



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2016년07월01일
(11) 등록번호 10-1633197
(24) 등록일자 2016년06월17일

- (51) 국제특허분류(Int. Cl.)
H04N 19/86 (2014.01) H04N 19/124 (2014.01)
- (21) 출원번호 10-2014-7011940
- (22) 출원일자(국제) 2012년10월05일
심사청구일자 2014년05월01일
- (85) 번역문제출일자 2014년05월01일
- (65) 공개번호 10-2014-0074979
- (43) 공개일자 2014년06월18일
- (86) 국제출원번호 PCT/US2012/059010
- (87) 국제공개번호 WO 2013/052835
국제공개일자 2013년04월11일
- (30) 우선권주장
13/645,345 2012년10월04일 미국(US)
(뒷면에 계속)
- (56) 선행기술조사문헌
JCT-VC of ITU-T SG16 WP3 and ISO/IEC
JTC1/SC29/WG11 6th Meeting: Trino,
(JCTVC-F142_r1), (CE12: Deblocking filter
parameter adjustment in slice level), 14-22
July, 2011*
JCT-VC of ITU-T SG16 WP3 and ISO/IEC
JTC1/SC29/WG11 3rd Meeting: Guangzhou,
(JCTVC-C508_r1), (Description of CE8: In-loop
filtering), 7-15 October, 2010*
JCT-VC(WD4: Working Draft 4 of
High-Efficiency Video Coding)
KR1020080093426 A
*는 심사관에 의하여 인용된 문헌

- (73) 특허권자
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하
우스 드라이브 5775
- (72) 발명자
판 더 아우베라 게르트
미국 92121-1714 캘리포니아주 샌디에고 모어하
우스 드라이브 5775
조쉬 라잔 렉스맨
미국 92121-1714 캘리포니아주 샌디에고 모어하
우스 드라이브 5775
카르체비츠 마르타
미국 92121-1714 캘리포니아주 샌디에고 모어하
우스 드라이브 5775
- (74) 대리인
특허법인코리아나

전체 청구항 수 : 총 79 항

심사관 : 조우연

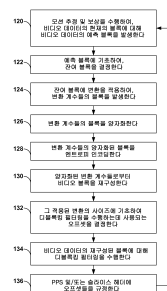
(54) 발명의 명칭 변환 의존적 디블록킹 필터링의 수행

(57) 요약

일반적으로, 비디오 인코딩 디바이스에 의해 구현될 수도 있는 변환 의존적인 디블록킹 필터링을 수행하는 기법들이 설명된다. 비디오 인코딩 디바이스는 변환을 비디오 데이터 블록에 적용하여 변환 계수들의 블록을 발생하고, 양자화 파라미터를 적용하여 변환 계수들을 양자화하고, 그리고 양자화된 변환 계수들로부터 비디오 데

(뒷면에 계속)

대표도 - 도5



이터의 블록을 재구성할 수도 있다. 비디오 인코딩 디바이스는 추가로, 그 적용된 변환의 사이즈에 기초하여 디블록킹 필터링을 제어하는데 사용되는 적어도 하나의 오프셋을 결정하고, 그리고, 그 결정된 오프셋에 기초하여, 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행할 수도 있다. 게다가, 비디오 인코더는 오프셋이 독립적으로 디코딩가능한 유닛의 헤더와 PPS 중 하나 또는 양쪽에 규정되는지 여부를 나타내는 플래그를 화상 파라미터 세트 (PPS) 에 규정할 수도 있다.

(30) 우선권주장

61/544,960	2011년10월07일	미국(US)
61/554,260	2011년11월01일	미국(US)
61/588,480	2012년01월19일	미국(US)
61/663,469	2012년06월22일	미국(US)
61/682,167	2012년08월10일	미국(US)
61/700,167	2012년09월12일	미국(US)

명세서

청구범위

청구항 1

비디오 데이터를 인코딩하는 방법으로서,

변환 계수들의 블록을 발생하기 위해 상기 비디오 데이터의 블록에 변환을 적용하는 단계;

상기 변환 계수들의 블록을 양자화하기 위해 양자화 파라미터를 적용하는 단계;

상기 변환 계수들의 양자화된 블록에 기초하여 비디오 비트스트림에 대한 상기 비디오 데이터의 블록의 인코딩된 버전을 생성하는 단계;

상기 변환 계수들의 양자화된 블록으로부터 상기 비디오 데이터의 블록을 재구성하는 단계;

상기 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 단계로서, 상기 적어도 하나의 오프셋은 t_c 오프셋과 β 오프셋 중 하나 이상을 포함하고, 상기 t_c 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 및 크로마 양태들 양쪽에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하며, 상기 β 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 양태들에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하는, 상기 적어도 하나의 오프셋을 결정하는 단계;

결정된 상기 적어도 하나의 오프셋에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하는 단계;

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 상기 인코딩된 버전을 포함하는 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (picture parameter set; PPS) 중 하나 또는 양쪽에 적어도 하나의 구문 (syntax) 엘리먼트로서 규정되는지 여부를 나타내는 플래그를 상기 비디오 비트스트림의 상기 PPS 에 규정하는 단계; 및

상기 적어도 하나의 오프셋을, 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 상기 헤더와 상기 PPS 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 2

제 1 항에 있어서,

상기 플래그는 de-blocking_filter_control_present_flag 를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 3

삭제

청구항 4

삭제

청구항 5

제 1 항에 있어서,

상기 적어도 하나의 오프셋을 결정하는 단계는,

상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 최대 변환 유닛 사이즈인지 여부를 결정하는

단계; 및

상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 상기 최대 변환 유닛 사이즈라는 결정시, 상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 최대 변환 유닛 사이즈가 아닐 때 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹을 수행할지 여부를 결정하기 위해 다르게 사용되는 상기 t_c 오프셋 및 상기 β 오프셋 대신 사용되는 최대 t_c 오프셋 및 최대 β 오프셋 중 하나 이상을 결정하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 6

제 5 항에 있어서,

상기 최대 t_c 오프셋 및 상기 최대 β 오프셋을, 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 7

제 5 항에 있어서,

상기 최대 β 오프셋 및 상기 최대 t_c 오프셋이 상기 독립적으로 디코딩가능한 유닛의 헤더 또는 상기 PPS 에 각각 규정되는지 여부를 나타내는 `de-blocking_max_tu_offset_enabled_flag` 를, 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 8

제 1 항에 있어서,

상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하는 단계를 더 포함하고,

상기 적어도 하나의 오프셋을 결정하는 단계는, 결정된 상기 경계 강도 값에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 9

제 1 항에 있어서,

상기 비디오 데이터의 블록 또는 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛에 포함되는지 여부, 및 상기 비디오 데이터의 블록 또는 상기 비디오 데이터의 블록에 인접한 블록이 최대 사이즈의 변환 유닛에 포함되는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하는 단계를 더 포함하고,

상기 적어도 하나의 오프셋을 결정하는 단계는, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 10

제 1 항에 있어서,

상기 비디오 데이터의 블록이 인트라-코딩된 코딩 유닛에 포함되어 있고 최대 사이즈의 변환 유닛에 포함되는지 여부, 또는 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛에 포함되어 있고 최대 사이즈의 변환 유닛에 포함되는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하는 단계를 더 포함하고;

상기 적어도 하나의 오프셋을 결정하는 단계는, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구

성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 11

제 1 항에 있어서,

상기 비디오 데이터의 블록도 상기 비디오 데이터의 블록에 인접한 블록도 인트라-코딩된 코딩 유닛에 포함되어 있지 않고 상기 비디오 데이터의 블록과 상기 비디오 데이터의 블록에 인접한 블록 중 적어도 하나가 최대 사이즈의 변환 유닛에 포함되어 있을 때, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하는 단계를 더 포함하고,

상기 적어도 하나의 오프셋을 결정하는 단계는, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 12

제 1 항에 있어서,

상기 비디오 데이터의 블록과 상기 비디오 데이터의 블록에 인접한 블록 중 적어도 하나가 최대 사이즈의 변환 유닛에 포함되어 있을 때, 상기 비디오 데이터의 블록 및 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛 내에 포함되는지 여부를 결정함이 없이, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하는 단계를 더 포함하고;

상기 적어도 하나의 오프셋을 결정하는 단계는, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 13

제 1 항에 있어서,

상기 적어도 하나의 오프셋을 결정하는 단계는, 상기 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 14

제 13 항에 있어서,

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정되어 있다는 것을 나타내는 플래그를 상기 PPS 에 규정하는 단계, 및

상기 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부에 기초하여 결정된 적어도 하나의 오프셋을, 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽에 상기 적어도 하나의 구문 엘리먼트로서 규정하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 15

제 1 항에 있어서,

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 상기 적어도 하나의 구문 엘리먼트로서

규정되어 있지 않다는 것을 나타내는 플래그를 상기 PPS 에 규정하는 단계; 및

상기 적어도 하나의 오프셋을 상기 PPS 에 구문 엘리먼트로서 규정하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 16

비디오 데이터를 인코딩하도록 구성되는 비디오 인코딩 디바이스로서,

상기 비디오 데이터를 저장하도록 구성된 메모리; 및

변환 계수들의 블록을 발생하기 위해 비디오 데이터의 블록에 변환을 적용하고, 상기 변환 계수들의 블록을 양자화하기 위해 양자화 파라미터를 적용하고, 상기 변환 계수들의 양자화된 블록에 기초하여 비디오 비트스트림에 대한 상기 비디오 데이터의 블록의 인코딩된 버전을 생성하고, 상기 변환 계수들의 양자화된 블록으로부터 상기 비디오 데이터의 블록을 재구성하고, 상기 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하고, 결정된 상기 적어도 하나의 오프셋에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하며, 상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 상기 인코딩된 버전을 포함하는 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정되는지 여부를 나타내는 플래그를 상기 비디오 비트스트림의 상기 PPS 에 규정하고, 그리고 상기 적어도 하나의 오프셋을 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩가능한 유닛의 상기 헤더와 상기 PPS 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정하도록 구성되는 하나 이상의 프로세서들을 포함하고,

상기 적어도 하나의 오프셋은 t_c 오프셋과 β 오프셋 중 하나 이상을 포함하고, 상기 t_c 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 및 크로마 양태들 양쪽에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하며, 상기 β 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 양태들에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하는, 비디오 인코딩 디바이스.

청구항 17

제 16 항에 있어서,

상기 플래그는 de-blocking_filter_control_present_flag 를 포함하는, 비디오 인코딩 디바이스.

청구항 18

삭제

청구항 19

삭제

청구항 20

제 16 항에 있어서,

상기 하나 이상의 프로세서들은 또한,

상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 최대 변환 유닛 사이즈인지 여부를 결정하고, 상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 상기 최대 변환 유닛 사이즈라는 결정시, 상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 최대 변환 유닛 사이즈가 아닐 때 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹을 수행할지 여부를 결정하기 위해 다르게 사용되는 상기 t_c 오프셋 및 상기 β 오프셋 대신 사용되는 최대 t_c 오프셋 및 최대 β 오프셋 중 하나 이상을 결정하도록 구성되는, 비디오 인코딩 디바이스.

청구항 21

제 20 항에 있어서,

상기 하나 이상의 프로세서들은 또한,

상기 최대 t_c 오프셋 및 상기 최대 β 오프셋을, 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정하도록 구성되는, 비디오 인코딩 디바이스.

청구항 22

제 20 항에 있어서,

상기 하나 이상의 프로세서들은 또한,

상기 최대 β 오프셋 및 상기 최대 t_c 오프셋이 상기 독립적으로 디코딩가능한 유닛의 헤더 또는 상기 PPS 에 각각 규정되는지 여부를 나타내는 `de-blocking_max_tu_offset_enabled_flag` 를, 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정하도록 구성되는, 비디오 인코딩 디바이스.

청구항 23

제 16 항에 있어서,

상기 하나 이상의 프로세서들은 또한,

상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하고, 결정된 상기 경계 강도 값에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하도록 구성되는, 비디오 인코딩 디바이스.

청구항 24

제 16 항에 있어서,

상기 하나 이상의 프로세서들은 또한,

상기 비디오 데이터의 블록 또는 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛에 포함되는지 여부, 및 상기 비디오 데이터의 블록 또는 상기 비디오 데이터의 블록에 인접한 블록이 최대 사이즈의 변환 유닛에 포함되는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하고, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 상기 적어도 하나의 오프셋을 결정하도록 구성되는, 비디오 인코딩 디바이스.

청구항 25

제 16 항에 있어서,

상기 하나 이상의 프로세서들은 또한,

상기 비디오 데이터의 블록이 인트라-코딩된 코딩 유닛에 포함되어 있고 최대 사이즈의 변환 유닛에 포함되는지 여부, 또는 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛에 포함되어 있고 최대 사이즈의 변환 유닛에 포함되는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하고, 결정된 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 상기 적어도 하나의 오프셋을 결정하도록 구성되는, 비디오 인코딩 디바이스.

청구항 26

제 16 항에 있어서,

상기 하나 이상의 프로세서들은 또한,

상기 비디오 데이터의 블록도 상기 비디오 데이터의 블록에 인접한 블록도 인트라-코딩된 코딩 유닛에 포함되어 있지 않고 상기 비디오 데이터의 블록과 상기 비디오 데이터의 블록에 인접한 블록 중 적어도 하나가 최대 사이즈의 변환 유닛에 포함되어 있을 때, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하고, 결정된 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 상기 적어도 하나의 오프셋을 결정하도록 구성되는, 비디오 인코딩 디바이스.

청구항 27

제 16 항에 있어서,

상기 하나 이상의 프로세서는 또한,

상기 비디오 데이터의 블록과 상기 비디오 데이터의 블록에 인접한 블록 중 적어도 하나가 최대 사이즈의 변환 유닛에 포함되어 있을 때, 상기 비디오 데이터의 블록 및 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛 내에 포함되는지 여부를 결정함이 없이, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하고, 결정된 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 상기 적어도 하나의 오프셋을 결정하도록 구성되는, 비디오 인코딩 디바이스.

청구항 28

제 16 항에 있어서,

상기 하나 이상의 프로세서는 또한,

상기 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 상기 적어도 하나의 오프셋을 결정하도록 구성되는, 비디오 인코딩 디바이스.

청구항 29

제 28 항에 있어서,

상기 하나 이상의 프로세서는 또한,

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정되어 있다는 것을 나타내는 플래그를 상기 PPS 에 규정하고,

상기 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부 및 적용된 상기 변환의 사이즈 양쪽에 기초하여 결정된 적어도 하나의 오프셋을, 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽에 상기 적어도 하나의 구문 엘리먼트로서 규정하도록

구성되는, 비디오 인코딩 디바이스.

청구항 30

제 16 항에 있어서,

상기 하나 이상의 프로세서는 또한,

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 상기 적어도 하나의 구문 엘리먼트로서 규정되어 있지 않다는 것을 나타내는 플래그를 상기 PPS 에 규정하고,

상기 적어도 하나의 오프셋을 상기 PPS 에 구문 엘리먼트로서 규정하도록

구성되는, 비디오 인코딩 디바이스.

청구항 31

제 16 항에 있어서,

상기 하나 이상의 프로세서는 또한,

상기 변환 계수들의 블록을 양자화하기 위해 상기 변환의 사이즈에 기초하여 변하는 양자화 파라미터를 적용하도록 구성되는, 비디오 인코딩 디바이스.

청구항 32

비디오 데이터를 인코딩하도록 구성되는 비디오 인코딩 디바이스로서,

변환 계수들의 블록을 발생하기 위해 상기 비디오 데이터의 블록에 변환을 적용하는 수단;

상기 변환 계수들의 블록을 양자화하기 위해 양자화 파라미터를 적용하는 수단;

상기 변환 계수들의 양자화된 블록에 기초하여 비디오 비트스트림에 대한 상기 비디오 데이터의 블록의 인코딩된 버전을 생성하는 수단;

상기 변환 계수들의 양자화된 블록으로부터 상기 비디오 데이터의 블록을 재구성하는 수단;

상기 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하는 수단으로서, 상기 적어도 하나의 오프셋은 t_c 오프셋과 β 오프셋 중 하나 이상을 포함하고, 상기 t_c 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 및 크로마 양태들 양쪽에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하며, 상기 β 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 양태들에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하는, 상기 적어도 하나의 오프셋을 결정하는 수단;

결정된 상기 적어도 하나의 오프셋에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하는 수단;

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 상기 인코딩된 버전을 포함하는 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정되는지 여부를 나타내는 플래그를 상기 비디오 비트스트림의 상기 PPS 에 규정하는 수단; 및

상기 적어도 하나의 오프셋을, 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 상기 헤더와 상기 PPS 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정하는 수단

을 포함하는, 비디오 인코딩 디바이스.

청구항 33

제 32 항에 있어서,

상기 플래그는 de-blocking_filter_control_present_flag 를 포함하는, 비디오 인코딩 디바이스.

청구항 34

삭제

청구항 35

삭제

청구항 36

제 32 항에 있어서,

상기 적어도 하나의 오프셋을 결정하는 수단은,

상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 최대 변환 유닛 사이즈인지 여부를 결정하는 수단; 및

상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 상기 최대 변환 유닛 사이즈라는 결정시, 상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 최대 변환 유닛 사이즈가 아닐 때 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹을 수행할지 여부를 결정하기 위해 다르게 사용되는 상기 t_c 오프셋 및 상기 β 오프셋 대신 사용되는 최대 t_c 오프셋 및 최대 β 오프셋 중 하나 이상을 결정하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 37

제 36 항에 있어서,

상기 최대 t_c 오프셋 및 상기 최대 β 오프셋을, 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정하는 수단을 더 포함하는, 비디오 인코딩 디바이스.

청구항 38

제 36 항에 있어서,

상기 최대 β 오프셋 및 상기 최대 t_c 오프셋이 상기 독립적으로 디코딩가능한 유닛의 헤더 또는 화상 파라미터 세트 (PPS) 에 각각 규정되는지 여부를 나타내는 `de-blocking_max_tu_offset_enabled_flag` 를, 상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽에 규정하는 수단을 더 포함하는, 비디오 인코딩 디바이스.

청구항 39

제 32 항에 있어서,

상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하는 수단을 더 포함하고,

상기 적어도 하나의 오프셋을 결정하는 수단은, 결정된 상기 경계 강도 값에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 40

제 32 항에 있어서,

상기 비디오 데이터의 블록 또는 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛에 포함되는지 여부, 및 상기 비디오 데이터의 블록 또는 상기 비디오 데이터의 블록에 인접한 블록이 최대 사이즈의 변환 유닛에 포함되는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하는 수단을 더 포함하고,

상기 적어도 하나의 오프셋을 결정하는 수단은, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 41

제 32 항에 있어서,

상기 비디오 데이터의 블록이 인트라-코딩된 코딩 유닛에 포함되어 있고 최대 사이즈의 변환 유닛에 포함되는지 여부, 또는 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛에 포함되어 있고 최대 사이즈의 변환 유닛에 포함되는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하는 수단을 더 포함하고,

상기 적어도 하나의 오프셋을 결정하는 수단은, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 42

제 32 항에 있어서,

상기 비디오 데이터의 블록도 상기 비디오 데이터의 블록에 인접한 블록도 인트라-코딩된 코딩 유닛에 포함되어 있지 않고 상기 비디오 데이터의 블록과 상기 비디오 데이터의 블록에 인접한 블록 중 적어도 하나가 최대 사이즈의 변환 유닛에 포함되어 있을 때, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하는 수단을 더 포함하고,

상기 적어도 하나의 오프셋을 결정하는 수단은, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 43

제 32 항에 있어서,

상기 비디오 데이터의 블록과 상기 비디오 데이터의 블록에 인접한 블록 중 적어도 하나가 최대 사이즈의 변환 유닛에 포함되어 있을 때, 상기 비디오 데이터의 블록 및 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛 내에 포함되는지 여부를 결정함이 없이, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하는 수단을 더 포함하고,

상기 적어도 하나의 오프셋을 결정하는 수단은, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 44

제 32 항에 있어서,

상기 적어도 하나의 오프셋을 결정하는 수단은,

상기 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 45

제 44 항에 있어서,

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정되어 있다는 것을 나타내는 플래그를 상기 PPS 에 규정하는 수단; 및

상기 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부 및 적용된 상기 변환의 사이즈 양쪽에 기초하여 결정된 적어도 하나의 오프셋을, 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽에 상기 적어도 하나의 구문 엘리먼트로서 규정하는 수단을 더 포함하는, 비디오 인코딩 디바이스.

청구항 46

제 32 항에 있어서,

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 상기 적어도 하나의 구문 엘리먼트로서 규정되어 있지 않다는 것을 나타내는 플래그를 상기 PPS 에 규정하는 수단; 및

상기 적어도 하나의 오프셋을 상기 PPS 에 구문 엘리먼트로서 규정하는 수단을 더 포함하는, 비디오 인코딩 디바이스.

청구항 47

제 32 항에 있어서,

상기 양자화 파라미터를 적용하는 수단은, 상기 변환 계수들의 블록을 양자화하기 위해 상기 변환의 사이즈에 기초하여 변하는 양자화 파라미터를 적용하는 수단을 포함하는, 비디오 인코딩 디바이스.

청구항 48

명령들을 저장하고 있는 비일시적 컴퓨터 판독가능 저장 매체로서,

상기 명령들은, 실행될 때, 하나 이상의 프로세서들로 하여금,

변환 계수들의 블록을 발생하기 위해 비디오 데이터의 블록에 변환을 적용하게 하고;

상기 변환 계수들의 블록을 양자화하기 위해 양자화 파라미터를 적용하게 하고;

상기 변환 계수들의 양자화된 블록에 기초하여 비디오 비트스트림에 대한 상기 비디오 데이터의 블록의 인코딩된 버전을 생성하게 하고;

상기 변환 계수들의 양자화된 블록으로부터 상기 비디오 데이터의 블록을 재구성하게 하고;

상기 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하게 하고;

결정된 상기 적어도 하나의 오프셋에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하게 하며;

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 상기 인코딩된 버전을 포함하는 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정되는지 여부를 나타내는 플래그를 상기 비디오 비트스트림의 상기 PPS 에 규정하게 하고; 그리고

상기 적어도 하나의 오프셋을, 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정하게 하고,

상기 적어도 하나의 오프셋은 t_c 오프셋과 β 오프셋 중 하나 이상을 포함하고, 상기 t_c 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 및 크로마 양태들 양쪽에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하며, 상기 β 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 양태들에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 49

제 48 항에 있어서,

상기 플래그는 de-blocking_filter_control_present_flag 를 포함하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 50

삭제

청구항 51

삭제

청구항 52

제 48 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금, 적어도 하나의 오프셋을 결정할 때,

상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 최대 변환 유닛 사이즈인지 여부를 결정하게 하고;

상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 상기 최대 변환 유닛 사이즈라는 결정시, 상기 비디오 데이터의 블록에 적용되는 상기 변환의 사이즈가 최대 변환 유닛 사이즈가 아닐 때 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹을 수행할지 여부를 결정할 때에 다르게 사용되는 상기 t_c 오프셋 및 상기 β 오프셋 대신 사용되는 최대 t_c 오프셋 및 최대 β 오프셋 중 하나 이상을 결정하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 53

제 52 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 최대 t_c 오프셋 및 상기 최대 β 오프셋을, 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 54

제 52 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 최대 β 오프셋 및 상기 최대 t_c 오프셋이 상기 독립적으로 디코딩가능한 유닛의 헤더 또는 상기 PPS 에 각각 규정되는지 여부를 나타내는 `de-blocking_max_tu_offset_enabled_flag` 를, 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 55

제 48 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하게 하고;

상기 적어도 하나의 오프셋을 결정할 때, 결정된 상기 경계 강도 값에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 56

제 48 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 비디오 데이터의 블록 또는 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛에 포함되는지 여부, 및 상기 비디오 데이터의 블록 또는 상기 비디오 데이터의 블록에 인접한 블록이 최대 사이즈의 변환 유닛에 포함되는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하

게 하고;

적어도 하나의 오프셋을 결정할 때, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 57

제 48 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 비디오 데이터의 블록이 인트라-코딩된 코딩 유닛에 포함되어 있고 최대 사이즈의 변환 유닛에 포함되는지 여부, 또는 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛에 포함되어 있고 최대 사이즈의 변환 유닛에 포함되는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하게 하고;

상기 적어도 하나의 오프셋을 결정할 때, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 58

제 48 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 비디오 데이터의 블록도 상기 비디오 데이터의 블록에 인접한 블록도 인트라-코딩된 코딩 유닛에 포함되어 있지 않고 상기 비디오 데이터의 블록과 상기 비디오 데이터의 블록에 인접한 블록 중 적어도 하나가 최대 사이즈의 변환 유닛에 포함되어 있을 때, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하게 하고;

상기 적어도 하나의 오프셋을 결정할 때, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 59

제 48 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 비디오 데이터의 블록과 상기 비디오 데이터의 블록에 인접한 블록 중 적어도 하나가 최대 사이즈의 변환 유닛에 포함되어 있을 때, 상기 비디오 데이터의 블록 및 상기 비디오 데이터의 블록에 인접한 블록이 인트라-코딩된 코딩 유닛 내에 포함되는지 여부를 결정함이 없이, 상기 비디오 데이터의 재구성된 블록에 대한 경계 강도 값을 결정하게 하고;

상기 적어도 하나의 오프셋을 결정할 때, 결정된 상기 경계 강도에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 60

제 48 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 적어도 하나의 오프셋을 결정할 때, 상기 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되

있는지 여부에 기초하여, 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 상기 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 결정하게 하는, 비밀시적 컴퓨터 판독가능 저장 매체.

청구항 61

제 60 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 적어도 하나의 구문 엘리먼트로서 규정되어 있다는 것을 나타내는 플래그를 상기 PPS 에 규정하게 하고;

상기 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부 및 적용된 상기 변환의 사이즈 양쪽에 기초하여 결정된 적어도 하나의 오프셋을, 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽에 상기 적어도 하나의 구문 엘리먼트로서 규정하게 하는, 비밀시적 컴퓨터 판독가능 저장 매체.

청구항 62

제 48 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 적어도 하나의 오프셋이 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩 가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 상기 적어도 하나의 구문 엘리먼트로서 규정되어 있지 않다는 것을 나타내는 플래그를 상기 PPS 에 규정하게 하고;

상기 적어도 하나의 오프셋을 상기 PPS 에 구문 엘리먼트로서 규정하게 하는, 비밀시적 컴퓨터 판독가능 저장 매체.

청구항 63

제 48 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 양자화 파라미터를 적용할 때, 상기 변환의 사이즈에 기초하여 변하는 양자화 파라미터를 적용하여 상기 변환 계수들의 블록을 양자화하게 하는, 비밀시적 컴퓨터 판독가능 저장 매체.

청구항 64

인코딩된 비디오 데이터를 비디오 디코더로 디코딩하는 방법으로서,

비디오 비트스트림 내의 상기 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 상기 인코딩된 비디오 데이터의 블록에 역양자화 파라미터를 적용하는 단계;

잔여 비디오 데이터의 블록을 발생하기 위해 상기 인코딩된 비디오 데이터의 양자화 해제된 블록에 역변환을 적용하는 단계;

상기 잔여 비디오 데이터의 블록으로부터 비디오 데이터의 블록을 재구성하는 단계;

상기 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋이 상기 인코딩된 비디오 데이터의 블록을 포함하는 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정되는지 여부를 나타내는, 상기 비디오 비트스트림 내의 상기 인코딩된 비디오 데이터의 상기 PPS 에 규정된 플래그를 결정하는 단계;

상기 플래그에 기초하여, 상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽으로부터 상기 적어도 하나의 오프셋을 추출하는 단계; 및

상기 적어도 하나의 오프셋에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하는 단계를 포함하고,

상기 적어도 하나의 오프셋은 t_c 오프셋과 β 오프셋 중 하나 이상을 포함하고, 상기 t_c 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 및 크로마 양태들 양쪽에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하며, 상기 β 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 양태들에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하는, 인코딩된 비디오 데이터를 비디오 디코더로 디코딩하는 방법.

청구항 65

제 64 항에 있어서,

상기 플래그는 `de-blocking_filter_control_present_flag` 를 포함하는, 인코딩된 비디오 데이터를 비디오 디코더로 디코딩하는 방법.

청구항 66

삭제

청구항 67

삭제

청구항 68

제 64 항에 있어서,

상기 적어도 하나의 오프셋은,

상기 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나, 또는 상기 인코딩된 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부와 연관되는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 포함하는, 인코딩된 비디오 데이터를 비디오 디코더로 디코딩하는 방법.

청구항 69

제 64 항에 있어서,

상기 적어도 하나의 오프셋을 결정하는 단계는,

상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽으로부터, 상기 비디오 데이터의 블록에 적용되는 변환의 사이즈가 최대 변환 유닛 사이즈가 아닐 때 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹을 수행할지 여부를 결정할 때에 다르게 사용되는 상기 t_c 오프셋 및 상기 β 오프셋 대신 사용되는 최대 t_c 오프셋 및 최대 β 오프셋을 추출하는 단계를 포함하는, 인코딩된 비디오 데이터를 비디오 디코더로 디코딩하는 방법.

청구항 70

제 69 항에 있어서,

상기 최대 t_c 오프셋 및 최대 β 오프셋을 추출하는 단계는,

상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽에 규정된 `de-blocking_max_tu_offset_enabled_flag` 에 기초하여, 상기 최대 t_c 오프셋 및 상기 최대 β 오프셋을 추출하는 단계를 포함하는, 인코딩된 비디오 데이터를 비디오 디코더로 디코딩하는 방법.

청구항 71

제 64 항에 있어서,

상기 역양자화 파라미터를 적용하는 단계는,

상기 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 역변환의 사이즈에 기초하여 변하는 역양자화 파라미터를 적용하는 단계를 포함하는, 인코딩된 비디오 데이터를 비디오 디코더로 디코딩하는 방법.

청구항 72

인코딩된 비디오 데이터를 디코딩하도록 구성된 비디오 디코딩 디바이스로서,

상기 인코딩된 비디오 데이터를 저장하도록 구성된 메모리; 및

상기 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 비디오 비트스트림 내의 상기 인코딩된 비디오 데이터의 블록에 역양자화 파라미터를 적용하고; 잔여 비디오 데이터의 블록을 발생하기 위해 상기 인코딩된 비디오 데이터의 양자화 해제된 블록에 역변환을 적용하고; 상기 잔여 비디오 데이터의 블록으로부터 비디오 데이터의 블록을 재구성하고; 적어도 하나의 오프셋을 결정할 때, 상기 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 상기 적어도 하나의 오프셋이 상기 인코딩된 비디오 데이터의 블록을 포함하는 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정되는지 여부를 나타내는, 상기 비디오 비트스트림 내의 상기 인코딩된 비디오 데이터의 상기 PPS 에 규정된 플래그를 결정하고; 상기 플래그에 기초하여, 상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽으로부터 상기 적어도 하나의 오프셋을 추출하며; 결정된 상기 적어도 하나의 오프셋에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하도록 구성되는 하나 이상의 프로세서들을 포함하고,

상기 적어도 하나의 오프셋은 t_c 오프셋과 β 오프셋 중 하나 이상을 포함하고, 상기 t_c 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 및 크로마 양태들 양쪽에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하며, 상기 β 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 양태들에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하는, 비디오 디코딩 디바이스.

청구항 73

제 72 항에 있어서,

상기 플래그는 de-blocking_filter_control_present_flag 를 포함하는, 비디오 디코딩 디바이스.

청구항 74

삭제

청구항 75

삭제

청구항 76

제 72 항에 있어서,

상기 적어도 하나의 오프셋은,

상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것, 또는 상기 인코딩된 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부와 연관되는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 포함하는, 비디오 디코딩 디바이스.

청구항 77

제 72 항에 있어서,

상기 하나 이상의 프로세서들은 또한, 상기 적어도 하나의 오프셋을 결정할 때,

상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽으로부터, 상기 비디오 데이터의 블록에 적용되는 변환의 사이즈가 최대 변환 유닛 사이즈가 아닐 때 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹을 수행할지 여부를 결정하기 위해 사용되는 상기 t_c 오프셋 및 상기 β 오프셋 대신 사용되는 최대 t_c 오프셋 및 최대 β 오프셋을 추출하도록 구성되는, 비디오 디코딩 디바이스.

청구항 78

제 77 항에 있어서,

상기 하나 이상의 프로세서들은 또한,

상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽에 규정된 `de-blocking_max_tu_offset_enabled_flag` 에 기초하여, 상기 최대 t_c 오프셋 및 상기 최대 β 오프셋을 추출하도록 구성되는, 비디오 디코딩 디바이스.

청구항 79

제 72 항에 있어서,

상기 하나 이상의 프로세서들은 또한,

상기 역양자화 파라미터를 적용할 때, 상기 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 역변환의 사이즈에 기초하여 변하는 역양자화 파라미터를 적용하도록 구성되는, 비디오 디코딩 디바이스.

