



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2015년09월09일

(11) 등록번호 10-1552031

(24) 등록일자 2015년09월03일

(51) 국제특허분류(Int. Cl.)

H04N 19/00 (2014.01) H04N 19/61 (2014.01)

(21) 출원번호 10-2013-7024783

(22) 출원일자(국제) 2012년02월22일

심사청구일자 2013년09월17일

(85) 번역문제출일자 2013년09월17일

(65) 공개번호 10-2013-0129440

(43) 공개일자 2013년11월28일

(86) 국제출원번호 PCT/US2012/026165

(87) 국제공개번호 WO 2012/116094

국제공개일자 2012년08월30일

(30) 우선권주장

13/401,573 2012년02월21일 미국(US)

(뒷면에 계속)

(56) 선행기술조사문헌

KARCZEWICZ M et al: "Video coding technology proposal by Qualcomm", JCT-VC Meeting; 15-4-2010 - 23-4-2010, Dresden, page 1-13.

W-J CHIEN and M KARCZEWICZ: "Adaptive Filter Based on Combination of Sum-Modified Laplacian Filter Indexing and Quadtree Partitioning", VCEG MEETING;89,1-7-2009 - 8-7-2009, VCEG-AL27.

WO2011126759 A1

(73) 특허권자

퀄컴 인코포레이티드

미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775

(72) 발명자

정 인석

미국 92121 캘리포니아주 샌디에고 모어하우스 드라이브 5775

카르체비츠 마르타

미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775

(74) 대리인

특허법인코리어나

전체 청구항 수 : 총 40 항

심사관 : 김영태

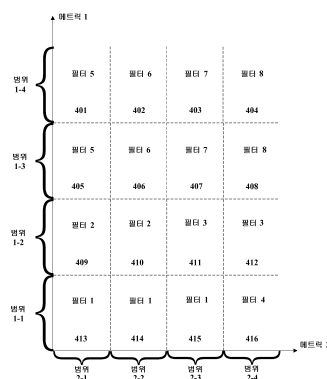
(54) 발명의 명칭 멀티-메트릭 필터링

(57) 요약

인코더는 2 이상의 메트릭들에 대한 범위 조합들의 매핑을 디코더로 시그널링하기 위해 코드워드들의 시리즈를 생성할 수 있다. 인코더는 디코딩되고 있는 현재 조합이 제 1 메트릭에 대하여 동일한 범위를 공유하는 가장 최근에 디코딩된 조합과 동일한 필터에 매핑하는 것을 나타내도록 제 1 코드워드를 생성할 수 있다. 인코더

(뒷면에 계속)

대표도 - 도4a



는 현재 조합이 제 2 메트릭에 대하여 동일한 범위를 공유하는 가장 최근에 디코딩된 조합과 동일한 필터에 맵핑하는 것을 나타내도록 제 2 코드워드를 생성할 수 있다. 현재 조합이 가장 최근에 디코딩된 조합들 중 어느 하나와 동일한 필터에 맵핑하지 않는다면, 인코더는 현재 조합에 맵핑하는 필터를 나타내는 제 3 코드워드를 생성할 수 있다.

(30) 우선권주장

61/445,967	2011년02월23일	미국(US)
61/448,771	2011년03월03일	미국(US)
61/473,713	2011년04월08일	미국(US)
61/476,260	2011년04월16일	미국(US)
61/478,287	2011년04월22일	미국(US)
61/503,426	2011년06월30일	미국(US)
61/503,434	2011년06월30일	미국(US)
61/503,440	2011년06월30일	미국(US)
61/527,463	2011년08월25일	미국(US)
61/531,571	2011년09월06일	미국(US)

특허청구의 범위

청구항 1

비디오 데이터를 디코딩하는 방법으로서,

2 이상의 매트릭들에 대한 범위 조합들의 필터들에의 맵핑을 생성하는 단계로서, 상기 범위 조합은 제 1 매트릭에 대한 범위와 제 2 매트릭에 대한 범위를 포함하고, 상기 제 1 매트릭은 픽셀들의 블록의 픽셀 값들에 기초하여 결정되고, 상기 제 2 매트릭은 상기 픽셀들의 블록의 픽셀 값들에 기초하여 결정되며, 그리고 상기 제 2 매트릭은 상기 제 1 매트릭과 상이한, 상기 범위 조합들의 필터들에의 맵핑을 생성하는 단계;

디코딩 중인 현재 범위 조합이 이전에 디코딩된 이전 범위 조합과 동일한 필터에 맵핑되는 것을 시그널링하는 제 1 코드워드를 수신하는 것에 응답하여, 상기 현재 범위 조합을 상기 동일한 필터에 맵핑하는 단계; 및

상기 현재 범위 조합이 상기 이전 범위 조합과 상이한 필터에 맵핑되는 것을 시그널링하는 제 2 코드워드를 수신하는 것에 응답하여, 상기 현재 범위 조합을 새로운 필터에 맵핑하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 2

제 1 항에 있어서,

상기 새로운 필터는 필터 계수들이 시그널링되는 순서에 기초하여 결정되는, 비디오 데이터를 디코딩하는 방법.

청구항 3

제 1 항에 있어서,

상기 현재 범위 조합은 상기 범위 조합들의 알려진 송신 순서에 기초하여 결정되는, 비디오 데이터를 디코딩하는 방법.

청구항 4

제 1 항에 있어서,

상기 제 1 매트릭은 활동 매트릭을 포함하고, 상기 제 2 매트릭은 방향 매트릭을 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 5

제 4 항에 있어서,

상기 방향 매트릭은 수평 활동의 측정치를 수직 활동의 측정치와 비교하는 것에 기초하여 결정되는, 비디오 데이터를 디코딩하는 방법.

청구항 6

제 5 항에 있어서,

상기 수평 활동의 측정치는 현재의 픽셀의 픽셀 값을 좌측 이웃하는 픽셀의 픽셀 값 및 우측 이웃하는 픽셀의 픽셀 값과 비교하는 것에 기초하여 결정되는, 비디오 데이터를 디코딩하는 방법.

청구항 7

제 5 항에 있어서,

상기 수직 활동의 측정치는 현재의 픽셀의 픽셀 값을 상부 이웃하는 픽셀의 픽셀 값 및 하부 이웃하는 픽셀의 픽셀 값과 비교하는 것에 기초하여 결정되는, 비디오 데이터를 디코딩하는 방법.

청구항 8

제 5 항에 있어서,

상기 활동 메트릭은 상기 수평 활동의 측정치와 상기 수직 활동의 측정치의 합에 기초하여 결정되는, 비디오 데이터를 디코딩하는 방법.

청구항 9

제 1 항에 있어서,

비디오 인코더는 다수의 필터들로 쿼드트리-기반의 적응형 루프 필터 (QALF) 방식을 구현하는, 비디오 데이터를 디코딩하는 방법.

청구항 10

비디오 디코더를 포함하는, 비디오 데이터를 코딩하는 장치로서,

상기 비디오 디코더는,

2 이상의 매트릭들에 대한 범위 조합들의 필터들에의 맵핑을 생성하는 것으로서, 상기 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하고, 상기 제 1 메트릭은 픽셀들의 블록의 픽셀 값들에 기초하여 결정되고, 상기 제 2 메트릭은 상기 픽셀들의 블록의 픽셀 값들에 기초하여 결정되며, 그리고 상기 제 2 메트릭은 상기 제 1 메트릭과 상이한, 상기 범위 조합들의 필터들에의 맵핑을 생성하고;

디코딩 중인 현재 범위 조합이 이전에 디코딩된 이전 범위 조합과 동일한 필터에 맵핑되는 것을 시그널링하는 제 1 코드워드를 수신하는 것에 응답하여, 상기 현재 범위 조합을 상기 이전 범위 조합과 동일한 필터에 맵핑하며; 그리고

상기 현재 범위 조합이 상기 이전 범위 조합과 상이한 필터에 맵핑되는 것을 시그널링하는 제 2 코드워드를 수신하는 것에 응답하여, 상기 현재 범위 조합을 상기 제 2 코드워드에 의해 식별되는 필터에 맵핑하도록 구성되는, 비디오 데이터를 코딩하는 장치.

청구항 11

제 10 항에 있어서,

새로운 필터는 필터 계수들이 시그널링되는 순서에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 12

제 10 항에 있어서,

상기 현재 범위 조합은 상기 범위 조합들의 알려진 송신 순서에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 13

제 10 항에 있어서,

상기 제 1 메트릭은 활동 메트릭을 포함하고, 상기 제 2 메트릭은 방향 메트릭을 포함하는, 비디오 데이터를 코딩하는 장치.

청구항 14

제 13 항에 있어서,

상기 방향 메트릭은 수평 활동의 측정치를 수직 활동의 측정치와 비교하는 것에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 15

제 14 항에 있어서,

상기 수평 활동의 측정치는 현재의 픽셀의 픽셀 값을 좌측 이웃하는 픽셀의 픽셀 값 및 우측 이웃하는 픽셀의

픽셀 값과 비교하는 것에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 16

제 14 항에 있어서,

상기 수직 활동의 측정치는 현재의 픽셀의 픽셀 값을 상부 이웃하는 픽셀의 픽셀 값 및 하부 이웃하는 픽셀의 픽셀 값과 비교하는 것에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 17

제 14 항에 있어서,

상기 활동 메트릭은 상기 수평 활동의 측정치와 상기 수직 활동의 측정치의 합에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 18

제 10 항에 있어서,

상기 비디오 디코더는 다수의 필터들로 쿼드트리-기반의 적응형 루프 필터 (QALF) 방식을 구현하는, 비디오 데이터를 코딩하는 장치.

청구항 19

2 이상의 매트릭들에 대한 범위 조합들의 필터들에의 맵핑을 생성하는 수단으로서, 상기 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하고, 상기 제 1 메트릭은 픽셀들의 블록의 픽셀 값들에 기초하여 결정되고, 상기 제 2 메트릭은 상기 픽셀들의 블록의 픽셀 값들에 기초하여 결정되며, 그리고 상기 제 2 메트릭은 상기 제 1 메트릭과 상이한, 상기 범위 조합들의 필터들에의 맵핑을 생성하는 수단;

디코딩 중인 현재 범위 조합이 이전에 디코딩된 이전 범위 조합과 동일한 필터에 맵핑되는 것을 시그널링하는 제 1 코드워드를 수신하는 것에 응답하여, 상기 현재 범위 조합을 상기 이전 범위 조합과 동일한 필터에 맵핑하는 수단; 및

상기 현재 범위 조합이 상기 이전 범위 조합과 상이한 필터에 맵핑되는 것을 시그널링하는 제 2 코드워드를 수신하는 것에 응답하여, 상기 현재 범위 조합을 상기 제 2 코드워드에 의해 식별되는 필터에 맵핑하는 수단을 포함하는, 비디오 데이터를 디코딩하기 위한 장치.

청구항 20

명령들을 저장하는 컴퓨터-판독가능 저장 매체로서,

상기 명령들은, 실행될 경우, 비디오 데이터를 디코딩하도록 구성되는 하나 이상의 프로세서들로 하여금,

2 이상의 매트릭들에 대한 범위 조합들의 필터들에의 맵핑을 생성하게 하는 것으로서, 상기 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하고, 상기 제 1 메트릭은 픽셀들의 블록의 픽셀 값들에 기초하여 결정되고, 상기 제 2 메트릭은 상기 픽셀들의 블록의 픽셀 값들에 기초하여 결정되며, 그리고 상기 제 2 메트릭은 상기 제 1 메트릭과 상이한, 상기 범위 조합들의 필터들에의 맵핑을 생성하게 하고;

디코딩 중인 현재 범위 조합이 이전에 디코딩된 이전 범위 조합과 동일한 필터에 맵핑되는 것을 시그널링하는 제 1 코드워드를 수신하는 것에 응답하여, 상기 현재 범위 조합을 상기 이전 범위 조합과 동일한 필터에 맵핑하게 하며; 그리고

상기 현재 범위 조합이 상기 이전 범위 조합과 상이한 필터에 맵핑되는 것을 시그널링하는 제 2 코드워드를 수신하는 것에 응답하여, 상기 현재 범위 조합을 상기 제 2 코드워드에 의해 식별되는 필터에 맵핑하게 하는, 컴퓨터-판독가능 저장 매체.

청구항 21

비디오 데이터를 인코딩하는 방법으로서,

2 이상의 매트릭들에 대한 범위 조합들의 필터들에의 맵핑을 결정하는 단계로서, 상기 범위 조합은 제 1 메트릭

에 대한 범위와 제 2 메트릭에 대한 범위를 포함하고, 상기 제 1 메트릭은 픽셀들의 블록의 픽셀 값들에 기초하여 결정되고, 상기 제 2 메트릭은 상기 픽셀들의 블록의 픽셀 값들에 기초하여 결정되며, 그리고 상기 제 2 메트릭은 상기 제 1 메트릭과 상이한, 상기 범위 조합들의 필터들에의 맵핑을 결정하는 단계;

디코딩 중인 현재 범위 조합이 이전에 디코딩된 이전 범위 조합과 동일한 필터에 맵핑될 경우, 제 1 코드워드를 생성하는 단계; 및

상기 현재 범위 조합이 상기 이전 범위 조합과 상이한 필터에 맵핑될 경우, 제 2 코드워드를 생성하는 단계를 포함하며,

상기 제 2 코드워드는 상기 현재 범위 조합에 맵핑된 필터를 식별하는, 비디오 데이터를 인코딩하는 방법.

청구항 22

제 21 항에 있어서,

새로운 필터는 필터 계수들이 시그널링되는 순서에 기초하여 결정되는, 비디오 데이터를 인코딩하는 방법.

청구항 23

제 21 항에 있어서,

상기 현재 범위 조합은 상기 범위 조합들의 알려진 송신 순서에 기초하여 결정되는, 비디오 데이터를 인코딩하는 방법.

청구항 24

제 21 항에 있어서,

상기 제 1 메트릭은 활동 메트릭을 포함하고, 상기 제 2 메트릭은 방향 메트릭을 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 25

제 24 항에 있어서,

상기 방향 메트릭은 수평 활동의 측정치를 수직 활동의 측정치와 비교하는 것에 기초하여 결정되는, 비디오 데이터를 인코딩하는 방법.

청구항 26

제 25 항에 있어서,

상기 수평 활동의 측정치는 현재의 픽셀의 픽셀 값을 좌측 이웃하는 픽셀의 픽셀 값 및 우측 이웃하는 픽셀의 픽셀 값과 비교하는 것에 기초하여 결정되는, 비디오 데이터를 인코딩하는 방법.

청구항 27

제 25 항에 있어서,

상기 수직 활동의 측정치는 현재의 픽셀의 픽셀 값을 상부 이웃하는 픽셀의 픽셀 값 및 하부 이웃하는 픽셀의 픽셀 값과 비교하는 것에 기초하여 결정되는, 비디오 데이터를 인코딩하는 방법.

청구항 28

제 25 항에 있어서,

상기 활동 메트릭은 상기 수평 활동의 측정치와 상기 수직 활동의 측정치의 합에 기초하여 결정되는, 비디오 데이터를 인코딩하는 방법.

청구항 29

제 21 항에 있어서,

비디오 인코더는 다수의 필터들로 쿼드트리-기반의 적응형 루프 필터 (QALF) 방식을 구현하는, 비디오 데이터를 인코딩하는 방법.

청구항 30

비디오 인코더를 포함하는, 비디오 데이터를 코딩하는 장치로서,

상기 비디오 인코더는,

2 이상의 매트릭들에 대한 범위 조합들의 필터들에의 맵핑을 결정하는 것으로서, 상기 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하고, 상기 제 1 메트릭은 픽셀들의 블록의 픽셀 값들에 기초하여 결정되고, 상기 제 2 메트릭은 상기 픽셀들의 블록의 픽셀 값들에 기초하여 결정되며, 그리고 상기 제 2 메트릭은 상기 제 1 메트릭과 상이한, 상기 범위 조합들의 필터들에의 맵핑을 결정하고;

디코딩 중인 현재 범위 조합이 이전에 디코딩된 이전 범위 조합과 동일한 필터에 맵핑될 경우, 제 1 코드워드를 생성하며; 그리고

상기 현재 범위 조합이 상기 이전 범위 조합과 상이한 필터에 맵핑될 경우, 제 2 코드워드를 생성하도록

구성되며, 상기 제 2 코드워드는 상기 현재 범위 조합에 맵핑된 필터를 식별하는, 비디오 데이터를 코딩하는 장치.

청구항 31

제 30 항에 있어서,

새로운 필터는 필터 계수들이 시그널링되는 순서에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 32

제 30 항에 있어서,

상기 현재 범위 조합은 상기 범위 조합들의 알려진 송신 순서에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 33

제 30 항에 있어서,

상기 제 1 메트릭은 활동 메트릭을 포함하고, 상기 제 2 메트릭은 방향 메트릭을 포함하는, 비디오 데이터를 코딩하는 장치.

청구항 34

제 33 항에 있어서,

상기 방향 메트릭은 수평 활동의 측정치를 수직 활동의 측정치와 비교하는 것에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 35

제 34 항에 있어서,

상기 수평 활동의 측정치는 현재의 픽셀의 픽셀 값을 좌측 이웃하는 픽셀의 픽셀 값 및 우측 이웃하는 픽셀의 픽셀 값과 비교하는 것에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 36

제 34 항에 있어서,

상기 수직 활동의 측정치는 현재의 픽셀의 픽셀 값을 상부 이웃하는 픽셀의 픽셀 값 및 하부 이웃하는 픽셀의 픽셀 값과 비교하는 것에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 37

제 34 항에 있어서,

상기 활동 메트릭은 상기 수평 활동의 측정치와 상기 수직 활동의 측정치의 합에 기초하여 결정되는, 비디오 데이터를 코딩하는 장치.

청구항 38

제 30 항에 있어서,

상기 비디오 인코더는 다수의 필터들로 쿼드트리-기반의 적응형 루프 필터 (QALF) 방식을 구현하는, 비디오 데이터를 코딩하는 장치.

청구항 39

2 이상의 매트릭들에 대한 범위 조합들의 필터들에의 맵핑을 결정하는 수단으로서, 상기 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하고, 상기 제 1 메트릭은 픽셀들의 블록의 픽셀 값들에 기초하여 결정되고, 상기 제 2 메트릭은 상기 픽셀들의 블록의 픽셀 값들에 기초하여 결정되며, 그리고 상기 제 2 메트릭은 상기 제 1 메트릭과 상이한, 상기 범위 조합들의 필터들에의 맵핑을 결정하는 수단;

디코딩 중인 현재 범위 조합이 이전에 디코딩된 이전 범위 조합과 동일한 필터에 맵핑될 경우, 제 1 코드워드를 생성하는 수단; 및

상기 현재 범위 조합이 상기 이전 범위 조합과 상이한 필터에 맵핑될 경우, 제 2 코드워드를 생성하는 수단을 포함하며,

상기 제 2 코드워드는 상기 현재 범위 조합에 맵핑된 필터를 식별하는, 비디오 데이터를 인코딩하기 위한 장치.

청구항 40

명령들을 저장하는 컴퓨터-판독가능 저장 매체로서,

상기 명령들은, 실행될 경우, 비디오 데이터를 인코딩하도록 구성되는 하나 이상의 프로세서들로 하여금,

2 이상의 매트릭들에 대한 범위 조합들의 필터들에의 맵핑을 결정하게 하는 것으로서, 상기 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하고, 상기 제 1 메트릭은 픽셀들의 블록의 픽셀 값들에 기초하여 결정되고, 상기 제 2 메트릭은 상기 픽셀들의 블록의 픽셀 값들에 기초하여 결정되며, 그리고 상기 제 2 메트릭은 상기 제 1 메트릭과 상이한, 상기 범위 조합들의 필터들에의 맵핑을 결정하게 하고;

디코딩 중인 현재 범위 조합이 이전에 디코딩된 이전 범위 조합과 동일한 필터에 맵핑될 경우, 제 1 코드워드를 생성하게 하며; 그리고

상기 현재 범위 조합이 상기 이전 범위 조합과 상이한 필터에 맵핑될 경우, 제 2 코드워드를 생성하게 하며,

상기 제 2 코드워드는 상기 현재 범위 조합에 맵핑된 필터를 식별하는, 컴퓨터-판독가능 저장 매체.

명세서

기술분야

[0001]

본 출원은, 2011년 2월 23일에 출원된 미국 가특허출원 번호 제 61/445,967호; 2011년 3월 03일에 출원된 미국 가특허출원 번호 제 61/448,771호; 2011년 4월 08일에 출원된 미국 가특허출원 번호 제 61/473,713호; 2011년 4월 16일에 출원된 미국 가특허출원 번호 제 61/476,260호; 2011년 4월 22일에 출원된 미국 가특허출원 번호 제 61/478,287호; 2011년 6월 30일에 출원된 미국 가특허출원 번호 제 61/503,426호; 2011년 6월 30일에 출원된 미국 가특허출원 번호 제 61/503,434호; 2011년 6월 30일에 출원된 미국 가특허출원 번호 제 61/503,440호; 2011년 8월 25일에 출원된 미국 가특허출원 번호 제 61/527,463호; 2011년 9월 06일에 출원된 미국 가특허출원 번호 제 61/531,571호에 대해 우선권을 주장하며, 이들 각각의 전체 내용들이 본원에 전체적으로 참고로 포함된다.

[0002]

본 개시물은 비디오 데이터를 압축하는데 사용되는 블록-기반의 디지털 비디오 코딩에 관한 것으로, 좀더 구체

적으로는, 비디오 블록들의 필터링을 위한 기법들에 관한 것이다.

배경 기술

[0003]

디지털 비디오 능력들은 디지털 텔레비전, 디지털 직접 브로드캐스트 시스템들, 무선 전화기 핸드셋들과 같은 무선 통신 디바이스들, 무선 브로드캐스트 시스템들, 개인 휴대정보 단말기들 (PDAs), 랩탑 컴퓨터들, 데스크탑 컴퓨터들, 태블릿 컴퓨터들, 디지털 카메라들, 디지털 레코딩 디바이스들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들 등을 포함한, 광범위한 디바이스들에 통합될 수 있다. 디지털 비디오 디바이스들은 디지털 비디오를 좀더 효율적으로 송수신하기 위해, MPEG-2, MPEG-4, 또는 ITU-T H.264/MPEG-4, 파트 10, AVC (Advanced Video Coding) 와 같은 비디오 압축 기법들을 구현한다. 비디오 압축 기법들은 비디오 시퀀스들에 고유한 리던던시를 감소시키거나 또는 제거하기 위해 공간 및 시간 예측을 수행한다. "Joint Collaborative Team - Video Coding" (JCTVC) 에 의해 개발되고 있는, MPEG 과 ITU-T 사이의 공동연구의 성과인, HEVC (High Efficiency Video Coding) 표준과 같은 새로운 비디오 표준들이 계속 대두하여 진화하고 있다. 이 새로운 HEVC 표준은 또한 H.265 으로 종종 지칭된다.

[0004]

블록-기반의 비디오 압축 기법들은 공간 예측 및/또는 시간 예측을 수행할 수도 있다. 인트라-코딩은 비디오 프레임, 비디오 프레임의 슬라이스 등일 수도 있는 소정의 코딩된 비디오 유닛 내 비디오 블록들 사이의 공간 리던던시를 감소시키거나 또는 제거하기 위해 공간 예측에 의존한다. 이에 반해, 인터-코딩은 비디오 시퀀스의 연속적인 코딩 유닛들의 비디오 블록들 사이의 시간 리던던시를 감소시키거나 또는 제거하기 위해 시간 예측에 의존한다. 인트라-코딩에 있어서, 비디오 인코더는 코딩된 비디오의 동일한 유닛 내의 다른 데이터에 기초하여 데이터를 압축하기 위해 공간 예측을 수행한다. 인터-코딩에 있어서, 비디오 인코더는 2개 이상의 인접한 코딩된 비디오의 유닛들의 대응하는 비디오 블록들의 이동을 추적하기 위해 모션 추정 및 모션 보상을 수행한다.

[0005]

코딩된 비디오 블록은 예측 블록을 생성하거나 또는 식별하는데 사용될 수 있는 예측 정보, 및 코딩중인 블록과 예측 블록 사이의 차이들을 나타내는 데이터의 잔여 블록으로 표현될 수도 있다. 인터-코딩의 경우, 하나 이상의 모션 벡터들이 이전 또는 후속 코딩 유닛으로부터 데이터의 예측 블록을 식별하는데 사용되는 한편, 인트라-코딩의 경우, 예측 모드가 코딩중인 비디오 블록과 연관되는 CU 내 데이터에 기초하여 예측 블록을 생성하는데 사용될 수 있다. 인트라 코딩 및 인터-코딩 양자는 여러 상이한 예측 모드들을 정의할 수도 있으며, 상이한 블록 사이즈들 및/또는 코딩에 사용되는 예측 기법들을 정의할 수도 있다. 추가적인 유형들의 구문 엘리먼트들이 또한 코딩 프로세스에 이용되는 코딩 기법들 또는 파라미터들을 제어하거나 또는 정의하기 위해, 인코딩된 비디오 데이터의 부분으로서 포함될 수도 있다.

[0006]

블록-기반 예측 코딩 후, 비디오 인코더는 잔여 블록의 통신과 연관되는 비트 레이트를 추가로 감소시키기 위해 변환, 양자화 및 엔트로피 코딩 프로세스들을 적용할 수도 있다. 변환 기법들은 이산 코사인 변환들 (DCTs) 또는 개념적으로 유사한 프로세스들, 예컨대 웨이블릿 변환들, 정수 변환들, 또는 다른 유형들의 변환들을 포함할 수도 있다. 이산 코사인 변환 프로세스에서, 일 예로서, 변환 프로세스는 픽셀 차이 값들의 세트를 변환 계수들로 변환하며, 이 변환 계수들은 주파수 도메인에서 픽셀 값들의 에너지를 나타낼 수도 있다. 양자화가 변환 계수들에 적용되며, 일반적으로 임의의 주어진 변환 계수와 연관되는 비트수를 제한하는 프로세스를 수반한다. 엔트로피 코딩은 양자화된 변환 계수들의 시퀀스를 일괄 압축하는 하나 이상의 프로세스들을 포함한다.

[0007]

비디오 블록들의 필터링이 재구성된 비디오 블록들에, 인코딩 및 디코딩 루프들의 부분으로서, 또는 사후-필터링 프로세스의 부분으로서 제공될 수도 있다. 필터링은 예를 들어, 블록-기반의 비디오 코딩에 공통적인 블록킹 현상 또는 다른 아티팩트들을 감소시키기 위해서, 일반적으로 이용된다. 필터 계수들 (종종, 필터 탭들로서 지칭됨) 은, 다른 방법들로 블록킹 현상을 감소시키거나 및/또는 비디오 품질을 향상시킬 수 있는 바람직한 비디오 블록 필터링의 레벨들을 증진시키기 위해, 정의되거나 또는 선택될 수도 있다. 필터 계수들의 세트는, 예를 들어, 어떻게 필터링이 비디오 블록들의 에지 또는 비디오 블록들 내 다른 로케이션들을 따라서 적용되는 지를 정의할 수도 있다. 상이한 필터 계수들은 비디오 블록들의 상이한 픽셀들에 대해 상이한 필터링 레벨들을 초래할 수도 있다. 필터링은, 예를 들어, 원치않는 아티팩트들을 제거하는 것을 돕기 위해서 인접한 픽셀 값들의 강도에서의 차이들을 평활하거나 또는 날카롭게 할 수도 있다.

발명의 내용

과제의 해결 수단

- [0008] 본 개시물은 비디오 인코딩 및/또는 비디오 디코딩 프로세스에서 비디오 데이터의 필터링과 연관되는 기법들을 기술한다. 본 개시물에 따르면, 필터링이 인코더에 적용되며, 필터 정보가 디코더로 하여금 인코더에 적용되었던 필터링을 식별할 수 있도록, 비트스트림으로 인코딩된다. 디코더는 필터 정보를 포함하는 인코딩된 비디오 데이터를 수신하며, 비디오 데이터를 디코딩하고, 그리고 필터링 정보에 기초하여 필터링을 적용한다.
- 이러한 방법으로, 디코더는 인코더에서 적용되었던 동일한 필터링을 적용한다. 본 개시물의 기법들에 따르면, 프레임 단위, 슬라이스 단위, 또는 LCU 단위로, 인코더는 필터들의 하나 이상의 세트들을 선택할 수도 있으며, 코딩된-유닛 단위로, 인코더는 필터링을 적용할지 여부를 결정할 수도 있다. 필터링되는 코딩된 유닛들 (CUs)에 대해, 인코더는 픽셀 단위 또는 그룹 단위로 필터링을 수행할 수 있으며, 여기서, 그룹은 예를 들어, 2 x 2 픽셀들의 블록 또는 4 x 4 픽셀들의 블록일 것이다.
- [0009] 일 예에서, 비디오 데이터를 디코딩하는 방법은, 범위 조합들의 필터들에 맵핑을 생성하는 단계로서, 그 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하는, 범위 조합들의 필터들에 맵핑을 생성하는 단계; 현재 범위 조합이 이전 범위 조합과 동일한 필터에 맵핑되는 것을 시그널링하는 제 1 코드워드를 수신하는 것에 응답하여, 현재 범위 조합을 동일한 필터에 맵핑하는 단계; 및 현재 범위 조합이 이전 범위 조합과 상이한 필터에 맵핑되는 것을 시그널링하는 제 2 코드워드를 수신하는 것에 응답하여, 현재 범위 조합을 새로운 필터에 맵핑하는 단계를 포함한다.
- [0010] 다른 예에서, 비디오 데이터를 코딩하는 장치는, 범위 조합들의 필터들에 맵핑을 생성하는 것으로서, 그 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하는, 범위 조합들의 필터들에 맵핑을 생성하고; 현재 범위 조합이 이전 범위 조합과 동일한 필터에 맵핑되는 것을 시그널링하는 제 1 코드워드를 수신하는 것에 응답하여, 현재 범위 조합을 이전 범위 조합과 동일한 필터에 맵핑하며; 그리고 현재 범위 조합이 이전 범위 조합과 상이한 필터에 맵핑되는 것을 시그널링하는 제 2 코드워드를 수신하는 것에 응답하여, 현재 범위 조합을 제 2 코드워드에 의해 식별되는 필터에 맵핑하도록 구성되는 비디오 디코더를 포함한다.
- [0011] 또다른 예에서, 장치는, 범위 조합들의 필터들에 맵핑을 생성하는 수단으로서, 그 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하는, 범위 조합들의 필터들에 맵핑을 생성하는 수단; 현재 범위 조합이 이전 범위 조합과 동일한 필터에 맵핑되는 것을 시그널링하는 제 1 코드워드를 수신하는 것에 응답하여, 현재의 범위 조합을 이전 범위 조합과 동일한 필터에 맵핑하는 수단; 및 현재 범위 조합이 이전 범위 조합과 상이한 필터에 맵핑되는 것을 시그널링하는 제 2 코드워드를 수신하는 것에 응답하여, 현재 범위 조합을 제 2 코드워드에 의해 식별되는 필터에 맵핑하는 수단을 포함한다.
- [0012] 또다른 예에서, 컴퓨터-판독가능 저장 매체는 명령들을 저장하며, 그 명령들은, 실행될 경우, 하나 이상의 프로세서들로 하여금, 범위 조합들의 필터들에 맵핑을 생성하게 하는 것으로서, 그 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하는, 범위 조합들의 필터들에 맵핑을 생성하게 하고; 현재 범위 조합이 이전 범위 조합과 동일한 필터에 맵핑되는 것을 시그널링하는 제 1 코드워드를 수신하는 것에 응답하여, 현재 범위 조합을 이전 범위 조합과 동일한 필터에 맵핑하게 하며; 그리고 현재 범위 조합이 이전 범위 조합과 상이한 필터에 맵핑되는 것을 시그널링하는 제 2 코드워드를 수신하는 것에 응답하여, 현재 범위 조합을 제 2 코드워드에 의해 식별되는 필터에 맵핑하게 한다.
- [0013] 또다른 예에서, 비디오 데이터를 인코딩하는 방법은, 범위 조합들의 필터들에 맵핑을 결정하는 단계로서, 그 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하는, 범위 조합들의 필터들에 맵핑을 결정하는 단계; 현재 범위 조합이 이전 범위 조합과 동일한 필터에 맵핑될 경우, 제 1 코드워드를 생성하는 단계; 및 현재 범위 조합이 이전 범위 조합과 상이한 필터에 맵핑될 경우, 제 2 코드워드를 생성하는 단계를 포함하며, 여기서 제 2 코드워드는 현재 범위 조합에 맵핑된 필터를 식별한다.
- [0014] 또다른 예에서, 비디오 데이터를 코딩하는 장치는, 범위 조합들의 필터들에 맵핑을 결정하는 것으로서, 그 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하는, 범위 조합들의 필터들에 맵핑을 결정하고; 현재 범위 조합이 이전 범위 조합과 동일한 필터에 맵핑될 경우, 제 1 코드워드를 생성하며; 그리고 현재 범위 조합이 이전 범위 조합과 상이한 필터에 맵핑될 경우, 제 2 코드워드를 생성하도록 구성된 비디오 인코더를 포함하며, 여기서 제 2 코드워드는 현재 범위 조합에 맵핑된 필터를 식별한다.
- [0015] 또다른 예에서, 장치는 범위 조합들의 필터들에 맵핑을 결정하는 수단으로서, 그 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하는, 범위 조합들의 필터들에 맵핑을 결정하는 수단; 현재 범위

조합이 이전 범위 조합과 동일한 필터에 맵핑될 경우, 제 1 코드워드를 생성하는 수단; 및 현재 범위 조합이 이전 범위 조합과 상이한 필터에 맵핑될 경우, 제 2 코드워드를 생성하는 수단을 포함하며, 여기서 제 2 코드워드는 현재 범위 조합에 맵핑된 필터를 식별한다.

