



등록특허 10-2644707



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2024년03월06일
(11) 등록번호 10-2644707
(24) 등록일자 2024년03월04일

- (51) 국제특허분류 (Int. Cl.)
HO4N 19/70 (2014.01) *HO4N 19/176* (2014.01)
HO4N 19/44 (2014.01) *HO4N 19/513* (2014.01)
- (52) CPC특허분류
HO4N 19/70 (2015.01)
HO4N 19/176 (2015.01)
- (21) 출원번호 10-2021-7003181
- (22) 출원일자(국제) 2019년06월18일
심사청구일자 2021년02월01일
- (85) 번역문제출일자 2021년02월01일
- (65) 공개번호 10-2021-0027446
- (43) 공개일자 2021년03월10일
- (86) 국제출원번호 PCT/FI2019/050469
- (87) 국제공개번호 WO 2020/008103
국제공개일자 2020년01월09일
- (30) 우선권주장
62/693,118 2018년07월02일 미국(US)
- (56) 선행기술조사문헌
US20150016503 A1*
JP2015526006 A*
- *는 심사관에 의하여 인용된 문헌

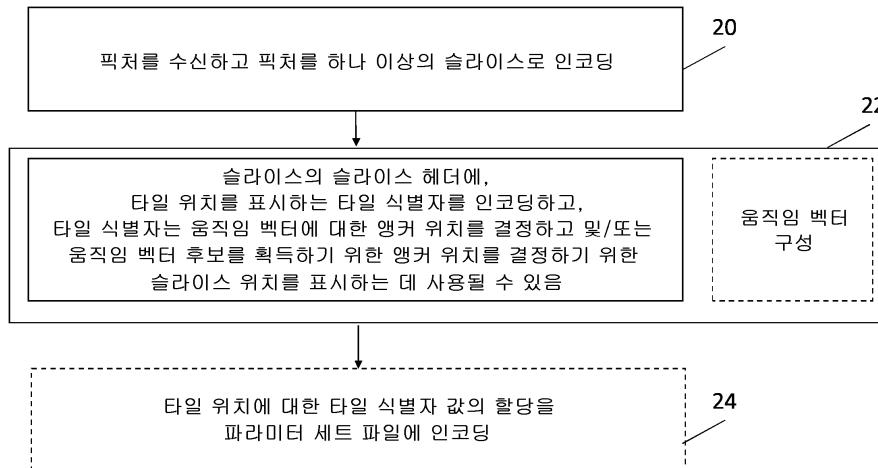
전체 청구항 수 : 총 27 항

심사관 : 김해린

(54) 발명의 명칭 비디오 코딩에서 타일 관련 어드레싱을 위한 방법 및 장치

(57) 요약

픽처를 인코딩 또는 디코딩하기 위한 방법, 장치 및 컴퓨터 프로그램 제품이 제공된다. 방법, 장치 및 컴퓨터 프로그램 제품은, 픽처를 수신하고, 이후 픽처를 하나 이상의 슬라이스로 인코딩한다. 방법, 장치 및 컴퓨터 프로그램 제품은, 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더에, 슬라이스의 하나 이상의 타일에 (뒷면에 계속)

대 표 도 - 도2

대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자를 인코딩한다. 방법, 장치 및 컴퓨터 프로그램 제품은, 타일 위치를 표시하는 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 파라미터 세트 파일에 인코딩한다. 또 다른 또는 같은 방법, 장치 및 컴퓨터 프로그램 제품이 제공되어, 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별하고, 하나 이상의 종속 타일을 포함하는 인-픽처-예측-가능 타일 세트의 표시를 인코딩 또는 디코딩할 수 있다. 하나 이상의 종속 타일은 코딩된 픽처 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 논리적 원자 단위를 형성한다.

(52) CPC특허분류

HO4N 19/44 (2015.01)

HO4N 19/513 (2015.01)

명세서

청구범위

청구항 1

방법으로서,

픽처를 수신하는 것과,

상기 픽처를 하나 이상의 슬라이스로 인코딩하는 것과,

상기 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더에, 상기 슬라이스의 제 1 블록을 포함하는 타일 위치를 표시하는 타일 식별자 - 상기 타일 식별자는 상기 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자 중 하나이고, 상기 슬라이스는 하나 이상의 타일들을 포함함 - 를 인코딩하는 것과,

상기 슬라이스의 상기 제 1 블록에 대한 블록 인덱스 - 상기 블록 인덱스는 상기 타일에 관련된 것임 - 를 구성하는 것과,

상기 하나 이상의 타일 식별자 값들 - 상기 하나 이상의 타일 식별자 값들은 타일 그리드 내의 타일들의 래스터 스캔 순서로 할당된 값들의 사전 정의된 시퀀스 중에 있음- 각각에 대하여 타일 위치를 표시하는 타일 식별자에 대한 타일 식별자 값의 할당을 파라미터 세트 파일에 인코딩하는 것을 포함하는,

방법.

청구항 2

제 1 항에 있어서,

상기 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 구성하는 것을 더 포함하고, 상기 하나 이상의 움직임 벡터는 상기 하나 이상의 타일 식별자로 표시되는 상기 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용되는,

방법.

청구항 3

제 1 항에 있어서,

상기 픽처와 관련된 디코딩 순서 제한 파일을 구성하는 것을 더 포함하고, 상기 디코딩 순서 제한 파일은 상기 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 상기 하나 이상의 타일이 재정렬되는지 여부를 표시하는 재정렬 표시를 포함하는,

방법.

청구항 4

적어도 하나의 프로세서 및 하나 이상의 프로그램에 대한 컴퓨터 프로그램 코드를 포함하는 적어도 하나의 메모리를 포함하는 장치로서,

상기 적어도 하나의 메모리 및 상기 컴퓨터 프로그램 코드는, 상기 적어도 하나의 프로세서를 사용하여, 상기 장치로 하여금 적어도,

픽처를 수신하고,

상기 픽처를 하나 이상의 슬라이스로 인코딩하고,

상기 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더에, 상기 슬라이스의 제 1 블록을 포함하는 타일 위치를 표시하는 타일 식별자 - 상기 타일 식별자는 상기 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자 중 하나이고, 상기 슬라이스는 하나 이상의 타일들을 포함함 - 를 인코딩하고,

상기 슬라이스의 상기 제 1 블록에 대한 블록 인덱스 - 상기 블록 인덱스는 상기 타일에 관련된 것임 - 를 구성하고,

상기 하나 이상의 타일 식별자 값들 - 상기 하나 이상의 타일 식별자 값들은 타일 그리드 내의 타일들의 래스터 스캔 순서로 할당된 값들의 사전 정의된 시퀀스 중에 있음- 각각에 대하여 타일 위치를 표시하는 타일 식별자에 대한 타일 식별자 값의 할당을 파라미터 세트 파일에 인코딩하도록 구성되는,

장치.

청구항 5

제 4 항에 있어서,

상기 적어도 하나의 메모리 및 상기 컴퓨터 프로그램 코드는, 상기 적어도 하나의 프로세서를 사용하여, 상기 장치로 하여금 적어도,

상기 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 구성하도록 더 구성되고, 상기 하나 이상의 움직임 벡터는 상기 하나 이상의 타일 식별자로 표시되는 상기 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용되는,

장치.

청구항 6

제 4 항에 있어서,

상기 적어도 하나의 메모리 및 상기 컴퓨터 프로그램 코드는, 상기 적어도 하나의 프로세서를 사용하여, 상기 장치로 하여금 적어도,

상기 픽처와 관련된 디코딩 순서 제한 파일을 구성하도록 더 구성되고, 상기 디코딩 순서 제한 파일은 상기 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 상기 하나 이상의 타일이 재정렬되는지 여부를 표시하는 재정렬 표시를 포함하는,

장치.

청구항 7

컴퓨터 실행 가능 프로그램 코드 명령어를 포함하는, 적어도 하나의 비일시적 컴퓨터 판독 가능 저장 매체에 저장된 컴퓨터 프로그램으로서, 상기 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때,

픽처를 수신하고,

상기 픽처를 하나 이상의 슬라이스로 인코딩하고,

상기 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더에, 상기 슬라이스의 제 1 블록을 포함하는 타일 위치를 표시하는 타일 식별자 - 상기 타일 식별자는 상기 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자 중 하나이고, 상기 슬라이스는 하나 이상의 타일들을 포함함 -

를 인코딩하고,

상기 슬라이스의 상기 제 1 블록에 대한 블록 인덱스 - 상기 블록 인덱스는 상기 타일에 관련된 것임 - 를 구성하고,

상기 하나 이상의 타일 식별자 값들 - 상기 하나 이상의 타일 식별자 값들은 타일 그리드 내의 타일들의 레스터 스캔 순서로 할당된 값들의 사전 정의된 시퀀스 중에 있음- 각각에 대하여 타일 위치를 표시하는 타일 식별자에 대한 타일 식별자 값의 할당을 파라미터 세트 파일에 인코딩하도록 구성된 프로그램 코드 명령어를 포함하는,

컴퓨터 프로그램.

청구항 8

제 7 항에 있어서,

상기 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때,

상기 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 구성하도록 구성된 프로그램 코드 명령어를 더 포함하고, 상기 하나 이상의 움직임 벡터는 상기 하나 이상의 타일 식별자에 의해 표시되는 상기 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용되는,

컴퓨터 프로그램.

청구항 9

제 7 항에 있어서,

상기 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때,

상기 픽처와 관련된 디코딩 순서 제한 파일을 구성하도록 구성된 프로그램 코드 명령어를 더 포함하고, 상기 디코딩 순서 제한 파일은 상기 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 상기 하나 이상의 타일이 재정렬되는지 여부를 표시하는 재정렬 표시를 포함하는,

컴퓨터 프로그램.

청구항 10

방법으로서,

하나 이상의 슬라이스를 포함하는 코딩된 픽처를 수신하는 것과,

상기 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더로부터, 상기 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자 - 타일 식별자는 상기 슬라이스의 제 1 블록을 포함하는 타일 위치를 표시하고, 상기 하나 이상의 타일 식별자 값들은 타일 그리드 내의 타일들의 레스터 스캔 순서로 할당된 값들의 사전 정의된 시퀀스 중에 있음 - 를 디코딩하는 것과,

상기 슬라이스의 상기 제 1 블록에 대한 블록 인덱스 - 상기 블록 인덱스는 상기 타일에 관련된 것임 - 를 디코딩하는 것과,

상기 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치로부터 시작하여 상기 슬라이스의 슬라이스 데이터를 디코딩된 픽처로 디코딩하는 것을 포함하는,

방법.

청구항 11

제 10 항에 있어서,

상기 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 디코딩하는 것을 더 포함하고, 상기 하나 이상의 움직임 벡터는 상기 하나 이상의 타일 식별자로 표시되는 상기 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용되는,

방법.

청구항 12

제 10 항에 있어서,

파라미터 세트로부터 상기 타일 위치를 표시하는 상기 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 디코딩하는 것을 더 포함하는,

방법.

청구항 13

제 10 항에 있어서,

상기 픽처와 관련된 디코딩 순서 제한 파일에 따라 상기 하나 이상의 타일을 디코딩하는 것을 더 포함하고, 상기 디코딩 순서 제한 파일은 상기 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 상기 하나 이상의 타일이 재정렬되는지 여부를 표시하는 재정렬 표시를 포함하는,

방법.

청구항 14

적어도 하나의 프로세서 및 하나 이상의 프로그램에 대한 컴퓨터 프로그램 코드를 포함하는 적어도 하나의 메모리를 포함하는 장치로서,

상기 적어도 하나의 메모리 및 상기 컴퓨터 프로그램 코드는, 상기 적어도 하나의 프로세서를 사용하여, 상기 장치로 하여금 적어도,

하나 이상의 슬라이스를 포함하는 코딩된 픽처를 수신하고,

상기 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더로부터, 상기 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자 - 타일 식별자는 상기 슬라이스의 제 1 블록을 포함하는 타일 위치를 표시하고, 상기 하나 이상의 타일 식별자 값들은 타일 그리드 내의 타일들의 래스터 스캔 순서로 할당된 값들의 사전 정의된 시퀀스 중에 있음 - 를 디코딩하고,

상기 슬라이스의 상기 제 1 블록에 대한 블록 인덱스 - 상기 블록 인덱스는 상기 타일에 관련된 것임 - 를 디코딩하고,

상기 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치로부터 시작하여 상기 슬라이스의 슬라이스 데이터를 디코딩된 픽처로 디코딩하도록 구성되는,

장치.

청구항 15

제 14 항에 있어서,

상기 적어도 하나의 메모리 및 상기 컴퓨터 프로그램 코드는, 상기 적어도 하나의 프로세서를 사용하여, 상기

장치로 하여금 적어도,

상기 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 디코딩하도록 더 구성되고, 상기 하나 이상의 움직임 벡터는 상기 하나 이상의 타일 식별자로 표시되는 상기 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용되는,

장치.

청구항 16

제 14 항에 있어서,

상기 적어도 하나의 메모리 및 상기 컴퓨터 프로그램 코드는, 상기 적어도 하나의 프로세서를 사용하여, 상기 장치로 하여금 적어도,

파라미터 세트로부터 상기 타일 위치를 표시하는 상기 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 디코딩하도록 더 구성되는,

장치.

청구항 17

제 14 항에 있어서,

상기 적어도 하나의 메모리 및 상기 컴퓨터 프로그램 코드는, 상기 적어도 하나의 프로세서를 사용하여, 상기 장치로 하여금 적어도,

상기 꾹처와 관련된 디코딩 순서 제한 파일에 따라 상기 하나 이상의 타일을 디코딩하도록 더 구성되고, 상기 디코딩 순서 제한 파일은 상기 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 상기 하나 이상의 타일이 재정렬되는지 여부를 표시하는 재정렬 표시를 포함하는,

장치.

청구항 18

컴퓨터 실행 가능 프로그램 코드 명령어를 포함하는, 적어도 하나의 비일시적 컴퓨터 판독 가능 저장 매체에 저장된 컴퓨터 프로그램으로서, 상기 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때,

하나 이상의 슬라이스를 포함하는 코딩된 꾹처를 수신하고,

상기 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더로부터, 상기 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자 - 타일 식별자는 상기 슬라이스의 제 1 블록을 포함하는 타일 위치를 표시하고, 상기 하나 이상의 타일 식별자 값들은 타일 그리드 내의 타일들의 래스터 스캔 순서로 할당된 값들의 사전 정의된 시퀀스 중에 있음 - 를 디코딩하고,

상기 슬라이스의 상기 제 1 블록에 대한 블록 인덱스 - 상기 블록 인덱스는 상기 타일에 관련된 것임 - 를 디코딩하고,

상기 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치로부터 시작하여 상기 슬라이스의 슬라이스 데이터를 디코딩된 꾹처로 디코딩하도록 구성된 프로그램 코드 명령어를 포함하는,

컴퓨터 프로그램.

청구항 19

제 18 항에 있어서,

상기 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때,

상기 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 디코딩하도록 구성된 프로그램 코드 명령어를 더 포함하고, 상기 하나 이상의 움직임 벡터는 상기 하나 이상의 타일 식별자로 표시되는 상기 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용되는,

컴퓨터 프로그램.

청구항 20

제 18 항에 있어서,

상기 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때,

파라미터 세트로부터 상기 타일 위치를 표시하는 상기 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 디코딩하도록 구성된 프로그램 코드 명령어를 더 포함하는,

컴퓨터 프로그램.

청구항 21

제 18 항에 있어서,

상기 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때,

상기 꾹처와 관련된 디코딩 순서 제한 파일에 따라 상기 하나 이상의 타일을 디코딩하도록 구성된 프로그램 코드 명령어를 더 포함하고, 상기 디코딩 순서 제한 파일은 상기 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 상기 하나 이상의 타일이 재정렬되는지 여부를 표시하는 재정렬 표시를 포함하는,

컴퓨터 프로그램.

청구항 22

제 1 항에 있어서,

상기 하나 이상의 슬라이스 각각이 상기 하나 이상의 타일의 좌상단으로부터 시작하는지 각각의 슬라이스에서 결정하는 것을 더 포함하는,

방법.

청구항 23

제 4 항에 있어서,

상기 적어도 하나의 메모리 및 상기 컴퓨터 프로그램 코드는, 상기 적어도 하나의 프로세서를 사용하여, 상기 장치로 하여금 적어도,

상기 하나 이상의 슬라이스 각각이 상기 하나 이상의 타일의 좌상단으로부터 시작하는지 각각의 슬라이스에서 결정하도록 더 구성되는,

장치.

청구항 24

제 7 항에 있어서,

상기 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때,

상기 하나 이상의 슬라이스 각각이 상기 하나 이상의 타일의 좌상단으로부터 시작하는지 각각의 슬라이스에서 결정하도록 구성된 프로그램 코드 명령어를 더 포함하는,

컴퓨터 프로그램.

청구항 25

제 10 항에 있어서,

상기 하나 이상의 슬라이스 각각이 상기 하나 이상의 타일의 좌상단으로부터 시작하는지 각각의 슬라이스에서 디코딩하는 것을 더 포함하는,

방법.

청구항 26

제 14 항에 있어서,

상기 적어도 하나의 메모리 및 상기 컴퓨터 프로그램 코드는, 상기 적어도 하나의 프로세서를 사용하여, 상기 장치로 하여금 적어도,

상기 하나 이상의 슬라이스 각각이 상기 하나 이상의 타일의 좌상단으로부터 시작하는지 각각의 슬라이스에서 디코딩하도록 더 구성되는,

장치.

청구항 27

제 18 항에 있어서,

상기 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때,

상기 하나 이상의 슬라이스 각각이 상기 하나 이상의 타일의 좌상단으로부터 시작하는지 각각의 슬라이스에서 디코딩하도록 구성된 프로그램 코드 명령어를 더 포함하는,

컴퓨터 프로그램.

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

청구항 31

삭제

청구항 32

삭제

청구항 33

삭제

청구항 34

삭제

청구항 35

삭제

청구항 36

삭제

청구항 37

삭제

청구항 38

삭제

청구항 39

삭제

청구항 40

삭제

청구항 41

삭제

청구항 42

삭제

청구항 43

삭제

청구항 44

삭제

청구항 45

삭제

청구항 46

삭제

청구항 47

삭제

청구항 48

삭제

청구항 49

삭제

청구항 50

삭제

청구항 51

삭제

청구항 52

삭제

청구항 53

삭제

청구항 54

삭제

청구항 55

삭제

청구항 56

삭제

청구항 57

삭제

청구항 58

삭제

청구항 59

삭제

청구항 60

삭제

청구항 61

삭제

청구항 62

삭제

청구항 63

삭제

발명의 설명

기술 분야

[0001] 예시적인 실시예는 일반적으로 비디오 인코딩 및 디코딩에 관한 것이다.

배경 기술

[0002] HEVC(High Efficiency Video Coding) 코덱과 같은 일부 비디오 코덱에서, 비디오 팩처는 팩처의 영역을 커버하는 코딩 단위(coding unit, CU)로 분할된다. CU는 CU 내의 샘플에 대한 예측 프로세스를 정의하는 하나 이상의 예측 단위(prediction unit, PU) 및 CU의 샘플에 대한 예측 오류 코딩 프로세스를 정의하는 하나 이상의 변환 단위(transform unit, TU)로 구성된다. CU는 가능한 CU 크기의 사전 정의된 세트로부터 선택할 수 있는 크기를 갖는 샘플의 정사각형 블록으로 구성된다. 최대 허용 크기를 갖는 CU는 LCU(largest coding unit) 또는 코딩 트리 단위(coding tree unit, CTU)로 지칭될 수 있고, 비디오 팩처는 비중첩(non-overlapping) LCU로 분할된다. LCU는, 예를 들어 LCU와 산출된 CU를 반복적으로(recursively) 분할하여 더 작은 CU의 조합으로 더 분할될 수 있다. 각각의 산출되는 CU는 CU와 관련된 적어도 하나의 PU 및 적어도 하나의 TU를 가질 수 있다. 각각의 PU 및 TU는 각각 예측 및 예측 오류 코딩 프로세스의 입도(granularity)를 높이기 위해 더 작은 PU 및 TU로 더 분할될 수 있다. 각각의 PU는 PU 내의 픽셀에 대해 어떤 종류의 예측이 적용되는지를 정의하는 PU와 관련된 예측 정보(예를 들어, 인터 예측된 PU에 대한 움직임 벡터 정보 및 인트라 예측된 PU에 대한 인트라 예측 방향성 정보)를 갖는다.

[0003] 슬라이스는 하나의 독립 슬라이스 세그먼트 및 같은 액세스 단위 내에서 (존재하는 경우) 다음 독립 슬라이스 세그먼트에 선행하는 (존재하는 경우) 모든 후속하는 종속 슬라이스 세그먼트에 포함된 정수 개수의 CTU이다. HEVC에서, 슬라이스 헤더는 현재 슬라이스 세그먼트 또는 현재 종속 슬라이스 세그먼트에 선행하는 독립 슬라이스 세그먼트인 독립 슬라이스 세그먼트의 슬라이스 세그먼트 헤더로 정의되고, 슬라이스 세그먼트 헤더는 슬라이스 세그먼트에 표현된 제 1 또는 모든 코딩 트리 단위에 속하는 데이터 요소를 포함하는 코딩된 슬라이스 세그먼트의 일부로 정의된다. CU는 타일 내에서 또는 타일이 사용되지 않는 경우 팩처 내에서 LCU의 래스터 스캔 순서로 스캔된다. LCU 내에서, CU는 특정 스캔 순서를 갖는다. 팩처 내에서 슬라이스의 위치는 팩처 내에서 래스터 스캔 순서로 블록 인덱싱을 통해 표시된다.

[0004] 팩처 내의 슬라이스의 위치는 팩처 내에서 래스터 스캔 순서로 블록 인덱싱을 통해 표시되기 때문에, 슬라이스의 위치 변경 또는 타일 또는 슬라이스의 서브세트(팩처의 좌상단(top-left corner)으로부터 시작하는 팩처 전체 서브세트는 제외) 추출은 슬라이스 헤더의 재작성을 필요로 한다. 슬라이스의 위치 변경은, 타일 세트를 자체 비트스트림으로 추출할 때, 디코딩된 팩처에서 360도 비디오를 위해 인코딩된 팩처와는 다른 타일 순서 또는 팩처 크기로 뷰포트(viewport) 전환 등과 같은 다양한 상황에서 발생한다. 슬라이스 헤더 재작성은 스트림의 비트 전송률에 영향을 미치고 및/또는 특별한 재작성 명령어를 필요로 하며, 이 모든 것이 활용해야 할 추가 계산 시간 및 리소스를 발생시키므로 슬라이스의 위치 변경을 목적으로 슬라이스 헤더를 재작성하는 것은 비효율적이다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0005] 일부 실시예에서, 팩처를 코딩된 팩처로 인코딩하기 위해 예시적인 실시예에 따라 방법, 장치 및 컴퓨터 프로그램 제품이 제공된다. 예시적인 일 실시예에서, 팩처를 수신하는 것을 포함하는 방법이 제공된다. 방법은 팩처를 하나 이상의 슬라이스로 인코딩하는 것을 더 포함한다. 방법은, 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더에, 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자를 인코딩하는 것을 더 포함한다. 방법은 타일 위치를 표시하는 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 파라미터 세트 파일에 인코딩하는 것을 더 포함한다.

[0006] 이러한 방법의 일부 구현예에서, 방법은, 슬라이스 헤더에, 하나 이상의 타일 중 하나의 타일에 관련된 블록 인덱스를 인코딩하는 것을 더 포함한다. 블록 인덱스는 타일 내의 블록에 대한 블록 위치를 표시하는 신택스(syntax) 요소를 포함한다. 일부 실시예에서, 하나 이상의 타일 식별자 중 하나의 타일 식별자는 슬라이스의

사전 정의된 블록을 포함하는 타일을 표시한다. 일부 실시예에서, 방법은 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 구성하는 것을 더 포함한다. 하나 이상의 움직임 벡터는 하나 이상의 타일 식별자에 의해 표시되는 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용된다. 일부 실시예에서, 방법은 픽처와 관련된 디코딩 순서 제한 파일을 구성하는 것을 더 포함한다. 디코딩 순서 제한 파일은 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 하나 이상의 타일이 재정렬(reorder)될 수 있는지 여부를 표시하는 재정렬 표시를 포함한다.

[0007] 또 다른 예시적인 실시예에서, 적어도 하나의 프로세서 및 하나 이상의 프로그램에 대한 컴퓨터 프로그램 코드를 포함하는 적어도 하나의 메모리를 포함하는 장치가 제공되고, 적어도 하나의 메모리 및 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 적어도 픽처를 수신하도록 구성된다. 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 픽처를 하나 이상의 슬라이스로 인코딩하도록 더 구성된다. 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금, 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더에, 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자를 인코딩하도록 더 구성된다. 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 타일 위치를 표시하는 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 파라미터 세트 파일에 인코딩하도록 더 구성된다.

[0008] 이러한 장치의 일부 구현예에서, 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금, 슬라이스 헤더에, 하나 이상의 타일 중 하나의 타일에 관련된 블록 인덱스를 인코딩하도록 더 구성된다. 블록 인덱스는 타일 내의 블록에 대한 블록 위치를 표시하는 신택스 요소를 포함한다. 일부 실시예에서, 하나 이상의 타일 식별자 중 하나의 타일 식별자는 슬라이스의 사전 정의된 블록을 포함하는 타일을 표시한다. 일부 실시예에서, 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 구성하도록 더 구성된다. 하나 이상의 움직임 벡터는 하나 이상의 타일 식별자에 의해 표시되는 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용된다. 일부 실시예에서, 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 픽처와 관련된 디코딩 순서 제한 파일을 구성하도록 더 구성된다. 디코딩 순서 제한 파일은 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 하나 이상의 타일이 재정렬될 수 있는지 여부를 표시하는 재정렬 표시를 포함한다.

