



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2014-0098111
 (43) 공개일자 2014년08월07일

(51) 국제특허분류(Int. Cl.)
H04N 19/13 (2014.01) **H04N 19/60** (2014.01)
 (21) 출원번호 10-2014-7014978
 (22) 출원일자(국제) 2012년11월08일
 심사청구일자 2014년06월02일
 (85) 번역문제출일자 2014년06월02일
 (86) 국제출원번호 PCT/US2012/064229
 (87) 국제공개번호 WO 2013/070970
 국제공개일자 2013년05월16일
 (30) 우선권주장
 13/671,811 2012년11월08일 미국(US)
 (뒷면에 계속)

(71) 출원인
모토로라 모빌리티 엘엘씨
 미국 60654 일리노이주 시카고 웨스트 머천다이즈
 마트 플라자 222
 (72) 발명자
로우, 지안
 미국 92122 캘리포니아주 샌 디에고 에이퍼티 넘
 버413 피오레 테르 5240
왕, 쉘
 미국 92127 캘리포니아주 샌 디에고 핀토 릿지 드
 라이브 16354
왕, 리민
 미국 92128 캘리포니아주 샌 디에고 솔 서밋 드라
 이브 13641
 (74) 대리인
백만기, 양영준, 정은진

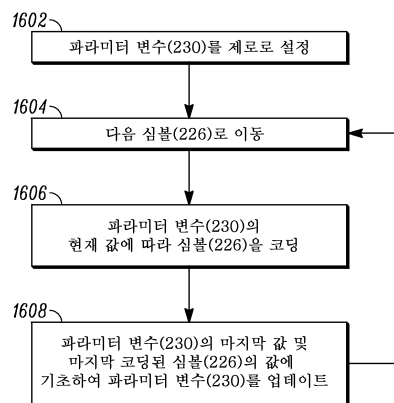
전체 청구항 수 : 총 19 항

(54) 발명의 명칭 **변환 계수들에 대한 이진 코드워드들을 결정하는 방법**

(57) 요약

고효율성 비디오 코딩(HEVC: High Efficiency Video Coding) 시스템에서 코딩 유닛(CU)들을 분할하는 변환 유닛(TU)들을 관련시키기 위해 사용되는 변환 계수들에 대한 레벨 파라미터 업데이트 코드워드들을 생성하기 위한 시스템이 제공된다. 시스템은 코드워드들의 이진화를 제공하고, 불필요한 동작들을 제거하여 시스템 복잡도를 감소시키고 압축 성능을 증가시킨다. 시스템은 TU들을 관련시키는 변환 계수들을 생성하고, 초기 값이 0으로 설정된 파라미터 변수(cRiceParam)를 제공함으로써 시작한다. 파라미터 변수는 이후, 파라미터 변수의 현재 값 및 심볼의 값에 기초한 이진 코드워드로 변환되고, 이후, 각각의 심볼이 변환된 후에 새로운 현재 값으로 업데이트된다. 업데이트는 표 값들에 관련하여 제공될 수 있거나, 또는 값들이 조합 논리로부터 제공될 수 있다.

대표도 - 도16



(30) 우선권주장

61/556,826 2011년11월08일 미국(US)

61/563,774 2011년11월26일 미국(US)

61/564,248 2011년11월28일 미국(US)

특허청구의 범위

청구항 1

변환 계수들에 대한 이진 코드워드들을 결정하는 방법으로서,

상기 변환 계수들의 하나 이상의 서브세트들을 포함하는 변환 유닛을 제공하는 단계 - 상기 변환 계수들 각각은 양자화된 값을 가짐 -;

상기 변환 계수의 양자화된 값으로부터 임계 값을 차감함으로써, 상기 임계 값보다 더 크거나 같은 양자화된 값을 갖는 상기 변환 계수들 각각에 대한 심볼을 결정하는 단계;

초기 값이 0으로 설정된 파라미터 변수를 제공하는 단계;

상기 파라미터 변수의 현재 값 및 상기 심볼의 값에 기초하여 상기 심볼들을 이진 코드워드로 변환하는 단계; 및

각각의 심볼들이 변환된 이후, 상기 파라미터 변수의 값을 상기 심볼들 각각에 대한 새로운 현재 값으로 업데이트하는 단계 - 상기 새로운 현재 값은 상기 파라미터 변수의 마지막 값 및 마지막으로 변환된 심볼의 값에 적어도 부분적으로 기초함

를 포함하는 방법.

청구항 2

제1항에 있어서,

상기 변환하는 단계는 상기 심볼의 값 및 상기 파라미터 변수의 상기 업데이트된 값에 기초하는 표로부터 상기 이진 코드워드를 검색하는 단계를 포함하는, 방법.

청구항 3

제1항에 있어서,

상기 임계 값은 3인, 방법.

청구항 4

제1항에 있어서,

상기 파라미터 변수를 업데이트하는 단계는:

(1) 상기 파라미터 변수의 마지막 값 및 (2) 상기 마지막으로 변환된 심볼의 값에 기초하는 표로부터 상기 새로운 값을 검색하는 단계를 포함하는, 방법.

청구항 5

제1항에 있어서,

상기 파라미터 변수를 업데이트하는 단계는:

상기 파라미터 변수의 마지막 값 및 상기 마지막으로 변환된 심볼의 값이 모두 하나 이상의 조건들을 만족시키는지 결정하는 단계를 포함하는, 방법.

청구항 6

제5항에 있어서,

상기 하나 이상의 조건들 각각은 조건부 심볼 임계값 및 조건부 파라미터 임계값을 포함하는, 방법.

청구항 7

제5항에 있어서,

상기 하나 이상의 조건들 중 하나는 상기 마지막으로 변환된 심볼의 값이 해당 조건에 대한 상기 조건부 심볼 임계값보다 더 크거나 같고 상기 파라미터 변수의 마지막 값이 해당 조건에 대한 상기 조건부 파라미터 임계값보다 더 작거나 같을 때 만족되는, 방법.

청구항 8

제7항에 있어서,

상기 조건부 심볼 임계값은 상기 하나 이상의 조건들 각각에 대해 상이한, 방법.

청구항 9

제1항에 있어서,

상기 파라미터 변수의 값은 0, 1, 2 또는 3이도록 구성되는, 방법.

청구항 10

제1항에 있어서,

상기 파라미터 변수의 값은 0, 1, 2, 또는 2와 지정된 상한값 사이의 임의의 정수가 되도록 구성되는, 방법.

청구항 11

제1항에 있어서,

상기 변환 계수들은 고효율성 비디오 코딩(HEVC: High Efficiency Video Coding) 신호에서 코딩 유닛(CU)의 세 부분할(subdivision)을 제공하는 변환 유닛(TU) 내에 제공되는, 방법.

청구항 12

제1항에 있어서,

상기 변환 계수들은 고효율성 비디오 코딩(HEVC) 신호에서 코딩 유닛(CU)의 세부분할을 제공하는 변환 유닛(TU)의 서브세트 내에 제공되는, 방법.

청구항 13

변환 계수들에 대한 이진 코드워드들을 결정하는 방법으로서,

상기 변환 계수들의 하나 이상의 서브세트들을 포함하는 변환 유닛을 제공하는 단계 - 상기 변환 계수들 각각은 양자화된 값을 가짐 -;

상기 변환 계수의 양자화된 값으로부터 임계 값을 차감함으로써, 상기 임계 값보다 더 크거나 같은 양자화된 값을 갖는 상기 변환 계수들 각각에 대한 심볼을 결정하는 단계;

초기 값이 0으로 설정된 파라미터 변수를 제공하는 단계;

상기 파라미터 변수의 현재 값 및 상기 심볼의 값에 기초하여 상기 심볼들을 이진 코드워드로 변환하는 단계;

상기 파라미터 변수의 마지막 값 및 마지막으로 변환된 심볼의 값에 기초한 표로부터 새로운 현재 값을 검색하는 단계; 및

상기 파라미터 변수의 값을 상기 새로운 현재 값으로 대체하는 단계

를 포함하는 방법.

청구항 14

변환 계수들에 대한 이진 코드워드들을 결정하는 방법으로서,

변환 계수들의 하나 이상의 서브세트들을 포함하는 변환 유닛을 제공하는 단계 - 상기 변환 계수들 각각은 양자화된 값을 가짐 -;

상기 변환 계수의 양자화된 값으로부터 임계 값을 차감함으로써, 상기 임계 값보다 더 크거나 같은 양자화된 값을 갖는 상기 변환 계수들 각각에 대한 심볼을 결정하는 단계;

초기 값이 0으로 설정된 파라미터 변수를 제공하는 단계;

상기 파라미터 변수의 현재 값 및 상기 심볼의 값에 기초하여 상기 심볼들을 이진 코드워드로 변환하는 단계;

상기 파라미터 변수의 마지막 값 및 마지막으로 변환된 심볼의 값이 모두 하나 이상의 조건들을 만족시키는지를 결정하는 단계; 및

만족되는 상기 하나 이상의 조건들 각각에 대한 상기 파라미터 변수의 마지막 값에 정수값 1을 수학적으로 가산하는 단계

를 포함하는 방법.

청구항 15

제14항에 있어서,

상기 하나 이상의 조건들 각각은 조건부 심볼 임계값 및 조건부 파라미터 임계값을 포함하는, 방법.

청구항 16

제14항에 있어서,

상기 하나 이상의 조건들 중 하나는, 상기 마지막으로 변환된 심볼의 값이 해당 조건에 대한 상기 조건부 심볼 임계값보다 더 크거나 같고, 상기 파라미터 변수의 마지막 값이 해당 조건에 대한 상기 조건부 파라미터 임계값보다 더 작거나 같을 때 만족되는, 방법.

청구항 17

제16항에 있어서,

상기 조건부 심볼 임계값은 상기 하나 이상의 조건들 각각에 대해 상이한, 방법.

청구항 18

제14항에 있어서,

상기 파라미터 변수의 값은 0, 1, 2, 또는 3이도록 구성되는, 방법.

청구항 19

제14항에 있어서,

상기 파라미터 변수의 값은 0, 1, 2, 또는 2와 지정된 상한값 사이의 임의의 정수가 되도록 구성되는, 방법.

명세서

기술분야

[0001] 관련 출원들에 대한 교차 참조

[0002] 이 출원은, 35 U.S.C. § 119(e) 규정에 따라, 2011년 11월 8일에 출원된 조기 출원된 미국 가출원 번호 제 61/556,826호; 2011년 11월 26일에 출원된 조기 출원된 미국 가출원 번호 제 61/563,774호; 및 2011년 11월 28일에 출원된 조기 출원된 미국 가출원 번호 제 61/564,248호로부터 우선권을 주장하며, 그 전체 내용이 인용에 의해 본원에 포함된다.

[0003] 본 개시내용은 비디오 압축 분야, 특히, 블록 프로세싱을 사용하는 고효율성 비디오 코딩(HEVC: High Efficiency Video Coding)을 사용하는 비디오 압축에 관한 것이다.

배경기술

[0004] 도 1은 HEVC 데이터를 전송하고 수신하기 위해 사용될 수 있는 코딩 시스템(110) 및 디코딩 시스템(140)을 포함

하는 컨텐츠 분배 시스템(100)을 도시한다. 일부 실시예들에서, 코딩 시스템(110)은 입력 인터페이스(130), 제어기(111), 카운터(112), 프레임 메모리(113), 인코딩 유닛(114), 송신기 버퍼(115) 및 출력 인터페이스(135)를 포함할 수 있다. 디코딩 시스템(140)은 수신기 버퍼(150), 디코딩 유닛(151), 프레임 메모리(152) 및 제어기(153)를 포함할 수 있다. 코딩 시스템(110) 및 디코딩 시스템(140)은 압축된 비트스트림(105)을 전달할 수 있는 전송 경로를 통해 서로 연결될 수 있다. 코딩 시스템(110)의 제어기(111)는 수신기 버퍼(150)의 용량에 기반하여 전송될 데이터의 양을 제어할 수 있고, 시간 단위 당 데이터의 양과 같은 다른 파라미터들을 포함할 수 있다. 제어기(111)는 디코딩 시스템(140)의 수신된 신호 디코딩 동작의 실패의 발생을 방지하도록 인코딩 유닛(114)을 제어할 수 있다. 제어기(111)는 프로세서일 수 있으며, 비-제한적인 예시로서, 프로세서, 랜덤 액세스 메모리 및 판독 전용 메모리를 갖는 마이크로컴퓨터를 포함할 수 있다.

