



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2018년02월05일

(11) 등록번호 10-1825793

(24) 등록일자 2018년01월30일

- (51) 국제특허분류(Int. Cl.)
HO4N 19/21 (2014.01) *HO4N 19/119* (2014.01)
HO4N 19/122 (2014.01) *HO4N 19/176* (2014.01)
- (52) CPC특허분류
HO4N 19/21 (2015.01)
HO4N 19/119 (2015.01)
- (21) 출원번호 10-2015-7010767
- (22) 출원일자(국제) 2013년09월26일
 심사청구일자 2017년02월28일
- (85) 번역문제출일자 2015년04월24일
- (65) 공개번호 10-2015-0064116
- (43) 공개일자 2015년06월10일
- (86) 국제출원번호 PCT/US2013/061890
- (87) 국제공개번호 WO 2014/052567
 국제공개일자 2014년04월03일
- (30) 우선권주장
 61/706,035 2012년09월26일 미국(US)
 14/037,159 2013년09월25일 미국(US)

- (56) 선행기술조사문헌
 US20120155546 A1
 KR1020130056183 A
 T. Nguyen, et al. Improved context modeling for coding quantized transform coefficients in video compression. 28th Picture Coding Symposium. Dec. 2010, pp.378-381
 T. Tsukuba, et al. Refined significance map context derivation for large TU. JCT-VC of ITU-T and ISO/IEC. JCTVC-J0068 Ver.2, Jul. 10, 2012. pp.1-5

전체 청구항 수 : 총 14 항

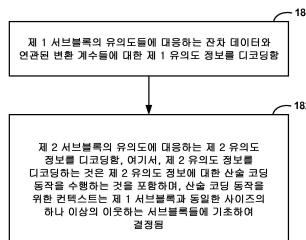
심사관 : 이상래

(54) 발명의 명칭 컨텍스트 적응 멀티-레벨 유의도 코딩을 위한 컨텍스트 도출

(57) 요 약

비디오 데이터를 코딩하기 위한 디바이스는 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 코딩하는 것으로서, 제 1 유의도 정보는 제 1 서브블록이 적어도 하나의 비-체로 계수를 포함하는지를 표시하고, 제 1 서브블록은 전체 변환 블록의 서브블록인, 상기 제 1 유의도 정보를 코딩하고; 그리고 제 2 유의도 정보를 코딩(뒷면에 계속)

대 표 도 - 도13



당하도록 구성된 비디오 코더를 포함하고, 여기서, 제 2 유의도 정보는 제 2 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 제 2 서브블록은 제 1 서브블록의 서브블록이며, 제 2 유의도 정보를 코딩하는 것은 제 2 유의도 정보에 대한 산술 코딩 동작을 수행하는 것을 포함하고, 산술 코딩 동작을 위한 컨텍스트는 제 1 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들에 기초하여 결정된다.

(52) CPC특허분류

H04N 19/122 (2015.01)

H04N 19/176 (2015.01)

명세서

청구범위

청구항 1

비디오 데이터를 디코딩하기 위한 방법으로서,

하나 이상의 프로세서들을 포함하는 디바이스에 의해, 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 (significance) 정보를 디코딩하는 단계로서, 상기 제 1 유의도 정보는 제 1 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 상기 제 1 서브블록은 전체 변환 블록의 서브블록이고, 상기 제 1 유의도 정보를 디코딩하는 단계는 상기 제 1 유의도 정보에 대한 제 1 산술 디코딩 동작을 수행하는 단계, 및 상기 제 1 서브블록과 동일한 사이즈의 우측의 이웃 서브블록 및 상기 제 1 서브블록과 동일한 사이즈의 하위의 이웃 서브블록의 유의도 정보에 기초하여 상기 제 1 산술 디코딩 동작을 위한 컨텍스트를 결정하는 단계를 포함하는, 상기 제 1 유의도 정보를 디코딩하는 단계;

상기 하나 이상의 프로세서들을 포함하는 디바이스에 의해, 제 2 유의도 정보를 디코딩하는 단계로서, 상기 제 2 유의도 정보는 제 2 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 상기 제 2 서브블록은 상기 제 1 서브블록의 서브블록이고, 상기 제 2 유의도 정보를 디코딩하는 단계는 제 1 산술 디코딩 동작을 위해 결정된 상기 컨텍스트를 이용하여 상기 제 2 유의도 정보에 대한 제 2 산술 디코딩 동작을 수행하는 단계를 포함하고, 상기 제 2 서브블록은 사이즈가 적어도 2×2 블록이고, 상기 제 1 서브블록은 상기 제 2 서브블록보다 큼, 상기 제 2 유의도 정보를 디코딩하는 단계;

상기 하나 이상의 프로세서들을 포함하는 디바이스에 의해, 제 3 유의도 정보를 디코딩하는 단계로서, 상기 제 3 유의도 정보는 상기 제 2 서브블록의 개별 계수가 비-제로 계수를 포함하는지를 표시하는, 상기 제 3 유의도 정보를 디코딩하는 단계;

상기 제 1 유의도 정보, 상기 제 2 유의도 정보, 및 상기 제 3 유의도 정보에 기초하여, 잔차 블록을 생성하는 단계;

비디오 데이터의 복원된 블록을 형성하기 위해 상기 잔차 블록을 예측 블록에 부가하는 단계; 및

상기 비디오 데이터의 복원된 블록을 출력하는 단계를 포함하는, 비디오 데이터를 디코딩하기 위한 방법.

청구항 2

제 1 항에 있어서,

상기 제 1 서브블록은 4×4 블록이고, 상기 제 2 서브블록은 2×2 서브블록인, 비디오 데이터를 디코딩하기 위한 방법.

청구항 3

제 1 항에 있어서,

무선 통신 디바이스의 수신기에서 상기 비디오 데이터를 수신하는 단계;

상기 무선 통신 디바이스의 메모리에 상기 비디오 데이터를 저장하는 단계; 및

상기 무선 통신 디바이스의 하나 이상의 프로세서들 상에서 상기 비디오 데이터를 프로세싱하는 단계를 더 포함하는, 비디오 데이터를 디코딩하기 위한 방법.

청구항 4

비디오 데이터를 디코딩하기 위한 디바이스로서,

상기 비디오 데이터와 연관된 잔차 데이터를 저장하도록 구성된 메모리; 및

상기 메모리에 커플링되고 하나 이상의 프로세서들을 포함하는 비디오 디코더를 포함하고,

상기 비디오 디코더는,

상기 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 디코딩하는 것으로서, 상기 제 1 유의도 정보는 제 1 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 상기 제 1 서브블록은 전체 변환 블록의 서브블록이고, 상기 제 1 유의도 정보를 디코딩하기 위해, 상기 비디오 디코더는 상기 제 1 유의도 정보에 대한 제 1 산술 디코딩 동작을 수행하고, 상기 제 1 서브블록과 동일한 사이즈의 우측의 이웃 서브블록 및 상기 제 1 서브블록과 동일한 사이즈의 하위의 이웃 서브블록의 유의도 정보에 기초하여 상기 제 1 산술 디코딩 동작을 위한 컨텍스트를 결정하는, 상기 제 1 유의도 정보를 디코딩하고;

제 2 유의도 정보를 디코딩하는 것으로서, 상기 제 2 유의도 정보는 제 2 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 상기 제 2 서브블록은 상기 제 1 서브블록의 서브블록이고, 상기 제 2 유의도 정보를 디코딩하기 위해, 상기 비디오 디코더는 제 1 산술 디코딩 동작을 위해 결정된 상기 컨텍스트를 이용하여 상기 제 2 유의도 정보에 대한 제 2 산술 디코딩 동작을 수행하고, 상기 제 2 서브블록은 사이즈가 적어도 2×2 블록이고, 상기 제 1 서브블록은 상기 제 2 서브블록보다 큼, 상기 제 2 유의도 정보를 디코딩하고;

제 3 유의도 정보를 디코딩하는 것으로서, 상기 제 3 유의도 정보는 상기 제 2 서브블록의 개별 계수가 비-제로 계수를 포함하는지를 표시하는, 상기 제 3 유의도 정보를 디코딩하고;

상기 제 1 유의도 정보, 상기 제 2 유의도 정보, 및 상기 제 3 유의도 정보에 기초하여, 잔차 블록을 생성하고;
비디오 데이터의 복원된 블록을 형성하기 위해 상기 잔차 블록을 예측 블록에 부가하고; 그리고

상기 비디오 데이터의 복원된 블록을 출력하도록

구성되는, 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 5

제 4 항에 있어서,

상기 제 1 서브블록은 4×4 블록이고, 상기 제 2 서브블록은 2×2 서브블록인, 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 6

제 4 항에 있어서,

상기 비디오 데이터를 디코딩하기 위한 디바이스는,

집적 회로;

마이크로 프로세서; 또는

상기 비디오 디코더를 포함하는 무선 통신 디바이스

중 적어도 하나를 포함하는, 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 7

제 4 항에 있어서,

상기 비디오 데이터를 디코딩하기 위한 디바이스는 무선 통신 디바이스를 포함하고,

상기 비디오 데이터를 디코딩하기 위한 디바이스는 상기 비디오 데이터를 수신하도록 구성된 수신기를 더 포함하는, 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 8

명령들을 저장하는 비-일시적인 컴퓨터 판독가능 저장 매체로서,

상기 명령들은, 하나 이상의 프로세서들에 의해 실행될 경우, 상기 하나 이상의 프로세서들로 하여금,

잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 디코딩하게 하는 것으로서, 상기 제 1 유의도 정보는 제 1 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 상기 제 1 서브블록은 전체 변환

블록의 서브블록이고, 상기 제 1 유의도 정보를 디코딩하기 위해, 상기 명령들은 상기 하나 이상의 프로세서들로 하여금 상기 제 1 유의도 정보에 대한 제 1 산술 디코딩 동작을 수행하게 하고, 상기 제 1 서브블록과 동일한 사이즈의 우측의 이웃 서브블록 및 상기 제 1 서브블록과 동일한 사이즈의 하위의 이웃 서브블록의 유의도 정보에 기초하여 상기 제 1 산술 디코딩 동작을 위한 컨텍스트를 결정하게 하는, 상기 제 1 유의도 정보를 디코딩하게 하고;

제 2 유의도 정보를 디코딩하게 하는 것으로서, 상기 제 2 유의도 정보는 제 2 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 상기 제 2 서브블록은 상기 제 1 서브블록의 서브블록이고, 상기 제 2 유의도 정보를 디코딩하기 위해, 상기 명령들은 상기 하나 이상의 프로세서들로 하여금 제 1 산술 디코딩 동작을 위해 결정된 상기 컨텍스트를 이용하여 상기 제 2 유의도 정보에 대한 제 2 산술 디코딩 동작을 수행하게 하고, 상기 제 2 서브블록은 사이즈가 적어도 2×2 블록이고, 상기 제 1 서브블록은 상기 제 2 서브블록보다 큰, 상기 제 2 유의도 정보를 디코딩하게 하고;

제 3 유의도 정보를 디코딩하게 하는 것으로서, 상기 제 3 유의도 정보는 상기 제 2 서브블록의 개별 계수가 비-제로 계수를 포함하는지를 표시하는, 상기 제 3 유의도 정보를 디코딩하게 하고;

상기 제 1 유의도 정보, 상기 제 2 유의도 정보, 및 상기 제 3 유의도 정보에 기초하여, 잔차 블록을 생성하게 하고;

비디오 데이터의 복원된 블록을 형성하기 위해 상기 잔차 블록을 예측 블록에 부가하게 하고; 그리고

상기 비디오 데이터의 복원된 블록을 출력하게 하는, 비-일시적인 컴퓨터 판독가능 저장 매체.

청구항 9

제 8 항에 있어서,

상기 제 1 서브블록은 4×4 블록이고, 상기 제 2 서브블록은 2×2 서브블록인, 비-일시적인 컴퓨터 판독가능 저장 매체.

청구항 10

제 8 항에 있어서,

컨텍스트 할당을 위한 패턴은 상기 제 1 서브블록 내에서의 상기 제 2 서브블록의 포지션에 기초하는, 비-일시적인 컴퓨터 판독가능 저장 매체.

청구항 11

비디오 데이터를 디코딩하기 위한 장치로서,

잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 디코딩하는 수단으로서, 상기 제 1 유의도 정보는 제 1 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 상기 제 1 서브블록은 전체 변환 블록의 서브블록이고, 상기 제 1 유의도 정보를 디코딩하는 수단은 상기 제 1 유의도 정보에 대한 제 1 산술 디코딩 동작을 수행하는 수단, 및 상기 제 1 서브블록과 동일한 사이즈의 우측의 이웃 서브블록 및 상기 제 1 서브블록과 동일한 사이즈의 하위의 이웃 서브블록의 유의도 정보에 기초하여 상기 제 1 산술 디코딩 동작을 위한 컨텍스트를 결정하는 수단을 포함하는, 상기 제 1 유의도 정보를 디코딩하는 수단;

제 2 유의도 정보를 디코딩하는 수단으로서, 상기 제 2 유의도 정보는 제 2 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 상기 제 2 서브블록은 상기 제 1 서브블록의 서브블록이고, 상기 제 2 유의도 정보를 디코딩하는 수단은 제 1 산술 디코딩 동작을 위해 결정된 상기 컨텍스트를 이용하여 상기 제 2 유의도 정보에 대한 제 2 산술 디코딩 동작을 수행하는 수단을 포함하고, 상기 제 2 서브블록은 사이즈가 적어도 2×2 블록이고, 상기 제 1 서브블록은 상기 제 2 서브블록보다 큰, 상기 제 2 유의도 정보를 디코딩하는 수단;

제 3 유의도 정보를 디코딩하는 수단으로서, 상기 제 3 유의도 정보는 상기 제 2 서브블록의 개별 계수가 비-제로 계수를 포함하는지를 표시하는, 상기 제 3 유의도 정보를 디코딩하는 수단;

상기 제 1 유의도 정보, 상기 제 2 유의도 정보, 및 상기 제 3 유의도 정보에 기초하여, 잔차 블록을 생성하는 수단;

비디오 데이터의 복원된 블록을 형성하기 위해 상기 잔차 블록을 예측 블록에 부가하는 수단; 및
상기 비디오 데이터의 복원된 블록을 출력하는 수단을 포함하는, 비디오 데이터를 디코딩하기 위한 장치.

청구항 12

제 11 항에 있어서,

컨텍스트 할당을 위한 패턴은 상기 제 1 서브블록 내에서의 상기 제 2 서브블록의 포지션에 기초하는, 비디오 데이터를 디코딩하기 위한 장치.

청구항 13

제 3 항에 있어서,

상기 무선 통신 디바이스는 전화기 핸드셋을 포함하고,

상기 무선 통신 디바이스의 수신기에서 상기 비디오 데이터를 수신하는 단계는 상기 비디오 데이터를 포함하는 신호를, 무선 통신 표준에 따라, 복조하는 단계를 포함하는, 비디오 데이터를 디코딩하기 위한 방법.

청구항 14

제 7 항에 있어서,

상기 무선 통신 디바이스는 전화기 핸드셋을 포함하고,

상기 수신기는 상기 비디오 데이터를 포함하는 신호를, 무선 통신 표준에 따라, 복조하도록 구성되는, 비디오 데이터를 디코딩하기 위한 디바이스.

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

청구항 22

삭제

청구항 23

삭제

청구항 24

삭제

청구항 25

삭제

청구항 26

삭제

청구항 27

삭제

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

발명의 설명

기술 분야

[0001] 본 출원은 2012년 9월 26일자로 출원되어 그 전체 내용이 본 명세서에 참조로 통합되는 미국 가특허출원 제 61/706,035호의 이익을 주장한다.

[0002] 본 개시는 일반적으로 비디오 코딩에 관한 것으로서, 더 상세하게는, 변환 계수들의 코딩에 관한 것이다.

배경 기술

[0003] 디지털 비디오 능력들은 디지털 텔레비전들, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인용 디지털 보조기들 (PDA들), 랩탑 또는 데스크탑 컴퓨터들, 태블릿 컴퓨터들, e-북 리더들, 디지털 카메라들, 디지털 레코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 소위 "스마트 폰들", 비디오 텔레컨퍼런싱 디바이스들, 비디오 스트리밍 디바이스들 등을 포함한, 광범위한 디바이스들에 통합될 수 있다. 디지털 비디오 디바이스들은 MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, 파트 10, 어드밴스드 비디오 코딩 (AVC), 현재 개발 중인 고 효율 비디오 코딩 (HEVC) 표준, 및 그러한 표준들의 확장들에 의해 정의된 표준들에서 설명된 기술들과 같은 비디오 압축 기술들을 구현한다. 비디오 디바이스들은 그러한 비디오 압축 기술들을 구현함으로써 디지털 비디오 정보를 더 효율적으로 송신, 수신, 인코딩, 디코딩, 및/또는 저장할 수도 있다.

[0004] 비디오 압축 기술들은 비디오 시퀀스들에 내재한 리던던시를 감소 또는 제거하기 위해 공간 (인트라-픽처) 예측 및/또는 시간 (인터-픽처) 예측을 수행한다. 블록 기반 비디오 코딩에 있어서, 비디오 슬라이스 (즉, 비디오 프레임 또는 비디오 프레임의 일부)는 비디오 블록들로 파티셔닝될 수도 있으며, 이 비디오 블록들은 또한 트리블록들, 코딩 유닛들 (CU들) 및/또는 코딩 노드들로 지칭될 수도 있다. 픽처의 인트라-코딩된 (I) 슬라이스에서의 비디오 블록들은 동일 픽처의 이웃 블록들에서의 레퍼런스 샘플들에 대한 공간 예측을 이용하여 인코딩된다. 픽처의 인터-코딩된 (P 또는 B) 슬라이스에서의 비디오 블록들은 동일 픽처의 이웃 블록들에서의 레퍼런스 샘플들에 대한 공간 예측, 또는 다른 레퍼런스 픽처들에서의 레퍼런스 샘플들에 대한 시간 예측을 이용할 수도 있다. 픽처들은 프레임들로서 지칭될 수도 있으며, 레퍼런스 픽처들은 레퍼런스 프레임들로 지칭될 수도 있다.

