



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2019년04월25일  
(11) 등록번호 10-1972366  
(24) 등록일자 2019년04월19일

(51) 국제특허분류(Int. Cl.)  
H04N 19/117 (2014.01) H04N 19/132 (2014.01)  
H04N 19/172 (2014.01) H04N 19/186 (2014.01)  
(52) CPC특허분류  
H04N 19/117 (2015.01)  
H04N 19/132 (2015.01)  
(21) 출원번호 10-2018-7032810(분할)  
(22) 출원일자(국제) 2012년11월07일  
심사청구일자 2018년11월13일  
(85) 번역문제출일자 2018년11월13일  
(65) 공개번호 10-2018-0125041  
(43) 공개일자 2018년11월21일  
(62) 원출원 특허 10-2014-7017658  
원출원일자(국제) 2012년11월07일  
심사청구일자 2017년11월03일  
(86) 국제출원번호 PCT/EP2012/072069  
(87) 국제공개번호 WO 2013/068427  
국제공개일자 2013년05월16일  
(30) 우선권주장  
1119206.9 2011년11월07일 영국(GB)  
1203285.0 2012년02월24일 영국(GB)  
(56) 선행기술조사문헌  
EP02237557 A1  
KR1020110111852 A  
C-M Fu et al. "Sample Adaptive Offset for  
chroma", JCT-VC of ITU-T SG16 WP3 and ISO/IEC  
JTC1/SC29/WG11, 6th Meeting: Torino, IT, 14-22  
July, 2011, JCTVC-F057.

(73) 특허권자  
캐논 가부시끼가이샤  
일본 도쿄도 오오따꾸 시모마루코 3조메 30방 2고  
(72) 발명자  
프랑수와 에두아르  
프랑스 에프-35890 부르그 데 콩뜨 알레 뒤 로까  
르 18  
라로슈 기욤  
프랑스 에프-35520 플레스 르 오 로내  
(74) 대리인  
장수길, 이중희

전체 청구항 수 : 총 3 항

심사관 : 김영태

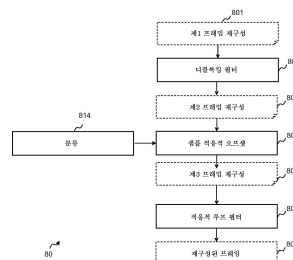
(54) 발명의 명칭 샘플 적응적 오프셋을 적용하는 장치

(57) 요약

영상의 재구성된 샘플 세트에 대한 보상 오프셋들이 제공된다. 각각의 샘플은 샘플 값을 가진다. 보상 오프셋들을 제공하는 방법은 레이트 왜곡 기준(rate distortion criterion)에 기초하여, 복수의 사전 결정된 분류들 중에서 분류를 선택하는 단계를 포함한다. 각각의 사전 결정된 분류는 샘플 값의 전체 범위보다 작은 분류 범위

(뒷면에 계속)

대표도 - 도8



(classification range)를 가지며 복수의 클래스들로 이루어져 있고, 각각의 클래스는 분류 범위 내에서의 샘플 값의 범위를 정의하고, 샘플의 샘플 값이 관련 클래스의 범위 내에 있는 경우 샘플이 그 클래스 내에 배치된다. 클래스의 각각의 샘플의 샘플 값에 적용하기 위한 보상 오프셋이 그 선택된 분류의 각각의 클래스와 연관된다.

(52) CPC특허분류

**H04N 19/172** (2015.01)

**H04N 19/186** (2015.01)

## 명세서

### 청구범위

#### 청구항 1

샘플 적응적 오프셋(sample adaptive offset)을 영상에 적용하는 장치로서,

비트스트림으로부터, 샘플 적응적 오프셋의 대역 유형에 관한 위치 데이터를 취득하기 위한 수단으로서, 취득된 상기 위치 데이터는, 크로마 샘플 값들의 전체 범위로부터 선택된 상기 샘플 적응적 오프셋이 적용될 크로마 샘플 값들의 범위의 위치를 나타내고, 상기 범위는 상기 전체 범위의 절반보다 작고, 상기 범위는 미리 결정된 수의 연속적 하위 범위를 갖고, 상기 미리 결정된 수의 연속적 하위 범위의 각각은 복수의 크로마 샘플 값을 포함하는 것인, 상기 위치 데이터를 취득하기 위한 수단;

상기 비트스트림으로부터, 상기 연속적 하위 범위에 대한 각 오프셋 값을 취득하기 위한 수단;

상기 비트스트림으로부터, 상기 비트스트림으로부터 디코딩된 영상의 샘플들의 크로마 샘플 값들을 취득하기 위한 수단; 및

각각의 상기 하위 범위에 대한 오프셋 값을 해당 하위 범위의 크로마 샘플 값들에 적용하기 위한 수단을 포함하는, 장치.

#### 청구항 2

제1항에 있어서, 상기 위치는 상기 샘플 적응적 오프셋이 적용되는 범위의 끝부분 위치인, 장치.

#### 청구항 3

제1항에 있어서, 상기 위치는 상기 전체 범위의 한쪽 끝부분으로부터의 변위로서 표현되는, 장치.

### 발명의 설명

#### 기술 분야

[0001] 본 발명은 영상의 재구성된 샘플 세트에 대한 보상 오프셋들(compensation offsets)을 제공하는 방법 및 장치에 관한 것이다. 본 발명은 또한 디지털 영상 시퀀스를 인코딩 또는 디코딩하는 방법 및 장치에 관한 것이다.

#### 배경 기술

[0002] 본 발명은 디지털 신호 처리의 분야에서 그리고 특히 비디오 스트림에서 공간 및 시간 중복성을 감소시키기 위해 움직임 보상을 사용하는 비디오 압축의 분야에서 적용될 수 있다.

[0003] 예를 들어, H.263, H.264, MPEG-1, MPEG-2, MPEG-4, SVC 등의 많은 비디오 압축 형식들은 공간 및 시간 중복성을 제거하기 위해 블록 기반 DCT(discrete cosine transform) 및 움직임 보상을 사용한다. 이들은 종종 예측 비디오 형식(predictive video format)이라고 한다. 비디오 신호의 각각의 프레임 또는 영상은 독립적으로 인코딩되고 디코딩될 수 있는 슬라이스들로 나뉘어진다. 슬라이스는 통상적으로 프레임의 직사각형 부분, 또는 보다 일반적으로, 프레임 또는 전체 프레임의 일부분이다. 게다가, 각각의 슬라이스는 매크로블록들(macroblock, MB)로 나뉘어질 수 있고, 각각의 매크로블록은 블록들(통상적으로, 64x64, 32x32, 16x16 또는 8x8 픽셀의 블록들)로 추가로 나뉘어진다.

[0004] HEVC(High Efficiency Video Coding)에서, 64x64 내지 4x4의 블록들이 사용될 수 있다. 이 분할은 LCU(largest coding unit, 최대 코딩 단위)에 기초한 쿼드-트리 구조(quad-tree structure)에 따라 구성되어 있다. LCU는 64x64의 정사각형 블록에 대응한다. LCU가 나뉘어질 필요가 있는 경우, 분할 플래그(split flag)는 LCU가 4개의 32x32 블록들로 분할되어 있다는 것을 나타낸다. 동일한 방식으로, 이 4개의 블록들 중 임의의 것이 분할될 필요가 있는 경우, 분할 플래그는 참(true)으로 설정되고, 32x32 블록이 4개의 16x16 블록들로 나누어지며, 이하 마찬가지이다. 분할 플래그가 거짓(false)으로 설정되어 있을 때, 현재의 블록이 CU(coding unit, 코딩 단위)이다. CU는 64x64, 32x32, 16x16 또는 8x8 픽셀의 크기를 가진다.

- [0005] 영상의 블록들을 코딩하는 2가지 계열의 코딩 모드 - 공간 예측에 기초한 코딩 모드[인트라 예측(INTRA prediction)이라고 함], 및 시간 예측에 기초한 코딩 모드[인터(INTER), 병합(Merge), 건너뛰(Skip)] - 가 있다. 공간 예측 모드 및 시간 예측 모드 둘 다에서, 원래의 블록으로부터 예측을 차감함으로써 잔차(residual)가 계산된다.
- [0006] 인트라 블록(INTRA block)은 일반적으로 인트라 예측 프로세스에 의해 그의 인과 경계(causal boundary)에서의 인코딩된 픽셀로부터 예측된다. 인트라 예측에서, 예측 방향이 인코딩된다.
- [0007] 시간 예측은, 비디오 시퀀스의 이전 또는 장래 프레임(previous or a future frame)인 참조 프레임(reference frame)에서, 인코딩된 블록에 가장 가까운 영상 부분(image portion) 또는 참조 영역(reference area)을 찾아내는 데 있다. 이 단계는 통상적으로 움직임 추정(motion estimation)이라고 한다. 그 다음에, 통상적으로 움직임 보상(motion compensation)이라고 하는 단계에서 참조 영역을 사용하여 인코딩된 블록이 예측된다 - 인코딩된 블록과 참조 부분 사이의 차가, 움직임 보상을 위해 사용할 참조 영역을 나타내는 움직임 벡터에 대한 움직임 정보의 항목과 함께, 인코딩된다 -. 시간 예측에서, 적어도 하나의 움직임 벡터가 인코딩된다.
- [0008] 움직임 정보를 인코딩하는 비용을 추가로 감소시키기 위해, 움직임 벡터를 직접 인코딩하기보다는, 움직임이 균질(homogeneous)하다고 가정하여, 통상적으로 인코딩된 블록을 둘러싸고 있는 블록들의 하나 이상의 움직임 벡터들로부터 계산되는 움직임 벡터와 움직임 벡터 예측자 간의 차로 움직임 벡터가 인코딩될 수 있다.
- [0009] H.264에서, 예를 들어, 움직임 벡터들이 인코딩된 블록의 인과 이웃(causal neighbourhood)에 위치해 있는 움직임 벡터들로부터, 예를 들어, 인코딩된 블록의 위쪽에 그리고 좌측에 위치해 있는 3개의 블록들로부터, 계산되는 중간 예측자(median predictor)에 대해 인코딩된다. 인코딩 비용을 감소시키기 위해, 중간 예측자와 현재 블록 움직임 벡터 간의 차[잔차 움직임 벡터(residual motion vector)라고 함]만이 비트스트림에 인코딩된다.
- [0010] 잔차 움직임 벡터를 사용한 인코딩은 얼마간의 비트레이트를 절감하지만, 디코딩된 블록의 움직임 벡터의 값을 디코딩하기 위해 디코더가 움직임 벡터 예측자의 동일한 계산을 수행하는 것을 필요로 한다.
- [0011] 인코딩 프로세스 및 디코딩 프로세스 둘 다는 인코딩된 영상의 디코딩 프로세스를 포함할 수 있다. 이 프로세스는 통상적으로 장래 움직임 추정을 위해 인코더측에서 수행되고, 이는 인코더 및 대응하는 디코더가 동일한 참조 프레임을 가질 수 있게 해준다.
- [0012] 코딩된 프레임을 재구성하기 위해, 픽셀 영역에서 "디코딩된" 잔차를 제공하기 위해 잔차가 역양자화 및 역변환된다. 제1 재구성은 이어서 하나의 또는 몇개의 종류의 사후 필터링(post filtering) 프로세스에 의해 필터링된다. 동일한 참조 프레임이 인코더측 및 디코더측 둘 다에서 사용되도록, 이들 사후 필터(post filter)가 인코더측 및 디코더측에서 재구성된 프레임에 적용된다. 이 사후 필터링의 목표는 압축 아티팩트를 제거하는 것이다. 예를 들어, H.264/AVC는 디블록킹 필터(deblocking filter)를 사용한다. 이 필터는 잔차의 DCT 양자화 및 블록 움직임 보상으로 인한 블록킹 아티팩트(blocking artifact)를 제거할 수 있다. 현재의 HEVC 표준에서, 3가지 유형의 루프 필터 - 디블록킹 필터, SAO(sample adaptive offset, 샘플 적응적 오프셋), 및 ALF(adaptive loop filter, 적응적 루프 필터) - 가 사용되고 있다.
- [0013] 도 1은 공지된 HEVC 구현예의 루프 필터링 프로세스의 단계들을 나타낸 플로우차트이다. 초기 단계(101)에서, 인코더 또는 디코더는 전체 프레임의 재구성을 발생한다. 그 다음에, 단계(102)에서, 디블록킹된 재구성(103)을 발생시키기 위해 이 제1 재구성에 디블록킹 필터가 적용된다. 디블록킹 필터의 목표는 잔차 양자화 및 블록 움직임 보상 또는 블록 인트라 예측에 의해 발생된 블록 아티팩트를 제거하는 것이다. 이들 아티팩트는 낮은 비트레이트에서 시각적으로 중요하다. 디블록킹 필터는 2개의 이웃하는 블록들의 특성에 따라 블록 경계를 매끄럽게 하는 동작을 한다. 각각의 블록의 인코딩 모드, 잔차 코딩을 위해 사용되는 양자화 파라미터들, 및 경계에서의 이웃 픽셀 차가 고려된다. 모든 프레임들에 대해 동일한 기준/분류가 적용되고, 어떤 부가의 데이터도 전송되지 않는다. 디블록킹 필터는, 블록킹 아티팩트를 제거함으로써, 현재 프레임의 시각적 품질을 향상시키고, 또한 후속 프레임들에 대한 움직임 추정 및 움직임 보상을 향상시킨다. 실제로, 블록 아티팩트의 고주파수가 제거되고, 따라서 후속 프레임들의 텍스처 잔차(texture residual)에 대해 이들 고주파수가 보상될 필요가 없다.
- [0014] 디블록킹 필터 후에, 디블록킹된 재구성이 단계(104)에서 SAO(sample adaptive offset) 루프 필터에 의해 필터링된다. 얻어진 프레임(105)은 이어서 디스플레이되고 후속 인터 프레임들에 대한 참조 프레임으로서 사용될 재구성된 프레임(107)을 발생시키기 위해 단계(106)에서 ALF(adaptive loop filter)에 의해 필터링된다.

- [0015] SAO 루프 필터 및 ALF의 목표는 어떤 정보도 전송되지 않는 디블록킹 필터와 달리 부가의 데이터를 송신함으로써 프레임 재구성을 향상시키는 것이다.
- [0016] SAO 루프 필터의 원리는 각각의 픽셀을 클래스로 분류하고 클래스의 각각의 픽셀의 각각의 픽셀 값에 동일한 오프셋 값을 추가하는 것이다. 이와 같이, 각각의 클래스에 대해 하나의 오프셋이 전송된다. SAO 루프 필터링은 프레임 영역에 대한 2가지 종류의 분류 - 에지 오프셋(edge offset) 및 대역 오프셋(band offset) - 을 제공한다.
- [0017] 에지 오프셋 분류는 각각의 픽셀의 대응하는 픽셀 값을 2개의 이웃 픽셀의 픽셀 값과 비교함으로써 각각의 픽셀에 대한 클래스를 결정하는 것을 수반한다. 더욱이, 2개의 이웃 픽셀은 2개의 이웃 픽셀의 방향을 나타내는 파라미터에 의존한다. 이들 방향은 0도(수평 방향), 45도(대각 방향), 90도(수직 방향), 및 135도(제2 대각 방향)이다. 이하에서, 이들 방향은 에지 오프셋 분류의 "유형"이라고 불리운다.
- [0018] 제2 유형의 분류는 픽셀 값에 의존하는 대역 오프셋 분류이다. SAO 대역 오프셋에서의 클래스는 픽셀 값 범위에 대응한다. 이와 같이, 동일한 오프셋이 주어진 픽셀 값 범위 내의 픽셀 값을 가지는 모든 픽셀들에 가산된다.
- [0019] 프레임 콘텐츠에 보다 적응적이기 위해, SAO를 인코딩하는 데 쿼드-트리 구조에 기초하여 SAO 필터링을 적용하는 것이 제안되었다. 그 결과, 어떤 영역들만이 필터링되도록 쿼드 트리의 리프 노드(leaf node)에 대응하는 프레임 영역이 SAO에 의해 필터링되거나 그렇지 않을 수 있다. 더욱이, SAO가 인에이블될 때, 단지 하나의 SAO 분류 - 각각의 분류에 대해 전송되는 관련 파라미터들에 따라 에지 오프셋 또는 대역 오프셋 - 가 사용된다. 마지막으로, 각각의 SAO 리프 노드에 대해, SAO 분류는 물론 그의 파라미터 및 모든 클래스의 오프셋이 전송된다.
- [0020] 쿼드 트리의 주된 장점은 신호의 로컬 특성을 효율적으로 추종하는 것이다. 그렇지만, 이는 비트스트림에 전용 인코딩을 필요로 한다. SAO 파라미터들의 쿼드 트리 기반 인코딩을 LCU 레벨에서의 인코딩으로 대체하는 다른 해결책이 또한 예상될 수 있다.
- [0021] 인코딩될 비디오 데이터의 영상이 샘플 값들의 2차원 어레이(색상 채널이라고도 함) 세트로서 제공될 수 있고, 그의 각각의 항목은 루마 휘도(luma brightness) 및 중성 그레이스케일 색상(neutral grayscale colour)으로부터 청색 또는 적색 쪽으로의 크로마 색상 편차(chroma colour deviation)의 척도(YUV) 또는 적색, 녹색 또는 청색 광 성분 세기의 척도(RGB) 등의 색상 성분의 세기(intensity)를 나타낸다. YUV 모델은 하나의 루마(Y) 성분 및 2개의 크로미넌스(chrominance)(UV) 성분으로 색 공간을 정의한다. 일반적으로, Y는 루마 성분(휘도)을 나타내고, U 및 V는 크로미넌스(색상) 또는 크로마 성분이다.
- [0022] SAO 필터링이 통상적으로 루마 성분에 그리고 U 및 V 크로마 성분 둘 다에 독립적으로 적용된다.
- [0023] SAO 대역 오프셋의 공지된 구현에는 픽셀 값 범위를 도 2에 도시된 것과 동일한 크기의 사전 정의된 32개의 범위로 분할한다. 픽셀 값 범위의 최소값은 시스템적으로 0이고, 최대값은 다음과 같은 관계식  $\text{Max} = 2^{\text{Bitdepth}} - 1$ 에 따라 픽셀 값들의 비트 심도(bit-depth)에 의존한다. 예를 들어, 비트 심도가 8 비트일 때, 픽셀의 최대값은 255일 수 있다. 이와 같이, 픽셀 값 범위는 0과 255 사이이다. 이 8 비트의 비트 심도에 대해, 각각의 클래스는 16개 픽셀 값으로 된 범위를 포함한다. 더욱이, SAO 대역 오프셋에 대해, 2개의 클래스 그룹이 고려되고 있다. 제1 그룹은 도 2에 회색으로 나타난 바와 같이 픽셀 값 범위의 중앙에 있는 16개의 연속적인 클래스를 포함한다. 제2 그룹도 역시 16개의 클래스를 포함하지만, 이들 클래스가 도 2에 해치로 나타난 바와 같이 픽셀 값 범위의 양쪽 단부에 있다. 프레임 영역의 SAO 대역 오프셋에 대해, 분류를 위해 사용되는 그룹 및 16개의 오프셋이 비트스트림에 삽입된다.
- [0024] 도 3은 현재 프레임 영역(303)에 대해 인코더에서 오프셋을 선택하는 방법의 단계들을 나타낸 플로우차트이다. 프레임 영역은 N개의 픽셀을 포함하고 있다. 초기 단계(301)에서, 변수  $\text{Sum}_j$  및  $\text{SumNbPiX}_j$ 는 16개의 범위 각각에 대해 0의 값으로 설정된다. j는 현재 범위 또는 클래스 번호를 나타낸다.  $\text{Sum}_j$ 는 범위 j에서의 픽셀들의 값과 그의 대응하는 원래의 픽셀들의 값 사이의 차의 합이다.  $\text{SumNbPiX}_j$ 는 범위 j 내의 픽셀들의 수를 나타낸다.
- [0025] 단계(302)에서, 카운터 변수 i는 값 0으로 설정된다. 그 다음에, 단계(304)에서 프레임 영역(303)의 제1 픽셀이 추출된다. 처리되고 있는 현재 SAO 그룹(도 2에 나타난 바와 같이 제1 또는 제2)이 알려져 있는 것으로 가

정된다. 단계(305)에서 픽셀 값  $P_i$ 가 현재 SAO 그룹에 있지 않은 것으로 판정되는 경우, 프레임 영역(303)의 후속 픽셀들을 분류하기 위해 단계(308)에서 카운터 변수  $i$ 의 값이 증가된다. 그렇지 않고 단계(305)에서 픽셀 값  $P_i$ 가 현재 SAO 그룹에 있는 것으로 판정되는 경우, 단계(306)에서  $P_i$ 의 값에 대응하는 범위 번호(또는 클래스 번호)  $j$ 가 발견된다. 후속 단계(307)에서, 대응하는  $\text{SumNbPix}_j$  변수가 증가되고,  $P_i$ 와 그의 원래의 값  $P_i^{org}$  사이의 차가  $\text{Sum}_j$ 에 가산된다. 그 다음 단계에서, 프레임 영역(303)의 다른 픽셀들에 분류를 적용하기 위해 카운터 변수  $i$ 가 증가된다. 단계(309)에서, 프레임 영역(303)의  $N$ 개의 픽셀들 모두가 분류되었는지(즉,  $i \geq N$ 인지) 여부가 판정되고, 예인 경우, 오프셋 선택 알고리즘의 최종 결과로서 각각의 클래스  $j$ 에 대한 오프셋을 제공하는 오프셋 테이블(311)을 생성하기 위해 단계(310)에서 각각의 클래스에 대한  $\text{Offset}_j$ 가 계산된다. 이 오프셋은 클래스  $j$ 의 픽셀의 픽셀 값과 그 각자의 원래의 픽셀 값 사이의 차의 평균으로서 계산된다. 클래스  $j$ 에 대한  $\text{Offset}_j$ 는 이하의 식에 의해 주어진다:

### 수학식 1

$$\text{Offset}_j = \frac{\text{Sum}_j}{\text{SumNbPix}_j}$$

[0026]

[0027]

도 4는 SAO 대역 오프셋을 대응하는 클래스 그룹에 적용하는 디코딩 프로세스의 단계들을 나타낸 플로우차트이다. 후속 프레임의 움직임 추정 및 보상을 위해 사용되는 참조 프레임을 생성하기 위해 이 프로세스가 또한 인코더측에서 적용될 수 있다.

[0028]

프로세스의 초기 단계(401)는 오프셋 테이블(402)을 생성하기 위해 각각의 클래스의 픽셀 값들에 대한 오프셋 값들을 디코딩하는 것을 수반한다. 인코더측에서, 오프셋 테이블(402)은 도 3에 도시된 선택 알고리즘의 결과이다. 이와 같이, 인코더측에서, 단계(401)는 도 3의 오프셋 선택 알고리즘으로 대체된다.

[0029]

단계(403)에서, 카운터 변수  $i$ 가 0으로 설정된다. 단계(405)에서, 픽셀  $P_i$ 가  $N$ 개의 픽셀을 포함하는 프레임 영역(404)으로부터 추출된다. 단계(406)에서, 픽셀  $P_i$ 가 현재 클래스 그룹에 속하는지가 판정된다. 픽셀  $P_i$ 가 현재 클래스 그룹에 속하는 것으로 판정되는 경우, 관련 클래스 번호  $j$ 가 식별되고, 단계(409)에서, 관련 오프셋 값  $\text{Offset}_j$ 가 오프셋 테이블(402)로부터 추출된다. 단계(411)에서 필터링된 픽셀 값  $P'_i$ 를 생성하기 위해, 단계(410)에서, 추출된 오프셋 값  $\text{Offset}_j$ 가 이어서  $P_i$ 의 픽셀 값에 가산된다. 단계(412)에서, 대응하는 픽셀에서, 필터링된 픽셀 값이 이어서 필터링된 프레임 영역(415)에 삽입된다.

