

명세서

발명의 명칭: 움직임 정보를 획득하기 위한 방법

기술분야

- [1] 본 발명은 비디오 부호화 또는 복호화에 있어서 움직임 정보를 획득하기 위한 방법에 관한 것이다.

배경기술

- [2] 일반적으로 사용되는 비디오 부호화(encoding) 또는 복호화(decoding) 기술들은 모두 하이브리드 구조에 기초하고, 주요 부호화 도구(coding tool)들은 예측 부호화(prediction coding), 변환 부호화(transform coding) 및 엔트로피 부호화(entropy coding)를 포함한다. 예측 부호화는 각각 시간축 안의 비디오 신호들의 상관관계(correlation)와 공간축 안의 비디오 신호들의 상관관계를 제거하는 인터-프레임 예측 부호화(inter-frame prediction coding)와 인트라-프레임 예측 부호화(intra-frame prediction coding)를 포함한다.
- [3] 인터-프레임 예측 부호화는 부호화된 프레임의 이미지에 따라 현재 부호화된 프레임의 이미지를 예측할 수 있다. 그리고 예측 오차들(prediction residuals)은 변환되어 엔트로피 부호화된다. 현재 부호화된 프레임을 예측하기 위한 먼저 부호화된 프레임은 참조 프레임이라고 부른다. 참조 프레임 위의 예측 블록과 현재 부호화된 블록 간의 공간 위치 관계는 움직임 벡터로 복호화될 수 있다.
- [4] 예측 기술들의 발전으로, 이미지의 예측 오차는 감소하고 있다. 따라서, 비트 스트림 안에서 이미지의 예측 오차 정보의 비율은 감소하고, 이에 따라 움직임 정보의 비중은 증가하였다. 움직임 벡터의 효율적인 부호화는 비디오 부호화의 성능에 있어서 중요하다.
- [5] 비디오 부호화 또는 복호화에서 움직임 벡터를 처리하기 위한 두 가지 방법이 있다. 첫 번째 방법에서, 부호기는 움직임 추정을 통해 움직임 벡터를 획득한 후 움직임 벡터를 예측하고, 움직임 벡터와 움직임 벡터의 예측자 간의 움직임 벡터 차이를 비트 스트림(bit-stream)에 기록한다. 복호기는 같은 방법으로 움직임 벡터를 예측하고, 비트 스트림으로부터 읽은 움직임 벡터의 예측자와 움직임 벡터 차이에 따라 움직임 벡터를 재구성한다. 두 번째 방법은 부호기/복호기가 같은 방법으로 직접 움직임 벡터를 유도한다. 두 번째 방법으로 인해 비트 스트림 안에 움직임 벡터 차이를 전송할 필요가 없다. 그러나 어떠한 방법을 적용하여도, 움직임 정보는 유도된다. 유도된 움직임 정보는 움직임 벡터의 예측자, 움직임 벡터, 움직임 벡터의 예측자의 후보 또는 움직임 벡터의 후보가 될 수 있다.
- [6] H.264/Advanced Video Coding (AVC)에서, 움직임 벡터는 인터-프레임 예측 모드(inter-frame prediction mode)에서 중간값 예측 방법으로 예측된다. 중간값 예측 방법에서, 현재 부호화된 블록과 인접한 세 개의 부호화된 블록들은

선택되고, 세 개의 부호화된 블록들의 움직임 벡터들의 수평 성분들의 중간 값과 세 개의 부호화된 블록들의 움직임 벡터들의 수직 성분들의 중간 값은 각각 현재 부호화된 블록의 움직임 벡터의 예측자의 수평 성분과 수직 성분으로 채택된다. 시간적 다이렉트 모드(temporal direct mode)에서, 현재 부호화된 블록과 동일한 공간 위치를 가지는 참조 프레임의 블록의 움직임 벡터는 스케일링되고(scaled), 현재 부호화된 블록의 움직임 벡터로 채택된다.

- [7] Audio Video coding Standard (AVS)에서, 움직임 벡터는 인터-프레임 예측 모드에서 거리 중간값 방법으로 예측된다. 거리 중간값 방법에서 공간상 현재 부호화된 블록과 근접한 세 개의 부호화된 블록들은 선택되고, 세 개의 부호화된 블록들의 움직임 벡터들 중 두 움직임 벡터 사이의 거리는 계산되며, 계산된 거리들의 중간값에 대응되는 움직임 벡터가 현재 부호화된 블록의 움직임 벡터의 예측자로 채택된다. 시간적 다이렉트 모드에서 현재 부호화된 블록과 동일한 공간 위치를 가지는 참조 프레임의 블록의 움직임 벡터는 스케일링되고 현재 부호화된 블록의 움직임 벡터로 채택된다.
- [8] 최신 국제 비디오 부호화 표준인 High Efficiency Video Coding (HEVC)에서, 움직임 벡터 경쟁 기법(motion vector competition scheme)이 소개되었다. 움직임 벡터 경쟁 계획에서, 후보 움직임 벡터 세트는 규칙에 따라 유도되고, 적합한 후보 움직임 벡터가 후보 움직임 벡터 세트 중에서 선택되어, 현재 부호화된 움직임 벡터의 예측자로 채택된다. 그리고 선택된 움직임 벡터는 비트 스트림으로 전송된다. 후보 움직임 벡터 세트 공간적 후보 움직임 벡터들과 시간적 후보 움직임 벡터들을 포함할 수 있다. 공간적 후보 움직임 벡터들은 현재 부호화된 블록과 인접한 좌측 블록들과 상단 블록들의 움직임 벡터들로부터 특정 규칙과 우선권에 따라 선택될 것이다. 시간적 후보 움직임 벡터들은 현재 부호화된 블록과 공간적으로 인접하거나 같은 공간 위치를 가지는 참조 프레임의 블록들의 움직임 벡터들로부터 특정 규칙과 우선권에 따라 선택될 것이다.
- [9] 기존의 기술들에서 움직임 벡터는 오직 한 번 유도되거나 예측되고, 유도 또는 예측된 값은 최종 움직임 벡터 또는 예측자로 채택된다. 그러나 서로 공간적 또는 시간적으로 인접한 블록들의 움직임 벡터 차이들은 여전히 상관관계를 가지고 있고, 움직임 벡터 차이들 간의 상관관계는 기존의 비디오 부호화 또는 복호화 방법에서 이용되지 않는다. 예를 들어 중간값 또는 거리에 기초한 움직임 벡터 예측 또는 유도 방법에서, 오직 현재 부호화된 블록의 동일한 프레임에 위치한 블록들의 움직임 벡터들만이 고려되고, 따라서 오직 공간 영역의 움직임 벡터들의 상관관계만이 제거될 수 있다. 그러나 움직임 벡터 차이들은 여전히 시간 영역의 움직임 벡터들의 상관관계를 갖는다. H.264/AVC 및 AVS의 시간적 다이렉트 모드에서, 움직임 벡터 유도 방법은 오직 현재 부호화된 블록의 동일한 공간 위치에 위치한 참조 프레임들 위의 블록들의 움직임 벡터들만이 고려된다. 이 방법을 통해 유도된 움직임 벡터는 시간적 상관관계를 활용하지만, 동일한

프레임 안의 현재 부호화된 블록과 현재 부호화된 블록에 인접한 블록들과 간의 상관관계를 고려하지 않는다. 따라서, 유도된 움직임 벡터들은 최적이지 않다. HEVC의 경계 계획에 기초한 움직임 벡터 예측 및 유도 방법에서, 시간적 후보 움직임 벡터와 공간적 후보 움직임 벡터 모두 후보 움직임 벡터 세트 안에 포함되고 최적 움직임 벡터가 선택되더라도, 인접 블록들의 움직임 벡터 차이들은 여전히 상관관계를 가지고 있다.

발명의 상세한 설명

기술적 과제

- [10] 문 발명의 일 실시 예는, 비디오 이미지의 현재 블록 또는 서브 블록의 움직임 정보를 결정하는 방법을 포함하는 비디오 부호화 또는 복호화에 있어서 움직임 정보를 결정하는 방법은, 비디오 부호화 또는 복호화에 있어서 움직임 정보를 결정하는 방법은, 비디오 부호화 또는 복호화에 있어서 움직임 정보를 결정하는 방법이다.
- [11] 문 발명의 다른 실시 예는, 비디오 부호화 또는 복호화에 있어서 움직임 정보를 결정하기 위한 장치를 제공한다.

과제 해결 수단

- [12] 비디오 부호화 또는 복호화에 있어서 움직임 정보를 획득하기 위한 방법은, 비디오 이미지의 현재 블록 또는 서브 블록의 움직임 정보를 결정하는 과정을 포함한다. 이 과정은
- [13] 현재 블록 또는 서브 블록의 제 1 참조 블록 세트의 각 참조 움직임 벡터에 따라 상기 현재 블록 또는 서브 블록의 움직임 벡터의 제 1 예측자를 계산하는 단계,
- [14] 현재 블록 또는 서브 블록의 제 2 참조 블록 세트의 각 참조 움직임 벡터와 현재 블록 또는 서브 블록의 움직임 벡터 차이의 예측자 간의 제 1 움직임 벡터 차이를 예측하는 단계, 및

- [15] 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 예측자와 현재 블록 또는 서브 블록의 움직임 벡터의 제 1 예측자를 결정하는 단계를 포함할 수 있다.
- [16] 비디오 부호화 또는 복호화에서의 움직임 정보를 획득하기 위한 장치에 있어서,
- [17] 현재 블록 또는 서브 블록의 제 1 참조 블록 세트 안의 각 참조 움직임 벡터에 따라 현재 블록 또는 서브 블록의 움직임 벡터 차이를 예측하고, 상기 현재 블록 또는 서브 블록의 움직임 벡터 차이를 제 2 참조 블록 세트 안의 각 참조 움직임 벡터와 움직임 벡터 차이를 예측하는 단계를, 및

상기 참조 블록의 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이를 결정하는 일련부,

- [18] 상기 제 1 참조 블록 세트 안의 각 참조 블록의 상기 움직임 벡터에 따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자를 계산하고, 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자를 획득하기 위하여 상기 제 2 참조 블록 세트 안의 각 참조 블록의 상기 제 1 움직임 벡터 차이에 따라 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이를 예측하는 예측부, 및
- [19] 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자에 따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 제 2 예측자를 결정하는 계산부를 포함할 수 있다.
- [20] 상기 제 1 참조 블록 세트는 최초로 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터를 예측하거나 유도하기 위한 참조 블록들을 포함하고,
- [21] 상기 제 2 참조 블록 세트는 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이를 예측하기 위한 참조 블록들을 포함할 수 있다.

발명의 효과

- [22] 기존의 비디오 부호화 또는 복호화 방법에서, 움직임 벡터는 오직 한 번만 유도되거나 예측되었다. 그러나 공간적 또는 시간적으로 인접한 블록의 움직임 벡터 차이는 여전히 상관 관계를 가지고 있다. 따라서, 움직임 벡터 차이들의 상관관계를 완전히 활용하기 위해, 본 발명은 움직임 벡터를 2회 유도하거나 예측하여 움직임 벡터를 더 정확히 유도 또는 예측할 수 있다.

도면의 간단한 설명

- [23] 도 1은 본 발명의 실시 예에 따른 움직임 벡터를 구하는 방법을 설명하는 순서도이다.
- [24] 도 2는 본 발명에 따른 제 1 실시 예의 흐름도이다.
- [25] 도 3은 본 발명의 일 실시 예에 따라 현재 블록의 제 1 카테고리의 참조 블록을 나타낸 개략도이다.
- [26] 도 4는 본 발명에 따른 제 2 실시 예의 흐름도이다.
- [27] 도 5는 본 발명의 일 실시 예에 따라 현재 블록의 제 2 카테고리의 참조 블록을 나타낸 개략도이다.
- [28] 도 6은 본 발명에 따른 제 3 실시 예의 흐름도이다.
- [29] 도 7은 본 발명에 따른 제 4 실시 예의 흐름도이다.
- [30] 도 8은 본 발명에 따른 제 5 실시 예의 흐름도이다.
- [31] 도 9는 본 발명의 다른 실시 예에 따라 현재 블록의 제 1 카테고리의 참조 블록을 나타낸 개략도이다.
- [32] 도 10은 본 발명의 다른 실시 예에 따라 현재 블록의 제 2 카테고리의 참조 블록을 나타낸 개략도이다.
- [33] 도 11은 본 발명에 따른 제 6 실시 예의 흐름도이다.
- [34] 도 12는 본 발명의 다른 실시 예에 따라 현재 블록의 제 1 카테고리의 참조

블록을 나타낸 개략도이다.

[35] 도 13은 본 발명의 다른 실시 예에 따라 현재 블록의 제 2 카테고리의 참조 블록을 나타낸 개략도이다.

[36] 도 14는 본 발명의 실시 예에 따른 움직임 정보를 획득하기 위한 장치를 나타내는 개략도이다.

[37] 도 15는 본 발명의 실시 예에 따른 장치에서의 입력부의 구성을 나타내는 개략도이다.

[38] 도 16은 본 발명의 실시 예에 따른 장치에 있어서의 예측부 구성을 나타내는 개략도이다.

발명의 실시를 위한 최선의 형태

[39] 본 발명은 비디오 부호화 또는 복호화에 있어서 움직임 정보를 획득하기 위한 방법 및 장치를 제공한다. 상기 방법 및 장치로써 부호화 성능을 향상시킬 수 있다. 다음과 같이 본 발명의 해결책이 구현된다.

