

**(19) 대한민국특허청(KR)**
(12) 공개특허공보(A)**(11) 공개번호** 10-2020-0058565
(43) 공개일자 2020년05월27일

- | | |
|--|--|
| <p>(51) 국제특허분류(Int. Cl.)
 <i>H04N 19/11</i> (2014.01) <i>H04N 19/105</i> (2014.01)
 <i>H04N 19/132</i> (2014.01) <i>H04N 19/176</i> (2014.01)
 <i>H04N 19/593</i> (2014.01) <i>H04N 19/70</i> (2014.01)</p> <p>(52) CPC특허분류
 <i>H04N 19/11</i> (2015.01)
 <i>H04N 19/105</i> (2015.01)</p> <p>(21) 출원번호 10-2020-7013507</p> <p>(22) 출원일자(국제) 2018년11월30일
 심사청구일자 2020년05월12일</p> <p>(85) 번역문제출일자 2020년05월12일</p> <p>(86) 국제출원번호 PCT/KR2018/015033</p> <p>(87) 국제공개번호 WO 2019/107999
 국제공개일자 2019년06월06일</p> <p>(30) 우선권주장
 62/593,195 2017년11월30일 미국(US)</p> | <p>(71) 출원인
 엘지전자 주식회사
 서울특별시 영등포구 여의대로 128 (여의도동)</p> <p>(72) 발명자
 허진
 서울특별시 서초구 양재대로11길 19 LG전자 특허센터</p> <p>김승환
 서울특별시 서초구 양재대로11길 19 LG전자 특허센터</p> <p>임재현
 서울특별시 서초구 양재대로11길 19 LG전자 특허센터</p> <p>(74) 대리인
 특허법인(유한)케이비케이</p> |
|--|--|

전체 청구항 수 : 총 14 항

(54) 발명의 명칭 비디오 신호의 처리 방법 및 장치**(57) 요약**

본 발명은 비디오 신호를 위한 비트스트림을 디코딩하는 방법 및 이를 위한 장치에 관한 것으로서, 현재 블록의 인트라 예측을 위한 참조 샘플 간의 유사도를 나타내는 값을 획득하는 단계; 상기 획득한 값과 특정 문턱 값에 기반하여 상기 현재 블록의 인트라 예측 모드를 결정하는 단계; 및 상기 결정된 인트라 예측 모드에 기반하여 상기 현재 블록을 복원하는 단계를 포함하되, 상기 현재 블록의 인트라 예측 모드를 결정하는 단계는, 상기 유사도를 나타내는 값이 상기 특정 문턱 값보다 작은 경우 상기 비트스트림으로부터 인덱스 정보를 획득하는 것과, 상기 현재 블록의 인트라 예측 모드가 상기 현재 블록의 이웃 블록으로부터 유도되는지 여부를 지시하는 플래그 정보를 획득하는 것은 생략되며, 상기 현재 블록의 이웃 블록의 인트라 예측 모드에 기반하여 후보 모드 리스트를 구성하는 것과, 상기 구성된 후보 모드 리스트 중에서 상기 인덱스 정보가 지시하는 후보 모드를 상기 현재 블록의 인트라 예측 모드로 결정하는 것을 포함하는 방법 및 이를 위한 장치에 관한 것이다.

(52) CPC특허분류

H04N 19/132 (2015.01)

H04N 19/176 (2015.01)

H04N 19/593 (2015.01)

H04N 19/70 (2015.01)

명세서

청구범위

청구항 1

디코딩 장치에서 비디오 신호를 위한 비트스트림을 디코딩하는 방법으로서,

현재 블록의 인트라 예측을 위한 참조 샘플 간의 유사도를 나타내는 값을 획득하는 단계;

상기 획득한 값과 특정 문턱 값에 기반하여 상기 현재 블록의 인트라 예측 모드를 결정하는 단계; 및

상기 결정된 인트라 예측 모드에 기반하여 상기 현재 블록을 복원하는 단계를 포함하되,

상기 현재 블록의 인트라 예측 모드를 결정하는 단계는,

상기 유사도를 나타내는 값이 상기 특정 문턱 값보다 작은 경우 상기 비트스트림으로부터 인덱스 정보를 획득하는 것과, 상기 현재 블록의 인트라 예측 모드가 상기 현재 블록의 이웃 블록으로부터 유도되는지 여부를 지시하는 플래그 정보를 획득하는 것은 생략되며,

상기 현재 블록의 이웃 블록의 인트라 예측 모드에 기반하여 후보 모드 리스트를 구성하는 것과,

상기 구성된 후보 모드 리스트 중에서 상기 인덱스 정보가 지시하는 후보 모드를 상기 현재 블록의 인트라 예측 모드로 결정하는 것을 포함하는, 방법.

청구항 2

청구항 1에 있어서,

상기 유사도를 나타내는 값은 상기 현재 블록의 인트라 예측을 위한 참조 샘플 중에서 가장 큰 값과 가장 작은 값의 차이에 기반하여 획득되는, 방법.

청구항 3

청구항 1에 있어서,

상기 유사도를 나타내는 값은 상기 현재 블록의 인트라 예측을 위한 참조 샘플의 분산에 기반하여 획득되는, 방법.

청구항 4

청구항 1에 있어서,

상기 특정 문턱 값은 상기 현재 블록의 양자화 파라미터에 기반하여 결정되는, 방법.

청구항 5

청구항 1에 있어서,

상기 현재 블록의 인트라 예측을 위한 참조 샘플은 상기 현재 블록에 인접한 상단 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 우측 상단 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 좌측 상단 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 좌측 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 좌측 하단 이웃 블록의 참조 샘플을 포함하는, 방법.

청구항 6

청구항 1에 있어서,

상기 현재 블록의 인트라 예측 모드를 결정하는 단계는,

상기 유사도를 나타내는 값이 상기 특정 문턱 값보다 큰 경우 상기 비트스트림으로부터 상기 플래그 정보를 획득하는 것과, 상기 플래그 정보가 상기 현재 블록의 이웃 블록으로부터 상기 현재 블록의 인트라 예측 모드가 유도됨을 지시하는 경우 상기 비트스트림으로부터 상기 인덱스 정보를 획득하는 것을 더 포함하는, 방법.

청구항 7

청구항 6에 있어서,

상기 현재 블록의 인트라 예측 모드를 결정하는 단계는,

상기 플래그 정보가 상기 현재 블록의 이웃 블록으로부터 상기 현재 블록의 인트라 예측 모드가 유도되지 않음을 지시하는 경우 상기 비트스트림으로부터 상기 현재 블록을 위한 인트라 예측 모드 정보를 획득하는 것을 더 포함하는, 방법.

청구항 8

비디오 신호를 위한 비트스트림을 디코딩하기 위한 장치로서,

메모리; 및

상기 메모리에 동작시 연결되는 프로세서를 포함하되, 상기 프로세서는,

현재 블록의 인트라 예측을 위한 참조 샘플 간의 유사도를 나타내는 값을 획득하고,

상기 획득한 값과 특정 문턱 값에 기반하여 상기 현재 블록의 인트라 예측 모드를 결정하고,

상기 결정된 인트라 예측 모드에 기반하여 상기 현재 블록을 복원하도록 구성되며,

상기 현재 블록의 인트라 예측 모드를 결정하는 것은,

상기 유사도를 나타내는 값이 상기 특정 문턱 값보다 작은 경우 상기 비트스트림으로부터 인덱스 정보를 획득하는 것과, 상기 현재 블록의 인트라 예측 모드가 상기 현재 블록의 이웃 블록으로부터 유도되는지 여부를 지시하는 플래그 정보를 획득하는 것은 생략되며,

상기 현재 블록의 이웃 블록의 인트라 예측 모드에 기반하여 후보 모드 리스트를 구성하는 것과,

상기 구성된 후보 모드 리스트 중에서 상기 인덱스 정보가 지시하는 후보 모드를 상기 현재 블록의 인트라 예측 모드로 결정하는 것을 포함하는, 장치.

청구항 9

청구항 8에 있어서,

상기 유사도를 나타내는 값은 상기 현재 블록의 인트라 예측을 위한 참조 샘플 중에서 가장 큰 값과 가장 작은 값의 차이에 기반하여 획득되는, 장치.

청구항 10

청구항 8에 있어서,

상기 유사도를 나타내는 값은 상기 현재 블록의 인트라 예측을 위한 참조 샘플의 분산에 기반하여 획득되는, 장치.

청구항 11

청구항 8에 있어서,

상기 특정 문턱 값은 상기 현재 블록의 양자화 파라미터에 기반하여 결정되는, 장치.

청구항 12

청구항 8에 있어서,

상기 현재 블록의 인트라 예측을 위한 참조 샘플은 상기 현재 블록에 인접한 상단 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 우측 상단 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 좌측 상단 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 좌측 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 좌측 하단 이웃 블록의 참조 샘플을 포함하는, 장치.

청구항 13

청구항 8에 있어서,

상기 현재 블록의 인트라 예측 모드를 결정하는 것은,

상기 유사도를 나타내는 값이 상기 특정 문턱 값보다 큰 경우 상기 비트스트림으로부터 상기 플래그 정보를 획득하는 것과, 상기 플래그 정보가 상기 현재 블록의 이웃 블록으로부터 상기 현재 블록의 인트라 예측 모드가 유도됨을 지시하는 경우 상기 비트스트림으로부터 상기 인덱스 정보를 획득하는 것을 더 포함하는, 장치.

청구항 14

청구항 8에 있어서,

상기 현재 블록의 인트라 예측 모드를 결정하는 것은,

상기 플래그 정보가 상기 현재 블록의 이웃 블록으로부터 상기 현재 블록의 인트라 예측 모드가 유도되지 않음을 지시하는 경우 상기 비트스트림으로부터 상기 현재 블록을 위한 인트라 예측 모드 정보를 획득하는 것을 더 포함하는, 장치.

발명의 설명

기술 분야

[0001] 본 발명은 비디오 처리 방법에 관한 것으로서, 보다 구체적으로는 인트라 예측을 이용한 비디오 신호의 처리 방법 및 이를 위한 장치에 관한 것이다.

배경 기술

[0002] 디지털 동영상 처리 기술이 급격히 발전함에 따라 고화질 디지털방송, 디지털 멀티미디어 방송, 인터넷 방송 등과 같은 다양한 매체를 이용한 디지털 멀티미디어 서비스가 활성화되고 있으며, 고화질 디지털 방송이 일반화되면서 다양한 서비스 애플리케이션이 개발되고 있고, 고화질, 고해상도의 영상을 위한 고속 동영상 처리 기술들이 요구되고 있다. 이를 위해, H.265/HEVC(High Efficiency Video Coding), H.264/AVC(Advanced Video Coding)와 같은 비디오 신호의 코딩에 관한 표준이 활발히 논의되고 있다.

발명의 내용

해결하려는 과제

[0003] 본 발명의 목적은 비디오 신호를 효율적으로 처리할 수 있는 방법 및 이를 위한 장치를 제공하는 데 있다.

[0004] 본 발명의 다른 목적은 현재 처리되는 블록의 인트라 예측을 위한 참조 샘플의 유사도에 기반하여 인트라 예측 모드를 결정하게 함으로써 코딩 효율 및 예측 성능을 향상시키는 데 있다.

[0005] 본 발명에서 이루고자 하는 기술적 과제들은 상기 기술적 과제로 제한되지 않으며, 언급하지 않은 또 다른 기술적 과제들은 아래의 기재로부터 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에게 명확하게 이해될 수 있을 것이다.

과제의 해결 수단

[0006] 본 발명의 제1 양상으로, 디코딩 장치에서 비디오 신호를 위한 비트스트림을 디코딩하는 방법이 제공되며, 상기 방법은 현재 블록의 인트라 예측을 위한 참조 샘플 간의 유사도를 나타내는 값을 획득하는 단계; 상기 획득한 값과 특정 문턱 값에 기반하여 상기 현재 블록의 인트라 예측 모드를 결정하는 단계; 및 상기 결정된 인트라 예측 모드에 기반하여 상기 현재 블록을 복원하는 단계를 포함하되, 상기 현재 블록의 인트라 예측 모드를 결정하는 단계는, 상기 유사도를 나타내는 값이 상기 특정 문턱 값보다 작은 경우 상기 비트스트림으로부터 인덱스 정보를 획득하는 것과, 상기 현재 블록의 인트라 예측 모드가 상기 현재 블록의 이웃 블록으로부터 유도되는지 여부를 지시하는 플래그 정보를 획득하는 것은 생략되며, 상기 현재 블록의 이웃 블록의 인트라 예측 모드에 기반하여 후보 모드 리스트를 구성하는 것과, 상기 구성된 후보 모드 리스트 중에서 상기 인덱스 정보가 지시하는 후보 모드를 상기 현재 블록의 인트라 예측 모드로 결정하는 것을 포함할 수 있다.

- [0007] 본 발명의 제2 양상으로, 비디오 신호를 위한 비트스트림을 디코딩하기 위한 장치가 제공되며, 상기 장치는 메모리; 및 상기 메모리에 동작시 연결되는 프로세서를 포함하되, 상기 프로세서는, 현재 블록의 인트라 예측을 위한 참조 샘플 간의 유사도를 나타내는 값을 획득하고, 상기 획득한 값과 특정 문턱 값에 기반하여 상기 현재 블록의 인트라 예측 모드를 결정하고, 상기 결정된 인트라 예측 모드에 기반하여 상기 현재 블록을 복원하도록 구성되며, 상기 현재 블록의 인트라 예측 모드를 결정하는 것은, 상기 유사도를 나타내는 값이 상기 특정 문턱 값보다 작은 경우 상기 비트스트림으로부터 인덱스 정보를 획득하는 것과, 상기 현재 블록의 인트라 예측 모드가 상기 현재 블록의 이웃 블록으로부터 유도되는지 여부를 지시하는 플래그 정보를 획득하는 것은 생략되며, 상기 현재 블록의 이웃 블록의 인트라 예측 모드에 기반하여 후보 모드 리스트를 구성하는 것과, 상기 구성된 후보 모드 리스트 중에서 상기 인덱스 정보가 지시하는 후보 모드를 상기 현재 블록의 인트라 예측 모드로 결정하는 것을 포함할 수 있다.
- [0008] 바람직하게는, 상기 유사도를 나타내는 값은 상기 현재 블록의 인트라 예측을 위한 참조 샘플 중에서 가장 큰 값과 가장 작은 값의 차이에 기반하여 획득될 수 있다.
- [0009] 바람직하게는, 상기 유사도를 나타내는 값은 상기 현재 블록의 인트라 예측을 위한 참조 샘플의 분산에 기반하여 획득될 수 있다.
- [0010] 바람직하게는, 상기 특정 문턱 값은 상기 현재 블록의 양자화 파라미터에 기반하여 결정될 수 있다.
- [0011] 바람직하게는, 상기 현재 블록의 인트라 예측을 위한 참조 샘플은 상기 현재 블록에 인접한 상단 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 우측 상단 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 좌측 상단 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 좌측 이웃 블록의 참조 샘플, 상기 현재 블록에 인접한 좌측 하단 이웃 블록의 참조 샘플을 포함할 수 있다.
- [0012] 바람직하게는, 상기 현재 블록의 인트라 예측 모드를 결정하는 것은, 상기 유사도를 나타내는 값이 상기 특정 문턱 값보다 큰 경우 상기 비트스트림으로부터 상기 플래그 정보를 획득하는 것과, 상기 플래그 정보가 상기 현재 블록의 이웃 블록으로부터 상기 현재 블록의 인트라 예측 모드가 유도됨을 지시하는 경우 상기 비트스트림으로부터 상기 인덱스 정보를 획득하는 것을 더 포함할 수 있다.
- [0013] 바람직하게는, 상기 현재 블록의 인트라 예측 모드를 결정하는 것은, 상기 플래그 정보가 상기 현재 블록의 이웃 블록으로부터 상기 현재 블록의 인트라 예측 모드가 유도되지 않음을 지시하는 경우 상기 비트스트림으로부터 상기 현재 블록을 위한 인트라 예측 모드 정보를 획득하는 것을 더 포함할 수 있다.