[0005] 공간 또는 시간 예측은 코딩될 블록에 대한 예측 블록을 발생시킨다. 잔차 데이터는 코딩될 오리지널 블록

과 예측 블록 간의 픽셀 차이들을 나타낸다. 인터-코딩된 블록은 예측 블록을 형성하는 레퍼런스 샘플들의 블록을 포인팅하는 모션 벡터, 및 코딩된 블록과 예측 블록 간의 차이를 나타내는 잔차 데이터에 따라 인코딩된다. 인트라-코딩된 블록은 인트라-코딩 모드 및 잔차 데이터에 따라 인코딩된다. 추가적인 압축을 위해, 잔차 데이터는 픽셀 도메인으로부터 변환 도메인으로 변환되어, 잔차 변환 계수들을 발생시킬 수도 있으며, 그 후, 이 잔차 변환 계수들은 양자화될 수도 있다. 2차원 어레이로 초기에 배열되는 양자화된 변환 계수들은 변환 계수들의 1차원 벡터를 생성하기 위해 스캐닝될 수도 있으며, 엔트로피 코딩이 훨씬 더 많은 압축을 달성하도록 적용될 수도 있다.

발명의 내용

과제의 해결 수단

[0006]

비디오 코딩 프로세스에 있어서, 인트라- 또는 인터-예측 코딩은, 변환 도메인에 있어서 변환 계수들에 의해 표현될 수도 있는 잔차 데이터를 발생시킨다. 변환 계수들은 변환 블록에서 제시될 수도 있다. 본 개시는 변환 블록의 변환 계수들의 시그널링 레벨들과 관련되고, 더 구체적으로, 변환 계수들의 컨텍스트 적응 멀티-레벨 유의도 코딩을 위한 컨텍스트 도출과 관련된다. 도출된 컨텍스트는, 예를 들어, 컨텍스트 적응 기반 산술 코딩 (CABAC) 과 같은 산술 코딩 프로세스를 위한 컨텍스트로서 사용될 수 있다.

[0007]

일 예에 있어서, 비디오 데이터를 코딩하기 위한 방법은 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 코딩하는 단계로서, 제 1 유의도 정보는 제 1 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 제 1 서브블록은 전체 변환 블록의 서브블록인, 상기 제 1 유의도 정보를 코딩하는 단계; 및 제 2 유의도 정보를 코딩하는 단계를 포함하고, 여기서, 제 2 유의도 정보는 제 2 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 제 2 서브블록은 제 1 서브블록의 서브블록이며, 제 2 유의도 정보를 코딩하는 단계는 제 2 유의도 정보에 대한 산술 코딩 동작을 수행하는 단계를 포함하고, 산술 코딩 동작을 위한 컨텍스트는 제 1 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들에 기초하여 결정된다.

[0008]

다른 예에 있어서, 비디오 데이터를 코딩하기 위한 디바이스는 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 코딩하는 것으로서, 제 1 유의도 정보는 제 1 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 제 1 서브블록은 전체 변환 블록의 서브블록인, 상기 제 1 유의도 정보를 코딩하고; 그리고 제 2 유의도 정보를 코딩하도록 구성된 비디오 코더를 포함하고, 여기서, 제 2 유의도 정보는 제 2 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 제 2 서브블록은 제 1 서브블록의 서브블록이며, 제 2 유의도 정보를 코딩하는 것은 제 2 유의도 정보에 대한 산술 코딩 동작을 수행하는 것을 포함하고, 산술 코딩 동작을 위한 컨텍스트는 제 1 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들에 기초하여 결정된다.

[0009]

다른 예에 있어서, 컴퓨터 판독가능 저장 매체는, 하나 이상의 프로세서들에 의해 실행될 경우, 하나 이상의 프로세서들로 하여금 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 코딩하게 하는 것으로서, 제 1 유의도 정보는 제 1 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 제 1 서브블록은 전체 변환 블록의 서브블록인, 상기 제 1 유의도 정보를 코딩하게 하고; 그리고 제 2 유의도 정보를 코딩하게 하는 명령들을 저장하고, 여기서, 제 2 유의도 정보는 제 2 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 제 2 서브블록은 제 1 서브블록의 서브블록이며, 제 2 유의도 정보를 코딩하는 것은 제 2 유의도 정보에 대한 산술 코딩 동작을 수행하는 것을 포함하고, 산술 코딩 동작을 위한 컨텍스트는 제 1 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들에 기초하여 결정된다.

[0010]

다른 예에 있어서, 비디오 데이터를 코딩하기 위한 장치는 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 코딩하는 수단으로서, 제 1 유의도 정보는 제 1 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 제 1 서브블록은 전체 변환 블록의 서브블록인, 상기 제 1 유의도 정보를 코딩하는 수단; 및 제 2 유의도 정보를 코딩하는 수단을 포함하고, 여기서, 제 2 유의도 정보는 제 2 서브블록이 적어도 하나의 비-제로 계수를 포함하는지를 표시하고, 제 2 서브블록은 제 1 서브블록의 서브블록이며, 제 2 유의도 정보를 코딩하는 것은 제 2 유의도 정보에 대한 산술 코딩 동작을 수행하는 것을 포함하고, 산술 코딩 동작을 위한 컨텍스트는 제 1 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들에 기초하여 결정된다.

[0011]

하나 이상의 예들의 상세들이 첨부 도면들 및 하기의 설명에 개시된다. 다른 특징들, 목적들, 및 이점들은 그 설명 및 도면들로부터, 그리고 청구항들로부터 명백할 것이다.

도면의 간단한 설명

[0012]

도 1 은 본 개시에서 설명된 기술들을 활용할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템을 도시한 블록 다이어그램이다.

도 2 는 변환 계수들의 4×4 계수 그룹의 예시적인 역 대각선 스캔을 도시한 개념도이다.

도 3a 내지 도 3d 는 하부 및 우측 계수 그룹 플래그에 의존하여 4×4 서브블록에서의 계수들에 대한 컨텍스트 할당을 위한 패턴들을 도시한다.

도 4 는 본 개시에서 설명된 기술들을 구현할 수도 있는 예시적인 비디오 인코더를 도시한 블록 다이어그램이다.

도 5 는 본 개시에서 설명된 기술들을 구현할 수도 있는 예시적인 비디오 디코더를 도시한 블록 다이어그램이다.

도 6 은 비디오 블록에서의 변환 계수들과 비디오 블록과 연관된 유의도 맵 간의 관계를 도시한 개념 다이어그램이다.

도 7a 내지 도 7d 는 지그재그 스캐닝 순서, 수평 스캐닝 순서, 수직 스캐닝 순서, 및 대각선 스캐닝 순서를 이용하여 스캐닝된 비디오 데이터의 블록들의 예들을 도시한 개념 다이어그램들이다.

도 8 은 변환 계수 코딩을 위해 서브블록들로 분할된 예시적인 비디오 블록을 도시한 개념 다이어그램이다.

도 9 는 역 대각선 스캐닝 순서를 이용하여 스캐닝된 비디오 블록에서의 계수들의 유의도 맵을 위한 컨텍스트 모델을 정의하는데 사용된 예시적인 5포인트 서포트 (support) 를 도시한 개념 다이어그램이다.

도 10a 및 도 10b 는 5포인트 서포트 내의 컨텍스트 의존성을 도시한 개념 다이어그램들이다.

도 11 은 비디오 블록의 각각의 영역에 대한 이웃 기반 컨텍스트 또는 포지션 기반 컨텍스트의 예시적인 할당을 도시한 개념 다이어그램이다.

도 12 는 본 개시의 기술들에 따른 예시적인 비디오 인코딩 방법을 도시한 플로우차트이다.

도 13 은 본 개시의 기술들에 따른 예시적인 비디오 디코딩 방법을 도시한 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

[0013]

비디오 코딩 프로세스에 있어서, 비디오 디코더는 비디오 데이터의 이미 디코딩된 블록에 기초하여 비디오 데이터의 현재 블록에 대한 인트라- 또는 인터-예측을 수행한다. 복원된 비디오 블록을 오리지널 비디오 데이터에 더 근접하게 매칭하게 하기 위해, 비디오 디코더는 또한, 예측된 비디오 데이터와 오리지널 비디오 데이터 간의 차이에 일반적으로 대응하는 잔차 데이터를 비디오 인코더로부터 수신한다. 따라서, 디코딩될 때, 예측된 비디오 데이터 플러스 잔차 비디오 데이터는 예측된 비디오 데이터 혼자보다 오리지널 비디오 데이터의 더 우수한 근사를 제공할 수도 있다. 하기에서 더 상세히 설명될 바와 같이, 잔차 데이터를 코딩하는데 필요한 비트들의 수를 감소하기 위해, 비디오 인코더는 잔차 데이터를 변환 및 양자화한다. 일단 양자화 및 변환되면, 잔차 데이터는 변환 도메인에서 변환 계수들에 의해 표현된다. 변환 계수들은 변환 블록에서 제시될 수도 있다.

[0014]

본 개시는 변환 블록의 변환 계수들의 시그널링 레벨들과 관련된 기술들을 설명한다. 부가적으로, 본 개시는 변환 계수들의 컨텍스트 적응 멀티-레벨 유의도 코딩을 위한 컨텍스트 도출에 대한 기술들을 설명한다. 도출된 컨텍스트는, 예를 들어, 컨텍스트 적응 기반 산술 코딩 (CABAC) 과 같은 산술 코딩 프로세스를 위한 컨텍스트로서 사용될 수 있다. 용어 '레벨' 은 본 개시에서 2개의 상이한 방식들로 사용됨이 주목되어야 한다. 용어 '레벨' 은, 일부 예들에 있어서, 계수의 값 또는 절대값을 지칭하도록 사용될 수도 있다. 하지만, 용어 '레벨' 은 또한, 계위 구조 내의 상이한 위치들을 지칭하도록 사용될 수도 있다. 예를 들어, 전체 변환 블록은 서브블록들의 제 1 세트로 분할될 수도 있고, 서브블록들의 제 1 세트의 각각의 서브블록은 서브블록 레벨들의 제 2 세트로 더 분할될 수도 있으며, 제 2 서브블록들 각각은 개별 계수들을 포함할 수도 있다. 변환 블록 계위의 이러한 상이한 스테이지들은, 예를 들어, 전체 변환 블록 레벨, 제 1 서브블록 레벨, 제 2 서브블록 레벨, 및 개별 계수 레벨로 지칭될 수도 있다.

[0015]

유의도 코딩은 일반적으로, 변환 계수 또는 변환 계수들의 그룹이 제로 이외의 값을 갖는 적어도 하나의 변환

계수를 포함하는지 여부를 나타내는 정보를 코딩하는 것을 지칭한다. 변환 계수는, 제로 이외의 값 (또한 레벨로서도 지칭됨) 을 갖는다면 유의한 것으로 고려된다. 변환 계수들의 블록은 적어도 하나의 유의 계수를 포함하면 유의한 것으로 고려된다.

[0016] 코딩되는 비디오 데이터의 특성들에 의존하여, 비디오 인코더들은 변환 블록들을 32×32 , 16×16 , 및 8×8 과 같은 다양한 사이즈들로 코딩할 수도 있다. 또한, 추후 비디오 코딩 표준들, 또는 현재 비디오 코딩 표준들의 추후 확장들은 64×64 또는 128×128 과 같은 훨씬 더 큰 변환 사이즈들을 통합할 수도 있음이 고려된다.

변환 블록의 정보를 더 효율적으로 코딩하기 위해, 변환 블록은 전체 변환 블록보다 더 작은 서브블록들로 분할될 수도 있다. 예를 들어, 16×16 변환 블록은 4×4 서브블록들 및/또는 2×2 서브블록들로 분할될 수도 있다. 본 개시는 멀티-레벨 유의도 코딩을 도입한다. 따라서, 본 개시의 기술들에 따르면, 변환 블록을 코딩할 경우, 제 1 레벨 (예를 들어, 전체 변환 블록 레벨) 에서, 비디오 인코더는, 변환 블록이 적어도 하나의 유의 계수 (즉, 적어도 하나의 비-제로 계수) 를 포함하는지를 표시하기 위한 제 1 유의도 정보 (예를 들어, 코딩된 블록 플래그) 를 코딩할 수 있다. 전체 변환 블록이 적어도 하나의 유의 계수를 포함하면, 제 2 레벨 (예를 들어, 제 1 서브블록 레벨) 에 대해, 비디오 인코더는, 서브블록이 유의 계수를 포함하는지를 표시하는 제 2 유의도 정보를 각각의 서브블록에 대해 코딩할 수 있다. 유의 계수를 포함하는 각각의 서브블록에 대해, 비디오 코더는, 서브블록의 서브블록이 유의 계수를 포함하는지를 표시하는 제 3 레벨 (즉, 제 2 서브블록 레벨) 유의도 정보를 코딩할 수 있다. 이러한 멀티-레벨 유의도 코딩은 최저 레벨 (예를 들어, 개별 계수 레벨) 까지 계속될 수 있다.

[0017] 본 개시의 기술들에 따르면, 하위 레벨들에 대해 사용된 서브블록들의 사이즈는 변환 블록의 사이즈에 독립적일 수도 있다. 일 예로서, 제 2 레벨에 대한 서브블록 사이즈는, 제 1 레벨 변환 블록이 32×32 이든 또는 그 보다 큰 어떤 것이든지에 무관하게, 16×16 일 수도 있다. 다른 예로서, 제 1 서브블록 레벨에 대한 서브블록 사이즈는, 전체 변환 블록이 16×16 이든 또는 32×32 이든지에 무관하게, 항상 8×8 일 수도 있다. 본 개시에 있어서, 용어 '서브블록' 은 일반적으로, 더 큰 블록의 부분인 더 작은 블록을 지칭하도록 의도된다.

예를 들어, 32×32 블록은 4개의 16×16 서브블록들, 16개의 8×8 서브블록들로 분할될 수도 있거나, 또는 일부 다른 사이즈의 블록들로 분할될 수도 있다. 서브블록들은 더 작은 서브블록들로 더 분할될 수도 있다.

일 예로서, 32×32 블록은 4개의 16×16 서브블록들로 분할될 수도 있지만, 16×16 서브블록들 각각은 4개의 8×8 서브블록들 또는 16개의 4×4 서브블록들로 더 분할될 수도 있다.

[0018] 각각의 서브블록 레벨에 대해, 비디오 코더는 컨텍스트에 기초하여 유의도 정보를 산술적으로 코딩할 수 있다.

본 개시의 기술들에 따르면, 특정 레벨의 유의도 정보에 대한 컨텍스트는 동일 레벨의 이미 코딩된 서브블록들뿐 아니라 더 높은 레벨의 이미 코딩된 블록들 또는 서브블록들 양자 모두에 기초할 수 있다.

[0019] 도 1 은 본 개시에서 설명된 변환 블록 코딩 기술들을 활용할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (10) 을 도시한 블록 다이어그램이다. 도 1 에 도시된 바와 같이, 시스템 (10) 은, 목적지 디바이스 (14) 에 의해 더 나중 시간에 디코딩될 인코딩된 비디오 데이터를 생성하는 소스 디바이스 (12) 를 포함한다.

소스 디바이스 (12) 및 목적지 디바이스 (14) 는 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 태블릿 컴퓨터들, 셋톱 박스들, 소위 "스마트" 폰들과 같은 전화기 핸드셋들, 소위 "스마트" 패드들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들, 비디오 스트리밍 디바이스 등 을 포함한 광범위한 디바이스들 중 임의의 디바이스를 포함할 수도 있다. 일부 경우들에 있어서, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 무선 통신을 위해 장비될 수도 있다.

[0020] 목적지 디바이스 (14) 는 디코딩될 인코딩된 비디오 데이터를 링크 (16) 를 통해 수신할 수도 있다. 링크 (16) 는 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 이동 가능한 임의의 타입의 매체 또는 디바이스를 포함할 수도 있다. 일 예에 있어서, 링크 (16) 는, 소스 디바이스 (12) 로 하여금 인코딩된 비디오 데이터를 직접 목적지 디바이스 (14) 로 실시간으로 송신할 수 있게 하는 통신 매체를 포함할 수도 있다. 인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라 변조되고, 목적지 디바이스 (14) 로 송신될 수도 있다. 통신 매체는 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적인 송신 라인들과 같은 임의의 무선 또는 유선 통신 매체를 포함할 수도 있다. 통신 매체는 로컬 영역 네트워크, 광역 네트워크, 또는 인터넷과 같은 글로벌 네트워크와 같은 패킷 기반 네트워크의 부분을 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로의 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0021] 대안적으로, 인코딩된 데이터는 출력 인터페이스 (22) 로부터 저장 디바이스 (32) 로 출력될 수도 있다. 유

사하게, 인코딩된 데이터는 입력 인터페이스에 의해 저장 디바이스 (32)로부터 액세스될 수도 있다. 저장 디바이스 (32)는 하드 드라이브, 블루-레이 디스크들, DVD들, CD-ROM들, 플래시 메모리, 휴발성 또는 비휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체들과 같은 다양한 분산된 또는 국부적으로 액세스된 데이터 저장 매체들 중 임의의 데이터 저장 매체를 포함할 수도 있다. 추가의 예에 있어서, 저장 디바이스 (32)는, 소스 디바이스 (12)에 의해 생성된 인코딩된 비디오를 유지할 수도 있는 파일 서버 또는 다른 중간 저장 디바이스에 대응할 수도 있다. 목적지 디바이스 (14)는 저장 디바이스 (32)로부터의 저장된 비디오 데이터에 스트리밍 또는 다운로드를 통해 액세스할 수도 있다. 파일 서버는, 인코딩된 비디오 데이터를 저장하고 그 인코딩된 비디오 데이터를 목적지 디바이스 (14)로 송신하는 것이 가능한 임의의 타입의 서버일 수도 있다. 예시적인 파일 서버들은 웹 서버 (예를 들어, 웹 사이트용), FTP 서버, 네트워크 접속형 저장 (NAS) 디바이스들, 또는 로컬 디스크 드라이브를 포함한다. 목적지 디바이스 (14)는 인코딩된 비디오 데이터에, 인터넷 커넥션을 포함한 임의의 표준 데이터 커넥션을 통해 액세스할 수도 있다. 이는 파일 서버 상에 저장되는 인코딩된 비디오 데이터에 액세스하기에 적합한 무선 채널 (예를 들어, Wi-Fi 커넥션), 유선 커넥션 (예를 들어, DSL, 케이블 모뎀 등), 또는 이를 양자의 조합을 포함할 수도 있다. 인코딩된 비디오 데이터의 저장 디바이스 (32)로부터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이를 양자의 조합일 수도 있다.

