



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2007-0117432
(43) 공개일자 2007년12월12일

(51) Int. Cl.

H04N 7/32 (2006.01) H04N 7/24 (2006.01)

(21) 출원번호 10-2006-0132575

(22) 출원일자 2006년12월22일

심사청구일자 없음

(30) 우선권주장

1020060051601 2006년06월08일 대한민국(KR)

(71) 출원인

경희대학교 산학협력단

경기도 용인시 기흥구 서천동 1 경희대학교 수원 캠퍼스내

(72) 발명자

박광훈

경기 성남시 분당구 분당동 45번지 동아빌라 B동 302호

박민우

경기 수원시 권선구 세류2동 1167-26

서덕영

경기 성남시 분당구 수내동 푸른마을 벽산아파트 201-202

(74) 대리인

양문옥

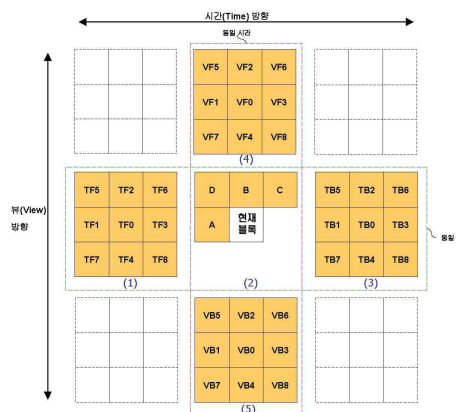
전체 청구항 수 : 총 36 항

(54) 다시점 비디오 코딩에서의 움직임 벡터 예측 방법 및 이를이용한 다시점 영상의 부호화/복호화 방법 및 장치

(57) 요약

본 발명은 다시점 비디오 코딩에서 움직임 벡터 예측값(MV_P)을 효율적으로 구함으로써 다시점 영상에 대한 부호화 효율을 향상시킬 수 있는 다시점 비디오 코딩에서의 움직임 벡터 예측 방법 및 이를 이용한 다시점 영상의 부호화/복호화 방법 및 장치에 관한 것이다. 본 발명의 일 실시예에 따른 움직임 벡터 예측 방법에서는 현재 블록의 주변 블록의 움직임 벡터(MV_N)와 함께, 현재 영상이 속한 뷰와 동일한 뷰에서 시간 방향 참조 영상의 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 현재 영상의 시간대와 동일한 시간대에서 뷰 방향 참조 영상의 뷰 방향 대응 블록의 움직임 벡터(MV_V) 중에서 선택된 하나 이상의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_P)을 구한다. 본 발명에 의하면, 현재 블록이 참조하는 영상의 특성, 즉 시간 방향 참조 영상을 참조하는지 또는 뷰 방향 참조 영상을 참조하는지를 고려하여 움직임 벡터 예측값(MV_P)을 구하기 때문에, 보다 정확한 예측을 함으로써 예측 부호화 효율을 향상시킬 수가 있다.

대표도 - 도7



특허청구의 범위

청구항 1

다시점 비디오 코딩에서 움직임 벡터 예측값(Motion Vector Predictor)을 구하는 방법에 있어서,

현재 블록의 주변 블록의 움직임 벡터들, 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터들, 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터들 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 움직임 벡터의 예측 방법.

청구항 2

제1항에 있어서,

상기 움직임 벡터 예측값은 상기 선택된 움직임 벡터의 중간값인 것을 특징으로 하는 움직임 벡터의 예측 방법.

청구항 3

제1항에 있어서,

상기 선택된 움직임 벡터는 하나이고, 상기 움직임 벡터 예측값은 상기 선택된 움직임 벡터와 같은 것을 특징으로 하는 움직임 벡터의 예측 방법.

청구항 4

제1항에 있어서,

상기 움직임 벡터 예측값은 상기 선택된 움직임 벡터와 (0, 0)의 중간값인 것을 특징으로 하는 움직임 벡터의 예측 방법.

청구항 5

다시점 비디오 코딩에서 움직임 벡터 예측값을 구하는 방법에 있어서,

현재 블록의 참조 영상이 현재 영상과 같은 시간대에서의 다른 뷰의 영상이고,

상기 현재 블록의 주변 블록의 움직임 벡터들 및 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터들 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 움직임 벡터의 예측 방법.

청구항 6

제5항에 있어서,

상기 주변 블록 및 상기 시간 방향 대응 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 뷰와 같은 뷰에 속하는 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 것을 특징으로 하는 움직임 벡터의 예측 방법.

청구항 7

제5항에 있어서,

상기 주변 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 뷰와 같은 뷰에 속하는 것이 있는 경우에는, 해당되는 상기 주변 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 것을 특징으로 하는 움직임 벡터의 예측 방법.

청구항 8

제5항에 있어서,

상기 시간 방향 대응 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 뷰와 같은 뷰에 속하는 블록이 없는 경우에는, 상기 주변 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 것을 특징

으로 하는 움직임 벡터의 예측 방법.

청구항 9

다시점 비디오 코딩에서 움직임 벡터 예측값을 구하는 방법에 있어서,

현재 블록의 참조 영상이 현재 영상과 같은 뷰에서의 다른 시간대의 영상이고,

상기 현재 블록의 주변 블록의 움직임 벡터들 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터들 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 움직임 벡터의 예측 방법.

청구항 10

제9항에 있어서,

상기 주변 블록 및 상기 뷰 방향 대응 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 시간대와 같은 시간대에 속하는 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 것을 특징으로 하는 움직임 벡터의 예측 방법.

청구항 11

제9항에 있어서,

상기 주변 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 시간대와 같은 시간대에 속하는 것이 있는 경우에는, 해당되는 상기 주변 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 것을 특징으로 하는 움직임 벡터의 예측 방법.

청구항 12

제9항에 있어서,

상기 뷰 방향 대응 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 시간대와 같은 시간대에 속하는 블록이 없는 경우에는, 상기 주변 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 것을 특징으로 하는 움직임 벡터의 예측 방법.

청구항 13

다시점 비디오 코딩에서 움직임 벡터 예측값을 구하는 방법에 있어서,

현재 블록의 주변 블록의 움직임 벡터들, 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터들, 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터들 중에서, 상기 현재 블록의 참조 영상이 현재 영상과 같은 시간대에서의 다른 뷰의 영상인지 또는 상기 현재 영상과 같은 뷰에서의 다른 시간대의 영상인지에 따라서, 적응적으로 선택된 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 움직임 벡터의 예측 방법.

청구항 14

다시점 영상의 복호화 방법에 있어서,

제1항, 제5항, 제9항, 또는 제13항의 움직임 벡터 예측 방법을 이용하여 움직임 벡터 예측값을 구하는 단계; 및
상기 움직임 벡터 예측값과 입력되는 상기 현재 블록의 움직임 벡터 차분값을 합산하여 움직임 벡터를 구하는 단계를 포함하는 다시점 영상의 복호화 방법.

청구항 15

다시점 영상의 복호화 방법에 있어서,

움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터 및 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터, 상기 현재 블록의 상기 시간 방향 대응 블록의 움직임 벡터 및 상기 현재 블록의 상기 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 단계; 및

상기 움직임 벡터 예측값과 입력되는 상기 현재 블록의 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하는 단계를 포함하는 다시점 영상의 복호화 방법.

청구항 16

제15항에 있어서,

상기 정보는 시퀀스 헤더에 포함된 정보, 슬라이스 헤더에 포함된 정보, 및/또는 블록에 대한 부호화 정보인 것을 특징으로 하는 다시점 영상의 복호화 방법.

청구항 17

제16항에 있어서,

상기 정보는 가변장 부호, 고정장 부호, 또는 Exp-Golomb 부호인 것을 특징으로 하는 다시점 영상의 복호화 방법.

청구항 18

제15항에 있어서, 상기 움직임 벡터 예측값을 구하는 단계에서는

상기 정보가 상기 시간 방향 대응 블록의 움직임 벡터 및 상기 뷰 방향 대응 블록의 움직임 벡터를 이용하지 않는다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터만을 이용하고,

상기 정보가 상기 시간 방향 대응 블록의 움직임 벡터 및 상기 뷰 방향 대응 블록의 움직임 벡터를 이용한다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터, 상기 시간 방향 대응 블록의 움직임 벡터 및 상기 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하는 것을 특징으로 하는 다시점 영상의 복호화 방법.

청구항 19

다시점 영상의 복호화 방법에 있어서,

움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하여 상기 현재 블록의 움직임 벡터 예측값을 구하는 단계; 및

상기 움직임 벡터 예측값과 입력되는 상기 현재 블록의 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하는 단계를 포함하는 다시점 영상의 복호화 방법.

청구항 20

제19항에 있어서, 상기 움직임 벡터 예측값을 구하는 단계에서는

상기 정보가 상기 시간 방향 대응 블록의 움직임 벡터를 이용하지 않는다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터만을 이용하고,

상기 정보가 상기 시간 방향 대응 블록의 움직임 벡터를 이용한다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터 및 상기 시간 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하는 것을 특징으로 하는 다시점 영상의 복호화 방법.

청구항 21

다시점 영상의 복호화 방법에 있어서,

움직임 벡터 예측값을 구할 경우에 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 현재 블록의 움직임 벡터 예측값을 구하는 단계; 및

상기 움직임 벡터 예측값과 입력되는 상기 현재 블록의 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임

임 벡터를 구하는 단계를 포함하는 다시점 영상의 복호화 방법.

청구항 22

제21항에 있어서, 상기 움직임 벡터 예측값을 구하는 단계에서는

상기 정보가 상기 뷰 방향 대응 블록의 움직임 벡터를 이용하지 않는다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터만을 이용하고,

상기 정보가 상기 뷰 방향 대응 블록의 움직임 벡터를 이용한다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터 및 상기 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하는 것을 특징으로 하는 다시점 영상의 복호화 방법.

청구항 23

다시점 영상의 부호화 방법에 있어서,

제1항, 제5항, 제9항, 또는 제13항의 움직임 벡터 예측 방법을 이용하여 움직임 벡터 예측값을 구하는 단계; 및
상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하는 단계를 포함하는 다시점 영상의 부호화 방법.

청구항 24

다시점 영상의 부호화 방법에 있어서,

움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터 및 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하는 단계;

상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터, 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 움직임 벡터 예측값을 구하는 단계; 및

상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하는 단계를 포함하는 다시점 영상의 부호화 방법.

청구항 25

다시점 영상의 부호화 방법에 있어서,

움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하는 단계;

상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 움직임 벡터 예측값을 구하는 단계; 및

상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하는 단계를 포함하는 다시점 영상의 부호화 방법.

청구항 26

다시점 영상의 부호화 방법에 있어서,

움직임 벡터 예측값을 구할 경우에 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하는 단계;

상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 움직임 벡터 예측값을 구하는 단계; 및

상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하는 단계를 포함하는 다시점 영상의 부호화 방법.

청구항 27

다시점 비디오 코딩의 신택스에 있어서,

움직임 벡터 예측값을 구하는 경우에 시간 방향 대응 블록의 움직임 벡터 및/또는 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 포함하는 다시점 비디오 코딩의 신택스.

청구항 28

제27항에 있어서,

상기 신택스는 시퀀스에 대한 신택스, 슬라이스에 대한 신택스, 및/또는 매크로블록에 대한 신택스인 것을 특징으로 하는 다시점 비디오 코딩의 신택스.

청구항 29

다시점 영상의 복호화 장치에 있어서,

제1항, 제5항, 제9항, 또는 제13항의 움직임 벡터 예측 방법을 이용하여 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부; 및

상기 움직임 벡터 예측값과 입력되는 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 생성하기 위한 가산기를 포함하는 다시점 영상의 복호화 장치.

청구항 30

다시점 영상의 복호화 장치에 있어서,

움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터 및 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 입력 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터, 상기 현재 블록의 상기 시간 방향 대응 블록의 움직임 벡터 및 상기 현재 블록의 상기 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부; 및

상기 움직임 벡터 예측값과 입력되는 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하기 위한 가산기를 포함하는 다시점 영상의 복호화 장치.

청구항 31

다시점 영상의 복호화 장치에 있어서,

움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 입력 정보에 기초하여, 상기 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 상기 시간 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부; 및

상기 움직임 벡터 예측값과 입력되는 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하기 위한 가산기를 포함하는 다시점 영상의 복호화 장치.

청구항 32

다시점 영상의 복호화 장치에 있어서,

움직임 벡터 예측값을 구할 경우에 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 입력 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 상기 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부; 및

상기 움직임 벡터 예측값과 입력되는 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하기 위한 가산기를 포함하는 다시점 영상의 복호화 장치.

청구항 33

다시점 영상의 부호화 장치에 있어서,

제1항, 제5항, 제9항, 또는 제13항의 움직임 벡터 예측 방법을 이용하여 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부;

상기 시간 방향 대응 블록을 포함하는 시간 방향 참조 영상 및 상기 뷰 방향 대응 블록을 포함하는 뷰 방향 참조 영상을 이용하여 상기 현재 블록의 움직임 벡터를 구하기 위한 움직임 예측부; 및

상기 움직임 벡터와 상기 움직임 벡터 예측값을 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하기 위한 가산기를 포함하는 다시점 영상의 부호화 장치.

청구항 34

다시점 영상의 부호화 장치에 있어서,

움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터 및 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하기 위한 이용 여부 판단부;

상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터, 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부; 및

상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하기 위한 가산기를 포함하는 다시점 영상의 부호화 장치.

청구항 35

다시점 영상의 부호화 장치에 있어서,

움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하기 위한 이용 여부 판단부;

상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부; 및

상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하기 위한 가산기를 포함하는 다시점 영상의 부호화 장치.

청구항 36

다시점 영상의 부호화 장치에 있어서,

움직임 벡터 예측값을 구할 경우에 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하기 위한 이용 여부 판단부;

상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부; 및

상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하기 위한 가산기를 포함하는 다시점 영상의 부호화 장치.

명 세 서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 다시점 비디오 코딩(Multi-view Video Coding, MVC)에 관한 것으로, 보다 구체적으로 다시점 영상의

부호화/복호화에 있어서 움직임 벡터(Motion Vector, MV)를 예측하는 방법과 이를 이용하는 다시점 영상의 부호화/복호화 방법 및 장치에 관한 것이다.

<20> 다시점 비디오 코딩(MVC)은 복수의 카메라로부터 획득한 각 뷰의 영상 시퀀스들을 처리하는 것이다. 이들 복수의 카메라들은 동일한 객체(object)에 대하여 거리 및/또는 대향 방향에 있어서 다소 차이가 있도록 배치된다. 이러한 복수의 카메라들을 통해서 획득한 다시점 영상은, 동일한 객체의 표면에서 빛이 반사되는 정도가 그 방향에 따라서 다소 차이가 있는 반면에 각 뷰의 영상이 상관관계가 높다는 특성을 함께 갖는다. 다시점 영상이 갖는 전자의 특성으로 인하여 동일한 객체에 대한 각 뷰의 영상 사이에도 밝기와 색차의 차이가 존재하므로, 다시점 비디오 코딩에서 부호화 효율을 향상시키기 위해서는 다시점 영상이 갖는 전자의 특성을 고려하여야 한다. 아울러 각 뷰의 영상이 갖는 높은 상관관계를 갖는다는 후자의 특성을 이용하면 다시점 영상의 부호화 효율을 향상시킬 수가 있다.

<21> 현재 표준화가 활발하게 진행 중인 다시점 비디오 코딩에서는, 기존의 동영상 코딩에 관한 국제 표준인 H.264/MPEG-4 part 10 Advanced Video Coding (이하, H.264/AVC라 한다)을 기반으로 하면서, 전술한 다시점 영상의 2가지 특성을 고려하여 부호화 효율을 향상시키는 방법을 모색하고 있다. 예를 들어, H.264/AVC에 규정되어 있는 조인트 스케일러블 비디오 코딩(Joint Scalable Video Coding, JSVC)에서 시간적 스케일러빌리티(Temporal scalability)를 지원하기 위해 수행되는 방법인 계층적인 B-영상(hierarchical B-pictures) 코딩 처리를 다시점 비디오 코딩에서의 뷰내(intra-view) 예측 코딩에도 적용하고 있다. 그리고 다시점 비디오 코딩에서는 뷰간(inter-view) 예측 처리도 병행해서 수행하여 부호화 효율의 향상을 이루고자 한다.

<22> 도 1은 현재 표준화가 진행 중인 다시점 비디오 코딩에서의 뷰내 예측 및 뷰간 예측 관계를 보여주는 예측 구조의 일례를 도시한 도면이다. 도 1은 8개의 뷰가 존재할 경우의 예로써, 시간 방향의 영상 그룹(Group of Pictures, GOP)의 크기가 8인 경우이다. 도 1에서 S0, S1, S2, S3, ..., S7은 각각 하나의 뷰를 나타내며, T0, T1, T2, T3, ..., T100은 시간 방향을 나타낸다.

<23> 도 1을 참조하면, 각 뷰 내에서는 H.264/AVC에 규정되어 있는 계층적인 B-영상 구조가 사용되어 시간 방향으로의 예측 부호화를 수행하고 있는 것을 알 수 있다. 그리고 각 뷰의 첫번째 시간(T0)의 영상들과 그 이후로 시간 방향으로 8 프레임씩 떨어져 있는 영상 즉, 시간 방향의 영상 그룹(GOP)의 크기만큼 떨어져 있는 영상들(T8, T16, T24, ... 에서의 영상)은 이웃하는 뷰로부터의 예측(뷰간 예측, inter-view prediction)만을 수행하고 있다. 예컨대, 시간 T0, T8, T16, T24, ... 에서, S2 뷰는 S0 뷰로부터, S1 뷰는 S0 뷰와 S2 뷰로부터, S4 뷰는 S2 뷰로부터, S3 뷰는 S2 뷰와 S4 뷰로부터, S6 뷰는 S4 뷰로부터, S5 뷰는 S4 뷰와 S6 뷰로부터, S7 뷰는 마지막 뷰이기 때문에 S6 뷰로부터 예측을 수행한다. 또한, 매 두번째 뷰(즉, S1, S3, S5, S7)에서는 그 이외의 시간(T1, T2, T3, ..., T7, T9, T10, ...)에서도 시간 방향의 뷰내 예측과 함께 뷰간 예측도 수행한다. 즉, 시간 방향의 뷰내 예측 외에도 S1 뷰는 S0 뷰와 S2 뷰로부터, S3 뷰는 S2 뷰와 S4 뷰로부터, S5 뷰는 S4 뷰와 S6 뷰로부터 뷰간 예측을 수행한다.

<24> 현재 표준화가 진행 중인 다시점 비디오 코딩에서는 움직임 정보(motion information), 예컨대 움직임 벡터(MV)의 예측 방법으로서 H.264/AVC에 규정되어 있는 것과 같은 방법을 사용할 것을 제안하고 있다. H.264/AVC에 규정되어 있는 움직임 벡터(MV)의 예측 방법은, 시간 방향으로의 움직임 예측(Motion Estimation, ME)을 통해 구해지는 현재 블록(current block)의 움직임 벡터(MV_C, Motion Vector of Current block)는 그 주변 블록의 움직임 벡터와 유사하다는 가정에 기초하고 있다.

<25> 보다 구체적으로, H.264/AVC에서는 상기 주변 블록의 움직임 벡터(MV_N, Motion Vector of Neighboring block)들을 이용하여 움직임 벡터 예측값(MV_P, Motion Vector Predictor)을 구한 다음, 현재 블록의 움직임 벡터(MV_C)에서 움직임 벡터 예측값(MV_P)을 뺀 움직임 벡터 차분값(MV_D, Motion Vector Difference)을 구한다. 그리고 움직임 예측을 통해 구한 현재 블록의 움직임 벡터(MV_C)가 아닌 상기 움직임 벡터 차분값(MV_D)을 복호화기(decoder)에 전송한다. 부호화기에서 복호화기로 전송되는 움직임 벡터 차분값(MV_D)을 구하는 식은 아래의 수학식 1과 같이 표현할 수 있다.

수학식 1

$$MV_D = MV_C - MV_P$$

<26>

<27> 그리고 복호화기에서는 부호화기에서와 동일한 방법으로 움직임 벡터 예측값(MV_p)을 구하고 부호화기로부터 전송되는 움직임 벡터 차분값(MV_D)을 움직임 벡터 예측값(MV_p)에 더하여, 현재 블록의 움직임 벡터(MV_C)를 구한다. 복호화기에서 현재 블록의 움직임 벡터(MV)를 구하는 식은 다음의 수학적 식 2와 같이 표현할 수 있다.

수학적 식 2

$$MV_C = MV_p + MV_D$$

<28>

<29> 그리고 H.264/AVC에는 현재 블록의 주변 블록이 여러 개가 존재하면, 이 주변 블록의 움직임 벡터(MV_N)들의 전부 또는 일부의 중간값(Median)을 구하여 이를 현재 블록의 움직임 벡터 예측값(MV_p)으로 사용하는 것으로 규정되어 있다. 또한, H.264/AVC에는 주변 블록의 크기가 매크로블록(macroblock)인 16×16 픽셀인 경우만이 아니라, 16×8 픽셀, 8×16 픽셀, 8×8 픽셀, 8×4 픽셀, 4×8 픽셀, 또는 4×4 픽셀 등으로 가변적인 경우와 현재 블록의 크기가 16×8 픽셀이거나 8×16 픽셀인 경우에 대해서도 각각 규정되어 있다. 이러한 경우 각각의 경우에 움직임 벡터 예측값(MV_p)을 구하는 구체적인 방법은 상당히 복잡하고 다양하다. 이하에서는 이에 대하여 간략히 설명한다.

<30> 도 2에는 H.264/AVC에서 주변 블록의 움직임 벡터(MV_N)를 이용하여 움직임 벡터 예측값(MV_p)을 구하는 방법을 설명하기 위한 도면이다. 도 2를 참조하면, 현재 블록의 움직임 벡터 예측값(MV_p)을 구하기 위해 이용하는 주변 블록은 원칙적으로 블록 'A', 'B', 및 'C'가 된다. 이 경우에, 움직임 벡터 예측값(MV_p)은, 수학적 식 3과 같이, 블록 'A', 'B', 및 'C'의 움직임 벡터(MV)들의 중간값이 된다.

수학적 식 3

$$MV_p = Median(MV(A), MV(B), MV(C))$$

<31>

<32> 여기서, $MV(A)$ 는 블록 'A'의 움직임 벡터(MV), $MV(B)$ 는 블록 'B'의 움직임 벡터(MV), $MV(C)$ 는 블록 'C'의 움직임 벡터(MV), MV_p 는 현재 블록의 움직임 벡터 예측값이고, $Median(x, y, z)$ 은 x , y , 및 z 의 중간값을 나타낸다.