청구항 80

인코딩된 비디오 데이터를 디코딩하는 비디오 디코딩 디바이스로서,

비디오 비트스트림 내의 상기 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 상기 인코딩된 비디오 데이터의 블록에 역양자화 파라미터를 적용하는 수단;

잔여 비디오 데이터의 블록을 발생하기 위해 상기 인코딩된 비디오 데이터의 양자화 해제된 블록에 역변환을 적용하는 수단;

상기 잔여 비디오 데이터의 블록으로부터 비디오 데이터의 블록을 재구성하는 수단;

상기 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋이 상기 인코딩된 비디오 데이터의 블록을 포함하는 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정되는지 여부를 나타내는, 상기 비디오 비트스트림 내의 상기 인코딩된 비디오 데이터의 상기 PPS 에 규정된 플래그를 결정하는 수단;

상기 플래그에 기초하여, 상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽으로부터 상기 적어도 하나의 오프셋을 추출하는 수단으로서, 상기 적어도 하나의 오프셋은 t_c 오프셋과 β 오프셋 중 하나 이상을 포함하고, 상기 t_c 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 및 크로마 양태들 양쪽에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하며, 상기 β 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 양태들에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하는, 상기 적어도 하나의 오프셋을 추출하는 수단; 및

결정된 상기 적어도 하나의 오프셋에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하는 수단을 포함하는, 비디오 디코딩 디바이스.

청구항 81

제 80 항에 있어서,

상기 플래그는 de-blocking_filter_control_present_flag 를 포함하는, 비디오 디코딩 디바이스.

청구항 82

삭제

청구항 83

삭제

청구항 84

제 80 항에 있어서,

상기 적어도 하나의 오프셋은,

상기 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나, 또는 상기 인코딩된 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부와 연관되는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 포함하는, 비디오 디코딩 디바이스.

청구항 85

제 80 항에 있어서,

상기 적어도 하나의 오프셋을 결정하는 수단은,

상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽으로부터, 상기 비디오 데이터의 블록에 적용되는 변환의 사이즈가 최대 변환 유닛 사이즈가 아닐 때 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹을 수행할지 여부를 결정할 때에 다르게 사용되는 상기 t_c 오프셋 및 상기 β 오프셋 대신 사용되는 최대 t_c 오프셋 및 최대 β 오프셋을 추출하는 수단을 포함하는, 비디오 디코딩 디바이스.

청구항 86

제 85 항에 있어서,

상기 최대 t_c 오프셋 및 최대 β 오프셋을 추출하는 수단은,

상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽에 규정된 de-blocking_max_tu_offset_enabled_flag 에 기초하여, 상기 최대 t_c 오프셋 및 상기 최대 β 오프셋을 추출하는 수단을 포함하는, 비디오 디코딩 디바이스.

청구항 87

제 80 항에 있어서,

상기 역양자화 파라미터를 적용하는 수단은,

상기 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 역변환의 사이즈에 기초하여 변하는 역양자화 파라미터를 적용하는 수단을 포함하는, 비디오 디코딩 디바이스.

청구항 88

명령들을 저장하고 있는 비일시적 컴퓨터 판독가능 매체로서,

상기 명령들은, 실행될 때, 하나 이상의 프로세서들로 하여금,

비디오 비트스트림 내의 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 상기 인코딩된 비디오 데이터의 블록에 역양자화 파라미터를 적용하게 하고;

잔여 비디오 데이터의 블록을 발생하기 위해 상기 인코딩된 비디오 데이터의 양자화 해제된 블록에 역변환을 적용하게 하고;

상기 잔여 비디오 데이터의 블록으로부터 비디오 데이터의 블록을 재구성하게 하고;

상기 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋이 상기 인코딩된 비디오 데이터의 블록을 포함하는 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽에 규정되는지 여부를 나타내는, 상기 비디오 비트스트림 내의 상기 인코딩된 비디오 데이터의 상기 PPS 에 규정된 플래그를 결정하게 하고;

상기 플래그에 기초하여, 상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽으로부터 상기 적어도 하나의 오프셋을 추출하게 하며;

결정된 상기 적어도 하나의 오프셋에 기초하여, 상기 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하게 하고,

상기 적어도 하나의 오프셋은 t_c 오프셋과 β 오프셋 중 하나 이상을 포함하고, 상기 t_c 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 및 크로마 양태들 양쪽에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하며, 상기 β 오프셋은, 상기 비디오 데이터의 재구성된 블록의 루마 양태들에 대해서 디블록킹 필터링을 수행할지 여부를 결정하는 것 또는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 오프셋을 포함하는, 비밀시적 컴퓨터 판독가능 매체.

청구항 89

제 88 항에 있어서,

상기 플래그는 de-blocking_filter_control_present_flag 를 포함하는, 비밀시적 컴퓨터 판독가능 매체.

청구항 90

삭제

청구항 91

삭제

청구항 92

제 88 항에 있어서,

상기 적어도 하나의 오프셋은,

상기 비디오 데이터의 재구성된 블록의 상기 디블록킹 필터링을 수행할지 여부를 결정하는 것, 또는 상기 인코딩된 비디오 데이터의 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부와 연관되는 디블록킹 필터링 강도를 결정하는 것 중 하나 또는 양쪽을 위해 사용되는 적어도 하나의 오프셋을 포함하는, 비밀시적 컴퓨터 판독가능 매체.

청구항 93

제 88 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 적어도 하나의 오프셋을 결정할 때, 상기 비디오 데이터의 블록의 인코딩된 버전을 포함하는 상기 독립적으로 디코딩가능한 유닛의 헤더와 화상 파라미터 세트 (PPS) 중 하나 또는 양쪽으로부터, 상기 비디오 데이터의 블록에 적용되는 변환의 사이즈가 최대 변환 유닛 사이즈가 아닐 때 상기 비디오 데이터의 재구성된 블록의 상기 디블록킹을 수행할지 여부를 결정할 때에 다르게 사용되는 상기 t_c 오프셋 및 상기 β 오프셋 대신 사용되는 최대 t_c 오프셋 및 최대 β 오프셋을 추출하게 하는, 비밀시적 컴퓨터 판독가능 매체.

청구항 94

제 93 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

최대 t_c 오프셋 및 최대 β 오프셋을 추출할 때, 상기 독립적으로 디코딩가능한 유닛의 헤더와 상기 PPS 중 하나 또는 양쪽에 규정된 `de-blocking_max_tu_offset_enabled_flag` 에 기초하여, 상기 최대 t_c 오프셋 및 상기 최대 β 오프셋을 추출하게 하는, 비일시적 컴퓨터 판독가능 매체.

청구항 95

제 88 항에 있어서,

상기 명령들은 또한, 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

상기 역양자화 파라미터를 적용할 때, 상기 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 역변환의 사이즈에 기초하여 변하는 역양자화 파라미터를 적용하게 하는, 비일시적 컴퓨터 판독가능 매체.

청구항 96

삭제

청구항 97

삭제

청구항 98

삭제

청구항 99

삭제

청구항 100

삭제

청구항 101

삭제

청구항 102

삭제

청구항 103

삭제

청구항 104

삭제

청구항 105

삭제

청구항 106

삭제

청구항 107

삭제

청구항 108

삭제

청구항 109

삭제

청구항 110

삭제

청구항 111

삭제

청구항 112

삭제

청구항 113

삭제

청구항 114

삭제

청구항 115

삭제

청구항 116

삭제

청구항 117

삭제

청구항 118

삭제

청구항 119

삭제

청구항 120

삭제

청구항 121

삭제

청구항 122

삭제

청구항 123

삭제

청구항 124

삭제

청구항 125

삭제

청구항 126

삭제

청구항 127

삭제

청구항 128

삭제

청구항 129

삭제

청구항 130

삭제

청구항 131

삭제

청구항 132

삭제

청구항 133

삭제

청구항 134

삭제

청구항 135

삭제

청구항 136

삭제

청구항 137

삭제

청구항 138

삭제

청구항 139

삭제

청구항 140

삭제

청구항 141

삭제

청구항 142

삭제

청구항 143

삭제

청구항 144

삭제

청구항 145

삭제

청구항 146

삭제

청구항 147

삭제

청구항 148

삭제

청구항 149

삭제

청구항 150

삭제

청구항 151

삭제

청구항 152

삭제

청구항 153

삭제

청구항 154

삭제

청구항 155

삭제

청구항 156

삭제

발명의 설명

기술 분야

[0001] 본 출원은 2011년 10월 7일자에 출원된 미국 가출원 번호 제 61/544,960호, 2011년 11월 1일자에 출원된 미국 가출원 번호 제 61/554,260호, 2012년 1월 19일자에 출원된 미국 가출원 번호 제 61/588,480호, 2012년 6월 22일자에 출원된 미국 가출원 번호 제 61/663,469호, 2012년 8월 10일자에 출원된 미국 가출원 번호 제 61/682,167호, 및 2012년 9월 12일자에 출원된 미국 가출원 번호 제 61/700,167호의 이익을 주장하며, 이들 모두의 전체 내용이 본원에서 참고로 각각 포함된다.

[0002] 기술분야

[0003] 본 개시물은 비디오 코딩에 관한 것으로, 좀더 구체적으로는, 비디오 데이터를 디블록킹하는 것에 관한 것이다.

배경 기술

[0004] 디지털 비디오 능력들은 디지털 텔레비전, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인 휴대 정보 단말기들 (PDAs), 랩탑 또는 데스크탑 컴퓨터들, 디지털 카메라들, 디지털 리코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 원격 화상회의 디바이스들 등을 포함한, 광범위한 디바이스들에 포함될 수 있다. 디지털 비디오 디바이스들은 디지털 비디오 정보를 좀더 효율적으로 송수신하기 위해서, MPEG-2, MPEG-4, ITU-T H.263 또는 ITU-T H.264/MPEG-4, 파트 10, AVC (Advanced Video Coding) 에 의해 정의된 표준들, 및 이런 표준들의 확장관들에 설명된 기법들과 같은, 비디오 압축 기법들을 구현한다.

[0005] 비디오 압축 기법들은 비디오 시퀀스들에 고유한 리던던시를 감소시키거나 또는 제거하기 위해 공간 예측 및/또는 시간 예측을 수행한다. 블록-기반의 비디오 코딩에 있어, 비디오 프레임 또는 슬라이스는 매크로블록들로 파티셔닝될 수도 있다. 각각의 매크로블록은 더 파티셔닝될 수도 있다. 인트라-코딩된 (I) 프레임 또는 슬라이스에서 매크로블록들은 이웃하는 매크로블록들에 대한 공간 예측을 이용하여 인코딩된다. 인터-코딩된 (P 또는 B) 프레임 또는 슬라이스에서의 매크로블록들은 동일한 프레임 또는 슬라이스에서의 이웃하는 매크로블록들에 대한 공간 예측 또는 다른 참조 프레임들에 대한 시간 예측을 이용할 수도 있다.

발명의 내용

과제의 해결 수단

[0006] 일반적으로, 본 개시물은 비디오 블록들의 잔여 비디오 데이터를 공간 도메인으로부터 주파수 도메인으로 변환하는데 사용되는 변환의 사이즈에 기초하여, 비디오 블록들 사이에 예지들의 디블록킹을 수행하는 기법들을 설명한다. 예를 들어, 이 기법들은 비디오 인코더 및/또는 비디오 디코더로 하여금, 이들 잔여 비디오 데이터 블록들을 변환하는데 사용되는 변환의 사이즈에 기초하여, 디블록킹을 수행하는데 사용되는 여러 오프셋들을 선택가능하게 할 수도 있다. 더욱이, 이 기법들은 비디오 인코더 및/또는 비디오 디코더로 하여금, 변환의 사이즈 뿐만 아니라, 현재의 비디오 블록이 인트라-코딩되었는지 또는 인터-코딩되었는지 여부에 기초하여, 디블록킹을 수행하는데 사용되는 여러 오프셋들을 선택가능하게 할 수도 있다. 본 개시물의 기법들은 인코딩, 후속 디코딩 및 디스플레이 후 최종 비디오 데이터의 인지되는 시각적 품질을 향상시킬 수도 있다. 이 기법들은 또한 종래의 기법들을 이용하여 발생된 인코딩된 비디오 데이터에 비해, 최종 인코딩된 비디오 데이터의 사이즈의 관점에서 코딩 이득들을 증진시킬 수도 있다.

[0007] 일 양태에서, 비디오 데이터를 인코딩하는 방법은, 변환 계수들의 블록을 발생하기 위해 비디오 데이터의 블록에 변환을 적용하는 단계, 및 변환 계수들의 블록을 양자화하기 위해 양자화 파라미터를 적용하는 단계를 포함한다. 본 방법은 변환 계수들의 양자화된 블록으로부터 비디오 데이터의 블록을 재구성하는 단계, 그 적용된 변환의 사이즈에 기초하여, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하는 단계, 및 그 결정된 적어도 하나의 오프셋에 기초하여, 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하

는 단계를 더 포함한다.

[0008] 또 다른 양태에서, 비디오 데이터를 인코딩하도록 구성되는 비디오 인코딩 디바이스는, 변환 계수들의 블록을 발생하기 위해 비디오 데이터의 블록에 변환을 적용하고, 변환 계수들의 블록을 양자화하기 위해 양자화 파라미터를 적용하고, 변환 계수들의 양자화된 블록으로부터 비디오 데이터의 블록을 재구성하고, 그 적용된 변환의 사이즈에 기초하여, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하고, 그리고 그 결정된 적어도 하나의 오프셋에 기초하여, 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하도록 구성되는 하나 이상의 프로세서들을 포함한다.

[0009] 또 다른 양태에서, 비디오 데이터를 인코딩하도록 구성되는 비디오 인코딩 디바이스는, 변환 계수들의 블록을 발생하기 위해 비디오 데이터의 블록에 변환을 적용하는 수단, 및 변환 계수들의 블록을 양자화하기 위해 양자화 파라미터를 적용하는 수단을 포함한다. 비디오 인코딩 디바이스는 변환 계수들의 양자화된 블록으로부터 비디오 데이터의 블록을 재구성하는 수단, 그 적용된 변환의 사이즈에 기초하여, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하는 수단, 및 그 결정된 적어도 하나의 오프셋에 기초하여, 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하는 수단을 더 포함한다.

[0010] 또 다른 양태에서, 실행될 때, 하나 이상의 프로세서들로 하여금, 변환 계수들의 블록을 발생하기 위해 비디오 데이터의 블록에 변환을 적용하고, 변환 계수들의 블록을 양자화하기 위해 양자화 파라미터를 적용하고, 변환 계수들의 양자화된 블록으로부터 비디오 데이터의 블록을 재구성하고, 그 적용된 변환의 사이즈에 기초하여, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하고, 그리고 그 결정된 적어도 하나의 오프셋에 기초하여, 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하도록 하는 명령들을 저장하고 있는 비일시적 컴퓨터 판독가능 저장 매체.

[0011] 또 다른 양태에서, 인코딩된 비디오 데이터를 비디오 디코더로 디코딩하는 방법은, 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 그 인코딩된 비디오 데이터의 블록에 역양자화 파라미터를 적용하는 단계, 및 잔여 비디오 데이터의 블록을 발생하기 위해 그 인코딩된 비디오 데이터의 양자화 해제된 블록에 역변환을 적용하는 단계를 포함한다. 본 방법은 잔여 비디오 데이터의 블록으로부터 비디오 데이터의 블록을 재구성하는 단계, 및 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하는 단계를 더 포함한다. 적어도 하나의 오프셋은 인코딩된 비디오 데이터의 블록을 인코딩하는데 적용되는 변환의 사이즈와 연관된다. 본 방법은 또한 그 결정된 적어도 하나의 오프셋에 기초하여, 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하는 단계를 포함한다.

[0012] 또 다른 양태에서, 인코딩된 비디오 데이터를 디코딩하도록 구성되는 비디오 디코딩 디바이스는, 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 그 인코딩된 비디오 데이터의 블록에 역양자화 파라미터를 적용하고, 잔여 비디오 데이터의 블록을 발생하기 위해 그 인코딩된 비디오 데이터의 양자화 해제된 블록에 역변환을 적용하고, 잔여 비디오 데이터의 블록으로부터 비디오 데이터의 블록을 재구성하고, 그리고 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하도록 구성되는 하나 이상의 프로세서들을 포함한다. 적어도 하나의 오프셋은 인코딩된 비디오 데이터의 블록을 인코딩하는데 적용되는 변환의 사이즈와 연관된다. 하나 이상의 프로세서들은 또한, 그 결정된 적어도 하나의 오프셋에 기초하여, 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하도록 구성된다.

[0013] 또 다른 양태에서, 디코딩 인코딩된 비디오 데이터를 디코딩하는 비디오 디코딩 디바이스는, 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 그 인코딩된 비디오 데이터의 블록에 역양자화 파라미터를 적용하는 수단, 및 잔여 비디오 데이터의 블록을 발생하기 위해 그 인코딩된 비디오 데이터의 양자화 해제된 블록에 역변환을 적용하는 수단을 포함한다. 비디오 디코딩 디바이스는 잔여 비디오 데이터의 블록으로부터 비디오 데이터의 블록을 재구성하는 수단, 및 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하는 수단을 더 포함한다. 적어도 하나의 오프셋은 인코딩된 비디오 데이터의 블록을 인코딩하는데 적용되는 변환의 사이즈와 연관된다. 비디오 디코딩 디바이스는 그 결정된 적어도 하나의 오프셋에 기초하여, 비디오 데이터의

재구성된 블록에 대해 디블록킹 필터링을 수행하는 수단을 더 포함한다.

[0014] 또 다른 양태에서, 실행될 때, 하나 이상의 프로세서들로 하여금, 인코딩된 비디오 데이터의 블록을 양자화 해제하기 위해 그 인코딩된 비디오 데이터의 블록에 역양자화 파라미터를 적용하고, 잔여 비디오 데이터의 블록을 발생하기 위해 그 인코딩된 비디오 데이터의 양자화 해제된 블록에 역변환을 적용하고, 잔여 비디오 데이터의 블록으로부터 비디오 데이터의 블록을 재구성하고, 인코딩된 비디오 데이터의 블록을 인코딩하는데 적용되는 변환의 사이즈와 연관되는, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하고, 그리고, 그 결정된 적어도 하나의 오프셋에 기초하여, 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행하도록 하는 명령을 저장하고 있는 비일시적 컴퓨터 판독가능 매체.

[0015] 또 다른 양태에서, 비디오 데이터를 코딩하는 방법은 경계 강도 값에 기초하여, 하나 이상의 디블록킹 필터 클리핑 임계치들을 결정하는 단계, 클리핑된 디블록킹 필터를 결정하기 위해, 그 결정된 하나 이상의 디블록킹 필터 클리핑 임계치들을 이용하여 디블록킹 필터에 대해 클리핑 동작 (clipping operation) 을 수행하는 단계, 및 그 클리핑된 디블록킹 필터를 비디오 데이터의 블록에 적용하는 단계를 포함한다.

[0016] 또 다른 양태에서, 비디오 데이터를 코딩하는 비디오 코딩 디바이스는, 경계 강도 값에 기초하여, 하나 이상의 디블록킹 필터 클리핑 임계치들을 결정하고, 클리핑된 디블록킹 필터를 결정하기 위해 그 결정된 하나 이상의 디블록킹 필터 클리핑 임계치들을 이용하여 디블록킹 필터에 대해 클리핑 동작을 수행하고, 그리고 그 클리핑된 디블록킹 필터를 비디오 데이터의 블록에 적용하도록 구성되는 하나 이상의 프로세서들을 포함한다.

[0017] 또 다른 양태에서, 비디오 데이터를 코딩하는 비디오 코딩 디바이스로서, 상기 비디오 코딩 디바이스는 경계 강도 값에 기초하여, 하나 이상의 디블록킹 필터 클리핑 임계치들을 결정하는 수단, 클리핑된 디블록킹 필터를 결정하기 위해 그 결정된 하나 이상의 디블록킹 필터 클리핑 임계치들을 이용하여 디블록킹 필터에 대해 클리핑 동작을 수행하는 수단, 및 그 클리핑된 디블록킹 필터를 비디오 데이터의 블록에 적용하는 수단을 포함한다.

[0018] 또 다른 양태에서, 실행될 때, 비디오 코딩 디바이스의 하나 이상의 프로세서들로 하여금, 경계 강도 값에 기초하여, 하나 이상의 디블록킹 필터 클리핑 임계치들을 결정하고, 클리핑된 디블록킹 필터를 결정하기 위해 그 결정된 하나 이상의 디블록킹 필터 클리핑 임계치들을 이용하여 디블록킹 필터에 대해 클리핑 동작을 수행하고, 그리고 그 클리핑된 디블록킹 필터를 비디오 데이터의 블록에 적용하도록 하는 명령들을 저장하고 있는 비일시적 컴퓨터 판독가능 저장 매체.

[0019] 하나 이상의 예들의 세부 사항들이 첨부도면 및 아래의 상세한 설명에서 개시된다. 다른 특성들, 목적들, 및 이점들은 설명 및 도면들로부터, 그리고 청구항들로부터 명백히 알 수 있을 것이다.

도면의 간단한 설명

[0020] 도 1 은 본 개시물에서 설명하는 기법들의 여러 양태들을 수행하는 예시적인 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.

도 2 는 본 개시물에서 설명하는 기법들의 여러 양태들을 수행하는 예시적인 비디오 인코더를 예시하는 블록도이다.

도 3 은 본 개시물에서 설명하는 기법들의 여러 양태들을 수행하는 예시적인 비디오 디코더를 예시하는 블록도이다.

도 4 는 디블록킹 필터링을 제어하는데 사용되는 오프셋들이 슬라이스 헤더에 포함되는지 여부를 나타내는 플래그를 비트스트림의 화상 파라미터 세트 (picture parameter set; PPS) 에 규정 (specify) 할 때에 비디오 인코더의 예시적인 동작을 예시하는 플로우차트이다.

도 5 는 디블록킹 필터링을 제어하는데 사용되는 오프셋들을 적용 파라미터 세트 (APS) 에 규정할 때에 비디오 인코더의 예시적인 동작을 예시하는 플로우차트이다.

도 6 은 디블록킹 필터링을 제어하는데 사용되는 오프셋들이 슬라이스 헤더에 포함되는지 여부를 나타내는 플래그를 화상 파라미터 세트 (PPS) 로부터 추출할 때에 비디오 디코더의 예시적인 동작을 예시하는 플로우차트이다.

도 7 은 본 개시물에서 설명하는 기법들에 따라 적용 파라미터 세트로부터 디블록킹 필터링을 제어하는데 사용

되는 오프셋들을 추출할 때에 비디오 디코더의 예시적인 동작을 예시하는 플로우차트이다.

도 8 은 디블록킹 필터 파라미터들을 계산할 때에 사용되는 경계 강도 값을 결정하는 본 기법들의 여러 양태들을 구현할 때에 비디오 코더의 예시적인 동작을 예시하는 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

- [0021] 도 1 은 변환 의존적인 디블록킹 필터링을 촉진하기 위해 여러 양태들 본 개시물에서 설명하는 기법들을 이용하도록 구성될 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (10) 을 예시하는 블록도이다. 도 1 에 나타난 바와 같이, 시스템 (10) 은 인코딩된 비디오를 목적지 디바이스 (14) 로 통신 채널 (16) 을 통해서 송신하는 소스 디바이스 (12) 를 포함한다. 인코딩된 비디오 데이터는 또한 저장 매체 (34) 또는 파일 서버 (36) 상에 저장될 수도 있으며, 원하는 바에 따라 목적지 디바이스 (14) 에 의해 액세스될 수도 있다. 저장 매체 또는 파일 서버에 저장될 때, 비디오 인코더 (20) 는 코딩된 비디오 데이터를 네트워크 인터페이스, 콤팩트 디스크 (CD), 블루레이 또는 디지털 비디오 디스크 (DVD) 버너 또는 스탬핑 기능 디바이스, 또는 다른 디바이스들과 같은, 코딩된 비디오 데이터를 저장 매체에 저장하기 위한, 또 다른 디바이스에 제공할 수도 있다. 이와 유사하게, 네트워크 인터페이스, CD 또는 DVD 리더, 또는 기타 등등과 같은 비디오 디코더 (30) 로부터 분리된 디바이스가, 저장 매체로부터 코딩된 비디오 데이터를 추출하고 그 추출된 데이터를 비디오 디코더 (30) 로 제공할 수도 있다.
- [0022] 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 태블릿 컴퓨터들, 셋-탑 박스들, 소위 스마트폰들과 같은 전화기 핸드셋들, 텔레비전, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들, 또는 기타 등등을 포함하는, 매우 다양한 디바이스들 중 임의의 디바이스를 포함할 수도 있다. 많은 경우에, 이런 디바이스들은 무선 통신용으로 탑재될 수도 있다. 그러므로, 통신 채널 (16) 은 인코딩된 비디오 데이터의 송신에 적합한, 무선 채널, 유선 채널, 또는 무선 채널과 유선 채널의 조합을 포함할 수도 있다. 이와 유사하게, 파일 서버 (36) 는 인터넷 접속을 포함하는, 임의의 표준 데이터 접속을 통해서, 목적지 디바이스 (14) 에 의해 액세스될 수도 있다. 이것은 파일 서버 상에 저장된 인코딩된 비디오 데이터에 액세스하는데 적합한, 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀 등), 또는 양쪽의 조합을 포함할 수도 있다.
- [0023] 본 개시물의 예들에 따른, 변환 의존적인 디블록킹 필터링을 수행하는 기법들은, 오버-디-에어 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 예컨대, 인터넷을 통한 스트리밍 비디오 송신들, 데이터 저장 매체 상에의 저장을 위한 디지털 비디오의 인코딩, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션들과 같은 다양한 멀티미디어 애플리케이션들 중 임의의 애플리케이션의 지원 하에, 비디오 코딩에 적용될 수도 있다. 일부 예들에서, 시스템 (10) 은 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 및/또는 비디오 전화 통신과 같은, 지원 애플리케이션들로의 1-방향 또는 2-방향 비디오 송신을 지원하도록 구성될 수도 있다.
- [0024] 도 1 의 예에서, 소스 디바이스 (12) 는 비디오 소스 (18), 비디오 인코더 (20), 변조기/복조기 (22) 및 송신기 (24) 를 포함한다. 소스 디바이스 (12) 에서, 비디오 소스 (18) 는 비디오 캡처 디바이스, 예컨대 비디오 카메라, 이전에 캡처된 비디오를 포함하는 비디오 아카이브, 비디오 콘텐츠 제공자로부터 비디오를 수신하는 비디오 공급 인터페이스, 및/또는 컴퓨터 그래픽스 데이터를 소스 비디오로서 발생하는 컴퓨터 그래픽스 시스템, 또는 이런 소스들의 조합과 같은 소스를 포함할 수도 있다. 일 예로서, 비디오 소스 (18) 가 비디오 카메라이면, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 소위 카메라 폰들 또는 비디오 폰들일 수도 있다. 그러나, 본 개시물에서 설명하는 기법들은 일반적으로는 비디오 코딩에 적용가능할 수도 있으며, 무선 및/또는 유선 애플리케이션들, 또는 인코딩된 비디오 데이터가 로컬 디스크 상에 저장되는 애플리케이션에 적용될 수도 있다.
- [0025] 캡처되거나, 사전-캡처 (pre-captured) 되거나, 또는 컴퓨터-발생된 비디오는 비디오 인코더 (20) 에 의해 인코딩될 수도 있다. 인코딩된 비디오 정보는 무선 통신 프로토콜과 같은, 통신 표준에 따라서 모뎀 (22) 에 의해 변조되어, 송신기 (24) 를 통해서 목적지 디바이스 (14) 로 송신될 수도 있다. 모뎀 (22) 은 여러 믹서들, 필터들, 증폭기들 또는 신호 변조용으로 설계된 다른 구성요소들을 포함할 수도 있다. 송신기 (24) 는 증폭기들, 필터들, 및 하나 이상의 안테나들을 포함한, 데이터를 송신하도록 설계된 회로들을 포함할 수도 있다.
- [0026] 비디오 인코더 (20) 에 의해 인코딩되는 캡처된, 사전-캡처된, 또는 컴퓨터-발생된 비디오는 또한 추후 소비를

위해 저장 매체 (34) 또는 파일 서버 (36) 상에 저장될 수도 있다. 저장 매체 (34) 는 블루-레이 디스크들, DVDs, CD-ROMs, 플래시 메모리, 또는 인코딩된 비디오를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체들을 포함할 수도 있다. 저장 매체 (34) 상에 저장된 인코딩된 비디오는 그 후 디코딩 및 플레이백을 위해 목적지 디바이스 (14) 에 의해 액세스될 수도 있다.

[0027] 파일 서버 (36) 는 인코딩된 비디오를 저장하고 그 인코딩된 비디오를 목적지 디바이스 (14) 로 송신할 수 있는 임의의 유형의 서버일 수도 있다. 예시적인 파일 서버들은 웹 서버 (예컨대, 웹사이트용), FTP 서버, 네트워크 부착된 스토리지 (NAS) 디바이스들, 로컬 디스크 드라이브, 또는 인코딩된 비디오 데이터를 저장하고 이를 목적지 디바이스로 송신할 수 있는 임의의 다른 유형의 디바이스들을 포함한다. 파일 서버 (36) 로부터의 인코딩된 비디오 데이터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이 양쪽의 조합일 수도 있다. 파일 서버 (36) 는 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해서, 목적지 디바이스 (14) 에 의해 액세스될 수도 있다. 이것은 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀, 이더넷, USB 등), 또는 파일 서버 상에 저장된 인코딩된 비디오 데이터에 액세스하는데 적합한 이들 양쪽의 조합을 포함할 수도 있다.

[0028] 목적지 디바이스 (14) 는, 도 1 의 예에서, 수신기 (26), 모뎀 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32) 를 포함한다. 목적지 디바이스 (14) 의 수신기 (26) 는 채널 (16) 을 통해서 정보를 수신하고, 모뎀 (28) 은 비디오 디코더 (30) 를 위한 복조된 비트스트림을 발생한다. 채널 (16) 을 통해서 통신된 정보는 비디오 데이터를 디코딩할 때에 비디오 디코더 (30) 에 의한 사용을 위해 비디오 인코더 (20) 에 의해 발생된 다양한 구문 (syntax) 정보를 포함할 수도 있다. 위에서 설명한 슬라이스 헤더 구문을 포함한, 이런 구문은 또한 저장 매체 (34) 또는 파일 서버 (36) 상에 저장된 인코딩된 비디오 데이터에 포함될 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 비디오 데이터를 인코딩 또는 디코딩할 수 있는 각각의 인코더-디코더 (코덱) 의 부분을 형성할 수도 있다.

[0029] 디스플레이 디바이스 (32) 는 목적지 디바이스 (14) 와 통합되거나 이의 외부에 있을 수도 있다. 일부 예들에서, 목적지 디바이스 (14) 는 통합된 디스플레이 디바이스를 포함하며, 또한 외부 디스플레이 디바이스와 인터페이스하도록 구성될 수도 있다. 다른 예들에서, 목적지 디바이스 (14) 는 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32) 는 그 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 액정 디스플레이 (LCD), 플라스마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 또 다른 유형의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 디바이스를 포함할 수도 있다.

[0030] 도 1 의 예에서, 통신 채널 (16) 은 임의의 무선 또는 유선 통신 매체, 예컨대 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적인 송신 라인들, 또는 무선 매체와 유선 매체들의 임의의 조합을 포함할 수도 있다. 통신 채널 (16) 은 근거리 네트워크, 광역 네트워크, 또는 글로벌 네트워크, 예컨대 인터넷과 같은 패킷-기반 네트워크의 일부를 형성할 수도 있다. 통신 채널 (16) 은 일반적으로 유선 또는 무선 매체들의 임의의 적합한 조합을 포함한, 비디오 데이터를 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 송신하는, 임의의 적합한 통신 매체, 또는 상이한 통신 매체들의 컬렉션을 나타낸다. 통신 채널 (16) 은 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 통신을 촉진하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0031] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 현재 개발중인 HEVC (High Efficiency Video Coding) 표준과 같은 비디오 압축 표준에 따라서 동작할 수도 있으며, HEVC 테스트 모델 (HM) 에 따를 수도 있다. 이의 대안으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 MPEG 4, 파트 10, AVC (Advanced Video Coding) 로서 대안적으로 지칭되는 ITU-T H.264 표준, 또는 이런 표준들의 확장판들과 같은 다른 사유 (proprietary) 또는 산업 표준들에 따라서 동작할 수도 있다. 본 개시물의 기법들은, 그러나, 임의의 특정의 코딩 표준에 한정되지 않는다. 다른 예들은 MPEG-2 및 ITU-T H.263 를 포함한다.

[0032] 도 1 에 나타내지는 않지만, 일부 양태들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 오디오 인코더 및 디코더와 각각 통합될 수도 있으며, 오디오 및 비디오 양쪽의 인코딩을 공통 데이터 스트림 또는 별개의 데이터 스트림들로 처리하기 위해 적합한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능한 경우, 일부 예들에서, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜, 또는 다른 프로토콜들, 예컨대, 사용자 데이터그램 프로토콜 (UDP) 을 따를 수도 있다.

