



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2018-0056688
(43) 공개일자 2018년05월29일

- (51) 국제특허분류(Int. Cl.)
 HO4N 19/119 (2014.01) HO4N 19/105 (2014.01)
 HO4N 19/11 (2014.01) HO4N 19/146 (2014.01)
 HO4N 19/154 (2014.01) HO4N 19/156 (2014.01)
 HO4N 19/167 (2014.01) HO4N 19/176 (2014.01)
 HO4N 19/467 (2014.01) HO4N 19/593 (2014.01)
- (52) CPC특허분류
 HO4N 19/119 (2015.01)
 HO4N 19/105 (2015.01)
- (21) 출원번호 10-2018-7010761
- (22) 출원일자(국제) 2016년09월15일
 심사청구일자 없음
- (85) 번역문제출일자 2018년04월16일
- (86) 국제출원번호 PCT/US2016/051987
- (87) 국제공개번호 WO 2017/048991
 국제공개일자 2017년03월23일
- (30) 우선권주장
 62/220,699 2015년09월18일 미국(US)
 (뒷면에 계속)
- (71) 출원인
 쉘컴 인코포레이티드
 미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
- (72) 발명자
 제이콥슨 네이탄 해임
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
 티루말라이 비자야라가반
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
 조쉬 라잔 렉스맨
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
- (74) 대리인
 특허법인코리아나

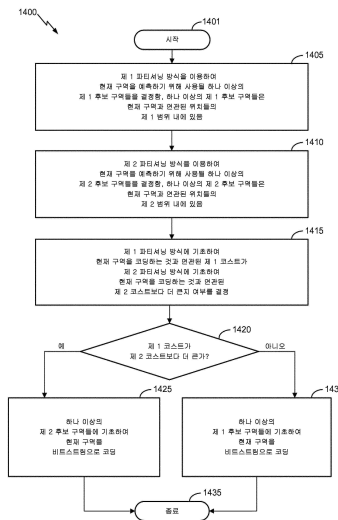
전체 청구항 수 : 총 30 항

(54) 발명의 명칭 디스플레이 스트림 압축 (DSC) 에 대한 블록 예측 모드를 위한 가변 파티션 사이즈

(57) 요약

디스플레이 링크들을 통한 송신을 위해 고정 비트레이트 비디오 코딩 방식의 블록 예측 모드에서 비디오 데이터의 블록을 코딩하는 방법이 개시된다. 하나의 양태에서, 이 방법은, 제 1 파티셔닝 방식을 이용하여 비디오 데이터의 블록 내의 현재 구역을 예측하기 위해 사용될 하나 이상의 제 1 후보 구역들을 결정하는 단계, 제 2 파티셔닝 방식을 이용하여 현재 구역을 예측하기 위해 사용될 하나 이상의 제 2 후보 구역들을 결정하는 단계, 제 1 파티셔닝 방식을 이용하여 현재 구역을 코딩하는 것과 연관된 제 1 코스트가 제 2 파티셔닝 방식을 이용하여 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 큰 것을 결정하는 단계, 및, 제 2 파티셔닝 방식을 이용하여 현재 구역을 코딩하는 단계를 포함한다.

대표도 - 도14



(52) CPC특허분류

HOAN 19/11 (2015.01)
HOAN 19/146 (2015.01)
HOAN 19/154 (2015.01)
HOAN 19/156 (2015.01)
HOAN 19/167 (2015.01)
HOAN 19/176 (2015.01)
HOAN 19/467 (2015.01)
HOAN 19/593 (2015.01)

(30) 우선권주장

62/244,690 2015년10월21일 미국(US)
15/265,743 2016년09월14일 미국(US)

명세서

청구범위

청구항 1

고정 비트레이트 비디오 코딩 방식의 블록 예측 모드에서 비디오 데이터의 블록을 코딩하는 방법으로서,

블록 예측 모드와 연관된 제 1 파티셔닝 방식에 기초하여 상기 비디오 데이터의 블록 내의 현재 구역을 예측하기 위해 사용될 하나 이상의 제 1 후보 구역들을 결정하는 단계로서, 상기 하나 이상의 제 1 후보 구역들은 상기 현재 구역과 연관된 위치들의 제 1 범위 내에 있고, 상기 하나 이상의 제 1 후보 구역들은 비디오 인코딩 디바이스의 메모리에 저장되는, 상기 하나 이상의 제 1 후보 구역들을 결정하는 단계;

상기 블록 예측 모드와 연관된 제 2 파티셔닝 방식에 기초하여 상기 현재 구역을 예측하기 위해 사용될 하나 이상의 제 2 후보 구역들을 결정하는 단계로서, 상기 하나 이상의 제 2 후보 구역들은 상기 현재 구역과 연관된 위치들의 제 2 범위 내에 있고, 상기 하나 이상의 제 2 후보 구역들은 상기 비디오 인코딩 디바이스의 상기 메모리에 저장되는, 상기 하나 이상의 제 2 후보 구역들을 결정하는 단계;

상기 제 1 파티셔닝 방식에 기초하여 상기 현재 구역을 코딩하는 것과 연관된 제 1 코스트가 상기 제 2 파티셔닝 방식에 기초하여 상기 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 큰지 여부를 결정하는 단계; 및

상기 제 1 코스트가 상기 제 2 코스트보다 더 크다고 결정하는 것에 응답하여, 상기 하나 이상의 제 2 후보 구역들에 기초하여 상기 현재 구역을 비트스트림으로 코딩하는 단계를 포함하는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 2

제 1 항에 있어서,

상기 현재 구역에 대한 상기 하나 이상의 제 2 후보 구역들의 위치를 나타내는 하나 이상의 예측 벡터들, 및 상기 하나 이상의 제 2 후보 구역들과 상기 현재 구역 사이의 차이를 나타내는 잔차를 상기 비트스트림에서 시그널링하는 단계를 더 포함하는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 3

제 2 항에 있어서,

상기 잔차는 루마 잔차 및 크로마 잔차를 포함하고, 상기 방법은, (i) 제 1 수의 엔트로피 코딩 그룹들을 이용하여 상기 루마 잔차에 대해 엔트로피 코딩을 수행하는 것, 및 (ii) 상기 제 1 수와는 상이한 제 2 수의 엔트로피 코딩 그룹들을 이용하여 상기 크로마 잔차에 대해 엔트로피 코딩을 수행하는 것을 더 포함하는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 4

제 2 항에 있어서,

상기 블록 내의 적어도 하나의 다른 구역에 대한 하나 이상의 제 1 후보 구역들의 위치를 나타내는 하나 이상의 예측 벡터들을 상기 비트스트림에서 시그널링하는 단계를 더 포함하고, 상기 적어도 하나의 다른 구역에 대해 시그널링되는 예측 벡터들의 수는 상기 현재 구역에 대해 시그널링되는 예측 벡터들의 수와는 상이한, 비디오 데이터의 블록을 코딩하는 방법.

청구항 5

제 1 항에 있어서,

상기 하나 이상의 제 1 후보 구역들 및 상기 하나 이상의 제 2 후보 구역들은 동일한 후보 구역 또는 구역들을 포함하는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 6

제 1 항에 있어서,

(i) 상기 현재 구역과 상기 하나 이상의 제 2 후보 구역들 사이의 절대 차이들의 합 및 (ii) 상기 하나 이상의 예측 벡터들 및 대응하는 잔차들을 상기 비트스트림에서 시그널링하기 위해 필요한 비트들의 수에 적어도 부분적으로 기초하여 상기 제 2 코스트를 결정하는 단계를 더 포함하는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 7

제 1 항에 있어서,

파티션 표시자를 상기 비트스트림에서 시그널링하는 단계를 더 포함하고, 상기 파티션 표시자는 상기 블록 내의 각각의 구역과 연관된 파티셔닝 방식을 나타내고, 상기 블록은 상기 현재 구역 이외의 적어도 하나의 구역을 포함하는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 8

제 7 항에 있어서,

상기 파티션 표시자는, 현재 구역이 상기 제 2 파티셔닝 방식과 연관되는 것, 및 상기 블록에서의 상기 현재 구역 이외의 상기 적어도 하나의 구역이 상기 제 2 파티셔닝 방식과는 상이한 상기 제 1 파티셔닝 방식과 연관되는 것을 표시하는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 9

제 1 항에 있어서,

상기 하나 이상의 제 1 후보 구역들 및 상기 하나 이상의 제 2 후보 구역들은 상이한 사이즈들을 갖는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 10

제 1 항에 있어서,

상기 하나 이상의 제 1 후보 구역들 및 상기 하나 이상의 제 2 후보 구역들은 동일한 사이즈들을 갖는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 11

제 1 항에 있어서,

상기 비디오 데이터의 상기 블록은 4:2:0 크로마 서브-샘플링 포맷 또는 4:2:2 크로마 서브-샘플링 포맷 중 하나와 연관되는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 12

제 1 항에 있어서,

상기 하나 이상의 제 2 후보 구역들은, 제 1 수의 크로마 샘플들을 갖는 제 1 구역, 및 상기 제 1 수와는 상이한 제 2 수의 크로마 샘플들을 갖는 제 2 구역을 포함하고, 상기 제 1 및 제 2 구역들은 중첩하지 않는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 13

제 1 항에 있어서,

상기 하나 이상의 제 2 후보 구역들은 어떤 크로마 샘플들과도 연관되지 않는 적어도 하나의 구역을 포함하는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 14

제 1 항에 있어서,

상기 현재 구역은, 제 1 수의 크로마 샘플들과 연관된 제 1 현재 구역, 및 상기 제 1 수와는 상이한 제 2 수의 크로마 샘플들과 연관된 제 2 현재 구역을 포함하고, 상기 하나 이상의 제 2 후보 구역들은, 상기 제 1 수의 크로마 샘플들을 포함하는 상기 하나 이상의 제 2 후보 구역들의 제 1 의 구역, 및 상기 제 2 수의 크로마 샘플들을 포함하는 상기 하나 이상의 제 2 후보 구역들의 제 2 의 구역을 포함하는, 비디오 데이터의 블록을 코딩하는 방법.

청구항 15

고정 비트레이트 비디오 코딩 방식의 블록 예측 모드에서 비디오 데이터의 블록을 코딩하도록 구성된 장치로서, 상기 장치는,

하나 이상의 후보 구역들과 연관된 비디오 데이터를 저장하도록 구성된 메모리;

상기 메모리와 통신하는 하나 이상의 프로세서들을 포함하고,

상기 하나 이상의 프로세서들은,

블록 예측 모드와 연관된 제 1 파티셔닝 방식에 기초하여 상기 비디오 데이터의 블록 내의 현재 구역을 예측하기 위해 사용될 하나 이상의 제 1 후보 구역들을 결정하는 것으로서, 상기 하나 이상의 제 1 후보 구역들은 상기 현재 구역과 연관된 위치들의 제 1 범위 내에 있는, 상기 하나 이상의 제 1 후보 구역들을 결정하는 것을 행하고;

상기 블록 예측 모드와 연관된 제 2 파티셔닝 방식에 기초하여 상기 현재 구역을 예측하기 위해 사용될 하나 이상의 제 2 후보 구역들을 결정하는 것으로서, 상기 하나 이상의 제 2 후보 구역들은 상기 현재 구역과 연관된 위치들의 제 2 범위 내에 있는, 상기 하나 이상의 제 2 후보 구역들을 결정하는 것을 행하며;

상기 제 1 파티셔닝 방식에 기초하여 상기 현재 구역을 코딩하는 것과 연관된 제 1 코스트가 상기 제 2 파티셔닝 방식에 기초하여 상기 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 큰지 여부를 결정하고; 그리고 상기 제 1 코스트가 상기 제 2 코스트보다 더 크다고 결정하는 것에 응답하여, 상기 하나 이상의 제 2 후보 구역들에 기초하여 상기 현재 구역을 비트스트림으로 코딩하도록 구성되는, 비디오 데이터의 블록을 코딩하도록 구성된 장치.

청구항 16

제 15 항에 있어서,

상기 하나 이상의 프로세서들은, 상기 현재 구역에 대한 상기 하나 이상의 제 2 후보 구역들의 위치를 나타내는 하나 이상의 예측 벡터들, 및 상기 하나 이상의 제 2 후보 구역들과 상기 현재 구역 사이의 차이를 나타내는 잔차를 상기 비트스트림에서 시그널링하도록 더 구성되는, 비디오 데이터의 블록을 코딩하도록 구성된 장치.

청구항 17

제 16 항에 있어서,

상기 잔차는 루마 잔차 및 크로마 잔차를 포함하고, 방법은, (i) 제 1 수의 엔트로피 코딩 그룹들을 이용하여 상기 루마 잔차에 대해 엔트로피 코딩을 수행하는 것, 및 (ii) 상기 제 1 수와는 상이한 제 2 수의 엔트로피 코딩 그룹들을 이용하여 상기 크로마 잔차에 대해 엔트로피 코딩을 수행하는 것을 더 포함하는, 비디오 데이터의 블록을 코딩하도록 구성된 장치.

청구항 18

제 16 항에 있어서,

상기 하나 이상의 프로세서들은, 상기 블록 내의 적어도 하나의 다른 구역에 대한 하나 이상의 제 1 후보 구역들의 위치를 나타내는 하나 이상의 예측 벡터들을 상기 비트스트림에서 시그널링하도록 더 구성되고, 상기 적어도 하나의 다른 구역에 대해 시그널링되는 예측 벡터들의 수는 상기 현재 구역에 대해 시그널링되는 예측 벡터들의 수와는 상이한, 비디오 데이터의 블록을 코딩하도록 구성된 장치.

청구항 19

제 15 항에 있어서,

상기 하나 이상의 제 1 후보 구역들 및 상기 하나 이상의 제 2 후보 구역들은 동일한 후보 구역 또는 구역들을 포함하는, 비디오 데이터의 블록을 코딩하도록 구성된 장치.

청구항 20

제 15 항에 있어서,

상기 하나 이상의 프로세서들은, (i) 상기 현재 구역과 상기 하나 이상의 제 2 후보 구역들 사이의 절대 차이들의 합 및 (ii) 상기 하나 이상의 예측 벡터들 및 대응하는 잔차들을 상기 비트스트림에서 시그널링하기 위해 필요한 비트들의 수에 적어도 부분적으로 기초하여 상기 제 2 코스트를 결정하도록 더 구성되는, 비디오 데이터의 블록을 코딩하도록 구성된 장치.

청구항 21

제 15 항에 있어서,

상기 하나 이상의 프로세서들은, 파티션 표시자를 상기 비트스트림에서 시그널링하도록 더 구성되고, 상기 파티션 표시자는 상기 블록 내의 각각의 구역과 연관된 파티셔닝 방식을 나타내고, 상기 블록은 상기 현재 구역 이외의 적어도 하나의 구역을 포함하며, 상기 파티션 표시자는, 현재 구역이 상기 제 2 파티셔닝 방식과 연관되는 것, 및 상기 블록에서의 상기 현재 구역 이외의 상기 적어도 하나의 구역이 상기 제 2 파티셔닝 방식과는 상이한 상기 제 1 파티셔닝 방식과 연관되는 것을 표시하는, 비디오 데이터의 블록을 코딩하도록 구성된 장치.

청구항 22

제 15 항에 있어서,

상기 하나 이상의 제 2 후보 구역들은, 제 1 수의 크로마 샘플들을 갖는 제 1 구역, 및 상기 제 1 수와는 상이한 제 2 수의 크로마 샘플들을 갖는 제 2 구역을 포함하고, 상기 제 1 및 제 2 구역들은 중첩하지 않는, 비디오 데이터의 블록을 코딩하도록 구성된 장치.

청구항 23

제 15 항에 있어서,

상기 하나 이상의 제 2 후보 구역들은 어떤 크로마 샘플들과도 연관되지 않는 적어도 하나의 구역을 포함하는, 비디오 데이터의 블록을 코딩하도록 구성된 장치.

청구항 24

제 15 항에 있어서,

상기 현재 구역은, 제 1 수의 크로마 샘플들과 연관된 제 1 현재 구역, 및 상기 제 1 수와는 상이한 제 2 수의 크로마 샘플들과 연관된 제 2 현재 구역을 포함하고, 상기 하나 이상의 제 2 후보 구역들은, 상기 제 1 수의 크로마 샘플들을 포함하는 상기 하나 이상의 제 2 후보 구역들의 제 1의 구역, 및 상기 제 2 수의 크로마 샘플들을 포함하는 상기 하나 이상의 제 2 후보 구역들의 제 2의 구역을 포함하는, 비디오 데이터의 블록을 코딩하도록 구성된 장치.

청구항 25

고정 비트레이트 비디오 코딩 방식의 블록 예측 모드에서 비디오 데이터의 블록을 코딩하도록 구성된 코드를 포함하는 비-일시적 물리적 컴퓨터 스토리지로서,

상기 코드는, 실행될 때, 장치로 하여금,

블록 예측 모드와 연관된 제 1 파티셔닝 방식에 기초하여 상기 비디오 데이터의 블록 내의 현재 구역을 예측하기 위해 사용될 하나 이상의 제 1 후보 구역들을 결정하는 것으로서, 상기 하나 이상의 제 1 후보 구역들은 상기 현재 구역과 연관된 위치들의 제 1 범위 내에 있는, 상기 하나 이상의 제 1 후보 구역들을 결정하는 것을 행

하게 하고;

상기 블록 예측 모드와 연관된 제 2 파티셔닝 방식에 기초하여 상기 현재 구역을 예측하기 위해 사용될 하나 이상의 제 2 후보 구역들을 결정하는 것으로서, 상기 하나 이상의 제 2 후보 구역들은 상기 현재 구역과 연관된 위치들의 제 2 범위 내에 있는, 상기 하나 이상의 제 2 후보 구역들을 결정하는 것을 행하게 하며;

상기 제 1 파티셔닝 방식에 기초하여 상기 현재 구역을 코딩하는 것과 연관된 제 1 코스트가 상기 제 2 파티셔닝 방식에 기초하여 상기 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 큰지 여부를 결정하게 하고; 그리고

상기 제 1 코스트가 상기 제 2 코스트보다 더 크다고 결정하는 것에 응답하여, 상기 하나 이상의 제 2 후보 구역들에 기초하여 상기 현재 구역을 비트스트림으로 코딩하게 하는, 비-일시적 물리적 컴퓨터 스토리지.

청구항 26

제 25 항에 있어서,

상기 코드는 추가적으로 상기 장치로 하여금, 상기 현재 구역에 대한 상기 하나 이상의 제 2 후보 구역들의 위치를 나타내는 하나 이상의 예측 벡터들, 및 상기 하나 이상의 제 2 후보 구역들과 상기 현재 구역 사이의 차이를 나타내는 잔차를 상기 비트스트림에서 시그널링하게 하는, 비-일시적 물리적 컴퓨터 스토리지.

청구항 27

제 25 항에 있어서,

상기 코드는 추가적으로 상기 장치로 하여금, 파티션 표시자를 상기 비트스트림에서 시그널링하게 하고, 상기 파티션 표시자는 상기 블록 내의 각각의 구역과 연관된 파티셔닝 방식을 나타내고, 상기 블록은 상기 현재 구역 이외의 적어도 하나의 구역을 포함하며, 상기 파티션 표시자는, 현재 구역이 상기 제 2 파티셔닝 방식과 연관되는 것, 및 상기 블록에서의 상기 현재 구역 이외의 상기 적어도 하나의 구역이 상기 제 2 파티셔닝 방식과는 상이한 상기 제 1 파티셔닝 방식과 연관되는 것을 표시하는, 비-일시적 물리적 컴퓨터 스토리지.

청구항 28

고정 비트레이트 비디오 코딩 방식의 블록 예측 모드에서 비디오 데이터의 블록을 코딩하도록 구성된 비디오 코딩 디바이스로서,

블록 예측 모드와 연관된 제 1 파티셔닝 방식에 기초하여 상기 비디오 데이터의 블록 내의 현재 구역을 예측하기 위해 사용될 하나 이상의 제 1 후보 구역들을 결정하는 수단으로서, 상기 하나 이상의 제 1 후보 구역들은 상기 현재 구역과 연관된 위치들의 제 1 범위 내에 있는, 상기 하나 이상의 제 1 후보 구역들을 결정하는 수단;

상기 블록 예측 모드와 연관된 제 2 파티셔닝 방식에 기초하여 상기 현재 구역을 예측하기 위해 사용될 하나 이상의 제 2 후보 구역들을 결정하는 수단으로서, 상기 하나 이상의 제 2 후보 구역들은 상기 현재 구역과 연관된 위치들의 제 2 범위 내에 있는, 상기 하나 이상의 제 2 후보 구역들을 결정하는 수단;

상기 제 1 파티셔닝 방식에 기초하여 상기 현재 구역을 코딩하는 것과 연관된 제 1 코스트가 상기 제 2 파티셔닝 방식에 기초하여 상기 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 큰지 여부를 결정하는 수단; 및

상기 제 1 코스트가 상기 제 2 코스트보다 더 크다고 결정하는 것에 응답하여, 상기 하나 이상의 제 2 후보 구역들에 기초하여 상기 현재 구역을 비트스트림으로 코딩하는 수단을 포함하는, 비디오 코딩 디바이스.

청구항 29

제 28 항에 있어서,

상기 현재 구역에 대한 상기 하나 이상의 제 2 후보 구역들의 위치를 나타내는 하나 이상의 예측 벡터들, 및 상기 하나 이상의 제 2 후보 구역들과 상기 현재 구역 사이의 차이를 나타내는 잔차를 상기 비트스트림에서 시그널링하는 수단을 더 포함하는, 비디오 코딩 디바이스.

청구항 30

제 28 항에 있어서,

파티션 표시자를 상기 비트스트림에서 시그널링하는 수단을 더 포함하고, 상기 파티션 표시자는 상기 블록 내의 각각의 구역과 연관된 파티셔닝 방식을 나타내고, 상기 블록은 상기 현재 구역 이외의 적어도 하나의 구역을 포함하며, 상기 파티션 표시자는, 현재 구역이 상기 제 2 파티셔닝 방식과 연관되는 것, 및 상기 블록에서의 상기 현재 구역 이외의 상기 적어도 하나의 구역이 상기 제 2 파티셔닝 방식과는 상이한 상기 제 1 파티셔닝 방식과 연관되는 것을 표시하는, 비디오 코딩 디바이스.

발명의 설명

기술 분야

[0001] 이 개시물은 비디오 코딩 및 압축, 특히, 디스플레이 링크 비디오 압축과 같이, 디스플레이 링크들을 통한 송신을 위한 비디오 압축에 관한 것이다.

배경 기술

[0002] 디지털 비디오 기능들은, 디지털 텔레비전들, 개인 정보 단말 (personal digital assistant; PDA) 들, 랩톱 컴퓨터들, 데스크톱 모니터들, 디지털 카메라들, 디지털 레코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게임용 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 라디오 전화들, 화상 원격회의 디바이스들 등등을 포함하는 광범위한 디스플레이들 내로 편입될 수 있다. 디스플레이 링크들은 디스플레이들을 적합한 소스 디바이스들에 접속하기 위하여 이용된다. 디스플레이 링크들의 대역폭 요건들은 디스플레이들의 해상도에 비례하고, 이에 따라, 고해상도 디스플레이들은 큰 대역폭 디스플레이 링크들을 요구한다. 일부 디스플레이 링크들은 고해상도 디스플레이들을 지원하기 위한 대역폭을 가지지 않는다. 비디오 압축은 더 낮은 대역폭의 디스플레이 링크들이 디지털 비디오를 고해상도 디스플레이들에 제공하기 위하여 이용될 수 있도록, 대역폭 요건들을 감소시키기 위하여 이용될 수 있다.

[0003] 픽셀 데이터에 관한 이미지 압축을 수반하는 코딩 방식들이 존재한다. 그러나, 이러한 방식들은 때때로 시각적으로 무손실이 아니거나, 기존의 디스플레이 디바이스들에서 구현하기가 어려울 수 있고 고가일 수 있다.

[0004] 비디오 전자 표준 협회 (Video Electronics Standards Association; VESA) 는 디스플레이 스트림 압축 (Display Stream Compression; DSC) 을 디스플레이 링크 비디오 압축을 위한 표준으로서 개발하였다. DSC 와 같은 디스플레이 링크 비디오 압축 기법은 무엇보다도, 시각적으로 무손실인 픽처 품질 (즉, 사용자들이 압축이 활성화된 것으로 말할 수 없도록 하는 품질의 레벨을 가지는 픽처들) 을 제공해야 한다. 디스플레이 링크 비디오 압축 기법은 또한, 기존의 하드웨어로 실시간으로 구현하기가 용이하고 덜 고가인 방식을 제공해야 한다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0005] 디스플레이 스트림 압축 (DSC) 표준은 비디오 데이터의 각각의 블록이 인코더에 의해 인코딩되고, 그리고 유사하게, 디코더에 의해 디코딩될 수도 있는 다수의 코딩 모드들을 포함한다. 일부 구현들에서, 인코더 및/또는 디코더는 이전에 코딩된 블록에 기초하여 코딩될 현재 블록을 예측할 수도 있다.

[0006] 하지만, 기존의 코딩 모드들 (예컨대, 변환 코딩, 차동 펄스-코드 변조 등) 은 비디오 데이터에서 고도로 복잡한 구역들을 압축하는 만족할 만한 방식을 제공하지 않는다. 종종, 이러한 유형의 데이터 (즉, 고도로 압축된 비디오 데이터) 에 대해, 코딩될 현재 블록 (또는 현재 블록의 구성적 서브-블록들) 은 코더 (예컨대, 인코더 또는 디코더) 에 의해 조우되었던 이전 블록들에 대한 콘텐츠와 유사하다. 하지만, 기존의 인트라 예측은 이러한 현재 블록의 만족스러운 예측 (현재 블록에 충분히 유사하고 따라서 충분히 작은 잔차를 산출할 현재 블록의 예측) 을 제공하기에는 너무 제한적일 수도 있다. 따라서, 비디오 데이터의 블록들을 코딩하는 향상된 방법이 요망된다.

