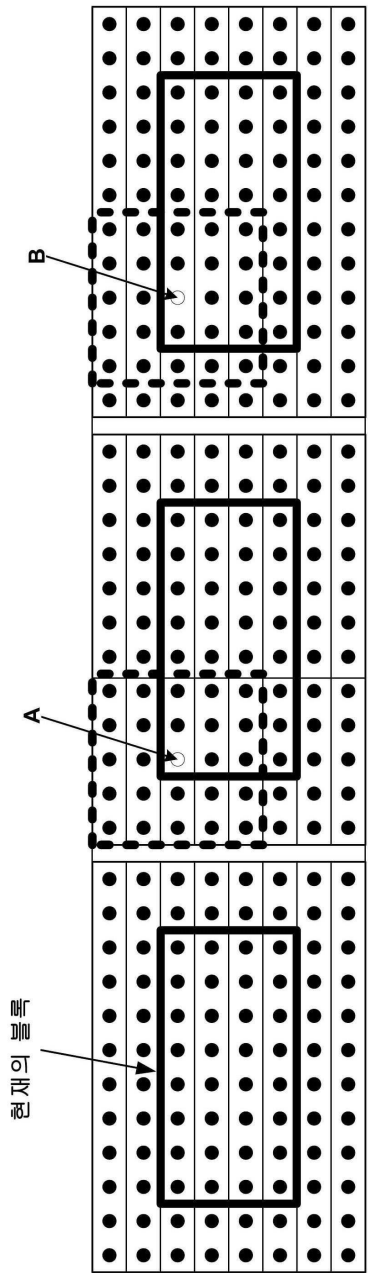
도면9



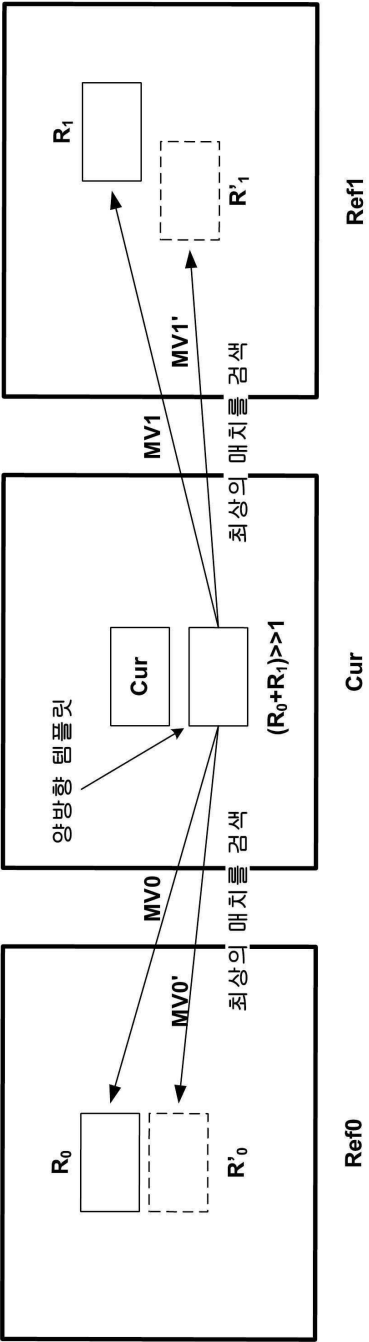
도면10



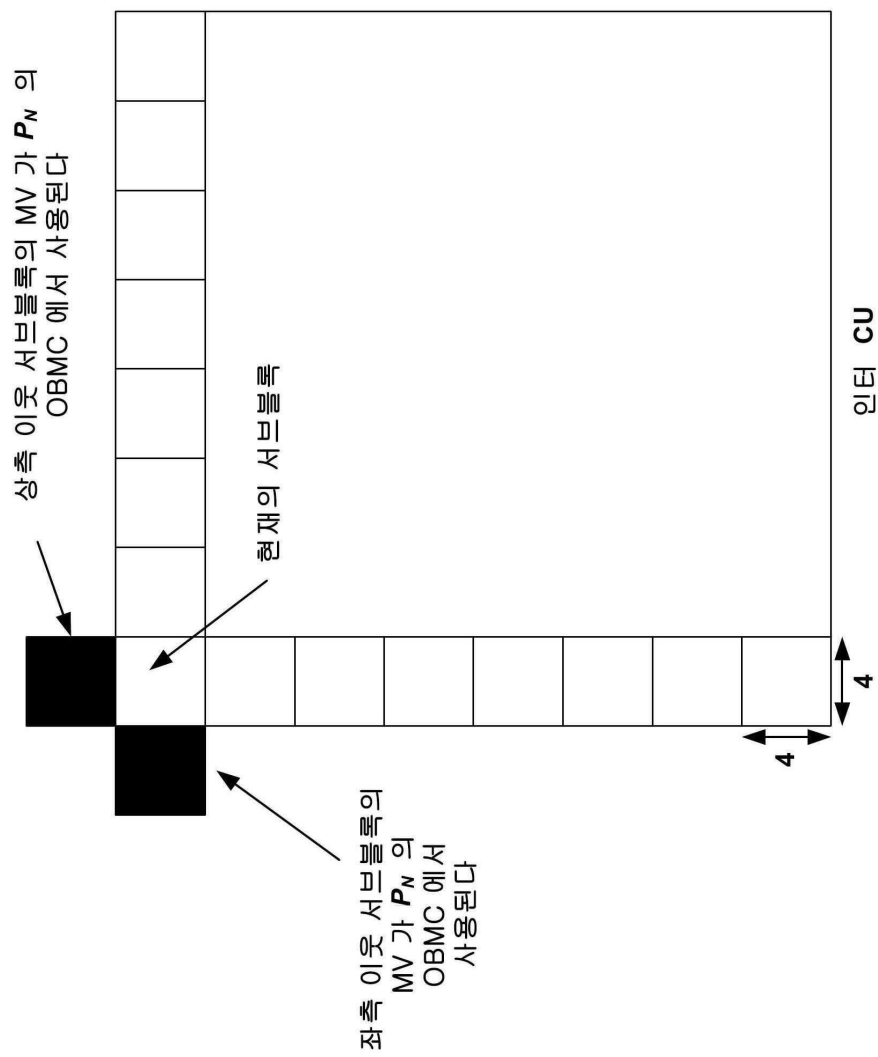