발명의 효과

- [0014] 본 발명에 의하면, 비디오 신호를 효율적으로 처리할 수 있다.
- [0015] 또한, 본 발명에 의하면, 현재 처리되는 블록의 인트라 예측을 위한 참조 샘플의 유사도에 기반하여 인트라 예측 모드를 결정함으로써 코딩 효율 및 예측 성능을 향상시킬 수 있다.
- [0016] 본 발명에서 얻을 수 있는 효과는 이상에서 언급한 효과들로 제한되지 않으며, 언급하지 않은 또 다른 효과들은 아래의 기재로부터 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에게 명확하게 이해될 수 있을 것이다.

도면의 간단한 설명

- [0017] 첨부 도면은 본 발명에 관한 이해를 돕기 위해 상세한 설명의 일부로 포함되며, 본 발명에 대한 실시예를 제공하고, 상세한 설명과 함께 본 발명의 기술적 사상을 설명한다.
 - 도 1은 비디오 신호를 인코딩하기 위한 인코더를 예시한다.
 - 도 2는 비디오 신호를 디코딩하기 위한 디코더를 예시한다.
 - 도 3은 코딩 유닛의 분할 구조를 예시한다.
 - 도 4는 코딩 유닛의 분할 구조 중 쿼드트리-바이너리 트리를 예시한다.
 - 도 5는 인트라 예측 모드를 예시한다.
 - 도 6 및 도 7은 인트라 예측 모드의 시그널링 방법을 예시한다.

도 8은 확장된 인트라 예측 모드를 예시한다.

도 9는 확장된 인트라 예측 모드를 위한 MPM 후보를 예시한다.

도 10은 본 발명에 따라 유사도를 측정하기 위해 사용하는 현재 블록의 주변 참조 샘플을 예시한다.

도 11 내지 도 13은 본 발명에 따라 인트라 예측 모드를 결정하는 방법의 순서도를 예시한다.

도 14는 각 방향의 참조 샘플의 유사도가 서로 다른 경우를 예시한다.

도 15는 본 발명이 적용될 수 있는 영상 처리 장치를 예시한다.

발명을 실시하기 위한 구체적인 내용

- [0018] 이하의 기술은 비디오 신호(video signal)를 인코딩(encoding) 및/또는 디코딩하도록 구성된 영상 신호 처리 장치에서 사용될 수 있다. 일반적으로 비디오 신호는 눈으로 인지가 가능한 영상 신호(image signal) 또는 픽처들의 시퀀스를 지칭하지만, 본 명세서에서 비디오 신호는 코딩된 픽처(picture)를 나타내는 비트들의 시퀀스(sequence) 또는 비트 시퀀스에 해당하는 비트스트림을 지칭할 수 있다. 픽처(picture)는 샘플들의 배열을 지칭할 수 있으며, 프레임(frame), 영상(image) 등으로 지칭될 수 있다. 보다 구체적으로, 픽처는 샘플들의 이차원 배열 또는 이차원 샘플 배열을 지칭할 수 있다. 샘플은 픽처를 구성하는 최소 단위를 지칭할 수 있고, 픽셀(pixel), 화소(picture element), 펠(pel) 등으로 지칭될 수 있다. 샘플은 휘도(luminance, luma) 성분 및/또는 색차(chrominance, chroma, color difference) 성분을 포함할 수 있다. 본 명세서에서, 코딩은 인코딩을 지칭하는 데 사용될 수도 있고, 혹은 인코딩/디코딩을 통칭할 수 있다.
- [0019] 픽처는 적어도 하나의 슬라이스를 포함할 수 있으며, 슬라이스는 적어도 하나의 블록을 포함할 수 있다. 슬라이스는 병렬 처리 등의 목적, 데이터 손실 등으로 인해 비트스트림이 훼손된 경우 디코딩의 재동기화 등의 목적을 위해 정수 개의 블록을 포함하도록 구성될 수 있으며, 각 슬라이스는 서로 독립적으로 코딩될 수 있다. 블록은 적어도 하나의 샘플을 포함할 수 있으며, 샘플들의 배열을 지칭할 수 있다. 블록은 픽처보다 작거나 같은 크기를 가질 수 있다. 블록은 유닛으로 지칭될 수 있다. 현재 코딩(인코딩 또는 디코딩)되는 픽처를 현재 픽처라고 지칭하고, 현재 코딩(인코딩 또는 디코딩)되는 블록을 현재 블록이라고 지칭할 수 있다. 픽처를 구성하는 다양한 블록 단위가 존재할 수 있으며, 예를 들어 ITU-T(International Telecommunication Union Telecommunication Standardization Sector) H.265 표준(또는 HEVC(High Efficiency Video Coding) 표준)의 경우 코딩 트리 블록(CTB)(또는 코딩 트리 유닛(CTU)), 코딩 블록(CB)(또는 코딩 유닛(CU)), 예측 블록(Prediction Block, PB)(또는 예측 유닛(PU)), 변환 블록(Transform Block, TB)(또는 변환 유닛(TU)) 등의 블록 단위가 존재할 수 있다.
- [0020] 코딩 트리 블록은 픽처를 구성하는 가장 기본적인 단위를 지칭하며, 픽처의 텍스처(texture)에 따라 코딩 효율을 높이기 위해 쿼드-트리(quad-tree) 형태의 코딩 블록들로 분할될 수 있다. 코딩 블록은 코딩을 수행하는 기본 단위를 지칭할 수 있으며, 코딩 블록 단위로 인트라 코딩 또는 인터 코딩이 수행될 수 있다. 인트라 코딩은 인트라 예측을 이용하여 코딩을 수행하는 것을 지칭할 수 있으며, 인트라 예측은 동일한 픽처 또는 슬라이스 내에 포함된 샘플들을 이용하여 예측을 수행하는 것을 지칭할 수 있다. 인터 코딩은 인터 예측을 이용하여 코딩을 수행하는 것을 지칭할 수 있으며, 인터 예측은 현재 픽처와 서로 다른 픽처에 포함된 샘플들을 이용하여 예측을 수행하는 것을 지칭할 수 있다. 인트라 코딩을 이용하여 코딩되는 블록 또는 인트라 예측 모드로 코딩된 블록을 인트라 블록이라고 지칭할 수 있고, 인터 코딩을 이용하여 코딩되는 블록 또는 인터 예측 모드로 코딩된 블록을 인터 블록이라고 지칭할 수 있다. 또한, 인트라 예측을 이용한 코딩 모드를 인트라 모드라고 지칭할 수 있고, 인터 예측을 이용한 코딩 모드를 인터 모드라고 지칭할 수 있다.
- [0021] 예측 블록은 예측을 수행하기 위한 기본 단위를 지칭할 수 있다. 하나의 예측 블록에 대해서는 동일한 예측이 적용될 수 있다. 예를 들어, 인터 예측의 경우 하나의 예측 블록에 대해서 동일한 움직임 벡터가 적용될 수 있다. 변환 블록은 변환을 수행하기 위한 기본 단위를 지칭할 수 있다. 변환은 픽셀 도메인(또는 공간 도메인 또는 시간 도메인)의 샘플들을 주파수 도메인(또는 변환 계수 도메인)의 변환 계수로 변환하는 동작을 지칭하거나, 그 반대의 동작을 통칭할 수 있다. 특히, 주파수 도메인(또는 변환 계수 도메인)의 변환 계수를 픽셀 도메인(또는 공간 도메인 또는 시간 도메인)의 샘플들로 변환하는 동작을 역변환이라고 지칭할 수 있다. 예를 들어, 변환은 이산 코사인 변환(DCT), 이산 사인 변환(DST), 푸리에 변환 등을 포함할 수 있다. 예측 블록 및/또는 변환 블록은 코딩 블록과 동일한 크기로 설정될 수 있으며, 이 경우 코딩 블록 단위로 예측이 수행되거나 및/또는 변환이 수행될 수 있다.

- [0022] 본 명세서에서, 코딩 트리 블록(CTB)은 코딩 트리 유닛(CTU)과 혼용될 수 있고, 코딩 블록(CB)은 코딩 유닛(CU)과 혼용될 수 있고, 예측 블록(PB)은 예측 유닛(PU)과 혼용될 수 있고, 변환 블록(PB)은 변환 유닛(PU)과 혼용될 수 있다.
- [0023] 도 1은 본 발명이 적용되는 실시예로서, 비디오 신호의 인코딩이 수행되는 인코더의 개략적인 블록도를 나타낸다.
- [0024] 도 1을 참조하면, 인코더(100)는 영상 분할부(110), 변환부(120), 양자화부(130), 역양자화부(140), 역변환부(150), 필터링부(160), 복호 픽처 버퍼(DPB: Decoded Picture Buffer)(170), 인터 예측부(180), 인트라 예측부(185) 및 엔트로피 인코딩부(190)를 포함하여 구성될 수 있다.
- [0025] 영상 분할부(110)는 인코더(100)에 입력된 입력 영상(또는, 픽처, 프레임)를 하나 이상의 처리 유닛으로 분할할 수 있다. 예를 들어, 상기 처리 유닛은 코딩 트리 유닛(CTU: Coding Tree Unit), 코딩 유닛(CU: Coding Unit), 예측 유닛(PU: Prediction Unit) 또는 변환 유닛(TU: Transform Unit)일 수 있다.
- [0026] 인코더(100)는 입력 영상 신호에서 인터 예측부(180) 또는 인트라 예측부(185)로부터 출력된 예측 신호를 감산하여 잔여 신호(residual signal)를 생성할 수 있고, 생성된 잔여 신호는 변환부(120)로 전송된다.
- [0027] 변환부(120)는 잔여 신호에 기반하여 변환을 수행하여 변환 계수(transform coefficient)를 생성할 수 있다. 예를 들어, 변환 기법은 DCT(Discrete Cosine Transform), DST(Discrete Sine Transform), KLT(Karhunen-Loeve Transform), GBT(Graph-Based Transform), 또는 CNT(Conditionally Non-linear Transform) 중 적어도 하나를 포함할 수 있다. 보다 구체적으로, 4×4 내지 32×32 크기의 정수 기반 DCT가 사용될 수 있으며, 4×4, 8×8, 16×16, 32×32 변환이 이용될 수 있다. GBT는 픽셀 간의 관계 정보를 그래프로 표현한다고 할 때 이 그래프로부터 얻어진 변환을 의미한다. CNT는 이전에 복원된 모든 픽셀(all previously reconstructed pixel)를 이용하여 예측 신호를 생성하고 그에 기초하여 획득되는 변환을 의미한다. 또한, 변환 과정은 정사각형의 동일한 크기를 갖는 픽셀 블록에 적용될 수도 있고, 정사각형이 아닌 가변 크기의 블록에도 적용될 수 있다.
- [0028] 양자화부(130)는 변환 계수를 양자화하여 엔트로피 인코딩부(190)로 전송하고, 엔트로피 인코딩부(190)는 양자화된 신호를 엔트로피 코딩하여 비트스트림으로 출력할 수 있다. 예를 들어, 엔트로피 인코딩은 고정 길이 코딩(fixed length coding, FLC), 가변 길이 코딩(variable length coding, VLC), 산술 코딩(arithmetic coding)을 기반으로 수행될 수 있다. 보다 구체적으로, 산술 부호화를 기반으로 한 문맥 기반 적응적 이진 산술 코딩(context adaptive binary arithmetic coding, CABAC), 가변 길이 코딩을 기반으로 한 Exp-Golomb 코딩, 및 고정 길이 코딩이 적용될 수 있다.
- [0029] 양자화부(130)로부터 출력된 양자화된 신호는 예측 신호를 생성하기 위해 이용될 수 있다. 예를 들어, 양자화된 신호는 루프 내의 역양자화부(140) 및 역변환부(150)를 통해 역양자화 및 역변환을 적용함으로써 잔여 신호를 복원할 수 있다. 복원된 잔여 신호를 인터 예측부(180) 또는 인트라 예측부(185)로부터 출력된 예측 신호에 더함으로써 복원 신호가 생성될 수 있다.
- [0030] 필터링부(160)는 복원 신호에 필터링을 적용하여 이를 재생 장치로 출력하거나 복호 픽처 버퍼(170)에 전송한다. 필터링을 위해 예를 들어 디블록킹(deblocking) 필터, 샘플 적응적 오프셋(sample adaptive offset, SAO) 필터가 적용될 수 있다. 복호 픽처 버퍼(170)에 전송된 필터링된 신호는 인터 예측부(180)에서 참조 픽처로 사용될 수 있다. 이처럼, 필터링된 픽처를 인터 예측 모드에서 참조 픽처로 이용함으로써 화질 뿐만 아니라 부호화 효율도 향상시킬 수 있다.
- [0031] 복호 픽처 버퍼(170)는 필터링된 픽처를 인터 예측부(180)에서의 참조 픽처로 사용하기 위해 저장할 수 있다.
- [0032] 인터 예측부(180)는 복원 픽처를 참조하여 시간적 중복성 및/또는 공간적 중복성을 제거하기 위해 시간적 예측 및/또는 공간적 예측을 수행한다. 이때, 인터 예측 모드에서 전송되는 움직임 정보의 양을 줄이기 위해 주변 블록과 현재 블록 간의 움직임 정보의 상관성에 기초하여 움직임 정보를 예측할 수 있다.
- [0033] 인트라 예측부(185)는 현재 블록의 주변에 있는 샘플들을 참조하여 현재 블록을 예측할 수 있다. 인트라 예측부(185)는 인트라 예측을 수행하기 위해 다음과 같은 과정을 수행할 수 있다. 먼저, 예측 신호를 생성하기 위해 필요한 참조 샘플을 준비할 수 있다. 그리고, 준비된 참조 샘플을 이용하여 예측 신호를 생성할 수 있다. 이후, 예측 모드를 부호화하게 된다. 이때, 참조 샘플은 참조 샘플 패딩 및/또는 참조 샘플 필터링을 통해 준비될 수 있다. 참조 샘플은 예측 및 복원 과정을 거쳤기 때문에 양자화 에러가 존재할 수 있다. 따라서, 이러한 에러를 줄이기 위해 인트라 예측에 이용되는 각 예측 모드에 대해 참조 샘플 필터링 과정이 수행될 수 있다.