<33> 그리고 중간값(Median)을 구하는 방법을 수학적 함수로 나타내면 수학적 식 4와 같다.

수학적 식 4

$$Median(x, y, z) = x + y + z - Min(x, Min(y, z)) - Max(x, Max(y, z))$$

<34>

<35> 여기서, $Min(x, y)$ 는 x , y 중에서 작거나 같은 값을 나타내고, $Max(x, y)$ 는 x , y 중에서 크거나 같은 값을 나타낸다.

<36> 도 3에는 H.264/AVC에서 주변 블록의 움직임 벡터(MV_N)를 이용하여 움직임 벡터 예측값(MV_p)을 구하는 다른 방법을 설명하기 위한 도면으로서, 매크로블록 크기의 주변 블록이 여러 가지 모양의 파티션 블록으로 분할되어 있는 경우이다. 도 3을 참조하면, 현재 블록의 왼쪽으로 인접하는 주변 블록의 경우에는 그 주변 블록 내의 최우상측에 있는 파티션 블록을 블록 'A'로 선택하고, 현재 블록의 위쪽으로 인접하는 주변 블록의 경우에는 그 주변 블록 내의 최좌하측에 있는 파티션 블록을 블록 'B'로 선택하고, 현재 블록의 우상측으로 인접하는 주변 블록의 경우에는 그 주변 블록 내의 최좌하측에 있는 파티션 블록을 블록 'C'로 선택하여, 선택된 파티션 블록 'A', 'B', 및 'C'의 움직임 벡터들에 대한 중간값 연산으로 구해진 값을 움직임 벡터 예측값(MV_p)으로 한다.

<37> 도 4에는 H.264/AVC에서 주변 블록의 움직임 벡터(MV_N)를 이용하여 움직임 벡터 예측값(MV_p)을 구하는 또 다른 방법을 설명하기 위한 도면으로서, 도 2와 도 3을 참조하여 전술한 예에서, 블록 'C'의 움직임 벡터를 이용할 수 없을 경우이다. 도 4를 참조하면, 블록 'C'의 움직임 벡터를 이용할 수 없는 경우에는 블록 'C' 대신에 블록 'D'의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구한다. 즉, 도 4의 예에서는 블록 'A', 'B', 및 'D'의 움직임 벡터(MV)들에 대한 중간값 연산으로 구해진 값이 움직임 벡터 예측값(MV_p)이 된다.

- <38> 그리고 앞에서 설명한 각각의 주변 블록(블록 'A', 'B', 'C', 및/또는 'D')의 움직임 벡터(MV_N)를 이용할 수 없는 경우이거나, 상기 주변 블록이 인트라 예측 모드의 블록이거나, 또는 시간 방향의 예측을 사용하지 않는 블록 중인 경우라면, 그 주변 블록의 움직임 벡터(MV)는 (0.0, 0.0)으로 설정하고 참조 영상 번호는 '-1'로 설정하여 움직임 벡터 예측값(MV_P)을 구한다.
- <39> 또한, 도 5a에 도시되어 있는 것과 같이, 현재 블록이 8×16 픽셀 모드인 경우에, 좌측에 위치한 8×16 블록의 움직임 벡터 예측값(MV_P)으로는 블록 'A'의 움직임 벡터(MV)만 이용하여 구하고, 우측에 위치한 8×16 블록의 움직임 벡터 예측값(MV_P)으로는 블록 'C'의 움직임 벡터(MV)만을 이용하여 구한다. 그리고 도 5b에 도시되어 있는 것과 같이, 현재 블록이 16×8 픽셀 모드인 경우에, 상측에 위치한 16×8 픽셀의 크기를 갖는 블록의 움직임 벡터 예측값(MV_P)으로는 블록 'B'의 움직임 벡터(MV)를 사용하고, 매크로블록의 하측에 위치한 16×8 픽셀의 크기를 갖는 블록의 움직임 벡터 예측값(MV_P)으로는 블록 'A'의 움직임 벡터(MV)를 사용한다.
- <40> 또한, 전술한 예에서, 현재 블록이 현재 영상 또는 슬라이스의 최상측에 위치하는 경우에는 블록 'B' 및 'C'는 영상 또는 슬라이스의 경계 밖에 위치한다. 이 경우에는 블록 'B'와 'C'는 이용할 수 없기 때문에, 블록 'A'의 움직임 벡터(MV)를 현재 블록의 움직임 벡터 예측값(MV_P)으로 한다.
- <41> 그리고 전술한 예에서, 블록 'A', 'B', 및 'C'의 움직임 벡터를 모두 이용할 수 있는 경우라도, 블록 'A', 'B', 및 'C'의 움직임 벡터들 중에서 현재 블록이 참조하는 영상과 같은 영상을 참조하는 블록이 하나만 있을 경우에는, 그 블록의 움직임 벡터를 움직임 벡터 예측값(MV_P)으로 한다.
- <42> 또한, H.264/AVC에서는 매크로블록 정보를 전혀 보내지 않고 움직임 보상(Motion Compensation, MC)을 수행하는 스킵 매크로블록 모드(Skip Macroblock Mode)가 존재하는데, 여기서도 움직임 벡터 예측값(MV_P)을 구하여 부호화를 수행한다. 그리고 H.264/AVC에서는 공간 직접 모드(Spatial Direct Mode)가 존재하는데, 여기서도 주변 블록의 움직임 벡터 예측값(MV_P)을 이용하여 현재 블록의 움직임 벡터 예측값(MV_P)을 구한다.

발명이 이루고자 하는 기술적 과제

- <43> 그런데 다시점 비디오 코딩에서는 시간 방향의 예측과 함께 뷰 방향의 예측도 수행하기 때문에, 움직임 예측의 방향에 따라서 2가지 종류의 움직임 벡터를 구분할 수 있다. 즉, 같은 뷰 내에서 시간 방향으로 움직임 예측을 수행하여 다른 시간대의 영상을 참조하여 구하는 경우의 움직임 벡터와 같은 시간에서 뷰 방향으로 움직임 예측을 수행하여 다른 뷰의 영상을 참조하여 구하는 경우의 움직임 벡터가 있다. 후자의 경우와 같이, 뷰 방향으로 움직임 예측을 수행하여 구해진 움직임 벡터를 디스패리티 벡터(Disparity Vector, DV)라고도 한다. 하나의 영상 내에서 이 두 가지 종류의 움직임 벡터 중에서 어느 하나만 존재하거나 또는 두 가지 종류의 움직임 벡터가 동시에 존재할 수도 있다.
- <44> 현재 국제 표준화가 진행 중인 다시점 비디오 코딩에서는 하나의 영상(픽처 또는 슬라이스)에 두 가지 종류의 움직임 벡터가 모두 포함되어 있는 경우, 예컨대 현재 블록의 움직임 벡터(MV_C)와 주변 블록의 움직임 벡터(MV_N)가 참조하는 영상의 방향이 서로 다른 경우에도 H.264/AVC에서 규정된 것과 같은 방법으로 움직임 벡터 예측값(MV_P)을 구한다. 즉, 움직임 벡터가 참조하는 영상의 방향에 관계없이 주변 블록들의 움직임 벡터(MV_N)들만을 이용하여 움직임 벡터 예측값(MV_P)을 구한다. 그런데, H.264/AVC에 규정된 것과 같은 방법을 일률적으로 적용하여 다시점 비디오 코딩에서의 움직임 벡터 예측 방법에 적용할 경우에는 다음과 같은 문제가 발생할 수 있다.
- <45> 예측을 수행하는데 있어서, 만약 주변 블록의 움직임 벡터(MV_N)가 시간 방향으로 예측을 수행하여 구해진 것이고 (즉, 주변 블록의 참조 영상이 같은 뷰 내에서 다른 시간의 영상인 경우), 현재 블록의 움직임 벡터(MV_C)가 뷰 방향으로 예측을 수행하여 구해진 것인 경우(즉, 현재 블록의 참조 영상이 같은 시간대에서 다른 뷰의 영상인 경우)에, 주변 블록의 움직임 벡터(MV_N)와 현재 블록의 움직임 벡터(MV_C)는 서로 특성이 다르게 된다. 이와 같은 경우에, H.264/AVC에 규정된 방법을 그대로 적용하여, 주변 블록의 움직임 벡터(MV_N)들만을 이용하여 움직임 벡터 예측값(MV_P)을 구하면, 현재 블록의 움직임 벡터(MV_C)에 대한 정확한 예측이 이루어지지 않는다. 그 결과, 상기 움직임 벡터 예측값(MV_P)과 현재 블록의 움직임 벡터(MV_N)를 차분하여 구해지는 움직임 벡터 차분값(MV_D)에

대한 정보량(엔트로피)이 증가하여 예측 부호화의 효율이 떨어지게 된다.

<46> 마찬가지로, 주변 블록의 움직임 벡터(MV_N)가 뷰 방향으로 예측을 수행하여 구해진 것이고, 현재 블록의 움직임 벡터(MV_C)가 시간 방향으로 예측을 수행하여 구해진 것인 경우에도, 주변 블록의 움직임 벡터(MV_N)와 현재 블록의 움직임 벡터(MV_C)는 서로 특성이 다르게 된다. 이와 같은 경우에도 H.264/AVC에 규정된 방법을 그대로 적용하면, 움직임 벡터에 대한 정확한 예측이 이루어지지 않을 수가 있다.

<47> 따라서 본 발명에서는 현재 블록의 움직임 벡터(MV_C)와 주변 블록의 움직임 벡터(MV_N)의 특성(뷰 방향으로 움직임 예측을 수행하는지 또는 시간 방향으로 움직임 예측을 수행하는지 여부)이 같은 경우는 물론이고, 상기 특성이 다른 경우에도 움직임 벡터에 대한 예측 부호화의 효율을 향상시킬 수 있는 다시점 비디오 코딩에서의 움직임 벡터의 예측 방법과 이를 이용한 다시점 영상 부호화/복호화 방법 및 장치를 제안하고자 한다.

발명의 구성 및 작용

<48> 상기한 기술적 과제를 달성하기 위한 본 발명의 일 실시예에 따른 다시점 비디오 코딩에서 움직임 벡터 예측값(Motion Vector Predictor)을 구하는 방법은 현재 블록의 주변 블록의 움직임 벡터들, 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터들, 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터들 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구한다.

<49> 상기 실시예의 일 측면에 의하면, 상기 움직임 벡터 예측값은 상기 선택된 움직임 벡터의 중간값이거나, 상기 선택된 움직임 벡터와 (0, 0)의 중간값일 수 있다. 또는, 상기 선택된 움직임 벡터는 하나이고, 상기 움직임 벡터 예측값은 상기 선택된 움직임 벡터와 같을 수 있다.

<50> 상기한 기술적 과제를 달성하기 위한 본 발명의 다른 실시예에 따른 다시점 비디오 코딩에서 움직임 벡터 예측값을 구하는 방법은 현재 블록의 참조 영상이 현재 영상과 같은 시간대에서의 다른 뷰의 영상이고, 상기 현재 블록의 주변 블록의 움직임 벡터들 및 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터들 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구한다.

<51> 상기 실시예의 일 측면에 의하면, 상기 주변 블록 및 상기 시간 방향 대응 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 뷰와 같은 뷰에 속하는 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구할 수 있다.

<52> 상기 실시예의 다른 측면에 의하면, 상기 주변 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 뷰와 같은 뷰에 속하는 것이 있는 경우에는, 해당되는 상기 주변 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구할 수 있다.

<53> 상기 실시예의 또 다른 측면에 의하면, 상기 시간 방향 대응 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 뷰와 같은 뷰에 속하는 블록이 없는 경우에는, 상기 주변 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구할 수 있다.

<54> 상기한 기술적 과제를 달성하기 위한 본 발명의 또 다른 실시예에 따른 다시점 비디오 코딩에서 움직임 벡터 예측값을 구하는 방법은 현재 블록의 참조 영상이 현재 영상과 같은 뷰에서의 다른 시간대의 영상이고, 상기 현재 블록의 주변 블록의 움직임 벡터들 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터들 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구한다.

<55> 상기 실시예의 일 측면에 의하면, 상기 주변 블록 및 상기 뷰 방향 대응 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 시간대와 같은 시간대에 속하는 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구할 수 있다.

<56> 상기 실시예의 다른 측면에 의하면, 상기 주변 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 시간대와 같은 시간대에 속하는 것이 있는 경우에는, 해당되는 상기 주변 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구할 수 있다.

<57> 상기 실시예의 또 다른 측면에 의하면, 상기 뷰 방향 대응 블록 중에서 그 참조 영상이 상기 현재 블록의 참조 영상이 속하는 시간대와 같은 시간대에 속하는 블록이 없는 경우에는, 상기 주변 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구할 수 있다.

- <58> 상기한 기술적 과제를 달성하기 위한 본 발명의 또 다른 실시예에 따른 다시점 비디오 코딩에서 움직임 벡터 예측값을 구하는 방법은 현재 블록의 주변 블록의 움직임 벡터들, 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터들, 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터들 중에서, 상기 현재 블록의 참조 영상이 현재 영상과 같은 시간대에서의 다른 뷰의 영상인지 또는 상기 현재 영상과 같은 뷰에서의 다른 시간대의 영상인지에 따라서, 적응적으로 선택된 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구한다.
- <59> 상기한 기술적 과제를 달성하기 위한 본 발명의 일 실시예에 따른 다시점 영상의 복호화 방법은 전술한 본 발명의 일 실시예에 따른 움직임 벡터 예측 방법을 이용하여 움직임 벡터 예측값을 구하는 단계, 및 상기 움직임 벡터 예측값과 입력되는 상기 현재 블록의 움직임 벡터 차분값을 합산하여 움직임 벡터를 구하는 단계를 포함한다.
- <60> 상기한 기술적 과제를 달성하기 위한 본 발명의 다른 실시예에 따른 다시점 영상의 복호화 방법은 움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터 및 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터, 상기 현재 블록의 상기 시간 방향 대응 블록의 움직임 벡터 및 상기 현재 블록의 상기 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하는 단계, 및 상기 움직임 벡터 예측값과 입력되는 상기 현재 블록의 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하는 단계를 포함한다.
- <61> 상기 실시예의 일 측면에 의하면, 상기 정보는 시퀀스 헤더에 포함된 정보, 슬라이스 헤더에 포함된 정보, 및/또는 블록에 대한 부호화 정보일 수 있다. 이 경우에 상기 정보는 가변장 부호, 고정장 부호, 또는 Exp-Golomb 부호일 수 있다.
- <62> 상기 실시예의 다른 측면에 의하면, 상기 움직임 벡터 예측값을 구하는 단계에서는 상기 정보가 상기 시간 방향 대응 블록의 움직임 벡터 및 상기 뷰 방향 대응 블록의 움직임 벡터를 이용하지 않는다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터만을 이용하고, 상기 정보가 상기 시간 방향 대응 블록의 움직임 벡터 및 상기 뷰 방향 대응 블록의 움직임 벡터를 이용한다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터, 상기 시간 방향 대응 블록의 움직임 벡터 및 상기 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용할 수 있다.
- <63> 상기한 기술적 과제를 달성하기 위한 본 발명의 또 다른 실시예에 따른 다시점 영상의 복호화 방법은 움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하여 상기 현재 블록의 움직임 벡터 예측값을 구하는 단계, 및 상기 움직임 벡터 예측값과 입력되는 상기 현재 블록의 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하는 단계를 포함한다.
- <64> 상기 실시예의 일 측면에 의하면, 상기 움직임 벡터 예측값을 구하는 단계에서는 상기 정보가 상기 시간 방향 대응 블록의 움직임 벡터를 이용하지 않는다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터만을 이용하고, 상기 정보가 상기 시간 방향 대응 블록의 움직임 벡터를 이용한다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터 및 상기 시간 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용할 수 있다.
- <65> 상기한 기술적 과제를 달성하기 위한 본 발명의 다른 실시예에 따른 다시점 영상의 복호화 방법은 움직임 벡터 예측값을 구할 경우에 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 현재 블록의 움직임 벡터 예측값을 구하는 단계, 및 상기 움직임 벡터 예측값과 입력되는 상기 현재 블록의 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하는 단계를 포함한다.
- <66> 상기 실시예의 일 측면에 의하면, 상기 움직임 벡터 예측값을 구하는 단계에서는 상기 정보가 상기 뷰 방향 대응 블록의 움직임 벡터를 이용하지 않는다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터만을 이용하고, 상기 정보가 상기 뷰 방향 대응 블록의 움직임 벡터를 이용한다는 것을 나타내는 정보인 경우에는 상기 주변 블록의 움직임 벡터 및 상기 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용할 수 있다.

- <67> 상기한 기술적 과제를 달성하기 위한 본 발명의 다른 실시예에 따른 다시점 영상의 부호화 방법은 전술한 실시예에 따른 움직임 벡터 예측 방법을 이용하여 움직임 벡터 예측값을 구하는 단계, 및 상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하는 단계를 포함한다.
- <68> 상기한 기술적 과제를 달성하기 위한 본 발명의 다른 실시예에 따른 다시점 영상의 부호화 방법은 움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터 및 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하는 단계, 상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터, 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 움직임 벡터 예측값을 구하는 단계, 및 상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하는 단계를 포함한다.
- <69> 상기한 기술적 과제를 달성하기 위한 본 발명의 또 다른 실시예에 따른 다시점 영상의 부호화 방법은 움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하는 단계, 상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 움직임 벡터 예측값을 구하는 단계, 및 상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하는 단계를 포함한다.
- <70> 상기한 기술적 과제를 달성하기 위한 본 발명의 또 다른 실시예에 따른 다시점 영상의 부호화 방법은 움직임 벡터 예측값을 구할 경우에 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하는 단계, 상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 움직임 벡터 예측값을 구하는 단계, 및 상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하는 단계를 포함한다.
- <71> 상기한 기술적 과제를 달성하기 위한 본 발명의 일 실시예에 따른 다시점 비디오 코딩의 신택스는 움직임 벡터 예측값을 구하는 경우에 시간 방향 대응 블록의 움직임 벡터 및/또는 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 포함한다.
- <72> 상기 실시예의 일 측면에 의하면, 상기 신택스는 시퀀스에 대한 신택스, 슬라이스에 대한 신택스, 및/또는 매크로블록에 대한 신택스일 수 있다.
- <73> 상기한 기술적 과제를 달성하기 위한 본 발명의 일 실시예에 따른 다시점 영상의 복호화 장치는 전술한 본 발명의 실시예에 따_{fms} 움직임 벡터 예측 방법을 이용하여 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부, 및 상기 움직임 벡터 예측값과 입력되는 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 생성하기 위한 가산기를 포함한다.
- <74> 상기한 기술적 과제를 달성하기 위한 본 발명의 다른 실시예에 따른 다시점 영상의 복호화 장치는 움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터 및 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 입력 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터, 상기 현재 블록의 상기 시간 방향 대응 블록의 움직임 벡터 및 상기 현재 블록의 상기 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부, 및 상기 움직임 벡터 예측값과 입력되는 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하기 위한 가산기를 포함한다.
- <75> 상기한 기술적 과제를 달성하기 위한 본 발명의 또 다른 실시예에 따른 다시점 영상의 복호화 장치는 움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 입력 정보에 기초하여, 상기 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 상기 시간 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부, 및 상기 움직임 벡터 예측값과 입력되는 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하기 위한 가산기를 포함한다.
- <76> 상기한 기술적 과제를 달성하기 위한 본 발명의 또 다른 실시예에 따른 다시점 영상의 복호화 장치는 움직임 벡터 예측값을 구할 경우에 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 입력 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 상기 뷰 방향 대응 블록의 움직임 벡터 중에서 적응

적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부, 및 상기 움직임 벡터 예측값과 입력되는 움직임 벡터 차분값을 합산하여 상기 현재 블록의 움직임 벡터를 구하기 위한 가산기를 포함한다.

<77> 상기한 기술적 과제를 달성하기 위한 본 발명의 일 실시예에 따른 다시점 영상의 부호화 장치는 전술한 본 발명의 실시예에 따른 움직임 벡터 예측 방법을 이용하여 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부, 상기 시간 방향 대응 블록을 포함하는 시간 방향 참조 영상 및 상기 뷰 방향 대응 블록을 포함하는 뷰 방향 참조 영상을 이용하여 상기 현재 블록의 움직임 벡터를 구하기 위한 움직임 예측부, 및 상기 움직임 벡터와 상기 움직임 벡터 예측값을 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하기 위한 가산기를 포함한다.

<78> 상기한 기술적 과제를 달성하기 위한 본 발명의 다른 실시예에 따른 다시점 영상의 부호화 장치는 움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터 및 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하기 위한 이용 여부 판단부, 상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터, 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 블록의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부, 및 상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하기 위한 가산기를 포함한다.

<79> 상기한 기술적 과제를 달성하기 위한 본 발명의 또 다른 실시예에 따른 다시점 영상의 부호화 장치는 움직임 벡터 예측값을 구할 경우에 시간 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하기 위한 이용 여부 판단부, 상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 시간 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부, 및 상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하기 위한 가산기를 포함한다.

<80> 상기한 기술적 과제를 달성하기 위한 본 발명의 또 다른 실시예에 따른 다시점 영상의 부호화 장치는 움직임 벡터 예측값을 구할 경우에 뷰 방향 대응 블록의 움직임 벡터를 이용할지 여부를 나타내는 정보를 생성하기 위한 이용 여부 판단부, 상기 정보에 기초하여, 현재 블록의 주변 블록의 움직임 벡터 및 상기 현재 블록의 뷰 방향 대응 블록의 움직임 벡터 중에서 적응적으로 선택된 하나 이상의 움직임 벡터를 이용하여 상기 움직임 벡터 예측값을 구하기 위한 움직임 벡터 예측부, 및 상기 움직임 벡터 예측값과 상기 현재 블록의 움직임 벡터를 감산하여 상기 현재 블록의 움직임 벡터 차분값을 구하기 위한 가산기를 포함한다.

<81> 이하에서는, 첨부 도면을 참조하여 본 발명의 바람직한 실시예에 대하여 상세하게 설명한다. 후술하는 실시예는 본 발명의 기술적 사상을 설명하기 위한 목적이므로, 본 발명의 기술적 사상은 이 실시예에 의하여 한정되는 것으로 해석되어서는 안된다. 그리고 이하의 실시예의 설명에서 각각의 구성요소의 명칭은 당업계에서 다른 명칭으로 호칭될 수 있는데, 이들이 기능적 유사성과 동일성이 있다면 비록 다른 명칭을 사용하더라도 본 발명의 실시예와 균등한 구성이라고 볼 수 있다. 마찬가지로 도면상의 구성을 일부 변형한 실시예가 채용되더라도 기능적 유사성과 동일성이 있다면 양자는 균등한 구성으로 볼 수 있다. 본 실시예에 대한 설명 및 도면에서 각각의 구성요소에 부가된 참조 부호는 단지 설명의 편의를 위하여 기재된 것일 뿐이다.

