



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2016-0129026
(43) 공개일자 2016년11월08일

- | | |
|--|---|
| <p>(51) 국제특허분류(Int. Cl.)
 <i>H04N 19/86</i> (2014.01) <i>H04N 19/11</i> (2014.01)
 <i>H04N 19/119</i> (2014.01) <i>H04N 19/13</i> (2014.01)
 <i>H04N 19/14</i> (2014.01) <i>H04N 19/154</i> (2014.01)
 <i>H04N 19/172</i> (2014.01) <i>H04N 19/176</i> (2014.01)
 <i>H04N 19/27</i> (2014.01) <i>H04N 19/463</i> (2014.01)</p> <p>(52) CPC특허분류
 <i>H04N 19/86</i> (2015.01)
 <i>H04N 19/11</i> (2015.01)</p> <p>(21) 출원번호 10-2016-7026653
 (22) 출원일자(국제) 2015년03월05일
 심사청구일자 없음
 (85) 번역문제출일자 2016년09월27일
 (86) 국제출원번호 PCT/US2015/018966
 (87) 국제공개번호 WO 2015/134748
 국제공개일자 2015년09월11일
 (30) 우선권주장
 61/948,469 2014년03월05일 미국(US)
 14/638,965 2015년03월04일 미국(US)</p> | <p>(71) 출원인
 켈컴 인코포레이티드
 미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775</p> <p>(72) 발명자
 조쉬 라잔 랙스맨
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775</p> <p>리 상
 미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
 (뒷면에 계속)</p> <p>(74) 대리인
 특허법인코리아나</p> |
|--|---|

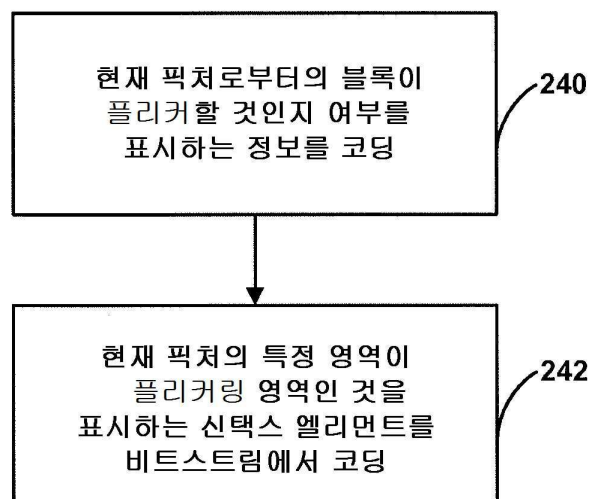
전체 청구항 수 : 총 30 항

(54) 발명의 명칭 비디오 코딩에서 플리커 검출 및 완화

(57) 요약

비디오 데이터를 코딩하도록 구성된 디바이스는: 비디오 데이터를 저장하도록 구성된 메모리, 및 적어도 하나의 프로세서를 포함한다. 적어도 하나의 프로세서는: 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하도록 구성된다. 현재 픽처로부터의 블록이 플리커할 것인지 여부를 결정은 디스플레이 순서에서 현재 픽처에서의 블록 및 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초한다.

대표도 - 도5



(52) CPC특허분류

HO4N 19/119 (2015.01)

HO4N 19/13 (2015.01)

HO4N 19/14 (2015.01)

HO4N 19/154 (2015.01)

HO4N 19/172 (2015.01)

HO4N 19/176 (2015.01)

HO4N 19/27 (2015.01)

HO4N 19/463 (2015.01)

(72) 발명자

카르체비츠 마르타

미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775

라파카 크리스티나칸트

미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775

명세서

청구범위

청구항 1

비디오 데이터를 코딩하는 방법으로서,

현재 픽처로부터의 블록이 플리커 (flicker) 할 것인지 여부를 표시하는 정보를 코딩하는 단계를 포함하며,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 결정은 디스플레이 순서에서 상기 현재 픽처에서의 블록 및 상기 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초하는, 비디오 데이터를 코딩하는 방법.

청구항 2

제 1 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 생성하는 단계를 더 포함하며,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 코딩하는 단계는, 상기 정보를 생성하는 단계에 응답하여 상기 정보를 컨텍스트 적응형 이진 산술 코딩 (CABAC) 인코딩하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 3

제 1 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 수신하는 단계를 더 포함하며,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 코딩하는 단계는, 상기 정보를 수신하는 단계에 응답하여 상기 정보를 컨텍스트 적응형 이진 산술 코딩 (CABAC) 디코딩하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 4

제 1 항에 있어서,

상기 정보를 코딩하는 단계는 플리커링 플래그를 코딩하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 5

제 4 항에 있어서,

상기 현재 픽처로부터의 블록에 대하여 공간적으로 이웃하는 블록들의 플리커링 플래그들의 값들을 CABAC 코딩에 대한 컨텍스트들로서 사용하여 상기 플리커링 플래그를 컨텍스트 적응형 이진 코딩 (CABAC) 코딩하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 6

제 1 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것을 표시하는 정보를 코딩하는 단계는, 상기 현재 픽처로부터의 블록과 상기 다음 픽처로부터의 블록 간의 절대 차이들의 합 (SAD) 또는 상기 현재 픽처로부터의 블록과 상기 다음 픽처로부터의 블록 간의 평균 제곱 에러 (MSE) 중 적어도 하나에 기초하여, 상기 정보를 코딩하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 7

제 6 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 코딩하는 단계는, 상기 SAD 또는

상기 MSE 중 적어도 하나가 임계치 미만일 경우,

상기 현재 픽처로부터의 블록의 팔레트 사이즈가 임계 팔레트 사이즈를 초과하는지 여부에 기초하여 상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 코딩하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 8

제 1 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보는 오직 영역의 코딩 유닛 (CU) 이 최소 사이즈를 갖는 경우에만 코딩되는, 비디오 데이터를 코딩하는 방법.

청구항 9

제 1 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 코딩하는 단계는,

상기 블록의 팔레트 사이즈가 임계 팔레트 사이즈를 초과하는 것, 상기 현재 픽처로부터의 블록이 예지 블록인 것, 또는 상기 현재 픽처로부터의 블록이 텍스처 블록인 것 중 적어도 하나의 경우에, 상기 정보를 코딩하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 10

제 1 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하는 단계는,

상기 병치된 블록이 플리커하는 것으로 마킹되었는지 여부에 기초하여 상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 코딩하는 단계를 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 11

제 1 항에 있어서,

상기 블록이 플리커할 것인지 여부를 표시하는 상기 정보에 기초하여 상기 현재 픽처로부터의 블록을 코딩하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 12

제 1 항에 있어서,

상기 정보를 코딩하는 단계는 상기 현재 픽처의 특정 영역이 플리커링 영역인 것을 표시하는 신텍스 엘리먼트를 비트스트림에서 코딩하는 단계를 포함하며,

상기 블록은 상기 플리커링 영역 내에 있는, 비디오 데이터를 코딩하는 방법.

청구항 13

제 12 항에 있어서,

상기 현재 픽처로부터의 블록을 코딩하는 단계는,

상기 플리커링 영역에 대한 CABAC 컨텍스트 모델링의 상태를 리셋하는 단계; 및

상기 CABAC 컨텍스트 모델링의 리셋된 상태에 기초하여 상기 현재 픽처로부터의 상기 블록을 코딩하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 14

제 12 항에 있어서,

상기 플리커링 영역인 것으로 표시된 상기 특정 영역에 대한 코딩 툴들에 관한 제약들에 기초하여 상기 현재 픽

처로부터의 블록을 코딩하는 단계를 더 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 15

제 14 항에 있어서,

상기 제약들은 상기 현재 픽처로부터의 및 상기 플리커링 영역의 상기 블록에 대하여 공간적으로 이웃하는 블록을 인트라-예측에 대하여 사용불가능한 것으로 마킹함으로써 상기 플리커링 영역에서 공간 인트라 예측을 디스에이블하는 것을 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 16

제 12 항에 있어서,

제약들은 상기 플리커링 영역 내부의 블록들을 인트라 예측에 대하여 사용불가능한 것으로 마킹하는 것을 포함하는, 비디오 데이터를 코딩하는 방법.

청구항 17

비디오 데이터를 코딩하는 디바이스로서,

비디오 데이터를 저장하도록 구성된 메모리; 및

현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하도록 구성된 적어도 하나의 프로세서를 포함하며,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 결정은 디스플레이 순서에서 상기 현재 픽처에서의 블록 및 상기 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초하는, 비디오 데이터를 코딩하는 디바이스.

청구항 18

제 17 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 코딩하기 위해, 상기 적어도 하나의 프로세서는 추가로,

상기 현재 픽처의 특정 영역이 플리커링 영역인 것을 표시하는 신택스 엘리먼트를 비트스트림에서 코딩하도록 구성되며,

상기 블록은 상기 플리커링 영역 내에 있는, 비디오 데이터를 코딩하는 디바이스.

청구항 19

제 18 항에 있어서,

상기 적어도 하나의 프로세서는 추가로,

상기 플리커링 영역인 것으로 표시된 상기 특정 영역에 대한 코딩 툴들에 관한 제약들에 기초하여 상기 현재 픽처로부터의 블록을 코딩하도록

구성되는, 비디오 데이터를 코딩하는 디바이스.

청구항 20

제 18 항에 있어서,

상기 현재 픽처로부터의 블록을 코딩하기 위해, 상기 적어도 하나의 프로세서는 추가로,

상기 플리커링 영역에 대한 CABAC 컨텍스트 모델링의 상태를 리셋하고; 그리고

상기 CABAC 컨텍스트 모델링의 리셋된 상태에 기초하여 상기 현재 픽처로부터의 블록을 코딩하도록 구성되는, 비디오 데이터를 코딩하는 디바이스.

청구항 21

제 17 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것을 표시하는 정보를 코딩하기 위해, 상기 적어도 하나의 프로세서는 추가로,

상기 현재 픽처로부터의 블록과 상기 다음 픽처로부터의 블록 간의 절대 차이들의 합 (SAD) 또는 상기 현재 픽처로부터의 블록과 상기 다음 픽처로부터의 블록 간의 평균 제곱 에러 (MSE) 중 적어도 하나에 기초하여, 상기 정보를 코딩하도록

구성되는, 비디오 데이터를 코딩하는 디바이스.

청구항 22

제 17 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보는 오직 영역의 코딩 유닛 (CU) 이 최소 사이즈를 갖는 경우에만 코딩되는, 비디오 데이터를 코딩하는 디바이스.

청구항 23

제 17 항에 있어서,

상기 적어도 하나의 프로세서는 추가로,

상기 블록이 플리커할 것인지 여부를 표시하는 정보에 기초하여 상기 현재 픽처로부터의 블록을 코딩하도록

구성되는, 비디오 데이터를 코딩하는 디바이스.

청구항 24

제 17 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 코딩하기 위해, 상기 적어도 하나의 프로세서는 추가로,

상기 블록의 팔레트 사이즈가 임계 팔레트 사이즈를 초과하는 것, 상기 현재 픽처로부터의 블록이 에지 블록인 것, 또는 상기 현재 픽처로부터의 블록이 텍스처 블록인 것 중 적어도 하나의 경우에, 상기 정보를 코딩하도록

구성되는, 비디오 데이터를 코딩하는 디바이스.

청구항 25

비디오 데이터를 코딩하는 디바이스로서,

현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하는 수단; 및

디스플레이 순서에서 상기 현재 픽처에서의 블록 및 상기 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초하여 상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 결정하는 수단을 포함하는, 비디오 데이터를 코딩하는 디바이스.

청구항 26

제 25 항에 있어서,

상기 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 상기 정보를 코딩하는 수단은, 상기 현재 픽처의 특정 영역이 플리커링 영역인 것을 표시하는 신택스 엘리먼트를 비트스트림에서 코딩하는 수단을 더 포함하며,

상기 블록은 상기 플리커링 영역 내에 있는, 비디오 데이터를 코딩하는 디바이스.

청구항 27

제 26 항에 있어서,

상기 폴리커링 영역인 것으로 표시된 상기 특정 영역에 대한 코딩 툴들에 관한 제약들에 기초하여 상기 현재 픽처로부터의 블록을 코딩하는 수단을 더 포함하는, 비디오 데이터를 코딩하는 디바이스.

청구항 28

제 26 항에 있어서,

상기 폴리커링 영역에 대한 CABAC 컨텍스트 모델링의 상태를 리셋하는 수단; 및

상기 CABAC 컨텍스트 모델링의 리셋된 상태에 기초하여 상기 현재 픽처로부터의 블록을 코딩하는 수단을 더 포함하는, 비디오 데이터를 코딩하는 디바이스.

청구항 29

제 26 항에 있어서,

상기 블록이 폴리커링 것인지 여부를 표시하는 상기 정보에 기초하여 상기 현재 픽처로부터의 블록을 코딩하는 수단을 더 포함하는, 비디오 데이터를 코딩하는 디바이스.

청구항 30

저장된 명령들을 포함하는 비-일시적 컴퓨터 판독가능 저장 매체로서,

상기 명령들은 실행될 경우, 적어도 하나의 프로세서로 하여금,

현재 픽처로부터의 블록이 폴리커링 것인지 여부를 표시하는 정보를 코딩하게 하며,

상기 현재 픽처로부터의 블록이 폴리커링 것인지 여부를 결정은 디스플레이 순서에서 상기 현재 픽처에서의 블록 및 상기 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초하는, 비-일시적 컴퓨터 판독가능 저장 매체.

발명의 설명

기술 분야

[0001] 본 출원은 2014 년 3 월 5 일에 출원된 미국 가출원 제 61/948,469 호를 우선권 주장하며, 가출원 전체는 본 명세서에 참조로서 통합된다.

[0002] 본 개시물은 비디오 데이터를 인코딩 및 디코딩하는 것과 같은 비디오 코딩 및/또는 비디오 프로세싱에 관한 것이다.

배경 기술

[0003] 디지털 비디오 능력들은 디지털 텔레비전들, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인 디지털 보조장치들 (PDA들), 랩탑 또는 데스크탑 컴퓨터들, 디지털 카메라들, 디지털 레코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 비디오 텔레컨퍼런싱 디바이스들 등을 포함한, 광범위한 디바이스들에 통합될 수 있다. 디지털 비디오 디바이스들은 MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, 파트 10, 어드밴스드 비디오 코딩 (AVC), 현재 개발 중인 고 효율 비디오 코딩 (HEVC) 표준, 및 그러한 표준들의 확장들에 의해 정의된 표준들에서 설명된 기술들과 같은 비디오 압축 기술들을 구현하여, 디지털 비디오 정보를 더 효율적으로 송신, 수신 및 저장한다.

[0004] 비디오 압축 기술들은 비디오 시퀀스들에 고유한 리던던시를 감소 또는 제거하기 위해 공간 예측 및/또는 시간 예측을 포함한다. 블록 기반 비디오 코딩에 대해, 비디오 프레임 또는 슬라이스는 비디오 블록들로 파티셔닝될 수도 있다. 각각의 블록은 추가로 파티셔닝될 수 있다. 인트라-코딩된 (I) 프레임 또는 슬라이스에서의 블록들은 동일한 프레임 또는 슬라이스에서의 이웃 블록들에서의 참조 샘플들에 대한 공간 예측을 이용하여 인코딩된다. 인터-코딩된 (P 또는 B) 프레임 또는 슬라이스에서의 비디오 블록들은 동일한 프레임 또는 슬라이스에서의 이웃 블록들에서의 참조 샘플들에 대한 공간 예측, 또는 다른 참조 프레임들에서의 참조 샘플들에 대한 시간 예측을 이용할 수도 있다. 공간 또는 시간 예측은 코딩될 블록에 대한 예측 블록을 발생

시킨다. 잔차 데이터는 코딩될 오리지널 블록과 예측 블록 간의 픽셀 차이들을 나타낸다.

[0005] 인터-코딩된 블록은 예측 블록을 형성하는 참조 샘플들의 블록을 포인팅하는 모션 벡터, 및 코딩된 블록과 예측 블록 간의 차이를 나타내는 잔차 데이터에 따라 인코딩된다. 인트라-코딩된 블록은 인트라-코딩 모드 및 잔차 데이터에 따라 인코딩된다. 추가의 압축을 위해, 잔차 데이터는 픽셀 도메인으로부터 변환 도메인으로 변환되어, 잔차 변환 계수들을 발생시킬 수도 있으며, 그 후, 이 잔차 변환 계수들은 양자화될 수도 있다. 초기에 2 차원 어레이로 정렬된 양자화된 변환 계수들은 특히 엔트로피 코딩을 위한 변환 계수들의 1 차원 벡터를 생성하기 위해 스캔될 수도 있다.

발명의 내용

과제의 해결 수단

[0006] 일반적으로, 본 개시물은 비디오 데이터에서 플리커링 (flickering) 을 검출하고 완화시키기 위한 기술들을 설명한다. 플리커링은 코딩 모드들 및 시간 특징들의 선택으로 인해 비디오 코딩에서 발생할 수도 있는 시각적 아티팩트이다. 예를 들어, 플리커링은 비디오 시퀀스가 예컨대, 인트라-예측 모드에서 오직 I-슬라이스들만을 사용하여 낮은 비트 레이트들로 코딩될 경우, 통상적으로 관측된다. 본 개시물이 기술들에 따라 구성된 비디오 코더 (즉, 비디오 인코더 또는 비디오 디코더) 는 현재 프레임의 블록을 이전 프레임의 블록과 비교하는 다양한 메트릭들을 사용하여 블록이 비디오 디코더에 의해 디코딩된 후에 플리커할 것인지 여부를 결정할 수도 있다.