[0016]

또다른 예에서, 컴퓨터-판독가능 저장 매체는 명령들을 저장하며, 그 명령들은 실행될 경우, 하나 이상의 프로세서들로 하여금, 범위 조합들의 필터들에의 맵핑을 결정하게 하는 것으로서, 그 범위 조합은 제 1 메트릭에 대한 범위와 제 2 메트릭에 대한 범위를 포함하는, 범위 조합들의 필터들에의 맵핑을 결정하게 하고; 현재 범위 조합이 이전 범위 조합과 동일한 필터에 맵핑될 경우, 제 1 코드워드를 생성하게 하며; 그리고 현재 범위 조합이 이전 범위 조합과 상이한 필터에 맵핑될 경우, 제 2 코드워드를 생성하게 하며, 여기서 제 2 코드워드는 현재 범위 조합에 맵핑된 필터를 식별한다.

[0017]

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

[0018]

도 1 은 예시적인 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.

도 2a 및 도 2b 는 최대 코딩 유닛 (LCU) 에 적용되는 쿼드트리 파티셔닝의 일 예를 예시하는 개념도들이다.

도 2c 및 도 2d 는 도 2a 및 도 2b 의 예시적인 쿼드트리 파티셔닝에 대응하는 비디오 블록들의 시리즈에 대한 필터 맵의 일 예를 예시하는 개념도들이다.

도 3 은 본 개시물과 부합하는 예시적인 비디오 인코더를 예시하는 블록도이다.

도 4a 는 필터들에의 2개의 메트릭들에 대한 범위들의 맵핑을 예시하는 개념도이다.

도 4b 는 필터들에의 활동 메트릭 및 방향 메트릭에 대한 범위들의 맵핑을 예시하는 개념도이다.

도 5 는 본 개시물과 부합하는 예시적인 비디오 디코더를 예시하는 블록도이다.

도 6a, 도 6b 및 도 6c 는 4 x 4 픽셀들의 블록의 개념도들을 나타낸다.

도 7 은 본 개시물과 부합하는 코딩 기법들을 예시하는 흐름도이다.

도 8a 및 도 8b 는 본 개시물과 부합하는 코딩 기법들을 예시하는 흐름도들이다.

도 9a 및 도 9b 는 본 개시물과 부합하는 코딩 기법들을 예시하는 흐름도들이다.

도 10 은 본 개시물과 부합하는 코딩 기법들을 예시하는 흐름도이다.

도 11 은 본 개시물과 부합하는 코딩 기법들을 예시하는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0019]

본 개시물은 비디오 인코딩 및/또는 비디오 디코딩 프로세스에서 비디오 데이터의 필터링과 연관되는 기법들을 설명한다. 본 개시물에 따르면, 필터링이 인코더에서 적용되며, 필터 정보는 디코더로 하여금 인코더에서 적용되었던 필터링을 식별할 수 있도록, 비트스트림으로 인코딩된다. 디코더는 필터 정보를 포함하는 인코딩된 비디오 데이터를 수신하며, 비디오 데이터를 디코딩하고, 그리고 필터링 정보에 기초하여 필터링을 적용한다. 이러한 방법으로, 디코더는 인코더에서 적용되었던 동일한 필터링을 적용한다. 본 개시물의 기법들에 따르면, 프레임 단위, 슬라이스 단위, 또는 LCU 단위, 인코더는 필터들의 하나 이상의 세트들을 선택할 수도 있으며, 코딩된-유닛 단위로, 인코더는 필터링을 적용할지 여부를 결정할 수도 있다. 필터링되는 코딩된 유닛들 (CUs) 에 대해, 인코더는 픽셀 단위 또는 그룹 단위로 필터링을 수행할 수 있으며, 여기서, 그룹은 예를 들어, 2 x 2 픽셀들의 블록 또는 4 x 4 픽셀들의 블록일 것이다.

[0020]

본 개시물의 기법들에 따르면, 비디오 데이터는 코딩된 유닛들 (CUs) 로서 지칭되는 유닛들로 코딩될 수 있다.

CUs 는 더 작은 CUs, 또는 서브-유닛들로, 쿼드트리 파티셔닝 방식을 이용하여 파티셔닝될 수 있다. 특정의 CU 에 대한 쿼드트리 파티셔닝 방식을 식별하는 구문이 인코더로부터 디코더로 송신될 수 있다. 주어진 CU 의 각각의 서브-유닛과 연관되는 다수의 입력들이, 인코딩된 비디오 데이터를 디코딩하고 재구성하는 프로세스 동안 필터링될 수 있다. 본 개시물의 기법들에 따르면, 필터 기술 구문은 얼마나 많은 필터들이 그 세트 내에 있는지 또는 그 필터들이 어떤 형태를 취하는지와 같이, 필터들의 세트를 기술할 수 있다. 디코

더에 의해 수신된 비트스트림에서 추가적인 구문이 특정의 서브-유닛에 대해 인코더에서 사용되는 필터들 (즉, 필터 계수들) 을 식별할 수 있다. 특정의 입력에 대해 사용되는 필터가 2개 이상의 메트릭들에 기초하여 선택될 수 있으며, 여기서, 2개 이상의 메트릭들에 대한 값들의 특정 조합들이 필터들의 세트 내 특정의 필터들에 인덱스된다. 다른 경우, 2개 이상의 메트릭들이 결합되어 단일 메트릭을 형성할 수도 있다. 메트릭들에 의 필터들의 맵핑이 또한 비트스트림으로 시그널링될 수 있다.

[0021]

상이한 유형들의 필터링이 비디오 데이터에 대해 결정된 2개 이상의 메트릭들에 기초하여 픽셀들 또는 픽셀들의 블록들에 적용될 수도 있다. 특정의 픽셀에 대해 사용되는 필터는 2개 이상의 메트릭들, 예컨대 활동 메트릭과 방향 메트릭의 일부 조합에 기초하여 선택될 수 있다. 활동 메트릭은, 예를 들어, 비디오 데이터 내 픽셀들의 하나 이상의 블록들과 연관되는 활동을 정량화할 수도 있다. 활동 메트릭은 픽셀들의 세트 내 픽셀 분산 (pixel variance) 을 나타내는 분산 메트릭 (variance metric) 을 포함할 수도 있다. 활동 메트릭은 방향-특정적이거나 또는 비-방향-특정적일 수도 있다. 예를 들어, 비-방향-특정의 활동 메트릭은 아래에서 더욱더 자세하게 설명하는 바와 같이 합산-수정된 라플라시안 (Laplacian) 값을 포함할 수도 있다.

[0022]

방향-특정의 활동 메트릭들의 예들은 수평 활동 메트릭, 수직 활동 메트릭, 45 도 활동 메트릭, 및 135 도 활동 메트릭을 포함한다. 방향 메트릭은 픽셀 또는 픽셀들의 그룹의, 수평 활동, 수직 활동, 또는 대각선 활동 중 임의의 활동을 픽셀들의 블록에 대해 정량화할 수도 있거나, 또는 방향 메트릭은 수평 활동, 수직 활동, 및/또는 대각선 활동의 비교를 포함할 포함할 수도 있으며, 여기서, 수평 활동은 일반적으로 수평 방향에서 픽셀 값들의 변화들을 지칭하며, 수직 활동은 일반적으로 수직 방향에서 픽셀 값들의 변화들을 지칭하며, 대각선 활동은 일반적으로 대각선 방향에서 픽셀 값들의 변화들을 지칭한다.

[0023]

본 개시물의 기법들에 따르면, 픽셀들의 블록에 대한 필터를 결정할 때, 블록 내 픽셀들의 서브세트가 인코딩 및 디코딩 복잡성을 감소시키기 위해 사용될 수도 있다. 예를 들어, 4 x 4 픽셀들의 블록에 대한 필터를 결정할 때, 4 x 4 블록의 모든 16개의 픽셀들을 사용하는 것이 필요하지 않을 수도 있다. 게다가, 본 개시물의 기법들에 따르면, 코딩중인 현재의 블록 내로부터의 픽셀들의 서브세트는, 이웃하는 블록들의 픽셀 값들이 아닌, 단지 현재의 블록의 픽셀 값들을 이용하여 메트릭들이 계산되도록, 선택될 수 있다. 예를 들어, 평가중인 픽셀에 대한 메트릭은 픽셀을 인접한 픽셀들과 비교하는 것에 기초하여 계산될 것이다. 일부의 경우, 평가중인 픽셀에 대한 인접한 픽셀들 중 하나 이상의 픽셀들은 평가중인 픽셀과는 상이한 블록에 있을 것이다. 다른 경우, 그러나, 픽셀에 대한 인접한 픽셀들 중 하나 이상의 픽셀들은 그 픽셀과 동일한 블록에 있을 것이다. 본 개시물의 기법들에 따르면, 픽셀들의 서브세트는 이웃하는 블록들에서 인접한 픽셀들을 갖지 않는 픽셀들을 포함하도록 선택될 수 있다. 이에 추가적으로 또는 대안적으로, 픽셀들의 서브세트는 이웃하는 블록들에서 인접한 픽셀들을 갖는 픽셀들을 포함할 수도 있지만, 이웃하는 블록들에서 그들 인접한 픽셀들은 메트릭을 결정할 때 사용되지 않을 수도 있다. 특정의 메트릭의 결정을 이웃하는 블록들의 픽셀들이 아닌, 현재의 블록 내 픽셀들에 기초함으로써, 인코더 및/또는 디코더에서의 버퍼들에 대한 필요성은 일부 경우, 감소되거나 또는 심지어 제거될 수도 있다.

[0024]

일부의 경우, 본 개시물의 기법들에 따르면, 코딩중인 현재의 블록 내로부터의 픽셀들의 서브세트는, 메트릭들이 상부 이웃하는 블록들 또는 하부 이웃하는 블록들의 픽셀 값들이 아닌, 오직 현재의 블록 및 좌측 및 우측 이웃하는 블록들의 픽셀 값들을 이용하여 계산되도록, 선택될 수 있다. 비디오 블록들을 코딩할 때에 사용되는 래스터 스캔 순서의 결과로서, 상부 및 하부 이웃하는 블록들에 대한 라인 버퍼들은 좌측 및 우측 이웃하는 블록들의 픽셀 값들을 저장하기 위한 라인 버퍼들보다 훨씬 더 많은 픽셀 값들을 저장하는 것을 필요로 하는 경향이 있다.

[0025]

본 개시물의 기법들에 따르면, 적응형 인-루프 필터와 같은 필터 유닛은 멀티-메트릭 필터 맵핑에 기초하여 다수의 필터들을 이용하도록 구성될 수 있다. 다수의 필터들이 단일 입력 또는 다수의 입력들과 함께 사용될 수도 있다. 이하에서 좀더 자세하게 설명하는 바와 같이, 본 개시물에서 설명되는 다수의 입력들은 일반적으로 인코딩 및 디코딩 프로세스들 동안 생성되는 중간 비디오 블록 데이터 또는 이미지 데이터를 지칭한다. 주어진 비디오 블록과 연관되는 다수의 입력들은 예를 들어, 재구성된 블록 또는 이미지 (RI), 사전-디블록킹된 재구성된 블록 또는 이미지 (pRI), 예측 블록 또는 이미지 (PI), 및/또는 양자화된 예측 에러 이미지 (EI) 를 포함할 수 있다. 단일 입력 방식에서, 필터가 상기 입력들 중 하나, 예컨대 RI 에 오직 적용될 수도 있다. 또한, 아래에 더욱더 자세하게 설명하는 바와 같이, 본 개시물의 필터링 기법들은 쿼드트리 파티셔닝 방식을 이용하여 여러 사이즈들의 CUs 에 적용될 수 있다. 쿼드트리 파티셔닝 방식을 이용하여 파티셔닝된 CUs 에 대한 멀티-메트릭 필터 맵핑에 다수의 필터들을 이용함으로써, 비디오 코딩 성능이, 압축 레이트 및 재

구성된 비디오 품질 중 하나 또는 양자에 의해 측정될 때, 향상될 것이다.

[0026] 위에서 설명한 멀티-메트릭 필터링 기법들을 구현하기 위해, 인코더는 생성, 업데이트, 저장, 또는 다른 수단에 의해, 필터들에의 범위들의 조합들의 맵핑을 유지한다. 일 예로서, 제 1 메트릭에 대한 제 1 범위와 제 2 메트릭에 대한 제 1 범위의 조합이 제 1 필터에 맵핑할 수도 있다. 제 1 메트릭에 대한 제 1 범위와 제 2 메트릭에 대한 제 2 범위의 조합이 또한 제 1 필터에 맵핑하거나 또는 제 2 필터에 맵핑할 수도 있다. 제 1 메트릭이 8개의 범위들을 갖고 제 2 메트릭이 4개의 범위들을 가지면, 예를 들어, 제 1 및 제 2 메트릭은 32개의 범위들의 조합들을 가질 수 있으며, 32개의 조합들 각각이 필터에 맵핑될 수 있다. 그러나, 각각의 조합이 반드시 고유한 필터에 맵핑되는 것은 아니다. 따라서, 32개의 조합들이 4개의 필터들, 8개의 필터들, 10개의 필터들, 또는 어떤 다른 개수의 필터들에 맵핑될 것이다. 인코더와 동일한 필터들을 적용하기 위해, 디코더는 또한 필터들에의 범위 조합들의 동일한 맵핑들을 유지할 수도 있다.

[0027] 본 개시물은 인코더로부터 디코더로, 인코딩된 비트스트림으로, 필터들에의 범위 조합들의 맵핑을 시그널링하는 기법들을 기술한다. 맵핑은 예를 들어, 각각의 범위 조합을 필터 식별 (ID) 과 연관시킬 수도 있다. 이 맵핑을 시그널링하는 하나의 간단한 방법은 각각의 필터 ID 에 대한 하나의 코드워드를 이용하고, 따라서, 각각의 범위들의 조합에 대해, 대응하는 필터 ID 의 코드워드를 전송하는 것이다. 그러나, 이 기법은 일반적으로 비효율적이다. 본 개시물의 기법들은 차분 코딩 방법들을 이용함으로써 그 맵핑 내 상관관계를 활용할 수도 있다. 공통 범위를 공유하는 범위들의 조합들은 종종 동일한 필터를 사용한다. 일 예로서, 제 1 메트릭에 대한 제 1 범위와 제 2 메트릭에 대한 제 1 범위의 조합, 및 제 1 메트릭에 대한 제 1 범위와 제 2 메트릭에 대한 제 2 범위의 조합은 공통 범위 (제 1 메트릭의 제 1 범위) 를 공유한다. 따라서, 이들 2개의 조합들은 일부 경우, 동일한 필터 ID 에 맵핑할 것이다. 이 상관을 이용함으로써, 본 개시물의 기법들은 필터 ID들에의 범위 조합들의 맵핑을 인코더로부터 디코더로 시그널링하는데 요구되는 비트수를 감소시킬 수도 있다.

[0028] 필터 ID들에의 범위 조합들의 맵핑을 시그널링하는 것에 더해서, 본 개시물은 또한 필터들에 대한 필터 계수들을 인코딩된 비트스트림으로 시그널링하는 기법들을 기술한다. 본 개시물의 기법들은 필터 계수들을 인코더로부터 디코더로 시그널링하는데 차분 코딩 방법들 (differential coding methods) 을 이용하는 것을 포함한다. 이 방법으로, 제 2 필터에 대한 필터 계수들은 차이 정보로서 디코더로 통신될 것이며, 여기서, 차이 정보는 제 2 필터의 필터 계수들을 생성하는 방법으로 제 1 필터의 필터 계수들을 변경하는 방법을 기술한다. 차분 코딩 기법들은 제 1 및 제 2 필터의 필터 계수들이 덜 유사한 경우에 비해 제 1 및 제 2 필터의 필터 계수들이 더 유사한 경우에 더 효과적일 수도 있다 (즉, 더 많은 비트들의 절감들을 초래할 수도 있다). 본 개시물의 기법들은 필터들에 대한 필터 계수들을 시그널링하는 순차적인 순서를 결정하는 것을 포함한다. 본 개시물에서 설명하는 기법들을 이용하여 결정된 순서 (ordering) 들은 필터 계수들의 향상된 차분 코딩을 초래할 수도 있으며, 따라서, 일부 경우, 필터 계수들을 시그널링할 때 비트들의 절감들을 초래할 수도 있다.

[0029] 본 개시물의 기법들이 때로는 인-루프 필터링을 참조하여 설명될 수도 있지만, 이 기법들은 인-루프 필터링 (in-loop filtering), 사후-루프 필터링 (post-loop filtering), 및 다른 필터링 방식들, 예컨대 스위칭 필터링 (switched filtering) 에 적용될 수도 있다. 인-루프 필터링은, 일반적으로 필터링된 데이터가 예측 인트라- 또는 인터-코딩용으로 사용되는 경우에 필터링된 데이터가 인코딩 및 디코딩 루프들의 부분인 필터링을 지칭한다. 사후-루프 필터링은 인코딩 루프 이후, 재구성된 비디오 데이터에 적용되는 필터링을 지칭한다. 사후-루프 필터링에 의하면, 필터링된 데이터와 반대로, 미필터링된 데이터가, 예측 인트라- 또는 인터-코딩용으로 사용된다. 일부 구현예들에서, 필터링의 유형은 예를 들어, 프레임 단위, 슬라이스 단위, 또는 다른 이런 단위로 사후-루프 필터링과 인-루프 필터링 사이에 스위칭될 수도 있으며, 사후-루프 필터링 또는 인-루프 필터링을 사용할지의 결정이 각각의 프레임, 슬라이스 등에 대해 인코더로부터 디코더로 시그널링될 수 있다. 본 개시물의 기법들은 인-루프 필터링 또는 사후 필터링에 한정되지 않으며, 비디오 코딩 동안 적용되는 광범위한 필터링에 적용될 수도 있다.

[0030] 본 개시물에서, 용어 "코딩" 은 인코딩 또는 디코딩을 지칭한다. 이와 유사하게, 용어 "코더 (coder)" 는 일반적으로 임의의 비디오 인코더, 비디오 디코더, 또는 결합된 인코더/디코더 (코덱) 를 지칭한다. 따라서, 용어 "코더" 는 본원에서 비디오 인코딩 또는 비디오 디코딩을 수행하는 특수화된 컴퓨터 디바이스 또는 장치를 지칭하기 위해 사용된다.

[0031] 게다가, 본 개시물에서, 용어 "필터" 는 일반적으로 필터 계수들의 세트를 지칭한다. 예를 들어, 3 x 3 필터는 9개의 필터 계수들의 세트로 정의될 수도 있으며, 5 x 5 필터는 25개의 필터 계수들의 세트로 정의될 수도 있으며, 9 x 5 필터는 45개의 필터 계수들의 세트로 정의될 수도 있다. 용어 "필터들의 세트" 는 일반적인

로 하나 보다 많은 필터의 그룹을 지칭한다. 예를 들어, 2개의 3×3 필터들의 세트는, 9개의 필터 계수들의 제 1 세트, 및 9개의 필터 계수들의 제 2 세트를 포함할 것이다. 프레임, 슬라이스, 또는 최대 코딩 유닛 (LCU) 과 같은 비디오 블록들의 시리즈에 대한, 본 개시물에서 설명되는 기법들에 따르면, 필터들의 세트들을 식별하는 정보가 비디오 블록들의 시리즈에 대한 헤더에서 인코더로부터 디코더로 시그널링된다. "필터 서포트" 로서 종종 지칭되는, 용어 "형태 (shape)" 는, 일반적으로 특정의 필터에 대하여 필터 계수들의 로우들의 개수 및 필터 계수들의 칼럼들 개수를 지칭한다. 예를 들어, 9×9 는 제 1 형태의 일 예이고, 9×5 는 제 2 형태의 일 예이고, 5×9 는 제 3 형태의 일 예이다. 일부의 경우, 필터들은 다이아몬드-형태들, 다이아몬드-유사 형태들, 원형의 형태들, 원형-유사 형태들, 6각형 형태들, 8각형 형태들, 십자가 형태들, X-형태들, T-형태들, 다른 기하학적 형태들, 또는 매우 많은 다른 형태들 또는 구성을 포함한, 비-직사각형의 형태들 (non-rectangular shapes) 을 취할 수도 있다.

[0032]

도 1 은 본 개시물의 기법들을 구현할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (110) 을 예시하는 블록도이다. 도 1 에 나타난 바와 같이, 시스템 (110) 은 인코딩된 비디오 데이터를 통신 채널 (115) 을 통해서 목적지 디바이스 (116) 로 송신하는 소스 디바이스 (112) 를 포함한다. 소스 디바이스 (112) 및 목적지 디바이스 (116) 는 광범위한 디바이스들 중 임의의 디바이스를 포함할 수도 있다. 일부의 경우, 소스 디바이스 (112) 및 목적지 디바이스 (116) 는 무선 통신 디바이스 핸드셋들, 예컨대 소위 셀룰러 또는 위성 무선 전화기들을 포함할 수도 있다. 그러나, 비디오 데이터의 필터링에 좀더 일반적으로 적용하는 본 개시물의 기법들은 무선 애플리케이션들 또는 설정들에 반드시 한정되지 않으며, 비디오 인코딩 및/또는 디코딩 능력들을 포함하는 비-무선 디바이스들에 적용될 수도 있다.

[0033]

도 1 의 예에서, 소스 디바이스 (112) 는 비디오 소스 (120), 비디오 인코더 (122), 변조기/복조기 (모뎀) (123) 및 송신기 (124) 를 포함한다. 목적지 디바이스 (116) 는 수신기 (126), 모뎀 (127), 비디오 디코더 (128), 및 디스플레이 디바이스 (130) 를 포함한다. 본 개시물에 따르면, 소스 디바이스 (112) 의 비디오 인코더 (122) 는 비디오 블록 필터링 프로세스에서 다수의 입력들에 대한 하나 이상의 필터 계수들의 세트들을 선택하고, 그 후 그 선택된 하나 이상의 필터 계수들의 세트들을 인코딩하도록 구성될 수도 있다. 하나 이상의 필터 계수들의 세트들로부터의 특정의 필터들이 하나 이상의 입력들에 대한 하나 이상의 메트릭들에 기초하여 선택될 수도 있으며, 필터 계수들은 하나 이상의 입력들을 필터링하는데 사용될 수도 있다. 본 개시물의 필터링 기법들은 일반적으로 필터 계수들을 인코딩된 비트스트림으로 코딩하거나 또는 시그널링하는 임의의 기법들과 호환가능하다.

[0034]

본 개시물의 기법들에 따르면, 비디오 인코더 (122) 를 포함하는 디바이스는 비디오 디코더 (128) 를 포함하는 디바이스로, 프레임 또는 슬라이스와 같은 비디오 블록들의 시리즈에 대한 하나 이상의 필터 계수들의 세트들을 시그널링할 수 있다. 비디오 블록들의 시리즈에 대해, 비디오 인코더 (122) 는 예를 들어, 모든 입력들과 함께 사용되는 하나의 필터들의 세트를 시그널링할 수도 있거나, 또는 다수의 입력들과 함께 사용되는 다수의 필터들의 세트들 (예를 들어, 입력 당 하나의 세트) 을 시그널링할 수도 있다. 그러면, 각각의 비디오 블록 또는 비디오 블록들의 시리즈 내 CU 는, 필터들의 세트 중 어느 필터 또는 필터들이 그 비디오 블록의 각각의 입력에 대해, 또는 본 개시물의 기법들에 따라서, 사용되는지를 식별하는 추가적인 구문을 포함할 수 있으며, 필터들의 세트 중 어느 필터 또는 필터들이 사용되는 지는 하나 이상의 입력들과 연관되는 2개 이상의 메트릭들에 기초하여 결정될 수 있다.

[0035]

좀더 구체적으로는, 소스 디바이스 (112) 의 비디오 인코더 (122) 는 비디오 블록들의 시리즈에 대한 필터들의 하나 이상의 세트들을 선택하고, 인코딩 프로세스 동안 그 세트(들) 로부터 비디오 블록들의 시리즈의 CUs 와 연관되는 입력들의 픽셀들 또는 픽셀들의 그룹들에 필터들을 적용하고, 그후, 목적지 디바이스 (116) 의 비디오 디코더 (128) 로의 통신을 위해 필터들의 세트들 (즉, 필터 계수들의 세트들) 을 인코딩할 수도 있다. 비디오 인코더 (122) 는 그 필터들의 세트(들) 중 어느 필터(들) 를 그 특정의 CU 에 대한 픽셀들 또는 픽셀들의 그룹들과 함께 사용할 지를 선택하기 위해, 코딩된 CUs 의 입력들과 연관되는 하나 이상의 메트릭들을 결정할 수도 있다. 비디오 인코더 (122) 는 또한 비디오 디코더 (128) 로, 코딩된 비트스트림의 부분으로서, 필터들의 세트 내 필터들에의 범위들의 조합들의 맵핑을 시그널링할 수도 있다.

[0036]

디코더 측에서, 비디오 디코더 (128) 는 비트스트림 구문으로 수신된 필터 정보에 기초하여 필터 계수들을 결정할 수도 있다. 비디오 디코더 (128) 는 직접 디코딩 또는 예측 디코딩에 기초하여, 필터 계수들이 어떻게 인코딩되었는 지에 따라서 필터 계수들을 디코딩할 수도 있으며, 이 필터 계수들은 비트스트림 구문의 부분으로서 시그널링될 수도 있다. 게다가, 비트스트림은 필터들의 세트에 대한 필터들을 기술하는 필터 기술 구문 정보를 포함할 수도 있다. 필터 기술 구문에 기초하여, 디코더 (128) 는 인코더 (122) 로부터 수신된 추가

정보에 기초하여 필터 계수들을 재구성할 수 있다. 도 1의 예시된 시스템 (110)은 단지 예시적이다. 본 개시물의 필터링 기법들은 임의의 인코딩 또는 디코딩 디바이스들에 의해 수행될 수도 있다. 소스 디바이스 (112) 및 목적지 디바이스 (116)는 단지 이런 기법들을 지원하는 코딩 디바이스들의 예들이다. 비디오 디코더 (128)는 또한 비트스트림 구문으로 수신된 필터 정보에 기초하여 필터들에의 범위들의 조합들의 맵핑을 결정할 수도 있다.

[0037]

소스 디바이스 (112)의 비디오 인코더 (122)는 비디오 소스 (120)로부터 수신된 비디오 데이터를 본 개시물의 기법들을 이용하여 인코딩할 수도 있다. 비디오 소스 (120)는 비디오 캡처 디바이스, 예컨대 비디오 카메라, 이전에 캡처된 비디오를 포함하는 비디오 아카이브, 또는 비디오 콘텐츠 제공자로부터의 비디오 공급을 포함할 수도 있다. 추가 대안적인 예로서, 비디오 소스 (120)는 컴퓨터 그래픽스-기반의 데이터를 소스 비디오, 또는 라이브 비디오, 아카이브된 비디오, 및 컴퓨터 생성된 비디오의 조합으로서 발생할 수도 있다. 일부의 경우, 비디오 소스 (120)가 비디오 카메라이면, 소스 디바이스 (112) 및 목적지 디바이스 (116)는 소위 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 각 경우에, 캡처된, 사전-캡처된 또는 컴퓨터-생성된 비디오는 비디오 인코더 (122)에 의해 인코딩될 수도 있다.

[0038]

일단 비디오 데이터가 비디오 인코더 (122)에 의해 인코딩되면, 그 인코딩된 비디오 정보는 그후 모뎀 (123)에 의해 예컨대, 코드분할 다중접속 (CDMA), 주파수 분할 다중접속 (FDMA), 직교 주파수 분할 멀티플렉싱 (OFDM), 또는 임의의 다른 통신 표준 또는 기법과 같은 통신 표준에 따라서 변조되어, 송신기 (124)를 통해서 목적지 디바이스 (116)로 송신될 수도 있다. 모뎀 (123)은 여러 믹서들, 필터들, 증폭기들 또는 신호 변조용으로 설계된 다른 구성요소들을 포함할 수도 있다. 송신기 (124)는 증폭기들, 필터들, 및 하나 이상의 안테나들을 포함한, 데이터를 송신하도록 설계된 회로들을 포함할 수도 있다.

[0039]

목적지 디바이스 (116)의 수신기 (126)는 채널 (115)을 통해서 정보를 수신하고, 모뎀 (127)은 그 정보를 복조한다. 비디오 디코더 (128)에 의해 수행된 비디오 디코딩 프로세스는 필터링을, 예컨대, 인-루프 디코딩의 부분으로서 또는 디코딩 루프 다음의 사후 필터링 단계로서 포함할 수도 있다. 어느 방법이든, 특정의 슬라이스 또는 프레임에 대해 비디오 디코더 (128)에 의해 적용되는 필터들의 세트는 본 개시물의 기법들을 이용하여 디코딩될 수도 있다. 디코딩된 필터 정보는 코딩된 비트스트림에서 필터 기술 구문을 식별하는 것을 포함할 수도 있다. 예를 들어, 필터 계수들에 대해 예측 코딩이 사용되면, 상이한 필터 계수들 사이의 유사성들 (similarities)이 채널 (115)을 통해서 운반되는 정보의 양을 감소시키기 위해 이용될 수도 있다. 특히, 필터 (즉, 필터 계수들의 세트)는 상이한 필터와 연관되는 또다른 필터 계수들의 세트에 상대적인 차이 값들로서 예측 코딩될 수 있다. 상이한 필터는 예를 들어, 상이한 슬라이스 또는 프레임과 연관될 수도 있다. 이런 경우, 비디오 디코더 (128)는 비디오 블록들 및 그 상이한 필터가 연관되는 상이한 프레임 또는 슬라이스를 식별하는 필터 정보를 포함하는 인코딩된 비트스트림을 수신할 것이다. 필터 정보는 또한 상이한 CU의 필터에 상대적인, 현재의 필터를 정의하는 차이 값들을 포함한다. 특히, 차이 값들은 상이한 CU에 대해 사용되는 상이한 필터의 필터 계수들에 상대적인, 현재의 필터에 대한 필터 계수들을 정의하는 필터 계수 차이 값들을 포함할 수도 있다.

[0040]

비디오 디코더 (128)는 비디오 블록들을 디코딩하고, 필터 계수들을 생성하고, 생성된 필터 계수들에 기초하여 디코딩된 비디오 블록들을 필터링한다. 비디오 디코더 (128)는 필터 계수들을 비트스트림으로부터 추출된 필터 기술 구문에 기초하여 생성할 수 있다. 디코딩된 및 필터링된 비디오 블록들은 비디오 프레임들로 조합되어 디코딩된 비디오 데이터를 형성할 수 있다. 디스플레이 디바이스 (130)는 그 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 음극선관 (CRT), 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 또다른 유형의 디스플레이 디바이스와 같은, 다양한 디스플레이 디바이스들 중 임의의 디바이스를 포함할 수도 있다.

[0041]

통신 채널 (115)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 (115)은 근거리 네트워크, 광역 네트워크, 또는 글로벌 네트워크, 예컨대 인터넷과 같은 패킷-기반 네트워크의 부분을 형성할 수도 있다. 통신 채널 (115)은 일반적으로 비디오 데이터를 소스 디바이스 (112)로부터 목적지 디바이스 (116)로 송신하는, 임의의 적합한 통신 매체, 또는 상이한 통신 매체들의 컬렉션을 나타낸다. 다시, 도 1은 단지 예시적이며, 본 개시물의 기법들은 인코딩 디바이스와 디코딩 디바이스 사이의 임의의 데이터 통신을 필수적으로 포함하지 않는 비디오 코딩 설정들 (예컨대, 비디오 인코딩 또는 비디오 디코딩)에 적용할 수도 있다. 다른 예들에서, 데이터는 로컬 메모리로부터 추출되거나, 네트워크를 통해서 스트리밍되거나,

또는 기타 등등이 이루어질 수 있다.

[0042] 이의 대안으로, 인코딩된 데이터는 비디오 인코더 (122)로부터 저장 디바이스 (132)로 출력될 수도 있다. 이와 유사하게, 인코딩된 데이터는 저장 디바이스 (132)로부터 비디오 디코더 (128)에 의해 액세스될 수도 있다. 저장 디바이스 (132)는 하드 드라이브, 블루-레이 디스크들, DVDs, CD-ROMs, 플래시 메모리, 휘발성 또는 비-휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체들과 같은 다양한 분산된 또는 로컬로 액세스되는 데이터 저장 매체들 중 임의의 데이터 저장 매체를 포함할 수도 있다. 추가 예에서, 저장 디바이스 (132)는 소스 디바이스 (112)에 의해 생성된 인코딩된 비디오를 유지할 수도 있는 파일 서버 또는 다른 중간 저장 디바이스에 대응할 수도 있다. 목적지 디바이스 (116)는 저장 디바이스 (132)로부터 스트리밍 또는 다운로드를 통해서, 저장된 비디오 데이터에 액세스할 수도 있다. 파일 서버는 인코딩된 비디오 데이터를 저장하고 그 인코딩된 비디오 데이터를 목적지 디바이스 (116)로 송신하는 것이 가능한 임의 종류의 서버일 수도 있다. 예시적인 파일 서버들은 웹 서버 (예컨대, 웹사이트용), FTP 서버, NAS (network attached storage) 디바이스들, 또는 로컬 디스크 드라이브를 포함한다. 목적지 디바이스 (116)는 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해서, 인코딩된 비디오 데이터에 액세스할 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 양자의 조합을 포함할 수도 있다. 저장 디바이스 (132)로부터의 인코딩된 비디오 데이터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이들 양자의 조합일 수도 있다.