<82> 본 발명에 따른 다시점 비디오 코딩에서는 현재 블록의 움직임 벡터 예측값(MV_p)을 구할 때, 다음 세 가지 움직임 벡터(MV) 중에서 한 가지 이상의 움직임 벡터(MV)를 이용한다.

<83> (1) 현재 블록의 주변 블록의 움직임 벡터(MV_x).

<84> (2) 시간 방향 대응 블록의 움직임 벡터(MV_T), 즉 현재 영상과 같은 뷰 내에서 다른 시간에서의 영상(이하, '시간 방향 참조 영상')에서 현재 블록의 위치와 동일한 위치의 블록과 그 주변 블록의 움직임 벡터.

<85> (3) 뷰 방향 대응 블록의 움직임 벡터(MV_V), 즉, 현재 영상과 같은 시간에서의 다른 뷰의 영상(이하, '뷰 방향 참조 영상')에서 현재 블록의 위치와 동일한 위치의 블록과 그 주변 블록의 움직임 벡터(MV).

<86> 인터 영상(inter-picture)으로 부호화될 영상의 경우에는, 정확한 움직임 벡터 예측값(MV_p)을 구할 수 있다면 예측 부호화 효율을 향상시킬 수 있다. 즉, 정확한 움직임 벡터 예측값(MV_p)을 구하면, 복호기로 전송되는 움직임 벡터 차분값(MV_d)의 정보량을 그만큼 줄일 수 있기 때문에 영상의 압축율을 높일 수 있다. 그런데 전술한 바와 같이, 다시점 비디오 코딩에서는 움직임 벡터가 그 특성에 따라서 시간 방향으로 예측을 수행하여 구해진

경우와 뷰 방향으로 예측을 수행하여 구해진 경우의 두 가지가 존재한다. 이것은 한편으로는 현재 블록에 대한 더 많은 정보를 이용하여 예측 부호화에 사용할 수 있다는 것을 의미한다. 예측 부호화에 많은 정보를 이용한다면, 보다 정확한 움직임 벡터 예측값(MV_p)을 구할 수 있으며, 움직임 벡터 차분값(MV_d)의 정보량을 그 만큼 감소시킬 수 있다.

<87> 본 발명의 따른 다시점 비디오 코딩에서의 움직임 벡터 예측 방법에서는 전술한 세 가지 종류의 움직임 벡터(MV_N , MV_T , MV_V)들 중에서 일부를 선택하여 움직임 벡터 예측값(MV_p)을 구하며, 바람직하게는 현재 블록의 움직임 벡터(MV_C)의 특성과 같은 특성을 갖는 움직임 벡터를 선택한다. 이와 같이, 움직임 벡터 예측값(MV_p)을 구하기 위하여 이용할 수 있는 블록이 주변 블록에만 한정되지 않기 때문에, 현재 블록의 움직임 벡터(MV_C)와 그 주변 블록의 움직임 벡터(MV_N)의 특성이 같은 경우는 물론이고, 상기 특성이 다른 경우에도 보다 정확한 움직임 벡터 예측값(MV_p)을 구할 수 있다. 그리고 보다 정확한 움직임 벡터 예측값(MV_p)을 구함으로써, 움직임 벡터 차분값(MV_d)의 정보량을 줄이고 부호화 효율을 향상시킬 수 있다. 이하에서는, 이에 대하여 보다 구체적인 예를 들어 설명한다.

<88> 도 6a, 도 6b 및 도 6c는 모두 다시점 비디오 코딩의 표준화 과정에 있어서 테스트 영상 시퀀스의 하나인 영상 "Ballroom"을, 시간 방향의 영상 그룹의 크기를 12로 하고 H.264/AVC에 규정된 기존의 움직임 벡터 예측 방법을 이용하여 부호화하였을 경우에 대한 예이다. 여기서, 도 6a는 전체 예측 구조의 일부분을 보여주는 것으로서, 각 영상의 타입 및 번호와 영상들 사이의 참조 관계가 표시되어 있다. 그리고 도 6b는 도 6a에 도시된 예측 구조에서 S1 뷰의 T0 시간에 해당하는 영상 $B_1(1)$ 의 일부 매크로블록에 대한 움직임 정보를 확대하여 표시한 것이고, 도 6c는 도 6a에 도시된 예측 구조에서 S1 뷰의 T1 시간에 해당하는 영상 $b_4(21)$ 의 일부 매크로블록에 대한 움직임 정보를 확대하여 표시한 것이다. 도 6b 및 도 6c에서 각 매크로블록 내의 정보는 위에서부터 차례대로 움직임 벡터(MV), 움직임 벡터 예측값(MV_p), 움직임 벡터 차분값(MV_d), 및 참조 영상에 대한 정보(즉, 참조 영상의 번호)를 나타낸다.

<89> 먼저 도 6b를 참조하면, 영상 $B_1(1)$ 의 매크로블록 (12, 14)은 참조 영상의 번호가 0이므로 움직임 예측에서는 S0 뷰의 T0 시간에 해당하는 영상 $I_0(0)$ 을 참조하였고, 움직임 벡터는 (17.8, 2.0)이다. 그리고 매크로블록 (12, 14)의 위쪽과 우상쪽에 위치하는 매크로블록, 즉 매크로블록 (12, 13)과 (13, 13)에는 움직임 벡터(MV)에 관한 정보가 존재하지 않으므로(도면에는 not available을 나타내는 N/A로 표기되어 있음), 움직임 벡터 예측값(MV_p)은 매크로블록 (12, 14)의 왼쪽에 위치하는 매크로블록, 즉 매크로블록 (11, 14)의 움직임 벡터(MV)인 (11.8, 2.5)이 된다. 따라서 영상 $B_1(1)$ 의 매크로블록 (12, 14)에 대한 움직임 벡터 차분값($MV_d = MV - MV_p$)은 (6.0, -0.5)이다.

<90> 그리고 도 6c를 참조하면, 영상 $b_4(21)$ 의 매크로블록 (12, 14)은 참조 영상의 번호가 8이므로 움직임 예측에서는 S0 뷰의 T1 시간에 해당하는 영상 $B_3(8)$ 을 참조하였고, 움직임 벡터는 (18.3, 2.0)이다. 그리고 매크로블록 (12, 14)의 왼쪽에 위치하는 매크로블록 (11, 14)과 위쪽에 위치하는 매크로블록 (12, 13)의 움직임 벡터(MV)는 모두 (0.0, 0.0)이고, 우상쪽에 위치하는 매크로블록 (13, 13)은 움직임 벡터(MV)가 존재하지 않는다. 이와 같은 경우에는, 도 4를 참조하여 설명한 바와 같이, 블록 'D' 위치, 즉 좌상쪽에 위치하는 매크로블록 (11, 13)의 움직임 벡터(MV)를 이용하여 움직임 벡터 예측값(MV_p)을 구한다. 매크로블록 (11, 13)의 움직임 벡터(MV)는 (0.0, -0.3)이므로, 매크로블록 (11, 14), (12, 13), 및 (11, 13)의 움직임 벡터(MV)의 중간값인 (0.0, 0.0)이 움직임 벡터 예측값(MV_p)이 된다. 따라서 영상 $b_4(21)$ 의 매크로블록 (12, 14)에 대한 움직임 벡터 차분값(MV_d)은 (18.3, 2.0)이다.

<91> 도 6b와 도 6c의 경우를 비교하면, 현재 블록 (12, 14)의 움직임 벡터 차분값(MV_d)은 각각 (6.0, -0.5)과 (18.3, 2.0)으로서 도 6c의 경우가 훨씬 크다. 이것의 원인은 여러 가지가 있을 수 있지만, 그 중의 하나는 도 6b의 경우에는 현재 블록(12, 14)의 참조 영상과 움직임 벡터 예측값(MV_p)을 구하는데 이용되는 주변 블록(11, 14)의 참조 영상이 S0 뷰의 T0 시간의 영상 $I_0(0)$ 으로 같지만, 도 6c의 경우에는 현재 블록(12, 14)의 참조 영상은 S0 뷰의 T1 시간의 영상 $B_3(8)$ 이지만, 움직임 벡터 예측값(MV_p)을 구하는데 이용되는 주변 블록(11, 13),

(12, 13), 및 (11, 14)의 참조 영상은 S1 뷰의 T0 시간의 영상 $B_1(1)$ 으로서, 현재 블록의 참조 영상과 주변 블록의 참조 영상의 특성이 서로 다르다는 것이다.

<92> 따라서 도 6c의 경우와 같이, 현재 블록의 참조 영상과 움직임 벡터 예측값(MV_P)을 구할 때 이용되는 블록의 참조 영상의 특성이 다른 경우에는, 상대적으로 그 값이 큰 움직임 벡터 차분값(MV_D)을 부호화하여 전송해야 하므로 예측 부호화 효율이 떨어지게 된다. 하지만 본 발명에 따른 다시점 비디오 코딩에서의 움직임 벡터 예측 방법에서는, 전술한 (1), (2), 및 (3) 중의 어느 하나 또는 그 이상의 움직임 벡터(MV_S , MV_T 및/또는 MV_V)를 선택하여 움직임 벡터 예측값(MV_P)을 구하기 때문에, 이러한 문제점을 해결할 수가 있다. 도 6a 내지 도 6c를 참조하여 이에 대하여 구체적으로 설명한다.

<93> 부호화하려는 현재 블록이 도 6c의 매크로블록 (12, 14)라고 하자. 이 경우, 현재 블록 (12, 14)의 참조 영상은 현재 영상 $b_4(21)$ 의 시간과 동일한 T1 시간에서 S0 뷰에 위치하는 영상 $B_3(8)$ 이다. 즉, 움직임 예측에서 현재 블록(12, 14)은 뷰 방향(S0 뷰)의 영상을 참조한다. 그리고 도 6b에서 현재 블록과 동일한 위치에 위치하는 매크로블록 (12, 13)도 동일한 T0 시간에서 S0 뷰에 위치하는 영상 $I_0(0)$ 을 참조한다. 결국, 움직임 예측에서 현재 영상 $b_4(21)$ 의 매크로블록 (12, 14)이 참조하는 영상의 방향과 영상 $B_1(1)$ 의 매크로블록 (12, 14)이 참조하는 영상은 모두 같은 뷰의 영상으로서, 참조 영상의 특성이 동일하다.

<94> 이와 같은 경우에는 같은 뷰 내에서 상기 시간 방향 대응 블록의 움직임 벡터(MV_T)를 이용하여 움직임 벡터 예측값을 구하면 부호화 효율을 향상시킬 수가 있다. 전술한 예에서, 영상 $B_1(1)$ 의 매크로블록 (12, 14)의 움직임 벡터(MV)인 (17.8, 2.0)을 움직임 벡터 예측값(MV_P)으로 사용하면, 움직임 벡터 차분값(MV_D)은 (0.5, 0.0)가 된다. 이러한 결과는 움직임 벡터 차분값(MV_D)이 (18.3, 2.0)인 기존의 방법과 비교하였을 때, 움직임 벡터 차분값(MV_D)에 대한 정보량을 줄여서 예측 부호화 효율을 향상시킬 수가 있다는 것을 보여준다.

<95> 결국 본 발명에 따른 다시점 비디오 코딩에서의 움직임 벡터의 예측 방법은, 현재 블록(본 명세서에서 블록의 의미는 16X16단위의 매크로블록에 한정되지 않으며 여러 가지 크기의 블록이 될 수 있으며, 또는 소정의 크기를 갖는 영역(Region)이 될 수도 있다)의 주변 블록들의 움직임 벡터(MV_N)만을 이용하여 움직임 벡터 예측값(MV_P)을 구하는 기존의 방법과는 다른 예측 방법을 사용함으로써 예측 부호화 효율을 향상시킬 수 있다. 즉, 본 발명에 따른 다시점 비디오 코딩에서의 움직임 벡터의 예측 방법에서는 (1) 현재 블록의 주변 블록의 움직임 벡터(MV_N), (2) 현재 영상에 대한 '시간 방향 참조 영상'의 '시간 방향 대응 블록'들의 움직임 벡터(MV_T), 및/또는 (3) 현재 영상에 대한 '뷰 방향 참조 영상'의 '뷰 방향 대응 블록'들의 움직임 벡터(MV_V) 중에서 움직임 벡터를 선택하고, 선택된 움직임 벡터를 이용하여 움직임 벡터 예측값(MV)을 구한다.

<96> 움직임 벡터 예측값(MV_P)을 구하는데 이용할 수 있는 참조 블록들

<97> 도 7에는 본 발명에 따라서 현재 블록(current block)의 움직임 벡터 예측값(MV_P)을 구할 때 이용할 수 있는 참조 블록들이 도시되어 있다. 여기서 참조 블록은 예컨대 매크로블록(macroblock, MB)이 될 수 있다. 또는, 다른 예로써 8×8 화소 블록이거나 4×4 화소 블록이 될 수도 있으며, H.264/AVC에서 사용되는 임의의 크기를 갖는 블록(16×8, 8×16, 8×4, 4×8) 등이 될 수도 있다.

<98> 우선, 도 1에 도시된 다시점 비디오 코딩에서의 예측 구조를 참조하여 도 7에 대하여 보다 구체적으로 설명한다. 예를 들어, 현재 영상(도 7의 참조 번호 10을 포함하는 영상)이 도 1에서 S1 뷰의 T1 시간에서의 영상 b_4 이라면, 참조 번호 12를 포함하는 영상은 도 1에서 S1 뷰의 T0 시간에서의 영상 B_1 이 되고, 참조 번호 14를 포함하는 영상은 도 1에서 S1 뷰의 T2 시간에서의 영상 B_3 이 된다. 그리고 참조 번호 16을 포함하는 영상은 도 1에서 S0 뷰의 T1 시간에서의 영상 B_3 이 되고, 참조 번호 18을 포함하는 영상은 도 1에서 S2 뷰의 T1 시간에서의 영상 B_3 이 된다.

<99> 만일, 현재 영상이 도 1에서 S1뷰의 T2 시간에서의 영상 B_3 이라면, 참조 번호 12를 포함하는 영상은 도 1에서 S1 뷰의 T0 시간에서의 영상 B_1 이 되고, 참조 번호 14를 포함 영상은 도 1에서 S1 뷰의 T4 시간에서의 영상 B_2 가

된다. 그리고 참조 번호 16을 포함하는 영상은 도 1에서 S0 뷰의 T2 시간에서의 영상 B₂가 되고, 참조 번호 18을 포함하는 영상은 도 1에서 S2 뷰의 T2 시간에서의 영상 B₂가 된다.

<100> 도 7을 참조하면, 참조 번호 10은 기존의 H.264/AVC에서 움직임 벡터 예측값(MV_p)을 구할 때 이용하는 현재 블록의 주변 블록들이다. 본 발명의 일 실시예에 의하면, 주변 블록 'A', 'B', 'C', 및 'D'의 움직임 벡터(MV_N)를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다.

<101> 그리고 참조 번호 12와 참조 번호 14는 '시간 방향 참조 영상'의 '시간 방향 대응 블록'들이다. 보다 구체적으로, 참조 번호 12는 현재 영상과 동일한 뷰에서 시간 방향 리스트0(list0)의 참조 영상의 블록 또는 시간적으로 앞선 참조 영상의 블록을 나타내고, 참조 번호 14는 현재 영상과 동일한 뷰에서 시간 방향 리스트1(list1)의 참조 영상의 블록 또는 시간적으로 뒤지는 참조 영상의 블록을 나타낸다. 본 발명의 일 실시예에 의하면, 동일한 뷰 내에서 현재 영상보다 시간적으로 앞선 '시간 방향 참조 영상'에서 '시간 방향 대응 블록'인 블록 'TF0', 'TF1', 'TF2', 'TF3', 'TF4', 'TF5', 'TF6', 'TF7', 및 'TF8'와 현재 영상보다 시간적으로 뒤지는 '시간 방향 참조 영상'에서 '시간 방향 대응 블록'인 블록 'TB0', 'TB1', 'TB2', 'TB3', 'TB4', 'TB5', 'TB6', 'TB7', 및 'TB8'의 움직임 벡터(MV_T)를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다. 참조 번호 12와 14에서, 블록 'TF0' 및 'TB0'는 각각 현재 블록과 동일한 위치의 블록이고, 나머지 블록들은 그 주변 블록들이다.

<102> 또한 참조 번호 16와 참조 번호 18은 '뷰 방향 참조 영상'의 '뷰 방향 대응 블록'들을 나타낸다. 보다 구체적으로, 참조 번호 16은 현재 영상과 동일한 시간에서 뷰 방향 리스트0(list0)의 참조 영상의 블록 또는 앞쪽 뷰에서의 참조 영상의 블록을 나타내고, 참조 번호 18은 현재 영상과 동일한 시간에서 뷰 방향 리스트1(list1)의 참조 영상의 블록 또는 뒤쪽 뷰에서의 참조 영상의 블록을 나타낸다. 본 발명의 일 실시예에 의하면, 동일한 시간에서 현재 영상이 속한 뷰보다 앞쪽 뷰에서의 '뷰 방향 참조 영상'에서 '뷰 방향 대응 블록'인 블록 'VF0', 'VF1', 'VF2', 'VF3', 'VF4', 'VF5', 'VF6', 'VF7', 및 'VF8'과 현재 영상이 속한 뷰보다 뒤쪽 뷰에서의 '뷰 방향 참조 영상'에서 '뷰 방향 대응 블록'인 블록 'VB0', 'VB1', 'VB2', 'VB3', 'VB4', 'VB5', 'VB6', 'VB7', 및 'VB8'의 움직임 벡터(MV_V)를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다. 참조 번호 16 및 18에서, 블록 'VF0' 및 'VB0'은 현재 블록과 동일한 위치의 블록이고, 나머지 블록들은 그 주변 블록들이다.

<103> 현재 블록의 움직임 벡터 예측값(MV_p)을 구하는데 있어서, 전술한 '움직임 벡터 예측값(MV_p)을 구하는데 이용할 수 있는 참조 블록들 또는 그 참조 블록의 움직임 벡터들'의 전부가 아닌 그 일부를 적절하게 선택하여, 선택된 참조 블록들의 움직임 벡터만을 이용하는 것이 바람직하다. 즉, 움직임 벡터 예측값(MV_p)을 구하는데 있어서 실제로 이용하는 참조 블록 즉 선택되는 참조 블록의 집합은, 도 7에 도시되어 있는 모든 참조 블록들을 하나의 집합 S = {'A', 'B', 'C', 'D', 'TF0', 'TF1', 'TF2', 'TF3', 'TF4', 'TF5', 'TF6', 'TF7', 'TF8', 'TB0', 'TB1', 'TB2', 'TB3', 'TB4', 'TB5', 'TB6', 'TB7', 'TB8', 'VF0', 'VF1', 'VF2', 'VF3', 'VF4', 'VF5', 'VF6', 'VF7', 'VF8', 'VB0', 'VB1', 'VB2', 'VB3', 'VB4', 'VB5', 'VB6', 'VB7', 'VB8'}로 했을 때, S의 모든 부분 집합(subset) 중에서 어느 하나의 부분 집합에 속하는 참조 블록만을 이용할 수 있다. 예컨대, S의 부분 집합 중의 하나인 집합 S₁ = {'A', 'B', 'C', 'D', 'VF0', 'VB0', 'TF0', 'TB0'}를 선택하여, 집합 S₁에 포함되는 참조 블록의 전부 또는 일부를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다.

<104> 바람직하게는, 움직임 예측에서 현재 블록의 움직임 벡터(MV_C)가 참조하는 영상의 특성에 기초하여, 움직임 벡터 예측값(MV_p)을 구하기 위하여 실제로 이용하는 참조 블록의 집합인 상기 집합 S의 부분 집합을 한정할 수 있다. 예를 들어, (가) 현재 블록의 움직임 벡터(MV_C)가 뷰 방향의 영상을 참조하는 경우와 (나) 현재 블록의 움직임 벡터(MV_C)가 시간 방향의 영상을 참조하는 경우로 구분하여 선택되는 상기 집합 S의 부분 집합을 보다 구체적으로 한정할 수도 있다.

<105> 도 8은 다시점 비디오 코딩에서의 예측 구조의 다른 예를 보여주는 도면으로서, 3개의 뷰에서 3개의 연속된 시간 프레임에 대한 경우이다. 이하에서는, 도 8을 참조하여, '시간 방향 참조 영상'과 '뷰 방향 참조 영상'을 보다 구체적으로 설명한다.

<106> 도 8을 참조하면, S1 뷰의 T1 시간에서의 영상 b₄(21)가 현재 영상이라고 하면, 현재 영상 b₄(21)의 시간 방향 리스트0(list0)의 참조 영상 또는 시간적으로 앞선 참조 영상은 S1 뷰의 T0 시간에서의 영상 B₁이고, 현재 영상

$b_4(21)$ 의 시간 방향 리스트1(list1)의 참조 영상 또는 시간적으로 뒤진 참조 영상은 S1 뷰의 T2 시간에서의 영상 B_3 이다. 그리고 현재 영상 $b_4(21)$ 의 뷰 방향 리스트0(list0)의 참조 영상 또는 현재 영상이 속한 뷰보다 앞쪽 뷰의 참조 영상은 S0 뷰의 T1 시간에서의 영상 B_3 이고, 현재 영상 $b_4(21)$ 의 뷰 방향 리스트0(list0)의 참조 영상 또는 현재 영상이 속한 뷰보다 뒤쪽 뷰의 참조 영상은 S2 뷰의 T1 시간에서의 영상 B_3 이다.

