



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2014-0120342
(43) 공개일자 2014년10월13일

(51) 국제특허분류(Int. Cl.)
H04N 19/86 (2014.01) H04N 19/70 (2014.01)
H04N 19/30 (2014.01)
(21) 출원번호 10-2014-7022908
(22) 출원일자(국제) 2013년01월18일
심사청구일자 없음
(85) 번역문제출일자 2014년08월14일
(86) 국제출원번호 PCT/US2013/022230
(87) 국제공개번호 WO 2013/109946
국제공개일자 2013년07월25일
(30) 우선권주장
13/743,592 2013년01월17일 미국(US)
(뒷면에 계속)

(71) 출원인
켈컴 인코퍼레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
(72) 발명자
판 더 아우베라 게르트
미국 92121 캘리포니아주 샌디에고 모어하우스 드라이브 5775
왕 예-쿠이
미국 92121 캘리포니아주 샌디에고 모어하우스 드라이브 5775
카르체비츠 마르타
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
(74) 대리인
특허법인코리아나

전체 청구항 수 : 총 60 항

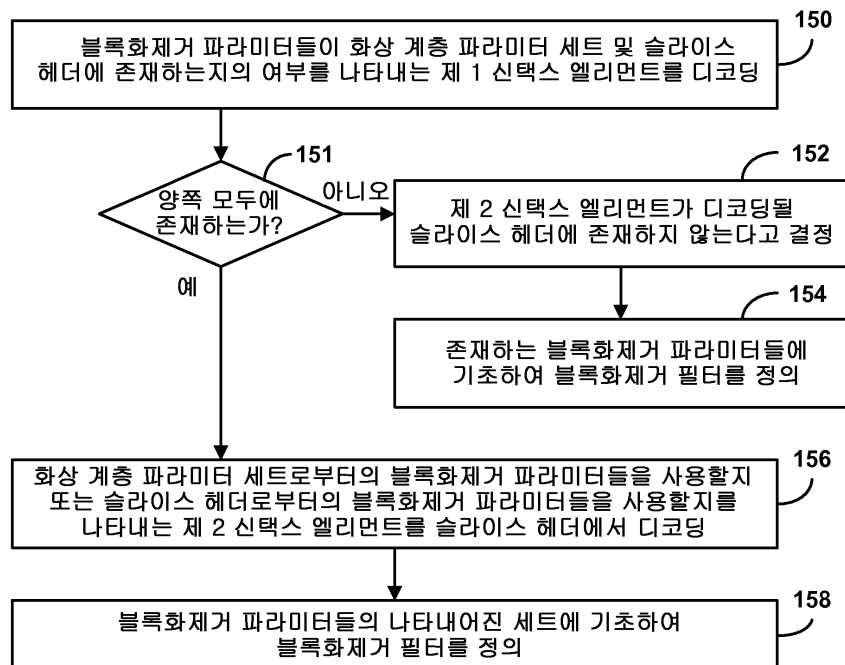
(54) 발명의 명칭 비디오 코딩에서 블록화제거 필터 파라미터들의 시그널링

(57) 요약

본 개시물은 감소된 비트스트림 오버헤드를 가지고서 비디오 데이터의 현재 슬라이스에 대한 블록화제거 필터 파라미터들을 시그널링하는 기법들을 설명한다. 블록화제거 필터 파라미터들은 화상 계층 파라미터 세트 및 슬라이스 헤더 중 하나 이상에서 코딩될 수도 있다. 그 기법들은, 블록화제거 필터 파라미터들이 상기 화상 계

(뒷면에 계속)

대표도 - 도7



층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내는 제 1 선택스 엘리먼트를 디코딩하고, 블록화제거 필터 파라미터들의 세트들 양쪽 모두가 존재하는 경우, 상기 슬라이스 헤더에서의 제 2 선택스 엘리먼트만을 코딩함으로써, 블록화제거 필터 파라미터들을 시그널링하는데 사용된 비트들의 수를 감소시킨다. 제 2 선택스 엘리먼트를 코딩하는 것은, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 하나에만 존재하는 경우 제거된다. 제 2 선택스 엘리먼트는 블록화제거 필터 파라미터들 중 어떤 세트가 현재 슬라이스에 적용되는 블록화제거 필터를 정의하는데 사용되는지를 나타낸다.

(30) 우선권주장

61/588,454	2012년01월19일	미국(US)
61/593,015	2012년01월31일	미국(US)
61/620,339	2012년04월04일	미국(US)

특허청구의 범위

청구항 1

비디오 데이터를 디코딩하는 방법으로서,

블록화제거 (deblocking) 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 디코딩하는 단계;

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재함을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 상기 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 디코딩하는 단계; 및

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 제 2 선택스 엘리먼트가 디코딩될 상기 슬라이스 헤더에 존재하지 않는다고 결정하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 2

제 1 항에 있어서,

상기 화상 계층 파라미터 세트는, 화상 파라미터 세트 (PPS) 또는 적응 파라미터 세트 (APS) 중 하나를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 3

제 1 항에 있어서,

상기 제 1 선택스 엘리먼트를 디코딩하는 단계는, 상기 화상 계층 파라미터 세트 또는 시퀀스 파라미터 세트 (SPS) 중 하나에서의 상기 제 1 선택스 엘리먼트를 디코딩하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 4

제 1 항에 있어서,

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 중 하나에 존재하는 블록화제거 필터 파라미터들에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 5

제 1 항에 있어서,

상기 제 1 선택스 엘리먼트를 디코딩하는 단계는, 블록화제거 필터 파라미터들의 상기 제 2 세트에 의한 블록화제거 필터 파라미터들의 상기 제 1 세트의 오버라이드가 가능하게 되는지의 여부를 나타내는 오버라이드 가능 플래그를 상기 화상 계층 파라미터 세트에서 디코딩하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 6

제 5 항에 있어서,

상기 오버라이드가 가능하게될 경우, 상기 제 2 선택스 엘리먼트를 디코딩하는 단계는, 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트로부터의 블록화제거 필터 파라미터들의 상기 제 1 세트를 사용할지 또는 블록화제거 필터 파라미터들의 상기 제 1 세트를 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트로 오버라이드할지를 나타내는 오버라이드 플래그를

상기 슬라이스 헤더에서 디코딩하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 7

제 5 항에 있어서,

상기 오버라이드가 가능하지 않을 경우, 상기 제 2 선택스 엘리먼트가 상기 슬라이스 헤더에 존재하지 않는다고 결정하는 단계는, 오버라이드 플래그가 디코딩될 상기 슬라이스 헤더에 존재하지 않는다고 결정하는 단계를 포함하며,

상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 상기 제 1 세트에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 8

제 1 항에 있어서,

상기 제 1 선택스 엘리먼트를 디코딩하는 단계는, 블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 상기 화상 계층 파라미터 세트에 존재하는지의 여부를 나타내는 상속 가능 플래그를 디코딩하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 9

제 8 항에 있어서,

블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 존재하는 경우, 상기 제 2 선택스 엘리먼트를 디코딩하는 단계는, 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트를 사용할지 또는 상기 화상 계층 파라미터 세트로부터의 블록화제거 필터 파라미터들의 상기 제 1 세트를 상속할지를 나타내는 상속 플래그를 상기 슬라이스 헤더에서 디코딩하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 10

제 8 항에 있어서,

블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 존재하지 않는 경우, 상기 제 2 선택스 엘리먼트가 상기 슬라이스 헤더에 존재하지 않는다고 결정하는 단계는, 상속 플래그가 디코딩될 상기 슬라이스 헤더에 존재하지 않는다고 결정하는 단계를 포함하며,

상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 11

제 1 항에 있어서,

상기 제 1 선택스 엘리먼트를 디코딩하기 전에, 블록화제거 필터 제어 선택스 엘리먼트들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더에 존재한다는 것을 나타내도록 정의된 제 3 선택스 엘리먼트를 디코딩하는 단계를 더 포함하며,

상기 제 1 선택스 엘리먼트 및 상기 제 2 선택스 엘리먼트는 블록화제거 필터 제어 선택스 엘리먼트들을 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 12

제 11 항에 있어서,

상기 제 3 선택스 엘리먼트를 디코딩하는 단계는, 상기 화상 계층 파라미터 세트 및 시퀀스 파라미터 세트 (SPS) 중 하나에서의 제어 존재 선택스 엘리먼트를 디코딩하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 13

제 1 항에 있어서,

상기 블록화제거 필터가 상기 현재 비디오 슬라이스에 대해 가능하게 됨을 나타내도록 정의된 제 4 선택스 엘리먼트를 디코딩하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 14

제 13 항에 있어서,

상기 제 4 선택스 엘리먼트를 디코딩하는 단계는, 상기 제 1 선택스 엘리먼트를 디코딩하기 전에 시퀀스 파라미터 세트 (SPS) 에서의 블록화제거 필터 가능 플래그를 디코딩하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 15

비디오 디코딩 디바이스로서,

비디오 데이터를 저장하는 메모리; 및

블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 디코딩하도록 구성된 프로세서를 포함하며,

상기 프로세서는, 블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재함을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 상기 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 디코딩하도록 구성되며, 그리고

상기 프로세서는, 블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 제 2 선택스 엘리먼트가 디코딩될 상기 슬라이스 헤더에 존재하지 않는다고 결정하도록 구성되는, 비디오 디코딩 디바이스.

청구항 16

제 15 항에 있어서,

상기 화상 계층 파라미터 세트는, 화상 파라미터 세트 (PPS) 또는 적응 파라미터 세트 (APS) 중 하나를 포함하는, 비디오 디코딩 디바이스.

청구항 17

제 15 항에 있어서,

상기 프로세서는, 상기 화상 계층 파라미터 세트 또는 시퀀스 파라미터 세트 (SPS) 중 하나에서의 상기 제 1 선택스 엘리먼트를 디코딩하도록 구성되는, 비디오 디코딩 디바이스.

청구항 18

제 15 항에 있어서,

상기 프로세서는, 블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 중 하나에 존재하는 블록화제거 필터 파라미터들에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하도록 구성되는, 비디오 디코딩 디바이스.

청구항 19

제 15 항에 있어서,

상기 프로세서는, 블록화제거 필터 파라미터들의 상기 제 2 세트에 의한 블록화제거 필터 파라미터들의 상기 제

1 세트의 오버라이드가 가능하게 되는지의 여부를 나타내는 오버라이드 가능 플래그를 상기 화상 계층 파라미터 세트에서 디코딩하도록 구성되는, 비디오 디코딩 디바이스.

청구항 20

제 19 항에 있어서,

상기 프로세서는, 상기 오버라이드가 가능하게될 경우, 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트로부터의 블록화제거 필터 파라미터들의 상기 제 1 세트를 사용할지 또는 블록화제거 필터 파라미터들의 상기 제 1 세트를 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트로 오버라이드할지를 나타내는 오버라이드 플래그를 상기 슬라이스 헤더에서 디코딩하도록 구성되는, 비디오 디코딩 디바이스.

청구항 21

제 19 항에 있어서,

상기 프로세서는, 상기 오버라이드가 가능하지 않을 경우, 오버라이드 플래그가 디코딩될 상기 슬라이스 헤더에 존재하지 않는다고 결정하고, 상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 상기 제 1 세트에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하도록 구성되는, 비디오 디코딩 디바이스.

청구항 22

제 15 항에 있어서,

상기 프로세서는, 블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 상기 화상 계층 파라미터 세트에 존재하는지의 여부를 나타내는 상속 가능 플래그를 디코딩하도록 구성되는, 비디오 디코딩 디바이스.

청구항 23

제 22 항에 있어서,

상기 프로세서는, 블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 존재하는 경우, 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트를 사용할지 또는 상기 화상 계층 파라미터 세트로부터의 블록화제거 필터 파라미터들의 상기 제 1 세트를 상속할지를 나타내는 상속 플래그를 상기 슬라이스 헤더에서 디코딩하도록 구성되는, 비디오 디코딩 디바이스.

청구항 24

제 22 항에 있어서,

상기 프로세서는, 블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 존재하지 않는 경우, 상속 플래그가 디코딩될 상기 슬라이스 헤더에 존재하지 않는다고 결정하고, 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하도록 구성되는, 비디오 디코딩 디바이스.

청구항 25

비디오 디코딩 디바이스로서,

블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 디코딩하는 수단;

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재함을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 상기 슬라이스

스 헤더에서의 제 2 선택스 엘리먼트를 디코딩하는 수단; 및

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 제 2 선택스 엘리먼트가 디코딩될 상기 슬라이스 헤더에 존재하지 않는다고 결정하는 수단을 포함하는, 비디오 디코딩 디바이스.

청구항 26

제 25 항에 있어서,

상기 화상 계층 파라미터 세트는 화상 파라미터 세트 (PPS) 를 포함하는, 비디오 디코딩 디바이스.

청구항 27

제 25 항에 있어서,

상기 제 1 선택스 엘리먼트를 디코딩하는 수단은, 블록화제거 필터 파라미터들의 상기 제 2 세트에 의한 블록화제거 필터 파라미터들의 상기 제 1 세트의 오버라이드가 가능하게 되는지의 여부를 나타내는 오버라이드 가능 플래그를 상기 화상 계층 파라미터 세트에서 디코딩하는 수단을 포함하는, 비디오 디코딩 디바이스.

청구항 28

제 27 항에 있어서,

상기 오버라이드가 가능하게될 경우, 상기 제 2 선택스 엘리먼트를 디코딩하는 수단은, 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트로부터의 블록화제거 필터 파라미터들의 상기 제 1 세트를 사용할지 또는 블록화제거 필터 파라미터들의 상기 제 1 세트를 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트로 오버라이드할지를 나타내는 오버라이드 플래그를 상기 슬라이스 헤더에서 디코딩하는 수단을 포함하는, 비디오 디코딩 디바이스.

청구항 29

제 27 항에 있어서,

상기 오버라이드가 가능하지 않을 경우, 상기 제 2 선택스 엘리먼트가 상기 슬라이스 헤더에 존재하지 않는다고 결정하는 수단은, 오버라이드 플래그가 디코딩될 상기 슬라이스 헤더에 존재하지 않는다고 결정하는 수단을 포함하며,

상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 상기 제 1 세트에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하는 수단을 더 포함하는, 비디오 디코딩 디바이스.

청구항 30

비디오 데이터를 디코딩하는 명령들을 포함하는 컴퓨터 판독가능 매체로서,

상기 명령들은, 실행되는 경우, 하나 이상의 프로세서들로 하여금,

블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 디코딩하게 하고;

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재함을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 상기 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 디코딩하게 하며; 그리고

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 제 2 선택스 엘리먼트가 디코딩될 상기 슬라이스 헤더에 존재하지 않는다고 결정하게 하는, 컴퓨터 판독가능 매체.

청구항 31

비디오 데이터를 인코딩하는 방법으로서,

블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 인코딩하는 단계;

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재함을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 제 2 선택스 엘리먼트를 상기 슬라이스 헤더에서 인코딩하는 단계; 및

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 슬라이스 헤더에서의 상기 제 2 선택스 엘리먼트의 인코딩을 제거하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 32

제 31 항에 있어서,

상기 화상 계층 파라미터 세트는, 화상 파라미터 세트 (PPS) 또는 적응 파라미터 세트 (APS) 중 하나를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 33

제 31 항에 있어서,

상기 제 1 선택스 엘리먼트를 인코딩하는 단계는, 상기 화상 계층 파라미터 세트 또는 시퀀스 파라미터 세트 (SPS) 중 하나에서의 상기 제 1 선택스 엘리먼트를 인코딩하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 34

제 31 항에 있어서,

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 중 하나에 존재하는 블록화제거 필터 파라미터들에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 35

제 31 항에 있어서,

상기 제 1 선택스 엘리먼트를 인코딩하는 단계는, 블록화제거 필터 파라미터들의 상기 제 2 세트에 의한 블록화제거 필터 파라미터들의 상기 제 1 세트의 오버라이드가 가능하게 되는지의 여부를 나타내는 오버라이드 가능 플래그를 상기 화상 계층 파라미터 세트에서 인코딩하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 36

제 35 항에 있어서,

상기 오버라이드가 가능하게될 경우, 상기 제 2 선택스 엘리먼트를 인코딩하는 단계는, 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트로부터의 블록화제거 필터 파라미터들의 상기 제 1 세트를 사용할지 또는 블록화제거 필터 파라미터들의 상기 제 1 세트를 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트로 오버라이드할지를 나타내는 오버라이드 플래그를 상기 슬라이스 헤더에서 인코딩하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 37

제 35 항에 있어서,

상기 오버라이드가 가능하지 않을 경우, 상기 제 2 선택스 엘리먼트의 인코딩을 제거하는 단계는, 상기 슬라이스 헤더에서의 오버라이드 플래그의 인코딩을 제거하는 단계를 포함하며,

상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 상기 제 1 세트에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 38

제 31 항에 있어서,

상기 제 1 선택스 엘리먼트를 인코딩하는 단계는, 블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 상기 화상 계층 파라미터 세트에 존재하는지의 여부를 나타내는 상속 가능 플래그를 인코딩하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 39

제 38 항에 있어서,

블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 존재하는 경우, 상기 제 2 선택스 엘리먼트를 인코딩하는 단계는, 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트를 사용할지 또는 상기 화상 계층 파라미터 세트로부터의 블록화제거 필터 파라미터들의 상기 제 1 세트를 상속할지를 나타내는 상속 플래그를 상기 슬라이스 헤더에서 인코딩하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 40

제 38 항에 있어서,

블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 존재하지 않는 경우, 상기 제 2 선택스 엘리먼트의 인코딩을 제거하는 단계는, 상기 슬라이스 헤더에서의 상속 플래그의 인코딩을 제거하는 단계를 포함하며,

상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 41

제 31 항에 있어서,

상기 제 1 선택스 엘리먼트를 인코딩하기 전에 블록화제거 필터 제어 선택스 엘리먼트들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더에 존재한다는 것을 나타내도록 정의된 제 3 선택스 엘리먼트를 인코딩하는 단계를 더 포함하며,

상기 제 1 선택스 엘리먼트 및 상기 제 2 선택스 엘리먼트는 블록화제거 필터 제어 선택스 엘리먼트들을 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 42

제 41 항에 있어서,

상기 제 3 선택스 엘리먼트를 인코딩하는 단계는, 상기 화상 계층 파라미터 세트 및 시퀀스 파라미터 세트 (SPS) 중 하나에서의 제어 존재 선택스 엘리먼트를 인코딩하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 43

제 31 항에 있어서,

상기 블록화제거 필터가 상기 현재 비디오 슬라이스에 대해 가능하게 됨을 나타내도록 정의된 제 4 선택스 엘리먼트를 인코딩하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 44

제 43 항에 있어서,

상기 제 4 선택스 엘리먼트를 인코딩하는 단계는, 상기 제 1 선택스 엘리먼트를 인코딩하기 전에 시퀀스 파라미터 세트 (SPS) 에서의 블록화제거 필터 가능 플래그를 인코딩하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 45

비디오 인코딩 디바이스로서,

비디오 데이터를 저장하는 메모리; 및

블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 인코딩하도록 구성된 프로세서를 포함하며,

상기 프로세서는, 블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재함을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 제 2 선택스 엘리먼트를 상기 슬라이스 헤더에서 인코딩하도록 구성되며, 그리고

상기 프로세서는, 블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 슬라이스 헤더에서의 상기 제 2 선택스 엘리먼트의 인코딩을 제거하도록 구성되는, 비디오 인코딩 디바이스.

청구항 46

제 45 항에 있어서,

상기 화상 계층 파라미터 세트는, 화상 파라미터 세트 (PPS) 또는 적응 파라미터 세트 (APS) 중 하나를 포함하는, 비디오 인코딩 디바이스.

청구항 47

제 45 항에 있어서,

상기 프로세서는, 상기 화상 계층 파라미터 세트 또는 시퀀스 파라미터 세트 (SPS) 중 하나에서의 상기 제 1 선택스 엘리먼트를 인코딩하도록 구성되는, 비디오 인코딩 디바이스.

청구항 48

제 45 항에 있어서,

상기 프로세서는, 블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 중 하나에 존재하는 블록화제거 필터 파라미터들에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하도록 구성되는, 비디오 인코딩 디바이스.

청구항 49

제 45 항에 있어서,

상기 프로세서는, 블록화제거 필터 파라미터들의 상기 제 2 세트에 의한 블록화제거 필터 파라미터들의 상기 제 1 세트의 오버라이드가 가능하게 되는지의 여부를 나타내는 오버라이드 가능 플래그를 상기 화상 계층 파라미터 세트에서 인코딩하도록 구성되는, 비디오 인코딩 디바이스.

청구항 50

제 49 항에 있어서,

상기 프로세서는, 상기 오버라이드가 가능하게될 경우, 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트로부터의 블록화제거 필터 파라미터들의 상기 제 1 세트를 사용할지 또는 블록화제거 필터 파라미터들의 상기 제 1 세트를 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트로 오버라이드할지를 나타내는 오버라이드 플래그를 상기 슬라이스 헤더에서 인코딩하도록 구성되는, 비디오 인코딩 디바이스.

청구항 51

제 49 항에 있어서,

상기 프로세서는, 상기 오버라이드가 가능하지 않을 경우, 상기 슬라이스 헤더에서의 오버라이드 플래그의 인코딩을 제거하고, 상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 상기 제 1 세트에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하도록 구성되는, 비디오 인코딩 디바이스.

청구항 52

제 45 항에 있어서,

상기 프로세서는, 블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 상기 화상 계층 파라미터 세트에 존재하는지의 여부를 나타내는 상속 가능 플래그를 인코딩하도록 구성되는, 비디오 인코딩 디바이스.

청구항 53

제 52 항에 있어서,

상기 프로세서는, 블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 존재하는 경우, 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트를 사용할지 또는 상기 화상 계층 파라미터 세트로부터의 블록화제거 필터 파라미터들의 상기 제 1 세트를 상속할지를 나타내는 상속 플래그를 상기 슬라이스 헤더에서 인코딩하도록 구성되는, 비디오 인코딩 디바이스.

청구항 54

제 52 항에 있어서,

상기 프로세서는, 블록화제거 필터 파라미터들의 상기 제 1 세트가 상기 슬라이스 헤더에 의해 상속되도록 존재하지 않는 경우, 상기 슬라이스 헤더에서의 상속 플래그의 인코딩을 제거하고, 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하도록 구성되는, 비디오 인코딩 디바이스.

청구항 55

비디오 인코딩 디바이스로서,

블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 인코딩하는 수단;

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재함을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 제 2 선택스 엘리먼트를 상기 슬라이스 헤더에서 인코딩하는 수단; 및

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 슬라이스 헤더에서의 상기 제 2 선택스 엘리먼트의 인코딩을 제거하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 56

제 55 항에 있어서,

상기 화상 계층 파라미터 세트는 화상 파라미터 세트 (PPS) 를 포함하는, 비디오 인코딩 디바이스.

청구항 57

제 55 항에 있어서,

상기 제 1 선택스 엘리먼트를 인코딩하는 수단은, 블록화제거 필터 파라미터들의 상기 제 2 세트에 의한 블록화 제거 필터 파라미터들의 상기 제 1 세트의 오버라이드가 가능하게 되는지의 여부를 나타내는 오버라이드 가능 플래그를 상기 화상 계층 파라미터 세트에서 인코딩하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 58

제 57 항에 있어서,

상기 오버라이드가 가능하게될 경우, 상기 제 2 선택스 엘리먼트를 인코딩하는 수단은, 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트로부터의 블록화제거 필터 파라미터들의 상기 제 1 세트를 사용할지 또는 블록화제거 필터 파라미터들의 상기 제 1 세트를 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 상기 제 2 세트로 오버라이드할지를 나타내는 오버라이드 플래그를 상기 슬라이스 헤더에서 인코딩하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 59

제 57 항에 있어서,

상기 오버라이드가 가능하지 않을 경우, 상기 제 2 선택스 엘리먼트의 인코딩을 제거하는 수단은, 상기 슬라이스 헤더에서의 오버라이드 플래그의 인코딩을 제거하는 수단을 포함하며,

상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 상기 제 1 세트에 기초하여 상기 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하는 수단을 더 포함하는, 비디오 인코딩 디바이스.

청구항 60

비디오 데이터를 인코딩하는 명령들을 포함하는 컴퓨터 판독가능 매체로서,

상기 명령들은, 실행되는 경우, 하나 이상의 프로세서들로 하여금,

블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 인코딩하게 하고;

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재함을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 상기 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 상기 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 제 2 선택스 엘리먼트를 상기 슬라이스 헤더에서 인코딩하게 하며; 그리고

블록화제거 필터 파라미터들이 상기 화상 계층 파라미터 세트 및 상기 슬라이스 헤더 양쪽 모두에 존재하지 않음을 상기 제 1 선택스 엘리먼트가 나타내는 경우, 상기 슬라이스 헤더에서의 상기 제 2 선택스 엘리먼트의 인코딩을 제거하게 하는, 컴퓨터 판독가능 매체.

명세서

기술분야

[0001] 본 출원은 2012년 1월 19일자로 출원된 미국 가출원 제61/588,454호, 2012년 1월 31일자로 출원된 미국 가출원 제61/593,015호, 및 2012년 4월 4일자로 출원된 미국 가출원 제61/620,339호를 우선권 주장하며, 그것들의 각각의 전체 내용은 참조로 본원에 통합된다.

[0002] 기술분야

[0003] 본 개시물은 비디오 코딩에 관한 것이고, 더 상세하게는, 비디오 데이터를 블록화제거하는 것에 관한 것이다.

배경 기술

[0004] 디지털 비디오 능력들은 디지털 텔레비전들, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인휴대 정보단말들 (PDA들), 랩톱 또는 데스크톱 컴퓨터들, 태블릿 컴퓨터들, e-북 리더들, 디지털 카메라들, 디지털 레코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 이른바 "스마트 폰들", 비디오 원격회의 디바이스들, 비디오 스트리밍 디바이스들 등을 포함하는 넓은 범위의 디바이스들에 통합될 수 있다. 디지털 비디오 디바이스들은 MPEG-2, MPEG-4, ITU-T H.263 또는 ITU-T H.264/MPEG-4, 파트 10, 고급 비디오 코딩 (AVC) 에 의해 규정된 표준들, 현재 개발중이고 효율 비디오 코딩 (High Efficiency Video Coding, HEVC) 표준, 및 이러한 표준들의 확장들에 기재된 것들과 같은 비디오 압축 기법들을 구현한다. 비디오 디바이스들은 그런 비디오 압축 기법들을 구현하는 것에 의해 디지털 비디오 정보를 더 효율적으로 송신, 수신, 인코딩, 디코딩, 및/또는 저장할 수도 있다.

