



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2024년12월04일  
(11) 등록번호 10-2738552  
(24) 등록일자 2024년12월02일

- (51) 국제특허분류(Int. Cl.)  
H04N 19/593 (2014.01) H04N 19/105 (2014.01)  
H04N 19/119 (2014.01) H04N 19/436 (2014.01)
- (52) CPC특허분류  
H04N 19/593 (2015.01)  
H04N 19/105 (2015.01)
- (21) 출원번호 10-2021-7042534
- (22) 출원일자(국제) 2020년05월29일  
심사청구일자 2022년01월05일
- (85) 번역문제출일자 2021년12월24일
- (65) 공개번호 10-2022-0003631
- (43) 공개일자 2022년01월10일
- (86) 국제출원번호 PCT/US2020/035363
- (87) 국제공개번호 WO 2020/243596  
국제공개일자 2020년12월03일
- (30) 우선권주장  
62/854,746 2019년05월30일 미국(US)  
16/886,641 2020년05월28일 미국(US)
- (56) 선행기술조사문헌  
KR1020140049098 A  
KR1020130064781 A  
US20150319451 A1  
Hui Yong Kim, CU-based Merge Candidate List,  
JCT-VC of ITU-T and ISO/IEC,  
JCTVC-G416(2011.11.21.)
- (73) 특허권자  
홀루, 엘엘씨  
미국 캘리포니아 90404 산타모니카 스위트 200  
2500 브로드웨이 스트리트
- (72) 발명자  
장 원하오  
미국 90404 캘리포니아주 산타 모니카 브로드웨이  
2500 스위트 200  
추이 정  
미국 90404 캘리포니아주 산타 모니카 브로드웨이  
2500 스위트 200
- (74) 대리인  
특허법인코리아나

전체 청구항 수 : 총 20 항

심사관 : 황수진

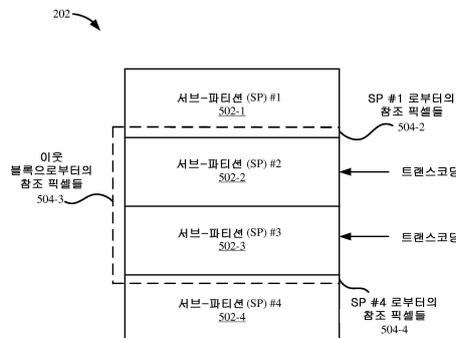
(54) 발명의 명칭 **서브-파티션들의 병렬 양방향 인트라-코딩**

(57) 요약

일부 실시형태들에서, 방법은 코딩될 현재 블록에 대한 제 1 세트의 참조 픽셀들을 결정한다. 현재 블록은 복수의 서브-파티션들로 스플리팅된다. 방법은 제 1 세트의 참조 픽셀들을 사용하여 복수의 서브-파티션들에서의 제 1 서브-파티션을, 그리고 제 1 세트의 참조 픽셀들을 사용하여 복수의 서브-파티션들에서의 제 2 서브-

(뒷면에 계속)

대표도 - 도5b



파티션을 병렬로 프로세싱한다. 그 다음, 방법은 제 1 서브-파티션의 프로세싱으로부터 결정된 제 2 세트의 참조 픽셀들에 기초하여 복수의 서브-파티션들에서의 제 3 서브-파티션을 프로세싱하고 제 2 서브-파티션의 프로세싱으로부터 결정된 제 3 세트의 참조 픽셀들에 기초하여 복수의 서브-파티션들에서의 제 4 서브-파티션을 프로세싱한다.

(52) CPC특허분류

*H04N 19/119* (2015.01)

*H04N 19/436* (2015.01)

---

## 명세서

### 청구범위

#### 청구항 1

방법으로서,

컴퓨팅 디바이스에 의해, 코딩될 현재 블록에 대한 제 1 세트의 참조 픽셀들을 결정하는 단계로서, 상기 현재 블록은 복수의 서브-파티션들로 스플리팅되는, 상기 제 1 세트의 참조 픽셀들을 결정하는 단계;

상기 컴퓨팅 디바이스에 의해, 상기 제 1 세트의 참조 픽셀들을 사용하여 상기 복수의 서브-파티션들에서의 제 1 서브-파티션 및 제 2 서브-파티션을 병렬로 프로세싱하는 단계;

상기 컴퓨팅 디바이스에 의해, 상기 제 1 서브-파티션으로부터 제 2 세트의 참조 픽셀들 및 상기 제 2 서브-파티션으로부터 제 3 세트의 참조 픽셀들을 생성하는 단계; 및

상기 컴퓨팅 디바이스에 의해, 상기 제 1 세트의 참조 픽셀들의 적어도 부분, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 상기 복수의 서브-파티션들에서의 제 3 서브-파티션을 프로세싱하고 상기 제 1 세트의 참조 픽셀들의 적어도 부분, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 상기 복수의 서브-파티션들에서의 제 4 서브-파티션을 프로세싱하는 단계를 포함하는, 방법.

#### 청구항 2

제 1 항에 있어서,

상기 제 3 서브-파티션 및 상기 제 4 서브-파티션을 프로세싱하는 단계는,

상기 제 1 세트의 참조 픽셀들, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나로부터의 제 1 참조 픽셀을 사용하는 단계; 및

상기 제 1 참조 픽셀에 의해 사용되지 않는 상기 제 1 세트의 참조 픽셀들, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나로부터의 제 2 참조 픽셀을 사용하는 단계를 포함하는, 방법.

#### 청구항 3

제 2 항에 있어서,

상기 제 3 서브-파티션 및 상기 제 4 서브-파티션을 프로세싱하는 단계는,

상기 제 1 참조 픽셀에 대해 제 1 가중치를 사용하고 상기 제 2 참조 픽셀에 대해 제 2 가중치를 사용하여 상기 현재 블록에서 코딩될 픽셀에 대한 예측된 픽셀 값을 결정하는 단계를 포함하고, 상기 제 1 가중치 및 상기 제 2 가중치는 상이한 값들인, 방법.

#### 청구항 4

제 3 항에 있어서,

상기 제 3 서브-파티션 및 상기 제 4 서브-파티션을 프로세싱하는 단계는,

상기 제 1 참조 픽셀로부터 상기 픽셀까지의 제 1 거리를 계산하는 단계;

상기 제 2 참조 픽셀로부터 상기 픽셀까지의 제 2 거리를 계산하는 단계; 및

상기 제 1 거리 및 상기 제 2 거리를 사용하여 상기 제 1 가중치 및 상기 제 2 가중치를 계산하는 단계를 포함하는, 방법.

#### 청구항 5

제 1 항에 있어서,

상기 제 3 서브-파티션 및 상기 제 4 서브-파티션을 프로세싱하는 단계는,

상기 제 1 세트의 참조 픽셀들, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나로부터의 단일 참조 픽셀을 사용하는 단계를 포함하는, 방법.

**청구항 6**

제 1 항에 있어서,

상기 제 1 세트의 참조 픽셀들은 프레임에서 상기 현재 블록에 대한 하나 이상의 이웃 블록들로부터의 픽셀들을 포함하는, 방법.

**청구항 7**

제 1 항에 있어서,

상기 제 1 서브-파티션은 상기 현재 블록의 에지를 따라 제 1 위치에 위치되고,

상기 제 2 서브-파티션은 상기 제 1 서브-파티션 바로 옆에 위치되지 않는, 방법.

**청구항 8**

제 7 항에 있어서,

상기 제 3 서브-파티션 및 상기 제 4 서브-파티션은 상기 제 1 서브-파티션과 상기 제 2 서브-파티션 사이에 위치되는, 방법.

**청구항 9**

제 1 항에 있어서,

상기 제 1 세트의 참조 픽셀들의 상기 적어도 부분은 상기 제 2 서브-파티션 및 상기 제 3 서브-파티션 바로 옆에 위치되고,

상기 제 1 세트의 참조 픽셀들의 상기 적어도 부분은 상기 현재 블록에 대한 이웃 블록에 있는, 방법.

**청구항 10**

제 1 항에 있어서,

상기 제 1 세트의 참조 픽셀들을 사용하는 제 1 트랜스코딩 프로세스를 사용하여 상기 제 1 서브-파티션을 프로세싱하는 단계; 및

상기 제 1 서브-파티션의 프로세싱과 병렬로 상기 제 1 세트의 참조 픽셀들을 사용하는 제 2 트랜스코딩 프로세스를 사용하여 상기 제 2 서브-파티션을 프로세싱하는 단계를 더 포함하는, 방법.

**청구항 11**

제 10 항에 있어서,

상기 제 1 트랜스코딩 프로세스로부터의 상기 제 2 세트의 참조 픽셀들을 상기 제 4 서브-파티션을 프로세싱하는데 있어서 사용하기 위해 상기 제 2 트랜스코딩 프로세스에 제공하는 단계, 및

상기 제 2 트랜스코딩 프로세스로부터의 상기 제 3 세트의 참조 픽셀들을 상기 제 3 서브-파티션을 프로세싱하는데 있어서 사용하기 위해 상기 제 1 트랜스코딩 프로세스에 제공하는 단계를 더 포함하는, 방법.

**청구항 12**

제 1 항에 있어서,

상기 제 2 세트의 참조 픽셀들은 상기 제 1 서브-파티션의 디코딩된 픽셀들로부터 생성되고,

상기 제 3 세트의 참조 픽셀들은 상기 제 2 서브-파티션의 디코딩된 픽셀들로부터 생성되는, 방법.

**청구항 13**

제 1 항에 있어서,  
 상기 제 2 세트의 참조 픽셀들은 상기 제 1 서브-파티션의 제 1 잔차로부터 생성되고,  
 상기 제 3 세트의 참조 픽셀들은 상기 제 2 서브-파티션의 제 2 잔차로부터 생성되는, 방법.

**청구항 14**

제 1 항에 있어서,  
 상기 제 1 세트의 참조 픽셀들을 사용하여 상기 제 1 서브-파티션을, 그리고 상기 제 1 세트의 참조 픽셀들을 사용하여 상기 제 2 서브-파티션을 병렬로 프로세싱하는 단계는,  
 상기 제 1 세트의 참조 픽셀들로부터 제 1 예측된 픽셀들을 생성하는 단계;  
 상기 제 1 예측된 픽셀들을 사용하여 상기 제 1 서브-파티션을 트랜스코딩하는 단계;  
 상기 제 2 세트의 참조 픽셀들로부터 제 2 예측된 픽셀들을 생성하는 단계; 및  
 상기 제 2 예측된 픽셀들을 사용하여 상기 제 2 서브-파티션을 트랜스코딩하는 단계를 포함하는, 방법.

**청구항 15**

제 14 항에 있어서,  
 상기 제 1 세트의 참조 픽셀들의 적어도 부분, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 상기 복수의 서브-파티션들에서의 제 3 서브-파티션을 프로세싱하고 상기 제 1 세트의 참조 픽셀들의 적어도 부분, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 상기 복수의 서브-파티션들에서의 제 4 서브-파티션을 프로세싱하는 단계는,  
 상기 제 1 세트의 참조 픽셀들의 적어도 부분, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 제 3 예측된 픽셀들을 생성하는 단계;  
 상기 제 3 예측된 픽셀들을 사용하여 상기 제 3 서브-파티션을 트랜스코딩하는 단계;  
 상기 제 1 세트의 참조 픽셀들의 적어도 부분, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 제 4 예측된 픽셀들을 생성하는 단계; 및  
 상기 제 4 예측된 픽셀들을 사용하여 상기 제 4 서브-파티션을 트랜스코딩하는 단계를 포함하는, 방법.

**청구항 16**

제 15 항에 있어서,  
 상기 제 2 세트의 참조 픽셀들은 상기 제 1 서브-파티션의 제 1 디코딩된 픽셀들 및 상기 제 1 예측된 픽셀들로부터 생성되고,  
 상기 제 3 세트의 참조 픽셀들은 상기 제 2 서브-파티션의 제 2 디코딩된 픽셀들 및 상기 제 2 예측된 픽셀들로부터 생성되는, 방법.

**청구항 17**

제 1 항에 있어서,  
 상기 제 1 세트의 참조 픽셀들을 사용하여 상기 제 1 서브-파티션을, 그리고 상기 제 1 세트의 참조 픽셀들을 사용하여 상기 제 2 서브-파티션을 병렬로 프로세싱하는 단계는,  
 상기 제 1 서브-파티션을 디코딩하는 단계;  
 상기 제 1 세트의 참조 픽셀들로부터 제 1 예측된 픽셀들을 생성하는 단계;  
 디코딩된 상기 제 1 서브-파티션 및 상기 제 1 예측된 픽셀들로부터 상기 제 2 세트의 참조 픽셀들을 생성하는 단계;

상기 제 2 서브-파티션을 디코딩하는 단계;

상기 제 2 세트의 참조 픽셀들로부터 제 2 예측된 픽셀들을 생성하는 단계; 및

디코딩된 상기 제 2 서브-파티션 및 상기 제 2 예측된 픽셀들로부터 상기 제 3 세트의 참조 픽셀들을 생성하는 단계를 포함하는, 방법.

**청구항 18**

명령들을 포함하는 비밀시적 컴퓨터 판독가능 저장 매체로서,

상기 명령들은, 실행될 경우,

코딩될 현재 블록에 대한 제 1 세트의 참조 픽셀들을 결정하는 것으로서, 상기 현재 블록은 복수의 서브-파티션들로 스플리팅되는, 상기 제 1 세트의 참조 픽셀들을 결정하고;

상기 제 1 세트의 참조 픽셀들을 사용하여 상기 복수의 서브-파티션들에서의 제 1 서브-파티션 및 제 2 서브-파티션을 병렬로 프로세싱하고;

상기 제 1 서브-파티션으로부터 제 2 세트의 참조 픽셀들 및 상기 제 2 서브-파티션으로부터 제 3 세트의 참조 픽셀들을 생성하고;

상기 제 1 세트의 참조 픽셀들의 적어도 부분, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 상기 복수의 서브-파티션들에서의 제 3 서브-파티션을 프로세싱하고 상기 제 1 세트의 참조 픽셀들의 적어도 부분, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 상기 복수의 서브-파티션들에서의 제 4 서브-파티션을 프로세싱하기

위해 동작가능하도록 컴퓨터 시스템을 제어하는, 비밀시적 컴퓨터 판독가능 저장 매체.

**청구항 19**

제 18 항에 있어서,

상기 제 3 서브-파티션 및 상기 제 4 서브-파티션을 프로세싱하는 것은,

상기 제 1 세트의 참조 픽셀들, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나로부터의 제 1 참조 픽셀을 사용하는 것; 및

상기 제 1 참조 픽셀에 의해 사용되지 않는 상기 제 1 세트의 참조 픽셀들, 상기 제 2 세트의 참조 픽셀들, 및 상기 제 3 세트의 참조 픽셀들 중 하나로부터의 제 2 참조 픽셀을 사용하는 것을 포함하는, 비밀시적 컴퓨터 판독가능 저장 매체.

**청구항 20**

장치로서,

하나 이상의 컴퓨터 프로세서들; 및

명령들을 포함하는 비밀시적 컴퓨터 판독가능 저장 매체를 포함하고,

상기 명령들은, 실행될 경우,

코딩될 현재 블록에 대한 제 1 세트의 참조 픽셀들을 결정하는 것으로서, 상기 현재 블록은 복수의 서브-파티션들로 스플리팅되는, 상기 제 1 세트의 참조 픽셀들을 결정하고;

상기 제 1 세트의 참조 픽셀들을 사용하여 상기 복수의 서브-파티션들에서의 제 1 서브-파티션을, 그리고 상기 제 1 세트의 참조 픽셀들을 사용하여 상기 복수의 서브-파티션들에서의 제 2 서브-파티션을 병렬로 프로세싱하고;

상기 제 1 서브-파티션의 프로세싱으로부터 결정된 제 2 세트의 참조 픽셀들에 기초하여 상기 복수의 서브-파티션들에서의 제 3 서브-파티션을 프로세싱하고 상기 제 2 서브-파티션의 프로세싱으로부터 결정된 제 3 세트의 참조 픽셀들에 기초하여 상기 복수의 서브-파티션들에서의 제 4 서브-파티션을 프로세싱하기

위해 동작가능하도록 상기 하나 이상의 컴퓨터 프로세서들을 제어하는, 장치.

**발명의 설명**

**기술 분야**

[0001] **관련 출원들에 상호 참조**

[0002] 본 개시는, 2019년 5월 30일자로 출원된 미국 가출원 제62/854,736호에 대한 우선권을 주장하는, "PARALLEL BI-DIRECTIONAL INTRA-CODING OF SUB-PARTITIONS" 를 명칭으로 하여 2020년 5월 28일자로 출원된 미국 정규출원 제16/886,641호에 대한 우선권을 주장하고, 이들 모두는 모든 목적들을 위해 전부 참조로 본 명세서에 통합된다.