- <107> 그리고 도 9는 도 7 및 도 8의 그림은 함께 도시한 것이다. 이하, 도 9를 참조하여 상기 (가)와 (나)의 경우에 선택되는 상기 집합 S의 부분 집합을 보다 구체적으로 한정하는 것에 대하여 구체적으로 설명한다.
- <108> (가) 움직임 예측에서 현재 블록이 '뷰 방향 참조 영상'을 참조하는 경우, 즉 현재 블록의 움직임 벡터(MV_C)가 뷰 방향으로 참조하는 경우(후술하는 예에서, (i)의 경우는 (ii), (iii)의 경우를 포함한다)
- <109> (i) 이 경우에 선택되는 상기 집합 S의 부분 집합 S_2 는 현재 블록의 주변 블록과 함께 '시간 방향 참조 영상'에서 '시간 방향 대응 블록'(즉, 도 7에서 참조 번호 12 및 14의 블록)으로 구성되도록 한정될 수 있다. 그리고 움직임 벡터 예측값(MV_P)은 이 집합 S_2 에 속하는 블록의 전부 또는 일부의 움직임 벡터들(MV_N 및/또는 MV_T)을 이용하여 구한다.
- <110> 예를 들어, 도 9에서 현재 영상이 S1 뷰의 T1 시간에서의 영상 b_4 이고 또한 현재 블록이 S0 뷰의 T1 시간에서의 영상 B_3 을 참조한다면, 집합 S_2 는 현재 블록의 주변 블록인 블록 'A', 'B', 'C', 및 'D'와 함께, 현재 영상 b_4 의 시간 방향 참조 영상인 S1 뷰의 T0 시간에서의 영상 B_1 과 S1 뷰의 T2 시간에서의 영상 B_3 에서 현재 블록의 시간 방향 대응 블록인 블록 'TF0', 'TF1', 'TF2', 'TF3', 'TF4', 'TF5', 'TF6', 'TF7', 'TF8', 'TB0', 'TB1', 'TB2', 'TB3', 'TB4', 'TB5', 'TB6', 'TB7', 및 'TB8'로 구성된다. 그리고 움직임 벡터 예측값(MV_P)은 집합 S_2 를 구성하는 블록의 전부 또는 일부의 움직임 벡터(MV 및/또는 MV_T)를 이용하여 구할 수 있다.
- <111> (ii) 실시예에 따라서는 움직임 벡터 예측값(MV_P)을 구할 때 이용되는 상기 집합 S_2 를 그 부분 집합인 후술하는 집합 S_3 로 보다 구체적으로 한정할 수도 있다. 예컨대, 상기 집합 S_3 은 현재 블록의 움직임 벡터(MV_C)가 참조하는 뷰와 같은 뷰 방향으로 참조하는 움직임 벡터를 갖는 블록들로만 구성되도록 한정할 수도 있다. 그리고 움직임 벡터 예측값(MV_P)은 상기 집합 S_3 에 속하는 블록의 전부 또는 일부의 움직임 벡터들(MV_N 및/또는 MV_T)을 이용하여 구한다. 예를 들어, 도 9에서 현재 블록의 움직임 벡터(MV_C)가 S0 뷰를 참조한다면, 상기 집합 S_2 를 구성하는 블록들 중에서 그 움직임 벡터가 S0 뷰를 참조하는 블록들로만 구성되도록 상기 집합 S_3 을 한정하고, 상기 집합 S_3 에 속하는 블록들의 전부 또는 일부의 움직임 벡터(MV_N 및/또는 MV_T)를 이용하여 움직임 벡터 예측값(MV_P)을 구할 수도 있다.
- <112> (iii) 그리고 상기 집합 S_2 에 속하는 블록들 중에서 상기 집합 S_3 에도 속하는 블록이 없는 경우에는, 기존의 방법과 마찬가지로 주변 블록의 움직임 벡터(MV_N)만을 이용하여 움직임 벡터 예측값(MV_P)을 구할 수도 있다. 예컨대, 현재 블록의 움직임 벡터(MV_C)가 S0 뷰를 참조하고, 상기 집합 S_2 를 구성하는 블록들 중에서 그 움직임 벡터(MV_N 및/또는 MV_T)가 S0 뷰를 참조하는 블록이 하나도 없다면, 주변 블록인 블록 'A', 'B', 'C', 및 'D'의 전부 또는 일부의 움직임 벡터(MV_N)를 이용하여 움직임 벡터 예측값(MV_P)을 구할 수 있다.
- <113> (나) 움직임 예측에서 현재 블록이 '시간 방향 참조 영상'을 참조하는 경우, 즉 현재 블록의 움직임 벡터(MV_C)가 시간 방향으로 참조하는 경우(후술하는 예에서, (i)의 경우는 (ii), (iii)의 경우를 포함한다).
- <114> (i) 이 경우에 선택되는 상기 집합 S의 부분 집합 S_4 는 현재 블록의 주변 블록과 함께 '뷰 방향 참조 영상'에서 '뷰 방향 대응 블록'(즉, 도 7에서 참조 번호 16 및 18의 블록)으로 구성되도록 한정될 수 있다. 그리고 움직임 벡터 예측값(MV_P)은 이 집합 S_4 에 속하는 블록의 전부 또는 일부의 움직임 벡터들(MV_N 및/또는 MV_V)을 이용하여 구한다.
- <115> 예를 들어, 도 9에서 현재 영상이 S1 뷰의 T1 시간에서의 영상 b_4 이고 또한 현재 블록이 S1 뷰의 T0 시간에서의 영상 B_1 을 참조한다면, 집합 S_4 는 현재 블록의 주변 블록인 블록 'A', 'B', 'C', 및 'D'와 함께, 현재 영상 b_4 의

뷰 방향 참조 영상인 S0 뷰의 T1 시간에서의 영상 B₃과 S2 뷰의 T1 시간에서의 영상 B₃에서 현재 블록의 뷰 방향 대응 블록인 블록 'VF0', 'VF1', 'VF2', 'VF3', 'VF4', 'VF5', 'VF6', 'VF7', 'VF8', 'VB0', 'VB1', 'VB2', 'VB3', 'VB4', 'VB5', 'VB6', 'VB7', 및 'VB8'로 구성된다. 그리고 움직임 벡터 예측값(MV_P)은 집합 S₄를 구성하는 블록의 전부 또는 일부의 움직임 벡터(MV 및/또는 MV_V)를 이용하여 구할 수 있다.

<116> (ii) 실시예에 따라서는 움직임 벡터 예측값(MV_P)을 구할 때 이용되는 상기 집합 S₄를 그 부분 집합인 후술하는 집합 S₅로 보다 구체적으로 한정할 수도 있다. 예컨대, 상기 집합 S₅은 현재 블록의 움직임 벡터(MV_C)가 참조하는 시간대와 같은 시간대의 영상을 참조하는 움직임 벡터를 갖는 블록들로만 구성되도록 한정할 수도 있다. 그리고 움직임 벡터 예측값(MV_P)은 상기 집합 S₅에 속하는 블록의 전부 또는 일부의 움직임 벡터들(MV_N 및/또는 MV_V)을 이용하여 구한다. 예를 들어, 도 9에서 현재 블록의 움직임 벡터(MV)가 T0 시간대를 참조한다면, 상기 집합 S₄를 구성하는 블록들 중에서 그 움직임 벡터(MV)가 T0 시간대를 참조하는 블록들로만 구성되도록 상기 집합 S₅을 한정하고, 상기 집합 S₅에 속하는 블록들의 전부 또는 일부의 움직임 벡터(MV_N 및/또는 MV_V)를 이용하여 움직임 벡터 예측값(MV_P)을 구할 수도 있다.

<117> (iii) 그리고 상기 집합 S₄에 속하는 블록들 중에서 상기 집합 S₅에도 속하는 블록이 없는 경우에는, 기존의 방법과 마찬가지로 현재 블록의 주변 블록의 움직임 벡터(MV_N)만을 이용하여 움직임 벡터 예측값(MV_P)을 구할 수도 있다. 예컨대, 현재 블록의 움직임 벡터(MV)가 T0 시간대를 참조하고, 상기 집합 S₄를 구성하는 블록들 중에서 그 움직임 벡터(MV_N 및/또는 MV_V)가 T0 시간대를 참조하는 블록이 하나도 없다면, 현재 블록의 주변 블록인 블록 'A', 'B', 'C', 및 'D'의 전부 또는 일부의 움직임 벡터(MV_N)를 이용하여 움직임 벡터 예측값(MV_P)을 구할 수 있다.

<118> 움직임 벡터 예측값(MV_P)을 구하는데 이용되는 블록의 선택 방법에 대한 알고리즘

<119> 본 발명에 의하면, 현재 블록의 움직임 벡터 예측값(MV_P)을 구하는데 있어서 상기 집합 S의 부분 집합 중의 하나에 속하는 모든 블록 또는 일부 블록의 움직임 벡터(MV)를 선택하고 선택된 움직임 벡터를 이용한다. 그리고 선택되는 상기 집합 S의 부분 집합은, 전술한 바와 같이, 현재 블록의 움직임 벡터(MV_C)가 뷰 방향의 영상을 참조하는지(상기 (가)의 경우) 또는 시간 방향의 영상을 참조하는지(상기 (나)의 경우)에 기초하여 보다 구체적으로 한정될 수 있다. 앞에서 상세하게 설명한 바와 같이, 상기 (가)의 경우에 선택되는 상기 집합 S의 부분 집합은 현재 블록의 주변 블록 및/또는 시간 방향 대응 블록들의 전부 또는 일부로 구성될 수 있고, 상기 (나)의 경우에 선택되는 상기 집합 S의 부분 집합은 현재 블록의 주변 블록 및/또는 뷰 방향 대응 블록들의 전부 또는 일부로 구성될 수 있다. 그리고 전술한 (가)-(ii), (가)-(iii), (나)-(ii), 및 (나)-(iii) 중에서 어디에 해당하는가에 따라서 선택되는 상기 집합 S의 부분 집합은 더 한정될 수도 있다. 이하에서는 전술한 (가)-(ii) 및 (가)-(iii)에 해당하는 경우와 (나)-(ii) 및 (나)-(iii)에 해당하는 경우에 적용할 수 있는 알고리즘의 일례에 대하여 설명한다.

<120> 도 10에는 현재 블록의 움직임 벡터(MV_C)가 뷰 방향의 영상을 참조하는 경우에 움직임 벡터 예측값(MV_P)을 구하는 방법의 일례를 보여주는 흐름도가 도시되어 있다. 전술한 바와 같이, 이 경우에는 집합 S₂ = {'A', 'B', 'C', 'D', 'TF0', 'TF1', 'TF2', 'TF3', 'TF4', 'TF5', 'TF6', 'TF7', 'TF8', 'TB0', 'TB1', 'TB2', 'TB3', 'TB4', 'TB5', 'TB6', 'TB7', 'TB8'}의 부분 집합이 움직임 벡터 예측값(MV_P)을 구하는데 이용되기 위하여 선택될 수 있다.

<121> 도 10을 참조하면, 우선 집합 S₂를 구성하는 블록들 중에서 주변 블록인 블록 'A', 'B', 'C', 및 'D'의 움직임 벡터(MV_N) 중에서 뷰 방향의 영상을 참조하는 것이 있는지를 판단한다(S11). 상기 판단 결과, 하나 이상의 움직임 벡터가 존재하면 단계 S14로 분기하고, 하나도 존재하지 않으면 단계 S12로 분기한다.

<122> 그리고 단계 S12에서는 시간 방향 대응 블록인 블록 'TF0', 'TF1', 'TF2', 'TF3', 'TF4', 'TF5', 'TF6', 'TF7', 'TF8', 'TB0', 'TB1', 'TB2', 'TB3', 'TB4', 'TB5', 'TB6', 'TB7', 및 'TB8'의 움직임 벡터(MV_T)들 중에서 뷰 방향의 영상을 참조하는 것이 있는지를 판단한다(S12). 상기 판단 결과, 하나 이상의 움직임 벡터가

존재하면 단계 S13으로 분기하고, 하나도 존재하지 않으면 단계 S14로 분기한다.

- <123> 계속해서 상기 단계 S13, 즉 시간 방향 대응 블록의 움직임 벡터(MV_T)들 중에서 뷰 방향의 영상을 참조하는 것이 하나 이상 있으면, 해당되는 움직임 벡터(MV_T)의 전부 또는 일부를 이용하여 움직임 벡터 예측값(MV_P)을 구한다. 또한, 실시예에 따라서는 주변 블록의 움직임 벡터(MV_N)와 해당되는 시간 방향 대응 블록의 움직임 벡터(MV_T)를 함께 이용하여 움직임 벡터 예측값(MV_P)을 구할 수도 있다.
- <124> 그리고 단계 S14, 즉 주변 블록의 움직임 벡터(MV_N)들 중에서 뷰 방향의 영상을 참조하는 것이 하나 이상이 있거나 또는 시간 방향 대응 블록의 움직임 벡터(MV_T)들 중에서 뷰 방향의 영상을 참조하는 것이 하나도 없는 경우에는, 주변 블록의 움직임 벡터(MV_N)의 전부 또는 일부만을 이용하여 현재 블록의 움직임 벡터 예측값(MV_P)을 구한다.
- <125> 도 11에는 현재 블록의 움직임 벡터(MV_C)가 시간 방향의 영상을 참조하는 경우에 움직임 벡터 예측값(MV_P)을 구하는 방법의 일례를 보여주는 흐름도가 도시되어 있다. 전술한 바와 같이, 이 경우에는 집합 $S_4 = \{ 'A', 'B', 'C', 'D', 'VF0', 'VF1', 'VF2', 'VF3', 'VF4', 'VF5', 'VF6', 'VF7', 'VF8', 'VB0', 'VB1', 'VB2', 'VB3', 'VB4', 'VB5', 'VB6', 'VB7', 'VB8' \}$ 의 부분 집합이 움직임 벡터 예측값(MV_P)을 구하는데 이용되기 위하여 선택될 수 있다.
- <126> 도 11을 참조하면, 우선 집합 S_4 를 구성하는 블록들 중에서 주변 블록인 블록 'A', 'B', 'C', 및 'D'의 움직임 벡터(MV_N) 중에서 시간 방향의 영상을 참조하는 것이 있는지를 판단한다(S21). 상기 판단 결과, 해당되는 움직임 벡터가 하나 이상이 존재하면 단계 S24로 분기하고, 하나도 존재하지 않으면 단계 S22로 분기한다.
- <127> 그리고 단계 S22에서는 뷰 방향 대응 블록인 블록 'VF0', 'VF1', 'VF2', 'VF3', 'VF4', 'VF5', 'VF6', 'VF7', 'VF8', 'VB0', 'VB1', 'VB2', 'VB3', 'VB4', 'VB5', 'VB6', 'VB7', 및 'VB8'의 움직임 벡터(MV_V) 중에서 시간 방향의 영상을 참조하는 것이 있는지를 판단한다(S22). 상기 판단 결과, 해당되는 움직임 벡터가 하나 이상이 존재하면 단계 S23으로 분기하고, 하나도 존재하지 않으면 단계 S24로 분기한다.
- <128> 계속해서 단계 S23, 즉 뷰 방향 대응 블록의 움직임 벡터(MV_V)들 중에서 시간 방향의 영상을 참조하는 것이 하나 이상이 존재하면, 해당되는 움직임 벡터(MV_V)의 전부 또는 일부를 이용하여 움직임 벡터 예측값(MV_P)을 구한다. 또한, 실시예에 따라서는 주변 블록의 움직임 벡터(MV_N)와 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 함께 이용하여 움직임 벡터 예측값(MV_P)을 구할 수도 있다.
- <129> 그리고 단계 S24, 즉 주변 블록의 움직임 벡터(MV_N)들 중에서 시간 방향의 영상을 참조하는 것이 하나 이상이 있거나 또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)들 중에서 시간 방향의 영상을 참조하는 것이 하나도 없는 경우에는, 주변 블록의 움직임 벡터(MV_N)의 전부 또는 일부를 이용하여 현재 블록의 움직임 벡터 예측값(MV_P)을 구한다.
- <130> 움직임 벡터 예측값(MV_P)을 구하는데 이용되는 블록이 다양한 크기의 움직임 추정을 사용하는 경우
- <131> 현재 표준화가 진행 중인 다시점 비디오 코딩에서는 기본적으로 H.264/AVC에 규정된 내용을 기초로 하고 있기 때문에, 본 발명의 실시예에 따라서 움직임 벡터 예측값(MV_P)을 구하는데 이용되는 블록도 다양한 크기의 움직임 추정을 사용할 수 있다. 이러한 경우에는 움직임 벡터 예측값(MV_P)을 구할 때 이용되는 블록, 즉 전술한 집합 S에 속하는 각각의 블록은 그 파티션 블록의 수에 해당하는 움직임 벡터(MV)를 가진다. 따라서 움직임 벡터 예측값(MV_P)을 구하는데 이용되는 블록이 다양한 크기의 움직임 추정을 사용하는 경우에는, 그 블록의 파티션 블록들 중에서 하나의 파티션 블록을 선택하고, 선택된 파티션 블록의 움직임 벡터(MV)만을 이용하여 움직임 벡터 예측값(MV_P)을 구하도록 하는 것이 바람직하다. 이 경우에 선택되는 파티션 블록의 위치는 예측 부호화 효율을 고려하여 적절하게 결정될 수 있다.
- <132> 도 12에는 움직임 벡터 예측값(MV_P)을 구하는데 이용되는 블록이 여러 개의 파티션 블록으로 나뉘어 있을 경우,

움직임 벡터 예측값(MV_p)을 구하는데 실제로 사용하는 파티션 블록의 위치가 도식적으로 예시되어 있다. 도 12를 참조하면, 첫 번째 도면은 해당 블록의 우측 상단에 위치한 파티션 블록이 선택되는 경우이고, 두 번째 도면은 해당 블록의 우측 하단에 위치한 파티션 블록이 선택되는 경우이고, 세 번째 도면은 해당 블록의 좌측 하단에 위치한 파티션 블록이 선택되는 경우이고, 그리고 네 번째 도면은 해당 블록의 좌측 상단에 위치한 파티션 블록이 선택되는 경우이다.

- <133> A. 해당 블록이 도 7의 참조 번호 10에 해당하는 블록, 즉 주변 블록이고, 그 주변 블록이 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에, 움직임 벡터 예측값(MV)은 다음과 같이 구할 수 있다.
- <134> (i) 도 7의 블록 'A'가 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에는 블록 'A'의 최우측 상단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <135> (ii) 도 7의 블록 'B'가 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에는 블록 'B'의 최좌측 하단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <136> (iii) 도 7의 블록 'C'가 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에는 블록 'C'의 최좌측 하단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <137> (iv) 도 7의 블록 'D'가 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에는 블록 'D'의 최우측 하단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <138> B. 참조 블록이 도 7의 참조 번호 12, 14에 해당하는 블록, 즉 시간 방향 대응 블록이고, 그 시간 방향 대응 블록이 여러 가지 모양의 파티션 블록으로 나누어져 있거나 또는 참조 블록이 도 7의 참조 번호 16, 18에 해당하는 블록, 즉 뷰 방향 대응 블록이고, 그 뷰 방향 대응 블록이 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에, 움직임 벡터 예측값(MV_p)은 다음에 같이 구할 수 있다.
- <139> (i) 도 7의 블록 'TF0', 'TB0', 'VF0', 또는 'VB0'이 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에는 해당 블록의 최좌측 상단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <140> (ii) 도 7의 블록 'TF1', 'TB1', 'VF1', 또는 'VB1'이 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에는 해당 블록의 최우측 상단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <141> (iii) 도 7의 블록 'TF2', 'TB2', 'VF2', 또는 'VB2'가 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에는 해당 블록의 최좌측 하단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <142> (iv) 도 7의 블록 'TF3', 'TB3', 'VF3', 또는 'VB3'이 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에는 해당 블록의 최좌측 상단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <143> (v) 도 7의 블록 'TF4', 'TB4', 'VF4', 또는 'VB4'가 여러 가지 모양의 파티션으로 나누어져 있을 경우에는 해당 블록의 최좌측 상단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <144> (vi) 도 7의 블록 'TF5', 'TB5', 'VF5', 또는 'VB5'가 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에는 해당 블록의 최우측 하단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <145> (vii) 도 7의 블록 'TF6', 'TB6', 'VF6', 또는 'VB6'이 여러 가지 모양의 파티션으로 나누어져 있을 경우에는 해당 블록의 최좌측 하단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <146> (viii) 도 7의 블록 'TF7', 'TB7', 'VF7', 또는 'VB7'이 여러 가지 모양의 파티션 블록으로 나누어져 있을 경우에는 해당 블록의 최우측 상단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <147> (ix) 도 7의 블록 'TF8', 'TB8', 'VF8', 또는 'VB8'이 여러 가지 모양의 파티션으로 블록으로 나누어져 있을 경우에는 해당 블록의 최좌측 상단에 위치한 파티션 블록의 움직임 벡터를 이용한다.
- <148> 예를 들어, 움직임 벡터 예측값(MV_p)을 구하기 위하여 이용할 수 있는 블록의 집합으로 전술한 부분 집합 $S_1 = \{ 'A', 'B', 'C', 'D', 'VF0', 'VB0', 'TF0', 'TB0' \}$ 이 선택되었고, 상기 부분 집합 S_1 을 구성하는 각 블록이 여러 가지 모양의 파티션 블록으로 나누어져 있다고 하자. 이 경우에, 블록 'A'의 경우에는 전술한 A-(i)를 적용하고, 블록 'B'의 경우에는 전술한 A-(ii)를 적용하고, 블록 'C'의 경우에는 전술한 A-(iii)를 적용하고, 블록 'D'의 경우에는 전술한 A-(iv)를 적용하고, 블록 'VF0' 'VB0', 'TF0', 및 'TB0'의 경우에는 전술한 B-(i)를

적용함으로써, 적용되는 파티션 블록의 움직임 벡터의 전부 또는 일부를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다.

<149> 현재 블록이 8×16 픽셀 크기인 경우

<150> 도 5a에 도시된 바와 같이 현재 블록이 8×16 픽셀 크기를 가지는 경우, 움직임 벡터 예측값(MV_p)을 구할 때 이용하는 블록은 8×16 좌측 블록과 8×16 우측 블록인 경우를 나누어서 생각할 수 있다.

<151> (가) 8×16 좌측 블록의 움직임 벡터 예측값(MV_p)을 구할 때 이용할 수 있는 블록의 집합은 상기 집합 S의 다른 부분 집합인 다음의 집합 S₅로 한정될 수 있다.

<152> (i) 상기 집합 S₅는 주변 블록인 블록 'A'(도 7 참조)만을 포함하도록 한정될 수 있다. 이 경우, 블록 'A'의 움직임 벡터를 움직임 벡터 예측값(MV_p)으로 바로 사용한다.

<153> (ii) 다른 예로써, 현재 블록이 뷰 방향의 움직임 벡터를 갖는 경우에, 상기 집합 S₅는 주변 블록인 블록 'A'와 시간 방향 대응 블록인 블록 'TF0', 'TF1', 'TF2', 'TF4', 'TF5', 'TF7', 'TB0', 'TB1', 'TB2', 'TB4', 'TB5', 및 'TB7'(도 7 참조)을 포함하도록 한정될 수 있다. 이 경우, 상기 집합 S₅ = {'A', 'TF0', 'TF1', 'TF2', 'TF4', 'TF5', 'TF7', 'TB0', 'TB1', 'TB2', 'TB4', 'TB5', 'TB7'}이 되며, 이 집합 S₅에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다. 예컨대, 집합 S₅의 부분 집합의 하나인 블록 'TF5', 'TF1', 'TF7', 'TB5', 'TB1', 및 'TB7'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수 있다. 또한 집합 S₅의 다른 부분 집합인 블록 'A', 'TF1', 및 'TB1'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수도 있다.