- [0034] 인터 예측부(180) 또는 인트라 예측부(185)를 통해 생성된 예측 신호는 복원 신호를 생성하기 위해 이용되거나 잔여 신호를 생성하기 위해 이용될 수 있다.
- [0035] 도 2는 본 발명이 적용되는 실시예로서, 비디오 신호의 디코딩이 수행되는 디코더의 개략적인 블록도를 나타낸다.
- [0036] 도 2를 참조하면, 디코더(200)는 엔트로피 디코딩부(210), 역양자화부(220), 역변환부(230), 필터링부(240), 복호 픽처 버퍼(DPB: Decoded Picture Buffer Unit)(250), 인터 예측부(260) 및 인트라 예측부(265)를 포함하여 구성될 수 있다.
- [0037] 그리고, 디코더(200)를 통해 출력된 복원 영상 신호는 재생 장치를 통해 재생될 수 있다.
- [0038] 디코더(200)는 도 1의 인코더(100)로부터 출력된 신호를 수신할 수 있고, 수신된 신호는 엔트로피 디코딩부(210)를 통해 엔트로피 디코딩될 수 있다.
- [0039] 역양자화부(220)에서는 양자화 스텝 사이즈 정보를 이용하여 엔트로피 디코딩된 신호로부터 변환 계수(transform coefficient)를 획득한다.
- [0040] 역변환부(230)에서는 역양자화된 변환 계수에 기반하여 역변환을 수행하여 잔여 신호를 획득하게 된다.
- [0041] 획득된 잔여 신호를 인터 예측부(260) 또는 인트라 예측부(265)로부터 출력된 예측 신호에 더함으로써 복원 신호가 생성된다.
- [0042] 필터링부(240)는 복원 신호에 필터링을 적용하여 이를 재생 장치로 출력하거나 복호 픽처 버퍼부(250)에 전송한다. 복호 픽처 버퍼부(250)에 전송된 필터링된 신호는 인터 예측부(260)에서 참조 픽처로 사용될 수 있다.
- [0043] 본 명세서에서, 인코더(100)의 필터링부(160), 인터 예측부(180) 및 인트라 예측부(185)에서 설명된 실시예들은 각각 디코더의 필터링부(240), 인터 예측부(260) 및 인트라 예측부(265)에도 동일하게 적용될 수 있다.
- [0044] 도 3은 코딩 유닛의 분할 구조를 설명하기 위한 도면이다.
- [0045] 인코더 및 디코더는 하나의 영상(또는 픽처)을 사각형 형태의 코딩 트리 유닛(CTU: Coding Tree Unit) 단위로 분할하고 CTU 단위로 인코딩 및 디코딩을 수행할 수 있다.
- [0046] 하나의 CTU는 쿼드트리(quadtree, 이하 'QT'라 함) 구조에 기반하여 분할될 수 있다. 예를 들어, 하나의 CTU는 정사각형 형태를 가지면서 각 변의 길이가 절반씩 감소하는 4개의 유닛으로 분할될 수 있다. 이러한 QT 구조의 분할은 재귀적으로 수행될 수 있다.
- [0047] 도 3을 참조하면, QT의 루트 노드(root node)는 CTU와 관련될 수 있다. QT는 리프 노드(leaf node)에 도달할 때까지 분할될 수 있고, 리프 노드는 코딩 유닛(CU: Coding Unit)으로 지칭될 수 있다. CTU는 루트 노드(root node)에 해당되고, 가장 작은 깊이(depth)(즉, 레벨 0) 값을 가진다. 입력 영상의 특성에 따라 CTU가 분할되지 않을 수도 있으며, 이 경우 CTU는 CU에 해당된다.
- [0048] 도 3의 예에서, CTU는 QT 형태로 분할될 수 있으며, 그 결과 레벨 1의 깊이를 가지는 하위 노드들이 생성될 수 있다. 그리고, 레벨 1의 깊이를 가지는 하위 노드에서 더 이상 분할되지 않은 노드(즉, 리프 노드)는 CU에 해당한다. 예를 들어, 도 3(b)에서 노드 a, b 및 j에 대응하는 CU(a), CU(b), CU(j)는 CTU에서 한 번 분할되었으며, 레벨 1의 깊이를 가진다.
- [0049] 하나의 CU에 대하여, 해당 CU이 분할 되는지 여부를 나타내는 정보가 디코더에 전달될 수 있다. 예를 들어, 상기 정보는 분할 플래그로 정의될 수 있으며, 선택스 정보 "split_cu_flag"로 표현될 수 있다. 상기 분할 플래그는 최소 크기의 CU를 제외한 모든 CU에 포함될 수 있다. 예를 들어, 상기 분할 플래그의 값이 '1'이면 해당 CU는 다시 4개의 CU로 나누어지고, 상기 분할 플래그의 값이 '0'이면 해당 CU는 더 이상 나누어지지 않고 해당 CU에 대한 코딩 과정이 수행될 수 있다.
- [0050] 도 3의 예에서는 CU의 분할 과정에 대해 예로 들어 설명하였으나, 변환을 수행하는 기본 단위인 변환 유닛(TU: Transform Unit)의 분할 과정에 대해서도 상술한 QT 구조를 적용할 수 있다.
- [0051] TU는 코딩하려는 CU로부터 QT 구조로 계층적으로 분할될 수 있다. 예를 들어, CU는 변환 유닛(TU)에 대한 트리의 루트 노드(root node)에 해당될 수 있다.
- [0052] TU는 QT 구조로 분할되므로 CU로부터 분할된 TU는 다시 더 작은 하위 TU로 분할될 수 있다. 예를 들어, TU의 크

기는 32x32, 16x16, 8x8, 4x4 중 어느 하나로 정해질 수 있으나, 본 발명은 이에 한정되지 않으며, 고해상도 영상일 경우, TU의 크기는 더 커지거나 다양해질 수 있다.

- [0053] 하나의 TU에 대하여, 해당 TU가 분할되는지 여부를 나타내는 정보가 디코더에 전달될 수 있다. 예를 들어, 상기 정보는 분할 변환 플래그로 정의될 수 있으며, 선택스 정보 "split_transform_flag"로 표현될 수 있다.
- [0054] 도 4는 코딩 유닛의 분할 구조 중 쿼드트리-바이너리 트리를 설명하기 위한 도면이다.
- [0055] 인코더 및 디코더는 하나의 영상(또는 픽처)을 사각형 형태의 코딩 트리 유닛(CTU: Coding Tree Unit) 단위로 분할하고 CTU 단위로 인코딩 및 디코딩을 수행할 수 있다.
- [0056] 도 3의 예와 비교하여, 도 4의 예에서 하나의 CTU는 쿼드트리 및 바이너리 트리(binary tree, BT) 구조에 기반하여 분할될 수 있다. 예를 들어, 하나의 CTU는 정사각형 형태를 가지면서 각 변의 길이가 절반씩 감소하는 4개의 유닛으로 분할하거나 직사각형 형태를 가지면서 너비 또는 높이 길이가 절반씩 감소하는 2개의 유닛으로 분할할 수 있다. 이러한 QT BT구조의 분할은 재귀적으로 수행될 수 있다.
- [0057] 도 4를 참조하면, QT의 루트 노드(root node)는 CTU와 관련될 수 있다. QT는 QT 리프 노드(QT leaf node)에 도달할 때까지 분할될 수 있고, QT의 리프노드는 BT로 분할할 수 있으며 BT 리프노드에 도달할 때까지 분할 될 수 있다.
- [0058] 도 4의 예에서, CTU는 루트 노드(root node)에 해당되고, 가장 작은 깊이(depth)(즉, 레벨 0) 값을 가진다. 입력 영상의 특성에 따라 CTU가 분할되지 않을 수도 있으며, 이 경우 CTU는 CU에 해당된다.
- [0059] CTU는 QT 형태로 분할될 수 있으며 QT 리프노드는 BT 형태로 분할될 수 있다. 그 결과 레벨 n의 깊이를 가지는 하위 노드들이 생성될 수 있다. 레벨 n의 깊이를 가지는 하위 노드에서 더 이상 분할되지 않은 노드(즉, 리프 노드)는 CU에 해당한다.
- [0060] 하나의 CU에 대하여, 해당 CU가 분할되는지 여부를 나타내는 정보가 디코더에 전달될 수 있다. 예를 들어, 상기 정보는 분할 플래그로 정의될 수 있으며, 선택스 정보 "split_cu_flag"로 표현될 수 있다. 또한, QT 리프노드에서 BT로 분할되는지 여부를 나타내는 정보가 디코더에 전달될 수 있다. 예를 들어, 상기 정보는 BT 분할 플래그로 정의될 수 있으며, 선택스 정보 "bt_split_flag"로 표현될 수 있다. 추가적으로 bt_split_flag에 의하여 BT로 분할되는 경우, 절반 크기의 너비를 가지는 직사각형 또는 절반 크기의 높이를 가지는 직사각형 형태로 분할되도록 BT 분할 모양이 디코더에 전달될 수 있다. 예를 들어, 상기 정보는 BT 분할 모드로 정의될 수 있으며, 선택스 정보 "bt_split_mode"로 표현될 수 있다.
- [0061] 도 5는 인트라 예측 모드를 예시한다.
- [0062] 인트라 예측 모드는 값에 따라 다양한 예측 모드를 나타낸다. 예를 들어, 인트라 예측 모드의 값은 표 1에 예시된 바와 같이 인트라 예측 모드와 대응될 수 있다.

표 1

Intra prediction mode	Associated name
0	INTRA_PLANAR
1	INTRA_DC
2 ... 34	INTRA_ANGULAR2 ... INTRA_ANGULAR34

- [0063]
- [0064] 표 1에서 INTRA_PLANAR는 인트라 평면 예측 모드(intra planar prediction mode)를 나타내며, 현재 블록에 인접한 상단(upper) 이웃 블록의 복원 샘플(reconstructed sample), 좌측(left) 이웃 블록의 복원 샘플, 좌측 하단(lower-left) 이웃 블록의 복원 샘플, 우측 상단(right-upper) 이웃 블록의 복원 샘플에 대해 보간을 수행하여 현재 블록의 예측값을 획득하는 모드를 나타낸다. INTRA_DC는 인트라 DC(Direct Current) 예측 모드를 나타내며, 좌측 이웃 블록의 복원 샘플들과 상단 이웃 블록의 복원 샘플들의 평균을 이용하여 현재 블록의 예측값을 획득하는 모드를 나타낸다. INTRA_ANGULAR2 내지 INTRA_ANGULAR34는 인트라 각도 예측 모드(intra angular prediction mode)를 나타내며, 현재 블록 내의 현재 샘플에 대해 특정 각도의 방향에 위치한 이웃 블록의 복원 샘플을 이용하여 현재 샘플의 예측값을 구하는 모드를 나타낸다(예, 도 5 참조). 특정 각도의 방향에 실제 샘플

이 존재하지 않는 경우 이웃 복원 샘플들에 대해 보간 또는 패딩을 수행하여 해당 방향에 대한 가상 샘플을 생성하여 예측 샘플을 구할 수 있다.

- [0065] 인트라 예측 모드는 코딩 블록 별로 유도할 수 있지만, 인트라 예측은 코딩 블록 또는 변환 블록 단위로 수행될 수 있다. 인트라 예측을 위해 현재 픽처 내에서 현재 블록의 이웃 블록 내에 존재하는 복원 샘플을 참조할 수 있으며, 인트라 예측을 위해 참조되는 샘플을 참조 샘플(reference sample)이라고 지칭한다. 인트라 예측에서는 유도되는 예측 모드에 기반하여 현재 블록에 대한 예측을 수행한다. 예측 모드에 따라 예측에 사용되는 참조 샘플과 구체적인 예측 방법이 달라질 수 있다.
- [0066] 인코더와 디코더는 현재 블록의 주변 샘플들(neighboring samples)이 예측에 사용될 수 있는지 확인하고, 예측에 사용할 참조 샘플들을 구성할 수 있다. 예를 들어, 현재 블록의 크기가 $nS \times nS$ 인 경우, 인트라 예측에서 현재 블록의 주변 샘플들은 현재 블록의 좌측(left)(또는 좌측 경계)에 인접한 nS 개의 참조 샘플, 현재 블록의 좌측 하단(bottom-left)에 이웃하는 nS 개의 참조 샘플, 현재 블록의 상단(top)(또는 상단 경계)에 인접한 nS 개의 참조 샘플, 우측 상단(top-right)에 이웃하는 nS 개의 참조 샘플, 현재 블록의 좌측 상단(top-left)에 이웃하는 1개 샘플이 참조 샘플로서 이용될 수 있다. 만일 현재 처리 블록의 주변 샘플들 중 일부가 이용 가능하지 않은 경우, 인코더와 디코더는 이용 가능한 샘플들로 이용 가능하지 않은 샘플들에 기반하여 보간 또는 패딩을 수행하여 예측에 사용할 참조 샘플들을 구성할 수 있다. 인트라 예측 모드에 기반하여 참조 샘플의 필터링이 수행될 수 있다.
- [0067] 인코더와 디코더는 인트라 예측 모드와 참조 샘플들에 기반하여 현재 블록에 대한 예측값을 생성할 수 있다. 구체적으로, 인코더는 비트율-왜곡(Rate-Distortion, RD) 최적화에 기반하여 참조 샘플들 또는 필터링된 참조 샘플을 이용하여 인트라 예측 모드를 결정하고, 결정된 인트라 예측 모드를 나타내는 신택스 정보를 비트스트림으로 인코딩하며, 결정된 인트라 예측 모드에 기반하여 현재 블록에 대한 예측값을 생성하고, 생성된 예측값을 이용하여 현재 블록을 인코딩할 수 있다. 디코더는 인트라 예측 모드와 참조 샘플들에 기반하여 현재 블록에 대한 예측값을 생성한 후 생성된 예측값에 기반하여 현재 블록을 복원할 수 있다. 즉, 디코더는 인트라 예측 모드 유도 단계에서 유도된 인트라 예측 모드와 참조 샘플 구성 단계와 참조 샘플 필터링 단계를 통해 획득한 참조 샘플들에 기반하여 현재 블록에 대한 예측값을 생성할 수 있다.
- [0068] 도 6 및 도 7은 인트라 예측 모드의 시그널링 방법을 예시한다.
- [0069] 일반적으로 영상이 블록들로 분할되면, 코딩하려는 현재 블록과 이웃 블록은 비슷한 영상 특성을 가질 수 있다. 인트라 예측 모드의 경우, 현재 블록과 이웃 블록은 서로 동일하거나 비슷한 인트라 예측 모드를 가질 확률이 높다. 따라서, 인코더 또는 디코더는 현재 블록의 예측 모드를 인코딩 또는 유도하기 위해 이웃 블록의 예측 모드를 이용할 수 있다.
- [0070] 먼저, 인코더는 이웃 블록이 인트라 코딩된 경우, 이웃 블록의 예측 모드를 확인 또는 유도할 수 있다(S610). 예를 들어, 좌측 이웃 블록의 예측 모드와 상단 이웃 블록의 예측 모드를 기초로 현재 블록의 예측 모드를 결정할 수 있고, 이때 해당 이웃 블록의 예측 모드를 MPM(Most Probable Mode)으로 결정할 수 있다. MPM이라 함은 인트라 예측 모드 코딩시 현재 블록과 이웃 블록의 유사성을 고려하여 코딩 효율을 향상시키기 위해 이용되는 모드를 의미할 수 있다. MPM을 결정하는 것은, MPM(most probable modes) 후보(또는, MPM 리스트)를 리스트 업(list up)한다고 표현할 수도 있다.
- [0071] 인코더는 좌측 이웃 블록의 예측 모드와 상단 이웃 블록의 예측 모드가 동일한지 여부를 확인할 수 있다(S620).
- [0072] 만일 좌측 이웃 블록의 예측 모드와 상단 이웃 블록의 예측 모드가 동일하지 않은 경우, 첫 번째 MPM은 좌측 이웃 블록의 예측 모드로 설정될 수 있고, 두 번째 MPM은 상단 이웃 블록의 예측 모드로 설정될 수 있으며, 세 번째 MPM은 인트라 평면 모드, 인트라 DC 모드, 또는 인트라 수직 모드 중 어느 하나로 설정될 수 있다(S630).
- [0073] 만일 좌측 이웃 블록의 예측 모드와 상단 이웃 블록의 예측 모드가 동일한 경우, 인코더는 좌측 이웃 블록의 예측 모드가 2보다 작은지 여부를 확인할 수 있다(S640).
- [0074] 만약 좌측 이웃 블록의 예측 모드가 2보다 작은 경우, 첫 번째 MPM은 인트라 평면 모드로 설정될 수 있고, 두 번째 MPM은 인트라 DC 모드로 설정될 수 있으며, 세 번째 MPM은 인트라 수직 모드로 설정될 수 있다(S650).
- [0075] 만약 좌측 이웃 블록의 예측 모드가 2보다 작은 않은 경우, 첫 번째 MPM은 좌측 이웃 블록의 예측 모드로 설정될 수 있고, 두 번째 MPM은 (좌측 이웃 블록의 예측 모드-1)로 설정될 수 있으며, 세 번째 MPM은 (좌측 이웃 블록의 예측 모드+1)로 설정될 수 있다(S660).

[0076] 인코더는 현재 블록에 적용될 최적의 인트라 예측 모드가 앞서 구성된 MPM 후보 내에 속하는지 판단할 수 있다. 만약 현재 블록의 인트라 예측 모드가 MPM 후보에 속하는 경우, 인코더는 MPM 플래그와 MPM 인덱스를 비트스트림으로 인코딩할 수 있다.

[0077] 도 7을 참조하면, S710 단계에서 인코더/디코더는 MPM 플래그 정보를 인코딩/디코딩할 수 있다. MPM 플래그 정보는 현재 블록의 인트라 예측 모드가 현재 블록의 주변 인트라 예측된 블록으로부터 유도(즉, 현재 블록의 인트라 예측 모드가 MPM 내 속함)되는지 여부를 지시할 수 있다. 예를 들어, MPM 플래그 정보는 예를 들어 prev_intra_luma_pred_flag로 지칭될 수 있으며, MPM 플래그 정보가 1의 값을 가지는 경우 현재 블록의 인트라 예측 모드가 현재 블록의 (인트라 예측으로 코딩된) 이웃 블록으로부터 유도됨을 지시할 수 있고, MPM 플래그 정보가 0의 값을 가지는 경우 현재 블록의 인트라 예측 모드가 현재 블록의 (인트라 예측으로 코딩된) 이웃 블록으로부터 유도되지 않음을 지시할 수 있다. 혹은 MPM 플래그 정보의 값은 반대로 설정될 수도 있으며, 혹은 MPM 플래그 정보의 값은 0과 1이 아닌 다른 값으로 설정될 수도 있다.