[0033] 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 다양한 적합한 인코더 회로, 예컨대 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSPs), 주문형 집적회로들 (ASICs), 필드 프로그래밍가능 게이트 어레이들

(FPGAs), 이산 로직, 소프트웨어, 하드웨어, 펌웨어 또는 임의의 이들의 조합들 중 임의의 것으로 구현될 수도 있다. 이 기법들이 소프트웨어로 부분적으로 구현되는 경우, 디바이스는 본 개시물의 기법들을 수행하기 위해 소프트웨어용 명령들을 적합한 비일시적 컴퓨터 판독가능 매체에 저장하고, 그 명령들을 하드웨어에서 하나 이상의 프로세서들을 이용하여 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이들 중 어느 쪽이든 각각 디바이스에서 결합된 인코더/디코더 (코덱) 의 부분으로서 통합될 수도 있다.

[0034] 일부의 경우, 비디오 인코더 (20) 는 HM 을 구현하여, 비디오 코딩을 수행할 수도 있다. HM 은 디블록킹 필터들을 이용하여 디블록킹에 제공하여, "블로키니스 (blockiness)" 아티팩트 (artifact) 들을 제거한다. 디블록킹은 프레임을 블록들 (LCU들 및 그의 서브-CU들) 로 분할하고, 그 블록들을 코딩하고, 그 후 그 블록들을 디코딩한 후 수행될 수도 있다. 즉, 비디오 인코더 (20) 는, HM 에 따라서 동작할 때, 블록들을 결정하고, (모션 추정 및 보상과 같은) 인트라-예측 또는 인터-예측을 수행하여 잔여 데이터를 발생하고, 잔여 데이터를 공간 도메인으로부터 주파수 도메인으로 변환하고, 그리고 이산 코사인 변환들 (DCTs) 과 같은 하나 이상의 변환들을 적용하여 DCT 주파수 도메인에서 잔여 데이터를 나타내는 계수들을 출력하고, 그리고 DCT 계수들을 양자화할 수도 있으며, 여기서 이런 양자화는 잔여 데이터에 적용된 변환의 사이즈에 기초하여 변환할 수도 있다. 비디오 인코더 (20) 는 그 후 인코딩된 블록들 (이 상황에서는, 양자화된 DCT 계수들을 의미함) 을 재구성하여, 인터-예측을 수행할 때에 참조 블록들로서 사용되는 비디오 데이터의 재구성된 블록들을 형성할 수도 있다. 비디오 블록들을 인코딩한 후 이들 블록들을 재구성하는 이 프로세스는 인지가능한 아티팩트들을 프레임에서 이웃하는 블록들 사이의 에지들에 삽입할 수도 있다.

[0035] 그 결과, 비디오 인코더 (20) 는 프레임의 비디오 데이터를 인코딩하고, 그 후 후속하여 인코딩된 비디오 데이터를 디코딩하고, 그리고 그 후 참조 비디오 데이터로서의 사용을 위해, 예컨대, 후속하여 코딩되는 비디오 데이터의 인터-예측을 위해, 디블록킹 필터들을 디코딩된 비디오 데이터에 적용할 수도 있다. 비디오 디코더 (30) 와 같은, 비디오 디코더들은 유사한 동작들을 수행하여 인코딩된 비디오 데이터로부터 비디오 데이터를 재구성하고, 그 후 디블록킹을 수행하여 프레임에서 이웃하는 블록들 사이의 에지들에서 인지가능한 아티팩트들의 도입을 유사하게 감소시킬 수도 있다.

[0036] 참조 데이터로서 사용하기 위해 데이터를 저장하기 전에 (이 상황에서는, 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 로 지칭될 수도 있는) 비디오 코딩 디바이스에 의해 수행되는 이런 디블록킹 필터링은 일반적으로 필터링이 코딩 루프 내에서 수행된다는 점에서, "인-루프" 필터링으로서 지칭된다. 비디오 디코더 (20) 는 비디오 데이터를 디스플레이하는 목적들을 위해서, 뿐만 아니라, 인코딩된 또는 디코딩된 후속 비디오 데이터에 대한 참조 데이터로서 사용을 위해서, 수신된 비디오 데이터를 디코딩하고, 그 후 동일한 디블록킹 필터들을 디코딩된 비디오 데이터에 적용하도록 구성될 수도 있다. 동일한 디블록킹 기법들을 적용하도록 인코더 (20) 및 디코더 (30) 양쪽을 구성함으로써, 인코더 (20) 및 디코더 (30) 는, 디블록킹이 참조를 위한 디블록킹된 비디오 데이터를 이용하는 후속하여 코딩되는 비디오 데이터에 에러를 도입하지 않도록, 동기화될 수 있다.

[0037] HM 은 일반적으로 2개의 PU들 또는 TU들을 분리하는 각각의 에지에 대해, 에지를 디블록킹하기 위해 디블록킹 필터를 적용할지 여부를 결정하도록 구성된다. HM 은 에지에 수직한 픽셀들의 하나 이상의 라인들, 예컨대, 8개의 픽셀들의 라인의 분석에 기초하여 디블록킹할지 여부를 결정하도록 구성될 수도 있다. 따라서, 예를 들어, 수직 에지에 있어, HM 은 공통 라인을 따라서 에지의 좌측으로 4개의 픽셀들, 그리고 에지의 우측으로 4개의 픽셀들을 검사함으로써, 에지를 디블록킹할지 여부를 검사할 수도 있다. 선택된 픽셀들의 개수는 일반적으로 디블록킹을 위한 가장 작은 블록, 예컨대, 8 x 8 픽셀들에 대응한다. 이러한 방법으로, 분석에 사용되는 픽셀들의 라인은 단지 2개의 PU들 또는 TU들, 즉 바로 에지의 좌측 및 에지의 우측 PU들 또는 TU들에 들어갈 수도 있다. 에지에 대한 디블록킹을 수행할지 여부를 분석에 사용되는 픽셀들의 라인은 또한 서포트 픽셀들의 세트, 또는 간단히 "서포트" 로서 지칭된다.

[0038] 비디오 코딩 디바이스들 (20, 30) 은 현재의 HM 의 버전에 따라서 특정의 에지에 대한 서포트에 대해 디블록킹 결정 기능들을 실행하도록 구성될 수도 있다. 일반적으로, 디블록킹 결정 기능들은 현재의 HM 의 버전에서 그 서포트 내에서 높은-주파수 변화들을 검출하도록 구성된다. 이들 결정들은 종종 소위 " t_c 임계치" (종종, " t_c " 로서 지칭됨) 및 " β 임계치" (종종, " β " 로서 지칭됨) 와 같은, 임계치들에 의존한다. 게다가, t_c 및 β 임계치들은 또한 디블록킹 필터링의 강도에 영향을 미칠 수도 있다. 비디오 코딩 디바이스 (20, 30) 는 루마 및 크로마 샘플들 양쪽을 디블록킹하기 위해 t_c 임계치를 적용할 수도 있지만, 비디오 코딩 디바이스는 루마 샘플 디블록킹을 수행하기 위해 β 임계치를 적용한다. 이들 임계치들은 일반적으로 양자

화 단계 사이즈를 정의하는 양자화 파라미터 (QP) 및 블록 경계를 가로질러서 픽셀 값들의 코히어런스를 정의하는 경계 강도 (BS)로부터 유도되는 파라미터 Q를 이용하여 결정된다. (HM에서 정의되는) 소위 TcOffset 값은 경계 강도의 값에 따라서 tc 임계치에 대한 Q 파라미터를 결정하는데 사용되는 QP를 수정한다.

[0039] 예시하기 위하여, HM은 먼저 다음 수식 (1)에 따라서 QP를 수정하는데 사용되는 이 TcOffset를 결정하도록 구성된다:

[0040] $TcOffset = (BS > 2) ? DEFAULT_INTRA_TC_OFFSET(=2) : 0$ (1)

[0041] 수식 (1)에 따르면, BS가 2보다 더 크면, 현재의 블록 또는 이웃하는 블록은 인트라 코딩되며 TcOffset는 2인 디폴트 인트라-코딩된 TcOffset ("DEFAULT_INTRA_TC_OFFSET")와 동일하다. 그렇지 않으면, BS가 2보다 작거나 또는 동일하면, 현재의 블록은 인터-코딩되며 TcOffset는 제로와 동일하다. 다음으로, 비디오 코딩 디바이스는 다음 수식 (2)에 따라서 tc 임계치에 대한 Q 값을 결정한다:

[0042] $t_c: Q = Clip3(0, MAX_QP+4, QP + TcOffset)$ (2)

[0043] 수식 (2)에서, Q 값은 최종 Q 값이 0과 MAX_QP+4 (여기서, MAX_QP는 일반적으로 51과 동일하다)의 범위 내에 들어가도록 클리핑하는 범위를 조건으로 하여, tc 임계치에 대한 QP 플러스 그 결정된 TcOffset 값으로서 결정된다.

[0044] 다음으로, 현재의 HM의 버전은 수식 (3)에 따라서 β 임계치에 대한 Q 값을 결정하도록 구성될 수도 있다:

[0045] $\beta: Q = Clip3(0, MAX_QP, QP)$ (3)

[0046] 수식 (3)은 제로 내지 상수 MAX_QP의 범위 내에 들어가도록 최종 Q 값을 클리핑하는 것을 조건으로 하여, Q의 값을 QP의 값으로 설정한다. tc 및 β 임계치들 양쪽에 대한 결정된 Q 값을 이용하여, 현재의 HM의 버전은 다음 테이블 1에서 tc 및 β 임계치들의 각각의 값들을 탐색할 수도 있다:

[0047] 테이블 1

Q	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
β	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	7	8	
tc	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Q	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37
β	9	10	11	12	13	14	15	16	17	18	20	22	24	26	28	30	32	34	36
tc	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4
Q	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	
β	38	40	42	44	46	48	50	52	54	56	58	60	62	64	64	64	64	64	
tc	5	5	6	6	7	8	9	9	10	10	11	11	12	12	13	13	14	14	

[0048]

[0049] tc 및 β 임계치들에 대한 값들을 이용하여, HM은 주어진 상부 또는 좌측 경계에 대해 디블록킹을 수행할지 여부를 결정하고 디블록킹 필터링 강도를 결정하도록 구성될 수도 있다. 유사한 프로세스 (그리고, 가능한 한 동일한 프로세스)가 비디오 디코딩을 수행할 때에 사용하기 위해 HEVC의 현재의 제안된 버전에 규정 (specify) 될 수도 있다. HEVC의 현재의 제안된 버전은 2012년 7월 11일 내지 2012년 7월 20일, 스웨덴, 스톡홀름: 10차 회의, Bross 등, "HEVC (High Efficiency Video Coding) Text Specification Draft 8", ITU-T SG16 WP3 과 ISO/IEC JTC1/SC29/WG11의 JCT-VC (Joint Collaborative Team on Video Coding), 문서 HCTVC-J1003_d7 이다.

[0050] HEVC 및 HM의 현재의 제안된 버전에서는, 그러나, 디블록킹 필터 거동은 자각할 수 있는 품질을 향상시키거나 또는 비트 레이트 효율을 향상시키기 위해 인코더에 의해 영향을 받지 않을 수 있다. 이 제한 사항을 완화하기 위해, HEVC의 현재의 버전을 수정하기 위한 최근의 제안들은 디블록킹 필터 거동의 변경을 가능하게 하는 slice_tc_offset 및 slice_beta_offset의 슬라이스-레벨 시그널링을 제안하고 있다. 이 제안은 ("HM4"로서 지칭되는 테스트 모델의 제 4 버전에 대해 정의된) 앵커 시퀀스들과 유사한 자각할 수 있는 품질을 유지하면서 BD-레이트 이득들 (Bjontegaard Delta-rate gains)을 명시하지만, 변환 사이즈 독립적인, 하나의

slice_tc_offset 및 하나의 slice_beta_offset 의 시그널링은 상이한 변환 사이즈들 (4 x 4, 8 x 8, 16 x 16, 32 x 32) 에 대응하는 블록 아티팩트들과 연관된 현재하게 상이한 자각할 수 있는 품질 열화 때문에, 가장 효율적인 솔루션이 아닐 수도 있다. 따라서, 하나의 오프셋을 QP 값에 적용하는 영향은 각각의 변환 유형에 대해 상이한 자각할 수 있거나 또는 BD-레이트 영향을 가질 수도 있다.

[0051] 변환 사이즈에 대해 정적으로 정의된 t_c 오프셋 및 β 오프셋은, 인코더들로 하여금, 자각할 수 있는 인지된 비디오 품질을 증진시키거나 및/또는 모든 변환 사이즈들에 대한 코딩 이득들을 증진시키기 위해 적합한 오프셋들을 완전히 정의할 수 없게 할 수도 있다.

[0052] 본 개시물에서 정의된 기법들에 따르면, 이 기법들은 (비디오 인코더 또는 비디오 디코더와 같은) 비디오 코딩 디바이스로 하여금, 잔여 비디오 데이터를 공간 도메인으로부터 주파수 도메인으로 변환하는데 사용되는 변환의 사이즈에 기초하여, 비디오 블록들 사이의 에지들의 디블록킹을 수행가능하게 할 수도 있다. 예를 들어, 이 기법들은 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 로 하여금, 이들 잔여 데이터 블록들을 변환하는데 사용되는 변환의 사이즈에 기초하여, 디블록킹을 수행할지 여부를 결정할 때에 사용되는 여러 오프셋들을 선택가능하게 할 수도 있다. 더욱이, 이 기법들은 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 로 하여금, 변환의 사이즈 뿐만 아니라 현재의 비디오 블록이 인트라- 또는 인터-코딩되었는지 여부에 기초하여, 디블록킹을 수행할지 여부를 결정할 때에 사용되는 여러 오프셋들을 선택가능하게 할 수도 있다. 본 개시물의 기법들은 인코딩, 후속 디코딩 및 디스플레이 후 최종 비디오 데이터의 인지되는 시각적 품질을 추가로 향상시킬 수도 있다. 이 기법들은 또한 종래의 기법들을 이용하여 발생된 인코딩된 비디오 데이터에 비해, 최종 인코딩된 비디오 데이터의 사이즈의 관점에서 코딩 이득들을 증진시킬 수도 있다.

[0053] 단일 slice_tc_offset 및 slice_beta_offset 를 슬라이스 헤더로 시그널링하는 대신, 비디오 인코더 (20) 는 본 개시물에서 설명하는 기법들을 구현하여, 변환 유형 (4 x 4, 8 x 8, 16 x 6, 32 x 32) 당 하나의 slice_tc_offset 및 하나의 slice_beta_offset 까지 슬라이스 헤더로 시그널링할 수도 있다. 블록 X 와 블록 Y 사이에 공유되며 여기서 X 가 Y 의 좌측 또는 상부에 있는 에지의 경우에, 블록 Y 의 변환 유형이 사용될 수도 있거나 또는 규칙이 예를 들어, X 및 Y 로부터 (또한, "유형" 으로서 지칭될 수도 있는) 최소/최대/평균 변환 사이즈를 결정하기 위해 X 및 Y 의 변환 유형들 (사이즈들) 에 적용될 수도 있다. 변환 의존성에 더해, slice_tc_offset 및 slice_beta_offset 이 인트라 또는 인터 코딩 모드들에 대해 별개로 시그널링될 수도 있다. 일부의 경우, (종종, "비-정사각형 쿼드트리 변환들" 또는 "NSQT" 로서 지칭되는) 직사각형의 변환 유형들이 채용될 때, 동등한 정사각형 변환 유형이 사용될 수 있다. 예를 들어, 본 개시물에서 설명하는 기법들의 여러 양태들을 수행하는 목적을 위해, 32 x 8 의 비-정사각형 변환 사이즈들은 16 x 16 의 변환 사이즈와 동등하게 취급될 수도 있으며, 비-정사각형 변환 사이즈 16 x 4 는 8 x 8 의 변환 사이즈와 동등하게 취급될 수도 있다. 즉, 32 x 8 변환에 대해 정의된 QP 는 16 x 16 변환에 대해 정의된 것과 동일하며, 반면 16 x 4 변환을 위한 QP 는 8 x 8 변환에 대해 정의된 것과 동일하다. 따라서, 8 x 8 정사각형 변환을 위한 QP 가 비-정사각형 16 x 4 변환에 대해 사용될 수 있다.

[0054] 게다가, 그 적용된 변환의 사이즈에 기초하여, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정하는 이들 여러 형태들을 지원하기 위해, 이 기법들은 이들 오프셋들을 화상 파라미터 세트 (picture parameter set; PPS), 시퀀스 파라미터 세트 (sequence parameter set; SPS), 적응 파라미터 세트 (adaptation parameter set; APS) 및 슬라이스 헤더 (이들 각각은 인코딩된 비디오 데이터를 나타내는 비트스트림에 규정될 수도 있다) 중 하나 이상으로 시그널링하는 여러 방법들을 위해 제공할 수도 있다. 예를 들어, (위에서 언급한 여러 오프셋들 중 하나 이상과 같은) 디블록킹 필터 제어 정보가 PPS 와 연관된 하나 이상의 슬라이스 헤더들에 규정되는지 여부를 나타내는 플래그가 PPS 로 시그널링될 수도 있다. 이 플래그는 PPS 에 deblocking_filter_control_present_flag 구문 엘리먼트로서 규정될 수도 있다.

[0055] 또 다른 예로서, 이들 여러 slice_tc_offset들 및 slice_beta_offset들을 슬라이스 헤더로 시그널링하는 대신, 이 기법들은 유사한 오프셋들이 PPS 또는 APS 에 규정될 수 있게 함으로써, 슬라이스 헤더들이 인코딩된 비디오 데이터를 나타내는 비트스트림으로 발생하는 레이트 보다 덜 빈번하게 PPS 들 또는 APS 들이 시그널링될 수도 있다는 점에서 슬라이스 오버헤드를 감소시킬 수도 있다. 일부 예들에서, 슬라이스 헤더로 시그널링되는 것으로 설명되는 여러 slice_tc_offset들 및 slice_beta_offset들은, 이들 오프셋들이 APS 와 연관된 슬라이스들에 적용하는 것을 고려하여, 그들의 각각의 이름들을 변경함이 없이 APS 로 시그널링될 수도 있다. APS 는 일반적으로 현재 규정된 구문 엘리먼트들, 따라서 이름 적응 파라미터 세트에 대한 적응들을 규정한다. 즉,

이전 슬라이스 헤더는 이들 오프셋들 중 하나 이상을 규정하고 있을 수도 있으며, (디코딩 순서에서) 후속 APS 는 이전 슬라이스 헤더에 의해 규정된 오프셋들 중 하나를 추가하거나, 제거하거나, 또는 아니면 변경하거나 또는 대체하는 적응을 규정할 수도 있다. 더욱이, (디코딩 순서에서) 후속 APS 는 이전에 적용된 오프셋들을 업데이트하는 적응들을 규정할 수도 있다. 적응들은 또 다른 슬라이스 헤더가 (현재 적용된 오프셋들의 모 두를 슬라이스 헤더에 규정된 오프셋들로 대체하는) 새로운 오프셋들에 제공하거나 또는 또 다른 APS 가 현재의 오프셋들을 수정할 때까지 적용될 수도 있다. 일부 예들에서, 잠재적으로 슬라이스 헤더로 시그널링되는 것으로 설명되는 여러 slice_tc_offset들 및 slice_beta_offset들은 PPS 로 시그널링될 수도 있다.

[0056] 동작 시, 비디오 인코더 (20) 는 위에서 설명한 방법으로 비디오 데이터의 블록에 변환을 적용하여, 변환 계수 들의 블록을 발생할 수도 있다. 이 변환은 아래에서 좀더 자세히 설명하는 바와 같이 비디오 데이터의 블록 을 참조하는 코딩 노드를 규정하는 코딩 유닛 (CU) 내에 포함될 수도 있는 HEVC 에 따라서 변환 유닛 (TU) 과 연관되거나 아니면 이에 의해 식별될 수도 있다. 즉, 일반적으로, HEVC 는 비디오 데이터의 블록들을 회귀 적으로 식별하는 프레임워크를 제공하며, 여기서, 최대 코딩 유닛 (LCU) 은 블록의 시작 사이즈를 나타낼 수도 있다. HEVC 는 그 후 (어느 경우에도 코딩 유닛 또는 CU 로서 지칭되는) 최소 코딩 유닛에 도달되거나 또는 어떤 추가적인 파티션이 요구되지 않을 때까지 이 LCU 를 회귀적으로 파티셔닝할 수도 있다. 종종, 어떤 추 가적인 파티션도 요구되지 않는다고 결정하는 것은 블록의 각각을 각각의 회귀적 레벨에서 코딩하고, 제공된 비 트레이트 및 도입된 왜곡의 관점에서 그들 여러 인코딩들을 비교하고, 그리고 주어진 인코딩 프로파일에 적합한 레이트-왜곡 최적화 (RDO) 를 제공하는 CU들을 선택하는 것을 수반하는 RDO 분석과 같은, 분석을 수반한다. CU 는 비디오 데이터의 블록을 식별하는 코딩 블록, 비디오 데이터의 블록에 적용되는 하나 이상의 변환들을 정의하는 TU, 및 모션 벡터들, 참조 화상 리스트들 및 인터- 또는 인트라-예측을 수행하는 것에 관련된 다른 데 이터와 같은 예측 데이터를 정의하는 예측 유닛 (PU) 을 포함할 수도 있다.

[0057] 위에서 언급한 바와 같이, TU 는 코딩 노드에 의해 식별되는 비디오 데이터의 블록에 적용되는 하나 이상의 변 환들을 식별할 수도 있다. TU 는 동일한 비디오 데이터의 블록에 각각 적용되는 2개 이상의 변환들을 정의 할 수도 있으며, 여기서, 비디오 데이터의 블록은 동일한 또는 상이한 사이즈의 변환들이 적용되는 별개의 변환 영역들로 파티셔닝될 수도 있다. 적용된 변환들의 사이즈는 비디오 코딩의 다수의 다른 양태들에 영향을 미 칠 수도 있다. 예를 들어, 비디오 인코더 (20) 는 적용된 변환의 사이즈에 기초하여 변하는 양자화 파라미 터를 적용하여, 변환 계수들의 블록을 양자화할 수도 있다.

[0058] 어느 경우라도, 양자화를 수행한 후, 비디오 인코더 (20) 는 변환 계수들의 양자화된 블록으로부터 비디오 데이 터의 블록을 재구성하고, 그리고 그 후 디블록킹 필터링을 수행할지 여부를 결정하여, 비디오 데이터의 재구성 된 블록을 필터링할 수도 있다. 비디오 인코더 (20) 는 비디오 코딩의 여러 양태들에 영향을 미치는 변환 사이즈의 또 다른 예를 제공하는 그 적용된 변환의 사이즈에 기초하여, 비디오 데이터의 재구성된 블록의 디블 록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적 어도 하나의 오프셋을 결정할 수도 있다. 디블록킹 필터링을 수행할지 여부를 결정하는 이 프로세스는 아래 에서 좀더 자세히 설명된다. 비디오 인코더 (20) 는 그 후 그 결정된 적어도 하나의 오프셋에 기초하여, 비 디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행할 수도 있다. 비디오 인코더 (20) 는 그 후 적어도 하나의 오프셋이 비디오 데이터의 블록의 인코딩된 버전을 포함하는 독립적으로 디코딩가능한 유닛 (예 컨대, 슬라이스) 의 헤더에 적어도 하나의 구문 엘리먼트로서 규정되는지 여부를 나타내는 플래그를 화상 파라 미터 세트 (PPS) 에 규정할 수도 있다. 일부의 경우, 이 플래그는 구문 테이블들에 "de-blocking_filter_control_present_flag" 구문 엘리먼트로서 표시될 수도 있다.

[0059] 일부의 경우, 비디오 인코더 (20) 는 플래그를 PPS 로 시그널링 또는 아니면 규정하지 않을 수도 있으며, 대신, 적어도 하나의 오프셋을 소위 적응 파라미터 세트 (APS) 로 제공할 수도 있다. 다른 경우, 비디오 인코더 (20) 는 플래그를 규정하지만, 플래그를, 적어도 하나의 오프셋이 비디오 데이터의 블록의 인코딩된 버전을 포 함하는 슬라이스의 헤더에 적어도 하나의 구문 엘리먼트로서 규정되어 있지 않다는 것을 나타내도록, 설정할 수 도 있다. 비디오 인코더 (20) 는 이 플래그를 규정한 후, 적어도 하나의 오프셋을 APS 에 규정할 수도 있으 며, 여기서, 적어도 하나의 오프셋이 슬라이스의 헤더에 규정되어 있지 않다고 나타내도록 설정하는 것은 적어 도 하나의 오프셋이 APS 에 규정되어 있다는 것을 나타낸다.

[0060] 이 플래그를 규정한 후, 비디오 인코더 (20) 는 비디오 디코더 (30) 가 적어도 하나의 오프셋 자체를 결정하도 록 요구받기보다는, 슬라이스 헤더 또는 APS 로부터 이 오프셋을 추출하거나 또는 추출할 수도 있기 때문에, 후 속 디코더 동작을 촉진하고 잠재적으로 복잡성을 감소시킬 수도 있다. 즉, 복잡한 동작들을 수반할 수도 있 는, 적어도 하나의 오프셋에 대한 값들을 암시해야 하는 대신, 비디오 디코더 (30) 는 이 값들을 추출할 수도

있다. 비록 HEVC 가 H.264/AVC 와 같은 이전 비디오 코딩 표준들 (따라서, 이름 "고효율 비디오 코딩") 과 비교하여 비디오 데이터를 코딩할 때에 증가된 효율을 촉진하려고 시도하지만, 일부 경우에, 복잡성을 감소시키는 것이 좀더 효율적인 디코더 동작을 증진시킬 수도 있으며, 어떤 상황들에서는, 코딩 효율을 증진시키기 위해 HEVC의 중대한 목표 (overarching goal) 를 무시할 수도 있다.

[0061] 더욱이, APS 에서 (본 개시물에서 규정하는 것과 등가인 것으로 해석될 수도 있지만, 실시간 또는 거의-실시간 시그널링을 암시하도록 해석되지 않아야 하는) 적어도 하나의 오프셋의 시그널링을 제공함으로써, 이 기법들은 APS 들이 일반적으로 비트스트림에서 슬라이스 헤더들 보다 덜 빈번하게 발생한다는 점에서 코딩 효율을 증진시킬 수도 있다. 따라서, 적어도 하나의 오프셋을 APS 로 단지 시그널링함으로써, 이 기법들은 적어도 하나의 오프셋을 시그널링하는 것 또는 아니면 규정하는 것과 연관된 오버헤드를 감소시킬 수도 있다.

[0062] 도 2 는 디블록킹 필터들의 적용을 촉진하기 위해 본 개시물에서 설명하는 기법들을 수행할 수도 있는 비디오 인코더 (20) 의 일 예를 예시하는 블록도이다. 비디오 인코더 (20) 는 예시의 목적을 위해, 그러나 다른 코딩 표준들 또는 방법들에 관한 본 개시물의 한정 없이, HEVC 코딩의 상황에서 설명될 것이다. 비디오 인코더 (20) 는 비디오 프레임들 내 CU들의 인트라-코딩 및 인터-코딩을 수행할 수도 있다. 인트라 코딩은 주어진 비디오 프레임 내 비디오 데이터에서 공간 리던던시를 감소시키거나 또는 제거하기 위해 공간 예측에 의존한다. 인터-코딩은 비디오 시퀀스의 현재의 프레임과 이전에 코딩된 프레임들 사이에 시간 리던던시를 감소시키거나 또는 제거하기 위해 시간 예측에 의존한다. 인트라-모드 (I-모드) 는 여러 공간-기반의 비디오 압축 모드들 중 임의의 모드를 지칭할 수도 있다. 단방향 예측 (P-모드) 또는 양방향 예측 (B-모드) 와 같은 인터-모드들은 여러 시간-기반의 비디오 압축 모드들 중 임의의 모드를 지칭할 수도 있다.

[0063] 도 2 에 나타낸 바와 같이, 비디오 인코더 (20) 는 인코딩될 비디오 프레임 내에서 현재의 비디오 블록을 수신한다. 도 2 의 예에서, 비디오 인코더 (20) 는 모션 보상 유닛 (44), 모션 추정 유닛 (42), 인트라-예측 유닛 (46), 디코딩된 화상 버퍼 (64), 합산기 (50), 변환 모듈 (52), 양자화 유닛 (54), 및 엔트로피 인코딩 유닛 (56) 을 포함한다. 도 2 에 예시된 변환 모듈 (52) 은 실제 변환 또는 변환의 조합들을 잔여 데이터의 블록에 적용하는 유닛이며, 또한 CU 의 변환 유닛 (TU) 으로서 지칭될 수도 있는 변환 계수들의 블록이 혼동되지 않아야 한다. 비디오 블록 재구성을 위해, 비디오 인코더 (20) 는 또한 역양자화 유닛 (58), 역변환 모듈 (60), 및 합산기 (62) 를 포함한다.

[0064] 인코딩 프로세스 동안, 비디오 인코더 (20) 는 코딩될 비디오 프레임 또는 슬라이스를 수신한다. 프레임 또는 슬라이스는 최대 코딩 유닛들 (LCUs) 과 같은 다수의 비디오 블록들로 분할될 수도 있다. 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 하나 이상의 참조 프레임들에서 하나 이상의 블록들에 대해 그 수신된 비디오 블록의 인트라-예측 코딩을 수행하여, 시간 압축을 제공할 수도 있다. 인트라-예측 유닛 (46) 은 코딩될 블록과 동일한 프레임 또는 슬라이스에서 하나 이상의 이웃하는 블록들에 대해서 그 수신된 비디오 블록의 인트라-예측 코딩을 수행하여, 공간 압축을 제공할 수도 있다.

[0065] 모드 선택 유닛 (40) 은 각각의 모드에 대한 레이트 및 에러 (즉, 왜곡) 결과들에 기초하여 코딩 모드들 중 하나, 즉, 인트라 또는 인터 코딩 모드를 선택하고, 최종 인트라- 또는 인터-예측된 블록 (예컨대, 예측 유닛 (PU)) 을 합산기 (50) 에 제공하여 잔여 블록 데이터를 발생하고, 그리고 합산기 (62) 에 제공하여 참조 프레임에 사용하기 위한 인코딩된 블록을 재구성할 수도 있다. 합산기 (62) 는 아래에서 더 자세히 설명하는 바와 같이 예측된 블록을 그 블록에 대한 역변환 모듈 (60) 로부터의 역 양자화된, 역변환된 데이터와 합산하여, 인코딩된 블록을 재구성한다. 일부 비디오 프레임들은 I-프레임들로서 지정될 수도 있으며, 여기서, I-프레임에서의 모든 블록들은 인트라-예측 모드에서 인코딩된다. 일부의 경우, 인트라-예측 유닛 (46) 은 예컨대, 모션 추정 유닛 (42) 에 의해 수행되는 모션 탐색이 블록의 충분한 예측을 야기하지 않을 때 P- 또는 B-프레임 내 블록의 인트라-예측 인코딩을 수행할 수도 있다.

[0066] 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 고도로 통합될 수도 있지만, 개념적인 목적들을 위해 별개로 예시된다. 모션 추정 (또는, 모션 탐색) 은 모션 벡터들을 발생하는 프로세스이며, 이 프로세스는 비디오 블록들에 대한 모션을 추정한다. 모션 벡터는, 예를 들어, 참조 프레임의 참조 샘플에 대한, 현재의 프레임에서의 예측 유닛의 변위를 나타낼 수도 있다. 모션 추정 유닛 (42) 은 예측 유닛을 디코딩된 화상 버퍼 (64) 에 저장된 참조 프레임의 참조 샘플들과 비교함으로써 인터-코딩된 프레임의 예측 유닛에 대한 모션 벡터를 계산한다. 참조 샘플은 코딩중인 PU 를 포함하는 CU 의 부분에 픽셀 차이의 관점에서 가깝게 매칭하는 것으로 발견되는 블록일 수도 있으며, 이것은 SAD (sum of absolute difference), SSD (sum of squared difference), 또는 다른 차이 메트릭들에 의해 결정될 수도 있다. 참조 샘플은 반드시 참조 프레임 또는 슬

라이스의 블록 (예컨대, 코딩 유닛) 경계에서는 아닌, 참조 프레임 또는 참조 슬라이스 내 어느 곳에서도 발생할 수도 있다. 일부 예들에서, 참조 샘플은 분수 픽셀 위치에서 발생할 수도 있다.