**배경 기술**

[0003] 비디오 코딩에서, 비디오 콘텐츠는, 예측, 변환, 양자화, 및 엔트로피 코딩을 포함한, 프로세스들의 세트를 통해 압축된다. 인트라-예측에서는, 비디오 프레임에서 인접한 픽셀들 간의 공간 리던던시가 감소된다. 예를 들어, 비디오 프레임의 원래 픽셀들은 인코딩되지 않고 디코더로 송신될 수도 있다. 오히려, 예측된 픽셀들이 원래 픽셀들로부터 감소되어 잔차가 형성될 수도 있다. 잔차는 인코딩하여 디코더로 송신할 원래 픽셀들보다 적은 양의 정보이다. 이것은 인코딩된 비트스트림을 송신하는데 사용된 대역폭을 감소시킨다.

**도면의 간단한 설명**

- [0004] 도 1 은 일부 실시형태들에 따른 인트라 예측을 수행하기 위한 단순화된 시스템을 도시한다.
- 도 2 는 일부 실시형태들에 따른 인트라 예측을 위한 프레임의 예를 도시한다.
- 도 3 은 일부 실시형태들에 따른 트랜스코더의 예를 도시한다.
- 도 4 는 일부 실시형태들에 따른 디코더의 보다 상세화된 예를 도시한다.
- 도 5a 는 일부 실시형태들에 따른 현재 블록에서의 서브-파티션들 #1 및 #4 를 트랜스코딩하는 예를 도시한다.
- 도 5b 는 일부 실시형태들에 따른 현재 블록에서의 서브-파티션들 #2 및 #3 을 트랜스코딩하는 예를 도시한다.
- 도 6a 는 일부 실시형태들에 따른 제 1 인트라 예측 방향에 대한 제 1 시나리오를 도시한다.
- 도 6b 는 일부 실시형태들에 따른 제 2 인트라 예측 방향에 대한 제 2 시나리오를 도시한다.
- 도 6c 는 일부 실시형태들에 따른 제 3 인트라 예측 방향에 대한 제 3 시나리오를 도시한다.
- 도 7a 는 일부 실시형태들에 따른 제 4 인트라 예측 방향에 대한 제 4 시나리오를 도시한다.
- 도 7b 는 일부 실시형태들에 따른 제 5 인트라 예측 방향에 대한 제 5 시나리오를 도시한다.
- 도 7c 는 일부 실시형태들에 따른 제 6 인트라 예측 방향에 대한 제 4 시나리오를 도시한다.
- 도 8 은 일부 실시형태들에 따른 병렬의 서브-파티션들의 프로세싱의 예를 도시한다.
- 도 9 는 일부 실시형태들에 따른 블록의 트랜스코딩의 보다 상세화된 플로우차트를 도시한다.
- 도 10 은 일부 실시형태들에 따른 블록의 디코딩의 보다 상세화된 플로우차트를 도시한다.
- 도 11 은 일부 실시형태들에 따른 인코딩 시스템의 예를 도시한다.
- 도 12 는 일부 실시형태들에 따른 디코딩 시스템의 예를 도시한다.

**발명을 실시하기 위한 구체적인 내용**

[0005] 본 명세서에서는 비디오 코딩 시스템에 대한 기법들이 설명된다. 다음의 설명에 있어서, 설명의 목적으로, 다수의 예들 및 특정 상세들이 일부 실시형태들의 철저한 이해를 제공하기 위해 기재된다. 청구항들에 의해 정의된 바와 같은 일부 실시형태들은 이들 예들에서의 특징들 중 일부 또는 전부를, 단독으로 또는 이하 설명되는 다른 특징들과 조합하여 포함할 수도 있고, 본 명세서에서 설명된 특징들 및 개념들의 수정들 및 등가물들을 더 포함할 수도 있다.

[0006] 인트라-예측에서, 트랜스코더는 비디오 프레임의 원래 픽셀들로부터 예측된 픽셀들을 감산하여 트랜스코딩되어야 하는 값들의 양을 감소시킬 수도 있다. 인트라-예측 코딩 모드 중 하나의 방법은, 인트라-코딩 블록을 다중 서브-파티션들로 스플리팅하고 그 서브-파티션들을 별도로 프로세싱하는 것이다. 종래에, 서브-파티션들은, 하나의 서브-파티션이 다른 서브-파티션에 대한 트랜스코딩 프로세스를 시작하기 전에 트랜스코딩될 때, 또는 하나의 서브-파티션이 다른 서브-파티션에 대한 디코딩 프로세스를 시작하기 전에 디코딩될 때 직렬로 프로세싱된다. 그러나, 일부 실시형태들은 서브-파티션들의 그룹들을 병렬로 프로세싱한다. 이것은 서브-파티션들을 직렬로 코딩하는 것에 비해 적어도 2 배만큼 코딩 프로세스의 속도를 향상시킨다. 서브-파티션을 병렬로 프로세싱할 경우, 2 개의 서브-파티션들로부터의 참조 픽셀들은 다음 2 개의 서브-파티션들을 프로세싱하는데 사용될 수도 있다. 또한, 2 개의 참조 픽셀들이 사용되는 경우, 2 개의 참조 픽셀들은, 이를 테면 트랜스코딩 또는 디코딩되는 픽셀로부터의 거리에 기초하여, 픽셀의 값을 예측할 때 상이하게 가중화될 수도 있다.

[0007] 시스템

[0008] 도 1 은 일부 실시형태들에 따른 인트라 예측을 수행하기 위한 단순화된 시스템 (100) 을 도시한다. 시스템 (100) 은, 텔레비전 쇼, 영화, 또는 비디오 클립에 대한 것과 같은, 임의의 타입의 비디오일 수도 있는, 소스 비디오 애셋 (asset) 을 트랜스코딩한다. 소스 비디오는 하나 이상의 비트레이트들과 같은 하나 이상의 포맷들로 트랜스코딩되어야 할 수도 있다. 일부 실시형태들에서, 서버 시스템 (102) 은 인코딩된 비트스트림을 클라이언트 (104) 로 전송한다. 예를 들어, 서버 시스템 (102) 은 플레이백을 위해 비디오를 클라이언트 (104) 로 전송하고 있을 수도 있다.

[0009] 서버 시스템 (102) 은 비디오를 인코딩된 비트스트림으로 트랜스코딩하는 트랜스코더 (106) 를 포함한다. 트랜스코더 (106) 는 중앙 프로세싱 유닛 (CPU) 상에서 구성된 소프트웨어 비디오 프로세서/트랜스코더, 필드 프로그래밍가능 게이트 어레이 (FPGA), 그래픽 프로세싱 유닛 (GPU) 을 갖는 하드웨어 가속 비디오 프로세서/트랜스코더, 및/또는 주문형 집적 회로 (ASIC) 에서 구현된 하드웨어 프로세서/트랜스코더일 수도 있다. 트랜스코딩은 일 디지털 포맷으로부터 다른 디지털 포맷으로의 변환일 수도 있다. 트랜스코딩은 소스 포맷을 디코딩하고 소스 비디오를 다른 디지털 포맷으로 인코딩하거나, 또는 소스 콘텐츠를 특정 해상도, 프레임레이트, 비트레이트, 코덱 등을 갖는 비디오들로 변환하는 것을 수반할 수도 있다. 또한, 인코딩은 아날로그 소스 콘텐츠의 디지털 포맷으로의 변환일 수도 있다. 사용된 바와 같이, 용어 트랜스코딩은 인코딩을 포함할 수도 있다.

[0010] 트랜스코더 (106) 는 인트라 예측을 수행하는 트랜스코더 인트라 예측 블록 (108) 을 포함한다. 인트라-코딩 모드는 비디오 프레임에서 이웃 픽셀들의 공간 리던던시를 제거함으로써 비디오를 압축하기 위해 트랜스코더 예측 및 디코더 측에 의해 사용된다. 전통적으로, 인트라-코딩은 픽셀 사이즈가  $N \times N$  과 동일한 현재 블록을 인트라-코딩할 수도 있다. 일부 실시형태들에서, 현재 블록 내부의 픽셀들은 현재 블록에 대한 인접한 블록들, 이를 테면 현재 블록에 대해 상부, 좌측, 및 대각선 위치에 있는 블록들로부터의 픽셀들을 사용하여 인트라-코딩된다.

[0011] 클라이언트 (104) 는 인코딩된 비트스트림을 디코딩하는 디코더 (112) 를 포함한다. 디코딩 프로세스 동안, 디코더 인트라 예측 블록 (110) 은, 현재 블록에 대한 인접한 블록들, 이를 테면 현재 블록에 대해 상부, 좌측, 및 대각선 위치에 있는 블록들로부터의 픽셀들을 사용하여 블록을 디코딩하기 위해 인트라 예측을 수행한다.

[0012] 도 2 는 일부 실시형태들에 따른 인트라 예측을 위한 프레임 (200) 의 예를 도시한다. 프레임 (200) 은 트랜스코딩 또는 디코딩되고 있는 이미지일 수도 있다. 프레임 (200) 은 다중 블록들을 포함하고 현재 블록 (202) (#16) 이 트랜스코딩 또는 디코딩되고 있다. 현재 블록 (202) 의 상부에 있는 상부 블록 (204) (#10) 및 현재 블록 (202) 의 좌측에 있는 좌측 블록 (206) (#15) 은 이미 디코딩되었다. 좌측 블록 (206) 및 상부 블록 (204) 으로부터의 픽셀들은 현재 블록 (202) 의 콘텐츠를 예측하는데 사용될 수도 있다. 이들 픽셀들은 이미 디코딩되었기 때문에, 그 픽셀들은 현재 블록의 콘텐츠를 예측하는데 사용될 수 있다. 예를 들어, 이웃 픽셀들은 현재 블록의 픽셀들 중 일부와 유사할 수도 있고 현재 블록 (202) 에서 현재 픽셀들의 양호한 예측을 제공할 수도 있다. 블록들 #9, #11, 및 #21 (대각선에 위치됨) 로부터의 픽셀들이 이용가능한 경우 (예컨대, 트랜스코딩된 후 디코딩되었음), 이들 블록들로부터의 픽셀들과 같은, 다른 블록들로부터의 픽셀들이 또한 사용될 수도 있다.

[0013] 트랜스코더

[0014] 도 3 은 일부 실시형태들에 따른 트랜스코더 (106) 의 예를 도시한다. 도 3 은 트랜스코더의 단순화된 버전을 포함하고 보다 상세화된 버전이 이하에 설명될 것이다. 트랜스코더 (106) 는 트랜스코딩되는 현재 블록 (202) 으로부터 원래 픽셀들을 수신한다. 302 에서, 트랜스코더 (106) 는 그 다음, 원래 픽셀들로부터 예측된 픽셀들을 감산하여, 트랜스코딩될 픽셀들을 포함하는 잔차를 생성한다. 예를 들어, 트랜스코딩 프로세스의 일부로서, 변환 및 양자화 블록 (304) 은 잔차 값들에 대해 변환 단계 및 양자화 단계 (T/Q) 를 수행하여 트랜스코딩될 데이터 사이즈를 추가로 압축한다. 마지막으로, 엔트로피 인코딩 블록 (306) 은 데이터를 엔트로피 코딩함으로써 결과의 데이터를 프로세싱한다. 트랜스코더 (106) 는 트랜스코더 (106) 가 디코더 측으로 송신하는 데이터를 인코딩된 비트스트림으로 패키징한다.

[0015] 디코더 측이 트랜스코더 측과 동일한 행동으로 디코딩 프로세스를 수행하고 있음을 보장하기 위해, 트랜스코더 (106) 는 역 양자화 및 역 변환 블록 (308) 및 디코딩 파이프라인을 미러링하는 인트라-예측 블록 (예컨대, 트랜스코더 인트라 예측 블록 (108)) 을 포함한다. 이 루프는 디코딩된 현재 블록 (202) 에 대한 복원된 픽셀들을 생성하고, 그 픽셀들은 인코딩 순서에서 다음 코딩 블록에 대한 참조 픽셀들의 역할을 한다. 예를 들어, 312 에서, 트랜스코더 (106) 는 현재 블록을 트랜스코딩하는데 사용된 예측된 픽셀들을 잔차의 디코딩된 픽셀들과 가산하여 참조 픽셀들을 생성한다.

[0016] 트랜스코더 인트라 예측 블록 (108) 은 트랜스코딩될 다음 현재 블록에 대한 이웃 참조 픽셀 값들을 사용하여 예측된 픽셀 값들을 생성한다. 예를 들어, 트랜스코더 인트라 예측 블록 (108) 은 참조 픽셀 값들을 수신하고 사용된 인트라-모드에 의존하여, 예측된 픽셀 값들을 생성한다. 인트라-예측 모드는 예측의 방향을 포함할 수도 있고, 상이한 참조 픽셀들이 사용된 방향에 기초하여 사용된다. 즉, 수평 인트라-예측 방향은 현재 블록의 좌측에 있는 참조 픽셀들을 사용할 수도 있고, 수직 인트라-예측 방향은 현재 블록의 상부에 있는 참조 픽셀들을 사용할 수도 있다. 방향들은 또한 대각선 방향들을 포함할 수도 있다. 트랜스코더 인트라 예측 블록 (108) 은 참조 픽셀들 및 예측의 방향으로부터 예측된 픽셀들을 생성할 수도 있다. 트랜스코더 인트라 예측 블록 (108) 은 예측의 방향을 상이한 방식들로 결정할 수도 있다. 일부 실시형태들에서, 트랜스코더 인트라 예측 블록 (108) 은 모든 가능한 인트라 예측 방향들을 검사하고, 레이트-왜곡 최적화 메트릭과 같은 규칙들에 기초하여 최상의 인트라 예측 방향을 선택할 수도 있다. 예측된 픽셀들이 사용될 수도 있는데, 일부 방향들이 참조 픽셀 값들의 일부 보간을 필요로 할 수도 있기 때문이다; 예를 들어, 대각선 방향은 하나 이상의 참조 픽셀들로부터 픽셀 값들을 보간할 수도 있다.

[0017] 디코더

[0018] 도 4 는 일부 실시형태들에 따른 디코더 (112) 의 보다 상세화된 예를 도시한다. 도 4 는 디코더의 단순화된 버전을 포함하고, 보다 상세화된 버전이 이하에 설명될 것이다. 디코더 측에서, 디코더 (112) 는 인코딩된 비트스트림을 수신하고 엔트로피 디코딩 블록 (402) 이 엔트로피 디코딩을 수행한다. 그 다음, 역 양자화 및 역 변환 블록 (404) 은 역 양자화 단계 및 역 변환 단계  $(Q^{-1}T^{-1})$  를 수행한다. 역 양자화 및 역 변환 단계들의 출력은 잔차이다. 408 에서, 디코더 (112) 는 그 다음, 잔차를, 트랜스코더 (106) 에서 잔차를 생성하는데 사용되었던 예측된 픽셀들과 결합하여 디코딩된 픽셀들을 생성한다. 디코딩된 픽셀들은 원래 픽셀들의 복원된 표현이다.

[0019] 디코더 인트라-예측 블록 (406) 은 디코딩되는 현재 블록으로부터의 디코딩된 참조 픽셀들과 같은, 참조 픽셀들로부터 예측된 픽셀들을 생성한다. 디코딩된 픽셀들은 디코딩 순서에서 다음 코딩 블록에 대한 참조 픽셀들로서 사용된다. 코딩 순서에서의 다른 코딩 블록은 다른 블록에 대한 인코딩된 비트스트림을 디코딩하기 위해 참조 픽셀들로부터 생성되는 예측된 픽셀들을 사용할 수도 있다.

[0020] 인트라 예측

[0021] 인트라 예측 동안, 트랜스코더 측 및 디코더 측은 인트라-서브-파티션 (ISP) 모드를 사용하여 인트라-서브-파티션링을 수행할 수도 있다. 인트라-서브-파티션링은 인트라-코딩 블록을 다중 서브-파티션들 (SP) 로 스플리팅하고 서브-파티션들을 별도로 트랜스코딩 또는 디코딩한다.

[0022] 도 5a 는 일부 실시형태들에 따른 현재 블록 (202) 에서의 서브-파티션들 #1 및 #4 를 트랜스코딩하는 예를 도시한다. 현재 블록 (202) 은 서브-파티션 #1 (502-1), 서브-파티션 #2 (502-2), 서브-파티션 #3 (502-3), 및 서브-파티션 #4 (502-4) 의 4 개의 서브-파티션들로 스플리팅되었다. 4 개의 서브-파티션들이 설명되지만, 2 개의 서브-파티션들, 6 개의 서브-파티션들, 8 개의 서브-파티션들 등과 같은 다른 수의 서브-파티션들이

사용될 수도 있다. 또한, 서브-파티션들은 수평으로 또는 수직으로 스플리팅될 수도 있다. 도 5a 에서, 현재 블록은 수평으로 스플리팅되었다; 그러나, 현재 블록은 수직으로 또는 다른 패턴들로 스플리팅될 수 있다.

서브-파티션 #1 (502-1) 은 블록의 상부 에지에서 발견될 수도 있고 서브-파티션 #4 (502-4) 는 서브-파티션 #1 (502-1) 바로 옆에 있지 않다. 오히려, 서브-파티션 #2 (502-2) 및 서브-파티션 #3 (502-3) 이 서브-파티션 #1 (502-1) 과 서브-파티션 #4 (502-4) 사이에 있다.