[0078] 만일 MPM 플래그 정보가 현재 블록의 인트라 예측 모드가 이웃 블록으로부터 유도됨을 지시하는 경우, 인코더/디코더는 MPM 인덱스 정보를 인코딩/디코딩할 수 있다. MPM 인덱스 정보는 MPM 후보 리스트에 포함된 MPM 후보 중에서 현재 블록의 인트라 예측 모드로서 어떠한 MPM 모드가 적용되는지를 나타낼 수 있다. 예를 들어, S630 단계와 같이 MPM 후보 리스트가 구성되었을 때 MPM 인덱스 정보는 0 내지 2 중 하나의 값을 가질 수 있고, MPM 인덱스 정보가 0의 값을 가지는 경우 MPM 후보 리스트 중에서 좌측 이웃 블록의 인트라 예측 모드를 현재 블록의 인트라 예측 모드로 결정/적용할 수 있다. 다른 예로, S650 단계와 같이 MPM 후보 리스트가 구성되었을 때 MPM 인덱스 정보는 0 내지 2 중 하나의 값을 가질 수 있고, MPM 인덱스 정보가 1인 경우 MPM 후보 리스트 중에서 인트라 DC 모드를 현재 블록의 인트라 예측 모드로 결정/적용할 수 있다. MPM 인덱스 정보는 예를 들어 mpm_idx로 지칭될 수 있다.

[0079] 반면, 현재 블록의 인트라 예측 모드가 MPM 후보에 속하지 않는 경우(또는 MPM 플래그 정보가 현재 블록의 인트라 예측 모드가 이웃 블록으로부터 유도되지 않음을 지시하는 경우), 인코더/디코더는 현재 블록의 인트라 예측 모드 정보를 비트스트림으로/으로부터 인코딩/디코딩할 수 있다. 현재 블록의 인트라 예측 모드 정보는 예를 들어 rem_intra_luma_pred_mode로 지칭될 수 있다.

[0080] 인코더는 MPM 플래그 정보(예, prev_intra_luma_pred_flag), MPM 인덱스 정보(예, mpm_idx), 현재 블록의 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)에 기반하여 현재 블록의 인트라 예측 모드를 비트스트림을 통해 시그널링할 수 있고, 디코더는 MPM 플래그 정보(예, prev_intra_luma_pred_flag), MPM 인덱스 정보(예, mpm_idx), 현재 블록의 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)에 기반하여 현재 블록의 인트라 예측 모드를 비트스트림으로부터 획득할 수 있다.

[0081] 도 8은 확장된 인트라 예측 모드를 예시한다.

[0082] 고해상도 영상의 인트라 부호화 및 더 정확한 예측을 위해 35가지 인트라 예측 모드를 67가지 인트라 예측 모드로 확장할 수 있다. 도 8에서 점선으로 나타난 화살표는 새로 추가된 32가지 인트라 각도 예측 모드를 나타내며, 예를 들어 확장된 인트라 예측 모드의 값은 표 2에 예시된 바와 같이 인트라 예측 모드와 대응될 수 있다.

표 2

Intra prediction mode	Associated name
0	INTRA_PLANAR
1	INTRA_DC
2 ... 66	INTRA_ANGULAR2 ... INTRA_ANGULAR66

[0083]

[0084] 확장된 인트라 예측 모드에서 인트라 평면 예측 모드(INTRA_PLANAR) 및 인트라 DC 예측 모드(INTRA_DC)는 기존의 인트라 평면 예측 모드 및 인트라 DC 예측 모드와 동일하다(예, 표 1 및 관련 설명 참조). 새로 추가된 32가지 방향성 모드는 모든 블록 크기에서 적용될 수 있고, 휘도 성분과 색차 성분의 인트라 부호화에 적용될 수 있다.

- [0085] 도 9는 확장된 인트라 예측 모드를 위한 MPM 후보를 예시한다.
- [0086] 인트라 예측 모드의 수가 67개로 증가함에 따라 인트라 예측 모드의 효율적인 부호화를 위해 주변 블록으로부터 유도하는 MPM(Most Probable Mode)의 개수를 기존 3개에서 6개로 증가하고 MPM 리스트를 구성하는 방법도 변경할 수 있다.
- [0087] 6개 MPM 후보에 기반하여 MPM 리스트를 구성하는 방법은 크게 다음 3가지 순서로 진행한다.
- [0088] -주변의 인트라 예측 모드 사용
- [0089] -유도된 인트라 예측 모드 사용
- [0090] -디폴트(default) 인트라 예측 모드 사용
- [0091] 6개 MPM 후보를 포함하는 MPM 리스트는 가장 먼저 주변의 인트라 예측 모드를 사용하여 생성한다. 도 9에서 예시된 바와 같이, 현재 인트라 예측을 수행하고자 하는 블록(즉, 현재 블록)의 주변 5개 블록(AL, A, AR, L, B L)을 탐색하여 인트라 예측 모드를 6개 MPM 후보 리스트에 추가한다. 이때, 중복 검사를 수행하여 동일한 인트라 예측 모드는 제외하고 새로운 인트라 예측 모드를 6개 MPM 후보 리스트에 추가한다. 예를 들어, 주변 블록의 탐색 순서는 L → A → 인트라 평면 예측 모드 → 인트라 DC 예측 모드 → BL → AR → AL 순으로 진행할 수 있다. 만약 주변의 인트라 예측 모드를 사용하여 6개 MPM 후보 리스트를 완성하면 후보 리스트 생성 과정을 종료한다.
- [0092] 만약 주변의 인트라 예측 모드를 사용하여 6개 MPM 후보 리스트를 완성하지 못하면 유도된 인트라 예측 모드를 사용하여 후보 리스트를 구성한다. 유도된 인트라 예측 모드는 이미 후보 리스트 내에 존재하는 인트라 예측 모드에 -1 혹은 +1을 더하여 생성하고, 생성한 유도된 인트라 예측 모드를 후보 리스트에 추가한다. 이때, 역시 중복 검사를 수행하여 동일한 모드는 제외하고 새로운 모드일 경우 후보 리스트에 추가한다.
- [0093] 만약 여전히 후보 리스트를 완성하지 못하면 마지막으로 디폴트 인트라 예측 모드를 사용하여 후보 리스트를 구성한다. 디폴트 인트라 예측 모드의 경우 {Vertical, Horizontal, Intra_Angular2, Diagonal} 모드 중 하나일 수 있다. 확장된 인트라 예측 모드에서 Vertical은 인트라 수직 예측 모드(예, Intra_Angular50)를 나타내고, Horizontal은 인트라 수평 예측 모드(예, Intra_Angular18)를 나타내고, Diagonal은 대각선 방향 인트라 예측 모드(예, Intra_Angular34 또는 Intra_Angular66)를 나타낼 수 있다. 순서대로 중복 검사를 수행하여 후보 리스트에 추가한다.
- [0094] 앞서 설명한 바와 같이, 인코더는 가용한 인트라 예측 모드를 모두 적용하여 현재 블록을 인코딩해보고 비트율-왜곡(Rate-Distortion, RD) 최적화에 기반하여 최적의 인트라 예측 모드를 결정한 후 결정된 인트라 예측 모드에 기반하여 현재 블록을 인코딩할 수 있다. 하지만, 인트라 예측 모드의 종류 및 개수가 증가함에 따라 인코더 측에서 수행해야 할 연산량도 증가하며 이로 인해 인코딩 성능이 저하될 수 있다. 예를 들어, 도 5에 예시된 인트라 예측 모드를 적용되는 경우 인코더는 35번의 인트라 예측을 수행하면 되지만, 도 8에 예시된 인트라 예측 모드를 적용하는 경우 인코더는 67번의 인트라 예측을 수행해야 한다. 따라서, 인트라 예측 모드가 증가할수록 인트라 예측의 정확도를 향상시킬 수 있지만 인코딩에 더 많은 연산과 시간이 소요될 수 있으므로 인코딩 성능을 저하시킬 수 있다.
- [0095] 본 발명에서는 종래 기술에 비해 인코딩 성능을 향상시킬 수 있고 인트라 예측 모드를 시그널링하는데 필요한 선택스 정보를 감소시킴으로써 코딩 효율을 향상시킬 수 있는 방법 및 이에 관한 장치를 제안한다. 본 발명은 정지 영상 또는 동영상 인코딩/디코딩 방법에 관한 것으로서, 화면 내 예측 부호화를 수행할 때 간단한 방법을 사용하여 화면 내 예측 부호화의 복잡도를 감소시키는 것을 기반으로 정지 영상 또는 동영상을 인코딩/디코딩하는 방법 및 이를 지원하는 장치에 관한 것이다.
- [0096] 구체적으로, 본 발명에서는 화면 내 예측(intra prediction) 부호화를 수행할 때, 주변의 참조 샘플(neighboring reference sample) 간의 유사도를 측정하는 방법을 제안한다(아래 방법 1 참조). 보다 구체적으로, 본 발명에서는 주변 샘플 간의 차이(difference), 분산(variance), 예측값(predictor 또는 predicted block)의 변화 등을 사용하여 주변 참조 샘플 간의 유사도를 측정하는 방법을 제안한다.
- [0097] 또한, 본 발명에서는 인트라 예측 부호화를 수행할 때, 주변의 참조 샘플 간의 유사도를 기반으로 후보 예측 모드를 선택하는 방법을 제안한다(아래 방법 2 참조). 보다 구체적으로, 본 발명에서는 주변 참조 샘플 간의 유사도가 큰 경우 미리 정한 특정 모드를 현재 블록의 예측 모드로 사용하여 화면 내 예측 부호화를 수행하는 방법을 제안한다(아래 방법 2-1 참조). 또한, 본 발명에서는 주변 참조 샘플 간의 유사도가 큰 경우 MPM 후보 모드

를 사용하여 현재 블록의 후보 모드를 결정하는 방법을 제안한다(아래 방법 2-2 참조). 또한, 본 발명에서는 주변 참조 샘플 간의 특정 방향으로 유사도가 큰 경우 특정 방향의 모드를 현재 블록의 후보 모드로 결정하는 방법을 제안한다(아래 방법 2-3 참조).

[0098] 이하에서, 인코더 동작을 중심으로 본 발명의 방법들을 설명하지만, 인코더 동작과 디코더 동작은 대응되므로 디코더 측에서도 본 발명의 방법들이 동일/유사하게 수행될 수 있다.

[0099] 방법 1

[0100] 방법 1에서는 현재 처리(예, 부호화 또는 복호화)하고자 하는 블록의 주변 참조 샘플 간의 유사도를 측정하는 방법을 제안한다. 본 명세서에서 현재 블록은 현재 처리(예, 부호화 또는 복호화)하고자 하는 블록을 지칭한다. 제한적이지 않은 예로, 현재 블록은 앞서 설명한 코딩 블록에 해당할 수 있다. 본 명세서에서 주변 참조 샘플은 현재 블록에 인접한 이웃 블록의 샘플로서 현재 블록 이전에 복원된 샘플을 지칭한다. 보다 구체적으로, 본 명세서에서 주변 참조 샘플은 현재 블록을 포함하는 현재 픽처 내에서 현재 블록에 인접한 이웃 블록의 복원 샘플(reconstructed sample)을 지칭한다. 본 발명에서 이용되는 현재 블록에 인접한 이웃 블록은 예를 들어 현재 블록의 좌측 이웃 블록, 좌측 하단 이웃 블록, 좌측 상단 이웃 블록, 상단 이웃 블록, 우측 상단 이웃 블록 중에서 적어도 하나를 포함할 수 있다.

[0101] 도 10은 유사도를 측정하기 위해 사용하는 현재 블록의 주변 참조 샘플을 예시한다. 도 10의 예는 오로지 예시를 위한 것일 뿐 본 발명은 이에 제한되지 않는다.

[0102] 도 10(a)는 현재 블록과 동일한 크기의 참조 샘플을 사용하는 방법을 예시한다. 도 10(a)를 참조하면, 현재 블록의 주변 참조 샘플은 현재 블록의 상단 이웃 블록의 참조 샘플(A, B, C, D), 좌측 상단 이웃 블록의 참조 샘플(E), 좌측 이웃 블록의 참조 샘플(F, G, H, I)을 포함할 수 있다.

[0103] 도 10(b)는 현재 블록보다 큰 크기의 참조 샘플을 사용하는 방법을 예시한다. 도 10(b)를 참조하면, 현재 블록의 주변 참조 샘플은 현재 블록의 상단 이웃 블록의 참조 샘플(A, B, C, D), 현재 블록의 우측 상단 이웃 블록의 참조 샘플(E, F, G, H), 현재 블록의 좌측 상단 이웃 블록의 참조 샘플(I), 현재 블록의 좌측 이웃 블록의 참조 샘플(J, K, L, M), 현재 블록의 좌측 하단 이웃 블록의 참조 샘플(N, O, P, Q)을 포함할 수 있다.

[0104] 만일 참조 샘플들 중에서 일부가 이용가능하지 않은 경우에는 이용가능한 샘플에 기반하여 패딩을 수행함으로써 이용가능하지 않은 참조 샘플들을 생성할 수 있다.

[0105] 본 발명의 방법 1에서는 참조 샘플간의 유사도를 측정하는 방법으로서 1) 샘플 간의 차이(difference)를 이용하는 방법(아래 ‘방법 1-1’ 참조), 2) 샘플 간의 분산(variance)를 이용하는 방법(아래 ‘방법 1-2’ 참조), 3) 주어진 샘플을 통해 생성된 예측 블록들의 변화를 이용하는 방법(아래 ‘방법 1-3’ 참조)을 제안한다. 참조 샘플간의 유사도를 측정하는 방법은 본 특허에서 제안하는 방법 이외에 다양한 방법이 있을 수 있다. 각 방법에 대해 자세히 설명한다.

[0106] 방법 1-1: 참조 샘플 간의 차이(difference)를 이용하는 방법

[0107] 본 발명의 방법 1-1은 주변 참조 샘플 간의 차이를 이용하여 유사도를 측정하는 방법이다. 구체적으로, 방법 1-1은 현재 블록의 주변 참조 샘플들 중 가장 큰 값을 갖는 참조 샘플과 가장 작은 값을 갖는 참조 샘플의 차이 값을 이용하는 방법이다. 보다 구체적으로, 방법 1-1의 경우, 참조 샘플 간의 유사도는 현재 블록의 주변 참조 샘플들 중 가장 큰 값을 갖는 참조 샘플과 가장 작은 값을 갖는 참조 샘플의 차이 값으로서 표현될 수 있다.

[0108] 예를 들어, 도 10(a)의 예에서, 참조 샘플 {A, B, C, D, E, F, G, H, I} 중 가장 큰 값을 갖는 참조 샘플과 가장 작은 값을 갖는 참조 샘플을 각각 선택한 후, 선택된 두 참조 샘플간의 차이 값을 계산한다. 다른 예로, 도 10(b)의 예에서, 참조 샘플 {A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q} 중 가장 큰 값을 갖는 참조 샘플과 가장 작은 값을 갖는 참조 샘플을 각각 선택한 후, 선택된 두 참조 샘플의 차이 값을 계산한다. 이렇게 계산된 차이 값을 미리 정의한 문턱 값(threshold value)과 비교하여 유사도를 측정한다. 예를 들어, 차이 값이 문턱 값보다 작은 경우 유사도는 크다고 판별될 수 있고, 차이 값이 문턱 값보다 큰 경우 유사도는 작다고 판별될 수 있다.

[0109] 방법 1-2: 샘플 간의 분산(variance)을 이용하는 방법

[0110] 본 발명의 방법 1-2는 참조 샘플 값의 분산을 이용하여 유사도를 측정하는 방법이다. 구체적으로, 방법 1-2에서 분산은 참조 샘플의 평균과 각 참조 샘플 간의 차이 값의 거듭제곱의 평균으로 계산할 수 있다. 일 예로, 방법

1-2에서 분산은 수학적 식 1을 이용하여 계산할 수 있다.

