



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0100726
(43) 공개일자 2022년07월15일

<p>(51) 국제특허분류(Int. Cl.) H04N 19/61 (2014.01) H04N 19/11 (2014.01) H04N 19/122 (2014.01) H04N 19/13 (2014.01) H04N 19/593 (2014.01) H04N 19/70 (2014.01)</p> <p>(52) CPC특허분류 H04N 19/61 (2015.01) H04N 19/11 (2015.01)</p> <p>(21) 출원번호 10-2022-7022398</p> <p>(22) 출원일자(국제) 2021년06월29일 심사청구일자 2022년06월30일</p> <p>(85) 번역문제출일자 2022년06월30일</p> <p>(86) 국제출원번호 PCT/US2021/039633</p> <p>(87) 국제공개번호 WO 2022/103444 국제공개일자 2022년05월19일</p> <p>(30) 우선권주장 63/112,529 2020년11월11일 미국(US) 17/360,431 2021년06월28일 미국(US)</p>	<p>(71) 출원인 텐센트 아메리카 엘엘씨 미국 94306 캘리포니아주 팔로 알토 파크 블러바드 2747</p> <p>(72) 발명자 페링가세리 크리슈난 마두 미국 94306 캘리포니아주 팔로 알토 2747 파크 블러바드 텐센트 아메리카 엘엘씨 내</p> <p>자오 신 미국 94306 캘리포니아주 팔로 알토 2747 파크 블러바드 텐센트 아메리카 엘엘씨 내</p> <p>리우 산 미국 94306 캘리포니아주 팔로 알토 2747 파크 블러바드 텐센트 아메리카 엘엘씨 내</p> <p>(74) 대리인 유미특허법인</p>
---	--

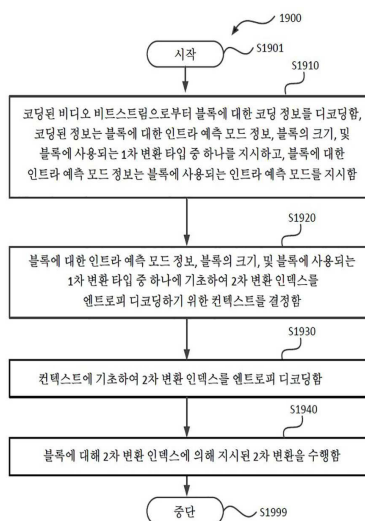
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 비디오 코딩을 위한 방법 및 장치

(57) 요약

본 개시내용의 양상은 비디오 디코딩을 위한 처리 회로를 포함하는 방법 및 장치를 제공한다. 처리 회로는 코딩된 비디오 비트스트림으로부터 변환 블록(TB)에 대한 코딩된 정보를 디코딩한다. 코딩된 정보는 TB에 사용되는 인트라 예측 모드를 지시하는 인트라 예측 모드 정보, TB의 크기, 및 TB에 사용되는 1차 변환 타입 중 하나를 지시한다. 처리 회로는 TB에 대한 인트라 예측 모드 정보, TB의 크기, 및 TB에 사용된 1차 변환 타입 중 하나에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정한다. 2차 변환 인덱스는 한 세트의 2차 변환 중에서 TB에 대해 수행될 2차 변환을 지시한다. 처리 회로는 컨텍스트에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하고 2차 변환을 수행한다.

대표도 - 도19



(52) CPC특허분류

H04N 19/122 (2015.01)

H04N 19/13 (2015.01)

H04N 19/593 (2015.01)

H04N 19/70 (2015.01)

명세서

청구범위

청구항 1

디코더(decoder)에서의 비디오(video) 디코딩을 위한 방법으로서,

코딩된 비디오 비트스트림(coded video bitstream)으로부터 변환 블록(TB: transform block)에 대한 코딩된 정보를 디코딩하는 단계 - 상기 코딩된 정보는 상기 TB에 대한 인트라(intra) 예측 모드 정보, 상기 TB의 크기, 및 상기 TB에 사용되는 1차 변환 타입 중 하나를 지시하고, 상기 TB에 대한 인트라 예측 모드 정보는 상기 TB에 사용되는 인트라 예측 모드를 지시함 -;

상기 TB에 대한 인트라 예측 모드 정보, 상기 TB의 크기, 및 상기 TB에 사용되는 1차 변환 타입 중 하나에 기초하여 2차 변환 인덱스를 엔트로피(entropy) 디코딩하기 위한 컨텍스트(context)를 결정하는 단계 - 상기 2차 변환 인덱스는 한 세트의 2차 변환 중에서 상기 TB에 대해 수행될 2차 변환을 지시함 -;

상기 컨텍스트에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하는 단계; 및

상기 TB에 대해 상기 2차 변환 인덱스에 의해 지시되는 2차 변환을 수행하는 단계를 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 2

제1항에 있어서,

상기 TB에 대한 인트라 예측 모드 정보, 상기 TB의 크기, 및 상기 TB에 사용되는 1차 변환 타입 중 하나는 상기 TB의 크기를 지시하고; 그리고

상기 컨텍스트를 결정하는 단계는 상기 TB의 크기에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 3

제2항에 있어서,

상기 TB의 크기는 상기 TB의 폭(W)과 상기 TB의 높이(H)를 지시하고,

상기 TB의 폭(W)과 상기 TB의 높이(H)의 최소값은 L이고, 그리고

상기 컨텍스트를 결정하는 단계는 L 또는 L×L에 기초하여 상기 컨텍스트를 결정하는 단계를 더 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 4

제1항에 있어서,

상기 TB에 대한 인트라 예측 모드 정보, 상기 TB의 크기, 및 상기 TB에 사용되는 1차 변환 타입 중 하나는 상기 TB에 대한 인트라 예측 모드 정보를 지시하고; 그리고

상기 컨텍스트를 결정하는 단계는 상기 TB에 대한 인트라 예측 모드 정보에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 5

제4항에 있어서,

상기 TB에 대한 인트라 예측 모드 정보는 공칭 모드 인덱스를 지시하고, 상기 TB는 상기 공칭 모드 인덱스 및 각도 오프셋에 기초하여 결정되는 방향성 예측 모드를 사용하여 예측되며, 그리고

상기 컨텍스트를 결정하는 단계는 상기 공칭 모드 인덱스에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 6

제4항에 있어서,

상기 TB에 대한 인트라 예측 모드 정보는 공칭 모드 인덱스를 지시하고, 상기 TB는 상기 공칭 모드 인덱스 및 각도 오프셋에 기초하여 결정되는 방향성 예측 모드를 사용하여 예측되며, 그리고

상기 컨텍스트를 결정하는 단계는 상기 공칭 모드 인덱스와 연관된 인덱스 값에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 7

제4항에 있어서,

상기 TB에 대한 인트라 예측 모드 정보는 무지향성 예측 모드 인덱스를 지시하고, 상기 TB는 상기 무지향성 예측 모드 인덱스에 의해 지시된 무지향성 예측 모드를 사용하여 예측되며, 그리고

상기 컨텍스트를 결정하는 단계는 상기 무지향성 예측 모드 인덱스에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 8

제4항에 있어서,

상기 TB에 대한 인트라 예측 모드 정보는 상기 TB를 예측하는데 사용되는 재귀적 필터링 모드를 지시하고,

상기 방법은 상기 재귀적 필터링 모드에 기초하여 공칭 모드 인덱스를 결정하는 단계를 포함하며, 상기 공칭 모드 인덱스는 공칭 모드를 지시하고, 그리고

상기 컨텍스트를 결정하는 단계는 상기 공칭 모드 인덱스에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 9

제1항에 있어서,

상기 TB에 대한 인트라 예측 모드 정보, 상기 TB의 크기, 및 상기 TB에 사용되는 1차 변환 타입 중 하나는 상기 TB에 사용되는 1차 변환 타입을 지시하고; 그리고

상기 컨텍스트를 결정하는 단계는 상기 TB에 사용되는 1차 변환 타입에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 10

제9항에 있어서,

상기 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함하고, 그리고

상기 컨텍스트를 결정하는 단계는 상기 수평 1차 변환 타입 및 상기 수직 1차 변환 타입이 둘 다 이산 코사인 변환(DCT: discrete cosine transform)인지 또는 둘 다 비대칭 이산 사인 변환(ADST: asymmetric discrete sine transform)인지에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 더 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 11

제9항에 있어서,

상기 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함하고, 그리고

상기 컨텍스트를 결정하는 단계는 상기 수평 1차 변환 타입 및 상기 수직 1차 변환 타입이 둘 다 이산 코사인 변환(DCT)인지 또는 둘 다 선 그래프 변환(LGT: line graph transform)인지에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 더 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 12

제9항에 있어서,

상기 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함하고, 그리고

상기 컨텍스트를 결정하는 단계는 상기 수평 1차 변환 타입 및 상기 수직 1차 변환 타입이 (i) 둘 다 이산 코사인 변환(DCT)인지, (ii) 둘 다 선 그래프 변환(LGT)인지, (iii) 각각 DCT 및 LGT인지, 또는 (iv) 각각 LGT 및 DCT인지에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 더 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 13

제9항에 있어서,

상기 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함하고, 그리고

상기 컨텍스트를 결정하는 단계는 상기 수평 1차 변환 타입 및 상기 수직 1차 변환 타입이 (i) 둘 다 이산 코사인 변환(DCT)인지, (ii) 둘 다 선 그래프 변환(LGT)인지, (iii) 각각 DCT 및 항등 변환(IDTX: identity transform)인지, 또는 (iv) 각각 LGT 및 DCT인지에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하는 단계를 더 포함하는,

디코더에서의 비디오 디코딩을 위한 방법.

청구항 14

비디오 디코딩을 위한 장치로서,

처리 회로를 포함하며,

상기 처리 회로는:

코딩된 비디오 비트스트림으로부터 변환 블록(TB)에 대한 코딩된 정보를 디코딩하고 - 상기 코딩된 정보는 상기 TB에 대한 인트라 예측 모드 정보, 상기 TB의 크기, 및 상기 TB에 사용되는 1차 변환 타입 중 하나를 지시하고, 상기 TB에 대한 인트라 예측 모드 정보는 상기 TB에 사용되는 인트라 예측 모드를 지시함 -;

상기 TB에 대한 인트라 예측 모드 정보, 상기 TB의 크기, 및 상기 TB에 사용되는 1차 변환 타입 중 하나에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하고 - 상기 2차 변환 인덱스는 한 세트의

2차 변환 중에서 상기 TB에 대해 수행될 2차 변환을 지시함 -;

상기 컨텍스트에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하고; 그리고

상기 TB에 대해 상기 2차 변환 인덱스에 의해 지시되는 2차 변환을 수행하도록 구성되는,

비디오 디코딩을 위한 장치.

청구항 15

제14항에 있어서,

상기 TB에 대한 인트라 예측 모드 정보, 상기 TB의 크기, 및 상기 TB에 사용되는 1차 변환 타입 중 하나는 상기 TB의 크기를 지시하고; 그리고

상기 처리 회로는 상기 TB의 크기에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하도록 구성되는,

비디오 디코딩을 위한 장치.

청구항 16

제14항에 있어서,

상기 TB에 대한 인트라 예측 모드 정보, 상기 TB의 크기, 및 상기 TB에 사용되는 1차 변환 타입 중 하나는 상기 TB에 대한 인트라 예측 모드 정보를 지시하고; 그리고

상기 처리 회로는 상기 TB에 대한 인트라 예측 모드 정보에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하도록 구성되는,

비디오 디코딩을 위한 장치.

청구항 17

제14항에 있어서,

상기 TB에 대한 인트라 예측 모드 정보, 상기 TB의 크기, 및 상기 TB에 사용되는 1차 변환 타입 중 하나는 상기 TB에 사용되는 1차 변환 타입을 지시하고; 그리고

상기 처리 회로는 상기 TB에 사용되는 1차 변환 타입에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하도록 구성되는,

비디오 디코딩을 위한 장치.

청구항 18

제17항에 있어서,

상기 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함하고; 그리고

상기 처리 회로는 상기 수평 1차 변환 타입 및 상기 수직 1차 변환 타입이 둘 다 이산 코사인 변환(DCT)인지 또는 둘 다 비대칭 이산 사인 변환(ADST)인지에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하도록 구성되는,

비디오 디코딩을 위한 장치.

청구항 19

제17항에 있어서,

상기 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함하고; 그리고

상기 처리 회로는 상기 수평 1차 변환 타입 및 상기 수직 1차 변환 타입이 둘 다 이산 코사인 변환(DCT)인지 또

는 둘 다 선 그래프 변환(LGT)인지에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하도록 구성되는,

비디오 디코딩을 위한 장치.

청구항 20

제17항에 있어서,

상기 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함하고; 그리고

상기 처리 회로는 상기 수평 1차 변환 타입 및 상기 수직 1차 변환 타입이 (i) 둘 다 이산 코사인 변환(DCT)인지, (ii) 둘 다 선 그래프 변환(LGT)인지, (iii) 각각 DCT 및 LGT인지, 또는 (iv) 각각 LGT 및 DCT인지에 기초하여 상기 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정하도록 구성되는,

비디오 디코딩을 위한 장치.

발명의 설명

기술 분야

[0001] 본 출원은 "Method and apparatus for video coding"이라는 명칭으로 2021년 6월 28일자 출원된 미국 특허출원 제17/360,431호의 우선권의 이익을 주장하는데, 이 특허출원은 "CONTEXT DESIGN FOR ENTROPY CODING OF SECONDARY TRANSFORM INDEX"라는 명칭으로 2020년 11월 11일자 출원된 미국 가출원 제63/112,529호에 대한 우선권의 이익을 주장한다. 선행 출원의 전체 개시내용은 이로써 그 전체가 인용에 의해 포함된다.

[0002] 본 개시내용은 일반적으로 비디오 코딩(video coding)과 관련된 실시예를 설명한다.

배경 기술

[0003] 본 명세서에서 제공되는 배경 설명은 일반적으로 본 개시내용의 맥락을 제시하는 것을 목적으로 한다. 현재 지명된 발명자의 작업은, 이 작업이 이 배경 섹션뿐만 아니라, 출원 시에 종래 기술로서 달리 한정할 수 없는 설명의 양상에서 설명되는 한도까지, 명확하게도 묵시적으로도 본 개시내용에 대한 종래 기술로서 인정되지 않는다.

[0004] 비디오 코딩 및 디코딩은 움직임 보상에 의한 인터 픽처(inter-picture) 예측을 사용하여 수행될 수 있다. 압축되지 않은 디지털 비디오는 일련의 픽처를 포함할 수 있으며, 각각의 픽처는 예를 들어, 1920 × 1080 휘도 샘플 및 연관된 색차 샘플의 공간 차원을 갖는다. 일련의 픽처는 예를 들어, 초당 60개의 픽처 또는 60Hz의 (비공식적으로는 프레임 레이트로도 또한 지칭되는) 고정 또는 가변 픽처 레이트를 가질 수 있다. 압축되지 않은 비디오는 특정 비트레이트 요건을 갖는다. 예를 들어, 샘플당 8비트의 1080p60 4:2:0 비디오(60Hz 프레임 레이트의 1920×1080 휘도 샘플 해상도)는 1.5Gbit/s에 가까운 대역폭을 필요로 한다. 이러한 비디오 1시간은 600기가바이트(Gbytes)를 넘는 저장 공간을 필요로 한다.

[0005] 비디오 코딩 및 디코딩의 한 가지 목적은 압축을 통한 입력 비디오 신호의 중복성 감소일 수 있다. 압축은 앞서 언급한 대역폭 및/또는 저장 공간 요건을 줄이는 데, 일부 경우에는 100배 이상 줄이는 데 도움이 될 수 있다. 무손실 압축 및 손실 압축뿐만 아니라 이들의 조합이 모두 이용될 수 있다. 무손실 압축은 압축된 원본 신호로부터 원본 신호의 정확한 사본이 재구성될 수 있는 기법을 의미한다. 손실 압축을 사용할 때, 재구성된 신호는 원본 신호와 동일하지 않을 수 있지만, 원본 신호와 재구성된 신호 사이의 왜곡은 재구성된 신호를 의도된 애플리케이션에 유용하게 하기에 충분히 작다. 비디오의 경우, 손실 압축이 널리 이용된다. 허용되는 왜곡의 양은 애플리케이션에 좌우되는데; 예를 들어, 특정 소비자 스트리밍 애플리케이션의 사용자는 텔레비전 배포 애플리케이션의 사용자보다 더 높은 왜곡을 허용할 수 있다. 달성 가능한 압축비는: 허용 가능한/견딜 수 있는 왜곡이 높을수록 압축비가 높아질 수 있다는 것을 반영할 수 있다.

[0006] 비디오 인코더(encoder) 및 디코더(decoder)는 예를 들어 움직임 보상, 변환, 양자화 및 엔트로피(entropy) 코딩을 포함하는 여러 광범위한 카테고리로부터의 기법을 이용할 수 있다.

[0007] 비디오 코덱(codec) 기술은 인트라 코딩(intra coding)으로 알려진 기법을 포함할 수 있다. 인트라 코딩에서, 샘플 값은 이전에 재구성된 참조 픽처로부터의 샘플 또는 다른 데이터에 대한 참조 없이 표현된다. 일부 비디오

오 코덱에서, 픽처는 샘플의 블록으로 공간적으로 세분된다. 샘플의 모든 블록이 인트라 모드에서 코딩될 때, 그 픽처는 인트라 픽처가 될 수 있다. 독립적 디코더 리프레시 픽처와 같은 인트라 픽처 및 이들의 파생물은 디코더 상태를 재설정하는 데 사용될 수 있고, 따라서 코딩된 비디오 비트스트림(coded video bitstream) 및 비디오 세션(session)의 첫 번째 픽처로서 또는 정지 이미지로서 사용될 수 있다. 인트라 블록의 샘플은 변환에 노출될 수 있고, 변환 계수는 엔트로피 코딩 전에 양자화될 수 있다. 인트라 예측은 변환 전 도메인(pre-transform domain)에서 샘플 값을 최소화하는 기법일 수 있다. 일부 경우에는, 변환 후 DC 값이 더 작을수록, 그리고 AC 계수가 더 작을수록, 엔트로피 코딩 후 블록을 나타내기 위해 주어진 양자화 단계 크기에 필요한 비트가 더 적다.

[0008] 예를 들어, MPEG-2 생성 코딩 기술로부터 알려진 것과 같은 종래의 인트라 코딩은 인트라 예측을 사용하지 않는다. 그러나 일부 더 새로운 비디오 압축 기술은 예를 들어, 주변의 샘플 데이터 및/또는 공간적으로 이웃하고 디코딩 순서가 선행하는 데이터 블록의 인코딩 및/또는 디코딩하는 동안 획득된 메타데이터로부터 시도하는 기법을 포함한다. 이러한 기법은 이후 "인트라 예측" 기법이라 한다. 적어도 일부 경우에, 인트라 예측은 참조 픽처가 아닌 재구성 중인 현재 픽처로부터의 참조 데이터만을 사용한다는 점에 주목한다.

[0009] 많은 다양한 형태의 인트라 예측이 있을 수 있다. 이러한 기법 중 하나보다 많은 기법이 주어진 비디오 코딩 기술에서 사용될 수 있는 경우, 사용 중인 기법은 인트라 예측 모드에서 코딩될 수 있다. 특정 경우에, 모드는 하위 모드 및/또는 파라미터를 가질 수 있으며, 이들은 개별적으로 코딩되거나 모드 코드워드에 포함될 수 있다. 주어진 모드, 하위 모드 및/또는 파라미터 조합에 어떤 코드워드를 사용할지는 인트라 예측을 통한 코딩 효율 이득에 영향을 미칠 수 있으며, 코드워드를 비트스트림으로 변환하는 데 사용되는 엔트로피 코딩 기술도 그러할 수 있다.

[0010] 특정 모드의 인트라 예측은 H.264로 도입되었으며, H.265로 개정되었고, 공동 탐사 모델(JEM: joint exploration model), 다용도 비디오 코딩(VVC: versatile video coding) 및 벤치마크 세트(BMS: benchmark set)와 같은 더 최신 코딩 기술에서 더 개정되었다. 이미 이용 가능한 샘플에 속하는 이웃 샘플 값을 사용하여 예측자 블록이 형성될 수 있다. 이웃 샘플의 샘플 값은 방향에 따라 예측자 블록에 복사된다. 사용 중인 방향에 대한 참조는 비트스트림으로 코딩될 수 있거나 자체적으로 예측될 수 있다.

[0011] 도 1a를 참조하면, 오른쪽 최하부에는 (35개의 인트라 모드의 33개의 각도 모드에 대응하는) H.265의 33개의 가능한 예측자 방향으로부터 알려진 9개의 예측자 방향의 서브세트가 도시된다. 화살표가 수렴하는 지점(101)은 예측되고 있는 샘플을 나타낸다. 화살표는 샘플이 예측되고 있는 방향을 나타낸다. 예를 들어, 화살표(102)는 샘플(101)이 수평에서 45도 각도로 오른쪽 상부에 있는 샘플 또는 샘플들로부터 예측된다는 것을 지시한다. 유사하게, 화살표(103)는 샘플(101)이 수평에서 22.5도 각도로 샘플(101)의 왼쪽 아래에 있는 샘플 또는 샘플들로부터 예측된다는 것을 지시한다.

[0012] 도 1a를 참조하면, 왼쪽 최상부에 (볼드체 파선으로 지시된) 4×4 샘플의 정사각형 블록(104)이 도시되어 있다. 정사각형 블록(104)은 각각 "S", Y 차원에서의 포지션(예컨대, 행 인덱스) 및 X 차원에서의 포지션(예컨대, 열 인덱스)으로 표기된 16개의 샘플을 포함한다. 예를 들어, 샘플 S21은 Y 차원에서 (위에서) 두 번째 샘플이고 X 차원에서 (왼쪽에서) 첫 번째 샘플이다. 유사하게, 샘플 S44는 Y 차원 및 X 차원 모두에서 블록(104) 내의 네 번째 샘플이다. 블록은 크기가 4×4 샘플이므로, S44는 오른쪽 하단에 있다. 유사한 번호 매기기 체계를 따르는 참조 샘플이 추가로 도시된다. 참조 샘플은 블록(104)에 대해 R, Y 포지션(예컨대, 행 인덱스) 및 X 포지션(열 인덱스)으로 표기된다. H.264 및 H.265 모두에서, 예측 샘플은 재구성 중인 블록에 인접하고; 따라서 음수 값이 사용될 필요가 없다.

[0013] 인트라 픽처 예측은 시그널링된 예측 방향에 의해 적절한 이웃 샘플로부터 참조 샘플 값을 복사함으로써 작동할 수 있다. 예를 들어, 코딩된 비디오 비트스트림이 이 블록에 대해 화살표(102)와 일치하는 예측 방향을 지시하는 시그널링을 포함한다고 가정하는데, 즉 수평에서 45도 각도로 샘플 또는 샘플들로부터 오른쪽 상부까지 샘플이 예측된다. 그 경우, 동일한 참조 샘플 R05로부터 샘플 S41, S32, S23 및 S14가 예측된다. 그런 다음, 참조 샘플 R08로부터 샘플 S44가 예측된다.

[0014] 특정 경우에, 참조 샘플을 계산하기 위해; 특히 방향이 45도로 균등하게 나누어지지 않을 때, 예를 들어 보간을 통해 다수의 참조 샘플의 값이 조합될 수 있다.

[0015] 비디오 코딩 기술이 발달함에 따라 가능한 방향의 수가 증가했다. H.264(2003년)에서는, 9개의 상이한 방향이 표현될 수 있다. H.265(2013년)에서는 그것이 33으로 증가했고, 공개 당시 JEM/VVC/BMS는 최대 65개의 방향을

지원할 수 있다. 가장 가능성이 높은 방향을 식별하기 위한 실험이 수행되었으며, 가능성이 적은 방향에 대해 특정 페널티를 수용하여, 소수의 비트에서 그러한 가능성 있는 방향을 나타내는 데 엔트로피 코딩의 특정 기법이 사용된다. 또한, 방향 자체는 간혹, 이미 디코딩된 이웃 블록에서 사용되는 이웃 방향으로부터 예측될 수 있다.

[0016] 도 1b는 시간 경과에 따라 증가하는 수의 예측 방향을 제시하기 위해 JEM에 따른 65개의 인트라 예측 방향을 도시하는 개략도(180)를 도시한다.

[0017] 방향을 나타내는 코딩된 비디오 비트스트림의 인트라 예측 방향 비트의 매핑은 비디오 코딩 기술마다 다를 수 있으며; 예를 들어, 예측 방향의 단순한 직접 매핑에서부터 인트라 예측 모드, 코드워드, 가장 가능성 있는 모드를 수반하는 복잡한 적응 방식 및 유사한 기술에 이르기까지 다양할 수 있다. 그러나 모든 경우에, 통계적으로 다른 특정 방향보다 비디오 콘텐츠에서 발생할 가능성이 적은 특정 방향이 있을 수 있다. 비디오 압축의 목표는 중복성의 감소이므로, 잘 작동하는 비디오 코딩 기술에서 가능성이 더 적은 그러한 방향은 가능성이 더 높은 방향보다 더 많은 수의 비트로 표현될 것이다.

[0018] 움직임 보상은 손실 압축 기법일 수 있으며, 이전에 재구성된 픽처 또는 그 일부(참조 픽처)로부터의 샘플 데이터 블록이 움직임 벡터(이후 MV)로 지시되는 방향으로 공간적으로 시프트된 후, 새로 재구성된 픽처 또는 픽처 부분의 예측에 사용되는 기법과 관련될 수 있다. 일부 경우에는, 참조 픽처가 현재 재구성 중인 픽처와 동일할 수 있다. MV는 X 차원과 Y 차원인 2차원 또는 3차원을 가질 수 있으며, 세 번째 차원은 사용 중인 참조 픽처의 지시이다(세 번째 차원은 간접적으로는 시간 차원이 될 수 있음).

[0019] 일부 비디오 압축 기법에서, 샘플 데이터의 특정 영역에 적용 가능한 MV는 다른 MV로부터, 예를 들어 재구성 중인 영역에 공간적으로 인접하고 디코딩 순서에서 해당 MV에 선행하는 샘플 데이터의 다른 영역과 관련된 것들로부터 예측될 수 있다. 그렇게 하면 MV를 코딩하는 데 필요한 데이터의 양을 상당히 줄일 수 있으며, 이로써 중복성을 제거하고 압축성을 높일 수 있다. 예를 들어, MV 예측은 효과적으로 작동할 수 있는데, 이는 카메라로부터 도출된 (본래의 비디오로 알려진) 입력 비디오 신호를 코딩할 때 단일 MV가 적용 가능한 영역보다 더 큰 영역이 유사한 방향으로 이동하고 따라서 일부 경우에는, 인접 영역의 MV에서 도출된 유사한 움직임 벡터를 사용하여 예측될 수 있는 통계적 가능성이 있기 때문이다. 그 결과, 주어진 영역에 대해 발견된 MV가 주변 MV로부터 예측된 MV와 유사하거나 동일하게 되며, 이는 결국, 엔트로피 코딩 이후, MV를 직접 코딩하는 경우에 사용될 것보다 더 적은 수의 비트로 표현될 수 있다. 일부 경우에, MV 예측은 원본 신호(즉: 샘플 스트림)에서 도출된 신호(즉: MV)의 무손실 압축의 일례일 수 있다. 다른 경우에, 예를 들어, 여러 주변 MV로부터 예측자를 계산할 때 반올림 오류로 인해 MV 예측 자체는 손실이 있을 수 있다.

[0020] H.265/HEVC(ITU-T Rec. H.265, "High Efficiency Video Coding", 2016년 12월)에서 다양한 MV 예측 메커니즘이 설명된다. H.265가 제공하는 많은 MV 예측 메커니즘 중에서, 여기서는 이후 "공간 병합"으로 지칭되는 기법이 설명된다.

