



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2016년11월17일
(11) 등록번호 10-1677356
(24) 등록일자 2016년11월11일

(51) 국제특허분류(Int. Cl.)
H04N 19/13 (2014.01)
(21) 출원번호 10-2014-7015290
(22) 출원일자(국제) 2012년11월06일
심사청구일자 2014년06월05일
(85) 번역문제출일자 2014년06월05일
(65) 공개번호 10-2014-0098120
(43) 공개일자 2014년08월07일
(86) 국제출원번호 PCT/US2012/063717
(87) 국제공개번호 WO 2013/070610
국제공개일자 2013년05월16일
(30) 우선권주장
13/669,096 2012년11월05일 미국(US)
(뒷면에 계속)
(56) 선행기술조사문헌
L. Guo, et al. CABAC Contexts Reduction for
Last Position Coding. JCT-VC of ITU-T and
ISO/IEC. JCTVC-H0537 Ver.5, Feb. 3, 2012,
pp.1-5
C. Auyeung, et al. Context reduction of the
last transform position in JCTVC-D262 for
CE11.1. JCT-VC of ITU-T and ISO/IEC.
JCTVC-E344 Ver.4, Mar. 18, 2011, pp.1-6
J. Sole, et al. Parallel Context Processing
for the significance map in high coding
efficiency. JCT-VC of ITU-T and ISO/IEC.
JCTVC-D262 Ver.2, Jan. 21, 2011, pp.1-4
V. Seregin, et al. Binarisation modification
for last position coding. JCT-VC of ITU-T and
ISO/IEC. JCTVC-F375 Ver.3, Jul. 16, 2011,
pp.1-10

(73) 특허권자
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하
우스 드라이브 5775
(72) 발명자
귀 리웨이
미국 92121-1714 캘리포니아주 샌디에고 모어하
우스 드라이브 5775
치엔 웨이-정
미국 92121-1714 캘리포니아주 샌디에고 모어하
우스 드라이브 5775
카르체비츠 마르타
미국 92121-1714 캘리포니아주 샌디에고 모어하
우스 드라이브 5775
(74) 대리인
특허법인코리어나

전체 청구항 수 : 총 29 항

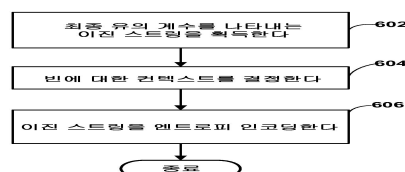
심사관 : 이상래

(54) 발명의 명칭 최종 유의 계수 위치 코딩을 위한 컨텍스트 최적화

(57) 요약

비디오 인코더는 비디오 블록 내 최종 유의 계수의 위치를 나타내는 이진 스트링을 인코딩하도록 구성된다. 비디오 디코더는 인코딩된 이진 스트링을 디코딩하도록 구성된다. 스트링은 컨텍스트 적응 이진 산술 코딩(CABAC)을 이용하여 코딩될 수도 있다. 이진 스트링의 이진 인덱스들이 컨텍스트에 할당될 수도 있다. 컨텍스트는 맵핑 함수에 따라 결정될 수도 있다. 컨텍스트는 각각의 인덱스가 상이한 블록 사이즈와 연관되는 하나 이상의 이진 인덱스들에 할당될 수도 있다. 16x16 비디오 블록의 최종 이진 인덱스는 32x32 비디오 블록의 최종 이진 인덱스와 컨텍스트를 공유할 수도 있다.

대표도



(30) 우선권주장

61/557,317	2011년11월08일	미국(US)
61/561,909	2011년11월20일	미국(US)
61/588,579	2012년01월19일	미국(US)
61/596,049	2012년02월07일	미국(US)

명세서

청구범위

청구항 1

비디오 블록에 대한 변환 계수들을 인코딩하는 방법으로서,

상기 비디오 블록과 연관된 변환 계수들의 변환 블록 내 최종 유의 계수의 위치를 나타내는 이진 스트링을 결정하는 단계로서, 상기 이진 스트링은 접두사 부분 및 접미사 부분을 포함하는, 이진 스트링을 결정하는 단계;

상기 변환 블록의 사이즈에 기초하여 상기 이진 스트링의 상기 접두사 부분 내의 이진 인덱스에 대한 컨텍스트를 결정하는 단계로서, 상기 컨텍스트는, 다음 함수, 즉 $ctx_index = (n \gg (\log_2(T)-2)) + 15$ 로 정의되는 컨텍스트 인덱스 (ctx_index) 에 따라서 적어도 2개의 이진 인덱스들에 할당되며, n 은 이진 인덱스이고, \gg 는 이진 시프트 연산 (shift operation) 이고, T 는 상기 변환 블록의 치수이고, 상기 적어도 2개의 이진 인덱스들은 상이한 변환 블록 사이즈들과 연관되는, 상기 결정하는 단계; 및

결정된 상기 컨텍스트에 적어도 부분적으로 기초하여, 컨텍스트 적응 이진 산술 코딩 (CABAC) 을 이용하여 상기 이진 스트링을 인코딩하는 단계를 포함하는, 변환 블록에 대한 변환 계수들을 인코딩하는 방법.

청구항 2

삭제

청구항 3

제 1 항에 있어서,

상기 컨텍스트는 16x16 변환 블록의 최종 이진 인덱스 및 32x32 변환 블록의 최종 이진 인덱스에 할당되는, 변환 블록에 대한 변환 계수들을 인코딩하는 방법.

청구항 4

제 3 항에 있어서,

제 2 컨텍스트는 상기 16x16 변환 블록의 인접한 이진 인덱스들에 할당되는, 변환 블록에 대한 변환 계수들을 인코딩하는 방법.

청구항 5

제 1 항에 있어서,

상기 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 이진 스트링을 인코딩하는 단계는, 상기 이진 스트링의 값에 기초하여 상기 컨텍스트를 업데이트하는 단계를 포함하고,

상기 변환 블록은 제 1 변환 블록이고,

상기 방법은,

제 2 변환 블록 내 최종 유의 계수의 위치를 나타내는 제 2 이진 스트링을 결정하는 단계로서, 상기 제 1 변환 블록 및 상기 제 2 변환 블록은 상이한 사이즈들인, 상기 제 2 이진 스트링을 결정하는 단계; 및

업데이트된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 제 2 이진 스트링을 인코딩하는 단계를 더 포함하는, 변환 블록에 대한 변환 계수들을 인코딩하는 방법.

청구항 6

비디오 블록에 대한 변환 계수들을 디코딩하는 방법으로서,

상기 비디오 블록과 연관된 변환 계수들의 변환 블록 내 최종 유의 계수의 위치를 나타내는 인코딩된 이진 스트링을 획득하는 단계로서, 상기 인코딩된 이진 스트링은 접두사 부분과 접미사 부분을 포함하고, 상기 인코딩된

이진 스트링은 컨텍스트 적응 이진 산술 코딩 (CABAC) 을 이용하여 인코딩되는, 상기 획득하는 단계;

상기 변환 블록의 사이즈에 기초하여 상기 인코딩된 이진 스트링의 상기 접두사 부분 내의 이진 인덱스에 대한 컨텍스트를 결정하는 단계로서, 상기 컨텍스트는, 다음 함수, 즉 $ctx_index = (n \gg (\log_2(T)-2)) + 15$ 로 정의되는 컨텍스트 인덱스 (ctx_index) 에 따라서 적어도 2개의 이진 인덱스들에 할당되며, n 은 이진 인덱스이고, \gg 는 이진 시프트 연산 (shift operation) 이고, T 는 상기 변환 블록의 치수이고, 상기 적어도 2개의 이진 인덱스들은 상이한 변환 블록 사이즈들과 연관되는, 상기 결정하는 단계;

결정된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 인코딩된 이진 스트링을 디코딩하는 단계; 및

상기 디코딩된 이진 스트링에 기초하여 상기 비디오 블록과 연관된 상기 변환 계수들의 변환 블록에 상기 최종 유의 계수의 위치를 결정하는 단계를 포함하는, 변환 계수들을 디코딩하는 방법.

청구항 7

삭제

청구항 8

제 6 항에 있어서,

상기 컨텍스트는 16x16 변환 블록의 최종 이진 인덱스 및 32x32 변환 블록의 최종 이진 인덱스에 할당되는, 변환 계수들을 디코딩하는 방법.

청구항 9

제 8 항에 있어서,

제 2 컨텍스트는 상기 16x16 변환 블록의 인접한 이진 인덱스들에 할당되는, 변환 계수들을 디코딩하는 방법.

청구항 10

제 6 항에 있어서,

상기 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 인코딩된 이진 스트링을 디코딩하는 단계는, 상기 인코딩된 이진 스트링의 값에 기초하여 상기 컨텍스트를 업데이트하는 단계를 포함하고,

상기 변환 블록은 제 1 변환 블록이고,

상기 방법은,

제 2 변환 블록 내 최종 유의 계수의 위치를 나타내는 제 2 인코딩된 이진 스트링을 획득하는 단계로서, 상기 제 1 변환 블록 및 상기 제 2 변환 블록은 상이한 사이즈들인, 상기 제 2 인코딩된 이진 스트링을 획득하는 단계; 및

업데이트된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 제 2 인코딩된 이진 스트링을 디코딩하는 단계를 더 포함하는, 변환 계수들을 디코딩하는 방법.

청구항 11

비디오 블록에 대한 변환 계수들을 인코딩하도록 구성되는 장치로서,

상기 장치는,

변환 계수들의 변환 블록 내 최종 유의 계수의 위치를 나타내는 이진 스트링을 결정하는 수단으로서, 상기 이진 스트링은 접두사 부분 및 접미사 부분을 포함하는, 이진 스트링을 결정하는 수단;

상기 변환 블록의 사이즈에 기초하여 상기 이진 스트링의 상기 접두사 부분 내의 이진 인덱스에 대한 컨텍스트를 결정하는 수단으로서,

상기 컨텍스트는, 다음 함수, 즉 $ctx_index = (n \gg (\log_2(T)-2)) + 15$ 로 정의되는 컨텍스트 인덱스 (ctx_index) 에 따라서 적어도 2개의 이진 인덱스들에 할당되며, n 은 이진 인덱스이고, \gg 는 이진 시프트 연

산 (shift operation) 이고, T 는 상기 변환 블록의 치수이고, 상기 적어도 2개의 이진 인덱스들은 상이한 변환 블록 사이즈들과 연관되는, 상기 결정하는 수단; 및

결정된 상기 컨텍스트에 적어도 부분적으로 기초하여, 컨텍스트 적응 이진 산술 코딩 (CABAC) 을 이용하여 상기 이진 스트링을 인코딩하는 수단을 포함하는, 변환 블록에 대한 변환 계수들을 인코딩하도록 구성되는 장치.

청구항 12

삭제

청구항 13

제 11 항에 있어서,

상기 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 이진 스트링을 인코딩하는 수단은, 상기 이진 스트링의 값에 기초하여 상기 컨텍스트를 업데이트하는 수단을 포함하며,

상기 변환 블록은 제 1 변환 블록이고,

상기 장치는,

제 2 변환 블록 내 최종 유의 계수의 위치를 나타내는 제 2 이진 스트링을 결정하는 수단으로서, 상기 제 1 변환 블록 및 상기 제 2 변환 블록은 상이한 사이즈들인, 상기 제 2 이진 스트링을 결정하는 수단; 및

업데이트된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 제 2 이진 스트링을 엔트로피 코딩하는 수단을 더 포함하는, 변환 블록에 대한 변환 계수들을 인코딩하도록 구성되는 장치.

청구항 14

비디오 블록에 대한 변환 계수들을 디코딩하도록 구성되는 장치로서,

상기 장치는,

상기 비디오 블록과 연관된 변환 계수들의 변환 블록 내 최종 유의 계수의 위치를 나타내는 인코딩된 이진 스트링을 획득하는 수단으로서, 상기 인코딩된 이진 스트링은 접두사 부분과 접미사 부분을 포함하고, 상기 인코딩된 이진 스트링은 컨텍스트 적응 이진 산술 코딩 (CABAC) 을 이용하여 인코딩되는, 상기 획득하는 수단;

상기 변환 블록의 사이즈에 기초하여 상기 인코딩된 이진 스트링의 상기 접두사 부분 내의 이진 인덱스에 대한 컨텍스트를 결정하는 수단으로서, 상기 컨텍스트는, 다음 함수, 즉 $ctx_index = (n \gg (\log_2(T)-2)) + 15$ 로 정의되는 컨텍스트 인덱스 (ctx_index) 에 따라서 적어도 2개의 이진 인덱스들에 할당되며, n 은 이진 인덱스이고, \gg 는 이진 시프트 연산 (shift operation) 이고, T 는 상기 변환 블록의 치수이고, 상기 적어도 2개의 이진 인덱스들은 상이한 변환 블록 사이즈들과 연관되는, 상기 결정하는 수단;

결정된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 인코딩된 이진 스트링을 디코딩하는 수단; 및

상기 디코딩된 이진 스트링에 기초하여 상기 비디오 블록과 연관된 상기 변환 계수들의 변환 블록에 상기 최종 유의 계수의 위치를 결정하는 수단을 포함하는, 변환 블록에 대한 변환 계수들을 디코딩하도록 구성되는 장치.

청구항 15

삭제

청구항 16

제 14 항에 있어서,

상기 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 인코딩된 이진 스트링을 디코딩하는 수단은, 상기 인코딩된 이진 스트링의 값에 기초하여 상기 컨텍스트를 업데이트하는 수단을 포함하고,

상기 변환 블록은 제 1 변환 블록이고,

상기 장치는,

제 2 변환 블록 내 최종 유의 계수의 위치를 나타내는 제 2 인코딩된 이진 스트링을 획득하는 수단으로서, 상기 제 1 변환 블록 및 상기 제 2 변환 블록은 상이한 사이즈들인, 상기 제 2 인코딩된 이진 스트링을 획득하는 수단; 및

업데이트된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 제 2 인코딩된 이진 스트링을 디코딩하는 수단을 더 포함하는, 변환 블록에 대한 변환 계수들을 디코딩하도록 구성되는 장치.

청구항 17

비디오 블록에 대한 변환 계수들을 인코딩하도록 구성된 디바이스로서,

상기 디바이스는,

상기 변환 계수들을 저장하도록 구성된 메모리; 및

비디오 인코더를 포함하고,

상기 비디오 인코더는,

상기 비디오 블록과 연관된 변환 계수들의 변환 블록 내 최종 유의 계수의 위치를 나타내는 이진 스트링을 결정하되, 상기 이진 스트링은 접두사 부분 및 접미사 부분을 포함하고;

상기 변환 블록의 사이즈에 기초하여 상기 이진 스트링의 상기 접두사 부분 내의 이진 인덱스에 대한 컨텍스트를 결정하고, 상기 컨텍스트는, 다음 함수, 즉 $ctx_index = (n \gg (\log_2(T)-2)) + 15$ 로 정의되는 컨텍스트 인덱스 (ctx_index) 에 따라서 적어도 2개의 이진 인덱스들에 할당되며, n 은 이진 인덱스이고, \gg 는 이진 시프트 연산 (shift operation) 이고, T 는 상기 변환 블록의 치수이고, 상기 적어도 2개의 이진 인덱스들은 상이한 변환 블록 사이즈들과 연관되며; 그리고

결정된 상기 컨텍스트에 적어도 부분적으로 기초하여, 컨텍스트 적응 이진 산술 코딩 (CABAC) 을 이용하여 상기 이진 스트링을 인코딩하도록 구성되는, 디바이스.

청구항 18

삭제

청구항 19

제 17 항에 있어서,

상기 컨텍스트는 16x16 변환 블록의 최종 이진 인덱스 및 32x32 변환 블록의 최종 이진 인덱스에 할당되는, 디바이스.

청구항 20

제 19 항에 있어서,

제 2 컨텍스트는 상기 16x16 변환 블록의 인접한 이진 인덱스들에 할당되는, 디바이스.

청구항 21

제 17 항에 있어서,

상기 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 이진 스트링을 인코딩하는 것은, 상기 이진 스트링의 값에 기초하여 상기 컨텍스트를 업데이트하는 것을 포함하고,

상기 변환 블록은 제 1 변환 블록이고,

상기 비디오 인코더는,

제 2 변환 블록 내 최종 유의 계수의 위치를 나타내는 제 2 이진 스트링을 획득하고, 상기 제 1 변환 블록 및 상기 제 2 변환 블록은 상이한 사이즈들이며; 그리고

업데이트된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 제 2 이진 스트링을 인코딩

하도록 추가로 구성되는, 디바이스.

청구항 22

비디오 블록에 대한 변환 계수들을 디코딩하도록 구성된 디바이스로서,

상기 디바이스는,

상기 변환 계수들을 저장하도록 구성된 메모리; 및

비디오 디코더를 포함하고,

상기 비디오 디코더는,

상기 비디오 블록과 연관된 변환 계수들의 변환 블록 내 최종 유의 계수의 위치를 나타내는 인코딩된 이진 스트링을 획득하고, 상기 인코딩된 이진 스트링은 접두사 부분 및 접미사 부분을 포함하고, 상기 인코딩된 이진 스트링은 컨텍스트 적응 이진 산술 코딩 (CABAC) 을 이용하여 인코딩되며;

상기 변환 블록의 사이즈에 기초하여 상기 인코딩된 이진 스트링의 상기 접두사 부분의 이진 인덱스에 대한 컨텍스트를 결정하고, 상기 컨텍스트는, 다음 함수, 즉 $ctx_index = (n \gg (\log_2(T)-2)) + 15$ 로 정의되는 컨텍스트 인덱스 (ctx_index) 에 따라서 적어도 2개의 이진 인덱스들에 할당되며, n 은 이진 인덱스이고, \gg 는 이진 시프트 연산 (shift operation) 이고, T 는 상기 변환 블록의 치수이고, 상기 적어도 2개의 이진 인덱스들은 상기 변환 블록 사이즈들과 연관되고;

결정된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 인코딩된 이진 스트링을 디코딩하고;

상기 디코딩된 이진 스트링에 기초하여 상기 비디오 블록과 연관된 상기 변환 계수들의 변환 블록에 상기 최종 유의 계수의 위치를 결정하도록 구성되는, 디바이스.

청구항 23

삭제

청구항 24

제 22 항에 있어서,

상기 컨텍스트는 16x16 변환 블록의 최종 이진 인덱스 및 32x32 변환 블록의 최종 이진 인덱스에 할당되는, 디바이스.

청구항 25

제 24 항에 있어서,

제 2 컨텍스트는 상기 16x16 변환 블록의 인접한 이진 인덱스들에 할당되는, 디바이스.

청구항 26

제 22 항에 있어서,

상기 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 인코딩된 이진 스트링을 디코딩하는 것은, 상기 인코딩된 이진 스트링의 값에 기초하여 상기 컨텍스트를 업데이트하는 것을 포함하고,

상기 변환 블록은 제 1 변환 블록이고,

상기 비디오 디코더는,

제 2 변환 블록 내 최종 유의 계수의 위치를 나타내는 제 2 인코딩된 이진 스트링을 획득하고, 상기 제 1 변환 블록 및 상기 제 2 변환 블록은 상이한 사이즈들이며; 그리고

업데이트된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 제 2 인코딩된 이진 스트링을 디코딩하도록 추가로 구성되는, 디바이스.

청구항 27

명령들이 저장된 비밀시성 컴퓨터-판독가능 저장 매체로서,

상기 명령들은, 실행 시, 비디오 인코딩 디바이스의 하나 이상의 프로세서들로 하여금,

비디오 블록과 연관된 변환 계수들의 변환 블록 내 최종 유의 계수의 위치를 나타내는 이진 스트링을 결정하되, 상기 이진 스트링은 접두사 부분 및 접미사 부분을 포함하고;

상기 변환 블록의 사이즈에 기초하여 상기 이진 스트링의 접두사 부분 내의 이진 인덱스에 대한 컨텍스트를 결정하고, 상기 컨텍스트는, 다음 함수, 즉 $ctx_index = (n \gg (\log_2(T)-2)) + 15$ 로 정의되는 컨텍스트 인덱스 (ctx_index) 에 따라서 적어도 2개의 이진 인덱스들에 할당되며, n 은 이진 인덱스이고, \gg 는 이진 시프트 연산 (shift operation) 이고, T 는 상기 변환 블록의 치수이고, 상기 적어도 2개의 이진 인덱스들은 상이한 변환 블록 사이즈들과 연관되며; 그리고

결정된 상기 컨텍스트에 적어도 부분적으로 기초하여, 컨텍스트 적응 이진 산술 코딩 (CABAC) 을 이용하여 상기 이진 스트링을 인코딩하게 하는, 비밀시성 컴퓨터-판독가능 저장 매체.