[0111] [수학적 식 1]

[0112]
$$S^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2$$

[0113] 수학적 식 1에서, S^2 은 참조 샘플 값의 분산을 나타내고, N은 참조 샘플의 개수를 나타내고, \bar{x} 는 참조 샘플의 평균을 나타내고, x_i 는 i번째 참조 샘플을 나타낸다. 예를 들어, 도 10(a)의 경우 N=9 (A~I)이고, 도 10(b)의 경우 N=17 (A~Q)이다. 각각 계산된 분산 값을 미리 정의된 문턱 값과 비교하여 유사도를 측정한다. 예를 들어, 분산 값이 문턱 값보다 작은 경우 유사도는 크다고 판별될 수 있고, 분산 값이 문턱 값보다 큰 경우 유사도는 작다고 판별될 수 있다.

[0114] 도 11은 본 발명에 따라 인트라 예측 모드를 결정하는 방법의 순서도를 예시한다. 구체적으로, 도 11은 본 발명의 방법 1-1 및 방법 1-2에 따라 측정된 참조 샘플 간의 유사도를 기반으로 인트라 예측 모드를 결정하는 방법의 순서도를 나타낸다.

[0115] 도 11을 참조하면, 본 발명의 방법 1-1에 따라 샘플 간의 차이를 이용하여 혹은 본 발명의 방법 1-2에 따라 샘플 간의 분산을 이용하여 유사도를 계산한다(S1110). S1110 단계에서 계산한 유사도가 각각의 방법을 위해 설정된 문턱 값보다 작으면 본 발명의 방법 2를 사용하여 후보 예측 모드를 선택하고 선택한 후보 예측 모드 중에서 비트율 왜곡 최적화(Rate Distortion Optimization, RDO)에 기반하여 최적의 인트라 예측 모드를 결정하고(S1120), 결정된 인트라 예측 모드에 기반하여 화면 내 예측 부호화를 수행한다(S1140).

[0116] 만약 계산한 유사도가 설정된 문턱 값보다 크면 기존의 화면 내 예측 방법을 사용하여 화면 내 예측 부호화를 수행한다(S1130, S1140). 앞서 설명한 바와 같이, 기존의 화면 내 예측 방법에서는 인트라 예측 모드 후보(예, 도 5 및 관련 설명 또는 도 8 및 관련 설명 참조) 중에서 비트율 왜곡 최적화(Rate Distortion Optimization, RDO)에 기반하여 최적의 인트라 예측 모드가 결정되거나, 혹은 근사화 방법을 사용하여 후보 모드를 선택한 후 선택한 인트라 예측 모드 중에서 비트율 왜곡 최적화에 기반하여 최적의 인트라 예측 모드가 결정될 수 있다.

[0117] 방법 3-3: 주어진 샘플을 통해 생성된 예측값들의 변화를 이용하는 방법

[0118] 실제 주어진 샘플들의 값이 유사한 경우 어떠한 화면 내 예측 모드를 사용하던지 실제 생성되는 화면 내 예측값(intra predictor 또는 intra predicted block)은 매우 유사하게 된다. 이 경우 화면 내 예측 모드를 별도로 전송하는 기존의 비디오 부호화 시스템이 효율적이지 못하며, 이를 위해 실제 화면 내 예측 모드를 전송해야 하는가 아니면 추정해야 하는가를 다음과 같은 방법으로 결정한다.

[0119] 모든(또는 미리 정의된 대표) 화면 내 예측 모드들을 통해 화면 내 예측을 수행하고, 각 모드 별로 생성된 예측값들 간의 변화를 이용한다. 이때, 모든 모드에 대해서 화면 내 예측을 수행하여 비교할 수 있지만 이는 복호기 측면에서 계산적 복잡도를 요구하기 때문에 실제로 대표되는 모드들(예: DC, Vertical, Horizontal)에 대해서만 화면 내 예측을 수행할 수 있다. 즉, 미리 정해놓은 대표 모드를 사용하여 생성된 예측값들에 대해 예측값들 간의 차이를 비교하고, 만약 그 차이가 작은 경우(즉, 생성된 예측값이 모두 유사한 경우) 현재 블록에 대한 화면 내 예측 모드를 대표 모드들 중에서 특정 모드로 선택하거나 혹은 대표 모드들에 대해서만 비트율 왜곡 최적화를 수행하여 선택할 수 있다.

[0120] 예를 들어, 대표 모드들(예, DC, Vertical, Horizontal)에 기반하여 방법 1-3을 적용한다고 가정하면, 인트라 DC 예측 모드에 기반하여 생성된 예측값, 인트라 수직 예측 모드에 기반하여 생성된 예측값, 인트라 수평 예측 모드에 기반하여 생성된 예측값 간의 차이를 구하고, 그 차이가 특정 문턱 값보다 작은 경우 대표 모드에 대한 예측값들이 유사하다고 판별할 수 있다. 만일 대표 모드에 대한 예측값이 유사하다고 판별된 경우, 대표 모드들 중에서 특정 인트라 예측 모드를 선택하거나 혹은 대표 모드들에 대해서만 비트율 왜곡 최적화를 수행하여 최적의 인트라 예측 모드를 결정할 수 있다. 만일 대표 모드에 대한 예측값이 유사하지 않다고 판별된 경우(예, 대표 모드들에 대한 예측값들 간의 차이가 특정 문턱 값보다 큰 경우), 기존의 방법을 이용하여 최적의 인트라 예측 모드를 결정할 수 있다.

[0121] 방법 1-4: 문턱 값의 결정

[0122] 앞서 설명한 방법 1-1 내지 방법 1-3에서는 문턱 값의 결과에 따라 화면 내 예측 모드를 전송할지 아니면 추정할지가 결정되고 이는 곧 비트 수의 감소 및 화면 내 예측의 정확도와 연결될 수 있다. 본 발명의 방법 1-4에서

는 방법 1-1 내지 방법 1-3에서 사용될 수 있는 문턱 값을 결정하는 방법을 제안한다.

[0123] 본 발명의 방법 1-1 내지 방법 1-3에서는 문턱 값과의 비교에 기반하여 인트라 예측 모드를 결정하고, 아래에서 설명하는 바와 같이 본 발명의 방법 2에서는 문턱 값과의 비교에 기반하여 비트스트림을 통해 시그널링되는 인덱스 정보의 양이 달라질 수 있다. 예를 들어, 아래에서 설명하는 바와 같이 본 발명의 방법 2에 따르면 문턱 값과의 비교에 기반하여 MPM 플래그 정보(예, prev_intra_luma_pred_flag) 및/또는 MPM 인덱스 정보(예, mpm_idx) 및/또는 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)가 비트스트림에서 생략될 수 있다. 따라서, 문턱 값의 결과에 따라 화면 내 예측 모드를 전송할지 아니면 추정할지가 결정되고 이는 곧 비트 수의 감소 및 화면 내 예측의 정확도와 연결될 수 있다.

[0124] 한편, 양자화 파라미터가 큰 저 비트율 비디오 부호화 환경에서는 양자화 파라미터가 작은 고 비트율 비디오 부호화 환경에 비해 비트 수에 더 민감하여 상대적으로 비트 수에 많은 가중치를 할당하여 부호화를 수행한다. 따라서, 양자화 파라미터가 큰 값을 가지는 경우에는 인트라 예측의 정확도보다 인트라 예측 모드의 시그널링에 필요한 비트 수를 줄이기 위해 문턱 값을 보다 크게 결정할 수 있다. 반면, 양자화 파라미터가 작은 값을 가지는 경우에는 비트 수를 줄이기보다 인트라 예측의 정확도를 올리는 것이 중요할 수 있으므로 문턱 값을 보다 낮게 결정할 수 있다. 따라서, 문턱 값은 양자화 파라미터에 기반하여 결정될 수 있다.

[0125] 일 예로, 본 발명의 문턱 값은 양자화 파라미터(Quantization Parameter, QP)에 비례하여 결정될 수 있다. 보다 구체적인 예로, 본 발명의 문턱 값은 양자화 파라미터에 특정 값을 곱하거나 나눔으로써 결정될 수 있다. 이 경우, 본 발명의 문턱 값은 $(QP * A)$ 또는 (QP / A) 에 기반하여 결정될 수 있고, A는 디코더/인코더 간에 미리 정의된 양의 실수일 수 있다. 혹은, 연산량을 줄이기 위해 시프트 연산을 이용할 수 있으며, 본 발명의 문턱 값은 $(QP \ll B)$ 또는 $(QP \gg B)$ 에 기반하여 결정될 수 있고, \gg 는 시프트 연산을 나타내고, B는 디코더/인코더 간에 미리 정의된 양의 정수일 수 있다.

[0126] 양자화 파라미터와 함께 또는 독립적으로, 현재 블록의 인트라 예측 모드에 할당된 비트 수를 고려하여 문턱 값을 결정할 수 있다.

[0127] 따라서, 본 발명의 방법 1-3에 따른 최적의 문턱 값은 양자화 파라미터 및/또는 화면 내 부호화 모드 비트 수를 고려하여 결정한다.

[0128] 방법 2

[0129] 본 발명의 방법 2에서는 본 발명의 방법 1에 따라 계산한 주변의 참조 샘플 간의 유사도를 기반으로 후보 예측 모드를 선택하는 방법을 제안한다. 방법 2는 방법 1을 사용하여 계산한 유사도가 높은 경우에 적용될 수 있다. 예를 들어, 유사도가 높은 경우는 (1) 주변 참조 샘플 간의 차이 값 < 문턱 값, (2) 주변 참조 샘플 간의 분산 < 문턱 값, (3) 주어진 샘플을 통해 생성한 예측값의 변화 값 < 문턱 값인 경우를 포함할 수 있다. 본 발명의 방법 2에서는 후보 예측 모드를 선택하는 3가지 방법을 설명한다.

[0130] 방법 2-1: 특정 모드를 현재 블록의 예측 모드로 사용하는 방법

[0131] 방법 1에서 언급한 방법으로 유사도를 계산하고 유사도가 큰 경우(방법 1에서 언급한 임의의 방법으로 계산한 값 < 문턱 값) 미리 정한 특정 인트라 예측 모드를 현재 블록의 인트라 예측 모드로 사용하여 화면 내 예측 부호화를 수행하는 방법을 설명한다. 극단적으로 주변 참조 샘플 간의 유사도가 매우 큰 경우 즉, 모든 참조 샘플의 값이 모두 동일한 경우, 기존의 화면 내 예측 모드 중 임의의 인트라 예측 모드를 선택하여 예측값(predictor 또는 predicted block)을 생성해도 모두 동일한 예측값이 생성된다. 이것은 주변 참조 샘플 간의 유사도가 큰 경우 제한된 화면 내 예측 모드를 사용하여 예측값을 생성해도 정확한 예측값을 생성할 수 있다는 것을 의미한다. 따라서, 주변 참조 샘플 간의 유사도가 큰 경우 미리 정해 놓은 대표 예측 모드를 사용하여 화면 내 예측 부호화를 수행한다. 이 때, 대표 후보 예측 모드의 수와 모드는 모두 임의로 결정할 수 있다.

[0132] 도 12는 본 발명의 방법 2-1에 따라 현재 블록의 인트라 예측 모드를 결정하는 방법의 순서도를 예시한다.

[0133] 도 12를 참조하면, 본 발명의 방법 1에서 언급한 방법 중 임의의 방법을 사용하여 유사도를 계산한다(S1210). 유사도를 계산한 값이 문턱 값보다 크면 기존의 방법을 사용하여 최적의 예측 블록을 선택한다(S1240). 만약 계산한 값이 문턱 값보다 작으면 다시 주변 참조 샘플 값의 차이를 비교한다. 주변 참조 샘플 값이 모두 동일하면 특정 하나의 인트라 예측 모드를 최종 인트라 예측 모드로 선택한다(S1220). 만약 주변 참조 샘플 값이 모두 동일하지 않으면 특정 대표 모드 중 비트율 왜곡 최적화 방법을 통해 특정 하나의 인트라 예측 모드를 선택한다(S1230). 예를 들어, 특정 대표 모드는 인트라 DC 예측 모드, 인트라 수직 예측 모드, 인트라 수평 예측 모드를

포함할 수 있으나, 본 발명은 이에 제한되지 않는다. 따라서, 본 발명의 방법 2-1에서는 주변 참조 샘플의 유사도와 함께 주변 참조 샘플의 동일 여부를 추가적으로 판단하여 최종 예측 모드를 선택한다(S1250).

- [0134] 도 12의 예에서는 하나의 문턱 값을 사용하였지만, 도 12에 예시된 방법은 복수의 문턱 값을 사용하는 방식으로 변형될 수 있으며, 변형된 방법은 본 명세서에서 방법 2-1-1로 지칭한다.
- [0135] 본 발명의 방법 2-1-1에서는 다수의 문턱 값을 사용하며 다수의 문턱 값을 단계적으로 적용할 수 있다. 예를 들어, 참조 샘플 간의 유사도가 매우 크면(유사도를 나타내는 값 < 문턱 값1) 특정 하나의 인트라 예측 모드를 사용하고(예, S1220 참조), 참조 샘플 간의 유사도가 크면(문턱 값1 ≤ 유사도를 나타내는 값 < 문턱 값2) 특정 M개의 대표 모드를 사용하여 인트라 예측 모드를 선택하고(예, S1230 참조), 참조 샘플 간의 유사도가 작으면(문턱 값2 ≤ 유사도를 나타내는 값 < 문턱 값3) 특정 N개의 대표 모드를 사용하여 인트라 예측 모드를 선택하고(예, S1230 참조), 참조 샘플 간의 유사도가 매우 작으면(문턱 값3 < 유사도를 나타내는 값) 기존의 방법으로 인트라 예측 모드를 선택할 수 있다(예, S1240 참조). 이 예에서, M < N으로 설정될 수 있다. 특정 M개와 N개의 대표 후보 모드 중 최적의 인트라 예측 모드는 비트율 왜곡 최적화를 통해 결정될 수 있고, 결정된 인트라 예측 모드를 적절한 방법을 사용하여 부호화한다.
- [0136] 본 발명의 방법 2-1-1에서 문턱 값의 개수는 3개로 제한되지 않으며, 방법 2-1-1을 사용할 때 각 문턱 값과 문턱 값의 개수는 모두 임의로 선택할 수 있다.
- [0137] 본 발명의 방법 2-1에 따르면, 결정된 인트라 예측 모드를 부호화/복호화하기 위해 선택스 정보에 대한 비트가 발생하지만 기존의 방법보다 더 적은 정보량을 사용하여 부호화/복호화를 수행할 수 있다. 예를 들어, 본 발명의 방법 2-1에 따라 특정 하나의 인트라 예측 모드를 선택하는 경우(예, S1220 단계 참조), MPM 플래그 정보(예, prev_intra_luma_pred_flag) 및/또는 MPM 인덱스 정보(예, mpm_idx) 및/또는 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)가 비트스트림에서 생략될 수 있으며, 생략되는 선택스 정보에 대응되는 정보량을 비트스트림에서 줄일 수 있으므로 코딩 효율을 향상시킬 수 있다. 이 예에서, 특정 하나의 인트라 예측 모드는 MPM 후보 리스트(예, 도 6 및 관련 설명 또는 도 8 및 관련 설명 참조) 중에서 가장 낮은 인덱스를 가지는 MPM 후보로 결정될 수 있고, 인코더와 디코더는 결정된 특정 하나의 인트라 예측 모드를 현재 블록의 인트라 예측 모드로 사용할 수 있다. 다른 예로, 특정 하나의 인트라 예측 모드는 인코더/디코더 간에 미리 정의된 인트라 예측 모드로 결정될 수 있다.
- [0138] 다른 예로, 본 발명의 방법 2-1에 따라 특정 대표 모드에서 인트라 예측 모드를 선택하는 경우(예, S1230 단계 참조), MPM 플래그 정보(예, prev_intra_luma_pred_flag)와 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)가 비트스트림에서 생략되고, MPM 인덱스 정보(예, mpm_idx)가 비트스트림을 통해 시그널링되며, 이 경우에도 생략되는 선택스 정보에 대응되는 정보량을 비트스트림에서 줄일 수 있으므로 코딩 효율을 향상시킬 수 있다. 일 예로, 본 발명의 방법 2-1에 따라 S1230 단계를 수행하는 경우, 인코더는 MPM 플래그 정보를 비트스트림에 인코딩하는 것을 생략(skip)하고, 특정 대표 모드에 대해 비트율 왜곡 최적화를 수행하여 선택된 인트라 예측 모드를 현재 블록의 인트라 예측 모드로 결정하고, 특정 대표 모드 중에서 결정된 인트라 예측 모드를 지시하는 MPM 인덱스 정보를 비트스트림에 인코딩할 수 있다. 유사하게, 이 예에서 본 발명의 방법 2-1에 따라 S1230 단계를 수행하는 경우, 디코더는 MPM 플래그 정보를 비트스트림으로부터 파싱(parsing)하지 않고, MPM 인덱스 정보를 비트스트림으로부터 파싱하고, 특정 대표 모드 중에서 파싱된 MPM 인덱스 정보가 지시하는 인트라 예측 모드를 현재 블록의 인트라 예측 모드로 결정할 수 있다.
- [0139] 방법 2-2: MPM 후보 모드를 현재 블록의 후보 모드로 사용하는 방법
- [0140] 본 발명의 방법 2-2에서는 본 발명의 방법 1에서 언급한 방법으로 유사도를 계산하고, 유사도가 큰 경우(예, 방법 1에서 언급한 임의의 방법으로 계산한 값 < 문턱 값) MPM 후보 모드를 사용하여 현재 블록의 후보 모드를 결정하는 방법을 설명한다. 방법 2-2는 참조 샘플 간의 유사도가 큰 경우 MPM 후보 모드 중에서 일부 또는 전부를 현재 블록의 예측 후보 모드로 사용하는 방법이다. 방법 2-2에서, MPM 후보 모드 중 몇 개의 모드를 현재 모드를 위한 예측 후보 모드로 사용할지 여부에 따라 전송할 정보량이 결정된다.
- [0141] 예를 들어, MPM 후보 리스트(예, 도 6 및 관련 설명 또는 도 8 및 관련 설명 참조)의 모드 중 가장 낮은 인덱스를 가지는 모드(예, MPM[0]의 모드)만을 후보 모드로 사용하면 인트라 예측 모드를 전송하기 위한 비트가 필요하지 않다. 따라서, MPM 플래그 정보(예, prev_intra_luma_pred_flag), MPM 인덱스 정보(예, mpm_idx), 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)와 같은 인트라 예측 모드를 위한 선택스 정보는 비트스트림을 통해 시그널링되지 않는다. 참조 샘플 간의 유사도가 커서 제안한 방법 2-2를 적용하게 되면, MPM 후보 리스트