<154> (iii) 또 다른 예로써 현재 블록이 시간 방향의 움직임 벡터를 갖는 경우에, 상기 집합 S₅는 주변 블록인 블록 'A'와 뷰 방향 대응 블록인 블록 'VF0', 'VF1', 'VF2', 'VF4', 'VF5', 'VF7', 'VB0', 'VB1', 'VB2', 'VB4', 'VB5', 및 'VB7'(도 7 참조)을 포함하도록 한정될 수 있다. 이 경우, 상기 집합 S₅ = {'A', 'VF0', 'VF1', 'VF2', 'VF4', 'VF5', 'VF7', 'VB0', 'VB1', 'VB2', 'VB4', 'VB5', 'VB7'}가 되며, 이 집합 S₅에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다. 예컨대, 집합 S₅의 부분 집합의 하나인 블록 'VF5', 'VF1', 'VF7', 'VB5', 'VB1', 및 'VB7'의 움직임 벡터(MV)의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수 있다. 또한 집합 S₅의 다른 부분 집합인 블록 'A', 'VF1', 및 'VB1'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수도 있다.

<155> (나) 8×16 우측 블록의 움직임 벡터 예측값(MV_p)을 구할 때 이용할 수 있는 블록의 집합은 상기 집합 S의 다른 부분 집합인 다음의 집합 S₆으로 한정될 수 있다.

<156> (i) 상기 집합 S₆은 참조 블록인 블록 'C'(도 7 참조)만을 포함하도록 한정될 수 있다. 이 경우, 블록 'C'의 움직임 벡터를 움직임 벡터 예측값(MV_p)으로 바로 사용한다.

<157> (ii) 다른 예로써 현재 블록이 뷰 방향의 움직임 벡터를 갖는 경우에, 상기 집합 S₆은 주변 블록인 블록 'C'와 시간 방향 대응 블록인 블록 'TF0', 'TF2', 'TF3', 'TF4', 'TF6', 'TF8', 'TB0', 'TB2', 'TB3', 'TB4', 'TB6', 및 'TB8'을 포함하도록 한정될 수 있다. 이 경우, 상기 집합 S₆ = {'C', 'TF0', 'TF2', 'TF3', 'TF4', 'TF6', 'TF8', 'TB0', 'TB2', 'TB3', 'TB4', 'TB6', 'TB8'}이 되며, 이 집합 S₆에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다. 예컨대, 집합 S₆의 부분 집합의 하나인 블록 'TF6', 'TF3', 'TF8', 'TB6', 'TB3', 및 'TB8'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수 있다. 또한 집합 S₆의 다른 부분 집합인 블록 'C', 'TF3', 및 'TB3'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수도 있다.

<158> (iii) 또 다른 예로써 현재 블록이 시간 방향의 움직임 벡터를 갖는 경우에, 상기 집합 S₆은 주변 블록인 블록 'C'와 뷰 방향 대응 블록인 블록 'VF0', 'VF2', 'VF3', 'VF4', 'VF6', 'VF8', 'VB0', 'VB2', 'VB3', 'VB4',

'VB6', 및 'VB8'(도 7 참조)을 포함하도록 한정될 수 있다. 이 경우, 상기 집합 $S_6 = \{ 'C', 'VF0', 'VF2', 'VF3', 'VF4', 'VF6', 'VF8', 'VB0', 'VB2', 'VB3', 'VB4', 'VB6', 'VB8' \}$ 이 되며, 이 집합 S_6 에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다. 예컨대, 집합 S_6 의 부분 집합의 하나인 블록 'VF6', 'VF3', 'VF8', 'VB6', 'VB3', 및 'VB8'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수 있다. 또한 집합 S_6 의 다른 부분 집합인 블록 'C', 'VF3', 및 'VB3'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수도 있다.

<159> 현재 블록이 16×8 픽셀 크기인 경우

<160> 도 5b에 도시된 바와 같이 현재 블록이 16×8 픽셀 크기를 가지는 경우에, 움직임 벡터 예측값(MV_p)을 구할 때 이용할 수 있는 블록은 현재 블록이 16×8 상단 블록과 16×8 하단 블록인 경우를 나눠서 생각할 수 있다.

<161> (가) 16×8 상단 블록의 움직임 벡터 예측값(MV_p)을 구할 때 이용할 수 있는 블록의 집합은 상기 집합 S 의 다른 부분 집합인 다음의 집합 S_7 로 한정될 수 있다.

<162> (i) 상기 집합 S_7 은 주변 블록인 블록 'B'(도 7 참조)만을 포함하도록 한정될 수 있다. 이 경우, 블록 'B'의 움직임 벡터를 움직임 벡터 예측값(MV_p)으로 바로 사용한다.

<163> (ii) 다른 예로써 현재 블록의 움직임 벡터(MV_c)가 뷰 방향의 영상을 참조하는 경우에, 상기 집합 S_7 은 주변 블록인 블록 'B'와 시간 방향 대응 블록인 블록 'TF0', 'TF1', 'TF2', 'TF3', 'TF5', 'TF6', 'TB0', 'TB1', 'TB2', 'TB3', 'TB5', 및 'TB6'을 포함하도록 한정될 수 있다. 이 경우, 상기 집합 $S_7 = \{ 'B', 'TF0', 'TF1', 'TF2', 'TF3', 'TF5', 'TF6', 'TB0', 'TB1', 'TB2', 'TB3', 'TB5', 'TB6' \}$ 이 되며, 이 집합 S_7 에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다. 예컨대, 집합 S_7 의 부분 집합의 하나인 블록 'TF5', 'TF2', 'TF6', 'TB5', 'TB2', 및 'TB6'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수 있다. 또한 집합 S_7 의 다른 부분 집합인 블록 'B', 'TF2', 및 'TB2'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수도 있다.

<164> (iii) 또 다른 예로써 현재 블록의 움직임 벡터(MV_c)가 시간 방향의 영상을 참조하는 경우에, 상기 집합 S_7 은 주변 블록 'B'와 뷰 방향 대응 블록인 블록 'VF0', 'VF1', 'VF2', 'VF3', 'VF5', 'VF6', 'VB0', 'VB1', 'VB2', 'VB3', 'VB5', 및 'VB6'을 포함하도록 한정될 수 있다. 이 경우, 상기 집합 $S_7 = \{ 'B', 'VF0', 'VF1', 'VF2', 'VF3', 'VF5', 'VF6', 'VB0', 'VB1', 'VB2', 'VB3', 'VB5', 'VB6' \}$ 이 되며, 이 집합 S_7 에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다. 예컨대, 집합 S_7 의 부분 집합의 하나인 블록 'VF5', 'VF2', 'VF6', 'VB5', 'VB2', 및 'VB6'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수 있다. 또한 집합 S_7 의 다른 부분 집합인 블록 'B', 'VF2', 및 'VB2'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수도 있다.

<165> (나) 16×8 하단 블록의 움직임 벡터 예측값(MV_p)을 구할 때 이용할 수 있는 블록의 집합은 상기 집합 S 의 다른 부분 집합인 다음의 집합 S_8 로 한정될 수 있다.

<166> (i) 상기 집합 S_8 은 주변 블록인 블록 'A'만을 포함하도록 한정될 수 있다. 이 경우, 블록 'A'의 움직임 벡터를 움직임 벡터 예측값(MV_p)으로 바로 사용한다.

<167> (ii) 다른 예로써 현재 블록의 움직임 벡터(MV_c)가 뷰 방향의 영상을 참조하는 경우에, 상기 집합 S_8 은 주변 블록인 블록 'A'와 시간 방향 대응 블록인 블록 'TF0', 'TF1', 'TF3', 'TF4', 'TF7', 'TF8', 'TB0', 'TB1', 'TB3', 'TB4', 'TB7', 및 'TB8'을 포함하도록 한정될 수 있다. 이 경우, 상기 집합 $S_8 = \{ 'A', 'TF0', 'TF1', 'TF3', 'TF4', 'TF7', 'TF8', 'TB0', 'TB1', 'TB3', 'TB4', 'TB7', 'TB8' \}$ 이 되며, 이 집합 S_8 에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다. 예컨대, 집합 S_8 의 부분 집합의 하나인 블록 'TF7', 'TF4', 'TF8', 'TB7', 'TB4', 및 'TB8'의 움직임 벡터의 중간값을 구하여, 이

것을 움직임 벡터 예측값(MV_p)으로 할 수 있다. 또한 집합 S_8 의 다른 부분 집합인 블록 'A', 'TF4', 및 'TB4'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV)으로 할 수도 있다.

<168> (iii) 또 다른 예로써 현재 블록의 움직임 벡터(MV_c)가 시간 방향의 영상을 참조하는 경우에, 상기 집합 S_8 은 주변 블록인 블록 'A', 뷰 방향 대응 블록인 블록 'VF0', 'VF1', 'VF3', 'VF4', 'VF7', 'VF8', 'VB0', 'VB1', 'VB3', 'VB4', 'VB7', 및 'VB8'을 포함하도록 한정될 수 있다. 이 경우, 상기 집합 $S_8 = \{ 'A', 'VF0', 'VF1', 'VF3', 'VF4', 'VF7', 'VF8', 'VB0', 'VB1', 'VB3', 'VB4', 'VB7', 'VB8' \}$ 이 되며, 이 집합 S_8 에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다. 예컨대, 집합 S_8 의 부분 집합의 하나인 블록 'VF7', 'VF4', 'VF8', 'VB7', 'VB4', 및 'VB8'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV_p)으로 할 수 있다. 또한 집합 S_8 의 다른 부분 집합인 블록 'A', 'VF4', 및 'VB4'의 움직임 벡터의 중간값을 구하여, 이것을 움직임 벡터 예측값(MV)으로 할 수도 있다.

<169> 참조 영상의 거리를 고려하는 경우

<170> 영상간 예측 부호화에서 현재 영상(현재 블록)이 시간 방향이나 뷰 방향으로 참조하는 참조 영상, 즉 시간 방향 참조 영상 또는 뷰 방향 참조 영상까지의 거리에 따라서 적응적으로 움직임 벡터 예측값(MV_p)을 구하는데 이용하는 블록을 선택할 수 있다. 이것은 현재 블록의 움직임 벡터 예측값(MV_p)을 구하는데 있어서, 현재 영상이 참조하는 참조 영상까지의 거리가 길 경우에는, 현재 블록의 움직임 벡터(MV_c)와 시간 방향 대응 블록의 움직임 벡터(MV_T)나 뷰 방향 대응 블록의 움직임 벡터(MV_V)가 차이가 크게 나거나 또는 서로 다른 특성을 가질 수 있는 가능성이 높아져, 움직임 벡터 예측값(MV_p)이 정확하지 않을 수 있다는 점을 고려한 것이다. 따라서 현재 영상으로부터 참조 영상까지의 거리가 길수록, 움직임 벡터 예측값(MV_p)을 구하는데 이용하는 참조 블록의 수를 감소시키거나 또는 그 참조 영상으로부터의 예측을 허용하지 않도록 함으로써, 예측 부호화 효율을 보다 향상시킬 수 있다. 이하에서는 이에 대하여 보다 구체적으로 설명한다.

<171> 현재 블록의 움직임 벡터(MV_c)가 뷰 방향의 영상을 참조한다고 가정하자. 이러한 경우에는 현재 영상에서 시간 방향 참조 영상까지의 거리에 따라서 움직임 벡터 예측값(MV_p)을 구하는데 이용되는 시간 방향 대응 블록의 집합을 보다 구체적으로 한정할 수 있다.

<172> 예를 들어, 현재 영상에서 시간 방향 참조 영상까지의 거리가 1인 경우에는, 움직임 벡터 예측값(MV_p)을 구하는데 이용할 수 있는 참조 블록을 도 7의 참조 번호 10에 해당하는 주변 블록들과 도 7의 참조 번호 12 및 14에 해당하는 모든 시간 방향 대응 블록으로 한정할 수 있다. 즉, 움직임 벡터 예측값(MV_p)을 구하기 위하여 이용할 수 있는 블록의 집합을 상기 집합 S 의 부분 집합인 집합 $S_9 = \{ 'A', 'B', 'C', 'D', 'TF0', 'TF1', 'TF2', 'TF3', 'TF4', 'TF5', 'TF6', 'TF7', 'TF8', 'TB0', 'TB1', 'TB2', 'TB3', 'TB4', 'TB5', 'TB6', 'TB7', 'TB8' \}$ 로 한정할 수 있다. 그리고 집합 S_9 에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구한다.

<173> 그리고 현재 영상에서 상기 시간 방향 참조 영상까지의 거리가 2인 경우에는, 움직임 벡터 예측값(MV_p)을 구하는데 이용할 수 있는 블록을 도 7의 참조 번호 10의 주변 블록들과 참조 번호 12 및 14에서 현재 블록과 동일한 위치의 블록인 블록 'TF0', 'TB0'으로 보다 더 한정할 수 있다. 즉, 움직임 벡터 예측값(MV_p)을 구하기 위하여 이용할 수 있는 블록의 집합을 상기 집합 S 의 다른 부분 집합인 집합 $S_{10} = \{ 'A', 'B', 'C', 'D', 'TF0', 'TB0' \}$ 으로 한정할 수 있다. 그리고 집합 S_{10} 에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구한다.

<174> 또한, 현재 영상에서 상기 시간 방향 참조 영상까지의 거리가 3이상일 경우에는, 시간 방향 대응 블록의 움직임 벡터(MV_T)는 이용하지 않고 주변 블록의 움직임 벡터(MV_N)만을 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다.

<175> 표 1은 현재 영상에서 상기 시간 방향 참조 영상까지의 거리에 따라 적응적으로 블록을 선택하여 움직임 벡터

예측값(MV_p)을 구하는 전술한 예들을 정리한 것이다.

표 1

<176>	시간 방향 참조 영상까지의 거리(D)	움직임 벡터 예측값(MV_p)을 구하는데 이용할 수 있는 참조 블록의 집합
	D=1	'A', 'B', 'C', 'D', 'TF0', 'TF1', 'TF2', 'TF3', 'TF4', 'TF5', 'TF6', 'TF7', 'TF8', 'TB0', 'TB1', 'TB2', 'TB3', 'TB4', 'TB5', 'TB6', 'TB7', 'TB8'
	D=2	'A', 'B', 'C', 'D', 'TF0', 'TB0'
	$3 \leq D$	'A', 'B', 'C', 'D'

<177> 다른 예로써, 현재 블록의 움직임 벡터(MV)가 시간 방향의 영상을 참조한다고 하자. 이러한 경우에는 현재 영상에서 뷰 방향 참조 영상까지의 거리에 따라서 움직임 벡터 예측값(MV)을 구하는데 이용되는 블록의 집합을 보다 구체적으로 한정할 수 있다.

<178> 예를 들어, 현재 영상에서 뷰 방향 참조 영상까지의 거리가 1 인 경우에는, 움직임 벡터 예측값(MV_p)을 구하는데 이용할 수 있는 블록을 도 7의 참조 번호 10에 해당하는 주변 블록들과 도 7의 참조 번호 16 및 18에 해당하는 뷰 방향 대응 블록으로 한정할 수 있다. 즉, 움직임 벡터 예측값(MV_p)을 구하기 위하여 이용할 수 있는 블록의 집합을 상기 집합 S의 다른 부분 집합인 집합 $S_{11} = \{ 'A', 'B', 'C', 'D', 'VF0', 'VF1', 'VF2', 'VF3', 'VF4', 'VF5', 'VF6', 'VF7', 'VF8', 'VB0', 'VB1', 'VB2', 'VB3', 'VB4', 'VB5', 'VB6', 'VB7', 'VB8' \}$ 로 한정할 수 있다. 그리고 집합 S_{11} 에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV_p)을 구한다.

<179> 그리고 현재 영상에서 상기 뷰 방향 참조 영상까지의 거리가 2인 경우에는, 움직임 벡터 예측값(MV_p)을 구하는데 이용할 수 있는 블록을 도 7의 참조 번호 10에서 주변 블록들과 참조 번호 16 및 18에서 현재 블록과 동일한 위치의 뷰 방향 대응 블록인 블록 'VF0' 및 'VB0'으로 보다 더 한정할 수 있다. 즉, 움직임 벡터 예측값(MV_p)을 구하기 위하여 이용할 수 있는 블록의 집합을 상기 집합 S의 다른 부분 집합인 집합 $S_{12} = \{ 'A', 'B', 'C', 'D', 'VF0', 'VB0' \}$ 으로 한정할 수 있다. 그리고 집합 S_{12} 에 속하는 블록들의 전부 또는 일부의 움직임 벡터를 이용하여 움직임 벡터 예측값(MV)을 구한다.

<180> 또한, 현재 영상에서 상기 뷰 방향 참조 영상까지의 거리가 3이상일 경우에는, 뷰 방향 대응 블록의 움직임 벡터(MV_v)는 이용하지 않고 주변 블록들의 움직임 벡터(MV_N)만을 이용하여 움직임 벡터 예측값(MV_p)을 구할 수 있다.

<181> 표 2는 현재 영상에서 상기 뷰 방향 참조 영상까지의 거리에 따라 적응적으로 블록을 선택하여 움직임 벡터 예측값(MV_p)을 구하는 전술한 예들을 정리한 것이다.

표 2

<182>	뷰 방향 참조 영상까지의 거리(D)	움직임 벡터 예측값(MV_p)을 구하는데 이용할 수 있는 참조 블록의 집합
	D=1	'A', 'B', 'C', 'D', 'VF0', 'VF1', 'VF2', 'VF3', 'VF4', 'VF5', 'VF6', 'VF7', 'VF8', 'VB0', 'VB1', 'VB2', 'VB3', 'VB4', 'VB5', 'VB6', 'VB7', 'VB8'
	D=2	'A', 'B', 'C', 'D', 'VF0', 'VB0'
	$3 \leq D$	'A', 'B', 'C', 'D'

<183> 움직임 벡터의 예측 방법을 나타내는 선택스

<184> 전술한 바와 같이, 본 발명에 따른 다시점 비디오 코딩에서는 현재 블록의 움직임 벡터 예측값(MV_p)을 구할 때,

(1) 주변 블록의 움직임 벡터(MV_N), (2) '시간 방향 참조 영상'에서 '시간 방향 대응 블록'의 움직임 벡터(MV_T), 및/또는 (3) '뷰 방향 참조 영상'에서 '뷰 방향의 참조 블록'의 움직임 벡터(MV_V)를 이용한다. 이러한 본 발명에 따른 다시점 비디오 코딩에서의 움직임 벡터 예측 방법은 가변장 부호, 고정장 부호, 및/또는 Exp-Golomb 부호 등을 사용하여 시퀀스 헤더(sequence header), 슬라이스 헤더(slice header), 및/또는 매크로블록 헤더(macroblock header) 등에 적절하게 표현할 수 있다.

<185> 본 발명의 바람직한 일 실시예에 따르면, H.264/AVC에 규정된 기존의 방법, 즉 상기 (1)의 움직임 벡터(MV_N)만을 이용하여 움직임 벡터 예측값(MV_P)을 구하는 것에 추가하여, (2)의 움직임 벡터(MV_T) 및/또는 (3)의 움직임 벡터(MV_V)도 함께 이용하는 것이다. 이러한 경우에는 (1)의 움직임 벡터(MV_N)를 이용하는 것은 H.264/AVC에 규정된 것과 동일한 방법으로 표현할 수 있다. 따라서 이 경우에는 (2)의 움직임 벡터(MV_T) 및/또는 (3)의 움직임 벡터(MV_V)를 이용할지 여부에 대해서만 추가적으로 표현하면 된다. 이하에서는 (2)의 움직임 벡터(MV_T) 및/또는 (3)의 움직임 벡터(MV_V)를 이용할지 여부에 대하여 여러 가지 선택스에 표현하는 방법에 대하여 설명한다. 다만, 후술하는 예에서 (1)의 움직임 벡터(MV_N)를 이용하는 것을 표현하는 방법이 H.264/AVC에 규정된 기존의 방법에 한정되는 것은 아니다.

<186> (i) 움직임 벡터 예측값(MV_P)을 구할 때 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부에 관한 정보(코드)를 슬라이스 헤더에 표현할 수 있다. 예를 들어, 시간 방향 대응 블록의 움직임 벡터(MV_T)를 이용할지에 대한 정보가 슬라이스 헤더에 플래그 정보(flag information)로 표현되는 경우에, 그 플래그가 '1'이 되면 시간 방향 대응 블록의 움직임 벡터(MV_T)를 이용하고, 그 플래그가 '0'이 되면 시간 방향의 참조 블록의 움직임 벡터(MV)를 이용하지 않는 것으로 할 수 있다.

<187> 전술한 예에서, 현재 블록의 움직임 벡터(MV_C)가 뷰 방향의 영상을 참조하고 또한 주변 블록들의 움직임 벡터(MV_N)를 이용할 수 없거나 주변 블록의 움직임 벡터(MV_N)들 중에서 뷰 방향의 영상을 참조하는 것이 없는 경우에는, 시간 방향 대응 블록의 움직임 벡터(MV_T)를 이용하여 움직임 벡터 예측값(MV_P)을 구하면 부호화 효율을 향상시킬 수가 있다. 이러한 경우에는 슬라이스 헤더에 상기 시간 방향 대응 블록을 이용한다는 플래그 정보가 포함된다.

<188> 그러나 시간 방향으로 장면의 전환이 있거나 또는 급격한 밝기의 변화가 발생하였을 경우에는 상기 시간 방향 대응 블록의 움직임 벡터(MV_T)는 현재 블록의 움직임 벡터(MV_C)와는 전혀 다른 값을 가질 가능성이 높다. 이러한 경우에는 상기 시간 방향 대응 블록의 움직임 벡터(MV_T)가 현재 블록의 움직임 벡터(MV_C)와 같은 뷰 방향의 영상을 참조하는 경우라도, 움직임 벡터 예측값(MV_P)을 구하는데 있어서 상기 시간 방향 대응 블록의 움직임 벡터(MV_T)는 이용하지 않는 것이 바람직하다. 즉, 시간 방향으로 장면의 전환이 있거나 또는 급격한 밝기의 변화가 발생하였을 경우인데도 불구하고, 상기 시간 방향 대응 블록의 움직임 벡터(MV_T)를 이용하여 움직임 벡터 예측값(MV_P)을 구하고 이 움직임 벡터 예측값(MV_P)으로 움직임 벡터 차분값(MV_D)을 구하면, 움직임 벡터 차분값(MV_D)의 정보량이 증가하여 오히려 예측 부호화의 효율이 나빠질 수가 있다. 따라서 이러한 경우에는 슬라이스 헤더에 상기 시간 방향의 참조 블록을 이용하지 않는다는 플래그 정보를 포함시키는 것이 바람직하다.

