(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2025년04월21일
(11) 등록번호 10-2796983
(24) 등록일자 2025년04월14일

(51) 국제특허분류(Int. Cl.)
H04N 19/117 (2014.01) *H04N 19/136* (2014.01)
H04N 19/14 (2014.01) *H04N 19/176* (2014.01)
H04N 19/463 (2014.01) *H04N 19/70* (2014.01)
H04N 19/80 (2014.01) *H04N 19/82* (2014.01)
(52) CPC특허분류
H04N 19/117 (2015.01)
H04N 19/136 (2015.01)
(21) 출원번호 10-2018-7023065
(22) 출원일자(국제) 2017년02월15일
심사청구일자 2022년01월26일
(85) 번역문제출일자 2018년08월09일
(65) 공개번호 10-2018-0113527
(43) 공개일자 2018년10월16일
(86) 국제출원번호 PCT/US2017/017958
(87) 국제공개번호 WO 2017/142939
국제공개일자 2017년08월24일
(30) 우선권주장
62/295,461 2016년02월15일 미국(US)
(뒷면에 계속)
(56) 선행기술조사문헌
US20110069752 A1*
Ehsan Maani, et al. Parametric Adaptive Loop Filter, JCTVC-E320, 2011.03.11.*
Jie Zhao and Andrew Segall,Improvements to ALF,Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11,7th Meeting: Geneva,2011년 11월, JCTVC-G610,pp.1-9
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
(72) 발명자
카르체비츠 마르타
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
치엔 웨이-정
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
장 리
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
(74) 대리인
특허법인코리아나

전체 청구항 수 : 총 37 항 심사관 : 이남숙

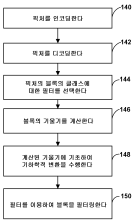
(54) 발명의 명칭 **비디오 코딩을 위한 필터들에 대한 기하학적 변환들**

(57) 요 약

비디오 데이터의 디코딩된 블록을 필터링하는 예시적인 디바이스는 회로부 내에 구현되는 하나 이상의 프로세서들을 포함하며, 하나 이상의 프로세서들은, 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하고, 현재의 블록의 픽셀들을 필터링하는데 사용될 (적응 루프 필터와 같은) 필터를 선택하고, 현재의 블록에 대한 적어도 하

(뒷면에 계속)

대 표 도



나의 픽셀의 기울기를 계산하고, 선택된 필터의 필터 서포트 영역 또는 계수들 중 하나에 대해 수행될 기하학적 변환을 선택하는 것으로서, 하나 이상의 프로세서들은 적어도 하나의 픽셀의 기울기의 방위에 대응하는 기하학적 변환을 선택하도록 구성되는, 상기 기하학적 변환을 선택하고, 선택된 필터의 필터 서포트 영역 또는 계수들에 대해 기하학적 변환을 수행하고, 그리고 기하학적 변환을 수행한 후, 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하도록 구성된다.

(52) CPC특허분류
H04N 19/14 (2015.01)
H04N 19/176 (2015.01)
H04N 19/463 (2015.01)
H04N 19/70 (2015.01)
H04N 19/80 (2015.01)
H04N 19/82 (2015.01)

(30) 우선권주장
62/324,776 2016년04월19일 미국(US)
15/432,839 2017년02월14일 미국(US)

명 세 서

청구범위

청구항 1

비디오 데이터의 디코딩된 블록을 필터링하는 방법으로서,

임의의 디블록 필터들과 분리된 복수의 적응 루프 필터들을 형성하는 단계;

상기 비디오 데이터의 현재의 슬라이스의 제 1 블록을 디코딩하는 단계;

상기 제 1 블록 내 픽셀들의 제 1 하나 이상의 기울기들을 계산하는 단계로서, 상기 제 1 하나 이상의 기울기들은 상기 제 1 블록에 대한 제 1 방향성 및 활동도를 나타내는, 상기 제 1 하나 이상의 기울기들을 계산하는 단계;

상기 복수의 적응 루프 필터들을 형성하는 단계 후에, 상기 제 1 블록의 적어도 하나의 픽셀을 필터링하는데 사용될 상기 복수의 적응 루프 필터들로부터 제 1 적응 루프 필터를 선택하는 단계로서, 상기 제 1 적응 루프 필터를 선택하는 단계는 상기 제 1 블록에 대한 상기 제 1 방향성 및 활동도에 따라 상기 제 1 적응 루프 필터를 선택하는 단계를 포함하는, 상기 제 1 적응 루프 필터를 선택하는 단계;

상기 제 1 적응 루프 필터의 제 1 필터 서포트 영역 또는 제 1 계수들 중 하나에 대해 수행될 제 1 기하학적 변환을 선택하는 단계로서, 상기 제 1 기하학적 변환을 선택하는 단계는 상기 제 1 하나 이상의 기울기들에 따라 상기 제 1 기하학적 변환을 선택하는 단계를 포함하는, 상기 제 1 기하학적 변환을 선택하는 단계;

상기 제 1 적응 루프 필터의 상기 제 1 필터 서포트 영역 또는 상기 제 1 계수들에 대해 상기 제 1 기하학적 변환을 수행하는 단계;

상기 제 1 기하학적 변환을 수행한 후, 상기 제 1 적응 루프 필터를 이용하여, 상기 제 1 블록의 적어도 하나의 픽셀을 필터링하는 단계;

상기 비디오 데이터의 상기 현재의 슬라이스의 제 2 블록을 디코딩하는 단계로서, 상기 제 2 블록은 상기 제 1 블록과 상이한, 상기 제 2 블록을 디코딩하는 단계;

상기 제 2 블록 내 픽셀들의 제 2 하나 이상의 기울기들을 계산하는 단계로서, 상기 제 2 하나 이상의 기울기들은 상기 제 2 블록에 대한 제 2 방향성 및 활동도를 나타내고, 상기 제 2 하나 이상의 기울기들은 상기 제 1 하나 이상의 기울기들과 상이한, 상기 제 2 하나 이상의 기울기들을 계산하는 단계;

상기 복수의 적응 루프 필터들을 형성하는 단계 후에, 상기 제 2 블록의 적어도 하나의 픽셀을 적응 루프 필터링하는데 사용될 상기 복수의 적응 루프 필터들로부터 제 2 적응 루프 필터를 선택하는 단계로서, 상기 제 2 적응 루프 필터는 상기 제 1 적응 루프 필터와 상이하고, 상기 제 2 적응 루프 필터를 선택하는 단계는 상기 제 2 블록에 대한 상기 제 2 방향성 및 활동도에 따라 상기 제 2 적응 루프 필터를 선택하는 단계를 포함하는, 상기 제 2 적응 루프 필터를 선택하는 단계;

상기 제 2 적응 루프 필터의 제 2 필터 서포트 영역 또는 제 2 계수들 중 하나에 대해 수행될 제 2 기하학적 변환을 선택하는 단계로서, 상기 제 2 기하학적 변환을 선택하는 단계는 상기 제 2 하나 이상의 기울기들에 따라 상기 제 2 기하학적 변환을 선택하는 단계를 포함하는, 상기 제 2 기하학적 변환을 선택하는 단계;

상기 제 2 적응 루프 필터의 상기 제 2 필터 서포트 영역 또는 상기 제 2 계수들에 대해 상기 제 2 기하학적 변환을 수행하는 단계; 및

상기 제 2 기하학적 변환을 수행한 후, 상기 제 2 적응 루프 필터를 이용하여, 상기 제 2 블록의 적어도 하나의 픽셀을 필터링하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법.

청구항 2

제 1 항에 있어서,

상기 제 1 기하학적 변환을 선택하는 단계는 회전 변환, 대각선 플립 변환, 또는 수직 플립 변환 중에서 상기

제 1 기하학적 변환을 선택하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법.

청구항 3

제 2 항에 있어서,

f(k,l) 이 상기 제 1 적응 루프 필터를 나타내며,

상기 회전 변환은 $f_R(k,l)=f(K-l-1,k)$ 를 포함하며,

상기 대각선 플립 변환은 $f_D(k,l)=f(l,k)$ 를 포함하며,

상기 수직 플립 변환은 $f_V(k,l)=f(k,K-l-1)$ 을 포함하며,

상기 K 는 상기 제 1 적응 루프 필터의 사이즈이고, k 및 l 은 상기 제 1 적응 루프 필터의 계수들의 좌표들 또는 상기 제 1 필터 서포트 영역에서의 값들의 좌표들이고, $0 \le k$, $l \le K-1$ 이며, 로케이션 (0, 0) 은 상기 제 1 적응 루프 필터 또는 상기 제 1 필터 서포트 영역의 좌상단 모서리에 있으며, 로케이션 (K-1, K-1) 은 상기 제 1 적응 루프 필터 또는 상기 제 1 필터 서포트 영역의 상기 좌상단 모서리의 우하단 모서리에 있는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법.

청구항 4

제 2 항에 있어서,

f(k,l) 이 상기 제 1 적응 루프 필터를 나타내며, R(i, j) 는 상기 현재의 슬라이스를 포함하는 현재의 픽처의 위치 (i, j) 에서의 픽셀을 나타내며,

상기 제 1 기하학적 변환을 선택하는 단계는,

다음에 따라서 수평 기울기 g_h 를 계산하는 단계:

$$g_h = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} H_{k,l}$$, 여기서, $$H_{k,l} = |2R(k,l) - R(k-1,l) - R(k+1,l)|$$;

다음에 따라서 수직 기울기 g_v 를 계산하는 단계:

$$g_v = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} V_{k,l}$$, 여기서, $$V_{k,l} = |2R(k,l) - R(k,l-1) - R(k,l+1)|$$;

다음에 따라서 제 1 대각선 기울기 g_{d1} 을 계산하는 단계:

$$g_{d1} = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} D1_{k,l}$$, 여기서,

$$D1_{k,l} = |2R(k,l) - R(k-1,l-1) - R(k+1,l+1)|$$;

다음에 따라서 제 2 대각선 기울기 g_{d2} 를 계산하는 단계:

$$g_{d2} = \sum_{k=i-M}^{i+N+M-1} \sum_{j=j-M}^{j+N+M-1} D2_{k,l}$$, 여기서,

$$D2_{k,l} = |2R(k,l) - R(k-1,l+1) - R(k+1,l-1)|$$;

$g_{d2} < g_{d1}$ 및 $g_v < g_h$ 일 때 상기 대각선 플립 변환을 선택하는 단계;

$g_{d1} < g_{d2}$ 및 $g_h < g_v$ 일 때 상기 수직 플립 변환을 선택하는 단계; 및

$g_{d1} < g_{d2}$ 및 $g_v < g_h$ 일 때 상기 회전 변환을 선택하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법.

청구항 5

삭제

청구항 6

제 1 항에 있어서,

상기 제 1 기하학적 변환을 선택하는 단계는 상기 제 1 하나 이상의 기울기들의 방위들에 대응하는 상기 제 1 기하학적 변환을 선택하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법.

청구항 7

제 1 항에 있어서,

상기 제 1 하나 이상의 기울기들을 계산하는 단계는 수평 기울기, 수직 기울기, 45 도 대각선 기울기, 또는 135 도 대각선 기울기 중 적어도 하나를 계산하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법.

청구항 8

제 1 항에 있어서,

상기 제 1 필터 서포트 영역은 상기 제 1 적응 루프 필터의 계수들이 적용될 상기 제 1 블록의 상기 적어도 하나의 픽셀에 대한 복수의 이웃하는 픽셀들을 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법.

청구항 9

삭제

청구항 10

제 1 항에 있어서,

상기 제 1 블록 및 상기 제 2 블록을 디코딩하기 전에, 상기 제 1 블록 및 상기 제 2 블록을 인코딩하는 단계를 더 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법.