[0030]

단계(406)에서, 픽셀  $P_i$ 가 SAO 대역 오프셋 그룹에 있지 않은 경우, 단계(412)에서 필터링 없이  $P_i$ 의 픽셀 값이 필터링된 프레임 영역(415)에 넣어진다. 단계(412) 후에, 필요한 경우, 현재 프레임 영역(404)의 후속 픽셀들을 필터링하기 위해 카운터 변수  $i$ 가 증가된다. 단계(414)에서, 프레임 영역의  $N$ 개의 픽셀 전부가 처리된 것으로(즉,  $i \geq N$ 인 것으로) 판정된 후에, 필터링된 프레임 영역(415)이 재구성되고, SAO 재구성된 프레임[도 1의 프레임(105) 참조]에 추가될 수 있다.

[0031]

공지된 보상 선택 프로세스의 단점은 상이한 영상 픽셀 콘텐츠 변동들에 그리고 상이한 유형의 영상 픽셀 성분 에 적응되지 않는다는 것이다.

### 발명의 내용

[0032]

이상의 문제점들 중 하나 이상을 해결하기 위해 본 발명이 안출되었다.

[0033]

본 발명의 제1 측면에 따르면, 영상의 재구성된 샘플 세트에 대한 보상 오프셋들을 제공하는 방법이 제공되고, 각각의 샘플은 샘플 값을 가지며, 이 방법은,

[0034]

레이트 왜곡 기준(rate distortion criterion)에 기초하여, 복수의 사전 결정된 분류들 중에서 분류를 선택하는 단계 - 각각의 상기 사전 결정된 분류는 샘플 값의 전체 범위보다 작은 분류 범위(classification range)를 가지며 복수의 클래스들로 이루어져 있고, 각각의 클래스는 상기 분류 범위 내에서의 샘플 값의 범위를 정의하고,

샘플의 샘플 값이 관련 클래스의 범위 내에 있는 경우 샘플이 그 클래스 내에 배치됨 -; 및

- [0035] 관련 클래스의 각각의 샘플의 샘플 값에 적용하기 위한 보상 오프셋을 선택된 분류의 각각의 클래스와 연관시키는 단계를 포함한다.
- [0036] 본 발명의 제2 측면에 따르면, 복수의 샘플들로 이루어져 있는 영상을 인코딩하는 방법이 제공되고, 이 방법은,
- [0037] 샘플들을 인코딩하는 단계;
- [0038] 재구성된 샘플들을 제공하기 위해 인코딩된 샘플들을 디코딩하는 단계;
- [0039] 재구성된 샘플들에 대해 루프 필터링을 수행하는 단계 - 루프 필터링은 각자의 재구성된 샘플들의 샘플 값들에 보상 오프셋들을 적용하는 것을 포함하고, 각각의 보상 오프셋은 샘플 값의 범위와 연관되어 있으며, 보상 오프셋들은 본 발명의 전술한 제1 측면을 구현하는 방법에 따라 제공됨 -; 및
- [0040] 인코딩된 샘플들의 비트스트림을 발생시키는 단계를 포함한다.
- [0041] 본 발명의 제3 측면에 따르면, 복수의 샘플들로 이루어져 있는 영상을 인코딩하는 방법이 제공되고, 각각의 샘플은 샘플 값을 가지며, 이 방법은,
- [0042] 인코딩된 샘플 값들을 수신하는 단계;
- [0043] 인코딩된 분류 데이터를 수신하는 단계;
- [0044] 인코딩된 보상 오프셋들을 수신하는 단계;
- [0045] 분류 데이터를 디코딩하고, 디코딩된 분류 데이터에 기초하여, 복수의 사전 결정된 분류들 중에서 분류를 선택하는 단계 - 각각의 상기 사전 결정된 분류는 샘플 값의 전체 범위보다 작은 분류 범위(classification range)를 가지며 복수의 클래스들로 이루어져 있고, 각각의 클래스는 상기 분류 범위 내에서의 샘플 값의 범위를 정의하고, 샘플의 샘플 값이 관련 클래스의 범위 내에 있는 경우 샘플이 그 클래스 내에 배치됨 -;
- [0046] 재구성된 샘플 값들을 제공하기 위해 인코딩된 샘플들을 디코딩하고 인코딩된 보상 오프셋들을 디코딩하는 단계;
- [0047] 디코딩된 보상 오프셋들을 선택된 분류의 클래스들과, 각각, 연관시키는 단계; 및
- [0048] 재구성된 샘플 값들에 대해 루프 필터링을 수행하는 단계 - 루프 필터링은 선택된 분류의 각각의 클래스와 연관되어 있는 디코딩된 보상 오프셋을 관련 클래스의 범위 내의 재구성된 샘플 값들에 적용하는 것을 포함함 - 를 포함한다.
- [0049] 본 발명의 제4 측면에 따르면, 비디오 비트스트림에 의해 표현되는 영상에 대한 정보 데이터 세트(information dataset)를 전달하는 신호가 제공되고, 영상은 재구성가능 샘플 세트를 포함하며, 각각의 재구성가능 샘플은 샘플 값을 갖고, 정보 데이터 세트는 복수의 사전 결정된 분류들 중에서 인코더에 의해 선택된 분류에 관한 분류 데이터를 포함하며, 각각의 상기 사전 결정된 분류는 샘플 값의 전체 범위보다 작은 분류 범위를 갖고 복수의 클래스들로 이루어져 있으며, 각각의 클래스는 상기 분류 범위 내의 샘플 값의 범위를 정의하고, 샘플의 샘플 값이 관련 클래스의 범위 내에 있는 경우 샘플이 그 클래스 내에 배치되며, 선택된 분류의 복수의 클래스들의 각각의 클래스는 관련 클래스의 범위 내의 재구성가능 샘플들의 샘플 값들에 적용하기 위한 보상 오프셋과 연관되어 있다.
- [0050] 본 발명의 제5 측면에 따르면, 영상의 재구성된 샘플 세트에 대한 보상 오프셋들을 제공하는 장치가 제공되고, 각각의 샘플은 샘플 값을 가지며, 이 장치는,
- [0051] 레이트 왜곡 기준에 기초하여, 복수의 사전 결정된 분류들 중에서 분류를 선택하는 수단 - 각각의 상기 사전 결정된 분류는 샘플 값의 전체 범위보다 작은 분류 범위를 가지며 복수의 클래스들로 이루어져 있고, 각각의 클래스는 상기 분류 범위 내에서의 샘플 값의 범위를 정의하고, 샘플 값이 관련 클래스의 범위 내에 있는 경우 샘플이 그 클래스 내에 배치됨 -; 및
- [0052] 관련 클래스의 각각의 샘플의 샘플 값에 적용하기 위한 보상 오프셋을 선택된 분류의 각각의 클래스와 연관시키는 수단을 포함한다.
- [0053] 본 발명의 제6 측면에 따르면, 복수의 샘플들로 이루어져 있는 영상을 인코딩하는 인코딩 장치가 제공되고, 이 장치는,

- [0054] 샘플들을 인코딩하는 인코더;
- [0055] 재구성된 샘플들을 제공하기 위해 인코딩된 샘플들을 디코딩하는 디코더;
- [0056] 재구성된 샘플들을 필터링하는 루프 필터 - 루프 필터링 수단은 각자의 재구성된 샘플들의 샘플 값들에 보상 오프셋들을 적용하는 오프셋 적용 수단을 포함하고, 각각의 보상 오프셋은 샘플 값의 범위와 연관되어 있으며, 보상 오프셋들은 본 발명의 전술한 제5 측면을 구현하는 장치에 의해 제공됨 -; 및
- [0057] 인코딩된 샘플들의 비트스트림을 발생시키는 비트스트림 발생기를 포함한다.
- [0058] 본 발명의 제7 측면에 따르면, 복수의 샘플들로 이루어져 있는 영상을 디코딩하는 장치가 제공되고, 각각의 샘플은 샘플 값을 가지며, 이 장치는,
- [0059] 인코딩된 샘플 값들을 수신하는 수단;
- [0060] 인코딩된 분류 데이터를 수신하는 수단;
- [0061] 인코딩된 보상 오프셋들을 수신하는 수단;
- [0062] 분류 데이터를 디코딩하고, 디코딩된 분류 데이터에 기초하여, 복수의 사전 결정된 분류들 중에서 분류를 선택하는 수단 - 각각의 상기 사전 결정된 분류는 샘플 값의 전체 범위보다 작은 분류 범위를 가지며 복수의 클래스들로 이루어져 있고, 각각의 클래스는 상기 분류 범위 내에서의 샘플 값의 범위를 정의하고, 샘플의 샘플 값이 관련 클래스의 범위 내에 있는 경우 샘플이 그 클래스 내에 배치됨 -;
- [0063] 재구성된 샘플 값들을 제공하기 위해 인코딩된 샘플들을 디코딩하고 인코딩된 보상 오프셋들을 디코딩하는 수단;
- [0064] 디코딩된 보상 오프셋들을 선택된 분류의 클래스들과, 각각, 연관시키는 수단; 및
- [0065] 재구성된 샘플 값들에 대해 루프 필터링을 수행하는 수단 - 루프 필터링은 선택된 분류의 각각의 클래스와 연관되어 있는 디코딩된 보상 오프셋을 관련 클래스의 범위 내의 재구성된 샘플 값들에 적용하는 것을 포함함 - 을 포함한다.
- [0066] 본 발명의 제8 측면은 복수의 샘플들로 이루어져 있는 영상을 디코딩하는 방법을 제공하고, 각각의 샘플은 샘플 값을 가지며, 이 방법은,
- [0067] 인코딩된 샘플 값들을 수신하는 단계;
- [0068] 복수의 사전 결정된 분류들 중에서 인코더에 의해 선택된 분류에 관한 인코딩된 분류 데이터를 수신하는 단계 - 각각의 상기 사전 결정된 분류는 샘플 값의 전체 범위보다 작은 분류 범위(classification range)를 가지며 복수의 클래스들로 이루어져 있고, 각각의 클래스는 상기 분류 범위 내에서의 샘플 값의 범위를 정의하고, 샘플 값이 관련 클래스의 범위 내에 있는 경우 샘플이 그 클래스 내에 배치됨 -;
- [0069] 선택된 분류의 클래스들과, 각각, 연관되어 있는 인코딩된 보상 오프셋들을 수신하는 단계;
- [0070] 재구성된 샘플 값들을 제공하기 위해 인코딩된 샘플 값들을 디코딩하고 인코딩된 분류 데이터 및 보상 오프셋들을 디코딩하는 단계; 및
- [0071] 재구성된 샘플 값들에 대해 루프 필터링을 수행하는 단계 - 루프 필터링은 선택된 분류의 각각의 클래스와 연관되어 있는 디코딩된 보상 오프셋을 관련 클래스의 범위 내의 재구성된 샘플 값들에 적용하는 것을 포함함 - 를 포함한다.
- [0072] 인코더는 레이트 왜곡 기준에 기초하는 것 또는 샘플 값들의 통계적 분포의 특성들에 의존하는 것을 비롯하여 임의의 적당한 방식으로 분류를 선택할 수 있다.
- [0073] 본 발명과 관련하여, 샘플은 단일 픽셀에 대응할 수 있고, 샘플 값은 각자의 픽셀 값에 대응할 수 있다. 다른 대안으로서, 샘플은 복수의 픽셀들을 포함할 수 있고, 샘플 값은 복수의 픽셀들의 픽셀 값들로부터 결정되는 픽셀 값에 대응할 수 있다.
- [0074] 본 발명에 따른 방법들 중 적어도 일부는 컴퓨터로 구현될 수 있다. 그에 따라, 본 발명은 하드웨어로만 된 실시예, 소프트웨어로만 된 실시예(펌웨어, 상주 소프트웨어, 마이크로 코드 등을 포함함), 또는 소프트웨어와 하드웨어 측면을 겸비한 실시예의 형태를 취할 수 있고, 이들 모두가 일반적으로 본 명세서에서 "회로", "모듈",

또는 "시스템"으로 지칭되어 있을 수 있다. 게다가, 본 발명은 컴퓨터 사용가능 프로그램 코드가 실시되어 있는 임의의 유형의(tangible) 표현 매체로 구현되어 있는 컴퓨터 프로그램 제품의 형태를 취할 수 있다.

[0075] 본 발명이 소프트웨어로 구현될 수 있기 때문에, 본 발명은 프로그램가능 장치 또는 임의의 적당한 전달 매체(carrier medium)에 프로비저닝하기 위한 컴퓨터 판독가능 코드로 실시될 수 있다. 유형의 전달 매체는 플로피 디스크, CD-ROM, 하드 디스크 드라이브, 자기 테이프 장치 또는 고상 메모리 장치 등과 같은 저장 매체를 포함할 수 있다. 일시적 전달 매체(transient carrier medium)는 전기 신호, 전자 신호, 광학 신호, 음향 신호, 자기 신호, 또는 전자기 신호(예컨대, 마이크로파 또는 RF 신호) 등의 신호를 포함할 수 있다.

[0076] 이와 같이, 본 발명의 제9 측면에 따르면, 프로그램가능 장치에 대한 컴퓨터 프로그램 제품이 제공되고, 컴퓨터 프로그램 제품은 프로그램가능 장치에 로드되어 실행될 때 전술한 제1 측면, 제2 측면, 제3 측면, 및 제8 측면 중 임의의 측면을 실시하는 방법을 구현하는 명령어 시퀀스를 포함한다.

[0077] 본 발명의 제10 측면에 따르면, 전술한 제1 측면, 제2 측면, 제3 측면 및 제8 측면 중 임의의 측면을 실시하는 방법을 구현하는 컴퓨터 프로그램의 명령어들을 저장하는 컴퓨터 판독가능 저장 매체가 제공된다.

### 도면의 간단한 설명

[0078] 이하의 도면들을 참조하여, 단지 예로서, 본 발명의 실시예들이 이제부터 기술될 것이다.

도 1은 종래 기술의 루프 필터링 프로세스의 단계들을 나타낸 플로우차트.

도 2는 종래 기술의 HEVC 프로세스의 샘플 적응적 대역 오프셋 분류를 그래픽으로 나타낸 도면.

도 3은 HEVC의 SAO 대역 오프셋에 대한 보상 오프셋을 결정하는 프로세스의 단계들을 나타낸 플로우차트.

도 4는 HEVC의 SAO 대역 오프셋 필터링 프로세스의 단계들을 나타낸 플로우차트.

도 5는 본 발명의 하나 이상의 실시예들이 구현될 수 있는 데이터 통신 시스템을 개략적으로 나타낸 블록도.

도 6은 본 발명의 하나 이상의 실시예들이 구현될 수 있는 처리 장치의 구성요소들을 나타낸 블록도.

도 7은 본 발명의 실시예들에 따른 인코딩 방법의 단계들을 나타낸 플로우차트.

도 8은 본 발명의 하나 이상의 실시예들에 따른 루프 필터링 프로세스의 단계들을 나타낸 플로우차트.

도 9는 본 발명의 실시예들에 따른 디코딩 방법의 단계들을 나타낸 플로우차트.

도 10은 본 발명의 제1 실시예에 따른, SAO 대역 오프셋 분류를 결정하는 방법의 단계들을 나타낸 플로우차트.

도 11은 본 발명의 일 실시예에 따른, 적응된 분류(adapted classification)를 결정하는 방법의 단계들을 나타낸 플로우차트.

도 12는 본 발명의 대안의 실시예에 따른, 적응된 분류를 결정하는 방법의 단계들을 나타낸 플로우차트.

도 13은 본 발명의 일 실시예에 따른, 분류를 위한 유효 범위의 몇개의 크기들을 나타낸 도면.

도 14는 본 발명의 일 실시예에 따른, 분류를 위한 클래스들의 몇개의 크기들을 나타낸 도면.

도 15는 본 발명의 일 실시예에 따른, 분류를 위한 유효 범위에서의 클래스들의 몇개의 크기들을 나타낸 도면.

도 16은 본 발명의 일 실시예에 따른, 분류를 위한 제1 그룹의 유효 범위의 몇개의 중앙 위치들을 나타낸 도면.

도 17은 본 발명의 일 실시예에 따른, 분류를 위한 제2 그룹의 유효 범위의 몇개의 중앙 위치들을 나타낸 도면.

도 18은 본 발명의 일 실시예에 따른, 파라미터 분류의 레이트 왜곡 선택(rate distortion selection)을 나타낸 도면.

도 19a 및 도 19b는 본 발명의 다른 실시예에 따른, 전체 범위 내에서의 유효 범위의 가능한 위치들을 나타낸 도면.

도 20a는 LCU 레벨에서 SAO 파라미터들을 인코딩하는 종래 기술에서 적용되는 의사 코드를 나타낸 도면.

도 20b는 LCU 레벨에서 SAO 파라미터들을 인코딩하는 본 발명의 일 실시예에 따른 개선된 의사 코드를 나타낸 도면.

도 21은 도 20a의 의사 코드에 대응하는 플로우차트.

도 22는 도 20b의 의사 코드에 대응하는 플로우차트.

도 23은 본 발명의 추가의 실시예에 따른, SAO 파라미터들의 인코딩을 설명하는 데 사용하기 위한 플로우차트.

도 24는 본 발명의 또 다른 실시예에 따른, SAO 파라미터들을 인코딩하는 데 사용되는 의사 코드를 나타낸 도면.

도 25는 도 24의 의사 코드에 대응하는 플로우차트.

도 26은 본 발명의 다른 추가의 실시예에 따른, SAO 파라미터들의 인코딩을 설명하는 데 사용하기 위한 플로우차트.

### 발명을 실시하기 위한 구체적인 내용

- [0079] 도 5는 본 발명의 하나 이상의 실시예들이 구현될 수 있는 데이터 통신 시스템을 나타낸 것이다. 데이터 통신 시스템은 데이터 통신 네트워크(500)를 통해 데이터 스트림의 데이터 패킷들을 수신 장치[이 경우에, 클라이언트 단말(502)]로 전송하는 동작을 하는 전송 장치[이 경우에, 서버(501)]를 포함하고 있다. 데이터 통신 네트워크(500)는 WAN(Wide Area Network) 또는 LAN(Wide Area Network)일 수 있다. 이러한 네트워크는, 예를 들어, 무선 네트워크(WiFi / 802.11a 또는 b 또는 g), 이더넷 네트워크, 인터넷 네트워크, 또는 몇 개의 상이한 네트워크들로 이루어진 혼합 네트워크일 수 있다. 본 발명의 특정의 실시예에서, 데이터 통신 시스템은 서버(501)가 동일한 데이터 콘텐츠를 다수의 클라이언트들로 송신하는 디지털 텔레비전 방송 시스템일 수 있다.
- [0080] 서버(501)에 의해 제공되는 데이터 스트림(504)은 비디오 및 오디오 데이터를 나타내는 멀티미디어 데이터로 이루어져 있을 수 있다. 본 발명의 어떤 실시예들에서, 오디오 및 비디오 데이터 스트림은, 각각, 마이크 및 카메라를 사용하여 서버(501)에 의해 포착될 수 있다. 어떤 실시예들에서, 데이터 스트림은 다른 데이터 제공자로부터 서버(501)에 저장되거나 서버(501)에 의해 수신될 수 있거나, 또는 서버(501)에서 발생될 수 있다. 서버(501)는 특히 인코더에 입력으로서 제시되는 데이터의 보다 콤팩트한 표현인 전송을 위한 압축된 비트스트림을 제공하기 위해 비디오 및 오디오 스트림을 인코딩하는 인코더를 구비하고 있다.
- [0081] 전송된 데이터의 품질 대 전송된 데이터의 양에 대한 보다 나은 비를 획득하기 위해, 비디오 데이터의 압축은, 예를 들어, HEVC 형식 또는 H.264/AVC 형식을 따를 수 있다.
- [0082] 클라이언트(502)는 전송된 비트스트림을 수신하고, 디스플레이 장치 상에서 비디오 영상을 그리고 스피커에 의해 오디오 데이터를 재생하기 위해 재구성된 비트스트림을 디코딩한다.
- [0083] 도 5의 예에서 스트리밍 시나리오가 고려되고 있지만, 본 발명의 어떤 실시예들에서, 인코더와 디코더 사이의 데이터 통신이, 예를 들어, 광 디스크 등의 미디어 저장 장치를 사용하여 수행될 수 있다는 것을 잘 알 것이다.
- [0084] 본 발명의 하나 이상의 실시예들에서, 비디오 영상이 최종 영상에 필터링된 픽셀들을 제공하기 위해 영상의 재구성된 픽셀들에 적용하기 위한 보상 오프셋들을 나타내는 데이터와 함께 전송된다.
- [0085] 도 6은 본 발명의 적어도 하나의 실시예를 구현하도록 구성되어 있는 처리 장치(600)를 개략적으로 나타낸 것이다. 처리 장치(600)는 마이크로컴퓨터, 워크스테이션, 또는 경량 휴대용 장치 등의 장치일 수 있다. 장치(600)는 다음과 같은 것들에 연결되어 있는 통신 버스(613)를 포함하고 있다:
- [0086] - CPU로 표시된 마이크로프로세서 등의 중앙 처리 유닛(611);
- [0087] - 본 발명을 구현하는 컴퓨터 프로그램들을 저장하는, ROM으로 표시된 판독 전용 메모리(607);
- [0088] - 본 발명의 실시예들의 방법의 실행가능 코드를 저장하는 RAM으로 표시된 랜덤 액세스 메모리(612)는 물론, 본 발명의 실시예들에 따른 디지털 영상 시퀀스를 인코딩하는 방법 및/또는 비트스트림을 디코딩하는 방법을 구현하는 데 필요한 변수들 및 파라미터들을 기록하도록 구성되어 있는 레지스터들; 및
- [0089] - 통신 네트워크(603)(처리될 디지털 데이터가 이를 통해 전송되거나 수신됨)에 연결되어 있는 통신 인터페이스(602).
- [0090] 선택적으로, 장치(600)는 또한 다음과 같은 구성요소들을 포함할 수 있다:
- [0091] - 본 발명의 하나 이상의 실시예들의 방법들을 구현하는 컴퓨터 프로그램들 및 본 발명의 하나 이상의 실시예들

의 구현 동안 사용되거나 생성되는 데이터를 저장하는, 하드 디스크 등의 데이터 저장 수단(604);