의 모드 중 가장 낮은 인덱스를 가지는 모드(예, MPM[0]의 모드)로 예측값을 생성할 수 있다.

- [0142] 다른 예로, 만약 MPM 후보 리스트의 모드 중 2개 또는 그 이상의 모드를 후보 모드로 사용할 경우, 사용되는 후보 모드에 대해 먼저 비트율 왜곡 최적화를 통해 최적의 모드를 결정하고 결정된 최적의 모드를 지시하는 정보가 비트스트림을 통해 시그널링될 수 있다. 이 경우, MPM 플래그 정보(예, prev_intra_luma_pred_flag) 및 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)와 같은 선택스 정보는 비트스트림을 통해 시그널링되지 않고, MPM 후보 리스트 중에서 최적의 모드를 지시하는 MPM 인덱스 정보(예, mpm_idx)가 비트스트림을 통해 시그널링될 수 있다. 예를 들어, MPM 후보 리스트 중에서 가장 낮은 인덱스를 가지는 2개의 모드(예, MPM[0]과 MPM[1] 두 모드)를 후보 모드로 사용하면 먼저 비트율 왜곡 최적화를 통해 최적의 모드를 결정하고 두 모드 중 선택된 모드를 결정하기 위해 하나의 비트가 필요하다. 이와 같이, MPM 후보 리스트의 모드 중 임의의 모드를 선택하여 현재 블록의 후보 모드로 사용하여 예측 블록을 생성할 수 있다.
- [0143] 본 발명의 방법 2-2에서 MPM 후보 리스트 중에서 최종 예측 모드를 선택하기 위해 사용하는 후보 모드의 수는 임의로 결정될 수 있다. 방법 2-2에서 사용되는 후보 모드의 개수는 비트스트림을 통해 시그널링되거나 또는 인코더/디코더 간에 미리 정의된 값일 수 있다.
- [0144] 본 발명의 방법 2-2를 적용할 경우, 기존 방법에 비해 선택스 정보의 시그널링에 필요한 비트 수를 줄일 수 있다. 예를 들어, MPM 플래그 정보(예, prev_intra_luma_pred_flag), MPM 인덱스 정보(예, mpm_idx), 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)와 같은 인트라 예측 모드를 위한 선택스 정보에 필요한 비트 수를 줄일 수 있다. 또한, MPM 후보 리스트 중에서 실제 사용되는 후보 모드의 개수에 따라, 실제 사용되는 후보 모드를 지시하는 인덱스 정보에 필요한 비트 수도 줄일 수 있다. 예를 들어, MPM 후보 리스트가 3개의 MPM 후보 모드를 포함하는 경우 MPM 인덱스 정보의 시그널링을 위해 2 비트가 필요하지만, 본 발명의 방법 2-2에서 2개의 후보 모드를 사용하는 경우 선택된 후보 모드의 시그널링을 위해 1 비트가 필요하므로, 코딩 효율을 향상시킬 수 있다. 또한, 본 발명의 방법 2-2를 적용할 경우, 일부 인트라 예측 모드에 대해서만 비트율 왜곡 최적화를 수행하면 되므로 인코딩 성능을 비약적으로 향상시킬 수 있다.
- [0145] 도 13은 본 발명의 방법 2-2의 순서도를 예시한다. 도 13에서 인코더 동작을 중심으로 설명하지만 도 13의 방법은 디코더에서도 동일/유사하게 수행될 수 있다.
- [0146] S1310 단계에서, 현재 블록의 인트라 예측에 사용되는 참조 샘플(예, 도 10 및 관련 설명 참조) 간의 유사도를 나타내는 값을 획득할 수 있다. 예를 들어, 참조 샘플 간의 유사도를 나타내는 값은 본 발명의 방법 1에서 설명한 방법에 따라 획득될 수 있다.
- [0147] S1310 단계에서 획득한 값과 특정 문턱 값에 기반하여 현재 블록의 인트라 예측 모드를 결정할 수 있다(S1320 단계 또는 S1330 단계 참조). 예를 들어, 특정 문턱 값은 본 발명의 방법 1-4에 따라 결정될 수 있다.
- [0148] 만일 S1310 단계에서 획득한 유사도를 나타내는 값이 특정 문턱 값보다 작은 경우, 인코더는 MPM 후보 리스트 중에서 일부 또는 전부를 포함하는 후보 모드 리스트를 구성하고 구성된 후보 모드 리스트에 대해 비트율 왜곡 최적화를 통해 최적의 모드를 선택한 후(S1320 단계) 선택된 인트라 예측 모드를 현재 블록의 인트라 예측 모드로 결정할 수 있다(S1340 단계).
- [0149] 앞서 설명한 바와 같이, S1320 단계에서 MPM 후보 리스트 중에서 하나의 후보 모드(예, 가장 낮은 인덱스를 가지는 모드)만이 사용되는 경우, 그 하나의 후보 모드가 현재 블록의 인트라 예측 모드로 결정되며, MPM 플래그 정보(예, prev_intra_luma_pred_flag), MPM 인덱스 정보(예, mpm_idx), 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)와 같은 선택스 정보는 비트스트림으로 인코딩되지 않으며, 후보 모드 리스트도 구성할 필요가 없다. 이와 같이 MPM 후보 리스트 중에서 하나의 후보 모드만이 사용되는 경우, 디코더 측에서는 MPM 플래그 정보(예, prev_intra_luma_pred_flag), MPM 인덱스 정보(예, mpm_idx), 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)와 같은 선택스 정보를 비트스트림으로부터 파싱하지 않고, 상기 하나의 후보 모드(예, 가장 낮은 인덱스를 가지는 모드)를 현재 블록의 인트라 예측 모드로 결정할 수 있다.
- [0150] S1320 단계에서 MPM 후보 리스트 중에서 복수의 후보 모드가 사용되는 경우, 인코더는 사용되는 복수의 후보 모드를 포함하여 후보 모드 리스트를 구성하고 사용되는 후보 모드에 대해 비트율 왜곡 최적화를 수행하여 최적의 모드를 선택하고 구성된 후보 모드 리스트 중에서 선택된 인트라 예측 모드를 지시하는 인덱스 정보를 비트스트림으로 인코딩할 수 있다. 이 경우 MPM 플래그 정보(예, prev_intra_luma_pred_flag) 및 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)와 같은 선택스 정보는 비트스트림으로 인코딩되지 않는다. 디코더 측에서는 비트스트림으로부터 인덱스 정보를 파싱하고, 인코더와 마찬가지로 방식으로 MPM 후보 리스트 중에서 사용되는 복

수의 후보 모드를 포함하여 후보 모드 리스트를 구성하고, 구성된 후보 모드 리스트 중에서 파싱된 인덱스 정보가 지시하는 후보 모드를 현재 블록의 인트라 예측 모드로 결정할 수 있다. 디코더는 비트스트림으로부터 MPM 플래그 정보(예, prev_intra_luma_pred_flag)를 파싱하는 동작을 생략할 수 있다.

- [0151] 앞서 설명한 바와 같이, MPM 후보 리스트 중에서 방법 2-2를 위해 사용되는 후보 모드의 개수는 비트스트림을 통해 시그널링되거나 또는 인코더/디코더 간에 미리 정의될 수 있다.
- [0152] 만일 S1310 단계에서 획득한 유사도를 나타내는 값이 특정 문턱 값보다 큰 경우, 기존의 방법을 사용하여 최적의 인트라 예측 모드를 선택할 수 있다(S1330 단계). 예를 들어, 도 5 또는 도 8을 참조하여 설명한 인트라 예측 모드에 대해 비트율 왜곡 최적화를 수행하여 최적의 인트라 예측 모드를 선택하거나, 혹은 근사화 방법을 사용하여 후보 모드를 선택한 후 선택한 인트라 예측 모드 중에서 비트율 왜곡 최적화에 기반하여 최적의 인트라 예측 모드를 선택할 수 있다(예, S1130 참조). 인코더는 선택된 최적의 인트라 예측 모드에 기반하여 도 6 및 도 7을 참조하여 설명된 방법에 따라 MPM 플래그 정보(예, prev_intra_luma_pred_flag), MPM 인덱스 정보(예, mpm_idx), 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)를 결정하고 비트스트림으로 인코딩할 수 있다.
- [0153] S1310 단계에서 획득한 유사도를 나타내는 값이 특정 문턱 값보다 큰 경우, 디코더는 비트스트림으로부터 플래그 정보(예, prev_intra_luma_pred_flag)를 획득하고, 획득한 MPM 플래그 정보가 현재 블록의 인트라 예측 모드가 현재 블록의 이웃 블록으로부터 유도됨을 지시하는 경우 MPM 인덱스 정보(예, mpm_idx)를 획득하고, MPM 후보 리스트 중에서 획득한 MPM 인덱스 정보가 지시하는 모드를 현재 블록의 인트라 예측 모드로 결정할 수 있다. 만일 획득한 MPM 플래그 정보가 현재 블록의 인트라 예측 모드가 현재 블록의 이웃 블록으로부터 유도되지 않음을 지시하는 경우 비트스트림으로부터 인트라 예측 모드 정보(예, rem_intra_luma_pred_mode)를 획득하고, 이에 기반하여 현재 블록의 인트라 예측 모드를 결정할 수 있다.
- [0154] S1320 단계 또는 S1330 단계에서 최적의 인트라 예측 모드가 선택되면, 선택된 인트라 예측 모드에 기반하여 현재 블록을 인코딩할 수 있다. 디코더 측에서는 S1320 단계 또는 S1330 단계에서 설명한 바와 같이 결정된 인트라 예측 모드에 기반하여 현재 블록을 복원할 수 있다.
- [0155] 방법 2-3: 특정 방향의 모드를 현재 블록의 후보 모드로 사용하는 방법
- [0156] 주변 참조 샘플 간의 특정 방향으로 유사도가 큰 경우 특정 방향의 모드를 현재 블록의 후보 모드로 결정하는 방법을 설명한다. 설명의 편의를 위해 도 14의 예를 참조하여 본 발명의 방법 2-3을 설명하지만, 본 발명은 도 14의 예로 제한되지 않으며 본 발명은 다른 경우에도 동일/유사하게 적용될 수 있다.
- [0157] 도 14는 각 방향의 참조 샘플의 유사도가 서로 다른 경우를 예시한다. 도 14의 예에서 설명의 편의를 위해 현재 블록과 동일한 크기의 주변 참조 샘플이 사용된다고 가정하지만, 도 10을 참조하여 설명한 바와 같이 다양한 주변 참조 샘플이 사용될 수 있다. 또한, 도 14는 상단 참조 샘플(A, B, C, D)의 경우 유사도가 작고 좌측 참조 샘플(F, G, H, I)의 경우 유사도가 크다고 가정한다.
- [0158] 도 14를 참조하면, 먼저 본 발명의 방법 1에 기반하여 참조 샘플 간의 차이 혹은 참조 샘플 간의 변화량을 계산 후, 유사도를 측정한다. 만약 유사도 큰 경우(예, 본 발명의 방법 1에 기반하여 계산한 값 < 문턱 값) 다시 좌측 참조 샘플의 유사도와 상단 참조 샘플의 유사도를 각각 계산한다. 계산된 각 방향의 참조 샘플을 비교하여 유사도가 더 큰 쪽을 결정한다. 도 14의 예에서는 단순히 상단 참조 샘플과 좌측 참조 샘플의 유사도를 비교하였는데, 유사도의 비교 영역은 보다 세분화할 수 있다. 예를 들어, 우측 상단, 상단, 좌단, 좌측 하단으로 4분할 할 수도 있고, 우측 상단, 상단, 좌측 상단, 좌측 상단, 좌단, 좌측 하단으로 6분할 할 수도 있다. 이는 예이고 더욱 다양한 방법으로 세분화를 수행할 수 있다.
- [0159] 도 14의 경우를 예로 설명하면 본 발명의 방법 1을 사용하여 참조 샘플 간의 유사도를 계산하고, 계산된 값이 미리 정의한 문턱 값보다 작으면 다시 좌측 참조 샘플의 유사도와 상단 참조 샘플의 유사도를 각각 계산한다. 도 14에서 보듯이 좌측 참조 샘플의 유사도가 크면 좌측 참조 샘플을 사용하여 예측값(predictor 또는 predicted block)을 생성하는 수평 방향 모드(Horizontal mode) 중 임의의 모드를 선택하여도 거의 유사한 예측 블록을 생성할 수 있다. 이와는 반대로 상단 참조 샘플을 사용하여 예측값을 생성하는 수직 방향 모드(Vertical mode) 중 임의의 모드를 선택하게 되면 선택되는 모드의 방향성에 따라 서로 다른 예측 블록을 생성한다. 따라서, 도 14의 예에서는 유사도가 높은 수평 방향의 모드의 수는 줄이는 것이 효율적이다.
- [0160] 예를 들어, 좌측 참조 샘플의 유사도가 매우 큰 경우 해당 방향의 모드인 수평 방향 모드 중 하나의 대표 모드를 사용하여 수평 방향에 대한 예측을 수행한다. 만약 좌측 참조 샘플의 유사도가 큰 경우 해당 방향의 모드인

수평 방향 모드 중 특정 대표 모드들을 사용하여 수평 방향에 대한 예측을 수행한다. 상단 참조 샘플의 유사도를 비교하여 동일한 방법으로 후보 대표 모드의 수를 결정할 수 있다. 이러한 방법을 사용하여 최적의 예측 모드를 결정하기 위해 비트율 왜곡 최적화를 통해 계산할 모드의 수를 줄여 복잡도를 줄일 수 있고, 또한 비트율 왜곡 최적화 관점에서 중복되는 모드의 수를 효율적으로 줄여 전송할 정보량을 줄일 수 있다.