청구항 11

제 1 항에 있어서,

상기 방법은 무선 통신 디바이스 상에서 실행가능하며,

상기 무선 통신 디바이스는,

상기 비디오 데이터를 저장하도록 구성된 메모리;

상기 메모리에 저장된 상기 비디오 데이터를 프로세싱하는 명령들을 실행하도록 구성된 프로세서; 및

상기 비디오 데이터를 수신하고 상기 비디오 데이터를 상기 메모리에 저장하도록 구성된 수신기를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법.

청구항 12

제 11 항에 있어서,

상기 무선 통신 디바이스는 셀룰러 전화기이며, 상기 비디오 데이터는 수신기에 의해 수신되어 셀룰러 통신 표준에 따라서 변조되는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법.

청구항 13

비디오 데이터의 디코딩된 블록을 필터링하는 디바이스로서,

상기 비디오 데이터를 저장하도록 구성된 메모리; 및

회로부 내에 구현되는 하나 이상의 프로세서들을 포함하며,

상기 하나 이상의 프로세서들은,

임의의 디블록 필터들과 분리된 복수의 적응 루프 필터들을 형성하고;

상기 비디오 데이터의 현재의 슬라이스의 제 1 블록을 디코딩하고;

상기 제 1 블록 내 픽셀들의 제 1 하나 이상의 기울기들을 계산하는 것으로서, 상기 제 1 하나 이상의 기울기들은 상기 제 1 블록에 대한 제 1 방향성 및 활동도를 나타내는, 상기 제 1 하나 이상의 기울기들을 계산하고;

상기 복수의 적응 루프 필터들을 형성한 후에, 상기 제 1 블록의 적어도 하나의 픽셀을 필터링하는데 사용될 상기 복수의 적응 루프 필터들로부터 제 1 적응 루프 필터를 선택하는 것으로서, 상기 하나 이상의 프로세서들은 상기 제 1 블록에 대한 상기 제 1 방향성 및 활동도에 따라 상기 제 1 적응 루프 필터를 선택하도록 구성되는, 상기 제 1 적응 루프 필터를 선택하고;

상기 제 1 적응 루프 필터의 제 1 필터 서포트 영역 또는 제 1 계수들 중 하나에 대해 수행될 제 1 기하학적 변환을 선택하는 것으로서, 상기 하나 이상의 프로세서들은 상기 제 1 하나 이상의 기울기들에 따라 상기 제 1 기하학적 변환을 선택하도록 구성되는, 상기 제 1 기하학적 변환을 선택하고;

상기 제 1 적응 루프 필터의 상기 제 1 필터 서포트 영역 또는 상기 제 1 계수들에 대해 상기 제 1 기하학적 변환을 수행하고;

상기 제 1 기하학적 변환을 수행한 후, 상기 제 1 적응 루프 필터를 이용하여, 상기 제 1 블록의 적어도 하나의 픽셀을 필터링하고;

상기 비디오 데이터의 상기 현재의 슬라이스의 제 2 블록을 디코딩하는 단계로서, 상기 제 2 블록은 상기 제 1 블록과 상이한, 상기 제 2 블록을 디코딩하고;

상기 제 2 블록 내 픽셀들의 제 2 하나 이상의 기울기들을 계산하는 단계로서, 상기 제 2 하나 이상의 기울기들은 상기 제 2 블록에 대한 제 2 방향성 및 활동도를 나타내고, 상기 제 2 하나 이상의 기울기들은 상기 제 1 하나 이상의 기울기들과 상이한, 상기 제 2 하나 이상의 기울기들을 계산하고;

상기 복수의 적응 루프 필터들을 형성한 후에, 상기 제 2 블록의 적어도 하나의 픽셀을 적응 루프 필터링하는데 사용될 상기 복수의 적응 루프 필터들로부터 제 2 적응 루프 필터를 선택하는 것으로서, 상기 제 2 적응 루프 필터는 상기 제 1 적응 루프 필터와 상이하고, 상기 제 2 적응 루프 필터를 선택하기 위해 상기 하나 이상의 프로세서들은 상기 제 2 블록에 대한 상기 제 2 방향성 및 활동도에 따라 상기 제 2 적응 루프 필터를 선택하도록 구성되는, 상기 제 2 적응 루프 필터를 선택하고;

상기 제 2 적응 루프 필터의 제 2 필터 서포트 영역 또는 제 2 계수들 중 하나에 대해 수행될 제 2 기하학적 변환을 선택하는 것으로서, 상기 제 2 기하학적 변환을 선택하기 위해 상기 하나 이상의 프로세서들은 상기 제 2 하나 이상의 기울기들에 따라 상기 제 2 기하학적 변환을 선택하도록 구성되는, 상기 제 2 기하학적 변환을 선택하고;

상기 제 2 적응 루프 필터의 상기 제 2 필터 서포트 영역 또는 상기 제 2 계수들에 대해 상기 제 2 기하학적 변환을 수행하고; 그리고

상기 제 2 기하학적 변환을 수행한 후, 상기 제 2 적응 루프 필터를 이용하여, 상기 제 2 블록의 적어도 하나의 픽셀을 필터링하도록

구성되는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 14

제 13 항에 있어서,

상기 하나 이상의 프로세서들은, 회전, 대각선 플립, 또는 수직 플립 중에서 상기 제 1 기하학적 변환을 선택하도록 구성되는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 15

제 14 항에 있어서,

f(k,l) 이 상기 제 1 적응 루프 필터를 나타내며,

상기 회전 변환은 $f_R(k,l)=f(K-l-1,k)$ 를 포함하며,

상기 대각선 플립 변환은 $f_D(k,l)=f(l,k)$ 를 포함하며,

상기 수직 플립 변환은 $f_V(k,l)=f(k,K-l-1)$ 을 포함하며,

상기 K 는 상기 제 1 적응 루프 필터의 사이즈이고, k 및 l 은 상기 제 1 적응 루프 필터의 계수들의 좌표들 또는 상기 제 1 필터 서포트 영역에서의 값들의 좌표들이고, $0 \le k, l \le K-1$ 이며, 로케이션 (0,0) 은 상기 제 1 적응 루프 필터 또는 상기 제 1 필터 서포트 영역의 좌상단 모서리에 있으며, 로케이션 (K-1, K-1) 은 상기 제 1 적응 루프 필터 또는 상기 제 1 필터 서포트 영역의 상기 좌상단 모서리의 우하단 모서리에 있는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 16

제 14 항에 있어서,

f(k,l) 이 상기 제 1 적응 루프 필터를 나타내며, R(i, j) 는 상기 현재의 슬라이스의 위치 (i, j) 에서의 픽셀을 나타내며,

상기 제 1 기하학적 변환을 선택하기 위해, 상기 하나 이상의 프로세서들은,

다음에 따라서 수평 기울기 g_h 를 계산하고:

$$g_h = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} H_{k,l}$$, 여기서, $$H_{k,l} = |2R(k,l) - R(k-1,l) - R(k+1,l)|$$;

다음에 따라서 수직 기울기 g_v 를 계산하고:

$$g_v = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} V_{k,l}$$, 여기서, $$V_{k,l} = |2R(k,l) - R(k,l-1) - R(k,l+1)|$$;

다음에 따라서 제 1 대각선 기울기 g_{d1} 을 계산하고:

$$g_{d1} = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} D1_{k,l}$$, 여기서, $$D1_{k,l} = |2R(k,l) - R(k-1,l-1) - R(k+1,l+1)|$$;

다음에 따라서 제 2 대각선 기울기 g_{d2} 를 계산하고:

$$g_{d2} = \sum_{k=i-M}^{i+N+M-1} \sum_{j=j-M}^{j+N+M-1} D2_{k,l}$$, 여기서, $$D2_{k,l} = |2R(k,l) - R(k-1,l+1) - R(k+1,l-1)|$$;

$g_{d2} < g_{d1}$ 및 $g_v < g_h$ 일 때 상기 대각선 플립 변환을 선택하고;

$g_{d1} < g_{d2}$ 및 $g_h < g_v$ 일 때 상기 수직 플립 변환을 선택하고; 그리고

$g_{d1} < g_{d2}$ 및 $g_v < g_h$ 일 때 상기 회전 변환을 선택하도록

구성되는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 17

삭제

청구항 18

제 13 항에 있어서,

상기 하나 이상의 프로세서들은 상기 제 1 하나 이상의 기울기들의 방위들에 대응하는 상기 제 1 기하학적 변환을 선택하도록 구성되는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 19

제 13 항에 있어서,

상기 하나 이상의 프로세서들은 수평 기울기, 수직 기울기, 45 도 대각선 기울기, 또는 135 도 대각선 기울기 중 적어도 하나를 계산하도록 구성되는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 20

제 13 항에 있어서,

상기 제 1 필터 서포트 영역은 상기 제 1 적응 루프 필터의 계수들이 적용될 상기 제 1 블록의 상기 적어도 하나의 픽셀에 대한 복수의 이웃하는 픽셀들을 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 21

삭제

청구항 22

제 13 항에 있어서,

하나 이상의 프로세싱 유닛들은 상기 제 1 블록 및 상기 제 2 블록을 디코딩하기 전에, 상기 제 1 블록 및 상기 제 2 블록을 인코딩하도록 더 구성되는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 23

제 13 항에 있어서,

상기 디바이스는 무선 통신 디바이스이며,

상기 현재의 슬라이스를 포함하는 비디오 데이터를 수신하도록 구성된 수신기를 더 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 24

제 23 항에 있어서,

상기 무선 통신 디바이스는 셀룰러 전화기이며, 상기 비디오 데이터는 상기 수신기에 의해 수신되어 셀룰러 통신 표준에 따라서 변조되는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 25

비디오 데이터의 디코딩된 블록을 필터링하는 디바이스로서,

임의의 디블록 필터들과 분리된 복수의 적응 루프 필터들을 형성하는 수단;

상기 비디오 데이터의 현재의 슬라이스의 제 1 블록을 디코딩하는 수단;

상기 제 1 블록 내 픽셀들의 제 1 하나 이상의 기울기들을 계산하는 수단으로서, 상기 제 1 하나 이상의 기울기들은 상기 제 1 블록에 대한 제 1 방향성 및 활동도를 나타내는, 상기 제 1 하나 이상의 기울기들을 계산하는 수단;

상기 복수의 적응 루프 필터들을 형성한 후, 상기 제 1 블록의 적어도 하나의 픽셀을 필터링하는데 사용될 상기 복수의 적응 루프 필터들로부터 제 1 적응 루프 필터를 선택하는 수단으로서, 상기 제 1 적응 루프 필터를 선택

하는 수단은 상기 제 1 블록에 대한 상기 제 1 방향성 및 활동도에 따라 상기 제 1 적응 루프 필터를 선택하는 수단을 포함하는, 상기 제 1 적응 루프 필터를 선택하는 수단;

상기 제 1 적응 루프 필터의 제 1 필터 서포트 영역 또는 제 1 계수들 중 하나에 대해 수행될 제 1 기하학적 변환을 선택하는 수단으로서, 상기 제 1 기하학적 변환을 선택하는 수단은 상기 제 1 하나 이상의 기울기들에 따라 상기 제 1 기하학적 변환을 선택하는 수단을 포함하는, 상기 제 1 기하학적 변환을 선택하는 수단;

상기 제 1 적응 루프 필터의 상기 제 1 필터 서포트 영역 또는 상기 제 1 계수들에 대해 상기 제 1 기하학적 변환을 수행하는 수단;

상기 기하학적 변환을 수행한 후, 상기 제 1 적응 루프 필터를 이용하여, 상기 제 1 블록의 적어도 하나의 픽셀을 필터링하는 수단;

상기 비디오 데이터의 상기 현재의 슬라이스의 제 2 블록을 디코딩하는 수단으로서, 상기 제 2 블록은 상기 제 1 블록과 상이한, 상기 제 2 블록을 디코딩하는 수단;

상기 제 2 블록 내 픽셀들의 제 2 하나 이상의 기울기들을 계산하는 수단으로서, 상기 제 2 하나 이상의 기울기들은 상기 제 2 블록에 대한 제 2 방향성 및 활동도를 나타내고, 상기 제 2 하나 이상의 기울기들은 상기 제 1 하나 이상의 기울기들과 상이한, 상기 제 2 하나 이상의 기울기들을 계산하는 수단;