[0009] 다른 예시적인 실시예에서, 픽처를 수신하기 위한 수단을 포함하는 장치가 제공된다. 장치는 픽처를 하나 이상의 슬라이스로서 인코딩하기 위한 수단을 더 포함한다. 장치는, 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더에, 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자를 인코딩하는 수단을 더 포함한다. 장치는 타일 위치를 표시하는 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 파라미터 세트 파일에 인코딩하는 수단을 더 포함한다.

[0010] 이러한 장치의 일부 구현예에서, 장치는, 슬라이스 헤더에, 하나 이상의 타일 중 하나의 타일에 관련된 블록 인덱스를 인코딩하는 수단을 더 포함한다. 블록 인덱스는 타일 내의 블록에 대한 블록 위치를 표시하는 신택스 요소를 포함한다. 일부 실시예에서, 하나 이상의 타일 식별자 중 하나의 타일 식별자는 슬라이스의 사전 정의된 블록을 포함하는 타일을 표시한다. 일부 실시예에서, 장치는 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 구성하기 위한 수단을 더 포함한다. 하나 이상의 움직임 벡터는 하나 이상의 타일 식별자에 의해 표시되는 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용된다. 일부 실시예에서, 장치는 픽처와 관련된 디코딩 순서 제한 파일을 구성하기 위한 수단을 더 포함한다. 디코딩 순서 제한 파일은 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 하나 이상의 타일이 재정렬될 수 있는지 여부를 표시하는 재정렬 표시를 포함한다.

[0011] 또 다른 예시적인 실시예에서, 컴퓨터 실행 가능 프로그램 코드 명령어가 저장된 적어도 하나의 비일시적 컴퓨터 판독 가능 저장 매체를 포함하는 컴퓨터 프로그램 제품이 제공되고, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 픽처를 수신하도록 구성된 프로그램 코드 명령어를 포함한다. 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 픽처를 하나 이상의 슬라이스로 인코딩하도록 구성된 프로그램 코드 명령어를 더 포함한다. 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더에, 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자를 인코딩하도록 구성된 프로그램 코드 명령어를 더 포함한다. 컴퓨터 실행 가능 프로그램 코드 명령어는, 실

행될 때, 타일 위치를 표시하는 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 파라미터 세트 파일에 인코딩하도록 구성된 프로그램 코드 명령어를 더 포함한다.

[0012] 이러한 컴퓨터 프로그램 제품의 일부 구현예에서, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 슬라이스 헤더에, 하나 이상의 타일 중 하나의 타일에 관련된 블록 인덱스를 인코딩하도록 구성된 프로그램 코드 명령어를 더 포함한다. 블록 인덱스는 타일 내의 블록에 대한 블록 위치를 표시하는 십진수 요소를 포함한다. 일부 실시예에서, 하나 이상의 타일 식별자 중 하나의 타일 식별자는 슬라이스의 사전 정의된 블록을 포함하는 타일을 표시한다. 일부 실시예에서, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 구성하도록 구성된 프로그램 코드 명령어를 더 포함한다. 하나 이상의 움직임 벡터는 하나 이상의 타일 식별자에 의해 표시되는 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용된다. 일부 실시예에서, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 픽처와 관련된 디코딩 순서 제한 파일을 구성하도록 구성된 프로그램 코드 명령어를 더 포함한다. 디코딩 순서 제한 파일은 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 하나 이상의 타일이 재정렬될 수 있는지 여부를 표시하는 재정렬 표시를 포함한다.

[0013] 일부 실시예에서, 인-픽처-예측-가능(in-picture-prediction-enabled, IPPE) 타일 세트를 구성하기 위해 예시적인 실시예에 따라 방법, 장치 및 컴퓨터 프로그램 제품이 제공된다. 하나의 예시적인 실시예에서, 하나 이상의 타일을 포함하는 코딩된 픽처를 포함하는 비트스트림을 수신 또는 인코딩하는 것을 포함하는 방법이 제공된다. 방법은 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별하는 것을 더 포함한다. 하나 이상의 종속 타일은 코딩된 픽처 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 논리적 원자 단위를 형성한다. 방법은 하나 이상의 종속 타일을 포함하는 인-픽처-예측-가능 타일 세트의 표시를 구성하는 것을 더 포함한다. 방법은 비트스트림 및 인-픽처-예측-가능 타일 세트의 표시를 저장하도록 하는 것을 더 포함한다.

[0014] 이러한 방법의 일부 구현예에서, 방법은 인-픽-예측-가능 타일 세트의 표시를, 비트스트림과 관련된 시퀀스 파라미터 세트, 비트스트림과 관련된 픽처 파라미터 세트, 또는 비트스트림과 관련된 보충 개선 정보(supplemental enhancement information) 메시지 중 하나 이상에 인코딩하는 것을 더 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트가 시간적 움직임 제한 타일 세트인지 여부를 표시한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 하나 이상의 타일에 대한 다른 인-픽처 예측되지 않은 타일이 아닌 하나 이상의 타일의 예측 종속성의 세트를 표시하는 종속성 포레스트(forest) 또는 그래프를 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 종속 타일의 세트의 인트라 블록 카피 움직임 벡터의 세트에 대한 앵커 위치의 세트를 포함한다.

[0015] 다른 예시적인 실시예에서, 적어도 하나의 프로세서 및 하나 이상의 프로그램에 대한 컴퓨터 프로그램 코드를 포함하는 적어도 하나의 메모리를 포함하는 장치가 제공되고, 적어도 하나의 메모리 및 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 적어도 하나 이상의 타일을 포함하는 코딩된 픽처를 포함하는 비트스트림을 수신 또는 인코딩하도록 구성된다. 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별하도록 더 구성된다. 하나 이상의 종속 타일은 코딩된 픽처 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 논리적 원자 단위를 형성한다. 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 하나 이상의 종속 타일을 포함하는 인-픽처-예측-가능 타일 세트의 표시를 구성하도록 더 구성된다. 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 비트스트림 및 인-픽처-예측-가능 타일 세트의 표시를 저장하게 하도록 더 구성된다.

[0016] 이러한 장치의 일부 구현예에서, 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금, 인-픽처-예측-가능 타일 세트의 표시를, 비트스트림과 관련된 시퀀스 파라미터 세트, 비트스트림과 관련된 픽처 파라미터 세트, 또는 비트스트림과 관련된 보충 개선 정보 메시지 중 하나 이상에 인코딩하도록 더 구성된다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트가 시간적 움직임 제한 타일 세트인지 여부를 표시한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트 중 하나 이상의 타일에 대한 다른 인-픽처 예측되지 않은 타일이 아닌 하나 이상의 타일의 예측 종속성의 세트를 표시하는 종속성 포레스트 또는 그래프를 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 종속 타일의 세트의 인트라 블록 카피 움직임 벡터의 세트에 대한 앵커 위치의 세트를 포함한다.

[0017] 또 다른 예시적인 실시예에서, 하나 이상의 타일을 포함하는 코딩된 픽처를 포함하는 비트스트림을 수신 또는

인코딩하기 위한 수단을 포함하는 장치가 제공된다. 장치는 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별하기 위한 수단을 더 포함한다. 하나 이상의 종속 타일은 코딩된 픽처 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 논리적 원자 단위를 형성한다. 장치는 하나 이상의 종속 타일을 포함하는 인-픽처-예측-가능 타일 세트의 표시를 구성하기 위한 수단을 더 포함한다. 장치는 비트스트림 및 인-픽처-예측-가능 타일 세트의 표시를 저장하도록 하는 수단을 더 포함한다.

[0018] 이러한 장치의 일부 구현예에서, 장치는 인-픽처-예측-가능 타일 세트의 표시를, 비트스트림과 관련된 시퀀스 파라미터 세트, 비트스트림과 관련된 픽처 파라미터 세트, 또는 비트스트림과 관련된 보충 개선 정보 메시지 중 하나 이상에 인코딩하는 수단을 더 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트가 시간적 움직임 제한 타일 세트인지 여부를 표시한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트 중 하나 이상의 타일에 대한 다른 인-픽처 예측되지 않은 타일이 아닌 하나 이상의 타일의 예측 종속성의 세트를 표시하는 종속성 포레스트 또는 그래프를 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 종속 타일의 세트의 인트라 블록 카피 움직임 벡터의 세트에 대한 앵커 위치의 세트를 포함한다.

[0019] 또 다른 예시적인 실시예에서, 컴퓨터 실행 가능 프로그램 코드 명령어가 저장된 적어도 하나의 비일시적 컴퓨터 판독 가능 저장 매체를 포함하는 컴퓨터 프로그램 제품이 제공되고, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 하나 이상의 타일을 포함하는 코딩된 픽처를 포함하는 비트스트림을 수신 또는 인코딩하도록 구성된 프로그램 코드 명령어를 포함한다. 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별하도록 구성된 프로그램 코드 명령어를 더 포함한다. 하나 이상의 종속 타일은 코딩된 픽처 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 논리적 원자 단위를 형성한다. 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 하나 이상의 종속 타일을 포함하는 인-픽처-예측-가능 타일 세트의 표시를 구성하도록 구성된 프로그램 코드 명령어를 더 포함한다. 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 비트스트림 및 인-픽처-예측-가능 타일 세트의 표시를 저장하게 하도록 구성되는 프로그램 코드 명령어를 더 포함한다.

[0020] 이러한 컴퓨터 프로그램 제품의 일부 구현예에서, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 인-픽처-예측-가능 타일 세트의 표시를, 비트스트림과 관련된 시퀀스 파라미터 세트, 비트스트림과 관련된 픽처 파라미터 세트, 또는 비트스트림과 관련된 보충 개선 정보 메시지 중 하나 이상에 인코딩하도록 구성되는 프로그램 코드 명령어를 더 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트가 시간적 움직임 제한 타일 세트인지 여부를 표시한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 하나 이상의 타일에 대한 다른 인-픽처 예측되지 않은 타일이 아닌 하나 이상의 타일의 예측 종속성의 세트를 표시하는 종속성 포레스트 또는 그래프를 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 종속 타일의 세트의 인트라 블록 카피 움직임 벡터의 세트에 대한 앵커 위치의 세트를 포함한다.

[0021] 일부 실시예에서, 코딩된 픽처를 디코딩하기 위해서 예시적인 실시예에 따라 방법, 장치 및 컴퓨터 프로그램 제품이 제공된다.

[0022] 하나의 예시적인 실시예에서, 하나 이상의 슬라이스를 포함하는 코딩된 픽처를 수신하는 것을 포함하는 방법이 제공된다. 방법은, 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더로부터, 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자를 디코딩하는 것을 더 포함한다. 방법은 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치로부터 시작하여 슬라이스의 슬라이스 데이터를 디코딩된 픽처로 디코딩하는 것을 더 포함한다.

[0023] 이러한 방법의 일부 구현예에서, 방법은, 슬라이스 헤더로부터, 하나 이상의 타일 중 하나의 타일에 관련된 블록 인덱스를 디코딩하는 것을 더 포함한다. 블록 인덱스는 타일 내의 블록에 대한 블록 위치를 표시하는 십액스 요소를 포함한다. 일부 실시예에서, 하나 이상의 타일 식별자 중 하나의 타일 식별자는 슬라이스의 사전 정의된 블록을 포함하는 타일을 표시한다. 일부 실시예에서, 방법은 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 디코딩하는 것을 더 포함한다. 하나 이상의 움직임 벡터는 하나 이상의 타일 식별자에 의해 표시되는 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용된다. 일부 실시예에서, 방법은 파라미터 세트 파일로부터 타일 위치를 표시하는 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 디코딩하는 것을 더 포함한다. 일부 실시예에서, 방법은 픽처와 관련된 디코딩 순서 제한 파일에 따라 하나 이상의 타일을 디코딩하는 것을 더 포함한다. 디코딩 순서 제한 파일은 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 하나 이상의 타일이 재정렬될 수 있는지 여부를 표시하는 재정렬 표시를 포함

한다.

[0024] 또 다른 예시적인 실시예에서, 적어도 하나의 프로세서 및 하나 이상의 프로그램에 대한 컴퓨터 프로그램 코드를 포함하는 적어도 하나의 메모리를 포함하는 장치가 제공되고, 적어도 하나의 메모리 및 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 적어도 하나 이상의 슬라이스를 포함하는 코딩된 픽처를 수신하도록 구성된다. 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금, 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더로부터, 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자를 디코딩하도록 더 구성된다. 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치로부터 시작하여 슬라이스의 슬라이스 데이터를 디코딩된 픽처로 디코딩하도록 구성된다.

[0025] 이러한 장치의 일부 구현예에서, 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금, 슬라이스 헤더로부터, 하나 이상의 타일 중 하나의 타일에 관련된 블록 인덱스를 디코딩하도록 더 구성된다. 블록 인덱스는 타일 내의 블록에 대한 블록 위치를 표시하는 선택스 요소를 포함한다. 일부 실시예에서, 하나 이상의 타일 식별자 중 하나의 타일 식별자는 슬라이스의 사전 정의된 블록을 포함하는 타일을 표시한다. 일부 실시예에서, 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 디코딩하도록 더 구성된다. 하나 이상의 움직임 벡터는 하나 이상의 타일 식별자에 의해 표시되는 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용된다. 일부 실시예에서, 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 파라미터 세트 파일로부터 타일 위치를 표시하는 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 디코딩하도록 더 구성된다. 일부 실시예에서, 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 픽처와 관련된 디코딩 순서 제한 파일에 따라 하나 이상의 타일을 디코딩하도록 더 구성된다. 디코딩 순서 제한 파일은 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 하나 이상의 타일이 재정렬될 수 있는지 여부를 표시하는 재정렬 표시를 포함한다.

[0026] 또 다른 예시적인 실시예에서, 하나 이상의 슬라이스를 포함하는 코딩된 픽처를 수신하기 위한 수단을 포함하는 장치가 제공된다. 장치는, 하나 이상의 슬라이스 중 하나의 슬라이스 헤더로부터, 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자를 디코딩하는 수단을 더 포함한다. 장치는 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치로부터 시작하여 슬라이스의 슬라이스 데이터를 디코딩된 픽처로 디코딩하기 위한 수단을 더 포함한다.

[0027] 이러한 장치의 일부 구현예에서, 장치는, 슬라이스 헤더로부터, 하나 이상의 타일 중 하나의 타일에 관련된 블록 인덱스를 디코딩하는 수단을 더 포함한다. 블록 인덱스는 타일 내의 블록에 대한 블록 위치를 표시하는 선택스 요소를 포함한다. 일부 실시예에서, 하나 이상의 타일 식별자 중 하나의 타일 식별자는 슬라이스의 사전 정의된 블록을 포함하는 타일을 표시한다. 일부 실시예에서, 장치는 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 디코딩하기 위한 수단을 더 포함한다. 하나 이상의 움직임 벡터는 하나 이상의 타일 식별자에 의해 표시되는 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용된다. 일부 실시예에서, 장치는 파라미터 세트 파일로부터 타일 위치를 표시하는 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 디코딩하기 위한 수단을 더 포함한다. 일부 실시예에서, 장치는 픽처와 관련된 디코딩 순서 제한 파일에 따라 하나 이상의 타일을 디코딩하기 위한 수단을 더 포함한다. 디코딩 순서 제한 파일은 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 하나 이상의 타일이 재정렬될 수 있는지 여부를 표시하는 재정렬 표시를 포함한다.

[0028] 또 다른 예시적인 실시예에서, 컴퓨터 실행 가능 프로그램 코드 명령어를 포함하는 적어도 하나의 비일시적 컴퓨터 판독 가능 저장 매체를 포함하는 컴퓨터 프로그램 제품이 제공되고, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 하나 이상의 슬라이스를 포함하는 코딩된 픽처를 수신하도록 구성되는 프로그램 코드 명령어를 포함한다. 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 하나 이상의 슬라이스 중 하나의 슬라이스의 슬라이스 헤더로부터, 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치를 표시하는 하나 이상의 타일 식별자를 디코딩하도록 구성되는 프로그램 코드 명령어를 더 포함한다. 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 슬라이스의 하나 이상의 타일에 대한 하나 이상의 타일 위치로부터 시작하여 슬라이스의 슬라이스 데이터를 디코딩된 픽처로 디코딩하도록 구성되는 프로그램 코드 명령어를 더 포함한다.

[0029]

이러한 컴퓨터 프로그램 제품의 일부 구현예에서, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 슬라이스 헤더로부터, 하나 이상의 타일 중 하나의 타일에 관련된 블록 인덱스를 디코딩하도록 구성된 프로그램 코드 명령어를 더 포함한다. 블록 인덱스는 타일 내의 블록에 대한 블록 위치를 표시하는 십진 요소를 포함한다. 일부 실시예에서, 하나 이상의 타일 식별자 중 하나의 타일 식별자는 슬라이스의 사전 정의된 블록을 포함하는 타일을 표시한다. 일부 실시예에서, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 디코딩하도록 구성된 프로그램 코드 명령어를 더 포함한다. 하나 이상의 움직임 벡터는 하나 이상의 타일 식별자에 의해 표시되는 하나 이상의 타일에 관련된 위치를 정의하는 하나 이상의 앵커 위치 식별자로부터 도출되는 하나 이상의 앵커 위치에 적용된다. 일부 실시예에서, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 파라미터 세트 파일로부터 타일 위치를 표시하는 하나 이상의 타일 식별자에 대한 하나 이상의 타일 식별자 값의 하나 이상의 할당을 디코딩하도록 구성된 프로그램 코드 명령어를 더 포함한다. 일부 실시예에서, 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 픽처와 관련된 디코딩 순서 제한 파일에 따라 하나 이상의 타일을 디코딩하도록 구성된 프로그램 코드 명령어를 더 포함한다. 디코딩 순서 제한 파일은 하나 이상의 타일에 대한 디코딩 가능성에 영향을 주지 않고 하나 이상의 타일이 재정렬될 수 있는지 여부를 표시하는 재정렬 표시를 포함한다.

[0030]

일부 실시예에서, 인-픽처-예측-가능(in-picture-prediction-enabled, IPPE) 타일 세트를 디코딩하기 위해서 예시적인 실시예에 따라 방법, 장치 및 컴퓨터 프로그램 제품이 제공된다. 하나의 예시적인 실시예에서, 하나 이상의 타일을 포함하는 코딩된 픽처를 포함하는 비트스트림을 수신하는 것을 포함하는 방법이 제공된다. 방법은 인-픽처-예측-가능 타일 세트의 표시를 수신하는 것을 더 포함한다. 인-픽처-예측-가능 타일 세트는 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별한다. 하나 이상의 종속 타일은 코딩된 픽처 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 논리적 원자 단위를 형성한다. 방법은 인-픽처-예측-가능 타일 세트의 표시에 따라 비트스트림을 디코딩하는 것을 더 포함한다.

[0031]

이러한 방법의 일부 구현예에서, 인-픽처-예측-가능 타일 세트의 표시는, 비트스트림과 관련된 시퀀스 파라미터 세트, 비트스트림과 관련된 픽처 파라미터 세트, 또는 비트스트림과 관련된 보충 개선 정보 메시지 중 하나 이상에 인코딩된다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트가 시간적 움직임 제한 타일 세트인지 여부를 표시한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 하나 이상의 타일에 대한 다른 인-픽처 예측되지 않은 타일이 아닌 하나 이상의 타일의 예측 종속성의 세트를 표시하는 종속성 포레스트 또는 그 래프를 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 종속 타일의 세트의 인트라 블록 카피 움직임 벡터의 세트에 대한 앵커 위치의 세트를 포함한다.

[0032]

또 다른 예시적인 실시예에서, 적어도 하나의 프로세서 및 하나 이상의 프로그램에 대한 컴퓨터 프로그램 코드를 포함하는 적어도 하나의 메모리를 포함하는 장치가 제공되고, 적어도 하나의 메모리 및 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 적어도 하나 이상의 타일을 포함하는 코딩된 픽처를 포함하는 비트스트림을 수신하도록 구성된다. 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 인-픽처-예측-가능 타일 세트의 표시를 수신하도록 구성된다. 인-픽처-예측-가능 타일 세트는 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별한다. 하나 이상의 종속 타일은 코딩된 픽처 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 논리적 원자 단위를 형성한다. 컴퓨터 프로그램 코드는, 적어도 하나의 프로세서와 함께, 장치로 하여금 인-픽처-예측-가능 타일 세트의 표시에 따라 비트스트림을 디코딩하도록 구성된다.

[0033]

이러한 장치의 일부 구현예에서, 인-픽처-예측-가능 타일 세트의 표시는, 비트스트림과 관련된 시퀀스 파라미터 세트, 비트스트림과 관련된 픽처 파라미터 세트, 또는 비트스트림과 관련된 보충 개선 정보 메시지 중 하나 이상에 인코딩된다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트가 시간적 움직임 제한 타일 세트인지 여부를 표시한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 하나 이상의 타일에 대한 다른 인-픽처 예측되지 않은 타일이 아닌 하나 이상의 타일의 예측 종속성의 세트를 표시하는 종속성 포레스트 또는 그 래프를 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 종속 타일의 세트의 인트라 블록 카피 움직임 벡터의 세트에 대한 앵커 위치의 세트를 포함한다.

[0034]

또 다른 예시적인 실시예에서, 하나 이상의 타일을 포함하는 코딩된 픽처를 포함하는 비트스트림을 수신하기 위한 수단을 포함하는 장치가 제공된다. 장치는 인-픽처-예측-가능 타일 세트의 표시를 수신하기 위한 수단을 더 포함한다. 인-픽처-예측-가능 타일 세트는 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별한다. 하나 이상의 종속 타일은 코딩된 픽처 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 논리적 원자 단위를 형성한다. 장치는 인-픽처-예측-가능 타일 세트의 표시에 따라 비트스트림을 디코딩하기 위한 수단을

더 포함한다.

[0035] 이러한 장치의 일부 구현예에서, 인-픽처-예측-가능 타일 세트의 표시는, 비트스트림과 관련된 시퀀스 파라미터 세트, 비트스트림과 관련된 픽처 파라미터 세트, 또는 비트스트림과 관련된 보충 개선 정보 메시지 중 하나 이상에 인코딩된다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트가 시간적 움직임 제한 타일 세트인지 여부를 표시한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 하나 이상의 타일에 대한 다른 인-픽처 예측되지 않은 타일이 아닌 하나 이상의 타일의 예측 종속성의 세트를 표시하는 종속성 포레스트 또는 그 그래프를 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 종속 타일의 세트라 블록 카피 움직임 벡터의 세트에 대한 앵커 위치의 세트를 포함한다.

[0036] 또 다른 예시적인 실시예에서, 컴퓨터 실행 가능 프로그램 코드 명령어가 저장된 적어도 하나의 비일시적 컴퓨터 판독 가능 저장 매체를 포함하는 장치가 제공되고, 컴퓨터 실행 가능 프로그램 코드는, 실행될 때, 하나 이상의 타일을 포함하는 코딩된 픽처를 포함하는 비트스트림을 수신하도록 구성된 프로그램 코드 명령어를 포함한다. 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 인-픽처-예측-가능 타일 세트의 표시를 수신하도록 구성된 프로그램 코드 명령어를 더 포함한다. 인-픽처-예측-가능 타일 세트는 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별한다. 하나 이상의 종속 타일은 코딩된 픽처 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 논리적 원자 단위를 형성한다. 컴퓨터 실행 가능 프로그램 코드 명령어는, 실행될 때, 인-픽처-예측-가능 타일 세트의 표시에 따라 비트스트림을 디코딩하도록 구성된 프로그램 코드 명령어를 더 포함한다.

[0037] 이러한 컴퓨터 프로그램 제품의 일부 구현예에서, 인-픽처-예측-가능 타일 세트의 표시는, 비트스트림과 관련된 시퀀스 파라미터 세트, 비트스트림과 관련된 픽처 파라미터 세트, 또는 비트스트림과 관련된 보충 개선 정보 메시지 중 하나 이상에 인코딩된다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트가 시간적 움직임 제한 타일 세트인지 여부를 표시한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 하나 이상의 타일에 대한 다른 인-픽처 예측되지 않은 타일이 아닌 하나 이상의 타일의 예측 종속성의 세트를 표시하는 종속성 포레스트 또는 그레프를 포함한다. 일부 실시예에서, 표시는 인-픽처-예측-가능 타일 세트의 종속 타일의 세트라 블록 카피 움직임 벡터의 세트에 대한 앵커 위치의 세트를 포함한다.

도면의 간단한 설명

[0038] 이와 같이 본 개시의 특정 예시적인 실시예를 일반적인 용어로 설명하였으므로, 이하에서는 첨부 도면을 참조할 것이며, 이는 반드시 같은 축척으로 도시된 것은 아니다.

도 1은 본 개시의 예시적인 실시예에 따라 구체적으로 구성될 수 있는 장치의 블록도이다.

도 2는 본 개시의 예시적인 실시예에 따라, 도 1의 장치와 같은 장치에 의해 수행되는 일련의 동작을 도시하는 흐름도이다.

도 3은 본 개시의 예시적인 실시예에 따른 예시적인 움직임 벡터를 도시한다.

도 4는 본 개시의 예시적인 실시예에 따라, 도 1의 장치와 같은 장치에 의해 수행되는 일련의 동작을 도시하는 흐름도이다.

도 5a 및 도 5b는 본 발명의 예시적인 실시예에 따라 구성 및 추출된 인-픽처-예측-가능 타일 세트의 예를 도시한다.

도 6은 본 개시의 예시적인 실시예에 따라, 도 1의 장치와 같은 장치에 의해 수행되는 일련의 동작을 도시하는 흐름도이다.

도 7은 본 개시의 예시적인 실시예에 따라, 도 1의 장치와 같은 장치에 의해 수행되는 일련의 동작을 도시하는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0039] 이하에서는 일부 실시예가 첨부된 도면을 참조하여 더 자세하게 설명될 것이고, 도면에는 본 발명의 전부는 아닌 일부 실시예가 도시된다. 실제로, 본 발명의 다양한 실시예는 많은 다른 형태로 구현될 수 있고 여기에 설명된 실시예에 한정되는 것으로 해석되어서는 안되며, 오히려, 이들 실시예가 제공되므로 본 개시는 적용 가능한 법적 요건을 만족할 것이다. 동일한 참조 번호는 전체적으로 동일한 요소를 지칭한다. 여기에서 사용되는

것과 같이, "데이터", "콘텐츠", "정보" 및 유사한 용어는 본 발명의 실시예에 따라 전송, 수신 및/또는 저장될 수 있는 데이터를 지칭하기 위해 상호 교환적으로 사용될 수 있다. 따라서, 이러한 용어의 사용이 본 발명의 실시예의 사상 및 범위를 한정하는 것으로 간주되어서는 안된다.

