



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2025년03월06일
(11) 등록번호 10-2775507
(24) 등록일자 2025년02월26일

(51) 국제특허분류(Int. Cl.)
H04N 19/527 (2014.01) H04N 19/119 (2014.01)
H04N 19/137 (2014.01) H04N 19/159 (2014.01)
H04N 19/176 (2014.01) H04N 19/186 (2014.01)
H04N 19/513 (2014.01) H04N 19/54 (2014.01)
H04N 19/80 (2014.01) H04N 19/96 (2014.01)
(52) CPC특허분류
H04N 19/527 (2015.01)
H04N 19/119 (2015.01)
(21) 출원번호 10-2021-7015100
(22) 출원일자(국제) 2020년03월11일
심사청구일자 2021년05월18일
(85) 번역문제출일자 2021년05월18일
(65) 공개번호 10-2021-0076125
(43) 공개일자 2021년06월23일
(86) 국제출원번호 PCT/US2020/022066
(87) 국제공개번호 WO 2020/185876
국제공개일자 2020년09월17일
(30) 우선권주장
62/817,517 2019년03월12일 미국(US)
(56) 선행기술조사문헌
Benjamin Bross, et. al., "Versatile Video
Coding (Draft 3)", Joint Video Experts Team
(JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC
1/SC 29/WG 11 12th Meeting: Macao, CN, 3-12
Oct. 2018, JVET-L1001-v9.*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
텐센트 아메리카 엘엘씨
미국 94306 캘리포니아주 팔로 알토 파크 블러바
드 2747
(72) 발명자
자오, 신
미국 94306 캘리포니아 팔로 알토 파크 블러버드
2747 텐센트 아메리카 엘엘씨 내
쉬, 샤오중
미국 94306 캘리포니아 팔로 알토 파크 블러버드
2747 텐센트 아메리카 엘엘씨 내
(뒷면에 계속)
(74) 대리인
양영준, 김연송, 백만기

전체 청구항 수 : 총 10 항

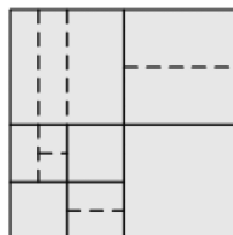
심사관 : 박상철

(54) 발명의 명칭 비디오 인코딩 또는 디코딩을 위한 방법 및 장치

(57) 요약

비디오 시퀀스를 인코딩 또는 디코딩하는 방법 및 장치는 4:4:4 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩하는 것, 또는 4:2:2 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩하는 것을 포함하고, 4:4:4 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩할 때, 평균화 연산 이외의 연산을 이용하여 하나의 4x4 크로마 블록의 아핀 모션 벡터를 복사하고, 및 4:2:2 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩할 때, 각각의 4x4 크로마 블록을 2개의 4x4 동위치 루마 블록과 연관시켜 하나의 4x4 크로마 블록의 아핀 모션 벡터가 2개의 동위치 크로마 블록의 모션 벡터들의 평균이도록 한다.

대표도



(52) CPC특허분류

H04N 19/137 (2015.01)
H04N 19/159 (2015.01)
H04N 19/176 (2015.01)
H04N 19/186 (2015.01)
H04N 19/513 (2015.01)
H04N 19/54 (2015.01)
H04N 19/80 (2015.01)
H04N 19/96 (2015.01)

(72) 발명자

류, 산

미국 94306 캘리포니아 팔로 알토 파크 불러버드
2747 텐센트 아메리카 엘엘씨 내

류, 산

미국 94306 캘리포니아 팔로 알토 파크 불러버드
2747 텐센트 아메리카 엘엘씨 내

명세서

청구범위

청구항 1

비디오 시퀀스를 디코딩하기 위한 방법으로서,

4:4:4 크로마 포맷 및 4:2:2 크로마 포맷 중 하나를 이용하여 상기 비디오 시퀀스를 디코딩하는 단계를 포함하고,

상기 4:4:4 크로마 포맷을 이용하여 상기 비디오 시퀀스를 디코딩할 때, 상기 방법은 평균화 연산 이외의 연산을 이용하여 하나의 4x4 루마 블록의 아핀 모션 벡터를 복사하고 상기 아핀 모션 벡터를 동위치 4x4 크로마 블록에 연관시키는 단계를 추가로 포함하고,

상기 4:2:2 크로마 포맷을 이용하여 상기 비디오 시퀀스를 디코딩할 때, 상기 방법은 각각의 4x4 크로마 블록을 2개의 4x4 동위치 루마 블록과 연관시켜 하나의 4x4 크로마 블록의 아핀 모션 벡터가 상기 2개의 동위치 루마 블록의 모션 벡터들의 평균이도록 하는 단계를 추가로 포함하고,

최대 허용 변환 크기는 상이한 컬러 성분들에 대해 동일하고,

상기 4:2:2 크로마 포맷을 이용하여 상기 비디오 시퀀스를 디코딩할 때, 최대 수직 변환 크기는 상이한 컬러 성분들 중에서 동일하고, 컬러 성분들에 대한 최대 수평 변환 크기는 루마 성분들에 대한 최대 수평 변환 크기의 절반인,

방법.

청구항 2

제1항에 있어서,

상기 크로마 포맷에 관계없이, 현재 4x4 크로마 블록을 4개의 2x2 서브-블록으로 분할하는 단계,

상단-좌측 2x2 크로마 서브-블록에 대해 동위치 루마 블록의 제1 아핀 모션 벡터를 도출하는 단계,

하단-우측 2x2 크로마 블록에 대해 상기 동위치 루마 블록의 제2 아핀 모션 벡터를 도출하는 단계, 및

상기 제1 아핀 모션 벡터와 상기 제2 아핀 모션 벡터의 평균을 사용하여 상기 현재 4x4 크로마 블록의 아핀 모션 벡터를 도출하는 단계를 추가로 포함하는 방법.

청구항 3

제1항에 있어서,

루마 성분과 크로마 성분 사이의 모션 보상을 위해 사용되는 보간 필터를 정렬하는 단계를 추가로 포함하는 방법.

청구항 4

제3항에 있어서,

4:2:0 크로마 포맷을 이용하여 비디오 시퀀스가 입력될 때, 루마 성분들 및 크로마 성분들에 대해 8-탭 보간 필터를 적용하는 방법.

청구항 5

제1항에 있어서,

성분들 Y, Cb, 및 Cr를 3개의 개별 트리로서 코딩하는 단계를 추가로 포함하고,

상기 3개의 개별 트리 중 각각의 트리는 상기 성분들 Y, Cb, 및 Cr 중 하나의 성분을 코딩하는 방법.

청구항 6

제5항에 있어서,

상기 3개의 개별 트리로서의 코딩이 I 슬라이스 또는 I 타일 그룹에 대해 수행되는 방법.

청구항 7

삭제

청구항 8

삭제

청구항 9

제1항에 있어서,

PDPC(Position-Dependent Predictor combination), MTS(Multiple Transform Selection), NSST(Non-Separable Secondary Transform), ISP(Intra-Sub Partitioning), 및 MRL(Multiple reference line) 인트라 예측 중 적어도 하나가 루마 성분 및 크로마 성분 둘 다에 적용되는 방법.

청구항 10

제9항에 있어서,

상기 MRL(Multiple reference line) 인트라 예측이 상기 루마 성분 및 상기 크로마 성분 둘 다에 적용될 때, 그리고 상기 비디오 시퀀스를 디코딩하는 것이 상기 4:4:4 크로마 포맷을 사용하여 수행될 때, 상기 방법은 인트라 예측을 위한 N번째 참조를 선택하고, 크로마 성분들에 대한 명시적 시그널링 없이 동일한 참조 라인을 사용하는 단계를 추가로 포함하고,

상기 ISP(Intra-Sub Partitioning)이 상기 루마 성분 및 상기 크로마 성분 둘 다에 적용될 때, 상기 방법은 성분들 Y, Cb, 및 Cr에 대한 현재 블록에 대한 블록 레벨에서 상기 ISP를 적용하는 단계를 추가로 포함하고,

상기한 트리들이 상이한 컬러 성분들에 대해 사용될 때, 상기 방법은 시그널링 없이 병치된 Y 성분들로부터 U 및 V 성분들에 대한 코딩 파라미터들을 암시적으로 도출하는 단계를 추가로 포함하는 방법.

청구항 11

비디오 시퀀스를 디코딩하기 위한 디바이스로서:

프로그램 코드를 저장하도록 구성된 적어도 하나의 메모리;

제1항 내지 제6항, 제9항 및 제10항 중 어느 한 항의 방법을 수행하기 위해 상기 프로그램 코드를 판독하고 상기 프로그램 코드에 의해 지시된 바와 같이 동작하도록 구성된 적어도 하나의 프로세서를 포함하는 디바이스.

청구항 12

프로그램 코드를 저장하는 비일시적 컴퓨터 판독가능 매체로서, 상기 프로그램 코드는 하나 이상의 명령어를 포함하고, 상기 하나 이상의 명령어는 디바이스의 하나 이상의 프로세서에 의해 실행될 때, 상기 하나 이상의 프로세서로 하여금 제1항 내지 제6항, 제9항 및 제10항 중 어느 한 항의 방법을 수행하게 하는 비일시적 컴퓨터 판독가능 매체.

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

발명의 설명

기술 분야

[0001] [관련 출원에 대한 상호 참조]

[0002] 본 출원은 미국 특허상표청에 2019년 3월 12일자로 출원된 미국 가출원 제62/817,517호로부터 35U.S.C. § 119 하의 우선권을 주장하며, 그 개시내용은 그 전체가 본 명세서에 참조로 포함된다.

[0003] 실시예들에 따른 방법들 및 장치들은 비디오 처리에 관한 것으로, 더 특정하게는, VVC(Versatile Video Coding)에서 상이한 크로마 포맷들(예를 들어, 4:4:4, 4:2:2)을 지원할 수 있는 비디오 시퀀스를 인코딩 또는 디코딩하는 것에 관한 것이다.

배경 기술

[0004] 최근에, ITU-T(ITU Telecommunication Standardization Sector)의 VCEG(Video Coding Experts Group), ITU(International Telecommunication Union)의 섹터, 및 ISO/IEC MPEG(JTC 1/SC 29/WG 11), ISO(International Organization for Standardization) 및 IEC(International Electrotechnical Commission)의 공동 기술 위원회 ISO/IEC JTC 1의 표준화 하위 위원회는 2013년에 H.265/HEVC(High Efficiency Video Coding) 표준(버전 1)을 공개했다. 이 표준은 2014년에 버전 2로, 2015년에 버전 3으로, 그리고 2016에 버전 4로 업데이트되었다.

[0005] 그때 이후로 그들은 HEVC 표준(그것의 확장들을 포함함)의 것을 상당히 초과하는 압축 능력을 갖는 미래의 비디오 코딩 기술의 표준화에 대한 잠재적 필요성을 연구하고 있다. 2017년 10월에, 그들은 HEVC를 뛰어넘는 능력을 갖는 비디오 압축에 대한 공동 제안 요청서(CfP)를 발행했다. 2018년 2월 15일자로, 표준 동적 범위(standard dynamic range, SDR)에 대한 총 22개의 CfP 응답, 높은 동적 범위(high dynamic range, HDR)에 대한 12개의 CfP 응답, 및 360개의 비디오 범주에 대한 12개의 CfP 응답이 제각기 제출되었다. 2018년 4월에, 모든 접수된 CfP 응답들은 122 MPEG/10th JVET(Joint Video Exploration Team - Joint Video Expert Team) 회의에서 평가되었다. 주의 깊은 평가에 의해, JVET는 HEVC를 뛰어넘는 차세대 비디오 코딩의 표준, 즉, 소위 다용도 비디오 코딩(Versatile Video Coding, VVC)을 공식적으로 론칭하였다.

[0006] HEVC 블록 파티셔닝 구조가 이제 설명될 것이다. HEVC에서, 코딩 트리 유닛(coding tree unit, CTU)은 다양한 로컬 특성들에 적응하기 위해 코딩 트리로 나타내어진 쿼드트리 구조를 이용하여 코딩 유닛(coding unit, CU)들로 스플릿된다. 인터 픽처(시간적) 또는 인트라 픽처(공간적) 예측을 사용하여 픽처 영역을 코딩할지에 대한 결정이 CU 레벨에서 이루어질 수 있다. 각각의 CU는 PU 스플릿 타입에 따라 1개, 2개 또는 4개의 예측 유닛(prediction unit, PU)으로 추가로 스플릿될 수 있다. 하나의 PU 내부에서는, 동일한 예측 프로세스가 적용될 수 있고 관련 정보가 PU 기준으로 디코더에 송신될 수 있다. PU 스플릿 타입에 기초하여 예측 프로세스를 적용

함으로써 잔차 블록을 획득한 후에, CU는 CU에 대한 코딩 트리과 같은 또 다른 쿼드트리 구조에 따라 변환 유닛(transform unit, TU)들로 파티셔닝될 수 있다. HEVC 구조의 특징은 그것이 CU, PU, 및 TU를 포함하는 다중 파티션 개념을 포함할 수 있다는 것이다. HEVC에서, CU 또는 TU는 단지 정사각형 형상일 수 있는 반면, PU는 인터 예측된 블록에 대해 정사각형 또는 직사각형 형상일 수 있다. HEVC에서, 하나의 코딩 블록은 4개의 정사각형 서브-블록으로 추가로 스플릿될 수 있고, 변환은 각각의 서브-블록, 즉 TU에 대해 수행될 수 있다. 각각의 TU는 재귀적으로 (쿼드트리 스플릿을 사용하여) 더 작은 Tus로 추가로 스플릿될 수 있다. 이것은 RQT(Residual Quad-Tree)로서 지칭된다.

[0007] 픽처 경계에서, HEVC는, 블록이 그 크기가 픽처 경계에 맞추어질 때까지 쿼드-트리 스플릿을 유지하도록 암시적 쿼드-트리 스플릿을 채택한다.

[0008] 이제, QT(quad-tree)에 BT(binary tree)를 더한 것을 사용한 블록 파티셔닝 구조를 설명할 것이다. HEVC에서, CTU는 다양한 로컬 특성들에 적응하기 위해 코딩 트리로 나타내어진 쿼드트리 구조를 이용하여 CU들로 스플릿된다. 인터-픽처(시간) 또는 인트라-픽처(공간) 예측을 이용하여 픽처 영역을 코딩할지에 대한 결정은 CU 레벨에서 이루어질 수 있다. 각각의 CU는 PU 스플릿 타입에 따라 하나의, 2개의, 또는 4개의 PU로 추가로 스플릿될 수 있다. 하나의 PU 내부에서는, 동일한 예측 프로세스가 적용될 수 있고 관련 정보가 PU 기준으로 디코더에 송신될 수 있다. PU 스플릿 타입에 기초하여 예측 프로세스를 적용함으로써 잔차 블록을 획득한 후에, CU는 CU에 대한 코딩 트리과 같은 또 다른 쿼드트리 구조에 따라 변환 유닛(transform unit, TU)들로 파티셔닝될 수 있다. HEVC 구조의 하나의 특징은 그것이 CU, PU, 및 TU를 포함하는 다중의 파티션 개념을 포함한다는 것이다.

[0009] QTBT 구조는 다중의 파티션 타입의 개념들을 제거하는데, 즉, 그것은 CU, PU, 및 TU 개념들의 분리를 제거하고, CU 파티션 형상들에 대한 더 많은 유연성을 지원한다. QTBT 블록 구조에서, CU는 정사각형 또는 직사각형 형상을 가질 수 있다. 도 1a에 도시된 바와 같이, CTU(coding tree unit)가 쿼드트리 구조에 의해 먼저 파티셔닝된다. 이후 쿼드트리 리프 노드들은 이진 트리 구조에 의해 추가로 파티셔닝될 수 있다. 이진 트리 스플릿에는, 대칭 수평 스플릿 및 대칭 수직 스플릿의 2개의 스플릿 타입이 존재할 수 있다. 이러한 바이너리 트리 리프 노드들은 코딩 유닛들(coding units, CU들)이라고 불리고, 어떠한 추가의 파티셔닝도 없이 예측 및 변환 처리를 위해 해당 세그먼트화가 사용될 수 있다. 이는 CU, PU 및 TU가 QTBT 코딩 블록 구조에서 동일한 블록 크기를 갖는다는 점을 의미한다. JEM에서, CU는 때때로 상이한 컬러 성분들의 코딩 블록들(coding blocks, CB들)로 구성될 수 있는데, 예를 들어, 하나의 CU는 4:2:0 크로마 포맷의 P 및 B 슬라이스들의 경우에 하나의 루마 CB 및 2개의 크로마 CB를 포함할 수 있고; 그리고 단일 성분의 CB를 포함할 수 있는데, 예를 들어, 하나의 CU는 I 슬라이스들의 경우에 단지 하나의 루마 CB 또는 단지 2개의 크로마 CB를 포함할 수 있다.