[0043] 본 개시물의 기법들은 그러나, 반드시 무선 애플리케이션들 또는 설정들에 한정되지는 않는다. 이 기법들은 공중 경유 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 (110)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.

[0044] 비디오 인코더 (122) 및 비디오 디코더 (128)는 설명의 목적들을 위해서 본 개시물의 부분들에 사용되는, MPEG 4, 파트 10, AVC (Advanced Video Coding)로서 대안적으로 지칭되는 ITU-T H.264 표준과 같은, 비디오 압축 표준에 따라서 동작할 수도 있다. 그러나, 본 개시물의 기법들 중 많은 기법들이 새로운 차기 HEVC 표준을 포함한, 다양한 다른 비디오 코딩 표준들 중 임의의 표준에 용이하게 적용될 수도 있다. 일반적으로, 인코더 및 디코더에서 필터링을 가능하게 하는 어떤 표준도 본 개시물의 교시의 여러 양태들로부터 이득을 얻을 수도 있다.

[0045] 도 1에 나타내지는 않지만, 일부 양태들에서, 비디오 인코더 (122) 및 비디오 디코더 (128)는 오디오 인코더 및 디코더와 각각 통합될 수도 있으며, 오디오 및 비디오 양자의 인코딩을 공통 데이터 스트림 또는 별개의 데이터 스트림들로 처리하기 위해 적합한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능한 경우, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜, 또는 다른 프로토콜들, 예컨대, 사용자 데이터그램 프로토콜 (UDP)에 따를 수도 있다.

[0046] 비디오 인코더 (122) 및 비디오 디코더 (128) 각각은 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSPs), 주문형 집적회로들 (ASICs), 필드 프로그래밍가능 게이트 어레이들 (FPGAs), 이산 로직, 소프트웨어, 하드웨어, 펌웨어 또는 임의의 이들의 조합들로서 구현될 수도 있다. 비디오 인코더 (122) 및 비디오 디코더 (128) 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이들 중 어느 쪽이든, 각각의 모바일 디바이스, 가입자 디바이스, 브로드캐스트 디바이스, 서버 등에서, 결합된 인코더/디코더 (코덱)의 부분으로서 통합될 수도 있다.

[0047] 일부의 경우, 디바이스들 (112, 116)은 실질적으로 대칭적 방법으로 동작할 수도 있다. 예를 들어, 디바이스들 (112, 116)의 각각은 비디오 인코딩 및 디코딩 구성요소들을 포함할 수도 있다. 그러므로, 시스템 (110)은 예컨대, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 또는 비디오 전화 통신을 위해, 비디오 디바이스들 (112, 116) 사이에 1-방향 또는 2-방향 비디오 송신을 지원할 수도 있다.

[0048] 인코딩 프로세스 동안, 비디오 인코더 (122)는 다수의 코딩 기법들 또는 단계들을 실행할 수도 있다. 일반적으로, 비디오 인코더 (122)는 비디오 데이터를 인코딩하기 위해 개개의 비디오 프레임들 내 비디오 블록들에 대해 동작할 수도 있다. 일 예에서, 비디오 블록은 매크로블록 또는 매크로블록의 파티션에 대응할 수도 있

다. 매크로블록들은 ITU H.264 표준 및 다른 표준들에 의해 정의되는 비디오 블록의 한 유형이다. 매크로블록들은, 이 용어가 또한 $N \times N$ 또는 $N \times M$ 사이즈의 임의의 비디오 블록을 일반적으로 지칭하는데 종종 사용되지만, 일반적으로 16×16 데이터의 블록들을 지칭한다. ITU-T H.264 표준은 루마 성분들에 대해서 16×16 , 8×8 , 또는 4×4 , 그리고 크로마 성분들에 대해 8×8 과 같은 여러 블록 사이즈들에서 인트라 예측, 뿐만 아니라, 루마 성분들에 대해 16×16 , 16×8 , 8×16 , 8×8 , 8×4 , 4×8 및 4×4 , 그리고 크로마 성분들에 대해 대응하는 스케일링된 사이즈들과 같은 여러 블록 사이즈들에서 인터 예측을 지원한다. 본 개시물에서, " $N \times N$ " 은 수직 및 수평 치수들로 환산한, 블록의 픽셀 치수들, 예컨대, 16×16 픽셀들을 지칭한다.

일반적으로, 16×16 블록은 수직 방향으로 16 개의 픽셀들 및 수평 방향으로 16 개의 픽셀들을 가질 것이다. 이와 유사하게, $N \times N$ 블록은 일반적으로 수직 방향으로 N 개의 픽셀들 및 수평 방향으로 N 개의 픽셀들을 가지며, 여기서 N 은 양의 정수 값을 나타낸다. 블록에서 픽셀들은 로우들 및 칼럼들로 배열될 수도 있다.

[0049]

차기 HEVC 표준은 비디오 블록들에 대한 새로운 용어들을 정의한다. 특히, 비디오 블록들 (또는, 그의 파티션들) 은 "코딩 유닛들" (또는, CUs) 로서 지칭될 수도 있다. HEVC 표준에 의하면, 가장 큰 코딩된 유닛들 (LCUs) 은 쿼드트리 파티셔닝 방식에 따라서 더 작은 CUs 로 분할될 수도 있으며, 그 방식으로 정의된 상이한 CUs 는 소위 예측 유닛들 (PUs) 로 추가로 파티셔닝될 수도 있다. LCUs, CUs, 및 PUs 는 모두 본 개시물의 의미 내에서 비디오 블록들이다. 다른 유형들의 비디오 블록들이 또한 HEVC 표준 또는 다른 비디오 코딩 표준들에 부합하여, 사용될 수도 있다. 따라서, 어구 "비디오 블록들 (video blocks)" 은 임의의 사이즈의 비디오 블록을 지칭한다. 별개의 CUs 가 루마 성분들에 대해 포함되고 스케일링된 사이즈들이 주어진 픽셀에 대한 크로마 성분들에 대해 포함될 수도 있지만, 다른 칼라 공간들이 또한 사용될 수 있다.

[0050]

비디오 블록들은 고정 또는 가변 사이즈들을 가질 수도 있으며, 규정된 코딩 표준에 따라서 사이즈가 상이할 수도 있다. 각각의 비디오 프레임은 복수의 슬라이스들을 포함할 수도 있다. 각각의 슬라이스는 복수의 비디오 블록들을 포함할 수도 있으며, 이 매크로블록들은 파티션들에 배열되고, 또한 서브-블록들로서 지칭될 수도 있다. 위에서 언급하고 아래에서 좀더 자세히 설명하는 쿼드트리 파티셔닝 방식에 따르면, $N/2 \times N/2$ 의 제 1 CU 는 $N \times N$ LCU 의 서브-블록들을 포함할 수도 있으며, $N/4 \times N/4$ 의 제 2 CU 는 또한 제 1 CU 의 서브-블록들을 포함할 수도 있다. $N/8 \times N/8$ 의 PU 는 제 2 CU 의 서브-블록들을 포함할 수도 있다. 이와 유사하게, 추가 예로서, 16×16 미만인 블록 사이즈들은 16×16 비디오 블록의 파티션들로서 또는 16×16 비디오 블록의 서브-블록들로서 지칭될 수도 있다. 이와 유사하게, $N \times N$ 블록에 있어, $N \times N$ 미만인 블록 사이즈들은 $N \times N$ 블록의 파티션들 또는 서브-블록들로서 지칭될 수도 있다. 비디오 블록들은, 예컨대, 코딩된 비디오 블록들과 예측 비디오 블록들 사이의 픽셀 차이들을 나타내는 잔여 비디오 블록 데이터에의, 이산 코사인 변환 (DCT), 정수 변환, 웨이블릿 변환, 또는 개념적으로 유사한 변환과 같은 변환의 적용 후, 픽셀 도메인에서 픽셀 데이터의 블록들, 또는 그 변환 도메인에서 변환 계수들의 블록들을 포함할 수도 있다. 일부의 경우, 비디오 블록은 변환 도메인에서 양자화된 변환 계수들의 블록들을 포함할 수도 있다.

[0051]

비트스트림 내 구문 데이터는 프레임 또는 슬라이스에 대한 LCU 를 정의할 수도 있으며, 이 LCU 는 그 프레임 또는 슬라이스에 대한 픽셀들의 개수의 관점에서 최대 코딩 유닛이다. 일반적으로, LCU 또는 CU 는 LCUs 및 CUs 가 특정의 사이즈 구별을 갖지 않는다는 점을 제외하고는, H.264 에 따라서 코딩된 매크로블록과 유사한 목적을 갖는다. 대신, LCU 사이즈는 프레임 단위 또는 슬라이스 단위로 정의될 수 있으며, LCU 는 CUs 로 분할된다. 일반적으로, CU 에 대한 본 개시물에서의 언급들은 화상의 LCU 또는 LCU 의 서브-CU 를 지칭할 수도 있다. LCU 는 서브-CUs 로 분할될 수도 있으며, 각각의 서브-CU 는 서브-CUs 로 분할될 수도 있다. 비트스트림에 대한 구문 데이터는 CU 깊이로 지칭되는, LCU 가 분할되는 최대 횟수를 정의할 수도 있다. 따라서, 비트스트림은 또한 가장 작은 코딩 유닛 (SCU) 을 정의할 수도 있다. 본 개시물은 또한 LCU, CU, PU, SCU, 또는 TU 중 임의의 것을 지칭하기 위해 용어들 "블록" 및 "비디오 블록" 을 사용한다.

[0052]

위에서 소개한 바와 같이, LCU 는 쿼드트리 데이터 구조와 연관될 수도 있다. 일반적으로, 쿼드트리 데이터 구조는 CU 당 하나의 노드를 포함하며, 여기서, 루트 노드는 LCU 에 대응한다. CU 가 4개의 서브-CUs 로 분할되면, CU 에 대응하는 노드는 4개의 리프 노드들을 포함하며, 각각의 리프 노드는 서브-CUs 중 하나에 대응한다. 쿼드트리 데이터 구조의 각각의 노드는 대응하는 CU 에 대해 구문 데이터를 제공할 수도 있다. 예를 들어, 쿼드트리에서 노드는 그 노드에 대응하는 CU 가 서브-CUs 로 분할될지 여부를 나타내는 분할 플래그를 포함할 수도 있다. CU 에 대한 구문 엘리먼트들은 회귀적으로 정의될 수도 있으며, CU 가 서브-CUs 로 분할되는지 여부에 의존할 수도 있다.

[0053]

분할되지 않은 CU 는 하나 이상의 예측 유닛들 (PUs) 을 포함할 수도 있다. 일반적으로, PU 는 대응하는 CU

의 모두 또는 부분을 나타내며, PU 에 대한 참조 샘플을 추출하기 위한 데이터를 포함한다. 예를 들어, PU 가 인트라-모드 인코딩될 때, PU 는 PU 에 대한 인트라-예측 모드를 기술하는 데이터를 포함할 수도 있다. 또 다른 예로서, PU 가 인터-모드 인코딩될 때, PU 는 PU 에 대한 모션 벡터를 정의하는 데이터를 포함할 수도 있다. 모션 벡터를 정의하는 데이터는 예를 들어, 모션 벡터의 수평 성분, 모션 벡터의 수직 성분, 모션 벡터에 대한 해상도 (예컨대, 1/4 픽셀 정밀도 또는 1/8 픽셀 정밀도), 모션 벡터가 가리키는 참조 프레임, 및/또는 모션 벡터에 대한 참조 리스트 (예컨대, 리스트 0 또는 리스트 1) 를 기술할 수도 있다. PU(s) 를 정의하는 CU 에 대한 데이터는 또한 예를 들어, 하나 이상의 PUs 로의 CU 의 파티셔닝을 기술할 수도 있다. 모드들을 파티셔닝하는 것은 CU 가 코딩되지 않거나, 인트라-예측 모드 인코딩되거나, 또는 인터-예측 모드 인코딩되는지 여부에 따라 상이할 수도 있다.

[0054]

하나 이상의 PUs 를 갖는 CU 는 또한 하나 이상의 변환 유닛들 (TUs) 을 포함할 수도 있다. TUs 는 잔여 변환 계수들을 포함하는 데이터 구조를 포함하며, 이 잔여 변환 계수들은 일반적으로 양자화된다. 특히, PU 를 이용하는 예측에 뒤이어서, 비디오 인코더는 PU 에 대응하는 CU 의 부분에 대해 잔여 값들을 계산할 수도 있다. 잔여 값들은 변환되고, 양자화되고, 스캐닝되고, TU 로 저장될 수도 있으며, 이 TU 는 수행되었던 변환의 사이즈에 대응하는 가변 사이즈들을 가질 수도 있다. 따라서, TU 는 PU 의 사이즈에 반드시 제한되지는 않는다. 따라서, TUs 는 동일한 CU 에 대한 대응하는 PUs 보다 더 크거나 또는 더 작을 수도 있다. 일부 예들에서, TU 의 최대 사이즈는 CU 의 사이즈에 대응할 수도 있다. 또, TUs 는 주어진 CU 와 연관되는 잔여 변환 계수들을 포함하는 데이터 구조들을 포함할 수도 있다.

[0055]

도 2a 및 도 2b 는 예시적인 쿼드트리 (quadtree; 250) 및 대응하는 LCU (272) 를 예시하는 개념도들이다. 도 2a 는 예시적인 쿼드트리 (250) 를 도시하며, 이 쿼드트리는 계층적 방식으로 배열된 노드들을 포함한다. 쿼드트리 (250) 와 같은 쿼드트리에서 각각의 노드는 자식들이 없는 리프 노드이거나, 또는 4개의 자식 노드들을 가질 수도 있다. 도 2a 의 예에서, 쿼드트리 (250) 는 루트 노드 (252) 를 포함한다. 루트 노드 (252) 는 리프 노드들 (256A-256C) (리프 노드들 (256)) 및 노드 (254) 를 포함한, 4개의 자식 노드들을 갖는다. 노드 (254) 가 리프 노드가 아니기 때문에, 노드 (254) 는 이 예에서, 리프 노드들 (258A-258D) (리프 노드들 (258)) 인 4개의 자식 노드들을 포함한다.

[0056]

이 예에서, 쿼드트리 (250) 는 LCU (272) 과 같은 대응하는 LCU 의 특성들을 기술하는 데이터를 포함할 수도 있다. 예를 들어, 쿼드트리 (250) 는, 그의 구조에 의해, 서브-CUs 로의 LCU 의 분할을 기술할 수도 있다. LCU (272) 가 $2N \times 2N$ 의 사이즈를 갖는다고 가정한다. LCU (272) 는, 이 예에서, 각각 사이즈 $N \times N$ 의 4개의 서브-CUs (276A-276C) (서브-CUs (276)) 및 (274) 를 갖는다. 서브-CU (274) 는 각각 사이즈 $N/2 \times N/2$ 의 4개의 서브-CUs (278A-278D) (서브-CUs (278)) 로 추가로 분할된다. 이 예에서, 쿼드트리 (250) 의 구조는 LCU (272) 의 분할에 대응한다. 즉, 루트 노드 (252) 는 LCU (272) 에 대응하며, 리프 노드들 (256) 은 서브-CUs (276) 에 대응하며, 노드 (254) 는 서브-CU (274) 에 대응하며, 그리고, 리프 노드들 (258) 은 서브-CUs (278) 에 대응한다.

[0057]

쿼드트리 (250) 의 노드들에 대한 데이터는 노드에 대응하는 CU 가 분할되는지를 기술할 수도 있다. CU 가 분할되면, 4개의 추가적인 노드들이 쿼드트리 (250) 에 존재할 수도 있다. 일부 예들에서, 쿼드트리의 노드는 다음의 의사 코드와 유사하게 구현될 수도 있다:

```

quadtree_node {
    boolean split_flag(1);
    // signaling data
    if (split_flag) {
        quadtree_node child1;
        quadtree_node child2;
        quadtree_node child3;
        quadtree_node child4;
    }
}

```

[0058]

[0059]

split_flag 값은 현재의 노드에 대응하는 CU 가 분할되는 지 여부를 나타내는 1-비트 값일 수도 있다. CU 가 분할되지 않으면, split_flag 값은 '0' 일 수도 있으며, 반면, CU 가 분할되면, split_flag 값은 '1' 일 수도 있다. 쿼드트리 (250) 의 예에 있어, 분할 플래그 값들의 어레이는 101000000 일 수도 있다.

[0060]

일부 예들에서, 서브-CUs (276) 및 서브-CUs (278) 각각은 동일한 인트라-예측 모드를 이용하여 인트라-예측 인코딩될 수도 있다. 따라서, 비디오 인코더 (122) 는 인트라-예측 모드의 표시를 루트 노드 (252) 에 제공할 수도 있다. 더욱이, 특정 사이즈들의 서브-CUs 는 특정의 인트라-예측 모드에 대해 다수의 가능한 변환들을 가질 수도 있다. 비디오 인코더 (122) 는 이런 서브-CUs 에 사용할 변환의 표시를 루트 노드 (252) 에 제공할 수도 있다. 예를 들어, 사이즈 $N/2 \times N/2$ 의 서브-CUs 는 이용가능한 다수의 가능한 변환들을 가질 수도 있다. 비디오 인코더 (122) 는 루트 노드 (252) 에 사용할 변환을 시그널링할 수도 있다. 따라서, 비디오 디코더 (128) 는 루트 노드 (252) 에서 시그널링된 인트라-예측 모드 및 루트 노드 (252) 에서 시그널링된 변환에 기초하여, 서브-CUs (278) 에 적용할 변환을 결정할 수도 있다.

[0061]

이와 같이, 비디오 인코더 (122) 는 리프 노드들 (256) 및 리프 노드들 (258) 에서, 서브-CUs (276) 및 서브-CUs (278) 에 적용할 변환들을 시그널링할 필요가 없지만, 대신, 본 개시물의 기법에 따라서, 루트 노드 (252) 에서, 인트라-예측 모드를, 그리고 일부 예들에서는 어떤 사이즈들의 서브-CUs 에 적용할 변환을 간단히 시그널링할 수도 있다. 이 방법으로, 이들 기법들은 LCU (272) 과 같은 LCU 의 각각의 서브-CU 에 대한 변환 함수들을 시그널링하는 오버헤드 비용 (overhead cost) 을 감소시킬 수도 있다.

[0062]

일부 예들에서, 서브-CUs (276) 및/또는 서브-CUs (278) 에 대한 인트라-예측 모드들은 LCU (272) 에 대한 인트라-예측 모드들과는 상이할 수도 있다. 비디오 인코더 (122) 및 비디오 디코더 (128) 는 루트 노드 (252) 에서 시그널링된 인트라-예측 모드를 서브-CUs (276) 및/또는 서브-CUs (278) 에 대한 가용 인트라-예측 모드에 맵핑하는 함수들로 구성될 수도 있다. 이 함수는 서브-CUs (276) 및/또는 서브-CUs (278) 에 대한 인트라-예측 모드들로의, LCU (272) 에 대해 이용가능한 인트라-예측 모드들의 다-대-일 맵핑을 제공할 수도 있다.

[0063]

슬라이스는 비디오 블록들 (또는, LCUs) 로 분할될 수도 있으며, 각각의 비디오 블록은 도 2a 내지 도 2b 와 관련하여 설명한 쿼드트리 구조에 따라서 파티셔닝될 수도 있다. 게다가, 도 2c 에 나타난 바와 같이, "온 (ON)" 으로 표시되는 쿼드트리 서브-블록들은 본원에서 설명하는 루프 필터들에 의해 필터링될 수도 있으며, 반면, "오프 (OFF)" 로 표시된 쿼드트리 서브-블록들은 필터링되지 않을 수도 있다. 주어진 블록 또는 서브-블록들을 필터링할 지 여부의 결정은 인코더에서, 코딩중인 원래 블록에 대해, 필터링된 결과 및 비-필터링된 결과를 비교함으로써, 결정될 수도 있다. 도 2d 는 도 2c 에 나타난 쿼드트리 파티셔닝을 초래하는 파티셔닝 결정들을 나타내는 결정 트리 (decision tree) 이다. "온 (ON)" 블록들에 대한 임의의 픽셀들에 적용되는 실제 필터링은 위에서 설명한 메트릭들에 기초하여 결정될 수도 있다.

- [0064] 특히, 도 2c 는 쿼드트리 분할 (portioning) 방식에 따라서 가변 사이즈들의 더 작은 비디오 블록들로 파티셔닝 되는 상대적으로 큰 비디오 블록을 나타낼 수도 있다. 각각의 비디오 블록은 그 비디오 블록에 대해 필터링이 적용되거나 또는 회피되어야 하는지를 예시하기 위해, 도 2c 에 (온 또는 오프) 라벨링된다. 비디오 인코더는 각각의 비디오 블록의 필터링된 및 미필터링된 버전들을 코딩중인 원래 비디오 블록과 비교하여, 이 필터 맵을 정의할 수도 있다.
- [0065] 또, 도 2d 는 도 2c 에 나타난 쿼드트리 파티셔닝을 초래하는 파티셔닝 결정들에 대응하는 결정 트리이다. 도 2d 에서, 각각의 원은 CU 에 대응할 수도 있다. 그 원이 "1" 플래그를 포함하면, 그 CU 는 4개의 더 많은 CUs 로 추가로 분할되지만, 그 원이 "0" 플래그를 포함하면, 그 CU 는 추가로 파티셔닝되지 않는다. 각각의 (예컨대, CUs 에 대응하는) 원은 또한 연관되는 다이아몬드를 포함한다. 주어진 CU 에 대한 다이아몬드에서 플래그가 1 로 설정되면, 필터링은 그 CU 에 대해 턴 "온 (ON)"되지만, 주어진 CU 에 대한 다이아몬드에서 플래그가 0 으로 설정되면, 필터링은 턴 오프된다. 이 방법으로, 도 2c 및 도 2d 는, 필터링을 각각의 파티셔닝된 비디오 블록 (예컨대, LCU 내 각각의 CU) 에 적용하는 안하든 주어진 비디오 블록 (예컨대, LCU) 에 대한 쿼드트리 파티셔닝의 레벨을 통신하기 위해서 인코더에서 생성되어 디코더로 인코딩된 비디오 데이터의 슬라이스 당 적어도 한번 통신될 수 있는 필터 맵으로서 개별적으로 또는 집합적으로 간주될 수도 있다.
- [0066] 더 작은 비디오 블록들은 더 나은 해상도를 제공할 수 있으며, 높은 세부 레벨들을 포함하는 비디오 프레임의 로케이션들에 사용될 수도 있다. 더 큰 비디오 블록들은 더 큰 코딩 효율을 제공할 수 있으며, 낮은 세부 레벨을 포함하는 비디오 프레임의 로케이션들에 대해 사용될 수도 있다. 슬라이스는 복수의 비디오 블록들 및/또는 서브-블록들로 간주될 수도 있다. 각각의 슬라이스는 비디오 프레임의 독립적으로 디코딩가능한 비디오 블록들의 시리즈일 수도 있다. 이의 대안으로, 프레임들 자체는 디코딩가능한 비디오 블록들의 시리즈일 수도 있거나, 또는 프레임의 다른 부분들은 디코딩가능한 비디오 블록들의 시리즈로서 정의될 수도 있다. 용어 "비디오 블록들의 시리즈" 는 전체 프레임, 프레임의 슬라이스, 시퀀스로서 또한 지칭되는 화상들의 그룹 (GOP), 또는 적용가능한 코딩 기법들에 따라서 정의된 또다른 독립적으로 디코딩가능한 유닛과 같은, 임의의 독립적으로 디코딩가능한 비디오 프레임의 부분을 지칭할 수도 있다. 본 개시물의 양태들은 프레임들 또는 슬라이스들을 참조하여 설명될 것이며, 그러나 이런 참조들은 단지 예시적이다. 일반적으로 임의의 비디오 블록들의 시리즈는 프레임 또는 슬라이스 대신 사용될 수도 있는 것으로 이해되어야 한다.
- [0067] 구문 데이터는 각각의 CU 가 연관되는 구문 데이터를 포함하도록, 코딩된-유닛 단위로 정의될 수도 있다. 본원에서 설명되는 필터 정보는 CU 에 대한 이런 구문의 부분일 수도 있지만, CU 대신, 프레임, 슬라이스, GOP, LCU, 또는 비디오 프레임들의 시퀀스와 같은 비디오 블록들의 시리즈에 대한 구문의 부분일 가능성이 더 많을 것이다. 구문 데이터는 슬라이스 또는 프레임의 CUs 와 함께 사용되는 세트 또는 필터들의 세트들을 나타낼 수 있다. 게다가, 모든 필터 정보가 반드시 비디오 블록들의 공통 시리즈의 헤더에 포함되어야 하는 것은 아니다. 예를 들어, 필터 기술 구문은 프레임 헤더로 송신될 것이며, 반면 다른 필터 정보는 LCU 를 위한 헤더로 시그널링된다.
- [0068] 비디오 인코더 (122) 는 예측 코딩을 수행할 수도 있으며, 여기서, 코딩중인 비디오 블록이 예측 블록을 식별하기 위해 예측 프레임 (또는, 다른 CU) 과 비교된다. 코딩중인 현재의 비디오 블록과 예측 블록 사이의 차이는 잔여 블록으로서 코딩되며, 예측 구문이 예측 블록을 식별하는데 사용된다. 잔여 블록은 변환되어 양자화될 수도 있다. 변환 기법들은 DCT 프로세스 또는 개념적으로 유사한 프로세스, 정수 변환들, 웨이블릿 변환들, 또는 다른 유형들의 변환들을 포함할 수도 있다. DCT 프로세스에서, 일 예로서, 변환 프로세스는 픽셀 값들의 세트를 변환 계수들로 변환하며, 이 변환 계수들은 주파수 도메인에서 픽셀 값들의 에너지를 나타낼 수도 있다. 양자화가 일반적으로 변환 계수들에 적용되며, 일반적으로 임의의 주어진 변환 계수와 연관되는 비트수를 제한하는 프로세스를 수반한다.
- [0069] 변환 및 양자화에 뒤이어서, 엔트로피 코딩이 양자화 및 변환된 잔여 비디오 블록들에 대해 수행될 수도 있다. 인코딩 동안 정의된 예측 벡터들 및 필터 정보와 같은 구문 엘리먼트들이 또한 각각의 CU 에 대한 엔트로피 코딩된 비트스트림에 포함될 수도 있다. 일반적으로, 엔트로피 코딩은 양자화된 변환 계수들 및/또는 다른 구문 정보의 시퀀스를 일괄 압축하는 하나 이상의 프로세스들을 포함한다. 지그 재그 스캐닝 기법들과 같은 스캐닝 기법들이, 2차원 비디오 블록들로부터 계수들의 하나 이상의 직렬화된 1차원 벡터들을 정의하기 위해, 양자화된 변환 계수들에 대해, 예컨대, 엔트로피 코딩 프로세스의 부분으로서, 수행된다. 다른 스캔 순서들 또는 적응적 스캐닝들을 포함한, 다른 스캐닝 기법들이 또한 사용되고, 가능한 인코딩된 비트스트림으로 시그널링될 수도 있다. 어쨌든, 스캐닝된 계수들이 그후 예컨대, 콘텐츠 적응 가변 길이 코딩 (CAVLC), 컨텍스트 적응 2진 산술 코딩 (CABAC), 또는 또다른 엔트로피 코딩 프로세스를 통해서 임의의 구문 정보와 함께 엔트로피

코딩된다.

- [0070] 인코딩 프로세스의 부분으로서, 인코딩된 비디오 블록들은 후속 비디오 블록들의 후속 예측-기반의 코딩에 사용되는 비디오 데이터를 생성하기 위해 디코딩될 수도 있다. 이 단계에서, 비디오 품질을 향상시키고 그리고 예컨대, 디코딩된 비디오로부터 블록킹 현상 아티팩트들을 제거하기 위해, 필터링이 수행될 수도 있다. 이 필터링된 데이터가 다른 비디오 블록들의 예측을 위해 사용될 수도 있으며, 이 경우 필터링은 "인-루프" 필터링으로서 지칭된다. 이의 대안으로, 다른 비디오 블록들의 예측은 미필터링된 데이터에 기초할 수도 있으며, 이 경우 필터링은 "사후 필터링"으로서 지칭된다.
- [0071] 프레임 단위, 슬라이스 단위, 또는 LCU 단위로, 비디오 인코더 (122)는 필터들의 하나 이상의 세트들을 선택할 수도 있으며, 그리고, 코딩된-유닛 단위로, 인코더는 필터링을 적용할지 여부를 결정할 수도 있다. 필터링되는 CUs에 대해, 인코더는 픽셀 단위 또는 그룹 단위로 필터링을 수행할 수 있으며, 여기서, 그룹은 예를 들어, 2 x 2 픽셀들의 블록 또는 4 x 4 픽셀들의 블록일 것이다. 이들 선택들은 비디오 품질을 증진시키는 방법으로 이루어질 수 있다. 이런 필터들의 세트들은 사전-정의된 필터들의 세트들로부터 선택될 수도 있거나, 또는 비디오 품질을 증진하도록 적응적으로 정의될 수도 있다. 일 예로서, 비디오 인코더 (122)는 상이한 필터들이 그 프레임 또는 슬라이스의 CUs의 상이한 픽셀들 또는 픽셀들의 그룹들에 대해 사용되도록, 주어진 프레임 또는 슬라이스에 대해 필터들의 여러 세트들을 선택하거나 또는 정의할 수도 있다. 특히, CU와 연관되는 각각의 입력에 대해, 필터 계수들의 여러 세트들이 정의될 수도 있으며, CU의 픽셀들과 연관되는 2개 이상의 메트릭들이 필터들의 세트 중에서 어느 필터를 이런 픽셀들 또는 픽셀들의 그룹들과 함께 사용할지 결정하는데 사용될 수도 있다.
- [0072] 일부의 경우, 비디오 인코더 (122)는 여러 필터 계수들의 세트들을 적용하고, 코딩된 블록과 원래 블록 사이의 왜곡의 양, 및/또는 최고 압축 레벨들의 관점에서 최상의 품질 비디오를 생성하는 하나 이상의 세트들을 선택할 수도 있다. 어쨌든, 일단 선택되면, 비디오 인코더 (128)가 각각의 주어진 CU에 대한 인코딩 프로세스 동안 적용되었던 동일한 필터링을 적용할 수 있도록, 각각의 CU에 대해 비디오 인코더 (122)에 의해 적용된 필터 계수들의 세트가, 인코딩되어 목적지 디바이스 (118)의 비디오 인코더 (128)로 통신될 수도 있다.
- [0073] 어느 필터를 CU에 대한 특정의 입력과 함께 사용할지를 결정하는데 2개 이상의 메트릭들이 사용될 때, 그 특정의 CU에 대한 필터의 선택이 비디오 인코더 (128)로 반드시 통신될 필요는 없다. 대신, 비디오 인코더 (128)는 또한 2개 이상의 메트릭들을 계산하고, 비디오 인코더 (122)에 의해 이전에 제공된 필터 정보에 기초하여, 2개 이상의 메트릭들의 조합을 특정의 필터에 매칭할 수 있다.
- [0074] 도 3은 본 개시물과 부합하는 비디오 인코더 (350)를 예시하는 블록도이다. 비디오 인코더 (350)는 디바이스 (120)의 비디오 인코더 (122), 또는 상이한 디바이스의 비디오 인코더에 대응할 수도 있다. 도 3에 나타낸 바와 같이, 비디오 인코더 (350)는 예측 모듈 (332), 가산기들 (348 및 351), 및 메모리 (334)를 포함한다. 비디오 인코더 (350)는 또한 변환 유닛 (338) 및 양자화 유닛 (340)뿐만 아니라, 역양자화 유닛 (342) 및 역변환 유닛 (344)을 포함한다. 비디오 인코더 (350)는 또한 디블록킹 필터 (347) 및 적응형 필터 유닛 (349)을 포함한다. 비디오 인코더 (350)는 또한 엔트로피 인코딩 유닛 (346)을 포함한다. 비디오 인코더 (350)의 필터 유닛 (349)은 필터링 동작들을 수행할 수도 있으며, 또한 디코딩에 사용되는 바람직한 또는 선호하는 필터 또는 필터들의 세트를 식별하기 위한 필터 선택 유닛 (FSU; 353)을 포함할 수도 있다. 필터 유닛 (349)은 또한 그 선택된 필터들이 디코딩 동작 동안 사용되는 또다른 디바이스로 필터 정보로서 효율적으로 통신될 수 있도록, 선택된 필터들을 식별하는 필터 정보를 생성할 수도 있다.
- [0075] 인코딩 프로세스 동안, 비디오 인코더 (350)는 코딩되는 비디오 블록, 예컨대 LCU를 수신하고, 예측 모듈 (332)은 비디오 블록 상에서 예측 코딩 기법들을 수행한다. 위에서 설명한 쿼트트리 파티셔닝 방식을 이용하여, 예측 모듈 (332)은 비디오 블록을 파티셔닝하고 상이한 사이즈들의 CUs 상에서 예측 코딩 기법들을 수행할 수 있다. 인터-코딩에 있어, 예측 모듈 (332)은 예측 블록을 정의하기 위해, 비디오 블록의 서브-블록들을 포함한, 인코딩중인 비디오 블록을 하나 이상의 비디오 참조 프레임들 또는 슬라이스들에서의 여러 블록들과 비교한다. 인트라 코딩에 있어, 예측 모듈 (332)은 동일한 CU 내 이웃하는 데이터에 기초하여 예측 블록을 생성한다. 예측 모듈 (332)은 예측 블록을 출력하고, 가산기 (348)는 잔여 블록을 생성하기 위해, 코딩중인 비디오 블록으로부터 그 예측 블록을 감산한다.
- [0076] 인터-코딩에 있어, 예측 모듈 (332)은 예측 블록을 가리키는 모션 벡터를 식별하고 그 모션 벡터에 기초하여 예측 블록을 생성하는 모션 추정 및 모션 보상 유닛들을 포함할 수도 있다. 일반적으로, 모션 추정은 모션 벡터를 생성하는 프로세스로 간주되며, 모션을 추정한다. 예를 들어, 모션 벡터는 현재의 프레임 내 코딩중

인 현재의 블록에 상대적인, 예측 프레임 내 예측 블록의 변위를 나타낼 수도 있다. 모션 보상은 일반적으로 모션 추정에 의해 결정된 모션 벡터에 기초하여 예측 블록을 폐지하거나 또는 생성하는 프로세스로서 간주된다. 인트라 코딩에 있어, 예측 모듈 (332) 은 동일한 CU 내 이웃하는 데이터에 기초하여 예측 블록을 생성한다. 하나 이상의 인트라-예측 모드들이 인트라 예측 블록이 정의될 수 있는 방법을 정의할 수도 있다.

