



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2023-0170037  
(43) 공개일자 2023년12월18일

- |  |  |
|--|--|
| <p>(51) 국제특허분류(Int. Cl.)<br/>H04N 19/513 (2014.01) H04N 19/105 (2014.01)<br/>H04N 19/119 (2014.01) H04N 19/132 (2014.01)<br/>H04N 19/139 (2014.01) H04N 19/70 (2014.01)</p> <p>(52) CPC특허분류<br/>H04N 19/513 (2015.01)<br/>H04N 19/105 (2015.01)</p> <p>(21) 출원번호 10-2023-7038732</p> <p>(22) 출원일자(국제) 2022년04월11일<br/>심사청구일자 없음</p> <p>(85) 번역문제출일자 2023년11월09일</p> <p>(86) 국제출원번호 PCT/CN2022/086106</p> <p>(87) 국제공개번호 WO 2022/218256<br/>국제공개일자 2022년10월20일</p> <p>(30) 우선권주장<br/>63/173,540 2021년04월12일 미국(US)<br/>(뒷면에 계속)</p> | <p>(71) 출원인<br/>알리바바 (차이나) 씨오., 엘티디.<br/>중국 저지양 311121 항저우 유항 디아이에스티.<br/>웨이 시 알디. 엔오. 969 빌딩 3 플로어 5 룸 554</p> <p>(72) 발명자<br/>리아오, 루-링<br/>중국 베이징 100102 차오양 디스트릭트 왕징 이스<br/>트 파크 4티에이치 디스트릭트 엔오.6 진후이 빌<br/>딩<br/>예, 안<br/>미국 캘리포니아 92130 샌디에고 필맨 웨이 5001<br/>(뒷면에 계속)</p> <p>(74) 대리인<br/>특허법인 광장리앤코</p> |
|--|--|

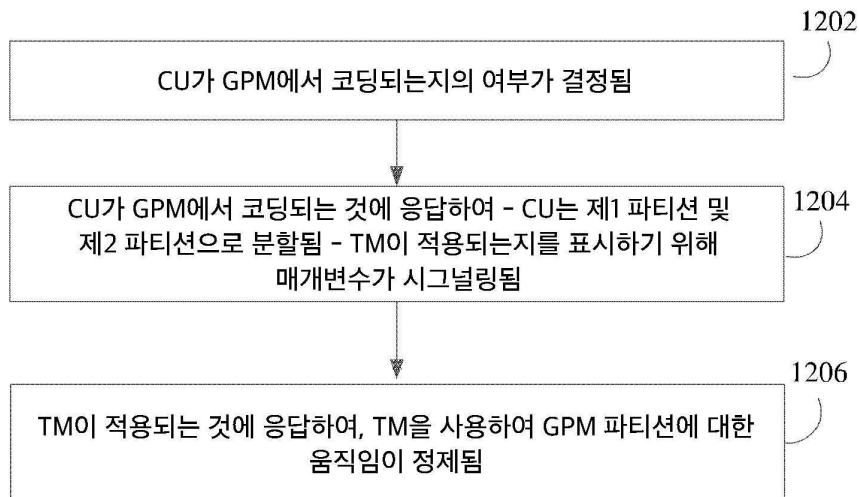
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 기하학적 파티션 모드에 대한 움직임 벡터 정제를 위한 방법, 장치, 및 비일시적 컴퓨터 판독 가능 저장 매체

(57) 요약

비디오 데이터 프로세스를 위한 방법, 장치 및 비일시적 컴퓨터 판독 가능 저장 매체가 제공된다. 방법은 기하학적 파티션 모드(GPM)에서 코딩된 코딩 유닛을 포함하는 비트스트림을 수신하는 단계; 코딩 유닛과 연관된 제1 매개변수를 디코딩하는 단계 - 제1 매개변수는 템플릿 매칭이 코딩 유닛에 적용되는지를 표시함 -; 및 제1 매개변수에 따라, 코딩 유닛에 대한 움직임 정보를 결정하는 단계 - 제1 매개변수가 템플릿 매칭이 코딩 유닛에 적용됨을 표시할 때, 템플릿 매칭을 사용하여 움직임 정보가 정제됨 - 를 포함한다.

대표도 - 도12



(52) CPC특허분류

*HOAN 19/119* (2015.01)

*HOAN 19/132* (2015.01)

*HOAN 19/139* (2015.01)

*HOAN 19/70* (2015.01)

(72) 발명자

**리, 신웨이**

중국 베이징 100102 차오양 디스트릭트 왕징 이스  
트 파크 4티에이치 디스트릭트 엔오.6 진후이 빌딩

**첸, 지에**

중국 베이징 100102 차오양 디스트릭트 왕징 이스  
트 파크 4티에이치 디스트릭트 엔오.6 진후이 빌딩

(30) 우선권주장

63/194,260 2021년05월28일 미국(US)

63/215,519 2021년06월27일 미국(US)

17/658,378 2022년04월07일 미국(US)

## 명세서

### 청구범위

#### 청구항 1

비디오 디코딩 방법으로서,

기하학적 파티션 모드(geometric partition mode, GPM)에서 코딩된 코딩 유닛을 포함하는 비트스트림을 수신하는 단계;

상기 코딩 유닛과 연관된 제1 매개변수를 디코딩하는 단계 - 상기 제1 매개변수는 템플릿 매칭(template matching)이 상기 코딩 유닛에 적용되는지를 표시함 -; 및

상기 제1 매개변수에 따라, 상기 코딩 유닛에 대한 움직임 정보를 결정하는 단계 - 상기 제1 매개변수가 상기 템플릿 매칭이 상기 코딩 유닛에 적용됨을 표시할 때, 상기 템플릿 매칭을 사용하여 상기 움직임 정보가 정제됨(refined) -

를 포함하는, 방법.

#### 청구항 2

제1항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 코딩 유닛에 대한 상기 움직임 정보는 상기 제1 파티션의 제1 움직임 및 상기 제2 파티션의 제2 움직임을 포함하고, 상기 제1 매개변수는 제1 플래그를 포함하고, 상기 템플릿 매칭을 사용하여 상기 움직임 정보가 정제되는 것은:

상기 제1 플래그에 따라 상기 템플릿 매칭을 사용하여 상기 제1 파티션의 상기 제1 움직임 및 상기 제2 파티션의 상기 제2 움직임 양쪽 모두를 정제하는 단계를 더 포함하는, 방법.

#### 청구항 3

제1항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 제1 매개변수는 상기 템플릿 매칭이 상기 제1 파티션에 적용되는지를 표시하는 제1 플래그, 및 상기 템플릿 매칭이 상기 제2 파티션에 적용되는지를 표시하는 제2 플래그를 포함하고, 상기 움직임 정보의 상기 정제는:

상기 제1 플래그에 따라 상기 템플릿 매칭을 사용하여 상기 제1 파티션의 제1 움직임을 정제하는 단계; 및

상기 제2 플래그에 따라 상기 템플릿 매칭을 사용하여 상기 제2 파티션의 제2 움직임을 정제하는 단계

를 더 포함하는, 방법.

#### 청구항 4

제1항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 제1 매개변수는 상기 템플릿 매칭이 상기 제1 파티션 및 상기 제2 파티션에 적용되는지를 표시하는 인덱스를 포함하고, 상기 움직임 정보의 상기 정제는:

상기 인덱스의 값에 따라, 상기 제1 파티션의 제1 움직임, 상기 제2 파티션의 제2 움직임, 또는 상기 제1 파티션의 상기 제1 움직임 및 상기 제2 파티션의 상기 제2 움직임 양쪽 모두를 정제하는 단계를 더 포함하는, 방법.

#### 청구항 5

제1항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 움직임 정보는 상기 제1 파티션의 제1 움직임 및 상기 제2 파티션의 제2 움직임을 포함하고, 상기 움직임 정보 정제는:

상기 제1 파티션에 대한 제1 템플릿을 구성하는 단계 - 상기 제1 템플릿은 이웃 샘플의 제1 세트로부터 구성됨 -;

상기 제2 파티션에 대한 제2 템플릿을 구성하는 단계 - 상기 제2 템플릿은 이웃 샘플의 제2 세트로부터 구성됨 -; 및

상기 제1 및 제2 템플릿을 개별적으로 사용하여 상기 제1 및 제2 움직임 정제하는 단계를 더 포함하고,

상기 이웃 샘플의 제1 및 제2 세트의 각각은:

오직 좌측 이웃 샘플만,

오직 상부 이웃 샘플만, 또는

상기 좌측 이웃 샘플 및 상기 상부 이웃 샘플 양쪽 모두

로부터 선택된 하나 이상의 이웃 샘플을 포함하는, 방법.

#### 청구항 6

제1항에 있어서,

상기 템플릿 매칭이 상기 코딩 유닛에 적용되지 않는 것에 응답하여, 움직임 벡터 차이를 갖는 병합 모드(merge mode with motion vector difference, MMVD)가 적용되는지를 표시하는 제2 매개변수를 디코딩하는 단계;

상기 MMVD가 적용되는 것에 응답하여, 움직임 벡터 차이(MVD) 정보를 디코딩하는 단계; 및

상기 MVD 정보를 사용하여 상기 움직임을 정제하는 단계를

를 더 포함하는, 방법.

#### 청구항 7

제1항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 방법은:

MMVD가 상기 제1 파티션에 적용되는지를 표시하는 제2 매개변수를 디코딩하는 단계;

상기 MMVD가 상기 제2 파티션에 적용되는지를 표시하는 제3 매개변수를 디코딩하는 단계;

상기 MMVD가 상기 제1 파티션 또는 상기 제2 파티션에 적용되지 않는 것에 응답하여, 상기 제1 매개변수에 따라, 상기 템플릿 매칭이 상기 제1 파티션 및 상기 제2 파티션 양쪽 모두에 적용되는지를 결정하는 단계; 및

상기 템플릿 매칭이 상기 제1 파티션 및 상기 제2 파티션 양쪽 모두에 적용되는 것에 응답하여, 상기 템플릿 매칭을 사용하여 상기 제1 파티션 및 상기 제2 파티션에 대한 움직임 정보를 정제하는 단계를

를 더 포함하는, 방법.

#### 청구항 8

비디오 데이터 처리를 수행하기 위한 장치로서, 상기 장치는:

명령어를 저장하도록 구성되는 메모리; 및

하나 이상의 프로세서로서, 상기 장치가:

기하학적 파티션 모드(GPM)에서 코딩된 코딩 유닛을 포함하는 비트스트림을 수신하는 것;

상기 코딩 유닛과 연관된 제1 매개변수를 디코딩하는 것 - 상기 제1 매개변수는 템플릿 매칭이 상기 코딩 유닛에 적용되는지를 표시함 -; 및

상기 제1 매개변수에 따라, 상기 코딩 유닛에 대한 움직임 정보를 결정하는 것 - 상기 제1 매개변수가 상기 템플릿 매칭이 상기 코딩 유닛에 적용됨을 표시할 때, 상기 템플릿 매칭을 사용하여 상기 움직임 정보가 정제됨 -

을 수행하게 하기 위해 상기 명령어를 실행하도록 구성되는, 상기 하나 이상의 프로세서

를 포함하는, 장치.

#### 청구항 9

제8항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 코딩 유닛에 대한 상기 움직임 정보는 상기 제1 파티션의 제1 움직임 및 상기 제2 파티션의 제2 움직임을 포함하고, 상기 제1 매개변수는

제1 플래그를 포함하고, 상기 프로세서는 상기 장치가:

상기 제1 플래그에 따라 상기 템플릿 매칭을 사용하여 상기 제1 파티션의 상기 제1 움직임 및 상기 제2 파티션의 상기 제2 움직임 양쪽 모두를 정제하는 것을 수행하게 하기 위해 상기 명령어를 실행하도록 더 구성되는, 장치.

**청구항 10**

제8항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 제1 매개변수는 상기 템플릿 매칭이 상기 제1 파티션에 적용되는지를 표시하는 제1 플래그, 및 상기 템플릿 매칭이 상기 제2 파티션에 적용되는지를 표시하는 제2 플래그를 포함하고, 상기 프로세서는 상기 장치가:

상기 제1 플래그에 따라 상기 템플릿 매칭을 사용하여 상기 제1 파티션의 제1 움직임을 정제하는 것; 및

상기 제2 플래그에 따라 상기 템플릿 매칭을 사용하여 상기 제2 파티션의 제2 움직임을 정제하는 것

을 수행하게 하기 위해 상기 명령어를 실행하도록 더 구성되는, 장치.

**청구항 11**

제8항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 제1 매개변수는 상기 템플릿 매칭이 상기 제1 파티션 및 상기 제2 파티션에 적용되는지를 표시하는 인덱스를 포함하고, 상기 프로세서는 상기 장치가:

상기 인덱스의 값에 따라, 상기 제1 파티션의 제1 움직임, 상기 제2 파티션의 제2 움직임, 또는 상기 제1 파티션의 상기 제1 움직임 및 상기 제2 파티션의 상기 제2 움직임 양쪽 모두를 정제하는 것을 수행하게 하기 위해 상기 명령어를 실행하도록 더 구성되는, 장치.

**청구항 12**

제8항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 움직임 정보는 상기 제1 파티션의 제1 움직임 및 상기 제2 파티션의 제2 움직임을 포함하고, 상기 프로세서는 상기 장치가:

상기 제1 파티션에 대한 제1 템플릿을 구성하는 것 - 상기 제1 템플릿은 이웃 샘플의 제1 세트로부터 구성됨 -;

상기 제2 파티션에 대한 제2 템플릿을 구성하는 것 - 상기 제2 템플릿은 이웃 샘플의 제2 세트로부터 구성됨 -;

및

상기 제1 및 제2 템플릿을 개별적으로 사용하여 상기 제1 및 제2 움직임을 정제하는 것을 수행하게 하기 위해 상기 명령어를 실행하도록 더 구성되고,

상기 이웃 샘플의 제1 및 제2 세트의 각각은:

오직 좌측 이웃 샘플만,

오직 상부 이웃 샘플만, 또는

상기 좌측 이웃 샘플 및 상기 상부 이웃 샘플 양쪽 모두

로부터 선택된 하나 이상의 이웃 샘플을 포함하는, 장치.

**청구항 13**

제8항에 있어서, 상기 프로세서는 상기 장치가:

상기 템플릿 매칭이 상기 코딩 유닛에 적용되지 않는 것에 응답하여, 움직임 벡터 차이를 갖는 병합 모드(MMV D)가 적용되는지를 표시하는 제2 매개변수를 디코딩하는 것;

상기 MMVD가 적용되는 것에 응답하여, 움직임 벡터 차이(MVD) 정보를 디코딩하는 것; 및

상기 MVD 정보를 사용하여 상기 움직임을 정제하는 것

을 수행하게 하기 위해 상기 명령어를 실행하도록 더 구성되는, 장치.

**청구항 14**

제8항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 프로세서는 상기 장치가:

MMVD가 상기 제1 파티션에 적용되는지를 표시하는 제2 매개변수를 디코딩하는 것;

상기 MMVD가 상기 제2 파티션에 적용되는지를 표시하는 제3 매개변수를 디코딩하는 것;

상기 MMVD가 상기 제1 파티션 또는 상기 제2 파티션에 적용되지 않는 것에 응답하여, 상기 제1 매개변수에 따라, 상기 템플릿 매칭이 상기 제1 파티션 및 상기 제2 파티션 양쪽 모두에 적용되는지를 결정하는 것; 및

상기 템플릿 매칭이 상기 제1 파티션 및 상기 제2 파티션 양쪽 모두에 적용되는 것에 응답하여, 상기 템플릿 매칭을 사용하여 상기 제1 파티션 및 상기 제2 파티션에 대한 움직임 정보를 정제하는 것

을 수행하게 하기 위해 상기 명령어를 실행하도록 더 구성되는, 장치.

**청구항 15**

장치가 비디오 데이터 처리를 수행하기 위한 방법을 개시하게 하기 위해 상기 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트를 저장하는 비일시적 컴퓨터 판독 가능 매체로서, 상기 방법은:

기하학적 파티션 모드(GPM)에서 코딩된 코딩 유닛을 포함하는 비트스트림을 수신하는 단계;

상기 코딩 유닛과 연관된 제1 매개변수를 디코딩하는 단계 - 상기 제1 매개변수는 템플릿 매칭이 상기 코딩 유닛에 적용되는지를 표시함 -; 및

상기 제1 매개변수에 따라, 상기 코딩 유닛에 대한 움직임 정보를 결정하는 단계 - 상기 제1 매개변수가 상기 템플릿 매칭이 상기 코딩 유닛에 적용됨을 표시할 때, 상기 템플릿 매칭을 사용하여 상기 움직임 정보가 정제됨 -

를 포함하는, 비일시적 컴퓨터 판독 가능 매체.

**청구항 16**

제15항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 코딩 유닛에 대한 상기 움직임 정보는 상기 제1 파티션의 제1 움직임 및 상기 제2 파티션의 제2 움직임을 포함하고, 상기 제1 매개변수는 제1 플래그를 포함하고, 장치의 하나 이상의 프로세서에 의해 실행 가능한 상기 명령어의 세트는 상기 장치가:

상기 제1 플래그에 따라 상기 템플릿 매칭을 사용하여 상기 제1 파티션의 상기 제1 움직임 및 상기 제2 파티션의 상기 제2 움직임 양쪽 모두를 정제하는 것을 더 수행하게 하는, 비일시적 컴퓨터 판독 가능 매체.

**청구항 17**

제15항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 제1 매개변수는 상기 템플릿 매칭이 상기 제1 파티션에 적용되는지를 표시하는 제1 플래그, 및 상기 템플릿 매칭이 상기 제2 파티션에 적용되는지를 표시하는 제2 플래그를 포함하고, 장치의 하나 이상의 프로세서에 의해 실행 가능한 상기 명령어의 세트는 상기 장치가:

상기 제1 플래그에 따라 상기 템플릿 매칭을 사용하여 상기 제1 파티션의 제1 움직임을 정제하는 것; 및

상기 제2 플래그에 따라 상기 템플릿 매칭을 사용하여 상기 제2 파티션의 제2 움직임을 정제하는 것

을 더 수행하게 하는, 비일시적 컴퓨터 판독 가능 매체.

**청구항 18**

제15항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 제1 매개변수는 상기 템플릿 매칭이 상기 제1 파티션 및 상기 제2 파티션에 적용되는지를 표시하는 인덱스를 포함하고, 장치의 하나 이상의 프로세서에 의해 실행 가능한 상기 명령어의 세트는 상기 장치가:

상기 인덱스의 값에 따라, 상기 제1 파티션의 제1 움직임, 상기 제2 파티션의 제2 움직임, 또는 상기 제1 파티션의 상기 제1 움직임 및 상기 제2 파티션의 상기 제2 움직임 양쪽 모두를 정제하는 것을 더 수행하게 하는, 비

일시적 컴퓨터 판독 가능 매체.

**청구항 19**

제15항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 상기 움직임 정보는 상기 제1 파티션의 제1 움직임 및 상기 제2 파티션의 제2 움직임을 포함하고, 장치의 하나 이상의 프로세서에 의해 실행 가능한 상기 명령어의 세트는 상기 장치가:

상기 제1 파티션에 대한 제1 템플릿을 구성하는 것 - 상기 제1 템플릿은 이웃 샘플의 제1 세트로부터 구성됨 -;

상기 제2 파티션에 대한 제2 템플릿을 구성하는 것 - 상기 제2 템플릿은 이웃 샘플의 제2 세트로부터 구성됨 -;

및  
상기 제1 및 제2 템플릿을 개별적으로 사용하여 상기 제1 및 제2 움직임을 정제하는 것을 더 수행하게 하고,

상기 이웃 샘플의 제1 및 제2 세트의 각각은:

오직 좌측 이웃 샘플만,

오직 상부 이웃 샘플만, 또는

상기 좌측 이웃 샘플 및 상기 상부 이웃 샘플 양쪽 모두

로부터 선택된 하나 이상의 이웃 샘플을 포함하는, 비밀시적 컴퓨터 판독 가능 매체.

**청구항 20**

제15항에 있어서, 상기 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 장치의 하나 이상의 프로세서에 의해 실행 가능한 상기 명령어의 세트는 상기 장치가:

MMVD가 상기 제1 파티션에 적용되는지를 표시하는 제2 매개변수를 디코딩하는 것;

상기 MMVD가 상기 제2 파티션에 적용되는지를 표시하는 제3 매개변수를 디코딩하는 것;

상기 MMVD가 상기 제1 파티션 또는 상기 제2 파티션에 적용되지 않는 것에 응답하여, 상기 제1 매개변수에 따라, 상기 템플릿 매칭이 상기 제1 파티션 및 상기 제2 파티션 양쪽 모두에 적용되는지를 결정하는 것; 및

상기 템플릿 매칭이 상기 제1 파티션 및 상기 제2 파티션 양쪽 모두에 적용되는 것에 응답하여, 상기 템플릿 매칭을 사용하여 상기 제1 파티션 및 상기 제2 파티션에 대한 움직임 정보를 정제하는 것

을 더 수행하게 하는, 비밀시적 컴퓨터 판독 가능 매체.

**발명의 설명**

**기술 분야**

[0001] 관련 출원에 대한 상호-참조

[0002] 본 개시는 2021년 4월 12일에 출원된 미국 가출원 제63/173,540호, 2021년 5월 29일에 출원된 미국 가출원 제 63/194,260호 및 2021년 6월 27일에 출원된 미국 가출원 제63/215,519호에 대한 우선권의 이익을 주장하며, 상기 출원 내용 모두는 그 전체가 참조로 본원에 포함된다.

[0003] 기술분야

[0004] 본 개시는 일반적으로 비디오 처리에 관한 것으로, 보다 구체적으로는 기하학적 파티션 모드에 대한 움직임 벡터 정제를 위한 방법 및 시스템에 관한 것이다.

**배경 기술**

[0005] 비디오는 시각적 정보를 캡처하는 정적 픽처(또는 "프레임")의 세트이다. 저장 메모리 및 송신 대역폭을 감소 시키기 위해, 비디오는 저장 또는 송신 이전에 압축되고, 디스플레이 이전에 압축 해제될 수 있다. 압축 프로세스는 일반적으로 인코딩으로 지칭되고, 압축 해제 프로세스는 일반적으로 디코딩으로 지칭된다. 가장 보편적

으로 예측, 변환, 양자화, 엔트로피 코딩(entropy coding) 및 인-루프 필터링(in-loop filtering)에 기초하는 표준화된 비디오 코딩 기술을 사용하는 다양한 비디오 코딩 포맷이 있다. 특정 비디오 코딩 포맷을 지정하는 고효율 비디오 코딩(High Efficiency Video Coding)(HEVC/H.265) 표준, 다용도 비디오 코딩(Versatile Video Coding)(VVC/H.266) 표준, 및 AVS 표준과 같은 비디오 코딩 표준이 표준화 기구에 의해 개발된다. 비디오 표준에 채택되는 점점 더 많은 진보된 비디오 코딩 기술과 함께 새로운 비디오 코딩 표준의 코딩 효율이 점점 더 높아지고 있다.

**발명의 내용**

- [0006] 본 개시의 실시형태는 비디오 데이터 처리를 위한 방법을 제공한다. 방법은 기하학적 파티션 모드(GPM)에서 코딩된 코딩 유닛을 포함하는 비트스트림을 수신하는 단계; 코딩 유닛과 연관된 제1 매개변수를 디코딩하는 단계 - 제1 매개변수는 템플릿 매칭이 코딩 유닛에 적용되는지를 표시함 -; 및 제1 매개변수에 따라, 코딩 유닛에 대한 움직임 정보를 결정하는 단계 - 제1 매개변수가 템플릿 매칭이 코딩 유닛에 적용됨을 표시할 때, 템플릿 매칭을 사용하여 움직임 정보가 정제됨 - 를 포함한다.
- [0007] 본 개시의 실시형태는 비디오 데이터 처리를 수행하기 위한 장치를 제공한다. 장치는 명령어를 저장하도록 구성되는 메모리; 및 하나 이상의 프로세서로서, 장치가: 기하학적 파티션 모드(GPM)에서 코딩된 코딩 유닛을 포함하는 비트스트림을 수신하는 것; 코딩 유닛과 연관된 제1 매개변수를 디코딩하는 것 - 제1 매개변수는 템플릿 매칭이 코딩 유닛에 적용되는지를 표시함 -; 및 제1 매개변수에 따라, 코딩 유닛에 대한 움직임 정보를 결정하는 것 - 제1 매개변수가 템플릿 매칭이 코딩 유닛에 적용됨을 표시할 때, 템플릿 매칭을 사용하여 움직임 정보가 정제됨 - 을 수행하게 하기 위해 명령어를 실행하도록 구성되는, 하나 이상의 프로세서를 포함한다.
- [0008] 본 개시의 실시형태는 장치가 비디오 데이터 처리를 수행하기 위한 방법을 개시하게 하기 위해 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트를 저장하는 비일시적 컴퓨터 판독 가능 저장 매체를 제공하며, 방법은 기하학적 파티션 모드(GPM)에서 코딩된 코딩 유닛을 포함하는 비트스트림을 수신하는 단계; 코딩 유닛과 연관된 제1 매개변수를 디코딩하는 단계 - 제1 매개변수는 템플릿 매칭이 코딩 유닛에 적용되는지를 표시함 -; 및 제1 매개변수에 따라, 코딩 유닛에 대한 움직임 정보를 결정하는 단계 - 제1 매개변수가 템플릿 매칭이 코딩 유닛에 적용됨을 표시할 때, 템플릿 매칭을 사용하여 움직임 정보가 정제됨 - 를 포함한다.

**도면의 간단한 설명**

- [0009] 본 개시의 실시형태 및 다양한 측면이 다음의 상세한 설명 및 첨부 도면에 예시된다. 도면에 도시되는 다양한 특징이 축척대로 도시되는 것은 아니다.
  - 도 1은 본 개시의 일부 실시형태에 따른, 예시적인 비디오 시퀀스의 구조를 예시하는 개략도이다.
  - 도 2a는 본 개시의 실시형태와 부합하는, 하이브리드 비디오 코딩 시스템의 예시적인 인코딩 프로세스를 예시하는 개략도이다.
  - 도 2b는 본 개시의 실시형태와 부합하는, 하이브리드 비디오 코딩 시스템의 다른 예시적인 인코딩 프로세스를 예시하는 개략도이다.
  - 도 3a는 본 개시의 실시형태와 부합하는, 하이브리드 비디오 코딩 시스템의 예시적인 디코딩 프로세스를 예시하는 개략도이다.
  - 도 3b는 본 개시의 실시형태와 부합하는, 하이브리드 비디오 코딩 시스템의 다른 예시적인 디코딩 프로세스를 예시하는 개략도이다.
  - 도 4는 본 개시의 일부 실시형태에 따른, 비디오를 인코딩 또는 디코딩하기 위한 예시적인 장치의 블록도이다.
  - 도 5는 본 개시의 일부 실시형태에 따른, 동일한 각도에 의해 그룹화된 예시적인 기하학적 파티션 모드(GPM) 분할을 예시한다.
  - 도 6은 본 개시의 일부 실시형태에 따른, 기하학적 파티셔닝 모드에 대한 예시적인 단방향-예측 움직임 벡터(MV) 선택 프로세스를 예시한다.
  - 도 7은 본 개시의 일부 실시형태에 따른, 기하학적 파티셔닝 모드를 사용한 밴딩 가중치( $w_0$ )의 예시적인 생성을 예시한다.

도 8은 본 개시의 일부 실시형태에 따른, 각도 가중 예측(angular weighted prediction, AWP) 가중치 예측을 위한 예시적인 프로세스를 예시한다.

도 9는 본 개시의 일부 실시형태에 따른, AWP 모드에서 지원되는 8개의 예시적인 인트라 예측 각도를 예시한다.

도 10은 본 개시의 일부 실시형태에 따른, AWP 모드에서의 7개의 상이한 예시적인 가중치 어레이 설정을 예시한다.

도 11은 본 개시의 일부 실시형태에 따른, 초기 MV 주변의 검색 에리어(search area)에 대해 수행되는 템플릿 매칭을 예시한다.

도 12는 본 개시의 일부 실시형태에 따른, 움직임을 정제하기 위해 GPM에 템플릿 매칭을 적용하기 위한 방법의 예시적인 흐름도를 예시한다.

도 13a 내지 도 13c는 개별적으로, 본 개시의 일부 실시형태에 따른, GPM에 대한 3개의 예시적인 템플릿을 예시한다.

도 14는 본 개시의 일부 실시형태에 따른, 움직임을 정제하기 위해 GPM에 템플릿 매칭을 적용하기 위한 방법의 다른 예시적인 흐름도를 예시한다.

도 15a 및 도 15b는 본 개시의 일부 실시형태에 따른, GPM에 대한 예시적인 템플릿의 다른 변형을 예시한다.

도 16은 본 개시의 일부 실시형태에 따른, GPM 파티션 모드와 GPM 파티션 각도 사이의 예시적인 관계를 예시한다.

도 17은 본 개시의 일부 실시형태에 따른, 16×16 블록에 대한 예시적인 각도를 예시한다.

도 18a 내지 도 18t는 본 개시의 일부 실시형태에 따른, 16×16 블록에 대해 도 17에서 도시되는 각 파티션 각도에 대한 상이한 GPM 파티션 모드에 대해 각 샘플에 대한 예시적인 가중치를 예시한다.

도 19는 본 개시의 일부 실시형태에 따른, 움직임을 정제하기 위해 GPM에 템플릿 매칭을 적용하기 위한 방법의 다른 예시적인 흐름도를 예시한다.

도 20은 본 개시의 일부 실시형태에 따른, 움직임을 정제하기 위해 GPM에 템플릿 매칭을 적용하기 위한 방법의 다른 예시적인 흐름도를 예시한다.

도 21은 본 개시의 일부 실시형태에 따른, GPM, TM, 및 MMVD를 적용하기 위한 방법의 예시적인 흐름도를 예시한다.

도 22는 본 개시의 일부 실시형태에 따른, GPM, TM, 및 MMVD를 적용하기 위한 방법의 예시적인 흐름도를 예시한다.

### 발명을 실시하기 위한 구체적인 내용

[0010] 이제 예시적인 실시형태에 대한 참조가 상세히 이루어질 것이며, 이의 예가 첨부 도면에 예시된다. 다음의 설명은 달리 표시되지 않는 한, 상이한 도면에서 동일한 번호가 동일하거나 유사한 요소를 나타내는 첨부 도면을 참조한다. 예시적인 실시형태의 다음 설명에서 제시되는 구현이 본 발명과 부합하는 모든 구현을 나타내는 것은 아니다. 그 대신에, 이들은 첨부된 청구범위에서 인용되는 본 발명에 관련된 측면과 부합하는 장치 및 방법의 예시일 뿐이다. 본 개시의 특정 측면은 아래에서 보다 상세히 설명된다. 참조로 통합된 용어 및/또는 정의와 상충하는 경우, 본원에서 제공되는 용어 및 정의가 우선한다.

[0011] 2020년 7월, ITU-T 비디오 코딩 전문가 그룹(ITU-T VCEG) 및 ISO/IEC 이동 픽처 전문가 그룹(Moving Picture Expert Group)(ISO/IEC MPEG)의 합동 비디오 전문가 팀(JVET)에 의해 개발된 다중도 비디오 코딩(VVC/H.266) 표준이 최종 확정되어 국제 표준으로서 공개되었다. VVC 표준은 그의 이전 버전인 고효율 비디오 코딩(HEVC/H.265) 표준의 압축 효율을 두 배로 높이는 것을 목표로 한다. 다시 말해, VVC의 목표는 대역폭의 절반을 사용하여 HEVC/H.265와 동일한 주관적 품질을 달성하는 것이다.