[0010] QTBT 파티셔닝 스킴에 대해 다음 파라미터들이 정의된다:

[0011] CTU 크기: HEVC에서와 동일한 개념인 쿼드트리의 루트 노드 크기;

[0012] MinQTSIZE: 최소 허용 쿼드트리 리프 노드 크기;

[0013] MaxBTSIZE: 최대 허용 이진 트리 루트 노드 크기;

[0014] MaxBTDepth: 최대 허용 이진 트리 깊이;

[0015] MinBTSIZE: 최소 허용 이진 트리 리프 노드 크기;

[0016] QTBT 파티셔닝 구조의 일 예에서, CTU 크기는 2개의 대응하는 64x64 블록의 크로마 샘플들을 가지며 128x128 루마 샘플들로서 설정될 수 있고, MinQTSIZE는 16x16으로 설정될 수 있고, MaxBTSIZE는 64x64로 설정될 수 있고, MinBTSIZE(폭과 높이 둘 다에 대해)는 4x4로 설정될 수 있고, MaxBTDepth는 4로 설정될 수 있다. 쿼드트리 파티셔닝은 먼저 CTU에 적용되어 쿼드트리 리프 노드들을 생성할 수 있다. 쿼드트리 리프 노드들은 16x16(즉, MinQTSIZE) 내지 128x128(즉, CTU 크기)의 크기를 가질 수 있다. 리프 쿼드트리 노드가 128x128이면, 그것은 이진 트리에 의해 추가로 스플릿되지 않는데 그 이유는 그 크기가 MaxBTSIZE(즉, 64x64)를 초과하기 때문이다. 그렇지 않으면, 리프 쿼드트리 노드는 이진 트리에 의해 추가로 파티셔닝될 수 있다. 따라서, 쿼드트리 리프 노드는 또한 이진 트리에 대한 루트 노드일 수 있고 그것은 0으로서 이진 트리 깊이를 가질 수 있다. 이진 트리 깊이가 MaxBTDepth(즉, 4)에 도달할 때, 어떤 추가 스플릿도 고려되지 않는다. 이진 트리 노드가 MinBTSIZE(즉, 4)와 동등한 폭을 갖는 경우, 어떤 추가 수평 스플릿도 고려되지 않는다. 유사하게, 이진 트리 노드가 MinBTSIZE와 동등한 높이를 갖는 경우, 어떤 추가 수직 스플릿도 고려되지 않는다. 이진 트리의 리프 노드들은 어떠한 추가 파티셔닝도 없이 예측 및 변환 처리에 의해 추가로 처리될 수 있다. JEM에서, 최대 CTU

크기는 256x256 루마 샘플들일 수 있다.

- [0017] 도 1a는 QTBT를 이용한 블록 파티셔닝의 일례를 예시하고, 도 1b는 대응하는 트리 표현을 예시한다. 실선들은 쿼드트리 스플릿을 나타내고 점선들은 이진 트리 스플릿을 나타낸다. 이진 트리의 각각의 스플릿(즉, 비-리프) 노드에서, 어느 스플릿 타입(즉, 수평 또는 수직)이 사용될 수 있는지를 나타내기 위해 하나의 플래그가 시그널링될 수 있으며, 여기서 0은 수평 스플릿을 나타내고 1은 수직 스플릿을 나타낸다. 쿼드트리 스플릿에 대해서는, 스플릿 타입을 나타낼 필요가 없는데 그 이유는 쿼드트리 스플릿은 동등한 크기를 갖는 4개의 서브-블록을 생성하기 위해 수평으로 및 수직으로 둘 다에서 블록을 스플릿하기 때문이다.
- [0018] 또한, QTBT 스킴은 루마 및 크로마가 별개의 QTBT 구조를 갖기 위한 유연성을 지원한다. 현재, P 및 B 슬라이스들에 대해, 하나의 CTU 내의 루마 및 크로마 CTB들이 동일한 QTBT 구조를 공유한다. 그러나, I 슬라이스들에 대해, 루마 CTB는 QTBT 구조에 의해 CU들로 파티셔닝될 수 있고, 크로마 CTB들은 또 다른 QTBT 구조, 즉, DT(DualTree) 구조에 의해 크로마 CU들로 파티셔닝될 수 있다. 이는 I 슬라이스 내의 CU가 루마 성분의 코딩 블록 또는 2개의 크로마 성분의 코딩 블록들로 구성되고, P 또는 B 슬라이스 내의 CU가 모든 3개의 색 성분의 코딩 블록들로 구성된다는 것을 의미한다.
- [0019] HEVC에서는, 모션 보상의 메모리 액세스를 감소시키기 위해 작은 블록들에 대한 인터 예측이 제한되고, 따라서 4x8 및 8x4 블록들에 대해 양방향 예측이 지원되지 않고, 4x4 블록들에 대해 인터 예측이 지원되지 않는다. JEM-7.0에서 구현된 QTBT 스킴에서, 이러한 제한들은 제거될 수 있다.
- [0020] 이제, 3진 트리(ternary tree, TT)를 이용한 블록 파티셔닝을 설명한다. MTT(Multi-type-tree) 구조가 제안되었다. MTT는 QTBT보다 더 유연한 트리 구조이다. MTT에서는, 도 2a 및 도 2b에 도시된 바와 같이, 쿼드-트리 및 이진-트리 이외에, 수평 및 수직 센터-사이드 트리플-트리(center-side triple-tree)들이 도입된다.
- [0021] 트리플 트리 파티셔닝의 몇 가지 이점은 다음을 포함한다.
- [0022] 쿼드-트리 및 이진-트리 파티셔닝에 보완을 제공함, 트리플 트리 파티셔닝은 블록 중심에 위치하는 객체들을 캡처할 수 있는 한편, 쿼드-트리 및 이진-트리는 블록 중심을 따라 스플릿될 수 있다;
- [0023] 트리플 트리들의 파티션들의 폭 및 높이는 2의 거듭제곱일 수 있고 따라서 추가적인 변환들이 필요하지 않다;
- [0024] 2-레벨 트리들의 설계는 주로 복잡도 감소에 의해 동기 부여된다; 이론적으로, 트리의 트래버싱의 복잡도는 T^D 이며, 여기서 T는 스플릿 타입들의 수를 나타내고, D는 트리의 깊이이다.
- [0025] 이제, YUV 포맷들이 설명될 것이다. 상이한 YUV 포맷들, 즉 크로마 포맷들이 도 3에 도시되어 있다. 상이한 크로마 포맷들은 상이한 컬러 성분들의 상이한 다운-샘플링 그리드들을 정의한다.
- [0026] 이제 교차-성분 선형 모델링(Cross-component linear modeling, CCLM)이 설명될 것이다. VTM에서, 인트라 PU의 크로마 성분에 대해, 인코더는, 평면, DC, 수평, 수직, 루마 성분으로부터의 인트라 예측 모드(DM)의 직접 사본, 좌측 및 상단 교차-성분 선형 모드(LT_CCLM), 좌측 교차-성분 선형 모드(L_CCLM), 및 상단 교차 성분 선형 모드(T_CCLM)를 포함하는 8개의 모드 중에서 최상의 크로마 예측 모드들을 선택한다. 이들 모드들 중에서, LT_CCLM, L_CCLM, 및 T_CCLM은 교차-성분 선형 모드(CCLM)로서 분류될 수 있다. 이러한 3개의 모드 사이의 차이는 이웃 샘플들의 상이한 영역들이 파라미터들 α 및 β 를 도출하기 위해 사용될 수 있다는 것이다. LT_CCLM의 경우, 좌측 및 상단 이웃 샘플들 둘 다는 파라미터들 α 및 β 를 도출하기 위하여 사용될 수 있다. L_CCLM의 경우, 좌측 이웃 샘플들만이 파라미터들 α 및 β 를 도출하기 위해 사용된다. T_CCLM에 대하여, 상단 이웃 샘플들만이 파라미터들 α 및 β 를 도출하기 위하여 사용된다.
- [0027] 교차-성분 선형 모델(CCLM) 예측 모드들은 교차 성분 중복성을 감소시키기 위해 사용될 수 있고, 여기서 크로마 샘플들은 다음과 같이 선형 모델을 사용함으로써 동일한 CU의 재구성된 루마 샘플들에 기초하여 예측될 수 있다:

수확식 1

$$pred_c(i,j) = \alpha \cdot rec_L'(i,j) + \beta.$$

여기서, $pred_c(i,j)$ 는 CU 내의 예측된 크로마 샘플들을 나타내고, $rec_L'(i,j)$ 는 동일한 CU의 다운샘플링된 재

구성된 루마 샘플들을 나타낸다. 파라미터들 α 및 β 는 직선 방정식, 예를 들어, max-min 방법에 의해 도출될 수 있다. 이러한 계산 프로세스는 인코더 검색 동작으로서만이 아니라 디코딩 프로세스의 일부로서 수행될 수 있으며, 따라서 α 및 β 값들을 운반하기 위해 어떠한 선택스도 사용되지 않는다.

[0030] 크로마 4:2:0 포맷의 경우, CCLM 예측은 6-탭 보간 필터를 적용하여, 도 3에 도시된 바와 같은 크로마 샘플에 대응하는 다운-샘플링된 루마 샘플을 획득한다. 여기서, 다운-샘플링된 루마 샘플 $\text{Rec}'L[x, y]$ 가 재구성된 루마 샘플들로부터 계산될 수 있다.

[0031] 다운-샘플링된 루마 샘플들은 최대 및 최소 샘플 포인트들을 찾는 데 사용될 수 있다. 2개의 포인트(루마와 크로마의 쌍)(A, B)는 도 4에 묘사된 바와 같이 이웃 루마 샘플들의 세트 내에서의 최소 및 최대 값들일 수 있다. 여기서 선형 모델 파라미터들 α 및 β 는 다음의 수학식에 따라 획득될 수 있다:

수학식 2

$$\alpha = \frac{y_B - y_A}{x_B - x_A}$$

[0032]

수학식 3

$$\beta = y_A - \alpha x_A$$

[0033]

[0034] 여기서, 분할은 회피되고 곱셈 및 시프트에 의해 대체될 수 있다. 룩업 테이블(Look-up Table, LUT)은 사전 계산된 값들을 저장하기 위해 사용될 수 있고, 최대 루마 샘플과 최소 루마 샘플 사이의 절대 차이 값들은 LUT의 엔트리 인덱스를 지정하기 위해 사용될 수 있으며, LUT의 크기는 512일 수 있다.

[0035] T_CCLM 모드에서, 도 5a 및 도 5b에 도시된 이웃 샘플들($2 \times W$ 샘플들을 포함함)만이 선형 모델 계수들을 계산하기 위해 사용될 수 있다.

[0036] L_CCLM 모드에서, 도 6a 및 도 6b에 도시된 바와 같이, 좌측 이웃 샘플들($2 \times H$ 샘플들을 포함함)만이 선형 모델 계수들을 계산하기 위해 사용될 수 있다.

[0037] CCLM 예측 모드는 또한 2개의 크로마 성분 사이의 예측을 포함하는데, 즉, Cr 성분이 Cb 성분으로부터 예측될 수 있다. 재구성된 샘플 신호를 이용하는 대신에, CCLM Cb-대-Cr 예측이 잔차 도메인에서 적용될 수 있다. 이는 최종 Cr 예측을 형성하기 위해 원래의 Cr 인트라 예측에 가중 재구성된 Cb 잔차를 더함으로써 구현될 수 있다:

수학식 4

$$\text{pred}_{cr}^*(i, j) = \text{pred}_{cr}(i, j) + \alpha \cdot \text{resi}_{cb}'(i, j).$$

[0038]

[0039] CCLM 루마-대-크로마 예측 모드는 하나의 추가적인 크로마 인트라 예측 모드로서 추가될 수 있다. 인코더 측에서, 크로마 인트라 예측 모드를 선택하기 위해 크로마 성분들에 대한 하나 더 RD 비용 검사가 추가될 수 있다. CCLM 루마-대-크로마 예측 모드 이외의 cb 인트라 예측 모드들이 CU의 크로마 성분들에 대해 사용될 수 있는 경우, CCLM Cb-대-Cr 예측이 Cr 성분 예측에 대해 사용될 수 있다.

[0040] 다중 모델 CCLM(MMLM)은 CCLM의 또 다른 확장이다. 명칭에 의해 표시된 바와 같이, MMLM에는 하나보다 많은 모델이 있을 수 있고, 예를 들어, 2개의 모델이 이용될 수 있다. MMLM에서, 현재 블록의 이웃 루마 샘플들 및 이웃 크로마 샘플들은 2개의 그룹으로 분류될 수 있고, 각각의 그룹은 선형 모델을 도출하기 위한 훈련 세트로서 사용될 수 있다(즉, 특정 α 및 β 가 특정 그룹에 대해 도출될 수 있다). 더욱이, 현재 루마 블록의 샘플들도 이웃 루마 샘플들의 분류에 대한 동일한 규칙에 기초하여 분류될 수 있다.

[0041] 도 8은 이웃 샘플들을 2개의 그룹으로 분류하는 예를 도시한다. 임계값을 이웃하는 재구성된 루마 샘플들의 평균값으로서 계산될 수 있다. $Rec'_L[x,y] \leq$ 임계값을 갖는 이웃 샘플은 그룹 1로 분류될 수 있는 한편; $Rec'_L[x,y] >$ 임계값을 갖는 이웃 샘플은 그룹 2로 분류될 수 있다:

수학식 5

$$\begin{cases} Pred_C[x,y] = \alpha_1 \times Rec'_L[x,y] + \beta_1 & \text{만일 } Rec'_L[x,y] \leq \text{임계값} \\ Pred_C[x,y] = \alpha_2 \times Rec'_L[x,y] + \beta_2 & \text{만일 } Rec'_L[x,y] > \text{임계값} \end{cases}$$

[0042]

[0043] 아핀 모션 보상된 예측이 이제 설명될 것이다. HEVC에서, 변환 모션 모델이 모션 보상 예측(motion compensation prediction, MCP)에 적용될 수 있다. 그러나, 많은 종류의 모션, 예를 들어, 줌 인/아웃, 회전, 원근 모션 및 다른 불규칙한 모션이 있을 수 있다. VTM4에서, 블록 기반 아핀 변환 모션 보상 예측이 적용될 수 있다. 도 9에 도시된 바와 같이, 블록의 아핀 모션 필드는 2개의 컨트롤 포인트 모션 벡터(4-파라미터) 또는 3개의 컨트롤 포인트 모션 벡터(6-파라미터)의 모션 정보에 의해 기술될 수 있다.

[0044] 4-파라미터 아핀 모션 모델에 대해, 블록 내의 샘플 위치 (x, y)에서의 모션 벡터는 다음과 같이 도출될 수 있다:

수학식 6

$$\begin{cases} mv_x = \frac{mv_{1x}-mv_{0x}}{W}x + \frac{mv_{1y}-mv_{0y}}{W}y + mv_{0x} \\ mv_y = \frac{mv_{1y}-mv_{0y}}{W}x + \frac{mv_{1x}-mv_{0x}}{W}y + mv_{0y} \end{cases}$$

[0045]

[0046] 6-파라미터 아핀 모션 모델에 대해, 블록 내의 샘플 위치 (x, y)에서의 모션 벡터는 다음과 같이 도출될 수 있다:

수학식 7

$$\begin{cases} mv_x = \frac{mv_{1x}-mv_{0x}}{W}x + \frac{mv_{2x}-mv_{0x}}{H}y + mv_{0x} \\ mv_y = \frac{mv_{1y}-mv_{0y}}{W}x + \frac{mv_{2y}-mv_{0y}}{H}y + mv_{0y} \end{cases}$$

[0047]

[0048] 여기서, mv_{0x} , mv_{0y} 는 상단-좌측 코너 컨트롤 포인트의 모션 벡터이고, mv_{1x} , mv_{1y} 는 상단-우측 코너 컨트롤 포인트의 모션 벡터이고, mv_{2x} , mv_{2y} 는 하단-좌측 코너 컨트롤 포인트의 모션 벡터이다.

[0049] 모션 보상 예측을 단순화하기 위해, 블록 기반 아핀 변환 예측이 적용될 수 있다. 각각의 4x4 루마 서브-블록의 모션 벡터들을 도출하기 위해, 도 10에 도시된 바와 같이, 각각의 서브-블록의 중심 샘플의 모션 벡터는 상기 수학식들에 따라 계산될 수 있고, 1/16 분수 정확도로 라운딩될 수 있다. 이어서, 도출된 모션 벡터로 각각의 서브-블록의 예측을 생성하기 위해 모션 보상 보간 필터들이 적용될 수 있다. 크로마-성분들의 서브-블록 크기는 또한 4x4에 설정될 수 있다. 4x4 크로마 서브-블록의 MV는 4개의 대응하는 4x4 루마 서브-블록들의 MV들의 평균으로서 계산될 수 있다. 병진 모션 인터 예측에 대해 행해지는 바와 같이, 2개의 아핀 모션 인터 예측 모드: 아핀 병합 모드 및 아핀 AMVP 모드가 또한 존재할 수 있다.