- [0092] - 디스크(606)에 대한 디스크 드라이브(605)[디스크 드라이브는 디스크(606)로부터 데이터를 판독하거나 상기 디스크에 데이터를 기입하도록 구성되어 있음];
- [0093] - 데이터를 디스플레이하는 및/또는 키보드(610) 또는 임의의 다른 포인팅 수단에 의해 사용자와의 그래픽 인터페이스로서 역할하는 화면(609).
- [0094] 장치(600)는, 예를 들어, 디지털 카메라(620) 또는 마이크(608) 등의 다양한 주변 장치들에 연결될 수 있고, 각각은 멀티미디어 데이터를 장치(600)에 제공하기 위해 입출력 카드(도시 생략)에 연결되어 있다.
- [0095] 통신 버스는 장치(600)에 포함되어 있거나 그에 연결되어 있는 다양한 요소들 간의 통신 및 상호운용성을 제공한다. 버스의 표현은 제한하는 것이 아니며, 상세하게는, 중앙 처리 유닛은 명령어들을 직접 또는 장치(600)의 다른 요소에 의해 장치(600)의 임의의 요소로 전달하는 동작을 한다.
- [0096] 디스크(606)는, 예를 들어, 콤팩트 디스크(CD-ROM)(재기입가능하거나 그렇지 않음), ZIP 디스크 또는 메모리 카드 등의 임의의 정보 매체로, 그리고, 일반적인 표현으로, 마이크로컴퓨터에 의해 또는 마이크로프로세서에 의해 판독될 수 있고, 장치 내에 통합되어 있거나 그렇지 않으며, 어쩌면 이동식이고 하나 이상의 프로그램들 - 그의 실행은 구현될 본 발명에 따른 디지털 영상 시퀀스를 인코딩하는 방법 및/또는 비트스트림을 디코딩하는 방법을 가능하게 함 - 을 저장하도록 구성되어 있는 정보 저장 수단으로 대체될 수 있다.
- [0097] 실행가능 코드는 판독 전용 메모리(607)에, 하드 디스크(604)에, 또는 이동식 디지털 매체[예를 들어, 앞서 기술한 바와 같은 디스크(606) 등]에 저장될 수 있다. 한 변형예에 따르면, 프로그램들의 실행가능 코드는, 실행되기 전에 하드 디스크(604) 등의 장치(600)의 저장 수단들 중 하나에 저장되기 위해, 인터페이스(602)를 통해 통신 네트워크(603)에 의해 수신될 수 있다.
- [0098] 중앙 처리 유닛(611)은 본 발명에 따른 프로그램 또는 프로그램들의 소프트웨어 코드의 명령어들 또는 일부분들, 상기한 저장 수단들 중 하나에 저장되어 있는 명령어들의 실행을 제어 및 지시하도록 구성되어 있다. 전원을 켜 시에, 비휘발성 메모리에[예를 들어, 하드 디스크(604) 상에 또는 판독 전용 메모리(607)에] 저장되어 있는 프로그램 또는 프로그램들은 랜덤 액세스 메모리(612) 내로 전송되는 것(그러면, 랜덤 액세스 메모리(612)는 프로그램 또는 프로그램들의 실행가능 코드를 포함하고 있음)은 물론, 본 발명을 구현하는 데 필요한 변수들 및 파라미터들을 저장하는 레지스터들로도 전송된다.
- [0099] 이 실시예에서, 장치는 본 발명을 구현하기 위해 소프트웨어를 사용하는 프로그램가능 장치이다. 그렇지만, 다른 대안으로서, 본 발명은 하드웨어로[예를 들어, ASIC(Application Specific Integrated Circuit)의 형태로] 구현될 수 있다.
- [0100] 도 7은 본 발명의 적어도 하나의 실시예에 따른 인코더의 블록도를 나타낸 것이다. 인코더가 연결된 모듈들로 나타내어져 있고, 각각의 모듈은, 예를 들어, 장치(600)의 CPU(611)에 의해 실행될 프로그래밍 명령어들의 형태로, 본 발명의 하나 이상의 실시예들에 따른, 영상 시퀀스의 영상을 인코딩하는 적어도 하나의 실시예를 구현하는 방법의 적어도 하나의 대응하는 단계를 구현하도록 구성되어 있다.
- [0101] 원래의 디지털 영상 시퀀스  $i_0$  내지  $i_n$ (701)이 인코더(70)에 의해 입력으로서 수신된다. 각각의 디지털 영상은 픽셀이라고 하는 샘플 세트로 표현된다.
- [0102] 인코딩 프로세스의 구현 후에 비트스트림(710)이 인코더(70)에 의해 출력된다. 비트스트림(710)은 복수의 인코딩 단위들 또는 슬라이스들을 포함하고, 각각의 슬라이스는 슬라이스를 인코딩하는 데 사용되는 인코딩 파라미터들의 인코딩 값들을 전송하는 슬라이스 헤더(slice header), 및 인코딩된 비디오 데이터를 포함하는 슬라이스 보디(slice body)를 포함하고 있다.
- [0103] 입력 디지털 영상들  $i_0$  내지  $i_n$ (701)은 모듈(702)에 의해 픽셀 블록들로 나누어진다. 블록들은 영상 일부분들에 대응하고, 가변 크기(예컨대, 4x4, 8x8, 16x16, 32x32, 64x64 픽셀)를 가질 수 있다. 각각의 입력 블록에 대해 코딩 모드가 선택된다. 2가지 계열의 코딩 모드 - 공간 예측 코딩에 기초한 코딩 모드(인트라 예측), 및 시간 예측에 기초한 코딩 모드[인터 코딩, 병합, 건너뛰기] - 가 제공된다. 가능한 코딩 모드들이 테스트된다.
- [0104] 모듈(703)은 인트라 예측 프로세스를 구현하고, 여기서 인코딩될 주어진 블록이 인코딩될 상기 블록의 이웃의 픽셀들로부터 계산되는 예측자에 의해 예측된다. 인트라 코딩이 선택되는 경우 잔차를 제공하기 위해, 선택된 인트라 예측자의 표시 및 주어진 블록과 그의 예측자 간의 차가 인코딩된다.

- [0105] 시간 예측은 움직임 추정 모듈(704) 및 움직임 보상 모듈(705)에 의해 구현된다. 먼저, 참조 영상 세트(716) 중에서 참조 영상이 선택되고, 인코딩될 주어진 블록에 가장 가까운 영역인 참조 영상의 일부분(참조 영역 또는 영상 부분이라고도 불리움)이 움직임 추정 모듈(704)에 의해 선택된다. 움직임 보상 모듈(705)은 이어서 선택된 영역을 사용하여 인코딩될 블록을 예측한다. 선택된 참조 영역과 주어진 블록 간의 차(잔차 블록이라고도 함)가 움직임 보상 모듈(705)에 의해 계산된다. 선택된 참조 영역은 움직임 벡터로 표시된다.
- [0106] 이와 같이, 이들 경우(공간 예측 및 시간 예측) 둘 다에서, 잔차는 원래의 블록으로부터 예측을 차감함으로써 계산된다.
- [0107] 모듈(703)에 의해 구현되는 인트라 예측에서, 예측 방향이 인코딩된다. 시간 예측에서, 적어도 하나의 움직임 벡터가 인코딩된다.
- [0108] 인터 예측이 선택되는 경우, 움직임 벡터에 대한 정보 및 잔차 블록이 인코딩된다. 비트레이트를 추가로 감소시키기 위해, 움직임이 균질하다고 가정하여, 움직임 벡터가 움직임 벡터 예측자에 대한 차에 의해 인코딩된다. 움직임 정보 예측자 세트의 움직임 벡터 예측자들이 움직임 벡터 예측 및 코딩 모듈(717)에 의해 움직임 벡터 필드(718)로부터 획득된다.
- [0109] 인코더(70)는 레이트 왜곡 기준(rate-distortion criterion) 등의 인코딩 비용 기준을 적용함으로써 코딩 모드를 선택하는 선택 모듈(706)을 추가로 포함하고 있다. 중복성을 추가로 감소시키기 위해, 변환 모듈(707)에 의해 변환(DCT 등)이 잔차 블록에 적용되고, 획득되는 변환된 데이터는 이어서 양자화 모듈(708)에 의해 양자화되고, 엔트로피 인코딩 모듈(709)에 의해 엔트로피 인코딩된다. 마지막으로, 인코딩되고 있는 현재 블록의 인코딩된 잔차 블록이 비트스트림(710)에 삽입된다.
- [0110] 인코더(70)는 또한, 후속 영상의 움직임 추정을 위한 참조 영상을 생성하기 위해, 인코딩된 영상의 디코딩을 수행한다. 이것은 비트스트림을 수신하는 인코더 및 디코더가 동일한 참조 프레임을 가질 수 있게 해준다. 역양자화 모듈(711)은 양자화된 데이터의 역양자화를 수행하고, 이어서 역변환 모듈(712)에 의한 역변환이 있게 된다. 역 인트라 예측(reverse intra prediction) 모듈(713)은 주어진 블록에 대해 어느 예측자를 사용할지를 결정하기 위해 예측 정보를 사용하고, 역 움직임 보상(reverse motion compensation) 모듈(714)은 실제로 모듈(712)에 의해 획득된 잔차를 참조 영상 세트(716)로부터 획득된 참조 영역에 가산한다.
- [0111] 재구성된 픽셀 프레임을 필터링하기 위해 사후 필터링이 이어서 모듈(715)에 의해 적용된다. 본 발명의 실시예들에서, 보상 오프셋이 재구성된 영상의 재구성된 픽셀의 픽셀 값에 가산되는 SAO 루프 필터가 사용된다.
- [0112] 도 8은 본 발명의 적어도 하나의 실시예에 따른 루프 필터링 프로세스의 단계들을 나타낸 플로우차트이다. 초기 단계(801)에서, 인코더는 전체 프레임의 재구성을 발생한다. 그 다음에, 단계(802)에서, 디블록킹된 재구성을 발생(803)하기 위해 이 제1 재구성에 디블록킹 필터가 적용된다. 디블록킹 필터의 목표는 잔차 양자화 및 블록 움직임 보상 또는 블록 인트라 예측에 의해 발생된 블록 아티팩트를 제거하는 것이다. 이들 아티팩트는 낮은 비트레이트에서 시각적으로 중요하다. 디블록킹 필터는 2개의 이웃하는 블록들의 특성에 따라 블록 경계를 매끄럽게 하는 동작을 한다. 각각의 블록의 인코딩 모드, 잔차 코딩을 위해 사용되는 양자화 파라미터들, 및 경계에서의 이웃 픽셀 차가 고려된다. 모든 프레임들에 대해 동일한 기준/분류가 적용되고, 어떤 부가의 데이터도 전송되지 않는다. 디블록킹 필터는, 블로킹 아티팩트를 제거함으로써, 현재 프레임의 시각적 품질을 향상시키고, 또한 후속 프레임들에 대한 움직임 추정 및 움직임 보상을 향상시킨다. 실제로, 블록 아티팩트의 고주파수가 제거되고, 따라서 후속 프레임들의 텍스처 잔차(texture residual)에 대해 이들 고주파수가 보상될 필요가 없다.
- [0113] 디블록킹 필터 이후에, 디블록킹된 재구성이 본 발명의 실시예들에 따라 결정된 픽셀의 분류(814)에 기초하여, 단계(804)에서, SAO(sample adaptive offset) 루프 필터에 의해 필터링된다. 얻어진 프레임(805)은 이어서 디스플레이되고 후속 인터 프레임들에 대한 참조 프레임으로서 사용될 재구성된 프레임(807)을 발생시키기 위해 단계(806)에서 ALF(adaptive loop filter)에 의해 필터링될 수 있다.
- [0114] 단계(804)에서, 프레임 영역의 각각의 픽셀이 그의 픽셀 값에 따라 결정된 분류의 클래스로 분류된다. 클래스는 결정된 픽셀 값 범위에 대응한다. 동일한 보상 오프셋 값이 주어진 픽셀 값 범위 내의 픽셀 값을 가지는 모든 픽셀들의 픽셀 값에 가산된다.
- [0115] 샘플 적응적 오프셋 필터링을 위한 픽셀의 분류의 결정은 도 10 내지 도 17 중 임의의 것을 참조하여 이후에 더 상세히 설명될 것이다.

- [0116] 도 9는 본 발명의 일 실시예에 따른, 인코더로부터 데이터를 수신하기 위해 사용될 수 있는 디코더(90)의 블록도를 나타낸 것이다. 디코더는 연결된 모듈들로 나타내어져 있고, 각각의 모듈은, 예를 들어, 장치(600)의 CPU(611)에 의해 실행될 프로그래밍 명령어들의 형태로, 디코더(90)에 의해 구현되는 방법의 대응하는 단계를 구현하도록 구성되어 있다.
- [0117] 디코더(90)는 인코딩 단위들을 포함하는 비트스트림(901)을 수신하고, 각각의 인코딩 단위는 인코딩 파라미터들에 관한 정보를 포함하는 헤더 및 인코딩된 비디오 데이터를 포함하는 보디로 이루어져 있다. 도 7과 관련하여 설명한 바와 같이, 인코딩된 비디오 데이터는 엔트로피 인코딩되어 있고, 움직임 벡터 예측자의 인덱스가, 주어진 블록에 대해, 사전 결정된 수의 비트에 인코딩되어 있다. 수신된 인코딩된 비디오 데이터는 모듈(902)에 의해 엔트로피 디코딩된다. 잔차 데이터가 이어서 모듈(903)에 의해 역양자화되고, 픽셀 값을 획득하기 위해 모듈(904)에 의해 역변환이 적용된다.
- [0118] 코딩 모드를 나타내는 모드 데이터가 또한 엔트로피 디코딩되고, 모드에 기초하여, 인트라 유형 디코딩 또는 인터 유형 디코딩이 인코딩된 영상 데이터 블록에 대해 수행된다.
- [0119] 인트라 모드의 경우에, 비트스트림에 명시된 인트라 예측 모드에 기초하여 역 인트라 예측 모듈(905)에 의해 인트라 예측자가 결정된다.
- [0120] 모드가 인터인 경우, 인코더에 의해 사용되는 참조 영역을 찾아내기 위해 움직임 예측 정보가 비트스트림으로부터 추출된다. 움직임 예측 정보는 참조 프레임 인덱스 및 움직임 벡터 잔차로 이루어져 있다. 움직임 벡터 디코딩 모듈(910)에 의해 움직임 벡터를 획득하기 위해 움직임 벡터 예측자가 움직임 벡터 잔차에 가산된다.
- [0121] 움직임 벡터 디코딩 모듈(910)은 움직임 예측에 의해 인코딩된 각각의 현재 블록에 대해 움직임 벡터 디코딩을 적용한다. 현재 블록에 대한 움직임 벡터 예측자의 인덱스가 획득되면, 현재 블록과 연관되어 있는 움직임 벡터의 실제 값이 디코딩되고 모듈(906)에 의해 역 움직임 보상을 적용하는 데 사용될 수 있다. 역 움직임 보상(906)을 적용하기 위해, 디코딩된 움직임 벡터에 의해 표시되는 참조 영상 부분이 참조 영상(908)으로부터 추출된다. 후속하는 디코딩된 움직임 벡터의 역 예측에 사용되기 위해, 움직임 벡터 필드 데이터(911)가 디코딩된 움직임 벡터로 업데이트된다.
- [0122] 마지막으로, 디코딩된 블록이 획득된다. 도 8을 참조하여 기술된 바와 같이 인코더에서 적용되는 사후 필터링 모듈(815)과 유사하게, 사후 필터링 모듈(907)에 의해 사후 필터링이 적용된다. 디코딩된 비디오 신호(909)가 최종적으로 디코더(90)에 의해 제공된다.
- [0123] 도 10은 보상 오프셋의 적용을 위해 영상의 재구성된 픽셀을 분류하는 본 발명의 제1 실시예에 따른 방법의 단계들을 나타낸 플로우차트이다. 이 실시예에서, 프레임 영역의 재구성된 픽셀의 픽셀 값에 따른 그 픽셀의 분류에 대한 클래스가 프레임 영역의 재구성된 픽셀 값의 통계적 분포에 기초하여 결정된다. 중앙, 유효 범위 및 클래스당 픽셀의 양이 픽셀 값의 분포에 기초하여 결정된다. 이 실시예에서, 디코더는 분포의 세그먼트화를 위한 디코더와 완전히 동일한 프로세스를 적용할 수 있다.
- [0124] 이 프로세스의 초기 단계에서, 모듈(1002)은 프레임 영역(1001)의 픽셀들의 픽셀 값의 통계적 분포를 결정하기 위해 그리고 대응하는 히스토그램(1003)을 발생시키기 위해 현재 프레임 영역(1001)을 스캔한다. 한 특정의 실시예에서, 이 프로세스는 각각의 픽셀 값에 대한 픽셀들의 수를 포함하는 테이블을 업데이트하는 것을 포함한다 (즉, 각각의 픽셀 값에 대해, 그 픽셀 값을 가지는 픽셀들의 수가 표로 작성됨). 테이블은 픽셀들의 비트 심도에 기초하여 표현식  $Max = 2^{Bitdepth} - 1$ 에 따라 결정된 최대 픽셀 값 MAX와 같은 셀들의 수를 포함한다.
- [0125] 모듈(1004)은 이어서 발생된 히스토그램(1003)의 중앙을 결정한다. 히스토그램의 유효 픽셀 값 범위가 이어서 히스토그램(1003)에 표현된 픽셀 값들의 분포에 따라 그리고, 적절한 경우, 히스토그램의 중앙에 기초하여 모듈(1006)에 의해 결정된다. 마지막으로, 픽셀 값 범위를 정의하는 동일 확률의(equiprobable) 클래스가 결정된다. 이와 같이, 각각의 클래스의 픽셀 값 범위를 포함하는 테이블(1009) 또는, 다른 대안으로서, 각각의 픽셀의 픽셀 값을 포함하는 테이블이 제공된다. 본 발명의 어떤 실시예들에서, 동일 확률의 클래스의 결정은 사전 결정된 클래스 수(1000)에 의존할 수 있다.
- [0126] 단계(1004)에서, 발생된 히스토그램(1003)의 중앙을 결정하기 위해 다양한 알고리즘들이 이용될 수 있다. 일 실시예에서, 히스토그램의 최소값  $Min_{Hist}$  및 최대값  $Max_{Hist}$ 가 찾아질 수 있다. 최소값  $Min_{Hist}$ 를 찾아내기 위해, 히스토그램의 셀들  $Hist_k$ 이 픽셀 값 0부터 0이 아닌 히스토그램의 제1 셀  $Hist_k$ 까지 스캔된다. 그리고  $Max_{Hist}$

를 찾아내기 위해, 셀들이 역순으로(최대 픽셀 값 MAX로부터 0이 아닌 히스토그램의 제1 셀  $Hist_k$ 까지) 스캔된다. 히스토그램의 중앙  $Center_{Hist}$ 는 다음과 같이 계산된다:

$$Center_{Hist} = (Max_{Hist} - Min_{Hist}) / 2 + Min_{Hist}$$

대안의 실시예에서, 히스토그램의 중앙은 분포의 가중 평균 중앙(weighted average center)인 것으로 간주된다. 히스토그램 셀  $Hist_k$ 의 값이 값  $k$ 를 가지는 픽셀의 수인 것으로 간주되는 경우,  $Center_{Hist}$ 는 다음과 같이 계산되고:

$$Center_{Hist} = \frac{\sum_{k=0}^{MAX} k \times Hist_k}{N}$$

여기서  $N$ 은 현재 프레임 영역에 있는 픽셀의 수이다.

단계(1006)에서, 발생된 히스토그램의 유효 범위를 결정하는 한가지 가능한 기법은 유효 범위의 양쪽 끝에 대해 앞서 기술한  $Min_{Hist}$  및  $Max_{Hist}$ 를 선택하는 것이다.

다른 실시예에서, 히스토그램의 최소값  $Min_{Range}$ 는 0부터 임계값  $\alpha$ 보다 큰 값을 가지는 제1  $Hist_k$ 까지 스캔하는 것에 의해 결정된다. 동일한 방식으로,  $Max_{Range}$ 는 최대 픽셀 값 MAX로부터 임계값  $\alpha$ 보다 큰 제1  $Hist_k$ 까지 역 스캔하는 것에 의해 결정된다. 임계값  $\alpha$ 는 사전 결정된 값일 수 있다. 다른 대안으로서, 임계값  $\alpha$ 는 프레임 영역에 있는 픽셀들의 수에 및/또는 입력 신호의 성분 유형(크로마 및 루마)에 의존할 수 있다.

한 특정의 실시예에서, 클래스의 수가 인코더측 및 디코더측에 알려져 있는 것으로 간주될 수 있다. 픽셀 값의 클래스의 수는, 예를 들어, 각각의 성분(루마, 크로마 U 및 V)에 따른 현재 프레임 영역 내의 픽셀들의 수에 의존할 수 있다.

동일 확률의 클래스를 생성하기 위해, 유효 범위(1007) 내의 픽셀들의 수  $NbPix_{Range}$ 가 정의된다. 유효 범위 내의 픽셀들의 수  $NbPix_{Range}$ 는 각각의 히스토그램 셀  $Hist_k$ 를  $k = Min_{Range}$ 로부터  $k = Max_{Range}$ 까지 스캔하는 것에 의해 결정된다. 이어서, 각각의 클래스  $NbPix_{classes}$  내의 최적의 픽셀 수를 결정하기 위해, 유효 범위 내의 사전 결정된 픽셀 수  $NbPix_{Range}$ 가 클래스 수(1000)로 나누어진다.

도 11은 본 발명의 일 실시예에 따른, 동일 확률의 클래스를 결정하는 알고리즘의 단계들을 나타낸 플로우차트이다. 초기 단계(1101)에서, 클래스 번호  $j$ 가 0으로 설정되고, 현재 픽셀 값  $k$ 가  $Min_{Range}$ 로 설정된다. 동일 확률의 분류(equiprobable classification)에 대해, 클래스는 그의 픽셀 값 범위에 의해 식별된다. 클래스 번호

$j$ 는 이와 같이 그의 최소 픽셀 값  $Min_j$ 부터 그의 최대 픽셀 값  $Max_j$ 까지의 그의 범위  $[Min_j; Max_j]$ 에 의해 식별된다.

단계(1103)에서,  $j$ 에 의해 인덱싱되는 현재 클래스의 최소 픽셀 값  $Min_j$ 는 현재 픽셀 값  $k$ 로 설정된다. 이어서, 단계(1104)에서,  $SumNbPix_j$ 는 0으로 설정된다.  $SumNbPix_j$ 는 범위  $j$  내의 픽셀들의 수에 대응한다. 이어서, 단계(1105)에서, 픽셀 값  $k$ 를 가지는 픽셀들의 수( $Hist_k$ )가  $SumNbPix_j$ 에 가산된다. 단계(1106)에서, 현재 클래스  $j$ 에 대한 픽셀들의 수의 합  $SumNbPix_j$ 이 클래스들 내의 픽셀들의 수  $NbPix_{classes}$ 보다 큰지 여부가 판정된다. 이 조건에 도달되지 않은 경우, 단계(1107)에서  $k$  값이 증가되고, 단계(1105)에서 픽셀들의 수  $Hist_k$  또는 픽셀 값  $k$ 가  $SumNbPix_j$ 에 가산된다.  $SumNbPix_j > NbPix_{classes}$ 인 것으로 판정되는 경우 또는  $k$ 가 유효 범위의 최대값  $Max_{Range}$ 에 도달하는 경우, 단계(1108)에서 현재 클래스  $j$ 에 대한 최대값이  $k$ 의 현재 값과 같다. 이 스텝

이지에서, 클래스  $j$ 가 정의된다. 즉, 클래스  $j$ 의 범위  $[Min_j; Max_j]$ 가 결정된다. 2개 이상의 클래스에서 동일한 픽셀 값을 획득하는 것을 피하기 위해 단계(1109)에서 변수  $k$ 가 증가된다. 더욱이, 그 다음 클래스에

대한 픽셀 값 범위를 정의하기 위해 단계(1110)에서 변수  $j$ 가 또한 증가된다. 변수  $j$ 가 클래스 수  $NbPix_{classes}$ 보다 큰 경우, 단계(1112)에서 모든 클래스가 정의된 것으로 간주될 수 있다.