[0005] 비-제한적 예시로서, 컨텐츠 제공자로부터 공급된 소스 픽처들(120)은 비디오 시퀀스 내에 소스 픽처들을 포함하는 프레임들의 비디오 시퀀스를 포함할 수 있다. 소스 픽처들(120)은 압축되지 않거나 압축될 수 있다. 소스 픽처들(120)이 압축되지 않는 경우, 코딩 시스템(110)은 인코딩 기능을 가질 수 있다. 소스 픽처들(120)이 압축되는 경우, 코딩 시스템(110)은 트랜스코딩 기능을 가질 수 있다. 코딩 유닛들은 제어기(111)를 이용하여 소스 픽처들로부터 유도될 수 있다. 프레임 메모리(113)는 소스 픽처들(120)로부터 인입 프레임들을 저장하기 위해 사용될 수 있는 제1 영역, 및 프레임들을 판독하고 이들을 인코딩 유닛들(114)에 출력하기 위해 사용될 수 있는 제2 영역을 가질 수 있다. 제어기(111)는 프레임 메모리(113)에 영역 스위칭 제어 신호(123)를 출력할 수 있다. 영역 스위칭 제어 신호(123)는 제1 영역이 이용될 것인지 또는 제2 영역이 이용될 것인지를 표시할 수 있다.

[0006] 제어기(111)는 인코딩 유닛(114)에 인코딩 제어 신호(124)를 출력할 수 있다. 인코딩 제어 신호(124)는 인코딩 유닛(114)으로 하여금, 소스 픽처에 기초하여 코딩 유닛들을 준비하는 것과 같은, 인코딩 동작을 시작하게 할 수 있다. 제어기(111)로부터의 인코딩 제어 신호(124)에 응답하여, 인코딩 유닛(114)은, 코딩 유닛들과 연관된 소스 픽처들에 기초하여 비디오 압축 데이터를 생성하는 준비된 코딩 유닛들을 프로세싱하는 예측 코딩 프로세스 또는 변환 코딩 프로세스와 같은, 고효율성 인코딩 프로세스에 대해 준비된 코딩 유닛들을 판독하기 시작할 수 있다.

[0007] 인코딩 유닛(114)은 비디오 패킷들을 포함하는 패킷화된 기본 스트림(PES) 내에 생성된 비디오 압축 데이터를 패키징할 수 있다. 인코딩 유닛(114)은 제어 정보 및 프로그램 시간 스탬프(PTS)를 사용하여 비디오 패킷들을 인코딩된 비디오 신호(122) 내로 매핑할 수 있고, 인코딩된 비디오 신호(122)는 송신기 버퍼(115)에 전송될 수 있다.

[0008] 생성된 비디오 압축 데이터를 포함하는 인코딩된 비디오 신호(122)는 송신기 버퍼(115)에 저장될 수 있다. 정보량 카운터(112)는 송신기 버퍼(115) 내의 전체 데이터량을 표시하도록 증분될 수 있다. 데이터가 검색되어 버퍼로부터 제거됨에 따라, 카운터(112)는 송신기 버퍼(115) 내의 데이터량을 반영하기 위해 감소할 수 있다. 점유된 영역 정보 신호(126)는 인코딩 유닛(114)으로부터의 데이터가 송신기 버퍼(115)에 추가되었는지 또는 그로부터 제거되었는지를 표시하기 위해 카운터(112)에 전송될 수 있고, 따라서, 카운터(112)는 증분되거나 감소할 수 있다. 제어기(111)는, 송신기 버퍼(115)에서 오버 플로우 또는 언더 플로우가 발생하는 것을 예견하고, 회피하고, 방지하고 그리고/또는 검출하기 위해 전달될 수 있는 점유된 영역 정보(126)에 기반하여 인코딩 유닛(114)에 의해 생성된 비디오 패킷들의 생성을 제어할 수 있다.

[0009] 정보량 카운터(112)는 제어기(111)에 의해 생성되고 출력된 사전설정(preset) 신호(128)에 응답하여 리셋될 수 있다. 정보 카운터(112)가 리셋된 이후, 정보 카운터(112)는 인코딩 유닛(114)에 의해 출력된 데이터를 카운팅하고, 생성된 비디오 압축 데이터 및/또는 비디오 패킷들의 양을 획득할 수 있다. 정보량 카운터(112)는 제어기(111)에 획득된 정보량을 나타내는 정보량 신호(129)를 공급할 수 있다. 제어기(111)는 송신기 버퍼(115)에 오버플로우가 존재하지 않도록 인코딩 유닛(114)을 제어할 수 있다.

[0010] 일부 실시예들에서, 디코딩 시스템(140)은 입력 인터페이스(170), 수신기 버퍼(150), 제어기(153), 프레임 메모리(152), 디코딩 유닛(151) 및 출력 인터페이스(175)를 포함할 수 있다. 디코딩 시스템(140)의 수신기 버퍼(150)는 소스 픽처들(120)로부터의 소스 픽처들에 기초하여 수신된 비디오 압축 데이터 및 비디오 패킷들을 포함하는, 압축된 비트스트림(105)을 일시적으로 저장할 수 있다. 디코딩 시스템(140)은 수신된 데이터 내의 비디오 패킷들과 연관된 제어 정보 및 표시 타임스탬프 정보를 판독하고, 제어기(153)에 적용될 수 있는 프레임 수 신호(163)를 출력할 수 있다. 제어기(153)는 미리 결정된 간격으로 카운팅된 프레임들의 수를 감독할 수 있다. 비-제한적인 예시에 의해, 제어기(153)는, 디코딩 유닛(151)이 디코딩 동작을 완료할 때마다, 카운팅된 프

레이들의 수를 감독할 수 있다.

- [0011] 일부 실시예들에서, 프레임 수 신호(163)가, 수신기 버퍼(150)가 미리 결정된 용량에 있음을 표시할 때, 제어기(153)는 디코딩 시작 신호(164)를 디코딩 유닛(151)에 출력할 수 있다. 프레임 수 신호(163)가 수신된 버퍼(150)가 미리 결정된 용량보다 더 적음을 표시할 때, 제어기(153)는 카운트된 프레임들의 수가 미리 결정된 양과 동일해지는 상황의 발생을 대기할 수 있다. 제어기(153)는 그 상황이 발생할 때 디코딩 시작 신호(164)를 출력할 수 있다. 비-제한적인 예시에 의해, 제어기(153)는 프레임 수 신호(163)가 수신기 버퍼(150)가 미리 결정된 용량에 있음을 표시할 때 디코딩 시작 신호(164)를 출력할 수 있다. 인코딩된 비디오 패킷들 및 비디오 압축 데이터는 인코딩된 비디오 패킷들과 연관된 표시 타임스탬프들에 기초하여 단조 순서로(즉, 증가하거나 감소하는) 디코딩될 수 있다.
- [0012] 디코딩 시작 신호(164)에 응답하여, 디코딩 유닛(151)은 프레임과 연관된 하나의 픽처에 해당하는 데이터, 및 수신기 버퍼(150)로부터의 비디오 패킷들과 연관된 픽처와 연관된 압축된 비디오 데이터를 디코딩할 수 있다. 디코딩 유닛(151)은 디코딩된 비디오 신호(162)를 프레임 메모리(152) 내에 기록할 수 있다. 프레임 메모리(152)는 디코딩된 비디오 신호가 기록되는 제1 영역, 및 출력 인터페이스(175)로 디코딩된 픽처(160)를 판독해 내기 위해 사용되는 제2 영역을 가질 수 있다.
- [0013] 다양한 실시예들에서, 코딩 시스템(110)은 헤드엔드에서 트랜스코더 또는 인코딩 장치에 포함되거나 또는 다른 방식으로 연관될 수 있고, 디코딩 시스템(140)은 모바일 디바이스, 셋톱 박스 또는 트랜스코더와 같은 다운스트림 디바이스에 포함되거나 또는 다른 방식으로 연관될 수 있다.
- [0014] 코딩 시스템(110) 및 디코딩 시스템(140)은, 고효율성 비디오 코딩(HEVC)을 포함하는, 다양한 코딩 포맷들에 따라 비디오 데이터를 인코딩하고 디코딩하기 위해 별도로 또는 함께 이용될 수 있다. HEVC는 블록 기반 하이브리드 공간적 및 시간적 예측 코딩 방식이다. HEVC에서, 비디오 프레임들과 같은 입력 이미지들은, 도 2에 도시된 바와 같이, 가장 큰 코딩 유닛(LCU)들(200)로서 명명되는 사각형 블록들로 분할될 수 있다. 입력 이미지들을 16x16 픽셀들의 매크로블록들로 나누는 다른 코딩 방식들과는 달리, LCU들(200)은 각각 128x128 픽셀들만큼 클 수 있다. 도 3에 도시된 바와 같이, 각각의 LCU(200)는 LCU(200)를 4개의 코딩 유닛(CU)들(202)로 분할함으로써 파티셔닝될 수 있다. CU들(202)은 각각이 LCU(200)의 1/4 사이즈인 사각형 블록들일 수 있다. 각각의 CU(202)는 각각이 더 큰 CU(202)의 1/4 사이즈인 4개의 더 작은 CU들(202)로 추가로 분할될 수 있다. 비-제한적인 예시에 의해, 도 3에 도시된 LCU(200)의 상단 우측 코너의 CU(202)는 4개의 더 작은 CU들(202)로 분할될 수 있다. 일부 실시예들에서, 이들 더 작은 CU들(202)은 훨씬 더 작은 크기의 쿼터(quarter)들로 추가로 분할될 수 있고, CU들(202)을 더 작은 CU들(202)로 분할하는 이 프로세스는 여러 번 완료될 수 있다.
- [0015] 비디오 데이터 밀도가 점점 더 높아짐에 따라, 큰 입력 이미지들 및/또는 매크로블록들이 신속하게, 효율적으로 그리고 정확하게 인코딩되고 디코딩되도록, CU들을 코딩하기 위한 더욱 개선된 방법들이 요구된다.

발명의 내용

- [0016] 본 발명은 HEVC에 대한 개선된 시스템을 제공한다. 시스템에 대한 실시예들에서, 효율적 방식으로 변환 계수들에 대한 이진 코드워드들을 결정하는 방법이 제공된다. CU들(202)의 세부분할들인 변환 유닛(TU)들 내의 변환 계수들에 대한 코드워드들은 입력 이미지들 및/또는 매크로블록들을 인코딩할 시에 사용된다.
- [0017] 일 실시예에서, 변환 계수들의 하나 이상의 서브세트들을 포함하는 변환 유닛을 제공하는 단계 - 각각의 변환 계수는 양자화된 값을 가짐 - , 변환 계수의 양자화된 값으로부터 임계값을 차감함으로써 임계값보다 더 크거나 같은 양자화된 값을 갖는 각각의 변환 계수에 대한 심볼을 결정하는 단계, 제로의 초기값으로 설정된 파라미터 변수를 제공하는 단계, 파라미터 변수의 현재 값 및 심볼의 값에 기초하여 각각의 심볼을 이진 코드워드로 변환하는 단계, 및 각각의 심볼이 변환된 이후 파라미터 변수의 값을 새로운 현재 값으로 업데이트하는 단계 - 새로운 현재 값은 파라미터 변수의 마지막 값 및 현재 또는 이전 서브세트 내의 마지막으로 변환된 심볼의 값에 적어도 부분적으로 기초함 - 를 포함하는 방법이 제공된다.
- [0018] 또 다른 실시예에서, 발명은 변환 계수들을 결정하기 위해 룩업 테이블을 사용하는 변환 계수들에 대한 이진 코드워드들을 결정하는 방법을 포함한다. 방법은 변환 계수들의 하나 이상의 서브세트들을 포함하는 변환 유닛을 제공하는 단계 - 각각의 변환 계수는 양자화된 값을 가짐 - , 변환 계수의 양자화된 값으로부터 임계 값을 차감함으로써, 임계 값보다 더 크거나 같은 양자화된 값을 갖는 각각의 변환 계수에 대한 심볼을 결정하는 단계, 초기 값이 0으로 설정된 파라미터 변수를 제공하는 단계, 파라미터 변수의 현재 값 및 심볼의 값에 기초하여 각각의 심볼을 이진 코드워드로 변환하는 단계, 파라미터 변수의 마지막 값 및 마지막으로 변환된 심볼의 값에 기초

하여 표로부터 새로운 현재 값을 검색하는 단계, 및 파라미터 변수의 값을 새로운 현재 값으로 교체하는 단계를 포함한다.

[0019] 또 다른 실시예에서, 발명은 룩업 테이블을 요구하는 것보다는, 로직을 사용하여 수행될 수 있는 하나 이상의 수학적 조건들을 사용하는 변환 계수들에 대한 이진 코드워드들을 결정하는 방법을 포함한다. 방법은 변환 계수들의 하나 이상의 서브세트들을 포함하는 변환 유닛을 제공하는 단계 - 각각의 변환 계수는 양자화된 값을 가짐 - , 변환 계수의 양자화된 값으로부터 임계 값을 차감함으로써, 임계값보다 더 크거나 같은 양자화된 값을 갖는 각각의 변환 계수에 대한 심볼을 결정하는 단계, 초기 값이 0으로 설정된 파라미터 변수를 제공하는 단계, 파라미터 변수의 현재 값 및 심볼의 값에 기초하여 각각의 심볼을 이진 코드워드로 변환하는 단계, 파라미터 변수의 마지막 값 및 마지막으로 변환된 심볼의 값이 함께 하나 이상의 조건들을 만족시키는지의 여부를 결정하는 단계, 및 만족되는 하나 이상의 조건들 각각에 대한 파라미터 변수의 마지막 값에 1의 정수를 수학적으로 가산하는 단계를 포함한다.