[0023] 트랜스코딩 프로세스에서, 트랜스코더 (106) 는 서브-파티션들 (502) 을 별도로 프로세싱하고, 이는 트랜스코더 (106) 가 서브-파티션 (502-1) 의 픽셀들을 먼저 트랜스코딩한 다음, 다음 서브-파티션 (502-2) 의 픽셀들을 트랜스코딩하는 등입을 의미한다. 상기 논의된 바와 같이, 트랜스코더 (106) 는 참조 픽셀들을 사용하여 각각의 서브-파티션 (502) 을 트랜스코딩한다. 예를 들어, 이미 트랜스코딩된 블록들로부터의 참조 픽셀들은 각각의 서브-파티션 (502) 을 트랜스코딩하는데 사용된다.

[0024] 트랜스코더 (106) 는 트랜스코더가 일부 서브-파티션들을 병렬로 트랜스코딩할 수 있도록 하는 방식으로 각각의 서브-파티션 (502) 에 사용되는 참조 픽셀들을 선택한다. 종래에, 서브-파티션들은 직렬로 트랜스코딩되었을 수도 있다; 예를 들어, 트랜스코더는 먼저 서브-파티션 #1 (502-1) 을 트랜스코딩한 다음, 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들을 사용하여 서브-파티션 #2 (502-2) 를 트랜스코딩하고, 그 다음 서브-파티션 #2 (502-2) 로부터의 참조 픽셀들을 사용하여 서브-파티션 #3 (502-3) 을 트랜스코딩하고, 그 다음 마지막으로 서브-파티션 #3 (502-3) 으로부터의 참조 픽셀들을 사용하여 서브-파티션 #4 (502-4) 를 트랜스코딩한다. 이것은 트랜스코더가 서브-파티션들 (502) 을 직렬로 트랜스코딩하도록 강제한다. 서브-파티션들 (502) 을 직렬로 트랜스코딩하는 대신, 트랜스코더는 서브-파티션들 (502) 의 그룹들을 병렬로 트랜스코딩하며, 이는 현재 블록의 트랜스코딩 시간을 가속화한다. 트랜스코더 (106) 는 서브-파티션들 (502) 의 트랜스코딩을 병렬로 수행할 수 있는 다중 트랜스코딩 프로세스들을 포함할 수도 있거나, 또는 별도의 트랜스코더들이 사용될 수도 있다.

[0025] 서브-파티션 (502) 을 병렬로 트랜스코딩하기 위해, 서브-파티션들 (502) 중 일부에 대한 참조 픽셀들은 서브-파티션들을 직렬로 트랜스코딩하는데 사용되는 참조 픽셀들로부터 변경된다. 예를 들어, 서브-파티션 #1 (502-1) 과 서브-파티션 #4 (502-4) 에 사용되는 참조 픽셀들 (504-1) 은 공유된다. 그 다음, 트랜스코더 (106) 는 서브-파티션 #1 및 서브-파티션 #4 를 병렬로 트랜스코딩한다. 일단, 트랜스코더 (106) 가 서브-파티션 #1 및 서브-파티션 #4 를 트랜스코딩 및 디코딩하면, 서브-파티션 #1 및 서브-파티션 #4 로부터의 참조 픽셀들은 서브-파티션 #2 및 서브-파티션 #3 을 병렬로 트랜스코딩하는데 사용될 수 있다.

[0026] 도 5b 는 일부 실시형태들에 따른 현재 블록 (202) 에서의 서브-파티션들 #2 및 #3 을 트랜스코딩하는 예를 도시한다. 트랜스코더 (106) 는 이웃 블록, 서브-파티션 #1 및/또는 서브-파티션 #4 로부터의 참조 픽셀들을 사용하여 서브-파티션 #2 및 서브-파티션 #3 을 트랜스코딩한다. 예를 들어, 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들 (504-2) 은 서브-파티션 #2 (502-2) 위에서 발견된다. 이웃 블록으로부터의 참조 픽셀들 (504-3) 은 현재 블록 (202) 의 좌측에서 발견되고 서브-파티션들 중 임의의 것 내에 있지 않다. 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들 (504-4) 은 서브-파티션 #3 (502-3) 아래에서 발견된다. 알 수 있는 바와 같이, 서브-파티션 #2 (502-2) 는 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들 (504-2) 과 서브-파티션 #3 (502-3) 사이에 있다. 유사하게, 서브-파티션 #3 (502-3) 은 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들 (504-4) 과 서브-파티션 #2 (502-2) 사이에 있다. 이들 참조 픽셀들을 사용하는 것은 서브-파티션들을 직렬로 프로세싱하는 것과는 상이하다. 예를 들어, 서브-파티션 #1 (504-1) 로부터의 참조 픽셀들은 서브-파티션 #2 (502-2) 로부터의 참조 픽셀들에 비해 서브-파티션 #3 (502-3) 으로부터 더 멀리 떨어져 있다. 유사하게, 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들 (504-4) 은 서브-파티션 #1 (504-1) 로부터의 참조 픽셀들에 비해 서브-파티션 #2 (502-2) 로부터 더 멀리 떨어져 있다. 또한, 이웃 블록으로부터의 참조 픽셀들 (504-3) 은 서브-파티션 #1 (502-2) 및 서브-파티션 #4 (502-4) 에 대한 참조 픽셀들과는 상이할 수도 있다. 예를 들어, 이웃 블록으로부터의 참조 픽셀들 (504-3) 은 서브-파티션 #2 (502-2) 및 서브-파티션 #3 (502-4) 바로 옆에 (예컨대, 좌측에) 위치되고, 서브-파티션 #1 (502-2) 및 서브-파티션 #4 (502-4) 바로 옆에 위치되지 않는다. 다음은 서브-파티션 #2 (502-2) 및 서브-파티션 #3 (502-3) 을 트랜스코딩할 때 서브-파티션 #1 (502-2) 로부터의 참조 픽셀들 및 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들 (504-4) 을 사용하는 인트라 예측 시나리오들을 설명한다.

[0027] 트랜스코딩이 위에서 설명되었지만, 디코더 (112) 는 디코딩 프로세스에서 유사한 방식으로 서브-파티션들을 사용할 수도 있으며, 이는 이하에 더 상세히 설명될 것이다.

[0028] 인트라 예측 시나리오들

[0029] 트랜스코더 (106) 는 다음 시나리오들을 설명하는데 사용될 것이지만, 디코더 (112) 가 유사한 액션들을 수행할 수도 있음이 인식된다. 트랜스코더 (106) 는 서브-파티션들의 그룹들을 병렬로 프로세싱하기 위해 모든 서브-파티션들 간의 데이터 상호의존성 (inter-dependency) 을 분해한다. 그 다음, 트랜스코더 (106) 는 동일한 참조 픽셀들을 사용하여 서브-파티션 #2 및 서브-파티션 #3 에 대해 인트라 예측을 수행할 수도 있다. 예측은 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들 (504-2), 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들 (504-4), 및 이웃 블록으로부터의 참조 픽셀들 (504-3) 의 조합을 사용할 수도 있다. 예를 들어, 예측은 양방향일 수도 있으며, 이는 2 개의 서브-파티션들 또는 서브-파티션 및 이웃 블록으로부터의 픽셀들을 사용한다. 또한, 예측은 하나의 서브-파티션 또는 이웃 블록으로부터의 픽셀을 사용할 수도 있다. 예를 들어, 서브-파티션 #2 또는 서브-파티션 #3 의 영역 내부의 픽셀이 주어지면, 트랜스코더 (106) 는 인트라 예측 방향에 따라 참조 픽셀(들)을 결정한다. 인트라 예측 방향에 의존하여, 트랜스코더 (106) 는 하나 또는 2 개의 참조 픽셀들을 위치시킬 수도 있다. 트랜스코더 (106) 는 서브-파티션 #2 및 서브-파티션 #3 의 최상의 인트라 예측 방향을 함께 결정할 수도 있다. 인트라 예측 방향은 디코더 (112) 로 송신되는 압축된 비트스트림에서 시그널링된다. 구체적으로, 트랜스코더 (106) 는 다중 가능한 인트라 예측 방향들, 예컨대, 수평, 수직, 또는 규칙적인 각도 방향들을 따라 서브-파티션 #2 및 서브-파티션 #3 에 대해 인트라 예측 프로세스를 수행한다. 일부 실시형태들에서, 트랜스코더 (106) 는 각각의 가능한 모드에 대한 레이트 왜곡 값들을 계산할 수도 있고, 그 다음, 최소 레이트 왜곡 값을 갖는 방향이 최적의 방향으로 선택될 것이다. 트랜스코더 (106) 는 서브-파티션 #2 및 서브-파티션 #3 의 모든 픽셀들을 트랜스코딩하기 위해 동일한 인트라 예측 방향을 사용할 수도 있다.

[0030] 트랜스코더 (106) 가 서브-파티션 #2 및 서브-파티션 #3 의 인트라 예측 모드에 대해 상이한 방향들을 선택할 때 다음의 예들이 적용된다. 도 6a 는 일부 실시형태들에 따른 제 1 인트라 예측 방향에 대한 제 1 시나리오를 도시한다. 트랜스코더 (106) 는 서브-파티션 #2 (502-2) 에서의 픽셀 p (604) 를 트랜스코딩하고 있다. 트랜스코더 (106) 는 참조 픽셀들을 선택하기 위해 최적으로 결정된 인트라 예측 방향을 사용한다. 상기 논의된 바와 같이, 트랜스코더 인트라 예측 블록 (108) 은 모든 가능한 인트라 예측 방향들을 검사하고 규칙들에 기초하여 최적의 방향 (606) 을 선택하는 것에 기초하여 참조 픽셀들 및 방향을 선택한다. 이 예에서, 참조 픽셀 r0 (602-1) 은 서브-파티션 #1 (502-1) 로부터의 것이고 참조 픽셀 r1 (602-2) 은 서브-파티션 #4 (502-4) 로부터의 것이다. 인트라-예측 방향은 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들 (504-2) 및 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들 (504-4) 과 교차하고, 이웃 블록으로부터의 참조 픽셀들 (504-3) 과는 교차하지 않는다.

[0031] 트랜스코더 인트라 예측 블록 (108) 은 상이한 프로세스들을 사용하여 픽셀 p 의 예측 값을 계산할 수도 있다. 일부 실시형태들에서, 트랜스코더 인트라 예측 블록 (108) 은 참조 픽셀 r0 (602-1) 및 참조 픽셀 r1 (602-2) 로부터의 값들을 사용하여 픽셀 p (604) 에 대한 예측된 값을 계산할 수도 있다. 그 계산은 참조 픽셀 r0 (602-1) 및 참조 픽셀 r1 (602-2) 의 값들에 대한 가중치들에 대해 상이한 값들을 사용할 수도 있으며, 이는 이하에 더 상세히 설명될 것이다. 이제, 가능한 인트라 예측 방향들의 다른 시나리오들이 먼저 논의될 것이다.

[0032] 트랜스코더 (106) 는 서브-파티션 #2 (502-2) 에서의 픽셀 p (604) 를 트랜스코딩하고 있지만, 트랜스코더 (106) 는 서브-파티션 #3 (502-3) 에서의 픽셀을 예측하기 위해 동일한 프로세스를 수행할 수도 있다. 이것은 다음의 시나리오들 모두에 대해 적용가능하다.

[0033] 도 6b 는 일부 실시형태들에 따른 제 2 인트라 예측 방향에 대한 제 2 시나리오를 도시한다. 트랜스코더 (106) 는 서브-파티션 #2 (502-2) 에서의 픽셀 p (604) 를 트랜스코딩하고 있다. 트랜스코더 인트라 예측 블록 (108) 은 최적의 방향 (606) 을 선택한다. 이 예에서, 참조 픽셀 r0 (602-1) 은 서브-파티션 #1 (502-1) 로부터의 것이고 참조 픽셀 r1 (602-2) 은 이웃 블록으로부터의 것이다. 인트라-예측 방향은 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들 (504-2) 및 이웃 블록으로부터의 참조 픽셀들 (504-3) 과 교차하고, 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들 (504-4) 과는 교차하지 않는다. 참조 픽셀 r1 (602-2) 은 서브-파티션 #4 (502-4) 로부터의 참조 픽셀 (504-3) 에 비해 픽셀 p (604) 에 더 가까울 수도 있고, 일부 경우들에서, 픽셀 p (604) 의 더 양호한 예측일 수도 있다.

[0034] 도 6c 는 일부 실시형태들에 따른 제 3 인트라 예측 방향에 대한 제 3 시나리오를 도시한다. 트랜스코더 (106) 는 서브-파티션 #2 (502-2) 에서의 픽셀 p (604) 를 트랜스코딩하고 있다. 트랜스코더 인트라 예측

블록 (108) 은 최적의 방향 (606) 을 선택한다. 이 예에서, 참조 픽셀 r0 (602-1) 은 이웃 블록으로부터의 것이고 참조 픽셀 r1 (602-2) 은 서브-파티션 #4 (502-4) 로부터의 것이다. 인트라-예측 방향은 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들 (504-4) 및 이웃 블록으로부터의 참조 픽셀들 (504-3) 과 교차하고, 서브-파티션 #1 (504-1) 로부터의 참조 픽셀들과는 교차하지 않는다. 참조 픽셀 r0 (602-1) 은 서브-파티션 #1 (502-1) 로부터의 참조 픽셀 (504-2) 에 비해 픽셀 p (604) 에 더 가까울 수도 있고, 일부 경우들에서, 픽셀 p (604) 의 더 양호한 예측일 수도 있다.

[0035] 상기 시나리오들은 2 개의 참조 픽셀들을 사용한 양방향 예측을 사용하였다. 일부 시나리오들은 오직 단일 픽셀만을 사용한 단방향 예측을 사용할 수도 있다. 도 7a 는 일부 실시형태들에 따른 제 4 인트라 예측 방향에 대한 제 4 시나리오를 도시한다. 트랜스코더 (106) 는 서브-파티션 #2 (502-2) 에서의 픽셀 p (604) 를 트랜스코딩하고 있다. 트랜스코더 인트라 예측 블록 (108) 은 최적의 방향 (606) 을 선택한다. 이 예에서, 참조 픽셀 r0 (602-1) 은 이웃 블록으로부터의 것이다. 인트라-예측 방향은 이웃 블록으로부터의 참조 픽셀들 (504-3) 과 교차하고, 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들 (504-4) 및 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들 (504-2) 과는 교차하지 않는다. 참조 픽셀 r0 (602-1) 은 서브-파티션 #4 (502-4) 및 서브-파티션 #1 (502-1) 로부터의 참조 픽셀 (504-2) 에 비해 픽셀 p (604) 에 더 가까울 수도 있고, 일부 경우들에서, 픽셀 p (604) 의 더 양호한 예측일 수도 있다.

[0036] 도 7b 는 일부 실시형태들에 따른 제 5 인트라 예측 방향에 대한 제 5 시나리오를 도시한다. 트랜스코더 (106) 는 서브-파티션 #2 (502-2) 에서의 픽셀 p (604) 를 트랜스코딩하고 있다. 트랜스코더 인트라 예측 블록 (108) 은 최적의 방향 (606) 을 선택한다. 이 예에서, 참조 픽셀 r0 (602-1) 은 서브-파티션 #1 (502-2) 로부터의 것이다. 인트라-예측 방향은 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들 (504-2) 과 교차하고, 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들 (504-4) 및 이웃 블록으로부터의 참조 픽셀들 (504-3) 과는 교차하지는 않는다. 참조 픽셀 r0 (602-1) 은 서브-파티션 #4 (502-4) 및 이웃 블록으로부터의 참조 픽셀 (504-3) 에 비해 픽셀 p (604) 에 더 가까울 수도 있고, 일부 경우들에서, 픽셀 p (604) 의 더 양호한 예측일 수도 있다.

[0037] 도 7c 는 일부 실시형태들에 따른 제 6 인트라 예측 방향에 대한 제 4 시나리오를 도시한다. 트랜스코더 (106) 는 서브-파티션 #3 (502-3) 에서의 픽셀 p (604) 를 트랜스코딩하고 있다. 트랜스코더 인트라 예측 블록 (108) 은 최적의 방향 (606) 을 선택한다. 이 예에서, 참조 픽셀 r0 (602-1) 은 서브-파티션 #4 (502-4) 로부터의 것이다. 인트라-예측 방향은 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들과 교차하고, 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들 (504-2) 및 이웃 블록으로부터의 참조 픽셀들 (504-3) 과는 교차하지 않는다. 참조 픽셀 r0 (602-1) 은 이웃 블록 및 서브-파티션 #1 (502-2) 로부터의 참조 픽셀에 비해 픽셀 p (604) 에 더 가까울 수도 있고, 일부 경우들에서, 픽셀 p (604) 의 더 양호한 예측일 수도 있다.