[0012] 대역폭의 절반을 사용하여 HEVC/H.265와 동일한 주관적 품질을 달성하기 위해, JVET는 공동 탐색 모델(joint exploration model, JEM) 참조 소프트웨어를 사용하여 HEVC를 뛰어넘는 기술을 개발하고 있다. 코딩 기술이 JEM에 통합됨에 따라, JEM은 실질적으로 HEVC보다 높은 코딩 성능을 달성하였다.

- [0013] VVC 표준은 최근에 개발되었으며, 보다 양호한 압축 성능을 제공하는 더 많은 코딩 기술을 계속 포함하고 있다. VVC는 HEVC, H.264/AVC, MPEG2, H.263 등과 같은 최신 비디오 압축 표준에서 사용된 것과 동일한 하이브리드 비디오 코딩 시스템을 기초로 한다.
- [0014] VVC 표준이 최종 확정된 후, JVET는 VVC 표준의 코딩 성능을 추가로 향상시키기 위해 새로운 코딩 툴을 탐색하기 시작한다. 2021년 1월, 강화 압축 모델(Enhanced Compression Model, ECM)이 제안되고 VVC 표준 이상의 툴을 개발하기 위한 새로운 소프트웨어 베이스로서 사용되어 왔다.
- [0015] 비디오는 시각적 정보를 저장하기 위해 시간적 시퀀스로 배열되는 정적 픽처(또는 "프레임")의 세트이다. 비디오 캡처 디바이스(예컨대, 카메라)는 이 픽처를 시간적인 시퀀스로 캡처하고 저장하는 데 사용될 수 있고, 비디오 재생 디바이스(예컨대, 디스플레이의 기능을 갖는 텔레비전, 컴퓨터, 스마트폰, 태블릿 컴퓨터, 비디오 플레이어 또는 임의의 최종-사용자 단말기)는 이러한 픽처를 시간적인 시퀀스로 디스플레이하는 데 사용될 수 있다. 또한, 일부 애플리케이션에서, 비디오 캡처링 디바이스는 감시, 회의 또는 실시간 방송과 같이 캡처된 비디오를 비디오 재생 디바이스(예컨대, 모니터를 갖는 컴퓨터)에 실시간으로 송신할 수 있다.
- [0016] 이러한 애플리케이션에 의해 필요되는 저장 공간 및 송신 대역폭을 감소시키기 위해, 비디오는 저장 및 송신 이전에 압축되고, 디스플레이 이전에 압축 해제될 수 있다. 압축 및 압축 해제는 프로세서(예컨대, 일반 컴퓨터의 프로세서) 또는 특수화된 하드웨어에 의해 실행되는 소프트웨어에 의해 구현될 수 있다. 압축을 위한 모듈은 일반적으로 "인코더"로 지칭되고, 압축 해제를 위한 모듈은 일반적으로 "디코더"로 지칭된다. 인코더 및 디코더는 집합적으로 "코덱(codec)"으로 지칭될 수 있다. 인코더 및 디코더는 다양한 적합한 하드웨어, 소프트웨어, 또는 이들의 조합 중 임의의 것으로서 구현될 수 있다. 예컨대, 인코더 및 디코더의 하드웨어 구현은 하나 이상의 마이크로프로세서, 디지털 신호 프로세서(DSP), 애플리케이션-특정 집적 회로(application-specific integrated circuits, ASIC), 필드-프로그램머블 게이트 어레이(Field-Programmable Gate Array, FPGA), 이산 로직 또는 임의의 조합과 같은 회로를 포함할 수 있다. 인코더 및 디코더의 소프트웨어 구현은 컴퓨터 판독 가능 매체에 고정된 프로그램 코드, 컴퓨터-실행가능 명령어, 펌웨어 또는 임의의 적합한 컴퓨터 구현 알고리즘 또는 프로세스를 포함할 수 있다. 비디오 압축 및 압축 해제는 MPEG-1, MPEG-2, MPEG-4, H.26x 시리즈 등과 같은 다양한 알고리즘 또는 표준에 의해 구현될 수 있다. 일부 애플리케이션에서, 코덱은 제1 코딩 표준으로부터 비디오를 압축 해제하고 제2 코딩 표준을 사용하여 압축 해제된 비디오를 재압축할 수 있으며, 이 경우 코덱은 "트랜스코더(transcoder)"로서 지칭될 수 있다.
- [0017] 비디오 인코딩 프로세스는 픽처를 재구성하는 데 사용될 수 있는 유용한 정보를 식별 및 유지할 수 있고, 재구성에 중요하지 않은 정보를 폐기할 수 있다. 폐기되고 중요하지 않은 정보가 완전히 재구성될 수 없는 경우, 이러한 인코딩 프로세스는 "손실이 있는 것(lossy)"으로 지칭될 수 있다. 그렇지 않으면, 이는 "무손실(lossless)"로 지칭될 수 있다. 대부분의 인코딩 프로세스는 손실이 있고, 이는 필요한 저장 공간 및 송신 대역폭을 감소시키기 위한 트레이드오프(tradeoff)이다.
- [0018] 인코딩되는 픽처("현재 픽처"로 지칭됨)의 유용한 정보는 참조 픽처(예컨대, 이전에 인코딩 및 재구성된 픽처)에 대한 변경을 포함한다. 이러한 변경은 픽셀의 위치 변경, 광도 변경 또는 색상 변경을 포함할 수 있으며, 그 중 위치 변경이 주로 관련된다. 객체를 나타내는 픽셀의 그룹의 위치 변경은 참조 픽처와 현재 픽처 사이의 객체의 움직임을 반영할 수 있다.
- [0019] 다른 픽처를 참조하지 않고 코딩된 픽처(즉, 자신이 그 자신의 참조 픽처인 것)는 "I-픽처"로 지칭된다. 픽처에서 일부 또는 모든 블록(예컨대, 일반적으로 비디오 픽처의 부분을 지칭하는 블록)이 하나의 참조 픽처를 통한 인트라 예측 또는 인터 예측을 사용하여 예측되는 경우(예컨대, 단방향-예측), 픽처는 "P-픽처"로 지칭된다. 픽처에서 적어도 하나의 블록이 두 개의 참조 픽처를 통해 예측되는 경우(예컨대, 양방향-예측), 픽처는 "B-픽처"로 지칭된다.
- [0020] 도 1은 본 개시의 일부 실시형태에 따른 예시적인 비디오 시퀀스(100)의 구조를 예시한다. 비디오 시퀀스(100)는 실시간 비디오 또는 캡처되고 보관되는 비디오일 수 있다. 비디오(100)는 실제 비디오, 컴퓨터-생성된 비디오(예컨대, 컴퓨터 게임 비디오) 또는 이들의 조합(예컨대, 증강-현실 효과를 갖는 실제 비디오)일 수 있다. 비디오 시퀀스(100)는 비디오 캡처 디바이스(예컨대, 카메라), 이전에 캡처된 비디오를 포함하는 비디오 아카이브(예컨대, 저장 디바이스에 저장된 비디오 파일), 또는 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 피드 인터페이스(예컨대, 비디오 브로드캐스트 트랜시버(transceiver))로부터 입력될 수 있다.
- [0021] 도 1에 도시되는 바와 같이, 비디오 시퀀스(100)는 픽처(102, 104, 106 및 108)를 포함하는, 시간선을 따라 시

간적으로 배열된 일련의 픽처를 포함할 수 있다. 픽처(102 내지 106)는 연속적이고, 픽처(106)와 픽처(108) 사이에 더 많은 픽처가 있다. 도 1에서, 픽처(102)는 I-픽처이고, 이의 참조 픽처는 픽처(102) 그 자체이다. 픽처(104)는 P-픽처이고, 이의 참조 픽처는 화살표로 표시된 바와 같이 픽처(102)이다. 픽처(106)는 B-픽처이고, 이의 참조 픽처는 화살표로 표시된 바와 같이 픽처(104 및 108)이다. 일부 실시형태에서, 픽처(예컨대, 픽처(104))의 참조 픽처가 픽처 직전 또는 직후가 아닐 수 있다. 예컨대, 픽처(104)의 참조 픽처는 픽처(102)에 선행하는 픽처일 수 있다. 픽처(102 내지 106)의 참조 픽처는 단지 예이며, 본 개시는 도 1에 도시된 예로서 참조 픽처의 실시형태를 제한하지 않는다는 것에 유의해야 한다.

[0022] 전형적으로, 비디오 코덱은 이러한 태스크(task)의 컴퓨팅 복잡성으로 인해 전체의 픽처를 동시에 인코딩하거나 또는 디코딩하지 않는다. 오히려, 이는 픽처를 기본 세그먼트로 분할할 수 있고, 픽처를 세그먼트별로 인코딩하거나 또는 디코딩할 수 있다. 이러한 기본 세그먼트는 본 개시에서 기본 처리 유닛(basic processing units, BPU)으로 지칭된다. 예를 들어, 도 1에서의 구조(110)는 비디오 시퀀스(100)의 픽처(예컨대, 픽처(102 내지 108) 중 임의의 것)의 예시적인 구조를 도시한다. 구조(110)에서, 픽처는 4X4 기본 처리 유닛으로 분할되고, 이의 경계는 점선으로 도시된다. 일부 실시형태에서, 기본 처리 유닛은 일부 비디오 코딩 표준(예컨대, MPEG 패밀러, H.261, H.263 또는 H.264/AVC)에서 "매크로블록"으로 지칭될 수 있거나, 일부 다른 비디오 코딩 표준(예컨대, H.265/HEVC 또는 H.266/VVC)에서 "코딩 트리 유닛"("CTU")으로 지칭될 수 있다. 기본 처리 유닛은 128×128, 64×64, 32×32, 16×16, 4×8, 16×32와 같은 픽처에서의 가변적인 사이즈, 또는 픽셀의 어느 임의의 형상 및 사이즈를 가질 수 있다. 기본 처리 유닛의 사이즈 및 형상은 기본 처리 유닛에서 유지될 세부사항의 레벨 및 코딩 효율의 균형에 기초하여 픽처에 대해 선택될 수 있다.

[0023] 기본 처리 유닛은 논리적 유닛일 수 있으며, 이는 컴퓨터 메모리에 (예컨대, 비디오 프레임 버퍼에) 저장되는 상이한 유형의 비디오 데이터의 그룹을 포함할 수 있다. 예를 들어, 색상 픽처의 기본 처리 유닛은 무색의 밝기 정보를 나타내는 루마 성분(Y), 색상 정보를 나타내는 하나 이상의 크로마 성분(예컨대, Cb 및 Cr) 및 연관된 신텍스 요소를 포함할 수 있고, 여기서 루마 및 크로마 성분은 기본 처리 유닛과 동일한 사이즈를 가질 수 있다. 루마 및 크로마 성분은 일부 비디오 코딩 표준(예컨대, H.265/HEVC 또는 H.266/VVC)에서 "코딩 트리 블록"("CTB")으로 지칭될 수 있다. 기본 처리 유닛에 대해 수행되는 임의의 동작은 그의 루마 및 크로마 성분 각각에 대해 반복적으로 수행될 수 있다.

[0024] 비디오 코딩은 다수의 동작 스테이지를 가지며, 이의 예는 도 2a-2b 및 도 3a-3b에 도시된다. 각 스테이지에 대해, 기본 처리 유닛의 사이즈는 처리하기에 여전히 너무 클 수 있으며, 따라서 본 개시에서 "기본 처리 서브-유닛"으로 지칭되는 세그먼트로 더 분할될 수 있다. 일부 실시형태에서, 기본 처리 서브-유닛은 일부 비디오 코딩 표준(예컨대, MPEG 패밀러, H.261, H.263 또는 H.264/AVC)에서 "블록"으로 지칭될 수 있거나, 일부 다른 비디오 코딩 표준(예컨대, H.265/HEVC 또는 H.266/VVC)에서 "코딩 유닛"("CU")으로 지칭될 수 있다. 기본 처리 서브-유닛은 기본 처리 유닛과 동일하거나 더 작은 사이즈를 가질 수 있다. 기본 처리 유닛과 유사하게, 기본 처리 서브-유닛은 또한, 논리적 유닛이며, 이는 컴퓨터 메모리에 (예컨대, 비디오 프레임 버퍼에) 저장되는 상이한 유형의 비디오 데이터(예컨대, Y, Cb, Cr 및 연관된 신텍스 요소)의 그룹을 포함할 수 있다. 기본 처리 서브-유닛에 대해 수행되는 임의의 동작은 그의 루마 및 크로마 성분 각각에 대해 반복적으로 수행될 수 있다. 이러한 분할은 처리 요구에 의존하는 추가적인 레벨로 수행될 수 있다는 것에 유의해야 한다. 또한, 상이한 스테이지가 상이한 체계를 사용하여 기본 처리 유닛을 분할할 수 있다는 것이 유의되어야 한다.

[0025] 예를 들어, 모드 결정 스테이지(그의 일례가 도 2b에 도시됨)에서, 인코더는 기본 처리 유닛에 대해 어떤 예측 모드(예컨대, 인트라-픽처 예측 또는 인터-픽처 예측)를 사용할지를 결정할 수 있으며, 기본 처리 유닛은 이러한 결정을 하기에 너무 클 수 있다. 인코더는 기본 처리 유닛을 다수의 기본 처리 서브-유닛(예컨대, H.265/HEVC 또는 H.266/VVC에서와 같이 CU)으로 분할할 수 있으며, 각 개별적인 기본 처리 서브-유닛에 대해 예측 유형을 결정할 수 있다.

[0026] 다른 예로서, 예측 스테이지(그의 일례가 도 2a-2b에 도시됨)에서, 인코더는 기본 처리 서브-유닛(예컨대, CU)의 레벨에서 예측 동작을 수행할 수 있다. 그러나, 일부 경우에서, 기본 처리 서브-유닛은 처리하기에 여전히 너무 클 수 있다. 인코더는 기본 처리 서브-유닛을 보다 작은 세그먼트(예컨대, H.265/HEVC 또는 H.266/VVC에서 "예측 블록" 또는 "PB"로 지칭됨)로 더 분할할 수 있고, 이의 레벨에서 예측 동작이 수행될 수 있다.

[0027] 다른 예로서, 변환 스테이지(그의 일례가 도 2a-2b에 도시됨)에서, 인코더는 잔차 기본 처리 서브-유닛(residual basic processing sub-unit)(예컨대, CU)에 대한 변환 동작을 수행할 수 있다. 하지만, 일부 경우에서, 기본 처리 서브-유닛은 처리하기에 여전히 너무 클 수 있다. 인코더는 기본 처리 서브-유닛을 보다 작은

세그먼트(예컨대, H.265/HEVC 또는 H.266/VVC에서 "변환 블록" 또는 "TB"로 지칭됨)로 더 분할할 수 있고, 이의 레벨에서 변환 동작이 수행될 수 있다. 동일한 기본 처리 서브-유닛의 분할 체계가 예측 스테이지 및 변환 스테이지에서 상이할 수 있다는 것에 유의해야 한다. 예를 들어, H.265/HEVC 또는 H.266/VVC에서, 동일한 CU의 예측 블록 및 변환 블록은 상이한 사이즈 및 개수를 가질 수 있다.

[0028] 도 1의 구조(110)에서, 기본 처리 유닛(112)은 그의 경계가 점선으로 도시된, 3X3 기본 처리 서브-유닛으로 더 분할된다. 동일한 픽처의 상이한 기본 처리 유닛이 상이한 체계의 기본 처리 서브-유닛으로 분할될 수 있다.

[0029] 일부 구현에서, 비디오 인코딩 및 디코딩에 대한 병렬 처리 및 오류 복원의 능력을 제공하기 위해, 픽처는 픽처의 영역에 대해, 인코딩 또는 디코딩 프로세스가 픽처의 임의의 다른 영역으로부터의 정보에 의존하지 않을 수 있도록, 처리를 위한 영역으로 분할될 수 있다. 다시 말해, 픽처의 각 영역은 독립적으로 처리될 수 있다. 이렇게 함으로써, 코덱은 픽처의 상이한 영역을 병렬로 처리할 수 있으며, 따라서 코딩 효율을 증가시킨다. 또한, 영역의 데이터가 처리에서 훼손되거나 또는 네트워크 송신에서 분실될 때, 코덱은 훼손되거나 또는 분실된 데이터에 대한 의존(reliance) 없이, 동일한 픽처의 다른 영역을 정확하게 인코딩 또는 디코딩할 수 있으며, 따라서 오류 복원의 능력을 제공한다. 일부 비디오 코딩 표준에서, 픽처는 상이한 유형의 영역으로 분할될 수 있다. 예를 들어, H.265/HEVC 및 H.266/VVC는 두 개의 유형의 영역: "슬라이스" 및 "타일"을 제공한다. 또한, 비디오 시퀀스(100)의 상이한 픽처가, 픽처를 영역으로 분할하기 위한 상이한 파티션 체계를 가질 수 있다는 것에 유의해야 한다.

[0030] 예를 들어, 도 1에서, 구조(110)는 그의 경계가 구조(110) 내에서 실선으로 도시된 세 개의 영역(114, 116 및 118)으로 분할된다. 영역(114)은 네 개의 기본 처리 유닛을 포함한다. 영역(116 및 118) 각각은 여섯 개의 기본 처리 유닛을 포함한다. 도 1에서 구조(110)의 기본 처리 유닛, 기본 처리 서브-유닛 및 영역은 단지 예이며, 본 개시는 이의 실시형태를 제한하지 않는다는 것이 유의되어야 한다.

[0031] 도 2a는 본 개시의 실시형태와 부합하는, 예시적인 인코딩 프로세스(200A)의 개략도를 예시한다. 예를 들어, 인코딩 프로세스(200A)는 인코더에 의해 수행될 수 있다. 도 2a에 도시되는 바와 같이, 인코더는 프로세스(200A)에 따라 비디오 시퀀스(202)를 비디오 비트스트림(228)으로 인코딩할 수 있다. 도 1에서의 비디오 시퀀스(100)와 유사하게, 비디오 시퀀스(202)는 시간적인 순서로 배열된 픽처("원본 픽처"로 지칭됨)의 세트를 포함할 수 있다. 도 1에서의 구조(110)와 유사하게, 비디오 시퀀스(202)의 각 원본 픽처는 인코더에 의해 기본 처리 유닛, 기본 처리 서브-유닛 또는 처리를 위한 영역으로 분할될 수 있다. 일부 실시형태에서, 인코더는 비디오 시퀀스(202)의 각 원본 픽처에 대해 기본 처리 유닛의 레벨에서 프로세스(200A)를 수행할 수 있다. 예를 들어, 인코더는 프로세스(200A)를 반복 방식으로 수행할 수 있으며, 여기서 인코더는 프로세스(200A)의 하나의 반복으로 기본 처리 유닛을 인코딩할 수 있다. 일부 실시형태에서, 인코더는 비디오 시퀀스(202)의 각 원본 픽처의 영역(예컨대, 영역(114-118))에 대해 프로세스(200A)를 병렬로 수행할 수 있다.

[0032] 도 2a에서, 인코더는 예측 데이터(206) 및 예측된 BPU(208)를 생성하기 위해 비디오 시퀀스(202)의 원본 픽처의 기본 처리 유닛("원본 BPU"로 지칭됨)을 예측 스테이지(204)로 공급할 수 있다. 인코더는 잔차 BPU(210)를 생성하기 위해 원본 BPU로부터 예측된 BPU(208)를 감산할 수 있다. 인코더는 양자화된 변환 계수(216)를 생성하기 위해, 잔차 BPU(210)를 변환 스테이지(212) 및 양자화 스테이지(214)에 공급할 수 있다. 인코더는 비디오 비트스트림(228)을 생성하기 위해, 예측 데이터(206) 및 양자화된 변환 계수(216)를 이진 코딩 스테이지(226)에 공급할 수 있다. 구성요소(202, 204, 206, 208, 210, 212, 214, 216, 226 및 228)는 "순방향 경로(forward path)"로서 지칭될 수 있다. 프로세스(200A) 동안, 양자화 스테이지(214) 이후에, 인코더는 재구성된 잔차 BPU(222)를 생성하기 위해, 양자화된 변환 계수(216)를 역 양자화 스테이지(218) 및 역변환 스테이지(220)에 공급할 수 있다. 인코더는 예측 참조(224)를 생성하기 위해 재구성된 잔차 BPU(222)를 예측된 BPU(208)에 더할 수 있고, 상기 예측 참조(224)는 프로세스(200A)의 다음 반복을 위해 예측 스테이지(204)에서 사용된다. 프로세스(200A)의 구성요소(218, 220, 222 및 224)는 "재구성 경로"로서 지칭될 수 있다. 재구성 경로는 인코더 및 디코더 양쪽 모두가 예측을 위해 동일한 참조 데이터를 사용하는 것을 보장하도록 사용될 수 있다.

[0033] 인코더는 (순방향 경로에서) 원본 픽처의 각 원본 BPU를 인코딩하고, (재구성 경로에서) 원본 픽처의 다음 원본 BPU를 인코딩하기 위한 예측된 참조(224)를 생성하기 위해 프로세스(200A)를 반복적으로 수행할 수 있다. 원본 픽처의 모든 원본 BPU를 인코딩한 이후에, 인코더는 비디오 시퀀스(202)에서 다음 픽처를 인코딩하도록 진행할 수 있다.

[0034] 프로세스(200A)를 참조하면, 인코더는 비디오 캡처링 디바이스(예컨대, 카메라)에 의해 생성되는 비디오 시퀀스(202)를 수신할 수 있다. 본원에서 사용되는 "수신하다"라는 용어는 수신, 입력, 취득, 리트리브(retrieve),

획득, 판독, 액세스 또는 데이터를 입력하기 위한 임의의 방식에서의 임의의 액션을 지칭할 수 있다.

- [0035] 현재 반복에서의 예측 스테이지(204)에서, 인코더는 원본 BPU 및 예측 참조(224)를 수신할 수 있고, 예측 데이터(206) 및 예측된 BPU(208)를 생성하기 위해 예측 동작을 수행할 수 있다. 예측 참조(224)는 프로세스(200A)의 이전의 반복의 재구성 경로로부터 생성될 수 있다. 예측 스테이지(204)의 목적은 예측 데이터(206) 및 예측 참조(224)로부터 원본 BPU를 예측된 BPU(208)로서 재구성하기 위해 사용될 수 있는 예측 데이터(206)를 추출함으로써 정보 리던던시(information redundancy)를 감소시키는 것이다.
- [0036] 이상적으로, 예측된 BPU(208)는 원본 BPU와 동일할 수 있다. 그러나, 비-이상적 예측 및 재구성 동작으로 인해, 예측된 BPU(208)는 일반적으로 원본 BPU와는 약간 상이하다. 이러한 차이를 기록하기 위해, 예측된 BPU(208)를 생성한 이후에, 인코더는 잔차 BPU(210)를 생성하기 위해, 원본 BPU로부터 예측된 BPU(208)를 감산할 수 있다. 예를 들어, 인코더는 원본 BPU의 대응하는 픽셀의 값으로부터 예측된 BPU(208)의 픽셀의 값(예컨대, 그레이스케일(greyscale) 값 또는 RGB 값)을 감산할 수 있다. 잔차 BPU(210)의 각 픽셀은 원본 BPU 및 예측된 BPU(208)의 대응하는 픽셀 사이에서 이러한 감산의 결과로서의 잔차 값을 가질 수 있다. 원본 BPU와 비교하여, 예측 데이터(206) 및 잔차 BPU(210)는 보다 적은 수의 비트를 가질 수 있지만, 이들은 현저한 품질 저하 없이 원본 BPU를 재구성하는 데 사용될 수 있다. 따라서, 원본 BPU가 압축된다.
- [0037] 잔차 BPU(210)를 더 압축하기 위해, 변환 스테이지(212)에서, 인코더는 이를 2차원 "기본 패턴" - 각 기본 패턴은 "변환 계수"와 연관된 - 의 세트에 분해함으로써 잔차 BPU(210)의 공간 리던던시를 감소시킬 수 있다. 기본 패턴은 동일한 크기(예컨대, 잔차 BPU(210)의 크기)를 가질 수 있다. 각 기본 패턴은 잔차 BPU(210)의 변동 주파수(variation frequency)(예컨대, 밝기 변동의 주파수) 성분을 나타낼 수 있다. 기본 패턴 중 어느 것도 임의의 다른 기본 패턴의 임의의 조합(예컨대, 선형 조합)으로부터 재현(reproduced)될 수 없다. 다시 말해, 분해는 잔차 BPU(210)의 변동을 주파수 도메인으로 분해할 수 있다. 이러한 분해는 함수의 이산 푸리에 변환과 유사하며, 여기서 기본 패턴은 이산 푸리에 변환의 기본 함수(예컨대, 삼각 함수)와 유사하고 변환 계수는 기본 함수와 연관된 계수와 유사하다.
- [0038] 상이한 변환 알고리즘이 상이한 기본 패턴을 사용할 수 있다. 예를 들어, 이산 코사인 변환, 이산 사인 변환 등과 같은 다양한 변환 알고리즘이 변환 스테이지(212)에서 사용될 수 있다. 변환 스테이지(212)에서의 변환은 역으로 이루어질 수 있다(invertible). 즉, 인코더는 변환의 역동작("역변환"으로 지칭됨)에 의해 잔차 BPU(210)를 복원할 수 있다. 예를 들어, 잔차 BPU(210)의 픽셀을 복원하기 위해, 역변환은 기본 패턴의 대응하는 픽셀의 값에 각각의 연관된 계수를 곱하고, 그 결과 값을 더하여 가중합(weighted sum)을 생성하는 것일 수 있다. 비디오 코딩 표준에 대해, 인코더 및 디코더 양쪽 모두는 동일한 변환 알고리즘(따라서, 동일한 기본 패턴)을 사용할 수 있다. 따라서, 인코더는 변환 계수만을 기록할 수 있고, 이로부터 디코더는 인코더로부터 기본 패턴을 수신하지 않으면서, 잔차 BPU(210)를 재구성할 수 있다. 잔차 BPU(210)와 비교하여, 변환 계수는 보다 적은 수의 비트를 가질 수 있지만, 이들은 현저한 품질 저하 없이 잔차 BPU(210)를 재구성하는 데 사용될 수 있다. 따라서, 잔차 BPU(210)가 더 압축된다.
- [0039] 인코더는 양자화 스테이지(214)에서 변환 계수를 더 압축할 수 있다. 변환 프로세스에서, 상이한 기본 패턴이 상이한 변동 주파수(예컨대, 밝기 변동 주파수)를 나타낼 수 있다. 인간의 눈은 일반적으로 저주파수 변동을 더 잘 인식하기 때문에, 인코더는 디코딩에서 현저한 품질 저하를 초래하지 않으면서 고주파수 변동의 정보를 무시할 수 있다. 예를 들어, 양자화 스테이지(214)에서, 인코더는 각 변환 계수를 정수 값("양자화 스케일 인자"로 지칭됨)으로 나누고, 몫을 그의 가장 가까운 정수로 반올림함으로써, 양자화된 변환 계수(216)를 생성할 수 있다. 이러한 동작 이후에, 고주파수 기본 패턴의 일부 변환 계수는 0으로 변환될 수 있고, 저주파수 기본 패턴의 변환 계수는 더 작은 정수로 변환될 수 있다. 인코더는 0-값 양자화된 변환 계수(216)를 무시할 수 있으며, 이에 의해 변환 계수는 더 압축된다. 또한, 양자화 프로세스는 역으로 이루어질 수 있고, 여기서 양자화된 변환 계수(216)는 양자화의 역동작("역 양자화"로 지칭됨)에서 변환 계수로 재구성될 수 있다.
- [0040] 인코더가 반올림 동작에서 이러한 나눗셈의 나머지를 무시하기 때문에, 양자화 스테이지(214)는 손실이 있을 수 있다. 전형적으로, 양자화 스테이지(214)는 프로세스(200A)의 대부분의 정보 손실에 기여할 수 있다. 정보 손실이 크면 클수록, 양자화된 변환 계수(216)는 보다 적은 수의 비트를 필요로 할 수 있다. 상이한 레벨의 정보 손실을 획득하기 위해, 인코더는 상이한 값의 양자화 매개변수 또는 양자화 프로세스의 임의의 다른 매개변수를 사용할 수 있다.
- [0041] 이진 코딩 스테이지(226)에서, 인코더는 예를 들어, 엔트로피 코딩, 가변 길이 코딩, 산술 코딩, 허프만 코딩(Huffman coding), 컨텍스트-적응적 이진 산술 코딩(context-adaptive binary arithmetic coding) 또는 임의의

다른 무손실 또는 손실 압축 알고리즘과 같은 이진 코딩 기법을 사용하여 예측 데이터(206) 및 양자화된 변환 계수(216)를 인코딩할 수 있다. 일부 실시형태에서, 예측 데이터(206) 및 양자화된 변환 계수(216) 이외에, 인코더는 예를 들어, 예측 스테이지(204)에서 사용되는 예측 모드, 예측 동작의 매개변수, 변환 스테이지(212)에서의 변환 유형, 양자화 프로세스의 매개변수(예컨대, 양자화 매개변수), 인코더 제어 매개변수(예컨대, 비트레이트 제어 매개변수) 등과 같은 다른 정보를 이진 코딩 스테이지(226)에서 인코딩할 수 있다. 인코더는 비디오 비트스트림(228)을 생성하기 위해 이진 코딩 스테이지(226)의 출력 데이터를 사용할 수 있다. 일부 실시형태에서, 비디오 비트스트림(228)은 네트워크 송신을 위해 더 패킷화(packetized)될 수 있다.

[0042] 프로세스(200A)의 재구성 경로를 참조하면, 역 양자화 스테이지(218)에서, 인코더는 재구성된 변환 계수를 생성하기 위해 양자화된 변환 계수(216)에 역 양자화를 수행할 수 있다. 역변환 스테이지(220)에서, 인코더는 재구성된 변환 계수에 기초하여, 재구성된 잔차 BPU(222)를 생성할 수 있다. 인코더는 프로세스(200A)의 다음 반복에서 사용될 예측 참조(224)를 생성하기 위해, 재구성된 잔차 BPU(222)를 예측된 BPU(208)에 더할 수 있다.

[0043] 프로세스(200A)의 다른 변형이 비디오 시퀀스(202)를 인코딩하기 위해 사용될 수 있다는 것에 유의해야 한다. 일부 실시형태에서, 프로세스(200A)의 스테이지는 인코더에 의해 상이한 순서로 수행될 수 있다. 일부 실시형태에서, 프로세스(200A)의 하나 이상의 스테이지는 단일 스테이지로 조합될 수 있다. 일부 실시형태에서, 프로세스(200A)의 단일 스테이지는 다수의 스테이지로 분할될 수 있다. 예를 들어, 변환 스테이지(212) 및 양자화 스테이지(214)가 단일 스테이지로 조합될 수 있다. 일부 실시형태에서, 프로세스(200A)는 추가적인 스테이지를 포함할 수 있다. 일부 실시형태에서, 프로세스(200A)는 도 2a에서의 하나 이상의 스테이지를 생략할 수 있다.