[0067] 모션 추정 유닛 (42) 은 계산된 모션 벡터를 엔트로피 인코딩 유닛 (56) 및 모션 보상 유닛 (44) 으로 전송한다. 특히, 모션 추정 유닛 (42) 은 위에서 설명한 병합 후보들을 이용하여, 병합 모드에 따라서 모션 벡터를 시그널링할 수도 있다 (즉, 모션 벡터를 복사할 이웃 블록의 인덱스를 시그널링할 수도 있다). 모션 벡터에 의해 식별된 참조 프레임의 부분은 참조 샘플로서 지칭될 수도 있다. 모션 보상 유닛 (44) 은 현재의 CU 의 예측 유닛에 대한 예측 값을, 예컨대, PU 에 대한 모션 벡터에 의해 식별되는 참조 샘플을 취출함으로써, 계산할 수도 있다.

[0068] 인트라-예측 유닛 (46) 은 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 수행되는 인터-예측에 대한 대안으로서, 수신된 블록을 인트라-예측할 수도 있다. 인트라-예측 유닛 (46) 은 이웃하는, 이전에 코딩된 블록들, 예컨대, 블록들에 대한 좌우, 상하 인코딩 순서를 가정할 때, 현재의 블록의 상부, 상부 및 우측, 상부 및 좌측, 또는 좌측 블록들에 대해, 수신되는 블록을 예측할 수도 있다. 인트라-예측 유닛 (46) 은 다양한 상이한 인트라-예측 모드들로 구성될 수도 있다. 예를 들어, 인트라-예측 유닛 (46) 은 인코딩되는 CU 의 사이즈에 기초하여, 어떤 개수의 방향 예측 모드들, 예컨대, 35개의 방향 예측 모드들로 구성될 수도 있다.

[0069] 인트라-예측 유닛 (46) 은 예를 들어, 여러 인트라-예측 모드들에 대한 여러 값들을 계산하여 최저 여러 값을 산출하는 모드를 선택함으로써, 인트라-예측 모드를 선택할 수도 있다. 방향 예측 모드들은 공간적으로 이웃하는 픽셀들의 값들을 결합하여 그 결합된 값들을 PU 내 하나 이상의 픽셀 위치들에 적용하는 기능들을 포함할 수도 있다. 일단 PU 내 모든 픽셀 위치들에 대한 값들이 계산되었으면, 인트라-예측 유닛 (46) 은 PU 와 인코딩되는 수신된 블록 사이의 픽셀 차이들에 기초하여, 예측 모드에 대한 여러 값을 계산할 수도 있다. 인트라-예측 유닛 (46) 은 허용가능한 여러 값을 산출하는 인트라-예측 모드가 발견될 때까지 인트라-예측 모드들을 테스트하는 것을 계속할 수도 있다. 인트라-예측 유닛 (46) 은 그 후 PU 를 합산기 (50) 로 전송할 수도 있다.

[0070] 비디오 인코더 (20) 는 코딩중인 원래 비디오 블록으로부터 모션 보상 유닛 (44) 또는 인트라-예측 유닛 (46) 에 의해 계산된 예측 데이터를 감산함으로써, 잔여 블록을 형성한다. 합산기 (50) 는 이 감산 동작을 수행하는 구성요소 또는 구성요소들을 나타낸다. 잔여 블록은 픽셀 차이 값들의 2차원 매트릭스에 대응할 수도 있으며, 여기서, 잔여 블록에서의 값들의 수는 잔여 블록에 대응하는 PU 내 픽셀들의 개수와 동일하다. 잔여 블록에서의 값들은 PU 에 그리고 코딩되는 원래 블록에서 동일 장소에 배치된 픽셀들의 값들 사이의 차이들, 즉, 에러에 대응할 수도 있다. 그 차이들은 코딩되는 블록의 유형에 따라서 크로마 또는 루마 차이들일 수도 있다.

[0071] 변환 모듈 (52) 은 잔여 블록으로부터의 하나 이상의 변환 유닛들 (TUs) 을 포함할 수도 있다. 변환 모듈 (52) 은 (종종, 8 x 8, 16 x 16, 8 x 16, 16 x 8 등과 같은 상이한 사이즈들의) 복수의 변환들 가운데서 변환을 선택한다. 이 변환은 블록 사이즈, 코딩 모드, 또는 기타 등등과 같은 하나 이상의 코딩 특성들에 기초하여 선택될 수도 있다. 변환 모듈 (52) 은 그 후 그 선택된 변환을 TU 에 적용하여, 변환 계수들의 2차원 어레이를 포함하는 비디오 블록을 발생한다. 게다가, 변환 모듈 (52) 은 그 선택된 변환 파티션을 인코딩된 비디오 비트스트림으로 시그널링할 수도 있다.

[0072] 변환 모듈 (52) 은 최종 변환 계수들을 양자화 유닛 (54) 으로 전송할 수도 있다. 양자화 유닛 (54) 은 그 후 변환 계수들을 양자화할 수도 있다. 엔트로피 인코딩 유닛 (56) 은 그 후 양자화된 변환 계수들의 스캐닝을 매트릭스로 스캐닝 모드에 따라서 수행할 수도 있다. 본 개시물은 스캐닝을 수행하는 것으로 엔트로피 인코딩 유닛 (56) 을 기술한다. 그러나, 다른 예들에서, 양자화 유닛 (54) 과 같은, 다른 프로세싱 유닛들이 스캐닝을 수행할 수 있는 것으로 이해되어야 한다.

[0073] 일단 변환 계수들이 1차원 어레이로 스캐닝되면, 엔트로피 인코딩 유닛 (56) 은 CAVLC, CABAC, 구문-기반의 컨텍스트-적응 2진 산술 코딩 (SBAC), 또는 또 다른 엔트로피 코딩 방법론과 같은 엔트로피 코딩을 그 계수들에 적용할 수도 있다.

[0074] CAVLC 을 수행하기 위해, 엔트로피 인코딩 유닛 (56) 은 송신되는 심볼에 대해 가변 길이 코드를 선택할 수도 있다. VLC 에서의 코드워드들은 상대적으로 더 짧은 코드들이 더 가능성 있는 심볼들에 대응하는 반면, 더 긴 코드들이 덜 가능성 있는 심볼들에 대응하도록, 구성될 수도 있다. 이러한 방법으로, VLC 의 사용은 예를 들어, 송신되는 각각의 심볼에 대해 동일-길이 코드워드들을 사용하는 것을 넘어서는 비트 절감을 달성할 수

도 있다.

- [0075] CABAC 을 수행하기 위해, 엔트로피 인코딩 유닛 (56) 은 송신되는 심볼들을 인코딩하기 위해 어떤 컨텍스트에 적용할 컨텍스트 모델을 선택할 수도 있다. 컨텍스트는 예를 들어, 이웃하는 값들이 논-제로인지 여부에 관련될 수도 있다. 엔트로피 인코딩 유닛 (56) 은 또한 그 선택된 변환을 나타내는 신호와 같은 구문 엘리먼트들을 엔트로피 인코딩할 수도 있다. 엔트로피 인코딩 유닛 (56) 에 의한 엔트로피 코딩 이후, 최종 인코딩된 비디오는 비디오 디코더 (30) 와 같은 또 다른 디바이스로 전송되거나, 또는 추후 송신 또는 취출을 위해 아카이브될 수도 있다.
- [0076] 일부의 경우, 엔트로피 인코딩 유닛 (56) 또는 비디오 인코더 (20) 의 또 다른 유닛은 엔트로피 코딩에 더해서, 다른 코딩 기능들을 수행하도록 구성될 수도 있다. 예를 들어, 엔트로피 인코딩 유닛 (56) 은 CU들 및 PU들에 대한 코딩된 블록 패턴 (CBP) 값들을 결정하도록 구성될 수도 있다. 또한, 일부의 경우, 엔트로피 인코딩 유닛 (56) 은 계수들의 런 길이 코딩 (run length coding) 을 수행할 수도 있다.
- [0077] 역양자화 유닛 (58) 및 역변환 모듈 (60) 은 역양자화 및 역변환을 각각 적용하여, 예컨대, 참조 블록으로 추후 사용을 위해, 픽셀 도메인에서 잔여 블록을 재구성한다. 모션 보상 유닛 (44) 은 잔여 블록을 디코딩된 화상 버퍼 (64) 의 프레임들 중 하나의 예측 블록에 가산함으로써, 참조 블록을 계산할 수도 있다. 모션 보상 유닛 (44) 은 또한 하나 이상의 내삽 필터들을 그 재구성된 잔여 블록에 적용하여, 모션 추정에 사용하기 위한 서브-정수 픽셀 값들을 계산할 수도 있다. 합산기 (62) 는 재구성된 잔여 블록을 모션 보상 유닛 (44) 에 의해 발생된 모션 보상된 예측 블록에 가산하여, 디코딩된 화상 버퍼 (64) 에의 저장을 위해 재구성된 비디오 블록을 발생한다. 비디오 인코더 (20) 는 또한 또는 다른 디블록킹 필터를 적용하여 본 개시물에서 설명하는 기법들의 여러 양태들에 따라서 디블록킹 필터링을 수행할 수도 있는 디블록킹 유닛 (63) 을 포함할 수도 있다. 이 필터는 블로키니스 아티팩트들을 제거하기 위해 디코딩된 블록들을 필터링하는데 적용될 수도 있다. 재구성된 비디오 블록은 후속 비디오 프레임에서 블록을 인터-코딩하기 위해 참조 블록으로서 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 사용될 수도 있다.
- [0078] 본 개시물에서 설명하는 기법들에 따르면, 변환 모듈 (52) 은 위에서 설명한 방법으로 비디오 데이터의 블록에 변환을 적용하여 변환 계수들의 블록을 발생할 수도 있다. 일부의 경우, 변환 모듈 (52) 은 잔여 비디오 데이터의 동일한 블록으로부터 다수의 상이한 TU들을 형성할 수도 있으며, 여기서, 상이한 TU들의 각각은 비디오 데이터의 동일한 잔여 블록에 적용되는 변환들의 상이한 조합을 식별한다.
- [0079] 예를 들어, 변환 모듈 (52) 은 사이즈 16×16 의 비디오 데이터의 잔여 블록을 수신할 수도 있다. 변환 모듈 (52) 은 16×16 변환이 16×16 잔여 비디오 데이터 블록에 적용되었다는 것을 나타내는 제 1 TU 를 발생하고 16×16 변환을 16×16 비디오 데이터 블록에 적용하여, 비디오 데이터의 16×16 블록을 변환 계수들의 16×16 블록으로 변환할 수도 있다. 변환 모듈 (52) 은 4개의 8×8 변환들이 16×16 잔여 비디오 데이터 블록에 적용되었다는 것을 나타내는 제 2 TU 를 발생하고 8×8 변환들을 16×16 비디오 데이터 블록에 적용하여, 비디오 데이터의 16×16 블록을 변환 계수들의 16×16 블록으로 변환할 수도 있다. 변환 모듈 (52) 은 2개의 16×8 변환들이 16×16 잔여 비디오 데이터 블록에 적용되었다는 것을 나타내는 제 3 TU 를 발생하고 16×8 변환들을 16×16 비디오 데이터 블록에 적용하여, 비디오 데이터의 16×16 블록을 변환 계수들의 16×16 블록으로 변환할 수도 있다. 변환 모듈 (52) 은 8×16 변환이 16×16 잔여 비디오 데이터 블록에 적용되었다는 것을 나타내는 제 4 TU 를 발생하고 8×16 변환을 16×16 비디오 데이터 블록에 적용하여, 비디오 데이터의 16×16 블록을 변환 계수들의 16×16 블록으로 변환할 수도 있다.
- [0080] 비디오 인코더 (20) 는 그 후 위에서 설명한 방법으로 CU들의 각각을 계속해서 인코딩하여, 변환 계수들의 16×16 블록의 4개의 상이한 버전들을 양자화할 수도 있다. 비디오 인코더 (20) 는 다음으로, 위에서 설명한 방법으로 TU들의 각각에 대응하는 변환 계수들의 양자화된 블록으로부터 비디오 데이터의 블록의 각각을 재구성할 수도 있다. 모드 선택 유닛 (40) 은 그 후 비디오 데이터의 재구성된 블록들의 이들 상이한 버전들의 각각을 평가하여, 어느 것이, 일 예로서, 비디오 데이터를 인코딩하기 위한 현재의 프로파일에 적합한 레이트-왜곡 최적화를 제공하는지를 식별할 수도 있다. 모드 선택 유닛 (40) 은 비디오 데이터의 재구성된 블록의 버전들의 선택된 하나를 디코딩된 화상 버퍼 (64) 에 저장할 수도 있다.
- [0081] (TU 의 상이한 버전들 각각과 연관된) CU 의 상이한 버전들 각각을 인코딩할 때에, 양자화 유닛 (54) 은 적용된 변환의 사이즈에 기초하여 변하는 양자화 파라미터를 적용하여 변환 계수들의 블록을 양자화할 수도 있다. 양자화를 수행한 후, 비디오 인코더 (20) 는 변환 계수들의 양자화된 블록으로부터 비디오 데이터의 블록을 재구성하고, 그리고 그 후 디블록킹 필터링을 수행할지 여부를 결정하여, 비디오 데이터의 재구성된 블록을 필터

링할 수도 있다. 좀더 구체적으로는, 디블록킹 유닛 (63) 은 그 적용된 변환의 사이즈에 기초하여, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정할 수도 있다.

[0082] 예를 들어, 디블록킹 유닛 (63) 은 디블록킹 필터링을 수행할지 여부를 결정하여 비디오 데이터의 재구성된 블록을 필터링하기 위해 다음 의사-코드를 구현할 수도 있다:

[0083] - transform_type = N x N (= 4 x 4 또는 8 x 8 또는 16 x 16 또는 32 x 32)

[0084] • INTRA_TC_OFFSET = slice_tc_offset_NxN_intra

[0085] • INTER_TC_OFFSET = slice_tc_offset_NxN_inter

[0086] • INTRA_BETA_OFFSET = slice_beta_offset_NxN_intra

[0087] • INTER_BETA_OFFSET = slice_beta_offset_NxN_inter

[0088] - TcOffset = (Bs > 2) ? DEFAULT_INTRA_TC_OFFSET(=2) + INTRA_TC_OFFSET : INTER_TC_OFFSET

[0089] - BetaOffset = (Bs > 2) ? INTRA_BETA_OFFSET : INTER_BETA_OFFSET

[0090] - t_c : $Q = \text{Clip3}(0, \text{MAX_QP}+4, \text{QP} + \text{TcOffset})$ (루마 및 크로마 디블록킹)

[0091] - β : $Q = \text{Clip3}(0, \text{MAX_QP}+4, \text{QP} + \text{BetaOffset})$ (루마 디블록킹 단독)

[0092] 이 의사-코드는 슬라이스-레벨 구문 유닛들, 즉, slice_tc_offset_NxN_intra, slice_tc_offset_NxN_inter, slice_beta_offset_NxN_intra 및 slice_beta_offset_NxN_inter 가 N x N (예컨대, 4 x 4, 8 x 8, 16 x 16 및 32 x 32) 와 동일한 것으로 리스트된 변환 유형들 중 임의의 변환 유형에 대해, 그리고 부가적으로 인트라- 또는 인터-코딩된 블록들에 대해, 정의될 수도 있다는 것을 나타낸다.

[0093] 상기 의사-코드에 따르면, 디블록킹 유닛 (63) 은 변환 유형 (이 예에서는, CU 의 상이한 버전들 각각에 적용되는 변환의 사이즈를 지칭함) 을 결정하고, INTRA_TC_OFFSET, INTER_TC_OFFSET, INTRA_BETA_OFFSET 및 INTER_BETA_OFFSET 를 slice_tc_offset_NxN_intra, slice_tc_offset_NxN_inter, slice_beta_offset_NxN_intra 및 slice_beta_offset_NxN_inter 중 대응하는 하나로 설정할 수도 있다. 이들 오프셋들을 설정한 후, 디블록킹 유닛 (63) 은 그 후 그 결정된 경계 강도 (Bs) 가 2 보다 클 때 TcOffset 를 DEFAULT_INTRA_TC_OFFSET (2 와 동일한) 플러스 INTRA_TC_OFFSET 와 동일한 것으로 결정한다. 디블록킹 유닛 (63) 은 디블록킹되는 경계에서 경계 강도를 픽셀 값들의 함수로서 결정할 수도 있다. 그 결정된 경계 강도가 2 보다 크지 않을 때, 디블록킹 유닛 (63) 은 TcOffset 를 INTER_TC_OFFSET 와 동일한 것으로 결정할 수도 있다. 디블록킹 유닛 (63) 은 또한, 상기 의사-코드에 따르면, 경계 강도가 2 보다 크게 결정될 때 BetaOffset 를 INTRA_BETA_OFFSET 와 동일한 것으로 결정할 수도 있다. 2 보다 크지 않을 때, 디블록킹 필터 (63) 는 BetaOffset 를 INTER_BETA_OFFSET 과 동일한 것으로 결정할 수도 있다. 디블록킹 유닛 (63) 은 그 후 t_c 를 Q 와 동일한 것으로 결정할 수도 있으며, Q 자체는 양자화 파라미터 (QP) 의 클리핑된 버전 플러스 TcOffset 와 동일하며, 여기서, 가능한 값들의 범위는 제로로부터 MAX_QP (일반적으로, 52 로 설정됨) 플러스 4 까지 이른다. 디블록킹 유닛 (63) 은 그 후 β 를 Q 과 동일한 것으로 결정할 수도 있으며, Q 자체는 양자화 파라미터 (QP) 의 클리핑된 버전 플러스 BetaOffset 와 동일하며, 여기서, 가능한 값들의 범위는 제로로부터 MAX_QP (또한 일반적으로 52 로 설정됨) 플러스 4 까지 이른다. 디블록킹 유닛 (63) 은 루마 및 크로마 디블록킹에 대해 t_c 를, 그리고 루마 디블록킹에 대해 β 를 이용한다.

[0094] 이의 대안으로, 디블록킹 유닛 (63) 은 슬라이스-레벨 구문 유닛들, 즉, slice_tc_offset_intra, slice_tc_offset_inter, slice_beta_offset_intra, slice_beta_offset_inter 에 기초하여, 수학적 관계를 적용하여 QP 값에 대한 오프셋들을 결정하는 것을 예시하는 다음 의사-코드를 구현할 수도 있다:

[0095] transform_type = N x N (= 4 x 4 또는 8 x 8 또는 16 x 16 또는 32 x 32)

[0096] a. INTRA_TC_OFFSET = slice_tc_offset_intra + tc_offset_NxN_intra_delta

[0097] b. INTER_TC_OFFSET = slice_tc_offset_inter + tc_offset_NxN_inter_delta

[0098] c. INTRA_BETA_OFFSET = slice_beta_offset_intra + beta_offset_NxN_intra_delta

- [0099] d. $INTER_BETA_OFFSET = slice_beta_offset_inter + beta_offset_NxN_inter_delta$
- [0100] $Tc_offset_NxN_intra_delta = tc_offset_intra_delta * factor_NxN$
- [0101] $Tc_offset_NxN_inter_delta = tc_offset_inter_delta * factor_NxN$
- [0102] $Beta_offset_NxN_intra_delta = beta_offset_intra_delta * factor_NxN$
- [0103] $Beta_offset_NxN_inter_delta = beta_offset_inter_delta * factor_NxN$
- [0104] 계산에 사용되는 (그러나, 시그널링될 수도 없는) 상수 값들의 예들:
- [0105] $Factor_4x4 = 3$
- [0106] $Factor_8x8 = 2$
- [0107] $Factor_16x16 = 1$
- [0108] $Factor_32x32 = 0$
- [0109] $Tc_offset_intra_delta = -1$
- [0110] $Tc_offset_inter_delta = -1$
- [0111] $Beta_offset_intra_delta = -1$
- [0112] $Beta_offset_inter_delta = -1$
- [0113] $TcOffset = (Bs > 2) ? DEFAULT_INTRA_TC_OFFSET(=2) + INTRA_TC_OFFSET : INTER_TC_OFFSET$
- [0114] $BetaOffset = (Bs > 2) ? INTRA_BETA_OFFSET : INTER_BETA_OFFSET$
- [0115] $t_c: Q = Clip3(0, MAX_QP+4, QP + TcOffset)$ (루마 및 크로마 디블록킹)
- [0116] $\beta: Q = Clip3(0, MAX_QP+4, QP + BetaOffset)$ (루마 디블록킹 단독)
- [0117] 선행하는 의사-코드에 따르면, 디블록킹 유닛 (63)은 $TcOffset$ 를 BS의 함수로서 결정할 수도 있으며, 여기서, BS가 2보다 더 크면, 비디오 코딩 디바이스는 $TcOffset$ 의 값이 $DEFAULT_INTRA_TC_OFFSET$ (보통 2와 동일함) 플러스 그 결정된 $INTRA_TC_OFFSET$ 과 동일하다고 결정할 수도 있으며, 이 자체는 정의된 $slice_tc_offset_NxN_intra$ 중 임의의 하나 (이 예에서는, 블록 Y의 사이즈에 기초하여 선택됨)와 동일할 수도 있다. 그러나, BS가 2보다 작거나 또는 동일하면, 디블록킹 유닛 (63)은 $TcOffset$ 의 값이 $INTER_TC_OFFSET$ 과 동일하다고 결정하며, 이 자체는 정의된 $slice_tc_offset_NxN_inter$ 중 임의의 하나 (이 예에서는, 블록 Y의 사이즈에 기초하여 선택됨)와 동일할 수도 있다.
- [0118] 게다가, 디블록킹 유닛 (63)은 $BetaOffset$ 에 대한 값을 또한 BS의 함수로서 결정할 수도 있으며, BS가 2보다 더 크면, 비디오 코딩 디바이스는 $BetaOffset$ 의 값이 $INTRA_BETA_OFFSET$ 과 동일하다고 결정할 수도 있으며, 이 자체는 정의된 $slice_beta_offset_NxN_intra$ 중 임의의 하나 (이 예에서는, 블록 Y의 사이즈에 기초하여 선택됨)와 동일할 수도 있다. 이의 대안으로, BS가 2보다 작거나 또는 동일하면, 비디오 코딩 디바이스는 $BetaOffset$ 의 값이 $INTER_BETA_OFFSET$ 과 동일하다고 결정할 수도 있으며, 이 자체는 정의된 $slice_beta_offset_NxN_inter$ 값들 중 임의의 하나 (이 예에서는, 블록 Y의 사이즈에 기초하여 선택됨)와 동일할 수도 있다.
- [0119] 디블록킹 유닛 (63)은 유사한 방법으로 그 결정된 $TcOffset$ 및 $BetaOffset$ 양쪽을 이용하여, 편의상 수식 포맷으로 아래에 재현되는, 위에서 나타낸 의사-코드 또는 수식들에 따라서 대응하는 t_c 임계치 및 β 임계치를 탐색하기 위한 각각의 Q 값들을 유도할 수도 있다.
- [0120] $t_c: Q = Clip3(0, MAX_QP+4, QP + TcOffset)$ (4)
- [0121] $\beta: Q = Clip3(0, MAX_QP+4, QP + BetaOffset)$ (5)
- [0122] 수식 (4)에 따르면, 디블록킹 유닛 (63)은 제로로부터 MAX_QP+4 (여기서, MAX_QP 는 대개 51과 동일하다)까지의 범위에서 클리핑하는 것을 조건으로 하여, Q 값을, QP 플러스 그 결정된 $TcOffset$ 으로 설정함으로써, 상기 테이블 1에서 t_c 임계치를 탐색하기 위한 Q 값을 결정한다. 이와 유사하게, 디블록킹 유닛 (63)은 제

로부터 MAX_QP+4 (여기서, MAX_QP 는 대개 51 과 동일하다) 까지의 범위에서 클리핑하는 것을 조건으로 하여, Q 값을, QP 플러스 그 결정된 $BetaOffset$ 로 설정함으로써, 상기 테이블 1 에서 β 임계치를 탐색하기 위한 Q 값을 결정할 수도 있다. 디블록킹 유닛 (63) 은 그 후 이들 임계치들을 이용하여, 디블록킹 필터링이 비디오 데이터의 현재의 블록의 경계들에 대해 수행되어야 하는지 여부를 결정할 수도 있다. 이 결정에 기초하여, 디블록킹 유닛 (63) 은 디블록킹 필터링을 수행한다. t_c 및 β 임계치들에 대한 Q 파라미터를 결정하는데 사용되는 오프셋들의 더 미세한 정의를 가능하게 함으로써, 이 기법들은 자각할 수 있는 비디오 품질보다 더 나은 제어를 증진시키거나 및/또는 코딩 이득을 증진시킬 수 있도록, 디블록킹을 제어하는 인코더들의 능력을 향상시킬 수도 있다.

[0123] 일부의 경우, 디블록킹 유닛 (63) 은 $INTRA_BETA_OFFSET$ 및 $INTER_BETA_OFFSET$ 양쪽을 계산하거나 또는 결정하지 않고, 또한 $INTRA_TC_OFFSET$ 및 $INTER_TC_OFFSET$ 양쪽을 결정하는 한편 단지 단일 $BETA_OFFSET$ 를 계산하거나 또는 결정할 수도 있다. 다음 의사 코드는 이 단일 $BETA_OFFSET$ 경우에, 슬라이스-레벨 구문 유닛들, $slice_tc_offset_intra$, $slice_tc_offset_intra_delta$, $slice_tc_offset_inter$, $slice_tc_offset_inter_delta$, 및 $slice_beta_offset$ 에 기초하여, 수학적 관계를 적용하여 QP 값에 대한 오프셋들을 결정하는 것을 예시한다.

[0124] - $transform_type = N \times N$ (= 4 x 4 또는 8 x 8 또는 16 x 16 또는 32 x 32)

[0125] e. $INTRA_TC_OFFSET = slice_tc_offset_intra + tc_offset_NxN_intra_delta$

[0126] f. $INTER_TC_OFFSET = slice_tc_offset_inter + tc_offset_NxN_inter_delta$

[0127] g. $BETA_OFFSET = slice_beta_offset$

[0128] - $Tc_offset_NxN_intra_delta = slice_tc_offset_intra_delta * factor_NxN$

[0129] - $Tc_offset_NxN_inter_delta = slice_tc_offset_inter_delta * factor_NxN$

[0130] 이 계산에 사용되는 상수 값들의 예들은 다음과 같다 (여기서, 이들 상수들은 디코더로 시그널링되거나 또는 아니면 규정되지 않을 수도 있다):

[0131] $Factor_4x4 = 3$

[0132] $Factor_8x8 = 2$

[0133] $Factor_16x16 = 1$

[0134] $Factor_32x32 = 0$

[0135] 단일 $BETA_OFFSET$ 및 $INTRA_TC_OFFSET$ 및 $INTER_TC_OFFSET$ 양쪽의 경우, 디블록킹 유닛 (63) 은 $TcOffset$ 및 $BetaOffset$ 를 다음 의사-코드에 대해 제시된 방법과는 약간 상이한 방법으로 계산할 수도 있다:

[0136] - $TcOffset = (Bs > 2) ? INTRA_TC_OFFSET : INTER_TC_OFFSET$

[0137] - $BetaOffset = BETA_OFFSET$

[0138] 이 의사-코드에 따르면, 디블록킹 유닛 (63) 은 $TcOffset$ 를 Bs 의 함수로서 결정하며, 여기서, Bs 가 2 보다 더 크면, 디블록킹 유닛 (63) 은 $TcOffset$ 의 값이 $INTRA_TC_OFFSET$ 과 동일하다고 결정할 수도 있다. 그러나, Bs 가 2 보다 작거나 또는 동일하면, 디블록킹 유닛 (63) 은 $TcOffset$ 의 값이 $INTER_TC_OFFSET$ 과 동일하다고 결정한다. 디블록킹 유닛 (63) 은 유사한 방법으로 그 결정된 $TcOffset$ 및 $BetaOffset$ 양쪽을 이용하여, 수식들 (4) 및 (5) 로서 위에서 나타낸 의사-코드 또는 수식들에 따라서 대응하는 t_c 임계치 및 β 임계치를 탐색하기 위해 각각의 Q 값들을 유도한다.

[0139] 대응하는 디코더, 예컨대, 도 1 의 예에 나타낸 비디오 디코더 (30) 를 위해서, 비디오 인코더 (20) 는 여러 오프셋들을 SPS, PPS, APS 및 슬라이스 헤더와 같은, 비트스트림의 상이한 부분들로 시그널링할 수도 있다. 다음은 이들 오프셋들의 시그널링을 추가적인 구문 엘리먼트들로서 지원하기 위해서, HEVC 의 현재의 작업 초안 (working draft) 에 요구될 수도 있는 HEVC 구문 변경들을 기술한다. 다수의 예들이 아래에 제공되지만, 이들 구문 엘리먼트들을 시그널링하는 임의의 방법이 본 기법들의 여러 양태들을 달성하는데 가능할 수도 있다. 게다가, 이들 추가적인 구문 엘리먼트들을 시그널링하는 여러 방법들이 주어진 컨텍스트에 따라서 사용될 수도 있으며, 컨텍스트는 또한 비디오 디코더가 어떻게 비트스트림을 적합하게 파싱할지를 결정할 수 있도록, 시

그널링될 수도 있다. 더욱이, 일부 경우에, 인트라-코딩된 및 인터-코딩된 블록들 양쪽에 대한 TcOffset 및 BetaOffset 구문 엘리먼트들이 명시적으로 시그널링될 수도 있지만, 다른 경우에, 하나 이상의 변환 유형들에 대해 단일 TcOffset 및 단일 BetaOffset 가 시그널링될 수도 있으며, 여기서, 비디오 디코더는 시그널링된 TcOffset 및 BetaOffset 으로부터 이들 오프셋들의 인트라-코딩된 및 인터-코딩된 버전들을 유도한다.

[0140] 어느 경우라도, 모드 선택 유닛 (40) 은 디블록킹된 후 레이트 및 왜곡의 관점에서 주어진 인코딩 프로파일을 증진시킬 수도 있는, 비디오 데이터의 블록의 재구성된 버전들의 대응하는 하나를 선택할 수도 있다. 모드 선택 유닛 (40) 은 인코딩된 비디오 데이터를 나타내는 비트스트림을 조직화할 수도 있는, 디블록킹을 수행하는 데 사용한 대응하는 오프셋들을 엔트로피 인코딩 유닛 (56) 으로 시그널링할 수도 있다. 엔트로피 인코딩 유닛 (56) 은 다음 테이블 2 에 예시된 바와 같이 de-blocking_filter_control_present_flag 를 화상 파라미터 세트 (PPS) 구문에 추가할 수도 있다. de-blocking_filter_control_present_flag 가 1 과 동일하면, 이 플래그는 디블록킹 필터의 특성들을 제어하는 구문 엘리먼트들의 세트가 슬라이스 헤더에 존재한다는 것을 규정한다. 이 플래그가 0 과 동일하면, 이 플래그는 디블록킹 필터의 특성들을 제어하는 구문 엘리먼트들의 세트가 슬라이스 헤더에 존재하지 않는다는 것을 규정하며, 그들의 추론된 값들이 비디오 디코더에 의해 사용되지 않는다. 플래그에 대한 파싱 프로세스를 규정하는 구문적 설명은 단일 비트와 같이 적합하게 선택되어야 한다.

[0141] 테이블 2

pic_parameter_set_rbsp() {
pic_parameter_set_id
seq_parameter_set_id
...
de-blocking_filter_control_present_flag
...
rbsp_trailing_bits()
}

[0142]

[0143] 다음 테이블들은 slice_tc_offset 및 slice_beta_offset 값들을 시그널링하는 슬라이스와 같은, 독립적으로 디코딩가능한 유닛의 헤더의 구문에 대해 상이한 경우들을 나타낸다 (여기서, 이런 구문은 "슬라이스 헤더 구문" 으로서 지칭될 수도 있다). 이의 대안으로, 오프셋들은 어떤 다른 곳으로, 예컨대 적응 파라미터 세트 (APS) 또는 화상 파라미터 세트 (PPS) 로 시그널링될 수 있다. 다음 테이블 3 은 각각의 변환 유형 (4 x 4, 8 x 8, 16 x 16, 32 x 32) 에 대해, 그리고 인트라 및 인터 모드들에 대해, 모든 tc 및 베타 오프셋들을 포함하는 슬라이스 헤더 구문을 나타낸다. 이의 대안으로, 테이블 4 및 테이블 5 은 더 적은 오프셋 엘리먼트들을 가진 슬라이스 헤더 구문 예들을 예시한다.