[0040] 또한, 본 명세서에서 사용되는 '회로'라는 용어는 (a) 하드웨어만의 회로 구현예(예를 들어, 아날로그 회로 및/또는 디지털 회로의 구현예), (b) 회로 및 함께 작동하여 장치로 하여금 여기에 설명된 하나 이상의 기능을 수행하도록 하는 하나 이상의 컴퓨터 판독 가능 메모리에 저장된 소프트웨어 및/또는 펌웨어 명령어를 포함하는 컴퓨터 프로그램 제품(들)의 조합, 및 (c) 소프트웨어 또는 펌웨어가 물리적으로 존재하지 않더라도 작동을 위해 소프트웨어 또는 펌웨어를 필요로 하는 마이크로프로세서(들) 또는 마이크로프로세서(들)의 일부와 같은 회로를 지칭한다. 이 '회로'의 정의는 임의의 청구 범위를 포함하여 여기에서 이 용어의 모든 사용에 적용된다. 다른 예로서, 여기에 사용되는 것과 같이, '회로'라는 용어는 또한 하나 이상의 프로세서 및/또는 그 일부(들) 및 수반되는 소프트웨어 및/또는 펌웨어를 포함하는 구현예를 포함한다. 또 다른 예로서, 여기에 사용되는 것과 같이, '회로'라는 용어는 또한, 예를 들어 모바일 폰을 위한 기저대역 접속 회로 또는 애플리케이션 프로세서 접속 회로 또는 서버, 셀룰러 네트워크 장치, 다른 네트워크 장치 및/또는 다른 컴퓨팅 장치의 유사한 접속 회로를 포함한다.

[0041] 여기에서 정의된 것과 같이, 비일시적 물리적 저장 매체(예를 들어, 휴발성 또는 비휘발성 메모리 장치)를 지칭하는 "컴퓨터 판독 가능 저장 매체"는 전자기 신호를 지칭하는 "컴퓨터 판독 가능 전송 매체"와 구별될 수 있다.

[0042] 비디오 인코딩에서 타일 상대 어드레싱을 가능하도록 하기 위해 예시적인 실시예에 따른 방법, 장치 및 컴퓨터 프로그램 제품이 제공된다. 방법, 장치 및 컴퓨터 프로그램 제품은 HEVC(High Efficiency Video Coding) 표준(HEVC 또는 H.265/HEVC) 또는 진행되고 있는 VVC(Versatile Video Coding) 표준(VVC, H.266 또는 H.266/VVC)을 포함하는 다양한 비디오 포맷과 함께 활용될 수 있다. 예시적인 실시예는 HEVC와 관련하여 설명되지만, 본 개시는 HEVC에 한정되지 않고, 오히려 본 개시의 예시적인 실시예가 부분적으로 또는 완전히 실시될 수 있는 하나의 가능한 기초로서 설명이 제공된다. 특히, 본 개시는 H.266 표준 초안과 함께 부분적으로 또는 완전히 실시될 수 있다.

[0043] 본 개시의 일부 측면은 ISO(International Standards Organization) 기본 미디어 파일 포맷(ISO/IEC 14496-12로서, 줄여서 ISOBMFF일 수 있음), MPEG(Moving Picture Experts Group)-4 파일 포맷(ISO/IEC 14496-14로서, MP4 포맷으로도 알려짐), NAL(Network Abstraction Layer) 단위 구조 비디오를 위한 파일 포맷(ISO/IEC 14496-15) 및 3GPP(3rd Generation Partner Project) 파일 포맷(3GPP Technical Specification 26.244로서, 3GP 포맷으로도 알려짐)과 같은 컨테이너 파일 포맷에 관한 것이다.

[0044] HEVC 및 예시적인 실시예를 설명할 때, 예를 들어 HEVC에 지정된 것과 같은 산술 연산자, 논리 연산자, 관계 연산자, 비트-단위 연산자, 할당 연산자 및 범위 표기법에 대한 일반적인 표기법이 사용될 수 있다. 또한, 예를 들어 HEVC에 지정된 것과 같은 일반적인 수학 함수가 사용될 수 있고, 예를 들어 HEVC에 지정된 것과 같은 연산자의 일반적인 순서 또는 우선 순위 및 실행 순서(왼쪽에서 오른쪽으로 또는 오른쪽에서 왼쪽으로)가 사용될 수 있다.

[0045] HEVC 및 예시적인 실시예를 설명할 때, 다음의 수학 함수가 사용될 수 있다. Ceil(x)는 x보다 크거나 같은 가장 작은 정수로 정의된다. Log2(x)는 x의 밑(base)이 2인 로그로 정의된다.

[0046] HEVC 및 예시적인 실시예를 설명할 때, 각각의 선택스 요소의 파싱 프로세스를 지정하기 위해 다음의 설명이 사용될 수 있다.

[0047] - u(n): n비트를 사용하는 부호없는 정수. 선택스 테이블에서 n이 "v"일 때, 비트 수는 다른 선택스 요소의 값에 따른 방식으로 달라진다. 이 디스크립터에 대한 파싱 프로세스는 최상위 비트가 먼저 기록되는 부호없는 정수의 이진 표현으로 해석되는 비트스트림으로부터 n개의 다음 비트로 지정된다.

[0048] - ue(v): 왼쪽 비트가 먼저 수행되는 부호없는 정수 Exponential-Golomb 코딩된 선택스 요소.

- [0049] Exponential-Golomb 비트 문자열은, 예를 들어 다음의 표를 사용하여 코드 번호(codeNum)로 변환될 수 있다.
- | Bit string | codeNum |
|---------------|---------|
| 1 | 0 |
| 0 1 0 | 1 |
| 0 1 1 | 2 |
| 0 0 1 0 0 | 3 |
| 0 0 1 0 1 | 4 |
| 0 0 1 1 0 | 5 |
| 0 0 1 1 1 | 6 |
| 0 0 0 1 0 0 0 | 7 |
| 0 0 0 1 0 0 1 | 8 |
| 0 0 0 1 0 1 0 | 9 |
| ... | ... |
- [0050]
- [0051] HEVC 및 예시적인 실시예를 설명할 때, 선택스 구조와 관련 부분만이 선택스 및 시맨틱스(semantics)의 설명에 포함될 수 있다. 다른 선택스 요소가 선택스 구조에 포함될 수 있음을 표시하기 위해 3개의 점('...')이 사용될 수 있다. 실시예는 선택스 요소의 다른 순서 및 이름으로 유사하게 실시될 수 있음이 이해될 필요가 있다.
- [0052] HEVC 및 예시적인 실시예를 설명할 때, 선택스 구조, 선택스 요소의 시맨틱스 및 디코딩 프로세스는 다음과 같이 지정될 수 있다. 비트스트림의 선택스 요소는 볼드(**bold**) 타입으로 표현된다. 각각의 선택스 요소는 이름(모두 밑줄이 있는 소문자), 선택적으로 하나 또는 두 개의 선택스 카테고리 및 코딩된 표현의 방법에 대한 하나 또는 두 개의 디스크립터로 기재된다. 디코딩 프로세스는 선택스 요소의 값과 이전에 디코딩된 선택스 요소의 값에 따라 동작한다. 선택스 요소의 값이 선택스 테이블 또는 텍스트에서 사용될 때, 선택스 요소의 이름은 소문자와 대문자가 혼합되고 밑줄이 없는 것으로 정해진다. 대문자로 시작하는 변수는 현재 선택스 구조 및 모든 종속 선택스 구조의 디코딩을 위해 도출된다. 대문자로 시작하는 변수는 변수의 원래의 선택스 구조에 관계 없이 나중의 선택스 구조에 대한 디코딩 프로세스에서 사용될 수 있다. 소문자로 시작하는 변수는 그것이 도출된 컨텍스트 내에서만 사용된다. 일부 경우에, 선택스 요소 값 또는 변수 값에 대한 "연상(mnemonic)" 이름이 그들의 숫자 값과 교환 가능하게 사용된다. 때때로 "연상" 이름은 관련 숫자 값 없이 사용된다. 값과 이름의 관련성은 텍스트에 지정되어 있다. 이름은 밑줄로 분리된 하나 이상의 문자 그룹으로부터 구성된다. 각각의 그룹은 대문자로 시작하고 더 많은 대문자를 포함할 수 있다.
- [0053] 예시적인 실시예를 설명할 때, 파일이라는 용어는 때때로 선택스 구조의 동의어 또는 선택스 구조의 인스턴스(instance)로 사용된다. 예를 들어, 구문(phrase) 파라미터 세트 파일은 파라미터 세트 선택스 구조 또는 파라미터 세트의 인스턴스(그 선택스 요소의 특정 값 포함)를 의미하는 데 사용될 수 있다. 다른 맥락에서, 파일이라는 용어는 컴퓨터 파일, 즉 저장소의 독립형 유닛을 형성하는 리소스를 의미하는 데 사용될 수 있다.
- [0054] HEVC 및 예시적인 실시예를 설명할 때, 선택스 구조는 다음을 사용하여 지정될 수 있다. 중괄호로 묶인 명령문 그룹은 복합 명령문이고 기능적으로 단일 명령문으로 처리된다. "while" 구조는 조건이 참인지 여부에 대한 테스트를 지정하고, 참인 경우, 조건이 더 이상 참이 아닐 때까지 명령문(또는 복합 명령문)의 평가를 반복적으로 지정한다. "do ... while" 구조는 명령문의 평가를 한 번 지정한 다음 조건이 참인지 여부를 테스트하고, 참인 경우, 조건이 더 이상 참이 아닐 때까지 명령문의 반복 평가를 지정한다. "if ... else" 구조는 조건이 참인지 여부에 대한 테스트를 지정하고, 조건이 참인 경우, 기본 명령문의 평가를 지정하고, 그렇지 않으면 다른 명령문의 평가를 지정한다. 다른 명령문 평가가 필요하지 않은 경우 구조 및 관련된 다른 명령문의 "else" 부분은 생략된다. "for" 구조는 초기 명령문의 평가를 지정한 다음 조건을 테스트하고, 조건이 참인 경우, 조건이 더 이상 참이 아닐 때까지 후속 명령문이 후속하는 기본 명령문의 반복 평가를 지정한다.
- [0055] 비디오 코덱은 입력 비디오를 저장/전송에 적합한 압축된 표현으로 변환하는 인코더 및 압축된 비디오 표현을 다시 볼 수 있는 형태로 압축 해제할 수 있는 디코더를 포함한다. 비디오 인코더 및/또는 비디오 디코더는 또한 서로 분리될 수 있으며, 즉 코덱을 형성할 필요는 없다. 전형적으로 인코더는 비디오를 보다 간결한 형태(즉, 낮은 비트 전송율)로 표현하기 위해 원본 비디오 시퀀스의 일부 정보를 버린다.
- [0056] 많은 하이브리드 비디오 인코더는 비디오 정보를 2 단계로 인코딩한다. 먼저, 특정 픽셀 영역(또는 "블록")의 픽셀 값은, 예를 들어 (코딩되는 블록에 밀접하게 대응하는 이전에 코딩된 비디오 프레임 중 하나의 영역을 찾아서 표시하는) 움직임 보상 수단 또는 (코딩될 블록 주변의 픽셀 값은 지정된 방식으로 사용하는) 공간적 수단에 의해 예측된다. 다음으로, 예측 오차, 즉 픽셀의 예측된 블록과 픽셀의 원본 블록 사이의 차이가 코딩된다. 이는 전형적으로 지정된 변환(예를 들어, DCT(Discrete Cosine Transform) 또는 이것의 변형)을 사용하여 픽셀

값의 차이를 변환하고, 계수를 양자화하고 양자화된 계수를 엔트로피 코딩하여 수행된다. 양자화 프로세스의 충실도(fidelity)를 변경함으로써, 인코더는 픽셀 표현의 정확도(화질)와 순출되는 코딩된 비디오 표현의 크기(파일 크기 또는 전송율) 사이의 균형을 제어할 수 있다.

[0057] 시간적 예측에서, 예측의 소스는 이전에 디코딩된 픽처(일명, 참조 픽처)이다. IBC(intra block copy; 일명, 인트라-블록-카피 예측)에서, 예측은 시간적 예측과 유사하게 적용되지만 참조 픽처는 현재 픽처이고 이전에 디코딩된 샘플만이 예측 프로세스에서 참조될 수 있다. 인터-레이어 또는 인터-뷰 예측은 시간적 예측과 유사하게 적용될 수 있지만, 참조 픽처는 각각 또 다른 스케일러블 레이어 또는 또 다른 뷰로부터 디코딩된 픽처이다. 일부 경우에, 인터 예측은 시간적 예측만을 지칭할 수 있고, 다른 경우에 인터 예측은 시간적 예측과 동일 또는 유사한 프로세스로 수행되는 경우 인트라 블록 카피, 인터-레이어 예측 및 인터-뷰 예측 중 임의의 것과 시간적 예측을 총괄적으로 지칭할 수 있다. 인터 예측 또는 시간적 예측은 때때로 움직임 보상 또는 움직임-보상된 예측으로 지칭될 수 있다.

[0058] 시간적 예측, 움직임 보상 또는 움직임-보상된 예측이라고도 지칭될 수 있는 인터 예측은 시간적 중복성을 감소시킨다. 인터 예측에서 예측의 소스는 이전에 디코딩된 픽처이다. 인트라 예측은 동일한 픽처 내의 인접 픽셀이 상관될 가능성이 있다는 사실을 활용한다. 인트라 예측은 공간 또는 변환 도메인에서 수행될 수 있다. 즉, 샘플 값 또는 변환 계수가 예측될 수 있다. 공간 도메인에서 적용될 때, 인접 이웃하는 블록의 샘플 값은 현재 블록의 샘플 값을 예측하는 데 사용된다. 인트라 예측은 전형적으로 인트라 코딩에서 활용되고, 여기에 인터 예측은 적용되지 않는다.

[0059] 코딩 절차의 하나의 결과는 움직임 벡터 및 양자화된 변환 계수와 같은 코딩 파라미터 세트이다. 많은 파라미터가 공간적으로 또는 시간적으로 이웃하는 파라미터로부터 먼저 예측되는 경우 더 효율적으로 엔트로피 코딩될 수 있다. 예를 들어, 움직임 벡터는 공간적으로 인접한 움직임 벡터로부터 예측될 수 있고 움직임 벡터 예측자(predictor)에 대한 차이만이 코딩될 수 있다.

[0060] 움직임 벡터는 인코딩 또는 디코딩되고 있는 현재 예측 단위와 같이 예측되고 있는 블록의 위치에 대한 참조 픽처 내의 예측 블록의 위치를 표시하기 위해 인터 예측에 사용되는 2차원 유클리드 벡터로 간주될 수 있다. 움직임 벡터는 단편적인(fractional) 샘플 위치를 가리킬 수 있고, 이 경우 예측 블록의 샘플 값이 참조 픽처의 샘플 값으로부터 보간될 수 있다.

[0061] 인-픽처 예측은, 코딩 파라미터의 공간 예측(예를 들어, 공간 움직임 벡터 예측), 인트라 예측, 인트라 블록 복사 예측, 블록에 걸친 인-루프 필터링 및 컨텍스트 적용 엔트로피 코딩에서 블록에 걸친 컨텍스트 상태 업데이트 중 하나 이상을 집합적으로 포함하도록 정의될 수 있지만 반드시 이에 한정되는 것은 아니다. 인-픽처 예측의 정의는 용어를 참조하는 맥락에 따라 조정될 수 있다.

[0062] 디코더는 인코더와 유사한 예측 수단을 적용하여 픽셀 블록의 예측된 표현(인코더에 의해 생성되고 압축된 표현으로 저장된 움직임 또는 공간 정보를 사용)을 형성하고 예측 오류 디코딩(공간 픽셀 도메인에서 양자화된 예측 에러 신호를 복구하는 예측 에러 코딩의 역방향 동작)을 수행하여 출력 비디오를 재구성한다. 예측 수단 및 예측 오류 디코딩 수단을 적용한 이후 디코더는 예측 및 예측 오류 신호(픽셀 값)를 합산하여 출력 비디오 프레임을 형성한다. 디코더(및 인코더)는 또한 디스플레이를 위해 출력 비디오를 전달하고 및/또는 비디오 시퀀스에서 이후의 프레임에 대한 예측 참조로서 출력 비디오를 저장하기 이전에 출력 비디오의 품질을 개선하기 위해 추가 필터링 수단을 적용할 수 있다. 필터링은, 예를 들어 디블로킹(deblocking), SAO(sample adaptive offset) 및/또는 ALF(adaptive loop filtering) 중 하나 이상을 포함할 수 있다.

[0063] 전형적인 비디오 코덱에서 움직임 정보는 예측 단위와 같은 각각의 움직임 보상된 이미지 블록과 관련된 움직임 벡터로 표시된다. 이를 움직임 벡터 각각은 (인코더 측에서) 코딩될 픽처 또는 (디코더 측에서) 디코딩될 픽처의 이미지 블록의 변위 및 이전에 코딩된 픽처 또는 디코딩된 픽처 중 하나의 예측 소스 블록을 표현한다. 움직임 벡터를 효율적으로 표현하기 위해 이들은 전형적으로 블록 지정 예측된 움직임 벡터에 대해 차등적으로 코딩된다. 전형적인 비디오 코덱에서 예측된 움직임 벡터는, 예를 들어 인접 블록의 인코딩된 또는 디코딩된 움직임 벡터의 중앙값을 계산하는 것과 같이 사전 정의된 방식으로 생성된다. 움직임 벡터 예측을 생성하는 또 다른 방법은 시간적 참조 픽처에서 인접 블록 및/또는 함께 위치하는(co-located) 블록으로부터 후보 예측의 리스트를 생성하고 선택된 후보를 움직임 벡터 예측자(predictor)로 시그널링하는 것이다. 움직임 벡터 값을 예측하는 것에 더하여, 움직임 보상된 예측에 어떤 참조 픽처(들)가 사용되는지 예측할 수 있고, 이 예측 정보는, 예를 들어 이전에 코딩된/디코딩된 픽처의 참조 인덱스로 표현될 수 있다. 참조 인덱스는 전형적으로 시간적 참조 픽처의 인접 블록 및/또는 함께 위치하는 블록으로부터 예측된다. 또한, 전형적인 고효율 비디오 코덱은

종종 병합/병합된 모드라고 불리는 추가 움직임 정보 코딩/디코딩 메커니즘을 사용하는데, 여기에서 움직임 벡터 및 각각의 이용 가능한 참조 픽처 리스트에 대한 대응하는 참조 픽처 인덱스를 포함하는 모든 움직임 필드 정보가 예측되고 아무런 수정/보정없이 사용된다. 유사하게, 움직임 필드 정보를 예측하는 것은 시간적 참조 픽처의 인접 블록 및/또는 함께 위치하는 블록의 움직임 필드 정보를 사용하여 수행되고, 사용된 움직임 필드 정보는 이용 가능한 인접/함께 위치하는 블록의 움직임 필드 정보로 채워진 움직임 필드 후보 리스트 중 하나의 리스트에 시그널링된다.

[0064] 전형적인 비디오 코덱에서 움직임 보상 이후의 예측 잔차는 먼저 (DCT와 같은) 변환 커널로 변환된 다음 코딩된다. 그 이유는 잔차와 변환 사이에 여전히 약간의 상관 관계가 존재하는 것이 많은 경우에 이 상관 관계를 줄이고 효율적인 코딩을 제공하는데 도움을 줄 수 있다.

[0065] 움직임 벡터 및 양자화된 변환 계수와 같은 코딩 파라미터는 비트스트림으로 무손실 인코딩된다. 이 무손실 인코딩은 엔트로피 코딩 또는 엔트로피 인코딩으로 지칭될 수 있다. 비트스트림을 코딩 파라미터로 디코딩하는 역 동작(reverse operation)은 엔트로피 디코딩으로 지칭될 수 있다. 엔트로피 코딩/디코딩은 많은 방식으로 수행될 수 있다. 예를 들어, 컨텍스트-기반 코딩/디코딩이 적용될 수 있고, 인코더 및 디코더 모두는 이전에 코딩된/디코딩된 코딩 파라미터에 근거하여 코딩 파라미터의 컨텍스트 상태를 수정한다. 컨텍스트-기반 코딩은, 예를 들어 CABAC(context adaptive binary arithmetic coding) 또는 CA VLC(context-based variable length coding) 또는 임의의 유사한 엔트로피 코딩일 수 있다. 엔트로피 코딩/디코딩은 이와 달리 또는 추가적으로 Huffman 코딩/디코딩 또는 Exp-Golomb 코딩/디코딩과 같은 가변 길이 코딩 방식을 사용하여 수행될 수 있다. 엔트로피 코딩된 비트스트림 또는 코드워드로부터 코딩 파라미터를 디코딩하는 것은 파싱이라고 지칭될 수 있다.

[0066] 파티셔닝은 세트의 각각의 요소가 정확히 하나의 서브세트에 존재하도록 세트를 서브세트로 분할하는 것으로 정의될 수 있다.

[0067] HEVC에서, 비디오 픽처는 픽처의 영역을 커버하는 코딩 단위(CU)로 분할된다. CU는 CU 내의 샘플에 대한 예측 프로세스를 정의하는 하나 이상의 예측 단위(PU) 및 CU의 샘플에 대한 예측 오류 코딩 프로세스를 정의하는 하나 이상의 변환 단위(TU)로 구성된다. CU는 가능한 CU 크기의 사전 정의된 세트로부터 선택할 수 있는 크기를 갖는 샘플의 정사각형 블록으로 구성된다. 최대 허용 크기를 갖는 CU는 LCU 또는 CTU로 지칭될 수 있고, 비디오 픽처는 비중첩 LCU로 분할된다. LCU는, 예를 들어 LCU와 산출된 CU를 반복적으로 분할하여 더 작은 CU의 조합으로 더 분할될 수 있다. 각각의 산출되는 CU는 CU와 관련된 적어도 하나의 PU 및 적어도 하나의 TU를 가질 수 있다. 각각의 PU 및 TU는 각각 예측 및 예측 오류 코딩 프로세스의 입도를 높이기 위해 더 작은 PU 및 TU로 더 분할될 수 있다. 각각의 PU는 PU 내의 픽셀에 대해 어떤 종류의 예측이 적용되는지를 정의하는 PU와 관련된 예측 정보(예를 들어, 인터 예측된 PU에 대한 움직임 벡터 정보 및 인트라 예측된 PU에 대한 인트라 예측 방향성 정보)를 갖는다.

[0068] 코딩 블록은 일부 값의 N에 대해 샘플의 NxN 블록으로 정의될 수 있으므로 코딩 트리 블록을 코딩 블록으로 분할하는 것은 파티셔닝이다. CTB(coding tree block)는 일부 값의 N에 대해 샘플의 NxN 블록으로 정의될 수 있으므로 컴포넌트를 코딩 트리 블록으로 분할하는 것은 파티셔닝이다. CTU(coding tree unit)는 3개의 샘플 어레이를 갖는 픽처의 루마 샘플의 코딩 트리 블록 및 크로마 샘플의 2개의 대응하는 코딩 트리 블록, 또는 단색 픽처 또는 3개의 개별 색상 평면을 사용하여 코딩되는 픽처의 코딩 트리 블록 및 샘플을 코딩하는 데 사용되는 신팩스 구조로 정의될 수 있다. CU(coding unit)는 3개의 샘플 어레이를 갖는 픽처의 루마 샘플의 코딩 블록 및 크로마 샘플의 2개의 대응하는 코딩 블록, 또는 단색 픽처 또는 3개의 개별 색상 평면을 사용하여 코딩되는 픽처의 코딩 블록 및 샘플을 코딩하는 데 사용되는 신팩스 구조로 정의될 수 있다.

[0069] 각각의 TU는 해당 TU 내의 샘플에 대한 예측 오류 디코딩 프로세스를 설명하는 정보(예를 들어, 이산 코사인 변환 계수 정보를 포함)와 관련될 수 있다. 예측 오류 코딩이 각각의 CU에 대해 적용되는지 여부는 CU 레벨에서 시그널링된다. CU와 관련된 예측 오차 잔차가 없는 경우, CU에 대한 TU가 없다고 볼 수 있다. 이미지를 CU로 분할하고, CU를 PU 및 TU로 분할하는 것은 디코더가 이를 단위의 의도된 구조를 재현할 수 있도록 하기 위해 비트스트림에서 시그널링될 수 있다.

[0070] HEVC 표준에서, 픽처는 직사각형이고 정수 개수의 CTU를 포함하는 타일로 파티션될 수 있다. HEVC 표준에서, 타일로의 파티셔닝은 (CTU의) 타일 열 너비의 리스트 및 (CTU의) 타일 행 높이의 리스트로 특정될 수 있는 그리드를 형성한다. 타일은 타일 그리드의 래스터 스캔 순서로 비트스트림에서 연속적으로 정렬된다. 타일은 정수의 개수의 슬라이스(slice)를 포함할 수 있다. 인트라 예측, 인트라 블록 카피 예측 및 엔트로피 코딩을 위한

컨텍스트 상태 예측과 같은 특정 타입의 인-픽처 예측은 타일에 걸쳐 불가능하게 된다. 인코더는 디블로킹 필터링과 같은 인-루프 필터링이 타일에 걸쳐 적용되는지 여부를 제어하고 비트스트림에 표시할 수 있다.