[0007] 본 개시물의 기술들은 플리커링을 완화하기 위해 비디오 코더에 의해 수행될 수도 있다. 비디오 코더는 비디오 코더가 플리커할 것으로 결정하는 비디오 데이터의 블록에 대한 코딩 틀들, 예컨대 코딩 모드들을 제한함으로써 플리커링을 완화시키는 것을 시도할 수도 있다. 일 예로서, 비디오 인코더 또는 디코더는 플리커링 블록을 인트라-예측 모드로 제한할 수도 있거나, 그 블록에 대한 인트라-블록 코딩 (인트라-BC) 의 사용을 금지할 수도 있다. 일부 경우들에서, 본 개시물의 기술들은 단지 플리커링 (또는 잠재적으로 플리커링 블록) 을 식별하여, 원하는 경우, 추가의 비디오 프로세싱 단계들이 취득될 수 있게 한다.

[0008] 본 개시물의 추가의 기술들은 픽처의 특정 영역이 플리커할 것을 표시하는 정보를 코딩 (즉, 인코딩 또는 디코딩) 하기 위한 기술들에 관한 것이다. 코딩된 정보는 예컨대, 플리커링 영역을 표시하는 플리커링 플래그를 포함할 수도 있다. 비디오 인코더는 플리커링 플래그를 인코딩할 수도 있고 및/또는 디코더는 다양한 예들에서 플리커링 플래그의 값을 추론할 수도 있다. 부가적으로, 본 개시물은 플리커링 플래그의 코딩 효율을 개선하기 위한 기술들을 설명한다.

[0009] 일 예에서, 비디오 데이터를 코딩하는 방법은: 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하는 단계를 포함하며, 여기서 현재 픽처로부터의 블록이 플리커할 것인지 여부를 결정하는 디스플레이 순서에서 현재 픽처에서의 블록 및 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초한다.

[0010] 다른 예에서, 비디오 데이터를 코딩하도록 구성된 디바이스는: 비디오 데이터를 저장하도록 구성된 메모리, 및 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하도록 구성된 적어도 하나의 프로세서를 포함하며, 여기서 현재 픽처로부터의 블록이 플리커할 것인지 여부를 결정하는 디스플레이 순서에서 현재 픽처에서의 블록 및 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초한다.

[0011] 또 다른 예에서, 비디오 데이터를 코딩하도록 구성된 디바이스는: 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하는 수단, 및 디스플레이 순서에서 현재 픽처에서의 블록 및 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초하여 현재 픽처로부터의 블록이 플리커할 것인지 여부를 결정하는 수단을 포함한다.

[0012] 또 다른 예에서, 비-일시적 컴퓨터 판독가능 저장 매체는, 실행될 경우, 적어도 하나의 프로세서로 하여금: 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하게 하는 명령들을 포함하며, 여기서 현재 픽처로부터의 블록이 플리커할 것인지 여부를 결정하는 디스플레이 순서에서 현재 픽처에서의 블록 및 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초한다.

[0013] 본 개시물의 하나 이상의 양태들의 상세들은 첨부된 도면과 하기의 설명으로부터 설명된다. 본 개시물에서 설명된 기술들의 다른 피쳐들, 목적들, 및 이점들은 그 설명 및 도면들로부터 그리고 청구항들로부터 명백할 것이다.

도면의 간단한 설명

- [0014] 도 1 은 본 개시물에서 설명된 기술들을 활용할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템을 도시하는 블록 다이어그램이다.
- 도 2 는 본 개시물에 설명된 기술들을 구현할 수도 있는 예시적인 비디오 인코더를 도시하는 블록 다이어그램이다.
- 도 3 은 본 개시물에서 설명된 기술들을 구현할 수도 있는 예시적인 비디오 디코더를 도시하는 블록 다이어그램이다.
- 도 4 는 본 개시물의 기술들에 따라 플리커 검출을 수행하기 위한 프로세스를 도시하는 플로우차트이다.
- 도 5 는 본 개시물의 기술들에 따라 플리커 검출을 수행하기 위한 프로세스를 도시하는 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

- [0015] 비디오 코딩 표준들은, 그 스케일러블 비디오 코딩 (SVC) 및 멀티뷰 비디오 코딩 (MVC) 확장들을 포함하여, 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 및 ITU-T H.264 (ISO/IEC MPEG-4 AVC 로서 또한 공지됨) 를 포함한다. MVC 의 최신 공동 드래프트는 "Advanced video coding for generic audiovisual services," ITU-T Recommendation H.264, 2010 년 3 월에 기술된다.
- [0016] 부가적으로, 새로운 비디오 코딩 표준, 즉, HEVC (High Efficiency Video Coding) 은 ITU-T 비디오 코딩 전문가 그룹 (VCEG) 및 ISO/IEC 모션 픽처 전문가 그룹 (MPEG) 의 JCT-VC (Joint Collaboration Team on Video Coding) 에 의해 개발된다. HEVC Working Draft 10 또는 WD10 로 지칭되는 HEVC 표준의 최신 드래프트는 Bross 등에 의한 문서 JCTVC-L1003v34, High efficiency video coding (HEVC) text specification draft 10 (for FDIS & Last Call), Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 및 ISO/IEC JTC1/SC29/WG11, 12th Meeting: Geneva, CH, 14-23 January, 2013 에 설명되며, http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v34.zip 에서 입수가능하다.
- [0017] 본 개시물의 기술들은 비디오 코딩에 관련되고, 비디오의 뷰어에게 "플리커하는" 것으로 보이는 시각적 아티팩트들 또는 시각적 불일치를 경감시킬 수도 있다. 플리커링 아티팩트들은 코딩된 비디오 데이터의 뷰어들에게 부적절하게 보이고, 뷰어에 대한 비디오의 주관적 품질을 개선하기 위해 감소되거나 제거되어야만 한다.
- [0018] 이하 더 상세히 논의되는 것과 같이, 본 개시물의 기술들은 비디오 코더가 플리커링 아티팩트들을 감소시키거나 제거할 수 있게 할 수도 있다. 본 개시물의 기술들에 따라 구성된 비디오 인코더는 플리커할 비디오 데이터의 픽처의 영역을 식별하는 것을 시도할 수도 있다. 비디오 인코더는 현재 픽처의 블록과 다른 픽처의 병치된 블록 간의 차이들에 기초하여 잠재적으로 플리커하는 영역을 식별할 수도 있다.
- [0019] 비디오 인코더가 특정 영역이 플리커할 수도 있는 것으로 결정한다면, 비디오 인코더는 플리커링 영역을 표시하는 정보, 예컨대 신덱스 엘리먼트를 인코딩할 수도 있다. 인코딩된 정보는 일반적으로 상호적인 디코딩 프로세스를 수행하도록 구성된 비디오 디코더가 이하 더 상세히 설명되는 것과 같이, 플리커링을 감소시키기 위해 현재 블록에 특정 제약들을 적용해야만 하는 것을 표시할 수도 있다.
- [0020] 도 1 은 본 개시물의 예들에 따라 플리커 검출 및 플리커 완화를 위한 기술들을 활용하도록 구성될 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (10) 을 도시하는 블록 다이어그램이다. 도 1 에 도시된 바와 같이, 시스템 (10) 은 인코딩된 비디오를 통신 채널 (16) 을 통해 목적지 디바이스 (14) 로 송신하는 소스 디바이스 (12) 를 포함한다. 인코딩된 비디오 데이터는 또한 저장 매체 (34) 또는 파일 서버 (36) 상에 저장될 수도 있고, 원하는 바에 따라 목적지 디바이스 (14) 에 의해 액세스될 수도 있다. 저장 매체 또는 파일 서버에 저장될 경우, 비디오 인코더 (20) 는 네트워크 인터페이스, 콤팩트 디스크 (CD), 블루-레이 또는 디지털 비디오 디스크 (DVD) 버너 또는 스탬핑 기능 디바이스와 같은 또 다른 디바이스, 또는 코딩된 비디오 데이터를 저장 매체에 저장하기 위한 다른 디바이스들에 코딩된 비디오 데이터를 제공할 수도 있다. 유사하게, 네트워크 인터페이스, CD 또는 DVD 리더기, 등등과 같은 비디오 디코더 (30) 와 분리된 디바이스는 저장 매체로부터 코딩된 비디오 데이터를 추출하고 추출된 데이터를 비디오 디코더 (30) 에 제공할 수도 있다.
- [0021] 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 태블릿 컴퓨

터들, 셋탑 박스들, 소위 스마트폰들과 같은 전화기 핸드셋들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들, 등을 포함한 광범위한 디바이스들 중 임의의 디바이스를 포함할 수도 있다. 다수의 경우들에서, 그러한 디바이스들은 무선 통신을 위해 장비될 수도 있다. 따라서, 통신 채널 (16) 은 인코딩된 비디오 데이터의 송신에 적합한, 무선 채널, 유선 채널, 또는 무선 채널과 유선 채널의 조합을 포함할 수도 있다. 유사하게, 파일 서버 (36) 는 인터넷 접속을 포함하는 임의의 표준 데이터 접속을 통해 목적지 디바이스 (14) 에 의해 액세스될 수도 있다. 이는 파일 서버 상에 저장되는 인코딩된 비디오 데이터에 액세스하기에 적합한 무선 채널 (예를 들어, Wi-Fi 커넥션), 유선 커넥션 (예를 들어, DSL, 케이블 모뎀, 등), 또는 이들 양자의 조합을 포함할 수도 있다.

[0022] 그 기술들은, 본 개시물의 예들에 따라, 공중 경유 (over-the-air) 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 예를 들어, 인터넷을 통한 스트리밍 비디오 송신들, 데이터 저장 매체 상의 저장을 위한 디지털 비디오의 인코딩, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션들과 같은 다양한 멀티미디어 애플리케이션들 중 임의의 애플리케이션들의 지원으로 비디오 코딩에 적용될 수도 있다. 일부 예들에 있어서, 시스템 (10) 은 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 및/또는 비디오 전화와 같은 애플리케이션들을 지원하기 위해 일방향 또는 양방향 비디오 송신을 지원하도록 구성될 수도 있다.

[0023] 일부 예들에서, 본 개시물에 설명된 기술들은 HEVC 에 따른 비디오 코딩에서 특히 유용할 수도 있다. 그러나, 본 개시물에서 설명된 기술들은 임의의 특정 비디오 코딩 표준에 한정되는 것으로 고려되지 않는다. 본 개시물에서 설명된 기술들은 또한, 비디오 코딩에 기반한 비-표준들에 적용가능할 수도 있다.

[0024] 도 1 의 예에서, 소스 디바이스 (12) 는 비디오 소스 (18), 비디오 인코더 (20), 변조기/복조기 (22) 및 송신기 (24) 를 포함한다. 소스 디바이스 (12) 에 있어서, 비디오 소스 (18) 는 비디오 카메라와 같은 비디오 캡처 디바이스, 이전에 캡처된 비디오를 포함하는 비디오 아카이브, 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 피드 인터페이스, 및/또는 컴퓨터 그래픽 데이터를 소스 비디오로서 생성하기 위한 컴퓨터 그래픽 시스템과 같은 소스, 또는 그러한 소스들의 조합을 포함할 수도 있다. 일 예로서, 비디오 소스 (18) 가 비디오 카메라이면, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 예컨대, 스마트폰들 또는 태블릿 컴퓨터들 내에서 제공될 수도 있는, 소위 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 하지만, 본 개시물에서 설명된 기술들은 일반적으로 비디오 코딩에 적용가능할 수도 있으며, 무선 및/또는 유선 애플리케이션, 또는 인코딩된 비디오 데이터가 로컬 디스크 상에 저장되는 애플리케이션에 적용될 수도 있다.

[0025] 캡처, 프리-캡처, 또는 컴퓨터에 의해 생성된 비디오는 비디오 인코더 (20) 에 의해 인코딩될 수도 있다. 인코딩된 비디오 정보는 유선 또는 무선 통신 프로토콜과 같은 통신 표준에 따라 모뎀 (22) 에 의해 변조되고, 송신기 (24) 를 통해 목적지 디바이스 (14) 로 송신될 수도 있다. 모뎀 (22) 은 다양한 믹서들, 필터들, 증폭기들 또는 신호 변조용으로 설계된 다른 컴포넌트들을 포함할 수도 있다. 송신기 (24) 는 증폭기들, 필터들, 및 무선 통신의 경우에 하나 이상의 안테나들을 포함한, 데이터를 송신하기 위해 설계된 회로들을 포함할 수도 있다.

[0026] 비디오 인코더 (20) 에 의해 인코딩되는 캡처, 프리-캡처, 또는 컴퓨터에 의해 생성된 비디오는 또한, 이후 소비를 위해 저장 매체 (34) 또는 파일 서버 (36) 상에 저장될 수도 있다. 저장 매체 (34) 는 블루-레이 디스크들, DVD들, CD-ROM들, 플래시 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 적합한 디지털 저장 매체들을 포함할 수도 있다. 저장 매체 (34) 에 저장된 인코딩된 비디오는 그 후, 디코딩 및 플레이백을 위해 목적지 디바이스 (14) 에 의해 액세스될 수도 있다. 도 1 에 도시되지 않았지만, 일부 예들에서, 저장 매체 (34) 및/또는 파일 서버 (36) 는 송신기 (24) 의 출력을 저장할 수도 있다.

[0027] 또한, 저장 매체 (34) 와 파일 서버 (36) 가 비디오 인코더 (20) 로부터 비디오 데이터를 수신하는 것으로 도시되지만, 일부 예들에서, 저장 매체 (34) 와 파일 서버 (36) 는 비디오 인코더 (20) 보다, 송신기 (24) 또는 일부 다른 유닛으로부터 비디오 데이터를 수신할 수도 있다. 유사하게, 저장 매체 (34) 와 파일 서버 (36) 가 비디오 데이터를 비디오 디코더 (30) 에 출력하는 것으로 도시되지만, 일부 예들에서, 저장 매체 (34) 와 파일 서버 (36) 는 비디오 디코더 (30) 보다, 수신기 (26) 또는 일부 다른 유닛에 비디오 데이터를 출력할 수도 있다. 이들 예들에서, 비디오 디코더 (30) 는 수신기 (26) 또는 일부 다른 유닛으로부터 비디오 데이터를 수신할 수도 있다.

[0028] 파일 서버 (36) 는, 인코딩된 비디오를 저장하고 그 인코딩된 비디오를 목적지 디바이스 (14) 로 송신하는 것이 가능한 임의의 타입의 서버일 수도 있다. 예시적인 파일 서버들은 웹 서버 (예를 들어, 웹 사이트용), FTP

서버, 네트워크 부착형 스토리지 (NAS) 디바이스들, 로컬 디스크 드라이브, 또는 인코딩된 비디오 데이터를 저장하고 목적지 디바이스에 송신할 수 있는 임의의 다른 타입의 디바이스를 포함한다. 인코딩된 비디오 데이터의 파일 서버 (36)로부터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이들 양자의 조합일 수도 있다. 파일 서버 (36)는 인터넷 접속을 포함하는 임의의 표준 데이터 접속을 통해 목적지 디바이스 (14)에 의해 액세스될 수도 있다. 이는 파일 서버 상에 저장되는 인코딩된 비디오 데이터에 액세스하기에 적합한 무선 채널 (예를 들어, Wi-Fi 커넥션), 유선 커넥션 (예를 들어, DSL, 케이블 모뎀, 이더넷, USB, 등), 또는 이들 양자의 조합을 포함할 수도 있다.

[0029] 도 1의 예에서, 목적지 디바이스 (14)는 수신기 (26), 모뎀 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32)를 포함한다. 목적지 디바이스 (14)의 수신기 (26)는 채널을 통해 정보를 수신하고, 모뎀 (28)은 비디오 디코더 (30)에 대한 복조된 비트스트림을 생성하기 위해 정보를 복조한다. 채널 (16)을 통해 통신된 정보는 비디오 디코더 (30)에 의해 비디오 데이터를 디코딩하는데 사용하기 위해 비디오 인코더 (20)에 의해 생성된 다양한 신택스 정보를 포함할 수도 있다. 그러한 신택스는 또한, 저장 매체 (34) 또는 파일 서버 (36)에 저장된 인코딩된 비디오 데이터가 포함될 수도 있다. 비디오 인코더 (20)와 비디오 디코더 (30)의 각각은 비디오 데이터를 인코딩하거나 디코딩할 수 있는 개별 인코더-디코더 (CODEC)의 부분을 형성할 수도 있다.

[0030] 디스플레이 디바이스 (32)는 목적지 디바이스 (14)와 통합될 수도 있거나 또는 목적지 디바이스 (14)외부에 있을 수도 있다. 일부 예들에 있어서, 목적지 디바이스 (14)는 통합된 디스플레이 디바이스를 포함할 수도 있고, 또한, 외부 디스플레이 디바이스와 인터페이스하도록 구성될 수도 있다. 다른 예들에 있어서, 목적지 디바이스 (14)는 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32)는 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 다른 유형의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 것을 포함할 수도 있다.