청구항 28

삭제

청구항 29

제 27 항에 있어서,

상기 컨텍스트는 16x16 변환 블록의 최종 이진 인덱스 및 32x32 변환 블록의 최종 이진 인덱스에 할당되는, 비밀시성 컴퓨터-판독가능 저장 매체.

청구항 30

제 29 항에 있어서,

제 2 컨텍스트는 상기 16x16 변환 블록의 인접한 이진 인덱스들에 할당되는, 비밀시성 컴퓨터-판독가능 저장 매체.

청구항 31

제 27 항에 있어서,

상기 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 이진 스트링을 인코딩하는 것은, 상기 이진 스트링의 값에 기초하여 상기 컨텍스트를 업데이트하는 것을 포함하고,

상기 변환 블록은 제 1 변환 블록이고,

상기 명령들은, 실행 시, 추가로, 비디오 인코딩 디바이스의 하나 이상의 프로세서들로 하여금,

제 2 변환 블록 내 최종 유의 계수의 위치를 나타내는 제 2 이진 스트링을 결정하고, 상기 제 1 변환 블록 및 상기 제 2 변환 블록은 상이한 사이즈들이며; 그리고

업데이트된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 제 2 이진 스트링을 인코딩하게 하는, 비밀시성 컴퓨터-판독가능 저장 매체.

청구항 32

명령들이 저장된 비밀시성 컴퓨터-판독가능 저장 매체로서,

상기 명령들은, 실행 시, 비디오 디코딩 디바이스의 하나 이상의 프로세서들로 하여금,

비디오 블록과 연관된 변환 계수들의 변환 블록 내 최종 유의 계수의 위치를 나타내는 인코딩된 이진 스트링을 획득하고, 상기 인코딩된 이진 스트링은 접두사 부분 및 접미사 부분을 포함하고, 상기 인코딩된 이진 스트링은 컨텍스트 적응 이진 산술 코딩 (CABAC) 을 이용하여 인코딩되며;

상기 변환 블록의 사이즈에 기초하여 상기 인코딩된 이진 스트링의 상기 접두사 부분 내의 이진 인덱스에 대한 컨텍스트를 결정하고, 상기 컨텍스트는, 다음 함수, 즉 $ctx_index = (n \gg (\log_2(T)-2)) + 15$ 로 정의되는 컨텍스트 인덱스 (ctx_index) 에 따라서 적어도 2개의 이진 인덱스들에 할당되며, n 은 이진 인덱스이고, \gg 는 이진 시프트 연산 (shift operation) 이고, T 는 상기 변환 블록의 치수이고, 상기 적어도 2개의 이진 인덱스들은 상이한 변환 블록 사이즈들과 연관되며;

결정된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 인코딩된 이진 스트링을 디코딩하고;

상기 디코딩된 이진 스트링에 기초하여 상기 비디오 블록과 연관된 상기 변환 계수들의 변환 블록에 상기 최종 유의 계수의 위치를 결정하게 하는, 비일시성 컴퓨터-판독가능 저장 매체.

청구항 33

삭제

청구항 34

제 32 항에 있어서,

상기 컨텍스트는 16x16 변환 블록의 최종 이진 인덱스 및 32x32 변환 블록의 최종 이진 인덱스에 할당되는, 비일시성 컴퓨터-판독가능 저장 매체.

청구항 35

제 34 항에 있어서,

제 2 컨텍스트는 상기 16x16 변환 블록의 인접한 이진 인덱스들에 할당되는, 비일시성 컴퓨터-판독가능 저장 매체.

청구항 36

제 32 항에 있어서,

상기 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 인코딩된 이진 스트링을 디코딩하는 것은, 상기 인코딩된 이진 스트링의 값에 기초하여 상기 컨텍스트를 업데이트하는 것을 포함하고,

상기 변환 블록은 제 1 변환 블록이고,

상기 명령들은, 실행 시, 추가로, 비디오 인코딩 디바이스의 하나 이상의 프로세서들로 하여금,

제 2 변환 블록 내 최종 유의 계수의 위치를 나타내는 제 2 인코딩된 이진 스트링을 획득하고, 상기 제 1 변환 블록 및 상기 제 2 변환 블록은 상이한 사이즈들이며; 그리고

업데이트된 상기 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 상기 제 2 인코딩된 이진 스트링을 디코딩하게 하는, 비일시성 컴퓨터-판독가능 저장 매체.

청구항 37

제 22 항에 있어서,

상기 디바이스는,

무선 통신 디바이스;

마이크로프로세서; 또는

집적 회로 중 하나 이상을 포함하며,

상기 무선 통신 디바이스는, 상기 비디오 블록을 포함하는 화상을 디스플레이하도록 구성된 디스플레이 또는 상기 비디오 블록을 포함하는 화상을 캡처하도록 구성된 카메라 중 적어도 하나를 포함하는, 디바이스.

발명의 설명

기술 분야

관련 출원들

본 출원은, 2011년 11월 8일자에 출원된 미국 가특허출원 번호 제 61/557,317호; 2011년 11월 20일자에 출원된 미국 가특허출원 번호 제 61/561,909호;

2012년 1월 19일자에 출원된 미국 가특허 출원번호 제 61/588,579호; 및 2012년 2월 7일자에 출원된 미국 가특허 출원번호 제 61/596,049호의 이익을 주장하며, 이들 각각은 그 각각의 전체적으로 참고로 포함된다.

기술 분야

본 개시물은 비디오 코딩에 관한 것으로, 좀더 구체적으로는, 변환 계수들을 코딩하는 기법들에 관한 것이다.

배경 기술

디지털 비디오 능력들은 디지털 텔레비전, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인 휴대정보 단말기들 (PDAs), 랩탑 또는 데스크탑 컴퓨터들, 태블릿 컴퓨터들, e-북 리더들, 디지털 카메라들, 디지털 리코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 소위 "스마트폰들", 원격 화상회의 디바이스들, 비디오 스트리밍 디바이스들 등을 포함한, 광범위한 디바이스들에 포함될 수 있다. 디지털 비디오 디바이스들은 MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, 파트 10, AVC (Advanced Video Coding), 현재 개발중인 HEVC (High Efficiency Video Coding) 표준, 및 이런 표준들의 확장판들에 의해 정의된 표준들에서 설명되는 비디오 코딩 기법들과 같은, 비디오 압축 기법들을 구현한다. 비디오 디바이스들은 이런 비디오 압축 기법들을 구현함으로써, 디지털 비디오 정보를 좀더 효율적으로 송신, 수신, 인코딩, 디코딩, 및/또는 저장할 수도 있다.

비디오 압축 기법들은 비디오 시퀀스들에 고유한 리던던시를 감소시키거나 또는 제거하기 위해 공간 (인트라-화상) 예측 및/또는 시간 (인터-화상) 예측을 수행한다. 블록-기반의 비디오 코딩에 있어, 비디오 슬라이스 (즉, 비디오 프레임 또는 비디오 프레임의 부분) 는 비디오 블록들로 파티셔닝될 수도 있으며, 이 비디오 블록들은 또한 트리블록들, 코딩 유닛들 (CU들) 및/또는 코딩 노드들로서 지칭될 수도 있다. 화상의 인트라-코딩된 (I) 슬라이스에서 비디오 블록들은 동일한 화상에서 이웃하는 블록들에서의 참조 샘플들에 대한 공간 예측을 이용하여 인코딩된다. 화상의 인터-코딩된 (P 또는 B) 슬라이스에서 비디오 블록들은 동일한 화상에서 이웃하는 블록들에서의 참조 샘플들에 대한 공간 예측 또는 다른 참조 화상들에서의 참조 샘플들에 대한 시간 예측을 이용할 수도 있다. 화상들은 프레임들로 지칭될 수 있으며, 참조 화상들은 참조 프레임들로서 지칭될 수도 있다.

공간 또는 시간 예측은 코딩되는 블록에 대한 예측 블록을 초대한다. 잔여 데이터는 코딩되는 원래 블록과 예측 블록 사이의 픽셀 차이들을 나타낸다. 인터-코딩된 블록은 예측 블록을 형성하는 참조 샘플들의 블록을 가리키는 모션 벡터, 및 코딩된 블록과 예측 블록 사이의 차이를 나타내는 잔여 데이터에 따라서 인코딩된다. 인트라-코딩된 블록은 인트라-코딩 모드 및 잔여 데이터에 따라서 인코딩된다. 추가적인 압축을 위해, 잔여 데이터는 픽셀 도메인으로부터 변환 도메인으로 변환되어, 잔여 변환 계수들을 초대할 수도 있으며, 이 잔여 변환 계수는 그후 양자화될 수도 있다. 처음에 2차원 어레이로 배열된, 양자화된 변환 계수들은 변환 계수들의 1차원 벡터를 발생하기 위해 스캐닝될 수도 있으며, 엔트로피 코딩이 더욱 더 많은 압축을 달성하기 위해 적용될 수도 있다.

발명의 내용

과제의 해결 수단

일반적으로, 본 개시물은 비디오 데이터를 코딩하는 기법들을 설명한다. 특히, 본 개시물은 변환 계수들을 코딩하는 기법들을 설명한다.

본 개시물의 일 예에서, 변환 계수들을 인코딩하는 방법은 비디오 블록과 연관되는 변환 계수들의 블록 내 최종 유의 계수의 위치를 나타내는 이진 스트링을 획득하는 단계; 비디오 블록 사이즈에 기초하여, 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하는 단계로서, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관되는, 상기 결정하는 단계; 및 결정된 컨텍스트에 적어도 부분적으로 기초하여, 컨텍스트 적용 이진 산술 코딩 (CABAC) 을 이용하여 이진 스트링

을 인코딩하는 단계를 포함한다.

- [0011] 본 개시물의 또 다른 예에서, 변환 계수들을 디코딩하는 방법은 비디오 블록과 연관되는 변환 계수들의 블록 내 최종 유의 계수의 위치를 나타내는 인코딩된 이진 스트링을 획득하는 단계로서, 상기 인코딩된 이진 스트링은 CABAC 를 이용하여 인코딩되는, 상기 획득하는 단계; 비디오 블록 사이즈에 기초하여, 인코딩된 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하는 단계로서, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관되는, 상기 결정하는 단계; 및 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 그 인코딩된 이진 스트링을 디코딩하는 단계를 포함한다.
- [0012] 본 개시물의 또 다른 예에서, 비디오 인코딩 프로세스에서 변환 계수들을 인코딩하도록 구성되는 장치는 비디오 블록과 연관되는 변환 계수들의 블록 내 최종 유의 계수의 위치를 나타내는 이진 스트링을 획득하는 수단; 비디오 블록 사이즈에 기초하여, 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하는 수단으로서, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관되는, 상기 결정하는 수단; 및 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 이진 스트링을 인코딩하는 수단을 포함한다.
- [0013] 본 개시물의 또 다른 예에서, 비디오 디코딩 프로세스에서 변환 계수들을 디코딩하도록 구성되는 장치는 비디오 블록과 연관되는 변환 계수들의 블록 내 최종 유의 계수의 위치를 나타내는 인코딩된 이진 스트링을 획득하는 수단으로서, 상기 인코딩된 이진 스트링은 CABAC 를 이용하여 인코딩되는, 상기 획득하는 수단; 비디오 블록 사이즈에 기초하여, 인코딩된 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하는 수단으로서, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관되는, 상기 결정하는 수단; 및 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 그 인코딩된 이진 스트링을 디코딩하는 수단을 포함한다.
- [0014] 본 개시물의 또 다른 예에서, 디바이스는 비디오 블록과 연관되는 변환 계수들의 블록 내 최종 유의 계수의 위치를 나타내는 이진 스트링을 획득하고; 비디오 블록 사이즈에 기초하여, 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하고; 및 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 이진 스트링을 인코딩하도록 구성되는 비디오 인코더를 포함하며, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관된다.
- [0015] 본 개시물의 또 다른 예에서, 디바이스는 비디오 블록과 연관되는 변환 계수들의 블록 내 최종 유의 계수의 위치를 나타내는 인코딩된 이진 스트링을 획득하고; 비디오 블록 사이즈에 기초하여, 인코딩된 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하고; 그리고, 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 그 인코딩된 이진 스트링을 디코딩하도록 구성되는 비디오 디코더를 포함하며, 상기 인코딩된 이진 스트링은 CABAC 를 이용하여 인코딩되며, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관된다.
- [0016] 본 개시물의 또 다른 예에서, 비일시성 컴퓨터-판독가능 저장 매체는, 실행 시, 비디오 인코딩 디바이스로 하여금, 비디오 블록과 연관되는 변환 계수들의 블록 내 최종 유의 계수의 위치를 나타내는 이진 스트링을 획득하고; 비디오 블록 사이즈에 기초하여, 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하고; 그리고, 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 이진 스트링을 인코딩하도록 하는 명령들을 저장하고 있으며, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관된다.
- [0017] 본 개시물의 또 다른 예에서, 비일시성 컴퓨터-판독가능 저장 매체는, 실행 시, 비디오 디코딩 디바이스로 하여금, 비디오 블록과 연관되는 변환 계수들의 블록 내 최종 유의 계수의 위치를 나타내는 인코딩된 이진 스트링을 획득하고; 비디오 블록 사이즈에 기초하여, 인코딩된 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하고; 그리고, 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 그 인코딩된 이진 스트링을 디코딩하도록 하는 명령들을 저장하고 있으며, 상기 인코딩된 이진 스트링은 CABAC 를 이용하여 인코딩되며, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관된다.
- [0018] 하나 이상의 예들의 세부 사항들이 첨부도면 및 아래의 상세한 설명에서 개시된다. 다른 특성들, 목적들, 및 이점들은 설명 및 도면들로부터, 그리고 청구항들로부터 명백히 알 수 있을 것이다.

도면의 간단한 설명

[0019]

도 1 은 예시적인 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.

도 2a 내지 도 2d 는 예시적인 계수 값 스캐닝 순서들을 예시한다.

도 3 은 계수 값들의 블록에 대한, 유의도 맵의 일 예를 예시한다.

도 4 는 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 비디오 인코더를 예시하는 블록도이다.

도 5 는 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 엔트로피 인코더를 예시하는 블록도이다.

도 6 은 본 개시물의 기법들에 따른, 최종 유의 계수의 위치를 나타내는 이진 스트링 값을 인코딩하는 일 예를 예시하는 플로우차트이다.

도 7 은 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 비디오 디코더를 예시하는 블록도이다.

도 8 은 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 엔트로피 디코더를 예시하는 블록도이다.

도 9 는 본 개시물의 기법들에 따른, 최종 유의 계수의 위치를 나타내는 이진 스트링 값을 디코딩하는 일 예를 예시하는 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

[0020]

일반적으로, 본 개시물은 비디오 데이터를 코딩하는 기법들을 설명한다. 특히, 본 개시물은 비디오 인코딩 및/또는 디코딩 프로세스에서 변환 계수들을 코딩하는 기법들을 설명한다. 블록-기반의 비디오 코딩에서, 변환 계수들의 블록은 2차원 (2D) 어레이로 배열될 수도 있다. 스캐닝 프로세스는 변환 계수들의 2차원 (2D) 어레이를 변환 계수들의 순서화된, 1차원 (1D) 어레이, 즉, 벡터로 재배열하도록 수행될 수도 있다. 하나 이상의 구문 엘리먼트들이 스캐닝 순서에 기초하여 변환 계수들의 블록 내 최종 유의 계수 (즉, 비-제로 계수) 의 위치를 나타내기 위해 사용될 수도 있다. 최종 유의 계수의 위치는 변환 계수들의 인코딩을 최적화하기 위해 비디오 인코더에 의해 사용될 수도 있다. 이와 유사하게, 비디오 디코더는 최종 유의 계수의 위치를 이용하여 변환 계수들의 디코딩을 최적화할 수도 있다. 따라서, 최종 유의 계수의 위치를 나타내는 하나 이상의 구문 엘리먼트들을 효율적으로 코딩하는 것이 바람직할 수도 있다.

[0021]

본 개시물은 최종 유의 계수의 위치를 나타내는 하나 이상의 구문 엘리먼트들을 코딩하는 기법들을 설명한다.

일부 예들에서, 최종 유의 계수의 위치를 나타내는 구문 엘리먼트들의 모두 또는 일부는 다음 기법들 중 임의의 기법에 따라서 엔트로피 코딩될 수도 있다: 컨텍스트 적응 가변 길이 코딩 (CAVLC), 컨텍스트 적응 이진 산술 코딩 (CABAC), 확률 간격 파티셔닝 엔트로피 코딩 (PIPE), 또는 기타 등등. "빈들" 또는 "빈 인덱스들" 로서 또한 지칭될 수도 있는 이진 인덱스들 및 컨텍스트 할당들을 이용하는 엔트로피 코딩 기법들에서, 공통 컨텍스트 할당이 상이한 변환 블록 (TU) 사이즈들 및/또는 상이한 칼라 성분들에 대한 빈들에 대해 이용될 수도 있다. 이러한 방법으로, 컨텍스트들의 총수가 감소될 수 있다. 컨텍스트들의 총수를 감소시킴으로써, 비디오 인코더 및/또는 비디오 디코더는 더 적은 컨텍스트들이 저장되는 것을 필요로 하기 때문에, 최종 유의 계수의 위치를 나타내는 구문 엘리먼트들 좀더 효율적으로 코딩할 수도 있다.

[0022]

도 1 은 본 개시물의 예들에 따른, 변환 계수들을 코딩하는 기법들을 이용하도록 구성될 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (10) 을 예시하는 블록도이다. 도 1 에 나타낸 바와 같이, 시스템 (10) 은 인코딩된 비디오를 목적지 디바이스 (14) 로 통신 채널 (16) 을 통해서 송신하는 소스 디바이스 (12) 를 포함한다. 인코딩된 비디오 데이터는 또한 저장 매체 (34) 또는 파일 서버 (36) 상에 저장될 수도 있으며, 원하는 바에 따라 목적지 디바이스 (14) 에 의해 액세스될 수도 있다. 저장 매체 또는 파일 서버에 저장될 때, 비디오 인코더 (20) 는 코딩된 비디오 데이터를 저장 매체에 저장하기 위해, 코딩된 비디오 데이터를 네트워크 인터페이스, 콤팩트 디스크 (CD), 블루-레이 또는 디지털 비디오 디스크 (DVD) 버너 또는 스탬핑 기능 디바이스, 또는 다른 디바이스들과 같은, 또 다른 디바이스에 제공할 수도 있다. 이와 유사하게, 네트워크 인터페이스, CD 또는 DVD 리더 등과 같은, 비디오 디코더 (30) 와는 별개인 디바이스가 저장 매체로부터 코딩된 비디오 데이터를 추출하고, 그 추출된 데이터를 비디오 디코더 (30) 에 제공할 수도 있다.

[0023]

소스 디바이스 (12) 및 목적지 디바이스 (14) 는 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 태블릿 컴퓨터들, 셋-탑 박스들, 전화기 핸드셋들, 예컨대 소위 스마트폰들, 텔레비전, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들 등을 포함한, 매우 다양한 디바이스들 중 임의의 디바이스들

포함할 수도 있다. 많은 경우에, 이런 디바이스들은 무선 통신용으로 탑재될 수도 있다. 그러므로, 통신 채널 (16) 은 인코딩된 비디오 데이터의 송신에 적합한, 무선 채널, 유선 채널, 또는 무선 채널과 유선 채널의 조합을 포함할 수도 있다. 이와 유사하게, 파일 서버 (36) 는 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해서, 목적지 디바이스 (14) 에 의해 액세스될 수도 있다. 이것은 파일 서버 상에 저장된 인코딩된 비디오 데이터에 액세스하는데 적합한, 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀 등), 또는 양자의 조합을 포함할 수도 있다.