[0071] 비디오 코딩 표준 및 사양은 인코더가 코딩된 픽처를 코딩된 슬라이스 등으로 분할하도록 할 수 있다. 인-픽처 예측은 전형적으로 슬라이스 경계에 걸쳐 불가능하게 된다. 따라서, 슬라이스는 코딩된 픽처를 독립적으로 디코딩 가능한 조각으로 분할하는 방법으로 간주될 수 있다. HEVC에서, 인-픽처 예측은 슬라이스 경계에 걸쳐 불가능하게 될 수 있다. 따라서, 슬라이스는 코딩된 픽처를 독립적으로 디코딩 가능한 조각으로 분할하는 방법으로 간주될 수 있고, 따라서 슬라이스는 종종 전송을 위한 기본 단위로 간주된다. 많은 경우에, 인코더는 슬라이스 경계에 걸쳐 어떤 타입의 인-픽처 예측이 꺼져 있는지 비트스트림에 표시할 수 있고, 디코더 동작은, 예를 들어 어떤 예측 소스가 이용 가능한지 결정할 때 이 정보를 고려한다. 예를 들어, 이웃하는 CU가 다른 슬라이스에 있는 경우, 이웃하는 CU로부터의 샘플은 인트라 예측에 이용 불가능한 것으로 간주될 수 있다.

[0072] HEVC에서, 슬라이스는 정수 개수의 CTU로 구성된다. CTU는 타일 내에서 또는 타일이 사용되지 않는 경우 픽처 내에서 CTU의 래스터 스캔 순서로 스캔된다. 슬라이스는 정수 개수의 타일을 포함하거나 슬라이스는 타일에 포함될 수 있다. CTU 내에서, CU는 특정 스캔 순서를 갖는다.

[0073] HEVC에서, 슬라이스는 하나의 독립 슬라이스 세그먼트 및 같은 액세스 단위 내에서 (존재하는 경우) 다음 독립 슬라이스 세그먼트에 선행하는 (존재하는 경우) 모든 후속하는 종속 슬라이스 세그먼트에 포함된 정수 개수의 코딩 트리 단위로 정의된다. HEVC에서, 슬라이스 세그먼트는 타일 스캔에서 연속적으로 정렬되고 단일 NAL 단위에 포함되는 코딩 트리 단위의 정수 개수로 정의된다. 각각의 픽처를 슬라이스 세그먼트로 나누는 것은 파티셔닝이다. HEVC에서, 독립 슬라이스 세그먼트는 슬라이스 세그먼트 헤더의 신팩스 요소의 값이 선행하는 슬라이스 세그먼트에 대한 값으로부터 도출되지 않는 슬라이스 세그먼트로 정의되고, 종속 슬라이스 세그먼트는 슬라이스 세그먼트 헤더의 일부 신팩스 요소의 값이 디코딩 순서에서 선행하는 독립 슬라이스 세그먼트에 대한 값으로부터 도출되는 세그먼트 슬라이스로 정의된다. HEVC에서, 슬라이스 헤더는 현재 슬라이스 세그먼트 또는 현재 종속 슬라이스 세그먼트에 선행하는 독립 슬라이스 세그먼트인 독립 슬라이스 세그먼트의 슬라이스 세그먼트 헤더로 정의되고, 슬라이스 세그먼트 헤더는 슬라이스 세그먼트로 표현된 제 1 또는 모든 코딩 트리 단위에 속하는 데이터 요소를 포함하는 코딩된 슬라이스 세그먼트의 일부로 정의된다. CU는 타일 내에서 또는 타일이 사용되지 않는 경우 픽처 내의 LCU의 래스터 스캔 순서로 스캔된다. LCU 내에서, CU는 특정 스캔 순서를 갖는다.

[0074] MCTS(motion-constrained tile set)은 인코딩에서 인터 예측 프로세스가 제한되므로 움직임 제한 타일 세트의 외부의 샘플 값 및 움직임 제한 타일 세트의 외부의 하나 이상의 샘플 값을 사용하여 도출된 단편적인 (fractional) 샘플 위치에서의 샘플 값은 움직임 제한 타일 세트 내의 임의의 샘플의 인터 예측에 사용되지 않는다. 또한, MCTS의 인코딩은 MCTS 외부의 신팩스 요소 또는 변수가 MCTS를 디코딩하는 데 사용되지 않는 방식으로 제한된다. 예를 들어, MCTS의 인코딩은 움직임 벡터 후보가 MCTS 외부의 블록으로부터 도출되지 않는 방식으로 제한된다. 이것은 HEVC의 TMVP(temporal motion vector prediction)를 끄거나 또는 인코더가 MCTS의 오른쪽 아래에 있는 마지막 하나를 제외하고 MCTS의 오른쪽 타일 경계의 바로 왼쪽에 위치하는 PU에 대한 병합 또는 AMVP(advanced motion vector prediction) 후보 리스트에서 TMVP 후보 또는 TMVP 후보에 후속하는 임의의 움직임 벡터 예측 후보를 사용하는 것을 허용하지 않음으로써 강제될 수 있다. 일반적으로, MCTS는 MCTS 외부에 있는, 움직임 벡터와 같은, 임의의 샘플 값 및 코딩된 데이터와는 독립적인 타일 세트로 정의될 수 있다. 일부 경우에는, 직사각형 영역을 형성하기 위해 MCTS가 필요할 수 있다. 맥락에 따라, MCTS는 픽처 내의 타일 세트 또는 픽처의 시퀀스의 각각의 타일 세트를 지칭할 수 있다는 것이 이해되어야 한다. 각각의 타일 세트는 픽처의 시퀀스에 나란히 배치될 수 있지만, 일반적으로 그럴 필요는 없다.

[0075] 인터 예측에 사용되는 샘플 위치는 인코딩 및/또는 디코딩 프로세스에 의해 포화될 수 있으므로, 픽처의 외부일 수 있는 위치는 다르게 포화되어 픽처의 대응하는 경계 샘플을 지시한다. 따라서, 타일 경계가 또한 픽처 경계 이기도 한 경우, 일부 사용 예에서, 샘플 위치가 경계 상에서 포화되므로, 인코더는 움직임 벡터가 해당 경계를 효과적으로 교차하거나 또는 움직임 벡터가 해당 경계 외부의 위치를 참조하는 단편적인 샘플을 효과적으로 보간하도록 허용할 수 있다. 다른 사용 예에서, 특히 코딩된 타일이 픽처 경계에 인접한 위치에 위치하는 비트스트림으로부터 픽처 경계에 인접하지 않은 위치에 위치하는 또 다른 비트스트림으로 추출될 수 있는 경우, 인코더는 움직임 벡터를 임의의 MCTS 경계와 유사하게 픽처 경계 상으로 제한할 수 있다.

[0076] HEVC의 시간적 움직임 제한 타일 세트 SEI(supplemental enhancement information) 메시지는 움직임 제한 타일 세트의 존재를 비트스트림에 표시하는 데 사용될 수 있다.

- [0077] MCTS와 관련하여 일부 예 및 실시예가 설명되었지만, 이들은 독립적으로 디코딩 가능한 시공간 단위의 다른 유사한 개념으로 유사하게 실시될 수 있다는 것이 이해될 필요가 있다. 또한, 이러한 시공간 단위에 대한 움직임 제한은 위의 MCTS와 유사하게 지정될 수 있다. 이러한 시공간 단위의 예는 움직임 제한 슬라이스 및 움직임 제한 픽처를 포함하지만 이에 한정되는 것은 아니다. 움직임 제한 슬라이스는 인코딩에서 인터 예측 프로세스가 제한되므로 움직임 제한 슬라이스의 외부의 신팩스나 또는 도출된 변수, 움직임 제한 슬라이스 외부의 샘플 값, 움직임 제한 슬라이스의 외부의 하나 이상의 샘플 값을 사용하여 도출된 단편적인 샘플 위치에서의 샘플 값은 움직임 제한 슬라이스 내의 임의의 샘플의 인터 예측에 사용되지 않는다. 시공간 단위라는 문구가 단일 시간 인스턴스의 맵에서 사용될 때, 이는 코딩된 픽처의 특정 서브세트 및 디코딩될 때에는 디코딩된 픽처 영역의 특정 서브세트에 대응하는 공간 단위로 간주될 수 있다.
- [0078] HEVC 인코더의 출력 및 HEVC 디코더의 입력을 위한 기본 단위는 NAL(Network Abstraction Layer) 단위이다. 패킷 지향 네트워크를 통한 전송 또는 구조화된 파일로의 저장을 위해, NAL 단위는 패킷 또는 유사한 구조로 캡슐화될 수 있다. 프레이밍 구조를 제공하지 않는 전송 또는 저장 환경을 위해 HEVC에 바이트스트림(bytestream) 포맷이 지정되어 있다. 바이트스트림 포맷은 각각의 NAL 단위 앞에 시작 코드를 첨부하여 NAL 단위를 서로 구분한다. NAL 단위 경계를 잘못 감지하지 않도록, 인코더는 바이트 지향 시작 코드에 블레이션 방지 알고리즘을 실행하여 시작 코드가 다르게 발생한 경우 NAL 단위 페이로드에 블레이션 방지 바이트를 추가한다. 패킷 지향 시스템 및 스트림 지향 시스템 사이의 간단한 게이트웨이 동작을 가능하도록 하기 위해, 바이트스트림 포맷의 사용 여부에 관계없이 항상 시작 코드 블레이션 방지가 수행될 수 있다. NAL 단위는 후속하는 데이터 탑의 표시를 포함하는 신팩스 구조 및 블레이션 방지 바이트에 필요한 산재된 RBSP(raw byte sequence payload)의 형태로 해당 데이터를 포함하는 바이트로 정의될 수 있다. RBSP는 NAL 단위로 캡슐화된 정수 개수의 바이트를 포함하는 신팩스 구조로 정의될 수 있다. RBSP는 비어 있거나 또는 RBSP 정지 비트가 후속하고 0의 0개 이상의 후속 비트가 후속하는 신팩스 요소를 포함하는 데이터 비트의 문자열의 형태를 갖는다.
- [0079] NAL 단위는 헤더와 페이로드로 구성된다. HEVC에서, 지정된 모든 NAL 단위 탑에 대해 2-바이트의 NAL 단위 헤더가 사용된다. NAL 단위 헤더는 예비된 1-비트, 6-비트의 NAL 단위 탑 표시, 시간적 레벨을 표시하는 3-비트의 nuh_temporal_id_plus1(1보다 크거나 같을 필요가 있을 수 있음) 및 6-비트의 nuh_layer_id 신팩스 요소를 포함한다. temporal_id_plus1 신팩스 요소는 NAL 단위에 대한 시간적 식별자로 간주될 수 있고, 제로 기반 TemporalId 변수는 TemporalId = temporal_id_plus1 - 1로 도출될 수 있다. 약어 TID는 TemporalId 변수와 상호 교환적으로 사용될 수 있다. TemporalId가 0인 경우, 이는 가장 낮은 시간적 레벨에 대응한다. 2개의 NAL 단위 헤더 바이트와 관련된 시작 코드 블레이션을 피하기 위해서는 temporal_id_plus1의 값은 0이 아닐 필요가 있다. 선택된 값보다 크거나 같은 TemporalId를 갖는 모든 VCL NAL 단위를 제외하고 그 외 모든 VCL NAL 단위를 포함하는 비트스트림은 적합하다. 결과적으로, tid_value와 같은 TemporalId를 갖는 픽처는 tid_value보다 큰 TemporalId를 갖는 어떤 픽처도 인터 예측 참조로 사용하지 않는다. 서브-레이어 또는 시간적 서브-레이어는 시간적 스케일 가능(scalable) 비트스트림의 시간적 스케일 가능 레이어(또는 시간적 레이어, TL)로 정의될 수 있고, TemporalId 변수의 특정 값을 갖는 VCL NAL 단위 및 관련된 비-VCL NAL 단위로 구성된다. nuh_layer_id는 스케일 가능성(scalability) 레이어 식별자로 이해될 수 있다.
- [0080] NAL 단위는 VCL(Video Coding Layer) NAL 단위 및 비-VCL NAL 단위로 분류될 수 있다. VCL NAL 단위는 전형적으로 코딩된 슬라이스 NAL 단위이다. HEVC에서, VCL NAL 단위는 하나 이상의 CU를 표현하는 신팩스 요소를 포함한다. HEVC에서, 특정 범위 내의 NAL 단위 탑은 VCL NAL 단위를 표시하고, VCL NAL 단위 탑은 픽처 탑을 표시한다.
- [0081] 비-VCL NAL 단위는, 예를 들어 시퀀스 파라미터 세트, 픽처 파라미터 세트, SEI(supplemental enhancement information) NAL 단위, 액세스 단위 구분자(delimiter), 시퀀스 NAL 단위의 끝, 비트스트림 NAL 단위의 끝, 또는 필터 데이터 NAL 단위 중 한 탑일 수 있다. 파라미터 세트는 디코딩된 픽처의 재구성을 위해 필요할 수 있지만, 다른 많은 비-VCL NAL 단위는 디코딩된 샘플 값의 재구성에 필요하지 않다.
- [0082] 코딩된 비디오 시퀀스를 통해 변경되지 않은 파라미터는 시퀀스 파라미터 세트에 포함될 수 있다. 디코딩 프로세스에 필요할 수 있는 파라미터 외에도, 시퀀스 파라미터 세트는 버퍼링, 픽처 출력 탑, 렌더링 및 리소스 예약에 중요할 수 있는 파라미터를 포함하는 VUI(video usability information)를 선택적으로 포함할 수 있다. HEVC에서 시퀀스 파라미터 세트 RBSP는 하나 이상의 픽처 파라미터 세트 RBSP 또는 버퍼링 주기 SEI 메시지를 포함하는 하나 이상의 SEI NAL 단위에 의해 참조될 수 있는 파라미터를 포함한다. 픽처 파라미터 세트는 여러 개의 코딩된 픽처에서 변경되지 않을 가능성이 큰 파라미터를 포함한다. 픽처 파라미터 세트 RBSP는 하나 이상

의 코딩된 픽처의 코딩된 슬라이스 NAL 단위에 의해 참조될 수 있는 파라미터를 포함할 수 있다.

[0083] HEVC에서, 비디오 파라미터 세트(video parameter set, VPS)는, 각각의 슬라이스 세그먼트 헤더에서 발견되는 신택스 요소에 의해 참조되는 PPS(picture parameter set)에서 발견되는 신택스 요소에 의해 참조되는 SPS(sequence parameter set)에서 발견되는 신택스 요소의 콘텐츠에 의해 결정되는 0개 이상의 전체 코딩된 비디오 시퀀스에 적용되는 신택스 요소를 포함하는 신택스 구조로 정의될 수 있다.

[0084] 비디오 파라미터 세트 RBSP는 하나 이상의 시퀀스 파라미터 세트 RBSP에 의해 참조될 수 있는 파라미터를 포함할 수 있다.

[0085] VPS(video parameter set), SPS 및 PPS 사이의 관계 및 계층은 다음과 같이 설명될 수 있다. VPS는 파라미터 세트 계층에서 및 스케일 가능성 및/또는 3D 비디오의 맥락에서 SPS보다 한 레벨 위에 있다. VPS는 전체 코딩된 비디오 시퀀스의 모든(스케일 가능성 또는 뷰) 레이어에 걸쳐 모든 슬라이스에 공통적인 파라미터를 포함할 수 있다. SPS는 전체 코딩된 비디오 시퀀스의 특정(스케일 가능성 또는 뷰) 레이어의 모든 슬라이스에 공통적인 파라미터를 포함하고, 복수(스케일 가능성 또는 뷰) 레이어에 의해 공유될 수 있다. PPS는 특정 레이어 표현(하나의 액세스 단위에서 하나의 스케일 가능성 또는 뷰 레이어의 표현)의 모든 슬라이스에 대해 공통적이고 복수 레이어 표현의 모든 슬라이스에 의해 공유될 가능성이 큰 파라미터를 포함한다.

[0086] VPS는 전체 코딩된 비디오 시퀀스의 모든(스케일 가능성 또는 뷰) 레이어에 걸쳐 모든 슬라이스에 적용될 수 있는 많은 다른 정보뿐만 아니라 비트스트림에서 레이어의 종속성 관계에 대한 정보를 제공할 수 있다. VPS는 기본 VPS와 VPS 확장의 2개의 부분을 포함하는 것으로 간주될 수 있고, VPS 확장은 선택적으로 존재할 수 있다.

[0087] 대역-외 전송, 시그널링 또는 저장은 액세스의 용이성 또는 세션 협상과 같은 전송 오류에 대한 내성(tolerance) 이외의 다른 목적으로 추가적으로 또는 이와는 달리 사용될 수 있다. 예를 들어, ISO 기본 미디어 파일 포맷(ISO Base Media File Format)을 준수하는 파일에서 트랙의 샘플 엔트리는 파라미터 세트를 포함할 수 있는 반면, 비트스트림의 코딩된 데이터는 파일의 다른 위치 또는 다른 파일에 저장된다. 비트스트림을 따라라는 문구(예를 들어, 비트스트림을 따라 표시하는) 또는 비트스트림의 코딩된 단위를 따라라는 문구(예를 들어, 코딩된 타일을 따라 표시하는)는 청구 범위 및 설명된 실시예에서 대역-외 데이터가 각각 비트스트림 또는 코딩된 단위와 관련되는 방식으로의 대역-외 전송, 시그널링 또는 저장을 지칭하는 데 사용될 수 있다. 비트스트림을 따라 또는 비트스트림의 코딩된 단위를 따라 디코딩이라는 문구는 각각 비트스트림 또는 코딩된 단위와 관련되는 참조된 대역-외 데이터(대역-외 전송, 시그널링 또는 저장으로부터 획득될 수 있음)를 디코딩하는 것을 지칭할 수 있다.

[0088] SEI NAL 단위는 출력 픽처의 디코딩에 필요하지 않지만 픽처 출력 타이밍, 렌더링, 오류 검출, 오류 은닉 및 리소스 예약과 같은 관련 프로세스를 지원할 수 있는 하나 이상의 SEI 메시지를 포함할 수 있다. 몇몇의 SEI 메시지가 HEVC에 지정되어 있고, 사용자 데이터 SEI 메시지는 기관 및 회사가 자체 사용을 위해 SEI 메시지를 지정할 수 있도록 한다. HEVC는 지정된 SEI 메시지에 대한 신택스와 시맨틱스를 포함하지만, 수신자가 메시지를 처리하는 프로세스는 정의되어 있지 않다. 결과적으로, 인코더는 SEI 메시지를 생성할 때 HEVC 표준을 따라야 할 필요가 있고, HEVC 표준을 준수하는 디코더는 출력 순서에 적합하게 SEI 메시지를 프로세스할 필요는 없다. HEVC에 SEI 메시지의 신택스와 시맨틱스를 포함시킨 이유 중 하나는 다른 시스템 사양이 보충 정보를 동일하게 해석하여 상호 운용할 수 있도록 하기 위함이다. 시스템 사양은 인코딩 단(encoding end)과 디코딩 단(decoding end) 모두에서 특정 SEI 메시지의 사용을 필요로 할 수 있고, 추가적으로 수신자가 특정 SEI 메시지를 처리하기 위한 프로세스가 지정될 있다는 것이 의도된다.

[0089] HEVC에는, 서로 다른 nal_unit_type 값을 갖는 2가지 타입의 SEI NAL 단위, 즉 접미사(suffix) SEI NAL 단위와 접두사(prefix) SEI NAL 단위가 있다. 접미사 SEI NAL 단위에 포함된 SEI 메시지(들)는 디코딩 순서에서 접미사 SEI NAL 단위에 선행하는 VCL NAL 단위와 관련된다. 접두사 SEI NAL 단위에 포함된 SEI 메시지(들)는 디코딩 순서에서 접두사 SEI NAL 단위에 후속하는 VCL NAL 단위와 관련된다.

[0090] 코딩된 픽처는 픽처의 코딩된 표현이다. 비트스트림은 하나 이상의 코딩된 비디오 시퀀스를 형성하는 코딩된 픽처 및 관련 데이터의 표현을 형성하는, NAL 단위 스트림 또는 바이트 스트림의 형태의, 비트의 시퀀스로서 정의될 수 있다. 같은 파일 또는 통신 프로토콜의 같은 접속에서와 같이 같은 논리 채널에서 제 2 비트스트림은 제 1 비트스트림에 후속할 수 있다. (비디오 코딩의 맥락에서) 기본 스트림은 하나 이상의 비트스트림의 시퀀스로 정의될 수 있다. 제 1 비트스트림의 끝은 비트스트림의 끝(end of bitstream, EOB) NAL 단위로 지정될 수 있고 비트스트림의 마지막 NAL 단위인 특정 NAL 단위로 표시될 수 있다.

- [0091] 코딩된 비디오 시퀀스는 독립적으로 디코딩 가능하고 또 다른 코딩된 비디오 시퀀스 또는 비트스트림의 끝 또는 시퀀스 NAL 단위의 끝이 후속하는 디코딩 순서로 코딩된 픽처의 시퀀스로 정의될 수 있다.
- [0092] HEVC 또는 다른 코덱의 많은 코딩 모드에서, 인터 예측을 위한 참조 픽처는 참조 픽처 리스트에 대한 인덱스로 표시된다. 인덱스는 가변 길이 코딩으로 코딩될 수 있고, 일반적으로 더 작은 인덱스가 대응하는 신팩스 요소에 대해 더 짧은 값을 갖도록 한다. HEVC에서는, 이중 예측(bi-predictive)(B) 슬라이스마다 2개의 참조 픽처 리스트(참조 픽처 리스트 0 및 참조 픽처 리스트 1)가 생성되고, 인터 코딩된(P) 슬라이스마다 1개의 참조 픽처 리스트(참조 픽처 리스트 0)가 형성된다.
- [0093] 참조 픽처 리스트 0 및 참조 픽처 리스트 1과 같은 참조 픽처 리스트는 2단계로 구성될 수 있다. 먼저, 초기 참조 픽처 리스트가 생성된다. 초기 참조 픽처 리스트는, 예를 들어 frame_num, POC, temporal_id, 또는 GOP 구조와 같은 예측 계층에 대한 정보, 또는 이들의 임의의 조합에 근거하여 생성될 수 있다. 다음으로, 초기 참조 픽처 리스트는 슬라이스 헤더에 포함될 수 있고 참조 픽처 리스트 수정 신팩스 구조로도 알려진 RPLR(reference picture list reordering) 신팩스에 의해 재정렬될 수 있다. 초기 참조 픽처 리스트는 참조 픽처 리스트 수정 신팩스 구조를 통해 수정될 수 있고, 초기 참조 픽처 리스트의 픽처는 리스트에 대한 엔트리 인덱스를 통해 식별될 수 있다.
- [0094] HEVC를 포함하는 많은 코딩 표준은 참조 픽처 리스트에 대한 참조 픽처 인덱스를 도출하기 위한 디코딩 프로세스를 가질 수 있고, 이는 복수의 참조 픽처 중 어느 하나가 특정 블록에 대한 인터 예측에 사용되는지를 표시하는 데 사용될 수 있다. 참조 픽처 인덱스는 일부 인터 코딩 모드에서 인코더에 의해 비트스트림으로 코딩될 수 있거나, 또는 일부 다른 인터 코딩 모드에서, 예를 들어 이웃하는 블록을 사용하여 (인코더 및 디코더에 의해) 도출될 수 있다.
- [0095] 단일 예측 단위에 대해 몇몇의 후보 움직임 벡터가 도출될 수 있다. 예를 들어, HEVC의 움직임 벡터 예측은 2 가지 움직임 벡터 예측 방식, 즉 AMVP 및 병합(merge) 모드를 포함한다. AMVP 또는 병합 모드에서는, PU에 대한 움직임 벡터 후보의 리스트가 도출된다. 2가지 종류의 후보, 즉 공간적 후보 및 시간적 후보가 존재하고, 시간적 후보는 또한 TMVP 후보로 지칭될 수 있다.
- [0096] 후보 리스트의 도출은, 예를 들어 다음과 같이 수행될 수 있지만, 후보 리스트의 도출을 위해 다른 가능성의 존재할 수 있음이 이해되어야 한다. 후보 리스트가 최대로 채워지지 않은 경우, 이용 가능하고 후보 리스트에 아직 없는 공간적 후보가 먼저 후보 리스트에 포함된다. 이후, 후보 리스트가 아직 최대로 채워지지 않은 경우, 시간적 후보가 후보 리스트에 포함된다. 후보의 개수가 여전히 최대 허용 개수에 도달하지 않은 경우, (B 슬라이스에 대한) 결합된 이중 예측 후보(combined bi-predictive candidate) 및 제로 움직임 벡터가 추가된다. 후보 리스트가 구성된 이후, 인코더는, 예를 들어 RDO(rate-distortion optimization) 결정에 근거하여 후보로부터 최종 움직임 정보를 결정하고 선택된 후보의 인덱스를 비트스트림에 인코딩한다. 마찬가지로, 디코더는 비트스트림으로부터 선택된 후보의 인덱스를 디코딩하고, 후보 리스트를 구성하고, 디코딩된 인덱스를 사용하여 후보 리스트로부터 움직임 벡터 예측자(predictor)를 선택한다.
- [0097] HEVC에서, AMVP 및 병합 모드는 다음과 같은 특징을 가질 수 있다. AMVP에서, 인코더는 움직임 벡터 차이를 인코딩할뿐만 아니라 단일 예측 또는 이중 예측이 사용되는지 여부와 사용되는 참조 픽처를 표시한다. 병합 모드에서는, 후보 리스트로부터 선택된 후보만이 비트스트림에 인코딩되어 현재 예측 단위가 표시된 예측자와 같은 움직임 정보를 가지고 있음을 표시한다. 따라서, 병합 모드는 동일한 움직임 정보를 공유하는 이웃하는 예측 블록으로 구성된 영역을 생성하고, 이는 각각의 영역에 대해 한 번만 시그널링된다.
- [0098] 이하에서 진보된 움직임 벡터 예측의 동작의 예가 제공되지만, 진보된 움직임 벡터 예측의 다른 유사한 실시는 또한, 예를 들어 다른 후보 위치 세트 및 후보 위치 세트를 갖는 후보 위치를 이용하여 가능하다. 병합 모드와 같은 다른 예측 모드도 유사하게 동작할 수 있다는 것이 이해될 필요가 있다. 2개의 공간적 MVP(motion vector predictor)가 도출될 수 있고 시간적 움직임 벡터 예측자가 도출될 수 있다. 이들은 현재 예측 블록 위쪽에 위치한 3개의 공간적 움직임 벡터 예측자 후보 위치(B_0, B_1, B_2)와 왼쪽에 있는 2개의 위치(A_0, A_1) 중에서 선택될 수 있다. 각각의 후보 위치 세트, (B_0, B_1, B_2) 또는 (A_0, A_1) 의 사전 정의된 순서로 이용 가능한(예를 들어, 같은 슬라이스에 있고, 인터 코딩된 등) 제 1 움직임 벡터 예측자가 선택되어 움직임 벡터 경쟁에서 해당 예측 방향(위쪽 또는 왼쪽)을 표현할 수 있다. 시간적 움직임 벡터 예측자에 대한 참조 인덱스는 인코더에 의해 슬라이스 헤더에 (예를 들어, collocated_ref_index 신팩스 요소로서) 표시될 수 있다. 잠재적인 시간적 후보 위치의 사전 정의된 순서, 예를 들어 순서(C_0, C_1)로 이용 가능한(예를 들어, 인터 코딩된) 제 1 움직임 벡터 예측자

는 시간적 움직임 벡터 예측자를 위한 소스로서 선택될 수 있다. 함께 위치한 픽처의 제 1 이용 가능한 후보 위치로부터 획득된 움직임 벡터는 시간적 움직임 벡터 예측자의 참조 픽처, 함께 위치한 픽처 및 현재 픽처의 픽처 순서 카운트(picture order count)의 차이의 비율에 따라 스케일링될 수 있다. 또한, 중복 검사가 후보 간에 수행되어 동일한 후보를 제거할 수 있고, 이로 인해 제로 움직임 벡터가 후보 리스트에 포함될 수 있다. 움직임 벡터 예측자는, 예를 들어 공간적 움직임 벡터 예측자의 방향(위쪽 또는 왼쪽) 또는 시간적 움직임 벡터 예측자 후보의 선택을 표시함으로써 비트스트림에 표시될 수 있다. 함께 위치한 픽처는 또한 병치된 픽처, 움직임 벡터 예측을 위한 소스 또는 움직임 벡터 예측을 위한 소스 픽처로 지칭될 수 있다.