[0021] 도 2를 참조하면, 현재 블록(201)은 공간적으로 시프트된 동일한 크기의 이전 블록으로부터 예측 가능하도록 움직임 탐색 프로세스 동안 인코더에 의해 발견된 샘플을 포함한다. 해당 MV를 직접 코딩하는 대신에, MV는 A0, A1 및 B0, B1, B2(각각 202 내지 206)로 표시된 5개의 주변 샘플 중 하나와 연관된 MV를 사용하여 하나 이상의 참조 픽처와 연관된 메타데이터로부터, 예를 들어 (디코딩 순서에서) 가장 최근의 참조 픽처로부터 도출될 수 있다. H.265에서, MV 예측은 이웃 블록이 사용하고 있는 동일한 참조 픽처의 예측자를 사용할 수 있다.

발명의 내용

[0022] 본 개시내용의 양상은 비디오 인코딩 및/또는 디코딩을 위한 방법 및 장치를 제공한다. 일부 예에서, 비디오 디코딩을 위한 장치는 처리 회로를 포함한다. 처리 회로는 코딩된 비디오 비트스트림으로부터 변환 블록(TB: transform block)에 대한 코딩된 정보를 디코딩할 수 있다. 코딩된 정보는 TB에 대한 인트라 예측 모드 정보, TB의 크기, 및 TB에 사용되는 1차 변환 타입 중 하나를 지시할 수 있다. TB에 대한 인트라 예측 모드 정보는 TB에 사용되는 인트라 예측 모드를 지시할 수 있다. 처리 회로는 TB에 대한 인트라 예측 모드 정보, TB의 크기, 및 TB에 사용된 1차 변환 타입 중 하나에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트(context)를 결정할 수 있다. 2차 변환 인덱스는 한 세트의 2차 변환 중에서 TB에 대해 수행될 2차 변환을 지시할 수 있다. 처리 회로는 컨텍스트에 기초하여 2차 변환 인덱스를 엔트로피 디코딩할 수 있고 TB에 대해 2차 변환 인덱스에 의해 지시된 2차 변환을 수행할 수 있다.

[0023] 일 실시예에서, TB에 대한 인트라 예측 모드 정보, TB의 크기, 및 TB에 사용되는 1차 변환 타입 중 하나는 TB의

크기를 지시할 수 있다. 처리 회로는 TB의 크기에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정할 수 있다. 일례로, TB의 크기는 TB의 폭(W) 및 TB의 높이(H)를 지시하고, TB의 폭(W) 및 TB의 높이(H)의 최소값은 L이며, 처리 회로는 L 또는 L×L을 기반으로 하는 컨텍스트를 결정할 수 있다.

[0024] 일 실시예에서, TB에 대한 인트라 예측 모드 정보, TB의 크기, 및 TB에 사용되는 1차 변환 타입 중 하나는 TB에 대한 인트라 예측 모드 정보를 지시할 수 있다. 처리 회로는 TB에 대한 인트라 예측 모드 정보에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정할 수 있다.

[0025] 일례로, TB에 대한 인트라 예측 모드 정보는 공칭 모드 인덱스를 지시하고, TB는 공칭 모드 인덱스와 각도 오프셋을 기반으로 결정되는 방향성 예측 모드를 사용하여 예측되며, 처리 회로는 공칭 모드 인덱스에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정할 수 있다.

[0026] 일례로, TB에 대한 인트라 예측 모드 정보는 공칭 모드 인덱스를 지시한다. TB는 공칭 모드 인덱스와 각도 오프셋을 기반으로 결정되는 방향성 예측 모드를 이용하여 예측될 수 있다. 처리 회로는 공칭 모드 인덱스와 연관된 인덱스 값에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정할 수 있다.

[0027] 일례로, TB에 대한 인트라 예측 모드 정보는 무지향성 예측 모드 인덱스를 지시한다. TB는 무지향성 예측 모드 인덱스에 의해 지시된 무지향성 예측 모드를 사용하여 예측될 수 있다. 처리 회로는 무지향성 예측 모드 인덱스에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정할 수 있다.

[0028] 일례로, TB에 대한 인트라 예측 모드 정보는 TB를 예측하는데 사용되는 재귀적 필터링 모드를 지시한다. 처리 회로는 재귀적 필터링 모드에 기초하여 공칭 모드 인덱스를 결정할 수 있다. 공칭 모드 인덱스는 공칭 모드를 지시할 수 있다. 처리 회로는 공칭 모드 인덱스에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정할 수 있다.

[0029] 일 실시예에서, TB에 대한 인트라 예측 모드 정보, TB의 크기, 및 TB에 사용되는 1차 변환 타입 중 하나는 TB에 사용되는 1차 변환 타입을 지시할 수 있다. 처리 회로는 TB에 사용되는 1차 변환 타입에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정할 수 있다.

[0030] 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함할 수 있다. 일례로, 처리 회로는 수평 1차 변환 타입 및 수직 1차 변환 타입이 둘 다 이산 코사인 변환(DCT: discrete cosine transform)인지 또는 둘 다 비대칭 이산 사인 변환(ADST: asymmetric discrete sine transform)인지에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정한다.

[0031] 일례로, 처리 회로는 수평 1차 변환 타입 및 수직 1차 변환 타입이 둘 다 이산 코사인 변환(DCT)인지 또는 둘 다 선 그래프 변환(LGT: line graph transform)인지에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정한다.

[0032] 일례로, 처리 회로는 수평 1차 변환 타입 및 수직 1차 변환 타입이 (i) 둘 다 이산 코사인 변환(DCT)인지, (ii) 둘 다 선 그래프 변환(LGT)인지, (iii) 각각 DCT 및 LGT인지, 또는 (iv) 각각 LGT 및 DCT인지에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정한다.

[0033] 일례로, 처리 회로는 수평 1차 변환 타입 및 수직 1차 변환 타입이 (i) 둘 다 이산 코사인 변환(DCT)인지, (ii) 둘 다 선 그래프 변환(LGT)인지, (iii) 각각 DCT 및 항등 변환(IDTX: identity transform)인지, 또는 (iv) 각각 IDTX 및 DCT인지에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트를 결정한다.

[0034] 본 개시내용의 양상들은 또한, 비디오 디코딩을 위해 컴퓨터에 의해 실행될 때, 컴퓨터로 하여금 비디오 디코딩 및/또는 인코딩을 위한 방법을 수행하게 하는 명령을 저장하는 비-일시적 컴퓨터 판독 가능 매체를 제공한다.

도면의 간단한 설명

[0035] 개시된 청구 대상의 추가 특징, 성질 및 다양한 이점은 다음의 상세한 설명 및 첨부 도면으로부터 보다 명백해질 것이다.

도 1a는 인트라 예측 모드의 예시적인 서브세트의 개략적인 예시이다.

도 1b는 예시적인 인트라 예측 방향의 예시이다.

- 도 2는 일 예에서 현재 블록 및 그 주변 공간 병합 후보의 개략적인 예시이다.
- 도 3은 일 실시예에 따른 통신 시스템(300)의 단순화된 블록도의 개략적인 예시이다.
- 도 4는 일 실시예에 따른 통신 시스템(400)의 단순화된 블록도의 개략적인 예시이다.
- 도 5는 일 실시예에 따른 디코더의 단순화된 블록도의 개략적인 예시이다.
- 도 6은 일 실시예에 따른 인코더의 단순화된 블록도의 개략적인 예시이다.
- 도 7은 다른 실시예에 따른 인코더의 블록도를 도시한다.
- 도 8은 다른 실시예에 따른 디코더의 블록도를 도시한다.
- 도 9는 본 개시내용의 일 실시예에 따른 코딩 블록에 대한 공칭 모드의 일례를 도시한다.
- 도 10은 본 개시내용의 양상에 따른 무지향성 평활 인트라 예측에 대한 예를 도시한다.
- 도 11은 본 개시내용의 일 실시예에 따른 재귀적 필터링 기반 인트라 예측자의 일례를 도시한다.
- 도 12는 본 개시내용의 일 실시예에 따른 코딩 블록에 대한 다수의 기준선의 일례를 도시한다.
- 도 13은 본 발명의 일 실시예에 따른 블록에 대한 변환 블록 파티션의 일례를 도시한다.
- 도 14는 본 개시내용의 일 실시예에 따른 블록에 대한 변환 블록 파티션의 일례를 도시한다.
- 도 15는 본 개시내용의 실시예에 따른 1차 변환 기저 함수의 예를 도시한다.
- 도 16a는 본 개시내용의 실시예에 따른 변환 블록 크기 및 예측 모드에 기반한 다양한 변환 커널의 이용 가능성의 예시적인 종속성을 도시한다.
- 도 16b는 본 개시내용의 실시예에 따른 인트라 예측 모드에 기반한 예시적인 변환 타입 선택을 도시한다.
- 도 16c는 본 개시내용의 일 실시예에 따른 자가 루프 가중치 및 에지 가중치를 특징으로 하는 일반 선 그래프 변환(LGT)의 일례를 도시한다.
- 도 16d는 본 개시내용의 일 실시예에 따른 예시적인 일반화된 그래프 라플라시안(GGL: generalized graph Laplacian) 행렬을 도시한다.
- 도 17 - 도 18은 본 개시내용의 실시예에 따른 각각 16x64 변환 및 16x48 변환을 사용하는 2개의 변환 코딩 프로세스(1700, 1800)의 예를 도시한다.
- 도 19는 본 개시내용의 일 실시예에 따른 프로세스(1900)의 개요를 서술하는 흐름도를 도시한다.
- 도 20은 일 실시예에 따른 컴퓨터 시스템의 개략적인 예시이다.

발명을 실시하기 위한 구체적인 내용

[0036] 도 3은 본 개시내용의 일 실시예에 따른 통신 시스템(300)의 단순화된 블록도를 예시한다. 통신 시스템(300)은 예를 들어, 네트워크(350)를 통해 서로 통신할 수 있는 복수의 단말 디바이스를 포함한다. 예를 들어, 통신 시스템(300)은 네트워크(350)를 통해 상호 연결된 제1 쌍의 단말 디바이스(310, 320)를 포함한다. 도 3의 예에서, 제1 쌍의 단말 디바이스(310, 320)는 데이터의 단방향 전송을 수행한다. 예를 들어, 단말 디바이스(310)는 네트워크(350)를 통해 다른 단말 디바이스(320)로 전송하기 위한 비디오 데이터(예컨대, 단말 디바이스(310)에 의해 캡처되는 비디오 픽처의 스트림)를 코딩할 수 있다. 인코딩된 비디오 데이터는 하나 이상의 코딩된 비디오 비트스트림의 형태로 전송될 수 있다. 단말 디바이스(320)는 네트워크(350)로부터 코딩된 비디오 데이터를 수신하고, 코딩된 비디오 데이터를 디코딩하여 비디오 픽처를 복원하고, 복원된 비디오 데이터에 따라 비디오 픽처를 디스플레이할 수 있다. 단방향 데이터 전송은 미디어 서버 애플리케이션 등에서 공통적일 수 있다.

[0037] 다른 예에서, 통신 시스템(300)은 예를 들어, 화상 회의 중에 발생할 수 있는 코딩된 비디오 데이터의 양방향 전송을 수행하는 제2 쌍의 단말 디바이스(330, 340)를 포함한다. 데이터의 양방향 전송을 위해, 일례로, 단말 디바이스(330, 340)의 각각의 단말 디바이스는 네트워크(350)를 통해 단말 디바이스(330, 340) 중 다른 단말 디바이스로 전송하기 위한 비디오 데이터(예컨대, 단말 디바이스에 의해 캡처되는 비디오 픽처의 스트림)를 코딩할 수 있다. 단말 디바이스(330, 340)의 각각의 단말 디바이스는 또한, 단말 디바이스(330, 340) 중 다른 단말

디바이스에 의해 전송된 코딩된 비디오 데이터를 수신할 수 있고, 코딩된 비디오 데이터를 디코딩하여 비디오 픽처를 복원할 수 있으며, 복원된 비디오 데이터에 따라 액세스 가능한 디스플레이 디바이스에 비디오 픽처를 디스플레이할 수 있다.

[0038] 도 3의 예에서, 단말 디바이스(310, 320, 330, 340)는 서버, 개인용 컴퓨터 및 스마트폰으로서 예시될 수 있지만, 본 개시내용의 원리는 이에 제한되지 않을 수 있다. 본 개시내용의 실시예는 랩톱 컴퓨터, 태블릿 컴퓨터, 미디어 플레이어 및/또는 전용 화상 회의 장비를 사용한 애플리케이션을 찾아낸다. 네트워크(350)는 예를 들어, 와이어라인(유선) 및/또는 무선 통신 네트워크를 포함하여, 코딩된 비디오 데이터를 단말 디바이스(310, 320, 330, 340) 사이에 전달하는 임의의 수의 네트워크를 나타낸다. 통신 네트워크(350)는 회선 교환 및/또는 패킷 교환 채널에서 데이터를 교환할 수 있다. 대표적인 네트워크는 원격 통신 네트워크, 근거리 네트워크, 광역 네트워크 및/또는 인터넷을 포함한다. 본 논의의 목적으로, 네트워크(350)의 아키텍처 및 토폴로지는 본 명세서에서 아래에 설명되지 않는 한 본 개시내용의 동작에 중요하지 않을 수 있다.

[0039] 도 4는 개시된 청구 대상에 대한 애플리케이션에 대한 일례로, 스트리밍 환경에서의 비디오 인코더 및 비디오 디코더의 배치를 예시한다. 개시된 청구 대상은 예를 들어, 화상 회의, 디지털 TV, CD, DVD, 메모리 스틱 등을 포함하는 디지털 매체 상의 압축된 비디오의 저장 등을 포함하는 다른 비디오 가능 애플리케이션에 동일하게 적용 가능할 수 있다.

[0040] 스트리밍 시스템은 예를 들어, 압축되지 않은 비디오 픽처의 스트림(402)을 생성하는 비디오 소스(401), 예를 들어 디지털 카메라를 포함할 수 있는 캡처 서브시스템(413)을 포함할 수 있다. 일례로, 비디오 픽처의 스트림(402)은 디지털 카메라에 의해 촬영되는 샘플을 포함한다. 인코딩된 비디오 데이터(404)(또는 코딩된 비디오 비트스트림)와 비교할 때 높은 데이터 볼륨을 강조하기 위해 굵은 선으로 표시된 비디오 픽처의 스트림(402)은 비디오 소스(401)에 결합된 비디오 인코더(403)를 포함하는 전자 디바이스(420)에 의해 처리될 수 있다. 비디오 인코더(403)는 아래에서 더 상세히 설명되는 바와 같이, 개시된 청구 대상의 양상을 가능하게 하거나 구현하기 위한 하드웨어, 소프트웨어, 또는 이들의 조합을 포함할 수 있다. 비디오 픽처의 스트림(402)과 비교할 때 더 낮은 데이터 볼륨을 강조하기 위해 가는 선으로 표시된 인코딩된 비디오 데이터(404)(또는 인코딩된 비디오 비트스트림(404))는 향후 사용을 위해 스트리밍 서버(405)에 저장될 수 있다. 도 4의 클라이언트 서브시스템(406, 408)과 같은 하나 이상의 스트리밍 클라이언트 서브시스템은 스트리밍 서버(405)에 액세스하여 인코딩된 비디오 데이터(404)의 사본(407, 409)을 리트리브(retrieve)할 수 있다. 클라이언트 서브시스템(406)은 예를 들어, 전자 디바이스(430)에 비디오 디코더(410)를 포함할 수 있다. 비디오 디코더(410)는 인코딩된 비디오 데이터의 착신(incoming) 사본(407)을 디코딩하고, 디스플레이(412)(예컨대, 디스플레이 스크린) 또는 (도시되지 않은) 다른 렌더링 디바이스에 렌더링될 수 있는 발신 비디오 픽처의 스트림(411)을 생성할 수 있다. 일부 스트리밍 시스템에서, 인코딩된 비디오 데이터(404, 407, 409)(예컨대, 비디오 비트스트림)은 특정 비디오 코딩/압축 표준에 따라 인코딩될 수 있다. 그러한 표준의 예는 ITU-T 권고 H.265를 포함한다. 일례로, 개발 중인 비디오 코딩 표준은 비공식적으로, 다용도 비디오 코딩(VVC)으로 알려져 있다. 개시되는 청구 대상은 VVC와 관련하여 사용될 수 있다.

[0041] 전자 디바이스(420, 430)는 (도시되지 않은) 다른 컴포넌트를 포함할 수 있다는 점이 주목된다. 예를 들어, 전자 디바이스(420)는 (도시되지 않은) 비디오 디코더를 포함할 수 있고, 전자 디바이스(430)는 (도시되지 않은) 비디오 인코더를 또한 포함할 수 있다.

[0042] 도 5는 본 개시내용의 일 실시예에 따른 비디오 디코더(510)의 블록도를 도시한다. 비디오 디코더(510)는 전자 디바이스(530)에 포함될 수 있다. 전자 디바이스(530)는 수신기(531)(예컨대, 수신 회로)를 포함할 수 있다. 비디오 디코더(510)는 도 4의 예에서의 비디오 디코더(410) 대신에 사용될 수 있다.

[0043] 수신기(531)는 비디오 디코더(510)에 의해 디코딩될 하나 이상의 코딩된 비디오 시퀀스를; 동일한 또는 다른 실시예에서는 한 번에 하나의 코딩된 비디오 시퀀스씩 수신할 수 있으며, 여기서 각각의 코딩된 비디오 시퀀스의 디코딩은 다른 코딩된 비디오 시퀀스와 독립적이다. 코딩된 비디오 시퀀스는 하드웨어/소프트웨어 링크일 수 있는 채널(501)로부터, 인코딩된 비디오 데이터를 저장하는 저장 디바이스에 수신될 수 있다. 수신기(531)는 (도시되지 않은) 엔티티를 사용하여 이들 각각에 포워딩될 수 있는 다른 데이터, 예를 들어 코딩된 오디오 데이터 및/또는 보조 데이터 스트림과 함께 인코딩된 비디오 데이터를 수신할 수 있다. 수신기(531)는 코딩된 비디오 시퀀스를 다른 데이터로부터 분리할 수 있다. 네트워크 지터(jitter)에 대처하기 위해, 버퍼 메모리(515)는 수신기(531)와 엔트로피 디코더/파서(parser)(520)(이하 "파서(520)") 사이에 결합될 수 있다. 특정 애플리케이션에서, 버퍼 메모리(515)는 비디오 디코더(510)의 일부이다. 다른 애플리케이션에서, 버퍼 메모리(515)는

(도시되지 않은) 비디오 디코더(510) 외부에 있을 수 있다. 또 다른 애플리케이션에서, 예를 들어 네트워크 지터를 방지하기 위해 비디오 디코더(510) 외부에 (도시되지 않은) 버퍼 메모리가 있을 수 있고, 추가로 예를 들어, 재생 타이밍을 처리하기 위해 비디오 디코더(510) 내부에 다른 버퍼 메모리(515)가 있을 수 있다. 수신기(531)가 충분한 대역폭 및 제어 가능성의 저장/전달 디바이스로부터, 또는 등시성(isosynchronous) 네트워크로부터 데이터를 수신하고 있을 때, 버퍼 메모리(515)는 필요하지 않을 수 있거나 작을 수 있다. 인터넷과 같은 최선형(best effort) 패킷 네트워크 상에서의 사용을 위해, 버퍼 메모리(515)가 요구될 수 있고, 비교적 클 수 있고, 유리하게는 적응적 크기일 수 있으며, 비디오 디코더(510) 외부의 운영 체제 또는 유사한 엘리먼트(도시되지 않음)에서 적어도 부분적으로 구현될 수 있다.

[0044] 비디오 디코더(510)는 코딩된 비디오 시퀀스로부터 심벌(521)을 재구성하기 위한 파서(520)를 포함할 수 있다. 그러한 심벌의 카테고리는 비디오 디코더(510)의 동작을 관리하는 데 사용되는 정보, 및 잠재적으로는, 도 5에 도시된 바와 같이, 전자 디바이스(530)의 내장 부품이 아니라 전자 디바이스(530)에 결합될 수 있는 렌더링 디바이스(512)(예컨대, 디스플레이 스크린)와 같은 렌더링 디바이스를 제어하기 위한 정보를 포함한다. 렌더링 디바이스(들)에 대한 제어 정보는 부가 확장 정보(SEI: Supplemental Enhancement Information) 메시지 또는 비디오 사용성 정보(VUI: Video Usability Information) 파라미터 세트 프래그먼트(도시되지 않음)의 형태일 수 있다. 파서(520)는 수신되는 코딩된 비디오 시퀀스를 파싱(parse)/엔트로피 디코딩할 수 있다. 코딩된 비디오 시퀀스의 코딩은 비디오 코딩 기술 또는 표준에 따를 수 있고, 가변 길이 코딩, 허프만(Huffman) 코딩, 상황 민감도를 갖거나 갖지 않는 산술 코딩 등을 포함하여, 다양한 원리를 따를 수 있다. 파서(520)는 그룹에 대응하는 적어도 하나의 파라미터에 기초하여, 비디오 디코더 내의 픽셀의 하위 그룹 중 적어도 하나에 대한 한 세트의 하위 그룹 파라미터를 코딩된 비디오 시퀀스로부터 추출할 수 있다. 하위 그룹은 픽처의 그룹(GOP: Group of Pictures), 픽처, 타일, 슬라이스, 매크로 블록, 코딩 유닛(CU: Coding Unit), 블록, 변환 유닛(TU: Transform Unit), 예측 유닛(PU: Prediction Unit) 등을 포함할 수 있다. 파서(520)는 또한, 변환 계수, 양자화기 파라미터 값, 움직임 벡터 등과 같은 정보를 코딩된 비디오 시퀀스로부터 추출할 수 있다.

[0045] 파서(520)는 심벌(521)을 생성하기 위해, 버퍼 메모리(515)로부터 수신된 비디오 시퀀스에 대해 엔트로피 디코딩/파싱 동작을 수행할 수 있다.

[0046] 심벌(521)의 재구성은 코딩된 비디오 픽처 또는 그 부분의 타입(이를테면: 인터(inter) 및 인트라 픽처, 인터 및 인트라 블록) 및 다른 팩터에 따라 다수의 상이한 유닛을 수반할 수 있다. 어떤 유닛이 어떻게 수반되는지는 코딩된 비디오 시퀀스로부터 파서(520)에 의해 파싱된 하위 그룹 제어 정보에 의해 제어될 수 있다. 파서(520)와 아래의 다수의 유닛 사이의 그러한 하위 그룹 제어 정보의 흐름은 명확성을 위해 도시되지 않는다.

[0047] 이미 언급된 기능 블록 외에도, 비디오 디코더(510)는 아래에서 설명되는 바와 같이 다수의 기능 유닛으로 개념적으로 세분될 수 있다. 상업적 제약 하에서 동작하는 실제 구현에서, 이러한 유닛 중 다수는 서로 밀접하게 상호 작용하고, 적어도 부분적으로 서로 통합될 수 있다. 그러나 개시된 청구 대상을 설명할 목적으로, 아래의 기능 유닛으로의 개념적 세분이 적절하다.

[0048] 제1 유닛은 스케일러/역변환 유닛(551)일 수 있다. 스케일러/역변환 유닛(551)은 파서(520)로부터의 심벌(들)(521)로서, 사용할 변환, 블록 크기, 양자화 팩터, 양자화 스케일링 행렬 등을 포함하는 제어 정보뿐만 아니라 양자화된 변환 계수를 수신한다. 스케일러/역변환 유닛(551)은 애그리게이터(aggregator)(555)에 입력될 수 있는 샘플 값을 포함하는 블록을 출력할 수 있다.

[0049] 일부 경우에서, 스케일러/역변환(551)의 출력 샘플은 인트라 코딩된 블록; 즉: 이전에 재구성된 픽처로부터의 예측 정보를 사용하고 있는 것이 아니라, 현재 픽처의 이전에 재구성된 부분으로부터의 예측 정보를 사용할 수 있는 블록에 속할 수 있다. 그러한 예측 정보는 인트라 픽처 예측 유닛(552)에 의해 제공될 수 있다. 일부 경우에서, 인트라 픽처 예측 유닛(552)은 현재 픽처 버퍼(558)로부터 페치(fetch)된 주변의 이미 재구성된 정보를 사용하여, 재구성 하의 블록과 동일한 크기 및 형상의 블록을 생성한다. 현재 픽처 버퍼(558)는 예를 들어, 부분적으로 재구성된 현재 픽처 및/또는 완전히 재구성된 현재 픽처를 버퍼링한다. 애그리게이터(555)는 일부 경우에서, 인트라 예측 유닛(552)이 생성한 예측 정보를 스케일러/역변환 유닛(551)에 의해 제공되는 출력 샘플 정보에 샘플 단위로 추가한다.

[0050] 다른 경우에서, 스케일러/역변환 유닛(551)의 출력 샘플은, 인터 코딩되고 잠재적으로 움직임 보상된 블록에 속할 수 있다. 그러한 경우에, 움직임 보상 예측 유닛(553)은 예측에 사용되는 샘플을 페치하기 위해 참조 픽처 메모리(557)에 액세스할 수 있다. 블록에 속하는 심벌(521)에 따라 페치된 샘플을 움직임 보상한 후에, 이러한 샘플은 애그리게이터(555)에 의해 (이 경우, 잔차 샘플 또는 잔차 신호로 지칭되는) 스케일러/역변환 유닛(55

1)의 출력에 추가되어 출력 샘플 정보를 생성할 수 있다. 움직임 보상 예측 유닛(553)이 예측 샘플을 폐치하는 참조 픽처 메모리(557) 내의 어드레스는 예를 들어 X, Y 및 참조 픽처 성분을 가질 수 있는 심벌(521)의 형태로 움직임 보상 예측 유닛(553)에 이용 가능한 움직임 벡터에 의해 제어될 수 있다. 움직임 보상은 또한, 서브샘플 정밀 움직임 벡터가 사용 중일 때 참조 픽처 메모리(557)로부터 폐치된 샘플 값의 보간, 움직임 벡터 예측 메커니즘 등을 포함할 수 있다.

[0051] 애그리게이터(555)의 출력 샘플은 루프 필터 유닛(556)에서 다양한 루프 필터링 기법을 겪을 수 있다. 비디오 압축 기술은 (코딩된 비디오 비트스트림으로도 또한 지칭되는) 코딩된 비디오 시퀀스에 포함된 파라미터에 의해 제어되고 파서(520)로부터의 심벌(521)로서 루프 필터 유닛(556)에 이용 가능하게 되는 루프 내 필터 기술을 포함할 수 있지만, 코딩된 픽처 또는 코딩된 비디오 시퀀스의 (디코딩 순서로) 이전 부분의 디코딩 동안 획득된 메타 정보에 응답할 뿐만 아니라, 이전에 재구성되고 루프 필터링된 샘플 값에 또한 응답할 수 있다.

[0052] 루프 필터 유닛(556)의 출력은 렌더링 디바이스(512)에 출력될 수 있을 뿐만 아니라 향후 인터 픽처 예측에서의 사용을 위해 참조 픽처 메모리(557)에 저장될 수 있는 샘플 스트림일 수 있다.

[0053] 특정 코딩된 픽처는 일단 완전히 재구성되면, 향후 예측을 위한 참조 픽처로서 사용될 수 있다. 예를 들어, 현재 픽처에 대응하는 코딩된 픽처가 완전히 재구성되고 코딩된 픽처가 (예를 들어, 파서(520)에 의해) 참조 픽처로서 식별되면, 현재 픽처 버퍼(558)는 참조 픽처 메모리(557)의 일부가 될 수 있고, 다음의 코딩된 픽처의 재구성을 시작하기 전에 새로운 현재 픽처 버퍼가 재할당될 수 있다.