[0050] 아핀 병합 예측이 이제 설명될 것이다. AF_MERGE 모드는 8 이상의 폭 및 높이 둘 다를 갖는 CU들에 대해 적용될 수 있다. 이 모드에서, 현재 CU의 CPMV들은 공간적 이웃 CU들의 모션 정보에 기초하여 생성될 수 있다. 최대 5개의 CPMVP 후보가 존재할 수 있고, 현재 CU에 대해 사용될 후보를 표시하기 위해 인덱스가 시그널링될 수 있다. 아핀 병합 후보 리스트를 형성하기 위해 다음의 3가지 타입의 CPMV 후보가 사용될 수 있다:

- [0051] (1) 이웃 CU들의 CPMV들로부터 외삽된 상속된 아핀 병합 후보들;
- [0052] (2) 구성된 아핀 병합 후보들 CPMVP들 - 이들은 이웃 CU들의 병진 MV들을 이용하여 도출될 수 있음 -; 및
- [0053] (3) 제로 MV들.
- [0054] VTM4에서는, 최대 2개의 상속된 아핀 후보가 존재할 수 있으며, 이것들은 이웃 블록들의 아핀 모션 모델, 즉 좌측 이웃 CU들로부터의 하나 및 상부 이웃 CU들로부터의 하나로부터 도출될 수 있다. 후보 블록들은 도 11에 도시되어 있다. 좌측 예측기의 경우, 스캔 순서는 A0->A1일 수 있고, 상부 예측기의 경우, 스캔 순서는 B0->B1->B2일 수 있다. 각각의 측으로부터의 제1 상속된 후보가 선택될 수 있다. 2개의 상속된 후보 사이에 어떤 프루닝(pruning) 체크도 수행될 필요가 없다. 이웃 아핀 CU가 식별될 때, 그것의 컨트롤 포인트 모션 벡터들은 현재 CU의 아핀 병합 리스트에서 CPMVP 후보를 도출하기 위해 이용될 수 있다. 도 12에 도시된 바와 같이, 이웃 좌측 하단 블록 A가 아핀 모드에서 코딩되면, 블록 A를 포함하는 CU의 상단 좌측 코너, 상부 우측 코너 및 좌측 하단 코너의 모션 벡터들 v_2 , v_3 및 v_4 가 획득될 수 있다. 블록 A가 4-파라미터 아핀 모델로 코딩될 때, 현재 CU의 2개의 CPMV는 v_2 , 및 v_3 에 따라 계산될 수 있다. 블록 A가 6-파라미터 아핀 모델로 코딩될 수 있는 경우, 현재 CU의 3개의 CPMV가 v_2 , v_3 및 v_4 에 따라 계산될 수 있다.
- [0055] 구성된 아핀 후보는 후보가 각각의 컨트롤 포인트의 이웃 병진 모션 정보를 조합함으로써 구성될 수 있다는 것을 의미한다. 컨트롤 포인트들에 대한 모션 정보는 도 13에 도시된 지정된 공간적 이웃 및 시간적 이웃으로부터 도출될 수 있다. CPMV_k (k=1, 2, 3, 4)는 k번째 컨트롤 포인트를 나타낸다. CPMV1에 대해, B2->B3->A2 블록들이 체크될 수 있고 제1 이용가능한 블록의 MV가 사용될 수 있다. CPMV2에 대해, B1->B0 블록들이 체크될 수 있고, CPMV3에 대해, A1->A0 블록들이 체크될 수 있다. TMVP에 대해, 이것이 이용가능한 경우에 CPMV4로서 이용된다.
- [0056] 4개의 컨트롤 포인트의 MV들이 획득된 후에, 아핀 병합 후보들이 해당 모션 정보에 기초하여 구성될 수 있다. 컨트롤 포인트 MV들의 다음 조합들이 순서대로 구성하기 위해 사용될 수 있다: {CPMV₁, CPMV₂, CPMV₃}, {CPMV₁, CPMV₂, CPMV₄}, {CPMV₁, CPMV₃, CPMV₄}, {CPMV₂, CPMV₃, CPMV₄}, {CPMV₁, CPMV₂}, 및 {CPMV₁, CPMV₃}
- [0057] 3개의 CPMV의 조합은 6-파라미터 아핀 병합 후보를 구성하고, 2개의 CPMV의 조합은 4-파라미터 아핀 병합 후보를 구성한다. 모션 스케일링 프로세스를 회피하기 위해, 컨트롤 포인트들의 참조 인덱스들이 상이한 경우, 컨트롤 포인트 MV들의 관련 조합이 폐기될 수 있다.
- [0058] 상속된 아핀 병합 후보들 및 구성된 아핀 병합 후보들이 체크될 수 있고, 리스트가 여전히 가득 차 있지 않은 경우, 제로 MV들이 리스트의 끝에 삽입될 수 있다.
- [0059] 아핀 AMVP 예측이 이제 설명될 것이다. 아핀 AMVP 모드는 16 이상의 폭 및 높이 둘 다를 갖는 CU들에 대하여 적용될 수 있다. CU 레벨에서의 아핀 플래그는 아핀 AMVP 모드가 이용될 수 있는지를 표시하기 위하여 비트스트림에서 시그널링될 수 있고, 또 다른 플래그는 4-파라미터 아핀 또는 6-파라미터 아핀이 이용될 것인지를 표시하기 위하여 시그널링될 수 있다. 이 모드에서, 현재 CU의 CPMV들과 그들의 예측기 CPMVP들의 차이가 비트스트림에서 시그널링될 수 있다. 아핀 AMVP 후보 리스트 크기는 2일 수 있고, 이것은 다음과 같은 4가지 타입의 CPMV 후보를 순서대로 사용하여 생성될 수 있다:
- [0060] (1) 이웃 CU들의 CPMV들로부터 외삽될 수 있는 상속된 아핀 AMVP 후보들;
- [0061] (2) 구성된 아핀 AMVP 후보들 CPMVP들 - 이들은 이웃 CU들의 병진 MV들을 이용하여 도출될 수 있음 -;
- [0062] (3) 이웃 CU들로부터의 병진 MV들; 및
- [0063] (4) 제로 MV들
- [0064] 상속된 아핀 AMVP 후보들의 체크 순서는 상속된 아핀 병합 후보들의 체크 순서와 동일할 수 있다. 차이점은 AMVP 후보에 대해, 현재 블록에서와 동일한 참조 픽처를 갖는 아핀 CU가 고려될 수 있다는 것이다. 상속된 아핀 모션 예측기를 후보 리스트에 삽입할 때 어떤 프루닝 프로세스도 적용될 필요가 없다.
- [0065] 구성된 AMVP 후보는 도 13에 도시된 특정된 공간적 이웃들로부터 도출될 수 있다. 동일한 체크 순서가 아핀 병합 후보 구성에서 수행되는 바와 같이 사용될 수 있다. 또한, 이웃 블록의 참조 픽처 인덱스가 또한 체크될 수 있다. 인터 코딩될 수 있고 현재 CU들에서와 동일한 참조 픽처를 갖는 체크 순서에서의 제1 블록이 사용될 수

있다. 현재 CU가 4-파라미터 아핀 모드로 코딩되고, mv_0 및 mv_1 이 둘 다 이용가능할 수 있는 때, 이들은 아핀 AMVP 리스트에서 하나의 후보로서 추가될 수 있다. 현재 CU가 6-파라미터 아핀 모드로 코딩되고, 3개의 CPMV 모두가 이용가능할 때, 이들은 아핀 AMVP 리스트에서 하나의 후보로서 추가될 수 있다. 그렇지 않으면, 구성된 AMVP 후보는 이용가능하지 않은 것으로 설정된다.

[0066] 상속된 아핀 AMVP 후보들 및 구성된 AMVP 후보가 체크된 후에 아핀 AMVP 리스트 후보들이 여전히 2 미만이라면, mv_0 , mv_1 , 및 mv_2 가, 이용가능할 때, 현재 CU의 모든 컨트롤 포인트 MV들을 예측하기 위해 병진 MV들로서 순서대로 추가될 것이다. 마지막으로, 제로 MV들은 아핀 AMVP 리스트가 여전히 가득 차 있지 않은 경우에 그것을 채우기 위해 사용될 수 있다.

[0067] 아핀 모션 정보 스토리지가 이제 설명될 것이다. VTM4에서, 아핀 CU들의 CPMV들은 별개의 버퍼에 저장될 수 있다. 저장된 CPMV들은 최근에 코딩된 CU들에 대한 아핀 병합 모드 및 아핀 AMVP 모드에서 상속된 CPMV들을 생성하기 위하여 이용될 수 있다. CPMV들로부터 도출된 서브-블록 MV들은 모션 보상, 병진 MV들의 병합/AMVP 리스트의 MV 도출, 및 디블로킹을 위하여 이용될 수 있다.

[0068] 추가적인 CPMV들에 대한 픽처 라인 버퍼를 회피하기 위해, 상부 CTU로부터의 CU들로부터의 아핀 모션 데이터 상속(affine motion data inheritance)은 정상 이웃 CU들로부터의 상속(inheritance)과 상이하게 취급될 수 있다. 아핀 모션 데이터 상속을 위한 후보 CU가 상부 CTU 라인에 있다면, CPMV들 대신에 라인 버퍼에서의 하단-좌측 및 하단-우측 서브-블록 MV들이 아핀 MVP 도출을 위해 사용될 수 있다. 이러한 방식으로, CPMV들은 로컬 버퍼에 저장될 수 있다. 후보 CU가 6-파라미터 아핀 코딩된다면, 아핀 모델은 4-파라미터 모델로 열화될 수 있다. 도 14에 도시된 바와 같이, 상단 CTU 경계를 따라, CU의 하단-좌측 및 하단-우측 서브-블록 모션 벡터들은 하단 CTU들에서의 CU들의 아핀 상속을 위해 사용될 수 있다.

[0069] 전술한 진보에도 불구하고, VTM-4.0에서, 아핀 코딩된 코딩 블록 내의 4x4 크로마 서브-블록의 MV는 4개의 대응하는 4x4 루마 서브-블록의 MV들의 평균으로서 계산된다. 그러나, 크로마 4:4:4 및 4:2:2 포맷들의 경우 - 각각의 4x4 크로마 서브-블록은 단지 하나 또는 2개의 4x4 루마 서브-블록과 연관됨 -, 4x4 크로마 성분에 대한 MV 도출의 현재 스킴은 크로마 4:4:4 및 4:2:2 포맷들의 경우들을 수용하기 위한 개선의 여지를 남겨둔다.

발명의 내용

과제의 해결 수단

[0070] 본 개시내용의 양태에 따르면, 비디오 시퀀스를 인코딩 또는 디코딩하기 위한 방법은: 4:4:4 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩하는 단계, 또는 4:2:2 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩하는 단계를 포함할 수 있는데, 여기서 4:4:4 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩할 때, 이 방법은 평균화 연산 이외의 연산을 이용하여 하나의 4x4 크로마 블록의 아핀 모션 벡터를 복사하는 단계를 추가로 포함할 수 있고, 4:2:2 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩할 때, 이 방법은 각각의 4x4 크로마 블록을 2개의 4x4 동위치(co-located) 크로마 블록과 연관시켜서 하나의 4x4 크로마 블록의 아핀 모션 벡터가 2개의 동위치 크로마 블록의 모션 벡터들의 평균이도록 하는 단계를 추가로 포함할 수 있다.

[0071] 본 개시내용의 양태에 따르면, 방법은, 크로마 포맷에 관계없이, 현재의 4x4 크로마 블록을 4개의 2x2 서브-블록으로 분할하는 단계, 상단-좌측 2x2 크로마 서브-블록에 대한 동위치 루마 블록의 제1 아핀 모션 벡터를 도출하는 단계, 하단-우측 2x2 크로마 블록에 대한 동위치 루마 블록의 제2 아핀 모션 벡터를 도출하는 단계, 및 제1 아핀 모션 벡터와 제2 아핀 모션 벡터의 평균을 사용하여 현재의 4x4 크로마 블록의 아핀 모션 벡터를 도출하는 단계를 추가로 포함할 수 있다.

[0072] 본 개시내용의 양태에 따르면, 방법은 루마 성분과 크로마 성분 사이의 모션 보상을 위해 사용되는 보간 필터를 정렬하는 단계를 포함할 수 있다.

[0073] 본 개시내용의 이런 양태에 따르면, 비디오 시퀀스가 4:2:0 크로마 포맷을 사용하여 입력될 때, 방법은 루마 성분들 및 크로마 성분들에 대해 8-탭 보간 필터를 적용하는 단계를 추가로 포함할 수 있다.

[0074] 본 개시내용의 양태에 따르면, 방법은: 성분들 Y, Cb, 및 Cr을 3개의 개별 트리로서 코딩하는 단계를 추가로 포함할 수 있고, 3개의 개별 트리 중 각각의 트리는 컴포넌트들 Y, Cb, 및 Cr 중 하나의 성분을 코딩한다.

- [0075] 본 개시내용의 이런 양태에 따르면, 3개의 개별 트리로서의 코딩은 I 슬라이스 또는 I 타일 그룹에 대해 수행될 수 있다.
- [0076] 본 개시내용의 양태에 따르면, 최대 허용 변환 크기는 상이한 컬러 성분들에 대해 동일할 수 있다.
- [0077] 본 개시내용의 이런 양태에 따르면, 4:2:2 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩할 때, 최대 수직 크기는 상이한 컬러 성분들 중에서 동일할 수 있고, 크로마 성분들에 대한 최대 수평 변환 크기는 루마 성분들에 대한 최대 수평 변환 크기의 절반일 수 있다.
- [0078] 본 개시내용의 양태에 따르면, PDPC(Position-Dependent Predictor combination), MTS(Multiple Transform Selection), NSST(Non-Separable Secondary Transform), ISP(Intra-Sub Partitioning), 및 MRL(Multiple reference line) 인트라 예측 중 적어도 하나가 루마 성분 및 크로마 성분 둘 다에 적용될 수 있다.
- [0079] 본 개시내용의 이런 양태에 따르면, 다중 참조 라인(Multiple reference line, MRL) 인트라 예측이 루마 성분 및 크로마 성분 둘 다에 적용될 때, 그리고 비디오 시퀀스를 인코딩 또는 디코딩하는 것이 4:4:4 크로마 포맷을 사용하여 수행될 때, 방법은 인트라 예측을 위해 N번째 참조를 선택하고, 크로마 성분들에 대한 명시적 시그널링 없이 동일한 참조 라인을 사용하는 단계를 추가로 포함할 수 있다; 인트라-서브 파티셔닝(Intra-Sub Partitioning, ISP)이 루마 성분 및 크로마 성분 둘 다에 적용될 때, 방법은 성분들 Y, Cb, 및 Cr에 대한 현재 블록에 대한 블록 레벨에서 인트라-서브 파티셔닝(Intra-Sub Partitioning, ISP)을 적용하는 단계를 추가로 포함할 수 있고; 그리고 상이한 트리들이 상이한 컬러 성분들에 대해 사용될 때, 방법은 시그널링 없이 병치된 Y 성분들로부터 U 및 V 성분들에 대한 코딩 파라미터들을 암시적으로 도출하는 단계를 추가로 포함할 수 있다.
- [0080] 본 개시내용의 양태에 따르면, 비디오 시퀀스를 인코딩 또는 디코딩하기 위한 디바이스는: 프로그램 코드를 저장하도록 구성된 적어도 하나의 메모리; 및 프로그램 코드를 판독하고 프로그램 코드에 의해 지시된 바와 같이 동작하도록 구성된 적어도 하나의 프로세서를 포함할 수 있고, 프로그램 코드는: 적어도 하나의 프로세서로 하여금 4:4:4 크로마 포맷을 사용하여 비디오 시퀀스를 인코딩 또는 디코딩하게 야기하거나, 4:2:2 크로마 포맷을 사용하여 비디오 시퀀스를 인코딩 또는 디코딩하게 야기하도록 구성된 제1 인코딩 또는 디코딩 코드를 포함하고, 여기서 제1 인코딩 또는 디코딩 코드가 적어도 하나의 프로세서로 하여금 4:4:4 크로마 포맷을 사용하여 비디오 시퀀스를 인코딩 또는 디코딩하게 야기하도록 구성될 때, 제1 인코딩 또는 디코딩 코드는 적어도 하나의 프로세서로 하여금 평균화 연산 이외의 연산을 사용하여 하나의 4x4 크로마 블록의 아핀 모션 벡터를 복사하게 야기하도록 구성된 코드를 추가로 포함할 수 있고, 제1 인코딩 또는 디코딩 코드가 적어도 하나의 프로세서로 하여금 4:2:2 크로마 포맷을 사용하여 비디오 시퀀스를 인코딩 또는 디코딩하게 야기하도록 구성될 때, 제1 인코딩 또는 디코딩 코드는 적어도 하나의 프로세서로 하여금 각각의 4x4 크로마 블록을 2개의 4x4 동위치 크로마 블록과 연관시켜 하나의 4x4 크로마 블록의 아핀 모션 벡터가 2개의 동위치 크로마 블록의 모션 벡터들의 평균이도록 야기하게 구성된 코드를 추가로 포함할 수 있다.
- [0081] 본 개시내용의 양태에 따르면, 제1 인코딩 또는 디코딩 코드는 적어도 하나의 프로세서로 하여금 현재의 4x4 크로마 블록을 4개의 2x2 서브-블록으로 분할하고, 상단-좌측 2x2 크로마 서브-블록에 대해 동위치 루마 블록의 제1 아핀 모션 벡터를 도출하고, 하단-우측 2x2 크로마 블록에 대해 동위치 루마 블록의 제2 아핀 모션 벡터를 도출하고, 제1 아핀 모션 벡터 및 제2 아핀 모션 벡터의 평균을 사용하여 현재 4x4 크로마 블록의 아핀 모션 벡터를 도출하게 야기하도록 구성된 코드를 추가로 포함할 수 있다.
- [0082] 본 개시내용의 양태에 따르면, 제1 인코딩 또는 디코딩 코드는 적어도 하나의 프로세서로 하여금 루마 성분과 크로마 성분 사이의 모션 보상을 위해 사용되는 보간 필터를 정렬하게 야기하도록 구성된 코드를 추가로 포함할 수 있다.
- [0083] 본 개시내용의 이런 양태에 따르면, 제1 인코딩 또는 디코딩 코드가 적어도 하나의 프로세서로 하여금 4:2:2 크로마 포맷을 사용하여 비디오 시퀀스를 인코딩 또는 디코딩하게 야기하도록 구성될 때, 제1 인코딩 또는 디코딩 코드는 적어도 하나의 프로세서로 하여금 루마 성분들 및 크로마 성분들에 대해 8-탭 보간 필터를 적용하게 야기하도록 구성된 코드를 추가로 포함할 수 있다.
- [0084] 본 개시내용의 양태에 따르면, 제1 인코딩 또는 디코딩 코드는 적어도 하나의 프로세서로 하여금 성분들 Y, Cb, 및 Cr을 3개의 개별 트리로서 코딩하게 야기하도록 구성된 코드를 추가로 포함할 수 있고, 3개의 개별 트리 중 각각의 트리는 성분들 Y, Cb, 및 Cr 중 하나의 성분을 코딩한다.
- [0085] 본 개시내용의 이런 양태에 따르면, 3개의 개별 트리로서 코딩하는 구성은 I 슬라이스 또는 I 타일 그룹에 대해

수행되도록 구성될 수 있다.