[0099] 움직임 파라미터 타입 또는 움직임 정보는 다음의 타입 중 하나 이상을 포함할 수 있지만 이에 한정되는 것은 아니다.

- 예측 타입(예를 들어, 인트라 예측, 단일 예측, 이중 예측) 및/또는 복수의 참조 픽처의 표시,

[0100] [0101] - 인터(일명, 시간적) 예측, 인터-레이어 예측, 인터-뷰 예측, 뷰 합성 예측(view synthesis prediction, VSP) 및 인터 컴포넌트 예측과 같은 예측 방향의 표시(참조 픽처마다 및/또는 예측 타입마다 표시될 수 있고, 일부 실시예에서 인터-뷰 및 뷰 합성 예측은 하나의 예측 방향으로서 함께 고려될 수 있음) 및/또는

[0102] - 단기 참조 픽처 및/또는 장기 참조 픽처 및/또는 인터-레이어 참조 픽처와 같은 참조 픽처 타입의 표시(예를 들어, 참조 픽처마다 표시될 수 있음),

[0103] - 참조 픽처 리스트에 대한 참조 인덱스 및/또는 참조 픽처의 임의의 다른 식별자(예를 들어, 참조 픽처마다 표시될 수 있고 그 타입은 예측 방향 및/또는 참조 픽처 타입에 따라 달라질 수 있고, 참조 인덱스가 적용되는 참조 픽처 리스트 등과 같은 기타 관련 정보에 동반될 수 있음),

[0104] - 수평 움직임 벡터 컴포넌트(예를 들어, 예측 블록마다 또는 참조 인덱스마다 등으로 표시될 수 있음),

[0105] - 수직 움직임 벡터 컴포넌트(예를 들어, 예측 블록마다 또는 참조 인덱스마다 등으로 표시될 수 있음),

[0106] - 하나 이상의 움직임 벡터 예측 프로세스에서 수평 움직임 벡터 컴포넌트 및/또는 수직 움직임 벡터 컴포넌트의 스케일링에 사용될 수 있는, 움직임 파라미터를 포함하거나 또는 그와 관련된 픽처와 그의 참조 픽처 사이의 픽처 순서 카운트의 차이 및/또는 상대적 카메라 분리와 같은 하나 이상의 파라미터(상기 하나 이상의 파라미터는, 예를 들어 각각의 참조 픽처 또는 각각의 참조 인덱스 등마다 표시될 수 있음),

[0107] - 움직임 파라미터 및/또는 움직임 정보가 적용되는 블록의 좌표, 예를 들어 루마 샘플 단위에서 블록의 좌상(top-left) 샘플의 좌표,

[0108] - 움직임 파라미터 및/또는 움직임 정보가 적용되는 블록의 범위(예를 들어, 너비 및 높이).

[0109] 일반적으로, 예로서 위에서 제시된 움직임 벡터 예측 메커니즘과 같은 움직임 벡터 예측 메커니즘은 특정의 사전 정의되거나 또는 표시된 움직임 파라미터의 예측 또는 상속(inheritance)을 포함할 수 있다.

[0110] 픽처와 관련된 움직임 필드는 픽처의 모든 코딩된 블록에 대해 생성된 움직임 정보의 세트를 포함하는 것으로 간주될 수 있다. 예를 들어, 움직임 필드는 블록의 좌표로 액세스할 수 있다. 움직임 필드는, 예를 들어 TMVP 또는 현재 (디)코딩된 픽처와는 다른 예측을 위한 소스 또는 참조가 사용되는 임의의 다른 움직임 예측 메커니즘에서 사용될 수 있다.

[0111] 픽처 내의 슬라이스의 위치는 픽처 내에서 래스터 스캔 순서로 블록을 인덱싱하여 표시된다. HEVC에서, slice_segment_header()의 slice_segment_address 선택스 요소는 픽처의 코딩 트리 블록 래스터 스캔에서 슬라이스 세그먼트의 제 1 코딩 트리 블록의 주소를 지정한다.

[0112] 움직임 벡터 앵커 위치는 움직임 벡터가 적용되는 것에 관련된 픽처 영역 내의 위치(예를 들어, 수평 및 수직 좌표)이다. 움직임 벡터 앵커 위치는 움직임 벡터의 초기 지점으로 간주될 수 있다. 종래에는 움직임 벡터 앵커 위치가 예측되는 블록의 위치와 같다고 추정되었지만, 움직임 벡터 앵커 위치를 적응시키기 위한 수평 오프셋 및 수직 오프셋을 비트스트림에 인코딩하거나 및/또는 비트스트림으로부터 디코딩할 수 있는 것이 제안되었다. 앵커 위치에 대한 수평 오프셋 및 수직 오프셋은 슬라이스 헤더, 슬라이스 파라미터 세트, 타일 헤더, 타일 파라미터 세트 등에 제공될 수 있다.

[0113] 움직임 벡터 앵커 위치를 이용하는 예시적인 인코딩 방법은, 입력 픽처를 코딩된 구성 픽처로 인코딩하는 것과,

해당 인코딩의 일부로서, 코딩된 구성 픽처에 대응하는 디코딩된 구성 픽처를 재구성하는 것과, 공간 영역을 코딩된 타일로 인코딩하는 것을 포함하고, 인코딩은, 디코딩된 구성 픽처 내의 공간 영역의 영역 별 앵커 위치를 표시하는 수평 오프셋 및 수직 오프셋을 결정하는 것과, 수평 오프셋 및 수직 오프셋을 인코딩하는 것과, 코딩된 타일의 제 1 수평 좌표 및 제 1 수직 좌표의 위치에서의 예측 단위가 영역 별 앵커 위치에 관련되어 예측되는 것을 결정하되, 제 1 수평 좌표 및 제 1 수직 좌표는 각각 공간 영역 내의 수평 및 수직 좌표인 것과, 예측 단위가 영역 별 앵커 위치에 관련된 예측 단위 앵커 위치에 관련되어 예측되는 것을 표시하는 것과, 제 1 수평 좌표와 수평 오프셋 및 제 1 수직 좌표 및 수직 오프셋의 합과 동일한 예측 단위 앵커 위치를 각각 도출하는 것과, 예측 단위에 대한 움직임 벡터를 결정하는 것과, 예측 단위 앵커 위치에 관련된 움직임 벡터를 적용하여 예측 블록을 획득하는 것을 포함한다.

[0114] 움직임 벡터 앵커 위치가 사용되는 예시적인 디코딩 방법은, 코딩된 타일을 디코딩된 타일로 디코딩하는 것을 포함하고, 디코딩은 수평 오프셋 및 수직 오프셋을 디코딩하는 것과, 코딩된 타일의 제 1 수평 좌표 및 제 1 수직 좌표의 위치에서의 예측 단위가 수평 및 수직 오프셋에 관련된 예측 단위 앵커 위치에 관련되어 예측되는 것이라는 표시를 디코딩하는 것과, 제 1 수평 좌표와 수평 오프셋 및 제 1 수직 좌표와 수직 오프셋의 합과 동일한 예측 단위 앵커 위치를 각각 도출하는 것과, 예측 단위에 대한 움직임 벡터를 결정하는 것과, 예측 단위 앵커 위치에 관련된 움직임 벡터를 적용하여 예측 블록을 획득하는 것을 포함한다.

[0115] 구성 픽처는 전체 입력 픽처의 표현에 대응하는 둘러싼(enclosing) (디)코딩된 픽처의 일부로서 정의될 수 있다. 구성 픽처에 추가하여, 둘러싼 (디)코딩된 픽처는 또 다른 구성 픽처와 같은 다른 데이터를 포함할 수 있다.

[0116] 프레임 패킹은 (입력) 구성 프레임 또는 구성 픽처로 지칭될 수 있는 하나보다 많은 입력 픽처를 출력 픽처로 배열하는 것을 포함하는 것으로 정의될 수 있다. 일반적으로, 프레임 패킹은 특정 타입의 구성 프레임에 제한되지 않거나 또는 구성 프레임은 서로 특정 관계를 가질 필요가 없다. 많은 경우, 프레임 패킹은 입체적인 비디오 클립의 구성 프레임을 단일 픽처 시퀀스로 배열하는 데 사용된다. 배열은 입력 픽처를 출력 픽처 내의 공간적으로 겹치지 않는 영역에 배치하는 것을 포함할 수 있다. 예를 들어, 나란히 배치하면 2개의 입력 픽처가 출력 픽처 내에 서로 수평으로 인접하게 배치된다. 배열은 또한 하나 이상의 입력 픽처를 둘 이상의 구성 프레임 파티션으로 파티셔닝하고 구성 프레임 파티션을 출력 픽처 내의 공간적으로 겹치지 않는 영역에 배치하는 것을 포함할 수 있다. 출력 픽처 또는 프레임 패킹된 출력 픽처의 시퀀스는, 예를 들어 비디오 인코더에 의해, 비트스트림으로 인코딩될 수 있다. 비트스트림은, 예를 들어 비디오 디코더에 의해, 디코딩될 수 있다. 디코더 또는 디코딩 이후 후처리 동작은, 예를 들어 디스플레이를 위해 디코딩된 픽처(들)로부터 디코딩된 구성 프레임을 추출할 수 있다.

[0117] 스트리밍 전송률을 감소시키기 위한 360° 비디오 스트리밍에서의 최근 경향은 뷰포트(viewport)-의존적 스트리밍 또는 뷰포트-적응형 스트리밍이라고 불릴 수 있고, 다음과 같이 간략하게 설명될 수 있다. 주 뷰포트(즉, 현재 뷰 방향)를 커버하는 360도 비디오 콘텐츠의 서브세트는 최상의 화질/해상도로 전송되고, 360도 영상의 나머지는 더 낮은 화질/해상도로 전송된다. 뷰포트-적응형 스트리밍은 타일-기반 인코딩 및 스트리밍 방식을 통해 구현될 수 있다. 타일-기반 인코딩 및 스트리밍의 하나의 방식에서, 인코딩은 산출되는 비트스트림이 움직임 제한 타일 세트를 포함하는 방식으로 수행된다. 같은 소스 콘텐츠의 몇몇 비트스트림이 움직임 제한 타일 세트를 사용하여 인코딩된다. 하나 이상의 MCTS(motion-constrained tile set) 시퀀스가 비트스트림으로부터 추출되고, 추출된 각각의 움직임 제한 타일 세트 시퀀스는 파일에 트랙(예를 들어, HEVC 타일 트랙 또는 서브-픽처 트랙)으로 저장된다. 클라이언트는 현재 뷰포트에 대해 현재 보이지 않는 나머지 뷰포트를 커버하는 품질 또는 해상도에 비해 더 나은 품질 또는 더 높은 해상도의 MCTS 트랙이 수신될 수 있는 방식으로 트랙을 선택적으로 수신한다. 클라이언트는 수신된 트랙의 콘텐츠, 즉 수신된 MCTS를 디코딩될 비트스트림에 병합한다. 뷰잉(viewing) 방향은 시간이 지남에 따라 변경될 수 있으므로, 선택된 트랙과 MCTS는 또한 시간이 지남에 따라 변경될 수 있고 추가적으로 병합된 픽처 내에서 각각의 MCTS의 위치는 하나의 병합된 픽처에서 또 다른 픽처로 변경될 수 있다.

[0118] 슬라이스/타일의 위치를 변경하거나 또는 슬라이스/타일의 서브세트를 추출하는 것은 종종 바람직하다. 하나의 전형적인 경우는 기존 스트림으로부터 타일 세트를 추출하여 또 다른 스트림을 생성할 수 있는 상황이다. 예를 들어, 움직임 제한 타일 세트를 비트스트림으로부터 자신의 비트스트림으로 추출할 수 있다. 또 다른 전형적인 경우는 360° 비디오의 뷰포트-의존적 스트리밍에서 뷰포트를 전환하는 것이다. 슬라이스 또는 타일 콘텐츠를 변경하지 않고 디코딩할 비트스트림에서 슬라이스/타일의 위치를 변경할 수 있을 때, 암호화된 스트림에 대해서

도 위치 변경이 수행될 수 있다.

[0119] 꽂쳐 내의 슬라이스의 위치는 꽂쳐 내의 래스터 스캔 순서로 블록 인덱싱을 통해 표시되기 때문에, 전통적으로, 슬라이스의 위치 변경 또는 타일 또는 슬라이스의 서브세트(꽉쳐의 좌상단으로부터 시작하는 꽂쳐 전체 서브세트의 추출은 슬라이스 헤더의 재작성을 필요로 한다. 마찬가지로, 움직임 벡터 앵커 위치가 사용될 때, tile_ref_x 및 tile_ref_y와 같은 위치를 표시하는 신택스 요소는 슬라이스/타일의 위치를 변경하거나 또는 타일 또는 슬라이스의 서브세트를 자체 비트스트림으로 추출할 때 재작성될 필요가 있다. 슬라이스 헤더의 재작성은 스트림의 전송률에 영향을 미치고 및/또는 특별한 재작성 명령어를 필요하기 때문에 슬라이스의 위치 변경을 목적으로 슬라이스 헤더를 재작성하는 것은 비효율적이다.

[0120] 여기에 설명된 예시적인 실시예는 타일을 타일 그리드 내의 타일의 래스터 스캔 순서로 표시하는 타일 식별자를 비트스트림과 관련된 파라미터 세트에 인코딩함으로써 슬라이스 헤더를 재작성하지 않고도 비트스트림에서 타일의 위치 변경을 가능하도록 한다.

[0121] 비디오 비트스트림의 파일 포맷에 관계없이, 예시적인 실시예의 장치는, 예를 들어 비디오 인코더, 컴퓨터 워크 스테이션, 서버 등을 포함하는 다양한 컴퓨팅 장치 중 임의의 것에 의해, 또는 모바일 단말기, 예를 들어 스마트 폰, 태블릿 컴퓨터, 비디오 게임 플레이어 등과 같은 다양한 모바일 컴퓨팅 장치 중 임의의 것에 의해 제공될 수 있다.

[0122] 장치를 구현하는 컴퓨팅 장치에 관계없이, 예시적인 실시예의 장치(10)는 도 1에 도시된 것과 같이 프로세싱 회로(12), 메모리(14), 통신 인터페이스(16) 및 선택적으로 사용자 인터페이스(18)를 포함하거나, 관련되거나 또는 이와 달리 통신한다.

[0123] 프로세싱 회로(12)는 장치(10)의 컴포넌트 사이에서 정보를 전달하기 위해 버스를 통해 메모리 장치(14)와 통신 할 수 있다. 메모리 장치는 비일시적일 수 있고, 예를 들어 하나 이상의 휘발성 및/또는 비휘발성 메모리를 포함할 수 있다. 다시 말해, 예를 들어, 메모리 장치는 머신(예를 들어, 프로세싱 회로와 같은 컴퓨팅 장치)에 의해 검색될 수 있는 데이터(예를 들어, 비트)를 저장하도록 구성된 게이트를 포함하는 전자적 저장 장치(예를 들어, 컴퓨터 판독 가능 저장 매체)일 수 있다. 메모리 장치는 장치가 본 개시의 예시적인 실시예에 따라 다양한 기능을 수행할 수 있도록 하는 정보, 데이터, 콘텐츠, 애플리케이션, 명령어 등을 저장하도록 구성될 수 있다. 예를 들어, 메모리 장치는 프로세싱 회로에 의한 프로세싱을 위해 입력 데이터를 버퍼링하도록 구성될 수 있다. 추가적으로 또는 이와 달리, 메모리 장치는 프로세싱 회로에 의한 실행을 위해 명령어를 저장하도록 구성될 수 있다.

[0124] 장치(10)는, 일부 실시예에서, 전술한 것과 같이 다양한 컴퓨팅 장치로 구현될 수 있다. 그러나, 일부 실시예에서, 장치는 칩 또는 칩셋으로 구현될 수 있다. 다시 말해, 장치는 구조적 어셈블리(예를 들어, 베이스보드) 위에 재료, 컴포넌트 및/또는 배선을 포함하는 하나 이상의 물리적 패키지(예를 들어, 칩)를 포함할 수 있다. 구조적 어셈블리는 그 위에 포함된 컴포넌트 회로에 대한 물리적 강도, 크기의 유지 및/또는 전기적 상호 작용의 제한을 제공할 수 있다. 따라서, 일부 경우에, 장치는 단일 칩 상에서 또는 단일 "시스템 온 칩"으로서 본 개시의 실시예를 구현하도록 구성될 수 있다. 이와 같이, 일부 경우에, 칩 또는 칩셋은 여기에 설명된 기능을 제공하기 위한 하나 이상의 동작을 수행하는 수단을 구성할 수 있다.

[0125] 프로세싱 회로(12)는 다수의 다른 방식으로 구현될 수 있다. 예를 들어, 프로세싱 회로는 코프로세서, 마이크로프로세서, 컨트롤러, DSP(digital signal processor), DSP를 구비하거나 또는 구비하지 않는 프로세싱 요소, 또는, 예를 들어 ASIC(application specific integrated circuit), FPGA(field programmable gate array), MCU(microcontroller unit), 하드웨어 가속기, 특수 목적 컴퓨터 칩 등과 같은 집적 회로를 포함하는 다양한 다른 회로와 같은 다양한 하드웨어 프로세싱 수단 중 하나 이상으로 구현될 수 있다. 이와 같이, 일부 실시예에서, 프로세싱 회로는 독립적으로 수행하도록 구성된 하나 이상의 프로세싱 코어를 포함할 수 있다. 다중 코어 프로세싱 회로는 단일 물리적 패키지 내에서 멀티프로세싱을 가능하게 할 수 있다. 추가적으로 또는 이와 달리, 프로세싱 회로는 명령어, 파이프라인 및/또는 멀티스레딩의 독립적인 실행을 가능하도록 하기 위해 버스를 통해 직렬로 구성된 하나 이상의 프로세서를 포함할 수 있다.

[0126] 예시적인 실시예에서, 프로세싱 회로(12)는 메모리 장치(14)에 저장된 명령어를 실행하도록 구성되거나 그렇지 않으면 프로세싱 회로에 액세스 가능할 수 있다. 이와 달리 또는 추가적으로, 프로세싱 회로는 하드 코딩된 기능을 실행하도록 구성될 수 있다. 이와 같이, 하드웨어 또는 소프트웨어 방법에 의해 구성되거나, 또는 이들의 조합에 의해 구성되는 것과 관계없이, 프로세싱 회로는 그에 따라 구성되는 동안 본 개시의 실시예에 따른 동작

을 수행할 수 있는 엔티티(예를 들어, 회로에 물리적으로 구현됨)를 표현할 수 있다. 따라서, 예를 들어, 프로세싱 회로가 ASIC, FPGA 등으로 구현될 때, 프로세싱 회로는 여기에 설명된 동작을 수행하기 위해 특별히 구성된 하드웨어일 수 있다. 이와 달리, 또 다른 예로서, 프로세싱 회로가 명령어의 실행자로서 구현될 때, 명령어는 명령어가 실행될 때 여기에 설명된 알고리즘 및/또는 동작을 수행하도록 프로세서를 구체적으로 구성할 수 있다. 그러나, 일부 경우에, 프로세싱 회로는 알고리즘 및/또는 여기에 설명된 동작을 수행하기 위한 명령어에 의한 프로세싱 회로의 추가 구성에 의해 본 발명의 실시예를 사용하도록 구성된 특정 장치(예를 들어, 이미지 또는 비디오 프로세싱 시스템)의 프로세서일 수 있다. 프로세싱 회로는 특히 클록, ALU(arithmetic logic unit) 및 프로세싱 회로의 동작을 지원하도록 구성된 논리 게이트를 포함할 수 있다.

[0127] 통신 인터페이스(16)는 비디오 비트스트림을 포함하는 데이터를 수신 및/또는 전송하도록 구성된 하드웨어 또는 하드웨어와 소프트웨어의 조합으로 구현된 장치 또는 회로와 같은 임의의 수단일 수 있다. 이와 관련하여, 통신 인터페이스는, 예를 들어 안테나(또는 다중 안테나) 및 무선 통신 네트워크와의 통신을 가능하도록 하는 지원 하드웨어 및/또는 소프트웨어를 포함할 수 있다. 추가적으로 또는 이와 달리, 통신 인터페이스는 안테나(들)를 통해 신호를 전송하거나 안테나(들)를 통해 수신된 신호의 수신을 처리하기 위해 안테나(들)와 상호 작용하기 위한 회로를 포함할 수 있다. 일부 환경에서는, 통신 인터페이스가 이와 다르게 또는 통신 인터페이스 또한 유선 통신을 지원할 수 있다. 이와 같이, 예를 들어, 통신 인터페이스는 케이블, DSL(digital subscriber line), USB(universal serial bus) 또는 다른 메커니즘을 통해 통신을 지원하기 위한 통신 모뎀 및/또는 다른 하드웨어/소프트웨어를 포함할 수 있다.

[0128] 일부 실시예에서, 장치(10)가 비디오 비트스트림을 인코딩하도록 구성된 경우, 장치(10)는 선택적으로 사용자 인터페이스(18)를 포함할 수 있고, 이는, 이번에는, 프로세싱 회로(12)와 통신하여, 인코딩된 비디오 비트스트림을 출력하는 것과 같이 사용자에게 출력을 제공하고, 일부 실시예에서, 사용자 입력의 표시를 수신한다. 이와 같이, 사용자 인터페이스는 디스플레이를 포함할 수 있고, 일부 실시예에서는, 또한 키보드, 마우스, 조이스틱, 터치 스크린, 터치 영역, 소프트 키, 마이크, 스피커 또는 다른 입력/출력 메커니즘을 포함할 수 있다. 이와 달리 또는 추가적으로, 프로세싱 회로는 디스플레이 및, 일부 실시예에서는, 스피커, 링어(ringer), 마이크 등과 같은 하나 이상의 사용자 인터페이스 요소의 적어도 일부 기능을 제어하도록 구성된 사용자 인터페이스 회로를 포함할 수 있다. 프로세싱 회로 및/또는 프로세싱 회로를 포함하는 사용자 인터페이스 회로는 프로세싱 회로에 액세스할 수 있는 메모리(예를 들어, 메모리 장치(14) 등)에 저장된 컴퓨터 프로그램 명령어(예를 들어, 소프트웨어 및/또는 펌웨어)를 통해 하나 이상의 사용자 인터페이스 요소의 하나 이상의 기능을 제어하도록 구성될 수 있다.

[0129] 이제 도 2를 참조하면, 예시적인 실시예에 따라, 픽처를 2개 이상의 타일로 파티션되어 있는 코딩된 픽처로 인코딩하기 위해, 도 1의 장치(10)와 같은 장치에 의해 수행되는 동작이 도시된다. 블록(20)에 도시된 것과 같이, 장치는 픽처를 수신하기 위한 프로세싱 회로(12), 통신 인터페이스(16) 등과 같은 수단을 포함한다. 픽처는 하나 이상의 슬라이스를 포함하는 코딩된 픽처로 인코딩된다. 픽처는 하나 이상의 타일로 파티션된다. 코딩된 픽처는 비트스트림의 일부일 수 있다. 비트스트림은 HEVC, H.266/VVC 또는 전술된 임의의 파일 포맷에 따라 포맷된 비트스트림일 수 있다.

[0130] 블록(22)에 도시된 것과 같이, 장치(10)는 다음의 목적의 임의의 세트를 위한 하나 이상의 타일 식별자를 슬라이스의 슬라이스 헤더로 인코딩하기 위한 프로세싱 회로(12)와 같은 수단을 포함한다.

[0131] - 슬라이스 위치 정보. 타일 식별자는 슬라이스의 제 1 블록과 같은 슬라이스의 사전 정의되거나 또는 표시된 블록을 포함하는 타일을 표시한다. 블록은, 예를 들어 코딩 트리 블록 또는 코딩 트리 단위 등일 수 있다.