[40] 비디오 부호화 또는 복호화에 있어서 움직임 정보를 획득하기 위한 방법은 비디오 이미지의 현재 블록 또는 서브 블록의 움직임 정보를 결정하는 과정을 포함한다. 이 과정은

[41] 현재 블록 또는 서브 블록의 제 1 참조 블록 세트의 각 참조 블록의 움직임 벡터에 따라 상기 현재 블록 또는 서브 블록의 움직임 벡터의 제 1 예측자를 계산하는 단계,

[42] 현재 블록 또는 서브 블록의 제 2 참조 블록 세트의 각 참조 블록의 움직임 벡터와 현재 블록 또는 서브 블록의 제 2 참조 블록 세트의 참조 블록의 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이를 결정하고, 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 예측자를 획득하기 위하여 제 2 참조 블록 세트의 각 참조 블록의 제 1 움직임 벡터 차이에 따라 현재 블록 또는 서브 블록의 움직임 벡터와 현재 블록 또는 서브 블록의 움직임 벡터의 예측자 간의 제 1 움직임 벡터 차이를 예측하는 단계, 및

[43] 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 예측자와 현재 블록 또는 서브 블록의 움직임 벡터의 제 1 예측자에 따라 현재 블록 또는 서브 블록의 움직임 벡터의 제 2 예측자를 결정하는 단계를 포함할 수 있고,

[44] 상기 제 1 참조 블록 세트는 최초로 현재 블록 또는 서브 블록의 움직임 벡터를 예측하거나 유도하기 위한 참조 블록들을 포함하고,

[45] 상기 제 2 참조 블록 세트는 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이를 예측하기 위한 참조 블록들을 포함할 수 있다.

[46] 상기 현재 블록의 상기 제 1 참조 블록 세트는 현재 프레임 안의 상기 현재 블록에 인접한 부호화된 블록 또는 복호화된 블록을 포함하거나, 상기 현재 프레임의 참조 프레임 안의 공간적으로 상기 현재 블록에 인접하거나 겹치는 부호화된 블록 또는 복호화된 블록을 포함하고,

- [47] 상기 현재 블록의 상기 제2 참조 블록 세트는 상기 현재 블록에 인접한 상기 현재 프레임에서 부호화된 블록 또는 복호화된 블록을 포함하거나, 상기 현재 프레임의 상기 참조 프레임 안에 공간적으로 상기 현재 블록에 인접하거나 프레임에서 부호화된 블록 또는 복호화된 블록을 포함하거나, 공간적으로 상기 현재 서브 블록 또는 상기 현재 서브 블록의 상기 블록에 인접한 상기 현재 프레임에서 부호화된 블록 또는 복호화된 블록을 포함하거나, 공간적으로 상기 현재 서브 블록 또는 상기 현재 서브 블록의 상기 제2 참조 블록 세트는 상기 현재 서브 블록 또는 프레임에서 부호화된 블록 또는 복호화된 블록을 포함하고, 몇

[48] 상기 현재 서브 블록의 상기 제1 참조 블록 세트는 상기 현재 서브 블록 또는 상기 현재 서브 블록의 상기 블록에 인접한 상기 현재 프레임에서 부호화된 블록 또는 복호화된 블록을 포함하거나, 공간적으로 상기 현재 서브 블록 또는 상기 현재 서브 블록의 상기 블록에 인접하거나 겹치는 상기 현재 프레임의 상기 참조 프레임에서 부호화된 블록 또는 복호화된 블록을 포함하고, 몇

[49] 상기 현재 서브 블록의 상기 제2 참조 블록 세트는 상기 현재 서브 블록 또는 상기 현재 서브 블록의 상기 블록에 인접한 상기 현재 프레임에서 부호화된 블록 또는 복호화된 블록을 포함하거나, 공간적으로 상기 현재 서브 블록 또는 상기 현재 서브 블록의 상기 블록에 인접하거나 겹치는 상기 현재 프레임의 상기 참조 프레임에서 부호화된 블록 또는 복호화된 블록을 포함할 수 있다.

[50] 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제1 예측자를 계산하는 단계에 있어서,

[51] 부호기에 의하여, 상기 현재 블록 또는 서브 블록의 상기 제1 참조 블록 세트 안의 각 참조 블록의 상기 모션 벡터에 따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제1 예측자를 유도하고, 비트 스트림에 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제1 예측자와 상기 비트 스트림 안의 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제1 예측자의 차이로 정보를 제공하는 단계, 및

[52] 복호기에 의하여, 상기 현재 블록 또는 서브 블록의 상기 제1 참조 블록 세트 안의 각 참조 블록의 상기 모션 벡터에 따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제1 예측자를 유도하고, 비트 스트림에 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제1 예측자의 차이로 정보를 제공하는 단계에 있어서,

[53] 상기 제2 참조 블록 세트 안의 각 참조 블록의 상기 움직임 벡터와 상기 현재 블록 또는 서브 블록의 상기 제2 참조 블록 세트 안의 상기 참조 블록의 상기 움직임 벡터의 차이로 상기 제2 참조 블록 세트 안의 상기 참조 블록을 결정하는 단계에 있어서,

[54] 상기 현재 블록 또는 서브 블록의 상기 움직임 정보 결정된 때, 상기 제2 참조 블록 세트 안의 각 참조 블록의 상기 움직임 벡터와 상기 제2 참조 블록 세트 안의 상기 참조 블록의 차이로 상기 제1 예측자의 차이를 선택하는 단계, 및

[55] 상기 현재 블록 또는 서브 블록 세트 안의 각 참조 블록에

대하여, 상기 참조 블록의 움직임 정보가 결정된 후, 상기 참조 블록의 다음에 처리되는 블록 또는 서브 블록의 움직임 정보가 결정되기 전에, 상기 참조 블록의 상기 제 1 움직임 벡터 차이로 상기 참조 블록의 움직임 벡터와 상기 참조 블록의 상기 움직임 벡터의 제 1 예측자 간의 차이를 선택하고, 상기 참조 블록의 상기 제 1 움직임 벡터 차이를 저장하는 단계, 및

- [56] 상기 현재 블록 또는 서브 블록의 상기 움직임 정보가 결정되면, 상기 제 2 참조 블록 세트 안의 각 참조 블록의 저장된 제 1 움직임 벡터 차이를 획득하는 단계를 포함할 수 있다.
- [57] 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트의 참조 블록의 움직임 벡터 또는 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트의 상기 참조 블록의 상기 움직임 벡터의 예측자가 존재하지 않을 때, 상기 움직임 정보를 획득하기 위한 방법은,
- [58] 상기 참조 블록의 제 1 움직임 벡터 차이는 상기 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 상기 예측자의 계산에 사용되지 않을 때, 상기 참조 블록의 상기 제 1 움직임 벡터 차이가 존재하지 않는다고 판단하는 단계, 또는
- [59] 상기 참조 블록의 제 1 움직임 벡터 차이는 상기 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 상기 예측자의 계산에 사용될 때, 상기 참조 블록의 상기 제 1 움직임 벡터 차이를 영 벡터로 설정하는 단계를 포함할 수 있다.
- [60] 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트 안의 상기 참조 블록의 상기 제 1 움직임 벡터 차이가 존재하지 않을 때, 상기 움직임 정보를 획득하기 위한 방법은 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자를 영 벡터로 설정하는 단계를 포함하거나,
- [61] 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트 안의 모든 참조 블록들의 제 1 움직임 벡터 차이들이 존재하지 않을 때, 상기 움직임 정보를 획득하기 위한 방법은 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자를 영 벡터로 설정하는 단계를 포함할 수 있다.
- [62] 상기 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이를 예측하는 단계는,
- [63] 상기 제 2 참조 블록 세트의 상기 참조 블록들의 제 1 움직임 벡터 차이들의 수평 성분들 중 최소값을 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자의 수평 성분으로 채택하고, 상기 제 2 참조 블록 세트의 상기 참조 블록들의 제 1 움직임 벡터 차이들의 수직 성분들 중 최소값을 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자의 수직 성분으로 채택하는 단계, 또는
- [64] 상기 제 2 참조 블록 세트의 상기 참조 블록들의 제 1 움직임 벡터 차이들의 평균을 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자로 채택하는 단계, 또는
- [65] 상기 제 2 참조 블록 세트의 상기 참조 블록들의 제 1 움직임 벡터 차이들의 수평 성분들 중 중간값을 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임

- 벡터 사이의 수평 성분으로 채택하고, 상기 제 2 참조 블록 세트의 상기 참조 블록들의 수직 성분들 중 중간값을 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 사이의 상기 예측자의 수직 성분으로 채택하는 단계, 또는
- [66] 상기 제 2 참조 블록 세트의 상기 참조 블록들의 제 1 움직임 벡터 차이들 중 각 두 개 사이의 거리를 계산하고, 상기 거리에 따라 상기 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 상기 제 2 참조 블록 세트의 각 참조 블록의 상기 제 1 움직임 벡터 차이에 의하여, 상기 제 2 참조 블록 세트 안의 각 참조 블록의 상기 제 1 움직임 벡터 차이에 따라 상기 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이를 유도하고, 상기 현재 블록 또는 서브 블록의 제 1 움직임 벡터 사이의 수평 성분으로 채택하는 단계, 및 복호기에 의하여, 상기 제 2 참조 블록 세트 안의 각 참조 블록의 상기 제 1 움직임 벡터 차이에 따라 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자의 수직 성분으로 채택하는 단계, 및 상기 움직임 벡터의 상기 움직임 벡터의 상기 제 2 예측자를 유도하는 단계는,
- [68] 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 2 예측자를 상기 예측자의 가중합을 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 2 예측자로 채택하는 단계를 포함할 수 있다.
- [70] 상기 현재 블록 또는 서브 블록의 상기 제 1 참조 블록 세트 안의 상기 참조 블록들은 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트 안의 상기 참조 블록들과 동일하거나 다를 수 있다.
- [71] 블록 내에서 다른 서브 블록들에 대하여, 서브 블록의 제 1 참조 블록 세트 안의 참조 블록들은 또 다른 서브 블록 세트 안의 참조 블록들과 동일하거나 다를 수 있다.
- [72] 상기 제 1 참조 블록 세트 또는 상기 제 2 참조 블록 세트 안의 참조 블록의 움직임 정보가 서브 블록 안에서 결정될 때, 상기 참조 블록은 서브 블록일 수 있다.
- [73] 비디오 부호화 또는 복호화에서의 움직임 정보 획득하기 위한 장치에 있어서,
- [74] 현재 블록 또는 서브 블록의 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 정보를 획득하고, 상기 현재 블록 또는 서브 블록의 제 2 참조 블록 세트 안의 각

참조 블록과 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트 안의 상기 참조 블록의 상기 동작의 상기 제 1 예측자 간의 제 1 움직임 벡터 차이를 결정하는 일련부,

[75] 상기 제 1 참조 블록 세트 안의 각 참조 블록의 상기 움직임 벡터에 따라 상기 제 1 참조 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자를 계산하고, 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터 차이의 상기 예측자를 획득하기 위하여 상기 제 2 참조 블록 세트 안의 각 참조 블록의 상기 제 1 움직임 벡터 차이에 따라 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터를 예측하는 예측부, 및

[76] 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자에 따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터를 결정하는 계산부를 포함하고,

[77] 상기 제 1 참조 블록 세트는 최초로 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터를 예측하거나 유도하기 위한 참조 블록들을 포함하고,

[78] 상기 제 2 참조 블록 세트는 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이를 예측하기 위한 참조 블록들을 포함할 수 있다.

[79] 상기 해결책에서 얻을 수 있는 바와 같이, 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터는 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 제 1

블록들의 제 1 움직임 벡터 차이들에 따라 예측된다. 그 후에, 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 2 예측자는 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자와 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자에 따라 획득되고, 현재 블록의 움직임 벡터의 최종 예측자, 현재 블록의 움직임 벡터의 예측자의 후보, 현재 블록의 움직임 벡터 또는 현재 블록의 움직임 벡터의 후보로 채택될 수 있다. 상기 방법에 의해, 상기 참조 블록의 움직임 벡터 차이들과 현재 블록 또는 서브 블록 간의 상관관계는 현재 블록 / 서브 블록의 움직임 벡터의 차이를 추정하는 것에 충분히 활용될 수 있고, 또한 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 정확한 제 2 예측자를 구하기 위하여 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자를 결정할 수 있다. 이로 인하여 움직임 정보와 부호화 성능의 정확도가 크게 향상될 수 있다.

발명의 실시를 위한 형태

[80] 본 발명의 목적, 기술적 해결책 및 효과를 명확히 하기 위하여, 아래에서 본

발명은 첨부 도면을 참조하여 상세히 설명될 것이다.

[81] 배경 기술에서 언급한 바와 같이, 기존의 비디오 부호화 또는 복호화 방법에서,

목적임 벡터는 오직 한 번만 유도되거나 예측되었다. 그러나 공간적 또는

시간적으로 인접한 벡터의 공간 차이는 여전히 상관 관계를 가지고 있다. 따라서, 부호화 성능은 상기 상관 관계를 활용함으로써 개선될 수 있다. 목적임 벡터 차이들의 상관관계를 완전히 활용하기 위해, 본 발명은 도 1에서 도시되어 있는 것과 같이 목적임 벡터를 획득하기 위한 방법을 제공한다.

[82] 벡터 (101)에서 현재 벡터 또는 서버 벡터의 목적임 벡터의 제 1 예측자는 현재 벡터 또는 서버 벡터의 제 1 참조 벡터 세트 안의 각 참조 벡터에 따라 계산된다.

[83] 제 1 참조 벡터 세트는 최초로 현재 벡터 또는 서버 벡터의 목적임 벡터를 예측하거나 유도하기 위하여 참조 벡터들을 포함한다.

[84] 벡터 (102)에서 현재 벡터 또는 서버 벡터의 제 2 참조 벡터 세트 안의 각 참조 벡터의 목적임 벡터와 현재 벡터 또는 서버 벡터의 제 2 참조 벡터 세트 안의 참조 벡터의 목적임 벡터의 제 1 예측자 간의 제 1 목적임 벡터 차이는 현재 벡터 또는 서버

벡터의 제 1 목적임 벡터 차이를 예측하기 위하여 제 2 참조 벡터 세트 안의 각 참조 벡터의 제 1 목적임 벡터 차이에 따라 예측된다.