[0044] 도 2b는 본 개시의 실시형태와 부합하는, 다른 예시적인 인코딩 프로세스(200B)의 개략도를 예시한다. 프로세스(200B)는 프로세스(200A)로부터 수정될 수 있다. 예를 들어, 프로세스(200B)는 하이브리드 비디오 인코딩 표준(예컨대, H.26x 시리즈)을 따르는 인코더에 의해 사용될 수 있다. 프로세스(200A)와 비교하여, 프로세스(200B)의 순방향 경로는 모드 결정 스테이지(230)를 추가적으로 포함하고, 예측 스테이지(204)를 공간 예측 스테이지(2042) 및 시간 예측 스테이지(2044)로 분할한다. 프로세스(200B)의 재구성 경로는 루프 필터 스테이지(232) 및 버퍼(234)를 추가적으로 포함한다.

[0045] 일반적으로, 예측 기법은 두 개의 유형: 공간 예측 및 시간 예측으로 카테고리화될 수 있다. 공간 예측(예컨대, 인트라-픽처 예측 또는 "인트라 예측")은 현재 BPU를 예측하기 위해, 동일한 픽처에서 하나 이상의 이미 코딩된 이웃하는 BPU로부터의 픽셀을 사용할 수 있다. 즉, 공간 예측에서의 예측 참조(224)는 이웃하는 BPU를 포함할 수 있다. 공간 예측은 픽처의 내재적인 공간 리던던시를 감소시킬 수 있다. 시간 예측(예컨대, 인터-픽처 예측 또는 "인터 예측")은 현재 BPU를 예측하기 위해 하나 이상의 이미 코딩된 픽처로부터의 영역을 사용할 수 있다. 즉, 시간 예측에서의 예측 참조(224)는 코딩된 픽처를 포함할 수 있다. 시간 예측은 픽처의 내재적인 시간 리던던시를 감소시킬 수 있다.

[0046] 프로세스(200B)를 참조하면, 순방향 경로에서, 인코더는 공간 예측 스테이지(2042) 및 시간 예측 스테이지(2044)에서 예측 동작을 수행한다. 예를 들어, 공간 예측 스테이지(2042)에서, 인코더는 인트라 예측을 수행할 수 있다. 인코딩되는 픽처의 원본 BPU에 대해, 예측 참조(224)는 동일한 픽처에서 (순방향 경로에서) 인코딩되고 (재구성 경로에서) 재구성된 하나 이상의 이웃하는 BPU를 포함할 수 있다. 인코더는 이웃하는 BPU를 외삽함으로써 예측된 BPU(208)를 생성할 수 있다. 외삽 기법은 예를 들어, 선형 외삽(linear extrapolation) 또는 내삽(interpolation), 다항식 외삽 또는 내삽 등을 포함할 수 있다. 일부 실시형태에서, 인코더는 예컨대, 예측된 BPU(208)의 각 픽셀에 대해 대응하는 픽셀의 값을 외삽함으로써, 픽셀 레벨에서 외삽을 수행할 수 있다. 외삽을 위해 사용되는 이웃하는 BPU는 (예컨대, 원본 BPU의 상부에서) 수직 방향, (예컨대, 원본 BPU의 좌측에) 수평 방향, (예컨대, 원본 BPU의 좌측-하단, 우측-하단, 좌측-상단 또는 우측-상단에) 대각선 방향 또는 사용되는 비디오 코딩 표준에서 정의되는 임의의 방향에서와 같은 다양한 방향으로부터 원본 BPU에 대해 위치될 수 있다. 인트라 예측에 대해, 예측 데이터(206)는 예를 들어, 원본 BPU에 대해 사용된 이웃하는 BPU의 위치(예컨대, 좌표), 사용된 이웃하는 BPU의 사이즈, 외삽의 매개변수, 사용된 이웃하는 BPU의 방향 등을 포함할 수 있다.

[0047] 다른 예에 대해, 시간 예측 스테이지(2044)에서, 인코더는 인터 예측을 수행할 수 있다. 현재 픽처의 원본 BPU에 대해, 예측 참조(224)는 (순방향 경로에서) 인코딩되고 (재구성된 경로에서) 재구성된 하나 이상의 픽처("참조 픽처"로 지칭됨)를 포함할 수 있다. 일부 실시형태에서, 참조 픽처는 BPU별로 인코딩되고 재구성될 수 있다. 예컨대, 인코더는 재구성된 BPU를 생성하기 위해, 재구성된 잔차 BPU(222)를 예측된 BPU(208)에 더할 수 있다. 동일한 픽처의 모든 재구성된 BPU가 생성될 때, 인코더는 재구성된 픽처를 참조 픽처로서 생성할 수 있다.

다. 인코더는 참조 픽처의 범주("검색 윈도우"로 지칭됨)에서 매칭 영역을 검색하기 위해, "움직임 추정"의 동작을 수행할 수 있다. 참조 픽처에서 검색 윈도우의 위치는 현재 픽처에서 원본 BPU의 위치에 기초하여 결정될 수 있다. 예를 들어, 검색 윈도우는 현재 픽처에서 원본 BPU와 참조 픽처에서 동일한 좌표를 갖는 위치에 중심이 맞춰질 수 있고, 미리 결정된 거리에 대해 확장될 수 있다. 인코더가 검색 윈도우에서 원본 BPU와 유사한 영역을 (예컨대, 픽셀-순환 알고리즘(pel-recursive algorithm), 블록-매칭 알고리즘 등을 사용함으로써) 식별할 때, 인코더는 이러한 영역을 매칭 영역으로서 결정할 수 있다. 매칭 영역은 원본 BPU로부터 상이한 치수(예컨대, 그보다 작거나, 이와 동일하거나, 그보다 크거나 또는 상이한 형상인)를 가질 수 있다. 참조 픽처 및 현재 픽처가 (예컨대, 도 1에 도시되는 바와 같이) 시간선에서 시간적으로 분리되기 때문에, 시간이 지남에 따라, 매칭 영역이 원본 BPU의 위치로 "이동하는" 것으로 여겨질 수 있다. 인코더는 "움직임 벡터"로서 이러한 움직임의 방향 및 거리를 기록할 수 있다. (예컨대, 도 1에서의 픽처(106)와 같이) 다수의 참조 픽처가 사용될 때, 인코더는 매칭 영역을 검색하고, 각 참조 픽처에 대해 그의 연관된 움직임 벡터를 결정할 수 있다. 일부 실시형태에서, 인코더는 각각의 매칭 참조 픽처의 매칭 영역의 픽셀값에 대해 가중치를 할당할 수 있다.

[0048] 움직임 추정은 예를 들어, 병진(translation), 회전, 주밍(zooming) 등과 같은 다양한 유형의 움직임을 식별하기 위해 사용될 수 있다. 인터 예측에 대해, 예측 데이터(206)는 예를 들어, 매칭 영역의 위치(예컨대, 좌표), 매칭 영역과 연관된 움직임 벡터, 참조 픽처의 개수, 참조 픽처와 연관된 가중치 등을 포함할 수 있다.

[0049] 예측된 BPU(208)를 생성하기 위해, 인코더는 "움직임 보상"의 동작을 수행할 수 있다. 움직임 보상은 예측 데이터(206)(예컨대, 움직임 벡터) 및 예측 참조(224)에 기초하여 예측된 BPU(208)를 재구성하는 데 사용될 수 있다. 예를 들어, 인코더는 움직임 벡터에 따라 참조 픽처의 매칭 영역을 이동시킬 수 있으며, 여기서 인코더는 현재 픽처의 원본 BPU를 예측할 수 있다. (예컨대, 도 1에서의 픽처(106)와 같이) 다수의 참조 픽처가 사용될 때, 인코더는 매칭 영역의 각각의 움직임 벡터 및 평균 픽셀 값에 따라 참조 픽처의 매칭 영역을 이동시킬 수 있다. 일부 실시형태에서, 인코더가 각각의 매칭 참조 픽처의 매칭 영역의 픽셀값에 가중치를 할당한 경우, 인코더는 이동된 매칭 영역의 픽셀값의 가중합을 더할 수 있다.

[0050] 일부 실시형태에서, 인터 예측은 단방향 또는 양방향일 수 있다. 단방향 인터 예측은 현재 픽처에 대해 동일한 시간 방향으로 하나 이상의 참조 픽처를 사용할 수 있다. 예를 들어, 도 1에서의 픽처(104)는 단방향 인터-예측된 픽처이며, 여기서 참조 픽처(예컨대, 픽처(102))가 픽처(104)에 선행한다. 양방향 인터 예측은 현재 픽처에 대해 시간 방향 양쪽 모두에서 하나 이상의 참조 픽처를 사용할 수 있다. 예를 들어, 도 1에서의 픽처(106)는 양방향 인터-예측된 픽처이며, 여기서 참조 픽처(예컨대, 픽처(104 및 108))는 픽처(104)에 대해 시간 방향 양쪽 모두에 있다.

[0051] 프로세스(200B)의 순방향 경로를 계속 참조하면, 공간 예측 스테이지(2042) 및 시간 예측 스테이지(2044) 이후에, 모드 결정 스테이지(230)에서, 인코더는 프로세스(200B)의 현재 반복에 대해 예측 모드(예컨대, 인트라 예측 또는 인터 예측 중 하나)를 선택할 수 있다. 예를 들어, 인코더는 레이트-왜곡 최적화 기법(rate-distortion optimization technique)을 수행할 수 있으며, 여기서 인코더는 후보 예측 모드의 비트레이트 및 후보 예측 모드 하에서 재구성된 참조 픽처의 왜곡에 의존하여, 비용 함수의 값을 최소화하기 위해 예측 모드를 선택할 수 있다. 선택된 예측 모드에 의존하여, 인코더는 대응하는 예측된 BPU(208) 및 예측된 데이터(206)를 생성할 수 있다.

[0052] 프로세스(200B)의 재구성 경로에서, 인트라 예측 모드가 순방향 경로에서 선택된 경우, 예측 참조(224)(예컨대, 현재 픽처에서 인코딩되고 재구성된 현재 BPU)를 생성한 이후에, 인코더는 차후 사용을 위해(예컨대, 현재 픽처의 다음 BPU의 외삽을 위해), 예측 참조(224)를 공간 예측 스테이지(2042)에 직접적으로 공급할 수 있다. 인코더는 예측 참조(224)를 루프 필터 스테이지(232)에 공급할 수 있고, 여기서 인코더는, 예측 참조(224)의 코딩 동안 도입된 왜곡(예컨대, 블로킹 아티팩트(blocking artifact))를 감소시키거나 제거하기 위해 예측 참조(224)에 루프 필터를 적용할 수 있다. 인코더는 예를 들어, 디블로킹(deblocking), 샘플 적응적 오프셋, 적응적 루프 필터 등과 같은 다양한 루프 필터 기법을 루프 필터 스테이지(232)에서 적용할 수 있다. 루프-필터링된 참조 픽처는 차후 사용을 위해(예컨대, 비디오 시퀀스(202)의 향후 픽처에 대한 인터-예측 참조 픽처로서 사용되기 위해) 버퍼(234)(또는 "디코딩된 픽처 버퍼")에 저장될 수 있다. 인코더는 시간 예측 스테이지(2044)에서 사용되도록 버퍼(234)에 하나 이상의 참조 픽처를 저장할 수 있다. 일부 실시형태에서, 인코더는 양자화된 변환 계수(216), 예측 데이터(206) 및 다른 정보와 함께, 이진 코딩 스테이지(226)에서 루프 필터의 매개변수(예컨대, 루프 필터 강도)를 인코딩할 수 있다.

[0053] 도 3a는 본 개시의 실시형태와 부합하는 예시적인 디코딩 프로세스(300A)의 개략도를 예시한다. 프로세스

(300A)는 도 2a에서의 압축 프로세스(200A)에 대응하는 압축 해제 프로세스일 수 있다. 일부 실시형태에서, 프로세스(300A)는 프로세스(200A)의 재구성 경로와 유사할 수 있다. 디코더는 프로세스(300A)에 따라 비디오 비트스트림(228)을 비디오 스트림(304)으로 디코딩할 수 있다. 비디오 스트림(304)은 비디오 시퀀스(202)와 매우 유사할 수 있다. 그러나, 압축 및 압축 해제 프로세스에서의 정보 손실(예컨대, 도 2a-2b에서의 양자화 스테이지(214))로 인해, 일반적으로 비디오 스트림(304)은 비디오 시퀀스(202)와 동일하지 않다. 도 2a-2b에서의 프로세스(200A 및 200B)와 유사하게, 디코더는 비디오 비트스트림(228)에서 인코딩된 각 픽처에 대해 기본 처리 유닛(BPU)의 레벨에서 프로세스(300A)를 수행할 수 있다. 예를 들어, 디코더는 반복적인 방식으로 프로세스(300A)를 수행할 수 있으며, 여기서 디코더는 프로세스(300A)의 한 번의 반복으로 기본 처리 유닛을 디코딩할 수 있다. 일부 실시형태에서, 디코더는 비디오 비트스트림(228)에서 인코딩된 각 픽처의 영역(예컨대, 영역(114 내지 118))에 대해 병렬로 프로세스(300A)를 수행할 수 있다.

[0054] 도 3a에서, 디코더는 인코딩된 픽처의 기본 처리 유닛("인코딩된 BPU"로 지칭됨)과 연관된 비디오 비트스트림(228)의 부분을 이진 디코딩 스테이지(302)에 공급할 수 있다. 이진 디코딩 스테이지(302)에서, 디코더는 그 부분을 예측 데이터(206) 및 양자화된 변환 계수(216)로 디코딩할 수 있다. 디코더는 재구성된 잔차 BPU(222)를 생성하기 위해 양자화된 변환 계수(216)를 역 양자화 스테이지(218) 및 역변환 스테이지(220)에 공급할 수 있다. 디코더는 예측된 BPU(208)를 생성하기 위해 예측 데이터(206)를 예측 스테이지(204)에 공급할 수 있다. 디코더는 예측된 참조(224)를 생성하기 위해 재구성된 잔차 BPU(222)를 예측된 BPU(208)에 더할 수 있다. 일부 실시형태에서, 예측된 참조(224)는 버퍼(예컨대, 컴퓨터 메모리의 디코딩된 픽처 버퍼)에 저장될 수 있다. 디코더는 프로세스(300A)의 다음 반복에서 예측 동작을 수행하기 위해, 예측된 참조(224)를 예측 스테이지(204)에 공급할 수 있다.

[0055] 디코더는 인코딩된 픽처의 각 인코딩된 BPU를 디코딩하고, 인코딩된 픽처의 다음 인코딩된 BPU를 인코딩하기 위해 예측된 참조(224)를 생성하도록, 프로세스(300A)를 반복적으로 수행할 수 있다. 인코딩된 픽처의 모든 인코딩된 BPU를 디코딩한 이후에, 디코더는 디스플레이를 위해 픽처를 비디오 스트림(304)에 출력하고, 비디오 비트스트림(228)에서 다음 인코딩된 픽처를 디코딩하도록 진행할 수 있다.

[0056] 이진 디코딩 스테이지(302)에서, 디코더는 인코더에 의해 사용되는 이진 코딩 기법(예컨대, 엔트로피 코딩, 가변 길이 코딩, 산술 코딩, 허프만 코딩, 컨텍스트-적응적 이진 산술 코딩 또는 임의의 다른 무손실 압축 알고리즘)의 역동작을 수행할 수 있다. 일부 실시형태에서, 예측 데이터(206) 및 양자화된 변환 계수(216) 이외에, 디코더는 예를 들어, 예측 모드, 예측 동작의 매개변수, 변환 유형, 양자화 프로세스의 매개변수(예컨대, 양자화 매개변수), 인코더 제어 매개변수(예컨대, 비트레이트 제어 매개변수) 등과 같은 다른 정보를 이진 디코딩 스테이지(302)에서 디코딩할 수 있다. 일부 실시형태에서, 비디오 비트스트림(228)이 네트워크를 통해 패킷으로 송신되는 경우, 디코더는 비디오 비트스트림(228)을 이진 디코딩 스테이지(302)에 공급하기 전에, 이를 디패킷화(depaketize)할 수 있다.

[0057] 도 3b는 본 개시의 실시형태와 부합하는 다른 예시적인 디코딩 프로세스(300B)의 개략도를 예시한다. 프로세스(300B)는 프로세스(300A)로부터 수정될 수 있다. 예를 들어, 프로세스(300B)는 하이브리드 비디오 코딩 표준(예컨대, H.26x 시리즈)에 따르는 디코더에 의해 사용될 수 있다. 프로세스(300A)와 비교하면, 프로세스(300B)는 예측 스테이지(204)를 공간 예측 스테이지(2042) 및 시간 예측 스테이지(2044)로 추가적으로 분할하고, 루프 필터 스테이지(232) 및 버퍼(234)를 추가적으로 포함한다.

[0058] 프로세스(300B)에서, 디코딩되는 인코딩된 픽처("현재 픽처"로 지칭됨)의 인코딩된 기본 처리 유닛("현재 BPU"로 지칭됨)에 대해, 디코더에 의해 이진 디코딩 스테이지(302)로부터 디코딩된 예측 데이터(206)는, 인코더에 의해 현재 BPU를 인코딩하는 데 어느 예측 모드가 사용되었는지에 의존하여 다양한 유형의 데이터를 포함할 수 있다. 예를 들어, 현재 BPU를 인코딩하기 위해 인코더에 의해 인트라 예측이 사용된 경우, 예측 데이터(206)는 인트라 예측을 나타내는 예측 모드 표시자(예컨대, 플래그 값), 인트라 예측 동작의 매개변수 등을 포함할 수 있다. 인트라 예측 동작의 매개변수는 예를 들어, 참조로서 사용된 하나 이상의 이웃하는 BPU의 위치(예컨대, 좌표), 이웃하는 BPU의 크기, 외삽의 매개변수, 원본 BPU에 대해 이웃하는 BPU의 방향 등을 포함할 수 있다. 다른 예에 대해, 현재 BPU를 인코딩하기 위해 인코더에 의해 인터 예측이 사용된 경우, 예측 데이터(206)는 인터 예측을 나타내는 예측 모드 표시자(예컨대, 플래그 값), 인터 예측 동작의 매개변수 등을 포함할 수 있다. 인터 예측 동작의 매개변수는 예를 들어, 현재 BPU와 연관된 참조 픽처의 개수, 참조 픽처와 개별적으로 연관되는 가중치, 각각의 참조 픽처에서의 하나 이상의 매칭 영역의 위치(예컨대, 좌표), 매칭 영역과 개별적으로 연관되는 하나 이상의 움직임 벡터 등을 포함할 수 있다.

- [0059] 예측 모드 표시자에 기초하여, 디코더는 공간 예측 스테이지(2042)에서 공간 예측(예컨대, 인트라 예측)을 수행할지, 또는 시간 예측 스테이지(2044)에서 시간 예측(예컨대, 인터 예측)을 수행할지를 결정할 수 있다. 이러한 공간 예측 또는 시간 예측을 수행하는 것에 대한 세부사항이 도 2b에서 설명되며, 이하에서는 반복되지 않을 것이다. 이러한 공간 예측 또는 시간 예측을 수행한 이후에, 디코더는 예측된 BPU(208)를 생성할 수 있다. 도 3a에서 설명된 바와 같이, 디코더는 예측 참조(224)를 생성하기 위해 예측된 BPU(208) 및 재구성된 잔차 BPU(222)를 더할 수 있다.
- [0060] 프로세서(300B)에서, 디코더는 프로세서(300B)의 다음 반복에서 예측 동작을 수행하기 위해, 예측된 참조(224)를 공간 예측 스테이지(2042) 또는 시간 예측 스테이지(2044)에 공급할 수 있다. 예를 들어, 현재 BPU가 공간 예측 스테이지(2042)에서 인트라 예측을 사용하여 디코딩되는 경우, 예측 참조(224)(예컨대, 디코딩된 현재 BPU)를 생성한 이후에, 디코더는 차후 사용을 위해(예컨대, 현재 픽처의 다음 BPU의 외삽을 위해) 예측 참조(224)를 공간 예측 스테이지(2042)에 직접적으로 공급할 수 있다. 현재 BPU가 시간 예측 스테이지(2044)에서 인터 예측을 사용하여 디코딩되는 경우, 예측 참조(224)(예컨대, 모든 BPU가 디코딩된 참조 픽처)를 생성한 이후에, 디코더는 왜곡(예컨대, 블로킹 아티팩트)을 감소시키거나 제거하기 위해, 예측 참조(224)를 루프 필터 스테이지(232)에 공급할 수 있다. 도 2b에 설명된 방식으로, 디코더는 루프 필터를 예측 참조(224)에 적용할 수 있다. 루프-필터링된 참조 픽처는 차후 사용을 위해(예컨대, 비디오 비트스트림(228)의 향후 인코딩된 픽처에 대한 인터-예측 참조 픽처로서 사용되기 위해) 버퍼(234)(예컨대, 컴퓨터 메모리의 디코딩된 픽처 버퍼)에 저장될 수 있다. 디코더는 시간 예측 스테이지(2044)에서 사용되도록 버퍼(234)에 하나 이상의 참조 픽처를 저장할 수 있다. 일부 실시형태에서, 예측 데이터는 루프 필터의 매개변수(예컨대, 루프 필터 강도)를 추가로 포함할 수 있다. 일부 실시형태에서, 예측 데이터(206)의 예측 모드 표시자가 인터 예측이 현재 BPU를 인코딩하는 데 사용된 것을 나타낼 때, 예측 데이터는 루프 필터의 매개변수를 포함한다.
- [0061] 도 4는 본 개시의 실시형태와 부합하는, 비디오를 인코딩하거나 또는 디코딩하기 위한 예시적인 장치(400)의 블록도이다. 도 4에 도시되는 바와 같이, 장치(400)는 프로세서(402)를 포함할 수 있다. 프로세서(402)가 본원에서 설명되는 명령어를 실행할 때, 장치(400)는 비디오 인코딩 또는 디코딩을 위한 특수화된 기계가 될 수 있다. 프로세서(402)는 정보를 조작하거나 또는 처리할 수 있는 임의의 유형의 회로일 수 있다. 예를 들어, 프로세서(402)는 임의의 개수의 중앙 처리 유닛(또는 "CPU"), 그래픽 처리 유닛(또는 "GPU"), 신경 처리 유닛(neural processing unit, "NPU"), 마이크로컨트롤러 유닛("MCU"), 광학 프로세서, 프로그래머블 로직 컨트롤러, 마이크로컨트롤러, 마이크로프로세서, 디지털 신호 프로세서, 지적 재산권(intellectual property, IP) 코어, 프로그래머블 로직 어레이(PLA), 프로그래머블 어레이 로직(PAL), 일반 어레이 로직(Generic Array Logic, GAL), 복합 프로그래머블 논리 소자(Complex Programmable Logic Device, CPLD), 필드-프로그래머블 게이트 어레이(FPGA), 시스템 온 칩(System On Chip, SoC), 애플리케이션-특정 집적 회로(ASIC) 등의 임의의 조합을 포함할 수 있다. 일부 실시형태에서, 프로세서(402)는 또한, 단일 로직 구성요소로서 그룹화되는 프로세서의 세트일 수 있다. 예를 들어, 도 4에 도시되는 바와 같이, 프로세서(402)는 프로세서(402a), 프로세서(402b) 및 프로세서(402n)를 포함하는 다수의 프로세서를 포함할 수 있다.
- [0062] 장치(400)는 또한, 데이터(예컨대, 명령어의 세트, 컴퓨터 코드, 중간 데이터 등)를 저장하도록 구성되는 메모리(404)를 포함할 수 있다. 예를 들어, 도 4에 도시되는 바와 같이, 저장된 데이터는 프로그램 명령어(예컨대, 프로세서(200A, 200B, 300A 또는 300B)에서의 스테이지를 구현하기 위한 프로그램 명령어) 및 처리를 위한 데이터(예컨대, 비디오 시퀀스(202), 비디오 비트스트림(228) 또는 비디오 스트림(304))을 포함할 수 있다. 프로세서(402)는 프로그램 명령어 및 처리를 위한 데이터에 (예컨대, 버스(410)를 통해) 액세스할 수 있고, 처리를 위한 데이터에 대한 동작 또는 조작을 수행하기 위해 프로그램 명령어를 실행할 수 있다. 메모리(404)는 고속 랜덤-액세스 저장 디바이스 또는 비-휘발성 저장 디바이스를 포함할 수 있다. 일부 실시형태에서, 메모리(404)는 임의의 개수의 랜덤-액세스 메모리(random-access memory, RAM), 읽기-전용 메모리(ROM), 광학 디스크, 자기 디스크, 하드 드라이브, 솔리드-스테이트 드라이브(solid-state drive), 플래시 드라이브, 보안 디지털(SD) 카드, 메모리 스틱, 콤팩트 플래시(compact flash, CF) 카드 등의 임의의 조합을 포함할 수 있다. 메모리(404)는 또한, 단일 로직 구성요소로서 그룹화되는 메모리의 그룹(도 4에 도시되지 않음)일 수 있다.
- [0063] 버스(410)는 내부 버스(예컨대, CPU-메모리 버스), 외부 버스(예컨대, 범용 직렬 버스 포트(universal serial bus port), 주변 구성요소 상호연결 고속 포트(peripheral component interconnect express port)) 등과 같은, 장치(400) 내의 구성요소 사이에서 데이터를 전달하는 통신 디바이스일 수 있다.
- [0064] 모호성을 야기하지 않으면서 설명의 용이함을 위해, 프로세서(402) 및 다른 데이터 처리 회로는 본 개시에서, 집합적으로 "데이터 처리 회로"로 지칭된다. 데이터 처리 회로는 전체적으로 하드웨어로서 구현되거나, 또는

소프트웨어, 하드웨어 또는 펌웨어의 조합으로서 구현될 수 있다. 이에 더하여, 데이터 처리 회로는 단일 독립 모듈일 수 있거나, 또는 장치(400)의 임의의 다른 구성요소로 전체적으로 또는 부분적으로 조합될 수 있다.

[0065] 장치(400)는 네트워크(예컨대, 인터넷, 인트라넷, 근거리 네트워크(local area network), 모바일 통신 네트워크 등)와의 유선 통신 또는 무선 통신을 제공하기 위해 네트워크 인터페이스(406)를 더 포함할 수 있다. 일부 실시형태에서, 네트워크 인터페이스(406)는 임의의 개수의 네트워크 인터페이스 컨트롤러(NIC), 라디오 주파수(RF) 모듈, 트랜스폰더(transponder), 트랜시버, 모뎀, 라우터, 게이트웨이, 유선 네트워크 어댑터, 무선 네트워크 어댑터, 블루투스 어댑터, 적외선 어댑터, 근거리 통신(near-field communication, "NFC") 어댑터, 셀룰러 네트워크 칩(cellular network chip) 등의 임의의 조합을 포함할 수 있다.

[0066] 일부 실시형태에서, 선택적으로, 장치(400)는 하나 이상의 주변 디바이스에 대한 연결을 제공하기 위한 주변 인터페이스(408)를 더 포함할 수 있다. 도 4에 도시되는 바와 같이, 주변 디바이스는 커서 제어 디바이스(예컨대, 마우스, 터치패드 또는 터치스크린), 키보드, 디스플레이(예컨대, 음극선관 디스플레이, 액정 디스플레이 또는 발광 다이오드 디스플레이), 비디오 입력 디바이스(예컨대, 비디오 아카이브(video archive)에 커플링된(coupled) 입력 인터페이스 또는 카메라) 등을 포함할 수 있지만, 이에 제한되지 않는다.

[0067] 비디오 코덱(예컨대, 프로세스(200A, 200B, 300A 또는 300B)를 수행하는 코덱)이 장치(400)에서 임의의 소프트웨어 또는 하드웨어 모듈의 임의의 조합으로서 구현될 수 있다는 것이 유의되어야 한다. 예를 들어, 프로세스(200A, 200B, 300A 또는 300B)의 일부 또는 모든 스테이지는 메모리(404)에 로딩될 수 있는 프로그램 명령어와 같은, 장치(400)의 하나 이상의 소프트웨어 모듈로서 구현될 수 있다. 다른 예에 대해, 프로세스(200A, 200B, 300A 또는 300B)의 일부 또는 모든 스테이지는 특수화된 데이터 처리 회로(예컨대, FPGA, ASIC, NPU 등)와 같은 장치(400)의 하나 이상의 하드웨어 모듈로서 구현될 수 있다.

[0068] 본 개시는 기하학적 파티션 모드(geometric partition mode)에 대한 움직임 벡터 정제(motion vector refinement)를 수행하기 위한 방법을 제공한다.

[0069] VVC에서, 기하학적 파티셔닝 모드(GPM)가 인터 예측을 위해 지원된다. GPM은 레귤러 병합 모드(regular merge mode), 움직임 벡터 차이를 갖는 병합 모드(merge mode with motion vector difference, MMVD) 모드, 조합된 인터-인트라 예측(combined inter-intra prediction, CIIP) 모드 및 서브블록 병합 모드와 같은 다른 병합 모드와 함께, 병합 모드의 일종으로서 CU-레벨 플래그를 사용하여 시그널링된다. 총 64개의 파티션이 각각의 가능한 CU 사이즈  $w \times h = 2^m \times 2^n$ 에 대해 GPM에 의해 지원되고, 여기서  $8 \times 64$  및  $64 \times 8$ 을 제외하고  $m, n \in \{3, \dots, 6\}$ 이다.

[0070] GPM이 사용될 때, CU는 기하학적으로 위치되는 직선에 의해 2개의 파트로 분할된다. 도 5는 본 개시의 일부 실시형태에 따른, 동일한 각도에 의해 그룹화된 예시적인 기하학적 파티션 모드(GPM) 분할을 예시한다. 도 5에 도시되는 바와 같이, 분할선의 위치는 특정 파티션의 각도 및 오프셋 매개변수로부터 수학적으로 도출된다. CU에서 기하학적 파티션의 각 파트는 그 자체의 움직임을 사용하여 인터-예측되고, 오직 단방향-예측만이 각 파티션에 대해 허용된다. 즉, 각 파트는 하나의 움직임 벡터 및 하나의 참조 인덱스를 갖는다. 각 CU에 대해 오직 2개의 움직임 보상된 예측이 필요되는 것을 보장하기 위해 단방향-예측 움직임 제약이 적용되며, 이는 전통적인 양방향-예측과 동일하다. 각 파티션에 대한 단방향-예측 움직임은 아래에서 보다 상세히 설명되는 단방향-예측 후보 리스트 구성 프로세스를 사용하여 도출된다.

[0071] GPM이 현재 CU에 대해 사용되는 경우, 전체 CU에 대한 예측 신호는 다음과 같이 설명된다. 기하학적 파티션의 파티션 모드(각도 및 오프셋)를 표시하는 기하학적 파티션 인덱스가 시그널링된다. 그 후, 2개의 병합 인덱스(각 파티션에 대해 하나)가 추가로 시그널링된다. 최대 GPM 후보 사이즈의 개수는, SPS에서 명시적으로 시그널링되고 GPM 병합 인덱스에 대한 신택스 이진화(binanzation)를 지정한다. 기하학적 파티션의 파트 각각을 예측한 후에, 기하학적 파티션 예지에 따른 샘플 값이 적응형 가중치를 갖는 블렌딩 처리를 사용하여 조정되며, 이는 아래에서 보다 상세히 설명된다. 전체 CU에 적용되는 변환 및 양자화 프로세스는 다른 예측 모드에서 적용되는 프로세스와 동일할 것이다. 최종적으로, GPM을 사용하여 예측된 CU의 움직임 필드가 저장된다. GPM에 대한 움직임 필드 저장을 위한 상세한 프로세스가 아래에서 보다 상세히 설명된다.

