



공개특허 10-2023-0092029



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2023-0092029  
(43) 공개일자 2023년06월23일

- (51) 국제특허분류(Int. Cl.)  
*H04N 19/70* (2014.01) *H04N 19/105* (2014.01)  
*H04N 19/174* (2014.01) *H04N 19/573* (2014.01)  
*H04N 19/58* (2014.01)
- (52) CPC특허분류  
*H04N 19/70* (2015.01)  
*H04N 19/105* (2015.01)
- (21) 출원번호 10-2023-7020078(분할)
- (22) 출원일자(국제) 2013년09월23일  
심사청구일자 2023년06월14일
- (62) 원출원 특허 10-2022-7038323  
원출원일자(국제) 2013년09월23일  
심사청구일자 2022년11월01일
- (85) 번역문제출일자 2023년06월14일
- (86) 국제출원번호 PCT/US2013/061241
- (87) 국제공개번호 WO 2014/052249  
국제공개일자 2014년04월03일
- (30) 우선권주장  
61/708,042 2012년09월30일 미국(US)  
13/781,710 2013년02월28일 미국(US)

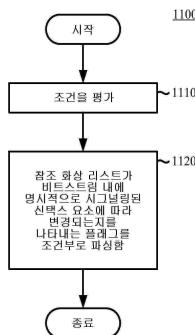
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 참조 화상 리스트 변경 정보를 조건부로 시그널링하는 기법

### (57) 요 약

참조 화상 리스트("RPL") 변경 정보를 시그널링하는 혁신적인 기법이 제공된다. 예를 들어, 비디오 인코더는 전체 참조 화상의 수를 나타내는 변수에 적어도 부분적으로 의존하는 조건을 평가한다. 평가의 결과에 의존하여, 인코더는 RPL이 비트스트림 내에 명시적으로 시그널링된 신팩스 요소에 따라 변경되는지를 나타내는 플래그를 비트스트림 내에 시그널링한다. 비디오 디코더는 전술한 조건을 평가하고 평가 결과에 의존하여, RPL이 비트스트림 내에 명시적으로 시그널링된 신팩스 요소에 따라 변경되는지를 나타내는 플래그를 비트스트림으로부터 파싱 한다. 조건은 플래그를 포함하는 RPL 변경 구조에 대한 프로세싱의 일부로서, 또는 슬라이스 헤더에 대한 프로세싱의 일부로서 평가될 수 있다. 인코더 및 디코더는 또한 RPL 변경 정보의 리스트 엔트리들에 대한 신팩스 요소에 영향을 미치는 다른 조건을 평가할 수 있다.

### 대 표 도



(52) CPC특허분류

*H04N 19/174* (2015.01)

*H04N 19/573* (2015.01)

*H04N 19/58* (2015.01)

---

## 명세서

### 청구범위

#### 청구항 1

비디오 디코더를 구현하는 컴퓨터 시스템에서의 방법으로서,

비트스트림으로 인코딩된 데이터를 수신하는 단계와,

제1 조건이 만족되는지 결정하는 단계 - 상기 제1 조건은 제1 임계값을 초과하는 참조 화상 리스트(RPL) 변수의

값에 적어도 부분적으로 의존함 - 와,

상기 제1 조건이 만족되는 것에 응답하여, 추가 RPL 정보가 상기 비트스트림에 표시되어 있는지를 나타내는 플래그를 상기 비트스트림으로부터 파싱하는 단계와,

상기 추가 RPL 정보가 상기 비트스트림에 표시되어 있다고 결정되는 것에 응답하여, 제2 조건이 만족되는지 여부를 확인하는 단계 - 상기 제2 조건은 제2 임계값을 초과하는 상기 RPL 변수의 값에 적어도 부분적으로 의존하고, 상기 제2 임계값은 상기 제1 임계값보다 큼 - 와,

RPL 구조에서 엔트리를 나타내는 인덱스의 값을 결정하는 단계 - 상기 결정하는 단계는, 상기 제2 조건이 만족되면 상기 비트스트림으로부터 상기 인덱스의 값을 파싱하고, 그렇지 않고 상기 제2 조건이 만족되지 않으면 상기 인덱스의 값을 추론하는 단계를 포함함 - 를 포함하는,

방법.

#### 청구항 2

제1항에 있어서,

상기 제2 조건이 만족되면, 상기 인덱스의 값은 상기 RPL 변수의 값에 의존하는 비트 수를 갖는,

방법.

#### 청구항 3

제2항에 있어서,

상기 인덱스의 값은 0 내지 상기 RPL 변수의 값에서 1을 뺀 값의 범위에 있는,

방법.

#### 청구항 4

제2항에 있어서,

상기 비트 수는 상기 RPL 변수의 값의 로그 베이스 2이며, 천장 함수(ceiling function)에 따라 가장 가까운 정수로 반올림되는,

방법.

#### 청구항 5

제1항에 있어서,

상기 제1 임계값은 상기 제2 임계값보다 1 작은,

방법.

#### 청구항 6

제1항에 있어서,

상기 제1 조건이 만족되는지 결정하는 단계, 상기 파싱하는 단계, 상기 추가 RPL 정보가 상기 비트스트림에 표시되어 있는지 결정하는 단계, 상기 확인하는 단계 및 상기 인덱스의 값을 결정하는 단계는 현재 슬라이스의 슬라이스 헤더의 처리의 일부로서 수행되는,

방법.

### 청구항 7

제1항에 있어서,

상기 제2 조건이 만족되지 않는 경우, 상기 인덱스의 값은 0으로 추론되는,

방법.

### 청구항 8

인코딩된 데이터를 비트스트림으로 저장한 하나 이상의 컴퓨터 판독가능 매체로서, 상기 인코딩된 데이터는 컴퓨터-구현된 비디오 인코더를 사용하여 동작에 의해 비디오를 인코딩하여 생성되고, 상기 동작은,

제1 조건이 만족되는지 결정하는 단계 - 상기 제1 조건은 제1 임계값을 초과하는 참조 화상 리스트(RPL) 변수의 값에 적어도 부분적으로 의존함 - 와,

상기 제1 조건이 만족되는 것에 응답하여, 추가 RPL 정보가 상기 비트스트림에 표시되어 있는지를 나타내는 플래그를 상기 비트스트림에 시그널링하는 단계와,

상기 플래그와 일치하게, 상기 추가 RPL 정보가 상기 비트스트림에 표시되어 있는지를 결정하는 단계와,

상기 추가 RPL 정보가 상기 비트스트림에 표시되어 있다고 결정되는 것에 응답하여, 제2 조건이 만족되는지 여부를 확인하는 단계 - 상기 제2 조건은 제2 임계값을 초과하는 상기 RPL 변수의 값에 적어도 부분적으로 의존하고, 상기 제2 임계값은 상기 제1 임계값보다 큼 - 와,

RPL 구조에서 엔트리를 나타내는 인덱스의 값을 조건부로 시그널링 하는 단계 - 상기 조건부로 시그널링하는 단계는, 상기 제2 조건이 만족되면 상기 비트스트림에 상기 인덱스의 값을 시그널링하고, 그렇지 않고 상기 제2 조건이 만족되지 않으면 상기 비트스트림에 상기 인덱스의 값을 시그널링하는 것을 건너뛰는 것을 포함함 - 를 포함하는,

하나 이상의 컴퓨터 판독가능 매체.

### 청구항 9

제8항에 있어서,

상기 제2 조건이 만족되면, 상기 인덱스의 값은 상기 RPL 변수의 값에 의존하는 비트 수를 갖는,

하나 이상의 컴퓨터 판독가능 매체.

### 청구항 10

제9항에 있어서,

상기 인덱스의 값은 0 내지 상기 RPL 변수의 값에서 1을 뺀 값의 범위에 있고, 상기 비트 수는 상기 RPL 변수의 값의 로그 베이스 2이며, 천장 함수(ceiling function)에 따라 가장 가까운 정수로 반올림되는,

하나 이상의 컴퓨터 판독가능 매체.

### 청구항 11

제8항에 있어서,

상기 제1 임계값은 상기 제2 임계값보다 1 작은,

하나 이상의 컴퓨터 판독가능 매체.

**청구항 12**

제8항에 있어서,

상기 동작은 현재 슬라이스의 슬라이스 헤더의 처리의 일부로서 수행되는,  
하나 이상의 컴퓨터 판독가능 매체.

**청구항 13**

제8항에 있어서,

상기 제2 조건이 만족되지 않는 경우, 상기 인덱스의 값은 0으로 추론되는,  
하나 이상의 컴퓨터 판독가능 매체.

**청구항 14**

메모리와 하나 이상의 프로세싱 유닛을 포함하는 컴퓨터 시스템으로서, 상기 컴퓨터 시스템은 비디오 인코더를 구현하고, 상기 비디오 인코더는

비디오를 수신하도록 구성되는 버퍼와,

동작에 의해 비트스트림으로 인코딩된 데이터를 생성하도록 상기 비디오를 인코딩하도록 구성된 비디오 인코더를 포함하고, 상기 동작은,

제1 조건이 만족되는지 결정하는 단계 - 상기 제1 조건은 제1 임계값을 초과하는 참조 화상 리스트(RPL) 변수의 값에 적어도 부분적으로 의존함 - 와,

상기 제1 조건이 만족되는 것에 응답하여, 추가 RPL 정보가 상기 비트스트림에 표시되어 있는지를 나타내는 플래그를 상기 비트스트림에 시그널링하는 단계와,

상기 플래그와 일치하게, 상기 추가 RPL 정보가 상기 비트스트림에 표시되어 있는지를 결정하는 단계와,

상기 추가 RPL 정보가 상기 비트스트림에 표시되어 있다고 결정되는 것에 응답하여, 제2 조건이 만족되는지 여부를 확인하는 단계 - 상기 제2 조건은 제2 임계값을 초과하는 상기 RPL 변수의 값에 적어도 부분적으로 의존하고, 상기 제2 임계값은 상기 제1 임계값보다 큼 - 와,

RPL 구조에서 엔트리를 나타내는 인덱스의 값을 조건부로 시그널링 하는 단계 - 상기 조건부로 시그널링하는 단계는, 상기 제2 조건이 만족되면 상기 비트스트림에 상기 인덱스의 값을 시그널링하고, 그렇지 않고 상기 제2 조건이 만족되지 않으면 상기 비트스트림에 상기 인덱스의 값을 시그널링하는 것을 전너뛰는 것을 포함함 - 를 포함하는,

컴퓨터 시스템.

**청구항 15**

제14항에 있어서,

상기 제2 조건이 만족되면, 상기 인덱스의 값은 상기 RPL 변수의 값에 의존하는 비트 수를 갖는,  
컴퓨터 시스템.

**청구항 16**

제15항에 있어서,

상기 인덱스의 값은 0 내지 상기 RPL 변수의 값에서 1을 뺀 값의 범위에 있는,  
컴퓨터 시스템.

**청구항 17**

제15항에 있어서,

상기 비트 수는 상기 RPL 변수의 값의 로그 베이스 2이며, 천장 함수(ceiling function)에 따라 가장 가까운 정수로 반올림되는,  
컴퓨터 시스템.

### 청구항 18

제14항에 있어서,  
상기 제1 임계값은 상기 제2 임계값보다 1 작은,  
컴퓨터 시스템.

### 청구항 19

제14항에 있어서,  
상기 동작은 현재 슬라이스의 슬라이스 헤더의 처리의 일부로서 수행되는,  
컴퓨터 시스템.

### 청구항 20

제14항에 있어서,  
상기 제2 조건이 만족되지 않는 경우, 상기 인덱스의 값은 0으로 추론되는,  
컴퓨터 시스템.

## 발명의 설명

### 기술 분야

## 배경 기술

[0001] 엔지니어들은 디지털 비디오의 비트 레이트를 감소시키기 위해 (소스 코딩 또는 소스 인코딩이라고도 불리는) 압축(compression)을 사용한다. 압축은, 비디오 정보를 보다 낮은 비트 레이트 형태로 변환함으로써 비디오 정보의 저장 및 전송 비용을 감소시킨다. (디코딩이라고도 하는) 압축해제(decompression)는 정보를 압축된 형태로부터 원래 버전으로 재구성한다. "코덱"은 인코더/디코더 시스템이다.

[0002] 지난 20년간에 걸쳐, H.261, H.262 (MPEG-2 또는 ISO/IEC 13818-2), H.263 및 H.264 (AVC 또는 ISO/IEC 14496-10) 표준 및 MPEG-1 (ISO/IEC 11172-2), MPEG-4 Visual (ISO/IEC 14496-2) 및 SMPTE 421M 표준을 비롯하여, 다양한 비디오 코덱 표준들이 채택되어 왔다. 보다 최근에는, HEVC 표준이 개발되고 있다. 통상적으로, 비디오 코덱 표준은 인코딩된 비디오 비트스트림의 신택스(syntax)에 관한 옵션을 정의하여, 특정 기능이 인코딩 및 디코딩에 사용될 때 비디오 비트스트림에서의 파라미터를 상세히 표시한다. 많은 경우에, 비디오 코덱 표준은 디코더가 올바른 디코딩 결과를 얻기 위해 수행해야 하는 디코딩 동작에 관한 세부사항도 제공한다. 코덱 표준 외에도, 다양한 전유(proprietary) 코덱 포맷이 인코딩된 비디오 비트스트림의 신택스(syntax)에 대한 다른 옵션 및 대응하는 디코딩 동작을 정의한다.

[0003] 비트스트림 내의 몇몇 유형의 파라미터는 비디오 인코딩 및 디코딩 동안 사용되는 참조 화상(reference picture)에 대한 정보를 나타낸다. 참조 화상이란 일반적으로 다른 화상들의 디코딩 프로세스에서 화상간 예측(inter-picture prediction)을 위해 사용될 수 있는 샘플들을 포함하는 화상이다. 전형적으로, 다른 화상들은 디코딩 순서로 참조 화상을 따라가고 모션 보상 예측을 위해 그 참조 화상을 사용한다. 몇몇 비디오 코덱 표준 및 포맷에 있어서, 모션 보상 예측에 사용될 다수의 참조 화상이 주어진 시간에 이용가능하다. 이러한 비디오 코덱 표준/포맷은 다수의 참조 화상을 어떻게 관리해야 하는지를 지정한다.

[0004] 일반적으로, 참조 화상 리스트(reference picture list)(“RPL”)은 모션 보상 예측을 위해 사용되는 참조 화상들의 리스트이다. 몇몇 비디오 코덱 표준 및 포맷에서, 참조 화상 세트(“RPS”)는 주어진 시간에서 모션 보상 예측에 이용가능한 참조 화상들의 세트이고, RPL은 RPS 내의 참조 화상을 중 일부이다. RPL 내의 참조 화상들은

참조 인덱스를 통해 어드레싱된다. 참조 인덱스는 RPL 내의 참조 화상을 식별한다. 인코딩 및 디코딩 동안, RPS는 새롭게 디코딩된 화상과 참조 화상으로서 더 이상 사용되지 않는 오래된 화상을 고려하여 업데이트될 수 있다. 또한, RPL 내의 참조 화상들은 보다 흔히 사용되는 참조 화상들이 시그널링에 보다 효율적인 참조 인덱스로 식별되도록 재배열(reorder)될 수 있다. 최근의 몇몇 코덱 표준에서, RPL은 RPS에 대한 이용가능한 정보, 규칙에 따른 변경 및/또는 비트스트림 내에서 시그널링된 변경에 기초하여 인코딩 및 디코딩 동안 구성된다. RPL에 대한 변경의 시그널링은 많은 양의 비트를 소비할 수 있다.

### 발명의 내용

[0005]

요약하면, 발명의 상세한 설명에서는, 참조 화상 리스트("RPL") 변경 정보의 시그널링의 혁신(innovations)을 제공한다. 보다 일반적으로는, 이 혁신은 RPL 변경 정보가 사용될 수 없거나 이러한 정보의 값이 추론될 수 있는 경우 RPL 변경 정보의 시그널링을 피하기 위한 여러 방식에 관한 것이다.

[0006]

본 명세서에 기술되어 있는 혁신들 중 하나의 측면에 따르면, 비디오 인코더는 소정의 조건을 평가한다. 이 평가의 결과에 따라, 인코더는 비트스트림에 명시적으로 시그널링된 선택스 요소에 따라 RPL이 변경되는지를 나타내는 플래그를 비트스트림 내에 조건부로 시그널링한다. 대응하는 비디오 디코더는 소정의 조건을 평가한다. 이 평가의 결과에 따라, 디코더는 비트스트림에 명시적으로 시그널링된 선택스 요소에 따라 RPL이 변경되는지를 나타내는 플래그를 비트스트림으로부터 조건부로 파싱(parse)한다. 몇몇 예시적인 구현에서, RPL은 예측("P") 슬라이스 또는 쌍방향 예측(bi-predictive)("B") 슬라이스를 위한 것일 수 있다. 이와 달리, 보다 상위 레벨의 선택스 구조가 조건의 평가에 기초하여 조건부로 시그널링/파싱된다.

[0007]

몇몇 예시적인 구현에서, RPL이 변경되지 않는 경우, RPS로부터 RPL 구성에 대한 규칙에 기초하여 기본 RPL이 구성된다. RPL이 변경된 경우, RPS로부터 참조 화상의 선택을 나타내는 시그널링된 RPL 변경 정보에 기초하여 대체 RPL이 구성된다. 이와 달리, 기본 RPL을 재배열하거나, 참조 화상을 기본 RPL에 추가하거나, 기본 RPL로부터 참조 화상을 제거하는 변경은 기본 RPL을 조정하기 위해 보다 정밀한 방식으로 시그널링된다.

[0008]

예를 들어, 평가되는 조건은 전체 참조 화상의 수를 나타내는 변수에 적어도 부분적으로 의존한다. 몇몇 예시적인 구현에서, 조건은 변수의 값이 1보다 큰지 여부이다.

[0009]

조건은 플래그를 포함하는 RPL 변경 구조에 대한 프로세싱의 일부로서 평가될 수 있다. 또는, 조건은 슬라이스 헤더에 대한 프로세싱의 일부로서 평가될 수 있는데, 이 경우 (플래그를 포함하는) RPL 변경 구조는 평가의 결과에 따라 조건부로 시그널링 또는 파싱된다.