[85] 제 2 참조 벡터 세트는 현재 벡터 또는 서버 벡터의 제 1 목적임 벡터 차이를 예측하기 위하여 참조 벡터들을 포함한다.

[86] 벡터 (103)에서 현재 벡터 또는 서버 벡터의 제 1 목적임 벡터 차이의 예측자와 현재 벡터 또는 서버 벡터의 목적임 벡터의 제 1 예측자에 따라 현재 벡터 또는

서버 벡터의 목적임 벡터의 제 2 예측자는 결정된다.

[87] 그러므로, 목적임 정보를 획득하기 위한 방법의 흐름도는 종료된다. 위

흐름도에서 획득된 목적임 벡터의 제 2 예측자는 현재 벡터 또는 서버 벡터의 부호화 또는 복호화를 위해 사용될 수 있는 현재 벡터 또는 서버 벡터의 목적임

벡터의 예측자, 현재 벡터 또는 서버 벡터의 목적임 벡터의 예측자의 후보, 현재 벡터 또는 서버 벡터의 목적임 벡터 또는 서버 벡터의 목적임

벡터의 후보로 채택될 수 있다. 상기 흐름도에서 안 수 있는 바와 같이, 목적임 정보를 획득하기 위한 방법은 목적임 벡터의 보조적인 예측 또는 유도 방법으로

간주될 수 있다. 목적임 벡터의 예측 프로세스에서, 목적임 벡터의 예측 이후

최초로 생성되는 목적임 벡터 차이는 목적임 벡터 차이의 예측자를 획득하기 위하여 다시 예측될 수 있다. 그리고 목적임 벡터의 예측자는 상기 두 개의

예측들 (predictions)의 결과에 따라 획득될 수 있다. 목적임 벡터의 유도

프로세스에서, 목적임 벡터의 유도된 값이 먼저 획득되고, 목적임 벡터 차이의 유도된 값이 근접한 벡터들의 제 1 목적임 벡터 차이들의 정보에 따라 획득된다.

[88] 그리고 목적임 벡터의 최종 유도 벡터는 상기 두 개의 유도에 의해 따라

획득된다. 그렇게 함으로써 유도된 목적임 벡터의 정확성이 향상될 수 있다.

최종로 예측 프로세스를 통해 획득된 현재 벡터 또는 서버 벡터의 목적임

벡터의 예측자와 최초로 유도 프로세스를 통해 획득된 현재 블록 또는 서브 블록의 움직임 벡터의 유도된 값은 움직임 벡터의 제 1 예측자라고 칭한다. 본 발명에서 두 번의 예측들 (predictions) 또는 두 번의 유도들 (derivations)의 결과에 따른 획득된 움직임 벡터의 추정 결과는 움직임 벡터의 제 2 예측자라고 칭한다. 비트 스트림 안의 유도 정보 전송 방법은 현재 블록 또는 서브 블록의 움직임 벡터의 제 1 예측자 및/또는 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 예측자의 계산에 적용될 수 있다. 이 방법에서 부호기)는 움직임 벡터의 제 1 예측자의 유도 정보 및/또는 제 1 움직임 벡터 차이의 예측자의 유도 정보를 비트 스트림에 기록한다. 그리고 복호기는 비트 스트림의 유도 정보를 따라 움직임 벡터의 제 1 예측자 및/또는 제 1 움직임 벡터 차이의 예측자를 계산한다.

- [89] 현재 프레임(current frame)과 현재 블록과 공간상 근접하거나 겹치는 참조 프레임(reference frame)의 블록들은 모두 본 발명의 두 번의 예측 또는 유도 프로세스들에 사용되거나, 각각 본 발명의 두 예측 또는 유도 프로세스들에 사용될 수 있다. 따라서 다른 블록들의 움직임 벡터들의 공간적, 시간적 상관관계는 최대한 제거될 수 있고, 움직임 벡터들의 부호화 효율은 증가할 수 있다.
- [90] 블록이 여러 개의 서브 블록으로 분리되면, 움직임 정보를 획득하기 위한 방법은 각 서브 블록의 움직임 정보를 획득하기 위하여 각 서브 블록에 적용된다. 블록과 서브 블록 안의 움직임 정보를 획득하기 위한 방법은 본 발명의 실시 예들을 따라 구현될 수 있다. 본 발명의 솔루션은 실시 예를 참고하여 설명된다.
- [91] 제 1, 2, 3 및 4 실시 예들에서, 움직임 정보는 현재 블록에 대하여 결정된다. 그리고 현재 블록의 움직임 정보를 유도하는 흐름도는 각 실시 예에 제공된다. 제 5 및 6 실시 예들에서, 현재 블록이 여러 개의 서브 블록들로 분리된다. 그리고 움직임 정보는 각 서브 블록에 대하여 결정된다. 그리고 서브 블록들의 움직임 정보를 유도하는 흐름도는 각 실시 예에 제공된다.
- [92] 제 1 실시 예는 다음과 같이 설명된다.
- [93] 도 2는 본 발명에 따른 제 1 실시 예의 흐름도이다. 도 3은 본 발명의 일 실시 예에 따른 현재 블록의 제 1 카테고리의 참조 블록들을 설명하는 개략도이다. 도 2에서 볼 수 있듯이 제 1 실시 예에 따라 움직임 정보를 획득하는 방법은 다음 블록들을 포함한다.
- [94] 블록 (201)에서 현재 블록의 제 1 참조 블록 세트는 결정된다.
- [95] 본 발명에서, 제 1 참조 블록 세트와 제 2 참조 블록 세트는 적어도 제 1 카테고리의 참조 블록과 제 2 카테고리의 참조 블록 중 하나를 포함한다. 제 1 카테고리의 참조 블록들은 현재 블록과 인접한 현재 프레임의 부호화 또는 복호화된 블록이다. 제 2 카테고리의 참조 블록들은 공간적으로 현재 블록들과 겹치거나 인접한 참조 프레임의 부호화 또는 복호화된 블록이다.
- [96] 이 실시 예에서 블록(201)은 다음과 같이 구현될 수 있다. 제 1 참조 블록 세트는

오직 제 1 카테고리의 참조 블록들을 포함한다. 도 3에서 도시되어 있듯이, 현재 블록의 제 1 카테고리의 참조 블록들은 AL1, AL2, AU1, AU2, AU3, AC1 및 AC2을 포함한다. 이 실시 예에서 현재 블록의 제 1 참조 블록 세트 안의 참조 블록들은 현재 블록의 좌측 상단 꼭지점 위에 위치한 인접 블록 AU1, 현재 블록의 좌측 상단 꼭지점 좌측에 위치한 인접 블록 AL2 및 현재 블록의 좌측 상단 꼭지점의 좌측 상단에 위치한 인접 블록 AC2으로 구성될 수 있다.

- [97] 블록(202)에서 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터는 획득된다.
- [98] 블록(202)에서 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터가 획득될 때, 만약 제 1 참조 블록 세트 안의 특정 참조 블록의 움직임 벡터가 존재하지 않으면, 참조 블록의 움직임 벡터가 존재하지 않는다고 표시하는 표시가 리턴(return)된다.
- [99] 블록 (203)에서 블록(202)에서 획득한 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터에 따라 현재 블록의 움직임 벡터의 제 1 예측자는 계산된다.
- [100] 블록 (203)에서 기존의 부호화 또는 복호화 방법에서 사용되던 현재 블록의 움직임 벡터를 예측하고 유도하기 위한 동일한 방법에 따라 현재 블록의 움직임 벡터의 제 1 예측자는 계산될 수 있다.
- [101] 블록 (203)은 다음과 같이 구현될 수 있다. 블록 (202)에서 획득된 움직임 벡터들의 수평 성분들의 평균 값과 블록 (202)에서 획득된 움직임 벡터들의 수직 성분들의 평균 값이 각각 계산되고, 현재 블록의 움직임 벡터의 예측자의 수평 성분과 현재 블록의 움직임 벡터의 예측자의 수직 성분으로 채택한다. 만약 제 1 참조 블록 세트 안의 특정 참조 블록의 움직임 벡터가 존재하지 않을 경우, 존재하지 않는 움직임 벡터는 영(zero) 벡터로 설정된다.
- [102] 더 나아가, 제 1 참조 블록 세트 안의 참조 블록이 위치한 프레임과 제 1 참조 블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리가 현재 블록이 위치한 프레임과 현재 블록이 위치한 프레임의 참조 프레임 간의 거리와 다를 수 있다. 이에 따라, 부호화 또는 복호화 성능을 향상시키기 위하여, 제 1 참조 블록 세트 안의 참조 블록의 움직임 벡터는 블록 (203)에서 현재 블록의 움직임 벡터의 제 1 예측자가 결정되기 전에 제 1 참조 블록 세트 안의 참조 블록이 위치한 프레임과 제 1 참조 블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리와 현재 블록이 위치한 프레임과 현재 블록이 위치한 프레임의 참조 프레임 간의 거리에 따라 스케일링될 수 있다. 그리고 현재 블록의 움직임 벡터의 제 1 예측자는 스케일링된 움직임 벡터에 따라 계산될 수 있다. 그렇게 함으로써 제 1 예측자의 정확도가 향상될 수 있다.
- [103] 블록 (204)에서 현재 블록의 제 2 참조 블록 세트는 결정된다.
- [104] 블록 (204)는 다음과 같이 구현될 수 있다. 제 2 참조 블록 세트는 오직 제 1 카테고리의 참조 블록들을 포함한다. 도 3에서 볼 수 있듯이, 현재 블록의 제 1 카테고리의 참조 블록들은 AL1, AL2, AU1, AU2, AU3, AC1 및 AC2을 포함한다.

이 실시 예에서 현재 블록의 제 2 참조 블록 세트 안의 참조 블록들은 현재 블록의 좌측 상단 꼭지점 위에 위치한 인접 블록 AU1, 현재 블록의 좌측 상단 꼭지점 좌측에 위치한 인접 블록 AL2 및 현재 블록의 좌측 상단 꼭지점의 좌측 상단에 위치한 인접 블록 AC2으로 구성될 수 있다.

[105] 블록 (205)에서 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터의 제 1 예측자는 획득된다. 그리고 참조 블록의 움직임 벡터와 참조 블록의 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이가 계산된다.

[106] 블록 (205)에서 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터의 제 1 예측자가 획득될 때, 만약 각 참조 블록의 움직임 벡터 또는 각 참조 블록의 움직임 벡터의 제 1 예측자가 존재하지 않을 때, 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않는다고 표시하는 표시가 리턴(return)된다.

[107] 제 2 참조 블록 세트 안의 참조 블록들은 모두 부호화된 블록들이다. 그러므로 제 2 참조 블록 세트 안의 참조 블록들의 움직임 벡터들과 제 2 참조 블록 세트 안의 참조 블록들의 움직임 벡터들의 제 1 예측자들은 생성된다. 여기서 각 참조 블록의 움직임 벡터와 참조 블록의 움직임 벡터의 제 1 예측자는 직접적으로 획득된다. 그리고 참조 블록의 제 1 움직임 벡터 차이를 획득하기 위하여 참조 블록의 움직임 벡터의 제 1 예측자는 참조 블록의 움직임 벡터에서 제(除)한다.

[108] 블록 (203)과 유사하게, 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임과 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리가 현재 블록이 위치한 프레임과 현재 블록이 위치한 프레임의 참조 프레임 간의 거리와 다를 수 있다. 이에 따라, 부호화 또는 복호화 성능을 향상시키기 위하여, 블록 (205)에서 제 2 참조 블록 세트 안의 참조 블록의 움직임 벡터와 제 2 참조 블록 세트 안의 참조 블록의 움직임 벡터의 제 1 예측자는 제 2 참조 블록 세트 안의 참조 블록의 제 1 움직임 벡터 차이가 계산되기 전에 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임과 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리와 현재 블록이 위치한 프레임과 현재 블록이 위치한 프레임의 참조 프레임 간의 거리에 따라 스케일링될 수 있다. 그렇게 함으로써 제 2 참조 블록 세트 안의 참조 블록의 제 1 움직임 벡터 차이의 정확도가 향상될 수 있다.

[109] 도 2를 참조하여 현재 블록과 참조 블록 A의 움직임 정보가 유도된 대로 제 2 참조 블록 세트 안의 참조 블록 A가 정해질 때, 만약 참조 블록 A의 제 1 참조 블록 세트 안의 참조 블록의 움직임 벡터가 블록 203에서 스케일링되지 않은 경우, 참조 블록 A의 움직임 벡터의 제 1 예측자는 제 1 참조 블록 세트 안의 참조 블록이 위치한 프레임과 제 1 참조 블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리와 현재 블록이 위치한 프레임과 현재 블록이 위치한 프레임의 참조 프레임 간의 거리에 따라 스케일링될 수 있다. 또는, 도 2를

참조하여 현재 블록과 참조 블록 A의 움직임 정보가 유도된 대로 제 2 참조 블록 세트 안의 참조 블록 A가 정해진 때, 만약 참조 블록 A의 제 1 참조 블록 세트 (즉, 축적비(scale ratio))가 참조 블록 A가 위치한 프레임과 참조 블록 A가 위치한 프레임의 참조 프레임 간의 거리와 참조 블록 B가 위치한 프레임과 참조 블록 B가 위치한 프레임의 참조 프레임 간의 거리를 완전히

보장하지 못하는 경우, 거리 차이를 완전히 보상하기 위하여 참조 블록 A의 움직임 벡터의 제 1 예측자는 스케일링될 수 있다. 그리고 위 문단의 스케일링 프로세스와 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이의 계산은 스케일링된 움직임 벡터들에 적용될 수 있다.

[110] 블록 (206)에서 현재 블록의 제 1 움직임 벡터 차이의 예측자는 블록 (205)에서 계산된 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이에 따라

계산된다.