[0072] 단방향-예측 후보 리스트 구성 프로세스에 대한 세부사항이 다음과 같이 설명된다. 단방향-예측 후보 리스트는 레귤러 병합 모드를 위해 구성된 병합 후보 리스트로부터 직접적으로 도출된다.  $n$ 을 기하학적 단방향-예측 후보 리스트에서의 단방향-예측 움직임의 인덱스로서 표시한다. LX(여기서, X는  $n$ 의 패리티(parity)와 동일함)로서 표기되는,  $n$ 번째 확장된 병합 후보의 움직임 벡터가 기하학적 파티셔닝 모드에 대한  $n$ 번째 단방향-예측 움직임

임 벡터로서 사용된다. 도 6은 본 개시의 일부 실시형태에 따른, 기하학적 파티셔닝 모드에 대한 예시적인 단 방향-예측 움직임 벡터(MV) 선택 프로세스를 예시한다. 도 6에 도시되는 바와 같이, 움직임 벡터가 "x"로 마킹된다. n번째 확장된 병합 후보의 대응하는 LX 움직임 벡터가 존재하지 않는 경우, 동일한 후보의 L(1-X) 움직임 벡터가 GPM에 대한 단방향-예측 움직임 벡터로서 대신 사용된다.

[0073] 기하학적 파티셔닝 에지를 따라 블렌딩하기 위한 프로세스에 관하여, 기하학적 파티션의 각 파트를 그 자체의 움직임을 사용하여 예측한 후에, 블렌딩이 각 파트의 2개의 예측 신호에 적용되어 기하학적 파티션 에지 주변의 샘플을 도출한다. CU의 각 위치에 대한 블렌딩 가중치가 개별 위치와 파티션 에지 사이의 거리에 기초하여 도출된다.

[0074] 도 7은 본 개시의 일부 실시형태에 따른, 기하학적 파티셔닝 모드를 사용한 블렌딩 가중치( $w_0$ )의 예시적인 생성을 예시한다. 도 7에 도시되는 바와 같이, 파티션 에지까지의 위치(x,y)에 대한 거리가 다음과 같이 도출된다.

$$d(x,y) = (2x + 1 - w) \cos(\varphi_i) + (2y + 1 - h) \sin(\varphi_i) - \rho_j \quad (1)$$

$$\rho_j = \rho_{x,j} \cos(\varphi_i) + \rho_{y,j} \sin(\varphi_i) \quad (2)$$

$$\rho_{x,j} = \begin{cases} 0 & i \% 16 = 8 \text{ or } (i \% 16 \neq 0 \text{ and } h \geq w) \\ \pm(j \times w) \gg 2 & \text{그렇지 않으면} \end{cases} \quad (3)$$

$$\rho_{y,j} = \begin{cases} \pm(j \times h) \gg 2 & i \% 16 = 8 \text{ or } (i \% 16 \neq 0 \text{ and } h \geq w) \\ 0 & \text{그렇지 않으면} \end{cases} \quad (4)$$

[0075] 여기서,  $i, j$ 는 개별적으로, 기하학적 파티션의 각도 및 오프셋에 대한 인덱스이며, 이들은 시그널링된 기하학적 파티션 인덱스에 의존한다.  $\rho_{x,j}$  및  $\rho_{y,j}$ 의 부호는 각도 인덱스  $i$ 에 의존한다.  $w$ 는 CU의 폭이고,  $h$ 는 CU의 높이이다.

[0077] 기하학적 파티션의 각 파트에 대한 가중치가 다음과 같이 도출된다.

$$wIdxL(x,y) = partIdx ? 32 + d(x,y) : 32 - d(x,y) \quad (5)$$

$$w_0(x,y) = \frac{Clip(0,8,(wIdxL(x,y)+4) \gg 3)}{8} \quad (6)$$

$$w_1(x,y) = 1 - w_0(x,y) \quad (7)$$

[0078] 여기서,  $wIdxL(x,y)$ 는 가중치를 도출하기 위한 중간 값이고,  $w_0(x,y)$  및  $w_1(x,y)$ 는 개별적으로 각 파티션에 대해 대응하는 가중치를 나타낸다.  $partIdx$ 는 각도 인덱스  $i$ 에 의존한다.

[0080] GPM에 대한 필드 저장과 관련하여, 기하학적 파티션의 제1 파트로부터의 MV1, 기하학적 파티션의 제2 파트로부터의 MV2, 및 MV1 및 MV2의 조합된 MV가 GPM 코딩된 CU의 움직임 필드에 저장된다.

[0081] 움직임 필드 내의 각 개별 위치에 대해 저장되는 움직임 벡터 유형이 다음과 같이 결정된다.

$$sType = abs(motionIdx) < 32 ? 2 : (motionIdx \leq 0 ? (1 - partIdx) : partIdx) \quad (8)$$

[0082] 여기서,  $motionIdx$ 는  $d(4x+2,4y+2)$ 와 동일하며, 이는 수학식 1로부터 재계산된다.  $partIdx$ 는 각도 인덱스  $i$ 에 의존한다.

[0084]  $sType$ 이 0 또는 1과 동일하다면, MV1 또는 MV2는 대응하는 움직임 필드에 저장되고; 그렇지 않은 경우,  $sType$ 이 2와 동일하다면, MV1 및 MV2로부터의 조합된 MV가 저장된다. 조합된 MV는 다음 프로세스를 사용하여 생성된다. (1) MV1 및 MV2가 상이한 참조 픽처 리스트(L0로부터 하나 및 L1으로부터 다른 하나)로부터 유래한다면, MV1 및 MV2는 단순히 조합되어 양방향-예측 움직임 벡터를 형성하고; (2) 그렇지 않은 경우, MV1 및 MV2가 동일한 리스트로부터 유래한다면, 오직 단방향-예측 움직임 MV만이 저장된다.

[0085] VVC의 GPM과 유사하게, 각도 가중 예측(AWP)이라 불리는 하나의 틀이 오디오 비디오 코딩 표준 3(AVS3)에서 채택된다. AVS3 비디오 표준은 2002년 중국에서 설립된 AVS Workgroup에 의해 개발된다. AVS3 표준의 이전 버전

인 AVS1 및 AVS2는 각각 2006년과 2016년에 중국 국가 표준으로서 발행되었다. AVS3에서, AWP 모드는 스킵 및 다이렉트 모드에 대해 지원된다. AWP 모드는 CU-레벨 플래그를 일종의 스킵 또는 다이렉트 모드로서 사용하여 시그널링된다. AWP 모드에서, 공간적 이웃 블록 및 시간적 움직임 벡터 예측자(predictor)로부터 움직임 벡터를 도출함으로써 5개의 상이한 단방향-예측 움직임 벡터를 포함하는 움직임 벡터 후보 리스트가 구성된다. 그 후, 움직임 벡터 후보 리스트로부터 2개의 단방향-예측 움직임 벡터가 선택되어 현재 블록을 예측한다. 모든 샘플에 대해 동일한 가중치를 갖는 양방향-예측 인터 모드와 달리, AWP 모드에서 코딩된 각 샘플은 상이한 가중치를 가질 수 있다. 각 샘플에 대한 가중치는 0 내지 8의 값을 갖는 가중치 어레이로부터 예측된다.

[0086] 도 8은 본 개시의 일부 실시형태에 따른, 각도 가중 예측(AWP) 가중치 예측을 위한 예시적인 프로세스를 예시한다. 도 9는 본 개시의 일부 실시형태에 따른, AWP 모드에서 지원되는 8개의 예시적인 인트라 예측 각도를 예시한다. 도 10은 본 개시의 일부 실시형태에 따른, AWP 모드에서의 7개의 상이한 예시적인 가중치 어레이 설정을 예시한다. 도 8에 도시되는 바와 같이, 각도 가중치 예측은 인트라 예측 모드의 프로세스와 유사하다. 총 56개의 상이한 종류의 가중치가, 8개의 인트라 예측 각도(도 9에 도시됨) 및 7개의 상이한 가중치 어레이 설정(도 10에 도시됨)을 포함하는  $m, n \in \{3 \dots 6\}$ 의 각각의 가능한 CU 사이즈  $w \times h = 2^m \times 2^n$ 에 대해 AWP 모드에 의해 지원된다. AWP 모드는 예측 없이 디코더에 직접적으로 시그널링된다는 것에 유의한다. AWP 모드 인덱스는 절삭형 이진(truncated binary)을 사용하여 이진화된다. 즉, 인덱스 0 내지 7은 5비트를 사용하여 코딩되고, 인덱스 8 내지 55는 6비트를 사용하여 코딩된다.

[0087] 두 개의 선택된 단방향-예측 움직임 벡터가 MV1 및 MV2라고 가정한다. 두 개의 예측 블록 P0 및 P1이, 개별적으로 MV1 및 MV2를 사용하여 움직임 보상을 수행함으로써 획득된다. 최종 예측 블록(P)은 다음과 같이 계산된다.

$$P = (P0 \times w0 + P1 \times (8 - w0)) \gg 3 \tag{9}$$

[0088] 여기서,  $w0$ 는 전술한 가중치 예측 방법에 의해 도출되는 가중치 행렬이다.

[0089] 여기서,  $w0$ 는 전술한 가중치 예측 방법에 의해 도출되는 가중치 행렬이다.

[0090] 예측 후에, 단방향-예측 움직임 벡터가 4×4 입도(granularity)로 저장된다. 각 4×4 유닛에 대해, 2개의 단방향-예측 움직임 벡터 중 하나가 저장된다.

[0091] 템플릿 매칭(template matching, TM)은 현재 픽처의 템플릿(예컨대, 현재 CU의 상부 및/또는 좌측 이웃 블록)과 참조 픽처의 블록(예컨대, 템플릿과 동일한 사이즈) 간의 가장 근접한 매치를 발견함으로써 현재 CU의 움직임 정보를 정제하기 위한 디코더측 MV 도출 방법이다. 도 11은 본 개시의 일부 실시형태에 따른, 초기 MV 주변의 검색 어리어에 대해 수행되는 템플릿 매칭을 예시한다. 도 11에 도시되는 바와 같이, [- 8, +8]-픽셀 검색 범위 내 현재 CU의 초기 움직임 주변에서 보다 양호한 MV가 검색된다. TM 모드는 병합 모드 및 AMVP 모드에 적용될 수 있다.

[0092] 병합 모드에 적용되는 경우, 시그널링된 병합 인덱스에 의해 표시되는 병합 후보가 초기 움직임으로서 사용된다. 움직임을 정제하기 위해 표 1에 도시되는 검색 방법이 수행된다. TM은 (AMVP이 반-픽셀(half-pel) 모드일 때 사용되는) 대안적인 보간 필터가 병합된 움직임 정보에 따라 사용되는지의 여부에 따라, 1/8-픽셀 MVD 정밀도까지 내내 수행하거나, 또는 반-픽셀 MVD 정밀도 이상(beyond)의 정밀도는 스킵할 수 있다.

[0093] [표 1] AMVR의 검색 패턴 및 AMVR과의 병합 모드

검색 패턴	AMVR 모드				병합 모드	
	4-픽셀	전-픽셀	반-픽셀	1/4-픽셀	AltIF=0	AltIF=1
4-픽셀 다이아몬드	√					
4-픽셀 교차	√					
전-픽셀 다이아몬드		√	√	√	√	√
전-픽셀 교차		√	√	√	√	√
반-픽셀 교차			√	√	√	√
1/4-픽셀 교차				√	√	
1/8-픽셀 교차					√	

[0094]

[0095]

AMVP 모드에서, 현재 블록 템플릿과 참조 블록 템플릿 간의 최소 차이에 도달하는 하나를 픽업(pick up)하기 위해 템플릿 매칭 오류에 기초하여 MVP 후보가 결정되고, 그 후, TM은 MV 정제를 위해 이 특정 MVP 후보에 대해서만 수행한다. TM은 반복적인 다이아몬드 검색을 사용함으로써 [-8, +8]-픽셀 검색 범위 내에서 전-픽셀(full-pel) MVD 정밀도(또는 4-픽셀 AMVR 모드의 경우 4-픽셀)로부터 시작하여 이 MVP 후보를 정제한다. 표 1에 명시되는 바와 같이, AMVR 모드에 의존하여 순차적으로 반-픽셀 및 1/4-픽셀 MVD 정밀도가 뒤따르는 전-픽셀 MVD 정밀도(또는 4-픽셀 AMVR 모드의 경우 4-픽셀)를 갖는 교차 검색을 사용함으로써 AMVP 후보가 추가로 정제될 수 있다. 이러한 검색 프로세스는 MVP 후보가 TM 프로세스 후에 여전히 AMVR 모드에 의해 표시된 것과 동일한 MV 정밀도를 유지하는 것을 보장한다.

[0096]

병합 후보에 대한 MVD를 시그널링하는 움직임 벡터 차이를 갖는 병합 모드(MMVD)가 VVC에 도입된다. MMVD 모드가 CU에 대해 사용되는지를 명시하기 위해 레귤러 병합 플래그를 전송한 직후에 MMVD 플래그가 시그널링된다. MMVD에서 병합 후보가 선택된 후, 이는 시그널링된 MVD 정보에 의해 추가로 정제된다. 추가 정보는 병합 후보 플래그, 움직임 크기를 명시하는 인덱스 및 움직임 방향의 표시를 위한 인덱스를 포함한다. MMVD 모드에서, 병합 리스트 내 2개의 제1 후보 중 하나가 MV 기준(basis)으로서 사용되도록 선택된다. MMVD 후보 플래그는 제1 및 제2 병합 후보 사이에서 어느 것이 사용되는지를 명시하기 위해 시그널링된다.

[0097]

거리 인덱스는 움직임 크기 정보를 명시하고 시작점으로부터 미리 규정된 오프셋을 표시한다. MMVD 모드에서, 시작 MV의 수평 성분 또는 수직 성분 중 어느 하나에 오프셋이 더해진다. 거리 인덱스와 미리 규정된 오프셋의 관계가 다음의 표 2에서 명시된다.

[0098]

[표 2] 거리 인덱스와 미리 규정된 오프셋의 관계

거리 IDX	0	1	2	3	4	5	6	7
(루마 샘플 단위의) 오프셋	1/4	1/2	1	2	4	8	16	32

[0099]

[0100]

방향 인덱스는 시작점에 대한 MVD의 방향을 나타낸다. 방향 인덱스는 다음의 표에서 도시되는 바와 같이 4개의 방향을 나타낼 수 있다. MVD 부호의 의미는 시작 MV의 정보에 따라 변동될 수 있음에 유의한다. 시작 MV(들)가 단방향-예측 MV이거나, 또는 현재 픽처의 동일한 사이드(side)를 가리키는 두 리스트 모두를 갖는 양방향-예측 MV(예컨대, 2개의 참조의 픽처 순서 카운트(picture order count, POC)가 모두 현재 픽처의 POC보다 크거나, 모두 현재 픽처의 POC보다 작음)일 때, 다음 표에서의 부호는 시작 MV에 더해지는 MV 오프셋의 부호를 명시한다. 시작 MV가 현재 픽처의 상이한 사이드를 가리키는 2개의 MV를 갖는 양방향-예측 MV(예컨대, 하나의 참조의 POC가 현재 픽처의 POC보다 크고, 다른 하나의 참조의 POC가 현재 픽처의 POC보다 작음)이고, 리스트 0에서의 POC의 차이가 리스트 1에서의 POC의 차이보다 클 때, 다음 표에서의 부호는 시작 MV의 리스트 0 MV 성분에 더해지는 MV 오프셋의 부호를 명시하고, 리스트 1 MV에 대한 부호는 반대 값을 갖는다. 그렇지 않은 경우, 리스트 1에서의 POC의 차이가 리스트 0보다 크면, 다음 표에서의 부호는 시작 MV의 리스트 1 MV 성분에 더해지는 MV 오프셋의 부호를 명시하고, 리스트 0 MV에 대한 부호는 반대 값을 갖는다.

[0101] [표 3] 방향 인덱스에 의해 명시되는 MV 오프셋의 부호

방향 IDX	00	01	10	11
x-축	+	-	N/A	N/A
y-축	N/A	N/A	+	-

[0102]

[0103] 최근에는 GPM에 MMVD가 적용되는 것이 제안된다. GPM 모드를 사용하여 CU가 코딩될 때, 각 기하학적 파티션은 그의 움직임이 시그널링된 MVD 정보에 의해 정제되는지의 여부를 자유롭게 선택할 수 있다. MMVD가 2개의 기하학적 파티션에 개별적으로 적용되는지를 표시하기 위해 2개의 추가적인 플래그가 시그널링된다. 2개의 GPM 파티션에 대한 MV 정제의 보다 유연한 조합을 가능하게 하기 위해, 다음의 조건이 2개의 GPM 파티션의 2개의 선택된 MV에 적용될 수 있다는 것에 유의한다.

[0104] a. 제1 및 제2 GPM 파티션 양쪽 모두가 MV 정제를 적용하지 않을 때, 2개의 GPM 파티션의 2개의 선택된 MV는 동일하도록 허용되지 않는다.

[0105] b. 2개의 GPM 파티션 중 하나가 MV 정제를 적용하고 다른 하나는 적용하지 않을 때, 2개의 GPM 파티션의 2개의 선택된 MV는 동일하도록 허용된다.

[0106] c. 2개의 GPM 파티션 양쪽 모두가 MV 정제를 적용할 때, 2개의 선택된 MV는 2개의 파티션의 MV 정제가 상이할 때 동일하도록 허용되고, 2개의 MV 정제가 동일할 때 동일하도록 허용되지 않는다.

[0107] TM 모드는 움직임 벡터 차이를 시그널링하지 않고 디코더 측에서의 움직임을 정제한다. 그러나, 이는 GPM이 아닌 레귤러 병합 모드에만 적용된다. 따라서, GPM은 GPM 파티션 중 어느 하나, 또는 양쪽 모두에 대해 보다 정밀한 움직임 예측을 제공할 수 있는 TM 모드로부터 이익을 얻을 수 없다.

[0108] 본 개시에서, 움직임을 정제하기 위해 GPM에 템플릿 매칭을 적용하기 위한 방법이 제안된다.

[0109] 도 12는 본 개시의 일부 실시형태에 따른, 움직임을 정제하기 위해 GPM에 템플릿 매칭을 적용하기 위한 방법(1200)의 예시적인 흐름도를 예시한다. 방법(1200)은 비디오 인코딩 프로세스(예컨대, 도 2a의 프로세스(200A) 또는 도 2b의 프로세스(200B))의 일부로서 수행될 수 있거나, 또는 장치(예컨대, 도 4의 장치(400))의 하나 이상의 소프트웨어 또는 하드웨어 구성요소에 의해 수행될 수 있다. 예를 들어, 프로세서(예컨대, 도 4의 프로세서(402))가 방법(1200)을 수행할 수 있다. 일부 실시형태에서, 방법(1200)은 컴퓨터(예컨대, 도 4의 장치(400))에 의해 실행되는 프로그램 코드와 같은 컴퓨터 실행 가능 명령어를 포함하는 컴퓨터 판독 가능 매체에 구현되는 컴퓨터 프로그램 제품에 의해 구현될 수 있다. 도 12를 참조하면, 방법(1200)은 다음의 단계(1202-1206)를 포함할 수 있다.

[0110] 단계 1202에서, CU가 GPM에서 코딩되는지의 여부가 결정된다.

[0111] 단계 1204에서, CU가 GPM에서 코딩되는 것에 응답하여 - CU는 제1 파티션 및 제2 파티션으로 분할됨 - TM이 적용되는지를 표시하기 위해 매개변수가 시그널링된다. 예를 들어, 매개변수는 TM이 전체 CU에 대해 적용되는지를 표시하기 위해 사용되는 플래그(예컨대, 플래그는 CU 레벨에서 시그널링됨)일 수 있거나, 또는 TM이 상이한 파티션에 대해 개별적으로 적용되는지를 표시하기 위해 복수의 매개변수(예컨대, 플래그)가 사용될 수 있다. 매개변수에 관한 보다 상세한 내용이 아래에서 추가로 설명될 것이다.

[0112] 단계 1206에서, TM이 적용되는 것에 응답하여, TM을 사용하여 GPM 파티션에 대한 움직임이 정제된다. 만약 TM이 CU에 적용되지 않으면, 움직임은 정제되지 않는다. 일부 실시형태에서, TM이 CU에 적용되지 않으면, 움직임은 다른 방법을 사용하여 정제될 수 있다.

[0113] 일부 실시형태에서, 보다 많은 유연함(flexibility)을 위해, 각 파티션에 대해 개별적으로 TM을 적용할지의 여부가 결정된다. 예를 들어, 코딩 유닛이 GPM에서 코딩될 때, 제1 파티션의 제1 움직임(제1 병합 인덱스에 의해 표시됨)이 TM을 사용하여 정제되는지를 표시하기 위해 제1 매개변수(예컨대, 제1 플래그)가 시그널링된다. 그 후, 제2 파티션의 제2 움직임(제2 병합 인덱스에 의해 표시됨)이 TM을 사용하여 정제되는지를 표시하기 위해 제2 매개변수(예컨대, 제2 플래그)가 시그널링된다. 하나의 예가 표 4에 도시된다.

[0114] [표 4] GPM에 템플릿 매칭을 적용하는 것의 예

제1 매개변수	제2 매개변수	
0	0	TM은 파티션 중 어느 하나에도 적용되지 않는다
0	1	TM은 제2 파티션에만 적용된다
1	0	TM은 제1 파티션에만 적용된다
1	1	TM은 파티션 양쪽 모두에 적용된다

[0115]

[0116] 제1 매개변수 및 제2 매개변수가 모두 0과 동일할 때, TM은 두 파티션에 적용되지 않는다. 제1 매개변수가 0과 동일하고 제2 매개변수가 1과 동일할 때, TM은 제2 파티션에만 적용된다. 제1 매개변수가 1과 동일하고, 제2 매개변수가 0과 동일할 때, TM은 제1 파티션에만 적용된다. 제1 매개변수 및 제2 매개변수가 모두 1과 동일할 때, TM은 파티션 양쪽 모두에 적용된다.

[0117] 제1 및 제2 매개변수는 표 5에 도시되는 바와 같이, 제3 매개변수(예컨대, 인덱스)로 조합될 수 있음에 유의한다.

[0118] [표 5] GPM에 템플릿 매칭을 적용하는 것의 예

제3 매개변수	
0	TM은 파티션 중 어느 하나에도 적용되지 않는다
1	TM은 제2 파티션에만 적용된다
2	TM은 제1 파티션에만 적용된다
3	TM은 파티션 양쪽 모두에 적용된다

[0119]

[0120] 제3 매개변수가 0과 동일할 때, TM은 두 파티션에 적용되지 않는다. 제1 파티션의 제1 움직임 또는 제2 파티션의 제2 움직임 중 어느 것도 TM을 사용하여 정제되지 않는다. 제3 매개변수가 1과 동일할 때, TM은 제2 파티션에만 적용된다. 제2 파티션의 제2 움직임이 TM을 사용하여 정제되는 한편, 제1 파티션의 제1 움직임은 정제되지 않는다. 제3 매개변수가 2와 동일할 때, TM은 제1 파티션에만 적용된다. 제1 파티션의 제1 움직임이 TM을 사용하여 정제되는 한편, 제2 파티션의 제2 움직임은 정제되지 않는다. 제3 매개변수가 3과 동일할 때, TM은 파티션 양쪽 모두에 적용된다. 제1 움직임 및 제2 움직임 양쪽 모두가 TM을 사용하여 정제된다. 일부 실시형태에서, 제3 매개변수가 1과 동일할 때, TM은 제1 파티션에만 적용된다. 제3 매개변수가 2와 동일할 때, TM은 제2 파티션에만 적용된다.

[0121] 일부 실시형태에서, GPM의 움직임을 정제할 때, 템플릿은 좌측 및/또는 상측 이웃 샘플로부터 구성된다. 도 13a 내지 도 13c는 개별적으로, 본 개시의 일부 실시형태에 따른, GPM에 대한 3개의 예시적인 템플릿을 예시한다.

[0122] 도 13a에 도시되는 바와 같이, 좌측 및 상측 이웃 샘플 양쪽 모두로부터 템플릿이 구성된다. 보다 많은 유연함을 위해, 일부 실시형태에서, 항상 상측 및 좌측 이웃 샘플 모두를 사용하는 대신에 오직 좌측 또는 상측 이웃 샘플만이 사용될 수 있다. 도 13b에 도시되는 바와 같이, 오직 상측 이웃 샘플로부터 템플릿이 구성된다. 도 13c에 도시되는 바와 같이, 오직 좌측 이웃 샘플로부터 템플릿이 구성된다. 나아가, TM이 GPM 코딩된 블록에 적용되는 경우, 3개의 템플릿 중 어느 것이 사용되는지를 표시하기 위해 복수의 매개변수가 시그널링된다. 예를 들어, 템플릿을 표시하기 위해 인덱스가 시그널링된다. 인덱스가 0과 동일한 경우, (도 13a에 도시되는 바와 같이) 상측 및 좌측 이웃 샘플이 템플릿을 구성하는 데 사용된다. 인덱스가 1과 동일한 경우, (도 13b에 도시되는 바와 같이) 상측 이웃 샘플이 사용된다. 인덱스가 2와 동일한 경우, (도 13c에 도시되는 바와 같이) 좌측 이웃 샘플이 사용된다. 일부 실시형태에서, 3개의 템플릿을 개별적으로 표시하는 3개의 매개변수가 시그널링된다.

[0123] 각 파티션이 템플릿을 개별적으로 선택할 수 있다는 것이 상정된다. 예를 들어, 제1 파티션은 템플릿으로서 상측 이웃 샘플을 선택할 수 있고, 제2 파티션은 템플릿으로서 좌측 이웃 샘플을 선택할 수 있다.

[0124] 진술한 실시형태는 임의의 적합한 방식으로 조합될 수 있다. 도 14는 본 개시의 일부 실시형태에 따른, 움직임을 정제하기 위해 GPM에 템플릿 매칭을 적용하기 위한 방법(1400)의 다른 예시적인 흐름도를 예시한다. 도 14

에 도시되는 바와 같이, 방법(1400)은 다음의 단계(1402-1410)를 포함할 수 있다.

- [0125] 단계 1402에서, CU가 GPM에서 코딩되는지의 여부가 결정된다.
- [0126] 단계 1404에서, CU가 GPM에서 코딩되는 것에 응답하여 - CU는 제1 파티션 및 제2 파티션으로 분할됨 - TM이 제1 파티션에 적용되는지를 표시하기 위해 제1 매개변수(예컨대, 제1 플래그)가 시그널링된다.
- [0127] 단계 1406에서, TM이 제2 파티션에 적용되는지를 표시하기 위해 제2 매개변수(예컨대, 제2 플래그)가 시그널링된다. 따라서, TM이 파티션에 적용되는지의 여부가 개별적으로 결정될 수 있다.
- [0128] 단계 1408에서, TM이 제1 파티션에 적용되는 것에 응답하여, 3개의 템플릿 중 어느 것이 제1 파티션의 움직임 정제하기 위해 사용되는지를 표시하기 위해 제1 인덱스가 시그널링된다. 만약 TM이 제1 파티션에 적용되지 않으면, 제1 인덱스는 시그널링되지 않는다.
- [0129] 단계 1410에서, TM이 제2 파티션에 적용되는 것에 응답하여, 3개의 템플릿 중 어느 것이 제2 파티션의 움직임을 정제하기 위해 사용되는지를 표시하기 위해 제2 인덱스가 시그널링된다. 만약 TM이 제2 파티션에 적용되지 않으면, 제2 인덱스는 시그널링되지 않는다.
- [0130] 그 후, 제1 파티션에 대한 움직임이 제1 인덱스에 의해 결정된 템플릿으로 TM을 사용하여 정제될 수 있고, 제2 파티션에 대한 움직임이 제2 인덱스에 의해 결정된 템플릿으로 TM을 사용하여 정제될 수 있다.
- [0131] 일부 실시형태에서, 어느 템플릿이 사용되는지를 시그널링하는 대신에 GPM에 기초하여 템플릿을 도출하는 것이 제안된다.
- [0132] 도 15a 및 도 15b는 본 개시의 일부 실시형태에 따른, GPM에 대한 예시적인 템플릿의 다른 변형을 예시한다. 일반적으로, 주어진 GPM 파티션에 대한 템플릿의 인덱스를 시그널링하는 대신에, 템플릿의 선택(좌측 이웃 샘플, 상부 이웃 샘플, 또는 좌측 및 상부 이웃 샘플 양쪽 모두)이 GPM의 파티션 모드에 의존할 수 있다. GPM 파티션 중 어느 하나를 일례로 취할 때, GPM 파티션이 오직 상부 이웃 샘플을 가지면 오직 상부 템플릿만이 사용되고; GPM 파티션이 오직 좌측 이웃 샘플을 가지면 오직 좌측 템플릿만이 사용되며; GPM 파티션이 상부 및 좌측 이웃 샘플 양쪽 모두를 가지면 상부 및 좌측 템플릿 양쪽 모두가 사용된다.
- [0133] 도 15a에 도시되는 바와 같이, 제1 파티션(1510A)에 대해, 오직 상부 템플릿(1511A)만이 움직임을 정제하기 위해 사용된다. 제2 파티션(1520A)에 대해, 오직 좌측 템플릿(1521A)만이 움직임을 정제하기 위해 사용된다.
- [0134] 도 15b에 도시되는 바와 같이, 제1 파티션(1510B)에 대해, 좌측 및 상부 템플릿(1511B) 양쪽 모두가 움직임을 정제하기 위해 사용된다. 제2 파티션(1520B)에 대해, 오직 상부 템플릿(1521B)만이 움직임을 정제하기 위해 사용된다.
- [0135] 일부 실시형태에서, 템플릿은 GPM 파티션 각도에 기초하여 도출된다. 도 16은 본 개시의 일부 실시형태에 따른, GPM 파티션 모드와 GPM 파티션 각도 사이의 예시적인 관계를 예시한다. 신덱스에서, GPM 파티션 모드는 `merge_gpm_partition_idx`로서 표기될 수 있고, GPM 파티션 각도는 `angleIdx`로서 표기될 수 있고, 거리는 `distanceIdx`로서 표기될 수 있다. 도 16에 도시되는 바와 같이, 20개의 각도 및 4개의 거리를 포함하는 총 64개의 파티션 모드가 있다.
- [0136] GPM의 움직임의 정제 동안, 템플릿이 먼저, 파티션 각도에 따라, 오직 좌측 이웃 샘플로부터, 오직 상부 이웃 샘플로부터, 또는 좌측 및 상부 이웃 샘플 양쪽 모두로부터 선택된다. 템플릿 선택을 위한 기본 원리는 다음과 같다. 상부 및 좌측 이웃 샘플은 파티션에 대해 결정된다. 만약 파티션에 대해 오직 상부 이웃 샘플만 이용 가능하고 좌측 이웃 샘플이 이용 가능하지 않으면, 템플릿은 오직 상부 이웃 샘플로부터 선택된다. 만약 파티션에 대해 오직 좌측 이웃 샘플만 이용 가능하고 상부 이웃 샘플이 이용 가능하지 않으면, 템플릿은 오직 좌측 이웃 샘플로부터 선택된다. 만약 좌측 및 상부 이웃 샘플 양쪽 모두가 이용 가능하다면, 템플릿은 좌측 및 상부 이웃 샘플 양쪽 모두로부터 선택된다.
- [0137] 도 17은 본 개시의 일부 실시형태에 따른, 16×16 블록에 대한 예시적인 각도를 예시한다. 도 17에 도시되는 바와 같이, 블록은 예를 들어, 상이한 숫자를 갖는 파티션 각도로서 표기되는 상이한 각도로 파티셔닝(partitioned)될 수 있다. 도 18a 내지 도 18t는 본 개시의 일부 실시형태에 따른, 16×16 블록에 대해 도 17에서 도시되는 각 파티션 각도에 대한 상이한 GPM 파티션 모드에 대해 각 샘플에 대한 예시적인 가중치를 예시한다. 도 18a 내지 도 18t에 도시되는 바와 같이, CU가 분할선에 의해 제1 파티션(1810) 및 제2 파티션(1820)으로 분할된다. 기하학적 파티션 예지(분할선)를 따르는 샘플 값은 적응형 가중치를 사용하여 조정된다. 제1

파티션(1810)은 제1 병합 인덱스에 의해 표시되는 제1 움직임을 사용하여 예측되고, 제2 파티션(1820)은 제2 병합 인덱스에 의해 표시되는 제2 움직임을 사용하여 예측된다. 제1 및 제2 병합 인덱스는, 블록이 GPM을 사용하여 코딩될 때 시그널링된다. 구체적으로, 도 18a, 18b, 18c, 18i, 18j, 18k, 18l, 18m, 18s 및 18t에 도시되는 바와 같이, 파티션 각도 0, 2, 3, 13, 14, 16, 18, 19, 29 및 30에 대해, 개별적으로, 상부 이웃 샘플이 제1 움직임을 사용하여 예측되는 제1 파티션에 대한 템플릿으로서 선택되고, 좌측 및 상부 이웃 샘플이 제2 움직임을 사용하여 예측되는 제2 파티션에 대한 템플릿으로서 선택된다. 도 18d 및 도 18n에 도시되는 바와 같이, 파티션 각도 4 및 20에 대해, 개별적으로, 상부 이웃 샘플이 제1 파티션에 대한 템플릿으로서 선택되고, 좌측 이웃 샘플이 제2 파티션에 대한 템플릿으로서 선택된다. 도 18e, 18f, 18g, 18o, 18p, 및 18q에 도시되는 바와 같이, 파티션 각도 5, 8, 11, 21, 24 및 27에 대해, 개별적으로, 좌측 및 상부 이웃 샘플 양쪽 모두가 제1 파티션에 대한 템플릿으로서 선택되고, 좌측 이웃 샘플이 제2 파티션에 대한 템플릿으로서 선택된다. 파티션 각도 12 및 28에 대해, 좌측 및 상부 이웃 샘플 양쪽 모두가 제1 및 제2 파티션에 대한 템플릿으로서 선택된다.