상기 복수의 적응 루프 필터들을 형성한 후에, 상기 제 2 블록의 적어도 하나의 픽셀을 필터링하는데 사용될 상기 복수의 적응 루프 필터들로부터 제 2 적응 루프 필터를 선택하는 수단으로서, 상기 제 2 적응 루프 필터는 상기 제 1 적응 루프 필터와 상이하고, 상기 제 2 적응 루프 필터를 선택하는 수단은 상기 제 2 블록에 대한 상기 제 2 방향성 및 활동도에 따라 상기 제 2 적응 루프 필터를 선택하는 수단을 포함하는, 상기 제 2 적응 루프 필터를 선택하는 수단;

상기 제 2 적응 루프 필터의 제 2 필터 서포트 영역 또는 제 2 계수들 중 하나에 대해 수행될 제 2 기하학적 변환을 선택하는 수단으로서, 상기 제 2 기하학적 변환을 선택하는 수단은 상기 제 2 하나 이상의 기울기들에 따라 상기 제 2 기하학적 변환을 선택하는 수단을 포함하는, 상기 제 2 기하학적 변환을 선택하는 수단;

상기 제 2 적응 루프 필터의 상기 제 2 필터 서포트 영역 또는 상기 제 2 계수들에 대해 상기 제 2 기하학적 변환을 수행하는 수단; 및

상기 제 2 기하학적 변환을 수행한 후, 상기 제 2 적응 루프 필터를 이용하여, 상기 제 2 블록의 적어도 하나의 픽셀을 필터링하는 수단을 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 26

제 25 항에 있어서,

상기 제 1 기하학적 변환을 선택하는 수단은 회전 변환, 대각선 플립 변환, 또는 수직 플립 변환 중에서 상기 제 1 기하학적 변환을 선택하는 수단을 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 27

제 26 항에 있어서,

$f(k,l)$ 이 상기 제 1 적응 루프 필터를 나타내며,

상기 회전 변환은 $f_R(k,l)=f(K-l-1,k)$ 를 포함하며,

상기 대각선 플립 변환은 $f_D(k,l)=f(l,k)$ 를 포함하며,

상기 수직 플립 변환은 $f_V(k,l)=f(k,K-l-1)$ 을 포함하며,

상기 K 는 상기 제 1 적응 루프 필터의 사이즈이고, k 및 l 은 상기 제 1 적응 루프 필터의 계수들의 좌표들 또는 상기 제 1 필터 서포트 영역에서의 값들의 좌표들이고, $0 \le k, l \le K-1$ 이며, 로케이션 (0,0) 은 상기 제 1 적응 루프 필터 또는 상기 제 1 필터 서포트 영역의 좌상단 모서리에 있으며, 로케이션 (K-1, K-1) 은 상기 제 1 적응 루프 필터 또는 상기 제 1 필터 서포트 영역의 상기 좌상단 모서리의 우하단 모서리에 있는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 28

제 26 항에 있어서,

f(k,l) 이 상기 제 1 적응 루프 필터를 나타내며, R(i, j) 는 상기 현재의 슬라이스의 위치 (i, j) 에서의 픽셀을 나타내며,

상기 제 1 기하학적 변환을 선택하는 수단은,

다음에 따라서 수평 기울기 g_h 를 계산하는 수단:

$g_h = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} H_{k,l}$, 여기서, $H_{k,l} = |2R(k,l) - R(k-1,l) - R(k+1,l)|$;

다음에 따라서 수직 기울기 g_v 를 계산하는 수단:

$g_v = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} V_{k,l}$, 여기서, $V_{k,l} = |2R(k,l) - R(k,l-1) - R(k,l+1)|$;

다음에 따라서 제 1 대각선 기울기 g_{d1} 을 계산하는 수단:

$g_{d1} = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} D1_{k,l}$, 여기서,

$D1_{k,l} = |2R(k,l) - R(k-1,l-1) - R(k+1,l+1)|$;

다음에 따라서 제 2 대각선 기울기 g_{d2} 를 계산하는 수단:

$g_{d2} = \sum_{k=i-M}^{i+N+M-1} \sum_{j=j-M}^{j+N+M-1} D2_{k,l}$, 여기서,

$D2_{k,l} = |2R(k,l) - R(k-1,l+1) - R(k+1,l-1)|$;

$g_{d2} < g_{d1}$ 및 $g_v < g_h$ 일 때 상기 대각선 플립 변환을 선택하는 수단;

$g_{d1} < g_{d2}$ 및 $g_h < g_v$ 일 때 상기 수직 플립 변환을 선택하는 수단; 및

$g_{d1} < g_{d2}$ 및 $g_v < g_h$ 일 때 상기 회전 변환을 선택하는 수단을 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 29

삭제

청구항 30

제 25 항에 있어서,

상기 제 1 기하학적 변환을 선택하는 수단은 하나 이상의 픽셀의 상기 제 1 하나 이상의 기울기들의 방위들에 대응하는 제 1 기하학적 변환을 선택하는 수단을 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 31

제 25 항에 있어서,

상기 제 1 하나 이상의 기울기들을 계산하는 수단은 수평 기울기, 수직 기울기, 45 도 대각선 기울기, 또는 135 도 대각선 기울기 중 적어도 하나를 계산하는 수단을 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 32

제 25 항에 있어서,

상기 제 1 필터 서포트 영역은 상기 제 1 적응 루프 필터의 계수들이 적용될 상기 제 1 블록의 상기 적어도 하나의 픽셀에 대한 복수의 이웃하는 픽셀들을 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 33

제 25 항에 있어서,

상기 제 1 적응 루프 필터를 선택하는 수단은 상기 제 1 블록에 대한 제 1 클래스에 기초하여 상기 제 1 적응 루프 필터를 선택하는 수단을 포함하며,

상기 제 2 적응 루프 필터를 선택하는 수단은 상기 제 2 블록에 대한 제 2 클래스에 기초하여 상기 제 2 적응 루프 필터를 선택하는 수단을 포함하고,

상기 제 1 클래스는 상기 제 2 클래스와 상이하고,

상기 제 1 클래스 및 상기 제 2 클래스는 텍스처, 강한 수평/수직, 수평/수직, 강한 대각선, 또는 대각선 중 하나를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 34

제 25 항에 있어서,

상기 제 1 블록 및 상기 제 2 블록을 디코딩하기 전에, 상기 제 1 블록 및 상기 제 2 블록을 인코딩하는 수단을 더 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스.

청구항 35

명령들을 저장하는 비-일시적 컴퓨터-판독가능 저장 매체로서,

상기 명령들은, 실행될 때, 프로세서로 하여금,

임의의 디블록 필터들과 분리된 복수의 적응 루프 필터들을 형성하게 하고;

비디오 데이터의 현재의 슬라이스의 제 1 블록을 디코딩하게 하고;

상기 제 1 블록 내 픽셀들의 제 1 하나 이상의 기울기들을 계산하게 하는 것으로서, 상기 제 1 하나 이상의 기울기들은 상기 제 1 블록에 대한 제 1 방향성 및 활동도를 나타내는, 상기 제 1 하나 이상의 기울기들을 계산하게 하고;

상기 복수의 적응 루프 필터들을 형성한 후에, 상기 제 1 블록의 적어도 하나의 픽셀을 필터링하는데 사용될 상기 복수의 적응 루프 필터들로부터 제 1 적응 루프 필터를 선택하게 하는 것으로서, 상기 프로세서로 하여금, 상기 제 1 적응 루프 필터를 선택하게 하는 명령들은, 상기 프로세서로 하여금, 상기 제 1 블록에 대한 상기 제 1 방향성 및 활동도에 따라 상기 제 1 적응 루프 필터를 선택하게 하는 명령들을 포함하는, 상기 제 1 적응 루프 필터를 선택하게 하고;

상기 제 1 적응 루프 필터의 제 1 필터 서포트 영역 또는 제 1 계수들 중 하나에 대해 수행될 제 1 기하학적 변환을 선택하게 하는 것으로서, 상기 프로세서로 하여금, 상기 제 1 기하학적 변환을 선택하게 하는 명령들은, 상기 프로세서로 하여금, 상기 제 1 하나 이상의 기울기들에 따라 상기 제 1 기하학적 변환을 선택하게 하는 명령들을 포함하는, 상기 제 1 기하학적 변환을 선택하게 하고;

상기 제 1 적응 루프 필터의 상기 제 1 필터 서포트 영역 또는 상기 제 1 계수들에 대해 상기 제 1 기하학적 변환을 수행하게 하고; 그리고

상기 제 1 기하학적 변환을 수행한 후, 상기 제 1 적응 루프 필터를 이용하여, 상기 제 1 블록의 적어도 하나의 픽셀을 필터링하게 하고;

상기 비디오 데이터의 상기 현재의 슬라이스의 제 2 블록을 디코딩하는 것으로서, 상기 제 2 블록은 상기 제 1 블록과 상이한, 상기 제 2 블록을 디코딩하게 하고;

상기 제 2 블록 내 픽셀들의 제 2 하나 이상의 기울기들을 계산하는 것으로서, 상기 제 2 하나 이상의 기울기들은 상기 제 2 블록에 대한 제 2 방향성 및 활동도를 나타내고, 상기 제 2 하나 이상의 기울기들은 상기 제 1 하

나 이상의 기울기들과 상이한, 상기 제 2 하나 이상의 기울기들을 계산하게 하고;

상기 복수의 적응 루프 필터들을 형성한 후에, 상기 제 2 블록의 적어도 하나의 픽셀을 필터링하는데 사용될 상기 복수의 적응 루프 필터들로부터 제 2 적응 루프 필터를 선택하는 것으로서, 상기 제 2 적응 루프 필터는 상기 제 1 적응 루프 필터와 상이하고, 상기 프로세스로 하여금, 상기 제 2 적응 루프 필터를 선택하게 하는 명령들은, 상기 프로세스로 하여금, 상기 제 2 블록에 대한 상기 제 2 방향성 및 활동도에 따라 상기 제 2 적응 루프 필터를 선택하게 하는 명령들을 포함하는, 상기 제 2 적응 루프 필터를 선택하게 하고;

상기 제 2 적응 루프 필터의 제 2 필터 서포트 영역 또는 제 2 계수들 중 하나에 대해 수행될 제 2 기하학적 변환을 선택하는 것으로서, 상기 프로세스로 하여금, 상기 제 2 기하학적 변환을 선택하게 하는 명령들은, 상기 프로세스로 하여금, 상기 제 2 하나 이상의 기울기들에 따라 상기 제 2 기하학적 변환을 선택하게 하는 명령들을 포함하는, 상기 제 2 기하학적 변환을 선택하게 하고;

상기 제 2 적응 루프 필터의 상기 제 2 필터 서포트 영역 또는 상기 제 2 계수들에 대해 상기 제 2 기하학적 변환을 수행하게 하고; 및

상기 제 2 기하학적 변환을 수행한 후, 상기 제 2 적응 루프 필터를 이용하여, 상기 제 2 블록의 적어도 하나의 픽셀을 필터링하게 하는, 비-일시적 컴퓨터-판독가능 저장 매체.

청구항 36

제 35 항에 있어서,

상기 프로세스로 하여금, 상기 제 1 기하학적 변환을 선택하게 하는 명령들은, 상기 프로세스로 하여금, 회전 변환, 대각선 플립 변환, 또는 수직 플립 변환 중에서 상기 제 1 기하학적 변환을 선택하게 하는 명령들을 포함하는, 비-일시적 컴퓨터-판독가능 저장 매체.