- [0137] 그 결과, 인코더는, 도 3과 관련하여 기술된 바와 같이, 각각의 클래스  $j$ 에 대한 오프셋 값을 결정하고 이를 디코더로 전송할 것이다. 인코더 및 디코더는, 도 4를 참조하여 기술된 바와 같이, 프레임 영역을 필터링할 것이다.
- [0138] 이 실시예에서, 클래스 수가 구문 값(syntax value)에 기초하여 사전 결정되어 있기 때문에, 클래스 수  $NbClasses$ 가 픽셀 값에 의존하지 않는다는 것에 유의해야 한다. 그 결과, 이 실시예에서, SAO 대역 오프셋의 파싱(parsing)은 다른 프레임들의 디코딩에 독립적이다. SAO 대역 오프셋에 대한 파싱이 각각의 오프셋의 파싱을 포함한다는 것에 유의해야 한다.
- [0139] 동일 확률의 분류를 결정하는 추가의 실시예에서, 클래스 수가 발생된 히스토그램에서의 픽셀 값의 분포에 따라 결정될 수 있다. 실제로, 유효 범위의 진폭이 높거나 낮을 때, 클래스 수가 코딩 효율에 영향을 미칠 것이다. 그 결과, 각각의 클래스에서의 픽셀들의 수는 물론 픽셀 값들의 수를 결정함으로써 보다 나은 적응성있는 분류가 제공될 수 있다.
- [0140] 도 12는 보다 적응성있는 분류를 제공하는 추가의 실시예에 따른 알고리즘의 단계들을 나타낸 플로우차트이다. 이 플로우차트는 도 11의 실시예의 플로우차트에 기초하고 있으며, 여기서 끝 번호가 같은 모듈들은 동등한 기능을 수행한다. 그렇지만, 이 실시예의 결정 모듈(1206 및 1211)은 도 11의 대응하는 모듈(1106 및 1111)에 의해 처리되는 테스트 조건과 상이한 테스트 조건을 처리한다.
- [0141] 이 실시예에서,  $SumNbPix_j > NbPix_{classes}$ 인 경우 또는  $k$ 가 유효 범위의 최대값  $Max_{Range}$ 에 도달하는 경우 또는  $k - Min_j$ 가 클래스에 대한 최대 범위( $MaxClass_{Range}$ ) 미만인 경우, 결정 모듈(1206)은  $k$  값에 기초하여 루프를 중단하고 클래스  $j$ 에 대해  $Max_j$ 를 선택한다.  $k - Min_j$ 는 클래스  $j$ 의 현재 범위 내의 픽셀 값들의 수에 대응한다.  $MaxClass_{Range}$ 는 범위 내의 픽셀 값들의 사전 결정된 최대 수이다. 이 범위는 비트 심도, 프레임 영역 내의 픽셀들의 수  $N$ , 및 신호의 유형(루마, 크로마  $U$  및  $V$ )에 의존할 수 있다. 예를 들어, 비트 심도가 8일 때, 루마 성분에 대한  $MaxClass_{Range}$ 는 HEVC 구현예에서와 같이 16일 수 있다.
- [0142] 도 11의 실시예와 비교하여 도 12의 실시예의 장점은 큰 진폭을 갖는 픽셀 값 분포에 대한 그의 코딩 효율이다. 이 실시예는 분포에 대해 보다 적응성이 있다.
- [0143] 이 실시예에서, 결정된 클래스 수가 픽셀 값에 의존하고, 따라서 현재 프레임의 파싱이 이전 프레임의 디코딩에 의존한다는 것에 유의해야 한다. 전송 오류에 보다 강건하기 위해, 클래스 수  $NbClasses$ 가 비트스트림에 삽입된다. 이러한 데이터의 전송은 코딩 효율에 그다지 영향을 미치지 않는다.
- [0144] 도 10 내지 도 12의 분류의 제1 실시예의 주된 장점은 분류가 픽셀 값 분포에 적응된다는 것이다. 더욱이, 각각의 클래스의 중앙, 유효 범위 및 크기 그리고 그의 양이 전송될 필요가 없다. 그 결과, 공지된 HEVC 구현예에서와 같이, 결정된 분류에 대해 각각의 클래스의 오프셋을 나타내는 데이터 이외의 어떤 부가의 데이터도 전송될 필요가 없다.
- [0145] 분류의 파라미터들의 시그널링을 수반하는, 분류를 결정하는 본 발명의 추가의 실시예가 도 13을 참조하여 이제부터 기술될 것이다. 분류의 추가의 실시예의 목적은 픽셀 값들의 분포의 최적의 분류를 제공하는 것이다. 이전의 실시예들과 비교하여 차이점은 분류가 직접 픽셀 값의 분포에 기초하여 결정되지 않고 레이트 왜곡 기준에 기초하여 결정된다는 것이다. 추가의 실시예에서, 인코더는 사전 정의된 가능한 분류들 중에서 픽셀 값 분포에 가장 잘 적응된 분류를 선택한다. 이 선택은 레이트 왜곡 기준에 기초하고 있다. 이전의 실시예들에서와 같이, 픽셀 값의 분포를 나타내는 발생된 히스토그램의 클래스들의 중앙, 유효 범위 및 크기가 결정된다. 추가의 실시예에서, 이들 파라미터가 비트스트림에서 전송된다. 이러한 데이터의 전송의 영향을 최소화하기 위해, 클래스 및 관련 범위의 크기가 사전 정의된 값들 중에서 선택된다. 그 결과, 인코더는 선택된 분류의 중앙, 선택된 분류에 관련된 인덱스, 및 분류의 클래스의 크기를 비트스트림에 삽입한다.
- [0146] 픽셀 값의 분포에 대한 적응을 제공하기 위해, 도 13에 나타난 바와 같이, 몇개의 픽셀 값 범위 크기가 정의된다. 도 13에서, 전체 픽셀 값 범위가 32개의 하위 범위로 나누어진다. 픽셀 값 범위의 중앙에 위치해 있는 픽셀 값에 관한 제1 클래스 그룹에 대해, 4개의 예(1301, 1302, 1303, 1304)가 나타내어져 있다. 제1 예(1301)는 가능한 32개의 범위 중 26개의 범위를 포함하고 있다. 이와 같이, 유효 범위(1301)는 전체 범위의 13/16

을 나타낸다. 동일한 방식으로, 1302는 32개의 가능한 범위들 중 단지 8개의 범위(즉, 유효 범위의 1/4)를 나타내고, 1303은 전체 범위의 1/8을 나타내며, 1304는 전체 범위의 1/16을 나타낸다. 제안된 방식에 대해, 전체 범위로부터 단지 하나의 픽셀 값에 대응하는 범위까지의 모든 가능한 크기가 고려될 수 있다. 가능한 유효 범위들의 수는 코딩 효율에 따라 사전 결정되어야만 하거나, 프레임 영역 내의 픽셀들의 수에 대해 사전 결정되어야만 한다.

[0147] 도 13도 역시 픽셀 값 범위의 가장자리에 위치해 있는 픽셀 값에 관한 제2 클래스 그룹에 대한 크기의 몇가지 예를 나타낸 것이다. 제2 그룹은 2개의 하위 클래스 그룹 - 하나는 히스토그램의 각각의 가장자리 쪽에 위치해 있음 - 을 포함하고 있다. 예(1305, 1306, 1307, 1308)는, 각각, 제1 그룹의 예(1301, 1302, 1303, 1304)와 동일한 픽셀 값 수를 나타낸다.

[0148] 본 발명의 실시예에서, 종래 기술의 방법과 비교하여, 클래스의 크기(즉, 클래스당 픽셀 값 범위)가 고정되어 있지 않다. 도 14는 몇개의 크기의 예(1401 내지 1406)를 나타낸 것이다. 이 예에서, 클래스 크기가 32개의 픽셀(1401)부터 단지 하나의 픽셀(1406)까지이다. 도 13과 관련하여 앞서 기술한 바와 같이, 이들 클래스 크기는 모든 가능한 유효 범위들과 결합될 수 있다. 이 실시예에서, 모든 클래스가 특정의 픽셀 값 범위에 대해 동일한 크기를 가지는 것으로 간주된다. 이와 같이, 한 그룹에 대해, 유효 범위의 크기 및 클래스의 크기를 나타내는 데이터가 비트스트림에 삽입된다.

[0149] 다른 실시예에서, 주어진 유효 범위에 대한 클래스의 크기가 유효 범위에서의 클래스의 위치에 따라 적응된다. 보다 정확하게는, 클래스의 크기가 픽셀 값들의 분포에 적응된다. 추가의 실시예에서, 이들 크기가 각각의 유효 범위에 대해 픽셀 값 분포에 따라 사전 결정된다. 실제로, 픽셀 값 분포의 히스토그램은 일반적으로 가우스 분포에 대응한다. 픽셀 값이 히스토그램의 중앙에 가까울수록, 이 값에 가까운 픽셀 값을 가지는 픽셀들이 더 많다. 이는 중앙에 가까운 히스토그램 셀  $Hist_k$ 가 히스토그램의 유효 범위의 양쪽 끝에 있는 히스토그램 셀  $Hist_k$ 보다 더 큰 값(대응하는 픽셀들의 수)을 가진다.

[0150] 도 15는 클래스의 크기에 대한 2개의 기술된 실시예의 예를 나타낸 것이다. 예(1501)는 32개의 픽셀 값의 유효 범위에 대해 8개의 픽셀 값의 고정된 크기를 나타낸 것이다. 1502는 동일한 유효 범위 크기에 대해 4개의 픽셀 값의 고정된 크기를 나타낸 것이다.

[0151] 예(1503)는 32개의 픽셀 값의 현재 범위에 대해 적응적 클래스 크기에 대한 다른 실시예를 나타낸 것이다. 이 예에서, 유효 범위의 양쪽 끝에 있는 클래스는 중앙에 있는 클래스보다 더 크며(즉, 보다 넓은 픽셀 값 범위를 가짐), 이들 클래스는, 각각, 8개의 픽셀 값 및 2개의 픽셀 값을 가진다. 이들 클래스 사이에 있는, 2개의 다른 클래스는 4개의 픽셀 값으로 된 범위를 가진다.

[0152] 제2 그룹에 대한 클래스들의 크기도 역시 픽셀 값들의 분포에 적응될 수 있다. 현재의 HEVC 구현예의 제2 그룹의 목표는 히스토그램의 2개의 끝부분만을 이용하는 것이다. 실제로, 히스토그램의 양쪽 끝은 (손실 코딩으로 인한) 오차가 저주파수와 비교하여 보통 더 큰 고주파수에 종종 관련되어 있는 극단적인 값을 포함한다. 제1 그룹에서와 동일한 방식으로, 제2 그룹의 유효 범위들에 대해 몇개의 클래스 크기가 테스트될 수 있다. 그 경우에, 제2 그룹의 2개의 하위 그룹에 대해, 레이트 왜곡 기준을 사용하여 세부 구역(subdivision)(1501 및 1502)이 비교될 수 있다.

[0153] 더욱이, 클래스의 크기가 적응되는 실시예가 적용될 수 있다. 예(1504)는 제2 그룹의 제1 범위(좌측)에 대한 제안된 적응된 클래스 크기를 나타낸 것이다. 그리고 예(1505)는 제2 그룹의 제2 하위 그룹(우측)에 대한 제안된 적응된 클래스 크기를 나타낸 것이다. 그 경우에, 중앙에 가까운 클래스보다 양쪽 끝에 있는 클래스가 더 많은 픽셀 값을 포함한다.

[0154] 제2 그룹의 목표는 히스토그램의 양쪽 끝을 이용하는 것이며; 그 결과 제2 그룹에 대해 크기의 역적응(inverse adaptation)을 사용하는 것이 때때로 유용하다. 그 경우에, 예(1504)는 제2 하위 그룹(우측)에 대해 사용되고, 예(1505)는 제2 그룹의 제1 하위 그룹(좌측)에 대해 사용된다. 이 실시예에서, 중앙에 가까운 클래스보다 양쪽 끝에 있는 클래스가 더 적은 픽셀 값을 포함한다. 그 경우에, 목표는 동일 확률의 클래스 분류를 생성하는 것이 아니라 제2 그룹의 양쪽 끝의 보다 나은 세그먼트화를 찾아내는 것이다.

[0155] 픽셀 값들의 통계적 분포의 중앙이 꼭 전체 픽셀 값 범위의 가운데에 있지는 않기 때문에, 유효 범위에 기초한 분포의 중앙이 결정되고 비트스트림에서 영상 데이터와 함께 전송되어야만 한다. 도 16은 전체 범위의 1/4에 대응하는 유효 범위에 대해 상이한 중앙 위치들을 갖는 전체 범위의 한 예를 나타낸 것이다. 도 13의 예(130

2)와 달리, 도 16의 4개의 예(1601, 1602, 1603, 1604)에 대해, 유효 범위의 중앙이 전체 범위의 중앙에 위치해 있지 않다. 이 해결책은 선택된 분류가 픽셀 값들의 분포에 적응될 수 있게 한다.

[0156] 결정된 중앙은 이어서 비트스트림에서 전송하기 위해 코딩될 수 있다. 데이터의 코딩을 위한 몇가지 기법들이 예상될 수 있다.

[0157] 현재 프레임 영역의 비트 심도가 8 비트인 것으로 간주되는 경우, 중앙 값에 대해 고려될 수 있는 위치들의 수는 '256 - 최소 유효 범위의 크기'에 대응한다. 예를 들어, 도 13과 비교하여, 유효 범위의 최소 크기는 2이고, 이들 2개의 클래스가 적어도 1개의 픽셀을 포함할 수 있다. 따라서, 이 특정의 예에 대해, 중앙은 1부터 254까지의 값을 가질 수 있고, 따라서 중앙에 대해 254개의 위치가 고려될 수 있다.

[0158] 다른 해결책은 중앙 값을 정량화하는 것이다. 일 실시예에서, 중앙이 클래스들의 크기에 따라 코딩된다. 따라서, 예를 들어, 클래스들의 크기(또는 적용된 클래스 크기 방식이 사용될 때 현재 유효 범위의 모든 클래스들의 최소 크기)가 1개의 픽셀 값인 경우, 중앙이 정량화되지 않고, 현재 유효 범위에 대한 모든 가능한 중앙 위치들 일 수 있다. 도 16에 도시되어 있는 바와 같이, 클래스들의 크기가 16개의 픽셀 값인 경우, 매 16개의 픽셀 값들마다의 픽셀 값들만이 고려될 수 있다. 이와 같이, 도 16에서, 예(1601, 1602, 1603 및 1604)에 대한 중앙은, 각각, 9, 23, 27 및 6이다. 다른 실시예에서, 알고리즘에서 정의된 클래스들의 최대 크기의 배수인 중앙 위치들만이 고려될 수 있다. 이와 같이, 중앙은 픽셀 값을 클래스들의 최대 크기로 나눈 것이다. 이것은 전송될 비트의 수의 면에서의 감소를 제공한다.

[0159] 더욱이, 이론적으로, 가장 가능성있는 중앙은 전체 범위의 중앙이다. 이와 같이, 디코더측에서 중앙 위치를 결정하기 위해 전송되는 데이터는 전체 범위의 중앙과 현재 분류의 유효 범위의 중앙 사이의 차이이다. 이와 같이, 예를 들어, 도 16에서, 예(1601, 1602, 1603, 1604)에 대해 중앙과 관련하여 전송되는 데이터는, 각각,  $16-9=7$ ,  $16-23=-7$ ,  $16-27=-11$ ,  $16-6=10$ 이다.

[0160] 제2 그룹에 대해, 히스토그램의 중앙이 코딩될 필요가 없다. 따라서, 제2 그룹에 대한 2개의 하위 그룹의 변위를 코딩하기 위해 몇개의 방식들이 고려될 수 있다. 제1 그룹에 대해 기술된 중앙 값의 양자화에 관한 제안된 실시예들은 제2 그룹에 대한 제안된 실시예들로 쉽게 확장될 수 있다.

[0161] 본 발명의 실시예들에서, 유효 범위(선택된 분류)의 위치가 전체 범위에 걸쳐, 즉 전체 범위 내에서의 분류의 위치에 관계없이, 동일한 정밀도(precision) 또는 입도(granularity)로 지정될 수 있다. 이것은 도 16에 도시되어 있는 예(1601 내지 1604)에서의 경우이고, 여기서 위치들(중앙 위치들)은 9, 23, 27 및 6이다. 전체 범위는 0부터 32까지로 표시되어 있다. 32개의 가능한 위치들이 있고, 전체 범위에 걸쳐 입도가 동일하다.

[0162] 그렇지만, 도 19a 및 도 19b에 도시된 바와 같이, 전체 범위의 한 부분에서 전체 범위의 다른 부분에서보다 많은 가능한 위치들을 제공하는 것도 가능하다. 환언하면, 전체 범위 내에서 분류가 어디에 있는지에 의존하여 위치의 입도가 변한다. 이들 실시예는 분류(유효 범위)의 중앙을 전체 범위의 가장 중요한(또는 유망한) 부분에 보다 정밀하게 위치시키기 위해 가변 입도에 의한 전체 범위(여기서 0부터 32까지로 표시되어 있음)의 불균등 양자화를 제안하고 있다. 또한, 불균등 양자화는, 전체 범위의 중요한 부분에서 여전히 적절한 정밀도를 제공하면서, 분류의 위치를 신호하는 데 필요한 비트의 수가 제한될 수 있게 해준다. 예를 들어, 도 19a에 나타낸 바와 같이 전체 범위의 가운데에서, 이러한 보다 세밀한 입도가 적용될 수 있다. 이 도면에서, 가능한 중앙 위치들은 굵은 실선으로 나타내어져 있는 인덱스들에 대응한다. 2개의 가능한 중앙 위치들 사이의 간격은 끝에서보다 전체 범위의 가운데에서 더 작다. 따라서, 전체 범위의 끝에서보다 전체 범위의 가운데에서 중앙 위치가 보다 정밀하게 설정될 수 있다.

[0163] 도 19b에서, 2개의 가능한 중앙 위치들 사이의 간격은 가운데에서보다 전체 범위의 양끝에서 더 작다. 예를 들어, 이 실시예는 분포의 극단 값(extreme value)에서 중요한 샘플 값들을 가지는 경우에 특히 유용할 수 있다.

[0164] 보다 일반적으로, 전체 범위의 어느 곳에서라도 보다 미세한 양자화가 적용될 수 있다.

[0165] 앞서 기술한 바와 같은 가변적인 양자화가 사용될 때, 모든 위치들에 대해 분류 범위(유효 범위의 크기)가 고정될 수 있다. 예를 들어, 분류 범위가 4개의 클래스를 포함할 수 있고, 각각의 클래스는 8개의 픽셀 값으로 이루어져 있다.

[0166] 분류 범위/클래스 크기를 위치에 따라 변하게 하는 것도 가능하고, 따라서 도 19a에서, 분류 범위는, 예컨대, 위치 12 내지 20에서는 8개의 픽셀 값이고, 위치 10 및 26에서는 16개의 픽셀 값이며, 위치 2 및 28에서는 32개의 픽셀 값이다.

- [0167] 가변적인 양자화는, 여기에서 기술된 바와 같이, 분류 범위를 결정하기 위해 적용되는 방법에 관계없이 사용될 수 있다. 이 방법은, 예를 들어, 샘플 값들의 통계적 분포의 특성들을 사용하거나 레이트 왜곡 기준을 사용할 수 있다.
- [0168] 가변적인 양자화는 인코더에서도 디코더에서도 사전 결정될 수 있다. 예를 들어, 인코더 및 디코더는 가능한 중앙 위치들(또는 좌측 위치들)에 인덱스들을 할당할 수 있고, 예컨대, 도 19a에서, 위치 2는 인덱스 0이고, 위치 6은 인덱스 1이며, 위치 10은 인덱스 2이고, 위치 12는 인덱스 3이며, 위치 13은 인덱스 4이고, 기타이다. 그러면, 인코더가 선택된 분류의 인덱스를 디코더로 전송하는 것으로 충분하다. 다른 대안으로서, 가변적인 양자화에 관한 정보가 인코더에서 결정되고 비트스트림을 통해 디코더로 신호될 수 있다.
- [0169] 한 특징의 실시예에서, 히스토그램의 중앙이 항상 전체 범위의 중앙이라는 것이 고려될 수 있다. 따라서, 그 경우에, 하나의 변위만이 고려된다. 양쪽 그룹이 동일한 변위에 의해 중앙으로 스케일링된다. 그 결과, 단지 하나의 데이터 - 제2 그룹의 제1 범위의 변위 - 만 코딩되면 된다. 도 17의 예(1701, 1702, 1703 및 1704)는 이러한 변위들의 예이다. 예(1701, 1702, 1703, 1704)에서, 변위는, 각각, 4, 6, 10 및 0이다. 변위가 예측 없이 직접 코딩될 수 있다.
- [0170] 추가의 실시예에서, 예(1705, 1706, 1707 및 1708)에 나타난 바와 같이, 제2 그룹의 양쪽 하위 그룹이 전체 범위에서 독립적인 위치를 가진다. 2가지 코딩 방식이 고려될 수 있다.
- [0171] 제1 방식에서, 존재하지 않는 제1 그룹의 중앙이 이 존재하지 않는 제1 그룹의 유효 범위의 크기로 코딩된다.
- [0172] 양쪽 그룹을 독립적으로 코딩하는 제2 방식은 전체 범위의 2개의 끝으로부터의 2개의 변위(각각의 그룹에 대해 하나씩)를 전송하는 것이다. 따라서, 예(1705, 1706, 1707 및 1708)에 대해, 전송되는 변위는, 각각, 1705에 대해 11 및  $32-28=4$ 이고, 1706에 대해 21 및 0이며, 1707에 대해 3 및  $32-16=16$ 이고, 1708에 대해 7 및  $32-31=1$ 이다.
- [0173] 도 18은 본 발명의 일 실시예에 따른, 레이트 왜곡 선택 알고리즘의 단계들을 나타낸 플로우차트이다. 간략화된 설명을 위해, 적용된 클래스 크기를 갖지 않는 제1 그룹에 대한 선택만이 고려된다. 앞서 기술된 다른 실시예들에 대한 선택이 쉽게 적용될 수 있다.
- [0174] 초기 단계(1801)에서, 현재 프레임 영역의 통계가 계산된다. 이것은 모든 픽셀 값  $k$ 에 대해 변수들  $Hist_k$  및  $Sum_k$ 을 결정하는 것을 포함한다.  $Hist_k$ 는 변수  $k$ 와 같은 픽셀 값을 가지는 픽셀들의 수에 대응하고,  $Sum_k$ 는 픽셀 값이 값  $k$ 인 모든 픽셀들과 그의 원래의 픽셀 값 사이의 차의 합에 대응한다. 이 알고리즘은 3개의 파라미터 - 클래스의 크기  $S$ , 범위의 크기  $R$  및 중앙  $C$  - 에 관한 3개의 루프를 포함한다. 단계(1803)에서, 제1 루프는 각각의 가능한 클래스 크기를 테스트한다. 예를 들어, 크기는 도 14에 정의되어 있다. 단계(1804)에서, 전체 범위의 각각의 하위 범위에 대한 오프셋이 계산된다. 예를 들어, 비트 심도가 8이고 클래스의 크기가 16인 경우, 전체 범위 내의 32개의 가능한 범위들에 대한 왜곡 및 오프셋이 계산된다. 특성들에 의해, 오프셋 및 왜곡이  $Hist_k$  및  $Sum_k$  또는 현재 범위에서의 모든  $k$  값의 선형 결합에 의해 계산된다. 이어서, 단계(1807)에서, 각각의 가능한 범위  $R$ (1805) 및 각각의 가능한 중앙  $C$ (1806)에 대해, 레이트 왜곡 비용(rate distortion cost)이 평가된다. 이 평가는 레이트 왜곡 기준에 기초하고 있다. 모든 중앙  $C$ (1808), 모든 범위(1809) 및 모든 크기(1810)가 테스트될 때, 단계(1811)에서, 최상의 레이트 왜곡 비용에 기초하여 최상의 파라미터들  $S$ ,  $R$ ,  $C$ 가 선택된다. 동일 확률의 분류를 생성하는 이 제2 방식의 장점은 복잡도의 감소 및 코딩 효율의 향상을 포함한다. 클래스의 중앙, 범위 및 크기의 분류 선택은, 분류가 픽셀 값들의 통계적 분포에 기초하고 있는 실시예들과 비교하여, 최적의 레이트 왜곡 선택을 제공한다. 물론, 이 실시예는 현재의 HEVC 구현예와 비교하여 코딩 효율의 면에서의 향상을 제공한다. 이 방식은, 이전의 방식과 비교하여, 디코더측에서 덜 복잡한데, 그 이유는 픽셀들의 분포가 디코더에서 결정될 필요가 없기 때문이다. 더욱이, 이 방식은 HEVC에서의 공지된 기법들보다 덜 복잡할 수 있는데, 그 이유는 어떤 그룹들에서, 보다 적은 수의 클래스들이 사용되기 때문이다.
- [0175] 도 18에 나타난 알고리즘은 모든 대역 오프셋 파라미터들 - 클래스의 크기  $S$ , 범위  $R$ , 중앙  $C$ 를 나타내는 값의 위치 - 의 전체 레이트 왜곡 기반 선택을 수행한다. 복잡도를 제한하기 위해, 어떤 파라미터들을 고정시키는 것이 가능하다. 도 18의 알고리즘의 하나의 특징의 구현예에서, 크기  $S$  및 범위  $R$ 이 인코더 및 디코더가 알고 있는 주어진 값들에 고정되어 있다. 예를 들어,  $S$ 는 8개의 픽셀 값을 나타낼 수 있고,  $R$ 은 8 픽셀의 클래스 4개에 대응하는 32개의 픽셀 값을 나타낼 수 있다. 그 결과, 최적화될 유일한 파라미터는 중앙  $C$ 를 나타내는 값이다.