[0005] 비디오 압축 기법들은 공간적 (화상 내) 예측 및/또는 시간적 (화상 간) 예측을 수행하여 비디오 시퀀스들에 내재하는 리던던시를 감소시키거나 제거한다. 블록 기반 비디오 코딩을 위해, 비디오 슬라이스 (즉, 비디오 프레임 또는 비디오 프레임의 부분) 는 비디오 블록들로 구획될 수도 있으며, 그 비디오 블록들은 또한 트리블록들 (treeblocks), 코딩 단위들 (CU들) 및/또는 코딩 노드들이라고 지칭될 수도 있다. 화상의 인트라 코딩식 (intra-coded; I) 슬라이스에서의 비디오 블록들은 동일한 화상의 이웃 블록들에서의 참조 샘플들에 관한 공간적 예측을 이용하여 인코딩된다. 화상의 인터 코딩식 (inter-coded; P 또는 B) 슬라이스에서의 비디오 블록들은 동일한 화상의 이웃 블록들에서의 참조 샘플들에 관한 공간적 예측 또는 다른 참조 화상들에서의 참조 샘플들에 관한 시간적 예측을 이용할 수도 있다. 화상들은 프레임들이라고 지칭될 수도 있고, 참조 화상들은 참조 프레임들이라고 말해질 수도 있다.

[0006] 공간적 또는 시간적 예측은 코딩될 블록에 대한 예측 블록으로 나타나게 된다. 잔차 데이터는 코딩될 원본 블록과 예측 블록 사이의 화소 차이들을 나타낸다. 인터 코딩식 블록은 예측 블록을 형성하는 참조 샘플들의 블록을 가리키는 모션 벡터와, 코딩된 블록 및 예측 블록 사이의 차이를 나타내는 잔차 데이터에 따라 인코딩된다. 인트라 코딩식 블록은 예측 블록이 생성되는 방법을 정의하는 인트라 코딩 모드 및 잔차 데이터에 따라 인코딩된다. 추가 압축을 위해, 잔차 데이터는 화소 도메인으로부터 변환 도메인으로 변환될 수도 있으며, 결과적으로 잔차 변환 계수들이 생겨나며, 그 계수들은 그 다음에 양자화될 수도 있다. 처음에는 2차원 어레이로 배열된 양자화된 변환 계수들은, 변환 계수들의 1차원 벡터를 생성하기 위하여 스캐닝될 수도 있고, 엔트로피 코딩이 더 많은 압축을 달성하기 위해 적용될 수도 있다.

발명의 내용

과제의 해결 수단

[0007] 대체로, 본 개시물은 비디오 데이터의 현재 슬라이스에 대해 감소된 비트스트림 오버헤드로 블록화제거 필터 파라미터들을 시그널링하는 기법들을 설명한다. 블록화제거 필터 파라미터들은 슬라이스의 디코딩된 비디오 블록들로부터 블록현상 아티팩트들을 제거하는데 사용되는 블록화제거 필터를 정의한다. 블록화제거 필터 파라미터들은, 블록화제거 필터링이 가능하게 되는지 또는 가능하지 않게 되는지를 나타내도록 정의된 선택스 엘리먼트들과, 가능하게 되면, 임계 값들 (t_c 및 β) 에 대한 블록화제거 필터 파라미터 오프셋들을 포함한다.

블록화제거 필터 파라미터들은 화상 계층 파라미터 세트 및 슬라이스 헤더 중 하나 이상에서 코딩될 수도 있다. 화상 계층 파라미터 세트는 화상 파라미터 세트 (PPS) 또는 적응 파라미터 세트 (APS) 중 어느 하나를 포함할 수도 있다.

[0008] 그 기법들은, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 코딩하고, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는 경우, 슬라이스 헤더에서의 제 2 선택스 엘리먼트만을 코딩함으로써, 블록화제거 필터 파라미터들을 시그널링하는데 사용된 비트들의 수를 감소시킬 수도 있다. 제 2 선택스 엘리먼트는, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된다. 이 경우, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 하나에만 존재하는 경우, 비디오 인코더는 슬라이스 헤더에서

의 제 2 선택스 엘리먼트의 인코딩을 제거할 수도 있고, 비디오 디코더는, 제 1 선택스 엘리먼트에 기초하여, 제 2 선택스 엘리먼트가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정할 수도 있다.

[0009] 하나의 예에서, 본 개시물은 비디오 데이터를 디코딩하는 방법을 지향하며, 그 방법은, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 디코딩하는 단계, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재함을 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 디코딩하는 단계, 및 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않음을 제 1 선택스 엘리먼트가 나타내는 경우, 제 2 선택스 엘리먼트가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정하는 단계를 포함한다.

[0010] 다른 예에서, 본 개시물은, 비디오 데이터를 저장하는 메모리, 및 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 디코딩하도록 구성된 프로세서를 포함하는 비디오 디코딩 디바이스를 지향한다. 그 프로세서는, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재함을 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 디코딩하도록 구성된다. 한편, 그 프로세서는, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않음을 제 1 선택스 엘리먼트가 나타내는 경우, 제 2 선택스 엘리먼트가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정하도록 구성된다.

[0011] 추가의 예에서, 본 개시물은, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 디코딩하는 수단, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재함을 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 디코딩하는 수단, 및 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않음을 제 1 선택스 엘리먼트가 나타내는 경우, 제 2 선택스 엘리먼트가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정하는 수단을 포함하는 비디오 디코딩 디바이스를 지향한다.

[0012] 다른 예에서, 본 개시물은, 비디오 데이터를 디코딩하기 위한 명령들을 포함하는 컴퓨터 판독가능 매체를 지향하며, 그 명령들은, 실행되는 경우, 하나 이상의 프로세서들로 하여금, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 디코딩하게 하며, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재함을 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 디코딩하게 하고, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않음을 제 1 선택스 엘리먼트가 나타내는 경우, 제 2 선택스 엘리먼트가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정하게 한다.

[0013] 부가적인 예에서, 본 개시물은, 비디오 데이터를 인코딩하는 방법을 지향하며, 그 방법은, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 인코딩하는 단계, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재함을 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 인코딩하는 단계, 및 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않음을 제 1 선택스 엘리먼트가 나타내는 경우, 슬라이스 헤더에

서의 제 2 선택스 엘리먼트의 인코딩을 제거하는 단계를 포함한다.

[0014] 추가의 예에서, 본 개시물은, 비디오 데이터를 저장하는 메모리, 및 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 인코딩하도록 구성된 프로세서를 포함하는 비디오 인코딩 디바이스를 지향한다. 그 프로세서는, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재함을 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 인코딩하도록 구성된다. 한편, 그 프로세서는, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않음을 제 1 선택스 엘리먼트가 나타내는 경우, 슬라이스 헤더에서의 제 2 선택스 엘리먼트의 인코딩을 제거하도록 구성된다.

[0015] 다른 예에서, 본 개시물은, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 인코딩하는 수단, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재함을 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 인코딩하는 수단, 및 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않음을 제 1 선택스 엘리먼트가 나타내는 경우, 슬라이스 헤더에서의 제 2 선택스 엘리먼트의 인코딩을 제거하는 수단을 포함하는 비디오 인코딩 디바이스를 지향한다.

[0016] 추가의 예에서, 본 개시물은, 비디오 데이터를 디코딩하기 위한 명령들을 포함하는 컴퓨터 판독가능 매체를 지향하며, 그 명령들은, 실행되는 경우, 하나 이상의 프로세서들로 하여금, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 인코딩하게 하며, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재함을 제 1 선택스 엘리먼트가 나타내는 경우, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해, 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 인코딩하게 하고, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않음을 제 1 선택스 엘리먼트가 나타내는 경우, 슬라이스 헤더에서의 제 2 선택스 엘리먼트의 인코딩을 제거하게 한다.

[0017] 하나 이상의 예들의 세부사항들은 첨부 도면들 및 다음의 설명에서 언급된다. 다른 특징들, 목적들, 및 이점들은 상세한 설명 및 도면들로부터, 그리고 청구항들로부터 명확하게 될 것이다.

도면의 간단한 설명

[0018] 도 1은 본 개시물에서 설명되는 기법들에 따라 블록화제거 필터 파라미터들을 코딩할 수도 있는 일 예의 비디오 인코딩 및 디코딩 시스템을 도시하는 블록도이다.

도 2는 감소된 비트스트림 오버헤드로 블록화제거 필터 파라미터들을 인코딩하기 위해 본 개시물에서 설명되는 기법들을 구현할 수도 있는 비디오 인코더의 일 예를 도시하는 블록도이다.

도 3은 비디오 슬라이스들에 적용되는 블록화제거 필터들을 정의하는데 사용된 블록화제거 필터 파라미터들을 디코딩하기 위해 본 개시물에서 설명된 기법들을 구현할 수도 있는 비디오 디코더의 일 예를 도시하는 블록도이다.

도 4는 본 개시물에서 설명된 기법들에 따라 시그널링된 블록화제거 필터 파라미터들에 기초하여 정의된 예시적인 블록화제거 필터의 컴포넌트들을 도시하는 블록도이다.

도 5는 서브블록들 사이의 비디오 블록의 예지 근처의 화소 포지션들을 예시하는 개념도이다.

도 6은 본 개시물에서 설명된 기법들에 따른 현재 비디오 슬라이스에 대해 감소된 비트스트림 오버헤드로 블록화제거 필터 파라미터들을 인코딩하는 일 예의 동작을 도시하는 흐름도이다.

도 7은 본 개시물에서 설명된 기법들에 따른 현재 비디오 슬라이스에 대해 감소된 비트스트림 오버헤드로 블록 화제거 필터 파라미터들을 디코딩하는 일 예의 동작을 도시하는 흐름도이다.

도 8은 슬라이스 헤더에서의 블록화제거 필터 파라미터들에 의해 오버라이드될 수도 있는 화상 파라미터 세트 (PPS) 에서의 현재 비디오 슬라이스에 대한 블록화제거 필터 파라미터들을 인코딩하는 일 예의 동작을 도시하는 흐름도이다.

도 9는 슬라이스 헤더에서의 블록화제거 필터 파라미터들에 의해 오버라이드될 수도 있는 화상 파라미터 세트 (PPS) 에서의 현재 비디오 슬라이스에 대한 블록화제거 필터 파라미터들을 디코딩하는 일 예의 동작을 도시하는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0019] 본 개시물의 일부 예의 기법들은, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 코딩하고, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는 경우, 슬라이스 헤더에서의 제 2 선택스 엘리먼트만을 코딩함으로써, 현재 비디오 슬라이스에 대한 블록화제거 필터 파라미터들을 시그널링하는데 사용된 비트들의 수를 감소시킨다. 제 2 선택스 엘리먼트는, 현재 비디오 슬라이스에 적용되는 블록화제거 필터를 정의하기 위해 화상 계층 파라미터 세트에 포함된 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된다. 이 경우, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 하나에만 존재하는 경우, 비디오 인코딩 디바이스는 슬라이스 헤더에서의 제 2 선택스 엘리먼트의 인코딩을 제거할 수도 있고, 비디오 디코딩 디바이스는, 제 1 선택스 엘리먼트에 기초하여, 제 2 선택스 엘리먼트가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정할 수도 있다.

[0020] 도 1은 본 개시물에서 설명되는 기법들에 따라 블록화제거 필터 파라미터들 코딩할 수도 있는 일 예의 비디오 인코딩 및 디코딩 시스템 (10) 을 도시하는 블록도이다. 도 1에 도시된 바와 같이, 시스템 (10) 은 목적지 디바이스 (14) 에 의해 나중에 디코딩될 인코딩된 비디오 데이터를 생성하는 소스 디바이스 (12) 를 포함한다. 소스 디바이스 (12) 와 목적지 디바이스 (14) 는 데스크톱 컴퓨터들, 노트북 (즉, 랩톱) 컴퓨터들, 태블릿 컴퓨터들, 셋톱 박스들, 이른바 "스마트" 폰들과 같은 전화기 핸드셋들, 이른바 "스마트" 패드들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들, 비디오 스트리밍 디바이스 등을 포함하는 매우 다양한 디바이스들 중 임의의 것을 포함할 수도 있다. 일부 경우들에서, 소스 디바이스 (12) 와 목적지 디바이스 (14) 는 무선 통신을 위해 장비될 수도 있다.

[0021] 목적지 디바이스 (14) 는 디코딩될 인코딩된 비디오 데이터를 링크 (16) 를 통해 수신할 수도 있다. 링크 (16) 는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 인코딩된 비디오 데이터를 이동시킬 수 있는 임의의 유형의 매체 또는 디바이스를 포함할 수도 있다. 하나의 예에서, 링크 (16) 는 소스 디바이스 (12) 가 인코딩된 비디오 데이터를 목적지 디바이스 (14) 로 직접 실시간으로 송신하는 것을 가능하게 하는 통신 매체를 포함할 수도 있다. 인코딩된 비디오 데이터는 통신 표준, 이를테면 무선 통신 프로토콜에 따라 변조되고 목적지 디바이스 (14) 에 송신될 수도 있다. 통신 매체는 임의의 무선 또는 유선 통신 매체, 이를테면 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적 송신 라인들을 포함할 수도 있다. 통신 매체는 패킷 기반 네트워크, 이를테면 로컬 영역 네트워크, 광역 네트워크, 또는 인터넷과 같은 글로벌 네트워크의 부분을 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로의 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0022] 다른 예에서, 링크 (16) 는, 소스 디바이스 (12) 에 의해 생성된 인코딩된 비디오 데이터를 저장할 수도 있고 목적지 디바이스 (14) 가 디스크 액세스 또는 카드 액세스를 통해 원하는 대로 액세스할 수도 있는 저장 매체에 대응할 수도 있다. 그 저장 매체는 블루레이 디스크들, DVD들, CD-ROM들, 플래시 메모리, 또는 인코딩된 비디오 데이터를 저장하는 임의의 다른 적합한 디지털 저장 매체들과 같은 다양한 국소적으로 액세스되는 데이터 저장 매체들 중 임의의 것을 포함할 수도 있다. 추가의 예에서, 링크 (16) 는, 소스 디바이스 (12) 에 의해 생성된 인코딩된 비디오를 보유할 수도 있고 목적지 디바이스 (14) 가 스트리밍 또는 다운로드를 통해 원하는 대로 액세스할 수도 있는 파일 서버 또는 다른 중간 저장 디바이스에 대응할 수도 있다. 그 파일 서버는 인코딩된 비디오 데이터를 저장할 수 있고 그 인코딩된 비디오 데이터를 목적지 디바이스 (14) 로 송신할 수 있는 임의의 유형의 서버일 수도 있다. 예의 파일 서버들은 웹 서버 (예컨대, 웹사이트용), FTP 서버, 네트워크

부속 스토리지 (network attached storage; NAS) 디바이스들, 또는 로컬 디스크 드라이브를 포함한다. 목적지 디바이스 (14) 는 인터넷 접속을 포함하여 임의의 표준 데이터 접속을 통해 인코딩된 비디오 데이터에 액세스할 수도 있다. 이는 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀 등), 또는 파일 서버 상에 저장된 인코딩된 비디오 데이터에 액세스하기에 적합한 양쪽 모두의 조합을 포함할 수도 있다.

파일 서버로부터의 인코딩된 비디오 데이터의 송신은 스트리밍 송신, 다운로드 송신, 또는 양쪽 모두의 조합일 수도 있다.

[0023] 본 개시물의 기법들은 무선 애플리케이션들 또는 설정 (setting) 들로 반드시 제한되지는 않는다. 그 기법들은, 다양한 멀티미디어 애플리케이션들, 이를테면 OTA (over-the-air) 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 예컨대, 인터넷을 통한 스트리밍 비디오 송신들 중 임의의 것의 지원 하의 비디오 코딩, 데이터 저장 매체 상의 저장을 위한 디지털 비디오의 인코딩, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션들에 적용될 수도 있다. 일부 예들에서, 시스템 (10) 은 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 및/또는 화상 통화와 같은 애플리케이션들을 지원하기 위해 단방향 또는 양방향 비디오 송신을 지원하도록 구성될 수도 있다.

[0024] 도 1의 예에서, 소스 디바이스 (12) 는 비디오 소스 (18), 비디오 인코더 (20) 및 출력 인터페이스 (22) 를 구비한다. 일부 경우들에서, 출력 인터페이스 (22) 는 변조기/복조기 (모뎀) 및/또는 송신기를 구비할 수도 있다. 소스 디바이스 (12) 에서, 비디오 소스 (18) 는 비디오 캡처 디바이스, 예컨대, 비디오 카메라, 이전에 캡처된 비디오를 담고 있는 비디오 아카이브, 비디오 콘텐츠 제공자로부터 비디오를 수신하는 비디오 피드 인터페이스, 및/또는 컴퓨터 그래픽 데이터를 소스 비디오로서 생성하는 컴퓨터 그래픽 시스템과 같은 소스, 또는 그런 소스들의 조합을 포함할 수도 있다. 하나의 예로서, 비디오 소스 (18) 가 비디오 카메라이면, 소스 디바이스 (12) 와 목적지 디바이스 (14) 는 이른바 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 그러나, 본 개시물에서 설명된 기법들은 일반적으로 비디오 코딩에 적용가능할 수도 있고, 무선 및/또는 유선 애플리케이션들에 적용될 수도 있다.

[0025] 캡처된, 사전 캡처된 (pre-captured), 또는 컴퓨터 생성된 비디오는 비디오 인코더 (20) 에 의해 인코딩될 수도 있다. 인코딩된 비디오 데이터는 소스 디바이스 (12) 의 출력 인터페이스 (22) 를 통해 목적지 디바이스 (14) 로 직접 송신될 수도 있다. 인코딩된 비디오 데이터는 또한 디코딩 및/또는 플레이백을 위한 목적지 디바이스 (14) 에 의한 나중의 액세스를 위해 저장 매체 또는 파일 서버 상에 저장될 수도 있다.

[0026] 목적지 디바이스 (14) 는 입력 인터페이스 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32) 를 구비한다. 일부 경우들에서, 입력 인터페이스 (28) 는 수신기 및/또는 모뎀을 구비할 수도 있다. 목적지 디바이스 (14) 의 입력 인터페이스 (28) 는 링크 (16) 를 통해 인코딩된 비디오 데이터를 수신한다. 링크 (16) 를 통해 통신되거나 또는 데이터 저장 매체 상에 제공된 인코딩된 비디오 데이터는, 비디오 데이터의 디코딩 시에, 비디오 디코더, 이를테면 비디오 디코더 (30) 에 의한 사용을 위해 비디오 인코더 (20) 에 의해 생성된 다양한 신택스 엘리먼트들을 포함할 수도 있다. 이러한 신택스 엘리먼트들은 통신 매체 상에서 송신되는, 저장 매체 상에 저장되는, 또는 파일 서버에 저장되는 인코딩된 비디오 데이터에 포함될 수도 있다.

[0027] 디스플레이 디바이스 (32) 는 목적지 디바이스 (14) 와 통합되거나, 또는 그것 외부에 있을 수도 있다. 일부 예들에서, 목적지 디바이스 (14) 는 통합형 디스플레이 디바이스를 포함할 수도 있고 또한 외부 디스플레이 디바이스와 인터페이스하도록 구성될 수도 있다. 다른 예들에서, 목적지 디바이스 (14) 는 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32) 는 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 다른 유형의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 것을 포함할 수도 있다.

[0028] 비디오 인코더 (20) 와 비디오 디코더 (30) 는 비디오 압축 표준, 이를테면 현재 개발 중인 고 효율 비디오 코딩 (HEVC) 표준에 따라 동작할 수도 있고, HEVC 테스트 모델 (HEVC Test Model; HM) 을 준수할 수도 있다. 대안으로, 비디오 인코더 (20) 와 비디오 디코더 (30) 는, 다르게는 MPEG-4, 파트 10, 고급 비디오 코딩 (AVC) 이라고 지칭되는 ITU-T H.264 표준과 같은 다른 독점 또는 산업 표준들, 또는 그런 표준들의 확장들에 따라 동작할 수도 있다. 본 개시물의 기법들은, 그러나, 임의의 특정 코딩 표준으로 제한되지 않는다. 비디오 압축 표준들의 다른 예들은 MPEG-2와 ITU-T H.263을 포함한다.

[0029] 비록 도 1에 도시되지 않았지만, 일부 양태들에서, 비디오 인코더 (20) 와 비디오 디코더 (30) 는 각각이 오디오 인코더 및 디코더와 통합될 수도 있고, 적절한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및 소프트웨어를 포함하여, 공통 데이터 스트림 또는 개별 데이터 스트림들에서의 오디오 및 비디오 양쪽 모두의 인코딩을 핸들링할

수도 있다. 적용가능하다면, 일부 예들에서, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜, 또는 사용자 데이터그램 프로토콜 (UDP) 과 같은 다른 프로토콜들을 준수할 수도 있다.

[0030] 비디오 인코더 (20) 와 비디오 디코더 (30) 각각은 다양한 적합한 인코더 회로, 이를테면 하나 이상의 마이크로 프로세서들, 디지털 신호 프로세서들 (DSP들), 주문형 집적회로들 (ASIC들), 필드 프로그램가능 게이트 어레이들 (FPGA들), 개별 로직, 소프트웨어, 하드웨어, 펌웨어 또는 그것들의 임의의 조합 중 임의의 것으로서 구현될 수도 있다. 그 기법들이 소프트웨어에서 부분적으로 구현되는 경우, 디바이스는 본 개시물의 기법들을 수행하기 위해, 적합한 비일시적 컴퓨터 판독가능 매체 내에 소프트웨어에 대한 명령을 저장하고 하나 이상의 프로세서들을 사용하여 하드웨어에서 그 명령들을 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 의 각각은 하나 이상의 인코더들 또는 디코더들 내에 구비될 수도 있고, 그것들 중 어느 하나는 결합형 인코더/디코더 (CODEC) 의 부분으로서 개별 디바이스 내에 통합될 수도 있다.

[0031] JCT-VC (Joint Collaborative Team on Video Coding) 는 HEVC 표준의 개발에 노력하고 있다. HEVC 표준화 노력들은 HEVC 테스트 모델 (HM) 이라고 지칭되는 비디오 코딩 디바이스의 진화 모델에 기초하고 있다. HM 은, 예컨대, ITU-T H.264/AVC에 따른 현존하는 디바이스들에 비해 비디오 코딩 디바이스들의 몇몇 추가적인 능력들을 상정한다. 예를 들어, H.264가 9 개의 인트라 예측 인코딩 모드들을 제공하는 반면, HM은 33 개 정도의 인트라 예측 인코딩 모드들을 제공할 수도 있다.

[0032] 일반적으로, HM의 작업 모델은 비디오 프레임 또는 화상이 루마 및 크로마 샘플들 양쪽 모두를 포함하는 트리블록들 또는 최대 코딩 단위 (largest coding unit; LCU) 들의 시퀀스로 분할될 수도 있다는 것을 설명한다. 트리블록은 H.264 표준의 매크로블록과 유사한 목적을 가진다. 슬라이스는 코딩 순서에서 연속적인 다수의 트리블록들을 포함한다. 비디오 프레임 또는 화상은 하나 이상의 슬라이스들로 구획될 수도 있다. 각각의 트리블록은 쿼드트리에 따라 코딩 단위들 (CU들) 로 분할될 수도 있다. 예를 들어, 쿼드트리의 루트 노트인 트리블록은 4 개의 자식 노드들로 분할될 수도 있고, 각각의 자식 노드는 다시 부모 노드가 될 수도 있고 다른 4 개의 자식 노드들로 분할될 수도 있다. 쿼드트리의 리프 노트인, 최종의 분할되지 않는 자식 노드는 코딩 노드, 즉 코딩된 비디오 블록을 포함한다. 코딩된 비트스트림에 연관된 선택스 데이터는 트리블록이 분할될 수도 있는 최대 횡수를 정의할 수도 있고, 또한 코딩 노드들의 최소 사이즈를 정의할 수도 있다.

[0033] CU는 코딩 노드와 그 코딩 노드에 연관된 예측 단위들 (PU들) 및 변환 단위들 (TU들) 을 포함한다. CU의 사이즈는 코딩 노드의 사이즈에 대응한다. CU의 사이즈는 8x8 화소들로부터 최대 64x64 화소들 또는 그 이상을 갖는 트리블록의 사이즈까지의 범위일 수도 있다. 각각의 CU는 하나 이상의 PU들 및 하나 이상의 TU들을 포함할 수도 있다. CU에 연관된 선택스 데이터는, 예를 들어, 하나 이상의 PU들로의 CU의 구획화를 기술할 수도 있다. 구획화 모드들은 CU가 스킵되는지 또는 직접 모드 인코딩되는지, 인트라 예측 모드 인코딩되는지, 또는 인터 예측 모드 인코딩되는지의 사이에서 상이할 수도 있다. PU들은 형상이 정사각형이 되도록 또는 정사각형이 아니도록 구획될 수도 있다. CU에 연관된 선택스 데이터는 또한, 예를 들어 쿼드트리에 따른 하나 이상의 TU들로의 CU의 구획화를 기술할 수도 있다. PU는 형상이 정사각형이 되도록 또는 정사각형이 아니도록 구획될 수도 있다.

[0034] 일반적으로, PU는 예측 프로세스에 관련된 데이터를 포함한다. 예를 들어, PU가 인트라 모드 인코딩되는 경우, PU는 PU에 대한 인트라 예측 모드를 기술하는 데이터를 포함할 수도 있다. 다른 예로서, PU가 인터 모드 인코딩되는 경우, PU는 PU에 대한 모션 벡터를 정의하는 데이터를 포함할 수도 있다. PU에 대한 모션 벡터를 정의하는 데이터는, 예를 들어, 모션 벡터의 수평 성분, 모션 벡터의 수직 성분, 모션 벡터에 대한 분해능 (예컨대, 1/4 화소 정밀도 또는 1/8 화소 정밀도), 모션 벡터가 가리키는 참조 화상, 및/또는 모션 벡터에 대한 참조 화상 리스트 (예컨대, 리스트 0, 또는 리스트 1) 를 기술할 수도 있다.

[0035] 대체로, TU는 변환 및 양자화 프로세스들을 위해 사용된다. 하나 이상의 PU들을 갖는 CU는 또한 하나 이상의 TU들을 포함할 수도 있다. 예측을 뒤따라, 비디오 인코더 (20) 는 PU에 대응하는 잔차 값들을 계산할 수도 있다. 그 잔차 값들은 엔트로피 코딩을 위한 직렬화된 (serialized) 변환 계수들을 생성하기 위해 TU들을 사용하여 변환 계수들로 변환되며, 양자화되고, 스캐닝될 수도 있는 화소 차이 값들을 포함한다. 본 개시물은 통상 용어 "비디오 블록"을 CU의 코딩 노드를 지칭하기 위해 사용한다. 일부 특정 경우들에서, 본 개시물은 또한 용어 "비디오 블록"을 트리블록, 즉, LCU, 또는 코딩 노드와 PU들 및 TU들을 포함하는 CU를 지칭하기 위해 사용할 수도 있다.