청구항 37

제 36 항에 있어서,

f(k,l) 이 상기 제 1 적응 루프 필터를 나타내며,

상기 회전 변환은 $f_R(k,l)=f(K-l-1,k)$ 를 포함하며,

상기 대각선 플립 변환은 $f_D(k,l)=f(l,k)$ 를 포함하며,

상기 수직 플립 변환은 $f_V(k,l)=f(k,K-l-1)$ 을 포함하며,

상기 K 는 상기 제 1 적응 루프 필터의 사이즈이고, k 및 l 은 상기 제 1 적응 루프 필터의 계수들의 좌표들 또는 상기 제 1 필터 서포트 영역에서의 값들의 좌표들이고, $0 \le k, l \le K-1$ 이며, 로케이션 (0,0) 은 상기 제 1 적응 루프 필터 또는 상기 제 1 필터 서포트 영역의 좌상단 모서리에 있으며, 로케이션 (K-1, K-1) 은 상기 제 1 적응 루프 필터 또는 상기 제 1 필터 서포트 영역의 상기 좌상단 모서리의 우하단 모서리에 있는, 비-일시적 컴퓨터-판독가능 저장 매체.

청구항 38

제 36 항에 있어서,

f(k,l) 이 상기 제 1 적응 루프 필터를 나타내며, R(i, j) 는 상기 현재의 슬라이스를 포함하는 현재의 픽처의 위치 (i, j) 에서의 픽셀을 나타내며,

상기 프로세서로 하여금 상기 기하학적 변환을 선택하게 하는 명령들은, 상기 프로세서로 하여금,

다음에 따라서 수평 기울기 g_h 를 계산하게 하고:

$g_h = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} H_{k,l}$, 여기서, $H_{k,l} = |2R(k,l) - R(k-1,l) - R(k+1,l)|$;

다음에 따라서 수직 기울기 g_v 를 계산하게 하고:

$g_v = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} V_{k,l}$, 여기서, $V_{k,l} = |2R(k,l) - R(k,l-1) - R(k,l+1)|$;

다음에 따라서 제 1 대각선 기울기 g_{d1} 을 계산하게 하고:

$g_{d1} = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} D1_{k,l}$, 여기서,

$D1_{k,l} = |2R(k,l) - R(k-1,l-1) - R(k+1,l+1)|$;

다음에 따라서 제 2 대각선 기울기 g_{d2} 를 계산하게 하고:

$g_{d2} = \sum_{k=i-M}^{i+N+M-1} \sum_{j=j-M}^{j+N+M-1} D2_{k,l}$, 여기서,

$D2_{k,l} = |2R(k,l) - R(k-1,l+1) - R(k+1,l-1)|$;

$g_{d2} < g_{d1}$ 및 $g_v < g_h$ 일 때 상기 대각선 플립 변환을 선택하게 하고;

$g_{d1} < g_{d2}$ 및 $g_h < g_v$ 일 때 상기 수직 플립 변환을 선택하게 하고; 그리고

$g_{d1} < g_{d2}$ 및 $g_v < g_h$ 일 때 상기 회전 변환을 선택하게 하는

명령들을 포함하는, 비-일시적 컴퓨터-판독가능 저장 매체.

청구항 39

삭제

청구항 40

제 35 항에 있어서,

상기 프로세서로 하여금, 상기 제 1 기하학적 변환을 선택하게 하는 명령들은, 상기 프로세서로 하여금, 상기 제 1 하나 이상의 기울기들의 방위들에 대응하는 상기 제 1 기하학적 변환을 선택하게 하는 명령들을 포함하는, 비-일시적 컴퓨터-판독가능 저장 매체.

청구항 41

제 35 항에 있어서,

상기 프로세서로 하여금, 상기 제 1 하나 이상의 기울기들을 계산하게 하는 명령들은, 상기 프로세서로 하여금, 수평 기울기, 수직 기울기, 45 도 대각선 기울기, 또는 135 도 대각선 기울기 중 적어도 하나를 계산하게 하는 명령들을 포함하는, 비-일시적 컴퓨터-판독가능 저장 매체.

청구항 42

제 35 항에 있어서,

상기 필터 서포트 영역은 상기 제 1 적응 루프 필터의 계수들이 적용될 상기 제 1 블록의 상기 적어도 하나의 픽셀에 대한 복수의 이웃하는 픽셀들을 포함하는, 비-일시적 컴퓨터-판독가능 저장 매체.

청구항 43

삭제

청구항 44

제 35 항에 있어서,

상기 프로세서로 하여금, 상기 제 1 블록 및 상기 제 2 블록을 디코딩하기 전에, 상기 제 1 블록 및 상기 제 2 블록을 인코딩하게 하는 명령들을 더 포함하는, 비-일시적 컴퓨터-판독가능 저장 매체.

발명의 설명

기 술 분 야

[0001] 본 출원은 2016년 2월 15일에 출원된 미국 가출원 번호 제 62/295,461호, 및 2016년 4월 19일에 출원된 미국 가출원 번호 제 62/324,776호의 이익을 주장하며, 이의 각각의 전체 내용이 본원에 참고로 포함된다.

[0002] **기술 분야**

[0003] 본 개시물은 비디오 코딩에 관한 것이다.

배 경 기 술

[0004] 디지털 비디오 능력들은, 디지털 텔레비전들, 디지털 직접 브로드캐스트 시스템들, 무선 브로드캐스트 시스템들, 개인 휴대정보 단말기들 (PDAs), 랩탑 또는 데스크탑 컴퓨터들, 태블릿 컴퓨터들, e-북 리더들, 디지털 카메라들, 디지털 리코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 무선 전화기들, 소위 "스마트폰들", 원격 화상회의 디바이스들, 비디오 스트리밍 디바이스들 등을 포함한, 광범위한 디바이스들에 통합될 수 있다. 디지털 비디오 디바이스들은 MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, 파트 10, AVC (Advanced Video Coding), 최근 완결된 ITU-T H.265, HEVC (High Efficiency Video Coding) 표준, 및 이런 표준들의 확장판들에 의해 정의된 표준들에서 설명되는 것들과 같은, 비디오 압축 기법들을 구현한다. 비디오 디바이스들은 이런 비디오 압축 기법들을 구현함으로써, 디지털 비디오 정보를 좀더 효율적으로 송신, 수신, 인코딩, 디코딩, 및/또는 저장할 수도 있다.

[0005] 비디오 압축 기법들은 비디오 시퀀스들에 내재하는 리던던시를 감소시키거나 또는 제거하기 위해 공간 (인트라-픽처) 예측 및/또는 시간 (인터-픽처) 예측을 수행한다. 블록-기반 비디오 코딩에 있어, 비디오 슬라이스 (즉, 비디오 프레임 또는 비디오 프레임의 부분) 는 비디오 블록들로 파티셔닝될 수도 있으며, 이 비디오 블록들은 또한 트리블록들, 코딩 유닛들 (CUs) 및/또는 코딩 노드들로서 지칭될 수도 있다. 픽처의 인트라-코딩된 (I) 슬라이스에서 비디오 블록들은 동일한 픽처에서 이웃하는 블록들에서의 참조 샘플들에 대한 공간 예측을 이용하여 인코딩된다. 화상의 인터-코딩된 (P 또는 B) 슬라이스에서 비디오 블록들은 동일한 픽처에서 이웃하는 블록들에서의 참조 샘플들에 대한 공간 예측, 또는 다른 참조 픽처들에서의 참조 샘플들에 대한 시간 예측을 이용할 수도 있다. 픽처들은 프레임들로 지칭될 수 있으며, 참조 픽처들은 참조 프레임들로서 지칭될 수도 있다.

[0006] 공간 또는 시간 예측은 코딩될 블록에 대한 예측 블록을 초래한다. 잔차 데이터는 코딩될 원래 블록과 예측 블록 사이의 픽셀 차이들을 나타낸다. 인터-코딩된 블록은 예측 블록을 형성하는 참조 샘플들의 블록을 가리키는 모션 벡터, 및 코딩된 블록과 예측 블록 사이의 차이를 나타내는 잔차 데이터에 따라서 인코딩된다. 인트라-코딩된 블록은 인트라-코딩 모드 및 잔차 데이터에 따라서 인코딩된다. 추가적인 압축을 위해, 잔차 데이터는 픽셀 도메인으로부터 변환 도메인으로 변환되어, 잔차 변환 계수들을 초래할 수도 있으며, 이 잔차 변환 계수는 그후 양자화될 수도 있다. 처음에 2차원 어레이로 배열된, 양자화된 변환 계수들은 변환 계수들의 1차원 벡터를 발생하기 위해 스캐닝될 수도 있으며, 더욱 더 많은 압축을 달성하기 위해 엔트로피 코딩이 적용될 수도 있다.

발명의 내용

과제의 해결 수단

[0007] 일반적으로, 본 개시물은 적응 루프 필터링 (ALF) 과 같은, 필터링에 관련된 기법들을 기술한다. 특히, 비디오 데이터의 블록들의 상이한 클래스들에 대한 필터들을 발생시키는 다양한 기법들이 비디오 인코더 또는 비디오 디코더와 같은, 비디오 코더에 의해 수행될 수도 있다. 일 예에서, 비디오 코더는 필터 계수들의 다수의 세트들을 구성하고 블록들의 다양한 클래스들에 대한 필터 계수들의 세트들로의 인덱스들을 코딩할 수도 있으며, 여기서, 각각의 인덱스는 블록들의 대응하는 클래스에 대한 필터 계수들의 세트를 식별한다. 다른 예에서, 비디오 코더는 이전에 발생된 필터의 필터 계수들의 세트를 이용하고 (회전, 수직 플립, 또는 대각선 플립과 같은) 기하학적 변환을 필터 서포트 영역 또는 필터 계수들 자체에 적용함으로써 블록들의 클래스에 대한 필터를 발생시킬 수도 있다. 또한 다른 예에서, 비디오 코더는 고정 필터 또는 이전에 코딩된 픽처의 필터로부터 블록들의 클래스에 대한 필터 계수들을 예측할 수도 있다.

[0008] 일 예에서, 비디오 데이터의 디코딩된 블록을 필터링하는 방법은, 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하는 단계; 현재의 블록의 픽셀들을 필터링하는데 사용될 (예컨대, 적응 루프 필터링 (ALF) 에 따른) 필터를 선택하는 단계; 선택된 필터의 필터 서포트 영역 또는 계수들 중 하나에 대해 수행될 기하학적 변환을 선택하는 단계; 선택된 필터의 필터 서포트 영역 또는 계수들에 대해 기하학적 변환을 수행하는 단계; 및 기하학적 변환을 수행한 후, 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 단계를 포함한다.

[0009] 다른 예에서, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스는, 비디오 데이터를 저장하도록 구성된 메모리; 및 회로부 내에 구현되는 하나 이상의 프로세서들을 포함하며, 상기 하나 이상의 프로세서들은, 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하고, 현재의 블록의 픽셀들을 필터링하는데 사용될 (예컨대, 적응 루프 필터링 (ALF) 에 따른) 필터를 선택하고, 선택된 필터의 필터 서포트 영역 또는 계수들 중 하나에 대해 수행될 기하학적 변환을 선택하고, 그리고 기하학적 변환을 수행한 후, 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하도록 구성된다.

[0010] 다른 예에서, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스는, 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하는 수단; 현재의 블록의 픽셀들을 필터링하는데 사용될 (예컨대, 적응 루프 필터링 (ALF) 에 따른) 필터를 선택하는 수단; 선택된 필터의 필터 서포트 영역 또는 계수들 중 하나에 대해 수행될 기하학적 변환을 선택하는 수단; 선택된 필터의 필터 서포트 영역 또는 계수들에 대해 기하학적 변환을 수행하는 수단; 및 기하학적 변환을 수행한 후, 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 수단을 포함한다.

[0011] 다른 예에서, (비일시성 컴퓨터-판독가능 저장 매체와 같은) 컴퓨터-판독가능 저장 매체는, 명령들을 안에 저장하고 있으며, 상기 명령들은, 실행될 때, 프로세서로 하여금, 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하게 하고, 현재의 블록의 픽셀들을 필터링하는데 사용될 (예컨대, 적응 루프 필터링 (ALF) 에 따른) 필터를 선택하게 하고, 선택된 필터의 필터 서포트 영역 또는 계수들 중 하나에 대해 수행될 기하학적 변환을 선택하게 하고, 선택된 필터의 필터 서포트 영역 또는 계수들에 대해 기하학적 변환을 수행하게 하고, 그리고 기하학적 변환을 수행한 후, 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하게 한다.