[0054] 비디오 디코더(510)는 ITU-T Rec. H.265와 같은 표준의 미리 결정된 비디오 압축 기술에 따라 디코딩 동작을 수행할 수 있다. 코딩된 비디오 시퀀스는, 코딩된 비디오 시퀀스가 비디오 압축 기술 또는 표준에 문서화된 프로파일 및 비디오 압축 기술 또는 표준의 선택스 모두를 고수한다는 의미로, 사용되고 있는 비디오 압축 기술 또는 표준에 의해 특정된 선택스를 따를 수 있다. 구체적으로, 프로파일은 비디오 압축 기술 또는 표준에서 이용 가능한 모든 도구 중에서 해당 프로파일 하에서의 사용을 위해 이용 가능한 유일한 도구로서 특정 도구를 선택할 수 있다. 준수를 위해서는 또한, 코딩된 비디오 시퀀스의 복잡성은 비디오 압축 기술 또는 표준의 레벨에 의해 정의된 범위 내에 있다. 일부 경우에서, 레벨은 최대 픽처 크기, 최대 프레임 레이트, (예를 들어, 초당 메가샘플로 측정된) 최대 재구성 샘플 레이트, 최대 참조 픽처 크기 등을 제한한다. 레벨에 의해 설정된 제한은 일부 경우에는, 코딩된 비디오 시퀀스로 시그널링되는 가상 참조 디코더(HRD: Hypothetical Reference Decoder) 버퍼 관리를 위한 메타데이터(metadata) 및 HRD 규격을 통해 추가로 제한될 수 있다.

[0055] 일 실시예에서, 수신기(531)는 인코딩된 비디오와 함께 추가 (중복) 데이터를 수신할 수 있다. 추가 데이터는 코딩된 비디오 시퀀스(들)의 일부로서 포함될 수 있다. 추가 데이터는 데이터를 적절히 디코딩하기 위해 그리고/또는 원본 비디오 데이터를 보다 정확하게 재구성하기 위해 비디오 디코더(510)에 의해 사용될 수 있다. 추가 데이터는 예를 들어, 시간적, 공간적 또는 신호 잡음비(SNR: signal noise ratio) 강화 계층, 중복 슬라이스, 중복 픽처, 순방향 에러 정정 코드 등의 형태일 수 있다.

[0056] 도 6은 본 개시내용의 일 실시예에 따른 비디오 인코더(603)의 블록도를 도시한다. 비디오 인코더(603)는 전자 디바이스(620)에 포함된다. 전자 디바이스(620)는 전송기(640)(예컨대, 전송 회로)를 포함한다. 비디오 인코더(603)는 도 4의 예에서의 비디오 인코더(403) 대신에 사용될 수 있다.

[0057] 비디오 인코더(603)는 비디오 인코더(603)에 의해 코딩된 비디오 이미지(들)를 캡처할 수 있는 (도 6의 예에서 전자 디바이스(620)의 일부가 아닌) 비디오 소스(601)로부터 비디오 샘플을 수신할 수 있다. 다른 예에서, 비디오 소스(601)는 전자 디바이스(620)의 일부이다.

[0058] 비디오 소스(601)는 임의의 적절한 비트 깊이(예를 들어: 8비트, 10비트, 12비트, ...), 임의의 색 공간(예를 들어, BT.601 Y CrCb, RGB, ...) 및 임의의 적절한 샘플링 구조(예를 들어, Y CrCb 4:2:0, Y CrCb 4:4:4)일 수 있는 디지털 비디오 샘플 스트림의 형태로 비디오 인코더(603)에 의해 코딩된 소스 비디오 시퀀스를 제공할 수 있다. 미디어 서버 시스템에서, 비디오 소스(601)는 이전에 준비된 비디오를 저장하는 저장 디바이스일 수 있다. 화상 회의 시스템에서, 비디오 소스(601)는 로컬 이미지 정보를 비디오 시퀀스로서 캡처하는 카메라일 수 있다. 비디오 데이터는 순차적으로 볼 때 움직임을 부여하는 복수의 개별 픽처로서 제공될 수 있다. 픽처 자체는 픽셀의 공간 어레이로서 구성될 수 있으며, 여기서 각각의 픽셀은 사용 중인 샘플링 구조, 색 공간 등에 따라 하나 이상의 샘플을 포함할 수 있다. 당해 기술분야에서 통상의 지식을 가진 자는 픽셀과 샘플 간의 관계를 쉽게 이해할 수 있다. 아래의 설명은 샘플에 초점을 맞춘다.

[0059] 일 실시예에 따르면, 비디오 인코더(603)는 실시간으로 또는 애플리케이션에 의해 요구되는 임의의 다른 시간

제약 하에서 소스 비디오 시퀀스의 픽처를 코딩된 비디오 시퀀스(643)로 코딩 및 압축할 수 있다. 적절한 코딩 속도를 시행하는 것은 제어기(650)의 하나의 기능이다. 일부 실시예에서, 제어기(650)는 아래에서 설명되는 바와 같이 다른 기능 유닛을 제어하고, 다른 기능 유닛에 기능적으로 결합된다. 명확성을 위해 결합은 도시되지 않는다. 제어기(650)에 의해 설정된 파라미터는 레이트 제어 관련 파라미터(픽처 스킵(skip), 양자화기, 레이트 왜곡 최적화 기법의 람다(lambda) 값, ...), 픽처 크기, 픽처의 그룹(GOP) 레이아웃, 최대 움직임 벡터 탐색 범위 등을 포함할 수 있다. 제어기(650)는 특정 시스템 설계에 최적화된 비디오 인코더(603)에 속하는 다른 적절한 기능을 갖도록 구성될 수 있다.

[0060] 일부 실시예에서, 비디오 인코더(603)는 코딩 루프에서 동작하도록 구성된다. 지나치게 단순화된 설명으로서, 일례로, 코딩 루프는 (예컨대, 참조 픽처(들) 및 코딩될 입력 픽처에 기초하여 심벌 스트림과 같은 심벌의 생성을 담당하는) 소스 코더(630) 및 비디오 인코더(603)에 내장된 (로컬) 디코더(633)를 포함할 수 있다. 디코더(633)는 (개시된 청구 대상에서 고려되는 비디오 압축 기술에서 심벌과 코딩된 비디오 비트스트림 간의 임의의 압축은 무손실이기 때문에) (원격) 디코더가 또한 생성할 것과 유사한 방식으로 샘플 데이터를 생성하도록 심벌을 재구성한다. 재구성된 샘플 스트림(샘플 데이터)은 참조 픽처 메모리(634)에 입력된다. 심벌 스트림의 디코딩이 (로컬 또는 원격) 디코더 위치와 독립적인 비트 정확한 결과로 이어지므로, 참조 픽처 메모리(634) 내의 콘텐츠는 또한 로컬 인코더와 원격 인코더 사이에서 비트가 정확하다. 다시 말해서, 인코더의 예측 부분은, 디코딩 동안 예측을 사용할 때 디코더가 "보는" 것과 정확히 동일한 샘플 값을 참조 픽처 샘플로서 "본다". 참조 픽처 동기성의 이러한 기본 원리(및 예를 들어, 채널 에러로 인해 동기성이 유지될 수 없다면, 결과적인 드리프트(drift))는 일부 관련 기술에서도 사용된다.

[0061] "로컬" 디코더(633)의 동작은 도 5와 함께 위에서 이미 상세히 설명된 비디오 디코더(510)와 같은 "원격" 디코더의 동작과 동일할 수 있다. 그러나 도 5를 또한 간략하게 참조하면, 심벌이 이용 가능하고 엔트로피 코더(645) 및 파서(520)에 의한 코딩된 비디오 시퀀스로의 심벌의 인코딩/디코딩이 무손실일 수 있기 때문에, 버퍼 메모리(515) 및 파서(520)를 포함하는 비디오 디코더(510)의 엔트로피 디코딩 부분은 로컬 디코더(633)에서 완전히 구현되지 않을 수 있다.

[0062] 이 시점에서 이루어질 수 있는 관찰은, 디코더에 존재하는 파싱/엔트로피 디코딩을 제외한 임의의 디코더 기술이 실질적으로 동일한 기능적 형태로 대응하는 인코더에 반드시 존재할 필요가 있다는 것이다. 이런 이유로, 개시된 청구 대상은 디코더 동작에 초점을 맞춘다. 인코더 기술은 포괄적으로 설명된 디코더 기술의 역이기 때문에 인코더 기술의 설명은 축약될 수 있다. 특정 영역에서만 더 상세한 설명이 요구되고 아래에 제공된다.

[0063] 동작 중에, 일부 예에서, 소스 코더(630)는 "참조 픽처"로서 지정된 비디오 시퀀스로부터의 하나 이상의 이전에 코딩된 픽처를 참조하여 입력 픽처를 예측 코딩하는 움직임 보상 예측 코딩을 수행할 수 있다. 이런 식으로, 코딩 엔진(632)은 입력 픽처의 픽셀 블록과 입력 픽처에 대한 예측 참조(들)로서 선택될 수 있는 참조 픽처(들)의 픽셀 블록 간의 차이를 코딩한다.

[0064] 로컬 비디오 디코더(633)는 소스 코더(630)에 의해 생성된 심벌에 기반하여, 참조 픽처로서 지정될 수 있는 픽처의 코딩된 비디오 데이터를 디코딩할 수 있다. 코딩 엔진(632)의 동작은 유리하게는 손실 프로세스일 수 있다. 코딩된 비디오 데이터가 (도 6에 도시되지 않은) 비디오 디코더에서 디코딩될 수 있을 때, 재구성된 비디오 시퀀스는 통상적으로 일부 에러를 갖는 소스 비디오 시퀀스의 복제물일 수 있다. 로컬 비디오 디코더(633)는 비디오 디코더에 의해 참조 픽처에 수행될 수 있는 디코딩 프로세스를 복제하고, 재구성된 참조 픽처가 참조 픽처 캐시(634)에 저장되게 할 수 있다. 이런 식으로, 비디오 인코더(603)는 (전송 에러 없이) 원단(far-end) 비디오 디코더에 의해 획득될 재구성된 참조 픽처로서 공통 콘텐츠를 갖는 재구성된 참조 픽처의 사본을 국소적으로 저장할 수 있다.

[0065] 예측기(635)는 코딩 엔진(632)에 대한 예측 탐색을 수행할 수 있다. 즉, 코딩될 새로운 픽처에 대해, 예측기(635)는 (후보 참조 픽셀 블록으로서) 샘플 데이터, 또는 새로운 픽처에 대한 적절한 예측 참조로서의 역할을 할 수 있는 참조 픽처 움직임 벡터, 블록 형상 등과 같은 특정 메타데이터에 대해 참조 픽처 메모리(634)를 탐색할 수 있다. 예측기(635)는 적절한 예측 참조를 발견하기 위해 샘플 블록 × 픽셀 블록 단위로 동작할 수 있다. 일부 경우에는, 예측기(635)에 의해 획득된 탐색 결과에 의해 결정된 대로, 입력 픽처는 참조 픽처 메모리(634)에 저장된 다수의 참조 픽처로부터 인출된 예측 참조를 가질 수 있다.

[0066] 제어기(650)는 예를 들어, 비디오 데이터를 인코딩하기 위해 사용되는 파라미터 및 하위 그룹 파라미터의 설정을 포함하는, 소스 코더(630)의 코딩 동작을 관리할 수 있다.

- [0067] 앞서 언급된 모든 기능 유닛의 출력은 엔트로피 코더(645)에서 엔트로피 코딩을 거칠 수 있다. 엔트로피 코더(645)는 허프만 코딩, 가변 길이 코딩, 산술 코딩 등은 기술에 따라 심벌을 무손실 압축함으로써, 다양한 기능 유닛에 의해 생성된 심벌을 코딩된 비디오 시퀀스로 변환한다.
- [0068] 전송기(640)는 엔트로피 코더(645)에 의해 생성된 코딩된 비디오 시퀀스(들)를 버퍼링하여, 인코딩된 비디오 데이터를 저장할 저장 디바이스에 대한 하드웨어/소프트웨어 링크일 수 있는 통신 채널(660)을 통한 전송을 준비할 수 있다. 전송기(640)는 비디오 코더(603)로부터의 코딩된 비디오 데이터를 전송될 다른 데이터, 예를 들어 코딩된 오디오 데이터 및/또는 보조 데이터 스트림(소스는 도시되지 않음)과 병합할 수 있다.
- [0069] 제어기(650)는 비디오 인코더(603)의 동작을 관리할 수 있다. 코딩 동안, 제어기(650)는 각각의 코딩된 픽처에 특정 코딩된 픽처 타입을 할당할 수 있으며, 이는 개개의 픽처에 적용될 수 있는 코딩 기술에 영향을 미칠 수 있다. 예를 들어, 픽처는 종종 다음의 픽처 타입 중 하나로서 할당될 수 있다:
- [0070] 인트라 픽처(I 픽처)는 예측의 소스로서 시퀀스 내의 임의의 다른 픽처를 사용하지 않고 코딩 및 디코딩될 수 있는 것일 수 있다. 일부 비디오 코덱은 예를 들어, 독립적 디코더 리프레시(IDR: Independent Decoder Refresh) 픽처를 포함하는 상이한 타입의 인트라 픽처를 허용한다. 당해 기술분야에서 통상의 지식을 가진 자는 I 픽처의 그러한 변형 그리고 이들 개개의 애플리케이션 및 특징을 알고 있다.
- [0071] 예측 픽처(P 픽처)는 각각의 블록의 샘플 값을 예측하기 위해 최대 하나의 움직임 벡터 및 참조 인덱스를 사용하는 인트라 예측 또는 인터 예측을 사용하여 코딩 및 디코딩될 수 있는 것일 수 있다.
- [0072] 양방향 예측 픽처(B 픽처)는 각각의 블록의 샘플 값을 예측하기 위해 최대 2개의 움직임 벡터 및 참조 인덱스를 사용하는 인트라 예측 또는 인터 예측을 사용하여 코딩 및 디코딩될 수 있는 것일 수 있다. 유사하게, 다수의 예측 픽처는 단일 블록의 재구성을 위해 2개보다 많은 참조 픽처 및 연관된 메타데이터를 사용할 수 있다.
- [0073] 소스 픽처는 일반적으로, 공간적으로 복수의 샘플 블록(예를 들어, 각각 4×4, 8×8, 4×8 또는 16×16 샘플의 블록)으로 세분되고 블록 단위로 코딩될 수 있다. 블록은 블록의 개개의 픽처에 적용된 코딩 할당에 의해 결정된 다른 (이미 코딩된) 블록을 참조하여 예측 코딩될 수 있다. 예를 들어, I 픽처의 블록은 비-예측 코딩될 수 있거나, 이들은 동일한 픽처의 이미 코딩된 블록을 참조하여 예측(공간 예측 또는 인트라 예측) 코딩될 수 있다. P 픽처들의 픽셀 블록들은 하나의 이전에 코딩된 참조 픽처를 참조하여 시간 예측을 통해 또는 공간 예측을 통해 예측 코딩될 수 있다. B 픽처의 블록은 하나 또는 2개의 이전에 코딩된 참조 픽처를 참조하여 시간 예측을 통해 또는 공간 예측을 통해 예측 코딩될 수 있다.
- [0074] 비디오 인코더(603)는 ITU-T Rec. H.265와 같은 미리 결정된 비디오 코딩 기술 또는 표준에 따라 코딩 동작을 수행할 수 있다. 비디오 인코더(603)의 동작에서, 비디오 인코더(603)는 입력 비디오 시퀀스에서 시간적 및 공간적 중복성을 활용하는 예측 코딩 동작을 포함하는 다양한 압축 동작을 수행할 수 있다. 따라서 코딩된 비디오 데이터는 사용되는 비디오 코딩 기술 또는 표준에 의해 특정된 신택스를 따를 수 있다.
- [0075] 일 실시예에서, 전송기(640)는 인코딩된 비디오와 함께 추가 데이터를 전송할 수 있다. 소스 코더(630)는 코딩된 비디오 시퀀스의 일부로서 이러한 데이터를 포함할 수 있다. 추가 데이터는 시간적/공간적/SNR 강화 계층, 다른 형태의 중복 데이터, 이를테면 중복 픽처 및 슬라이스, SEI 메시지, VUI 파라미터 세트 프래그먼트 등을 포함할 수 있다.
- [0076] 비디오는 시간 순서로 복수의 소스 픽처(비디오 픽처)로서 캡처될 수 있다. (종종 인트라 예측으로 축약되는) 인트라 픽처 예측은 주어진 픽처에서 공간적 상관을 사용하고, 인터 픽처 예측은 픽처 간의 (시간적 또는 다른) 상관을 사용한다. 일례로, 현재 픽처로 지칭되는 인코딩/디코딩 하의 특정 픽처가 블록으로 파티셔닝된다. 현재 픽처 내의 블록이 비디오에서 이전에 코딩되고 여전히 버퍼링된 참조 픽처 내의 참조 블록과 유사한 경우, 현재 픽처의 블록은 움직임 벡터로 지칭되는 벡터에 의해 코딩될 수 있다. 움직임 벡터는 참조 픽처 내의 참조 블록을 가리키며, 다수의 참조 픽처가 사용 중인 경우, 참조 픽처를 식별하는 3차원을 가질 수 있다.
- [0077] 일부 실시예에서, 양방향 예측 기법이 인터 픽처 예측에서 사용될 수 있다. 양방향 예측 기법에 따르면, 비디오의 현재 픽처에 대해 디코딩 순서가 둘 다 앞선(그러나 디스플레이 순서에서 각각 과거 및 미래일 수 있는) 제1 참조 픽처 및 제2 참조 픽처와 같은 2개의 참조 픽처가 사용된다. 현재 픽처 내의 블록은 제1 참조 픽처 내의 제1 참조 블록을 가리키는 제1 움직임 벡터 및 제2 참조 픽처 내의 제2 참조 블록을 가리키는 제2 움직임 벡터에 의해 코딩될 수 있다. 블록은 제1 참조 블록과 제2 참조 블록의 조합에 의해 예측될 수 있다.
- [0078] 또한, 코딩 효율을 향상시키기 위해 인터 픽처 예측에 병합 모드 기법이 사용될 수 있다.

- [0079] 본 개시내용의 일부 실시예에 따르면, 인터 픽처 예측 및 인트라 픽처 예측과 같은 예측이 블록 단위로 수행된다. 예를 들어, HEVC 표준에 따르면, 비디오 픽처 시퀀스의 픽처는 압축을 위해 코딩 트리 유닛(CTU: coding tree unit)으로 분할되며, 픽처의 CTU는 64×64 픽셀, 32×32 픽셀 또는 16×16 픽셀과 같은 동일한 크기를 갖는다. 일반적으로, CTU는 1개의 루마(luma) 코딩 트리 블록(CTB: coding tree block)과 2개의 크로마(chroma) CTB인 3개의 CTB를 포함한다. 각각의 CTU는 하나 또는 다수의 코딩 유닛(CU)으로 재귀적으로 쿼드 트리 분할될 수 있다. 예를 들어, 64×64 픽셀의 CTU는 64×64 픽셀의 하나의 CU, 또는 32×32 픽셀의 4개의 CU, 또는 16×16 픽셀의 16개의 CU로 분할될 수 있다. 일례로, 각각의 CU는 분석되어 인터 예측 타입 또는 인트라 예측 타입과 같은 CU에 대한 예측 타입을 결정한다. CU는 시간적 그리고/또는 공간적 예측 가능성에 따라 하나 이상의 예측 유닛(PU)으로 분할된다. 일반적으로, 각각의 PU는 루마 예측 블록(PB: prediction block)과 2개의 크로마 PB를 포함한다. 일 실시예에서, 코딩(인코딩/디코딩)에서의 예측 동작은 예측 블록 단위로 수행된다. 예측 블록의 예로서 루마 예측 블록을 사용하면, 예측 블록은 8×8 픽셀, 16×16 픽셀, 8×16 픽셀, 16×8 픽셀 등과 같은 픽셀에 대한 값(예컨대, 루마 값)의 행렬을 포함한다.
- [0080] 도 7은 본 개시내용의 다른 실시예에 따른 비디오 인코더(703)의 도면을 도시한다. 비디오 인코더(703)는 비디오 픽처의 시퀀스에서 현재 비디오 픽처 내의 샘플 값의 처리 블록(예컨대, 예측 블록)을 수신하고, 그 처리 블록을 코딩된 픽처의 일부인 코딩된 픽처로 인코딩하도록 구성된다. 일례로, 비디오 인코더(703)는 도 4의 예에서의 비디오 인코더(403) 대신에 사용된다.
- [0081] HEVC 예에서, 비디오 인코더(703)는 8x8 샘플의 예측 블록 등과 같은 처리 블록에 대한 샘플 값의 행렬을 수신한다. 비디오 인코더(703)는 예를 들어, 레이트 왜곡 최적화를 사용하여 처리 블록이 인트라 모드를 사용하여 가장 잘 코딩되는지, 인터 모드를 사용하여 가장 잘 코딩되는지 또는 양방향 예측 모드를 사용하여 가장 잘 코딩되는지 여부를 결정한다. 처리 블록이 인트라 모드에서 코딩되어야 할 때, 비디오 인코더(703)는 처리 블록을 코딩된 픽처로 인코딩하기 위한 인트라 예측 기술을 사용할 수 있고; 처리 블록이 인터 모드 또는 양방향 예측 모드에서 코딩되어야 할 때, 비디오 인코더(703)는 처리 블록을 코딩된 픽처로 인코딩하기 위해 인터 예측 또는 양방향 예측 기술을 각각 사용할 수 있다. 특정 비디오 코딩 기술에서, 병합 모드는 움직임 벡터가 예측자 외부의 코딩된 움직임 벡터 성분의 이점 없이 하나 이상의 움직임 벡터 예측자로부터 도출되는 인터 픽처 예측 하위 모드일 수 있다. 특정 다른 비디오 코딩 기술에서, 대상 블록에 적용 가능한 움직임 벡터 성분이 존재할 수 있다. 일례로, 비디오 인코더(703)는 처리 블록의 모드를 결정하기 위한 (도시되지 않은) 모드 결정 모듈과 같은 다른 컴포넌트를 포함한다.
- [0082] 도 7의 예에서, 비디오 인코더(703)는 도 7에 도시된 바와 같이 함께 결합되는 인터 인코더(730), 인트라 인코더(722), 잔차 계산기(723), 스위치(726), 잔차 인코더(724), 일반 제어기(721) 및 엔트로피 인코더(725)를 포함한다.
- [0083] 인터 인코더(730)는 현재 블록(예컨대, 처리 블록)의 샘플을 수신하고, 블록을 참조 픽처의 하나 이상의 참조 블록(예컨대, 이전 픽처 및 이후 픽처의 블록)과 비교하고, 인터 예측 정보(예컨대, 인터 인코딩 기술에 따른 중복 정보의 설명, 움직임 벡터, 병합 모드 정보), 및 임의의 적절한 기법을 사용하여 인터 예측 정보를 기초로 인터 예측 결과(예컨대, 예측된 블록)를 계산하도록 구성된다. 일부 예에서, 참조 픽처는 인코딩된 비디오 정보에 기초하여 디코딩되는 디코딩된 참조 픽처이다.
- [0084] 인트라 인코더(722)는 현재 블록(예컨대, 처리 블록)의 샘플을 수신하고, 일부 경우에 블록을 동일한 픽처에 이미 코딩된 블록과 비교하고, 변환 이후 양자화된 계수, 및 일부 경우에는 또한 인트라 예측 정보(예컨대, 하나 이상의 인트라 인코딩 기법에 따른 인트라 예측 방향 정보)를 생성하도록 구성된다. 일례로, 인트라 인코더(722)는 또한 동일한 픽처의 인트라 예측 정보 및 참조 블록에 기초하여 인트라 예측 결과(예컨대, 예측된 블록)를 계산한다.
- [0085] 일반 제어기(721)는, 일반 제어 데이터를 결정하고 일반 제어 데이터에 기초하여 비디오 인코더(703)의 다른 컴포넌트를 제어하도록 구성된다. 일례로, 일반 제어기(721)는 블록의 모드를 결정하고, 그 모드에 기초하여 스위치(726)에 제어 신호를 제공한다. 예를 들어, 모드가 인트라 모드인 경우, 일반 제어기(721)는 잔차 계산기(723)에 의해 사용할 인트라 모드 결과를 선택하도록 스위치(726)를 제어하고, 인트라 예측 정보를 선택하고 비트스트림에 인트라 예측 정보를 포함시키도록 엔트로피 인코더(725)를 제어하고; 모드가 인터 모드인 경우, 일반 제어기(721)는 잔차 계산기(723)에 의해 사용할 인터 예측 결과를 선택하도록 스위치(726)를 제어하고, 인터 예측 정보를 선택하고 비트스트림에 인터 예측 정보를 포함시키도록 엔트로피 인코더(725)를 제어한다.
- [0086] 잔차 계산기(723)는 수신된 블록과 인트라 인코더(722) 또는 인터 인코더(730)로부터 선택된 예측 결과 사이의

차이(잔차 데이터)를 계산하도록 구성된다. 잔차 인코더(724)는 잔차 데이터를 인코딩하여 변환 계수를 생성하기 위해 잔차 데이터를 기초로 동작하도록 구성된다. 일례로, 잔차 인코더(724)는 잔차 데이터를 공간 도메인에서 주파수 도메인으로 변환하고 변환 계수를 생성하도록 구성된다. 그 다음, 변환 계수는 양자화 처리를 거쳐 양자화된 변환 계수를 얻는다. 다양한 실시예에서, 비디오 인코더(703)는 또한 잔차 디코더(728)를 포함한다. 잔차 디코더(728)는 역변환을 수행하고 디코딩된 잔차 데이터를 생성하도록 구성된다. 디코딩된 잔차 데이터는 인트라 인코더(722) 및 인터 인코더(730)에 의해 적절하게 사용될 수 있다. 예를 들어, 인터 인코더(730)는 디코딩된 잔차 데이터 및 인터 예측 정보에 기초하여 디코딩된 블록을 생성할 수 있고, 인트라 인코더(722)는 디코딩된 잔차 데이터 및 인트라 예측 정보에 기초하여 디코딩된 블록을 생성할 수 있다. 디코딩된 블록은 디코딩된 픽처를 생성하도록 적절하게 처리되고, 디코딩된 픽처는 (도시되지 않은) 메모리 회로에 버퍼링될 수 있고 일부 예에서 참조 픽처로서 사용될 수 있다.

[0087] 엔트로피 인코더(725)는 인코딩된 블록을 포함하게 비트스트림을 포맷화하도록 구성된다. 엔트로피 인코더(725)는 HEVC 표준과 같은 적절한 표준에 따라 다양한 정보를 포함하도록 구성된다. 일례로, 엔트로피 인코더(725)는 비트스트림에 일반 제어 데이터, 선택된 예측 정보(예컨대, 인트라 예측 정보 또는 인터 예측 정보), 잔차 정보 및 다른 적절한 정보를 포함하도록 구성된다. 개시된 청구 대상에 따르면, 인터 모드 또는 양방향 예측 모드의 병합 하위 모드에서 블록을 코딩할 때는, 잔차 정보가 없다는 점에 주목한다.

[0088] 도 8은 본 개시내용의 다른 실시예에 따른 비디오 디코더(810)의 도면을 도시한다. 비디오 디코더(810)는 코딩된 비디오 시퀀스의 일부인 코딩된 픽처를 수신하고, 코딩된 픽처를 디코딩하여 재구성된 픽처를 생성하도록 구성된다. 일례로, 비디오 디코더(810)는 도 4의 예에서의 비디오 디코더(410) 대신에 사용된다.

[0089] 도 8의 예에서, 비디오 디코더(810)는 도 8에 도시된 바와 같이 함께 결합되는 엔트로피 디코더(871), 인터 디코더(880), 잔차 디코더(873), 재구성 모듈(874) 및 인트라 디코더(872)를 포함한다.