[0031] 도 1의 예에서, 통신 채널 (16)은 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적 송신 라인들과 같은 임의의 무선 또는 유선 통신 매체, 또는 무선 및 유선 매체들의 임의의 조합을 포함할 수도 있다. 통신 채널 (16)은 로컬 영역 네트워크, 광역 네트워크, 또는 인터넷과 같은 글로벌 네트워크와 같은 패킷 기반 네트워크의 부분을 형성할 수도 있다. 통신 채널 (16)은 일반적으로, 유선 또는 무선 매체의 임의의 적절한 조합을 포함하여 소스 디바이스 (12)로부터 목적지 디바이스 (14)로 비디오 데이터를 송신하기 위한, 임의의 적절한 통신 매체, 또는 상이한 통신 매체의 집합체 (collection)를 나타낸다. 통신 채널 (16)은 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12)로부터 목적지 디바이스 (14)로의 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0032] 비디오 인코더 (20) 및 비디오 디코더 (30)는 비디오 압축 표준, 예컨대 ITU-T VCEG (Video Coding Experts Group) 및 ISO/IEC MPEG (Motion Picture Experts Group)의 JCT-VC (Joint Collaborative Team on Video Coding)에 의해 개발되는 HEVC (High Efficiency Video Coding) 표준과 같은 비디오 압축 표준 및/또는 HEVC 표준의 확장물들에 따라 동작할 수도 있다. HEVC 범위 확장물들은 현재 개발 중에 있다. "HEVC Working Draft 10" 또는 "WD10"로 지칭되는 HEVC 표준의 최근 드래프트는 Bross 등에 의한 문서 JCTVC-L1003v34, "High efficiency video coding (HEVC) text specification draft 10 (for FDIS & Last Call)," Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 12th Meeting: Geneva, CH, 14-23 January, 2013에 설명되며, 2015년 2월 23일자로 http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v34.zip에서 다운로드할 수 있다. 따라서, HEVC WD10의 전체 콘텐츠는 참조에 의해 본 명세서에 통합된다.

[0033] 비록 도 1에 도시되지는 않지만, 일부 양태들에 있어서, 비디오 인코더 (20) 및 비디오 디코더 (30)는 오디오 인코더 및 디코더와 각각 통합될 수도 있으며, 오디오 및 비디오 양자의 인코딩을 공통 데이터 스트림 또는 별개의 데이터 스트림들로 처리하기 위해 적절한 MUX-DEMUX 유닛들 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능하다면, 일부 예들에 있어서, MUX-DEMUX 유닛들은 멀티플렉서 프로토콜에 부합할 수도 있다. 예들은 ITU H.223 멀티플렉서 프로토콜, 또는 사용자 데이터그램 프로토콜 (UDP)과 같은 다른 프로토콜을 포함한다.

[0034] 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSP들), 주문형 집적회로들 (ASIC들), 필드 프로그래밍가능 게이트 어레이들 (FPGA들), 이산 로직,

소프트웨어, 하드웨어, 펌웨어 또는 이들의 임의의 조합들과 같은 다양한 적합한 인코더 회로 중 임의의 회로로서 구현될 수도 있다. 기술들이 부분적으로 소프트웨어로 구현될 경우, 디바이스는 적합한 비-일시적인 컴퓨터 판독가능 매체에 소프트웨어에 대한 명령들을 저장하고, 본 개시물의 기술들을 수행하기 위해 하나 이상의 프로세서들을 사용하는 하드웨어로 명령들을 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이들 중 어느 하나는 개별 디바이스에 있어서 결합된 인코더/디코더 (CODEC) 의 부분으로서 통합될 수도 있다.

[0035] 비디오 디코더 (30) 를 포함하는 디바이스의 예들은 무선 통신 디바이스, 집적 회로 (IC), 및 마이크로프로세서를 포함한다. 유사하게, 비디오 인코더 (20) 를 포함하는 디바이스의 예들은 무선 통신 디바이스, 집적 회로 (IC), 및 마이크로프로세서를 포함한다.

[0036] 일부 예들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 각각, 본 개시물에 설명된 기술들을 구현하도록 구성된 하나 이상의 프로세서들을 포함할 수도 있다. 비디오 인코더 (20) 또는 비디오 디코더 (30) 를 포함하는 디바이스는 또한, 비디오 인코더 (20) 와 비디오 디코더 (30) 가 활용하는 비디오 데이터를 저장하는 메모리를 포함할 수도 있다. 비디오 인코더 (20) 또는 비디오 디코더 (30) 를 포함하는 디바이스가 IC 칩 또는 마이크로프로세서를 포함하는 예들에서, 메모리는 IC 칩 또는 마이크로프로세서의 로컬 메모리일 수도 있거나, 또는 IC 칩 또는 마이크로프로세서를 포함하는 장치 (예컨대, 스마트폰 또는 태블릿과 같은 무선 통신 디바이스) 의 시스템 메모리일 수도 있거나, 또는 로컬 메모리와 시스템 메모리의 조합일 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 를 포함하는 디바이스가 무선 통신 디바이스 (또는 다른 그러한 디바이스) 인 예들에서, 메모리는 비디오 인코더 (20) 또는 비디오 디코더 (30) 의 로컬 메모리, 비디오 인코더 (20) 또는 비디오 디코더 (30) 를 포함하는 디바이스의 시스템 메모리, 또는 로컬 메모리와 시스템 메모리의 조합일 수도 있다.

[0037] 비디오 인코더 (20) 는 플리커 검출 및 완화를 위해 본 개시물의 기술들 중 임의의 것 또는 전부를 구현할 수도 있다. 유사하게, 비디오 디코더 (30) 는 플리커 검출 및 완화를 위해 상기 기술들 중 임의의 것 또는 전부를 구현할 수도 있다. 본 개시물에서 설명된 바와 같은 비디오 코더는 비디오 인코더 또는 비디오 디코더를 지칭할 수도 있다. 유사하게, 비디오 코딩 유닛은 비디오 인코더 또는 비디오 디코더를 지칭할 수도 있다. 유사하게, 비디오 코딩은, 비디오 인코딩 또는 비디오 디코딩을 지칭할 수도 있다.

[0038] 디지털 비디오 디바이스들은 디지털 비디오 정보를 더 효율적으로 인코딩 및 디코딩하기 위해 비디오 압축 기술들을 구현한다. 비디오 압축은 비디오 시퀀스들에 고유한 리던던시를 감소 또는 제거하기 위해 공간 (인트라-프레임) 예측 및/또는 시간 (인터-프레임) 예측을 적용할 수도 있다.

[0039] JCT-VC 는 예컨대, 앞서 논의된 HEVC WD10 에서 설명된 것과 같은 HEVC 표준을 개발하였다. HEVC 표준화 노력들은 HEVC 테스트 모델 (HM) 로서 지칭되는 비디오 코딩 디바이스의 모델에 기초한다. HM 은 예를 들어, ITU-T H.264/AVC 에 따른 기존의 디바이스들에 대하여 비디오 코딩 디바이스들의 수개의 부가적인 능력들을 가정한다. 예를 들어, H.264 가 9 개의 인트라-예측 인코딩 모드들을 제공하지만, HEVC HM 은 33 개 만큼 많은 인트라-예측 인코딩 모드들을 제공할 수도 있다. 이하 섹션은 HM 의 특정 양태들을 더 상세히 논의할 것이다.

[0040] HEVC 에 따른 비디오 코딩을 위해, 비디오 프레임은 코딩 유닛들로 파티셔닝될 수도 있다. 코딩 유닛 (CU) 은 일반적으로, 비디오 압축을 위해 다양한 코딩 툴들이 적용되는 기본 유닛으로서 기능하는 이미지 영역을 지칭한다. CU 는 통상적으로, Y 로 표시된 루미넌스 성분, 및 U 및 V 로 표시된 2 개의 크로마 성분들을 갖는다. 비디오 샘플링 포맷에 의존하여, U 및 V 성분들의 사이즈는, 샘플들의 수와 관련하여, Y 성분의 사이즈와 동일하거나 상이할 수도 있다.

[0041] CU 는 통상적으로 정사각형이고, 예컨대 ITU-T H.264 와 같은 다른 비디오 코딩 표준들 하에서 소위 매크로블록과 유사한 것으로 간주될 수도 있다. HEVC 표준의 현재 제안된 양태들 중 일부에 따른 코딩은 본 출원에서 예시의 목적으로 설명될 것이다. 하지만, 본 개시물에서 설명된 기술들은, HEVC, H.264 또는 다른 표준의 범위 확장물들 또는 전매특허의 비디오 코딩 프로세스들에 따라 정의된 프로세서들과 같은 다른 비디오 코딩 프로세스들에 대해 유용할 수도 있다.

[0042] HM 에 따르면, CU 는 하나 이상의 예측 유닛들 (PU들) 및/또는 하나 이상의 변환 유닛들 (TU들) 을 포함할 수도 있다. 비트스트림 내의 선택스 데이터는 최대 코딩 유닛 (LCU) 을 정의할 수도 있는데, 이것은 픽셀들의 수의 관점에서 가장 큰 코딩 유닛이다. 일반적으로, CU가 사이즈 구별을 갖지 않는다는 점을 제외하면, CU 는

H.264 의 매크로 블록과 유사한 목적을 갖는다. 따라서, CU 는 서브-CU들로 분할될 수도 있다. 일반적으로, 본 개시물에서 CU 의 참조들은 LCU 의 서브-CU 또는 픽처의 최대 코딩 유닛을 지칭할 수도 있다. LCU 는 서브-CU들로 분할될 수도 있고, 각각의 서브-CU 는 추가로 서브-CU들로 분할될 수도 있다. 비트스트림에 대한 선택스 데이터는 CU 깊이로 지칭되는, LCU 가 분할될 수도 있는 최대 횡수를 정의한다. 따라서, 비트스트림은 최소 코딩 단위 (SCU) 를 또한 정의할 수도 있다. 본 개시물은 또한, CU, PU, 또는 TU 중 임의의 것을 지칭하기 위해 용어 "블록" 또는 "부분 (portion)" 을 사용한다. 일반적으로, "부분" 은 비디오 프레임의 임의의 서브-세트를 지칭할 수도 있다.

[0043] LCU 는 쿼드트리 데이터 구조와 연관될 수도 있다. 일반적으로, 쿼드트리 데이터 구조는 CU 당 하나의 노드를 포함하는데, 여기서 루트 노드는 LCU 에 대응한다. CU 가 4 개의 서브-CU들로 분할된다면, 그 CU 에 대응하는 노드는 4 개의 리프 노드들을 포함하고, 그 각각은 서브-CU들 중 하나에 대응한다. 쿼드트리 데이터 구조의 각각의 노드는 대응하는 CU 에 대해 선택스 데이터를 제공할 수도 있다. 예를 들면, 쿼드트리 내의 노드는, 그 노드에 대응하는 CU 가 서브-CU들로 분할되는지의 여부를 나타내는 분할 플래그를 포함할 수도 있다. CU 에 대한 선택스 엘리먼트들은 재귀적으로 정의될 수도 있고, CU가 서브 CU들로 분할되는지 여부에 의존할 수도 있다. CU 가 추가로 분할되지 않으면, 그것은 리프-CU 로서 지칭된다. 본 개시물에서, 리프-CU 의 4 개의 서브-CU들은, 원래의 리프-CU 의 명백한 분할이 존재하지 않을 경우에도 리프-CU들로서 또한 지칭될 것이다. 예를 들면, 16x16 사이즈에서 CU가 추가로 분할되지 않는다면, 4 개의 8x8 서브-CU들은, 16x16 CU 가 분할되지 않았지만, 리프-CU들로서 또한 지칭될 것이다.

[0044] 리프-CU 는 하나 이상의 예측 유닛들 (PU들) 을 포함할 수도 있다. 일반적으로, PU 는 대응하는 CU 의 전부 또는 부분을 나타내며, PU 에 대한 참조 샘플을 추출하기 위한 데이터를 포함할 수도 있다. 예를 들면, PU 가 인터-모드 인코딩될 경우, PU 는 그 PU 에 대한 모션 벡터를 정의하는 데이터를 포함할 수도 있다. 모션 벡터를 정의하는 데이터는 예를 들어, 모션 벡터의 수평 컴포넌트, 모션 벡터의 수직 컴포넌트, 모션 벡터에 대한 분해능 (예를 들어, 1/4 픽셀 정밀도 또는 1/8 픽셀 정밀도), 모션 벡터가 포인팅하는 참조 프레임, 및/또는 모션 벡터에 대한 참조 리스트 (예를 들어, 리스트 0 또는 리스트 1) 를 기술할 수도 있다. PU(들) 을 정의하는 리프-CU 에 대한 데이터는, 예를 들어, 하나 이상의 PU들로의 CU 의 파티셔닝을 기술할 수도 있다. 파티셔닝 모드들은 CU 가 코딩되지 않는지, 인트라-예측 모드 인코딩되는지, 또는 인터-예측 모드 인코딩되는지 여부에 의존하여 상이할 수도 있다. 인트라 코딩에 대해, PU 는 이하 설명되는 리프 변환 유닛과 동일한 것으로 취급될 수도 있다.

[0045] HEVC 표준은 변환 유닛들 (TU들) 에 따른 변환들에 대해 허용하며, 이는 상이한 CU들에 대해 상이할 수도 있다. TU들은 통상적으로, 파티셔닝된 LCU 에 대해 정의된 소정의 CU 내에서의 PU들의 사이즈에 기초하여 사이징되지만, 이것이 항상 그 경우인 것은 아닐 수도 있다. TU들은 통상적으로 PU들과 동일한 사이즈이거나 그 보다 더 작다. 몇몇 실시예들에서, CU 에 대응하는 잔차 샘플들은, "잔차 쿼드 트리 (RQT)"로서 알려진 쿼드트리 구조를 사용하여 더 작은 유닛들로 서브분할될 수도 있다. RQT 의 리프 노드들은 변환 유닛들 (TU들) 로서 지칭될 수도 있다. TU들과 연관된 픽셀 차이 값들은 변환 계수들을 생성하도록 변환될 수도 있으며, 이 변환 계수들은 양자화될 수도 있다. TU 는 루마 변환 블록 및 2 개의 크로마 변환 블록들을 포함한다. 이와 같이, TU 에 적용되는 이하 논의되는 임의의 코딩 프로세스는 실제로, 루마 및 크로마 변환 블록들에 적용될 수도 있다.

[0046] 일반적으로, PU 는 예측 프로세스와 관련된 데이터를 포함한다. 예를 들어, PU 가 인트라-모드 인코딩될 경우, PU 는 그 PU 에 대한 인트라-예측 모드를 기술하는 데이터를 포함할 수도 있다. 다른 예로서, PU 가 인터-모드 인코딩될 경우, PU 는 그 PU 에 대한 모션 벡터를 정의하는 데이터를 포함할 수도 있다.

[0047] 일반적으로, TU 는 변환 및 양자화 프로세스들을 위해 사용된다. 하나 이상의 PU들을 갖는 소정의 CU 는 또한 하나 이상의 변환 유닛들 (TU들) 을 포함할 수도 있다. 예측에 후속하여, 비디오 인코더 (20) 는 PU 에 따른 코딩 노드에 의해 식별되는 비디오 블록으로부터 잔차 값들을 계산할 수도 있다. 그 후에, 코딩 노드는 원래의 비디오 블록보다는 잔차 값들을 참조하도록 업데이트된다. 잔차 값들은, 변환들 및 TU들에 명시된 다른 변환 정보를 사용하여 변환 계수들로 변환되고, 양자화되고, 스캐닝되어 엔트로피 코딩을 위한 직렬화된 변환 계수들을 생성할 수도 있는 픽셀 차이값들을 포함한다. 코딩 노드는 이들 직렬화된 변환 계수들을 지칭하도록 다시 한번 업데이트될 수도 있다. 본 개시물은 통상적으로 용어 "비디오 블록" 을 사용하여, CU 의 코딩 노드를 지칭한다. 일부 특정 경우들에 있어서, 본 개시물은 또한 용어 "비디오 블록" 을 사용하여, 코딩 노드 그리고 PU들 및 TU들을 포함하는 트리블록, 즉, LCU 또는 CU 를 지칭할 수도 있다.