[0010]

본 명세서에서 기술된 혁신들 중 또 다른 측면에 따르면, 비디오 인코더는 또 다른 조건을 평가한다. 평가의 결과에 따라, 인코더는 RPL을 어떻게 변경(예를 들어, RPL을 대체, RPL을 조정)할 것인지를 나타내는 리스트 엔트리들에 대한 하나 이상의 선택스 요소를 비트스트림 내에 조건부로 시그널링한다. 대응하는 비디오 디코더는 조건을 평가한다. 평가의 결과에 따라, 디코더는 RPL을 어떻게 변경(예를 들어, RPL을 대체, RPL을 조정)할 것인지를 나타내는 리스트 엔트리들에 대한 하나 이상의 선택스 요소를 비트스트림으로부터 조건부로 파싱한다. 몇몇 예시적인 구현에서, RPL은 P 슬라이스 또는 B 슬라이스를 위한 것일 수 있다(조건 평가 및 조건부적인 시그널링/파싱은 B 슬라이스에 대한 다수의 RPL 각각마다 반복된다). 예를 들어, 다른 조건은 전체 참조 화상의 수, RPL의 활성 참조 화상의 수 및/또는 가중치 부여된 예측(weighted prediction)이 디스에이블링되는지 여부를 나타내는 변수에 적어도 부분적으로 의존한다. 현재 슬라이스가 P 슬라이스인지 또는 B 슬라이스인지에 따라 및/또는 어떤 RPL이 시그널링/파싱되는지에 따라 가중치 부여된 예측이 디스에이블링되는지 여부를 확인하는데 다른 로직이 사용될 수 있다. 몇몇 예시적인 구현에서, (a) 전체 참조 화상의 수가 2이고 (b) RPL에 대한 활성 참조 화상의 수가 1인 경우, 리스트 엔트리들에 대한 하나 이상의 선택스 요소는 비트스트림으로부터 빠져 있고, 리스트 엔트리들 중 하나에 대한 값이 추론된다. 또한, 몇몇 예시적인 구현에서, (c) 전체 참조 화상의 수가 2이고 (d) RPL에 대한 활성 참조 화상의 수가 2이며, (e) 가중치 부여된 예측이 디스에이블링되는 경우, 리스트 엔트리들에 대한 하나 이상의 선택스 요소는 비트스트림으로부터 빠져 있고, 리스트 엔트리들 중 둘에 대한 값이 추론된다.

[0011]

본 명세서에 기술된 혁신들 중 또 다른 측면에 따르면, 비디오 인코더는 또 다른 조건을 평가한다. 평가의 결과에 따라, 인코더는 RPL을 어떻게 변경(예를 들어, RPL을 대체, RPL을 조정)할 것인지를 나타내는 리스트 엔트리들에 대한 하나 이상의 선택스 요소를 비트스트림에 시그널링하는 것을 조정한다. 특히, 하나 이상의 선택스 요소 중 적어도 하나의 길이(비트들)가 조정된다. 대응하는 비디오 디코더는 조건을 평가한다. 평가의 결과에 따라, 디코더는 RPL를 어떻게 변경할 것인지(이 경우, 하나 이상의 선택스 요소 중 적어도 하나의 길이(비트

들)가 조정된다)를 나타내는 리스트 엔트리들에 대한 하나 이상의 선택스 요소를 비트스트림으로부터 파싱하는 것을 조정한다. 예를 들어, 조건은 가중치 부여된 예측이 디스에이블링되는지 여부에 적어도 부분적으로 의존한다. 현재 슬라이스가 P 슬라이스인지 또는 B 슬라이스인지에 따라 및/또는 어떤 RPL이 시그널링/파싱되는지에 따라 가중치 부여된 예측이 디스에이블링되는지 여부를 확인하는데 다른 로직이 사용될 수 있다. 몇몇 예시적인 구현에서, 리스트 엔트리들에 대한 인덱스 i에 대해, 가중치 부여된 예측이 디스에이블링되는 경우, 선택스 요소들 중 적어도 하나의 길이(비트들)는 i가 증가함에 따라 감소한다. 예를 들어, 몇몇 예시적인 구현에서, 가중치 부여된 예측이 디스에이블링되는 경우, 리스트 엔트리[i]에 대한 주어진 선택스 요소의 길이는  $\text{Ceil}(\text{Log}_2(\text{NumPocTotalCurr}-i))$  비트이다. 한편, 가중치 부여된 예측이 인에이블링되는 경우, 리스트 엔트리[i]에 대한 주어진 선택스 요소의 길이는  $\text{Ceil}(\text{Log}_2(\text{NumPocTotalCurr}))$  비트이다.

[0012] 인코딩 또는 디코딩은, 방법의 일부로서, 이 방법을 수행하도록 적응된 컴퓨팅 장치의 일부로서, 또는 컴퓨팅 장치로 하여금 방법을 수행하게 하는 컴퓨터 실행가능 명령어들을 저장하는 유형의 컴퓨터 판독가능 매체의 일부로서 구현될 수 있다.

[0013] 본 발명의 전술한 및 다른 목적, 특징 및 장점은 첨부한 도면을 참조하여 진행되는 후속하는 상세한 설명으로부터 보다 명확해질 것이다.

### 도면의 간단한 설명

[0014] 도 1은 일부 실시예가 구현될 수 있는 예시적 컴퓨팅 시스템의 도면이다.

도 2a 및 2b는 일부 실시예가 구현될 수 있는 예시적 네트워크 환경의 도면이다.

도 3은 일부 실시예가 구현될 수 있는 예시적 인코더 시스템의 도면이다.

도 4는 일부 실시예가 구현될 수 있는 예시적 디코더 시스템의 도면이다.

도 5는 일부 실시예가 구현될 수 있는 예시적 비디오 인코더를 도시하는 도면이다.

도 6는 일부 실시예가 구현될 수 있는 예시적 비디오 디코더를 도시하는 도면이다.

도 7a은 몇몇 예시적인 구현에 따라, RPL이 변경되는지를 나타내는 플래그의 조건부적 시그널링을 나타내는 표이다.

도 7b 및 도 7c는 다른 예시적인 구현에 따라, RPL이 변경되는지를 나타내는 하나 이상의 플래그의 조건부적 시그널링을 나타내는 표이다.

도 8 및 도 9는 몇몇 예시적인 구현에 따라, RPL을 어떻게 변경하는지를 나타내는 리스트 엔트리들에 대한 선택스 요소들의 조건부적 시그널링을 나타내는 표이다.

도 10 및 도 11은 RPL이 변경되는지를 나타내는 플래그의 조건부적 시그널링 및 파싱을 위한 일반화된 기법들을 제각각 나타내는 흐름도이다.

도 12 및 도 13은 RPL을 어떻게 변경하는지를 나타내는 리스트 엔트리들에 대한 선택스 요소들의 조건부적 시그널링 및 파싱을 위한 일반화된 기법들을 제각각 나타내는 흐름도이다.

도 14 및 도 15는 RPL을 어떻게 변경하는지를 나타내는 리스트 엔트리들에 대한 선택스 요소들의 시그널링 및 파싱을 조정하기 위한 일반화된 기법들을 제각각 나타내는 흐름도이다.

### 발명을 실시하기 위한 구체적인 내용

[0015] 상세한 설명은 참조 화상 리스트("RPL") 변경 정보의 시그널링의 혁신을 제공한다. 이를 혁신은 RPL 변경 정보가 사용될 수 없거나 이러한 정보의 값이 추론될 수 있는 경우 RPL 변경 정보의 시그널링을 피하는 것을 도울 수 있다.

[0016] 몇몇 최근 코덱 표준에서, 참조 화상 세트("RPS")는 모션 보상 예측에 사용될 수 있는 참조 화상들의 세트이고, RPL은 RPS로부터 구성된다. 예측("P") 슬라이스의 디코딩 프로세스의 경우, RPL 0으로 불리는 하나의 RPL이 존재한다. 쌍방향 예측("B") 슬라이스의 디코딩 프로세스의 경우, RPL 0 및 RPL 1로 불리는 2개의 RPL이 존재한다. P 슬라이스의 디코딩 프로세스의 시작시, RPL 0에 대한 이용가능 정보(예를 들어, 현재 화상을 디코딩하는 디코더에서 참조 화상들의 세트가 이용가능함), 규칙에 따른 변경 및/또는 비트스트림 내에서 시그널링된 변경

으로부터 RPL 0이 도출된다. 유사하게, B 슬라이스의 디코딩 프로세스의 시작시에, RPL 0에 대한 이용가능 정보 및 RPL 1에 대한 이용가능 정보(예를 들어, 현재 화상을 디코딩하는 디코더에서 참조 화상들의 세트가 이용 가능함), 규칙에 따른 변경 및/또는 비트스트림 내에서 시그널링된 변경으로부터 RPL 0 및 RPL 1이 도출된다. 보다 일반적으로, RPL은 RPL에 대한 이용가능 정보, 규칙에 따른 변경 및/또는 비트스트림 내에서 시그널링된 변경에 기초하여 인코딩 및 디코딩 동안 구성된다. RPL에 대한 변경의 시그널링은 상당한 양의 비트를 소비할 수 있다. 몇몇 최근의 코덱 표준의 경우, RPL 변경 정보가 시그널링되는 방식에 있어 비효율성이 존재한다.

[0017] 상세한 설명은 RPL 변경 정보를 시그널링하는 분야에서 다양한 혁신을 제공한다. 몇몇 상황에서, 이를 혁신은 RPL 변경 정보에 대한 선택스 요소들의 보다 효율적인 시그널링을 제공한다. 예를 들어, 상세한 설명은 RPL을 어떻게 변경할지를 나타내는 리스트 엔트리에 대한 선택스 요소들의 조건부적 시그널링을 설명한다. 상세한 설명은 또한 이러한 선택스 요소들을 시그널링하는데 보다 적은 수의 비트를 사용하는 방식을 설명한다. 또 다른 예로서, 상세한 설명은 RPL이 변경되는지 여부를 나타내는 플래그의 조건부적 시그널링을 설명한다.

[0018] 몇몇 예시적 구현에서, RPL이 변경되지 않는 경우, RPS로부터 RPL 구성에 대한 규칙을 사용하여 "암시적" 접근 방식에 따라 기본 RPL이 구성된다. RPL이 변경되는 경우, RPS로부터 참조 화상들의 선택을 나타내는 시그널링된 RPL 변경 정보를 사용하여 "명시적" 시그널링 접근방식에 따라 대체 RPL이 구성된다. 이와 달리, 참조 화상을 재배열, 추가 또는 기본 RPL로부터의 참조 화상을 제거하는 변경은 기본 RPL에 대한 특정 변경으로서 보다 정밀한 방식으로 시그널링될 수 있다.

[0019] 본 명세서에서 설명되는 혁신들 중 일부는 HEVC 표준에 특정한 선택스 요소 및 동작을 참조하여 설명된다. 예를 들어, HEVC 표준의 드래프트 버전 JCTVC-I1003 - "High efficiency video coding (HEVC) text specification draft 7", JCTVC-I1003\_d5, 9<sup>th</sup> meeting of the Joint Collaborative Team on Video Coding ("JCT-VC"), Geneva, April 2012을 참조한다. 또한, 드래프트 버전 "High Efficiency Video Coding (HEVC) text specification draft 9," JCTVC-K1003\_d11, 11<sup>th</sup> meeting of the JCT-VC, Shanghai, October 2012를 참조한다. 본 명세서에서 설명되는 혁신들은 다른 표준 또는 포맷에 대해서도 구현될 수 있다.

[0020] 보다 일반적으로, 본 명세서에서 설명되는 예에 대한 다양한 대안이 가능하다. 예를 들어, 본 명세서에서 설명되는 방법 중 몇몇 방법은 설명되어 있는 방법의 동작의 순서를 변경하거나, 특정 방법의 동작을 분할, 반복 또는 생략하는 등에 의해 변경될 수 있다. 개시된 기술의 다양한 형태는 조합하여 사용되거나 개별적으로 사용될 수 있다. 상이한 실시예가 설명되는 혁신들 중 하나 이상을 사용한다. 본 명세서에서 설명되는 혁신들 중 일부는 전술한 배경기술 단락에서 설명되는 문제점 중 하나 이상을 해결한다. 통상적으로, 주어진 기술/툴(too l)은 이러한 문제 전부를 해결하지는 못한다.

## I. 예시적 컴퓨팅 시스템

[0021] 도 1은 설명되는 혁신들 중 몇몇이 구현되기에 적합한 컴퓨팅 시스템(100)의 일반적인 예를 도시하고 있다. 혁신들은 다양한 범용 컴퓨팅 시스템 또는 특수 목적 컴퓨팅 시스템에서 구현될 수 있으므로, 컴퓨팅 시스템(100)은 사용 또는 기능의 범위에 관해 어떠한 제한도 두려하지 않는다.

[0022] 도 1을 참조하면, 컴퓨팅 시스템(100)은 하나 이상의 프로세싱 유닛(110, 115) 및 메모리(120, 125)를 포함한다. 도 1에서, 이러한 가장 기본적인 구성(130)은 파선 내에 포함된다. 프로세싱 유닛(110, 115)은 컴퓨터로 실행 가능한 명령어들을 실행한다. 프로세싱 유닛은 범용 CPU(central processing unit), ASIC(application-specific integrated circuit) 내의 프로세서, 또는 임의의 다른 유형의 프로세서일 수 있다. 멀티프로세싱 시스템에서, 프로세싱 파워를 증가시키기 위해 다수의 프로세싱 유닛이 컴퓨터로 실행 가능한 명령어들을 실행한다. 예를 들어, 도 1은 중앙 처리 유닛(110)과 함께 그래픽 처리 유닛 또는 코프로세싱 유닛(co-processing unit, 115)을 도시하고 있다. 유형의(tangible) 메모리(120, 125)는 휘발성 메모리(가령, 레지스터, 캐시, RAM), 비휘발성 메모리(가령, ROM, EEPROM, 플래시 메모리 등), 또는 이들 둘의 어떤 조합일 수 있으며, 프로세싱 유닛(들)에 의해 액세스될 수 있다. 메모리(120, 125)는 프로세싱 유닛(들)에 의해 실행 되기에 적합한 컴퓨터로 실행 가능한 명령어들의 형태로, RPL 변경 정보를 시그널링하기 위한 하나 이상의 혁신을 구현하는 소프트웨어(180)를 저장한다.

[0023] 컴퓨팅 시스템은 추가적인 기능을 가질 수 있다. 예를 들어, 컴퓨팅 시스템(100)은 스토리지(140), 하나 이상의 입력 디바이스(150), 하나 이상의 출력 디바이스(160), 및 하나 이상의 통신 접속부(170)를 포함한다. 버스, 컨트롤러 또는 네트워크와 같은 상호접속 메커니즘(미도시)은 컴퓨팅 시스템(100)의 컴퓨트들을 상호접속시킨다. 통상적으로, 운영 시스템 소프트웨어(미도시)는 컴퓨팅 시스템(100)에서 실행되는 다른 소프트웨어

에 대한 동작 환경을 제공하며, 컴퓨팅 시스템(100)의 컴포넌트들의 동작들을 조정한다(coordinates).

[0025] 유형의 스토리지(140)는 제거 가능할 수도 제거 가능하지 않을 수도 있으며, 자기 디스크, 자기 테이프 또는 카세트, CD-ROM, DVD, 또는 정보를 저장하는데 사용될 수 있고 컴퓨팅 시스템(100) 내에서 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 스토리지(140)는 RPL 변경 정보를 시그널링하는 하나 이상의 혁신을 구현하는 소프트웨어(180)의 명령어들을 저장한다.

[0026] 입력 디바이스(들)(150)는 키보드, 마우스, 펜 또는 트랙볼(trackball)과 같은 터치 입력 디바이스, 음성 입력 디바이스, 스캐닝 디바이스, 또는 컴퓨팅 시스템(100)에 입력을 제공하는 다른 디바이스일 수 있다. 비디오 인코딩에 있어서, 입력 디바이스(들)(150)는 카메라, 비디오 카드, TV 투너 카드, 또는 아날로그 또는 디지털 형태로 비디오 입력을 수신하는 유사한 디바이스, 또는 컴퓨팅 시스템(100)으로 비디오 샘플을 판독하는 CD-ROM 또는 CD-RW일 수 있다. 출력 디바이스(들)(160)는 디스플레이, 프린터, 스피커, CD-라이터(writer), 또는 컴퓨팅 시스템(100)으로부터의 출력을 제공하는 다른 디바이스일 수 있다.

[0027] 통신 접속부(들)(170)는 통신 매체를 통해 다른 컴퓨팅 개체로의 통신을 가능하게 한다. 통신 매체는 컴퓨터로 실행 가능한 명령어들, 오디오 또는 비디오 입력 또는 출력, 또는 기타 데이터와 같은 정보를 변조된 데이터 신호를 통해 전달한다. 변조된 데이터 신호는 신호 내에 정보를 인코딩하기 위해 자신의 특성을 중 하나 이상을 설정 또는 변경한 신호이다. 제한적이 아닌 예시적으로, 통신 매체는 전기, 광학, RF, 또는 기타 캐리어를 사용할 수 있다.

[0028] 혁신들은 컴퓨터로 판독 가능한 매체의 일반적인 맥락에서 설명될 수 있다. 컴퓨터로 판독 가능한 매체는 컴퓨팅 환경 내에서 액세스될 수 있는 임의의 이용 가능한 유형의 매체이다. 제한적인 아닌 예시적으로, 이 컴퓨팅 시스템(100)에서, 컴퓨터로 판독 가능한 저장 매체는 메모리(120, 125), 스토리지(140) 및 이들의 임의의 조합을 포함한다.

[0029] 혁신들은 컴퓨팅 시스템에서 실제 또는 가상의 타겟 프로세서 상에서 실행되는 프로그램 모듈에 포함되는 것과 같은, 컴퓨터로 실행 가능한 명령어의 일반적인 맥락에서 설명될 수 있다. 일반적으로, 프로그램 모듈은, 특정 작업을 수행하거나 특정 추상적 데이터 유형(abstract data types)을 구현하는 루틴, 프로그램, 라이브러리, 오브젝트, 클래스, 컴포넌트, 데이터 구조 등을 포함한다. 프로그램 모듈의 기능은 다양한 실시예에서 원하는 대로 프로그램 모듈들 사이에서 조합되거나 분할될 수 있다. 프로그램 모듈을 위한 컴퓨터로 실행 가능한 명령어는 로컬 또는 분산형 컴퓨팅 시스템 내에서 실행될 수 있다.

[0030] 본 명세서에서, "시스템"이라는 용어와 "디바이스"라는 용어는 서로 바꾸어 사용된다. 맥락상으로 명백하게 표시되지 않는 한, 두 용어는 컴퓨팅 시스템 또는 컴퓨팅 디바이스의 유형에 어떠한 제한도 암시하지 않는다. 일반적으로, 컴퓨팅 시스템 또는 컴퓨팅 디바이스는 로컬 또는 분산형일 수 있으며, 특수 목적 하드웨어 및/또는 범용 하드웨어와, 본 명세서에서 설명되는 기능을 구현하는 소프트웨어와의 임의의 조합을 포함할 수 있다.