[0036] 비디오 시퀀스는 통상 일련의 비디오 프레임들 또는 화상들을 포함한다. 화상들의 그룹 (GOP) 은 일반적으로 일련의 하나 이상의 비디오 화상들을 포함한다. GOP는 GOP에 포함된 다수의 화상들을 기술하는 선택스

데이터를 GOP의 헤더, 그 화상들 중 하나 이상의 화상들의 헤더, 또는 다른 곳에 포함할 수도 있다. 화상의 각각의 슬라이스는 개별 슬라이스에 대한 인코딩 모드를 기술하는 슬라이스 선택스 데이터를 포함할 수도 있다.

비디오 인코더 (20) 는 비디오 데이터를 인코딩하기 위하여 개개의 비디오 슬라이스들 내의 비디오 블록들에 대해 통상 동작한다. 비디오 블록은 CU 내의 코딩 노드에 대응할 수도 있다. 비디오 블록들은 고정된 또는 가변하는 사이즈들을 가질 수도 있고, 특정된 코딩 표준에 따라 사이즈를 달리할 수도 있다.

[0037] 일 예로서, HM은 다양한 PU 사이즈들에서 예측을 지원한다. 특정 CU의 사이즈가 $2N \times 2N$ 이라고 가정하면, HM은 $2N \times 2N$ 또는 $N \times N$ 의 PU 사이즈들에서의 인트라 예측과, $2N \times 2N$, $2N \times N$, $N \times 2N$, 또는 $N \times N$ 의 대칭적 PU 사이즈들에서의 인터 예측을 지원한다. HM은 또한 $2N \times nU$, $2N \times nD$, $nL \times 2N$, 및 $nR \times 2N$ 의 PU 사이즈들에서의 인터 예측을 위한 비대칭 구획화를 지원한다. 비대칭 구획화 시, CU의 하나의 방향은 구획되지 않는 반면, 다른 방향은 25% 및 75%로 구획된다. 25% 구획에 대응하는 CU의 부분은 "Up", "Down", "Left", 또는 "Right"의 표시가 뒤따르는 "n"에 의해 나타내어진다. 따라서, 예를 들어, " $2N \times nU$ "는 상단의 $2N \times 0.5N$ PU 및 하단의 $2N \times 1.5N$ PU로 수평으로 구획되는 $2N \times 2N$ CU를 의미한다.

[0038] 본 개시물에서, " $N \times N$ " 및 " N 바이 N "는, 수직 및 수평 치수들의 측면에서의 비디오 블록의 화소 치수들, 예컨대, 16×16 화소들 또는 16 바이 16 화소들을 상호교환적으로 말하는데 이용될 수도 있다. 일반적으로, 16×16 블록은 수직 방향의 16 개 화소들 ($y = 16$) 과 수평 방향의 16 개 화소들 ($x = 16$) 을 가질 것이다. 비슷하게, $N \times N$ 블록은 일반적으로 수직 방향의 N 개 화소들 및 수평 방향의 N 개 화소들을 가지며, 여기서 N 은 음이 아닌 정수 값을 나타낸다. 블록에서의 화소들은 행들 및 열들로 배열될 수도 있다. 더구나, 블록들은 수직 방향에서와 동일한 수의 화소들을 수평 방향에서 반드시 가질 필요는 없다. 예를 들어, 블록들은 M 이 N 과 반드시 동일하지는 않은 $N \times M$ 개 화소들을 포함할 수도 있다.

[0039] CU의 PU들을 이용하는 인트라 예측 또는 인터 예측 코딩에 뒤따라, 비디오 인코더 (20) 는 CU의 TU들에 대한 잔차 데이터를 계산할 수도 있다. PU들은 공간 도메인 (또한 화소 도메인이라고 지칭됨) 에서의 화소 데이터를 포함할 수도 있고 TU들은, 잔차 비디오 데이터에 대한 변환, 예컨대, 이산 코사인 변환 (DCT), 정수 변환, 웨이브릿 변환, 또는 개념적으로 유사한 변환의 적용에 뒤따르는 변환 도메인에서의 계수들을 포함할 수도 있다. 잔차 데이터는 인코딩되지 않은 화상의 화소들 및 PU들에 대응하는 예측 값들 사이의 화소 차이들에 대응할 수도 있다. 비디오 인코더 (20) 는 CU에 대한 잔차 데이터를 포함하는 TU들을 형성한 다음 CU에 대한 변환 계수들을 생성하기 위해 TU들을 변환할 수도 있다.

[0040] 변환 계수들을 생성하는 임의의 변환들에 뒤따라, 비디오 인코더 (20) 는 변환 계수들의 양자화를 수행할 수도 있다. 양자화는 변환 계수들이 그 계수들을 나타내는데 사용된 데이터의 양을 가능한 한 줄이도록 양자화되어서, 추가의 압축을 제공하는 프로세스를 일반적으로 말한다. 양자화 프로세스는 그 계수들의 일부 또는 전부에 연관된 비트 깊이를 감소시킬 수도 있다.

[0041] 일부 예들에서, 비디오 인코더 (20) 는 엔트로피 인코딩될 수 있는 직렬화된 벡터를 생성하기 위해 미리 정의된 스캔 순서를 활용하여 양자화된 변환 계수들을 스캔할 수도 있다. 다른 예들에서, 비디오 인코더 (20) 는 적응적 스캔을 수행할 수도 있다. 1차원 벡터를 형성하기 위해 양자화된 변환 계수들을 스캔한 후, 비디오 인코더 (20) 는, 예컨대, 콘텍스트 적응 가변 길이 코딩 (context adaptive variable length coding; CAVLC), 콘텍스트 적응 이진 산술 코딩 (context adaptive binary arithmetic coding; CABAC), 선택스 기반 콘텍스트 적응 이진 산술 코딩 (syntax-based context-adaptive binary arithmetic coding; SBAC), 또는 확률 간격 구획화 엔트로피 코드들 (PIPE) 또는 다른 엔트로피 인코딩 방법론에 따라, 1차원 벡터를 엔트로피 인코딩할 수도 있다. 비디오 인코더 (20) 는 또한 비디오 데이터를 디코딩함에 있어서 비디오 디코더 (30) 에 의한 사용을 위해 인코딩된 비디오 데이터에 연관된 선택스 엘리먼트들을 엔트로피 인코딩할 수도 있다.

[0042] CABAC를 수행하기 위해, 비디오 인코더 (20) 는 콘텍스트 모델 내의 콘텍스트를 송신될 심볼에 할당할 수도 있다. 그 콘텍스트는, 예를 들어, 심볼의 이웃 값들이 영이 아닌지의 여부에 관련될 수도 있다. CAVLC를 수행하기 위해, 비디오 인코더 (20) 는 송신될 심볼에 대한 가변 길이 코드를 선택할 수도 있다. VLC에서의 코드워드들은 상대적으로 더 짧은 코드들이 더 가능성있는 (probable) 심볼들에 대응하는 반면, 더 긴 코드들은 덜 가능성있는 심볼들에 대응하도록 구성될 수도 있다. 이런 식으로, VLC의 사용은, 예를 들어, 송신될 각각의 심볼에 대해 동일 길이 코드워드들을 사용하는 것에 비해, 비트 절약을 달성할 수도 있다. 확률 결정은 심볼에 할당된 콘텍스트에 기초할 수도 있다.

[0043] 인코딩된 비디오 데이터를 비트스트림으로 목적지 디바이스 (14) 의 비디오 디코더 (30) 로 시그널링하는 것에 더하여, 비디오 인코더 (20) 는 또한 인트라 예측 또는 인터 예측 프로세스 동안 그 뒤에 코딩되는 블록들에 대

한 참조 데이터로서 사용하기 위해 인코딩된 비디오 데이터를 디코딩하고 비디오 프레임 또는 화상 내의 블록들을 복원할 수도 있다. 그러나, 비디오 프레임 또는 화상을 블록들 (예컨대, LCU들 및 그것의 서브CU들) 로 분할하며, 그 블록들을 인코딩한 다음, 그 블록들을 디코딩한 후, 블록들 사이의 에지들에서 인지되는 아티팩트들이 발생할 수도 있다. 이들 "블록현상" 아티팩트들을 제거하기 위하여, 비디오 인코더 (20) 는 참조 블록들로서 저장하기 전에 디코딩된 비디오 블록들에 블록화제거 필터들을 적용할 수도 있다. 마찬가지로, 비디오 디코더 (30) 는, 소스 디바이스 (12) 의 비디오 인코더 (20) 로부터 비트스트림으로 수신된 비디오 데이터를 디코딩하고, 비디오 데이터를 디스플레이할 목적뿐만 아니라 그 비디오 데이터를 그 뒤에 디코딩되는 비디오 데이터에 대한 참조 데이터로서 사용할 목적을 위해 동일한 또는 유사한 블록화제거 필터들을 디코딩된 비디오 데이터에 적용하도록 구성될 수도 있다.

[0044] 그 데이터를 참조 데이터로서 사용하기 위해 저장하기 전에 비디오 코딩 디바이스, 이를테면 비디오 인코더 (20) 또는 비디오 디코더 (30) 에 의해 수행된 블록화제거 필터링은, 그 필터링이 코딩 루프 내에서 수행된다는 점에서 "인루프 (in-loop)" 필터링이라고 일반적으로 지칭된다. 비디오 인코더 (20) 및 비디오 디코더 (30) 양쪽 모두를 동일한 블록화제거 기법들을 적용하기 위해 구성함으로써, 비디오 코딩 디바이스들은 동기화될 수 있고, 그래서 블록화제거는 블록화제거된 비디오 데이터를 참조 데이터로서 사용하는 그 뒤에 코딩되는 비디오 데이터에 대한 에러를 도입하지 않는다.

[0045] 비디오 인코더 (20) 와 비디오 디코더 (30) 는, PU 및 TU 에지들을 포함하는 비디오 블록의 각각의 에지에 대해, 에지를 블록화제거하기 위해 블록화제거 필터를 적용할지의 여부를 결정하도록 일반적으로 구성된다. 비디오 코딩 디바이스들은 에지에 수직인 화소들의 하나 이상의 라인들, 예컨대, 8 화소들의 라인의 분석에 기초하여 에지를 블록화제거할지의 여부를 결정하도록 구성될 수도 있다. 따라서, 예를 들어, 수직 에지의 경우, 비디오 코딩 디바이스는 공통선을 따라 에지의 좌측의 4 개의 화소들 및 그 에지 우측의 4 개의 화소들을 검사함으로써 에지를 블록화제거할지의 여부를 결정할 수도 있다. 선택된 화소들의 수는 블록화제거를 위한 최소 블록, 예컨대, 8x8 화소들에 일반적으로 대응한다. 이러한 방식으로, 분석을 위해 사용된 화소들의 라인은, 에지의 양쪽에서, 예컨대, 에지의 좌측 및 우측 또는 에지 상측 및 하측으로 화소들로 비디오 블록의 PU 및 TU 에지들을 가로질러 뻗어 있다. 에지에 대한 블록화제거를 수행할지의 분석을 위해 사용된 화소들의 라인은 또한 지원 화소들의 세트, 또는 간단히 "지원"으로서 지칭된다.

[0046] 비디오 코딩 디바이스들은 특정 에지에 대한 지원에 기초하여 블록화제거 결정 기능들을 실행하도록 구성될 수도 있다. 대체로, 블록화제거 결정 기능부들은 지원 화소들 내의 고 빈도의 변화들을 검출하도록 구성된다. 보통, 고 빈도의 변화가 검출되는 경우, 블록화제거 결정 기능부는, 인지가 가능한 아티팩트들이 에지에 존재하고 블록화제거가 발생해야 한다는 표시를 제공한다. 블록화제거 결정 기능부들은 또한 지원에 기초하여 에지에 적용할 블록화제거 필터의 유형 및 강도를 결정하도록 구성될 수도 있다. 블록화제거 필터의 유형 및 강도는 임계 값들 (t_c 및 β) 에 의해 나타내어질 수도 있다.

[0047] 본 개시물은 비디오 데이터의 현재 슬라이스에 대해 감소된 비트스트림 오버헤드로 블록화제거 필터 파라미터들을 시그널링하는 기법들을 설명한다. 블록화제거 필터 파라미터들은, 현재 슬라이스의 디코딩된 비디오 블록들로부터 블록현상 아티팩트들을 감소시키거나 또는 제거하는데 사용되는 블록화제거 필터를 정의한다. 블록화제거 필터 파라미터들은, 블록화제거 필터링이 가능하게 되는지 또는 가능하지 않게 되는지를 나타내도록 구성된 선택스 엘리먼트들과, 가능하게 되면, 임계 값들 (t_c 및 β) 에 대한 블록화제거 필터 파라미터 오프셋들을 포함한다.

[0048] 블록화제거 필터 파라미터들은 화상 계층 파라미터 세트 및 슬라이스 헤더 중 하나 이상에서 코딩될 수도 있다. 화상 계층 파라미터 세트는 화상 파라미터 세트 (PPS) 또는 적응 파라미터 세트 (APS) 중 어느 하나를 포함할 수도 있다. PPS는 PPS를 참조하는 화상들 사이에서 변화할 가능성이 없는 데이터를 포함하는 화상 계층 파라미터 세트이다. APS는 화상에서 화상으로 변화할 가능성이 있는 화상 적응 데이터와 함께 사용하기 위해 의도된 화상 계층 파라미터 세트이다. 하나의 예에서, APS는 블록화제거 필터, 적응 루프 필터 (Adaptive Loop Filter; ALF), 및 샘플 적응 오프셋 (Sample Adaptive Offset; SAO) 에 대한 파라미터들을 포함한다. PPS 대신에 APS에 이들 파라미터들을 포함시키는 것은 비디오 시퀀스를 위해 송신되는 비트들의 수를 감소시킬 수도 있는데, 블록화제거 필터, ALF, 또는 SAO 파라미터들이 변하는 경우 일정한 PPS 데이터가 반복될 필요가 없기 때문이다.

[0049] 도 2는 감소된 비트스트림 오버헤드로 블록화제거 필터 파라미터들을 인코딩하기 위해 본 개시물에서 설명되는 기법들을 구현할 수도 있는 일 예의 비디오 인코더 (20) 를 도시하는 블록도이다. 비디오 인코더 (20) 는

비디오 슬라이스들 내의 비디오 블록들의 인트라 코딩 및 인터 코딩을 수행할 수도 있다. 인트라 코딩은 공간적 예측에 의존하여, 주어진 비디오 프레임 또는 화상 내의 비디오에서의 공간적 리던던시를 감소시키거나 또는 제거한다. 인터 코딩은 시간적 예측에 의존하여, 비디오 시퀀스의 인접한 프레임들 또는 화상들 내의 비디오에서의 시간적 리던던시를 감소시키거나 또는 제거한다. 인트라 모드 (I 모드) 는 여러 공간 기반 압축 모드들 중 임의의 것을 말할 수도 있다. 단방향 예측 (P 모드) 및 양방향 예측 (B 모드) 과 같은 인터 모드들은 여러 시간 기반 압축 모드들 중 임의의 것을 말할 수도 있다.

[0050] 도 2의 예에서, 비디오 인코더 (20) 는 모드 선택 유닛 (40), 모션 추정 유닛 (42), 모션 보상 유닛 (44), 인트라 예측 프로세싱 유닛 (46), 참조 화상 메모리 (64), 합산기 (50), 변환 프로세싱 유닛 (52), 양자화 유닛 (54), 및 엔트로피 인코딩 유닛 (56) 을 구비한다. 비디오 블록 복원을 위해, 비디오 인코더 (20) 는 또한 역 양자화 유닛 (58), 역 변환 프로세싱 유닛 (60), 및 합산기 (62) 를 구비한다. 블록화제거 (deblocking) 필터 (63) 가 또한 블록 경계들을 필터링하여, 복원된 비디오 블록들로부터 블록현상 아티팩트들을 제거하기 위해 포함된다.

[0051] 도 2에 도시된 바와 같이, 비디오 인코더 (20) 는 인코딩될 비디오 슬라이스 내의 현재 비디오 블록을 수신한다. 슬라이스는 다수의 비디오 블록들로 분할될 수도 있다. 모드 선택 유닛 (40) 은, 현재 비디오 블록에 대해, 여러 결과들에 기초하여 코딩 모드들 중 하나, 즉, 인트라 또는 인터를 선택할 수도 있다. 인트라 또는 인터 모드들이 선택되면, 모드 선택 유닛 (40) 은 결과적인 인트라 코딩된 또는 인터 코딩된 블록을, 잔차 블록 데이터를 생성하는 합산기 (50) 로, 그리고 참조 화상 메모리 (64) 에 저장된 참조 화상 내의 참조 블록으로서 사용하기 위해 인코딩된 블록을 복원하는 합산기 (62) 로 제공한다. 인트라 예측 프로세싱 유닛 (46) 은 공간적 압축을 제공하기 위해 코딩된 현재 블록과 동일한 프레임 또는 슬라이스에서의 하나 이상의 이웃하는 블록들에 대해, 현재 비디오 블록의 인트라 예측 코딩을 수행한다. 모션 추정 유닛 (42) 과 모션 보상 유닛 (44) 은 시간적 압축을 제공하기 위해 하나 이상의 참조 화상들에서 하나 이상의 예측 블록들에 관하여 현재 비디오 블록의 인터 예측 코딩을 수행한다.

[0052] 인터 코딩의 경우, 모션 추정 유닛 (42) 은 비디오 시퀀스에 대한 미리 결정된 패턴에 따라 비디오 슬라이스에 대한 인터 예측 모드를 결정하도록 구성될 수도 있다. 미리 결정된 패턴은 시퀀스에서의 비디오 슬라이스들을 P 슬라이스들 또는 B 슬라이스들이라고 지정할 수도 있다. 모션 추정 유닛 (42) 과 모션 보상 유닛 (44) 은 고도로 통합될 수도 있지만 개념적 목적들을 위해 별개로 예시된다. 모션 추정 유닛 (42) 에 의해 수행된 모션 추정은 비디오 블록들에 대한 모션을 추정하는 모션 벡터들을 생성하는 프로세스이다. 모션 벡터는, 예를 들어, 참조 화상 내의 예측 블록에 대한 현재 비디오 프레임 또는 화상 내의 비디오 블록의 PU의 변위 (displacement) 를 나타낼 수도 있다.

[0053] 예측 블록은 차의 절대값 합 (SAD), 차의 제곱 합 (SSD), 또는 다른 차이 메트릭들에 의해 결정될 수도 있는, 화소 차이의 관점에서 코딩될 비디오 블록의 PU에 밀접하게 매칭된다고 생각되는 블록이다. 일부 예들에서, 비디오 인코더 (20) 는 참조 화상 메모리 (64) 에 저장된 참조 화상들의 부 정수 (sub-integer) 화소 위치들에 대한 값들을 계산할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 참조 화상의 1/4 화소 위치들, 1/8 화소 위치들, 또는 다른 분수 화소 위치들의 값들을 계산할 수도 있다. 그러므로, 모션 추정 유닛 (42) 은 풀 (full) 화소 위치들 및 분수 화소 위치들에 대한 모션 검색을 수행하고 분수 화소 정밀도를 갖는 모션 벡터를 출력할 수도 있다.

[0054] 모션 추정 유닛 (42) 은 PU의 위치와 참조 화상의 예측 블록의 위치를 비교함으로써 인터 코딩된 슬라이스에서의 비디오 블록의 PU에 대한 모션 벡터를 계산한다. 참조 화상은 참조 화상 메모리 (64) 에 저장된 하나 이상의 참조 화상들을 각각 식별하는 제 1 참조 화상 리스트 (리스트 0) 또는 제 2 참조 화상 리스트 (리스트 1) 로부터 선택될 수도 있다. 모션 추정 유닛 (42) 은 계산된 모션 벡터를 엔트로피 인코딩 유닛 (56) 및 모션 보상 유닛 (44) 에 전송한다.

[0055] 모션 보상 유닛 (44) 에 의해 수행되는 모션 보상은 모션 추정에 의해 결정된 모션 벡터에 기초하여 예측 블록을 패치하는 것 또는 생성하는 것을 수반할 수도 있다. 현재 비디오 블록의 PU에 대한 모션 벡터를 수신 시, 모션 보상 유닛 (44) 은 참조 화상 리스트들 중 하나에서 모션 벡터가 가리키는 예측 블록을 찾을 수도 있다. 비디오 인코더 (20) 는, 코딩되고 있는 현재 비디오 블록의 화소 값들로부터 예측 블록의 화소 값들을 감산하여 화소 차이 값들을 형성함으로써 잔차 비디오 블록을 형성한다. 화소 차이 값들은 블록에 대한 잔차 데이터를 형성하고, 루마 및 크로마 차이 성분들 양쪽 모두를 포함할 수도 있다. 합산기 (50) 는 이 감산 동작을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 모션 보상 유닛 (44) 은 또한 비디오 슬라이스

의 비디오 블록들의 디코딩 시에 비디오 디코더 (30) 에 의한 사용을 위해 비디오 블록들 및 비디오 슬라이스에 연관된 신택스 엘리먼트들을 생성할 수도 있다.

[0056] 모션 보상 유닛 (44) 이 현재 비디오 블록에 대한 예측 블록을 생성한 후, 비디오 인코더 (20) 는 현재 비디오 블록으로부터 예측 블록을 감산함으로써 잔차 비디오 블록을 형성한다. 잔차 블록에서의 잔차 비디오 데이터는, 하나 이상의 TU들에 포함되고 변환 프로세싱 유닛 (52) 에 인가될 수도 있다. 변환 프로세싱 유닛 (52) 은 변환, 이를테면 이산 코사인 변환 (DCT) 또는 개념적으로 유사한 변환을 사용하여 잔차 비디오 데이터를 잔차 변환 계수들로 변환시킨다. 변환 프로세싱 유닛 (52) 은 잔차 비디오 데이터를 화소 도메인으로부터 변환 도메인, 이를테면 주파수 도메인으로 변환할 수도 있다.

[0057] 변환 프로세싱 유닛 (52) 은 결과적인 변환 계수들을 양자화 유닛 (54) 에 전송할 수도 있다. 양자화 유닛 (54) 은 변환 계수들을 양자화하여 비트 레이트를 더욱 감소시킨다. 양자화 프로세스는 그 계수들의 일부 또는 전부에 연관된 비트 깊이를 감소시킬 수도 있다. 양자화 정도는 양자화 파라미터를 조정함으로써 수정될 수도 있다. 일부 예들에서, 양자화 유닛 (54) 은 그 후 양자화된 변환 계수들을 포함하는 매트릭스의 스캔을 수행할 수도 있다. 대안으로, 엔트로피 인코딩 유닛 (56) 이 그 스캔을 수행할 수도 있다.

[0058] 양자화를 뒤따라, 엔트로피 인코딩 유닛 (56) 은 양자화된 변환 계수들을 엔트로피 인코딩한다. 예를 들어, 엔트로피 인코딩 유닛 (56) 은 콘텍스트 적응 가변 길이 코딩 (CAVLC), 콘텍스트 적응 이진 산술 코딩 (CABAC), 또는 다른 엔트로피 인코딩 기법을 수행할 수도 있다. 엔트로피 코딩 유닛 (56) 에 의한 엔트로피 코딩에 뒤이어, 인코딩된 비트스트림은 비디오 디코더 (30) 로 송신되거나 또는 나중의 송신 또는 비디오 디코더 (30) 에 의한 취출을 위해 보관될 수도 있다. 엔트로피 인코딩 유닛 (56) 은 또한 코딩되고 있는 현재 비디오 슬라이스에 대한 모션 벡터들 및 다른 신택스 엘리먼트들을 엔트로피 인코딩할 수도 있다.

[0059] 역 양자화 유닛 (58) 과 역 변환 프로세싱 유닛 (60) 은 역 양자화 및 역 변환을 각각 적용하여, 참조 화상의 참조 블록으로서 나중에 사용하기 위해 화소 도메인에서 잔차 블록을 복원한다. 모션 보상 유닛 (44) 은 참조 화상 리스트들 중 하나 내의 참조 화상들 중 하나의 참조 화상의 예측 블록에 잔차 블록을 가산함으로써 참조 블록을 계산할 수도 있다. 모션 보상 유닛 (44) 은 또한 하나 이상의 보간 필터들을 복원된 잔차 블록에 적용하여 모션 추정에서 사용하기 위한 부 정수 화소 값들을 계산할 수도 있다. 합산기 (62) 는 복원된 잔차 블록을 모션 보상 유닛 (44) 에 의해 생성된 모션 보상된 예측 블록에 가산하여, 참조 화상 메모리 (64) 에 저장하기 위한 참조 화상의 참조 블록을 생성한다. 참조 블록은 블록현상 아티팩트들을 제거하기 위하여 블록화제거 필터 (63) 에 의해 필터링된다. 참조 블록은 그 다음에 참조 화상 메모리 (64) 에 저장된다. 참조 블록은 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 후속 비디오 프레임 또는 화상에서 블록을 인터 예측하기 위한 참조 블록으로서 사용될 수도 있다.

[0060] 본 개시물의 기법들에 따라, 비디오 인코더 (20) 는 합산기 (62) 의 출력을 선택적으로 필터링하는 블록화제거 필터 (63) 를 포함한다. 특히, 블록화제거 필터 (63) 는, 역 양자화된 및 역 변환된 잔차 데이터에 가산된, 모션 보상 유닛 (44) 또는 인트라 예측 유닛 (46) 중 어느 하나로부터 수신된 예측 데이터에 대응하는 합산기 (62) 로부터의 복원된 비디오 데이터를 수신한다. 이러한 방식으로, 블록화제거 필터 (63) 는 비디오 데이터의 디코딩된 블록들, 예컨대, 슬라이스 또는 화상의 LCU 및/또는 LCU들의 CU들에 대응하는 디코딩된 블록들을 수신하고, 그 블록들을 선택적으로 필터링하여 블록현상 아티팩트들을 제거한다.

[0061] 비디오 인코더 (20) 에서의 블록화제거 필터 (63) 는 경계 강도 컴퓨테이션 및 블록화제거 결정들의 결과에 기초하여 디코딩된 비디오 블록의 특정한 TU 및 PU 에지들을 필터링한다. 블록화제거 필터 (63) 는 블록의 주어진 에지 근처의 비디오 블록의 화소들을 그 에지를 블록화제거할지의 여부 및 그 에지를 블록화제거하는 방법을 결정하기 위해 분석하도록 일반적으로 구성된다. 더 상세하게는, 블록화제거 결정들은, 블록화제거 필터가 온인지 또는 오프인지, 블록화제거 필터가 약한지 또는 강한지, 및 주어진 비디오 블록에 대한 약한 필터의 강도를 포함할 수도 있다. 블록화제거 필터 (63) 는 주어진 에지 근처의 화소들의 값들을, 그 값들에서 고빈도의 변화가 검출되는 경우 변경하여 에지에서 인지가 가능한 블록현상 아티팩트들을 제거할 수도 있다.