[0038] 2 개의 참조 픽셀들이 이용가능한 시나리오들의 경우, 트랜스코더 (106) 는 상이한 방법들을 사용하여 픽셀 p 의 예측 값을 계산할 수도 있다. 예를 들어, 트랜스코더 (106) 는 참조 픽셀들에 가중치들을 적용할 수도 있다. 일부 실시형태들에서, 트랜스코더 (106) 는 다음의 계산을 사용할 수도 있으며:

[0039] 
$$p = w_0 r_0 + w_1 r_1,$$

[0040] 여기서 가중치  $w_0$  및 가중치  $w_1$  은 가중 팩터들이고, p 는 픽셀 값이고, r0 및 r1 은 참조 픽셀 값들이다. 여기서, r0 및 r1 의 값들은 참조 픽셀들과 동일할 수 있거나, 또는 참조 픽셀들의 프로세싱된 값들일 수도 있다. 일부 실시형태들에서, 보간, 필터링 및 풀링 프로세스가 압축 효율을 향상시키기 위해 참조 픽셀 값들에 대해 수행된다. 트랜스코더 (106) 는 상이한 방식들로 가중치들  $w_0$  및  $w_1$  의 값을 도출할 수도 있다.

예를 들어, 트랜스코더 (106) 는 고정된 가중 팩터 및 적응적 가중 팩터를 사용할 수도 있다. 고정된 가중 팩터는 가중치들  $w_0$  및  $w_1$  을 고정된 값, 이를 테면 각각 0.5 로 설정할 수도 있다. 또한, 가중치들  $w_0$  및  $w_1$  은 상이한 고정된 값들, 이를 테면 각각 0.45 및 0.55 로 설정될 수도 있다. 상이한 방향들은 또한 상이한 고정된 값들을 가질 수도 있다. 예를 들어, 도 6a 에서, 가중치  $w_0$  은 가중치  $w_1$  보다 더 높은 고정된 값으로 설정될 수도 있는데, 참조 픽셀 r0 이 픽셀 p 에 더 가까울 수도 있기 때문이다.

[0041] 트랜스코더 (106) 는 상이한 방법들에 기초하여, 이를 테면 픽셀 p 로부터의 거리를 사용함으로써 적응적 가중 팩터를 결정할 수도 있다. 예를 들어, 트랜스코더 (106) 는 다음을 사용할 수도 있으며:

$$w_0 = \frac{dist(p,r1)}{dist(r0,r1)}, w_1 = \frac{dist(p,r0)}{dist(r0,r1)}$$

[0042]

[0043]

여기서 dist(A,B) 는 픽셀 A 및 픽셀 B 의 2 개의 픽셀 위치들 사이의 공간 거리를 측정하는 함수이다. 픽셀들 A 및 B 의 위치들이 2 차원 좌표들 (Ax, Ay) 및 (Bx, By) 에 의해 표현될 수 있음을 가정하면, 가능한 거리 계산은 다음일 수 있다:

[0044]

$$dist(A, B) = |Ax - Bx| + |Ay - By|, \text{ 또는}$$

[0045]

$$dist(A, B) = \sqrt{(Ax - Bx)^2 + (Ay - By)^2}.$$

[0046]

거리에 의한 참조 픽셀들의 가중화는 픽셀 p 에 더 가까운 참조 픽셀을 더 멀리 떨어져 있는 참조 픽셀보다 더 크게 가중화할 수도 있다. 더 가까운 참조 픽셀은 픽셀 p 와 유사할 가능성이 더 높을 수도 있고 픽셀 p 의 더 양호한 예측일 수도 있다. 그 다음, 적응적 가중치들을 사용하면, 픽셀 p 와 더 유사할 수도 있는 참조 픽셀은 더 높게 가중화되기 때문에 압축 및 정확도가 향상될 수도 있다. 어느 픽셀들이 픽셀과 더 유사한지를 결정하기 위해 참조 픽셀들의 특성들을 분석하는 것에 의한 것과 같은 다른 방법들이 가중치들을 결정하는데 또한 사용될 수도 있다.

[0047]

단 하나의 참조 픽셀만이 이용가능한 경우들에 대해, 예측 방법은 참조 픽셀에 대한 픽셀 값과 동일한 픽셀 p 에 대한 픽셀 값을 예측할 수도 있다, 예컨대,  $p = r$ . 여기서, r 의 값은 참조 픽셀과 동일할 수 있거나, 또는 참조 픽셀들의 프로세싱된 값들일 수도 있다. 일부 실시형태들에서, 보간, 필터링 및 풀링 프로세스가 압축 효율을 향상시키기 위해 참조 픽셀 값들에 대해 수행된다.

[0048]

서브-파티션 프로세싱

[0049]

상기 방법을 사용하여, 트랜스코더 (106) 및 디코더 (112) 는 4 개의 서브 파티션들을 2 개의 그룹들로 병렬로 프로세싱할 수 있다. 도 8 은 일부 실시형태들에 따른 병렬의 서브-파티션들의 프로세싱의 예를 도시한다.

다음의 프로세싱 파이프라인은 트랜스코딩 프로세스 또는 디코딩 프로세스에서 사용될 수도 있다. 트랜스코딩 프로세스에서, 파이프라인은 서브-파티션에 대한 예측된 픽셀들을 생성한 다음, 서브-파티션의 트랜스코딩 후, 서브-파티션에 대한 디코딩된 픽셀들이 다른 서브-파티션에 대한 참조 픽셀들로서의 사용을 위해 생성된다. 프로세스가 직렬로 수행되면, 서브-파티션에 대한 예측된 픽셀들을 생성한 다음 서브-파티션에 대한 디코딩된 픽셀들을 구축하는 상기의 프로세스는 각각의 서브-파티션에 대해 직렬로 수행된다. 예를 들어, 직렬 프로세스에서, 먼저, 트랜스코더 (106) 는 서브-파티션 #1 (502-1) 에 대한 예측된 픽셀들을 생성하고, 잔차를 생성하고 그 잔차를 트랜스코딩한 다음, 트랜스코더 (106) 는 서브-파티션 #1 (502-1) 에 대한 디코딩된 픽셀들을 구축한다. 트랜스코더 (106) 는 이들 디코딩된 픽셀들을 참조 픽셀들로서 사용하여 서브-파티션 #2 (502-2) 에 대한 예측된 픽셀들을 생성한다. 서브-파티션 #2 (502-2) 를 인트라-코딩한 후, 트랜스코더 (106) 는 서브-파티션 #2 (502-2) 에 대한 디코딩된 픽셀들을 구축한다. 트랜스코더 (106) 는 이들 디코딩된 픽셀들을 사용하여 서브-파티션 #3 (502-3) 에 대한 예측된 픽셀들을 생성하고, 프로세스는 트랜스코더 (106) 가 현재 블록을 트랜스코딩할 때까지 직렬로 계속된다. 따라서, 서브-파티션들 (502) 을 직렬로 프로세싱할 때, 트랜스코더 (106) 는 종래에는 다른 서브-파티션을 트랜스코딩하는 것으로 이동하기 전에 서브-파티션을 트랜스코딩하고 서브-파티션에 대한 픽셀들을 디코딩하는 것을 완료해야 한다.

[0050]

상기 논의된 바와 같이, 트랜스코더 (106) 는 다른 서브-파티션에 대한 트랜스코딩 프로세스를 시작하기 전에 이전 서브-파티션이 트랜스코딩된 다음 디코딩되는 것을 대기할 필요가 없다. 도 8 에서, 현재 블록의 트랜스코딩이 시작될 때, 서브-파티션 #1 (502-1) 및 서브-파티션 #4 (502-4) 에 대한 참조 픽셀들은 동일하고, 트랜스코더 (106) 는 802 에서 서브-파티션 #1 (502-1) 을 그리고 806 에서 서브-파티션 #4 (502-4) 를 병렬로 트랜스코딩하는 것을 시작한다. 예를 들어, 트랜스코딩 프로세스 #1 (800-1) 은 서브-파티션 #1 (502-1) 을 트랜스코딩하고 트랜스코딩 프로세스 #2 (800-2) 는 서브-파티션 #4 (502-4) 를 트랜스코딩한다.

[0051]

802 에서, 트랜스코딩 프로세스 #1 (800-1) 은 현재 블록 (202) 의 상부에 있는 블록 및 현재 블록 (202) 의 좌측에 있는 블록과 같은, 인접한 블록들로부터의 참조 픽셀들을 사용하여 서브-파티션 #1 (502-1) 에 대한 예측된 픽셀들을 생성한다. 그 다음, 트랜스코딩 프로세스 #1 (800-1) 은 서브-파티션 #1 (502-1) 에 대해 트랜스코딩 프로세스를 수행한다. 픽셀들을 트랜스코딩한 후, 804 에서, 트랜스코딩 프로세스 #1 (800-1) 은 그

다음, 서브-파티션 #1 (502-1) 에 대한 픽셀들을 복원 (예컨대, 디코딩) 한다. 병렬로, 806 에서, 트랜스코딩 프로세스 #2 (800-2) 는 서브-파티션 #1 (502-1) 에 사용되었던 동일한 참조 픽셀들을 사용하여 서브-파티션 #4 (502-4) 에 대한 예측된 픽셀들을 생성한다. 그 다음, 트랜스코딩 프로세스 #2 (800-2) 는 서브-파티션 #4 (502-4) 에 대해 트랜스코딩 프로세스를 수행한다. 픽셀들을 트랜스코딩한 후, 808 에서, 트랜스코딩 프로세스 #2 (800-2) 는 그 다음, 서브-파티션 #3 (502-3) 에 대한 픽셀들을 복원 (예컨대, 디코딩) 한다.

[0052] 서브-파티션 #2 (502-2) 는 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들, 서브-파티션 #4 (502) 로부터의, 그리고 좌측에 있는 이웃 블록과 같은, 인접한 블록으로부터의 참조 픽셀들을 사용할 수도 있다. 일부 실시형태들에서, 트랜스코딩 프로세스 #1 (800-1) 은 트랜스코딩 프로세스 #2 (800-2) 로부터 서브-파티션 #4 (502-4) 로부터의 디코딩된 픽셀들을 수신한다. 이러한 방식으로, 트랜스코딩 프로세스 #1 (800-1) 은 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들을 사용할 수 있다. 디코딩된 픽셀들을 수신하는 것이 논의될 때, 상이한 구현들이 사용될 수도 있다. 예를 들어, 각각의 트랜스코딩 프로세스 (800-1 및 800-2) 는 개별의 디코딩된 픽셀들을 메모리에 저장할 수도 있다. 일단, 디코딩된 픽셀들이 메모리에 저장되면, 각각의 트랜스코딩 프로세스 (800-1 및 800-2) 는 서브-파티션 #2 및 서브-파티션 #3 을 트랜스코딩하는 것을 계속할 수 있다.

[0053] 서브-파티션 #1 (502-1) 에 대한 디코딩된 픽셀들을 복원한 후, 트랜스코딩 프로세스 #1 (800-1) 은 필요한 경우, 서브-파티션 #4 (502-4) 에 대한 디코딩된 픽셀들을 수신하기를 대기해야 할 수도 있다. 그 다음, 810 에서, 트랜스코딩 프로세스 #1 (800-1) 은 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들, 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들, 및/또는 좌측에 있는 이웃 블록으로부터의 참조 픽셀들을 사용하여 예측된 픽셀들을 생성할 수 있다. 트랜스코딩 프로세스 #1 (800-1) 은 서브-파티션 #2 (502-2) 에 대해 트랜스코딩 프로세스를 수행한다. 픽셀들을 트랜스코딩한 후, 812 에서, 트랜스코딩 프로세스 #1 (800-1) 은 그 다음, 서브-파티션 #2 (502-2) 에 대한 픽셀들을 복원 (예컨대, 디코딩) 한다.

[0054] 또한, 서브-파티션 #3 (502-3) 은 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들, 서브-파티션 #4 (502-4) 로부터의, 그리고 좌측에 있는 블록과 같은 인접한 블록들로부터의 참조 픽셀들을 사용한다. 일부 실시형태들에서, 트랜스코딩 프로세스 #2 (800-2) 는 서브-파티션 #1 (502-1) 에 대한 트랜스코딩 프로세스 #1 (800-1) 로부터 디코딩된 픽셀들을 수신한다. 이러한 방식으로, 트랜스코딩 프로세스 #2 (800-2) 는 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들을 사용할 수 있다.

[0055] 서브-파티션 #4 (502-4) 에 대한 디코딩된 픽셀들을 복원한 후, 트랜스코딩 프로세스 #2 (800-2) 는 필요한 경우 서브-파티션 #1 (502-1) 에 대한 디코딩된 픽셀들을 수신하기를 대기해야 할 수도 있다. 그 다음, 814 에서, 트랜스코딩 프로세스 #2 (800-2) 는 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들, 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들, 및/또는 좌측에 있는 이웃 블록으로부터의 참조 픽셀들을 사용하여 예측된 픽셀들을 생성할 수 있다. 트랜스코더 (106) 는 서브-파티션 #3 (502-3) 에 대해 트랜스코딩 프로세스를 수행한다. 픽셀들을 트랜스코딩한 후, 818 에서, 트랜스코딩 프로세스 #2 (800-2) 는 그 다음, 서브-파티션 #3 (502-3) 에 대한 픽셀들을 복원 (예컨대, 디코딩) 한다.

[0056] 트랜스코딩 프로세스는, 서브-파티션 #2 (502-2) 에 대한 트랜스코딩된 픽셀들을 복원하고 서브-파티션 #3 (502-3) 에 대한 트랜스코딩된 픽셀들을 복원한 후 완료된다. 따라서, 2 개의 그룹들의 서브-파티션들 (502) 의 병렬 프로세싱으로 인해, 트랜스코더 (106) 는 직렬로 프로세싱을 수행하는 것에 비해 대략 2 배 더 빠르게 트랜스코딩 프로세스를 수행할 수 있다. 2 개의 그룹들이 설명되지만, 트랜스코더 (106) 는 현재 블록 (202) 을 상이한 수의 그룹들로 스플리팅할 수도 있다. 예를 들어, 8 개의 서브-파티션들 (502) 이 사용되는 경우, 2 개의 그룹들이 사용될 수도 있다; 그러나, 트랜스코더는 서브-파티션 (502) 을 4 개의 그룹들로 스플리팅하여 직렬에 비해 4 배 더 빠르게 그리고 2 개의 그룹들을 사용하는 것에 비해 2 배 더 빠르게 프로세스를 수행할 수도 있다.

[0057] 디코딩 프로세스에서, 서브-파티션 (502) 은 또한 병렬로 디코딩될 수도 있다. 예를 들어, 디코더 (112) 가 현재 블록 (202) 을 디코딩하는 것을 시작할 때, 서브-파티션 #1 (502-1) 및 서브-파티션 #4 (502-4) 에 대한 참조 픽셀들은 동일하고, 디코더 (112) 는 서브-파티션 #1 (502-1) 및 서브-파티션 #4 (502-4) 를 병렬로 디코딩하는 것을 시작한다. 디코더 (112) 는 또한 트랜스코더 (106) 와 유사하게 디코딩 프로세스 #1 (801-1) 및 디코딩 프로세스 #2 (801-2) 로서 지칭되는 2 개의 디코딩 프로세스들을 사용할 수도 있다.

[0058] 디코딩 프로세스 #1 (801-1) 은 현재 블록 (202) 의 상부에 있는 블록 및 현재 블록 (202) 의 좌측에 있는 블록 으로부터와 같은, 인접한 블록들로부터의 픽셀들을 사용하여 서브-파티션 #1 (502-1) 에 대한 예측된 픽셀들을 생성한다. 그 다음, 디코딩 프로세스 #1 (801-1) 은 서브-파티션 #1 (502-1) 에 대해 디코딩 프로세스를 수

행하여 잔차를 생성한다. 픽셀들을 디코딩한 후, 디코딩 프로세스 #1 (801-1)은 그 다음, 예측된 픽셀들 및 잔차를 사용하여 서브-파티션 #1 (502-1)에 대한 픽셀들을 복원한다. 병렬로, 디코딩 프로세스 #2 (801-2)는 서브-파티션 #1 (502-1)에 사용되었던 동일한 참조 픽셀들을 사용하여 서브-파티션 #4 (502-4)에 대한 예측된 픽셀들을 생성한다. 그 다음, 디코딩 프로세스 #2 (801-2)는 서브-파티션 #4 (502-4)에 대해 디코딩 프로세스를 수행한다. 픽셀들을 디코딩한 후, 디코딩 프로세스 #2 (801-2)는 그 다음, 예측된 픽셀들 및 잔차를 사용하여 서브-파티션 #4 (502-4)에 대한 참조 픽셀들을 복원한다.

[0059] 서브-파티션 #2 (502-2)는 서브-파티션 #1 (502-1)로부터의 참조 픽셀들, 서브-파티션 #4 (502-4)로부터의 참조 픽셀들, 및 좌측 이웃 블록과 같은 인접한 블록들로부터의 참조 픽셀들을 사용한다. 서브-파티션 #1 (502-1)에 대한 픽셀들을 디코딩한 후, 디코딩 프로세스 #1 (801-1)은 서브-파티션 #1 (502-1)로부터의 참조 픽셀들, 서브-파티션 #4 (502-4)로부터의 참조 픽셀들, 및/또는 좌측에 있는 이웃 블록으로부터의 참조 픽셀들 (504-3)을 사용하여 예측된 픽셀들을 생성할 수 있다. 디코딩 프로세스 #1 (801-1)은 서브-파티션 #2 (502-2)에 대해 디코딩 프로세스를 수행하여 잔차를 생성한다. 픽셀들을 디코딩한 후, 디코딩 프로세스 #1 (801-1)은 그 다음, 예측된 픽셀들 및 잔차를 사용하여 서브-파티션 #2 (502-2)에 대한 참조 픽셀들을 복원한다.