[0132] - 움직임 백터의 앵커 위치를 결정하기 위한 수평 및 수직 오프셋 도출. 타일 식별자는 수평 및 수직 오프셋을 도출하는 데 사용되는 타일 위치를 표시한다.

[0133] - TMVP 또는 유사 동작에서 표시된 픽처로부터 움직임 백터 후보를 획득하기 위한 앵커 위치를 결정하기 위한 수평 및 수직 오프셋 도출.

[0134] 슬라이스 위치 정보에 대한 타일 식별자를 인코딩하는 실시예에서, 슬라이스 세그먼트 헤더의 선택스는 다음과 같다:

	Descriptor
slice_segment_header()	
slice_pic_parameter_set_id	ue(v)
slice_tile_id	u(v)
...	
}	

[0135]

slice_pic_parameter_set_id는 사용 중인 픽처 파라미터 세트를 지정한다. 예를 들어, 픽처 파라미터 세트 선택스가 PPS의 식별자를 제공하는 **pps_pic_parameter_set_id**를 포함할 때, **slice_pic_parameter_set_id**는 사용 중인 PPS에 대한 **pps_pic_parameter_set_id**의 값을 지정한다. **slice_tile_id**는 슬라이스 세그먼트의 제 1 코딩 트리 블록이 **slice_tile_id** 값으로 식별되는 타일의 좌상 코딩 트리 블록임을 지정한다. **slice_tile_id** 선택스 요소의 길이는 사용 중인 PPS에 의해 결정된다. 이와 달리, 또 다른 실시예에서, **slice_tile_id**는 ue(v) 타입이고 따라서 그 길이는 사용 중인 PPS에 의해 결정되기 보다는 암시적으로 표시된다.

[0137]

실시예에서, 다양한 실시예의 **slice_tile_id**와 같은 고정 길이 타일 식별자 코드워드의 길이는, 예를 들어 사용 중인 PPS로 표시되는 것과 같은 픽처의 타일 개수에 의해 추정된다. 일부 실시예에서, 픽처의 추정된 타일의 개수는 **NumTilesInPic**과 동일하게 설정되고 타일 식별자 코드워드의 길이는 $\text{Ceil}(\text{Log2}(\text{NumTilesInPic}))$ 비트로 결정될 것이다. 예를 들어, 사용 중인 PPS는 타일 행과 열의 개수를 표시할 수 있고 **NumTilesInPic**은 타일 행의 개수에 타일 열의 개수를 곱하여 획득될 수 있다.

[0138]

실시예에서, 다양한 실시예의 **slice_tile_id**와 같은 고정 길이 타일 식별자 코드워드의 길이는, 픽처 파라미터 세트 또는 시퀀스 파라미터 세트와 같은 고-레벨 선택스 구조로 (인코더에 의해) 표시되거나 또는 (디코더에 의해) 디코딩된다. 이 실시예는 슬라이스 헤더 등의 타일 식별자의 길이가 타일 그리드 또는 픽처 크기와 무관하게 되어 슬라이스 헤더의 수정 없이 또는 더 적은 슬라이스 헤더의 수정으로 타일의 서브세트를 자신의 비트스트림으로 추출할 수 있기 때문에, 일부 사용 예에서 효과적일 수 있다. 예를 들어, 픽처 파라미터 세트의 다음의 선택스가 사용될 수 있다.

	Descriptor
pic_parameter_set_rbsp()	
...	
tiles_enabled_flag	u(1)
if(tiles_enabled_flag) {	
...	
tile_id_len_minus1	ue(v)
...	
}	
...	
}	

[0139]

위의 선택스 요소의 시맨틱스는 다음과 같이 지정될 수 있다. **tiles_enabled_flag**는 PPS를 참조하는 각각의 픽처에 2개 이상의 타일이 있는지 여부를 표시한다. **tiles_enabled_flag**가 1인 경우, PPS를 참조하는 각각의 픽처에 2개 이상의 타일이 있다. **tiles_enabled_flag**가 0인 경우, PPS를 참조하는 각각의 픽처에 1개의 타일만이 있다. **tile_id_len_minus1 + 1**은 다양한 실시예에서 **slice_tile_id**와 같은 타일 식별자 선택스 요소(들)의 길이를 비트로 지정한다.

[0141]

또 다른 예시적인 실시예에서, 픽처 파라미터 세트의 다음의 선택스가 사용될 수 있다.

	Descriptor
pic_parameter_set_rbsp()	
...	
tiles_enabled_flag	u(1)
if(tiles_enabled_flag) {	
...	
num_tile_columns_minus1	ue(v)
num_tile_rows_minus1	ue(v)
rel_tile_id_len	ue(v)
...	
}	
...	
}	

[0142]

[0143]

일부 실시예에서, 위의 신택스 요소의 시멘틱스는 다음과 같이 지정될 수 있다. `tiles_enabled_flag`는 PPS를 참조하는 각각의 픽처에 2개 이상의 타일이 있는지 여부를 지정한다. `tiles_enabled_flag`가 1인 경우, PPS를 참조하는 각각의 픽처에 2개 이상의 타일이 있다. `tiles_enabled_flag`가 0인 경우, PPS를 참조하는 각각의 픽처에 1개의 타일만이 있다. `num_tile_columns_minus1`은 픽처를 파티셔닝한 타일 열의 개수에서 1을 뺀 것을 지정한다. `num_tile_rows_minus1`은 픽처를 파티셔닝한 타일 행의 개수에서 1을 뺀 것을 지정한다. 일부 실시예에서, `NumTilesInPic`은 $(\text{num_tile_columns_minus1} + 1) * (\text{num_tile_rows_minus1} + 1)$ 과 동일하게 설정된다. 일부 실시예에서, 도출된 변수 `MinBitsTileId`가 도입되고 `Ceil(\text{Log2}(\text{NumTilesInPic}))` 비트 신택스 요소와 동일하게 설정된다. `rel_tile_id_len`은 타일 식별자 신택스 요소(들)에 대해 사용된 `MinBitsTileId` 외에 추가 비트의 개수를 지정한다. 다양한 실시예의 `slice_tile_id`와 같은 타일 식별자 신택스 요소(들)의 길이의 비트는 `MinBitsTileId + rel_tile_id_len`과 동일하다.

[0144]

실시예에서, 인코더 및/또는 디코더는 타일 식별자가 도출되는 2개의 신택스 요소를 인코딩 및/또는 디코딩한다. 신택스 요소는 승수 인자(multiplier factor) 및 가산 인자(additive factor)로 지정될 수 있다. 실시예에서, 타일 식별자는 승수 인자를 타일 그리드 폭과 곱하고 가산 인자를 더하여 도출된다. 또 다른 실시예에서, 타일 식별자는 승수 인자를 타일 그리드 높이와 곱하고 가산 인자를 더하여 도출된다. 이 방법은 특히 픽처의 제 1 슬라이스 및 타일 행 또는 타일 열의 시작과 정렬되는 슬라이스에 대해 신택스 요소의 가변 길이 코딩이 사용될 때 효과적일 수 있다. 이 실시예에서는 다음의 신택스 등이 사용될 수 있다. 실시예는 다음의 신택스 구조의 신택스 요소의 임의의 순서로 유사하게 실시될 수 있다는 것에 유의한다.

<code>slice_segment_header()</code>	Descriptor
<code>slice_tile_mul</code>	<code>ue(v)</code>
<code>slice_tile_add</code>	<code>ue(v)</code>
<code>slice_pic_parameter_set_id</code>	<code>ue(v)</code>
...	
}	

[0145]

`slice_tile_mul` 및 `slice_tile_add`는 슬라이스 세그먼트의 제 1 코딩 트리 블록이 `slice_tile_mul * (num_tile_columns_minus1 + 1) + slice_tile_add`에 의해 식별되는 타일의 좌상 코딩 트리 블록임을 지정한다. `slice_pic_parameter_set_id`는 사용 중인 픽처 파라미터 세트를 지정한다. 예를 들어, 픽처 파라미터 세트 신택스가 PPS의 식별자를 제공하는 `pps_pic_parameter_set_id`를 포함할 때, `slice_pic_parameter_set_id`는 사용 중인 PPS에 대한 `pps_pic_parameter_set_id`의 값을 지정한다. 일부 실시예에서, 고정 길이 코딩, 즉 `u(v)`는 `slice_tile_mul` 및 `slice_tile_add` 중 하나 또는 둘 모두에 사용될 수 있다.

[0147]

실시예에서, 타일 또는 슬라이스 위치를 표시하는 그리드 또는 파티셔닝은 타일 그리드와 별도로 표시되거나 또는 디코딩된다. 예를 들어, 다음의 신택스가 사용될 수 있다.

<code>pic_parameter_set_rbsp()</code>	Descriptor
...	
<code>num_slice_tile_id_columns_minus1</code>	<code>ue(v)</code>
<code>num_slice_tile_id_rows_minus1</code>	<code>ue(v)</code>
<code>for(i = 0; i < num_slice_tile_id_columns_minus1; i++)</code>	
<code>id_column_width_minus1[i]</code>	<code>ue(v)</code>
<code>for(i = 0; i < num_slice_tile_id_rows_minus1; i++)</code>	
<code>id_row_height_minus1[i]</code>	<code>ue(v)</code>
...	
}	

[0148]

`num_slice_tile_id_columns_minus1`은 픽처를 파티셔닝하는 열의 개수에서 1을 뺀 것을 지정한다. `num_slice_tile_id_rows_minus1`은 픽처를 파티셔닝하는 행의 개수에서 1을 뺀 것을 지정한다. `id_column_width_minus1[i]`는 코딩 트리 블록의 단위로 `i`번째 열의 너비에서 1을 뺀 것을 지정한다. `id_row_height_minus1[i]`는 코딩 트리 블록의 단위로 `i`번째 행의 높이에서 1을 뺀 것을 지정한다. 타일 식별자 값(예를 들어, `slice_tile_id`)은 이를 신택스 요소로 지정된 그리드를 따르는 공간적 위치와 관련된다. 다양한 실시예에서 "타일 그리드"라는 용어가 언급될 때, 이를 실시예는 이를 신택스 요소 등에 의해 지정된 그리드로 유사하게 실시될 수 있고, 이하에서는 이를 식별자 그리드로 지정한다. `slice_tile_id`의 시멘틱스는 슬라이스 세그먼트의 제 1 코딩 트리 블록이 `slice_tile_id` 값에 의해 식별되는 (식별자 그리드의) 그리드 셀의 좌상 코

딩 트리 블록인 것으로 지정될 수 있다.

[0150] 실시예에서, 타일 그리드의 정점(vertice)은 정점 식별자 그리드와 병치하지만 더 희소할 수 있다. 타일 그리드의 정점은 식별자 그리드에 관련되어 표시될 수 있다. 즉, 식별자 그리드의 어느 정점이 또한 타일 그리드의 정점인지 표시될 수 있다. 예를 들어, 다음의 신택스가 사용될 수 있다.

pic_parameter_set_rbsp() {	Descriptor
...	
if(tiles_enabled_flag) {	
...	
for(i = 0; i < num_slice_tile_id_columns_minus1; i++)	
tile_column_flag[i]	u(1)
for(i = 0; i < num_slice_tile_id_rows_minus1; i++)	
tile_row_flag[i]	u(1)
...	
}	
}	

[0151]

tile_column_flag[i]는 식별자 그리드의 i번째(오른쪽) 열 경계가 타일 열 경계인지 여부를 지정한다. tile_column_flag[i]가 1인 경우, 식별자 그리드의 i번째(오른쪽) 열 경계는 타일 열 경계이다. tile_column_flag[i]가 0인 경우, 식별자 그리드의 i번째 열 경계는 타일 열 경계가 아니다. tile_row_flag[i]는 식별자 그리드의 i번째 행 경계(i번째 행의 아래)가 타일 행 경계인지 여부를 지정한다. tile_column_flag[i]가 1인 경우, 식별자 그리드의 i번째 행 경계는 타일 행 경계이다.

[0153] 식별자 그리드를 사용함으로써 일부 사용 예에서 움직임 제한 슬라이스 및 MCTS의 혼합 사용이 가능하다.

[0154] 일부 실시예에서 타일 식별자는 타일 그리드 내의 타일의 래스터 스캔 순서로 타일 위치에 할당된다. 일부 실시예에서, 타일 식별자는 tile_id 신택스 요소 등의 형태를 취한다. tile_id 값은 사전 정의된 방식, 예를 들어 타일 그리드 내의 타일의 래스터 스캔 순서의 0, 1, 2, 3, ...으로 할당될 수 있다.

[0155] 실시예에서, 타일 식별자 값(즉, tile_id 값 등)은, 예를 들어 픽쳐 파라미터 세트에서, 예를 들어 타일 그리드 내의 타일의 래스터 스캔 순서로 식별자 값의 리스트로서 비트스트림에 또는 비트스트림을 따라 타일 위치에 명시적으로 할당된다. 표시된 식별자가 없는 경우, 사전 정의된 인덱싱 방식이 활용될 수 있다.

[0156] 실시예에서, 타일 식별자 값이 사전 정의된 방식으로 할당되거나 또는 비트스트림에 또는 비트스트림을 따라 명시적으로 할당되는 경우, 이는 비트스트림에 또는 비트스트림을 따라, 예를 들어 시퀀스 파라미터 세트 또는 픽쳐 파라미터 세트에 표시된다.

[0157] 블록(24)에 도시된 것과 같이, 장치(10)는 타일 위치에 타일 식별자 값을 타일 위치에 할당하는 것을 선택적으로 파라미터 세트 파일에 인코딩하기 위한 프로세싱 회로(12)와 같은 수단을 더 포함할 수 있다. 예시적인 파라미터 세트 파일은 픽쳐 파라미터 세트 파일, 시퀀스 파라미터 세트 파일, 비디오 파라미터 세트 파일 등을 포함한다.

[0158] 예시적인 실시예에서, 픽쳐 파라미터 세트의 다음의 신택스가 사용될 수 있다.

pic_parameter_set_rbsp() {	Descriptor
...	
tiles_enabled_flag	u(1)
if(tiles_enabled_flag) {	
...	
explicit_tile_id_order_flag	u(1)
if(explicit_tile_id_order_flag)	
for(i = 0; i < NumTilesInPic; i++)	
tile_id[i]	u(v)
...	
}	
...	
}	

[0159]

[0160] tiles_enabled_flag는 전술된 것과 같이 지정될 수 있고, 위의 다른 신택스 요소의 시맨틱스는 다음과 같이 지

정될 수 있다. `explicit_tile_id_order_flag`가 0인 경우, 이는 타일 식별자 값이 사전 정의된 방식, 예를 들어 타일 그리드 내의 타일의 래스터 스캔 순서의 0, 1, 2, 3, ...으로 타일 위치에 할당되는 것을 표시한다. `explicit_tile_id_order_flag`는 이 PPS가 사용 중인 픽처의 타일에 대한 타일 식별자 순서가 `tile_id[i]` 선택 요소에 의해 지정되는지 여부를 지정한다. `explicit_tile_id_order_flag`가 1인 경우, 이 PPS가 사용 중인 픽처의 타일에 대한 타일 식별자 순서가 `tile_id[i]` 선택 요소에 의해 지정된다. `tile_id[i]`는 타일 그리드 내의 타일의 래스터 스캔 순서로 i번째 타일의 타일 식별자를 지정한다.

[0161] 장치는 또한 타일에 관련된 블록 인덱스를 구성할 수 있다. 일부 실시예에서, 블록 인덱스는 타일 내의 블록에 대한 블록 위치를 표시하는 선택 요소를 포함한다. 예를 들어, 블록 인덱스는 단일 선택 요소, 예를 들어 `block_index_wrt_tile_id` 선택 요소 또는 2개의 선택 요소, 예를 들어 `first_slice_segment_in_tile_flag` 및 `block_index_wrt_tile_id`로 표시될 수 있다. `first_slice_segment_in_tile_flag`는 슬라이스 주소가 타일의 좌상 블록과 일치하는지 여부를 표시한다. `block_index_wrt_tile_id`는 타일 내의 블록 위치를 표시한다. 일부 실시예에서, 슬라이스 주소에 대한 블록 인덱스의 존재는 픽처 파라미터 세트에서 제어된다.

[0162] 예시적인 실시예에서, 다음의 픽처 파라미터 세트 및 슬라이스 세그먼트 헤더 선택스 구조가 사용된다.

pic_parameter_set_rbsp()	Descriptor
...	
tiles_enabled_flag	u(1)
if(<code>tiles_enabled_flag</code>) {	
...	
tile_aligned_slices_flag	u(1)
...	
}	
...	
}	

slice_segment_header()	Descriptor
slice_pic_parameter_set_id	ue(v)
slice_tile_id	u(v)
if(<code>dependent_slice_segments_enabled_flag</code>)	
dependent_slice_segment_flag	u(1)
if(! <code>dependent_slice_segment_flag</code>) {	
if(<code>tile_aligned_slices_flag</code> == 0)	
block_index_wrt_tile_id	u(v)
...	
}	
...	
}	

[0163] [0164] [0165] `tiles_enabled_flag`, `slice_pic_parameter_set_id` 및 `slice_tile_id`의 시맨틱스는 전술되었다. `tile_aligned_slices_flag`는 모든 슬라이스(또는 모든 독립 슬라이스 세그먼트)가 타일의 좌상단으로부터 시작하는지, 즉 각각의 슬라이스의 제 1 CTU가 타일의 좌상 CTU인지 여부를 지정한다. `tile_aligned_slices_flag`가 0인 경우, 슬라이스(또는 독립 슬라이스 세그먼트)는 타일의 좌상단으로부터 시작하거나 또는 시작하지 않을 수 있다. 즉, 슬라이스의 제 1 CTU는 좌상 CTU일 수 있거나 또는 아닐 수 있다. `tile_aligned_slices_flag`가 1인 경우, 모든 슬라이스(또는 모든 독립 슬라이스 세그먼트)는 타일의 좌상단으로부터 시작한다. `dependent_slice_segment_flag`는 존재하지 않는 각각의 슬라이스 세그먼트 헤더 선택스 요소의 값이 슬라이스 헤더의 대응하는 슬라이스 세그먼트 헤더 선택스 요소의 값과 동일한 것으로 추정되는지 여부를 지정한다. `dependent_slice_segment_flag`가 1인 경우, 존재하지 않는 각각의 슬라이스 세그먼트 헤더 선택스 요소의 값은 슬라이스 헤더의 대응하는 슬라이스 세그먼트 헤더 선택스 요소의 값과 동일한 것으로 추정된다. `block_index_wrt_tile_id`는 `slice_tile_id`에 의해 식별되는 타일의 코딩 트리 블록 래스터 스캔에서, 슬라이스 세그먼트의 제 1 코딩 트리 블록의 주소를 지정한다. `slice_segment_address` 선택스 요소의 길이는 $\text{Ceil}(\text{Log2}(\text{TileSizeInCtbsY}))$ 비트이고, `TileSizeInCtbsY`는 타일의 루마 코딩 트리 블록의 개수이다. 또 다른 실시예에서, `block_index_wrt_tile_id`는 `u(v)` 대신 `ue(v)` 타입을 갖는다.

[0166] 실시예에서, 타일에 관련된 블록 인덱스는 2개의 선택스 요소로 표시되고, 하나는 타일 내의 블록의 수평 위치를 표시하고, 다른 하나는 타일 내의 블록의 수직 위치를 표시한다. 2개의 선택스 요소의 타입은, 예를 들어

ue(v) 또는 u(v)일 수 있다. u(v)가 사용될 때, 신택스 요소의 길이는 블록 행의 개수와 블록 열의 개수에 의해 결정될 수 있다.

[0167] 일부 실시예에서, 장치(10)는 하나 이상의 타일의 하나 이상의 코딩 단위의 하나 이상의 예측 단위와 관련된 하나 이상의 움직임 벡터를 구성하기 위한 프로세싱 회로(12)와 같은 수단을 더 포함한다. 하나 이상의 움직임 벡터는 하나 이상의 앵커 위치 식별자를 포함하고, 각각은 하나 이상의 타일 식별자에 의해 표시되는 하나 이상의 타일에 관련된 위치를 정의한다. 움직임 벡터는, 각각 수평 및 수직 오프셋으로 지정되는, tile_ref_x 및 tile_ref_y와 같은, 사전 정의된 위치를 표시하는 신택스 요소의 형태로 앵커 식별자를 포함한다. 전술된 것과 같이, 수평 및 수직 오프셋은 예측 블록을 획득하기 위해 움직임 벡터를 적용하기 위한 앵커 위치를 결정하는데 사용된다. 도 3에 도시된 예에서, 앵커 식별자는 360도 비디오에 사용되고, 기본 품질의 공유 코딩된 픽처와 향상된 품질의 타일은 같은 코딩된 픽처 내에서 코딩되고, 같은 코딩된 피처의 공유 코딩된 픽처와 향상된 품질의 타일은 같은 360도 픽처로부터 유래한다. 도 3에서, 향상된 품질의 타일에 대한 앵커 식별자는 공유 구성 픽처 내의 위치를 표시한다.

[0168] 일부 실시예에서, 움직임 벡터에 대한 앵커 위치(들)는 픽처에 관련된 x 및 y 샘플 좌표보다는 tile_id 값(들) 등과 같은 타일 식별자(들)를 통해 표시된다. 움직임 벡터 앵커 위치는 타일 식별자(예를 들어, tile_id 값)에 의해 식별된 타일의 좌상단과 같이 타일 식별자(예를 들어, tile_id 값)에 의해 식별된 타일에 관련된 사전 정의된 위치인 것으로 도출된다.

[0169] 코딩된 타일을 디코딩된 타일로 디코딩하기 위한 실시예는 다음을 포함한다.

[0170] - 움직임 벡터 앵커 위치를 도출하기 위한 타일 식별자를 디코딩하는 것,

[0171] - 타일 식별자로부터 타일 위치를 도출하는 것,

[0172] - 타일 위치로부터 수평 오프셋 및 수직 오프셋을 도출하는 것,

[0173] - 코딩된 타일의 제 1 수평 좌표 및 제 1 수직 좌표의 위치에서의 예측 단위가 수평 및 수직 오프셋에 관련된 예측 단위 앵커 위치에 관련되어 예측된다는 표시를 디코딩하는 것,

[0174] - 제 1 수평 좌표와 수평 오프셋, 및 제 1 수직 좌표와 수직 오프셋의 각각의 합과 동일한 예측 단위 앵커 위치를 도출하는 것,

[0175] - 예측 단위에 대한 움직임 벡터를 결정하는 것, 그리고

[0176] - 예측 단위 앵커 위치에 관련된 움직임 벡터를 적용하여 예측 블록을 획득하는 것.

[0177] 움직임 벡터 앵커 위치를 이용하는 인코딩을 위한 실시예는 다음을 포함한다.

[0178] - 입력 픽처를 코딩된 구성 픽처로 인코딩하는 것,

[0179] - 해당 인코딩의 일부로서, 코딩된 구성 픽처에 대응하는 디코딩된 구성 픽처를 재구성하는 것,

[0180] - 공간적 영역을 코딩된 타일로 인코딩하는 것으로, 인코딩은 다음을 포함함

[0181] ○ 코딩된 구성 픽처 내의 타일 위치를 결정하되, 타일 위치는 디코딩된 구성 픽처 내의 공간적 영역의 영역별 앵커 위치를 표시하는 수평 오프셋 및 수직 오프셋을 표시하는 것,

[0182] ○ 타일 위치를 표시하는 타일 식별자를 인코딩하는 것,

[0183] ○ 코딩된 타일의 제 1 수평 좌표 및 제 1 수직 좌표의 위치에서의 예측 단위가 영역별 앵커 위치에 관련되어 예측되는 것을 결정하되, 제 1 수평 좌표 및 제 1 수직 좌표는 각각 공간적 영역 내의 수평 및 수직 좌표인 것,

[0184] ○ 예측 단위가 영역별 앵커 위치에 관련된 예측 단위 앵커 위치에 관련되어 예측되는 것을 표시하는 것,

[0185] ○ 제 1 수평 좌표와 수평 오프셋, 및 제 1 수직 좌표와 수직 오프셋의 각각의 합과 동일한 예측 단위 앵커 위치를 도출하는 것,

[0186] ○ 예측 단위에 대한 움직임 벡터를 결정하는 것, 그리고

[0187] ○ 예측 단위 앵커 위치에 관련된 움직임 벡터를 적용하여 예측 블록을 획득하는 것.

[0188]

실시예에서, 움직임 벡터 앵커 위치를 도출하기 위한 타일 식별자는 슬라이스 위치 정보에 대한 타일 식별자와 같을 것으로 추정된다. 전술된 실시예는 다음과 같이 적용된다. 즉, 코딩된 타일(예측 단위를 포함)의 타일 식별자가 tileIdA로 지정된 특정 값과 동일할 때, 움직임 벡터 앵커 위치를 도출하기 위한 타일 식별자는 tileIdA와 동일할 것으로 추정되고, 움직임 벡터에 대한 참조 픽처에서 tileIdA와 동일한 타일 식별자를 갖는 타일의 타일 위치는 움직임 벡터 앵커 위치 도출에 사용된다. 이 실시예는 같은 타일 식별자 값에 근거하여 픽처 사이의 타일 대응을 설정한다. 인코딩을 위한 실시예에서, 움직임 벡터 앵커 위치를 도출하기 위한 타일 식별자가 슬라이스 위치 정보에 대한 타일 식별자와 같은 것으로 추정된다는 것은 비트스트림에 또는 비트스트림을 따라, 예를 들어 PPS, SPS 또는 VPS에 표시된다. 실시예에서, 움직임 벡터 앵커 위치를 도출하기 위한 타일 식별자가 슬라이스 위치 정보에 대한 타일 식별자와 같은 것으로 추정된다는 것은 비트스트림으로부터 또는 비트스트림을 따라, 예를 들어 PPS, SPS 또는 VPS로부터 디코딩된다. 표시는 타일 식별자의 표시된 세트로 특정될 수 있거나 또는 모든 타일 식별자에 적용될 수 있다. 예를 들어, 다음의 신팩스 등이 모든 타일 식별자에 적용되는 표시에 사용될 수 있다.

pic_parameter_set_rbsp() {	Descriptor
...	
tiles_enabled_flag	u(1)
if(tiles_enabled_flag) {	
...	
tile_id_relative_motion_flag	u(1)
...	
}	
...	
}	

[0189]

tile_id_relative_motion_flag는 움직임 벡터 앵커 위치를 도출하기 위한 수평 및 수직 오프셋이 사용되지 않는지 여부를 지정한다. **tile_id_relative_motion_flag**가 0인 경우, 이는 움직임 벡터 앵커 위치를 도출하기 위한 수평 및 수직 오프셋이 사용되지 않음을 표시한다. **tile_id_relative_motion_flag**가 1인 경우, 이는 움직임 벡터 앵커 위치를 도출하기 위한 수평 및 수직 오프셋이 사용 중이고 전술된 것과 같이 코딩된 타일의 타일 식별자로부터 도출되는 것을 표시한다.