- [0007] 이 개시물의 시스템들, 방법들, 및 디바이스들은 각각 몇몇 혁신적인 양태들을 가지며, 그 단 하나가 본원에서 개시된 바람직한 속성들을 전적으로 담당하지는 않는다.
- [0008] 하나의 양태에서, 고정 비트레이트 비디오 코딩 방식 (constant bitrate video coding scheme) 의 블록 예측 모드에서 비디오 데이터의 블록을 코딩하는 방법은: 블록 예측 모드와 연관된 제 1 파티셔닝 방식에 기초하여 비디오 데이터의 블록 내의 현재 구역을 예측하기 위해 사용될 하나 이상의 제 1 후보 구역들을 결정하는 단계로서, 하나 이상의 제 1 후보 구역들은 현재 구역과 연관된 위치들의 제 1 범위 내에 있고, 하나 이상의 제 1 후보 구역들은 비디오 인코딩 디바이스의 메모리에 저장되는, 상기 하나 이상의 제 1 후보 구역들을 결정하는 단계; 블록 예측 모드와 연관된 제 2 파티셔닝 방식에 기초하여 현재 구역을 예측하기 위해 사용될 하나 이상의 제 2 후보 구역들을 결정하는 단계로서, 하나 이상의 제 2 후보 구역들은 현재 구역과 연관된 위치들의 제 2 범위 내에 있고, 하나 이상의 제 2 후보 구역들은 비디오 인코딩 디바이스의 메모리에 저장되는, 상기 하나 이상의 제 2 후보 구역들을 결정하는 단계; 제 1 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 1 코스트 (cost) 가 제 2 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 큰지 여부를 결정하는 단계; 및, 제 1 코스트가 제 2 코스트보다 더 크다고 결정하는 것에 응답하여, 하나 이상의 제 2 후보 구역들에 기초하여 현재 구역을 비트스트림에 코딩하는 단계를 포함할 수도 있다.
- [0009] 다른 양태에서, 고정 비트레이트 비디오 코딩 방식의 블록 예측 모드에서 비디오 데이터의 블록을 코딩하도록 구성된 장치는: 하나 이상의 후보 구역들과 연관된 비디오 데이터를 저장하도록 구성된 메모리, 및 메모리와 통신하는 하나 이상의 프로세서들을 포함할 수도 있다. 이 하나 이상의 프로세서들은: 블록 예측 모드와 연관된 제 1 파티셔닝 방식에 기초하여 비디오 데이터의 블록 내의 현재 구역을 예측하기 위해 사용될 하나 이상의 제 1 후보 구역들을 결정하는 것으로서, 하나 이상의 제 1 후보 구역들은 현재 구역과 연관된 위치들의 제 1 범위 내에 있는, 상기 하나 이상의 제 1 후보 구역들을 결정하는 것을 행하고; 블록 예측 모드와 연관된 제 2 파티셔닝 방식에 기초하여 현재 구역을 예측하기 위해 사용될 하나 이상의 제 2 후보 구역들을 결정하는 것으로서, 하나 이상의 제 2 후보 구역들은 현재 구역과 연관된 위치들의 제 2 범위 내에 있는, 상기 하나 이상의 제 2 후보 구역들을 결정하는 것을 행하며; 제 1 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 1 코스트가 제 2 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 큰지 여부를 결정하고; 그리고, 제 1 코스트가 제 2 코스트보다 더 크다고 결정하는 것에 응답하여, 하나 이상의 제 2 후보 구역들에 기초하여 현재 구역을 비트스트림에 코딩하도록 구성되는, 비디오 데이터의 블록을 코딩하도록 구성될 수도 있다.
- [0010] 또 다른 양태에서, 비-일시적 물리적 컴퓨터 스토리지 (non-transitory physical computer storage) 는, 고정 비트레이트 비디오 코딩 방식의 블록 예측 모드에서 비디오 데이터의 블록을 코딩하도록 구성된 코드를 포함할 수도 있다. 이 코드는, 실행될 때, 장치로 하여금: 블록 예측 모드와 연관된 제 1 파티셔닝 방식에 기초하여 비디오 데이터의 블록 내의 현재 구역을 예측하기 위해 사용될 하나 이상의 제 1 후보 구역들을 결정하는 것으로서, 하나 이상의 제 1 후보 구역들은 현재 구역과 연관된 위치들의 제 1 범위 내에 있는, 상기 하나 이상의 제 1 후보 구역들을 결정하는 것을 행하게 하고; 블록 예측 모드와 연관된 제 2 파티셔닝 방식에 기초하여 현재 구역을 예측하기 위해 사용될 하나 이상의 제 2 후보 구역들을 결정하는 것으로서, 하나 이상의 제 2 후보 구역들은 현재 구역과 연관된 위치들의 제 2 범위 내에 있는, 상기 하나 이상의 제 2 후보 구역들을 결정하는 것을 행하게 하며; 제 1 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 1 코스트가 제 2 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 큰지 여부를 결정하게 하고; 그리고, 제 1 코스트가 제 2 코스트보다 더 크다고 결정하는 것에 응답하여, 하나 이상의 제 2 후보 구역들에 기초하여 현재 구역을 비트스트림에 코딩하게 할 수도 있다.
- [0011] 또 다른 양태에서, 비디오 코딩 디바이스는 고정 비트레이트 비디오 코딩 방식의 블록 예측 모드에서 비디오 데이터의 블록을 코딩하도록 구성될 수도 있다. 비디오 코딩 디바이스는: 블록 예측 모드와 연관된 제 1 파티셔닝 방식에 기초하여 비디오 데이터의 블록 내의 현재 구역을 예측하기 위해 사용될 하나 이상의 제 1 후보 구역들을 결정하는 수단으로서, 하나 이상의 제 1 후보 구역들은 현재 구역과 연관된 위치들의 제 1 범위 내에 있는, 상기 하나 이상의 제 1 후보 구역들을 결정하는 수단; 블록 예측 모드와 연관된 제 2 파티셔닝 방식에 기초하여 현재 구역을 예측하기 위해 사용될 하나 이상의 제 2 후보 구역들을 결정하는 수단으로서, 하나 이상의 제 2 후보 구역들은 현재 구역과 연관된 위치들의 제 2 범위 내에 있는, 상기 하나 이상의 제 2 후보 구역들을 결정하는 수단; 제 1 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 1 코스트가 제 2 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 큰지 여부를 결정하는 수단; 및, 제 1 코스트가 제 2 코스트보다 더 크다고 결정하는 것에 응답하여, 하나 이상의 제 2 후보 구역들에 기초하여 현재

구역을 비트스트림에 코딩하는 수단을 포함할 수도 있다.

도면의 간단한 설명

- [0012] 도 1a 는 이 개시물에서 설명된 양태들에 따라 기법들을 이용할 수도 있는 일 예의 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.
- 도 1b 는 이 개시물에서 설명된 양태들에 따라 기법들을 수행할 수도 있는 또 다른 예의 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.
- 도 2a 는 기법들을 구현할 수도 있는 비디오 인코더의 예를 예시하는 블록도이다.
- 도 2b 는 기법들을 구현할 수도 있는 비디오 디코더의 예를 예시하는 블록도이다.
- 도 3 은 1-D 블록에 대한 비-최초 라인 (non-first line) 을 위한 검색 공간을 예시하는 블록도이다.
- 도 4 는 2-D 블록에 대한 비-최초 라인을 위한 검색 공간을 예시하는 블록도이다.
- 도 5 는 1-D 블록에 대한 최초 라인을 위한 검색 공간을 예시하는 블록도이다.
- 도 6 은 2-D 블록에 대한 최로 라인을 위한 검색 공간을 예시하는 블록도이다.
- 도 7 은 블록 예측 모드에서 비디오 데이터의 블록을 예측하기 위한 방법을 예시하는 플로우차트이다.
- 도 8 은 파티션들을 가지는 블록을 예시하는 블록도이다.
- 도 9 는 적응적 파티션 사이즈를 갖는 블록 예측 모드에 대한 데이터 플로우를 예시하는 블록도이다.
- 도 10 은 블록 내에서 2x2 구역에 대한 2 개의 상이한 파티션 옵션들을 예시하는 블록도이다.
- 도 11 은 블록 예측 모드에 대한 엔트로피 코딩 그룹들을 예시하는 블록도이다.
- 도 12 는 2x8 블록에 대한 검색 공간을 예시하는 블록도이다.
- 도 13 은 블록의 상이한 구역들에 대해 사용되는 상이한 파티션 사이즈들을 예시하는 블록도이다.
- 도 14 는 가변 파티션 사이즈들을 이용하여 블록 예측 모드에서 비디오 데이터의 블록을 예측하기 위한 방법을 예시하는 플로우차트이다.
- 도 15 는 4:2:0 크로마 서브-샘플링을 위한 2x2 파티션들에 대한 일 예시적인 블록 예측 검색을 예시하는 블록도이다.
- 도 16 은 4:2:0 크로마 서브-샘플링을 위한 1x2 파티션들에 대한 일 예시적인 블록 예측 검색을 예시하는 블록도이다.
- 도 17 은 4:2:2 크로마 서브-샘플링을 위한 2x2 파티션들에 대한 일 예시적인 블록 예측 검색을 예시하는 블록도이다.
- 도 18 은 4:2:2 크로마 서브-샘플링을 위한 1x2 파티션들에 대한 일 예시적인 블록 예측 검색을 예시하는 블록도이다.

발명을 실시하기 위한 구체적인 내용

- [0013] 일반적으로, 이 개시물은 예를 들어 디스플레이 링크 비디오 압축에서 이용되는 것들과 같은, 비디오 압축 기법들을 개선시키는 방법들에 관한 것이다. 더욱 구체적으로, 본 개시물은 블록 예측 모드에서 비디오 데이터의 블록을 코딩하기 위한 시스템들 및 방법들에 관한 것이다.
- [0014] 어떤 실시형태들은 디스플레이 링크 비디오 압축 기법의 일 예인 DSC 표준의 맥락에서 본원에서 설명되지만, 당해 분야의 당업자는 본원에서 개시된 시스템들 및 방법들이 임의의 적당한 비디오 코딩 표준에 적용가능할 수도 있다는 것을 인식할 것이다. 예를 들어, 본원에서 개시된 실시형태들은 다음의 표준들 중의 하나 이상에 적용가능할 수도 있다: 국제 전기통신 연합 (International Telecommunication Union; ITU) 전기통신 표준화 섹터 (Telecommunication Standardization Sector) (ITU-T) H.261, 표준화를 위한 국제 기구 (International Organization for Standardization; ISO) / 국제 전기기술 위원회 (International Electrotechnical Commission; IEC) (ISO/IEC) 동화상 전문가 그룹-1 (Moving Picture Experts Group-1; MPEG-1) 비주열

(Visual), ITU-T H.262 또는 ISO/IEC MPEG-2 비주얼, ITU-T H.263, ISO/IEC MPEG-4 비주얼, (또한, ISO/IEC MPEG-4 AVC 로서 알려진) ITU-T H.264, 및 이러한 표준들에 대한 임의의 확장들. 또한, 이 개시물에서 설명된 기법들은 미래에 개발된 표준들의 일부가 될 수도 있다. 다시 말해서, 이 개시물에서 설명된 기법들은 이전에 개발된 비디오 코딩 표준들, 현재 개발 중인 비디오 코딩 표준들, 및 당면할 비디오 코딩 표준들에 적용 가능할 수도 있다.

[0015] DSC 표준은 비디오 데이터의 각각의 블록이 인코더에 의해 인코딩될 수도 있고, 유사하게, 디코더에 의해 디코딩될 수도 있는 다수의 코딩 모드들을 포함한다. 일부 구현예들에서, 인코더 및/또는 디코더는 이전에 코딩된 블록에 기초하여 코딩되어야 할 현재 블록을 예측할 수도 있다.

[0016] 그러나, 기존의 코딩 모드들 (예컨대, 변환 코딩, 차동 펄스-코드 변조 (differential pulse-code modulation) 등) 은 비디오 데이터에서의 고도로 복잡한 구역들을 압축하는 만족스러운 방법을 제공하지는 않는다. 종종, 이러한 유형의 데이터 (즉, 고도로 압축된 비디오 데이터) 에 대하여, 코딩되어야 할 현재 블록 (또는 현재 블록의 구성 서브-블록들) 은 코더 (예컨대, 인코더 또는 디코더) 에 의해 조우 (encounter) 되었던 이전 블록들과 콘텐츠에 있어서 유사하다. 그러나, 기존의 인트라 예측 (intra prediction) 은 너무 제한되어, 이러한 현재 블록의 만족스러운 예측 (예컨대, 충분히 작은 잔차 (residual) 를 산출할 예측) 을 제공하지 못할 수도 있다. 이에 따라, 비디오 데이터의 코딩 블록들의 개선된 방법이 요망된다.

[0017] 본 개시물에서는, 블록 예측 모드에서 블록을 코딩하는 개선된 방법이 설명된다. 예를 들어, 현재 블록 (또는 현재 블록 내의 현재 구역) 을 예측하기 위하여 이용될 후보 블록 (또는 후보 구역) 을 검색할 때, 검색 범위는 인코더가 검색 코스트 (search cost) 를 최소화하면서 양호한 정합일 수도 있는 잠재적인 후보들을 액세스하도록 정의될 수도 있다. 또 다른 예에서, 방법은 각각의 블록 (또는 각각의 파티션) 에 대한 예측을 명시적으로 시그널링하는 것을 포함할 수도 있다. 또 다른 예에서, 인코더는 레이트 왜곡 (RD) 분석에 기초하여 단일 파티션 또는 다수의 파티션들을 이용하여 현재 블록을 코딩할지 여부를 결정할 수도 있다. 인코더 측에서 더 많은 동작들 (예컨대, 컴퓨팅 자원들 또는 프로세싱 파워를 소비할 수도 있는, 현재 블록을 예측하기 위하여 이용되어야 할 후보 블록을 검색하는 것, 현재 블록에 대한 후보 블록의 위치를 식별하는 벡터를 계산하는 것, 상이한 파티션 사이즈들에 대해 RD 코스트를 계산하는 것 및 어느 파티션 사이즈가 최상의 코딩 효율을 산출하는지를 결정하는 것 등) 을 수행함으로써, 방법은 디코더 복잡도를 감소시킬 수도 있다. 또한, 인코더로 하여금 각각의 블록에 대한 파티션 사이즈를 적응적으로 선택하는 것을 허용함으로써, 블록 예측 방식의 퍼포먼스 (performance) 가 추가로 향상될 수도 있다.

[0018] **비디오 코딩 표준들**

[0019] 비디오 이미지, TV 이미지, 스틸 이미지 (still image), 또는 비디오 레코더 또는 컴퓨터에 의해 생성된 이미지와 같은 디지털 이미지는 수평 및 수직 라인들로 배열된 픽셀들 또는 샘플들을 포함할 수도 있다. 단일 이미지에서의 픽셀들의 수는 전형적으로 수만 개이다. 각각의 픽셀은 전형적으로 휘도 (luminance) 및 색차 (chrominance) 정보를 포함한다. 압축이 없다면, 이미지 인코더로부터 이미지 디코더로 전달되어야 할 정보의 순수한 분량은 실시간 이미지 송신을 비실용적으로 할 것이다. 송신되어야 할 정보의 양을 감소시키기 위하여, JPEG, MPEG, 및 H.263 표준들과 같은 다수의 상이한 압축 방법들이 개발되었다.

[0020] 비디오 코딩 표준들은 ITU-T H.261, ISO/IEC MPEG-1 비주얼, ITU-T H.262 또는 ISO/IEC MPEG-2 비주얼, ITU-T H.263, ISO/IEC MPEG-4 비주얼, (또한, ISO/IEC MPEG-4 AVC 로서 알려진) ITU-T H.264, 및 이러한 표준들의 확장들을 포함하는 HEVC 를 포함한다.

[0021] 게다가, 비디오 코딩 표준, 즉, DSC 는 VESA 에 의해 개발되었다. DSC 표준은 디스플레이 링크들을 통한 송신을 위하여 비디오를 압축할 수 있는 비디오 압축 표준이다. 디스플레이들의 해상도가 증가함에 따라, 디스플레이들을 구동하기 위하여 요구된 비디오 데이터의 대역폭이 대응하여 증가한다. 일부 디스플레이 링크들은 비디오 데이터의 전부를 이러한 해상도들을 위한 디스플레이로 송신하기 위한 대역폭을 가지지 않을 수도 있다. 따라서, DSC 표준은 디스플레이 링크들 상에서의 상호동작가능한, 시각적 무손실 압축을 위한 압축 표준을 특정한다.

[0022] DSC 표준은 H.264 및 HEVC 와 같은 다른 비디오 코딩 표준들과는 상이하다. DSC 는 인트라-프레임 (intra-frame) 압축을 포함하지만, 인터-프레임 (inter-frame) 압축을 포함하지 않으며, 이것은 시간적 정보가 비디오 데이터를 코딩할 시에 DSC 표준에 의해 이용되지 않을 수도 있다는 것을 의미한다. 대조적으로, 다른 비디오 코딩 표준들은 그 비디오 코딩 기법들에서 인터-프레임 압축을 채용할 수도 있다.

[0023] **비디오 코딩 시스템**

[0024] 신규한 시스템들, 장치들, 및 방법들의 다양한 양태들은 동반된 도면들을 참조하여 이하에서 더욱 완전하게 설명된다. 그러나, 이 개시물은 많은 상이한 형태들로 구체화될 수도 있고, 이 개시물의 전반에 걸쳐 제시된 임의의 특정 구조 또는 기능으로 제한된 것으로서 해석되지 않아야 한다. 오히려, 이 양태들은, 이 개시물이 철저하고 완전할 것이며, 개시물의 범위를 당해 분야의 당업자들에게 완전히 전달하도록 제공된다. 본원에서의 교시사항들에 기초하여, 당해 분야의 당업자는 개시물의 범위가, 본 개시물의 임의의 다른 양태에 독립적으로 또는 이와 조합하여 구현되든지 간에, 본원에서 개시된 신규한 시스템들, 장치들, 및 방법들의 임의의 양태를 커버 (cover) 하도록 의도되는 것을 인식해야 한다. 예를 들어, 본원에서 기재된 임의의 수의 양태들을 이용하여 장치가 구현될 수도 있거나 방법이 실시될 수도 있다. 게다가, 본 개시물의 범위는 본원에서 기재된 본 개시물의 다양한 양태들에 추가하여, 또는 이 다양한 양태들 이외에, 다른 구조, 기능성, 또는 구조 및 기능성을 이용하여 실시되는 이러한 장치 또는 방법을 커버하도록 의도된다. 본원에서 개시된 임의의 양태는 청구항의 하나 이상의 구성요소들에 의해 구체화될 수도 있다는 것을 이해해야 한다.

[0025] 특정한 양태들이 본원에서 설명되지만, 이 양태들의 많은 변형들 및 치환들은 개시물의 범위 내에 속한다. 바람직한 양태들의 일부 이득들 및 장점들이 언급되지만, 개시물의 범위는 특정한 이득들, 용도들, 또는 목적들에 제한되도록 의도된 것이 아니다. 오히려, 개시물의 양태들은 상이한 무선 기술들, 시스템 구성들, 네트워크들, 및 송신 프로토콜들에 폭넓게 적용가능하도록 의도되며, 이들의 일부는 바람직한 양태들의 도면들 및 다음의 설명에서 예로서 예시되어 있다. 상세한 설명 및 도면들은 제한하는 것이 아니라 개시물의 예시에 불과하고, 개시물의 범위는 첨부된 청구항들 및 그 등가물들에 의해 정의된다.

[0026] 첨부된 도면들은 예들을 예시한다. 첨부된 도면들에서 참조 번호들에 의해 표시된 구성요소들은 다음의 설명에서 유사한 참조 번호들에 의해 표시된 구성요소들에 대응한다. 이 개시물에서, 서수 단어 (예컨대, "제 1", "제 2", "제 3" 및 등등) 로 시작하는 명칭들을 가지는 구성요소들은 구성요소들이 특정한 순서를 가지는 것을 반드시 암시하지는 않는다. 오히려, 이러한 서수 단어들은 동일하거나 유사한 타입의 상이한 구성요소들을 지칭하기 위하여 이용되는 것에 불과하다.

[0027] 도 1a 는 이 개시물에서 설명된 양태들에 따라 기법들을 사용할 수도 있는 일 예의 비디오 코딩 시스템 (10) 을 예시하는 블록도이다. 본원에서 이용되고 설명된 바와 같이, 용어 "비디오 코더" 또는 "코더" 는 일반적으로 비디오 인코더들 및 비디오 디코더들의 양자를 지칭한다. 이 개시물에서, 용어들 "비디오 코딩" 또는 "코딩" 은 일반적으로 비디오 인코딩 및 비디오 디코딩을 지칭할 수도 있다. 비디오 인코더들 및 비디오 디코더들에 추가하여, 본 출원에서 설명된 양태들은 트랜스코더 (transcoder) 들 (예컨대, 비트스트림을 디코딩할 수 있고 또 다른 비트스트림을 재-인코딩 (re-encode) 할 수 있는 디바이스들) 및 미들박스 (middlebox) 들 (예컨대, 비트스트림을 수정할 수 있고, 변환할 수 있고, 및/또는 그렇지 않을 경우에 조작할 수 있는 디바이스들) 과 같은 다른 관련된 디바이스들로 확장될 수도 있다.

[0028] 도 1a 에서 도시된 바와 같이, 비디오 코딩 시스템 (10) 은 목적지 디바이스 (14) (즉, "비디오 코딩 디바이스 (14)" 또는 "코딩 디바이스 (14)") 에 의해 더 이후의 시간에 디코딩되어야 할 인코딩된 비디오 데이터를 생성하는 소스 디바이스 (12) (즉, "비디오 코딩 디바이스 (12)" 또는 "코딩 디바이스 (12)") 를 포함한다. 도 1a 의 예에서, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 별도의 디바이스들을 구성한다. 그러나, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 도 1b 의 예에서 도시된 바와 같이, 동일한 디바이스 상에 있을 수도 있거나 동일한 디바이스의 일부일 수도 있다는 것에 주목한다.

[0029] 도 1a 를 다시 한번 참조하면, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 각각, 데스크톱 컴퓨터들, 노트북 (예컨대, 랩톱) 컴퓨터들, 태블릿 컴퓨터들, 셋톱 박스 (set-top box) 들, 소위 "스마트" 폰들과 같은 전화 핸드셋들, 소위 "스마트" 패드들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게임용 콘솔들, 비디오 스트리밍 디바이스 등등을 포함하는 광범위한 디바이스들 (또한, 비디오 코딩 디바이스드로서 지칭됨) 중의 임의의 것을 포함할 수도 있다. 다양한 실시형태들에서, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 무선 통신을 위해 구비될 수도 있다 (즉, 무선 통신을 통해 통신하도록 구성될 수도 있다).

[0030] 비디오 코딩 시스템 (10) 의 비디오 코딩 디바이스들 (12, 14) 은 무선 광역 네트워크 (WWAN) (예컨대, 셀룰러) 및/또는 무선 로컬 영역 네트워크 (WLAN) 캐리어들과 같은, 무선 네트워크들 및 라디오 기술들을 통해 통신하도록 구성될 수도 있다. 용어들 "네트워크" 및 "시스템" 은 종종 상호교환가능하게 사용된다. 비디오 코딩 디바이스들 (12, 14) 의 각각은 사용자 장비 (UE), 무선 디바이스, 단말기, 모바일 스테이션, 가입자 유닛

등일 수도 있다.

- [0031] WWAN 캐리어들은, 예를 들어, 코드 분할 다중 액세스 (CDMA), 시간 분할 다중 액세스 (TDMA), 주파수 분할 다중 액세스 (FDMA), 직교 FDMA (OFDMA), 단일-캐리어 FDMA (SC-FDMA) 및 다른 네트워크들과 같은 무선 통신 네트워크들을 포함할 수도 있다. CDMA 네트워크는 유니버설 지상 라디오 액세스 (UTRA), CDMA2000 등과 같은 무선 기술을 구현할 수도 있다. UTRA 는 와이드밴드 CDMA (WCDMA) 및 CDMA 의 다른 변형들을 포함한다. CDMA2000 은 IS-2000, IS-95 및 IS-856 표준들을 커버한다. TDMA 네트워크는 모바일 통신을 위한 글로벌 시스템 (GSM) 과 같은 무선 기술을 구현할 수도 있다. OFDMA 네트워크는 진화형 UTRA (E-UTRA), 울트라 모바일 브로드밴드 (UMB), IEEE 802.11 (Wi-Fi), IEEE 802.16 (WiMAX), IEEE 802.20, 플래시-OFDMA 등과 같은 무선 기술을 구현할 수도 있다. UTRA 및 E-UTRA 는 유니버설 모바일 전기통신 시스템 (UMTS) 의 일부이다. 3GPP 롱 텀 에볼루션 (LTE) 및 LTE-어밴스드 (LTE-A) 는 UMTS 및 E-UTRA 의 새로운 릴리스들이다. UTRA, E-UTRA, UMTS, LTE, LTE-A 및 GSM 은 "3세대 파트너십 프로젝트" (3GPP) 라는 명칭의 조직으로부터의 문서들에서 기술된다. CDMA2000 및 UMB 는 "3세대 파트너십 프로젝트 2" (3GPP2) 라는 명칭의 조직으로부터의 문서들에서 기술된다.
- [0032] 비디오 코딩 시스템 (10) 의 비디오 코딩 디바이스들 (12, 14) 은 또한, 예를 들어, 이들 수정들: 802.11a-1999 (통상적으로 "802.11a" 로 불림), 802.11b-1999 (통상적으로 "802.11b" 로 불림), 802.11g-2003 (통상적으로 "802.11g" 로 불림), 및 기타 등등을 포함하는, IEEE 802.11 표준과 같은 하나 이상의 표준들에 따라 WLAN 기지국을 통해 서로에 대해 통신할 수도 있다.
- [0033] 목적지 디바이스 (14) 는 링크 (16) 를 통해, 디코딩되어야 할 인코딩된 비디오 데이터를 수신할 수도 있다. 링크 (16) 는 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 이동시킬 수 있는 임의의 타입의 매체 또는 디바이스를 포함할 수도 있다. 도 1a 의 예에서, 링크 (16) 는 소스 디바이스 (12) 가 인코딩된 비디오 데이터를 실시간으로 목적지 디바이스 (14) 로 송신하는 것을 가능하게 하기 위한 통신 매체를 포함할 수도 있다. 인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라 변조될 수도 있고, 목적지 디바이스 (14) 로 송신될 수도 있다. 통신 매체는 라디오 주파수 (radio frequency; RF) 스펙트럼 또는 하나 이상의 물리적 송신 라인들과 같은 임의의 무선 또는 유선 통신 매체를 포함할 수도 있다. 통신 매체는 로컬 영역 네트워크, 광역 네트워크, 또는 인터넷과 같은 글로벌 네트워크와 같은 패킷-기반 네트워크의 일부를 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로의 통신을 용이하게 하기 위해 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.
- [0034] 도 1a 의 예에서, 소스 디바이스 (12) 는 비디오 소스 (18), 비디오 인코더 (20) (또한 단순히 인코더 (20) 로서 지칭됨), 및 출력 인터페이스 (22) 를 포함한다. 일부의 경우들에는, 출력 인터페이스 (22) 가 변조기/복조기 (모뎀) 및/또는 송신기를 포함할 수도 있다. 소스 디바이스 (12) 에서, 비디오 소스 (18) 는 비디오 캡처 디바이스, 예컨대, 비디오 카메라, 이전에 캡처된 비디오를 포함하는 비디오 아카이브 (video archive), 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 공급 인터페이스, 및/또는 소스 비디오로서 컴퓨터 그래픽 데이터를 생성하기 위한 컴퓨터 그래픽 시스템과 같은 소스, 또는 이러한 소스들의 조합을 포함할 수도 있다. 하나의 예로서, 비디오 소스 (18) 가 비디오 카메라일 경우, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 도 1b 의 예에서 예시된 바와 같이, 소위 "카메라 폰들" 또는 "비디오 폰들" 을 형성할 수도 있다. 그러나, 이 개시물에서 설명된 기법들은 일반적으로 비디오 코딩에 적용가능할 수도 있고, 무선 및/또는 유선 애플리케이션들에 적용될 수도 있다.
- [0035] 캡처된 (captured), 프리-캡처된 (pre-captured), 또는 컴퓨터-생성된 비디오는 비디오 인코더 (20) 에 의해 인코딩될 수도 있다. 인코딩된 비디오 데이터는 소스 디바이스 (12) 의 출력 인터페이스 (22) 를 통해 목적지 디바이스 (14) 로 송신될 수도 있다. 인코딩된 비디오 데이터는 또한 (또는 대안적으로), 디코딩 및/또는 재생을 위하여, 목적지 디바이스 (14) 또는 다른 디바이스들에 의한 더 이후의 액세스를 위해 저장 디바이스 (31) 상으로 저장될 수도 있다. 도 1a 및 도 1b 에서 예시된 비디오 인코더 (20) 는 도 2a 에서 예시된 비디오 인코더 (20), 또는 본문에서 설명된 임의의 다른 비디오 인코더를 포함할 수도 있다.
- [0036] 도 1a 의 예에서, 목적지 디바이스 (14) 는 입력 인터페이스 (28), 비디오 디코더 (30) (또한 단순히 디코더 (30) 로서 지칭됨), 및 디스플레이 디바이스 (32) 를 포함한다. 일부 경우들에는, 입력 인터페이스 (28) 가 수신기 및/또는 모뎀을 포함할 수도 있다. 목적지 디바이스 (14) 의 입력 인터페이스 (28) 는 링크 (16) 를 통해, 및/또는 저장 디바이스 (31) 로부터 인코딩된 비디오 데이터를 수신할 수도 있다. 링크 (16) 를 통해

통신되거나 저장 디바이스 (31) 상에서 제공된 인코딩된 비디오 데이터는 비디오 데이터를 디코딩할 시에, 비디오 디코더 (30) 와 같은 비디오 디코더에 의한 이용을 위하여 비디오 인코더 (20) 에 의해 생성된 다양한 선택스 엘리먼트들을 포함할 수도 있다. 이러한 선택스 엘리먼트들은 통신 매체 상에서 송신되거나, 저장 매체 상에 저장되거나, 파일 서버에 저장된 인코딩된 비디오 데이터와 함께 포함될 수도 있다. 도 1a 및 도 1b 에서 예시된 비디오 디코더 (30) 는 도 2b 에서 예시된 비디오 디코더 (30), 또는 본원에서 설명된 임의의 다른 비디오 디코더를 포함할 수도 있다.