[0062] 경계 강도 컴퓨테이션 및 블록화제거 결정들은 임계 값들 (t_c 및 β) 에 의존한다. 블록화제거 필터의 임계 값들 (t_c 및 β) 은 파라미터 Q에 의존하는데, 그 파라미터 Q는 현재 비디오 블록에 대한 양자화 파라미터 (QP) 값 및 경계 강도 (B_s) 로부터 다음과 같이 도출된다:

[0063] $B_s = 2$ 이면, $TcOffset = 2$

- [0064] $B_s \leq 1$ 이면, $TcOffset = 0$
- [0065] t_c 의 경우: $Q = Clip3(0, MAX_QP + 2, QP + TcOffset)$; $MAX_QP = 51$
- [0066] β 의 경우: $Q = Clip3(0, MAX_QP, QP)$
- [0067] $Clip3(th1, th2, value) = \min(th1, \max(th2, value))$
- [0068] 임계 값들 (t_c 및 β)은 비디오 블록의 QP 값으로부터 도출된 파라미터 Q에 기초하여 액세스가능한 테이블에 저장될 수도 있다. 블록화제거 프로세스는 도 4에 예시된 블록화제거 필터 (100)에 관해 아래에서 더 상세히 설명된다.
- [0069] 본 개시물은, 감소된 비트스트림 오버헤드로, 비디오 데이터의 현재 슬라이스에 대한 블록화제거 필터 (63)를 정의하는데 사용되는 블록화제거 필터 파라미터들을 시그널링하는 기법들을 설명한다. 비디오 인코더 (20)는 블록화제거 필터 (63)를 정의하는 블록화제거 필터 파라미터들을 결정한 다음 블록화제거 필터 파라미터들을 시그널링하여서, 비디오 디코더 (30)는 동일한 또는 유사한 블록화제거 필터를 디코딩된 비디오 블록들에 적용할 수 있다. 블록화제거 필터 파라미터들은, 블록화제거 필터링이 가능하게 되는지 또는 가능하지 않게 되는지를 나타내도록 정의된 선택스 엘리먼트들과, 가능하게 되면, 임계 값들 (t_c 및 β)에 대한 블록화제거 필터 파라미터 오프셋들을 포함한다.
- [0070] 블록화제거 필터 파라미터들은 비디오 디코더 (30)로의 시그널링을 위해 화상 계층 파라미터 세트 및 슬라이스 헤더 중 하나 이상에서 코딩될 수도 있다. 화상 계층 파라미터 세트는 화상 파라미터 세트 (PPS) 또는 적응 파라미터 세트 (APS) 중 어느 하나를 포함할 수도 있다. PPS는 PPS를 참조하는 화상들 사이에서 변화할 가능성이 없는 데이터를 포함하는 화상 계층 파라미터 세트이다. APS는 화상에서 화상으로 변화할 가능성이 있는 화상 적응 데이터와 함께 사용하기 위해 의도된 화상 계층 파라미터 세트이다.
- [0071] 비디오 인코더 (20)의 엔트로피 인코딩 유닛 (56)은 화상 계층 파라미터 세트를 참조하는 화상들에 대해 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 인코딩한다. 본 발명에서 설명되는 기법들에 따르면, 엔트로피 인코딩 유닛 (56)은, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는 경우, 슬라이스 헤더에서의 제 2 선택스 엘리먼트만을 인코딩함으로써 현재 비디오 슬라이스에 대해 감소된 비트스트림 오버헤드로 블록화제거 필터 파라미터들을 인코딩한다.
- [0072] 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않는 경우, 엔트로피 인코딩 유닛 (56)은, 블록화제거 필터 파라미터들 중 어떤 세트가 현재 비디오 슬라이스에 대한 블록화제거 필터 (63)를 정의하는데 사용되는지를 나타내도록 정의된 슬라이스 헤더에서의 제 2 선택스 엘리먼트의 코딩을 제거한다. 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 하나에만 존재하는 경우에, 블록화제거 필터 (63)는 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 어느 하나에 존재하는 블록화제거 필터 파라미터들의 세트에 기초하여 현재 비디오 슬라이스에 대해 정의된다. 그러므로, 제 2 선택스 엘리먼트는, 블록화제거 필터 파라미터들을 비디오 디코더 (30)에 나타내는데 있어 불필요한데, 블록화제거 필터 파라미터들 중 어떤 세트를 비디오 디코더 (30)에서 블록화제거 필터를 정의하는데 사용할지에 관해 화상 계층 파라미터 세트 및 슬라이스 헤더 사이에서 결정이 이루어질 필요가 없기 때문이다.
- [0073] 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는 경우, 엔트로피 인코딩 유닛 (56)은 화상 계층 파라미터 세트에 포함된 블록화제거 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 제 2 선택스 엘리먼트를 슬라이스 헤더에서 인코딩한다. 이 경우, 블록화제거 필터 (63)는 블록화제거 파라미터들의 제 1 세트 또는 제 2 세트 중 하나에 기초하여 현재 비디오 슬라이스에 대해 정의된다. 그러므로, 제 2 선택스 엘리먼트는 비디오 인코더 (20)에서 블록화제거 필터 (63)를 정의하는데 사용된 블록화제거 필터 파라미터들을 나타내는데 필요하여서, 비디오 디코더 (30)는 동일한 또는 유사한 블록화제거 필터를 디코딩된 비디오 블록들에 적용할 수 있다.
- [0074] 일부 경우들에서, 엔트로피 인코딩 유닛 (56)은 또한 임의의 블록화제거 필터 제어 선택스 엘리먼트들이 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 어느 하나에 존재하는지의 여부를 나타내도록 정의된 제어 존재 선택스 엘리먼트를 인코딩한다. 제어 존재 선택스 엘리먼트는 화상 계층 파라미터 세트에서 또는 높은 계층 파라미터 세트, 예컨대, 시퀀스 파라미터 세트 (SPS)로부터 시그널링될 수도 있다. 블록화제거 필터 제어

신택스 엘리먼트들은 위에서 설명된 제 1 및 제 2 신택스 엘리먼트들을 포함한다. 그러므로, 엔트로피 인코딩 유닛 (56) 은, 제 1 신택스 엘리먼트를 인코딩하기 전에 제어 존재 신택스 엘리먼트를 인코딩한다. 블록 화제거 필터 제어 신택스 엘리먼트들이 존재하지 않으면, 비디오 인코더 (20) 는 비디오 디코더 (30) 에 통지하고 제 1 또는 제 2 신택스 엘리먼트들을 인코딩하지 않는다. 이 경우, 비디오 인코더 (20) 는 디코딩된 비디오 블록들에 적용되는 블록화제거 필터 (63) 를 정의하기 위해 디폴트 블록화제거 필터 파라미터들을 사용할 수도 있다.

[0075] 다른 경우들에서, 엔트로피 인코딩 유닛 (56) 은 제 1 신택스 엘리먼트를 인코딩하기 전에 블록화제거 필터 (63) 가 비디오 시퀀스의 하나 이상의 화상들에 대해 가능하게 되는지의 여부를 나타내도록 정의된 블록화제거 필터 가능 신택스 엘리먼트를 인코딩할 수도 있다. 블록화제거 필터 가능 신택스 엘리먼트는 높은 계층 파라미터 세트, 예컨대, 시퀀스 파라미터 세트 (SPS) 에서 시그널링될 수도 있다. 블록화제거 필터 (63) 가 비디오 시퀀스에 대해 가능하지 않게 되면, 비디오 인코더 (20) 는 비디오 디코더 (30) 에 통지하고 제 1 또는 제 2 신택스 엘리먼트들을 인코딩하지 않는데, 블록화제거 필터 (63) 가 디코딩된 비디오 블록들에 적용되지 않기 때문이다. 이 경우, 비디오 인코더 (20) 는 또한 제어 존재 신택스 엘리먼트를 인코딩하지 않는다.

[0076] 하나의 예에서, 제 1 신택스 엘리먼트는 주어진 화상에 대해 PPS에서 코딩된 오버라이드 가능 플래그를 포함한다. 이 경우, 블록화제거 필터 파라미터들의 제 1 세트가 PPS에서 코딩되고 오버라이드 가능 플래그는, 블록화제거 필터 파라미터들의 제 2 세트가 PPS로부터의 파라미터들을 오버라이드하는데 사용될 수 있는 주어진 화상의 하나 이상의 슬라이스들에 대한 슬라이스 헤더에 존재하는지의 여부를 나타낸다. 덧붙여서, 제 2 신택스 엘리먼트는 슬라이스 헤더에 코딩될 수도 있는 오버라이드 플래그를 포함한다. 블록화제거 필터 파라미터들의 제 2 세트가 슬라이스 헤더에 존재함을 PPS에서의 오버라이드 가능 플래그가 나타내는 경우, 엔트로피 인코딩 유닛 (56) 은 비디오 디코더 (30) 에서 블록화제거 필터를 정의하기 위해 PPS에서의 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 블록화제거 필터 파라미터들의 제 1 세트를 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트로 오버라이드할지를 비디오 디코더 (30) 에 나타내는 오버라이드 플래그를 인코딩한다. 그렇지 않고, PPS에서의 블록화제거 필터 파라미터들의 제 1 세트만이 존재함을 PPS에서의 오버라이드 가능 플래그가 나타내는 경우, 엔트로피 인코딩 유닛 (56) 은 슬라이스 헤더에서의 오버라이드 플래그의 인코딩을 제거한다. 이 예에 대한 특정 신택스 엘리먼트들은 도 3에서 비디오 디코더 (30) 에 관해 아래에서 더 상세히 설명된다.

[0077] 다른 예에서, 제 1 신택스 엘리먼트는 주어진 화상에 대해 SPS 및/또는 APS에 코딩된 상속 가능 플래그를 포함한다. 이 경우, 블록화제거 필터 파라미터들의 제 2 세트는 슬라이스 헤더에서 코딩되고 상속 가능 플래그는 블록화제거 필터 파라미터들의 제 1 세트가 슬라이스 헤더에 의해 상속될 수 있는 APS에 존재하는지의 여부를 나타낸다. 제 2 신택스 엘리먼트는 슬라이스 헤더에 코딩될 수도 있는 상속 플래그를 포함한다. 블록화제거 필터 파라미터들의 제 1 세트가 APS에 존재함을 SPS 및/또는 APS에서의 상속 가능 플래그가 나타내면, 엔트로피 인코딩 유닛 (56) 은 비디오 디코더 (30) 에서 블록화제거 필터를 정의하기 위해 슬라이스 헤더에서의 블록화제거 필터 파라미터들의 제 2 세트를 사용할지 또는 APS에서의 블록화제거 필터 파라미터들의 제 1 세트를 상속할지를 비디오 디코더 (30) 에 나타내는 상속 플래그를 인코딩한다. 그렇지 않고, 슬라이스 헤더에서의 블록화제거 필터 파라미터들의 제 2 세트만이 존재함을 SPS 및/또는 APS에서의 상속 가능 플래그가 나타내는 경우, 엔트로피 인코딩 유닛 (56) 은 슬라이스 헤더에서 상속 플래그의 인코딩을 제거한다. 이 예에 대한 특정 신택스 엘리먼트들은 도 3에서 비디오 디코더 (30) 에 관해 아래에서 더 상세히 설명된다.

[0078] 도 3은 비디오 슬라이스들에 적용되는 블록화제거 필터들을 정의하는데 사용된 블록화제거 필터 파라미터들을 디코딩하기 위해 본 개시물에서 설명된 기법들을 구현할 수도 있는 비디오 디코더 (30) 의 일 예를 도시하는 블록도이다. 도 3의 예에서, 비디오 디코더 (30) 는 엔트로피 디코딩 유닛 (80), 예측 프로세싱 유닛 (81), 역 양자화 유닛 (86), 역 변환 프로세싱 유닛 (88), 합산기 (90), 블록화제거 필터 (91), 및 참조 화상 메모리 (92) 를 구비한다. 예측 프로세싱 유닛 (81) 은 모션 보상 유닛 (82) 과 인트라 예측 프로세싱 유닛 (84) 을 구비한다. 비디오 디코더 (30) 는, 일부 예들에서, 도 2로부터 비디오 인코더 (20) 에 관해 설명된 인코딩 과정 (pass) 에 일반적으로 역인 디코딩 과정을 수행할 수도 있다.

[0079] 디코딩 프로세스 동안, 비디오 디코더 (30) 는 인코딩된 비디오 슬라이스의 비디오 블록들 및 연관된 신택스 엘리먼트들을 나타내는 인코딩된 비디오 비트스트림을 비디오 인코더 (20) 로부터 수신한다. 비트스트림에서 표현되는 비디오 블록들이 압축된 비디오 데이터를 포함하는 경우, 비디오 디코더 (30) 의 엔트로피 디코딩 유닛 (80) 은 그 비트스트림을 엔트로피 디코딩하여 양자화된 계수들, 모션 벡터들, 및 다른 신택스 엘리먼트들을 생성한다. 엔트로피 디코딩 유닛 (80) 은 모션 벡터들 및 다른 신택스 엘리먼트들을 예측 프로세싱 유닛

(81)에 포워딩한다. 비디오 디코더 (30)는 시퀀스 레벨, 화상 레벨, 슬라이스 레벨 및/또는 비디오 블록 레벨에서 선택스 엘리먼트들을 수신할 수도 있다. 일부 경우들에서, 엔트로피 디코딩 유닛 (80)은 주어진 비디오 슬라이스에 대한 블록화제거 필터 (91)를 정의하기 위해, 블록화제거 필터 파라미터들을 포함하는 블록화제거 필터 제어 선택스 엘리먼트들을 디코딩한다.

[0080] 비디오 슬라이스가 인트라 코딩된 (I) 슬라이스로서 코딩되는 경우, 예측 프로세싱 유닛 (81)의 인트라 예측 프로세싱 유닛 (84)은 현재 비디오 슬라이스의 비디오 블록을 위한 예측 데이터를 현재 프레임 또는 화상의 이전에 디코딩된 블록들로부터의 시그널링된 인트라 예측 모드 및 데이터에 기초하여 생성할 수도 있다. 비디오 프레임이 인터 코딩된 (즉, B 또는 P) 슬라이스로서 코딩되는 경우, 예측 프로세싱 유닛 (81)의 모션 보상 유닛 (82)은 엔트로피 디코딩 유닛 (80)으로부터 수신된 모션 벡터들 및 다른 선택스 엘리먼트들에 기초하여, 현재 비디오 슬라이스의 비디오 블록에 대한 예측 블록을 생성한다. 그 예측 블록들은 참조 화상 리스트들 중 하나 내의 참조 화상들 중 하나로부터 생성될 수도 있다. 비디오 디코더 (30)는 참조 화상 메모리 (92)에 저장된 참조 화상들에 기초하여 디폴트 구성 기법들을 이용하여, 참조 프레임 리스트들, 리스트 0 및 리스트 1을 구성할 수도 있다.

[0081] 모션 보상 유닛 (82)은 모션 벡터들 및 다른 선택스 엘리먼트들을 파싱함으로써 현재 비디오 슬라이스의 비디오 블록에 대한 예측 정보를 결정하고, 그 예측 정보를 이용하여 디코딩되고 있는 현재 비디오 블록에 대한 예측 블록들을 생성한다. 예를 들어, 모션 보상 유닛 (82)은 수신된 선택스 엘리먼트들의 일부를 이용하여, 비디오 슬라이스의 비디오 블록들을 코딩하는데 사용된 예측 모드 (예컨대, 인트라 또는 인터 예측), 인터 예측 슬라이스 유형 (예컨대, B 슬라이스 또는 P 슬라이스), 슬라이스에 대한 참조 화상 리스트들 중 하나 이상에 대한 구축 정보, 슬라이스의 각각의 인터 인코딩된 비디오 블록에 대한 모션 벡터들, 슬라이스의 각각의 인터 코딩된 비디오 블록에 대한 인터 예측 스테이터스, 및 현재 비디오 슬라이스 내의 비디오 블록들을 디코딩하기 위한 다른 정보를 결정한다.

[0082] 모션 보상 유닛 (82)은 또한 보간 필터들에 기초하여 보간을 수행할 수도 있다. 모션 보상 유닛 (82)은 비디오 블록들의 인코딩 동안에 비디오 인코더 (20)에 의해 사용된 것과 같은 보간 필터들을 사용하여 참조 블록들의 부 정수 화소들에 대한 보간된 값들을 계산할 수도 있다. 모션 보상 유닛 (82)은 수신된 선택스 엘리먼트들로부터 비디오 인코더 (20)에 의해 사용된 보간 필터들을 결정하고 그 보간 필터들을 사용하여 예측 블록들을 생성할 수도 있다.

[0083] 역 양자화 유닛 (86)은 비트스트림으로 제공되고 엔트로피 디코딩 유닛 (80)에 의해 디코딩된 양자화된 변환 계수들을 역 양자화, 즉, 디퀀타이즈 (dequantize) 한다. 역 양자화 프로세스는 양자화 정도와, 마찬가지로 적용되어야 할 역 양자화의 정도를 결정하기 위해, 비디오 슬라이스에서의 각각의 비디오 블록에 대해 비디오 인코더 (20)에 의해 계산된 양자화 파라미터의 사용을 포함할 수도 있다. 역 변환 프로세싱 유닛 (88)은 화소 도메인에서 잔차 블록들을 생성하기 위하여 역 변환, 예컨대, 역 DCT, 역 정수 변환, 또는 개념적으로 유사한 역 변환 프로세스를 변환 계수들에 적용한다.

[0084] 모션 보상 유닛 (82)이 현재 비디오 블록에 대한 예측 블록을 모션 벡터들 및 다른 선택스 엘리먼트들에 기초하여 생성한 후, 비디오 디코더 (30)는 역 변환 프로세싱 유닛 (88)으로부터의 잔차 블록들을 모션 보상 유닛 (82)에 의해 생성된 대응하는 예측 블록들과 합산함으로써 디코딩된 비디오 블록을 형성한다. 합산기 (90)는 이 합산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 블록화제거 필터 (91)는 블록현상 아티팩트들을 제거하기 위하여 합산기 (90)로부터 수신된 블록들을 필터링하는데 적용된다. 주어진 화상에서의 디코딩된 비디오 블록들은 그러면 참조 화상 메모리 (92)에 저장되며, 그 참조 화상 메모리는 후속 모션 보상을 위해 사용되는 참조 화상들을 저장한다. 참조 화상 메모리 (92)는 또한 디스플레이 디바이스, 이를테면 도 1의 디스플레이 디바이스 (32)상의 나중의 프레젠테이션을 위해 디코딩된 비디오를 저장한다.

[0085] 비디오 디코더 (30)에서의 블록화제거 필터 (91)는 경계 강도 컴퓨테이션 및 블록화제거 결정들의 결과에 기초하여 디코딩된 비디오 블록의 특정한 TU 및 PU 에지들을 필터링한다. 경계 강도 컴퓨테이션 및 블록화제거 결정들은 임계 값들 (t_c 및 β)에 의존하며, 그것들은 선택스 엘리먼트들을 사용하여 비디오 인코더 (20)로부터 비디오 디코더 (30)로 시그널링될 수도 있다. 블록화제거 필터 (91)는 비디오 블록의 주어진 에지 근처의 화소들의 값들을 그 에지에서 인지가 가능한 블록현상 아티팩트들을 제거하기 위하여 변경할 수도 있다. 블록화제거 필터 (91)는 블록화제거 필터 (91)가 블록화제거 필터 (63)에 관해 설명된 기법들 중 임의의 것 또는 모두를 수행하도록 구성될 수도 있다는 점에서 도 2에서의 블록화제거 필터 (63)에 실질적으로 일치한다. 블록화제거 프로세스는 도 4에 예시된 블록화제거 필터 (100)에 관해 아래에서 더 상세히 설명된다.

- [0086] 본 개시물의 기법들에 따라, 비디오 디코더 (30) 에서의 엔트로피 디코딩 유닛 (80) 은 비디오 인코더 (20) 로부터 수신된 비트스트림에 포함된 블록화제거 필터 제어 선택스 엘리먼트들을 디코딩한다. 블록화제거 필터 제어 선택스 엘리먼트들은, 블록화제거 필터링이 가능하게 되는지 또는 가능하지 않게 되는지를 나타내는 블록화제거 필터 파라미터들과, 가능하게 되면, 임계 값들 (t_c 및 β) 에 대한 블록화제거 필터 파라미터 오프셋들을 포함한다. 비디오 인코더 (30) 는 비트스트림에 포함된 블록화제거 필터 제어 선택스 엘리먼트들로부터 블록화제거 필터 (91) 를 위해 사용될 블록화제거 필터 파라미터들을 결정한다. 그 다음에 비디오 디코더 (30) 는 비트스트림에서의 비디오 블록들을 디코딩하기 위하여 비디오 인코더 (20) 에서의 블록화제거 필터 (63) 와 동일하거나 또는 유사하게 동작하기 위해 블록화제거 필터 파라미터들에 기초하여 블록화제거 필터 (91) 를 정의한다.
- [0087] 본 개시물은, 감소된 비트스트림 오버헤드로, 비디오 데이터의 현재 슬라이스에 대한 블록화제거 필터 (91) 를 정의하는데 사용되는 블록화제거 필터 파라미터들을 시그널링하는 기법들을 설명한다. 블록화제거 필터 파라미터들은 비디오 디코더 (30) 로의 시그널링을 위해 화상 계층 파라미터 세트 및 슬라이스 헤더 중 하나 이상에서 코딩될 수도 있다. 화상 계층 파라미터 세트는 화상 파라미터 세트 (PPS) 또는 적응 파라미터 세트 (APS) 중 어느 하나를 포함할 수도 있다. PPS는 PPS를 참조하는 화상들 사이에서 변화할 가능성이 없는 데이터를 포함하는 화상 계층 파라미터 세트이다. APS는 화상에서 화상으로 변화할 가능성이 있는 화상 적응 데이터와 함께 사용하기 위해 의도된 화상 계층 파라미터 세트이다.
- [0088] 비디오 디코더 (30) 의 엔트로피 디코딩 유닛 (80) 은 화상 계층 파라미터 세트를 참조하는 화상들에 대해 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 디코딩한다. 본 발명에서 설명되는 기법들에 따르면, 엔트로피 디코딩 유닛 (80) 은 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는 경우 슬라이스 헤더에서의 제 2 선택스 엘리먼트만을 디코딩한다.
- [0089] 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않는 경우, 엔트로피 디코딩 유닛 (80) 은, 블록화제거 필터 파라미터들 중 어떤 세트를 현재 비디오 슬라이스에 대한 블록화제거 필터 (91) 를 정의하는데 사용할지를 나타내도록 정의된 제 2 선택스 엘리먼트가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정한다. 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 하나에만 존재하는 경우에, 블록화제거 필터 (91) 는 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 어느 하나에 존재하는 블록화제거 필터 파라미터들의 세트에 기초하여 현재 비디오 슬라이스에 대해 정의된다. 그러므로, 제 2 선택스 엘리먼트는 불필요한데, 비디오 디코더 (30) 가 블록화제거 필터 파라미터들 중 어떤 세트를 비디오 디코더 (30) 에서의 블록화제거 필터 (91) 를 정의하는데 사용할지를 결정할 필요가 없기 때문이다.
- [0090] 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는 경우, 엔트로피 디코딩 유닛 (80) 은 화상 계층 파라미터 세트에 포함된 블록화제거 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 디코딩한다. 이 경우, 블록화제거 필터 (91) 는 블록화제거 파라미터들의 제 1 세트 또는 제 2 세트 중 하나에 기초하여 현재 비디오 슬라이스에 대해 정의된다. 그러므로, 블록화제거 필터 파라미터들 중 어떤 세트를 비디오 인코더 (20) 에서의 블록화제거 필터 (63) 와 동일하거나 또는 유사하게 되도록 블록화제거 필터 (91) 를 정의하는데 사용할지를 비디오 디코더 (30) 가 알게 하도록 제 2 선택스 엘리먼트는 필요하다.
- [0091] 일부 경우들에서, 엔트로피 디코딩 유닛 (80) 은 또한 임의의 블록화제거 필터 제어 선택스 엘리먼트들이 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 어느 하나에 존재하는지의 여부를 나타내도록 정의된 제어 존재 선택스 엘리먼트를 디코딩한다. 제어 존재 선택스 엘리먼트는 화상 계층 파라미터 세트로부터 또는 높은 계층 파라미터 세트, 예컨대, 시퀀스 파라미터 세트 (SPS) 로부터 디코딩될 수도 있다. 블록화제거 필터 제어 선택스 엘리먼트들은 위에서 설명된 제 1 및 제 2 선택스 엘리먼트들을 포함한다. 그러므로, 엔트로피 디코딩 유닛 (80) 은, 제 1 선택스 엘리먼트를 디코딩하기 전에 제어 존재 선택스 엘리먼트를 디코딩한다. 블록화제거 필터 제어 선택스 엘리먼트들이 존재하지 않음을 제어 존재 선택스 엘리먼트가 나타내면, 제 1 및 제 2 선택스 엘리먼트들이 디코딩될 비트스트림에 존재하지 않기 때문에, 비디오 디코더 (30) 는 제 1 또는 제 2 선택스 엘리먼트들을 디코딩할 필요가 없다는 것을 알게 된다. 이 경우, 비디오 디코더 (30) 는 디코딩된 비디오 블록들에 적용되는 블록화제거 필터 (91) 를 정의하기 위해 디폴트 블록화제거 필터 파라미터들을 사용할 수

도 있다.

[0092] 다른 경우들에서, 엔트로피 디코딩 유닛 (80) 은 제 1 선택스 엘리먼트를 디코딩하기 전에 블록화제거 필터 (91) 가 비디오 시퀀스의 하나 이상의 화상들에 대해 가능하게 되는지의 여부를 나타내도록 정의된 블록화제거 필터 가능 선택스 엘리먼트를 디코딩할 수도 있다. 블록화제거 필터 가능 선택스 엘리먼트는 높은 계층 파라미터 세트, 예컨대, 시퀀스 파라미터 세트 (SPS) 로부터 디코딩될 수도 있다. 블록화제거 필터 (91) 가 비디오 시퀀스에 대해 가능하지 않게 되면, 블록화제거 필터 (91) 가 디코딩된 비디오 블록들에 적용되지 않기 때문에, 비디오 디코더 (30) 는 제 1 또는 제 2 선택스 엘리먼트들을 디코딩할 필요가 없다는 것을 알게 된다. 이 경우, 비디오 디코더 (30) 는 또한 제어 존재 선택스 엘리먼트를 디코딩하지 않는다.