- [0138] GPM의 움직임을 정제할 때, 검색 패턴은 표 1에 도시되는 패턴 중 임의의 하나일 수 있다. 일부 실시형태에서, 검색 방법은 대안적인 보간 필터가 오프된 병합 모드에 대해 사용된 것과 동일할 수 있다.
- [0139] 도 19는 본 개시의 일부 실시형태에 따른, 움직임을 정제하기 위해 GPM에 템플릿 매칭을 적용하기 위한 방법(1900)의 다른 예시적인 흐름도를 예시한다. 방법(1900)은 (예컨대, 도 2a의 프로세스(200A) 또는 도 2b의 프로세스(200B)에 의한) 비디오 인코딩 프로세스로서 수행될 수 있거나, 또는 장치(예컨대, 도 4의 장치(400))의 하나 이상의 소프트웨어 또는 하드웨어 구성요소에 의해 수행될 수 있다. 예를 들어, 프로세서(예컨대, 도 4의 프로세서(402))가 방법(1900)을 수행할 수 있다. 일부 실시형태에서, 방법(1900)은 컴퓨터(예컨대, 도 4의 장치(400))에 의해 실행되는 프로그램 코드와 같은 컴퓨터 실행 가능 명령어를 포함하는 컴퓨터 판독 가능 매체에 구현되는 컴퓨터 프로그램 제품에 의해 구현될 수 있다. 도 19를 참조하면, 방법(1900)은 다음의 단계(1902-1912)를 포함할 수 있다.
- [0140] 단계 1902에서, CU가 TM 모드에서 코딩되는지의 여부가 결정된다.
- [0141] 단계 1904에서, CU가 TM 모드에서 코딩되는 것에 응답하여, CU가 2개의 파티션으로 분할되고 GPM을 사용하여 예측되는지를 표시하기 위해 플래그가 시그널링된다. 예를 들어, 플래그가 1과 동일할 때, CU는 제1 파티션 및 제2 파티션으로 분할되고, GPM이 예측을 위해 사용된다. 플래그가 0과 동일할 때, GPM은 적용되지 않고, CU는 분할되지 않는다.
- [0142] 단계 1906에서, GPM이 CU에 적용되는 것에 응답하여, 하나의 파티션 모드 및 2개의 병합 인덱스가 추가로 시그널링된다. 따라서, GPM이 적용될 때, CU는 파티션 모드에 기초하여 분할된다. 2개의 병합 인덱스는 개별적으로 제1 파티션 및 제2 파티션에 대한 2개의 움직임을 표시하기 위해 시그널링된다.
- [0143] 단계 1908에서, 2개의 병합 인덱스에 의해 표시되는 2개의 움직임이 TM을 사용하여 정제된다.
- [0144] 일부 실시형태에서, 방법(1900)은 단계 1910 및 단계 1912를 더 포함할 수 있다.
- [0145] 단계 1910에서, 정제된 움직임을 사용하여 움직임 보상이 수행된다.
- [0146] 단계 1912에서, 기하학적 파티셔닝 예지를 따른 블렌딩 프로세스가 파티션 모드에 따라 적용된다.
- [0147] 일부 실시형태에서, 단계 1910 및 단계 1912는 임의의 다른 방법, 예를 들어, 방법(1200 및 1400)에서 수행될 수 있고, 상기 방법에서, GPM에서 코딩된 CU에 대한 움직임을 정제하기 위해 TM이 사용된다.
- [0148] GPM에 TM을 적용하는 방법은 AVS3 표준에서의 AWP 모드에도 적용될 수 있음에 유의한다.
- [0149] 도 20은 본 개시의 일부 실시형태에 따른, 움직임을 정제하기 위해 GPM에 템플릿 매칭을 적용하기 위한 방법(2000)의 다른 예시적인 흐름도를 예시한다. 방법(2000)은 비디오 디코딩 프로세스(예컨대, 도 3a의 프로세스(300A) 또는 도 3b의 프로세스(300B))의 일부로서 수행될 수 있거나, 또는 장치(예컨대, 도 4의 장치(400))의 하나 이상의 소프트웨어 또는 하드웨어 구성요소에 의해 수행될 수 있다. 예를 들어, 프로세서(예컨대, 도 4의 프로세서(402))가 방법(2000)을 수행할 수 있다. 일부 실시형태에서, 방법(2000)은 컴퓨터(예컨대, 도 4의 장치(400))에 의해 실행되는 프로그램 코드와 같은 컴퓨터 실행 가능 명령어를 포함하는 컴퓨터 판독 가능 매체에 구현되는 컴퓨터 프로그램 제품에 의해 구현될 수 있다. 도 20을 참조하면, 방법(2000)은 다음의 단계(2002-2008)를 포함할 수 있다.
- [0150] 단계 2002에서, 비트스트림(예컨대, 도 3b의 비디오 비트스트림(228))은 디코더에 의해 수신되는 코딩 유닛을

포함한다.

- [0151] 단계 2004에서, CU가 GPM에서 코딩되는지의 여부가 결정된다.
- [0152] 단계 2006에서, 블록이 GPM에서 코딩되는 것에 응답하여 - CU는 제1 파티션 및 제2 파티션으로 분할됨 - TM이 적용되는지를 표시하기 위해 매개변수(예컨대, 플래그)가 디코딩된다. 일부 실시형태에서, 매개변수는 (표 4를 다시 참조하여) TM이 각 파티션에 개별적으로 적용되는지를 표시하는 복수의 플래그를 포함할 수 있다. 일부 실시형태에서, 매개변수는 (표 5를 다시 참조하여) TM이 적용되는 파티션의 조합을 표시하는 인덱스를 포함할 수 있다.
- [0153] 단계 2008에서, TM이 적용되는 것에 응답하여, TM을 사용하여 GPM 파티션에 대한 움직임이 정제된다. 만약 TM이 CU에 적용되지 않으면, 움직임은 정제되지 않는다. 일부 실시형태에서, TM이 CU에 적용되지 않으면, 움직임은 다른 방법을 사용하여 정제될 수 있다.
- [0154] 본 개시에서, GPM을 MMVD 및 TM과 조합하는 방법이 더 제공된다.
- [0155] 일부 실시형태에서, MMVD 및 TM은 동일한 CU에 적용될 수 없다. GPM을 사용하여 CU가 코딩될 때, TM이 CU에 적용되는지를 표시하기 위해 제1 매개변수(예컨대, 제1 플래그)가 시그널링된다. TM이 적용되는 경우, GPM 및 2개의 병합 인덱스가 추가로 시그널링된다. 그 후, 2개의 GPM 파티션의 움직임 양쪽 모두가 TM을 사용하여 정제된다. TM이 CU에 적용되지 않는 경우, MMVD가 GPM 파티션에 적용되는지를 표시하기 위해 제2 매개변수가 시그널링된다. MMVD가 GPM 파티션에 적용되는 경우, MVD 정보가 추가로 시그널링되고, 시그널링된 MVD 정보를 사용하여 움직임이 정제된다. 하나의 예에서, 제2 매개변수는 제2 플래그 및 제3 플래그를 포함한다. 제2 플래그는 MMVD가 제1 GPM 파티션에 적용되는지를 표시하고, 제3 플래그는 MMVD가 제2 GPM 파티션에 적용되는지를 표시한다. 다른 예에서, 제2 매개변수는 제4 플래그만을 포함하고, 제4 플래그는 MMVD가 2개의 GPM 파티션 양쪽에 적용되는지를 표시한다.
- [0156] 일부 실시형태에서, MMVD 및 TM은 동일한 GPM 파티션에 적용될 수 없다. CU가 GPM에서 코딩될 때, CU는 2개의 GPM 파티션으로 분할된다. 각 GPM 파티션에 대해, TM이 GPM 파티션에 적용되는지를 표시하기 위해 제1 매개변수가 시그널링된다. TM이 적용되는 경우, GPM 파티션의 움직임이 TM을 사용하여 정제된다. TM이 적용되지 않는 경우, MMVD가 GPM 파티션에 적용되는지를 표시하기 위해 제2 매개변수가 시그널링된다. 제1 매개변수가 TM이 GPM 파티션에 적용됨을 표시할 때, 제2 매개변수는 시그널링되지 않는 것에 유의한다. 또한, 2개의 GPM 파티션은, 움직임을 정제하기 위해 TM 또는 MMVD를 사용할지의 여부를 개별적으로 선택할 수 있다는 것에 유의한다. 즉, 하나의 GPM 파티션의 움직임은 TM을 사용하여 정제될 수 있고, 다른 하나의 GPM 파티션의 움직임은 MMVD를 사용하여 정제될 수 있다.
- [0157] 일부 실시형태에서, 2개의 매개변수(TM이 적용되는지를 표시하기 위한 하나 및 MMVD가 적용되는지를 표시하기 위한 다른 하나)의 시그널링 순서가 재순서화될 수 있다. MMVD가 적용되는지를 표시하는 매개변수가 TM이 적용되는지를 표시하는 매개변수 이전에 시그널링될 수 있다. MMVD의 여부를 표시하는 매개변수가 먼저 시그널링되고 MMVD가 GPM 파티션에 적용되는 경우, TM이 적용되는지를 표시하는 매개변수는 더 이상 시그널링되지 않고 오프되는 것으로 추론된다. 따라서, TM은 GPM 파티션에 적용되지 않는다. 도 21은 본 개시의 일부 실시형태에 따른, GPM, TM, 및 MMVD를 적용하기 위한 방법(2100)의 예시적인 흐름도를 예시한다. 방법(2100)은 비디오 인코딩 프로세스(예컨대, 도 2a의 프로세스(200A) 또는 도 2b의 프로세스(200B))의 일부로서 수행될 수 있거나, 또는 장치(예컨대, 도 4의 장치(400))의 하나 이상의 소프트웨어 또는 하드웨어 구성요소에 의해 수행될 수 있다. 예를 들어, 프로세서(예컨대, 도 4의 프로세서(402))가 방법(2100)을 수행할 수 있다. 일부 실시형태에서, 방법(2100)은 컴퓨터(예컨대, 도 4의 장치(400))에 의해 실행되는 프로그램 코드와 같은 컴퓨터 실행 가능 명령어를 포함하는 컴퓨터 판독 가능 매체에 구현되는 컴퓨터 프로그램 제품에 의해 구현될 수 있다. 도 21을 참조하면, 방법(2100)은 다음의 단계(2102-2110)를 포함할 수 있다.
- [0158] 단계 2102에서, CU가 GPM에서 코딩되는지의 여부가 결정된다.
- [0159] 단계 2104에서, CU가 GPM에서 코딩되는 것에 응답하여 - CU는 제1 파티션 및 제2 파티션으로 분할됨 - MMVD가 제1 파티션에 적용되는지를 표시하기 위해 제1 매개변수가 시그널링된다. 예를 들어, 제1 매개변수는 제1 플래그일 수 있다. 제1 플래그가 1과 동일할 때, MMVD는 제1 파티션에 적용된다. 제1 플래그가 0과 동일할 때, MMVD는 제1 파티션에 적용되지 않는다.
- [0160] 단계 2106에서, MMVD가 제2 파티션에 적용되는지를 표시하기 위해 제2 매개변수가 시그널링된다. 예를 들어, 제2 매개변수는 제2 플래그일 수 있다. 제2 플래그가 1과 동일할 때, MMVD는 제2 파티션에 적용된다. 제2 플

래그가 0과 동일할 때, MMVD는 제2 파티션에 적용되지 않는다.

- [0161] 단계 2108에서, MMVD가 제1 파티션 또는 제2 파티션 중 어느 하나에도 적용되지 않는다는 결정에 응답하여, TM이 제1 파티션 및 제2 파티션 양쪽 모두에 적용되는지를 표시하기 위해 제3 매개변수가 시그널링된다. 예를 들어, 제3 매개변수는 제3 플래그일 수 있다. 제3 플래그가 1과 동일할 때, TM은 파티션 양쪽 모두에 적용된다. 제3 플래그가 0과 동일할 때, TM은 파티션 중 어느 하나에도 적용되지 않는다.
- [0162] 단계 2110에서, TM이 제1 파티션 및 제2 파티션 양쪽 모두에 적용되는 것에 응답하여, 제1 파티션 및 제2 파티션에 대한 움직임이 TM을 사용하여 정제된다. 나아가, 일부 실시형태에서, 움직임을 정제하기 위해 사용되는 템플릿은 파티션 모드 또는 파티션 각도에 따라 결정된다.
- [0163] 도 22는 본 개시의 일부 실시형태에 따른, GPM, TM, 및 MMVD를 적용하기 위한 방법(2200)의 예시적인 흐름도를 예시한다. 방법(2100)은 비디오 디코딩 프로세스(예컨대, 도 3a의 프로세스(300A) 또는 도 3b의 프로세스(300B))의 일부로서 수행될 수 있거나, 또는 장치(예컨대, 도 4의 장치(400))의 하나 이상의 소프트웨어 또는 하드웨어 구성요소에 의해 수행될 수 있다. 예를 들어, 프로세서(예컨대, 도 4의 프로세서(402))가 방법(2200)을 수행할 수 있다. 일부 실시형태에서, 방법(2200)은 컴퓨터(예컨대, 도 4의 장치(400))에 의해 실행되는 프로그램 코드와 같은 컴퓨터 실행 가능 명령어를 포함하는 컴퓨터 판독 가능 매체에 구현되는 컴퓨터 프로그램 제품에 의해 구현될 수 있다. 도 22를 참조하면, 방법(2200)은 다음의 단계(2202-2212)를 포함할 수 있다.
- [0164] 단계 2202에서, 비트스트림(예컨대, 도 3b의 비디오 비트스트림(228))은 디코더에 의해 수신되는 코딩 유닛을 포함한다.
- [0165] 단계 2204에서, CU가 GPM에서 코딩되는지의 여부가 결정된다.
- [0166] 단계 2206에서, 블록이 GPM에서 코딩되는 것에 응답하여 - CU는 제1 파티션 및 제2 파티션으로 분할됨 - MMVD가 제1 파티션에 적용되는지를 표시하는 제1 매개변수가 디코딩된다. 예를 들어, 제1 매개변수는 제1 플래그일 수 있고, 제1 플래그가 1과 동일할 때, MMVD는 제1 파티션에 적용되고; 제1 플래그가 0과 동일할 때, MMVD는 제1 파티션에 적용되지 않는다.
- [0167] 단계 2208에서, MMVD가 제2 파티션에 적용되는지를 표시하는 제2 매개변수가 디코딩된다. 예를 들어, 제2 매개변수는 제2 플래그일 수 있고, 제2 플래그가 1과 동일할 때, MMVD는 제2 파티션에 적용되고; 제2 플래그가 0과 동일할 때, MMVD는 제2 파티션에 적용되지 않는다.
- [0168] 단계 2210에서, MMVD가 제1 파티션 또는 제2 파티션에 적용되지 않는다는 결정에 응답하여, TM이 제1 파티션 및 제2 파티션 양쪽 모두에 적용되는지를 표시하는 제3 매개변수가 디코딩된다. 예를 들어, 제3 매개변수는 제3 플래그일 수 있고, 제3 플래그가 1과 동일할 때, TM은 파티션 양쪽 모두에 적용되고; 제3 플래그가 0과 동일할 때, TM은 파티션 중 어느 하나에도 적용되지 않는다.
- [0169] 단계 2210에서, TM이 제1 파티션 및 제2 파티션 양쪽 모두에 적용되는 것에 응답하여, 제1 파티션 및 제2 파티션에 대한 움직임이 TM을 사용하여 정제된다. 나아가, 일부 실시형태에서, 움직임을 정제하기 위해 사용되는 템플릿은 파티션 모드 또는 파티션 각도에 따라 결정된다. 일부 실시형태에서, 움직임을 정제하기 위해 사용되는 템플릿이 복수의 매개변수를 디코딩함으로써 결정된다.
- [0170] 일부 실시형태에서, MMVD 및 TM이 동일한 GPM 파티션에 적용될 수 있다. 각 GPM 파티션에 대해, TM 및 MMVD가 개별적으로 적용되는지를 표시하기 위해 제1 매개변수 및 제2 매개변수가 시그널링된다. 하나의 예에서, TM 및 MMVD가 모두 GPM 파티션에 적용될 때, 움직임은 먼저 TM을 사용하여 정제된다. 그 후, 시그널링된 MVD 정보를 더함으로써 정제된 움직임이 추가로 수정된다. 다른 예에서, TM 및 MMVD가 모두 GPM 파티션에 적용될 때, 움직임이 시그널링된 MVD 정보와 먼저 더해진다. 그 후, 수정된 움직임이 TM의 시작점으로서 사용되고 TM에 의해 정제된다.
- [0171] 일부 실시형태에서, MMVD 및 TM은 오직 시그널링된 MVD 정보가 복수의 조건을 만족시킬 때만, 동일한 GPM 파티션에 적용될 수 있다. 하나의 예에서, 복수의 조건은 시그널링된 MVD 정보의 거리 인덱스가 미리 규정된 값보다 작은 것(예컨대, 1, 즉, MVD 오프셋이 1/2-픽셀 보다 작음)을 포함한다. 각 GPM 파티션에 대해, MMVD가 적용되는지를 표시하기 위해 제2 매개변수가 시그널링된다. MMVD가 적용되는 경우, 거리 인덱스 및 방향 인덱스를 포함하는 MVD 정보가 추가로 시그널링된다. 거리 인덱스가 미리 규정된 값보다 작은 경우, TM이 적용되는지를 표시하기 위해 제1 매개변수가 시그널링된다. MMVD가 적용되지 않는 경우, TM이 적용되는지를 표시하기 위해 제1 매개변수가 항상 시그널링된다. 다른 예에서, 복수의 조건은 시그널링된 MVD 정보의 거리 인덱스가 미

리 규정된 값보다 큰 것을 포함한다.

- [0172] 일부 실시형태에서, MMVD 및 TM은 오직 GPM 코딩된 CU의 사이즈/코딩 모드가 일부 조건을 만족시킬 때만, 동일한 GPM 파티션에 적용될 수 있다. 하나의 예에서, CU의 폭 및/또는 높이가 미리 규정된 임계값(예컨대, 16 또는 32)보다 큰 경우, MMVD 및 TM은 모두 동일한 GPM 파티션에 적용될 수 있다. 다른 예에서, CU의 종횡비(aspect ratio)가 미리 규정된 임계값보다 작을 때, MMVD 및 TM은 모두 동일한 GPM 파티션에 적용될 수 있다. 종횡비는  $Cu\_width > CU\_height$ 인 경우  $Cu\_width / CU\_height$ 로서, 또는  $CU\_height \geq CU\_width$ 인 경우  $CU\_height / CU\_width$ 로서 정의된다. 다른 예에서, CU가 스킵 모드가 아닌 병합 모드를 사용하여 코딩될 때, MMVD 및 TM은 모두 동일한 GPM 파티션에 적용될 수 있다. 다른 예에서, CU가 병합 모드가 아닌 스킵 모드를 사용하여 코딩될 때, MMVD 및 TM은 모두 동일한 GPM 파티션에 적용될 수 있다.
- [0173] 실시형태는 다음의 조항을 사용하여 추가로 설명될 수 있다.
- [0174] 1. 비디오 디코딩 방법으로서,
- [0175] 기하학적 파티션 모드(geometric partition mode, GPM)에서 코딩된 코딩 유닛을 포함하는 비트스트림을 수신하는 단계;
- [0176] 코딩 유닛과 연관된 제1 매개변수를 디코딩하는 단계 - 제1 매개변수는 템플릿 매칭(template matching)이 코딩 유닛에 적용되는지를 표시함 -; 및
- [0177] 제1 매개변수에 따라, 코딩 유닛에 대한 움직임 정보를 결정하는 단계 - 제1 매개변수가 템플릿 매칭이 코딩 유닛에 적용됨을 표시할 때, 템플릿 매칭을 사용하여 움직임 정보가 정제됨(refined) -
- [0178] 를 포함하는, 방법.
- [0179] 2. 조항 1에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 코딩 유닛에 대한 움직임 정보는 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임을 포함하고, 제1 매개변수는 제1 플래그를 포함하고, 템플릿 매칭을 사용하여 움직임 정보가 정제되는 것은:
- [0180] 제1 플래그에 따라 템플릿 매칭을 사용하여 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임 양쪽 모두를 정제하는 단계를 더 포함하는, 방법.
- [0181] 3. 조항 1에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 제1 매개변수는 템플릿 매칭이 제1 파티션에 적용되는지를 표시하는 제1 플래그 및 템플릿 매칭이 제2 파티션에 적용되는지를 표시하는 제2 플래그를 포함하고, 움직임 정보의 정제는:
- [0182] 제1 플래그에 따라 템플릿 매칭을 사용하여 제1 파티션의 제1 움직임을 정제하는 단계; 및
- [0183] 제2 플래그에 따라 템플릿 매칭을 사용하여 제2 파티션의 제2 움직임을 정제하는 단계
- [0184] 를 더 포함하는, 방법.
- [0185] 4. 조항 1에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 제1 매개변수는 템플릿 매칭이 제1 파티션 및 제2 파티션에 적용되는지를 표시하는 인덱스를 포함하고, 움직임 정보의 정제는:
- [0186] 인덱스의 값에 따라, 제1 파티션의 제1 움직임, 제2 파티션의 제2 움직임, 또는 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임 양쪽 모두를 정제하는 단계를 더 포함하는, 방법.
- [0187] 5. 조항 1 내지 조항 4 중 어느 한 조항에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 움직임 정보는 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임을 포함하고, 움직임 정보 정제는:
- [0188] 제1 파티션에 대한 제1 템플릿을 구성하는 단계 - 제1 템플릿은 이웃 샘플의 제1 세트로부터 구성됨 -;
- [0189] 제2 파티션에 대한 제2 템플릿을 구성하는 단계 - 제2 템플릿은 이웃 샘플의 제2 세트로부터 구성됨 -; 및
- [0190] 제1 및 제2 템플릿을 개별적으로 사용하여 제1 및 제2 움직임을 정제하는 단계
- [0191] 를 더 포함하고,
- [0192] 이웃 샘플의 제1 및 제2 세트의 각각은:
- [0193] 오직 좌측 이웃 샘플만,

- [0194] 오직 상부 이웃 샘플만, 또는
- [0195] 좌측 이웃 샘플 및 상부 이웃 샘플 양쪽 모두
- [0196] 로부터 선택된 하나 이상의 이웃 샘플을 포함하는, 방법.
- [0197] 6. 조항 5에 있어서, 이웃 샘플의 제1 세트는 제1 파티션의 좌측 이웃 샘플 및 상부 이웃 샘플의 이용 가능성(availability)에 기초하여 선택되고,
- [0198] 제2 세트 이웃 샘플은 제2 파티션의 좌측 이웃 샘플 및 상부 이웃 샘플의 이용 가능성에 기초하여 선택되는, 방법.
- [0199] 7. 조항 5 또는 조항 6에 있어서,
- [0200] GPM 파티션 모드에 기초하여 제1 템플릿 및 제2 템플릿을 구성하는 단계를 더 포함하는, 방법.
- [0201] 8. 조항 7에 있어서,
- [0202] GPM 파티션 각도에 기초하여 제1 템플릿 및 제2 템플릿을 구성하는 단계를 더 포함하는, 방법.
- [0203] 9. 조항 5 내지 조항 8 중 어느 한 조항에 있어서,
- [0204] 제1 템플릿과 연관된 제2 매개변수를 디코딩하는 단계,
- [0205] 제2 템플릿과 연관된 제3 매개변수를 디코딩하는 단계
- [0206] 를 더 포함하고,
- [0207] 제2 및 제3 매개변수는 개별적으로, 이웃 샘플의 제1 및 제2 세트가:
- [0208] 오직 좌측 이웃 샘플만,
- [0209] 오직 상부 이웃 샘플만, 또는
- [0210] 좌측 이웃 샘플 및 상부 이웃 샘플 양쪽 모두
- [0211] 로부터 선택되는지를 표시하는, 방법.
- [0212] 10. 조항 5 내지 조항 9 중 어느 한 조항에 있어서, 이웃 샘플의 제1 세트는 이웃 샘플의 제2 세트와 상이한, 방법.
- [0213] 11. 조항 5 내지 조항 9 중 어느 한 조항에 있어서, 이웃 샘플의 제1 세트는 이웃 샘플의 제2 세트와 동일한, 방법.
- [0214] 12. 조항 1 내지 조항 11 중 어느 한 조항에 있어서,
- [0215] 정제된 움직임 사용여 여부를 사용하여 움직임 보상을 수행하는 단계; 및
- [0216] GPM 파티션 모드에 따른 기하학적 파티셔닝 에지를 따라 블렌딩 프로세스를 적용하는 단계
- [0217] 를 더 포함하는, 방법.
- [0218] 13. 조항 1 내지 조항 12 중 어느 한 조항에 있어서,
- [0219] 템플릿 매칭이 코딩 유닛에 적용되지 않는 것에 응답하여, 움직임 벡터 차이를 갖는 병합 모드(merge mode with motion vector difference, MMVD)가 적용되는지를 표시하는 제2 매개변수를 디코딩하는 단계;
- [0220] MMVD가 적용되는 것에 응답하여, 움직임 벡터 차이(MVD) 정보를 디코딩하는 단계; 및
- [0221] MVD 정보를 사용하여 움직임을 정제하는 단계
- [0222] 를 더 포함하는, 방법.
- [0223] 14. 조항 13에 있어서, 제2 매개변수는 MMVD가 제1 파티션에 적용되는지를 표시하는 제1 플래그, 및 MMVD가 제2 파티션에 적용되는지를 표시하는 제2 플래그를 포함하는, 방법.
- [0224] 15. 조항 1 내지 조항 12 중 어느 한 조항에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 방법은:

- [0225] MMVD가 제1 파티션에 적용되는지를 표시하는 제2 매개변수를 디코딩하는 단계;
- [0226] MMVD가 제2 파티션에 적용되는지를 표시하는 제3 매개변수를 디코딩하는 단계;
- [0227] MMVD가 제1 파티션 또는 제2 파티션에 적용되지 않는 것에 응답하여, 제1 매개변수에 따라, 템플릿 매칭이 제1 파티션 및 제2 파티션 양쪽 모두에 적용되는지를 결정하는 단계; 및
- [0228] 템플릿 매칭이 제1 파티션 및 제2 파티션 양쪽 모두에 적용되는 것에 응답하여, 템플릿 매칭을 사용하여 제1 파티션 및 제2 파티션에 대한 움직임 정보를 정제하는 단계
- [0229] 를 더 포함하는, 방법.
- [0230] 16. 조항 15에 있어서, 움직임을 정제하기 위해 사용되는 템플릿이 파티션 모드에 기초하여 결정되는, 방법.
- [0231] 17. 비디오 데이터 처리를 수행하기 위한 장치로서, 장치는:
- [0232] 명령어를 저장하도록 구성되는 메모리; 및
- [0233] 하나 이상의 프로세서로서, 장치가:
- [0234] 기하학적 파티션 모드(GPM)에서 코딩된 코딩 유닛을 포함하는 비트스트림을 수신하는 것;
- [0235] 코딩 유닛과 연관된 제1 매개변수를 디코딩하는 것 - 제1 매개변수는 템플릿 매칭이 코딩 유닛에 적용되는지를 표시함 -; 및
- [0236] 제1 매개변수에 따라, 코딩 유닛에 대한 움직임 정보를 결정하는 것 - 제1 매개변수가 템플릿 매칭이 코딩 유닛에 적용됨을 표시할 때, 템플릿 매칭을 사용하여 움직임 정보가 정제됨 -
- [0237] 을 수행하게 하기 위해 명령어를 실행하도록 구성되는, 하나 이상의 프로세서
- [0238] 를 포함하는, 장치.
- [0239] 18. 조항 17에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 코딩 유닛에 대한 움직임 정보는 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임을 포함하고, 제1 매개변수는 제1 플래그를 포함하고, 프로세서는 장치가:
- [0240] 제1 플래그에 따라 템플릿 매칭을 사용하여 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임 양쪽 모두를 정제하는 것을 수행하게 하기 위해 명령어를 실행하도록 더 구성되는, 장치.
- [0241] 19. 조항 17에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 제1 매개변수는 템플릿 매칭이 제1 파티션에 적용되는지를 표시하는 제1 플래그, 및 템플릿 매칭이 제2 파티션에 적용되는지를 표시하는 제2 플래그를 포함하고, 프로세서는 장치가:
- [0242] 제1 플래그에 따라 템플릿 매칭을 사용하여 제1 파티션의 제1 움직임을 정제하는 것; 및
- [0243] 제2 플래그에 따라 템플릿 매칭을 사용하여 제2 파티션의 제2 움직임을 정제하는 것
- [0244] 을 수행하게 하기 위해 명령어를 실행하도록 더 구성되는, 장치.
- [0245] 20. 조항 17에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 제1 매개변수는 템플릿 매칭이 제1 파티션 및 제2 파티션에 적용되는지를 표시하는 인덱스를 포함하고, 프로세서는 장치가:
- [0246] 인덱스의 값에 따라, 제1 파티션의 제1 움직임, 제2 파티션의 제2 움직임, 또는 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임 양쪽 모두를 정제하는 것을 수행하게 하기 위해 명령어를 실행하도록 더 구성되는, 장치.
- [0247] 21. 조항 17 내지 조항 20 중 어느 한 조항에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 움직임 정보는 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임을 포함하고, 프로세서는 장치가:
- [0248] 제1 파티션에 대한 제1 템플릿을 구성하는 것 - 제1 템플릿은 이웃 샘플의 제1 세트로부터 구성됨 -;
- [0249] 제2 파티션에 대한 제2 템플릿을 구성하는 것 - 제2 템플릿은 이웃 샘플의 제2 세트로부터 구성됨 -; 및
- [0250] 제1 및 제2 템플릿을 개별적으로 사용하여 제1 및 제2 움직임을 정제하는 것