- [0048] 비디오 시퀀스는 통상적으로, 비디오 프레임들 또는 픽처들의 시리즈를 포함한다. 픽처들의 그룹 (GOP) 은 일반적으로 비디오 픽처들의 하나 이상의 시리즈를 포함한다. GOP 는, 그 GOP 에 포함된 다수의 픽처들을 기술하는 선택스 데이터를 GOP 의 헤더, 픽처들의 하나 이상의 헤더, 또는 다른 곳에 포함할 수도 있다. 픽처의 각각의 슬라이스는 개별 슬라이스에 대한 인코딩 모드를 기술하는 슬라이스 선택스 데이터를 포함할 수도 있다. 비디오 인코더 (20) 는 통상적으로, 비디오 데이터를 인코딩하기 위해 개별적인 비디오 슬라이스들 내의 비디오 블록들에 대해 동작한다. 비디오 블록은 CU 내의 코딩 노드에 대응할 수도 있다. 비디오 블록들은 고정 또는 가변 사이즈들을 가질 수도 있으며, 명시된 코딩 표준에 따라 사이즈가 상이할 수도 있다.
- [0049] 블록 (예컨대, 비디오 데이터의 예측 유닛) 을 코딩하기 위해, 블록에 대한 예측자가 먼저 도출된다. 예측 블록으로도 지칭되는 예측자는 인트라 (I) 예측 (즉, 공간 예측) 또는 인터 (P 또는 B) 예측 (즉, 시간 예측) 을 통해 도출될 수도 있다. 따라서, 일부 예측 유닛들은 동일한 프레임 (또는 슬라이스) 에서의 이웃 참조 블록들에서의 참조 샘플들에 대하여 공간 예측을 사용하여 인트라-코딩 (I) 될 수도 있고, 다른 예측 유닛들은 다른 이전-코딩된 프레임들 (또는 슬라이스들) 에서의 참조 샘플들의 블록들에 대하여 단방향으로 인터-코딩 (P) 되거나 양방향으로 인터-코딩 (B) 될 수도 있다. 각 경우에, 참조 샘플들은 블록이 코딩되기 위한 예측 블록을 형성하는데 사용될 수도 있다.
- [0050] 예측 블록의 식별시, 원래의 비디오 데이터 블록에서의 픽셀들과 그 예측 블록에서의 픽셀들 간의 차이가 결정된다. 이러한 차이는 예측 잔차 데이터로서 지칭될 수도 있고, 코딩될 블록에서의 픽셀 값들과 코딩된 블록을 나타내기 위해 선택된 예측 블록에서의 픽셀 값들 간의 픽셀 차이들을 표시한다. 더 양호한 압축을 달성하기 위해, 예측 잔차 데이터는 예컨대, 이산 코사인 변환 (DCT), 정수 변환, K-L (Karhunen-Loeve) 변환, 또는 변환 계수들을 생성하기 위한 다른 변환을 사용하여 변환될 수도 있다.
- [0051] TU 와 같은 변환 블록에서의 잔차 데이터는 공간적, 픽셀 도메인에 상주하는 픽셀 차이 값들의 2 차원 (2D) 어레이로 배열될 수도 있다. 변환은 잔차 픽셀 값들을 주파수 도메인과 같은 변환 도메인에서의 변환 계수들의 2 차원 어레이로 컨버팅한다.
- [0052] 추가의 압축을 위해, 변환 계수들은 엔트로피 코딩 이전에 양자화될 수도 있다. 그 후에, 엔트로피 코더는 컨텍스트 적응형 가변 길이 코딩 (CAVLC), 컨텍스트 적응형 이진 산술 코딩 (CABAC), 확률 간격 파터닝 엔트로피 (PIPE) 코딩, 등과 같은 엔트로피 코딩을 양자화된 변환 계수들에 적용한다. 일부 예들에 있어서, 비디오 인코더 (20) 는 양자화된 변환 계수들을 스캔하기 위한 미리 정의된 스캔 순서를 활용하여, 엔트로피 인코딩될 수 있는 직렬화된 벡터를 생성할 수도 있다. 다른 예들에 있어서, 비디오 인코더 (20) 는 적응적 스캔을 수행할 수도 있다. 1 차원 벡터를 형성하기 위해 양자화된 변환 계수들을 스캐닝한 후에, 비디오 인코더 (20) 는 1 차원 벡터를 엔트로피 코딩할 수도 있고, 비디오 인코더 (20) 는 또한, 비디오 데이터를 디코딩함에 있어서 비디오 디코더 (30) 에 의한 사용을 위해 인코딩된 비디오 데이터와 연관된 선택스 엘리먼트들을 엔트로피 인코딩할 수도 있다.
- [0053] 산술 코딩은 심볼들을 비-정수 길이 코드워드들로 맵핑할 수 있기 때문에, 높은 코딩 효율을 갖는 다수의 압축 알고리즘들에서 사용되는 엔트로피 코딩의 형태이다. CABAC 코딩은 일부 정보에 적용되고 다른 정보에 대하여 바이패스될 수도 있다.
- [0054] 전술된 것과 같이, 비디오 인코더 (20) 또는 비디오 디코더 (30) 가 모션 추정을 수행할 때, 플리커링 아티팩트들이 발생할 수도 있다. 플리커링 아티팩트들이 검출될 수 있는 프로세스가 지금부터 논의될 것이다. HEVC 에서, I-슬라이스들을 코딩할 때, 비디오 코더는 현재 슬라이스로부터 이전에 복원된 샘플들에 기초하여 예측을 수행한다. P 또는 B 슬라이스들의 인터-예측을 위해, 비디오 코더는 이전에 코딩된 픽처들 (디스플레이 순서에서 과거 또는 미래) 에 기초하여 예측을 수행할 수도 있다. 본 개시물의 기술들은 모든 슬라이스들이 I-슬라이스들이라고 가정하여 설명되지만, 본 개시물의 기술들은 비디오 시퀀스가 I, P 및 B 슬라이스들의 혼합일 경우에 적용가능할 수도 있다.
- [0055] 현재 슬라이스로부터의 예측 대 이전에 코딩된 픽처들에 기초한 예측에 있어서의 차이는 P 및 B 슬라이스들에 대한 양자화 및 예측 잔차들과 비교하여 I 슬라이스에 대한 예측 잔차 및 양자화 특징들에 매우 상이한 영향들을 미친다. 유사하게, 2 개의 연속하는 I 슬라이스들 간의 예측 잔차 및 양자화 특징들은 또한, 시간 예측의 부족으로 인해 매우 상이할 수도 있다. 2 개의 연속하는 I 슬라이스들 간의 예측 잔차 및 양자화 특징들이 상이한 것은 "플리커링" 으로 지칭되는 I 슬라이스들에 대한 시각적 아티팩트를 발생시킬 수도 있다.
- [0056] 본 개시물은 전술된 문제들을 어드레싱하기 위한 몇몇 기술들을 제안한다. 본 개시물의 기술들에 따라 구성

된 비디오 인코더 (20) 와 같은 비디오 코더는 플리커링 아티팩트들을 검출하기 위해 플리커 검출뿐만 아니라 플리커 완화를 수행할 수도 있다. 본 개시물의 일반적인 플리커 검출 기술들이 지금부터 논의될 것이다.

- [0057] 비디오 인코더 (20) 는 본 개시물의 기술들에 따라, 디스플레이 순서에서 이전 코딩된 픽처로부터 병치된 블록에 기초하여 현재 픽처 (즉, 현재 코딩되고 있는 비디오 데이터의 픽처) 로부터 블록의 플리커링을 검출하도록 구성될 수도 있다.
- [0058] 비디오 인코더 (20) 는 이전 픽처로부터 병치된 블록의 결정된 플리커링 블록의 비디오 코딩 모드가 현재 픽처의 플리커링 블록과 연관된 플리커링 아티팩트를 완화하도록 제약되거나 변경되어야만 하는 것으로 결정할 수도 있다. 이는 일반적으로, 비디오 인코더에서 달성하기에 매우 어렵다. 이전에 코딩된 병치된 블록의 비디오 코딩 모드들을 변경하는 것보다, 본 개시물의 기술들은 플리커링을 결정하고 완화하기 위해 비디오 데이터의 디스플레이 순서에서, 현재 픽처로부터 원래의 블록 및 다음 픽처로부터 병치된 원래의 블록을 사용하도록 비디오 인코더 (20) 를 구성할 수도 있다. 본 개시물은 디스플레이 순서에서 현재 및 다음 블록을 각각, $B_{current}$ 및 B_{next} 로 지칭한다.
- [0059] 비디오 인코더 (20) 는 디스플레이 순서에서 현재 프레임의 블록과 다음 프레임의 병치된 블록 간의 차이들을 결정할 수도 있다. 블록, $B_{current}$ 및 병치된 블록, B_{next} 간의 차이들에 기초하여, 비디오 인코더 (20) 는 현재 프레임의 블록, $B_{current}$ 이 플리커할 것으로 결정할 수도 있다. 비디오 인코더 (20) 는 $B_{current}$ 의 블록과 B_{next} 의 병치된 블록의 메트릭들을 결정 (예컨대, 그들 간의 차이를 계산) 할 수도 있다. 비디오 인코더 (20) 가 비교할 수도 있는 메트릭들의 예들은: 구배들 (gradients), 절대 차이들의 합 (SAD), 및 평균 제곱 에러 (MSE) 를 포함한다. 메트릭들에 기초하고, 이하 더 상세히 논의되는 것과 같이, 비디오 인코더 (20) 는 플리커 완화가 수행되어야만 한다고 결정하고, 블록, $B_{current}$, 현재 픽처가 플리커할 것임을 표시하는 정보를 코딩할 수도 있다.
- [0060] 일부 예들에서, 인코딩된 정보는 플리커링 영역을 표시하는, 플래그와 같은 선택스 엘리먼트를 포함할 수도 있다. 선택스 엘리먼트를 생성하거나 세팅하면, 비디오 인코더 (20) 또는 비디오 디코더 (30) 는 세팅되고 있는 선택스 엘리먼트에 기초하여 임의의 타입의 추가의 비디오 프로세싱을 수행할 수도 있다. 추가의 비디오 프로세싱의 예들은, 다양한 타입들의 필터링, 안티-에일리어싱, 디블록킹, 아티팩트 제거, 및/또는 본 명세서에서 명확히 설명되지 않는 임의의 다른 타입의 비디오 프로세싱을 포함할 수도 있다.
- [0061] 영역은 일부 예들에서, 현재 픽처로부터의 블록, $B_{current}$ 을 포함할 수도 있다. 블록이 플리커할 것임을 표시하는 정보는 LCU, CTU (코딩 트리 유닛), CU, PU 또는 블록과 같은 특정 영역이 플리커링 영역인 것을 표시할 수도 있다. 그러한 영역들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 플리커링 아티팩트를 완화하기 위해, 공간 인트라 예측 및 인트라 블록 카피 (인트라 BC) 와 같은 일부 코딩 모드들 및/또는 톨들을 제약할 수도 있다. 본 개시물의 기술들에 따른 플리커 완화 기술들의 추가의 예들이 지금부터 논의될 것이다.
- [0062] 일 예에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 플리커링 영역들에서 공간 인트라 예측을 디스에이블할 수도 있다. 비디오 인코더 (20) 는 공간 인트라 예측이 플리커링 영역의 공간적 이웃하는 블록들을 인트라 예측에 사용불가능한 것으로 마킹함으로써 디스에이블된다고 표시할 수도 있다. 다른 예에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 상이한 변환 사이즈들 및 결과적인 예측들에 의해 야기되는 플리커링 아티팩트가 추가로 완화되도록, 플리커링 영역 내의 블록들을 인트라 예측에 사용불가능한 것으로 마킹할 수도 있다.
- [0063] 비디오 인코더 (20) 는 블록을 인트라-예측에 사용불가능한 것으로 마킹하는 경우들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 예컨대, HEVC 의 섹션 8.4.4.2.2 에 따라 사용불가능한 샘플들을 채우기 위해 인트라 샘플 예측에 대하여 HEVC 참조 샘플 대체 프로세스를 사용할 수도 있다. 사용불가능한 샘플들을 패딩하는 것은, 비트스트림 복잡도를 희생하면서 플리커링을 완화시킬 수도 있다. 일부 다른 예들에서, 비디오 인코더 (20) 는 비-플리커링 영역들에서의 블록들만을 인트라 예측을 위해 사용가능한 것으로 마킹할 수도 있다.
- [0064] 플리커링 영역을 식별하기 위해 정보를 코딩하는 것에 관련된 다양한 기술들이 지금부터 논의될 것이다. 이전에 설명된 것과 같이, 비디오 코딩 데이터는 다수의 상이한 레벨들을 포함한다. 비트스트림의 예시적인 레벨들은: LCU, CU, PU, 또는 TU 를 포함할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 는 비트스트림의 임의의 레벨에서 플리커링 영역, 예컨대 플리커링 플래그를 표시하는 정보를 코딩할 수도 있다.

- [0065] 본 개시물의 일부 예들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 플리커하는 것으로 마킹된 블록에 대한 예측을 수행하지 않을 수도 있다. 그러한 경우에, 원래의 샘플들은 변환 또는 변환-스킵 모드를 사용하여 코딩된다. 이 경우, 및 플리커링 플래그가 CU 레벨에서 코딩되는 경우, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 CU 레벨에서 플리커링 플래그를 코딩할 수도 있고, 비디오 인코더 (20) 는 그 CU 에 대한 인트라-예측 모드를 시그널링하지 않을 수도 있다.
- [0066] 추가로, CU 가 플리커링 영역에 있을 경우, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 예컨대, CU 가 플리커링 영역에 있을 경우, CU 가 인트라 블록 카피, 팔레트 모드 및 컬러 변환과 같은 특정 툴들로 또는 특정 모드로서 코딩될 수 없도록, 추가의 제약들을 적용할 수도 있다. 그러한 제약들을 적용할 경우, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 그에 따라 관련된 CU/PU 모드/정보 및 시그널링을 조정한다. 추가로, 다양한 예들에서, 비디오 디코더 (30) 는 플리커가 식별되는 영역들에서 플리커링을 추가로 제거하기 위해 추가의 백엔드 프로세싱 사후 디코딩을 적용할 수도 있다.
- [0067] 또한 전술된 것과 같이, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 각각, CABAC 인코딩 및 CABAC 디코딩을 사용하여 플리커링 영역을 표시하는 신택스 엘리먼트들과 같은 정보를 코딩하도록 구성될 수도 있다. 본 개시물의 기술들에 따른 예들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 플리커링 플래그를 CABAC 코딩할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 는 공간적으로 이웃하는 블록들의 플리커링 플래그들의 값들을, 플리커링 플래그의 CABAC 코딩을 위한 컨텍스트들로서 사용할 수도 있다.
- [0068] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 현재 CU 위의 CU들과 연관된, 신택스 엘리먼트들과 같은 정보를 저장하도록 구성된 메모리를 포함할 수도 있는 라인 버퍼를 포함할 수도 있다. 라인 버퍼의 메모리의 사이즈를 감소시키기 위해, 현재 LCU (또는 CTU) 외부의 공간적으로 이웃하는 블록들이 컨텍스트로서 사용되기에 사용불가능한 것으로, 비디오 인코더 (20) 가 마킹할 수도 있고, 비디오 디코더 (30) 가 결정할 수도 있다.
- [0069] 일부 예들에서, 본 개시물에서 설명되는 기술들은 또한, 예컨대 예러 강인성을 증가시키기 위해, 다른 애플리케이션들에서 사용될 수도 있다. 본 개시물에서 설명되는 기술들은 일 예로서, 합성된 예측 에러들을 감소시키거나 제거할 수도 있다. 그러한 경우들에서, 플래그 신택스 엘리먼트의 명칭은 변화될 수도 있다. 그러나, 본 개시물의 일반적인 기술들이 여전히 적용된다.
- [0070] 도 2 는 본 개시물에서 설명되는 것과 같은 플리커 검출 및 완화를 위한 기술들을 사용할 수도 있는 비디오 인코더 (20) 의 일 예를 도시하는 블록 다이어그램이다. 비디오 인코더 (20) 는 예시의 목적들을 위해, 그러나 플리커 검출 및 완화를 요구할 수도 있는 다른 코딩 표준들 또는 방법들에 관한 본 개시물의 제한 없이, HEVC 코딩의 컨텍스트에서 설명될 것이다. 비디오 인코더 (20) 는 비디오 프레임들 내에서 CU들의 인트라 및 인터 코딩을 수행할 수도 있다. 인트라-코딩은 소정의 비디오 프레임 내의 비디오 데이터에서 공간 리던던시를 감소 또는 제거하기 위해 공간 예측에 의존한다. 인터-코딩은 비디오 시퀀스의 현재 프레임과 이전 에 코딩된 프레임들 간에 시간 리던던시를 감소 또는 제거하기 위해 시간 예측에 의존한다. 인트라-모드 (I 모드) 는 수개의 공간-기반 비디오 압축 모드들 중 임의의 모드를 지칭할 수도 있다. 단방향 예측 (P 모드) 또는 양방향 예측 (B 모드) 과 같은 인터-모드들은 수개의 시간-기반 비디오 압축 모드들 중 임의의 모드를 지칭할 수도 있다.
- [0071] 도 2 에 도시된 바와 같이, 비디오 데이터 메모리 (39) 는 비디오 프레임 내에서 현재 비디오 블록을 인코딩하기 위해 사용된 비디오 데이터를 수신한다. 비디오 데이터 메모리 (39) 는 (예컨대, 비디오 데이터를 저장하도록 구성된) 비디오 인코더 (20) 의 컴포넌트들에 의해 인코딩된 비디오 데이터를 저장할 수도 있다. 비디오 데이터 메모리 (39) 에 저장된 비디오 데이터는 예컨대, 비디오 소스 (18) 로부터 획득될 수도 있다. (또한 참조 픽처 버퍼 (64) 로 지칭되는) 참조 프레임 버퍼 (64) 는 (예컨대, 또한 인트라-예측 또는 인터-예측 코딩 모드들로서 지칭되는 인트라-코딩 또는 인터-코딩 모드들에서) 비디오 인코더 (20) 에 의해 비디오 데이터를 인코딩하는데 사용하기 위한 참조 비디오 데이터를 저장하는 디코딩 픽처 버퍼 (DPB) 의 일 예이다. 비디오 데이터 메모리 (39) 및 참조 프레임 버퍼 (64) 는 동기식 DRAM (SDRAM) 을 포함한 동적 랜덤 액세스 메모리 (DRAM), 자기저항성 RAM (MRAM), 저항성 RAM (RRAM), 또는 다른 타입들의 메모리 디바이스들과 같은 다양한 메모리 디바이스들 중 임의의 메모리 디바이스에 의해 형성될 수도 있다. 비디오 데이터 메모리 (39) 및 참조 프레임 버퍼 (64) 는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에서, 비디오 데이터 메모리 (39) 는 비디오 인코더 (20) 의 다른 컴포넌트들과 온-칩일 수도 있거나, 이들 컴포넌트들에 대하여 오프-칩일 수도 있다.
- [0072] 도 2 의 예에서, 비디오 인코더 (20) 는 모션 보상 유닛 (44), 모션 추정 유닛 (42), 플리커 검출 유닛 (43),

인트라-예측 프로세싱 유닛 (46), 참조 프레임 버퍼 (64), 합산기 (62), 변환 프로세싱 유닛 (52), 양자화 유닛 (54), 및 엔트로피 인코딩 유닛 (56) 을 포함한다. 도 2 에 도시된 변환 프로세싱 유닛 (52) 은, 잔차 데이터의 블록에 실제 변환 또는 변환의 조합들을 적용하는 유닛이고, 또한 CU 의 변환 유닛 (TU) 으로 지칭될 수도 있는 변환 계수들의 블록과 혼동되지 않을 것이다. 비디오 블록 복원을 위해, 비디오 인코더 (20) 는 또한 역양자화 유닛 (58), 역변환 프로세싱 유닛 (60), 및 합산기 (62) 를 포함한다. 디블로킹 (deblocking) 필터 (도 2 에 도시 안 됨) 가 또한, 블록 경계들을 필터링하여 복원된 비디오로부터 블록키니스 아티팩트들을 제거하기 위해 포함될 수도 있다. 요구된다면, 디블로킹 필터는 통상적으로 합산기 (62) 의 출력을 필터링할 것이다.