도면11



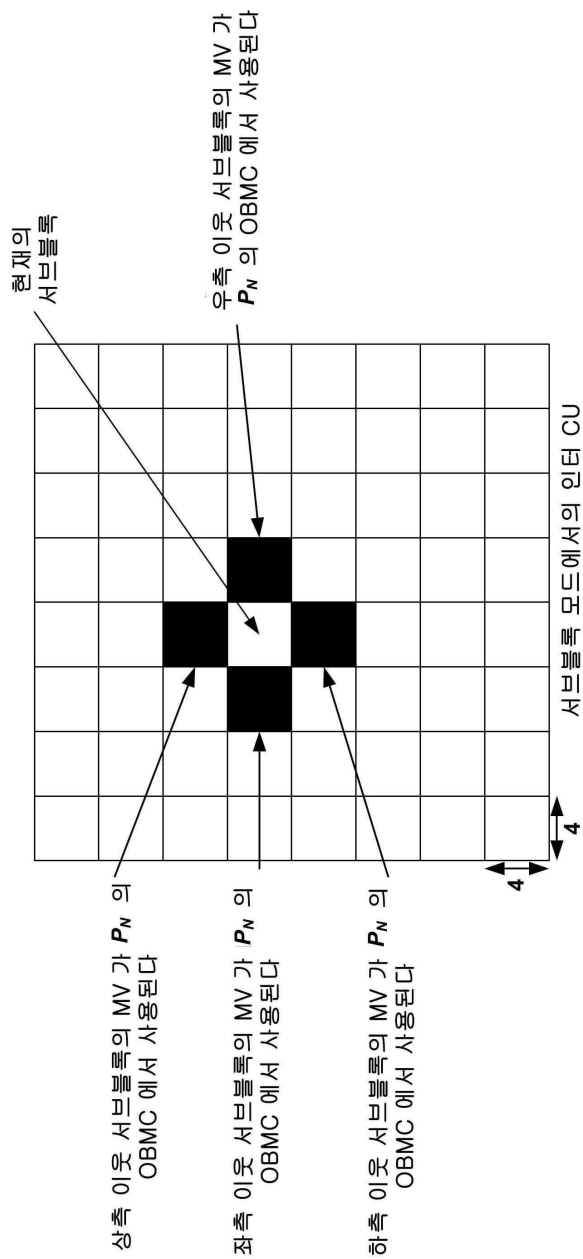
도면12



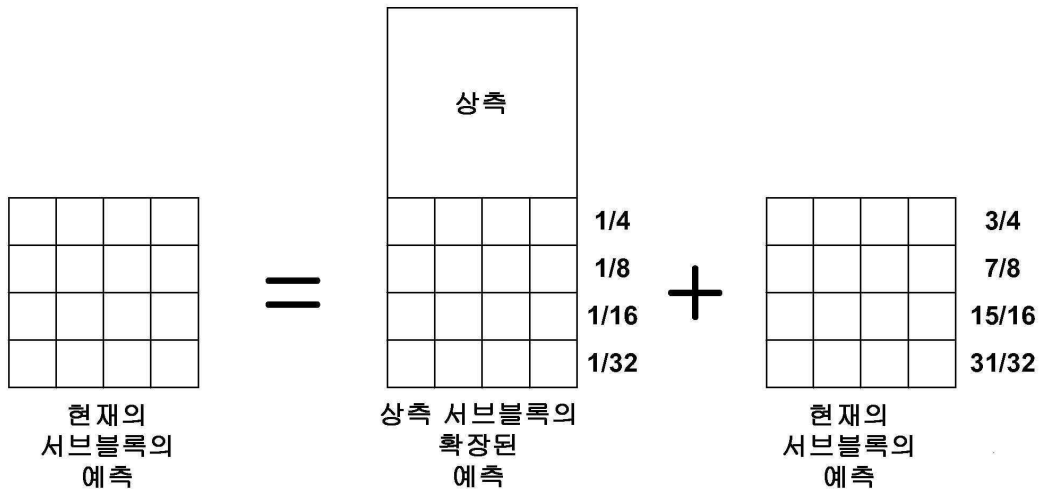