[0037] 디스플레이 디바이스 (32) 는 목적지 디바이스 (14) 와 통합될 수도 있거나, 목적지 디바이스 (14) 의 외부에 있을 수도 있다. 일부 예들에서, 목적지 디바이스 (14) 는 통합된 디스플레이 디바이스를 포함할 수도 있고, 또한, 외부의 디스플레이 디바이스와 인터페이스하도록 구성될 수도 있다. 다른 예들에서, 목적지 디바이스 (14) 는 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32) 는 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 액정 디스플레이 (liquid crystal display; LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (organic light emitting diode; OLED) 디스플레이, 또는 또 다른 타입의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중의 임의의 것을 포함할 수도 있다.

[0038] 관련된 양태들에서, 도 1b 는 일 예의 비디오 코딩 시스템 (10') 을 도시하고, 여기서, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 디바이스 (11) 상에 있거나, 디바이스 (11) 의 일부이다. 디바이스 (11) 는 "스마트" 폰 등과 같은 전화 핸드셋일 수도 있다. 디바이스 (11) 는 소스 디바이스 (12) 및 목적지 디바이스 (14) 와 동작가능하게 통신하는 (선택적으로 존재하는) 프로세서/제어기 디바이스 (13) 를 포함할 수도 있다. 도 1b 의 비디오 코딩 시스템 (10') 및 그 컴포넌트들은 그 외에는, 도 1a 의 비디오 코딩 시스템 (10) 및 그 컴포넌트들과 유사하다.

[0039] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 DSC 와 같은 비디오 압축 표준에 따라 동작할 수도 있다. 대안적으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 MPEG-4, Part 10, AVC, HEVC, 또는 이러한 표준들의 확장들로서 대안적으로 지칭된 ITU-T H.264 표준과 같은 다른 독점적 또는 산업 표준들에 따라 동작할 수도 있다. 그러나, 이 개시물의 기법들은 임의의 특정한 코딩 표준으로 제한되지는 않는다. 비디오 압축 표준들의 다른 예들은 MPEG-2 및 ITU-T H.263 을 포함한다.

[0040] 도 1a 및 도 1b 의 예들에서 도시되지 않았지만, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 각각 오디오 인코더 및 디코더와 통합될 수도 있고, 공통의 데이터 스트림 또는 별도의 데이터 스트림들에서 오디오 및 비디오 양자의 인코딩을 처리하기 위한 적합한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능한 경우, 일부 예들에서, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉스 프로토콜, 또는 사용자 데이터그램 프로토콜 (user datagram protocol; UDP) 과 같은 다른 프로토콜들을 준수할 수도 있다.

[0041] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 각각, 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서 (digital signal processor; DSP) 들, 애플리케이션 특정 집적 회로 (application specific integrated circuit; ASIC) 들, 필드 프로그래밍가능 게이트 어레이 (field programmable gate array; FPGA) 들, 개별 로직, 소프트웨어, 하드웨어, 펌웨어, 또는 그 임의의 조합들과 같은 다양한 적당한 인코더 회로부 중의 임의의 것으로서 구현될 수도 있다. 기법들이 소프트웨어로 부분적으로 구현될 때, 디바이스는 소프트웨어를 위한 명령들을 적당한 비-일시적 컴퓨터-판독가능 매체 내에 저장할 수도 있고, 이 개시물의 기법들을 수행하기 위하여 하나 이상의 프로세서들을 이용하여 명령들을 하드웨어로 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 의 각각은 하나 이상의 인코더들 또는 디코더들 내에 포함될 수도 있고, 그 어느 하나는 조합된 인코더/디코더의 일부로서 개개의 디바이스 내에 통합될 수도 있다.

[0042] **비디오 코딩 프로세스**

[0043] 위에서 간단히 언급된 바와 같이, 비디오 인코더 (20) 는 비디오 데이터를 인코딩한다. 비디오 데이터는 하나 이상의 픽처들을 포함할 수도 있다. 픽처들의 각각은 비디오의 일부를 형성하는 스틸 이미지이다. 일부 사례들에서, 픽처는 비디오 "프레임" 으로서 지칭될 수도 있다. 비디오 인코더 (20) 가 비디오 데이터 (예컨대, 비디오 코딩 계층 (VCL) 데이터 및/또는 비-VCL 데이터) 를 인코딩할 때, 비디오 인코더 (20) 는 비트스트림을 생성할 수도 있다. 비트스트림은 비디오 데이터의 코딩된 표현을 형성하는 비트들의 시퀀스를 포함할 수도 있다. 비트스트림은 코딩된 픽처들 및 연관된 데이터를 포함할 수도 있다. 코딩된 픽처는 픽처의 코딩된 표현이다. VCL 데이터는 코딩된 픽처 데이터 (즉, 코딩된 픽처(들)의 샘플들과 연관된 정보) 를 포함할 수도 있고, 비-VCL 데이터는 하나 이상의 코딩된 픽처들과 연관된 제어 정보 (예컨대, 파라미터 셋들 및/또는 보충적 강화 정보) 를 포함할 수도 있다.

- [0044] 비트스트림을 생성하기 위하여, 비디오 인코더 (20) 는 비디오 데이터에서의 각각의 픽처에 대해 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 픽처들에 대해 인코딩 동작들을 수행할 때, 비디오 인코더 (20) 는 일련의 코딩된 픽처들 및 연관된 데이터를 생성할 수도 있다. 연관된 데이터는 양자화 파라미터 (quantization parameter; QP) 와 같은 코딩 파라미터들의 세트를 포함할 수도 있다. 코딩된 픽처를 생성하기 위하여, 비디오 인코더 (20) 는 픽처를 동일한 크기의 비디오 블록들로 파티셔닝할 수도 있다. 비디오 블록은 샘플들의 2 차원 어레이일 수도 있다. 코딩 파라미터들은 비디오 데이터의 매 블록에 대한 코딩 옵션 (예컨대, 코딩 모드) 을 정의할 수도 있다. 코딩 옵션은 희망하는 RD 를 달성하기 위하여 선택될 수도 있다.
- [0045] 일부 예들에서, 비디오 인코더 (20) 는 픽처를 복수의 슬라이스들로 파티셔닝할 수도 있다. 슬라이스들의 각각은 이미지 또는 프레임에서의 구역들의 나머지로부터의 정보 없이 독립적으로 디코딩될 수 있는 이미지 (예컨대, 프레임) 에서의 공간적으로 별개의 구역을 포함할 수도 있다. 각각의 이미지 또는 비디오 프레임은 단일 슬라이스에서 인코딩될 수도 있거나, 각각의 이미지 또는 비디오 프레임은 몇몇 슬라이스들에서 인코딩될 수도 있다. DSC 에서, 각각의 슬라이스를 인코딩하기 위하여 할당된 비트들의 수는 실질적으로 일정할 수도 있다. 픽처에 대해 인코딩 동작을 수행하는 것의 일부로서, 비디오 인코더 (20) 는 픽처의 각각의 슬라이스에 대해 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 슬라이스에 대해 인코딩 동작을 수행할 때, 비디오 인코더 (20) 는 슬라이스와 연관된 인코딩된 데이터를 생성할 수도 있다. 슬라이스와 연관된 인코딩된 데이터는 "코딩된 슬라이스" 로서 지칭될 수도 있다.
- [0046] **DSC 비디오 인코더**
- [0047] 도 2a 는 이 개시물에서 설명된 양태들에 따라 기법들을 구현할 수도 있는 비디오 인코더 (20) 의 예를 예시하는 블록도이다. 비디오 인코더 (20) 는 이 개시물의 기법들 중의 일부 또는 전부를 수행하도록 구성될 수도 있다. 일부 예들에서, 이 개시물에서 설명된 기법들은 비디오 인코더 (20) 의 다양한 컴포넌트들 사이에서 공유될 수도 있다. 일부 예들에서, 추가적으로 또는 대안적으로, 프로세서 (도시되지 않음) 는 이 개시물에서 설명된 기법들 중의 일부 또는 전부를 수행하도록 구성될 수도 있다.
- [0048] 설명의 목적들을 위하여, 이 개시물은 DSC 코딩의 맥락에서 비디오 인코더 (20) 를 설명한다. 그러나, 이 개시물의 기법들은 다른 코딩 표준들 또는 방법들에 적용가능할 수도 있다.
- [0049] 도 2a 의 예에서, 비디오 인코더 (20) 는 복수의 기능적 컴포넌트들을 포함한다. 비디오 인코더 (20) 의 기능적 컴포넌트들은 컬러-공간 변환기 (105), 버퍼 (110), 평탄도 검출기 (115), 레이트 제어기 (120), 예측기, 양자화기, 및 재구성기 컴포넌트 (125), 라인 버퍼 (130), 인덱싱된 컬러 히스토리 (indexed color history; 135), 엔트로피 인코더 (140), 서브스트림 멀티플렉서 (145), 및 레이트 버퍼 (150) 를 포함한다. 다른 예들에서, 비디오 인코더 (20) 는 더 많거나, 더 적거나, 또는 상이한 기능적 컴포넌트들을 포함할 수도 있다.
- [0050] 컬러-공간 변환기 (105) 는 입력 컬러-공간을, 코딩 구현에서 이용된 컬러-공간으로 변환할 수도 있다. 예를 들어, 하나의 예시적인 실시형태에서, 입력 비디오 데이터의 컬러-공간은 적색, 녹색, 및 청색 (RGB) 컬러-공간에 있고, 코딩은 휘도 Y, 색차 녹색 Cg, 및 색차 황색 Co (YCbCr) 컬러-공간에서 구현된다. 컬러-공간 변환은 비디오 데이터에 대한 시프트 (shift) 들 및 추가들을 포함하는 방법 (들) 에 의해 수행될 수도 있다. 다른 컬러-공간들에서의 입력 비디오 데이터가 프로세싱될 수도 있고 다른 컬러-공간들로의 변환들이 또한 수행될 수도 있다는 것에 주목한다.
- [0051] 관련된 양태들에서, 비디오 인코더 (20) 는 버퍼 (110), 라인 버퍼 (130), 및/또는 레이트 버퍼 (150) 를 포함할 수도 있다. 예를 들어, 버퍼 (110) 는 비디오 인코더 (20) 의 다른 부분들에 의한 그 이용 이전에 컬러-공간 변환된 비디오 데이터를 유지할 수도 있다. 또 다른 예에서, 비디오 데이터는 RGB 컬러-공간에서 저장될 수도 있고, 컬러-공간 변환된 데이터는 더 많은 비트들을 요구할 수도 있으므로, 컬러-공간 변환은 필요한 대로 수행될 수도 있다.
- [0052] 레이트 버퍼 (150) 는, 레이트 제어기 (120) 와 관련하여 이하에서 더욱 상세하게 설명될 비디오 인코더 (20) 에서의 레이트 제어 메커니즘의 일부로서 기능할 수도 있다. 각각의 블록을 인코딩하는 것에 소비되는 비트들의 수는 블록의 본질에 기초하여 고도로 실질적으로 변동될 수 있다. 레이트 버퍼 (150) 는 압축된 비디오에서 레이트 변동들을 평활화 (smooth) 할 수 있다. 일부 실시형태들에서는, 레이트 버퍼 (예컨대, 레이트 버퍼 (150)) 에서 저장된 비트들이 일정한 비트 레이트로 레이트 버퍼로부터 제거되는 고정 비트 레이트 (constant bit rate; CBR) 버퍼 모델이 채용된다. CBR 버퍼 모델에서는, 비디오 인코더 (20) 가 너무 많은

비트들을 비트스트림에 추가할 경우, 레이트 버퍼 (150) 가 오버플로우 (overflow) 할 수도 있다. 다른 한편으로, 비디오 인코더 (20) 는 레이트 버퍼 (150) 의 언더플로우 (underflow) 를 방지하기 위하여 충분한 비트들을 추가할 필요성이 있을 수도 있다.

[0053] 비디오 디코더 측 상에서는, 비트들이 고정 비트 레이트에서 비디오 디코더 (30) 의 레이트 버퍼 (155) (이하에서 더욱 상세하게 설명되는 도 2b 참조) 에 추가될 수도 있고, 비디오 디코더 (30) 는 각각의 블록에 대한 가변적인 수들의 비트들을 제거할 수도 있다. 적절한 디코딩을 보장하기 위하여, 비디오 디코더 (30) 의 레이트 버퍼 (155) 는 압축된 비트 스트림의 디코딩 동안에 "언더플로우" 또는 "오버플로우" 하지 않아야 한다.

[0054] 일부 실시형태들에서, 버퍼 충만도 (buffer fullness; BF) 는, 버퍼에서 현재 비트들의 수를 나타내는 BufferCurrentSize 와, 레이트 버퍼 (150) 의 크기, 즉, 임의의 시간 포인트에서 레이트 버퍼 (150) 내에 저장될 수 있는 비트들의 최대 수를 나타내는 BufferMaxSize 의 값들에 기초하여 정의될 수 있다. BF 는 다음과 같이 계산될 수도 있다:

[0055]
$$BF = ((BufferCurrentSize * 100) / BufferMaxSize)$$

[0056] 평탄도 검출기 (115) 는 비디오 데이터에서의 복잡한 (즉, 비-평탄한 (non-flat)) 영역들로부터 비디오 데이터에서의 평탄한 (즉, 간단하거나 균일한) 영역들로의 변경들을 검출할 수 있다. 용어들 "복잡한" 및 "평탄한" 은 비디오 인코더 (20) 가 비디오 데이터의 개개의 구역들을 인코딩하기 위한 어려움을 일반적으로 지칭하기 위하여 본원에서 이용될 것이다. 이에 따라, 본원에서 이용된 바와 같은 용어 복잡은 비디오 데이터의 구역을, 비디오 인코더 (20) 가 인코딩하기가 복잡한 것으로서 일반적으로 설명하고, 예를 들어, 텍스처링된 비디오 데이터 (textured video data), 높은 공간적 주파수, 및/또는 인코딩하기가 복잡한 다른 특징들을 포함할 수도 있다. 본원에서 이용된 바와 같은 용어 평탄은 비디오 데이터의 구역을, 비디오 인코더 (20) 가 인코딩하기가 간단한 것으로서 일반적으로 설명하고, 예를 들어, 비디오 데이터에서의 평활한 경도 (smooth gradient), 낮은 공간적 주파수, 및/또는 인코딩하기가 간단한 다른 특징들을 포함할 수도 있다. 복잡한 그리고 평탄한 구역들 사이의 전이 (transition) 들은 인코딩된 비디오 데이터에서 양자화 아티팩트 (quantization artifact) 들을 감소시키기 위하여 비디오 인코더 (20) 에 의해 이용될 수도 있다. 구체적으로, 레이트 제어기 (120) 및 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 복잡한 구역으로부터 평탄한 구역들로의 전이들이 식별될 때에 이러한 양자화 아티팩트들을 감소시킬 수 있다.

[0057] 레이트 제어기 (120) 는 코딩 파라미터들, 예컨대, QP 의 세트를 결정한다. 레이트 버퍼 (150) 가 오버플로우하거나 언더플로우하지 않는다는 것을 보장하는 타겟 비트레이트에 대한 픽처 품질을 최대화하기 위하여, QP 는 레이트 버퍼 (150) 의 버퍼 충만도 및 비디오 데이터의 이미지 활성화에 기초하여 레이트 제어기 (120) 에 의해 조절될 수도 있다. 레이트 제어기 (120) 는 또한, 최적의 RD 성능을 달성하기 위하여, 비디오 데이터의 각각의 블록에 대한 특정한 코딩 옵션 (예컨대, 특정한 모드) 을 선택한다. 레이트 제어기 (120) 는, 그 레이트 제어기 (120) 가 비트-레이트 제약을 충족시키도록, 즉, 전체적인 실제의 코딩 레이트가 타겟 비트 레이트 내에서 적합하도록, 재구성된 이미지들의 왜곡을 최소화한다.

[0058] 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 비디오 인코더 (20) 의 적어도 3 개의 인코딩 동작들을 수행할 수도 있다. 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 다수의 상이한 모드들에서 예측을 수행할 수도 있다. 하나의 예의 예측 모드는 중간-적응적 예측 (median-adaptive prediction) 의 수정된 버전이다. 중간-적응적 예측은 무손실 JPEG 표준 (lossless JPEG standard; JPEG-LS) 에 의해 구현될 수도 있다. 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 에 의해 수행될 수도 있는 중간-적응적 예측의 수정된 버전은 3 개의 연속 샘플 값들의 병렬 예측을 허용할 수도 있다. 또 다른 예의 예측 모드는 블록 예측이다. 블록 예측에서는, 샘플들은 상부 라인에서의 또는 동일 라인에서의 좌측에 이전에 재구성된 픽셀들로부터 예측된다. 일부 실시형태들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 양자는 블록 예측 사용량들을 결정하기 위하여 재구성된 픽셀들에 대해 동일한 검색을 수행할 수도 있고, 이에 따라, 어떤 비트들도 블록 예측 모드에서 전송될 필요가 없다. 다른 실시형태들에서, 비디오 인코더 (20) 는, 비디오 디코더 (30) 가 별도의 검색을 수행할 필요성이 없도록, 검색을 수행하고 비트스트림에서 블록 예측 벡터들을 시그널링할 수도 있다. 샘플들이 컴포넌트 범위의 중점을 이용하여 예측되는 중점 예측 모드 (midpoint prediction mode) 가 또한 구현될 수도 있다. 중점 예측 모드는 심지어 최악-경우의 샘플에서의 압축된 비디오에 대하여 요구된 비트들의 수의 경계결정을 가능하게 할 수도 있다. 도 3 내지 도 18 을 참조하여 이하 추가로 논의된 바와 같이, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 본 명세서에서 기술된 하나 이상의 기법들에 기초하여 비디오 데이터의 블록 (또는 임의의 다른 예측의 유닛) 을 코딩 (예컨대, 인코딩 또는 디코딩) 하도록 구성될

수도 있다. 예를 들어, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 도 7 및 도 14 에서 예시된 방법들을 수행하도록 구성될 수도 있다. 다른 실시형태들에서, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 비디오 인코더 (20) 의 하나 이상의 다른 컴포넌트들로 본 명세서에서 기술된 하나 이상의 방법들 또는 기법들을 수행하도록 구성될 수도 있다.

[0059] 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 또한 양자화를 수행한다. 예를 들어, 양자화는 시프터 (shifter) 를 이용하여 구현될 수도 있는 2-제곱 (power-of-2) 양자화기를 통해 수행될 수도 있다. 다른 양자화 기법들이 2-제곱 양자화기 대신에 구현될 수도 있다는 것에 주목한다. 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 에 의해 수행된 양자화는 레이트 제어기 (120) 에 의해 결정된 QP 에 기초할 수도 있다. 최종적으로, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 또한, 역 양자화된 잔차를 예측된 값에 추가하는 것과, 결과가 샘플 값들의 유효한 범위의 외부에 속하지 않는다는 것을 보장하는 것을 포함하는 재구성을 수행한다.

[0060] 예측기, 양자화기, 재구성기 컴포넌트 (125) 에 의해 수행된 예측, 양자화, 및 재구성에 대한 상기 설명된 예의 접근법들은 예시적인 것에 불과하고 다른 접근법들이 구현될 수도 있다는 것에 주목한다. 또한, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 예측, 양자화, 및/또는 재구성을 수행하기 위한 하위컴포넌트 (들) 를 포함할 수도 있다는 것에 주목한다. 또한, 예측, 양자화, 및/또는 재구성은 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 대신에 몇몇 별도의 인코더 컴포넌트들에 의해 수행될 수도 있다는 것에 주목한다.

[0061] 라인 버퍼 (130) 는 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 로부터의 출력을 유지 (예컨대, 저장) 하여, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 및 인덱싱된 컬러 히스토리 (135) 는 버퍼링된 비디오 데이터를 이용할 수 있다. 인덱싱된 컬러 히스토리 (135) 는 최근에 이용된 픽셀 값들을 저장한다. 이 최근에 이용된 픽셀 값들은 전용 신텍스를 통해 비디오 인코더 (20) 에 의해 직접적으로 참조될 수 있다.

[0062] 엔트로피 인코더 (140) 는 인덱싱된 컬러 히스토리 (135) 와, 평탄도 검출기 (115) 에 의해 식별된 평탄도 전이들에 기초하여, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 로부터 수신된 예측 잔차들 및 임의의 다른 데이터 (예컨대, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 에 의해 식별된 인덱스들) 를 인코딩한다. 일부 예들에서, 엔트로피 인코더 (140) 는 서브스트림 인코더에 대하여 이 클럭 당 3 개의 샘플들을 인코딩할 수도 있다. 서브스트림 멀티플렉서 (145) 는 무헤더 패킷 (headless packet) 멀티플렉싱 방식에 기초하여 비트스트림을 멀티플렉싱할 수도 있다. 이것은 비디오 디코더 (30) 가 이 3 개의 엔트로피 디코더들을 병렬로 실행하도록 하여, 이 클럭 당 3 개의 픽셀들의 디코딩을 용이하게 한다. 서브스트림 멀티플렉서 (145) 는 패킷 순서를 최적화할 수도 있어서, 패킷들은 비디오 디코더 (30) 에 의해 효율적으로 디코딩될 수 있다. 클럭 당 2-제곱 픽셀들 (예컨대, 2 픽셀들/클럭 또는 4 픽셀들/클럭) 의 디코딩을 용이하게 할 수도 있는 엔트로피 코딩에 대한 상이한 접근법들이 구현될 수도 있다는 것에 주목한다.

[0063] **DSC 비디오 디코더**

[0064] 도 2b 는 이 개시물에서 설명된 양태들에 따라 기법들을 구현할 수도 있는 비디오 디코더 (30) 의 예를 예시하는 블록도이다. 비디오 디코더 (30) 는 이 개시물의 기법들 중의 일부 또는 전부를 수행하도록 구성될 수도 있다. 일부 예들에서, 이 개시물에서 설명된 기법들은 비디오 디코더 (30) 의 다양한 컴포넌트들 사이에서 공유될 수도 있다. 일부 예들에서, 추가적으로 또는 대안적으로, 프로세서 (도시되지 않음) 는 이 개시물에서 설명된 기법들 중의 일부 또는 전부를 수행하도록 구성될 수도 있다.

[0065] 설명의 목적들을 위하여, 이 개시물은 DSC 코딩의 맥락에서 비디오 디코더 (30) 를 설명한다. 그러나, 이 개시물의 기법들은 다른 코딩 표준들 또는 방법들에 적용가능할 수도 있다.

[0066] 도 2b 의 예에서, 비디오 디코더 (30) 는 복수의 기능적 컴포넌트들을 포함한다. 비디오 디코더 (30) 의 기능적 컴포넌트들은 레이트 버퍼 (155), 서브스트림 디멀티플렉서 (160), 엔트로피 디코더 (165), 레이트 제어기 (170), 예측기, 양자화기, 및 재구성기 컴포넌트 (175), 인덱싱된 컬러 히스토리 (180), 라인 버퍼 (185), 및 컬러-공간 변환기 (190) 를 포함한다. 비디오 디코더 (30) 의 예시된 컴포넌트들은 도 2a 에서의 비디오 인코더 (20) 와 관련하여 위에서 설명된 대응하는 컴포넌트들과 유사하다. 이와 같이, 비디오 디코더 (30) 의 컴포넌트들의 각각은 위에서 설명된 바와 같은 비디오 인코더 (20) 의 대응하는 컴포넌트들과 유사한 방식으로 동작할 수도 있다. 일부 실시형태들에서, 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 의 하나 이상의 컴포넌트들은 이러한 컴포넌트들의 태스크들을 수행하도록 구성된 소프트웨어 코드를 실행하도록 구성된 하나 이상의 하드웨어 프로세서들에 의해 구현될 수도 있다. 다른 실시형태들에서, 비디오 인코더 (20) 및/또는

비디오 디코더 (30) 의 하나 이상의 컴포넌트들은 이러한 컴포넌트들의 TASK들을 수행하도록 구성된 하드웨어 회로에 의해 구현될 수도 있다.