[0073] 인코딩 프로세스 동안, 비디오 인코더 (20) 는 코딩될 비디오 프레임 또는 슬라이스를 수신한다. 프레임 또는 슬라이스는 다수의 비디오 블록들, 예컨대 최대 코딩 유닛들 (LCU들) 로 분할될 수도 있다. 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 시간 압축을 제공하기 위해 하나 이상의 레퍼런스 프레임들에 있어서의 하나 이상의 블록들에 대해 수신된 비디오 블록의 인트라-예측 코딩을 수행한다. 인트라-예측 프로세싱 유닛 (46) 은 공간 압축을 제공하기 위해 코딩될 블록과 동일한 프레임 또는 슬라이스에 있는 하나 이상의 이웃하는 블록들에 대하여, 수신된 비디오 블록의 인트라-예측 코딩을 수행할 수도 있다.

[0074] 모드 선택 유닛 (40) 은, 예를 들어, 각각의 모드에 대한 에러 (즉, 왜곡) 결과들에 기초하여 코딩 모드들 중 하나, 즉, 인트라 코딩 모드 또는 인터 코딩 모드를 선택할 수도 있고, 결과적인 인트라-예측된 또는 인터-예측된 블록 (예컨대, 예측 유닛 (PU)) 을 합산기 (50) 에 제공하여 잔차 블록 데이터를 생성하고, 합산기 (62) 에 제공하여 참조 프레임에서의 사용을 위한 인코딩된 블록을 복원한다. 합산기 (62) 는 이하 더 상세히 설명되는 것과 같이, 인코딩된 블록을 복원하기 위해 블록에 대한 역변환 프로세싱 유닛 (60) 으로부터 역양자화된, 역변환된 데이터와 예측된 블록을 결합한다. 일부 비디오 프레임들은 I-프레임들로서 지정될 수도 있으며, 여기서 I-프레임에서의 모든 블록들은 인트라-예측 모드에서 인코딩된다. 일부 경우들에서, 인트라-예측 프로세싱 유닛 (46) 은 예컨대, 모션 추정 유닛 (42) 에 의해 수행되는 모션 탐색이 그 블록의 충분한 예측을 발생하지 않을 경우, P-프레임 또는 B-프레임에서 블록의 인트라-예측 인코딩을 수행할 수도 있다.

[0075] 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 고도로 통합될 수도 있지만, 개념적인 목적을 위해 별개로 도시된다. 모션 추정 (또는 모션 탐색) 은, 비디오 블록들에 대한 모션을 추정하는 모션 벡터들을 생성하는 프로세스이다. 모션 벡터는, 예를 들어, 참조 프레임의 참조 샘플에 대한 현재 프레임에서의 예측 유닛의 변위를 표시할 수도 있다. 모션 추정 유닛 (42) 은 인터-코딩된 프레임의 예측 유닛에 대한 모션 벡터를, 예측 유닛을 참조 프레임 버퍼 (64) 에 저장된 참조 프레임의 참조 샘플들과 비교함으로써 계산한다. 참조 샘플은 픽셀 차이의 관점에서 코딩되고 있는 PU 를 포함하는 CU 의 부분에 밀접하게 매칭하는 것으로 발견되는 블록일 수도 있고, 이 픽셀 차이는 절대 차이의 합 (SAD), 제곱 차이의 합 (SSD), 또는 다른 상이한 메트릭들에 의해 결정될 수도 있다. 참조 샘플은 참조 프레임 또는 참조 슬라이스 내의 어느 곳에서도 발생할 수도 있지만, 반드시 참조 프레임 또는 슬라이스의 블록 (예컨대, 코딩 유닛) 경계에서 발생하는 것은 아니다. 일부 예들에서, 참조 샘플은 단편적인 픽셀 위치에서 발생할 수도 있다.

[0076] 모션 추정 유닛 (42) 은 계산된 모션 벡터를 엔트로피 인코딩 유닛 (56) 및 모션 보상 유닛 (44) 으로 전송한다. 모션 벡터에 의해 식별되는 참조 프레임의 부분은 참조 샘플로서 지칭될 수도 있다. 모션 보상 유닛 (44) 은 예컨대, PU 에 대하여 모션 벡터에 의해 식별된 참조 샘플을 취출함으로써, 현재 CU 의 예측 유닛에 대한 예측 값을 계산할 수도 있다.

[0077] 인트라-예측 프로세싱 유닛 (46) 은 수신된 블록을, 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 수행된 인터-예측에 대한 대안으로서 인트라-예측할 수도 있다. 인트라-예측 프로세싱 유닛 (46) 은 블록들에 대하여 좌-우, 상-하로의 인코딩 순서를 가정할 때, 이웃하는 이전에 코딩된 블록들, 예컨대 현재 블록의 상위, 상위 우측, 상위 좌측, 또는 좌측의 블록들에 대하여 수신된 블록을 예측할 수도 있다. 인트라-예측 프로세싱 유닛 (46) 은 다양한 상이한 인트라-예측 모드들로 구성될 수도 있다. 예를 들어, 인트라-예측 프로세싱 유닛 (46) 은 인코딩되고 있는 CU 의 사이즈에 기초하여, 특정 수의 방향성 예측 모드들, 예를 들어, 33 개의 방향성 예측 모드들로 구성될 수도 있다.

[0078] 인트라-예측 프로세싱 유닛 (46) 은 예컨대, 다양한 인트라-예측 모드들에 대한 에러 값들을 계산하고 최저 에러 값을 산출하는 모드를 선택함으로써, 인트라-예측 모드를 선택할 수도 있다. 방향성 예측 모드들은 공간적으로 이웃하는 픽셀들의 값들을 결합하고, 결합된 값들을 PU 에서의 하나 이상의 픽셀 포지션들에 적용하기 위한 함수들을 포함할 수도 있다. PU 에서의 모든 픽셀 포지션들에 대한 값들이 계산되었으면, 인트라-예측

프로세싱 유닛 (46) 은 PU 와 인코딩될 수신된 블록 간의 픽셀 차이들에 기초하여 예측 모드에 대한 에러 값을 계산할 수도 있다. 인트라-예측 프로세싱 유닛 (46) 은 허용가능한 에러 값을 산출하는 인트라-예측 모드가 발견될 때까지 인트라-예측 모드들을 테스트하는 것을 계속할 수도 있다. 그 후에, 인트라-예측 프로세싱 유닛 (46) 은 PU 를 합산기 (50) 에 전송할 수도 있다.

[0079] 비디오 인코더 (20) 는 모드 보상 유닛 (44) 또는 인트라-예측 프로세싱 유닛 (46) 에 의해 계산된 예측 데이터를, 코딩되는 오리지널 비디오 블록으로부터 감산함으로써 잔차 블록을 형성한다. 합산기 (50) 는 이러한 감산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 잔차 블록은 픽셀 차이 값들의 2 차원 행렬에 대응할 수도 있고, 여기서 잔차 블록에서 값들의 수는 그 잔차 블록에 대응하는 PU 에서의 픽셀들의 수와 동일하다. 잔차 블록에서의 값들은 코딩될 원래의 블록에서 및 PU 에서의 병치된 픽셀들의 값들 간의 차이들, 즉 에러에 대응할 수도 있다. 차이들은 코딩되는 블록의 타입에 의존하여, 크로마 또는 루마 차이들일 수도 있다.

[0080] 변환 프로세싱 유닛 (52) 은 잔차 블록으로부터 하나 이상의 변환 유닛들 (TU들) 을 형성할 수도 있다. 변환 프로세싱 유닛 (52) 은 복수의 변환들 중에서부터 변환을 선택한다. 변환은 블록 사이즈, 코딩 모드, 등등과 같은 하나 이상의 코딩 특징들에 기초하여 선택될 수도 있다. 그 후, 변환 프로세싱 유닛 (52) 은 선택된 변환을 TU 에 적용하여, 변환 계수들의 2 차원 어레이를 포함하는 비디오 블록을 생성한다.

[0081] 변환 프로세싱 유닛 (52) 은 결과적인 변환 계수들을 양자화 유닛 (54) 으로 전송할 수도 있다. 그 후, 양자화 유닛 (54) 은 변환 계수들을 양자화할 수도 있다. 그 후, 엔트로피 인코딩 유닛 (56) 은 스캐닝 모드에 따라 행렬에서 양자화된 변환 계수들의 스캔을 수행할 수도 있다. 본 개시물은 엔트로피 인코딩 유닛 (56) 을 스캔을 수행하는 것으로 설명한다. 그러나, 다른 예들에서, 양자화 유닛 (54) 과 같은 다른 프로세싱 유닛들이 스캔을 수행할 수 있는 것이 이해되어야 한다.

[0082] 변환 계수들이 1 차원 어레이로 스캐닝되면, 엔트로피 인코딩 유닛 (56) 은 (CABAC), 신택스 기반 컨텍스트 적응형 이진 산술 코딩 (SBAC), 확률 간격 파티셔닝 엔트로피 (PIPE), 또는 다른 엔트로피 코딩 방법과 같은 엔트로피 코딩을 계수들에 적용할 수도 있다. 엔트로피 인코딩 유닛 (56) 은 변환 계수들을 코딩하도록 구성될 수도 있다. CABAC 를 수행하기 위해, 엔트로피 인코딩 유닛 (56) 은 송신될 심볼들을 인코딩하기 위해 특정 컨텍스트에 적용할 컨텍스트 모델을 선택할 수도 있다. 엔트로피 인코딩 유닛 (56) 에 의한 엔트로피 코딩 이후, 결과적인 인코딩된 비디오는 다른 디바이스, 예컨대 비디오 디코더 (30) 로 송신되거나 또는 더 나중의 송신 또는 취출을 위해 아카이브될 수도 있다.

[0083] 일부 케이스들에서, 엔트로피 인코딩 유닛 (56) 또는 비디오 인코더 (20) 의 다른 유닛은 엔트로피 코딩에 추가로, 다른 코딩 기능들을 수행하도록 구성될 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (56) 은 CU들 및 PU들에 대한 코딩 블록 패턴 (CBP) 을 결정하도록 구성될 수도 있다. 또한, 일부 케이스들에서, 엔트로피 인코딩 유닛 (56) 은 계수들의 런 렱스 (run length) 코딩을 수행할 수도 있다.

[0084] 역양자화 유닛 (58) 및 역변환 프로세싱 유닛 (60) 은, 예를 들어, 참조 블록으로서의 추후 사용을 위해 픽셀 도메인에서 잔차 블록을 복원하도록, 각각, 역양자화 및 역변환을 적용한다. 모션 보상 유닛 (44) 은 참조 프레임 버퍼 (64) 의 프레임들 중 하나의 예측 블록에 잔차 블록을 가산함으로써 참조 블록을 계산할 수도 있다. 모션 보상 유닛 (44) 은 또한, 하나 이상의 보간 필터들을 복원된 잔차 블록에 적용하여, 모션 추정에서 사용하기 위한 서브-정수 픽셀 값들을 계산할 수도 있다. 합산기 (62) 는 복원된 잔차 블록을, 모션 보상 유닛 (44) 에 의해 생성된 모션 보상된 예측 블록에 가산하여, 참조 프레임 버퍼 (64) 에서의 저장을 위한 복원된 비디오 블록을 생성한다. 복원된 비디오 블록은, 후속 비디오 프레임에서의 블록을 인터-코딩하기 위해 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 참조 블록으로서 사용될 수도 있다.

[0085] 플리커 검출 유닛 (43) 은 본 개시물의 임의의 기술들에 따라 플리커 검출 및/또는 완화 기술들을 수행하도록 구성된다. 플리커 검출 유닛 (43) 은 참조 프레임 버퍼 (64) 로부터 또는 비디오 데이터 메모리 (39) 로부터 디스플레이 순서로 현재 픽처 및 다음 픽처를 액세스할 수도 있다. 플리커 검출 유닛 (43) 은 또한, 모드 선택 유닛 (40) 을 시그널링함으로써 특정 비디오 코딩 모드 또는 코딩 툴들을 표시 또는 제한하도록 구성될 수도 있다.

[0086] 플리커 검출 유닛 (43) 은 또한, 신택스 엘리먼트들과 같은 정보를 엔트로피 인코딩 유닛 (56) 에 CABAC 인코딩하기 위해 엔트로피 인코딩 유닛 (56) 을 시그널링하도록 구성될 수도 있다. 그러한 정보는 특정 지역이 플리커할 것인지의 여부를 표시하는 플래그 신택스 엘리먼트를 일부 예들로서 포함할 수도 있다. 다른 예들에

서, 플리커 검출 유닛 (43) 은 선택스 엘리먼트들의 엔트로피 인코딩 유닛 (45) 으로의 CABAC 인코딩을 위한 컨텍스트 정보를 시그널링할 수도 있다.

[0087] 전술된 것과 같이, 플리커 검출 유닛 (43) 은 $B_{current}$ 의 블록이 플리커할 수도 있는 것을 결정하기 위해 상이한 메트릭들을 사용할 수도 있다. 비디오 인코더 (20) 가 사용할 수도 있는 플리커 검출 기술들의 예들이 지금부터 더 상세하게 설명될 것이다.

[0088] 일부 예들에서, 플리커링 아티팩트를 검출하기 위해, 플리커 검출 유닛 (43) 은 $B_{current}$ 의 각각의 루미넌스 샘플에 대한 구배 (에지 정보) 를 계산할 수도 있다. 현재 픽처의 샘플들이 RGB (적색, 녹색, 청색) 포맷이라면, 플리커 검출 유닛 (43) 은 픽처를 YCbCr 포맷으로 컨버팅하고, 그 픽처를 비디오 데이터 메모리 (39) 에 저장할 수도 있다. 대안적으로, 플리커 검출 유닛 (43) 은 루미넌스 성분에 대한 프록시로서 G 성분을 사용할 수도 있다. 일부 예들에서, 플리커 검출 유닛 (43) 은 구배를 결정하기 위해 Sobel 에지 검출기를 사용할 수도 있다. 그러나, 플리커 검출 유닛 (43) 은 구배를 결정하기 위해 임의의 다른 구배 계산기를 사용할 수도 있다. 구배 크기들에 기초하여, 플리커 검출 유닛 (43) 은 블록에 대하여 구배 크기의 평균 (avgGradMag) 및 구배 크기의 표준 편차 (stdDevGradMag) 를 계산할 수도 있다.

[0089] 추가로, 플리커 검출 유닛 (43) 은 소정의 임계치 (EDGE_THRESHOLD) 이상의 구배 크기를 갖는 블록에서의 샘플들의 퍼센티지를 계산할 수도 있다. 임계치 이상의 샘플들의 퍼센티지는 상기 예들에서 edgePixelPercentage 로 표시된다. 그 후, 플리커 검출 유닛 (43) 은 임계치 이상 또는 이하의 샘플들의 퍼센티지에 기초하여, 블록을 에지 블록 또는 텍스처 블록으로서 분류할 수도 있다.

[0090] 플리커 검출 유닛 (43) 은 edgePixelPercentage 가 임계치 (EDGE_PERCENT_THRESHOLD) 이상일 경우, 블록을 에지 블록으로서 분류한다. 플리커 검출 유닛 (43) 은 avgGradMag 가 임계치 (TEXTURE_LOW_THRESHOLD) 보다 크고, stdDevGradMag 가 임계치 (TEXTURE_HIGH_THRESHOLD) 미만이고, stdDevGradMag 와 avgGradMag 간의 비율이 임계치 (RATIO_TEXTURE_THRESHOLD) 미만일 경우, 블록을 텍스처 블록으로서 분류한다.