[0022] 본 개시의 기술들은 무선 어플리케이션들 또는 설정들에 반드시 한정되는 것은 아니다. 그 기술들은, 공중 경유 (over-the-air) 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 예를 들어, 인터넷을 통한 스트리밍 비디오 송신들, 데이터 저장 매체 상의 저장을 위한 디지털 비디오의 인코딩, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 어플리케이션들과 같은 다양한 멀티미디어 어플리케이션들 중 임의의 어플리케이션들의 서포트로 비디오 코딩에 적용될 수도 있다. 일부 예들에 있어서, 시스템 (10)은 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 및/또는 비디오 전화와 같은 어플리케이션들을 서포트하기 위해 일방향 또는 양방향 비디오 송신을 서포트하도록 구성될 수도 있다.

[0023] 도 1의 예에 있어서, 소스 디바이스 (12)는 비디오 소스 (18), 비디오 인코더 (20), 및 출력 인터페이스 (22)를 포함한다. 일부 경우들에 있어서, 출력 인터페이스 (22)는 변조기/복조기 (모뎀) 및/또는 송신기를 포함할 수도 있다. 소스 디바이스 (12)에 있어서, 비디오 소스 (18)는 비디오 캡쳐 디바이스, 예를 들어, 비디오 카메라, 이전에 캡쳐된 비디오를 포함하는 비디오 아카이브, 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 퍼드 인터페이스, 및/또는 컴퓨터 그래픽스 데이터를 소스 비디오로서 생성하기 위한 컴퓨터 그래픽스 시스템과 같은 소스, 또는 그러한 소스들의 조합을 포함할 수도 있다. 일 예로서, 비디오 소스 (18)가 비디오 카메라이면, 소스 디바이스 (12) 및 목적지 디바이스 (14)는 소위 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 하지만, 본 개시에서 설명된 기술들은 일반적으로 비디오 코딩에 적용가능할 수도 있으며, 무선 및/또는 유선 어플리케이션들에 적용될 수도 있다.

[0024] 캡쳐되거나 사전-캡쳐되거나 또는 컴퓨터 생성된 비디오는 비디오 인코더 (20)에 의해 인코딩될 수도 있다. 인코딩된 비디오 데이터는 소스 디바이스 (12)의 출력 인터페이스 (22)를 통해 목적지 디바이스 (14)로 직접 송신될 수도 있다. 인코딩된 비디오 데이터는 또한 (또한 대안적으로), 디코딩 및/또는 플레이백을 위한 목적지 디바이스 (14) 또는 다른 디바이스들에 의한 더 나중의 액세스를 위해 저장 디바이스 (32) 상으로 저장될 수도 있다.

[0025] 목적지 디바이스 (14)는 입력 인터페이스 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32)를 포함한다. 일부 경우들에 있어서, 입력 인터페이스 (28)는 수신기 및/또는 모뎀을 포함할 수도 있다. 목적지 디바이스 (14)의 입력 인터페이스 (28)는 인코딩된 비디오 데이터를 링크 (16) 상으로 수신한다. 링크 (16) 상으로 통신되는 또는 저장 디바이스 (32) 상에 제공되는 인코딩된 비디오 데이터는, 비디오 데이터를 디코딩함에 있어서 비디오 디코더 (30)와 같은 비디오 디코더에 의한 사용을 위해 비디오 인코더 (20)에 의해 생성된 다양한 선택스 엘리먼트들을 포함할 수도 있다. 그러한 선택스 엘리먼트들에는, 통신 매체 상으로 송신되거나 저장 매체 상에 저장되거나 또는 파일 서버에 저장되는 인코딩된 비디오 데이터가 포함될 수도 있다.

[0026] 디스플레이 디바이스 (32)는 목적지 디바이스 (14)와 통합되거나 그 외부에 있을 수도 있다. 일부 예들에 있어서, 목적지 디바이스 (14)는 통합된 디스플레이 디바이스를 포함할 수도 있고, 또한, 외부 디스플레이 디바이스와 인터페이싱하도록 구성될 수도 있다. 다른 예들에 있어서, 목적지 디바이스 (14)는 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32)는 디코딩된 비디오 데이터를 사용자에게 디스플레이하며, 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 다

른 타입의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 디바이스를 포함할 수도 있다.

[0027] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 현재 개발 중인 고 효율 비디오 코딩 (HEVC) 표준과 같은 비디오 압축 표준에 따라 동작할 수도 있고, HEVC 테스트 모델 (HM) 에 부합할 수도 있다. "HEVC 워킹 드래프트 8" 또는 "WD8" 로서 지정되는 HEVC 표준의 최근 드래프트는 문헌 JCTVC-J1003, 즉, Bross 등의 "High efficiency video coding (HEVC) text specification draft 8", ITU-T SG16 WP3 및 ISO/IEC JTC1/SC29/WG11 의 JCT-VC (Joint Collaborative Team on Video Coding), 제10차 회의: 스톡홀름, 스웨덴, 7월 11일-20일,에 기술되어 있다. WD8 은 본 명세서에 의해 전부 참조로 통합된다. "HEVC 워킹 드래프트 10" 또는 "WD10" 으로서 지정되는 HEVC 표준의 다른 최근 드래프트는 문헌 JCTVC-L1003v34, 즉, Bross 등의 "High efficiency video coding (HEVC) text specification draft 10 (for FDIS & Last Call)", ITU-T SG16 WP3 및 ISO/IEC JTC1/SC29/WG11 의 JCT-VC (Joint Collaborative Team on Video Coding), 제12차 회의: 제네바, 스위스, 2013년 1월 14일-23일,에 기술되어 있다. WD10 은 본 명세서에 의해 전부 참조로 통합된다.

[0028] 대안적으로 또는 부가적으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 MPEG-4, 파트 10, 어드밴스드 비디오 코딩 (AVC) 으로서 대안적으로 지정되는 ITU-T H.264 표준과 같은 다른 전매특허 또는 산업 표준들, 또는 그러한 표준들의 확장들에 따라 동작할 수도 있다. 하지만, 본 개시의 기술들은 임의의 특정 코딩 표준에 한정되지 않는다. 비디오 압축 표준들의 다른 예들은 MPEG-2 및 ITU-T H.263 을 포함한다.

[0029] 비록 도 1 에 도시되지는 않지만, 일부 양태들에 있어서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 오디오 인코더 및 디코더와 각각 통합될 수도 있으며, 오디오 및 비디오 양자의 인코딩을 공통 데이터 스트림 또는 별개의 데이터 스트림들로 처리하기 위해 적절한 MUX-DEMUX 유닛들 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능하다면, 일부 예들에 있어서, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜, 또는 사용자 데이터그램 프로토콜 (UDP) 과 같은 다른 프로토콜들에 부합할 수도 있다.

[0030] 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSP들), 주문형 집적회로들 (ASIC들), 필드 프로그래밍가능 게이트 어레이들 (FPGA들), 별도의 로직, 소프트웨어, 하드웨어, 펌웨어 또는 이들의 임의의 조합들과 같은 다양한 적합한 인코더 회로 중 임의의 회로로서 구현될 수도 있다. 기술들이 부분적으로 소프트웨어로 구현될 경우, 디바이스는 적합한 비-일시적인 컴퓨터 판독가능 매체에 소프트웨어에 대한 명령들을 저장하고, 본 개시의 기술들을 수행하기 위해 하나 이상의 프로세서들을 사용하는 하드웨어로 명령들을 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이들 중 어느 하나는 개별 디바이스에 있어서 결합된 인코더/디코더 (CODEC) 의 부분으로서 통합될 수도 있다.

[0031] 일반적으로, HM 의 작업 모델은, 비디오 프레임 또는 픽처가 루마 및 크로마 샘플들 양자를 포함하는 트리블록들 또는 최대 코딩 유닛들 (LCU) 의 시퀀스로 분할될 수도 있음을 기술한다. 트리블록은 H.264 표준의 매크로블록과 유사한 목적을 갖는다. 슬라이스는 코딩 순서에 있어서의 다수의 연속적인 트리블록들을 포함한다. 비디오 프레임 또는 픽처는 하나 이상의 슬라이스들로 파티셔닝될 수도 있다. 각각의 트리블록은 쿼드트리에 따라 코딩 유닛들 (CU들) 로 분할될 수도 있다. 예를 들어, 쿼드트리의 루트 노드로서 트리블록은 4개의 자식 노드들로 분할될 수도 있고, 각각의 자식 노드는 결국 부모 노드일 수도 있으며 또 다른 4개의 자식 노드들로 분할될 수도 있다. 쿼드트리의 리프 노드로서 최종의 미분할된 자식 노드는 코딩 노드, 즉, 코딩된 비디오 블록을 포함한다. 코딩된 비트스트림과 연관된 신택스 데이터는 트리블록이 분할될 수도 있는 최대 횟수를 정의할 수도 있으며, 또한 코딩 노드들의 최소 사이즈를 정의할 수도 있다.

[0032] CU 는 코딩 노드, 그리고 코딩 노드와 연관된 예측 유닛들 (PU들) 및 변환 유닛들 (TU들) 을 포함한다. CU 의 사이즈는 코딩 노드의 사이즈에 대응하고, 형상이 정방형이어야 한다. CU 의 사이즈는 8×8 픽셀들로부터, 최대 64×64 픽셀들 이상을 갖는 트리블록의 사이즈까지 이를 수도 있다. 각각의 CU 는 하나 이상의 PU들 및 하나 이상의 TU들을 포함할 수도 있다. CU 와 연관된 신택스 데이터는, 예를 들어, 하나 이상의 PU들로의 CU 의 파티셔닝을 기술할 수도 있다. 파티셔닝 모드들은 CU 가 스kip되거나 직접 모드 인코딩되거나, 인트라-예측 모드 인코딩되거나, 또는 인터-예측 모드 인코딩되는지 여부 간에 상이할 수도 있다. PU들은 형상이 비-정방형이도록 파티셔닝될 수도 있다. CU 와 연관된 신택스 데이터는 또한, 예를 들어, 쿼드트리에 따라 하나 이상의 TU들로의 CU 의 파티셔닝을 기술할 수도 있다. TU 는 형상이 정방형이거나 비-정방형일 수 있다.

[0033] HEVC 표준은 TU들에 따른 변환들에 대해 허용하며, 이는 상이한 CU들에 대해 상이할 수도 있다. TU들은 통상적으로, 파티셔닝된 LCU 에 대해 정의된 소정의 CU 내에서의 PU들의 사이즈에 기초하여 사이징되지만, 이것이

항상 그 경우인 것은 아닐 수도 있다. TU들은 통상적으로 PU들과 동일한 사이즈이거나 그 보다 더 작다. 일부 예들에 있어서, CU에 대응하는 잔차 샘플들은 "잔차 쿼드트리 (RQT)"로서 공지된 쿼드트리 구조를 이용하여 더 작은 유닛들로 세분될 수도 있다. RQT의 리프 노드들은 변환 유닛들 (TU들)로서 지정될 수도 있다. TU들과 연관된 픽셀 차이 값들은 변환 계수들을 생성하도록 변환될 수도 있으며, 이 변환 계수들은 양자화될 수도 있다.

[0034] 일반적으로, PU는 예측 프로세스와 관련된 데이터를 포함한다. 예를 들어, PU가 인트라-모드 인코딩될 경우, PU는 그 PU에 대한 인트라-예측 모드를 기술하는 데이터를 포함할 수도 있다. 다른 예로서, PU가 인터-모드 인코딩될 경우, PU는 그 PU에 대한 모션 벡터를 정의하는 데이터를 포함할 수도 있다. PU에 대한 모션 벡터를 정의하는 데이터는 예를 들어, 모션 벡터의 수평 컴포넌트, 모션 벡터의 수직 컴포넌트, 모션 벡터에 대한 분해능 (예를 들어, 1/4 픽셀 정밀도 또는 1/8 픽셀 정밀도), 모션 벡터가 포인팅하는 레퍼런스 픽처, 및/또는 모션 벡터에 대한 레퍼런스 픽처 리스트 (예를 들어, 리스트 0, 리스트 1, 또는 리스트 C)를 기술할 수도 있다.

[0035] 일반적으로, TU는 변환 및 양자화 프로세스들을 위해 사용된다. 하나 이상의 PU들을 갖는 소정의 CU는 또한 하나 이상의 TU들을 포함할 수도 있다. 예측 이후, 비디오 인코더 (20)는 PU에 대응하는 잔차 값들을 계산할 수도 있다. 잔차 값들은, 엔트로피 코딩을 위한 직렬화된 변환 계수들을 생성하기 위해 변환 계수들로 변환되고 양자화되고 TU들을 이용하여 스캐닝될 수도 있는 픽셀 차이 값들을 포함한다. 상기 도입된 바와 같이, 본 개시는 비디오 인코더 (20)로부터 비디오 디코더 (30)로 TU들을 코딩 및 시그널링하기 위한 기술들을 설명한다. 본 개시는 CU, LCU, PU, TU, 또는 다른 타입들의 비디오 블록들 중 임의의 것을 일반적으로 지칭하기 위해 용어 "비디오 블록"을 사용한다.

[0036] 비디오 시퀀스는 통상적으로, 비디오 프레임들 또는 픽처들의 시리즈를 포함한다. 픽처들의 그룹 (GOP)은 일반적으로 비디오 픽처들의 하나 이상의 시리즈를 포함한다. GOP는, 그 GOP에 포함된 다수의 픽처들을 기술하는 신팩스 데이터를 GOP의 헤더, 픽처들의 하나 이상의 헤더, 또는 다른 곳에 포함할 수도 있다. 픽처의 각각의 슬라이스는 개별 슬라이스에 대한 인코딩 모드를 기술하는 슬라이스 신팩스 데이터를 포함할 수도 있다. 비디오 인코더 (20)는 통상적으로, 비디오 데이터를 인코딩하기 위해 개별적인 비디오 슬라이스들 내의 비디오 블록들에 대해 동작한다. 비디오 블록은 CU 내의 코딩 노드에 대응할 수도 있다. 비디오 블록들은 고정 또는 가변 사이즈들을 가질 수도 있으며, 특정 코딩 표준에 따라 사이즈가 상이할 수도 있다.

[0037] 일 예로서, HM은 다양한 PU 사이즈들에서의 예측을 서포트한다. 특정 CU의 사이즈가 $2N \times 2N$ 이라고 가정하면, HM은 $2N \times 2N$ 또는 $N \times N$ 의 PU 사이즈들에서 인트라-예측을 서포트하고, 그리고 $2N \times 2N$, $2N \times N$, $N \times 2N$, 또는 $N \times N$ 의 대칭적인 PU 사이즈들에서 인터-예측을 서포트한다. HM은 또한, $2N \times nU$, $2N \times nD$, $nL \times 2N$, 및 $nR \times 2N$ 의 PU 사이즈들에서 인터-예측을 위한 비대칭 파티셔닝을 서포트한다. 비대칭 파티셔닝에 있어서, CU의 일 방향은 파티셔닝되지 않지만 다른 방향은 25% 및 75%로 파티셔닝된다. 25% 파티션에 대응하는 CU의 부분은 "n" 이후에 "상", "하", "좌", 또는 "우"의 표시에 의해 표시된다. 따라서, 예를 들어, " $2N \times nU$ "는, 상부에서의 $2N \times 0.5N$ PU 및 하부에서의 $2N \times 1.5N$ PU로 수평으로 파티셔닝된 $2N \times 2N$ CU를 지칭한다.

[0038] 본 개시에 있어서, " $N \times N$ " 및 "N 바이 N"은 수직 및 수평 치수들의 관점에서의 비디오 블록의 픽셀 치수들, 예를 들어, 16×16 픽셀들 또는 16 바이 16 픽셀들을 지칭하도록 상호대체가능하게 사용될 수도 있다. 일반적으로, 16×16 블록은 수직 방향에서 16개의 픽셀들 ($y = 16$) 및 수평 방향에서 16개의 픽셀들 ($x = 16$)을 가질 것이다. 유사하게, $N \times N$ 블록은 일반적으로, 수직 방향에서 N개의 픽셀들 및 수평 방향에서 N개의 픽셀들을 가지며, 여기서, N은 음이 아닌 정수 값을 나타낸다. 블록에 있어서의 픽셀들은 로우들 및 컬럼들에서 배열될 수도 있다. 더욱이, 블록들은 반드시 수평 방향에서 수직 방향에서와 동일한 수의 픽셀들을 가질 필요는 없다. 예를 들어, 블록들은 $N \times M$ 픽셀들을 포함할 수도 있으며, 여기서, M은 반드시 N과 동일할 필요는 없다.

[0039] CU의 PU들을 이용한 인트라-예측 또는 인터-예측 코딩 이후, 비디오 인코더 (20)는 CU의 TU들에 대한 잔차 데이터를 계산할 수도 있다. PU들은 공간 도메인 (픽셀 도메인으로서도 또한 지정됨)에 있어서의 픽셀 데이터를 포함할 수도 있고, TU들은 변환, 예를 들어, 이산 코사인 변환 (DCT), 정수 변환, 웨이블릿 변환, 또는 잔차 비디오 데이터에 대한 개념적으로 유사한 변환의 적용 이후에 변환 도메인에 있어서 계수들을 포함할 수도 있다. 잔차 데이터는 인코딩되지 않은 픽처의 픽셀들과 PU들에 대응하는 예측 값들 간의 픽셀 차이들에 대응할 수도 있다. 비디오 인코더 (20)는 CU에 대한 잔차 데이터를 포함하는 TU들을 형성하고, 그 후, TU들을 변환하여 CU에 대한 변환 계수들을 생성할 수도 있다.