[0190]

일부 실시예에서, 움직임 벡터 앵커 위치를 도출하기 위한 타일 식별자는 블록 인덱스를 수반하고, 블록 인덱스는 타일 내의 블록에 대한 블록 위치를 표시하는 신팩스 요소를 포함한다. 예를 들어, 블록 인덱스는 단일 신팩스 요소, 예를 들어 **block_index_wrt_tile_id** 신팩스 요소 또는 2개의 신팩스 요소, 예를 들어 **first_block_in_tile_flag** 및 **block_index_wrt_tile_id**로 표시될 수 있다. **first_block_in_tile_flag**는 수평 및 수직 오프셋이 타일의 좌상단과 일치하는지 여부를 표시한다. **block_index_wrt_tile_id**는 타일 내의 블록 위치를 표시하고, 수평 및 수직 오프셋은 사전 정의된 방식으로 블록 위치로부터, 예를 들어 블록 위치의 좌상단으로부터 도출된다. 일부 실시예에서, 움직임 벡터 앵커 위치를 도출하기 위한 블록 인덱스의 존재는 픽처 파라미터 세트에서 제어된다. 슬라이스 위치 정보를 표시하는 타일 식별자에 대한 블록 인덱스와 관련된 실시예는 움직임 벡터 앵커 위치를 도출하기 위한 타일 식별자에 대한 블록 인덱스와 유사하게 적용된다.

[0192]

일부 실시예에서, 움직임 벡터 앵커 위치를 도출하기 위한 참조 타일 식별자는 참조 픽처 리스트(들)의 표시된 픽처에 대해 (예를 들어, 슬라이스 헤더에) 제공될 수 있다. 예를 들어, 참조 픽처 리스트의 현재 발생된 픽처의 각각에 대해, 슬라이스 헤더는 조건부 움직임 벡터 앵커 위치가 해당 참조 인덱스에 대해 사용 중인지 여부를 표시하는 플래그를 포함할 수 있고, 사용 중인 경우, 슬라이스 헤더는 또한 움직임 벡터 앵커 위치를 표시하는 해당 참조 인덱스의 **tile_id**를 포함한다.

[0193]

예시적인 실시예에서, 슬라이스 세그먼트 헤더의 신팩사는 다음과 같고, 이는 움직임 벡터 앵커 위치의 도출을 위한 타일 식별자 값을 표시하기 위해, 여기에서는 **ref_pic_anchor_offsets()**으로 지정되는 신팩스 구조를 포함한다.

slice_segment_header() {	Descriptor
...	
if(slice_type == P slice_type == B) {	
num_ref_idx_active_override_flag	u(1)
if(num_ref_idx_active_override_flag) {	
num_ref_idx_l0_active_minus1	ue(v)
if(slice_type == B)	
num_ref_idx_l1_active_minus1	ue(v)
}	
if(lists_modification_present_flag && NumPicTotalCurr > 1)	
ref_pic_lists_modification()	
...	
ref_pic_anchor_offsets()	
...	
}	
...	
}	

[0194]

ref_pic_anchor_offsets() {	Descriptor
for(i = 0; i <= num_ref_idx_l0_active_minus1; i++)	
if(IsCurrPic(RefPicList0[i])) {	
mv_anchor_pos_l0_flag[i]	u(1)
if(mv_anchor_pos_l0_flag[i])	
anchor_tile_id_l0[i]	u(v)
}	
if(slice_type == B) {	
for(i = 0; i <= num_ref_idx_l1_active_minus1; i++)	
if(IsCurrPic(RefPicList1[i])) {	
mv_anchor_pos_l1_flag[i]	u(1)
if(mv_anchor_pos_l1_flag[i])	
anchor_tile_id_l1[i]	u(v)
}	
}	
}	

[0195]

[0196]

num_ref_idx_active_override_flag는 **num_ref_idx_l0_active_minus1** 선택스 요소가 P 및 B 슬라이스에 대해 존재하는지 그리고 **num_ref_idx_l1_active_minus1** 선택스 요소가 B 슬라이스에 대해 존재하는지 여부를 지정한다. **num_ref_idx_active_override_flag**가 1인 경우, **num_ref_idx_l0_active_minus1** 선택스 요소는 P 및 B 슬라이스에 대해 존재하고 **num_ref_idx_l1_active_minus1** 선택스 요소는 B 슬라이스에 대해 존재한다. **num_ref_idx_active_override_flag**가 0인 경우, **num_ref_idx_l0_active_minus1** 및 **num_ref_idx_l1_active_minus1** 선택스 요소는 존재하지 않는다. **num_ref_idx_l0_active_minus1**은 슬라이스를 디코딩하는 데 사용될 수 있는 참조 픽처 리스트 0에 대한 최대 참조 인덱스를 지정한다. 현재 슬라이스가 P 또는 B 슬라이스이고 **num_ref_idx_l0_active_minus1**이 존재하지 않을 때, **num_ref_idx_l0_active_minus1**은 PPS(그러나 이와 다른 설계에서는 SPS 또는 VPS에 있을 수 있음)에서 제공되는 **num_ref_idx_l0_default_active_minus1**과 동일한 것으로 추정된다. **num_ref_idx_l1_active_minus1**은 슬라이스를 디코딩하는 데 사용될 수 있는 참조 픽처 리스트 1에 대한 최대 참조 인덱스를 지정한다. **num_ref_idx_l1_active_minus1**이 존재하지 않을 때, **num_ref_idx_l1_active_minus1**은 PPS(그러나 이와 다른 설계에서는 SPS 또는 VPS에 있을 수 있음)에서 제공되는 **num_ref_idx_l1_default_active_minus1**과 동일한 것으로 추정된다.

[0197]

IsCurrPic(picInp)는, **picInp**가 인코딩 또는 디코딩 중인 현재 픽처가 아닐 때 0을 반환하고, **picInp**가 인코딩 또는 디코딩 중인 현재 픽처일 때 1을 반환하는 함수이다. 이하에서는, X는 참조 픽처 리스트 0 또는 1에 대해 각각 0 또는 1이다. **mv_anchor_pos_lX_flag[i]**는 움직임 벡터 앵커 위치가 이 참조 픽처에 대한 인터 예측에 사용되는지 여부를 지정한다. **mv_anchor_pos_lX_flag[i]**가 0인 경우, 이는 움직임 벡터 앵커 위치가 이 참조 픽처에 대한 인터 예측에 사용되지 않는 것을 표시한다. **mv_anchor_pos_lX_flag[i]**가 1인 경우, 이는 움직임 벡터 앵커 위치가 이 참조 픽처에 대한 인터 예측에 사용되는 것을 표시한다. **anchor_tile_id_lX[i]**가 0인 경우, 이는 다른 실시예에서 설명된 것과 같이 타일 식별자가 움직임 벡터 앵커 위치를 도출하기 위해 획득된 타

일 위치에 근거하는 것임을 표시한다.

- [0198] 일부 실시예에서, 참조 tile_id 식별자가 (예를 들어, 슬라이스 헤더에) 제공되거나 또는 참조 픽처 리스트(들)의 픽처에 대해 추정될 수 있다. 예를 들어, 타일 식별자는 참조 픽처 리스트(들)의 현재 픽처의 각각의 경우에 대해 (예를 들어, 슬라이스 헤더에서) 제공될 수 있다. 시간적 참조 픽처 또는 현재 픽처 이외의 참조 픽처에 대해, 참조 tile_id 식별자는 현재 tile_id 식별자와 동일한 것으로 추정될 수 있거나, 또는 움직임 벡터 앵커 위치는 사용되지 않는다고 추정될 수 있다.
- [0199] 일부 실시예에서, 타일 식별자에 추가하여, 움직임 벡터 앵커 위치에 대한 수평 및 수직 오프셋은 타일의 좌상단과 같이 타일 식별자에 의해 표시되는 타일의 사전 정의된 위치에 관련된 블록 위치를 표시하는 블록 인덱스로 표시될 수 있다.
- [0200] 일부 실시예에서, 타일 식별자에 추가하여, 움직임 벡터 앵커 위치에 대한 수평 및 수직 오프셋은 타일의 좌상단과 같이 타일 식별자에 의해 표시되는 타일의 사전 정의된 위치에 관련된 x 및 y 픽셀 좌표로 표시될 수 있다.
- [0201] 일부 실시예에서, 앵커 위치는 TMVP 동작 또는 유사한 동작에서 병치된 픽처에 대한 병치된 블록을 결정할 때 적용되도록 구성된다. 앵커 위치는, 예를 들어 사전 정의된 조건이 참일 때 또는 인코더에 의해 표시될 때, 조건부로 적용될 수 있다. 앵커 위치는 다양한 실시예에서 움직임 벡터 도출을 위한 앵커 위치가 도출되는 방법과 동일하거나 또는 유사하게 도출될 수 있다.
- [0202] 일부 실시예에서, TMVP를 위한 병치 픽처가 현재 픽처인 것으로 표시될 때, 움직임 벡터 앵커 위치는 병치 블록을 발견하기 위해 적용된다. 즉, 병치 블록은 (tile_x + rel_x, tile_y + rel_y)에 근거하여 선택되고, tile_x 및 tile_y는 참조 tile_id를 통해 획득될 수 있다. 참조 tile_id는 타일 식별자 기반 움직임 벡터 앵커 위치에 대해 전술된 것과 같이 표시될 수 있다. 이와 달리, 참조 tile_id는 병치된 픽처가 현재 픽처인 것으로 표시될 때 조건부로, 예를 들어 슬라이스 헤더에 제공될 수 있다.
- [0203] HEVC 슬라이스 헤더 신팩스는 first_slice_segment_in_pic_flag를 포함한다. 1일 때, 이는 슬라이스 세그먼트가 디코딩 순서에서 픽처의 제 1 슬라이스 세그먼트임을 지정한다. first_slice_segment_in_pic_flag가 0인 경우, 이는 슬라이스 세그먼트가 디코딩 순서에서 픽처의 제 1 슬라이스 세그먼트가 아님을 지정한다. 이 신팩스 요소는 디코딩 순서에서 연속적인 코딩된 픽처 사이의 경계의 검출에 사용될 수 있다. 슬라이스가 재정렬되거나 또는 슬라이스의 서브세트가 추출되는 경우, first_slice_segment_in_pic_flag(또는 유사한 정의를 가진 신팩스 요소)를 다시 작성해야 할 수 있다. 하나의 방법에서, first_slice_segment_in_pic_flag 등은 슬라이스 헤더 신팩스에 포함되지 않지만 픽처의 제 1 슬라이스 세그먼트에 대해 픽처의 다른 슬라이스 세그먼트에 대해서와는 다른 시작 코드가 사용된다. 다른 방법에서, first_slice_segment_in_pic_flag 등은 슬라이스 헤더 신팩스에 포함되지 않지만 픽처 순서 카운트와 같은 다른 신팩스 요소(들)가 픽처의 제 1 슬라이스 세그먼트를 결정하는 데 사용된다. 실시예에서, first_slice_segment_in_pic_flag 등은 슬라이스 헤더 신팩스에 포함되지 않지만 타일 식별자는 픽처의 제 1 슬라이스 세그먼트를 결정하는 데 사용된다. 타일 식별자가 목시적으로 타일 위치에 할당될 때, 0과 같은 특정 타일 식별자 값(예를 들어, slice_tile_id가 0)을 가진 슬라이스 세그먼트가 픽처의 제 1 슬라이스 세그먼트로 결정된다. 타일 식별자가 다른 실시예에서 설명된 것과 같이 명시적으로 타일 위치에 할당될 때, 명시적으로 표시된 특정 타일 식별자 값을 가진 슬라이스 세그먼트가 픽처의 제 1 세그먼트로 결정된다. 예를 들어, 타일 위치에 대한 타일 식별자 할당의 순서는 전술된 것과 같이 PPS에 표시될 수 있고, PPS에 주어진 리스트의 제 1 타일 식별자와 동일한 타일 식별자를 갖는 슬라이스 세그먼트가 픽처의 제 1 슬라이스 세그먼트로 결정된다.
- [0204] 일부 실시예에서, 장치(10)는 타일의 각각에 대한 디코딩 가능성에 영향을 주지 않고 타일이 재정렬될 수 있는지 여부를 표시하는 재정렬 표시를 포함하는 비트스트림에 또는 비트스트림을 따라 비트스트림과 관련된 디코딩 순서 제한 파일을 인코딩하기 위한 프로세싱 회로(12)와 같은 수단을 더 포함한다. 예를 들어, 재정렬 표시는 특정 타일 세트 내의 타일이 이들의 디코딩 가능성에 영향을 주지 않고 재정렬될 수 있거나 또는 (종속성 순서가 준수되는 한) 재정렬되지 않을 수 있음을 표시할 수 있다. 이와 달리, 재정렬 표시는 움직임 제한 타일 세트와 같은 타일 또는 표시되거나 또는 추정된 타일 세트가 이들의 디코딩 가능성에 영향을 주지 않고 정렬될 수 있거나 또는 정렬되지 않을 수 있음을 표시할 수 있다.
- [0205] 재정렬 표시는 다음 중 하나, 둘 중 하나 또는 둘 모두를 표시할 수 있다. 즉, 예를 들어, 전술된 것과 같이 타일 식별자 값의 리스트로서, 수정된 순서가 명시적으로 제공되는 한, 타일은 타일의 각각에 대한 디코딩 가능

성에 영향을 주지 않고 재정렬될 수 있다. 타일은, 비트스트림에 또는 비트스트림을 따라 수정 없이 타일의 각각에 대한 디코딩 가능성에 영향을 주지 않고 재정렬될 수 있다. 후자의 경우, 타일은 (예를 들어, CTU의 너비 및 높이에서) 같은 크기일 필요가 있을 수 있다.

[0206] 일부 실시예에서, 인코더는, 예를 들어 전술된 표시(들)에 근거하여, 표시되거나 또는 추정된 타일 그리드를 따르는 임의의 타일 디코딩 순서가 사용될 수 있다고 결정할 수 있다. 인코더는 표시되거나 또는 추정된 타일 그리드를 따르는 임의의 타일 디코딩 순서가 사용될 수 있다는 특정 표시를, 예를 들어 시퀀스 파라미터 세트 또는 SEI 메시지에 표시할 수 있다.

[0207] 일부 실시예에서, 표시되거나 또는 추정된 타일 그리드를 따르는 임의의 타일 디코딩 순서가 비트스트림에 또는 비트스트림을 따라 수정 없이 사용될 수 있다고 결정될 때, 인코더는 명시적 tile_id 리스트의 인코딩을 생략할 수 있고, 및/또는 디코더는 디코딩 순서 또는 타일을 표시하는 명시적 tile_id 리스트 등의 디코딩을 생략할 수 있다.

[0208] 일부 실시예에서, 디코더는, 예를 들어, 전술된 표시(들)에 근거하여, 표시되거나 또는 추정된 타일 그리드를 따르는 임의의 타일 디코딩 순서가 사용될 수 있음을 결정할 수 있다. 예를 들어, 각각의 타일이 움직임 제한 타일 세트를 형성하는 것이 비트스트림에 또는 비트스트림을 따라 표시될 수 있고, 디코더는 이 표시 및 타일 관련 어드레싱의 사용에 근거하여 임의의 타일 디코딩 순서가 사용될 수 있음을 결정할 수 있다. 예를 들어, 모든 타일의 크기가 동일한 경우, 디코더는 꽉쳐의 타일을 임의의 순서로 디코딩할 수 있고, 이는 타일의 인코딩 순서와 같을 필요가 없다. 예를 들어, 디코더는 수신 순서대로 타일을 디코딩할 수 있다. 타일 식별자를 활용하면 슬라이스 헤더의 재작성 없이, 예를 들어 파라미터 세트의 수정만으로 또는 더 간단한 슬라이스 헤더의 수정을 통해 타일 세트의 추출이 수행될 수 있다. 또한, 타일의 재정렬은 슬라이스 헤더의 재작성 없이, 예를 들어 파라미터 세트의 수정만으로 또는 더 간단한 슬라이스 헤더의 변경을 통해 수행될 수 있다. 또한, 슬라이스 헤더는 더 짧아질 것이다. 예를 들어, 8192x4096의 꽉쳐의 16x8의 타일 그리드, 64x64의 CTU, HEVC에서와 같은 고정 길이 코딩의 경우, 슬라이스 세그먼트 주소는 13-비트 코드 워드를 사용할 것이다. 타일 식별자를 활용함으로써 절약되는 꽉쳐 당 비트의 개수는 468개로, 이는 60Hz 비디오에서 초 당 약 28킬로비트에 대응한다.

[0209] 이제 도 4를 참조하면, 예시적인 실시예에 따라 IPPE(in-picture-prediction-enabled) 타일 세트를 구성하기 위해 도 1의 장치(10)와 같은 장치에 의해 수행되는 동작이 도시된다. 블록(40)에 도시된 것과 같이, 장치는 하나 이상의 타일을 포함하는 코딩된 꽉쳐를 포함하는 비트스트림을 수신하기 위한 프로세싱 회로(12), 통신 인터페이스(16) 등과 같은 수단을 포함한다. 이와 달리, 장치는 하나 이상의 타일을 포함하는 코딩된 꽉쳐를 포함하는 비트스트림을 인코딩하기 위한 프로세싱 회로(12)와 같은 수단을 포함한다. 인코딩은, 예를 들어 압축되지 않은 꽉쳐 시퀀스를 입력으로 취할 수 있다. 비트스트림은 HEVC, H.266/VVC 또는 전술된 임의의 파일 포맷에 따라 포맷된 비트스트림일 수 있다.

[0210] 블록(42)에 도시된 것과 같이, 장치(10)는 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별하기 위한 프로세싱 회로(12)와 같은 수단을 포함한다. 일부 실시예에서, 장치는 인코딩에 사용될 구성 설정 또는 이러한 타일 종속성 정보를 수신한다. 일부 실시예에서, 장치는, 예를 들어 제 2 뷔의 타일이 제 1 뷔의 각각의 타일에 종속할 수 있는 타일 종속성을 인코딩의 일부로서 결정하고, 제 1 및 제 2 뷔는 프레임 패킹된다. 일부 실시예에서, 장치는 움직임 벡터 앵커 위치 도출을 위한 타일 식별자와 같은, 비트스트림의 각각의 타일과 관련된 메타데이터 파일을 분석하여 하나 이상의 종속 타일을 식별한다. 일부 실시예에서, 장치는 각각의 타일의 종속성을 결정한다. 하나 이상의 종속 타일 각각은 코딩된 꽉쳐 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 함께 논리적 원자 단위(logical atomic unit)를 형성한다.

[0211] 블록(44)에 도시된 것과 같이, 장치(10)는 하나 이상의 종속 타일을 포함하는 인-꽉쳐-예측-가능 타일 세트의 표시를 구성하기 위한 프로세싱 회로(12)와 같은 수단을 포함한다. 일부 실시예에서, IPPE 타일 세트 내의 타일은 인-꽉쳐 예측을 위해 IPPE 타일 세트 외부의 임의의 타일에는 종속하지 않는다. 일부 실시예에서, 특정 IPPE 타일 세트 외부의 타일은 IPPE 타일 세트 내의 임의의 타일에 종속하지 않는다는 것이 더 표시될 수 있다.

[0212] 블록(46)에 도시된 것과 같이, 장치(10)는 비트스트림 및 인-꽉쳐-예측-가능 타일 세트의 표시를 저장하도록 하기 위한 프로세싱 회로(12)와 같은 수단을 포함한다. IPPE 타일 세트는 비트스트림에 따라 또는 비트스트림의 내부에 저장될 수 있다. IPPE 타일 세트의 표시는, 예를 들어 시퀀스 파라미터 세트 파일, 꽉쳐 파라미터 세트 파일, 보충 개선 정보 메시지 파일 등에 있을 수 있다.

[0213]

일부 실시예에서, IPPE 타일 세트는 또한 표시에 의해 또는 다른 방식으로(예를 들어, 시간적 움직임 제한 타일 세트 SEI 메시지를 사용하여) 시간적 움직임 제한 타일 세트인 것으로 표시될 수 있다. IPPE 타일 세트 및 시간적 움직임 제한 타일 세트 모두인 타일 세트는 공간적-시간적 랜덤 액세스 및 추출에 사용될 수 있다. 공간적 랜덤 액세스는 타일의 서브세트만이 렌더링에 필요한 상황이고, 표시를 활용하여 타일의 서브세트의 정확한 디코딩을 위해 어떤 타일이 필요한지 결정할 수 있다. 추출에서는, 표시를 활용하여 어떤 타일이 다른 타일과 독립적으로 추출 및 디코딩될 수 있는지 결정할 수 있다.

[0214]

일부 실시예에서, IPPE 타일 세트의 타일 사이에서 인-픽처 예측(예를 들어, 인트라 블록 카피)이 가능해질 수 있다. 실시예에서, IPPE 타일 세트의 타일에 걸쳐 가능하게 된 인-픽처 예측은, 인트라 예측, 인트라 블록 카피 예측, 참조 타일의 블록으로부터 같은 IPPE 타일 세트의 현재 타일의 각 블록으로 움직임 벡터 후보를 도출, 같은 IPPE 타일 세트의 인접 타일에 걸친 디블로킹 필터링과 같은 타일에 걸친 인-루프 필터링, 컨텍스트 적응형 엔트로피 코딩에서 타일에 걸친 컨텍스트 상태 업데이트 및 활용 중 하나 이상을 포함한다. 실시예에서, 슬라이스는 슬라이스 시작 위치 및/또는 슬라이스 종료 위치가 타일 경계와 정렬되지 않을 때에도 IPPE 타일 세트 내의 타일 경계에 걸쳐 연장될 수 있다. 실시예에서, 슬라이스 또는 IPPE 타일 세트 내의 VCL(Video Coding Layer) NAL 단위 경계에서 사전 정의되거나 또는 표시된 타입의 인-픽처 예측이 불가능하게 될 수 있다. 예를 들어, VCL NAL 단위 경계에 걸쳐 엔트로피 코딩 상태를 전파하는 것은 불가능하게 될 수 있고, 이는 VCL NAL 단위의 병렬 엔트로피 인코딩 및 디코딩을 가능하게 한다.

[0215]

일부 실시예에서, 인-픽처 예측은 슬라이스 경계에 걸쳐 가능하게 될 수 있고, 슬라이스의 인-픽처-예측-가능 세트는 IPPE 타일 세트와 유사하게 표시될 수 있다. 마찬가지로, 인-픽처 예측이 가능하게 되는 슬라이스와 타일의 혼재가 표시될 수 있고 IPPE 세트로 지칭될 수 있다.

[0216]

예시적인 실시예에서, 하나 이상의 IPPE 세트를 표시하기 위해 다음의 신택스 등이 사용된다. 신택스는 조건부로, 예를 들어 VPS, SPS, PPS 및/또는 SEI 메시지의 신택스 구조에 포함될 수 있다. 포함하는 구조에 ippe_sets() 신택스의 부재는 IPPE 세트의 부재를 표시할 수 있다.

ippe_sets() {	Descriptor
each_grid_cell_one_ippe_set_flag	u(1)
if(each_grid_cell_one_ippe_set_flag) {	
num_sets_minus1	ue(v)
for(i = 0; i <= num_sets_minus1 ; i++) {	
ippe_set_id[i]	ue(v)
num_cell_rects_in_set_minus1[i]	ue(v)
for(j = 0; j <= num_cell_rects_in_set_minus1[i] ; j++) {	
top_left_cell_index[i][j]	ue(v)
bottom_right_cell_index[i][j]	ue(v)
}	
}	
}	
}	

[0217]

each_grid_cell_one_ippe_set_flag는 픽처의 (다른 실시예에 따라 적용 가능한) 식별자 그리드 또는 타일 그리드의 각각의 그리고 모든 그리드 셀이 IPPE 세트에 포함되는지 여부를 지정한다. **each_grid_cell_one_ippe_set_flag**가 1인 경우, 픽처의 식별자 그리드 또는 타일 그리드의 각각의 그리고 모든 그리드 셀은 IPPE 세트에 포함된다. **each_grid_cell_one_ippe_set_flag**가 0인 경우, 이는 이러한 제한이 적용되지 않을 수 있음을 표시한다. **num_sets_minus1**은 이 신택스 구조에서 식별된 IPPE 세트의 개수에서 1을 뺀 것을 지정한다. **ippe_set_id[i]**는 i번째 IPPE 세트의 식별자를 지정한다. **num_cell_rects_in_set_minus1[i]**는 i번째 IPPE 세트의 (식별자 그리드 또는 타일 그리드 중 다른 실시예에 따라 적용 가능한 것의) 그리드 셀의 직사각형 영역의 개수에서 1을 뺀 것을 지정한다. **top_left_cell_index[i][j]** 및 **bottom_right_cell_index[i][j]**는 각각 i번째 식별된 IPPE 세트의 직사각형 영역에서 좌상 셀의 그리드 셀 위치 및 i번째 식별된 IPPE 세트의 직사각형 영역에서 우하(bottom-right) 셀의 그리드 셀 위치를 (식별자 그리드 또는 타일 그리드 중 다른 실시예에 따라 적용 가능한 것의) 타일 래스터 스캔 순서로 식별한다.