[0090] 엔트로피 디코더(871)는 코딩된 픽처가 구성되는 선택스 엘리먼트를 나타내는 특정 심벌을 코딩된 픽처로부터 재구성하도록 구성될 수 있다. 이러한 심벌은 예를 들어, (예를 들어, 인트라 모드, 인터 모드, 양방향 예측 모드, 병합 하위 모드 또는 다른 하위 모드에서의 후자의 2개의 모드와 같은) 블록이 코딩되는 모드, 인트라 디코더(872) 또는 인터 디코더(880) 각각에 의해 예측에 사용되는 특정 샘플 또는 메타데이터를 식별할 수 있는 (예를 들어, 인트라 예측 정보 또는 인터 예측 정보와 같은) 예측 정보, 예를 들어 양자화된 변환 계수 형태의 잔차 정보 등을 포함할 수 있다. 일례로, 예측 모드가 인터 또는 양방향 예측 모드인 경우, 인터 예측 정보가 인터 디코더(880)에 제공되고; 예측 타입이 인트라 예측 타입인 경우, 인트라 예측 정보가 인트라 디코더(872)에 제공된다. 잔차 정보는 역양자화를 거칠 수 있고 잔차 디코더(873)에 제공된다.

[0091] 인터 디코더(880)는 인터 예측 정보를 수신하고, 인터 예측 정보에 기초하여 인터 예측 결과를 생성하도록 구성된다.

[0092] 인트라 디코더(872)는 인트라 예측 정보를 수신하고, 인트라 예측 정보에 기초하여 예측 결과를 생성하도록 구성된다.

[0093] 잔차 디코더(873)는 역양자화를 수행하여 역양자화된 변환 계수를 추출하고, 역양자화된 변환 계수를 처리하여 주파수 도메인에서 공간 도메인으로 잔차를 변환하도록 구성된다. 잔류 디코더(873)는 또한 (양자화기 파라미터(QP: Quantizer Parameter)를 포함하도록) 특정 제어 정보를 요구할 수 있으며, 해당 정보는 엔트로피 디코더(871)에 의해 제공될 수 있다(데이터 경로는 단지 낮은 볼륨 제어 정보일 수 있으므로 이는 도시되지 않음).

[0094] 재구성 모듈(874)은 공간 도메인에서 잔차 디코더(873)에 의해 출력된 잔차와 (일부 경우에 인터 또는 인트라 예측 모듈에 의해 출력된) 예측 결과를 조합하여 재구성된 블록을 형성하도록 구성되는데, 재구성된 블록은 재구성된 픽처의 일부일 수 있고, 재구성된 픽처는 결국, 재구성된 비디오의 일부일 수 있다. 블록 분리(deblocking) 동작과 같은 다른 적절한 동작이 시각적 품질을 개선하기 위해 수행될 수 있다는 점이 주목된다.

[0095] 비디오 인코더(403, 603, 703) 및 비디오 디코더(410, 510, 810)는 임의의 적절한 기법을 사용하여 구현될 수 있다는 점이 주목된다. 일 실시예에서, 비디오 인코더(403, 603, 703) 및 비디오 디코더(410, 510, 810)는 하나 이상의 집적 회로를 사용하여 구현될 수 있다. 다른 실시예에서, 비디오 인코더(403, 603, 703) 및 비디오 디코더(410, 510, 810)는 소프트웨어 명령을 실행하는 하나 이상의 프로세서를 사용하여 구현될 수 있다.

[0096] 2차 변환 인덱스와 같은 인덱스의 엔트로피 코딩을 위한 컨텍스트 설계와 관련된 비디오 코딩 기술이 개시된다. 인덱스는 블록을 코딩(예컨대, 인코딩 및/또는 디코딩)하는 데 사용되는 한 세트의 비분리 2차 변환 중 하나를 식별하는 데 사용될 수 있다. 인덱스의 엔트로피 코딩을 위한 컨텍스트 설계는 임의의 적절한 비디오 코딩 포

맷 또는 표준에 적용 가능할 수 있다. 비디오 코딩 포맷은 AOMedia Video 1(AV1) 또는 AV1 이상의 차세대 AOMedia Video 포맷과 같이 인터넷을 통한 비디오 전송을 위해 설계된 개방형 비디오 코딩 포맷을 포함할 수 있다. 비디오 코딩 표준은 고효율 비디오 코딩(HEVC: High Efficiency Video Coding) 표준, HEVC 이상의 차세대 비디오 코딩(예컨대, 다용도 비디오 코딩(VVC)) 등을 포함할 수 있다.

[0097] 인트라 예측에, 예를 들어, AV1, VVC 등에 다양한 인트라 예측 모드가 사용될 수 있다. AV1과 같은 일 실시예에서는, 방향성 인트라 예측이 사용된다. 방향성 인트라 예측에서, 블록에 대한 예측된 샘플은 방향을 따라 이웃하는 재구성된 샘플로부터 외삽함으로써 생성될 수 있다. 방향은 각도에 대응한다. 블록에 대한 예측된 샘플을 예측하기 위해 방향성 인트라 예측에 사용되는 모드는 (방향성 예측 모드, 방향성 인트라 모드, 방향성 인트라 예측 모드, 각도 모드로도 또한 지칭되는) 방향성 모드로 지칭될 수 있다. 각각의 방향성 모드는 상이한 각도 또는 상이한 방향에 대응할 수 있다. 개방형 비디오 코딩 포맷 VP9와 같은 일례에서는, 도 9에 도시된 바와 같이, 8개의 방향성 모드가 45° 내지 207° 의 8개의 각도에 대응한다. 8개의 방향성 모드는 또한 공칭 모드(예컨대, V_PRED, H_PRED, D45_PRED, D135_PRED, D113_PRED, D157_PRED, D203_PRED, D67_PRED)로도 지칭될 수 있다. 방향성 텍스처에서 보다 다양한 공간 중복성을 활용하기 위해, 예를 들어 AV1에서, 방향성 모드는 예를 들어, 도 9에 도시된 바와 같이, 8개의 공칭 모드를 넘어 더 미세한 입도로 설정된 각도 및 더 많은 각도(또는 방향)으로 확장될 수 있다.

[0098] 도 9는 본 개시내용의 일 실시예에 따른 코딩 블록(CB: coding block)(910)에 대한 공칭 모드의 일례를 도시한다. (공칭 각도로 지칭되는) 특정 각도는 공칭 모드에 대응할 수 있다. 일례로, 8개의 공칭 각도(또는 공칭 인트라 각도)(901-908)는 8개의 공칭 모드(예컨대, V_PRED, H_PRED, D45_PRED, D135_PRED, D113_PRED, D157_PRED, D203_PRED, and D67_PRED)에 각각 대응한다. 8개의 공칭 각도(901-908)뿐만 아니라 8개의 공칭 모드는 각각 V_PRED, H_PRED, D45_PRED, D135_PRED, D113_PRED, D157_PRED, D203_PRED 및 D67_PRED로 지칭될 수 있다. 공칭 모드 인덱스는 공칭 모드(예컨대, 8개의 공칭 모드 중 하나)를 지시할 수 있다. 일례로, 공칭 모드 인덱스가 시그널링된다.

[0099] 또한, 각각의 공칭 각도는 복수의 더 미세한 각도(예컨대, 7개의 더 미세한 각도)에 대응할 수 있고, 따라서 56개의 각도(또는 예측 각도) 또는 56개의 방향성 모드(또는 각도 모드, 방향성 인트라 예측 모드)가 예를 들어, AV1에서 사용될 수 있다. 각각의 예측 각도는 공칭 각도 및 각도 오프셋(또는 각도 델타(delta))으로 제시될 수 있다. 각도 오프셋은 오프셋 정수 I(예컨대, -3, -2, -1, 0, 1, 2 또는 3)와 단계 크기(예컨대, 3°)를 곱함으로써 구해질 수 있다. 일례로, 예측 각도는 공칭 각도와 각도 오프셋의 합과 같다. AV1과 같은 일례에서, 특정 비-각도 평활(smooth) 모드(예컨대, 아래에서 설명되는 바와 같이, DC 모드, PAETH 모드, SMOOTH 모드, 수직 SMOOTH 모드 및 수평 SMOOTH 모드와 같은 5개의 비-각도 평활 모드)와 함께 공칭 모드(예컨대, 8개의 공칭 모드(901-908))가 시그널링될 수 있다. 이후, 현재 예측 모드가 방향성 모드(또는 각도 모드)라면, 공칭 각도에 대응하는 각도 오프셋(예컨대, 오프셋 정수 I)을 지시하기 위해 인덱스가 추가로 시그널링될 수 있다. 일례로, 방향성 모드(예컨대, 56개의 방향성 모드 중 하나)는 공칭 모드 인덱스 및 공칭 모드로부터의 각도 오프셋을 지시하는 인덱스에 기초하여 결정될 수 있다. 일례로, 일반적인 방식을 통해 방향성 예측 모드를 구현하기 위해, 이를테면 AV1에 사용되는 56개의 방향성 모드는, 각각의 픽셀을 참조 서브픽셀 위치에 투사하고 2-탭 양방향 선형 필터에 의해 참조 픽셀을 보간할 수 있는 통합 방향성 예측자로 구현된다.

[0100] (무지향성 평활 인트라 예측 모드, 무지향성 평활 모드, 비-각도 평활 모드로도 또한 지칭되는) 무지향성 평활 인트라 예측자는 CB와 같은 블록에 대한 인트라 예측에 사용될 수 있다. 일부 예에서(예컨대, AV1에서), 5개의 무지향성 평활 인트라 예측 모드는 DC 모드 또는 DC 예측자(예컨대, DC), PAETH 모드 또는 PAETH 예측자(예컨대, PAETH), SMOOTH 모드 또는 SMOOTH 예측자(예컨대, SMOOTH), (SMOOTH_V 모드, SMOOTH_V 예측자, SMOOTH_V로 지칭되는) 수직 SMOOTH 모드 및 (SMOOTH_H 모드, SMOOTH_H 예측자 또는 SMOOTH_H로 지칭되는) 수평 SMOOTH 모드를 포함한다.

[0101] 도 10은 본 개시내용의 양상에 따른 무지향성 평활 인트라 예측 모드(예컨대, DC 모드, PAETH 모드, SMOOTH 모드, SMOOTH_V 모드 및 SMOOTH_H 모드)에 대한 예를 도시한다. DC 예측자를 기반으로 CB(1000)에서 샘플(1001)을 예측하기 위해, 왼쪽 이웃 샘플(1012)의 첫 번째 값과 위쪽 이웃 샘플(또는 최상부 이웃 샘플)(1011)의 두 번째 값의 평균이 예측자로서 사용될 수 있다.

[0102] PAETH 예측자를 기반으로 샘플(1001)을 예측하기 위해, 왼쪽 이웃 샘플(1012)의 첫 번째 값, 최상부 이웃 샘플(1011)의 두 번째 값, 및 왼쪽 최상부 이웃 샘플(1013)에 대한 세 번째 값이 구해질 수 있다. 그런 다음, 식 1을 사용하여 기준 값이 구해진다.

- [0103] 기준 값 = 첫 번째 값 + 두 번째 값 - 세 번째 값 (식 1)
- [0104] 첫 번째 값, 두 번째 값 및 세 번째 값 중 기준 값에 가장 가까운 값이 샘플(1001)에 대한 예측자로서 설정될 수 있다.
- [0105] SMOOTH_V 모드, SMOOTH_H 모드 및 SMOOTH 모드는 각각 수직 방향, 수평 방향, 및 수직 방향과 수평 방향의 평균으로의 2차 보간을 사용하여 CB(1000)를 예측할 수 있다. SMOOTH 예측자를 기반으로 샘플(1001)을 예측하기 위해, 첫 번째 값, 두 번째 값, 오른쪽 샘플(1014)의 값 및 최하부 샘플(1016)의 값의 평균(예컨대, 가중 조합)이 사용될 수 있다. 다양한 예에서, 오른쪽 샘플(1014) 및 최하부 샘플(1016)은 재구성되지 않으며, 따라서 최상부 오른쪽 이웃 샘플(1015)의 값과 최하부 왼쪽 이웃 샘플(1017)의 값이 오른쪽 샘플(1014)과 최하부 샘플(1016)의 값을 각각 대체할 수 있다. 이에 따라, 첫 번째 값, 두 번째 값, 최상부 오른쪽 이웃 샘플(1015)의 값, 및 왼쪽 최하부 이웃 샘플(1017)의 값의 평균(예컨대, 가중 조합)이 또한 SMOOTH 예측자로서 사용될 수 있다. SMOOTH_V 예측자를 기반으로 샘플(1001)을 예측하기 위해, 최상부 이웃 샘플(1011)의 두 번째 값과 최하부 왼쪽 이웃 샘플(1017)의 값의 평균(예컨대, 가중 조합)이 사용될 수 있다. SMOOTH_H 예측자를 기반으로 샘플(1001)을 예측하기 위해, 왼쪽 이웃 샘플(1012)의 첫 번째 값과 최상부 오른쪽 이웃 샘플(1015) 값의 평균(예컨대, 가중 조합)이 사용될 수 있다.
- [0106] 도 11은 본 개시내용의 일 실시예에 따른 (필터 인트라 모드 또는 재귀적 필터링 모드로도 또한 지칭되는) 재귀적 필터링 기반 인트라 예측자의 일례를 도시한다. 예지에 대한 참조와 감쇠하는 공간 상관 관계를 캡처하기 위해, CB(1100)와 같은 블록에 필터 인트라 모드가 사용될 수 있다. 일례로, CB(1100)는 루마 블록이다. 루마 블록(1100)은 다수의 패치(patch)(예컨대, 8개의 4×2 패치(B0-B7))로 분할될 수 있다. 패치(B0-B7) 각각은 복수의 이웃 샘플을 가질 수 있다. 예를 들어, 패치(B0)는 4개의 최상부 이웃 샘플(R01-R04), 2개의 왼쪽 이웃 샘플(R05-R06) 및 최상부 왼쪽 이웃 샘플(R00)을 포함하는 7개의 이웃 샘플(또는 7개의 이웃)(R00-R06)을 갖는다. 유사하게, 패치(B7)는 4개의 최상부 이웃 샘플(R71-R74), 2개의 왼쪽 이웃 샘플(R75-R76) 및 최상부 왼쪽 이웃 샘플(R70)을 포함하는 7개의 이웃 샘플(R70-R76)을 갖는다.
- [0107] 일부 예에서, 다수(예컨대, 5개)의 필터 인트라 모드(또는 다수의 재귀적 필터링 모드)가 예를 들어, AV1에 대해 미리 설계된다. 각각의 필터 인트라 모드는 대응하는 4×2 패치(예컨대, B0)의 샘플(또는 픽셀)과 4×2 패치(B0)에 인접한 7개의 이웃(예컨대, R00-R06) 간의 상관을 반영하는 8개의 7-탭 필터의 세트에 표현될 수 있다. 7-탭 필터에 대한 가중 인자는 포지션 의존적일 수 있다. 패치(B0-B7) 각각에 대해, 7개의 이웃(예컨대, B0의 경우 R00-R06, B7의 경우 R70-R76)이 해당 패치의 샘플을 예측하는 데 사용될 수 있다. 일례로, 이웃(R00-R06)은 패치(B0)의 샘플을 예측하는 데 사용된다. 일례로, 이웃(R70-R76)은 패치(B7)의 샘플을 예측하는 데 사용된다. CB(1100)의 특정 패치, 이를테면 패치(B0)에 대해, 7개의 모든 이웃(예컨대, R00-R06)이 이미 재구성되었다. CB(1100)의 다른 패치의 경우, 7개의 이웃 중 적어도 하나는 재구성되지 않으며, 따라서 바로 이웃(들)(또는 바로 이웃(들)의 예측 샘플(들))의 예측 값(들)이 참조(들)로서 사용된다. 예를 들어, 패치(B7)의 7개의 이웃(R70-R76)은 재구성되지 않으므로, 바로 이웃의 예측 샘플이 사용될 수 있다.
- [0108] 크로마 샘플은 루마 샘플로부터 예측될 수 있다. 일 실시예에서, 루마 모드로부터의 크로마(예컨대, CfL 모드, CfL 예측자)는 일치하는 재구성된 루마 샘플(또는 픽셀)의 선형 함수로서 크로마 샘플(또는 픽셀)을 모델링할 수 있는 크로마 전용 인트라 예측자이다. 예를 들어, CfL 예측은 아래와 같은 식 2를 이용하여 표현될 수 있다.
- [0109]
$$CfL(a) = a L^A + D \quad (\text{식 } 2)$$
- [0110] 여기서 L^A 는 루마 성분의 AC 기여도(contribution)를 나타내고, a 는 선형 모델의 스케일링 파라미터를 나타내며, D 는 크로마 성분의 DC 기여도를 나타낸다. 일례로, 재구성된 루마 픽셀은 크로마 분해능에 기반하여 서브 샘플링되고, 평균 값이 감소되어 AC 기여도(예컨대, L^A)를 형성한다. 일부 예에서, 이를테면 AV1에서, 디코더가 스케일링 파라미터(a)를 계산할 것을 요구하는 대신에, AC 기여도로부터 크로마 AC 성분을 근사화하기 위해, CfL 모드는 원래의 크로마 픽셀에 기초하여 스케일링 파라미터(a)를 결정하고 비트스트림에서 스케일링 파라미터(a)를 시그널링하여, 디코더 복잡성을 줄이고 보다 정확한 예측을 산출한다. 크로마 성분의 DC 기여도는 인트라 DC 모드를 사용하여 계산될 수 있다. 인트라 DC 모드는 대부분의 크로마 콘텐츠에 대해 충분할 수 있으며 완성도 높은 빠른 구현을 할 수 있다.
- [0111] 다중 라인 인트라 예측은 인트라 예측을 위해 더 많은 기준선을 사용할 수 있다. 기준선은 픽처에 다수의 샘플

을 포함할 수 있다. 일례로, 기준선은 행의 샘플과 열의 샘플을 포함한다. 일례로, 인코더는 인트라 예측자를 생성하는 데 사용되는 기준선을 결정하고 시그널링할 수 있다. 기준선을 지시하는 (기준선 인덱스로도 또한 지칭되는) 인덱스는 인트라 예측 모드(들) 전에 시그널링될 수 있다. 일례로, 0이 아닌 기준선 인덱스가 시그널링될 때 MPM만 허용된다. 도 12는 CB(1210)에 대한 4개의 기준선의 일례를 도시한다. 도 12를 참조하면, 기준선은 최대 6개의 세그먼트, 예컨대 세그먼트 A 내지 세그먼트 F 및 최상부 왼쪽 참조 샘플을 포함할 수 있다. 예를 들어, 기준선 0은 세그먼트 B 및 세그먼트 E와 최상부 왼쪽 참조 샘플을 포함한다. 예를 들어, 기준선 3은 세그먼트 A 내지 세그먼트 F 및 최상부 왼쪽 참조 샘플을 포함한다. 세그먼트 A 및 F는 각각 세그먼트 B 및 E에서 가장 가까운 샘플로 패딩(pad)될 수 있다. 일부 예에서, 이를테면 HEVC에서, 단지 하나의 기준선(예컨대, CB(1210)에 인접한 기준선 0)이 인트라 예측을 위해 사용된다. 일부 예에서, 이를테면 VVC에서, 다수의 기준선(예컨대, 기준선 0, 기준선 1 및 기준선 3)이 인트라 예측을 위해 사용된다.

[0112] 일반적으로, 이를테면 도 9 - 도 12를 참조하여 위에서 설명된 다양한 인트라 예측 모드 중 하나 또는 이들의 적절한 조합을 사용하여 블록이 예측될 수 있다.

[0113] (변환 파티션, 변환 유닛 파티션으로도 또한 지칭되는) 변환 블록 파티션은 블록을 다수의 TU 또는 다수의 TB로 파티셔닝하도록 구현될 수 있다. 도 13 - 도 14는 본 개시내용의 실시예에 따른 예시적인 변환 블록 파티션을 도시한다. AV1과 같은 일부 예에서, 인트라 코딩된 블록 및 인터 코딩된 블록 모두가 다수의 레벨(예컨대, 2개의 레벨)까지의 파티셔닝 깊이를 갖는 다수의 변환 유닛으로 추가 파티셔닝될 수 있다. 변환 블록 파티션으로 인한 다수의 변환 유닛은 TB로 지칭될 수 있다. 다수의 TU 또는 다수의 TB 각각에 대해 변환(예컨대, 아래에서 설명되는 것과 같은 1차 변환 및/또는 2차 변환)이 수행될 수 있다. 이에 따라, 다수의 변환 유닛 또는 다중 TB로 파티셔닝되는 블록에 대해 다수의 변환이 수행될 수 있다.

[0114] 인트라 코딩된 블록의 경우, 인트라 코딩된 블록과 연관된 변환 블록이 동일한 크기를 갖도록 변환 파티션이 수행될 수 있고, 변환 블록이 래스터 스캔 순서로 코딩될 수 있다. 도 13을 참조하면, 블록(예컨대, 인트라 코딩된 블록)(1300)에 대해 변환 블록 파티션이 수행될 수 있다. 블록(1300)은 4개의 변환 유닛(예컨대, TB)(1301-1304)과 같은 변환 유닛으로 분할될 수 있고, 파티셔닝 깊이는 1이다. 4개의 변환 유닛(예컨대, TB)(1301-1304)은 동일한 크기를 가질 수 있고, 변환 유닛(1301)에서 변환 유닛(1304)까지 래스터 스캔 순서(1310)로 코딩될 수 있다. 일례로, 4개의 변환 유닛(예컨대, TB)(1301-1304)은 예를 들어, 상이한 변환 커널을 사용하여 개별적으로 변환된다. 일부 예에서, 4개의 변환 유닛(예컨대, TB들)(1301-1304) 각각은 4개의 변환 유닛으로 더 분할된다. 예를 들어, 변환 유닛(1301)은 변환 유닛(1321, 1322, 1325, 1326)으로 파티셔닝되고, 변환 유닛(1302)은 변환 유닛(1323, 1324, 1327, 1328)으로 파티셔닝되며, 변환 유닛(1303)은 변환 유닛(1329, 1330, 1333, 1334)으로 파티셔닝되고, 변환 유닛(1304)은 변환 유닛(1331, 1332, 1335, 1336)으로 파티셔닝된다. 파티셔닝 깊이는 2이다. 변환 유닛(예컨대, TB)(1321-1336)은 동일한 크기를 가질 수 있고, 변환 유닛(1321)에서 변환 유닛(1336)까지 래스터 스캔 순서(1320)로 코딩될 수 있다.

[0115] 인터 코딩된 블록에 대해, 변환 파티션은 다수의 레벨(예컨대, 2개의 레벨)까지의 파티셔닝 깊이로 재귀적 방식으로 수행될 수 있다. 변환 파티션은 임의의 적절한 변환 유닛 크기 및 형상을 지원할 수 있다. 변환 유닛 형상은 정사각형 형상 및 임의의 적절한 중형비를 갖는 정사각형이 아닌 형상(예컨대, 정사각형이 아닌 직사각형 형상)을 포함할 수 있다. 변환 유닛 크기는 4×4 내지 64×64 의 범위일 수 있다. 변환 유닛의 중형비(예컨대, 변환 유닛의 폭 대 변환 유닛의 높이의 비)는 1:1(정사각형), 1:2, 2:1, 1:4, 4:1 등일 수 있다. 변환 파티션은 4×4 내지 64×64 범위의 1:1(정사각형), 1:2, 2:1, 1:4 및/또는 4:1 변환 유닛 크기를 지원할 수 있다. 도 14를 참조하면, 변환 블록 파티션이 블록(예컨대, 인터 코딩된 블록)(1400)에 대해 재귀적으로 수행될 수 있다. 예를 들어, 블록(1400)은 변환 유닛(1401-1407)으로 파티셔닝된다. 변환 유닛(예컨대, TB)(1401-1407)은 상이한 크기를 가질 수 있고, 변환 유닛(1401)에서 변환 유닛(1407)까지 래스터 스캔 순서(1410)로 코딩될 수 있다. 일례로, 변환 유닛(1401, 1406, 1407)의 파티셔닝 깊이는 1이고, 변환 유닛(1402-1405)의 파티셔닝 깊이는 2이다.

[0116] 일례로, 코딩 블록이 64×64 이하라면, 변환 파티션은 루마 성분에만 적용될 수 있다. 일례로, 코딩 블록은 CTB를 의미한다.

[0117] 코딩 블록 폭(W) 또는 코딩 블록 높이(H)가 64보다 크다면, 코딩 블록은 다수의 TB로 암시적으로 분할될 수 있으며, 여기서 코딩 블록은 루마 코딩 블록이다. 다수의 TB 중 하나의 TB의 폭은 W와 64의 최소값일 수 있고, 다수의 TB 중 하나의 TB의 높이는 H와 64의 최소값일 수 있다.

[0118] 코딩 블록 폭(W) 또는 코딩 블록 높이(H)가 64보다 크다면, 코딩 블록은 다수의 TB로 암시적으로 분할될 수 있

으며, 여기서 코딩 블록은 크로마 코딩 블록이다. 다수의 TB 중 하나의 TB의 폭은 W와 32의 최소값일 수 있고, 다수의 TB 중 하나의 TB의 높이는 H와 32의 최소값일 수 있다.