도면의 간단한 설명

[0020] 본 발명의 추가적인 상세항목들이 첨부 도면들의 보조로 설명된다.

- 도 1은 콘텐츠 분배 시스템의 실시예를 도시하는 도면이다.
- 도 2는 큰 코딩 유닛들로 분할된 입력 이미지의 실시예를 도시하는 도면이다.
- 도 3은 코딩 유닛들로 분할된 큰 코딩 유닛의 실시예를 도시하는 도면이다.
- 도 4는 코딩 유닛들로 분할된 큰 코딩 유닛의 쿼드트리 표현을 도시하는 도면이다.
- 도 5는 코딩 유닛 내의 예측 유닛들의 가능한 예시적인 배열들을 도시하는 도면이다.
- 도 6은 예측 유닛을 인코딩 및/또는 디코딩하기 위한 방법의 실시예의 블록도를 도시하는 도면이다.
- 도 7은 예측 유닛들 및 변환 유닛들로 분할된 코딩 유닛의 예시적인 실시예를 도시하는 도면이다.
- 도 8은 변환 유닛들로 분할된 코딩 유닛들의 쿼드트리 표현의 예시적인 실시예를 도시하는 도면이다.
- 도 9는 상황-기반 적응형 이진 산술 코딩을 수행하는 방법의 실시예를 도시하는 도면이다.
- 도 10은 중요도 맵의 예시적인 실시예를 도시하는 도면이다.
- 도 11은 변환 유닛 내의 변환 계수들 및 변환 계수들의 서브세트들의 역 지그재그 스캔의 실시예를 도시하는 도면이다.
- 도 12는 변환 계수들에 대한 계수 레벨들 및 심볼들을 획득하는 방법의 실시예를 도시하는 도면이다.
- 도 13은 서브세트들 내의 변환 계수의 스캐닝 순서의 실시예를 도시하는 도면이다.
- 도 14는 연관된 파라미터 변수들에 대한 최대 심볼 값들의 예시적인 실시예들을 도시하는 도면이다.
- 도 15는 파라미터 변수들에 기초하여 심볼들을 이진 코드워드들로 변환하기 위한 표의 예시적인 실시예를 도시하는 도면이다.
- 도 16은 심볼들을 코딩하고 파라미터 변수들을 업데이트하기 위한 방법의 실시예를 도시하는 도면이다.
- 도 17은 2, 4, 13, 11, 및 10의 조건부 심볼 임계들을 갖는 낮은 복잡도의 업데이트 표의 예시적인 실시예를 도시하는 도면이다.
- 도 18은 3, 6, 및 12의 조건부 심볼 임계들을 갖는 낮은 복잡도의 업데이트 표의 예시적인 실시예를 도시하는 도면이다.
- 도 19는 2, 5, 및 11의 조건부 심볼 임계들을 갖는 낮은 복잡도의 업데이트 표의 예시적인 실시예를 도시하는 도면이다.
- 도 20은 2, 4, 13, 11, 및 10의 조건부 심볼 임계들에 대한 조건들의 조합 논리 표현의 예시적인 실시예를 도시하는 도면이다.
- 도 21은 3, 6, 및 12의 조건부 심볼 임계들에 대한 조건들의 조합 논리 표현의 예시적인 실시예를 도시하는 도

면이다.

도 22는 2, 5, 및 11의 조건부 심볼 임계들에 기초하여 파라미터 변수를 업데이트하기 위해 사용될 수 있는 예시적인 코드를 도시하는 도면이다.

도 23은 A, B, 및 C의 조건부 심볼 임계들을 갖는 낮은 복잡도의 업데이트 표의 예시적인 실시예를 도시하는 도면이다.

도 24는 A, B, 및 C의 조건부 심볼 임계들에 대한 조건들의 조합 논리 표현의 예시적인 실시예를 도시하는 도면이다.

도 25는 2, 4, 및 12의 조건부 심볼 임계들을 갖는 낮은 복잡도의 업데이트 표의 예시적인 실시예를 도시하는 도면이다.

도 26은 2, 4, 및 12의 조건부 심볼 임계들에 대한 조건들의 조합 논리 표현의 예시적인 실시예를 도시하는 도면이다.

도 27은 2, 4, 및 13의 조건부 심볼 임계들을 갖는 낮은 복잡도의 업데이트 표의 예시적인 실시예를 도시하는 도면이다.

도 28은 2, 4, 및 13의 조건부 심볼 임계들에 대한 조건들의 조합 논리 표현의 예시적인 실시예를 도시하는 도면이다.

도 29는 2, 4, 및 11의 조건부 심볼 임계들을 갖는 낮은 복잡도의 업데이트 표의 예시적인 실시예를 도시하는 도면이다.

도 30은 2, 4, 및 11의 조건부 심볼 임계들에 대한 조건들의 조합 논리 표현의 예시적인 실시예를 도시하는 도면이다.

도 31은 2, 4, 및 10의 조건부 심볼 임계들을 갖는 낮은 복잡도의 업데이트 표의 예시적인 실시예를 도시하는 도면이다.

도 32는 2, 4, 및 10의 조건부 심볼 임계들에 대한 조건들의 조합 논리 표현의 예시적인 실시예를 도시하는 도면이다.

도 33은 컴퓨터 하드웨어의 예시적인 실시예를 도시하는 도면이다.

발명을 실시하기 위한 구체적인 내용

[0021] HEVC에서, 비디오 프레임과 같은 입력 프레임은, 이후 코드에서 식별되는 CU들로 나누어진다. CU들은 이후 후속적으로 기술될 바와 같이 코딩되는 서브-유닛들로 추가로 나누어진다.

[0022] 초기에, 코딩을 위해, 쿼드트리 데이터 표현이 LCU(200)의 파티션을 기술하기 위해 사용될 수 있다. 쿼드트리 표현은 LCU(200) 및 CU들(202)에 대응하는 노드들을 가질 수 있다. 쿼드트리 표현의 각각의 노드에서, LCU(200) 또는 CU(202)가 4개의 CU들(202)로 분할되는 경우 플래그 "1"가 할당될 수 있다. 노드가 CU들(202)로 분할되지 않는 경우, 플래그 "0"가 할당될 수 있다. 비-제한적인 예시에 의해, 도 4에 도시된 쿼드트리 표현은 도 3에 도시된 LCU 파티션을 기술할 수 있고, LCU(200)는 4개의 CU들(202)로 분할되고, 제2 CU(202)는 4개의 더 작은 CU들(202)로 분할된다. 쿼드트리의 이진 데이터 표현은, 후속적으로 기술된 스킵 모드 플래그, 병합 모드 플래그, 및 PU 코딩 모드와 같은 다른 데이터와 함께, 오버헤드로서 코딩되고 전송될 수 있는 CU 분할 플래그일 수 있다. 비-제한적인 예시에 의해, 도 4에 도시된 CU 분할 플래그 쿼드트리 표현이 이진 데이터 표현 "10100"으로서 코딩될 수 있다.

[0023] 쿼드트리의 각각의 리프에서, 최종 CU들(202)은 예측 유닛(PU)들(204)로 명명되는 하나 이상의 블록들로 분할될 수 있다. PU들(204)은 정사각형 또는 직사각형일 수 있다. 2Nx2N의 디멘전을 갖는 CU(202)는 도 5에 도시된 PU들(204)의 4개의 예시적인 배열들 중 하나를 가질 수 있고, PU들(204)은 2Nx2N, 2NxN, Nx2N, 또는 NxN의 디멘전을 가진다.

[0024] PU는 공간적 또는 시간적 예측을 통해 획득될 수 있다. 시간적 예측은 인터 모드 픽처들에 관련된다. 공간적 예측은 인트라 모드 픽처에 관한 것이다. 따라서, 각각의 CU(202)의 PU들(204)은 인트라 모드 또는 인터 모드 중 어느 하나에서 코딩될 수 있다. 인트라 모드 및 인터 모드 픽처들에 관한 코딩의 특징들은 후속하는 문단들

에 기술된다.

- [0025] 인트라 모드 코딩은 I 픽처를 코딩하기 위해, 다른 이미지들을 참조하지 않고, 현재 입력 이미지로부터의 데이터를 사용할 수 있다. 인트라 모드에서, PU들(204)은 공간적으로 예측 코딩될 수 있다. CU(202)의 각각의 PU(204)는 자신의 고유한 공간적 예측 방향을 가질 수 있다. 공간적 예측 방향들은 수평, 수직, 45도 대각, 135도 대각, DC, 평면, 또는 임의의 다른 방향일 수 있다. PU(204)에 대한 공간적 예측 방향은 선택스 엘리먼트로서 코딩될 수 있다. 일부 실시예들에서, PU(204)에 대한 밝기 정보(Luma) 및 컬러 정보(Chroma)는 별도로 예측될 수 있다. 일부 실시예들에서, 4x4, 8x8, 16x16, 32x32, 및 64x64 블록들에 대한 Luma 인트라 예측 모드들의 수는 각각 18, 35, 35, 35, 및 4일 수 있다. 대안적인 실시예들에서, 임의의 사이즈의 블록들에 대한 Luma 인트라 예측 모드의 수는 35일 수 있다. 추가적인 모드가 Chroma 인트라 예측 모드에 대해 사용될 수 있다. 일부 실시예들에서, Chroma 예측 모드는 "IntraFromLuma" 로 명명될 수 있다.
- [0026] 인터 모드 코딩은 "P" 픽처들 및/또는 "B" 픽처들을 코딩하기 위해 현재 입력 이미지 및 하나 이상의 참조 이미지들로부터의 데이터를 사용할 수 있다. 일부 상황들 및/또는 실시예들에서, 인터 모드 코딩은 인트라 모드 코딩보다 더 높은 압축을 초래할 수 있다. 인터 모드에서, PU들(204)은 시간적으로 예측 코딩될 수 있고, 따라서 CU(202)의 각각의 PU(204)는 하나 이상의 모션 벡터들 및 하나 이상의 연관된 참조 이미지들을 가질 수 있다. 시간 예측은 연관된 참조 이미지들에 대해 PU(204)에 대한 최상의 매치 예측을 탐색하는 모션 추정 동작을 통해 수행될 수 있다. 최상의 매치 예측은 모션 벡터들 및 연관된 참조 이미지들에 의해 기술될 수 있다. P 픽처들은 현재 입력 이미지 및 하나 이상의 이전 참조 이미지들로부터의 데이터를 사용한다. B 픽처들은 현재 입력 이미지 및 이전 및 후속적인 참조 이미지들 모두로부터의 데이터를 사용하며, 2개까지의 모션 벡터들을 가질 수 있다. 모션 벡터들 및 참조 픽처들은 HEVC 비트스트림에서 코딩될 수 있다. 일부 실시예들에서, 모션 벡터들은 선택스 엘리먼트들 "MV"로서 코딩될 수 있고, 참조 픽처들은 선택스 엘리먼트들 "refIdx"로서 코딩될 수 있다. 일부 실시예들에서, 인터 모드 코딩은 공간적 및 시간적 예측 코딩 모두를 허용할 수 있다.
- [0027] 도 6은 PU(204) x가 인코딩되고 그리고/또는 디코딩될 수 있는 방법의 블록도를 도시한다. 606에서, 602에서의 인트라 모드 또는 604에서의 인터 모드에 의해 예측된 PU(206) x'는, 전송된 바와 같이, 잔여 PU(208) e를 획득하기 위해, 현재 PU(204) x로부터 차감될 수 있다. 608에서, 잔여 PU(208) e는 블록 변환을 이용하여 하나 이상의 변환 유닛(TU)들(210) E로 변환될 수 있다. 각각의 TU(210)는 하나 이상의 변환 계수들(212)을 포함할 수 있다. 일부 실시예들에서, 블록 변환은 정사각형일 수 있다. 대안적인 실시예들에서 블록 변환은 정사각형이 아닐 수 있다.
- [0028] 도 7에 도시된 바와 같이, HEVC에서, 상이한 사이즈들의 블록 변환들의 세트가 CU(202)에 대해 수행될 수 있고, 따라서, 일부 PU들(204)은 더 작은 TU들(210)로 분할될 수 있고, 다른 PU들(204)은 PU(204)와 동일한 사이즈인 TU들(210)을 가질 수 있다. CU들(202) 및 PU들(204)의 TU들(210)로의 분할은 쿼드트리 표현에 의해 도시될 수 있다. 비-제한적인 예시에 의해, 도 8에 도시된 쿼드트리 표현은 도 7에 도시된 CU(202) 내의 TU들(210)의 배열을 도시한다.
- [0029] 도 6을 다시 참조하면, 610에서, TU(210)의 변환 계수들(212), E은 유한 개수의 가능한 값들 중 하나로 양자화될 수 있다. 일부 실시예들에서, 이는 양자화에 의해 유실된 데이터가 복원가능하지 않을 수 있는 손실성 동작이다. 변환 계수들(212)이 양자화된 이후, 612에서, 양자화된 변환 계수들(212)은, 최종 압축 비트들(214)을 획득하기 위해, 하기에 논의되는 바와 같이, 엔트로피 코딩될 수 있다.
- [0030] 614에서, 양자화된 변환 계수들(212)은 역양자화된 변환 계수들(216) E'로 역양자화될 수 있다. 616에서, 역양자화된 변환 계수들(216) E'는 이후 잔여 PU(218) e'를 재구성하도록 역변환될 수 있다. 618에서, 재구성된 잔여 PU(218) e'는 이후, 602에서의 공간 예측 또는 604에서의 시간 예측을 통해 획득되는 대응하는 예측 PU(206) x'에 추가되어, 재구성된 PU(220) x"를 획득할 수 있다. 620에서, 디블로킹 필터가 재구성된 PU들(220) x"에 대해 사용되어 블록 결함들을 감소시킬 수 있다. 620에서, 재구성된 픽셀들 및 원래 픽셀들 사이의 픽셀 오프셋을 보상하도록 조건부로 수행될 수 있는 샘플 적응형 오프셋 프로세스가 또한 제공된다. 또한, 620에서, 적응형 루프 필터가 재구성된 PU들(220) x"에 대해 조건부로 사용되어, 입력 및 출력 이미지들 사이의 코딩 왜곡을 감소시키거나 최소화할 수 있다.
- [0031] 재구성된 이미지가 인터 모드 코딩에서 추후 시간적 예측을 위해 사용될 참조 이미지인 경우, 재구성된 이미지들은 참조 버퍼(622)에 저장될 수 있다. 인트라 모드 코딩된 이미지들은, 추가적인 재구성된 이미지들을 요구하지 않고 디코딩이 시작될 수 있는 가능한 포인트일 수 있다.