[111] 현재 블록의 제 1 움직임 벡터 차이는 제 2 참조 블록 세트 안의 참조 블록의 제 1 움직임 벡터 차이에 따라 예측된다. 예측 방법은 필요 조건들에 따라 선택될 수 있다. 예를 들어, 참조 블록들의 제 1 움직임 벡터 차이들의 가중 평균을 내는 것, 참조 블록들의 제 1 움직임 벡터 차이들의 수평 성분의 중간 값과 참조 블록들의 제 1 움직임 벡터 차이들의 수직 성분의 중간 값을 각각 선택하는 것, 또는 기존의 움직임 벡터 예측 방법들이 사용될 수 있다.

[112] 이 실시 예에서, 블록 (206)은 다음과 같이 구현될 수 있다. 블록 (205)에서 획득된 제 1 움직임 벡터 차이들의 최소 값은 현재 블록의 제 1 움직임 벡터의 예측자로 채택될 수 있다. 만약 제 2 참조 블록 세트 안의 특정 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않는다면, 그 참조 블록의 제 1 움직임 벡터 차이는 계산 프로세스에서 고려되지 않는다. 만약 제 2 참조 블록 세트 안의 모든 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않는다면, 현재 블록의 제 1 움직임 벡터 차이의 예측자는 영 벡터로 설정될 수 있다.

[113] 블록(203)과 유사하게, 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임과 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리와 현재 블록이 위치한 프레임과 현재 블록이 위치한 프레임 간의 거리에 제 1 예측자가 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임과 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리와 제 1 예측자가 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임과 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리를 가질 수 있다. 이에 따라, 만약 블록 (205)에서 제 2 참조 블록 세트 안의

블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리와 현재 블록이 위치한 프레임과 현재 블록이 위치한 프레임의 참조 프레임 간의 거리에 제 1 예측자가 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임과 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리와 제 2 참조 블록 세트 안의 참조 블록이 위치한 프레임의 참조 프레임 간의 거리와

- [illegible]

예측자의 후보, 현재 블록 또는 서브 블록의 움직임 벡터 또는 현재 블록의 움직임 벡터의 후보로 채택될 수 있다.

[118] 그리고 제 1 실시 예에서 제공하는 방법 흐름도는 종료된다.

[119] 제 2 실시 예는 다음과 같이 설명된다.

[120] 도 4는 본 발명에 따른 제 2 실시 예의 흐름도이다. 도 3는 본 발명의 일 실시 예에 따른 현재 블록의 제 1 카테고리의 참조 블록들을 설명하는 개략도이다. 도 5는 본 발명의 일 실시 예에 따른 현재 블록의 제 2 카테고리의 참조 블록들을 설명하는 개략도이다. 도 4에서 볼 수 있듯이 움직임 정보를 획득하는 방법은 다음과 같이 구현된다.

[121] 블록 (401)에서 현재 블록의 제 1 참조 블록 세트는 결정된다.

[122] 이 실시 예에서 블록 (401)은 다음과 같이 구현될 수 있다. 제 1 참조 블록 세트는 오직 제 1 카테고리의 참조 블록들을 포함한다. 도 3에서 도시되어 있듯이, 현재 블록의 제 1 카테고리의 참조 블록들은 AL1, AL2, AU1, AU2, AU3, AC1 및 AC2을 포함한다. 이 실시 예에서 현재 블록의 제 1 참조 블록 세트 안의 참조 블록들은 현재 블록의 좌측 상단 꼭지점 위에 위치한 인접 블록 AU1, 현재 블록의 좌측 상단 꼭지점 좌측에 위치한 인접 블록 AL2, 현재 블록의 좌측 상단 꼭지점의 좌측 상단에 위치한 인접 블록 AC2 및 현재 블록의 우측 상단 꼭지점의 우측 상단에 위치한 인접 블록 AU3으로 구성될 수 있다.

[123] 블록(402)에서 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터는 획득된다.

[124] 블록(402)에서 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터가 획득된 때, 만약 제 1 참조 블록 세트 안의 특정 참조 블록의 움직임 벡터가 존재하지 않으면, 참조 블록의 움직임 벡터가 존재하지 않는다고 표시하는 표시가 리턴된다.

[125] 움직임 벡터를 리턴하는 기존의 방법은 적용될 수 있다. 움직임 벡터를 리턴하는 방법은 다음과 같이 구현될 수 있다. 만약 블록 AU3가 움직임 벡터를 가진다면, 블록 AU3의 움직임 벡터는 리턴되어 mv3로 복호화되지만, 블록 AC2의 움직임 벡터는 리턴되지 않는다. 만약 블록 AU3는 움직임 벡터가 없고 대신 블록 AC2가 움직임 벡터를 가진다면, 블록 AC2의 움직임 벡터는 리턴되어 mv3으로 복호화된다. 만약 블록 AU3과 블록 AC2가 모두 움직임 벡터가 없다면, 영 벡터가 리턴되며 mv3로 복호화된다. 만약 블록 AU1가 움직임 벡터를 가진다면, 블록 AU1의 움직임 벡터는 리턴되어 mv1로 복호화된다. 만약 블록 AU2가 움직임 벡터가 없다면, 영 벡터는 리턴되어 mv2로 복호화된다. 만약 블록 AU1이 움직임 벡터가 없다면, 영 벡터는 리턴되어 mv1로 복호화된다. 만약 블록 AU2이 움직임 벡터가 없다면, 영 벡터는 리턴되어 mv2로 복호화된다. 만약 특정 블록이 존재하지 않는다면, 그 블록은 모션 벡터가 없는 것으로 간주된다. 여기서 설명한 프로세스 방법은 본 발명의 다른 실시 예에도 적용될 수 있다.

[127]

비율에 따라 계산될 수 있다.

m_{3-y} 으로 복호화된다. 만약 m_{v1-x} 와 m_{v2-x} 가 같은 기호(sign)를 가진다면, m_{v1-x} 와 m_{v2-x} 의 평균은 현재 블록의 움직임 벡터의 제 1 예측자의 수평 성분으로 채택된다. 만약 m_{v1-x} 와 m_{v2-x} 가 다른 기호를 가지고, m_{v1-x} 와

m_{v3_x} 가 같은 기호를 가진다면, m_{v1_x} 와 m_{v3_x} 의 평균이 현재 블록의 움직임 벡터의 제 1 예측자로 채택된다. 만약 m_{v1_x} 와 m_{v2_x} 가 다른 기호를 가지고, m_{v2_x} 와 m_{v3_x} 가 같은 기호를 가진다면, m_{v2_x} 와 m_{v3_x} 의

평면이 원점을 통과하는 동적임베딩의 제 1 예측자의 수적 성분으로 채택된다. 만약 m_{v1-y} 와 m_{v2-y} 가 같은 기호를 가진다면, m_{v1-y} 와 m_{v2-y} 의 평균은 현재 블록의 동적임베딩의 제 1 예측자의 수적 성분으로 채택된다. 만약 m_{v1-y} 와 m_{v2-y} 가 예측자의 수적 성분으로 채택된다, 만약 m_{v1-y} 와 m_{v2-y} 가

다른 기호를 가지고, m^1y 와 m^3y 가 같은 기호를 가진다면, m^1y 와 m^3y 의 평균이 현재 블록의 동적임 벡터의 제 1 예측자의 수직 성분으로 채택된다. 만약 m^1y 와 m^2y 가 다른 기호를 가지고, m^2y 와 m^3y 가 같은 기호를 가진다면, m^2y 와 m^3y 의 평균이 현재 블록의 동적임 벡터의 제 1 예측자의 수직

성공으로 채택된다. 만약 특정 참조 블록의 움직임 벡터와 제 1 예측자가 존재하지 않을 경우, 참조 블록의 움직임 벡터는 영 벡터로 설정된다.

[129] 이 사람에 대하여는 현제 불독의 몸조이뿐 아니라 현제 불독의 몸조이를 빼앗아 제 1

예측자가 결정된 후, 현재 블록의 제 1 동적 임베딩 차이가 계산될 수 있다. 이 방법에서 현재 블록이 다음으로 진행되는 블록의 제 2 참조 블록 세트 안의 참조 블록으로 채택되는 때, 참조 블록의 제 1 동적 임베딩 차이가 바로 회수될 수

블록으로 채택되는 때, 창조 블록의 제 1 움직임 벡터 차이는 바로 획득될 수 있다.

[130] 제 1 원시 예와 유사하게, 여기에 설명되지는 않았지만 블록 (403)에서 움직임 벡터는 스케일링된다.

[131] 법률 (404)에서 현재 법률의 제 2 장 조 법률 세트는 결정된다.

[132] 클록(404)은 다음과 같이 구현될 수 있다. 제 2 참조 클록 세트는 오직 제 1

카테고리의 참조 블록들과 제 2 카테고리 참조 블록들을 포함한다. 도 3에서 볼 수 있듯이, 현재 블록의 제 1 카테고리 참조 블록들은 AL1, AL2, AU1, AU2, AU3, AC1 및 AC2를 포함한다. 이 실시 예에서 현재 블록의 제 2 참조 블록 세트 안의 제 1 카테고리 참조 블록들은 현재 블록의 참조 수단 꼭지점 위에 위치한 인접 블록 AU1, 현재 블록의 참조 수단 꼭지점 좌측에 위치한 인접 블록 AL2 및 현재 블록의 참조 수단 꼭지점의 좌측 상단에 위치한 인접 블록 AC2으로 구성될

수 있다. 도 5에서 볼 수 있듯이, 현재 블록의 제 2 카테고리의 참조 블록들은 B1, B2, B3, B4, B5, B6 및 B7을 포함한다. B1, B2, B3 및 B4는 공간상 현재 블록과 겹친다. B5, B6 및 B7는 공간상 현재 블록과 인접해 있다. 이 실시 예에서, 현재 블록의 제 2 참조 블록 세트 안의 제 2 카테고리의 참조 블록들은 현재 블록의 좌측 상단 꼭지점과 같은 공간에 위치한 점을 포함하는 B1, 현재 블록의 우측 하단 꼭지점과 같은 공간에 위치한 점을 포함하는 B4으로 구성될 수 있다.

- [133] 블록 (405)에서 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 제 2 참조 블록 세트 안의 참조 블록의 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이가 획득된다.
- [134] 이 실시 예에서 현재 블록의 움직임 벡터와 현재 블록의 움직임 벡터의 제 1 예측자가 결정된 후, 현재 블록의 움직임 벡터와 현재 블록의 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이가 계산되고 저장된다. 이 블록 (405)에서 제 2 참조 블록 세트 안의 참조 블록들은 모두 부호화된 블록이다. 그리고 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이는 생성되고 저장된다. 따라서 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이는 바로 획득될 수 있다.
- [135] 블록 (405)에서 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이가 획득될 때, 특정 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않을 때, 그 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않는다고 표시하는 표시가 리턴된다.
- [136] 블록 (406)에서 현재 블록의 제 1 움직임 벡터 차이의 예측자는 블록 (405)에서 계산된 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이에 따라 계산된다.
- [137] 블록 (406)은 다음과 같이 구현될 수 있다. 부호기는 존재하는 제 1 움직임 벡터 차이들과 영 벡터 중 하나를 현재 블록의 제 1 움직임 벡터 차이의 예측자로 선택한다. 그리고 선택된 값의 인덱스를 비트 스트림에 기록한다. 복호기는 비트 스트림 안의 인덱스에 따라 블록 (405)에서 획득한 존재하는 제 1 움직임 벡터 차이들과 영 벡터로부터 올바른 값을 선택한다. 그리고 선택한 값을 현재 블록의 제 1 움직임 벡터 차이의 예측자로 채택한다. 비트 스트림 안에서 전송된 인덱스는, 도1에 대한 설명에서 제 1 움직임 벡터 차이의 예측자를 계산하기 위한 비트 스트림 안의 유도 정보를 전송하는 방법에서 언급한, 현재 블록의 제 1 움직임 벡터 차이의 예측자의 유도 정보로 채택될 수 있다.
- [138] 제 1 실시 예와 유사하게, 블록 (406)에서 제 1 움직임 벡터 차이는 스케일링될 수 있다.
- [139] 블록 (407)에서 현재 블록의 움직임 벡터의 제 2 예측자는 현재 블록의 움직임 벡터의 제 1 예측자와 현재 블록의 제 1 움직임 벡터 차이의 예측자에 따라 유도된다.
- [140] 블록 (407)은 여기서 설명되지 않았지만 제 1 실시 예의 블록 (207)과 동일하다.