- [0086] 본 개시내용의 양태에 따르면, 제1 인코딩 또는 디코딩 코드는 적어도 하나의 프로세서로 하여금 최대 변환 크기가 상이한 컬러 성분들에 대해 동일하도록 허용하게 야기하도록 구성된 코드를 추가로 포함할 수 있다.
- [0087] 본 개시내용의 이런 양태에 따르면, 제1 인코딩 또는 디코딩 코드가 적어도 하나의 프로세서로 하여금 4:2:2 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩하게 야기하도록 구성될 때, 제1 인코딩 또는 디코딩 코드는 적어도 하나의 프로세서로 하여금 최대 수직 크기를 상이한 컬러 성분들 중에서 동일하게 설정하고, 크로마 성분들에 대한 최대 수평 변환 크기를 루마 성분들에 대한 최대 수평 변환 크기의 절반으로 설정하게 야기하도록 구성된 코드를 추가로 포함할 수 있다.
- [0088] 본 개시내용의 양태에 따르면, 제1 인코딩 또는 디코딩 코드는 적어도 하나의 프로세서로 하여금, PDP(Position-Dependent Predictor combination), MTS(Multiple Transform Selection), NSST(Non-Separable Secondary Transform), ISP(Intra-Sub Partitioning), 및 MRL(Multiple reference line) 인트라 예측 중 적어도 하나를 루마 성분 및 크로마 성분 둘 다에 적용하게 야기하도록 구성된 코드를 추가로 포함할 수 있다.
- [0089] 본 개시내용의 양태에 따르면, 명령어들을 저장하는 비일시적 컴퓨터 판독가능 매체가 제공될 수 있고, 명령어들은: 디바이스의 하나 이상의 프로세서에 의해 실행될 때, 하나 이상의 프로세서로 하여금 4:4:4 크로마 포맷을 사용하여 비디오 시퀀스를 인코딩 또는 디코딩하거나, 또는 4:2:2 크로마 포맷을 사용하여 비디오 시퀀스를 인코딩 또는 디코딩하게 야기하는 하나 이상의 명령어를 포함하고, 여기서 명령어들은, 디바이스의 하나 이상의 프로세서에 의해 실행될 때, 하나 이상의 프로세서로 하여금 4:4:4 크로마 포맷을 사용하여 비디오 시퀀스를 인코딩 또는 디코딩하게 야기하고, 명령어들은, 디바이스의 하나 이상의 프로세서에 의해 실행될 때, 하나 이상의 프로세서로 하여금 평균화 연산 이외의 연산을 사용하여 하나의 4x4 크로마 블록의 아핀 모션 벡터를 복사하게 추가로 야기하고, 여기서 명령어들은, 디바이스의 하나 이상의 프로세서에 의해 실행될 때, 하나 이상의 프로세서로 하여금 4:2:2 크로마 포맷을 사용하여 비디오 시퀀스를 인코딩 또는 디코딩하도록 야기하고, 명령어들은, 디바이스의 하나 이상의 프로세서에 의해 실행될 때, 하나 이상의 프로세서로 하여금, 각각의 4x4 크로마 블록을 2개의 4x4 동위치 크로마 블록과 연관시켜 하나의 4x4 크로마 블록의 아핀 모션 벡터가 2개의 동위치 크로마 블록의 모션 벡터들의 평균이도록 추가로 야기한다.
- [0090] 앞서 설명된 방법들, 디바이스들, 및 비일시적 컴퓨터 판독가능 매체들이 개별적으로 설명되었지만, 이 설명들은 그 사용 또는 기능성의 범위에 대한 어떠한 제한도 암시하도록 의도되지 않는다. 실제로, 이러한 방법들, 디바이스들, 및 비일시적 컴퓨터 판독가능 매체들은 본 개시내용의 다른 양태들에서 조합될 수 있다.

도면의 간단한 설명

- [0091] 개시된 주제의 추가의 특징들, 본질, 및 다양한 이점들이 다음의 상세한 설명 및 첨부 도면들로부터 더 명백할 것이다.
- 도 1a는 일 실시예에 따른 파티셔닝된 코딩 트리 유닛의 도면이다.
- 도 1b는 일 실시예에 따른 코딩 트리 유닛의 도면이다.
- 도 2a는 일 실시예에 따른 코딩 트리 유닛의 도면이다.
- 도 2b는 일 실시예에 따른 코딩 트리 유닛의 도면이다.
- 도 3은 일 실시예에 따른 상이한 YUV 포맷들의 도면이다.
- 도 4는 일 실시예에 따른 상이한 루마 값들의 도면이다.
- 도 5a는 일 실시예에 따른 교차 성분 선형 모델링에 사용되는 샘플들의 도면이다.
- 도 5b는 일 실시예에 따른 교차 성분 선형 모델링에 사용되는 샘플들의 도면이다.
- 도 6a는 일 실시예에 따른 교차 성분 선형 모델링에 사용되는 샘플들의 도면이다.
- 도 6b는 일 실시예에 따른 교차 성분 선형 모델링에 사용되는 샘플들의 도면이다.
- 도 7a는 일 실시예에 따른 교차 성분 선형 모델링에 사용되는 샘플들의 도면이다.
- 도 7b는 일 실시예에 따른 교차 성분 선형 모델링에 사용되는 샘플들의 도면이다.

- 도 8은 일 실시예에 따른 다중 모델 CCLM을 이용한 분류의 예이다.
- 도 9a는 일 실시예에 따른 블록의 아핀 모션 필드의 예이다.
- 도 9b는 일 실시예에 따른 블록의 아핀 모션 필드의 예이다.
- 도 10은 일 실시예에 따른 아핀 모션 벡터 필드의 예이다.
- 도 11은 일 실시예에 따른 예측을 위한 후보 블록들의 예이다.
- 도 12는 일 실시예에 따른 예측을 위한 후보 블록들의 예이다.
- 도 13은 일 실시예에 따른 예측을 위한 후보 블록들의 예이다.
- 도 14는 일 실시예에 따른 모션 벡터 사용의 예이다.
- 도 15는 일 실시예에 따른 통신 시스템의 단순화된 블록도이다.
- 도 16은 일 실시예에 따른 스트리밍 환경의 도면이다.
- 도 17은 일 실시예에 따른 비디오 디코더의 블록도이다.
- 도 18은 일 실시예에 따른 비디오 인코더의 블록도이다.
- 도 19는 일 실시예에 따른, 비디오 시퀀스를 인코딩 또는 디코딩하는 예시적인 프로세스의 흐름도이다.
- 도 20은 일 실시예에 따른 컴퓨터 시스템의 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0092] 도 15는 본 개시내용의 실시예에 따른 통신 시스템(400)의 단순화된 블록도를 도시한다. 통신 시스템(400)은 네트워크(450)를 통해 상호접속된 적어도 2개의 단말(410-420)을 포함할 수 있다. 데이터의 단방향 송신을 위해, 제1 단말(410)은 네트워크(450)를 통해 다른 단말(420)로의 송신을 위해 로컬 위치에서 비디오 데이터를 코딩할 수 있다. 제2 단말(420)은 네트워크(450)로부터 다른 단말의 코딩된 비디오 데이터를 수신하고, 코딩된 데이터를 디코딩하고 복구된 비디오 데이터를 디스플레이할 수 있다. 단방향 데이터 송신은 미디어 서빙 응용들(media serving applications) 등에서 흔한 것일 수 있다.
- [0093] 도 15는 예를 들어, 영상 회의 동안 발생할 수 있는 코딩된 비디오의 양방향 송신을 지원하기 위해 제공되는 제2 쌍의 단말들(430, 440)을 도시한다. 데이터의 양방향 송신을 위해, 각각의 단말(430, 440)은 네트워크(450)를 통해 다른 단말로의 송신을 위해 로컬 위치에서 캡처된 비디오 데이터를 코딩할 수 있다. 각각의 단말(430, 440)은 또한 다른 단말에 의해 송신된 코딩된 비디오 데이터를 수신할 수 있고, 코딩된 데이터를 디코딩할 수 있고, 복구된 비디오 데이터를 로컬 디스플레이 디바이스에 디스플레이할 수 있다.
- [0094] 도 15에서, 단말들(410-440)은 서버들, 개인용 컴퓨터들 및 스마트 폰들로서 예시될 수 있지만, 본 개시내용의 원리들은 그렇게만 제한되지는 않는다. 본 개시내용의 실시예들은 랩톱 컴퓨터들, 태블릿 컴퓨터들, 미디어 플레이어들 및/또는 전용 영상 회의 장비를 이용한 응용을 찾는다. 네트워크(450)는 예를 들어, 와이어라인 및/또는 무선 통신 네트워크들을 포함하여, 단말들(410-440) 사이에 코딩된 비디오 데이터를 운반하는 임의의 수의 네트워크를 나타낸다. 통신 네트워크(450)는 회선 교환 및/또는 패킷 교환 채널들에서 데이터를 교환할 수 있다. 대표적인 네트워크들은 원격통신(telecommunications) 네트워크들, 로컬 영역 네트워크들, 광역 네트워크들 및/또는 인터넷을 포함한다. 본 논의의 목적을 위해, 네트워크(450)의 아키텍처 및 토폴로지는 아래에서 본 명세서에서 설명되지 않는 한 본 개시내용의 동작에 중요하지 않을 수 있다.
- [0095] 도 16은, 개시된 주제를 위한 응용에 대한 예로서, 스트리밍 환경에서의 비디오 인코더 및 비디오 디코더의 배치를 도시한다. 개시된 주제는, 예를 들어, 영상 회의, 디지털 TV, CD, DVD, 메모리 스틱 등을 포함하는 디지털 미디어상의 압축된 비디오의 저장 등을 포함하여, 다른 비디오 인에이블 응용들에 동등하게 적용가능할 수 있다.
- [0096] 스트리밍 시스템은, 예를 들어, 압축되지 않은 비디오 샘플 스트림(502)을 생성하는 비디오 소스(501), 예를 들어, 디지털 카메라를 포함할 수 있는 캡처 서브시스템(513)을 포함할 수 있다. 인코딩된 비디오 비트스트림들과 비교할 때 많은 데이터 양을 강조하기 위해 굵은 선으로 묘사된 샘플 스트림(502)은 카메라(501)에 결합된 인코더(503)에 의해 처리될 수 있다. 인코더(503)는 아래에서 더 상세히 설명되는 바와 같이 개시된 주제의 양

태들을 가능하게 하거나 구현하기 위해 하드웨어, 소프트웨어, 또는 이들의 조합을 포함할 수 있다. 샘플 스트림과 비교할 때 적은 데이터 양을 강조하기 위해 얇은 선으로서 묘사된 인코딩된 비디오 비트스트림(504)은 미래의 사용을 위해 스트리밍 서버(505)상에 저장될 수 있다. 하나 이상의 스트리밍 클라이언트(506, 508)는 스트리밍 서버(505)에 액세스하여 인코딩된 비디오 비트스트림(504)의 사본들(507, 509)을 검색할 수 있다. 클라이언트(506)는, 인코딩된 비디오 비트스트림(507)의 착신 사본을 디코딩하고 디스플레이(512) 또는 다른 렌더링 디바이스(묘사되지 않음)상에 렌더링될 수 있는 발신 비디오 샘플 스트림(511)을 생성하는 비디오 디코더(510)를 포함할 수 있다. 일부 스트리밍 시스템들에서, 비디오 비트스트림들(504, 507, 509)은 특정 비디오 코딩/압축 표준들에 따라 인코딩될 수 있다. 그 표준들의 예들은 H.265 HEVC를 포함한다. VVC(Versatile Video Coding)로서 비공식적으로 알려진 비디오 코딩 표준이 개발 중이다. 개시된 주제는 VVC의 맥락에서 사용될 수 있다.

[0097] 도 17은 본 발명의 실시예에 따른 비디오 디코더(510)의 기능 블록도일 수 있다.

[0098] 수신기(610)는 디코더(610)에 의해 디코딩될 하나 이상의 코덱 비디오 시퀀스를 수신할 수 있다; 동일한 또는 또 다른 실시예에서, 한 번에 하나의 코딩된 비디오 시퀀스가 수신될 수 있고, 여기서 각각의 코딩된 비디오 시퀀스의 디코딩은 다른 코딩된 비디오 시퀀스들과 독립적이다. 코딩된 비디오 시퀀스는, 인코딩된 비디오 데이터를 저장하는 저장 디바이스에 대한 하드웨어/소프트웨어 링크일 수 있는, 채널(612)로부터 수신될 수 있다. 수신기(610)는 인코딩된 비디오 데이터를 다른 데이터, 예를 들어, 코딩된 오디오 데이터 및/또는 보조 데이터 스트림들과 함께 수신할 수 있고, 이들은 그것들 각각의 사용 엔티티들(묘사되지 않음)에 포워딩될 수 있다. 수신기(610)는 코딩된 비디오 시퀀스를 다른 데이터로부터 분리할 수 있다. 네트워크 지터를 방지하기 위해, 수신기(610)와 엔트로피 디코더/파서(620)(이후 "파서") 사이 내에 버퍼 메모리(615)가 결합될 수 있다. 수신기(610)가 충분한 대역폭 및 제어가능성의 저장/포워드 디바이스로부터, 또는 동기식 네트워크(isosynchronous network)로부터 데이터를 수신하고 있을 때, 버퍼(615)는 필요하지 않을 수 있거나, 작을 수 있다. 인터넷과 같은 베스트 에포트 패킷 네트워크들상에서의 사용을 위해, 버퍼(615)는 요구될 수 있고, 비교적 클 수 있고, 유리하게는 적응적 크기일 수 있다.

[0099] 비디오 디코더(510)는 엔트로피 코딩된 비디오 시퀀스로부터 심벌들(621)을 재구성하기 위해 파서(620)를 포함할 수 있다. 해당 심벌들의 카테고리들은 디코더(510)의 동작을 관리하기 위해 사용되는 정보, 및 잠재적으로, 도 17에 도시된 바와 같이, 디코더의 일체 부분(integral part)은 아니지만 그것에 결합될 수 있는 디스플레이(512)와 같은 렌더링 디바이스를 제어하기 위한 정보를 포함한다. 렌더링 디바이스(들)에 대한 제어 정보는 SEI 메시지(Supplementary Enhancement Information) 또는 VUI(Video Usability Information) 파라미터 세트 프래그먼트들(묘사되지 않음)의 형식일 수 있다. 파서(620)는 수신되는 코딩된 비디오 시퀀스를 파싱/엔트로피 디코딩할 수 있다. 코딩된 비디오 시퀀스의 코딩은 비디오 코딩 기술 또는 표준에 따를 수 있고, 가변 길이 코딩, 허프만 코딩(Huffman coding), 컨텍스트 민감성(context sensitivity)을 갖거나 갖지 않는 산술 코딩 등을 포함하는, 본 기술분야의 통상의 기술자에게 잘 알려진 원리들을 따를 수 있다. 파서(620)는, 코딩된 비디오 시퀀스로부터, 그룹에 대응하는 적어도 하나의 파라미터에 기초하여, 비디오 디코더에서의 픽셀들의 서브그룹들 중 적어도 하나에 대한 서브그룹 파라미터들의 세트를 추출할 수 있다. 서브그룹들은 픽처 그룹들(Groups of Pictures, GOPs), 픽처들, 타일들, 슬라이스들, 매크로블록들, 코딩 유닛들(Coding Units, CUs), 블록들, 변환 유닛들(Transform Units, TUs), 예측 유닛들(Prediction Units, PUs) 등을 포함할 수 있다. 엔트로피 디코더/파서는 또한 코딩된 비디오 시퀀스로부터 변환 계수들, 양자화기 파라미터(QP) 값들, 모션 벡터들 등과 같은 정보를 추출할 수 있다.

[0100] 파서(620)는 버퍼(615)로부터 수신된 비디오 시퀀스에 대해 엔트로피 디코딩/파싱 동작을 수행하여, 심벌들(621)을 생성할 수 있다. 파서(620)는 인코딩된 데이터를 수신하고, 특정 심벌들(621)을 선택적으로 디코딩할 수 있다. 또한, 파서(620)는 특정 심벌들(621)이 모션 보상 예측 유닛(653), 스케일러/역변환 유닛(651), 인트라 예측 유닛(652), 또는 루프 필터(656)에 제공될지를 결정할 수 있다.

[0101] 심벌들(621)의 재구성은 코딩된 비디오 픽처 또는 그것의 부분들의 타입(예컨대: 인트라 및 인트라 픽처, 인트라 및 인트라 블록), 및 다른 인자들에 의존하여 다중의 상이한 유닛을 수반할 수 있다. 어느 유닛들이 수반되는지, 그리고 어떻게 되는지는 파서(620)에 의해 코딩된 비디오 시퀀스로부터 파싱된 서브그룹 제어 정보에 의해 제어될 수 있다. 파서(620)와 아래의 다중 유닛 사이의 그러한 서브그룹 제어 정보의 흐름은 명확성을 위해 묘사되어 있지 않다.