[0012] 다른 예에서, 비디오 데이터의 디코딩된 블록을 필터링하는 방법은, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 단계로서, 상기 복수의 필터들을 구성하는 단계는 클래스들의 각각에 대해, 고정 필터가 클래스의 필터 계수들의 세트를 예측하는데 사용되는지 여부를 결정하는 단계, 및 고정 필터가 필터 계수들의 세트를 예측하는데 사용된다고 결정하는 것에 응답하여, 클래스의 고정 필터들의 세트로의 인덱스 값을 결정하는 단계 및 인덱스 값에 의해 식별된 고정 필터들의 세트의 고정 필터를 이용하여 클래스의 필터 계수들의 세트를 예측하는 단계를 포함하는, 상기 복수의 필터들을 구성하는 단계; 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하는 단계; 현재의 블록에 대한 클래스를 결정하는 단계; 현재의 블록에 대한 클래스에 대응하는 복수의 필터들의 필터를 선택하는 단계; 및 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 단계를 포함한다.

[0013] 다른 예에서, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스는, 비디오 데이터를 저장하도록 구성된 메모리; 및 회로부 내에 구현되는 하나 이상의 프로세싱 유닛들을 포함하며, 상기 하나 이상의 프로세싱 유닛들은, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 것으로서, 클래스들의 각각에 대한 복수의 필터들을 구성하기 위해, 하나 이상의 프로세싱 유닛들은 고정 필터가 클래스의 필터 계수들의 세트를 예측하는데 사용되는지 여부를 결정하고, 그리고 고정 필터가 필터 계수들의 세트를 예측하는데 사용된다고 결정하는 것에 응답하여, 클래스의 고정 필터들의 세트로의 인덱스 값을 결정하고 인덱스 값에 의해 식별된 고정 필터들의 세트의 고정 필터를 이용하여 클래스의 필터 계수들의 세트를 예측하도록 구성되는, 상기 복수의 필터들을 구성하고; 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하고; 현재의 블록에 대한 클래스를 결정하고; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하고; 그리고 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하도록 구성된다.

[0014] 다른 예에서, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스는, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 수단을 포함하며, 상기 복수의 필터들을 구성하는 수단은, 고정 필터가 클래스의 각각의 필터 계수들의 세트를 예측하는데 사용되는지 여부를 결정하는 수단; 및 고정 필터가 필터 계수들의 세트를 예측하는데 사용된다고 결정하는 것에 응답하여, 클래스의 고정 필터들의 세트로의 인덱스 값을 결정하는 수단 및 인덱스 값에 의해 식별된 고정 필터들의 세트의 고정 필터를 이용하여 클래스의 필터

계수들의 세트를 예측하는 수단을 포함한다. 디바이스는 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하는 수단; 현재의 블록에 대한 클래스를 결정하는 수단; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하는 수단; 및 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 수단을 더 포함한다.

[0015] 다른 예에서, (비일시성 컴퓨터-판독가능 저장 매체와 같은) 컴퓨터-판독가능 저장 매체는, 실행될 때, 프로세서로 하여금, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하게 하고; 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하게 하고; 현재의 블록에 대한 클래스를 결정하게 하고; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하게 하고; 그리고 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하게 하는 명령들을 저장하며, 상기 프로세서로 하여금, 복수의 필터들을 구성하게 하는 명령들은, 프로세서로 하여금, 클래스들의 각각에 대해, 고정 필터가 클래스의 필터 계수들의 세트를 예측하는데 사용되는지 여부를 결정하게 하고, 그리고 고정 필터가 필터 계수들의 세트를 예측하는데 사용된다고 결정하는 것에 응답하여, 클래스의 고정 필터들의 세트로의 인덱스 값을 결정하게 하고 인덱스 값에 의해 식별된 고정 필터들의 세트의 고정 필터를 이용하여 클래스의 필터 계수들의 세트를 예측하게 하는 명령들을 포함한다.

[0016] 다른 예에서, 비디오 데이터의 디코딩된 블록을 필터링하는 방법은, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 단계로서, 상기 복수의 필터들을 구성하는 단계는 발생시킬 필터 계수들의 세트들의 수를 결정하는 단계, 상기 수의 필터 계수들의 세트들을 발생시키는 단계, 및 복수의 필터들의 서브세트에 대해, 서브세트의 대응하는 필터에 대한 필터 계수들의 세트들 중 하나를 식별하는 개별 인덱스들을 결정하는 단계를 포함하는, 상기 복수의 필터들을 구성하는 단계; 현재의 픽처의 현재의 블록을 디코딩하는 단계; 현재의 블록에 대한 클래스를 결정하는 단계; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하는 단계; 및 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 단계를 포함한다.

[0017] 다른 예에서, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스는, 비디오 데이터를 저장하도록 구성된 메모리; 및 회로부 내에 구현되는 하나 이상의 프로세싱 유닛들을 포함하며, 상기 하나 이상의 프로세싱 유닛들은, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 것으로서, 복수의 필터들을 구성하기 위해, 하나 이상의 프로세싱 유닛들은 발생시킬 필터 계수들의 세트들의 수를 결정하고, 상기 수의 필터 계수들의 세트들을 발생시키고, 그리고 복수의 필터들의 서브세트에 대해, 서브세트의 대응하는 필터에 대한 필터 계수들의 세트들 중 하나를 식별하는 개별 인덱스들을 결정하도록 구성되는, 상기 복수의 필터들을 구성하고; 현재의 픽처의 현재의 블록을 디코딩하고; 현재의 블록에 대한 클래스를 결정하고; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하고; 그리고 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하도록 구성된다.

[0018] 다른 예에서, 비디오 데이터의 디코딩된 블록을 필터링하는 디바이스는, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 수단을 포함하며, 상기 복수의 필터들을 구성하는 수단은 발생시킬 필터 계수들의 세트들의 수를 결정하는 수단; 상기 수의 필터 계수들의 세트들을 발생시키는 수단; 및 복수의 필터들의 서브세트에 대해, 서브세트의 대응하는 필터에 대한 필터 계수들의 세트들 중 하나를 식별하는 개별 인덱스들을 결정하는 수단을 포함하며, 상기 디바이스는 현재의 픽처의 현재의 블록을 디코딩하는 수단; 현재의 블록에 대한 클래스를 결정하는 수단; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하는 수단; 및 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 수단을 더 포함한다.

[0019] 다른 예에서, 컴퓨터-판독가능 저장 매체는, 실행될 때, 프로세서로 하여금, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하게 하고; 현재의 픽처의 현재의 블록을 디코딩하게 하고; 현재의 블록에 대한 클래스를 결정하게 하고; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하게 하고; 그리고 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하게 하는 명령들을 저장하며, 상기 프로세서로 하여금, 복수의 필터들을 구성하게 하는 명령들은, 프로세서로 하여금, 필터 계수들의 복수의 세트들을 발생시키게 하고; 그리고 복수의 필터들의 서브세트에 대해, 서브세트의 대응하는 필터에 대한 필터 계수들의 세트들 중 하나를 식별하는 개별 인덱스들을 결정하게 하는 명령들을 포함한다.

[0020] 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

[0021] 도 1 은 본 개시물에서 설명하는 기법들을 이용할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.

도 2 는 필터들에의, 활동도 메트릭 및 방향 메트릭에 대한 범위들의 맵핑을 예시하는 개념도이다.

도 3 은 필터 계수 차이들을 시그널링하는 예시적인 패턴을 예시하는 개념도이다.

도 4 는 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 비디오 인코더를 예시하는 블록도이다.

도 5 는 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 비디오 디코더를 예시하는 블록도이다.

도 6 은 비디오 인코더에 의해 (예컨대, 비디오 인코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다.

도 7 은 비디오 디코더에 의해 (예컨대, 비디오 디코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다.

도 8 은 비디오 인코더에 의해 (예컨대, 비디오 인코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다.

도 9 는 비디오 디코더에 의해 (예컨대, 비디오 디코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다.

도 10 은 비디오 인코더에 의해 (예컨대, 비디오 인코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다.

도 11 은 비디오 디코더에 의해 (예컨대, 비디오 디코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다.

발명을 실시하기 위한 구체적인 내용

[0022] 전형적인 비디오 인코더에서, 원래 비디오 시퀀스의 프레임은 인트라-모드 (I-모드) 또는 인터-모드 (P-모드) 에서 인코딩된 직사각형의 영역들 또는 블록들로 파티셔닝된다. 블록들은 DCT 코딩과 같은, 일종의 변환 코딩을 이용하여 코딩된다. 그러나, 순수 변환-기반의 코딩은 단지 픽셀들의 블록 간 상관을 고려함이 없이 특정의 블록 내 픽셀 간 상관을 감소시키며, 여전히 송신을 위해 높은 비트-레이트들을 발생시킨다. 현재의 디지털 이미지 코딩 표준들은 또한 블록들 사이의 픽셀 값들의 상관을 감소시키는 어떤 방법들을 이용한다. 이하의 설명에서 명확해지는 바와 같이, 본 개시물에서, 용어 비디오 코딩은 비디오 인코딩 또는 비디오 디코딩을 포괄적으로 지칭하기 위해 사용될 수도 있다.

[0023] 일반적으로, P-모드에서 인코딩된 블록들은 이전에 코딩된 및 송신된 프레임들 중 하나로부터 예측된다. 블록의 예측 정보는 2차원 (2D) 모션 벡터에 의해 표현된다. I-모드에서 인코딩되는 블록들에 대해, 예측된 블록은 동일한 프레임 내에서 이미 인코딩된 이웃하는 블록들로부터 공간 예측을 이용하여 형성된다. 예측 에러, 즉, 인코딩중인 블록과 예측된 블록 사이의 차이는, 일부 별개의 변환의 가중된 기저 함수들의 세트로서 표현된다. 변환은 일반적으로 8x8 또는 4x4 블록 단위로 수행된다. 가중치들이 - 변환 계수들이 - 그후 양자화된다. 양자화는 정보의 손실을 도입하며, 따라서, 양자화된 계수들은 원본들보다 낮은 정밀도를 갖는다.

[0024] 양자화된 변환 계수들은, 모션 벡터들 및 일부 제어 정보와 함께, 완전한 코딩된 시퀀스 표현을 형성하며, 신택스 엘리먼트들로서 지칭된다. 인코더로부터 디코더로의 송신 전에, 신택스 엘리먼트들은 신택스 엘리먼트들을 표현하는데 사용되는 비트수를 더욱 감소시키기 위해 엔트로피 인코딩된다.

[0025] 디코더에서, 현재의 프레임에서의 블록은 그의 예측을 인코더에서와 동일한 방법으로 먼저 구성하고 그 예측에 압축된 예측 에러를 가산함으로써, 얻어진다. 압축된 예측 에러는 양자화된 계수들을 이용하여 변환 기저 함수들을 가중함으로써 발견된다. 복원된 프레임과 원래 프레임 사이의 차이는 복원 에러로 불린다.

[0026] 디코딩된 비디오의 품질을 더욱 향상시키기 위해, 비디오 디코더는 복원된 비디오 블록들에 대해 하나 이상의 필터링 동작들을 수행할 수 있다. 이들 필터링 동작들의 예들은 디블록킹 필터링, 샘플 적응 오프셋 (SAO) 필터링, 및 적응 루프 필터링 (ALF) 을 포함한다. 이들 필터링 동작들에 대한 파라미터들은 비디오 인코더

에 의해 결정되어 인코딩된 비디오 비트스트림으로 명시적으로 시그널링될 수도 있거나 또는 비디오 디코더에 의해 내재적으로 결정될 수도 있다.