[0091] 플리커 검출 유닛 (43) 은 또한, 블록에 대한 팔레트 사이즈를 결정할 수도 있고, 본 개시물의 기술들에 따른 플리커 결정 프로세스의 부분으로서 팔레트 사이즈를 사용할 수도 있다. 블록에 대한 팔레트 사이즈는 블록에서의 별개의 픽셀 값들의 수로서 정의된다. 플리커 검출 유닛 (43) 은 픽셀 값들이 별개인지 여부를 결정할 때, 픽셀의 모든 성분들 (예컨대, RGB 또는 YUV) 을 고려할 수도 있다. 팔레트에서 픽셀 값으로부터의 약간의 편차가 가능할 수도 있다. 예를 들면, 2 개의 픽셀들이 개별 컴포넌트들이 임계값 미만으로 다르거나 절대 차이들의 총 합이 다른 임계값 미만으로 다르도록 한다면, 플리커 검출 유닛 (43) 은 상기 픽셀들을 동일하고 팔레트에서의 단일 엔트리에 대응하는 것으로 고려할 수도 있다. 기준은 개별 컴포넌트 차이들에 관한 임계치뿐만 아니라 절대 차이들의 총 합에 관한 임계치 양자를 포함한다.

[0092] 절대 차이들의 합, 및 팔레트 사이즈에 기초하여 플리커링 플래그를 인코딩하기 위한 프로세스가 지금부터 설명될 것이다. 제 1 단계에서, 플리커 검출 유닛 (43) 은 원래의 블록들 $B_{current}$ 및 B_{next} 간의 SAD 를 계산할 수도 있다. SAD 가 임계치 (SAD_THRESHOLD) 보다 크다면, 플리커 검출 유닛 (43) 은 블록을 플리커하지 않는 것으로 마킹한다.

[0093] 현재 블록까지의 현재 픽처에 대한 누적 SAD 가 임계치 (CUM_SAD_THRESHOLD) 미만이라면, 플리커 검출 유닛 (43) 은 블록을 플리커하지 않는 것으로 마킹한다. SAD 가 SAD_THRESHOLD 미만이고: (1) 팔레트 사이즈가 임계 팔레트 사이즈 (PALETTE_THRESHOLD) 보다 크고, 블록이 에지 블록이거나; 또는 (2) 블록이 텍스처 블록이라면, 플리커 검출 유닛 (43) 은 블록을 플리커링 블록으로 마킹한다. 그렇지 않으면, 즉 상기 조건들이 충족되지 않으면, 모션 추정 유닛 (42) 이 블록을 비-플리커링 블록으로 마킹한다.

[0094] 상기 예들에서, 플리커 검출 유닛 (43) 은 SAD 대신 MSE 와 같은 왜곡의 다른 측정치들을 사용할 수도 있다. 일부 케이스들에서, 모션 추정 유닛 (42) 은 SAD 또는 다른 왜곡 측정치들을 예들로서, 루마 또는 크로마 값들, RGB, 또는 YUV 값들과 같은 픽셀 값들에 적용할 수도 있다. 플리커 검출을 위한 앞의 기술들이 블록이 에지 블록인지 텍스처 블록인지 여부를 결정하기 위해 팔레트 사이즈 및 측정치들에 기초한 기준들을 사용하지만, 상기 기준들 중 오직 일부만을 사용하는 것이 가능하다. 플리커 검출 유닛 (43) 은 또한, 구배 또는 다른 기준들에 기초하여 블록이 텍스처 블록인지 에지 블록인지 여부를 결정하기 위해 다른 측정치들을 사용할 수도 있다.

[0095] 추가로, 플리커 검출 유닛 (43) 은 플리커링을 완화하기 위해 추가의 제약들을 적용할 수도 있다. 일 예에

서, 비디오 인코더 (20) 및 특히 플리커 검출 유닛 (43)은 플리커링 지역에 대하여 RDOQ (레이트-왜곡 최적화된 양자화)를 디스에이블할 수도 있다.

[0096] 플리커 완화의 다른 예들에서, 플리커 검출 유닛 (43)은 플리커 지역의 인코딩을 바이패스하는 것을 수행하기 위해 엔트로피 인코딩 유닛 (56)을 시그널링할 수도 있고, 이전 I-프레임에서의 동일한 영역의 정보, 예컨대 블록 모드, 변환 사이즈, 및 계수 레벨들을 인코딩하여 현재 프레임에서의 플리커링 영역의 복원이 이전 I-프레임에서와 정확히 동일하도록 할 수도 있다.

[0097] 플리커 검출 유닛 (43)은 또한, 플리커 완화를 수행하기 위해 CABAC 인코딩 프로세스들을 수정하도록 구성될 수도 있다. 일 예로서, 플리커 검출 유닛 (43)은 플리커링 영역에 대하여 계수 코딩에 관련된 컨텍스트들과 같은 특정 CABAC 컨텍스트 모델들의 상태를 리셋하도록 엔트로피 인코딩 유닛 (56)을 시그널링할 수도 있다. 도 3에 도시된 비디오 디코더 (30)의 플리커 검출 유닛 (73) 및 엔트로피 디코딩 유닛 (70)은 CABAC 디코딩 동안 CABAC 컨텍스트 모델들을 리셋하는 일반적으로 상호적인 프로세스를 수행하도록 구성될 수도 있다.

[0098] CABAC를 수반하는 플리커 완화의 일부 예들에서, 플리커 검출 유닛 (43)은 현재 픽처의 블록에 대한 좌측의 공간적으로 이웃하는 블록의 플리커링 플래그를 CABAC 코딩에 대한 오직 컨텍스트로서 사용할 수도 있도록 엔트로피 인코딩 유닛 (56)을 시그널링할 수도 있다. 비디오 디코더 (30)의 플리커 검출 유닛 (73) 및 엔트로피 디코딩 유닛 (70)은 CABAC 디코딩을 위한 일반적으로 상호적인 프로세스를 수행할 수도 있다. 일부 예안들에 있어서, 플리커링 플래그에 대한 컨텍스트는 비스트림에서 시그널링된 다른 정보를 포함할 수도 있다.

CABAC 컨텍스트로서 사용된 다른 정보의 예들은 CU 깊이, 블록 타입, 변환 사이즈, 또는 QP (양자화 파라미터)와 같은 값들을 포함할 수도 있다.

[0099] 본 개시물의 기술들에 따라서, 비디오 인코더 (20) 및 비디오 디코더 (30)는: 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하도록 구성될 수도 있으며, 여기서 현재 픽처로부터의 블록이 플리커할 것인지 여부를 결정은 디스플레이 순서에서 현재 픽처에서의 블록 및 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초한다.

[0100] 도 3은 인코딩된 비디오 시퀀스를 디코딩하는 비디오 디코더 (30)의 일 예를 예시하는 블록 다이어그램이다. 비디오 디코더 (30)는 플리커 검출 및 완화를 위한 방법들을 수행하도록 구성된 디바이스의 일 예를 나타낸다. 도 3의 예에서, 비디오 데이터 메모리 (69)는 인코딩된 비디오를 수신한다. 비디오 데이터 메모리 (69)는 비디오 디코더 (30)의 컴포넌트들에 의해 디코딩될, 인코딩된 비디오 비트스트림과 같은 비디오 데이터를 저장할 수도 있다 (예컨대, 비디오 데이터를 저장하도록 구성될 수도 있다). 비디오 데이터 메모리 (69)에 저장된 비디오 데이터는 비디오 데이터의 유선 또는 무선 네트워크 통신을 통해 또는 물리적인 데이터 저장 매체에 액세스함으로써, 카메라와 같은 로컬 비디오 소스로부터 획득될 수도 있다. 비디오 데이터 메모리 (69)는 인코딩된 비디오 비트스트림으로부터 인코딩된 비디오 데이터를 저장하는 코딩된 픽처 버퍼 (CPB)를 형성할 수도 있다.

[0101] 참조 픽처 메모리 (82)는 (예를 들어, 인트라 또는 인터 코딩 모드들에서) 비디오 디코더 (30)에 의해 비디오 데이터를 디코딩함에 있어서 사용하기 위한 참조 비디오 데이터를 저장하는 디코딩된 픽처 버퍼 (DPB)의 일 예이다. 비디오 데이터 메모리 (69) 및 레퍼런스 픽처 메모리 (82)는 동기식 DRAM (SDRAM)을 포함한 동적 랜덤 액세스 메모리 (DRAM), 자기저항성 RAM (MRAM), 저항성 RAM (RRAM), 또는 다른 타입들의 메모리 디바이스들과 같은 다양한 메모리 디바이스들 중 임의의 메모리 디바이스에 의해 형성될 수도 있다. 비디오 데이터 메모리 (69) 및 레퍼런스 픽처 메모리 (82)는 동일한 메모리 디바이스 또는 별도의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에서, 비디오 데이터 메모리 (69)는 비디오 디코더 (30)의 다른 컴포넌트들과 온-칩일 수도 있거나, 이들 컴포넌트들에 대하여 오프-칩일 수도 있다.

[0102] 비디오 디코더 (30)는 엔트로피 디코딩 유닛 (70), 모션 보상 유닛 (72), 플리커 검출 유닛 (73), 인트라-예측 프로세싱 유닛 (74), 역양자화 유닛 (76), 역변환 프로세싱 유닛 (78), 참조 프레임 버퍼 (82) 및 합산기 (80)를 포함한다. 일부 예들에 있어서, 도 3의 비디오 디코더 (30)는 비디오 인코더 (20; 도 2 참조)에 대하여 설명된 인코딩 패스에 일반적으로 상호적인 디코딩 패스를 수행할 수도 있다.

[0103] 엔트로피 디코딩 유닛 (70)은 변환 계수들의 1차원 어레이를 취출하기 위해 인코딩된 비트 스트림에 엔트로피 디코딩 프로세스를 수행할 수도 있다. 사용된 엔트로피 디코딩 프로세스는 비디오 인코더 (20)에 의해 사용된 엔트로피 코딩 (예컨대, CABAC)에 의존한다. 인코더에 의해 사용된 엔트로피 코딩 프로세스는 인코딩

된 비트스트림에서 시그널링될 수도 있거나, 미리 결정된 프로세스일 수도 있다. 엔트로피 디코딩 유닛 (70) 은 변환 계수들을 코딩하도록 구성될 수도 있다.

[0104] 일부 예들에 있어서, 엔트로피 디코딩 유닛 (70) 또는 역양자화 유닛 (76) 은 비디오 인코더 (20) 의 엔트로피 인코딩 유닛 (56) (또는 양자화 유닛 (54)) 에 의해 사용된 스캐닝 모드를 미리링하는 스캔을 사용하여 수신된 값들을 스캐닝할 수도 있다. 계수들의 스캐닝이 역양자화 유닛 (76) 에서 수행될 수도 있지만, 스캐닝은 예시의 목적들을 위해 엔트로피 디코딩 유닛 (70) 에 의해 수행되는 것으로 설명될 것이다. 추가로, 예시의 용이함을 위해 분리된 기능적 유닛들로서 도시되지만, 비디오 디코더 (30) 의 엔트로피 디코딩 유닛 (70), 역양자화 유닛 (76), 및 다른 유닛들의 구조와 기능성은 서로 고도로 통합될 수도 있다.

[0105] 역양자화 유닛 (76) 은 비트스트림에서 제공되고 엔트로피 디코딩 유닛 (70) 에 의해 디코딩되는 양자화된 변환 계수들을 역양자화, 즉, 탈양자화한다. 역양자화 프로세스는 예컨대, HEVC 에 대하여 제안되거나 H.264 디코딩 표준에 의해 정의된 프로세스들과 유사한, 종래의 프로세스를 포함할 수도 있다. 역양자화 프로세스는 양자화의 정도 및, 유사하게 적용되어야 하는 역양자화의 정도를 결정하기 위해 CU 에 대하여 비디오 인코더 (20) 에 의해 계산된 양자화 파라미터 (QP) 의 이용을 포함할 수도 있다. 역양자화 유닛 (76) 은 계수들이 1 차원 어레이로부터 2 차원 어레이로 컨버팅되기 이전 또는 이후에 변환 계수들을 역양자화할 수도 있다.

[0106] 역변환 프로세싱 유닛 (78) 은 역양자화된 변환 계수들에 역변환을 적용한다. 일부 예들에서, 역변환 프로세싱 유닛 (78) 은 비디오 인코더 (20) 로부터의 시그널링에 기초하여, 또는 블록 사이즈, 코딩 모드, 등등과 같은 하나 이상의 코딩 특징들로부터 변환을 추론함으로써 역변환을 결정할 수도 있다. 일부 예들에서, 역변환 프로세싱 유닛 (78) 은 현재 블록을 포함하는 LCU 에 대한 쿼드 트리의 루트 노드에서 시그널링된 변환에 기초하여 현재 블록에 적용하기 위한 변환을 결정할 수도 있다. 대안적으로, 변환은 LCU 쿼드트리에서 리프-노드 CU 에 대한 TU 쿼드트리의 루트에서 시그널링될 수도 있다. 일부 예들에서, 역변환 프로세싱 유닛 (78) 은 캐스케이드형 역변환을 적용할 수도 있으며, 여기서 역변환 프로세싱 유닛 (78) 은 2 이상의 역변환 변환들을 디코딩되고 있는 현재 블록의 변환 계수들에 적용한다. 인트라-예측 프로세싱 유닛 (74) 은 현재 프레임의 이전에 디코딩된 블록들로부터의 데이터 및 시그널링된 인트라-예측 모드에 기초하여 현재 프레임의 현재 블록에 대한 예측 데이터를 생성할 수도 있다.

[0107] 모션 보상 유닛 (72) 은 인코딩된 비트스트림으로부터 모션 벡터, 모션 예측 방향 및 참조 인덱스를 추출할 수도 있다. 참조 예측 방향은 이너-예측 모드가 단방향성인지 (예컨대, P 프레임) 또는 양방향성인지 (B 프레임) 의 여부를 표시한다. 참조 인덱스는 후보 모션 벡터가 어떤 참조 프레임에 기초하는지를 표시한다.

[0108] 추출된 모션 예측 방향, 참조 프레임 인덱스, 및 모션 벡터에 기초하여, 모션 보상 유닛은 현재 부분에 대한 모션 보상된 블록을 생성한다. 이들 모션 보상된 블록들은 본질적으로 잔차 데이터를 생성하는데 사용된 예측 블록을 재생성한다.

[0109] 모션 보상 유닛 (72) 은 가능하게는 보간 필터들에 기초하여 보간을 수행하는, 모션 보상된 블록들을 생성할 수도 있다. 서브-샘플 정확도로 모션 추정을 위해 사용될 보간 필터들에 대한 식별자들은 신텍스 엘리먼트들에 포함될 수도 있다. 모션 보상 유닛 (72) 은 비디오 블록의 인코딩 동안 비디오 인코더 (20) 에 의해 사용된 바와 같은 보간 필터들을 이용하여, 참조 블록의 서브-정수 픽셀들에 대한 보간된 값들을 계산할 수도 있다. 모션 보상 유닛 (72) 은 수신된 신텍스 정보에 따라 비디오 인코더 (20) 에 의해 사용된 보간 필터들을 결정하고, 그 보간 필터들을 사용하여 예측 블록을 생성할 수도 있다.

[0110] 추가로, HEVC 예에서, 모션 보상 유닛 (72) 및 인트라-예측 프로세싱 유닛 (74) 은 인코딩된 비디오 시퀀스의 프레임(들)을 인코딩하는데 사용된 LCU들의 사이즈들을 결정하기 위해 (예컨대, 쿼드트리에 의해 제공된) 신텍스 정보의 일부를 사용할 수도 있다. 모션 보상 유닛 (72) 및 인트라-예측 프로세싱 유닛 (74) 은 또한, 인코딩된 비디오 시퀀스의 프레임의 각각의 CU 가 어떻게 분할되는지 (및 유사하게, 서브-CU들이 어떻게 분할되는지) 를 설명하는 분할 정보를 결정하기 위해 신텍스 정보를 사용할 수도 있다. 신텍스 정보는 또한, 각각의 분할이 어떻게 인코딩되는지를 표시하는 모드들 (예컨대, 인트라- 또는 인터-예측, 및 인트라-예측에 대하여 인트라-예측 인코딩 모드), 각각의 인터-인코딩된 PU 에 대한 하나 이상의 참조 프레임들 (및/또는 참조 프레임들에 대한 식별자들을 포함하는 참조 리스트들), 및 인코딩된 비디오 시퀀스를 디코딩하기 위한 다른 정보를 포함할 수도 있다.

[0111] 합산기 (80) 는 잔차 블록들을 모션 보상 유닛 (72) 또는 인트라-예측 프로세싱 유닛 (74) 에 의해 생성된 대응하는 예측 블록들과 결합하여 디코딩된 블록들을 형성한다. 요구된다면, 디블로킹 필터가 또한, 블록키스

아티팩트들을 제거하기 위해, 디코딩된 블록들을 필터링하는데 적용될 수도 있다. 그 후, 디코딩된 비디오 블록들이 (또한 디코딩된 픽처 버퍼로 지칭되는) 참조 프레임 버퍼 (82) 에 저장되며, 그 버퍼 (82) 는 후속 모션 보상을 위한 참조 블록들을 제공하고, 또한 (도 1 의 디스플레이 디바이스 (32) 와 같은) 디스플레이 디바이스 상의 프리젠테이션을 위해 디코딩된 비디오를 생성한다.

[0112] 플리커 검출 유닛 (73) 은 본 개시물의 기술들에 따라 플리커링 검출 및/또는 플리커링 완화 기술들의 임의의 조합을 수행하도록 구성된다. 플리커 검출 유닛 (73) 은 참조 프레임 버퍼 (82) 로부터 또는 비디오 데이터 메모리 (69) 로부터 디스플레이 순서에서 현재 픽처 및 다음 픽처를 액세스할 수도 있다. 플리커 검출 유닛 (73) 은 또한, 모션 보상 유닛 (72) 및/또는 인트라-예측 프로세싱 유닛 (74) 을 시그널링함으로써 특정 비디오 코딩 모드 또는 코딩 툴들을 표시 또는 제한하도록 구성될 수도 있다.