[0067] **DSC 에서의 슬라이스들**

[0068] 위에서 언급된 바와 같이, 슬라이스는 일반적으로, 이미지 또는 프레임에서의 구역들의 나머지로부터의 정보 없이 독립적으로 디코딩될 수 있는 이미지 또는 프레임에서의 공간적으로 별개의 구역을 지칭한다. 각각의 이미지 또는 비디오 프레임은 단일 슬라이스에서 인코딩될 수도 있거나, 그것은 몇몇 슬라이스들에서 인코딩될 수도 있다. DSC 에서, 각각의 슬라이스를 인코딩하기 위하여 할당된 타겟 비트들은 실질적으로 일정할 수도 있다.

[0069] **블록 예측 모드**

[0070] 비디오 데이터의 단일 블록은 다수의 픽셀들을 포함할 수도 있고, 비디오 데이터의 각각의 블록은 블록이 코딩될 수 있는 다수의 잠재적인 코딩 모드들을 가진다. 이러한 코딩 모드들 중의 하나는 블록 예측 모드이다. 블록 예측 모드에서, 코더는 (예컨대, 현재 블록이 현재 슬라이스의 최초 라인에 있지 않을 경우에) 이전에 재구성된 라인에서 후보 블록을, 또는 코딩되어야 할 현재 블록에 (예컨대, 픽셀 값들에 있어서) 근접한 (예컨대, 현재 블록이 현재 슬라이스의 최초 라인에 있을 경우에) 동일한 라인에서 이전에 재구성된 블록들을 구하는 것을 시도한다. 일부 실시형태들에서, 픽셀 값들 사이의 근접성은 절대 차이들의 합 (Sum of Absolute Differences; SAD) 메트릭에 의해 결정된다. 코더는 (예컨대, 인코더 및 디코더의 양자에 알려진 미리 결정된 값일 수도 있는) 검색 범위에 의해 정의된 이전에 재구성된 블록들의 임의의 부분에서 후보 블록을 구하는 것을 시도할 수도 있다. 검색 범위는 인코더가 검색 코스트를 최소화하면서 양호한 정합을 구하기 위하여 검색 범위 내에서 잠재적인 후보들을 가지도록 정의된다. 블록 예측 모드의 코딩 효율은, 양호한 후보 (즉, 코딩되어야 할 현재 블록에 픽셀 값들에 있어서 근접한 것으로 결정되는 검색 범위 내의 후보) 가 탐색될 경우, 후보 블록과 현재 블록 사이의 (간차로서 알려진) 차이가 작을 것이라는 사실로부터 비롯된다. 작은 간차는 현재 블록의 실제적인 픽셀 값들을 시그널링하기 위하여 필요한 비트들의 수에 비해 시그널링하기 위하여 더 적은 비트들의 수를 취할 것이며, 이것에 의해, 더 낮은 RD 코스트로 귀착될 것이고, RD 메커니즘에 의해 선택되는 가능성을 증가시킬 것이다. 블록 예측 모드를 인에이블하는 것으로부터의 성능 증대는 어떤 타입들의 그래픽 콘텐츠에 대해서는 극도로 중요하다.

[0071] **블록 예측 모드에서의 파라미터들**

[0072] 블록 예측 모드는 특정된 검색 범위가 주어질 경우, 인코딩되어야 할 현재 블록으로부터의 최소 왜곡을 제공하는 후보 블록을 생성하도록 설계된다. 일부 실시형태들에서, 최소 왜곡은 SAD 를 이용하여 정의된다. 본 개시물의 일부 구현예들에서, 블록 예측 방법은 3 개의 파라미터들: 검색 범위 (search range; SR), 스큐 (skew) (α), 및 파티션 크기 (β) 에 의해 정의된다. 이 3 개의 파라미터들은 블록 예측 모드의 성능에 영향을 주고, 구현 동안에 튜닝될 (즉, 수정되거나 재구성됨) 수도 있다. 이 파라미터들은 인코더 및 디코더의 양자에 알려질 수도 있다.

[0073] **블록 예측 모드에서의 검색 공간**

[0074] 본 개시물의 일부 실시형태들에서, 검색 공간 (예컨대, 인코더가 후보 블록을 구하기 위하여 검색할 수도 있는 픽셀들의 공간적 위치들) 은 현재 블록의 특성들에 기초하여 상이할 수도 있다. 검색 공간은 모든 이전에 재구성된 블록들/픽셀들을 망라할 수도 있지만, 인코더 및/또는 디코더는 예를 들어, 연속 복잡도를 감소시키기 위하여, 후보 블록에 대한 검색을 검색 공간 내의 특정된 부분 (예컨대, 비트스트림에서 미리 정의되거나, 또는 시그널링되는 하나 이상의 파라미터들에 의해 정의된 "검색 범위") 으로 제한할 수도 있다. 블록 예측 검색 공간의 예들은 도 3 내지 도 6 에서 예시되어 있다. 도 3 및 도 4 는 현재 슬라이스의 최초 라인에 있지 않은 현재 블록 (예컨대, 현재 블록들 (308 및 408)) 을 수반하는 경우들을 예시한다. 도 5 및 도 6 은 현재 슬라이스의 최초 라인에 있는 현재 블록 (예컨대, 현재 블록들 (506 및 606)) 을 수반하는 경우들을 예시한다. 슬라이스에서의 최초 라인은 수직인 이웃들을 가지지 않으므로, 이 2 개의 경우들은 별도로 처리된다. 그러므로, 현재 라인으로부터의 재구성된 픽셀들은 검색 범위 (예컨대, 검색 범위들 (508 및 608)) 로서 레버리지로 활용될 수 있다. 본 개시물에서, 현재 슬라이스에서의 최초 라인은 FLS 로서 지칭될 수도 있고, 현재 슬라이스에서의 임의의 다른 라인은 NFLS 로서 지칭될 수도 있다.

[0075] 또한, 본원에서 설명된 블록 예측 기법들은 단일 라인 버퍼 (즉, 1-D 블록 크기) 를 이용한 코덱, 또는 다수 라인 버퍼들 (즉, 2-D 블록 크기) 을 이용한 코덱의 어느 하나에서 구현될 수도 있다. 코덱은 코정-비트 코덱

일 수도 있고, 여기서, 1-D 경우에 대한 검색 공간의 예들은 도 3 및 도 5 에서 도시되어 있고, 2-D 경우에 대한 검색 공간의 예들은 도 4 및 도 6 에서 도시되어 있다. 2-D 의 경우, 검색 범위는 이전 재구성된 라인 (예컨대, 이전 라인 (402)), 또는 2-D 블록에서의 라인들과 동일한 라인들로부터의 재구성된 블록들 (예컨대, 현재 블록 (606) 의 바로 좌측에 있는, 현재 라인 (602) 에서의 이전 (604)) 로부터의 픽셀들을 포함할 수도 있다. 2-D 블록은 수평으로 또는 수직으로 중의 어느 하나로, 또는 양자로 파티셔닝될 수도 있다. 블록 파티션들을 수반하는 경우에는, 블록 예측 벡터가 각각의 블록 파티션에 대하여 특정될 수도 있다.

[0076] **블록 예측 모드의 일 예의 구현예들**

[0077] 본 개시물의 일부 실시형태들에서는, SAD 이외의 왜곡 메트릭, 예컨대, 제공된 차이들의 합 (sum of squared differences; SSD) 이 이용될 수도 있다. 대안적으로 또는 추가적으로, 왜곡은 가중화에 의해 수정될 수도 있다. 예를 들어, YCoCg 컬러 공간이 이용되고 있을 경우, 코스트는 다음과 같이 계산될 수도 있다:

$$SAD(Y) + \frac{SAD(Co) + SAD(Cg)}{2}$$

[0078]

[0079] 본원에서 설명된 블록 예측 기법들은 RGB 또는 YCoCg 컬러 공간의 어느 하나에서 수행될 수도 있다. 게다가, 대안적인 구현예는 양자의 컬러 공간들을 이용할 수도 있고, 1-비트 플래그를 디코더로 시그널링할 수도 있어서, 2 개의 컬러 공간들 중의 어느 것이 선택되는지 (예컨대, 레이트 및 왜곡의 측면에서 최저 코스트는 가지는 어느 것인든지의 컬러 공간) 를 표시할 수도 있다.

[0080] FLS 에 관한 본 개시물의 일부 실시형태들에서, 직접적인 이전에 재구성된 블록 또는 블록들은 파이프라이닝 및 타이밍 제약들로 인해 검색 범위로부터 배제될 수도 있다. 예를 들어, 하드웨어 구현예에 따라서는, 코더가 현재 블록이 코더에 의해 프로세싱될 때까지 직접적인 이전에 재구성된 블록의 프로세싱을 완료하지 않았을 수도 있어서 (예컨대, 코더가 현재 블록의 프로세싱을 시작할 때, 이전 블록에 대한 재구성된 픽셀들이 알려지지 않을 수도 있음), 지연들 또는 실패들로 귀착될 수도 있다. 이러한 구현예에서는, (예컨대, 직접적인 이전에 재구성된 블록 또는 블록들을 배제함으로써) 이전에 재구성된 블록들의 이용을, 재구성된 픽셀 값들이 알려져 있는 그 블록들로 한정함으로써, 위에서 예시된 파이프라이닝 우려들이 해결될 수도 있다. NFLS 에 관한 본 개시물의 일부 실시형태들에서, 현재 블록 좌측의 검색 범위는 이전에 재구성된 라인이 아니라, 동일한 라인으로부터의 것일 수도 있다. 이러한 실시형태들의 일부에서는, 하나 이상의 이전에 재구성된 블록들이 파이프라이닝 및 타이밍 제약들로 인해 검색 범위로부터 배제될 수도 있다.

[0081] **NFLS 의 일 예의 구현예**

[0082] 도 3 에서 도시된 바와 같이, 블록 예측 방법은 현재 블록 (308) 에 대한 후보를 구하기 위하여 검색 공간에서 (그리고 도 4 의 검색 공간 (400) 에서와 유사하게) 검색 범위 (310) (SR) 를 통해 검색할 수도 있다. 인코딩되어야 할 현재 블록 (308) 의 최초 픽셀의 x-좌표 위치가 j 일 경우, 검색 공간 내의 모든 후보 블록들의 시작 위치들 k 의 세트는 다음과 같이 주어질 수도 있다:

$$k \in [j - SR + (\alpha + 1), j + \alpha]$$

[0083]

[0084] 이 예에서, 파라미터 α 는 인코딩되어야 할 현재 블록에 관련하여 검색 범위 (310) 의 x-좌표 위치를 스큐잉 (skewing) 한다. α 의 더 높은 값은 검색 범위 (310) 를 우측으로 시프트시키고, α 의 더 낮은 값은 검색 범위 (310) 를 좌측으로 시프트시킨다. 예를 들어, (i) 32 의 SR 및 15 의 α 는 이전 라인 (302) 의 중심에서 검색 범위 (310) 를 배치할 수도 있고, (ii) 32 의 SR 및 0 의 α 는 이전 라인 (302) 의 좌측 상에 검색 범위 (310) 를 배치할 수도 있고, (iii) 32 의 SR 및 31 의 α 는 이전 라인 (302) 의 우측 상에 검색 범위 (310) 를 배치할 수도 있다.

[0085] 본 개시물의 일부 구현예들에서, 검색 범위 내에 있지만 슬라이스 경계의 외부에 있는 픽셀은 그 픽셀에 대한 동적 범위의 절반으로 설정될 수도 있다. 예를 들어, 콘텐츠가 RGB888 일 경우, 128 의 디폴트 값은 R, G, 및 B 에 대하여 이용될 수도 있다. 콘텐츠가 YCoCg 공간에 있을 경우, 128 의 디폴트 값은 Y 에 대하여 이용될 수도 있고, 0 의 디폴트 값은 Co 및 Cg 에 대하여 이용될 수도 있다 (예컨대, Co 및 Cg 는 0 주위에 중심을 두는 9-비트 값들임).

[0086] **FLS의 일 예의 구현예**

[0087] 도 5에서 도시된 바와 같이, 검색 범위는 FLS 경우에 대하여 상이할 수도 있다. 이것은 수직인 이웃들이 이용가능하지 않거나, 이러한 수직인 이웃들이 현재 프레임의 외부에 있거나, 또는 이러한 수직인 이웃들이 상이한 슬라이스 내에 포함되기 때문이다. FLS 경우에 관한 본 개시물의 일부 실시형태들에서, 현재 라인에서의 픽셀들은 블록 예측을 위하여 이용될 수도 있다. 하나의 실시형태에서, 현재 블록의 좌측의 현재 라인에서의 임의의 픽셀은 검색 범위의 일부로서 고려될 수도 있다. 또 다른 실시형태에서, 하나 이상의 이전에 코딩된 블록들 (예컨대, 현재 블록의 바로 좌측에 있는 이전 블록 (504))은 파이프라이닝 및 타이밍 제약들로 인해 검색 범위로부터 배제될 수도 있다.

[0088] FLS의 일부 구현예들에서, 슬라이스의 최초 라인에서의 최초 몇몇 블록들에 대한 이용가능한 범위는 다른 블록들에 대하여 전형적으로 예상되는 검색 범위보다 더 작을 수도 있다. 이것은 후보 블록들에 대한 유효한 위치가 라인의 초반에서 시작하고 현재 블록 전에 종료되기 때문이다. FLS에서의 최초 몇몇 블록들에 대하여, 이 유효한 범위는 희망하는 범위 (예컨대, 32개 또는 64개의 위치들)보다 더 작을 수도 있다. 이에 따라, 이 블록들에 대하여, 검색 범위는 후보 블록의 각각의 블록 파티션이 검색 범위 내에 완전히 포함되도록 조절될 필요가 있을 수도 있다. NFLS에 대하여, 검색 범위는 검색 위치들의 총 수가 정의된 검색 범위 (예컨대, 32개 또는 64개의 픽셀 위치들)와 동일하도록, 좌측 또는 우측으로 시프트될 수도 있다. j 는 현재 블록에서의 최초 픽셀이므로, 현재 블록에서의 최종 픽셀은 $j + blkWidth - 1$ 일 것이다. 이 이유로, 검색 범위는 좌측으로 시프트된 ($blkWidth - 1$) 픽셀들일 필요가 있을 수도 있다.

[0089] FLS의 일부 구현예들에서, 인코딩되어야 할 현재 블록의 최초 픽셀의 x-좌표 위치가 j 로서 지칭될 경우, 검색 범위 내의 모든 후보 블록들의 시작 위치들의 세트는 다음과 같이 주어진다:

[0090] (i) 가장 최근의 이전에 재구성된 블록이 검색 범위의 일부, 예컨대, $\alpha = -1$ 일 경우:

[0091]
$$k \in [j - SR - (blkWidth - 1), j - 1 - (blkWidth - 1)]$$

[0092] (ii) n 개의 가장 최근의 이전에 재구성된 블록들이 검색 범위로부터 배제되어야 할 경우:

[0093]
$$k \in [j - (n \cdot blk_x + SR) - (blkWidth - 1), j - (n \cdot blk_x + 1) - (blkWidth - 1)]$$

[0094] 여기서, blk_x 는 블록 폭이다. 슬라이스 경계의 외부의 임의의 픽셀은 NFLS의 경우와 관련하여 위에서 설명된 바와 같이 디폴트 값으로 설정될 수도 있다. 또한, 스큐 파라미터는 FLS 경우와 연관될 필요가 있다는 것에 주목해야 한다.

[0095] **블록 예측 모드에서 코딩하기 위한 일 예의 플로우차트**

[0096] 도 7을 참조하면, 블록 예측 모드에서 비디오 데이터의 블록을 코딩하기 위한 일 예의 절차가 설명될 것이다. 도 7에서 예시된 단계들은 비디오 인코더 (예컨대, 도 2a에서의 비디오 인코더 (20)), 비디오 디코더 (예컨대, 도 2b에서의 비디오 디코더 (30)), 또는 그 컴포넌트 (들)에 의해 수행될 수도 있다. 편의를 위하여, 방법 (700)은 비디오 인코더 (20), 비디오 디코더 (30), 또는 또 다른 컴포넌트일 수도 있는 비디오 코더 (또한 간단하게 코더로서 지칭됨)에 의해 수행된 것으로서 설명된다.

[0097] 방법 (700)은 블록 (701)에서 시작된다. 블록 (705)에서, 코더는 현재 슬라이스에서 현재 블록을 예측하기 위하여 이용되어야 할 후보 블록을 결정한다. 후보 블록은 하나 이상의 블록 예측 파라미터들에 의해 정의된 위치들의 범위 내에 있을 수도 있다. 예를 들어, 블록 예측 파라미터들은 (i) 위치들의 범위의 크기를 정의하는 검색 범위 파라미터, (ii) 현재 블록에 대한 위치들의 범위의 상대적 위치를 정의하는 스큐 파라미터, 및 (iii) 현재 블록에서의 각각의 파티션의 크기를 정의하는 파티션 크기 파라미터를 포함할 수도 있다. 본 개시물의 일부 실시형태들에서, 검색 범위 파라미터, 스큐 파라미터, 및 파티션 크기 파라미터의 각각은 시간적으로가 아니라 공간적으로, 후보 블록의 위치들을 정의한다.

[0098] 블록 (710)에서, 코더는 후보 블록 및 현재 블록에 기초하여 예측 벡터를 결정한다. 예측 벡터는 현재 블록에 대한 후보 블록의 위치를 식별할 수도 있다. 예측 벡터는 하나 이상의 좌표 값들 (예컨대, 1-D 공간에서 오프셋을 표시하는 좌표 값)을 포함할 수도 있다. 블록 (715)에서, 코더는 예측 벡터를 시그널링하는 것을 통해 블록 예측 모드에서 현재의 블록을 적어도 부분적으로 코딩한다. 일부 실시형태들에서, 코더는

또한, 후보 블록과 현재 블록 사이의 잔차를 시그널링할 수도 있다. 비트 절감은 현재 블록의 실제적인 픽셀 값들을 시그널링해야 하는 대신에, 후보 블록의 위치를 식별하는 예측 벡터와, 현재 블록과 후보 블록 사이의 차이를 나타내는 잔차를 시그널링함으로써 달성될 수도 있다. 방법 (700) 은 블록 (720) 에서 종료된다.

[0099] 방법 (700) 에서, 도 7 에서 도시된 블록들 중의 하나 이상은 제거 (예컨대, 수행되지 않음) 될 수도 있고, 및/또는 방법이 수행되는 순서는 전환될 수도 있다. 일부 실시형태들에서는, 추가적인 블록들이 방법 (700) 에 추가될 수도 있다. 본 개시물의 실시형태들은 도 7 에서 도시된 예로, 또는 도 7 에서 도시된 예에 의하여 제한되지 않고, 다른 변형들이 이 개시물의 사상으로부터 이탈하지 않으면서 구현될 수도 있다.

[0100] **후보 블록을 구한 후**

[0101] 최상의 후보 블록이 결정된 후, 후보 블록의 픽셀 값들은 현재 블록의 픽셀 값들로부터 감산되어, 잔차로 귀착된다. 잔차는 블록 예측 모드와 연관된 사전-선택된 QP 에 기초하여 양자화될 수도 있다. 양자화된 잔차는 (고정-길이 또는 가변-길이의 어느 하나일 수 있는) 코드북을 이용하여 인코딩될 수도 있고, 고정-길이 코드 또는 가변-길이 코드를 이용하여 시그널링될 수도 있다. 선택된 코드북은 코딩 효율 및 하드웨어 복잡도 요건들에 기초할 수도 있다. 예를 들어, 선택된 코드북은 Exp-Golomb 코드북일 수도 있다. 본 개시물의 일부 실시형태들에서는, 기존의 DSC 구현예들의 델타 크기 유닛 가변 길이 코딩 (delta size unit variable length coding; DSU-VLC) 과 유사한 엔트로피 코딩 방식이 이용될 수도 있다. 일부 실시형태들에서, 잔차는 위에서 설명된 양자화 전에 (예컨대, 직접 코사인 변환, 하다마르 변환 (Hadamard transform), 또는 다른 알려진 변환들을 이용하여) 변환될 수도 있다.

[0102] 본 개시물의 일부 실시형태들에서, 현재 블록의 잔차에서의 샘플들은 다수의 그룹들로 파티셔닝될 수도 있다 (예컨대, 16 개의 샘플들을 포함하는 블록에 대하여 그룹 당 4 개의 샘플들). 블록에서의 모든 계수들이 제로일 경우, 블록의 잔차는 스킵 모드 (skip mode) 를 이용하여 코딩되고, 즉, 블록 당 (컴포넌트 당) 1-비트 플래그는 블록에서의 현재 컴포넌트가 스킵 모드를 이용하여 코딩되는지 아닌지를 표시하기 위하여 시그널링된다. 적어도 하나의 비-제로 (non-zero) 값이 블록 내에 포함될 경우, 각각의 그룹은 그룹이 하나의 비-제로 값을 가질 경우에만, DSU-VLC 를 이용하여 코딩될 수도 있다. 그룹 (예컨대, 잔차에서의 16 개의 샘플들 중의 4 개의 샘플들) 이 임의의 비-제로 값들을 포함하지 않을 경우, 그룹은 스킵 모드를 이용하여 코딩되고, 즉, 그룹 당 1-비트 플래그는 그룹이 스킵 모드를 이용하여 코딩되는지 아닌지를 표시하기 위하여 시그널링된다. 더욱 구체적으로, 각각의 그룹에 대하여, 검색은 그룹에서의 모든 값들이 제로인지 여부를 결정하기 위하여 수행될 수도 있다. 그룹에서의 모든 값들이 제로일 경우, '1' 의 값이 디코더로 시그널링될 수도 있고; 그렇지 않을 경우 (적어도 하나의 값이 비-제로일 경우), '0' 의 값은 디코더로 시그널링될 수도 있고, DSU-VLC 코딩의 코딩이 뒤따른다. 대안적인 예에서, '0' 의 값은 그룹에서의 모든 값들이 제로일 경우에 시그널링될 수도 있고, '1' 의 값은 그룹이 적어도 하나의 비-제로 값을 포함할 경우에 시그널링될 수도 있다.

[0103] 본 개시물의 일부 실시형태들에서, 최상의 후보 블록은 최상의 오프셋을 포함하는 고정-길이 코드를 송신함으로써 디코더로 명시적으로 시그널링된다. 오프셋은 "벡터" 로서 지칭될 수도 있다. 벡터를 디코더로 명시적으로 시그널링하는 장점은, 디코더가 블록 검색 자체를 수행할 필요가 없을 것이라는 것이다. 오히려, 디코더는 벡터를 명시적으로 수신할 것이고, 현재 블록의 픽셀 값들을 결정하기 위하여 후보 블록을 디코딩된, 역-양자화된 잔차 값들에 추가할 것이다.

[0104] **블록 파티셔닝**

[0105] 본 개시물의 일부 실시형태들에서, 코딩되어야 할 현재 블록은 파티셔닝될 수도 있어서, 다수의 후보 블록들 및 블록 당 다수의 벡터들로 귀착될 수도 있다. 이러한 실시형태들의 일부에서는, 벡터 (들) 가 고정-길이 코드를 이용하여 명시적으로 시그널링될 수도 있다. 예를 들어, 이 고정-길이 코드의 길이는 $\log_2(SR)$ 일 수도 있다. 또 다른 실시형태에서, 벡터 (들) 는 지수-골롬 (Exponential-Golomb) 또는 골롬-라이스 (Golomb-Rice) 코드 계열들로부터의 코드와 같은 가변-길이 코드를 이용하여 명시적으로 시그널링될 수도 있다. 이 코드북은 벡터 (들) 와 연관된 통계적 분포에 기초하여 선택될 수 있다. 또 다른 실시형태에서, 벡터 (들) 는 이전-코딩된 벡터 (들) 에 기초하여 예측될 수도 있고, 벡터 (들) 의 잔차는 일부 고정-길이 또는 가변-길이 코드를 이용하여 코딩될 수도 있다. 또 다른 실시형태에서, 벡터 (들) 는 이전-코딩된 벡터 (들) 에 기초하여 예측될 수도 있고, 1-비트 플래그는 2 개의 벡터들이 동일한지 여부를 시그널링하기 위하여 이용될 수도 있다. 이 플래그는 *SameFlag* 로서 지칭될 수도 있다. *SameFlag* = 1 일 경우, 벡터 값 자체는 디코더로 시그널링될 필요가 없다. *SameFlag* = 0 일 경우, 벡터는 (예컨대, 고정-길이 또는 가변-길이 코드의 어느

하나를 이용하여) 명시적으로 시그널링될 것이다. 일 예의 블록 파티셔닝 방식은 도 8 에서 예시되어 있다.

[0106] 도 8 에서 도시된 바와 같이, 현재 블록 (802) 은 단일 파티션을 포함한다. 현재 블록 (802) 에 대하여 시그널링된 정보는 모드 헤더, 벡터 *SameFlag*, 벡터 A, 및 페이로드를 포함한다. 현재 블록 (804) 은 2 개의 파티션들, 파티션 A 및 파티션 B 를 포함한다. 현재 블록 (804) 에 대하여 시그널링된 정보는 모드 헤더, 벡터 *SameFlag*, 벡터 A, 벡터 *SameFlag*, 벡터 B, 및 페이로드를 포함한다. 위에서 설명된 바와 같이, 위에서 열거된 하나 이상의 항목들은 시그널링되지 않을 수도 있다. 예를 들어, 벡터 *SameFlag* 가 1 과 동일한 경우, 다음의 벡터는 시그널링될 필요가 없다.

[0107] 파티션 크기 β 는 별도의 서브-블록들로의 현재 블록의 파티셔닝을 결정할 수도 있다. 이러한 경우, 별도의 블록 파티션은 각각의 서브-블록에 대하여 수행될 수도 있다. 예를 들어, 블록 크기가 $N = 16$ 이고 파티션 크기 $\beta = 8$ 일 경우, 검색은 $16 / 8 = 2$ 개의 파티션들의 각각에 대하여 수행될 것이다. 또 다른 예에서, $\beta = N$ 일 경우, 블록 파티셔닝은 디스에이블된다. $\beta < N$ 일 경우, 각각의 벡터는 디코더로 명시적으로 시그널링될 수도 있다. (예컨대, 현재 벡터들을 정의하기 위하여 이전에 시그널링된 벡터들을 이용한) 벡터 예측이 채용되지 않을 경우, 각각의 벡터는 고정-길이 또는 가변-길이 코드를 이용하여 시그널링될 것이다. 벡터 예측이 채용될 경우, 최초 벡터는 (예컨대, 메모리 내에 저장된) 이전에 코딩된 벡터로부터 예측될 수도 있고, $n > 0$ 에 대하여, 벡터 n 은 벡터 $n - 1$ 로부터 예측된다.