- [0119] AOMedia Video 1(AV1)에서 사용되는 것과 같은 1차 변환의 실시예가 아래에서 설명된다. 주파수 도메인(또는 공간 주파수 도메인)의 변환 계수를 포함하는 변환 블록(TB)을 획득하기 위해 잔차(예컨대, 공간 도메인의 잔차)를 포함하는 TB에 대해 (예컨대, 인코더에서의) 순방향 변환이 수행될 수 있다. 공간 도메인의 잔차를 포함하는 TB는 잔차 TB로 지칭될 수 있으며, 주파수 도메인의 변환 계수를 포함하는 TB는 계수 TB로 지칭될 수 있다. 일례로, 순방향 변환은 잔차 TB를 계수 TB로 변환할 수 있는 순방향 1차 변환을 포함한다. 일례로, 순방향 변환은 순방향 1차 변환 및 순방향 2차 변환을 포함하며, 여기서 순방향 1차 변환은 잔차 TB를 중간 계수 TB로 변환할 수 있고, 순방향 2차 변환은 중간 계수 TB를 계수 TB로 변환할 수 있다.
- [0120] 공간 도메인에서 잔차 TB를 획득하기 위해 주파수 도메인에서 계수 TB에 대해 (예컨대, 인코더에서의 또는 디코더에서의) 역변환이 수행될 수 있다. 일례로, 역변환은 계수 TB를 잔차 TB로 변환할 수 있는 역방향 1차 변환을 포함한다. 일례로, 역변환은 역방향 1차 변환 및 역방향 2차 변환을 포함하며, 여기서 역방향 2차 변환은 계수 TB를 중간 계수 TB로 변환할 수 있고, 역방향 1차 변환은 중간 계수 TB를 잔차 TB로 변환할 수 있다.
- [0121] 일반적으로, 1차 변환은 순방향 1차 변환 또는 역방향 1차 변환을 의미할 수 있고, 여기서 잔차 TB와 계수 TB 사이에서 1차 변환이 수행된다. 일부 실시예에서, 1차 변환은 분리 가능 변환일 수 있으며, 여기서 2D 1차 변환은 (수평 변환으로도 또한 지칭되는) 수평 1차 변환 및 (수직 변환으로도 또한 지칭되는) 수직 1차 변환을 포함할 수 있다. 2차 변환은 순방향 2차 변환 또는 역방향 2차 변환을 의미할 수 있으며, 여기서 중간 계수 TB와 계수 TB 사이에서 2차 변환이 수행된다.
- [0122] 이를테면, 본 개시내용에서 설명되는 확장된 코딩 블록 파티션을 지원하기 위해, (예컨대, 각각의 차원에 대해 4-점 내지 64-점의 범위에 이르는) 다수의 변환 크기 및 변환 형상(예컨대, 정사각형, 폭 대 높이 비가 2:1, 1:2, 4:1 또는 1:4인 직사각형 형상)이 이를테면, AV1에 사용될 수 있다.
- [0123] 2D 변환 프로세스는 코딩된 잔차 블록의 각각의 차원마다 상이한 1D 변환을 포함할 수 있는 하이브리드 변환 커널을 사용할 수 있다. 1차 1D 변환은 (a) 4-점, 8-점, 16-점, 32-점, 64-점 DCT-2; (b) 4-점, 8-점, 16-점 비대칭 DST(ADST)(예컨대, DST-4, DST-7) 및 대응하는 플립 버전(예컨대, ADST의 플립 버전 또는 FlipADST는 ADST를 역순으로 적용할 수 있음); 및/또는 (c) 4-점, 8-점, 16-점, 32-점 항등 변환(IDTX)을 포함할 수 있다. 도 15는 본 개시내용의 실시예에 따른 1차 변환 기저 함수의 예를 도시한다. 도 15의 예에서 1차 변환 기본 함수는 N-점 입력을 갖는 DCT-2 및 비대칭 DST(DST-4 및 DST-7)에 대한 기저 함수를 포함한다. AV1에서는 도 15에 도시된 1차 변환 기저 함수가 사용될 수 있다.
- [0124] 하이브리드 변환 커널의 이용 가능성은 변환 블록 크기 및 예측 모드에 의존할 수 있다. 도 16a는 변환 블록 크기(예컨대, 세 번째 열에 도시된 크기) 및 예측 모드(예컨대, 세 번째 열에 도시된 인트라 예측 및 인터 예측)에 기반한 다양한 변환 커널(예컨대, 첫 번째 열에 도시되며 두 번째 열에서 설명되는 변환 타입)의 이용 가능성의 예시적인 의존성을 도시한다. 예시적인 하이브리드 변환 커널, 및 예측 모드와 변환 블록 크기에 기반한 이용 가능성이 AV1에서 사용될 수 있다. 도 16a를 참조하면, "→" 및 "↓" 심벌은 각각 (수평 방향으로도 또한 지칭되는) 수평 치수 및 (수직 방향으로도 또한 지칭되는) 수직 치수를 나타낸다. "√" 및 "x" 심벌은 해당 블록 크기 및 예측 모드에 대한 변환 커널의 이용 가능성을 나타낸다. 예를 들어, "√" 심벌은 변환 커널이 이용 가능함을 나타내고, "x" 심벌은 변환 커널이 이용 가능하지 않음을 나타낸다.
- [0125] 일례로, 변환 타입(1610)은 도 16a의 첫 번째 열에 도시된 바와 같이 ADST_DCT로 표시된다. 변환 타입(1610)은 도 16a의 두 번째 열에 도시된 바와 같이 수직 방향의 ADST 및 수평 방향의 DCT를 포함한다. 도 16a의 세 번째 열에 따르면, 블록 크기가 16×16(예컨대, 16×16개의 샘플, 16×16개의 루마 샘플) 이하인 경우 인트라 예측 및 인터 예측에 변환 타입(1610)이 이용 가능하다.
- [0126] 일례로, 변환 타입(1620)은 도 16a의 첫 번째 열에 도시된 바와 같이 V_ADST로 표시된다. 변환 타입(1620)은 도 16a의 두 번째 열에 도시된 바와 같이 수직 방향의 ADST 및 수평 방향의 IDTX(즉, 단위 행렬)를 포함한다. 따라서 변환 타입(1620)(예컨대, V_ADST)은 수직 방향으로 수행되고 수평 방향으로는 수행되지 않는다. 도 16a의 세 번째 열에 따르면, 블록 크기에 관계없이 인트라 예측을 위해 변환 타입(1620)이 이용 가능하지 않다. 블록 크기가 16×16(예컨대, 16×16개의 샘플, 16×16개의 루마 샘플) 미만인 경우 인터 예측에 변환 타입(1620)이 이용 가능하다.
- [0127] 일례로, 도 16a는 루마 성분에 적용 가능하다. 크로마 성분의 경우, 변환 타입(또는 변환 커널) 선택이 암시적

으로 수행될 수 있다. 일례로, 인트라 예측 잔차의 경우, 도 16b에 도시된 바와 같이, 인트라 예측 모드에 따라 변환 타입이 선택될 수 있다. 일례로, 도 16b에 도시된 변환 타입 선택은 크로마 성분에 적용 가능하다. 인트라 예측 잔차의 경우, 콜로케이트(co-locate)된 루마 블록의 변환 타입 선택에 따라 변환 타입이 선택될 수 있다. 따라서 일례로, 크로마 성분에 대한 변환 타입은 비트스트림에서 시그널링되지 않는다.

[0128] 1차 변환과 같은 변환에, 예를 들어 AOMedia Video 2(AV2)에 선 그래프 변환(LGT)이 사용될 수 있다. 8-비트/10-비트 변환 코어가 AV2에 사용될 수 있다. 일례로, LGT는 아래에서 설명되는 바와 같이, 다양한 DCT, 이산 사인 변환(DST: discrete sine transform)을 포함한다. LGT는 32-점 및 64-점 1차원(1D: 1-dimensional) DST를 포함할 수 있다.

[0129] 그래프는 관심 대상 간의 유사함(affinity) 관계를 모델링하는 데 사용될 수 있는 정점 및 에지의 세트를 포함하는 일반적인 수학적 구조이다. 한 세트의 가중치가 에지에 그리고 선택적으로 정점에 할당되는 가중된 그래프는 신호/데이터의 강력한 모델링에 대한 희소(sparse) 표현을 제공할 수 있다. LGT는 다양한 블록 통계에 대한 더 나은 적응을 제공함으로써 코딩 효율을 향상시킬 수 있다. 데이터로부터 선 그래프를 학습하여 블록의 잔차 신호의 기본 행 및 열 단위 통계를 모델링함으로써 분리 가능한 LGT가 설계 및 최적화될 수 있으며, 연관된 일반화된 그래프 라플라시안(GGL) 행렬이 LGT를 도출하는 데 사용될 수 있다.

[0130] 도 16c는 본 개시내용의 일 실시예에 따른 자가 루프 가중치(예컨대, \mathbf{v}_{c1} , \mathbf{v}_{c2}) 및 에지 가중치(\mathbf{w}_c)를 특징으로 하는 일반 LGT의 일례를 도시한다. 가중된 그래프 $\mathbf{G}(\mathbf{W}, \mathbf{V})$ 가 주어지면, GGL 행렬은 아래와 같이 정의될 수 있다.

[0131]
$$\mathbf{L}_c = \mathbf{D} - \mathbf{W} + \mathbf{V} \text{ (식 3)}$$

[0132] 여기서 \mathbf{W} 는 음이 아닌 에지 가중치들(\mathbf{w}_c)을 포함하는 인접 행렬일 수 있고, \mathbf{D} 는 대각 차수 행렬일 수 있으며, \mathbf{V} 는 자가 루프 가중치(\mathbf{v}_{c1} , \mathbf{v}_{c2})를 나타내는 대각 행렬일 수 있다. 도 16d는 행렬(\mathbf{L}_c)의 일례를 도시한다.

[0133] LGT는 아래와 같이 GGL 행렬(\mathbf{L}_c)의 고유 분해에 의해 도출될 수 있다.

[0134]
$$\mathbf{L}_c = \mathbf{U} \Phi \mathbf{U}^T \text{ (식 4)}$$

[0135] 여기서 직교 행렬(\mathbf{U})의 열은 LGT의 기저 벡터가 될 수 있고, Φ 는 대각 고유값 행렬이 될 수 있다.

[0136] 다양한 예들에서, 특정 DCT 및 DST(예컨대, DCT-2, DCT-8, DST-7)는 GGL의 특정 형태로부터 도출된 한 세트의 LGT의 서브세트이다. DCT-2는 \mathbf{v}_{c1} 을 0으로 설정(예컨대, $\mathbf{v}_{c1} = 0$)함으로써 도출될 수 있다. DST-7은 \mathbf{v}_{c1} 을 \mathbf{w}_c 로 설정(예컨대, $\mathbf{v}_{c1} = \mathbf{w}_c$)함으로써 도출될 수 있다. DCT-8은 \mathbf{v}_{c2} 를 \mathbf{w}_c 로 설정(예컨대, $\mathbf{v}_{c2} = \mathbf{w}_c$)함으로써 도출될 수 있다. DST-4는 \mathbf{v}_{c1} 을 $2\mathbf{w}_c$ 로 설정(예컨대, $\mathbf{v}_{c1} = 2\mathbf{w}_c$)함으로써 도출될 수 있다. DCT-4는 \mathbf{v}_{c2} 을 $2\mathbf{w}_c$ 로 설정(예컨대, $\mathbf{v}_{c2} = 2\mathbf{w}_c$)함으로써 도출될 수 있다.

[0137] 일부 예에서, AV2에서, LGT는 행렬 곱셈으로서 구현될 수 있다. 4-점(4p) LGT 코어는 \mathbf{L}_c 에서 \mathbf{v}_{c1} 을 $2\mathbf{w}_c$ 로 설정함으로써 도출될 수 있고, 따라서 4p LGT 코어는 DST-4이다. 8-점(8p) LGT 코어는 \mathbf{L}_c 에서 \mathbf{v}_{c1} 을 $1.5\mathbf{w}_c$ 로 설정함으로써 도출될 수 있다. 일례로, 16-점(16p) LGT 코어, 32-점(32p) LGT 코어 또는 64-점(64p) LGT 코어와 같은 LGT 코어는 \mathbf{v}_{c1} 을 \mathbf{w}_c 로 그리고 \mathbf{v}_{c2} 를 0으로 설정함으로써 도출되며, LGT 코어는 DST-7이 될 수 있다.

[0138] 1차 변환, 2차 변환과 같은 변환은 CB와 같은 블록에 적용될 수 있다. 일례로, 변환은 1차 변환, 2차 변환의 조합을 포함한다. 변환은 비분리 변환, 분리 가능 변환, 또는 비분리 변환과 분리 가능 변환의 조합일 수 있다.

[0139] 이를테면, VVC에서 2차 변환이 수행될 수 있다. 일부 예에서, 이를테면 VVC에서, 도 17 - 도 18에 도시된 바와 같이, 축소 2차 변환(RST: reduced secondary transform)으로도 또한 알려진 저주파 비분리 변환(LFNST: low-frequency non-separable transform)이 인코더 측에서의 순방향 1차 변환과 양자화 사이에 그리고 디코더 측에서의 역양자화와 역방향 1차 변환 사이에 적용되어 1차 변환 계수를 추가로 역상관할 수 있다.

[0140] LFNST에서 사용될 수 있는 비분리 변환의 적용은 (식 5에 도시된) 4×4 입력 블록(또는 입력 행렬)(X)을 예로서 사용하여 다음과 같이 설명될 수 있다. 4×4 비분리 변환(예컨대, LFNST)을 적용하기 위해, 4×4 입력 블록

(X)은 식 5 - 식 6에 도시된 바와 같이, 벡터 \vec{X} 로 표현될 수 있다.

$$X = \begin{bmatrix} X_{00} & X_{01} & X_{02} & X_{03} \\ X_{10} & X_{11} & X_{12} & X_{13} \\ X_{20} & X_{21} & X_{22} & X_{23} \\ X_{30} & X_{31} & X_{32} & X_{33} \end{bmatrix}$$

[0141]

(식 5)

$$\vec{X} = [X_{00} \ X_{01} \ X_{02} \ X_{03} \ X_{10} \ X_{11} \ X_{12} \ X_{13} \ X_{20} \ X_{21} \ X_{22} \ X_{23} \ X_{30} \ X_{31} \ X_{32} \ X_{33}]^T$$
 (식 6)

[0142]

비분리 변환은 $\vec{F} = T \cdot \vec{X}$ 로서 계산될 수 있으며, 여기서 \vec{F} 는 변환 계수 벡터를 지시하고, T는 16×16 변환 행렬이다. 16×1 계수 벡터(\vec{F})는 후속하여 4×4 입력 블록에 대한 스캐닝 순서(예컨대, 수평 스캐닝 순서, 수직 스캐닝 순서, 지그재그 스캐닝 순서 또는 대각 스캐닝 순서)를 사용하여 4×4 출력 블록(또는 출력 행렬, 계수 블록)으로 재조직될 수 있다. 더 작은 인덱스를 가진 변환 계수는 4×4 계수 블록에서 더 작은 스캐닝 인덱스로 배치될 수 있다.

[0143]

비분리 2차 변환은 블록(예컨대, CB)에 적용될 수 있다. 일부 예에서, 이를테면 VVC에서, 도 17 - 도 18에 도시된 바와 같이, LFNST가 (예컨대, 인코더 측에서의) 순방향 1차 변환과 양자화 사이에 그리고 (예컨대, 디코더 측에서의) 역양자화와 역방향 1차 변환 사이에 적용된다.

[0144]

도 17 - 도 18은 각각 16×64 변환(또는 변환이 순방향 2차 변환인지 또는 역방향 2차 변환인지에 따라 64×16 변환) 및 16×48 변환(또는 변환이 순방향 2차 변환인지 또는 역방향 2차 변환인지에 따라 48×16 변환)을 사용하는 2개의 변환 코딩 프로세스(1700, 1800)의 예를 도시한다. 도 17을 참조하면, 프로세스(1700)에서, 인코더 측에서는, 블록(예컨대, 잔차 블록)에 대해 먼저 순방향 1차 변환(1710)이 수행되어 계수 블록(1713)을 획득할 수 있다. 이어서, 순방향 2차 변환(또는 순방향 LFNST)(1712)이 계수 블록(1713)에 적용될 수 있다. 순방향 2차 변환(1712)에서, 계수 블록(1713)의 최상부 왼쪽 모서리에 있는 4×4 서브블록(A-D)의 64개의 계수는 64-길이 벡터로 표현될 수 있고, 64-길이 벡터는 64×16의 변환 행렬(즉, 64의 폭과 16의 높이)과 곱해져 16-길이 벡터가 될 수 있다. 16-길이 벡터의 요소는 계수 블록(1713)의 최상부 왼쪽 4×4 서브블록(A)에 다시 채워진다. 서브블록(A-D)의 계수는 0일 수 있다. 순방향 2차 변환(1712) 후의 결과적인 계수는 다음에, 양자화 단계(1714)에서 양자화되고 엔트로피 코딩되어 비트스트림(1716)에서 코딩된 비트를 생성한다.

[0145]

코딩된 비트는 디코더 측에서 수신될 수 있고, 엔트로피 디코딩된 후 역양자화 단계(1724)가 이어져 계수 블록(1723)을 생성할 수 있다. 역 RST8×8과 같은 역방향 2차 변환(또는 역방향 LFNST)(1722)이 수행되어, 예를 들어 최상부 왼쪽 4×4 서브블록(E)의 16개의 계수로부터 64개의 계수를 획득할 수 있다. 64개의 계수는 4×4 서브블록(E-H)으로 다시 채워질 수 있다. 또한, 역방향 2차 변환(1722) 이후의 계수 블록(1723)의 계수는 역방향 1차 변환(1720)으로 처리되어 복원된 잔차 블록을 얻을 수 있다.

[0146]

도 18의 예의 프로세스(1800)는 더 적은 수(즉, 48개)의 계수가 순방향 2차 변환(1712) 동안 처리된다는 점을 제외하고는 프로세스(1700)와 유사하다. 구체적으로, 서브블록(A-C)의 48개의 계수가 48×16 크기의 더 작은 변환 행렬로 처리된다. 48×16의 더 작은 변환 행렬을 사용하는 것은 변환 행렬을 저장하기 위한 메모리 크기 및 계산(예컨대, 곱셈, 덧셈, 뺄셈 등)의 수를 감소시킬 수 있고, 따라서 계산 복잡성을 줄일 수 있다.

[0147]

일례로, 블록(예컨대, CB)의 크기에 따라 4×4 비분리 변환(예컨대, 4×4 LFNST) 또는 8×8 비분리 변환(예컨대, 8×8 LFNST)이 적용된다. 블록의 블록 크기는 폭, 높이 등을 포함할 수 있다. 예를 들어, 4×4 LFNST가 블록에 적용되며, 여기서 폭과 높이의 최소값은 임계치 미만, 이를테면 8(예컨대, min(폭, 높이) < 8)이다. 예를 들어, 8×8 LFNST가 블록에 적용되며, 여기서 폭과 높이의 최소값은 임계치보다 큰데, 이를테면 4(예컨대, min(폭, 높이) > 4)이다.

[0148]

직접 행렬 곱셈 접근 방식을 기반으로 할 수 있으며, 따라서 반복 없이 단일 패스(pass)로 구현될 수 있다. 비분리 변환 행렬 차원을 줄이기 위해 그리고 변환 계수를 저장할 메모리 공간 및 계산 복잡성을 최소화하기 위해, LFNST에서 축소 비분리 변환 방법(또는 RST)이 사용될 수 있다. 이에 따라, 축소 비분리 변환에서는, N(예컨대, 8×8 비분리 2차 변환(NSST: non-separable secondary transform)의 경우 N은 64임)차 벡터가 다른 공간의 R차 벡터에 매핑될 수 있으며, 여기서 N/R(R < N)은 축소 계수이다. 그러므로 RST 행렬은 N×N 행렬 대신에 식 7에서 설명되는 바와 같이 R×N 행렬이 된다.

[0149]

$$T_{R \times N} = \begin{bmatrix} t_{11} & t_{12} & t_{13} & \dots & t_{1N} \\ t_{21} & t_{22} & t_{23} & \dots & t_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{R1} & t_{R2} & t_{R3} & \dots & t_{RN} \end{bmatrix} \quad (\text{식 7})$$

[0150]

[0151]

식 7에서, R×N 변환 행렬의 R개의 행은 N차원 공간의 R개의 베이스이다. 역변환 행렬은 순방향 변환에 사용되는 변환 행렬의 전치(transpose)(예컨대, T_{R×N})일 수 있다. 8×8 LFNST의 경우, 4의 축소 계수가 적용될 수 있으며, 8×8 비분리 변환에 사용되는 64×64 직접 행렬은 도 17에 도시된 바와 같이, 16×64 직접 행렬로 축소될 수 있다. 대안으로, 4보다 큰 축소 계수가 적용될 수 있으며, 8×8 비분리 변환에 사용되는 64×64 직접 행렬은 도 18에 도시된 바와 같이, 16×64 직접 행렬로 축소될 수 있다. 그러므로 48×16 RST 역행렬이 디코더 측에서 8×8 최상부 왼쪽 영역에 코어(1차) 변환 계수를 생성하는 데 사용될 수 있다.

[0152]

도 18을 참조하면, 동일한 변환 세트 구성을 갖는 16×64 행렬 대신 16×48 행렬이 적용되는 경우, 16×48 행렬에 대한 입력은 오른쪽 아래 4×4 블록(D)을 제외한 최상부 왼쪽 8x8 블록에서 3개의 4×4 블록(A, B, C)으로부터의 48개의 입력 데이터를 포함한다. 차원의 축소에 따라, LFNST 행렬을 저장하기 위한 메모리 사용량이 예를 들어, 최소한의 성능 저하와 함께 10KB에서 8KB로 축소될 수 있다.

[0153]

복잡성을 줄이기 위해, LFNST는 제1 계수 서브그룹 외부의 계수가 비유효(non-significant)한 경우에 적용 가능하도록 제한될 수 있다. 일례로, LFNST는 제 서브그룹 외부의 모든 계수가 비유효한 경우에만 적용 가능하도록 제한될 수 있다. 도 17 - 도 18을 참조하면, 제1 계수 서브그룹은 최상부 왼쪽 블록(E)에 대응하고, 따라서 블록(E) 외부에 있는 계수는 비유효하다.

[0154]

일례로, LFNST가 적용될 때 1차 전용 변환 계수는 비유효하다(예컨대, 0). 일례로, LFNST가 적용될 때 모든 1차 전용 변환 계수는 0이다. 1차 전용 변환 계수는 2차 변환 없이 1차 변환에서 구해지는 변환 계수를 참조할 수 있다. 이에 따라, LFNST 인덱스 시그널링은 마지막 유효 포지션에 대해 조정될 수 있고, 따라서 LFNST에서 추가 계수 스케닝을 피할 수 있다. 일부 예에서, 추가 계수 스케닝은 특정 포지션에서 중요한 변환 계수를 확인하는 데 사용된다. 일례로, 예를 들어, 픽셀당 곱셈의 관점에서 LFNST의 최악의 경우의 처리는 4×4 블록 및 8×8 블록에 대한 비분리 변환을 각각 8×16 변환 및 8×48 변환으로 제한한다. 위의 경우, LFNST가 적용될 때 마지막 유효 스캔 포지션은 8 미만일 수 있다. 다른 크기의 경우, LFNST가 적용될 때 마지막 유효 스캔 포지션은 16 미만일 수 있다. 4×N 및 Nx4의 CB 그리고 8보다 큰 N에 대해, 제약은 LFNST가 CB의 최상부 왼쪽 4×4 영역에 적용됨을 의미할 수 있다. 일례로, 제약은 LFNST가 CB의 최상부 왼쪽 4×4 영역에만 한 번만 적용됨을 의미한다. 일례로, LFNST가 적용될 때 모든 1차 전용 계수는 비유효하고(예컨대, 0), 1차 변환을 위한 연산의 수가 감소된다. 인코더 관점에서, 변환 계수의 양자화는 LFNST 변환이 테스트될 때 상당히 단순화될 수 있다. 레이트 왜곡 최적화 양자화는 처음 16개의 계수에 대해 최대로 수행될 수 있으며, 예를 들어 스캔 순서에서 나머지 계수는 0으로 설정될 수 있다.

[0155]

LFNST 변환(예컨대, 변환 커널, 변환 코어 또는 변환 행렬)은 아래에서 설명되는 바와 같이 선택될 수 있다. 일 실시예에서, 다수의 변환 세트가 사용될 수 있고, 하나 이상의 비분리 변환 행렬(또는 커널)이 LFNST의 다수의 변환 세트 각각에 포함될 수 있다. 본 개시내용의 양상에 따르면, 다수의 변환 세트로부터 변환 세트가 선택될 수 있고, 변환 세트 내의 하나 이상의 비분리 변환 행렬로부터 비분리 변환 행렬이 선택될 수 있다.

[0156]

표 1은 본 개시내용의 일 실시예에 따른 인트라 예측 모드로부터 다수의 변환 세트로의 예시적인 매핑을 도시한다. 매핑은 인트라 예측 모드와 다수의 변환 세트 간의 관계를 지시한다. 이를테면, 표 1에 지시된 관계는 미리 정의되어 인코더 및 디코더에 저장될 수 있다.

[0157]

표 1: 변환 세트 선택 표

표 1

[0158]

IntraPredMode	Tr. set index
IntraPredMode < 0	1
0 <= IntraPredMode <= 1	0
2 <= IntraPredMode <= 12	1
13 <= IntraPredMode <= 23	2
24 <= IntraPredMode <= 44	3
45 <= IntraPredMode <= 55	2
56 <= IntraPredMode <= 80	1

81 <= IntraPredMode <= 83	0
---------------------------	---

- [0159] 표 1을 참조하면, 다수의 변환 세트는 4개의 변환 세트, 예컨대 각각 0 내지 3의 변환 세트 인덱스(예컨대, Tr. set index)로 표현되는 변환 세트 0 내지 3을 포함한다. 인덱스(예컨대, IntraPredMode)는 인트라 예측 모드를 지시할 수 있고, 인덱스 및 표 1에 기초하여 변환 세트 인덱스가 획득될 수 있다. 이에 따라, 변환 세트는 인트라 예측 모드에 기초하여 결정될 수 있다. 일례로, 3개의 교차 성분 선형 모델(CCLM: cross component linear model) 모드(예컨대, INTRA_LT_CCLM, INTRA_T_CCLM 또는 INTRA_L_CCLM) 중 하나가 CB에 사용된다면(예컨대, 81 <= IntraPredMode <= 83), 변환 세트 0이 CB에 대해 선택된다.
- [0160] 위에서 설명된 바와 같이, 각각의 변환 세트는 하나 이상의 비분리 변환 행렬을 포함할 수 있다. 하나 이상의 비분리 변환 행렬 중 하나는 예를 들어, 명시적으로 시그널링되는 LFNST 인덱스에 의해 선택될 수 있다. LFNST 인덱스는 예를 들어, 변환 계수를 시그널링한 후에 인트라 코딩된 CU(예컨대, CB)당 한 번 비트스트림에서 시그널링될 수 있다. 일 실시예에서, 각각의 변환 세트는 2개의 비분리 변환 행렬(커널)을 포함하고, 선택된 비분리 2차 변환 후보는 2개의 비분리 변환 행렬 중 하나일 수 있다. 일부 예에서, LFNST는 CB에 적용되지 않는다(예컨대, 변환 스킵 모드로 코딩된 CB 또는 CB의 0이 아닌 계수의 수가 임계치 미만이다). 일례로, LFNST가 CB에 적용되지 않을 때 LFNST 인덱스는 CB에 대해 시그널링되지 않는다. LFNST 인덱스에 대한 디폴트 값은 0일 수 있으며 시그널링되지 않아, LFNST가 CB에 적용되지 않음을 지시할 수 있다.
- [0161] 일 실시예에서, LFNST는 제1 계수 서브그룹 외부의 모든 계수가 비유효한 경우에만 적용 가능하도록 제한되고, LFNST 인덱스의 코딩은 마지막 유효 계수의 포지션에 의존할 수 있다. LFNST 인덱스는 컨텍스트 코딩될 수 있다. 일례로, LFNST 인덱스의 컨텍스트 코딩은 인트라 예측 모드에 의존하지 않고, 첫 번째 빈(bin)만이 컨텍스트 코딩된다. LFNST는 인트라 슬라이스에서 또는 인터 슬라이스에서 인트라 코딩된 CU에 그리고 루마 성분과 크로마 성분 모두에 대해 적용될 수 있다. 이중 트리가 인에이블된다면, 루마 성분 및 크로마 성분에 대한 LFNST 인덱스가 개별적으로 시그널링될 수 있다. 인터 슬라이스(예컨대, 이중 트리가 디세이بل됨)의 경우, 단일 LFNST 인덱스가 시그널링되고 루마 성분 및 크로마 성분 모두에 사용될 수 있다.
- [0162] 인트라 서브 파티션(ISP: Intra sub-partition) 코딩 모드가 사용될 수 있다. ISP 코딩 모드에서는, 루마 인트라 예측 블록이 블록 크기에 따라 수직 또는 수평으로 2개 또는 4개의 서브 파티션으로 분할될 수 있다. 일부 예에서, RST가 모든 실행 가능한 서브 파티션에 적용될 때는 성능 향상이 미미하다. 따라서 일부 예에서, ISP 모드가 선택될 때, LFNST가 디세이بل되고 LFNST 인덱스(또는 RST 인덱스)는 시그널링되지 않는다. ISP 예측 잔차에 대해 RST 또는 LFNST를 디세이블하는 것은 코딩 복잡성을 줄일 수 있다. 일부 예에서, 행렬 기반 인트라 예측 모드(MIP: matrix-based intra predication mode)가 선택되면, LFNST가 디세이بل되고 LFNST 인덱스는 시그널링되지 않는다.
- [0163] 어떤 예에서, 64×64보다 큰 CU는 최대 변환 크기 제약(예컨대, 64×64)으로 인해 암시적으로 분할(TU 타일링)되고, LFNST 인덱스 검색이 특정 수의 디코드 파이프라인 스테이지 동안 데이터 버퍼링을 4배 증가시킬 수 있다. 따라서 LFNST가 허용되는 최대 크기는 64×64로 제한될 수 있다. 예에서 LFNST는 이산 코사인 변환(DCT) 타입 2(DCT-2) 변환으로만 인에이블된다.
- [0164] 일부 예에서, 분리 가능한 변환 방식은 방향성 텍스처 패턴(예컨대, 45° 또는 135° 방향을 따르는 에지)을 캡처하는 데 효율적이지 않을 수 있다. 비분리 변환 방식은 예를 들어, 위의 시나리오에서 코딩 효율을 향상시킬 수 있다. 계산 복잡성과 메모리 사용량을 줄이기 위해, 1차 변환에서 얻어지는 저주파 변환 계수에 적용되는 2차 변환으로서 비분리 변환 방식이 사용될 수 있다. 2차 변환은 블록에 적용될 수 있으며, 2차 변환을 지시하는 정보, 예를 들어 인덱스(예컨대, 2차 변환 인덱스)가 예측 모드 정보, 1차 변환 타입, 주변 복원 샘플, 변환 블록 파티셔닝 정보, 블록의 크기 및 형상 등에 기반하여 블록에 대해 시그널링될 수 있다. 예측 모드 정보, 1차 변환 타입, 및 블록의 크기(또는 블록 크기) 중 하나 또는 조합이 컨텍스트 정보를 도출하는 데 이용될 수 있다. 컨텍스트 정보는 한 세트의 2차 변환(예컨대, 한 세트의 비분리 2차 변환) 중에서 2차 변환을 식별하는 데 사용되는 2차 변환 인덱스를 엔트로피 코딩하기 위해 사용될 컨텍스트(들)를 결정하는 데 사용될 수 있다. 2차 변환 인덱스는 블록을 디코딩하는 데 사용될 수 있다.
- [0165] 본 개시내용에서, 블록이라는 용어는 PB, CB, 코딩된 블록, 코딩 유닛(CU), 변환 블록(TB), 변환 유닛(TU), 루마 블록(예컨대, 루마 CB), 크로마 블록(예컨대, 크로마 CB) 등을 의미할 수 있다.
- [0166] 블록의 크기는 블록 폭, 블록 높이, 블록 중형비(예컨대, 블록 폭 대 블록 높이의 비, 블록 높이 대 블록 폭의

비), 블록 영역 크기 또는 블록 면적(예컨대, 블록 폭 x 블록 높이), 블록 폭과 블록 높이의 최소값, 블록 폭과 블록 높이의 최대값 등을 의미할 수 있다.