- [0032] HEVC는 상황-기반 적응형 이진 산술 코딩(CABAC)과 같은 단계(612) 동안 엔트로피 코딩 방식을 사용할 수 있다. CABAC에 대한 코딩 프로세스는 도 9에 도시되어 있다. 902에서, 변환 유닛들(210)의 마지막 중요 변환 계수의 포지션이 코딩될 수 있다. 도 6을 다시 참조하면, 양자화된 변환 계수들이 TU들(210)을 양자화함으로써 생성된다. 변환 계수들(212)은 중요하거나 중요하지 않을 수 있다. 도 10은 변환 계수들(212)의 중요도 맵(1002)을 도시한다. 중요하지 않은 변환 계수들(212)은 제로의 양자화된 값을 가질 수 있는 반면, 중요한 변환 계수들(212)은 1 이상의 양자화된 값을 가질 수 있다. 일부 실시예들에서, 중요한 변환 계수들(212)은 또한 난-제로의 양자화된 변환 계수들(212)로서 공지될 수 있다. TU(210)가 하나 이상의 중요한 변환 계수들(212)을 포함하는 경우, 도 10에 도시된 바와 같이, TU(210)의 상단 좌측 코너로부터 TU(210)의 하단 우측 코너로의 순방향 지그-재그 코딩 스캔을 따르는 마지막 중요한 변환 계수(212)의 좌표들이 코딩될 수 있다. 대안적인 실시예들에서, 중요한 변환 계수들(212)은 역 과면 스캔, 역 수평 스캔, 역 수직 스캔, 또는 임의의 다른 스캔 순서를 따라 스캐닝될 수 있다. 일부 실시예들에서, 이들 좌표들은 선택스 엘리먼트들 "last_significant_coeff_y" 및 "last_significant_coeff_x"로서 코딩될 수 있다. 비-제한적인 예시에 의해, 도 10은 도 9의 블록(902)에서 코딩되고 있는 TU(210) 내의 마지막 중요한 변환(212b)의 위치를 도시한다.
- [0033] 도 9의 블록(904)에서, 중요도 맵(1002)은 TU(210)에서 중요한 변환 계수들(212)의 각각의 위치들을 표시하도록 코딩될 수 있다. 중요도 맵(1002)은 TU(210) 내의 각각의 위치에 대한 이진 엘리먼트를 포함할 수 있다. 이진 엘리먼트는 그 위치에서의 변환 계수(212)가 중요하지 않음을 표시하기 위해 "0"으로서 코딩될 수 있다. 이진 엘리먼트는 그 위치에서의 변환 계수(212)가 중요함을 표시하기 위해 "1"로서 코딩될 수 있다.
- [0034] 도 11은 TU들(210)의 양자화된 변환 계수들(212)이 그룹들로 분할될 수 있는 방법을 예시한다. 일부 실시예들에서, 그룹들은 서브-블록들일 수 있다. 서브-블록들은 16개의 양자화된 변환 계수들(212)의 정사각형 블록들일 수 있다. 다른 실시예들에서, 그룹들은 서브세트들(1102)일 수 있다. 도 11에 도시된 바와 같이, 서브세트들(1102)은 역방향 지그재그 스캔의 스캔 순서에 따라 연속적인 16개의 양자화된 변환 계수들(212)을 포함할 수 있다. 제1 서브세트는, 마지막 중요한 변환 계수(212b)가 서브세트 내의 어디에 있는지와는 무관하게, 마지막 중요한 변환 계수(212b)를 포함하는 서브세트(1102)일 수 있다. 비-제한적인 예시에 의해, 마지막 중요한 변환 계수(212b)는 2개의 중요하지 않은 변환 계수들에 선행하여, 서브세트 내의 제14 변환 계수(212)일 수 있다.
- [0035] 일부 상황들 및/또는 실시예들에서, 마지막 중요한 변환 계수(212b)를 포함하는 그룹 이전에 역방향 스캔 순서에 따라 중요한 변환 계수를 포함하지 않는 16개의 양자화된 변환 계수들(212)의 하나 이상의 그룹들이 존재할 수 있다. 이들 상황들 및/또는 실시예들에서, 제1 서브세트는 마지막 중요한 변환 계수(212b)를 포함하는 서브세트(1102), 및 제1 서브세트(1102)가 서브세트(1102)의 일부로서 고려되기 이전의 임의의 그룹들일 수 있다. 비-제한적인 예시에 의해, 도 11에서, 제1 서브세트(1102) "서브세트 0"는 역방향 지그-재그 스캔 순서에 따른 16개의 변환 계수들(212)의 제2 그룹인 반면, TU(210)의 하위 우측 코너에서의 16개의 변환 계수들(212)의 그룹은, 그 변환 계수들(212) 중 어느 것도 중요하지 않으므로, 서브세트(1102)의 일부가 아니다. 일부 실시예들에서, 제1 서브세트(1102)는 "서브세트 0"로서 표기될 수 있고, 추가적인 서브세트들(1102)은 "서브세트 1" "서브세트 2", 최대 "서브세트 N"로서 표기될 수 있다. 마지막 서브세트(1102)는 TU(210)의 상단 좌측 코너에서의 위치 0, 0에서 DC 변환 계수(212)를 갖는 서브세트(1102)일 수 있다.
- [0036] 도 9를 다시 참조하면, 마지막 906에서, 각각의 양자화된 변환 계수(212)는 중요한 계수 레벨들에 대한 코딩을 포함하여, 도 6에 도시된 최종 압축 비트(214)를 획득하기 위해 이진 값들로 코딩될 수 있다. 코딩 동안 각각의 양자화된 변환 계수(212)의 절댓값은 양자화된 변환 계수(212)의 부호와는 별도로 코딩될 수 있다. 도 12는 양자화된 변환 계수들의 절댓값을 취하여 처리되는 코딩 단계들을 예시한다. 도 12에 도시된 바와 같이, 1202에서, 각각의 양자화된 변환 계수(212)의 절댓값은 블록(1204)에서 해당 양자화된 변환 계수(212)에 대한 계수 레벨(222)의 획득을 가능하게 하기 위해 취해질 수 있다.
- [0037] 더 높은 빈도로 발생할 것으로 예상되는 블록(1204)에서 획득된 계수 레벨들(222)은 더 낮은 빈도로 발생할 것으로 예상되는 계수 레벨들(222)에 앞서 코딩될 수 있다. 비-제한적인 예시에 의해, 일부 실시예들에서, 0, 1, 또는 2의 계수 레벨들(222)은 가장 자주 발생할 것으로 예상될 수 있다. 3개 부분들에서 계수 레벨들(222)을 코딩하는 것은 가장 자주 발생하는 계수 레벨들(222)을 식별하여, 가장 덜 빈번하게 발생할 것으로 예상될 수 있는 계수 레벨들(222)에 대한 더 복잡한 계산들을 남겨둘 수 있다. 일부 실시예들에서, 이는 계수 레벨들(222)을 3개 부분들에서 코딩함으로써 이루어질 수 있다. 먼저, 양자화된 변환 계수(212)의 계수 레벨(222)은 그것이 1보다 더 큰지의 여부를 결정하기 위해 체크될 수 있다. 계수 레벨(222)이 1보다 더 큰 경우, 계수 레벨(222)은 그것이 2보다 더 큰지의 여부를 결정하기 위해 체크될 수 있다.