[0060] 서브-파티션 #3 (502-3)은 서브-파티션 #2 (502-2)와 동일한 참조 픽셀들을 사용하고 디코딩 프로세스는 서브-파티션 #2 (502-2)와 병렬로 수행될 수도 있다. 디코딩 프로세스 #2 (801-2)는 서브-파티션 #3 (502-3)에 대해 디코딩 프로세스를 수행하여 잔차를 생성한다. 픽셀들을 디코딩한 후, 디코딩 프로세스 #2 (801-2)는 그 다음, 예측된 픽셀들 및 잔차를 사용하여 서브-파티션 #3 (502-3)에 대한 참조 픽셀들을 복원한다.

[0061] 트랜스코딩 프로세스

[0062] 도 9는 일부 실시형태들에 따른 블록의 트랜스코딩의 보다 상세화된 플로우차트 (900)를 도시한다. 상기 플로우차는 예측된 픽셀들의 생성을 논의하였다. 다음의 플로우차트는 트랜스코딩 프로세스를 더 상세히 설명한다.

[0063] 902에서, 트랜스코딩 프로세스 (800-1)는 서브-파티션 #1 (502-1)에 대한 원래 픽셀들 및 서브-파티션 #1 (502-1)에 대한 예측된 픽셀들을 수신한다. 904에서, 트랜스코딩 프로세스 (800-1)는 원래 픽셀들 및 예측된 픽셀들로부터 서브-파티션 #1에 대한 잔차를 생성한다. 906에서, 트랜스코딩 프로세스 (800-1)는 변환 및 양자화 및 엔트로피 인코딩을 수행함으로써 서브-파티션 #1 (502-1)에 대한 잔차를 트랜스코딩한다. 908에서, 트랜스코딩 프로세스 #1 (800-1)은 잔차를 디코딩하여 서브-파티션 #1 (502-1)에 대한 디코딩된 픽셀들을 생성한다. 910에서, 트랜스코딩 프로세스 #1 (800-1)은 디코딩된 픽셀들을 예측된 픽셀들과 결합하여 서브-파티션 #1 (502-1)에 대한 참조 픽셀들을 생성한다. 911에서, 트랜스코딩 프로세스 #1 (800-1)은 디코딩된 픽셀들을 트랜스코딩 프로세스 #2 (800-1)에 통신한다 (예컨대, 디코딩된 픽셀들을 메모리에 저장함). 912에서, 서브-파티션 #2 (502-2)에 대한 트랜스코딩 프로세스는 902 내지 911에서 설명된 것과 유사한 단계들을 사용하여 수행된다. 그러나, 서브-파티션 #2는 서브-파티션 #1 (502-1)로부터의 참조 픽셀들, 서브-파티션 #4 (502-4)로부터의 참조 픽셀들, 및/또는 좌측에 있는 이웃 블록으로부터의 참조 픽셀들을 사용한다. 서브-파티션 #4 (502-4)로부터의 디코딩된 픽셀들의 수신은 트랜스코딩 프로세스를 완료하는데 필요할 수도 있다.

[0064] 병렬로, 다음이 수행된다. 대응하는 단계들은 정확히 동시에 수행되지 않을 수도 있음에 주목한다. 즉, 902 및 912에서의 단계들은 동시에 시작 및 종료할 필요가 없다. 오히려, 902 내지 912에서의 단계들은 914 내지 924에서의 단계들과 병렬로 수행되며, 여기서 각각의 단계는 자신만의 페이스로 진행될 수도 있다. 그러나, 서브-파티션 #1 (502-1)로부터의 디코딩된 픽셀들의 수신은 서브-파티션 #3 (502-3)으로부터 트랜스코딩 프로세스를 완료하는데 필요하다. 914에서, 트랜스코딩 프로세스 #2 (800-2)는 서브-파티션 #3 (502-3)에 대한 원래 픽셀들 및 서브-파티션 #3 (502-3)에 대한 예측된 픽셀들을 수신한다. 916에서, 트랜스코딩 프로세스 #2 (800-2)는 원래 픽셀들 및 예측된 픽셀들로부터 서브-파티션 #1에 대한 잔차를 생성한다. 918에서, 트랜스코딩 프로세스 #2 (800-2)는 변환 및 양자화 및 엔트로피 인코딩을 수행함으로써 서브-파티션 #3 (502-3)에 대한 잔차를 트랜스코딩한다. 920에서, 트랜스코딩 프로세스 #2 (800-2)는 잔차를 디코딩하여 서브-파티션 #3 (502-3)에 대한 디코딩된 픽셀들을 생성한다. 922에서, 트랜스코딩 프로세스 #2 (800-2)는 디코딩된 픽셀들을 예측된 픽셀들과 결합하여 서브-파티션 #3 (502-3)에 대한 참조 픽셀들을 생성한다. 923에서, 트랜스코딩 프로세스 #2 (800-2)는 디코딩된 픽셀들을 트랜스코딩 프로세스 #1 (800-1)에 통신한다 (예컨대, 디코딩된 픽셀들을 메모리에 저장함). 924에서, 서브-파티션 #3 (502-3)에 대한

트랜스코딩 프로세스는 914 내지 923 에서 설명된 것과 유사한 단계들을 사용하여 수행된다. 그러나, 서브-파티션 #3 은 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들, 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들, 및/또는 좌측에 있는 이웃 블록으로부터의 참조 픽셀들을 사용한다. 서브-파티션 #1 (502-1) 로부터의 디코딩된 픽셀들의 수신은 트랜스코딩 프로세스를 완료하는데 필요할 수도 있다.

[0065] 디코딩 프로세스

[0066] 도 10 은 일부 실시형태들에 따른 블록의 디코딩의 보다 상세화된 플로우차트 (1000) 를 도시한다. 1002 에서, 디코딩 프로세스 #1 (801-1) 은 이전 블록의 참조 픽셀들로부터 예측된 픽셀들을 생성한다. 이전 블록은 현재 블록 (202) 의 상부 및/또는 현재 블록 (202) 의 좌측에 있는 블록일 수도 있다.

[0067] 1004 에서, 디코딩 프로세스 #1 (801-1) 은 서브-파티션 #1 (502-1) 에 대한 인코딩된 픽셀들을 수신한다. 1006 에서, 디코딩 프로세스 #1 (801-1) 은 인코딩된 픽셀들로부터 서브-파티션 #1 에 대한 잔차를 생성한다. 예를 들어, 디코딩 프로세스 #1 (801-1) 은 엔트로피 디코딩 및 역 양자화 및 변환을 수행할 수도 있다. 1008 에서, 디코딩 프로세스 #1 (801-1) 은 예측된 픽셀들 및 잔차로부터 디코딩된 픽셀들을 생성한다. 예를 들어, 디코딩 프로세스 #1 (801-1) 은 디코딩된 픽셀들을 예측된 픽셀들과 결합할 수도 있다.

[0068] 1010 에서, 디코딩 프로세스 #1 (801-1) 은 참조 픽셀들로부터 서브-파티션 #1 에 대한 예측된 픽셀들을 생성한다. 1011 에서, 디코딩 프로세스 #1 (801-1) 은 디코딩된 픽셀들을 트랜스코딩 프로세스 #2 (1002-2) 에 통신한다. 1012 에서, 디코딩 프로세스 #1 (801-1) 은 서브-파티션 #2 에 대해 예측 프로세스를 수행한다. 프로세스는 단계들 (1002-1011) 과 유사할 수도 있다. 그러나, 서브-파티션 #2 는 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들, 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들, 및/또는 좌측에 있는 블록으로부터의 참조 픽셀들을 사용한다. 서브-파티션 #4 (502-4) 로부터의 디코딩된 픽셀들의 수신은 트랜스코딩 프로세스를 완료하는데 필요하다.

[0069] 병렬로, 다음이 수행된다. 대응하는 단계들은 정확히 동시에 수행되지 않을 수도 있음에 주목한다. 즉, 1002 및 1014 에서의 단계들은 동시에 시작 및 종료할 필요가 없다. 오히려, 1002 내지 1012 에서의 단계들은 1014 내지 1024 에서의 단계들과 병렬로 수행되며, 여기서 각각의 단계는 자신만의 페이스로 진행될 수도 있다. 그러나, 서브-파티션 #1 (502-1) 로부터의 디코딩된 픽셀들의 수신은 필요할 수도 있다. 1014 에서, 디코딩 프로세스 #2 (801-2) 는 이전 블록의 참조 픽셀들로부터 예측된 픽셀들을 생성한다. 1016 에서, 디코딩 프로세스 #2 (801-2) 는 서브-파티션 #3 에 대한 인코딩된 픽셀들을 수신한다. 1018 에서, 디코딩 프로세스 #2 (801-2) 는 인코딩된 픽셀들로부터 서브-파티션 #3 에 대한 잔차를 생성한다. 1020 에서, 디코딩 프로세스 #2 (801-2) 는 예측된 픽셀들 및 잔차로부터 디코딩된 픽셀들을 생성한다. 1022 에서, 디코딩 프로세스 #2 (801-2) 는 참조 픽셀들로부터 서브-파티션 #3 에 대한 예측된 픽셀들을 생성한다. 1023 에서, 디코딩 프로세스 #1 (801-1) 은 디코딩된 픽셀들을 트랜스코딩 프로세스 #2 (1002-2) 에 통신한다. 1024 에서, 디코딩 프로세스 #2 (801-2) 는 서브-파티션 #3 에 대해 예측 프로세스를 수행한다. 프로세스는 단계들 (1014-1023) 과 유사할 수도 있다. 그러나, 서브-파티션 #3 은 서브-파티션 #1 (502-1) 로부터의 참조 픽셀들, 서브-파티션 #4 (502-4) 로부터의 참조 픽셀들, 및/또는 좌측에 있는 블록으로부터의 참조 픽셀들을 사용한다. 서브-파티션 #1 (502-1) 로부터의 디코딩된 픽셀들의 수신은 트랜스코딩 프로세스를 완료하는데 필요하다.

[0070] 결론

[0071] 요약하면, 일부 실시형태들은 서브-파티션들 간의 데이터 의존성을 분해하고 프로세스가 병렬로 수행되는 것을 허용한다. 더욱이, 일부 실시형태들은 코딩 블록의 중앙 영역에 있는 픽셀들에 대한 가중화된 양방향 예측 방법을 도입한다. 이것은 예측 정확성을 높이고 비디오의 더 양호한 압축 효율을 초래한다.

[0072] 시스템

[0073] 도 11 은 일부 실시형태들에 따른 인코딩 시스템의 예를 도시한다. 비디오 코덱 프레임워크는 기본 컴포넌트들의 세트: 블록 파티셔닝, 인터 및 인트라 예측, 변환 및 양자화, 및 엔트로피 코딩을 포함한다.

[0074] 트랜스코더 (106) 는, 추후 프로세싱을 위해 먼저 비중첩 코딩 블록들로 스플리팅되는 비디오의 프레임들을 수신한다. 상이한 비디오 콘텐츠 특성들에 대처하기 위해, 복잡한 영역들은 더 작은 사이즈들을 갖는 파티션들에 의해 커버될 것인 한편, 단순한 영역들은 더 큰 파티션들에 의해 커버될 것이다. 다중 블록 패턴들 및 형상들은 양자 모두 함께 사용될 수도 있고, 예를 들어, 쿼드-트리 패턴, 트리플-트리 패턴 및 바이너리-트리

패턴은 모두 함께 사용될 수 있는 한편, 정사각형 블록들 및 직사각형 블록들이 또한 함께 사용될 수 있다.

- [0075] 예측은 비디오 신호의 리던던시를 제거하는데 사용된다. 프로세싱되는 픽셀들로부터 예측된 픽셀 값들을 감산함으로써, 잔차 신호의 진폭이 상당히 감소될 수 있으며, 따라서 결과의 비트스트림 사이즈가 감소될 수 있다. 현재 프레임의 참조 픽셀들을 사용하고 있는 인트라 예측 블록 (1110)은 프레임 내의 공간 리던던시를 감소시키는 것을 목표로 한다. 이웃 프레임들로부터의 참조 픽셀들을 사용하고 있는 인터 예측 블록 (1112)은 프레임들 간의 시간 리던던시를 제거하려고 시도한다. 모션 추정 및 보상 블록 (1116)은 인코더 측에서 인터 예측의 서브-모듈일 수도 있으며, 이는 인접한 프레임들 간의 오브젝트들의 모션 트레이스를 캡처하고 인터 예측을 위한 참조 픽셀들을 생성한다.
- [0076] 변환 및 양자화 블록 (1104)은 인트라 또는 인터 예측 후에 잔차 픽셀들을 사용한다. 변환 및 양자화 블록 (1104)은 주파수 도메인에서 잔차 신호를 나타내는 변환 동작을 수행한다. 인간 시각 시스템이 고 주파수 컴포넌트들보다 비디오 신호의 저 주파수 컴포넌트들에 더 민감하다는 것을 고려하여, 양자화는 고 주파수 신호들에 대한 정밀도를 감소시킴으로써 잔차 신호를 추가로 압축하도록 설계된다.
- [0077] 인코더 (1100)와 디코더 (600) 사이의 비동기 (out-of-sync) 이슈를 회피하기 위해, 인코더 (1100)는 인코더 (1100) 및 디코더 (600) 양자 모두가 동일한 수학적 프로세스들을 사용하고 있음을 확실히 하기 위해 디코딩 모듈들을 포함한다. 따라서, 역 변환 및 역 양자화 블록 (1108)은 디코더 측의 대응하는 블록과 유사하다. 역 변환 및 역 양자화 블록 (1108)은 인트라 및 인터 예측을 사용하여 픽셀들을 복원한다.
- [0078] 인-루프 필터 (1114)는 상기 언급된 프로세스들에 의해 도입되는 임의의 시각적 아티팩트들을 제거한다. 블록킹 아티팩트들, 모스키토 아티팩트들, 컬러 밴딩 효과들 등을 포함하지만 이에 한정되지 않는 상이한 아티팩트들을 감소시키기 위해 캐스케이드 방식으로 복원된 프레임에 대해 다양한 필터링 방법들이 적용된다.
- [0079] 엔트로피 인코딩 블록 (1106)은 모델 기반 방법을 사용하여 비트스트림을 추가로 압축할 수도 있다. 트랜스코더 (106)는 결과의 인코딩된 비트스트림을 네트워크 또는 다른 타입들의 매체를 통해 디코더 (112)로 송신한다.
- [0080] 도 12는 일부 실시형태들에 따른 디코딩 시스템의 예를 도시한다. 디코더 (112)는 인코딩된 비트스트림을 수신하고 그것을 엔트로피 디코딩 블록 (1202)에 입력하여 디코딩 프로세스에 필요한 정보를 복구한다. 상기 언급된 바와 같이, 디코딩된 프레임은 디코딩된 프레임을 구축하기 위해 동일한 방식으로 역 변환 및 역 양자화 블록 (1204), 인트라 예측 블록 (1206) 또는 인터 예측 블록 (1208), 모션 보상 블록 (1210), 및 인-루프 필터링 블록 (1212)을 사용함으로써 디코딩될 수 있다.
- [0081] 예시적인 실시형태들
- [0082] 일부 실시형태들에서, 방법은, 컴퓨팅 디바이스에 의해, 코딩될 현재 블록에 대한 제 1 세트의 참조 픽셀들을 결정하는 단계로서, 현재 블록은 복수의 서브-파티션들로 스플리팅되는, 상기 제 1 세트의 참조 픽셀들을 결정하는 단계; 컴퓨팅 디바이스에 의해, 제 1 세트의 참조 픽셀들을 사용하여 복수의 서브-파티션들에서의 제 1 서브-파티션 및 제 2 서브-파티션을 병렬로 프로세싱하는 단계; 컴퓨팅 디바이스에 의해, 제 1 서브-파티션으로부터 제 2 세트의 참조 픽셀들 및 제 2 서브-파티션으로부터 제 3 세트의 참조 픽셀들을 생성하는 단계; 및 컴퓨팅 디바이스에 의해, 제 1 세트의 참조 픽셀들의 적어도 부분, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 복수의 서브-파티션들에서의 제 3 서브-파티션을 프로세싱하고 제 1 세트의 참조 픽셀들의 적어도 부분, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 복수의 서브-파티션들에서의 제 4 서브-파티션을 프로세싱하는 단계를 포함한다.
- [0083] 일부 실시형태들에서, 제 3 서브-파티션 및 제 4 서브-파티션을 프로세싱하는 단계는, 제 1 세트의 참조 픽셀들, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나로부터의 제 1 참조 픽셀을 사용하는 단계; 및 제 1 참조 픽셀에 의해 사용되지 않는 제 1 세트의 참조 픽셀들, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나로부터의 제 2 참조 픽셀을 사용하는 단계를 포함한다.
- [0084] 일부 실시형태들에서, 제 3 서브-파티션 및 제 4 서브-파티션을 프로세싱하는 단계는, 제 1 참조 픽셀에 대해 제 1 가중치를 사용하고 제 2 참조 픽셀에 대해 제 2 가중치를 사용하여 현재 블록에서 코딩되는 픽셀에 대한 예측된 픽셀 값을 결정하는 단계를 포함하고, 제 1 가중치 및 제 2 가중치는 상이한 값들이다.
- [0085] 일부 실시형태들에서, 제 3 서브-파티션 및 제 4 서브-파티션을 프로세싱하는 단계는, 제 1 참조 픽셀로부터 픽셀까지의 제 1 거리를 계산하는 단계; 제 2 참조 픽셀로부터 픽셀까지의 제 2 거리를 계산하는 단계; 및 제 1

거리 및 제 2 거리를 사용하여 제 1 가중치 및 제 2 가중치를 계산하는 단계를 포함한다.