- [141] 블록 (407)에서 현재 블록의 움직임 벡터의 제 2 예측자가 결정된 후에, 현재 블록의 움직임 벡터의 제 2 예측자는 현재 블록의 부호화 또는 복호화를 위해 사용될 수 있는 현재 블록의 움직임 벡터의 예측자, 현재 블록의 움직임 벡터의 예측자의 후보, 현재 블록의 움직임 벡터 또는 현재 블록의 움직임 벡터의 후보로 채택될 수 있다.
- [142] 그리고 제 2 실시 예에서 제공하는 방법 흐름도는 종료된다.
- [143] 위 두 개의 실시 예에서 현재 블록의 움직임 벡터의 제 1 예측자는 먼저 계산된다. 그리고 현재 블록의 제 1 움직임 벡터 차이의 예측자는 계산된다. 실제 응용들에 있어서, 현재 블록의 움직임 벡터의 제 1 예측자와 현재 블록의 제 1 움직임 벡터 차이의 예측자는 동시에 획득될 수 있다. 또는 현재 블록의 제 1 움직임 벡터 차이의 예측자는 먼저 계산되고, 현재 블록의 움직임 벡터의 제 1 예측자가 계산될 수 있다. 제 3 실시 예와 제 4 실시 예에 있어서 움직임 정보를 획득하는 방법은, 현재 블록의 제 1 움직임 벡터 차이의 예측자가 먼저 계산되고, 현재 블록의 움직임 벡터의 제 1 예측자가 계산된다.
- [144] 제 3 실시 예는 다음과 같이 설명된다.
- [145] 도 6은 본 발명에 따른 제 3 실시 예의 흐름도이다. 도 5는 본 발명의 일 실시 예에 따른 현재 블록의 제 2 카테고리의 참조 블록들을 설명하는 개략도이다. 도 6에서 볼 수 있듯이 움직임 정보를 획득하는 방법은 다음과 같이 구현된다.
- [146] 블록 (601)에서 현재 블록의 제 2 참조 블록 세트는 결정된다.
- [147] 블록 (601)은 다음과 같이 구현될 수 있다. 제 2 참조 블록 세트는 오직 제 2 카테고리의 참조 블록들을 포함한다. 도 5에서 도시되어 있듯이, 현재 블록의 제 2 카테고리의 참조 블록들은 B1, B2, B3, B4, B5, B6 및 B7을 포함한다. B1, B2, B3 및 B4는 공간상 현재 블록과 겹친다. B5, B6 및 B7는 공간상 현재 블록과 인접해 있다. 이 실시 예에서 현재 블록의 제 2 참조 블록 세트 안의 참조 블록들은 현재 블록의 중심점과 같은 공간에 위치한 점을 포함하는 B1, 현재 블록의 좌측 상단 꼭지점과 같은 공간에 위치한 점을 포함하는 B1, 현재 블록의 우측 하단 꼭지점과 같은 공간에 위치한 점을 포함하는 B4으로 구성될 수 있다.
- [148] 블록(602)에서 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 제 2 참조 블록 세트 안의 참조 블록의 움직임 벡터의 제 1 예측자는 획득된다. 그리고 참조 블록의 움직임 벡터와 참조 블록의 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이는 계산된다.
- [149] 블록 (602)는 제 1 실시 예에서 설명된 블록 (205)와 유사하다. 블록(602)에서 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 제 2 참조 블록 세트 안의 참조 블록의 움직임 벡터의 제 1 예측자는 획득된 때, 만약 특정 참조 블록의 움직임 벡터 또는 참조 블록의 움직임 벡터의 제 1 예측자가 존재하지 않으면, 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않는다고 표시하는 표시가 리턴된다.
- [150] 제 1 실시 예와 유사하게 스케일링 동작은 블록(602)에서 수행될 수 있다.

- [151] 블록 (603)에서 블록 (602)에서 획득한 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이에 따라 현재 블록의 제 1 움직임 벡터 차이의 예측자는 획득된다.
- [152] 블록 (603)에서, 움직임 벡터를 예측하는 기존의 방법에 따라 현재 블록의 제 1 움직임 벡터 차이는 예측될 수 있다.
- [153] 블록 (603)은 다음과 같이 구현될 수 있다. 블록 (602)에서 획득된 참조 블록들의 제 1 움직임 벡터 차이들의 수평 성분들의 중간 값과 블록 (602)에서 획득된 참조 블록들의 제 1 움직임 벡터 차이들의 수직 성분들의 중간 값이 각각 현재 블록의 제 1 움직임 벡터 차이의 예측자의 수평 성분과 현재 블록의 제 1 움직임 벡터 차이의 예측자의 수직 성분으로 채택한다. 만약 특정 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않거나 0과 같은 경우, 영 벡터가 현재 블록의 제 1 움직임 벡터 차이의 예측자로 채택한다.
- [154] 블록 (603)은 다음과 같이 구현될 수 있다. 부호기는 블록 (602)에서 획득한 존재하는 제 1 움직임 벡터 차이들 중 하나를 현재 블록의 제 1 움직임 벡터 차이의 예측자로 선택한다. 그리고 선택된 값의 인덱스를 비트 스트림에 기록한다. 복호기는 비트 스트림 안의 인덱스에 따라 블록 (602)에서 획득한 존재하는 제 1 움직임 벡터 차이들로부터 올바른 값을 선택한다. 그리고 선택한 값을 현재 블록의 제 1 움직임 벡터 차이의 예측자로 채택한다. 비트 스트림 안에서 전송된 인덱스는, 도 1에 대한 설명에서 제 1 움직임 벡터 차이의 예측자를 계산하기 위한 비트 스트림 안의 유도 정보를 전송하는 방법에서 언급한, 현재 블록의 제 1 움직임 벡터 차이의 예측자의 유도 정보로 채택될 수 있다.
- [155] 제 1 실시 예와 유사하게, 스케일링 동작을 수행 될 수 있다.
- [156] 블록 (604)에서 현재 블록의 제 1 참조 블록 세트는 결정된다.
- [157] 블록 (604)는 다음과 같이 구현될 수 있다. 제 1 참조 블록 세트는 오직 제 2 카테고리의 참조 블록들을 포함한다. 도 5에서 볼 수 있듯이, 현재 블록의 제 2 카테고리의 참조 블록들은 B1, B2, B3, B4, B5, B6 및 B7을 포함한다. B1, B2, B3 및 B4는 공간상 현재 블록과 겹친다. B5, B6 및 B7는 공간상 현재 블록과 인접해 있다. 이 실시 예에서, 현재 블록의 제 2 참조 블록 세트 안의 참조 블록들은 현재 블록의 좌측 상단 꼭지점과 같은 공간에 위치한 점을 포함하는 B1, 현재 블록의 우측 하단 꼭지점과 같은 공간에 위치한 점을 포함하는 B4으로 구성될 수 있다.
- [158] 블록 (605)에서 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터는 획득된다.
- [159] 블록 (605)에서 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터는 획득될 때, 만약 제 1 참조 블록 세트 안의 특정 참조 블록의 움직임 벡터가 존재하지 않으면, 그 참조 블록의 움직임 벡터가 존재하지 않는다고 표시하는 표시가 리턴된다.
- [160] 블록 (606)에서 현재 블록의 움직임 벡터의 제 1 예측자는 블록 (605)에서

- [161] 획득된 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터에 따라 계산된다. 현재 블록의 움직임 벡터의 제 1 예측자는 현재 블록의 움직임 벡터를 예측 또는 유도하는 기준의 복호화 방법에서 사용된 방법과 동일한 방법에 따라 계산될 수 있다.
- [162] 블록 (606)은 다음과 같이 구현될 수 있다. 부호기는 존재하는 움직임 벡터들 중 하나를 현재 블록의 제 1 움직임 벡터 차이의 예측자로 선택한다. 그리고 선택된 값의 인덱스를 비트 스트림에 기록한다. 복호기는 비트 스트림 안의 인덱스에 따라 블록 (605)에서 획득한 존재하는 움직임 벡터들로부터 올바른 값을 선택한다. 그리고 선택한 값을 현재 블록의 움직임 벡터의 제 1 예측자로 채택한다. 만약 제 1 참조 블록 세트 안의 특정 참조 블록의 움직임 벡터가 존재하지 않을 때, 참조 블록의 존재하지 않는 움직임 벡터는 영 벡터로 설정된다. 비트 스트림 안에서 전송된 인덱스는, 도 1에 대한 설명에서 움직임 벡터의 제 1 예측자를 계산하기 위한 비트 스트림 안의 유도 정보를 전송하는 방법에서 언급한, 현재 블록의 움직임 벡터의 제 1 예측자의 유도 정보로 채택될 수 있다.
- [163] 제 1 참조 예와 유사하게, 스케일링 동작은 블록 (606)에서 수행될 수 있다.
- [164] 블록 (607)에서 현재 블록의 움직임 벡터의 제 2 예측자는 현재 블록의 움직임 벡터의 제 1 예측자와 현재 블록의 제 1 움직임 벡터 차이의 예측자에 따라 유도된다.
- [165] 블록 (607)은 제 1 참조 예에서 설명한 블록 (207)과 동일하다.
- [166] 블록 (607)에서 현재 블록의 움직임 벡터의 제 2 예측자가 결정된 후에, 현재 블록의 움직임 벡터의 제 2 예측자는 현재 블록의 부호화 또는 복호화를 위해 사용될 수 있는 현재 블록의 움직임 벡터의 예측자, 현재 블록의 움직임 벡터의 예측자의 후보, 현재 블록의 움직임 벡터 또는 현재 블록의 움직임 벡터의 후보로 채택될 수 있다.
- [167] 그리고 제 3 참조 예에서 제공하는 방법 흐름도는 종료된다.
- [168] 제 4 참조 예는 다음과 같이 설명된다.
- [169] 도 7은 본 발명에 따른 제 4 참조 예의 흐름도이다. 도 3은 본 발명의 일 실시 예에 따른 현재 블록의 제 1 카테고리 참조 블록들을 설명하는 개략도이다. 도 5는 본 발명의 일 실시 예에 따른 현재 블록의 제 2 카테고리 참조 블록들을 설명하는 개략도이다. 도 7에서 볼 수 있듯이 움직임 정보를 획득하는 방법은 다음과 같이 구현된다.
- [170] 블록 (701)에서 현재 블록의 제 2 참조 블록 세트는 결정된다.
- [171] 블록 (701)은 다음과 같이 구현될 수 있다. 제 2 참조 블록 세트는 오직 제 2 카테고리 참조 블록들의 참조 블록들을 포함한다. 도 5에서 도시되어 있듯이, 현재 블록의 제 2 카테고리 참조 블록들은 B1, B2, B3, B4, B5, B6 및 B7을 포함한다. B1, B2, B3 및 B4는 공간상 현재 블록과 겹친다. B5, B6 및 B7은 공간상 현재 블록과 인접해 있다. 이 실시 예에서 현재 블록의 제 2 참조 블록 세트 안의 참조 블록들은 현재

블록의 좌측 상단 꼭지점과 같은 공간에 위치한 점을 포함하는 상단 블록 B1, 현재 블록의 좌측 상단 꼭지점과 같은 공간에 위치한 점을 포함하는 좌측 블록 B1, 현재 블록의 좌측 상단 꼭지점과 같은 공간에 위치한 점을 포함하는 좌측 상단 블록 B1으로 구성될 수 있다.

- [172] 블록 (702)에서 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 제 2 참조 블록 세트 안의 참조 블록의 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이는 획득된다.
- [173] 블록 (702)는 제 2 실시 예에서 설명된 블록 (405)와 유사하다. 블록 (702)에서 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이가 획득된 때, 만약 제 2 참조 블록 세트 안의 특정 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않으면, 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않는다고 표시하는 표시가 리턴된다.
- [174] 제 1 실시 예와 유사하게 스케일링 동작은 블록(702)에서 수행될 수 있다.
- [175] 블록 (703)에서 블록 (702)에서 획득한 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이에 따라 현재 블록의 제 1 움직임 벡터 차이의 예측자는 획득된다.
- [176] 블록 (703)에서, 움직임 벡터를 예측하는 기존의 방법에 따라 현재 블록의 제 1 움직임 벡터 차이의 예측자는 계산될 수 있다.
- [177] 블록 (703)은 다음과 같이 구현될 수 있다. 블록 (702)에서 상단 블록에 대하여 리턴된 제 1 움직임 벡터 차이는 mvd_U 로, 좌측 블록에 대하여 리턴된 제 1 움직임 벡터 차이는 mvd_L 로, 그리고 좌측 상단 블록에 대하여 리턴된 제 1 움직임 벡터 차이가 mvd_LU 로 복호화 된다. mvd_L , mvd_U 및 mvd_LU 중 두 개 사이의 거리가 계산된다. 즉, $d12=|mvd_L-mvd_U|$, $d13=|mvd_L-mvd_LU|$, 그리고 $d23=|mvd_U-mvd_LU|$ 와 같은 방법으로 계산될 수 있다. 만약 $d12$, $d13$ 및 $d23$ 중에서 $d12$ 이 가장 작은 값이라면, mvd_LU 가 현재 블록의 제 1 움직임 벡터 차이의 예측자로 채택된다. 만약 $d12$, $d13$ 및 $d23$ 중에서 $d13$ 이 가장 작은 값이라면, mvd_U 가 현재 블록의 제 1 움직임 벡터 차이의 예측자로 채택된다. 만약 $d12$, $d13$ 및 $d23$ 중에서 $d23$ 이 가장 작은 값이라면, mvd_L 가 현재 블록의 제 1 움직임 벡터 차이의 예측자로 채택된다. 만약 특정 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않거나 0과 같다면, 현재 블록의 제 1 움직임 벡터 차이의 예측자는 영 벡터로 설정된다.
- [178] 제 1 실시 예와 유사하게, 스케일링 동작을 수행 될 수 있다.
- [179] 블록 (704)에서 현재 블록의 제 1 참조 블록 세트는 결정된다.
- [180] 블록(704)는 다음과 같이 구현될 수 있다. 제 1 참조 블록 세트는 오직 제 1 카테고리의 참조 블록들과 제 2 카테고리의 참조 블록들을 포함한다. 도 3에서 볼 수 있듯이, 현재 블록의 제 1 카테고리의 참조 블록들은 AL1, AL2, AU1, AU2, AU3, AC1 및 AC2을 포함한다. 도 5에서 볼 수 있듯이, 현재 블록의 제 2 카테고리의 참조 블록들은 B1, B2, B3, B4, B5, B6 및 B7을 포함한다. B1, B2, B3

및 B4는 공간상 현재 블록과 접친다. B5, B6 및 B7는 공간상 현재 블록과 인접해 있다. 이 실시 예에서, 현재 블록의 제 1 참조 블록 세트 안의 참조 블록들은 현재 블록의 좌측 상단 꼭지점의 좌측 상단에 위치한 인접 블록 AU3, 현재 블록의 좌측 하단 꼭지점의 좌측 하단에 위치한 인접 블록 AC1 및 현재 블록의 중심점과 같은 공간에 위치한 점을 포함하는 B1으로 구성될 수 있다.

[181] 블록 (705)에서 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터가 획득된다.
[182] 블록 (705)에서 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터가 획득된 때, 만약 제 1 참조 블록 세트 안의 특정 참조 블록의 움직임 벡터가 존재하지 않으면, 그 참조 블록의 움직임 벡터가 존재하지 않는다고 표시하는 표시가 리턴된다.