- [0251] 을 수행하게 하기 위해 명령어를 실행하도록 더 구성되고,
- [0252] 이웃 샘플의 제1 및 제2 세트의 각각은:
- [0253] 오직 좌측 이웃 샘플만,
- [0254] 오직 상부 이웃 샘플만, 또는
- [0255] 좌측 이웃 샘플 및 상부 이웃 샘플 양쪽 모두
- [0256] 로부터 선택된 하나 이상의 이웃 샘플을 포함하는, 장치.
- [0257] 22. 조항 21에 있어서, 이웃 샘플의 제1 세트는 제1 파티션의 좌측 이웃 샘플 및 상부 이웃 샘플의 이용 가능성에 기초하여 선택되고,
- [0258] 제2 세트 이웃 샘플은 제2 파티션의 좌측 이웃 샘플 및 상부 이웃 샘플의 이용 가능성에 기초하여 선택되는, 장치.
- [0259] 23. 조항 21 또는 조항 22에 있어서, 프로세서는 장치가:
- [0260] GPM 파티션 모드에 기초하여 제1 템플릿 및 제2 템플릿을 구성하는 것을 수행하게 하기 위해 명령어를 실행하도록 더 구성되는, 장치.
- [0261] 24. 조항 23에 있어서, 프로세서는 장치가:
- [0262] GPM 파티션 각도에 기초하여 제1 템플릿 및 제2 템플릿을 구성하는 것을 수행하게 하기 위해 명령어를 실행하도록 더 구성되는, 장치.
- [0263] 25. 조항 21 내지 조항 24 중 어느 한 조항에 있어서, 프로세서는 장치가:
- [0264] 제1 템플릿과 연관된 제2 매개변수를 디코딩하는 것,
- [0265] 제2 템플릿과 연관된 제3 매개변수를 디코딩하는 것
- [0266] 을 수행하게 하기 위해 명령어를 실행하도록 더 구성되고,
- [0267] 제2 및 제3 매개변수는 개별적으로, 이웃 샘플의 제1 및 제2 세트가:
- [0268] 오직 좌측 이웃 샘플만,
- [0269] 오직 상부 이웃 샘플만, 또는
- [0270] 좌측 이웃 샘플 및 상부 이웃 샘플 양쪽 모두
- [0271] 로부터 선택되는지를 표시하는, 장치.
- [0272] 26. 조항 21 내지 조항 25 중 어느 한 조항에 있어서, 이웃 샘플의 제1 세트는 이웃 샘플의 제2 세트와 상이한, 장치.
- [0273] 27. 조항 21 내지 조항 25 중 어느 한 조항에 있어서, 이웃 샘플의 제1 세트는 이웃 샘플의 제2 세트와 동일한, 장치.
- [0274] 28. 조항 17 내지 조항 27 중 어느 한 조항에 있어서, 프로세서는 장치가:
- [0275] 정제된 움직임을 사용하여 움직임 보상을 수행하는 것; 및
- [0276] GPM 파티션 모드에 따른 기하학적 파티셔닝 에지를 따라 블렌딩 프로세스를 적용하는 것
- [0277] 을 수행하게 하기 위해 명령어를 실행하도록 더 구성되는, 장치.
- [0278] 29. 조항 17 내지 조항 28항 중 어느 한 조항에 있어서, 프로세서는 장치가:
- [0279] 템플릿 매칭이 코딩 유닛에 적용되지 않는 것에 응답하여, 움직임 벡터 차이를 갖는 병합 모드(MMVD)가 적용되는지를 표시하는 제2 매개변수를 디코딩하는 것;
- [0280] MMVD가 적용되는 것에 응답하여, 움직임 벡터 차이(MVD) 정보를 디코딩하는 것; 및
- [0281] MVD 정보를 사용하여 움직임을 정제하는 것

- [0282] 을 수행하게 하기 위해 명령어를 실행하도록 더 구성되는, 장치.
- [0283] 30. 조항 29에 있어서, 제2 매개변수는 MMVD가 제1 파티션에 적용되는지를 표시하는 제1 플래그, 및 MMVD가 제2 파티션에 적용되는지를 표시하는 제2 플래그를 포함하는, 장치.
- [0284] 31. 조항 17 내지 조항 28 중 어느 한 조항에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 프로세서는 장치가:
  - [0285] MMVD가 제1 파티션에 적용되는지를 표시하는 제2 매개변수를 디코딩하는 것;
  - [0286] MMVD가 제2 파티션에 적용되는지를 표시하는 제3 매개변수를 디코딩하는 것;
  - [0287] MMVD가 제1 파티션 또는 제2 파티션에 적용되지 않는 것에 응답하여, 제1 매개변수에 따라, 템플릿 매칭이 제1 파티션 및 제2 파티션 양쪽 모두에 적용되는지를 결정하는 것; 및
  - [0288] 템플릿 매칭이 제1 파티션 및 제2 파티션 양쪽 모두에 적용되는 것에 응답하여, 템플릿 매칭을 사용하여 제1 파티션 및 제2 파티션에 대한 움직임 정보를 정제하는 것
- [0289] 을 수행하게 하기 위해 명령어를 실행하도록 더 구성되는, 장치.
- [0290] 32. 조항 31에 있어서, 움직임을 정제하기 위해 사용되는 템플릿이 파티션 모드에 기초하여 결정되는, 장치.
- [0291] 33. 장치가 비디오 데이터 처리를 수행하기 위한 방법을 개시하게 하기 위해 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트를 저장하는 비밀시적 컴퓨터 판독 가능 매체로서, 방법은:
  - [0292] 기하학적 파티션 모드(GPM)에서 코딩된 코딩 유닛을 포함하는 비트스트림을 수신하는 단계;
  - [0293] 코딩 유닛과 연관된 제1 매개변수를 디코딩하는 단계 - 제1 매개변수는 템플릿 매칭이 코딩 유닛에 적용되는지를 표시함 -; 및
  - [0294] 제1 매개변수에 따라, 코딩 유닛에 대한 움직임 정보를 결정하는 단계 - 제1 매개변수가 템플릿 매칭이 코딩 유닛에 적용됨을 표시할 때, 템플릿 매칭을 사용하여 움직임 정보가 정제됨 -
  - [0295] 를 포함하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0296] 34. 조항 33에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 코딩 유닛에 대한 움직임 정보는 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임을 포함하고, 제1 매개변수는 제1 플래그를 포함하고, 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트는 장치가:
  - [0297] 제1 플래그에 따라 템플릿 매칭을 사용하여 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임 양쪽 모두를 정제하는 것을 더 수행하게 하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0298] 35. 조항 33에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 제1 매개변수는 템플릿 매칭이 제1 파티션에 적용되는지를 표시하는 제1 플래그, 및 템플릿 매칭이 제2 파티션에 적용되는지를 표시하는 제2 플래그를 포함하고, 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트는 장치가:
  - [0299] 제1 플래그에 따라 템플릿 매칭을 사용하여 제1 파티션의 제1 움직임을 정제하는 것; 및
  - [0300] 제2 플래그에 따라 템플릿 매칭을 사용하여 제2 파티션의 제2 움직임을 정제하는 것
  - [0301] 을 더 수행하게 하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0302] 36. 조항 33에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 제1 매개변수는 템플릿 매칭이 제1 파티션 및 제2 파티션에 적용되는지를 표시하는 인덱스를 포함하고, 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트는 장치가:
  - [0303] 인덱스의 값에 따라, 제1 파티션의 제1 움직임, 제2 파티션의 제2 움직임, 또는 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임 양쪽 모두를 정제하는 것을 더 수행하게 하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0304] 37. 조항 33 내지 조항 36 중 어느 한 조항에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 움직임 정보는 제1 파티션의 제1 움직임 및 제2 파티션의 제2 움직임을 포함하고, 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트는 장치가:
  - [0305] 제1 파티션에 대한 제1 템플릿을 구성하는 것 - 제1 템플릿은 이웃 샘플의 제1 세트로부터 구성됨 -;

- [0306] 제2 파티션에 대한 제2 템플릿을 구성하는 것 - 제2 템플릿은 이웃 샘플의 제2 세트로부터 구성됨 -; 및
- [0307] 제1 및 제2 템플릿을 개별적으로 사용하여 제1 및 제2 움직임을 정제하는 것
- [0308] 을 더 수행하게 하고,
- [0309] 이웃 샘플의 제1 및 제2 세트의 각각은:
- [0310] 오직 좌측 이웃 샘플만,
- [0311] 오직 상부 이웃 샘플만, 또는
- [0312] 좌측 이웃 샘플 및 상부 이웃 샘플 양쪽 모두
- [0313] 로부터 선택된 하나 이상의 이웃 샘플을 포함하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0314] 38. 조항 37에 있어서, 이웃 샘플의 제1 세트는 제1 파티션의 좌측 이웃 샘플 및 상부 이웃 샘플의 이용 가능성에 기초하여 선택되고,
- [0315] 제2 세트 이웃 샘플은 제2 파티션의 좌측 이웃 샘플 및 상부 이웃 샘플의 이용 가능성에 기초하여 선택되는, 비밀시적 컴퓨터 판독 가능 매체.
- [0316] 39. 조항 37 또는 조항 38에 있어서, 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트는 장치가:
- [0317] GPM 파티션 모드에 기초하여 제1 템플릿 및 제2 템플릿을 구성하는 것을 더 수행하게 하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0318] 40. 조항 39에 있어서, 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트는 장치가:
- [0319] GPM 파티션 각도에 기초하여 제1 템플릿 및 제2 템플릿을 구성하는 것을 더 수행하게 하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0320] 41. 조항 37 내지 조항 40 중 어느 한 조항에 있어서, 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트는 장치가:
- [0321] 제1 템플릿과 연관된 제2 매개변수를 디코딩하는 것,
- [0322] 제2 템플릿과 연관된 제3 매개변수를 디코딩하는 것
- [0323] 을 더 수행하게 하고,
- [0324] 제2 및 제3 매개변수는 개별적으로, 이웃 샘플의 제1 및 제2 세트가:
- [0325] 오직 좌측 이웃 샘플만,
- [0326] 오직 상부 이웃 샘플만, 또는
- [0327] 좌측 이웃 샘플 및 상부 이웃 샘플 양쪽 모두
- [0328] 로부터 선택되는지를 표시하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0329] 42. 조항 37 내지 조항 41 중 어느 한 조항에 있어서, 이웃 샘플의 제1 세트는 이웃 샘플의 제2 세트와 상이한, 비밀시적 컴퓨터 판독 가능 매체.
- [0330] 43. 조항 37 내지 조항 41 중 어느 한 조항에 있어서, 이웃 샘플의 제1 세트는 이웃 샘플의 제2 세트와 동일한, 비밀시적 컴퓨터 판독 가능 매체.
- [0331] 44. 조항 33 내지 조항 43 중 어느 한 조항에 있어서, 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트는 장치가:
- [0332] 정제된 움직임을 사용하여 움직임 보상을 수행하는 것; 및
- [0333] GPM 파티션 모드에 따른 기하학적 파티셔닝 에지를 따라 블렌딩 프로세스를 적용하는 것
- [0334] 을 더 수행하게 하는, 비밀시적 컴퓨터 판독 가능 매체.

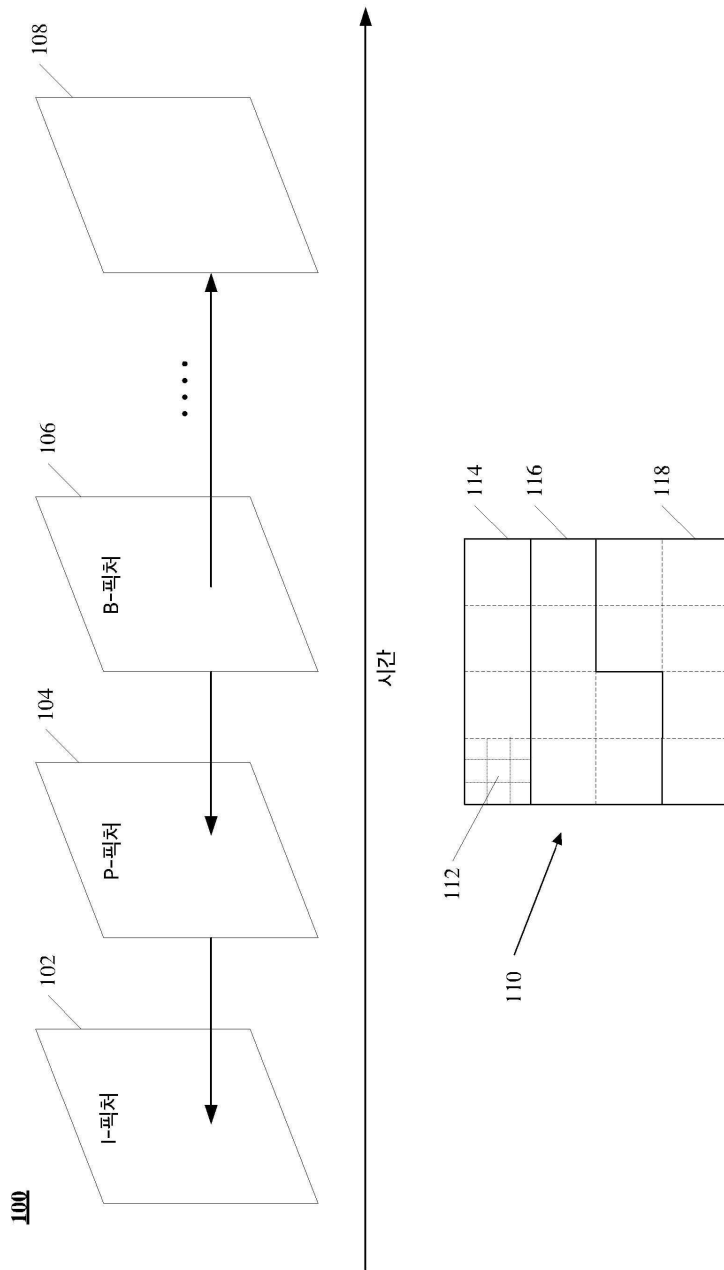
- [0335] 45. 조항 33 내지 조항 44 중 어느 한 조항에 있어서, 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트는 장치가:
- [0336] 템플릿 매칭이 코딩 유닛에 적용되지 않는 것에 응답하여, 움직임 벡터 차이를 갖는 병합 모드(MMVD)가 적용되는지를 표시하는 제2 매개변수를 디코딩하는 것;
- [0337] MMVD가 적용되는 것에 응답하여, 움직임 벡터 차이(MVD) 정보를 디코딩하는 것; 및
- [0338] MVD 정보를 사용하여 움직임을 정제하는 것
- [0339] 을 더 수행하게 하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0340] 46. 조항 45에 있어서, 제2 매개변수는 MMVD가 제1 파티션에 적용되는지를 표시하는 제1 플래그, 및 MMVD가 제2 파티션에 적용되는지를 표시하는 제2 플래그를 포함하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0341] 47. 조항 33 내지 조항 44 중 어느 한 조항에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 장치의 하나 이상의 프로세서에 의해 실행 가능한 명령어의 세트는 장치가:
- [0342] MMVD가 제1 파티션에 적용되는지를 표시하는 제2 매개변수를 디코딩하는 것;
- [0343] MMVD가 제2 파티션에 적용되는지를 표시하는 제3 매개변수를 디코딩하는 것;
- [0344] MMVD가 제1 파티션 또는 제2 파티션에 적용되지 않는 것에 응답하여, 제1 매개변수에 따라, 템플릿 매칭이 제1 파티션 및 제2 파티션 양쪽 모두에 적용되는지를 결정하는 것; 및
- [0345] 템플릿 매칭이 제1 파티션 및 제2 파티션 양쪽 모두에 적용되는 것에 응답하여, 템플릿 매칭을 사용하여 제1 파티션 및 제2 파티션에 대한 움직임 정보를 정제하는 것
- [0346] 을 더 수행하게 하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0347] 48. 조항 47에 있어서, 움직임을 정제하기 위해 사용되는 템플릿이 파티션 모드에 기초하여 결정되는, 비밀시적 컴퓨터 판독 가능 매체.
- [0348] 49. 비트스트림을 저장하는 비밀시적 컴퓨터 판독 가능 매체로서, 비트스트림은:
- [0349] 코딩 유닛과 연관된 제1 매개변수 - 제1 매개변수는 템플릿 매칭이 적용되는지를 표시하고, 코딩 유닛은 기하학적 파티션 모드(GPM)에서 코딩됨 - 를 포함하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0350] 50. 조항 49에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 제1 매개변수는:
- [0351] 템플릿 매칭이 제1 파티션에 적용되는지를 표시하는 제1 플래그; 및
- [0352] 템플릿 매칭이 제2 파티션에 적용되는지를 표시하는 제2 플래그
- [0353] 를 더 포함하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0354] 51. 조항 49에 있어서, 비트스트림은:
- [0355] 코딩 유닛과 연관된 제2 매개변수를 더 포함하고, 제2 매개변수는 템플릿 매칭을 위한 템플릿을 표시하며, 템플릿은:
- [0356] 오직 좌측 이웃 샘플만,
- [0357] 오직 상부 이웃 샘플만, 또는
- [0358] 좌측 이웃 샘플 및 상부 이웃 샘플 양쪽 모두
- [0359] 로부터 선택된 하나 이상의 이웃 샘플로부터 구성되는, 비밀시적 컴퓨터 판독 가능 매체.
- [0360] 52. 조항 49에 있어서, 비트스트림은:
- [0361] 코딩 유닛과 연관된 제2 매개변수를 더 포함하고, 제2 매개변수는 움직임 벡터 차이를 갖는 병합 모드(MMVD)가 적용되는지를 표시하는, 비밀시적 컴퓨터 판독 가능 매체.
- [0362] 53. 조항 49에 있어서, 코딩 유닛은 제1 파티션 및 제2 파티션으로 분할되고, 비트스트림은:
- [0363] 코딩 유닛과 연관된 제2 매개변수 - 제2 매개변수는 움직임 벡터 차이를 갖는 병합 모드(MMVD)가 제1 파티션에

적용되는지를 표시함 -; 및

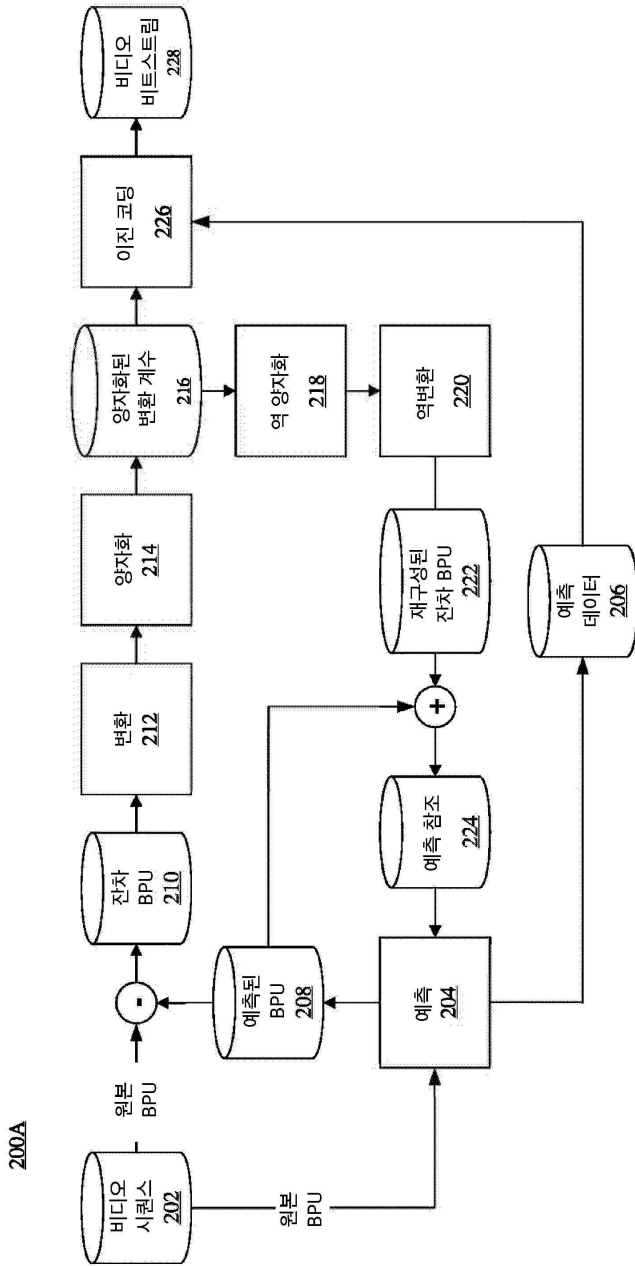
- [0364] 비디오 데이터와 연관된 제3 매개변수 - 제3 매개변수는 MMVD가 제2 파티션에 적용되는지를 표시함 -
- [0365] 를 더 포함하는, 비밀시적 컴퓨터 관독 가능 매체.
- [0366] 일부 실시형태에서, 명령어를 포함하는 비밀시적 컴퓨터 관독 가능 저장 매체가 또한 제공된다. 일부 실시형태에서, 매체는, GPM, TM 또는 MMVD가 CU 또는 파티션에 적용되는 것을 표시하는 하나 이상의 플래그를 갖는 비디오 비트스트림의 전부 또는 부분을 저장할 수 있다. 일부 실시형태에서, 매체는 위에서 설명되는 방법을 수행하기 위해 (개시된 인코더 및 디코더와 같은) 디바이스에 의해 실행될 수 있는 명령어를 저장할 수 있다. 비밀시적 매체의 보편적 형태는 예를 들어, 플로피 디스크, 플렉서블 디스크(flexible disk), 하드 디스크, 솔리드 스테이트 드라이브, 자기 테이프 또는 임의의 다른 자기 데이터 저장 매체, CD-ROM, 임의의 다른 광학 데이터 저장 매체, 구멍의 패턴을 갖는 임의의 물리적 매체, RAM, PROM 및 EPROM, FLASH-EPROM 또는 임의의 다른 플래시 메모리, NVRAM, 캐시, 레지스터, 임의의 다른 메모리 칩 또는 카트리지 및 이들의 네트워크화된 버전을 포함한다. 디바이스는 하나 이상의 프로세서(CPU), 입/출력 인터페이스, 네트워크 인터페이스 및/또는 메모리를 포함할 수 있다.
- [0367] 본원에서 "제1" 및 "제2"와 같은 관계 용어는 엔티티(entity) 또는 동작을 다른 엔티티 또는 동작과 구분하기 위해서만 사용되며, 이러한 엔티티 또는 동작 간의 임의의 실제 관계 또는 시퀀스를 요구하거나 암시하지 않는다는 것이 유의되어야 한다. 나아가, "포함하는(comprising)", "갖는(having)", "포함하는(containing)" 및 "포함하는(including)"이라는 단어 및 다른 유사한 형태는 그 의미가 동등하고, 이러한 단어 중 어느 하나를 따르는 항목(item) 또는 항목들이 이러한 항목 또는 항목들의 망라적인 열거(exhaustive listing)를 의미하지 않거나, 또는 열거된 항목 또는 항목들만으로 제한되도록 의미하지 않는다는 점에서 제약을 두지 않는 것(open ended)으로 의도된다.
- [0368] 본원에서 사용되는 바와 같이, 달리 구체적으로 언급되지 않는 한, "또는"이라는 용어는 실행 불가능한 경우를 제외하고 모든 가능한 조합을 포함한다. 예를 들어, 데이터베이스가 A 또는 B를 포함할 수 있다고 명시되어 있는 경우, 달리 구체적으로 명시되어 있거나 실행 불가능한 경우를 제외하고, 데이터베이스는 A 또는 B, 또는 A 및 B를 포함할 수 있다. 제2 예로서, 데이터베이스가 A, B 또는 C를 포함할 수 있다고 명시되어 있는 경우, 달리 구체적으로 명시되어 있거나 실행 불가능한 경우를 제외하고, 데이터베이스는 A 또는 B 또는 C, 또는 A 및 B, 또는 A 및 C, 또는 B 및 C, 또는 A 및 B 및 C를 포함할 수 있다.
- [0369] 상술한 실시형태는 하드웨어 또는 소프트웨어(프로그램 코드), 또는 하드웨어 및 소프트웨어의 조합에 의해 구현될 수 있다는 것이 인식된다. 소프트웨어에 의해 구현되는 경우, 이는 상술한 컴퓨터 관독 가능 매체에 저장될 수 있다. 소프트웨어는 프로세서에 의해 실행될 때 개시된 방법을 수행할 수 있다. 본 개시에서 설명되는 컴퓨팅 유닛 및 다른 기능 유닛은 하드웨어 또는 소프트웨어, 또는 하드웨어 및 소프트웨어의 조합에 의해 구현될 수 있다. 당해 기술 분야의 통상의 기술자는 또한 위에서 설명된 모듈/유닛 중 다수가 하나의 모듈/유닛으로 조합될 수 있고, 위에서 설명된 모듈/유닛 각각이 복수의 서브-모듈/서브-유닛으로 더 분할될 수 있음을 이해할 것이다.
- [0370] 전술한 명세서에서, 실시형태는 구현마다 변형될 수 있는 다수의 특정 세부사항을 참조하여 설명되었다. 설명되는 실시형태의 특정 적용 및 수정이 이루어질 수 있다. 본원에서 개시되는 본 발명의 명세서 및 실시를 고려함으로써, 다른 실시형태가 당해 기술 분야의 통상의 기술자에게 명백할 수 있다. 본 명세서 및 예는 단지 예시적인 것으로서 고려되도록 의도되며, 본 발명의 진정한 범주 및 사상은 다음의 청구범위에 의해 표시된다. 또한, 도면에서 도시되는 단계의 시퀀스는 단지 예시적인 목적을 위한 것으로 의도되며, 단계의 임의의 특정 시퀀스에 제한되도록 의도되지 않는다. 따라서, 당해 기술 분야의 통상의 기술자는 동일한 방법을 구현하면서 이들 단계가 상이한 순서로 수행될 수 있음을 인식할 수 있다.
- [0371] 본 도면 및 명세서에서 예시적인 실시형태가 개시되었다. 그러나, 이들 실시형태에 다수의 변형 및 수정이 이루어질 수 있다. 따라서, 특정 용어가 채용되고 있지만, 이들은 단지 일반적이고 설명적인 의미로 사용되며 제한의 목적으로는 사용되지 않는다.