<189> 이와 같이, 움직임 벡터 예측값(MV_P)을 구할 때 상기 시간 방향 대응 블록의 움직임 벡터(MV_T)를 이용할지 여부에 관한 정보를 슬라이스 헤더(Slice Header)에 표현함으로써, 예측 부호화 효율을 보다 향상시킬 수가 있다. 그리고 이러한 원리는 뷰 방향 대응 블록의 움직임 벡터(MV)에 대한 경우에도 동일하게 적용할 수 있다.

<190> 표 3에는 슬라이스 헤더에 시간 방향 대응 블록의 움직임 벡터(MV_T)를 이용할지 여부를 나타내는 플래그 정보를 포함하는 선택스의 일례가 예시되어 있다. 표 3을 참조하면, 상기 슬라이스 헤더에 대한 선택스는 리스트 0(list0)에 해당하는 시간 방향 참조 영상에 대한 예측의 수행 여부를 나타내는 플래그(flag)인 "use_list0_motion_prediction_flag"과 리스트1(list1)에 해당하는 시간 방향 참조 영상에 대한 예측의 수행 여부를 나타내는 플래그인 "use_list1_motion_prediction_flag"를 포함한다. 여기서, "use_list0_motion_prediction_flag"는 P-영상나 B-영상일 경우에 해당되고,

"use_list1_motion_prediction_flag"는 B-영상일 경우에만 해당된다. 그리고 표 3은 슬라이스 헤더에 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부를 나타내는 플래그 정보를 포함하는 신택스의 경우에도 동일하게 적용이 가능하다.

표 3

slice_header() {	C	Descriptor
first_mb_in_slice	2	ue(T)
slice_type	2	ue(T)
pic_parameter_set_id	2	ue(T)
frame_num	2	u(T)
...		
if((profile == MULTI_VIEW_PROFILE) && slice_Type != I && slice_Type != SI) {		
use_list0_motion_prediction_flag	2	u(1)
if(slice_type == B) {		
use_list1_motion_prediction_flag	2	u(1)
}		
}		
...		
}		

다른 실시예로써 표 3에 정의된 각각의 플래그의 의미를 다르게 사용할 수도 있다. 예컨대, "use_list0_motion_prediction_flag"는 움직임 벡터 예측값(MV_P)을 구하는데 있어서, 현재 영상보다 시간적으로 이전의 '시간 방향 참조 영상'을 이용하여 움직임 벡터 예측을 수행할지 여부를 표현하는 플래그로 사용하고, "use_list1_motion_prediction_flag"는 움직임 벡터 예측값(MV_P)을 구하는데 있어서, 현재 영상보다 시간적으로 이후의 '시간 방향 참조 영상'을 이용하여 움직임 벡터 예측을 수행할지 여부를 표현하는 플래그로 사용할 수도 있다.

또 다른 실시예로써, 슬라이스 헤더에 표현하는 움직임 벡터 예측값(MV_P)을 구할 때 상기 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부에 관한 정보는 표 3에 정의된 것과 같은 플래그 정보 대신에 H.264/AVC에서 사용하는 Exp-Golomb 코드, 가변장 부호, 또는 고정장 부호 등을 사용하여 표현할 수도 있다.

전술한 실시예들에서, 슬라이스 단위에서 시간 방향 대응 블록들의 움직임 벡터(MV_T) 또는 뷰 방향 대응 블록들의 움직임 벡터(MV_V)를 움직임 벡터 예측값(MV_P)을 구하는데 이용할지 여부는 여러 가지 방법으로 결정할 수 있다. 이러한 결정 방법의 일례로서 슬라이스 단위로 픽셀들의 휘도값의 평균값을 이용할 수 있다. 예컨대 시간 방향 대응 블록들의 움직임 벡터(MV_T)를 이용할 지 여부를 결정하는 경우에, 현재 영상과 시간 방향 참조 영상에서 슬라이스 단위로 각 픽셀의 휘도값의 평균값을 각각 구하여, 그 평균값의 차이가 소정의 값(α) 이상일 경우에는 시간 방향 대응 블록들의 움직임 벡터(MV_T)를 이용하지 않고, 반대로 그 평균값의 차이가 소정의 값(α)과 같거나 그 보다 작을 경우에는 시간 방향 대응 블록들의 움직임 벡터(MV_T)를 이용하도록 결정할 수 있다.

(ii) 움직임 벡터 예측값(MV_P)을 구할 때, 상기 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 상기 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부에 관한 정보를 시퀀스 헤더에 대한 신택스에 표현할 수도 있다. 그리고 실시예에 따라서는, 상기 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 상기 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부에 관한 정보는 매크로블록에 대한 신택스에 표현할 수도 있다.

(iii) 전술한 (i) 및 (ii)의 방법을 함께 사용하거나 또는 어느 하나의 방법만을 사용할 수도 있다.

- <197> 선택된 블록의 움직임 벡터들(MV_L , MV_T , 및/또는 MV_V)로부터 움직임 벡터 예측값(MV_p)을 구하는 방법
- <198> 움직임 벡터 예측값(MV_p)을 구하는데 이용하기 선택된 움직임 벡터가 두 개 이상인 경우에, 그 선택된 움직임 벡터들을 이용하여 움직임 벡터 예측값(MV_p)을 구하는 방법은 여러 가지가 있다. 예컨대, H.264/AVC에 규정된 것과 같이 선택된 움직임 벡터들의 중간값을 구하여, 이 중간값을 움직임 벡터 예측값(MV_p)으로 할 수 있다. 그리고 선택된 움직임 벡터들의 평균값을 움직임 벡터 예측값(MV_p)으로 하거나 또한 선택된 움직임 벡터들 중에서 현재 블록의 움직임 벡터와의 차이가 가장 작은 것을 움직임 벡터 예측값(MV_p)으로 할 수도 있다. 이하에서는, N개의 움직임 벡터들을 선택하고 이 선택된 움직임 벡터들의 중간값을 구하는 경우에 대하여 상세하게 설명한다.
- <199> (i) 전술한 집합 S나 또는 그 부분 집합, 예컨대 상기 집합 S_1 내지 S_{12} 중의 하나의 집합에서 N개의 움직임 벡터를 선택하기 위한 N개의 블록을 선택하는 일례로서, 상기 집합 S에 속하는 블록들 사이의 우선순위(이것은 절대적일 수도 있고, 상대적일 수도 있다)를 정하여 그 우선순위에 따라 N개의 블록을 선택할 수 있다. 우선순위는 어떠한 임의의 순서도 가질 수 있다. 예를 들어, 주변 블록(도 7의 참조 번호 10에 해당하는 블록)들 사이에서는 블록 'A', 'B', 'C', 및 'D'의 순서로 우선순위를 정할 수 있다.
- <200> 그리고 시간 방향 대응 블록이나 뷰 방향 대응 블록들 사이에서는 그 블록의 위치에 따라서 우선순위를 정할 수 있다. 예컨대, 현재 블록과 동일한 위치의 블록에 대해서는 높은 우선순위를 할당하고, 현재 블록과 동일한 위치의 블록에 대하여 수평 또는 수직 방향으로 위치하는 블록에 대하여 그 다음으로 높은 우선순위를 할당하고, 현재 블록과 동일한 위치의 블록에 대하여 대각선 방향으로 위치하는 블록에 대하여 가장 낮은 우선순위를 할당할 수 있다. 도 7을 참조하여 보다 구체적으로 살펴보면, 현재 블록과 동일한 위치의 블록인 블록 'TF0'에 가장 높은 우선순위를 할당하고, 블록 'TF0'에 대하여 수평 및 수직 방향에 위치하는 블록인 블록 'TF1', 'TF2', 'TF3', 및 'TF4'에 그 다음으로 높은 우선순위를 할당하고, 'TF0' 블록에 대하여 대각선 방향에 위치하는 블록인 블록 'TF5', 'TF6', 'TF7', 및 'TF8'에 가장 낮은 우선순위를 할당할 수 있다. 그리고 블록 'TF1', 'TF2', 'TF3', 및 'TF4'와 블록 'TF5', 'TF6', 'TF7', 및 'TF8'는 각각 임의의 순서로 우선순위를 할당하거나 나열된 순서대로 우선순위를 할당할 수 있다.
- <201> (ii) 상기 (i)의 예에서 움직임 벡터 예측값(MV_p)을 구하기 위하여 이용할 수 있는 블록의 개수가 N보다 작을 경우, 예컨대 $M(1 < M < N)$ 개일 경우에는, 그 M개 블록의 움직임 벡터를 이용하여 중간값을 구하고 그 중간값을 움직임 벡터 예측값(MV_p)으로 할 수 있다. 또는, 전술한 예에서 N-M개의 움직임 벡터(MV)를 (0.0, 0.0)으로 설정하고, N개의 움직임 벡터에 대한 중간값 연산을 통하여 움직임 벡터 예측값(MV_p)을 구할 수도 있다.
- <202> (iii) 상기 (i)의 예에서 움직임 벡터 예측값(MV_p)을 구하기 위하여 이용할 수 있는 블록의 개수가 1개일 경우에는, 예컨대 그 1개의 블록의 움직임 벡터(MV)를 움직임 벡터 예측값(MV_p)으로 할 수도 있다.
- <203> (iv) 상기 (i)의 예에서 움직임 벡터 예측값(MV_p)을 구하기 위하여 이용할 수 있는 블록이 존재하지 않을 경우에는, 기존의 방법과 마찬가지로 주변 블록의 움직임 벡터(MV)로부터 우선순위에 따라 N개의 블록을 선택하여, 선택된 N개의 블록의 움직임 벡터(MV)의 중간값을 움직임 벡터 예측값(MV_p)로 할 수 있다. 다른 예로서, 중간값을 구하지 않고 움직임 벡터 예측값(MV_p)을 (0.0, 0.0)로 할 수도 있다.
- <204> 도 13에는 전술한 실시예에 따라서 N개 또는 그 보다 작은 개수의 움직임 벡터에 대한 중간값 연산을 통하여 움직임 벡터 예측값(MV_p)을 구하는 방법을 보여주는 흐름도가 도시되어 있다. 이하에서는 도 13을 참조하여 상기 흐름도에 대하여 상세히 설명한다.
- <205> 우선, 상기 집합 S 또는 그 부분 집합으로서, 전술한 본 발명의 실시예들 중의 하나 또는 실시예들의 조합에 따라서, 움직임 벡터 예측값(MV_p)을 구하기 위하여 이용할 수 있는 블록의 개수(M)를 계산한다(S31).
- <206> 그리고 상기 블록의 수(M)와 N개의 크기를 비교한다(S32). 판단 결과, $M < N$ 이면 단계 S34로 분기하고, $M \geq N$ 이면 단계 S33으로 분기한다.
- <207> 단계 S33에서는 상기 M개의 블록들 중에서 우선순위에 따라 N개의 블록을 선택하고, 그 선택된 N개의 블록의 움직임 벡터들에 대한 중간값을 구한다(S33). 이렇게 구해진 중간값이 현재 블록에 대한 움직임 벡터 예측값

(MV_p)이 되며, 이 경우에는 여기서 절차가 종료된다.

- <208> 그리고 단계 S34에서는 M 이 0인지 아닌지를 판단한다(S34). 판단 결과, M 이 0이 아니면 단계 S36으로 분기하고, M 이 0이면 단계 S35로 분기한다.
- <209> 단계 S35에서는 주변 블록들 중에서 N 개의 블록을 선택하여, 그 선택된 블록의 움직임 벡터들에 대한 중간값을 구한다(S35). 이렇게 구해진 중간값이 현재 블록에 대한 움직임 벡터 예측값(MV_p)이 되며, 이 경우에는 여기서 절차가 종료된다.
- <210> 그리고 단계 S36에서는 M 이 1인지 아닌지를 판단한다(S36). 판단 결과, M 이 1이라면 단계 S38로 분기하고, M 이 1이 아니라면 단계 S37로 분기한다.
- <211> 단계 S37에서는 ($N-M$)개의 움직임 벡터(MV)를 (0.0, 0.0)로 설정하고, M 개의 블록의 움직임 벡터(MV)와 (0.0, 0.0)로 설정된 ($N-M$)개의 움직임 벡터들에 대한 중간값을 구한다(S37). 이렇게 구해진 중간값이 현재 블록에 대한 움직임 벡터 예측값(MV_p)이 되며, 이 경우에는 여기서 절차가 종료된다.
- <212> 그리고 단계 S38에서는 M 이 1인 경우이므로, 그 블록의 움직임 벡터(MV)를 움직임 벡터 예측값(MV_p)으로 하고, 절차가 종료된다(S38).
- <213> 본 발명의 실시예에 따라서는 시간 방향 대응 블록이나 뷰 방향 대응 블록을 이용하지 않고 움직임 벡터 예측값(MV_p)을 구할 수도 있다. 이 경우에는 H.264/AVC에 규정되어 있는 기존의 방법과 같이, 현재 블록의 주변 블록들만을 이용하여 움직임 벡터 예측값(MV_p)을 구한다. 즉, 도 7의 참조 번호 10에 해당하는 블록 'A', 'B', 및 'C'의 움직임 벡터(MV_N)들에 대한 중간값 연산을 통하여 움직임 벡터 예측값(MV_p)을 구하거나, 또는 블록 'C'의 움직임 벡터(MV)가 존재하지 않을 경우에는 블록 'C'의 대신에 블록 'D'를 사용하여, 블록 'A', 'B', 및 'D'의 움직임 벡터(MV_N)에 대한 중간값 연산을 통하여 움직임 벡터 예측값(MV_p)을 구한다.
- <214> 움직임 벡터 예측값(MV_p)을 구하는 방법의 응용예
- <215> 앞에서 설명한 방법들을 조합하면, 움직임 벡터 예측값(MV_p)을 구하는 방법은 다양하게 만들어 낼 수 있다. 이하에서는 도 14를 참조하여 그 중의 한 가지 방법에 대해서 설명한다.
- <216> 도 14에는 본 발명의 일 실시예에 따라서 움직임 벡터 예측값(MV_p)을 구하는 방법을 보여주는 흐름도가 도시되어 있다.
- <217> 도 14를 참조하면, 우선 현재 영상으로부터 현재 영상이 시간 방향으로 참조하는 영상(시간 방향 참조 영상)까지의 시간 방향의 거리(D)를 계산한다(S501). 여기서, 상기 시간 방향의 거리(D)는 예컨대, 현재 영상과 참조 영상의 프레임의 차이를 의미할 수 있다. 예를 들어, 현재 영상이 T2 시간의 영상이고 참조 영상이 T0 시간의 영상이면, 시간 방향의 거리(D)는 2가 된다.
- <218> 그리고 시간 방향의 거리(D)가 2이하인지 여부를 판단한다(S502). 판단 결과, D 가 2이하이면 단계 S503으로 분기하고, D 가 2보다 크면 단계 S517로 분기한다.
- <219> 그리고 단계 S503에서는 주변 블록의 움직임 벡터(MV_N)들 중에서 현재 블록의 움직임 벡터(MV_C)가 참조하는 영상과 같은 영상을 참조하는 것이 있는지 여부를 판단한다(S503). 판단 결과, 주변 블록의 움직임 벡터(MV_N)들 중에서 현재 블록의 움직임 벡터(MV_C)가 참조하는 영상과 같은 영상을 참조하는 블록이 있는 경우에는 단계 S517로 분기하고, 그렇지 않으면 단계 S504로 분기한다.
- <220> 그리고 단계 S504에서는 현재 블록의 움직임 벡터(MV_C)가 뷰 방향 참조 영상을 참조하는지 또는 시간 방향 참조 영상을 참조하는지를 판단한다(S504). 판단 결과, 현재 블록의 움직임 벡터(MV_C)가 뷰 방향 참조 영상을 참조하는 경우에는 단계 S505로 분기하고, 시간 방향 참조 영상을 참조하는 경우에는 단계 S511로 분기한다.
- <221> 그리고 단계 S505에서는 시간 방향 참조 영상의 시간 방향 대응 블록의 움직임 벡터(MV_T)들 중에서, 현재 블록의 움직임 벡터(MV_C)가 참조하는 뷰와 같은 뷰의 영상을 참조하는 블록의 개수(N_T)가 0인지 여부를 판단한다(S505). 판단 결과, 같은 뷰의 영상을 참조하는 블록의 개수(N_T)가 0이 아닌 경우에는 단계 S506으로

분기하고, 0인 경우에는 단계 S517로 분기한다.

- <222> 그리고 단계 S506에서는 현재 블록의 움직임 벡터(MV_C)가 참조하는 뷰와 같은 뷰의 영상을 참조하는 시간 방향 대응 블록의 개수(N_T)가 3미만인지 여부를 판단한다(S506). 판단 결과, 같은 뷰의 영상을 참조하는 시간 방향 대응 블록의 개수(N_T)가 3미만이면 단계 S508로 분기하고, 3이상이면 단계 S507로 분기한다.
- <223> 그리고 단계 S507에서는 현재 블록의 움직임 벡터(MV_C)가 참조하는 뷰와 같은 뷰를 참조하는 시간 방향 대응 블록의 움직임 벡터(MV_T)들 중에서 우선순위에 따라 3개의 움직임 벡터를 선택하고, 선택된 3개의 움직임 벡터(MV_T)에 대한 중간값 연산을 통하여 움직임 벡터 예측값(MV_P)을 구한다(S507). 이와 같이 움직임 벡터 예측값(MV_P)을 구하면 절차가 종료된다.
- <224> 그리고 단계 S508에서는 현재 블록의 움직임 벡터(MV_C)가 참조하는 뷰와 같은 뷰를 참조하는 시간 방향 대응 블록의 개수(N_T)가 1인지 여부를 판단한다(S508). 판단 결과, 같은 뷰를 참조하는 시간 방향 대응 블록의 개수(N_T)가 1인 경우에는 단계 S510으로 분기하고, 같은 뷰를 참조하는 시간 방향 대응 블록의 개수(N_T)가 2인 경우에는 단계 S509로 분기한다.
- <225> 그리고 단계 S509에서는 현재 블록의 움직임 벡터(MV_C)가 참조하는 뷰와 같은 뷰를 참조하는 시간 방향 대응 블록의 움직임 벡터(MV_T)들과 움직임 벡터 (0.0, 0.0)에 대한 중간값 연산을 통하여 움직임 벡터 예측값(MV_P)을 구한다(S509). 이와 같이 움직임 벡터 예측값(MV_P)을 구하면 절차가 종료된다.
- <226> 그리고 단계 S510에서는 현재 블록의 움직임 벡터(MV_C)가 참조하는 뷰와 같은 뷰를 참조하는 시간 방향 대응 블록의 움직임 벡터(MV_T)를 움직임 벡터 예측값(MV_P)으로 한다(S510). 이와 같이 움직임 벡터 예측값(MV_P)을 구하면 절차가 종료된다.
- <227> 그리고 단계 S511에서는 뷰 방향 참조 영상의 뷰 방향 대응 블록의 움직임 벡터(MV_V)들 중에서, 현재 블록의 움직임 벡터(MV_C)가 참조하는 시간대와 같은 시간대의 영상을 참조하는 블록의 개수(N_V)가 0인지 여부를 판단한다(S511). 판단 결과, 같은 시간대의 영상을 참조하는 블록의 개수(N_V)가 0이 아닌 경우에는 단계 S512로 분기하고, 0인 경우에는 단계 S517로 분기한다.
- <228> 그리고 단계 S512에서는 현재 블록의 움직임 벡터(MV_C)가 참조하는 시간대와 같은 시간대의 영상을 참조하는 뷰 방향 대응 블록의 개수(N_V)가 3미만인지 여부를 판단한다(S512). 판단 결과, 같은 시간대의 영상을 참조하는 뷰 방향 대응 블록의 개수(N_V)가 3미만이면 단계 S514로 분기하고, 3이상이면 단계 S513로 분기한다.
- <229> 그리고 단계 S513에서는 현재 블록의 움직임 벡터(MV_C)가 참조하는 시간대와 같은 시간대를 참조하는 뷰 방향 대응 블록의 움직임 벡터(MV_V)들 중에서 우선순위에 따라 3개의 움직임 벡터를 선택하고, 선택된 3개의 움직임 벡터(MV_V)에 대한 중간값 연산을 통하여 움직임 벡터 예측값(MV_P)을 구한다(S513). 이와 같이 움직임 벡터 예측값(MV_P)을 구하면 절차가 종료된다.
- <230> 그리고 단계 S514에서는 현재 블록의 움직임 벡터(MV_C)가 참조하는 시간대와 같은 시간대를 참조하는 뷰 방향 대응 블록의 개수(N_V)가 1인지 여부를 판단한다(S514). 판단 결과, 같은 시간대를 참조하는 뷰 방향 대응 블록의 개수(N_V)가 1인 경우에는 단계 S516으로 분기하고, 같은 시간대를 참조하는 뷰 방향 대응 블록의 개수(N_V)가 2인 경우에는 단계 S515로 분기한다.
- <231> 그리고 단계 S515에서는 현재 블록의 움직임 벡터(MV_C)가 참조하는 시간대와 같은 시간대를 참조하는 뷰 방향 대응 블록의 움직임 벡터(MV_V)들과 움직임 벡터 (0.0, 0.0)에 대한 중간값 연산을 통하여 움직임 벡터 예측값(MV_P)을 구한다(S515). 이와 같이 움직임 벡터 예측값(MV_P)을 구하면 절차가 종료된다.
- <232> 그리고 단계 S516에서는 현재 블록의 움직임 벡터(MV_C)가 참조하는 시간대와 같은 시간대를 참조하는 뷰 방향

대응 블록의 움직임 벡터(MV_V)를 움직임 벡터 예측값(MV_P)으로 한다(S516). 이와 같이 움직임 벡터 예측값(MV_P)을 구하면 절차가 종료된다.

<233> 마지막으로 단계 S517에서는 H.264/AVC에 규정되어 있는 기존의 방법과 동일한 방법, 즉 현재 블록의 주변 블록의 움직임 벡터(MV_N)들만을 이용하여 움직임 벡터 예측값(MV_P)을 구한다(S517). 이와 같이 움직임 벡터 예측값(MV_P)을 구하면 절차가 종료된다.

<234> 도 14를 참조하여 설명한 본 발명의 실시예에 대한 변형 실시예 또는 보다 구체적인 실시예로서, 후술하는 다음과 같은 방법들 중에서 하나 이상의 방법을 추가적으로 사용할 수도 있다.

<235> 1) 움직임 벡터 예측값(MV_P)을 구할 경우에 시간 방향 대응 블록의 움직임 벡터(MV_T)이나 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부를 나타내는 플래그 정보를 시퀀스 헤드의 선택스에 포함시킬 수 있다. 예컨대, 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할 경우에는 상기 플래그를 부호 '1'로 할당하고, 이용하지 않을 경우에는 부호 '0'으로 할당할 수 있으며, 반대의 경우도 가능하다.