[0077]

예측 모듈 (332) 이 예측 블록을 출력하고, 가산기 (348) 가 코딩중인 비디오 블록으로부터 예측 블록을 감산하여 잔여 블록을 생성한 후, 변환 유닛 (338) 은 그 잔여 블록에 변환을 적용한다. 이 변환은 이산 코사인 변환 (DCT), 또는 개념적으로 유사한 변환, 예컨대 HEVC 표준과 같은 코딩 표준에 의해 정의되는 변환을 포함할 수도 있다. 웨이블릿 변환들, 정수 변환들, 서브밴드 변환들 또는 다른 유형들의 변환들이 또한 이용될 수 있다. 어쨌든, 변환 유닛 (338) 은 그 변환을 잔여 블록에 적용하여, 잔여 변환 계수들의 블록을 생성한다. 이 변환은 잔여 정보를 픽셀 도메인으로부터 주파수 도메인으로 변환할 수도 있다.

[0078]

양자화 유닛 (340) 은 그후 잔여 변환 계수들을 양자화하여, 비트 레이트를 추가로 감소시킨다. 양자화 유닛 (340) 은, 예를 들어, 계수들 각각을 코딩하는데 사용되는 비트수를 제한할 수도 있다. 양자화 이후, 엔트로피 인코딩 유닛 (346) 은 양자화된 계수 블록을 2차원 표현으로부터 하나 이상의 직렬화된 1차원 벡터들로 스캐닝한다. 스캔 순서는 (지그-재그 스캐닝, 수평 스캐닝, 수직 스캐닝, 조합들, 또는 또다른 사전-정의된 순서와 같은) 정의된 순서로 발생하도록 사전 프로그래밍될 수도 있거나, 또는 가능한 한 이전 코딩 통계치들에 기초하여 적응적으로 정의될 수도 있다.

[0079]

이 스캐닝 프로세스에 뒤이어서, 엔트로피 인코딩 유닛 (346) 은 데이터를 추가로 압축하기 위해, 양자화된 변환 계수들을 (임의의 구문 엘리먼트들과 함께) CAVLC 또는 CABAC 과 같은 엔트로피 코딩 방법에 따라서, 인코딩할 수도 있다. 엔트로피 코딩된 비트스트림에 포함되는 구문 엘리먼트들은 인트라-코딩을 위한 모션 벡터들 또는 인트라 코딩을 위한 예측 모드들과 같은, 예측 모듈 (332) 로부터의 예측 구문을 포함할 수도 있다. 엔트로피 코딩된 비트스트림에 포함되는 구문 엘리먼트들은 또한 필터 유닛 (349) 으로부터의 필터 정보를 포함할 수도 있으며, 본문에서 설명하는 방법으로 인코딩될 수 있다.

[0080]

CAVLC 는 ITU H.264/MPEG4, AVC 표준에 의해 지원되는 엔트로피 인코딩 기법의 일 유형이며, 엔트로피 인코딩 유닛 (346) 에 의해 벡터화 방식으로 적용될 수도 있다. CAVLC 는 변환 계수들의 직렬화된 "런들 (runs)" 및/또는 구문 엘리먼트들을 효과적으로 압축하는 방법으로 가변 길이 코딩 (VLC) 테이블들을 이용한다. CABAC 는 ITU H.264/MPEG4, AVC 표준에 의해 지원되는 엔트로피 코딩 기법의 또다른 유형이며, 엔트로피 인코딩 유닛 (346) 에 의해 벡터화 방식으로 적용될 수도 있다. CABAC 는 2진수화, 컨텍스트 모델 선택, 및 2진 산술 코딩을 포함한, 여러 단계들을 수반한다. 이 경우, 엔트로피 인코딩 유닛 (346) 은 CABAC 에 따라서 변환 계수들 및 구문 엘리먼트들을 코딩한다. ITU H.264/MPEG4, AVC 표준과 유사하게, 차기 HEVC 표준은 또한 CAVLC 및 CABAC 엔트로피 코딩 양자를 지원할 수도 있다. 더욱이, 많은 다른 유형들의 엔트로피 코딩 기법들이 또한 존재하며, 새로운 엔트로피 코딩 기법들이 미래에 심층팔구 대두할 것이다. 본 개시물은 임의의 특정의 엔트로피 코딩 기법에 한정되지 않는다.

[0081]

엔트로피 인코딩 유닛 (346) 에 의한 엔트로피 코딩 이후, 인코딩된 비디오가 또다른 디바이스로 송신되거나 또는 추후 송신 또는 취출을 위해 아카이브될 수도 있다. 또, 인코딩된 비디오는 코딩된 벡터들 및 여러 구문을 포함할 수도 있으며, 디코딩 프로세스를 적절히 구성하기 위해 디코더에 의해 사용될 수 있다. 역양자화 유닛 (342) 및 역변환 유닛 (344) 은 역양자화 및 역변환을 각각 적용하여, 픽셀 도메인에서 잔여 블록을 재구성한다. 합산기 (351) 는 재구성된 잔여 블록을 예측 모듈 (332) 에 의해 생성된 예측 블록에 가산하여, 사전-디블록킹된 재구성된 이미지로서 종종 지칭되는 사전-디블록킹된 재구성된 비디오 블록을 생성한다. 디-블록킹 필터 (347) 는 그 사전-디블록킹된 재구성된 비디오 블록에 필터링을 적용하여, 블록킹 현상 또는 다른 아티팩트들을 제거함으로써 비디오 품질을 향상시킬 수도 있다. 디-블록킹 필터 (347) 의 출력은 사후-디블록킹된 비디오 블록, 재구성된 비디오 블록, 또는 재구성된 이미지로서 지칭될 수 있다.

[0082]

필터 유닛 (349) 은 단일 입력 또는 다수의 입력들을 수신하도록 구성될 수 있다. 도 3 의 예에서, 필터 유닛 (349) 은 사후-디블록킹된 재구성된 이미지 (RI), 사전-디블록킹된 재구성된 이미지 (pRI), 예측 이미지 (PI), 및 재구성된 잔여 블록 (EI) 을 입력으로서 수신한다. 필터 유닛 (349) 은 이들 입력들 중 임의의 입력을 개별적으로 또는 조합하여 사용하여, 재구성된 이미지를 생성함으로써, 메모리 (334) 에 저장할 수 있다. 게다가, 아래에서 좀더 자세히 설명하는 바와 같이, 2개 이상의 메트릭들에 기초하여, 입력(들) 에 적용되도록 하나 이상의 필터들이 선택될 수 있다. 일 예에서, 필터 유닛 (349) 의 출력은 RI 에 적용되는 하나의 추가적인 필터일 수도 있다. 또다른 예에서, 필터 유닛 (349) 의 출력은 pRI 에 적용되는 하나의 추가적인

필터일 수도 있다. 다른 예들에서, 그러나, 필터 유닛 (349) 의 출력은 다수의 입력들에 기초할 수도 있다.

예를 들어, 필터 유닛 (349) 은 제 1 필터를 pRI 에 적용한 후, EI 및 PI 의 필터링된 버전들과 함께, pRI 의 필터링된 버전을 이용하여, 재구성된 이미지를 생성할 수도 있다. 필터 유닛 (349) 의 출력이 단일 입력에 적용중인 하나의 추가적인 필터의 결과인 경우, 필터 유닛 (349) 은 필터들을 다른 입력들에 실제로 적용할 수도 있지만, 그들 필터들은 모두 제로 계수들을 가질 것이다. 이와 유사하게, 필터 유닛 (349) 의 출력이 3개의 필터들을 3개의 입력들에 적용하는 결과이면, 필터 유닛 (349) 은 필터를 제 4 입력에 실제로 적용할 수도 있지만, 그 필터는 모두 제로 계수들을 가질 것이다.

[0083]

필터 유닛 (349) 은 또한 단일 입력을 수신하도록 구성될 수도 있다. 예를 들어, 도 3 은 PI, EI, pRI, 및 RI 가 필터 유닛 (349) 로의 입력인 것으로 나타내지만, 일부 구현예들에서, RI 는 필터 유닛 (349) 에 의해 수신된 유일한 입력일 것이다. 이런 구현예에서, 필터 유닛 (349) 은 RI 의 필터링된 버전이 RI 의 미필터링된 버전보다 원래 이미지에 더 유사하도록 RI 에 필터를 적용할 것이다. 다른 구현예들에서, 필터 유닛 (349) 및 디-블록킹 필터 (347) 는 pRI 에 필터링을 적용하는 단일 필터링 유닛으로 결합될 수도 있다. 일반적으로 멀티-메트릭-기반의 필터 맵핑에 관련한, 본 개시물의 기법들은 단일-입력 필터링 방식 및 다수의 필터들을 이용하는 멀티-입력 필터링 방식 양자와 호환가능하다.

[0084]

필터 유닛 (349) 에 의한 필터링은 미필터링된 예측 비디오 블록들보다 코딩중인 비디오 블록들에 더 가깝게 매칭하는 예측 비디오 블록들을 생성함으로써 압축을 향상시킬 수도 있다. 필터링 이후, 그 재구성된 비디오 블록은 후속 비디오 프레임 또는 다른 CU 에서의 블록을 인터-코딩하기 위해, 예측 모듈 (332) 에 의해 참조 블록으로서 사용될 수도 있다. 필터 유닛 (349) 은 "인-루프" 로 나타내지만, 본 개시물의 기법들은 또한 사후 필터들과 함께 사용될 수 있으며, 이 경우, 후속 CUs 에서 데이터를 예측하는 목적들을 위해 (필터링된 데이터 대신) 비-필터링된 데이터가 사용될 것이다.

[0085]

슬라이스 또는 프레임과 같은 비디오 블록들의 시리즈에 대해, 필터 유닛 (349) 은 비디오 품질을 증진시키는 방법으로 각각의 입력에 대해 필터들의 세트들을 선택할 수도 있다. 예를 들어, 필터 유닛 (349) 은 비디오 품질 또는 향상된 압축을 증진시키기 위해서, 계수들의 사전-정의된 세트들로부터 필터들의 세트들을 선택할 수도 있거나, 또는 필터들을 적응적으로 정의할 수도 있다. 필터 유닛 (349) 은 동일한 필터들의 세트(들) 이 그 CU 의 상이한 비디오 블록들의 픽셀들에 사용되도록, 주어진 CU 에 대해 하나 이상의 필터들의 세트들을 선택하거나 또는 정의할 수도 있다. 특정의 프레임, 슬라이스, 또는 LCU 에 대해, 필터 유닛 (349) 은 다수의 입력들에 여러 필터들의 세트들을 적용할 수도 있으며, FSU (353) 는 최상의 품질 비디오 또는 최고 레벨들의 압축을 생성하는 세트를 선택할 수도 있다. 이의 대안으로, FSU (353) 는 자기-상관들 및 다수의 입력들과 원래 이미지 사이의 교차-상관들을 분석함으로써 새로운 필터를 추적할 수도 있다. 새로운 필터들의 세트는 예를 들어, 자기-상관 및 교차-상관에 기초하여 Wiener-Hopt 방정식들을 풀어써 결정될 수도 있다. 새로운 필터들의 세트가 추적되거나 또는 기존 필터들의 세트가 선택되는지 여부에 관계없이, 필터 유닛 (349) 은 디코더로 하여금 또한 특정의 프레임 또는 슬라이스에 사용되는 필터들의 세트 또는 세트들을 식별가능하게 하는, 비트스트림에의 포함을 위한 구문을 생성한다.

[0086]

본 개시물에 따르면, 비디오 블록들의 시리즈 내 CU 의 각각의 픽셀에 대해, 필터 유닛 (349) 은 그 필터들의 세트 중 어느 필터가 사용될 지를, CU 내 하나 이상의 픽셀들의 세트들과 연관되는 성질들 (properties) 을 정량화하는 2개 이상의 메트릭들에 기초하여, 선택할 수도 있다. 이러한 방법으로, FSU (353) 가 프레임 또는 슬라이스와 같은 더 높은 레벨 코딩된 유닛에 대해 필터들의 세트들을 결정할 수도 있으며, 반면, 필터 유닛 (349) 은 그 세트(들) 중 어느 필터(들) 이 하부 레벨 코딩된 유닛의 특정의 픽셀에 대해 사용될 지를, 그 하부 레벨 코딩된 유닛의 픽셀들과 연관되는 2개 이상의 메트릭들에 기초하여, 결정한다.

[0087]

M 개 필터들의 세트가 각각의 입력에 대해 사용될 수도 있다. 설계 선호사항들에 따라서, M 은 예를 들어, 2 만큼 작거나 또는 16 만큼 클 수도 있으며, 또는 심지어 더 높을 수도 있다. 입력 당 다수의 필터들이 비디오 품질을 향상시킬 수도 있지만, 또한 시그널링 필터들의 세트들을 인코더로부터 디코더로 시그널링하는 것과 연관되는 오버헤드를 증가시킬 수도 있다. M 개 필터들의 세트는 위에서 설명한 바와 같이 FSU (353) 에 의해 결정되어, 각각의 프레임 또는 슬라이스에 대해 디코더로 시그널링될 수 있다. 세그멘테이션 맵이 CU 가 어떻게 세그먼트화되는 지, 그리고 CU 의 특정의 서브-유닛이 필터링되는지 여부를 나타내는데 사용될 수 있다. 세그멘테이션 맵은 예를 들어, CU 에 대해 위에서 설명한 바와 같은 분할 플래그들의 어레이 뿐만 아니라, 각각의 서브-CU 가 필터링되든 아니든 추가적인 비트 시그널링을 포함할 수도 있다. 필터링되는 CU 의 픽셀과 연관되는 각각의 입력에 대해, 필터들의 세트 중의 특정의 필터가 2개 이상의 메트릭들에 기초하여 선정될 수 있다. 2개 이상의 메트릭들에 대한 값들의 조합들이 M 개 필터들의 세트 중의 특정의 필터들에 인력

스될 수 있다.

[0088]

도 4a 는 필터들의 세트 중의 필터들에 인덱스된 2개의 메트릭들에 대한 값들의 범위들을 예시하는 개념도이다.

도 4a 의 구체적인 예는 8개의 필터들 (즉, 필터 1, 필터 2... 필터 8) 을 나타내지만, 더 많거나 더 적은 필터들이 유사하게 사용될 수도 있다. 도 4a 는 본 개시물의 기법에 따라서 필터를 선택하는데 사용되는 2개의 메트릭들을 나타낸다. 2개의 메트릭들은 예를 들어, 비-방향 특성의 활동 (예컨대, 합산-수정된 라플라시안 값) 과 방향, 방향-특성의 활동과 에지 검출, 방향 메트릭과 에지 메트릭, 수평 활동 메트릭과 수직 활동 메트릭, 또는 2개의 다른 메트릭들과 관련되는 픽셀 데이터의 성질들을 정량화할 수도 있다. 일부의 경우, 3개 이상의 메트릭들이 사용될 것이며, 이 경우 도 4a 의 개념도는 필터들의 세트 중의 필터들에의 메트릭들의 범위들의 맵핑을 위한 제 3 치수 (third dimension) 를 포함할 것이다.

[0089]

도 4a 의 예에서, 제 1 메트릭 (메트릭 1) 은 4개의 범위들 (범위들 (1-1, 1-2, 1-3, 및 1-4)) 을 가지며, 제 2 메트릭 (메트릭 2) 은 또한 4개의 범위들 (범위들 (2-1, 2-2, 2-3, 및 2-4)) 을 갖는다. 따라서, 도 4a 의 예는 메트릭 1 및 메트릭 2 에 대해 16개의 범위들의 조합들을 갖는다. 도 4a 로부터 볼 수 있는 바와 같이, 그러나, 각각의 조합은 고유한 필터와 반드시 연관되지는 않는다. 예를 들어, 도 4a 의 예에서, 범위 (1-1) 과 범위 (2-1) 의 조합 뿐만 아니라, 조합들 (1-1 과 2-2), 및 (1-1 과 2-3) 의 조합이 모두 필터 1 에 맵핑된다. 이에 반해, 필터 4 는, 오직 하나의 조합 (1-1 및 2-4) 에만 맵핑된다. 도 4a 의 범위들은 상대적으로 동일한 것으로 나타내어 지지만, 범위들의 사이즈들은 변할 수도 있다. 예를 들어, 일부 구현예들에서, 범위 (1-1) 는 범위 (1-2) 보다 더 큰 값들의 범위를 포괄할 수도 있다. 게다가, 도 4a 는 메트릭 1 및 메트릭 2 를 동일한 개수의 범위들을 갖는 것으로 나타내지만, 제 1 메트릭에 대한 범위들의 개수 및 제 2 메트릭에 대한 범위들의 개수는 반드시 같을 필요가 없다. 예를 들어, 메트릭 1 이 분산 메트릭이고 메트릭 2 가 방향 메트릭이면, 메트릭 1 은 8개의 범위들을 사용하는 반면, 메트릭 2 은 3개의 범위들을 사용할 것이다.

[0090]

일부 예들에서, 메트릭 1 및 메트릭 2 의 범위들은 값들의 연속적인 스펙트럼을 나타낼 수도 있다. 예를 들어, 메트릭 1 이 합산-수정된 라플라시안 값이면, 범위 (1-2) 는 범위 (1-1) 보다 더 많은 활동, 그러나 범위 (1-3) 보다는 더 적은 활동에 대응할 수도 있으며, 범위 (1-4) 는 범위 (1-3) 보다 더 많은 활동에 대응할 수도 있다. 범위 내에서, 특성의 픽셀 또는 픽셀들의 그룹에 대해 결정된 활동의 양은 유사하게 메트릭 1 축을 따라서 증가할 수도 있다. 다른 예들에서, 메트릭 1 및 메트릭 2 의 범위들은 실제 범위들을 나타내지 않을 수도 있으며, 대신 별개의 결정들을 나타낼 수도 있다. 예를 들어, 메트릭 2 가 방향 메트릭이면, 범위 (1-1) 는 무방향 (no direction) 의 결정에 대응할 수도 있으며, 범위 (2-2) 는 수평 방향의 결정에 대응할 수도 있으며, 범위 (2-3) 는 수직 방향의 결정에 대응할 수도 있으며, 범위 (2-4) 는 대각선 방향의 결정을 나타낼 수도 있다. 이하에서 좀더 자세하게 설명하는 바와 같이, 무방향, 수평 방향, 수직 방향, 및 대각선 방향은 별개의 결정들일 수 있으며, 따라서, 메트릭 2 에 대한 범위들은 메트릭 1 의 범위들이 행하는 동일한 방법으로 값들의 연속적인 스펙트럼을 나타내지 않을 것이다.

[0091]

도 4b 는 활동 메트릭 및 방향 메트릭에 대한 값들의 범위들을 예시하는 개념도이다. 도 4b 의 예에서, 방향 메트릭은 3개의 별개의 결정들 (무방향, 수평, 및 수직) 을 포함한다. 활동을 결정하는 기법들 뿐만 아니라, 무방향, 수평, 및 수직을 결정하는 기법들이 이하에서 더욱더 자세하게 설명될 것이다. 도 4b 의 구체적인 예는 6개의 필터들 (즉, 필터 1, 필터 2... 필터 6) 을 나타내지만, 더 많거나 또는 더 적은 필터들이 유사하게 사용될 수도 있다. 도 4b 에 의해 볼 수 있는 바와 같이, 2개의 메트릭들 (활동 및 방향) 은 조합들 421 내지 435 로서 식별된, 15 개의 조합들을 생성한다. 일부의 경우, 그러나, 도 4b 에 명시적으로 나타내지 않은 추가적인 조합들이 또한 사용될 수도 있다. 예를 들어, 무활동에 대응하는 조합은 대응하는 필터를 또한 갖는 16번째 조합일 수도 있다.

[0092]

필터 유닛 (349) 은 도 4a 및 도 4b 의 예시적인 맵핑들과 같은, 2개 이상의 메트릭들의 범위들의 조합들에의 필터들의 맵핑을 저장하고, 그 맵핑을 이용하여, 필터들의 세트 중 어느 필터를 CU 에서 특성의 픽셀 또는 픽셀들의 그룹에 적용할지를 결정할 수 있다. 2개 이상의 메트릭들의 범위들의 조합들에의 필터들의 맵핑은 예를 들어, 위에서 설명한 필터 선택 프로세스의 부분으로서 필터 유닛 (349) 에 의해 결정될 수도 있다. 맵핑이 어떻게 결정되는 지에 관계없이, 필터 유닛 (349) 은 디코더가 그 맵핑을 재구성할 수 있도록 하는 정보를 생성할 수 있다. 이 정보는 필터들에의 범위들의 조합들의 맵핑을 시그널링하기 위해 그 코딩된 비트스트림에 포함될 수 있다. 시그널링되는 범위들에의 조합들의 맵핑은 범위 조합들을 필터 식별 ID들에 맵핑할 수도 있다. 특성의 필터에 대한 실제 계수들은 별개로 시그널링될 것이다.

[0093]

이 정보를 생성하기 위해, 필터 유닛 (349) 은 먼저 조합들에 대한 송신 순서를 결정한다. 송신 순서는 일반적으로 범위들의 조합들에 대해 필터들이 시그널링되는 순서를 지칭한다. 일 예로서 도 4a 를 이용하면, 필터 유닛 (349) 은 좌우, 상하 송신 순서를 이용할 것이며, 여기서는, 조합 (401) 에 대한 필터가 첫번째로 시그널링되고, 조합 (402) 에 대한 필터가 두번째로 시그널링되고, 나머지 조합들이 403 => 404 => 405 => 406 => 407 => 408 => 409 => 410 => 411 => 412 => 413 => 414 => 415 => 416 의 순서로 시그널링된다. 필터 유닛 (349) 은 또한 상하, 지그-재그 송신 순서를 이용할 것이며, 여기서는, 조합들에 대한 필터들은 401 => 402 => 403 => 404 => 408 => 407 => 406 => 405 => 409 => 410 => 411 => 412 => 416 => 415 => 414 => 413 의 순서로 시그널링된다. 필터 유닛 (349) 은 또한 상하, 좌우 송신 순서를 이용할 것이며, 여기서는, 조합들에 대한 필터들이 401 => 405 => 409 => 413 => 402 => 406 => 410 => 414 => 403 => 407 => 411 => 415 => 404 => 408 => 412 => 416 의 순서로 시그널링된다. 필터 유닛 (349) 은 또한 좌우, 지그-재그 송신 순서를 이용할 것이며, 여기서는, 조합들에 대한 필터들이 401 => 405 => 409 => 413 => 414 => 410 => 406 => 402 => 403 => 407 => 411 => 415 => 416 => 412 => 408 => 404 의 순서로 시그널링된다. 도 4b 를 참조하면, 필터 유닛 (349) 은 송신 순서가 421 => 422 => 423 => 424 => 425 => 426 => 427 => 428 => 429 => 430 => 431 => 432 => 433 => 434 => 435 가 되도록, 좌우, 하상 (bottom-to-top) 송신 순서를 이용할 수도 있다. 추측될 수 있는 바와 같이, 이들은 가능한 많은 송신 순서들 중 단지 몇 개이다.

[0094]

본 개시물의 기법에 따르면, 필터 유닛 (349) 은 코드워드들의 시리즈를 이용하여 디코더에의 맵핑을 시그널링할 수 있다. 예를 들어, 필터 유닛 (349) 은 제 1 코드워드를 생성하여, 디코딩중인 현재의 조합이 제 1 메트릭에 대한 동일한 범위를 공유하는 가장 최근에 디코딩된 조합과 동일한 필터에 맵핑하는 지를 나타낼 수 있다. 디코딩중인 현재의 조합이 제 2 메트릭에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합과 동일한 필터에 맵핑하면, 필터 유닛 (349) 은 제 1 코드워드 대신, 제 2 코드워드를 생성할 수 있다. 디코딩중인 현재의 조합이 이들 가장 최근에 디코딩된 조합들 중 어느 하나의 조합과 동일한 필터에 맵핑하지 않으면, 필터 유닛 (349) 은 제 1 코드워드 또는 제 2 코드워드 대신, 디코딩중인 현재의 조합에 대응하는 필터를 나타내는 제 3 코드워드를 생성할 수 있다. 현재의 예의 제 1 및 제 2 코드워드는 제 3 코드워드에 비해 상대적으로 짧을 수도 있다. 예를 들어, 제 1 코드워드 및 제 2 코드워드는 각각 2개의 비트들이 존재할 것이며 (예컨대, 각각, 00 및 01), 반면 제 3 코드워드는 더 많은 비트들이 존재한다 (1 의 제 1 비트, 더하기 추가적인 비트들). 이 특성의 상황에서, 디코딩중인 현재의 조합 또는 디코딩중인 이전 조합은 범위 조합들에의 필터들의 맵핑이 인코더에 의해 시그널링되거나 또는 디코더에 의해 구성되어지는 인코딩 및 디코딩 프로세스들의 부분을 지칭하며, 조합 자체의 송신 또는 디코딩을 반드시 지칭하지는 않는다.

[0095]

위에서 설명한 기법들의 예들은 도 4a 및 상하, 좌우 송신 순서를 참조하여 주어질 것이다. 예를 들어, 조합 (407) 이 현재 디코딩중인 조합이면, 조합 (406) 은 메트릭 1 에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합이며, 조합 (403) 은 메트릭 2 에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합이다. 조합 (407) 이 제 2 메트릭에 대한 동일한 범위 (즉, 메트릭 2 에 대해 범위 (2-3)) 를 공유하는 가장 최근에 디코딩된 조합과 동일한 필터 (도 4a 의 필터 7) 에 맵핑하면, 필터 유닛 (349) 은 제 2 코드워드 (예컨대, 01) 를 송신하여, 디코딩중인 현재의 조합 (조합 (407)) 이 제 2 메트릭에 대한 동일한 범위를 공유하는 가장 최근에 디코딩된 조합 (조합 (403)) 과 동일한 필터에 맵핑된다는 것을 나타낼 수 있다.

[0096]

예를 들어, 조합 (410) 이 디코딩중인 현재의 조합이면, 조합 (409) 은 메트릭 1 에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합이며, 조합 (406) 은 메트릭 2 에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합이다. 조합 (410) 이 제 1 메트릭에 대해 동일한 범위 (즉, 메트릭 1 에 대한 범위 (1-2)) 를 공유하는 가장 최근에 디코딩된 조합과 동일한 필터 (도 4a 의 필터 2) 에 맵핑하면, 필터 유닛 (349) 은 제 1 코드워드 (예컨대, 00) 를 송신하여, 디코딩중인 현재의 조합 (조합 (410)) 이 제 1 메트릭에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합 (조합 (409)) 과 동일한 필터 (필터 2) 에 맵핑한다는 것을 나타낼 수 있다.

[0097]

예를 들어, 조합 (411) 이 디코딩중인 현재의 조합이면, 조합 (410) 은 메트릭 1 에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합이고, 조합 (407) 은 메트릭 2 에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합이다. 조합 (411) 이 조합 (410) 또는 조합 (407) 중 어느 하나와 동일한 필터에 맵핑되지 않으면, 필터 유닛 (349) 은 제 3 코드워드 (예컨대, 1 + 추가적인 비트들) 를 송신하여, 디코딩중인 현재의 조합 (조합 (411)) 이 메트릭 1 에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합 및 메트릭 2 에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합 양자와는 상이한 필터 (필터 3) 에 맵핑한다는 것을 나타낼 수 있다.

- [0098] 메트릭 1에 대해 동일한 범위를 공유하는 조합 또는 메트릭 2에 대해 동일한 범위를 공유하는 조합이 아직 디코딩되지 않은, 그들 현재의 조합들에 대해, 그들 옵션들은 이용불가능한 것으로 간주되거나 또는 상이한 조합으로 대체될 수 있다. 예를 들어, 조합 (409)이 디코딩되는 현재의 조합이면, 조합 (405)이 메트릭 2에 대해 동일한 범위를 공유하는 가장 최근에 디코딩된 조합이며, 그러나, 메트릭 1에 대한 범위를 공유하는 조합은 아직 디코딩되어 있지 않다. 이런 경우들에서, 메트릭 1에 대한 범위를 공유하는 가장 최근에 디코딩된 조합이 디코딩중인 현재의 조합과 동일한 필터에 맵핑되지 않는 것으로 가정될 수 있다. 따라서, 이 경우, 제 1 코드워드가 조합 (409)에 대해 사용되지 않을 것이다. 이의 대안으로, 메트릭 1에 대한 범위를 공유하는 조합이 가장 최근에 디코딩된 조합 또는 상이한 이전에 디코딩된 조합과 같은, 또다른 조합으로 대체될 수 있다. 이런 경우에, 조합 (409) 이전에 가장 최근에 디코딩된 조합은 조합 (408)일 것이다. 따라서, 조합 (408)이 조합 (409)과 동일한 필터에 맵핑하면, 필터 유닛 (349)은 제 1 코드워드를 생성할 것이다. 유사한 기법들이 메트릭 1에 대해 공통 범위를 공유하는 이전 조합이 아직 디코딩되지 않은 그들 조합들에 대해 사용될 수 있다.
- [0099] 송신 순서에서 제 1 조합 (예컨대, 도 4a의 예에서 조합 (401))에 대해, 메트릭 1에 대해 동일한 범위를 공유하는 조합, 또는 메트릭 2에 대해 동일한 범위를 공유하는 조합 어느 것도 디코딩되지 않은 경우, 필터 유닛 (349)은 제 1 조합에 맵핑하는 필터를 나타내는 코드워드를 생성할 수 있다. 필터는 예를 들어, 제 3 코드워드를 이용하여 시그널링될 수도 있거나, 또는 상이한 기법을 이용하여 시그널링될 수도 있으며, 이 경우, 본 개시물에서 설명하는 기법들은 송신 순서에서의 제 2 조합 또는 후속 조합에서 시작할 것이다.
- [0100] 본 개시물의 다른 기법에 따르면, 필터 유닛 (349)은 코드워드들의 시리즈를 이용하여, 디코더에의 맵핑을 시그널링할 수 있다. 일부 구현예들에서, 필터 유닛 (349)은 제 1 코드워드를 이용하여, 디코딩중인 현재의 조합이 제 1 메트릭에 대한 동일한 범위를 공유하는 가장 최근에 디코딩된 조합과 동일한 필터에 맵핑되는지를 나타낼 수 있다. 디코딩중인 현재의 조합이 제 1 메트릭에 대한 범위를 공유하는 가장 최근에 디코딩된 조합과 동일한 필터에 맵핑하지 않으면, 필터 유닛 (349)은 제 1 코드워드 대신, 디코딩중인 현재의 조합에 맵핑하는 필터를 나타내는 제 2 코드워드를 생성할 수 있다. 이 예에서, 제 1 코드워드는 제 2 코드워드에 비해 상대적으로 짧을 수도 있다. 예를 들어, 제 1 코드워드는 1 비트들 (예컨대, 0)일 것이며, 반면, 제 2 코드워드는 더 많은 비트들 (예컨대, 1의 제 1 비트, 더하기 추가적인 비트들)이다. 현재의 조합이 메트릭 1 또는 메트릭 2에 대해 동일한 범위를 공유하는 이전에 디코딩된 조합과 동일한 필터에 맵핑하면 짧은 코드워드가 생성될지도 모르는 이전 기법과는 달리, 이 기법은 현재의 조합이 메트릭 1에 대해 동일한 범위를 공유하는 이전에 디코딩된 조합과 동일한 필터에 맵핑하면, 오직 짧은 코드워드만을 생성하는 것을 포함한다. 따라서, 비록 현재의 조합이 메트릭 2에 대해 동일한 범위를 공유하는 이전에 디코딩된 조합과 동일한 필터에 맵핑하더라도, 필터 유닛 (349)은 제 2 코드워드 (예컨대, 1 + 추가적인 비트들)를 여전히 생성한다. 본 개시물은 설명의 목적들을 위해 메트릭 1을 이용하고 있지만, 동일한 기법들이 또한 오직 메트릭 2를 이용하여 적용될 수 있다.
- [0101] 여전히, 본 개시물의 다른 기법에 따르면, 필터 유닛 (349)은 상이한 코드워드들의 시리즈를 이용하여, 디코더에의 맵핑을 시그널링할 수 있다. 예를 들어, 필터 유닛 (349)은 제 1 코드워드를 생성하여, 임의의 경우, 현재의 조합이 이전에 디코딩된 조합과 공통으로 어떤 범위를 가지는지에 상관없이, 디코딩중인 현재의 조합이 가장 최근에 디코딩된 조합과 동일한 필터에 맵핑하는 지를 나타낼 수 있다. 디코딩중인 현재의 조합이 가장 최근에 디코딩된 조합과 동일한 필터에 맵핑하지 않으면, 필터 유닛 (349)은 현재의 조합에 맵핑하는 필터를 식별하는 제 2 코드워드를 생성할 수 있다. 이 특징의 구현예에서, 제 1 코드워드는 제 2 코드워드에 비해 상대적으로 짧을 수도 있다. 예를 들어, 제 1 코드워드는 하나의 비트들 (예컨대, 0)일 것이지만, 제 2 코드워드는 더 많은 비트들 (예컨대, 1의 제 1 비트, 더하기 추가적인 비트들)이다.
- [0102] 또, 도 4a의 예 및 상하, 좌우 송신 순서를 이용하여, 조합 (401)은 조합 (402)이 현재 디코딩중이면 가장 최근에 디코딩된 조합일 것이며, 조합 (402)은 조합 (403)이 현재의 조합이면 가장 최근에 디코딩된 조합, 등 동일 것이다. 조합 (405)이 디코딩중인 현재의 조합인 경우, 404는 가장 최근에 디코딩된 조합일 것이다. 따라서, 필터 유닛 (349)은 조합 (402)이 조합 (401)과 동일한 필터에 맵핑하는 경우, 조합 (403)이 조합 (402)과 동일한 필터에 맵핑하는 등등의 경우에, 제 1 코드워드를 생성할 수 있다. 그렇지 않으면, 필터 유닛 (349)은 현재의 조합에 맵핑하는 필터를 식별하는 제 2 코드워드를 생성할 수 있다.
- [0103] 여전히, 본 개시물의 다른 기법에 따르면, 필터 유닛 (349)은 2개의 코드워드들을 이용하여, 조합들에의 필터들의 맵핑을 시그널링할 수 있다. "0"과 같은 제 1 코드워드가, 현재의 조합이 이전 조합과 동일한 필터를 이용한다는 것을 시그널링하는데 사용될 수 있다. "1"과 같은 제 2 코드워드가, 현재의 조합이 이전 조합

과는 상이한 필터를 갖는다는 것을 시그널링하는데 사용될 수 있다. 제 2 코드워드는, 그러나, 새로운 필터를 식별할 필요는 없다. 대신, 새로운 필터는 클래스에 대한 송신 순서 및 필터 계수들이 송신되는 순서에 기초하여 결정될 수 있다. 도 4b 에 대해 일 예로서 위에서 설명한, 좌우, 하상 송신 순서를 이용하여, 코드워드들은 421 (0) => 422 (0) => 423 (1) => 424 (0) => 425 (0) => 426 (0) => 427 (0) => 428(1) => 429 (0) => 430 (0) => 431 (0) => 432 (1) => 433 (0) => 434 (0) => 435 (0) 에 따라서 송신될 것이며, 이때, 괄호 내 번호는 그 조합에 대한 코드워드를 나타낸다. 이 예에서, 조합들 (421-422) 은 제 1 필터에, 조합들 (423-427) 은 제 2 필터에, 조합들 (428-431) 은 제 3 필터에, 그리고, 조합들 (432-435) 은 제 4 필터에 맵핑될 것이다. 제 1 필터, 제 2 필터, 제 3 필터, 및 제 4 필터에 대한 계수들은 필터 계수들의 세트들이 시그널링되는 순서에 대응할 수 있으며, 여기서, 시그널링된 필터 계수들의 제 1 세트는 제 1 필터에 대응하고, 시그널링된 필터 계수들의 제 2 세트는 제 2 필터에 대응하며, 그리고 기타 등등으로 대응한다. 필터 계수들의 세트들을 송신하는 순서를 결정하는 것이 아래에서 좀더 자세하게 설명된다.