- [0038] 도 12의 1206에서, 계수 레벨(222)이 2보다 더 큰 경우, 계수 레벨(222)은 심볼을 획득하기 위해 3의 임계 값(224)에 의해 차감될 수 있다. 비-제한적인 예시에 의해, 일부 실시예들에서, 계수 레벨(222)은 3개의 변수들: "coeff_abs_level_greater1_flag", "coeff_abs_level_greater2_flag", 및 "coeff_abs_level_minus3"로서 코딩될 수 있다. 2 이상의 계수 레벨(222)을 가지고 양자화된 변환 계수들(212)에 대해, "coeff_abs_level_greater1_flag"는 "1"로 설정될 수 있다. "coeff_abs_level_greater1_flag"가 "1"로 설정되고, 양자화된 변환 계수(212)가 또한 3 이상의 계수 레벨(222)을 갖는 경우, "coeff_abs_level_greater2_flag"가 "1"로 설정될 수 있다. "coeff_abs_level_greater2_flag"가 "1"로 설정되는 경우, 3의 임계 값(224)은 계수 레벨(222)로부터 차감되어, "coeff_abs_level_minus3"로서 코딩되는, 양자화된 변환 계수의 심볼(226)을 획득할 수 있다. 대안적인 실시예들에서, 계수 레벨(222)은 상이한 개수의 부분들에서 코딩될 수 있고, 그리고/또는 임계 값(224)은 3이 아닌 정수일 수 있다.
- [0039] 덜 빈번하게 발생하며 도 12의 블록들에서 결정된 바와 같이 3 이상의 계수 레벨들(222)을 갖는 양자화된 변환 계수들(212)에 대해, 도 12의 블록들에서 결정된 바와 같이, 양자화된 변환 계수의 심볼(226)은 도 6에서 도시된 바와 같이 생성된 최종 압축 비트들(214)의 일부분일 수 있는 이진 코드워드(228)로 변환될 수 있다.
- [0040] 도 13은 각각의 서브세트(1102)를 통해 스캔하고, 파라미터 변수(230)의 값에 따라 순서대로 서브세트(1102)의 각각의 심볼(226)을 변환하고, 이후 다음 서브세트(1102)의 심볼들(226)로 이동함으로써 각각의 심볼(226)이 코딩될 수 있는 방법을 예시한다. 이진 코드워드(228)로의 변환은 Truncated Rice 코드를 단독으로 이용하여, 또는 Truncated Rice 코드와 0차 지수-Golomb(Exp-Golomb) 코드의 조합을 이용하여 수행될 수 있다. Truncated Rice 코드는 파라미터 변수(230) 및 심볼(226)에 기초하여 이진 코드워드(228)를 획득할 수 있다. 이 코딩 진행을 보여주는 다이어그램이 도 13에서 도 11의 지그재그 라인들을 따라 서브세트들 0 및 1에 대해 도시되어 있다. 일부 실시예들에서, 현재 스캔 위치는 "n"으로 표기될 수 있다.
- [0041] 도 15를 참조하면, 파라미터 변수(230)는 각각의 심볼(226)로서 업데이트될 수 있는 전역 변수가 코딩된다. 파라미터 변수(230)는 코드워드 분포의 평탄도를 제어할 수 있다. 일부 실시예들에서, 파라미터 변수(230)는 0과 N 사이의 임의의 정수일 수 있다. 비-제한적인 예시에 의해, 일부 실시예들에서, N은 3일 수 있고, 따라서, 파라미터 변수(230)는 0, 1, 2, 또는 3일 수 있다. 일부 실시예들에서, 파라미터 변수(230)는 도 14 및 도 15에 예시된 바와 같이, "cRiceParam"로서 표기될 수 있다.
- [0042] 여전히 도 14를 참조하면, 각각의 파라미터 변수(230)는 Truncated Rice 코드에 대한 절단점(truncation point)을 표기하는 연관된 최대 심볼 값(232)을 가질 수 있다. 일부 실시예들에서, 특정 파라미터 변수(230)에 대한 최대 심볼 값(232)은, 파라미터 변수(230) "cRiceParam"에 대해 최대 심볼 값들(232) "cTRMax"의 예시적인 표를 도시하는 도 14에 예시된 바와 같이, "cTRMax"(232)로서 표기될 수 있다. 도 14의 표는, 최댓값 심볼들 cTRMax(232)에 대해 제1 리스팅 cRiceParam 값들(230)을 제공함에 따라, 표 1로서 라벨링되어 있다. 도 15의 심볼(226)이 파라미터 변수(230)에 대한 최대 심볼 값(232)보다 더 적거나 같은 경우, 심볼(226)은 오직 Truncated Rice 코드만을 사용하여 이진 코드워드(228)로 변환될 수 있다. 심볼(226)이 파라미터 변수(230)에 대한 최대 심볼 값(232)보다 더 큰 경우, 이진 코드워드(228)는 Truncated Rice 코드 및 Exp-Golomb 코드의 조합을 사용하여 생성될 수 있고, 최대 심볼 값(232)에 대한 Truncated Rice 코드워드는 심볼(226) 마이너스 최대 심볼 값(232) 마이너스 1에 대한 0차 Exp-Golomb 코드와 연결된다. 비-제한적인 예시에 의해, 도 15는 심볼들(226) 및 파라미터 변수들(230)에 기초하여 생성된 이진 코드워드(228)의 예시적인 표를 도시한다. 도 15가 다른 값들에 비해 cRiceParam 파라미터 변수들(230)을 열거하는 제2 표를 제공하므로, 그것은 표 2로서 라벨링된다.
- [0043] 일부 상황들 및/또는 실시예들에서, 더 낮은 파라미터 변수(230)를 가지고 Truncated Rice 코드에 따라 심볼(226)을 변환하는 것은 더 높은 파라미터 변수(230)를 가지고 Truncated Rice 코드에 따라 동일한 심볼(226)을 변환하는 것보다 더 적은 비트를 갖는 이진 코드워드(228)를 초래할 수 있다. 비-제한적인 예시로서, 도 15에 도시된 표에 의해 보여지는 바와 같이, 0의 심볼(226)을 변환하기 위해 0의 파라미터 변수(230)를 사용하는 것은 1 비트를 갖는 "0"의 이진 코드워드(228)를 초래할 수 있는 반면, 0의 심볼(226)을 변환하기 위해 1의 파라미터 변수(230)를 사용하는 것은 2비트를 갖는 "00"의 이진 코드워드(228)를 초래할 수 있다.
- [0044] 다른 상황들 및/또는 실시예들에서, 더 높은 파라미터 변수(230)를 가지고 Truncated Rice에 따라 심볼(226)을 변환하는 것은 더 낮은 파라미터 변수(230)를 가지고 Truncated Rice 코드에 따라 동일한 심볼(226)을 변환하는 것보다 더 적은 비트를 갖는 이진 코드워드(228)를 초래할 수 있다. 비-제한적인 예시에 의해, 도 14에 도시된 표에서 보여지는 바와 같이, 6의 심볼(226)을 변환하기 위해 0의 파라미터 변수(230)를 사용하는 것은 7비트를

갖는 "1111110"의 이진 코드워드(228)를 초래하는 반면, 6의 심볼(226)을 변환하기 위해 2의 파라미터 변수(230)를 사용하는 것은 4비트를 갖는 "1010"의 이진 코드워드(228)를 초래할 수 있다.

[0045] 도 16은 심볼들(226)을 엔트로피 코딩하는 방법을 도시하는 흐름도이다. 1602에서, 각각의 TU(210)에 대해, 파라미터 변수(230)는 제로의 값으로 초기에 설정될 수 있다. 1604에서, 코딩 시스템(110)은 다음 심볼(226)로 이동할 수 있다. 일부 상황들 및/또는 실시예들에서, 다음 심볼(226)은 도 11에 예시된 바와 같이 제1 서브세트(1102) 내의 제1 심볼(226)일 수 있다. 1606에서, 심볼(226)은 파라미터 변수(230)의 현재 값을 사용하여 Truncated Rice 및/또는 Exp-Golomb 코드를 이용하여 코딩될 수 있다. 1608에서, 파라미터 변수(230)는 파라미터 변수(230)의 마지막 값 및 코딩된 마지막 심볼(226)의 값에 기초하여 업데이트될 수 있다. 일부 상황들 및/또는 실시예들에서, 파라미터 변수(230)의 업데이트된 값은 파라미터 변수(230)의 마지막 값과 동일할 수 있다. 다른 상황들 및/또는 실시예들에서, 파라미터 변수(230)의 업데이트된 값은 파라미터 변수(230)의 마지막 값보다 더 클 수 있다. 파라미터 변수(230)는 계산들, 또는 후속적으로 본원에 기술되는 바와 같은 표로부터 유도된 값들에 기초하여 업데이트될 수 있다.

[0046] 1608에서 파라미터 변수(230)가 업데이트된 이후, 코딩 시스템(110)은 1604로 돌아가고 다음 심볼(226)로 이동할 수 있다. 다음 심볼(226)은 현재 서브세트(1102) 또는 다음 서브세트(1102)에 있을 수 있다. 다음 심볼(226)은 이후 파라미터 변수(230)의 업데이트된 값을 사용하여 1606에서 코딩될 수 있고, 프로세스는 TU(210)에서 모든 나머지 심볼들(226)에 대해 반복될 수 있다. 일부 실시예들에서, 후속적인 서브세트(1102)에서의 심볼들(226)이 코딩될 때, 파라미터 변수(230)는 이전 서브세트(1102)로부터의 파라미터 변수(230)의 마지막 값에 기초하여 업데이트될 수 있고, 따라서, 파라미터 변수(230)는 각각의 서브세트(1102)의 제1 심볼(226)에서 제로로 리셋되지 않는다. 대안적인 실시예들에서, 파라미터 변수(230)는 각각의 서브세트(1102)의 제1 심볼(226)에서 제로로 설정될 수 있다.

[0047] 일반적으로 도 15를 참조하면, 더 작은 cRiceParam 파라미터 값(230)을 갖는 Truncated Rice 코드가, 더 작은 코드워드들을 갖는 심볼들을 코딩하기 위해 선호할 수 있는데, 왜냐하면 이들을 표현하기 위해 더 적은 비트들이 필요하기 때문이다. 예를 들어, 심볼(226)이 0의 값을 갖는 경우, 0과 같은 cRiceParam 파라미터 값(230)을 갖는 Truncated Rice 코드를 사용하는 것은 오직 1비트를 요구하지만, cRiceParam 값이 각각 2, 3, 또는 4일 때 2, 3, 또는 4비트가 요구된다. 심볼이 6의 값을 갖는 경우, 0, 7 비트와 같은 cRiceParam 값을 갖는 Truncated Rice 코드를 사용하는 것이 요구된다. 그러나, cRiceParam 값이 각각 2, 3 또는 4일 때 5, 4, 또는 4비트가 요구된다.

[0048] 도 17의 표로 예시된 일 실시예에서, 변수 coeff_level_minus3[n]로 라벨링된 cRiceParam(230)이, 후속하는 바와 같이 표에 기초하여 유도되고 업데이트된다. TU 서브세트에 대해, cRiceParam(230)은 초기에 0으로 설정되고, 이후 도 17의 표에 따른 이전 cRiceParam 및 coeff_abs_level_minus3[n-1]에 기초하여 업데이트된다. 도 17이 cRiceParam 파라미터 값들(230)에 대해 심볼값들(226)을 열거하는 제3 표를 도시하므로, 표는 표 3으로서 라벨링된다. 유사한 비교를 보이는 후속하는 표들도 마찬가지로 연속적으로 라벨링될 것이다.

[0049] 종래의 구현예들에서, cRiceParam(230)은 초기 "0" 값들을 갖는 서브세트 마다 한번 리셋된다. 16개의 연속적인 심볼 계수들(226)의 하나 초과와 서브세트를 갖는 TU에 대해, coeff_abs_level_minus3에 대한 cRiceParam 계산이 각각의 서브세트에 대해 0으로 리셋될 수 있는데, 이는 더 작은 심볼 값 코딩을 선호한다. 일반적으로, 각각의 TU 내에서, 마지막 n -제로 양자화된 변환 계수로 시작하여, n -제로 양자화된 변환 계수들의 절댓값들은 점점 더 커지는 경향이 있다. 따라서, 각각의 서브세트에 대해 cRiceParam을 0으로 리셋하는 것은 최적의 압축 성능을 제공하지 않을 수 있다.

[0050] 도 13에서, 각각의 원은 양자화된 변환 계수를 나타내고, 각각의 원 내의 수는 coeff_abs_level_minus3의 값이다. 그것이 "NA"인 경우, 그것은 그 계수에 대해 coeff_abs_level_minus3의 선택자가 존재하지 않음을 의미한다. 도 13의 예에서 보여지는 바와 같이, 역 스캔 패턴에 후속하여, coeff_abs_level_minus3의 값들은 각각의 서브세트 내에서, 그리고 또한 서브세트마다 점점 더 커지는 경향이 있다. 도 15의 표 2에서 보여지는 바와 같이, 예에서, cRiceParam는 서브세트 0에서 "5"에 대해 2로 설정되고, 2로 설정된 cRiceParam를 이용하여, "5"의 값은 "1001", 즉 4비트의 코드워드로 이진화된다. 통상적인 구현예들에서, cRiceParam는 이후 서브세트 1에서 0으로 리셋된다. 표 2에 보여지는 바와 같이, 이제, 0의 리셋 cRiceParam을 가지고, 서브세트 1 내의 "5"의 동일한 값이 이제 111110, 즉 6비트의 코드워드로 이진화된다. 명백하게, 이 리셋 프로세스는 추가적인 체크 동작들을 도입할 뿐만 아니라 가능하게는 더 낮은 코딩 성능을 초래할 수 있다.

[0051] 각각 도 18 및 19에 예시된 바와 같은 표 4 및 5는 업데이트 표에 대한 대안적인 실시예들을 도시한다. 이들

및 다른 실시예들에 대해, cRiceParam 파라미터들(230)은 후속하는 바와 같이 유도된다. 먼저, TU에 대해, cRiceParam은 초기에 0으로 설정되고, 이후, 표 4 및 5와 같은 cRiceParam 업데이트 표에 따른 이전 cRiceParam 및 $\text{coeff_abs_level_minus3}[n-1]$ 에 기초하여 업데이트된다. 이들 실시예들에서, cRiceParam은 표 3을 사용하여 실시예에 대해 표시된 바와 같이 TU의 서브세트마다가 아니라, TU마다 오직 한번만 리셋된다.

[0052] 각각의 서브세트에서 cRiceParam을 0으로 리셋하지 않음으로써, 각각의 서브세트에 대한 리셋의 동작들이 절감되고, cRiceParam가 3에 도달하면, 심볼들은 항상 동일하게 설정된 절단된 라이스(truncated rice) 코드들을 이용하여 이진화될 것이며(cRiceParam은 3과 같음), 이는 하드웨어 복잡도를 감소시킬 수 있다.

[0053] 도 19의 표 5가, 다음 레벨 값이 역 스캔을 따라 현재 레벨보다 통계적으로 더 작지 않다고 가정하는 동안, 상이한 cRiceParam 값(230)을 가지고 각각의 심볼(226)에 대해 요구되는 비트수를 분석함으로써 표 15의 표 2로부터 생성된다는 점에 유의한다. 예를 들어, 현재 심볼(226)이 2이고 cRiceParam이 0인 경우, 다음 심볼이 2보다 더 클 가능성은 높으며, 1과 같은 cRiceParam을 갖는 Truncated Rice 코드를 적용하는 것은 비트수를 감소시킬 수 있다. 현재 심볼이 5이고 cRiceParam이 1인 경우, 다음 심볼이 5보다 더 클 가능성은 높으며, 2와 같은 cRiceParam를 갖는 Truncated Rice 코드를 적용하는 것은 비트수를 감소시킬 수 있다. 현재 심볼이 11이고 cRiceParam이 2인 경우, 다음 심볼이 11보다 더 클 가능성은 높으며, 3와 같은 cRiceParam를 갖는 Truncated Rice 코드를 적용하는 것은 비트수를 감소시킬 수 있다.