<236> 2) 움직임 벡터 예측값(MV_P)을 구할 경우에 시간 방향 대응 블록의 움직임 벡터(MV_T)이나 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부를 나타내는 플래그 정보를 슬라이스 헤드의 선택스에 포함시킬 수 있다. 예컨대, 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할 경우에는 상기 플래그를 부호 '1'로 할당하고, 이용하지 않을 경우에는 부호 '0'으로 할당할 수 있으며, 반대의 경우도 가능하다.

<237> 3) 움직임 벡터 예측값(MV_P)을 구하기 위하여 참조하는 블록, 예컨대 주변 블록, 시간 방향 대응 블록, 또는 뷰 방향 대응 블록이 여러 가지 크기의 파티션 블록으로 나뉘어 있을 경우에는, 상기 블록의 위치에 따라서 파티션 블록을 적절하게 선택할 수 있다. 도 7을 참조하여 이러한 경우에 대한 예를 보다 구체적으로 설명하면, 블록 'A'는 우측 상단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'B'는 좌측 하단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'C'는 좌측 하단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'D'는 우측 하단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'TF0', 'TB0', 'VF0', 및 'VB0'은 좌측 상단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'TF1', 'TB1', 'VF1', 및 'VB1'은 우측 상단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'TF2', 'TB2', 'VF2', 및 'VB2'은 좌측 하단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'TF3', 'TB3', 'VF3', 및 'VB3'은 좌측 상단 파티션의 움직임 벡터를 사용하고, 블록 'TF4', 'TB4', 'VF4', 'VB4'은 좌측 상단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'TF5', 'TB5', 'VF5', 및 'VB5'은 우측 하단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'TF6', 'TB6', 'VF6', 및 'VB6'은 좌측 하단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'TF7', 'TB7', 'VF7', 및 'VB7'은 우측 상단에 위치한 파티션 블록의 움직임 벡터를 사용하고, 블록 'TF8', 'TB8', 'VF8', 및 'VB8'은 좌측 상단에 위치한 파티션 블록의 움직임 벡터를 사용할 수 있다.

<238> 4) 현재 블록이 16×8 픽셀 크기의 블록일 경우에는 전술한 실시예를 약간 변형하여 그대로 적용할 수 있다. 예컨대, 전술한 단계 S503 및 S516에서, 현재 블록이 16×8 픽셀 크기의 상단 블록인 경우에 상기 주변 블록은 블록 'B'만 해당하고, 현재 블록이 16×8 픽셀 크기의 하단 블록인 경우에는 상기 주변 블록은 블록 'A'만 해당하는 것으로 변형하여, 전술한 실시예를 그대로 적용할 수 있다.

<239> 5) 현재 블록이 8×16 픽셀 크기의 블록일 경우에도 전술한 실시예를 약간 변형하여 그대로 적용할 수 있다. 예컨대, 전술한 단계 S503 및 S516에서, 현재 블록이 8×16 픽셀 크기의 좌측 블록인 경우에는 상기 주변 블록은 블록 'A'만 해당하고, 현재 블록이 8×16 픽셀 크기의 우측 블록인 경우에는 상기 주변 블록은 블록 'C'만 해당하는 것으로 변형하여, 전술한 실시예를 그대로 적용할 수 있다.

<240> 6) 움직임 벡터 예측값(MV_P)을 구하기 위하여 이용할 수 있는 시간 방향 대응 블록들과 뷰 방향 대응 블록들 사이의 우선순위는 현재 블록과 동일한 위치의 블록, 그 블록의 왼쪽에 위치한 블록, 위쪽에 위치한 블록, 오른쪽에 위치한 블록, 아래쪽에 위치한 블록, 좌상쪽에 위치한 블록, 우상쪽에 위치한 블록, 좌하쪽에 위치한 블록, 및 우하쪽에 위치한 블록의 순서를 가질 수 있다.

- <241> 본 발명의 실시예에 따른 움직임 벡터 예측값(MV_p)을 구하는 방법은 전술한 실시예에 한정되지 않으며, 여러 가지 다양한 방법이 가능하다는 것은 당업자에게 자명하다.
- <242> 움직임 벡터 예측값(MV_p)을 구하는 방법을 이용한 부호화 장치
- <243> 도 15에는 본 발명의 일 실시예에 따른 다시점 영상의 부호화 장치의 구성을 보여주는 블록도가 도시되어 있다. 도 15에 도시되어 있는 부호화 장치는 앞에서 설명한 움직임 벡터 예측값(MV_p)을 구하는 방법을 이용하는 다시점 영상의 부호화 장치이다.
- <244> 도 15를 참조하면, 부호화 장치는 움직임 예측부(110), 움직임 보상부(120), 변환 및 양자화부(130), 엔트로피 부호화부(135), 역양자화 및 역변환부(140), 슬라이스 단위 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)의 이용 여부 판단부(150, 이하 '이용 여부 판단부'라고 한다), 및 움직임 벡터 예측부(160)와 제1, 제2, 및 제3 가산기(102, 104, 106)를 포함하여 구성된다. 이하에서는 도 15를 참조하여, 상기 부호화 장치의 구성 및 동작에 대하여 구체적으로 설명한다.
- <245> 움직임 예측부(110)는, 인터 영상으로 부호화될 현재 영상에 관한 데이터가 부호화 장치로 입력되면, 블록 단위로 움직임 예측을 수행하여 현재 블록의 움직임 벡터(MV)를 생성한다. 상기 움직임 예측에서는 리스트0(list 0)의 참조 영상 및/또는 리스트1(list1)의 참조 영상을 참조한다. 다시점 영상의 부호화에서는 각 블록의 참조 영상이 시간 방향 참조 영상이거나 또는 뷰 방향 참조 영상일 수 있으므로, 리스트0(list0)의 참조 영상은 현재 영상보다 앞선 시간에서의 영상 및/또는 현재 영상이 속한 뷰 보다 앞쪽 뷰에서의 영상이고, 리스트1(list1)의 참조 영상은 현재 영상보다 늦은 시간에서의 영상 및/또는 현재 영상이 속한 뷰보다 뒤쪽 뷰에서의 영상일 수 있다. 따라서 상기 현재 블록의 움직임 벡터(MV)는 시간 방향의 영상을 참조하는 것이거나 또는 뷰 방향의 영상을 참조하는 것일 수 있다.
- <246> 움직임 보상부(120)는 움직임 예측부(110)에서 생성된 움직임 벡터(MV)와 움직임 예측부(110)에서 참조된 참조 영상을 이용하여 블록 단위로 움직임 보상된 예측 영상을 생성한다. 그리고 제1 가산기(102)는 움직임 보상부(120)로부터 입력되는 예측 영상과 현재 영상을 차분하여 블록 단위로 차분 영상을 생성한다. 제1 가산기(102)로부터 출력되는 차분 영상은 변환 및 양자화부(130)에서 변환 및 양자화 과정을 거치면, 변환 및 양자화부(130)는 양자화 계수를 출력한다. 그리고 양자화 계수는 엔트로피 부호화부(135)에서 엔트로피 부호화되며, 엔트로피 부호화된 영상 정보는 다른 부가 정보와 함께 비트스트림으로 만들어져서 복호화 장치로 전송되거나 저장 매체에 기록된다. 그리고 변환 및 양자화부(130)로부터 출력되는 양자화 계수는 역양자화 및 역변환부(140)에서 역양자화 및 역변환 과정을 거쳐서 차분 영상으로 복원되며, 복원된 차분 영상은 제2 가산기(104)에서 움직임 보상부(120)로부터 출력되는 예측 영상과 합쳐져서 재구성된 현재 영상을 생성한다. 이러한 움직임 보상부(120)에서 제2 가산기(104)까지의 일련의 과정은 기존의 영상 부호화 장치에서의 과정과 동일하므로, 여기에서 이에 대한 자세한 설명은 생략한다.
- <247> 그리고 이용 여부 판단부(150)에서는 현재 영상, 리스트0(list0)의 참조 영상 및 리스트1(list1)의 참조 영상의 데이터를 사용하여, 움직임 벡터 예측값(MV_p)을 구하는데 있어서 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부를 판단하고, 그 결과를 나타내는 정보를 부호화하여 출력한다. 본 실시예에서는 이러한 판단 과정이 슬라이스 단위로 진행된다. 그러나 본 발명의 다른 실시예에 의하면, 이러한 판단 과정을 영상 시퀀스 단위 또는 블록(예컨대 매크로블록) 단위로 수행할 수도 있다. 이용 여부 판단부(150)의 구성의 일례를 보여주는 블록도는 도 16에 도시되어 있다.
- <248> 도 16을 참조하면, 이용 여부 판단부(150)는 이용 가능 계산부(152) 및 이용 여부 정보의 부호화부(154)를 포함하여 구성된다.
- <249> 이용 가능 계산부(152)는 현재 영상, 리스트0(list0)의 참조 영상 및 리스트1(list1)의 참조 영상의 데이터를 사용하여, 움직임 벡터 예측값(MV_p)을 구하는데 있어서 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부를 계산하여 판단한다. 이러한 판단은 슬라이스 단위로 수행될 수 있으나, 여기에만 한정되는 것은 아니다. 그리고 이러한 판단을 수행하는 구체적인 과정은 앞에서 설명한 본 명세서에 기재되어 있는 어떠한 방법이라도 사용할 수 있다.
- <250> 그리고 이용 여부 정보의 부호화부(154)에서는 이용 가능 계산부(152)에서의 판단 결과를 부호화하여, 시간 방

향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부에 관한 슬라이스 단위의 정보를 생성하여 출력한다. 슬라이스 단위의 정보를 표현하는 방법은 앞에서 설명한 본 명세서에 기재되는 있는 어떠한 방법이라도 사용할 수 있다. 예를 들어, 상기 슬라이스 단위의 정보가 "1"이면 MV_T 또는 MV_V 를 이용하고, "0"이면 MV_T 및 MV_V 는 이용하지 않고, 주변 블록의 움직임 벡터(MV_N)만을 이용하는 것으로 표현할 수 있다. 그리고 이러한 슬라이스 단위의 정보는 슬라이스 헤더에 포함되어 복호화 장치로 전송되거나 또는 저장 매체에 저장될 수 있다.

<251> 그리고 움직임 벡터 예측부(160)에서는 리스트0(list0)의 참조 영상, 리스트1(list1)의 참조 영상, 및 재구성된 현재 영상의 정보와 이용 여부 선택부(150)로부터 출력되는 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부에 관한 슬라이스 단위의 정보를 이용하여 움직임 벡터 예측값(MV_P)을 생성하여 출력한다. 도 17에는 이러한 움직임 벡터 예측부(160)의 구성의 일례를 보여주는 블록도가 도시되어 있다.

<252> 도 17을 참조하면, 움직임 벡터 예측부(160)는 블록 선택부(162) 및 움직임 벡터 예측값(MV_P) 생성부(164)를 포함하여 구성된다.

<253> 블록 선택부(162)는 이용 여부 판단부(150)로부터 출력되는 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부에 관한 슬라이스 단위의 정보에 기초하여 움직임 벡터 예측값(MV_P)을 구하는데 있어서 이용할 수 있는 블록을 선택한다. 이러한 선택은 재구성된 현재 영상에서 현재 블록의 주변 블록인 블록 A, B, C, 및 D와 리스트0의 참조 영상에서 시간 방향 대응 블록 및/또는 뷰 방향 대응 블록인 블록 XF0, XF1, XF2, XF3, XF4, XF5, XF6, XF7, 및 XF8(여기서, X는 T 또는 V일 수 있다) 및 리스트1의 참조 영상에서 시간 방향 대응 블록 및/또는 뷰 방향 대응 블록인 블록 XB0, XB1, XB2, XB3, XB4, XB5, XB6, XB7, 및 XB8 중에서 이루어진다. 예를 들어, 상기 슬라이스 단위의 정보가 "1"이면 시간 방향 대응 블록 및/또는 뷰 방향 대응 블록도 선택하고, "0"이면 시간 방향 대응 블록과 뷰 방향 대응 블록은 선택하지 않고 주변 블록만 선택할 수 있다.

<254> 그리고 움직임 벡터 예측값(MV_P)에서는 상기 블록 선택부(162)에서 선택된 블록의 움직임 벡터들의 전부 또는 일부를 이용하여 움직임 벡터 예측값(MV_P)을 구하여 출력한다. 전술한 바와 같이, 움직임 벡터 예측값(MV_P)은 선택된 블록의 움직임 벡터들의 전부 또는 일부에 대한 중간값 연산을 통해서 구할 수 있다.

<255> 움직임 벡터 예측값(MV_P)을 구하는 방법을 이용한 복호화 장치

<256> 도 18에는 본 발명의 일 실시예에 따른 다시점 영상의 복호화 장치의 구성을 보여주는 블록도가 도시되어 있다. 도 18의 복호화 장치는 앞에서 설명한 움직임 벡터 예측값(MV_P)을 구하는 방법을 이용하는 다시점 영상의 복호화 장치로서, 도 15의 부호화 장치에 대응하는 복호화 장치이다.

<257> 도 18을 참조하면, 복호화 장치는 움직임 보상부(220), 엔트로피 복호화부(135), 역양자화 및 역변환부(240), 및 움직임 벡터 예측부(250)와 제1 및 제2 가산기(204, 206)를 포함하여 구성된다. 이하에서는 도 18을 참조하여, 상기 복호화 장치의 구성 및 동작에 대하여 구체적으로 설명한다.

<258> 엔트로피 복호화부(235)는 입력되는 비트스트림에 대한 엔트로피 복호화 처리를 수행하여 복원하고자 하는 현재 영상의 차분 영상에 대한 양자화 계수를 블록 단위로 생성한다. 그리고 역양자화 및 역변환부(240)는 상기 양자화 계수에 대한 역양자화 및 역변환 처리를 수행하여 상기 현재 영상의 차분 영상을 복원하여 출력하며, 복원된 차분 영상은 제1 가산기(204)로 입력된다.

<259> 한편 움직임 벡터 예측부(250)에서는, 도 15 및 도 17을 참조하여 설명한 부호화 장치에서와 마찬가지로, 리스트0의 참조 영상, 리스트1의 참조 영상, 및 재구성된 현재 영상의 정보로부터 움직임 벡터 예측값(MV_P)을 생성한다. 움직임 벡터 예측값(MV_P)을 구하는데 있어서, 리스트0의 참조 영상 및/또는 리스트1의 참조 영상을 이용할지 여부는, 부호화 장치로부터 전송되거나 저장 매체로부터 재생되어 입력되는 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용할지 여부에 관한 슬라이스 단위의 정보에 기초하여 결정된다. 움직임 벡터 예측부(250)의 구체적인 구성 및 동작은 전술한 부호화 장치의 움직임 벡터 예측

부(150)와 동일하므로, 여기에서는 상세한 설명은 생략한다. 즉, 도 17 및 그 관련 설명은 본 실시예에 따른 움직임 벡터 예측부(250)에도 동일하게 적용될 수 있다.

<260> 움직임 벡터 예측부(250)에서 생성된 움직임 벡터 예측값(MV_p)은 제2 가산기(206)로 입력되며, 제2 가산기(206)는 입력되는 움직임 벡터 차분값(MV_d)과 움직임 벡터 예측값(MV_p)을 합산하여 현재 블록의 움직임 벡터(MV)를 생성하여 출력한다. 그리고 움직임 보상부(220)는 리스트0의 참조 영상 및/또는 리스트1의 참조 영상의 정보와 움직임 벡터(MV)를 이용하여 예측 영상을 생성한다. 그리고 제1 가산기(204)는 움직임 보상부(220)로부터 입력되는 예측 영상과 역양자화 및 역변환부(240)로부터 입력되는 차분 영상을 합산하여 현재 영상을 복원한다.

발명의 효과

<261> 본 발명에 따른 다시점 비디오 코딩에서의 움직임 벡터의 예측 방법에서는 현재 블록의 주변 블록의 움직임 벡터(MV_N)와 더불어 현재 영상이 속한 뷰와 동일한 뷰에서 시간 방향 참조 영상의 시간 방향 대응 블록의 움직임 벡터(MV_T) 및/또는 현재 영상과 같은 시간대에서 뷰 방향 참조 영상의 뷰 방향 대응 블록의 움직임 벡터(MV_V)를 이용하여 움직임 벡터 예측값(MV_p)을 구한다. 따라서 본 발명에 의하면 다시점 비디오 코딩에서 보다 정확한 움직임 벡터 예측값(MV_p)을 구할 수 있으며, 그 결과 움직임 벡터 차분값(MV_d)의 정보량(엔트로피)을 감소시켜 예측 부호화 효율을 향상시킬 수가 있다. 즉, 본 발명에 의하면 다시점 비디오 코딩에서 여러 개의 카메라로부터 입력되는 영상간의 유사성을 효율적으로 활용함으로써, 예측 부호화 효율을 향상시키는 것이 가능하다.

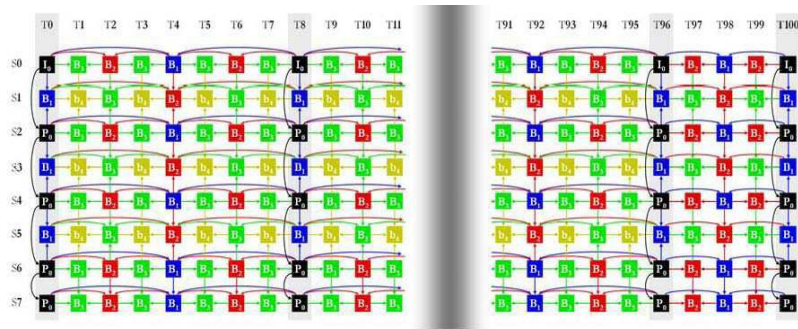
도면의 간단한 설명

- <1> 도 1은 다시점 비디오 코딩에서의 예측 구조의 일례를 보여주는 도면이다.
- <2> 도 2는 H.264/AVC에 규정되어 있는 움직임 벡터 예측값(MV_p)을 구하는 방법의 일례를 설명하기 위한 도면이다.
- <3> 도 3은 H.264/AVC에 규정되어 있는 움직임 벡터 예측값(MV_p)을 구하는 방법의 다른 예를 설명하기 위한 도면이다.
- <4> 도 4는 H.264/AVC에 규정되어 있는 움직임 벡터 예측값(MV_p)을 구하는 방법의 또 다른 예를 설명하기 위한 도면이다.
- <5> 도 5a 및 도 5b는 H.264/AVC에 규정되어 있는 움직임 벡터 예측값(MV_p)을 구하는 방법의 또 다른 예를 설명하기 위한 도면으로서, 도 5a는 현재 블록이 8×16 픽셀 크기를 갖는 경우이고 도 5b는 현재 블록이 16×8 픽셀 크기를 갖는 경우이다.
- <6> 도 6a, 도 6b 및 도 6c는 다시점 비디오 코딩의 표준화 과정에 있어서 테스트 영상 시퀀스의 하나인 영상 "Ballroom"을, 시간 방향의 영상 그룹의 크기를 12로 하고 H.264/AVC에 규정된 기존의 움직임 벡터 예측 방법을 이용하여 부호화하였을 경우에 대한 예로서, 도 6a는 전체 예측 구조의 일부분을 보여주는 것이고, 도 6b는 도 6a에 도시된 예측 구조에서 S1 뷰의 T0 시간에 해당하는 영상 B_1 의 일부 매크로블록에 대한 움직임 정보를 확대하여 표시한 것이고, 도 6c는 도 6a에 도시된 예측 구조에서 S1 뷰의 T1 시간에 해당하는 영상 b_4 의 일부 매크로블록에 대한 움직임 정보를 확대하여 표시한 것이다.
- <7> 도 7은 본 발명에 따라서 움직임 벡터 예측값(MV_p)을 구하는데 이용할 수 있는 블록의 위치를 보여주는 도면이다.
- <8> 도 8은 다시점 비디오 코딩에서 예측 구조의 다른 예를 보여주는 도면이다.
- <9> 도 9는 도 7과 도 8을 병합하여 도시한 도면이다.
- <10> 도 10은 현재 블록의 참조 영상이 뷰 방향 참조 영상인 경우에 움직임 벡터 예측값(MV_p)을 구하는 방법의 일례를 보여주는 흐름도이다.
- <11> 도 11은 현재 블록의 참조 영상이 시간 방향 참조 영상인 경우에 움직임 벡터 예측값(MV_p)을 구하는 방법의 일례를 보여주는 흐름도이다.

- <12> 도 12는 움직임 벡터 예측값(MV_p)을 구하는데 이용되는 블록이 여러 개의 파티션 블록으로 나뉘어 있을 경우, 움직임 벡터 예측값(MV_p)을 구하는데 실제로 사용하는 파티션 블록의 위치를 도식적으로 예시한 도면이다.
- <13> 도 13은 본 발명의 일 실시예에 따라서 N개 또는 그 보다 작은 개수의 움직임 벡터에 대한 중간값 연산을 통하여 움직임 벡터 예측값(MV_p)을 구하는 방법을 보여주는 흐름도이다.
- <14> 도 14는 본 발명의 다른 실시예에 따라서 움직임 벡터 예측값(MV_p)을 구하는 과정을 보여주는 흐름도이다.
- <15> 도 15는 본 발명의 일 실시예에 따른 움직임 벡터 예측 방법을 이용하는 부호화 장치의 구성을 보여주는 블록도이다.
- <16> 도 16은 도 15의 부호화 장치의 이용 여부 선택부(150)의 구성의 일례를 보여주는 블록도이다.
- <17> 도 17은 도 15의 부호화 장치의 움직임 벡터 예측부(160)의 구성의 일례를 보여주는 블록도이다.
- <18> 도 18은 본 발명의 일 실시예에 따른 움직임 벡터 예측 방법을 이용하는 복호화 장치의 구성을 보여주는 블록도이다.