[0144] 테이블 3

slice_header() {
...
if(de-blocking_filter_control_present_flag) {
disable_de-blocking_filter_idc
if(disable_de-blocking_filter_idc != 1) {
slice_tc_offset_4x4_intra
slice_tc_offset_8x8_intra
slice_tc_offset_16x16_intra
slice_tc_offset_32x32_intra
slice_beta_offset_4x4_intra
slice_beta_offset_8x8_intra
slice_beta_offset_16x16_intra
slice_beta_offset_32x32_intra
if (slice_type != I) {
slice_tc_offset_4x4_inter
slice_tc_offset_8x8_inter
slice_tc_offset_16x16_inter
slice_tc_offset_32x32_inter
slice_beta_offset_4x4_inter
slice_beta_offset_8x8_inter
slice_beta_offset_16x16_inter
slice_beta_offset_32x32_inter
}
}
}

[0145]

[0146] 다음 테이블 4 에서, 엔트로피 인코딩 유닛 (56) 은 인트라-코딩된 블록들에 적용되는 변환 유형들 모두에 대한 t_c 오프셋들을 정의하며, 단일 표준 β 오프셋은 인트라-코딩된 블록들에 적용되는 모든 변환 유형들에 대해 정의된다. 게다가, 슬라이스 유형이 I-슬라이스가 아니면 (P 또는 B 슬라이스임을 나타냄), 엔트로피 인코딩 유닛 (56) 은 인터-코딩된 블록들에 적용되는 변환 유형들 모두에 대한 t_c 오프셋들 및 인터-코딩된 블록들에 적용되는 모든 변환 유형들에 대한 단일 표준 β 오프셋을 정의한다.

[0147] 테이블 4

slice_header() {
...
if(de-blocking_filter_control_present_flag) {
disable_de-blocking_filter_idc
if(disable_de-blocking_filter_idc != 1) {
slice_tc_offset_4x4_intra
slice_tc_offset_8x8_intra
slice_tc_offset_16x16_intra
slice_tc_offset_32x32_intra
slice_beta_offset_intra
if (slice_type != I) {
slice_tc_offset_4x4_inter
slice_tc_offset_8x8_inter
slice_tc_offset_16x16_inter
slice_tc_offset_32x32_inter
slice_beta_offset_inter
}
}
}

[0148]

[0149] 다음 테이블 5 는 구문 시그널링을 정의하며, 이에 의해, 엔트로피 인코딩 유닛 (56) 은 인트라-코딩된 및 인터-코딩된 블록들 양쪽에 적용되는 모든 변환 유형들에 대한 일반적인 t_c 오프셋들 및 인트라-코딩된 및 인터-코딩된 블록들 양쪽에 적용되는 모든 변환 유형들에 대한 단일 표준 β 오프셋을 시그널링한다.

[0150] 테이블 5

slice_header() {
...
if(de-blocking_filter_control_present_flag) {
disable_de-blocking_filter_idc
if(disable_de-blocking_filter_idc != 1) {
slice_tc_offset_4x4
slice_tc_offset_8x8
slice_tc_offset_16x16
slice_tc_offset_32x32
slice_beta_offset
}
}
}

[0151]

[0152] 다음 슬라이스 구문 예들에서, 인코딩 유닛 (56) 은 다음 2개의 새로운 디블록킹 플래그들을 도입한다:

[0153] • 0 과 동일할 때, 동일한 오프셋이 인트라 및 인터 모드들에 대해 사용된다는 것을 시그널링하는, de-blocking_offset_intra_inter_flag. 1 과 동일할 때, 이 플래그는 오프셋들이 인트라 및 인터 코딩 모드들에 대해 시그널링된다는 것을 나타낸다.

[0154] • 0 과 동일할 때, 오프셋들이 변환 사이즈 독립적이고 그리고 1 과 동일할 때, 오프셋들이 변환 사이즈 의존적이라는 것을 시그널링하는, de-blocking_offset_transform_flag.

[0155] 다음 테이블 6 및 테이블 7 은 `beta_offset` 가 변환 유형 의존적이고 독립적인 경우들에 대해 이들 2개의 새로운 디블로킹 플래그들의 사용을 예시한다.

[0156] 테이블 6

<code>slice_header() {</code>
<code>...</code>
<code>if(de-blocking_filter_control_present_flag) {</code>
<code> disable_de-blocking_filter_idc</code>
<code> if(disable_de-blocking_filter_idc != 1) {</code>
<code> de-blocking_offset_intra_inter_flag</code>
<code> de-blocking_offset_transform_flag</code>
<code> if(!slice_offset_intra_inter) {</code>
<code> if(!de-blocking_offset_transform_flag) {</code>
<code> slice_tc_offset</code>
<code> slice_beta_offset</code>
<code> } else {</code>
<code> slice_tc_offset_4x4</code>
<code> slice_tc_offset_8x8</code>
<code> slice_tc_offset_16x16</code>
<code> slice_tc_offset_32x32</code>
<code> slice_beta_offset</code>
<code> }</code>
<code> } else {</code>
<code> if(!de-blocking_offset_transform_flag) {</code>
<code> slice_tc_offset_intra</code>
<code> slice_beta_offset_intra</code>
<code> if(slice_type != 1) {</code>
<code> slice_tc_offset_inter</code>
<code> slice_beta_offset_inter</code>
<code> }</code>
<code> } else {</code>
<code> slice_tc_offset_4x4_intra</code>
<code> slice_tc_offset_8x8_intra</code>
<code> slice_tc_offset_16x16_intra</code>
<code> slice_tc_offset_32x32_intra</code>

[0157]

[0159] 테이블 7

slice_header() {
...
if(de-blocking_filter_control_present_flag) {
disable_de-blocking_filter_idc
if(disable_de-blocking_filter_idc != 1) {
de-blocking_offset_intra_inter_flag
de-blocking_offset_transform_flag
if(!slice_offset_intra_inter) {
if(!de-blocking_offset_transform_flag) {
slice_tc_offset
slice_beta_offset
} else {
slice_tc_offset_4x4
slice_tc_offset_8x8
slice_tc_offset_16x16
slice_tc_offset_32x32
slice_beta_offset_4x4
slice_beta_offset_8x8
slice_beta_offset_16x16
slice_beta_offset_32x32
}
} else {
if(!de-blocking_offset_transform_flag) {
slice_tc_offset_intra
slice_beta_offset_intra
if(slice_type != I) {
slice_tc_offset_inter
slice_beta_offset_inter
}
} else {
slice_tc_offset_4x4_intra
slice_tc_offset_8x8_intra
slice_tc_offset_16x16_intra

[0160]

파라미터 세트 (PPS) 로 시그널링할 수도 있다. `de-blocking_filter_control_present_flag` 는 1 비트로 표현될 수도 있다.

[0168] 더욱이, 단일 `BETA_OFFSET` 및 `INTRA_TC_OFFSET` 및 `INTER_TC_OFFSET` 양쪽을 이용하는 구현예에서, 엔트로피 인코딩 유닛 (56) 은 다음 테이블 9 에 따라서 구문 유닛들, 즉, `slice_tc_offset_intra`, `slice_tc_offset_intra_delta`, `slice_tc_offset_inter`, `slice_tc_offset_inter_delta`, 및 `slice_beta_offset` 를 슬라이스 헤더로 시그널링할 수도 있다:

[0169] 테이블 9

<code>Slice_header() {</code>
<code>...</code>
<code>if(de-blocking_filter_control_present_flag) {</code>
<code> disable_de-blocking_filter_idc</code>
<code> if(disable_de-blocking_filter_idc != 1) {</code>
<code> slice_beta_offset</code>
<code> slice_tc_offset_intra</code>
<code> slice_tc_offset_intra_delta</code>
<code> if(slice_type != I) {</code>
<code> slice_tc_offset_inter</code>
<code> slice_tc_offset_inter_delta</code>
<code> }</code>
<code> }</code>
<code>}</code>

[0170]

[0171] 엔트로피 인코딩 유닛 (56) 은 구문 엘리먼트들에 대한 적합한 구문적 설명을, 예컨대, 부호화된 정수 Exp-Golomb-코딩된 구문 엘리먼트 ("`se(v)`") 를 선택함으로써, 선택할 수도 있다.

[0172] 위에서 언급한 바와 같이, t_c 및 β 에 대한 여러 오프셋들은 APS, PPS 및/또는 슬라이스 헤더로 시그널링될 수도 있다. APS 는 샘플 적응적 오프셋 (SAO) 및 적응 루프 필터 (ALF) 와 연관된 파라미터들을 포함할 수도 있다. HEVC 에서의 최근의 발달들은 APS 의 채택, 및 `disable_de-blocking_filter_flag` 을 포함한 디블록킹 필터 파라미터들 및 APS 및/또는 슬라이스 헤더에서의 t_c 및 β 에 대한 디블록킹 파라미터들 오프셋들의 시그널링을 이끌었다.

[0173] 일부의 경우, 슬라이스 헤더는 APS 또는 PPS 에 규정된 디블록킹 파라미터들을 "상속하거나 (inherit)", "오버라이드하거나 (override)" 또는 복사하는 것 또는 슬라이스 헤더로 시그널링된 파라미터들을 이용하는 것을 나타내는 플래그를 포함할 수도 있다. 다음 테이블들 10-12 은 본 개시물의 기법에 따라 정의된 바와 같이 시퀀스 파라미터 세트 (SPS), APS 및 슬라이스 헤더 구문을 각각 규정한다. 이들 테이블들의 각각에서의 제안된 칼럼은 "X" 로 표시될 때, 본 개시물에서 설명하는 기법들의 지원 하에 제안되는 구문 엘리먼트들을 나타내며, 반면, "Y" 로 표시된 것은 표준 단체에 의해 최근에 제안되거나 및/또는 채택된 구문 엘리먼트들을 나타낸다.

[0174] 테이블 10: SPS 파라미터들

seq_parameter_set_rbsp() {	디스크립터	제안
profile_idc	u(8)	
(생략됨)		
chroma_pred_from_luma_enabled_flag	u(1)	
de-blocking_filter_control_present_flag	u(1)	X
if (de-blocking_filter_control_present_flag) {		X
de-blocking_filter_in_aps_enabled_flag	u(1)	Y
Note that if de-blocking_filter_control_present_flag=0, then de-blocking_filter_in_aps_enabled_flag=0 and aps_de-blocking_filter_flag=0 and disable_de-blocking_filter_flag=0 and beta_offset_div2=0 and tc_offset_div2=0 are inferred		X
}		
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()		
}		

[0175]

[0176]

위 테이블 10에 언급한 바와 같이, de-blocking_filter_control_present 플래그가 제로와 동일할 때, 제로와 동일한 de-blocking_filter_in_aps_enabled_flag, 제로와 동일한 aps_deblockign_filter_flag, 제로와 동일한 disable_de-blocking_filter_flag, 제로와 동일한 beta_offset_div2, 및 제로와 동일한 tc_offset_div2가 추론될 수도 있다. 그 결과, 이들 엔트리들 중 하나 이상이 어떤 컨텍스트들에서 APS 및/또는 슬라이스 헤더에서 생략될 수도 있으며, 테이블들에서 이들 엔트리들을 줄을 쳐서 지움으로써 아래 테이블들에 반영된다.

[0177] 테이블 11: APS 파라미터들

aps_rbsp() {	디스크립터	제안
aps_id	ue(v)	
aps_de-blocking_filter_flag	u(1)	Y
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_de-blocking_filter_flag){		Y
disable_de-blocking_filter_flag	u(1)	Y
if (!disable_de-blocking_filter_flag) {		Y
beta_offset_div2	se(v)	Y
te_offset_div2	se(v)	Y
tc_offset_intra_div2	se(v)	X
tc_offset_intra_delta	se(v)	X
tc_offset_inter_div2	se(v)	X
tc_offset_inter_delta	se(v)	X
}		
}		
....		

[0178]

[0179] 테이블 12: 슬라이스 헤더 파라미터들

slice_header() {	디스크립터	제안
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 de- blocking_filter_in_aps_enabled_flag)		Y
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(de-blocking_filter_control_present_flag) {		X
inherit_dbl_params_from_APS_flag	u(1)	Y
if(!inherit_dbl_params_from_APS_flag){		Y
disable_de-blocking_filter_flag	u(1)	Y
if(!disable_de-blocking_filter_flag) {		Y
beta_offset_div2	se(v)	Y
tc_offset_div2	se(v)	Y
tc_offset_intra_div2	se(v)	X
tc_offset_intra_delta	se(v)	X
if (slice_type != I) {		X
tc_offset_inter_div2	se(v)	X
tc_offset_inter_delta	se(v)	X
}		X
}		Y
}		Y
}		X
.....		

[0180]

[0181]

먼저 테이블 10 을 참조하면, 파라미터 "de-blocking_filter_in_aps_enabled_flag" 는, 제로의 값으로 설정될 때, 디블록킹 파라미터들이 슬라이스 헤더에 존재한다는 것을 나타내며, 반면, 1 의 값으로 설정될 때, 디블록킹 파라미터들이 APS 에 존재한다는 것을 나타낸다. 다음으로 테이블 11 을 참조하면, 파라미터 "aps_de-blocking_filter_flag" 는 디블록킹 파라미터들이 (1 의 값으로 설정될 때) APS 에 존재하고 있다거나 또는 (제로의 값으로 설정될 때) 존재하지 않는다는 것을 나타낸다. 테이블 11 및 테이블 12 양쪽에서, 파라미터 "disable_de-blocking_filter_flag" 는, 제로 값으로 설정될 때, 디블록킹 필터가 인에이블되어 있다는 것을 나타내며, 1 의 값으로 설정될 때, 디블록킹 필터가 디스에이블되어 있다는 것을 나타낸다. 테이블 11 및 테이블 12 양쪽에서, "beta_offset_div2" 및 "tc_offset_div2" 파라미터들은 t_c 및 β 임계치들을 탐색하는데 사용되는 QP 에 추가되는 오프셋들을 지칭할 수도 있다 (여기서, 파라미터들은 2 로 나누어진다). 테이블 12 에서, 파라미터 "inherit_dbl_params_from_APS_flag" 는, 1 의 값으로 설정될 때, APS 에서의 디블록킹 파라미터들이 사용될 것임을 나타내며, 제로의 값으로 설정될 때, 슬라이스 헤더에서 뒤따르는 디블록킹 파라미터들이 사용될 것임을 나타낸다.

[0182]

엔트로피 인코딩 유닛 (56) 은 본 개시물에서 설명하는 기법들에 따라 추가적인 파라미터들을 SPS, PPS, APS 또

는 슬라이스 헤더에 규정할 수도 있다. 예를 들어, 테이블 10 을 참조하면, 엔트로피 인코딩 유닛 (56) 은 "de-blocking_filter_control_present_flag" 를 SPS 또는 PPS 에 규정할 수도 있으며, 여기서, 이 파라미터는, 1 의 값으로 설정될 때, 디블록킹 파라미터들이 APS, PPS 및/또는 슬라이스 헤더에 존재한다는 것을 나타내며, 제로의 값으로 설정될 때, 어떤 디블록킹 파라미터들도 APS, PPS 및 슬라이스 헤더에 존재하지 않고 디폴트 디블록킹 파라미터들이 사용될 것임을 나타낸다. 또 다른 예로서, 테이블 11 및 테이블 12 을 참조하면, 엔트로피 인코딩 유닛 (56) 은 "tc_offset_intra_div2", "tc_offset_intra_delta", "tc_offset_inter_div2", 및 "tc_offset_inter_delta" 파라미터들을 APS, PPS 및/또는 슬라이스 헤더에 규정할 수도 있으며, 여기서, 이들 디블록킹 필터 파라미터들의 사용이 다음 의사-코드에 대해 예시된다:

- [0183] transform_type = N x N (= 4 x 4 또는 8 x 8 또는 16 x 16 또는 32 x 32)
- [0184] a. INTRA_TC_OFFSET = (2×tc_offset_intra_div2) + tc_offset_NxN_intra_delta
- [0185] b. INTER_TC_OFFSET = (2×tc_offset_inter_div2) + tc_offset_NxN_inter_delta
- [0186] c. BETA_OFFSET = 2×beta_offset_div2
- [0187] Tc_offset_NxN_intra_delta = tc_offset_intra_delta × factor_NxN
- [0188] Tc_offset_NxN_inter_delta = tc_offset_inter_delta × factor_NxN
- [0189] (시그널링되지 않는) 계산에 사용되는 상수 값들의 예들:
- [0190] d. Factor_4x4 = 0
- [0191] e. Factor_8x8 = 1
- [0192] f. Factor_16x16 = 2
- [0193] g. Factor_32x32 = 3
- [0194] IF (Intra mode) THEN TcOffset = DEFAULT_INTRA_TC_OFFSET(=2) + INTRA_TC_OFFSET; ELSE TcOffset = INTER_TC_OFFSET
- [0195] BetaOffset = BETA_OFFSET
- [0196] t_c : $Q = \text{Clip3}(0, \text{MAX_QP} + \text{DEFAULT_INTRA_TC_OFFSET}, \text{QP} + \text{TcOffset})$ (루마 및 크로마 디블록킹)
- [0197] β : $Q = \text{Clip3}(0, \text{MAX_QP}, \text{QP} + \text{BetaOffset})$ (루마 디블록킹 단독)
- [0198] 상기 의사-코드는 일반적으로 INTRA_TC_OFFSET 가 2 곱하기 tc_offset_intra_div2 파라미터 플러스 tc_offset_NxN_intra_delta 파라미터와 동일할 수도 있다는 것을 나타낸다. 게다가, 이 의사-코드는 INTER_TC_OFFSET 가 2 곱하기 tc_offset_inter_div2 파라미터 플러스 tc_offset_NxN_inter_delta 파라미터와 동일할 수도 있으며, 반면 BETA_OFFSET 는 2 곱하기 beta_offset_div2 파라미터와 동일할 수도 있다는 것을 나타낸다. 더욱이, Tc_offset_NxN_intra_delta 는 tc_offset_intra_delta 파라미터 곱하기 factor_NxN 와 동일할 수도 있으며, 이의 예들이 상기 의사-코드에 리스트된다. 더욱이, Tc_offset_NxN_inter_delta 파라미터는 tc_offset_inter_delta 곱하기 factor_NxN 과 동일할 수도 있다. 엔트로피 인코딩 유닛 (56) 은 상기 의사-코드를 구현하여, 여러 파라미터들을 시그널링할 수도 있다.
- [0199] 인자들의 여러 예들을 뒤따르는 의사-코드의 나머지 부분들은, 그 모드가 인트라 모드이면, TcOffset 이 (2 와 동일한) DEFAULT_INTRA_TC_OFFSET 플러스 INTRA_TC_OFFSET 과 동일하다는 것을 나타낸다. 그러나, 그 모드가 인트라 모드가 아니면, TcOffset 는 INTER_TC_OFFSET 과 동일하다. BetaOffset 는 BETA_OFFSET 과 동일하다. 이 의사-코드에 따르면, Q 값은 최종 Q 값이 0 과 MAX_QP+DEFAULT_INTRA_TC_OFFSET (여기서, MAX_QP 는 일반적으로 51 과 동일함) 의 범위 내에 들어가도록 클리핑하는 범위를 조건으로 하여, QP 플러스 그 결정된 TcOffset 값으로서, t_c 임계치에 대해 결정된다. 이와 유사하게, Q 값은 최종 Q 값이 0 과 MAX_QP (여기서, 다시 MAX_QP 는 일반적으로 51 과 동일함) 의 범위 내에 들어가도록 클리핑하는 범위를 조건으로 하여, QP 플러스 그 결정된 BetaOffset 값을 β 임계치에 대해 결정된다.
- [0200] 테이블들 10-12 에 대해 위에서 설명한 기법들에 더해서, 인트라 및 인터에 대한 별개의 오프셋 파라미터들을 시그널링하는 de-blocking_offset_intra_inter_flag (상기 참조) 가 SPS, PPS, APS 또는 슬라이스 헤더에 도입

될 수 있다 (예컨대, 다음 테이블들 13 및 14 을 참조한다).

[0201]

테이블 13: APS 파라미터들

aps_rbsp() {	디스크립터	제안
aps_id	ue(v)	
aps_de-blocking_filter_flag	u(1)	Y
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_de-blocking_filter_flag){		Y
disable_de-blocking_filter_flag	u(1)	Y
if (!disable_de-blocking_filter_flag) {		Y
beta_offset_div2	se(v)	Y
tc_offset_div2	se(v)	Y
de-blocking_offset_intra_inter_flag	u(1)	X
if (de-blocking_offset_intra_inter_flag) {		X
beta_offset_intra_div2	se(v)	X
beta_offset_inter_div2	se(v)	X
tc_offset_intra_div2	se(v)	X
tc_offset_intra_delta	se(v)	X
tc_offset_inter_div2	se(v)	X
tc_offset_inter_delta	se(v)	X
} else {		X
beta_offset_div2	se(v)	X
tc_offset_div2	se(v)	X
tc_offset_delta	se(v)	X
}		X
}		Y
}		Y
....		

[0202]

[0203] 테이블 14: 슬라이스 파라미터들

slice_header() {	디스크립터	제안
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 de- blocking_filter_in_aps_enabled_flag)		Y
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(de-blocking_filter_control_present_flag) {		X
inherit_dbl_params_from_APS_flag	u(1)	Y
if(!inherit_dbl_params_from_APS_flag) {		Y
disable_de-blocking_filter_flag	u(1)	Y
if(!disable_de-blocking_filter_flag) {		Y
beta_offset_div2	se(v)	Y
tc_offset_div2	se(v)	Y
de-blocking_offset_intra_inter_flag	u(1)	X
if(de-blocking_offset_intra_inter_flag) {		X
beta_offset_intra_div2	se(v)	X
tc_offset_intra_div2	se(v)	X
tc_offset_intra_delta	se(v)	X
if(slice_type != I) {		X
beta_offset_inter_div2	se(v)	X
tc_offset_inter_div2	se(v)	X
tc_offset_inter_delta	se(v)	X
}		X
} else {		X
beta_offset_div2	se(v)	X
tc_offset_div2	se(v)	X
tc_offset_delta	se(v)	X
}		X
}		Y
}		Y
}		X
.....		

[0204]

[0205]

일부 대안들에서, 엔트로피 인코딩 유닛 (56) 은 de-blocking_filter_control_present_flag 및 de-blocking_filter_in_aps_enabled_flag 를 PPS 로 시그널링할 수도 있다. PPS 구문의 일 예가 다음 테이블 15 에 나타내며, 테이블 16 의 예시적인 슬라이스 헤더 구문에 대해 나타낸 바와 같이 파라미터들을 슬라이스 헤더 (또는, 독립적으로 디코딩가능한 유닛의 헤더) 에 규정함으로써 오버라이드될 수도 있다:

[0206] 테이블 15: PPS 파라미터들

...	
de-blocking_filter_control_present_flag	u(1)
if(de-blocking_filter_control_present_flag) {	
de-blocking_filter_override_enabled_flag	u(1)
pps_disable_de-blocking_filter_flag	u(1)
if(!pps_disable_de-blocking_filter_flag) {	
pps_beta_offset_div2	sc(v)
pps_tc_offset_div2	se(v)
}	
}	
...	

[0207]

[0208] 테이블 16: 슬라이스 헤더 구분

...	
if(de-blocking_filter_control_present_flag) {	
if(de-blocking_filter_override_enabled_flag)	
de-blocking_filter_override_flag	u(1)
if(de-blocking_filter_overriding_flag) {	
slice_header_disable_de-blocking_filter_flag	u(1)
if(!disable_de-blocking_filter_flag) {	
slice_header_beta_offset_div2	se(v)
slice_header_tc_offset_div2	se(v)
}	
}	
}	
...	

[0209]

[0210] t_c 또는 β 임계치들을 탐색하기 전에, QP 값 (Doc. JCTVC-I1003_d4, 2012년 4월-5월, 스위스, 제네바, 9차 JCT-VC 회의, B. Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, T. Wiegand, "High efficiency video coding (HEVC) text specification draft 7") 에 제시된 바와 같이 이웃하는 P 및 Q 블록들의 QP 값들의 평균) 을 수정하는 디블록킹 오프셋 파라미터들은 다음과 같다: $pps_beta_offset_div2$, $pps_tc_offset_div2$, $slice_header_beta_offset_div2$, $slice_header_tc_offset_div2$. 이들 오프셋 파라미터들 중 어느 파라미터도 디블록킹 필터 강도의 변환 사이즈 의존적인 제어를 지원하지 않을 수도 있으며, 반면 자각할 수 있는 품질 평가들은 어떤 비디오 시퀀스들이 변환 사이즈 의존적인 심한 블로킹 아티팩트들로 고통을 받을 수도 있다는 점을 입증하였다. 이들 관찰되는 블로킹 아티팩트들은 상대적으로 높은 QP 값들 (예를 들어, 30 을 초과하는 QP 값들에 대해) 에서, 그리고 SPS 구분 파라미터들 $log2_diff_max_min_transform_block_size$ 및 $log2_min_transform_block_size_minus2$ 에 의해 규정된 바와 같은 최대 변환 사이즈에 대해 볼 수 있을 수도 있다. 이들 큰 블로킹 아티팩트들은 최대 변환 사이즈가 32×32 이면 가장 볼 수 있을 수도 있으며; 그러나, 블로킹 아티팩트들은 또한 최대 변환 사이즈가 16×16 또는 8×8 이면 볼 수 있지만, 변환 사이즈 4×4 에 대해서는 어쩌면 덜 볼 수 있을 수도 있다.

[0211] 따라서, 본 개시물에서 설명하는 기법들은 엔트로피 인코딩 유닛 (56) 으로 하여금, 2개의 인접한 비디오 블록들 P 및 Q 중 적어도 하나가 최대 변환 사이즈를 가진 변환 유닛에 포함되어 있는 경우에 디블록킹 강도를 제어하기 위해 특정의 디블록킹 조정 파라미터들을 시그널링 가능하게 할 수도 있다. 즉, 본 개시물의 기법들은 2개의 인접한 비디오 블록들 P 및 Q 중 적어도 하나가 최대 변환 사이즈를 가진 변환 유닛에 포함되어 있는 경우에 디블록킹 필터 강도를 조정할 수도 있다. 블록들 P 와 Q 사이의 공통 에지는 일반적으로 디블록킹 필터링을 겪는다. 비디오 블록 P 또는 Q 중 어느 것도 최대 사이즈 변환 유닛에 포함되지 않으면, 현재의 디블록킹 조정 파라미터들이 적용된다. 다음 테이블들 17 및 18 은 PPS 구분 및 슬라이스 헤더 구분에 대한

변화들을 규정하며, 여기서, 변화들은 회색 하이라이팅을 이용하여 나타내어진다.

테이블 17: PPS 구분

...	
de-blocking_filter_control_present_flag	u(1)
if(de-blocking_filter_control_present_flag) {	
de-blocking_filter_override_enabled_flag	u(1)
pps_disable_de-blocking_filter_flag	u(1)
if(!pps_disable_de-blocking_filter_flag) {	
pps_beta_offset_div2	se(v)
pps_tc_offset_div2	se(v)
pps_beta_offset_max_tu_div2	se(v)
pps_tc_offset_max_tu_div2	se(v)
}	
}	
...	

테이블 18: 슬라이스 헤더 구분

...	
if(de-blocking_filter_control_present_flag) {	
if(de-blocking_filter_override_enabled_flag)	
de-blocking_filter_override_flag	u(1)
if(de-blocking_filter_overriding_flag) {	
slice_header_disable_de-blocking_filter_flag	u(1)
if(!disable_de-blocking_filter_flag) {	
slice_header_beta_offset_div2	se(v)
slice_header_tc_offset_div2	se(v)
slice_header_beta_offset_max_tu_div2	se(v)
slice_header_tc_offset_max_tu_div2	se(v)
}	
}	
}	
...	

제안된 구분 파라미터들의 의미들은 다음과 같다 (파라미터 값들은 존재하지 않으면 0 인 것으로 추론된다):

pps_beta_offset_max_tu_div2 / slice_header_beta_offset_max_tu_div2 는 β 에 대한 디폴트 디블록킹 파라미터 오프셋이 화상 파라미터 세트를 참조하는 화상들에 대한 슬라이스 헤더에 존재하는 디블록킹 파라미터 오프셋에 의해 오버라이드되지 않는 한, 화상 파라미터 세트를 참조하는 화상들에서 최대 변환 유닛 사이즈의 에지들에 적용되는 (2 로 나눈) β 에 대한 디폴트 디블록킹 파라미터 오프셋을 규정한다.

pps_tc_offset_max_tu_div2 / slice_header_tc_offset_max_tu_div2 는 t_c 에 대한 디폴트 디블록킹 파라미터 오프셋이 화상 파라미터 세트를 참조하는 화상들에 대한 슬라이스 헤더에 존재하는 디블록킹 파라미터 오프셋에 의해 오버라이드되지 않는 한, 화상 파라미터 세트를 참조하는 화상들에서 최대 변환 유닛 사이즈의 에지들에 적용되는 (2 로 나눈) t_c 에 대한 디폴트 디블록킹 파라미터 오프셋을 규정한다.

다른 관련되는 디블록킹 조정 파라미터들의 의미들은 다음과 같다:

pps_beta_offset_div2 / slice_header_beta_offset_div2 및 **pps_tc_offset_div2 / slice_header_tc_offset_div2** 는 디폴트 디블록킹 파라미터 오프셋들이 화상 파라미터 세트를 참조하는 화상들에 대한 슬라이스 헤더에 존재하는 디블록킹 파라미터 오프셋들에 의해 오버라이드되지 않는 한, 화상 파라미터

세트를 참조하는 화상들에서 최대 변환 유닛 사이즈의 에지들이 또한 아닌 변환 유닛들의 에지들에 적용되는 (2로 나눈) β 및 t_c 에 대한 디폴트 디블록킹 파라미터 오프셋들을 규정한다.

- [0221] 최대 변환 유닛 사이즈들에 대한 제안된 디블록킹 조정 파라미터들은 다음 의사-코드로 규정된다 (pps_ 또는 slice_header_ prefixes 는 간결성을 위해 생략되며; QP 는 블록들 P 및 Q 의 QP 값들의 평균 값이다):
- [0222] - IF (P 또는 Q 가 최대 변환 유닛 사이즈에 포함된다) THEN TC_OFFSET = tc_offset_max_tu_div2; ELSE INTRA_TC_OFFSET = tc_offset_div2
- [0223] - IF (P 또는 Q 가 최대 변환 유닛 사이즈에 포함된다) THEN BETA_OFFSET = beta_offset_max_tu_div2; ELSE BETA_OFFSET = beta_offset_div2
- [0224] - IF (P 또는 Q 가 MODE_INTRA 를 갖는다) THEN TcOffset = DEFAULT_INTRA_TC_OFFSET(=2) + TC_OFFSET; ELSE TcOffset = TC_OFFSET
- [0225] - BetaOffset = BETA_OFFSET
- [0226] - t_c : $Q = \text{Clip3}(0, \text{MAX_QP} + \text{DEFAULT_INTRA_TC_OFFSET}, \text{QP} + \text{TcOffset})$ (루마 및 크로마 디블록킹)
- [0227] - β : $Q = \text{Clip3}(0, \text{MAX_QP}, \text{QP} + \text{BetaOffset})$ (루마 디블록킹 단독)
- [0228] 일부의 경우, 단지 제안된 beta_offset_max_tu_div2 및 tc_offset_max_tu_div2 파라미터들 중 하나 또는 양쪽이 시그널링될 수도 있다. 또한, "div2" 또는 제안된 파라미터들의 2 스케일링으로 나누는 것은, 다른 스케일링 인자들 또는 정수 오프셋들 (예를 들어, X 가 정수인 _minusX) 이 사용될 수도 있거나, 또는 어떤 스케일링도 사용되지 않을 수도 있기 때문에 (beta_offset_max_tu 및 tc_offset_max_tu), 옵션적임에 유의한다.
- [0229] 상기의 제안된 beta_offset_max_tu_div2 및 tc_offset_max_tu_div2 파라미터들은 변환 유닛들을 포함하는 코딩 유닛의 인트라 또는 인터 코딩 모드와 독립적으로 적용될 수도 있다. 이의 대안으로, 제안된 파라미터들은 인트라 또는 인터 코딩 모드 의존적일 수도 있다. 예를 들어, 파라미터들 beta_offset_intra_max_tu_div2 및/또는 beta_offset_inter_max_tu_div2 및/또는 tc_offset_intra_max_tu_div2 및/또는 tc_offset_inter_max_tu_div2 는 상기 구문 예들에서 규정된 방법과 유사하게 시그널링될 수도 있다. 인트라 또는 인터 모드에 대한 별개의 파라미터들에 대한 시그널링은 위에서 설명한 기법들의 이전 양태들에 예시된 바와 같은 de-blocking_offset_intra_inter_flag 와 같은 인에이블 플래그에 의존할 수도 있다. 다음 규칙은 인트라 또는 인터 파라미터들을 사용할지 여부를 결정하여 비디오 블록들 P 와 Q 사이의 공통 에지를 디블록킹하기 위해 적용될 수도 있으며: P 또는 Q (또는 양쪽) 가 인트라-코딩된 코딩 유닛에 있으면, 인트라 파라미터들이 에지에 적용되며, 그렇지 않으면, 인터 파라미터들이 적용된다.
- [0230] 이의 대안으로, 제안된 기법들은 Doc. JCTVC-I1003_d4, 2012년 4월-5월, 스위스, 제네바, 9차 JCT-VC 회의, B. Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, T. Wiegand, "High efficiency video coding (HEVC) text specification draft 7" 에 제시된 HM7 디블록킹 필터에서 계산되는 값들 0 내지 2 에 더해서, 추가적인 경계 강도 값 3 (또는, 다른 개수) 를 계산하기를 제안하는, Doc. JCTVC-I0244, 2012년 4월-5월, 스위스, 제네바, 9차 JCT-VC 회의, D.-K. Kwon, M. Budagavi, "Transform size dependent deblocking filter" 에 제시된 제안들에 의하여 추가로 향상시킬 수도 있다. 본 개시물에서 설명하는 기법들에 따르면, 이웃하는 P 또는 Q 블록들 중 적어도 하나가 인트라-코딩된 코딩 유닛에 포함되면 그리고 인트라-코딩된 블록들 (P 및/또는 Q) 중 적어도 하나가 최대 사이즈의 변환 유닛, 예를 들어, 32 x 32 또는 16 x 16 등에 포함되면, 경계 강도는 3 의 값으로 설정된다.
- [0231] 경계 강도 값이 3 과 동일하면, 본 개시물에서 설명하는 기법들에 따라서 제시되는 최대 변환 유닛 사이즈들 (beta_offset_max_tu_div2 및/또는 beta_offset_intra_max_tu_div2 및/또는 tc_offset_max_tu_div2 및/또는 tc_offset_intra_max_tu_div2) 에 대한 디블록킹 조정 파라미터들은 beta 및 t_c 오프셋들로서 채용되며; 아니면, tc_offset_div2 및 beta_offset_div2 가 사용된다.
- [0232] 이의 대안으로, 경계 강도 값이 3 과 동일하면, 강한 디블록킹 필터가 적용되며; 아니면, 약한 필터가 적용된다. 이 경우, 본 개시물에서 설명하는 기법들에 따라서 규정된 최대 transform_unit_size들 (beta_offset_max_tu_div2 및/또는 beta_offset_intra_max_tu_div2 및/또는 tc_offset_max_tu_div2 및/또는 tc_offset_intra_max_tu_div2) 에 대한 디블록킹 조정 파라미터들이 적용되지 않고 시그널링되지 않으며, tc_offset_div2 및/또는 beta_offset_div2 이 적용된다.