[0024] 본 개시물의 예들에 따른, 변환 계수들을 코딩하는 기법들은 다양한 멀티미디어 애플리케이션들 중 임의의 애플리케이션, 예컨대 OTA (over-the-air) 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 예컨대, 인터넷을 통한 스트리밍 비디오 송신들, 데이터 저장 매체 상에의 저장을 위한 디지털 비디오의 인코딩, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션들의 지원 하에, 비디오 코딩에 적용될 수도 있다. 일부 예들에서, 시스템 (10) 은 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 및/또는 비디오 전화 통신과 같은, 지원 애플리케이션들로의 1-방향 또는 2-방향 비디오 송신을 지원하도록 구성될 수도 있다.

[0025] 도 1 의 예에서, 소스 디바이스 (12) 는 비디오 소스 (18), 비디오 인코더 (20), 변조기/복조기 (22) 및 송신기 (24) 를 포함한다. 소스 디바이스 (12) 에서, 비디오 소스 (18) 는 비디오 캡처 디바이스, 예컨대 비디오 카메라, 이전에 캡처된 비디오를 포함하는 비디오 아카이브, 비디오 콘텐츠 제공자로부터 비디오를 수신하는 비디오 공급 인터페이스, 및/또는 컴퓨터 그래픽스 데이터를 소스 비디오로서 발생하는 컴퓨터 그래픽스 시스템, 또는 이런 소스들의 조합과 같은, 소스를 포함할 수도 있다. 일 예로서, 비디오 소스 (18) 가 비디오 카메라이면, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 소위 카메라 폰들 또는 비디오 폰들일 수도 있다. 그러나, 본 개시물에서 설명하는 기법들은 일반적으로는 비디오 코딩에 적용가능할 수도 있으며, 무선 및/또는 유선 애플리케이션들, 또는 인코딩된 비디오 데이터가 로컬 디스크 상에 저장되는 애플리케이션에 적용될 수도 있다.

[0026] 캡처되거나, 사전-캡처되거나, 또는 컴퓨터-발생된 비디오는 비디오 인코더 (20) 에 의해 인코딩될 수도 있다. 인코딩된 비디오 정보는 무선 통신 프로토콜과 같은, 통신 표준에 따라서 모뎀 (22) 에 의해 변조되어, 송신기 (24) 를 통하여 목적지 디바이스 (14) 로 송신될 수도 있다. 모뎀 (22) 은 여러 믹서들, 필터들, 증폭기들 또는 신호 변조용으로 설계된 다른 구성요소들을 포함할 수도 있다. 송신기 (24) 는 증폭기들, 필터들, 및 하나 이상의 안테나들을 포함한, 데이터를 송신하도록 설계된 회로들을 포함할 수도 있다.

[0027] 비디오 인코더 (20) 에 의해 인코딩되는 캡처된, 사전-캡처된, 또는 컴퓨터-발생된 비디오는 또한 추후 소비를 위해 저장 매체 (34) 또는 파일 서버 (36) 상에 저장될 수도 있다. 저장 매체 (34) 는 블루-레이 디스크들, DVD들, CD-ROM들, 플래시 메모리, 또는 인코딩된 비디오를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체들을 포함할 수도 있다. 저장 매체 (34) 상에 저장된 인코딩된 비디오는 그후 디코딩 및 플레이백을 위해 목적지 디바이스 (14) 에 의해 액세스될 수도 있다.

[0028] 파일 서버 (36) 는 인코딩된 비디오를 저장하고 그 인코딩된 비디오를 목적지 디바이스 (14) 로 송신할 수 있는 임의 종류의 서버일 수도 있다. 예시적인 파일 서버들은 웹 서버 (예컨대, 웹사이트용), FTP 서버, 네트워크 부착된 스토리지 (NAS) 디바이스들, 로컬 디스크 드라이브, 또는 인코딩된 비디오 데이터를 저장하고 이를 목적지 디바이스로 송신할 수 있는 임의의 다른 유형의 디바이스들을 포함한다. 파일 서버 (36) 로부터의 인코딩된 비디오 데이터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이 양자의 조합일 수도 있다. 파일 서버 (36) 는 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해서, 목적지 디바이스 (14) 에 의해 액세스될 수도 있다. 이것은 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀, 이더넷, USB, 등), 또는 파일 서버 상에 저장된 인코딩된 비디오 데이터에 액세스하는데 적합한 이들 양자의 조합을 포함할 수도 있다.

[0029] 목적지 디바이스 (14) 는, 도 1 의 예에서, 수신기 (26), 모뎀 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32) 를 포함한다. 목적지 디바이스 (14) 의 수신기 (26) 는 채널 (16) 을 통해서 정보를 수신하고, 모뎀 (28) 은 비디오 디코더 (30) 를 위한 복조된 비트스트림을 발생한다. 채널 (16) 을 통해서 통신된 정보는 비디오 데이터를 디코딩할 때에 비디오 디코더 (30) 에 의한 사용을 위해 비디오 인코더 (20) 에 의해 발생된 다양한 구문 정보를 포함할 수도 있다. 이런 구문은 또한 저장 매체 (34) 또는 파일 서버 (36) 상에 저장된 인코딩된 비디오 데이터에 포함될 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 비디오 데이터를 인코딩 또는 디코딩할 수 있는 각각의 인코더-디코더 (코덱) 의 부분을 형성할 수도 있다.

- [0030] 디스플레이 디바이스 (32) 는 목적지 디바이스 (14) 와 통합되거나 이의 외부에 있을 수도 있다. 일부 예들에서, 목적지 디바이스 (14) 는 통합된 디스플레이 디바이스를 포함하며, 또한 외부 디스플레이 디바이스와 인터페이스하도록 구성될 수도 있다. 다른 예들에서, 목적지 디바이스 (14) 는 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32) 는 그 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 또 다른 유형의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 디바이스를 포함할 수도 있다.
- [0031] 도 1 의 예에서, 통신 채널 (16) 은 임의의 무선 또는 유선 통신 매체, 예컨대 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적인 송신 라인들, 또는 무선 매체와 유선 매체들의 임의의 조합을 포함할 수도 있다. 통신 채널 (16) 은 근거리 네트워크, 광역 네트워크, 또는 글로벌 네트워크, 예컨대 인터넷과 같은 패킷-기반 네트워크의 일부를 형성할 수도 있다. 통신 채널 (16) 은 일반적으로 비디오 데이터를 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 송신하는, 유선 또는 무선 매체들의 임의의 적합한 조합을 포함한, 임의의 적합한 통신 매체, 또는 상이한 통신 매체들의 컬렉션을 나타낸다. 통신 채널 (16) 은 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 통신을 촉진하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.
- [0032] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 ITU-T 비디오 코딩 전문가 그룹 (VCEG) 과 ISO/IEC 동화상 전문가 그룹 (MPEG) 의 JCT-VC (Joint Collaboration Team on Video Coding) 에 의해 현재 개발중인 HEVC (High Efficiency Video Coding) 표준과 같은, 비디오 압축 표준에 따라서 동작할 수도 있으며, HEVC 테스트 모델 (HM) 에 따를 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 는 ITU-T SG16 WP3 와 ISO/IEC JTC1/SC29/WG11 의 JCT-VC (Joint Collaborative Team on Video Coding), 7차 회의; 2012년 11월, 스위스, 제네바, 문서 JCTVC-G1103, Bross 등, "WD5: Working Draft 5 of High efficiency video coding (HEVC)" 에 설명되어 있는, "HEVC Working Draft 5" 또는 "WD5" 로서 지칭되는, HEVC 표준의 최신 안에 따라서 동작할 수도 있다. 또, HEVC 의 또 다른 최근의 작업 초안, 작업 초안 7 은 ITU-T SG16 WP3 와 ISO/IEC JTC1/SC29/WG11 의 JCT-VC (Joint Collaborative Team on Video Coding), 9차 회의; 2012년 4월 27일 내지 2012년 5월 7일, 스위스, 제네바, 문서 HCTVC-I1003, Bross 등, "High Efficiency Video Coding (HEVC) Text Specification Draft 7" 에 설명되어 있다. 이의 대안으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 MPEG 4, 파트 10, AVC (Advanced Video Coding) 로서 대안적으로 지칭되는 ITU-T H.264 표준, 또는 이런 표준들의 확장판들과 같은 다른 사유 (proprietary) 또는 산업 표준들에 따라서 동작할 수도 있다. 본 개시물의 기법들은, 그러나, 임의의 특정의 코딩 표준에 한정되지 않는다. 다른 예들은 MPEG-2 및 ITU-T H.263 를 포함한다.
- [0033] 도 1 에 나타내지는 않지만, 일부 양태들에서, 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 적합한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능한 경우, 일부 예들에서, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜, 또는 다른 프로토콜들, 예컨대, 사용자 데이터그램 프로토콜 (UDP) 을 따를 수도 있다.
- [0034] 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSP들), 주문형 집적회로들 (ASIC들), 필드 프로그래밍가능 게이트 어레이들 (FPGA들), 이산 로직, 소프트웨어, 하드웨어, 펌웨어 또는 임의의 이들의 조합들과 같은, 다양한 적합한 인코더 회로 중 임의의 것으로서 구현될 수도 있다. 이 기법들이 소프트웨어로 부분적으로 구현되는 경우, 디바이스는 본 개시물의 기법들을 수행하기 위해 소프트웨어용 명령들을 적합한 비일시성 컴퓨터-판독가능 매체에 저장하고, 그 명령들을 하드웨어에서 하나 이상의 프로세서들을 이용하여 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이들 중 어느 쪽이든 각각 디바이스에서 결합된 인코더/디코더 (코덱) 의 부분으로서 통합될 수도 있다.
- [0035] 비디오 인코더 (20) 는 비디오 인코딩 프로세스에서 변환 계수들을 코딩하는 본 개시물의 기법들 중 임의의 기법 또는 모두를 구현할 수도 있다. 이와 유사하게, 비디오 디코더 (30) 는 비디오 코딩 프로세스에서 변환 계수들을 코딩하는 이들 기법들 중 임의의 기법 또는 모두를 구현할 수도 있다. 비디오 코더는, 본 개시물에서 설명하는 바와 같이, 비디오 인코더 또는 비디오 디코더를 지칭할 수도 있다. 이와 유사하게, 비디오 코딩 유닛은 비디오 인코더 또는 비디오 디코더를 지칭할 수도 있다. 이와 유사하게, 비디오 코딩은 비디오 인코딩 또는 비디오 디코딩을 지칭할 수도 있다.
- [0036] 도 1 의 비디오 인코더 (20) 및 비디오 디코더 (30) 는 비디오 블록 내 최종 유의 계수의 위치를 나타내는 이진

스트링을 획득하고, 비디오 블록 사이즈에 기초하여, 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하고; 그리고, 그 결정된 컨텍스트에 적어도 부분적으로 기초하여, 컨텍스트 적응 이진 산술 코딩 (CABAC) 을 이용하여 이진 스트링을 코딩하도록 구성되는 비디오 코더들의 예들을 나타내며, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관된다.

[0037] 현재 개발중인 HEVC 표준에 따른 비디오 코딩에 있어, 비디오 프레임은 코딩 유닛들로 파티셔닝될 수도 있다.

코딩 유닛 (CU) 은 일반적으로 여러 코딩 틀들이 비디오 압축을 위해 제공되는 기본적인 유닛으로서 역할을 하는 이미지 영역을 지칭한다. CU 는 대개 Y 로 표기되는 휘도 성분, 및 U 및 V 로 표기되는 2개의 크로마 성분들을 갖는다. 비디오 샘플링 포맷에 따라서, U 및 V 성분들의 사이즈는, 샘플들의 개수의 관점에서, Y 성분의 사이즈와 동일하거나 또는 상이할 수도 있다. CU 는 대개 정사각형이며, 예컨대, ITU-T H.264 와 같은 다른 비디오 코딩 표준들 하에서, 소위 매크로블록과 유사한 것으로 간주될 수도 있다. 개발중인 HEVC 표준의 현재 제안된 양태들의 일부에 따른 코딩이 본 출원에서 예시의 목적을 위해 설명될 것이다. 그러나, 본 개시물에서 설명하는 기법들은 H.264 또는 다른 표준 또는 사유 비디오 코딩 프로세스들에 따라 정의된 것들과 같은 다른 비디오 코딩 프로세스들에 유용할 수도 있다. HEVC 표준화 노력들은 HEVC 테스트 모델 (HM) 로서 지칭되는 비디오 코딩 디바이스의 모델에 기초할 수도 있다. HM 은 예컨대, ITU-T H.264/AVC 에 따른 디바이스들을 넘어서는 비디오 코딩 디바이스들의 여러 능력들을 가정한다. 예를 들어, H.264 는 9개의 인트라-예측 인코딩 모드들을 제공하는 반면, HM 은 34개 만큼이나 많은 인트라-예측 인코딩 모드들을 제공한다.

[0038] 비디오 시퀀스는 일반적으로 비디오 프레임들 또는 화상들의 시리즈를 포함한다. 화상들의 그룹 (GOP) 은 일반적으로 비디오 화상들의 하나 이상의 시리즈를 포함한다. GOP 는 GOP 의 헤더, 화상들의 하나 이상의 헤더, 또는 다른 곳에, GOP 에 포함된 다수의 화상들을 기술하는 구문 데이터를 포함할 수도 있다. 화상의 각각의 슬라이스는 각각의 슬라이스에 대한 인코딩 모드를 기술하는 슬라이스 구문 데이터를 포함할 수도 있다.

비디오 인코더 (20) 는 일반적으로 비디오 데이터를 인코딩하기 위해 개개의 비디오 슬라이스들 내 비디오 블록들에 대해 동작한다. 비디오 블록은 CU 내 코딩 노드에 대응하는 하나 이상의 TU들 또는 PU들을 포함할 수도 있다. 비디오 블록들은 고정 또는 가변 사이즈들을 가질 수도 있으며, 규정된 코딩 표준에 따라서 사이즈가 상이할 수도 있다.

[0039] HM 에 따르면, CU 는 하나 이상의 예측 유닛들 (PU들) 및/또는 하나 이상의 변환 유닛들 (TU들) 을 포함할 수도 있다. 비트스트림 내 구문 데이터는 픽셀들의 개수의 관점에서 최대 CU 인 최대 코딩 유닛 (LCU) 을 정의한다. 일반적으로, CU 가 사이즈 구별을 갖지 않는다는 점을 제외하고는, CU 는 H.264 의 매크로블록과 유사한 목적을 갖는다. 따라서, CU 는 서브-CU들로 분할될 수도 있다. 일반적으로, 본 개시물에서 CU 에 대한 언급들은 화상의 최대 코딩 유닛 또는 LCU 의 서브-CU 를 지칭할 수도 있다. LCU 는 서브-CU들로 분할될 수도 있으며, 각각의 서브-CU 는 서브-CU들로 추가로 분할될 수도 있다. 비트스트림에 대한 구문 데이터는 LCU 가 분할되는 최대 횟수를 정의할 수도 있으며, 이는 CU 깊이로 지칭된다. 따라서, 비트스트림은 또한 최소 코딩 유닛 (SCU) 을 정의할 수도 있다. 본 개시물은 또한 CU, PU, 또는 TU 중 임의의 것을 지칭하기 위해 용어 "블록" 또는 "부분" 을 사용한다. 일반적으로, "부분" 은 임의의 비디오 프레임의 서브-세트를 지칭할 수도 있다.

[0040] LCU 는 쿼드트리 데이터 구조와 연관될 수도 있다. 일반적으로, 쿼드트리 데이터 구조는 CU 당 하나의 노드를 포함하며, 여기서, 루트 노드는 LCU 에 대응한다. CU 가 4개의 서브-CU들로 분할되면, CU 에 대응하는 노드는 4개의 잎 노드들을 포함하며, 각각의 잎 노드는 서브-CU들 중 하나에 대응한다. 쿼드트리 데이터 구조의 각각의 노드는 대응하는 CU 에 대해 구문 데이터를 제공할 수도 있다. 예를 들어, 쿼드트리에서 노드는 그 노드에 대응하는 CU 가 서브-CU들로 분할될지 여부를 나타내는 분할 플래그를 포함할 수도 있다. CU 에 대한 구문 엘리먼트들은 회귀적으로 정의될 수도 있으며, CU 가 서브-CU들로 분할되는지 여부에 의존할 수도 있다. CU 가 추가로 분할되지 않으면, 잎-CU 로서 지칭된다. 본 개시물에서, 원래 잎-CU 의 명시적인 분할이 없지만, 잎-CU 의 4 개의 서브-CU들은 또한 잎-CU들로서 지칭될 것이다. 예를 들어, 16x16 사이즈에서 CU 가 추가로 분할되지 않으면, 또한 16x16 CU 가 전혀 분할되지 않았지만 4개의 8x8 서브-CU들은 잎-CU들로서 지칭될 것이다.

[0041] 더욱이, 잎-CU들의 TU들은 또한 각각의 쿼드트리 데이터 구조들과 연관될 수도 있다. 즉, 잎-CU 는 잎-CU 가 어떻게 TU들로 파티셔닝되는지를 나타내는 쿼드트리를 포함할 수도 있다. 본 개시물은 어떻게 LCU 가 CU 쿼드트리로서 파티셔닝되는지를 나타내는 쿼드트리, 및 어떻게 잎-CU 가 TU들에 TU 쿼드트리로서 파티셔닝되는지를 나타내는 쿼드트리를 지칭한다. TU 쿼드트리의 루트 노드는 일반적으로 잎-CU 에 대응하는 반면, CU 쿼드트리의 루트 노드는 일반적으로 LCU 에 대응한다. 분할되지 않은 TU 쿼드트리의 TU들은 잎-TU들로서 지

칭된다.

[0042] 앞-CU 는 하나 이상의 예측 유닛들 (PU들) 을 포함할 수도 있다. 일반적으로, PU 는 대응하는 CU 의 모두 또는 부분을 나타내며, PU 에 대한 참조 샘플을 추출하기 위한 데이터를 포함할 수도 있다. 예를 들어, PU 가 인터-모드 인코딩될 때, PU 는 PU 에 대한 모션 벡터를 정의하는 데이터를 포함할 수도 있다. 모션 벡터를 정의하는 데이터는 예를 들어, 모션 벡터의 수평 성분, 모션 벡터의 수직 성분, 모션 벡터에 대한 해상도 (예컨대, 1/4 픽셀 정밀도 또는 1/8 픽셀 정밀도), 모션 벡터가 가리키는 참조 프레임, 및/또는 모션 벡터에 대한 참조 리스트 (예컨대, 리스트 0 또는 리스트 1) 를 기술할 수도 있다. PU(들) 을 정의하는 앞-CU 에 대한 데이터는 또한 예를 들어, 하나 이상의 PU들로의 CU 의 파티셔닝을 기술할 수도 있다. 파티셔닝 모드들은 CU 가 인코딩되지 않거나, 인트라-예측 모드 인코딩되거나, 또는 인터-예측 모드 인코딩되는지 여부에 따라 상이할 수도 있다. 인트라 코딩에 있어, PU 는 아래에서 설명되는 일 변환 유닛과 동일하게 취급될 수도 있다.

[0043] 일 예로서, HM 은 여러 PU 사이즈들에서 예측을 지원한다. 특정의 CU 의 사이즈가 $2N \times 2N$ 이라고 가정하면, HM 은 $2N \times 2N$ 또는 $N \times N$ 의 PU 사이즈들에서 인트라-예측을, 그리고 $2N \times 2N$, $2N \times N$, $N \times 2N$, 또는 $N \times N$ 의 대칭적인 PU 사이즈들에서 인터-예측을 지원한다. HM 은 또한 $2N \times N$ U, $2N \times ND$, $nL \times 2N$, 및 $nR \times 2N$ 의 PU 사이즈들에서의 인터-예측에 대해 비대칭적인 파티셔닝을 지원한다. 비대칭적인 파티셔닝에서, CU 의 하나의 방향은 파티셔닝되지 않지만, 다른 방향은 25% 및 75% 로 파티셔닝된다. 25% 파티션에 대응하는 CU 의 부분은 "상부 (Up)", "하부 (Down)", "좌측 (좌측)", 또는 "우측 (우측)" 의 표시가 뒤따르는 "n" 으로 표시된다. 따라서, 예를 들어, " $2N \times nU$ " 는 최상부에서 $2N \times 0.5N$ PU 으로 그리고 최저부에서 $2N \times 1.5N$ PU 으로 수평으로 파티셔닝된 $2N \times 2N$ CU 를 지칭한다.