[0054] 일부 실시예에서, 다시 도 16을 참조하면, 1608에서 파라미터 변수(230)를 업데이트하는 것은 표가 아닌 비교 수식으로부터 결정될 수 있다. 비교 시에, 파라미터 변수(230)의 마지막 값 및 마지막으로 코딩된 심볼(226)의 값 모두가 도 20에 예시된 바와 같은 하나 이상의 조건들(1702)을 만족시키는지의 여부가 결정된다. 일부 실시예들에서, 마지막으로 코딩된 심볼(226)의 값은 표들 3-5에서 그런 것처럼 " $\text{coeff_abs_level_minus3}[n-1]$ "로서 표기될 수 있다. 파라미터 변수(230)는 어느 조건들(1702)이 만족되는지에 따라 업데이트될 수 있고, 현재 심볼(226)의 값은 이후 Truncated Rice 코드 및/또는 Exp-Golomb 코드를 사용하여 업데이트된 파라미터 변수(230)에 기초하여 코딩될 수 있다.

[0055] 일부 실시예들에서, 각각의 조건(1702)은 2개의 부분들인, 조건부 심볼 임계 및 조건부 파라미터 임계를 포함할 수 있다. 이들 실시예들에서, 조건(1702)은 심볼(226)의 값이 조건부 심볼 임계보다 더 크거나 같고 파라미터 변수(230)가 조건부 파라미터 임계보다 더 크거나 같은 경우 만족될 수 있다. 대안적인 실시예들에서, 각각의 조건(1702)은 심볼(226) 및 파라미터 변수(230) 중 어느 하나 또는 둘 모두에 대해 임의의 개수의 부분들을 가지거나 임의의 타입의 조건을 가질 수 있다.

[0056] 업데이트 표가 데이터를 저장하고 패치(fetch)하기 위한 여분의 메모리를 요구할 수 있고, 메모리가 많은 프로세서 사이클들을 요구할 수 있으므로, 로직이 매우 적은 프로세서 사이클들을 사용함에 따라 업데이트 표 대신 비교를 수행하기 위해 조합 로직들을 사용하는 것이 바람직할 수 있다. 표 3 대신 업데이트를 위해 cRiceParam를 결정하는 조합 로직의 예가 도 20에 도시되어 있다. 표 4를 나타내기 위한 조합 로직의 예가 도 21에 도시되어 있다. 표 5를 나타내기 위한 조합 로직의 예가 도 22에 도시되어 있다.

[0057] 일부 실시예들에서, 파라미터 변수(230) 및 마지막으로 코딩된 심볼들(226)의 가능한 값들에 기초한 조건들(1702)의 가능한 결과들은 도 17 및 다른 후속 도면들의 표에 예시된 바와 같이 낮은 복잡도의 업데이트 표(1704)로서 메모리에 저장될 수 있다. 이들 실시예들에서, 파라미터 변수(230)는 파라미터 변수(230)의 마지막 값 및 마지막으로 코딩된 심볼(226)의 값에 기초하여 낮은 복잡도 업데이트 표(1704)로부터 표 검색을 수행함으로써 업데이트될 수 있다.

[0058] 추가적인 실시예들에서, 도 20-22에 예시된 로직을 요구하는 것이 아니라 일부 실시예들에서 이전 표들보다 더 효율적으로 작용할 수 있는 CABAC에서 낮은 복잡도 레벨 파라미터 업데이트 표가 제공될 수 있다. 이들 낮은 복잡도 레벨 파라미터 업데이트 표들에 대해, 다음이 적용된다: (1) 입력: 이전 cRiceParam 및 $\text{coeff_abs_level_minus3}[n-1]$. (2) 출력: cRiceParam. (3) 이전 cRiceParam 및 cRiceParam은 0, 1, 2 또는 3의 값을 가질 수 있다.

[0059] 또한 이러한 낮은 복잡도 레벨 파라미터 업데이트 표들에서, 다음이 추가로 적용된다: (1) 파라미터 변수(230)는: 마지막으로 코딩된 심볼(226)의 값이 0과 A-1 사이에 있을 때 동일하게 유지될 수 있다; (2) 파라미터 변수(230)는, 심볼(226)이 A와 B-1 사이에 있을 때, 어느 것이 더 크든, 1로 설정되거나 또는 파라미터 변수(230)의 마지막 값에서 유지될 수 있다; (3) 파라미터 변수(230)는 심볼(226)이 B와 C-1 사이에 있을 때, 어느 것이 더 크든, 2로 설정되거나 또는 파라미터 변수(230)의 마지막 값에서 유지될 수 있다; 또는 (4) 파라미터 변수(230)

0)는 심볼(226)이 C-1보다 더 클 때 3으로 설정될 수 있다. 표 6으로 라벨링된 낮은 복잡도 업데이트 표(1704)는, 이들 조건들(1702)에 대해, 도 23에 도시된다. 표 6의 조합 논리 표현이 도 24에 도시된다. A, B, 및 C의 값들은 임의의 원하는 값들로 설정될 수 있다. 이러한 예시적인 실시예에서, A, B 또는 C는 각각 조건부 심볼 임계일 수 있고, 0, 1, 또는 2의 값은 각각 파라미터 심볼 임계일 수 있다.

[0060] 업데이트 표들(1704) 및 A, B 및 C의 특정 값들을 갖는 이들의 연관된 조합 논리 표현들(1706)의 비-제한적인 예시들의 선택이 도 19-31에 도시되어 있다. 도 19 및 20은 각각 업데이트 표(1704) 및 3, 6 및 13의 조건부 심볼 임계들에 대한 조합 논리 표현을 도시한다. 도 29 및 30은 각각 업데이트 표 9 및 2, 4 및 11의 조건부 심볼 임계들에 대한 조합 논리 표현을 도시한다. 도 31 및 32는 각각 업데이트 표 10 및 2, 4 및 10의 조건부 심볼 임계들의 조합 논리 표현을 도시한다.

[0061] 실시예들을 구현하기 위해 요구되는 명령들의 시퀀스들의 실행은 도 33에 도시된 바와 같이 컴퓨터 시스템(3300)에 의해 수행될 수 있다. 실시예에서, 명령들의 시퀀스들의 실행은 단일 컴퓨터 시스템(3300)에 의해 수행된다. 다른 실시예들에 따라, 통신 링크(3315)에 의해 연결된 둘 이상의 컴퓨터 시스템들(3300)은 서로 합동으로 명령들의 시퀀스를 수행할 수 있다. 오직 하나의 컴퓨터 시스템(3300)의 기재만이 본원에 제시될 수 있지만, 임의의 개수의 컴퓨터 시스템들(3300)이 사용될 수 있다는 점이 이해되어야 한다.

[0062] 실시예에 따른 컴퓨터 시스템(3300)은 이제, 컴퓨터 시스템(3300)의 기능 컴포넌트의 블록도인 도 33에 관해 기술될 것이다. 본원에서 사용되는 바와 같이, 용어 컴퓨터 시스템(3300)은 하나 이상의 프로그램들을 저장하며 독립적으로 실행할 수 있는 임의의 컴퓨팅 디바이스를 기술하기 위해 널리 사용된다.

[0063] 컴퓨터 시스템(3300)은 버스(3306)에 연결된 통신 인터페이스(3314)를 포함할 수 있다. 통신 인터페이스(3314)는 컴퓨터 시스템들(3300) 사이의 양방향 통신을 제공한다. 개별 컴퓨터 시스템(3300)의 통신 인터페이스(3314)는 다양한 타입의 신호 정보, 예를 들어, 명령들, 메시지들 및 데이터를 나타내는 데이터 스트림들을 포함하는 전기, 전자기 또는 광학 신호들을 전송하고 수신한다. 통신 링크(3315)는 하나의 컴퓨터 시스템(3300)을 또 다른 컴퓨터 시스템(3300)과 연결시킨다. 예를 들어, 통신 링크(3315)는 LAN, 통합 서비스 디지털 네트워크(ISDN) 카드, 모뎀, 또는 인터넷일 수 있다.

[0064] 컴퓨터 시스템(3300)은 프로그램들, 즉 애플리케이션, 코드를 포함하는, 메시지들, 데이터 및 명령들을, 각자의 통신 링크(3315) 및 통신 인터페이스(3314)를 통해 전송하고 수신할 수 있다. 수신된 프로그램 코드는 그것이 수신될 때 각자의 프로세서(들)(3307)에 의해 실행되고 그리고/또는 추후 실행을 위해, 저장 디바이스(3310), 또는 다른 연관된 비휘발성 매체에 저장될 수 있다.

[0065] 실시예에서, 컴퓨터 시스템(3300)은 데이터 저장 시스템(3331), 예를 들어, 컴퓨터 시스템(3300)에 의해 용이하게 액세스가능한 데이터베이스(3332)를 포함하는 데이터 저장 시스템(3331)과 공조하여 동작한다. 컴퓨터 시스템(3300)은 데이터 인터페이스(3333)를 통해 데이터 저장 시스템(3331)과 통신한다.

[0066] 컴퓨터 시스템(3300)은 버스(3306) 또는 명령들, 메시지들 및 데이터, 집합적으로 정보를 통신하기 위한 다른 통신 메커니즘, 및 정보를 프로세싱하기 위해 버스(3306)와 연결된 하나 이상의 프로세서들(3307)을 포함할 수 있다. 컴퓨터 시스템(3300)은 또한, 프로세서(들)(3307)에 의해 실행될 동적 데이터 및 명령들을 저장하기 위해 버스(3306)에 연결된 랜덤 액세스 메모리(RAM) 또는 다른 동적 저장 디바이스와 같은, 주메모리(3308)를 포함한다. 컴퓨터 시스템(3300)은 프로세서(들)(3307)에 대한 정적 데이터 및 명령들을 저장하기 위해 버스(3306)에 연결된 판독 전용 메모리(ROM)(3309) 또는 다른 정적 저장 디바이스를 더 포함할 수 있다. 자기 디스크 또는 광학 디스크와 같은 저장 디바이스(3310)가 한 프로세서(들)(3307)에 대한 데이터 및 명령들을 저장하기 위해 제공되며 버스(3306)에 연결될 수 있다.

[0067] 컴퓨터 시스템(3300)은 LCD 스크린과 같은 디스플레이 디바이스(3311)에 버스(3306)를 통해 연결될 수 있다. 입력 디바이스(3312), 예를 들어, 영숫자 및 다른 키들이 프로세서(들)(3307)에 정보 및 커맨드 선택들을 전달하기 위해 버스(3306)에 연결된다.

[0068] 일 실시예에 따라, 개별 컴퓨터 시스템(3300)은 주메모리(3308)에 포함된 하나 이상의 명령들의 하나 이상의 시퀀스들을 실행하는 각자의 프로세서(들)(3307)에 의해 특정 동작들을 수행한다. 이러한 명령들은 ROM(3309) 또는 저장 디바이스(3310)와 같은, 또 다른 컴퓨터-사용가능한 매체로부터 주 메모리(3308) 내로 판독될 수 있다. 주 메모리(3308)에 포함된 명령들의 시퀀스들의 실행은 프로세서(들)(3307)로 하여금 본원에 기술된 프로세스들을 수행하게 한다. 대안적인 실시예들에서, 하드-베션 회로가 소프트웨어 명령들 대신 또는 소프트웨어 명령들과 함께 사용될 수 있다. 따라서, 실시예들은 하드웨어 회로 및/또는 소프트웨어의 임의의 특정 조합에 제한되

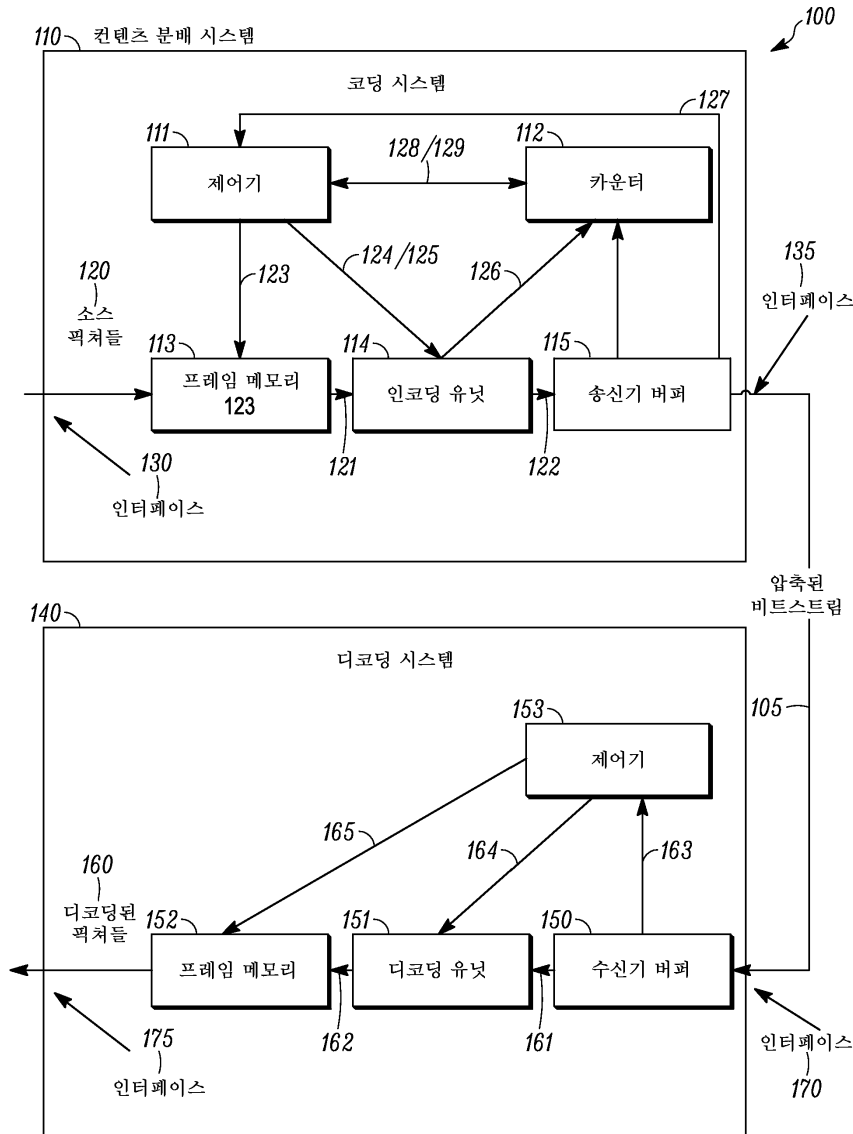
지 않는다.