[0093] 하나의 예에서, 제 1 선택스 엘리먼트는 주어진 화상에 대해 PPS에서 코딩된 오버라이드 가능 플래그를 포함한다. 이 경우, 블록화제거 필터 파라미터들의 제 1 세트가 PPS에서 코딩되고 오버라이드 가능 플래그는, 블록화제거 필터 파라미터들의 제 2 세트가 PPS로부터의 파라미터들을 오버라이드하는데 사용될 수 있는 주어진 화상의 하나 이상의 슬라이스들에 대한 슬라이스 헤더에 존재하는지의 여부를 나타낸다. 덧붙여서, 제 2 선택스 엘리먼트는 슬라이스 헤더에 코딩될 수도 있는 오버라이드 플래그를 포함한다. 블록화제거 필터 파라미터들의 제 2 세트가 슬라이스 헤더에 존재함을 PPS에서의 오버라이드 가능 플래그가 나타내는 경우, 엔트로피 디코딩 유닛 (80) 은 블록화제거 필터 (91) 를 정의하기 위해, PPS에서의 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 블록화제거 필터 파라미터들의 제 1 세트를 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트로 오버라이드할지를 결정하기 위해 오버라이드 플래그를 디코딩한다. 그렇지 않고, PPS에서의 블록화제거 필터 파라미터들의 제 1 세트만이 존재함을 PPS에서의 오버라이드 가능 플래그가 나타내는 경우, 엔트로피 디코딩 유닛 (80) 은 오버라이드 플래그가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정한다.

[0094] 표 1은 오버라이드 가능 플래그, 즉, deblocking_filter_override_enabled_flag, 및 제어 존재 선택스 엘리먼트, 즉, deblocking_filter_control_present_flag를 포함하는 PPS 선택스의 예시적인 부분을 제공한다.

[0095] 표 1. 오버라이드 가능 플래그를 갖는 PPS 선택스

pic_parameter_set_rbsp() {	디스크립터
...	
deblocking_filter_control_present_flag	u(1)
if(deblocking_filter_control_present_flag) {	
deblocking_filter_override_enabled_flag	u(1)
pic_disable_deblocking_filter_flag	u(1)
if(!pic_disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
...	

[0096]

[0097] 표 1의 PPS 선택스에 대한 의미체계 (semantics) 는 다음과 같이 정의된다. 1과 동일한 **deblocking_filter_control_present_flag**는 화상 파라미터 세트를 참조하는 화상들에 대해 화상 파라미터 세트에서 및 슬라이스 헤더에서 블록화제거 필터 제어 선택스 엘리먼트들의 존재를 특징한다. 0과 동일한 deblocking_filter_control_present_flag는 화상 파라미터 세트를 참조하는 화상들에 대해 화상 파라미터 세트에서 및 슬라이스 헤더에서 블록화제거 필터 제어 선택스 엘리먼트들의 부재를 특징한다.

[0098] 1과 동일한 **deblocking_filter_override_enabled_flag**는 화상 파라미터 세트를 참조하는 화상들에 대해 슬라이스 헤더에서 deblocking_filter_override_flag의 존재를 특징한다. 0과 동일한 deblocking_filter_override_enabled_flag는 화상 파라미터 세트를 참조하는 화상들에 대해 슬라이스 헤더에서 deblocking_filter_override_flag의 부재를 특징한다. 존재하지 않는 경우, deblocking_filter_override_enabled_flag의 값은 0과 동일한 것으로 유추된다.

[0099] 1과 동일한 **pic_disable_deblocking_filter_flag**는 deblocking_filter_override_enabled_flag가 0과 동일한 경우 블록화제거 필터의 동작이 화상 파라미터 세트를 참조하는 화상들에 적용되지 않아야 한다고 특징한다. 0과 동일한 pic_disable_deblocking_filter_flag는 deblocking_filter_override_enabled_flag가 0과 동일한 경

우 블록화제거 필터의 동작이 화상 파라미터 세트를 참조하는 화상들에 적용되어야 한다고 특정한다. 존재하지 않는 경우, `pic_disable_deblocking_filter_flag`의 값은 0과 동일한 것으로 유추된다.

[0100] **beta_offset_div2** 및 **tc_offset_div2** 신택스 엘리먼트들은, 디폴트 블록화제거 파라미터 오프셋들이 화상 파라미터 세트를 참조하는 화상들에 대한 슬라이스 세그먼트 헤더에 존재하는 블록화제거 파라미터 오프셋들에 의해 오버라이드되지 않는 한 화상 파라미터 세트를 참조하는 화상들에 적용되는 (2로 나누어진) β 및 t_c 에 대한 디폴트 블록화제거 파라미터 오프셋들을 특정한다. **beta_offset_div2** 및 **tc_offset_div2** 신택스 엘리먼트들의 값들은 양쪽 모두가 -6 내지 6의 범위에 있을 수 있다. 존재하지 않는 경우, **beta_offset_div2** 및 **tc_offset_div2** 신택스 엘리먼트들의 값들은 0과 동일한 것으로 유추된다.

[0101] 표 2는 PPS 신택스에서의 오버라이드 가능 플래그 및 제어 존재 신택스 엘리먼트에 기초하여 조건부로 코딩되는 오버라이드 플래그, 즉, `deblocking_filter_override_flag`를 포함하는 슬라이스 헤더 신택스의 예시적인 부분을 제공한다.

[0102] 표 2. 오버라이드 플래그를 갖는 슬라이스 헤더 신택스

<code>slice_segment_header()</code> {	디스크립터
...	
if(<code>deblocking_filter_control_present_flag</code>) {	
if(<code>deblocking_filter_override_enabled_flag</code>)	
<code>deblocking_filter_override_flag</code>	u(1)
if(<code>deblocking_filter_override_flag</code>) {	
<code>slice_disable_deblocking_filter_flag</code>	u(1)
if(! <code>slice_disable_deblocking_filter_flag</code>) {	
<code>beta_offset_div2</code>	se(v)
<code>tc_offset_div2</code>	se(v)
}	
}	
}	
...	

[0103]

[0104] 표 2의 슬라이스 헤더 신택스에 대한 의미체계는 다음과 같이 정의된다. 0과 동일한 **`deblocking_filter_override_flag`**는 액티브 화상 파라미터 세트로부터의 블록화제거 파라미터들이 현재 슬라이스를 블록화제거하는데 사용된다는 것을 특정한다. 1과 동일한 **`deblocking_filter_override_flag`**는 슬라이스 세그먼트 헤더로부터의 블록화제거 파라미터들이 현재 슬라이스를 블록화제거하는데 사용된다는 것을 특정한다. 존재하지 않는 경우, **`deblocking_filter_override_flag`**의 값은 0과 동일한 것으로 유추된다.

[0105] 1과 동일한 **`slice_disable_deblocking_filter_flag`**는 블록화제거 필터의 동작이 현재 슬라이스에 대해 적용되지 않는다는 것을 특정한다. 0과 동일한 **`slice_disable_deblocking_filter_flag`**는 블록화제거 필터의 동작이 현재 슬라이스에 대해 적용된다는 것을 특정한다. **`slice_disable_deblocking_filter_flag`**가 존재하지 않는 경우, 그것은 PPS 신택스에서의 **`pic_disable_deblocking_filter_flag`**와 동일한 것으로 유추된다.

[0106] **beta_offset_div2** 및 **tc_offset_div2** 신택스 엘리먼트들은 현재 슬라이스에 대해 (2로 나누어진) β 및 t_c 에 대한 블록화제거 파라미터 오프셋들을 특정한다. **beta_offset_div2** 및 **tc_offset_div2** 신택스 엘리먼트들의 값들은 -6 내지 6의 범위에 있을 수 있다.

[0107] 다른 예에서, 제 1 신택스 엘리먼트는 주어진 화상에 대해 SPS 및/또는 APS에 코딩된 상속 가능 플래그를 포함한다. 이 경우, 블록화제거 필터 파라미터들의 제 2 세트는 슬라이스 헤더에서 코딩되고 상속 가능 플래그는 블록화제거 필터 파라미터들의 제 1 세트가 슬라이스 헤더에 의해 상속될 수 있는 APS에 존재하는지의 여부를 나타낸다. 제 2 신택스 엘리먼트는 슬라이스 헤더에 코딩될 수도 있는 상속 가능 플래그를 포함한다. 블록화제거 필터 파라미터들의 제 1 세트가 APS에 존재함을 SPS 및/또는 APS에서의 상속 가능 플래그가 나타내는 경우, 엔트로피 디코딩 유닛 (80) 은, 블록화제거 필터 (91) 를 정의하기 위해, 슬라이스 헤더에서의 블록화제거 필터 파라미터들의 제 2 세트를 사용할지 또는 APS에서의 블록화제거 필터 파라미터들의 제 1 세트를 상속할지를 결정하기 위해 상속 가능 플래그를 디코딩한다. 그렇지 않고, 슬라이스 헤더에서의 블록화제거 필터 파라미터들의 제 2 세트만이 존재함을 SPS 및/또는 APS에서의 상속 가능 플래그가 나타내는 경우, 엔트로피 디코딩 유

닛 (80) 은 상속 플래그가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정한다.

[0108] 블록화제거 필터 파라미터들을 비디오 디코더로 시그널링하는 것은, 『A. Norkin, "BoG report on resolving deblocking filter description issues," 7th JCT-VC Meeting, Geneva, CH, Nov. 2011, Doc. JCT-VC G1035_r1 + update』에서 제안되었다. 표 3은 상속 가능 플래그, 즉, deblocking_filter_in_aps_enabled_flag를 포함하는 SPS 선택스의 일 예를 제공한다.

[0109] 표 3. 상속 가능 플래그를 갖는 SPS 선택스

seq_parameter_set_rbsp() {	디스크립터
profile_idc	u(8)
<i>(omitted)</i>	
chroma_pred_from_luma_enabled_flag	u(1)
deblocking_filter_in_aps_enabled_flag	u(1)
loop_filter_across_slice_flag	u(1)
sample_adaptive_offset_enabled_flag	u(1)
adaptive_loop_filter_enabled_flag	u(1)
pcm_loop_filter_disable_flag	u(1)
cu_qp_delta_enabled_flag	u(1)
temporal_id_nesting_flag	u(1)
inter_4x4_enabled_flag	u(1)
rbsp_trailing_bits()	
}	

[0110]

[0111] 표 4는 상속 가능 플래그, 즉, aps_deblocking_filter_flag를 포함하는 APS 선택스의 일 예를 제공한다.

[0112] 표 4. 상속 가능 플래그를 갖는 APS 선택스

aps_rbsp() {	디스크립터
aps_id	ue(v)
aps_deblocking_filter_flag	u(1)
aps_sample_adaptive_offset_flag	u(1)
aps_adaptive_loop_filter_flag	u(1)
if(aps_sample_adaptive_offset_flag aps_adaptive_loop_filter_flag) {	
aps_cabac_use_flag	u(1)
if(aps_cabac_use_flag) {	
aps_cabac_init_idc	ue(v)
aps_cabac_init_qp_minus26	se(v)
}	
}	
if(aps_deblocking_filter_flag){	
disable_deblocking_filter_flag	u(1)
if (!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
if(aps_sample_adaptive_offset_flag) {	
sao_data_byte_count	u(8)
byte_align()	
sao_param()	
byte_align()	
}	
if(aps_adaptive_loop_filter_flag) {	
alf_data_byte_count	u(8)
byte_align()	
alf_param()	
}	
rbsp_trailing_bits()	
}	

[0113]

[0114] 표 3 및 표 4의 SPS 및 APS 선택스에 대한 의미체계들은 다음과 같이 정의된다. 0과 동일한 SPS에서의 **deblocking_filter_in_aps_enabled_flag**는 블록화제거 필터 파라미터들이 슬라이스 헤더에 존재함을 의미하고, 1과 동일한 SPS에서의 **deblocking_filter_in_aps_enabled_flag**는 블록화제거 필터 파라미터들이 APS에 존재함을 의미한다. APS에서의 **aps_deblocking_filter_flag**는 SPS에서의 **deblocking_filter_in_aps_enabled_flag**와 동일하다. **aps_deblocking_filter_flag**는 블록화제거 필터 파라미터들이 APS에 존재하거나 (1과 동일) 또는 APS에 존재하지 않는다 (0과 동일) 는 것을 나타낸다.

[0115] 블록화제거 필터 파라미터들의 시그널링에 대한 하나의 문제는, 심지어 블록화제거 필터 파라미터들이 APS에 존재하지 않는 경우에도 상속 플래그가 슬라이스 헤더에서 시그널링된다는 것이다. 위에서 설명된 바와 같이, 블록화제거 필터 파라미터들이 슬라이스 헤더에만 존재하고 APS에 존재하지 않는 경우, 블록화제거 필터는 슬라이스 헤더에서의 현재 블록화제거 필터 파라미터들에 기초하여 정의되고, 상속 플래그는 불필요하다. 표 5는 APS 선택스 및 SPS 선택스에서 상속 가능 플래그들에 기초하여 조건부로 코딩되는 상속 플래그, 즉, **inherit_dbl_params_from_APS_flag**를 포함하는 슬라이스 헤더 선택스의 예시적인 부분을 제공한다.

[0116] 표 5. 상속 플래그를 갖는 슬라이스 헤더 선택스

slice_header() {	디스크립터
....	u(v)
if(!entropy_slice_flag) {	
slice_qp_delta	se(v)
if(deblocking_filter_in_aps_enabled_flag && aps_deblocking_filter_flag) {	
inherit_dbl_params_from_APS_flag	u(1)
/* deblocking_filter_in_aps_enabled_flag = 0 또는 aps_deblocking_filter_flag=0 이면 inherit_dbl_params_from_APS_flag = 0이 유추된다는 것에 주의한다. */	
}	
if(!inherit_dbl_params_from_APS_flag) {	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
.....	

[0117]

[0118] 대체예로서, 표 6은 SPS 선택스에서의 상속 가능 플래그에 기초하여 조건부로 코딩되는 상속 플래그, 즉, inherit_dbl_params_from_APS_flag를 포함하는 슬라이스 헤더 선택스의 예시적인 부분을 제공한다.

[0119] 표 6. 상속 플래그를 갖는 슬라이스 헤더 선택스

slice_header() {	디스크립터
....	u(v)
if(!entropy_slice_flag) {	
slice_qp_delta	se(v)
if(deblocking_filter_in_aps_enabled_flag) {	
inherit_dbl_params_from_APS_flag	u(1)
/* deblocking_filter_in_aps_enabled_flag = 0이면 inherit_dbl_params_from_APS_flag = 0이 유추된다는 것에 주의한다. */	
}	
if(!inherit_dbl_params_from_APS_flag) {	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
.....	

[0120]

[0121] 표 5 및 표 6의 슬라이스 헤더 선택스들에 대한 의미체계들은 다음과 같이 정의된다. 0과 동일한 **disable_deblocking_filter_flag**는 블록화제거 필터가 가능하게 됨을 의미하고, 1과 동일한 **disable_deblocking_filter_flag**는 블록화제거 필터가 가능하지 않게 됨을 의미한다. **beta_offset_div2** 및 **tc_offset_div2** 선택스 엘리먼트들은 (2로 나누어진) t_c 및 β 에 대한 블록화제거 파라미터 오프셋들을 나타낸다. 1과 동일한 **inherit_dbl_params_from_APS_flag**는 APS에 존재하는 블록화제거 필터 파라미터들이 사용될 것임을 의미하고, 0과 동일한 **inherit_dbl_params_from_APS_flag**는 슬라이스 헤더에서 뒤따르는 블록화제거 필터 파라미터들이 사용될 것임을 의미한다.

[0122] 블록화제거 필터 파라미터들의 시그널링에 대한 제 2의 문제는, 블록화제거 필터가 비디오 시퀀스의 화상들에 대해 가능하게 되는지의 여부를 나타내기 위해 SPS 레벨 가능/불능 플래그가 정의되지 않는다는 것이다. 블록화제거 필터가 가능하지 않게 되는 경우, 블록화제거 필터 파라미터들은 블록화제거 필터를 정의하는데 필요하지 않고 블록화제거 필터 파라미터들을 코딩하는 것은 불필요하다. 표 7은 블록화제거 필터 가능 플래그, 즉, deblocking_filter_enabled_flag에 기초하여 조건부로 코딩되는 상속 가능 플래그, 즉,

deblocking_in_aps_enabled_flag를 포함하는 SPS 선택스의 일 예를 제공한다.

표 7. 블록화제거 필터 가능 플래그 및 상속 가능 플래그를 갖는 SPS 선택스

seq_parameter_set_rbsp() {	디스크립터
profile_idc	u(8)
(omitted)	
chroma_pred_from_luma_enabled_flag	u(1)
deblocking_filter_enabled_flag	u(1)
if(deblocking_filter_enable_flag) {	
deblocking_filter_in_aps_enabled_flag	u(1)
/* deblocking_filter_enabled_flag=0이면, deblocking_filter_in_aps_enabled_flag=0 및 aps_deblocking_filter_flag=0 및 disable_deblocking_filter_flag=1 이 유추된다는 것에 주의한다 */	
}	
loop_filter_across_slice_flag	u(1)
sample_adaptive_offset_enabled_flag	u(1)
adaptive_loop_filter_enabled_flag	u(1)
pcm_loop_filter_disable_flag	u(1)
cu_qp_delta_enabled_flag	u(1)
temporal_id_nesting_flag	u(1)
inter_4x4_enabled_flag	u(1)
rbsp_trailing_bits()	
}	

표 7의 SPS 선택스에 대한 의미체계는 다음과 같이 정의된다. 0과 동일한 **deblocking_filter_enabled_flag**는 블록화제거 필터가 가능하지 않게 됨을 의미하고, 1과 동일한 **deblocking_filter_enabled_flag**는 블록화제거 필터가 가능하게 됨을 의미한다.

이런 식으로, 블록화제거 필터가 SPS 레벨에서 가능하지 않게 되는 경우, 블록화제거 파라미터들이 APS에서 시그널링되지 않는다 (즉, SPS에서의 **deblocking_filter_in_aps_enabled_flag**가 0과 동일하고 APS에서의 **aps_deblocking_filter_flag**가 0과 동일하다) 고 유추될 수도 있다. 덧붙여서, 블록화제거 필터가 SPS 레벨에서 가능하지 않게 되는 경우, 슬라이스 헤더에서의 **disable_deblocking_filter_flag**는, 블록화제거 필터가 슬라이스 레벨에서 가능하지 않게 됨을 나타내는 1과 동일하다고 유추될 수도 있다.

표 8은 블록화제거 필터가 SPS 레벨에서 가능하지 않게 되는 경우 상속 가능 플래그, 즉, **aps_deblocking_filter_flag**가 0과 동일하게 된다고 유추되는 APS 선택스의 일 예를 제공한다.

[0128] 표 8. 상속 가능 플래그를 갖는 APS 선택스

aps_rbsp() {	디스크립터
aps_id	ue(v)
aps_deblocking_filter_flag	u(1)
aps_sample_adaptive_offset_flag	u(1)
aps_adaptive_loop_filter_flag	u(1)
if(aps_sample_adaptive_offset_flag aps_adaptive_loop_filter_flag) {	
aps_cabac_use_flag	u(1)
if(aps_cabac_use_flag) {	
aps_cabac_init_idc	ue(v)
aps_cabac_init_qp_minus26	se(v)
}	
}	
if(aps_deblocking_filter_flag){	
disable_deblocking_filter_flag	u(1)
if (!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
...	

[0129]

[0130] 표 9는 SPS 선택스에서의 블록화제거 필터 가능 플래그, 즉, deblocking_filter_enabled_flag에 기초하여 조건부로 코딩되는 블록화제거 필터 파라미터들을 포함하는 슬라이스 헤더 선택스의 예시적인 부분을 제공한다.

[0131] 표 9. 상속 플래그 및 조건부로 코딩된 블록화제거 필터 파라미터들을 갖는 슬라이스 헤더 선택스

slice_header() {	디스크립터
entropy_slice_flag	u(1)
if(!entropy_slice_flag) {	
slice_type	ue(v)
pic_parameter_set_id	ue(v)
if(sample_adaptive_offset_enabled_flag adaptive_loop_filter_enabled_flag deblocking_filter_in_aps_enabled_flag)	
aps_id	ue(v)
frame_num	u(v)
if(IdrPicFlag)	
idr_pic_id	ue(v)
....	u(v)
if(!entropy_slice_flag) {	
slice_qp_delta	se(v)
if(deblocking_filter_in_aps_enabled_flag && aps_deblocking_filter_flag) {	
inherit_dbl_params_from_APS_flag	u(1)
/* deblocking_filter_in_aps_enabled_flag = 0 또는 aps_deblocking_filter_flag=0 이면 inherit_dbl_params_from_APS_flag = 0이 유추된다는 것에 주의한다. */	
}	
if((!inherit_dbl_params_from_APS_flag) && deblocking_filter_enabled_flag) {	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
.....	

[0132]

[0133] 대체예로서, SPS에서의 블록화제거 필터 가능 플래그에 기초하여 슬라이스 헤더에서의 블록화제거 필터 파라미터들을 코딩하는 조건을 도입하는 대신, 그 기법들은, 블록화제거 필터가 SPS 레벨에서 가능하지 않게 되는 경우, 블록화제거 필터 제어 선택스 엘리먼트들이 슬라이스 헤더에서 시그널링되지 않는다고 유추할 수도 있다.

PPS 선택스에 포함되는 제어 존재 선택스 엘리먼트, 즉, deblocking_filter_control_present_flag는, 블록화제거 필터 파라미터들을 포함하는 블록화제거 필터 제어 선택스 엘리먼트들이 슬라이스 헤더에서 시그널링되는지의 여부를 나타내기 위해 정의된다. 이 경우, 블록화제거 필터 제어 선택스 엘리먼트들이 슬라이스 헤더에서 시그널링되는 경우 블록화제거 필터 파라미터들은 슬라이스 헤더에서만 코딩되며, 이는 블록화제거 필터가 SPS 레벨에서 가능하게 되는 경우에만 발생한다. PPS에서의 제어 존재 선택스 엘리먼트는, HEVC "규격 초안 6," HEVC WD6, 또는 간단히 WD6라고 또한 지칭되는 『Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, T. Wiegand, "High efficiency video coding (HEVC) text specification draft 6," 8th JCT-VC Meeting, San Jose, CA, USA, Feb. 2012』에서 제안되어 있다.

[0134]

표 10은 블록화제거 필터 가능 플래그에 기초하여 조건부로 코딩된 상속 가능 플래그를 포함하는 SPS 선택스의 일 예를 제공하고, 거기서 PPS에서의 제어 존재 선택스 엘리먼트, 즉, deblocking_filter_control_present_flag는, 블록화제거 필터가 SPS 레벨에서 가능하지 않게 되는 경우 0과 동일하다고 유추된다.

[0135] 표 10. 블록화제거 필터 가능 플래그 및 상속 가능 플래그를 갖는 SPS 선택스

seq_parameter_set_rbsp() {	디스크립터
profile_idc	u(8)
<i>(omitted)</i>	
chroma_pred_from_luma_enabled_flag	u(1)
deblocking_filter_enabled_flag	u(1)
if (deblocking_filter_enabled_flag) {	
deblocking_filter_in_aps_enabled_flag	u(1)
deblocking_filter_enabled_flag=0이면, deblocking_filter_in_aps_enabled_flag=0 및 aps_deblocking_filter_flag=0 및 disable_deblocking_filter_flag=1 및 deblocking _filter_control_present_flag=0이 유추될 것이라는 것에 주의한다	
}	
loop_filter_across_slice_flag	u(1)
sample_adaptive_offset_enabled_flag	u(1)
adaptive_loop_filter_enabled_flag	u(1)
pcm_loop_filter_disable_flag	u(1)
cu_qp_delta_enabled_flag	u(1)
temporal_id_nesting_flag	u(1)
inter_4x4_enabled_flag	u(1)
rsp_trailing_bits()	
}	

[0136]

[0137] 표 11은 PPS 선택스에서의 제어 존재 선택스 엘리먼트, 즉, deblocking_filter_control_present_flag에 기초하여 조건부로 코딩되는 블록화제거 필터 파라미터들을 포함하는 슬라이스 헤더 선택스의 예시적인 부분을 제공한다.

[0138] 표 11. 상속 플래그 및 조건부로 코딩된 블록화제거 필터 파라미터들을 갖는 슬라이스 헤더 선택스

slice_header() {	디스크립터
slice_type	uc(v)
entropy_slice_flag	u(1)
if(!entropy_slice_flag) {	
pic_parameter_set_id	uc(v)
}	
if(scaling_list_enable_flag deblocking_filter_in_aps_enabled_flag (sample_adaptive_offset_enabled_flag && !slice_sao_interleaving_flag) adaptive_loop_filter_enabled_flag)	
aps_id	uc(v)
}	
....	
if(!entropy_slice_flag) {	
slice_qp_delta	se(v)
if(deblocking_filter_control_present_flag) {	
if(deblocking_filter_in_aps_enabled_flag) {	
inherit_dbl_params_from_APS_flag	u(1)
/*deblocking_filter_in_aps_enabled_flag=0이면, inherit_dbl_params_from_APS_flag=0 이 유추된다는 것에 주의한다 */	
}	
if((!inherit_dbl_params_from_APS_flag) {	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
}	
}	
.....	

[0139]

[0140]

표 10 및 표 11에 관해 설명된 예와 표 7 내지 표 9에 관해 설명된 예 사이의 차이는, SPS에서의 deblocking_filter_enable_flag가 0과 동일한 경우, PPS에서의 deblocking_filter_control_present_flag가 또한 0과 동일하다고 유추된다는 것이다. 이런 식으로, SPS 레벨에서 가능하지 않게 된 블록화제거 필터의 결과는, 블록화제거 필터 파라미터들이 APS 또는 슬라이스 헤더에서 시그널링되지 않을 것이라는 것, 및 블록화제거 필터가 슬라이스 레벨에서 효과적으로 가능하지 않게 될 것이라는 것이다. 더 구체적으로는, 블록화제거 필터가 SPS 레벨에서 가능하지 않게 되는 경우, 비디오 디코더 (30) 는, 블록화제거 필터 파라미터들이 APS에 존재하지 않는다 (즉, SPS에서의 deblocking_filter_in_aps_enabled_flag는 0과 동일하고 APS에서의 aps_deblocking_filter_flag는 0과 동일하다) 는 것과, 블록화제거 필터 파라미터들이 슬라이스 헤더에서 시그널링되지 않는다 (즉, PPS에서의 deblocking_filter_control_present_flag가 0과 동일하다) 는 것을 유추할 수도 있다. 덧붙여서, 비디오 디코더 (30) 는, 슬라이스 헤더에서의 disable_deblocking_filter_flag가, 블록화제거 필터가 슬라이스 레벨에서 가능하지 않게 됨을 나타내는 1과 동일하다고 유추할 수도 있다. 이 경우, 추가적인 조건들이 슬라이스 헤더 선택스에 추가될 필요가 없을 수도 있는데, PPS에서의 deblocking_filter_control_present_flag가 0과 동일한 경우, 블록화제거 필터 파라미터들 중 어느 것도 슬라이스 헤더에서 시그널링되지 않기 때문이다.