- [0176] 본 발명의 실시예들이 픽셀들의 분류를 결정하는 데 픽셀 값 범위에 걸쳐 픽셀 값들의 재분할을 고려하고 있기 때문에, 분류가 그에 따라 상이한 픽셀 값 분포들에 적용될 수 있다. 상세하게는, 분류가 픽셀들의 성분 유형에 따라 적용될 수 있다. 예를 들어, 크로마 성분 픽셀 세트의 경우에, 픽셀 값들이 루마 크로마 픽셀들의 픽셀 값들과 비교하여 더 낮은 경향이 있다. 그에 부가하여, 크로마 U 픽셀 값들은 보다 집중되어 있고 비교적 더 높은 픽셀 값들을 가지는 크로마 V 픽셀 값들의 분포와 상이한 분포를 가진다. 더욱이, 크로마 성분 픽셀들의 경우에, 픽셀 값들의 분포는, 보다 널리 확산된 분포를 제공하는 루마 크로마 픽셀들의 분포와 비교하여, 피크 픽셀 값들 부근에 더 집중되는 경향이 있다.
- [0177] 이상에서 알 수 있는 바와 같이, 디코더측에서 픽셀들의 분포를 결정하는 것을 피하기 위해, SAO 유형(SAO 없음, 에지 오프셋 또는 대역 오프셋) 및 보상 오프셋 값들에 부가하여, 파라미터들 S, R 및 C가 비트스트림에서 전송된다. 클래스 크기 및 범위가 고정되어 있을 때, 디코더가 범위의 중앙을 검색할 수 있게 해주기 위해 C만이 전송된다.
- [0178] 고정된 S 및 R의 경우에, SAO 파라미터들을 인코딩하는 하나의 공지된 해결책은 도 21의 플로우차트의 형태로 기술되는, 도 20a의 의사 코드를 적용하는 것이다.
- [0179] 이 프로세스는 SAO의 유형(코드워드 `sao_type_idx`에 저장되어 있음), 대역 오프셋 유형이 사용될 때 유효 범위의 중앙을 나타내는 값(코드워드 `sao_band_position`에 저장되어 있음), 및 SAO 오프셋들(코드워드들 `sao_offset`에 저장되어 있음)을 비롯한 SAO 파라미터들을 결정하는 것으로 시작한다. 도 20a에서, `cldx`는 SAO가 적용되는 색상 성분의 인덱스를 나타내고, `rx` 및 `ry`는 SAO가 적용되는 영역의 위치를 나타내며, `i`는 샘플 값들의 클래스의 인덱스이다.
- [0180] SAO 파라미터 인코딩은 이어서 단계(2003)에서 unsigned Exp Golomb 코드( $ue(v)$ )(즉, unsigned 가변 길이 코드)를 사용하여 SAO 유형을 인코딩하는 것으로 시작한다. SAO 유형이 유형 5(대역 오프셋)인 경우, 단계(2017)에서, 인코딩은 크기 5의 unsigned 고정 길이 코드( $u(5)$ )를 사용하여 유효 범위의 중앙의 위치를 나타내는 값의 인코딩을 계속한다. 이어서, 범위 내에 포함되어 있는 4개의 클래스에 대응하는 4개의 오프셋의 인코딩이 단계들(2019 내지 2025)에서 반복하여 수행된다. 여기서, 각각의 오프셋이 signed Exp Golomb 코드( $se(v)$ )(즉, signed 가변 길이 코딩(variable-length-coding, VLC) 코드)를 사용하여 인코딩된다. 이어서, 인코딩 프로세스가 단계(2027)로 종료된다.
- [0181] SAO 유형이 대역 오프셋이 아닌 경우, 먼저 SAO 유형이 SAO 없음인지를 검사한다(SAO 없음은 관련 샘플들에 어떤 오프셋도 적용되지 않음을 의미함). SAO 없음이 선택된 경우, 단계(2027)에서, 인코딩 프로세스가 종료한다.
- [0182] 그렇지 않은 경우, 단계들(2007 내지 2013)에서 4개의 에지 오프셋의 반복적 인코딩을 계속한다. 다시 말하지만, 단계(2027)에서, 인코딩 프로세스가 종료한다.
- [0183] 표현할 값들의 범위가 비교적 크지만 이 범위 내의 어떤 값들이 다른 값들보다 가능성이 많을 때 VLC 코드가 일반적으로 사용된다. 가장 가능성이 많은 값들은 짧은 코드를 부여받는 반면, 가능성이 적은 값들은 긴 코드를 부여받는다. 이들 코드의 주된 단점은 FLC(fixed length code, 고정 길이 코드)보다 높은 디코딩 복잡도를 유발한다는 것이다. 실제로, VLC 코드는 코드의 최종 크기를 모르기 때문에 비트별로 판독되어야만 하는 반면, FLC 코드는 그의 크기를 알고 있기 때문에 직접 판독될 수 있다.
- [0184] 도 20b 및 도 22에서, VLC 코드를 FLC 코드로 대체하는, 이 인코딩 프로세스에 대한 대안을 제안한다.
- [0185] 이 인코딩 프로세스는 단계(2001)와 동일한 단계(2201)로 시작한다. 단계(2203)에서, 코드워드 `sao_type_idx`의 VLC 코딩이 FLC 인코딩으로 대체되어 있다. 여기서 6개의 가능한 SAO 유형 값(즉, "SAO 없음" 유형, 4개의 "에지 오프셋" 유형, 및 "대역 오프셋" 유형)을 인코딩하기 위해 3 비트가 필요하다. 이어서, SAO의 유형이 "SAO 없음"인지를 검사한다. 그 경우에, 더 이상 아무 것도 인코딩되지 않고, 프로세스가 단계(2215)로 종료한다. 그렇지 않은 경우, SAO의 유형이 "대역 오프셋"인지를 검사한다. 예인 경우, 범위의 중앙의 위치를 나타내는 값이 크기 5의 unsigned FLC 코드의 형태로 코드워드 `SAO_band_position`에 인코딩된다. 실제로, 이 예에서, 클래스 크기가 8개의 샘플 값이고 범위가 4개의 클래스로 이루어져 있는 경우, 256개 값의 전체 범위에 대해 28개의 상이한 위치들이 가능하다.
- [0186] 이 단계에 뒤이어서, 단계들(2211 내지 2213)에서 4개의 SAO 오프셋의 인코딩이 있게 된다. 여기서, FLC 코드가 단계(2023 및 2011)의 VLC 코드를 대체한다. -31부터 32까지의 정수 오프셋 값들을 커버하는 최대 5 비트의 VLC 코드를 사용하는 대신에, 여기서 단지 4개의 상이한 값들[일반적으로, (-2, -1, 1, 2)]을 인코딩할 수 있

는 2 비트 크기의 FLC 코드를 사용한다. 가능한 값들의 수를 감소시키는 것은 인코딩을 가장 빈번히 사용되는 오프셋 값들에 집중시키는 효과를 가진다.

[0187] 이 프로세스는 단계(2215)에서 오프셋 인코딩 후에 종료한다.

[0188] 유의할 점은, 다른 실시예에서, 오프셋들에 의해 표현되는 범위가 픽처 헤더, 슬라이스 헤더 또는 LCU 헤더에서의 인코딩, 2 비트 코드에 의해 획득되는 오프셋들에 적용될 증배 인자에 의해 확장될 수 있다는 것이다. 예를 들어, 증배 인자가 4인 경우, 인코딩된 오프셋들(-2, -1, 1, 2)은 (-8, -4, 4, 8)로 된다. 증배 인자가 또한 표준화(고정)되어 있거나, 다른 LCU로부터 추론될 수 있다. 예를 들어, 이전 LCU에 적용가능한 증배 인자가 현재 LCU에 적용되는 것으로 가정될 수 있다.

[0189] 이와 유사하게, 다른 실시예에서, 픽처 헤더, 슬라이스 헤더 또는 LCU 헤더에 인코딩되어 있는 천이 값이 2 비트 코드에 의해 획득되는 오프셋들에 적용될 수 있다. 예를 들어, 천이 값이 5인 경우, 인코딩된 오프셋들(-2, -1, 1, 2)은 (3, 4, 6, 7)로 된다. 다시 말하지만, 천이 값이 또한 표준화(고정)되어 있거나, 다른 LCU로부터 추론될 수 있다. 예를 들어, 이전 LCU에 적용가능한 천이 값이 현재 LCU에 적용되는 것으로 가정될 수 있다.

[0190] 테스트로부터, 가능한 오프셋 값들을 보다 적게 가지는 것이 SAO 방법의 성능을 그다지 감소시키지 않는다는 것을 알았다. 어떤 오프셋 값들의 억제에 의해 유발되는 손실이 가능성이 적은 오프셋 값들의 과중한 비트레이트 비용의 억제로 보상되는 것처럼 보인다.

[0191] 부가의 테스트로부터, 또한 상이한 오프셋 값들의 수가, 성능 손실이 그다지 없으면서, 3개 그리고 심지어 2개의 오프셋들(1 비트만 인코딩하면 됨)로 추가로 감소될 수 있다는 것을 알았다.

[0192] 도 23에서, 인코딩 프로세스의 추가적인 개선이 제안되어 있다. 여기서, 에지 오프셋 유형의 경우에 사용되는 오프셋 값들이 에지 오프셋의 유형으로부터 직접 추론될 수 있는 것으로 생각한다. 그 경우에, 에지 오프셋 값들의 인코딩이 필요하지 않다. 다시 말하자면, 에지 오프셋의 각각의 유형이 신호 방향에 따라 4개의 클래스와 연관되어 있고, 각각의 클래스는 연관된 오프셋 값을 가진다. 이 실시예는 테스트가, 일반적으로 주어진 에지 오프셋 유형 및 주어진 클래스에 대해, 오프셋 값들이 서로 가깝고 일반적으로 동일하다는 것을 보여주는 것으로부터 촉발된 것이다. 그 결과, 각각의 에지 오프셋 유형에 대해 한 세트의 4개의 오프셋 값들을 고정시키는 것을 제안한다. 예를 들어, 다음과 같은 연관 관계를 제안한다:

[0193] • 수직 에지 오프셋: (-2, -1, 1, 2)

[0194] • 수평 에지 오프셋 (-2, -1, 1, 3)

[0195] • 제1 대각 에지 오프셋 (-3, -2, -1, 1)

[0196] • 제2 대각 에지 오프셋 (-1, 1, 2, 3)

[0197] 도 23에서의 단계들(2301 및 2303)은 도 22를 참조하여 이미 설명한 단계들(2201 및 2203)과 동일하다. 단계들(2305 및 2307)에서, SAO 유형이, 각각, "에지 오프셋" 또는 "SAO 없음"인지를 검사한다. 이들 경우 둘 다에서, 어떤 오프셋도 인코딩되지 않는다. SAO 유형이 "에지 오프셋"인 경우에, 에지 오프셋 유형 값을 판독할 때, 디코더는 고정된 오프셋 값과의 기지의 연관 관계의 덕분으로 에지 오프셋 유형으로부터 오프셋 값을 추론할 것이다.

[0198] 도 23의 실시예에서, SAO 유형이 "대역 오프셋"인 경우, 단계(2309)에서, 범위의 중앙의 위치를 나타내는 값이 인코딩되고, 4개의 오프셋 값이 단계들(2311 내지 2317)에 의해 반복적으로 인코딩된다. 인코딩 프로세스가 단계(2319)로 종료된다.

[0199] 도 24 및 도 25의 실시예에서, 도 20a 및 도 21에 기술된 인코딩 프로세스의 다른 수정을 적용한다. 이전의 실시예에서 이미 언급한 바와 같이, 범위의 중앙의 위치를 나타내는 정보(sao\_band\_position)를 인코딩하기 위해 5 비트의 FLC 코드가 사용되는 반면, 단지 28개의 상이한 위치들이 사용된다. 그 조건에서, 4개의 FLC 코드(각각이 5 비트 길이임)가 사용되지 않은 채로 있다. 여기서, SAO 유형을 인코딩하기 위해 사용되는 코드워드를 제거하기 위해 이들 4개의 여분의 FLC 코드를 이용하는 것을 제안한다. 범위 위치들 및 에지 오프셋 유형들을 결합하여 코딩하기 위해 새로운 코드워드 SAO\_band\_position\_and E0가 사용될 것이다. 이 새로운 코드워드도 역시 5 비트 길이이다.

- [0200] 중전과 같이, 이 프로세스는 단계(2501)에서 SAO 파라미터들을 정의하는 것으로 시작한다. 이 프로세스는 뒤이어서 단계(2503)에서 SAO가 사용되고 있는지 여부를 나타내는 1 비트 길이 플래그(SAO\_LCU\_flag)를 인코딩한다. SAO가 사용되지 않는 경우[단계(2505)], 프로세스가 종료한다[단계(2507)].
- [0201] SAO가 사용되는 경우, 단계(2509)에서, 어느 유형의 SAO가 사용되는지를 검사한다. SAO 유형이 "대역 오프셋"인 경우, 단계(2513)에서, 범위의 중앙의 위치를 나타내는 값을 인코딩하기 위해 코드워드 SAO\_band\_position\_and\_E0의 처음 28개의 코드가 사용된다. SAO 유형이 "에지 오프셋"인 경우, 단계(2511)에서, 에지 오프셋의 유형(수직, 수평, 제1 대각 또는 제2 대각)을 인코딩하기 위해 코드워드 SAO\_band\_position\_and\_E0의 마지막 4개의 코드가 사용된다. 단계들(2511 또는 2513)에 뒤이어서, 단계들(2515 내지 2521)에 의해 4개의 오프셋 값들을 인코딩한다.
- [0202] 그런데, 본 실시예에서 에지 오프셋의 유형을 인코딩하기 위해 여분의 코드워드들이 사용되지만, 여분의 코드워드들이, 다른 대안으로서, 다른 목적들을 위해 사용될 수 있다는 것을 잘 알 것이다. 인코더로부터 디코더로 송신될 필요가 있는 임의의 다른 정보가 여분의 코드워드들을 사용하여 인코딩될 수 있다.
- [0203] 도 26에서, 대역 오프셋의 경우에 적용될 오프셋들을 결정하는 추가의 실시예를 제안한다. 이 실시예는 테스트가, 대역 오프셋의 경우에, 대부분의 오프셋들이 절대값에서 낮은 진폭을 가진다는 것을 보여주는 것으로부터 촉발된 것이다. 실제로, 오프셋 값들은 일반적으로 -2, -1, 1 및 2이다. 예를 들어, 도 20a 내지 도 25의 예에서와 같이, 범위 내의 클래스 수가 4로 감소될 때, 4개의 오프셋 값들로 된 상이한 그룹들의 수도 역시 감소된다. 4개의 상이한 오프셋 값 및 4개의 클래스를 갖는 상기 예에서, 상이한 그룹들의 수는  $4^4 = 256$ 이다. 도 21, 도 22, 도 23 및 도 25의 실시예들에서, 오프셋 값들(4개의 오프셋 값, 각각이 2 비트를 사용하여 인코딩됨)을 인코딩하기 위해 8 비트가 사용된다. 여기서, 4개의 오프셋으로 된 그룹들 모두가 동일한 선택될 확률을 가지는 것으로 간주되고 있다. 그렇지만, 이들 그룹 중 일부가 다른 것들보다 가능성이 적다. 가능성이 적은 그룹들을 제거함으로써, 이들을 인코딩하는 데 필요한 비트의 수를 감소시키는 것이 가능하다. 그 결과, 각각의 오프셋 값에 대해 2 비트를 사용하여 4개의 상이한 오프셋 값을 인코딩하는 대신에, 4개의 오프셋 값으로 된 상이한 그룹들에 인덱스들을 할당하고 인덱스를 인코딩하는 것 - 가능성이 적은 그룹의 제거로 인해 인덱스가 8 비트 미만을 사용하여 인코딩됨 - 을 제안한다. 훈련 시퀀스 세트에 대해 SAO를 적용하고 그룹들에 관한 통계를 계산함으로써 그룹들의 확률이 결정될 수 있다. 선택될 확률에 따라 정렬되어 있는 모든 가능한 그룹들을 수집하는 테이블이 사전 결정되고 인코더 및 디코더에 알려져 있을 수 있다. 이 테이블에서, 각각의 오프셋 그룹은 인덱스 값에 연관될 것이다. 그룹 인덱스 인코딩에 할당되는 비트의 수가 고정(표준화)되거나 시퀀스, 프레임, 슬라이스 또는 LCU에 대해 고정되어 있고, 대응하는 헤더들에 인코딩되어 있을 수 있다. 가장 가능성있는 그룹들에 대응하는 테이블 내의 그룹들의 서브셋이 인덱스 인코딩에 할당된 비트의 수에 따라 인코더 및 디코더에 의해 결정될 것이다.
- [0204] 최상의 그룹의 선택을 나타내는 실시예가 도 26에 기술되어 있다. 이 프로세스는 변수 j가 초기화되는 단계(2601)에서 시작한다. 모든 가능한 오프셋 그룹들의 테스트를 가능하게 하기 위해 이 변수 j가 점차적으로 증가된다. 제안된 실시예에서, 4개의 오프셋들로 된 그룹을 생각하고 있지만, 다른 오프셋 수가 생각될 수 있을 것이다. 단계(2603)에서, 모든 그룹들이 테스트되었는지(예를 들어, NbOffsetGroup이 128일 수 있는지)를 검사한다. 예인 경우, 프로세스가 종료하고, 선택된 그룹에 대응하는 8 비트 미만의 코드워드가 인코딩된다. 아니오인 경우, 프로세스는 변수 i의 초기화로 계속되어, 범위 내의 모든 클래스들을 테스트할 수 있게 한다[단계(2605)]. 단계(2606)에서, 원래의 샘플들과 오프셋 그룹 j에 대응하는 SAO 필터링된 인코딩된 샘플들 간의 차의 합을 나타내는 변수 SumDiff(j)가 0으로 초기화된다. 여기서, 4개의 클래스만이 고려되고 있지만, 오프셋 수와 부합하는 다른 클래스 수가 가능하다. 단계(2607)에서, 테스트될 어떤 클래스들이 남아 있는 경우, 변수 k를 초기화하여, 클래스 i에 대응하는 샘플 범위 내의 모든 가능한 샘플들을 테스트할 수 있게 한다. 단계들(2611 내지 2619)에 의해, 오프셋 그룹 j 내의 클래스 i의 오프셋으로 필터링된 인코딩된 샘플들과 고려되는 클래스 i 내의 원래의 샘플들 간의 차의 절대값의 합을 계산한다. 여기서 orig(k)는 인코딩된 값 enc(k)에 대응하는 원래의 샘플 값들의 평균이다. filter(i,j)는 오프셋 그룹 j 내의 클래스 i에 대응하는 오프셋 값이다. SumDiff(j)는 범위를 구성하는 모든 클래스들(여기서, 4개의 클래스)에 대해 계산된 차의 합이다. 단계들(2621 내지 2626)을 포함하는 루프에서, 모든 계산된 차의 합들이 비교되고, 최소 합을 가지는 그룹의 인덱스가 선택된다. 선택된 인덱스는, 인코딩된 샘플들에 적용될 때, 필터링된 샘플들과 원래의 샘플 간의 차를 최소화할 수 있게 해주는 오프셋 그룹에 대응한다.
- [0205] 구문 인코딩 프로세스 동안, 예를 들어, 도 22의 단계들(2211 내지 2213)로 나타난 오프셋 값들의 인코딩이 선

택된 오프셋 그룹에 대응하는 인덱스의 인코딩으로 대체된다.