[0027] 본 개시물은 비디오 인코딩 및/또는 비디오 디코딩 프로세스들에서 복원된 비디오 데이터를 필터링하는 것과 연관된 기법들을 기술하며, 더 자세하게는, 본 개시물은 ALF 에 관련된 기법들을 기술한다. 본 개시물에 따르면, 필터링이 인코더에서 적용되며, 필터 정보는 디코더가 인코더에서 적용된 필터링을 식별가능하게 하기 위해서, 비트스트림으로 인코딩된다. 비디오 인코더는 여러 상이한 필터링 시나리오들을 테스트하고, 레이트-왜곡 분석에 기초하여, 복원된 비디오 품질과 압축 품질 사이의 원하는 상충관계를 발생시키는 필터 또는 필터들의 세트를 선택할 수도 있다. 비디오 디코더는 필터 정보를 포함하는 인코딩된 비디오 데이터를 수신하고, 비디오 데이터를 디코딩하고, 그리고, 필터링 정보에 기초하여 필터링을 적용한다. 이러한 방법으로, 비디오 디코더는 비디오 인코더에서 적용된 동일한 필터링을 적용한다.

[0028] 본 개시물에서, 용어 "필터" 는 일반적으로 필터 계수들의 세트를 지칭한다. 예를 들어, 3x3 필터는 9 개의 필터 계수들의 세트로 정의될 수도 있으며, 5x5 필터는 25 개의 필터 계수들의 세트로 정의될 수도 있으며, 9x5 필터는 45 개의 필터 계수들의 세트로 정의될 수도 있으며, 기타 등등으로 정의될 수도 있다. 용어 "필터들의 세트" 는 일반적으로 하나 보다 많은 필터의 그룹을 지칭한다. 예를 들어, 2개의 3x3 필터들의 세트는 9 개의 필터 계수들의 제 1 세트 및 9 개의 필터 계수들의 제 2 세트를 포함할 수 있다. "필터 서포트" 로서 종종 불리는, 용어 "형상 (shape)" 은 일반적으로 특정의 필터에 대한 필터 계수들의 로우들의 개수 및 필터 계수들의 칼럼들의 개수를 지칭한다. 예를 들어, 9x9 는 제 1 형상의 일 예이며, 7x5 는 제 2 형상의 일 예이며, 5x9 는 제 3 형상의 일 예이다. 일부의 경우, 필터들은 다이아몬드-형상들, 다이아몬드-형 형상들, 원형 형상들, 원형-형 형상들, 6각형 형상들, 8각형 형상들, 십자 형상들, X-형상들, T-형상들, 다른 기하학적 형상들, 또는 매우 많은 다른 형상들 또는 구성을 포함한, 비-직사각형의 형상들을 취할 수도 있다. 이 경우, 비-직사각형의 필터 서포트들에 대해, 필터 계수들의 개수는 상이할 수도 있다. 예를 들어, 9x9 다이아몬드 필터 서포트에 대해, 9x9 다이아몬드 필터는 (9x9/4+1) 개의 필터 계수들의 세트로 정의될 수도 있다.

[0029] 도 1 은 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 비디오 인코딩 및 디코딩 시스템 (10) 을 예시하는 블록도이다. 도 1 에 나타낸 바와 같이, 시스템 (10) 은 목적지 디바이스 (14) 에 의해 추후 디코딩될 인코딩된 비디오 데이터를 발생시키는 소스 디바이스 (12) 를 포함한다. 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 태블릿 컴퓨터들, 셋-탑 박스들, 소위 "스마트" 폰들과 같은 전화기 핸드셋들, 소위 "스마트" 패드들, 텔레비전, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게이밍 콘솔들, 비디오 스트리밍 디바이스 등을 포함한, 광범위한 디바이스들 중 임의의 디바이스를 포함할 수도 있다. 일부의 경우, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 무선 통신용으로 탑재될 수도 있다.

[0030] 목적지 디바이스 (14) 는 디코딩될 인코딩된 비디오 데이터를 링크 (16) 를 통해서 수신할 수도 있다. 링크 (16) 는 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 이동시킬 수 있는 임의 종류의 매체 또는 디바이스를 포함할 수도 있다. 일 예에서, 링크 (16) 는 소스 디바이스 (12) 로 하여금, 인코딩된 비디오 데이터를 직접 목적지 디바이스 (14) 로 실시간으로 송신할 수 있게 하는 통신 매체를 포함할 수도 있다. 인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라서 변조되어 목적지 디바이스 (14) 로 송신될 수도 있다. 통신 매체는 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적인 송신 라인들과 같은, 임의의 무선 또는 유선 통신 매체를 포함할 수도 있다. 통신 매체는 근거리 네트워크, 광역 네트워크, 또는 글로벌 네트워크, 예컨대 인터넷과 같은 패킷-기반 네트워크의 일부를 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0031] 대안적으로, 인코딩된 데이터는 출력 인터페이스 (22) 로부터 저장 디바이스 (26) 로 출력될 수도 있다. 이와 유사하게, 인코딩된 데이터는 입력 인터페이스에 의해 저장 디바이스 (26) 로부터 액세스될 수도 있다. 저장 디바이스 (26) 는 하드 드라이브, 블루-레이 디스크들, DVD들, CD-ROM들, 플래시 메모리, 휘발성 또는 비-휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체들과 같은 다양한 분산된 또는 로컬 액세스되는 데이터 저장 매체들 중 임의의 데이터 저장 매체를 포함할 수도 있다. 추가 예에서, 저장 디바이스 (26) 는 소스 디바이스 (12) 에 의해 발생된 인코딩된 비디오를 유지할 수도 있는 파일 서버 또는 또 다른 중간 저장 디바이스에 대응할 수도 있다. 목적지 디바이스 (14) 는 저장된 비디오 데이터에 저장 디바이스 (26) 로부터 스트리밍 또는 다운로드를 통해서 액세스할 수도 있다. 파일 서버는 인코딩된 비디오 데이터를 저장하고 그 인코딩된 비디오 데이터를 목적지 디바이스 (14) 로 송신하는 것이 가능

한 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 (14) 는 인터넷 접속을 포함한, 임의의 표준 데이터 접속을 통해서, 인코딩된 비디오 데이터에 액세스할 수도 있다.

이것은 파일 서버 상에 저장되는 인코딩된 비디오 데이터에 액세스하는데 적합한, 무선 채널 (예컨대, Wi-Fi 접속), 유선 접속 (예컨대, DSL, 케이블 모뎀 등), 또는 양쪽의 조합을 포함할 수도 있다. 저장 디바이스 (26) 로부터의 인코딩된 비디오 데이터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이 양쪽의 조합일 수도 있다.

[0032] 본 개시물의 기법들은 무선 애플리케이션들 또는 환경들에 반드시 한정되지는 않는다. 이 기법들은 다양한 멀티미디어 애플리케이션들 중 임의의 애플리케이션, 예컨대, 오버-디-에어 텔레비전 브로드캐스트들, 케이블 텔레비전 송신들, 위성 텔레비전 송신들, 예컨대, 인터넷을 통한 스트리밍 비디오 송신들, 데이터 저장 매체 상에의 저장을 위한 디지털 비디오의 인코딩, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션들의 지원 하에 비디오 코딩에 적용될 수도 있다. 일부 예들에서, 시스템 (10) 은 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 및/또는 비디오 전화 통신과 같은, 지원 애플리케이션들로의 단방향 또는 양방향 비디오 송신을 지원하도록 구성될 수도 있다.

[0033] 도 1 의 예에서, 소스 디바이스 (12) 는 비디오 소스 (18), 비디오 인코더 (20) 및 출력 인터페이스 (22) 를 포함한다. 일부의 경우, 출력 인터페이스 (22) 는 변조기/복조기 (모뎀) 및/또는 송신기를 포함할 수도 있다.

소스 디바이스 (12) 에서, 비디오 소스 (18) 는 비디오 캡쳐 디바이스, 예컨대, 비디오 카메라, 이전에 캡쳐된 비디오를 포함하는 비디오 아카이브, 비디오 콘텐츠 제공자로부터 비디오를 수신하는 비디오 공급 인터페이스, 및/또는 컴퓨터 그래픽스 데이터를 소스 비디오로서 발생시키는 컴퓨터 그래픽스 시스템과 같은 소스, 또는 이런 소스들의 조합을 포함할 수도 있다. 일 예로서, 비디오 소스 (18) 가 비디오 카메라이면, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 소위 카메라 폰들 또는 비디오 폰들일 수도 있다. 그러나, 본 개시물에서 설명하는 기법들은 비디오 코딩에 일반적으로 적용가능할 수도 있으며, 무선 및/또는 유선 애플리케이션들에 적용될 수도 있다.

[0034] 캡쳐되거나, 사전-캡쳐되거나, 또는 컴퓨터-발생된 비디오는 비디오 인코더 (20) 에 의해 인코딩될 수도 있다.

인코딩된 비디오 데이터는 소스 디바이스 (12) 의 출력 인터페이스 (22) 를 통해서 목적지 디바이스 (14) 로 바로 송신될 수도 있다. 인코딩된 비디오 데이터는 또한 (또는, 대안적으로) 디코딩 및/또는 플레이백을 위한, 목적지 디바이스 (14) 또는 다른 디바이스들에 의한 추후 액세스를 위해, 저장 디바이스 (26) 상에 저장될 수도 있다.

[0035] 목적지 디바이스 (14) 는 입력 인터페이스 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32) 를 포함한다. 일부의 경우, 입력 인터페이스 (28) 는 수신기 및/또는 모뎀을 포함할 수도 있다. 목적지 디바이스 (14) 의 입력 인터페이스 (28) 는 인코딩된 비디오 데이터를 링크 (16) 를 통해서 수신한다. 링크 (16) 를 통해서 통신되거나, 또는 저장 디바이스 (26) 상에 제공되는 인코딩된 비디오 데이터는, 비디오 데이터를 디코딩할 때에, 비디오 디코더 (30) 와 같은 비디오 디코더에 의해 사용하기 위한, 비디오 인코더 (20) 에 의해 발생되는 다양한 신택스 엘리먼트들을 포함할 수도 있다. 이런 신택스 엘리먼트들은 통신 매체 상으로 송신되거나, 저장 매체 상에 저장되거나, 또는 파일 서버에 저장된 인코딩된 비디오 데이터와 함께 포함될 수도 있다.

[0036] 디스플레이 디바이스 (32) 는 목적지 디바이스 (14) 와 통합되거나 또는 그 외부에 있을 수도 있다. 일부 예들에서, 목적지 디바이스 (14) 는 통합된 디스플레이 디바이스를 포함하며, 또한 외부 디스플레이 디바이스와 인터페이스하도록 구성될 수도 있다. 다른 예들에서, 목적지 디바이스 (14) 는 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32) 는 그 디코딩된 비디오 데이터를 사용자에게 디스플레이하며, 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 또 다른 유형의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 디바이스를 포함할 수도 있다.

[0037] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 최근 완결된 HEVC (High Efficiency Video Coding) 표준과 같은 비디오 압축 표준에 따라서 동작할 수도 있으며, HEVC 테스트 모델 (HM) 을 따를 수도 있다. 대안적으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 MPEG 4, 파트 10, AVC (Advanced Video Coding) 로서 대안적으로 지칭되는 ITU-T H.264 표준, 또는 이런 표준들의 확장판들과 같은 다른 사유 (proprietary) 또는 산업 표준들에 따라서 동작할 수도 있다. 그러나, 본 개시물의 기법들은 임의의 특정의 코딩 표준에 한정되지 않는다. 비디오 압축 표준들의 다른 예들은 MPEG-2 및 ITU-T H.263 을 포함한다.

[0038] 본 개시물의 기법들은 설명의 용이성을 위해 HEVC 전문용어를 이용할 수도 있다. 그러나, 본 개시물의 기법

들은 HEVC 에 한정되는 것으로 가정되어서는 안되며, 실제로, 본 개시물의 기법들은 HEVC 및 그의 확장판들에 대한 후속 표준들에서 구현될 수도 있는 것으로 명시적으로 고려된다.