[0233] 이의 대안으로, 경계 강도 값이 3 과 동일하면, 강한 디블록킹 필터가 적용되며; 아니면, 약한 필터가 적용된다. 이 경우, 최대 transform_unit_size들 (beta_offset_max_tu_div2 및/또는 beta_offset_intra_max_tu_div2 및/또는 tc_offset_max_tu_div2 및/또는 tc_offset_intra_max_tu_div2) 에 대한 디블록킹 조정 파라미터들은 강한 디블록킹 필터에 적용되며, tc_offset_div2 및/또는 beta_offset_div2 은 약한 디블록킹 필터에 적용된다.

[0234] 추가적인 경계 강도 값들은 인터-코딩된 경우에 대해 계산될 수도 있다. 예를 들어, 블록 P 도 또는 Q 도 인트라-코딩되지 않고, P 또는 Q 블록들 중 적어도 하나가 최대 사이즈의 변환 유닛에 포함되어 있으면, 추가적인 경계 강도 값, 예를 들어, 값 4 가 정의된다. 위 인트라 경우들과 유사하게, 최대 transform_unit_size 들 (beta_offset_max_tu_div2 및/또는 beta_offset_intra_max_tu_div2 및/또는 tc_offset_max_tu_div2 및/또는 tc_offset_intra_max_tu_div2) 에 대한 제안된 디블록킹 조정 파라미터들이 적용된다.

[0235] 일부의 경우, 최대 사이즈의 변환 유닛들은 디블록킹 강도를 조정하기 위해 그 제안된 beta_offset_max_tu_div2 및 tc_offset_max_tu_div2 파라미터들을 적용하기 전에 적어도 하나의 논-제로 코딩된 계수를 포함할 필요가 있을 수도 있다. 이것은 코딩된 블록 플래그 (cbf) 를 이용하여 체크될 수도 있다.

[0236] 제안된 파라미터들에 더해서, 플래그가 그들의 시그널링 및/또는 제안된 기능을 인에이블하기 위해 시그널링될 수도 있다. 다음 테이블들 19 및 20 은 구문 변화들을 예시하며, 여기서 다시 변화들은 회색 하이라이팅을 이용하여 표시된다:

[0237] 테이블 19: PPS 구문

...	
de-blocking_filter_control_present_flag	u(1)
if(de-blocking_filter_control_present_flag) {	
de-blocking_filter_override_enabled_flag	u(1)
pps_disable_de-blocking_filter_flag	u(1)
if(!pps_disable_de-blocking_filter_flag) {	
pps_beta_offset_div2	se(v)
pps_tc_offset_div2	se(v)
de-blocking_max_tu_offset_enabled_flag	u(1)
if(de-blocking_max_tu_offset_enabled_flag) {	

[0238]

pps_beta_offset_max_tu_div2	se(v)
pps_tc_offset_max_tu_div2	se(v)
}	
}	
...	

[0239]

[0240] 테이블 20: 슬라이스 헤더 구문

...	
if(de-blocking_filter_control_present_flag) {	
if(de-blocking_filter_override_enabled_flag)	
de-blocking_filter_override_flag	u(1)
if(de-blocking_filter_overriding_flag) {	
slice_header_disable_de-blocking_filter_flag	u(1)
if(!disable_de-blocking_filter_flag) {	
slice_header_beta_offset_div2	se(v)
slice_header_tc_offset_div2	se(v)
if(de-blocking_max_tu_offset_enabled_flag) {	
slice_header_beta_offset_max_tu_div2	se(v)
slice_header_tc_offset_max_tu_div2	se(v)
}	
}	
}	
...	

[0241]

[0242] de-blocking_max_tu_offset_enabled_flag 의 의미들은 다음 대안들 중 임의의 것일 수도 있다:

[0243] • de-blocking_max_tu_offset_enabled_flag 는 플래그 값이 1 과 동일하면 beta_offset_max_tu_div2 또는 tc_offset_max_tu_div2 구문 파라미터들의 시그널링을 인에이블할 수도 있으며; 아니면, 이들 파라미터들은 시그널링되지 않으며 그들의 값은 0 과 동일한 것으로 추론된다.

[0244] • de-blocking_max_tu_offset_enabled_flag 는 플래그 값이 1 과 동일하면 파라미터들 beta_offset_max_tu_div2 또는 tc_offset_max_tu_div2 구문 파라미터들의 시그널링을 인에이블할 수도 있으며, 그렇지 않으면, 이들 파라미터들은 시그널링되지 않으며, 그들의 값은, 존재하지 않으면 0 과 동일한 것으로 추론되는 대응하는 beta_offset_div2 및 tc_offset_div2 파라미터들과 동일한 것으로 추론된다.

[0245] 일부의 경우, 이 기법들은 2012년 6월 21일자에 출원된 미국 가특허출원 제 61/662,833호에 설명된 기법들과 함께 사용될 수도 있으며, 이의 전체 내용들이 여기 본 개시물에 포함되며, 여기서 본 기법들은 디블록킹 강도를 증가시키기 위해 위에서-포함된 가특허 출원에 설명된 베타-곡선들 및 tc-곡선들을 수정한다.

[0246] 일부의 경우, HM7 디블록킹 필터에서 계산되는 값들 0 내지 2 에 더해서, 추가적인 경계 강도 값 3 (또는, 다른 개수) 이 계산될 수도 있으며, 여기서, HM7 은 Doc. JCTVC-I1003_d4, 2012년 4월-5월, 스위스, 제노바, 9차 JCT-VC 회의, B. Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, T. Wiegand 에 의한, "High efficiency video coding (HEVC) text specification draft 7" 이라는 제목의 문서를 지칭하며, 이의 전체 내용들은 본원에 참고로 포함된다.

[0247] 이들의 경우, 경계 강도는, 이웃하는 P 또는 Q 블록들 중 적어도 하나가 인트라-코딩된 코딩 유닛에 포함되어 있으면, 그리고 인트라-코딩된 블록들 (P 및/또는 Q) 중 적어도 하나가 최대 사이즈의 변환 유닛, 예를 들어, 32 x 32 또는 16 x 16 등에 포함되어 있으면, 3 의 값으로 설정된다. 위에서 설명한 기법들의 양태들 중 하나 이상과는 대조적으로, 경계 강도 값이 3 과 동일하면, 상수 tc 오프셋 (constant_max_tu_tc_offset) 및/또는 베타 오프셋 (constant_max_tu_beta_offset) 이 적용되며, 예를 들어, tc_offset_div2 및 beta_offset_div2 를 시그널링하는 것을 통해서 추가로 수정될 수도 있다.

[0248] 다음은 어떻게 비디오 코더가 이 값들을 결정할 수 있는지 (그리고, 후속하여 이 값들을 비트스트림으로 시그널링) 할 수 있는지의 일 예를 예시한다:

[0249] - IF (P 또는 Q 가 MODE_INTRA 를 갖는다) 및 (P 또는 Q 가 최대 변환 유닛 사이즈에 포함된다) THEN TC_OFFSET = constant_max_tu_tc_offset + 2*tc_offset_div2; ELSE TC_OFFSET = 2*tc_offset_div2

[0250] - IF (P 또는 Q 가 MODE_INTRA 를 갖는다) 및 (P 또는 Q 가 최대 변환 유닛 사이즈에 포함된다) THEN

BETA_OFFSET = constant_max_tu_beta_offset + 2*beta_offset_div2; ELSE BETA_OFFSET = 2*beta_offset_div2

[0251] - 옵션적으로; IF (P 또는 Q 가 MODE_INTRA 를 갖는다) THEN TcOffset = DEFAULT_INTRA_TC_OFFSET(=2) + TC_OFFSET; ELSE TcOffset = TC_OFFSET

[0252] - BetaOffset = BETA_OFFSET

[0253] - t_c : $Q = \text{Clip3}(0, \text{MAX_QP} + \text{DEFAULT_INTRA_TC_OFFSET}, \text{QP} + \text{TcOffset})$ (루마 및 크로마 디블록킹)

[0254] - β : $Q = \text{Clip3}(0, \text{MAX_QP}, \text{QP} + \text{BetaOffset})$ (루마 디블록킹 단독)

[0255] 이의 대안으로, 비디오 코더는 이 값들을 뒤따르는 방법으로 결정할 수도 있다:

[0256] - IF (P 는 MODE_INTRA 를 가지며 P 는 최대 변환 유닛 사이즈에 포함되어 있다) 또는 (Q 가 MODE_INTRA 이며 Q 가 최대 변환 유닛 사이즈에 포함되어 있다) THEN TC_OFFSET = constant_max_tu_tc_offset + 2*tc_offset_div2; ELSE TC_OFFSET = 2*tc_offset_div2

[0257] - IF (P 는 MODE_INTRA 를 가지며 P 는 최대 변환 유닛 사이즈에 포함되어 있다) 또는 (Q 가 MODE_INTRA 이며 Q 가 최대 변환 유닛 사이즈에 포함되어 있다) THEN BETA_OFFSET = constant_max_tu_beta_offset + 2*beta_offset_div2; ELSE BETA_OFFSET = 2*beta_offset_div2

[0258] - 옵션적으로; IF (P 또는 Q 가 MODE_INTRA 를 갖는다) THEN TcOffset = DEFAULT_INTRA_TC_OFFSET(=2) + TC_OFFSET; ELSE TcOffset = TC_OFFSET

[0259] - BetaOffset = BETA_OFFSET

[0260] - t_c : $Q = \text{Clip3}(0, \text{MAX_QP} + \text{DEFAULT_INTRA_TC_OFFSET}, \text{QP} + \text{TcOffset})$ (루마 및 크로마 디블록킹)

[0261] - β : $Q = \text{Clip3}(0, \text{MAX_QP}, \text{QP} + \text{BetaOffset})$ (루마 디블록킹 단독)

[0262] 비디오 코더는 인터-코딩된 경우에 대해 추가적인 경계 강도 값들을 결정하거나 또는 계산할 수도 있다. 예를 들어, 블록 P 도 또는 Q 도 인트라-코딩되지 않고, P 또는 Q 블록들 중 적어도 하나가 최대 사이즈의 변환 유닛에 포함되어 있으면, 추가적인 경계 강도 값이 정의될 수도 있다 (예컨대, 4 의 값). 위에서 설명한 인트라 경우들과 유사하게, 상수 t_c 오프셋 (constant_max_tu_tc_offset) 및/또는 베타 오프셋 (constant_max_tu_beta_offset) 이 적용되며, 예를 들어, tc_offset_div2 및 beta_offset_div2 를 시그널링하는 것을 통해서 추가로 수정될 수도 있다.

[0263] 일부의 경우, 비디오 코더는 적어도 하나의 P 또는 Q 블록이 최대 사이즈의 변환 유닛에 포함되어 있으면, 블록 P 또는 Q 의 인터 또는 인트라 모드를 체크하지 않고, 3 의 경계 강도 값 (또는, 다른 값) 을 할당할 수도 있다. 위 인트라 경우들과 유사하게, 상수 t_c 오프셋 (constant_max_tu_tc_offset) 및/또는 베타 오프셋 (constant_max_tu_beta_offset) 이 제공되며, 예를 들어, tc_offset_div2 및 beta_offset_div2 를 시그널링하는 것을 통해서 추가로 수정될 수도 있다.

[0264] 본 기법들의 또 다른 양태는 위에서 설명한 바와 같이, 구문 엘리먼트들 log2_min_transform_block_size_minus2 및 log2_diff_max_min_transform_block_size 를 통해서 SPS 에 규정되는 최대 허용 사이즈의 변환 유닛 (TU) 또는 변환 블록 (TB) 에 포함되는 블록 P 또는 Q 에 기초하여, 추가적인 경계 (Bs) 값들을 포함할 수도 있다. 본 기법들의 이 양태는 이웃하는 P 또는 Q 블록들 중 적어도 하나가 인트라-코딩된 코딩 유닛 (CU) 에 포함되어 있고 인트라-코딩된 블록들 (P 및/또는 Q) 중 적어도 하나가 최대 사이즈 (예컨대, 32 x 32, 16 x 16 등) 의 TU 또는 TB 에 포함되어 있으면, 비디오 코더가 경계 강도를 4 의 값 (또는, 다른 적합한 값) 으로 설정한다는 점에서, 위에서 설명한 양태와는 상이할 수도 있다. 그렇지 않으면, 비디오 코더는 P 또는 Q 블록들 중 적어도 하나가 최대 사이즈의 TB 에 포함되어 있을 때 BS 값을 3 의 값 (또는, 다른 적합한 값) 으로 설정한다. 이들 조건들 중 어느 조건도 만족되지 않으면, 비디오 코더는 HEVC 에 정의된 것에 따라서 Bs 값을 계산할 수도 있다.

[0265] 다음 의사-코드는 Bs 유도를 예시한다:

[0266] - IF (P 가 MODE_INTRA 를 가지며 P 가 최대 TB 사이즈에 포함되어 있다) 또는 (Q 가 MODE_INTRA 를 가지며 Q 가 최대 TB 사이즈에 포함되어 있다) THEN Bs = 4,

- [0267] - ELSE IF (P 가 최대 TB 사이즈에 포함되어 있다) 또는 (Q 가 최대 TB 사이즈에 포함되어 있다) THEN Bs = 3,
- [0268] - ELSE IF (P 또는 Q 가 MODE_INTRA 를 갖는다) THEN Bs = 2,
- [0269] - ELSE 추가적인 Bs 유도는 HEVC 사양을 따른다.
- [0270] 일부의 경우, 비디오 코더는 다음 의사-코드를 이용하여, 오직 하나의 Bs 값, 예를 들어, 3 과 동일한 Bs 를 추가함으로써 상기 프로세스를 간략화할 수도 있다:
- [0271] - IF (P 가 최대 TB 사이즈에 포함되어 있다) 또는 (Q 가 최대 TB 사이즈에 포함되어 있다) THEN Bs = 3,
- [0272] - ELSE IF (P 또는 Q 가 MODE_INTRA 를 갖는다) THEN Bs = 2,
- [0273] - ELSE 추가적인 Bs 유도는 HEVC 사양을 따른다.
- [0274] 일부의 경우, 비디오 코더는 경계 강도 값 계산을 구현하여, 블록 P 또는 Q 가 단지 최대 TB 사이즈 대신 최소 사이즈 N x M (또는 N x N) 보다 더 큰 (또는, 더 작은) TB 사이즈에 포함되는지 여부의 체크를 포함할 수도 있다. 다음 의사-코드는 이 경계 강도 값 계산의 변형예를 예시할 수도 있다:
- [0275] - IF (P 가 MODE_INTRA 를 가지며 P 가 TB 사이즈 $\geq N \times M$ 에 포함되어 있다) 또는 (Q 가 MODE_INTRA 를 가지며 Q 가 TB 사이즈 $\geq N \times M$ 에 포함되어 있다) THEN Bs = 4,
- [0276] - ELSE IF (P 가 TB 사이즈 $\geq N \times M$ 에 포함되어 있다) 또는 (Q 가 TB 사이즈 $\geq N \times M$ 에 포함되어 있다) THEN Bs = 3,
- [0277] - ELSE IF (P 또는 Q 가 MODE_INTRA 를 갖는다) THEN Bs = 2,
- [0278] - ELSE 추가적인 Bs 유도는 HEVC 사양을 따른다.
- [0279] 이 변형예에 있어, 비디오 인코더 (20) 는 하이-레벨 구문 파라미터를 SPS, PPS 또는 최소 TB 사이즈 N x M (또는 N x N) 을 식별하는 슬라이스 헤더 (또는, 임의의 다른 헤더 또는 파라미터 세트) 로 시그널링할 수도 있다.
- [0280] 다음은 추가적인 Bs 값들에 대한 디블록킹 필터의 의존성을 규정한다. 일반적으로, 비디오 코더는 HEVC 디블록킹 필터의 현재의 동작과 비교하여, Bs 가 2 의 값보다 더 크면, 디블록킹 필터 강도를 증가시킨다. 본 개시물에서 설명하는 기법들의 이 양태에 따르면, 비디오 코더가 Bs 를 값 3 또는 4 과 동일하게 설정하면, 비디오 코더는 약한 필터 강도를 증가시키기 위해 HEVC 에 규정된 "약한" 또는 "정상" 디블록킹 필터의 클리핑 동작을 수정한다. 다음 의사-코드는 "약한 (weak)" 필터 동작 및 Bs 값의 영향을 예시한다 (여기서, p0, q0 는 디블록킹 필터링되는 에지 경계에서의 샘플들을 나타낼 수도 있으며, 여기서, p1 및 q1 는 p0, q0 에 인접한 샘플들을 나타낼 수도 있다):
- [0281] - HEVC 약한 필터 델타:
- [0282] $\Delta = (9(q_0 - p_0) - 3(q_1 - p_1) + 8) / 16$
- [0283] - IF ($\Delta < 10t_c$) 주: 대안 바람직한 임계값은 $6t_c$ 이다
- [0284] $\delta = \text{Clip3}(-c_1, c_1, \Delta)$ 주: $\text{Clip3}(a, b, x)$ 클립들 값 x to [a, b] 간격
- [0285] $p_0' = \text{Clip}(p_0 + \delta)$ 주: 클립(x) 클립들 값 x to 입력 샘플 비트 심도
- [0286] $q_0' = \text{Clip}(q_0 - \delta)$
- [0287] p_1 을 수정하면:
- [0288] $\Delta p = \text{Clip3}(-c_2, c_2, ((p_2 + p_0 + 1)/2 - p_1 + \delta) / 2)$
- [0289] $p_1' = \text{Clip}(p_1 + \Delta p)$
- [0290] q_1 을 수정하면:
- [0291] $\Delta q = \text{Clip3}(-c_2, c_2, ((q_2 + q_0 + 1)/2 - q_1 - \delta) / 2)$
- [0292] $q_1' = \text{Clip}(q_1 + \Delta q)$
- [0293] - 디폴트 필터 클리핑 임계치들 [6]

- [0294] • $c1 = t_c$
- [0295] • $c2 = t_c / 2$ (주: 대안적인 바람직한 임계값은 $t_c / 4$ 이다)
- [0296] - 필터 클리핑 임계치들 $c1$ 및 $c2$ 은 B_s 값 및 최대 TU 또는 TB 사이즈 (Log2MaxTUsiz) 에 따라서 다음과 같이 수정된다.
- [0297] • If ($B_s > 2 \ \&\& \ \text{Log2MaxTUsiz} > 2$)
- [0298] $c1 = 4t_c$
- [0299] $c2 = 2t_c$ (주: 대안 임계값들이 가능하다)
- [0300] 일부의 경우, "약한" 필터 클리핑 임계치들은 B_s 값에 의존하며 또한 최대 TU 사이즈 (Log2MaxTUsiz) 에 의존한다. 후자 조건의 목적은 최대 TU 사이즈가 작은 (예컨대, 4×4) 경우들에서 클리핑 범위가 증가하는 것을 방지하는 것일 수도 있다. 이것은 따라서, 하한을 설정할 수도 있다. 일부의 경우, 위에서 규정된 후자의 조건이 체크되지 않는다. 다른 경우, 하한은 하이-레벨 구문에서의 파라미터로서 시그널링될 수도 있다.
- [0301] 이러한 방법으로, 비디오 코딩 디바이스는 경계 강도 값에 기초하여, 하나 이상의 디블록킹 필터 클리핑 임계치들을 결정하고, 그 결정된 하나 이상의 디블록킹 필터 클리핑 임계치들을 이용하여, 디블록킹 필터에 대해 클리핑 동작을 수행하여 클리핑된 디블록킹 필터를 결정하고, 그리고 그 클리핑된 디블록킹 필터를 비디오 데이터의 블록에 적용하도록 구성될 수도 있다.
- [0302] 일부의 경우, 비디오 코딩 디바이스는 비디오 데이터의 블록 및 비디오 데이터의 블록에 인접한 블록 중 하나 이상이 규정된 사이즈의 비디오 데이터의 블록에 포함되는지 여부에 기초하여, 위에서 설명한 방법으로 경계 강도 값을 결정하도록 구성될 수도 있다.
- [0303] 일부의 경우, 비디오 코딩 디바이스는 비디오 데이터의 블록 및 비디오 데이터의 블록에 인접한 블록 중 하나 이상이 규정된 사이즈의 변환 유닛에 포함되는지 여부에 기초하여, 다시 위에서 설명한 바와 같이, 경계 강도 값을 결정하도록 구성될 수도 있다.
- [0304] 비디오 코딩 디바이스는 또한, 일부 경우, 비디오 데이터의 블록을 포함하는 비디오 데이터의 부분에 대해 결정되는 블록의 최대 사이즈 및 경계 강도 값에 기초하여, 하나 이상의 디블록킹 필터 클리핑 임계치들을 결정하도록 구성될 수도 있다.
- [0305] 일부의 경우, 비디오 코딩 디바이스는 비디오 인코더 (20) 와 같은, 비디오 인코딩 디바이스를 포함한다. 비디오 데이터의 블록은 이 경우, 비디오 데이터의 블록의 인코딩된 버전으로부터 비디오 인코딩 디바이스에 의해 재구성되는 비디오 데이터의 재구성된 블록을 포함할 수도 있다. 비디오 인코더 (20) 는 클리핑된 디블록킹 필터를 적용할 때, 클리핑된 디블록킹 필터를 비디오 데이터의 재구성된 블록에 적용하여, 비디오 데이터의 디블록킹 필터링된 재구성된 블록을 발생하도록 구성될 수도 있다. 비디오 인코더 (20) 는, 본 개시물에서 자세히 설명하는 바와 같이, 또한 비디오 데이터의 다른 블록들을 인코딩할 때 참조 블록으로서 사용을 위해 비디오 데이터의 디블록킹 필터링된 재구성된 블록을 메모리에 저장하도록 구성될 수도 있다.
- [0306] 다른 경우, 비디오 코딩 디바이스는 비디오 디코더 (30) 와 같은 비디오 디코딩 디바이스를 포함할 수도 있다. 이 경우, 비디오 데이터의 블록은 비디오 데이터의 블록의 인코딩된 버전으로부터 비디오 디코딩 디바이스에 의해 재구성되는 비디오 데이터의 재구성된 블록을 포함할 수도 있다. 비디오 디코더 (30) 는 클리핑된 디블록킹 필터를 적용할 때, 클리핑된 디블록킹 필터를 비디오 데이터의 재구성된 블록에 적용하여, 비디오 데이터의 디블록킹 필터링된 재구성된 블록을 발생하고, 그리고 비디오 데이터의 디블록킹 필터링된 재구성된 블록을 메모리에 비디오 데이터의 다른 블록들을 디코딩할 때 참조 블록으로서 사용을 위해 저장하도록 구성될 수도 있다.
- [0307] 게다가, 다음 디블록킹 필터 파라미터들 t_c 및 베타는 다음과 같이 결정될 수도 있다:
- [0308] - $\text{TC_OFFSET} = 2 * t_{c_offset_div2}$
- [0309] - $\text{BETA_OFFSET} = 2 * \text{beta_offset_div2}$
- [0310] - IF ($B_s = 2$ 또는 $B_s = 4$) THEN $\text{TcOffset} = \text{DEFAULT_INTRA_TC_OFFSET}(=2) + \text{TC_OFFSET}$; ELSE $\text{TcOffset} =$

TC_OFFSET (주: Bs = 2 또는 4 이면 P 또는 Q 는 인트라 코딩된다)

- BetaOffset = BETA_OFFSET

- t_c : $Q = \text{Clip3}(0, \text{MAX_QP} + \text{DEFAULT_INTRA_TC_OFFSET}, \text{QP} + \text{TcOffset})$ (루마 및 크로마 디블록킹)

- β : $Q = \text{Clip3}(0, \text{MAX_QP}, \text{QP} + \text{BetaOffset})$ (루마 디블록킹 단독)

크로마 디블록킹 필터가 Bs = 2 에 대해 수행되는 (P 또는 Q 가 인트라-코딩되는) HEVC 에 비해, 기법들의 전술한 양태들은 Bs = 2 또는 Bs = 4 일 때 크로마 디블록킹 필터링을 수행한다.

일부의 경우, 비디오 인코더 (20) 는 Bs 값들 3 및/또는 4 에 대해 디블록킹 강도를 조정하기 위해 디블록킹 제어 파라미터들, 즉 beta_offset_max_tu_div2 및 tc_offset_max_tu_div2 파라미터들을 위에서 설명한 방법과 유사한 방법으로 시그널링할 수도 있다. 다음 예시적인 의사-코드는 이 동작을 예시할 수도 있다:

- IF (Bs = 3 또는 Bs = 4) THEN TC_OFFSET = 2*tc_offset_max_tu_div2, ELSE TC_OFFSET = 2*tc_offset_div2

- IF (Bs = 3 또는 Bs = 4) THEN BETA_OFFSET = 2*beta_offset_max_tu_div2, ELSE BETA_OFFSET = 2*beta_offset_div2

- IF (Bs = 2 또는 Bs = 4) THEN TcOffset = DEFAULT_INTRA_TC_OFFSET(=2) + TC_OFFSET; ELSE TcOffset = TC_OFFSET (주: Bs = 2 또는 4 이면 P 또는 Q 는 인트라 코딩된다)

- BetaOffset = BETA_OFFSET

- t_c : $Q = \text{Clip3}(0, \text{MAX_QP} + \text{DEFAULT_INTRA_TC_OFFSET}, \text{QP} + \text{TcOffset})$ (루마 및 크로마 디블록킹)

- β : $Q = \text{Clip3}(0, \text{MAX_QP}, \text{QP} + \text{BetaOffset})$ (루마 디블록킹 단독)

이러한 방법으로, 비디오 코딩 디바이스는 비디오 데이터의 블록을 포함하는 비디오 데이터의 부분에 대해 결정된 변환 유닛의 최대 사이즈 및 경계 강도 값에 기초하여, 하나 이상의 디블록킹 필터 클리핑 임계치들을 결정하도록 추가로 구성될 수도 있다. 일부의 경우, 비디오 코딩 디바이스는 경계 강도 값이 2 보다 더 크고 변환 유닛의 최대 사이즈가 제 1 임계치 (예컨대, 2 의 값) 보다 더 클 때, 제 1 디블록킹 필터 클리핑 임계치를 비디오 데이터의 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 오프셋의 배수로서 결정하도록 추가로 구성될 수도 있다. 비디오 코딩 디바이스는 경계 강도 값이 2 보다 더 크고 변환 유닛의 최대 사이즈가 제 2 임계치 (제 1 임계치와 동일할 수도 있음, 예컨대, 2 의 값) 보다 더 클 때, 제 2 디블록킹 필터 클리핑 임계치를, 비디오 데이터의 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 오프셋의 배수로서 결정하도록 구성될 수도 있다. 비디오 코딩 디바이스는, 이 경우, 클리핑 동작을 수행할 때, 그 결정된 제 1 및 제 2 디블록킹 필터 클리핑 임계치들을 이용하여 디블록킹 필터에 대해 클리핑 동작을 수행하여, 클리핑된 디블록킹 필터를 결정하도록 구성될 수도 있다.

비디오 코딩 디바이스는 클리핑 동작을 수행할 때, 그 결정된 하나 이상의 디블록킹 필터 클리핑 임계치들을 이용하여 정상 디블록킹 필터에 대해 클리핑 동작을 수행하여, 클리핑된 정상 디블록킹 필터를 결정하도록 추가로 구성될 수도 있다. 더욱이, 비디오 코딩 디바이스는 클리핑된 디블록킹 필터를 적용할 때, 클리핑된 정상 디블록킹 필터를 비디오 데이터의 블록에 적용하도록 추가로 구성될 수도 있다.

비디오 코딩 디바이스는 비디오 데이터의 블록 및 비디오 데이터의 블록에 인접한 블록 중 하나 이상이 규정된 사이즈의 변환에 포함되는지 여부에 기초하여, 경계 강도 값을 결정하고, 그리고, 그 결정된 경계 강도 값이 4 와 동일할 때 크로마 디블록킹 필터를 비디오 데이터의 블록의 크로마 부분에 적용하기로 결정하도록 추가로 구성될 수도 있다.

비디오 코딩 디바이스는 비디오 데이터의 블록 및 비디오 데이터의 블록에 인접한 블록 중 하나 이상이 규정된 사이즈의 변환 유닛에 포함되는지 여부에 기초하여, 경계 강도 값을 결정하고, 비디오 데이터의 블록에 적용되는 변환의 사이즈가 최대 변환 유닛 사이즈인지 여부를 결정할 수도 있다. 비디오 데이터의 블록에 적용되는 변환의 사이즈가 최대 변환 유닛 사이즈라고 결정하자 마자, 비디오 코딩 디바이스는 비디오 데이터의 블록에 적용되는 변환의 사이즈가 최대 변환 유닛 사이즈가 아닐 때, 비디오 데이터의 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 그렇지 않으면 사용되는 t_c 오프셋 및 β 오프셋 대신 사용되는 최대 t_c 오프셋과 최

대 β 오프셋 중 하나 이상을 결정하도록 구성될 수도 있다.