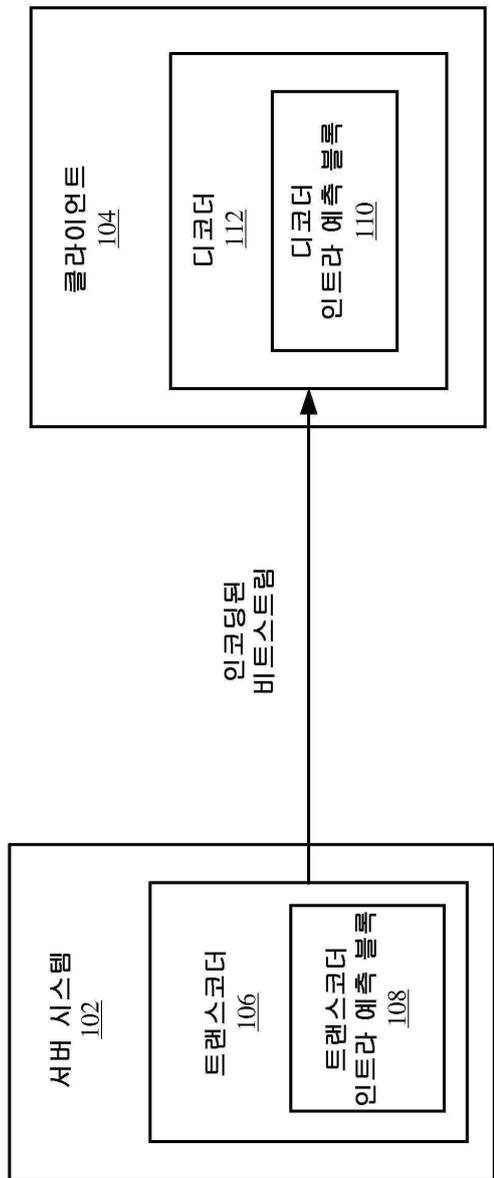
- [0086] 일부 실시형태들에서, 제 3 서브-파티션 및 제 4 서브-파티션을 프로세싱하는 단계는, 제 1 세트의 참조 픽셀들, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나로부터의 단일 참조 픽셀을 사용하는 단계를 포함한다.
- [0087] 일부 실시형태들에서, 제 1 세트의 참조 픽셀들은 프레임에서 현재 블록에 대한 하나 이상의 이웃 블록들로부터의 픽셀들을 포함한다.
- [0088] 일부 실시형태들에서, 제 1 서브-파티션은 현재 블록의 에지를 따라 제 1 위치에 위치되고, 제 2 서브-파티션은 제 1 서브-파티션 바로 옆에 위치되지 않는다.
- [0089] 일부 실시형태들에서, 제 3 서브-파티션 및 제 4 서브-파티션은 제 1 서브-파티션과 제 2 서브-파티션 사이에 위치된다.
- [0090] 일부 실시형태들에서, 제 1 세트의 참조 픽셀들의 적어도 부분은 제 2 서브-파티션 및 제 3 서브-파티션 바로 옆에 위치되고, 제 1 세트의 참조 픽셀들의 적어도 부분은 현재 블록에 대한 이웃 블록에 있다.
- [0091] 일부 실시형태들에서, 방법은, 제 1 세트의 참조 픽셀들을 사용하는 제 1 트랜스코딩 프로세스를 사용하여 제 1 서브-파티션을 프로세싱하는 단계; 및 제 1 서브-파티션의 프로세싱과 병렬로 제 1 세트의 참조 픽셀들을 사용하는 제 2 트랜스코딩 프로세스를 사용하여 제 2 서브-파티션을 프로세싱하는 단계를 더 포함한다.
- [0092] 일부 실시형태들에서, 방법은, 제 1 트랜스코딩 프로세스로부터의 제 2 세트의 참조 픽셀들을 제 4 서브-파티션을 프로세싱하는데 있어서 사용하기 위해 제 2 트랜스코딩 프로세스에 제공하는 단계, 및 제 2 트랜스코딩 프로세스로부터의 제 3 세트의 참조 픽셀들을 제 3 서브-파티션을 프로세싱하는데 있어서 사용하기 위해 제 1 트랜스코딩 프로세스에 제공하는 단계를 더 포함한다.
- [0093] 일부 실시형태들에서, 제 2 세트의 참조 픽셀들은 제 1 서브-파티션의 디코딩된 픽셀들로부터 생성되고, 제 3 세트의 참조 픽셀들은 제 2 서브-파티션의 디코딩된 픽셀들로부터 생성된다.
- [0094] 일부 실시형태들에서, 제 2 세트의 참조 픽셀들은 제 1 서브-파티션의 제 1 잔차로부터 생성되고, 제 3 세트의 참조 픽셀들은 제 2 서브-파티션의 제 2 잔차로부터 생성된다.
- [0095] 일부 실시형태들에서, 제 1 세트의 참조 픽셀들을 사용하여 제 1 서브-파티션을, 그리고 제 1 세트의 참조 픽셀들을 사용하여 제 2 서브-파티션을 병렬로 프로세싱하는 단계는, 제 1 세트의 참조 픽셀들로부터 제 1 예측된 픽셀들을 생성하는 단계; 제 1 예측된 픽셀들을 사용하여 제 1 서브-파티션을 트랜스코딩하는 단계; 제 2 세트의 참조 픽셀들로부터 제 2 예측된 픽셀들을 생성하는 단계; 및 제 2 예측된 픽셀들을 사용하여 제 2 서브-파티션을 트랜스코딩하는 단계를 포함한다.
- [0096] 일부 실시형태들에서, 제 1 세트의 참조 픽셀들의 적어도 부분, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 복수의 서브-파티션들에서의 제 3 서브-파티션을 프로세싱하고 제 1 세트의 참조 픽셀들의 적어도 부분, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 복수의 서브-파티션들에서의 제 4 서브-파티션을 프로세싱하는 단계는, 제 1 세트의 참조 픽셀들의 적어도 부분, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 제 3 예측된 픽셀들을 생성하는 단계; 제 3 예측된 픽셀들을 사용하여 제 3 서브-파티션을 트랜스코딩하는 단계; 제 1 세트의 참조 픽셀들의 적어도 부분, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 제 4 예측된 픽셀들을 생성하는 단계; 및 제 4 예측된 픽셀들을 사용하여 제 4 서브-파티션을 트랜스코딩하는 단계를 포함한다.
- [0097] 일부 실시형태들에서, 제 2 세트의 참조 픽셀들은 제 1 서브-파티션의 제 1 디코딩된 픽셀들 및 제 1 예측된 픽셀들로부터 생성되고, 제 3 세트의 참조 픽셀들은 제 2 서브-파티션의 제 2 디코딩된 픽셀들 및 제 2 예측된 픽셀들로부터 생성된다.
- [0098] 일부 실시형태들에서, 제 1 세트의 참조 픽셀들을 사용하여 제 1 서브-파티션을, 그리고 제 1 세트의 참조 픽셀들을 사용하여 제 2 서브-파티션을 병렬로 프로세싱하는 단계는, 제 1 서브-파티션을 디코딩하는 단계; 제 1 세트의 참조 픽셀들로부터 제 1 예측된 픽셀들을 생성하는 단계; 디코딩된 제 1 서브-파티션 및 제 1 예측된 픽셀들로부터 제 2 세트의 참조 픽셀들을 생성하는 단계; 제 2 서브-파티션을 디코딩하는 단계; 제 2 세트의 참조 픽셀들로부터 제 2 예측된 픽셀들을 생성하는 단계; 및 디코딩된 제 2 서브-파티션 및 제 2 예측된 픽셀들로부터

터 제 3 세트의 참조 픽셀들을 생성하는 단계를 포함한다.

- [0099] 일부 실시형태들에서, 명령들을 포함하는 비밀시적 컴퓨터 관독가능 저장 매체로서, 그 명령들은, 실행될 경우, 코딩될 현재 블록에 대한 제 1 세트의 참조 픽셀들을 결정하는 것으로서, 현재 블록은 복수의 서브-파티션들로 스플리팅되는, 상기 제 1 세트의 참조 픽셀들을 결정하고; 제 1 세트의 참조 픽셀들을 사용하여 복수의 서브-파티션들에서의 제 1 서브-파티션 및 제 2 서브-파티션을 병렬로 프로세싱하고; 제 1 서브-파티션으로부터 제 2 세트의 참조 픽셀들 및 제 2 서브-파티션으로부터 제 3 세트의 참조 픽셀들을 생성하고; 제 1 세트의 참조 픽셀들의 적어도 부분, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 복수의 서브-파티션들에서의 제 3 서브-파티션을 프로세싱하고 제 1 세트의 참조 픽셀들의 적어도 부분, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나 이상에 기초하여 복수의 서브-파티션들에서의 제 4 서브-파티션을 프로세싱하기 위해 동작가능하도록 컴퓨터 시스템을 제어한다.
- [0100] 일부 실시형태들에서, 제 3 서브-파티션 및 제 4 서브-파티션을 프로세싱하는 것은, 제 1 세트의 참조 픽셀들, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나로부터의 제 1 참조 픽셀을 사용하는 것; 및 제 1 참조 픽셀에 의해 사용되지 않는 제 1 세트의 참조 픽셀들, 제 2 세트의 참조 픽셀들, 및 제 3 세트의 참조 픽셀들 중 하나로부터의 제 2 참조 픽셀을 사용하는 것을 포함한다.
- [0101] 일부 실시형태들에서, 장치는, 하나 이상의 컴퓨터 프로세서들; 및 명령들을 포함하는 비밀시적 컴퓨터 관독가능 저장 매체를 포함하고, 그 명령들은, 실행될 경우, 코딩될 현재 블록에 대한 제 1 세트의 참조 픽셀들을 결정하는 것으로서, 현재 블록은 복수의 서브-파티션들로 스플리팅되는, 상기 제 1 세트의 참조 픽셀들을 결정하고; 제 1 세트의 참조 픽셀들을 사용하여 복수의 서브-파티션들에서의 제 1 서브-파티션을, 그리고 제 1 세트의 참조 픽셀들을 사용하여 복수의 서브-파티션들에서의 제 2 서브-파티션을 병렬로 프로세싱하고; 제 1 서브-파티션의 프로세싱으로부터 결정된 제 2 세트의 참조 픽셀들에 기초하여 복수의 서브-파티션들에서의 제 3 서브-파티션을 프로세싱하고 제 2 서브-파티션의 프로세싱으로부터 결정된 제 3 세트의 참조 픽셀들에 기초하여 복수의 서브-파티션들에서의 제 4 서브-파티션을 프로세싱하기 위해 동작가능하도록 하나 이상의 컴퓨터 프로세서들을 제어한다.
- [0102] 일부 실시형태들은 명령 실행 시스템, 장치, 시스템, 또는 머신에 의해 또는 그와 관련하여 사용하기 위해 비밀시적 컴퓨터 관독가능 저장 매체에서 구현될 수도 있다. 컴퓨터 관독가능 저장 매체는 일부 실시형태들에 의해 설명된 방법을 수행하기 위해 컴퓨터 시스템을 제어하기 위한 명령들을 포함한다. 컴퓨터 시스템은 하나 이상의 컴퓨팅 디바이스들을 포함할 수도 있다. 명령들은, 하나 이상의 컴퓨터 프로세서들에 의해 실행될 경우, 일부 실시형태들에서 설명되는 것을 수행하도록 구성되거나 또는 동작가능할 수도 있다.
- [0103] 본 명세서에서의 설명에서 그리고 다음에 오는 청구항들 전체에 걸쳐 사용된 바와 같이, "a", "an", 및 "the" 는 문맥에서 분명히 달리 나타내지 않는 한 복수의 참조물들을 포함한다. 또한, 본 명세서에서의 설명에서 그리고 다음에 오는 청구항들 전체에 걸쳐 사용된 바와 같이, "에서 (in)" 의 의미는 문맥에서 분명히 달리 나타내지 않는 한 "에서 (in)" 및 "상에서 (on)" 를 포함한다.
- [0104] 상기 설명은 일부 실시형태들의 양태들이 구현될 수도 있는 방법의 예들과 함께 다양한 실시형태들을 예시한다. 상기 예들 및 실시형태들은 유일한 실시형태들인 것으로 간주되어서는 안되며 다음의 청구항들에 의해 정의된 바와 같이 일부 실시형태들의 유연성 및 이점들을 예시하기 위해 제시된다. 상기 개시 및 다음의 청구항들에 기초하여, 청구항들에 의해 정의된 바와 같은 본 개시의 범위로부터 벗어남 없이 다른 배열들, 실시형태들, 구현들, 및 등가물들이 채용될 수도 있다.