[0108] **블록 예측 모드에서의 가변 파티션 사이즈**

[0109] 상기 예들은, (예컨대, 1 픽셀의 높이 및 8 픽셀들의 폭을 갖는) 1×8 또는 (예컨대, 2 픽셀들의 높이 및 8 픽셀들의 폭을 갖는) 2×8 의 사이즈를 갖는 블록들이 어떻게 블록 예측 모드에서 코딩될 수도 있는지를 나타낸다. 도 8 에서 도시된 바와 같이, 블록은 다수의 구역들로 파티셔닝될 수도 있고, 각각의 구역은 상이한 파티셔닝 방식들을 이용하여 (예컨대, 1×2 파티션들을 이용하여, 2×2 파티션들을 이용하여 등) 코딩될 수 있으며, 블록 예측 벡터는 각 파티션에 대해 특정될 수도 있다 (예컨대, 각 파티션과 연관된 잔차오 함께 비트스트림에서 시그널링될 수도 있다). 예를 들어, 각 블록은 2 개의 픽셀들 (또는 다른 고정된 사이즈들의 파티션들) 을 포함하는 다수의 1×2 파티션들로 파티셔닝될 수도 있다.

[0110] 다른 실시형태들에서, 인코더는 각 블록에 대해 (블록 내의 각 서브-구역에 대해) 가장 효율적인 블록 파티션 사이즈를 결정할 수도 있다. 효율성은 주어진 블록 파티션 사이즈를 이용하여 블록 (또는 그 안의 서브-구역) 을 코딩하는 것과 연관된 레이트 및 왜곡에 기초하여 측정될 수도 있다. 예를 들어, 4 개의 2×2 구역들을 포함하는 블록을 코딩할 때, 인코더는, 단일 파티션들 (예컨대, 각각이 2×2 구역에 대해 단일의 2×2 파티션) 을 이용하여 처음 3 개의 2×2 구역들을 코딩하고, 2 개의 파티션들 (예컨대, 2 개의 1×2 파티션들) 을 이용하여 4 번째 2×2 구역을 코딩함으로써 가장 큰 코딩 효율이 달성될 수 있음을 결정할 수도 있다. 인코더로 하여금 각 블록에 대한 파티션 사이즈를 적응적으로 선택하도록 허용함으로써, 블록 예측 방식의 퍼포먼스가 추가로 향상될 수 있다. 이것은, 큰 파티션들이 평활한 구역들 (예컨대, 구역에 걸쳐 픽셀 값들에서 변화의 임계량 보다 더 적거나 아무런 변화도 보이지 않는 구역들) 에 대해 사용될 수 있고, 이에 의해, (예컨대, 구역의 사이즈에 비해) 단일 블록 예측 벡터들에 대해 보다 적은 비트들을 필요로 하기 때문이고, 한편, 보다 작은 파티션들을 이용하는 것은 복잡한 구역들에 대해 사용될 수 있다 (여기서, 왜곡 및/또는 엔트로피 코딩 레이트에서의 감소는 추가적인 시그널링 코스트를 능가한다). 예를 들어, 인코더는 주어진 구역 또는 블록이 평활화 임계 조건을 만족하는지 여부를 결정할 수도 있고, 주어진 구역 또는 블록이 평활화 임계 조건을 만족한다고 결정하는 것에 응답하여, 더 큰 파티션 사이즈를 이용하여 블록 예측 모드에서 주어진 구역 또는 블록을 인코딩할 수도 있다 (그리고 그렇지 않으면, 더 작은 파티션 사이즈를 이용하여 블록 예측 모드에서 주어진 구역 또는 블록을 인코딩할 수도 있다). 다른 예로서, 인코더는, 주어진 구역 또는 블록이 복잡도를 만족하는지 여부를 결정할 수도 있고, 주어진 구역 또는 블록이 복잡도 임계 조건을 만족한다고 결정하는 것에 응답하여, 더 작은 파티션 사이즈를 이용하여 블록 예측 모드에서 주어진 구역 또는 블록을 인코딩할 수도 있다 (그리고 그렇지 않으면, 더 큰 파티션 사이즈를 이용하여 블록 예측 모드에서 주어진 구역 또는 블록을 인코딩할 수도 있다). 상이한 파티션 사이즈들을 적응적으로 선택하기 위한 능력은, 블록 예측 모드로 하여금 더 큰 범위의 콘텐츠 타입들 (예컨대, 그래픽 콘텐츠, 자연적 이미지들, 테스트 패턴들, 정세한 텍스트 렌더링 등) 에서 사용되도록 허용할 수도 있다.

[0111] **블록 예측 모드에서 코딩의 예시적인 데이터 플로우**

[0112] 도 9 는 적응적 파티션 사이즈를 이용하여 블록 예측 모드에서 블록을 코딩하기 위한 예시적인 데이터 플로우 (900) 를 나타낸다. 도 9 에서 예시된 바와 같이, 블록 예측 모드에서 예측될 현재 블록 (902) 은 블록 파

티션 (904) 을 포함한다. 하나의 예에서, 블록 파티션은 1x2 또는 2x2 의 사이즈를 갖는다. 블록 예측 (block prediction; BP) 검색 (906) 은 블록 예측 모드에서 현재 블록 (902) (또는 블록 패턴 (904)) 을 예측하기 위해 이용가능하고 이미 코딩된 블록 또는 파티션을 식별하기 위해 수행된다. 도 9 에서 도시된 바와 같이, BP 검색 (906) 은, 예를 들어, 이전 라인 (예컨대, 바로 선행하는 라인 또는 다른 선행하는 라인과 같이, 현재 블록을 포함하는 현재 라인을 코딩하기 이전에 코딩된 라인) 에서의 하나 이상의 이전에 재구성된 블록들 (907A) 및/또는 현재 라인 (예컨대, 현재 블록을 포함하는 라인) 으로부터 이전의 재구성된 블록들 (907B) 을 포함하는, 검색 범위 내에서 검색할 수도 있다.

[0113] 인코더는 검색 범위에서 식별된 후보 블록 또는 파티션에 기초하여 블록 예측자 (908) 를 결정한다. 블록 예측자 (908) 는 블록 (910) 에서 현재 블록 (902) (또는 후보 블록 (902) 내의 현재 블록 파티션 ((04)) 으로부터 감산되고, 그 감산에 기초하여 결정된 잔차는 블록 (912) 에서 양자화된다. 양자화된 잔차는 엔트로피 코더 (920) 에 의해 엔트로피 코딩된다. 또한, 역 양자화 (914) 가 양자화된 잔차에 대해 수행되고, 그 결과는 블록 (916) 에서 블록 예측자에 가산되어 재구성된 블록 (918) 을 생성한다. BP 파티션 사이즈 선택 (922) 은 재구성된 블록 (918) 의 왜곡 퍼포먼스 (D) 및 엔트로피 인코딩된 잔차의 레이트 퍼포먼스 (R) 에 기초하여 수행된다. 비트스트림 (924) 은 선택된 BP 사이즈에 기초하여 생성된다.

[0114] 예를 들어, BP 파티션 사이즈 선택 (922) 은, 현재 블록 (902) 내의 각 파티션 구역 (예컨대, 2x2) 의 레이트 (예컨대, R) 및 왜곡 (예컨대, D) 을 입력으로서 취하고, 2 개의 옵션들 사이의 RD 트레이드오프에 기초하여 예측을 위해 단일 블록 예측 벡터 (block prediction vector; BPV) (예컨대, 단일 2x2 파티션에 대해 총 1 BPV) 를 이용하여 코딩되어야 하는지 또는 다수의 BPV 들 (예컨대, 총 2 BPV 들, 2 개의 1x2 파티션들에 대해 각각 1 BPV) 을 이용하여 파티셔닝 및 코딩되어야 하는지 여부를 결정할 수도 있다. 비록 본 명세서에서 논의된 일부 예들은 2x2 의 파티션 구역 사이즈를 수반하지만 (이에 의해 선택가능한 옵션들로서 1x2, 2x1, 및 2x2 의 파티션 사이즈를 가짐), 인코더에 의해 선택가능한 파티션 사이즈들은 이러한 예들 (예컨대, 1x2 및 2x2) 에서 사용되는 것들에 제한되지 아니하고, 블록 사이즈 및/또는 구역 사이즈에 기초하여 다른 사이즈들 (예컨대, 2x1) 을 포함할 수도 있다.

[0115] 일부 실시형태들에서, 파티션 사이즈들은 고정된다 (예컨대, 1x2, 2x2, 또는 현재 파티션 구역 또는 블록에서의 픽셀들의 임의의 다른 하위-조합). 예를 들어, 블록은 2x8 의 블록 사이즈를 가질 수도 있고, 블록은 2x2 의 사이즈를 갖는 서브-블록들 또는 구역들로 나뉘질 수도 있다. 2x8 블록 내의 2x2 서브-블록들 또는 구역들은 1x2 의 사이즈를 갖는 파티션들로 추가로 파티셔닝될 수도 있다. 이러한 예에서, 각각의 1x2 파티션은 다른 파티션들과는 독립적으로 단일 BPV 를 이용하여 예측될 수도 있다. 다른 실시형태들에서, 파티션 사이즈들은 가변적이고, 어느 파티션 사이즈들을 이용하여 블록 예측에서 각 블록, 서브-블록, 및/또는 구역이 어떻게 코딩되는지가 각 파티셔닝 방식의 레이트 및 왜곡 퍼포먼스에 기초하여 인코더에 의해 결정될 수도 있다. 예를 들어, 현재 블록 내의 2x2 구역 (예컨대, 현재 구역) 에 대해, 현재 구역을 2 개의 1x2 파티션들로 분할함으로써 예측하는 것, 및 2 개의 1x2 파티션들을 별도로 2 개의 BPV 들 (각각 정의된 검색 범위 내에서 이전에 코딩된 1x2 파티션들을 포인팅) 을 이용하여 예측하는 것이 (예컨대, 2x2 와 같은 다른 파티셔닝 방식들에 비해) 더 양호한 레이트 및/또는 왜곡 퍼포먼스를 가져오는 경우에, 현재 구역은 1x2 파티셔닝 방식을 이용하여 예측될 수도 있다. 다른 한편, 현재 구역을 하나의 BPV (정의된 검색 범위 내에서 이전에 코딩된 2x2 파티션을 포인팅) 를 이용하여 단일의 2x2 파티션들로서 예측하는 것이 (예컨대, 1x2 와 같은 다른 파티셔닝 방식들에 비해) 더 양호한 레이트 및/또는 왜곡 퍼포먼스를 가져오는 경우에, 현재 구역은 2x2 파티셔닝 방식을 이용하여 예측될 수도 있다. 예측 모드에서 블록을 코딩하기 위해 사용될 파티셔닝 방식을 결정하는 프로세스는 도 14 를 참조하여 이하 보다 자세하게 설명된다.

[0116] **블록 사이즈들 및 서브-블록 사이즈들**

[0117] MxN 의 블록 사이즈에 대해, 일부 실시형태들은 사이즈 $M_{sub} \times N_{sub}$ 의 서브-블록들 (또한 본 명세서에서 구역들로서 지칭됨) 을 참조하여 설명되고, 여기서, $M_{sub} = M$ 및 $N_{sub} = N$ 이다. 일부 구현들에서, 계산의 용이성을 위해, 양 M_{sub} 및 N_{sub} 는 MxN 블록 내의 엔트로피 코딩 그룹들과 정렬된다. 블록 내의 각 서브-블록 $M_{sub} \times N_{sub}$ 은 (i) 추가로 파티셔닝됨이 없이 단일의 BPV 를 이용하여 예측되거나 또는 (ii) BPV 가 각 파티션에 대해 사용되어 다수의 파티션들 (예컨대, 2 개의 1x2 파티션들로) 파티셔닝되는 것 중 어느 일방일 수도 있다. 전체 서브-블록에 대해 단일 BPV 를 이용하는 것 또는 서브-블록을 각각이 그 자신의 BPV 를 갖는 파티션들로 파티셔닝하는 것 사이의 효과적인 트레이드-오프는, 보다 많은 BPV e들을 시그널링하는 것이 비트스트림에서 가외의 레이트를 초래할 것이고, 하지만, 보다 많은 BPV 들을 이용함으로써, 왜곡 및 엔트로피 코딩 레이트들이

감소될 수도 있다는 것이다. 달리 말하면, 단일의 추가적인 BPV 들에 대해 보다 많은 비트들을 사용함으로써, 잔차 (후보 블록/구역과 현재 블록/구역 사이의 차이) 를 시그널링하기 위해 사용되는 비트들의 수가 감소될 수도 있고, 이는 추가적으로 엔트로피 코딩을 위해 사용되는 비트들의 수가 역시 감소되게 할 수도 있다.

인코더는 RD 코스트의 면에서 각 옵션 (예컨대, 파티션 없음 vs. 다수의 파티션들) 을 비교하고, 코스트 비교에 기초하여 각 서브-블록 또는 구역을 파티셔닝할지 또는 파티셔닝하지 않을지 여부를 선택하거나, 최상의 RD 퍼포먼스를 제공하는 복수의 파티셔닝 방식들로부터 파티셔닝 방식을 선택할 수도 있다.

[0118] **예시적인 파티셔닝 방식**

[0119] 도 10 은 예시적인 파티셔닝 방식을 나타내는 도 (1000) 를 나타낸다. 도 10 에서, 2x2 서브-블록 또는 구역에 대한 2 개의 파티셔닝 옵션들이 예시된다. 이 예에서, (예컨대, 픽셀들 X_0 - X_{15} 을 포함하는) 블록 (1002) 은 2x8 의 사이즈를 가지고, 그 블록 내의 (예컨대, 픽셀들 X_0 , X_1 , X_8 , 및 X_9 을 포함하는) 서브-블록 또는 구역 (1004) 은 2x2 의 사이즈를 갖는다. 파티셔닝 옵션 (1006) 은 서브-블록 또는 구역 (1004) 이 단일 BPV 를 이용하여 예측되는 예를 나타내고, 파티셔닝 옵션 (1008) 은, 서브-블록 또는 구역 (1004) 이 서브-블록 또는 구역 (1004) 내의 각각의 1x2 파티션에 대해 2 개의 BPV 들을 이용하여 예측되는 예를 나타낸다. 2x2 의 사이즈를 갖는 서브-블록들 또는 구역들은, 도 11 에서 도시된, 블록 예측 모드를 위해 엔트로피 코딩 그룹 구조들 (1100) 과 서브-블록들 또는 구역들을 정렬시키기 위해 어드밴스드 DSC (Adv-DSC) 와 같은 일부 구현들에서 사용된다. 도 11 의 예에서, 엔트로피 코딩 그룹들 0, 1, 2, 및 3 이 예시되고, 각각은 블록 내의 4 개의 2x2 서브-블록들 또는 구역들 중 하나에 대응한다. 하지만, 본 명세서에서 기술된 기법들은 이러한 실시형태에 한정되지 않고, 임의의 블록 사이즈 $M \times N$ 및 임의의 서브-블록 사이즈 $M_{sub} \times N_{sub}$ 로 확장될 수도 있다. 하지만, 이하 예시되는 예들에서, 파라미터들 $M = 2$, $N = 8$, $M_{sub} = 2$, $N_{sub} = 2$ 이 사용된다. 일부 실시형태들에서, 서브-블록들 및/또는 파티셔닝 방식들은 엔트로피 코딩 그룹들에 기초하여 결정될 수도 있다. 예를 들어, 서브-블록들 및/또는 파티셔닝 방식들은, 각 서브-블록 및/또는 파티셔닝 방식이 단일의 엔트로피 코딩 그룹 내에 포함되도록 결정될 수도 있다.

[0120] **파티션 사이즈 결정**

[0121] 인코더는 최소 RD 코스트에 기초하여, (i) 각각의 2x2 구역을 단일의 2x2 파티션으로서 코딩할지 또는 (ii) 그 구역을 2 개의 1x2 파티션들로 나누고 각각의 1x2 파티션을 별도로 코딩할지 여부를 결정할 수도 있다. RD 코스트는 이하에 나타난 바와 같이 계산될 수도 있다:

$$cost(2x2) = D_{2x2} + \lambda \cdot R_{2x2}$$

$$cost(1x2) = D_{1x2} + \lambda \cdot R_{1x2}$$

$$R_{2x2} = 1 + BPV_{bits} + EC_{bits2x2}$$

[0122]
$$R_{1x2} = 1 + (2 \cdot BPV_{bits}) + EC_{bits1x2}$$

[0123] 일부 구현들에서, BPV 는 $\log_2(SR)$ 과 동일한, 고정 수의 비트들 (BPV_{bits}) 로 시그널링되고, 여기서, SR 은 블록 예측 모드와 연관된 검색 공간 (또는 검색 범위) 이다. 예를 들어, 검색 공간이 64 포지션들로 이루어지는 경우에, $\log_2(64) = 6$ 비트들이 각각의 BPV 를 시그널링하기 위해 사용된다.

[0124] 가변 파티션 사이즈를 이용한 블록 예측을 위한 검색 공간은 도 3 내지 도 6 을 참조하여 논의된 검색 범위와 약간 상이할 수도 있다. 특히, $M_{sub} \times N_{sub}$ 은 높이 M_{sub} 를 갖는 검색 공간을 이용할 수도 있다. 이러한 경우들에서, 가변 파티션 사이즈 없이 블록 예측에 대한 가변 파티션 사이즈로 블록 예측을 구현하기 위해 추가적인 라인 버퍼들이 필요할 수도 있다. 이러한 검색 공간의 예는 2x2 의 서브-블록 사이즈에 대해 도 12 에서 예시된다. 도 12 는 예시적인 검색 범위를 나타내는 도 (1200) 를 나타낸다. 도 12 에서 도시된 바와 같이, 현재 라인 (1202) 은 (i) 현재 서브-블록 (1206) 을 갖는 현재 블록 (1204) 및 (ii) 이전 블록 (1208) 을 포함한다. 도 12 의 예에서, 이전 라인 (1210) 은 검색 범위 (1212) 를 포함하고, 이 검색 범위 (1212) 로부터 인코더는 현재 서브-블록 (1206) 을 예측하기 위한 후보 서브-블록 (1214) 을 선택할 수도 있다.

1-D 파티션들 (예컨대, 1x2) 에 대한 검색 범위 또는 공간은, 단일의 이전 재구성된 라인에 의존하는, 도 3 을 참조하여 이전에 설명된 검색 범위와 유사할 수도 있다.

[0125] 일부 실시형태들에서, 왜곡들 D_{2x2} 및 D_{1x2} 은 YCoCg 컬러 공간에서 수정된 절대 차이들의 합 (sum of absolute

differences; SAD) 을 이용하여 계산될 수도 있다. 예를 들어, YCoCg 컬러 공간에서의 (예컨대, 현재 서브-블록 또는 파티션에서의) 픽셀 A 와 (예컨대, 후보 서브-블록 또는 구역에서의) 픽셀 B 사이의 SAD 왜곡은 다음과 같이 계산될 수도 있다:

$$SAD(A, B) = |A_y - B_y| + 0.5 \cdot |A_{Co} - B_{Co}| + 0.5 \cdot |A_{Cg} - B_{Cg}|$$

[0126]

[0127] 현재 서브-블록 또는 파티션이 하나보다 많은 픽셀을 가지는 경우에, 전체 서브-블록 또는 파티션에 대한 왜곡은 현재 서브-블록 또는 파티션에서의 각 픽셀에 대해 계산된 개별 SAD 들을 합산함으로써 계산될 수도 있다. 서브-블록 또는 파티션의 픽셀 값들은 실제 픽셀 값 또는 (예컨대, 후보 예측자 및 잔차에 기초하여 계산된) 재구성된 픽셀 값일 수도 있다. 일부 구현들에서, 람다 파라미터는 2 의 값으로 고정될 수도 있다. 다른 구현들에서, 이 파라미터는 블록 사이즈, 비트레이트, 또는 다른 코딩 파라미터들에 의존하여 튜닝될 수도 있다.

[0127]

[0128]

엔트로피 코딩 코스트 EC_{bits} 는 각각의 2x2 구역에 대해 계산될 수도 있다. 각각의 엔트로피 코딩 그룹에서의 4 개의 샘플들은 단일의 BPV 로부터 예측된 2x2 양자화된 잔차 (예컨대, 2x2 파티션), 또는 2 개의 벡터들 (예컨대, 2 개의 1x2 파티션들) 을 이용하는 2x2 양자화된 잔차 중 어느 일방으로부터 나올 수도 있다. 예를 들어, 엔트로피 코딩 코스트는 (예컨대, 벡터(들) 및 잔차를 포함하는) 비트스트림에서 각각의 엔트로피 코딩 그룹을 시그널링하기 위해 필요한 비트들의 수를 표현할 수도 있다. 계산된 엔트로피 코딩 코스트들에 기초하여, 인코더는 각각의 2x2 구역에 대해 최저 코스트를 갖는 파티셔닝 방식을 선택할 수도 있다. 비록 2x2 서브-블록 사이즈들, 2x2 엔트로피 코딩 그룹들, 및 2 개의 파티셔닝 방식들 (1x2 및 2x2) 을 갖는 2x8 블록들을 참조하여 일부 실시형태들이 논의되지만, 본 명세서에서 설명된 기법들은 다른 블록 사이즈들, 서브-블록 사이즈들, 엔트로피 코딩 그룹들, 및/또는 파티셔닝 방식들로 확장될 수도 있다.

[0129]

비트스트림에서의 코딩 정보 시그널링

[0130]

도 10 에서 도시된 2x8 블록 (1002) 에서, 4 개의 2x2 구역들의 각각은 상기 논의된 RD 코스트 분석에 기초하여 파티셔닝될 수도 있다. 예를 들어, 각각의 2x2 구역은 단일의 2x2 파티션 또는 2 개의 1x2 파티션들 중 어느 일방으로 파티셔닝될 수도 있다. 이러한 파티셔닝의 4 개의 예들이 도 13 의 도 (1300) 에 의해 예시된다. 도 13 에서 도시된 바와 같이, 블록 (1302) 은 2x2 파티셔닝 방식에 기초하여 예측된 4 개의 서브-블록들을 가지고, 블록 (1304) 은 2x2 파티셔닝 방식에 기초하여 예측된 3 개의 서브-블록들 및 1x2 파티셔닝 방식에 기초하여 예측된 하나의 서브-블록들을 가지며, 블록 (1306) 은 1x2 파티셔닝 방식에 기초하여 예측된 4 개의 서브-블록들을 가지고, 블록 (1308) 은 2x2 파티셔닝 방식에 기초하여 예측된 하나의 서브-블록들 및 1x2 파티셔닝 방식에 기초하여 예측된 3 개의 서브-블록들을 갖는다. 디코더에 대해 BPV 들을 시그널링하는 것에 추가하여, 인코더는 또한, 디코더가 파티셔닝을 적절하게 추론할 수 있도록 각각의 2x2 구역에 대해 하나의 비트를 전송할 수도 있다. Adv-DSC 구현과 같은 일부 구현들에서, 블록 내의 각각의 구역 (예컨대, 2x8 블록에서의 각각의 2x2 구역) 에 대해 선택된 파티셔닝 방식을 나타내는 4 개의 비트들의 그룹이 비트스트림에서 시그널링된다. 이러한 구현들에서, 4 개의 비트들 "1011" 은, 블록에서의 제 1, 제 3, 및 제 4 구역 (예컨대, 2x2 서브-블록) 이 (예컨대, 1x2 파티션들에 기초하여) 제 1 파티셔닝 방식에 기초하여 예측되거나 코딩될 것인 한편, 제 2 구역 (예컨대, 2x2 서브-블록) 은 (예컨대, 2x2 파티션들에 기초하여) 제 2 파티셔닝 방식에 기초하여 예측되거나 코딩될 것임을 나타낼 수도 있다. 일부 실시형태들에서, 비트스트림에서 이들 4 개의 비트들에 이어서, BPV 들이 BPV 당 고정된 비트들을 이용하여 시그널링될 수도 있다. 이전의 예 (예컨대, "1011" 의 비트 시퀀스) 에서, 7 개의 BPV 들이 시그널링될 수도 있다.

[0131]

블록 예측 모드에서의 코딩을 위한 예시적인 플로우차트

[0132]

도 14 를 참조하여, 블록 예측 모드에서 비디오 데이터의 블록을 코딩하기 위한 예시적인 프로시저가 설명될 것이다. 도 14 에서 예시된 단계들은 비디오 인코더 (예컨대, 도 2a 에서의 비디오 인코더 (20)) 또는 그것의 컴포넌트(들)에 의해 수행될 수도 있다. 편의를 위해, 방법 (1400) 은 비디오 인코더 (20) 또는 다른 컴포넌트일 수도 있는 코더에 의해 수행되는 것으로서 설명된다.

[0133]

방법 (1400) 은 블록 (1401) 에서 시작한다. 블록 (1405) 에서, 코더는 제 1 파티셔닝 방식에 기초하여 (예컨대, 블록 예측 모드에서 코딩되는 비디오 데이터의 블록 내의) 현재 구역을 예측하기 위해 사용될 하나 이상의 제 1 후보 구역들을 결정한다. 예를 들어, 현재 구역은 2x8 블록에서의 2x2 구역들 중 하나일 수도 있다. 제 1 파티셔닝 방식은, 현재 구역이 다수의 파티션들 (예컨대, 2 개의 1x2 파티션들, 또는 현재 구역의 사이즈에 기초하여 결정된 파티션 사이즈들을 갖는 파티션들의 다른 조합들) 로 파티셔닝되는 파티셔닝 방식

일 수도 있다. 대안적으로, 제 1 파티셔닝 방식은 현재 구역이 전체로서 (예컨대, 2x2 파티션으로서) 사용되고 다수의 파티션들로 파티셔닝되지 않는 파티셔닝 방식일 수도 있다. 일부 실시형태들에서, 하나 이상의 제 1 후보 구역들은 제 1 파티셔닝 방식과 연관된 위치들의 제 1 범위 (예컨대, 제 1 파티셔닝 방식과 연관된 검색 범위) 내에 있다. 하나 이상의 제 1 후보 구역들은 비디오 인코딩 디바이스의 메모리에 저장될 수도 있다.

[0134] 블록 (1410) 에서, 코더는 제 2 파티셔닝 방식을 이용하여 현재 구역을 예측하기 위해 사용될 하나 이상의 제 2 후보 구역들을 결정한다. 예를 들어, 제 2 파티셔닝 방식은 현재 구역이 다수의 파티션들로 파티셔닝되지 않는 파티셔닝 방식일 수도 있다 (예컨대, 현재 구역은 단일의 2x2 파티션으로서 코딩된다). 다른 예에서, 제 2 파티셔닝 방식은 현재 구역이 제 1 파티셔닝 방식에 대해 사용되는 파티션들의 수와는 상이한 수의 파티션들로 파티셔닝되는 파티셔닝 방식일 수도 있다. 또 다른 예에서, 제 2 파티셔닝 방식은, 현재 구역이 다수의 파티션들 (예컨대, 2 개의 1x2 파티션들, 또는 현재 구역의 사이즈에 기초하여 결정된 파티션 사이즈들을 갖는 파티션들의 다른 조합들) 로 파티셔닝되는 파티셔닝 방식일 수도 있다. 일부 실시형태들에서, 하나 이상의 제 2 후보 구역들은 제 2 파티셔닝 방식과 연관된 위치들의 제 2 범위 (예컨대, 제 2 파티셔닝 방식과 연관된 검색 범위) 내에 있다. 일부 실시형태들에서, 제 2 범위는 하나 이상의 제 1 후보 구역들을 식별하기 위해 사용되는 제 1 범위와 동일하다. 일부 경우들에서, 하나 이상의 제 2 후보 구역들은 하나 이상의 제 1 후보 구역들과 동일할 수도 있다. 다른 경우들에서, 하나 이상의 제 2 후보 구역들은 하나 이상의 제 1 후보 구역들을 포함한다. 대안적으로, 하나 이상의 제 1 후보 구역들은 하나 이상의 제 2 후보 구역들을 포함할 수도 있다. 일부 경우들에서, 하나 이상의 제 2 후보 구역들은 하나 이상의 제 1 후보 구역들과 중첩하지 않는다. 하나 이상의 제 2 후보 구역들의 사이즈는 하나 이상의 제 1 후보 구역들의 사이즈와는 상이할 수도 있다. 다른 실시형태들에서, 제 2 범위는 하나 이상의 제 1 후보 구역들을 식별하기 위해 사용되는 제 1 범위와는 상이하다. 하나 이상의 제 2 후보 구역들은 비디오 인코딩 디바이스의 메모리에 저장될 수도 있다.