[183] 블록 (706)에서 현재 블록의 움직임 벡터의 제 1 예측자는 블록 (705)에서 획득된 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터에 따라 계산된다. 현재 블록의 움직임 벡터의 제 1 예측자는 블록의 움직임 벡터를 예측 또는 유도하는 기준의 부호화 또는 복호화 방법에서 사용된 방법과 동일한 방법에 따라 계산될 수 있다.

[185] 이 실시 예에서 블록 (706)에서 다음과 같이 구현될 수 있다. AC1의 움직임 벡터 mv1, AC2의 움직임 벡터 mv2 및 AC3의 움직임 벡터 mv3의 평균 값은 계산되고 mva로 복호화된다. 만약 mv1, mv2 및 mv3 중 어느 하나가 존재하지 않는다면, 존재하지 않는 것은 영 벡터로 설정된다. B1의 움직임 벡터는 mvb로 복호화된다. 그리고 mva와 mvb 중 작은 것이 현재 블록의 움직임 벡터의 제 1 예측자로 채택된다. 만약 mvb가 존재하지 않는다면, mva가 현재 블록의 움직임 벡터의 제 1 예측자로 채택된다.

[186] 제 2 실시 예와 유사하게, 현재 블록의 움직임 벡터와 현재 블록의 움직임 벡터의 제 1 예측자와 현재 블록의 제 1 움직임 벡터 사이의 예측자에 따라 유도된다.
[187] 제 1 실시 예와 유사하게, 블록 (706)에서 스케일링 동작이 수행될 수 있다.
[188] 블록 (707)에서 현재 블록의 움직임 벡터의 제 2 예측자는 현재 블록의 움직임 벡터의 제 1 예측자와 현재 블록의 제 1 움직임 벡터 사이의 예측자에 따라 유도된다.

[189] 블록 (707)은 제 1 실시 예의 블록 (207)과 유사하다.
[190] 블록 (707)에서 현재 블록의 움직임 벡터의 제 2 예측자가 결정된 후, 현재 블록의 움직임 벡터의 제 2 예측자는 현재 블록의 부호화 또는 복호화를 위해 사용될 수 있는 현재 블록의 움직임 벡터의 예측자, 현재 블록의 움직임 벡터의 예측자의 후보, 현재 블록의 움직임 벡터 또는 현재 블록의 움직임 벡터의

후보로 채택될 수 있다.

- [191] 그리고 또한 이 실시 예가 제공하는 방법 흐름도는 종료된다.
- [192] 제 5 실시 예는 다음과 같이 설명된다.
- [193] 도 8은 본 발명의 제 5 실시 예의 흐름도이다. 도9는 본 발명의 일 실시 예에 따라 현재 블록의 제 1 카테고리의 참조 블록들을 묘사하는 개략도이다. 도 10은 본 발명의 일 실시 예에 따라 현재 블록의 제 2 카테고리의 참조 블록들을 묘사하는 개략도이다. 도 8에서 볼 수 있듯이 움직임 정보를 획득하는 방법은 아래와 같이 구현된다.
- [194] 블록 (801)에서 도 9에서 참조된 현재 블록 E는 4개의 서브 블록인 E1, E2, E3, 및 E4로 분리된다.
- [195] 이 실시 예에서 현재 블록은 4개의 서브 블록들로 분리된다. 그리고 본 발명에서 설명되는 방법에 따라 각 서브 블록의 움직임 정보는 계산될 수 있다. 이 실시 예에서 아래의 블록들에 따라 4 개의 서브 블록들의 움직임 정보는 획득된다.
- [196] 블록 (802)에서 각 현재 서브 블록의 제 1 참조 블록 세트는 결정된다.
- [197] 현재 서브 블록의 4 개의 다른 서브 블록들에 대하여, 제 1 참조 블록 세트 안의 참조 블록들은 동일할 수도 있고, 부분적으로 동일할 수도 있으며, 완전히 다를 수도 있다.
- [198] 블록 (802)는 다음과 같이 구현될 수 있다. 서브 블록들은 다른 제 1 참조 블록 세트들을 가진다. 그러나 모든 제 1 참조 블록 세트들은 오직 제2 카테고리의 참조 블록들을 포함한다. 도10에 의하면 E1, E2, E3, 및 E4 의 제2 카테고리의 참조 블록들은 B1, B2, B3, B4 및 B5를 포함한다. 이 실시 예에서 현재 서브 블록의 제 1 참조 블록 세트 안의 참조 블록들은 현재 서브 블록의 중심과 같은 공간 위치를 가지는 지점을 포함하는 블록으로 설정될 수 있다. 즉 말하자면, E1의 제 1 참조 블록 세트는 B1, E2의 제 1 참조 블록 세트는 B3, E3의 제 1 참조 블록 세트는 B1, E4의 제 1 참조 블록 세트는 B4이다.
- [199] 만약 현재 서브 블록의 제 1 참조 블록 세트 안의 참조 블록 A의 움직임 정보가 서브 블록 안에서 결정되었다면, 참조 블록 A는 서브 블록이다. 다른 말로 하면, 제 1 참조 블록 세트 안의 참조 블록들은 블록들이거나 서브블록들일 수 있다. 만약 움직임 정보의 단일 조각이 블록에 대응되는 경우, 대응되는 참조 블록은 블록이다. 만약 움직임 정보의 단일 조각이 서브 블록과 대응되면, 대응되는 참조 블록은 서브 블록이다
- [200] 예를 들어, 블록 A는 현재 서브 블록에 근접한 부호화된 블록이다. 블록 A의 움직임 정보가 결정되었을 때, 블록 A는 서브 블록 A1과 A2로 분리되어 있고, 서브 블록 A1의 움직임 정보와 서브 블록 A2의 움직임 정보는 각각 계산된다. A1은 현재 서브 블록 또는 현재 서브 블록의 블록에 근접해 있다. A2은 현재 서브 블록 또는 현재 서브 블록의 블록에 근접해 있지 않다. (왜냐하면 블록 A는 움직임 정보의 단일 조각에 대응되지 않지만, 서브 블록들 A1, A2의 움직임

정보에는 대응된다.) 왜냐하면 서버 블록 A2는 현재 서버 블록 및 현재 서버 블록의 블록과 근접해 있지 않기 때문에, 서버 블록 A2는 현재 서버 블록의 제 1 참조 블록 세트에 포함되지 않는다.

[201] 블록 (803)에서 각 현재 서버 블록의 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터는 획득된다.

[202] 각 현재 서버 블록의 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터를 획득하는 방법은 기존의 방법과 동일할 수 있다. 블록 (803)에서 각 현재 서버 블록의 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터는 획득되었을 때, 만약 제 1 참조 블록 세트 안의 특정 참조 블록의 움직임 벡터가 존재하지 않는다면, 참조 블록의 움직임 벡터가 존재하지 않는다고 표시하는 표시가 리턴된다.

[203] 블록 (804)에서, 블록 (803)에서 획득한 각 현재 서버 블록의 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터에 따라 각 현재 서버 블록의 움직임 벡터의 제 1 예측자는 계산된다.

[204] 참조 블록의 움직임 벡터에 따라 각 현재 서버 블록의 움직임 벡터의 제 1 예측자를 결정하는 방법은 움직임 벡터를 예측하거나 유도하는 기존의 방법과 동일할 수 있다. 현재 서버 블록들의 움직임 벡터들의 제 1 예측자들을 결정하는 방법들은 동일하거나 다를 수 있다. 블록 (804)는 다음과 같이 구현될 수 있다. 각 현재 서버 블록의 제 1 참조 블록 세트 안의 참조 블록의 움직임 벡터는 직접적으로 각 현재 서버 블록의 움직임 벡터의 제 1 예측자로 채택될 수 있다. 만약 특정 현재 서버 블록의 제 1 참조 블록 세트 안의 참조 블록에 움직임 벡터가 없다면, 현재 서버 블록의 움직임 벡터의 제 1 예측자는 영 벡터로 설정되거나, 현재 서버 블록의 움직임 벡터의 제 2 예측자는 블록들 (805 내지 808)을 실행하지 않고 다른 방법으로 유도되어 현재 서버 블록의 움직임 벡터의 예측자, 현재 서버 블록의 움직임 벡터의 예측자의 후보, 현재 서버 블록의 움직임 벡터 또는 현재 서버 블록의 움직임 벡터의 후보로 채택될 수 있다.

[205] 블록 (203)과 유사하게, 각 현재 서버 블록의 제 1 참조 블록 세트 안의 참조 블록의 움직임 벡터는 스케일링될 수 있다.

[206] 블록 (805)에서 각 현재 서버 블록의 제 2 참조 블록 세트는 결정된다.

[207] 실제 요구 사항들에 따라 각 현재 서버 블록의 제 2 참조 블록 세트는 결정된다. 블록 (805)은 다음과 같이 구현될 수 있다. E1, E2, E3 및 E4는 동일한 제 2 참조 블록 세트들을 가지고 있다. 그리고 각 제 2 참조 블록 세트는 오직 제 1 카테고리 의 참조 블록들을 포함한다. 도 9에서 볼 수 있듯이, E1, E2, E3 및 E4의 제 1 카테고리의 참조 블록들은 AL1, AC1, AU1, AU2 및 AU3를 포함한다. 이 실시 예에서 E1, E2, E3 및 E4의 제 1 참조 블록 세트의 참조 블록들은 E1의 왼쪽 상단 꼭지점의 좌측에 위치하는 블록 AL1, E1의 왼쪽 상단 꼭지점 위에 있는 블록 AU1 및 E1의 왼쪽 상단 꼭지점의 왼쪽 상단에 있는 블록 AC1으로 구성될 수 있다. 만약 E1의 왼쪽 상단 꼭지점의 좌측에 위치한 블록 AL1은 움직임

벡터가 없는 경우, E1의 왼쪽 상단 꼭지점의 좌측에 위치한 블록 AL1은 E3의 왼쪽 상단 꼭지점의 좌측에 위치한 블록 AL1으로 교체된다. 만약 E1의 왼쪽 상단 꼭지점 위에 위치한 블록 AU1 기준 움직임 벡터가 없는 경우, E1의 왼쪽 상단 꼭지점 위에 위치한 블록 AU1 기준 움직임 벡터는 E2의 왼쪽 상단 꼭지점 위에 위치한 블록 AU2로 교체된다.

- [208] 블록 (806)에서 각 현재 서브 블록에 대하여 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 제 2 참조 블록 세트 안의 참조 블록의 움직임 벡터의 제 1 예측자는 획득된다. 그리고 참조 블록의 움직임 벡터와 참조 블록의 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이가 계산된다.
- [209] 블록 (806)은 제 1 실시 예에서 설명한 블록 (205)와 유사하지만, 각 현재 서브 블록이 각각 프로세스 된다는 점에서 블록 (205)와 다르다. 그러나 이 실시 예에서 모든 현재 서브 블록들의 제 2 참조 블록 세트들은 동일하므로, 프로세스는 오직 한 번만 수행되면 된다. 각 현재 서브 블록의 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 각 현재 서브 블록의 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터의 제 1 예측자는 블록 (806)은 획득된 때, 만약 특정 참조 블록의 움직임 벡터 또는 참조 블록의 움직임 벡터의 제 1 예측자가 존재하지 않는다면, 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않는다고 표시하는 표시가 리턴된다.
- [210] 블록 (205)와 유사하게, 제 2 참조 블록 세트 안의 참조 블록의 움직임 벡터와 제 2 참조 블록 세트 안의 참조 블록의 움직임 벡터의 제 1 예측자는 이 블록의 각 현재 서브 블록에 맞게 스케일링될 수 있다.
- [211] 블록 (807)에서 블록 (806)에서 계산된 현재 서브 블록의 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이에 따라 각 현재 서브 블록의 제 1 움직임 벡터 차이의 예측자는 계산된다.
- [212] 현재 서브 블록들의 제 1 움직임 벡터 차이들의 예측자들은 동일한 방법 또는 다른 방법들을 통하여 계산될 수 있다. 계산 방법은 실제 요구 사항들에 따라 결정될 수도 있다. 예를 들어 위의 실시 예에서 언급된 방법 중 하나가 적용될 수도 있다. 이 실시 예에서, 블록 (807)는 다음과 같이 구현될 수 있다. 현재 서브 블록의 제 2 참조 블록 세트 안의 참조 블록들의 존재하는 제 1 움직임 벡터 차이들의 평균은 현재 서브 블록의 제 1 움직임 벡터 차이의 예측자로 채택될 수 있다. 특정 현재 서브 블록의 제 2 참조 블록 세트 안의 참조 블록들의 존재하는 제 1 움직임 벡터 차이들이 모두 존재하지 않는 경우, 현재 서브 블록의 제 1 움직임 벡터 차이의 예측자는 영벡터로 설정될 수 있다.
- [213] 또한 블록 (807)는 다음과 같이 구현될 수도 있다. 부호기는 블록 (806)에서 계산된 현재 서브 블록의 제 2 참조 블록 세트 안의 참조 블록들의 존재하는 제 1 움직임 벡터 차이들 중 하나를 선택하고, 선택된 값을 현재 서브 블록의 제 1 움직임 벡터 차이의 예측자로 정하고, 선택된 값의 인덱스(index)를 비트 스트림에 기록한다. 복호기는 블록 (806)에서 계산된 현재 서브 블록의 제 2 참조

블록 세트 안의 참조 블록들의 존재하는 제 1 움직임 벡터 차이들로부터 비트 스트림 안의 인덱스에 따라 올바른 값을 선택하고, 선택된 값을 현재 서브 블록의 제 1 움직임 벡터 차이의 예측자로 정한다. 비트 스트림 안의 전송된 인덱스는 도 1의 설명 부분에서 제 1 움직임 벡터 차이의 예측자를 계산하기 위하여 비트 스트림 안의 유도 정보를 전송하는 방법에 언급된 현재 서브 블록의 제 1 움직임 벡터 차이의 예측자의 유도 정보로 채택될 수 있다. 서브 블록들은 제 1 움직임 벡터 차이의 예측자를 공유할 수 있다. 이러한 경우, 제 1 움직임 벡터 차이의 예측자의 계산 프로세스는 오직 전체 현재 블록 E에 대해 한 번만 수행되어야 하고, 오직 하나의 제 1 움직임 벡터 차이의 인덱스만 비트 스트림 안으로 전송된다

- [214] 제 1 실시예의 블록 (206)과 유사하게, 제 2 참조 블록 세트 안의 참조 블록의 제 1 움직임 벡터 차이는 이 블록에 각각 현재의 서브 블록에 대하여 스케일링될 수 있다.