[0031] 개시되는 방법들은 이를 중 임의의 방법을 수행하도록 구성되는 전용 컴퓨팅 하드웨어를 사용하여 구현될 수도 있다. 예를 들어, 개시되는 방법들은 이를 중 임의의 방법을 구현하도록 특수하게 설계되거나 구성되는 집적 회로(가령, 애플리케이션 특정 집적 회로("ASIC")(예컨대, ASIC 디지털 신호 처리 유닛("DSP"), 그래픽 처리 유닛("GPU")), 또는 필드 프로그램가능 게이트 어레이("FPGA")와 같은 프로그램가능 로직 디바이스("PLD"))에 의해 구현될 수 있다.

[0032] 설명의 목적을 위해, 상세한 설명에서는 컴퓨팅 시스템에서의 컴퓨터 동작을 기술하기 위해 "결정한다" 및 "사용한다"와 같은 용어들이 사용된다. 이를 용어는 컴퓨터에 의해 수행되는 동작에 대한 고수준의 추상적 개념(high-level abstractions)이며, 인간에 의해 수행되는 행위와 혼동해서는 안된다. 이를 용어에 대응하는 실제 컴퓨터 동작은 구현에 따라 달라진다.

## II. 예시적 네트워크 환경

[0033] 도 2a 및 도 2b는 비디오 인코더(220) 및 비디오 디코더(270)를 포함하는 예시적 네트워크 환경(201, 202)을 도시하고 있다. 인코더(220) 및 디코더(270)는 적합한 통신 프로토콜을 사용하여 네트워크(250)를 통해 접속된다. 네트워크(250)는 인터넷 또는 다른 컴퓨터 네트워크를 포함할 수 있다.

[0034] 도 2a에 도시된 네트워크 환경(201)에서, 각 실시간 통신("RTC") 털(210)은 양방향 통신을 위해 인코더(220) 및 디코더(270) 모두를 포함한다. 소정의 인코더(220)는, SMPTE 421M 표준, (H.264 또는 AVC로도 알려진) ISO-IEC 14496-10 표준, HEVC 표준, 다른 표준 또는 전유 포맷(proprietary format)에 따라 출력을 생성할 수 있는

한편, 대응하는 디코더(270)는 이 인코더(220)로부터 인코딩된 데이터를 수신한다. 양방향 통신은, 영상 회의, 영상 통화 또는 기타 양자간 통신 시나리오의 일부일 수 있다. 도 2a의 네트워크 환경(201)은 2개의 실시간 통신 툴(210)을 포함하지만, 네트워크 환경(201)은 대신에 다자간 통신에 참가하는 3개 이상의 실시간 통신 툴(210)을 포함할 수도 있다.

[0036] 실시간 통신 툴(210)은 인코더(220)에 의한 인코딩을 관리한다. 도 3은 실시간 통신 툴(210)에 포함될 수 있는 예시적 인코더 시스템(300)을 도시하고 있다. 이와 달리, 실시간 통신 툴(210)은 다른 인코더 시스템을 사용할 수 있다. 실시간 통신 툴(210)은 또한 디코더(270)에 의한 디코딩을 관리한다. 도 4는 실시간 통신 툴(210)에 포함될 수 있는 예시적 디코더 시스템(400)을 도시하고 있다. 이와 달리, 실시간 통신 툴(210)은 다른 디코더 시스템을 사용할 수 있다.

[0037] 도 2b에 도시된 네트워크 환경(202)에서, 인코딩 툴(212)은 디코더(270)를 포함하는 다수의 재생 툴(playback tools, 214)로 전송하기 위해 비디오를 인코딩하는 인코더(220)를 포함한다. 비디오 감시 시스템, 웹 카메라 모니터링 시스템, 원격 데스크톱 회의 프리젠테이션, 또는 비디오가 인코딩되어 한 위치로부터 하나 이상의 다른 위치로 송신되는 다른 시나리오를 위해, 단방향 통신이 제공될 수 있다. 도 2b에 도시된 네트워크 환경(202)은 2개의 재생 툴(214)을 포함하지만, 네트워크 환경(202)은 더 많거나 적은 수의 재생 툴(214)을 포함할 수 있다. 일반적으로, 재생 툴(214)은 인코딩 툴(212)과 통신하여 재생 툴(214)이 수신할 비디오 스트림을 결정한다. 재생 툴(214)은 비디오 스트림을 수신하고, 수신된 인코딩된 데이터를 적합한 시간 동안 버퍼링하고, 디코딩 및 재생을 시작한다.

[0038] 도 3은 인코딩 툴(212)에 포함될 수 있는 예시적 인코더 시스템(300)을 도시하고 있다. 이와 달리, 인코딩 툴(212)은 다른 인코더 시스템을 사용할 수 있다. 인코딩 툴(21)은 하나 이상의 재생 툴(214)과의 접속을 관리하기 위한 서버측 콘트롤러 로직을 포함할 수도 있다. 도 4는 재생 툴(214)에 포함될 수 있는 예시적 디코더 시스템(400)을 도시하고 있다. 이와 달리, 재생 툴(214)은 다른 디코더 시스템을 사용할 수 있다. 재생 툴(214)은 인코딩 툴(212)과의 접속을 관리하기 위한 클라이언트측 콘트롤러 로직을 포함할 수도 있다.

### III. 예시적 인코더 시스템

[0040] 도 3은 일부 실시예가 구현될 수 있는 예시적 인코더 시스템(300)의 블록도이다. 인코더 시스템(300)은 실시간 통신용 로우-레이턴시(low-latency) 인코딩 모드, 트랜스코딩 모드(transcoding mode), 파일 또는 스트림으로부터의 미디어 재생을 위한 보통의 인코딩 모드와 같은 다수의 인코딩 모드들 중 임의의 모드에서 동작할 수 있는 범용 인코딩 툴이거나, 이러한 인코딩 모드 중 하나의 모드에 적응된 특수 목적 인코딩 툴일 수 있다. 인코더 시스템(300)은, 운영 시스템 모듈로서, 애플리케이션 라이브러리의 일부로서 또는 독립적 애플리케이션으로서 구현될 수 있다. 전체적으로, 인코더 시스템(300)은 비디오 소스(310)로부터 소스 비디오 프레임(311)의 시퀀스를 수신하고, 채널(390)로의 출력으로서 인코딩된 데이터를 생성한다. 채널로 출력되는 인코딩된 데이터는 RPL 변경 정보를 표시하는 선택스 요소를 포함할 수 있다.

[0041] 비디오 소스(310)는 카메라, 투너 카드, 저장매체 또는 기타 디지털 비디오 소스일 수 있다. 비디오 소스(310)는, 가령, 초당 30 프레임의 프레임 레이트로 비디오 프레임의 시퀀스를 생성한다. 본 명세서에서 사용되는 바와 같이, "프레임"이라는 용어는 일반적으로 소스, 코딩되거나 재구성된 이미지 데이터를 지칭한다. 프로그레시브 비디오(progressive video)에 있어서, 프레임은 프로그레시브 비디오 프레임이다. 인터레이스드 비디오(interlaced video)에 있어서, 예시적 실시예에서, 인터레이스드 비디오 프레임은 인코딩에 앞서 인터레이싱이 해제된다(de-interlaced). 이와 달리, 2개의 상보적 인터레이스드 비디오 필드가 인터레이스드 비디오 프레임 또는 별도의 필드로서 인코딩될 수 있다. 프로그레시브 비디오 프레임을 나타내는 것 외에도, "프레임"이라는 용어는 쌍을 이루지 않는 단일 비디오 필드, 상보적인 비디오 필드 쌍, 주어진 시각에서 비디오 오브젝트를 표현하는 비디오 오브젝트 평면, 또는 더 큰 이미지에서의 관심 영역을 표시할 수 있다. 비디오 오브젝트 평면 또는 영역은 한 장면의 다수의 오브젝트 또는 영역을 포함하는 보다 큰 이미지의 일부일 수 있다.

[0042] 도달하는 소스 프레임(311)은 다수의 프레임 버퍼 저장 영역(321, 322, …, 32n)을 포함하는 소스 프레임 임시 메모리 저장 영역(320)에 저장된다. 프레임 버퍼(321, 322 등)는 하나의 소스 프레임을 소스 프레임 저장 영역(320)에 유지한다. 소스 프레임(311) 중 하나 이상이 프레임 버퍼(321, 322 등)에 저장된 후, 프레임 셀렉터(330)가 소스 프레임 저장 영역(320)으로부터 개개의 소스 프레임을 주기적으로 선택한다. 인코더(340)에 입력하기 위해 프레임 셀렉터(330)에 의해 프레임이 선택되는 순서는, 비디오 소스(310)에 의해 프레임이 생성되는 순서와는 상이할 수 있는데, 가령, 시간적 역방향 예측(temporally backward prediction)을 돋기 위해 하나의 프레임이 순서상 앞설 수 있다. 인코더(340) 이전에, 인코더 시스템(300)은 인코딩에 앞서서 프레임의 사전 처

리(가령, 필터링)를 수행하는 프리-프로세서(pre-processor)(도시 생략)를 포함할 수 있다. 프리-프로세싱은 또한 인코딩을 위한 주요 성분 및 보조 성분으로의 컬러 공간 변환을 포함할 수 있다.

[0043] 인코더(340)는 선택된 프레임(331)을 인코딩하여 코딩된 프레임(341)을 생성하고, 또한 MMCO(memory management control operation) 신호(342) 또는 RPS(reference picture set) 정보를 생성한다. 현재 프레임이 인코딩된 첫 번째 프레임이 아닌 경우, 그 인코딩 프로세스를 수행할 때, 인코더(340)는 디코딩된 프레임 임시 메모리 저장 영역(360)에 저장된 하나 이상의 이전에 인코딩된/디코딩된 프레임(369)을 사용할 수 있다. 이러한 저장된 디코딩된 프레임(369)은 현재 소스 프레임(331)의 콘텐츠의 프레임간 예측(inter-frame prediction)을 위한 참조 프레임으로서 사용된다. 일반적으로, 인코더(340)는 모션 추정 및 보상, 주파수 변환, 양자화 및 엔트로피 코딩과 같은 인코딩 작업을 수행하는 다수의 인코딩 모듈을 포함한다. 인코더(340)에 의해 수행되는 정확한 동작은 압축 포맷에 따라 변할 수 있다. 출력되는 인코딩된 데이터의 포맷은 윈도우 미디어 비디오(Windows Media Video) 포맷, VC-1 포맷, MPEG-x 포맷(가령, MPEG-1, MPEG-2, 또는 MPEG-4), H.26x 포맷(가령, H.261, H.262, H.263, H.264), HEVC 포맷 또는 기타 포맷일 수 있다.

[0044] 예를 들어, 인코더(340) 내에서, 인터-코딩된(inter-coded), 예측된 프레임은 참조 화상의 예인 참조 프레임으로부터의 예측과 관련하여 표현된다. 모션 추정기는 하나 이상의 참조 프레임(369)에 대해 소스 프레임(341)의 샘플 블록 또는 기타 샘플 세트의 모션을 추정한다. 다수의 참조 프레임이 사용되는 경우, 이 다수의 참조 프레임은 상이한 시간적 방향 또는 동일한 시간적 방향으로부터의 것일 수 있다. 참조 프레임(참조 화상)은 하나 이상의 RPL의 일부일 수 있고, 참조 인덱스는 RPL(들) 내에서의 참조 화상을 어드레싱한다. RPL(들)은 새로운 참조 화상들이 적절할 때 추가되고, 모션 보상을 위해 더 이상 사용되지 않는 보다 이전의 참조 화상들이 적절할 때 제거되며, 참조 화상들이 적절할 때 재배열되도록 인코딩 동안 구성된다. 몇몇 구현에서, 예를 들어, 현재 화상을 인코딩하는 경우, 인코더(340)는 디코딩된 프레임 저장 영역(360) 내에 참조 화상들을 포함하는 RPS를 결정하고, 그런 다음 현재 화상의 주어진 슬라이스의 인코딩을 위한 하나 이상의 RPL을 생성한다. RPL은 RPS로부터 이용 가능한 참조 화상들의 선택에 대한 규칙을 적용함으로써 (암시적 방식으로) 생성될 수 있는데, 이 경우 RPL 변경 정보는 비트스트림 내에서 명시적으로 시그널링되지 않는다. 또는, RPL은 RPS로부터 이용 가능한 특정 참조 화상들 선택함으로써 생성될 수 있는데, 이 경우 선택된 참조 화상들은 비트스트림 내에서 시그널링된 RPL 변경 정보에 표시될 것이다. 암시적 접근방식의 규칙에 의해 구성될 수 있는 RPL에 비해, RPL 변경 정보는 대체 RPL을 RPS 내의 참조 화상들의 리스트로서 지정할 수 있다. 이와 달리, RPL 변경 정보는 암시적 접근방식의 규칙에 따라 구성된 RPL에서의 하나 이상의 참조 화상의 제거, 하나 이상의 참조 화상의 추가 및/또는 참조 화상의 재배열을 미세하게 지정할 수 있다.

[0045] 인터-코딩된 프레임을 인코딩하는 경우, 인코더(340)는 RPL이 비트스트림 내에 명시적으로 시그널링된 선택된 요소에 따라 변경되지 않는 모션 보상의 결과를 평가할 수 있고, 또한 RPL이 비트스트림 내에 명시적으로 시그널링된 선택된 요소에 따라 변경되는 모션 보상의 결과(또는 RPL을 변경하는 다수의 다른 방식의 결과)를 평가한다. 인코더(340)는 기본 RPL(비트스트림 내에 RPL 변경 정보가 시그널링되지 않음) 또는 변경된 RPL(비트스트림 내에 RPL 변경 정보가 시그널링됨)을 사용하기로 결정할 수 있다. RPL이 기본 RPL에 비해 변경(예를 들어, 대체, 조정)되는 경우, 인코더(340)는 (a) 참조 인덱스를 이용한 보다 효율적인 어드레싱을 위해 참조 화상들을 재배열하는 것, (b) 인코딩 동안의 사용 빈도에 적어도 부분적으로 기초하여 참조 화상들을 제거하는 것, (c) 인코딩 동안의 사용 빈도에 적어도 부분적으로 기초하여 참조 화상을 추가하는 것 중 하나 이상을 수행할 수 있다. 예를 들어, 인코더(340)는 모션 보상을 위한 참조 화상의 이용이 임계량 아래에 떨어지고 및/또는 다른 기준에 부합한 이후 RPL로부터 주어진 참조 화상을 제거하기로 결정할 수 있다. 또 다른 예로서, 인코더(340)는 모션 보상을 위한 참조 화상의 이용이 임계량 위에 있고 및/또는 다른 기준에 부합한 이후 RPL에 주어진 참조 화상을 추가하기로 결정할 수 있다. 또 다른 예로서, 인코더(340)는 제각기의 참조 화상의 이용 빈도에 기초하여 및/또는 다른 기준에 따라 RPL 내에서 참조 화상들을 재배열하는 방식을 결정할 수 있다.

[0046] 모션 추정기는 모션 벡터 정보와 같은 모션 정보를 출력하는데, 이 정보는 엔트로피 코딩된다. 모션 보상기는 모션 벡터를 참조 화상에 적용하여 모션 보상 예측 값을 결정한다. 인코더는 블록의 모션 보상 예측 값과 대응 원본 값 사이의 차를 (존재하는 경우) 결정한다. 이를 예측 잔여 값은 주파수 변환, 양자화 및 엔트로피 인코딩을 사용하여 더 인코딩된다. 유사하게, 인트라 예측(intra prediction)에 있어서, 인코더(340)는 블록에 대한 인트라 예측 값을 결정하고, 예측 잔여 값을 결정하며, (주파수 변환, 양자화 및 엔트로피 인코딩을 사용하여) 예측 잔여 값을 인코딩할 수 있다. 특히, 인코더(340)의 엔트로피 코더는 양자화된 변환 계수 값뿐만 아니라 소정 부가 정보(가령, 모션 벡터 정보, QP 값, 모드 결정, 파라미터 선택, 참조 인덱스, RPL 변경 정보)도 압축한다. 통상적인 엔트로피 코딩 기술은 Exp-Golomb 코딩, 산술 코딩(arithmetic coding), 차분 코딩

(differential coding), 헤프만 코딩, 런 길이 코딩(run length coding), V2V (variable-length-to-variable-length) 코딩, V2F(variable-length-to-fixed-length) 코딩, LZ 코딩, 사전 코딩(dictionary coding), PIPE(probability interval partitioning entropy) 코딩 및 이들의 조합을 포함한다. 엔트로피 코더는 정보의 상이한 종류에 따라 상이한 코딩 기술을 사용할 수 있으며, 한 특정 코딩 기술 내에서 다수의 코드 테이블로부터 선택할 수 있다.

[0047] 코딩된 프레임(341) 및 MMCO/RPS 정보(342)는 디코딩 프로세스 에뮬레이터(350)에 의해 처리된다. 디코딩 프로세스 에뮬레이터(350)는 디코더의 기능 중 일부, 예를 들어, 모션 보상에서 인코더(340)에 의해 사용되는 참조화상을 재구성하는 디코딩 작업을 구현한다. 디코딩 프로세스 에뮬레이터(350)는 MMCO/RPS 정보(342)를 사용하여 특정한 코딩된 프레임(341)이 인코딩될 후속 프레임의 인터-프레임 예측에서 기준 프레임으로서 사용되기 위해 재구성되고 저장되어야 하는지를 판단한다. MMCO/RPS 정보(342)가 코딩된 프레임(341)이 저장되어야 한다고 표시하면, 디코딩 프로세스 에뮬레이터(350)는 코딩된 프레임(341)을 수신하고 대응하는 디코딩된 프레임(351)을 생성하는 디코더에 의해 수행될 디코딩 프로세스를 모델링한다. 그러한 과정에서, 인코더(340)가 디코딩된 프레임 저장 영역(360)에 저장된 디코딩된 프레임(들)(369)을 사용한 경우, 디코딩 프로세스 에뮬레이터(350)는 또한 디코딩 프로세스의 일부로서 저장 영역(360)으로부터 디코딩된 프레임(들)(369)을 사용한다.

[0048] 디코딩된 프레임 임시 메모리 저장 영역(360)은 다수의 프레임 버퍼 저장 영역(361, 362, …, 36n)을 포함한다. 디코딩 프로세스 에뮬레이터(350)는 인코더에 의해(340) 더 이상 참조 프레임으로서 사용될 필요가 없는 프레임을 갖는 프레임 버퍼(361, 362 등)를 식별하기 위해 MMCO/RPS 정보(342)를 사용하여 저장 영역(360)의 콘텐츠 관리한다. 디코딩 프로세스를 모델링한 후, 디코딩 프로세스 에뮬레이터(350)는 이 방식으로 식별된 프레임 버퍼(361, 362 등)에 새롭게 디코딩된 프레임(351)을 저장한다.