[0135] 블록 (1415) 에서, 코더는 제 1 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 1 코스트가 제 2 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 큰지 여부를 결정한다. 예를 들어, 코더는 제 1 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 레이트 및 왜곡에 기초하여 제 1 코스트를 계산하고, 제 2 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 레이트 및 왜곡에 기초하여 제 2 코스트를 계산하고, 계산된 제 1 및 제 2 코스트들을 비교할 수도 있다. 하나의 예에서, 제 1 코스트는 (제 1 왜곡 값 + (람다 파라미터 * 제 1 레이트 값)) 으로서 결정될 수도 있고, 여기서, 제 1 왜곡 값은 하나 이상의 제 1 후보 구역들에 대해 YCoCg 컬러 공간에서 현재 구역 (또는 그것의 파티션) 에서의 개별 픽셀들의 수정된 SAD 에 기초하여 계산될 수도 있고, 제 2 코스트는 (제 2 왜곡 값 + (람다 파라미터 * 제 2 레이트 값)) 으로서 결정될 수도 있고, 여기서, 제 2 왜곡 값은 하나 이상의 제 2 후보 구역들에 대해 YCoCg 컬러 공간에서 현재 구역 (또는 그것의 파티션) 에서의 개별 픽셀들의 수정된 SAD 에 기초하여 계산될 수도 있다. 일부 실시형태들에서, 코더는 적어도 부분적으로, (i) 현재 구역과 하나 이상의 제 1 후보 구역들 사이의 절대 차이들의 합 및 (ii) 하나 이상의 예측 벡터들 및 대응하는 잔차들을 비트스트림에서 시그널링하기 위해 필요한 비트들의 수에 기초하여 제 1 코스트를 결정하고, 적어도 부분적으로, (i) 현재 구역과 하나 이상의 제 2 후보 구역들 사이의 절대 차이들의 합 및 (ii) 하나 이상의 예측 벡터들 및 대응하는 잔차들을 비트스트림에서 시그널링하기 위해 필요한 비트들의 수에 기초하여 제 2 코스트를 결정할 수도 있다.

[0136] 블록 (1420) 에서, 코더가, 제 1 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 1 코스트가 제 2 파티셔닝 방식에 기초하여 현재 구역을 코딩하는 것과 연관된 제 2 코스트보다 더 크다고 결정하는 경우에, 방법 (1400) 은 블록 (1425) 으로 진행한다. 그렇지 않은 경우에, 방법 (1400) 은 블록 (1430) 으로 진행한다.

[0137] 블록 (1425) 에서, 코더는 하나 이상의 제 2 후보 구역들에 기초하여 현재 구역을 비트스트림으로 코딩한다. 코더는, 비트스트림에서, 현재 구역에 대한 하나 이상의 제 2 후보 구역들의 위치를 나타내는 하나 이상의 예측 벡터들, 및 하나 이상의 제 2 후보 구역들과 현재 구역 사이의 차이 (예컨대, 대응하는 픽셀 값들 사이의 차이) 를 나타내는 양자화된 잔차를 비트스트림에서 시그널링할 수도 있다. 예를 들어, 코더는 하나 이상의 제 2 후보 구역들의 처음 또는 초기 픽셀의 위치를 나타내는 단일 벡터를 시그널링할 수도 있고, 여기서, 단일 벡터의 값은 이러한 처음 또는 초기 픽셀과 현재 구역이 처음 또는 초기 픽셀 사이의 거리에 기초한다. 하나 이상의 예측 벡터들이 다수의 벡터들을 포함하는 경우에, 코더는 현재 구역의 파티션들 중 하나를 예측하기 위해 사용될 각각의 후보 구역의 위치를 각각 나타내는 다수의 벡터들을 시그널링할 수도 있다.

[0138] 블록 (1430) 에서, 코더는 하나 이상의 제 1 후보 구역들에 기초하여 현재 구역을 비트스트림으로 코딩한다. 코더는, 비트스트림에서, 현재 구역에 대한 하나 이상의 제 1 후보 구역들의 위치를 나타내는 하나 이상의 예측 벡터들, 및 하나 이상의 제 1 후보 구역들과 현재 구역 사이의 차이를 나타내는 양자화된 잔차를 비트스트림에서 시그널링할 수도 있다. 예를 들어, 코더는 하나 이상의 제 2 후보 구역들의 처음 또는 초기 픽셀의 위치를 나타내는 단일 벡터를 시그널링할 수도 있고, 여기서, 단일 벡터의 값은 이러한 처음 또는 초기 픽셀과 현재 구역이 처음 또는 초기 픽셀 사이의 거리에 기초한다. 하나 이상의 예측 벡터들이 다수의 벡터들을 포함하는 경우에, 코더는 현재 구역의 파티션들 중 하나를 예측하기 위해 사용될 각각의 후보 구역의 위치를 각각 나타내는 다수의 벡터들을 시그널링할 수도 있다. 코더는 추가적으로, 비트스트림에서 파티션 표시자를 시그널링할 수도 있고, 파티션 표시자는 블록 내의 각 구역과 연관된 파티셔닝 방식을 나타내고, 블록은 현재 구역 이외의 적어도 하나의 구역을 포함한다. 예를 들어, 파티션 표시자는 현재 구역이 제 2 파티셔닝 방식과 연관되는 것을 나타낼 수도 있다. 파티션 표시자는 추가적으로, 블록에서의 현재 구역 이외의 적어도 하나의 구역이 제 2 파티셔닝 방식과는 상이한 제 1 파티셔닝 방식과 연관되는 것을 나타낼 수도 있다. 방법 (1400) 은 블록 (1435) 에서 종료된다.

[0139] 방법 (1400) 에서, 도 14 에서 도시된 블록들 중 하나 이상은 제거될 수도 있고 (예컨대, 수행되지 않을 수도 있다), 및/또는, 방법이 수행되는 순서는 전환될 수도 있다. 일부 실시형태들에서, 추가적인 블록들이 방법 (1400) 에 추가될 수도 있다. 본 개시의 실시형태들은 도 14 에서 도시된 예에 제한되거나 예에 의해 제한되지 않으며, 다른 변형들이 이 개시의 사상으로부터 벗어남이 없이 구현될 수도 있다.

[0140] **4:2:0 및 4:2:2 크로마 샘플링 포맷들의 확장**

[0141] 일부 구현들에서, (예컨대, 블록 예측 모드에서 가변 파티션 사이즈를 이용하는) 본 개시물에서 기술된 블록 파티션 기법들은 4:4:4 크로마 샘플링 포맷에 대해서만 이용될 수도 있다. 이 포맷은 통상적으로 그래픽 콘텐츠에 대해 사용된다. 예를 들어, 4:4:4 크로마 샘플링 포맷은 (예컨대, 크로마 서브-샘플링을 이용하지 않는) 동일한 샘플링 레이트를 갖는 컬러 컴포넌트들 (예컨대, 루마 컴포넌트들 및 크로마 컴포넌트들) 을 포함하는 이미지 또는 비디오 데이터를 이용한다. 하지만, 4:4:4 크로마 샘플링 포맷은 다른 비디오 애플리케이션들에 대해 통상적으로 덜 사용될 수도 있다. 크로마 서브-샘플링이 제공할 수도 있는 현저한 압축으로 인해, 양 4:2:0 및 4:2:2 크로마 서브-샘플링 포맷들은 통상적으로 비디오 애플리케이션들에 대해 사용된다. 예를 들어, DSC (예컨대, DSCv1.x) 의 일부 버전들은 4:2:0 및 4:2:2 를 지원할 수도 있다. 이러한 크로마 서브-샘플링 포맷들에 대한 지원은 장래의 DSC 구현들에 의해 이용되거나 요구될 수도 있다. 따라서, 일부 실시형태들에서, (예컨대, 블록 예측 모드에서 가변 파티션 사이즈들을 이용하는) 본 개시물에서 설명된 블록 예측 기법들은 4:2:0 및/또는 4:2:2 포맷들로 확장된다. 비록 4:2:0 및 4:2:2 크로마 서브-샘플링 포맷들이 본 명세서에서 사용되지만, 본원에서 설명된 다양한 기법들은 다른 알려진 샘플링 포맷들에 대해 적용될 수도 있다.

[0142] 일부 실시형태들에서, 가변 파티션 사이즈를 갖는 블록 예측에 대한 알고리즘은 크로마 샘플링 포맷에 독립적으로 훨씬 동일한 방식으로 작용한다. 이러한 실시형태들에서, 포맷 (예컨대, 4:4:4, 4:2:2, 4:2:0 등) 에 상관 없이, 단일 파티션 (예컨대, 2x2) 을 이용할지 또는 다수의 파티션들 (예컨대, 2 개의 분리된 1x2 파티션들) 을 이용할지 여부의 결정, 또는 현재의 서브-블록 또는 구역 (예컨대, 1, 2, 3, 4 등) 을 코딩하기 위해 사용될 파티션들의 수의 결정이 루마 샘플들의 각각의 서브-블록 또는 구역 (예컨대, 2x2 블록) 에 대해 이루어질 수도 있다. 하지만, 각 파티션에서 또는 각 블록에서 크로마 샘플들의 수는 서브-샘플링 포맷에 의존하여 상이할 수도 있다. 또한, 인코더 결정은, 엔트로피 코딩 그룹들과의 정렬이 크로마 컴포넌트들에 대해 더 이상 가능하지 않을 수도 있기 때문에, 4:2:2 및/또는 4:2:0 크로마 서브-샘플링 포맷들에서 수정될 필요성이 있을 수도 있다. 따라서, (예컨대, 인코더가 최소 RD 코스트에 기초하여 각각의 2x2 구역을 단일의 2x2 파티션 또는 2 개의 분리된 1x2 파티션들로 분할할지 여부를 결정할 때) 인코더 결정에 대한 각각의 파티션에 대한 레이트 (예컨대, 단일의 2x2 파티션 또는 2 개의 분리된 1x2 파티션들과 같은, 파티션들과 연관된 레이트 값) 는 4:2:2 및 4:2:0 에 대한 루마 샘플들에만 의존할 수도 있다. 예를 들어, SAD 왜곡을 계산할 때, 크로마 컴포넌트(들)에 관련된 임의의 항들은 제로로 설정될 수도 있다.

[0143] **4:2:0 크로마 서브샘플링 포맷에 대한 BP 검색**

[0144] 4:2:0 모드 (4:2:0 크로마 서브-샘플링 포맷) 에서의 2x2 파티션들에 대해, 각 파티션은 크로마 컴포넌트들 (예컨대, Co 및 Cg, 또는 Cb 및 Cr) 의 각각에 대해 단일의 크로마 샘플을 포함할 수도 있다. 일부 실시형태들에서, (예컨대, 현재 구역 또는 블록에서 샘플들을 예측하기 위해 및/또는 RD 코스트를 계산하기 위해) 사용될

크로마 샘플은 파티션과 교차하는 것이다. 다른 실시형태들에서, 사용될 크로마 샘플은 인접 파티션으로부터 도출될 수도 있다. 4:2:0 모드에 대한 예시적인 2x2 검색 (1500) 이 도 15 에서 도시된다. 도 15 에서, 크로마 사이트들 (예컨대, 크로마 샘플들을 갖는 샘플/픽셀 위치들) 은 "X" 를 이용하여 표시된다. 예를 들어, 파티션 A 의 상부 좌측 샘플, 파티션 B 의 상부 우측 샘플, 현재 파티션의 상부 좌측 샘플은 각각의 파티션을 교차하는 크로마 사이트들을 포함한다. 이러한 크로마 사이트들은 (예컨대, 크로마 샘플 값들을 이용하여 차이 값을 계산하기 위해) 각각의 파티션들에 대해 수행되는 모든 계산들에 대해 사용될 수도 있다.

[0145] 4:2:0 모드에서의 1x2 파티션들에 대해, 현재 블록의 제 2 라인에서 크로마 사이트들이 존재하지 않을 수도 있기 때문에, 현재 블록의 제 1 라인에서의 1x2 파티션들 및 현재 블록의 제 2 라인에서의 1x2 파티션들 사이에 구분이 이루어질 필요성이 있을 수도 있다. 예를 들어, 현재 블록의 제 1 라인에서의 파티션들에 대해, 왜곡 값들의 계산은 각각의 크로마 컴포넌트에 대해 하나의 크로마 샘플 및 2 개의 루마 샘플들을 수반할 수도 있다. 현재 블록의 제 2 라인에서의 파티션들에 대해, 왜곡 값들의 계산은 오직 루마 샘플들 (예컨대, 2 개의 루마 샘플들) 만을 수반할 수도 있다. 도 16 의 예 (1600) 에서, 현재의 1x2 파티션 A 는 제 1 라인에 있고, 크로마 사이트를 포함한다. 따라서, 현재의 1x2 파티션 A 를 예측하기 위해 선택된 후보 파티션은 크로마 사이트를 또한 포함하는 후보 1x2 파티션 A 이다. 유사하게, 현재의 1x2 파티션 B 는 제 2 라인에 있고, 크로마 사이트를 포함하지 않는다. 따라서, 현재의 1x2 파티션 B 를 예측하기 위해 선택된 후보 파티션은 크로마 사이트를 또한 포함하지 않는 후보 1x2 파티션 B 이다.

[0146] **4:2:2 크로마 서브샘플링 포맷을 위한 BP 검색**

[0147] 4:2:2 모드 (4:2:2 크로마 서브-샘플링 포맷) 에서의 2x2 파티션들에 대해, 각 파티션은 크로마 컴포넌트들 (예컨대, Co 및 Cg, 또는 Cb 및 Cr) 의 각각에 대해 2 개의 크로마 샘플들, 및 2 개의 루마 샘플들을 포함할 수도 있다. 4:2:2 모드에 대한 예시적인 2x2 검색 (1700) 이 도 17 에서 도시된다. 도 17 에서, 크로마 사이트들 (예컨대, 크로마 샘플들을 갖는 픽셀 위치들) 은 "X" 를 이용하여 표시된다. 예를 들어, 파티션 A 의 2 개의 좌측 샘플들, 파티션 B 의 2 개의 우측 샘플들, 및 현재 파티션의 2 개의 좌측 샘플들은 각각의 파티션들을 교차하는 크로마 사이트들을 포함한다. 이러한 크로마 사이트들은 (예컨대, 크로마 샘플 값들을 이용하여 차이 값을 계산하기 위해) 각각의 파티션들에 대해 수행되는 모든 계산들에 대해 사용될 수도 있다.

[0148] 4:2:2 모드에서의 1x2 파티션들에 대해, 각 파티션은 크로마 컴포넌트들 (예컨대, Co 및 Cg, 또는 Cb 및 Cr) 의 각각에 대해 1 개의 크로마 샘플, 및 2 개의 루마 샘플들을 포함한다. 4:2:0 모드와는 달리, 4:2:2 모드에서 현재 블록의 제 1 라인에서의 파티션들과 현재 블록의 제 2 라인에서의 파티션들 사이에 구분이 존재하지 않을 수도 있다. 4:2:2 크로마 서브-샘플링을 위한 1x2 파티션들에 대한 예시적인 블록 예측 검색 (1800) 이 도 18 에서 예시된다. 도 18 의 예에서, 현재의 1x2 파티션 A 는 제 1 라인에 있고, 현재의 1x2 파티션 B 는 제 2 라인에 있으며, 현재 파티션들 A 및 B 의 각각은 크로마 사이트를 포함한다. 현재의 파티션 A 는 제 1 샘플에서의 크로마 사이트를 포함하는 후보 1x2 파티션 A 에 기초하여 예측되고, 현재의 파티션 B 는 제 2 샘플에서의 크로마 사이트를 포함하는 후보 1x2 파티션 B 에 기초하여 예측된다. 따라서, 크로마 사이트가 후보 파티션 내에 어디에 위치되는지에 상관 없이, 크로마 샘플은 현재 파티션에서 크로마 샘플을 예측하기 위해 사용될 수도 있다.

[0149] **인코더 결정**

[0150] 4:2:2 및 4:2:0 포맷들에서, 각각의 크로마 컴포넌트에 대해 블록 당 4 개보다 더 적은 엔트로피 코딩 그룹들이 존재할 수도 있다. 예를 들어, 4 개의 엔트로피 코딩 그룹들이 루마 컴포넌트에 대해 사용될 수도 있고, 2 개 (또는 하나) 의 엔트로피 코딩 그룹들이 오렌지 크로마 컴포넌트에 대해 사용될 수도 있으며, 2 개 (또는 하나) 의 엔트로피 코딩 그룹들이 그린 크로마 컴포넌트에 대해 사용될 수도 있다. 주어진 블록을 코딩하기 위해 사용되는 엔트로피 코딩 그룹들의 수는 주어진 블록에서의 루마 또는 크로마 샘플들의 수에 기초하여 결정될 수도 있다. 일부 실시형태들에서, 엔트로피 코딩 그룹들은 주어진 블록이 코딩되는 코딩 모드에 기초하여 인코더에 의해 결정된다. 다른 실시형태들에서, 엔트로피 코딩 그룹들은 (예컨대, 주어진 블록이 코딩되는 코딩 모드에 기초하여) 적용가능한 코딩 표준에 의해 설정된다.

[0151] 일부 실시형태들에서, 양 EC_{bits} 은 크로마에 대해 인코더에 의해 정확하게 결정되지 않는다. 이러한 실시형태들의 일부에서, 인코더는 오직 4:2:2 및 4:2:0 포맷들에 대해 루마 샘플들만을 이용하여 계산되는 엔트로피 코딩 레이트에 기초하여, 1x2 또는 2x2 파티션들을 사용할지 여부를 결정할 수도 있다. 다른 실시형태들에서, 양 EC_{bits} 은 크로마에 대해 인코더에 의해 결정되고, 인코더는 4:2:2 및 4:2:0 포맷들에 대해 루마 및 크로

마 샘플들 양자를 이용하여 계산되는 엔트로피 코딩 레이트에 기초하여, 1x2 또는 2x2 파티션들을 이용할지 여부를 결정할 수도 있다.

[0152] **시그널링**

[0153] 일부 실시형태들에서, 각각의 블록에 대해 또는 각각의 컬러 컴포넌트에 대해 인코더로부터 디코더로 송신될 엔트로피 코딩 그룹들의 수는 크로마 서브-샘플링 포맷에 의존하여 변화될 수도 있다. 일부 구현들에서, 엔트로피 코딩 그룹들의 수는 코텍 스트루프티 충분히 높은 것을 보장하기 위해 변경될 수도 있다. 예를 들어, 4:4:4 모드에서, 2x8 블록은 도 11 에서 예시된 바와 같이 4 개의 엔트로피 코딩 그룹들을 포함할 수도 있다. 이러한 예에서, 4 개의 엔트로피 코딩 그룹들이 각각의 컬러 컴포넌트 (예컨대, Y, Co, 및 Cg) 에 대해 사용될 (예컨대, 인코더에 의해 시그널링될) 수도 있다. 표 1 은 4:2:2 및 4:2:0 모드들에 대해 사용되는 엔트로피 코딩 그룹들의 수에 대한 예시적인 변경들을 기술한다. 상기 설명된 시그널링의 나머지 (예컨대, BPV 들의 시그널링, 파티셔닝 방식의 표시의 시그널링 등) 는 4:2:2 및 4:2:0 모드들에 대해 (4:4:4 모드에 대해 설명된 시그널링으로부터) 변경되지 않을 수도 있다. 예를 들어, 표 1 에서, 컴포넌트 0 은 루마 (Y) 에 대응할 수도 있고, 컴포넌트 1 은 오렌지 크로마 (Co) 에 대응할 수도 있으며, 컴포넌트 2 는 그린 크로마 (Cg) 에 대응할 수도 있다.

표 1

[0154]

크로마 포맷	컴포넌트 0	컴포넌트 1	컴포넌트 2
4:4:4	4	4	4
4:2:2	4	2	2
4:2:0	4	1	1

[0155] 표 1: (2x8 의 블록 사이즈를 가정하여) 상이한 크로마 서브-샘플링 포맷들에 대한 컴포넌트 당 엔트로피 코딩 그룹들의 수

[0156] **이점들**

[0157] 본 개시물에서 설명된 하나 이상의 블록 예측 모드 기법들은 비대칭적 설계를 이용하여 구현될 수도 있다. 비대칭적 설계는 더욱 고가의 절차들이 인코더 측 상에서 수행되도록 하여, 디코더의 복잡도를 감소시킨다. 예를 들어, 벡터 (들) 는 디코더로 명시적으로 시그널링되므로, 인코더는 디코더에 비해 작업의 대부분을 행한다. 이것은 인코더가 종종 시스템 온 칩 (System on a Chip; SoC) 설계의 일부이어서, 최첨단 프로세스 노드 (예컨대, 20 nm 이하) 상에서 높은 주파수에서 실행될 때에 바람직하다. 한편, 디코더는 제한된 클럭 속도 및 훨씬 더 큰 프로세스 크기 (예컨대, 65 nm 이상) 를 갖는 디스플레이 구동기 집적 회로 (Display Driver Integrated Circuit; DDIC) 칩-온-글래스 (chip-on-glass; COG) 해결책 상에서 구현될 가능성이 있다.

[0158] 추가적으로, 블록 파티션 사이즈들의 적응적 선택은 블록 예측 모드가 보다 넓은 범위의 콘텐츠 타입들에 대해 사용되도록 허용한다. BPV 들을 시그널링하는 것은 명확하게 비쌀 수 있으므로, 가변 파티션 사이즈는 2x2 파티션을 이용하여 잘 예측될 수 있는 이미지 구역들에 대한 감소된 시그널링 비용을 허용한다. 고도로 복잡한 구역들에 대해, 엔트로피 코딩 레이트가 더 높은 시그널링 코스트에 대해 구성하기 위해 충분히 감소될 수 있는 경우, 또는 RD 트레이트오프가 여전히 1x2 를 선호하도록 왜곡이 충분히 감소될 수 있는 경우 중 어느 일방의 경우에, 1x2 파티션 사이즈가 선택될 수 있다. 예를 들어, 블록 파티션 사이즈들의 적응적 선택은 자연적 이미지들, 테스트 패턴들, 정세한 텍스트 렌더링 등을 포함하는, 모든 콘텐츠 타입들에 걸쳐 퍼포먼스를 증가시킬 수도 있다. 일부 실시형태들에서, 본 명세서에서 논의된 적응적 파티셔닝 기법들은 2x2 보다 더 큰 블록 파티션 사이즈들 및/또는 2x8 보다 더 큰 블록 사이즈들을 고려함으로써 확장될 수도 있다.

[0159] 본 명세서에서 기술된 하나 이상의 기법들은 고정 비트 레이트 버퍼 모델을 채용하는 고정-비트 코덱에서 구현될 수도 있다. 이러한 모델에서, 레이트 버퍼에서 저장된 비트들은 고정 비트 레이트에서 레이트 버퍼로부터 제거된다. 따라서, 비디오 인코더가 비트스트림에 너무 많은 비트들을 추가하는 경우에, 레이트 버퍼는 오버플로우할 수도 있다. 다른 한편, 비디오 인코더는 레이트 버퍼의 언더플로우를 방지하기 위해 충분한 비트들을 추가할 필요성이 있을 수도 있다. 또한, 비디오 디코더 측에서, 비트들은 고정 비트 레이트에서 레이트 버퍼에 추가될 수도 있고, 비디오 디코더는 각 블록에 대해 가변 수들의 비트들을 제거할 수도 있다. 적절한 디코딩을 보장하기 위해, 비디오 디코더의 레이트 버퍼는 압축된 비트스트림의 디코딩 동안 "언더플로우 (underflow)" 또는 "오버플로우 (overflow)" 하여서는 아니된다. 본원에 기술된 하나 이상의 기법들은

인코딩 및/또는 디코딩 동안 이러한 언더플로우 또는 오버플로우가 방지되는 것을 보장할 수도 있다. 일부 실시형태들에서, 인코더는 비트-버짓 제약 (bit-budget constraint) 하에서 동작할 수도 있고, 여기서, 인코더는 주어진 구역, 슬라이스, 또는 프레임을 코딩하기 위해 고정된 수의 비트들을 갖는다. 이러한 실시형태들에서, 주어진 구역, 슬라이스, 또는 프레임을 코딩할 수 있기 위해 복수의 코딩 모드들 중의 각각의 코딩 모드들이 얼마나 많은 비트들을 필요로 할 지를 정확하게 (그리고 추정할 필요 없이) 알아서, 인코더가 비트-버짓 또는 다른 비트-대역폭 관련 제약들이 만족될 수 있는 것을 보장할 수 있도록 하는 것은 인코더에게 매우 중요하다. 예를 들어, 인코더는, 주어진 구역, 슬라이스, 또는 프레임의 코딩이 추정된 것보다 더 많은 비트들을 필요로 하는 경우에, 임의의 예방적 조치들을 구현할 필요 없이 주어진 코딩 모드에서 주어진 구역, 슬라이스, 또는 프레임을 코딩할 수도 있다.

[0160] 추가적으로, 본원에서 기술된 하나 이상의 기법들은 디스플레이 링크들을 통한 송신에서 비디오 압축 기술과 연관된 특정 기술적인 문제점들을 극복한다. 구역이 다수의 후보 구역들 (예컨대, 다수의 후보 구역들 중 대응하는 후보 구역에 기초하여 예측된 구역에서의 각각의 파티션) 에 기초하여 코딩되도록 허용함으로써, 비디오 인코더들 및 디코더들은 구역의 성질 (예컨대, 평활, 복잡 등) 에 기초하여 커스터마이징된 예측을 제공할 수 있고, 이에 의해, 비디오 인코더 및 디코더 (예컨대, 하드웨어 및 소프트웨어 코덱들) 퍼포먼스를 향상시킬 수 있다.