도면

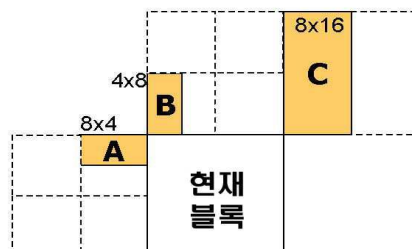
도면1



도면2



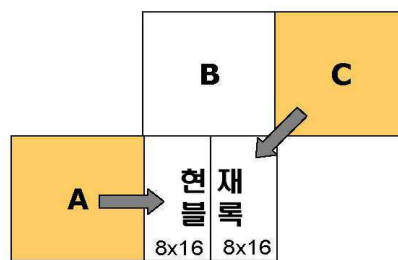
도면3



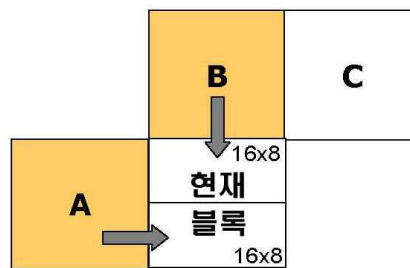
도면4



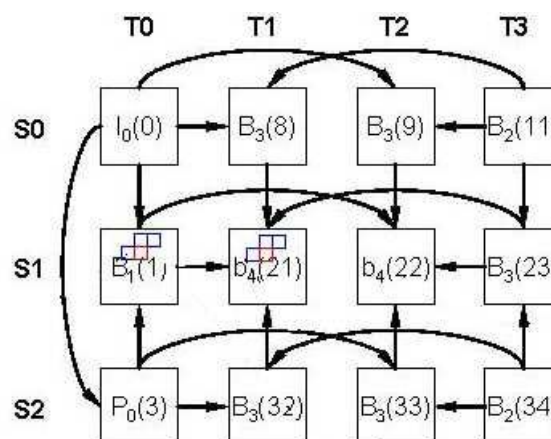
도면5a



도면5b



도면6a



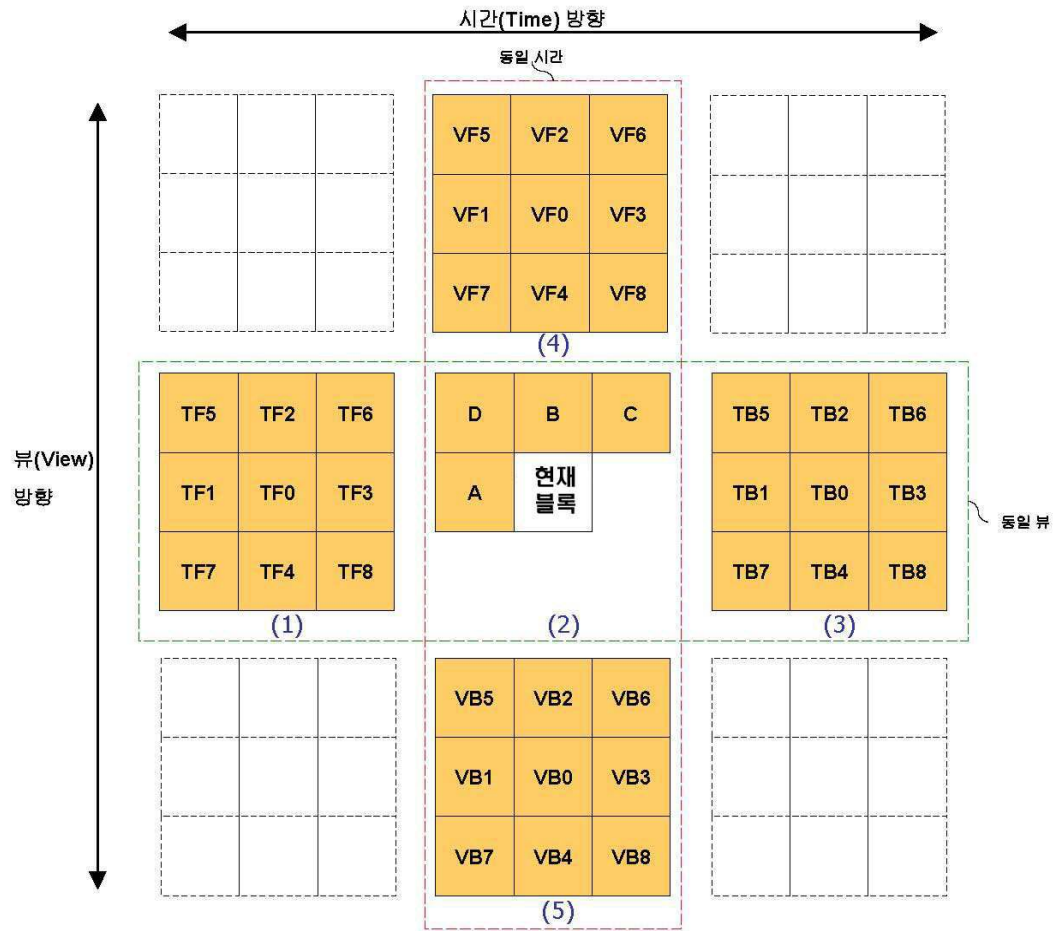
도면6b

확대 LIST_1(POC : 1, FrameNum : 2)					
	10	11	12	13	14
12	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A
13	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A
14	N/A N/A N/A N/A N/A N/A N/A	11.8, 2.5 0.0, 0.0 11.8, 2.5 0	17.8, 2.0 11.8, 2.5 6.0, -0.5 0	-16.3, 0.5 0.0, 0.0 -16.3, 0.5 2	-16.3, 0.5 -16.3, 0.5 0.0, 0.0 2
15	11.8, 2.0 11.8, 2.0 0.0, 0.0 0	18.3, 1.3 11.8, 2.0 6.5, -0.8 0	-16.3, 0.5 -16.3, 0.5 0.0, 0.0 2	-16.8, 1.0 -16.3, 0.5 -0.5, 0.5 2	-17.0, 1.0 -16.8, 0.5 -0.3, 0.5 2
16	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A	-16.8, 0.8 -16.8, 1.0 0.0, -0.3 2	-16.8, 1.0 -16.8, 1.0 0.0, 0.0 2

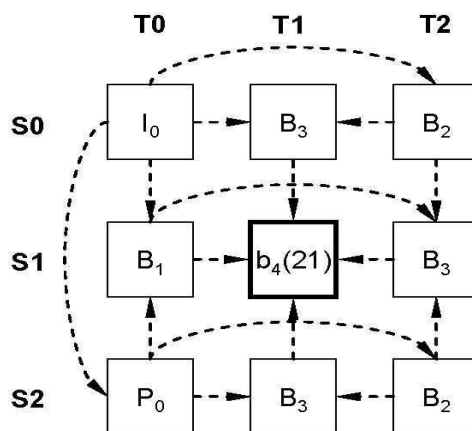
도면6c

확대 LIST_0(POC : 20, FrameNum : 36)					
	10	11	12	13	14
12	0.0, 0.0 0.0, 0.0 0.0, 0.0 1	0.0, 0.0 0.0, 0.0 0.0, 0.0 1	-0.3, 0.0 0.0, 0.0 -0.3, 0.0 1	0.0, 0.0 0.0, 0.0 0.0, 0.0 1	0.0, 0.0 0.0, 0.0 0.0, 0.0 1
13	0.0, 0.0 0.0, 0.0 0.0, 0.0 1	0.0, -0.3 0.0, 0.0 0.0, -0.3 1	0.0, 0.0 0.0, 0.0 0.0, 0.0 1	N/A N/A N/A N/A N/A N/A N/A	0.0, 0.0 0.0, 0.0 0.0, 0.0 1
14	0.0, 0.0 0.0, 0.0 0.0, 0.0 1	0.0, 0.0 0.0, 0.0 0.0, 0.0 1	18.3, 2.0 0.0, 0.0 18.3, 2.0 8	18.8, 1.8 18.3, 2.0 0.5, -0.3 8	18.8, 2.0 18.8, 1.8 0.0, 0.3 8
15	0.3, 0.0 0.0, 0.0 0.3, 0.0 1	N/A N/A N/A N/A N/A N/A N/A	N/A N/A N/A N/A N/A N/A N/A	19.0, 2.0 18.8, 1.8 0.3, 0.3 8	18.8, 1.8 18.8, 2.0 0.0, -0.3 8
16	0.0, 0.0 0.0, 0.0 0.0, 0.0 1	0.0, 0.0 0.0, 0.0 0.0, 0.0 1	0.0, 0.0 0.0, 0.0 0.0, 0.0 1	18.8, 2.0 18.8, 1.8 0.0, 0.3 8	18.8, 2.0 18.8, 2.0 0.0, 0.0 8

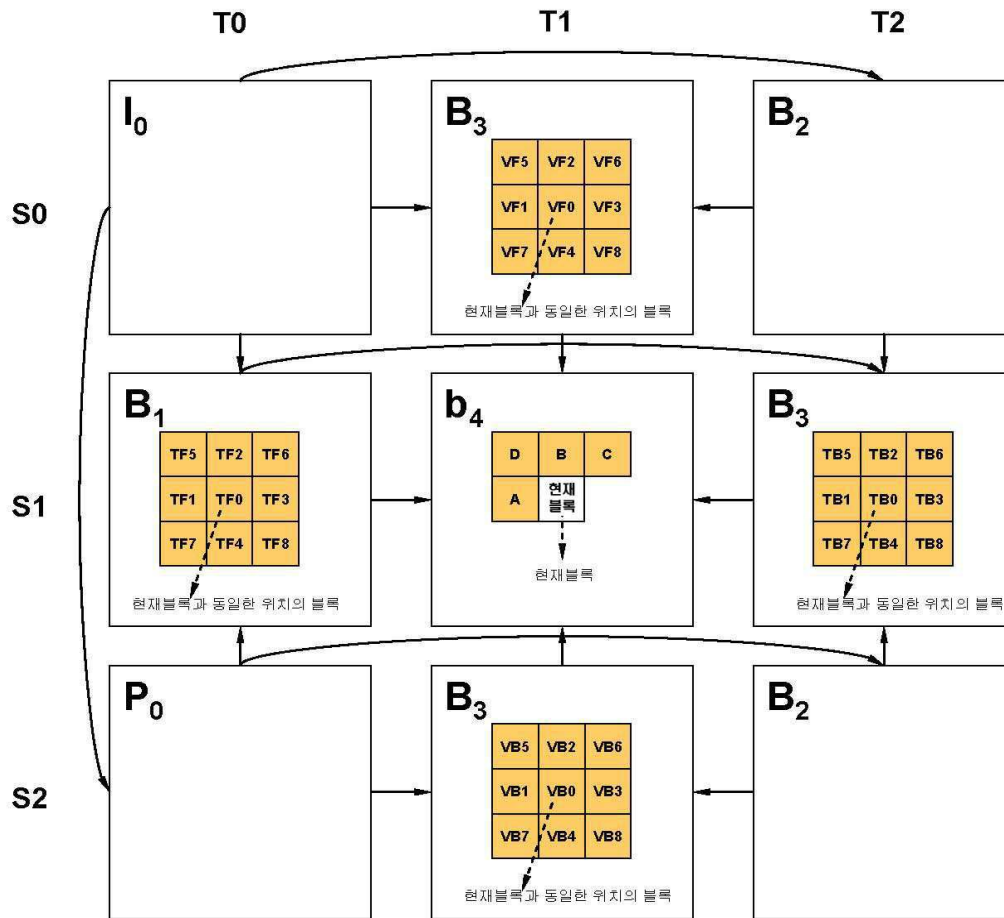
도면7



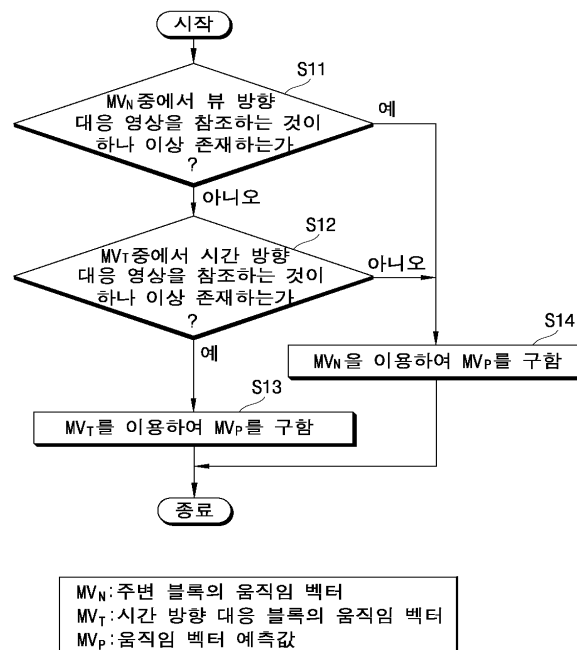
도면8



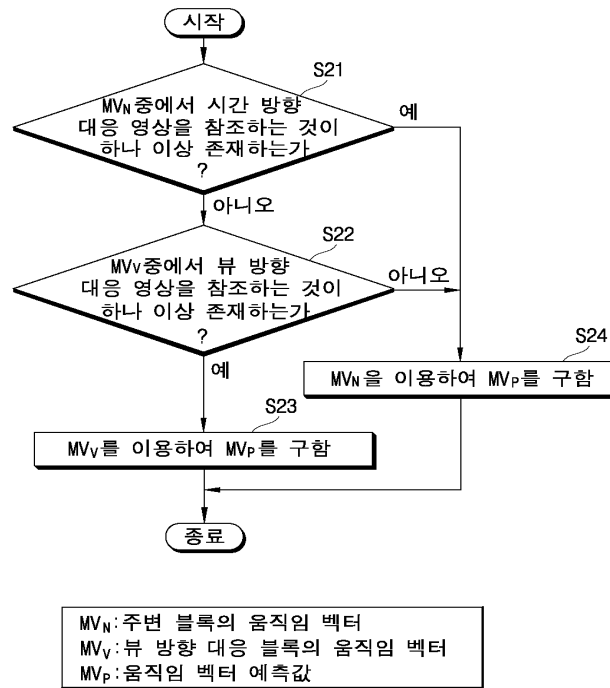
도면9



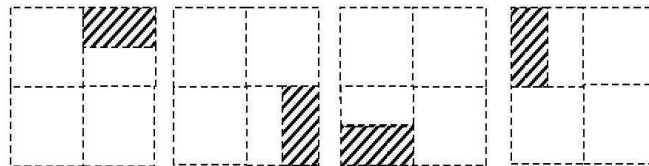
도면10



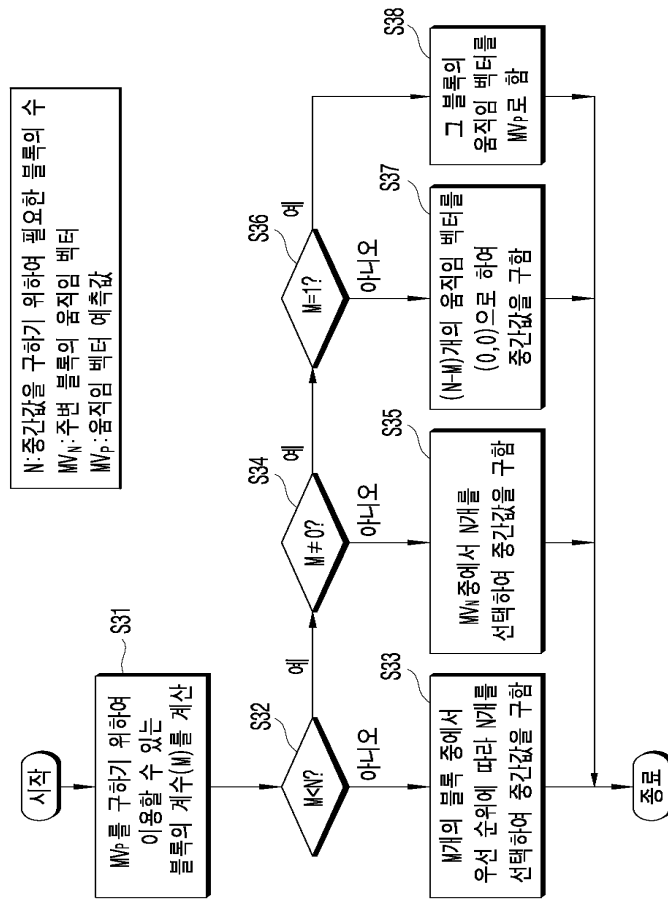
도면11



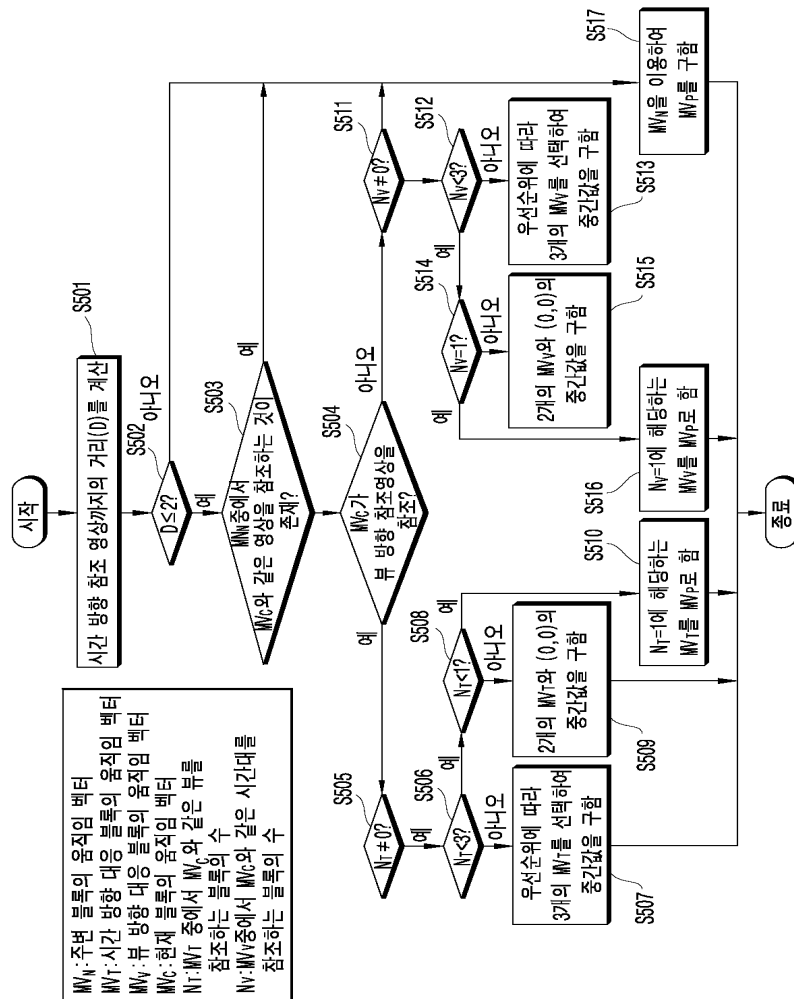
도면12



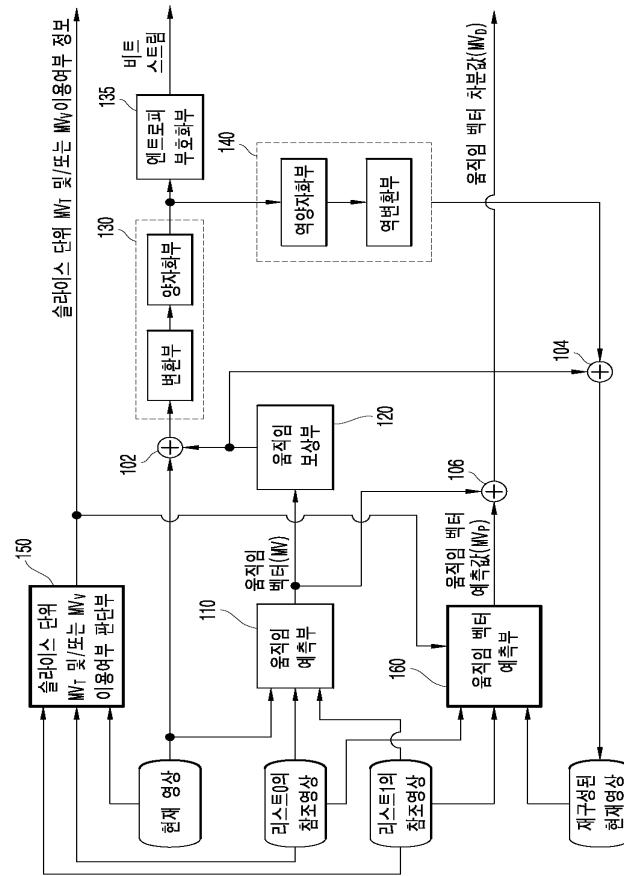
도면13



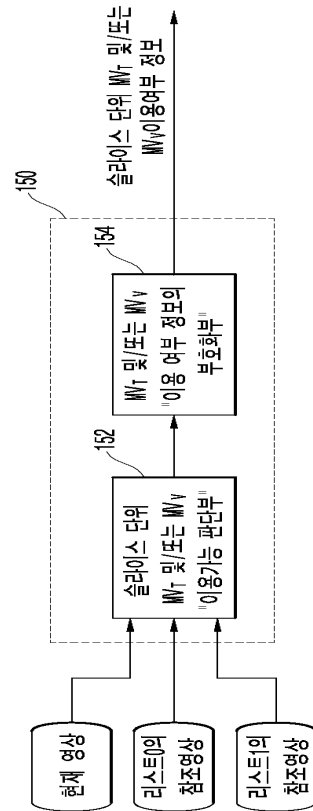
도면14



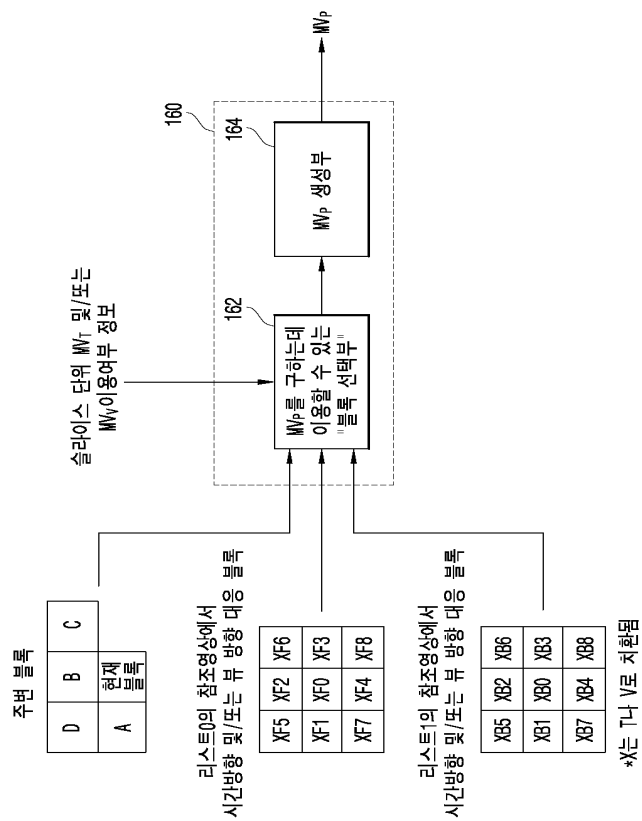
도면15



도면16



도면17



도면18