- [0040] 변환 계수들을 생성하기 위한 임의의 변환들 이후, 비디오 인코더 (20)는 변환 계수들의 양자화를 수행할 수도 있다. 양자화는 일반적으로, 변환 계수들이 그 계수들을 나타내는데 사용되는 데이터의 양을 가능하게는 감소시키도록 양자화되어 추가 압축을 제공하는 프로세스를 지칭한다. 양자화 프로세스는 그 계수들의 일부 또는 그 모두와 연관되는 비트 깊이를 감소시킬 수도 있다. 예를 들어, n 비트 값은 양자화 동안 m 비트 값으로 라운드-다운될 수도 있으며, 여기서, n 은 m 보다 크다.
- [0041] 상기 도입된 바와 같이 그리고 하기에서 더 상세히 설명될 바와 같이, 비디오 인코더 (20)는 본 개시에서 설명된 바와 같은 컨텍스트 적응 멀티-레벨 유의도 코딩을 이용하여 TU를 비디오 디코더 (30)로 시그널링할 수도 있다. 일부 예들에 있어서, 비디오 인코더 (20)는 양자화된 변환 계수들을 스캔하기 위한 미리정의된 스캔 순서를 활용하여, 엔트로피 인코딩될 수 있는 직렬화된 벡터를 생성할 수도 있다. 다른 예들에 있어서, 비디오 인코더 (20)는 적응적 스캔을 수행할 수도 있다. 1차원 벡터를 형성하기 위해 양자화된 변환 계수들을 스캔한 이후, 비디오 인코더 (20)는, 예를 들어, 컨텍스트 적응 가변 길이 코딩 (CAVLC), 컨텍스트 적응 이진 산술 코딩 (CABAC), 신택스 기반 컨텍스트 적응 이진 산술 코딩 (SBAC), 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩, 또는 다른 엔트로피 인코딩 방법에 따라 1차원 벡터를 엔트로피 인코딩할 수도 있다. 비디오 인코더 (20)는 또한, 비디오 데이터를 디코딩함에 있어서 비디오 디코더 (30)에 의한 사용을 위해 인코딩된 비디오 데이터와 연관된 신택스 엘리먼트들을 엔트로피 인코딩할 수도 있다.
- [0042] 일반적으로, CABAC를 이용하여 데이터 심볼들을 엔트로피 코딩하는 것은 다음의 단계들 중 하나 이상을 수반한다:
- [0043] (1) 이진화: 코딩될 심볼이 비-이진 값이면, 소위 "빈"들의 시퀀스로 매핑된다. 각각의 빈은 "0" 또는 "1"의 값을 가질 수 있다.
- [0044] (2) 컨텍스트 할당: (정규 모드에서) 각각의 빈은 컨텍스트에 할당된다. 컨텍스트 모델은, 소정의 빈에 대한 컨텍스트가 이전에 인코딩된 심볼들의 값을 또는 빈 번호와 같이 그 빈에 대해 이용가능한 정보에 기초하여 어떻게 계산되는지를 결정한다.
- [0045] (3) 빈 인코딩: 빈들은 산술 인코더로 인코딩된다. 빈을 인코딩하기 위해, 산술 인코더는 입력으로서, 빈의 값의 확률, 즉, 빈의 값이 "0"과 동일할 확률 및 빈의 값이 "1"과 동일할 확률을 요구한다. 각각의 컨텍스트의 (추정된) 확률은 "컨텍스트 상태"로 지칭되는 정수 값에 의해 표현된다. 각각의 컨텍스트는 상태들의 범위를 가지며, 따라서, 상태들 (즉, 추정된 확률들)의 범위는 하나의 컨텍스트에 할당된 빈들에 대해 동일하고 컨텍스트들 간에는 상이하다.
- [0046] (4) 상태 업데이트: 선택된 컨텍스트에 대한 확률 (상태)은 빈의 실제 코딩된 값에 기초하여 업데이트된다 (예를 들어, 빈 값이 "1"이었으면, "1들"의 확률은 증가함).
- [0047] PIPE는 산술 코딩의 원리들과 유사한 원리들을 이용하고, 따라서, 본 개시의 기술들을 또한 잠재적으로 활용할 수 있음이 주목되어야 한다.
- [0048] CABAC를 수행하기 위해, 비디오 인코더 (20)는 컨텍스트 모델 내의 컨텍스트를, 송신될 심볼에 할당할 수도 있다. 컨텍스트는, 예를 들어, 심볼의 이웃하는 값들이 제로가 아닌지 여부와 관련될 수도 있다.
- [0049] HEVC WD8에 있어서, 8×8 , 16×16 , 및 32×32 변환 블록들에 대해, 비디오 인코더 (20)는 4×4 서브블록 스캔을 이용한다. 서브블록들은 우상측-좌하측 스캔을 이용하여 역 방향으로 스캐닝된다 (예를 들어, 고 주파수들로부터 저 주파수들로). 서브블록 내에서, 변환 계수들은 또한 우상측-좌하측 스캔을 이용하여 역방향으로 스캐닝된다. 각각의 4×4 서브블록은 5개의 코딩 패스들, 즉, 유의도, 1초파의 레벨, 2초파의 레벨, 부호, 및 나머지 2초파의 계수 레벨로 코딩된다.
- [0050] HEVC 표준을 위한 일부 제안들에 있어서, 비디오 인코더 (20)는 계수들을 청크들 또는 서브세트들로 그룹핑하며, 이 청크들 또는 서브세트들은 또한 종종 계수 그룹 (CG)으로 지칭된다. 비디오 인코더 (20)는 각각의 서브세트에 대한 변환 계수들의 유의도 맵 및 레벨 정보 (절대값 및 부호)를 코딩한다. 일 예에 있어서, 더 큰 변환 블록 내에서의 변환 계수들의 4×4 서브블록 (또는 서브세트)이 서브세트로서 처리된다. 다음의 심볼들은, 서브세트 내에서의 계수 레벨 정보를 나타내기 위해 비디오 인코더 (20)로부터 비디오 디코더 (30)로 코딩 및 시그널링된다. 일 예에 있어서, 비디오 인코더 (20)는 심볼들 모두를 역 스캔 순서로 인코딩한다. 다음의 심볼들은 "플래그들"로서 지칭될 수도 있음이 주목되어야 한다. 본 개시에서 논의된 "플래그들" 중 임의의 플래그는 이진 심볼로 한정될 필요는 없지만, 또한 다중의 비트 신택스 엘리먼트들로서

구현될 수도 있음이 주목되어야 한다.

- [0051] **significant_coeff_flag (abbr.sigMapFlag):** 이 플래그는 CG에서의 각각의 계수의 유의도를 표시한다. 1 이상의 절대값을 갖는 계수는 유의한 것으로 고려된다. 일 예로서, 0의 sigMapFlag 값은 계수가 유의하지 않음을 표시하지만, 1의 값은 계수가 유의함을 표시한다. 이 플래그는 일반적으로, 유의도 플래그로서 지칭될 수도 있다.
- [0052] **coeff_abs_level_greater1_flag (abbr.gr1Flag):** 이 플래그는 계수의 절대값이 임의의 비-제로 계수들 (즉, 1로서의 sigMapFlag를 갖는 계수들)에 대해 1초과인지 여부를 표시한다. 일 예로서, 0의 gr1Flag 값은 계수가 1초과의 절대값을 갖지 않음을 표시하지만, gr1Flag에 대한 1의 값은 계수가 1초과의 절대값을 가짐을 표시한다. 이 플래그는 일반적으로, 1초과 플래그로서 지칭될 수도 있다.
- [0053] **coeff_abs_level_greater2_flag (abbr.gr2Flag):** 이 플래그는 계수의 절대값이 1초과의 절대값을 갖는 임의의 계수들 (즉, 1로서의 gr1Flag를 갖는 계수들)에 대해 2초과인지 여부를 표시한다. 일 예로서, 0의 gr2Flag 값은 계수가 2초과의 절대값을 갖지 않음을 표시하지만, gr2Flag에 대한 1의 값은 계수가 2초과의 절대값을 가짐을 표시한다. 이 플래그는 일반적으로, 2초과 플래그로서 지칭될 수도 있다.
- [0054] **coeff_sign_flag (abbr.signFlag):** 이 플래그는 임의의 비-제로 계수들 (즉, 1로서의 sigMapFlag를 갖는 계수들)에 대한 부호 정보를 표시한다. 예를 들어, 이 플래그에 대한 제로는 양의 부호를 표시하지만 1은 음의 부호를 표시한다.
- [0055] **coeff_abs_level_remaining (abbr.levelRem):** 이 선택스 엘리먼트는 나머지 계수들의 절대 레벨 값을 표시한다. 예를 들어, 이 플래그에 대해, 계수의 절대값 마이너스 3이 2초과의 절대값을 갖는 각각의 계수 (즉, 1로서의 gr2Flag를 갖는 계수들)에 대해 코딩된다 (abs(레벨)-3).
- [0056] 도 2는 4×4 블록 (100)에서의 양자화된 계수들의 예를 도시한다. 블록 (100)은 4×4 변환 블록일 수도 있거나, 또는 8×8 , 16×16 또는 32×32 변환 블록에서의 4×4 서브블록 (서브세트)일 수도 있다. 역 스캔 순서로 스캐닝된, 도 2에 도시된 계수들에 대한 인코딩된 심볼들이 표 1에 요약된다. 표 1에 있어서, scan_pos는 도 2에 도시된 역 대각선 스캔을 따른 계수의 포지션을 지칭한다. Scan_pos 15는 스캐닝된 제 1 계수이고, 블록 (100)의 우하측 코너에 위치된다. Scan_pos 15에서의 양자화된 계수는 0의 절대값을 갖는다. Scan_pos 0은 스캐닝된 마지막 계수이고, 블록 (100)의 좌상측 코너에 위치된다. Scan_pos 0에서의 양자화된 계수는 10의 절대값을 갖는다. 4×4 변환 블록 또는 더 큰 변환 블록에서의 마지막 4×4 서브블록의 경우, 첫번째 4개의 sigMapFlag들은 코딩될 필요가 없는데, 왜냐하면 마지막 비-제로 계수의 포지션은 공지되기 때문이다. 즉, sigMapFlag의 코딩은 마지막 비-제로 계수 (이 예에 있어서, scan_pos 11에서의 계수)에서 시작할 수도 있다. 일부 예들에 있어서, 첫번째 비-제로의 포지션이 또한 공지될 수도 있어서, scan_pos 11에 대한 SigMapFlag는 부가적으로 시그널링되지 않을 수도 있다.

Scan_pos	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
계수들	0	0	0	0	1	-1	0	2	0	3	2	-1	0	5	-7	10
sigMapFlag	0	0	0	0	1	1	0	1	0	1	1	0	1	1	1	
gr1Flag					0	0		1		1	1	0		1	1	1
gr2Flag							0		1	0			1	1	1	
signFlag						0	1		0		0	0	1	1	0	
levelRem									0				2	4	7	

[0057]

[표 1. 4×4 TU 또는 4×4 CG의 계수들에 대한 코딩된 심볼들]

- [0059] 이들 심볼들 중에서, sigMapFlag, gr1Flag 및 gr2Flag의 빈들은 적응 컨텍스트 모델들로 인코딩된다. signFlag 및 levelRem의 이진화된 빈들은 고정된 동등 확률 모델들 (예를 들어, 지수 골룸 코드)을 갖는 바이패스 모드를 통해 인코딩된다.

- [0060] 블록 또는 서브블록의 유의도의 코딩은 2부분들을 포함할 수도 있다. 첫째, 코딩된 서브블록 플래그 (CSBF)가, 서브블록에서 임의의 비-제로 계수들이 존재하는지 여부를 표시하기 위해 각각의 계수 그룹에 대해

코딩 (또는 추론) 된다. CSBF 가 1 이면, 계수 그룹에서의 각각의 변환 계수에 대한 유의 계수 플래그들이 코딩된다. WD8 에 있어서, 유의 플래그 컨텍스트들은 4×4 서브블록 내에서의 계수의 포지션, 서브블록이 DC 계수를 포함하는지 여부, 및 우측 (CSBF_R) 및 하부 (CSBF_L) 에 대한 서브블록의 CSBF 에 의존한다.

[0061] 도 3a 내지 도 3d 는 CSBF_R 및 CSBF_L 에 의존하여 4×4 서브블록의 컨텍스트 할당을 위한 4개의 상이한 패턴들을 도시한다. 4×4 서브블록 내의 계수들에는, 도 3a 내지 도 3d 에 도시된 바와 같이, CSBF_R 및 CSBF_L 에 의존하여 비디오 인코더 (20) 및 비디오 디코더 (30) 에 의해 컨텍스트들이 할당된다. 도 3a 내지 도 3d 의 블록들에서의 수치들 (0, 1, 및 2) 은 상이한 컨텍스트들에 대응한다. 도 3a 내지 도 3d 의 예에 있어서, 2 의 컨텍스트는 계수가 유의하다는 높은 가능성 (예를 들어, 50/50 초과) 을 나타낼 수도 있지만, 0 의 컨텍스트는 계수가 유의하다는 낮은 가능성 (예를 들어, 50/50 미만) 을 나타낼 수도 있다. 1 의 컨텍스트는 계수가 유의하다는 거의 50/50 가능성을 나타낼 수도 있다. 도 3a 내지 도 3d 는 컨텍스트 패턴들의 단지 일 예일 뿐이며, 본 개시의 기술들은 더 많거나 더 적은 컨텍스트들을 갖는 패턴들을 포함하여 상이한 컨텍스트 패턴들과 함께 사용될 수도 있음이 고려된다.

[0062] 4×4 서브블록이 DC 계수를 포함하지 않으면, 3 의 컨텍스트 오프셋이 적용된다. 예로서, 4×4 서브블록이 DC 계수를 포함하지 않고 도 3a 내지 도 3d 로부터의 컨텍스트 할당이 2 이면, 사용된 실제 컨텍스트는 5 일 것이다. 즉, 컨텍스트 도출 프로세스는 양자 모든 경우들에서 정확하게 동일하지만, DC 및 비-DC 서브블록들에 대한 컨텍스트의 상이한 세트들이 사용된다 (동일한 컨텍스트들을 공유하고 있지는 않음). 이러한 예시적인 프로세스는 루마 변환 계수들에 대한 것이다. 크로마 변환 계수들에 대해, 4×4 서브블록이 DC 계수를 포함하는지 여부에 기초한 컨텍스트 오프셋은 적용되지 않으며, 컨텍스트들은 모든 서브블록들에 대해 공유된다. 따라서, 오직 3개의 컨텍스트들이 크로마를 위해 사용된다. DC 계수는 항상 별도의 컨텍스트를 사용하며, 이 별도의 컨텍스트는 모든 TU 사이즈들에 대해 공유된다.

[0063] 도 3a 는, 비-제로 계수들을 갖지 않는 하위의 이웃 서브블록 (즉, CSBF_L = 0) 및 비-제로 계수들을 갖지 않는 우측의 이웃 서브블록 (즉, CSBF_R = 0) 을 갖는 서브블록에서의 계수들에 대한 컨텍스트들의 예를 도시한다. 도 3b 는, 비-제로 계수들을 갖지 않는 하위의 이웃 서브블록 (즉, CSBF_L = 0) 및 적어도 하나의 비-제로 계수들을 갖는 우측의 이웃 서브블록 (즉, CSBF_R = 1) 을 갖는 서브블록에서의 계수들에 대한 컨텍스트들의 예를 도시한다. 도 3c 는, 적어도 하나의 비-제로 계수들을 갖는 하위의 이웃 서브블록 (즉, CSBF_L = 1) 및 비-제로 계수들을 갖지 않는 우측의 이웃 서브블록 (즉, CSBF_R = 0) 을 갖는 서브블록에서의 계수들에 대한 컨텍스트들의 예를 도시한다. 도 3d 는, 적어도 하나의 비-제로 계수들을 갖는 하위의 이웃 서브블록 (즉, CSBF_L = 1) 및 적어도 하나의 비-제로 계수들을 갖는 우측의 이웃 서브블록 (즉, CSBF_R = 1) 을 갖는 서브블록에서의 계수들에 대한 컨텍스트들의 예를 도시한다.

[0064] 도 3a 내지 도 3d 는 HEVC 에서 사용된 실제 컨텍스트 오프셋들 대신 0 으로 시작하는 컨텍스트 넘버링을 도시한다. HEVC 컨텍스트들이 인덱싱된다. 예를 들어, sig_coeff_flag 에 대해, HEVC 컨텍스트들은 블록 사이즈 및 루마인지 또는 크로마인지 여부에 의존하여 인덱싱된다. 단순화를 위해, 컨텍스트들은, 블록 사이즈 및 컬러 컴포넌트로 기인하여 수반된 상이한 오프셋들을 무시하고 도 3a 내지 도 3d 에서 단지 0 으로부터 넘버링된다. 오프셋들은, 본 개시의 기술들의 이해 또는 구현에 필수적이지 않은 HEVC 에서의 관례이다.

[0065] HEVC 에 있어서, 유의도 정보는, 상기 도입된 바와 같은 다중의 레벨들로 코딩된다. 코딩된 블록 플래그 (CBF) 는 전체 변환 블록의 유의도를 시그널링한다. 즉, CBF 는 전체 변환 블록이 임의의 유의 (즉, 비-제로) 계수들을 포함하는지 여부를 표시한다. 변환 블록 내에서, 일 레벨은 (CSBF 를 이용하여) 서브블록의 유의도에 대응하고, 다른 레벨은 개별 계수들의 유의도에 대응한다. 이 레벨에서, 개별 계수들의 유의도들은 상기 설명된 선택스 엘리먼트 (significant_coeff_flag) 를 이용하여 시그널링된다.