[0044] 본 개시물에서, " $N \times N$ " 및 "N 곱하기 N" 은 수직 및 수평 치수들의 관점에서 비디오 블록의 픽셀 치수들, 예컨대, 16×16 픽셀들 또는 16 곱하기 16 픽셀들을 지칭하기 위해 상호교환가능하게 사용될 수도 있다. 일반적으로, 16×16 블록은 수직 방향으로 16 개의 픽셀들 ($y = 16$) 및 수평 방향으로 16 개의 픽셀들 ($x = 16$) 을 가질 것이다. 이와 유사하게, $N \times N$ 블록은 일반적으로 수직 방향으로 N 개의 픽셀들 및 수평 방향으로 N 개의 픽셀들을 가지며, 여기서 N 은 음이 아닌 정수 값을 나타낸다. 블록에서 픽셀들은 로우들 및 칼럼들로 배열될 수도 있다. 더욱이, 블록들은 수직 방향에서와 같이 수평 방향에서 동일한 픽셀들의 개수를 반드시 가질 필요는 없다. 예를 들어, 블록들은 $N \times M$ 픽셀들을 포함할 수도 있으며, 여기서 M 은 반드시 N 과 같을 필요는 없다.

[0045] 블록 (예컨대, 비디오 데이터의 예측 유닛) 을 코딩하기 위해, 그 블록에 대한 예측자가 먼저 유도된다. 예측 블록으로 또한 지칭되는 예측자는 인트라 (I) 예측 (즉, 공간 예측) 또는 인터 (P 또는 B) 예측 (즉, 시간 예측) 을 통해서 유도될 수 있다. 그러므로, 일부 예측 유닛들은 동일한 프레임 (또는, 슬라이스) 에서 이웃하는 참조 블록들에서의 참조 샘플들에 대해 공간 예측을 이용하여 인트라-코딩 (I) 될 수도 있으며, 다른 예측 유닛들은 다른 이전에-코딩된 프레임들 (또는, 슬라이스들) 에서의 참조 샘플들의 블록들에 대해 단방향 인터-코딩 (P) 되거나 또는 양방향 인터-코딩 (B) 될 수도 있다. 각 경우에, 코딩되는 블록에 대한 예측 블록을 형성하는데 참조 샘플들이 사용될 수도 있다.

[0046] 예측 블록의 식별 시, 원래 비디오 데이터 블록과 그의 예측 블록 사이의 차이가 결정된다. 이 차이는 예측 잔여 데이터로서 지칭될 수도 있으며, 코딩되는 블록에서의 픽셀 값들과, 코딩된 블록을 나타내기 위해 선택된 예측 블록에서의 픽셀 값들 사이의 픽셀 차이들을 나타낸다. 더 나은 압축을 달성하기 위해, 예측 잔여 데이터는 예컨대, 이산 코사인 변환 (DCT), 정수 변환, Karhunen-Loeve (K-L) 변환, 또는 또 다른 변환을 이용하여, 변환될 수도 있다.

[0047] TU 와 같은 변환 블록에서의 잔여 데이터는 공간, 픽셀 도메인에 상주하는 픽셀 차이 값들의 2차원 (2D) 어레이로 배열될 수도 있다. 변환은 잔여 픽셀 값들을 주파수 도메인과 같은 변환 도메인에서 변환 계수들의 2차원 어레이로 변환한다. 추가적인 압축을 위해, 변환 계수들은 엔트로피 코딩 이전에 양자화될 수도 있다. 엔트로피 코더는 그후 CAVLC, CABAC, PIPE 등과 같은 엔트로피 코딩을 양자화된 변환 계수들에 적용한다.

[0048] 양자화된 변환 계수들의 블록을 엔트로피 코딩하기 위해, 스캐닝 프로세스는 대개, 블록에서의 양자화된 변환 계수들의 2차원 (2D) 어레이가 특정의 스캐닝 순서에 따라서, 변환 계수들의 순서화된, 1차원 (1D) 어레이, 즉, 벡터로 재배열되도록, 수행된다. 엔트로피 코딩이 그후 변환 계수들의 벡터에 적용된다. 변환 유닛에서의 양자화된 변환 계수들의 스캐닝은 엔트로피 코더를 위한 변환 계수들의 2D 어레이를 직렬화한다. 유의도 맵이 유의한 (즉, 논-제로) 계수들의 위치들을 나타내기 위해 발생될 수도 있다. 스캐닝이 유의한 (즉, 논

제로) 계수들의 레벨들에 스캐닝하는데, 및/또는 유의 계수들의 부호들을 코딩하는데, 적용될 수도 있다.

- [0049] HEVC 에서, 스캐닝 순서에서 최종 비-제로 계수의 로케이션을 나타내기 위해 유의한 변환 계수들의 위치 정보 (예컨대, 유의도 맵) 가 TU 에 대해 먼저 코딩된다. 유의도 맵 및 레벨 정보 (계수들의 절대 값들 및 부호들) 는 각각의 계수에 대해 역 스캐닝 순서로 코딩된다.
- [0050] 변환 계수들을 발생하는 임의의 변환들 이후, 비디오 인코더 (20) 는 변환 계수들의 양자화를 수행할 수도 있다. 양자화는 일반적으로 변환 계수들이 계수들을 나타내는데 사용되는 데이터의 양을 가능한 한 감축하기 위해 양자화되는 프로세스를 지칭하며, 추가적인 압축을 제공한다. 양자화 프로세스는 그 계수들의 일부 또는 모두와 연관되는 비트 심도를 감소시킬 수도 있다. 예를 들어, n-비트 값은 양자화 동안 m-비트 값까지 절사될 수도 있으며, 여기서, n 은 m 보다 더 크다. 일부 예들에서, 비디오 인코더 (20) 는 엔트로피 인코딩될 수 있는 직렬화된 벡터를 발생하기 위해, 미리 정의된 스캐닝 순서를 이용하여, 양자화된 변환 계수들을 스캐닝할 수도 있다. 다른 예들에서, 비디오 인코더 (20) 는 적응적 스캐닝을 수행할 수도 있다.
- [0051] 도 2a 내지 도 2d 는 일부 상이한 예시적인 스캐닝 순서들을 예시한다. 다른 정의된 스캐닝 순서들, 또는 적응적 (변하는) 스캐닝 순서들이 또한 사용될 수도 있다. 도 2a 는 지그-재그 스캐닝 순서를 예시하며, 도 2b 는 수평 스캐닝 순서를 예시하며, 도 2c 는 수직 스캐닝 순서들을 예시하며, 그리고 도 2d 는 대각선 스캐닝 순서를 예시한다. 이들 스캐닝 순서들의 조합들이 또한 정의되어 사용될 수 있다. 일부 예들에서, 본 개시물의 기법들은 비디오 코딩 프로세스에서의 소위 유의도 맵의 코딩 동안 특히 적용가능할 수도 있다.
- [0052] 계수들의 블록과 연관되는 스캐닝 순서에 의존할 수도 있는 최종 유의 계수 (즉, 비-제로 계수) 의 위치를 나타내기 위해 하나 이상의 구문 엘리먼트들이 정의될 수도 있다. 예를 들어, 하나의 구문 엘리먼트는 계수 값들의 블록 내 최종 유의 계수의 칼럼 위치를 정의할 수도 있으며, 또 다른 구문 엘리먼트는 계수 값들의 블록 내 최종 유의 계수의 로우 위치를 정의할 수도 있다.
- [0053] 도 3 은 계수 값들의 블록에 대한, 유의도 맵의 일 예를 예시한다. 유의도 맵이 우측 상에 도시되며, 여기서, 1-비트 플래그들은 유의한, 즉, 논-제로인 좌측 상의 비디오 블록에서의 계수들을 식별한다. 일 예에서, (예컨대, 유의도 맵에 의해 정의되는) 유의 계수들의 세트 및 스캐닝 순서가 주어지면, 최종 유의 계수의 위치가 결정될 수도 있다. 차기 HEVC 표준에서, 변환 계수들은 청크로 그룹화될 수도 있다. 청크는 전체 TU 를 포함할 수도 있거나, 또는 일부 경우들에서, TU들은 더 작은 청크들로 서브-분할될 수도 있다. 유의도 맵 및 레벨 정보 (절대 값 및 부호) 가 청크에서의 각각의 계수에 대해 코딩된다. 일 예에서, 4×4 TU 및 8×8 TU 에 있어 청크는 역 스캐닝 순서 (예컨대, 대각선, 수평, 또는 수직) 로 16 개의 연속되는 계수들로 구성된다. 16×16 및 32×32 TU들에 있어, 4×4 서브-블록들 내 계수들은 청크로서 취급된다. 청크 내 계수들 레벨 정보를 나타내기 위해 구문 엘리먼트들이 코딩되어 시그널링된다. 일 예에서, 모든 심볼들은 역 스캐닝 순서로 인코딩된다. 본 개시물의 기법들은 계수들의 블록의 이 최종 유의 계수의 위치를 정의하는데 사용되는 구문 엘리먼트의 코딩을 향상시킬 수도 있다.
- [0054] 일 예로서, 본 개시물의 기법들은 계수들의 블록 (예컨대, TU 또는 TU 의 청크) 의 최종 유의 계수의 위치를 코딩하는데 사용될 수도 있다. 그 후, 최종 유의 계수의 위치를 코딩한 후, 변환 계수들과 연관되는 레벨 및 부호 정보가 코딩될 수도 있다. 레벨 및 부호 정보의 코딩은 다음 심볼들을 (예컨대, TU 또는 TU 의 청크에 대해) 역 스캐닝 순서로 코딩함으로써 5 과정 접근법 (five pass approach) 에 따라서 프로세싱할 수도 있다:
- [0055] significant_coeff_flag (약칭 sigMapFlag): 이 플래그는 청크에서의 각각의 계수의 유의도를 나타낼 수도 있다. 하나 더 큰 값을 가진 계수가 유의한 것으로 간주될 수도 있다.
- [0056] coeff_abs_level_greater1_flag (약칭 gr1Flag): 이 플래그는 계수의 절대 값이 비-제로 계수들 (즉, 1 과 같이 sigMapFlag 를 가진 계수들) 에 대해 1 보다 더 큰지 여부를 나타낼 수도 있다.
- [0057] coeff_abs_level_greater2_flag (약칭 gr2Flag): 이 플래그는 계수의 절대 값이 1 보다 더 큰 절대 값을 가진 계수들 (즉, 1 과 같이 gr1Flag 를 가진 계수들) 에 대해 2보다 더 큰지 여부를 나타낼 수도 있다.
- [0058] coeff_sign_flag (약칭 signFlag): 이 플래그는 비-제로 계수들에 대한 부호 정보를 나타낼 수도 있다. 예를 들어, 이 플래그에 있어, 제로는 양의 부호를 나타낼 수도 있지만, 반면 1 은 음의 부호를 나타낼 수도 있다.
- [0059] coeff_abs_level_remain (약칭 levelRem): 이 구문 엘리먼트는 변환 계수 레벨의 나머지 절대 값을 나타낼 수도 있다. 이 구문 엘리먼트에 있어, 계수 - x 의 절대 값은 x 보다 더 큰 진폭을 가진 각각의 계수에 대해

(abs(레벨)-x)로서 코딩될 수도 있다. x의 값은 gr1Flag 및 gr2Flag의 존재에 의존한다. gr2Flag가 코딩되었으면, levelRem 값이 (abs(레벨)-2)로서 계산된다. gr2Flag가 코딩되지 않았지만 gr1Flag가 코딩되었으면, levelRem 값이 (abs(레벨)-1)로서 계산된다.

[0060] 이러한 방법으로, TU 또는 TU의 부분(예컨대, 청크)에 대한 변환 계수들이 코딩될 수 있다. 어쨌든, 계수들의 블록의 최종 유의 계수의 위치를 정의하는데 사용되는 구문 엘리먼트의 코딩에 관련하는 본 개시물의 기법들은 또한 변환 계수들의 레벨 및 부호 정보를 궁극적으로 코딩하는 다른 유형들의 기법들과 함께 사용될 수도 있다. 유의도, 레벨 및 부호 정보를 코딩하는 5 과정 접근법은 본 개시물에서 제시하는 바와 같은, 블록의 최종 유의 계수의 위치의 코딩에 뒤이어서, 사용될 수도 있는 단지 하나의 예시적인 기법이다.

[0061] 양자화된 변환 계수들을 스캐닝하여 1차원 벡터를 형성한 후, 비디오 인코더 (20)는 변환 계수들의 1차원 벡터를 엔트로피 인코딩할 수도 있다. 비디오 인코더 (20)는 또한 비디오 데이터를 디코딩할 때에 비디오 디코더 (30)에 의해 사용하기 위한 인코딩된 비디오 데이터와 연관되는 구문 엘리먼트들을 엔트로피 인코딩할 수도 있다. 엔트로피 인코딩은 다음 기법들, 즉, CAVLC, CABAC, 구문-기반의 컨텍스트-적응 이진 산술 코딩 (SBAC), PIPE 코딩 또는 또 다른 엔트로피 인코딩 방법론 중 하나에 따라서 수행될 수도 있다. CAVLC을 수행하기 위해, 비디오 인코더 (20)는 송신되는 심볼에 대해 가변 길이 코드를 선택할 수도 있다. 가변 길이 코딩 (VLC)에서의 코드워드들은 상대적으로 더 짧은 코드들이 더 가능성 있는 심볼들에 대응하지만, 더 긴 코드들이 덜 가능성 있는 심볼들에 대응하도록, 구성될 수도 있다. 이러한 방법으로, VLC의 사용은 예를 들어, 송신되는 각각의 심볼에 대해 동일-길이 코드워드들을 사용하는 것을 넘어서는 비트 절감을 달성할 수도 있다.

[0062] 본 개시물의 엔트로피 코딩 기법들은 특히 CABAC에 적용가능한 것으로 설명되지만, 기법들은 또한 CAVLC, SBAC, PIPE, 또는 다른 기법들에도 적용가능할 수도 있다. PIPE는 산술 코딩의 원리들과 유사한 원리들을 이용한다는 점에 유의해야 한다.

[0063] 일반적으로, CABAC를 이용하여 데이터 심볼들을 코딩하는 것은 다음 단계들 중 하나 이상을 수반할 수도 있다:

[0064] (1) 이진화: 코딩되는 심볼이 비-이진 값이면, 이진 시퀀스, 즉 소위 "빈 스트링"에 매핑된다. 빈 스트링에서의 각각의 이진 인덱스(즉, "빈")는 "0" 또는 "1"의 값을 가질 수 있다.

[0065] (2) 컨텍스트 할당: 정규 모드(regular mode)에서, 각각의 빈은 컨텍스트에 할당된다. 빈은 또한 컨텍스트가 할당되는 않는 바이-패스 모드에 따라서 코딩될 수도 있다. 컨텍스트는 확률 모델이며, "컨텍스트 모델"로서 지칭된다. 본원에서 사용될 때, 용어 컨텍스트는 확률 모델 또는 확률 값을 지칭할 수도 있다. 컨텍스트는 어떻게 빈의 값의 확률이 주어진 빈에 대해 계산되는지를 결정한다. 컨텍스트는 이전에 인코딩된 심볼들의 값들 또는 빈 넘버(bin number)와 같은 정보에 기초하여, 빈의 값의 확률을 상관할 수도 있다. 또, 컨텍스트는 더 높은 레벨(예컨대, 슬라이스) 파라미터들에 기초하여 빈에 할당될 수도 있다.

[0066] (3) 빈 인코딩: 빈들은 산술 인코더로 인코딩된다. 빈을 인코딩하기 위해, 산술 인코더는 빈의 값의 확률, 즉, 빈의 값이 "0"과 동일할 확률, 및/또는 빈의 값이 "1"과 동일할 확률을 입력으로서 요구한다. (추정된) 확률은 "컨텍스트 상태"로서 지칭되는 정수 값에 의해 컨텍스트로 표현될 수도 있다.

[0067] (4) 상태 업데이트: 선택된 컨텍스트에 대한 확률(상태)은 빈의 실제 코딩된 값에 기초하여 업데이트될 수도 있다(예컨대, 빈 값이 "1"이었으면, "1"인 빈의 확률이 증가될 수도 있다). 확률들의 업데이트는 컨텍스트와 연관되는 전환 규칙들에 따라서 지배될 수도 있다.

[0068] 다음은 비디오 인코더 (20)에 의해 수행될 수도 있는 최종 유의 계수 구문 엘리먼트들의 예시적인 이진화 기법이다. 최종 유의 계수 구문 엘리먼트들은 2차원 블록 내에서 최종 유의 계수의 로우 및 칼럼 위치(즉, x 및 y 좌표)를 포함할 수도 있다. 8x8 블록에 있어, 칼럼 또는 로우 내 계수의 최종 위치, 즉, 0, 1, ..., 7에 대해 8개의 상이한 가능성들이 존재한다. 8개의 상이한 빈들이 이들 8개의 로우 또는 칼럼 위치들을 나타내기 위해 사용된다. 예를 들어, bin0 = 1은 로우 또는 칼럼 0에서의 계수가 최종 유의 계수임을 나타낼 수도 있다. 이 예에서, bin0 = 0이면, 로케이션 0에서의 계수는 최종 계수가 아니다. 1과 동일한 또 다른 빈이 최종 유의 계수의 위치를 나타낼 수도 있다. 예를 들어, bin1 = 1은 로케이션 1이 최종 유의 계수인 것을 나타낼 수도 있다. 또 다른 예로서, binX = 1은 로케이션 X가 최종 유의 계수인 것을 나타낼 수도 있다. 위에서 설명한 바와 같이, 각각의 빈은 2개의 상이한 방법들, 즉 (1) 컨텍스트로 빈을 인코딩하는 방법, 및 (2) 바이패스 모드를 이용하여(컨텍스트 없이) 빈을 인코딩하는 방법에 의해 인코딩될 수도 있다.

[0069] 표 1 은 일부 빈들이 컨텍스트로 인코딩되고 나머지들이 바이패스 모드를 이용하여 인코딩되는, 최종 유의 계수의 위치의 예시적인 이진화를 나타낸다. 표 1 에서의 예는 32x32 TU 에 대한 최종 유의 계수의 예시적인 이진화를 제공한다. 표 1 의 제 2 칼럼은 $2\log_2(T)-1$ 의 최대 트렁케이트된 일진수 접두사 (prefix) 길이로 정의되는 사이즈 T 의 TU 내 최종 유의 계수의 위치의 가능한 값들에 대해 대응하는 트렁케이트된 일진수 접두사 값들을 제공한다. 표 1 의 제 3 칼럼은 각각의 트렁케이트된 일진수 접두사에 대해 대응하는 고정된 길이 접미사 (suffix) 를 제공한다. 간결함을 위해, 표 1 은 하나 또는 제로 비트 값을 나타내는 X 값들을 포함한다. X 값들은 트렁케이트된 일진수 접두사를 공유하는 각각의 값을 고정된 길이 코드에 따라서 고유하게 맵핑한다는 점에 유의해야 한다. 표 1 에서의 최종 위치 성분의 크기는 x-좌표 값 및/또는 y-좌표 값에 대응할 수도 있다. 4x4, 8x8, 및 16x16 TU 에 대한 최종 유의 계수의 이진화는 표 1 과 관련하여 설명한 32x32 TU 의 이진화와 유사한 방법으로 정의될 수도 있다는 점에 유의해야 한다.

표 1

최종 위치 성분의 크기	트렁케이트된 일진수 (컨텍스트 모델)	고정된 일진수 (바이패스)	f_value
0	1	-	0
1	01	-	0
2	001	-	0
3	0001	-	0
4-5	00001	X	0-1
6-7	000001	X	0-1
8-11	0000001	XX	0-3
12-15	00000001	XX	0-3
16-23	000000001	XXX	0-7
24-31	000000000	XXX	0-7

[0070]

[표 1] 사이즈 32x32 의 TU 에 대한 이진화 (X 는 1 또는 0 을 의미한다).