[0049] 코딩된 프레임(341) 및 MMCO/RPS 정보(342)는 또한 임시 코딩된 데이터 영역(370)에서 버퍼링된다. 코딩된 데이터 영역(370)에서 수집되는 코딩된 데이터는, 기본적인(elementary) 코딩된 비디오 스트림의 신택스의 일부로서, RPL 변경 정보를 나타내는 신택스 요소를 포함할 수 있다. 코딩된 데이터 영역(370)에서 수집된 코딩된 데이터는, 코딩된 비디오 데이터에 관한 미디어 메타데이터(가령, 하나 이상의 SEI(supplemental enhancement information) 메시지 또는 VUI(video usability information) 메시지 내의 하나 이상의 파라미터)를 포함할 수도 있다.

[0050] 임시 코딩된 데이터 영역(370)으로부터의 수집된 데이터(371)는 채널 인코더(380)에 의해 처리된다. 채널 인코더(380)는 (가령, ISO/IEC 14496-12와 같은 미디어 콘테이너 포맷에 따라) 미디어 스트림으로서 전송을 위해 수집된 데이터를 패킷화하는데, 이 경우, 채널 인코더(380)는 미디어 전송 스트림의 신택스의 일부로서 신택스 요소를 추가할 수 있다. 또는, 채널 인코더(380)는 (가령, ISO/IEC 14496-12와 같은 미디어 콘테이너 포맷에 따라) 파일로서 저장하기 위해 수집된 데이터를 조직화(organize) 수 있는데, 이 경우, 채널 인코더(380)는 미디어 저장 파일의 신택스의 일부로서 신택스 요소를 추가할 수 있다. 또는, 보다 일반적으로, 채널 인코더(380)는 하나 이상의 미디어 시스템 멀티플렉싱 프로토콜 또는 전송 프로토콜을 구현할 수 있는데, 이 경우, 채널 인코더(380)는 프로토콜(들)의 신택스의 일부로서 신택스 요소를 추가할 수 있다. 채널 인코더(380)는 채널(390)로의 출력을 제공하는데, 이 채널은 그 출력에 대한 저장, 통신 접속 또는 다른 채널을 나타낸다.

#### IV. 예시적 디코더 시스템

[0052] 도 4는 일부 실시예와 함께 구현될 수 있는 예시적 디코더 시스템(400)의 블록도이다. 디코더 시스템(400)은 실시간 통신용 로우-레이턴시(low-latency) 디코딩 모드 및 파일 또는 스트림으로부터의 미디어 재생을 위한 보통의 디코딩 모드와 같은 다수의 디코딩 모드들 중 임의의 디코딩 모드에서 동작할 수 있는 범용 디코딩 툴일 수 있거나, 또는, 이러한 디코딩 모드 중 하나에 적응된 특수 목적 디코딩 툴일 수 있다. 디코더 시스템(400)은, 운영 시스템 모듈로서, 애플리케이션 라이브러리의 일부로서 또는 단독 애플리케이션으로서 구현될 수 있다. 전체적으로, 디코더 시스템(400)은 채널(410)로부터 코딩된 데이터를 수신하고, 출력 목적지(490)에 대한 출력으로서 재구성된 프레임을 생성한다. 코딩된 데이터는 RPL 변경 정보를 나타내는 신택스 요소를 포함할 수 있다.

[0053] 디코더 시스템(400)은 채널(410)을 포함하는데, 이 채널은 입력으로서 코딩된 데이터에 대한 저장, 통신 접속 또는 다른 채널을 나타낸다. 채널(410)은 채널 코딩된 데이터를 생성한다. 채널 디코더(420)는 코딩된 데이터를 처리할 수 있다. 예를 들어, 채널 디코더(420)는 (가령, ISO/IEC 14496-12와 같은 미디어 콘테이너 포맷에 따라) 미디어 스트림으로서 전송하기 위해 수집된 데이터를 패킷 해제(de-packetizes)하는데, 이 경우, 채널 디코더(420)는 미디어 전송 스트림의 신택스의 일부로서 추가된 신택스 요소를 파싱할(parse) 수 있다. 또는, 채

널 디코더(420)는 (가령, ISO/IEC 14496-12와 같은 미디어 콘테이너 포맷에 따라) 파일로 저장하기 위해 수집된 코딩된 비디오 데이터를 분리하는데, 이 경우, 채널 디코더(420)는 미디어 저장 파일의 선택스의 일부로서 추가된 선택스 요소를 파싱할 수 있다. 또는, 보다 일반적으로, 채널 디코더(420)는 하나 이상의 미디어 시스템 디멀티플렉싱 프로토콜 또는 전송 프로토콜을 구현할 수 있는데, 이 경우, 채널 디코더(420)는 프로토콜(들)의 선택스의 일부로서 추가된 선택스 요소를 파싱할 수 있다.

[0054] 채널 디코더(420)로부터 출력되는 코딩된 데이터(421)는 충분한 양의 이러한 데이터가 수신될 때까지 임시 코딩된 데이터 영역(430)에 저장된다. 코딩된 데이터(421)는 코딩된 프레임(431) 및 MMCO/RPS 정보(432)를 포함한다. 코딩된 데이터 영역(430) 내의 코딩된 데이터(421)는 기본적인(elementary) 코딩된 비디오 비트스트림의 선택스의 일부로서, RPL 변경 정보를 나타내는 선택스 요소를 포함할 수 있다. 코딩된 데이터 영역(430) 내의 코딩된 데이터(421)는 또한 인코딩된 비디오 데이터에 관한 미디어 메타데이터(가령, 하나 이상의 SEI 메시지 또는 VUI 메시지 내의 하나 이상의 파라미터)를 포함할 수 있다. 일반적으로, 코딩된 데이터 영역(430)은 이러한 코딩된 데이터(421)가 디코더(450)에 의해 사용될 때까지 코딩된 데이터(421)를 임시로 저장한다. 이 때, 코딩된 프레임(431) 및 MMCO/RPS 정보(432)에 대한 코딩된 데이터는 코딩된 데이터 영역(430)으로부터 디코더(450)로 전달된다. 디코딩이 계속됨에 따라, 새로운 코딩된 데이터가 코딩된 데이터 영역(430)에 추가되고, 코딩된 데이터 영역(430)에 남아 있는 가장 오래된 코딩된 데이터가 디코더(450)로 전달된다.

[0055] 디코더(450)는 코딩된 프레임(431)을 주기적으로 디코딩하여 대응하는 디코딩된 프레임(451)을 생성한다. 적합한 경우, 디코딩 프로세스를 수행할 때, 디코더(450)는 인터-프레임 예측을 위한 참조 프레임(참조 화상)으로서 하나 이상의 이전에 디코딩된 프레임(469)을 사용할 수 있다. 디코더(450)는 디코딩된 프레임 임시 메모리 저장 영역(460)으로부터 이러한 이전에 디코딩된 프레임(469)을 판독한다. 일반적으로, 디코더(450)는 (RPL 변경 정보를 이용하여 RPL(들)을 생성할 수 있는) 엔트로피 디코딩, 역 양자화, 역 주파수 변환 및 모션 보상과 같은 디코딩 작업을 수행하는 다수의 디코딩 모듈을 포함한다. 디코더(450)에 의해 수행되는 정확한 동작은 압축 포맷에 따라 변할 수 있다.

[0056] 예를 들어, 디코더(450)는 압축된 프레임 또는 프레임 시퀀스에 대한 인코딩된 데이터를 수신하고, 디코딩된 프레임(451)을 포함하는 출력을 생성한다. 디코더(450)에서, 베퍼는 압축된 프레임에 대한 인코딩된 데이터를 수신하고, 수신된 인코딩된 데이터가 엔트로피 디코더에 대해 이용 가능하게 한다. 엔트로피 디코더는 통상적으로 인코더에서 수행되는 엔트로피 인코딩을 역으로 적용하여, 엔트로피 코딩된 양자화 데이터 및 엔트로피 코딩된 부가 정보(참조 인덱스, RPL 변경 정보 등을 포함함)를 엔트로피 디코딩한다. 디코더는 참조 화상들에 대한 하나 이상의 RPL을 구성하되, 참조 인덱스는 RPL(들) 내에서 참조 화상을 어드레싱한다. RPL(들)은 적절할 때 새로운 참조 화상이 추가되고, 적절할 때 모션 보상에 더 이상 사용되지 않는 보다 오래된 참조 화상이 제거되며, 적절할 때 참조 화상들이 재배열되도록 구성된다. 몇몇 실시예에서, 예를 들어, 현재의 화상을 디코딩하는 경우, 디코더(450)는 디코딩된 프레임 저장 영역(460) 내의 참조 화상을 포함하는 RPS를 결정하고, 현재 화상의 주어진 슬라이스의 디코딩을 위한 하나 이상의 RPL을 생성한다. RPL은 RPS로부터 이용 가능한 참조 화상들의 선택에 대한 규칙을 적용함으로써 생성될 수 있고, 이 경우, RPL 변경 정보는 비트스트림으로부터 파싱되지 않는다. 또는, RPL은 RPS로부터 이용 가능한 특정 참조 화상을 선택함으로써 생성될 수 있는데, 이 경우, 선택되는 참조 화상은 비트스트림으로부터 파싱되는 RPL 변경 정보에 표시된다. 암시적 접근방식의 규칙에 의해 구성될 수 있는 RPL과 대조적으로, RPL 변경 정보는 RPS 내의 참조 화상들의 리스트로서 대체 RPL을 지정할 수 있다. 이와 달리, RPL 변경 정보는 암시적 접근방식의 규칙에 의해 구성되는 RPL 내에서 하나 이상의 참조 화상의 제거, 하나 이상의 참조 화상의 추가 및/또는 참조 화상들의 재배열을 보다 정밀한 방식으로 지정할 수 있다.

[0057] 모션 보상기는 모션 정보를 하나 이상의 참조 화상에 적용하여, 재구성되는 프레임의 서브 블록 및/또는 블록(총칭하여 블록)의 모션 보상 예측을 형성한다. 인트라 예측 모듈은 인접한 이전에 재구성된 샘플 값으로부터 현재 블록의 샘플 값을 공간적으로 예측할 수 있다. 디코더(450)는 또한 예측 잔여값(residuals)을 재구성한다. 역 양자화기(inverse quantizer)는 엔트로피-디코딩된 데이터를 역 양자화한다. 역 주파수 변환기는 재구성된 주파수 영역 데이터를 공간 영역 정보로 변환한다. 예측된 프레임에 있어서, 디코더(450)는 재구성된 예측 잔여값을 모션 보상 예측과 결합하여 재구성된 프레임을 형성한다. 유사하게 디코더(450)는 예측 잔여값을 인트라 예측으로부터의 공간적 예측과 결합할 수 있다. 비디오 디코더(450)의 모션 보상 루프는, 디코딩된 프레임(451)에서 블록 경계 행 및/또는 열에 걸쳐 있는 불연속성을 완만하게 하기 위해 적응적 디-블로킹 필터(adaptive de-blocking filter)를 포함한다.

[0058] 디코딩된 프레임 임시 메모리 저장 영역(460)은 다수의 프레임 베퍼 저장 영역(461, 462, ..., 46n)을 포함한다. 디코딩된 프레임 저장 영역(460)은 DPB의 일례이다. 디코더(450)는 MMCO/RPS 정보(432)를 사용하여 디코딩된

프레임(451)을 저장할 수 있는 프레임 버퍼(461, 462 등)를 식별한다. 디코더(450)는 그 프레임 버퍼에 디코딩된 프레임(451)을 저장한다.

[0059] 출력 시퀀서(480)는 MMCO/RPS 정보(432)를 사용하여 출력 순서상 다음에 생성될 프레임이 디코딩된 프레임 저장 영역(460)에서 이용 가능한 때를 식별한다. 출력 순서상 다음에 생성될 프레임(481)이 디코딩된 프레임 저장 영역(460)에서 이용 가능한 때에, 이는 출력 시퀀서(480)에 의해 준비되어 출력 목적지(490)(가령, 디스플레이)에 출력된다. 일반적으로, 출력 시퀀서(480)에 의해 디코딩된 프레임 저장 영역(460)으로부터 프레임이 출력되는 순서는 디코더(450)에 의해 프레임이 디코딩되는 순서와는 상이할 수 있다.

## V. 예시적 비디오 인코더

[0061] 도 5는 일부 실시예와 함께 구현될 수 있는 일반화된 비디오 인코더(500)의 블록도이다. 인코더(500)는 현재 프레임(505)을 포함하는 비디오 프레임 시퀀스를 수신하고 인코딩된 데이터(595)를 출력으로서 생성한다.

[0062] 인코더(500)는 블록 기반이며 구현에 의존하는 블록 포맷을 사용한다. 블록은 상이한 스테이지, 가령, 주파수 변환 및 엔트로피 인코딩 스테이지에서 더 세분될 수 있다. 예를 들어, 프레임은 64 x 64 블록, 32 x 32 블록 또는 16 x 16 블록으로 분할될 수 있으며, 이는 이어서 코딩 및 디코딩을 위해 픽셀 값의 더 작은 블록 및 서브 블록으로 분할될 수 있다.

[0063] 인코더 시스템(500)은 예측된 프레임 및 인트라-코딩된 프레임을 압축한다. 설명을 위한 목적으로, 도 5는 인트라-프레임 코딩을 위한 인코더(500)를 통한 "인트라 경로" 및 인터-프레임 코딩을 위한 "인터 경로"를 도시하고 있다. 인코더(500)의 구성요소들 중 많은 구성요소들이 인트라-프레임 코딩 및 인터-프레임 코딩 모두를 위해 사용된다. 이를 구성요소들에 의해 수행되는 정확한 동작은 압축되는 정보의 유형에 따라 변할 수 있다.

[0064] 현재 프레임(505)이 예측된 프레임이면, 모션 추정기(510)는 하나 이상의 참조 프레임(참조 화상)에 대해 현재 프레임(505)의 블록, 서브 블록 또는 기타 픽셀 값 세트의 모션을 추정한다. 프레임 스토어(520)는 하나 이상의 재구성된 이전 프레임(525)을 참조 프레임(참조 화상)으로서 사용하기 위해 버퍼링한다. 다수의 참조 프레임이 사용되는 경우, 이 다수의 참조 프레임은 상이한 시간적 방향 또는 동일한 시간적 방향으로부터의 것일 수 있다. 다수의 참조 화상은 참조 인덱스를 통해 어드레싱되는 하나 이상의 RPL에 표현될 수 있다. 모션 추정기(510)는 부가 정보로서 차분 모션 벡터 정보와 같은 모션 정보(515), 참조 인덱스 및 RPL 변경 정보를 출력한다. 인코딩 동안, 인코더(500)는 RPL(들)에서 적절한 때에 새로운 참조 화상이 추가되고, 적절한 때에 모션 보상에 더 이상 사용되지 않는 보다 오래된 참조 화상이 제거되며, 적절한 때에 참조 화상들이 재배열되도록 RPL(들)을 구성한다.

[0065] 몇몇 구현에서, 현재의 프레임을 인코딩하는 경우, 인코더(500)는 프레임 스토어(520)의 참조 프레임을 포함하는 RPS를 결정한다. 인코더(500)는 전형적으로 프레임의 제1 슬라이스에 대한 RPS를 결정한다. 슬라이스 단위로, 인코더(500)는 현재 프레임의 주어진 슬라이스의 인코딩을 위한 하나 이상의 RPL을 생성한다. RPL을 생성하기 위해, 인코더(500)는 RPS로부터 이용 가능한 참조 프레임의 선택에 대한 규칙을 적용할 수 있는데, 이 경우 RPL 변경 정보는 인코딩된 데이터(595)에 명시적으로 시그널링되지 않는다. 또는, 인코더(500)는 RPS로부터 이용 가능한 특정 참조 프레임을 선택할 수 있는데, 이 경우 선택된 참조 프레임은 인코딩된 데이터(595)에 시그널링되는 RPL 변경 정보에 표시될 것이다. 암시적 접근방식의 규칙에 의해 구성될 수 있는 RPL에 비해, RPL 변경 정보는 RPS 내의 참조 화상들의 리스트로서 대체 RPL을 지정할 수 있다. 이와 달리, RPL 변경 정보는 규칙에 의해 암시적으로 구성되는 RPL에서 하나 이상의 참조 프레임의 제거, 하나 이상의 참조 프레임의 추가 및/또는 참조 프레임들의 재배열을 보다 정밀하게 지정할 수 있다.

[0066] 모션 보상기(530)는 모션-보상된 현재 프레임(535)을 형성할 때 재구성된 모션 벡터를 재구성된 참조 프레임(들)(525)에 적용한다. 모션-보상된 현재 프레임(535)의 서브 블록, 블록 등과 본래의 현재 프레임(505)의 대응 부분 사이의 차이는 (이러한 차이가 존재한다면) 서브 블록, 블록 등에 대한 예측 잔여값(545)이다. 현재 프레임을 추후에 재구성하는 동안, 재구성된 예측 잔여값은 모션-보상된 현재 프레임(535)에 추가되어 본래의 현재 프레임(505)에 근접한 재구성된 프레임을 얻는다. 그러나, 비가역 압축(lossy compression)에서는 여전히 본래의 현재 프레임(505)으로부터 일부 정보가 손실된다. 인트라 경로는, 이전에 재구성된 인접 픽셀 값들로부터 현재 블록 또는 서브 블록의 픽셀 값을 공간적으로 예측하는 인트라 예측 모듈(도시 생략)을 포함할 수 있다.

[0067] 주파수 변환기(560)는 공간 영역 비디오 정보를 주파수 영역(가령, 스펙트럼 변환) 데이터로 변환한다. 블록 기반 비디오 프레임에 있어서, 주파수 변환기(560)는 이산 코사인 변환(discrete cosine transform), 그 정수

근사치 계산(integer approximation), 또는 다른 유형의 순방향 블록 변환(forward block transform)을 픽셀 값 데이터 또는 예측 잔여 데이터의 블록 또는 서브 블록에 적용하여, 주파수 변환 계수의 블록/서브 블록을 생성한다. 그리고, 양자화기(570)는 변환 계수를 양자화한다. 예를 들어, 양자화기(570)는, 균일하지 않은 스칼라 양자화를, 프레임 단위, 슬라이스 단위, 블록 단위 또는 기타 단위에 기반하여 변하는 스텝 사이즈(step size)로 주파수 영역 데이터에 적용한다.