- [0206] 루마 성분 픽셀의 분류가 크로마 U 또는 V 성분 픽셀의 분류와 별개로 수행되고, 따라서 각각의 클래스가 유사한 수의 픽셀들을 가지도록 각각의 분류가 그에 따라 적용될 수 있다.
- [0207] 따라서, 본 발명의 실시예들의 방법들은 루마 또는 크로마 신호 둘 다에 대해 독립적으로 보다 최적의 분류를 제공하도록 구성되어 있을 수 있는 보다 유연한 분류 방식을 제공하고, 그로써 코딩 효율의 향상을 가져온다.
- [0208] 본 발명이 특정의 실시예들을 참조하여 이상에서 기술되어 있지만, 본 발명이 특정의 실시예들로 제한되지 않고, 본 발명의 범위 내에 속하는 수정들이 기술 분야의 당업자에게 명백할 것이다.
- [0209] 예를 들어, 이전의 실시예들이 영상의 픽셀들 및 그의 대응하는 픽셀 값들과 관련하여 기술되어 있지만, 본 발명과 관련하여, 픽셀 그룹이 대응하는 그룹 픽셀 값과 함께 고려될 수 있다는 것을 잘 알 것이다. 샘플은 따라서 영상의 하나 이상의 픽셀에 대응할 수 있다.
- [0210] 본 발명의 추가의 측면들이 이하에 기재되어 있다.
- [0211] 본 발명의 제1 추가의 측면에 따르면, 영상의 재구성된 샘플 세트에 대한 보상 오프셋들을 제공하는 방법이 제공되고, 각각의 샘플은 각각의 샘플 값을 가지며, 세트의 복수의 샘플들의 샘플 값들은 샘플 값들의 통계적 분포로 표현가능하며, 이 방법은 샘플 값들의 통계적 분포의 특성들에 의존하여, 샘플들의 대응하는 샘플 값에 따라 샘플들을 재분할하기 위한 복수의 클래스들을 결정하는 단계 - 각각의 클래스는 각각의 샘플 값의 범위를 정의함 -; 및 상기 클래스의 각각의 샘플의 각각의 샘플 값에 적용하기 위한 각각의 보상 오프셋을 각각의 결정된 클래스와 연관시키는 단계를 포함한다.
- [0212] 본 발명의 이 측면에서, 샘플들의 분류를 결정하는 데 샘플 값들의 통계적 분포가 고려되기 때문에, 분류가 그에 따라 모든 가능한 샘플 값의 범위들에 적용될 수 있다. 더욱이, 분류가 샘플들의 성분 유형에 따라 적용될 수 있다. 예를 들어, 크로마 신호에 대응하는 픽셀들의 경우에, 픽셀 값들의 분포는, 보다 널리 확산된 분포를 제공하는 루마 신호와 비교하여, 피크 픽셀 값들 부근에 더 집중되는 경향이 있다. 따라서, 본 발명의 실시예들의 방법들은 루마 또는 크로마 신호 둘 다에 대해 독립적으로 보다 최적의 분류를 제공하도록 구성되어 있을 수 있는 보다 유연한 분류 방식을 제공하고, 그로써 코딩 효율의 향상을 가져온다.
- [0213] 일 실시예에서, 통계적 분포의 특성들은 영상 샘플 값들의 통계적 분포의 결정된 중앙을 포함한다.
- [0214] 일 실시예에서, 통계적 분포의 특성들은 통계적 분포의 유효 샘플 값의 범위를 포함한다.
- [0215] 일 실시예에서, 샘플들이 클래스들 간에 실질적으로 균일하게 공유되도록 클래스들이 결정된다.
- [0216] 일 실시예에서, 세트의 샘플들이 적어도 제1 성분 유형 또는 제2 성분 유형일 수 있고, 복수의 클래스들이 샘플 세트의 성분 유형에 의존하여 결정된다.
- [0217] 일 실시예에서, 클래스 수가 사전 결정되고, 각각의 클래스에 대해 정의된 샘플 값의 범위가 통계적 분포의 특성들에 의존하여 결정된다.
- [0218] 다른 실시예에서, 클래스 수 및 각각의 클래스에 대해 정의된 샘플 값의 범위가 통계적 분포의 특성들에 의존하여 결정된다.
- [0219] 일 실시예에서, 클래스 수가 유효 범위 내의 샘플 값을 가지는 샘플들의 수에 의존하여 결정된다.
- [0220] 일 실시예에서, 클래스 수가 성분 유형에 따라 결정된다.
- [0221] 일 실시예에서, 이 방법은 유효 범위 내의 샘플 값들의 수에 따라 클래스당 샘플 값들의 최대 수를 결정하는 단계를 포함한다.
- [0222] 일 실시예에서, 복수의 클래스들을 결정하는 단계는, 복수의 사전 결정된 분류들 중에서, 통계적 분포의 특성들에 적용되어 있는 복수의 클래스들을 정의하는 분류를 선택하는 단계를 포함한다.
- [0223] 일 실시예에서, 각각의 클래스에 대한 샘플 값의 범위, 통계적 분포의 중앙 및/또는 유효 샘플 값의 범위가 레이트 왜곡 기준에 기초하여 결정된다.
- [0224] 일 실시예에서, 각각의 클래스에 대한 샘플 값의 범위, 통계적 분포의 중앙 및/또는 유효 샘플 값의 범위가 사전 결정되어 있다. 일 실시예에서, 유효 샘플 값의 범위가 임계값에 대한 샘플 값들의 비교에 기초하여 결정되고, 임계값이 샘플들의 총수에 의존하거나, 임계값이 샘플들의 성분 유형에 의존하거나, 임계값이 사전 결정된

값이다.

- [0225] 일 실시예에서, 각각의 클래스에 대한 보상 오프셋이 클래스의 각각의 재구성된 샘플의 샘플 값과 대응하는 원래의 영상의 각각의 샘플 값 사이의 차의 평균으로부터 결정된다.
- [0226] 일 실시예에서, 샘플 세트는 영상의 복수의 샘플 세트들 중 하나이고, 각각의 세트에 대해 동일한 클래스 수가 결정된다.
- [0227] 일 실시예에서, 샘플 값은 비트 심도를 나타내고, 유효 범위, 각각의 클래스의 범위 및/또는 통계적 분포의 중앙은 비트 심도에 의존한다.
- [0228] 일 실시예에서, 주어진 클래스에 대한 샘플 값의 범위가 유효 범위 내에서의 클래스의 위치에 의존한다.
- [0229] 일 실시예에서, 통계적 분포의 가장자리에 위치한 주어진 클래스에 대한 샘플 값의 범위는 분포의 중앙 영역에 있는 주어진 클래스에 대한 샘플 값의 범위보다 크다.
- [0230] 일 실시예에서, 통계적 분포의 중앙은 유효 범위에 기초하여 결정된다.
- [0231] 일 실시예에서, 복수의 클래스들은 통계적 분포의 중앙 부분에 위치한 제1 클래스 그룹 및 통계적 분포의 각각의 가장자리 부분들에 위치한 제1 및 제2 하위 클래스 그룹들을 포함하는 제2 클래스 그룹을 포함한다.
- [0232] 일 실시예에서, 통계적 분포에서의 제2 하위 클래스 그룹들의 위치들이 코딩을 위한 데이터로서 제공된다.
- [0233] 일 실시예에서, 제2 그룹의 하위 그룹들의 위치들이 통계적 분포의 전체 범위와 무관하다.
- [0234] 일 실시예에서, 통계적 분포의 중앙은 코딩을 위해 제공되지 않는다.
- [0235] 일 실시예에서, 전체 범위 내에서의 유효 범위(분류)의 위치가 전체 범위에 걸쳐 분포되어 있는 복수의 가능한 위치들 중에서 선택되고, 2개의 연속적인 위치들 사이의 간격은 전체 범위의 적어도 하나의 부분에서 전체 범위의 다른 부분에서보다 더 작다.
- [0236] 예를 들어, 보다 작은 간격을 가지는 부분은 전체 범위의 가운데에 있을 수 있다.
- [0237] 다른 예에 따르면, 보다 작은 간격을 가지는 부분은 전체 범위의 한쪽 끝 또는 양쪽 끝에 있을 수 있다.
- [0238] 위치들이 중앙 위치일 수 있다. 다른 대안으로서, 위치들이 끝 위치일 수 있다.
- [0239] 가능한 위치들이 인덱스들을 할당받을 수 있다.
- [0240] 일 실시예에서, 각각의 클래스에 대한 샘플 값의 범위, 통계적 분포의 중앙 및/또는 유효 샘플 값의 범위가 사전 결정되어 있다.
- [0241] 일 실시예에서, 적어도 2개의 상이한 유형의 오프셋 값들이 발생될 수 있고, 유형들 중 하나는, 앞서 기술한 바와 같이, 클래스들(샘플 값의 범위들)과 각각 연관되어 있는 소위 대역 오프셋 값들이다. 하나 이상의 다른 유형의 오프셋 값들은 소위 에지 오프셋 값들일 수 있다. 방향마다 상이한 유형의 에지 오프셋 값들이 있을 수 있다 - 예를 들어, 4개의 상이한 유형들은 4개의 상이한 방향들에 각각 대응함 -. 이와 같이, 방향마다 에지 오프셋 값 세트가 있을 수 있다. 한 유형의 오프셋 값(대역 또는 에지)이 이어서 선택될 수 있다. 선택 기준이 제한되지 않으며, 하나의 적당한 기준은 레이트 왜곡 기준이다. 선택된 유형의 SAO 필터링(대역 또는 에지 오프셋)만이 인코더 및 디코더에 의해 적용된다. 각각의 프레임 영역에 대해 선택이 수행될 수 있다. 그에 부가하여, 예를 들어, 어느 유형에 의해서도 그다지 향상이 달성되지 않는 경우, "SAO 없음"을 선택하는 것, 즉 대역 오프셋 및 에지 오프셋 중 어느 것도 적용하지 않는 것이 가능할 수 있다. 일 실시예에서, 선택된 유형의 SAO 필터링이 인코더에 의해 인코딩되고 디코더로 전송된다.
- [0242] 일 실시예에서, 선택된 유형을 인코딩하기 위해 고정 길이 코드가 사용된다.
- [0243] 일 실시예에서, 선택된 유형이 대역 오프셋일 때, 유효 범위의 위치를 나타내는 값이 인코딩된다. 위치는 유효 범위의 중앙 위치 또는 끝 위치일 수 있다.
- [0244] 일 실시예에서, 유효 범위의 위치를 인코딩하기 위해 고정 길이 코드가 사용된다.
- [0245] 일 실시예에서, 유효 범위의 위치를 나타내는 값들에 대응하는 코드워드들을 인코딩하는 데 사용되는 고정 길이 코드의 길이에 의해 허용되는 상이한 코드워드들의 수가 상이한 가능한 위치들의 실제 수보다 많을 때, 기타 정보를 인코딩하기 위해 여분의 코드워드들이 사용된다. 예를 들어, 일 실시예에서, 에지 오프셋의 유형(방향)을

인코딩하기 위해 여분의 코드워드들이 사용된다.

- [0246] 일 실시예에서, 보상 오프셋들이 고정 길이 코드들로 인코딩된다. 이것은 대역 오프셋 또는 예지 오프셋 또는 이들 유형의 오프셋 둘 다에 적용될 수 있다.
- [0247] 일 실시예에서, 고정 길이 코드의 길이에 의해 허용되는 상이한 보상 오프셋들의 수가 가능한 보상 오프셋들의 수보다 적다.
- [0248] 일 실시예에서, 보상 오프셋 값들에 증배 인자(multiplication factor)가 적용되고, 필터링을 위해 증배된 보상 오프셋 값들이 사용된다.
- [0249] 일 실시예에서, 증배 인자가 인코더로부터 디코더로 전송된다.
- [0250] 일 실시예에서, 보상 오프셋 값들에 천이 값(shifting value)이 적용되고, 필터링을 위해 천이된 보상 오프셋 값들이 사용된다.
- [0251] 일 실시예에서, 천이 값이 인코더로부터 디코더로 전송된다.
- [0252] 일 실시예에서, 적어도 하나의 유형(방향)에 대한 예지 오프셋 값들이 사전 결정된다.
- [0253] 일 실시예에서, 영상의 재구성된 샘플 세트에 대해 보상 오프셋들이 적용될 필요가 있는지 여부를 나타내는 플래그가 인코딩된다.
- [0254] 일 실시예에서, 대역 오프셋 값 그룹들이 사전 정의되고, 그룹의 각각의 오프셋은 유효 범위의 하나의 클래스와 연관되어 있다.
- [0255] 일 실시예에서, 각각의 그룹은 인덱스를 할당받는다.
- [0256] 일 실시예에서, 사전 정의된 그룹들이 테이블에 수집되고, 테이블 정렬(table order)은 각각의 그룹의 선택 확률에 의존한다.
- [0257] 일 실시예에서, 각각의 그룹의 선택 확률은 본 발명을 구현하는 보상 오프셋들을 제공하는 방법들을 훈련 시퀀스 세트에 적용함으로써 계산된다.
- [0258] 일 실시예에서, 원래의 샘플들과 필터링된 샘플들 간에 최소 차를 가지는 사전 정의된 대역 오프셋 값 그룹들 중의 대역 오프셋 값 그룹이 선택된다.
- [0259] 일 실시예에서, 그룹들의 인덱스들이 고정 길이 코드를 사용하여 인코딩된다.
- [0260] 일 실시예에서, 고정 길이 코드의 길이에 의해 허용되는 상이한 인코딩된 그룹 인덱스들의 수가 가능한 상이한 그룹들의 수보다 적다.
- [0261] 일 실시예에서, 가능한 상이한 그룹들 내의 대역 오프셋 그룹들의 서브셋이 각각의 그룹의 선택 확률 및 고정 길이 코드의 길이에 의존하여 결정된다.
- [0262] 일 실시예에서, 대역 오프셋 그룹들의 인덱스들을 인코딩하는 데 사용되는 고정 길이 코드의 길이가 사전 결정되어 있다.
- [0263] 일 실시예에서, 보상 오프셋 그룹들의 인덱스들을 인코딩하는 데 사용되는 고정 길이 코드의 길이가 시퀀스 헤더, 픽처 그룹 헤더, 픽처 헤더(picture header), 슬라이스 헤더(slice header) 또는 LCU 헤더에 인코딩된다.
- [0264] 본 발명의 제2 추가의 측면에 따르면, 복수의 샘플들로 이루어져 있는 영상을 인코딩하는 방법이 제공되고, 이 방법은 샘플들을 인코딩하는 단계; 재구성된 샘플들을 제공하기 위해 인코딩된 샘플들을 디코딩하는 단계; 재구성된 샘플들에 대해 루프 필터링을 수행하는 단계 - 루프 필터링은 각자의 재구성된 샘플들의 샘플 값들에 보상 오프셋들을 적용하는 것을 포함하고, 각각의 보상 오프셋은 샘플 값의 범위와 연관되어 있으며, 보상 오프셋들은 제1 추가의 측면의 방법에 따라 제공됨 -; 및 인코딩된 샘플들의 비트스트림을 발생시키는 단계를 포함한다.
- [0265] 일 실시예에서, 이 방법은, 비트스트림에서, 각각의 클래스에 대한 각자의 보상 오프셋들을 나타내는 인코딩된 데이터를 전송하는 단계를 포함한다.
- [0266] 일 실시예에서, 이 방법은, 비트스트림에서, 복수의 결정된 클래스들을 정의하는 인코딩 분류 데이터를 전송하는 단계를 포함한다.

- [0267] 일 실시예에서, 분류 데이터는 통계적 분포의 중앙을 나타내는 데이터를 포함한다.
- [0268] 일 실시예에서, 분류 데이터는 복수의 클래스들 각각에 대해 정의된 샘플 값의 범위를 나타내는 데이터를 포함한다.
- [0269] 일 실시예에서, 분류 데이터는 통계적 분포의 유효 범위를 나타내는 데이터를 포함한다.
- [0270] 본 발명의 제3 추가의 측면에 따르면, 복수의 샘플들로 이루어져 있는 영상을 디코딩하는 방법이 제공되고, 이 방법은,
- [0271] 인코딩된 샘플들을 수신하는 단계;
- [0272] 재구성된 샘플들을 제공하기 위해 인코딩된 샘플들을 디코딩하는 단계; 및
- [0273] 재구성된 샘플들에 대해 루프 필터링을 수행하는 단계 - 루프 필터링은 각자의 재구성된 샘플들의 샘플 값들에 보상 오프셋들을 적용하는 것을 포함하고, 각각의 보상 오프셋은 샘플 값의 범위와 연관되어 있으며, 보상 오프셋들은 이전의 실시예들 중 임의의 실시예에 따라 제공됨 - 를 포함한다.
- [0274] 본 발명의 추가의 측면들 중 다른 측면은 복수의 샘플 값들로 이루어져 있는 영상을 디코딩하는 방법을 제공하고, 이 방법은
- [0275] 인코딩된 샘플 값들을 수신하는 단계;
- [0276] 이전의 실시예들 중 임의의 실시예의 방법에 따라 제공되는 각자의 보상 오프셋들과 연관되어 있는 복수의 클래스들을 정의하는 인코딩된 분류 데이터를 수신하는 단계 - 각각의 보상 오프셋은 샘플 값의 범위에 대응함 -;
- [0277] 재구성된 샘플들을 제공하기 위해 인코딩된 샘플들을 디코딩하고 인코딩된 보상 오프셋들을 디코딩하는 단계; 및
- [0278] 샘플의 샘플 값에 따라, 재구성된 샘플들에 대해 루프 필터링을 수행하는 단계 - 루프 필터링은 각자의 샘플들의 영상 샘플들에 수신된 보상 오프셋들을 적용하는 것을 포함함 - 를 포함한다.
- [0279] 일 실시예에서, 분류 데이터는 통계적 분포의 중앙을 나타내는 데이터, 복수의 클래스들 각각에 대해 정의된 샘플 값의 범위를 나타내는 데이터, 및/또는 통계적 분포의 유효 범위를 나타내는 데이터를 포함한다.
- [0280] 본 발명의 추가의 측면들 중 다른 측면에 따르면, 비디오 비트스트림에 의해 표현되는 영상에 대한 정보 데이터 세트를 전달하는 신호가 제공되고, 영상은 재구성가능 샘플 세트를 포함하며, 각각의 재구성가능 샘플은, 재구성 후에, 각자의 샘플 값을 가지며, 세트의 복수의 재구성된 샘플들의 샘플 값들은 샘플 값들의 통계적 분포로 표현가능하고; 정보 데이터 세트는 각자의 재구성된 샘플들의 샘플 값들에 적용하기 위한 각자의 보상 오프셋들과 연관되어 있는 복수의 클래스들을 나타내는 분류 데이터를 포함하며, 분류 데이터는 통계적 분포에 따라 결정된다.
- [0281] 일 실시예에서, 분류 데이터는 통계적 분포의 중앙을 나타내는 데이터, 복수의 클래스들 각각에 대해 정의된 샘플 값의 범위를 나타내는 데이터, 및/또는 통계적 분포의 유효 범위를 나타내는 데이터를 포함한다.
- [0282] 본 발명의 추가의 측면들 중 다른 측면에 따르면, 영상의 재구성된 샘플 세트에 대한 보상 오프셋들을 제공하는 장치가 제공되고, 각각의 샘플은 각자의 샘플 값을 가지며, 세트의 복수의 샘플들의 샘플 값들은 샘플 값들의 통계적 분포로 표현가능하며, 이 장치는
- [0283] 샘플 값들의 통계적 분포의 특성들에 의존하여, 샘플들의 대응하는 샘플 값에 따라 샘플들을 재분할하기 위한 복수의 클래스들을 결정하는 수단 - 각각의 클래스는 각자의 샘플 값의 범위를 정의함 -; 및
- [0284] 상기 클래스의 각각의 샘플의 각자의 샘플 값에 적용하기 위한 각자의 보상 오프셋을 각각의 결정된 클래스와 연관시키는 수단을 포함한다.
- [0285] 본 발명의 추가의 측면들 중 다른 측면은 복수의 샘플들로 이루어져 있는 영상을 인코딩하는 인코딩 장치를 제공하고, 이 장치는,
- [0286] 샘플들을 인코딩하는 인코더; 재구성된 샘플들을 제공하기 위해 인코딩된 샘플들을 디코딩하는 디코더; 재구성된 샘플들을 필터링하는 루프 필터 - 루프 필터링 수단은 각자의 재구성된 샘플들의 샘플 값들에 보상 오프셋들을 적용하는 오프셋 적용 수단을 포함하고, 각각의 보상 오프셋은 샘플 값의 범위와 연관되어 있으며, 보상 오프셋들은 이전의 실시예의 장치에 의해 제공됨 -; 및 인코딩된 샘플들의 비트스트림을 발생시키는 비트스트림

발생기를 포함한다.

[0287] 본 발명의 추가의 측면들 중 또 다른 측면은 복수의 샘플들로 이루어져 있는 영상을 디코딩하는 디코딩 장치를 제공하고, 이 장치는, 인코딩된 샘플들을 수신하는 수신기; 재구성된 샘플들을 제공하기 위해 인코딩된 샘플들을 디코딩하는 디코더; 및 재구성된 샘플들을 루프 필터링하는 루프 필터 - 루프 필터는 각자의 재구성된 샘플들의 샘플 값들에 보상 오프셋들을 적용하는 수단을 포함하고, 각각의 보상 오프셋은 샘플 값의 범위와 연관되어 있으며, 보상 오프셋들은 이전의 실시예에 따른 장치에 의해 제공됨 - 를 포함한다.

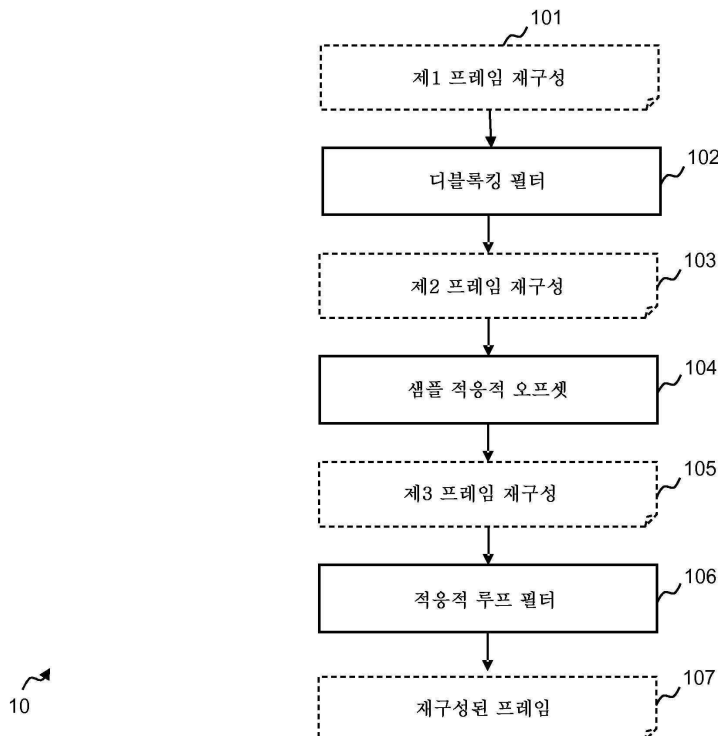
[0288] 본 발명의 추가의 측면들 중 다른 측면은 복수의 샘플 값들로 이루어져 있는 영상을 디코딩하는 디코딩 장치를 제공하고, 이 장치는, 인코딩된 샘플 값들을 수신하고 이전의 실시예에 따른 장치에 의해 제공되는 각자의 보상 오프셋들과 연관되어 있는 복수의 클래스들을 정의하는 인코딩된 분류 데이터를 수신하는 수신기 - 각각의 보상 오프셋은 샘플 값의 범위에 대응함 -; 재구성된 샘플들을 제공하기 위해 인코딩된 샘플들을 디코딩하고 인코딩된 보상 오프셋들을 디코딩하는 디코더; 및 샘플의 샘플 값에 따라, 재구성된 샘플들을 루프 필터링하는 루프 필터 - 루프 필터는 각자의 샘플들의 영상 샘플들에 수신된 보상 오프셋들을 적용하는 수단을 포함함 - 를 포함한다.

[0289] 단지 예로서 제공되어 있고 첨부된 특허청구범위에 의해서만 결정되는 본 발명의 범위를 제한하기 위한 것이 아닌 이상의 예시적인 실시예들을 참조하면 많은 추가의 수정들 및 변형들이 기술 분야의 당업자에게 안출될 것이다. 상세하게는, 상이한 실시예들로부터의 상이한 특징들이, 적절한 경우, 상호 교환될 수 있다.

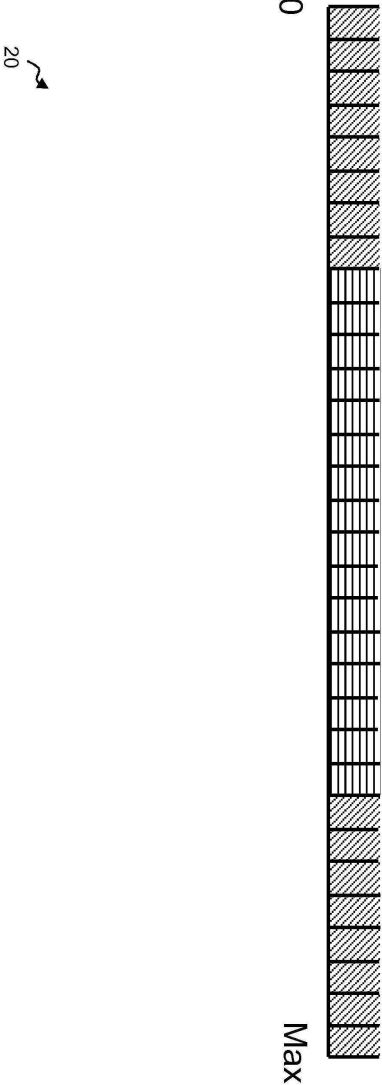
[0290] 특허청구범위에서, 용어 "포함하는"은 다른 구성요소들 또는 단계들을 배제하지 않으며, 단수 표현 "한" 또는 "어떤"은 복수를 배제하지 않는다. 상이한 특징들이 서로 다른 종속 청구항에 언급되어 있다는 단순한 사실이 이들 특징의 조합이 유리하게 사용될 수 없다는 것을 말하는 것은 아니다.