[0161] 도 15는 본 발명이 적용될 수 있는 영상 처리 장치의 블록도를 예시한다. 영상 처리 장치는 영상 신호의 인코딩 장치 및/또는 디코딩 장치를 포함할 수 있다. 예를 들어, 본 발명이 적용될 수 있는 영상 처리 장치는 스마트폰 등과 같은 이동 단말, 랩톱 컴퓨터 등과 같은 휴대용 기기, 디지털 TV, 디지털 비디오 플레이어 등과 같은 가전 제품 등을 포함할 수 있다.

[0162] 메모리(12)는 프로세서(11)의 처리 및 제어를 위한 프로그램을 저장할 수 있고, 부호화된 비트스트림, 복호화된 영상, 제어 정보 등을 저장할 수 있다. 또한, 메모리(12)는 각종 영상 신호를 위한 버퍼로서 활용될 수 있다. 메모리(12)는 ROM(Read Only Memory), RAM(Random Access Memory), EPROM(Erasable Programmable Read Only Memory), EEPROM(Electrically Erasable Programmable Read-Only Memory), 플래쉬(flash) 메모리, SRAM(Static RAM), HDD(Hard Disk Drive), SSD(Solid State Drive) 등과 같은 저장 장치로서 구현될 수 있다.

[0163] 프로세서(11)는 영상 처리 장치 내 각 모듈의 동작을 제어한다. 특히, 프로세서(11)는 본 발명에 따른 인코딩/디코딩을 수행하기 위한 각종 제어 기능을 수행할 수 있다. 프로세서(11)는 컨트롤러(controller), 마이크로 컨트롤러(microcontroller), 마이크로 프로세서(microprocessor), 마이크로 컴퓨터(microcomputer) 등으로도 불릴 수 있다. 프로세서(11)는 하드웨어(hardware) 또는 펌웨어(firmware), 소프트웨어, 또는 이들의 결합에 의해 구현될 수 있다. 하드웨어를 이용하여 본 발명을 구현하는 경우에는, 본 발명을 수행하도록 구성된 ASIC(application specific integrated circuit) 또는 DSP(digital signal processor), DSPD(digital signal processing device), PLD(programmable logic device), FPGA(field programmable gate array) 등이 프로세서(11)에 구비될 수 있다. 한편, 펌웨어나 소프트웨어를 이용하여 본 발명을 구현하는 경우에는 본 발명의 기능 또는 동작들을 수행하는 모듈, 절차 또는 함수 등을 포함하도록 펌웨어나 소프트웨어가 구성될 수 있으며, 본 발명을 수행할 수 있도록 구성된 펌웨어 또는 소프트웨어는 프로세서(11) 내에 구비되거나 메모리(12)에 저장되어 프로세서(11)에 의해 구동될 수 있다.

[0164] 또한, 장치(10)는 네트워크 인터페이스 모듈(network interface module, NIM)(13)을 선택적으로(optionally) 포함할 수 있다. 네트워크 인터페이스 모듈(13)은 프로세서(11)와 동작시 연결(operatively connected)되며, 프로세서(11)는 네트워크 인터페이스 모듈(13)을 제어하여 무선/유선 네트워크를 통해 정보 및/또는 데이터, 신호, 메시지 등을 나르는 무선/유선 신호를 전송 또는 수신할 수 있다. 네트워크 인터페이스 모듈(13)은 예를 들어 IEEE 802 계열, 3GPP LTE(-A), Wi-Fi, ATSC(Advanced Television System Committee), DVB(Digital Video Broadcasting) 등과 같은 다양한 통신 규격을 지원하며, 해당 통신 규격에 따라 제어 정보 및/또는 부호화된 비트스트림과 같은 영상 신호를 송수신할 수 있다. 네트워크 인터페이스 모듈(13)은 필요에 따라 장치에 포함되지 않을 수 있다.

[0165] 또한, 장치(10)는 입출력 인터페이스(14)를 선택적으로(optionally) 포함할 수 있다. 입출력 인터페이스(14)는 프로세서(11)와 동작시 연결(operatively connected)되며, 프로세서(11)는 입출력 인터페이스(14)를 제어하여 제어 신호 및/또는 데이터 신호를 입력받거나 출력할 수 있다. 입출력 모듈(14)은 예를 들어 키보드, 마우스, 터치패드, 카메라 등과 같은 입력 장치와 디스플레이 등과 같은 출력 장치와 연결될 수 있도록 USB(Universal Serial Bus), Bluetooth, NFC(Near Field Communication), 직렬/병렬 인터페이스, DVI(Digital Visual Interface), HDMI(High Definition Multimedia Interface) 등과 같은 규격을 지원할 수 있다.

[0166] 이상에서 설명된 방법들 및 실시예들은 본 발명의 구성요소들과 특징들이 소정 형태로 결합된 것들이다. 각 구성요소 또는 특징은 별도의 명시적 언급이 없는 한 선택적인 것으로 고려되어야 한다. 각 구성요소 또는 특징은 다른 구성요소나 특징과 결합되지 않은 형태로 실시될 수 있다. 또한, 일부 구성요소들 및/또는 특징들을 결합하여 본 발명의 실시예를 구성하는 것도 가능하다. 본 발명의 실시예들에서 설명되는 동작들의 순서는 변경될 수 있다. 어느 실시예의 일부 구성이나 특징은 다른 실시예에 포함될 수 있고, 또는 다른 실시예의 대응하는 구성 또는 특징과 교체될 수 있다. 특허청구범위에서 명시적인 인용 관계가 있지 않은 청구항들을 결합하여 실시예를 구성하거나 출원 후의 보정에 의해 새로운 청구항으로 포함시킬 수 있음은 자명하다.

[0167] 본 발명에 따른 방법 및 실시예는 다양한 수단, 예를 들어, 하드웨어, 펌웨어(firmware), 소프트웨어 또는 그것들의 결합 등에 의해 구현될 수 있다. 하드웨어에 의한 구현의 경우, 본 발명의 일 실시예는 하나 또는 그 이상의 ASIC(application specific integrated circuit), DSP(digital signal processor), DSPD(digital signal

processing device), PLD(programmable logic device), FPGA(field programmable gate array), 프로세서, 콘트롤러, 마이크로 콘트롤러, 마이크로 프로세서 등에 의해 구현될 수 있다.

[0168] 펌웨어나 소프트웨어에 의한 구현의 경우, 본 발명은 이상에서 설명된 기능 또는 동작들을 수행하는 모듈, 절차, 함수 등의 형태를 포함하는 소프트웨어 코드 또는 명령어(instruction)로 구현될 수 있다. 소프트웨어 코드 또는 명령어는 컴퓨터 판독가능한 매체에 저장되어 프로세서에 의해 구동될 수 있으며 프로세서에 의해 구동될 때 본 발명에 따른 동작들을 수행할 수 있다. 상기 컴퓨터 판독가능한 매체는 상기 프로세서 내부 또는 외부에 위치하거나 원격으로 네트워크를 통해 상기 프로세서와 연결될 수 있으며, 상기 프로세서와 데이터를 주고받을 수 있다.

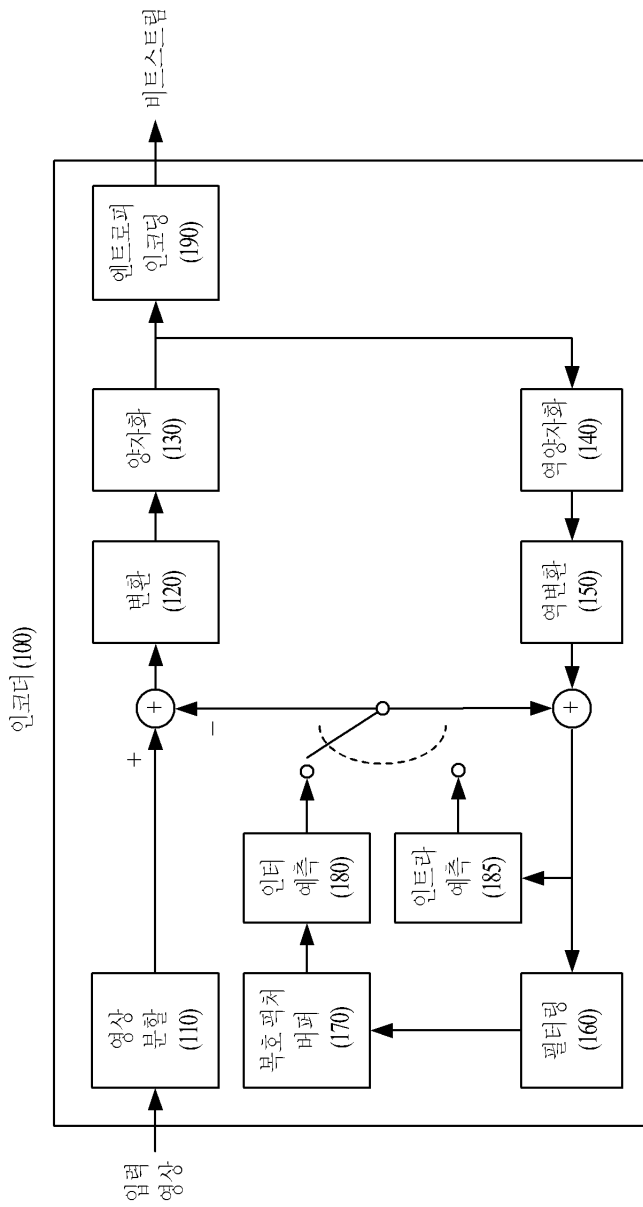
[0169] 본 발명은 본 발명의 특징을 벗어나지 않는 범위에서 다른 특정한 형태로 구체화될 수 있음은 당업자에게 자명하다. 따라서, 상기의 상세한 설명은 모든 면에서 제한적으로 해석되어서는 아니되고 예시적인 것으로 고려되어야 한다. 본 발명의 범위는 첨부된 청구항의 합리적 해석에 의해 결정되어야 하고, 본 발명의 등가적 범위 내에서의 모든 변경은 본 발명의 범위에 포함된다.

산업상 이용가능성

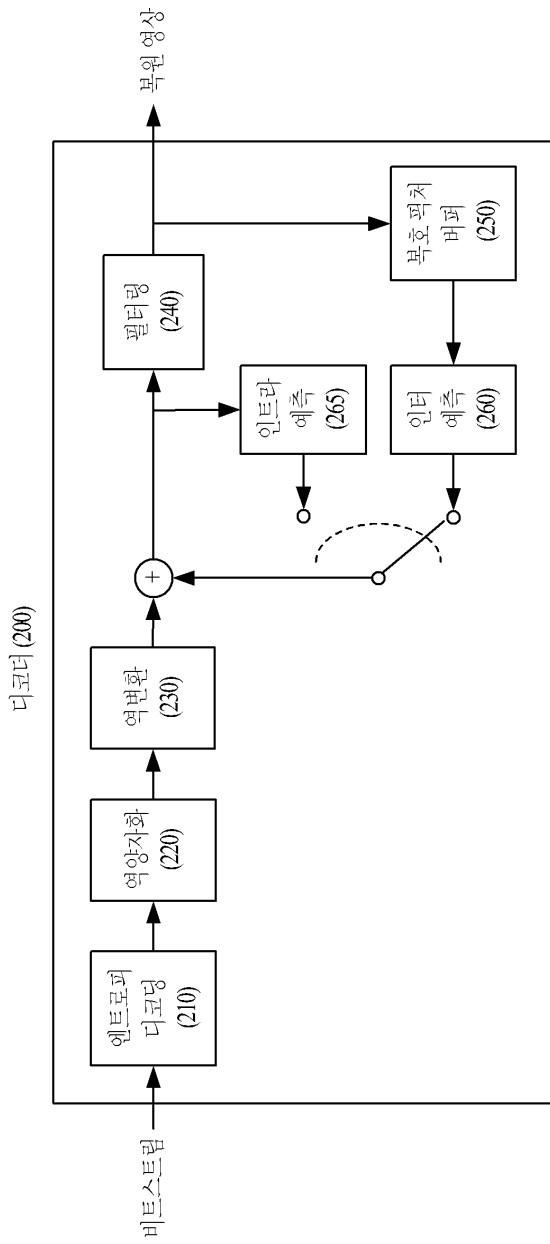
[0170] 본 발명은 디코딩 장치, 인코딩 장치와 같은 영상 처리 장치에 이용될 수 있다.

도면

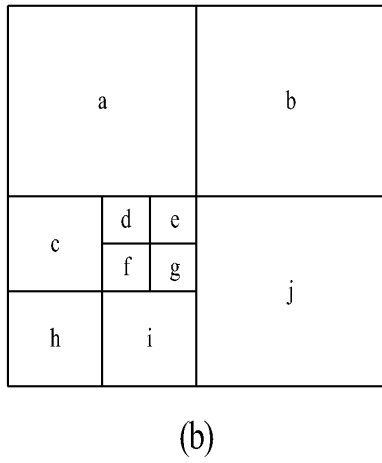
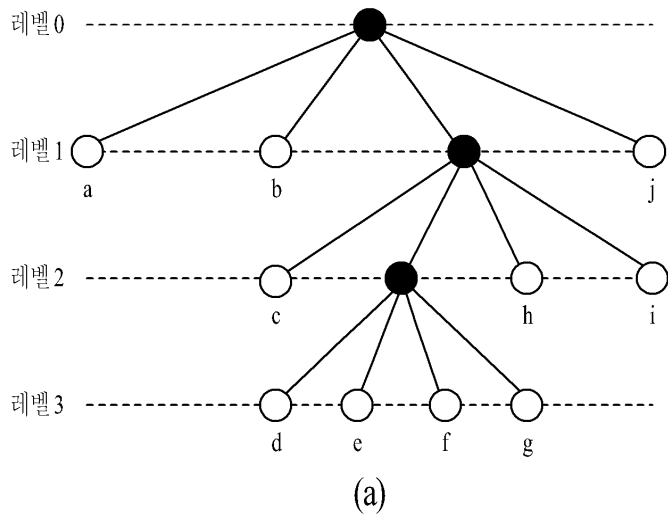
도면1



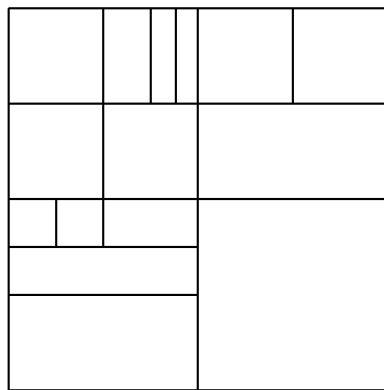
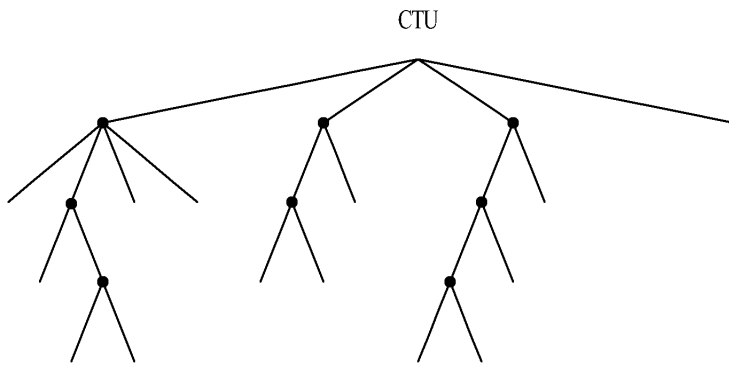
도면2