[0066] 이 예에서 설명된 바와 같이, HEVC 에서 시그널링하는 유의도의 3개 레벨들이 존재한다. 본 개시의 예들에 따르면, (4×4 서브블록 레벨에 있는) CSBF 와 (개별 계수 레벨에 있는) significant_coeff_flag 사이에서, 2×2 서브블록 레벨에서와 같이 유의도 정보를 시그널링하는 다른 레벨이 존재할 수 있다. 더 일반적으로, 본 개시의 기술들에 따르면, 유의도 시그널링은 전체 블록 레벨, 개별 계수 레벨, 및 2 이상의 서브블록 레벨들에서 발생할 수도 있다. 2 이상의 서브블록 레벨들은, 예를 들어, 2×2 서브블록 레벨 및 4×4 서브블록 레벨들일 수도 있지만, 다른 서브블록 레벨들이 또한 사용될 수도 있다.

- [0067] **2×2 서브블록 유의도는 자신의 플래그 및 컨텍스트 도출과 패턴들을 활용할 수도 있다.** 본 개시는, 변환 계수들을 시그널링할 경우에 잠재적으로 더 우수한 성능을 위해 서브블록 레벨들에서의 유의도 시그널링을 활용하기 위한 그리고 2×2 유의도 (또는 다른 중간 사이즈들) 를 시그널링하기 위한 방법들을 기술한다. 따라서, 다른 서브블록 레벨들은, 코딩되고 있는 현재 서브블록 레벨에 대한 유의도 정보를 시그널링하는데 사용되는 컨텍스트 정보를 제공할 수도 있다. 다른 서브블록 레벨의 유의도 정보는 엔트로피 인코더로 하여금 현재 레벨에서의 유의도 정보의 확률로 모드를 개선할 수 있게 하고, 따라서 코딩 효율을 잠재적으로 개선시킬 수도 있다. 비록 본 개시의 기술들이 설명의 목적들로 2×2 서브블록 레벨을 이용할 수도 있지만, 본 명세서에서 설명된 기술들은 2×2 서브블록 레벨로 한정되지 않고 다른 서브블록 레벨들로 적용될 수도 있음이 이해되어야 한다.
- [0068] 본 개시의 기술들에 따르면, 선택스 엘리먼트 (예를 들어, 플래그) 는 부가적인 서브블록의 유의도를 시그널링하는데 사용될 수 있다. 이 플래그는 일반적으로, 예를 들어, coded_additional_sub_block_flag (CASBF) 로 지칭될 수 있다. 일 예에 있어서, 부가적인 서브블록은 2×2 서브블록일 수도 있으며, 이 경우, 플래그는, 예를 들어, coded_2x2_sub_block_flag (C2SBF) 로 지칭될 수도 있다. C2SBF 는, 2×2 서브블록에서의 적어도 하나의 계수가 유의하다면 1 과 동일하게 설정될 수 있다. 그렇지 않으면, C2SBF 는 0 으로 설정될 수 있다.
- [0069] CASBF 또는 C2SBF 의 코딩을 잠재적으로 개선하기 위해, 플래그는 컨텍스트 코딩될 수 있다. 특정 서브블록에 대해 사용된 컨텍스트의 선택은 성능에 영향을 줄 수도 있다. 본 개시의 기술들에 따르면, C2SBF 컨텍스트는 이미 코딩된 이웃하는 2×2 서브블록들의 C2SBF, 및/또는 이웃하는 4×4 서브블록들의 CSBF, 및/또는 서브블록의 이웃하는 계수들의 significant_coeff_flag 에 의존할 수도 있다. 컨텍스트 도출을 위해 사용된 이웃하는 계수들/서브블록들은 현재 서브블록의 인과 (causal) 이웃을 포함할 수도 있다. 이에 관하여, "인과" 이웃들은 코딩되었던 이웃들이고, 따라서, 그 정보는, 현재의 유의도 정보를 코딩할 경우, 비디오 인코더 (20) 및 비디오 디코더 (30) 양자에게 액세스 가능하다. 이 예에 있어서, 현재의 유의도 정보는 C2SBF 플래그를 포함할 수도 있다. 일 예에 있어서, 이웃은 우측의 계수/서브블록, 하부의 계수/서브블록, 및/또는 우하측 대각선의 계수/서브블록을 포함할 수도 있다.
- [0070] 중간 서브블록/CG 사이즈에 대한 컨텍스트 할당이 이제 설명될 것이다. 본 개시의 일 예에 있어서, 2×2 서브블록들에 대한 컨텍스트 할당을 위한 패턴들은 이전에 코딩된 C2SBF 및/또는 이전에 코딩된 CSBF 에 의존할 수도 있다. 일 예에 있어서, 패턴은 현재의 서브블록 우측 및 하부에 대한 서브블록들에 의존한다. 다른 예에 있어서, 패턴은 또한, 4×4 서브블록 내에서 및/또는 변환 블록 내에서 2×2 서브블록의 포지션에 의존한다.
- [0071] 본 개시의 다른 예에 있어서, 컨텍스트 할당은 또한, 4×4 서브블록 내에서 및/또는 변환 블록 내에서 2×2 서브블록의 포지션에 의존할 수도 있다. 즉, 2개의 2×2 서브블록들은 이웃하는 서브블록들에서 동일한 유의도 맵을 가질 수도 있다. 2×2 서브블록들은 동일한 패턴을 할당받지만, 변환 블록 내에서의 그 포지션에 의존하여 상이한 컨텍스트 세트들을 가질 수도 있다.
- [0072] 도 4 는 본 개시에서 설명된 기술들을 구현할 수도 있는 예시적인 비디오 인코더 (20) 를 도시한 블록 다이어그램이다. 비디오 인코더 (20) 는 비디오 슬라이스들 내에서 비디오 블록들의 인트라-코딩 및 인터-코딩을 수행할 수도 있다. 인트라 코딩은 소정의 비디오 프레임 또는 픽처 내 비디오에 있어서 공간 리던던시를 감소 또는 제거하기 위해 공간 예측에 의존한다. 인터-코딩은 비디오 시퀀스의 인접한 프레임들 또는 픽처들 내 비디오에 있어서 시간 리던던시를 감소 또는 제거하기 위해 시간 예측에 의존한다. 인트라-모드 (I 모드) 는 수개의 공간 기반 압축 모드들 중 임의의 모드를 지정할 수도 있다. 단방향 예측 (P 모드) 또는 양방향 예측 (B 모드) 과 같은 인터-모드들은 수개의 시간 기반 압축 모드들 중 임의의 모드를 지정할 수도 있다.
- [0073] 도 4 의 예에 있어서, 비디오 인코더 (20) 는 파티셔닝 유닛 (35), 예측 프로세싱 유닛 (41), 필터 유닛 (63), 디코딩된 픽처 버퍼 (64), 합산기 (50), 변환 프로세싱 유닛 (52), 양자화 유닛 (54), 및 엔트로피 인코딩 유닛 (56) 을 포함한다. 예측 프로세싱 유닛 (41) 은 모션 추정 유닛 (42), 모션 보상 유닛 (44) 및 인트라 예측 프로세싱 유닛 (46) 을 포함한다. 비디오 블록 복원을 위해, 비디오 인코더 (20) 는 또한 역양자화 유닛 (58), 역변환 프로세싱 유닛 (60), 및 합산기 (62) 를 포함한다. 필터 유닛 (63) 은 디블록킹 (deblocking) 필터, 적응 루프 필터 (ALF), 및 샘플 적응 오프셋 (SAO) 필터와 같은 하나 이상의 루프 필터들을 나타내도록 의도된다. 비록 필터 유닛 (63) 이 인-루프 (in loop) 필터인 것으로서 도 4 에 도시되지만, 다른 구성들에 있어서, 필터 유닛 (63) 은 포스트 루프 필터로서 구현될 수도 있다.

[0074]

도 4 에 도시된 바와 같이, 비디오 인코더 (20) 는 비디오 데이터를 수신하고, 파티셔닝 유닛 (35) 은 그 데이터를 비디오 블록들로 파티셔닝한다. 이러한 파티셔닝은 또한, 예를 들어, LCU들 및 CU들의 쿼드트리 구조에 따른 비디오 블록 파티셔닝뿐 아니라 슬라이스들, 타일들, 또는 다른 더 큰 유닛들로의 파티셔닝을 포함할 수도 있다. 비디오 인코더 (20) 는 일반적으로, 인코딩될 비디오 슬라이스 내에서의 비디오 블록들을 인코딩하는 컴포넌트들을 예시한다. 슬라이스는 다중의 비디오 블록들로 (및 가능하게는 타일들로서 지정되는 비디오 블록들의 세트들로) 분할될 수도 있다. 예측 프로세싱 유닛 (41) 은 예를 들어, 코딩 레이트 및 왜곡의 레벨) 에 기반한 현재의 비디오 블록에 대해 복수의 인트라 코딩 모드들 중 하나 또는 복수의 인터 코딩 모드들 중 하나와 같이 복수의 가능한 코딩 모드들 중 하나를 선택할 수도 있다. 예측 프로세싱 유닛 (41) 은 결과적인 인트라- 또는 인터-코딩된 블록을 합산기 (50) 에 제공하여 잔차 블록 데이터를 생성하고, 합산기 (62) 에 제공하여 레퍼런스 픽처로서의 사용을 위한 인코딩된 블록을 복원할 수도 있다.

[0075]

예측 프로세싱 유닛 (41) 내의 인트라 예측 프로세싱 유닛 (46) 은 공간 압축을 제공하기 위해, 코딩될 현재 블록과 동일한 프레임 또는 슬라이스에 있어서의 하나 이상의 이웃하는 블록들에 대해 현재 비디오 블록의 인트라-예측 코딩을 수행할 수도 있다. 예측 프로세싱 유닛 (41) 내의 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 시간 압축을 제공하기 위해 하나 이상의 레퍼런스 픽처들에 있어서의 하나 이상의 예측 블록들에 대해 현재 비디오 블록의 인터-예측 코딩을 수행한다.

[0076]

모션 추정 유닛 (42) 은 비디오 시퀀스에 대한 미리결정된 패턴에 따라 비디오 슬라이스에 대한 인터-예측 모드를 결정하도록 구성될 수도 있다. 미리결정된 패턴은 시퀀스에서의 비디오 슬라이스들을, P 슬라이스들, B 슬라이스들 또는 GPB 슬라이스들로서 명시할 수도 있다. 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 고도로 통합될 수도 있지만, 개념적인 목적을 위해 별개로 도시된다. 모션 추정 유닛 (42) 에 의해 수행된 모션 추정은, 비디오 블록들에 대한 모션을 추정하는 모션 벡터들을 생성하는 프로세스이다. 모션 벡터는, 예를 들어, 레퍼런스 픽처 내에서의 예측 블록에 대한 현재 비디오 프레임 또는 픽처 내의 비디오 블록의 PU 의 변위를 표시할 수도 있다.

[0077]

예측 블록은 픽셀 차이의 관점에서 코딩될 비디오 블록의 PU 와 밀접하게 매칭되도록 발견되는 블록이며, 이 픽셀 차이는 절대 차이의 합 (SAD), 제곱 차이의 합 (SSD), 또는 다른 상이한 메트릭들에 의해 결정될 수도 있다. 일부 예들에 있어서, 비디오 인코더 (20) 는 디코딩된 픽처 베퍼 (64) 에 저장된 레퍼런스 픽처들의 서브-정수 픽셀 포지션들에 대한 값들을 계산할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 레퍼런스 픽처의 1/4 픽셀 포지션들, 1/8 픽셀 포지션들, 또는 다른 분수 픽셀 포지션들의 값들을 보간할 수도 있다. 따라서, 모션 추정 유닛 (42) 은 풀 픽셀 포지션들 및 분수 픽셀 포지션들에 대한 모션 탐색을 수행하고, 분수 픽셀 정밀도로 모션 벡터를 출력할 수도 있다.

[0078]

모션 추정 유닛 (42) 은 인터-코딩된 슬라이스에 있어서의 비디오 블록의 PU 에 대한 모션 벡터를, 그 PU 의 포지션을 레퍼런스 픽처의 예측 블록의 포지션과 비교함으로써 계산한다. 레퍼런스 픽처는 제 1 레퍼런스 픽처 리스트 (리스트 0) 또는 제 2 레퍼런스 픽처 리스트 (리스트 1) 로부터 선택될 수도 있으며, 이 리스트를 각각은 디코딩된 픽처 베퍼 (64) 에 저장된 하나 이상의 레퍼런스 픽처들을 식별한다. 모션 추정 유닛 (42) 은 계산된 모션 벡터를 엔트로피 인코딩 유닛 (56) 및 모션 보상 유닛 (44) 으로 전송한다.

[0079]

모션 보상 유닛 (44) 에 의해 수행된 모션 보상은 모션 추정에 의해 결정된 모션 벡터에 기초하여 예측 블록을 폐치 또는 생성하여, 가능하게는 서브픽셀 정밀도로 보간들을 수행하는 것을 수반할 수도 있다. 현재 비디오 블록의 PU 에 대한 모션 벡터를 수신할 시, 모션 보상 유닛 (44) 은, 모션 벡터가 레퍼런스 픽처 리스트들 중 하나에서 포인팅하는 예측 블록을 로케이팅할 수도 있다. 비디오 인코더 (20) 는, 코딩되고 있는 현재 비디오 블록의 픽셀 값들로부터 예측 블록의 픽셀 값들을 감산하여 픽셀 차이 값들을 형성함으로써 잔차 비디오 블록을 형성한다. 픽셀 차이 값들은 블록에 대한 잔차 데이터를 형성하고, 루마 및 크로마 차이 컴포넌트들 양자를 포함할 수도 있다. 합산기 (50) 는 이러한 감산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 모션 보상 유닛 (44) 은 또한, 비디오 슬라이스의 비디오 블록들을 디코딩함에 있어서 비디오 디코더 (30) 에 의한 사용을 위해 비디오 블록들 및 비디오 슬라이스와 연관된 신택스 엘리먼트들을 생성할 수도 있다.

[0080]

인트라-예측 프로세싱 유닛 (46) 은 현재 블록을, 상기 설명된 바와 같은 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 수행된 인터-예측에 대한 대안으로서 인트라-예측할 수도 있다. 특히, 인트라-예측 프로세싱 유닛 (46) 은 현재 블록을 인코딩하는데 이용하기 위한 인트라-예측 모드를 결정할 수도 있다. 일부 예들에 있어서, 인트라-예측 프로세싱 유닛 (46) 은 예를 들어 별도의 인코딩 패스를 동안에 다양한 인트라-예측 모드들을 이용하여 현재 블록을 인코딩할 수도 있으며, 인트라-예측 프로세싱 유닛 (46) (또는 일부 예들에서는

모드 선택 유닛 (40)) 은 테스팅된 모드들로부터의 이용을 위해 적절한 인트라-예측 모드를 선택할 수도 있다. 예를 들어, 인트라-예측 프로세싱 유닛 (46) 은 다양한 테스팅된 인트라-예측 모드들에 대한 레이트-왜곡 분석을 이용하여 레이트-왜곡 값들을 계산하고, 테스팅된 모드들 중 최상의 레이트-왜곡 특성들을 갖는 인트라-예측 모드를 선택할 수도 있다. 레이트-왜곡 분석은 일반적으로, 인코딩된 블록과 그 인코딩된 블록을 생성하도록 인코딩되었던 오리지널의 인코딩되지 않은 블록 간의 왜곡 (또는 에러) 의 양뿐 아니라 인코딩된 블록을 생성하는데 사용된 비트 레이트 (즉, 비트들의 수) 를 결정한다. 인트라-예측 프로세싱 유닛 (46) 은 다양한 인코딩된 블록들에 대한 왜곡들 및 레이트들로부터의 비율들을 계산하여, 어느 인트라-예측 모드가 그 블록에 대한 최상의 레이트-왜곡 값을 나타내는지를 결정할 수도 있다.

[0081] 어떤 경우든, 블록에 대한 인트라-예측 모드를 선택한 이후, 인트라-예측 프로세싱 유닛 (46) 은 블록에 대한 선택된 인트라-예측 모드를 표시하는 정보를 엔트로피 코딩 유닛 (56) 에 제공할 수도 있다. 엔트로피 코딩 유닛 (56) 은, 본 개시의 기술들에 따라, 선택된 인트라-예측 모드를 표시한 정보를 인코딩할 수도 있다. 비디오 인코더 (20) 는 송신된 비트스트림에, 복수의 인트라-예측 모드 인덱스 테이블들 및 복수의 변형된 인트라-예측 모드 인덱스 테이블들 (코드워드 매핑 테이블들로서도 또한 지칭됨) 을 포함할 수도 있는 구성 데이터, 다양한 블록들에 대한 인코딩 컨텍스트들의 정의들, 및 컨텍스트들 각각에 대한 사용을 위한 가장 가능성있는 인트라-예측 모드, 인트라-예측 모드 인덱스 테이블, 및 변형된 인트라-예측 모드 인덱스 테이블의 표시들을 포함할 수도 있다.

[0082] 예측 프로세싱 유닛 (41) 이 인터-예측 또는 인트라-예측 중 어느 하나를 통해 현재 비디오 블록에 대한 예측 블록을 생성한 이후, 비디오 인코더 (20) 는 현재 비디오 블록으로부터 예측 블록을 감산함으로써 잔차 비디오 블록을 형성한다. 잔차 블록에서의 잔차 비디오 데이터는 하나 이상의 TU들에 포함되고 변환 프로세싱 유닛 (52) 에 적용될 수도 있다. 변환 프로세싱 유닛 (52) 은 이산 코사인 변환 (DCT) 또는 개념적으로 유사한 변환과 같은 변환을 이용하여 잔차 비디오 데이터를 잔차 변환 계수들로 변환한다. 변환 프로세싱 유닛 (52) 은 잔차 비디오 데이터를 픽셀 도메인으로부터 주파수 도메인과 같은 변환 도메인으로 컨버팅할 수도 있다.