[0102] 이미 언급된 기능 블록들 이외에, 디코더(510)는 아래에 설명되는 바와 같이 개념적으로 다수의 기능 유닛으로

세분될 수 있다. 상업적 제약 하에서 동작하는 실제 구현에서, 이들 유닛 중 다수는 서로 밀접하게 상호작용하고, 적어도 부분적으로 서로 통합될 수 있다. 그러나, 개시된 주제를 설명하기 위한 목적으로, 아래의 기능 유닛들로의 개념적 세분이 적절하다.

- [0103] 제1 유닛은 스케일러/역변환 유닛(651)이다. 스케일러/역변환 유닛(651)은, 파서(620)로부터의 심벌(들)(621)로서, 어느 변환을 사용할지, 블록 크기, 양자화 인자, 양자화 스케일링 행렬들 등을 포함하여, 제어 정보뿐만 아니라 양자화된 변환 계수를 수신한다. 이것은 애그리게이터(aggregator)(655)에 입력될 수 있는 샘플 값들을 포함하는 블록들을 출력할 수 있다.
- [0104] 일부 경우들에서, 스케일러/역변환(651)의 출력 샘플들은 인트라 코딩된 블록에 관련될 수 있다; 즉, 이전에 재구성된 픽처들로부터의 예측 정보를 이용하는 것이 아니고, 현재 픽처의 이전에 재구성된 부분들로부터의 예측 정보를 이용할 수 있는 블록. 그러한 예측 정보는 인트라 픽처 예측 유닛(652)에 의해 제공될 수 있다. 일부 경우들에서, 인트라 픽처 예측 유닛(652)은 현재 (부분적으로 재구성된) 픽처(656)로부터 폐지된 주위의 이미 재구성된 정보를 사용하여, 재구성 중인 블록의 동일한 크기 및 형상의 블록을 생성한다. 애그리게이터(655)는, 일부 경우들에서, 샘플당 기준으로, 인트라 예측 유닛(652)이 생성한 예측 정보를 스케일러/역변환 유닛(651)에 의해 제공된 출력 샘플 정보에 더한다.
- [0105] 다른 경우들에서, 스케일러/역변환 유닛(651)의 출력 샘플들은 인터 코딩되고, 및 잠재적으로 모션 보상된 블록에 관련될 수 있다. 그러한 경우에, 모션 보상 예측 유닛(653)은 참조 픽처 메모리(657)에 액세스하여 예측을 위해 사용되는 샘플들을 폐지할 수 있다. 블록에 관련된 심벌들(621)에 따라 폐지된 샘플들을 모션 보당한 후에, 이들 샘플은 애그리게이터(655)에 의해 스케일러/역변환 유닛의 출력(이 경우 잔차 샘플들 또는 잔차 신호라고 불림)에 더해져서 출력 샘플 정보를 생성할 수 있다. 모션 보상 유닛이 예측 샘플들을 폐지하는 참조 픽처 메모리 내의 어드레스들은, 예를 들어, X, Y, 및 참조 픽처 컴포넌트들을 가질 수 있는 심벌들(621)의 형식으로 모션 보상 유닛에 이용가능 모션 벡터들에 의해 제어될 수 있다. 모션 보상은 또한 서브샘플 정확한 모션 벡터들이 사용 중일 때 참조 픽처 메모리로부터 폐지된 샘플 값들의 보간, 모션 벡터 예측 메커니즘 등을 포함할 수 있다.
- [0106] 애그리게이터(655)의 출력 샘플들은 루프 필터 유닛(656)에서의 다양한 루프 필터링 기법들을 겪을 수 있다. 비디오 압축 기술들은, 코딩된 비디오 비트스트림에 포함되고 파서(620)로부터의 심벌들(621)로서 루프 필터 유닛(656)에 이용가능하게 되는 파라미터들에 의해 제어되지만, 코딩된 픽처 또는 코딩된 비디오 시퀀스의 이전 (디코딩 순서로) 부분들의 디코딩 동안 획득된 메타-정보에 응답할 뿐만 아니라, 이전에 재구성된 및 루프-필터링된 샘플 값들에 응답할 수 있는 인-루프 필터(in-loop filter) 기술들을 포함할 수 있다.
- [0107] 루프 필터 유닛(656)의 출력은 렌더링 디바이스(512)에 출력될 뿐만 아니라 미래의 인터-픽처 예측에서 사용하기 위해 참조 픽처 메모리(656)에 저장될 수 있는 샘플 스트림일 수 있다.
- [0108] 특정 코딩된 픽처들은, 완전히 재구성되면, 미래 예측을 위한 참조 픽처들로서 사용될 수 있다. 일단 코딩된 픽처가 완전히 재구성되고 코딩된 픽처가 참조 픽처로서 식별되면(예를 들어, 파서(620)에 의해), 현재 참조 픽처(656)는 참조 픽처 버퍼(657)의 일부가 될 수 있고, 다음의 코딩된 픽처의 재구성을 개시하기 전에 새로운 현재 픽처 메모리가 재할당될 수 있다.
- [0109] 비디오 디코더(510)는 H.265 HEVC와 같은 표준에 문서화될 수 있는 미리 결정된 비디오 압축 기술에 따라 디코딩 동작들을 수행할 수 있다. 코딩된 비디오 시퀀스는, 이것이 비디오 압축 기술 문서 또는 표준 및 구체적으로 그 가운데 프로파일 문서에 특정된 대로 비디오 압축 기술 또는 표준의 선택스를 고수한다는 점에서, 사용되는 비디오 압축 기술 또는 표준에 의해 특정된 선택스를 따를 수 있다. 또한 준수를 위해 필요한 것은 코딩된 비디오 시퀀스의 복잡성이 비디오 압축 기술 또는 표준의 레벨에 의해 정의된 경계 내에 있는 것일 수 있다. 일부 경우들에서, 레벨들은 최대 픽처 크기, 최대 프레임 레이트, 최대 재구성 샘플 레이트(예를 들어, 초당 메가샘플수로 측정됨), 최대 참조 픽처 크기 등을 제한한다. 레벨들에 의해 설정된 한계들은, 일부 경우들에서, HRD(Hypothetical Reference Decoder) 사양들 및 코딩된 비디오 시퀀스에서 시그널링된 HRD 버퍼 관리를 위한 메타데이터를 통해 추가로 제한될 수 있다.
- [0110] 일 실시예에서, 수신기(610)는 인코딩된 비디오와 함께 추가적인(중복) 데이터를 수신할 수 있다. 이 추가적인 데이터는 코딩된 비디오 시퀀스(들)의 일부로서 포함될 수 있다. 이 추가적인 데이터는 데이터를 적절히 디코딩하고 및/또는 원래의 비디오 데이터를 더 정확하게 재구성하기 위해 비디오 디코더(510)에 의해 사용될 수 있다. 추가적인 데이터는 예를 들어, 시간, 공간, 또는 신호 대 잡음 비(SNR) 향상 계층들, 중복 슬라이스들, 중

복 픽처들, 순방향 에러 정정 코드들 등의 형식일 수 있다.

- [0111] 도 18은 본 개시내용의 실시예에 따른 비디오 인코더(503)의 기능 블록도일 수 있다.
- [0112] 인코더(503)는 인코더(503)에 의해 코딩될 비디오 이미지(들)를 캡처할 수 있는 비디오 소스(501)(인코더의 일부가 아님)로부터 비디오 샘플들을 수신할 수 있다.
- [0113] 비디오 소스(501)는, 임의의 적합한 비트 심도(예를 들어: 8 비트, 10 비트, 12 비트, ...), 임의의 색공간(예를 들어, BT.601Y CrCb, RGB, ...), 및 임의의 적합한 샘플링 구조(예를 들어, Y CrCb 4:2:0, Y CrCb 4:4:4)일 수 있는 디지털 비디오 샘플 스트림의 형식으로 인코더(503)에 의해 코딩될 소스 비디오 시퀀스를 제공할 수 있다. 미디어 서버 시스템에서, 비디오 소스(501)는 이전에 준비된 비디오를 저장하는 저장 디바이스일 수 있다. 영상 회의 시스템에서, 비디오 소스(503)는 비디오 시퀀스로서 로컬 이미지 정보를 캡처하는 카메라일 수 있다. 비디오 데이터는 순차적으로 볼 때 모션을 부여하는 복수의 개별 픽처로서 제공될 수 있다. 픽처들 자체는 픽셀들의 공간 어레이로서 조직될 수 있고, 여기서 각각의 픽셀은 사용 중인 샘플링 구조, 색 공간 등에 의존하여 하나 이상의 샘플을 포함할 수 있다. 본 기술분야의 통상의 기술자는 픽셀들과 샘플들 사이의 관계를 용이하게 이해할 수 있다. 이하의 설명은 샘플들에 초점을 맞춘다.
- [0114] 일 실시예에 따르면, 인코더(503)는 소스 비디오 시퀀스의 픽처들을 실시간으로 또는 응용에 의해 요구되는 임의의 다른 시간 제약들 하에서 코딩된 비디오 시퀀스(743)로 코딩 및 압축할 수 있다. 적절한 코딩 속도를 시행하는 것이 컨트롤러(750)의 하나의 기능이다. 컨트롤러(750)는 아래에 설명되는 바와 같이 다른 기능 유닛들을 제어하고, 이러한 유닛들에 기능적으로 결합된다. 결합은 명료성을 위해 묘사되지 않는다. 컨트롤러에 의해 설정된 파라미터들은 레이트 제어 관련 파라미터들(픽처 스킵, 양자화기, 레이트-왜곡 최적화 기법들의 램다 값들, ...), 픽처 크기, 픽처 그룹(GOP) 레이아웃, 최대 모션 벡터 검색 범위 등을 포함할 수 있다. 본 기술분야의 통상의 기술자는 컨트롤러(750)의 다른 기능들을 쉽게 식별할 수 있는데, 그 이유는 그것들이 특정 시스템 설계에 대해 최적화된 비디오 인코더(503)에 관한 것일 수 있기 때문이다.
- [0115] 일부 비디오 인코더들은 본 기술분야의 통상의 기술자가 "코딩 루프"로서 쉽게 인식하는 것에서 동작한다. 과도하게 단순화된 설명으로서, 코딩 루프는 인코더(730)(이후 "소스 코더")의 인코딩 부분(코딩될 입력 픽처, 및 참조 픽처(들)에 기초하여 심벌들을 생성하는 것을 담당함), 및 (원격) 디코더가 또한 생성할 샘플 데이터를 생성하기 위해 심벌들을 재구성하는 인코더(503)에 임베드된 (로컬) 디코더(733)로 구성될 수 있다(심벌들과 코딩된 비디오 비트 스트림 사이의 임의의 압축이 개시된 주제에서 고려되는 비디오 압축 기술들에서 무손실이므로). 해당 재구성된 샘플 스트림은 참조 픽처 메모리(734)에 입력된다. 심벌 스트림의 디코딩이 디코더 위치(로컬 또는 원격)와는 독립적으로 비트 정확한(bit-exact) 결과들을 야기하기 때문에, 참조 픽처 버퍼 콘텐츠가 또한 로컬 인코더와 원격 인코더 사이에서 비트 정확하다. 다시 말해서, 인코더의 예측 부분은 디코딩 동안 예측을 사용할 때 디코더가 "보는(see)" 것과 정확히 동일한 샘플 값들을 참조 픽처 샘플들로서 "본다". 참조 픽처 동기성(reference picture synchronicity)의 이러한 기본적인 원리(그리고 예를 들어, 채널 에러들 때문에 동기성이 유지될 수 없는 경우, 결과적인 드리프트)는 본 기술분야의 통상의 기술자에게 잘 알려져 있다.
- [0116] "로컬" 디코더(733)의 동작은 도 16과 관련하여 위에서 이미 상세히 설명된 "원격" 디코더(510)와 동일할 수 있다. 그러나, 또한 도 16을 잠시 참조하면, 심벌들이 이용가능하고 엔트로피 코더(745) 및 파서(620)에 의한 코딩된 비디오 시퀀스로의 심벌들의 인코딩/디코딩이 무손실일 수 있기 때문에, 채널(612), 수신기(610), 버퍼(615), 및 파서(620)를 포함하는, 디코더(510)의 엔트로피 디코딩 부분들은 로컬 디코더(733)에서 완전히 구현되지 않을 수 있다.
- [0117] 이 시점에서 이루어질 수 있는 관찰은, 디코더에 존재하는 파싱/엔트로피 디코딩을 제외한 임의의 디코더 기술이 또한 필수적으로, 대응하는 인코더에서, 실질적으로 동일한 기능 형식으로 존재할 필요가 있다는 점이다. 인코더 기술들은 포괄적으로 설명된 디코더 기술들의 역(inverse)이기 때문에 그것들에 대한 설명은 축약될 수 있다. 특정 영역들에서만 더 상세한 설명이 요구되고 아래에 제공된다.
- [0118] 그 동작의 일부로서, 소스 코더(730)는, "참조 프레임들"로서 지정된 비디오 시퀀스로부터의 하나 이상의 이전에 코딩된 프레임을 참조하여 예측적으로 입력 프레임을 코딩하는, 모션 보상된 예측 코딩을 수행할 수 있다. 이러한 방식으로, 코딩 엔진(732)은 입력 프레임의 픽셀 블록들과 입력 프레임에 대한 예측 참조(들)로서 선택될 수 있는 참조 프레임(들)의 픽셀 블록들 사이의 차이들을 코딩한다.
- [0119] 로컬 비디오 디코더(733)는, 소스 코더(730)에 의해 생성된 심벌들에 기초하여, 참조 프레임들로서 지정될 수 있는 프레임들의 코딩된 비디오 데이터를 디코딩할 수 있다. 코딩 엔진(732)의 동작들은 유리하게는 손실 프로

세스들일 수 있다. 코딩된 비디오 데이터가 비디오 디코더(도 17에 도시되지 않음)에서 디코딩될 수 있는 경우, 재구성된 비디오 시퀀스는 전형적으로 일부 에러들을 갖는 소스 비디오 시퀀스의 복제본(replica)일 수 있다. 로컬 비디오 디코더(733)는 참조 프레임들에 대해 비디오 디코더에 의해 수행될 수 있는 디코딩 프로세스들을 복제하고, 재구성된 참조 프레임들이 참조 픽처 캐시(734)에 저장되게 야기할 수 있다. 이러한 방식으로, 비디오 인코더(503)는 (송신 에러들 없이) 원단(far-end) 비디오 디코더에 의해 획득될 재구성된 참조 프레임으로서 공통 콘텐츠를 갖는 재구성된 참조 프레임들의 사본들을 로컬로 저장할 수 있다.

[0120] 예측기(735)는 코딩 엔진(732)을 위한 예측 검색들을 수행할 수 있다. 즉, 코딩될 새로운 프레임에 대해, 예측기(735)는 새로운 픽처들에 대한 적절한 예측 참조로서 역할할 수 있는 참조 픽처 모션 벡터들, 블록 형상들 등과 같은 특정 메타데이터 또는 (후보 참조 픽셀 블록들로서의) 샘플 데이터에 대해 참조 픽처 메모리(734)를 검색할 수 있다. 예측기(735)는 적절한 예측 참조들을 찾기 위해 샘플 블록 바이 픽셀 블록(sample block-by-pixel block) 기준으로 동작할 수 있다. 일부 경우들에서, 예측기(735)에 의해 획득된 검색 결과들에 의해 결정된 바와 같이, 입력 픽처는 참조 픽처 메모리(734)에 저장된 다중의 참조 픽처로부터 인출된 예측 참조들을 가질 수 있다.

[0121] 컨트롤러(750)는, 예를 들어, 비디오 데이터를 인코딩하기 위해 사용되는 파라미터들 및 서브그룹 파라미터들의 설정을 포함하여, 비디오 코더(730)의 코딩 동작들을 관리할 수 있다.

[0122] 진술한 모든 기능 유닛들의 출력은 엔트로피 코더(745)에서 엔트로피 코딩을 겪을 수 있다. 엔트로피 코더는, 예를 들어, 허프만 코딩, 가변 길이 코딩, 산술 코딩 등으로서 본 기술분야의 통상의 기술자에게 공지된 기술들에 따라 심벌들을 무손실 압축함으로써 다양한 기능 유닛들에 의해 생성된 심벌들을 코딩된 비디오 시퀀스로 변환한다.

[0123] 송신기(740)는, 인코딩된 비디오 데이터를 저장할 저장 디바이스에 대한 하드웨어/소프트웨어 링크일 수 있는, 통신 채널(760)을 통한 송신을 위해 코딩된 비디오 시퀀스(들)를 준비하기 위해 엔트로피 코더(745)에 의해 생성된 코딩된 비디오 시퀀스(들)를 버퍼링할 수 있다. 송신기(740)는 비디오 코더(730)로부터의 코딩된 비디오 데이터를 송신될 다른 데이터, 예를 들어, 코딩된 오디오 데이터 및/또는 보조 데이터 스트림들(소스들이 도시되지 않음)과 병합할 수 있다.