[0161] **다른 고려사항들**

[0162] 본원에서 개시된 정보 및 신호들은 다양한 상이한 기술들 및 기법들 중의 임의의 것을 이용하여 표현될 수도 있다. 예를 들어, 상기 설명의 전반에 걸쳐 참조될 수도 있는 데이터, 명령들, 커맨드 (command) 들, 정보, 신호들, 비트들, 심볼들, 및 칩들은 전압들, 전류들, 전자기파들, 자기장들 또는 입자들, 광학 필드들 또는 입자들, 또는 그 임의의 조합에 의해 표현될 수도 있다.

[0163] 본원에서 개시된 실시형태들과 관련하여 설명된 다양한 예시적인 논리적 블록들 및 알고리즘 단계들은 전자 하드웨어, 컴퓨터 소프트웨어, 또는 양자의 조합들로서 구현될 수도 있다. 하드웨어 및 소프트웨어의 이 교환 가능성을 명확하게 예시하기 위하여, 다양한 예시적인 컴포넌트들, 블록들, 및 단계들은 일반적으로 그 기능성의 측면에서 위에서 설명되었다. 이러한 기능이 하드웨어 또는 소프트웨어로서 구현되는지 여부는 특정한 애플리케이션과, 전체 시스템에 부과된 설계 제약들에 종속된다. 당업자들은 각각의 특별한 애플리케이션을 위한 다양한 방법들로 설명된 기능성을 구현할 수도 있지만, 이러한 구현 판단들은 본 개시물의 범위로부터의 이탈을 야기시키는 것으로 해석되지 않아야 한다.

[0164] 본원에서 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 그 임의의 조합으로 구현될 수도 있다. 이러한 기법들은 범용 컴퓨터들, 무선 통신 디바이스 핸드셋들, 또는 무선 통신 디바이스 핸드셋들 및 다른 디바이스들에서의 애플리케이션을 포함하는 다수의 용도들을 가지는 집적 회로 디바이스들과 같은 다양한 디바이스들 중의 임의의 것에서 구현될 수도 있다. 디바이스들 또는 컴포넌트들로서 설명된 임의의 특징들은 집적된 로직 디바이스에서 함께, 또는 개별적이지만 상호 동작가능한 로직 디바이스들로서 별도로 구현될 수도 있다. 소프트웨어로 구현될 경우, 기법들은, 실행될 때, 위에서 설명된 방법들 중의 하나 이상을 수행하는 명령들을 포함하는 프로그램 코드를 포함하는 컴퓨터-판독가능 데이터 저장 매체에 의해 적어도 부분적으로 실현될 수도 있다. 컴퓨터-판독가능 데이터 저장 매체는 패키징 재료들을 포함할 수도 있는 컴퓨터 프로그램 제품의 일부를 형성할 수도 있다. 컴퓨터-판독가능 매체는 동기식 랜덤 액세스 메모리 (synchronous dynamic random access memory; SDRAM) 와 같은 랜덤 액세스 메모리 (random access memory; RAM), 판독-전용 메모리 (read-only memory; ROM), 비-휘발성 랜덤 액세스 메모리 (non-volatile random access memory; NVRAM), 전기적 소거가능 프로그래밍가능 판독-전용 메모리 (electrically erasable programmable read-only memory; EEPROM), 플래시 메모리 (FLASH memory), 자기 또는 광학 데이터 저장 매체들 등과 같은 메모리 또는 데이터 저장 매체들을 포함할 수도 있다. 추가적으로 또는 대안적으로, 기법들은 전파된 신호들 또는 파 (wave) 들과 같이, 명령들 또는 데이터 구조들의 형태로 프로그램 코드를 반송하거나 통신하며 컴퓨터에 의해 액세스, 판독, 및/또는 실행될 수 있는 컴퓨터-판독가능 통신 매체에 의해 적어도 부분적으로 실현될 수도 있다.

[0165] 프로그램 코드는, 하나 이상의 디지털 신호 프로세서 (digital signal processor; DSP) 들, 범용 마이크로프로세서들, 애플리케이션 특정 집적 회로 (application specific integrated circuit; ASIC) 들, 필드 프로그래밍 가능한 로직 어레이 (field programmable logic array; FPGA) 들, 또는 다른 등가의 집적 또는 개별 로직 회로부와 같은 하나 이상의 프로세서들을 포함할 수도 있는 프로세서에 의해 실행될 수도 있다. 이러한 프로세서는 이 개시물에서 설명된 기법들 중의 임의의 것을 수행하도록 구성될 수도 있다. 범용 프로세서는 마이

크로프로세서일 수도 있지만, 대안적으로, 프로세서는 임의의 기존의 프로세서, 제어기, 마이크로제어기, 또는 상태 머신일 수도 있다. 프로세서는 또한, 컴퓨팅 디바이스들의 조합, 예컨대, DSP 및 마이크로프로세서, 복수의 마이크로프로세서들, DSP 코어와 함께 하나 이상의 마이크로프로세서들, 또는 임의의 다른 이러한 구성의 조합으로서 구현될 수도 있다. 따라서, 본원에서 이용된 바와 같은 용어 "프로세서" 는 상기한 구조, 상기 구조의 임의의 조합, 또는 본원에서 설명된 기법들의 구현을 위해 적당한 임의의 다른 구조 또는 장치 중의 임의의 것을 지칭할 수도 있다. 게다가, 일부의 양태들에서, 본원에서 설명된 기능성은, 인코딩 및 디코딩을 위해 구성되거나, 조합된 비디오 인코더-디코더 (combined video encoder-decoder; CODEC) 내에 편입된 전용 소프트웨어 또는 하드웨어 내에서 제공될 수도 있다. 또한, 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수 있다.

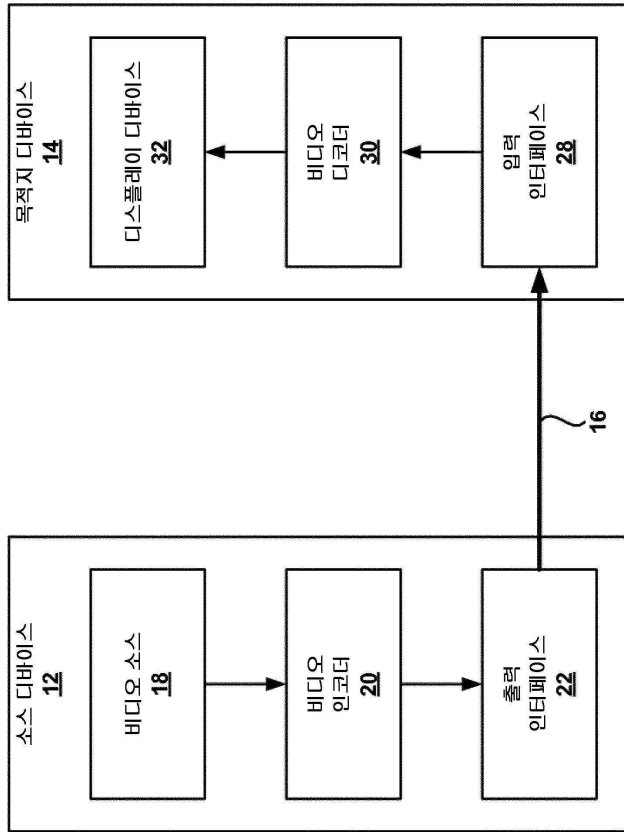
[0166] 이 개시물의 기법들은 무선 핸드셋 (wireless handset), 집적 회로 (integrated circuit; IC) 또는 IC 들의 세트 (예를 들어, 칩세트) 를 포함하는 광범위한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들 또는 유닛들은 개시된 기법들을 수행하도록 구성된 디바이스들의 기능적 양태들을 강조하기 위하여 이 개시물에서 설명되어 있지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 요구하지는 않는다. 오히려, 위에서 설명된 바와 같이, 다양한 유닛들은 코텍 하드웨어 유닛 내에 조합될 수도 있거나, 적당한 소프트웨어 및/또는 펌웨어와 함께, 위에서 설명된 바와 같은 하나 이상의 프로세서들을 포함하는 상호동작하는 하드웨어 유닛들의 집합에 의해 제공될 수도 있다.

[0167] 상기한 것은 다양한 상이한 실시형태들과 관련하여 설명되었지만, 하나의 실시형태로부터의 특징들 또는 구성요소들은 이 개시물의 교시사항들로부터 이탈하지 않으면서 다른 실시형태들과 조합될 수도 있다. 그러나, 개개의 실시형태들 사이의 특징들의 조합들은 반드시 그것으로 제한되지는 않는다. 개시물의 다양한 실시형태들이 설명되었다. 이러한, 그리고 다른 양태들은 다음의 청구항들의 범위 내에 있다.

도면

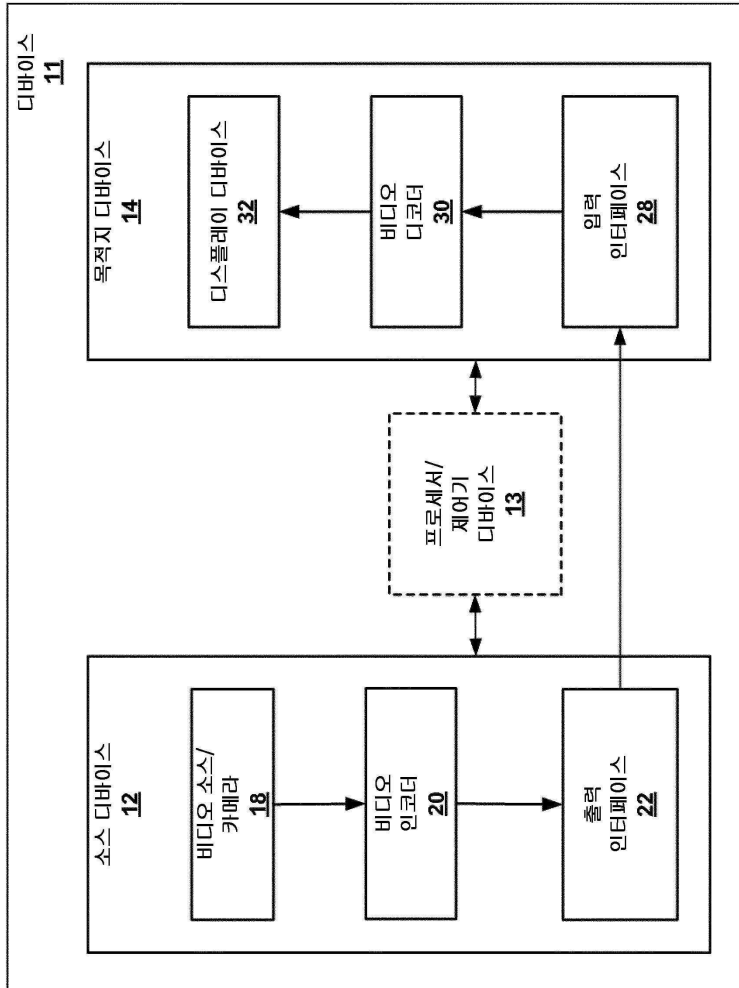
도면1a

10

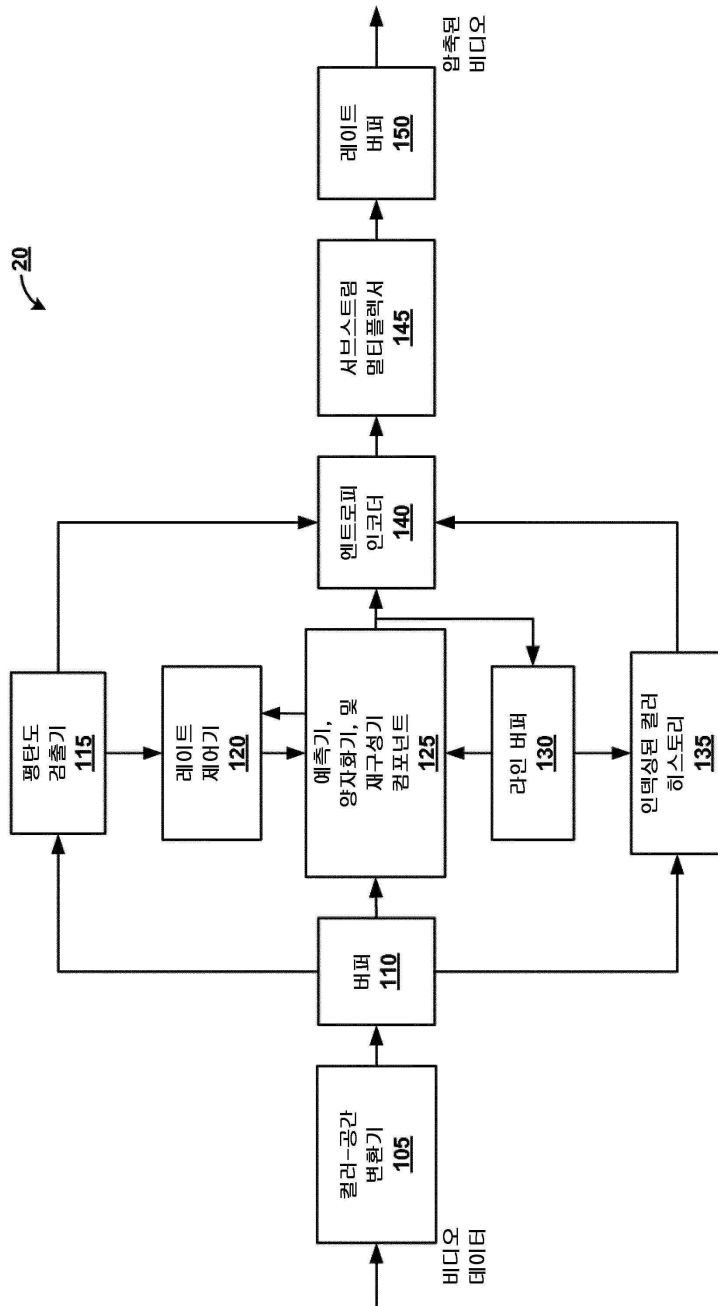


도면1b

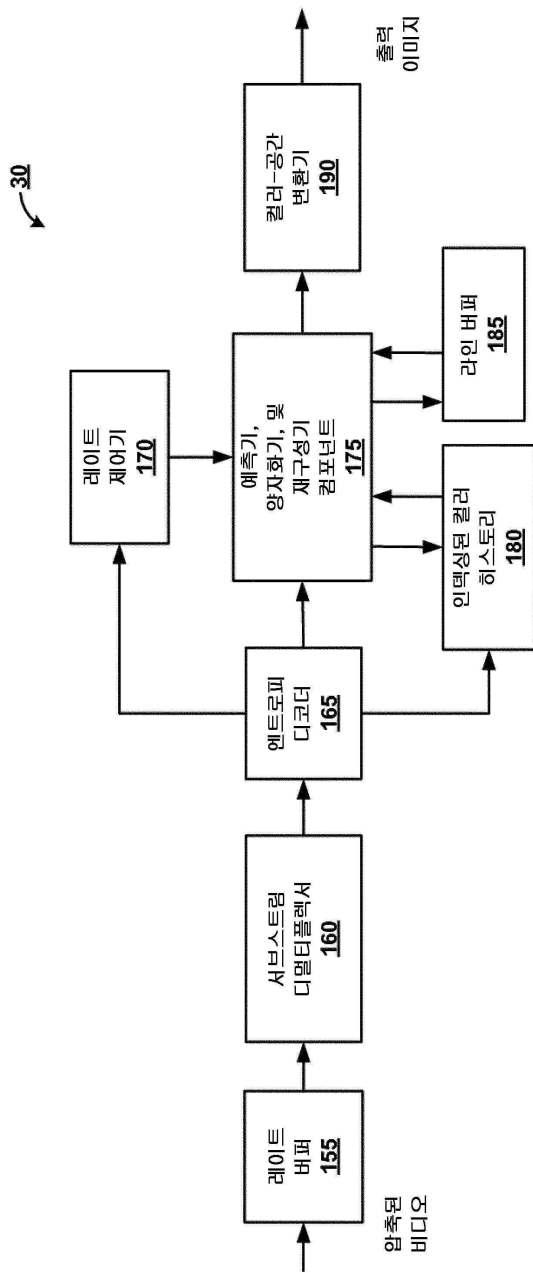
10'