[0083] 변환 프로세싱 유닛 (52) 은 결과적인 변환 계수들을 양자화 유닛 (54) 으로 전송할 수도 있다. 양자화 유닛 (54) 은 비트 레이트를 추가로 감소시키기 위해 변환 계수들을 양자화한다. 양자화 프로세스는 그 계수들의 일부 또는 그 모두와 연관되는 비트 깊이를 감소시킬 수도 있다. 양자화의 정도는 양자화 파라미터를 조정함으로써 변경될 수도 있다. 그 후, 일부 예들에 있어서, 양자화 유닛 (54) 은, 양자화된 변환 계수들을 포함하는 매트릭스의 스캔을 수행할 수도 있다. 대안적으로, 엔트로피 인코딩 유닛 (56) 은 스캔을 수행할 수도 있다. 양자화 이후, 엔트로피 인코딩 유닛 (56) 은 양자화된 변환 계수들을 엔트로피 인코딩한다. 엔트로피 인코딩 유닛 (56) 은, 예를 들어, 상기 설명된 CSBFR, CSBFB, CASBF, 및 C2SBF 플래그들뿐 아니라 상기 설명된 significant_coeff_flag, coeff_abs_level_greater1_flag, coeff_abs_level_greater2_flag, coeff_sign_flag, 및 coeff_abs_level_remaining 신택스 엘리먼트들을 생성할 수도 있다.

[0084] 예를 들어, 엔트로피 인코딩 유닛 (56) 은 컨텍스트 적응 가변 길이 코딩 (CAVLC), 컨텍스트 적응 이진 산술 코딩 (CABAC), 신택스 기반 컨텍스트 적응 이진 산술 코딩 (SBAC), 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩, 또는 다른 엔트로피 인코딩 방법 또는 기술을 수행할 수도 있다. 엔트로피 인코딩 유닛 (56) 에 의한 엔트로피 인코딩 이후, 인코딩된 비트스트림은 비디오 디코더 (30) 로 송신되거나, 또는 비디오 디코더 (30) 에 의한 더 나중의 송신 또는 취출을 위해 아카이브될 수도 있다. 엔트로피 인코딩 유닛 (56) 은 또한, 코딩되고 있는 현재의 비디오 슬라이스에 대한 모션 벡터들 및 다른 신택스 엘리먼트들을 엔트로피 인코딩할 수도 있다.

[0085] 역양자화 유닛 (58) 및 역변환 프로세싱 유닛 (60) 은, 각각, 역양자화 및 역변환을 적용하여, 레퍼런스 픽처의 레퍼런스 블록으로서의 더 나중 사용을 위해 픽셀 도메인에서 잔차 블록을 복원한다. 모션 보상 유닛 (44) 은, 레퍼런스 픽처 리스트들 중 하나 내에서의 레퍼런스 픽처들 중 하나의 예측 블록에 잔차 블록을 부가함으로써 레퍼런스 블록을 계산할 수도 있다. 모션 보상 유닛 (44) 은 또한, 하나 이상의 보간 필터들을 복원된 잔차 블록에 적용하여, 모션 추정에서 사용하기 위한 서브-정수 픽셀 값들을 계산할 수도 있다. 합산기 (62) 는 복원된 잔차 블록을, 모션 보상 유닛 (44) 에 의해 생성된 모션 보상된 예측 블록에 부가하여, 디코딩된 픽처 버퍼 (64) 로의 저장을 위한 레퍼런스 블록을 생성한다. 레퍼런스 블록은, 후속 비디오 프레임 또는 픽처에서의 블록을 인터-예측하기 위해 레퍼런스 블록으로서 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 사용될 수도 있다.

[0086] 이러한 방식으로, 도 4 의 비디오 인코더 (20) 는 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보

를 코딩하는 것으로서, 제 1 유의도 정보는 제 1 서브블록의 유의도들에 대응하고, 제 1 서브블록은 전체 변환 블록의 서브블록인, 상기 제 1 유의도 정보를 코딩하고; 제 2 유의도 정보를 코딩하는 것으로서, 제 2 유의도 정보는 제 2 서브블록의 유의도에 대응하고 제 2 서브블록은 제 1 서브블록의 서브블록인, 상기 제 2 유의도 정보를 코딩하며; 그리고 제 2 유의도 정보에 대한 산술 코딩 동작을 수행하도록 구성된 비디오 인코더의 예를 나타낸다. 산술 코딩 동작을 위한 컨텍스트는 제 1 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들에 기초하여 결정될 수 있다. 일 예에 있어서, 제 1 서브블록은 4×4 블록이고, 제 2 서브블록은 2×2 서브블록이다. 제 2 유의도 정보는, 제 2 서브블록의 적어도 하나의 계수가 비-제로 계수인지를 표시할 수 있다. 컨텍스트는 제 2 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들의 유의도 정보에 기초하거나, 하나 이상의 이웃하는 개별 계수들의 유의도 정보에 기초하거나, 또는 제 1 서브블록 내에서의 제 2 서브블록의 포지션에 기초할 수 있다. 컨텍스트 할당을 위한 패턴은 제 1 서브블록 내에서의 제 2 서브블록의 포지션에 기초할 수 있다.

[0087] 도 5 는 본 개시에서 설명된 기술들을 구현할 수도 있는 예시적인 비디오 디코더 (30)를 도시한 블록 다이어그램이다. 도 5 의 예에 있어서, 비디오 디코더 (30)는 엔트로피 디코딩 유닛 (80), 예측 프로세싱 유닛 (81), 역양자화 유닛 (86), 역변환 프로세싱 유닛 (88), 합산기 (90), 필터 유닛 (91), 및 디코딩된 픽처 버퍼 (92)를 포함한다. 예측 프로세싱 유닛 (81)은 모션 보상 유닛 (82) 및 인트라 예측 프로세싱 유닛 (84)을 포함한다. 일부 예들에 있어서, 비디오 디코더 (30)는 도 4로부터의 비디오 인코더 (20)에 대하여 설명된 인코딩 패스에 일반적으로 상호적인 디코딩 패스를 수행할 수도 있다.

[0088] 디코딩 프로세스 동안, 비디오 디코더 (30)는, 인코딩된 비디오 슬라이스의 비디오 블록들 및 관련 신팩스 엘리먼트들을 나타내는 인코딩된 비디오 비트스트림을 비디오 인코더 (20)로부터 수신한다. 비디오 디코더 (30)의 엔트로피 디코딩 유닛 (80)은 비트스트림을 엔트로피 디코딩하여, 양자화된 계수들, 모션 벡터들, 및 다른 신팩스 엘리먼트들을 생성한다. 엔트로피 디코딩 유닛 (80)은, 예를 들어, 상기 설명된 CSBFR, CSBFB, CASBF, 및 C2SBF 플래그들뿐 아니라 상기 설명된 significant_coeff_flag, coeff_abs_level_greater1_flag, coeff_abs_level_greater2_flag, coeff_sign_flag, 및 coeff_abs_level_remaining 신팩스 엘리먼트들을 수신 및 파싱함으로써 양자화된 계수들을 생성할 수도 있다. 엔트로피 디코딩 유닛 (80)은 모션 벡터들 및 다른 신팩스 엘리먼트들을 예측 프로세싱 유닛 (81)으로 포워딩한다. 비디오 디코더 (30)는 신팩스 엘리먼트들을 비디오 슬라이스 레벨 및/또는 비디오 블록 레벨로 수신할 수도 있다.

[0089] 비디오 슬라이스가 인트라-코딩된 (I) 슬라이스로서 코딩된 경우, 예측 프로세싱 유닛 (81)의 인트라 예측 프로세싱 유닛 (84)은 현재 프레임 또는 픽처의 이전에 디코딩된 블록들로부터의 데이터 및 시그널링된 인트라 예측 모드에 기초하여 현재 비디오 슬라이스의 비디오 블록에 대한 예측 데이터를 생성할 수도 있다. 비디오 프레임이 인터-코딩된 (즉, B, P 또는 GPB) 슬라이스로서 코딩된 경우, 예측 프로세싱 유닛 (81)의 모션 보상 유닛 (82)은 엔트로피 디코딩 유닛 (80)으로부터 수신된 모션 벡터들 및 다른 신팩스 엘리먼트들에 기초하여 현재 비디오 슬라이스의 비디오 블록에 대한 예측 블록들을 생성한다. 예측 블록들은 레퍼런스 픽처 리스트들 중 하나 내에서의 레퍼런스 픽처들 중 하나로부터 생성될 수도 있다. 비디오 디코더 (30)는 디코딩된 픽처 버퍼 (92)에 저장된 레퍼런스 픽처들에 기초한 디폴트 구성 기술들을 이용하여 레퍼런스 프레임 리스트들, 즉, 리스트 0 및 리스트 1을 구성할 수도 있다.

[0090] 모션 보상 유닛 (82)은 모션 벡터들 및 다른 신팩스 엘리먼트들을 파싱함으로써 현재 비디오 슬라이스의 비디오 블록에 대한 예측 정보를 결정하고, 그 예측 정보를 사용하여, 디코딩되고 있는 현재 비디오 블록에 대한 예측 블록들을 생성한다. 예를 들어, 모션 보상 유닛 (82)은 수신된 신팩스 엘리먼트들 중 일부를 사용하여, 비디오 슬라이스의 비디오 블록들을 코딩하는데 사용되는 예측 모드 (예를 들어, 인트라-예측 또는 인터-예측), 인터-예측 슬라이스 타입 (예를 들어, B 슬라이스, P 슬라이스, 또는 GPB 슬라이스), 슬라이스에 대한 레퍼런스 픽처 리스트들 중 하나 이상에 대한 구성 정보, 슬라이스의 각각의 인터-인코딩된 비디오 블록에 대한 모션 벡터들, 슬라이스의 각각의 인터-코딩된 비디오 블록에 대한 인터-예측 상태, 및 현재 비디오 슬라이스에 있어서의 비디오 블록들을 디코딩하기 위한 다른 정보를 결정한다.

[0091] 모션 보상 유닛 (82)은 또한, 보간 필터들에 기초하여 보간을 수행할 수도 있다. 모션 보상 유닛 (82)은 비디오 블록들의 인코딩 동안 비디오 인코더 (20)에 의해 사용된 바와 같은 보간 필터들을 이용하여, 레퍼런스 블록들의 서브-정수 픽셀들에 대한 보간된 값들을 계산할 수도 있다. 이 경우, 모션 보상 유닛 (82)은 수신된 신팩스 엘리먼트들로부터 비디오 인코더 (20)에 의해 사용된 보간 필터들을 결정하고, 보간 필터들을 이

용하여 예측 블록들을 생성할 수도 있다.

[0092] 역양자화 유닛 (86)은 비트스트림에서 제공되고 엔트로피 디코딩 유닛 (80)에 의해 디코딩되는 양자화된 변환 계수들을 역양자화, 즉, 양자화해제한다. 역양자화 프로세스는 비디오 슬라이스에 있어서의 각각의 비디오 블록에 대해 비디오 인코더 (20)에 의해 계산된 양자화 파라미터의 이용을 포함하여, 적용되어야 하는 양자화의 정도 및 유사하게 역양자화의 정도를 결정할 수도 있다. 역변환 프로세싱 유닛 (88)은 픽셀 도메인에서 잔차 블록들을 생성하기 위해, 역변환, 예를 들어, 역 DCT, 정수 역변환, 또는 개념적으로 유사한 역변환 프로세스를 변환 계수들에 적용한다.

[0093] 모션 보상 유닛 (82)이 모션 벡터들 및 다른 십택스 엘리먼트들에 기초하여 현재 비디오 블록에 대한 예측 블록을 생성한 이후, 비디오 디코더 (30)는 역변환 프로세싱 유닛 (88)으로부터의 잔차 블록들을 모션 보상 유닛 (82)에 의해 생성된 대응하는 예측 블록들과 합산함으로써 디코딩된 비디오 블록을 형성한다. 합산기 (90)는 이러한 합산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 요구된다면, (코딩 루프에 있어서 또는 코딩 루프 이후에) 루프 필터들이 또한 픽셀 천이들을 평활하게 하거나 그렇지 않으면 비디오 품질을 개선하기 위해 이용될 수도 있다. 필터 유닛 (91)은 디블록킹 필터, 적응 루프 필터 (ALF), 및 샘플 적응 오프셋 (SAO) 필터와 같은 하나 이상의 루프 필터들을 나타내도록 의도된다. 비록 필터 유닛 (91)이 인-루프 필터인 것으로서 도 5에 도시되지만, 다른 구성들에 있어서, 필터 유닛 (91)은 포스트 루프 필터로서 구현될 수도 있다. 그 후, 소정의 프레임 또는 픽처에 있어서의 디코딩된 비디오 블록들이 디코딩된 픽처 버퍼 (92)에 저장되고, 이 디코딩된 픽처 버퍼는 후속적인 모션 보상을 위해 사용된 레퍼런스 픽처들을 저장한다.

디코딩된 픽처 버퍼 (92)는 또한, 도 1의 디스플레이 디바이스 (32)와 같은 디스플레이 디바이스 상으로의 더 나중의 프리젠테이션을 위해 디코딩된 비디오를 저장한다.

[0094] 이러한 방식으로, 도 5의 비디오 디코더 (30)는 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 코딩하는 것으로서, 제 1 유의도 정보는 제 1 서브블록의 유의도들에 대응하고, 제 1 서브블록은 전체 변환 블록의 서브블록인, 상기 제 1 유의도 정보를 코딩하고; 제 2 유의도 정보를 코딩하는 것으로서, 제 2 유의도 정보는 제 2 서브블록의 유의도에 대응하고 제 2 서브블록은 제 1 서브블록의 서브블록인, 상기 제 2 유의도 정보를 코딩하며; 그리고 제 2 유의도 정보에 대한 산술 코딩 동작을 수행하도록 구성된 비디오 디코더의 예를 나타낸다. 산술 코딩 동작을 위한 컨텍스트는 제 1 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들에 기초하여 결정될 수 있다. 일 예에 있어서, 제 1 서브블록은 4×4 블록이고, 제 2 서브블록은 2×2 서브블록이다. 제 2 유의도 정보는, 제 2 서브블록의 적어도 하나의 계수가 비-제로 계수인지를 표시할 수 있다. 컨텍스트는 제 2 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들의 유의도 정보에 기초하거나, 하나 이상의 이웃하는 개별 계수들의 유의도 정보에 기초하거나, 또는 제 1 서브블록 내에서의 제 2 서브블록의 포지션에 기초할 수 있다. 컨텍스트 할당을 위한 패턴은 제 1 서브블록 내에서의 제 2 서브블록의 포지션에 기초할 수 있다.

[0095] 유의도 맵들의 부가적인 양태들이 이제 설명될 것이다. 도 6은 비디오 블록에서의 변환 계수들과 비디오 블록과 연관된 유의도 맵 간의 관계를 도시한 개념 다이어그램이다. 도 6에 도시된 바와 같이, 유의도 맵은 비디오 블록에 있어서 유의 계수 값, 즉, 제로 초과의 값의 각각의 인스턴스를 표시하기 위해 "1"을 포함한다. 유의도 맵은, 디코딩될 비디오 블록에 있어서 유의한, 즉, 제로 초과의 계수들의 위치를 결정하기 위해 비디오 디코더 (30)와 같은 비디오 디코더에 의해 디코딩 가능한 비트스트림에서 시그널링될 수도 있다. 더 상세하게, 비디오 블록 내에서의 마지막 비-제로 계수의 포지션이 비트스트림에서 시그널링될 수도 있다. 비디오 블록에 있어서의 마지막 비-제로 계수의 포지션은 비디오 블록에 대해 사용된 스캐닝 순서에 의존한다. 부가적인 십택스 엘리먼트들이, 공지된 또는 공지가능한 스캐닝 순서에 따라 마지막 비-제로 계수에 대한 다른 유의 계수들을 표시하기 위해 시그널링될 수도 있다.

[0096] 도 7a 내지 도 7d는, 각각, 지그재그 스캐닝 순서, 수평 스캐닝 순서, 수직 스캐닝 순서, 및 대각선 스캐닝 순서를 이용하여 스캐닝된 비디오 데이터의 블록들의 예들을 도시한 개념 다이어그램들이다. 도 7a 내지 도 7d에 도시된 바와 같이, 비디오 데이터의 8×8 블록, 예를 들어, CU의 TU는, 원으로 도시된 대응하는 블록 포지션들에 있어서의 64개의 잔차 계수들을 포함할 수도 있다. 예를 들어, 블록들 (101, 102, 103 및 104)은 각각 8×8 의 사이즈를 가지며, 따라서, 전술된 바와 같은 예측 기술들을 이용하여 생성된 64개의 잔차 계수들을 포함할 수도 있다.

[0097] 본 개시에서 설명된 기술들에 따르면, 블록들 (101, 102, 103 및 104) 각각에 있어서의 64개의 잔차 계수들은 2D 변환, 수평 1D 변환, 및 수직 1D 변환 중 하나를 이용하여 변환될 수도 있거나, 또는 잔차 계수들이 전혀 변

환되지 않을 수도 있다. 변환되든지 아니든지, 비디오 블록들 (101, 102, 103 및 104) 각각에 있어서의 계수들은 지그재그 스캐닝 순서, 수평 스캐닝 순서, 및 수직 스캐닝 순서 중 하나를 이용하여 엔트로피 코딩의 준비로 스캐닝된다.