[0068] 후속 모션 추정/보상을 위해 현재 프레임의 재구성된 버전이 필요할 때, 역 양자화기(576)는 양자화된 주파수 계수 데이터에 역 양자화를 수행한다. 역 주파수 변환기(566)는 역 주파수 변환을 수행하여, 재구성된 예측 잔여값 또는 픽셀 값들의 블록/서브 블록을 생성한다. 예측된 프레임에 있어서, 인코더(500)는 재구성된 예측 잔여값(545)을 모션 보상 예측(535)과 결합하여 참조 화상으로서 사용될 수 있는 재구성된 프레임(505)을 형성한다. (도 5에는 도시되어 있지 않지만, 인트라 경로에서, 인코더(500)는 참조 화상으로서 사용되는 프레임을 재구성하기 위해 예측 잔여 값을 인트라 예측으로부터의 공간 예측과 결합할 수 있다.) 프레임 스토어(520)는 재구성된 현재 프레임을 후속 모션 보상 예측에서 참조 화상으로서 사용하기 위해 버퍼링한다.

[0069] 인코더(500)의 모션 보상 루프는 프레임 스토어(520)의 전 또는 후에 적응적 인-루프 디블록 필터(510)를 포함한다. 디코더(500)는 인-루프 필터링을 재구성된 프레임에 적용하여, 프레임들의 경계에서의 불연속 부분을 적응적으로 매끄럽게 한다.

[0070] 엔트로피 코더(580)는 양자화기(570)의 출력과, 모션 정보(515) 및 특정 부가 정보(가령, QP 값, 참조 인덱스, RPL 변경 정보)를 압축한다. 엔트로피 코더(580)는 인코딩된 데이터(595)를 베파(590)에 제공하는데, 이 베파는 인코딩된 데이터를 출력 비트스트림으로 다중화한다(multiplexes). 인코딩된 데이터(595)는 RPL 변경 정보를 나타내는 신택스 요소를 포함할 수 있다. 단락 VII는 이러한 신택스 요소의 예를 설명한다.

[0071] 제어기(도시 생략)는 인코더의 다양한 모듈로부터 입력을 수신한다. 제어기는 인코딩 동안, 가령, QP 값을 설정하고 레이트-왜곡 분석을 수행하는 동안에 중간 결과를 평가한다. 제어기는 다른 모듈과 함께 동작하여 인코딩 동안에 파라미터를 설정하고 변경한다. 특히, 제어기는 RPL(들)을 변경(예를 들어, 대체, 조정)을 할지와 어떻게 변경할지를 정할 때, 제어기는 어떤 참조 화상이 RPL(들)에 추가되어야 하는지를 제어할 수 있고, 어떤 화상이 RPL(들)로부터 제거되어야 하는지를 제어할 수 있으며 참조 인덱스를 사용하여 보다 효율적으로 어드레싱하기 위해 RPL(들)에서 참조 화상을 재배열할 수 있다. 제어기는 예를 들어 장면 변경 이후 모든 참조 화상을 제거하고, IDR 화상과 같은 특정한 종류의 화상의 인코딩 이후 모든 참조 화상을 제거하며, 모션 보상을 위한 참조 화상의 활용이 임계량 밑으로 떨어진 이후 주어진 참조 화상을 제거하고 및/또는 다른 기준에 따라 참조 화상을 제거함으로써 참조 화상을 RPS(따라서 RPL)로부터 제거하는 것을 결정할 수 있다. 제어기는 예를 들어 화상들 내의 화상 유형/슬라이스 유형, 화상에 대한 시간적 레이어(temporal layer) 및/또는 다른 기준에 따라 화상을 추가함으로써 참조 화상을 RPS에 추가하는 것을 결정할 수 있다. RPL의 경우, 제어기는 RPL이 비트스트림 내에 명시적으로 시그널링되는 신택스 요소에 따라 변경되지 않는 경우의 모션 보상의 결과를 평가할 수 있고, 또한 RPL이 비트스트림 내에 명시적으로 시그널링되는 신택스 요소에 따라 변경되는 경우의 모션 보상의 결과(또는 RPL을 변경하는 다수의 상이한 방식의 결과)를 평가할 수 있다. 제어기는 비트레이트 및/또는 품질과 관련하여 결과를 평가할 수 있다. 제어기는 규칙에 의해 암시적으로 구성된 RPL(RPL 변경 정보는 없음)을 선택할 수 있거나, (RPL 변경 정보로 지정된 대로) 변경된 RPL을 선택할 수 있다. RPL을 변경(예를 들어, 교체, 조정)하기 위해, 암시적으로 구성된 RPL에 비해, 제어기는 (a) 참조 인덱스를 이용한 보다 효율적인 어드레싱을 위해 참조 화상을 재배열할 수 있고, (b) 인코딩 동안의 사용 빈도에 적어도 부분적으로 기초하여 참조 화상을 제거할 수 있고 및/또는 (c) 인코딩 동안의 사용 빈도에 적어도 부분적으로 기초하여 참조 화상을 추가할 수 있다. 예를 들어, 제어기는 모션 보상을 위한 참조 화상의 활용이 임계량 밑으로 떨어진 이후 및/또는 다른 기준에 따라 RPL로부터 주어진 참조 화상을 제거하기로 결정할 수 있다. 또는, 제어기는 모션 보상을 위한 참조 화상의 활용이 임계량 위에 있는 경우 및/또는 다른 기준에 따라 RP에 주어진 참조 화상을 추가하기로 결정할 수 있다. 또는, 제어기는 각 참조 화상의 활용 빈도 및/또는 다른 기준에 기초하여 RPL에서 참조 화상을 어떻게 재배열할지를 결정할 수 있다. 제어기는 화상 단위, 슬라이스 단위 또는 소정의 다른 단위로 RPL(들)을 구성할 수 있다.

[0072] 구현 및 원하는 압축의 유형에 따라, 인코더의 모듈들이 추가되거나, 생략되거나, 여러 모듈로 분할되거나, 다른 모듈과 결합되거나, 및/또는 유사한 모듈로 대체될 수 있다. 다른 실시예에서, 상이한 모듈 및/또는 다른 모듈 구성을 갖는 인코더들이 설명되는 기술들 중 하나 이상을 수행한다. 통상적으로, 인코더의 특정 실시예는 인코더(500)의 변형 또는 보완 버전을 사용한다. 인코더(500) 내의 모듈들 사이에 도시된 관계는 이 인코더에

서의 전반적인 정보 흐름을 표시하며, 설명을 간략히 하기 위해 다른 관계는 도시하지 않았다.

## VI. 예시적 비디오 디코더

[0073] 도 6은 일부 실시예와 함께 구현될 수 있는 일반화된 디코더(600)의 블록도이다. 디코더(600)는 압축된 프레임 또는 프레임 시퀀스에 대한 인코딩된 데이터(695)를 수신하고, 참조 화상으로서 사용될 수 있는 재구성된 프레임(605)을 포함하는 출력을 생성한다. 설명을 위한 목적으로, 도 6은 인트라-프레임 디코딩을 위한 디코더(600)를 통한 "인트라 경로" 및 인터-프레임 디코딩을 위한 "인터 경로"를 도시하고 있다. 디코더(600)의 구성 요소들 중 많은 구성요소들이 인트라-프레임 디코딩 및 인터-프레임 디코딩 모두를 위해 사용된다. 이들 구성 요소에 의해 수행되는 정확한 동작은 압축 해제되는 정보의 유형에 따라 변할 수 있다.

[0074] 버퍼(690)는 압축된 프레임에 대한 인코딩된 데이터(695)를 수신하고, 수신된 인코딩된 데이터가 파서(parser)/엔트로피 디코더(680)에 대해 이용 가능하게 한다. 인코딩된 데이터(695)는 RPL 변경 정보를 표시하는 선택스 요소를 포함할 수 있다. 단락 VII는 이러한 선택스 요소의 예를 설명한다. 파서/엔트로피 디코더(680)는 통상적으로 인코더에서 수행되는 엔트로피 인코딩을 역으로 적용하여, 엔트로피 코딩된 양자화 데이터 및 엔트로피 코딩된 부가 정보(참조 인덱스, RPL 변경 정보 등을 포함함)를 엔트로피 디코딩한다.

[0075] 디코딩 동안, 디코더(600)는 적절한 때에 새로운 참조 화상이 추가되고, 적절한 때에 모션 보상에 더 이상 사용되지 않는 보다 오래된 참조 화상이 제거되며, 적절한 때에 참조 화상들이 재배열되도록 RPL(들)을 구성한다. 디코더(600)는 RPL(들)에 대한 이용 가능한 정보(예를 들어, RPS에서 이용 가능한 참조 화상), 규칙에 따른 변경에 기초하여 및/또는 인코딩된 데이터(695)의 일부로서 시그널링된 변경에 따라 RPL(들)을 구성할 수 있다. 몇몇 구현에서, 예를 들어, 현재의 프레임을 디코딩하는 경우, 디코더(600)는 프레임 스토어(620)의 참조 프레임을 포함하는 RPS를 결정한다. 디코더(600)는 전형적으로 프레임의 제1 슬라이스에 대한 RPS를 결정한다. 슬라이스 단위로, 디코더(600)는 현재 프레임의 주어진 슬라이스의 디코딩을 위한 하나 이상의 RPL을 생성한다. RPL을 생성하기 위해, 몇몇 경우 (인코딩된 데이터(695)에 표시된 바와 같이), 디코더(600)는 RPS로부터 이용 가능한 참조 프레임의 선택에 대한 규칙을 적용하는데, 이 경우 RPL 변경 정보는 인코딩된 데이터(695)로부터 파싱되지 않는다. 다른 경우로, RPL을 생성하기 위해, 디코더(600)는 RPS로부터 이용 가능한 특정 참조 프레임을 선택하는데, 이 경우 선택된 참조 프레임은 인코딩된 데이터(695)로부터 파싱되는 RPL 변경 정보에 표시된다. RPL 변경 정보는 RPS 내의 참조 화상들의 리스트으로서 대체 RPL을 지정할 수 있다. 이와 달리, RPL 변경 정보는 규칙에 의해 암시적으로 구성된 RPL에서 하나 이상의 참조 프레임의 제거, 하나 이상의 참조 프레임의 추가 및/또는 참조 프레임들의 재배열을 보다 정밀하게 지정할 수 있다.

[0076] 모션 보상기(630)는 모션 정보(615)를 하나 이상의 참조 프레임(625)에 적용하여, 재구성되는 프레임(605)의 서브 블록 및/또는 블록의 모션-보상 예측(635)을 형성한다. 프레임 스토어(620)는 하나 이상의 이전에 재구성된 프레임을 참조 프레임으로서 사용하기 위해 저장한다.

[0077] 인트라 경로는, 이전에 재구성된 인접 픽셀 값들로부터 현재 블록 또는 서브 블록의 픽셀 값을 공간적으로 예측하는 인트라 예측 모듈(도시 생략)을 포함할 수 있다. 인터 경로에서, 디코더(600)는 예측 잔여값을 재구성한다. 역 양자화기(inverse quantizer)(670)는 엔트로피-디코딩된 데이터를 역 양자화한다. 역 주파수 변환기(660)는 재구성된 주파수 영역 데이터를 공간 영역 정보로 변환한다. 예를 들어, 역 주파수 변환기(660)는 역 블록 변환을 주파수 변환 계수에 적용하여 픽셀 값 데이터 또는 예측 잔여 데이터를 생성한다. 역 주파수 변환은 역 이산 코사인 변환, 그 정수 근사치 계산 또는 임의의 역 주파수 변환일 수 있다.

[0078] 예측된 프레임에 있어서, 디코더(600)는 참조 화상으로서 사용될 수 있는 재구성된 프레임(605)을 형성하기 위해 재구성된 예측 잔여값(645)을 모션 보상 예측(635)과 결합한다. (도 6에는 도시되어 있지 않지만, 인트라 경로에서, 디코더(600)는 예측 잔여값을 인트라 예측으로부터의 공간 예측과 결합하여 참조 화상으로서 사용될 수 있는 프레임을 재구성할 수 있다.) 디코더(600)의 모션 보상 루프는 프레임 스토어(620)의 전 또는 후에 적응적 인-루프 디블록 필터(610)을 포함한다. 디코더(600)는 인-루프 필터링을 재구성된 프레임에 적용하여, 프레임들의 경계에서의 불연속 부분을 적응적으로 매끄럽게 한다.

[0079] 도 6에서, 디코더(600)는 또한 포스트-프로세싱 디블록 필터(608)를 포함한다. 포스트-프로세싱 디블록 필터(608)는 재구성된 프레임에서의 불연속 부분을 선택적으로 매끄럽게 한다. 다른 필터링 (가령, 디링(de-ring) 필터링) 또한 포스트-프로세싱 필터링의 일부로서 적용될 수 있다.

[0080] 구현 및 원하는 압축 해제의 유형에 따라, 디코더의 모듈들이 추가되거나, 생략되거나, 여러 모듈로 분할되거나, 다른 모듈과 결합되거나, 및/또는 유사한 모듈로 대체될 수 있다. 다른 실시예에서, 상이한 모듈

및/또는 다른 모듈 구성을 갖는 디코더들이 설명되는 기술들 중 하나 이상을 수행한다. 통상적으로, 디코더의 특정 실시예는 디코더(600)의 변형 또는 보완 버전을 사용한다. 디코더(600) 내의 모듈들 사이에 도시된 관계는 이 디코더에서의 전반적인 정보 흐름을 표시하며, 설명을 간략히 하기 위해 다른 관계는 도시하지 않았다.

## [0082] VII. 참조 화상 리스트 변경 정보의 시그널링

[0083] 이 단락에서는, RPL 변경 정보를 시그널링하는 분야에서의 다양한 혁신을 제공한다. 일부 상황에서, 이들 혁신은 RPL 변경 정보에 대한 선택스 요소의 보다 효율적인 시그널링을 달성한다.

### [0084] A. 참조 화상 및 RPL

[0085] 일반적으로 참조 화상은 전형적으로 참조 화상을 디코딩 순서로 따라가는 다른 화상들의 디코딩 프로세스에서 인터-화상 예측에 사용될 수 있는 샘플들을 포함하는 화상이다. 주어진 시간에 모션 보상 예측용으로 다수의 참조 화상이 이용가능할 수 있다.

[0086] 일반적으로, 참조 화상 리스트("RPL")는 모션 보상 예측에 사용되는 참조 화상들의 리스트이다. RPL 내의 참조 화상들은 참조 인덱스로 어드레싱된다. 참조 인덱스는 RPL 내에서 참조 화상을 식별한다. 인코딩 및 디코딩 동안, RPL이 구성되는 경우, RPL 내의 참조 화상은 때때로 변경되어 RPL 내에 새롭게 디코딩된 화상을 추가하고, 참조 화상으로서 더 이상 사용되지 않는 보다 오래된 화상을 제거하며 및/또는 참조 화상들을 재배열하여 보다 일반적으로 사용되는 참조 인덱스의 시그널링을 보다 효율적이게 한다. 인코더 및 디코더는 동일한 규칙을 따라 그들의 RPL(들)을 구성, 변경 등을 행할 수 있다. 이러한 규칙에 대해(또는 이러한 규칙 대신), 인코더는 디코더가 인코더에 의해 사용되는 RPL(들)에 일치하도록 그의 RPL(들)을 어떻게 구성, 변경 등을 행해야 하는지를 나타내는 정보를 디코더에 알려줄 수 있다. 전형적으로, 인코딩 및 디코딩 동안 RPL은 RPL에 대한 이용가능한 정보(예를 들어, RPS 내에서 이용가능한 화상), 규칙에 따른 변경 및/또는 비트스트림 내에 시그널링된 변경에 기초하여 구성된다.

[0087] 몇몇 구현에서, 현재의 화상에 대해, 인코더 또는 디코더는 디코딩된 화상 버퍼("DPB")와 같은 디코딩된 프레임 저장 영역 내의 참조 화상을 포함하는 참조 화상 세트("RPS")를 결정한다. RPS는 현재 및 미래의 코딩된 화상의 디코딩 프로세스에서 사용되는 참조 화상의 설명이다. RPS에 포함된 참조 화상은 스트림 내에 명시적으로 나열된다.

[0088] 인코더 또는 디코더는 화상마다 한번 RPS를 결정한다. 예를 들어, 디코더는 슬라이스 헤더에서 시그널링된 선택스 요소를 사용하여, 화상의 슬라이스에 대한 슬라이스 헤더를 디코딩한 후 RPS를 결정한다. 참조 화상은 화상 순서 카운트("POC") 값, 이들의 일부 및/또는 비트스트림 내에 시그널링된 다른 정보를 통해 식별된다. 인코더 또는 디코더는 현재 화상의 화상간 예측에 사용될 수 있는 (또한 디코딩 순서에 있어서 현재의 화상을 뒤따르는 하나 이상의 화상의 화상간 예측에 사용될 수 있는) 단기간 참조 화상 및 장기간 참조 화상의 그룹을 결정한다. (인코더 또는 디코더는 또한 디코딩 순서에 있어서 현재의 화상을 뒤따르는 하나 이상의 화상의 화상간 예측에 사용될 수 있으나, 현재의 화상에 대해서는 사용되지 않는 참조 화상의 그룹을 결정한다.) 총칭하여, 참조 화상들의 그룹은 현재 화상에 대한 RPS이다.

[0089] 현재 화상의 주어진 슬라이스에 대해, 인코더 또는 디코더는 하나 이상의 RPL을 생성한다. 인코더 또는 디코더는 현재 화상의 화상간 예측에 사용될 수 있는 단기 참조 화상 및 장기 참조 화상의 그룹들을 결합함으로써 RPL의 임시 버전(예를 들어, RPL 0 또는 RPL 1)을 생성한다. "암시적" 접근방식의 규칙에 따라 RPL을 구성하기 위해, 인코더 또는 디코더는 RPL의 임시 버전 내의 참조 화상을 이용하거나, 또는 RPL의 임시 버전 내의 참조 화상의 일부만(예를 들어, RPL의 임시 버전 내의 처음 x개의 화상들)을 사용할 수 있다. "암시적" 접근방식에서, RPL 변경 정보는 비트스트림 내에 시그널링되지 않을 것이며, 비트스트림으로부터 파싱되지 않는다. "명시적" 접근방식에서, RPL을 구성하기 위해, 인코더 또는 디코더는 비트스트림 내에 시그널링된/비트스트림으로부터 파싱된 RPL 변경 정보를 사용하여 RPL의 임시 버전으로부터 특정 참조 화상을 선택한다. "암시적" 접근방식의 규칙에 의해 구성될 수 있는 RPL과 비교할 때, RPL 변경 정보는 RPL 내에서의 하나 이상의 참조 화상의 제거, 하나 이상의 참조 화상의 추가 및/또는 참조 화상들의 재배열을 지정할 수 있다.

[0090] 이와 달리, 인코더 또는 디코더는 참조 화상들로부터 RPL을 생성하는데 다른 접근방식을 사용한다.