[0113] 플리커 검출 유닛 (73) 은 또한, 선택스 엘리먼트들과 같은 정보를 엔트로피 디코딩 유닛 (70) 에 CABAC 디코딩하도록 엔트로피 디코딩 유닛 (70) 을 시그널링하도록 구성될 수도 있다. 그러한 정보는 특정 지역이 플리커할 것인지의 여부를 표시하는 플래그 선택스 엘리먼트를 일부 예들로서 포함할 수도 있다. 다른 예들에서, 플리커 검출 유닛 (73) 은 선택스 엘리먼트들의 엔트로피 디코딩 유닛 (70) 으로의 CABAC 디코딩을 위한 컨텍스트 정보를 시그널링할 수도 있다.

[0114] 이전에 설명된 것과 같이, 비디오 디코더 (30) 는 다수의 플리커링 완화 기술들을 적용할 수도 있다. 일 예로서, 플리커 검출 유닛 (73) 은 오직, 16x16 또는 그 이상과 같은 특정의 최소 사이즈로 CU들에 대한 플리커링 플래그를 디코딩할 수도 있다. 이 경우에, 16x16 보다 작은 CU들, 예컨대 사이즈 8x8 의 CU들에 대하여, 플리커 검출 유닛 (73) 은 플리커링 플래그를 디코딩하지 않을 수도 있다. 최소 사이즈 미만의 CU들에 대하여, 비디오 디코더 (30), 및 특히 플리커 검출 유닛 (73) 및 엔트로피 디코딩 유닛 (70) 은 플리커링 플래그가 거짓인 (즉, 제로와 동일한) 것을 추론할 수도 있다. 플리커링 플래그의 값을 시그널링하는 것보다, 추론하는 것은 일부 경우들에서 비트스트림 복잡도를 감소시키고 및/또는 비트들을 절약한다. 추가로, 비디오 인코더 (20) 는 플리커 완화가 비디오 데이터의 특정 레벨에 대하여 인에이블되는지 여부, 예컨대 플리커 완화가 픽처, 슬라이스, 또는 시퀀스에 대하여 인에이블되는지 여부를 표시하기 위해 시퀀스, 픽처, 또는 슬라이스 레벨에서 하이-레벨 제어 플래그를 인코딩할 수도 있다. 이 경우에 있어서, 플리커 검출 유닛 (73) 은 유사하게, 플리커링 플래그 CU들, PU들, TU들, 블록들 또는 비트스트림의 다른 엘리먼트들의 값을 추론할 수도 있다.

[0115] 도 2 에 대하여 앞서 설명된 예들은, 비디오 인코더 (20) 가 플리커링 플래그의 값을 결정하고 인코딩할 수도 있는 것을 설명하였다. 그러나, 비디오 디코더 (30) 는 또한, 플리커링 플래그의 값을 결정할 수도 있다. 예를 들어, 플리커 검출 유닛 (73) 은 블록이 특정 조건들 하에서 모든 블록 사이즈에 대하여 플리커할 것인지 여부를 결정하기 위해, 플리커 검출 프로세스를 수행할 수도 있다. 예를 들어, 비디오 인코더 (20) 가 CU 레벨에서 플리커링 플래그를 시그널링한다면, 플리커 검출 유닛 (73) 은 CU 의 각각의 블록에 대한 플리커링 플래그의 값을 결정하기 위해, 64x64 로부터 8x8 까지 각각의 CU 사이즈에 플리커링 검출을 수행할 수도 있다.

[0116] 다른 예들에서, 플리커 검출 유닛 (73) 은 또한, 각각의 16x16 블록이 플리커하고 있는지 여부를 결정할 수도 있다. 32x32 및 64x64 블록 사이즈들과 같은 더 큰 블록 사이즈에 대하여, 더 큰 블록 내의 16x16 블록들의 임의의 블록이 비디오 인코더 (20) 또는 비디오 디코더 (30) 에 의해 플리커하는 것으로 마킹된다면, 플리커 검출 유닛 (73) 은 더 큰 포괄하는 블록을 또한 플리커하고 있는 것으로 고려할 수도 있다. 다른 예에서, 플리커하고 있는 더 큰 포괄하는 블록 내의 16x16 블록들의 수가 임계치 이상일 경우에만, 플리커 검출 유닛 (73) 이 32x32 또는 64x64 블록을 플리커하는 것으로 마킹하는 것이 가능할 수도 있다.

[0117] 다른 부분에서 설명된 것과 같이 그리고 본 개시물의 기술들에 따르면, 비디오 디코더 (30) 의 플리커 검출 유닛 (73) 은: 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하도록 구성될 수도 있다. 현재 픽처로부터의 블록이 플리커할 것인지 여부의 결정은 디스플레이 순서에서 현재 픽처에서의 블록 및 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초할 수도 있다.

[0118] 도 4 는 본 개시물의 기술들에 따라 플리커 검출을 수행하기 위한 프로세스를 도시하는 플로우차트이다. 예의 목적들을 위해, 비디오 인코더 또는 디코더, 예컨대 비디오 인코더 (20) 의 플리커 검출 유닛 (43) 또는 비디오 디코더 (30) 의 플리커 검출 유닛 (73) 은 본 명세서에 설명된 임의의 기술들을 수행할 수도 있는 것이 이해되어야만 한다.

[0119] 도 4 의 예에서, 비디오 인코더 (20) 의 플리커 검출 유닛 (43) 은 디스플레이 순서에서 현재 프레임의 블록과

다음 프레임의 병치된 블록 간의 SAD (또는 또 다른 차이 메트릭) 을 결정할 수도 있다 (200). 플리커 검출 유닛 (43) 이 SAD 가 임계치보다 크다고 결정하면 (결정 블록 (204) 에서 예 브랜치), 플리커 검출 유닛 (43) 은 현재 블록을 플리커하지 않는 것으로 마킹한다 (202).

[0120] 플리커 검출 유닛 (43) 이 SAD 가 임계치 이하라고 결정하면 (결정 블록 (204) 에서 아니오 브랜치), 플리커 검출 유닛 (43) 은 현재 블록까지의 현재 픽처의 누적된 SAD 를 결정한다 (206). 누적된 SAD 가 임계 누적된 SAD 이하이면 (208 의 아니오 브랜치), 모션 추정 유닛은 블록을 플리커하지 않는 것으로 마킹한다 (202).

[0121] 누적된 SAD 가 임계치보다 크다면 (208 의 예 브랜치), 플리커 검출 유닛 (43) 은 블록의 팔레트 사이즈 및 블록이 에지 블록인지를 결정한다 (210). 플리커 검출 유닛 (43) 은 또한 블록이 텍스처 블록인지를 결정한다 (216). 현재 픽처의 블록의 팔레트 사이즈가 임계 팔레트 사이즈 이하이거나, 블록이 에지 블록이라면 (212 의 아니오 브랜치), 플리커 검출 유닛 (43) 은 블록을 플리커하지 않는 것으로 마킹한다 (220). 팔레트 사이즈가 팔레트 사이즈 임계치보다 크고 블록이 에지 블록이 아니라면 (212 의 예 브랜치), 플리커 검출 유닛 (43) 은 블록을 플리커하는 것으로 마킹한다 (214).

[0122] 플리커 검출 유닛 (43) 이 블록이 텍스처 블록이라고 결정한다면 (218 의 예 브랜치), 플리커 검출 유닛 (43) 은 블록을 플리커하는 것으로 마킹한다 (214). 플리커 검출 유닛 (43) 이 블록이 텍스처 블록이 아니라고 결정한다면 (218 의 아니오 브랜치), 플리커 검출 유닛 (43) 은 블록을 플리커하지 않는 것으로 마킹한다 (202).

[0123] 도 5 는 본 개시물의 기술들에 따라 플리커 검출 및 플리커 완화를 수행하기 위한 프로세스를 도시하는 플로우 차트이다. 예의 목적들을 위해, 비디오 인코더 또는 디코더, 예컨대 비디오 인코더 (20) 또는 비디오 디코더 (30) 가 본 명세서에 설명된 임의의 기술들을 수행할 수도 있는 것이 이해되어야만 한다.

[0124] 도 5 의 예에서, 비디오 인코더 (20) 의 플리커 검출 유닛 (43) 및 엔트로피 인코딩 유닛 (56), 또는 비디오 디코더 (30) 의 엔트로피 디코딩 유닛 (70) 및 플리커 검출 유닛 (73) 은 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보, 예컨대 플리커링 플래그를 코딩할 수도 있다 (240). 플리커 검출 유닛 (43) 또는 플리커 검출 유닛 (73) 은 디스플레이 순서에서 현재 픽처에서의 블록 및 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초하여 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩할 수도 있다. 일부 예들에서, 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하기 위해, 비디오 인코더 (20) 또는 비디오 디코더 (30) 는 현재 픽처의 특정 영역이 플리커링 영역인 것을 표시하는 신택스 엘리먼트를 비트스트림에서 코딩할 수도 있다 (242).

[0125] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 디스플레이 순서에서 현재 픽처 및 다음 픽처를, 비디오 데이터 메모리 (39), 비디오 데이터 메모리 (69), 참조 프레임 버퍼 (69), 및/또는 참조 프레임 버퍼 (82) 에 저장할 수도 있다. 일부 예들에서, 플리커 검출 유닛 (43) 또는 플리커 검출 유닛 (73) 은 블록이 플리커할 것인지 여부를 표시하는 정보에 기초하여 블록을 코딩하도록 추가로 구성될 수도 있다.

[0126] 예를 들어, 플리커 검출 유닛 (43) 은 현재 픽처로부터의 블록이 플리커할 것인지 여부를 결정할 수도 있다. 디스플레이 순서에서 현재 픽처에서의 블록 및 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초하여 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를, 플리커 검출 유닛 (43) 은 생성할 수도 있고 엔트로피 인코딩 유닛 (56) 은 인코딩할 수도 있다. 플리커 검출 유닛 (43) 및 엔트로피 인코딩 유닛 (56) 은 블록이 플리커할 것인지 여부를 표시하는 정보에 기초하여 블록을 인코딩할 수도 있다. 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 인코딩하기 위해, 엔트로피 인코딩 유닛 (56) 은 그 정보를 생성하는 플리커링 검출 유닛 (43) 에 응답하는 CABAC 인코딩을 수행할 수도 있다.

[0127] 다른 예로서, 엔트로피 디코딩 유닛 (70) 은 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 수신할 수도 있다. 엔트로피 디코딩 유닛 (70) 은 정보를 수신하는 것에 응답하여 수신된 정보에 CABAC 디코딩을 수행할 수도 있다. 플리커 검출 유닛 (73) 은 디스플레이 순서에서 현재 픽처에서의 블록 및 디스플레이 순서에서 다음 픽처로부터의 병치된 블록에 기초하여 블록이 플리커할 것인지 여부를 추가로 결정할 수도 있다. 플리커 검출 유닛 (73) 및 비디오 디코더 (30) 는 블록이 플리커할 것인지 여부를 표시하는 정보에 기초하여 블록을 추가로 디코딩할 수도 있다.

[0128] 다양한 예들에서, 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 는 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보에 기초하여 다양한 타입들의 프로세싱을 수행할 수도 있다. 일부 예들에서, 비디오 디코더 (30) 는 다양한 사후 디코딩 프로세싱 기술들을 수행할 수도 있다.

- [0129] 일부 예들에서, 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩하기 위해, 비디오 인코더 (20) 또는 비디오 디코더 (30) 는 현재 픽처의 특정 영역이 플리커링 영역인 것을 표시하는 신덱스 엘리먼트를 비트스트림에서 코딩할 수도 있다. 그 블록은 또한 일부 예들에서 플리커링 영역 내에 있을 수도 있다.
- [0130] 다른 예에서, 블록을 코딩하기 위해, 플리커 검출 유닛 (43) 또는 플리커 검출 유닛 (73) 은 플리커링 영역인 것으로 표시된 특정 영역에 대한 코딩 툴들에 관한 제약들에 기초하여 블록을 코딩할 수도 있다. 제약들은 플리커링 영역의, 그리고 현재 블록에 대하여 공간적으로 이웃하는 블록을 인트라-예측에 대하여 사용불가능한 것으로 마킹함으로써 플리커링 영역에서 공간 인트라 예측을 디스에이블하는 것을 포함한다. 제약들은 또한, 예컨대 플리커 검출 유닛 (43) 또는 플리커 검출 유닛 (73) 에 의해 블록들을 다양한 예들에서 인트라 예측에 대하여 사용불가능한 것으로 마킹하는 것을 포함할 수도 있다.
- [0131] 일부 예들에서, 엔트로피 인코딩 유닛 (56) 또는 엔트로피 디코딩 유닛 (70) 은 플리커링 영역에 대한 CABAC 컨텍스트 모델링의 상태를 리셋하고, CABAC 컨텍스트 모델링의 리셋된 상태에 기초하여 현재 픽처의 블록을 코딩 (즉, 인코딩 또는 디코딩) 할 수도 있다. 픽처가 플리커할 것을 표시하는 정보를 코딩하기 위해, 엔트로피 인코딩 유닛 (56) 또는 엔트로피 디코딩 유닛 (70) 은 다양한 예들에서, 각각 플리커링 플래그를 인코딩하거나 디코딩할 수도 있다.
- [0132] 일부 예들에서, 플리커 검출 유닛 (43) 은: 현재 픽처의 블록과 다음 픽처의 블록 간의 SAD 또는 현재 블록과 다음 블록 간의 MSE 중 적어도 하나에 기초하여, 블록이 플리커할 것을 표시하는 정보를 코딩할 수도 있다. 추가의 예들에서, 현재 블록이 플리커할 것을 표시하는 정보를 코딩하기 위해, 플리커 검출 유닛 (43) 은 SAD 또는 MSE 중 적어도 하나가 임계치 미만일 경우 블록의 팔레트 사이즈가 임계 팔레트 사이즈를 초과하는지 여부에 기초하여, 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩할 수도 있다.
- [0133] 다양한 예들에서, 플리커링 플래그를 CABAC 코딩하기 위해, 엔트로피 인코딩 유닛 (56) 또는 엔트로피 디코딩 유닛 (70) 은 현재 픽처의 블록에 대하여 공간적으로 이웃하는 블록의 플리커링 플래그들의 값들을 CABAC 코딩에 대한 컨텍스트로서 사용하여 플리커링 플래그를 CABAC 코딩할 수도 있다.
- [0134] 다양한 예들에서, 플리커 검출 유닛 (43) 은 오직 영역의 CU 가 최소 사이즈를 갖는 경우에만 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩할 수도 있다. 다양한 예들에서, 플리커 검출 유닛 (43) 은: 블록의 팔레트 사이즈가 임계 팔레트 사이즈를 초과하는 것, 블록이 에지 블록인 것, 또는 블록이 텍스처 블록인 것 중 적어도 하나의 경우에, 현재 픽처로부터의 블록이 플리커할 것인지 여부를 표시하는 정보를 코딩할 수도 있다.
- [0135] 하나 이상의 예들에 있어서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현된다면, 그 기능들은 하나 이상의 명령들 또는 코드로서 컴퓨터 판독가능 매체 상으로 저장 또는 전송되고 하드웨어 기반 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터 판독가능 매체들은, 데이터 저장 매체들과 같은 유형의 매체에 대응하는 컴퓨터 판독가능 저장 매체들, 또는 예를 들어, 통신 프로토콜에 따라, 일 장소에서 다른 장소로의 컴퓨터 프로그램의 전송을 용이하게 하는 임의의 매체를 포함하는 통신 매체들을 포함할 수도 있다. 이러한 방식으로, 컴퓨터 판독 가능한 매체들은 일반적으로 (1) 비-일시적인 유형의 컴퓨터 판독가능 저장 매체들 또는 (2) 신호 또는 반송파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체는 본 개시물에서 설명된 기술들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 추출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 가용 매체일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터 판독가능 매체를 포함할 수도 있다.
- [0136] 제한이 아닌 예로서, 그러한 컴퓨터 판독가능 저장 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 저장부, 자기 디스크 저장부 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 원하는 프로그램 코드를 명령들 또는 데이터 구조들의 형태로 저장하는데 이용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 커넥션이 컴퓨터 판독가능 매체로 적절히 명명된다. 예를 들면, 명령들이 동축 케이블, 광섬유 케이블, 연선, 디지털 가입자 회선 (DSL), 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들을 이용하여 웹사이트, 서버, 또는 다른 원격 소스로부터 송신된다면, 동축 케이블, 광섬유 케이블, 연선, DSL, 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들은 매체의 정의 내에 포함된다. 그러나, 컴퓨터 판독가능 저장 매체들 및 데이터 저장 매체들은 접속들, 반송파들, 신호들, 또는 다른 일시적 매체들을 포함하는 것이 아니라, 대신에 비-일시적인, 유형의 저장 매체들과 관련되는 것이 이해되어야만 한다. 본 명세서에서 사용된 바와 같은 디스크 (disk) 및 디스크 (disc) 는 컴팩트 디스크 (CD), 레이저 디스크, 광학 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크 및 블루레이 디스크를 포함하며, 여기서, 디스크 (disk) 는

통상적으로 데이터를 자기적으로 재생하지만 디스크 (disc) 는 레이저들을 이용하여 데이터를 광학적으로 재생한다. 위의 조합들도 컴퓨터 판독가능 매체들의 범위 내에 포함되어야 한다.