- [215] 블록 (808)에서 현재 서브 블록의 움직임 벡터의 제 1 예측자와 현재 서브 블록의 제 1 움직임 벡터 차이의 예측자에 따라 현재 서브 블록의 움직임 벡터의 제 2 예측자는 유도된다.

- [216] 블록 (808)은 제 1 실시 예에서 설명된 블록 (207)과 유사하고, 각 현재 서브 블록이 각각 프로세스된다는 점에서 블록 (207)과 다르다.

- [217] 블록 (808)에서 현재 서브 블록의 움직임 벡터의 제 2 예측자가 결정된 후에, 현재 서브 블록의 움직임 벡터의 제 2 예측자는 현재 서브 블록의 부호화 또는 부호화 프로세스에 사용되는 현재 서브 블록의 움직임 벡터의 예측자, 현재 서브 블록의 움직임 벡터의 예측자의 후보, 현재 서브 블록의 움직임 벡터 또는 현재 서브 블록의 움직임 벡터의 후보로 채택될 수 있다.

- [218] 그리고 또한 이 실시 예가 제공하는 방법 흐름도는 종료된다.

- [219] 제 6 실시 예는 다음과 같다.

- [220] 도 11은 본 발명의 제 6 실시 예의 흐름도이다. 도 12는 본 발명의 일 실시 예에 따라 현재 블록의 제 1 카테고리의 참조 블록들을 표시하는 개략도이다. 도 13은 본 발명의 일 실시 예에 따라 현재 블록의 제 2 카테고리의 참조 블록들을 표시하는 개략도이다. 도 11에서 볼 수 있듯이 움직임 정보를 획득하는 방법은 아래와 같이 구현된다.

- [221] 블록 (1101)에서 현재 블록 E는 E1과 E2로 분리된다.

- [222] 도 12에서 볼 수 있듯이 현재 블록은 E1과 E2로 분리된다. 다음 블록들 (1102-1108)은 각각 E1과 E2에 대하여 수행된다.

- [223] 블록 (1102)에서 현재 서브 블록의 제 2 참조 블록 세트는 결정된다.

- [224] 블록 (1102)는 다음과 같이 구현될 수 있다. E1과 E2는 다른 제 2 참조 블록 세트들을 가진다. 그러나 각 제 2 참조 블록 세트들은 오직 제 2 카테고리의 참조 블록들을 포함한다. 도 13에 의하면 E1과 E2의 제 2 카테고리의 참조 블록들은 B1, B2, B3, B4 및 B5를 포함한다. B1, B2, B3은 공간에서 블록 E와 접친다. B4 및

- B5는 공간에서 블록 E와 근접한다. 이 실시 예에서, 현재 서버 블록의 제 1 참조 세트 안의 참조 블록들은 현재 서버 블록의 중심과 같은 공간 위치를 가지는 지점을 포함하는 블록으로 설정될 수 있다. 예를 들어 만약 현재 서버 블록이 E1일 때, B1은 현재 서버 블록의 제 2 참조 블록 세트 안의 참조 블록일 수 있다. 만약 현재 서버 블록이 E2일 때, B3은 현재 서버 블록의 제 2 참조 블록 세트 안의 참조 블록일 수 있다.
- [225] 블록 (1103)에서 현재 서버 블록의 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 현재 서버 블록의 제 2 참조 블록 세트 안의 움직임 벡터와 현재 서버 블록의 제 1 예측자는 획득된다. 그리고 참조 블록의 움직임 벡터와 참조 블록의 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이는 계산된다.
- [226] 블록 (1103)에서 현재 서버 블록의 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 현재 서버 블록의 제 2 참조 블록 세트 안의 움직임 벡터의 움직임 벡터의 제 1 예측자가 획득되었을 때, 만약 제 2 참조 블록 세트 안의 움직임 벡터의 움직임 벡터가 존재하지 않는다면, 참조 블록의 제 1 움직임 벡터 차이가 존재하지 않는다고 표시하는 표시가 리턴된다.
- [227] 위의 실시 예와 유사하게, 움직임 벡터와 제 1 움직임 벡터 차이는 이 블록에 맞추어 스케일링될 수 있다.
- [228] 블록 (1104)에서, 블록 (1103)에서 획득한 현재 서버 블록의 제 2 참조 블록 세트 안의 각 참조 블록의 제 1 움직임 벡터 차이에 따라 현재 서버 블록의 제 1 움직임 벡터 차이의 예측자는 계산된다.
- [229] 블록 (1104)는 다음과 같이 구현될 수 있다. 블록 (1103)에서 획득한 현재 서버 블록의 제 2 참조 블록 세트 안의 참조 블록의 제 1 움직임 벡터 차이는 직접적으로 현재 서버 블록의 제 1 움직임 벡터 차이의 예측자로 채택될 수 있다. 만약, 현재 서버 블록의 제 1 움직임 벡터 차이의 예측자는 영 벡터로 설정된다.
- [230] 위의 실시 예와 유사하게, 이 블록에서 제 1 움직임 벡터 차이는 스케일링될 수 있다.
- [231] 블록 (1105)에서 현재 서버 블록의 제 1 움직임 벡터 차이는 결정된다.
- [232] 블록 (1105)는 다음과 같이 구현될 수 있다. E1과 E2는 다른 제 1 참조 블록 세트들을 가지고 있다. 그러나 각 제 1 참조 블록 세트는 오직 제 1 카테고리인 참조 블록들을 포함한다. 도 12에서 볼 수 있듯이, E1과 E2의 제 1 카테고리인 참조 블록들은 AL1, AL2, AC1, AU1, AU2 및 AU3를 포함한다. 이 실시 예에서 현재 서버 블록의 제 1 참조 블록 세트의 참조 블록들은 현재 서버 블록의 좌측 상단 꼭지점 위에 있는 블록으로 구성될 수 있다. 예를 들어 만약 현재 서버 블록이 E1인 경우, AL2 및 AU1은 현재 서버 블록의 제 1 참조 블록 세트 안의 참조 블록들이다. 만약

현재 서버 블록이 E2일 경우, E1 및 AU2는 현재 서버 블록의 제 1 참조 블록 세트 안의 참조 블록들이다.

[233] 블록 (1106)에서 현재 서버 블록의 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터는 획득된다.

[234] 블록 (1106)에서 현재 서버 블록의 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터는 획득될 때, 만약 제 1 참조 블록 세트 안의 특정 참조 블록들의 움직임 벡터가 존재하지 않으면, 참조 블록의 움직임 벡터가 존재하지 않는다고 표시하는 표시가 리턴된다.

[235] 블록 (1107)에서 블록 (1106)에서 현재 서버 블록의 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터에 따라 현재 서버 블록의 움직임 벡터의 제 1 예측자는

획득된다.

[236] 블록 (1107)은 다음과 같이 구현될 수 있다. 현재 서버 블록의 제 1 참조 블록 세트 안의 참조 블록들의 움직임 벡터를 중첩시켰을 때, 현재 서버 블록의 움직임 벡터의 제 1 예측자로 채택될 수 있다. 만약 현재 서버 블록의 제 1 참조 블록 세트 안의 참조 블록들의 움직임 벡터들이 모두 존재하지 않는다면, 현재 서버 블록의 움직임 벡터는 영 벡터로 설정된다.

[237] 위의 실시 예들과 유사하게, 이 블록에서 제 1 참조 블록 세트 안의 참조 블록들의 움직임 벡터들은 스케일링될 수 있다.

[238] 블록 (1108)에서 현재 서버 블록의 움직임 벡터의 제 1 예측자와 현재 서버 블록의 제 1 움직임 벡터 사이의 예측자에 따라 현재 서버 블록의 움직임 벡터의 제 2 예측자는 획득된다.

[239] 블록 (1108)은 제 1 실시 예에서 설명한 블록 (207)과 유사하다. 그리고

블록 (207)에 대하여 현재 서버 블록에서 수행되는 프로세스라는 점이 다르다.

블록 (1108)에서 현재 서버 블록의 움직임 벡터의 제 2 예측자를 획득한 후,

현재 서버 블록의 움직임 벡터의 제 2 예측자는 현재 서버 블록의 부호화 또는

부호화 프로세스에 사용될 수 있는 현재 서버 블록의 움직임 벡터의 예측자,

현재 서버 블록의 제 1 움직임 벡터 사이의 예측자의 후보, 현재 서버 블록의

움직임 벡터 또는 현재 서버 블록의 움직임 벡터의 후보로 채택될 수 있다.

[241] 이 실시 예의 방법 흐름도는 종표된다. 이 실시 예는 현재 블록의 서버 블록들이 순차적으로 프로세스되지만, 제5 실시 예는 현재 블록의 서버 블록들이 동시에

프로세스된다는 점에서 제 5 실시 예와 다르다.

[242] 제 7 실시 예는 다음과 같이 설명된다.

[243] 도 14는 본 발명의 일 실시 예에 따라 움직임을 예측하는 장치를 설명하는

개략도이다. 도 15는 본 발명의 일 실시 예에 따라 상기 장치의 입력부를

설명하는 개략도이다. 도 16는 본 발명의 일 실시 예에 따라 상기 장치의

예측부를 설명하는 개략도이다. 도 14에서 볼 수 있듯이, 부호기/복호기 또는

비디오 부호화 또는 복호화 시스템 안의 움직임을 예측하는 장치는 일 실시 예에 따라 제공된다. 상기 장치는 입력부 (141), 예측부 (142) 및 계산부 (143)를

계산하는 제 2 예측부를 포함한다.

계산부 (143)는 현재 블록 또는 서브 블록의 움직임 벡터의 제 1 예측자와 현재 블록 또는 서브 블록의 제 1 움직임 벡터 사이의 예측자에 따라 현재 블록 또는 서브 블록의 움직임 벡터의 제 2 예측자를 계산한다. 현재 블록 또는 서브 블록의 움직임 벡터의 제 2 예측자가 획득된 후, 계산부 (143)는 현재 블록 또는 서브 블록의 움직임 벡터의 제 2 예측자를 현재 블록 또는 서브 블록의 부호화 또는 블록화를 위해 사용될 수 있는 현재 블록 또는 서브 블록의 움직임 벡터의 예측자, 현재 블록 또는 서브 블록의 움직임 벡터의 예측자의 후보, 현재 블록 또는 서브 블록의 움직임 벡터 또는 현재 블록 또는 서브 블록의 움직임 벡터의 후보로 채택될 수 있다.

물류 세트 안의 참조 블록들의 움직임 정보를 입력한다. 입력부 (141)는 다음과 같이 구현될 수 있다. 도 15에서 볼 수 있듯이, 입력부 (141)는 제 1 참조 블록 세트 정보, 입력부와 제 2 참조 블록 세트 정보 입력부를 포함한다. 제 1 참조 블록 세트 정보 입력부는 현재 블록 또는 서브 블록의 제 1 참조 블록 세트 안의 참조 블록의 움직임을 입력한다. 제 2 참조 블록 세트 정보 입력부는 현재 블록 또는 서브 블록의 제 2 참조 블록 세트 안의 참조 블록 세트 안의 참조 블록 또는 서브 블록의 제 2 참조 블록 세트 안의 참조 블록의 움직임을 입력하고, 현재 블록 또는 서브 블록의 움직임 벡터 안의 참조 블록의 제 1 움직임 벡터 차이를 계산한다. 제 2 참조 블록 세트 안의 참조 블록 또는 서브 블록의 제 1 움직임 벡터 차이를 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 예측자를 계산한다. 예측부 (142)는 현재 블록 또는 서브 블록의 움직임 벡터의 제 1 예측자와 현재 블록 또는 서브 블록의 제 1 움직임 벡터의 제 1 예측자를 계산하는 제 1 예측부와 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 예측자를 계산한다. 예측부 (142)는 다음과 같이 구현될 수 있다. 도 16에서 볼 수 있듯이, 예측부 (142)는 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 예측자를 계산한다. 예측부 (142)는 현재 블록 또는 서브 블록의 움직임 벡터의 제 1 예측자와 현재 블록 또는 서브 블록의 제 1 움직임 벡터의 제 1 움직임 벡터 차이를 예측자를

영력부 (141)는 현재 법률 또는 사범 법률의 제 1 참조 법률 세트와 제 2 참조

있는 현재 블록 또는 서브 블록의 움직임 벡터의 예측자, 현재 블록 또는 서브 블록의 움직임 벡터의 예측자의 후보, 현재 블록 또는 서브 블록의 움직임 벡터 또는 현재 블록 또는 서브 블록의 움직임 벡터의 후보로 채택될 수 있다. 그러나 기존의 기술들에서는 움직임 벡터의 제 1 예측자는 현재 블록 또는 서브 블록의 움직임 벡터의 최종 예측자, 현재 블록 또는 서브 블록의 움직임 벡터의 예측자의 후보, 현재 블록 또는 서브 블록의 움직임 벡터 또는 현재 블록 또는 서브 블록의 움직임 벡터의 후보로 채택된다. 움직임을 예측하는 기존의 방법에 기초하여 움직임 벡터의 보조 예측은 소개되었다. 예측되거나 유도된 움직임 벡터의 정확성은 움직임 벡터 차이의 예측자를 통해 향상되었다. 배경 기술에서 언급한 첫 번째 부호화 또는 복호화 방법에 대하여, 움직임 벡터의 예측자의 정확도가 향상되었기 때문에 전송된 움직임 벡터 차이는 감소하였다. 따라서 움직임 벡터의 압축 효율이 향상되었다. 배경 기술에서 언급한 두 번째 부호화 또는 복호화 방법에 대하여 최적화된 움직임 벡터가 유도되었으므로, 비디오 부호화 성능이 향상되었다. 제 1 참조 블록 세트는 오직 제 1 카테고리의 참조 블록들을 포함하고 제 2 참조 블록 세트는 오직 제 2 카테고리의 참조 블록들을 포함할 때 또는 제 1 참조 블록 세트는 오직 제 2 카테고리의 참조 블록들을 포함하고 제 2 참조 블록 세트는 오직 제 1 카테고리의 참조 블록들을 포함할 때, 공간 영역과 시간 영역 모두에서 움직임 벡터의 상관 관계가 두 예측들의 복잡성을 증가시키지 않고 제거될 수 있다. 즉, 상대적으로 더 적은 비용으로 더 큰 부호화 성능 이득을 획득할 수 있다.