### [0091] B. RPL 변경 플래그의 조건부적 시그널링

[0092] 본 명세서에서 기술된 혁신들의 일 측면에 따르면, 인코더는 비트스트림 내에 명시적으로 시그널링된 선택스 요소에 따라 RPL이 변경되는지를 나타내는 플래그를 조건부로 시그널링한다. 대응하는 디코더는 이러한 플래그를

조건부로 파싱한다.

[0093] 몇몇 예시적인 구현에 있어서, 플래그는 ref\_pic\_list\_modification\_flag\_10 또는 ref\_pic\_list\_modification\_flag\_11이다(일반적으로, 플래그는 ref\_pic\_list\_modification\_flag\_1X이되, X는 0 또는 1일 수 있다). 플래그 ref\_pic\_list\_modification\_flag\_1X의 값이 1인 경우, RPL X는 list\_entry\_1X[i] 값들의 리스트로서 명시적으로 지정된다(X는 0 또는 1임). 플래그 ref\_pic\_list\_modification\_flag\_1X의 값이 0인 경우, RPL X는 암시적으로 결정된다. ref\_pic\_list\_modification\_flag\_1X가 존재하지 않는 경우, 그것은 0인 것으로 추론된다.

[0094] 도 7a는 예시적인 구현에서 ref\_pic\_lists\_modification() 선택스 구조에 대한 예시적인 선택스(700)를 나타낸다. 이 구조는 슬라이스 헤더의 일부로서 시그널링될 수 있다. 예시적인 선택스(700)에서, NumPocTotalCurr이 1보다 큰 경우, ref\_pic\_list\_modification\_flag\_1X만이 전송된다. NumPocTotalCurr은 현재의 인코딩 또는 디코딩에 적용 가능한 참조 화상의 총 수를 나타내는 변수이다. 인코딩 또는 디코딩의 예시적인 구현에서, 현재 화상의 슬라이스에 대한 변수 NumPocTotalCurr이 도출되는 경우, 이 변수는 현재 화상의 인코딩 또는 디코딩을 위한 참조 화상으로서 사용되는 단기 참조 화상 및 장기 참조 화상의 카운트를 나타낸다.

[0095] 도 7a에 도시되어 있는 바와 같이, ref\_pic\_list\_modification\_flag\_1X의 조건부적 시그널링은 변수 NumPocTotalCurr의 값에 의존한다. NumPocTotalCurr이 1이하인 경우, RPL의 변경 가능성은 없고, 따라서 플래그 ref\_pic\_list\_modification\_flag\_1X를 전송할 필요가 없다. 이러한 조건부적 시그널링은 조건이 이행되면 각 슬라이스마다 하나 또는 두 개의 플래그를 절약할 수 있다. 도 7a에서의 변경은 플래그 ref\_pic\_list\_modification\_flag\_1X가 시그널링되는지에 대한 조건 "if(NumPocTotalCurr>1)"을 포함한다. 이 조건은 리스트 0(P 슬라이스 또는 B 슬라이스에 대해) 및/또는 리스트 1(B 슬라이스에 대해)에 대해 체크될 수 있다.

[0096] 이와 달리, 하나 이상의 RPL 변경 플래그를 포함하는 RPL 변경 구조(예를 들어, ref\_pic\_lists\_modification() 구조)의 시그널링 및 파싱은 슬라이스 헤더 프로세싱의 일부로서 조건을 평가함으로써 또는 그와 다른 방식으로 제어될 수 있다. 도 7b는 이러한 조건에 기초하여 플래그 ref\_pic\_list\_modification\_flag\_10 및 ref\_pic\_list\_modification\_flag\_11의 조건부로 시그널링 및 파싱에 대한 접근방식을 나타낸다. 구체적으로, 도 7b는 도 7c의 선택스(760)에 도시되어 있는 ref\_pic\_lists\_modification() 선택스 구조를 포함할 수 있는 슬라이스 헤더 선택스 구조에 대한 예시적인 선택스(750)를 보여준다. 슬라이스 헤더의 예시적인 선택스(750)에 대해, 플래그 lists\_modification\_present\_flag는 슬라이스에 적용되는 화상 파라미터 세트에 시그널링된다. lists\_modification\_present\_flag가 0인 경우, 구조 ref\_pic\_lists\_modification()은 슬라이스 헤더에 존재하지 않는다. lists\_modification\_present\_flag가 1인 경우, 구조 ref\_pic\_lists\_modification()은 변수 NumPocTotalCurr의 값에 따라 슬라이스 헤더에 존재할 수 있다. 변수 NumPocTotalCurr이 1보다 큰 경우, ref\_pic\_lists\_modification() 구조는 도 7c의 선택스(760)에 도시되어 있는 바와 같이 시그널링된다. 그와 다른 경우(변수 NumPocTotalCurr이 1보다 크지 않은 경우), ref\_pic\_lists\_modification() 구조는 시그널링되지 않고, 리스트 엔트리들의 값은 추론된다.

[0097] 도 7a-7c, 8 및 9에서, "u(n)" 항은 n개의 비트를 사용하는 부호 없는 정수를 나타낸다. ("u(v)"에서와 같이) n이 "v"인 경우, 비트의 수는 다른 선택스 요소의 값에 따라 달라진다. u(n)에 대한 파싱 프로세스는 MSB(most significant bit)가 맨 처음 기록되는 부호 없는 정수의 이진 표현으로서 n개의 비트를 판독하는 함수의 리턴 값에 의해 지정될 수 있다.

### C. 리스트 엔트리들에 대한 선택스 요소들의 시그널링

[0099] 본 명세서에 기술된 혁신들의 또 다른 측면에 따르면, 인코더는 RPL을 어떻게 변경할 것인지를 나타내는 리스트 엔트리들에 대한 선택스 요소들을 조건부로 시그널링한다. 대응하는 디코더는 이러한 선택스 요소들을 조건부로 파싱한다.

[0100] 몇몇 예시적인 구현에서, 선택스 요소들은 RPL 0에 대한 list\_entry\_10[i] 선택스 요소를 또는 RPL 1에 대한 list\_entry\_11[i] 선택스 요소들에 관한 것이다(일반적으로, 선택스 요소는 list\_entry\_1X이며, X는 0 또는 1일 수 있다). 도 8은 슬라이스 헤더의 일부로서 시그널링될 수 있는 ref\_pic\_lists\_modification() 선택스 구조에 대한 예시적인 선택스(800)를 나타낸다. 예시적인 선택스(800)에서, 선택스 요소 list\_entry\_1X[0]는 비트스트림 내에 조건부로 시그널링된다. 특히, NumPocTotalCurr이 2이고 num\_ref\_idx\_1X\_active\_minus1이 0인 경우, 선택스 요소 list\_entry\_1X[0]는 비트스트림 내에 시그널링되지 않는다. 변수

`num_ref_idx_1X_active_minus1`은 슬라이스를 디코딩하는데 사용될 수 있는 RPL X에 대한 최대 참조 인덱스를 나타낸다. `num_ref_idx_1X-active_minus1` 변수는 디폴트 값(예를 들어, 적용 가능한 화상 파라미터 세트에서 지정된 0, ..., 15의 값)을 가질 수 있거나, `num_ref_idx_1X_active_minus1`은 현재 슬라이스에 대한 슬라이스 헤더에 시그널링된 값을 가질 수 있다.

[0101] 도 8에 도시되어 있는 바와 같이, RPL 변경 정보가 비트스트림 내에 시그널링됨을 `ref_pic_list_modification_flag_1X`가 나타내더라도, `list_entry_1X[0]`의 시그널링은 `NumPocTotalCurr` 및 `num_ref_idx_1X_active_minus1`에 의존한다. `NumPocTotalCurr`이 2이고 `num_ref_idx_1X_active_minus1`이 1인 경우, `list_entry_1X[0]`의 값은 `ref_pic_list_modification_flag_1X`에 기초하여 추론될 수 있는데, 그 이유는 선택 가능한 것이 2개만(0의 디폴트 값 또는 1의 비-디폴트 값)이 있기 때문이다.

[0102] 따라서, 도 8은 리스트 엔트리들에 대한 선택스 값들이 시그널링될지 여부에 대한 조건을 포함한다. RPL 0의 경우, 조건은 "`if(ref_pic_list_modification_flag_10 && !(NumPocTotalCurr == 2 && num_ref_idx_10_active_minus1 == 0))`"이다. RPL 1의 경우, 조건은 "`if(ref_pic_list_modification_flag_11 && !(NumPocTotalCurr == 2 && num_ref_idx_11_active_minus1 == 0))`"이다.

[0103] 도 8의 예에서, `list_entry_10[i]`는 RPL 0의 현재 위치에 배치될 `RefPicListTemp0`(RPL의 임시 버전)에서의 참조 화상의 인덱스를 지정한다. `list_entry_10[i]` 선택스 요소의 길이는 `Ceil(Log2(NumPocTotalCurr))` 비트이다. `list_entry_10[i]`의 값은 0 내지 `NumPocTotalCurr-1`의 범위에 속한다. `NumPocTotalCurr`이 2이고 `num_ref_idx_10_active_minus1`이 0인 경우, 선택스 요소 `list_entry_10[0]`은 `ref_pic_list_modification_flag_10`과 동일한 것으로 추론된다. 그렇지 않으면, 선택스 요소 `list_entry_10[i]`가 존재하지 않는 경우, 그것은 0인 것으로 추론된다.

[0104] 도 8의 예에서, `list_entry_11[i]`는 RPL 1의 현재 위치에 배치될 `RefPicListTemp1`(RPL의 임시 버전)에서의 참조 화상의 인덱스를 지정한다. `list_entry_11[i]` 선택스 요소의 길이는 `Ceil(Log2(NumPocTotalCurr))` 비트이다. `list_entry_11[i]`의 값은 0 내지 `NumPocTotalCurr-1`의 범위에 속한다. `NumPocTotalCurr`이 2이고 `num_ref_idx_11_active_minus1`이 0인 경우, 선택스 요소 `list_entry_11[0]`은 `ref_pic_list_modification_flag_11`과 동일한 것으로 추론된다. 그렇지 않으면, 선택스 요소 `list_entry_11[i]`가 존재하지 않는 경우, 그것은 0인 것으로 추론된다.

[0105] 도 9는 슬라이스 헤더의 일부로서 시그널링될 수 있는 `ref_pic_lists_modification()` 선택 구조에 대한 또 다른 예시적인 선택스(900)를 나타낸다. 예시적인 선택스(900)에서, 선택스 요소 `list_entry_1X[0]`는 비트스트림 내에 조건부로 시그널링된다. 그러나, 도 8의 예시적인 선택스와 비교해 볼 때, 체크되는 조건은 다르다. 또한, `list_entry_1X[]`에 대한 선택스 요소들의 시그널링은 가중치 부여된 예측이 사용되는지 여부에 따라 조정될 수 있다.

[0106] 도 9에 따르면, 가중치 부여된 예측이 인에이블링되느냐 또는 디스에이블링되느냐는 리스트 엔트리들에 대한 선택스 요소들이 비트스트림 내에 시그널링되는 방식에 영향을 준다. 0인 `weighted_pred_flag`를 갖는 P 슬라이스 또는 0인 `weighted_bipred_flag`를 갖는 B 슬라이스에 대해, 가중치 부여된 예측은 디스에이블링된다. 도 9의 예시적인 선택스(900)에 따르면, 가중치 부여된 예측이 디스에이블링되는 경우, `NumPocTotalCurr`이 2이고 `num_ref_idx_1X_active_minus1`이 1이라면 `list_entry_1X[0]` 및 `list_entry_1X[1]`은 전송되지 않는다. 이러한 경우, `list_entry_1X[0]` 및 `list_entry_1X[1]`은 각각 1 및 0인 것으로 추론되는데, 그 이유는 RPL 변경은 오직 다른 가능성(즉, `list_entry_1X[0]` 및 `list_entry_1X[1]`이 각각 0 및 1인 경우)에 대해서는 필요하지 않을 수 있기 때문이다.

[0107] 따라서, 도 9는 리스트 엔트리들에 대한 선택스 요소들이 시그널링될지 여부에 대한 조건을 포함한다. RPL 0의 경우, 조건은 "`if(ref_pic_list_modification_flag_10 && !(NumPocTotalCurr == 2 && num_ref_idx_10_active_minus1 == 0) && !(NumPocTotalCurr == 2 && num_ref_idx_10_active_minus1 == 1) && ((weighted_pred_flag != 1 && slice_type == P) || (weighted_bipred_flag != 1 && slice_type == B)))`"이다. RPL 1의 경우, 조건은 "`if(ref_pic_list_modification_flag_11 && !(NumPocTotalCurr == 2 && num_ref_idx_11_active_minus1 == 0) && !(NumPocTotalCurr == 2 && num_ref_idx_11_active_minus1 == 1) && weighted_bipred_flag != 1)`"이다.

[0108] 또한, `NumPocTotalCurr`이 2가 아니거나 `num_ref_idx_1X_active_minus1`이 1이 아니더라도, 가중치 부여된 예측이 디스에이블링된다면(P 슬라이스의 경우, `weighted_pred_flag`는 0이고, B 슬라이스의 경우,

weighted\_bipred\_flag는 0임), list\_entry\_1X[i] 선택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr-i)) 비트로 제한된다. 이 경우, 리스트 내에 각 참조 화상을 한번만 배치하는 것이 유용할 뿐이고, 따라서 유용한 가능성의 수는 인덱스 i가 증가함에 따라 감소한다.

[0109] 도 9의 예에서, list\_entry\_10[i]는 RPL 0의 현재 위치에 배치될 RefPicListTemp0(RPL의 임시 버전)에서의 참조 화상의 인덱스를 지정한다. 가중치 부여된 예측이 디스에이블링되는 경우(P 슬라이스의 경우, weighted\_pred\_flag는 0이고, B 슬라이스의 경우, weighted\_bipred\_flag는 0임), list\_entry\_10[i] 선택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr-i)) 비트이다. 그렇지 않으면, list\_entry\_10[i] 선택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr)) 비트이다. NumPocTotalCurr이 2이고 num\_ref\_idx\_10\_active\_minus1이 0인 경우, (도 8의 예에서와 같이) 선택스 요소 list\_entry\_10[0]은 ref\_pic\_list\_modification\_flag\_10와 등가인 것으로 추론된다. 그렇지 않으면, NumPocTotalCurr이 2이고, num\_ref\_idx\_10\_active\_minus1이 1이며 가중치 부여된 예측이 디스에이블링되는 경우(weighted\_pred\_flag가 0이고 현재의 슬라이스가 P 슬라이스이거나, weighted\_bipred\_flag가 0이고 현재의 슬라이스가 B 슬라이스인 경우), 선택스 요소 list\_entry\_10[0] 및 list\_entry\_10[1]은 각각 1 및 0인 것으로 추론된다. 그와 달리, 선택스 요소 list\_entry\_10[i]가 존재하지 않는 경우, 그것은 0인 것으로 추론된다.

[0110] 가중치 부여된 예측이 디스에이블링되는 경우(weighted\_pred\_flag가 0이고 현재의 슬라이스가 P 슬라이스이거나, weighted\_bipred\_flag가 0이고 현재의 슬라이스가 B 슬라이스인 경우), list\_entry\_10[i]의 값은 0 내지 NumPocTotalCurr-(i+1)의 범위에 속하고, 리스트 RefPicListTemp0는 엔트리 값이 파싱된 이후 리스트 RefPicListTemp0로부터 각 엔트리 list\_entry\_10[i]를 제거함으로써 단축된다. 그렇지 않으면, list\_entry\_10[i]의 값은 0 내지 NumPocTotalCurr-1)의 범위에 속한다.

[0111] 도 9의 예에서, list\_entry\_11[i]는 RPL 1의 현재 위치에 배치될 RefPicListTemp1(RPL의 임시 버전)에서의 참조 화상의 인덱스를 지정한다. 가중치 부여된 예측이 디스에이블링되는 경우(오직 B 슬라이스가 리스트 1을 사용하기 때문에 weighted\_bipred\_flag가 0인 경우), list\_entry\_11[i] 선택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr-i)) 비트이다. 그렇지 않으면, list\_entry\_11[i] 선택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr)) 비트이다. NumPocTotalCurr이 2이고 num\_ref\_idx\_11\_active\_minus1이 0인 경우, (도 8의 예에서와 같이) 선택스 요소 list\_entry\_11[0]은 ref\_pic\_list\_modification\_flag\_10와 등가인 것으로 추론된다. 그와 달리, NumPocTotalCurr이 2이고, num\_ref\_idx\_11\_active\_minus1이 1이며 가중치 부여된 예측이 디스에이블링되는 경우(weighted\_bipred\_flag가 0이고 현재의 슬라이스가 B 슬라이스인 경우), 선택스 요소 list\_entry\_11[0] 및 list\_entry\_11[1]은 각각 1 및 0인 것으로 추론된다. 그와 달리, 선택스 요소 list\_entry\_11[i]가 존재하지 않는 경우, 그것은 0인 것으로 추론된다.

[0112] 가중치 부여된 예측이 디스에이블링되는 경우(weighted\_bipred\_flag가 0이고 현재의 슬라이스가 B 슬라이스인 경우), list\_entry\_11[i]의 값은 0 내지 NumPocTotalCurr-(i+1)의 범위에 속하고, 리스트 RefPicListTemp1은 엔트리 값이 파싱된 이후 리스트 RefPicListTemp1부터 각 엔트리 list\_entry\_11[i]를 제거함으로써 단축된다. 그렇지 않으면, list\_entry\_10[i]의 값은 0 내지 NumPocTotalCurr-1)의 범위에 속한다.

#### D. RPL 변경 플래그의 조건부적 시그널링 및 파싱을 위한 일반화된 기법

[0114] 도 10은 RPL 변경 플래그의 조건부적 시그널링을 위한 일반화된 기법(1000)을 나타낸다. 예를 들어, 도 3을 참조하여 설명한 비디오 인코더를 구현하는 컴퓨팅 장치가 이 기법(1000)을 수행할 수 있다.

[0115] 장치는 조건을 평가한다(1010). 예를 들어, 조건은 전체 참조 화상의 개수를 나타내는 변수에 적어도 부분적으로 의존한다. 몇몇 예시적인 구현에서, 변수는 NumPocTotalCurr이고, 인코더는 변수가 1보다 큰지를 확인한다. 이와 달리, 인코더는 다른 및/또는 추가의 조건을 평가한다. 평가되는(1010) 조건은 단일 인자(예를 들어, 전체 참조 화상의 개수를 나타내는 변수의 값)를 포함할 수 있거나, 평가되는(1010) 조건은 다수의 인자(예를 들어, 전체 참조 화상의 개수를 나타내는 변수의 값 및 하나 이상의 다른 인자)를 포함할 수 있다. 조건은 RPL 변경 구조에 대한 프로세싱의 일부로서 평가될 수 있다(1010). 또는, 조건은 슬라이스 헤더의 프로세싱의 일부로서 평가될 수 있다(1010).