도면13a



도면13b



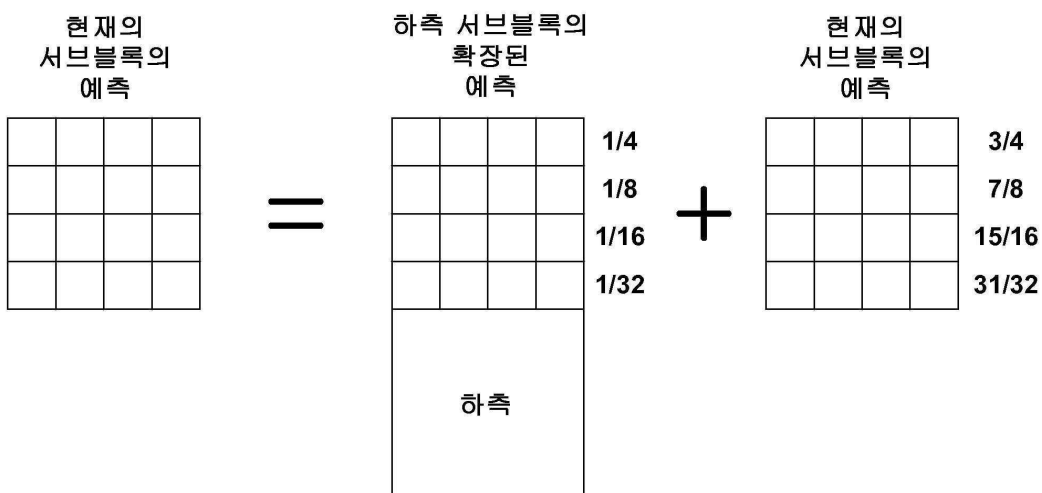
도면14a



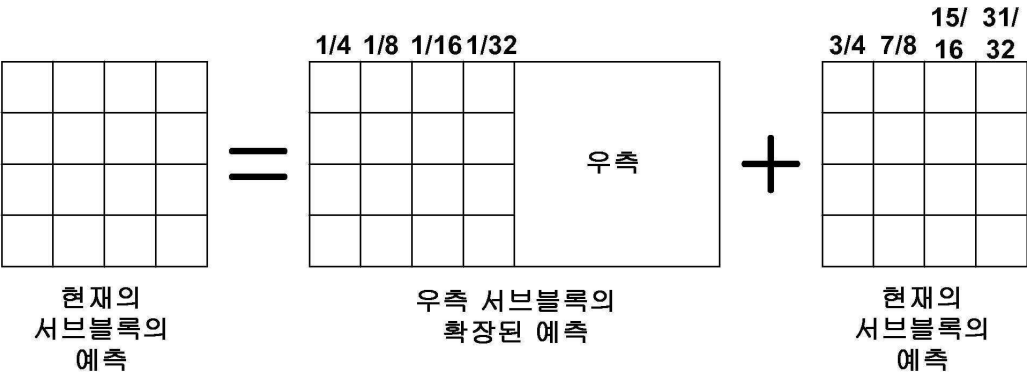
도면14b