[0124] 컨트롤러(750)는 인코더(503)의 동작을 관리할 수 있다. 코딩 동안, 컨트롤러(750)는, 각자의 픽처에 적용될 수 있는 코딩 기법들에 영향을 미칠 수 있는, 특정 코딩된 픽처 타입을 각각의 코딩된 픽처에 할당할 수 있다. 예를 들어, 픽처들은 종종 이하의 프레임 타입들 중 하나로서 할당될 수 있다:

[0125] 인트라 픽처(I 픽처)는 예측 소스로서 시퀀스에서의 임의의 다른 프레임을 사용하지 않고서 코딩되고 디코딩될 수 있는 것일 수 있다. 일부 비디오 코덱들은, 예를 들어, 독립적 디코더 리프레시 픽처들(Independent Decoder Refresh Pictures)을 포함하는, 상이한 타입의 인트라 픽처들을 허용한다. 본 기술분야의 통상의 기술자는 I 픽처들의 해당 변형들 및 그것들 각자의 응용들 및 특징들을 인식한다.

[0126] 예측 픽처(Predictive picture)(P 픽처)는 각각의 블록의 샘플 값들을 예측하기 위해 많아야 하나의 모션 벡터 및 참조 인덱스를 사용하여 인트라 예측 또는 인터 예측을 사용하여 코딩되고 디코딩될 수 있는 것일 수 있다.

[0127] 양방향 예측 픽처(Bi-directionally Predictive Picture)(B 픽처)는 각각의 블록의 샘플 값들을 예측하기 위해 많아야 2개의 모션 벡터 및 참조 인덱스를 사용하여 인트라 예측 또는 인터 예측을 사용하여 코딩되고 디코딩될 수 있는 것일 수 있다. 유사하게, 다중-예측 픽처는 단일 블록의 재구성을 위해 2개보다 많은 참조 픽처 및 연관된 메타데이터를 사용할 수 있다.

[0128] 소스 픽처들은 흔히 복수의 샘플 블록(예를 들어, 각각 4x4, 8x8, 4x8, 또는 16x16 샘플들의 블록들)으로 공간적으로 세분되고 블록 바이 블록(block-by-block) 기준으로 코딩될 수 있다. 블록들은 블록들의 각자의 픽처들에 적용되는 코딩 할당에 의해 결정된 다른(이미 코딩된) 블록들을 참조하여 예측적으로 코딩될 수 있다. 예를 들어, I 픽처들의 블록들은 비예측적으로 코딩될 수 있거나 또는 그것들은 동일한 픽처의 이미 코딩된 블록들을 참조하여 예측적으로 코딩될 수 있다(공간 예측 또는 인트라 예측). P 픽처들의 픽셀 블록들은, 하나의 이전에 코딩된 참조 픽처를 참조하여, 공간 예측을 통해 또는 시간 예측을 통해 비예측적으로 코딩될 수 있다. B 픽처들의 블록들은, 하나 또는 2개의 이전에 코딩된 참조 픽처를 참조하여 공간 예측을 통해 또는 시간 예측을 통해, 비예측적으로 코딩될 수 있다.

[0129] 비디오 코더(503)는 H.265 HEVC와 같은 표준 또는 미리 결정된 비디오 코딩 기술에 따라 코딩 동작들을 수행할

수 있다. 그것의 동작 중에, 비디오 코더(503)는, 입력 비디오 시퀀스에서 시간 및 공간 중복성을 활용하는 예측 코딩 동작들을 포함하여, 다양한 압축 동작을 수행할 수 있다. 따라서, 코딩된 비디오 데이터는 사용 중인 비디오 코딩 기술 또는 표준에 의해 특정된 선택스를 따를 수 있다.

[0130] 일 실시예에서, 송신기(740)는 인코딩된 비디오와 함께 추가적인 데이터를 송신할 수 있다. 비디오 코더(730)는 코딩된 비디오 시퀀스의 일부로서 그러한 데이터를 포함할 수 있다. 추가적인 데이터는 시간/공간/SNR 향상 계층들, 중복 픽처들 및 슬라이스들과 같은 다른 형태들의 중복 데이터, SEI(Supplementary Enhancement Information) 메시지들, VUI(Visual Usability Information) 파라미터 세트 프래그먼트들 등을 포함할 수 있다.

[0131] 본 개시내용은 모션 정보가 비디오 인코딩을 위한 트리 스플릿 동안 고려되는 몇몇 블록 파티셔닝 방법에 관한 것이다. 보다 구체적으로는, 본 개시내용에서의 기법들은 모션 필드 정보에 기초한 유연한 트리 구조들을 위한 트리 스플릿 방법들에 관한 것이다. 본 개시내용에서 제안된 기법들은 동종 및 이종 도출된 모션 필드들 둘 다에 적용될 수 있다.

[0132] 블록의 도출된 모션 필드는 도출된 모션 필드가 블록에서의 모든 서브-블록들에 대해 이용가능하고 도출된 모션 필드에서의 모든 모션 벡터들이 유사한 경우, 예컨대, 모션 벡터들이 동일한 참조 프레임을 공유하고 모션 벡터들 간의 절대 차이가 모두 특정 임계값 미만인 경우 동질인 것으로 정의된다. 임계값은 비트스트림들에서 시그널링되거나 미리 정의될 수 있다.

[0133] 블록의 도출된 모션 필드는, 도출된 모션 필드가 블록에서의 모든 서브-블록들에 대해 이용가능하고 도출된 모션 필드에서의 모션 벡터들이 유사하지 않은 경우, 예컨대, 적어도 하나의 모션 벡터가 다른 모션 벡터들에 의해 지칭되지 않는 참조 프레임을 지칭하거나, 또는 필드에서의 2개의 모션 벡터 사이의 적어도 하나의 절대 차이가 시그널링된 또는 미리 정의된 임계값보다 큰 경우, 이종으로서 정의된다.

[0134] 도 19는 비디오 시퀀스를 인코딩 또는 디코딩하기 위한 예시적인 프로세스(800)의 흐름도이다. 일부 구현들에서, 도 19의 하나 이상의 프로세스 블록은 디코더(510)에 의해 수행될 수 있다. 일부 구현들에서, 도 19의 하나 이상의 프로세스 블록들은 인코더(503)와 같이, 디코더(510)로부터 분리되거나 이를 포함하는 또 다른 디바이스 또는 디바이스들의 그룹에 의해 수행될 수 있다.

[0135] 도 19에 도시된 바와 같이, 프로세스(800)는 4:4:4 크로마 포맷 또는 4:2:2 크로마 포맷(810)을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩하는 것을 포함할 수 있다.

[0136] 프로세스(800)가 4:4:4 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩하는 것을 포함할 때, 도 19에 더 도시된 바와 같이, 프로세스(800)는 평균화 동작(820) 이외의 동작을 이용하여 하나의 4x4 크로마 블록의 아핀 모션 벡터를 복사하는 것을 추가로 포함할 수 있다.

[0137] 프로세스(800)가 4:2:2 크로마 포맷을 이용하여 비디오 시퀀스를 인코딩 또는 디코딩하는 것을 포함할 때, 도 19에 더 도시된 바와 같이, 프로세스(800)는 각각의 4x4 크로마 블록을 2개의 4x4 동위치 크로마 블록과 연관시켜 하나의 4x4 크로마 블록의 아핀 모션 벡터가 2개의 동위치 크로마 블록의 모션 벡터들의 평균이도록 하는 것(830)을 추가로 포함할 수 있다.

[0138] 도 19는 프로세스(800)의 예시적인 블록들을 도시하지만, 일부 구현들에서, 프로세스(800)는 도 19에 묘사된 것들보다 추가적인 블록들, 더 적은 블록들, 상이한 블록들, 또는 상이하게 배열된 블록들을 포함할 수 있다. 추가적으로 또는 대안적으로, 프로세스(800)의 블록들 중 2개 이상이 병행적으로 수행될 수 있다.

[0139] 또한, 제안된 방법들은 처리 회로(예를 들어, 하나 이상의 프로세서 또는 하나 이상의 집적 회로)에 의해 구현될 수 있다. 일 예에서, 하나 이상의 프로세서는 제안된 방법들 중 하나 이상을 수행하기 위해 비일시적 컴퓨터 판독가능 매체에 저장되는 프로그램을 실행한다.

[0140] 위에서 설명된 기법들은 컴퓨터 판독가능 명령어들을 이용하여 컴퓨터 소프트웨어로서 구현되고 하나 이상의 컴퓨터 판독가능 매체에 물리적으로 저장될 수 있다. 예를 들어, 도 20은 개시된 주제의 특정 실시예들을 구현하기에 적합한 컴퓨터 시스템(900)을 도시한다.

[0141] 컴퓨터 소프트웨어는, 컴퓨터 중앙 처리 유닛(CPU), 그래픽 처리 유닛(GPU) 등에 의해, 직접 실행될 수 있거나, 또는 해석, 마이크로코드 실행 등을 통해 명령어들을 포함하는 코드를 생성하기 위해, 어셈블리, 컴파일레이션, 링킹, 또는 유사한 메커니즘을 겪을 수 있는 임의의 적절한 머신 코드 또는 컴퓨터 언어를 사용하여 코딩될 수

있다.

- [0142] 명령어들은, 예를 들어, 개인용 컴퓨터, 태블릿 컴퓨터, 서버, 스마트폰, 게이밍 디바이스, 사물 인터넷 디바이스 등을 포함하여, 다양한 타입의 컴퓨터들 또는 그것의 컴포넌트들상에서 실행될 수 있다.
- [0143] 컴퓨터 시스템(900)에 대한 도 20에 도시된 컴포넌트들은 본질상 예시적인 것이고, 본 개시내용의 실시예들을 구현하는 컴퓨터 소프트웨어의 사용 또는 기능성의 범위에 대한 임의의 제한을 암시하도록 의도되지 않는다. 컴포넌트들의 구성이 컴퓨터 시스템(900)의 예시적인 실시예에서 예시된 컴포넌트들 중 임의의 하나 또는 이들의 조합과 관련한 임의의 종속성 또는 요건을 갖는 것으로 해석해서도 안 된다.
- [0144] 컴퓨터 시스템(900)은 특정 휴먼 인터페이스 입력 디바이스들을 포함할 수 있다. 그러한 휴먼 인터페이스 입력 디바이스는, 예를 들어, 촉각 입력(예컨대: 키스트로크, 스위치, 테이터 글러브 모션), 오디오 입력(예컨대: 음성, 손뼉), 시각적 입력(예컨대, 제스처), 후각적 입력(묘사되지 않음)을 통한 하나 이상의 인간 사용자에게 의한 입력에 응답할 수 있다. 휴먼 인터페이스 디바이스들은 또한 오디오(예컨대: 음성, 음악, 주변 사운드), 이미지들(예컨대: 스캐닝된 이미지들, 스틸 이미지 카메라로부터 획득된 사진 이미지들), 비디오(예컨대 2차원 비디오, 입체적 비디오를 포함하는 3차원 비디오)와 같은, 인간에 의한 의식적인 입력과 반드시 직접적으로 관련되는 것은 아닌 특정 미디어를 캡처하기 위해 사용될 수 있다.
- [0145] 입력 휴먼 인터페이스 디바이스들은: 키보드(901), 마우스(902), 트랙패드(903), 터치 스크린(910), 테이터-글러브(904), 조이스틱(905), 마이크로폰(906), 스캐너(907), 카메라(908) 중 하나 이상(각각의 하나만이 묘사됨)을 포함할 수 있다.
- [0146] 컴퓨터 시스템(900)은 특정 휴먼 인터페이스 출력 디바이스들을 또한 포함할 수 있다. 그러한 휴먼 인터페이스 출력 디바이스들은, 예를 들어, 촉각 출력, 사운드, 광, 및 냄새/맛을 통해 하나 이상의 인간 사용자의 감각들을 자극하고 있을 수 있다. 그러한 휴먼 인터페이스 출력 디바이스들은, 촉각 출력 디바이스들(예를 들어, 터치-스크린(910), 테이터-글러브(904), 또는 조이스틱(905)에 의한 촉각 피드백이지만, 입력 디바이스들로서 역할하지 않는 촉각 피드백 디바이스들도 있을 수 있음), 오디오 출력 디바이스들(예컨대: 스피커들(909), 헤드폰들(묘사되지 않음)), 시각적 출력 디바이스들(예컨대 CRT(cathode ray tube) 스크린들, LCD(liquid-crystal display) 스크린들, 플라즈마 스크린들, OLED(organic light-emitting diode) 스크린들을 포함하는 스크린들(910), 각각은 터치-스크린 입력 능력이 있거나 없고, 각각은 촉각 피드백 능력이 있거나 없고 - 이들 중 일부는 스테레오그래픽 출력과 같은 수단을 통해 2차원 시각적 출력 또는 3개보다 많은 차원의 출력을 출력할 수 있음 -); 가상 현실 안경(묘사되지 않음), 홀로그래픽 디스플레이들 및 연기 탱크들(묘사되지 않음)), 및 프린터들(묘사되지 않음)을 포함할 수 있다.
- [0147] 컴퓨터 시스템(900)은 인간 액세스 가능한 저장 디바이스들 및 그것들과 연관된 매체들, 예컨대 CD/DVD 등의 매체(921)를 갖는 CD/DVD ROM/RW(920)를 포함하는 광학 매체, 썸-드라이브(922), 이동식 하드 드라이브 또는 솔리드 스테이트 드라이브(923), 테이프 및 플로피 디스크(묘사되지 않음)와 같은 레거시 자기 매체, 보안 동글(묘사되지 않음)과 같은 특수화된 ROM/ASIC/PLD 기반 디바이스들 등을 또한 포함할 수 있다.
- [0148] 본 기술 분야의 통상의 기술자들은 현재 개시된 주제와 관련하여 사용되는 용어 "컴퓨터 판독가능 매체(computer readable media)"가 송신 매체, 반송파들, 또는 다른 일시적 신호들을 포함하지 않는다는 점을 또한 이해해야 한다.
- [0149] 컴퓨터 시스템(900)은 하나 이상의 통신 네트워크에 대한 인터페이스(들)를 또한 포함할 수 있다. 네트워크들은 예를 들어, 무선, 와이어라인, 광학적일 수 있다. 네트워크들은 추가로 로컬, 광역, 대도시, 차량 및 산업, 실시간, 지연-허용(delay-tolerant) 등일 수 있다. 네트워크들의 예들은 로컬 영역 네트워크들, 예컨대 이더넷, 무선 LAN들, GSM(global systems for mobile communications), 제3 세대(3G), 제4 세대(4G), 제5 세대(5G), 롱-텀 에볼루션(LTE) 등을 포함하는 셀룰러 네트워크들, 케이블 TV, 위성 TV 및 지상파 브로드캐스트 TV를 포함하는 TV 와이어라인 또는 무선 광역 디지털 네트워크들, CANBus를 포함하는 차량 및 산업 등을 포함한다. 특정 네트워크들은 흔히 특정 범용 데이터 포트들 또는 주변기기 버스들(949)(예를 들어, 컴퓨터 시스템(900)의 범용 직렬 버스(USB) 포트들과 같은 것)에 부착된 외부 네트워크 인터페이스 어댑터들을 요구한다; 다른 것들은 흔히 아래에 설명되는 바와 같은 시스템 버스로의 부착에 의해 컴퓨터 시스템(900)의 코어에 통합된다(예를 들어, PC 컴퓨터 시스템으로의 이더넷 인터페이스는 또는 스마트폰 컴퓨터 시스템으로의 셀룰러 네트워크 인터페이스). 이들 네트워크들 중 임의의 것을 사용하여, 컴퓨터 시스템(900)은 다른 엔티티들과 통신할 수 있다. 그러한 통신은 단방향성 수신 전용(예를 들어, 브로드캐스트 TV), 단방향성 송신 전용(예를 들어,

CANbus 대 특정 CANbus 디바이스들), 또는 예를 들어, 로컬 또는 광역 디지털 네트워크들을 사용하는 다른 컴퓨터 시스템들과의 양방향성일 수 있다. 위에서 설명된 바와 같은 네트워크들 및 네트워크 인터페이스들 각각에 대해 특정 프로토콜들 및 프로토콜 스택들이 사용될 수 있다.

[0150] 전술한 휴먼 인터페이스 디바이스들, 인간-액세스 가능한 저장 디바이스들, 및 네트워크 인터페이스들은 컴퓨터 시스템(900)의 코어(940)에 부착될 수 있다.

[0151] 코어(940)는 하나 이상의 중앙 처리 유닛(CPU)(941), 그래픽 처리 유닛(GPU)(942), 필드 프로그래머블 게이트 영역(FPGA)(943)의 형식으로 특수화된 프로그래머블 처리 유닛들, 특정 태스크들에 대한 하드웨어 가속기들(944) 등을 포함할 수 있다. 이들 디바이스는, 판독 전용 메모리(ROM)(945), 랜덤 액세스 메모리(RAM)(946), 내부 비-사용자 액세스 가능 하드 드라이브들, 솔리드-스테이트 드라이브들(SSD들) 등과 같은 내부 대용량 저장소(947)와 함께, 시스템 버스(948)를 통해 접속될 수 있다. 일부 컴퓨터 시스템들에서, 시스템 버스(948)는 추가적인 CPU들, GPU들 등에 의한 확장을 가능하게 하기 위해 하나 이상의 물리적 플러그의 형식으로 액세스 가능할 수 있다. 주변기기 디바이스들은 코어의 시스템 버스(948)에 직접, 또는 주변기기 버스(949)를 통해 부착될 수 있다. 주변기기 버스를 위한 아키텍처들은 PCI(peripheral component interconnect), USB 등을 포함한다.