[0071]

[0072] 위에서 설명한 바와 같이, CABAC 를 이용하여 데이터 심볼들을 코딩하는 것은 또한 컨텍스트 할당을 수반할 수도 있다. 일 예에서, 컨텍스트 모델링은 빈 스트링의 트렁케이트된 일진수 스트링들 부분의 산술적 인코딩을 위해 사용될 수도 있지만, 반면 컨텍스트 모델링은 빈 스트링의 고정된 이진 스트링들의 산술적 인코딩을 위해 사용되지 않는다 (즉, 고정된 이진 스트링이 바이패스된다). 트렁케이트된 일진수 스트링들이 컨텍스트 모델링을 이용하여 인코딩되는 경우, 컨텍스트는 이진 스트링의 빈 인덱스의 각각에 할당될 수도 있다.

[0073]

컨텍스트들이 이진 스트링의 각각의 빈 인덱스에 할당될 수도 있는 여러 방법들이 존재한다. 최종 유의 계수의 위치를 나타내는 빈 스트링에 대한 컨텍스트 할당들의 수는 빈 인덱스들의 수 또는 가능한 TU 사이즈들 및 칼라 성분들에 대한 트렁케이트된 일진수 스트링의 길이와 동일할 수도 있다. 예를 들어, 루마 성분의 가능한 사이즈들이 4x4, 8x8, 16x16 및 32x32 이면, 빈들 중 어느 것도 바이패스되지 않을 때, 하나의 치수에 대한 컨텍스트 할당들의 수는 60 (즉, $4+8+16+32$) 과 동일할 수도 있다. 이와 유사하게, 빈들 중 어느 것도 바이패스되지 않을 때, 4x4, 8x8, 및 16x16 의 가능한 사이즈들을 가진 각각의 크로마 성분에서, 컨텍스트 할당들의 수는 28 (즉, $4+8+16$) 과 동일할 수도 있다. 따라서, 최종 유의 계수의 위치가 x 및 y 좌표 양자를 이용하여 규정될 때, 컨텍스트 할당들의 최대 수는 각각의 치수에 대해 116 (즉, $60+28+28$) 과 동일할 수도 있다. 아래의 예시적인 컨텍스트 할당들은 일부 빈들이 표 1 과 관련하여 설명한 이진화 방식에 따라서 바이패스되는 것으로 가정한다. 그러나, 컨텍스트 할당 본원에서 설명하는 기법들은 여러 이진화 방식들에 적용가능할 수도 있다는 점에 유의해야 한다. 또, 일부 빈들이 바이패스되는 것으로 가정되더라도, 컨텍스트들이 최종 유의 계수의 위치를 나타내는 빈 스트링의 빈들에 할당될 수 있는 매우 많은 방법들이 여전히 존재한다.

[0074]

일부 경우들에서, 요구된 개수의 컨텍스트 할당들의 수에 대해 컨텍스트들의 총 수를 감소시키는 것이 바람직할 수도 있다. 이러한 방법으로, 인코더 또는 디코더는 많은 컨텍스트들로서 저장하여 유지할 필요가 없을 수도 있다. 그러나, 컨텍스트들의 수가 감소될 때, 예측 정확도는 예를 들어, 상이한 확률들을 가진 2개의 빈들에 대해 컨텍스트들이 공유되면, 또한 감소될 수도 있다. 또, 특정의 컨텍스트가 더 빈번하게 업데이트될

수도 있는데, 이것은 추정된 bin의 값의 확률에 영향을 미칠 수도 있다. 즉, 할당되는 컨텍스트를 이용하여 bin을 코딩하는 것은 컨텍스트를 업데이트하는 것을 수반할 수도 있다. 따라서, 컨텍스트에 할당되는 후속 bin은 업데이트된 컨텍스트를 이용하여 코딩될 수도 있다. 또, 일부 예들에서, 컨텍스트 모델들은 슬라이스 내 bin들의 값들이 후속 슬라이스 내 bin들의 코딩에 영향을 미치지 않도록, 슬라이스 레벨 상에서 초기화되지만, bin들이 동일한 컨텍스트에 할당된다는 점에 유의해야 한다. 본 개시물은 추정된 확률들에 대한 정확도를 유지하면서 다수의 컨텍스트들이 감소될 수 있도록, 컨텍스트 할당들을 최적화하는 기법들을 설명한다. 일 예에서, 본원에서 설명하는 컨텍스트 할당 기법들은 개개의 bin 인덱스들이 동일한 컨텍스트를 할당하는 기법들을 포함한다.

[0075] 아래 표들 2-13은 TU 내 최종 유의 계수의 위치를 나타내는 bin 스트링의 bin 인덱스들에 대한 컨텍스트 할당들을 예시한다. 표들 2-13에서 일부 bin들 (예컨대, 8x8 블록의 bin들 5-7)에 있어, 어떤 할당되는 컨텍스트들도 존재하지 않는다는 점에 유의해야 한다. 이것은 이들 bin들이 위에서 설명한 바와 같이, 바이패스 모드를 이용하여 코딩되는 것으로 가정되기 때문이다. 또한, 표 2-13에서의 값들이 컨텍스트의 인덱스를 나타낸다는 점에 유의해야 한다. 표들 2-13에서, 상이한 bin들이 동일한 컨텍스트 인덱스 값을 가질 때, 그들은 동일한 컨텍스트를 공유한다. 실제 컨텍스트에의 컨텍스트 인덱스의 맵핑은 비디오 코딩 표준에 따라서 정의될 수도 있다. 표들 2-13은 어떻게 컨텍스트가 bin들에 공통으로 (commonly) 할당되는지를 예시한다.

[0076] 표 2는 위 표 1과 관련하여 위에서 제공되는 예시적인 이진수화들에 있어, 상이한 TU 사이즈들의 각각의 bin에 대한 가능한 컨텍스트 인덱싱을 예시한다. 표 2에서의 예에서, 인접한 bin들이 동일한 컨텍스트들을 공유하도록 허용된다. 예를 들어, 8x8 TU의 bin들 2 및 3은 동일한 컨텍스트를 공유한다.

표 2

bin 인덱스	0	1	2	3	4	5	6	7	8
TU 4x4	0	1	2						
TU 8x8	3	4	5	5	6				
TU 16x16	7	8	9	9	10	10	11		
TU 32x32	12	13	14	14	15	15	16	16	17

[0077]

[표 2] 최종 위치 코딩을 위한 컨텍스트 할당

[0079] 표들 3-6은 각각 다음 규칙들에 따른 컨텍스트 할당들의 추가적인 예들을 예시한다:

[0080] 1. 처음 K개의 bin들은 컨텍스트들을 공유하지 않으며, 여기서, $K > 1$ 이다. K는 각각의 TU 사이즈에 대해 상이할 수 있다.

[0081] 2. 하나의 컨텍스트가 연속되는 bin들에 오직 할당될 수 있다. 예를 들어, bin3 - bin5는 컨텍스트 5를 이용할 수 있다. 그러나, 컨텍스트 5를 이용하는 bin3 및 bin5 및 컨텍스트 6을 이용하는 bin4는 허용되지 않는다.

[0082] 3. 상이한 TU 사이즈들의 마지막 N개의 bin ($N \geq 0$)은 동일한 컨텍스트를 공유할 수 있다.

[0083] 4. 동일한 컨텍스트를 공유하는 bin들의 수는 TU 사이즈들에 따라 증가한다.

[0084] 상기 규칙들 1-4는 표 1에 제공되는 이진화에 특히 유용할 수도 있다. 그러나, 규칙들 1-4는 다른 이진화 방식들에 동일하게 유용할 수도 있으며, 이에 따라서 실제 컨텍스트 할당들은 구현되는 이진화 방식에 조정될 수도 있다.

표 3

빈 인덱스	0	1	2	3	4	5	6	7	8
TU 4x4	0	1	2						
TU 8x8	3	4	5	6	7				
TU 16x16	8	9	10	11	11	12	12		
TU 32x32	13	14	14	15	16	16	16	16	17

[0085]

[0086] [표 3] 규칙들 1-4 에 따른 최종 위치 빈들의 예

표 4

빈 인덱스	0	1	2	3	4	5	6	7	8
TU 4x4	0	1	2						
TU 8x8	3	4	5	6	6				
TU 16x16	8	9	10	11	11	12	12		
TU 32x32	13	14	14	15	16	16	16	16	17

[0087]

[0088] [표 4] 규칙들 1-4 에 따른 최종 위치 빈들의 예

표 5

빈 인덱스	0	1	2	3	4	5	6	7	8
TU 4x4	0	1	2						
TU 8x8	3	4	5	6	7				
TU 16x16	8	9	10	11	11	12	12		
TU 32x32	13	14	14	15	16	16	16	12	12

[0089]

[0090] [표 5] 규칙들 1-4 에 따른 최종 위치 빈들의 예

표 6

빈 인덱스	0	1	2	3	4	5	6	7	8
TU 4x4	0	1	2						
TU 8x8	3	4	5	6	7				
TU 16x16	8	9	10	10	11	11	12		
TU 32x32	13	14	14	15	15	15	16	16	16

[0091]

[0092] [표 6] 규칙들 1-4 에 따른 최종 위치 빈들의 예

[0093] 아래 표들 7-8 은 컨텍스트들의 개수를 추가로 최적화할 수 있는, 상이한 블록 사이즈들로부터의 최종 빈들이 동일한 컨텍스트를 공유하는 예시적인 컨텍스트 할당들을 제공한다. 일 예에서, 어떻게 컨텍스트들이 2개의 이상 블록 사이즈들의 최종 빈들 사이에 공유되는지를 결정하는데 직접 맵핑이 사용될 수도 있다. 예를 들어, 사이즈들 M 및 N 를 각각 가진 블록 A 및 블록 B 에 있어, 블록 A 의 n-번째 빈의 컨텍스트는 블록 사이즈

B 의 n-번째 빈과 동일한 컨텍스트를 이용할 수도 있다

표 7

빈 인덱스	0	1	2	3	4	5	6	7	8
TU 4x4	12	13	14						
TU 8x8	12	13	14	14	15				
TU 16x16	12	13	14	14	15	15	16		
TU 32x32	12	13	14	14	15	15	16	16	17

[표 7] 블록 사이즈들이 동일한 컨텍스트들을 공유하는 최종 위치 빈들의 예.

표 8 은 일부 블록 사이즈들로부터의 최종 위치 빈들이 그 컨텍스트들을 서로 공유하는 또 다른 예를 나타낸다. 이 경우, 사이즈 8x8 및 16x16 의 TU들은 동일한 컨텍스트들을 공유한다.

표 8

빈 인덱스	0	1	2	3	4	5	6	7	8
TU 4x4	0	1	2						
TU 8x8	7	8	9	9	10				
TU 16x16	7	8	9	9	10	10	11		
TU 32x32	12	13	14	14	15	15	16	16	17

[표 8] 일부 블록 사이즈들 (8x8 및 16x16) 이 동일한 컨텍스트들을 공유하는 최종 위치 빈들의 예.

또 다른 예에서, 상이한 블록 사이즈들로부터의 최종 위치 빈들에 대한 컨텍스트들의 맵핑은 함수 $f(\cdot)$ 를 이용하여 유도될 수도 있다. 예를 들어, 블록 사이즈 A 에서의 n-번째 빈은 블록 사이즈 B 에서의 m-번째 빈과 동일한 컨텍스트들을 공유할 수도 있으며, 여기서, m 은 n 의 함수이다 ($m = f(n)$). 예를 들어, 함수는 선형, 즉, $m = n \cdot a + b$ 일 수 있으며, 여기서, a 및 b 는 선형 함수의 파라미터들이다. 표 9 은 $a = 1$, $b = 1$, $A = 8x8$ TU 및 $B = 16x16$ TU 인 예를 나타낸다.

표 9

빈 인덱스	0	1	2	3	4	5	6	7	8
TU 4x4	0	1	2						
TU 8x8	8	9	9	10	10				
TU 16x16	7	8	9	9	10	10	11		
TU 32x32	12	13	14	14	15	15	16	16	17

[표 9] 선형 함수에 기초한 공유된 컨텍스트들을 가진 최종 위치들 빈들의 예

어떤 경우들에서 상기 수식을 적용할 때, 정수 연산으로 인해, 사사오입이 수반될 수도 있다는 점에 유의해야 한다. 예를 들어, $7 \cdot 0.5 = 3$ 이다.

다음 예에 따르면, 8x8 블록 사이즈에서의 로케이션 n 으로부터 4x4 블록에서의 로케이션 m 으로의 맵핑은 다음 수식으로 계산될 수도 있다:

[0104] $m = f(n) = n \gg 1$, 이것은 $a = 0.5$, $b = 0$, $A = 8 \times 8$, $B = 4 \times 4$ 를 의미한다 (1)

[0105] 16×16 블록에서의 로케이션 n 으로부터 4×4 블록에서의 로케이션 m 으로의 맵핑은 다음 수식으로 계산될 수 있다:

[0106] $m = f(n) = n \gg 2$, 이것은 $a = 0.5$, $b = 0$, $A = 16 \times 16$, $B = 4 \times 4$ 를 의미한다 (2)

[0107] 위에서 설명한 바와 같이, 수식들 (1) 및 (2) 는 단지 상이한 사이즈들의 블록들 사이의 맵핑을 구현하는데 사용될 수도 있는 예들의 쌍이다. 수식들 (1) 및 (2) 는 맵핑 함수들로서 지칭될 수도 있다. 수식들 (1) 및 (2) 에서 ">>" 는 HEVC 과 같은, 비디오 코딩 표준에 따라서 정의된 시프트 연산 (shift operation) 을 나타낼 수도 있다는 점에 유의해야 한다. 또, 다른 수식들이 동일한 맵핑 또는 상이한 맵핑들을 달성하는데 사용될 수도 있다.

[0108] 표 10 은 수식들 (1) 및 (2) 에 따라서 4×4 , 8×8 , 및 16×16 TU들에 있어 최종 유의 계수에 대한 예시적인 컨텍스트 할당을 제공한다.

표 10

빈 인덱스	0	1	2	3	4	5	6	7	8
TU 4×4	0	1	2						
TU 8×8	0	0	1	1	2				
TU 16×16	0	0	0	0	1	1	1		

[0109]

[표 10] 상이한 사이즈의 변환 유닛들에 대한 컨텍스트 맵핑

[0111] 표 11 은 상이한 컨텍스트 인덱스 값들이 사용되는 (즉, 0-2 대신, 15-17) 표 10 의 컨텍스트 할당의 일 예를 제공한다. 위에서 설명한 바와 같이, 표들 3-12 에서의 컨텍스트 인덱스들의 값들은 이전 인덱스에 할당되는 실제 컨텍스트들을 한정하려고 의도된 것이 아니다.

표 11

빈 인덱스	0	1	2	3	4	5	6	7	8
TU 4×4	15	16	17						
TU 8×8	15	15	16	16	17				
TU 16×16	15	15	15	15	16	16	16		

[0112]

[표 11] 상이한 사이즈의 변환 유닛들에 대한 컨텍스트 맵핑

[0114] 표 11 에서의 컨텍스트들의 맵핑은 다음 맵핑 함수와 동등하다는 점에 유의해야 한다:

[0115]
$$ctx_index = (n \gg k) + 15 \quad (3)$$

[0116] 여기서, ctx_index 는 컨텍스트의 인덱스이며;

[0117] n = 빈 인덱스

[0118] $k = \log_2 TrafoDimension - 2$;

[0119] x 치수에서 최종 위치에 대해 $\log_2 TrafoDimension = \log_2 (\text{폭})$;

[0120] y 치수에서 최종 위치에 대해 $\log_2 \text{TrafoDimension} = \log_2(\text{높이})$.

[0121] 일부 경우, 메모리에 저장되어 컨텍스트 할당들을 탐색하는데 이용될 수도 있는 일련의 표들을 형성하기 위해 (1)-(3) 에서 정의된 함수가 코딩 디바이스에 의해 사용될 수도 있다. 일부의 경우, 표들은 본원에서 설명되는 수식들 및 규칙들에 기초하여 미리 결정되어, 비디오 인코더 (20) 및 비디오 디코더 (30) 양자에 저장될 수도 있다.

[0122] 또, 일부 예들에서, 특정의 빈들에 대한 컨텍스트들을 할당하는데 위에서 정의된 함수들 (1)-(3) 이 선택적으로 적용될 수도 있다. 이러한 방법으로, 상이한 빈들에 상이한 규칙에 기초하여 컨텍스트가 할당될 수도 있다. 일 예에서, 위에서 설명한 것들과 같은, 함수들은 임계치 Th1 보다 더 작거나, 및/또는 임계치 Th2 보다 더 큰 빈 인덱스 (즉, n 의 값) 에 대해 오직 적용할 수도 있다. 표 12 는 위에서 설명한 맵핑 기법들이 빈 인덱스의 값, 즉, $n > \text{Th2} = 2$ 에 기초하여 선택적으로 적용되는 예를 나타낸다.

표 12

빈 인덱스	0	1	2	3	4	5	6	7	8
TU 4x4	0	1	2						
TU 8x8	3	4	9	10	10				
TU 16x16	7	8	9	9	10	10	11		
TU 32x32	12	13	14	14	15	15	16	16	17

[0123]

[0124] [표 12] 선형 함수 및 임계치에 기초한 공유된 컨텍스트들을 가진 최종 위치 빈들의 예

[0125] 또 다른 예에서, 빈 인덱스들에 기법들을 적용하는 임계값은 상이한 블록 사이즈들, 상이한 프레임 유형들, 상이한 칼라 성분들 (Y,U,V), 및/또는 다른 측면 정보에 대해 상이할 수 있다. 이 임계치는 비디오 코딩 표준에 따라 사전-정의될 수 있거나 또는 하이 레벨 구문을 이용하여 시그널링될 수 있다. 예를 들어, 임계치는 시퀀스 파라미터 세트 (SPS), 화상 파라미터 세트 (PPS), 적응 파라미터 세트 (APS), 및/또는 슬라이스 헤더로 시그널링될 수도 있다.

[0126] 또 다른 예에서, 맵핑 함수는 상이한 블록 사이즈들, 상이한 프레임 유형들, 상이한 칼라 성분들 (Y, U, 및 V), 및/또는 다른 측면 정보에 대해 상이할 수도 있다. 맵핑 함수는 비디오 코딩 표준에 따라 사전-정의될 수 있거나 또는 하이 레벨 구문을 이용하여 시그널링될 수 있다. 예를 들어, 맵핑 함수는 SPS, PPS, APS, 및/또는 슬라이스 헤더로 시그널링될 수도 있다.

[0127] 또 다른 예에서, 위에서 설명한 직접 맵핑 및 함수 맵핑 기법들은 칼라 성분들, 프레임 유형, 양자화 파라미터 (QP) 및/또는 다른 측면 정보에 기초하여, 적응적으로 적용될 수도 있다. 예를 들어, 직접 맵핑 또는 함수 맵핑 기법들은 크로마 성분들에 오직 적용될 수도 있다. 이 적응성에 대한 규칙들은 사전-정의될 수 있거나 또는 하이 레벨 구문을 이용하여 시그널링될 수 있다. 예를 들어, 적응성에 대한 규칙들은 SPS, PPS, APS, 및/또는 슬라이스 헤더로 시그널링될 수도 있다.

[0128] 또 다른 예에서, 크로마 및 루마 성분들에 대한 최종 위치 빈들은 동일한 컨텍스트들을 공유할 수 있다. 이것은 임의의 블록 사이즈, 예컨대, 4x4, 8x8, 16x16, 또는 32x32 에 대해 적용할 수도 있다. 표 13 은 컨텍스트들이 4x4 TU 에 있어 루마 및 크로마 성분들의 최종 위치 빈들에 대해 공유되는 예를 나타낸다.