[0104] 범위들의 조합들에의 필터들의 맵핑을 시그널링하는 본 개시물에서 설명되는 여러 기법들은 상호 배타적인 대안들이기 보다는, 서로 함께 사용될 수도 있다. 예를 들어, 일부 구현예들에서, 어떤 조합들은 제 1 기법을 이용하여 시그널링될 것이며, 반면 다른 조합들은 제 2 기법을 이용하여 시그널링된다. 일 예로서, 메트릭 1 에 대해 동일한 범위를 공유하는 조합 또는 메트릭 2 에 대해 동일한 범위를 공유하는 조합 중 하나가 아직 디코딩되지 않은 경우 (예컨대, 조합들 (402, 403, 404, 405, 409, 및 413)), 필터 유닛 (349) 은 제 1 기법을 이용할 수도 있다. 양자의 메트릭 1 에 대해 동일한 범위를 공유하는 조합 및 메트릭 2 에 대해 동일한 범위를 공유하는 조합 양자가 디코딩된 경우 (예컨대, 조합들 (406, 407, 408, 410, 411, 412, 414, 415, 및 416)), 제 2 기법이 사용될 수 있을 것이다. 게다가, 위에서 설명한 제 1, 제 2, 및 제 3 코드워드들 중 임의의 코드워드에 사용되는 코드워드들은 고정된 길이 코드워드들, 가변 길이 코드워드들, 또는 컨텍스트-적응적 가변 길이 코드워드들 중 임의의 코드워드들일 수도 있다.

[0105] 디코더로 하여금 범위들의 조합들에의 필터들의 맵핑을 재구성할 수 있게 하는 정보를 생성하는 것에 더해, 필터 유닛 (349) 은 또한 디코더로 하여금 필터들 자체를 재구성할 수 있도록 하는 정보를 생성한다. 필터들을 재구성하는 것은 필터들의 필터 계수들을 재구성하는 것을 포함한다. 이하에서 좀더 자세하게 설명하는 바와 같이, 필터 유닛 (349) 은 차분 코딩 기법들을 이용하여, 필터 계수들을 시그널링할 수 있다. 차분 코딩 기법을 이용하기 위해, 필터 유닛 (349) 은 필터 계수들의 세트들을 시그널링하는 순서를 결정한다.

[0106] 순서를 결정하는 것의 부분으로서, 필터 유닛 (349) 은 범위들의 각각의 조합에 대한 순차적인 값을 나타내는 조합 식별 (ID) 을 결정한다. 도 4a 를 일 예로서 이용하여, 조합들은 순차적인 값들을 좌우, 상하 순서로 나타내는 할당된 조합 ID들일 것이며, 이 경우, 조합 (401) 은 제 1 순차적인 값을 할당받을 것이며, 조합 (402) 은 제 2 순차적인 값을 할당받을 것이며, 나머지 조합들은 순차적인 값들을 403 => 404 => 405 => 406 => 407 => 408 => 409 => 410 => 411 => 412 => 413 => 414 => 415 => 416 의 순서로 할당받을 것이다. 필터 유닛 (349) 은 또한 조합 ID들을 상하, 지그-재그 순서를 이용하여 할당할 것이며, 이 경우, 그 조합들은 401 => 402 => 403 => 404 => 408 => 407 => 406 => 405 => 409 => 410 => 411 => 412 => 416 => 415 => 414 => 413 의 순서인 순차적인 값들을 가진 조합 ID들을 할당받을 것이다. 필터 유닛 (349) 은 또한 조합 ID들을 상하, 좌우 순서를 이용하여 할당할 것이며, 이 경우, 이 조합들은 401 => 405 => 409 => 413 => 402 => 406 => 410=> 414 => 403 => 407 => 411 => 415 => 404 => 408 => 412 => 416 의 순서인 순차적인 값들로 조합 ID들을 할당받는다. 필터 유닛 (349) 은 또한 좌우, 지그-재그 순서를 이용할 것이며, 여기서, 이 조합들은 401 => 405 => 409 => 413 => 414 => 410 => 406=> 402 => 403 => 407 => 411 => 415 => 416 => 412 => 408 => 404 의 순서로 순차적인 값들을 가진 조합 ID들을 할당받는다. 추측될 수 있는 바와 같이, 이들은 사용될 수 있는 많은 순서들 중 단지 몇 개이다. 더욱이, 설명하는 순서들 중 임의의 순서는 최저-최고 또는 최고-최저일 수 있다.

[0107] 필터 유닛 (349) 이 범위 조합들에의 필터들의 맵핑을 결정한 후, 필터 유닛 (349) 은 동일한 필터에 맵핑되는 범위 조합들의 그룹핑들을 식별할 수 있다. 도 4a 를 일 예로서 이용하여, 그룹핑들은 다음과 같을 것이다.

[0108] 필터 1 그룹: 조합들 413, 414, 및 415

[0109] 필터 2 그룹: 조합들 409, 410

[0110] 필터 3 그룹: 조합들 411 및 412

[0111] 필터 4 그룹: 조합 416

[0112] 필터 5 그룹: 조합들 401 및 405

[0113] 필터 6 그룹: 조합들 402 및 406

[0114] 필터 7 그룹: 조합들 403 및 407

[0115] 필터 8 그룹: 조합들 404 및 408.

[0116] 필터 유닛 (349) 은 그후 각각의 그룹에 그룹 ID 를 할당할 수 있으며, 그룹 ID 는 순차적인 값을 나타낼 수 있다. 그룹 ID 들은 그 그룹을 포함하는 조합들과 연관되는 순차적인 값들에 기초하여 그룹들에 할당될 수 있다. 예를 들어, 조합 ID 들에 기초하여 최저 연관되는 순차적인 값을 가진 조합을 가지는 그룹은 최저 순차적인 값을 가진 그룹 ID 를 할당받을 것이다. 나머지 그룹들 중, 최저 연관되는 순차적인 값을 가진 조합을 갖는 나머지 그룹은 다음 최저 순차적인 값을 가진 그룹 ID 을 할당받을 수 있다. 이 프로세스는 모든 그룹들이 그룹 ID 를 할당받을 때까지 반복할 수 있다. 일부 구현예들에서, 그룹 ID 들은 최저 대신, 최고 연관되는 순차적인 값들을 가진 조합들에 기초하여 할당될 것이다. 일부 구현예들에서, 조합 ID 들에 기초하여 최저 연관되는 순차적인 값을 가진 조합을 가지는 그룹은 최고 순차적인 값을 가진 그룹 ID 를 할당받을 것이며, 또는 반대의 경우도 마찬가지이다.

[0117] 또, 도 4a 를 일 예로서 이용하고, 그리고, 조합들 (401-416) 이 순차적인 값들을 가진 조합 ID 들에 좌우, 상하 순서로 할당된다고 가정하면, 필터 유닛 (349) 은 아래 표 1 에 나타난 바와 같이, 그룹 ID 들을 필터 그룹들에 할당할 수 있다.

표 1

[0118]

그룹 이름	그룹에서의 조합	최저 순차적인 값을 가진 조합	그룹 ID
필터 1 그룹	413, 414, 415	413	7
필터 2 그룹	409, 410	409	5
필터 3 그룹	411, 412	411	6
필터 4 그룹	416	416	8
필터 5 그룹	401, 405	401	1
필터 6 그룹	402, 406	402	2
필터 7 그룹	403, 407	403	3
필터 8 그룹	404, 408	404	4

[0119]

표 1 에 나타난, 도 4a 의 예에서, 필터 유닛 (349) 은 필터 5 그룹이 최저 순차적인 값을 가진 범위 조합 (즉, 조합 (401)) 을 포함하기 때문에, 필터 5 그룹에 최저 순차적인 값을 가진 그룹 ID 를 할당한다. 필터 유닛 (349) 은 나머지 필터 그룹들 (즉, 필터 5 그룹을 제외한 모든 그룹들) 중에서, 필터 6 그룹이 제 2 최저 순차적인 값을 가진 범위 조합 (즉, 조합 (402)) 을 포함하기 때문에, 필터 6 그룹에 제 2 최저 순차적인 값을 가진 그룹 ID 를 할당한다. 필터 유닛 (349) 은 나머지 필터 그룹들 (즉, 필터 5 그룹 및 필터 6 그룹을 제외한 모든 필터 그룹들) 중에서, 필터 7 그룹이 최저 순차적인 값을 가진 범위 조합 (즉, 조합 (403)) 을 포함하기 때문에, 필터 7 그룹에 제 3 최저 순차적인 값을 가진 그룹 ID 을 할당한다. 필터 유닛 (349) 은 나머지 필터 그룹들 (즉, 필터 5 그룹, 필터 6 그룹, 및 필터 7 그룹을 제외한 모든 필터 그룹들) 중에서, 필터 8 그룹이 제 4 최저 순차적인 값을 가진 범위 조합 (조합 (404)) 을 포함하기 때문에, 필터 8 그룹에 제 4 최저 순차적인 값을 가진 그룹 ID 을 할당한다. 필터 유닛 (349) 은 나머지 필터 그룹들 (즉, 필터 5 그룹, 필터 6 그룹, 필터 7 그룹, 및 필터 8 그룹을 제외함) 중에서, 필터 2 그룹이 최저 순차적인 값을 가진 범위 조합 (조합 (409)) 을 포함하기 때문에, 필터 2 그룹에 제 5 최저 순차적인 값을 가진 그룹 ID 을 할당한다. 필터 유닛 (349) 은 나머지 필터 그룹들 (즉, 필터 5 그룹, 필터 6 그룹, 필터 7 그룹, 필터 8 그룹, 및 필터 2 그룹을 제외함) 중에서, 필터 3 그룹이 최저 순차적인 값을 가진 범위 조합 (조합 (411)) 을 포함하기 때문에, 필터 3 그룹에 제 6 최저 순차적인 값을 가진 그룹 ID 을 할당한다. 필터 유닛 (349) 은 나머지 필터 그룹들 (즉, 필터 5 그룹, 필터 6 그룹, 필터 7 그룹, 필터 8 그룹, 및 필터 2 그룹을 제외함) 중에서, 필터 1 그룹이 최저 순차적인 값 범위 조합 (조합 (413)) 을 포함하기 때문에, 필터 1 그룹에 제 7 최저 순차적인 값을 가진 그룹 ID 을 할당한다. 마지막으로, 필터 유닛 (349) 은 필터 4 그룹, 최종 나머지 필터 그룹, 최고 순차적인 값 (이 특정의 예에서는, 8) 을 가진 그룹 ID 를 할당한다.

[0120]

필터 그룹 ID 들에 기초하여, 필터 유닛 (349) 은 필터의 필터 계수들을 시그널링하는 순서를 결정한다. 또,

도 4a 의 예 및 표 1 를 이용하여, 필터 유닛 (349) 은 먼저 필터 5 에 대한 계수, 그 다음, 필터 6 에 대한 계수, 그 다음, 필터 7 에 대한 계수, 그 다음, 필터 8 에 대한 계수, 그 다음, 필터 2 에 대한 계수, 그 다음, 필터 3 에 대한 계수, 그 다음, 필터 1 에 대한 계수, 그리고 마지막으로, 필터 4 에 대한 계수를 시그널링한다. 본 개시물에서 설명하는 바와 같이, 차분 코딩 기법들을 이용하여, 필터 유닛 (349) 은 그룹 ID들의 순차적인 순서정렬 (ordering) 에 기초하여, 필터 6 에 대한 계수들을 필터 5 의 필터 계수들에 상대적인 차이 정보로서 코딩하고, 필터 7 에 대한 계수들을 필터 6 에 대한 필터 계수들에 상대적인 차이 정보로서 코딩하고, 그리고 등등으로 코딩할 수도 있다.

[0121]

필터들에의 입력들에 대한 2개 이상의 메트릭들의 맵핑은 다수의 방법들로 구현될 수 있다. 예를 들어, 일부 구현예들에서, 각각의 입력은 필터들의 고유한 세트를 가질 것이며, 반면, 일부 구현예들에서 입력들은 필터들의 공통 세트를 공유한다. 게다가, 일부 구현예들에서, 각각의 입력에 대한 2개 이상의 메트릭들이 각각의 입력에 대한 특정의 필터를 식별하는데 사용될 것이다. 다른 구현예들에서는, 그러나, 단일 입력에 대한 2개 이상의 메트릭들이 모든 입력들에 대한 필터들을 식별하는데 사용될 것이다. 여전히 다른 구현예들에서, 제 1 입력에 대한 2개 이상의 메트릭들이 제 2 의 상이한 입력에 대한 필터를 식별하는데 사용될 것이다.

[0122]

본 개시물에 따르면, 필터 유닛 (349) 은 필터 정보를 인코딩하여 인코더 (350) 로부터 또다른 디바이스로 운반하는데 필요로 하는 데이터의 양을 감소시킬 수도 있는 필터 정보에 대해 코딩 기법들을 수행할 수도 있다. 또, 각각의 프레임 또는 슬라이스에 대해, 필터 유닛 (349) 은 그 프레임 또는 슬라이스에 대한 CUs 의 픽셀들에 적용되는 하나 이상의 필터 계수들의 세트들을 정의하거나 또는 선택할 수도 있다. 필터 유닛 (349) 은 인-루프 필터링에 부합하는 예측 코딩에 사용될 수도 있는, 메모리 (334) 에 저장된 재구성된 비디오 프레임들의 비디오 블록들을 필터링하기 위해서, 필터 계수들을 적용한다. 필터 유닛 (349) 은 필터 계수들을 필터 정보로서 인코딩할 수도 있으며, 이 필터 정보는 인코딩된 비트스트림에의 포함을 위해서 엔트로피 인코딩 유닛 (346) 으로 포워딩된다.

[0123]

게다가, 본 개시물의 기법들은 FSU (353) 에 의해 정의된 또는 선택된 필터 계수들 중 일부가 또다른 프레임 또는 슬라이스의 CUs 의 픽셀들에 대해 적용된 다른 필터 계수들과 매우 유사할 수도 있다는 사실을 이용할 수도 있다. 동일한 유형의 필터가 상이한 프레임들 또는 슬라이스들에 대해 적용될 수도 있으며 (예컨대, 동일한 필터 서포트), 그러나, 그 필터들은 필터 서포트의 상이한 인덱스들과 연관되는 필터 계수 값들의 관점에서 상이할 수도 있다. 따라서, 이런 필터 계수들을 전달하는데 필요로 하는 데이터의 양을 감소시키기 위해, 필터 유닛 (349) 은 필터링에 사용될 하나 이상의 필터 계수들을 또다른 CU 의 필터 계수들에 기초하여, 잠재적으로는, 필터 계수들 사이의 유사성들을 이용하여, 예측 인코딩할 수도 있다. 일부의 경우, 그러나, 필터 계수들을 직접, 예컨대, 임의의 예측을 이용함이 없이, 인코딩하는 것이 바람직할 수도 있다. 예측 코딩 기법들을 이용하여 필터 계수들을 인코딩할 시점 및 임의의 예측 코딩 없이 필터 계수들을 직접 인코딩할 시점을 정의하는 활동 메트릭의 사용을 이용하는 기법들과 같은, 필터 계수들을 디코더로 효율적으로 통신하는 여러 기법들이 이용될 수 있다. 게다가, 디코더에 의해 알려진 계수들 (예컨대, 5, -2, 10) 의 알려진 서브세트가 계수들 (예컨대, 5, -2, 10, 10, -2, 5) 의 풀 세트를 정의하는데 사용될 수 있도록, 대칭이 또한 도입될 수도 있다. 대칭은 직접 코딩 시나리오 및 예측 코딩 시나리오 양자에서 도입될 수도 있다.

[0124]

위에서 설명한 바와 같이, 비디오 인코더 (350) 는 픽셀들의 블록 내 픽셀들의 그룹에 대한 제 1 메트릭을 결정하고, 픽셀들의 그룹에 대한 제 2 메트릭을 결정하고, 제 1 메트릭 및 제 2 메트릭에 기초하여 필터를 결정하고, 그리고, 그 필터를 픽셀들의 그룹에 적용함으로써, 필터링된 이미지를 생성하도록 구성된 비디오 인코더의 일 예를 나타낸다. 비디오 인코더 (350) 는 또한 픽셀들의 블록에 대한 제 1 메트릭을 결정하고; 픽셀들의 블록에 대한 제 2 메트릭을 결정하고; 제 1 메트릭 및 제 2 메트릭에 기초하여 필터를 결정하고; 그리고, 그 필터를 픽셀들의 블록에 적용하여 필터링된 이미지를 생성하도록 구성된 비디오 인코더의 일 예를 나타내며, 상기 제 1 메트릭은 그 블록 내 다른 픽셀들에 대한, 그 블록 내 픽셀들의 서브세트의 비교에 기초하여 결정된다.

[0125]

위에서 설명한 바와 같이, 비디오 인코더 (350) 는 또한 필터들에의 범위 조합들의 맵핑을 결정하고; 범위 조합들에 대한 순차적인 값들에 기초하여 고유한 그룹 ID들을 범위 조합들의 그룹들에 할당하고; 그리고, 고유한 그룹 ID들에 기초하여 필터들에 대응하는 필터 계수들의 세트들을 코딩하도록 구성된 비디오 인코더의 일 예를 나타내며, 상기 범위 조합은 제 1 메트릭에 대한 범위 및 제 2 메트릭에 대한 범위를 포함하며, 상기 각각의 범위 조합은 고유한 범위 조합 식별 (ID) 을 가지며, 상기 각각의 고유한 범위 조합 ID 은 범위 조합에 대한 순차적인 값에 대응하며, 상기 각각의 고유한 그룹 ID 는 그룹에 대한 순차적인 값에 대응한다. 비디오 인코더

(350) 는 고유한 그룹 ID들의 순차적인 값들에 기초하여 선택되는 순서로, 필터 계수들의 세트들을 코딩된 비트 스트림으로 시그널링함으로써, 필터 계수들의 세트들을 코딩할 수 있다. 비디오 인코더 (350) 는 필터 계수들의 세트들을 차분 코딩 기법들을 이용하여 시그널링할 수 있다.

[0126] 위에서 설명한 바와 같이, 비디오 인코더 (350) 는 또한 필터들에의 범위 조합들의 맵핑을 결정하고; 현재의 범위 조합이 제 1 메트릭에 대한 값들의 동일한 범위를 포함하는 이전 범위 조합과 동일한 필터에 맵핑되면, 제 1 코드워드를 생성하고; 현재의 범위 조합이 제 2 메트릭에 대한 값들의 동일한 범위를 포함하는 이전 범위 조합과 동일한 필터에 맵핑되면, 제 2 코드워드를 생성하고; 그리고, 현재의 범위 조합이 제 1 메트릭에 대한 값들의 동일한 범위를 포함하는 이전 범위 조합 및 제 2 메트릭에 대한 값들의 동일한 범위를 포함하는 이전 범위 조합과는 상이한 필터에 맵핑되면, 제 3 코드워드를 생성하도록 구성된 비디오 인코더의 일 예를 나타내며, 상기 범위 조합은 제 1 메트릭에 대한 값들의 범위 및 제 2 메트릭에 대한 값들의 범위를 포함한다. 비디오 인코더 (350) 는 또한 필터들에의 범위 조합들의 맵핑을 결정하고; 현재의 범위 조합이 이전 범위 조합과 동일한 동일한 필터에 맵핑되면, 제 1 코드워드를 생성하고; 그리고, 현재의 범위 조합이 이전 범위 조합과는 상이한 필터에 맵핑되면, 제 2 코드워드를 생성하도록 구성된 비디오 인코더의 일 예를 나타내며, 상기 범위 조합은 제 1 메트릭에 대한 범위 및 제 2 메트릭에 대한 범위를 포함하고, 상기 제 2 코드워드는 현재의 범위 조합에 맵핑된 필터를 식별한다.

[0127] 도 5 는 본원에서 설명하는 방법으로 인코딩되는 비디오 시퀀스를 디코딩하는 비디오 디코더의 일 예 (560) 를 예시하는 블록도이다. 수신된 비디오 시퀀스는 인코딩된 이미지 프레임들의 세트, 프레임 슬라이스들의 세트, 일반적으로 코딩된 화상들의 그룹 (GOPs), 또는 인코딩된 비디오 블록들을 포함하는 매우 다양한 유형들의 비디오 블록들의 시리즈 및 이런 비디오 블록들을 디코딩하는 방법을 정의하는 구문을 포함할 수도 있다.

[0128] 비디오 디코더 (560) 는 엔트로피 디코딩 유닛 (552) 을 포함하며, 이 엔트로피 디코딩 유닛은 도 3 의 엔트로피 인코딩 유닛 (346) 에 의해 수행되는 인코딩의 반대의 디코딩 기능을 수행한다. 특히, 엔트로피 디코딩 유닛 (552) 은 CAVLC 또는 CABAC 디코딩, 또는 비디오 인코더 (350) 에 의해 사용되는 임의의 다른 유형의 엔트로피 디코딩을 수행할 수도 있다. 1차원 직렬화된 포맷으로 엔트로피 디코딩된 비디오 블록들은 계수들의 하나 이상의 1차원 벡터들을 2차원 블록 포맷으로 역변환하기 위해 역 스캐닝될 수도 있다. 벡터들의 개수 및 사이즈뿐만 아니라, 비디오 블록들에 대해 정의된 스캔 순서가 2차원 블록이 어떻게 재구성되는 지를 정의할 수도 있다. 엔트로피 디코딩된 예측 구문은 엔트로피 디코딩 유닛 (552) 으로부터 예측 모듈 (554) 로 전송될 수도 있으며, 엔트로피 디코딩된 필터 정보는 엔트로피 디코딩 유닛 (552) 으로부터 필터 유닛 (559) 으로 전송될 수도 있다.

[0129] 비디오 디코더 (560) 는 또한 예측 모듈 (554), 역양자화 유닛 (556), 역변환 유닛 (558), 메모리 및 합산기 (564) 를 포함한다. 게다가, 비디오 디코더 (560) 는 또한 합산기 (564) 의 출력을 필터링하는 디-블록킹 필터 (557) 를 포함한다. 본 개시물과 부합하여, 필터 유닛 (559) 은 하나 이상의 입력들에 적용되는 하나 이상의 필터들을 포함하는 엔트로피 디코딩된 필터 정보를 수신할 수도 있다. 도 5 상에 나타내지는 않지만, 디-블록킹 필터 (557) 는 또한 적용되는 하나 이상의 필터들을 포함하는 엔트로피 디코딩된 필터 정보를 수신할 수도 있다.

[0130] 필터 유닛 (559) 에 의해 적용되는 필터들은 필터 계수들의 세트들에 의해 정의될 수도 있다. 필터 유닛 (559) 은 엔트로피 디코딩 유닛 (552) 으로부터 수신된 필터 정보에 기초하여 필터 계수들의 세트들을 생성하도록 구성될 수도 있다. 필터 정보는 예를 들어, 필터들의 세트 내 필터들의 최대 개수 및/또는 필터들의 세트 내 필터들의 형태를 식별하는 필터 기술 구문을 포함할 수도 있다. 필터 기술 구문은 LCU 헤더, 프레임 헤더, 슬라이스 헤더, GOP 헤더, 시퀀스 헤더 등과 같은 비디오 블록들의 시리즈의 헤더에 포함될 수 있다. 다른 예들에서, 필터 기술 구문은 꼬리말 (footer) 또는 다른 데이터 구조에 포함될 것이다. 필터 기술 구문에 기초하여, 필터 유닛 (559) 은 인코더에서 사용된 필터들의 세트를 재구성할 수 있다.

[0131] 필터 정보는 또한 임의의 주어진 계수들의 세트에 대해 사용되는 인코딩의 방법을 디코더로 시그널링하는 추가적인 시그널링 구문을 포함할 수도 있다. 일부 구현예들에서, 필터 정보는 예를 들어, 또한 임의의 주어진 계수들의 세트가 사용되어야 하는 2개 이상의 메트릭들에 대한 범위들을 포함할 수도 있다. 필터들의 디코딩에 뒤이어, 필터 유닛 (559) 은 하나 이상의 필터 계수들의 세트들 및 상이한 필터 계수들의 세트들이 사용되어야 하는 범위들을 포함하는 시그널링 구문에 기초하여, 디코딩된 비디오 블록들의 픽셀 값들을 필터링할 수도 있다.

[0132] 필터 유닛 (559) 은 2개 이상의 메트릭들에의 필터들의 맵핑 뿐만 아니라, 각각의 프레임 또는 슬라이스에 대한

필터들의 세트를 나타내는 하나 이상의 구문 엘리먼트들을 비트스트림으로 수신할 수도 있다. 예를 들어, 인코더가 도 4a 에 나타난 필터들에의 메트릭들에 대한 범위들의 맵핑을 이용하면, 인코더는 이 맵핑 또는 송신 데이터를 시그널링하여, 필터 유닛 (559) 으로 하여금 이 맵핑을 재구성하도록 할 것이다. 이 맵핑이 명시적으로 시그널링되는지 여부에 관계없이, 필터 유닛 (559) 은 인코더에 의해 사용되는 것과 같이 동일한 범위들의 조합들에의 필터들의 맵핑을 유지할 수 있다.

[0133]

위에서 언급한 바와 같이, 필터 유닛 (559) 은 비트스트림으로 시그널링되는 필터 정보에 기초하여 맵핑을 생성한다. 이 맵핑에 기초하여, 필터 유닛 (559) 은 그룹들을 결정하고, 그 그룹들에 그룹 ID들을 필터 유닛 (349) 과 관련하여 위에서 설명한 동일한 방법으로 할당할 수 있다. 이들 그룹 ID들을 이용하여, 필터 유닛 (559) 은 수신된 필터 계수들을 연관시킬 수 있다.

[0134]

프레임 또는 슬라이스 내 각각의 CU 에 대해, 필터 유닛 (559) 은 그 세트(들) 중 어느 필터(들) 을 각각의 입력에 적용할지를 결정하기 위해, 다수의 입력들에 대한 CU 의 디코딩된 픽셀들과 연관되는 하나 이상의 메트릭들 (즉, PI, EI, pRI, 및 RI) 을 계산할 수 있다. 이의 대안으로, 필터 유닛 (559) 은 pRI 또는 RI 과 같은, 단일 입력에 대한 하나 이상의 메트릭들을 계산할 수도 있다. 필터 유닛 (559) 은 특정의 픽셀 또는 픽셀들의 그룹에 대해 결정된 메트릭들에 기초하여 어느 필터를 적용할지를 결정한다. 합산-수정된 라플라시안 값 및 방향을 메트릭 1 및 메트릭 2 에 대한 예들로서 이용하고 도 4a 에 나타난 맵핑들을 일 예로서 이용하여, 픽셀 또는 픽셀들의 그룹이 범위 (1-2) 에서의 합산-수정된 라플라시안 값 및 범위 (2-3) 에 대응하는 방향을 갖는다고 필터 유닛 (559) 이 결정하면, 필터 유닛 (559) 은 필터 2 를 그 픽셀 또는 픽셀들의 그룹에 적용할 수 있다. 픽셀 또는 픽셀들의 그룹이 범위 (1-4) 에서의 합산-수정된 라플라시안 값 및 범위 (2-2) 에 대응하는 방향을 갖는다고 필터 유닛 (559) 이 결정하면, 필터 유닛 (559) 은 필터 6 을 그 픽셀 또는 픽셀들의 그룹에 적용할 수 있으며, 기타 등등으로 적용할 수 있다. 필터는 일반적으로 임의 종류의 필터 서포트 형태 또는 배열을 가정할 수도 있다. 필터 서포트는 필터링중인 주어진 픽셀에 대한 필터의 형태를 지칭하며, 필터 계수들은 필터 서포트에 따라서, 이웃하는 픽셀 값들에 적용되는 가중치를 정의할 수도 있다. 본 개시물의 기법들에 따르면, 구문 데이터는 그 필터들이 어떻게 인코딩되었는지 (예컨대, 그 필터 계수들이 어떻게 인코딩되었는지) 뿐만 아니라, 상이한 필터들이 사용되어야 하는 활동 메트릭의 범위들을 디코더로 시그널링하기 위해 비트스트림에 포함될 수도 있다.

[0135]

프레임 또는 슬라이스 내 각각의 CU 에 대해, 필터 유닛 (559) 은 그 세트(들) 중 어느 필터(들) 을 각각의 입력에 적용할지를 결정하기 위해, 다수의 입력들에 대한 CU 의 디코딩된 픽셀들과 연관되는 하나 이상의 메트릭들 (즉, PI, EI, pRI, 및 RI) 을 계산할 수 있다. 이의 대안으로, 필터 유닛 (559) 은 pRI 또는 RI 과 같은, 단일 입력에 대한 하나 이상의 메트릭들을 계산할 수도 있다. 필터 유닛 (559) 은 특정의 픽셀 또는 픽셀들의 그룹에 대해 결정된 메트릭들에 기초하여 어느 필터를 적용할지를 결정한다. 합산-수정된 라플라시안 값 및 방향을 메트릭 1 및 메트릭 2 에 대한 예들로서 이용하고 도 4a 에 나타난 맵핑들을 일 예로서 이용하여, 픽셀 또는 픽셀들의 그룹이 범위 (1-2) 에서의 합산-수정된 라플라시안 값 및 범위 (2-3) 에 대응하는 방향을 갖는다고 필터 유닛 (559) 이 결정하면, 필터 유닛 (559) 은 필터 2 를 그 픽셀 또는 픽셀들의 그룹에 적용할 수 있다. 픽셀 또는 픽셀들의 그룹이 범위 (1-4) 에서의 합산-수정된 라플라시안 값 및 범위 (2-2) 에 대응하는 방향을 갖는다고 필터 유닛 (559) 이 결정하면, 필터 유닛 (559) 은 필터 6 을 그 픽셀 또는 픽셀들의 그룹에 적용할 수 있으며, 기타 등등으로 적용할 수 있다. 필터는 일반적으로 임의 종류의 필터 서포트 형태 또는 배열을 가정할 수도 있다. 필터 서포트는 필터링중인 주어진 픽셀에 대한 필터의 형태를 지칭하며, 필터 계수들은 필터 서포트에 따라서, 이웃하는 픽셀 값들에 적용되는 가중치를 정의할 수도 있다. 본 개시물의 기법들에 따르면, 구문 데이터는 그 필터들이 어떻게 인코딩되었는지 (예컨대, 그 필터 계수들이 어떻게 인코딩되었는지) 뿐만 아니라, 상이한 필터들이 사용되어야 하는 활동 메트릭의 범위들을 디코더로 시그널링하기 위해 비트스트림에 포함될 수도 있다.