[0116] 평가의 결과에 따라, 장치는 비트스트림 내에 명시적으로 시그널링되는 선택스 요소들에 따라 RPL이 변경(예를 들어, 대체, 조정)되는지를 나타내는 플래그를 비트스트림에 조건부로 시그널링한다(1020). 예를 들어, 플래그는 ref\_pic\_list\_modification\_flag\_10 및 ref\_pic\_list\_modification\_flag\_11 중 하나이고, 슬라이스 헤더의 RPL 변경 구조의 일부로서 조건부로 시그널링될 수 있다. 또는, 조건이 평가된(1010) 이후, 평가의 결과에 따

라, (RPL이 비트스트림 내에 명시적으로 시그널링된 신택스 요소에 따라 변경되는지 여부를 나타내는 하나 이상의 플래그를 포함하는) RPL 변경 구조는 비트스트림 내에 조건부로 시그널링된다.

[0117] 장치는 RPL 변경 구조가 시그널링된 경우 슬라이스 단위로, 또는 몇몇 다른 단위로 기법(1000)을 반복할 수 있다.

[0118] 도 11은 RPL 변경 플래그의 조건부적 파싱을 위한 일반화된 기법(1100)을 나타낸다. 예를 들어, 도 4을 참조하여 설명한 비디오 디코더를 구현하는 컴퓨팅 장치가 이 기법(1100)을 수행할 수 있다.

[0119] 디코더는 조건을 평가한다(1110). 예를 들어, 조건은 전체 참조 화상의 개수를 나타내는 변수에 적어도 부분적으로 의존한다. 몇몇 예시적인 구현에서, 변수는 NumPocTotalCurr이고, 디코더는 변수가 1보다 큰지를 확인한다. 이와 달리, 디코더는 다른 및/또는 추가의 조건을 평가한다. 평가되는(1110) 조건은 단일 인자(예를 들어, 전체 참조 화상의 개수를 나타내는 변수의 값)를 포함할 수 있거나, 평가되는(1110) 조건은 다수의 인자(예를 들어, 전체 참조 화상의 개수를 나타내는 변수의 값 및 하나 이상의 다른 인자)를 포함할 수 있다. 조건은 RPL 변경 구조에 대한 프로세싱의 일부로서 평가될 수 있다(1110). 또는, 조건은 슬라이스 헤더의 프로세싱의 일부로서 평가될 수 있다(1110).

[0120] 평가의 결과에 따라, 장치는 비트스트림 내에 명시적으로 시그널링되는 신택스 요소들에 따라 RPL이 변경(예를 들어, 대체, 조정)되는지를 나타내는 플래그를 비트스트림으로부터 조건부로 파싱한다(1120). 예를 들어, 플래그는 ref\_pic\_list\_modification\_flag\_10 및 ref\_pic\_list\_modification\_flag\_11 중 하나이고, 슬라이스 헤더의 RPL 변경 구조의 일부로서 조건부로 시그널링될 수 있다. 또는, 조건이 평가된(1110) 이후, 평가의 결과에 따라, (RPL이 비트스트림 내에 명시적으로 시그널링된 신택스 요소에 따라 변경되는지 여부를 나타내는 하나 이상의 플래그를 포함하는) RPL 변경 구조는 비트스트림으로부터 조건부로 파싱된다.

[0121] 장치는 RPL 변경 구조가 시그널링된 경우 슬라이스 단위로, 또는 몇몇 다른 단위로 기법(1100)을 반복할 수 있다.

#### E. 리스트 엔트리의 조건부적 시그널링 및 파싱을 위한 일반화된 기법

[0123] 도 12는 RPL 변경을 위한 리스트 엔트리의 조건부적 시그널링을 위한 일반화된 기법(1200)을 나타낸다. 예를 들어, 도 3을 참조하여 설명한 비디오 인코더를 구현하는 컴퓨팅 장치가 이 기법(1200)을 수행할 수 있다.

[0124] 장치는 조건을 평가한다(1210). 예를 들어, 조건은 전체 참조 화상의 개수를 나타내는 변수(예를 들어, 몇몇 예시적인 구현에서, NumPocTotalCurr)에 적어도 부분적으로 의존한다. 또는, 조건은 RPL에 대한 활성 참조 화상의 수에 적어도 부분적으로 의존한다. 또는, 조건은 가중치 부여된 예측이 디스에이블링되는지 여부에 적어도 부분적으로 기초한다. 현재의 슬라이스가 P 슬라이스인지 또는 B 슬라이스인지에 의존하여 및/또는 어떤 RPL이 시그널링/파싱되는지에 의존하여 가중치 부여된 예측이 디스에이블링되는지 여부를 확인하는데 상이한 로직이 사용될 수 있다. 예를 들어, (P 슬라이스 또는 B 슬라이스에 의해 사용될 수 있는) 제1 RPL에 대한 조건을 확인하는 로직은 (B 슬라이스에 의해서만 사용될 수 있는) 제2 RPL에 대한 조건을 확인하는 로직과는 다르다. 이와 달리, 인코더는 다른 및/또는 추가의 조건을 평가한다.

[0125] 평가의 결과에 따라, 장치는 RPL을 어떻게 변경(예를 들어, 대체, 조정)할지를 나타내는 리스트 엔트리들에 대한 하나 이상의 신택스 요소를 비트스트림 내에 조건부로 시그널링한다(1220). 예를 들어, 리스트 엔트리들에 대한 신택스 요소(들)는 슬라이스 헤더의 RPL 변경 구조의 일부로서 조건부로 시그널링된다.

[0126] 몇몇 예시적인 구현에서, (a) 전체 참조 화상의 수가 2이고 (b) RPL에 대한 활성 참조 화상의 수가 1인 경우, 리스트 엔트리들에 대한 신택스 요소(들)는 비스트림으로부터 빠져있고, 리스트 엔트리들 중 하나의 값이 추론된다. 다른 예시적인 구현에서, 이 조건에 더해, (c) 전체 참조 화상의 수가 2이고 (d) RPL에 대한 활성 참조 화상의 수가 2이며 (e) 가중치 부여된 예측이 디스에이블링되는 경우, 리스트 엔트리들에 대한 하나 이상의 신택스 요소(들)는 비스트림으로부터 빠져있고, 리스트 엔트리들 중 두 개의 값이 추론된다.

[0127] 장치는 RPL 변경 구조가 시그널링된 경우 슬라이스 단위로, 또는 몇몇 다른 단위로 기법(1200)을 반복할 수 있다.

[0128] 도 13은 RPL 변경을 위한 리스트 엔트리들의 조건부적 파싱을 위한 일반화된 기법(1300)을 나타낸다. 예를 들어, 도 4를 참조하여 설명한 비디오 디코더를 구현하는 컴퓨팅 장치가 이 기법(1300)을 수행할 수 있다.

[0129] 디코더는 조건을 평가한다(1310). 예를 들어, 조건은 전체 참조 화상의 개수를 나타내는 변수(예를 들어, 몇몇

예시적인 구현에서, NumPocTotalCurr)에 적어도 부분적으로 의존한다. 또는, 조건은 RPL에 대한 활성 참조 화상의 수에 적어도 부분적으로 의존한다. 또는, 조건은 가중치 부여된 예측이 디스에이블링되는지 여부에 적어도 부분적으로 기초한다. 현재의 슬라이스가 P 슬라이스인지 또는 B 슬라이스인지에 의존하여 및/또는 어떤 RPL이 시그널링/파싱되는지에 의존하여 가중치 부여된 예측이 디스에이블링되는지 여부를 확인하는데 상이한 로직이 사용될 수 있다. 이와 달리, 디코더는 다른 및/또는 추가의 조건을 평가한다.

[0130] 평가의 결과에 따라, 장치는 RPL을 어떻게 변경(예를 들어, 대체, 조정)할지를 나타내는 리스트 엔트리들에 대한 하나 이상의 신택스 요소를 비트스트림으로부터 조건부로 파싱한다(1320). 예를 들어, 리스트 엔트리들에 대한 신택스 요소(들)는 슬라이스 헤더의 RPL 변경 구조로부터 조건부로 파싱된다.

[0131] 몇몇 예시적인 구현에서, (a) 전체 참조 화상의 수가 2이고 (b) RPL에 대한 활성 참조 화상의 수가 1인 경우, 리스트 엔트리들에 대한 신택스 요소(들)는 비스트림으로부터 빠져있고, 리스트 엔트리들 중 하나의 값이 추론된다. 다른 예시적인 구현에서, 이 조건에 더해, (c) 전체 참조 화상의 수가 2이고 (d) RPL에 대한 활성 참조 화상의 수가 2이며 (e) 가중치 부여된 예측이 디스에이블링되는 경우, 리스트 엔트리들에 대한 하나 이상의 신택스 요소(들)는 비스트림으로부터 빠져있고, 리스트 엔트리들 중 두 개의 값이 추론된다.

[0132] 장치는 RPL 변경 구조가 시그널링된 경우 슬라이스 단위로, 또는 몇몇 다른 단위로 기법(1300)을 반복할 수 있다.

#### F. 리스트 엔트리들의 시그널링 및 파싱을 조정하기 위한 일반화된 기법

[0134] 도 14는 RPL 변경을 위한 리스트 엔트리들의 시그널링을 조정하기 위한 일반화된 기법(1400)을 나타낸다. 예를 들어, 도 3을 참조하여 설명한 비디오 인코더를 구현하는 컴퓨팅 장치가 이 기법(1400)을 수행할 수 있다.

[0135] 장치는 조건을 평가한다(1410). 예를 들어, 조건은 가중치 부여된 예측이 디스에이블링되는지 여부에 적어도 부분적으로 기초한다. 현재의 슬라이스가 P 슬라이스인지 또는 B 슬라이스인지에 의존하여 및/또는 어떤 RPL이 시그널링/파싱되는지에 의존하여 가중치 부여된 예측이 디스에이블링되는지 여부를 확인하는데 상이한 로직이 사용될 수 있다. 예를 들어, (P 슬라이스 또는 B 슬라이스에 의해 사용될 수 있는) 제1 RPL에 대한 조건을 확인하는 로직은 (B 슬라이스에 의해서만 사용될 수 있는) 제2 RPL에 대한 조건을 확인하는 로직과는 다르다. 이와 달리, 인코더는 다른 및/또는 추가의 조건을 평가한다.

[0136] 평가의 결과에 따라, 장치는 RPL을 어떻게 변경(예를 들어, 대체, 조정)할지를 나타내는 리스트 엔트리들에 대한 하나 이상의 신택스 요소를 비트스트림 내에 시그널링하는 것을 조정한다(1420). 특히, 신택스 요소(들) 중 적어도 하나의 길이(비트들)가 조정된다. 예를 들어, 리스트 엔트리들에 대한 인덱스 i에 대해, 가중치 부여된 예측이 디스에이블링되는 경우, 신택스 요소들 중 적어도 하나의 길이(비트들)는 i가 증가함에 따라 감소한다. 몇몇 예시적인 구현에서, 가중치 부여된 예측이 디스에이블링되는 경우, 리스트 엔트리[i]에 대한 주어진 신택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr-i))이다. 그렇지 않은 경우(가중치 부여된 예측이 인에이블링되는 경우), 리스트 엔트리[i]에 대한 주어진 신택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr)) 비트이다.

[0137] 장치는 RPL 변경 구조가 시그널링된 경우 슬라이스 단위로, 또는 몇몇 다른 단위로 기법(1400)을 반복할 수 있다.

[0138] 도 15는 RPL 변경을 위한 리스트 엔트리들의 파싱을 조정하기 위한 일반화된 기법(1500)을 나타낸다. 예를 들어, 도 4를 참조하여 설명한 비디오 디코더를 구현하는 컴퓨팅 장치가 이 기법(1500)을 수행할 수 있다.

[0139] 장치는 조건을 평가한다(1510). 예를 들어, 조건은 가중치 부여된 예측이 디스에이블링되는지 여부에 적어도 부분적으로 기초한다. 현재의 슬라이스가 P 슬라이스인지 또는 B 슬라이스인지에 의존하여 및/또는 어떤 RPL이 시그널링/파싱되는지에 의존하여 가중치 부여된 예측이 디스에이블링되는지 여부를 확인하는데 상이한 로직이 사용될 수 있다. 이와 달리, 디코더는 다른 및/또는 추가의 조건을 평가한다.

[0140] 평가의 결과에 따라, 장치는 RPL을 어떻게 변경(예를 들어, 대체, 조정)할지를 나타내는 리스트 엔트리들에 대한 하나 이상의 신택스 요소를 비트스트림으로부터 파싱하는 것을 조정한다(1520). 특히, 신택스 요소(들) 중 적어도 하나의 길이(비트들)가 조정된다. 예를 들어, 리스트 엔트리들에 대한 인덱스 i에 대해, 가중치 부여된 예측이 디스에이블링되는 경우, 신택스 요소들 중 적어도 하나의 길이(비트들)는 i가 증가함에 따라 감소한다. 몇몇 예시적인 구현에서, 가중치 부여된 예측이 디스에이블링되는 경우, 리스트 엔트리[i]에 대한 주어진 신택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr-i))이다. 그렇지 않은 경우(가중치 부여된 예측이 인에이블링되는 경우), 리스트 엔트리[i]에 대한 주어진 신택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr)) 비트이다.

[0141] 장치는 RPL 변경 구조가 시그널링된 경우 슬라이스 단위로, 또는 몇몇 다른 단위로 기법(1500)을 반복할 수 있다.

#### G. 대체 방안들

[0143] 도 7a, 7b, 10 및 11은 조건에 기초한 ref\_pic\_list\_modification\_flag\_10 또는 ref\_pic\_list\_modification\_flag\_11과 같은 플래그의 조건부적 시그널링 및 과정을 나타낸다. 이러한 식으로, 추가의 RPL 변경 정보(예를 들어, 리스트 엔트리들에 대한 신택스 요소)의 시그널링이 제어된다. 도 7a를 참조하여 설명되는 바와 같이, RPL 변경 플래그의 시그널링 및 과정은 ref\_pic\_lists\_modification 구조의 일부로서 조건을 평가함으로써 제어될 수 있다. 이와 달리, 7b를 참조하여 설명되어 있는 바와 같이, RPL 변경 구조(예를 들어, ref\_pic\_lists\_modification 구조)의 시그널링 및 과정은 슬라이스 헤더 프로세싱 또는 그 외의 일부로서 동일한 조건을 평가함으로써 제어될 수 있다. 예를 들어, 변수 NumPocTotalCurr이 1보다 큰 경우, ref\_pic\_lists\_modification() 구조는 시그널링된다. 그와 달리(변수 NumPocTotalCurr이 1보다 크지 않은 경우), ref\_pic\_lists\_modification() 구조는 시그널링되지 않고, 도 7a를 참조하여 설명한 바와 같이 리스트 엔트리들의 값은 추론된다. 도 10으로 확장하면, 조건이 평가된 이후, 평가의 결과에 따라, RPL 변경 신택스 구조는 조건부로 시그널링된다. 도 11로 확장하면, 조건이 평가된 이후, 평가의 결과에 따라, RPL 변경 신택스 구조는 조건부로 과정된다.

[0144] 설명을 위해, 상세한 설명은 몇몇 파라미터 및 변수에 대한 특정 이름을 갖는 다양한 예를 포함한다. 본 명세서에서 기술된 혁신들은 이러한 이름을 갖는 파라미터 또는 변수를 갖는 구현에 국한되지 않는다. 대신, 본 명세서에서 기술된 혁신들은 다양한 유형의 파라미터 및 변수로 구현될 수 있다.

#### H. 추가의 혁신적인 특징들

[0146] 청구항에 더해, 본 명세서에서 기술된 혁신적인 특징은 이하의 표에 도시된 특징들을 포함하나 이에 국한되지 않는다.

#	특징
A. RPL의 리스트 엔트리들에 대한 신택스 요소들의 조건부적 시그널링	조건을 평가하는 단계와,
A1	상기 평가의 결과에 기초하여, RPL을 어떻게 변경할 것인지를 나타내는 리스트 엔트리들에 대한 하나 이상의 신택스 요소를 비트스트림 내에 조건부로 시그널링하는 단계를 포함하는 비디오 인코더에 의해 수행되는 방법.
A2	조건을 평가하는 단계와,
A3	상기 평가의 결과에 기초하여, RPL을 어떻게 변경할 것인지를 나타내는 리스트 엔트리들에 대한 하나 이상의 신택스 요소를 비트스트림으로부터 조건부로 과정하는 단계를 포함하는 비디오 디코더에 의해 수행되는 방법.
A4	상기 조건은 전체 참조 화상의 수를 나타내는 변수에 적어도 부분적으로 의존하는, 특정 A1 또는 A2의 방법.
A5	상기 변수는 NumPocTotalCurr인, 특정 A3의 방법.
A6	상기 조건은 A4 중 어느 하나의 방법.
A7	현재의 슬라이스가 P 슬라이스인지 B 슬라이스인지에 의존하여 및/또는 어떤 RPL이 시그널링/과정되는지에 의존하여, 가중치 부여된 예측이 디스에이블링되는지 여부를 확인하는데 상이한 로직이 사용되는, 특정 A6의 방법.
A8	상기 조건은 (a) 전체 참조 화상의 수가 2인자와 (b) RPL에 대한 활성 참조 화상의 수가 1인자에 적어도 부분적으로 의존하는, 특정 A1 또는 A2의 방법.
A9	(a) 전체 참조 화상의 수가 2이고 (b) RPL에 대한 활성 참조 화상의 수가 1인 경우, 리스트 엔트리들에 대한 하나 이상의 신택스 요소는 상기 비트스트림으로부터 빠져있고, 상기 리스트 엔트리들 중 하나의 값이 추론되는, 특정 A8의 방법.
A10	상기 조건은 (c) 전체 참조 화상의 수가 2이고, (d) 상기 RPL에 대한 활성 참조 화상의 수가 2이며 (e) 가중치 부여된 예측이 디스에이블링되는지 여부에 적어도 부분적으로 의존하는, 특정 A1 또는 A2의 방법.
A11	(c) 상기 전체 참조 화상의 수가 2이고, (d) 상기 RPL에 대한 활성 참조 화상의 수가 2이며 (e) 상기 가중치 부여된 예측이 디스에이블링되는 경우, 리스트 엔트리들에 대한 하나 이상의 신택스 요소는 상기 비트스트림으로부터 빠져있고, 상기 리스트 엔트리들 중 두 개의 값이 추론되는, 특정 A10의 방법.
A12	상기 리스트 엔트리들에 대한 하나 이상의 신택스 요소는 슬라이스 헤더의 RPL 변경 구조의 일부로서 조건부로 시그널링되는, 특정 A1 내지 A11 중 어느 하나의 방법.
A13	상기 RPL은 P 슬라이스와 연관된 RPL 0인, 특정 A1 내지 A12 중 어느 하나의 방법.
A14	다수의 RPL 각각에 대한 상기 평가 및 상기 조건부적 시그널링 또는 과정을 반복하는 단계를 더 포함하되, 상기 다수의 RPL은 B 슬라이스와 연관된 RPL 0 및 RPL 1을 포함하는, 특정 A1 내지 A12 중 어느 하나의 방법.
A15	특정 A1 내지 A14 중 어느 하나의 방법을 수행하도록 구성된 컴퓨팅 장치.
A16	컴퓨팅 장치로 하여금 특정 A1 내지 A14 중 어느 하나의 방법을 수행하게 하는 컴퓨터 실행 가능 명령어들을 저장하는 유형의 컴퓨터 판독 가능 매체.