- [248] 전술한 것은 본 발명의 단지 바람직한 실시 예이며, 본 발명의 보호 범위를 제한하기 위해 사용되지 않는다. 본 발명의 사상 및 원리를 벗어나지 않는 정도의 임의의 변형, 균등물 치환 및 개선은 본 발명의 보호 범위 내에 있다.

청구범위

[청구항 1]

비디오 부호화 또는 복호화에 있어서 움직임 정보를 획득하기 위한 방법은 비디오 이미지의 현재 블록 또는 서브 블록의 움직임 정보를 결정하는 방법을 포함하며, 상기 움직임 정보를 결정하는 방법은

상기 현재 블록 또는 서브 블록의 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터에 따라 상기 현재 블록 또는 서브 블록의 움직임 벡터의 제 1 예측자를 계산하는 단계;

상기 현재 블록 또는 서브 블록의 제 2 참조 블록 세트 안의 각 참조 블록의 움직임 벡터와 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트 안의 상기 참조 블록의 상기 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이를 결정하고, 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 예측자를 획득하기 위하여 상기 제 2 참조 블록 세트 안의 각 참조 블록의 상기 제 1 움직임 벡터 차이에 따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터와 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자 간의 제 1 움직임 벡터 차이를 예측하는 단계; 및

상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자와 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자에 따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 제 2 예측자를 결정하는 단계를 포함하고, 상기 제 1 참조 블록 세트는 최초로 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터를 예측하거나 유도하기 위한 참조 블록들을 포함하고; 및

상기 제 2 참조 블록 세트는 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이를 예측하기 위한 참조 블록들을 포함하는 것을 특징으로 하는 움직임 정보를 획득하기 위한 방법.

[청구항 2]

제 1 항에 있어서,

상기 현재 블록의 상기 제 1 참조 블록 세트는 현재 프레임 안에서 상기 현재 블록에 인접한 부호화된 블록 또는 복호화된 블록을 포함하거나, 상기 현재 프레임의 참조 프레임 안에서 공간적으로 상기 현재 블록에 인접하거나 겹치는 부호화된 블록 또는 복호화된 블록을 포함하고;

상기 현재 블록의 상기 제 2 참조 블록 세트는 상기 현재 블록에 인접한 상기 현재 프레임에서 부호화된 블록 또는 복호화된 블록을 포함하거나, 상기 현재 프레임의 상기 참조 프레임 안에서

공간적으로 상이한 현제 블록에 인접하거나 겹치는 부호화된 블록 또는 복호화된 블록을 포함하고;

상기 현재 서브 블록의 상기 제 1 참조 블록 세트는 상기 현재 서브 블록 또는 상기 현재 서브 블록의 상기 블록에 인접한 상기 현재 프레임에서 부호화된 블록 또는 복호화된 블록을 포함하거나,

공간적으로 자기 현재 서브 블록 또는 자기 현재 서브 블록의 자기
블록에 인접하거나 겹치는 자기 현재 블록의 자기 현재 서브 블록 또는
프레임에서 부호화된 블록 또는 복호화된 블록을 포함하고; 및

상기 형제 서브 블록의 상기 제 2 참조 블록 세트는 상기 형제 서브 블록 또는 상기 현재 서브 블록의 상기 블록에 인접한 상기 형제 프레임에서 부호화된 블록 또는 복호화된 블록을 포함하거나,

공간적으로 자기 현재 서브 블록 또는 자기 현재 서브 블록의 자기
블록에 인접하거나 겹치는 자기 현재 블록의 자기 참조

[참고문헌 3] 정모를 획득하기 위한 방법, 제 1 화에 있어서,

제 1 항에 있어서,
상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자를 계산하는 단계에 있어서,

부호기에 의하여, 상기 현재 블록 또는 서브 블록의 상기 제 1 참조 블록 세트 안의 각 참조 블록의 상기 모션 벡터에 따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자를 유도하고, 비트 스트림(bit-stream)에 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자의 유도 정보를

기록하는 단계, 및

복호기에 의하여, 상기 현재 블록 또는 서브 블록의 상기 제 1 참조 블록 세트 안의 각 참조 블록의 상기 모션 벡터에 따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자와 상기 비트스트림 안의 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자의 상기 유도 정보를 유도하는 단계를 포함하는 움직임 정보 획득하기 위한 방법.

[제 4 회 구성] 제 1 회에 있어서,

제 1 항에 있어서,
상기 제 2 참조 블록 세트 안의 각 참조 블록의 상기 동작임 벡터와
상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트 안의
상기 참조 블록의 상기 동작임 벡터의 상기 제 1 예측자 간의 상기
제 1 동작임 벡터 차이를 결정하는 단계에 있어서,
상기 현재 블록 또는 서브 블록의 상기 동작임 벡터가 결정된 때,
상기 제 2 참조 블록 세트 안의 각 참조 블록의 상기 동작임 벡터와
상기 제 2 참조 블록 세트 안의 상기 참조 블록의 상기 동작임

벡터의 제 1 예측자를 획득하고, 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트의 상기 참조 블록의 상기 제 1 움직임 벡터 차이로 상기 제 2 참조 블록 세트 안의 상기 참조 블록의 상기 움직임 벡터와 상기 제 2 참조 블록 세트 안의 상기 참조 블록의 상기 움직임 벡터의 상기 제 1 예측자 간의 차이를 선택하는 단계
 상기 현재 블록 또는 서브 블록의 제 2 참조 블록 세트 안의 각 참조 블록에 대하여, 상기 참조 블록의 움직임 정보가 결정된 후, 상기 참조 블록의 다음에 처리되는 블록 또는 서브 블록의 움직임 정보가 결정되기 전에, 상기 참조 블록의 상기 제 1 움직임 벡터 차이로 상기 참조 블록의 움직임 벡터와 상기 참조 블록의 상기 움직임 벡터의 제 1 예측자 간의 차이를 선택하고, 상기 참조 블록의 상기 제 1 움직임 벡터 차이를 저장하는 단계; 및
 상기 현재 블록 또는 서브 블록의 상기 움직임 정보가 결정되면, 상기 제 2 참조 블록 세트 안의 각 참조 블록의 저장된 제 1 움직임 벡터 차이를 획득하는 단계를 포함하는 움직임 정보를 획득하기 위한 방법.

[청구항 5]

제 4 항에 있어서,
 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트의 참조 블록의 움직임 벡터 또는 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트의 상기 참조 블록의 상기 움직임 벡터의 예측자가 존재하지 않을 때, 상기 움직임 정보를 획득하기 위한 방법은,
 상기 참조 블록의 제 1 움직임 벡터 차이는 상기 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 상기 예측자의 계산에 사용되지 않을 때, 상기 참조 블록의 상기 제 1 움직임 벡터 차이가 존재하지 않는다고 판단하는 단계; 또는
 상기 참조 블록의 제 1 움직임 벡터 차이는 상기 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 상기 예측자의 계산에 사용될 때, 상기 참조 블록의 상기 제 1 움직임 벡터 차이를 영 벡터로 설정하는 단계를 포함하는 움직임 정보를 획득하기 위한 방법.

[청구항 6]

제 5 항에 있어서,
 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트 안의 상기 참조 블록의 상기 제 1 움직임 벡터 차이가 존재하지 않을 때, 상기 움직임 정보를 획득하기 위한 방법은 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자를 영 벡터로 설정하는 단계를 포함하거나,
 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트 안의

제 1 항에 있어서,
상기 원재 블록 또는 서브 블록의 제 1 움직임 벡터 차이를 예측하는 단계는,

예측하는 단계는,

상기 제 2 참조 블록 세트의 상기 참조 블록들의 제 1 움직임 벡터
차이들의 수평 성분들 중 최소값을 상기 현재 블록 또는 서브
블록의 상기 제 1 움직임 벡터 차이의 상기 예측자의 수평
성분으로 채택하고, 상기 제 2 참조 블록 세트의 상기 참조
블록들의 제 1 움직임 벡터 차이들의 수직 성분들 중 최소값을
상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의
상기 예측자의 수직 성분으로 채택하는 단계; 또는

상기 제 2 참조 블록 세트의 상기 참조 블록들의 제 1 움직임 벡터 차이들의 평균을 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자로 채택하는 단계; 또는 움직임 벡터 차이의 상기 예측자로 채택하는 단계; 또는 움직임 벡터 차이의 수평 성분들 중 중간값을 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자의 수평

성분으로 채택하고, 상기 제2 참조 블록 세트의 상기 참조 블록들의 제1 몸적임 벡터 차이들의 수직 성분들 중 중간값을 상기 현재 블록 또는 서브 블록의 상기 제1 몸적임 벡터 차이의 상기 예측자의 수직 성분으로 채택하는 단계; 또는

상기 제 2 참조 블록 세트의 상기 참조 블록들의 제 1 움직임 벡터 차이를 응 각 두 개 사이의 거리를 계산하고, 상기 거리들에 따라 상기 현재 블록 또는 서브 블록의 제 1 움직임 벡터 차이의 상기 예측자를 유도하는 단계; 또는

부호기에 의하여, 상기 제2 참조 블록 세트의 각 참조 블록의 상기 제1 움직임 벡터 차이에 따라 상기 현재 블록 또는 서브 블록의 제1 움직임 벡터 차이의 상기 예측자를 유도하고, 상기 현재 블록 또는 서브 블록의 제1 움직임 벡터 차이의 상기 예측자의 유도

정보를 비트 스트림에 기록하는 단계; 및 복호기에 의하여, 상기 제 2 참조 블록 세트 안의 각 참조 블록의 상기 제 1 움직임 벡터 차이에 따라 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자, 및 상기 현재 블록 또는 서브 블록의

상기 제 1 몸적임 베터 카이의 상기 예측자의 상기 유도 점보를

- [청구항 8] 유도하는 단계를 포함하는 움직임 정보를 획득하기 위한 방법.
제 1 항에 있어서,
상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 2 예측자를 결정하는 단계는,
상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자와 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자의 합 또는 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자와 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자의 가중합을 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 2 예측자로 채택하는 단계를 포함하는 움직임 정보를 획득하기 위한 방법.
- [청구항 9] 제 1 항 내지 제 8 항에 있어서,
상기 현재 블록 또는 서브 블록의 상기 제 1 참조 블록 세트 안의 상기 참조 블록들은 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트 안의 상기 참조 블록들과 동일하거나 다른 것을 특징으로 하는 움직임 정보를 획득하기 위한 방법.
- [청구항 10] 제 1 항 내지 제 8 항에 있어서,
블록 내에서 다른 서브 블록들에 대하여, 서브 블록의 제 1 참조 블록 세트 안의 참조 블록들은 또 다른 서브 블록의 제 1 참조 블록 세트 안의 참조 블록들과 동일하거나 다르고, 서브 블록의 제 2 참조 블록 세트 안의 참조 블록들은 또 다른 서브 블록의 제 2 참조 블록 세트 안의 참조 블록들과 동일하거나 다른 것을 특징으로 하는 움직임 정보를 획득하기 위한 방법.
- [청구항 11] 제 1 항 내지 제 8 항에 있어서,
상기 제 1 참조 블록 세트 또는 상기 제 2 참조 블록 세트 안의 참조 블록의 움직임 정보가 서브 블록 안에서 결정될 때, 상기 참조 블록은 서브 블록인 것을 특징으로 하는 움직임 정보를 획득하기 위한 방법.
- [청구항 12] 비디오 부호화 또는 복호화에 관한 움직임 정보를 획득하기 위한 장치에 있어서,
현재 블록 또는 서브 블록의 제 1 참조 블록 세트 안의 각 참조 블록의 움직임 벡터를 획득하고, 상기 현재 블록 또는 서브 블록의 제 2 참조 블록 세트 안의 각 참조 블록과 상기 현재 블록 또는 서브 블록의 상기 제 2 참조 블록 세트 안의 상기 참조 블록의 상기 움직임 벡터의 제 1 예측자 간의 제 1 움직임 벡터 차이를 결정하는 입력부;
상기 제 1 참조 블록 세트 안의 각 참조 블록의 상기 움직임 벡터에

따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자를 계산하고, 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이의 상기 예측자를 획득하기 위하여 상기 제 2 참조 블록 세트 안의 각 참조 블록의 상기 제 1 움직임 벡터 차이에 따라 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이를 예측하는 예측부; 및

상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 상기 제 1 예측자에 따라 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터의 제 2 예측자를 결정하는 계산부를 포함하고,

상기 제 1 참조 블록 세트는 최초로 상기 현재 블록 또는 서브 블록의 상기 움직임 벡터를 예측하거나 유도하기 위한 참조 블록들을 포함하고; 및

상기 제 2 참조 블록 세트는 상기 현재 블록 또는 서브 블록의 상기 제 1 움직임 벡터 차이를 예측하기 위한 참조 블록들을 포함하는 것을 특징으로 하는 움직임 정보를 획득하기 위한 장치.