도면

도면1

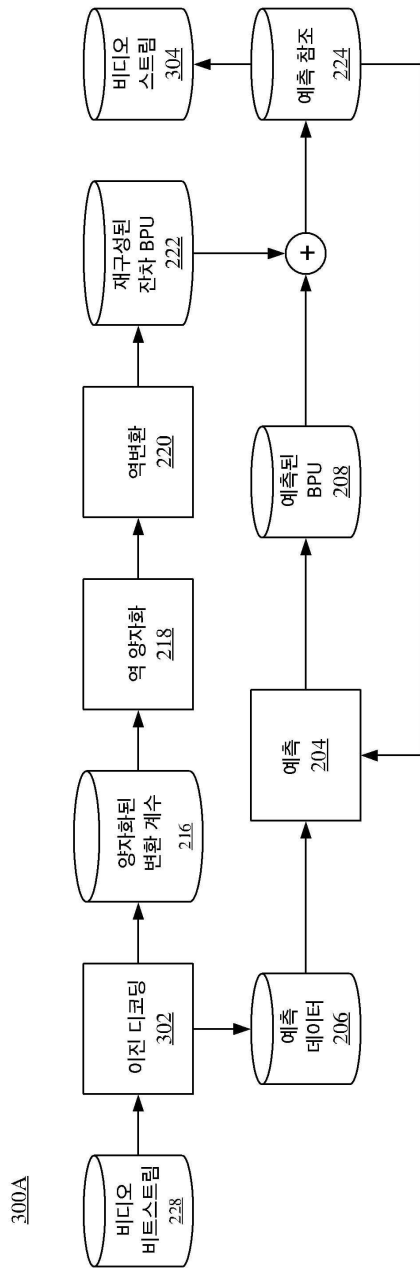


도면2a

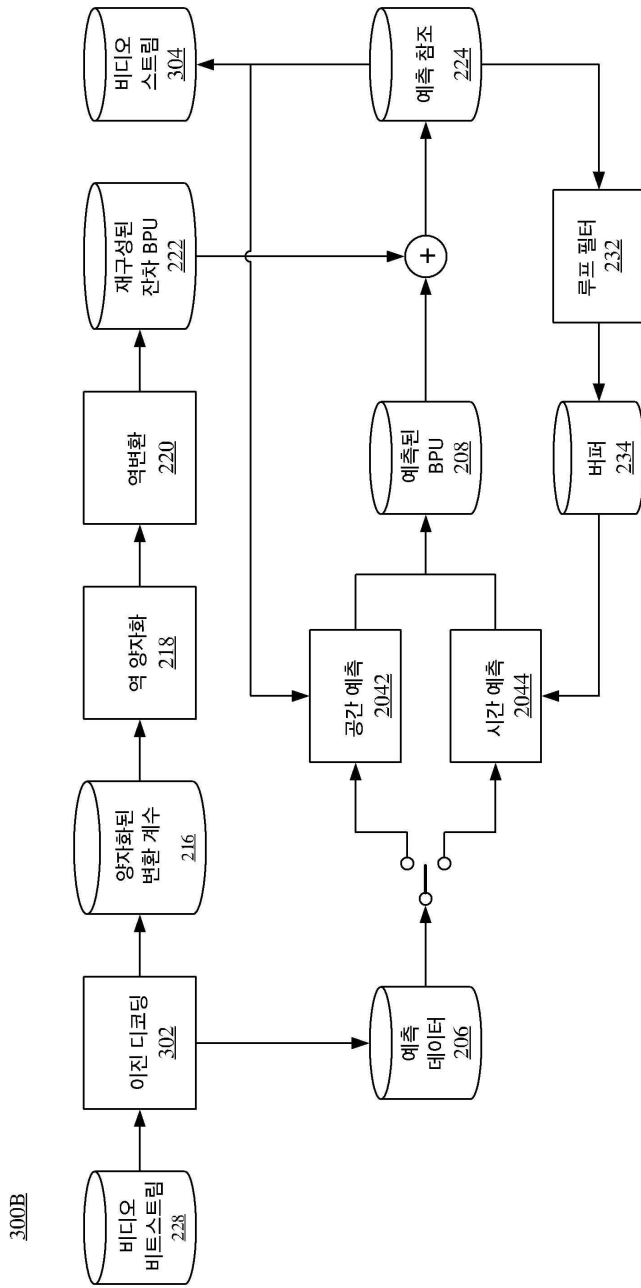




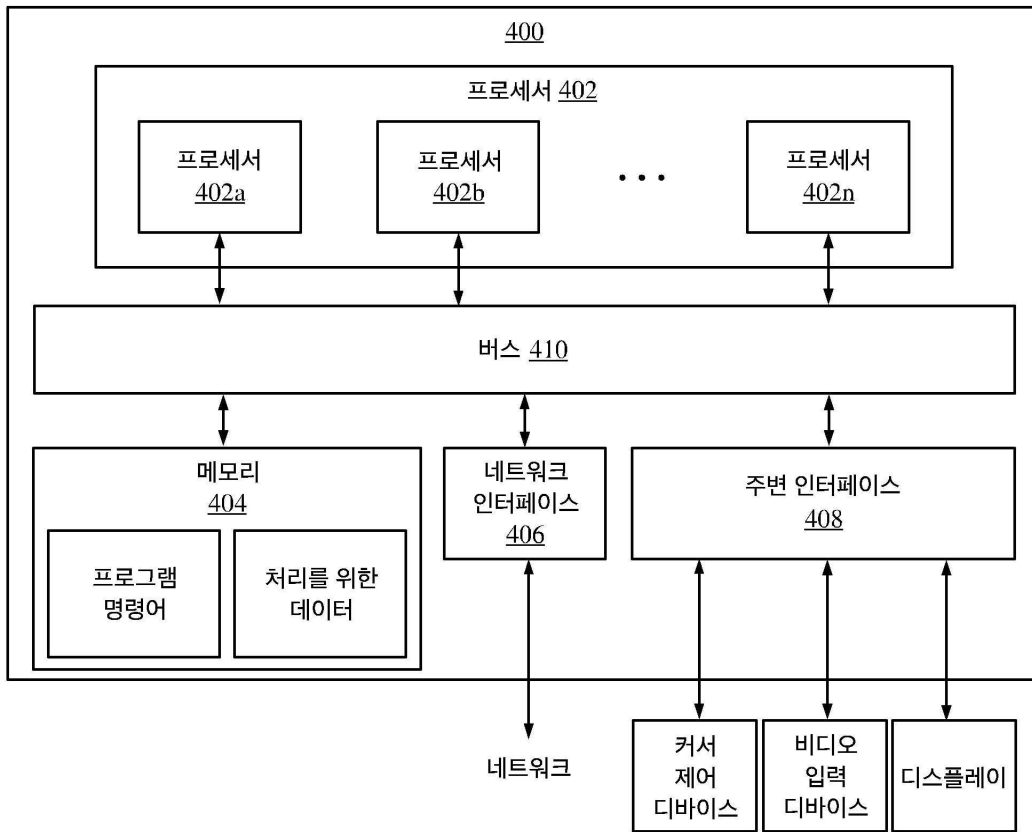
도면3a



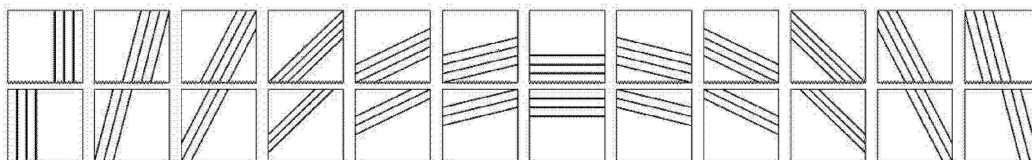
도면3b



도면4



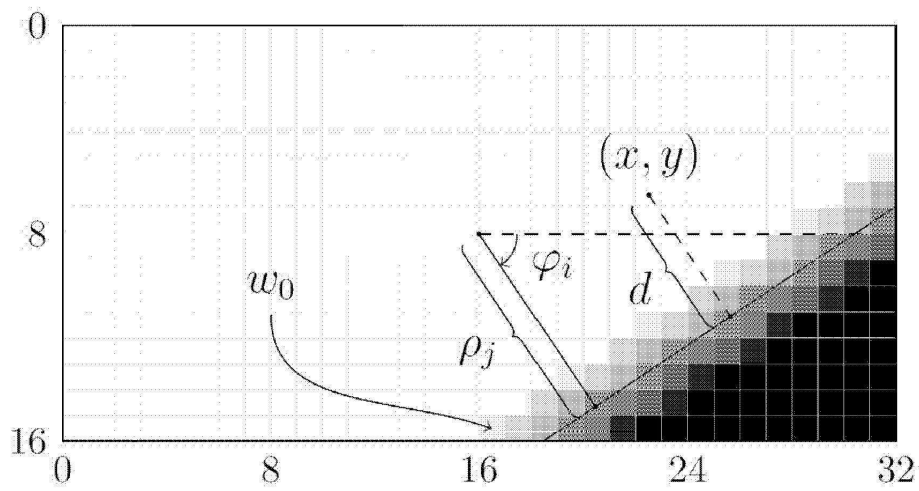
도면5



도면6

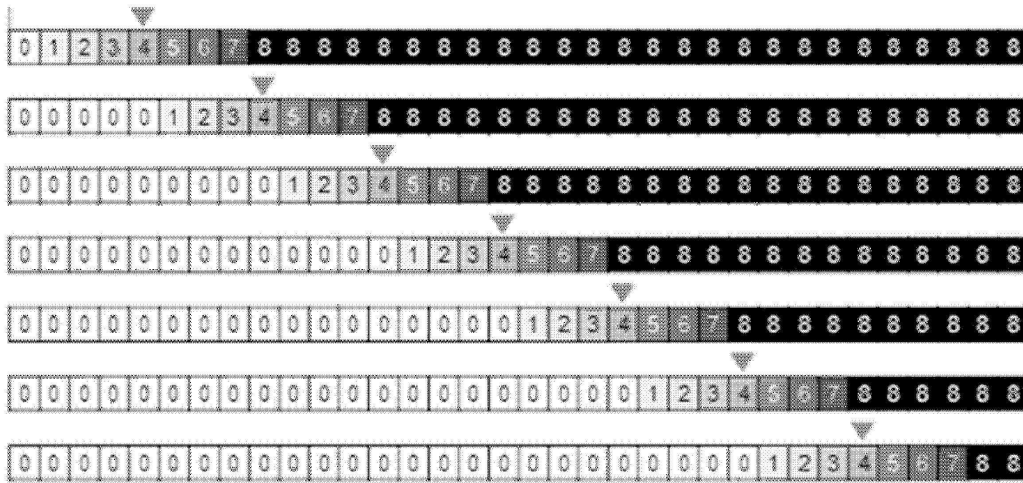
	LO MV	L1 MV
병합 인덱스 0	x	
1		x
2	x	
3		x
4	x	

도면7

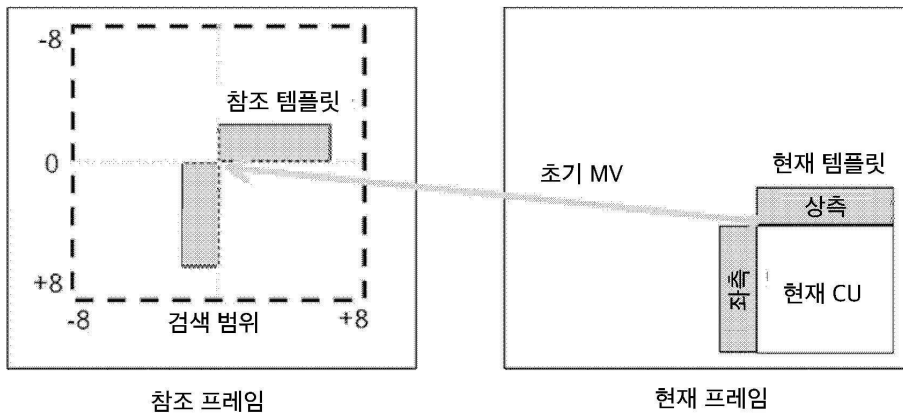




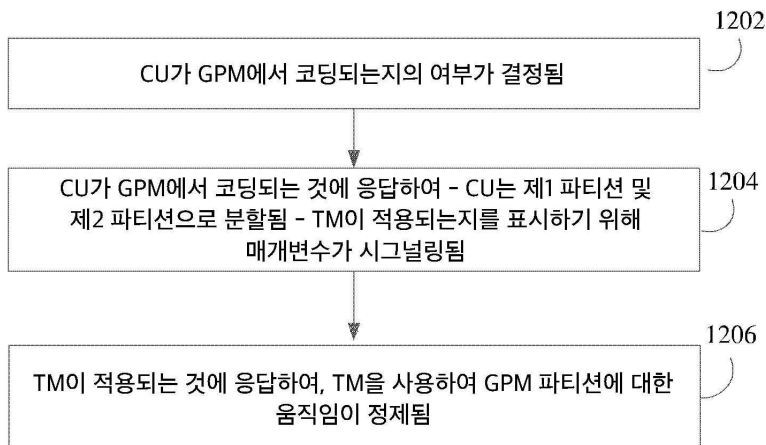
도면10



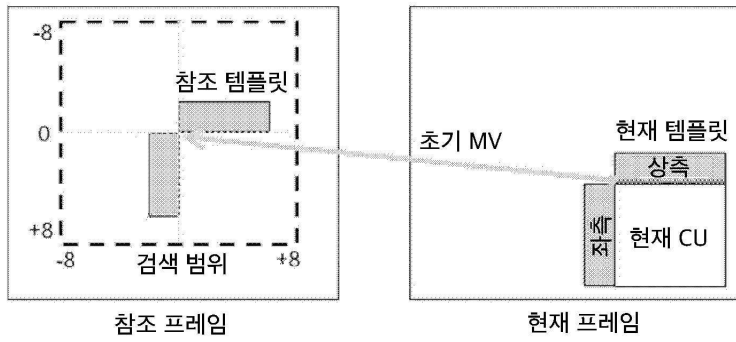
도면11



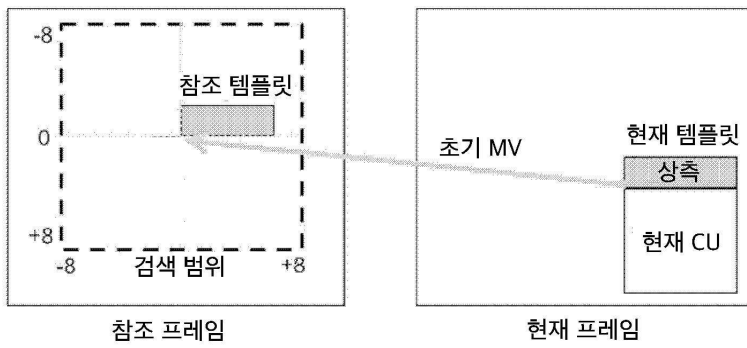
도면12



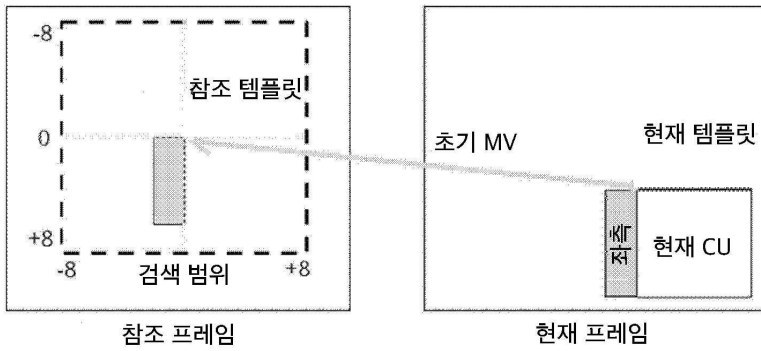
도면13a



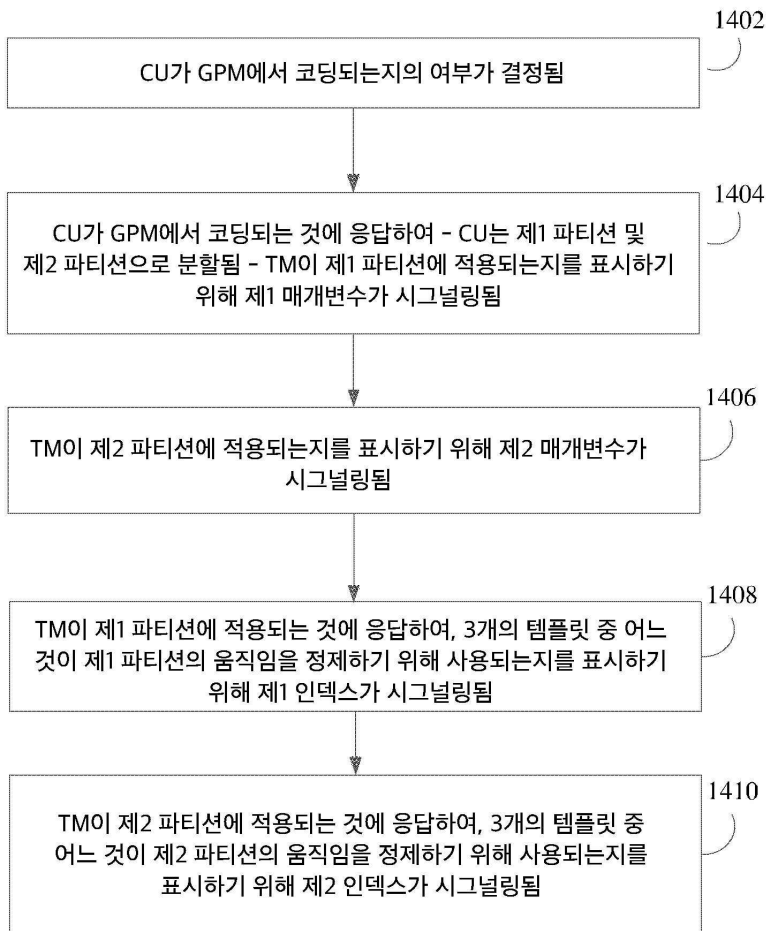
도면13b



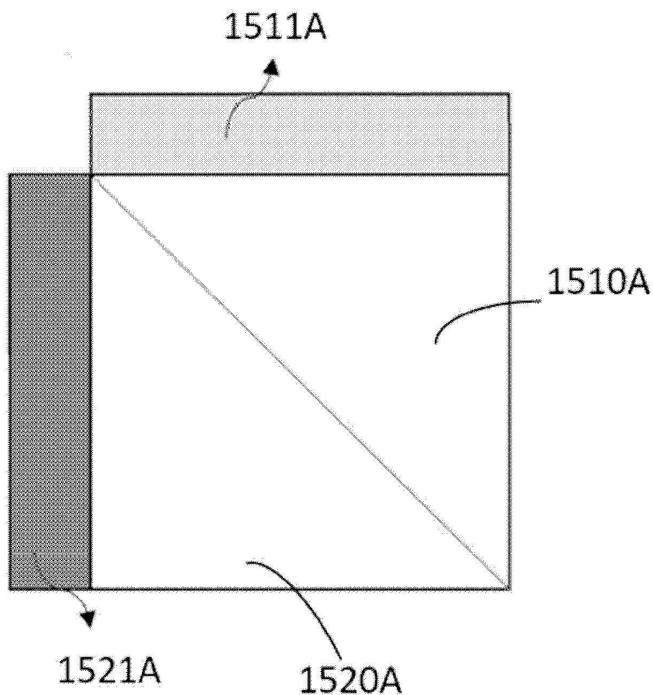
도면13c



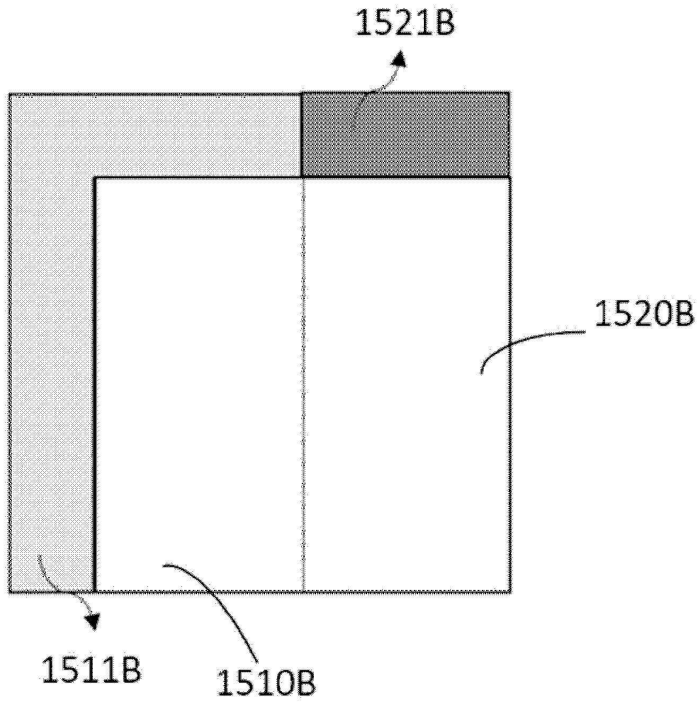
도면14



도면15a



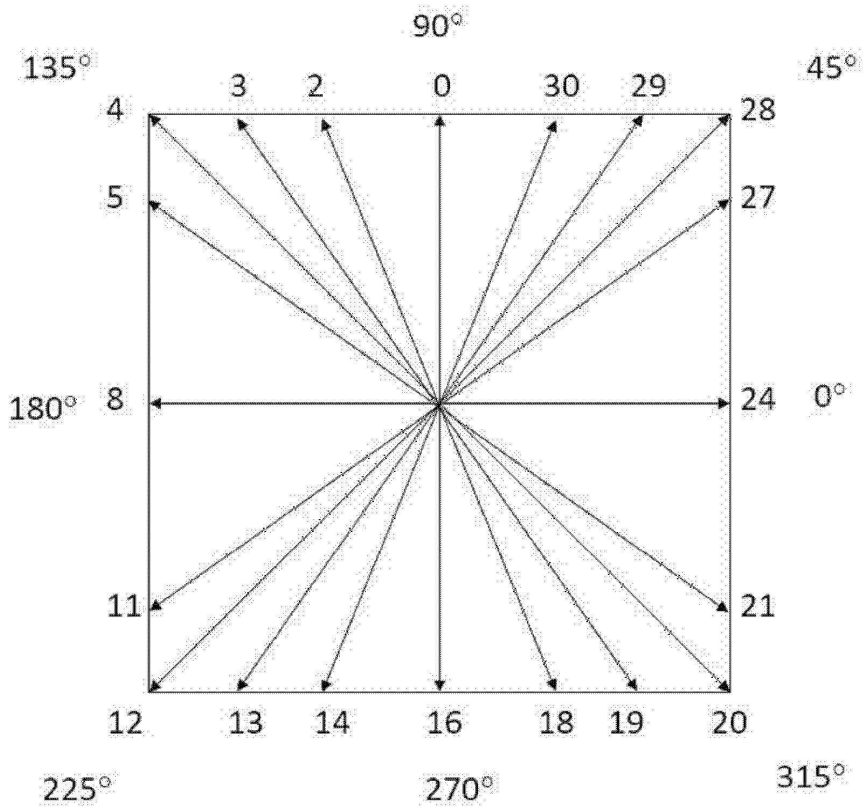
도면15b



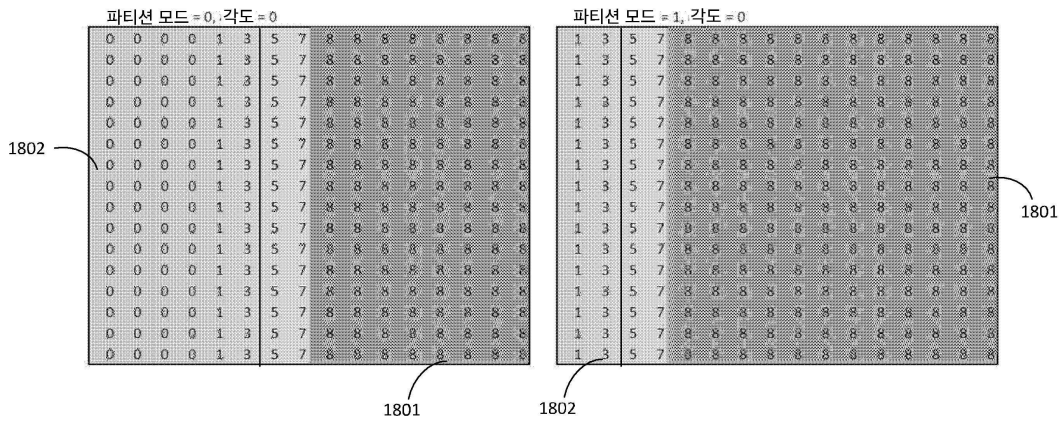
도면16

merge_gpm_partition_idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
angleIdx	0	0	2	2	2	2	3	3	3	3	4	4	4	4	5	5
distanceIdx	1	3	0	1	2	3	0	1	2	3	0	1	2	3	0	1
merge_gpm_partition_idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
angleIdx	5	5	8	8	11	11	11	11	12	12	12	12	13	13	13	13
distanceIdx	2	3	1	3	0	1	2	3	0	1	2	3	0	1	2	3
merge_gpm_partition_idx	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
angleIdx	14	14	14	14	16	16	18	18	18	19	19	19	20	20	20	21
distanceIdx	0	1	2	3	1	3	1	2	3	1	2	3	1	2	3	1
merge_gpm_partition_idx	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
angleIdx	21	21	24	24	27	27	27	28	28	28	29	29	29	30	30	30
distanceIdx	2	3	1	3	1	2	3	1	2	3	1	2	3	1	2	3