[0098] 도 7a에 도시된 바와 같이, 블록 (101)과 연관된 스캐닝 순서는 지그재그 스캐닝 순서이다. 지그재그 스캐닝 순서는 블록 (101)의 양자화된 변환 계수들을, 도 7a에 있어서의 화살표들에 의해 표시된 바와 같이 대각선 방식으로 스캐닝한다. 유사하게, 도 7d에 있어서, 대각선 스캐닝 순서는 블록 (104)의 양자화된 변환 계수들을, 도 7d에 있어서의 화살표들에 의해 표시된 바와 같이 대각선 방식으로 스캐닝한다. 도 7b 및 도 7c에 도시된 바와 같이, 블록들 (102 및 103)과 연관된 스캐닝 순서들은, 각각, 수평 스캐닝 순서 및 수직 스캐닝 순서이다. 도 7b 및 도 7c에 있어서의 화살표들에 의해 또한 표시된 바와 같이, 수평 스캐닝 순서는 블록 (102)의 양자화된 변환 계수들을 수평 라인별로 또는 "래스터" 방식으로 스캐닝하지만, 수직 스캐닝 순서는 블록 (103)의 양자화된 변환 계수들을 수직 라인별로 또는 "회전된 래스터" 방식으로 스캐닝한다.

[0099] 다른 예들에 있어서, 상기 설명된 바와 같이, 블록은 블록들 (101, 102, 103 및 104)의 사이즈 미만이거나 초과인 사이즈를 가질 수도 있으며, 더 많거나 더 적은 양자화된 변환 계수들 및 대응하는 블록 포지션들을 포함할 수도 있다. 이들 예들에 있어서, 특정 블록과 연관된 스캐닝 순서는 그 블록의 양자화된 변환 계수들을, 도 7a 내지 도 7d의 8×8 블록들의 예들에서 도시된 바와 실질적으로 유사한 방식으로 스캐닝할 수도 있으며, 예를 들어, 4×4 블록 또는 16×16 블록은 전술된 스캐닝 순서들 중 임의의 순서 이후에 스캐닝될 수도 있다.

[0100] 도 8은 변환 계수 코딩을 위해 서브블록들로 분할된 예시적인 비디오 블록 (110)을 도시한 개념 다이어그램이다. 현재 HM에 있어서, 서브블록 개념이 변환 계수 코딩에 대해 사용된다. 비디오 코더는, 결정된 서브블록 사이즈 초과인 임의의 변환 유닛 (TU)을 서브블록들로 세분할 수도 있다. 예를 들어, 비디오 블록 (110)은 4개의 4×4 서브블록들로 분할된다.

[0101] 도 8의 도시된 예에 있어서, 비디오 코더는 비디오 블록 (110)을 4×4 서브블록들로 분할한다. 다른 예에 있어서, 비디오 코더는 비디오 블록들을, 다른 사이즈들, 예를 들어, 8×8 , 16×16 등의 서브블록들로 분할할 수도 있다. 비디오 코더가 프레임 또는 슬라이스의 모든 TU들에 대해 동일한 서브블록 사이즈를 사용한다면, 서브블록 사이즈들로 달성된 균일성에 기인하여 하드웨어 구현에 있어서 이득들이 달성될 수도 있다. 예를 들어, 모든 프로세싱은 TU 사이즈에 무관하게 그러한 서브블록들로 분할될 수도 있다. 하지만, 균일한 서브블록 사이즈가 본 개시의 기술들을 실행하는데 필수적인 것은 아니다.

[0102] 계수 코딩에 대해, 비디오 블록 (110)의 각각의 4×4 서브블록은, 도 8에 도시된 바와 같이, 대각선 스캐닝 순서를 이용하여 스캐닝될 수도 있다. 일부 예들에 있어서, 비디오 코더는 각각의 서브블록의 변환 계수들을 스캐닝하기 위해 통합된 스캔을 이용할 수도 있다. 이 경우, 동일한 스캔 순서가 유의도 정보, 즉, 유의도 맵, 계수 레벨들, 부호 등에 대해 이용된다. 제 1 예에 있어서, 도 8에 도시된 바와 같이, 비디오 코더는 대각선 스캔을 이용하여 변환 계수들을 스캐닝할 수도 있다. 다른 예에 있어서, 비디오 코더는 도 8에 도시된 순서와 반대인 순서로, 예를 들어, 우하측 코너에서 시작하여 좌상측 코너로 진행하는 역 대각선 스캔으로 변환 계수들을 스캐닝할 수도 있다. 다른 예들에 있어서, 비디오 코더는 지그재그, 수평 또는 수직 스캔을 이용하여 변환 계수들을 스캐닝할 수도 있다. 다른 스캐닝 방향들/배향들이 또한 가능하다.

[0103] 설명의 용이를 위해, 본 개시는 비디오 블록의 서브블록들을, 4×4 서브블록들인 것으로서 설명한다. 하지만, 본 개시의 기술들은 또한, 상이한 사이즈들, 예를 들어, 2×2 , 8×8 , 16×16 등의 서브블록들에 대하여 적용될 수도 있다. 모든 4×4 블록에 대해, significant_coeffgroup_flag 가 코딩되고, 서브블록에서의 적어도 하나의 비-제로 계수가 존재하면 이 플래그는 1로 설정되고; 그렇지 않으면 제로로 설정된다. significant_coeffgroup_flag 가 소정의 서브블록에 대해 비-제로이면, 4×4 서브블록은 역방향 대각선 순서로 스캐닝되며, significant_coeff_flag 는 계수의 유의도를 표시하기 위해 서브블록의 모든 계수에 대해 코딩된다. 계수들의 절대값들, 즉, 계수 레벨들이 또한 코딩된다. 이들 플래그들의 그룹은 비디오 블록에 대한 유의도 맵으로서 지정될 수도 있다. 일부 예에 있어서, 유의도 맵을 명시적으로 시그널링하는 대신, significant_coeffgroup_flag 는 이웃한 4×4 서브블록 플래그들을 이용하여 또는 4×4 서브블록이 마지막 계수 또는 DC 계수를 포함할 경우에 암시적으로 도출될 수도 있다.

[0104] 도 9는 역 대각선 스캐닝 순서를 이용하여 스캐닝된 비디오 블록 (112)에서의 계수들의 유의도 맵을 위한 컨텍스트 모델을 정의하는데 사용된 예시적인 5포인트 서포트를 도시한 개념 다이어그램이다. 컨텍스트 적응 코딩에 대해, 변환 계수들은, 0의 값 또는 1의 값을 갖는 변환 계수의 확률들을 기술하는 컨텍스트 모델에 기초하여 코딩될 수도 있다. 유의도 맵 코딩에 대하여, 컨텍스트 모델은, 특정 변환 계수가 유의한지, 즉, 비

-제로인지 여부의 확률들을 기술한다.

[0105] 유의도 맵 코딩에 대해, 5포인트 서포트 (S) 가 비디오 블록 (112) 의 변환 계수들의 유의도 맵을 코딩하도록 컨텍스트 모델을 정의하는데 사용될 수도 있다. 5포인트 서포트는 "컨텍스트 서포트 이웃" 또는 단순히 "서포트 이웃"으로서 지정될 수도 있다. 즉, 비디오 코더는, 1 또는 제로인 현재 포지션의 유의도의 확률을 결정하기 위한 서포트를 기대할 수도 있다. 컨텍스트 서포트 이웃은, 현재 계수를 코딩하기 위한 컨텍스트들로서 사용될 수도 있는 이웃한 계수들 (예를 들어, 유의도 정보를 포함할 수도 있음) 을 정의한다. 본 개시의 일부 예들에 따르면, 컨텍스트 서포트 이웃은 블록 또는 서브블록 내에서의 상이한 계수 포지션들에 대해 상이할 수도 있다.

[0106] 도 9 에 도시된 예에 있어서, 5포인트 서포트 (S) 는, 원에 의해 둘러싸인 포인트에 의해 나타내어진 현재 또는 "타깃" 포지션에 대해, 정사각형에 의해 둘러싸인 포인트들에 의해 나타내어진다. 컨텍스트 모델 (Ctx) (하기의 수학식(1)) 은 서포트의 모든 포인트에 있어서의 유의도 플래그들의 합으로서 정의될 수도 있으며, 여기서, 유의도 플래그는, 대응하는 변환 계수가 비-제로이면 "1" 로 설정되고, 그렇지 않으면 "0" 으로 설정될 수도 있다.

$$Ctx = \sum_{p \in S} (coef_p != 0) \quad (1)$$

[0107]

[0108] 이에 따라, 유의도 플래그 카운트는 서포트 카디널리티 (support cardinality) 보다 작거나 같을 수 있다.

[0109] 하지만, 도 9 에 도시된 서포트 (S) 는 1 초과의 변환 계수에 대한 컨텍스트 (예를 들어, 변환 계수와 연관된 유의도 정보) 를 병렬로 계산할 경우 ("병렬 유의도 컨텍스트 계산" 또는 단순히 "병렬 컨텍스트 계산"으로서 지정됨) 에 적합하지 않을 수도 있다. 예를 들어, 도 9 에 도시된 서포트 (S) 를 이용하는 것은 유의도 정보에 대한 컨텍스트들을 병렬로 계산하기 위한 비디오 코더의 능력을 방해할 수도 있는데, 왜냐하면 그 서포트 (S) 에서의 모든 데이터는 컨텍스트들의 병렬 계산을 가능케 하기 위해 이용가능해야 (예를 들어, 이미 코딩되어야) 하기 때문이다. 일부 예들에 있어서, 도 10a 에 대하여 하기에 설명된 바와 같이, 코더는, 서포트 (S) 에 있어서의 다른 서포트 엘리먼트에 대한 컨텍스트를 결정하기 전에 코딩을 완료하기 위해 서포트 (S) 에 있어서의 서포트 엘리먼트를 대기하도록 강제될 수도 있다. 이러한 지연은 유의도 정보를 효율적으로 프로세싱하기 위한 비디오 코더의 능력을 감소시킨다.

[0110] 도 10a 및 도 10b 는 5포인트 서포트 내의 컨텍스트 의존성을 도시한 개념 다이어그램들이다. 예를 들어, 동그라미쳐진 포지션에 대한 유의도 컨텍스트를 계산하기 위해, (도 10a 에 도시된) 마름모들로 도시된 서포트 (S) 내에서의 포지션의 유의도 플래그를 파싱하는 것이 필요할 수도 있다. 그러한 파싱은 2개의 계수들의 유의도 컨텍스트들을 병렬로 계산하기 위한 요건이 존재한다면 지연을 도입할 수도 있는데, 왜냐하면 마름모들에서의 엘리먼트들은 스캐닝 순서에 있어서 동그라미쳐진 엘리먼트들 직전에 배치되기 때문이다. 즉, 동그라미쳐진 포지션의 컨텍스트는 마름모에서의 포지션에 의존하기 때문에 마름모에 표시된 포지션과 동시에 계산될 수 없으며, 따라서, 마름모에서의 포지션은 동그라미쳐진 포지션에 대한 컨텍스트를 결정하기 이전에 코딩되어야 한다.

[0111] 이러한 의존성을 해결하기 위해, 특정 엘리먼트들은 서포트 (S) 로부터 제거될 수도 있어서, 서포트가 소위 "홀" (도 10b 에 도시된 비-충진 도트 (삼각형)) 을 갖게 될 수도 있다. 예를 들어, 홀에 있어서의 유의도 플래그는 스kip되고, 컨텍스트 계산을 위해 고려되지 않는다 (즉, 제로인 것으로 가정됨). 이에 따라, 홀 포지션에 있어서의 유의도 플래그를 파싱할 필요가 없다. 5포인트 서포트 형상은 더 우수한 병렬 프로세싱을 허용하기 위해 포지션에 의존한다.

[0112] 도 11 은 비디오 블록의 각각의 영역에 대한 이웃 기반 컨텍스트 또는 포지션 기반 컨텍스트의 예시적인 할당을 도시한 개념 다이어그램이다. 도 11 에 도시된 바와 같이, 하이브리드 타입의 컨텍스트들이 또한 이용될 수도 있으며, 예를 들어, 일부 영역들에 대해, 컨텍스트들은 이웃 기반일 수 있고, 동일한 비디오 블록의 일부 영역들에 대해서는, 고정되거나 포지션 기반일 수 있다. 포지션 기반 접근법의 이점은 컨텍스트 계수별로 계산될 필요가 없고; 영역에 대해 한번 수행될 수 있다는 점이다.

[0113] (x, y) 좌표들을 갖는 계수에 대해, 영역들은 계수 포지션에 따라 정의될 수 있다. 예를 들어, 조건 $(x + y \geq \text{임계치})$ 이 참이면, 이 계수는 영역 (R2) 에 할당되고, 그렇지 않으면, 이 계수는 영역 (R1) 에 있다. 유사하게, 좌표들은 4×4 서브블록들에 기반한 영역들에 할당될 수 있다. (X, Y) 좌표들을 갖는 서브블록에

대해, 영역들은 4×4 서브블록 포지션에 따라 정의될 수 있다. 예를 들어, 조건 ($X + Y \geq$ 임계치) 이 참이면, 이 계수는 영역 (R2)에 할당되고, 그렇지 않으면, 이 계수는 영역 (R1)에 있다. 임계치는 4, 5, 6, 7 또는 8과 동일한 정수와 같은 일부 미리정의된 값으로 고정될 수도 있거나, 또는 비디오 블록, 예를 들어 TU, 사이즈에 의존할 수도 있다.

[0114] 도 12은 본 개시의 기술들에 따른 예시적인 비디오 인코딩 방법을 도시한 플로우차트이다. 도 12의 기술들은 비디오 인코더 (20)의 하나 이상의 하드웨어 유닛들에 의해 실행될 수도 있다. 일 예에 있어서, 엔트로피 인코딩 유닛 (56)은 부호 데이터 은닉과 관련된 기술들 중 일부를 수행할 수도 있다.

[0115] 비디오 인코더 (20)는 제 1 서브블록의 유의도들에 대응하는 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 인코딩할 수도 있다 (170). 제 1 서브블록은 전체 변환 블록의 서브블록일 수도 있다. 비디오 인코더 (20)는, 제 2 유의도 정보에 대한 산술 코딩 동작을 수행함으로써 제 2 서브블록의 유의도에 대응하는 제 2 유의도 정보를 인코딩하고, 그리고 제 1 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들에 기초하여 산술 코딩 동작을 위한 컨텍스트를 결정할 수도 있다 (172).

[0116] 도 13은 본 개시의 기술들에 따른 예시적인 비디오 디코딩 방법을 도시한 플로우차트이다. 도 13의 기술들은 비디오 디코더 (30)의 하나 이상의 하드웨어 유닛들에 의해 실행될 수도 있다. 일 예에 있어서, 엔트로피 디코딩 유닛 (80)은 부호 데이터 은닉과 관련된 기술들 중 일부를 수행할 수도 있다.

[0117] 비디오 디코더 (30)는 제 1 서브블록의 유의도들에 대응하는 잔차 데이터와 연관된 변환 계수들에 대한 제 1 유의도 정보를 디코딩할 수도 있다 (180). 제 1 서브블록은 전체 변환 블록의 서브블록일 수도 있다. 비디오 디코더 (30)는, 제 2 유의도 정보에 대한 산술 코딩 동작을 수행함으로써 제 2 서브블록의 유의도에 대응하는 제 2 유의도 정보를 디코딩하고, 그리고 제 1 서브블록과 동일한 사이즈의 하나 이상의 이웃하는 서브블록들에 기초하여 산술 코딩 동작을 위한 컨텍스트를 결정할 수도 있다 (182).

[0118] 하나 이상의 예들에 있어서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현된다면, 그 기능들은 하나 이상의 명령들 또는 코드로서 컴퓨터 판독가능 매체 상으로 저장 또는 전송되고 하드웨어 기반 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터 판독가능 매체들은 데이터 저장 매체들과 같은 유형의 매체에 대응하는 컴퓨터 판독가능 저장 매체들, 또는 예를 들어, 통신 프로토콜에 따라 일 장소로부터 다른 장소로의 컴퓨터 프로그램의 전송을 용이하게 하는 임의의 매체를 포함하는 통신 매체들을 포함할 수도 있다. 이러한 방식으로, 컴퓨터 판독가능 매체는 일반적으로 (1) 비-일시적인 유형의 컴퓨터 판독가능 저장 매체 또는 (2) 신호 또는 캐리어파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체는 본 개시에서 설명된 기술들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 취출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 가용 매체일 수도 있다. 컴퓨터 프로그램 제품이 컴퓨터 판독가능 매체를 포함할 수도 있다.

[0119] 한정이 아닌 예로서, 그러한 컴퓨터 판독가능 저장 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 저장부, 자기 디스크 저장부 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 원하는 프로그램 코드를 명령들 또는 데이터 구조들의 형태로 저장하는데 이용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 커넥션이 컴퓨터 판독가능 매체로 적절히 명명된다. 예를 들어, 동축 케이블, 광섬유 케이블, 꼬임쌍선, 디지털가입자 라인 (DSL), 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들을 이용하여 웹사이트, 서버, 또는 다른 원격 소스로부터 명령들이 송신된다면, 동축 케이블, 광섬유 케이블, 꼬임쌍선, DSL, 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들은 매체의 정의에 포함된다. 하지만, 컴퓨터 판독가능 저장 매체 및 데이터 저장 매체는 커넥션들, 캐리어파들, 신호들, 또는 다른 일시적 매체를 포함하지 않지만 대신 비-일시적인 유형의 저장 매체로 지향됨을 이해해야 한다. 본 명세서에서 사용된 바와 같은 디스크 (disk) 및 디스크 (disc)는 컴팩트 디스크 (CD), 레이저 디스크, 광학 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크 및 블루레이 디스크를 포함하며, 여기서, 디스크 (disk)는 통상적으로 데이터를 자기적으로 재생하지만 디스크 (disc)는 레이저들을 이용하여 데이터를 광학적으로 재생한다. 상기의 조합들이 또한, 컴퓨터 판독가능 매체의 범위 내에 포함되어야 한다.