[0039] 도 1 에 나타내지는 않지만, 일부 양태들에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 오디오 인코더 및 디코더와 각각 통합될 수도 있으며, 오디오 및 비디오 양쪽의 인코딩을 공통 데이터 스트림 또는 별개의 데이터 스트림들로 처리하기에 적합한 MUX-DEMUX 유닛들, 또는 다른 하드웨어 및 소프트웨어를 포함할 수도 있다. 적용가능한 경우, 일부 예들에서, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜, 또는 다른 프로토콜들, 예컨대, 사용자 데이터그램 프로토콜 (UDP) 을 따를 수도 있다.

[0040] 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSPs), 주문형 집적회로들 (ASICs), 필드 프로그래밍가능 게이트 어레이들 (FPGAs), 이산 로직, 소프트웨어, 하드웨어, 펌웨어 또는 임의의 이들의 조합들과 같은, 다양한 적합한 인코더 회로부 중 임의의 회로부로 구현될 수도 있다. 이 기법들이 소프트웨어로 부분적으로 구현되는 경우, 디바이스는 본 개시물의 기법들을 수행하기 위해 소프트웨어용 명령들을 적합한 비일시성 컴퓨터-판독가능 매체에 저장하고, 그 명령들을 하드웨어에서 하나 이상의 프로세서들을 이용하여 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 인코더들 또는 디코더들에 포함될 수도 있으며, 이들 중 어느 쪽이든 개별 디바이스에서 결합된 인코더/디코더 (CODEC) 의 일부분으로서 통합될 수도 있다.

[0041] 위에서 소개한 바와 같이, JCT-VC 는 HEVC 표준의 개발을 최근에 완료하였다. HEVC 표준화 노력들은 HEVC 테스트 모델 (HM) 으로서 지칭되는 비디오 코딩 디바이스의 진화 모델 (evolving model) 에 기초하였다. HM 은 예컨대, ITU-T H.264/AVC 에 따른 기존 디바이스들에 관련된 비디오 코딩 디바이스들의 여러 추가적인 능력들을 가정한다. 예를 들어, H.264 는 9개의 인트라-예측 인코딩 모드들을 제공하는 반면, HM 은 35개 만큼이나 많은 인트라-예측 인코딩 모드들을 제공할 수도 있다.

[0042] HEVC 및 다른 비디오 코딩 사양들에서, 비디오 시퀀스는 일반적으로 일련의 픽처들을 포함한다. 픽처들은 또한 "프레임들" 로서 지칭될 수도 있다. 픽처는 S_L, S_{Cb}, 및 S_{Cr} 로 표기되는, 3개의 샘플 어레이들을 포함할 수도 있다. S_L 은 루마 샘플들의 2차원 어레이 (즉, 블록) 이다. S_{Cb} 는 Cb 색차 샘플들의 2차원 어레이이다. S_{Cr} 은 Cr 색차 샘플들의 2차원 어레이이다. 색차 샘플들은 또한 본원에서 "크로마" 샘플들로서 지칭될 수도 있다. 다른 경우, 픽처는 단색일 수도 있으며 단지 루마 샘플들의 어레이 만을 포함할 수도 있다.

[0043] 픽처의 인코딩된 표현을 발생하기 위해, 비디오 인코더 (20) 는 코딩 트리 유닛들 (CTUs) 의 세트를 발생시킬 수도 있다. CTU들의 각각은 루마 샘플들의 코딩 트리 블록, 크로마 샘플들의 2개의 대응하는 코딩 트리 블록들, 및 코딩 트리 블록들의 샘플들을 코딩하는데 사용되는 신택스 구조들을 포함할 수도 있다. 단색 픽처들 또는 3개의 별개의 칼라 평면들을 가지는 픽처들에서, CTU 는 단일 코딩 트리 블록 및 코딩 트리 블록의 샘플들을 코딩하는데 사용되는 신택스 구조들을 포함할 수도 있다. 코딩 트리 블록은 샘플들의 NxN 블록일 수도 있다. CTU 는 또한 "트리 블록" 또는 "최대 코딩 유닛" (LCU) 으로서 지칭될 수도 있다. HEVC 의 CTU들은 H.264/AVC 와 같은, 다른 표준들의 매크로블록들과 대략 유사할 수도 있다. 그러나, CTU 는 특정의 사이즈에 반드시 제한되지 않으며, 하나 이상의 코딩 유닛들 (CUs) 을 포함할 수도 있다. 슬라이스는 래스터 스캔 순서로 연속적으로 순서화된 정수의 CTU들을 포함할 수도 있다.

[0044] 코딩된 CTU 를 발생시키기 위해, 비디오 인코더 (20) 는 코딩 트리 블록들을 코딩 블록들, 따라서 이름 "코딩 트리 유닛들" 로 분할하기 위해 CTU 의 코딩 트리 블록들에 대해 쿼드-트리 파티셔닝을 회귀적으로 수행할 수도 있다. 코딩 블록은 샘플들의 NxN 블록일 수도 있다. CU 는 루마 샘플들의 코딩 블록 및 루마 샘플 어레이, Cb 샘플 어레이, 및 Cr 샘플 어레이를 가지는 픽처의 크로마 샘플들의 2개의 대응하는 코딩 블록들, 및 코딩 블록들의 샘플들을 코딩하는데 사용되는 신택스 구조들을 포함할 수도 있다. 단색 픽처들 또는 3개의 별개의 칼라 평면들을 가지는 픽처들에서, CU 는 단일 코딩 블록 및 코딩 블록의 샘플들을 코딩하는데 사용되는 신택스 구조들을 포함할 수도 있다.

[0045] 비디오 인코더 (20) 는 CU 의 코딩 블록을 하나 이상의 예측 블록들로 파티셔닝할 수도 있다. 예측 블록은 동일한 예측이 적용되는 샘플들의 직사각형 (즉, 정사각형 또는 비-정사각형) 블록이다. CU 의 예측 유닛 (PU) 는 루마 샘플들의 예측 블록, 크로마 샘플들의 2개의 대응하는 예측 블록들, 및 예측 블록들을 예측하는데 사용되는 신택스 구조들을 포함할 수도 있다. 단색 픽처들 또는 3개의 별개의 칼라 플레인들을 가지는 픽처들에서, PU 는 단일 예측 블록 및 예측 블록을 예측하는데 사용되는 신택스 구조들을 포함할 수도 있다. 비

디오 인코더 (20) 는 CU 의 각각의 PU 의 루마, Cb, 및 Cr 예측 블록들에 대한, 예측 루마, Cb, 및 Cr 블록들을 발생시킬 수도 있다.

[0046] 비디오 인코더 (20) 는 PU 에 대한 예측 블록들을 발생시키기 위해 인트라 예측 또는 인터 예측을 이용할 수도 있다. 비디오 인코더 (20) 가 PU 의 예측 블록들을 발생시키기 위해 인트라 예측을 이용하면, 비디오 인코더 (20) 는 PU 와 연관되는 픽처의 디코딩된 샘플들에 기초하여 PU 의 예측 블록들을 발생시킬 수도 있다. 비디오 인코더 (20) 가 PU 의 예측 블록들을 발생하기 위해 인터 예측을 이용하면, 비디오 인코더 (20) 는 PU 와 연관되는 픽처 이외의 하나 이상의 픽처들의 디코딩된 샘플들에 기초하여, PU 의 예측 블록들을 발생시킬 수도 있다.

[0047] 비디오 인코더 (20) 가 CU 의 하나 이상의 PU들에 대한 예측 루마, Cb, 및 Cr 블록들을 발생한 후, 비디오 인코더 (20) 는 CU 에 대한 루마 잔차 블록을 발생시킬 수도 있다. CU 의 루마 잔차 블록에서의 각각의 샘플은 CU 의 예측 루마 블록들 중 하나에서의 루마 샘플과 CU 의 원래 루마 코딩 블록에서의 대응하는 샘플 사이의 차이를 나타낸다. 게다가, 비디오 인코더 (20) 는 CU 에 대한 Cb 잔차 블록을 발생시킬 수도 있다. CU 의 Cb 잔차 블록에서의 각각의 샘플은 CU 의 예측 Cb 블록들 중 하나에서의 Cb 샘플과 CU 의 원래 Cb 코딩 블록에서의 대응하는 샘플 사이의 차이를 나타낼 수도 있다. 비디오 인코더 (20) 는 또한 CU 에 대한 Cr 잔차 블록을 발생시킬 수도 있다. CU 의 Cr 잔차 블록에서의 각각의 샘플은 CU 의 예측 Cr 블록들 중 하나에서의 Cr 샘플과 CU 의 원래 Cr 코딩 블록에서의 대응하는 샘플 사이의 차이를 나타낼 수도 있다.

[0048] 더욱이, 비디오 인코더 (20) 는 쿼드-트리 파티셔닝을 이용하여, CU 의 루마, Cb, 및 Cr 잔차 블록들을 하나 이상의 루마, Cb, 및 Cr 변환 블록들로 분해할 수도 있다. 변환 블록은 동일한 변환이 적용되는 샘플들의 직사각형의 (예컨대, 정사각형 또는 비-정사각형) 블록이다. CU 의 변환 유닛 (TU) 는 루마 샘플들의 변환 블록, 크로마 샘플들의 2개의 대응하는 변환 블록들, 및 변환 블록 샘플들을 변환하는데 사용되는 신택스 구조들을 포함할 수도 있다. 따라서, CU 의 각각의 TU 는 루마 변환 블록, Cb 변환 블록, 및 Cr 변환 블록과 연관될 수도 있다. TU 와 연관되는 루마 변환 블록은 CU 의 루마 잔차 블록의 서브-블록일 수도 있다. Cb 변환 블록은 CU 의 Cb 잔차 블록의 서브-블록일 수도 있다. Cr 변환 블록은 CU 의 Cr 잔차 블록의 서브-블록일 수도 있다. 단색 픽처들 또는 3개의 별개의 칼라 평면들을 가지는 픽처들에서, TU 는 단일 변환 블록 및 변환 블록의 샘플들을 변환하는데 사용되는 신택스 구조들을 포함할 수도 있다.

[0049] 비디오 인코더 (20) 는 하나 이상의 변환들을 TU 의 루마 변환 블록에 적용하여, TU 에 대한 루마 계수 블록을 발생시킬 수도 있다. 계수 블록은 변환 계수들의 2차원 어레이일 수도 있다. 변환 계수는 스칼라 양일 수도 있다. 비디오 인코더 (20) 는 하나 이상의 변환들을 TU 의 Cb 변환 블록에 적용하여, TU 에 대한 Cb 계수 블록을 발생시킬 수도 있다. 비디오 인코더 (20) 는 하나 이상의 변환들을 TU 의 Cr 변환 블록에 적용하여, TU 에 대한 Cr 계수 블록을 발생시킬 수도 있다.

[0050] 계수 블록 (예컨대, 루마 계수 블록, Cb 계수 블록, 또는 Cr 계수 블록) 을 발생시킨 후, 비디오 인코더 (20) 는 계수 블록을 양자화할 수도 있다. 양자화는 일반적으로 변환 계수들이 변환 계수들을 나타내는데 사용되는 데이터의 양을 가능한 한 감소시키도록 양자화되어 추가적인 압축을 제공하는 프로세스를 지칭한다. 비디오 인코더 (20) 가 계수 블록을 양자화한 후, 비디오 인코더 (20) 는 양자화된 변환 계수들을 나타내는 신택스 엘리먼트들을 엔트로피 인코딩할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 양자화된 변환 계수들을 나타내는 신택스 엘리먼트들에 대해 컨텍스트-적응 2진 산술 코딩 (CABAC) 을 수행할 수도 있다.