- [0167] 블록을 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위해 사용되는 2차 변환(예컨대, 2차 변환 커널, 2차 변환 코어 또는 2차 변환 행렬)을 지시하는 정보는 인덱스(예컨대, 2차 변환 인덱스)를 포함할 수 있다. 위에서 논의된 바와 같이, 일 실시예에서, 다수의 변환 세트가 사용될 수 있고, 하나 이상의 2차 변환 행렬(또는 커널)이 다수의 변환 세트 각각에 포함될 수 있다. 본 개시내용의 양상에 따르면, 표 1을 참조하여 설명된 것을 포함하지만 그에 제한되지는 않는 임의의 적합한 방법을 사용하여 다수의 변환 세트들 중에서 변환 세트가 선택될 수 있고, 블록을 코딩(예컨대, 인코딩 및/또는 디코딩)하는 데 사용되는 2차 변환(예컨대, 2차 변환 행렬)이 인덱스(예컨대, 2차 변환 인덱스)에 의해 변환 세트의 하나 이상의 2차 변환 행렬로부터 결정(예컨대, 선택)될 수 있다. 인덱스(예컨대, 2차 변환 행렬)는 블록을 디코딩하는 데 사용되는 2차 변환의 변환 세트 중 하나의 2차 변환을 식별하는 데 사용될 수 있다. 일례로, 2차 변환의 변환 세트는 블록을 디코딩하는 데 사용되는 한 세트의 비분리 2차 변환을 포함한다. 일례로, 인덱스(예컨대, 2차 변환 행렬)는 stIdx로 표기된다. 인덱스(예컨대, 2차 변환 인덱스)는 예를 들어, 코딩된 비디오 비트스트림에서 명시적으로 시그널링될 수 있다.
- [0168] 일례로, 2차 변환은 LFNST이고, 2차 변환 인덱스는 LFNST 인덱스를 의미한다. 일부 예에서, 2차 변환은 블록에 적용되지 않는다(예컨대, 변환 스킵 모드로 코딩된 CB 또는 CB의 0이 아닌 계수의 수가 임계치 미만이다). 일례로, 2차 변환이 블록에 적용되지 않을 때 2차 변환 인덱스(예컨대, LFNST 인덱스)는 블록에 대해 시그널링되지 않는다. 2차 변환 인덱스에 대한 디폴트 값은 0일 수 있으며 시그널링되지 않아, 2차 변환이 블록에 적용되지 않음을 지시할 수 있다.
- [0169] 본 개시내용의 양상에 따르면, 코딩된 비디오 비트스트림으로부터 블록(예컨대, TB)에 대한 코딩된 정보가 디코딩될 수 있다. 코딩된 정보는 블록에 대한 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나 이상을 지시할 수 있다. 1차 변환 타입은 도 15 및 도 16a - 도 16d를 참조하여 설명된 것 중 하나 또는 이들의 조합과 같은 임의의 적합한 1차 변환을 의미할 수 있다.
- [0170] 일례로, 블록은 인트라 코딩 또는 인트라 예측되고, 예측 모드 정보는 인트라 예측 모드 정보로 지칭된다. 블록에 대한 예측 모드 정보(예컨대, 인트라 예측 모드 정보)는 블록에 사용되는 인트라 예측 모드를 지시할 수 있다. 인트라 예측 모드는 블록의 인트라 예측에 사용되는 예측 모드, 이를테면 도 9에서 설명된 방향성 모드(또는 방향성 예측 모드), 도 10에서 설명된 무지향성 예측 모드(예컨대, DC 모드, PAETH 모드, SMOOTH 모드, SMOOTH_V 모드 또는 SMOOTH_H 모드), 또는 도 11에서 설명된 재귀적 필터링 모드를 의미할 수 있다. 인트라 예측 모드는 또한 본 개시내용에서 설명되는 예측 모드, 본 개시내용에서 설명되는 예측 모드의 적절한 변형, 또는 본 개시내용에서 설명되는 예측 모드의 적절한 조합을 참조할 수 있다. 예를 들어, 인트라 예측 모드는 도 12에서 설명된 다중 라인 인트라 예측과 조합될 수 있다.
- [0171] 블록에 대한 예측 모드 정보, 블록의 크기 및 블록에 사용된 1차 변환 타입 중 하나 이상과 같은 코딩된 정보는 2차 변환 인덱스(예컨대, stIdx)를 엔트로피 코딩(예컨대, 엔트로피 인코딩 및/또는 엔트로피 디코딩)하기 위한 컨텍스트로서 사용될 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 엔트로피 인코딩 및/또는 엔트로피 디코딩)하기 위한 컨텍스트는 블록에 대한 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나와 같은 코딩된 정보에 기초하여 결정될 수 있다. 2차 변환 인덱스(예컨대, stIdx)는 한 세트의 2차 변환에서 블록에 대해 수행될 2차 변환을 지시할 수 있다.
- [0172] 컨텍스트 도출 프로세스가 컨텍스트를 결정하는 데 사용될 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트 도출 프로세스는 코딩된 정보에 의존할 수 있다.
- [0173] 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 블록에 대한 예측 모드 정보(예컨대, 인트라 예측 모드 정보), 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나 이상에 기초하여 결정될 수 있다. 일례로, 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 블록에 대한 예측 모드 정보(예컨대, 인트라 예측 모드 정보), 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나에 기초하여 결정된다. 2차 변환 인덱스는 컨텍스트에 기초하여 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)될 수 있다. 2차 변환 인덱스에 의해 지시되는 2차 변환(예컨대, 역방향 2차 변환)이 블록(예컨대, TB)에 대해 수행될 수 있다.
- [0174] 일 실시예에서, 2차 변환 인덱스(예컨대, stIdx)를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트 도출 프로세스는 블록의 크기에 의존할 수 있다. 일례로, 블록과 연관된 가장 큰 정사각형 블록 크기는

블록 내에서 가장 큰 정사각형 블록의 크기를 의미한다. 가장 큰 정사각형 블록 크기는 블록 크기보다 작거나 같다. 일례로, 블록이 TB를 의미할 때 블록의 크기는 TB 크기이다. 컨텍스트 도출 프로세스에서 가장 큰 정사각형 블록 크기가 사용될 수 있다. 일례로, 가장 큰 정사각형 블록 크기가 컨텍스트로서 사용된다. 예를 들어, 블록의 크기는 블록 폭(W)과 블록 높이(H)를 지시할 수 있다. 가장 큰 정사각형 블록 크기는 $L \times L$ 이며, 여기서 L은 W와 H의 최소값이다. 가장 큰 정사각형 블록 크기는 또한 L로 지시될 수 있다. W가 32라면, H는 8이고, L은 8이다. 가장 큰 정사각형 블록 크기는 8 또는 8×8 로 지시될 수 있다.

[0175] 일 실시예에서, 블록에 대한 인트라 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나가 블록의 크기를 지시할 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 블록의 크기에 기초하여 결정될 수 있다. 일례로, 블록의 크기는 블록의 폭(W)과 블록의 높이(H)를 지시한다. 블록의 폭(W)과 블록의 높이(H)의 최소값이 L로 표기된다. 컨텍스트는 가장 큰 정사각형 블록 크기(L 또는 $L \times L$)에 기초하여 결정될 수 있다. 일례로, 가장 큰 정사각형 블록 크기(L 또는 $L \times L$)가 컨텍스트로서 사용된다.

[0176] 일 실시예에서, 2차 변환 인덱스(예컨대, stIdx)를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트 도출 프로세스는 블록에 대한 예측 모드 정보(예컨대, 인트라 예측 모드 정보)에 의존할 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 블록에 대한 예측 모드 정보(예컨대, 인트라 예측 모드 정보)에 기초하여 도출될 수 있다.

[0177] 일례로, 블록은 이를테면, 도 9를 참조하여 설명된 (방향성 예측 모드로도 또한 지칭되는) 방향성 모드 중 하나를 사용하여 인트라 예측된다. 도 9를 참조하면, 공칭 모드 인덱스는 공칭 모드(예컨대, 8개의 공칭 모드 중 하나)를 지시할 수 있다. 일례로, 방향성 모드(예컨대, 56개의 방향성 모드 중 하나)는 공칭 모드 인덱스 및 공칭 모드로부터의 각도 오프셋을 지시하는 인덱스에 기초하여 결정될 수 있다. 블록에 대한 예측 모드 정보(예컨대, 인트라 예측 모드 정보)는 공칭 모드 인덱스를 지시할 수 있다. 공칭 모드 인덱스는 컨텍스트 도출 프로세스에 사용할 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 공칭 모드 인덱스에 기초하여 도출될 수 있다.

[0178] 일 실시예에서, 블록에 대한 인트라 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나는 블록에 대한 인트라 예측 모드 정보를 지시할 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 블록에 대한 인트라 예측 모드 정보에 기초하여 결정될 수 있다. 일례로, 블록에 대한 인트라 예측 모드 정보는 공칭 모드 인덱스를 지시한다. 블록은 앞서 설명된 바와 같이, 공칭 모드 인덱스와 각도 오프셋을 기반으로 결정되는 방향성 예측 모드를 이용하여 예측될 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 공칭 모드 인덱스에 기초하여 결정될 수 있다.

[0179] 일 실시예에서, 블록은 위에서 설명된 바와 같이 방향성 모드를 사용하여 인트라 예측된다. 공칭 모드 인덱스는 공칭 모드 인덱스와 다른 인덱스 값에 매핑될 수 있으며, 인덱스 값은 컨텍스트 도출 프로세스에서 사용될 수 있다. 일례로, 다수의 인트라 예측 모드(예컨대, 도 9에 도시된 다수의 공칭 모드)가 동일한 인덱스 값에 매핑될 수 있다. 다수의 공칭 모드(또는 다수의 공칭 모드에 대응하는 다수의 공칭 모드 인덱스)가 동일한 인덱스 값에 매핑될 수 있다. 다수의 공칭 모드는 서로 인접할 수 있다. 도 9를 참조하면, D67_PRED 및 D45_PRED가 동일한 인덱스 값에 매핑될 수 있다.

[0180] 일 실시예에서, 블록에 대한 인트라 예측 모드 정보는 공칭 모드 인덱스를 지시하고, 블록은 공칭 모드 인덱스 및 각도 오프셋에 기초하여 결정되는 방향성 모드(또는 방향성 예측 모드)를 사용하여 예측될 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 공칭 모드 인덱스와 연관된 인덱스 값에 기초하여 결정될 수 있다(예컨대, 공칭 모드 인덱스는 인덱스 값에 매핑된다).

[0181] 일 실시예에서, 블록은 도 10을 참조하여 설명한 바와 같이, 예를 들어 DC 모드, PAETH 모드, SMOOTH 모드, SMOOTH_V 모드 및 SMOOTH_H 모드를 포함하는 (무지향성 평활 인트라 예측 모드로도 또한 지칭되는) 무지향성 예측 모드 중 하나를 사용하여 인트라 예측된다. 무지향성 예측 모드 중 하나를 지시하는 (무지향성 예측 모드 인덱스로도 또한 지칭되는) 대응하는 모드 인덱스가 컨텍스트 도출 프로세스에 사용될 수 있다. 일례로, 동일한 컨텍스트가 적용되어 무지향성 예측 모드 중 하나 이상에 대한 2차 변환 인덱스(예컨대, stIdx)를 엔트로피 코딩할 수 있다. 동일한 컨텍스트는 복수의 무지향성 예측 모드에 대응할 수 있다.

[0182] 일 실시예에서, 블록에 대한 인트라 예측 모드 정보는 무지향성 예측 모드 인덱스를 지시할 수 있다. 블록은

무지향성 예측 모드 인덱스에 의해 지시된 무지향성 예측 모드들 중 하나(예컨대, 무지향성 예측 모드)를 사용하여 예측될 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 무지향성 예측 모드 인덱스에 기초하여 결정될 수 있다.

- [0183] 일례로, 블록은 이를테면, 도 11을 참조하여 설명된 재귀적 필터링 모드 중 하나를 사용하여 인트라 예측된다. 매핑은 먼저, 재귀 필터링 모드 중 하나로부터 공칭 모드 인덱스로 수행될 수 있으며, 여기서 공칭 모드 인덱스는 공칭 모드(예컨대, 도 9의 8개의 공칭 모드 중 하나)를 지시할 수 있다. 공칭 모드 인덱스는 이후에 컨텍스트 도출 프로세스에 사용될 수 있다. 일례로, 동일한 컨텍스트가 적용되어 재귀적 필터링 모드 중 하나 이상에 대한 2차 변환 인덱스(예컨대, stIdx)를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)할 수 있다. 동일한 컨텍스트가 복수의 재귀적 필터링 모드에 대응할 수 있다.
- [0184] 일 실시예에서, 블록에 대한 인트라 예측 모드 정보가 블록을 예측하는 데 사용되는 재귀적 필터링 모드(재귀적 필터링 모드 중 하나)를 지시할 수 있다. 공칭 모드를 지시하는 공칭 모드 인덱스는 재귀적 필터링 모드를 기반으로 결정될 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 공칭 모드 인덱스에 기초하여 결정될 수 있다.
- [0185] 일 실시예에서, 2차 변환 인덱스(예컨대, stIdx)를 엔트로피 코딩하기 위한 컨텍스트 도출 프로세스는 1차 변환 타입 정보에 의존할 수 있다. 1차 변환 타입 정보는 1차 변환 타입 또는 블록에 사용되는 1차 변환의 타입을 지시할 수 있다. 일례로, 컨텍스트는 1차 변환 타입을 기반으로 결정된다. 일례로, 컨텍스트는 1차 변환을 기반으로 결정된다.
- [0186] 일 실시예에서, 블록에 대한 인트라 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나는 블록에 사용되는 1차 변환 타입을 지시한다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 블록에 사용되는 1차 변환 타입에 기초하여 결정될 수 있다. 2차 변환 인덱스를 엔트로피 코딩하기 위한 컨텍스트는 블록에 사용되는 1차 변환에 기초하여 결정될 수 있다.
- [0187] 일 실시예에서, 블록에 대한 1차 변환은 (수평 변환으로도 지칭되는) 수평 1차 변환 및 (수직 변환으로도 지칭되는) 수직 1차 변환을 포함할 수 있다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 수평 1차 변환 타입 및 수직 1차 변환 타입에 기초하여 결정될 수 있다. 수평 1차 변환 타입은 수평 변환을 지시할 수 있고, 수직 1차 변환 타입은 수직 변환을 지시할 수 있다.
- [0188] (컨텍스트 값으로도 또한 지칭되는) 컨텍스트는 수평 1차 변환 타입과 수직 1차 변환 타입 둘 다 DCT인지 또는 수평 1차 변환 타입과 수직 1차 변환 타입 둘 다 ADST인지에 의존할 수 있다.
- [0189] 본 개시내용에서, 수평 1차 변환 타입과 수직 1차 변환 타입의 조합은 {수평 1차 변환 타입, 수직 1차 변환 타입}으로서 표현될 수 있다. 따라서 {DCT, DCT}는 수평 1차 변환 타입과 수직 1차 변환 타입이 둘 다 DCT임을 나타낸다. {ADST, ADST}는 수평 1차 변환 타입과 수직 1차 변환 타입이 둘 다 ADST임을 나타낸다. {LGT, LGT}는 수평 1차 변환 타입과 수직 1차 변환 타입이 둘 다 LGT임을 나타낸다. {DCT, LGT}는 수평 1차 변환 타입은 DCT이고 수직 1차 변환 타입은 LGT임을 나타낸다. {LGT, DCT}는 수평 1차 변환 타입은 LGT이고 수직 1차 변환 타입은 DCT임을 나타낸다.
- [0190] 일 실시예에서, 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함한다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 수평 1차 변환 타입과 수직 1차 변환 타입이 둘 다 DCT인지 또는 둘 다 ADST인지에 기초하여 결정될 수 있다.
- [0191] 일 실시예에서, 컨텍스트는 수평 1차 변환 타입과 수직 1차 변환 타입의 조합이 {DCT, DCT}인지 또는 {LGT, LGT}인지에 의존할 수 있다.
- [0192] 일 실시예에서, 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함한다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 수평 1차 변환 타입과 수직 1차 변환 타입이 둘 다 DCT인지 또는 둘 다 LGT인지를 기초로 결정될 수 있다.
- [0193] 일 실시예에서, 컨텍스트는 수평 1차 변환 타입과 수직 1차 변환 타입의 조합이 {DCT, DCT}인지, {LGT, LGT}인지, {DCT, LGT}인지 또는 {LGT, DCT}인지에 의존할 수 있다.
- [0194] 일 실시예에서, 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및

수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함한다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 수평 1차 변환 타입과 수직 1차 변환 타입이 (i) 둘 다 DCT인지, (ii) 둘 다 LGT인지, (iii) 각각 DCT 및 LGT인지, 또는 (iv) 각각 LGT 및 DCT에 기초하여 결정될 수 있다.

[0195] 일 실시예에서, 컨텍스트는 수평 1차 변환 타입과 수직 1차 변환 타입의 조합이 {DCT, DCT}인지, {LGT, LGT}인지, {DCT, IDTX}인지 또는 {IDTX, DCT}인지에 의존할 수 있으며, 여기서 IDTX는 항등 변환을 나타낸다. {DCT, IDTX}는 수평 1차 변환 타입은 DCT이고 수직 1차 변환 타입은 IDTX임을 나타낸다. {IDTX, DCT}는 수평 1차 변환 타입은 IDTX이고 수직 1차 변환 타입은 DCT임을 나타낸다.

[0196] 일 실시예에서, 1차 변환 타입에 의해 지시되는 1차 변환은 수평 1차 변환 타입에 의해 지시되는 수평 변환 및 수직 1차 변환 타입에 의해 지시되는 수직 변환을 포함한다. 2차 변환 인덱스를 엔트로피 코딩(예컨대, 인코딩 및/또는 디코딩)하기 위한 컨텍스트는 수평 1차 변환 타입 및 수직 1차 변환 타입이 (i) 둘 다 DCT인지, (ii) 둘 다 LGT인지, (iii) 각각 DCT 및 항등 변환(IDTX)인지, 또는 (iv) 각각 IDTX 및 DCT인지에 기초하여 결정될 수 있다.

[0197] 도 19는 본 개시내용의 일 실시예에 따른 프로세스(1900)의 개요를 서술하는 흐름도를 도시한다. 프로세스(1900)는 CB, TB, 루마 CB, 루마 TB, 크로마 CB, 크로마 TB 등과 같은 블록의 재구성에 사용될 수 있다. 다양한 실시예에서, 프로세스(1900)는 단말 디바이스(310, 320, 330, 340)의 처리 회로, 비디오 인코더(403)의 기능을 수행하는 처리 회로, 비디오 디코더(410)의 기능을 수행하는 처리 회로, 비디오 디코더(510)의 기능을 수행하는 처리 회로, 비디오 인코더(603)의 기능을 수행하는 처리 회로 등과 같은 처리 회로에 의해 실행된다. 일부 실시예에서, 프로세스(1900)는 소프트웨어 명령으로 구현되며, 따라서 처리 회로가 소프트웨어 명령을 실행할 때, 처리 회로는 프로세스(1900)를 수행한다. 프로세스는 (S1901)에서 시작하며 (S1910)으로 진행한다.

[0198] (S1910)에서, 코딩된 비디오 비트스트림으로부터 블록(예컨대, TB, 루마 TB, 인트라 코딩된 TB, CB)에 대한 코딩 정보가 코딩된 비디오 비트스트림으로부터 디코딩될 수 있다. 코딩된 정보는 블록에 대한 인트라 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나를 지시할 수 있다. 코딩된 정보는 블록에 대한 인트라 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나 이상을 지시할 수 있다.

[0199] 블록에 대한 인트라 예측 모드 정보는 블록의 인트라 예측에 사용되는 인트라 예측 모드, 이를테면 도 9 - 도 11을 참조하여 설명된 방향성 모드, 무지향성 예측 모드 또는 재귀적 필터링 모드를 지시할 수 있다. 1차 변환 타입은 블록에 사용되는 1차 변환을 지시한다. 블록에 사용되는 1차 변환 타입의 예는 도 15 및 도 16a - 도 16d에서 설명된다.

[0200] (S1920)에서, 블록에 대한 인트라 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나에 기초하여 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트가 결정될 수 있다. 2차 변환 인덱스는 한 세트의 2차 변환 중에서 2차 변환을 지시할 수 있는데, 여기서 그 2차 변환이 블록에 대해 수행될 것이다.

[0201] 일례로, 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트는 블록에 대한 인트라 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나 이상에 기초하여 결정될 수 있다.

[0202] 일례로, 블록에 대한 인트라 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나는 블록의 크기를 지시하고, 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트는 블록의 크기에 기초하여 결정될 수 있다.

[0203] 일례로, 블록에 대한 인트라 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나는 블록에 대한 인트라 예측 모드 정보를 지시하고, 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트는 블록에 대한 인트라 예측 모드 정보에 기초하여 결정될 수 있다.

[0204] 일례로, 블록에 대한 인트라 예측 모드 정보, 블록의 크기, 및 블록에 사용되는 1차 변환 타입 중 하나는 블록에 사용되는 1차 변환 타입을 지시하고, 2차 변환 인덱스를 엔트로피 디코딩하기 위한 컨텍스트는 블록에 사용되는 1차 변환 타입에 기초하여 결정될 수 있다.

[0205] (S1930)에서는, (S1920)에서 결정되는 컨텍스트에 기초하여 2차 변환 인덱스가 엔트로피 디코딩될 수 있다.

[0206] (S1940)에서, 2차 변환 인덱스에 의해 지시된 2차 변환이 블록에 대해 수행될 수 있다. 예를 들어, 2차 변환은 역방향 1차 변환 후에 수행되는 역방향 2차 변환이다. 일례로, 2차 변환은 LFNST이다. 일례로, 2차 변환은 비분리 2차 변환이다. 프로세스(1900)는 (S1999)로 진행하여 종료된다.

- [0207] 프로세스(1900)는 적절하게 적용될 수 있다. 프로세스(1900)의 단계(들)는 수정 및/또는 생략될 수 있다. 추가 단계가 추가될 수 있다. 임의의 적절한 구현 순서가 사용될 수 있다. 예를 들어, 도 9 - 도 11을 참조하여 설명된 방향성 모드, 무지향성 예측 모드 또는 재귀적 필터링 모드에 추가하여, 프로세스(1900)는 다른 인트라 예측 모드에도 또한, 적용 가능하다. 도 15 및 도 16a - 도 16d를 참조하여 설명된 1차 변환 타입에 추가하여, 프로세스(1900)는 또한 다른 적절한 1차 변환 타입 또는 수평 1차 변환 타입과 수직 1차 변환 타입의 적절한 조합에도 적용 가능하다.
- [0208] 본 개시내용의 실시예는 개별적으로 사용될 수 있거나 임의의 순서로 조합될 수 있다. 추가로, 방법(또는 실시예), 인코더 및 디코더 각각은 처리 회로(예컨대, 하나 이상의 프로세서 또는 하나 이상의 집적 회로)에 의해 구현될 수 있다. 일례로, 하나 이상의 프로세서는 비-일시적 컴퓨터 판독 가능 매체에 저장된 프로그램을 실행한다. 본 개시내용의 실시예는 루마 블록 또는 크로마 블록에 적용될 수 있다.
- [0209] 위에서 설명된 기법들은 컴퓨터 판독 가능 명령들을 사용하여 컴퓨터 소프트웨어로서 구현될 수 있고 하나 이상의 컴퓨터 판독 가능 매체에 물리적으로 저장될 수 있다. 예를 들어, 도 20은 개시된 청구 대상의 특정 실시예를 구현하기에 적합한 컴퓨터 시스템(2000)을 도시한다.
- [0210] 컴퓨터 소프트웨어는 하나 이상의 컴퓨터 중앙 처리 유닛(CPU: central processing unit), 그래픽 처리 유닛(GPU: Graphics Processing Unit) 등에 의해, 직접적으로 또는 해석, 마이크로코드 실행 등을 통해 실행될 수 있는 명령을 포함하는 코드를 생성하기 위해 어셈블리, 컴파일, 링크 또는 유사한 메커니즘을 거칠 수 있는 임의의 적절한 기계 코드 또는 컴퓨터 언어를 사용하여 코딩될 수 있다.
- [0211] 명령은 예를 들어 개인용 컴퓨터, 태블릿 컴퓨터, 서버, 스마트폰, 게임 디바이스, 사물 인터넷 디바이스 등을 포함하는 다양한 타입의 컴퓨터 또는 이들의 컴포넌트 상에서 실행될 수 있다.
- [0212] 컴퓨터 시스템(2000)에 대해 도 20에 도시된 컴포넌트는 본질적으로 예시이며, 본 개시내용의 실시예를 구현하는 컴퓨터 소프트웨어의 사용 또는 기능의 범위에 대해 어떠한 제한도 제안하도록 의도되지 않는다. 컴포넌트의 구성이 컴퓨터 시스템(2000)의 예시적인 실시예에 예시된 컴포넌트 중 임의의 컴포넌트 또는 그러한 컴포넌트의 조합과 관련된 어떠한 의존성이나 요건도 갖는 것으로 해석되지 않아야 한다.
- [0213] 컴퓨터 시스템(2000)은 특정 인간 인터페이스 입력 디바이스를 포함할 수 있다. 그러한 인간 인터페이스 입력 디바이스는 예를 들어, 촉각 입력(이를테면: 키 스트로크(keystroke), 스와이프(swipe), 데이터 글러브(glove) 움직임), 오디오 입력(이를테면: 음성, 박수), 시각적 입력(이를테면: 제스처), 후각적 입력(도시되지 않음)을 통해 하나 이상의 인간 사용자에게 의한 입력에 응답할 수 있다. 인간 인터페이스 디바이스는 또한, 오디오(이를테면: 스피치, 음악, 주변 사운드), 이미지(이를테면: 스캔된 이미지, 정지 이미지 카메라로부터 획득된 사진 이미지), 비디오(이를테면, 2차원 비디오, 입체 비디오를 포함하는 3차원 비디오)와 같은, 인간에 의한 의식적 입력과 반드시 직접적으로 관련되는 것은 아닌 특정 미디어를 캡처하는 데 사용될 수 있다.
- [0214] 입력 인간 인터페이스 디바이스는: 키보드(2001), 마우스(2002), 트랙 패드(2003), 터치 스크린(2010), (도시되지 않은) 데이터 글러브, 조이스틱(2005), 마이크로폰(2006), 스캐너(2007), 카메라(2008) 중 하나 이상(각각 도시된 것 중 단 하나)을 포함할 수 있다.
- [0215] 컴퓨터 시스템(2000)은 또한 특정 인간 인터페이스 출력 디바이스를 포함할 수 있다. 그러한 인간 인터페이스 출력 디바이스는 예를 들어, 촉각 출력, 사운드, 광 및 후각/미각을 통해 하나 이상의 인간 사용자의 감각을 자극할 수 있다. 그러한 인간 인터페이스 출력 디바이스는 촉각 출력 디바이스(예를 들어, 터치 스크린(2010), (도시되지 않은) 데이터 글러브 또는 조이스틱(2005)에 의한 촉각 피드백, 그러나 입력 디바이스로서 기능하지 않는 촉각 피드백 디바이스가 또한 존재할 수 있음), 오디오 출력 디바이스(이를테면: 스피커(2009), (도시되지 않은) 헤드폰들), 시각적 출력 디바이스(이를테면, 각각 터치 스크린 입력 능력을 갖거나 갖지 않고, 각각 촉각적 피드백 능력을 갖거나 갖지 않는 CRT 스크린, LCD 스크린, 플라즈마 스크린, OLED 스크린을 포함하는 스크린(2010) - 이들 중 일부는 스테레오그래픽 출력과 같은 수단을 통해 2차원 시각적 출력 또는 3차원 이상의 출력을 출력하는 것이 가능할 수 있음 -; (도시되지 않은) 가상 현실 안경, (도시되지 않은) 홀로그래픽 디스플레이 및 연기 탱크), 및 (도시되지 않은) 프린터를 포함할 수 있다.
- [0216] 컴퓨터 시스템(2000)은 또한, 인간 액세스 가능 저장 디바이스 및 이들의 연관된 매체, 이를테면 CD/DVD 또는 유사한 매체(2021)를 갖는 CD/DVD ROM/RW(2020), 썸 드라이브(thumb-drive)(2022), 착탈식 하드 드라이브 또는 솔리드 스테이트 드라이브(2023), 레저시 자기 매체, 이를테면 (도시되지 않은) 테이프 및 플로피 디스크, 특수 ROM/ASIC/PLD 기반 디바이스, 이를테면 (도시되지 않은) 보안 동글(security dongle) 등을 포함하는 광학 매체

를 포함할 수 있다.