도면

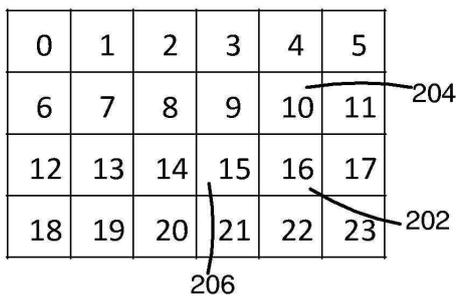
도면1



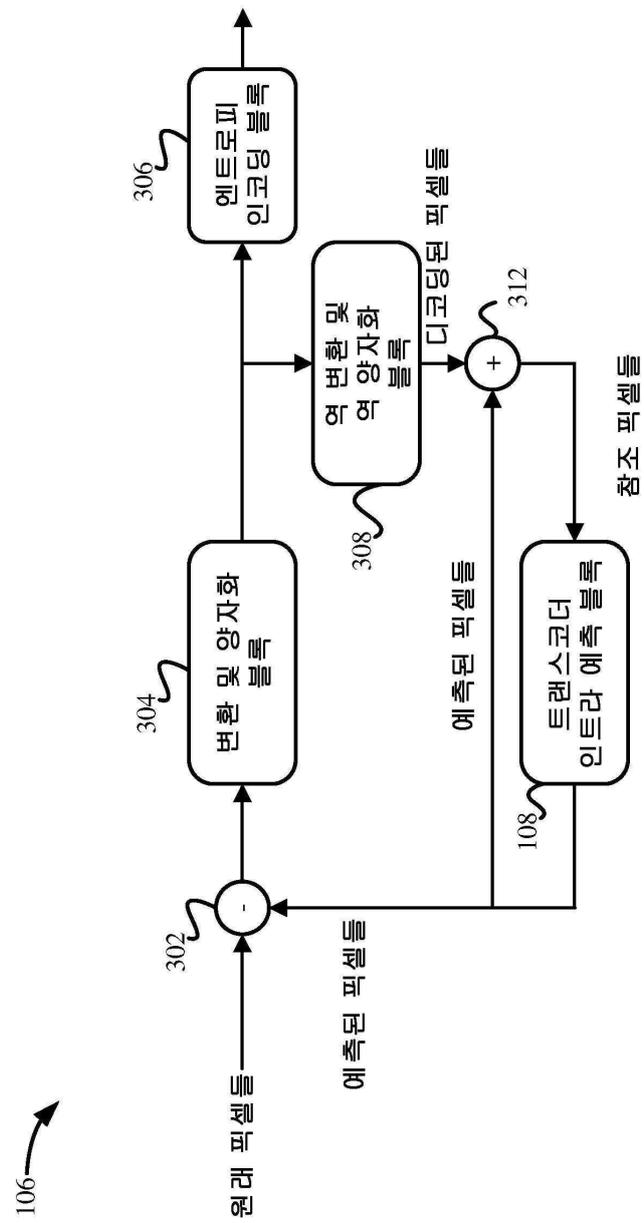
100 →

도면2

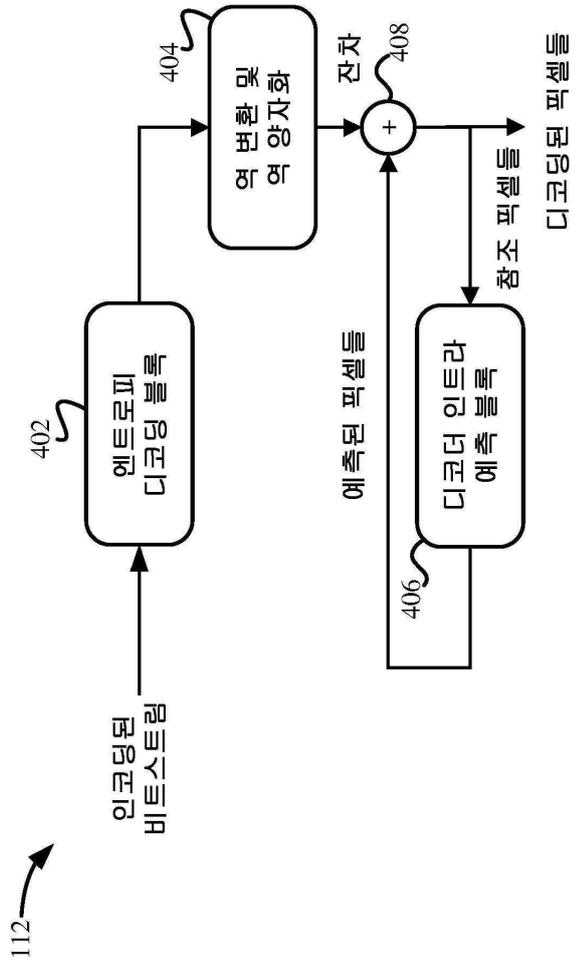
200 →



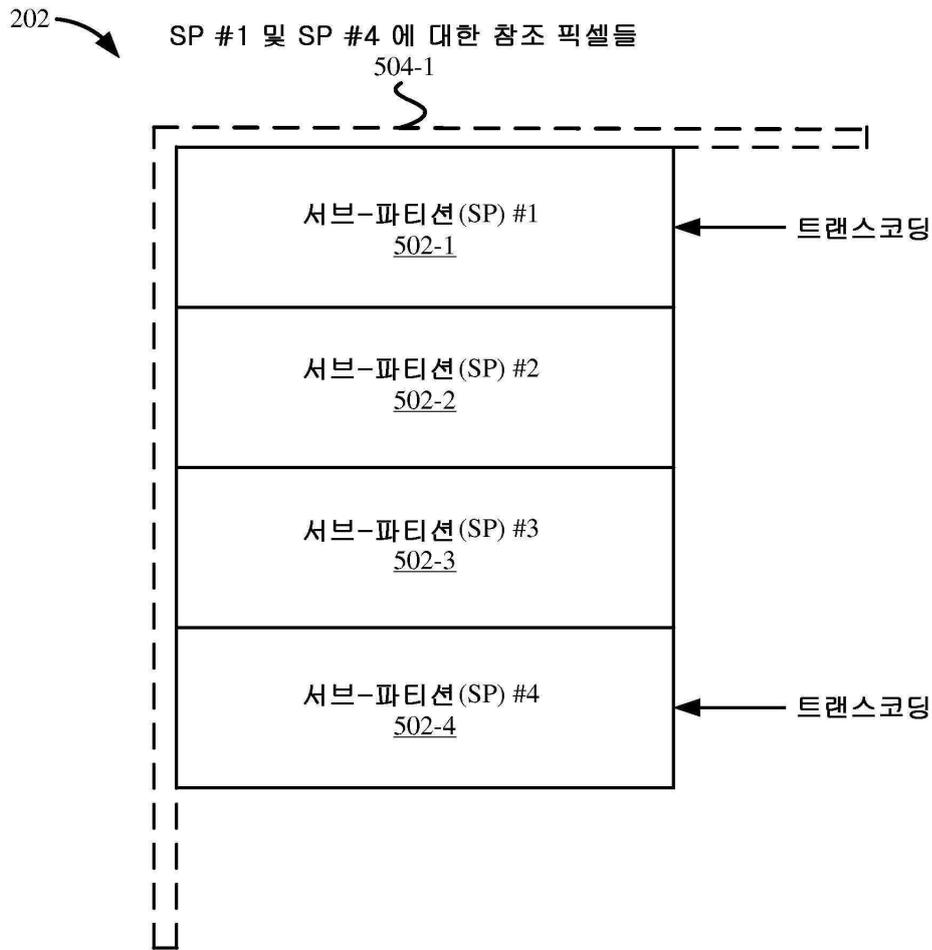
도면3



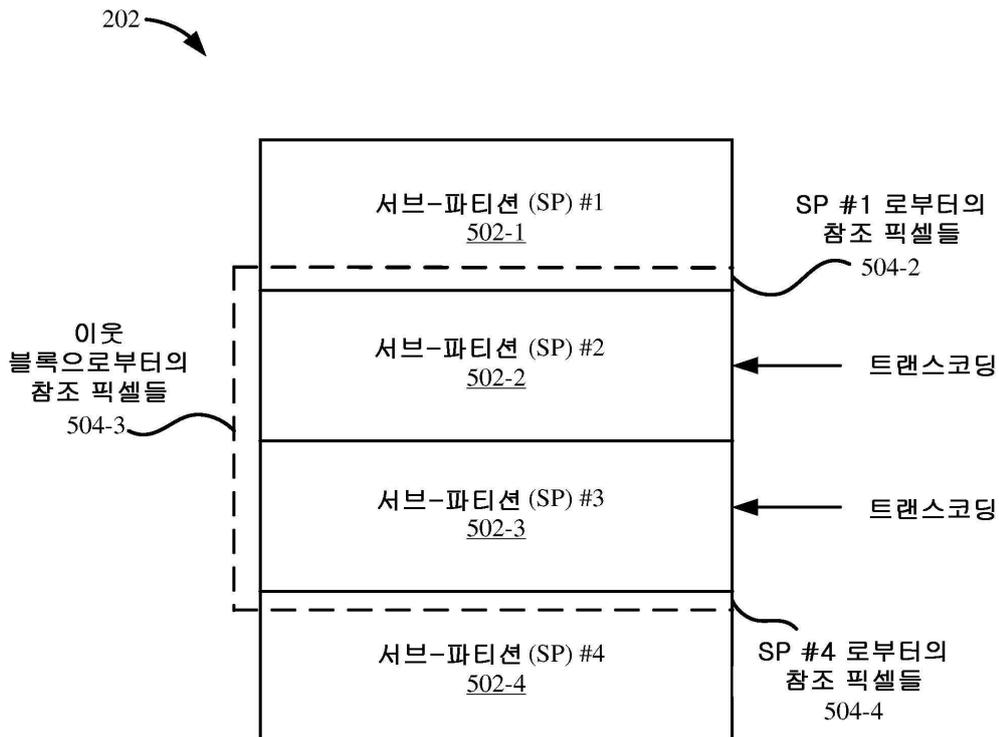
도면4



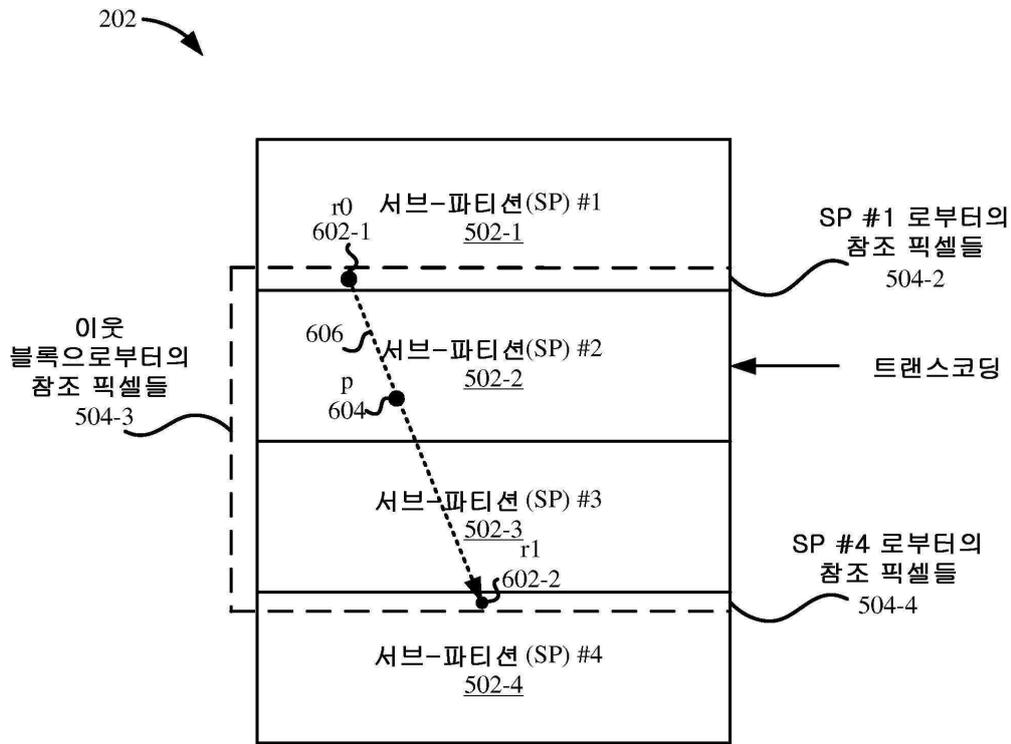
도면5a



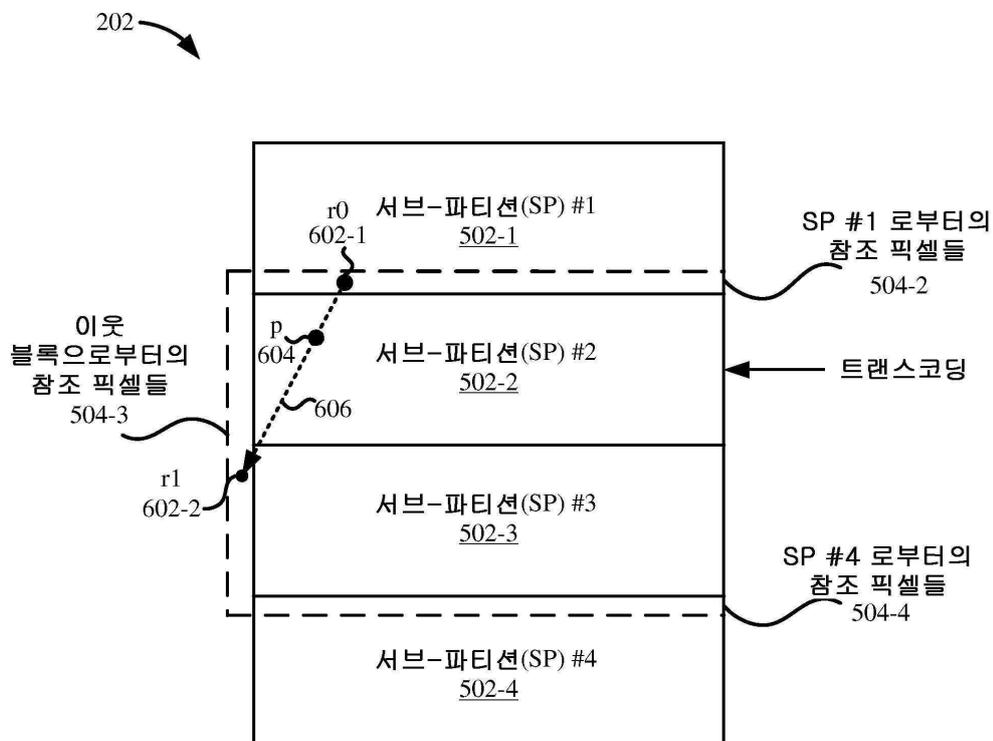
도면5b



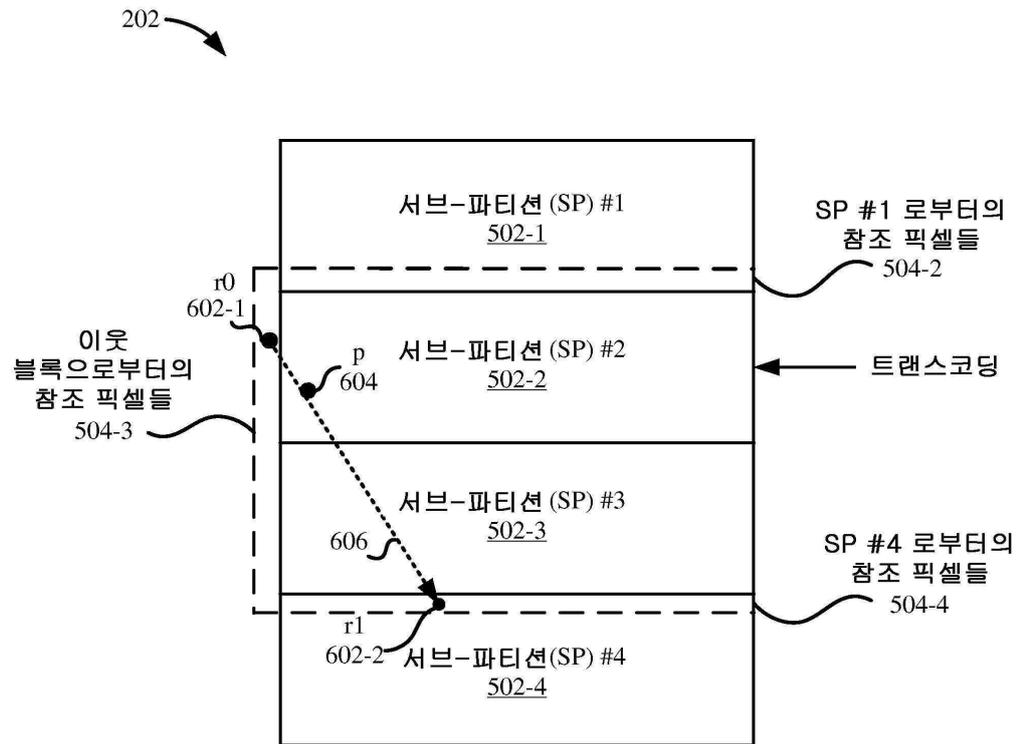