[0137] 명령들은 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로프로세서들, 주문형 집적회로들 (ASIC들), 필드 프로그래밍가능 로직 어레이들 (FPGA들), 또는 다른 등가의 집적된 또는 별도의 로직 회로와 같은 하나 이상의 프로세서들에 의해 실행될 수도 있다. 따라서, 본 명세서에서 사용되는 바와 같은 용어 "프로세서" 는 본 명세서에서 설명된 기술들의 구현에 적절한 전술한 구조 또는 임의의 다른 구조 중 임의의 구조를 지칭할 수도 있다. 부가적으로, 일부 양태들에 있어서, 본 명세서에서 설명된 기능은 인코딩 및 디코딩을 위해 구성되고 결합된 코덱에서 통합된 전용 하드웨어 모듈 및/또는 소프트웨어 모듈들 내에 제공될 수도 있다. 또한, 그 기술들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수 있다.

[0138] 예에 의존하여, 본 명세서에서 설명된 임의의 방법들의 소정의 행위들 또는 이벤트들은 상이한 시퀀스로 수행될 수 있거나, 추가되거나 병합되거나, 또는 함께 제거될 수도 있다 (예를 들면, 설명된 모든 행위들 또는 이벤트들이 방법의 실시예에 반드시 필요한 것은 아니다) 는 것이 인식될 것이다. 더욱이, 특정 실시형태들에 있어서, 행위들 또는 이벤트들은 순차적인 것보다는, 예를 들어, 멀티-스레드 프로세싱, 인터럽트 프로세싱 또는 다중의 프로세서들을 통해 동시에 수행될 수도 있다.

[0139] 당업자는 본명세서에 개시된 방법들, 시스템들, 및 장치들과 관련하여 설명된 다양한 예시적인 논리 블록들, 모듈들, 회로들, 및 알고리즘 단계들이 전자 하드웨어, 프로세서에 의해 실행되는 컴퓨터 소프트웨어, 또는 이들 양자의 조합들로서 구현될 수도 있음을 인식할 것이다. 하드웨어 및 소프트웨어의 이러한 상호 교환성을 명확하게 설명하기 위해, 다양한 예시적인 컴포넌트들, 블록들, 모듈들, 회로들, 및 단계들이 그들의 기능적 관점에서 일반적으로 상술되었다. 그러한 기능이 하드웨어로 구현되는지 또는 소프트웨어로 구현되는지의 여부는 특정 애플리케이션 및 전체 시스템에 부과되는 설계 제약들에 따라 달라진다. 당업자라면, 상기 상술한 기능성을 각각의 특정 애플리케이션에 대해 다양한 방식으로 구현할 수도 있지만, 이러한 구현 결정은 본 발명의 범위를 벗어나게 하는 것으로 이해되어서는 안된다.

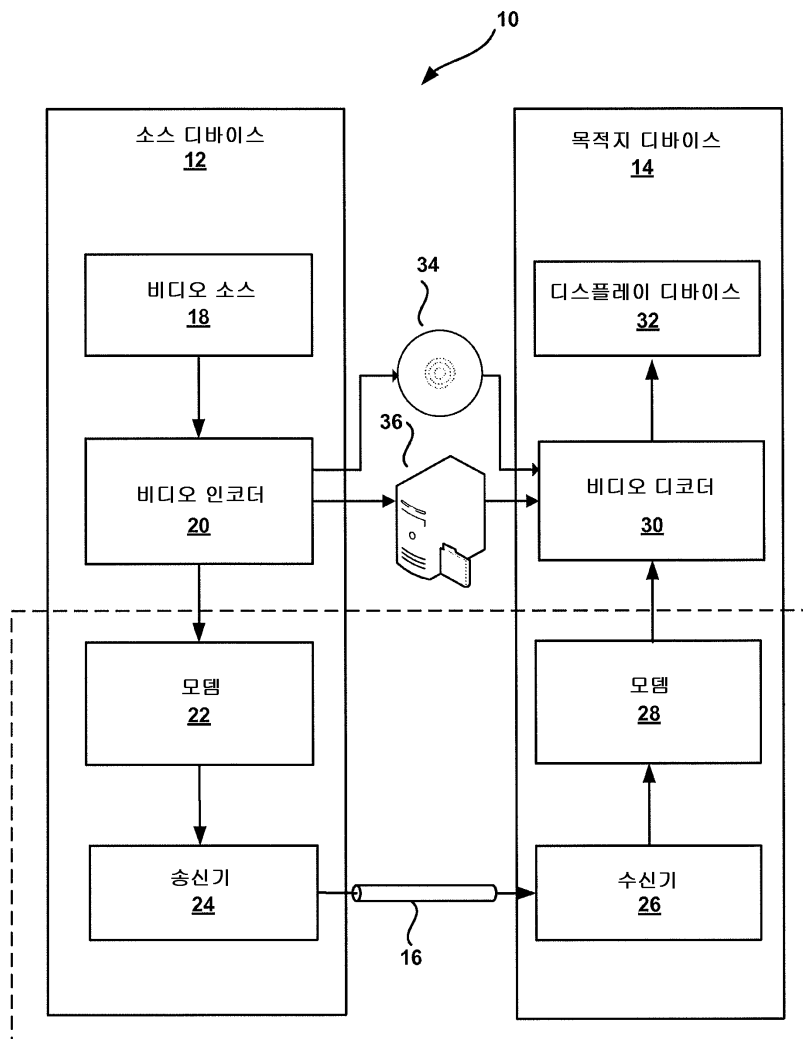
[0140] 추가로, 본 명세서에 개시된 예들은 범용 프로세서, 디지털 신호 프로세서 (DSP), 주문형 집적회로 (ASIC), 필드 프로그램가능 게이트 어레이 (FPGA) 또는 다른 프로그램가능 로직 디바이스, 별개의 게이트 또는 트랜지스터 로직, 별개의 하드웨어 컴포넌트들, 또는 본 명세서에서 설명된 기능들을 수행하도록 설계된 이들의 임의의 조합과 같은 전자 디바이스 또는 회로로 구현되거나 수행될 수도 있다. 범용 프로세서는 마이크로프로세서일 수도 있지만, 대안적으로, 그 프로세서는 임의의 종래의 프로세서, 제어기, 마이크로 제어기, 또는 상태 머신일 수도 있다. 프로세서는 또한 컴퓨팅 디바이스들의 조합, 예를 들면, DSP와 마이크로프로세서의 조합, 복수의 마이크로프로세서들의 조합, DSP 코어와 연계한 하나 이상의 마이크로프로세서들의 조합, 또는 임의의 다른 그러한 구성으로서 구현될 수도 있다.

[0141] 본 명세서에 개시된 실시형태들과 관련하여 설명된 방법 또는 알고리즘의 단계들은 하드웨어에서, 프로세서에 의해 실행되는 소프트웨어 모듈에서, 또는 이들 양자의 조합에서 직접 구현될 수도 있다. 소프트웨어 모듈은 RAM 메모리, 플래시 메모리, ROM 메모리, EPROM 메모리, EEPROM 메모리, 레지스터들, 하드 디스크, 착탈가능 디스크, CD-ROM, 또는 당업계에 공지된 임의의 다른 형태의 저장 매체에 상주할 수도 있다. 예시적인 저장 매체는, 프로세서가 저장 매체로부터 정보를 판독할 수 있고 저장 매체에 정보를 기입할 수 있도록 프로세서에 커플링된다. 대안에서, 저장 매체는 프로세서에 통합될 수도 있다. 프로세서 및 저장 매체는 ASIC에 상주할 수도 있다. ASIC 는 사용자 단말기에 상주할 수도 있다. 대안에서, 프로세서와 저장 매체는 사용자 단말기에서 개별 컴포넌트들로서 상주할 수도 있다.

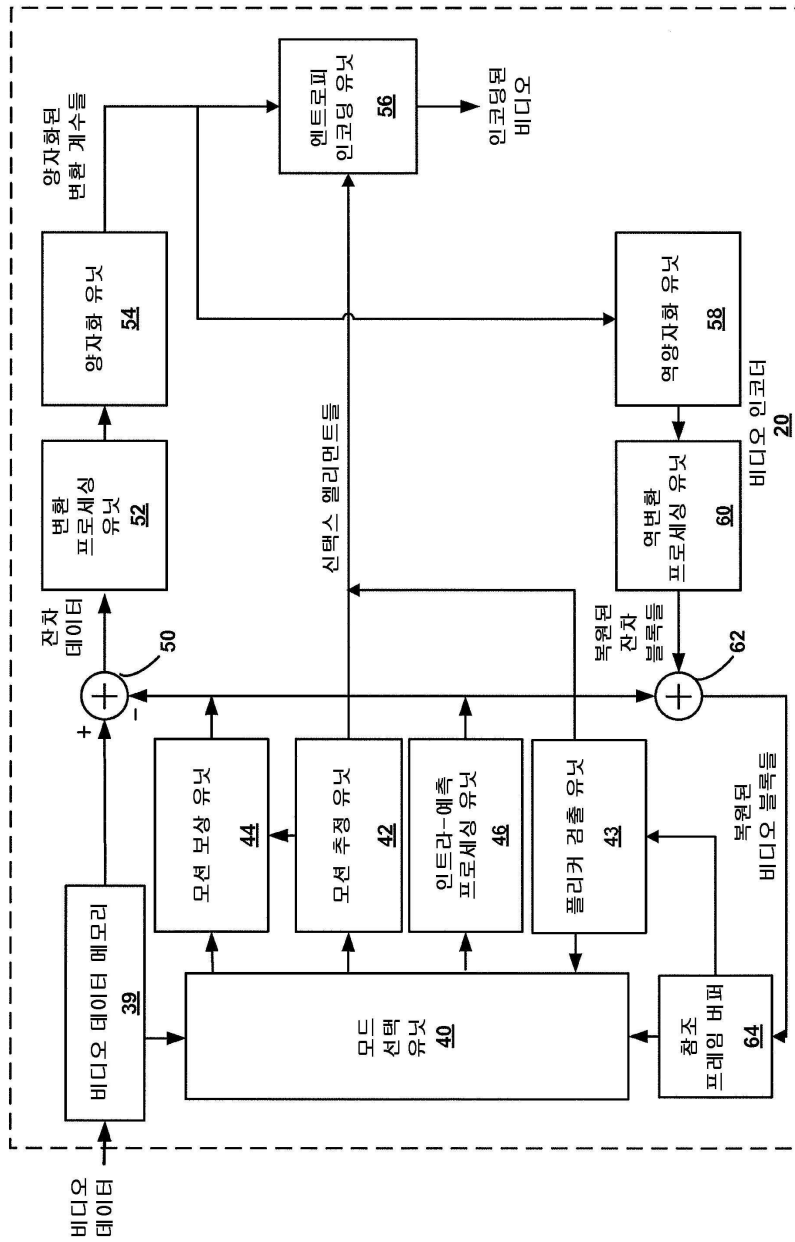
[0142] 본 개시물의 기술들은 무선 핸드셋, 집적 회로 (IC) 또는 IC들의 세트 (예를 들어, 칩 세트) 를 포함하여 매우 다양한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들, 모듈들 또는 유닛들이 개시된 기술들을 수행하도록 구성된 디바이스들의 기능적 양태들을 강조하기 위해 본 개시물에서 설명되지만, 반드시 상이한 하드웨어 유닛들에 의한 실현을 요구하지는 않는다. 오히려, 상기 설명된 바와 같이, 다양한 유닛들은 적절한 소프트웨어 및/또는 펌웨어와 함께 상기 설명된 바와 같은 하나 이상의 프로세서들을 포함하여 코덱 하드웨어 유닛으로 결합되거나 또는 상호운용식 하드웨어 유닛들의 집합에 의해 제공될 수도 있다.

도면

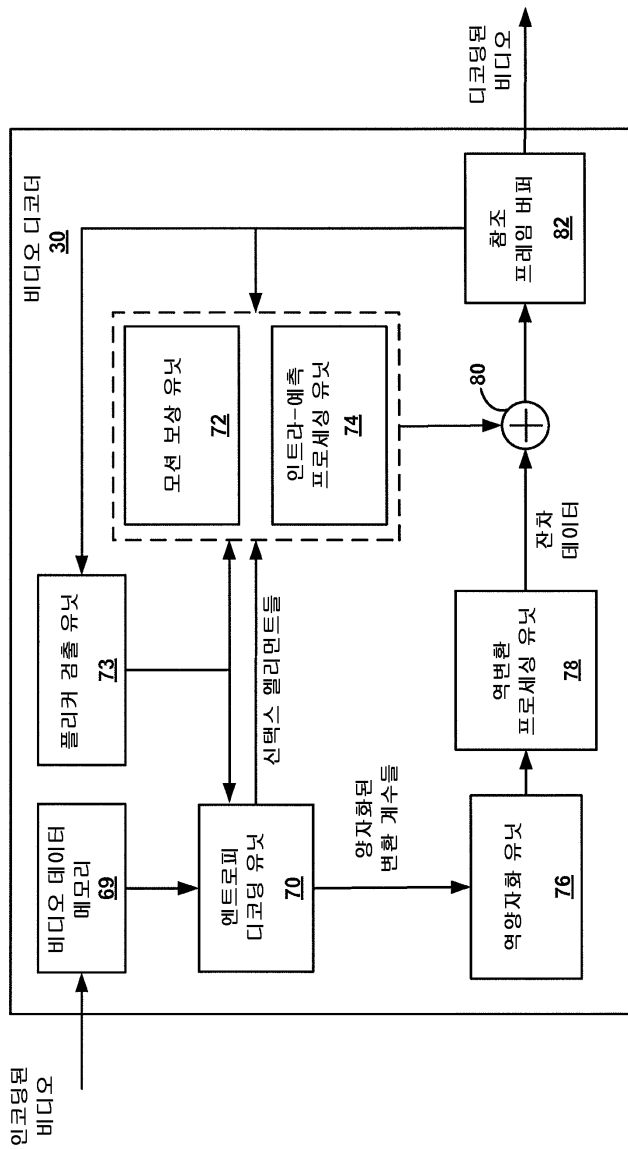
도면1



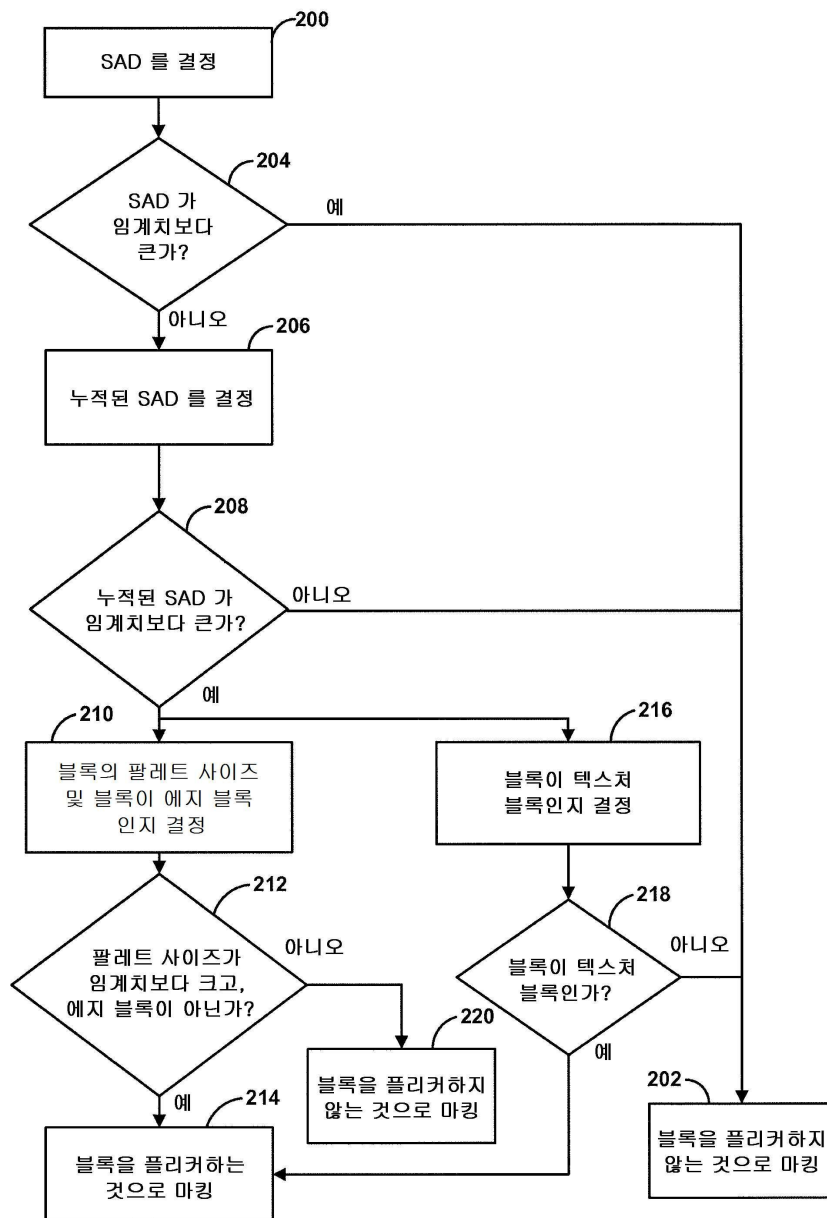
도면2



도면3



도면4



도면5