[0152] CPU들(941), GPU들(942), FPGA들(943), 및 가속기들(944)은, 조합하여, 전술한 컴퓨터 코드를 구성할 수 있는 특정 명령어들을 실행할 수 있다. 그 컴퓨터 코드는 ROM(945) 또는 RAM(946)에 저장될 수 있다. 과도적인 데이터(transitional data)는 또한 RAM(946)에 저장될 수 있는 반면, 영구 데이터는 예를 들어, 내부 대용량 저장소(947)에 저장될 수 있다. 메모리 디바이스들 중 임의의 것에 대한 고속 저장 및 검색은 하나 이상의 CPU(941), GPU(942), 대용량 저장소(947), ROM(945), RAM(946) 등과 밀접하게 연관될 수 있는 캐시 메모리의 사용을 통해 가능하게 될 수 있다.

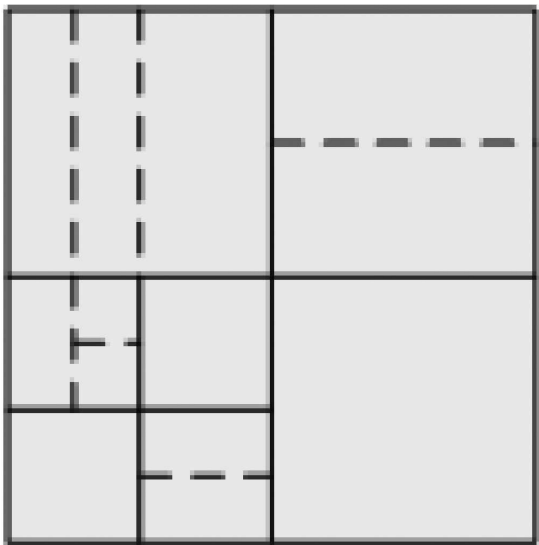
[0153] 컴퓨터 판독가능 매체는 다양한 컴퓨터 구현 동작들을 수행하기 위한 컴퓨터 코드를 그 위에 가질 수 있다. 매체 및 컴퓨터 코드는 본 개시내용의 목적을 위해 특별히 설계되고 구성된 것들일 수 있거나, 또는 그것들은 컴퓨터 소프트웨어 기술분야의 기술자들에게 잘 알려져 있고 이용가능한 종류의 것일 수 있다.

[0154] 제한이 아니라 예로서, 아키텍처를 갖는 컴퓨터 시스템(900), 및 구체적으로 코어(940)는 프로세서(들)(CPU들, GPU들, FPGA, 가속기들 등을 포함함)가 하나 이상의 유형의(tangible) 컴퓨터 판독가능 매체에 구현된 소프트웨어를 실행한 결과로서 기능성을 제공할 수 있다. 그러한 컴퓨터 판독가능 매체는 위에 소개된 바와 같은 사용자-액세스 가능한 대용량 저장소뿐만 아니라, 코어-내부 대용량 저장소(947) 또는 ROM(945)과 같은 비일시적 본질의 것인 코어(940)의 특정 저장소와 연관된 매체일 수 있다. 본 개시내용의 다양한 실시예들을 구현하는 소프트웨어가 그러한 디바이스들에 저장되고 코어(940)에 의해 실행될 수 있다. 컴퓨터 판독가능 매체는 특정 필요에 따라 하나 이상의 메모리 디바이스 또는 칩을 포함할 수 있다. 소프트웨어는 코어(940) 및 구체적으로 그 내부의 프로세서들(CPU, GPU, FPGA 등을 포함함)로 하여금, RAM(946)에 저장된 데이터 구조들을 정의하는 것 및 소프트웨어에 의해 정의된 프로세스들에 따라 그러한 데이터 구조들을 수정하는 것을 포함하여, 본 명세서에 설명된 특정 프로세스들 또는 특정 프로세스들의 특정 부분들을 실행하게 야기할 수 있다. 추가적으로 또는 대안으로서, 컴퓨터 시스템은, 본 명세서에 설명된 특정 프로세스들 또는 특정 프로세스들의 특정 부분들을 실행하기 위해 소프트웨어 대신에 또는 그와 함께 동작할 수 있는, 회로(예를 들어: 가속기(944))에 하드와이어링되거나 다른 방식으로 구현된 로직의 결과로서 기능성을 제공할 수 있다. 소프트웨어에 대한 참조는, 적절한 경우, 로직을 포함할 수 있고, 그 반대도 가능하다. 컴퓨터 판독가능 매체에 대한 참조는, 적절한 경우, 실행을 위한 소프트웨어를 저장하는 회로(예컨대 집적 회로(IC)), 또는 실행을 위한 로직을 구현하는 회로, 또는 둘 다를 포함할 수 있다. 본 개시내용은 하드웨어 및 소프트웨어의 임의의 적합한 조합을 포함한다.

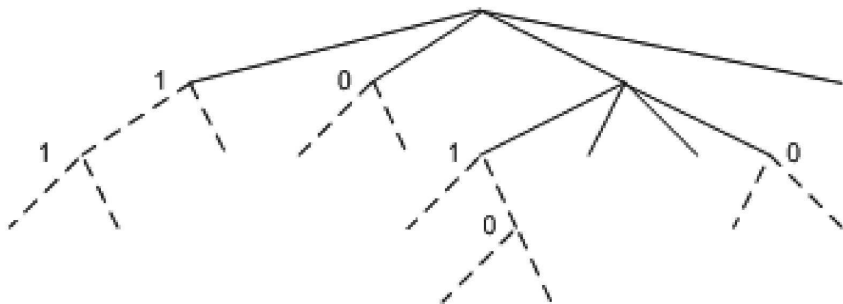
[0155] 본 개시내용이 몇몇 예시적인 실시예를 설명하였지만, 본 개시내용의 범위 내에 속하는 변경들, 치환들, 및 다양한 대체 균등물들이 존재한다. 따라서, 본 기술분야의 통상의 기술자들은, 비록 본 명세서에 명시적으로 도시되거나 설명되지는 않았지만, 본 개시내용의 원리들을 구현하고 따라서 그것의 사상 및 범위 내에 있는, 다수의 시스템 및 방법들을 고안할 수 있을 것이라는 점이 인정될 것이다