[0141]

블록화제거 필터 파라미터들의 시그널링에 대한 제 3 의 문제는, 블록화제거 필터 제어 선택스 엘리먼트들이 시그널링되지 않고 디폴트 파라미터들, 이를테면 0 값들이, 블록화제거 필터를 정의하는데 사용되어야 하는 경우를 나타내도록 SPS 레벨 플래그가 정의되지 않는다는 것이다. 블록화제거 필터 제어 선택스 엘리먼트들이 APS 또는 슬라이스 헤더 중 어느 하나로 시그널링되지 않는 경우, 블록화제거 필터 파라미터들은 블록화제거 필터를 정의하도록 시그널링되지 않는다. 표 12는 제어 존재 선택스 엘리먼트, 즉, deblocking_filter_control_present_flag에 기초하여 조건부로 코딩된 상속 가능 플래그, 즉, deblocking_in_aps_enabled_flag를 포함하는 SPS 선택스의 일 예를 제공한다.

[0142] 표 12. 제어 존재 선택스 엘리먼트 및 조건부로 코딩된 상속 가능 플래그를 갖는 SPS 선택스

seq_parameter_set_rbsp() {	디스크립터
profile_idc	u(8)
<i>(omitted)</i>	
chroma_pred_from_luma_enabled_flag	u(1)
deblocking_filter_control_present_flag	u(1)
if (deblocking_filter_control_present_flag) {	
deblocking_filter_in_aps_enabled_flag	u(1)
/* deblocking_filter_control_present_flag=0이면, deblocking_filter_in_aps_enabled_flag=0 및 aps_deblocking_filter_flag=0 및 disable_deblocking_filter_flag=0 및 beta_offset_div2=0 및 tc_offset_div2=0이 유추된다는 것에 주의한다 */	
}	
loop_filter_across_slice_flag	u(1)
sample_adaptive_offset_enabled_flag	u(1)
adaptive_loop_filter_enabled_flag	u(1)
pcm_loop_filter_disable_flag	u(1)
cu_qp_delta_enabled_flag	u(1)
temporal_id_nesting_flag	u(1)
inter_4x4_enabled_flag	u(1)
rbps_trailing_bits()	
}	

[0143]

[0144] 표 13은 블록화제거 필터 제어 선택스 엘리먼트들이 APS 또는 슬라이스 헤더 중 어느 하나에 존재하지 않는 경우, 즉, SPS 선택스에서의 deblocking_filter_control_present_flag가 0과 동일한 경우에 상속 가능 플래그, 즉, aps_deblocking_filter_flag가 0과 동일한 것으로 유추되는 APS 선택스의 일 예를 제공한다.

[0145] 표 13. 상속 가능 플래그를 갖는 APS 선택스

aps_rbsp() {	디스크립터
aps_id	ue(v)
aps_deblocking_filter_flag	u(1)
aps_sample_adaptive_offset_flag	u(1)
aps_adaptive_loop_filter_flag	u(1)
if(aps_sample_adaptive_offset_flag aps_adaptive_loop_filter_flag) {	
aps_cabac_use_flag	u(1)
if(aps_cabac_use_flag) {	
aps_cabac_init_idc	ue(v)
aps_cabac_init_qp_minus26	se(v)
}	
}	
if(aps_deblocking_filter_flag){	
disable_deblocking_filter_flag	u(1)
if (!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
....	

[0146]

[0147] 표 14는 SPS 선택스에서의 제어 존재 선택스 엘리먼트, 즉, deblocking_filter_control_present_flag에 기초하여 조건부로 코딩되는 블록화제거 필터 파라미터들을 포함하는 슬라이스 헤더 선택스의 일 예를 제공한다.

[0148] 표 14. 상속 플래그 및 조건부로 코딩된 블록화제거 필터 파라미터들을 갖는 슬라이스 헤더 신택스

slice_header() {	디스크립터
entropy_slice_flag	u(1)
if(!entropy_slice_flag) {	
slice_type	ue(v)
pic_parameter_set_id	ue(v)
if(sample_adaptive_offset_enabled_flag adaptive_loop_filter_enabled_flag deblocking_filter_in_aps_enabled_flag)	
aps_id	ue(v)
frame_num	u(v)
if(IdrPicFlag)	
idr_pic_id	ue(v)
....	u(v)
if(!entropy_slice_flag) {	
slice_qp_delta	se(v)
if(deblocking_filter_in_aps_enabled_flag && aps_deblocking_filter_flag) {	
inherit_dbl_params_from_APS_flag	u(1)
/* deblocking_filter_in_aps_enabled_flag = 0 또는 aps_deblocking_filter_flag=0 이면 inherit_dbl_params_from_APS_flag = 0이 유추된다는 것에 주의한다. */	
}	
if ((!inherit_dbl_params_from_APS_flag) && deblocking_filter_control_present_flag){	
disable_deblocking_filter_flag	u(1)
if (!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
.....	
}	

[0149]

[0150]

도 4는 본 개시물에서 설명된 기법들에 따라 시그널링된 블록화제거 필터 파라미터들에 기초하여 정의된 예시적인 블록화제거 필터 (100)의 컴포넌트들을 도시하는 블록도이다. 대체로, 도 2로부터의 블록화제거 필터 (63) 및 도 3으로부터의 블록화제거 필터 (91) 중 어느 하나 또는 양쪽 모두는 블록화제거 필터 (100)의 컴포넌트들과 실질적으로 유사한 컴포넌트들을 구비할 수도 있다. 다른 비디오 코딩 디바이스들, 이를테면 비디오 인코더들, 비디오 디코더들, 비디오 인코더/디코더들 (CODEC들) 등은 블록화제거 필터 (100)에 실질적으로 유사한 컴포넌트들을 또한 구비할 수도 있다. 블록화제거 필터 (100)는 하드웨어, 소프트웨어, 펌웨어, 또는 이것들의 임의의 조합으로 구현될 수도 있다. 소프트웨어 또는 펌웨어로 구현되는 경우, 대응하는 하드웨어 (이를테면 하나 이상의 프로세서들 또는 프로세싱 유닛들 및 소프트웨어 또는 펌웨어에 대한 명령들을 저장하는 메모리)가 또한 제공될 수도 있다.

[0151]

도 4의 예에서, 블록화제거 필터 (100)는 블록화제거 결정 유닛 (104), 메모리에 저장된 지원 정의들 (102), 블록화제거 필터링 유닛 (106), 메모리에 저장된 블록화제거 필터 정의들 (108), 에지 로케이팅 유닛 (103), 및 에지 로케이션들의 데이터 구조 (105)를 구비한다. 블록화제거 필터 (100)의 컴포넌트들의 임의의 것 또는 모두는 기능적으로 통합될 수도 있다. 블록화제거 필터 (100)의 컴포넌트들은 예시의 목적으로만 따로 따로 도시되어 있다. 대체로, 블록화제거 필터 (100)는, 예컨대, 블록들에 대한 잔차 데이터와 예측 데이터를 결합하는 합산 컴포넌트로부터 디코딩된 블록들에 대한 데이터를 수신한다. 그 데이터는 추가로 블록들이 예측되었던 방법의 표시를 포함할 수도 있다. 아래에서 설명되는 예에서, 블록화제거 필터 (100)는 LCU 및 그 LCU에 대한 CU 쿼드트리에 연관된 디코딩된 비디오 블록을 포함하는 데이터를 수신하도록 구성되며, 여기서 CU 쿼드트리는 LCU가 CU들로 구획되는 방법 및 리프 노드 CU들의 PU들 및 TU들에 대한 예측 모드들을 기술한다.

[0152]

블록화제거 필터 (100)는 에지 로케이션 데이터 구조 (105)를 블록화제거 필터 (100)의 메모리에서, 또는 대응하는 비디오 코딩 디바이스에 의해 제공된 외부 메모리에서 유지할 수도 있다. 일부 예들에서, 에지 로케이션

이팅 유닛 (103) 은 LCU가 CU들로 구획되는 방법을 나타내는 LCU에 대응하는 CU 쿼드트리를 수신할 수도 있다.

그러면 에지 로케이팅 유닛 (103) 은 CU 쿼드트리를 분석하여 블록화제거를 위한 후보들인 LCU에서의 CU들의 TU들 및 PU들에 연관된 디코딩된 비디오 블록들 사이의 에지들을 결정할 수도 있다.

[0153] 에지 로케이션 데이터 구조 (105) 는 수평 차원, 수직 차원, 및 수평 에지들 및 수직 에지들을 나타내는 차원을 갖는 어레이를 포함할 수도 있다. 대체로, 비디오 블록들 사이의 에지들은 LCU의 최소 사이즈로 된 CU들, 또는 CU들의 TU들 및 PU들에 연관된 2 개의 비디오 블록들 사이에서 발생할 수도 있다. LCU가 NxN의 사이즈를 가진다고 가정하고, LCU의 최소 사이즈로 된 CU가 사이즈 MxM으로 된다고 가정하면, 그 어레이는 $[N/M] \times [N/M] \times 2$ 의 사이즈를 포함할 수도 있으며, 여기서 "2"는 CU들 사이의 에지들의 2 개의 가능한 방향들 (수평 및 수직) 을 나타낸다. 예를 들어, LCU가 64x64 화소들 및 8x8 최소 사이즈로 된 CU를 가진다고 가정하면, 그 어레이는 $[8] \times [8] \times [2]$ 엔트리들을 포함할 수도 있다.

[0154] 각각의 엔트리는 일반적으로 2 개의 비디오 블록들 사이의 가능한 에지에 대응할 수도 있다. 에지들은 에지 로케이션 데이터 구조 (105) 의 엔트리들의 각각에 대응하는 LCU 내의 포지션들의 각각에서 사실상 존재하지 않을 것이다. 따라서, 데이터 구조의 값들은 거짓 (false) 으로서 초기화될 수도 있다. 대체로, 에지 로케이팅 유닛 (103) 은 CU 쿼드트리를 분석하여, LCU의 CU들의 TU들 및 PU들에 연관된 2 개의 비디오 블록들 사이의 에지들의 로케이션들을 결정하고 에지 로케이션 데이터 구조 (105) 에서의 대응하는 값들을 참 (true) 이라고 설정할 수도 있다.

[0155] 대체로, 어레이의 엔트리들은 대응하는 에지가 블록화제거를 위한 후보로서 LCU에 존재하는지의 여부를 기술할 수도 있다. 다시 말하면, LCU의 CU들의 TU들 및 PU들에 연관된 2 개의 이웃하는 비디오 블록들 사이에 에지가 존재한다고 에지 로케이팅 유닛 (103) 이 결정하는 경우, 에지 로케이팅 유닛 (103) 은 그 에지가 존재함을 나타내기 위해 에지 로케이션 데이터 구조 (105) 에서의 대응하는 엔트리의 값을 (예컨대, "참"의 값으로) 설정할 수도 있다.

[0156] 블록화제거 결정 유닛 (104) 은, 2 개의 이웃 블록들에 대해, 그 2 개의 블록들 사이의 에지가 블록화제거되어야 할지의 여부를 일반적으로 결정한다. 블록화제거 결정 유닛 (104) 은 에지 로케이션 데이터 구조 (105) 를 사용하여 에지들의 로케이션들을 결정할 수도 있다. 에지 로케이션 데이터 구조 (105) 의 값이 부울 값을 가지는 경우, 블록화제거 결정 유닛 (104) 은, 일부 예들에서, "참" 값이 에지의 존재를 나타내고, "거짓" 값은 에지가 존재하지 않음을 나타낸다고 결정할 수도 있다.

[0157] 대체로, 블록화제거 결정 유닛 (104) 은 하나 이상의 블록화제거 결정 기능부들로 구성될 수도 있다. 그 기능부들은 블록들 사이의 에지를 가로지르는 화소들의 라인들에 적용되는 복수의 계수들을 포함할 수도 있다. 예를 들어, 기능부들은 에지에 수직인 8 개 화소들의 라인에 적용될 수도 있으며, 여기서 화소들 중 4 개는 2 개의 블록들 중 하나 내에 있고 다른 4 개의 화소들은 2 개의 블록들 중 다른 하나 내에 있다. 지원 정의들 (102) 은 기능부들에 대한 지원들을 정의한다. 대체로, "지원"은 기능부들이 적용되는 화소들에 대응한다. 지원부의 세트들의 다양한 예들이 도 5에 관해 아래에서 더 상세히 설명된다.

[0158] 블록화제거 결정 유닛 (104) 은 비디오 데이터의 2 개의 블록들 사이의 특정 에지가 블록화제거되어야 하는지의 여부를 결정하기 위해, 지원 정의들 (102) 에 의해 정의된 바와 같이, 하나 이상의 블록화제거 결정 기능들을 지원의 하나 이상의 세트들에 적용하도록 구성될 수도 있다. 블록화제거 결정 유닛 (104) 으로부터 시작하는 파선은 필터링되지 않고 출력되고 있는 블록들에 대한 데이터를 나타낸다. 2 개의 블록들 사이의 에지가 필터링되지 않아야 한다고 블록화제거 결정 유닛 (104) 이 결정하는 경우들에서, 블록화제거 필터 (100) 는 그 블록들에 대한 데이터를 변경하는 일 없이 출력할 수도 있다. 다시 말하면, 그 데이터는 블록화제거 필터링 유닛 (106) 을 바이패스 할 수도 있다. 한편, 에지가 블록화제거되어야 한다고 블록화제거 결정 유닛 (104) 이 결정하는 경우, 블록화제거 결정 유닛 (104) 은 블록화제거 필터링 유닛 (106) 으로 하여금 에지를 블록화제거하기 위하여 그 에지 근처의 화소들에 대한 값들을 필터링하게 할 수도 있다.

[0159] 블록화제거 필터링 유닛 (106) 은, 블록화제거 결정 유닛 (104) 에 의해 나타난 바와 같이, 블록화제거될 에지들에 대해 블록화제거 필터 파라미터들 (108) 로부터 블록화제거 필터들의 정의들을 취출한다. 대체로, 에지의 필터링은 블록화제거될 현재 에지의 이웃으로부터의 화소들의 값들을 사용한다. 그러므로, 블록화제거 결정 기능부들 및 블록화제거 필터들 양쪽 모두는 에지의 양쪽에서 특정한 지원 지역을 가질 수도 있다. 블록화제거 필터를 에지 근처의 화소들에 적용함으로써, 블록화제거 필터링 유닛 (106) 은 에지 근처의 고 빈도의 트랜지션들이 약화되도록 화소들의 값들을 평활화 (smoothing) 할 수도 있다. 이런 방식으로, 에지 근처의 화소들에 대한 블록화제거 필터들의 적용은 에지 근처의 블록현상 아티팩트들을 감소시킬 수도 있다.

- [0160] 도 5는 서브블록들 (130 및 132) 사이의 비디오 블록의 에지 (134) 근처의 화소 포지션들을 예시하는 개념도이다. 하나의 예로서, 에지 (134)는 CU에서 정의된 2개의 TU들 사이의 TU 에지, 또는 CU에서 정의된 2개의 PU들 사이의 PU 에지와 같은 내부 CU 에지를 포함할 수도 있다. 화소 포지션들의 각각은 포맷 $[p|q]I_J$ 를 사용하여 지정되며, 여기서 p 는 서브블록 (130)에 대응하고 q 는 서브블록 (132)에 대응하며, I 는 에지 (134)로부터의 거리에 대응하고, J 는 서브블록들 (130 및 132)의 상단부터 하단까지의 행 표시자에 대응한다. 일부 예들에서, 블록화제거 결정 기능들 및 블록화제거 필터들을 위해 사용된 지원은 8개 화소들의 라인을 가진다. 이러한 예들에서, $0 \leq X \leq 7$ 인 주어진 라인 X 에 대해, 화소들 ($p3_X$ 내지 $q3_X$)의 각각은 지원으로서 사용될 수도 있다.
- [0161] 도 6은 본 개시물에서 설명된 기법들에 따른 현재 비디오 슬라이스에 대해 감소된 비트스트림 오버헤드로 블록화제거 필터 파라미터들을 인코딩하는 일 예의 동작을 도시하는 흐름도이다. 도 6에서 예시된 동작은 도 2에서 비디오 인코더 (20)에 관해 설명되어 있다.
- [0162] 비디오 인코더 (20)의 엔트로피 인코딩 유닛 (56)은 화상 계층 파라미터 세트를 참조하는 화상들에 대해 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 정의된 제 1 선택스 엘리먼트를 인코딩한다 (140). 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않으면 (141의 "아니오" 분기), 엔트로피 인코딩 유닛 (56)은 슬라이스 헤더에서의 제 2 선택스 엘리먼트의 인코딩을 제거한다 (142). 제 2 선택스 엘리먼트는 블록화제거 필터 파라미터들 중 어떤 세트를 현재 비디오 슬라이스에 대한 블록화제거 필터를 정의하는데 사용할지를 나타내도록 정의된다. 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 하나에만 존재하는 경우에, 어떤 블록화제거 필터 파라미터들을 블록화제거 필터를 정의하는데 사용할지에 관해 결정이 이루어질 필요가 없기 때문에 제 2 선택스 엘리먼트는 불필요하다. 대신, 블록화제거 필터 (63)는 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 어느 하나에 존재하는 블록화제거 필터 파라미터들의 단일 세트에 기초하여 현재 비디오 슬라이스에 대해 정의된다 (144).
- [0163] 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하면 (141의 "예" 분기), 엔트로피 인코딩 유닛 (56)은 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 인코딩하는데, 그 제 2 선택스 엘리먼트는 화상 계층 파라미터 세트에 포함된 블록화제거 파라미터들의 제 1 세트를 사용할지 또는 슬라이스 헤더에 포함된 블록화제거 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된다 (146). 이 경우, 블록화제거 필터 (63)는 블록화제거 파라미터들의 나타내어진 세트에 기초하여 현재 비디오 슬라이스에 대해 정의된다 (148). 그러므로, 본 개시물에서 설명된 기법들에 따르면, 블록화제거 필터 파라미터들은, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는 경우, 슬라이스 헤더에서의 제 2 선택스 엘리먼트만을 인코딩함으로써 현재 비디오 슬라이스에 대해 감소된 비트스트림 오버헤드로 인코딩된다.
- [0164] 도 7은 본 개시물에서 설명된 기법들에 따른 현재 비디오 슬라이스에 대해 감소된 비트스트림 오버헤드로 블록화제거 필터 파라미터들을 디코딩하는 일 예의 동작을 도시하는 흐름도이다. 도 7에서 예시된 동작은 도 3에서 비디오 디코더 (30)에 관해 설명되어 있다.
- [0165] 비디오 디코더 (30)의 엔트로피 디코딩 유닛 (80)은 화상 계층 파라미터 세트를 참조하는 화상들에 대해 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는지의 여부를 나타내도록 구성된 제 1 선택스 엘리먼트를 디코딩한다 (150). 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하지 않으면 (151의 "아니오" 분기), 엔트로피 디코딩 유닛 (80)은 제 2 선택스 엘리먼트가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정한다 (152). 제 2 선택스 엘리먼트는 블록화제거 필터 파라미터들 중 어떤 세트를 현재 비디오 슬라이스에 대한 블록화제거 필터를 정의하는데 사용할지를 나타내도록 정의된다. 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 하나에만 존재하는 경우에, 어떤 블록화제거 필터 파라미터들을 블록화제거 필터를 정의하는데 사용할지에 관해 결정이 이루어질 필요가 없기 때문에 제 2 선택스 엘리먼트는 불필요하다. 대신, 블록화제거 필터 (91)는 화상 계층 파라미터 세트 또는 슬라이스 헤더 중 어느 하나에 존재하는 블록화제거 필터 파라미터들의 단일 세트에 기초하여 현재 비디오 슬라이스에 대해 정의된다 (154).
- [0166] 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하면 (151의 "예" 분기), 엔트로피 디코딩 유닛 (80)은 슬라이스 헤더에서의 제 2 선택스 엘리먼트를 디코딩하는데, 그 제 2 선택스 엘리먼트는 화상 계층 파라미터 세트에 포함된 블록화제거 파라미터들의 제 1 세트를 사용할지 또는 슬라이스

이스 헤더에 포함된 블록화제거 파라미터들의 제 2 세트를 사용할지를 나타내도록 정의된다 (156). 이 경우, 블록화제거 필터 (91)는 블록화제거 파라미터들의 나타내어진 세트에 기초하여 현재 비디오 슬라이스에 대해 정의된다 (158). 그러므로, 본 개시물에서 설명된 기법들에 따르면, 블록화제거 필터 파라미터들은, 블록화제거 필터 파라미터들이 화상 계층 파라미터 세트 및 슬라이스 헤더 양쪽 모두에 존재하는 경우, 슬라이스 헤더에서의 제 2 선택스 엘리먼트만을 디코딩함으로써 현재 비디오 슬라이스에 대해 감소된 비트스트림 오버헤드로 디코딩된다.

[0167] 도 8은 슬라이스 헤더에서의 블록화제거 필터 파라미터들에 의해 오버라이드될 수도 있는 화상 파라미터 세트 (PPS)에서의 현재 비디오 슬라이스에 대한 블록화제거 필터 파라미터들을 인코딩하는 일 예의 동작을 도시하는 흐름도이다. 도 8에서 예시된 동작은 도 2에서 비디오 인코더 (20)에 관해 설명되어 있다.

[0168] 엔트로피 인코딩 유닛 (56)은 임의의 블록화제거 필터 제어 선택스 엘리먼트들이 PPS 및 슬라이스 헤더에 존재하는지의 여부를 나타내도록 정의된 제어 존재 선택스 엘리먼트를 PPS에서 인코딩한다 (160). 블록화제거 필터 제어 선택스 엘리먼트들은, PPS에서 시그널링된 오버라이드 가능 플래그, 슬라이스 헤더에서 시그널링된 오버라이드 플래그, 및 PPS 및/또는 슬라이스 헤더에서 시그널링된 블록화제거 필터 파라미터들을 포함한다. 블록화제거 필터 제어 선택스 엘리먼트들이 PPS 또는 슬라이스 헤더 중 어느 하나에 존재하지 않는 경우 (162의 "아니오" 분기), 엔트로피 인코딩 유닛 (56)은 블록화제거 필터 제어 선택스 엘리먼트들 중 임의의 것의 인코딩을 제거한다 (164). 이 경우, 비디오 인코더 (20)는 블록화제거 필터 (63)가 코딩된 블록화제거 필터 파라미터들에 기초하여 정의되지 않음을 비디오 디코더 (30)로 시그널링한다. 대신, 블록화제거 필터 (63)는 디폴트 블록화제거 필터 파라미터들에 기초하여 정의된다 (166).

[0169] 블록화제거 필터 제어 선택스 엘리먼트들이 PPS 또는 슬라이스 헤더에 존재하는 경우 (162의 "예" 분기), 엔트로피 인코딩 유닛 (56)은 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트에 의한 PPS에 포함된 블록화제거 필터 파라미터들의 제 1 세트의 오버라이드가 가능하게 되는지의 여부를 나타내는 PPS에서의 오버라이드 가능 플래그를 인코딩한다 (168).

[0170] PPS에서의 블록화제거 필터 파라미터들의 오버라이드가 가능하지 않게 되면 (170의 "아니오" 분기), 엔트로피 인코딩 유닛 (56)은 슬라이스 헤더에서의 오버라이드 플래그의 인코딩을 제거한다 (172). 그 오버라이드 플래그는 블록화제거 필터 파라미터들 중 어떤 세트를 현재 비디오 슬라이스에 대한 블록화제거 필터를 정의하는데 사용할지를 나타낸다. 블록화제거 필터 파라미터들이 PPS에만 존재하는 경우에, 어떤 블록화제거 필터 파라미터들을 블록화제거 필터를 정의하는데 사용할지에 관해 결정이 이루어질 필요가 없기 때문에 슬라이스 헤더에서의 오버라이드 플래그는 불필요하다. 대신, 블록화제거 필터 (63)는 PPS에 존재하는 블록화제거 필터 파라미터들에 기초하여 현재 비디오 슬라이스에 대해 정의된다 (174).

[0171] 슬라이스 헤더에서의 블록화제거 파라미터들에 의한 PPS에서의 블록화제거 필터 파라미터들의 오버라이드가 가능하게 되면 (170의 "예" 분기), 엔트로피 인코딩 유닛 (56)은 PPS로부터의 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 블록화제거 필터 파라미터들의 제 1 세트를 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트로 오버라이드할지를 나타내는 슬라이스 헤더에서의 오버라이드 플래그를 인코딩한다 (176). 이 경우, 블록화제거 필터 (63)는 블록화제거 파라미터들의 나타내어진 세트에 기초하여 현재 비디오 슬라이스에 대해 정의된다 (178).

[0172] 도 9는 슬라이스 헤더에서의 블록화제거 필터 파라미터들에 의해 오버라이드될 수도 있는 화상 파라미터 세트 (PPS)에서의 현재 비디오 슬라이스에 대한 블록화제거 필터 파라미터들을 디코딩하는 일 예의 동작을 도시하는 흐름도이다. 도 9에서 예시된 동작은 도 3에서 비디오 디코더 (30)에 관해 설명되어 있다.

[0173] 엔트로피 디코딩 유닛 (80)은 임의의 블록화제거 필터 제어 선택스 엘리먼트들이 PPS 및 슬라이스 헤더에 존재하는지의 여부를 나타내도록 정의된 제어 존재 선택스 엘리먼트를 PPS에서 디코딩한다 (180). 블록화제거 필터 제어 선택스 엘리먼트들은, PPS에서 시그널링된 오버라이드 가능 플래그, 슬라이스 헤더에서 시그널링된 오버라이드 플래그, 및 PPS 및/또는 슬라이스 헤더에서 시그널링된 블록화제거 필터 파라미터들을 포함한다. 블록화제거 필터 제어 선택스 엘리먼트들이 PPS 또는 슬라이스 헤더 중 어느 하나에 존재하지 않는 경우 (182의 "아니오" 분기), 엔트로피 디코딩 유닛 (80)은 디코딩될 블록화제거 필터 제어 선택스 엘리먼트들이 비트스트림에 존재하지 않는다고 결정한다 (184). 이 경우, 비디오 디코더 (30)는 블록화제거 필터 (91)가 코딩된 블록화제거 필터 파라미터들에 기초하여 정의되지 않음을 알게 된다. 대신, 블록화제거 필터 (91)는 디폴트 블록화제거 필터 파라미터들에 기초하여 정의된다 (186).