표 13

빈 인덱스	0	1	2	3
루마 TU 4x4	0	1	2	
크로마 TU 4x4	0	1	2	

[0129]

- [0130] [표 13] 4x4 TU 에서 루마 및 크로마 성분들에 대한 최종 위치 bin들의 예
- [0131] 도 4 는 본 개시물에서 설명되는 바와 같은 변환 계수들을 코딩하는 방법들을 이용할 수도 있는 비디오 인코더 (20) 의 일 예를 예시하는 블록도이다. 예를 들어, 비디오 인코더 (20) 는 비디오 블록 내 최종 유의 계수의 위치를 나타내는 이진 스트링을 획득하고; 비디오 블록 사이즈에 기초하여, 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하고; 그리고, 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 이진 스트링을 인코딩하도록 구성되는 비디오 인코더의 일 예를 나타내며, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관된다. 비디오 인코더 (20) 는 HEVC 코딩의 상황에서, 예시의 목적을 위해, 그러나 변환 계수들의 스케닝을 필요로 할 수도 있는 다른 코딩 표준들 또는 방법들에 관한 본 개시물의 한정 없이 설명될 것이다. 비디오 인코더 (20) 는 비디오 프레임들 내 CU들의 인트라-코딩 및 인터-코딩을 수행할 수도 있다. 인트라 코딩은 주어진 비디오 프레임 내 비디오 데이터에서 공간 리던던시를 감소시키거나 또는 제거하기 위해 공간 예측에 의존한다. 인터-코딩은 비디오 시퀀스의 현재의 프레임과 이전에 코딩된 프레임들 사이에 시간 리던던시를 감소시키거나 또는 제거하기 위해 시간 예측에 의존한다. 인트라-모드 (I-모드) 는 여러 공간-기반의 비디오 압축 모드들 중 임의의 모드를 지칭할 수도 있다. 단방향 예측 (P-모드) 또는 양방향 예측 (B-모드) 와 같은 인터-모드들은 여러 시간-기반의 비디오 압축 모드들 중 임의의 모드를 지칭할 수도 있다.
- [0132] 도 4 에 나타난 바와 같이, 비디오 인코더 (20) 는 인코딩될 비디오 프레임 내에서 현재의 비디오 블록을 수신한다. 도 4 의 예에서, 비디오 인코더 (20) 는 모드 선택 모듈 (40), 모션 추정 모듈 (42), 모션 보상 모듈 (44), 인트라-예측 모듈 (46), 참조 프레임 버퍼 (64), 합산기 (50), 변환 모듈 (52), 양자화 모듈 (54), 및 엔트로피 인코딩 모듈 (56) 을 포함한다. 도 4 에 예시된 변환 모듈 (52) 은 실제 변환 또는 변환의 조합들을 잔여 데이터의 블록에 적용하는 모듈이며, CU 의 변환 유닛 (TU) 으로서 또한 지칭될 수도 있는 변환 계수들의 블록과 혼동되지 않아야 한다. 비디오 블록 재구성을 위해, 비디오 인코더 (20) 는 또한 역양자화 모듈 (58), 역변환 모듈 (60), 및 합산기 (62) 를 포함한다. 디블로킹 필터 (도 4 에 미도시) 는 또한 블록 경계들을 필터링하여 재구성된 비디오로부터 블로킹 현상 아티팩트들을 제거하기 위해 포함될 수도 있다. 원할 경우, 디블로킹 필터는 일반적으로 합산기 (62) 의 출력을 필터링할 것이다.
- [0133] 인코딩 프로세스 동안, 비디오 인코더 (20) 는 코딩될 비디오 프레임 또는 슬라이스를 수신한다. 프레임 또는 슬라이스는 최대 코딩 유닛들 (LCU들) 과 같은 다수의 비디오 블록들로 분할될 수도 있다. 모션 추정 모듈 (42) 및 모션 보상 모듈 (44) 은 하나 이상의 참조 프레임들에서 하나 이상의 블록들에 대해 그 수신된 비디오 블록의 인트라-예측 코딩을 수행하여, 시간 압축을 제공한다. 인트라-예측 모듈 (46) 은 코딩될 블록과 동일한 프레임 또는 슬라이스에서 하나 이상의 이웃하는 블록들에 대해서 그 수신된 비디오 블록의 인트라-예측 코딩을 수행하여, 공간 압축을 제공할 수도 있다.
- [0134] 모드 선택 모듈 (40) 은 예컨대, 각각의 모드에 대한 에러 (즉, 왜곡) 결과들에 기초하여, 코딩 모드들 중 하나, 즉, 인트라 또는 인터 모드를 선택하고, 최종 인트라- 또는 인터-예측된 블록 (예컨대, 예측 유닛 (PU)) 을 합산기 (50) 에 제공하여 잔여 블록 데이터를 발생하고, 그리고 합산기 (62) 에 제공하여 참조 프레임에서의 사용을 위해, 인코딩된 블록을 재구성할 수도 있다. 합산기 (62) 는 아래에서 더 자세히 설명하는 바와 같이 예측된 블록을 그 블록에 대한 역변환 모듈 (60) 로부터의 역 양자화된, 역변환된 데이터와 합산하여, 인코딩된 블록을 재구성한다. 일부 비디오 프레임들은 I-프레임들로서 지정될 수도 있으며, 여기서, I-프레임에서의 모든 블록들은 인트라-예측 모드에서 인코딩된다. 일부의 경우, 인트라-예측 모듈 (46) 은 예컨대, 모션 추정 모듈 (42) 에 의해 수행되는 모션 탐색이 블록의 충분한 예측을 초래하지 않을 때, P- 또는 B-프레임에서 블록의 인트라-예측 인코딩을 수행할 수도 있다.
- [0135] 모션 추정 모듈 (42) 및 모션 보상 모듈 (44) 은 고도로 통합될 수도 있지만, 개념적인 목적들을 위해 별개로 예시된다. 모션 추정 (또는, 모션 탐색) 은 모션 벡터들을 발생하는 프로세스이며, 이 프로세스는 비디오 블록들에 대한 모션을 추정한다. 모션 벡터는, 예를 들어, 참조 프레임의 참조 샘플에 대한, 현재의 프레임에서의 예측 유닛의 변위를 나타낼 수도 있다. 모션 추정 모듈 (42) 은 예측 유닛을 참조 프레임 버퍼 (64) 에 저장된 참조 프레임의 참조 샘플들의 참조 샘플들과 비교함으로써, 인터-코딩된 프레임의 예측 유닛에 대한 모션 벡터를 계산한다. 참조 샘플은 코딩중인 PU 를 포함하는 CU 의 부분에 픽셀 차이의 관점에서 가깝게 매칭하는 것으로 발견되는 블록일 수도 있으며, 이것은 SAD (sum of absolute difference), SSD (sum of squared difference), 또는 다른 차이 메트릭들에 의해 결정될 수도 있다. 참조 샘플은 참조 프레임 또는 참조 슬라이스 내 어디서든지 발생할 수도 있으며, 반드시 참조 프레임 또는 슬라이스의 블록 (예컨대, 코딩 유닛) 경계에서 발생하지 않을 수도 있다. 일부 예들에서, 참조 샘플은 분수 픽셀 위치에서 발생할 수도

있다.

- [0136] 모션 추정 모듈 (42) 은 계산된 모션 벡터를 엔트로피 인코딩 모듈 (56) 및 모션 보상 모듈 (44) 로 전송한다. 모션 벡터에 의해 식별된 참조 프레임의 부분은 참조 샘플로서 지칭될 수도 있다. 모션 보상 모듈 (44) 은 현재의 CU 의 예측 유닛에 대한 예측 값을, 예컨대, PU 에 대한 모션 벡터에 의해 식별되는 참조 샘플을 취출함으로써, 계산할 수도 있다.
- [0137] 인트라-예측 모듈 (46) 은 모션 추정 모듈 (42) 및 모션 보상 모듈 (44) 에 의해 수행되는 인터-예측에 대한 대안으로서, 수신된 블록을 인트라-예측할 수도 있다. 인트라-예측 모듈 (46) 은 블록들에 대한 좌우, 상하 인코딩 순서를 가정할 때, 이웃하는, 이전에 코딩된 블록들, 예컨대, 현재의 블록의 상부, 상부 및 우측, 상부 및 좌측, 또는 좌측 블록들에 대해, 수신된 블록을 예측할 수도 있다. 인트라-예측 모듈 (46) 은 다양한 상이한 인트라-예측 모드들로 구성될 수도 있다. 예를 들어, 인트라-예측 모듈 (46) 은 인코딩되는 CU 의 사이즈에 기초하여, 어떤 수의 방향 예측 모드들, 예컨대, 34 개의 방향 예측 모드들로 구성될 수도 있다.
- [0138] 인트라-예측 모듈 (46) 은 인트라-예측 모드를, 예를 들어, 여러 인트라-예측 모드들에 대한 여러 값들을 계산하여 최저 여러 값을 산출하는 모드를 선택함으로써, 선택할 수도 있다. 방향 예측 모드들은 공간적으로 이웃하는 픽셀들의 값들을 결합하여 그 결합된 값들을 PU 내 하나 이상의 픽셀 위치들에 적용하는 기능들을 포함할 수도 있다. 일단 PU 내 모든 픽셀 위치들에 대한 값들이 계산되었으면, 인트라-예측 모듈 (46) 은 PU 와 인코딩되는 수신된 블록 사이의 픽셀 차이들에 기초하여, 예측 모드에 대한 여러 값을 계산할 수도 있다. 인트라-예측 모듈 (46) 은 허용가능한 여러 값을 산출하는 인트라-예측 모드가 발견될 때까지 인트라-예측 모드들을 테스트하는 것을 계속할 수도 있다. 인트라-예측 모듈 (46) 은 그후 PU 를 합산기 (50) 로 전송할 수도 있다.
- [0139] 비디오 인코더 (20) 는 코딩중인 원래 비디오 블록으로부터 모션 보상 모듈 (44) 또는 인트라-예측 모듈 (46) 에 의해 계산된 예측 데이터를 감산함으로써, 잔여 블록을 형성한다. 합산기 (50) 는 이 감산 동작을 수행하는 구성요소 또는 구성요소들을 나타낸다. 잔여 블록은 픽셀 차이 값들의 2차원 매트릭스에 대응할 수도 있으며, 여기서, 잔여 블록에서의 값들의 수는 잔여 블록에 대응하는 PU 내 픽셀들의 개수와 동일하다. 잔여 블록에서의 값들은 PU 에 그리고 코딩되는 원래 블록에서 동일 장소에 배치된 픽셀들의 값들 사이의 차이들, 즉, 에러에 대응할 수도 있다. 그 차이들은 코딩되는 블록의 유형에 따라서 크로마 또는 루마 차이들일 수도 있다.
- [0140] 변환 모듈 (52) 은 잔여 블록으로부터 하나 이상의 변환 유닛들 (TU들) 을 포함할 수도 있다. 변환 모듈 (52) 은 복수의 변환들 중에서 변환을 선택한다. 이 변환은 블록 사이즈, 코딩 모드 등과 같은 하나 이상의 코딩 특성들에 기초하여 선택될 수도 있다. 변환 모듈 (52) 은 그후 그 선택된 변환을 TU 에 적용하여, 변환 계수들의 2차원 어레이를 포함하는 비디오 블록을 발생한다.
- [0141] 변환 모듈 (52) 은 최종 변환 계수들을 양자화 모듈 (54) 로 전송할 수도 있다. 양자화 모듈 (54) 은 그후 변환 계수들을 양자화할 수도 있다. 엔트로피 인코딩 모듈 (56) 은 그후 양자화된 변환 계수들의 스캐닝을 매트릭스로 스캐닝 모드에 따라서 수행할 수도 있다. 본 개시물은 스캐닝을 수행하는 것으로 엔트로피 인코딩 모듈 (56) 을 설명한다. 그러나, 다른 예들에서, 양자화 모듈 (54) 과 같은, 다른 프로세싱 모듈들이 스캐닝을 수행할 수 있는 것으로 이해되어야 한다.
- [0142] 역양자화 모듈 (58) 및 역변환 모듈 (60) 은 역양자화 및 역변환을 각각 적용하여, 예컨대, 참조 블록으로 추후 사용을 위해, 픽셀 도메인에서 잔여 블록을 재구성한다. 모션 보상 모듈 (44) 은 잔여 블록을 참조 프레임 버퍼 (64) 의 프레임들 중 하나의 예측 블록에 가산함으로써, 참조 블록을 계산할 수도 있다. 참조 프레임 버퍼 (64) 는 종종 디코딩 화상 버퍼 (DPB) 로서 지칭된다. 모션 보상 모듈 (44) 은 또한 하나 이상의 내삽 필터들을 그 재구성된 잔여 블록에 적용하여, 모션 추정에 사용하기 위한 서브-정수 픽셀 값들을 계산할 수도 있다. 합산기 (62) 는 재구성된 잔여 블록을 모션 보상 모듈 (44) 에 의해 발생된 모션 보상된 예측 블록에 가산하여, 참조 프레임 버퍼 (64) 에의 저장을 위해 재구성된 비디오 블록을 발생한다. 재구성된 비디오 블록은 모션 추정 모듈 (42) 및 모션 보상 모듈 (44) 에 의해 후속 비디오 프레임에서의 블록을 인터-코딩하는데 참조 블록으로서 사용될 수도 있다.
- [0143] 일단 변환 계수들이 1차원 어레이로 스캐닝되면, 엔트로피 인코딩 모듈 (56) 은 CAVLC, CABAC, SBAC, PIPE, 또는 또 다른 엔트로피 코딩 방법과 같은 엔트로피 코딩을 계수들에 적용할 수도 있다. 일부의 경우, 엔트로피 인코딩 모듈 (56) 은 엔트로피 코딩에 더해서, 다른 코딩 기능들을 수행하도록 구성될 수도 있다. 예를

들어, 엔트로피 인코딩 모듈 (56)은 CU들 및 PU들에 대한 코딩된 블록 패턴 (CBP) 값들을 결정하도록 구성될 수도 있다. 또한, 일부의 경우, 엔트로피 인코딩 모듈 (56)은 계수들의 런 길이 코딩 (런 길이 코딩)을 수행할 수도 있다. 엔트로피 인코딩 모듈 (56)에 의한 엔트로피 코딩에 뒤이어서, 최종 인코딩된 비디오는 비디오 디코더 (30)와 같은 또 다른 디바이스로 송신되거나, 또는 추후 송신 또는 취출을 위해 아카이브될 수도 있다.

[0144] 본 개시물의 기법들에 따르면, 엔트로피 인코딩 모듈 (56)은 예를 들어, 표 2-13과 관련하여 위에서 설명한 컨텍스트 할당들, 및 인트라-예측 모드들에 대한 인트라-예측 방향, 구문 엘리먼트들에 대응하는 계수의 스캐닝 위치, 블록 유형, 변환 유형, 및/또는 다른 비디오 시퀀스 성질들의 임의의 조합에 기초하여, 구문 엘리먼트들을 인코딩하는데 사용되는 컨텍스트를 선택할 수도 있다.

[0145] 일 예에서, 엔트로피 인코딩 모듈 (56)은 표 1과 관련하여 위에서 설명한 HEVC에서 채택된 이진화 기법을 이용하여 최종 유의 계수의 위치를 인코딩할 수도 있다. 다른 예들에서, 엔트로피 인코딩 모듈 (56)은 다른 이진화 기법들을 이용하여 최종 유의 계수의 위치를 인코딩할 수도 있다. 일 예에서, 최종 유의 계수의 위치에 대한 코드워드는 고정된 길이 코드 접미사가 뒤따르는 트렁케이트된 일진수 코드 접두사를 포함할 수도 있다. 일 예에서, 최종 위치의 각각의 크기는 최종 위치가 TU 사이즈 마이너스 1과 동일할 때를 제외하고, 모든 가능한 TU 사이즈들에 대해 동일한 이진화를 이용할 수도 있다. 이 예외는 트렁케이트된 일진수 코딩의 성질들에 기인한다. 일 예에서, 직사각형의 변환 계수 내 최종 유의 계수의 위치는 x-좌표 값 및 y-좌표 값을 규정함으로써 규정될 수도 있다. 또 다른 예에서, 변환 계수 블록은 1xN 벡터의 유형일 수도 있으며, 그 벡터 내 최종 유의 계수의 위치는 단일 위치 값에 의해 규정될 수도 있다.

[0146] 도 5는 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 엔트로피 인코딩 모듈 (56)을 예시하는 블록도이다. 일 예에서, 도 5에 예시된 엔트로피 인코딩 모듈 (56)은 CABAC 인코더일 수도 있다. 예시적인 엔트로피 인코딩 모듈 (56)은 이진화 모듈 (502), 산술 인코딩 모듈 (510)을 포함할 수도 있으며, 이 산술 인코딩 모듈은 바이패스 인코딩 엔진 (504) 및 정규 인코딩 엔진 (regular encoding engine; 508), 및 컨텍스트 모델링 모듈 (506)을 포함한다. 엔트로피 인코딩 모듈 (56)은 변환 블록 계수들 내 최종 유의 변환 계수의 위치를 나타내는 하나 이상의 구문 엘리먼트들과 같은, 구문 엘리먼트들을 수신하여, 그 구문 엘리먼트를 비트스트림으로 인코딩한다. 구문 엘리먼트들은 변환 계수 블록 내 최종 유의 계수의 위치의 x-좌표를 규정하는 구문 엘리먼트 및 변환 계수 블록 내 최종 유의 계수의 위치의 y-좌표를 규정하는 구문 엘리먼트를 포함할 수도 있다.

[0147] 이진화 모듈 (502)은 구문 엘리먼트를 수신하여, 빈 스트링 (즉, 이진 스트링)을 발생한다. 일 예에서, 이진화 모듈 (502)은 변환 계수들의 블록 내 유의 계수의 최종 위치를 나타내는 구문 엘리먼트들을 수신하여, 표 1과 관련하여 위에서 설명한 예에 따라서 빈 스트링을 발생한다. 산술 인코딩 모듈 (510)은 이진화 모듈 (502)로부터 빈 스트링을 수신하여, 그 빈 스트링 상에서 산술적 인코딩을 수행한다. 도 5에 나타낸 바와 같이, 산술 인코딩 모듈 (510)은 바이패스 경로 또는 정규 코딩 경로로부터 빈 값들을 수신할 수도 있다. 위에서 설명된 CABAC 프로세스와 부합하여, 산술 인코딩 모듈 (510)이 바이패스 경로로부터 빈 값들을 수신하는 경우, 바이패스 인코딩 엔진 (504)은 빈 값에 할당된 컨텍스트를 이용함이 없이, 빈 값들 상에서 산술적 인코딩을 수행할 수도 있다. 일 예에서, 바이패스 인코딩 엔진 (504)은 빈의 가능한 값들에 대해 동일한 확률들을 가정할 수도 있다.

[0148] 산술 인코딩 모듈 (510)이 정규 경로를 통해서 빈 값들을 수신하는 경우, 컨텍스트 모델링 모듈 (506)은 정규 인코딩 엔진 (508)이 컨텍스트 모델링 모듈 (506)에 의해 제공되는 컨텍스트 할당들에 기초하여 산술적 인코딩을 수행할 수 있도록, 컨텍스트 변수 (예컨대, 컨텍스트 상태)를 제공할 수도 있다. 일 예에서, 산술 인코딩 모듈 (510)은 컨텍스트 할당을 이용하여 비트 스트링의 접두사 부분을 인코딩할 수도 있으며, 컨텍스트 할당들을 이용함이 없이 비트 스트링의 접미사 부분을 인코딩할 수도 있다. 컨텍스트 할당들은 표들 2-13과 관련하여 위에서 설명한 예들에 따라서 정의될 수도 있다. 컨텍스트 모델들은 메모리에 저장될 수도 있다. 컨텍스트 모델링 모듈 (506)은 일련의 인덱싱된 표들을 포함하거나 및/또는 맵핑 함수들을 이용하여 특정의 빈에 대한 컨텍스트 및 컨텍스트 변수를 결정할 수도 있다. 빈 값을 인코딩한 후, 정규 인코딩 엔진 (508)은 실제 빈 값들에 기초하여 컨텍스트를 업데이트하여, 그 인코딩된 빈 값을 비트스트림의 부분으로서 출력할 수도 있다. 이러한 방법으로, 엔트로피 인코딩 모듈은 컨텍스트 할당 본원에서 설명하는 기법들에 기초하여, 하나 이상의 구문 엘리먼트들을 인코딩하도록 구성된다.