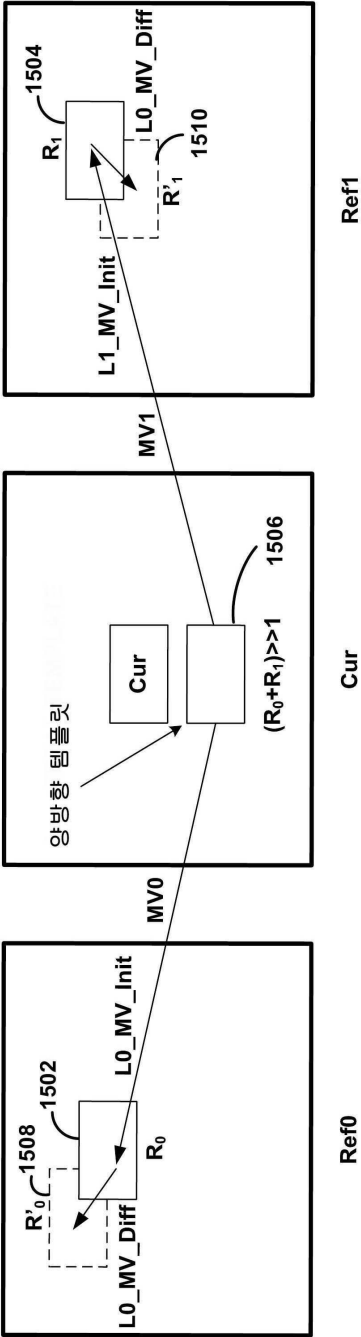
도면14c



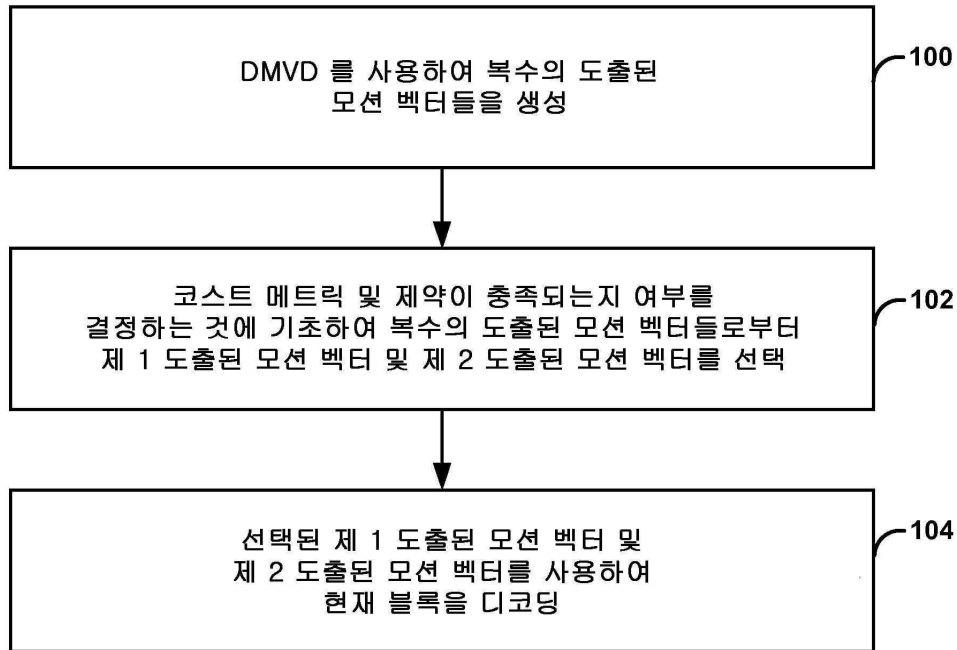
도면14d



도면15



도면16



도면17