도면

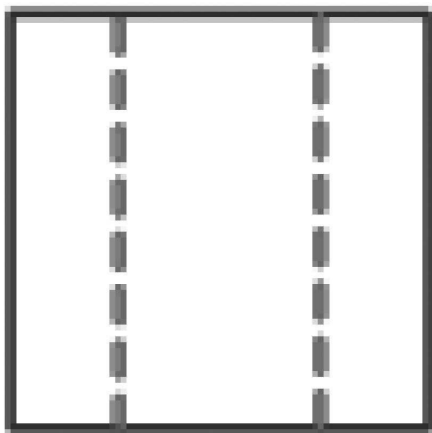
도면1a



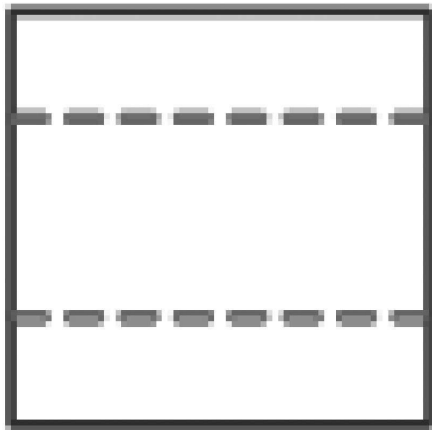
도면1b



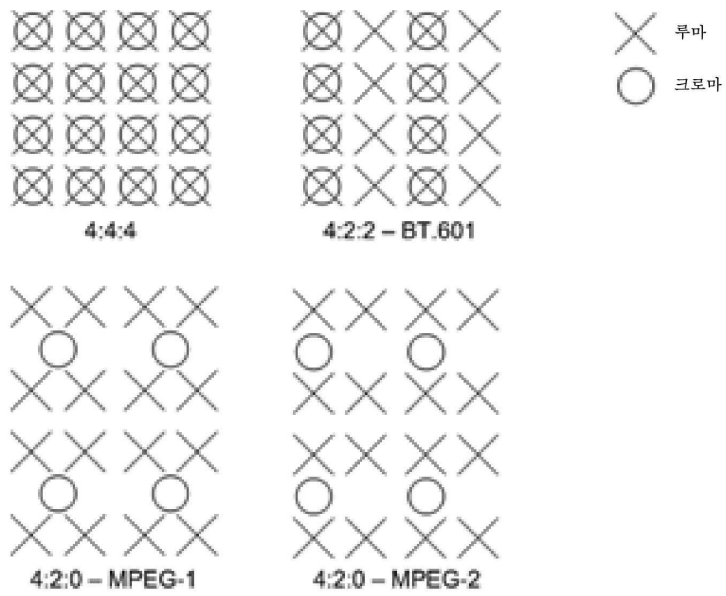
도면2a



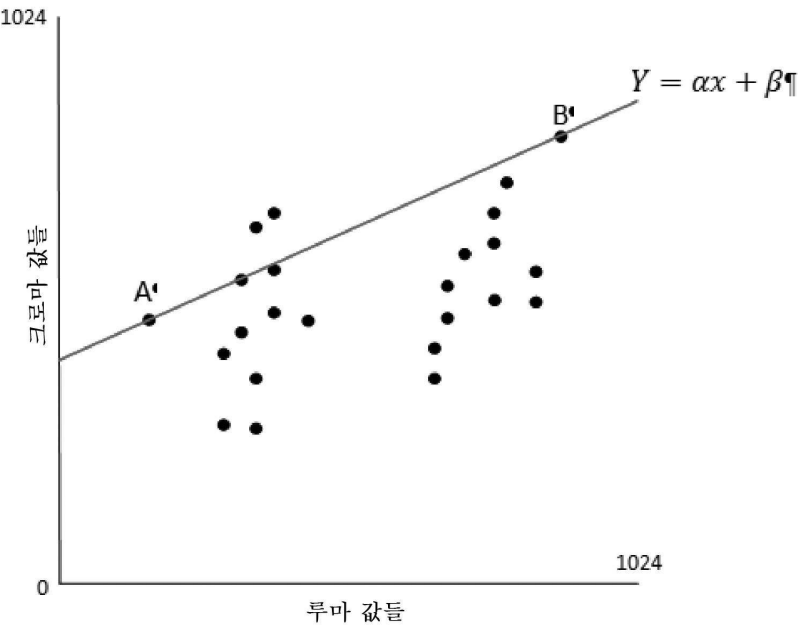
도면2b



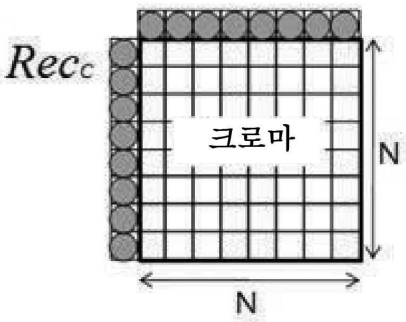
도면3



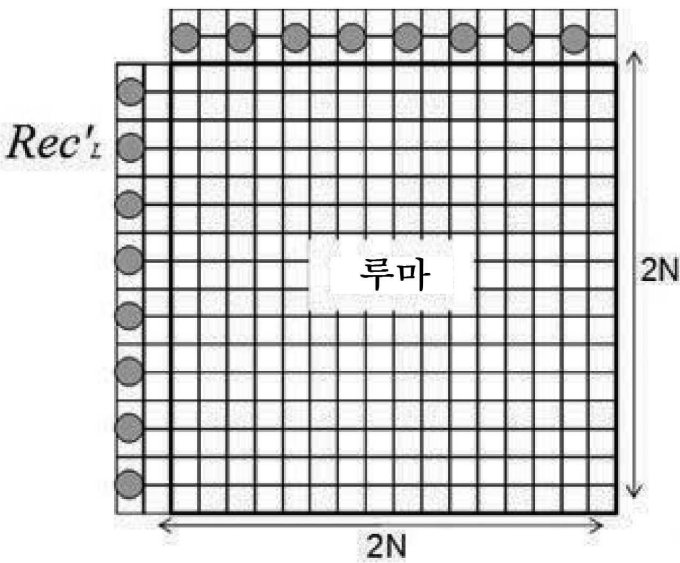
도면4



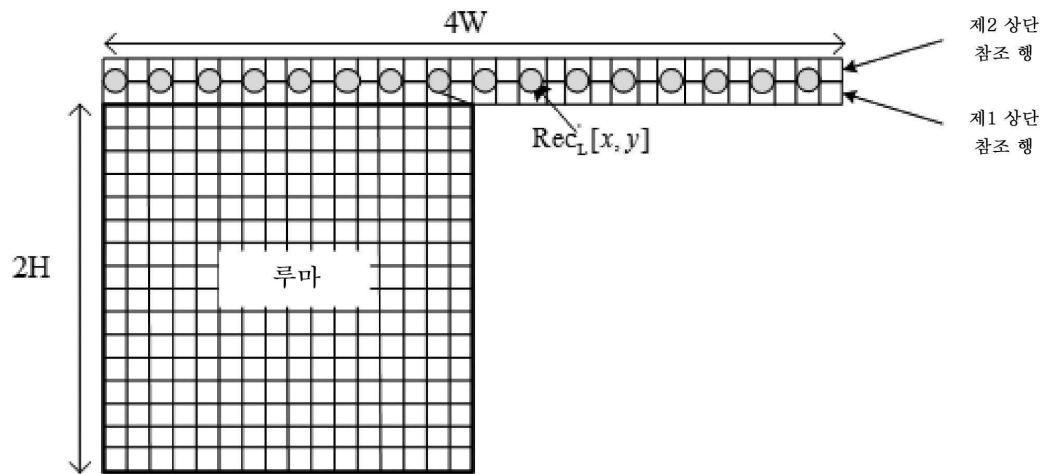
도면5a



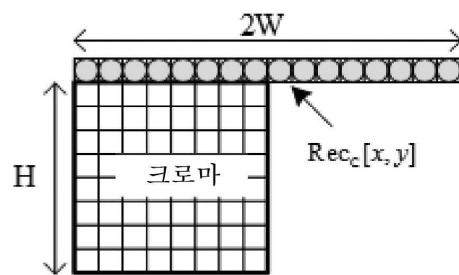
도면5b



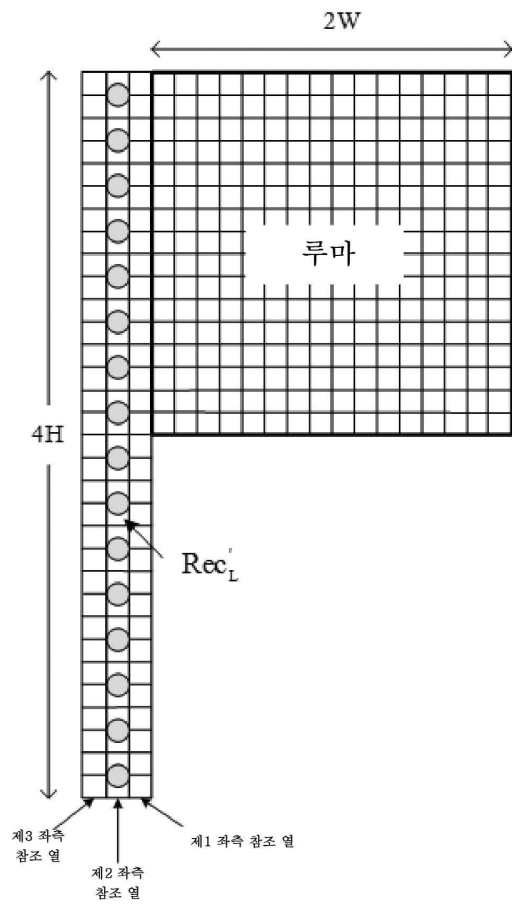
도면6a



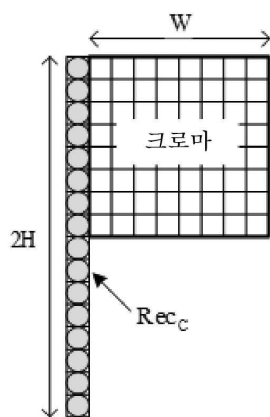
도면6b



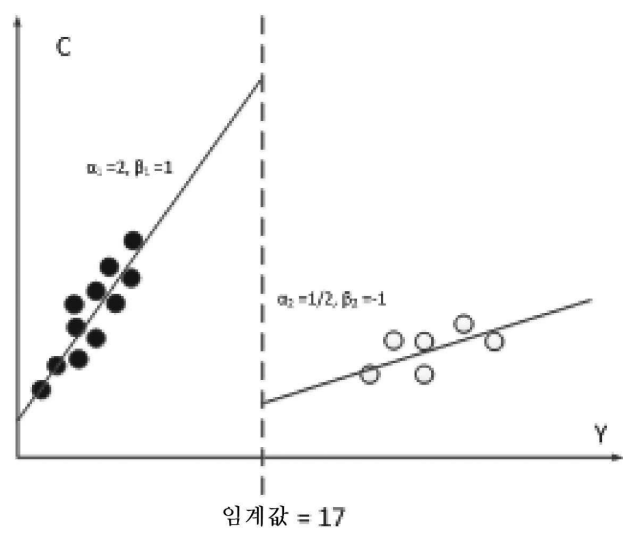
도면7a



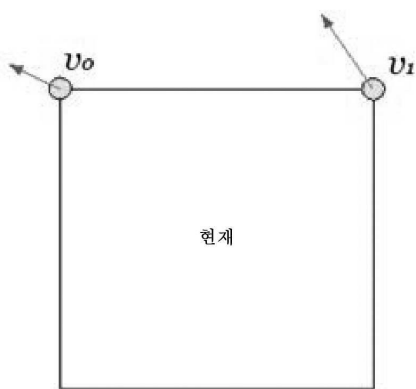
도면7b



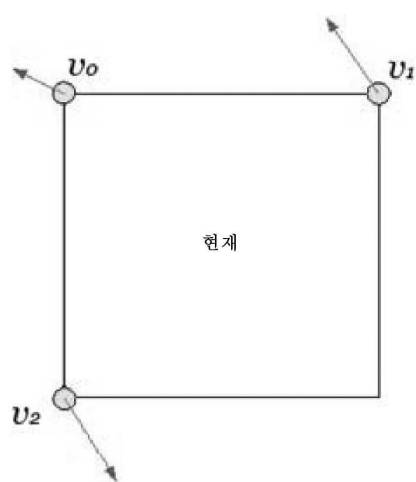
도면8



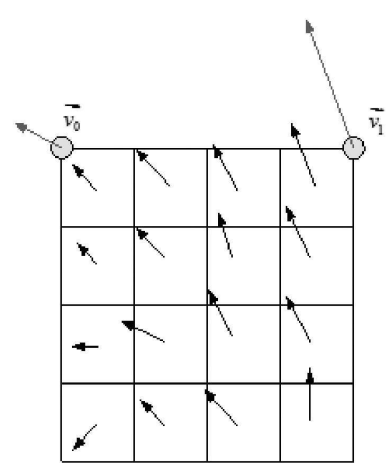
도면9a



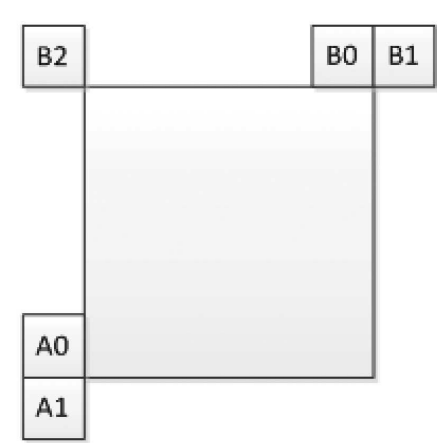
도면9b



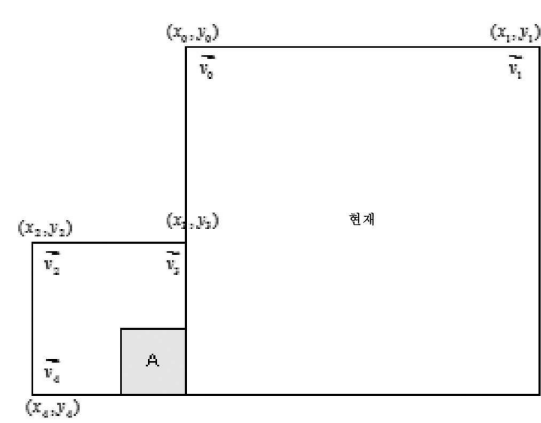
도면10



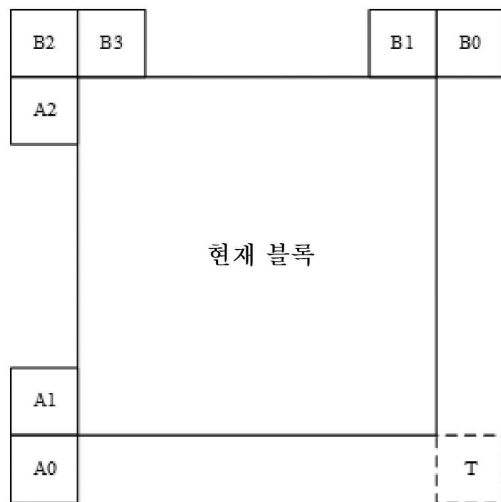
도면11



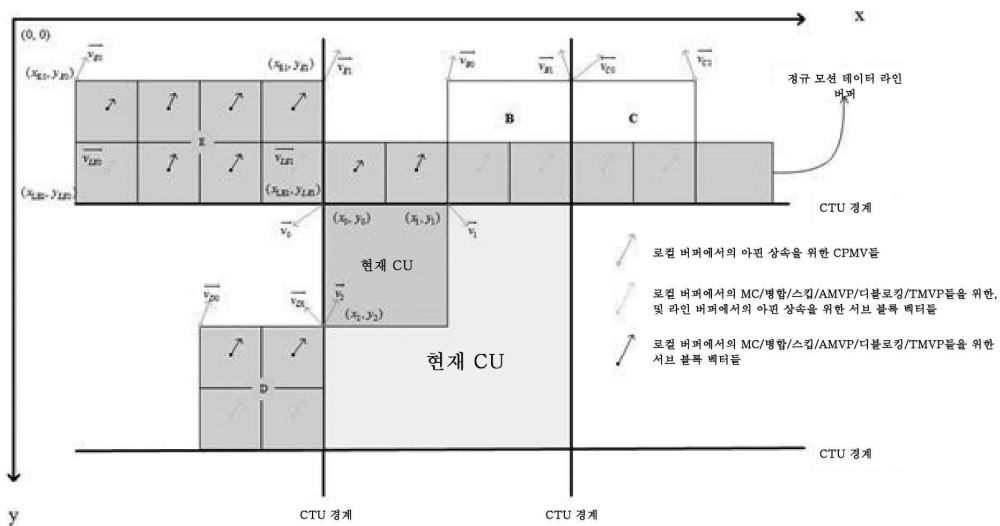
도면12



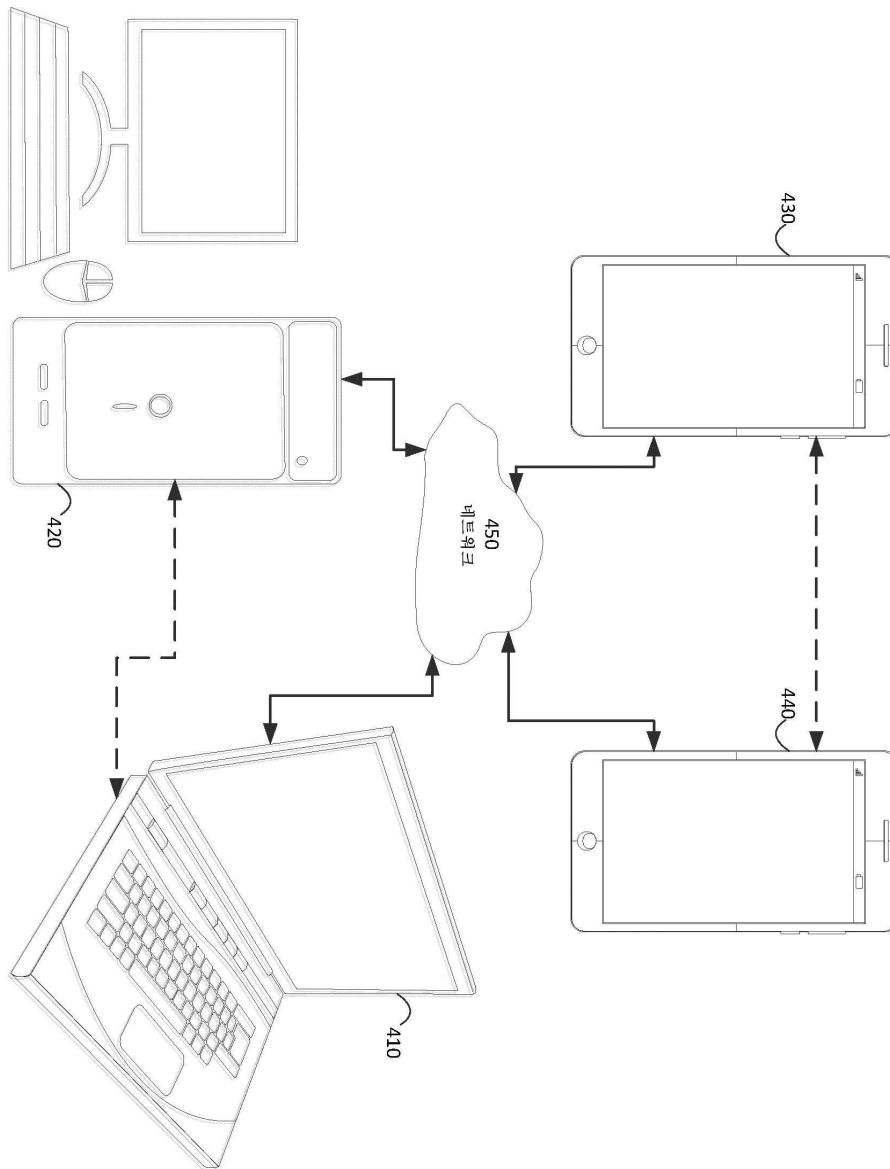
도면13



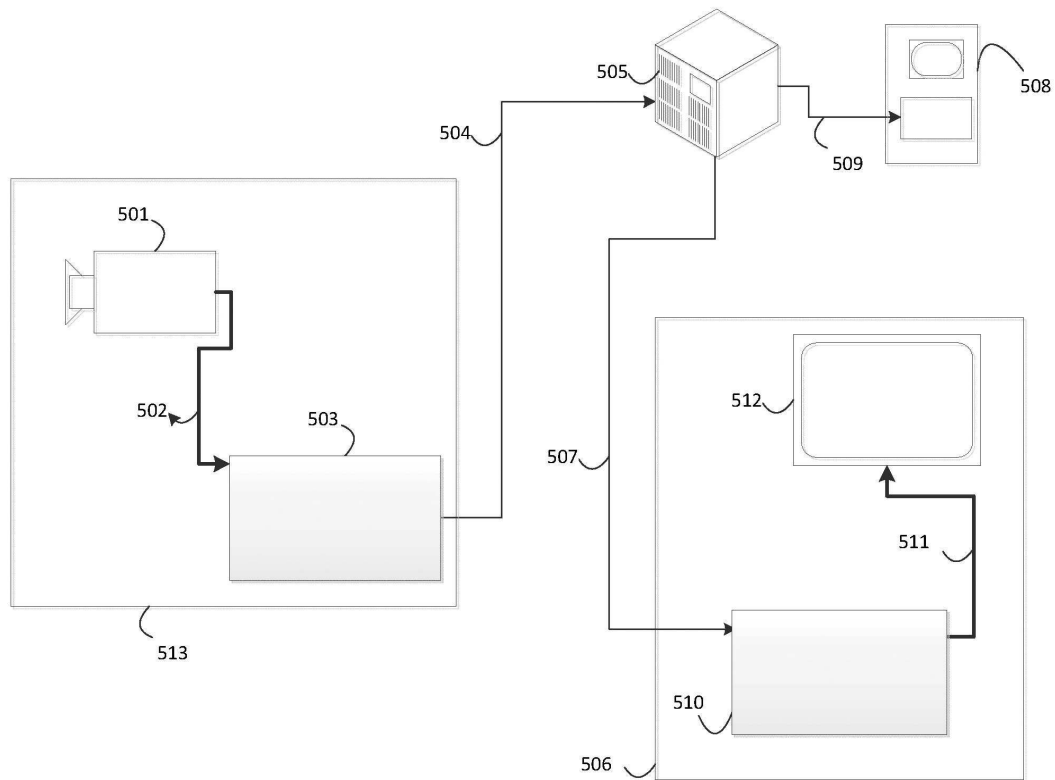
도면14



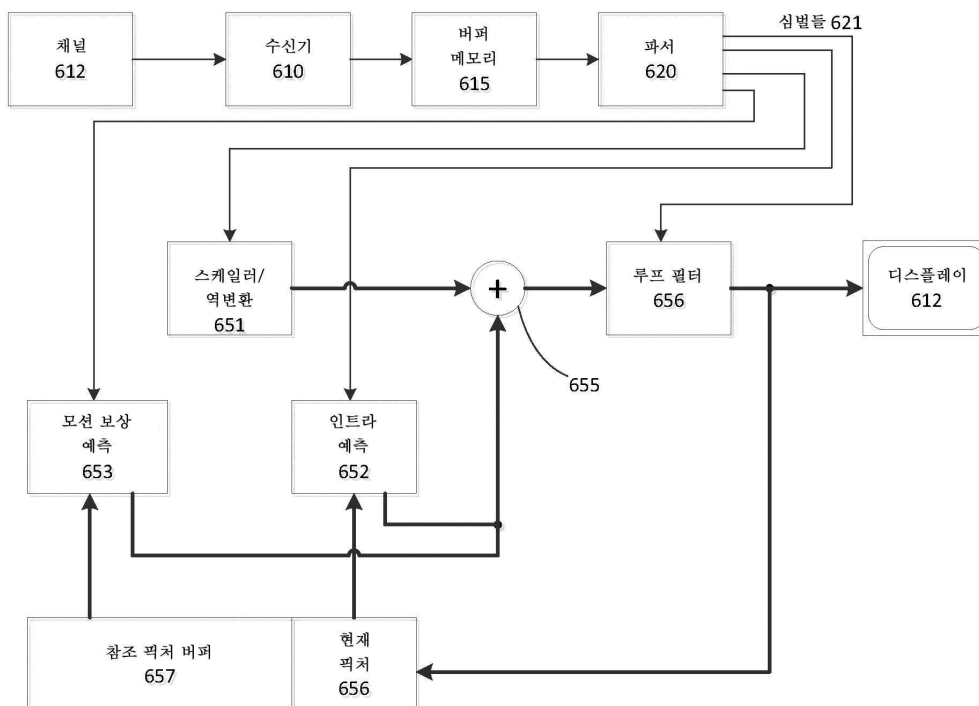
도면15



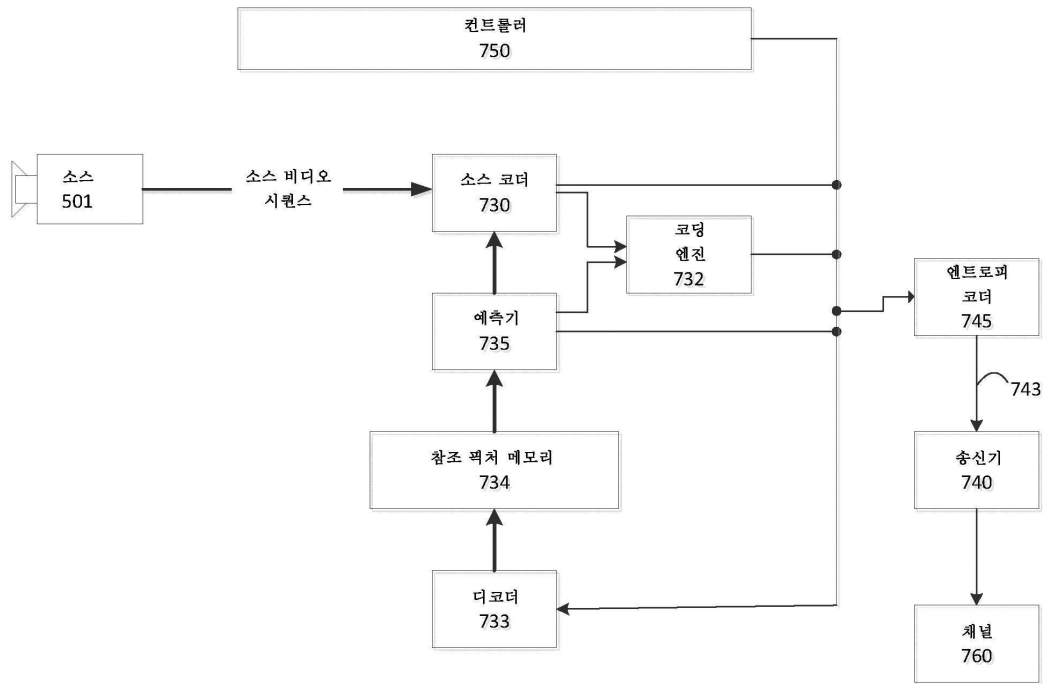
도면16



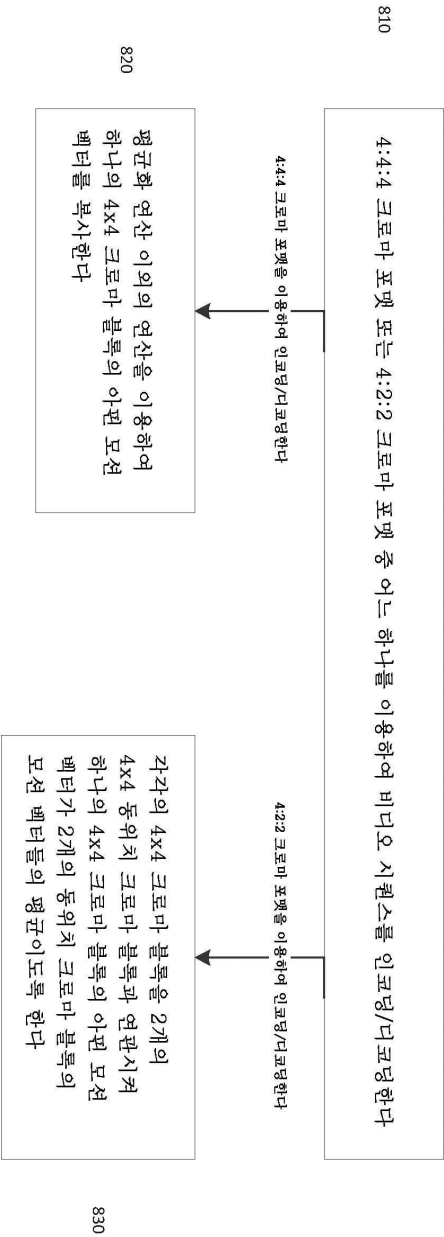
도면17



도면18



도면19



도면20