[0136]

예측 모듈 (554) 은 엔트로피 디코딩 유닛 (552) 으로부터 (모션 벡터들과 같은) 예측 구문을 수신한다. 예측 구문을 이용하여, 예측 모듈 (554) 은 비디오 블록들을 코딩하는데 사용되었던 예측 블록들을 생성한다. 역양자화 유닛 (556) 은 역양자화를 수행하고, 역변환 유닛 (558) 은 역변환들을 수행하여, 잔여 비디오 블록들의 계수들을 픽셀 도메인으로 반대로 변화시킨다. 가산기 (564) 는 비디오 블록을 재구성하기 위해 각각의 예측 블록을 역변환 유닛 (558) 에 의해 출력된 대응하는 잔여 블록과 합산한다.

[0137]

필터 유닛 (559) 은 그 CU 의 재구성된 비디오 블록들을 필터링하기 위해, CU 의 각각의 입력에 대해 적용되는 필터 계수들을 생성하고, 그 후 이런 필터 계수들을 적용한다. 필터링은, 예를 들어, 에지들을 평활화하거나 및/또는 비디오 블록들과 연관되는 아티팩트들을 제거하는 추가적인 디블록 필터링 (deblock filtering), 양자

화 잡음을 감소시키는 노이즈제거 필터링, 또는 코딩 품질을 향상시킬 수 있는 임의의 다른 유형의 필터링을 포함할 수도 있다. 필터링된 비디오 블록들은 비디오 정보의 디코딩된 프레임들 (또는, 다른 디코딩가능한 유닛들) 을 재구성하기 위해, 메모리 (562) 에 축적된다. 디코딩된 유닛들은 사용자에게의 프리젠테이션을 위해 비디오 디코더 (560) 로부터 출력될 수도 있으며, 그러나, 또한 후속 예측 디코딩에서 사용하기 위해 저장될 수도 있다.

[0138] 비디오 코딩의 분야에서는, 디코딩된 비디오 신호의 품질을 향상시키기 위해서 인코더 및 디코더에서 필터링을 적용하는 것이 일반적이다. 필터링은 사후-필터를 통해서 적용될 수 있으며, 이 경우 필터링된 프레임은 향후 프레임들의 예측에 사용되지 않는다. 이의 대안으로, 필터링은 "인-루프" 에 적용될 수 있으며, 이 경우, 필터링된 프레임은 향후 프레임들을 예측하는데 사용될 수도 있다. 바람직한 필터는 원래 신호와 디코딩된 필터링된 신호 사이의 에러를 최소화함으로써 설계될 수 있다. 일반적으로, 이런 필터링은 하나 이상의 필터들을 재구성된 이미지에 적용하는 것에 기초하고 있다. 예를 들어, 이미지가 메모리에 저장되기 전에, 재구성된 이미지에 디블록킹 필터가 적용되거나, 또는 이미지가 메모리에 저장되기 전에, 재구성된 이미지에 디블록킹 필터 및 하나의 추가적인 필터가 적용될 것이다.

[0139] 변환 계수들의 양자화와 유사한 방법으로, 필터 $h(k,l)$ 의 계수들이 또한 양자화될 수도 있으며, 여기서, $k = -K, \dots, K$, 및 $l = -L, \dots, L$ 이다. K 및 L 는 정수 값들을 나타낼 수도 있다. 필터 $h(k,l)$ 의 계수들은 다음과 같이 양자화될 수도 있다:

$$f(k,l) = \text{round}(\text{normFact} \cdot h(k,l))$$

[0140] 여기서, normFact 은 정규화 인자이고, round 는 원하는 비트-깊이까지 양자화를 달성하기 위해 수행되는 반올림 (rounding) 연산이다. 필터 계수들의 양자화는 인코딩 동안 도 3 의 필터 유닛 (349) 에 의해 수행될 수도 있으며, 양자화 해제 또는 역양자화는 도 5 의 필터 유닛 (559) 에 의해 디코딩된 필터 계수들에 관해 수행될 수도 있다. 필터 $h(k,l)$ 는 일반적으로 임의의 필터를 나타내도록 의도된다. 예를 들어, 필터 $h(k,l)$ 는 다수의 입력들 중 임의의 입력에 적용될 수 있다. 비디오 블록과 연관되는 다수의 입력들이 상이한 필터들을 이용하는 일부 경우, 이 경우 $h(k,l)$ 와 유사한 다수의 필터들이 양자화되고, 위에서 설명한 바와 같이, 양자화 해제될 수도 있다.

[0142] 양자화된 필터 계수들은 인코딩되어, 인코더 (350) 와 연관되는 소스 디바이스로부터 디코더 (560) 와 연관되는 목적지 디바이스로 인코딩된 비트스트림의 부분으로서 전송된다. 상기 예에서, normFact 의 값은 대개 $2n$ 와 같지만, 다른 값들이 사용될 수 있다. 더 큰 normFact 의 값들은 양자화된 필터 계수들 $f(k,l)$ 이 더 나은 성능을 제공하도록, 더 정확한 양자화를 초래한다. 그러나, 더 큰 normFact 의 값들은 디코더로 시그널링하는데 더 많은 비트들을 필요로 하는 계수들 $f(k,l)$ 을 생성할 수도 있다.

[0143] 디코더 (560) 에서, 디코딩된 필터 계수들 $f(k,l)$ 이 적합한 입력에 적용될 수도 있다. 예를 들어, 디코딩된 필터 계수들이 RI 에 적용되면, 필터 계수들은 다음과 같은 사후-디블록킹된 재구성된 이미지 $RI(i,j)$ 에 적용될 수도 있으며, 여기서, $i=0, \dots, M$ 및 $j=0, \dots, N$ 이다:

$$\tilde{RI}(i,j) = \sum_{k=-K}^K \sum_{l=-L}^L f(k,l) RI(i+k, j+l) \Bigg/ \sum_{k=-K}^K \sum_{l=-L}^L f(k,l)$$

[0144] 변수들 M, N, K 및 L 은 정수들을 나타낼 수도 있다. K 및 L 은 $-K$ 로부터 K 까지 그리고 $-L$ 로부터 L 까지 2개의-치수들을 포괄하는 픽셀들의 블록을 정의할 수도 있다. 다른 입력들에 적용되는 필터들이 유사한 방법으로 적용될 수 있다.

[0146] 본 개시물의 기법들은 사후-필터 또는 인-루프 필터의 성능을 향상시킬 수도 있으며, 또한 필터 계수들 $f(k,l)$ 을 시그널링하는데 필요로 하는 비트들의 개수를 감소시킬 수도 있다. 일부의 경우, 다수의 상이한 사후-필터들 또는 인-루프 필터들이 비디오 블록의 각각의 시리즈에 대해, 예컨대, 각각의 프레임, 슬라이스, 프레임의 부분, 프레임들의 그룹 (GOP) 등에 대해 디코더로 시그널링된다. 각각의 필터에 대해, 주어진 필터가 적용되어야 하는 CUs, 매크로블록들 및/또는 픽셀들을 식별하기 위해 추가 정보가 비트스트림에 포함된다.

[0147] 프레임들은 프레임 개수 및/또는 프레임 유형 (예컨대, I 프레임들, P-프레임들 또는 B-프레임들) 에 의해 식별될 수도 있다. I-프레임들은 인트라 예측된 인트라-프레임들을 지칭한다. P-프레임들은 하나의 데이터 (예컨대, 하나의 이전 프레임) 의 리스트에 기초하여 예측된 비디오 블록들을 갖는 예측 프레임들을 지칭한다.

B-프레임들은 2개의 데이터 (예컨대, 이전 및 후속 프레임)의 리스트들에 기초하여 예측된 양방향의 예측 프레임들을 지칭한다. 매크로블록들은 매크로블록 유형들 및/또는 매크로블록을 재구성하는데 사용할 양자화 파라미터 (QP) 값들의 범위를 리스트함으로써 식별될 수 있다.

[0148] 임의의 입력에 대한, 필터 계수들 $f(k,l)$ 은 이전 CUs에 대해 시그널링된 계수들로부터의 예측을 이용하여 코딩될 수도 있다. CU의 각각의 입력 (m) (예컨대, 각각의 프레임, 슬라이스 또는 GOP)에 대해, 인코더는 M 개 필터들의 세트를 인코딩하여 송신할 수도 있다:

[0149] g_i^m , 여기서, $i=0, \dots, M-1$ 이다.

[0150] 각각의 필터에 대해, 비트스트림은 또한 그 필터가 사용되어야 하는 2개 이상의 메트릭들에 대한 범위들의 조합을 식별하기 위해 인코딩될 수도 있다.

[0151] 필터 계수들은 이전 CU에서 사용된 재구성된 필터 계수들을 이용하여 예측될 수 있다. 이전 필터 계수들은 다음과 같이 나타낼 수도 있다:

[0152] f_i^n 여기서, $i=0, \dots, N-1$ 이다.

[0153] 이 경우, CU의 개수 (n)는 현재의 필터들의 예측을 위해 사용되는 하나 이상의 필터들을 식별하는데 사용될 수도 있으며, 개수 (n)는 디코더로 인코딩된 비트스트림의 부분으로서 전송될 수도 있다. 게다가, 정보는 예측 코딩이 사용되는 2개 이상의 메트릭들에 대한 범위들의 조합들을 식별하기 위해 인코딩되어 디코더로 송신될 수 있다.

[0154] 필터 계수들 $g(k,l)$ 의 크기는 k 및 l 값들에 의존한다. 대개, 가장 큰 크기를 가진 계수는 계수 $g(0,0)$ 이다. 큰 크기들을 가질 것으로 예상되는 다른 계수들은 k 또는 l 의 값이 0과 같은 계수들이다. 이 사상 (phenomenon)은 계수들을 시그널링하는데 요구되는 비트들의 양을 추가로 감소시키기 위해 이용될 수도 있다. 인덱스 값들 k 및 l 은 기지의 필터 서포트 내 로케이션들을 정의할 수도 있다.

[0155] 계수들:

[0156] $g_i^m(k,l), i=0, \dots, M-1$

[0157] 은 각각의 프레임 (m)에 대해, 파라미터 (p)에 따라서 정의된 exp-Golomb 또는 Golomb 코드들과 같은 파라미터화된 가변 길이 코드들을 이용하여 코딩될 수도 있으며, 여기서, $i=0, \dots, M-1$ 이다. 파라미터화된 가변 길이 코드들을 정의하는 파라미터 (p)의 값을 변경함으로써, 이들 코드들은 넓은 소스 분포들의 범위를 효율적으로 나타내기 위해 사용될 수 있다. 계수들 $g(k,l)$ 의 분포 (즉, 큰 또는 작은 값들을 가질 그들의 우도)는 k 및 l 의 값들에 의존한다. 그러므로, 코딩 효율을 증가시키기 위해, 각각의 프레임 m 에 대해, 파라미터 (p)의 값이 각각의 쌍 (k,l)에 대해 송신된다. 파라미터 (p)는 다음 계수들을 인코딩할 때 파라미터화된 가변 길이 코딩을 위해 사용될 수 있다:

[0158] $g_i^m(k,l)$, 여기서, $k = -K, \dots, K$, $l = -L, \dots, L$ 이다.

[0159] 위에서 설명한 바와 같이, 비디오 디코더 (560)는 픽셀들의 블록 내 픽셀들의 그룹에 대한 제 1 메트릭을 결정하고, 픽셀들의 그룹에 대한 제 2 메트릭을 결정하고, 제 1 메트릭 및 제 2 메트릭에 기초하여 필터를 결정하고, 그리고 그 필터를 픽셀들의 그룹에 적용함으로써, 필터링된 이미지를 생성하도록 구성된 비디오 디코더의 일 예를 나타낸다. 비디오 디코더 (560)는 또한 픽셀들의 블록에 대한 제 1 메트릭을 결정하고; 픽셀들의 블록에 대한 제 2 메트릭을 결정하고; 제 1 메트릭 및 제 2 메트릭에 기초하여 필터를 결정하고; 그리고, 그 필터를 픽셀들의 블록에 적용하여 필터링된 이미지를 생성하도록 구성된 비디오 인코더의 일 예를 나타내며, 상기 제 1 메트릭은 그 블록에서의 다른 픽셀들에 대한 그 블록에서의 픽셀들의 서브세트의 비교에 기초하여 결정된다.

[0160] 위에서 설명한 바와 같이, 비디오 디코더 (560)는 또한 필터들에의 범위 조합들의 맵핑을 결정하고; 범위 조합들에 대한 순차적인 값들에 기초하여 고유한 그룹 ID들을 범위 조합들의 그룹들에 할당하고; 그리고, 고유한 그룹 ID들에 기초하여 필터들에 대응하는 필터 계수들의 세트들을 코딩하도록 구성된 비디오 디코더의 일 예를 나타내며, 상기 범위 조합은 제 1 메트릭에 대한 범위 및 제 2 메트릭에 대한 범위를 포함하며, 상기 각각의 범위

조합은 고유한 범위 조합 식별 (ID) 을 가지며, 상기 각각의 고유한 범위 조합 ID 는 범위 조합에 대한 순차적인 값에 대응하며, 상기 각각의 고유한 그룹 ID 는 그룹에 대한 순차적인 값에 대응한다. 비디오 디코더 (560) 는 코딩된 비트스트림으로 수신된 정보에 기초하여 필터 계수들의 세트들을 생성함으로써 필터 계수들의 세트들을 코딩할 수 있다. 비디오 디코더 (560) 는 필터 계수들의 세트들을 차분 코딩 기법들을 이용하여 생성할 수 있다.

[0161]

비디오 디코더 (560) 는 또한 제 1 범위 조합을 제 1 필터에 맵핑하고; 제 2 범위 조합을 제 2 필터에 맵핑하고; 현재의 범위 조합을 필터에 맵핑하도록 구성된 비디오 디코더의 일 예를 나타내며, 상기 제 1 범위 조합은 제 1 메트릭에 대한 값들의 제 1 범위 및 제 2 메트릭에 대한 값들의 제 1 범위를 포함하고, 상기 제 2 범위 조합은 제 1 메트릭에 대한 값들의 제 2 범위 및 제 2 메트릭에 대한 값들의 제 2 범위를 포함하고, 상기 현재의 범위 조합은 제 1 메트릭의 값들의 제 1 범위 및 제 2 메트릭에 대한 값들의 제 2 범위를 포함한다. 필터에 현재의 범위 조합을 맵핑하는 것은 제 1 코드워드를 수신하는 것에 응답하여 현재의 범위 조합을 제 1 필터에 맵핑하는 것; 제 2 코드워드를 수신하는 것에 응답하여 현재의 범위 조합을 제 2 필터에 맵핑하는 것; 그리고, 제 3 코드워드를 수신하는 것에 응답하여 현재의 범위 조합을 제 3 필터에 맵핑하는 것을 포함할 수 있으며, 상기 제 1 코드워드는 현재의 범위 조합이 제 1 범위 조합과 동일한 필터에 맵핑된다는 것을 나타내고, 상기 제 2 코드워드는 현재의 범위 조합이 제 2 조합과 동일한 필터에 맵핑된다는 것을 나타내며, 상기 제 3 코드워드는 그 제 3 필터를 식별한다. 비디오 디코더 (560) 는 또한 필터들에의 범위 조합들의 맵핑을 생성하고; 현재의 범위 조합이 이전 범위 조합과 동일한 동일한 필터에 맵핑된다는 것을 시그널링하는 제 1 코드워드 시그널링을 수신하는 것에 응답하여 현재의 범위 조합을 이전 범위 조합과 동일한 동일한 필터에 맵핑하고; 그리고, 현재의 범위 조합이 이전 범위 조합과는 상이한 필터에 맵핑된다는 것을 시그널링하는 제 2 코드워드를 수신하는 것에 응답하여 현재의 범위 조합을 제 2 코드워드에 의해 식별되는 필터에 맵핑하도록 구성된 비디오 디코더의 일 예를 나타내며, 상기 범위 조합은 제 1 메트릭에 대한 범위 및 제 2 메트릭에 대한 범위를 포함한다.

[0162]

위에서 소개한 바와 같이, 여러 상이한 유형들의 메트릭들이 본 개시물에서 설명하는 멀티-메트릭 필터링 기법들과 함께 사용될 수 있다. 이들 메트릭들 중 일부는 비디오 데이터 내 하나 이상의 픽셀들의 블록들과 연관되는 활동을 정량화하는 활동 메트릭들이다. 활동 메트릭들은 픽셀들의 세트 내 픽셀 분산 (pixel variance) 을 나타내는 분산 메트릭들을 포함할 수 있다. 설명하는 바와 같이, 이들 활동 메트릭들의 일부는 방향-특정적이다. 예를 들어, 수평 활동 메트릭은 활동을 수평축을 따라서 정량화하고, 수직 활동 메트릭은 활동을 수직축을 따라서 정량화하며, 대각선 활동 메트릭은 활동을 대각선 축을 따라서 정량화하며, 기타 등등으로 정량화한다.

[0163]

일부 활동 메트릭들은 방향-특정적이지 않다. 예를 들어, 합산-수정된 라플라시안 값은 현재의 픽셀 또는 현재의 픽셀들의 그룹을 둘러싸는 픽셀들의 2차원 윈도우에 기초하는 활동 메트릭이다. 현재의 픽셀 (i, j) 에 대해, 합산-수정된 라플라시안 값은 다음과 같이 계산될 수 있다:

$$\text{var}(i, j) = \sum_{k=-K}^K \sum_{l=-L}^L |2R(i+k, j+l) - R(i+k-1, j+l) - R(i+k+1, j+l) + 2R(i+k, j+l) - R(i+k, j+l-1) - R(i+k, j+l+1)| \quad (1)$$

[0164]

[0165]

여기서, -K 로부터 K 까지 그리고 -L 로부터 L 까지 포괄하는 2차원 윈도우에 대해서 k 는 -K 로부터 K 까지의 픽셀 값들의 합산 값을 나타내고 l 은 -L 로부터 L 까지의 합산 값을 나타내며, 여기서, i 및 j 는 픽셀 데이터의 픽셀 좌표들을 나타내며, RI(i, j) 는 좌표들 i 및 j 에서 주어진 픽셀 값을 나타내며, 그리고 var(i, j) 는 활동 메트릭 (즉, 합산-수정된 라플라시안 값) 이다.

[0166]

본 개시물의 기법들은 또한 수평 활동, 수직 활동, 및 대각선 활동에 대한 방향-특정의 메트릭들을 이용하여 구현될 수도 있다. 수식들 2 및 3 은 현재의 픽셀의 픽셀 값 (Rec), 예컨대 강도를 이웃하는 픽셀들의 픽셀 값과 비교함으로써 어떻게 수평 활동 및 수직 활동이 현재의 픽셀 (x, y) 에 대해 계산될 수 있는지의 예들을 나타낸다.

[0167]

$$\text{Hor_act}(x, y) = R(2 * \text{Rec}[x][y] - \text{Rec}[x+1][y] - \text{Rec}[x-1][y]) \quad (2)$$

[0168]
$$\text{Ver_act}(x,y) = R(2*\text{Rec}[x][y]-\text{Rec}[x][y+1]-\text{Rec}[x][y-1]) \quad (3)$$

[0169] 수식 2 로 나타낸 바와 같이, 수평 활동을 결정할 때, 현재의 픽셀 (x,y) 은 좌측 이웃 (x-1, y) 및 우측 이웃 (x+1, y) 과 비교될 수 있다. 수식 3 으로 나타낸 바와 같이, 수직 활동을 결정할 때, 현재의 픽셀은 상부 이웃 (x, y+1) 및 하부 이웃 (x, y-1) 과 비교될 수 있다.

[0170] 수식들 4 및 5 는 현재의 픽셀의 픽셀 값 (Rec) 을 이웃하는 픽셀들의 픽셀 값들과 비교함으로써 어떻게 대각선 활동이 현재의 픽셀 (x, y) 에 대해 계산될 수 있는 지의 예들을 나타낸다.

[0171]
$$45\text{deg_act}(x,y) = R(2*\text{Rec}[x][y]-\text{Rec}[x+1][y+1]-\text{Rec}[x-1][y-1]) \quad (4)$$

[0172]
$$135\text{deg_act}(x,y) = R(2*\text{Rec}[x][y]-\text{Rec}[x-1][y+1]-\text{Rec}[x+1][y-1]) \quad (5)$$

[0173] 수식 4 로 나타낸 바와 같이, 대각선 활동은 현재의 픽셀 (x, y) 을 상부-우측 이웃 (x+1, y+1) 및 하부-좌측 이웃 (x-1, y-1) 과 비교함으로써 예를 들어, 45 도 방향에서 계산될 수 있다. 수식 5 로 나타낸 바와 같이, 대각선 활동은 또한 현재의 픽셀 (x, y) 을 좌측-상부 이웃 (x-1, y+1) 및 우측-하부 이웃 (x+1, y-1) 과 비교함으로써 135 도 방향에서 이루어질 수도 있다.

[0174] 상기 수식 2 내지 수식 5 는, 어떻게 수평 활동, 수직 활동, 및 대각선 활동이 픽셀 단위로 결정될 수 있는 지를 예시하며, 그러나, 일부 구현예들에서, 수평 활동, 수직 활동, 및 대각선 활동은 그룹 단위로 결정될 수도 있으며, 여기서, 픽셀들의 그룹은 2 x 2, 4 x 4, 또는 M x N 픽셀들의 블록이다. 이런 구현예에서, 수평 활동은, 예를 들어, 수식 2 와 유사한 방법으로, 현재의 그룹의 픽셀 값들을 좌측 그룹 및 우측 그룹의 픽셀 값들과 비교함으로써 결정될 수 있으며; 그리고, 수직 활동은 수식 3 과 유사한 방법으로, 현재의 그룹을 상부 그룹 및 하부 그룹과 비교함으로써 결정될 수 있다. 이와 유사하게, 45 도 대각선 활동은 수식 4 와 유사한 방법으로, 현재의 픽셀들의 그룹을 상부-우측 이웃하는 그룹 및 하부-좌측 이웃하는 그룹과 비교함으로써 결정될 수 있으며, 135 도 대각선 활동은 수식 5 와 유사한 방법으로, 현재의 픽셀들의 그룹을 상부-좌측 이웃하는 그룹 및 하부-우측 이웃하는 그룹과 비교함으로써 결정될 수 있다.

[0175] 일부 구현예들에서, 수평 활동, 수직 활동, 45 도 대각선 활동, 및 135 도 대각선 활동은 현재의 픽셀 또는 픽셀들의 그룹을 이웃하는 픽셀들 또는 픽셀들의 그룹들과 오직 하나의 방향으로 비교함으로써 결정될 수 있다.

예를 들어, 현재의 픽셀을 좌측 이웃 및 우측 이웃과 비교하는 것에 기초하여 수평 활동을 결정하는 대신, 수평 활동은 오직 좌측 이웃 또는 오직 우측 이웃에 기초하여 결정될 것이다. 게다가, 일부 구현예들에서, 수평 활동, 수직 활동, 45 도 대각선 활동, 및 135 도 대각선 활동은 단일 이웃하는 픽셀들 또는 픽셀들의 단일 그룹들 대신, 이웃하는 픽셀들의 영역들의 평균들 또는 가중 평균들을 이용하여 결정될 수도 있다.

[0176] 수식 2 내지 수식 5 로부터 기인하는 값들은 범위들의 유한한 개수, 예컨대 2, 4, 8, 또는 임의의 다른 유한한 개수로 분할될 수 있으며, 각각의 범위는 범위 식별을 할당받을 수 있다. 도 4a 를 다시 참조하면, 예를 들어, 범위 (1-1), 범위 (1-2), 범위 (2-1) 등은 모두 범위 식별들의 예들이다. 일 예로서, 수평 활동 값들은 4개의 범위들로 분할될 수 있으며, 범위들은 ID들 범위 (1-1), 범위 (1-2), 범위 (1-3), 및 범위 (1-4) 를 할당받을 것이다. 수평 임계값들 (즉, $\text{ThH}_1, \dots, \text{ThH}_{P-1}$) 은 어디서 범위들이 시작하고 끝나는 지를 결정할 수 있다. 아래 표 2 는 어떻게 수평 ID들이 P 범위들에 할당되는 지의 일반적인 경우를 나타낸다.

표 2

[0177] 활동 매트릭의 인덱스

Hor_act_B 의 조건	수평 ID
$\text{Hor_act_B} < \text{ThH}_1$	범위 2-1
$\text{ThH}_1 \leq \text{Hor_act_B} < \text{ThH}_2$	범위 2-2
...	...
$\text{ThH}_{P-1} \leq \text{Hor_act_B}$	범위 2-P

[0178] 표 2 의 예를 이용하여, 현재의 픽셀이 ThH_1 보다 크지만 ThH_2 미만인 수평 활동 값을 가지면, 현재의 픽셀은 매트릭 2 에 대한 범위 (2-2) 내에 존재한다. 현재의 픽셀들은 수평 범위들 및 수평 ID들에 대해 표 2 에 대

해 위에서 설명한 방법과 유사한 방법으로, 수직 ID들을 가진 수직 범위들, 45 도 대각선 ID들을 가진 45 도 대각선 범위들, 및 135 도 대각선 ID들을 가진 135 도 대각선 범위들에 할당될 수도 있다.

[0179] 수평 활동, 수직 활동, 45 도 대각선 활동, 및 135 도 대각선 활동 중 임의의 활동이 본 개시물에서 설명하는 멀티-메트릭 필터 필터링 기법들에 따라서 메트릭으로서 사용될 수 있다. 예를 들어, 도 4a 로 되돌아가 다시 참조하면, 메트릭 1 은 수직 활동의 측정치일 것이며, 메트릭 2 는 수평 활동의 측정치일 것이다. 이러한 예에서, 도 4a 의 필터 유닛 (349) 또는 도 5 의 필터 (559) 과 같은 필터 유닛은 픽셀 또는 픽셀의 그룹의 수평 활동 및 픽셀 또는 픽셀들의 그룹의 수직 활동에 기초하여, 픽셀 또는 픽셀들의 그룹에 대한 필터를 결정할 수 있다. 예를 들어, 현재의 픽셀이 범위 (2-3) 내에 들어가는 수평 활동 메트릭 및 수직 범위 (1-3) 내에 들어가는 활동 메트릭을 가지면, 필터 유닛은 그 픽셀을 필터 4 를 이용하여 필터링한다. 유사한 방법으로, 45 도 대각선 활동과 135 도 대각선 활동의 조합, 45 도 대각선 활동과 수평 활동의 조합, 45 도 대각선 활동과 수직 활동의 조합, 135 도 대각선 활동과 수평 활동의 조합, 또는 135 도 대각선 활동과 수직 활동의 조합이 또한 픽셀 또는 픽셀들의 그룹에 대한 필터를 선택하는 필터 유닛에 의해 사용될 수도 있다. 일부 구현예들에서, 수평 활동, 수직 활동, 45 도 대각선 활동, 및 135 도 대각선 활동 중 3개 또는 모든 4개가 픽셀 또는 픽셀들의 그룹의 필터를 선택하는 필터 유닛에 의해 사용될 수도 있다.

[0180] 위에서 설명한 구현예들에서, 수평 활동, 수직 활동, 45 도 대각선 활동, 및 135 도 대각선 활동은 모두 예를 들어, 도 4a 에서의 메트릭 1 및/또는 메트릭 2 과 같이, 메트릭들로서 사용될 수 있다. 일부 구현예들에서, 그러나, 수평 활동, 수직 활동, 45 도 대각선 활동, 및 135 도 대각선 활동 자체들은 메트릭들이 아닐 것이며, 그러나 대신, 전체 방향 메트릭을 결정하는 중간 결정들로서 사용될 수 있다. 방향 메트릭은 일반적으로 어느 방향 (예컨대, 무방향, 수평, 수직, 45 도 대각선, 또는 135 도 대각선) 에서 픽셀들이 가장 변하고 있는지를 기술한다.

[0181] 일 예에서, 수직 2 및 수직 3 에서 설명하는 바와 같이 오직 수평 활동 및 수직 활동을 이용하여, 픽셀에 대한 방향은 다음 조건들에 기초하여 결정될 것이다:

[0182] $\text{Hor_activity} > k1 * \text{Ver_activity}$ 이면, 방향 1 = 수평

[0183] $\text{Ver_activity} > k2 * \text{Hor_activity}$ 이면, 방향 2 = 수직

[0184] 그렇지 않으면, 방향 0 = 무방향.

[0185] 상수들 $k1$ 및 $k2$ 는, 수평 활동이 수직 활동보다 실질적으로 더 크거나 또는 수직 활동이 수평 활동보다 실질적으로 더 크면, 그 방향이 방향 1 또는 방향 2 인 것으로 오직 간주되도록, 선택될 수 있다. 수평 활동 및 수직 활동이 동일하거나 또는 거의 동일하면, 그 방향은 방향 0 이다. 방향 1 은 일반적으로 픽셀 값들이 수직 방향으로보다 수평 방향으로 더 많이 변하고 있다는 것을 나타내며, 방향 2 는 픽셀 값들이 수평 방향으로보다 수직 방향으로 더 많이 변하고 있다는 것을 나타낸다. 방향 0 은 수평 방향으로의 픽셀 값들의 변화가 수직 방향으로의 픽셀 값들의 변화와 거의 동일하다는 것을 나타낸다.

[0186] 결정된 방향 메트릭 (예컨대 방향 0, 방향 1, 방향 2) 은 본 개시물에서 설명하는 멀티-메트릭 필터링 기법들에서 메트릭으로서 사용될 수 있다. 도 4a 의 예를 다시 이용하여, 메트릭 1 은 합산-수정된 라플라시안 값과 같은 분산 메트릭일 것이며, 반면 메트릭 2 는 위에서 설명한 바와 같은 방향 결정일 것이다. 도 4a 를 참조하여 설명한 바와 같이, 비록 방향 1, 방향 2, 및 방향 0 이 값들의 스펙트럼 대신 유한한 결정들을 나타내더라도, 방향 1, 방향 2, 및 방향 0 각각은 메트릭 2 의 범위와 연관될 수 있다.

[0187] 위에서 설명한 바와 같이 오직 수평 활동 및 수직 활동을 이용하는 것에 더해서, 본 개시물의 기법들은 또한 다음 조건들에 기초하여, 방향들을 결정하기 위해, 수식들 4 및 5 에서 설명한 바와 같은, 45 도 대각선 활동 및 135 도 대각선 활동을 이용하는 것을 포함한다:

[0188] $45\text{deg_activity} > k1 * 135\text{deg_activity}$ 이면, 방향 = 1

[0189] $135\text{deg_activity} > k2 * 45\text{deg_activity}$ 이면, 방향 = 2

[0190] 그렇지 않으면, 방향 = 0.

[0191] 45 도 대각선 활동 및 135 도 대각선 활동에 기초한 방향 결정들은 위에서 설명한 바와 같이, 합산-수정된 라플라시안 값과 같은 또다른 메트릭을 가진 메트릭으로서 사용될 수 있다.