[0219]

실시예에서, 타일 및/또는 슬라이스의 움직임 제한 세트는 비트스트림에 또는 비트스트림을 따라, 예를 들어 VPS, SPS, PPS 및/또는 SEI 메시지의 신택스 구조에 표시된다. 포함하는 구조에서 움직임 제한 표시 신택스의 부재는 움직임 제한 타일 및/또는 슬라이스 세트의 부재를 표시할 수 있다. 예를 들어, 다음의 신택스가 사용

될 수 있다.

motion_constrained_sets()	Descriptor
each_grid_cell_one_mc_set_flag	u(l)
if(!each_grid_cell_one_mc_set_flag) {	
num_sets_minus1	ue(v)
for(i = 0; i <= num_sets_minus1; i++) {	
mc_set_id[i]	ue(v)
num_cell_rects_in_set_minus1[i]	ue(v)
for(j = 0; j <= num_cell_rects_in_set_minus1[i]; j++) {	
top_left_cell_index[i][j]	ue(v)
bottom_right_cell_index[i][j]	ue(v)
}	
{	
{	
}	

[0220]

[0221]

each_grid_cell_one_mc_set_flag는 픽처의 (다른 실시예에 따라 적용 가능한) 식별자 그리드 또는 타일 그리드의 각각의 그리고 모든 그리드 셀이 타일 또는 슬라이스의 움직임 제한 세트에 포함되는지 여부를 지정한다. each_grid_cell_one_mc_set_flag가 1인 경우, 이는 픽처의 (다른 실시예에 따라 적용 가능한) 식별자 그리드 또는 타일 그리드의 각각의 그리고 모든 그리드 셀이 타일 또는 슬라이스의 움직임 제한 세트에 포함되는 것을 표시한다. each_grid_cell_one_mc_set_flag가 0인 경우, 이는 이러한 제한이 적용되지 않을 수 있음을 표시한다. num_sets_minus1은 이 신팩스 구조에서 식별된 타일 또는 슬라이스의 움직임 제한 세트의 개수에서 1을 뺀 것을 지정한다. mc_set_id[i]는 i번째 타일 또는 슬라이스의 움직임 제한 세트의 식별자를 지정한다. num_cell_rects_in_set_minus1[i] + 1은 i번째 움직임 제한 세트의 (식별자 그리드 또는 타일 그리드 중 다른 실시예에 따라 적용 가능한 것의) 그리드 셀의 직사각형 영역의 개수를 지정한다. top_left_cell_index[i][j] 및 bottom_right_cell_index[i][j]는 각각 i번째 식별된 움직임 제한 세트의 직사각형 영역에서 좌상 셀의 그리드 셀 위치 및 i번째 식별된 움직임 제한 세트의 직사각형 영역에서 우하 셀의 그리드 셀 위치를 (식별자 그리드 또는 타일 그리드 중 다른 실시예에 따라 적용 가능한 것의) 타일 래스터 스캔 순서로 식별한다.

[0222]

실시예에서, 인코더는 비트스트림에 또는 비트스트림을 따라, 예를 들어 VPS 또는 SPS에, 타일 또는 슬라이스의 움직임 제한 세트가 픽처 경계처럼 취급되어야 하는지를 표시하고, 및/또는 디코더는 비트스트림으로부터 또는 비트스트림을 따라, 예를 들어 VPS 또는 SPS로부터, 이를 디코딩한다. 이러한 표시는 신팩스에 존재할 수 있거나, 또는 이러한 표시는, 타일 또는 슬라이스의 움직임 제한 세트가 코딩된 비디오 시퀀스에 걸쳐 같은 직사각형 모양 및 같은 크기일 것, 코딩된 비디오 시퀀스의 픽처 내에 같은 공간적 위치에 있을 것 또는 재정렬 표시가 타일 또는 슬라이스 각각에 대한 디코딩 가능성에 영향을 주지 않고 타일 또는 슬라이스가 재정렬될 수 있음을 표시할 것과 같지만 반드시 이에 한정되지는 않는 특정 제한을 충족할 때에만 유효한 것으로 표시될 수 있다. 타일 또는 슬라이스의 움직임 제한 세트의 경계를 픽처 경계로 취급하는 것은, 인터 예측에서 참조되는 샘플 위치를 움직임 제한 세트 내에 있도록 포화시키는 것과, 예를 들어 다양한 예측 프로세스에서, 타일 또는 슬라이스의 움직임 제한 세트의 외부에 있는 블록의 신팩스 요소 및 도출된 변수(예를 들어, 움직임 벡터)에 대한 참조를 이용 불가능한 데이터에 대한 참조로 처리하는 것 중 하나 이상을 포함 할 수 있지만 반드시 이에 한정되는 것은 아니다. 타일 또는 슬라이스의 움직임 제한 세트의 경계를 이와 같이 처리하면 움직임 벡터가 경계를 넘어 샘플 위치를 효과적으로 참조할 수 있게 되고, 경계에서의 샘플 값은 경계를 넘어 참조된 샘플 위치에 대한 샘플 값으로 사용된다. 일부 실시예에서, IPPE 세트의 표시는 또한 다른 타일로부터 인-픽처 예측되지 않은 하나 이상의 타일 및 같은 IPPE 타일 세트의 다른 타일에 대한 예측 종속성을 표시하는 종속성 포레스트 또는 그래프를 포함한다.

[0223]

일부 실시예에서, IPPE 타일 세트에서 종속 타일의 인트라 블록 카피 움직임 벡터에 대한 앵커 위치는 같은 IPPE 타일 세트에서 식별된 참조 타일에 관련된 것으로 표시되거나 또는 추정된다. 일부 실시예에서, 앵커 위치는 현재 픽처가 선택된 참조 픽처인 경우(예를 들어, 인트라 블록 카피가 사용 중일 때) 적용될 수 있고, 그렇지 않은 경우에는 적용되지 않을 수 있다. 전술된 것과 같은 타일 식별자는 앵커 위치를 표시하는 데 사용될 수 있다.

[0224]

플렌옵틱(plenoptic)(일명, 조밀한 라이트필드) 또는 멀티뷰 비디오를 코딩하기 위한 예시적인 실시예에서, 하나의 시간 인스턴스의 모든 뷰로부터의 플렌옵틱 이미지 또는 픽처는 서브-조리개 이미지 또는 하나의 뷰마다

하나의 코딩된 픽처, 하나의 타일 또는 타일 세트로 인코딩된다. 또한, 인-픽처 예측은 선택된 타일 또는 타일 세트 사이에서 가능해질 수 있다. 인코더는 다른 타일이나 또는 타일 세트가 예측되지 않는, 예를 들어 각각의 "리프(leaf)" 타일 또는 타일 세트에 대해 IPPE 타일 세트를 표시한다.

[0225] 투영된 포인트 클라우드의 비디오 코딩과 같은 패치(patch) 기반 비디오와 관련되는 예시적인 실시예에서, 프로세싱 회로(12)와 같은 장치(10)는 하나 이상의 전체 패치가 타일 세트를 형성하는 방식으로 다른 타입의 패치(예를 들어, 텍스처, 깊이, 제 2 레이어 텍스처, 제 2 레이어 깊이, 반사율 등)를 하나의 코딩된 픽처로서 인코딩할 수 있다. 장치는 다른 타입의 병치된 패치의 타일 세트와 같은 관련된 타일 세트를 포함하도록 IPPE 타일 세트를 선택하고 표시할 수 있다. 인-픽처 예측이 IPPE 타일 세트의 타일 사이에서 가능해질 수 있다. 예를 들어, 인트라 블록 카피 예측은 제 1 레이어 텍스처 타일로부터 병치된 제 2 레이어 텍스처 타일로 가능해질 수 있거나, 또는 다른 실시예에서 설명된 것과 같이 움직임 벡터 예측이 제 1 타입(예를 들어, 텍스처)의 타일로부터 제 2 타입(예를 들어, 깊이)의 병치된 타일로 가능해질 수 있다.

[0226] 이제 도 5a를 참조하면, 예시적인 NxM 타일 그리드(50)가 플렌옵틱 이미지/비디오를 타일마다 하나의 서브-조리개 이미지로 코딩하기 위해 사용된다($N = M = 5$). 인-픽처 예측은 픽처의 타일 내에서 사용될 수 있다. IPPE 타일 세트(52, 54)는 타일 그리드와 관련된 비트스트림에 또는 비트스트림을 따라 표시될 수 있다. 움직임 제한 타일 세트(motion-constrained tile set, MCTS)가, 예를 들어 타일마다 하나의 MCTS가 비트스트림에 또는 비트스트림을 따라 인코딩되고 또는 표시될 수 있다. 타일 그리드(50)의 각각의 타일은 자신에 할당된 타일 식별자를 갖는다. 타일 식별자는 타일 그리드 내의 타일의 래스터 스캔 순서를 표시한다. 다음의 tile_id 할당은 비트스트림에 또는 비트스트림을 따라 표시될 수 있다.

24, 20, 9, 13, 21,
19, 8, 1, 5, 14,
12, 4, 0, 2, 10,
18, 7, 3, 6, 15,

[0227] 23, 17, 11, 16, 22

[0228] IPPE 타일 세트(52)와 같은 IPPE 타일 세트가 추출될 수 있다. 도 5b는 추출된 IPPE 세트(52)를 도시한다.

[0229] 추출된 비트스트림에 대해, 다음의 tile_id 할당이 비트스트림에 또는 비트스트림을 따라 표시될 수 있다(타일 그리드 내의 타일의 래스터 스캔 순서에서 tile_id 순서를 표시함).

0, 2, 10,
3, 6, 15,
[0230] 11, 16, 22

[0231] 전술된 것과 같이, 타일 위치에 대한 타일 식별자 값의 할당은, 예를 들어 픽처 파라미터 세트에 표시될 수 있다. 전술된 것과 같이 ISOBMFF 트랙의 샘플 엔트리는 파라미터 세트를 포함할 수 있다. 따라서, 추출된 비트스트림은 원하는 슬라이스 및/또는 타일의 VCL NAL 단위를 추출하고 파라미터 세트(들)의 타일 식별자 값의 할당을 수정함으로써 구성될 수 있다. 수정된 파라미터 세트는 샘플이 추출된 VCL NAL 단위를 포함하는 트랙에 대한 샘플 엔트리에 제공될 수 있다.

[0232] 슬라이스 헤더와 관련하여 예시적인 실시예가 전술되었다. 다른 실시예는 슬라이스 세그먼트 헤더 또는 타일 헤더와 같은, 픽처 파티셔닝 단위를 위한 임의의 다른 타입의 헤더에 동일하게 적용된다는 것이 이해될 필요가 있다.

[0233] 특정 예시적인 실시예가 타일과 관련하여 전술되었다. 다른 실시예는 슬라이스 또는 슬라이스 그룹과 같은, 임의의 다른 타입의 픽처 파티셔닝 단위 또는 시공간 단위에 동일하게 적용된다는 것이 이해될 필요가 있다. 타일 및 슬라이스는 정수 개수의 전체 코딩 트리 단위를 포함하는 것으로 전술되었다. 타일 및 슬라이스는 이들의 경계에서 더 세밀할 수 있다는 것이 이해될 필요가 있다. 예를 들어, 타일 너비 및 높이는 8x8 코딩 단위와 같은, 특정 너비 및 높이의 코딩 단위로 지정될 수 있다. 다른 예에서, 슬라이스는 정수 개수의 코딩 단위를 포함할 수 있지만 정수 개수의 전체 코딩 트리 단위를 가질 필요는 없다.

[0234] 특정 예시적인 실시예는 타일 그리드 및 타일 그리드에서 타일 위치를 표시하는 타일 식별자와 관련하여 전술되었다. 다른 실시예는 특정 예시적인 실시예에서 설명된 식별자 그리드 및 다른 픽처 파티셔닝 단위의 그리드(예를 들어, MCTS 그리드)에 동일하게 적용된다는 것이 이해될 필요가 있다. 유사하게, 실시예는 사용 중인 그

리드 내에서 위치의 할당이 도출될 수 있는 임의의 식별자에 적용된다.

[0235] 실시예는 또한 그리드 이외의 다른 타입의 꾹처 파티셔닝 수단에 적용된다. 예를 들어, 실시예에서, 래스터 스캔 순서에서 어떤 인접 그리드 셀이 병합되어 식별자 할당에서 하나의 셀을 형성하는지 표시될 수 있다. 이러한 병합은, 슬라이스 헤더에서 더 짧은 타일 식별자 코드워드를 달성하는 데 유용할 수 있다.

[0236] 예시적인 실시예가 인코더를 참조하여 전술되었지만, 산출되는 비트스트림 및 디코더가 그에 대응하는 요소를 그 내부에 가질 수 있다는 것이 이해될 필요가 있다. 예를 들어, 도 6에 도시된 것과 같이, 블록(60)에서 디코더는 꾹처를 수신하기 위한 수단을 포함할 수 있고, 꾹처는 하나 이상의 슬라이스를 포함한다. 블록(62)에 도시된 것과 같이, 디코더는 슬라이스의 슬라이스 헤더로부터 타일 위치를 표시하는 타일 식별자를 디코딩하기 위한 수단을 더 포함할 수 있고, 타일 식별자는 움직임 백터에 대한 앵커 위치를 결정하기 위해 슬라이스 위치를 표시하고, 및/또는 움직임 백터 후보를 획득하기 위해 앵커 위치를 결정하는 데 사용될 수 있다. 블록(64)에 도시된 것과 같이, 디코더는 파라미터 세트 파일의 타일 위치에 대한 타일 식별자 값의 할당을 디코딩하기 위한 수단을 더 포함할 수 있다.

[0237] 유사하게, 도 7에 도시된 것과 같이, 블록(70)에서, 디코더는 코딩된 꾹처를 포함하는 비트스트림을 수신하기 위한 수단을 포함할 수 있고, 코딩된 꾹처는 하나 이상의 타일을 포함한다. 블록(72)에 도시된 것과 같이, 디코더는 인-꺼처-예측-가능 타일 세트의 표시를 수신하기 위한 수단을 더 포함할 수 있고, 인-꺼처-예측-가능 타일 세트는 하나 이상의 타일로부터 하나 이상의 종속 타일을 식별하고, 하나 이상의 종속 타일은 코딩된 꾹처 내의 또 다른 종속 타일에 종속하거나 또는 또 다른 종속 타일과 함께 논리적 원자 단위를 형성한다. 블록(74)에 도시된 것과 같이, 디코더는 인-꺼처-예측-가능 타일 세트의 표시에 따라 비트스트림을 디코딩하기 위한 수단을 더 포함할 수 있다.

[0238] 마찬가지로, 예시적인 실시예가 디코더를 참조하여 설명된 경우, 인코더는 디코더에 의해 디코딩될 비트스트림을 생성하기 위한 구조 및/또는 컴퓨터 프로그램을 가질 수 있다는 것이 이해될 필요가 있다.

[0239] 예시적인 실시예가 신택스 및 시맨틱스를 참조하여 전술되었지만, 마찬가지로 실시예는 신택스 및 시맨틱스에 따라 비트스트림 부분을 출력하는 인코더를 커버한다는 것이 이해될 필요가 있다. 마찬가지로, 실시예는 신택스 및 시맨틱스에 따라 비트스트림 부분을 디코딩하는 디코더를 포함한다.

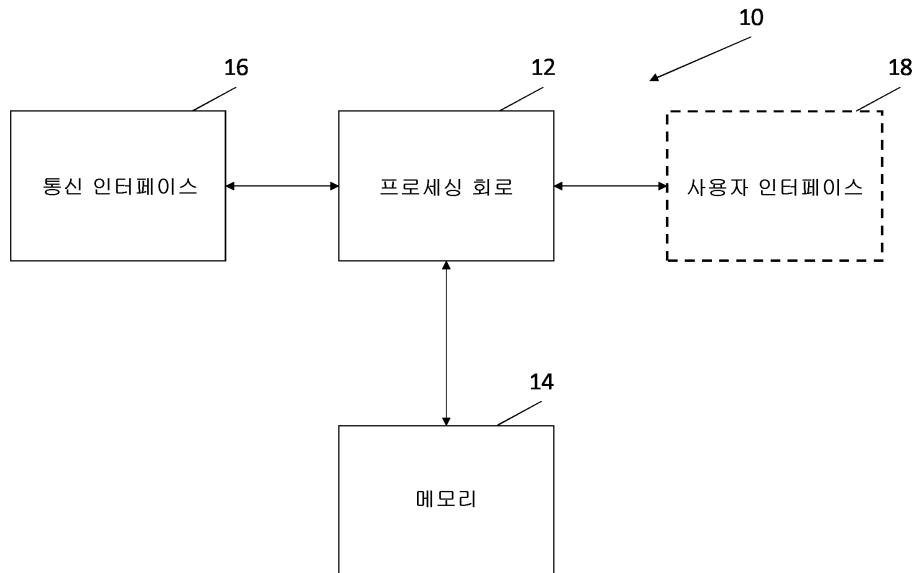
[0240] 전술된 본 발명의 실시예는 관련된 프로세스의 이해를 돋기 위해 별도의 인코더 및 디코더 장치의 관점에서 코덱을 설명한다. 그러나, 장치, 구조 및 동작은 단일 인코더-디코더 장치/구조/동작으로서 구현될 수 있다는 것이 이해될 것이다. 또한, 코더와 디코더는 일부 또는 모든 공통 요소를 공유할 수 있다.

[0241] 위의 예는 장치 내의 코덱 내에서 동작하는 본 발명의 실시예를 설명하지만, 청구 범위에 정의된 본 발명은 임의의 비디오 코덱의 일부로서 구현될 수 있음이 이해될 것이다. 따라서, 예를 들어 본 발명의 실시예는 고정 또는 유선 통신 경로를 통해 비디오 코딩을 구현할 수 있는 비디오 코덱으로 구현될 수 있다.

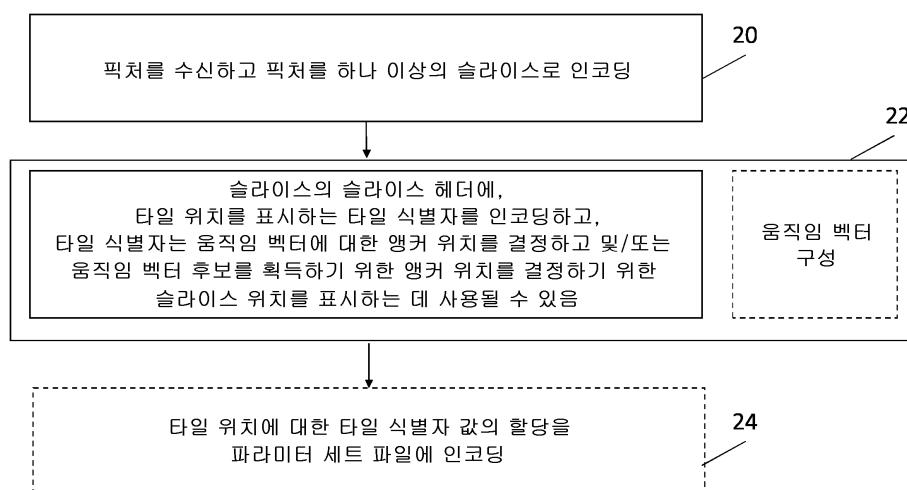
[0242] 여기에서 설명된 본 발명의 많은 수정 및 다른 실시예는 전술한 설명 및 관련 도면에서 제시된 교시의 이점을 갖는다는 것은 본 발명이 속하는 분야의 기술자에게 상기될 것이다. 따라서, 본 발명은 개시된 특정 실시예에 한정되지 않고, 수정 및 다른 실시예가 첨부된 청구 범위의 범위 내에 포함되도록 의도된다는 것이 이해되어야 한다. 또한, 전술한 설명 및 관련 도면은 예시적인 실시예를 요소 및/또는 기능의 특정 예시적인 조합의 맥락에서 설명하지만, 요소 및/또는 기능의 다른 조합이 첨부된 청구 범위의 범위를 벗어나지 않는 이와는 다른 실시예에 의해 제공될 수 있음이 이해되어야 한다. 이와 관련하여, 예를 들어 위에서 명시적으로 설명된 것과는 다른 요소 및/또는 기능의 조합이 또한 첨부된 청구 범위 중 일부에서 설명될 수 있는 것으로 고려된다. 여기에서 특정 용어가 사용되지만, 이들은 한정을 목적으로 하는 것이 아니라 일반적이고 설명적인 의미로만 사용된다.

도면

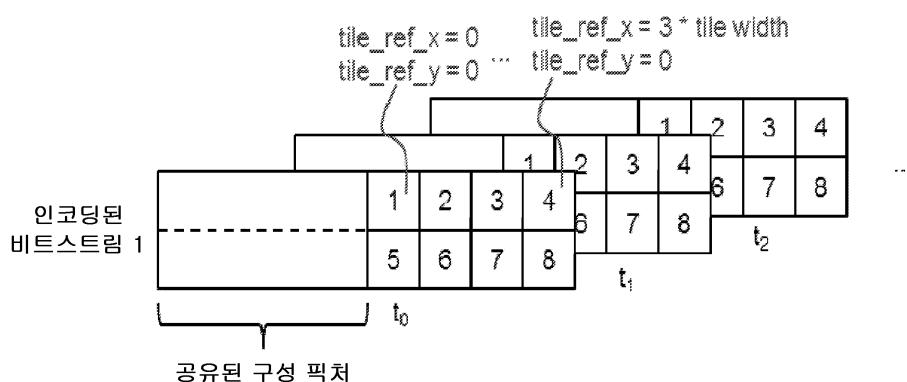
도면1



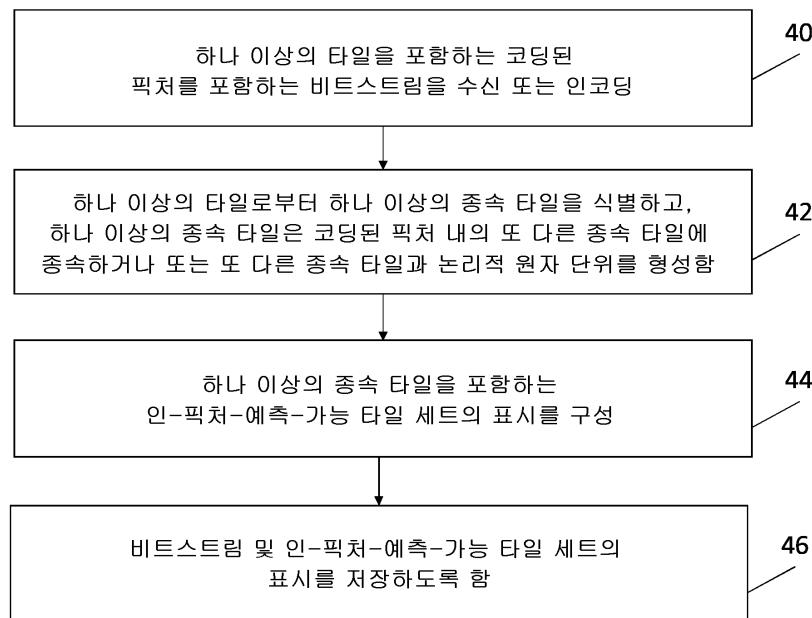
도면2



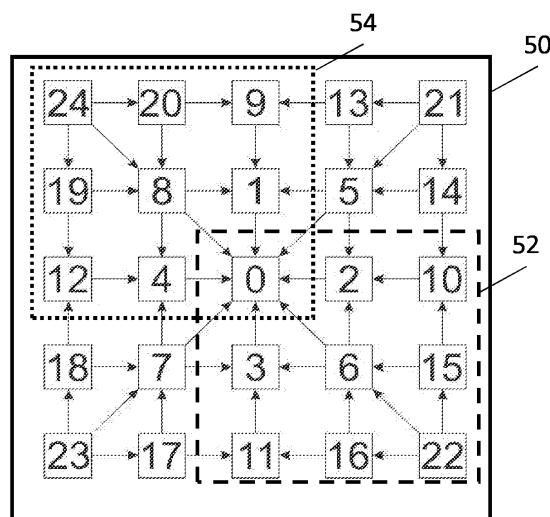
도면3



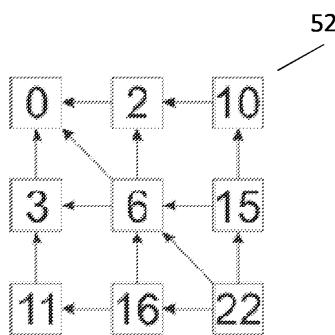
도면4



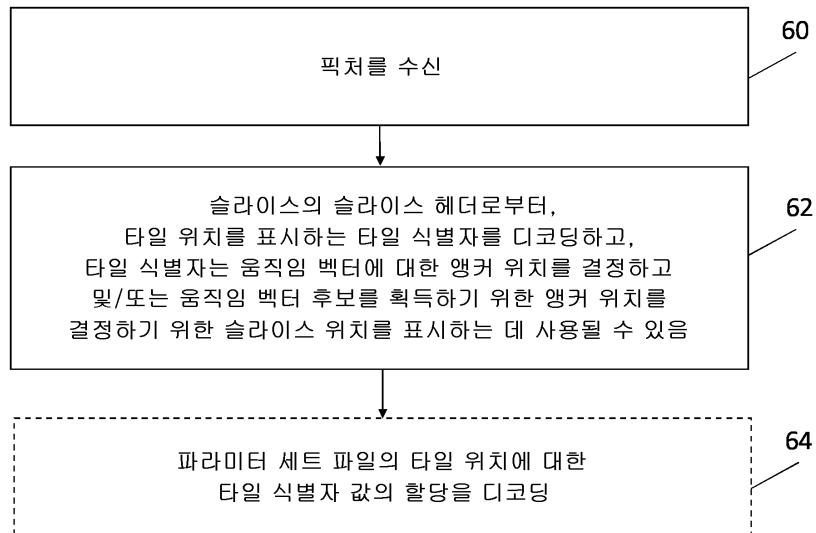
도면5a



도면5b



도면6



도면7