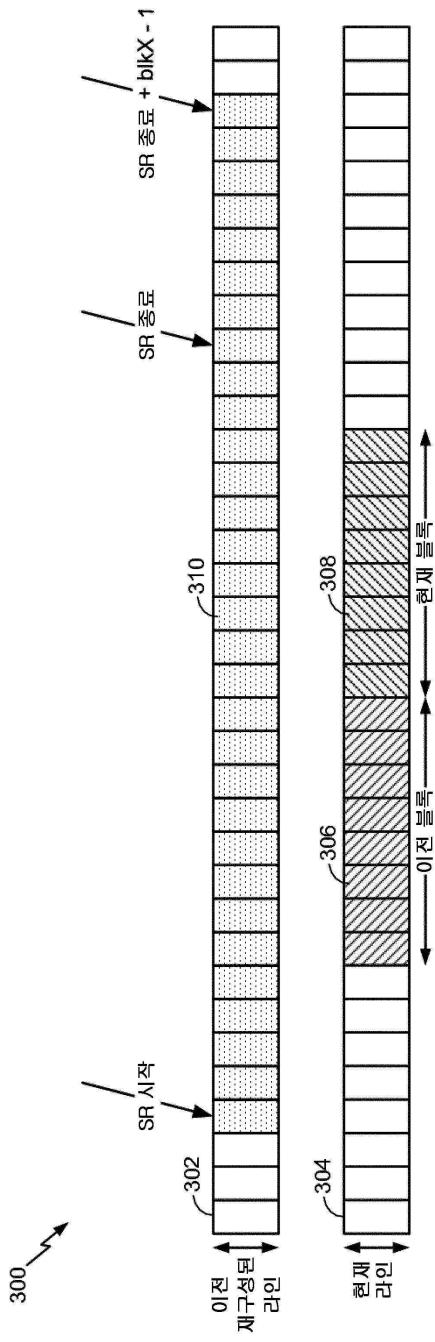
도면2a



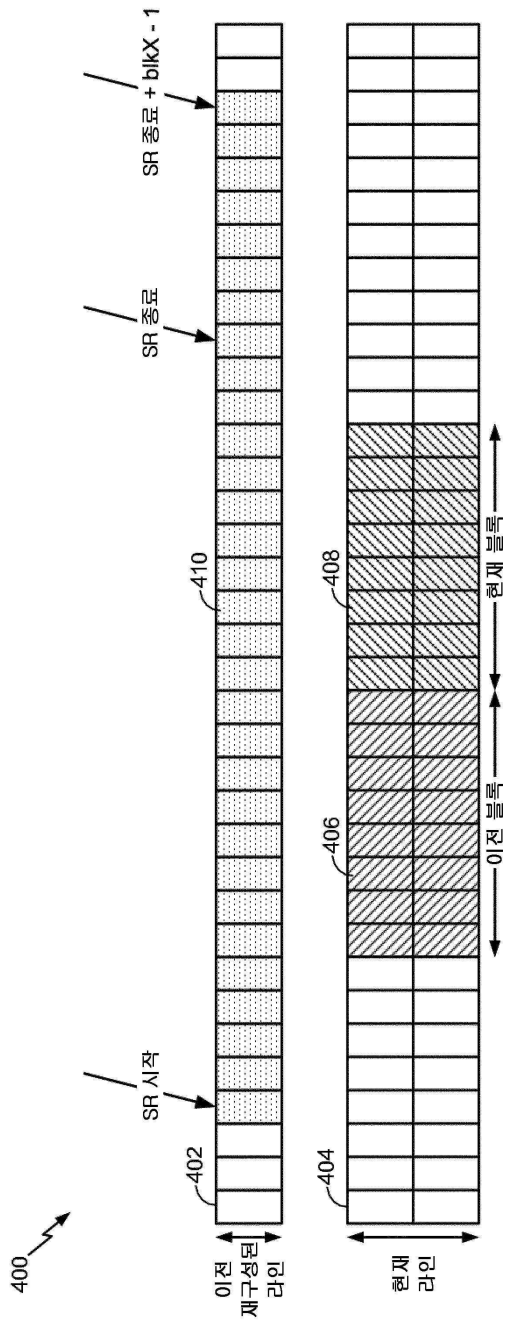
도면2b



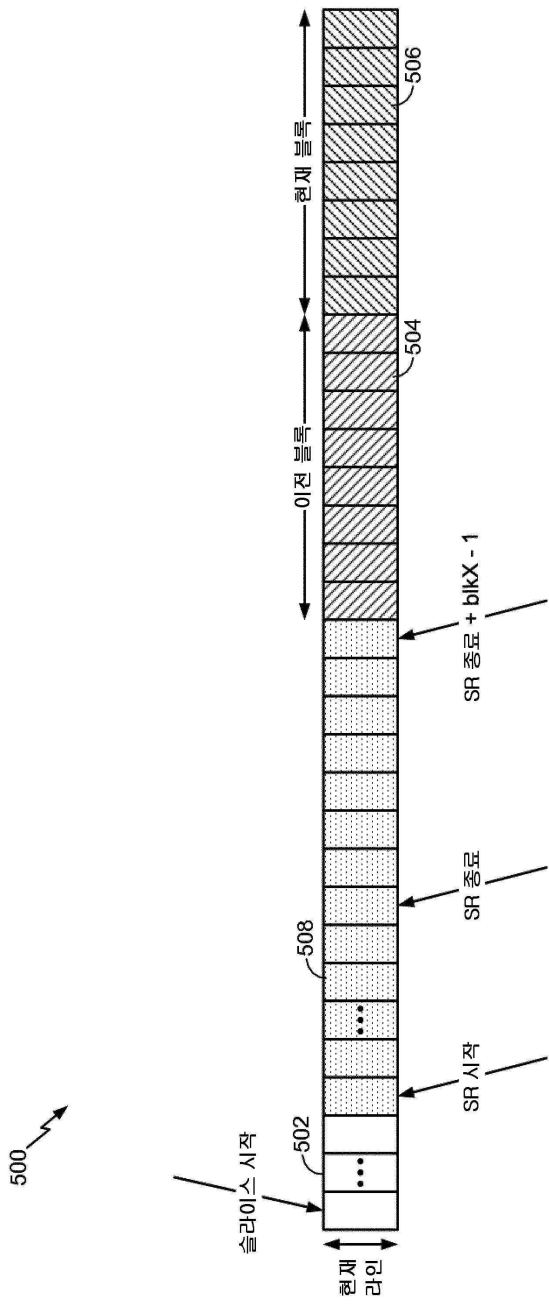
도면3



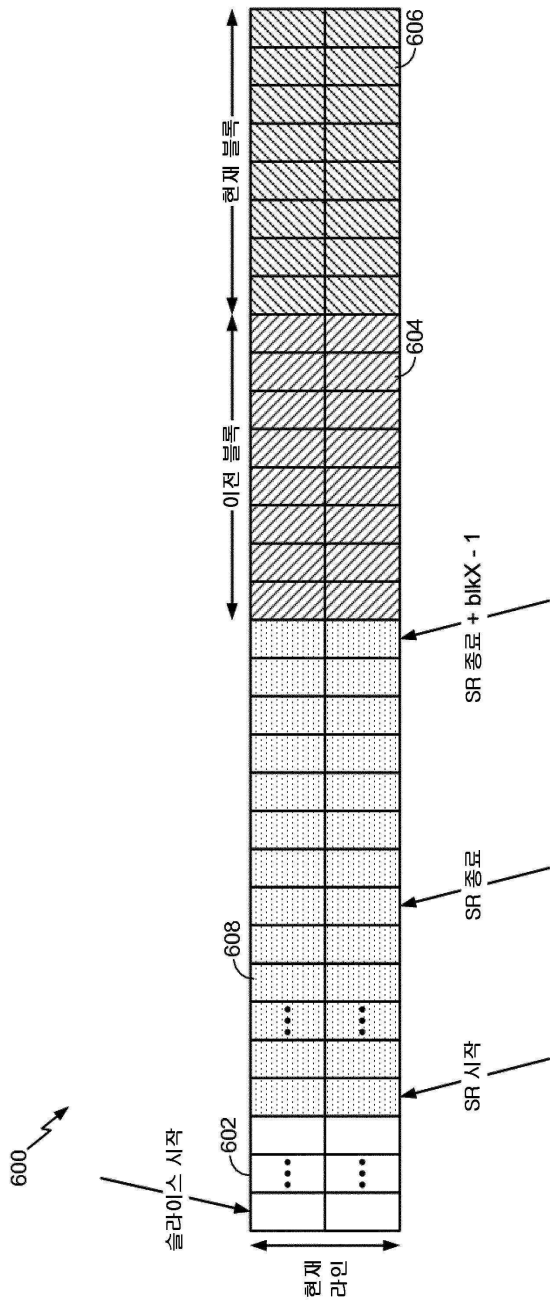
도면4



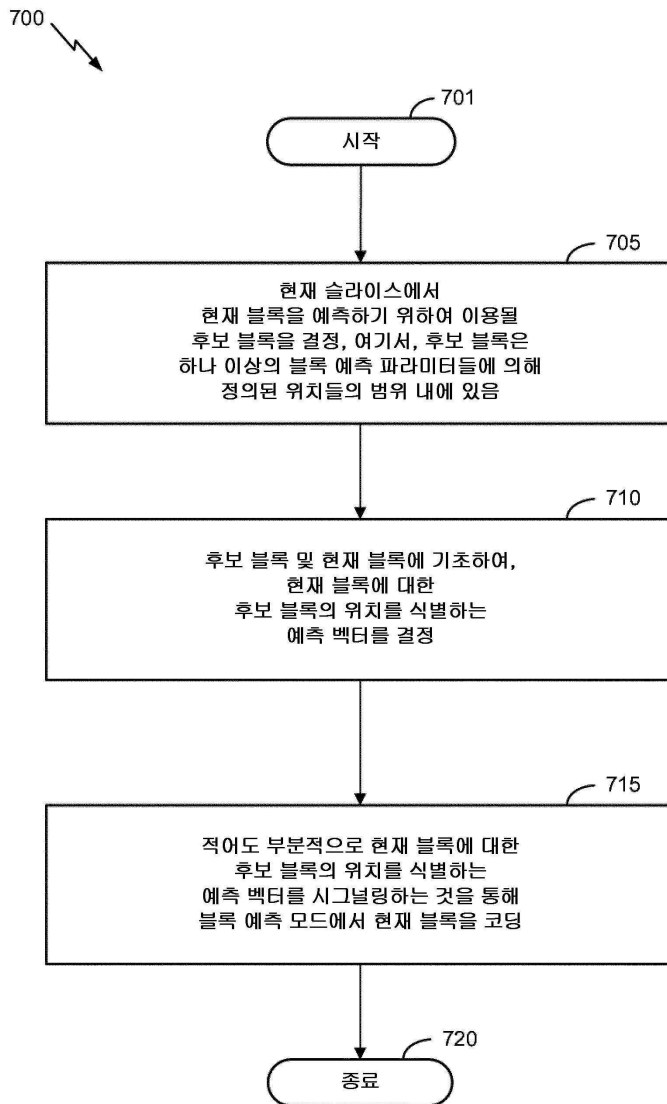
도면5



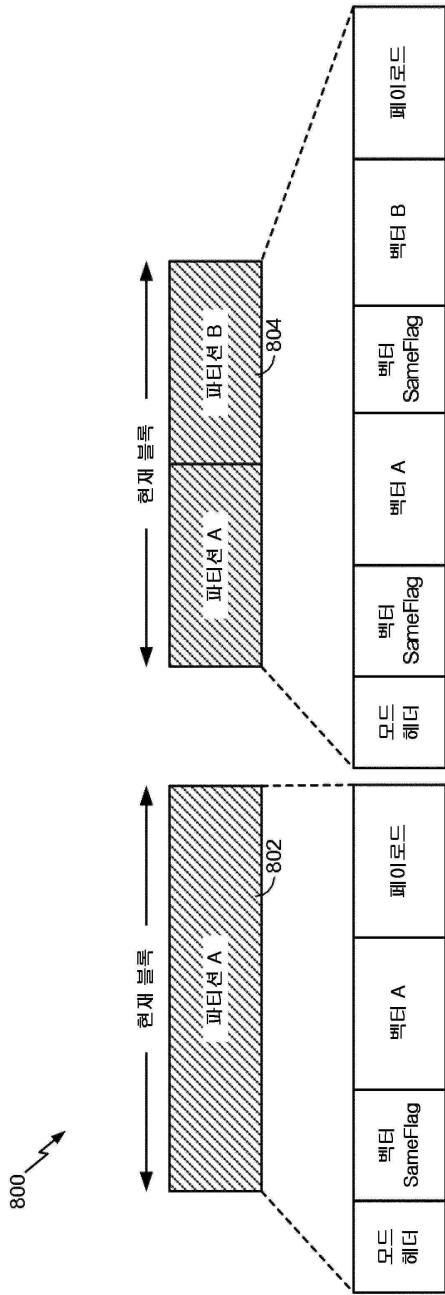
도면6



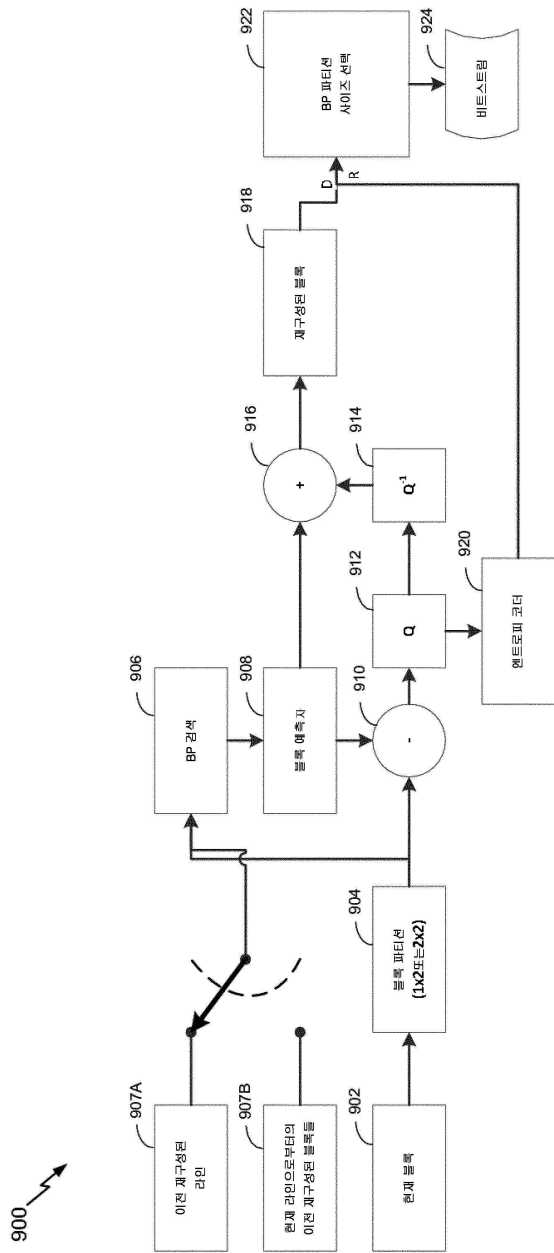
도면7



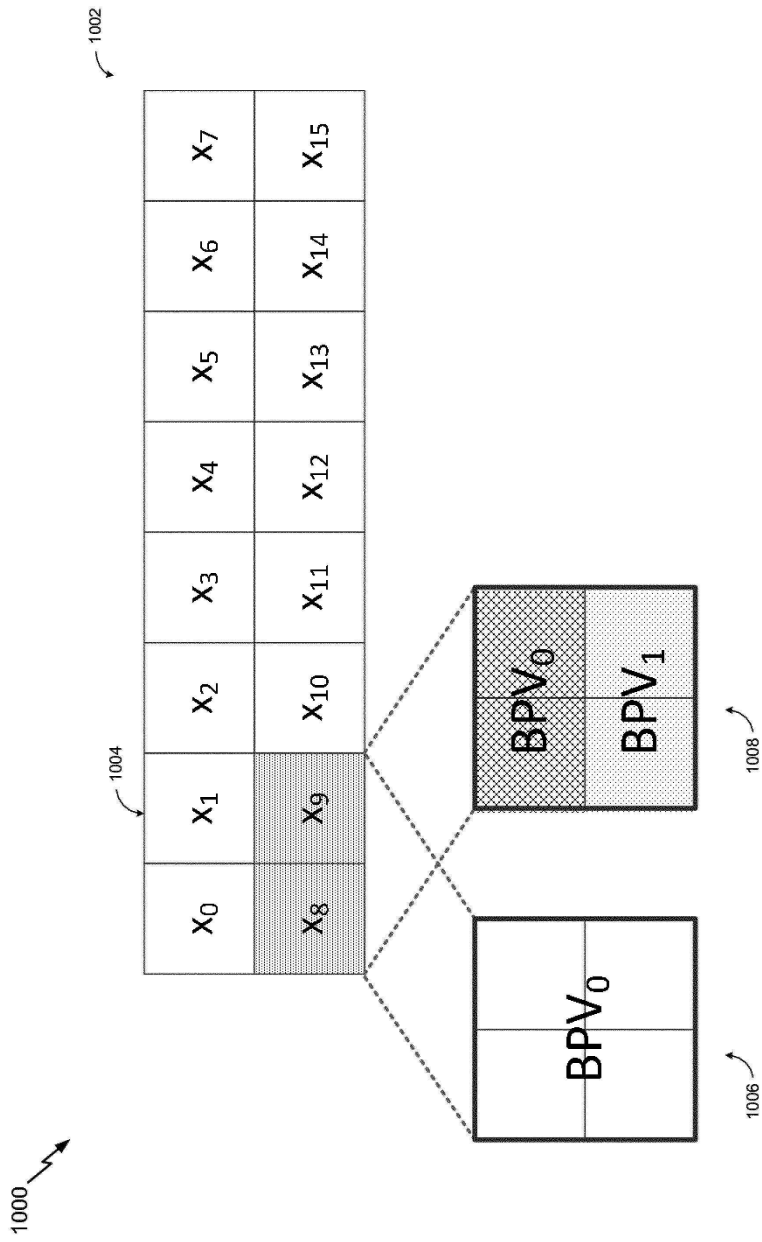
도면8



도면9

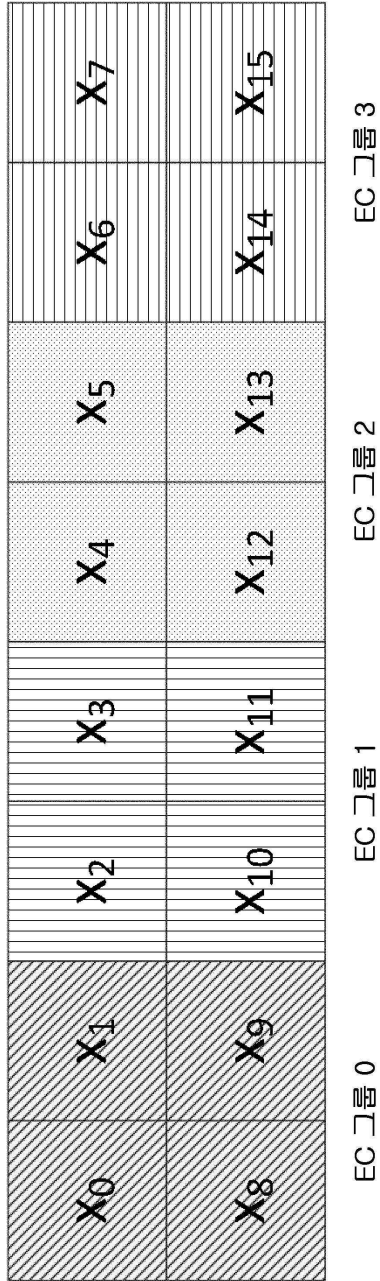


도면10

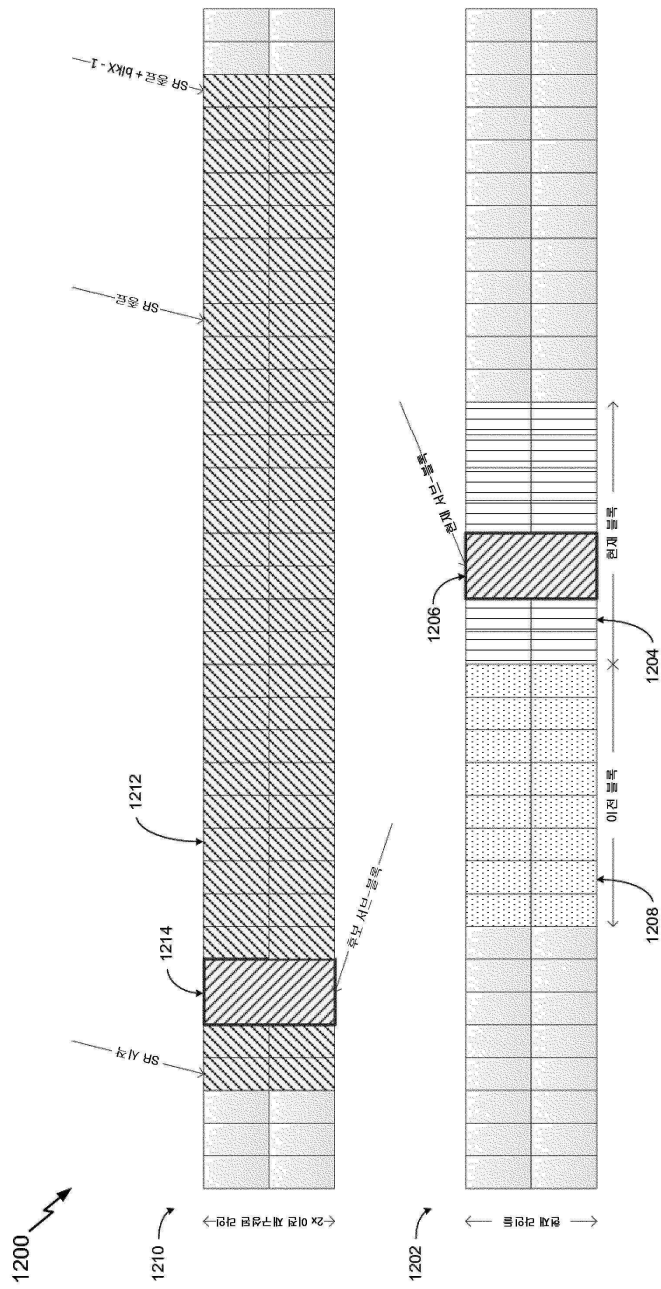


도면11

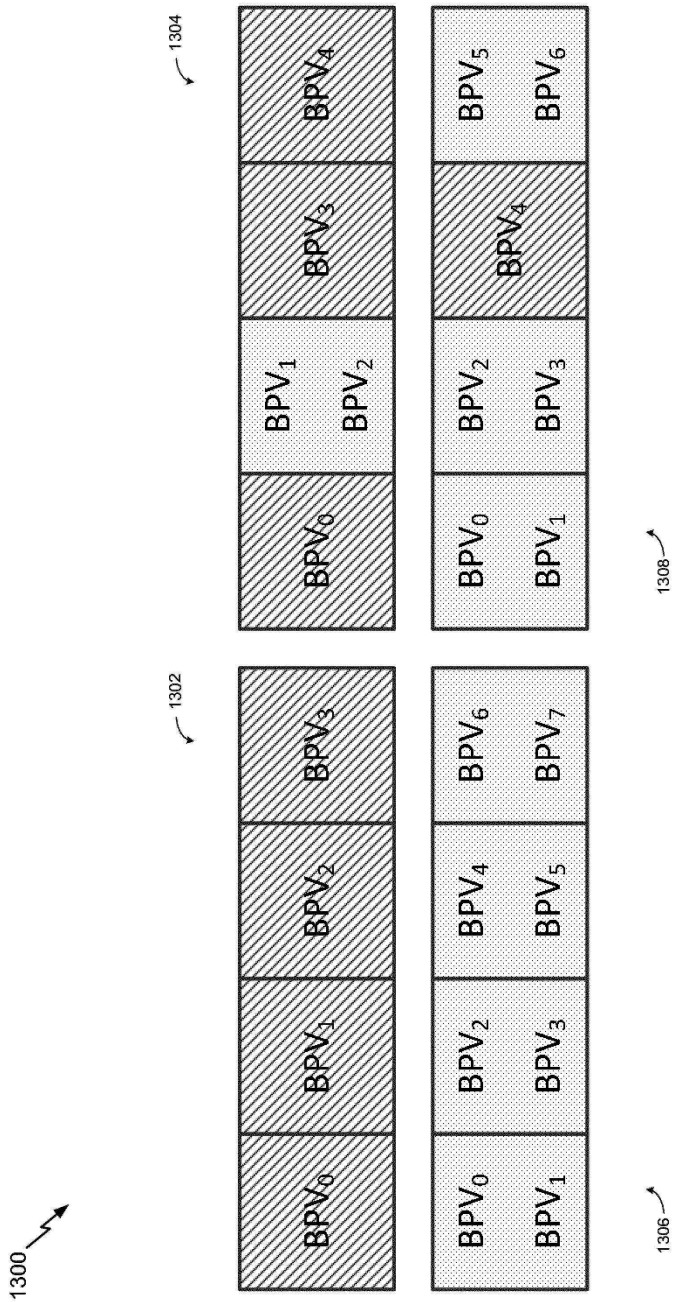
1100 ↗



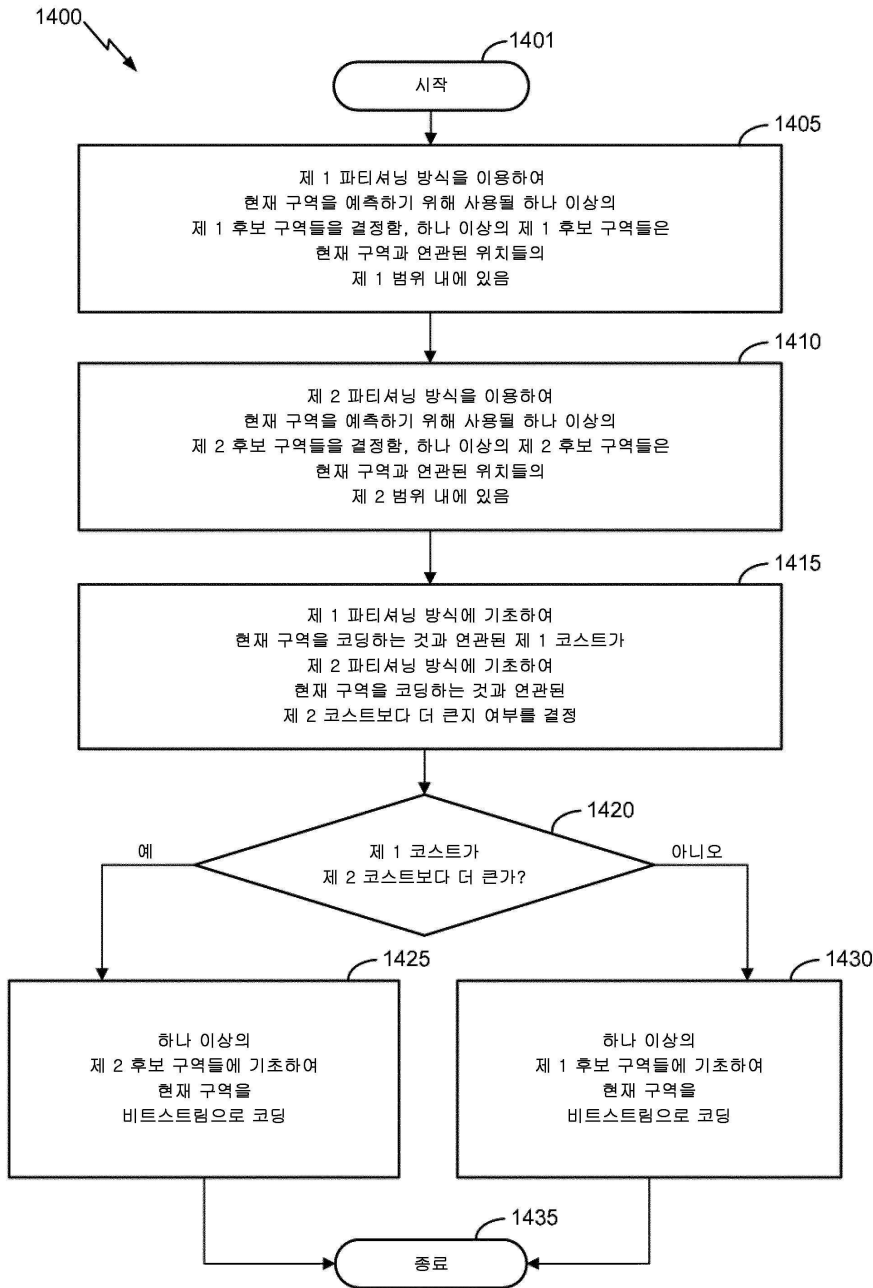
도면12



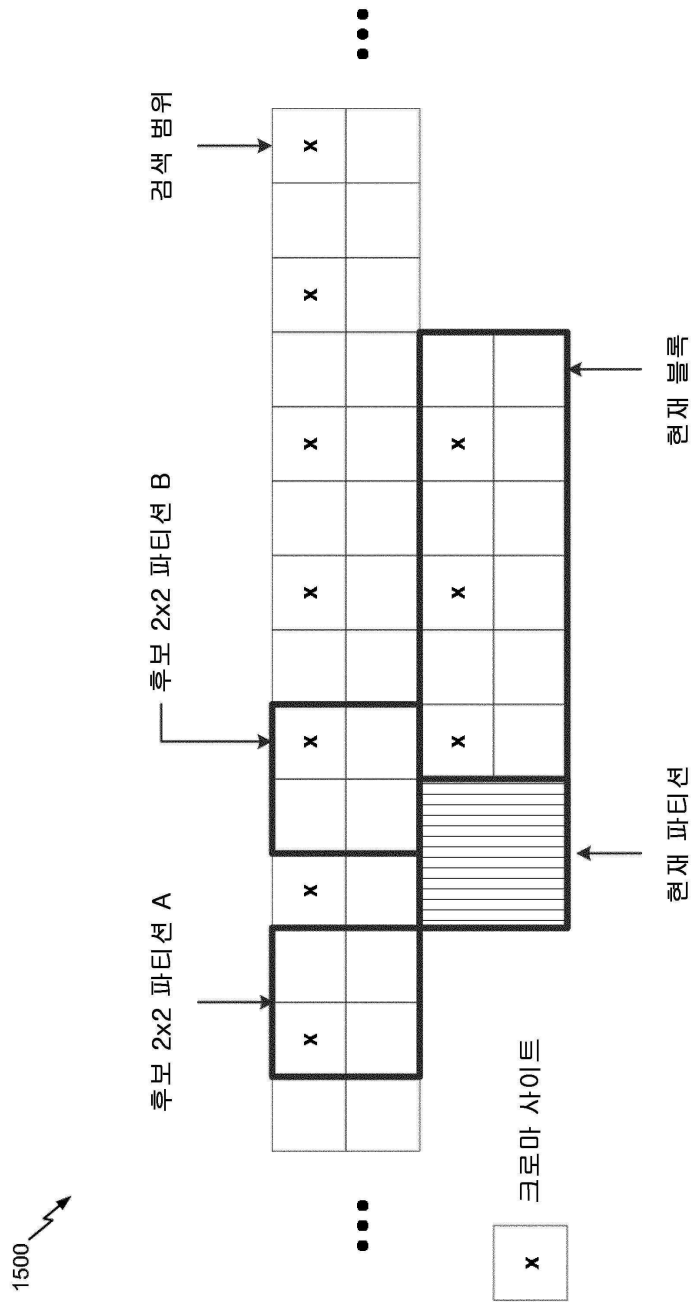
도면13



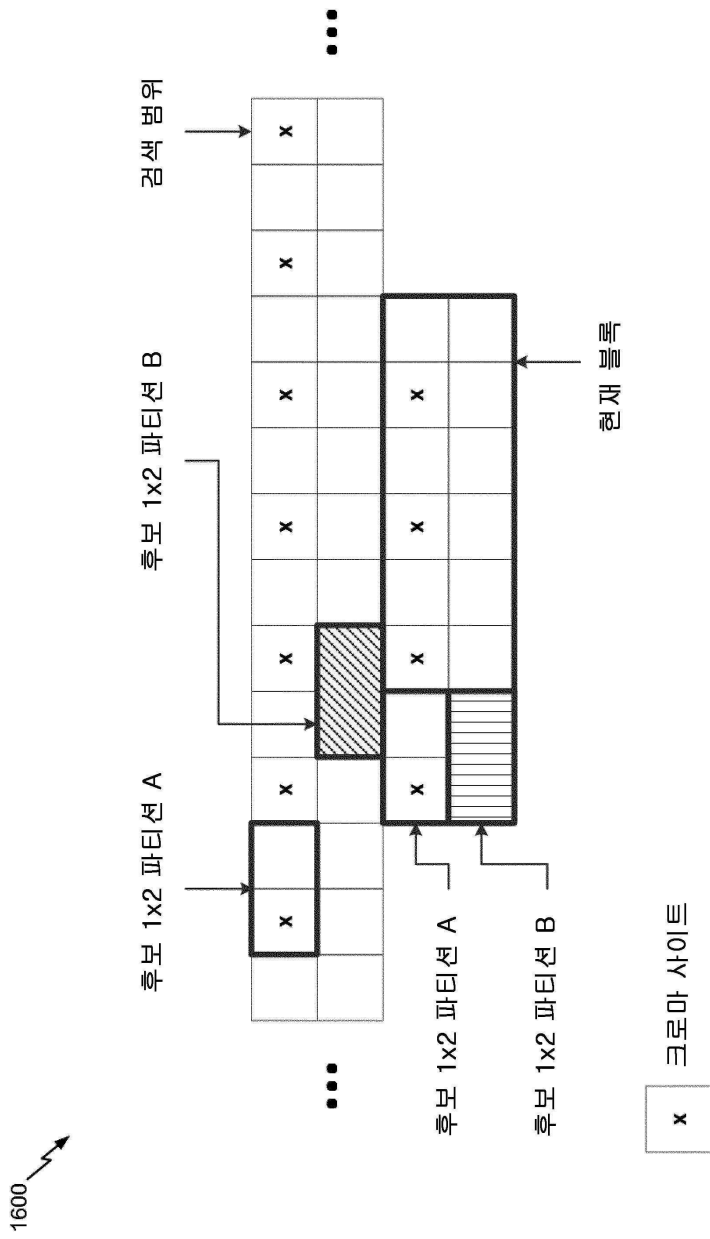
도면14



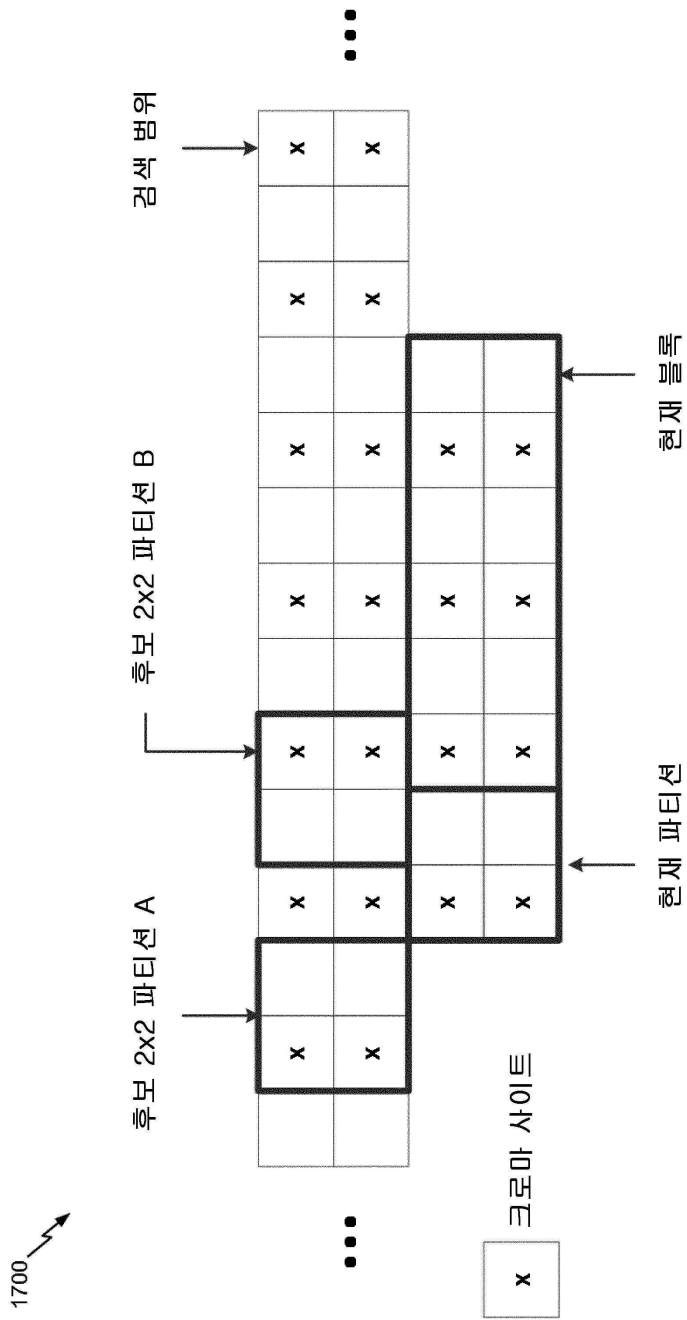
도면15



도면16



도면17



1700 ↗

도면18