[0192] 게다가, 방향 메트릭은 또한 다음 조건들에 기초하여 결정될 수도 있다:

- [0193] $45deg_activity > k1*135deg_activity, k2* Hor_activity, \text{ 및 } k3* Ver_activity$ 이면, 방향 = 1
- [0194] $135deg_activity > k4*45deg_activity, k5* Hor_activity, \text{ 및 } k6* Ver_activity$ 이면, 방향 = 2
- [0195] $Hor_activity > k7 * Ver_activity, k8*135deg_activity, \text{ 및 } k9*45deg_activity$ 이면, 방향 = 3
- [0196] $Ver_activity > k10 * Hor_activity, k11*135deg_activity, \text{ 및 } k12*45deg_activity$ 이면, 방향 = 4
- [0197] 그렇지 않으면, 방향 = 0.
- [0198] 위에서 설명한 바와 같이, k1 내지 k12 는 어떤 방향이 선택되게 하기 위해서, 수평 활동, 수직 활동, 45 도 활동, 및 135 도 활동 요구들 중 하나보다 얼마 만큼 많이 다른 것들과 비교되는 것이 필요한 지를 결정하기 위해 선택되는 상수들이다. 수평 활동, 수직 활동, 45 도 대각선 활동, 및 135 도 대각선 활동에 기초한 방향 결정들은 위에서 설명한 바와 같이, 합산-수정된 라플라시안 값과 같은 또다른 메트릭을 가진 메트릭으로서 사용될 수 있다.
- [0199] 본 개시물의 기법들과 함께 사용될 수 있는 또다른 메트릭은 에지 메트릭 (edge metric) 을 포함한다. 에지 메트릭 (edge metric) 은 일반적으로 픽셀들의 블록에서 에지의 존재를 나타낼지도 모르는 활동을 정량화한다. 그 픽셀들의 블록이 이미지 내 오브젝트의 경계를 포함하면, 에지가 예를 들어, 픽셀들의 블록에 나타날 수도 있다. 에지 검출의 일 예는 현재의 픽셀의 4개의 이웃하는 픽셀들 (예컨대, 좌측, 우측, 상부, 하부) 을 이용하는 것, 또는 현재의 픽셀의 8개의 이웃하는 픽셀들 (좌측, 우측, 상부, 하부, 상부 우측, 상부 좌측, 하부 우측, 하부 좌측) 을 이용하는 것을 포함한다. 게다가, 에지 유형 검출은 상부 및 하부, 좌측 및 우측, 상부 좌측 및 하부 우측, 또는 상부 우측 및 좌측 하부와 같은, 2개의 이웃하는 픽셀들을 이용하는 것을 포함할 수도 있다.
- [0200] 아래의 의사 코드는 강도와 같은, 현재의 픽셀의 픽셀 값 (Rec) 을 그들 이웃하는 픽셀들 (즉, 4/8 픽셀들) 의 픽셀 값들과 비교함으로써 어떻게 에지 정보가 현재의 픽셀 (x, y) 에 대해 계산될 수 있는지의 예들을 나타낸다.
- [0201] EdgeType 변수는 0 으로 시작된다. 스테이트먼트가 참일 때마다, EdgeType 변수는 (EdgeType ++ 에 의해 의사 코드로 나타낸 바와 같이) 1 만큼 증분되거나 또는 (EdgeType -- 에 의해 의사 코드로 나타낸 바와 같이) 1 만큼 감분된다. Rec[x][y] 는 픽셀 강도와 같은, (x, y) 에 위치된 픽셀의 픽셀 값을 지칭한다. "if" 스테이트먼트들 (statements) 의 제 1 그룹은 현재의 픽셀을 상부, 하부, 좌측, 및 우측 이웃들과 비교하기 위한 것이다. "if" 스테이트먼트들의 제 2 그룹은 현재의 픽셀을 상부-좌측, 상부-우측, 하부-좌측, 및 하부-우측 이웃들과 비교하기 위한 것이다. 본 개시물의 기법들은 그룹 또는 양자의 그룹들을 이용하여 구현될 수 있다.
- ```

EdgeType = 0;
if (Rec[x][y] > Rec[x-1][y]) EdgeType ++;
if (Rec[x][y] < Rec[x-1][y]) EdgeType --;
if (Rec[x][y] > Rec[x+1][y]) EdgeType ++;
if (Rec[x][y] < Rec[x+1][y]) EdgeType --;
if (Rec[x][y] > Rec[x][y-1]) EdgeType ++;
if (Rec[x][y] < Rec[x][y-1]) EdgeType --;
if (Rec[x][y] > Rec[x][y+1]) EdgeType ++;
if (Rec[x][y] < Rec[x][y+1]) EdgeType --;

if (Rec[x][y] > Rec[x-1][y-1]) EdgeType ++;
if (Rec[x][y] < Rec[x-1][y-1]) EdgeType --;

```
- [0202]

```

if (Rec[x][y] > Rec[x+1][y-1]) EdgeType ++;
if (Rec[x][y] < Rec[x+1][y-1]) EdgeType --;
if (Rec[x][y] > Rec[x-1][y+1]) EdgeType ++;
if (Rec[x][y] < Rec[x-1][y+1]) EdgeType --;
if (Rec[x][y] > Rec[x+1][y+1]) EdgeType ++;
if (Rec[x][y] < Rec[x+1][y+1]) EdgeType --;

```

[0203]

[0204]

현재의 픽셀이 로컬 최대치이면, 그 픽셀의 픽셀 값은 모든 그의 이웃들보다 클 것이며, 4개의 이웃들을 이용하면 4의 에지 유형 또는 8개의 이웃들을 이용하면 8의 에지 유형을 가질 것이다. 현재의 픽셀이 로컬 최소치이면, 그 픽셀의 픽셀 값은 모든 그의 이웃들 미만일 것이며, 4개의 이웃들을 이용하면 -4의 에지 유형 또는 8개의 이웃들을 이용하면 -8의 에지 유형을 가질 것이다. 따라서, -4과 4사이 또는 -8과 8사이에서 에지 유형을 결정하기 위한, 위에서 설명한 예시적인 기법들을 이용하는 것이 필터를 결정할 때에 이용될 수 있다. 에지 유형에 대해 결정된 값들 (즉, -4 내지 4 값들, 또는 -8 내지 8의 값들)은 도 4a의 메트릭 1 또는 메트릭 2과 같은 메트릭의 범위들에 맵핑될 수 있다. 일부 구현예들에서, 에지 유형 결정의 절대 값들이, 예를 들어, -3 및 3의 에지 유형이 동일한 필터에 맵핑되도록, 범위들에 맵핑될 것이다.

[0205]

본 개시물에서 설명하는 여러 메트릭들의 계산들은 오직 예들로서 의도되며 완전한 것은 아니다. 예를 들어, 메트릭들은 본 개시물에서 설명하는 것보다 더 많은 이웃하는 픽셀들을 포함하는 픽셀들의 윈도우들 또는 라인들을 이용하여 결정될 수 있다.

[0206]

게다가, 일부 구현예들에서, 본 개시물에서 설명하는 메트릭들은 특정의 라인 또는 윈도우에서 픽셀들의 서브-샘플링을 이용하여 계산될 수도 있다. 예를 들어, 4 x 4 픽셀들의 블록에 대한 블록 활동 메트릭을 계산하기 위해, 활동 및 방향에 대한 메트릭들은 다음과 같이 계산될 수 있다:

[0207]

● 방향 메트릭

[0208]

$$\bullet \text{ Ver\_act}(i, j) = \text{abs} (X(i, j) \ll 1 - X(i, j-1) - X(i, j+1) )$$

[0209]

$$\bullet \text{ Hor\_act}(i, j) = \text{abs} (X(i, j) \ll 1 - X(i-1, j) - X(i+1, j) )$$

[0210]

$$\bullet H_B = \sum_{i=0,2} \sum_{j=0,2} \text{Hor\_act}(i, j)$$

[0211]

$$\bullet V_B = \sum_{i=0,2} \sum_{j=0,2} \text{Vert\_act}(i, j)$$

[0212]

$$\bullet \text{ 방향} = 0, 1 (H_B > k1 * V_B), 2 (V_B > k2 * H_B)$$

[0213]

● 활동 메트릭

[0214]

$$\bullet L_B = H_B + V_B$$

[0215]

$$\bullet 5 \text{ 가지 등급들 } (0, 1, 2, 3, 4)$$

[0216]

● 메트릭

[0217]

● 활동과 방향의 조합 (예컨대, 도 4b의 예에서 위에서 설명한 바와 같이 15 또는 16개의 조합들)

[0218]

$\text{Hor\_act}(i, j)$ 은 일반적으로 현재의 픽셀  $(i, j)$ 의 수평 활동을 지칭하며,  $\text{Vert\_act}(i, j)$ 은 일반적으로 현재의 픽셀  $(i, j)$ 의 수직 활동을 지칭한다.  $X(i, j)$ 는 일반적으로 픽셀  $(i, j)$ 의 픽셀 값을 지칭한다.

$H_B$ 는 4 x 4 블록의 수평 활동을 지칭하며, 이 예에서는, 픽셀들  $(0, 0)$ ,  $(0, 2)$ ,  $(2, 0)$ , 및  $(2, 2)$ 에 대한 수평 활동의 합에 기초하여 결정된다.  $V_B$ 는 4 x 4 블록의 수직 활동을 지칭하며, 이 예에서는, 픽셀들

$(0, 0)$ ,  $(0, 2)$ ,  $(2, 0)$ , 및  $(2, 2)$ 에 대한 수직 활동의 합에 기초하여 결정된다. " $\ll 1$ "는 2배 곱셈 연산을 나타낸다. 위에서 설명한 바와 같이,  $H_B$  및  $V_B$ 의 값들에 기초하여, 방향이 결정될 수 있다. 상기

예를 이용하여,  $H_B$ 의 값이  $V_B$ 의 값의  $k$  배보다 크면, 방향은 방향 1 (즉, 수평)인 것으로 결정될 수 있으며, 수직 활동보다 더 많은 수평 활동에 대응할 것이다.  $V_B$ 의 값이  $H_B$ 의 값의  $k$  배보다 크면, 방향은 방향 2

(즉, 수직)로 결정될 수 있으며, 수평 활동보다 더 많은 수직 활동에 대응할 것이다. 그렇지 않으면, 방향은 방향 0 (즉, 무방향)인 것으로 결정될 수 있으며, 수평 활동도 수직 활동도 지배적이지 않는다는 것을 의미한다. 다른 라벨들 및 비들이 또한 사용될 수 있기 때문에, 여러 방향들에 대한 라벨들 및 방향들을 결정하는데 사용되는 비들은 단지 일 예를 구성한다.

[0219] 4 x 4 블록에 대한 활동 ( $L_B$ )은 수평 활동과 수직 활동의 합계로서 결정될 수 있다.  $L_B$ 의 값은 위에서 설명한 바와 같이, 범위로 분류될 수 있다. 이 특정의 예는 5개의 범위들을 나타내지만, 더 많거나 또는 더 적은 범위들이 유사하게 사용될 수도 있다. 활동 및 방향의 조합에 기초하여, 4 x 4 픽셀들의 블록에 대한 필터가 선택될 수 있다. 위에서 설명한 바와 같이, 필터는 도 4a 및 도 4b를 참조하여 설명한 바와 같이, 필터들에의 활동 및 방향의 2차원 매핑에 기초하여 선택될 수도 있거나, 또는 활동 및 방향은 단일 메트릭으로 결합될 수도 있으며, 그 단일 메트릭은 필터를 선택하는데 사용될 수도 있다.

[0220] 도 6a는 4 x 4 픽셀들의 블록을 나타낸다. 위에서 설명한 서브-샘플링 기법들을 이용하여, 16개의 픽셀들 중 오직 4개가 사용된다. 4개의 픽셀들은 픽셀 (601)로서 라벨링된 픽셀 (0, 0), 픽셀 (602)로서 라벨링된 픽셀 (2, 0), 픽셀 (603)로서 라벨링된 픽셀 (0, 2), 및 픽셀 (604)로서 라벨링된 픽셀 (2, 2)이다. 예를 들어, 픽셀 (601) (즉,  $hor\_act(0, 0)$ )의 수평 활동은 좌측 이웃하는 픽셀 및 우측 이웃하는 픽셀에 기초하여 결정된다. 우측 이웃하는 픽셀은 픽셀 (605)로서 라벨링된다. 좌측 이웃하는 픽셀은 4 x 4 블록과는 상이한 블록에 위치되며, 도 6a 상에 나타나지 않는다. 예를 들어, 픽셀 (602) (즉,  $ver\_act(2, 0)$ )의 수직 활동은 상부 이웃하는 픽셀 및 하부 이웃하는 픽셀에 기초하여 결정된다. 하부 이웃하는 픽셀은 픽셀 (606)로서 라벨링되며, 상부 이웃하는 픽셀은 4 x 4 블록과는 상이한 블록에 위치되며, 도 6a에 도시되지 않는다.

[0221] 일반적으로 위에서 설명한 동일한 기법들을 이용하여, 블록 활동 메트릭은 또한 다음과 같은 상이한 픽셀들의 서브세트를 이용하여 계산될 수도 있다:

[0222] ● 방향 메트릭

[0223] ●  $Ver\_act(i, j) = abs ( X(i, j) < 1 - X(i, j-1) - X(i, j+1) )$

[0224] ●  $Hor\_act(i, j) = abs ( X(i, j) < 1 - X(i-1, j) - X(i+1, j) )$

[0225] ●  $H_B = \sum_{i=1,2} \sum_{j=1,2} H(i, j)$

[0226] ●  $V_B = \sum_{i=1,2} \sum_{j=1,2} V(i, j)$

[0227] ● 방향 = 0, 1 ( $H > k1 * V$ ), 2 ( $V > k2 * H$ )

[0228] ● 활동 메트릭

[0229] ●  $L_B = H_B + V_B$

[0230] ● 5개 등급들 (0, 1, 2, 3, 4)

[0231] ● 메트릭

[0232] ● 활동과 방향의 조합 (예컨대, 위에서 도 4b의 예에서 설명한 바와 같이 15 또는 16개의 조합들)

[0233]  $H_B$  및  $V_B$ 를 계산하기 위한 이 상이한 픽셀들의 서브세트는 도 6b 상에 픽셀들 (611, 612, 613, 및 614)로서 각각 나타낸, 픽셀들 (1, 1), (2, 1), (1, 2), 및 (2, 2)을 포함한다. 도 6b에 의해 볼 수 있는 바와 같이, 픽셀들 (611, 612, 613, 및 614)에 대해 상부 이웃하는 픽셀들, 하부 이웃하는 픽셀들, 우측 이웃하는 픽셀들, 및 좌측 이웃하는 픽셀들 모두는 4 x 4 블록 내에 위치된다. 도 6b의 예에서, 픽셀들 (611, 612, 613, 및 614)은 모두 블록 경계 상에 위치되는 것과는 반대로, 블록의 내부에 위치된다. 도 6a에서의 픽셀들 (601, 602, 603, 및 605) 및 도 6c에서의 픽셀들 (621, 624, 625, 및 628)은 블록 경계 상에 위치한 픽셀들의 예들이다. 다른 구현예들에서, 추가적인 상이한 픽셀의 서브세트들이 선정될 수도 있다. 예를 들어, 서브세트들은, 서브세트의 픽셀들에 대한 상부 및 하부 이웃하는 픽셀들이 4 x 4 블록 내에 있지만 일부 좌측 및 우측 이웃하는 픽셀들이 이웃하는 블록들에 있도록, 선택될 수도 있다. 서브세트들은, 또한 서브세트의 픽셀들에 대한 좌측 및 우측 이웃하는 픽셀들이 4 x 4 블록 내에 있지만 일부 상부 및 하부 이웃하는 픽셀

들이 이웃하는 블록들에 있도록, 선택될 수도 있다.

[0234] 일반적으로 위에서 설명한 동일한 기법들을 이용하여, 블록 활동 메트릭은 또한 다음과 같은 8개의 픽셀들의 서브세트를 이용하여 계산될 수도 있다:

[0235] ● 방향 메트릭

[0236] ●  $Ver\_act(i,j) = \text{abs} ( X(i,j) << 1 - X(i,j-1) - X(i,j+1) )$

[0237] ●  $Hor\_act(i,j) = \text{abs} ( X(i,j) << 1 - X(i-1,j) - X(i+1,j) )$

[0238] ●  $H_B = \sum_{i=0,1,2,3} \sum_{j=1,2} H(i,j)$

[0239] ●  $V_B = \sum_{i=0,1,2,3} \sum_{j=1,2} V(i,j)$

[0240] ● 방향 = 0, 1( $H > k1*V$ ), 2 ( $V > k2*H$ )

[0241] ● 활동 메트릭

[0242] ●  $L_B = H_B + V_B$

[0243] ● 5개의 등급들 (0, 1, 2, 3, 4)

[0244] ● 메트릭

[0245] ● 활동과 방향의 조합 (예컨대, 위에서 도 4b 의 예에서 설명한 바와 같은 15 또는 16 개의 조합들)

[0246]  $H_B$  및  $V_B$  를 계산하기 위한, 이 상이한, 8개의 픽셀들의 서브세트는 도 6c 상에 픽셀들 (621, 622, 623, 및 624, 625, 626, 627, 및 628) 로서 각각 나타낸, 픽셀들 (0, 1), (1, 1), (2, 1), (3, 1), (0, 2), (1, 2), (2, 2), 및 (3, 2) 을 포함한다. 도 6c 에 의해 볼 수 있는 바와 같이, 픽셀들 (621 및 625) 각각이 좌측 이웃하는 블록에서 좌측 이웃하는 픽셀들을 갖고 픽셀들 (624 및 628) 각각이 우측 이웃하는 블록에서 우측 이웃하는 픽셀들을 갖지만, 픽셀들 (621, 622, 623, 및 624, 625, 626, 627, 및 628) 에 대한 상부 이웃하는 픽셀들 및 하부 이웃하는 픽셀들 모두는 4 x 4 블록 내에 위치된다. 이 특성의 픽셀들의 선택은 상부 및/또는 하부 이웃하는 블록들의 픽셀 값들을 저장하는 라인 버퍼에 대한 필요성을 제거함으로써, 인코더 및/또는 디코더 복잡성을 감소시킬 수도 있다. 좌우, 상하 래스터 스캔 순서로 인해, 상부 및 하부 이웃하는 블록들의 픽셀 값들에 대한 라인 버퍼들은 종종 전체 상부 또는 하부 라인에 대한 픽셀 값들을 저장하는 것을 필요로 하며, 픽셀 값들은 1080P 비디오의 경우에, 예를 들어, 1920 픽셀들일 것이다. 그러나, 좌측 및 우측 이웃하는 블록들에 대한 라인 버퍼들은 종종 하나의 LCU 또는 LCUs 의 쌍에 대한 픽셀 값들을 오직 저장하는 것을 필요로 하며, 그 픽셀 값들은 겨우 예를 들어, 64 또는 128 픽셀들일 것이다. 따라서, 상부 및 하부 이웃하는 블록들의 픽셀 값들에 대한 라인 버퍼들은 좌측 및 우측 이웃하는 블록들의 픽셀 값들에 대해 사용되는 라인 버퍼들보다 현저하게 더 큰 것을 필요로 할 수도 있다. 도 6c 에 나타낸 픽셀들의 선택은 상부 및 하부 이웃하는 블록의 픽셀 값들에 대한 라인 버퍼들의 사용을 회피하도록 함으로써, 코딩 복잡성을 감소시킬 수도 있다.

[0247] 도 6a 내지 도 6c 의 예들은 단지 본 개시물의 기법들이 소개된다. 이들 기법들은 꼭 4 x 4 이외의 블록들로 확장될 수 있으며 그리고 상이한 픽셀들의 서브세트들이 선택될 수도 있는 것으로 생각된다.

[0248] 블록 활동 메트릭을 계산할 때, 원래 픽셀들 대신, 양자화된 픽셀들 (즉,  $X(i,j) >> N$ ) 이 가산 연산들과 같은 연산들의 복잡성을 감소시키기 위해 사용될 수 있다. 게다가, 계산들은 라플라시안 기반 대신, 절대 차이 기반일 수 있다. 예를 들어,  $Hor\_act(i,j)$  또는  $Ver\_act(i,j)$  을 계산할 때, 절대 차이들이 다음과 같이 라플라시안 값들 대신 사용될 수 있다:

[0249] 방향 메트릭

[0250] ■  $Ver\_act(i,j) = \text{abs} ( X(i,j) - X(i,j-1) )$

[0251] ■  $Hor\_act(i,j) = \text{abs} ( X(i,j) - X(i-1,j) )$

[0252] ■  $H_B = \sum_{i=0,1,2} \sum_{j=0,1,2} H(i,j)$

- [0253]                   ■  $V_B = \sum_{i=0,1,2} \sum_{j=0,1,2} V(i, j)$
- [0254]                   ■ 방향 = 0, 1(H>2V), 2 (V>2H)
- [0255]                   ■ 활동 메트릭
- [0256]                   ■  $L_B = H_B + V_B$
- [0257]                   ■ 5개의 등급들 (0, 1, 2, 3, 4)
- [0258]                   ■ 메트릭
- [0259]                   ■ 활동 + 방향 (예컨대, 위에서 도 4b 의 예에서 설명한 바와 같이                   15 또는 16 개의 조합들)
- [0260]                   본 개시물은 특정의 메트릭들의 제한되는 그룹을 참조하여 서브-샘플링 기법들을 설명하였다.                   그러나, 이들 서브-샘플링 기법들은 일반적으로 본 개시물에서 설명한 다른 메트릭들과 같은, 필터를 결정하는 목적을 위해 사용될 수도 있는 다른 메트릭들에 적용가능한 것으로 생각된다.                   게다가, 본 개시물의 서브-샘플링 기법들이 4 x 4 픽셀들의 블록들을 참조하여 설명되었지만, 이 기법들은 또한 다른 사이즈들의 블록들에 적용가능할 수도 있다.
- [0261]                   도 7 은 본 개시물과 부합하는 비디오 코딩 기법을 예시하는 흐름도이다.                   도 7 에서 설명되는 기법들은 비디오 인코더 (350) 의 필터 유닛 (349) 또는 비디오 디코더 (560) 의 필터 유닛 (559) 과 같은, 비디오 인코더 또는 비디오 디코더의 필터 유닛에 의해 수행될 수 있다.                   필터 유닛은 픽셀들의 블록 내 픽셀들의 그룹에 대한 제 1 메트릭을 결정한다 (710).                   제 1 메트릭은 예를 들어, 합산-수정된 라플라시안 값과 같은 활동 메트릭일 수도 있거나, 또는 제 1 메트릭은 방향 메트릭일 수도 있다.                   제 1 메트릭은 예를 들어, 그 블록 내 다른 픽셀들에 대한, 블록에서의 픽셀들의 세트에 기초하거나, 또는 그 블록 내 픽셀들의 서브세트의 비교에 기초하여 결정될 수도 있다.                   필터 유닛은 그 블록에 대한 제 2 메트릭을 추가로 결정한다 (720).                   제 2 메트릭은 예를 들어, 수평 활동의 측정치를 수직 활동의 측정치와 비교하는 것에 기초하여 결정되는 방향 메트릭일 수도 있다.                   제 1 메트릭 및 제 2 메트릭에 기초하여, 필터 유닛은 필터를 결정한다 (730).                   필터 유닛은 그 필터를 블록에 적용하여 필터링된 이미지를 생성한다 (740).                   위에서 설명한 바와 같이, 일부 구현 예들에서, 블록은 제 1 메트릭 또는 제 2 메트릭을 결정하기 위해 사용되는, 2 x 2, 4 x 4, 또는 M x N 픽셀들의 블록일 수도 있다.                   일부 구현예들에서, 제 1 메트릭은 수평 활동 메트릭일 수도 있지만 제 2 메트릭은 수직 활동 메트릭이거나, 또는 제 1 메트릭은 에지 메트릭일 수도 있지만 제 2 메트릭은 방향 메트릭이다.
- [0262]                   도 8a 는 본 개시물과 부합하는 비디오 코딩 기법들을 예시하는 흐름도이다.                   도 8a 에서 설명되는 기법들은 비디오 디코더 (560) 의 필터 유닛 (559) 과 같은 비디오 디코더의 필터 유닛에 의해 수행될 수 있다.                   필터 유닛 (559) 은 제 1 범위 조합을 제 1 필터에 맵핑한다 (810A).                   제 1 범위 조합은 제 1 메트릭에 대한 값들의 제 1 범위와 제 2 메트릭에 대한 값들의 제 1 범위의 조합이다.                   제 1 메트릭은 예를 들어, 합산-수정된 라플라시안 값일 수도 있으며, 제 2 메트릭은 방향 메트릭일 수도 있지만, 다른 메트릭들이 또한 사용될 수도 있다.                   필터 유닛 (559) 은 제 2 범위 조합을 제 2 필터에 맵핑한다 (820A).                   제 2 범위 조합은 제 1 메트릭에 대한 값들의 제 2 범위와 제 2 메트릭에 대한 값들의 제 2 범위의 조합이다.                   필터 유닛 (559) 은 그 후, 수신된 코드워드에 기초하여 현재의 범위 조합을 필터에 맵핑한다.                   현재의 범위 조합은 제 1 메트릭의 값들의 제 1 범위 및 제 2 메트릭에 대한 값들의 제 2 범위를 포함한다.                   코드워드가 제 1 코드워드이면 (830A, 예), 필터 유닛 (559) 은 현재의 범위 조합을 제 1 필터에 맵핑한다 (840A).                   제 1 코드워드는 현재의 범위 조합이 제 1 범위 조합과 동일한 필터에 맵핑된다는 것을 나타낸다.                   그 코드워드가 제 2 코드워드이면 (850A, 예), 필터 유닛 (559) 은 현재의 범위 조합을 제 2 필터에 맵핑한다 (860A).                   제 2 코드워드는 현재의 범위 조합이 제 2 조합과 동일한 필터에 맵핑된다는 것을 나타낸다.                   그 코드워드가 제 1 코드워드도 제 2 코드워드도 아니면 (850A, 아니오), 필터 유닛 (559) 은 현재의 범위 조합을 제 3 필터에 맵핑한다 (870A).                   제 3 코드워드를 수신하는 것에 응답하여, 제 3 코드워드는 그 제 3 필터를 식별한다.                   도 8a 의 예에서, 제 1 코드워드 및 제 2 코드워드는 각각 제 3 코드워드보다 더 적은 비트들을 포함할 수도 있다.
- [0263]                   도 8b 는 본 개시물과 부합하는 비디오 코딩 기법들을 예시하는 흐름도이다.                   도 8b 에서 설명하는 기법들은 비디오 디코더 (560) 의 필터 유닛 (559) 과 같은 비디오 디코더의 필터 유닛에 의해 수행될 수 있다.                   필터 유닛 (559) 은 필터들에의 범위 조합들의 맵핑을 생성한다 (810B).                   각각의 범위 조합은 예를 들어, 제 1 메트릭에 대한 범위 및 제 2 메트릭에 대한 범위를 포함할 수 있다.                   현재의 범위 조합이 이전 범위 조합과 동

일한 필터에 맵핑된다는 것을 시그널링하는 제 1 코드워드를 수신하는 것에 응답하여 (820B, 예), 필터 유닛 (559)은 현재의 범위 조합을 이전 범위 조합과 동일한 동일한 필터에 맵핑한다 (830B). 현재의 범위 조합이 이전 범위 조합과는 상이한 필터에 맵핑된다는 것을 시그널링하는 제 2 코드워드를 수신하는 것에 응답하여 (820B, 아니오), 필터 유닛 (559)은 현재의 범위 조합을 새로운 필터에 맵핑한다 (840B). 위에서 설명한 바와 같이, 현재의 범위 조합은 기지의 송신 순서에 기초하여 결정될 수 있다. 일부 예들에서, 새로운 필터는 제 2 코드워드에 기초하여 식별될 수 있으며, 반면, 다른 예들에서, 새로운 필터는 필터 계수들이 시그널링되는 순서에 기초하여 결정될 것이다.

[0264]

도 9a는 본 개시물과 부합하는 비디오 코딩 기법들을 예시하는 흐름도이다. 도 9a에서 설명하는 기법들은 비디오 인코더 (350)의 필터 유닛 (349)과 같은 비디오 인코더의 필터 유닛에 의해 수행될 수 있다. 필터 유닛 (349)은 필터들에의 범위 조합들의 맵핑을 결정한다 (910A). 각각의 범위 조합은 제 1 메트릭에 대한 값들의 범위 및 제 2 메트릭에 대한 값들의 범위를 포함한다. 현재의 범위 조합에 대해, 현재의 범위 조합이 제 1 메트릭에 대한 값들의 동일한 범위를 포함하는 이전 범위 조합과 동일한 필터에 맵핑되면 (920A, 예), 필터 유닛 (349)은 제 1 코드워드를 생성한다 (930A). 현재의 범위 조합이 제 2 메트릭에 대한 값들의 동일한 범위를 포함하는 이전 범위 조합과 동일한 필터에 맵핑되면 (940A, 예), 필터 유닛 (349)은 제 2 코드워드를 생성한다 (950A). 현재의 범위 조합이 제 1 메트릭에 대한 값들의 동일한 범위를 포함하는 이전 범위 조합, 또는 제 2 메트릭에 대한 값들의 동일한 범위를 포함하는 이전 범위 조합 중 어느 하나에 맵핑되지 않으면 (950A, 아니오), 필터 유닛 (349)은 제 3 코드워드를 생성한다 (960A). 제 3 코드워드는 현재의 범위 조합에 맵핑되는 필터를 식별할 수 있다.

[0265]

도 9b는 본 개시물과 부합하는 비디오 코딩 기법들을 예시하는 흐름도이다. 도 9b에서 설명하는 기법들은 비디오 인코더 (350)의 필터 유닛 (349)과 같은 비디오 인코더의 필터 유닛에 의해 수행될 수 있다. 필터 유닛 (349)은 필터들에의 범위 조합들의 맵핑을 결정한다 (910B). 각각의 범위 조합은 예를 들어, 제 1 메트릭에 대한 범위 및 제 2 메트릭에 대한 범위를 포함할 수 있다. 코딩중인 현재의 범위 조합이 이전에 코딩된 범위 조합과 동일한 필터를 가질 때 (920B, 예), 필터 유닛 (349)은 현재의 범위 조합이 이전 범위 조합과 동일한 동일한 필터에 맵핑된다는 것을 시그널링하기 위해 제 1 코드워드를 생성할 수 있다 (930B). 코딩중인 현재의 범위 조합이 이전에 코딩된 범위 조합과 동일한 필터를 갖지 않을 때 (920B, 아니오), 필터 유닛 (349)은 제 2 코드워드를 생성할 수 있다 (940B). 제 2 코드워드는 현재의 범위 조합에 맵핑되는 필터를 식별할 수 있다. 위에서 설명한 바와 같이, 현재의 범위 조합은 기지의 송신 순서에 기초하여 결정될 수 있다. 도 9b의 예에서, 제 1 코드워드는 제 2 코드워드보다 적은 비트들을 포함할 수도 있다.

[0266]

도 8a와 도 8b, 그리고, 도 9a와 도 9b의 예들에서, 용어를 "제 1 코드워드", "제 2 코드워드", 및 "제 3 코드워드"는 상이한 코드워드들 사이에 구별하기 위해 사용되며, 코드워드들의 순차적인 순서를 암시하려고 의도된 것이 아니다.

[0267]

도 10은 본 개시물과 부합하는 비디오 코딩 기법들을 예시하는 흐름도이다. 도 10에서 설명하는 기법들은 비디오 인코더 (350)의 필터 유닛 (349)과 같은 비디오 인코더의 필터 유닛, 또는 필터 유닛 (559)과 같은 비디오 디코더의 필터 유닛에 의해 수행될 수 있다. 필터 유닛은 필터들에의 범위 조합들의 맵핑을 결정한다 (1010). 범위 조합들은 제 1 메트릭에 대한 범위 및 제 2 메트릭에 대한 범위를 포함한다. 필터 유닛은 각각의 범위 조합에 대한 고유한 범위 조합 식별 (ID)을 결정한다 (1020). 고유한 범위 조합 ID들은 순차적인 값들에 대응한다. 필터 유닛은 범위 조합들의 제 1 그룹에서 적어도 하나의 범위 조합의 범위 조합 ID의 순차적인 값에 기초하여 제 1 고유한 그룹 ID를 범위 조합들의 제 1 그룹에 할당한다 (1030). 범위 조합들의 그룹들은 동일한 필터에 맵핑되는 범위 조합들을 포함하며, 고유한 그룹 ID들이 순차적인 값들의 세트에 대응한다. 필터 유닛은 제 1 고유한 필터 ID의 순차적인 값에 기초하여, 동일한 필터에 대응하는 필터 계수들의 제 1 세트를 코딩한다 (1040). 비디오 인코더의 경우, 필터 계수들의 제 1 세트를 코딩하는 것은 예를 들어, 필터 계수들을 인코딩된 비트스트림으로 차분 코딩 기법들을 이용하여 시그널링하는 것을 포함할 수 있다. 비디오 디코더의 경우, 필터 계수들의 제 1 세트를 코딩하는 것은 인코딩된 비트스트림으로 수신된 정보에 기초하여, 필터 계수들을 재구성하는 것을 포함할 수 있다.

[0268]

도 11은 본 개시물과 부합하는 비디오 코딩 기법들을 예시하는 흐름도이다. 도 11에서 설명하는 기법들은 비디오 인코더 (350)의 필터 유닛 (349)과 같은 비디오 인코더의 필터 유닛, 또는 필터 유닛 (559)과 같은 비디오 디코더의 필터 유닛에 의해 수행될 수 있다. 필터 유닛은 필터들에의 범위 조합들의 맵핑을 결정한다 (1110). 범위 조합들은 제 1 메트릭에 대한 범위 및 제 2 메트릭에 대한 범위를 포함할 수 있다. 각각의 범위 조합은 고유한 범위 조합 식별 (ID)을 가질 수 있으며, 각각의 고유한 범위 조합 ID는 범위 조합에

대한 순차적인 값에 대응할 수 있다. 필터 유닛은 고유한 그룹 ID 을 범위 조합들의 각각의 그룹에 할당할 수 있다 (1120). 필터 유닛은 예를 들어, 범위 조합들의 순차적인 값들에 기초하여, 고유한 그룹 ID들을 할당할 수 있다. 범위 조합들의 그룹은 공통 필터에 맵핑되는 범위 조합들을 포함할 수 있으며, 고유한 그룹 ID들은 순차적인 값들의 세트에 대응할 수 있다. 필터 유닛은 고유한 그룹 ID들에 기초하여 필터들에 대한 필터 계수들의 세트들을 코딩할 수 있다 (1130).

[0269]

도 11 의 예에서, 필터 유닛은 예를 들어, 고유한 그룹 ID들의 최저 순차적인 값에 대응하는 고유한 그룹 ID 를, 범위 조합 ID들의 최저 순차적인 값에 대응하는 범위 조합 ID 를 가진 범위 조합을 포함하는 범위 조합들의 그룹에 할당함으로써, 고유한 그룹 ID들을 할당할 수 있다. 또다른 예에서, 필터 유닛은 고유한 그룹 ID들의 최고 순차적인 값에 대응하는 고유한 그룹 ID 을, 범위 조합 ID들의 최고 순차적인 값에 대응하는 범위 조합 ID 를 가진 범위 조합을 포함하는 범위 조합들의 그룹에 할당할 수 있다.