- [0217] 당해 기술분야에서 통상의 지식을 가진 자는 또한, 현재 개시된 청구 대상과 관련하여 사용되는 바와 같은 "컴퓨터 판독 가능 매체"라는 용어가 전송 매체, 반송파, 또는 다른 일시적인 신호를 포괄하지 않는다고 이해해야 한다.
- [0218] 컴퓨터 시스템(2000)은 또한 하나 이상의 통신 네트워크(2055)에 대한 인터페이스(2054)를 포함할 수 있다. 네트워크는 예를 들어 무선, 유선, 광일 수 있다. 네트워크는 추가로, 근거리, 광역, 대도시, 차량 및 산업, 실시간, 지연 용인(delay-tolerant) 등일 수 있다. 네트워크의 예는 근거리 네트워크, 이더넷, 무선 LAN, GSM, 3G, 4G, 5G, LTE 등을 포함하는 셀룰러 네트워크, 케이블 TV, 위성 TV 및 지상 브로드캐스트 TV를 포함하는 TV 유선 또는 무선 광역 디지털 네트워크, CANBus를 포함하는 차량 및 산업용 등을 포함한다. 특정 네트워크는 일반적으로, (예를 들어, 컴퓨터 시스템(2000)의 USB 포트와 같은) 특정 범용 데이터 포트 또는 주변 버스(2049)에 부착된 외부 네트워크 인터페이스 어댑터를 필요로 하며; 다른 것은 일반적으로, 아래에서 설명되는 바와 같은 시스템 버스에 대한 부착(예를 들어, PC 컴퓨터 시스템으로의 이더넷 인터페이스 또는 스마트폰 컴퓨터 시스템으로의 셀룰러 네트워크 인터페이스)에 의해 컴퓨터 시스템(2000)의 코어에 통합된다. 이러한 네트워크 중 임의의 네트워크를 사용하여, 컴퓨터 시스템(2000)은 다른 엔티티와 통신할 수 있다. 그러한 통신은 예를 들어, 근거리 또는 광역 디지털 네트워크를 사용하는 다른 컴퓨터 시스템에 대해 단방향, 수신 전용(예를 들어, 브로드캐스트 TV), 단방향 송신 전용(예를 들어, 특정 CANbus 디바이스에 대한 CANbus), 또는 양방향 일 수 있다. 위에서 설명된 바와 같이, 특정 프로토콜 및 프로토콜 스택이 그러한 네트워크 및 네트워크 인터페이스 각각에 사용될 수 있다.
- [0219] 앞서 언급한 인간 인터페이스 디바이스, 인간 액세스 가능 저장 디바이스 및 네트워크 인터페이스는 컴퓨터 시스템(2000)의 코어(2040)에 부착될 수 있다.
- [0220] 코어(2040)는 하나 이상의 중앙 처리 유닛(CPU)(2041), 그래픽 처리 유닛(GPU)(2042), 필드 프로그래밍 가능 게이트 영역(FPGA: Field Programmable Gate Area)(2043) 형태의 특수 프로그래밍 가능 처리 유닛, 특정 작업을 위한 하드웨어 가속기(2044), 그래픽 어댑터(2050) 등을 포함할 수 있다. 이러한 디바이스는 판독 전용 메모리(ROM: Read-only memory)(2045), 랜덤 액세스 메모리(2046), 내부 비-사용자 액세스 가능 하드 드라이브, SSD 등과 같은 내부 대용량 저장소(2047)와 함께, 시스템 버스(2048)를 통해 접속될 수 있다. 일부 컴퓨터 시스템에서, 시스템 버스(2048)는 추가 CPU, GPU 등에 의한 확장을 가능하게 하도록 하나 이상의 물리적 플러그의 형태로 액세스 가능할 수 있다. 주변 디바이스는 코어의 시스템 버스(2048)에 직접적으로 또는 주변 버스(2049)를 통해 부착될 수 있다. 일례로, 스크린(2010)은 그래픽 어댑터(2050)에 접속될 수 있다. 주변 버스를 위한 아키텍처는 PCI, USB 등을 포함한다.
- [0221] CPU(2041), GPU(2042), FPGA(2043) 및 가속기(2044)는 조합하여, 앞서 언급된 컴퓨터 코드를 구성할 수 있는 특정 명령을 실행할 수 있다. 그 컴퓨터 코드는 ROM(2045) 또는 RAM(2046)에 저장될 수 있다. 전이(transitional) 데이터가 또한 RAM(2046)에 저장될 수 있는 반면, 영구 데이터는 예를 들어, 내부 대용량 저장소(2047)에 저장될 수 있다. 메모리 디바이스 중 임의의 디바이스에 대한 고속 저장 및 리트리브는 하나 이상의 CPU(2041), GPU(2042), 대용량 저장소(2047), ROM(2045) 및 RAM(2046) 등과 밀접하게 연관될 수 있는 캐시 메모리의 사용을 통해 가능해질 수 있다.
- [0222] 컴퓨터 판독 가능 매체는 다양한 컴퓨터 구현 동작을 수행하기 위한 컴퓨터 코드를 가질 수 있다. 매체 및 컴퓨터 코드는 본 개시내용의 목적으로 특별히 설계 및 구성된 것일 수 있거나, 이들은 컴퓨터 소프트웨어 분야에서 통상의 지식을 가진 자에게 잘 알려져 있고 이용 가능한 종류일 수 있다.
- [0223] 제한이 아닌 예로서, 아키텍처(2000)를 갖는 컴퓨터 시스템, 그리고 구체적으로 코어(2040)는 (CPU, GPU, FPGA, 가속기 등을 포함하는) 프로세서(들)가 하나 이상의 유형(tangible) 컴퓨터 판독 가능 매체에 구현된 소프트웨어를 실행하는 결과로서 기능을 제공할 수 있다. 그러한 컴퓨터 판독 가능 매체는 위에서 소개된 바와 같은 사용자 액세스 가능 대용량 저장소뿐만 아니라, 비-일시적 성질의 코어(2040)의 특정 저장소, 이더넷 코어 내부 대용량 저장소(2047) 또는 ROM(2045)과 연관된 매체일 수 있다. 본 개시내용의 다양한 실시예를 구현하는 소프트웨어는 그러한 디바이스에 저장되고 코어(2040)에 의해 실행될 수 있다. 컴퓨터 판독 가능 매체는 특정 요구에 따라 하나 이상의 메모리 디바이스 또는 칩을 포함할 수 있다. 소프트웨어는 코어(2040) 및 구체적으로는 코어(3040) 내의 (CPU, GPU, FPGA 등을 포함하는) 프로세서로 하여금, RAM(2046)에 저장된 데이터 구조를 정의하는 것과 소프트웨어에 의해 정의된 프로세스에 따라 그러한 데이터 구조를 수정하는 것을 포함하여, 본 명세서에서 설명된 특정 프로세스 또는 특정 프로세스의 특정 부분을 실행하게 할 수 있다. 추가로 또는 대안으로

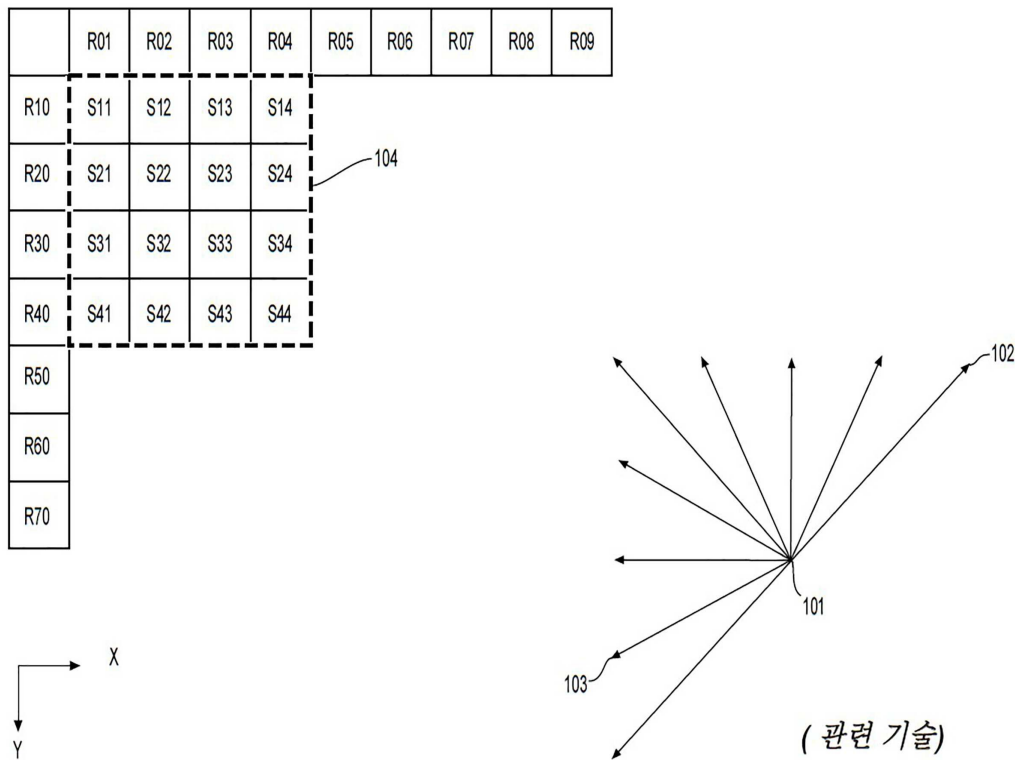
서, 컴퓨터 시스템은 본 명세서에서 설명되는 특정 프로세스 또는 특정 프로세스의 특정 부분을 실행하도록 소프트웨어 대신 또는 소프트웨어와 함께 동작할 수 있는 회로(예를 들어: 가속기(2044))에 하드와이어링되거나 달리 구현된 로직의 결과로서 기능을 제공할 수 있다. 적절한 경우, 소프트웨어에 대한 참조는 로직을 포함할 수 있고, 그 반대로 가능하다. 컴퓨터 판독 가능 매체에 대한 참조는 적절한 경우, 실행을 위한 소프트웨어를 저장하는 회로(이를테면, 집적 회로(IC: integrated circuit)), 실행을 위한 로직을 구현하는 회로, 또는 이 둘 모두를 포괄할 수 있다. 본 개시내용은 하드웨어와 소프트웨어의 임의의 적절한 조합을 포괄한다.

- [0224] 부록 A: 약어
- [0225] JEM: 공동 탐사 모델(joint exploration model)
- [0226] VVC: 다용도 비디오 코딩(versatile video coding)
- [0227] BMS: 벤치마크 세트(benchmark set)
- [0228] MV: 움직임 벡터(Motion Vector)
- [0229] HEVC: 고효율 비디오 코딩(High Efficiency Video Coding)
- [0230] SEI: 부가 확장 정보(Supplementary Enhancement Information)
- [0231] VUI: 비디오 사용성 정보(Video Usability Information)
- [0232] GOP: 픽처의 그룹(Group of Pictures)
- [0233] TU: 변환 유닛(Transform Unit)
- [0234] PU: 예측 유닛(Prediction Unit)
- [0235] CTU: 코딩 트리 유닛(Coding Tree Unit)
- [0236] CTB: 코딩 트리 블록(Coding Tree Block)
- [0237] PB: 예측 블록(Prediction Block)
- [0238] HRD: 가상 참조 디코더(Hypothetical Reference Decoder)
- [0239] SNR: 신호 잡음비(Signal Noise Ratio)
- [0240] CPU: 중앙 처리 유닛(Central Processing Unit)
- [0241] GPU: 그래픽 처리 유닛(Graphics Processing Unit)
- [0242] CRT: 음극선관(Cathode Ray Tube)
- [0243] LCD: 액정 디스플레이(Liquid-Crystal Display)
- [0244] OLED: 유기 발광 다이오드(Organic Light-Emitting Diode)
- [0245] CD: 콤팩트 디스크(Compact Disc)
- [0246] DVD: 디지털 비디오 디스크(Digital Video Disc)
- [0247] ROM: 판독 전용 메모리(Read-Only Memory)
- [0248] RAM: 랜덤 액세스 메모리(Random Access Memory)
- [0249] ASIC: 주문형 집적 회로(Application-Specific Integrated Circuit)
- [0250] PLD: 프로그래밍 가능한 로직 디바이스(Programmable Logic Device)
- [0251] LAN: 근거리 네트워크(Local Area Network)
- [0252] GSM: 글로벌 모바일 통신 시스템(Global System for Mobile communications)
- [0253] LTE: 롱 텀 에볼루션(Long-Term Evolution)
- [0254] CANBus: 제어기 영역 네트워크 버스(Controller Area Network Bus)

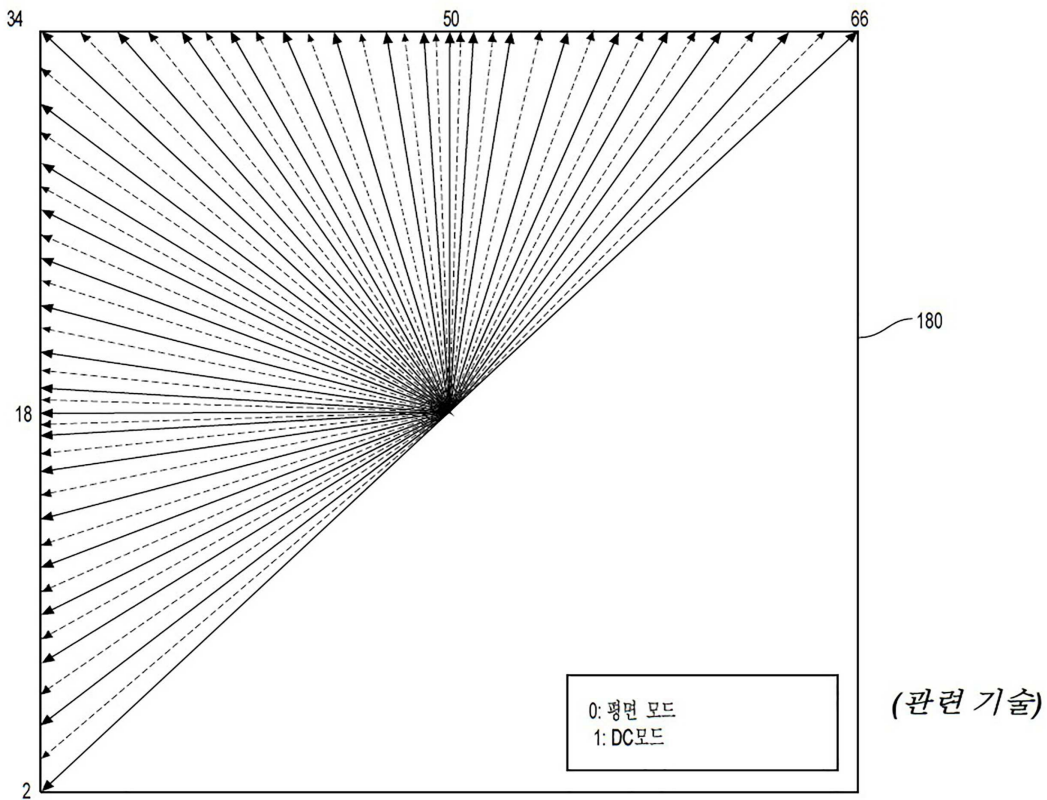
- [0255] USB: 범용 직렬 버스(Universal Serial Bus)
- [0256] PCI: 주변 컴포넌트 상호 접속(Peripheral Component Interconnect)
- [0257] FPGA: 필드 프로그래밍 가능 게이트 영역(Field Programmable Gate Areas)
- [0258] SSD: 솔리드 스테이트 드라이브(solid-state drive)
- [0259] IC: 집적 회로(Integrated Circuit)
- [0260] CU: 코딩 유닛(Coding Unit)
- [0261] 본 개시내용은 여러 예시적인 실시예를 설명했지만, 본 개시내용의 범위 내에 속하는 변경, 치환 및 다양한 대체 등가물이 있다. 따라서 당해 기술분야에서 통상의 지식을 가진 자는, 본 명세서에서 명시적으로 도시되거나 설명되지는 않지만, 본 개시내용의 원리를 구현하고 따라서 본 개시내용의 사상 및 범위 내에 있는 다수의 시스템 및 방법을 안출할 수 있을 것이라고 인식될 것이다.

도면

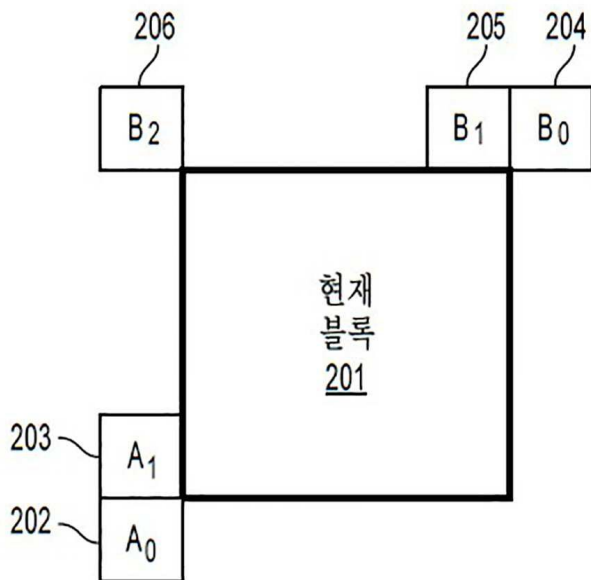
도면1a



도면1b

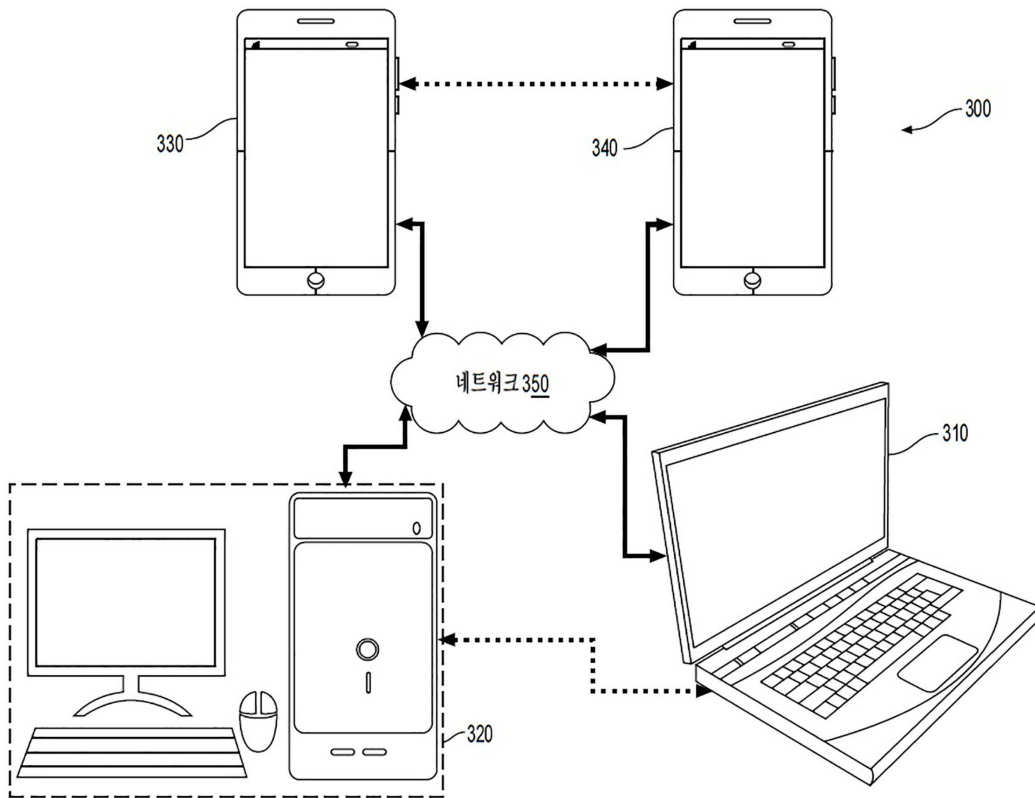


도면2

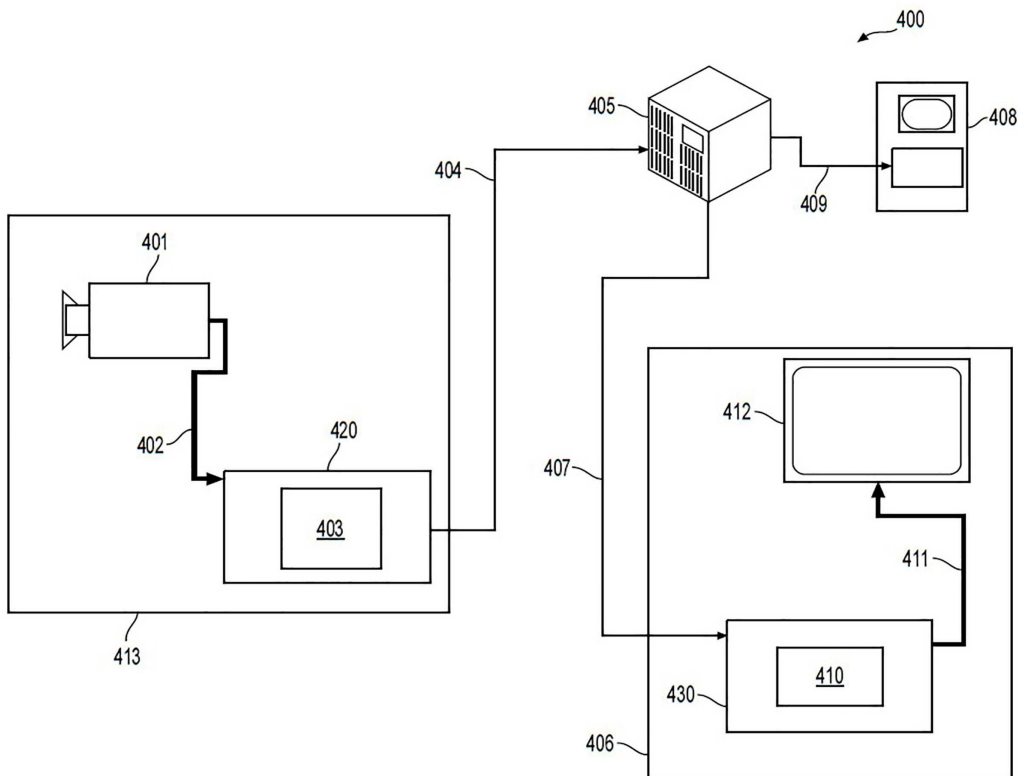


(관련 기술)