도면6a



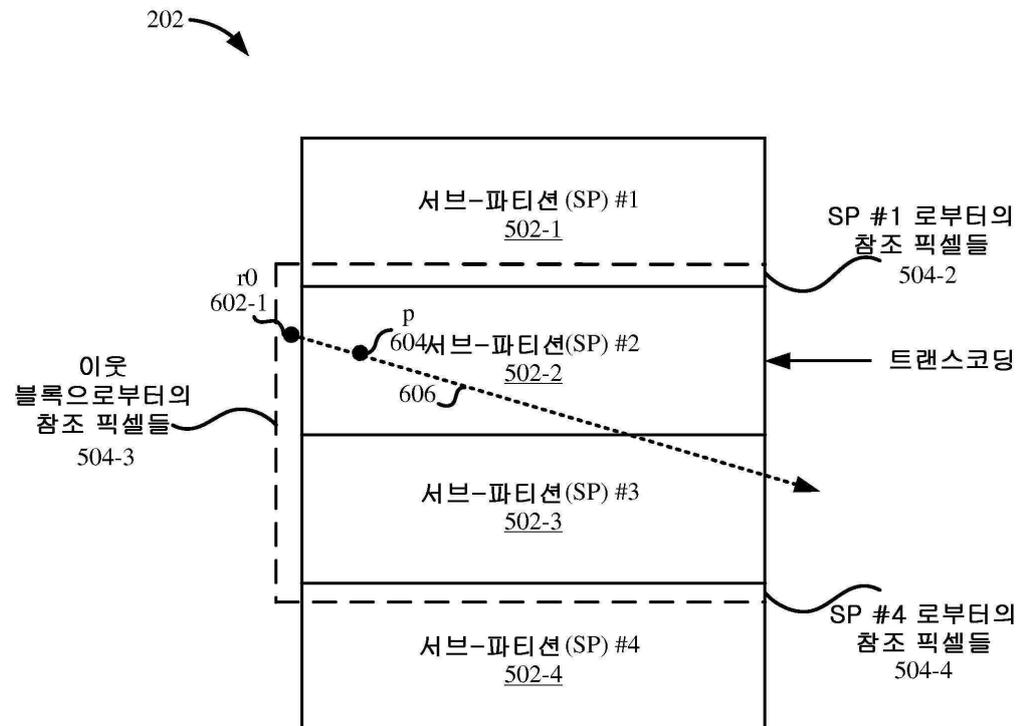
도면6b



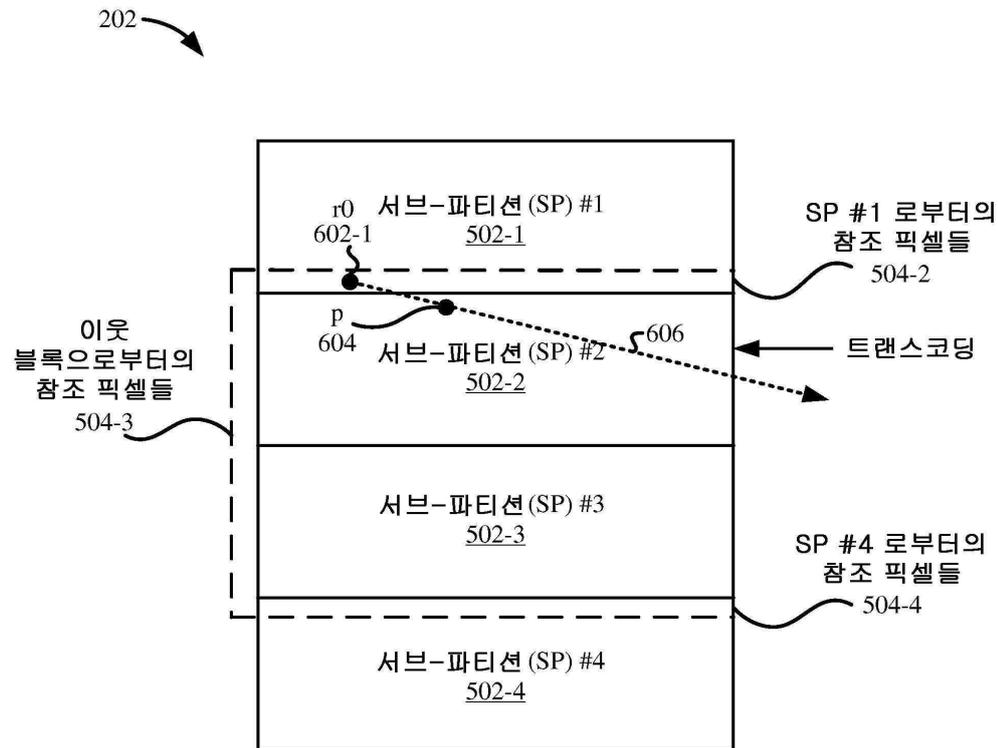
도면6c



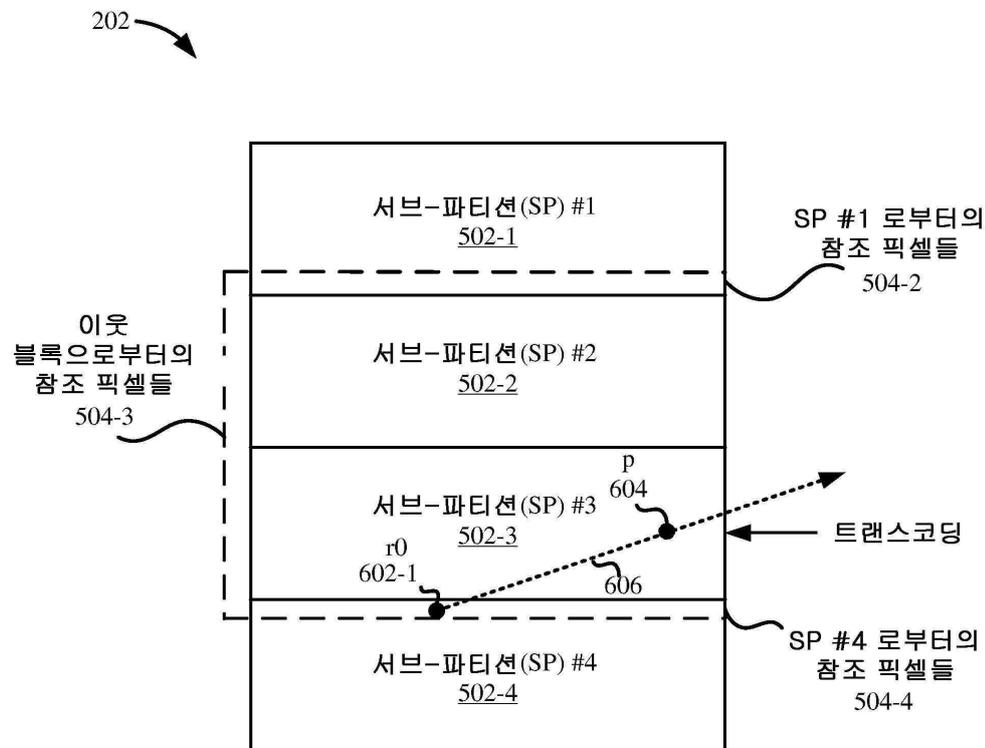
도면7a



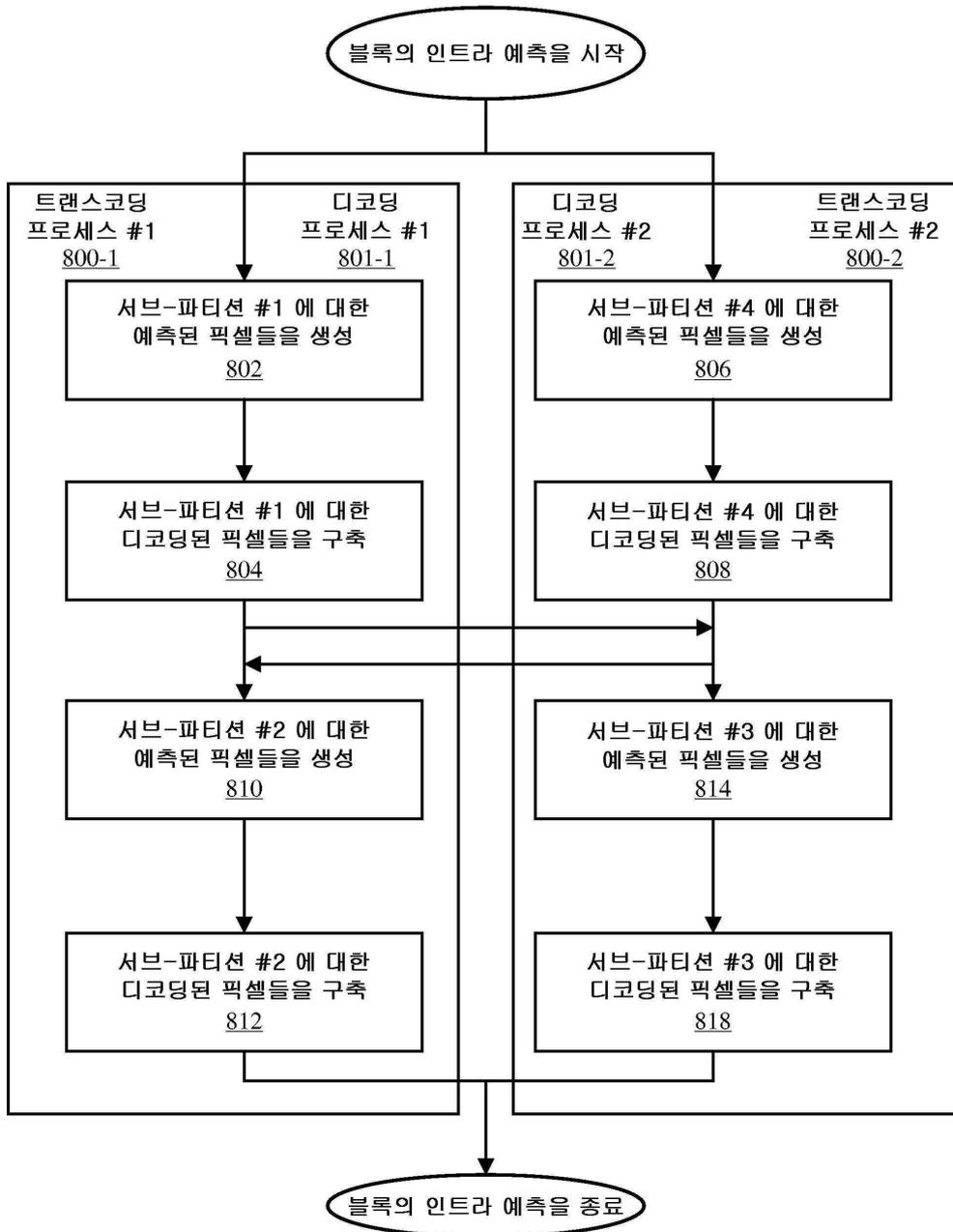
도면7b



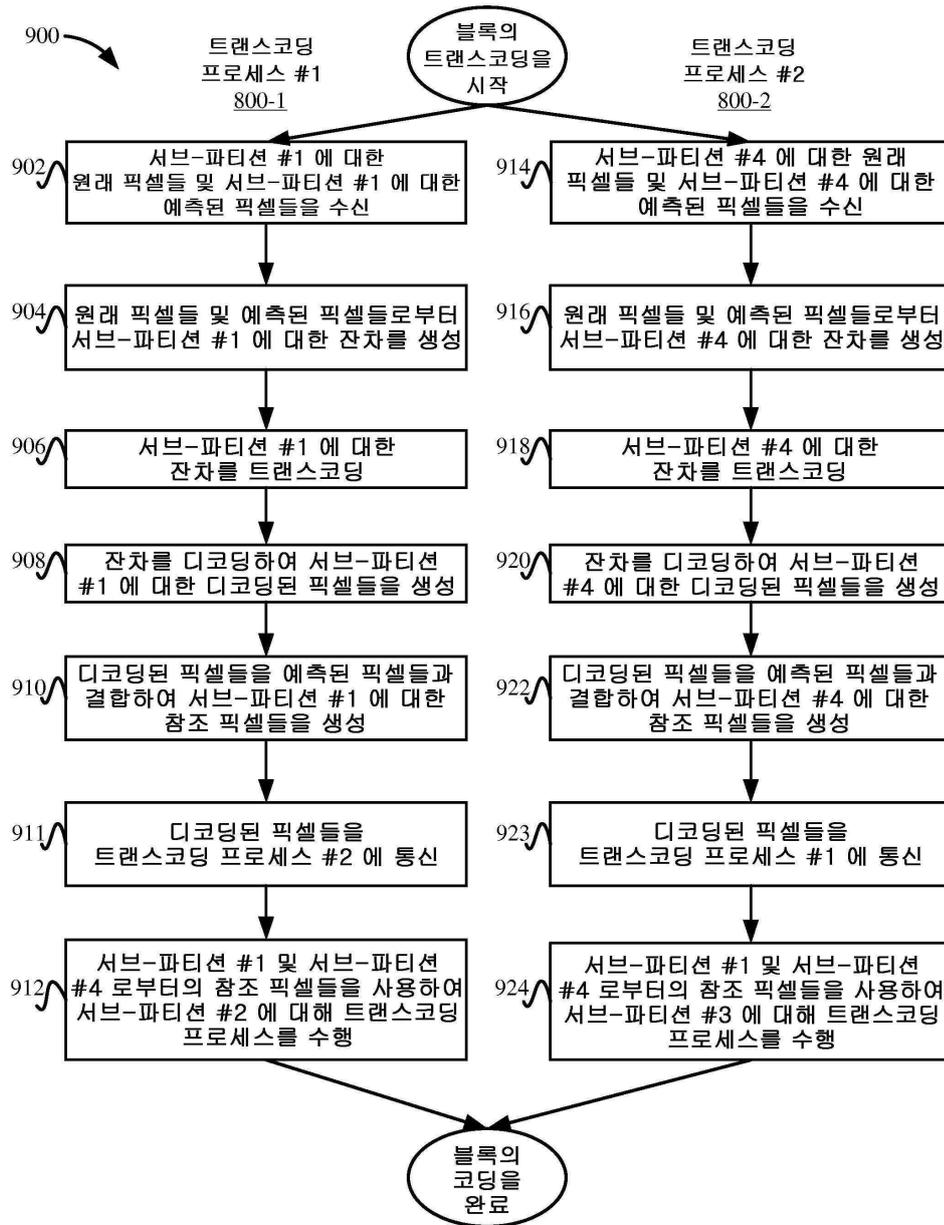
도면7c



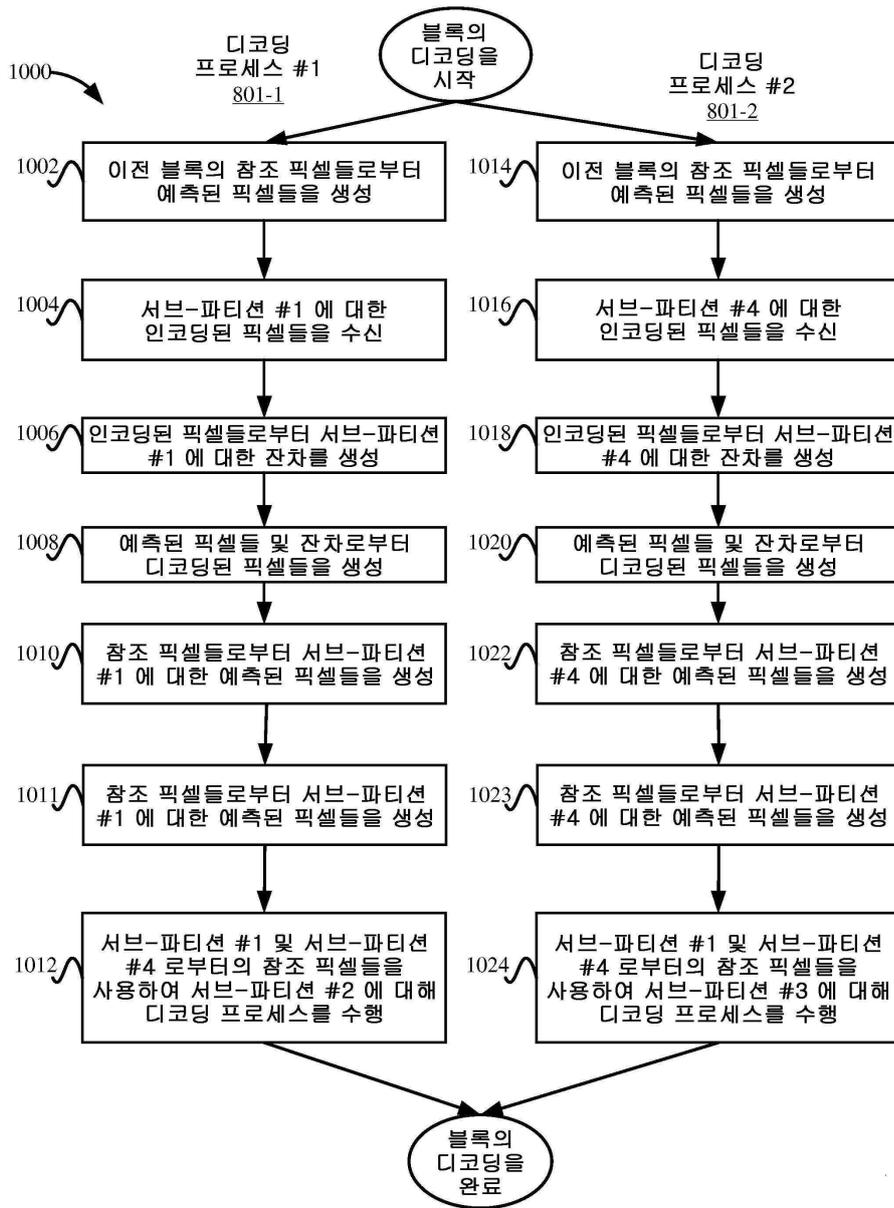
도면8



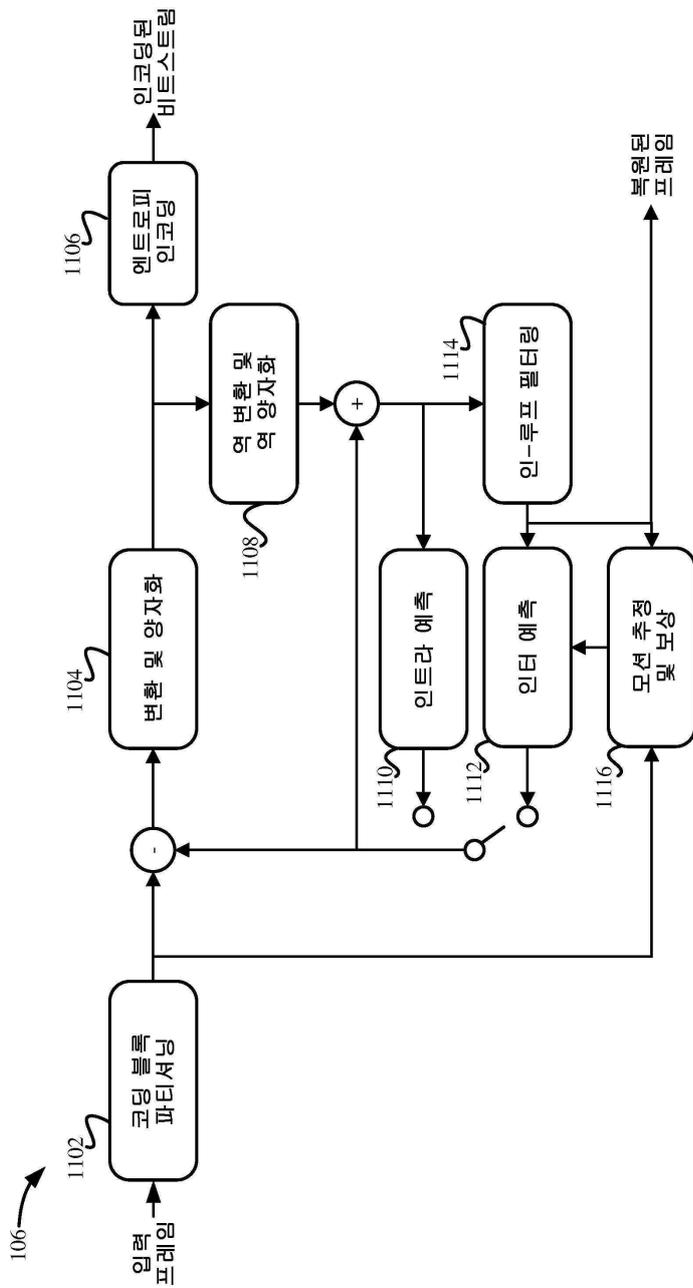
도면9



도면10



도면11



도면12