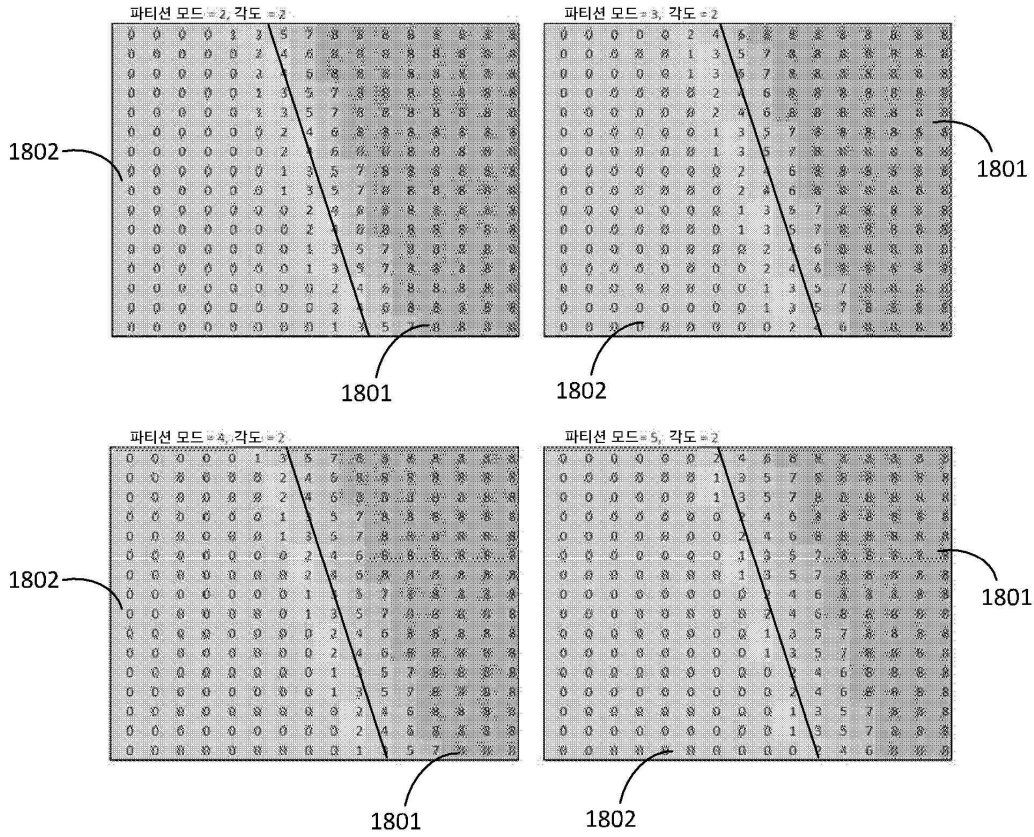
도면17



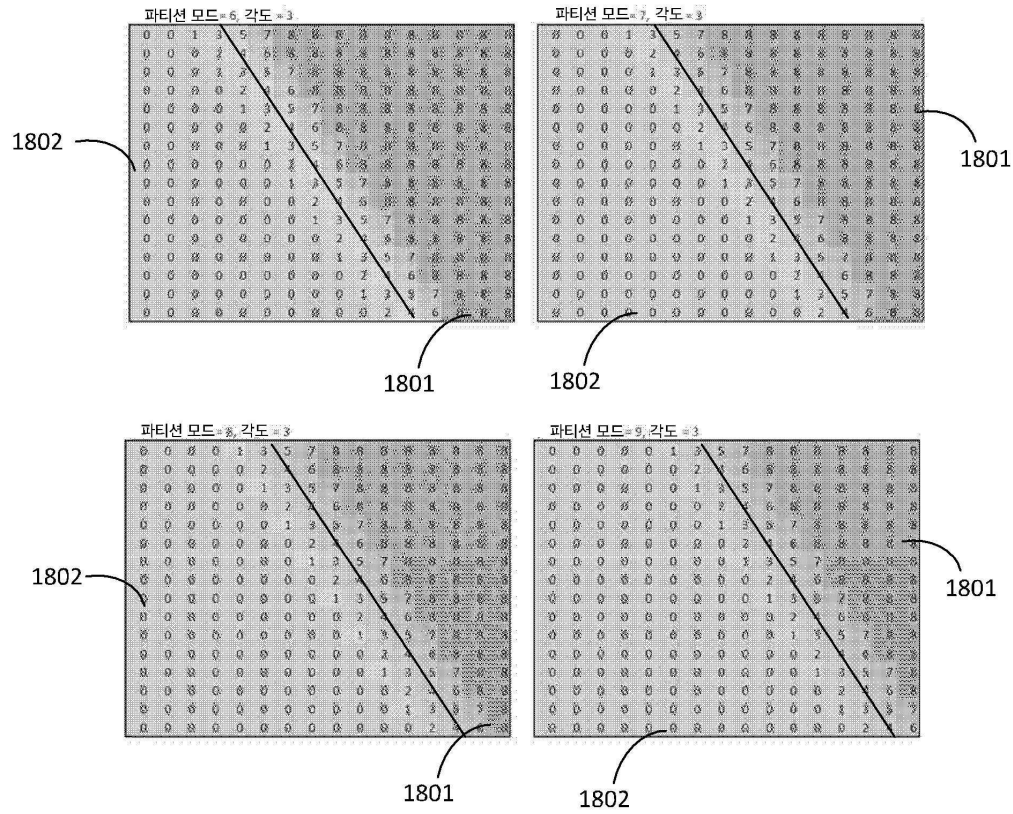
도면18a



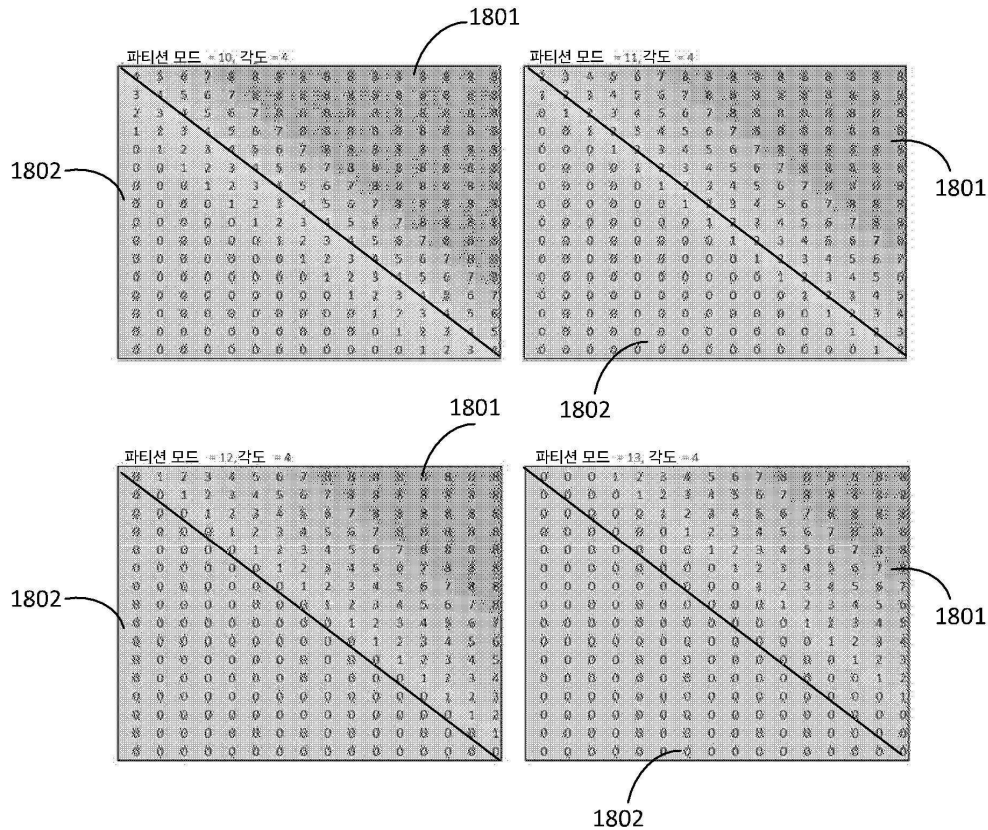
도면 18b



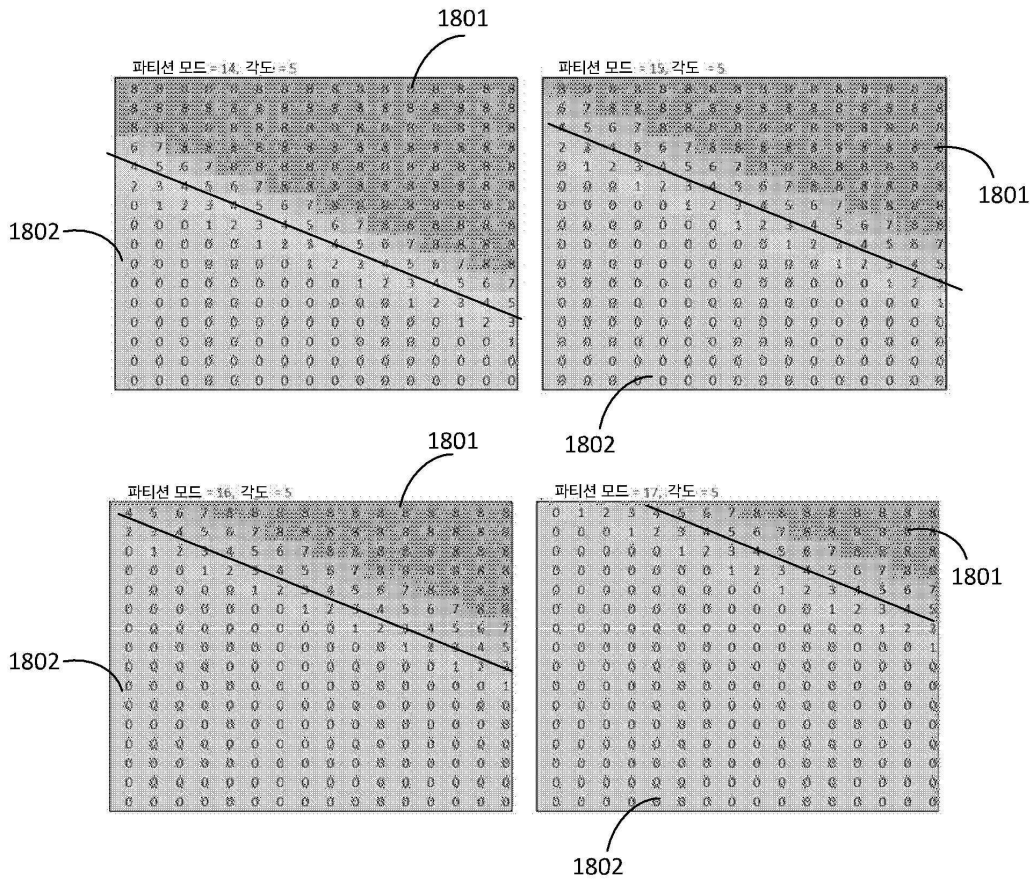
도면18c



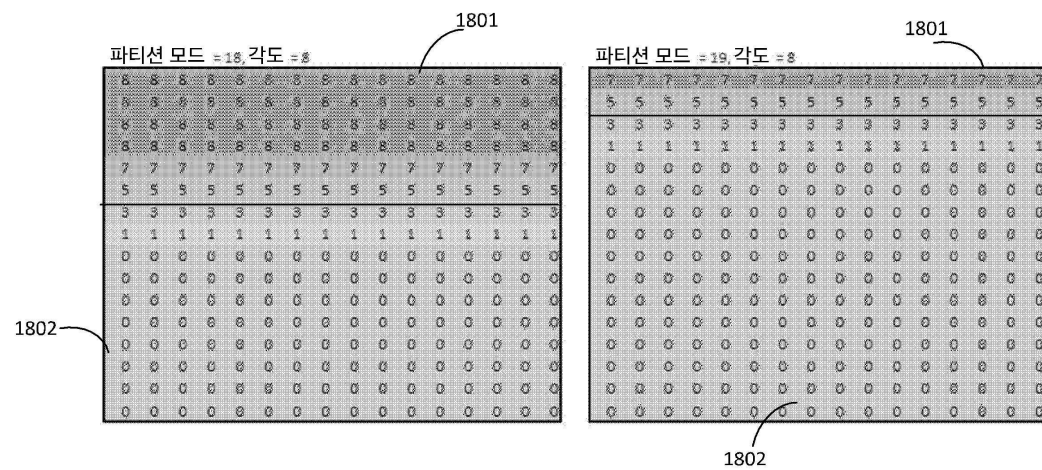
도면18d



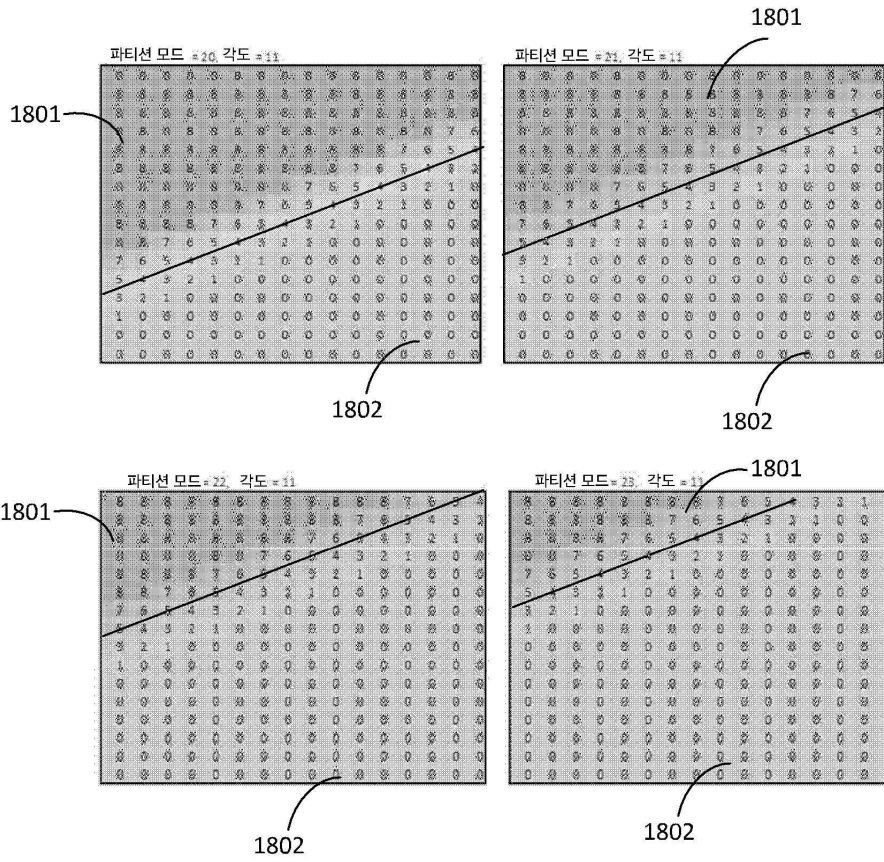
도면18e



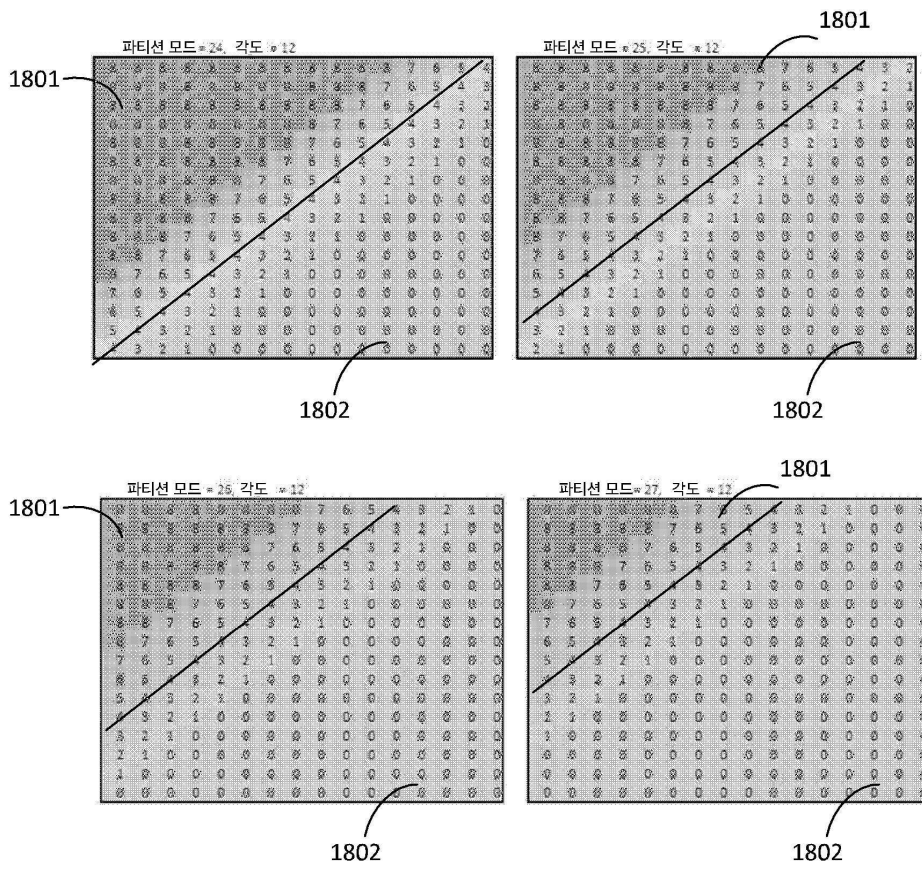
도면18f



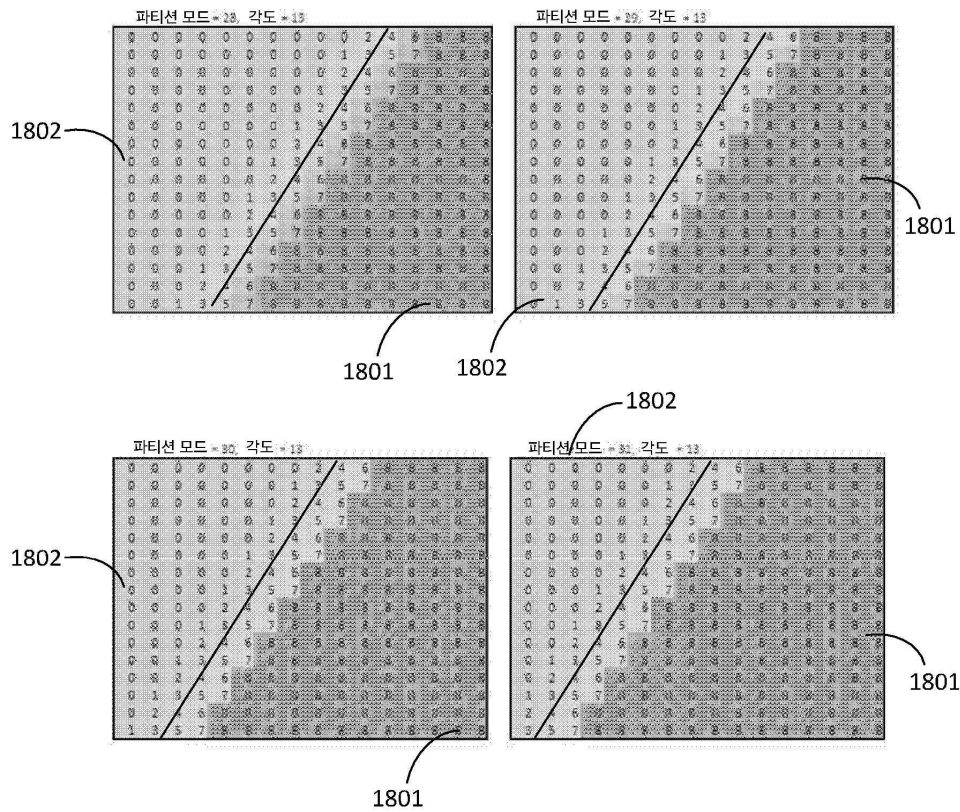
도면 18g



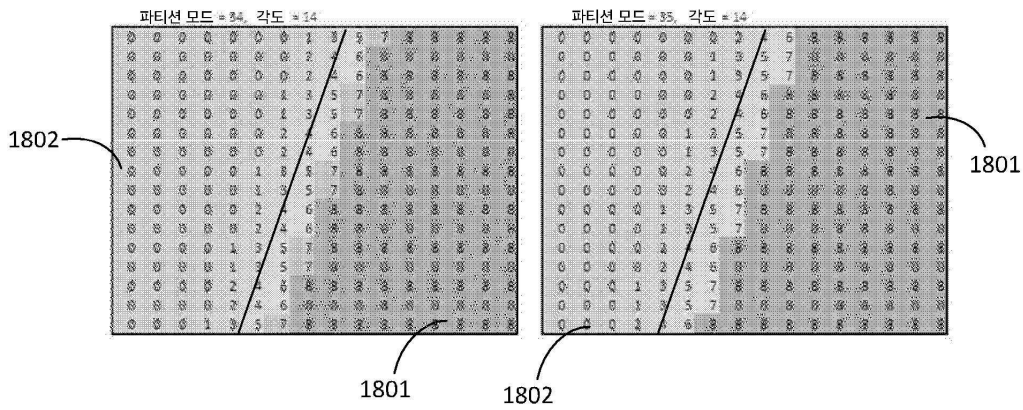
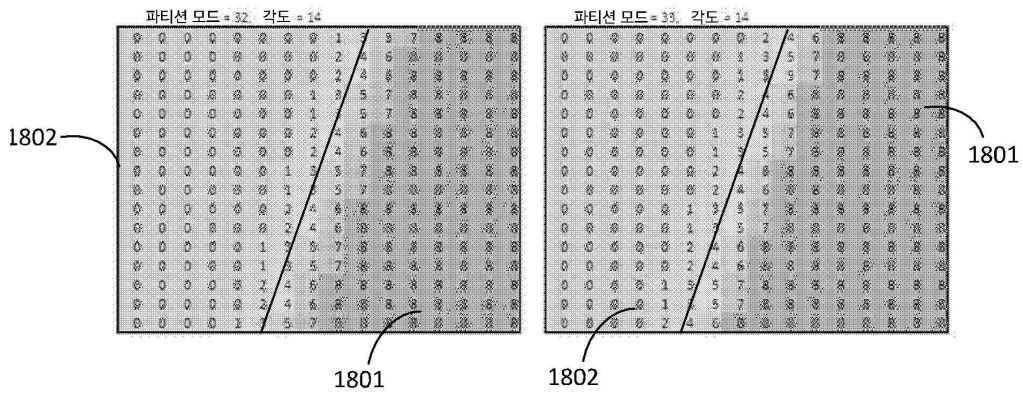
도면18h



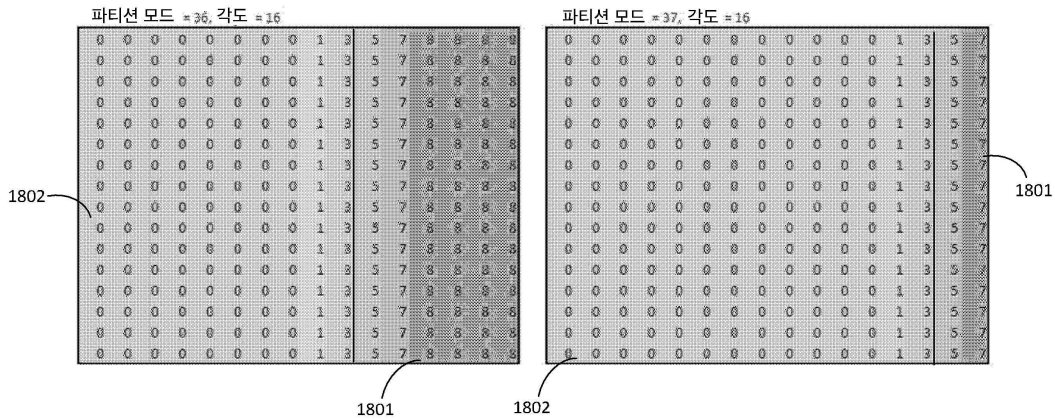
도면18i



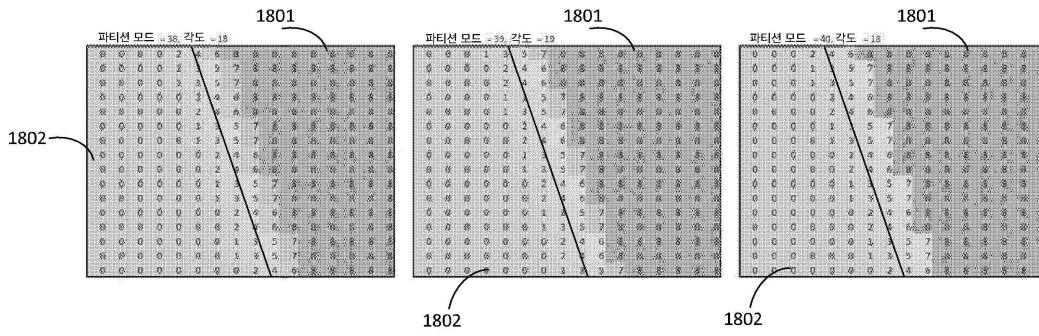
도면18j



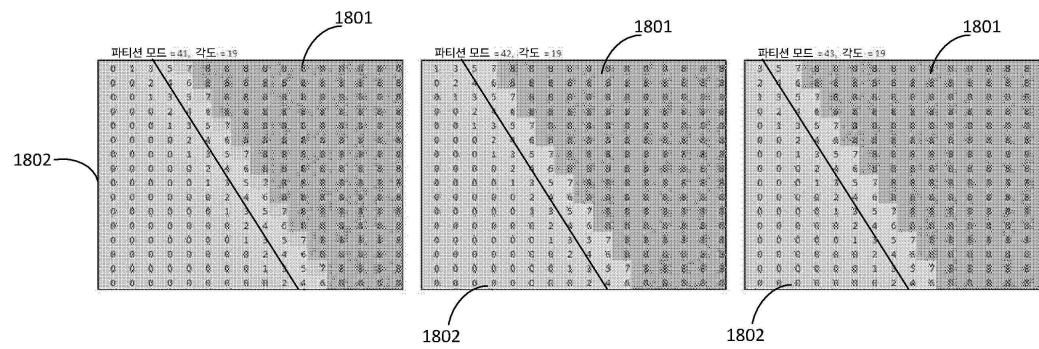
도면18k



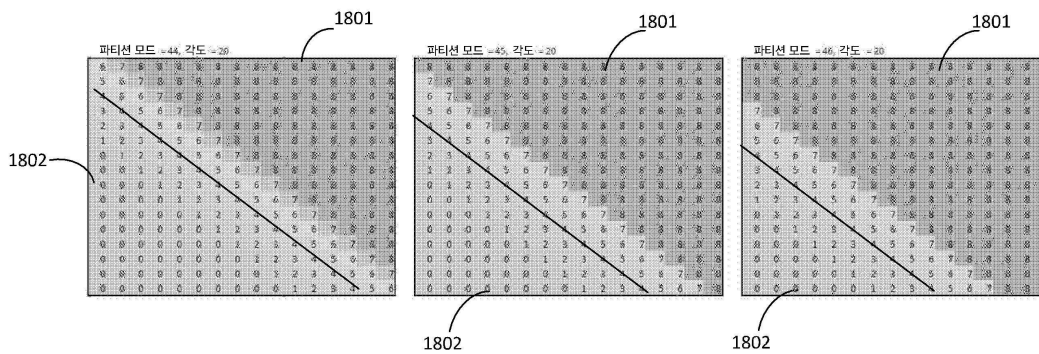
도면18l



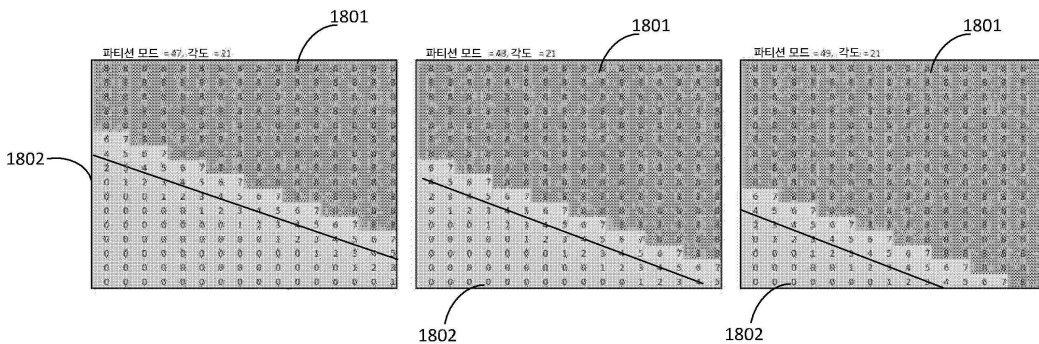
도면18m



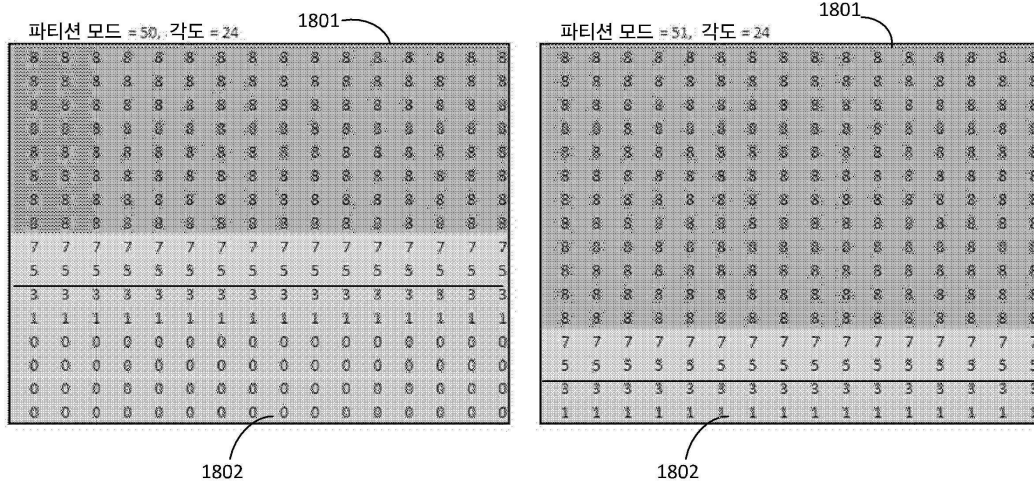
도면18n



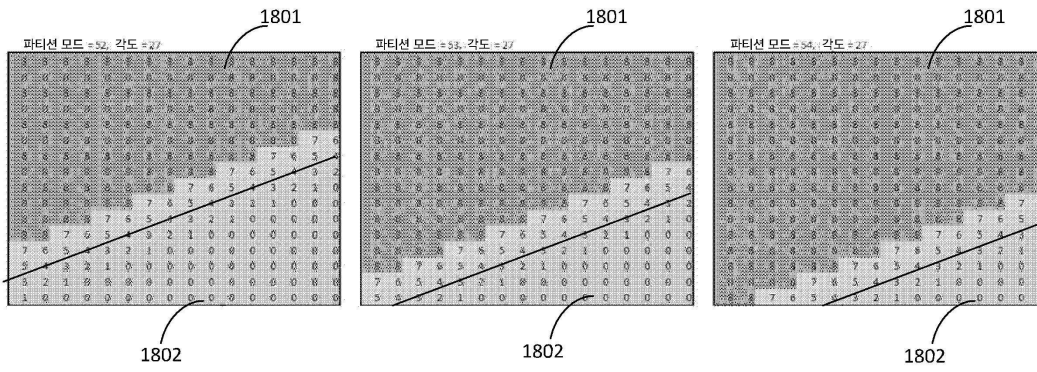
도면18o



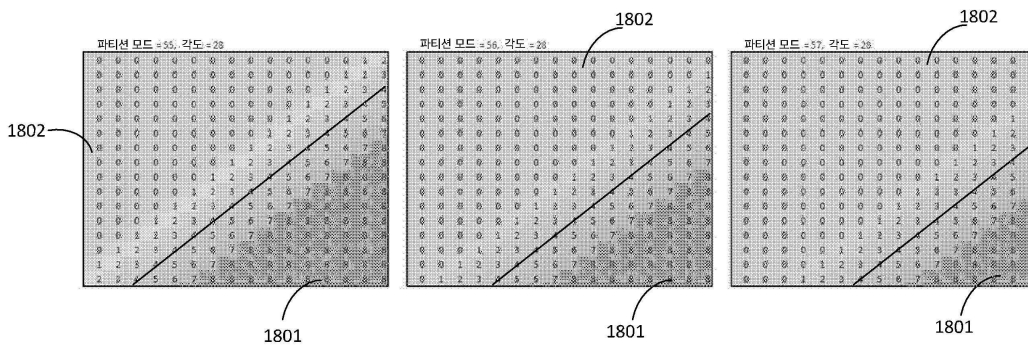
도면18p



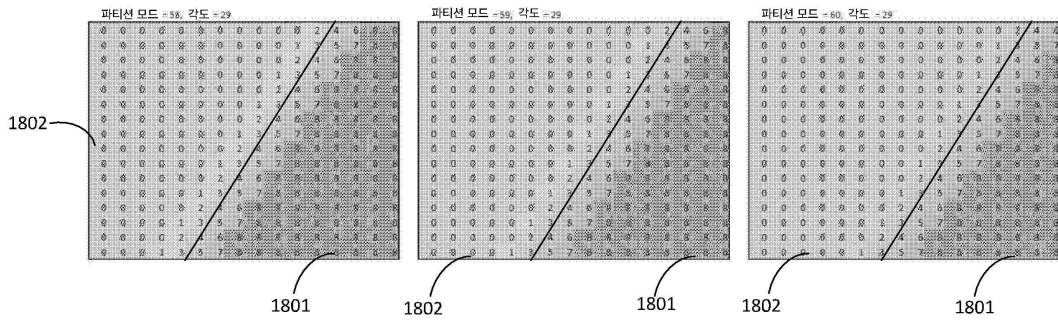
도면18q



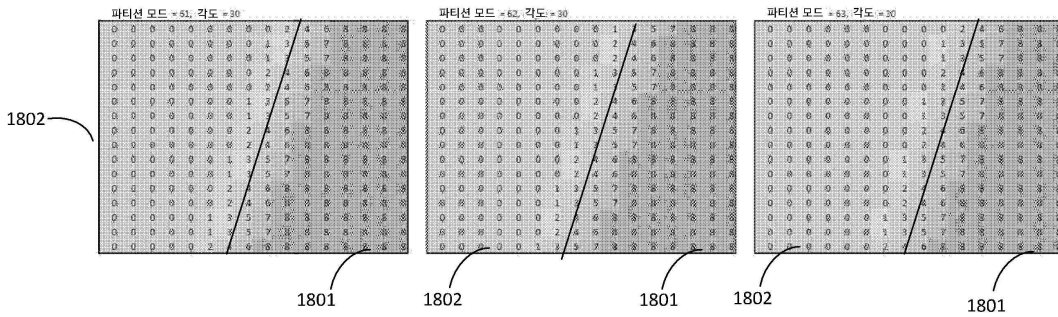
도면18r



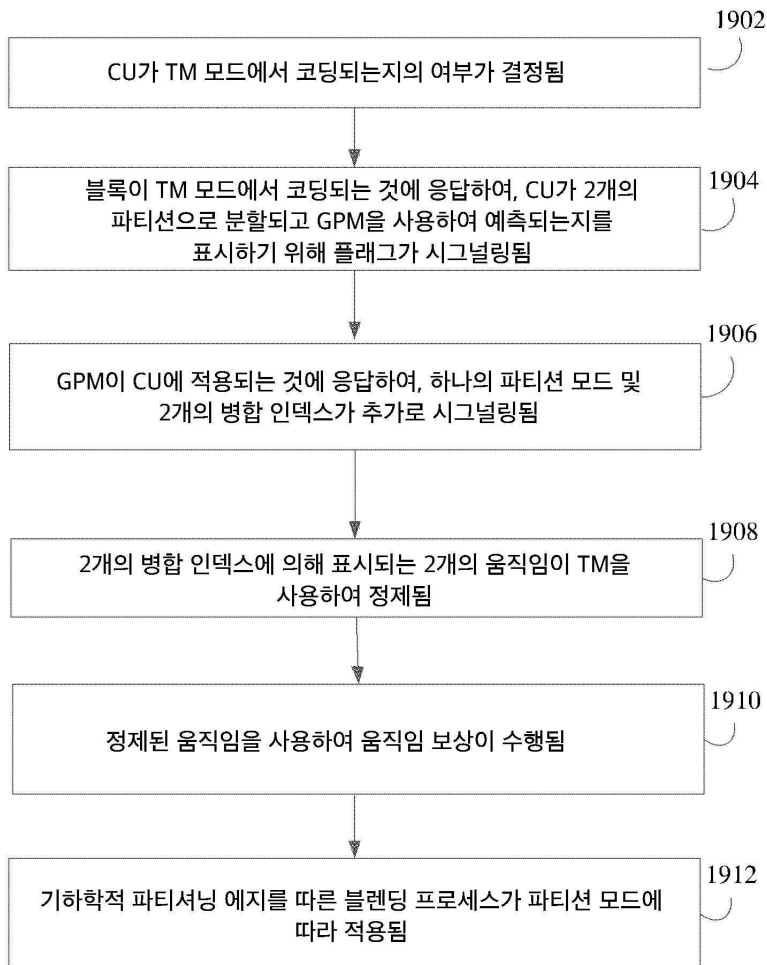
도면18s



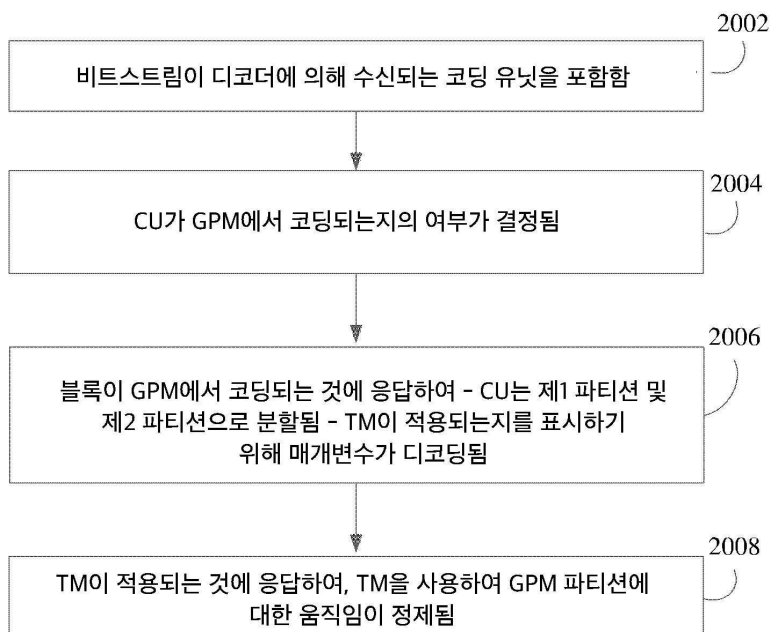
도면18t



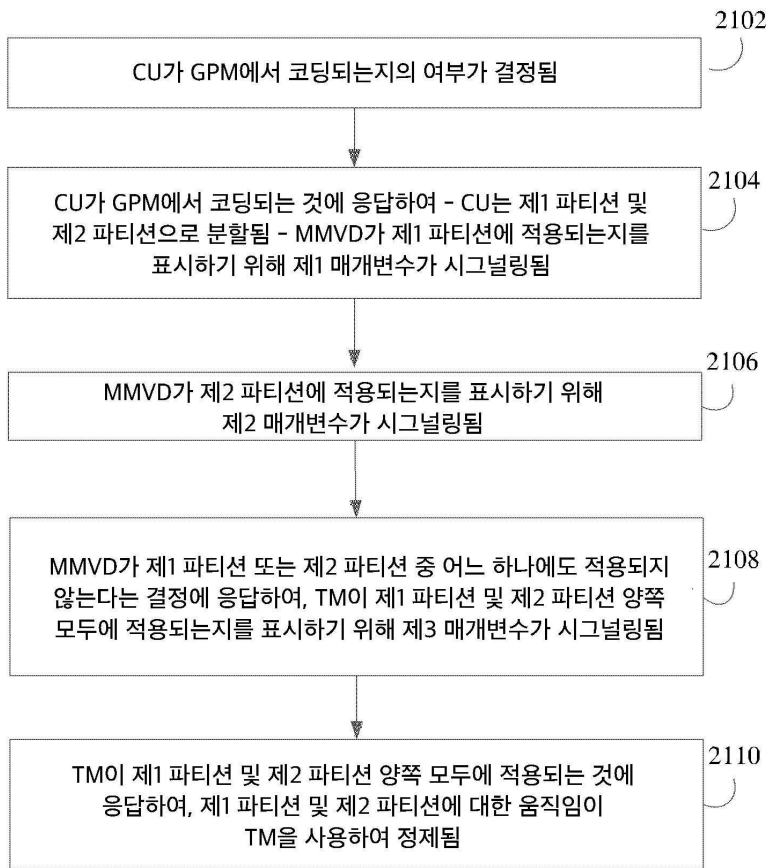
도면19



도면20



도면21



도면22