- [0326] 비디오 코딩 디바이스는 t_c 오프셋과 β 오프셋 중 하나 이상이 위에서 설명한 방법으로 적어도 하나의 구문 엘리먼트로서 PPS 및 비디오 데이터의 블록을 포함하는 독립적으로 디코딩가능한 유닛의 헤더 중 하나 또는 양쪽에 규정되는지 여부를 나타내는 플래그를 화상 파라미터 세트 (PPS) 에 규정하도록 추가로 구성될 수도 있다. 플래그는 t_c 오프셋과 β 오프셋 중 하나 이상이 적어도 하나의 구문 엘리먼트로서 PPS 및 비디오 데이터의 블록의 인코딩된 버전을 포함하는 독립적으로 디코딩가능한 유닛의 헤더 중 하나 또는 양쪽에 규정되는지 여부를 나타내는, 위에서 설명되는 `de-blocking_filter_control_present_flag` 을 포함할 수도 있다.
- [0327] 도 3 은 인코딩된 비디오 시퀀스를 디코딩하는 비디오 디코더 (30) 의 일 예를 예시하는 블록도이다. 도 3 의 예에서, 비디오 디코더 (30) 는 엔트로피 디코딩 유닛 (70), 모션 보상 유닛 (72), 인트라-예측 유닛 (74), 역양자화 유닛 (76), 역변환 유닛 (78), 디코딩된 화상 버퍼 (82) 및 합산기 (80) 를 포함한다. 비디오 디코더 (30) 는 일부 예들에서, 비디오 인코더 (20) (도 2 참조) 에 대해서 설명한 인코딩 과정과는 일반적으로 반대인 디코딩 과정을 수행할 수도 있다.
- [0328] 엔트로피 디코딩 유닛 (70) 은 인코딩된 비트스트림 상에 엔트로피 디코딩 프로세스를 수행하여, 변환 계수들의 1차원 어레이를 추출한다. 사용되는 엔트로피 디코딩 프로세스는 비디오 인코더 (20) 에 의해 사용되는 엔트로피 코딩 (예컨대, CABAC, CAVLC 등) 에 의존한다. 인코더에 의해 사용되는 엔트로피 코딩 프로세스는 인코딩된 비트스트림으로 시그널링될 수도 있거나 또는 미리 결정된 프로세스일 수도 있다. 엔트로피 디코딩 유닛 (70) 은 본 개시물에서 설명하는 기법들에 따라서 비트스트림으로부터 위에서 언급한 구문 엘리먼트들을 파싱할 수도 있다.
- [0329] 일부 예들에서, 엔트로피 디코딩 유닛 (70) 은 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 위에서 언급한 오프셋들 중 적어도 하나가 인코딩된 비디오 데이터의 블록을 포함한 슬라이스의 헤더에 규정되어 있는지를 나타내는, 인코딩된 비디오 데이터를 나타내는 비트스트림의 화상 파라미터 세트 (PPS) 에 규정되는 플래그를 결정한다. 적어도 하나의 오프셋은, 위에서 언급한 바와 같이, 위에서 설명한 방법으로, 인코딩된 비디오 데이터의 블록을 인코딩하는데 적용되는 변환의 사이즈와 연관될 수도 있다. 이 플래그에 기초하여, 엔트로피 디코딩 유닛 (70) 은 슬라이스의 헤더로부터 적어도 하나의 오프셋을 추출할 수도 있다. 이 플래그는, 위에서 언급한 바와 같이, `de-blocking_filter_control_present_flag` 를 지칭할 수도 있다.
- [0330] 다른 예들에서, 엔트로피 디코딩 유닛 (70) 은 인코딩된 비디오 데이터를 나타내는 비트스트림 내에 포함된 APS 또는 PPS 로부터, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 위에서 언급된 오프셋들을 추출한다. 또, 하나 이상의 오프셋들은 인코딩된 비디오 데이터의 블록을 인코딩하는데 적용되는 변환의 사이즈와 연관될 수도 있다. 플래그를 PPS 로 시그널링하는 것 및 APS 또는 PPS 로 오프셋들의 시그널링에 관련되는 기법들의 2개의 양태들은 함께 또는 별개로 이용될 수도 있다. 함께 이용될 때, 플래그는, 제로로 설정될 때, 오프셋들이 APS 또는 PPS 로 시그널링된다는 것을 나타낼 수도 있으며, 1 로 설정될 때, 오프셋들이 슬라이스 헤더로 시그널링된다는 것을 나타낼 수도 있다. 어느 경우라도, 엔트로피 디코딩 유닛 (70) 은, 비트스트림으로부터 오프셋들을 추출하고 이들 오프셋들을 (양쪽이 실질적으로 동일한 또는 유사한 디블록킹 프로세스들을 수행하여 비디오 데이터의 재구성된 블록들을 디블록킹할 수도 있다는 점에서, 도 2 의 예에 나타난 디블록킹 유닛 (63) 과 실질적으로 유사할 수도 있는) 도 3 의 예에 나타난 디블록킹 유닛 (63) 에 제공할 수도 있다.
- [0331] 일부 예들에서, 엔트로피 디코딩 유닛 (70) (또는, 역양자화 유닛 (76)) 은 비디오 인코더 (20) 의 엔트로피 인코딩 유닛 (56) (또는, 양자화 유닛 (54)) 에 의해 사용되는 스캐닝 모드를 미러링하는 스캐닝을 이용하여, 수신된 값들을 스캐닝할 수도 있다. 계수들의 스캐닝이 역양자화 유닛 (76) 에서 수행될 수도 있지만, 스캐닝은 예시의 목적을 위해 엔트로피 디코딩 유닛 (70) 에 의해 수행되는 것으로 설명될 것이다. 게다가, 예시의 용이를 위해 별개의 기능적 유닛들로서 나타내지만, 엔트로피 디코딩 유닛 (70), 역양자화 유닛 (76), 및 비디오 디코더 (30) 의 다른 유닛들의 구조 및 기능은 서로 고도로 통합될 수도 있다.
- [0332] 역양자화 유닛 (76) 은 비트스트림으로 제공되어 엔트로피 디코딩 유닛 (70) 에 의해 디코딩되는 양자화된 변환 계수들을 역양자화한다, 즉 양자화 해제한다. 역양자화 프로세스는 종래의 프로세스, 예컨대, HEVC 에 대해 제안되거나 또는 H.264 디코딩 표준에 의해 정의된 프로세스들과 유사한, 프로세스를 포함할 수도 있다. 역양자화 프로세스는 양자화의 정도를 결정하기 위해, 그리고, 이와 유사하게, 적용되어야 하는 역양자화의 정도

를 결정하기 위해, CU 에 대해 비디오 인코더 (20) 에 의해 계산된 양자화 파라미터 QP 의 사용을 포함할 수도 있다. 양자화 파라미터는 위에서 언급한 바와 같이, 잔여 비디오 데이터로부터 변환 계수들을 발생하는데 적용된 변환의 사이즈에 기초하여 변환될 수도 있다. 역양자화 유닛 (76) 은 계수들이 1차원 어레이로부터 2차원 어레이로 변환되기 전후에 변환 계수들을 역양자화할 수도 있다.

[0333] 역변환 모듈 (78) 은 역 양자화된 변환 계수들에 역변환을 적용한다. 일부 예들에서, 역변환 모듈 (78) 은 비디오 인코더 (20) 로부터 시그널링하는 것에 기초하여, 또는 블록 사이즈, 코딩 모드, 또는 기타 등등과 같은 하나 이상의 코딩 특성들로부터 변환을 추론함으로써, 역변환을 결정할 수도 있다. 일부 예들에서, 역변환 모듈 (78) 은 현재의 블록을 포함하는 LCU 에 대한 쿼드트리의 루트 노드에서의 시그널링된 변환에 기초하여 현재의 블록에 적용할 변환을 결정할 수도 있다. 이의 대안으로, 변환은 LCU 쿼드트리에서 잎-노드 CU 에 대한 TU 쿼드트리의 루트에서 시그널링될 수도 있다. 일부 예들에서, 역변환 모듈 (78) 은 캐스케이드된 역변환을 적용할 수도 있으며, 여기서, 역변환 모듈 (78) 은 2개 이상의 역변환들을 디코딩되는 현재의 블록의 변환 계수들에 적용한다. 역변환 모듈 (78) 은 잔여 비디오 데이터의 블록을 발생할 수도 있다.

[0334] 인트라-예측 유닛 (74) 은 시그널링된 인트라-예측 모드 및 현재의 프레임의 이전에 디코딩된 블록들로부터의 데이터에 기초하여, 현재의 프레임의 현재의 블록에 대한 예측 데이터를 발생할 수도 있다.

[0335] 취출된 모션 예측 방향, 참조 프레임 인덱스, 및 계산된 현재의 모션 벡터 (예컨대, 이웃하는 블록으로부터 병합 모드에 따라 복사된 모션 벡터) 에 기초하여, 모션 보상 유닛 (72) 은 현재의 부분에 대해 모션 보상된 블록을 발생한다. 이들 모션 보상된 블록들은 본질적으로 잔여 데이터의 블록을 발생하는데 사용되는 예측 블록을 재생성한다.

[0336] 모션 보상 유닛 (72) 은 모션 보상되는 블록들을 발생하여, 어쩌면 내삽 필터들에 기초하여 내삽을 수행할 수도 있다. 서브-픽셀 정밀도를 갖는 모션 추정에 이용되는 내삽 필터들에 대한 식별자들이 구문 엘리먼트들에 포함될 수도 있다. 모션 보상 유닛 (72) 은 비디오 블록의 인코딩 동안 비디오 인코더 (20) 에 의해 사용된 것과 같은 내삽 필터들을 이용하여, 참조 블록의 서브-정수 픽셀들에 대한 내삽된 값들을 계산할 수도 있다. 모션 보상 유닛 (72) 은 수신된 구문 정보에 따라서 비디오 인코더 (20) 에 의해 이용된 내삽 필터들을 결정하고 그 내삽 필터들을 이용하여, 예측 블록들을 발생할 수도 있다.

[0337] 게다가, 모션 보상 유닛 (72) 및 인트라-예측 유닛 (74) 은, HEVC 예에서, (예컨대, 쿼드트리에 의해 제공되는) 구문 정보의 일부를 이용하여, 인코딩된 비디오 시퀀스의 프레임(들) 을 인코딩하는데 사용되는 LCU들의 사이즈들을 결정할 수도 있다. 모션 보상 유닛 (72) 및 인트라-예측 유닛 (74) 은 또한 구문 정보를 이용하여, 인코딩된 비디오 시퀀스의 프레임의 각각의 CU 가 분할되는지 (그리고, 유사하게, 서브-CU들이 분할되는지) 를 기술하는 분할 정보를 결정할 수도 있다. 구문 정보는 또한 각각의 분할이 어떻게 인코딩되는지를 나타내는 모드들 (예컨대, 인트라- 또는 인터-예측, 그리고 인트라-예측에 있어 인트라-예측 인코딩 모드), 각각의 인터-인코딩된 PU 에 대한 하나 이상의 참조 프레임들 (및/또는 참조 프레임들에 대한 식별자들을 포함하는 참조 리스트들), 및 인코딩된 비디오 시퀀스를 디코딩하기 위한 다른 정보를 포함할 수도 있다.

[0338] 합산기 (80) 는 잔여 블록들을 모션 보상 유닛 (72) 또는 인트라-예측 유닛 (74) 에 의해 발생된 대응하는 예측 블록들과 합산하여, 디코딩된 블록들을 형성한다. 비디오 디코더 (30) 는 또한 디블록킹 필터를 적용하여 본 개시물에서 위에서 설명한 본 기법들의 여러 양태들에 따라서 디블록킹 필터링을 수행할 수도 있는 디블록킹 유닛 (81) 을 포함할 수도 있다. 비디오 디코더 (30) 의 디블록킹 유닛 (81) 은 비디오 디코더 (30) 의 디블록킹 유닛 (81) 이 비디오 인코더 (20) 의 디블록킹 유닛 (63) 에 대해서 위에서 설명한 방법과 유사한 방법으로 동작할 수도 있다는 점에서, 비디오 인코더 (20) 의 디블록킹 유닛 (63) 과 유사하고, 아니면 실질적으로 유사할 수도 있다. 어느 경우라도, 이 필터는 블로키니스 아티팩트들을 제거하기 위해 디코딩된 블록들을 필터링하는데 적용될 수도 있다. 디코딩된 비디오 블록들은 그 후 디코딩된 화상 버퍼 (82) 에 저장되며, 그 디코딩된 화상 버퍼는 후속 모션 보상을 위해 참조 블록들을 제공하며 또한 (도 1 의 디스플레이 디바이스 (32) 와 같은) 디스플레이 디바이스 상에의 프리젠테이션을 위해 디코딩된 비디오를 발생한다.

[0339] 도 4 는 디블록킹 필터링을 제어하는데 사용되는 오프셋들이 슬라이스 헤더에 포함되는지 여부를 나타내는 플래그를 비트스트림의 화상 파라미터 세트 (PPS) 에 규정할 때에, 도 2 의 예에 나타난 비디오 인코더 (20) 와 같은, 비디오 인코더의 예시적인 동작을 예시하는 플로우차트이다. 먼저, 비디오 인코더 (20) 는 비디오 데이터의 블록을 수신할 수도 있다. 비디오 인코더 (20) 는 이 비디오 데이터의 블록을 인코딩하기 위해, 인터-또는 인트라-코딩을 수행할 수도 있으며, 여기서 모드 선택 유닛 (40) 은 위에서 설명한 방법으로 비디오 데이터의 블록을 블록의 분석을 통해서 인터- 또는 인트라-코딩할지 여부를 결정할 수도 있다. 도 4 에 제시된

예는 모드 선택 유닛 (40) 이 예시 목적들을 위해 비디오 데이터의 현재의 블록의 인터-코딩을 선택하였다고 가정한다. 기법들은 이 예에 한정되지 않아야 한다. 어느 경우라도, 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 모션 추정 및 보상을 수행하여 비디오 데이터의 현재의 블록에 대해 예측 블록을 발생한다 (90). 합산기 (50) 는 비디오 데이터의 현재의 블록으로부터 예측 블록을 감산하여, 비디오 데이터의 잔여 블록 또는 잔여 블록을 발생한다 (92).

[0340] 변환 모듈 (52) 은 그 후 하나 이상의 변환들을 비디오 데이터의 잔여 블록에 적용하여 위에서 설명한 방법으로 변환 계수들의 블록을 발생할 수도 있다 (94). 양자화 유닛 (54) 은 그 후 변환 계수들의 블록을 양자화하여, 다시 위에서 설명한 방법으로, 변환 계수들의 양자화된 블록을 발생할 수도 있다 (96). 위에서 언급한 바와 같이, 양자화 유닛 (54) 은 적용된 변환의 사이즈에 기초하여 변하는 양자화 파라미터를 적용하여, 변환 계수들의 블록을 양자화할 수도 있다. 어느 경우라도, 양자화를 수행한 후, 엔트로피 인코딩 유닛 (56) 은 변환 계수들의 블록을 엔트로피 인코딩하여, 변환 계수들의 엔트로피 인코딩된 블록을 인코딩된 비디오 데이터를 나타내는 비트스트림에 저장할 수도 있다 (98).

[0341] 비디오 인코더 (20) 는 그 후 위에서 설명한 방법으로 변환 계수들의 양자화된 블록으로부터 비디오 데이터의 블록을 재구성할 수도 있다 (100). 디블록킹 유닛 (63) 은 그 후 비디오 데이터의 재구성된 블록을 필터링하기 위해 디블록킹 필터링을 수행할지 여부를 결정할 수도 있다. 디블록킹 유닛 (63) 은 그 적용된 변환의 사이즈에 기초하여, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정할 수도 있다 (102). 디블록킹 유닛 (63) 은 그 후 그 결정된 적어도 하나의 오프셋에 기초하여, 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행할 수도 있다 (104).

[0342] 엔트로피 인코딩 유닛 (56) 은 그 후 적어도 하나의 오프셋이 비디오 데이터의 블록의 인코딩된 버전을 포함하는 독립적으로 디코딩가능한 유닛 (예컨대, 슬라이스) 의 헤더에 적어도 하나의 구문 엘리먼트로서 규정되는지 여부를 나타내는 플래그를 화상 파라미터 세트 (PPS) 에 규정할 수도 있다 (106). 또, 일부 경우, 이 플래그는 구문 테이블들에 "de-blocking_filter_control_present_flag" 구문 엘리먼트로서 나타내어질 수도 있다. 엔트로피 인코딩 유닛 (56) 은 현재의 슬라이스 헤더 및 이전 또는 후속 슬라이스들의 헤더의 분석에 기초하여, 이 플래그에 대한 값을 결정하고, 이들 다른 슬라이스 헤더들에서의 유사한 오프셋들의 발생 및 값들을 비교하여, 그 오프셋들이 APS, PPS 와 같은 다른 파라미터 세트들에 또는 각각의 슬라이스 헤더에 별개로 규정될 수도 있는지 여부를 결정할 수도 있다. 엔트로피 인코딩 유닛 (56) 은, 일 예로서, 슬라이스 헤더들에서의 이들 오프셋들의 변화율을 결정하고, 이 결정된 변화율을 임계치 변화율과 비교하고, 그리고 그 비교에 기초하여 오프셋들이 2개 이상의 슬라이스들에 대해 일괄적으로 규정될 수 있는지 여부를 결정할 수도 있다. 이 예와 관련하여 설명하지만, 이 기법들은 비디오 인코더가 구문 엘리먼트들을 SPS, PPS, APS 또는 다른 파라미터 세트 또는 헤더가 아니라, 슬라이스 헤더에 규정할지 여부를 결정할 수도 있는 임의의 방법에 대해서 구현될 수도 있다.

[0343] 어느 경우라도, 이 플래그가 제로로 설정되거나 또는 동일하면 ("예" 108), 엔트로피 인코딩 유닛 (56) 은 오프셋들을 PPS 또는 슬라이스 헤더에 규정하지 않으며 프로세스는 또 다른 블록에 대해 반복될 수도 있다 (현재의 블록이 비디오 데이터의 최종 블록이 아니면; 90 내지 110). 그렇지 않으면, 플래그가 제로와 동일하지 않다는 것을 나타내는 1 로 이 플래그가 설정되면 ("아니오" 108), 엔트로피 인코딩 유닛 (56) 은 위에서 언급한 오프셋들 중 하나 이상을 PPS 및/또는 비디오 데이터의 현재의 블록의 인코딩된 버전과 연관된 슬라이스 헤더에 규정할 수도 있다 (110). 또, 이것이, 인코딩되는 비디오 데이터의 최종 블록이 아닌 한, 비디오 인코더 (20) 는 이 프로세스를 계속해서 수행할 수도 있다 (90 내지 110).

[0344] 도 5 는 디블록킹 필터링을 제어하는데 사용되는 오프셋들을 적응 파라미터 세트 (APS) 또는 화상 파라미터 세트 (PPS) 에 규정할 때에, 도 2 의 예에 나타난 비디오 인코더 (20) 와 같은, 비디오 인코더의 예시적인 동작을 예시하는 플로우차트이다. 먼저, 비디오 인코더 (20) 는 비디오 데이터의 블록을 수신할 수도 있다. 비디오 인코더 (20) 는 이 비디오 데이터의 블록을 인코딩하기 위해, 인터- 또는 인트라-코딩을 수행할 수도 있으며, 여기서 모드 선택 유닛 (40) 은 위에서 설명한 방법으로 비디오 데이터의 블록을 블록의 분석을 통해서 인터- 또는 인트라-코딩할지 여부를 결정할 수도 있다. 도 4 에 제시된 예는 모드 선택 유닛 (40) 이 예시 목적들을 위해 비디오 데이터의 현재의 블록의 인터-코딩을 선택하였다고 가정한다. 기법들은 이 예에 한정되지 않아야 한다. 어느 경우라도, 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 모션 추정 및 보상을 수행하여, 비디오 데이터의 현재의 블록에 대해 비디오 데이터의 예측 블록을 발생한다 (120). 합산기 (50) 는 비디오 데이터의 현재의 블록으로부터 예측 블록을 감산하여, 비디오 데이터의 잔여 블록 또는 잔여 블록을

결정한다 (122).

[0345] 변환 모듈 (52) 은 그 후 하나 이상의 변환들을 비디오 데이터의 잔여 블록에 적용하여, 위에서 설명한 방법으로 변환 계수들의 블록을 발생할 수도 있다 (124). 양자화 유닛 (54) 은 그 후 변환 계수들의 블록을 양자화하여, 다시 위에서 설명한 방법으로, 변환 계수들의 양자화된 블록을 발생할 수도 있다 (126). 위에서 언급한 바와 같이, 양자화 유닛 (54) 은 적용된 변환의 사이즈에 기초하여 변환하는 양자화 파라미터를 적용하여, 변환 계수들의 블록을 양자화할 수도 있다. 어느 경우라도, 양자화를 수행한 후, 엔트로피 인코딩 유닛 (56) 은 변환 계수들의 블록을 엔트로피 인코딩하여, 변환 계수들의 엔트로피 인코딩된 블록을 인코딩된 비디오 데이터를 나타내는 비트스트림에 저장할 수도 있다 (128).

[0346] 비디오 인코더 (20) 는 그 후 위에서 설명한 방법으로 변환 계수들의 양자화된 블록으로부터 비디오 데이터의 블록을 재구성할 수도 있다 (130). 디블록킹 유닛 (63) 은 그 후 디블록킹 필터링을 수행할지 여부를 결정하여, 비디오 데이터의 재구성된 블록을 필터링할 수도 있다. 디블록킹 유닛 (63) 은 그 적용된 변환의 사이즈에 기초하여, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 오프셋을 결정할 수도 있다 (132). 디블록킹 유닛 (63) 은 그 후 그 결정된 적어도 하나의 오프셋에 기초하여, 비디오 데이터의 재구성된 블록에 대해 디블록킹 필터링을 수행할 수도 있다 (134).

[0347] 엔트로피 인코딩 유닛 (56) 은 그 후 위에서 설명한 방법으로 디블록킹 필터링을 제어하는데 사용되는 오프셋들을 PPS 및/또는 슬라이스 헤더에 규정할 수도 있다 (136). 엔트로피 인코딩 유닛 (56) 은 현재의 슬라이스 헤더 및 이전 또는 후속 슬라이스들의 헤더의 분석을 수행하여, 이들 다른 슬라이스 헤더들에서 유사한 오프셋들의 발생 및 값들을 비교하고, 오프셋들이 APS 또는 PPS 와 같은 다른 파라미터 세트들에, 또는 각각의 슬라이스 헤더에, 별개로 규정될 수 있는지 여부를 결정할 수도 있다. 엔트로피 인코딩 유닛 (56) 은, 일 예로서, 슬라이스 헤더들에서의 이들 오프셋들의 변화율을 결정하고, 이 결정된 변화율을 임계치 변화율과 비교하고, 그리고 그 비교에 기초하여, 오프셋들이 2개 이상의 슬라이스들에 대해 일괄적으로 규정될 수 있는지 여부를 결정할 수도 있다. 이 예와 관련하여 설명하지만, 이 기법들은, 비디오 인코더가 구문 엘리먼트들을 SPS, PPS, APS 또는 다른 파라미터 세트 또는 헤더가 아니라, 슬라이스 헤더에 규정할지 여부를 결정할 수도 있는 임의의 방법에 대해서 구현될 수도 있다. 위에서 언급한 바와 같이, 이것이, 인코딩되는 비디오 데이터의 최종 블록이 아닌 한, 비디오 인코더 (20) 는 이 프로세스를 계속해서 수행할 수도 있다 (120 내지 136).

[0348] 도 6 은 디블록킹 필터링을 제어하는데 사용되는 오프셋들이 슬라이스 헤더에 포함되는지 여부를 나타내는 플래그를 화상 파라미터 세트 (PPS) 로부터 추출할 때에 도 3 의 예에 나타난 비디오 디코더 (30) 와 같은, 비디오 디코더의 예시적인 동작을 예시하는 플로우차트이다. 먼저, 엔트로피 디코딩 유닛 (70) 은 인코딩된 비트스트림 상에 엔트로피 디코딩 프로세스를 수행하여, 변환 계수들의 1차원 어레이를 추출한다. 엔트로피 디코딩 유닛 (70) 은 본 개시물에서 설명하는 기법들에 따라서 비트스트림으로부터 위에서 언급한 구문 엘리먼트들을 파싱할 수도 있다.

[0349] 일부 예들에서, 엔트로피 디코딩 유닛 (70) 은 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 위에서 언급한 오프셋들 중 적어도 하나가 인코딩된 비디오 데이터의 블록을 포함하는 슬라이스의 헤더에 규정되어 있는지를 나타내는, 인코딩된 비디오 데이터를 나타내는 비트스트림의 화상 파라미터 세트 (PPS) 에 규정된 플래그를 결정하거나 또는 파싱한다 (140). 적어도 하나의 오프셋은, 위에서 언급한 바와 같이, 위에서 설명한 방법으로, 인코딩된 비디오 데이터의 블록을 인코딩하는데 적용되는 변환의 사이즈와 연관될 수도 있다. 플래그 (142) 에 기초하여, 엔트로피 디코딩 유닛 (70) 은 PPS 및 슬라이스의 헤더 중 하나 또는 양쪽으로부터 적어도 하나의 오프셋을 추출할 수도 있다. 이 플래그는, 위에서 언급한 바와 같이, de-blocking_filter_control_present_flag 를 지칭할 수도 있다. 플래그가 1 과 동일할 때 ("예" 142), 엔트로피 디코딩 유닛 (70) 은 PPS 또는 슬라이스 헤더로부터 위에서 언급한 오프셋들을 파싱하거나 또는 아니면 추출할 수도 있다 (144).

[0350] 플래그가 제로일 때 ("아니오" 142), 엔트로피 디코딩 유닛 (70) 은 기존 오프셋들을 이용할 수도 있다 (146). 또, 하나 이상의 오프셋들은 인코딩된 비디오 데이터의 블록을 인코딩하는데 적용되는 변환의 사이즈와 연관될 수도 있다. 플래그를 PPS 로 시그널링하는 것 및 APS 또는 PPS 로 오프셋들의 시그널링에 관련되는 기법들의 2개의 양태들은 함께 또는 별개로 이용될 수도 있다. 함께 이용될 때, 플래그는, 제로로 설정될 때, 오프셋들이 APS 로 시그널링된다는 것을 나타낼 수도 있으며, 1 로 설정될 때, 오프셋들이 슬라이스 헤더에 시그널링된다는 것을 나타낼 수도 있다. 어느 경우라도, 엔트로피 디코딩 유닛 (70) 은 비트스트림으로부터

오프셋들을 추출하고 이들 오프셋들을 디블록킹 유닛 (81) 에 제공할 수도 있다. 엔트로피 디코딩 유닛 (70) 은 또한 위에서 설명한 방법으로, 인코딩된 비디오 데이터의 대응하는 블록을 추출하여 인코딩된 비디오 데이터의 블록을 엔트로피 디코딩할 수도 있다 (148, 150).

[0351] 일부 예들에서, 엔트로피 디코딩 유닛 (70) (또는, 역양자화 유닛 (76)) 은 비디오 인코더 (20) 의 엔트로피 인코딩 유닛 (56) (또는, 양자화 유닛 (54)) 에 의해 사용되는 스캐닝 모드를 미리링하는 스캐닝을 이용하여, 수신된 값들을 스캐닝할 수도 있다. 역양자화 유닛 (76) 은 비트스트림으로 제공되어 엔트로피 디코딩 유닛 (70) 에 의해 디코딩되는 양자화된 변환 계수들을 역양자화한다, 즉 양자화 해제한다 (인코딩된 비디오 데이터의 엔트로피 디코딩된 블록으로서 지칭될 수도 있음; 152). 역양자화 프로세스는 양자화의 정도를 결정하기 위해, 그리고, 이와 유사하게, 적용되어야 하는 역양자화의 정도를 결정하기 위해, CU 에 대해 비디오 인코더 (20) 에 의해 계산된 양자화 파라미터 QP 의 사용을 포함할 수도 있다. 양자화 파라미터는 위에서 언급한 바와 같이, 잔여 비디오 데이터로부터 변환 계수들을 발생하는 적용된 변환의 사이즈에 기초하여 변환 수도 있다. 역양자화 유닛 (76) 은 계수들이 1차원 어레이로부터 2차원 어레이로 변환되기 전후에 변환 계수들을 역양자화할 수도 있다.

[0352] 역변환 모듈 (78) 은 인코딩된 비디오 데이터의 양자화 해제된 블록에 역변환을 적용한다 (154). 일부 예들에서, 역변환 모듈 (78) 은 비디오 인코더 (20) 로부터 시그널링하는 것에 기초하여, 또는 블록 사이즈, 코딩 모드, 또는 기타 등등과 같은 하나 이상의 코딩 특성들로부터 변환을 추론함으로써, 역변환을 결정할 수도 있다. 역변환 모듈 (78) 은 비디오 디코더 (30) 가 위에서 설명한 방법으로 인터- 또는 인트라-코딩 기법들을 통해서 비디오 데이터를 재구성하는데 이용할 수도 있는, 일반적으로 잔여 비디오 데이터로서 지칭되는 인코딩된 비디오 데이터의 역변환된 블록을 출력할 수도 있다 (156).

[0353] 디블록킹 유닛 (81) 은 본 개시물에서 위에서 설명한 본 기법들의 여러 양태들에 따라서 디블록킹 필터링을 수행하기 위해 오프셋들에 기초하여 디블록킹 필터를 적용할 수도 있다 (158). 디블록킹 유닛 (81) 은, 디블록킹 유닛 (81) 이 비디오 인코더 (20) 의 디블록킹 유닛 (63) 에 대해 위에서 설명한 방법과 유사한 방법으로 동작할 수도 있다는 점에서, 비디오 인코더 (20) 의 디블록킹 유닛 (63) 과 유사하고 그리고 아마도 실질적으로 유사하거나 또는 동일할 수도 있다. 디블록킹 유닛 (81) 은 엔트로피 디코딩 유닛 (70) 에 의해 추출된 오프셋들을 이용하여, 디블록킹 필터링을 수행할지 또는 아니면 제어할지 여부를 결정할 수도 있다. 어느 경우라도, 이 필터는 블로키니스 아티팩트들을 제거하기 위해 디코딩된 블록들을 필터링하는데 적용될 수도 있다. 디코딩된 비디오 블록들은 그 후 디코딩된 화상 버퍼 (82) 에 저장되며, 그 디코딩된 화상 버퍼는 후속 모션 보상을 위해 참조 블록들을 제공하며, 또한 (도 1 의 디스플레이 디바이스 (32) 와 같은) 디스플레이 디바이스 상에의 프리젠테이션을 위해 디코딩된 비디오를 발생한다.

[0354] 도 7 은 본 개시물에서 설명하는 기법들에 따라 적응 파라미터 세트로부터 디블록킹 필터링을 제어하는데 사용되는 오프셋들을 추출할 때에, 도 3 의 예에 나타난 비디오 디코더 (30) 와 같은, 비디오 디코더의 예시적인 동작을 예시하는 플로우차트이다. 먼저, 엔트로피 디코딩 유닛 (70) 은 인코딩된 비트스트림 상에 엔트로피 디코딩 프로세스를 수행하여, 변환 계수들의 1차원 어레이를 추출한다. 엔트로피 디코딩 유닛 (70) 은 본 개시물에서 설명하는 기법들에 따라서 비트스트림으로부터 위에서 언급한 구문 엘리먼트들을 파싱할 수도 있다.

[0355] 일부 예들에서, 엔트로피 디코딩 유닛 (70) 은 인코딩된 비디오 데이터를 나타내는 비트스트림 내에 포함된 APS 또는 PPS 로부터, 비디오 데이터의 재구성된 블록의 디블록킹 필터링을 수행할지 여부를 결정할 때에 사용되거나 또는 디블록킹 필터링 강도를 결정할 때에 사용되는 적어도 하나의 위에서 언급된 오프셋들을 추출하거나 또는 아니면 파싱한다 (160). 또, 하나 이상의 오프셋들은 인코딩된 비디오 데이터의 블록을 인코딩하는데 적용되는 변환의 사이즈와 연관될 수도 있다. 엔트로피 디코딩 유닛 (70) 은 이들 오프셋들을 디블록킹 유닛 (81) 에 제공할 수도 있다. 엔트로피 디코딩 유닛 (70) 은 또한 위에서 설명한 방법으로, 인코딩된 비디오 데이터의 대응하는 블록을 추출하고, 인코딩된 비디오 데이터의 블록을 엔트로피 디코딩할 수도 있다 (162, 164).

[0356] 일부 예들에서, 엔트로피 디코딩 유닛 (70) (또는, 역양자화 유닛 (76)) 은 비디오 인코더 (20) 의 엔트로피 인코딩 유닛 (56) (또는, 양자화 유닛 (54)) 에 의해 사용되는 스캐닝 모드를 미리링하는 스캐닝을 이용하여, 수신된 값들을 스캐닝할 수도 있다. 역양자화 유닛 (76) 은 비트스트림으로 제공되어 엔트로피 디코딩 유닛 (70) 에 의해 디코딩되는 양자화된 변환 계수들을 역양자화한다, 즉, 양자화 해제한다 (인코딩된 비디오 데이터의 엔트로피 디코딩된 블록으로서 지칭될 수도 있음; 166). 역양자화 프로세스는 양자화의 정도를 결정하기 위해, 그리고, 이와 유사하게, 적용되어야 하는 역양자화의 정도를 결정하기 위해, CU 에 대해 비디오 인코더

(20)에 의해 계산된 양자화 파라미터 QP의 사용을 포함할 수도 있다. 양자화 파라미터는 위에서 언급한 바와 같이, 잔여 비디오 데이터로부터 변환 계수들을 발생하는데 적용된 변환의 사이즈에 기초하여 변환 수도 있다. 역양자화 유닛 (76)은 계수들이 1차원 어레이로부터 2차원 어레이로 변환되기 전후에 변환 계수들을 역양자화할 수도 있다.

[0357] 역변환 모듈 (78)은 인코딩된 비디오 데이터의 양자화 해제된 블록에 역변환을 적용한다 (168). 일부 예들에서, 역변환 모듈 (78)은 비디오 인코더 (20)로부터 시그널링하는 것에 기초하여, 또는 블록 사이즈, 코딩 모드, 또는 기타 등등과 같은 하나 이상의 코딩 특성들로부터 변환을 추론함으로써, 역변환을 결정할 수도 있다. 역변환 모듈 (78)은 비디오 디코더 (30)가 위에서 설명한 방법으로 인터-또는 인트라-코딩 기법들을 통해서 비디오 데이터를 재구성하는데 이용할 수도 있는, 일반적으로 잔여 비디오 데이터로서 지칭되는 인코딩된 비디오 데이터의 역변환된 블록을 출력할 수도 있다 (170).