[0149] 도 6은 본 개시물의 기법에 따른, 최종 유의 계수의 위치를 나타내는 이진 스트링 값에 대한 컨텍스트를 결정

하는 예시적인 방법을 예시하는 플로우차트이다. 도 6 에서 설명되는 방법은 본원에서 설명되는 예시적인 비디오 인코더들 또는 엔트로피 인코더들 중 임의의 인코더에 의해 수행될 수도 있다. 단계 (602) 에서, 비디오 블록 내 최종 유의 변환 계수의 위치를 나타내는 이진 스트링이 획득된다. 이진 스트링은 표 1 과 관련하여 설명된 이진화 방식에 따라서 정의될 수도 있다. 단계 (604) 에서, 컨텍스트가 이진 스트링의 빈 값에 대해 결정된다. 컨텍스트가 본원에서 설명되는 기법들에 기초하여 빈들에 할당될 수도 있다. 컨텍스트는 룩업 테이블에 액세스하거나 또는 맵핑 함수를 실행하는 비디오 또는 엔트로피 인코더에 의해 결정될 수도 있다. 컨텍스트는 특정의 빈에 대한 특정의 컨텍스트 변수를 유도하는데 사용될 수도 있다. 컨텍스트 변수는 64 개의 가능한 확률들 (상태들) 중 하나 및 가장 가능성있는 상태 (예컨대, "1" 또는 "0") 를 나타내는 7-비트 이진 값일 수도 있다. 위에서 설명한 바와 같이, 일부의 경우, 빈들은 위에서 설명한 맵핑 함수들 및 표들 2-13 에 따라서 컨텍스트들을 공유할 수도 있다. 단계 (606) 에서, 빈 값이 CABAC 과 같은, 컨텍스트 변수를 이용하는 산술적 인코딩 프로세스를 이용하여 인코딩된다. 빈들이 컨텍스트들을 공유할 때, 하나의 빈의 값이 컨텍스트 적응적 인코딩 기법들에 따라서 후속 빈을 인코딩하는데 사용되는 컨텍스트 변수의 값에 영향을 미칠 수도 있다는 점에 유의해야 한다. 예를 들어, 특정의 빈이 "1" 이면, 후속 빈은 1 인 증가된 확률에 기초하여 인코딩될 수도 있다. 이러한 방법으로, 이진 스트링을 엔트로피 인코딩하는 것은 컨텍스트 모델의 컨텍스트 상태를 업데이트하는 것을 포함할 수도 있다. 또, 일부 예들에서, 컨텍스트 모델들은 슬라이스 내 빈들의 값들이 후속 슬라이스 내 빈들의 인코딩에 영향을 미칠 수 없도록, 슬라이스 레벨 상에서 초기화될 수도 있다는 점에 유의해야 한다.

[0150] 도 7 은 본 개시물에서 설명되는 바와 같은 변환 계수들을 코딩하는 기법들을 이용할 수도 있는 비디오 디코더 (30) 의 일 예를 예시하는 블록도이다. 예를 들어, 비디오 디코더 (30) 는 비디오 블록 내 최종 유의 계수의 위치를 나타내는 인코딩된 이진 스트링을 획득하고; 비디오 블록 사이즈에 기초하여, 인코딩된 이진 스트링의 이진 인덱스에 대한 컨텍스트를 결정하고; 그리고, 결정된 컨텍스트에 적어도 부분적으로 기초하여, CABAC 를 이용하여 그 인코딩된 이진 스트링을 디코딩하도록 구성되는 비디오 디코더의 일 예를 나타내며, 상기 인코딩된 이진 스트링은 CABAC 를 이용하여 인코딩되며, 상기 컨텍스트는 적어도 2개의 이진 인덱스들에 할당되며, 상기 적어도 2개의 이진 인덱스들의 각각은 상이한 비디오 블록 사이즈들과 연관된다.

[0151] 도 7 의 예에서, 비디오 디코더 (30) 는 엔트로피 디코딩 모듈 (70), 모션 보상 모듈 (72), 인트라-예측 모듈 (74), 역양자화 모듈 (76), 역변환 모듈 (78), 참조 프레임 버퍼 (82) 및 합산기 (80) 를 포함한다. 비디오 디코더 (30) 는 일부 예들에서, 비디오 인코더 (20) 과 관련하여 설명한 인코딩 과정과는 일반적으로 반대인 디코딩 과정을 수행할 수도 있다.

[0152] 엔트로피 디코딩 모듈 (70) 은 인코딩된 비트스트림 상에 엔트로피 디코딩 프로세스를 수행하여, 변환 계수들의 1차원 어레이를 추출한다. 사용되는 엔트로피 디코딩 프로세스는 비디오 인코더 (20) 에 의해 사용되는 엔트로피 코딩 (예컨대, CABAC, CAVLC 등) 에 의존한다. 인코더에 의해 사용되는 엔트로피 코딩 프로세스는 인코딩된 비트스트림으로 시그널링될 수도 있거나 또는 미리 결정된 프로세스일 수도 있다.

[0153] 일부 예들에서, 엔트로피 디코딩 모듈 (70) (또는, 역양자화 모듈 (76)) 은 그 수신된 값들을 비디오 인코더 (20) 의 엔트로피 인코딩 모듈 (56) (또는, 양자화 모듈 (54)) 에 의해 사용되는 스캐닝 모드를 미러링하는 스캐닝을 이용하여, 스캐닝할 수도 있다. 계수들의 스캐닝은 역양자화 모듈 (76) 에서 수행될 수도 있지만, 스캐닝은 예시의 목적을 위해 엔트로피 디코딩 모듈 (70) 에 의해 수행되는 것으로 설명될 것이다. 게다가, 예시의 용이를 위해 별개의 기능 모듈들로서 나타내지만, 엔트로피 디코딩 모듈 (70), 역양자화 모듈 (76), 및 비디오 디코더 (30) 의 다른 모듈들의 구조 및 기능은 서로 고도로 통합될 수도 있다.

[0154] 역양자화 모듈 (76) 은 비트스트림으로 제공되어 엔트로피 디코딩 모듈 (70) 에 의해 디코딩되는 양자화된 변환 계수들을 역양자화한다, 즉 양자화 해제한다. 역양자화 프로세스는 종래의 프로세스, 예컨대, HEVC 에 대해 제안되거나 또는 H.264 디코딩 표준에 의해 정의된 프로세스들과 유사한, 프로세스를 포함할 수도 있다. 역양자화 프로세스는 양자화의 정도를 결정하기 위해, 그리고, 이와 유사하게, 적용되어야 하는 역양자화의 정도를 결정하기 위해, CU 에 대해 비디오 인코더 (20) 에 의해 계산된 양자화 파라미터 QP 의 사용을 포함할 수도 있다. 역양자화 모듈 (76) 은 계수들이 1차원 어레이로부터 2차원 어레이로 변환되기 전후에 변환 계수들을 역양자화할 수도 있다.

[0155] 역변환 모듈 (78) 은 역 양자화된 변환 계수들에 역변환을 적용한다. 일부 예들에서, 역변환 모듈 (78) 은 비디오 인코더 (20) 로부터의 시그널링에 기초하여, 또는 블록 사이즈, 코딩 모드 등과 같은 하나 이상의 코딩 특성들로부터 변환을 추론함으로써, 역변환을 결정할 수도 있다. 일부 예들에서, 역변환 모듈 (78) 은 현재

의 블록을 포함하는 LCU 에 대한 쿼드트리 루트 노드에서의 시그널링된 변환에 기초하여 현재의 블록에 적용할 변환을 결정할 수도 있다. 이의 대안으로, 변환은 LCU 쿼드트리에서 잎-노드 CU 에 대한 TU 쿼드트리의 루트에서 시그널링될 수도 있다. 일부 예들에서, 역변환 모듈 (78) 은 캐스케이트된 역변환을 적용할 수도 있으며, 여기서, 역변환 모듈 (78) 은 2개의 이상의 역변환들을 디코딩되는 현재의 블록의 변환 계수들에 적용한다.

[0156] 인트라-예측 모듈 (74) 은 현재의 프레임의 이전에 디코딩된 블록들로부터, 시그널링된 인트라-예측 모드 및 데이터에 기초하여, 현재의 프레임의 현재의 블록에 대한 예측 데이터를 발생할 수도 있다. 모션 보상 모듈 (72) 은 인코딩된 비트스트림으로부터 모션 벡터, 모션 예측 방향 및 참조 인덱스를 추출할 수도 있다. 참조 예측 방향은 인터-예측 모드가 단방향 (예컨대, P 프레임) 또는 양방향 (B 프레임) 인지 여부를 나타낸다. 참조 인덱스는 어느 참조 프레임에 후보 모션 벡터가 기초하는지를 나타낸다. 추출된 모션 예측 방향, 참조 프레임 인덱스, 및 모션 벡터에 기초하여, 모션 보상 모듈 (72) 은 현재의 부분에 대한 모션 보상된 블록을 발생한다. 이들 모션 보상된 블록들은 본질적으로 잔여 데이터를 발생하는데 사용되는 예측 블록을 재생성한다.

[0157] 모션 보상 모듈 (72) 은 모션 보상된 블록들을 발생하며, 어쩌면 내삽 필터들에 기초하여 내삽을 수행할 수도 있다. 서브-픽셀 정밀도를 갖는 모션 추정에 이용되는 내삽 필터들에 대한 식별자들이 구문 엘리먼트들에 포함될 수도 있다. 모션 보상 모듈 (72) 은 비디오 블록의 인코딩 동안 비디오 인코더 (20) 에 의해 사용되는 것과 같은 내삽 필터들을 이용하여, 참조 블록의 서브-정수 픽셀들에 대한 내삽된 값들을 계산할 수도 있다. 모션 보상 모듈 (72) 은 수신된 구문 정보에 따라서 비디오 인코더 (20) 에 의해 사용되는 내삽 필터들을 결정하고, 그 내삽 필터들을 이용하여, 예측 블록들을 발생할 수도 있다.

[0158] 게다가, HEVC 예에서, 모션 보상 모듈 (72) 및 인트라-예측 모듈 (74) 은 (예컨대, 쿼드트리에 의해 제공되는) 구문 정보의 일부를 이용하여, 인코딩된 비디오 시퀀스의 프레임(들) 을 인코딩하는데 사용되는 LCU들의 사이즈들을 결정할 수도 있다. 모션 보상 모듈 (72) 및 인트라-예측 모듈 (74) 은 또한 구문 정보를 이용하여, 인코딩된 비디오 시퀀스의 프레임의 각각의 CU 가 분할되는지 (그리고, 마찬가지로, 서브-CU들 가 분할되는지) 를 기술하는 분할 정보를 결정할 수도 있다. 구문 정보는 또한 어떻게 각각의 분할이 인코딩되는지를 나타내는 모드들 (예컨대, 인트라- 또는 인터-예측, 그리고 인트라-예측에 있어, 인트라-예측 인코딩 모드), 각각의 인터-인코딩된 PU 에 대한 하나 이상의 참조 프레임들 (및/또는 참조 프레임들에 대한 식별자들을 포함하는 참조 리스트들), 및 인코딩된 비디오 시퀀스를 디코딩하기 위한 다른 정보를 포함할 수도 있다.

[0159] 합산기 (80) 는 잔여 블록들을 모션 보상 모듈 (72) 또는 인트라-예측 모듈 (74) 에 의해 발생된 대응하는 예측 블록들과 합산하여, 디코딩된 블록들을 형성한다. 원할 경우, 블로킹 현상 아티팩트들을 제거하기 위해, 디블로킹 필터가 또한 그 디코딩된 블록들을 필터링하는데 적용될 수도 있다. 디코딩된 비디오 블록들은 그후 참조 프레임 버퍼 (82) 에 저장될 수도 있으며, 그 참조 프레임 버퍼는 후속 모션 보상을 위해 참조 블록들을 제공하며 또한 (도 1 의 디스플레이 디바이스 (32) 와 같은) 디스플레이 디바이스 상에의 프리젠테이션을 위해 디코딩된 비디오를 발생한다. 참조 프레임 버퍼 (82) 는 또한 DPB 로서 지칭될 수도 있다.

[0160] 도 8 은 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 엔트로피 디코딩 모듈 (70) 을 예시하는 블록도이다. 엔트로피 디코딩 모듈 (70) 은 엔트로피 인코딩된 비트스트림을 수신하여, 그 비트스트림으로부터 구문 엘리먼트들을 디코딩한다. 일 예에서, 구문 엘리먼트들은 변환 블록 계수들 내 최종 유의 변환 계수의 위치를 나타낼 수도 있다. 구문 엘리먼트들은 변환 계수 블록 내 최종 유의 계수의 위치의 x-좌표를 규정하는 구문 엘리먼트 및 변환 계수 블록 내 최종 유의 계수의 위치의 y-좌표를 규정하는 구문 엘리먼트를 포함할 수도 있다. 일 예에서, 도 8 에 예시된 엔트로피 디코딩 모듈 (70) 은 CABAC 디코더일 수도 있다. 도 8 에서의 예시적인 엔트로피 디코딩 모듈 (70) 은 산술 디코딩 모듈 (702) 을 포함하며, 이 산술 디코딩 모듈들은 바이패스 디코딩 엔진 (704) 및 정규 디코딩 엔진 (706) 을 포함할 수도 있다. 예시적인 엔트로피 디코딩 모듈 (70) 은 또한 컨텍스트 모델링 유닛 (708) 및 역 이진화 모듈 (710) 을 포함한다. 예시적인 엔트로피 디코딩 모듈 (70) 은 도 5 와 관련하여 설명된 예시적인 엔트로피 인코딩 모듈 (56) 의 반대 기능들을 수행할 수도 있다. 이러한 방법으로 엔트로피 디코딩 모듈 (70) 은 본원에서 설명하는 컨텍스트 할당 기법들에 기초하여 엔트로피 디코딩을 수행할 수도 있다.

[0161] 산술 디코딩 모듈 (702) 은 인코딩된 비트 스트림을 수신한다. 도 8 에 나타낸 바와 같이, 산술 디코딩 모듈 (702) 은 인코딩된 빈 값들을 바이패스 경로 또는 정규 코딩 경로에 따라서 프로세싱할 수도 있다. 인코딩된 빈 값이 바이패스 경로 또는 정규 경로 경로에 따라서 프로세싱되어야 하는지 여부를 표시가 더 높은 레벨

구문을 가진 비트스트림으로 시그널링될 수도 있다. 위에서 설명한 CABAC 프로세스에 부합하여, 산술 디코딩 모듈 (702) 이 바이패스 경로로부터 빈 값들을 수신하는 경우, 바이패스 디코딩 엔진 (704) 은 빈 값에 할당된 컨텍스트를 이용함이 없이 빈 값들 상에서 산술적 디코딩을 수행할 수도 있다. 일 예에서, 바이패스 디코딩 엔진 (704) 은 빈의 가능한 값들에 대해 동일한 확률들을 가정할 수도 있다.

[0162] 산술 디코딩 모듈 (702) 이 정규 경로를 통해서 빈 값들을 수신하는 경우, 컨텍스트 모델링 모듈 (708) 은 정규 디코딩 엔진 (706) 이 컨텍스트 모델링 모듈 (708) 에 의해 제공되는 컨텍스트 할당들에 기초하여 산술적 디코딩을 수행할 수 있도록, 컨텍스트 변수를 제공할 수도 있다. 컨텍스트 할당들은 표들 2-13 과 관련하여 위에서 설명한 예들에 따라서 정의될 수도 있다. 컨텍스트 모델들은 메모리에 저장될 수도 있다. 컨텍스트 모델링 모듈 (708) 은 일련의 인덱싱된 표들을 포함하거나, 및/또는 맵핑 함수들을 이용하여, 인코딩된 비트스트림의 컨텍스트 및 컨텍스트 변수 부분을 결정할 수도 있다. 빈 값을 디코딩한 후, 정규 디코딩 엔진 (706) 은 디코딩된 빈 값들에 기초하여 컨텍스트를 업데이트할 수도 있다. 또, 역 이진화 모듈 (710) 은 빈 값 상에서 역 이진화를 수행하고, 빈 매칭 함수를 이용하여 빈 값이 유효한지 여부를 결정할 수도 있다. 역 이진화 모듈 (710) 은 또한 그 매칭 결정에 기초하여 컨텍스트 모델링 모듈을 업데이트할 수도 있다. 따라서, 역 이진화 모듈 (710) 은 컨텍스트 적응적 디코딩 기법에 따라서 구문 엘리먼트들을 출력한다. 이러한 방법으로, 엔트로피 디코딩 모듈 (70) 은 본원에서 설명하는 컨텍스트 할당 기법들에 기초하여 하나 이상의 구문 엘리먼트들을 디코딩하도록 구성된다.

[0163] 도 9 는 본 개시물의 기법에 따른, 이진 스트림으로부터 변환 계수 내 최종 유의 계수의 위치를 나타내는 값을 결정하는 예시적인 방법을 예시하는 플로우차트이다. 도 9 에서 설명되는 방법은 본원에서 설명되는 예시적인 비디오 디코더들 또는 엔트로피 디코딩 유닛들 중 임의의 것에 의해 수행될 수도 있다. 단계 (902) 에서, 인코딩된 비트스트림이 획득된다. 인코딩된 비트스트림은 메모리로부터 또는 송신을 통해서 추출될 수도 있다. 인코딩된 비트스트림은 CABAC 인코딩 프로세스 또는 또 다른 엔트로피 코딩 프로세스에 따라서 인코딩될 수도 있다. 단계 (904) 에서, 컨텍스트가 인코딩된 이진 스트림의 부분에 대해 결정된다. 컨텍스트는 본원에서 설명되는 기법들에 기초하여, 인코딩된 빈에 할당될 수도 있다. 컨텍스트는 룩업 테이블에 액세스하거나 또는 맵핑 함수를 실행하는 비디오 또는 엔트로피 디코더에 의해 결정될 수도 있다. 컨텍스트는 인코딩된 비트스트림으로 제공되는 더 높은 레벨 구문에 기초하여 결정될 수도 있다. 컨텍스트는 특정의 인코딩된 빈에 대한 특정의 컨텍스트 변수를 유도하는데 사용될 수도 있다. 위에서 설명한 바와 같이, 컨텍스트 변수는 64 개의 가능한 확률들 (상태들) 중 하나 및 가장 가능성있는 상태 (예컨대, "1" 또는 "0") 를 나타내는 7-비트 이진 값일 수도 있으며, 일부 경우들에서, 빈들은 컨텍스트들을 공유할 수도 있다. 단계 (906) 에서, 이진 스트림이 CABAC 과 같은, 컨텍스트 변수를 이용하는 산술적 디코딩 프로세스를 이용하여 디코딩된다. 이진 스트림은 빈 단위로 디코딩될 수도 있으며, 여기서 컨텍스트 모델은 각각의 빈을 디코딩한 후에 업데이트된다. 디코딩된 비트스트림은 인코딩된 비디오 데이터와 연관되는 변환 계수들을 디코딩하는데 추가로 사용되는 구문 엘리먼트들을 포함할 수도 있다. 이러한 방법으로, 위에서 설명한 기법들을 이용한, 특정의 빈들에의 컨텍스트들의 할당은 인코딩 비디오 데이터의 효율적인 디코딩을 위해 제공할 수도 있다.

[0164] 하나 이상의 예들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현되는 경우, 그 기능들은 하나 이상의 명령들 또는 코드들로서, 컴퓨터-판독가능 매체 상에 저장되거나 또는 컴퓨터-판독가능 매체를 통해서 송신될 수도 있으며, 하드웨어-기반의 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터-판독가능 매체는 컴퓨터-판독가능 저장 매체들을 포함할 수도 있으며, 이 컴퓨터-판독가능 저장 매체들은 데이터 저장 매체와 같은 유형의 매체, 또는 예컨대, 통신 프로토콜에 따라서 한 장소로부터 다른 장소로의 컴퓨터 프로그램의 전송을 용이하게 하는 임의의 매체를 포함한 통신 매체들에 대응한다. 이러한 방법으로, 컴퓨터-판독가능 매체들은 일반적으로 (1) 비일시성 유형의 컴퓨터-판독가능 저장 매체 또는 (2) 신호 또는 캐리어 파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체는 본 개시물에서 설명하는 기법들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 추출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 가용 매체들일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터-판독가능 매체를 포함할 수도 있다.