[0147]

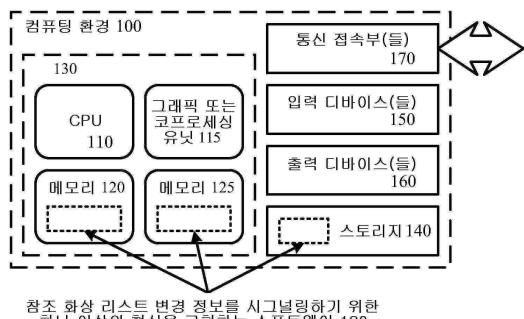
B. RPL의 리스트 엔트리들에 대한 신택스 요소의 길이 조정	
B1	조건을 평가하는 단계와, 상기 평가의 결과에 따라, RPL을 어떻게 변경할 것인지를 나타내는 리스트 엔트리들에 대한 하나 이상의 신택스 요소를 비트스트림 내에 시그널링하는 것을 조정하는 단계- 상기 하나 이상의 신택스 요소 중 적어도 하나의 길이가 조정됨 - 를 포함하는 비디오 디코더에 의해 수행되는 방법.
B2	조건을 평가하는 단계와, 상기 평가의 결과에 따라, RPL을 어떻게 변경할 것인지를 나타내는 리스트 엔트리들에 대한 하나 이상의 신택스 요소를 비트스트림으로부터 파싱하는 것을 조정하는 단계- 상기 하나 이상의 신택스 요소 중 적어도 하나의 길이가 조정됨 - 를 포함하는 비디오 디코더에 의해 수행되는 방법.
B3	상기 조건은 가중치 부여된 예측이 디스에이블링되는지에 적어도 부분적으로 의존하는, 특정 B1 또는 B2의 방법.
B4	현재의 슬라이스가 P 슬라이스인지 B 슬라이스인지에 의존하여 및/또는 어떤 RPL이 시그널링/파싱되는지에 의존하여 가중치 부여된 예측이 디스에이블링되는지 여부를 확인하는데 상이한 로직이 사용되는, 특정 B3의 방법.
B5	상기 리스트 엔트리들에 대한 인덱스 i에 대해, 가중치 부여된 예측이 디스에이블링되는 경우, 상기 신택스 요소의 적어도 하나의 길이는 i가 증가함에 따라 감소하는, 특정 B3의 방법.
B6	상기 리스트 엔트리들에 대한 인덱스 i에 대해, 가중치 부여된 예측이 디스에이블링되는 경우, 리스트 엔트리[i]에 대해 주어진 신택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr-i)) 비트이고, 가중치 부여된 예측이 디스에이블링되는 경우, 리스트 엔트리[i]에 대해 주어진 신택스 요소의 길이는 Ceil(Log2(NumPocTotalCurr)) 비트인, 특정 B3의 방법.
B7	특정 B1 내지 B6 중 어느 하나의 방법을 수행하도록 구성된 컴퓨팅 장치.
B8	컴퓨팅 장치로 하여금 특정 B1 내지 B6 중 어느 하나의 방법을 수행하게 하는 컴퓨터 실행 가능 명령어들을 저장하는 유형의 컴퓨터 판독가능 매체.
C. 일반적인 특징	
C1	비디오를 인코딩하는 단계와, 본 명세서에서 기술한 혁신들 중 하나에 따라 RPL 정보를 시그널링하는 것을 포함하여, 인코딩된 비디오를 포함한 비트스트림의 적어도 일부를 출력하는 단계를 포함하는, 인코더에 의해 수행되는 방법.
C2	본 명세서에서 기술한 혁신들 중 하나에 따라 시그널링된 RPL 정보를 파싱하는 것을 포함하여, 인코딩된 비디오를 포함한 비트스트림의 적어도 일부를 수신하는 단계와, 상기 인코딩된 비디오를 디코딩하는 단계를 포함하는, 디코더에 의해 수행되는 방법.
C3	특정 C1 또는 C2의 방법을 수행하도록 구성된 컴퓨팅 장치.
C4	컴퓨팅 장치로 하여금 특정 C1 또는 C2의 방법을 수행하게 하는 컴퓨터 실행 가능 명령어들을 저장하는 유형의 컴퓨터 판독가능 매체.

[0148]

[0149] 개시된 발명의 원리가 적용될 수 있는 많은 가능한 실시예의 관점에서, 도시된 실시예는 본 발명의 바람직한 예일 뿐이며 본 발명의 범위를 제한하는 것으로 고려되어서는 안된다는 것을 인식해야 한다. 오히려, 본 발명의 범위는 이하의 청구항에 의해 정의된다. 그러므로, 이들 청구항의 범위 및 사상 내에 해당하는 모든 것을 본 발명으로서 청구한다.

## 도면

### 도면1

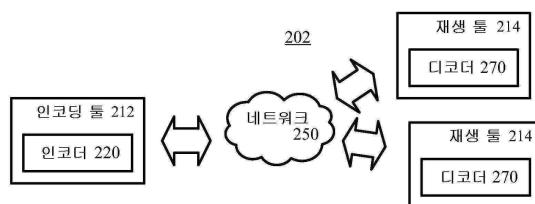


### 도면2a

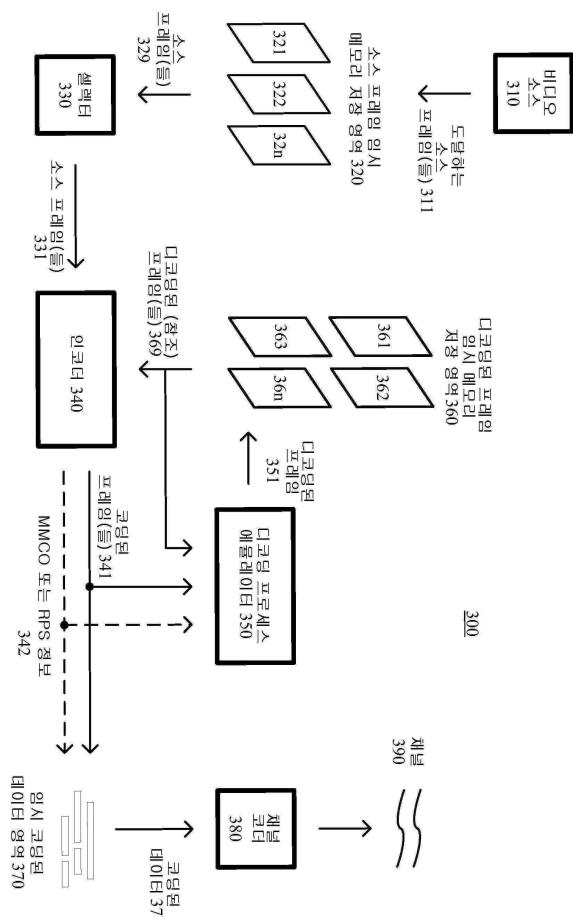
201



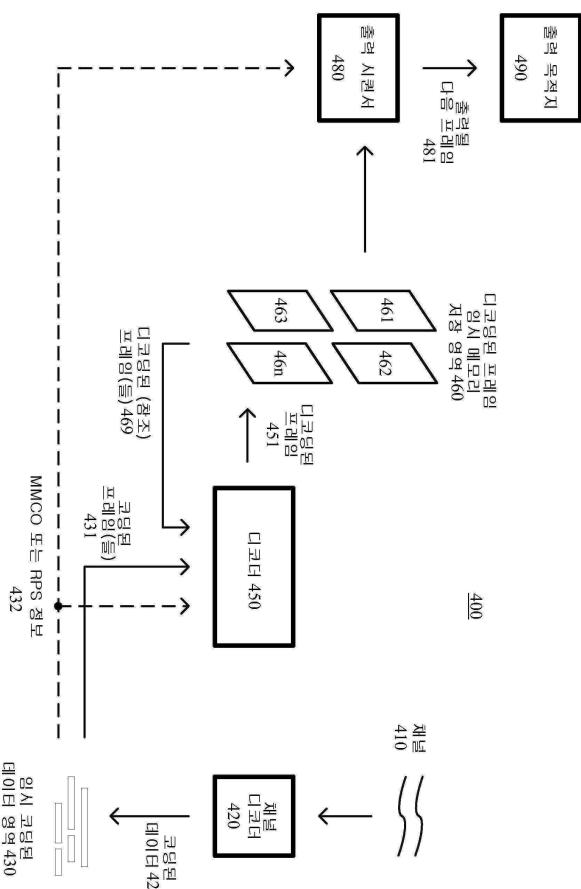
## 도면2b



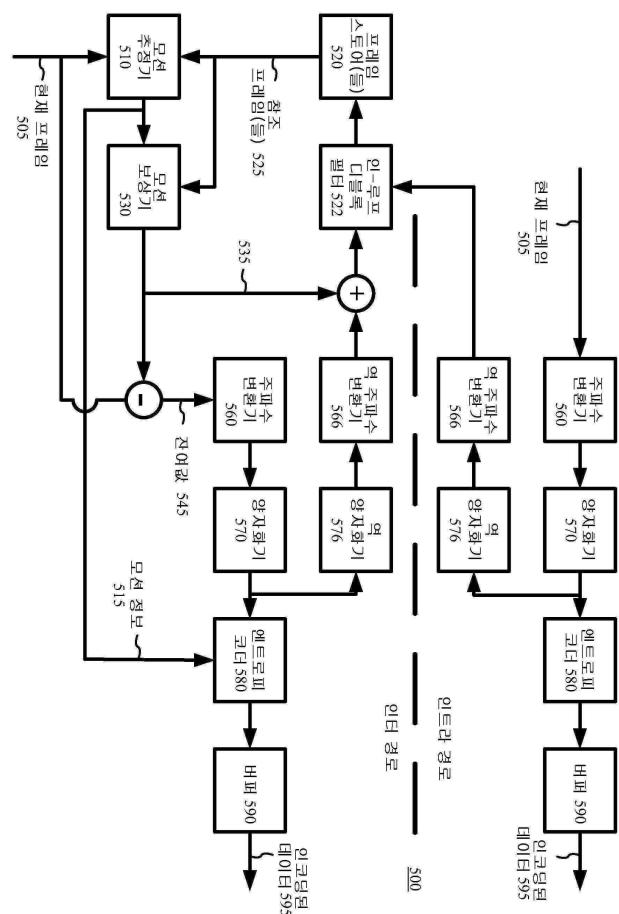
## 도면3



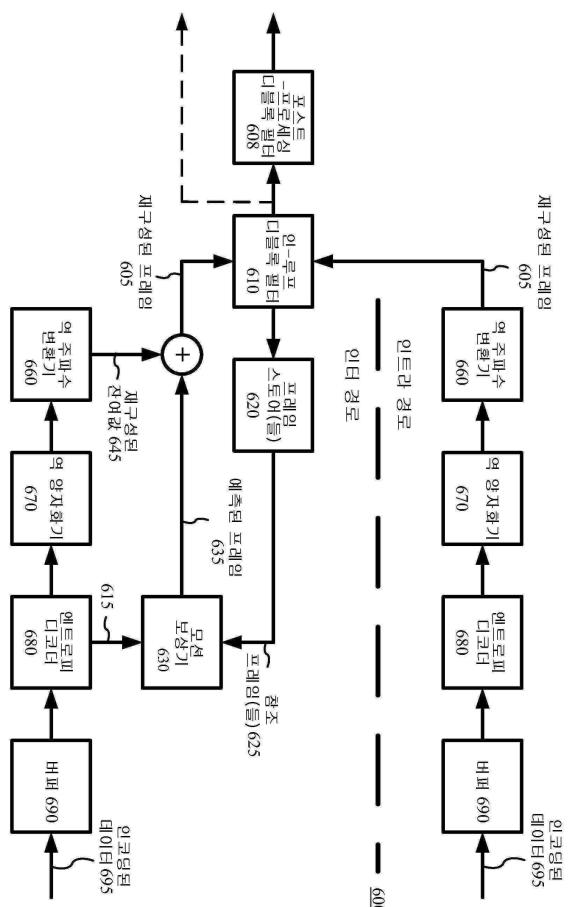
## 도면4



도면5



## 도면6



### 도면7a

700

```

ref_pic_lists_modification() {
    if( NumPocTotalCurr > 1 )
        ref_pic_list_modification_flag_10
    if( ref_pic_list_modification_flag_10 )
        for( i = 0; i <= num_ref_idx_10_active_minus1; i++ )
            list_entry_10[ i ]
    if( slice_type == B ) {
        if( NumPocTotalCurr > 1 )
            ref_pic_list_modification_flag_11
        if( ref_pic_list_modification_flag_11 )
            for( i = 0; i <= num_ref_idx_11_active_minus1; i++ )
                list_entry_11[ i ]
    }
}

```

도면7b

750

```

slice_segment_header() {
    ...
    if( slice_type == P || slice_type == B ) {
        num_ref_idx_active_override_flag
        u(1)
        if( num_ref_idx_active_override_flag ) {
            num_ref_idx_10_active_minus1
            uc(v)
            if( slice_type == B )
                num_ref_idx_11_active_minus1
                uc(v)
        }
    }
    if( list_modification_present_flag && NumPocTotalCurr > 1 )
        ref_pic_lists_modification()
    ...
}

```

**도면7c**760

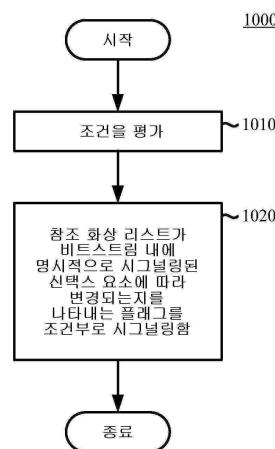
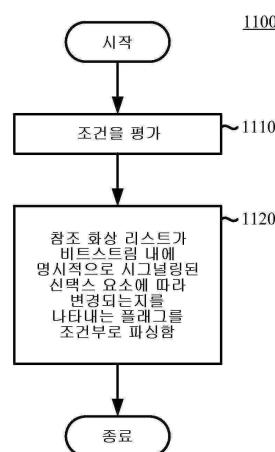
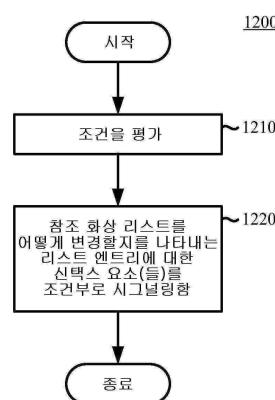
ref_pic_lists_modification()	설명자
ref_pic_list_modification_flag_10	u(1)
if ref_pic_list_modification_flag_10	
for(i = 0; i <= num_ref_idx_l0_active_minus1; i++)	
list_entry_10[i]	u(v)
if slice_type == B {	
ref_pic_list_modification_flag_11	u(1)
if ref_pic_list_modification_flag_11	
for(i = 0; i <= num_ref_idx_l1_active_minus1; i++)	
list_entry_11[i]	u(v)
}	
}	
}	

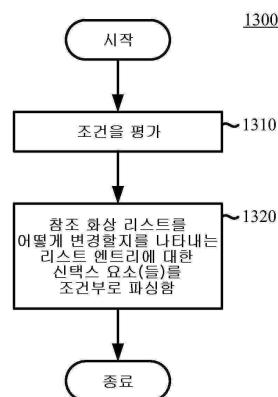
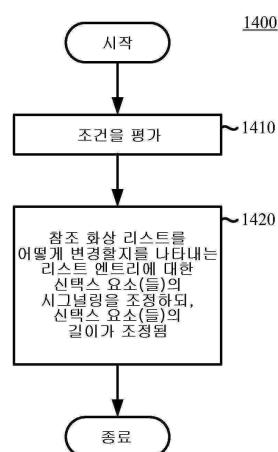
**도면8**800

ref_pic_lists_modification()	설명자
ref_pic_list_modification_flag_10	u(1)
if ref_pic_list_modification_flag_10 &&	
!(NumPocTotalCurr == 2 &&	
num_ref_idx_l0_active_minus1 == 0))	
for(i = 0; i <= num_ref_idx_l0_active_minus1; i++)	
list_entry_10[i]	u(v)
if slice_type == B {	
ref_pic_list_modification_flag_11	u(1)
if ref_pic_list_modification_flag_11 &&	
!(NumPocTotalCurr == 2 &&	
num_ref_idx_l1_active_minus1 == 0))	
for(i = 0; i <= num_ref_idx_l1_active_minus1; i++)	
list_entry_11[i]	u(v)
}	
}	
}	

**도면9**900

ref_pic_lists_modification()	설명자
ref_pic_list_modification_flag_10	u(1)
if ref_pic_list_modification_flag_10 &&	
!(NumPocTotalCurr == 2 && num_ref_idx_l0_active_minus1 == 0) &&	
!(NumPocTotalCurr == 2 && num_ref_idx_l0_active_minus1 == 1) &&	
( (weighted_pred_flag != 1 && slice_type == P)	
( weighted_bipred_flag != 1 && slice_type == B ) ) )	
for(i = 0; i <= num_ref_idx_l0_active_minus1; i++)	
list_entry_10[i]	u(v)
if slice_type == B {	
ref_pic_list_modification_flag_11	u(1)
if ref_pic_list_modification_flag_11 &&	
!(NumPocTotalCurr == 2 && num_ref_idx_l1_active_minus1 == 0) &&	
!(NumPocTotalCurr == 2 && num_ref_idx_l1_active_minus1 == 1) &&	
weighted_bipred_flag != 1 ) )	
for(i = 0; i <= num_ref_idx_l1_active_minus1; i++)	
list_entry_11[i]	u(v)
}	
}	

**도면10****도면11****도면12**

**도면13****도면14****도면15**