[0069]

본 발명이 특수성을 가지고 진술되었지만, 이것은 단지 당업자에게 발명을 실시하고 사용하는 방법을 교시하는 것이었다. 많은 추가적인 수정들이 발명의 범위 내에 들 것이며, 그 범위는 후속하는 청구항들에 의해 정의된다.

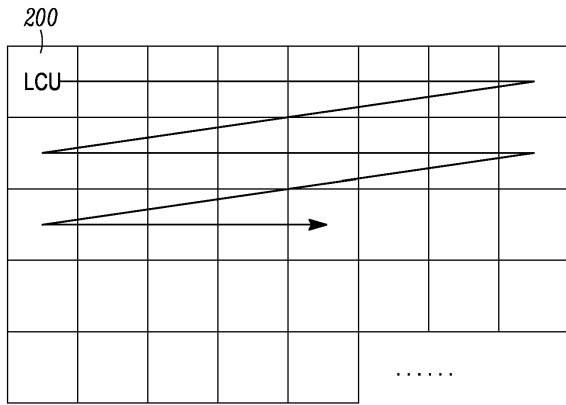
도면

도면1



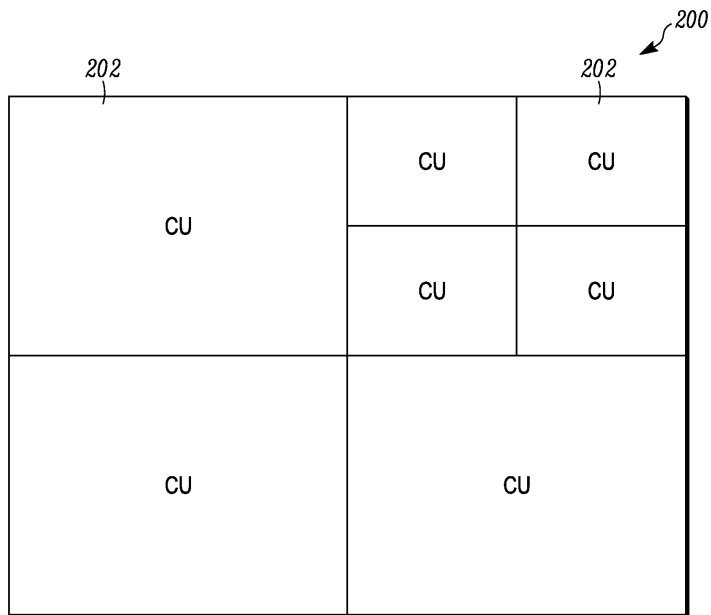
종래 기술

도면2



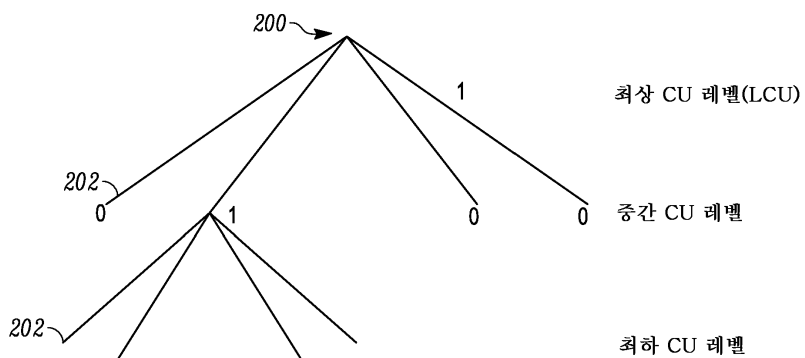
종래 기술

도면3

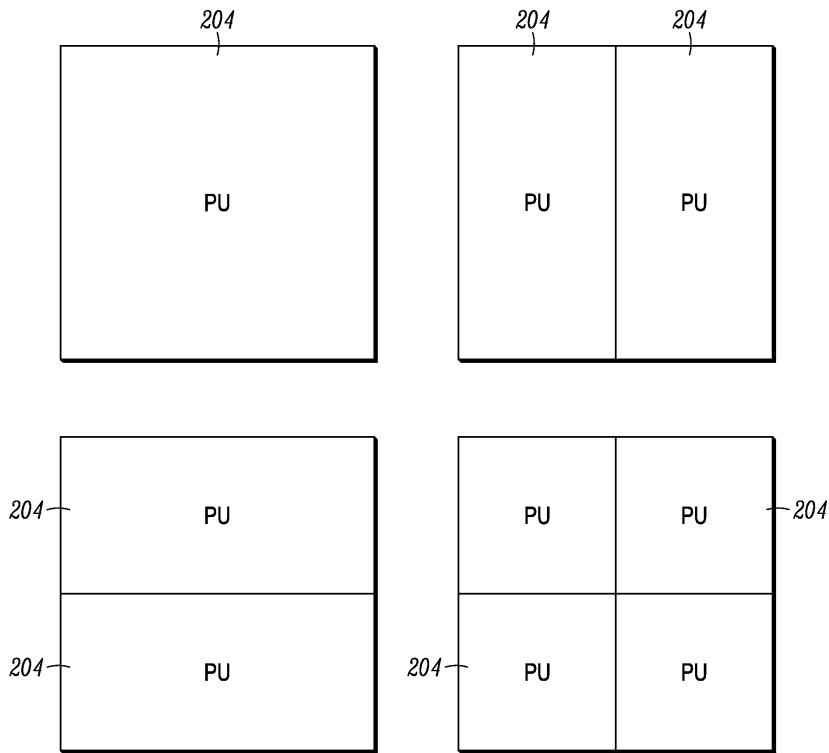


종래 기술

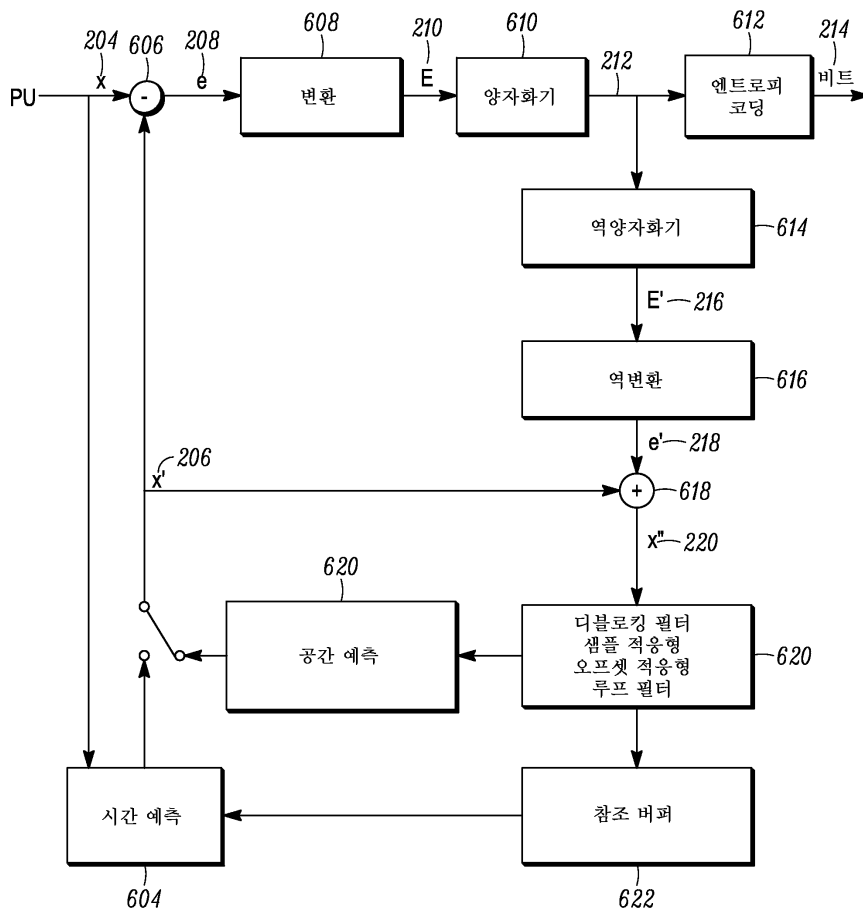
도면4



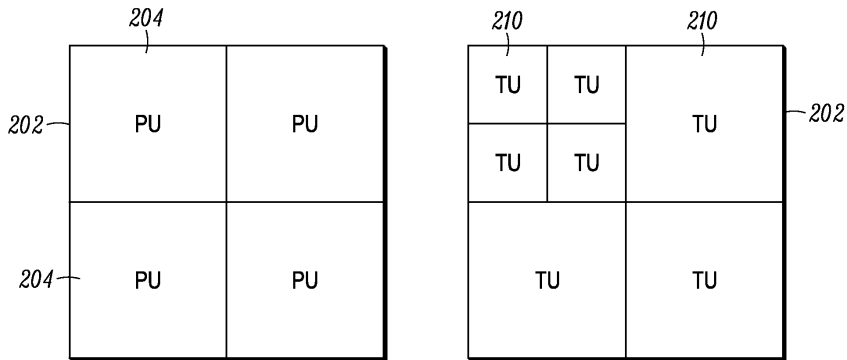
도면5



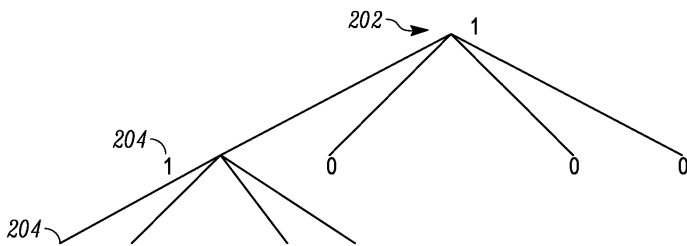
도면6



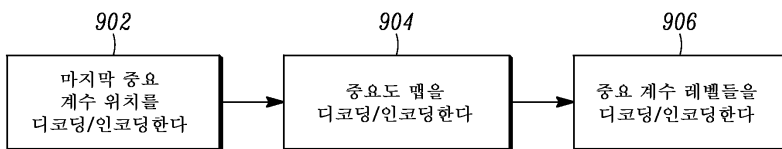
도면7



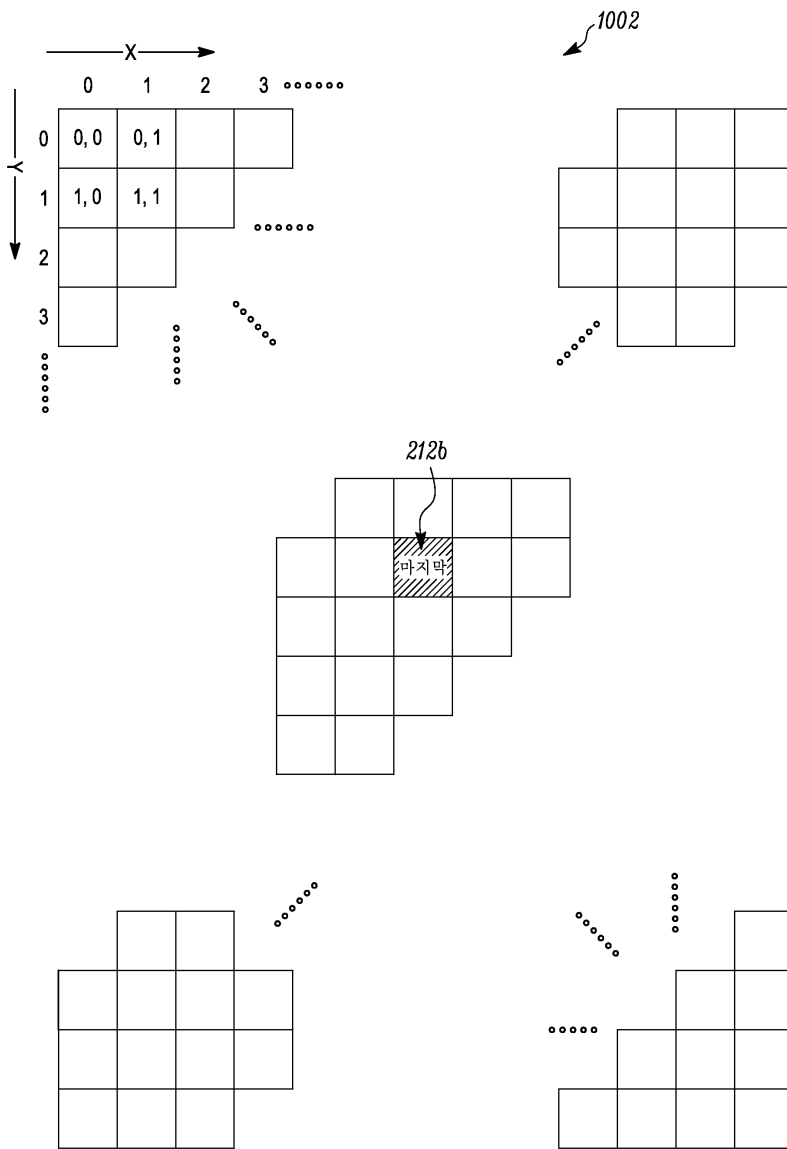
도면8



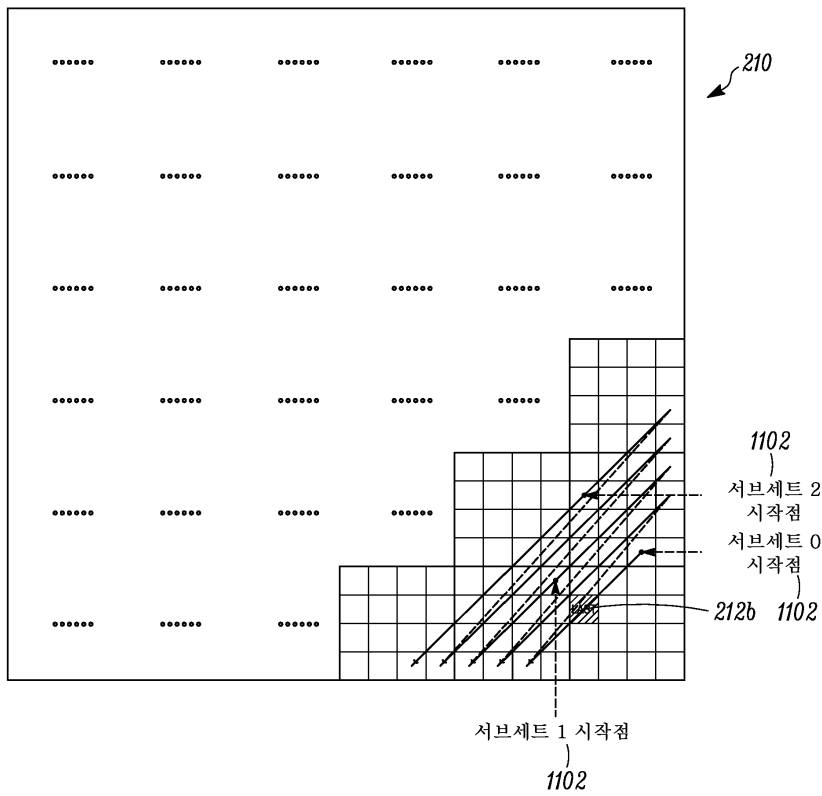
도면9



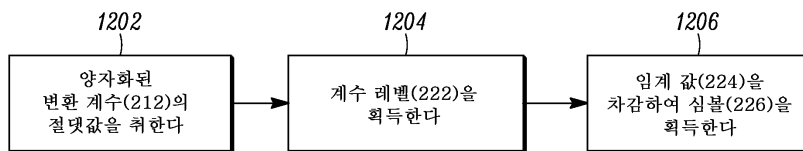
도면10



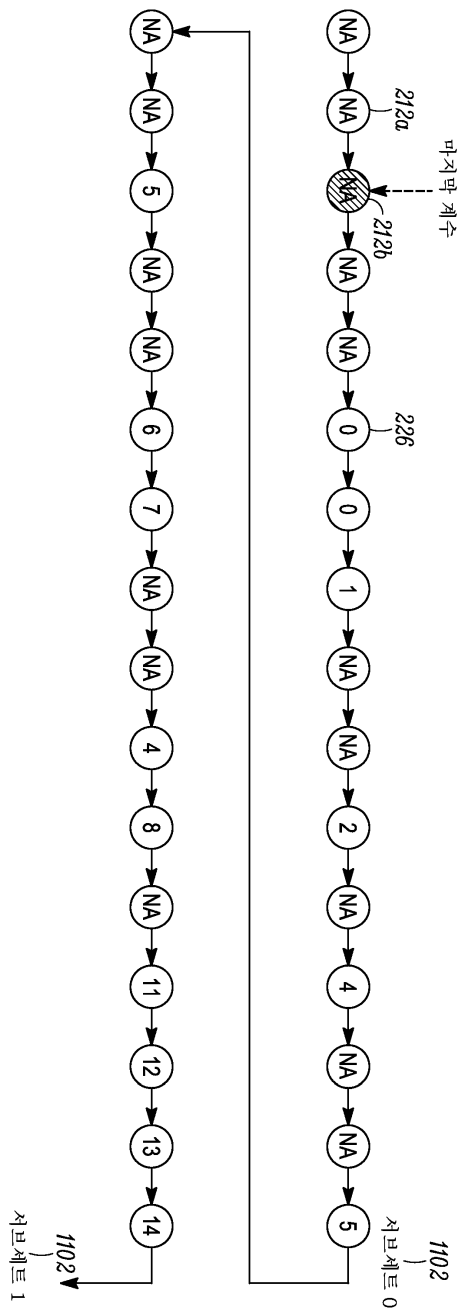
도면11



도면12



도면13



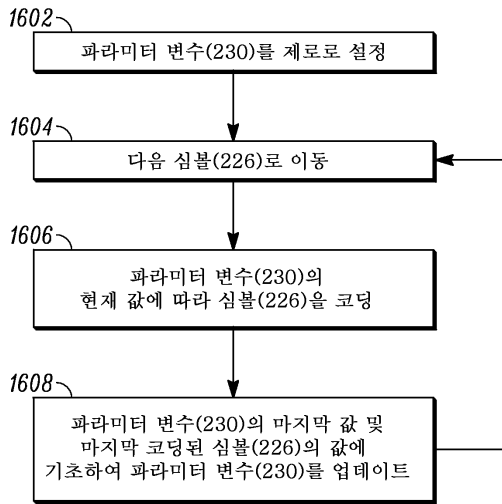
도면14

cRICEPARAM	cTRMAX
0	7
1	20
2	42
3	70

도면15

cRICEPARAM	228 0		230 1		2		3	
	코드워드	비트	코드워드	비트	코드워드	비트	코드워드	비트
0	0	1	00	2	000	3	0000	4
1	10	2	01	2	001	3	0001	4
2	110	3	100	3	010	3	0010	4
3	1110	4	101	3	011	3	0011	4
4	11110	5	1100	4	1000	4	0100	4
5	111110	6	1101	4	1001	4	0101	4
6	1111110	7	11100	5	1010	4	0110	4
7	11111110	8	11101	5	1011	4	0111	4
8	11111111, EGO	9	111100	6	11000	5	10000	5
9			111101	6	11001	5	10001	5
10			1111100	7	11010	5	10010	5
11			1111101	7	11011	5	10011	5
12			11111100	8	111000	6	10100	5
13			11111101	8	111001	6	10101	5
14			111111100	9	111010	6	10110	5
15			111111101	9	111011	6	10111	5
16			1111111100	10	1111000	7	110000	6
17			1111111101	10	1111001	7	110001	6
18			11111111100	11	1111010	7	110010	6
19			11111111101	11	1111011	7	110011	6
20			11111111110	11	11111000	8	110100	6
21			11111111111, EGO	12	11111001	8	110101	6
22					11111010	8	110110	6
23					11111011	8	110111	6
24					111111000	9	1110000	7
25					111111001	9	1110001	7
26					111111010	9	1110010	7
27					111111011	9	1110011	7
28					1111111000	10	1110100	7
29					1111111001	10	1110101	7
30					1111111010	10	1110110	7
31					1111111011	10	1110111	7
32					11111111000	11	11110000	8
33					11111111001	11	11110001	8
34					11111111010	11	11110010	8
35					11111111011	11	11110011	8
36					111111111000	12	11110100	8
37					111111111001	12	11110101	8
38					111111111010	12	11110110	8
39					111111111011	12	11110111	8
40					111111111100	12	111110000	9
41					111111111101	12	111110001	9
42					111111111110	12	111110010	9
43					111111111111, EGO	13	111110011	9
44							111110100	9
45							111110101	9
46							111110110	9
47							111110111	9
48							1111110000	10
49							1111110001	10
50							1111110010	10