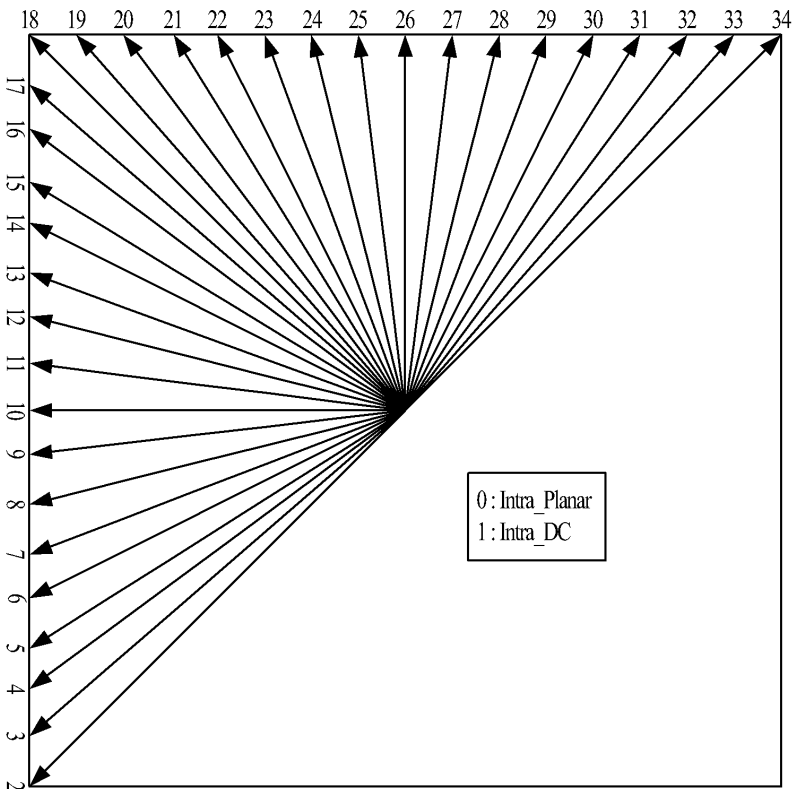
도면3



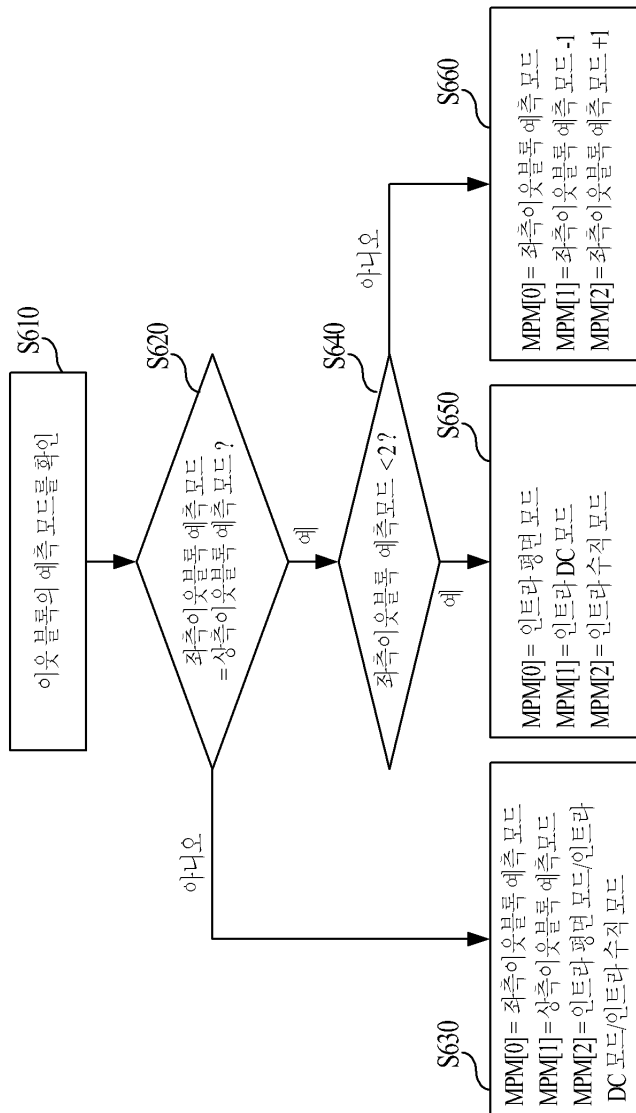
도면4



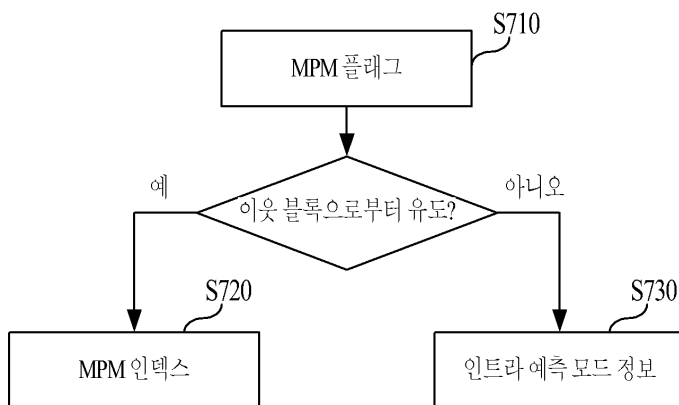
도면5



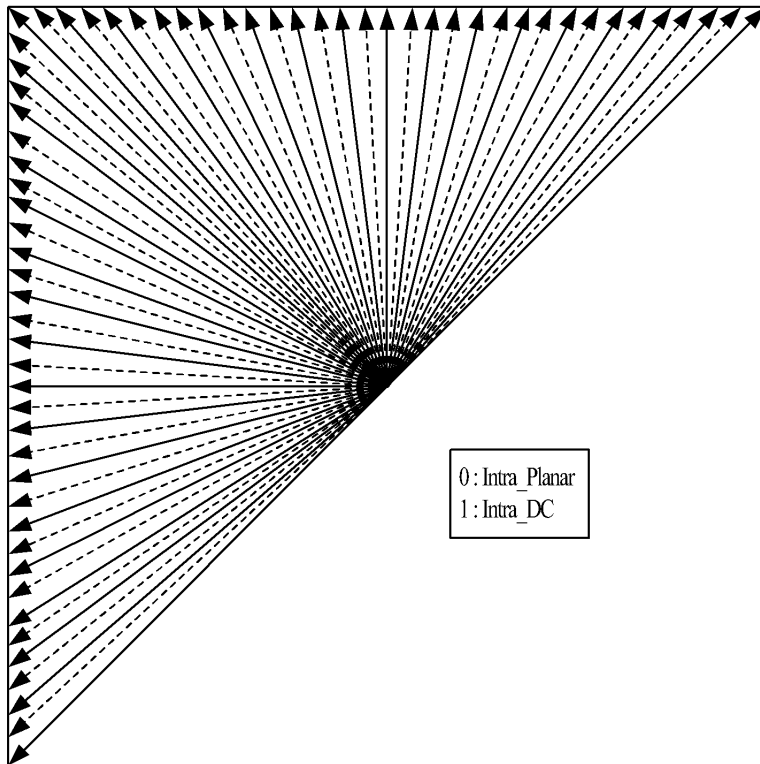
도면6



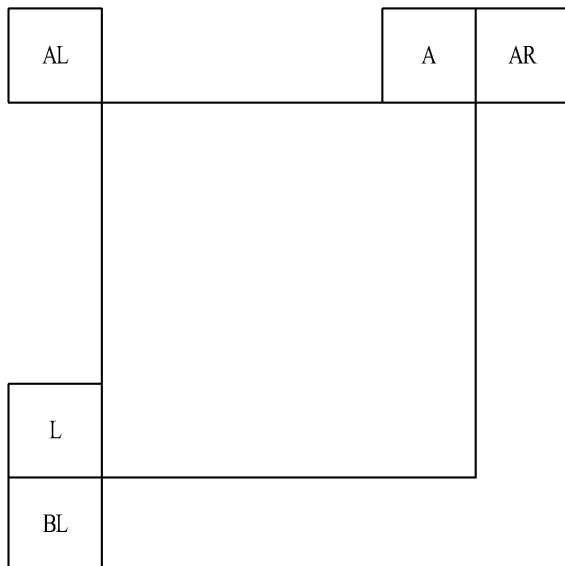
도면7



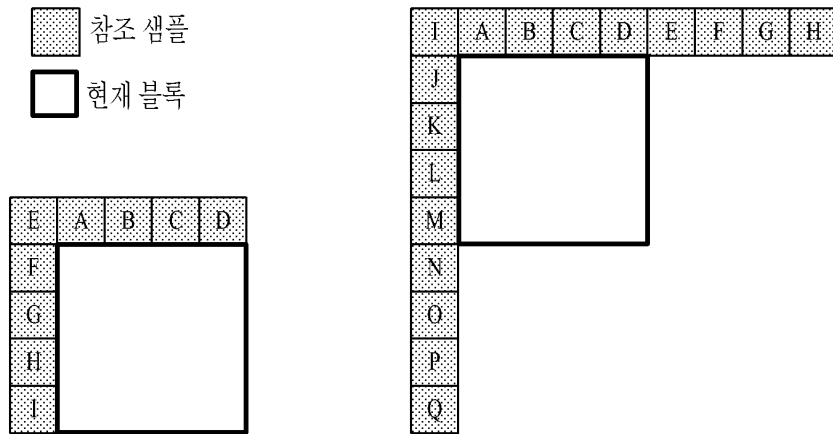
도면8



도면9



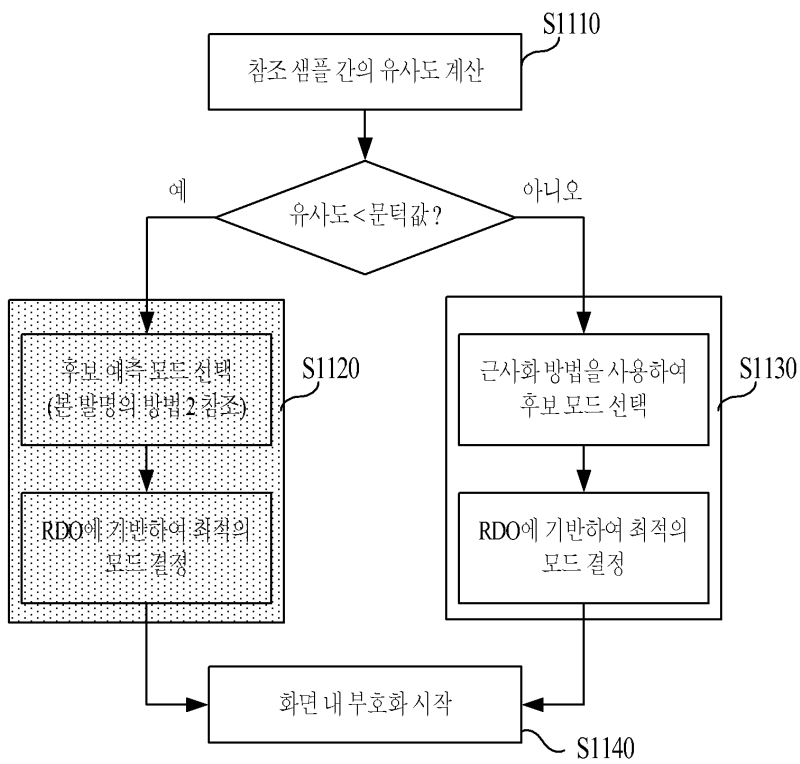
도면10



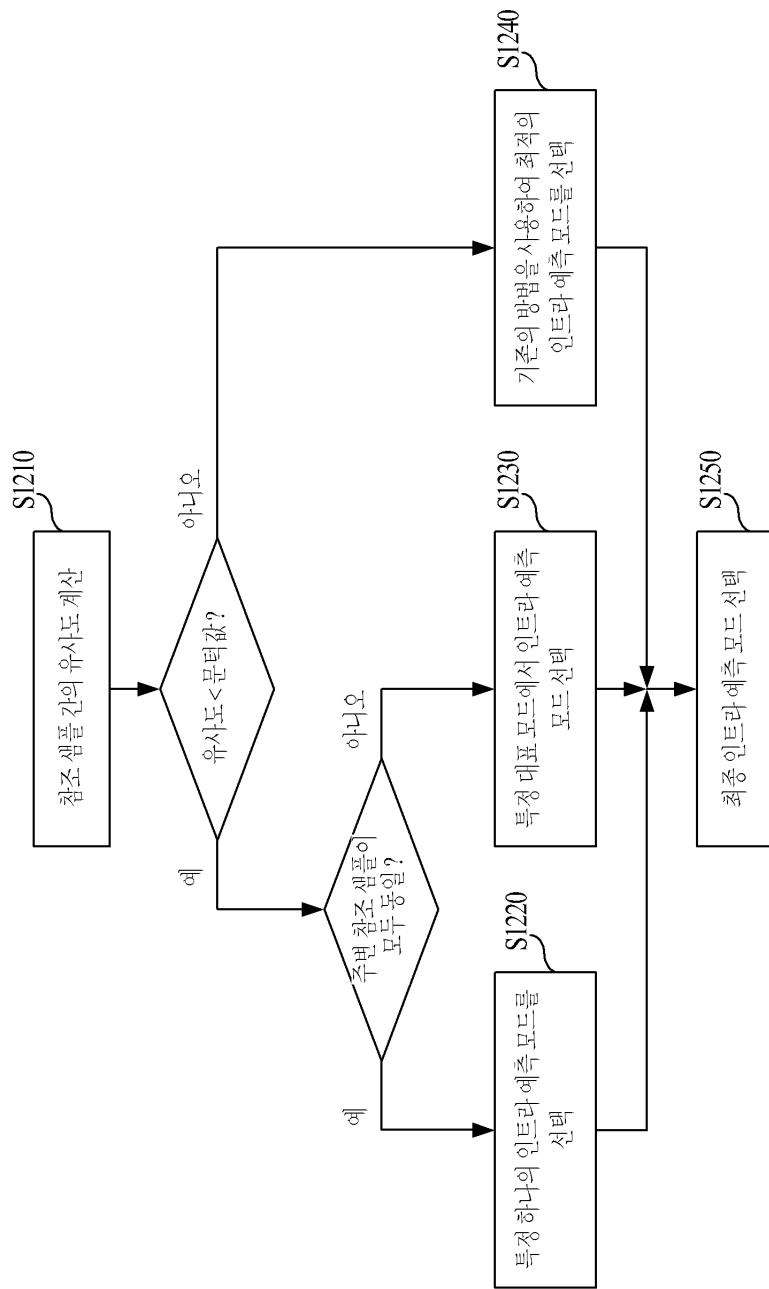
(a) 현재 블록과 동일한 크기의 참조 샘플

(b) 현재 블록보다 큰 크기의 참조 샘플

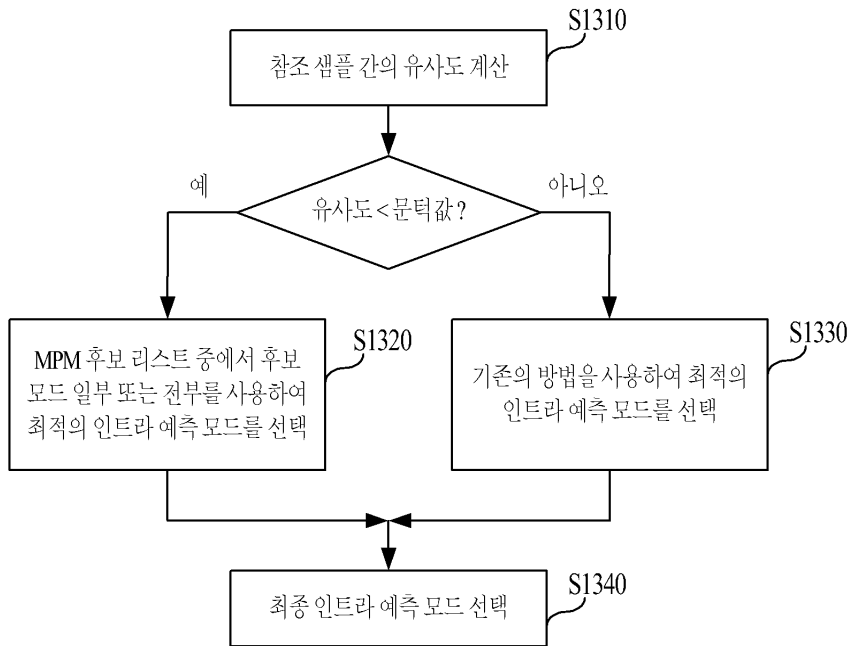
도면11



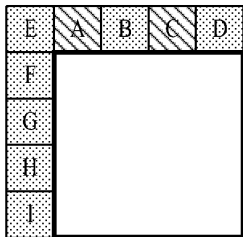
도면12



도면13



도면14



도면15

영상 처리 장치 (10)