## 도면

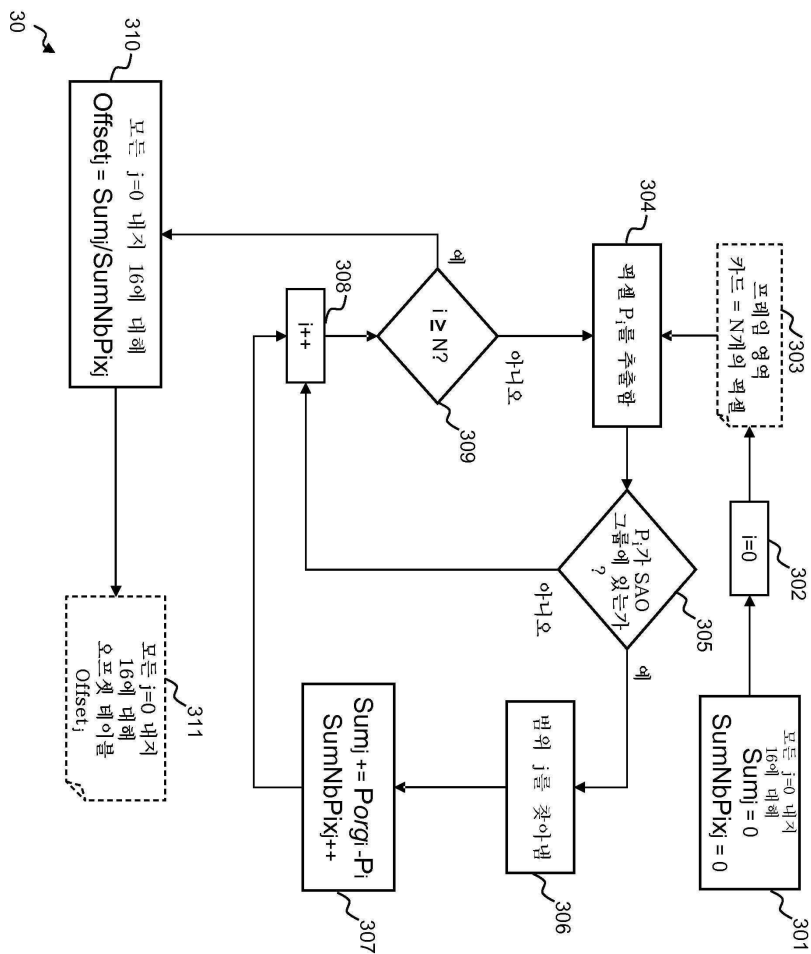
### 도면1



도면2

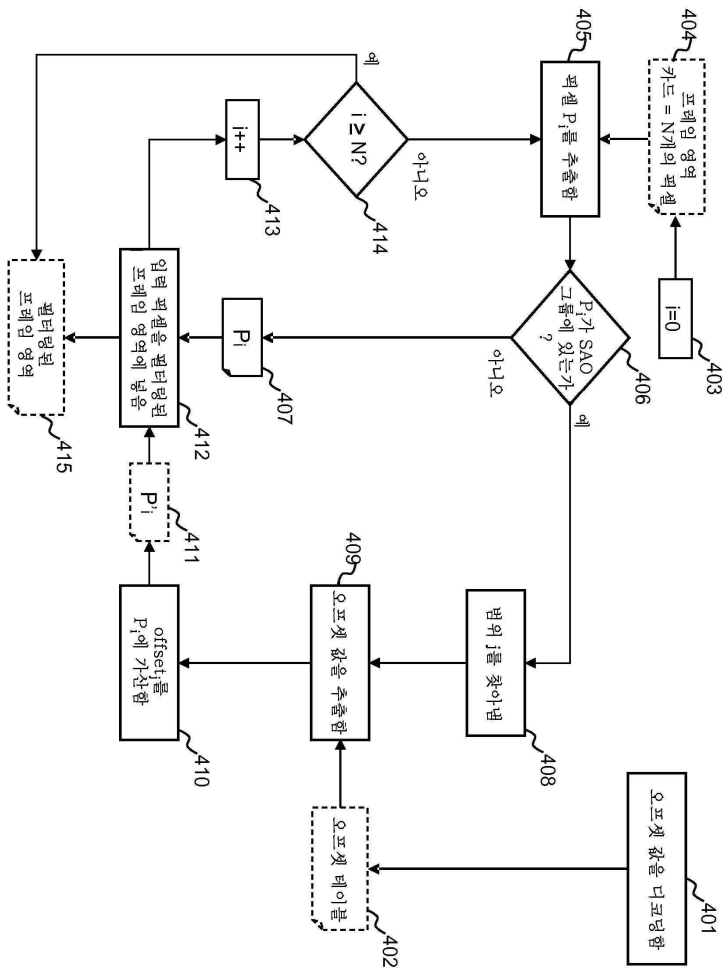


도면3



도면4

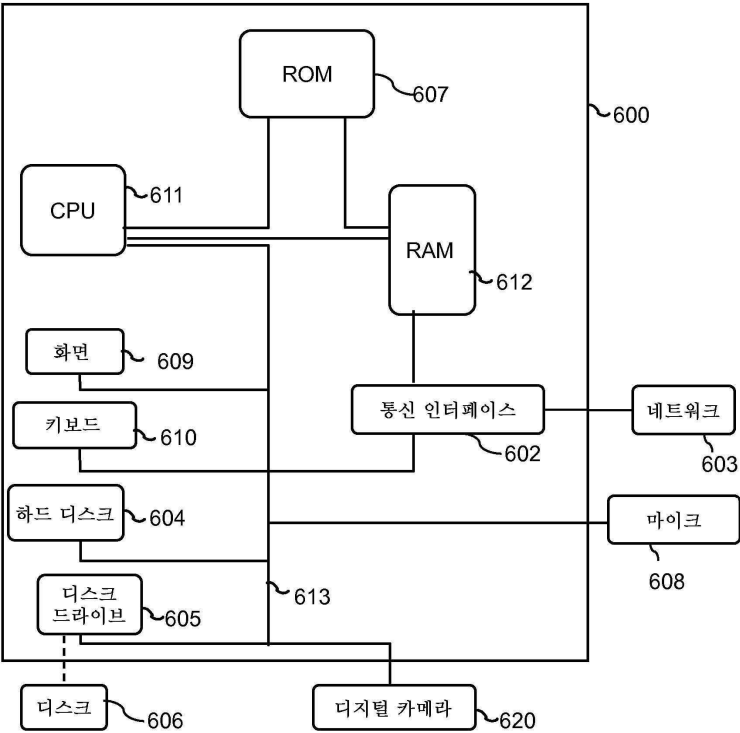
60



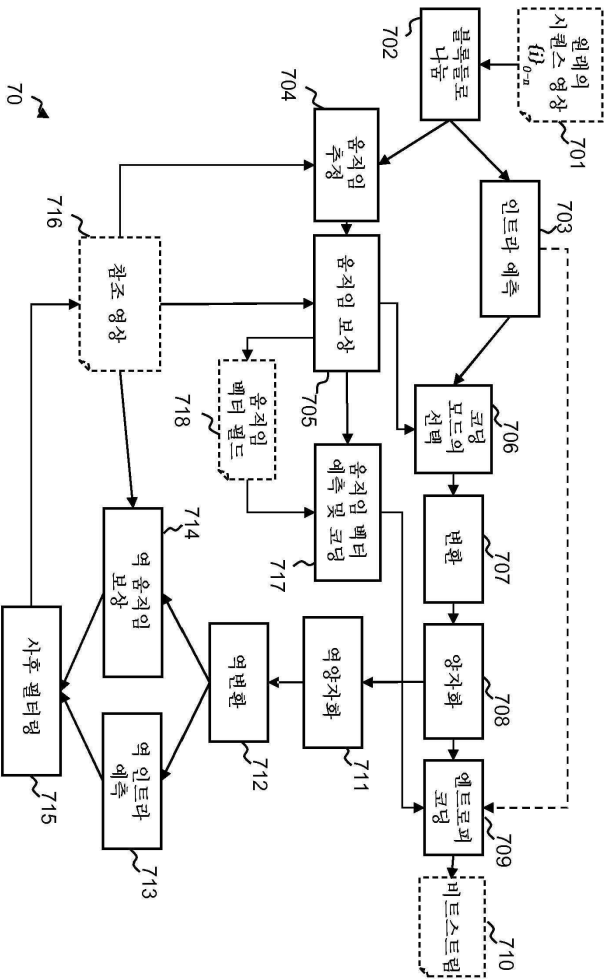
도면5



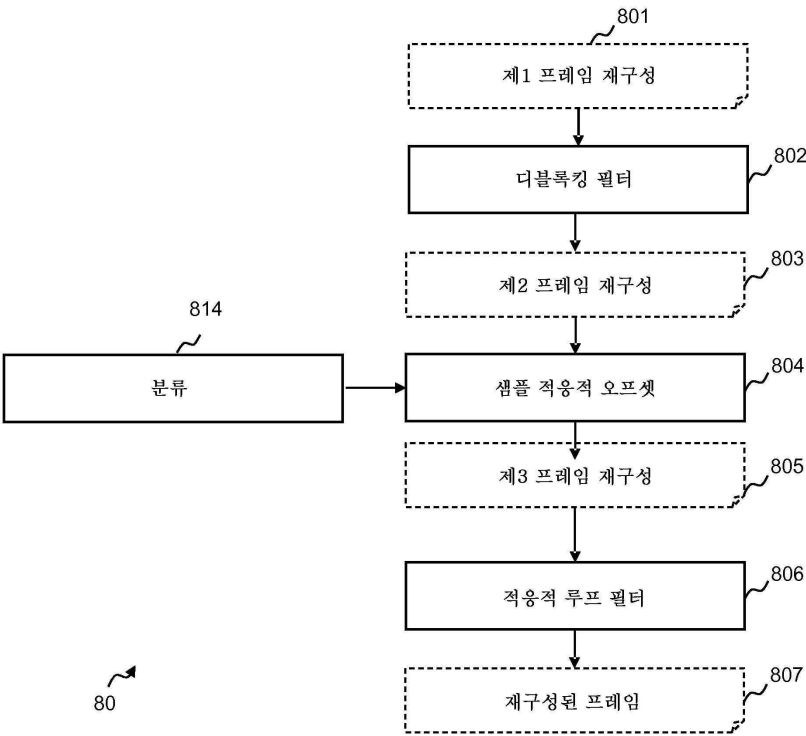
도면6



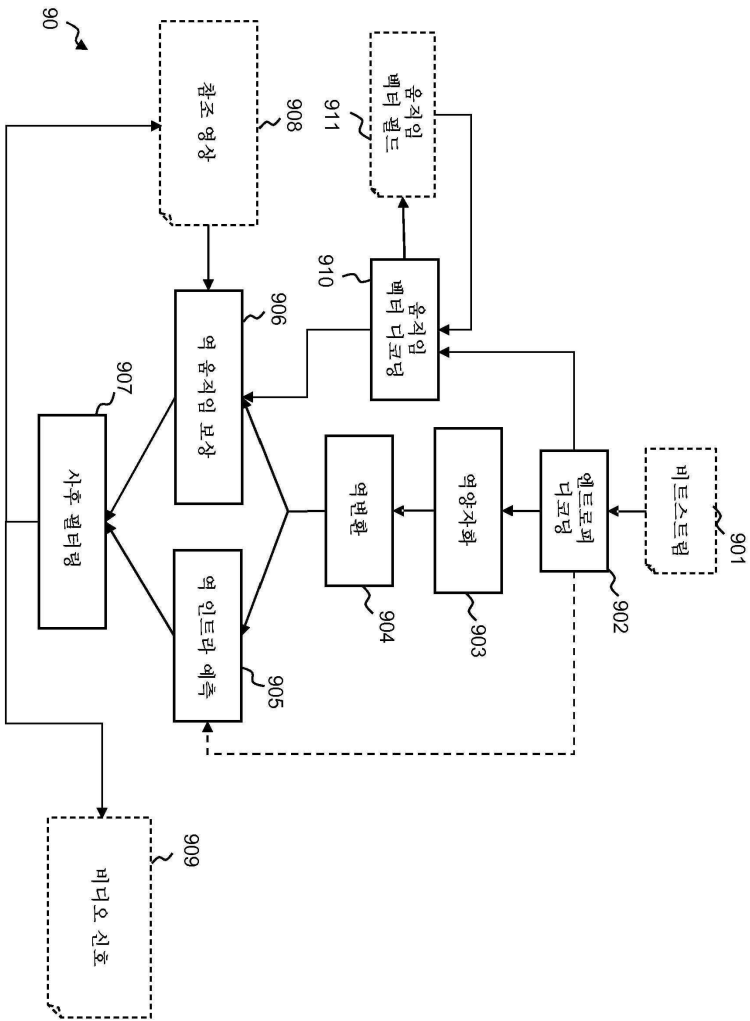
도면7



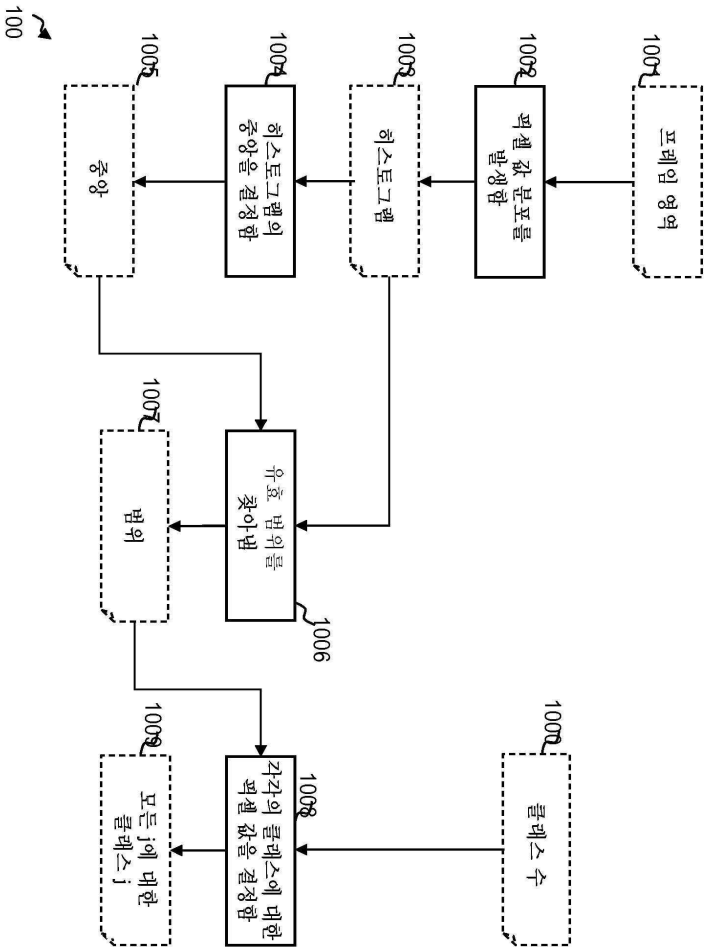
도면8



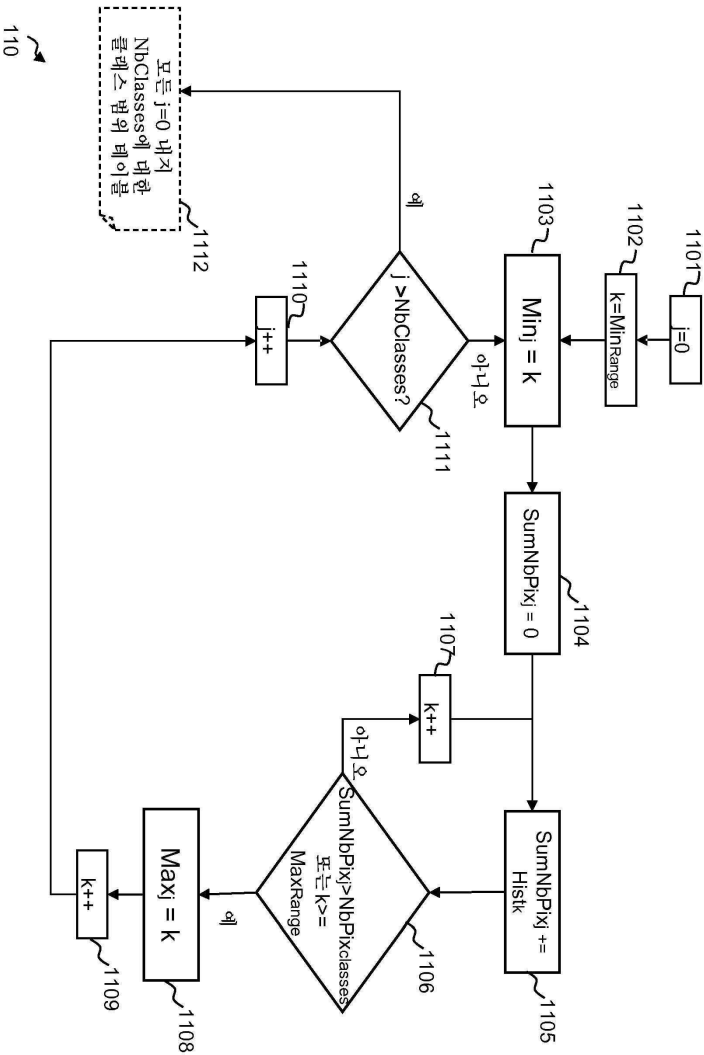
도면9



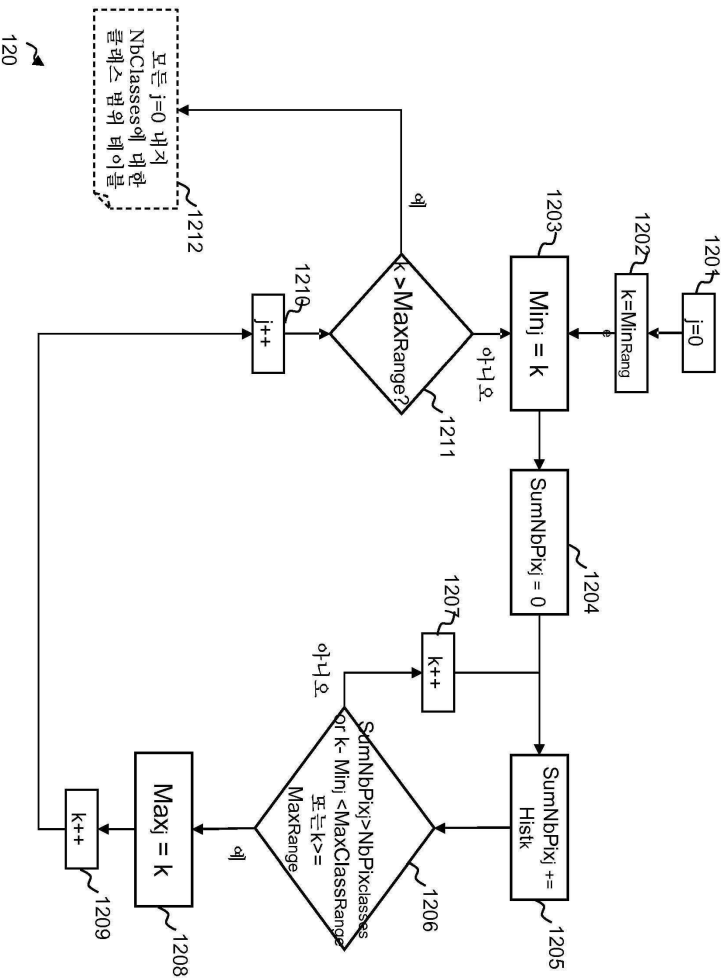
도면10



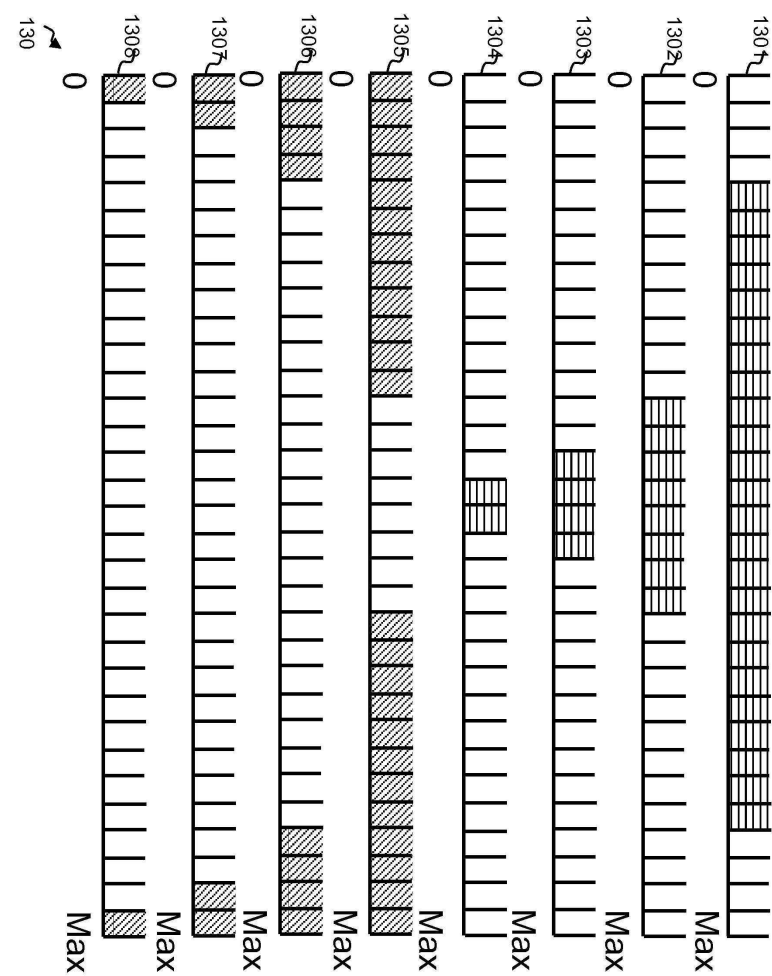
도면11



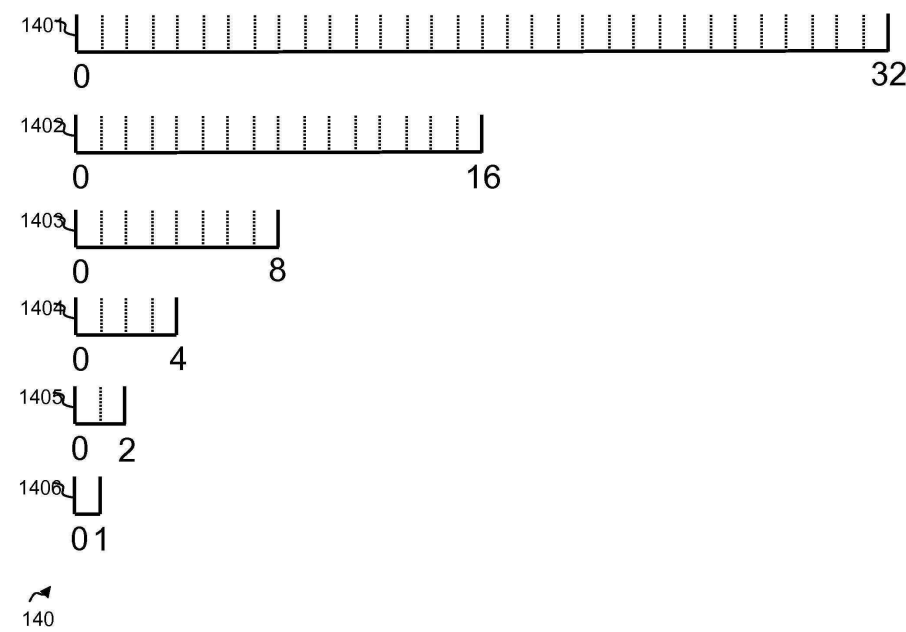
도면12



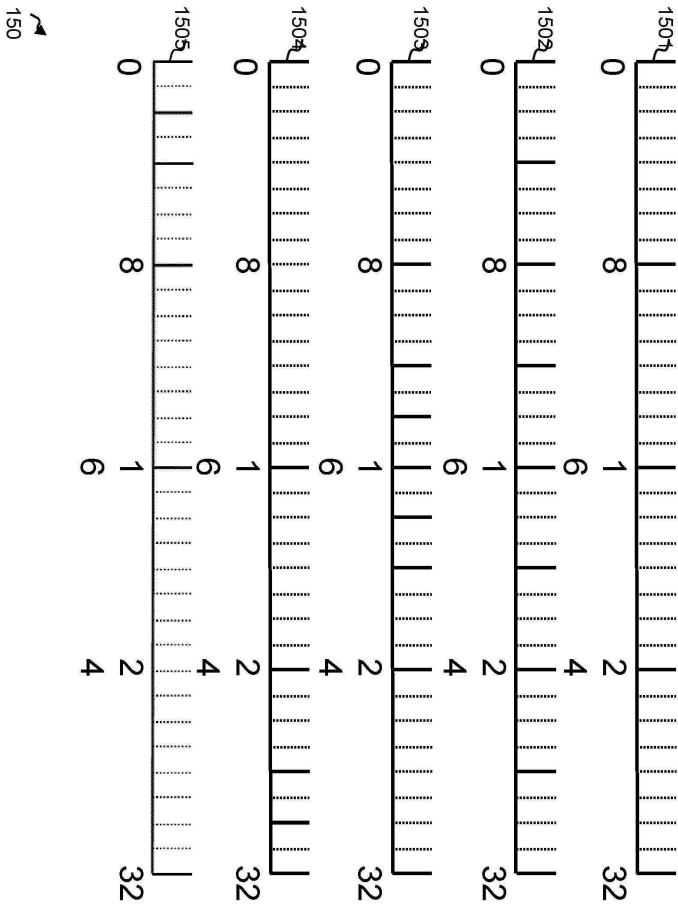
도면13

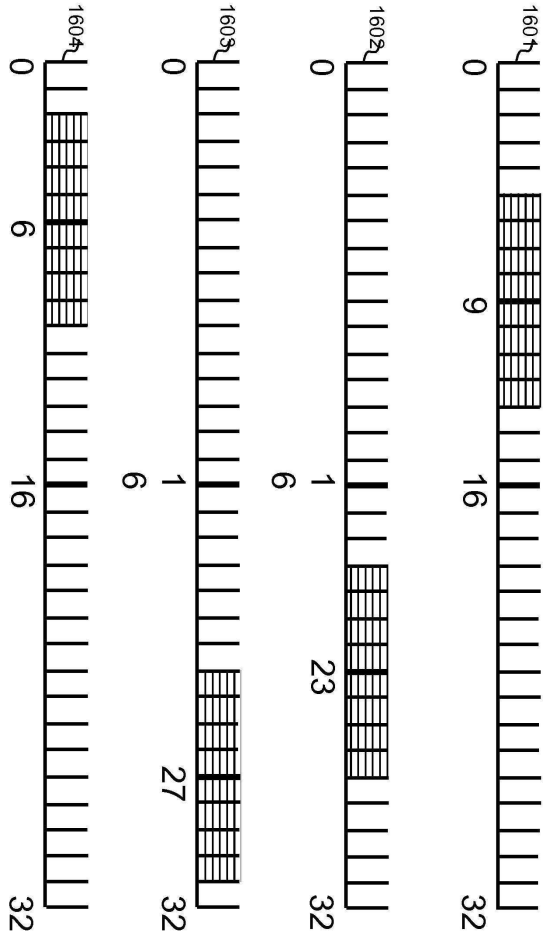


도면14



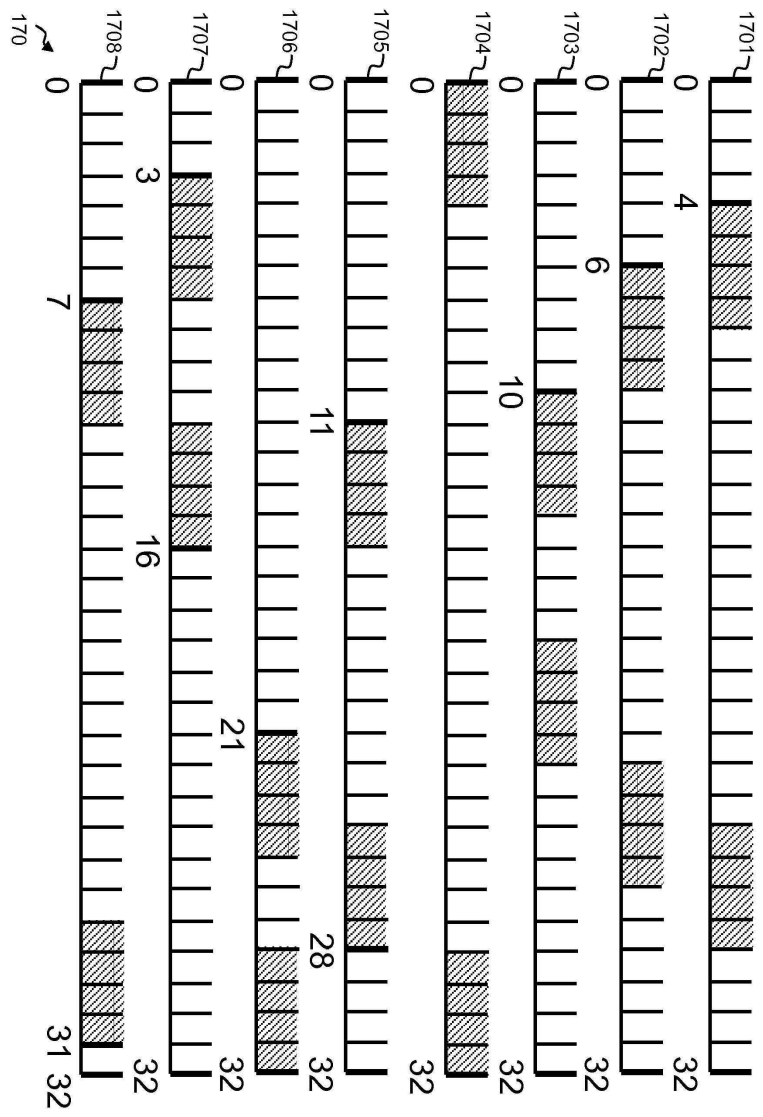
도면15





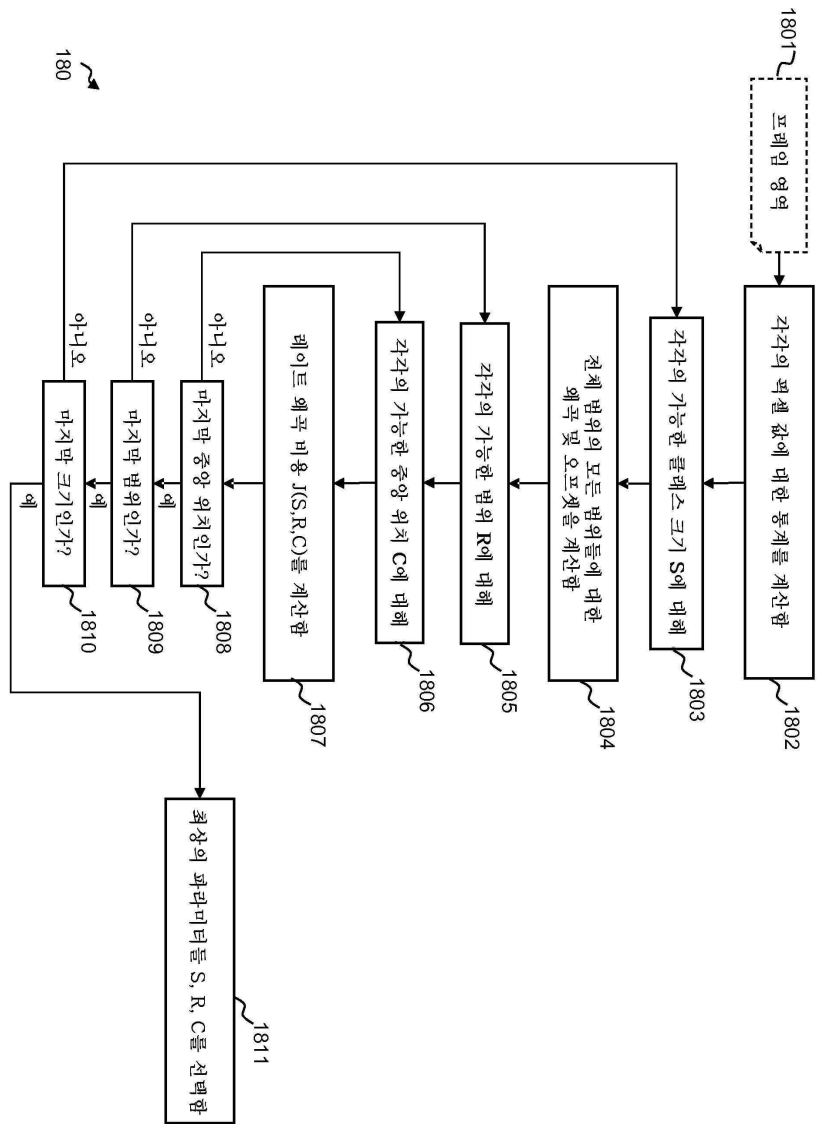
도면16

160

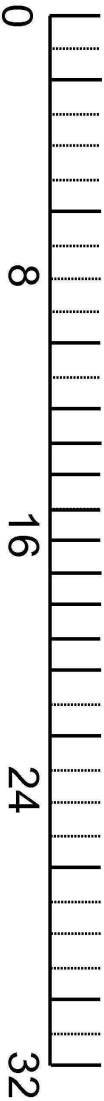


도면17

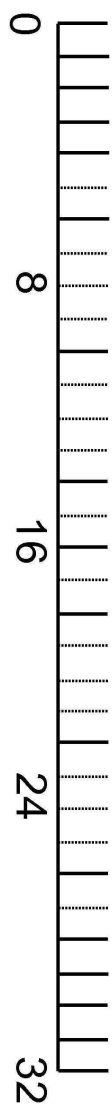
도면18



도면19a



도면19b



도면20a

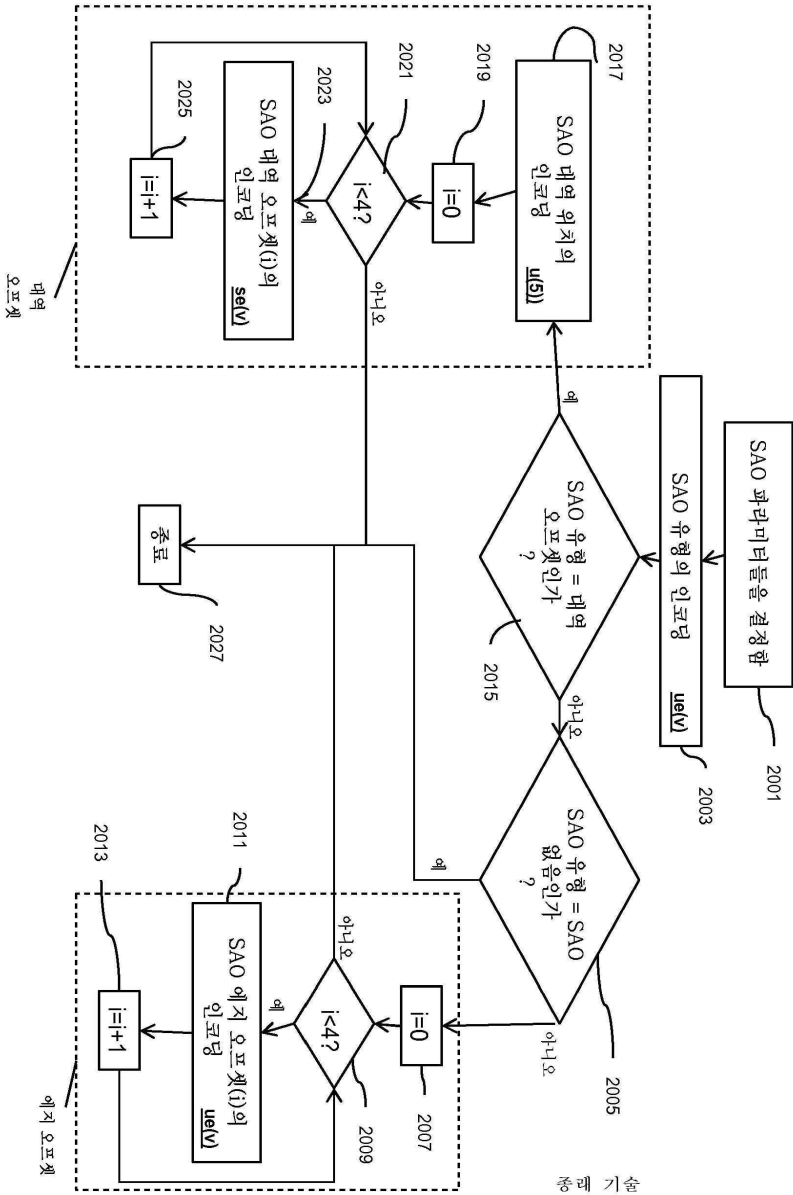
sao_offset_vlc( rx, ry, cldx ) {	서술자
sao_type_idx[ cldx ][ rx ][ ry ]	ue(v)