51							1111110011	10
52							1111110100	10
53							1111110101	10
54							1111110110	10
55							1111110111	10
56							11111110000	11
57							11111110001	11
58							11111110010	11
59							11111110011	11
60							11111110100	11
61							11111110101	11
62							11111110110	11
63							11111110111	11
64							11111111000	11
65							11111111001	11
66							11111111010	11
67							11111111011	11
68							11111111100	11
69							11111111101	11
70							11111111110	11
71							11111111111, EGO	12

도면16



도면17

표 3

1704

		COEFF_ABS_LEVEL_MINUS 3[n-1]															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	≥15
이전 PRICEPARAM	0	0	0	0	1	1	2	2	2	2	2	2	2	2	2	3	3
	1	1	1	1	1	2	2	2	2	2	2	2	3	3	3	3	3
	2	2	2	2	2	2	2	2	2	2	2	3	3	3	3	3	3
CRICEPARAM	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3

226

230

230

도면18

표 4

1704

		COEFF_ABS_LEVEL_MINUS 3[n-1]															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	≥15
이전 PRICEPARAM	0	0	0	0	1	1	1	2	2	2	2	2	2	3	3	3	3
	1	1	1	1	1	1	2	2	2	2	2	2	3	3	3	3	3
	2	2	2	2	2	2	2	2	2	2	2	2	3	3	3	3	3
PRICEPARAM	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3

226

230

230

도면22

cRICEPARAM = cRICEPARAM
 + (COEFF_ABS_LEVEL_MINUS 3(n-1) >= 2 && cRICEPARAM == 0) — 1702
 + (COEFF_ABS_LEVEL_MINUS 3(n-1) >= 5 && cRICEPARAM <= 1) — 1702
 + (COEFF_ABS_LEVEL_MINUS 3(n-1) >= 11 && cRICEPARAM <= 2); — 1702

1706 ↗

도면23

1704 ↘

	COEFF_ABS_LEVEL_MINUS 3(n-1)														
	0	...	0	...	A	B	>C
이전 cRICEPARAM	0	1	2	3	3	3	3	3	3	3	3	3	3	3	3

230 ↗ (pointing to the first column)

230 ↗ (pointing to the last column)

226 ↗ (pointing to the >C column)

표 6

도면24

cRICEPARAM = cRICEPARAM
 + (COEFF_ABS_LEVEL_MINUS 3(n-1) >= A && cRICEPARAM == 0) — 1702
 + (COEFF_ABS_LEVEL_MINUS 3(n-1) >= B && cRICEPARAM <= 1) — 1702
 + (COEFF_ABS_LEVEL_MINUS 3(n-1) >= C && cRICEPARAM <= 2); — 1702

도면25

1704

		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	>15
		0	0	0	1	1	2	2	2	2	2	2	2	2	2	3	3
		1	1	1	1	1	2	2	2	2	2	2	2	2	3	3	3
		2	2	2	2	2	2	2	2	2	2	2	2	2	3	3	3
		3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
		COEFF_ABS_LEVEL_MINUS_3(n-1)															
		226															
이전																	
cRICEPARAM																	
	230																
	230																

도면26

cRICEPARAM = cRICEPARAM

1706

- + (COEFF_ABS_LEVEL_MINUS_3(n-1) >= 2 && cRICEPARAM == 0) — 1702
- + (COEFF_ABS_LEVEL_MINUS_3(n-1) >= 4 && cRICEPARAM <= 1) — 1702
- + (COEFF_ABS_LEVEL_MINUS_3(n-1) >= 12 && cRICEPARAM <= 2) — 1702

도면33