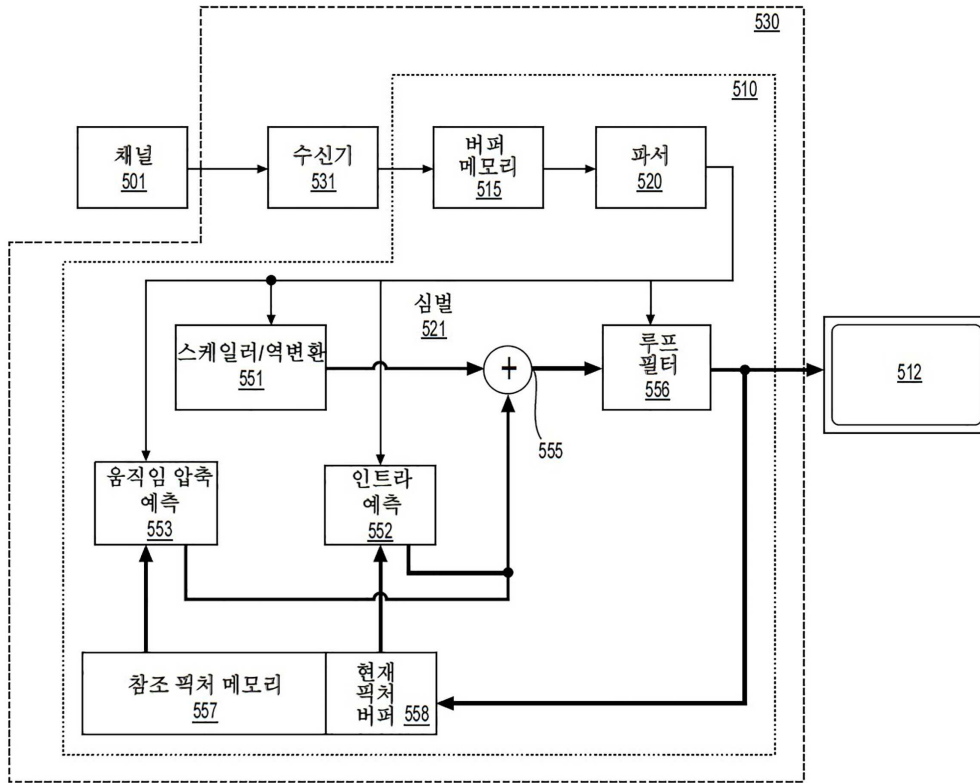
도면3



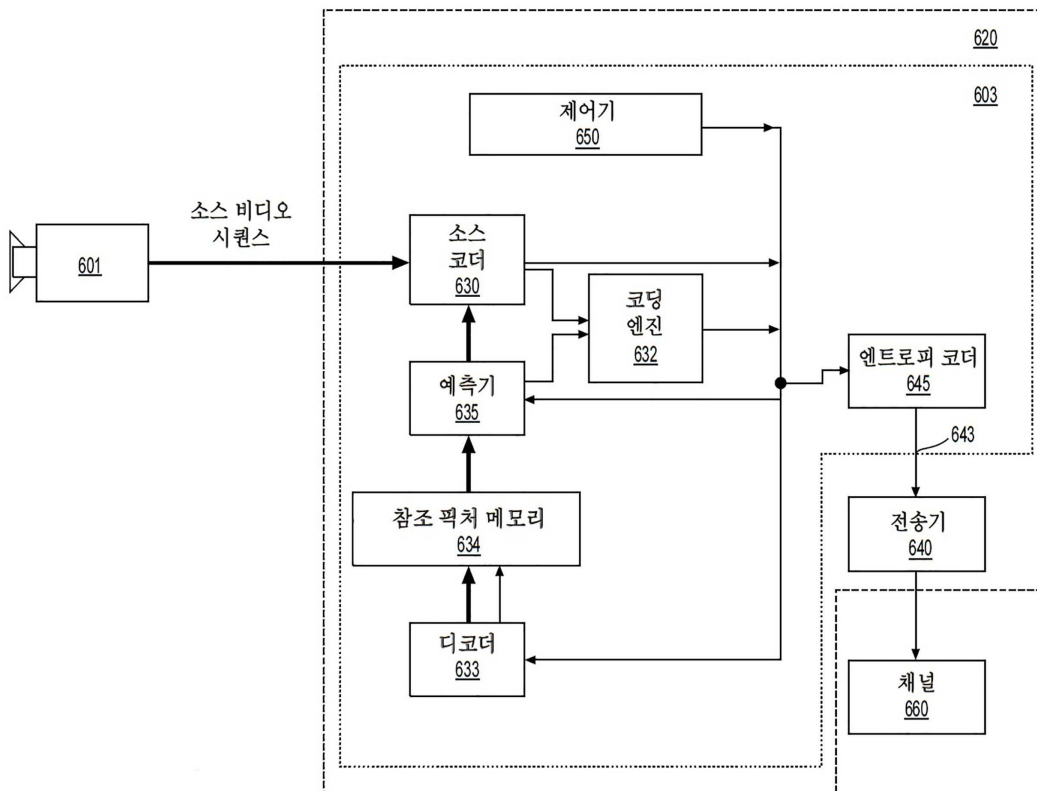
도면4



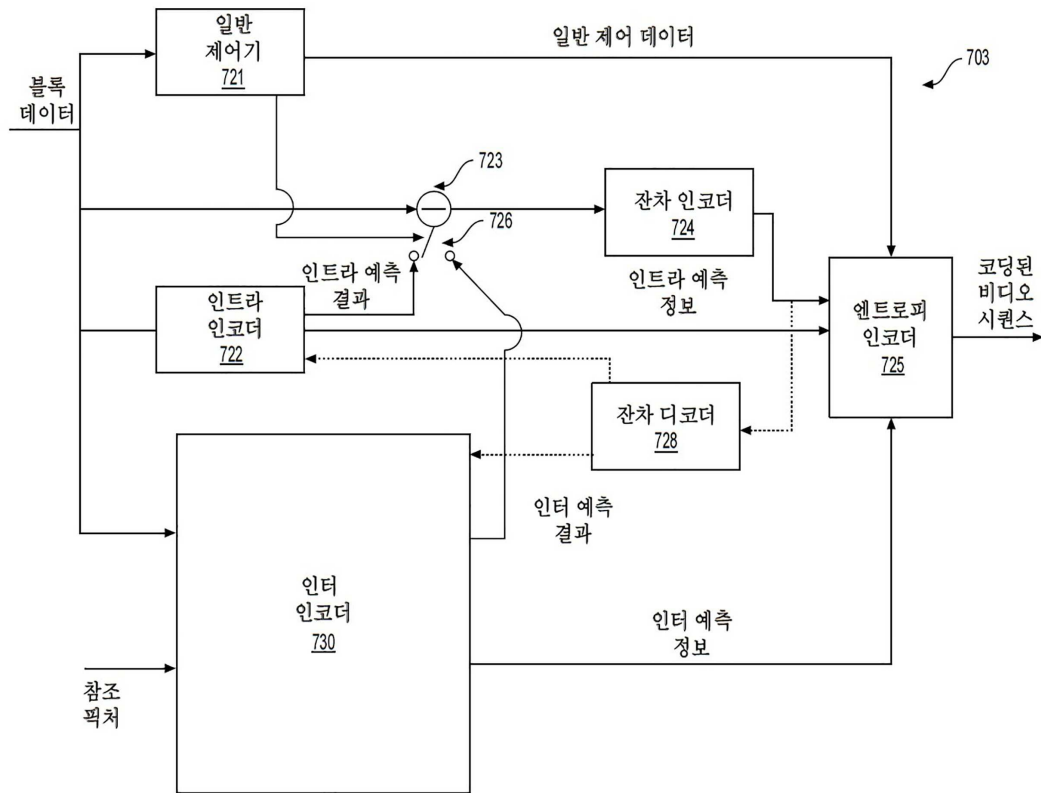
도면5



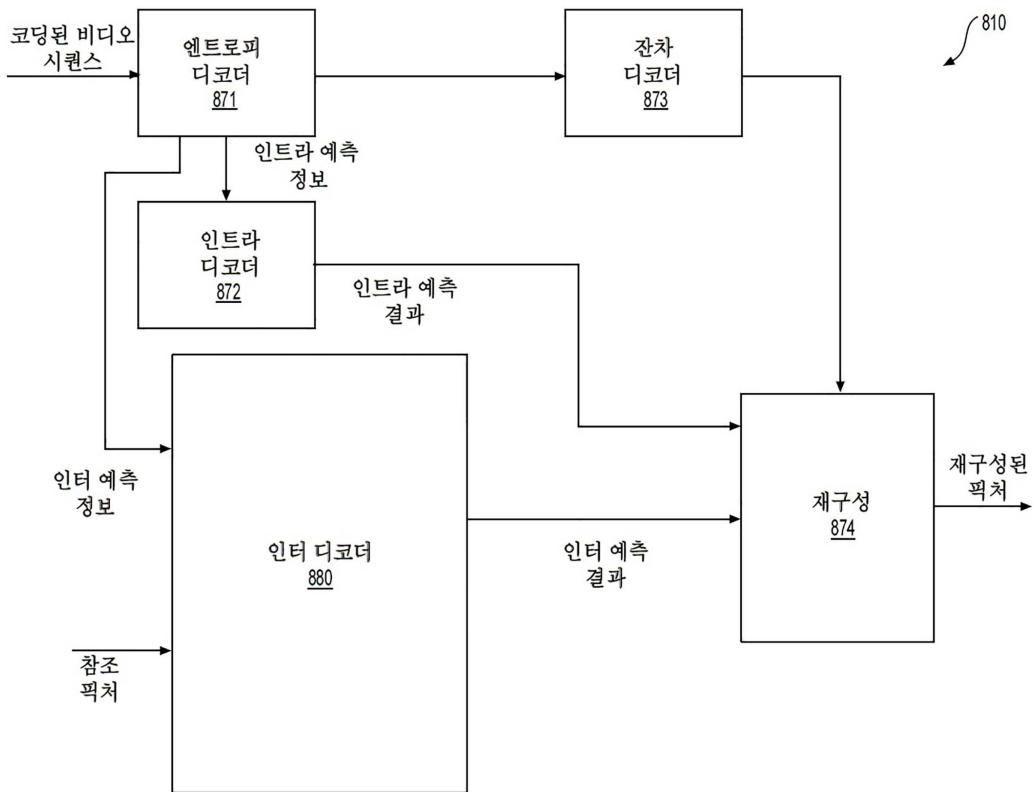
도면6



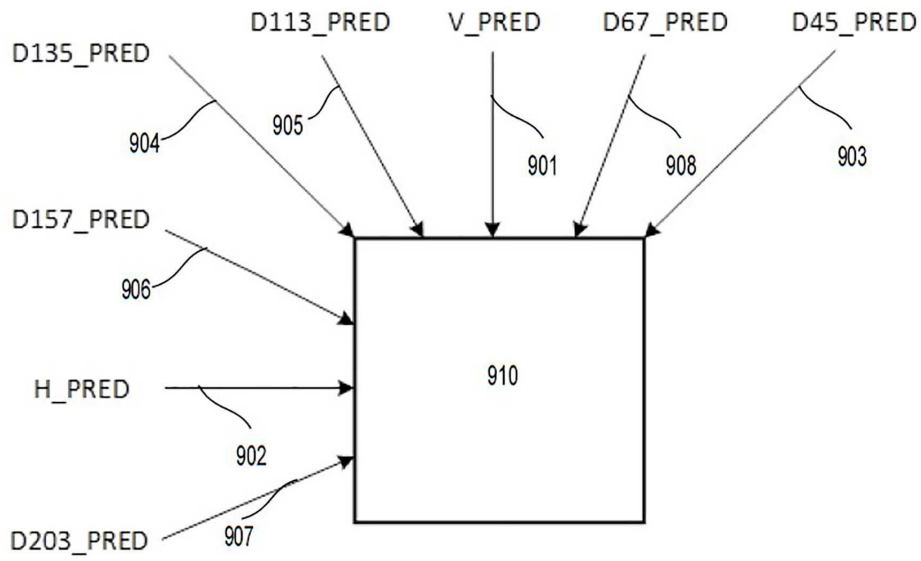
도면7



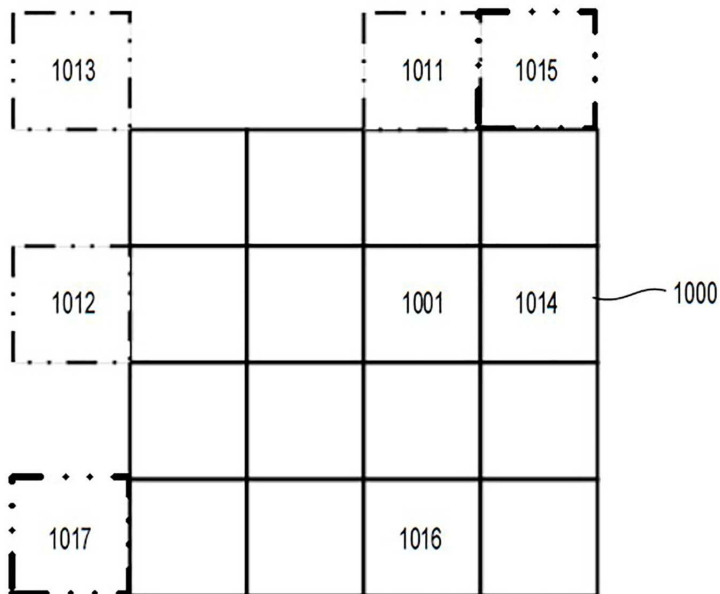
도면8



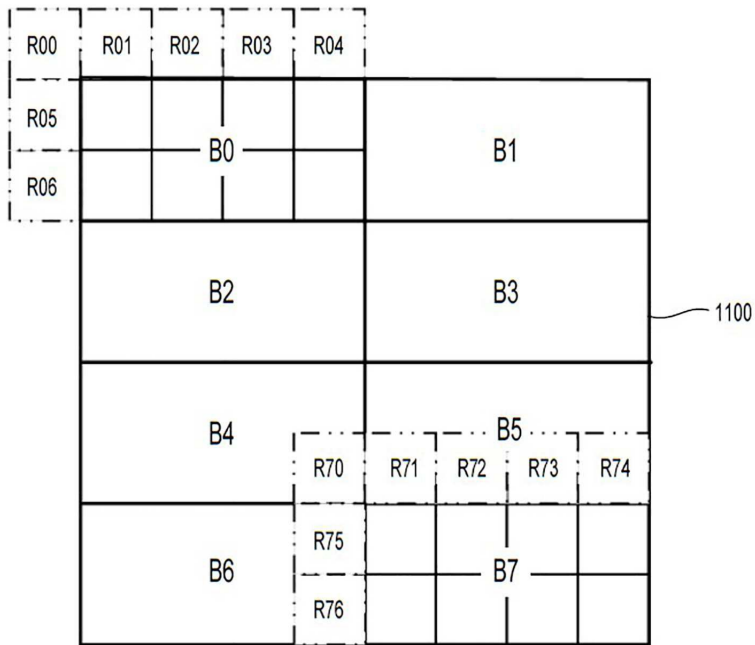
도면9



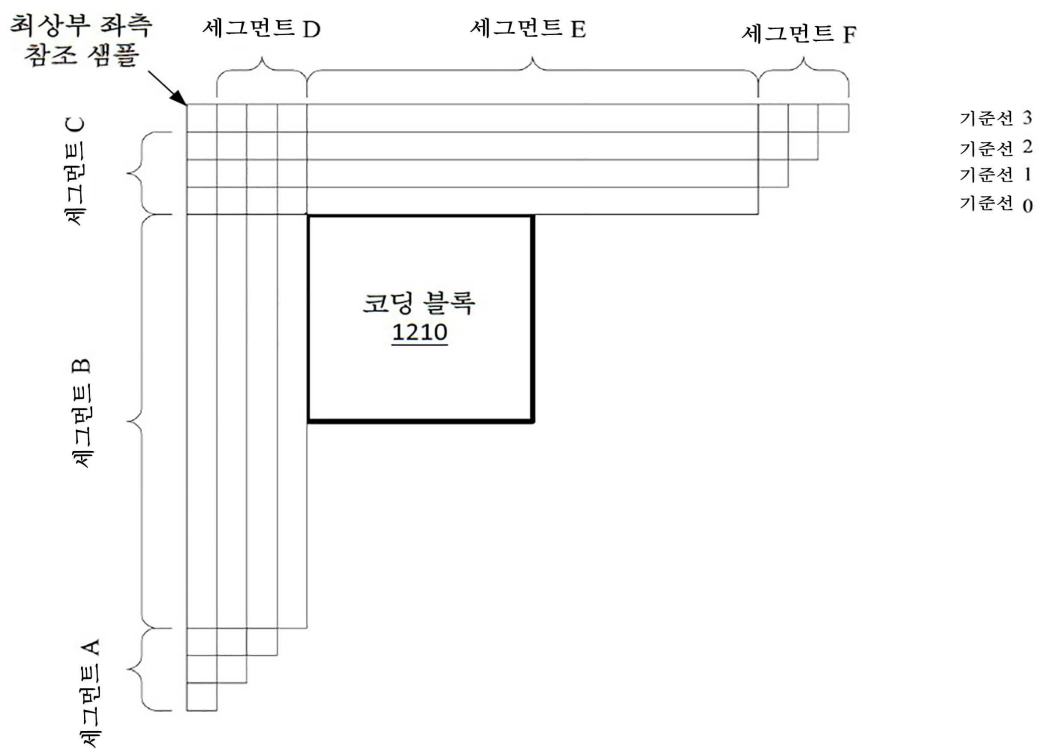
도면10



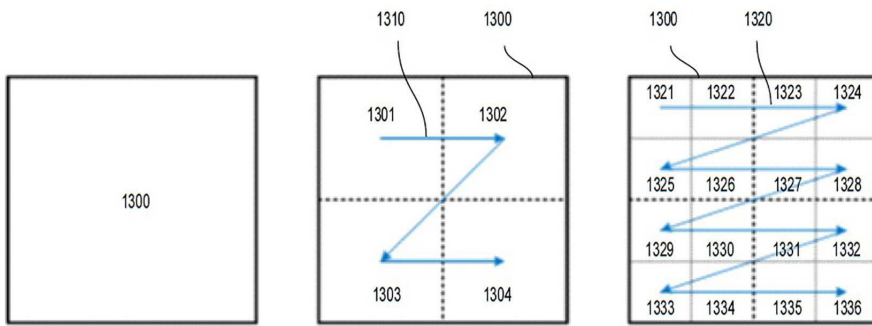
도면11



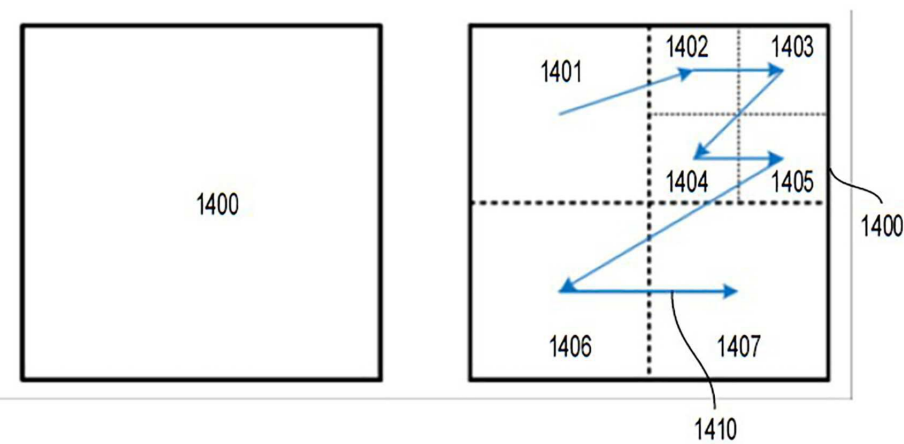
도면12



도면13



도면14



도면15

변환 타입	기저 함수 $T_i(j), i, j=0, 1, \dots, N-1$
DCT-2	$T_i(j) = \omega_0 \cdot \frac{2}{N} \cdot \cos \frac{\pi \cdot i \cdot (2j + 1)}{2N}$ <p>where $\omega_0 = \begin{cases} \frac{2}{N} & i = 0 \\ 1 & i \neq 0 \end{cases}$</p>
DST-4	$T_i(j) = \frac{2}{N} \cdot \sin \frac{\pi \cdot (2i + 1) \cdot (2j + 1)}{4N}$
DST-7	$T_i(j) = \frac{4}{2N + 1} \cdot \sin \frac{\pi \cdot (2i + 1) \cdot (j + 1)}{2N + 1}$

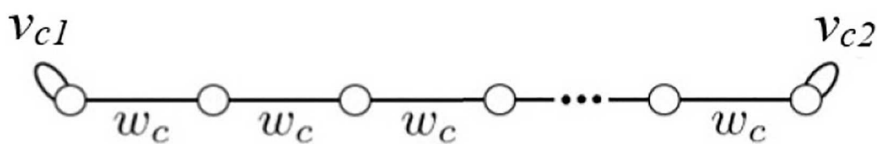
도면16a

변환 타입	설명	예측 모드	
		인트라	인터
DCT_DCT	DCT ↓ and →	✓ (모든 블록 크기)	✓ (모든 블록 크기)
ADST_DCT	ADST ↓; DCT →	✓ (블록 크기 ≤ 16×16)	✓ (블록 크기 ≤ 16×16)
DCT_ADST	DCT ↓; ADST →		
ADST_ADST	ADST ↓ and →		
FLIPADST_DCT	FLIPADST ↓; DCT →	×	✓ (블록 크기 ≤ 16×16)
DCT_FLIPADST	DCT ↓; FLIPADST →		
FLIPADST_FLIPADST	FLIPADST ↓ and →		
ADST_FLIPADST	ADST ↓; FLIPADST →		
FLIPADST_ADST	FLIPADST ↓; ADST →		
IDTX	IDTX ↓ and →	✓ (블록 크기 ≤ 16×16)	✓ (블록 크기 ≤ 32×32)
V_DCT	DCT ↓; IDTX →	✓ (블록 크기 < 16×16)	✓ (블록 크기 ≤ 16×16)
H_DCT	IDTX ↓; DCT →		
V_ADST	ADST ↓; IDTX →	×	✓ (블록 크기 < 16×16)
H_ADST	IDTX ↓; ADST →		
V_FLIPADST	FLIPADST ↓; IDTX →	×	✓ (블록 크기 < 16×16)
H_FLIPADST	IDTX ↓; FLIPADST →		

도면16b

인트라 예측	수직 변환	수평 변환
DC_PRED	DCT	DCT
V_PRED	ADST	DCT
H_PRED	DCT	ADST
D45_PRED	DCT	DCT
D135_PRED	ADST	ADST
D113_PRED	ADST	DCT
D157_PRED	DCT	ADST
D203_PRED	DCT	ADST
D67_PRED	ADST	DCT
SMOOTH_PRED	ADST	ADST
SMOOTH_V_PRED	ADST	DCT
SMOOTH_H_PRED	DCT	ADST
PAETH_PRED	ADST	ADST

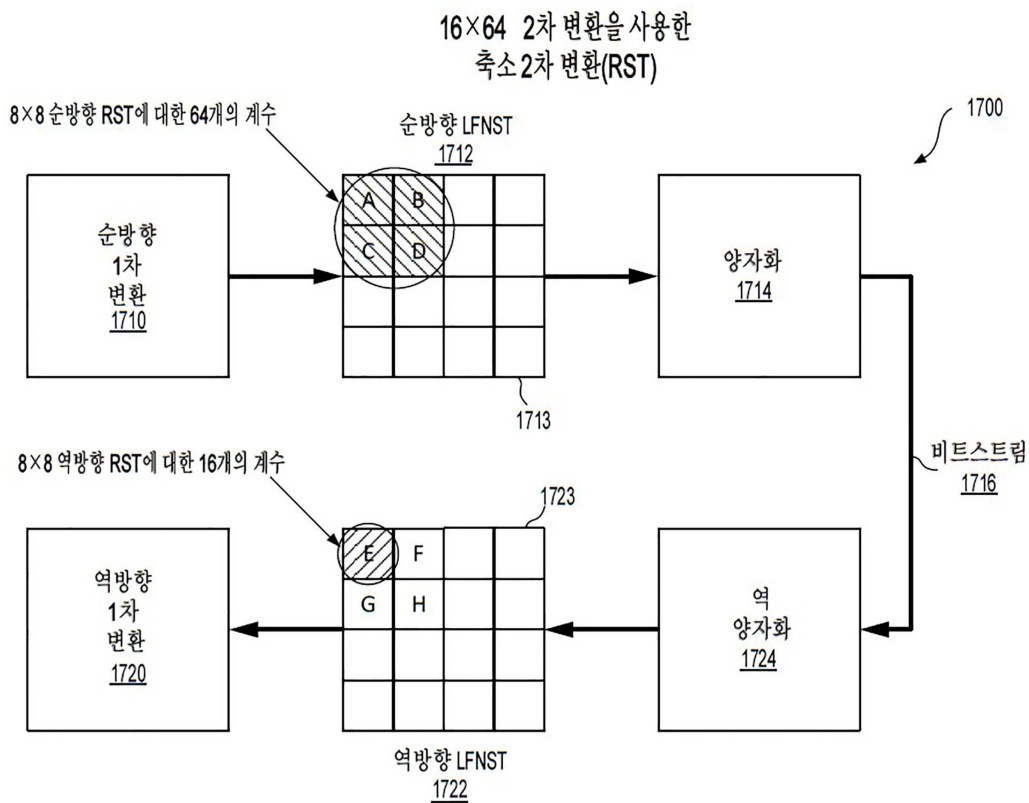
도면16c



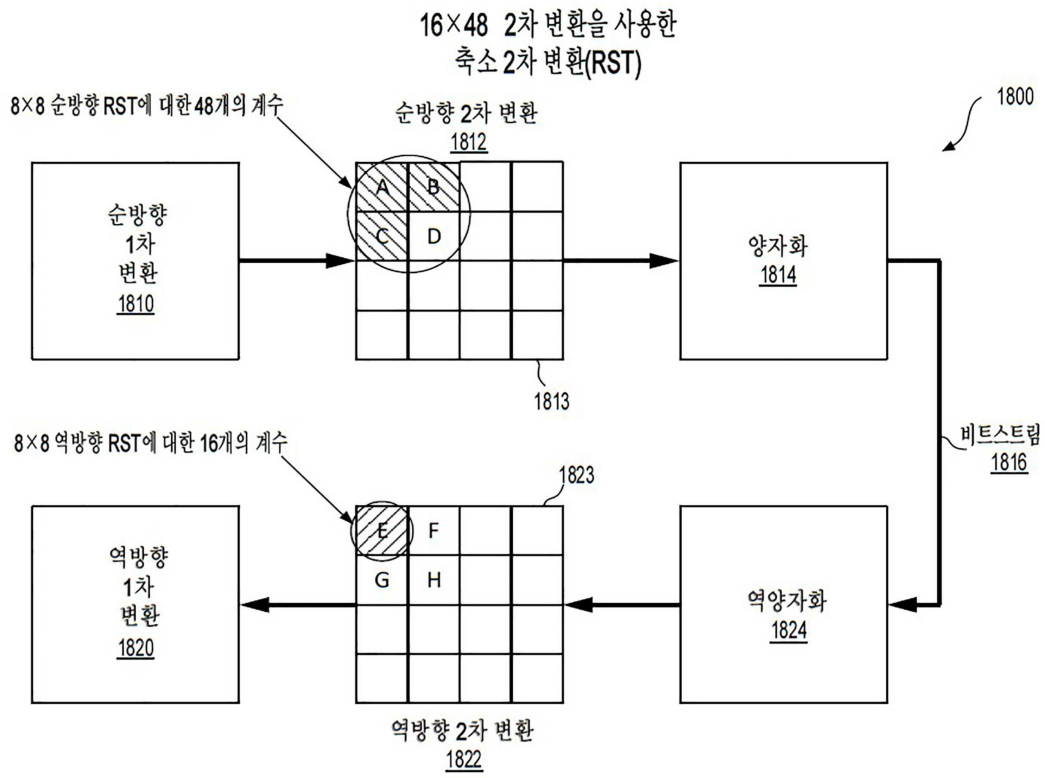
도면16d

$$\mathbf{L}_c = \begin{bmatrix} w_c + v_{c1} - w_c & & & & 0 \\ -w_c & 2w_c & -w_c & & \\ & \ddots & \ddots & \ddots & \\ & & -w_c & 2w_c & -w_c \\ 0 & & & -w_c & w_c + v_{c2} \end{bmatrix} \text{ for } w_c > 0$$

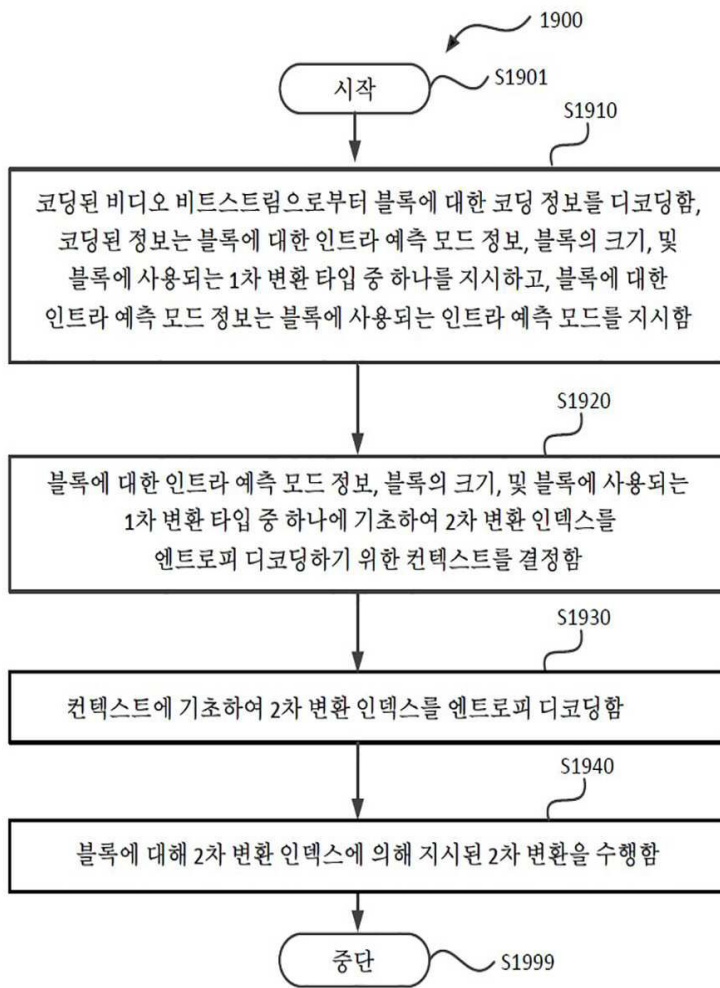
도면17



도면18



도면19



도면20