- [0174] 블록화제거 필터 제어 신택스 엘리먼트들이 PPS 또는 슬라이스 헤더에 존재하는 경우 (182의 "예" 분기), 엔트로피 디코딩 유닛 (80) 은 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트에 의한 PPS에 포함된 블록화제거 필터 파라미터들의 제 1 세트의 오버라이드가 가능하게 되는지의 여부를 나타내는 PPS에서의 오버라이드 가능 플래그를 디코딩한다 (188).
- [0175] PPS에서의 블록화제거 필터 파라미터들의 오버라이드가 가능하지 않게 되면 (190의 "아니오" 분기), 엔트로피 디코딩 유닛 (80) 은 오버라이드 플래그가 디코딩될 슬라이스 헤더에 존재하지 않는다고 결정한다 (192). 그 오버라이드 플래그는 블록화제거 필터 파라미터들 중 어떤 세트를 현재 비디오 슬라이스에 대한 블록화제거 필터를 정의하는데 사용할지를 나타낸다. 블록화제거 필터 파라미터들이 PPS에만 존재하는 경우에, 어떤 블록화제거 필터 파라미터들을 블록화제거 필터를 정의하는데 사용할지에 관해 결정이 이루어질 필요가 없기 때문에 슬라이스 헤더에서의 오버라이드 플래그는 불필요하다. 대신, 블록화제거 필터 (91) 는 PPS에 존재하는 블록화제거 필터 파라미터들에 기초하여 현재 비디오 슬라이스에 대해 정의된다 (194).
- [0176] 슬라이스 헤더에서의 블록화제거 파라미터들에 의한 PPS에서의 블록화제거 필터 파라미터들의 오버라이드가 가능하게 되면 (190의 "예" 분기), 엔트로피 디코딩 유닛 (80) 은 PPS로부터의 블록화제거 필터 파라미터들의 제 1 세트를 사용할지 또는 블록화제거 필터 파라미터들의 제 1 세트를 슬라이스 헤더에 포함된 블록화제거 필터 파라미터들의 제 2 세트로 오버라이드할지를 나타내는 슬라이스 헤더에서의 오버라이드 플래그를 디코딩한다 (196). 이 경우, 블록화제거 필터 (91) 는 블록화제거 파라미터들의 나타내어진 세트에 기초하여 현재 비디오 슬라이스에 대해 정의된다 (198).
- [0177] 블록화제거 필터 파라미터들의 세트가 APS에 포함되는 위에서 설명된 예들에서, APS는 ALF, SAO 또는 블록화제거에 대해 파라미터들이 변화하면 언제든지 비디오 디코더 (30) 로 시그널링될 수도 있다. ALF 및 SAO 파라미터들은 아마 블록화제거 필터 파라미터들보다 더 빈번하게 변화한다. 이 경우, SAO 또는 ALF 파라미터들이 변화할 때마다 동일한 블록화제거 필터 파라미터들이 APS에 반복되는 경우 비트들이 낭비될 수도 있다. APS 비트스트림 오버헤드를 감소시키는 하나의 해법으로서, 일부 예들에서, 기법들은 ALF 및 SAO 파라미터들과는 별개로 블록화제거 필터 파라미터들을 업데이트하기 위해 다수의 APS들을 도입한다. 후속 APS들에서의 상수 블록화제거 파라미터들의 사본들을 시그널링하는 것을 피하기 위해, 슬라이스 헤더는 슬라이스들의 비디오 데이터를 디코딩하기 위해 다수의 APS들을 참조할 수도 있다. 다수의 APS들을 사용하여 블록화제거 필터 파라미터들을 시그널링하는 여러 선택사항들은 아래에서 설명되는 바와 같다. APS들 및 슬라이스 헤더들의 신택스들은 취소선 (strikethrough) 들이 APS 및 슬라이스 헤더 신택스로부터 제거될 수도 있는 신택스 엘리먼트들 및 조건들을 나타내는 아래의 표들에서 제시된다.
- [0178] 제 1 예로서, 슬라이스 헤더는 각각의 유효한 APS에 대한 고유 APS 식별자들, 예컨대, `aps_id[i]`를 사용하여 다수의 상이한 APS들을 참조할 수도 있다. 이 경우, 그 기법들은 다수의 APS 접근법에서의 블록화제거 필터 파라미터들을 포함하며, 예컨대, 여기서 다수의 APS 접근법은 『M. Li, P. Wu, "Multiple Adaptation Parameter Sets Referring," 7th JCT-VC Meeting, Geneva, Switzerland, Nov. 2011, Doc. JCTVC-G332』에서 제안되는 것과 유사할 수도 있다.

[0179] 표 15. APS 식별자들을 갖는 슬라이스 헤더 선택스

slice_header() {	디스크립터
.....	
lightweight_slice_flag	u(1)
if(!lightweight_slice_flag) {	
slice_type	ue(v)
pic_parameter_set_id	ue(v)
if(sample_adaptive_offset_enabled_flag adaptive_loop_filter_enabled_flag deblocking_filter_in_aps_enabled_flag) {	
number_of_valid_aps	u(v)
if (number_of_valid_aps != 0) {	
for (i = 0; i < number_of_valid_aps; i++) {	
aps_id[i]	ue(v)
}	
}	
}	
.....	
}	
.....	
}	

[0180]

[0181] 표 15의 슬라이스 헤더 선택스에 대한 의미체계는 다음과 같이 정의된다. 0과 동일한 **deblocking_filter_enabled_flag**는 블록화제거 필터가 가능하지 않게 됨을 의미하고, 1과 동일한 **deblocking_filter_enabled_flag**는 블록화제거 필터가 가능하게 됨을 의미한다. **number_of_valid_aps** 선택스 엘리먼트는 슬라이스를 디코딩하기 위한 유효한 APS들의 수를 특정한다. **number_of_valid_aps**의 값은 [0, MaxNumberValidAps]의 범위에 있을 것이다. MaxNumberValidAps의 값은 프로파일/레벨에서 특정된다. **number_of_valid_aps**가 존재하지 않으면, 그것의 값은 0일 것이라고 유추된다. **number_of_valid_aps**가 0과 동일하면, 액티브 APS들이 없다고 유추된다. **aps_id[i]**는 슬라이스를 디코딩하기 위한 유효한 APS ID들을 특정하며, 여기서 i는 [0, **number_of_valid_aps**-1]의 범위에 있을 것이다.

[0182] 위의 슬라이스 헤더 선택스에 따르면, 잠재적으로 다수의 APS들에 따른 슬라이스에 대한 디코딩 규칙들은 다음과 같다. **number_of_valid_aps** 값이 0과 동일하거나 또는 존재하지 않는, 즉, 슬라이스 헤더에서 참조되는 APS가 없는 경우, 이 슬라이스의 디코딩 시에 **aps_sample_adaptive_offset_flag** 및 **aps_adaptive_loop_filter_flag** 및 **aps_deblocking_filter_flag**의 플래그들은 0으로 설정될 것이고, SAO도 ALF도 그 슬라이스의 디코딩 시에 적용되지 않을 것이다. **inherit_dbl_params_from_APS_flag**의 값이 0과 동일하면, 블록화제거 필터는 이 슬라이스의 디코딩 시에 적용될 것이고 이 슬라이스 헤더에 포함된 블록화제거 파라미터들은 사용될 것이다.

[0183] **number_of_valid_aps** 값이 1과 동일한, 즉, 하나의 APS만이 슬라이스 헤더에서 참조되는 경우, 이 슬라이스의 디코딩 시에 **aps_sample_adaptive_offset_flag** 및 **aps_adaptive_loop_filter_flag** 및 **aps_deblocking_filter_flag**의 플래그들과 SAO 및 ALF의 도구 파라미터들은 이 참조된 APS에서 제시된 대응하는 선택스 엘리먼트들의 값들과 동일하게 설정될 것이다. **inherit_dbl_params_from_APS_flag**의 값이 1과 동일하면, 블록화제거 파라미터들은 참조되는 APS에서의 대응하는 파라미터들의 값과 동일하게 설정될 것이다.

[0184] 블록화제거 필터에 대해, **number_of_valid_aps**가 1보다 큰 경우, 즉, 다수의 APS들의 접근법이 적용되는 경우, 참조된 APS들에서의 모든 **aps_deblocking_filter_flag** 엘리먼트들이 0과 동일하면, 이 슬라이스의 디코딩 시의 **aps_deblocking_filter_flag**는 0으로 설정될 것이고, **inherit_dbl_params_from_APS_flag**가 0과 동일하면, 블록화제거 필터는 이 슬라이스 헤더에 존재하는 블록화제거 파라미터들을 사용하여 이 슬라이스에 적용될 것이다. 그렇지 않고, 하나의 및 오직 하나의 APS가 1과 동일한 **aps_deblocking_filter_flag**를 포함하면, 이 슬라이스의 디코딩 시의 **aps_deblocking_filter_flag**는 1로 설정될 것이고, **inherit_dbl_params_from_APS_flag**가 1과 동일하면, 이 APS로부터 파생된 블록화제거 파라미터들을 갖는 블록화제거 필터가 이 슬라이스의 디코딩 시에 적용될 것이다. 그렇지 않고, 1과 동일한 **aps_deblocking_filter_flag**를 포함하는 하나를 초과하는 APS가 있으면, 이 슬라이스의 디코딩 시의 **aps_deblocking_filter_flag**는 0으로 설정될 것이고, **inherit_dbl_params_from_APS_flag**가 0과 동일하면, 블록화제거 필터는 이 슬라이스 헤더에 존재하는 블록화제거

거 파라미터들을 사용하여 이 슬라이스에 적용될 것이다.

[0185] 본 개시물의 기법들에 따르면, `aps_deblocking_filter_flag`는, 아래의 표 16에 제공된 바와 같이, 블록화제거 파라미터들이 APS에 존재함을 나타낼 수 있다.

[0186] 표 16. 상속 가능 플래그를 갖는 APS 선택스

<code>aps_rbsp() {</code>	디스크립터
<code>aps_id</code>	<code>ue(v)</code>
<code>aps_deblocking_filter_flag</code>	<code>u(1)</code>
<code>aps_sample_adaptive_offset_data_present_flag</code>	<code>u(1)</code>
<code>aps_adaptive_loop_filter_data_present_flag</code>	<code>u(1)</code>
<code>if(aps_sample_adaptive_offset_data_present_flag </code> <code>aps_adaptive_loop_filter_data_present_flag) {</code>	
<code>aps_cabac_use_flag</code>	<code>u(1)</code>
<code>if(aps_cabac_use_flag) {</code>	
<code>aps_cabac_init_idc</code>	<code>ue(v)</code>
<code>aps_cabac_init_qp_minus26</code>	<code>se(v)</code>
<code>}</code>	
<code>}</code>	
<code>if(aps_deblocking_filter_flag){</code>	
<code>disable_deblocking_filter_flag</code>	<code>u(1)</code>
<code>if(!disable_deblocking_filter_flag) {</code>	
<code>beta_offset_div2</code>	<code>se(v)</code>
<code>tc_offset_div2</code>	<code>se(v)</code>
<code>}</code>	
<code>}</code>	
<code>if(aps_adaptive_loop_filter_data_present_flag) {</code>	
<code>aps_adaptive_loop_filter_flag</code>	<code>u(1)</code>
<code>if(aps_adaptive_loop_filter_flag) {</code>	
<code>alf_data_byte_count</code>	<code>u(8)</code>
<code>alf_param()</code>	
<code>byte_align()</code>	
<code>}</code>	
<code>}</code>	
<code>if(aps_sample_adaptive_offset_data_present_flag) {</code>	
<code>aps_sample_adaptive_offset_flag</code>	<code>u(1)</code>
<code>if(aps_sample_adaptive_offset_flag) {</code>	
<code>sao_data_byte_count</code>	<code>u(8)</code>
<code>byte_align()</code>	
<code>sao_param()</code>	
<code>}</code>	
<code>}</code>	
<code>rbps_trailing_bits()</code>	
<code>}</code>	

[0187]

[0188] 표 16의 APS 선택스에 대한 의미체계는 다음과 같이 정의된다. 1과 동일한 `aps_sample_adaptive_offset_data_present_flag`는 SAO 파라미터들이 이 APS에 존재함을 특정하고, 0과 동일한 `aps_sample_adaptive_offset_data_present_flag`는 SAO 파라미터들이 이 APS에 존재하지 않음을 특정하며, 여기서 SAO 가능 플래그가 1인 경우 SAO 파라미터들은 SAO 가능 플래그 및 SAO 파라미터들을 참조한다.

[0189] 1과 동일한 `aps_sample_adaptive_offset_flag`는 현재 APS를 참조하는 슬라이스들에 대해 SAO가 온임을 특정하고, 0과 동일한 `aps_sample_adaptive_offset_flag`는 현재 APS를 참조하는 슬라이스들에 대해 SAO가 오프임을 특정한다. 액티브 APS가 없거나 또는 `aps_sample_adaptive_offset_flag` 값이 0과 동일하면,

aps_sample_adaptive_offset_flag 값은 0이라고 유추된다.

[0190] 1과 동일한 **aps_adaptive_loop_filter_data_present_flag**는 ALF 파라미터들이 이 APS에 존재함을 특정하고, 0과 동일한 **aps_adaptive_loop_filter_data_present_flag**는 ALF 파라미터들이 APS에 존재하지 않음을 특정하며, 여기서 ALF 가능 플래그가 1인 경우 ALF 파라미터들은 ALF 가능 플래그 및 ALF 파라미터들을 참조한다. 1과 동일한 **aps_adaptive_loop_filter_flag**는 현재 APS를 참조하는 슬라이스들에 대해 ALF가 온임을 특정하고, 0과 동일한 **aps_adaptive_loop_filter_flag**는 현재 APS를 참조하는 슬라이스들에 대해 ALF가 오프임을 특정한다. 액티브 APS가 없거나 또는 **aps_adaptive_loop_filter_data_present_flag** 값이 0과 동일하면, **aps_adaptive_loop_filter_flag** 값은 0이라고 유추된다.

[0191] 위의 APS 선택스에 따르면, 잠재적으로 다수의 APS들에 따른 슬라이스에 대한 디코딩 규칙들은 다음과 같다. **number_of_valid_aps**가 1보다 큰 경우 (이전과는 다른 경우), **aps_id[0]**과 동일한 그것의 ID를 갖는 APS는 현재 슬라이스의 디코딩 시에 기본 APS로서 가정적으로 취해질 것인 반면, **aps_id[1]**, **aps_id[2]**, ... **aps_id[number_valid_aps - 1]**과 동일한 그것들의 ID들을 갖는 다른 APS들은 보정된 APS들로서 취해질 수 있다. **aps_id[i]** ($i > 0$)의 그것의 APS ID를 갖는 보정된 APS들에서의 제시된 정보는, **aps_deblocking_filter_flag**가 1과 동일하고 및/또는 **aps_sample_adaptive_offset_data_present_flag**가 1이고 및/또는 **aps_adaptive_loop_filter_data_present_flag**가 1인 경우 APS에 존재하는 도구 파라미터들 (즉, 블록 화제거 필터에 대한 **aps_deblocking_filter_flag** 및 블록화제거 파라미터들 및/또는 SAO에 대한 **aps_sample_adaptive_offset_flag** 및 **sao_param()** 및/또는 ALF에 대한 **aps_adaptive_loop_filter_flag** 및 **alf_param()**)을 참조하는 것인데, **aps_id[i-1]**, ..., **aps_id[0]**의 그것들의 APS ID를 갖는 APS들에 의해 이전에 그리고 일시적으로 결정된 대응하는 도구들의 정보를 덮어쓴다. 그리하여, **aps_deblocking_filter_flag**의 및 블록화제거 파라미터들의 최종적으로 획득된 값들은 (**inherit_dbl_params_from_APS_flag**의 값에 추가적으로 의존하여) 현재 슬라이스를 블록화제거하는데 사용된다. 최종적으로 획득된 **aps_sample_adaptive_offset_flag** 및 **sao_param()**에서의 파라미터들은 현재 슬라이스의 디코딩 시에 SAO를 구현하는데 사용되고, 최종적으로 획득된 **aps_adaptive_loop_filter_flag** 및 **alf_param()**에서의 파라미터들은 현재 슬라이스의 디코딩 시에 ALF를 구현하는데 사용된다.

[0192] 다르게 말하면, 다수의 APS들이 슬라이스에 의해 참조되는 경우, 각각의 도구에 대해, APS ID들의 프레젠테이션 순서에 따르면, 1과 동일한 이 도구의 **data_present_flag** (즉 **aps_deblocking_filter_flag** 또는 **aps_sample_adaptive_offset_data_present_flag** 또는 **aps_adaptive_loop_filter_data_present_flag**)를 포함하는 마지막 APS는 이 슬라이스를 디코딩하기 전에 이 도구를 초기화하기 위해 활성화된다.

[0193] 다수의 APS들을 사용하여 블록화제거 필터 파라미터들을 시그널링하는 제 2의 예로서, 슬라이스 헤더는 APS의 각각의 유형에 대한 고유 서브APS 식별자들을 사용하여 다수의 상이한 서브APS들을 참조할 수도 있다. 이 경우, 기법들은 『J. Tanaka, Y. Morigami, T. Suzuki, "Non-CE4 Subtest3: Extension of Adaptation Parameter Sets syntax for Quantization matrix," 7th JCT-VC Meeting, Geneva, Switzerland, Nov. 2011, Doc. JCTVC-G295』에서 제안된 것과 같은 서브APS 접근법에서의 블록화제거 필터 파라미터들을 포함한다.

[0194] 본 개시물의 기법들에 따르면, 다음 표 17 및 표 18에서와 같이 APS에 **aps_dbl_id**를 포함하는 것이 제안된다. **inherit_dbl_params_from_APS_flag=1**이면 블록화제거 파라미터들은 **aps_dbl_id**를 갖는 APS로부터 복사된다.

[0195] 표 17. 상속 가능 플래그 및 조건부 코딩된 블록화제거 APS ID를 갖는 APS 섹스

aps_rbsp() {	디스크립터
aps_id	ue(v)
aps_deblocking_filter_flag	u(1)
aps_sample_adaptive_offset_flag	u(1)
aps_adaptive_loop_filter_flag	u(1)
....	
if (aps_deblocking_filter_flag){	
aps_dbl_id	ue(v)
disable_deblocking_filter_flag	u(1)
if (!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
if(aps_sample_adaptive_offset_flag) {	
aps_sao_id	ue(v)
sao_data_byte_count	u(8)
byte_align ()	
sao_param()	
byte_align ()	
}	
if(aps_adaptive_loop_filter_flag) {	
aps_alf_id	ue(v)
alf_data_byte_count	u(8)
byte_align ()	
alf_param()	
byte_align()	
}	
rbsp_trailing_bits()	
}	

[0196]

[0197] 표 18. 서브APS 식별자들을 갖는 슬라이스 헤더 선택스

slice_header() {	디스크립터
entropy_slice_flag	u(1)
if(!entropy_slice_flag) {	
slice_type	ue(v)
pic_parameter_set_id	ue(v)
if(sample_adaptive_offset_enabled_flag adaptive_loop_filter_enabled_flag)	
aps_id	ue(v)
if(deblocking_filter_in_aps_enabled_flag)	
aps_dbl_id	ue(v)
if(sample_adaptive_offset_enabled_flag)	
aps_sao_id	ue(v)
if(adaptive_loop_filter_enabled_flag)	
aps_alf_id	ue(v)
if(quantization_matrix_enabled_flag)	
aps_qmatrix_id	ue(v)
frame_num	u(v)
.....	
if(!entropy_slice_flag) {	
slice_qp_delta	se(v)
inherit_dbl_params_from_APS_flag	u(1)
if(!inherit_dbl_params_from_APS_flag) {	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
if(slice_type == B)	
collocated_from_l0_flag	u(1)
if(adaptive_loop_filter_enabled_flag && aps_adaptive_loop_filter_flag) {	
byte_align()	
alf_cu_control_param()	
byte_align()	
}	
if((weighted_pred_flag && slice_type == P)	
(weighted_bipred_idc == 1 && slice_type == B))	
pred_weight_table()	
}	
if(slice_type == P slice_type == B)	
5_minus_max_num_merge_cand	ue(v)
for(i = 0; i < num_substreams_minus1 + 1; i++) {	
substream_length_mode	u(2)
substream_length[i]	u(v)
}	
}	

[0198]

[0199] 위의 슬라이스 헤더 선택스에 대한 문제는 심지어 inherit_dbl_params_from_APS_flag=0인 경우에도 aps_dbl_id가 시그널링된다는 것이다. 대체예로서, aps_dbl_id는 아래의 표 19에서 제공된 바와 같이 슬라이스 헤더에서 시그널링될 수도 있다.

[0200] 표 19. 조건부로 코딩된 서브APS 식별자들을 갖는 슬라이스 헤더 선택스

slice_header() {	디스크립터
entropy_slice_flag	u(1)
if(!entropy_slice_flag) {	
slice_type	ue(v)
pic_parameter_set_id	ue(v)
if(sample_adaptive_offset_enabled_flag adaptive_loop_filter_enabled_flag)	
aps_id	ue(v)
if(sample_adaptive_offset_enabled_flag)	
aps_sao_id	ue(v)
if(adaptive_loop_filter_enabled_flag)	
aps_alf_id	ue(v)
if(quantization_matrix_enabled_flag)	
aps_qmatrix_id	ue(v)
frame_num	u(v)
.....	
if(!entropy_slice_flag) {	
slice_qp_delta	se(v)
inherit_dbl_params_from_APS_flag	u(1)
if(inherit_dbl_params_from_APS_flag) {	
aps_dbl_id	ue(v)
}	
if(!inherit_dbl_params_from_APS_flag) {	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
.....	

[0201]

[0202]

다수의 APS들을 사용하여 블록화제거 필터 파라미터들을 시그널링하는 제 3의 예로서, 슬라이스 헤더는 링크형 리스트 (linked-list) APS들을 사용하여 다수의 상이한 APS들을 참조할 수도 있다. 이 경우, 기법들은, 예컨대, 『M. Li, P. Wu, S. Wenger, J. Boyce, "APS Referencing," 8th JCT-VC Meeting, San Jose, Feb. 2011』에서 설명된 바와 같은, 링크형 리스트 APS들에 기초한 APS 참조 방법에서의 블록화제거 필터 파라미터들을 포함한다.

[0203]

APS 참조 문서는 그것이 또한 루프 필터 및 스케일링 리스트 파라미터들의 존재를 시그널링하기 위해 APS에서의 플래그들을 도입한다는 의미에서 JCTVC-G332에서 제안된 부분적 APS 업데이트 방법에 기초하여 작성된다. 덧붙여서, ref_aps_flag 및 ref_aps_id 선택스 엘리먼트들은 링크형 리스트 메커니즘을 통한 파라미터들의 부분 업데이트를 가능하게 하기 위해 APS에 도입된다. 본 개시물의 기법들에 따르면, aps_deblocking_filter_flag는, 아래의 표 20에 제공된 바와 같이, 블록화제거 파라미터들이 APS에 존재함을 나타낼 수 있다. 슬라이스 헤더의 선택스에 대한 연관된 변화들은 아래의 표 21에서 제공된다.

[0204] 표 20. 상속 가능 플래그 및 링크형 리스트 APS들을 갖는 APS 선택스

aps_rbsp() {	디스크립터
aps_id	ue(v)
ref_aps_flag	u(1)
if(ref_aps_flag) {	
ref_aps_id	ue(v)
}	
aps_deblocking_filter_flag	u(1)
aps_scaling_list_data_present_flag	u(1)
aps_sample_adaptive_offset_data_present_flag	u(1)
aps_adaptive_loop_filter_data_present_flag	u(1)
.....	
if(aps_deblocking_filter_flag){	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
if(aps_scaling_list_data_present_flag) {	
—— scaling_list_data_byte_count	u(8)
—— byte_align()	
scaling_list_param()	
—— byte_align()	
}	
if(aps_sample_adaptive_offset_data_present_flag) {	
—— sao_data_byte_count	u(8)
—— byte_align()	
sao_param()	
—— byte_align()	
}	
if(aps_adaptive_loop_filter_data_present_flag) {	
—— alf_data_byte_count	u(8)
—— byte_align()	
alf_param()	
}	
rbsp_trailing_bits()	
}	

[0205]

[0206] 표 21. 링크형 리스트 APS 식별자들을 갖는 슬라이스 헤더 선택스

slice_header() {	디스크립터
lightweight_slice_flag	u(1)
if(!lightweight_slice_flag) {	
slice_type	ue(v)
pic_parameter_set_id	ue(v)
if(scaling_list_enabled_flag sample_adaptive_offset_enabled_flag adaptive_loop_filter_enabled_flag deblocking_filter_in_aps_enabled_flag)	
aps_id	ue(v)
.....	
if(deblocking_filter_in_aps_enabled_flag && aps_deblocking_filter_flag) {	
inherit_dbl_params_from_APS_flag	u(1)
/* deblocking_filter_in_aps_enabled_flag = 0 또는 aps_deblocking_filter_flag=0 이면 inherit_dbl_params_from_APS_flag = 0이 유추된다는 것에 주의한다. */	
}	
if(!inherit_dbl_params_from_APS_flag){	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
}	
if(scaling_list_enabled_flag)	
scaling_list_flag	u(1)
if(sample_adaptive_offset_enabled_flag)	
sample_adaptive_offset_flag	u(1)
if(adaptive_loop_filter_enabled_flag)	
adaptive_loop_filter_flag	u(1)
}	
.....	
}	

[0207]

[0208] 표 20 및 표 21의 APS 및 슬라이스 헤더 선택스들에 대한 의미체계들은 다음과 같이 정의된다. **aps_id**는 슬라이스 헤더에 의해 또는 다른 적응 파라미터 세트에서의 ref_aps_id에 의해 참조되는 적응 파라미터 세트를 식별한다. **aps_id**의 값은 0 내지 7의 범위에 있을 것이다. 1과 동일한 **ref_aps_flag**는 이 적응 파라미터 세트가 다른 적응 파라미터 세트를 참조한다는 것을 특징한다. 0과 동일한 ref_aps_flag는 이 적응 파라미터 세트가 임의의 다른 적응 파라미터 세트를 참조하지 않는다는 것을 특징한다. **ref_aps_id**는 ref_aps_id와 동일한 **aps_id**를 갖는 이전의 적응 파라미터 세트가 이 적응 파라미터 세트에 의해 참조된다는 것을 특징한다.

[0209]

aps_deblocking_filter_flag는 블록화제거 파라미터들이 APS에 존재하거나 (1과 동일) 또는 존재하지 않는다 (0과 동일) 는 것을 나타낸다. 1과 동일한 **aps_scaling_list_data_present_flag**는 스케일링 리스트 파라미터들이 이 적응 파라미터 세트에 존재한다는 것을 특징하고, 0과 동일한 **aps_scaling_list_data_present_flag**는 스케일링 리스트 파라미터들이 이 적응 파라미터 세트에 존재하지 않는다는 것을 특징한다. 1과 동일한 **aps_sample_adaptive_offset_data_present_flag**는 SAO 파라미터들이 이 적응 파라미터 세트에 존재한다는 것을 특징하고, 0과 동일한 **aps_sample_adaptive_offset_data_present_flag**는 SAO 파라미터들이 이 적응 파라미터 세트에 존재하지 않는다는 것을 특징한다. 1과 동일한 **aps_adaptive_loop_filter_data_present_flag**는 ALF 파라미터들이 이 적응 파라미터 세트에 존재한다는 것을 특징하고, 0과 동일한 **aps_adaptive_loop_filter_data_present_flag**는 ALF 파라미터들이 이 적응 파라미터 세트에 존재하지 않는다는 것을 특징한다.