if( sao_type_idx[ cldx ][ rx ][ ry ] == 5 ) {	
sao_band_position[ cldx ][ rx ][ ry ]	u(5)
for( i = 0; i < 4; i++ )	
sao_offset[ cldx ][ rx ][ ry ][ i ]	se(v)
}	
else if( sao_type_idx[ cldx ][ rx ][ ry ] != 0 )	
for( i = 0; i < 4; i++ )	
sao_offset[ cldx ][ rx ][ ry ][ i ]	ue(v)
}	

종래 기술  
(prior art)

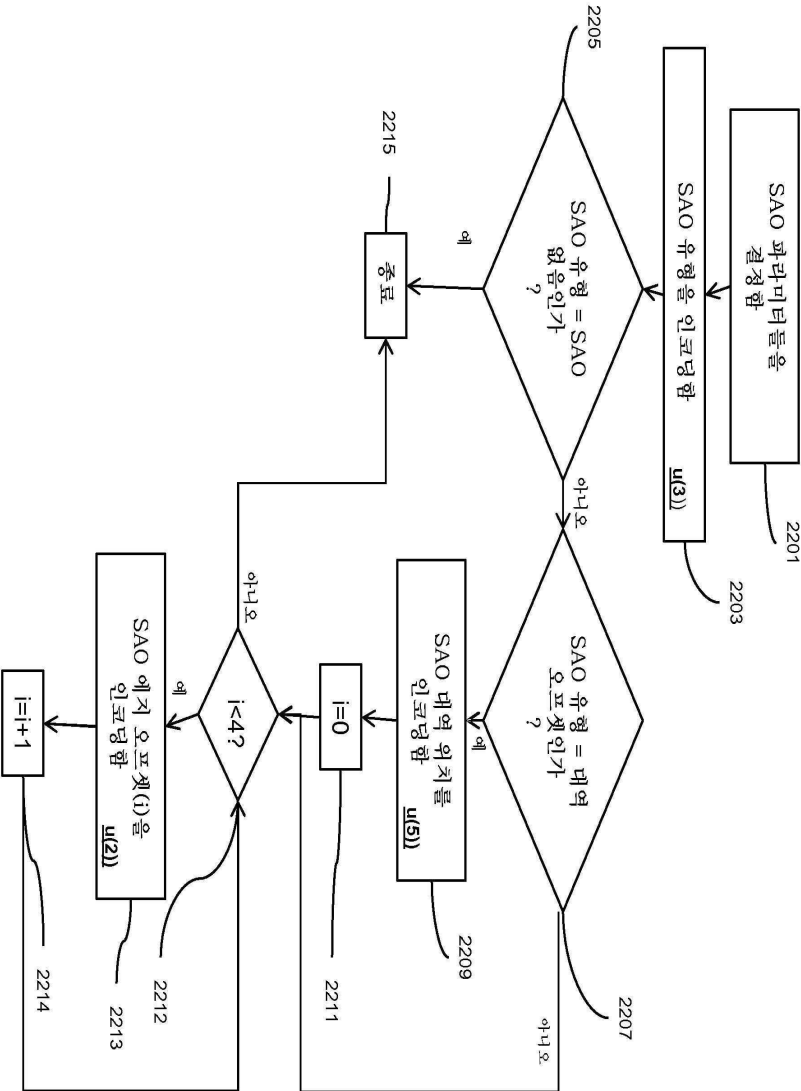
도면20b

sao_offset_vlc( rx, ry, cldx ) {	서술자
sao_type_idx[ cldx ][ rx ][ ry ]	u(3)
if( sao_type_idx[ cldx ][ rx ][ ry ] != 0 ) {	
if( sao_type_idx[ cldx ][ rx ][ ry ] == 5 ) {	
sao_band_position[ cldx ][ rx ][ ry ]	u(5)
}	
for( i = 0; i < 4; i++ )	
sao_offset[ cldx ][ rx ][ ry ][ i ]	u(2)
}	
}	

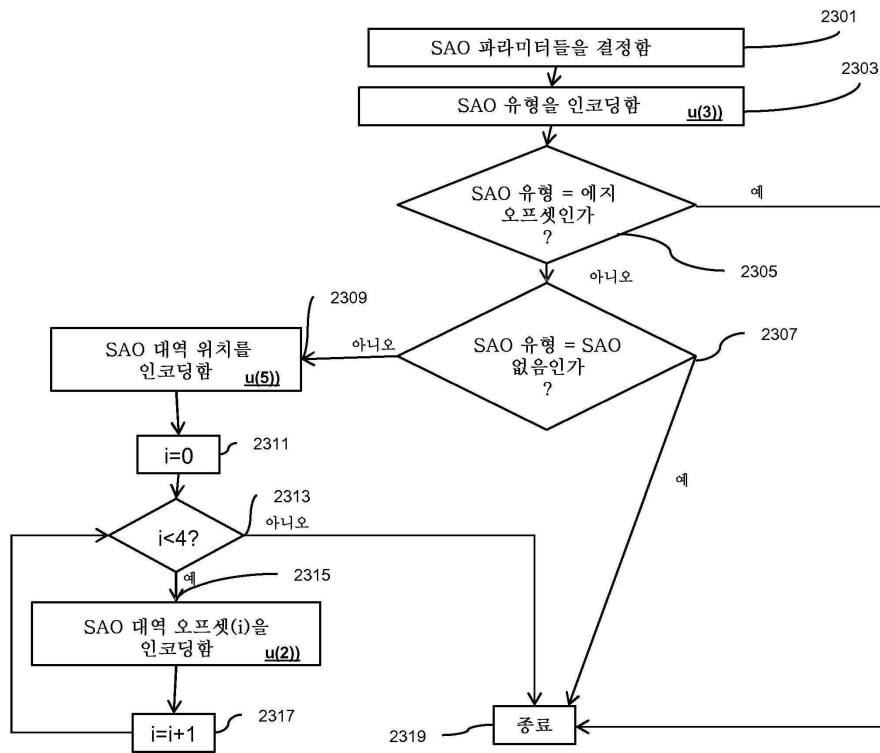
도면21



도면22



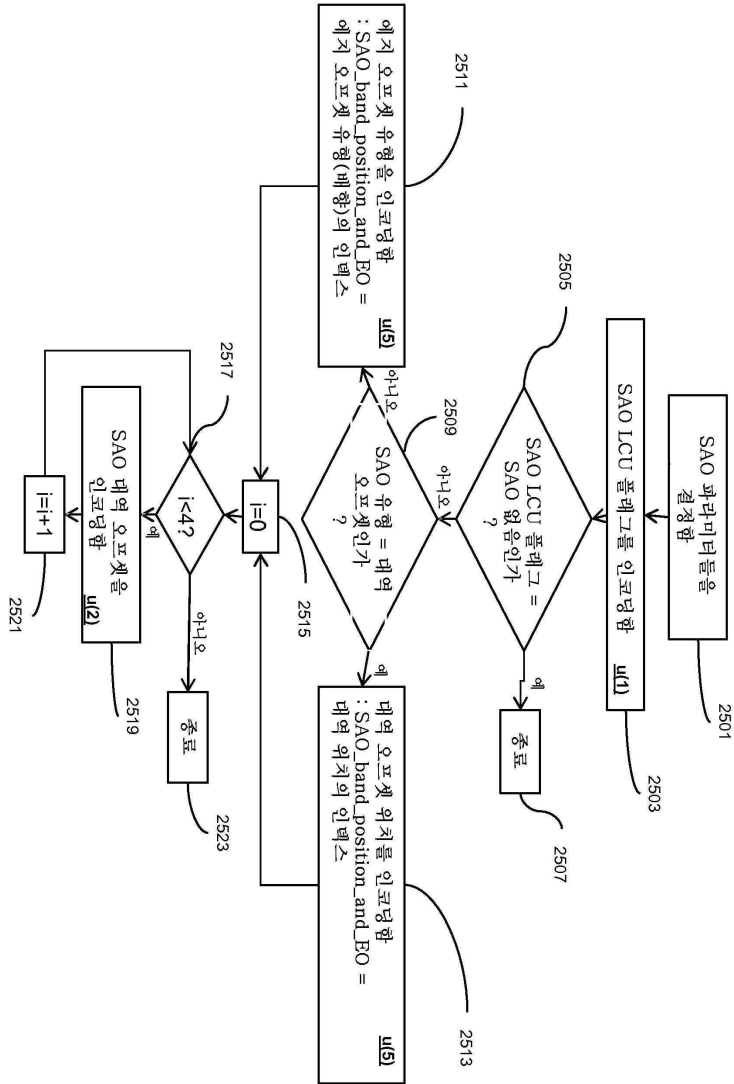
도면23



도면24

sao_offset_vlc( rx, ry, cldx ) {	서술자
sao_lcu_flag[ cldx ][ rx ][ ry ]	u(1)
if(sao_lcu_flag[ cldx ][ rx ][ ry ]){	
sao_band_position_and_EO[ cldx ][ rx ][ ry ]	u(5)
}	
for( i = 0; i < 4; i++ )	
sao_offset[ cldx ][ rx ][ ry ][ i ]	u(2)
}	
}	

도면25



도면26