[0120] 명령들은 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로프로세서들, 주문형 집적회로들 (ASIC들), 필드 프로그래밍가능 로직 어레이들 (FPGA들), 또는 다른 등가의 집적된 또는 별도의 로직 회로와 같은 하나 이상의 프로세서들에 의해 실행될 수도 있다. 이에 따라, 본 명세서에서 사용된 바와 같은 용어 "프로세서"는 본 명세서에서 설명된 기술들의 구현에 적절한 전술한 구조 또는 임의의 다른 구조 중 임의의 구조를 지칭할 수도 있다. 부가적으로, 일부 양태들에 있어서, 본 명세서에서 설명된 기능은 인코딩 및 디코딩을 위

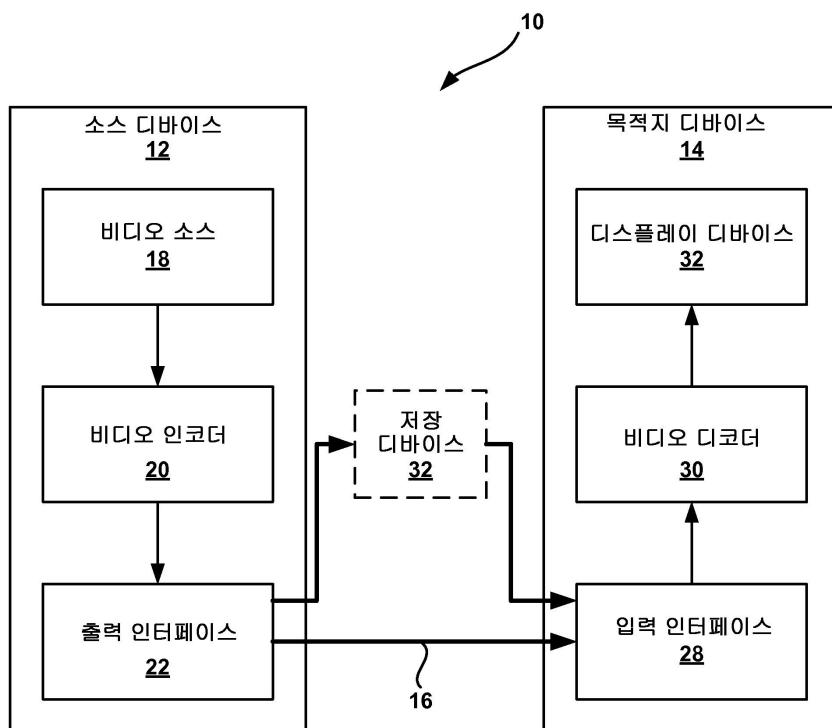
해 구성되고 결합된 코덱에서 통합된 전용 하드웨어 및/또는 소프트웨어 모듈들 내에 제공될 수도 있다. 또 한, 그 기술들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수 있다.

[0121] 본 개시의 기술들은 무선 핸드셋, 집적 회로 (IC) 또는 IC들의 세트 (예를 들어, 칩 세트)를 포함하여 매우 다양한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들, 모듈들 또는 유닛들이 개시된 기술들을 수행하도록 구성된 디바이스들의 기능적 양태들을 강조하기 위해 본 개시에서 설명되지만, 반드시 상이한 하드웨어 유닛들에 의한 실현을 요구하지는 않는다. 오히려, 상기 설명된 바와 같이, 다양한 유닛들은 적절한 소프트웨어 및/또는 펌웨어와 함께 상기 설명된 바와 같은 하나 이상의 프로세서들을 포함하여 코덱 하드웨어 유닛으로 결합되거나 상호운용식 하드웨어 유닛들의 집합에 의해 제공될 수도 있다.

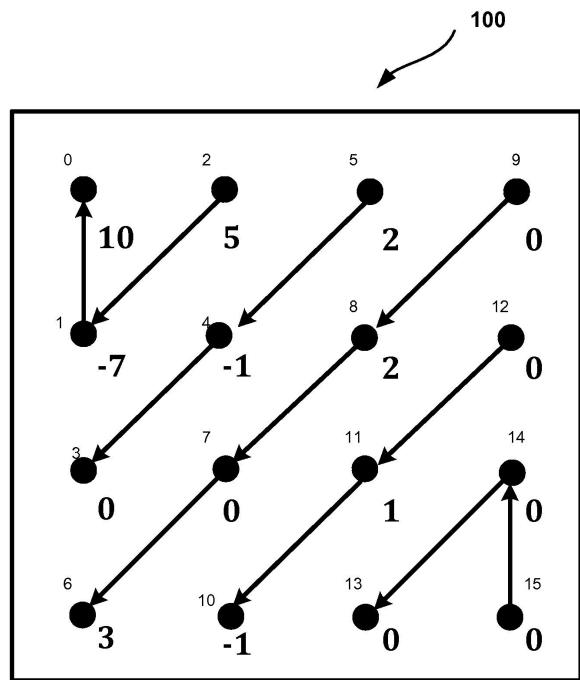
[0122] 다양한 예들이 설명되었다. 이들 및 다른 예들은 다음의 청구항들의 범위 내에 있다.

도면

도면1



도면2



도면3a

2	1	1	0
1	1	0	0
1	0	0	0
0	0	0	0

$\text{CSBF}_r = 0, \text{CSBF}_l = 0$

도면3b

2	2	2	2
1	1	1	1
0	0	0	0
0	0	0	0

$\text{CSBF}_r = 1, \text{CSBF}_l = 0$

도면3c

2	1	0	0
2	1	0	0
2	1	0	0
2	1	0	0

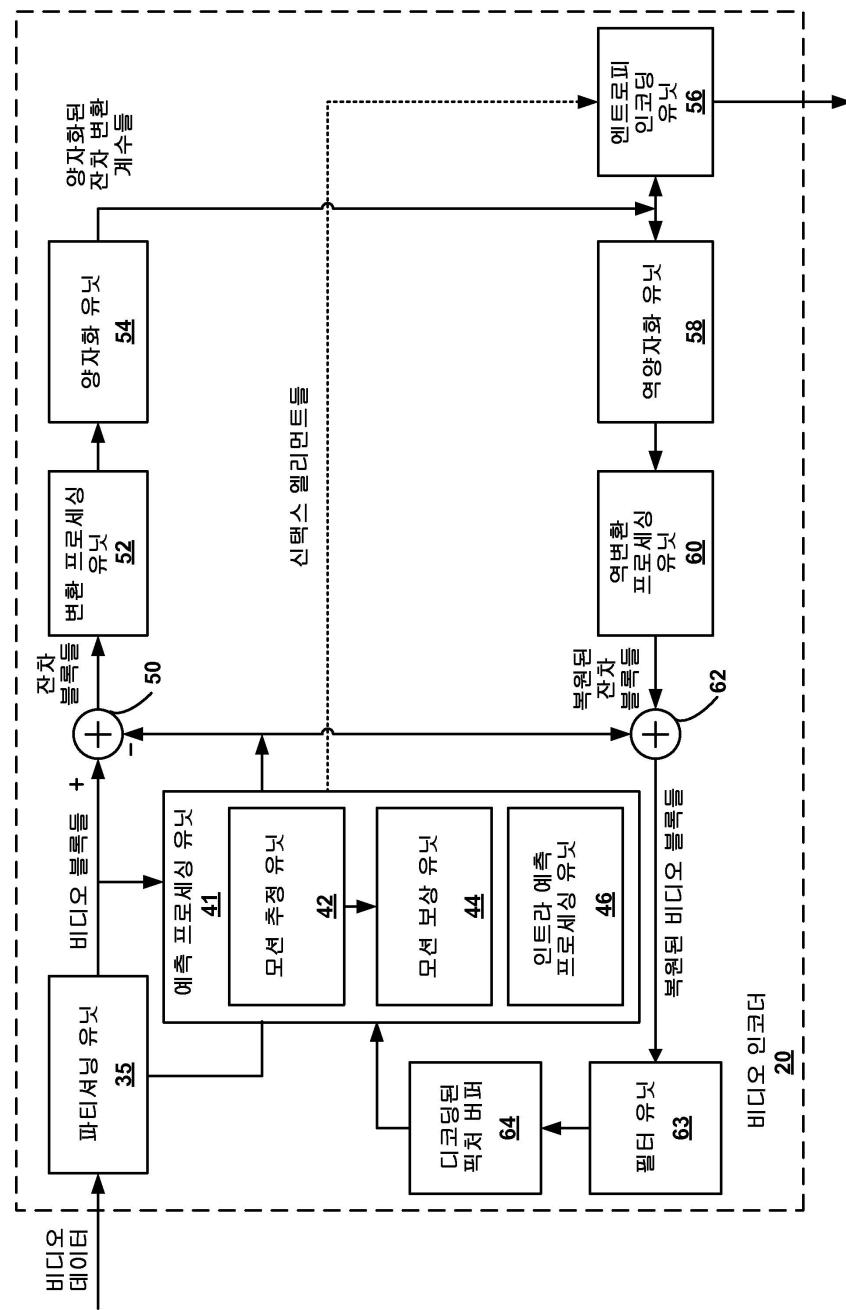
$\mathbf{CSBF_r = 0, CSBF_i = 1}$

도면3d

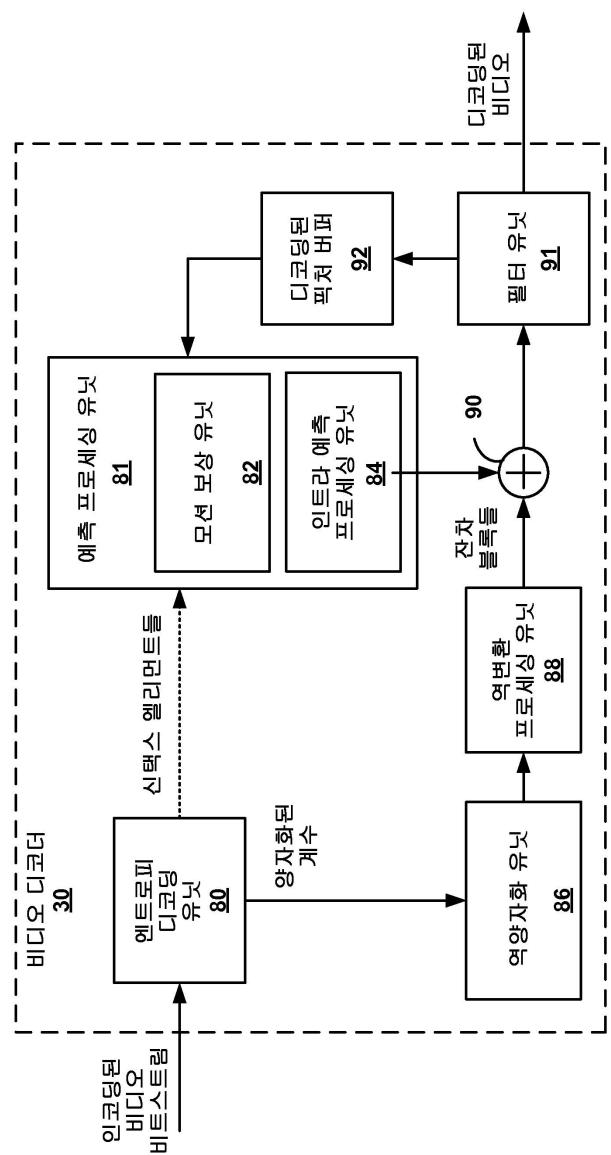
2	2	2	2
2	2	2	2
2	2	2	2
2	2	2	2

$\mathbf{CSBF_r = 1, CSBF_i = 1}$

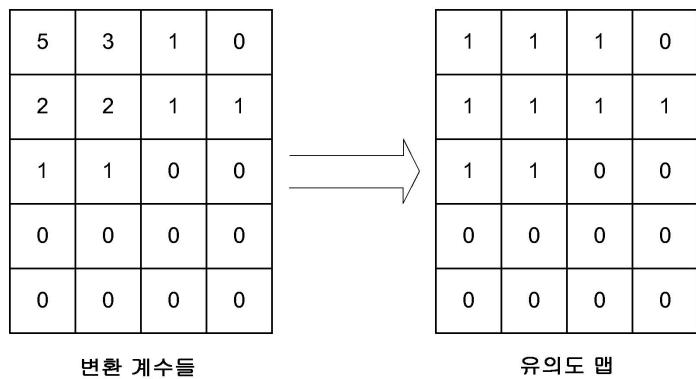
도면4



도면5

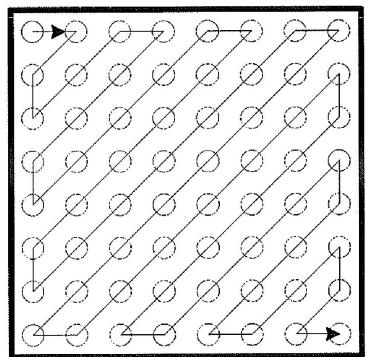


도면6



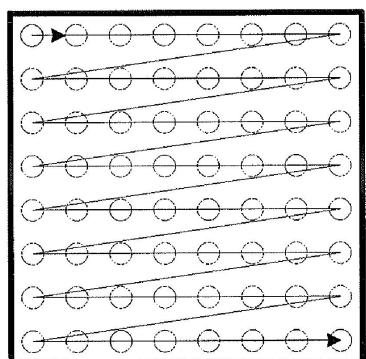
도면7a

101



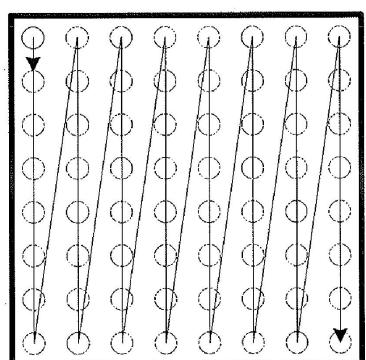
도면7b

102

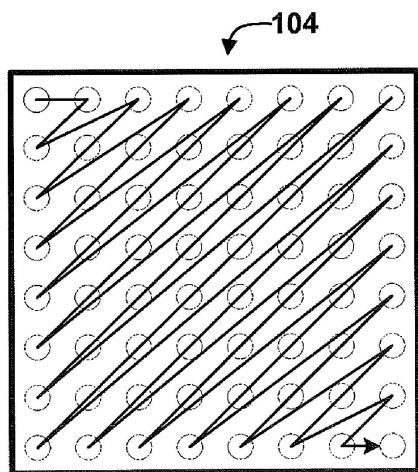


도면7c

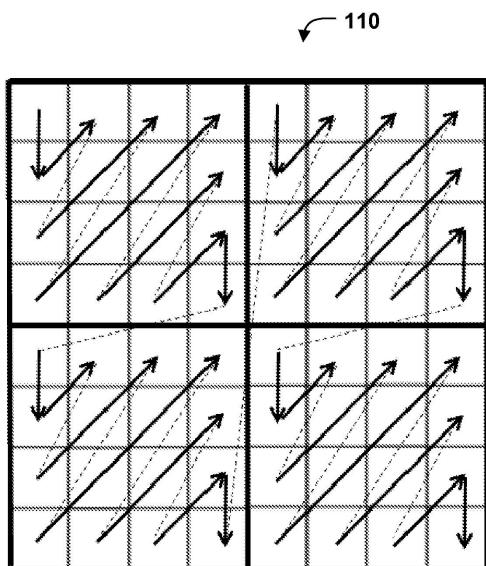
103



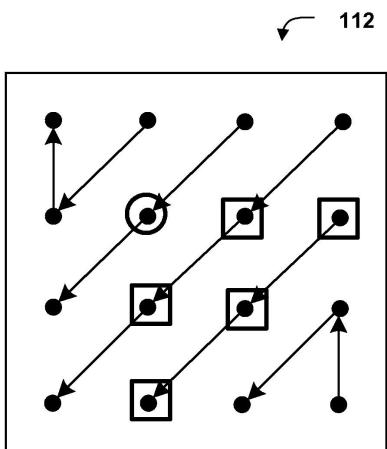
도면7d



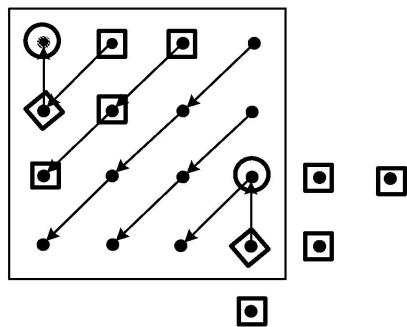
도면8



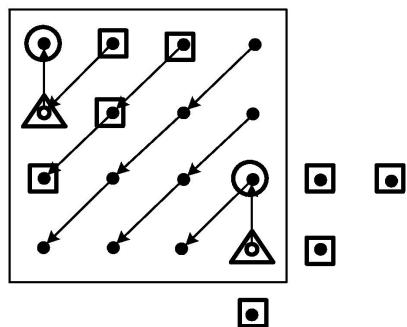
도면9



도면10a



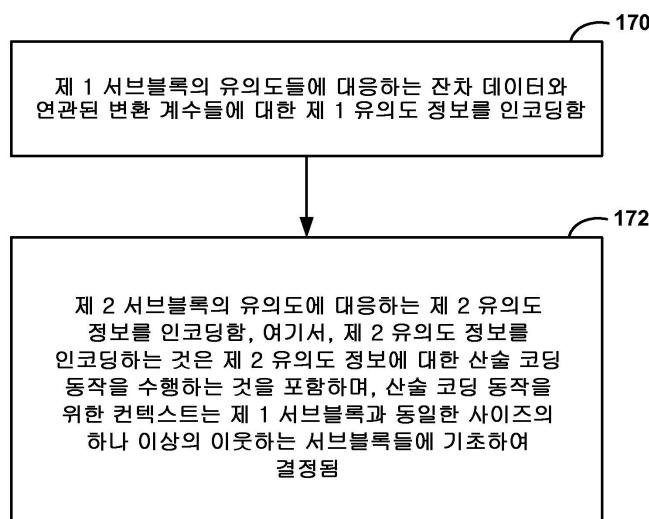
도면10b



도면11

4x4						
이웃 컨텍 스트	기반				포지션	기반 컨텍스트2
	포지션	기반 컨텍스트1				
	포지션	기반 컨텍스트3			포지션	기반 컨텍스트4

도면12



도면13