[0358] 디블록킹 유닛 (81)은 오프셋들에 기초하여 디블록킹 필터를 적용하여, 본 개시물에서 위에서 설명한 본 기법들의 여러 양태들에 따라서 디블록킹 필터링을 수행할 수도 있다 (172). 디블록킹 유닛 (81)은, 디블록킹 유닛 (81)이 비디오 인코더 (20)의 디블록킹 유닛 (63)에 대해서 위에서 설명한 방법과 유사한 방법으로 동작할 수도 있다는 점에서, 비디오 인코더 (20)의 디블록킹 유닛 (63)과 유사하고, 아니면 실질적으로 유사할 수도 있다. 디블록킹 유닛 (81)은 엔트로피 디코딩 유닛 (70)에 의해 추출된 오프셋들을 이용하여, 디블록킹 필터링을 수행할지 또는 아니면 제어할지 여부를 결정할 수도 있다. 어느 경우라도, 이 필터는 블로키니스 아티팩트들을 제거하기 위해 디코딩된 블록들을 필터링하는데 적용될 수도 있다. 디코딩된 비디오 블록들은 그 후 디코딩된 화상 버퍼 (82)에 저장되며, 그 디코딩된 화상 버퍼는 후속 모션 보상을 위해 참조 블록들을 제공하며, 또한 (도 1의 디스플레이 디바이스 (32)와 같은) 디스플레이 디바이스 상에의 프리젠테이션을 위해 디코딩된 비디오를 발생한다.

[0359] 도 8은 디블록킹 필터 파라미터들을 계산할 때 사용되는 경계 강도 값을 결정하는 본 기법들의 여러 양태들을 구현할 때에, 도 1 및 2의 예들에 나타난 비디오 인코더 (20) 및/또는 도 1 및 3의 예들에 나타난 비디오 디코더 (30)와 같은, 비디오 코더의 예시적인 동작을 예시하는 플로우차트이다. 이 플로우차트의 단계들은 일반적으로 상기 의사-코드를 반영하며, 이 의사-코드는 검토의 목적들을 위해서 도 8의 예에 개시된 플로우차트를 고려하여 아래에서 다시 재형성된다.

[0360] - IF (P는 MODE_INTRA를 가지며 P는 최대 TB 사이즈에 포함된다) 또는 (Q는 MODE_INTRA를 가지며 Q는 최대 TB 사이즈에 포함된다) THEN Bs = 4,

[0361] - ELSE IF (P는 최대 TB 사이즈에 포함된다) 또는 (Q는 최대 TB 사이즈에 포함된다) THEN Bs = 3,

[0362] - ELSE IF (P 또는 Q가 MODE_INTRA를 갖는다) THEN Bs = 2,

[0363] - ELSE 추가적인 Bs 유도는 HEVC 사양을 따른다.

[0364] 도 8의 예에서, 비디오 코더는 P 또는 Q 블록이 인트라-코딩된 CU에 그리고, 최대 사이즈의 TU (max TU)에 있는지 여부를 먼저 결정한다 (180). P 또는 Q 블록들 중 어느 것이 인트라-코딩된 CU에, 그리고 max TU에 있다면 ("예" 180), 비디오 코더는 경계 강도 값을 4로 설정할 수도 있다 (182). P 또는 Q 블록들 중 어느 것이 인트라-코딩된 CU에 있지 않거나 및/또는 max TU에 있지 않으면 ("아니오" 180), 비디오 코더는 P 또는 Q 블록들이 max TU에 있는지 여부를 결정할 수도 있다 (184). P 또는 Q 블록들 중 어느 것이 max TU에 있으면 ("예" 184), 비디오 코더는 Bs 값을 3으로 설정할 수도 있다 (186). 그러나, P 및 Q 블록들 양쪽이 max TU에 있지 않으면 ("아니오" 184), 비디오 코더는 P 또는 Q 블록들이 인트라-코딩된 CU에 있는지 여부를 결정할 수도 있다 (188).

[0365] P 또는 Q 블록들 중 어느 것이 인트라-코딩된 CU에 있으면 ("예" 188), 비디오 코더는 Bs 값을 2로 설정할 수도 있다 (190). 그러나, P 및 Q 블록들 양쪽이 인트라-코딩된 CU에 있지 않으면 ("아니오" 188), 비디오 코더는 그 후 변환 유닛 (TU) 에지인지 여부 및 P 또는 Q 블록들이 논-제로 변환 계수 레벨을 가진 TU에 있는지 여부를 결정할 수도 있다 (192). TU 에지이고 P 또는 Q 블록들 중 어느 것이 논-제로 변환 계수 레벨을 가진 TU에 있다고 비디오 코더가 결정하면 ("예" 192), 비디오 코더는 Bs 값을 1로 설정할 수도 있다 (193).

[0366] 그러나, TU 에지가 아니거나 및/또는 P 및 Q 블록들 양쪽이 논-제로 변환 계수 레벨을 가진 TU에 있지 않다고 비디오 코더가 결정하면 ("아니오" 192), 비디오 코더는 P 또는 Q 블록들이 상이한 참조 화상을 또는 다수의 모션 벡터들 (MVs)를 가지는지 여부를 결정할 수도 있다 (194). P 또는 Q 블록들이 상이한 참조 화상을 및/또는 다수의 MVs를 갖는다고 비디오 코더가 결정하면 ("예" 194), 비디오 디코더는 Bs를 1로 설정할 수도 있다

다 (193).

[0367] 그러나, P 또는 Q 블록들이 동일한 참조 화상들을 갖거나 (또는, 상이한 참조 화상들을 갖지 않거나) 또는 다수의 MVs 를 갖는다 (또는, 상이한 다수의 MVs 를 갖지 않는다) 고 비디오 코더가 결정하면 ("아니오" 194), 비디오 디코더는 P 및 Q 블록들이 하나의 MV 를 각각, 그리고, 1) Q 블록 (MV_{q_x}) 에 대한 모션 벡터의 x-구성요소로 감산된 P 블록 (MV_{p_x}) 에 대한 모션 벡터의 x-구성요소의 절대 값이 하나의 ($|MV_{p_x} - MV_{q_x}| \geq 1$) 보다 크거나 또는 동일하다; 또는 2) Q 블록 (MV_{q_y}) 에 대한 모션 벡터의 y-구성요소에 의해 감산된 P 블록 (MV_{p_y}) 에 대한 모션 벡터의 y-구성요소의 절대 값이 또는 하나의 ($|MV_{p_y} - MV_{q_y}| \geq 1$) 보다 크거나 또는 동일하다는 조건들을 만족하는 적어도 하나의 MV 쌍을 갖는지 여부를 결정한다 (196). 이들 조건들 중 어느 조건이 만족되면 ("예" 196), 비디오 코더는 Bs 값을 1 로 설정한다 (193).

[0368] 그러나, 이들 조건들 중 어느 것도 만족되지 않으면 ("아니오" 196), 비디오 코더는 P 및 Q 블록들이, 2개의 MVs 를 각각, 그리고 1) Q 블록 (MV_{q_x}) 에 대한 모션 벡터들 중 하나의 x-구성요소에 의해 감산된 P 블록 (MV_{p_x}) 에 대한 모션 벡터들 중 하나의 x-구성요소의 절대 값이 하나의 ($|MV_{p_x} - MV_{q_x}| \geq 1$) 보다 크거나 또는 동일하다; 또는 2) Q 블록 (MV_{q_y}) 에 대한 모션 벡터들 중 하나의 y-구성요소에 의해 감산된 P 블록 (MV_{p_y}) 에 대한 모션 벡터들 중 하나의 y-구성요소의 절대 값이 하나의 ($|MV_{p_y} - MV_{q_y}| \geq 1$) 보다 크거나 또는 동일하다는 조건을 만족하는 적어도 하나의 MV 쌍을 갖는지 여부를 결정한다 (198). 이들 조건들 중 어느 조건이 만족되면 ("예" 198), 비디오 코더는 Bs 값을 1 로 설정한다 (193). 그러나, 이들 조건들 중 어느 조건도 만족되지 않으면 ("아니오" 198), 비디오 코더는 Bs 값을 제로로 설정한다 (200). 위에서 설명한 바와 같이, 비디오 코더는 디블록킹 필터 파라미터들을 계산할 때 Bs 값을 이용할 수도 있다.

[0369] 여러 예들이 위에서 설명되었지만, 기법들은 본 개시물에서 설명하는 기법들의 여러 양태들의 특징의 조합들에 한정되지 않아야 한다. 대신, 본 기법들의 여러 양태들 또는 엘리먼트들은 임의의 실행가능한 조합으로 구현될 수도 있다.

[0370] 하나 이상의 예들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현되는 경우, 그 기능들은 하나 이상의 명령들 또는 코드들로서, 컴퓨터 판독가능 매체 상에 저장되거나 또는 컴퓨터 판독가능 매체를 통해서 송신될 수도 있으며, 하드웨어-기반의 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터 판독가능 매체는 컴퓨터 판독가능 저장 매체들을 포함할 수도 있으며, 이 컴퓨터 판독가능 저장 매체들은 데이터 저장 매체와 같은 유형의 매체, 또는 예컨대, 통신 프로토콜에 따라서 한 장소로부터 다른 장소로의 컴퓨터 프로그램의 전송을 용이하게 하는 임의의 매체를 포함한 통신 매체들에 대응한다. 이러한 방법으로, 컴퓨터 판독가능 매체들은 일반적으로 (1) 비일시성 유형의 컴퓨터 판독가능 저장 매체 또는 (2) 신호 또는 캐리어 파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체는 본 개시물에서 설명하는 기법들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 추출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 가용 매체들일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터 판독가능 매체를 포함할 수도 있다.

[0371] 일 예로서, 이에 한정하지 않고, 이런 컴퓨터 판독가능 저장 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광디스크 스토리지, 자기디스크 스토리지, 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 원하는 프로그램 코드를 명령들 또는 데이터 구조들의 형태로 저장하는데 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 접속이 컴퓨터 판독가능 매체로 적절히 지칭된다. 예를 들어, 동축 케이블, 광섬유 케이블, 이중 권선, 디지털 가입자 회선 (DSL), 또는 무선 기술들 예컨대 적외선, 무선, 및 마이크로파를 이용하여 명령들이 웹사이트, 서버, 또는 다른 원격 소스로부터 송신되는 경우, 동축 케이블, 광섬유 케이블, 이중 권선, DSL, 또는 무선 기술들, 예컨대 적외선, 무선, 및 마이크로파가 그 매체의 정의에 포함된다. 그러나, 컴퓨터 판독가능 저장 매체 및 데이터 저장 매체는 접속부들, 캐리어 파들, 신호들, 또는 다른 일시적 매체를 포함하지 않고, 그 대신, 비-일시적인 유형의 저장 매체에 송신되는 것으로 해석되어야 한다. 디스크 (disk) 및 디스크 (disc) 는, 본원에서 사용할 때, 콤팩트 디스크 (CD), 레이저 디스크, 광 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크 및 블루-레이 디스크를 포함하며, 디스크들 (disks) 은 데이터를 자기적으로 보통 재생하지만, 디스크들 (discs) 은 레이저로 데이터를 광학적으로 재생한다. 앞에서 언급한 것들의 조합들이 또한 컴퓨터 판독가능 매체들의 범위 내에 포함되어야 한다.

[0372] 명령들은 하나 이상의 디지털 신호 프로세서들 (DSPs), 범용 마이크로프로세서들, 주문형 집적회로들 (ASICs),

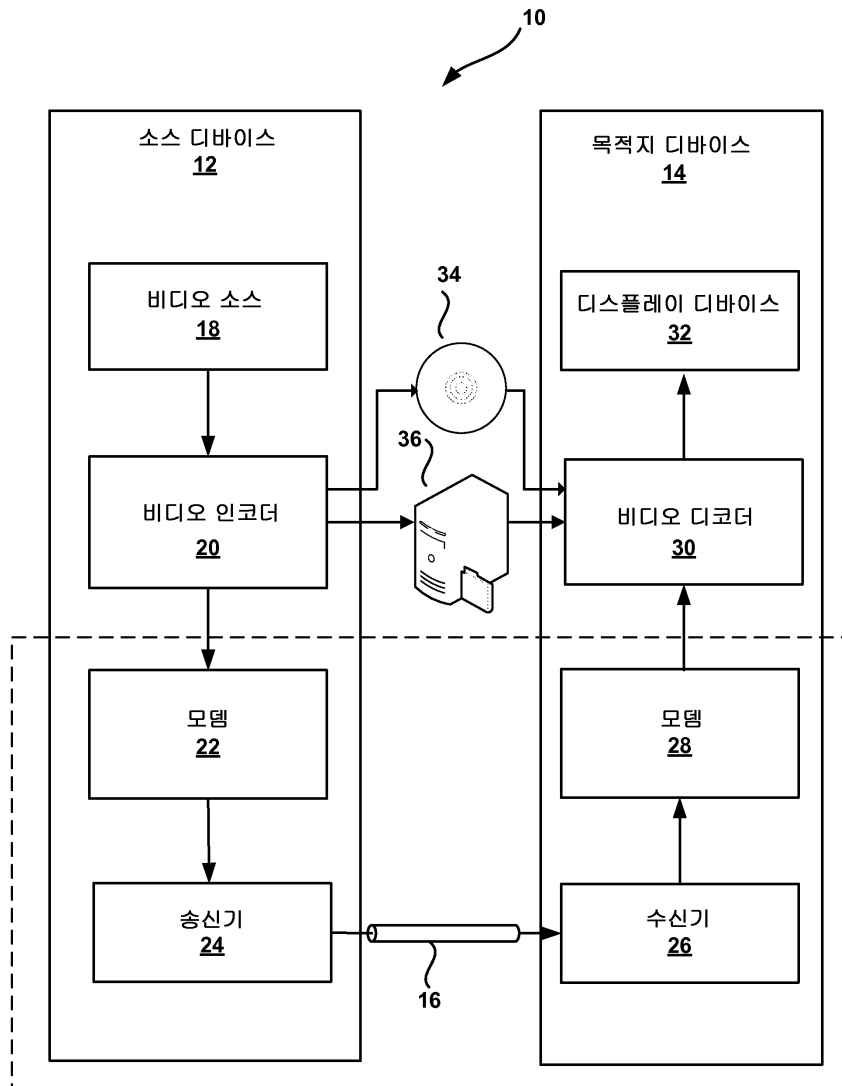
필드 프로그래밍가능 로직 어레이들 (FPGAs), 또는 다른 등가의 통합 또는 이산 로직 회로와 같은, 하나 이상의 프로세서들에 의해 실행될 수도 있다. 따라서, 용어 “프로세서” 는, 본원에서 사용될 때 전술한 구조 중 임의의 구조 또는 본원에서 설명하는 기법들의 구현에 적합한 임의의 다른 구조를 지칭할 수도 있다. 게다가, 일부 양태들에서, 본원에서 설명하는 기능 전용 하드웨어 및/또는 인코딩 및 디코딩을 위해 구성되는 소프트웨어 모듈들 내에 제공되거나, 또는 결합된 코덱에 포함될 수도 있다. 또한, 이 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들로 전적으로 구현될 수 있다.

[0373] 본 개시물의 기법들은 무선 핸드셋, 집적 회로 (IC) 또는 ICs 의 세트 (예컨대, 칩 세트) 를 포함한, 매우 다양한 디바이스들 또는 장치들로 구현될 수도 있다. 개시한 기법들을 수행하도록 구성되는 디바이스들의 기능적 양태들을 강조하기 위해서 여러 구성요소들, 모듈들, 또는 유닛들이 본 개시물에서 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 필요로 하지는 않는다. 더 정확히 말하면, 위에서 설명한 바와 같이, 여러 유닛들이 코덱 하드웨어 유닛에 결합되거나 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 위에서 설명한 바와 같은 하나 이상의 프로세서들을 포함한, 상호작용하는 하드웨어 유닛들의 컬렉션으로 제공될 수도 있다.

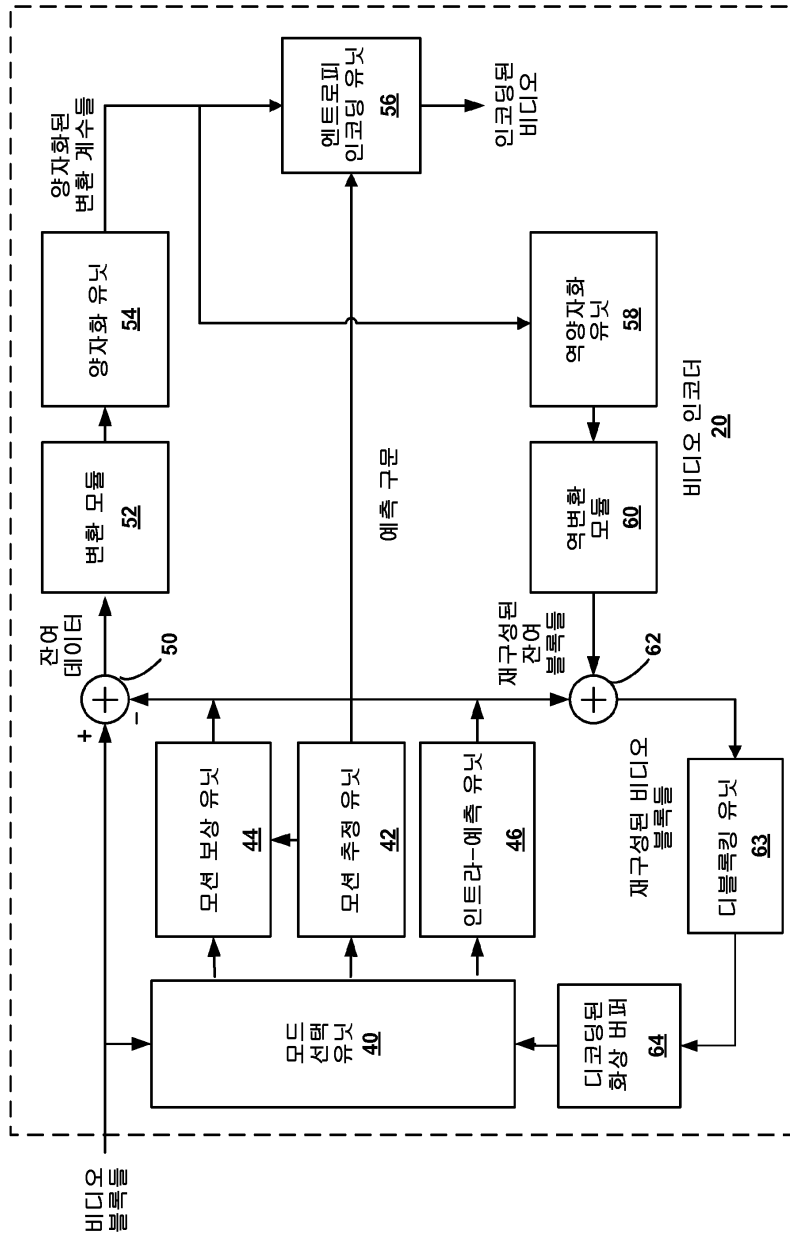
[0374] 여러 예들이 설명되었다. 이들 및 다른 예들은 다음 청구항들의 범위 이내이다.

도면

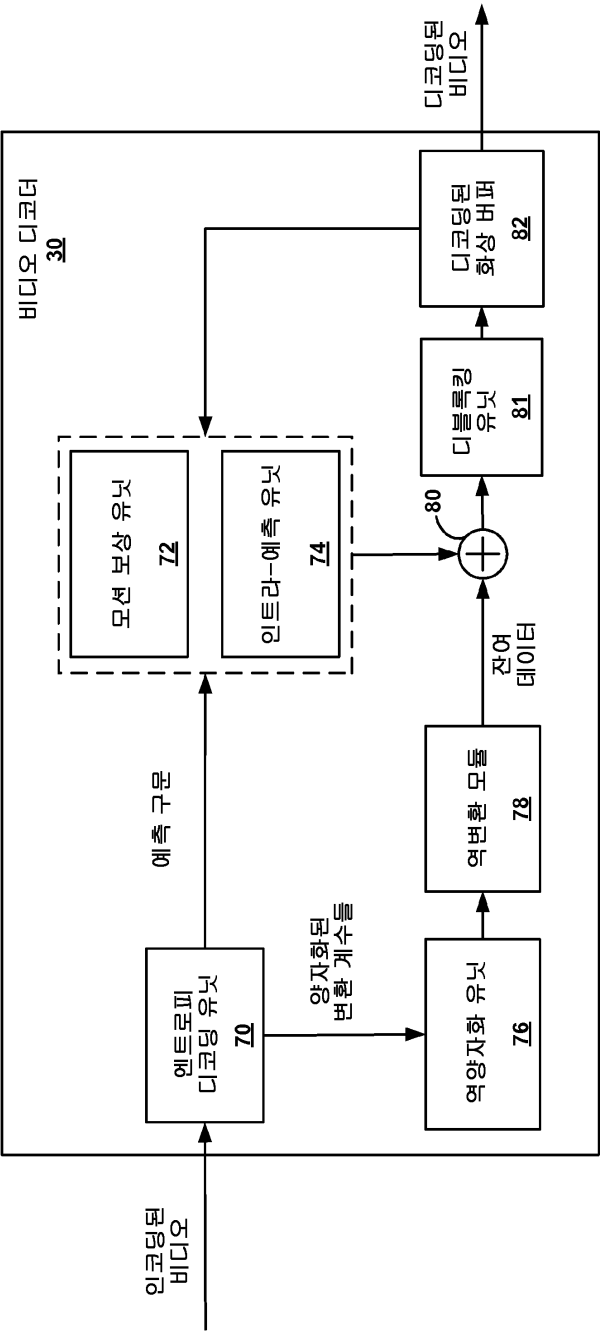
도면1



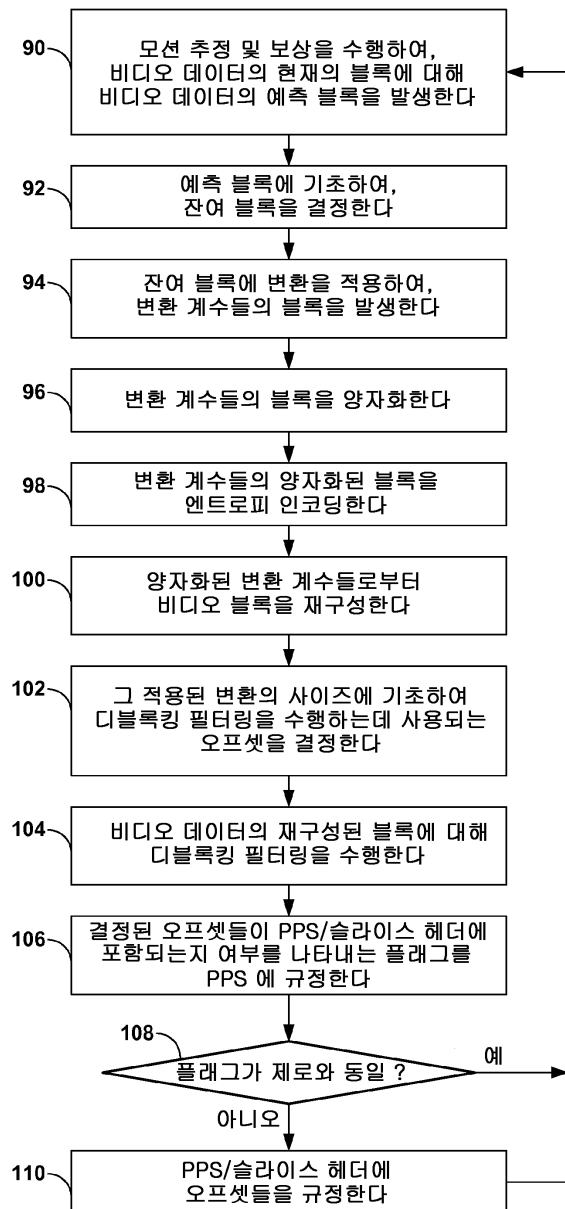
도면2



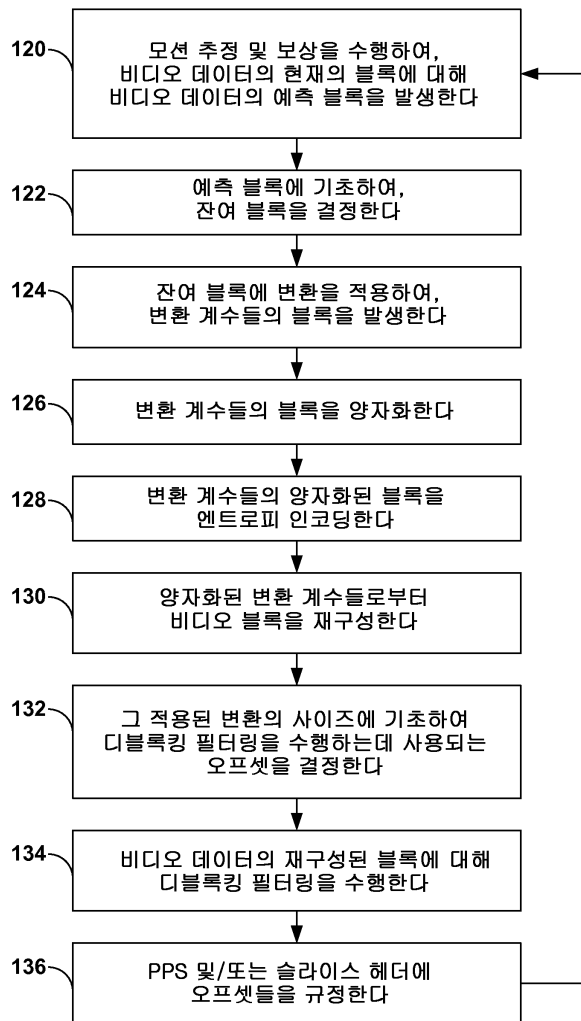
도면3



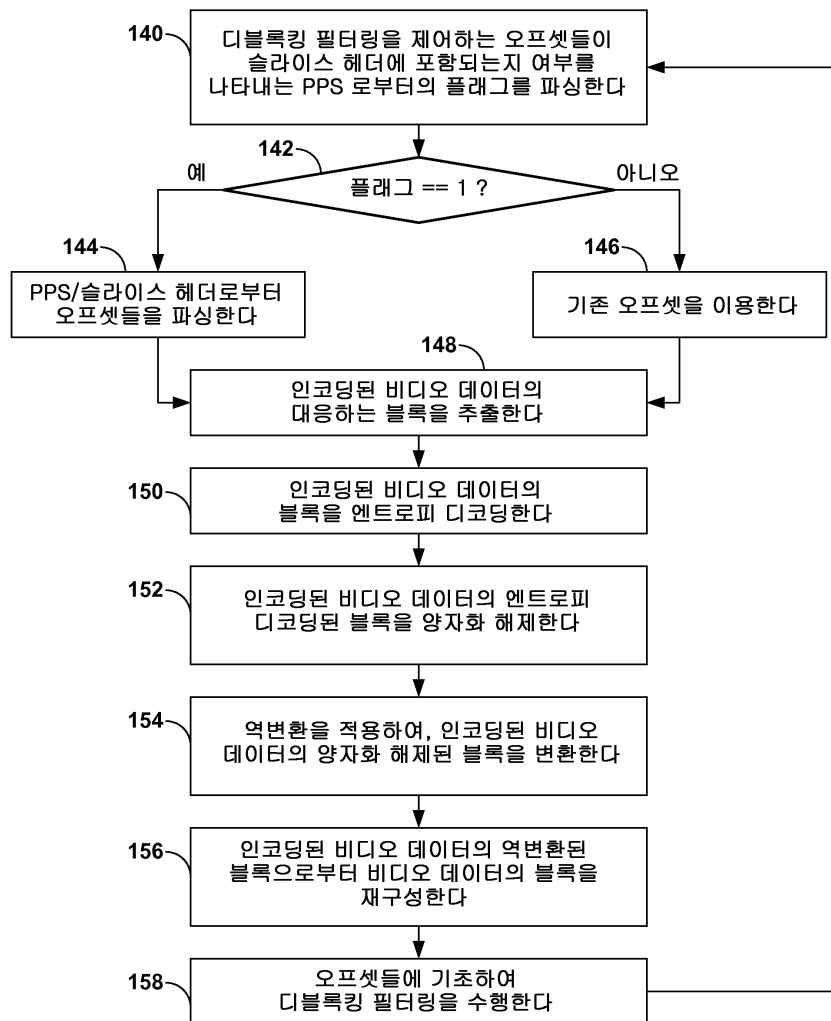
도면4



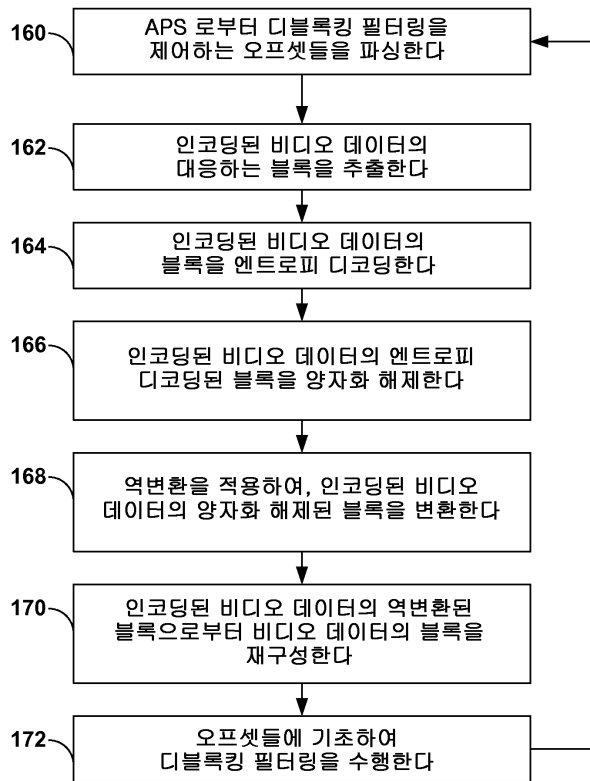
도면5



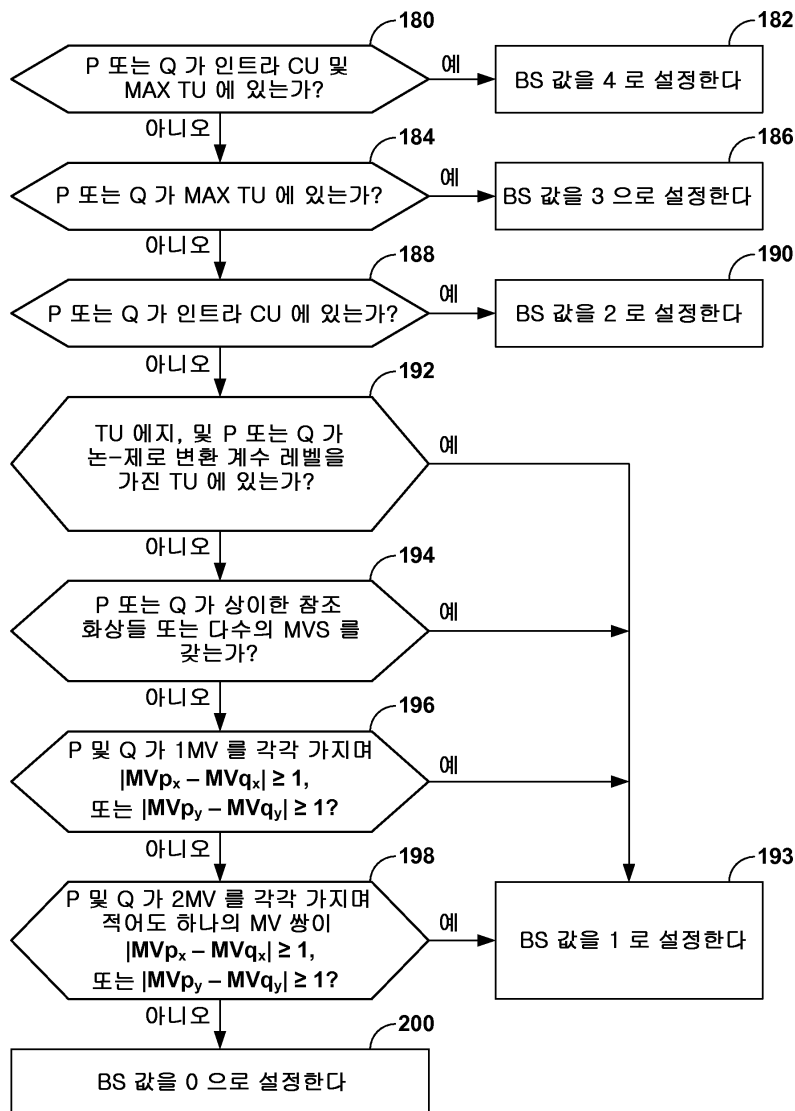
도면6



도면7



도면8



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 69, 77, 85, 93

【변경전】

상기 변환의 사이즈가 최대 변환 유닛 사이즈가~

【변경후】

변환의 사이즈가 최대 변환 유닛 사이즈가~