[0270]

필터 유닛이 비디오 디코더의 부분인 경우, 필터 유닛은 코딩된 비트스트림으로 수신된 정보에 기초하여 필터 계수들의 세트들을 생성함으로써 필터 계수들의 세트들을 코딩할 수 있다. 필터 유닛은 예를 들어, 필터 계수들의 세트들을 차분 코딩 기법들을 이용하여 생성할 수 있다. 필터 유닛이 비디오 인코더의 부분인 경우, 필터 유닛은 필터 계수들의 세트들을 코딩된 비트스트림으로 고유한 그룹 ID들의 순차적인 값들에 기초하여 선택된 순서로 시그널링함으로써, 필터 계수들의 세트들을 코딩할 수 있다. 필터 유닛은 예를 들어, 필터 계수들의 세트들을 차분 코딩 기법들을 이용하여 시그널링할 수 있다.

[0271]

전술한 개시물은 세부 사항들을 전달하기 위해 다소 단순화되었다. 예를 들어, 본 개시물은 일반적으로 필터들의 세트들이 프레임 단위 또는 슬라이스 단위로 시그널링되는 것으로 설명하지만, 필터들의 세트들은 또한 시퀀스 단위, 화상 그룹 단위, 슬라이스들 그룹 단위, CU 단위, LCU 단위, 또는 다른 그러한 단위로 시그널링될 수도 있다. 일반적으로, 필터들은 하나 이상의 CUs 의 임의의 그룹핑을 위해 시그널링될 수도 있다. 게다가, 구현예에서, CU 에 대한 입력에 대해 매우 많은 필터들, 필터에 대해 매우 많은 계수들, 그리고, 필터들의 각각이 상이한 분산의 범위에 대해 정의된 상태에서 매우 많은 상이한 분산의 레벨들이 존재할 수도 있다. 예를 들어, 일부 경우, CU 의 각각의 입력에 대해 정의된 16개 또는 더 이상의 필터들, 및 각각의 필터에 대응하는 16개의 상이한 분산의 범위들이 존재할 수도 있다. 게다가, 본 개시물이 필터 정보를 송신하는 것을 설명할 때, 모든 필터 정보가 동일한 코딩 레벨에서 송신되는 것으로 가정되지 않아야 한다. 예를 들어, 일부 구현예들에서, 필터 기술 구분과 같은 일부 필터 정보는 프레임 단위 또는 슬라이스 단위로 시그널링될 수도 있지만, 다른 필터 계수들과 같은 필터 정보는 LCU 단위로 시그널링된다. 구분은, 시퀀스 레벨, GOP-레벨, 또는 다른 레벨들과 같은 코딩 계층의 다른 레벨들에서, 또한 이런 필터 정보의 일부 또는 모두를 운반하기 위해 정의될 수 있다.

[0272]

각각의 입력에 대한 필터들 각각은 많은 계수들을 포함할 수도 있다. 일 예에서, 필터들은 2-차원으로 확장하는 필터 서포트에 대해 정의된 81 개의 상이한 계수들을 가진 2차원 필터들을 포함한다. 그러나, 일부 경우, 각각의 필터에 대해 시그널링되는 필터 계수들의 개수는 81 개 보다 적을 수도 있다. 계수 대칭은 예를 들어, 하나의 치수들 또는 사분면에서 필터 계수들이 다른 치수들 또는 사분면들에서의 계수들에 상대적인, 반전된 (inverted) 또는 대칭 값들에 대응할 수 있도록, 도입될 수도 있다. 계수 대칭은 81 개의 상이한 계수들이 더 적은 계수들로 표현가능하게 할 수도 있으며, 이 경우, 인코더 및 디코더는 계수들의 반전된 (inverted) 또는 미러된 (mirrored) 값들이 다른 계수들을 정의한다고 가정할 수도 있다. 예를 들어, 계수들 (5, -2, 10, 10, -2, 5) 은 계수들 (5, -2, 10) 의 서브세트로서 인코딩되어 시그널링될 수도 있다. 이 경우, 디코더는 이들 3개의 계수들이 계수들 (5, -2, 10, 10, -2, 5) 의 더 큰 대칭 세트를 정의한다는 것을 알 수도 있다.

[0273]

본 개시물의 기법들은 무선 핸드셋, 및 집적 회로 (IC) 또는 ICs의 세트 (즉, 칩 세트) 를 포함한, 매우 다양한 디바이스들 또는 장치들로 구현될 수도 있다. 임의의 구성요소들, 모듈들 또는 유닛들이 기능적 양태들을 강조하기 위해 제공되어 설명되었으며, 상이한 하드웨어 유닛들에 의한 실현을 반드시 필요로 하지는 않는다.

[0274]

따라서, 본원에서 설명되는 기법들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 하드웨어로 구현되는 경우, 모듈들, 유닛들 또는 구성요소들로서 설명되는 임의의 특성들 (features) 은 통합 로직 디바이스 내에 함께, 또는 별개의 공용가능한 로직 디바이스들로서 별개로 구현될 수도 있다. 소프트웨어로 구현되는 경우, 이 기법들은 프로세서에서 실행될 때, 위에서 설명한 방법들 중 하나 이상을 수행하는 명령들을 포함하는 컴퓨터-판독가능 매체에 의해 적어도 부분적으로 실현될 수도 있다. 컴퓨터-판독가능 매체는 컴퓨터 판독가능 저장 매체를 포함할 수도 있으며, 패키징 재료들을 포함할 수도 있는 컴퓨터 프로

그럼 제품의 부분을 형성할 수도 있다. 컴퓨터-판독가능 저장 매체는 동기식 동적 랜덤 액세스 메모리 (SDRAM) 와 같은 랜덤 액세스 메모리 (RAM), 판독 전용 메모리 (ROM), 비-휘발성 랜덤 액세스 메모리 (NVRAM), 전기적으로 소거가능한 프로그래밍가능 판독 전용 메모리 (EEPROM), 플래시 메모리, 자기 또는 광학적 데이터 저장 매체들 등을 포함할 수도 있다. 추가적으로, 또는 대안적으로, 이 기법들은 코드를 명령들 또는 데이터 구조들의 형태로 반송하거나 또는 통신하고, 그리고 컴퓨터에 의해 액세스되거나, 판독되거나 및/또는 실행될 수 있는 컴퓨터-판독가능 통신 매체에 의해 적어도 부분적으로 실현될 수도 있다.

[0275]

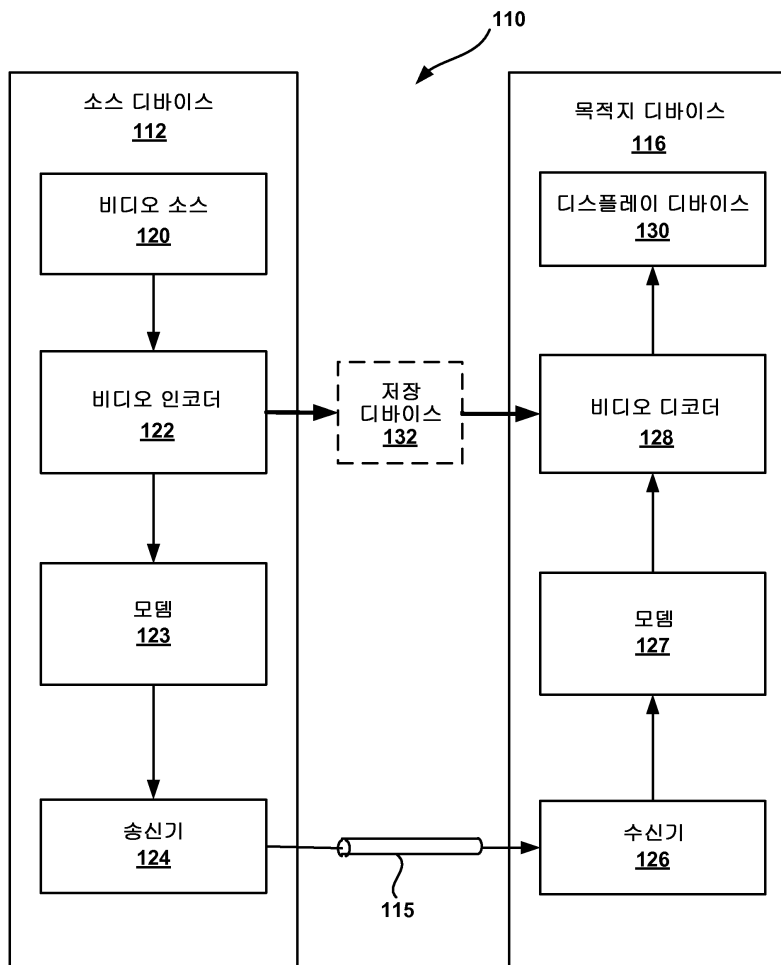
이 코드는 하나 이상의 디지털 신호 프로세서들 (DSPs), 범용 마이크로프로세서들, 주문형 집적회로들 (ASICs), 필드 프로그래밍가능 로직 어레이들 (FPGAs), 또는 다른 등가의 통합 또는 이산 로직 회로와 같은 하나 이상의 프로세서들에 의해 실행될 수도 있다. 따라서, 용어 “프로세서” 는, 본원에서 사용될 때 전술한 구조 중 임의의 구조 또는 본원에서 설명하는 기법들의 구현에 적합한 임의의 다른 구조를 지칭할 수도 있다. 게다가, 일부 양태들에서, 본원에서 설명하는 기능은 인코딩 및 디코딩을 위해 구성되는 전용 소프트웨어 모듈들 또는 하드웨어 모듈들 내에 제공되거나, 또는 결합된 비디오 코덱에 통합될 수도 있다. 또한, 이 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들로 완전히 구현될 수 있다.

[0276]

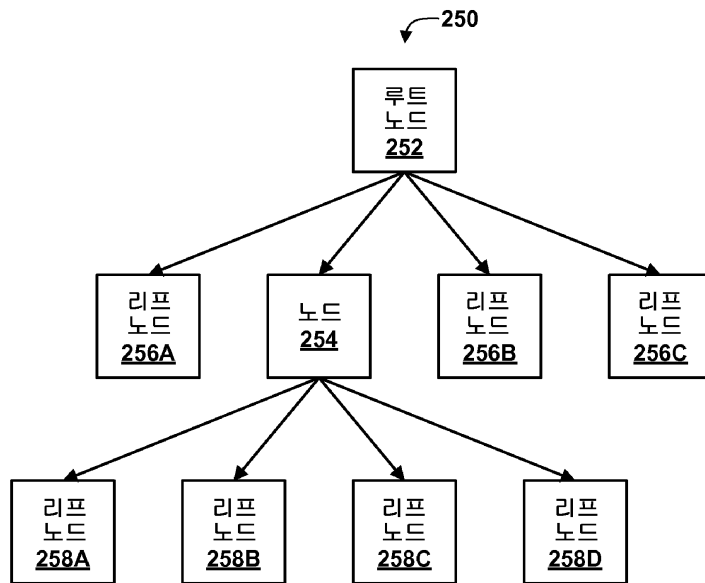
여러 본 개시물의 양태들이 설명되었다. 이들 및 다른 양태들은 다음 청구항들의 범위 내이다.

## 도면

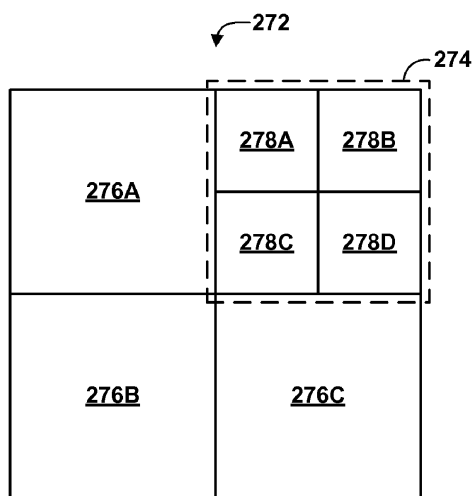
### 도면1



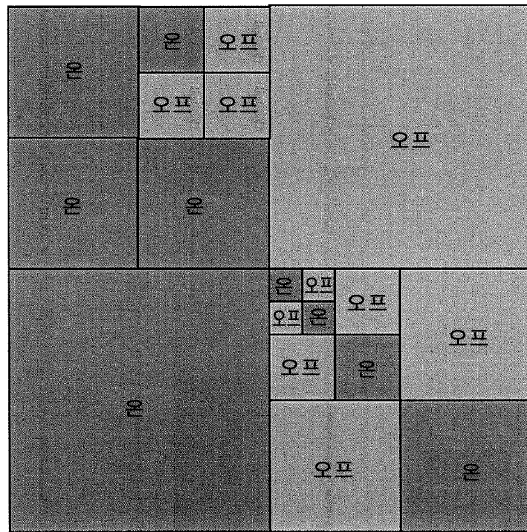
도면2a



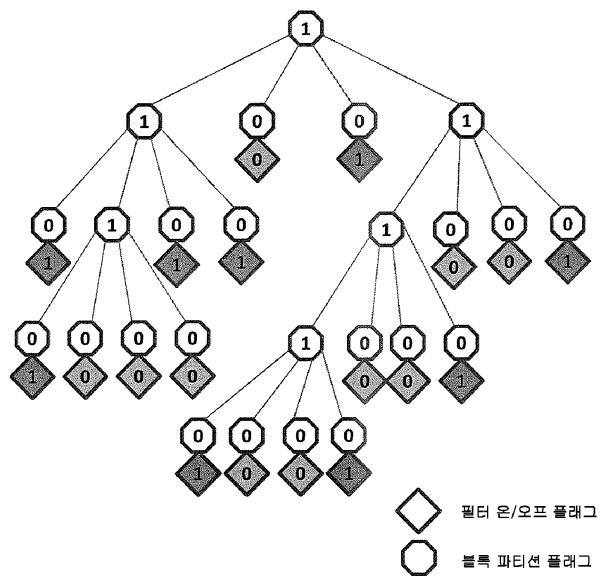
도면2b



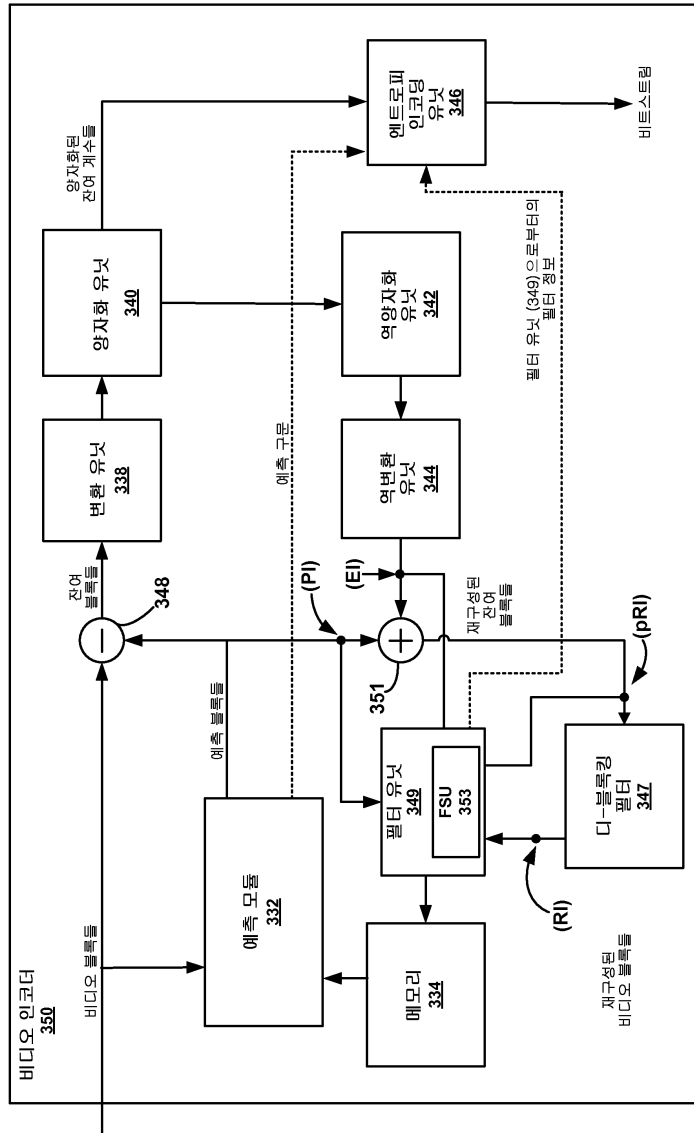
도면2c



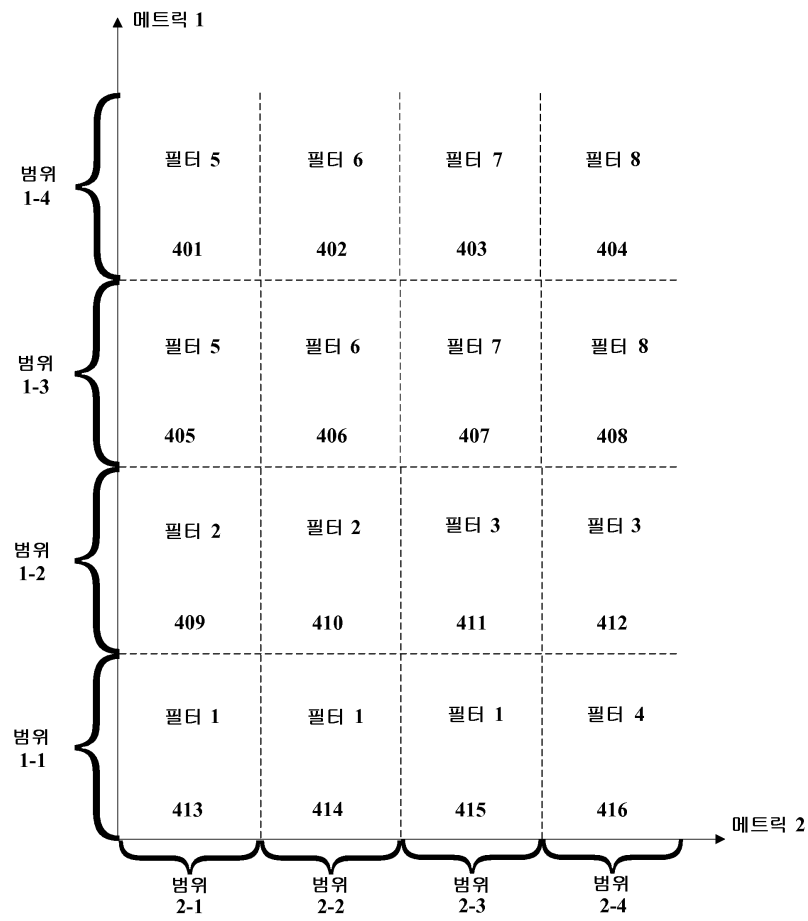
도면2d



도면3



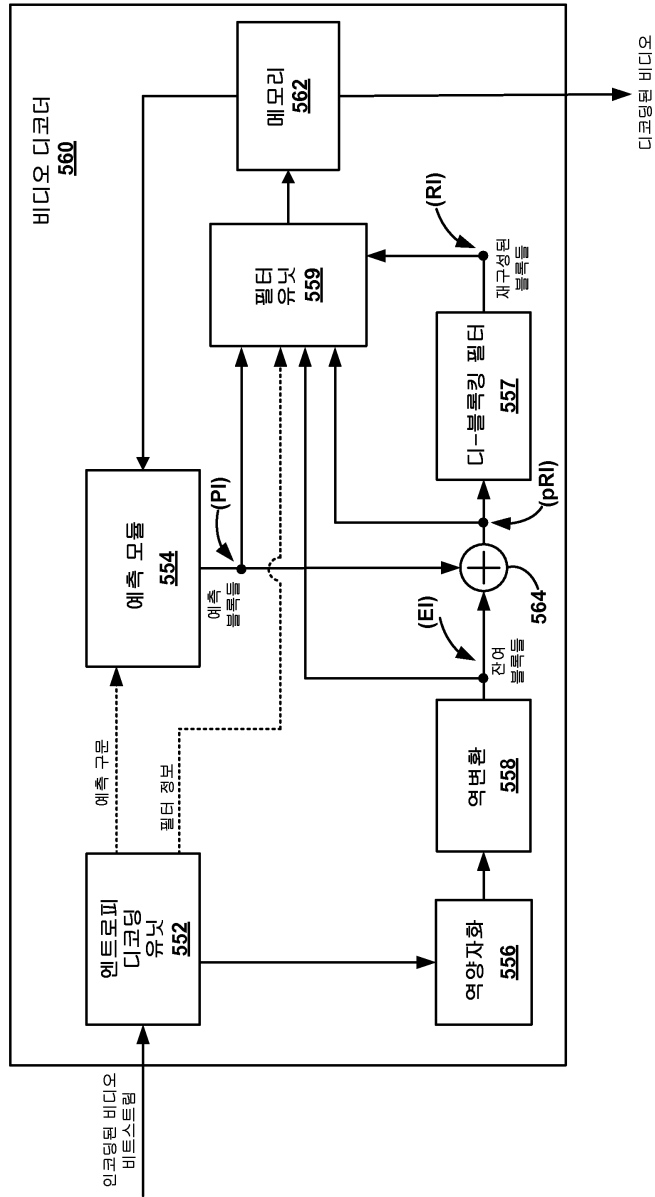
도면4a



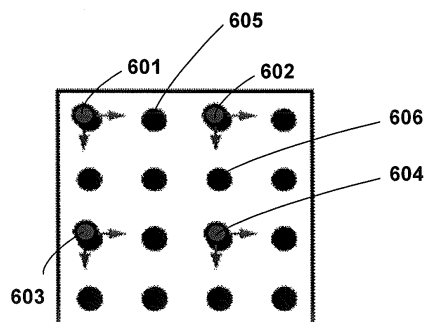
도면4b

| 방향  | 범위 1 |      |      |      |      | 범위 2 |      |      |      |      | 범위 3 |      |      |      |      | 범위 4 |      |      |      |      | 범위 5 |      |      |      |      |
|-----|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
|     | 필터 4 | 필터 4 | 필터 4 | 필터 4 | 필터 6 | 필터 1 | 필터 1 | 필터 1 | 필터 1 | 필터 2 | 필터 3 | 필터 3 | 필터 3 | 필터 3 | 필터 5 | 필터 2 | 필터 2 | 필터 2 | 필터 2 | 필터 5 | 필터 6 | 필터 6 | 필터 6 | 필터 6 | 필터 6 |
| 수직  | 431  | 432  | 433  | 434  | 435  |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |
| 수평  | 필터 3 | 필터 3 | 필터 3 | 필터 3 | 필터 5 | 필터 1 | 필터 1 | 필터 1 | 필터 1 | 필터 2 | 필터 3 | 필터 3 | 필터 3 | 필터 3 | 필터 5 | 필터 2 | 필터 2 | 필터 2 | 필터 2 | 필터 5 | 필터 6 | 필터 6 | 필터 6 | 필터 6 | 필터 6 |
| 대각선 | 421  | 422  | 423  | 424  | 425  | 426  | 427  | 428  | 429  | 430  | 431  | 432  | 433  | 434  | 435  | 436  | 437  | 438  | 439  | 440  | 441  | 442  | 443  | 444  | 445  |

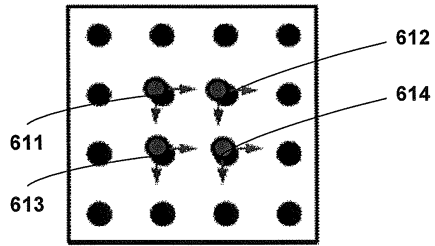
도면5



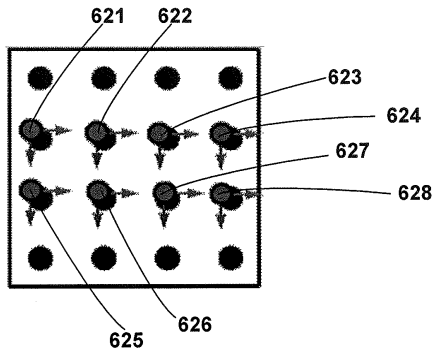
도면6a



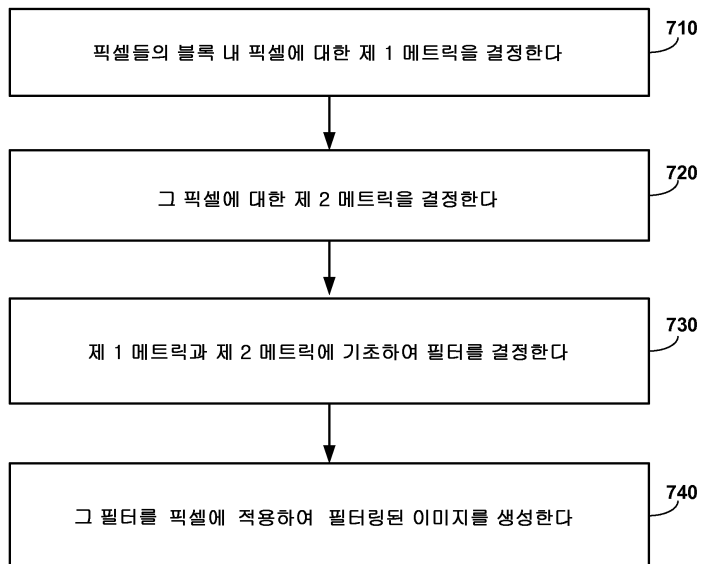
도면6b



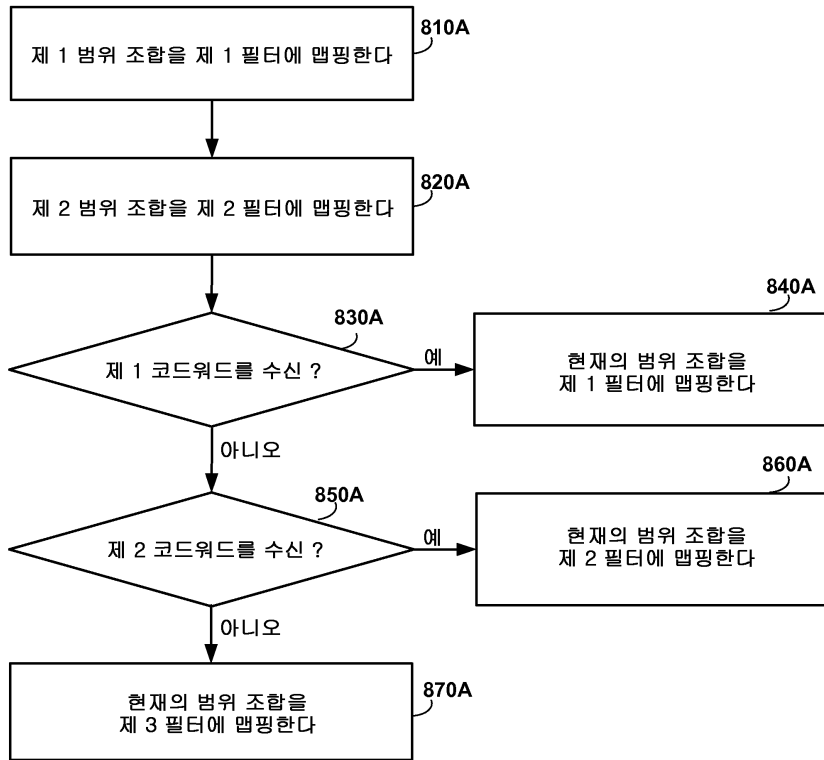
도면6c



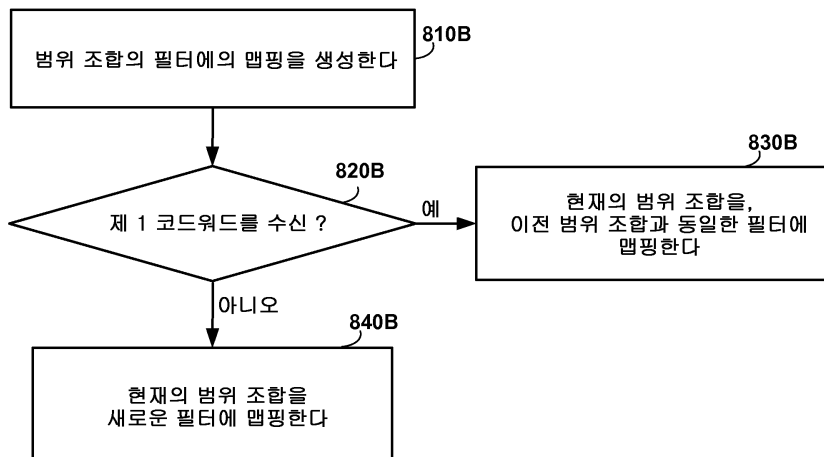
도면7



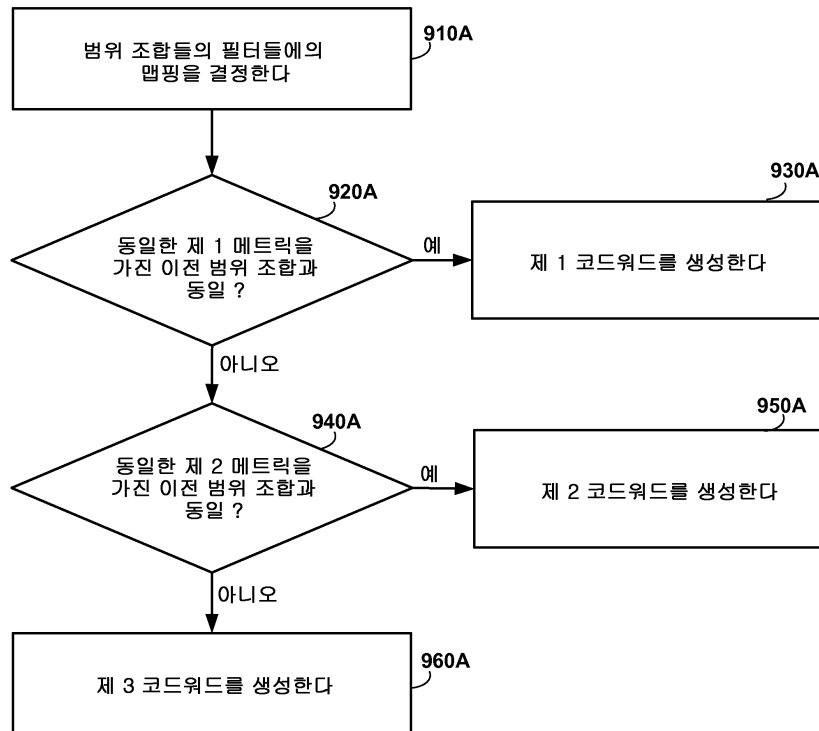
도면8a



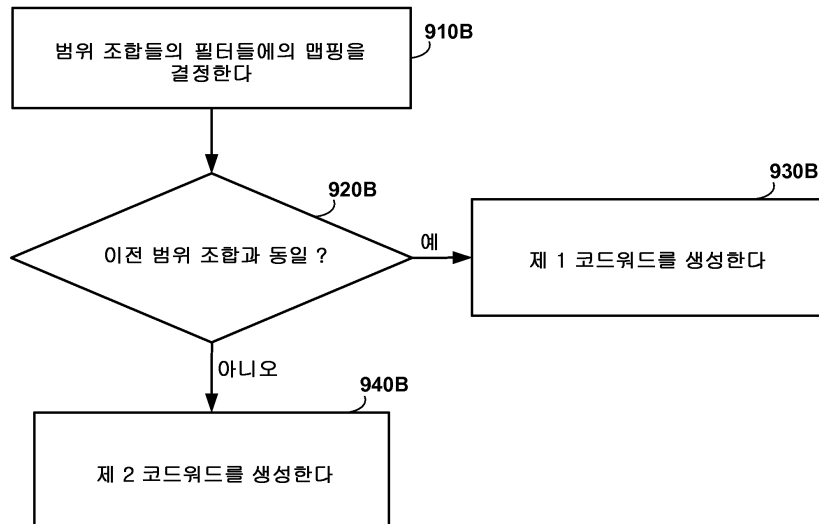
도면8b



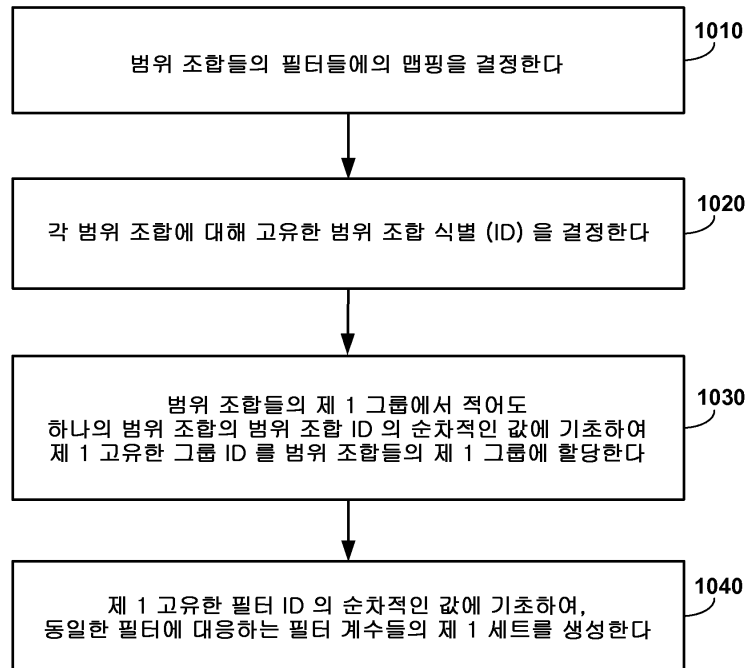
도면9a



도면9b



도면10



도면11