[0165] 일 예로서, 이에 한정하지 않고, 이런 컴퓨터-판독가능 저장 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광디스크 스토리지, 자기디스크 스토리지, 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 원하는 프로그램 코드를 명령들 또는 데이터 구조들의 형태로 저장하는데 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 접속이 컴퓨터-판독가능 매체로 적절히 지칭된다. 예를 들어, 동축 케이블, 광섬유 케이블, 이중 권선, 디지털 가입자 회선 (DSL), 또는 무선 기술들 예컨대 적외선,

무선, 및 마이크로파를 이용하여 명령들이 웹사이트, 서버, 또는 다른 원격 소스로부터 송신되는 경우, 동축 케이블, 광섬유 케이블, 이중 권선, DSL, 또는 무선 기술들 예컨대 적외선, 무선, 및 마이크로파가 그 매체의 정의에 포함된다. 그러나, 컴퓨터-관독가능 저장 매체 및 데이터 저장 매체는 접속부들, 캐리어 파들, 신호들, 또는 다른 일시적 매체를 포함하지 않고, 그 대신, 비-일시적인 유형의 저장 매체에 송신되는 것으로 해석되어야 한다. 디스크 (disk) 및 디스크 (disc) 는, 본원에서 사용할 때, 콤팩트 디스크 (CD), 레이저 디스크, 광 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크 및 블루-레이 디스크를 포함하며, 디스크들 (disks) 은 데이터를 자기적으로 보통 재생하지만, 디스크들 (discs) 은 레이저로 데이터를 광학적으로 재생한다. 앞에서 언급한 것들의 조합들이 또한 컴퓨터-관독가능 매체들의 범위 내에 포함되어야 한다.

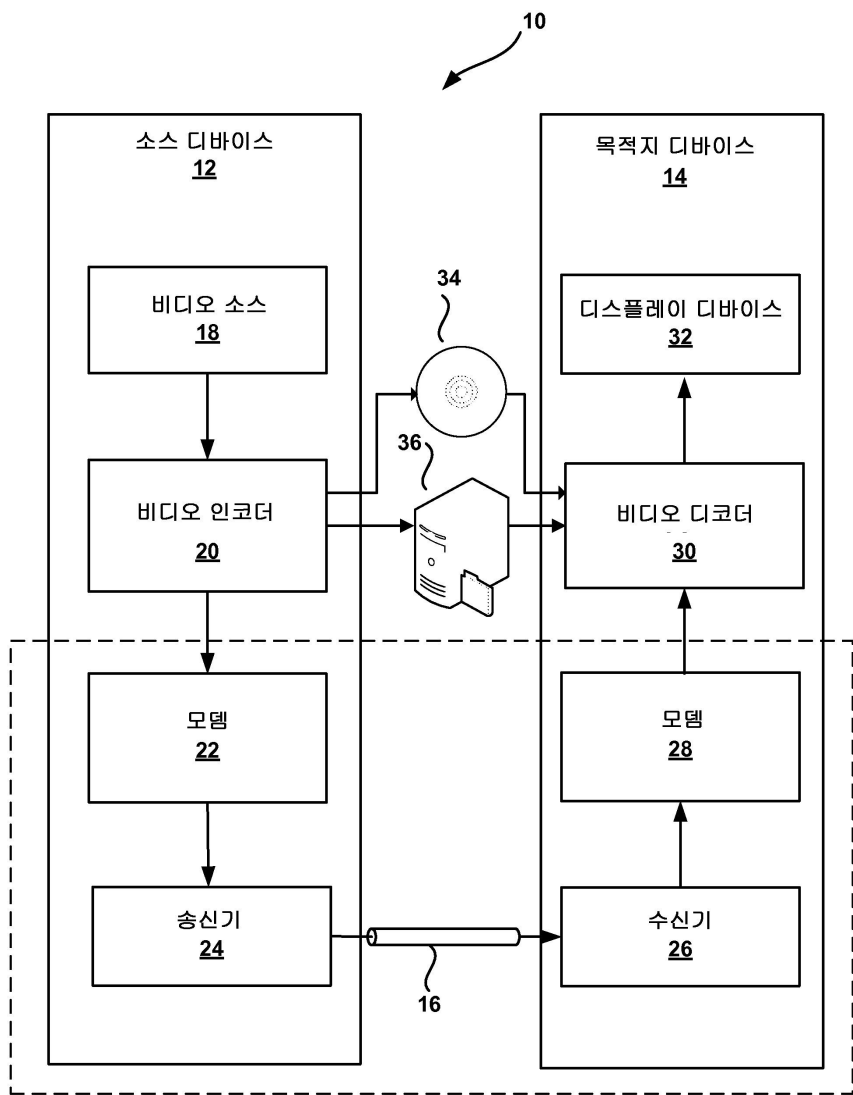
[0166] 명령들은 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로프로세서들, 주문형 집적회로들 (ASIC들), 필드 프로그래밍가능 로직 어레이들 (FPGA들), 또는 다른 등가의 통합 또는 이산 로직 회로와 같은, 하나 이상의 프로세서들에 의해 실행될 수도 있다. 따라서, 용어 “프로세서” 는, 본원에서 사용될 때 전술한 구조 중 임의의 구조 또는 본원에서 설명하는 기법들의 구현에 적합한 임의의 다른 구조를 지칭할 수도 있다. 게다가, 일부 양태들에서, 본원에서 설명하는 기능 전용 하드웨어 및/또는 인코딩 및 디코딩을 위해 구성되는 소프트웨어 모듈들 내에 제공되거나, 또는 결합된 코덱에 포함될 수도 있다. 또한, 이 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들로 전적으로 구현될 수 있다.

[0167] 본 개시물의 기법들은 무선 핸드셋, 집적 회로 (IC) 또는 IC들의 세트 (예컨대, 칩 세트) 를 포함한, 매우 다양한 디바이스들 또는 장치들로 구현될 수도 있다. 개시한 기법들을 수행하도록 구성되는 디바이스들의 기능적 양태들을 강조하기 위해서 여러 구성요소들, 모듈들, 또는 유닛들이 본 개시물에서 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 필요로 하지는 않는다. 더 정확히 말하면, 위에서 설명한 바와 같이, 여러 유닛들이 코덱 하드웨어 유닛에 결합되거나 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 위에서 설명한 바와 같은 하나 이상의 프로세서들을 포함한, 상호작용하는 하드웨어 유닛들의 컬렉션으로 제공될 수도 있다.

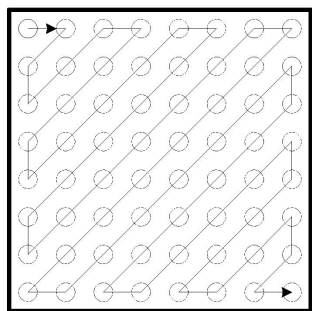
[0168] 여러 예들이 설명되었다. 이들 및 다른 예들은 다음 청구항들의 범위 이내이다.

도면

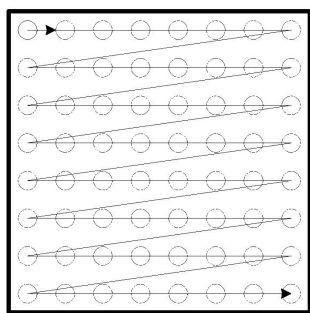
도면1



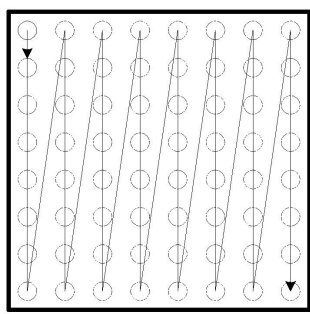
도면2a



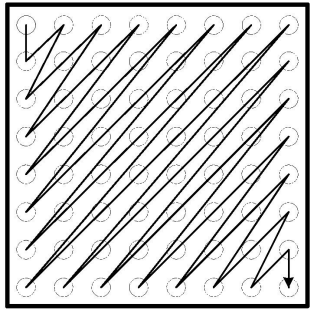
도면2b



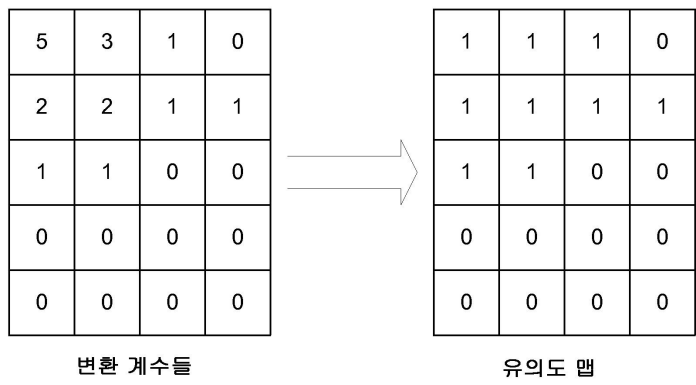
도면2c



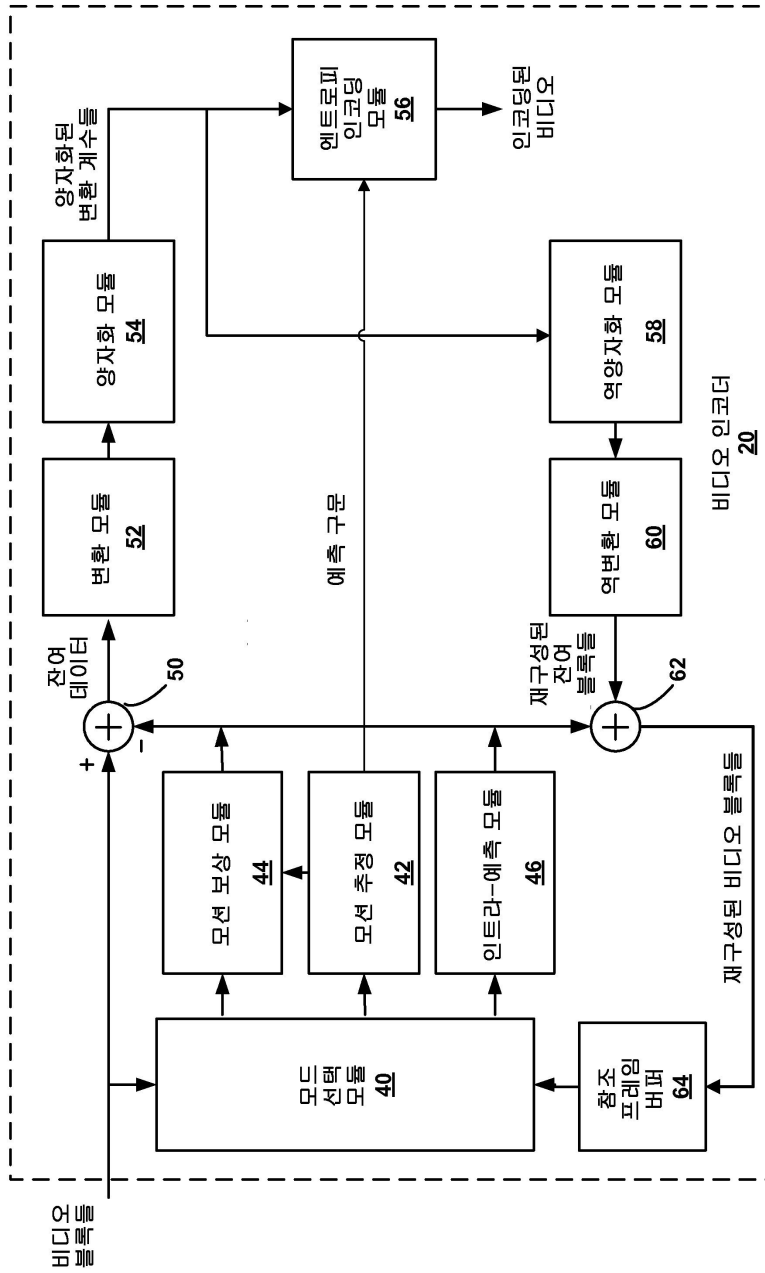
도면2d



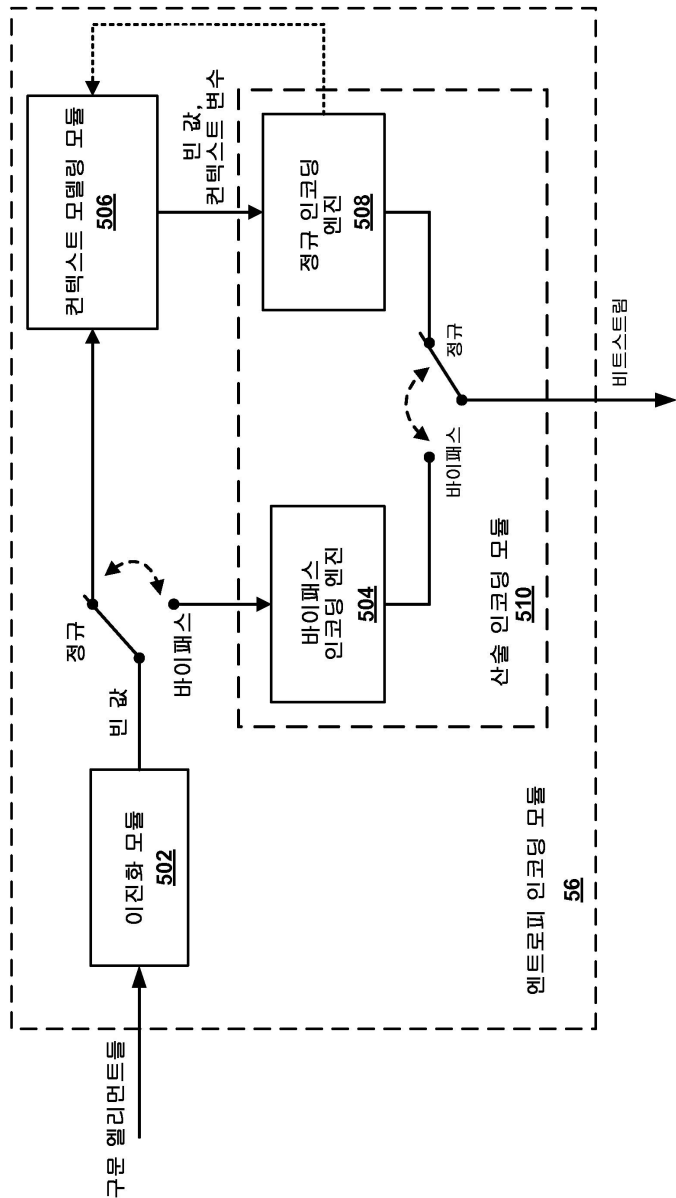
도면3



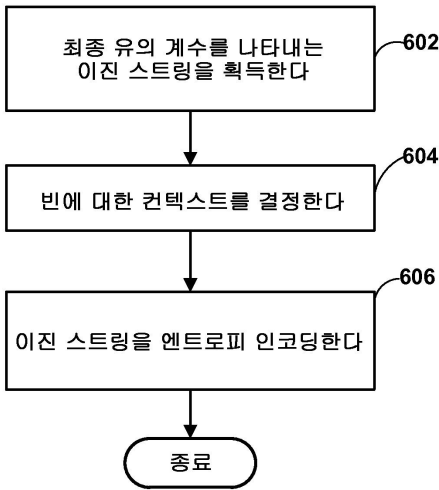
도면4



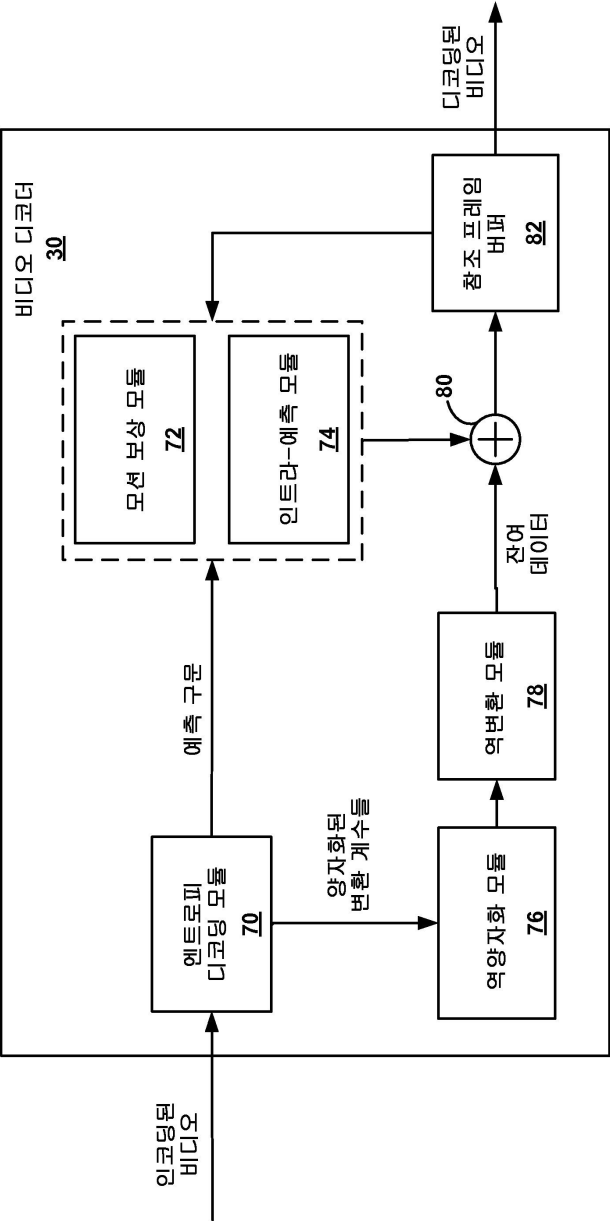
도면5



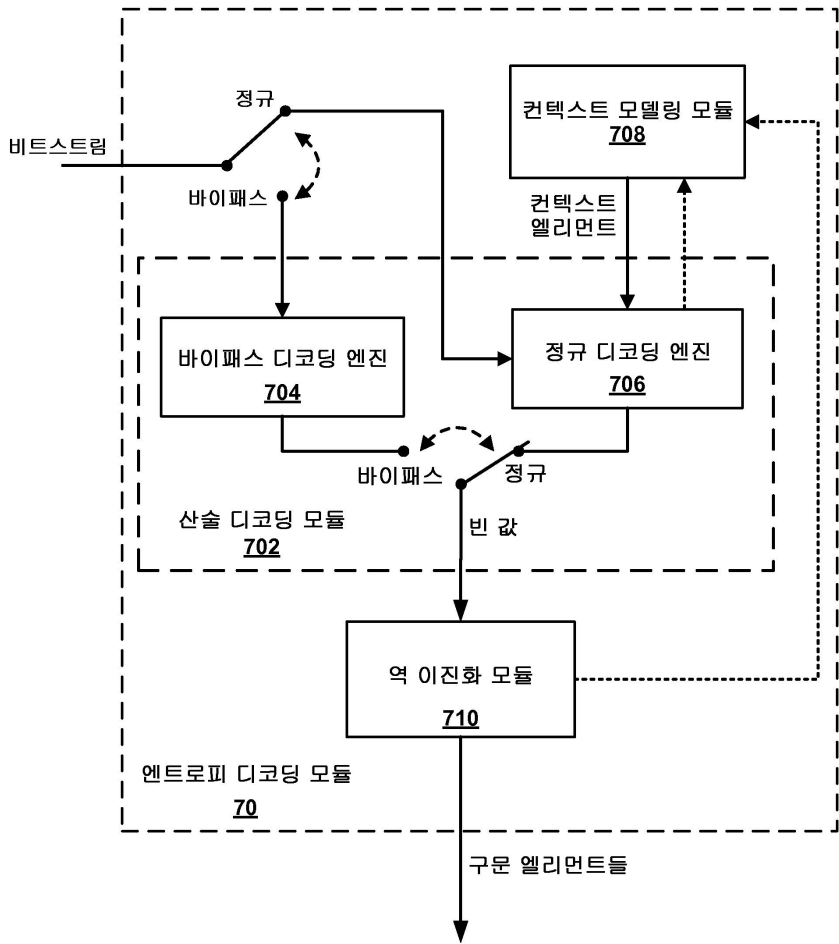
도면6



도면7



도면8



도면9