[0210]

1과 동일한 **scaling_list_flag**는 스케일링 매트릭스가 현재 슬라이스에 대해 적용된다는 것을 특징하고, 0과 동일한 **scaling_list_flag**는 스케일링 매트릭스가 현재 슬라이스에 대해 적용되지 않는다는 것을 특징한다.

scaling_list_flag_flag의 값은 현재 프레임에서의 모든 슬라이스들에 대해 동일할 것이다. 1과 동일한 adaptive_loop_filter_flag는 적응적 루프 필터가 현재 슬라이스에 대해 적용된다는 것을 특징하고, 0과 동일한 adaptive_loop_filter_flag는 적응적 루프 필터가 현재 슬라이스에 대해 적용되지 않는다는 것을 특징한다. adaptive_loop_filter_flag의 값은 현재 프레임에서의 모든 슬라이스들에 대해 동일할 것이다. 1과 동일한 sample_adaptive_offset_flag는 샘플 적응 오프셋이 현재 슬라이스에 대해 적용된다는 것을 특징하고, 0과 동일한 sample_adaptive_offset_flag는 샘플 적응 오프셋이 현재 슬라이스에 대해 적용되지 않는다는 것을 특징한다. sample_adaptive_offset_flag의 값은 현재 프레임에서의 모든 슬라이스들에 대해 동일할 것이다.

[0211] 다수의 APS들을 사용하여 블록화제거 필터 파라미터들을 시그널링하는 제 4 선택사항으로서, 슬라이스 헤더는 APS에서 특정된 블록화제거 필터 파라미터들의 부분 업데이트들을 포함할 수도 있다. 이 경우, 기법들은 블록화제거 필터 파라미터들의 부분 업데이트들을 포함한다. 이들 기법들은, 예를 들어, 『A. Minezawa, K. Sugimoto, S.-I. Sekiguchi, "On partial updating of APS parameters," 8th JCT-VC Meeting, San Jose, CA, USA, Feb. 2012, Doc. JCTVC-H0255』에서 설명된 슬라이스 헤더 시그널링을 사용하여 APS 파라미터들을 업데이트하는 방법에 적용될 수도 있다.

[0212] 본 개시물의 기법들은 슬라이스 헤더에 포함된 블록화제거 필터 조정 파라미터들에 기초하여 슬라이스 헤더에서의 aps_id에 의해 식별되는 APS에서의 블록화제거 필터 조정 파라미터들을 업데이트할 수도 있다. 그 기법들은, 아래의 표 22에서 제시된 바와 같이, APS에서의 블록화제거 필터 조정 파라미터들이 슬라이스 헤더에서의 블록화제거 필터 조정 파라미터들에 의해 업데이트될 경우를 나타내기 위해, 슬라이스 헤더에 update_dbl_params_in_APS_flag를 도입할 수도 있다.

[0213] 표 22. APS 플래그에서 업데이트 블록화제거 파라미터들을 갖는 슬라이스 헤더 선택스

slice_header() {	디스크립터
entropy_slice_flag	u(1)
if(!entropy_slice_flag) {	
slice_type	ue(v)
pic_parameter_set_id	ue(v)
if(sample_adaptive_offset_enabled_flag adaptive_loop_filter_enabled_flag)	
aps_id	ue(v)
....	
if(!entropy_slice_flag) {	
slice_qp_delta	se(v)
inherit_dbl_params_from_APS_flag	u(1)
update_dbl_params_in_APS_flag	u(1)
if((!inherit_dbl_params_from_APS_flag) update_dbl_params_in_APS_flag) {	
disable_deblocking_filter_flag	u(1)
if(!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
.....	
}	

[0214]

[0215] 표 22의 슬라이스 헤더 선택스에 대한 의미체계는 다음과 같이 정의된다. 1과 동일한 update_dbl_params_in_APS_flag는 aps_id와 동일한 ID를 갖는 APS에서의 블록화제거 필터 조정 파라미터들이 슬라이스 헤더에서의 블록화제거 필터 조정 파라미터들로 업데이트될 것이라는 것을 의미하고, 0과 동일한 update_dbl_params_in_APS_flag는 업데이트가 없을 것임을 의미한다.

[0216] 덧붙여서, 본 개시물의 기법들은 슬라이스 헤더에 포함된 블록화제거 필터 조정 파라미터들에 기초하여 슬라이스 헤더에서의 aps_dbl_id에 의해 식별되는 블록화제거 필터 조정 파라미터들을 APS에서 업데이트하는 것을 포함할 수도 있다. 슬라이스 헤더 블록화제거 필터 조정 파라미터들로 업데이트될 APS의 aps_dbl_id는 아래의 표 23에서 제시된 바와 같이, 슬라이스 헤더에서 따로따로 시그널링될 수도 있다.

[0217] 표 23. APS 플래그에서의 업데이트 블록화제거 파라미터들과 서브APS 식별자들을 갖는 슬라이스 헤더 선택스

slice_header() {	디스크립터
entropy_slice_flag	u(1)
if(!entropy_slice_flag) {	
slice_type	ue(v)
pic_parameter_set_id	ue(v)
if(sample_adaptive_offset_enabled_flag adaptive_loop_filter_enabled_flag)	
aps_id	ue(v)
....	
if(!entropy_slice_flag) {	
slice_qp_delta	se(v)
inherit_dbl_params_from_APS_flag	u(1)
update_dbl_params_in_APS_flag	u(1)
if(update_dbl_params_in_APS_flag)	
aps_dbl_id	
if ((!inherit_dbl_params_from_APS_flag) update_dbl_params_in_APS_flag) {	
disable_deblocking_filter_flag	u(1)
if (!disable_deblocking_filter_flag) {	
beta_offset_div2	se(v)
tc_offset_div2	se(v)
}	
}	
.....	
}	

[0218]

[0219]

하나 이상의 예들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 그것들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현된다면, 그 기능들은 하나 이상의 명령들 또는 코드로서 컴퓨터 판독가능 매체 상에 저장되거나 또는 송신될 수도 있고 하드웨어 기반 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터 판독가능 매체들은, 데이터 저장 매체들과 같은 유형의 (tangible) 매체에 대응하는 컴퓨터 판독가능 저장 매체들, 또는 예컨대 통신 프로토콜에 따라 한 장소에서 다른 장소로 컴퓨터 프로그램의 전달을 용이하게 하는 임의의 매체를 포함하는 통신 매체들을 포함할 수도 있다. 이런 방식으로, 컴퓨터 판독가능 매체들은 일반적으로 (1) 비일시적 (non-transitory) 인 유형의 (tangible) 컴퓨터 판독가능 저장 매체들 또는 (2) 신호 또는 반송파와 같은 통신 매체에 해당할 수도 있다. 데이터 저장 매체들은 본 개시물에서 설명된 기법들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 취출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 이용가능 매체들일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터 판독가능 매체를 포함할 수도 있다.

[0220]

비제한적인 예로, 이러한 컴퓨터 판독가능 저장 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광 디스크 스토리지, 자기 디스크 스토리지, 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 소망의 프로그램 코드를 컴퓨터에 의해 액세스될 수 있는 명령들 또는 데이터 구조들의 형태로 저장하는데 사용될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 접속이 컴퓨터 판독가능 매체로 적절히 칭해진다. 예를 들어, 명령들이 웹사이트, 서버, 또는 다른 원격 자원으로부터 동축 케이블, 광섬유 케이블, 연선 (twisted pair), 디지털 가입자 회선 (DSL), 또는 무선 기술들 이를테면 적외선, 라디오, 및/또는 마이크로파를 이용하여 송신된다면, 동축 케이블, 광섬유 케이블, 연선, DSL, 또는 적외선, 라디오, 및 마이크로파와 같은 무선 기술은 매체의 정의에 포함된다. 그러나, 컴퓨터 판독가능 저장 매체들 및 데이터 저장 매체들은 커백선들, 반송파들, 신호들, 또는 다른 일시적 매체들을 포함하지 않지만, 대신 비일시적 (non-transitory), 유형의 저장 매체들을 지향하고 있음이 이해되어야 한다. 디스크 (disk 및 disc) 는 본원에서 사용되는 바와 같이, 콤팩트 디스크 (compact disc, CD), 레이저 디스크, 광 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크 (floppy disk) 및 블루레이 디스크를 포함하는데, disk들은 보통 데이터를 자기적으로 재생하지만, disc들은 레이저들으로써 광적으로 데이터를 재생한다. 상기한 것들의 조합들은 또한 컴퓨터 판독가능 매체들의 범위 내에 포함되어야 한다.

[0221]

명령들은 하나 이상의 프로세서들, 이를테면 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로프로세서들, 주문형 집적회로들 (ASIC들), 필드 프로그램가능 로직 어레이들 (FPGA들), 또는 다른 동등한 집적 또는 개별 로직 회로에 의해 실행될 수도 있다. 따라서, 본원에서 사용되는 바와 같은 용어 "프로세서"는 앞서의

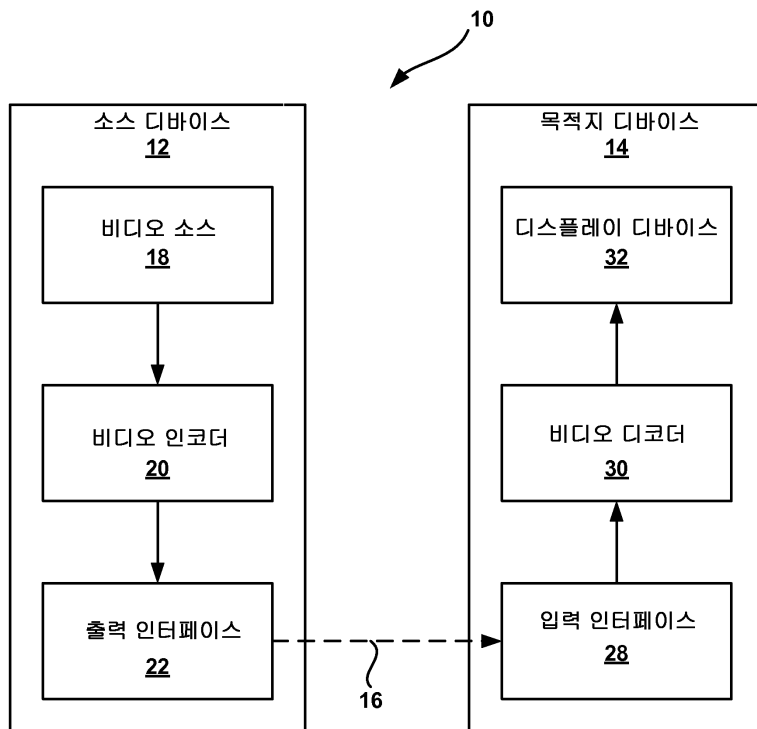
구조 또는 본원에서 설명된 기법들의 구현에 적합한 임의의 다른 구조 중 임의의 것을 말할 수도 있다. 덧붙여서, 일부 양태들에서, 본원에서 설명된 기능성은 인코딩 및 디코딩을 위해 구성되는, 또는 결합형 코덱(codec)으로 통합되는 전용 하드웨어 및/또는 소프트웨어 모듈들 내에 제공될 수도 있다. 또한, 본 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들 내에 완전히 구현될 수 있다.

[0222] 본 개시내용의 기법들은 무선 핸드셋, 집적회로(IC) 또는 한 세트의 IC들(예컨대, 칩 셋)을 포함하는 매우 다양한 디바이스들 또는 장치들로 구현될 수도 있다. 다양한 컴포넌트들, 모듈들, 또는 유닛들은 개시된 기법들을 수행하도록 구성된 디바이스들의 기능적 양태들을 강조하기 위해 본 개시물에서 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 요구하지는 않는다. 대신에, 위에서 설명된 바와 같이, 다양한 유닛들은 코덱 하드웨어 유닛에 결합되거나 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 위에서 설명된 바와 같은 하나 이상의 프로세서들을 포함하는, 상호운용적 하드웨어 유닛들의 컬렉션에 의해 제공될 수도 있다.

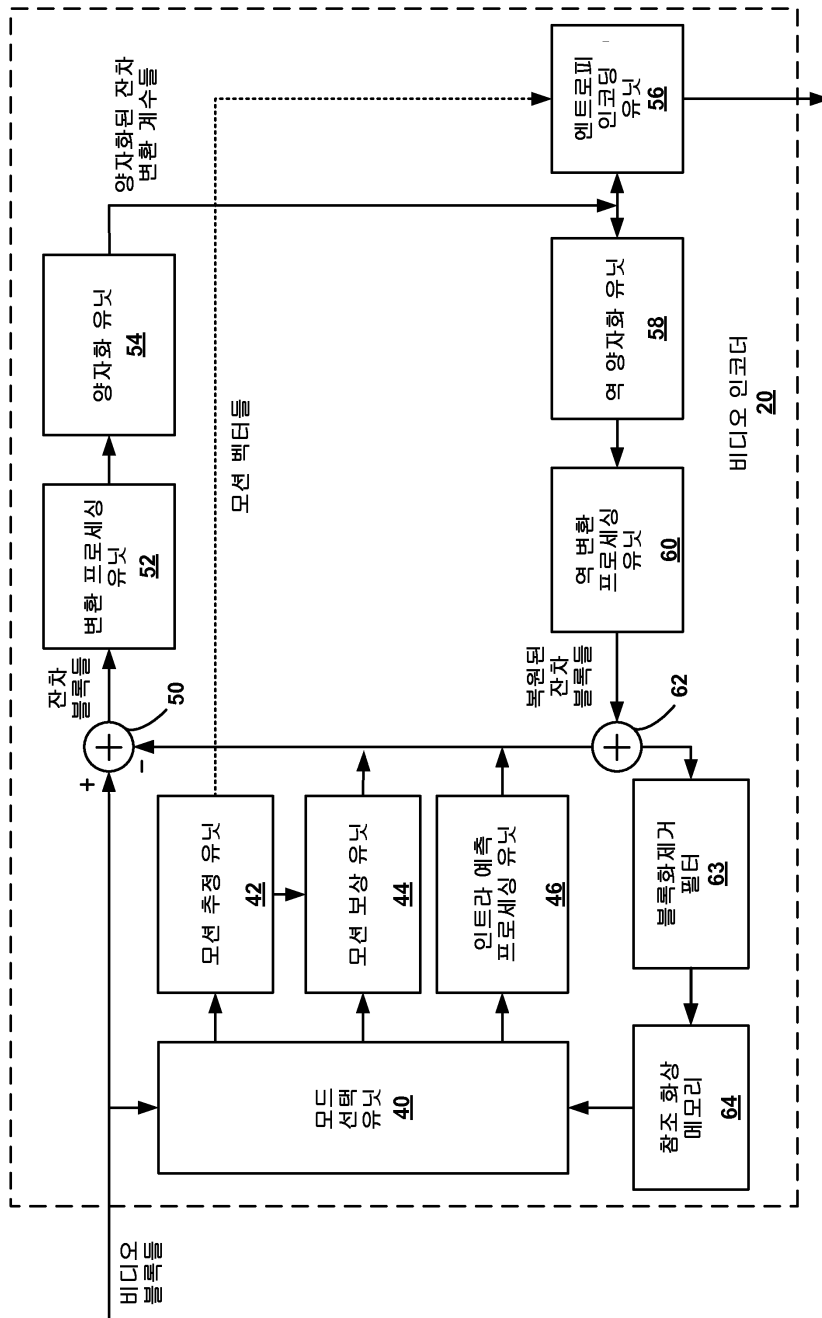
[0223] 다양한 예들이 설명되어 있다. 이들 및 다른 예들은 다음의 청구항들의 범위 내에 있다.

도면

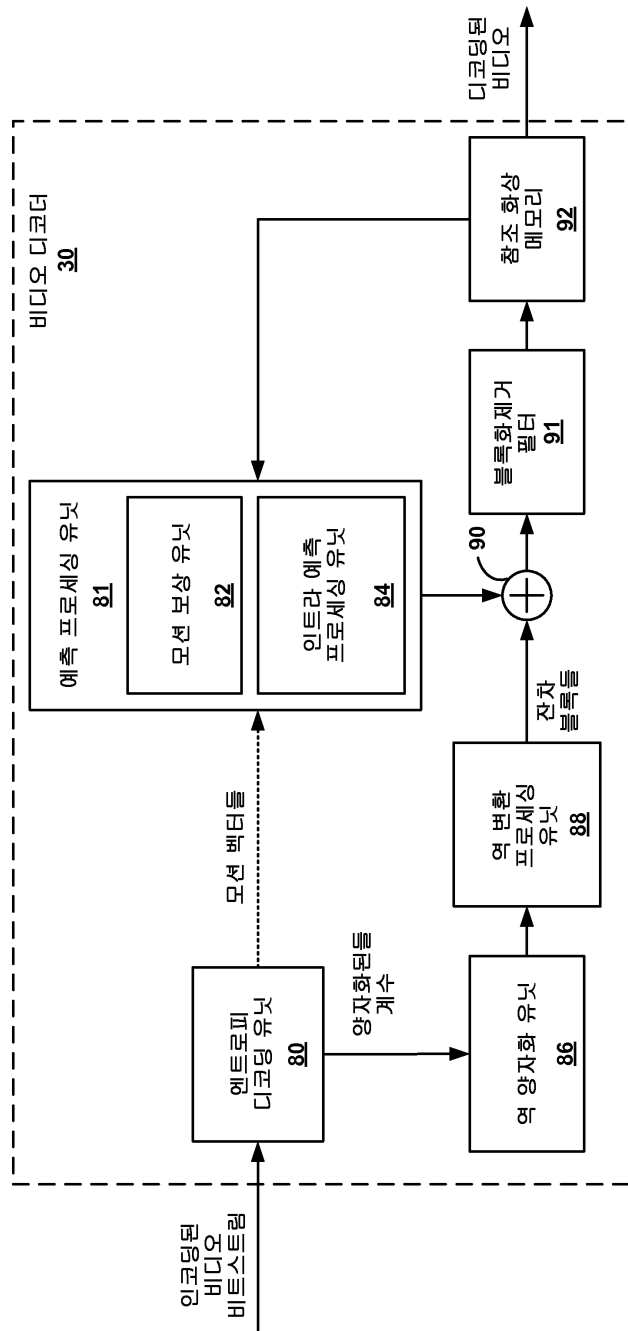
도면1



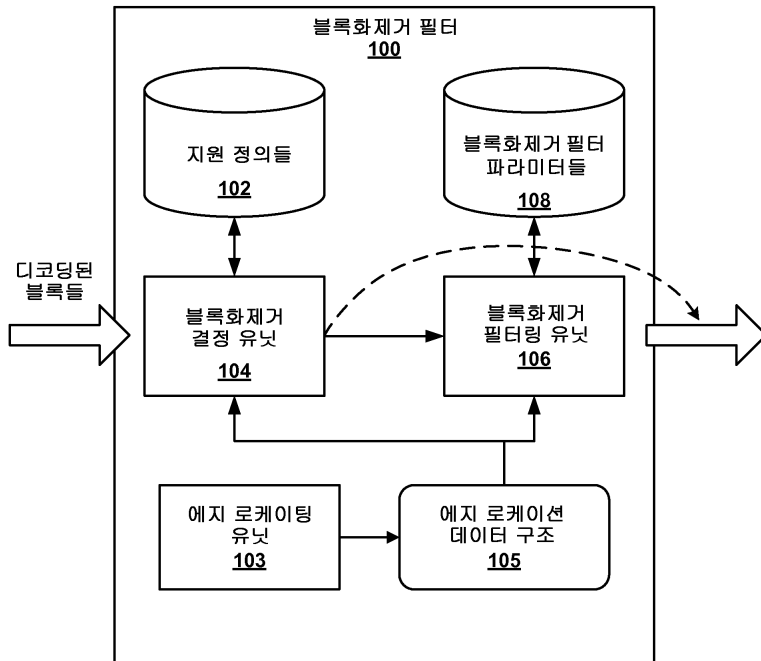
도면2



도면3



도면4



도면5

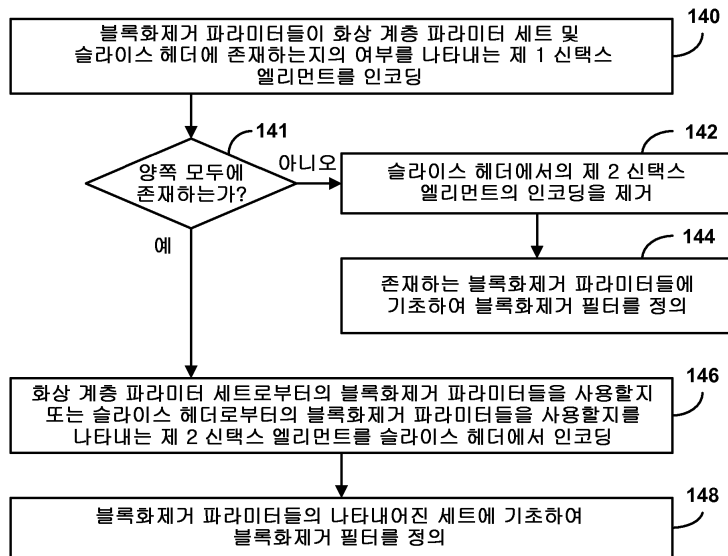
130

132

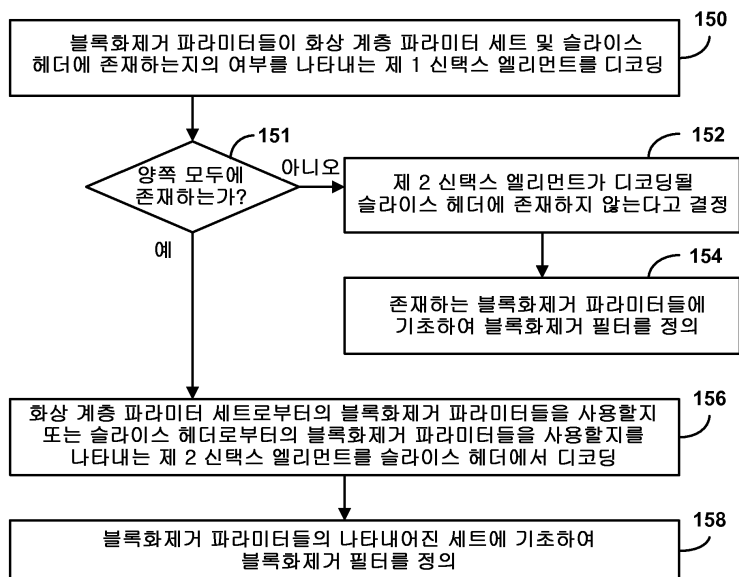
134

p7 ₀	p6 ₀	p5 ₀	p4 ₀	p3 ₀	p2 ₀	p1 ₀	p0 ₀	q0 ₀	q1 ₀	q2 ₀	q3 ₀	q4 ₀	q5 ₀	q6 ₀	q7 ₀
p7 ₁	p6 ₁	p5 ₁	p4 ₁	p3 ₁	p2 ₁	p1 ₁	p0 ₁	q0 ₁	q1 ₁	q2 ₁	q3 ₁	q4 ₁	q5 ₁	q6 ₁	q7 ₁
p7 ₂	p6 ₂	p5 ₂	p4 ₂	p3 ₂	p2 ₂	p1 ₂	p0 ₂	q0 ₂	q1 ₂	q2 ₂	q3 ₂	q4 ₂	q5 ₂	q6 ₂	q7 ₂
p7 ₃	p6 ₃	p5 ₃	p4 ₃	p3 ₃	p2 ₃	p1 ₃	p0 ₃	q0 ₃	q1 ₃	q2 ₃	q3 ₃	q4 ₃	q5 ₃	q6 ₃	q7 ₃
p7 ₄	p6 ₄	p5 ₄	p4 ₄	p3 ₄	p2 ₄	p1 ₄	p0 ₄	q0 ₄	q1 ₄	q2 ₄	q3 ₄	q4 ₄	q5 ₄	q6 ₄	q7 ₄
p7 ₅	p6 ₅	p5 ₅	p4 ₅	p3 ₅	p2 ₅	p1 ₅	p0 ₅	q0 ₅	q1 ₅	q2 ₅	q3 ₅	q4 ₅	q5 ₅	q6 ₅	q7 ₅
p7 ₆	p6 ₆	p5 ₆	p4 ₆	p3 ₆	p2 ₆	p1 ₆	p0 ₆	q0 ₆	q1 ₆	q2 ₆	q3 ₆	q4 ₆	q5 ₆	q6 ₆	q7 ₆
p7 ₇	p6 ₇	p5 ₇	p4 ₇	p3 ₇	p2 ₇	p1 ₇	p0 ₇	q0 ₇	q1 ₇	q2 ₇	q3 ₇	q4 ₇	q5 ₇	q6 ₇	q7 ₇

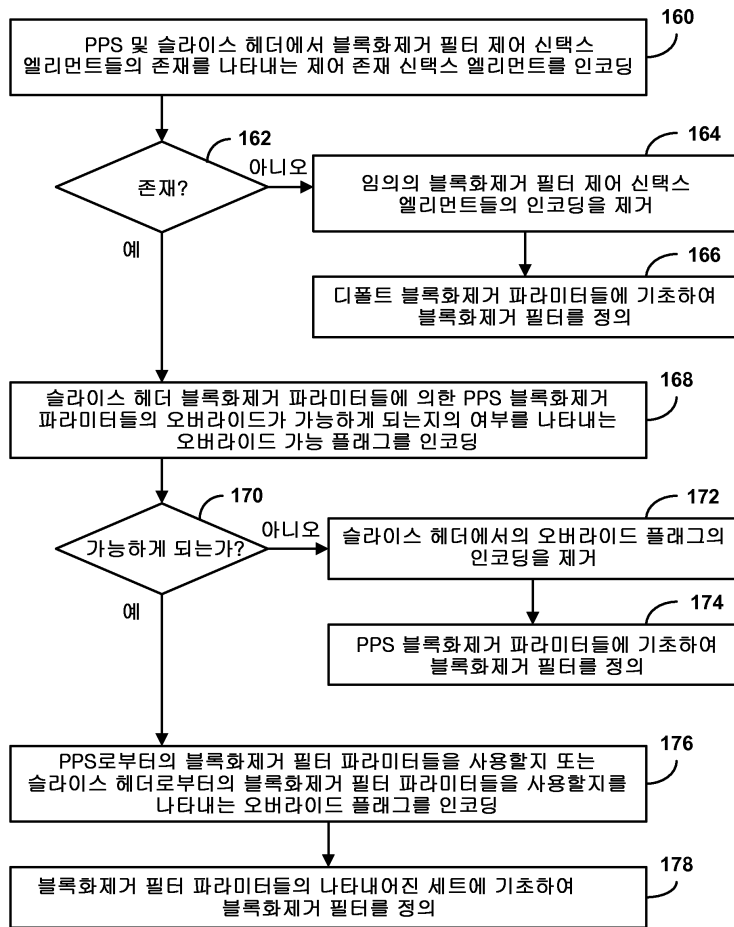
도면6



도면7



도면8



도면9