[0051] 비디오 인코더 (20) 는 코딩된 픽처들 및 연관되는 데이터의 표현을 형성하는 비트들의 시퀀스를 포함하는 비트스트림을 출력할 수도 있다. 비트스트림은 일련의 NAL 유닛들을 포함할 수도 있다. NAL 유닛은 NAL 유닛에서의 데이터의 형태, 및 그 데이터를 RBSP 의 유형으로 에뮬레이션 방지 비트들과 필요에 따라 섞어서 포함하는 바이트들의 표시를 포함하는 신택스 구조이다. NAL 유닛들의 각각은 NAL 유닛 헤더를 포함하며 RBSP 를 캡슐화한다. NAL 유닛 헤더는 NAL 유닛 유형 코드를 나타내는 신택스 엘리먼트를 포함할 수도 있다. NAL 유닛의 NAL 유닛 헤더에 의해 규정된 NAL 유닛 유형 코드는 NAL 유닛의 형태를 나타낸다. RBSP 는 NAL 유닛 내에 캡슐화된 정수의 바이트들을 포함하는 신택스 구조일 수도 있다. 일부의 경우, RBSP 는 제로 비트들을 포함한다.

[0052] 상이한 유형들의 NAL 유닛들이 상이한 유형들의 RBSP들을 캡슐화할 수도 있다. 예를 들어, NAL 유닛의 제 1 유형은 PPS 에 대한 RBSP 를 캡슐화할 수도 있으며, NAL 유닛의 제 2 유형은 코딩된 슬라이스에 대한 RBSP 를 캡슐화할 수도 있으며, NAL 유닛의 제 3 유형은 SEI 메시지들에 대한 RBSP 를 캡슐화할 수도 있으며, 기타 등등을 캡슐화할 수도 있다. (파라미터 세트들 및 SEI 메시지들에 대한 RBSP들과는 반대로) 비디오 코딩 데이터

에 대한 RBSP들을 캡슐화하는 NAL 유닛들은 VCL NAL 유닛들로서 지칭될 수도 있다.

[0053] 비디오 디코더 (30) 는 비디오 인코더 (20) 에 의해 발생되는 비트스트림을 수신할 수도 있다. 게다가, 비디오 디코더 (30) 는 비트스트림을 파싱하여, 비트스트림으로부터 신택스 엘리먼트들을 획득할 수도 있다. 비디오 디코더 (30) 는 비트스트림으로부터 획득된 신택스 엘리먼트들에 적어도 부분적으로 기초하여 비디오 데이터의 픽처들을 복원할 수도 있다. 비디오 데이터를 복원하는 프로세스는 일반적으로 비디오 인코더 (20) 에 의해 수행되는 프로세스와 반대일 수도 있다. 게다가, 비디오 디코더 (30) 는 현재의 CU 의 TU들과 연관되는 계수 블록들을 역양자화할 수도 있다. 비디오 디코더 (30) 는 계수 블록들에 대해 역변환들을 수행하여, 현재의 CU 의 TU들과 연관되는 변환 블록들을 복원할 수도 있다. 비디오 디코더 (30) 는 현재의 CU 의 PU들에 대한 예측 블록들의 샘플들을 현재의 CU 의 TU들의 변환 블록들의 대응하는 샘플들에 가산함으로써, 현재의 CU 의 코딩 블록들을 복원할 수도 있다. 픽처의 각각의 CU 에 대해 코딩 블록들을 복원함으로써, 비디오 디코더 (30) 는 그 픽처를 복원할 수도 있다.

[0054] 비디오 코딩의 분야에서, 디코딩된 비디오 신호의 품질을 향상시키기 위해 필터링을 적용하는 것이 일반적이다. 필터는 사후-필터로서 적용될 수 있으며, 여기서, 필터링된 프레임은 미래 프레임들의 예측을 위해, 또는 인-루프 필터로서 사용되지 않으며, 여기서, 필터링된 프레임은 미래 프레임들을 예측하는데 사용될 수도 있다. 필터는 예를 들어, 원래 신호와 디코딩된 필터링된 신호 사이의 에러를 최소화함으로써 설계될 수 있다. 변환 계수들과 유사하게, 필터 h(k,l), k=-K,…,K,l=-K,…K 의 계수들은 $f(k,l) = round\big(normFactor \cdot h(k,l)\big)$ 로 양자화되고, 코딩되어, 디코더로 전송될 수도 있다.

[0055] normFactor 는 보통 2^n 과 동일하다. normFactor 의 값이 더 클수록, 더 정확하게 양자화되며, 이는 일반적으로 양자화된 필터 계수들을 초래하여 더 나은 성능을 제공한다. 한편, normFactor 의 더 큰 값들은 일반적으로 송신하는데 더 많은 비트들을 필요로 하는 계수들 f(k,l) 을 발생시킨다.

[0056] 디코더에서, 디코딩된 필터 계수들 f(k,l) 은 다음과 같이, 복원된 이미지 R(i,j) 에 적용되며,

[0057] $$\tilde{R}(i,j) = \sum_{k=-K}^{K} \sum_{l=-K}^{K} f(k,l) R(i+k, j+l) / \sum_{k=-k}^{K} \sum_{l=-K}^{K} f(k,l) \quad (1)$$

[0058] 여기서, i 및 j 는 프레임 내 픽셀들의 좌표들이다.

[0059] ALF 가 HEVC 에서 제안되었으며, ALF 가 HEVC 의 최종 버전에는 포함되지 않지만, 다양한 작업 초안들 및 테스트 모델 소프트웨어, 즉, HEVC 테스트 모델 (또는, "HM") 에 포함되었다. 관련된 기술들 중에서, HEVC 테스트 모델 버전 HM-3.0 에서의 필터 설계가 가장 효율적인 설계로서 일반적으로 간주되었다.

[0060] HM-3.0 에서의 ALF 는 픽처 레벨 최적화에 기초한다. 즉, 전체 프레임이 코딩된 후 필터 계수들이 유도된다. 루마 성분, 블록 기반의 적응 (BA) 및 영역 기반의 적응 (RA) 에 대해 2개의 모드들이 있었다. 이들 2개의 모드들은 신택스 엘리먼트들 뿐만 아니라, 동일한 필터 형상들 및 필터링 동작들을 공유한다. 단지 이들 사이의 차이는 분류 방법이다.

[0061] HM-3.0 에서의 블록 기반의 적응에 의한 적응 루프 필터링 (ALF) 은 인-루프 필터의 일 예이다. 루마 성분에 대한 ALF 에서, 전체 픽처에서의 4x4 블록들은 상이한 특성들을 가지는 이미지에서의 영역들에 대해 상이한 필터들이 있어야 한다는 가정 하에서 블록의 방향성 (최고 3 개의 방향들) 및 블록의 2D 라플라시안 활동도 (최고 5 개의 활동도 값들) 에 기초하여 분류될 수도 있다. 방향성 D 및 비양자화된 활동도 A 의 계산은 다음 수식 (2) 내지 수식 (5) 에 나타낸다:

[0062] $$g_v = \sum_{i=0}^{3} \sum_{j=0}^{3} V_{i,j}, \quad V_{i,j} = |2R(i,j) - R(i,j-1) - R(i,j+1)|, \quad (2)$$

[0063] $$g_h = \sum_{i=0}^{3} \sum_{j=0}^{3} H_{i,j}, \quad H_{i,j} = |2R(i,j) - R(i-1,j) - R(i+1,j)|, \quad (3)$$

[0064] $$D = \begin{cases} 1 & g_h > 2 * g_v \\ 2 & g_v > 2 * g_h, \\ 0 & \text{이외의 경우} \end{cases} \quad (4)$$

[0065] $$A = \sum_{i=0}^{3} \sum_{j=0}^{3} \left(\sum_{k=i-1}^{i+1} \sum_{l=j-1}^{j+1} (V_{k,l} + H_{k,l}) \right), \quad (5)$$

[0066] 여기서, R(i,j) 는 4x4 블록의 좌상부에 대한 상대 좌표 (i,j) 를 갖는 복원된 픽셀을 표시한다. A 는 HM-3.0 에서 설명된 바와 같은 0 내지 4 를 포함한 범위로 추가로 양자화되며, A 의 양자화된 값은 $\hat{A}$ 로서 표시될 것이다.

[0067] 전체로서, 각각의 블록은 15 (5x3) 개의 클래스들 중 하나로 분류될 수 있으며, 인덱스가 블록의 D 및 $\hat{A}$ 의 값에 따라서 각각의 4x4 블록에 할당된다: $5D + \hat{A}$. 따라서, 픽처의 루마 성분에 대해 ALF 파라미터들의 최고 15 개의 세트들이 시그널링될 수 있다. 시그널링 비용을 절약하기 위해, 클래스들은 클래스 인덱스 값에 따라서 병합될 수도 있다. 각각의 클래스 (또는, 병합된 클래스) 에 대해, 필터 계수들의 세트가 시그널링될 수도 있다. 추가적으로, HM-3.0 에서, ALF 가 CU 에 적용되는지 여부를 표시하기 위해 CU 레벨에서 플래그가 시그널링된다.

[0068] 본 개시물의 기법들에 따르면, (비디오 인코더 (20) 또는 비디오 디코더 (30) 와 같은) 비디오 코딩 디바이스가 디코딩된 비디오 데이터의 필터링을 수행할 수도 있다. 필터 계수들을 시그널링하는 단위가 픽처/프레임, 슬라이스, 타일 또는 등일 수도 있다는 점에 유의한다. 간결성을 위해, 다음 설명들은 단위를 픽처로서 주로 취급한다. 그러나, 본 개시물의 기법들은 픽처-레벨 ALF 시그널링에 한정되지 않는다. 아래에서 더 자세히 설명되는 바와 같이, 비디오 인코더 (20) 가 비디오 데이터의 픽처를 인코딩한 후, 비디오 인코더 (20) 는 그후 예컨대, 참조 픽처로서의 사용을 위해, 픽처를 디코딩할 수도 있다. 이와 같이, 비디오 인코더 (20) 는 비디오 데이터를 인코딩하고 디코딩한다. 비디오 인코더 (20) 및 비디오 디코더 (30) 는 "인-루프" 필터링 프로세스로서 필터링을 수행하여, 후속 인코딩된 및/또는 디코딩된 픽처들에 대한 참조 픽처들로서 후속하여 사용되는 디코딩된 픽처들에 ALF 를 적용할 수도 있다. "비디오 코더" 의 본 개시물의 설명들은 비디오 인코더 (20) 와 같은 비디오 인코더, 및 비디오 디코더 (30) 와 같은 비디오 디코더의 설명들로서 이해되어야 한다.

[0069] 본 개시물의 기법들에 따르면, (비디오 인코더 (20) 및/또는 비디오 디코더 (30) 와 같은) 비디오 코더가 블록들의 다양한 클래스들에 대한 필터들을 구성할 수도 있다. 비디오 코더는 예를 들어, 블록의 픽셀들로부터 계산된 기울기 (gradient) 의 방향성에 기초하여 블록을 분류할 수도 있다. 비디오 코더는 수평 방향 기울기, 수직 방향 기울기, 45 도 대각선 방향 기울기, 또는 135 도 대각선 방향 기울기 중 하나 이상을 계산할 수도 있다. 일반적으로, 비디오 코더는 블록의 각각의 클래스에 대한 상이한 필터를 구성할 수도 있다. 그러나, 일부의 경우, 비디오 코더는 블록들의 2개 이상의 클래스들에 의해 사용되는 필터 계수들의 세트를 계산할 수도 있다. 예를 들어, 비디오 코더는 필터 계수들의 복수의 세트들을 계산할 수도 있다. 비디오 코더는 필터 계수들의 복수의 세트들에서 하나의 필터 계수들의 세트를 식별하는 블록들의 클래스들의 각각 (또는, 블록들의 클래스들의 서브세트) 에 대한 인덱스를 추가로 코딩할 수도 있다. 따라서, 비디오 코더가 블록들의 2개 이상의 클래스들에 대해 동일한 인덱스를 코딩하는 경우, 이들 블록들의 클래스들은 동일한 필터 계수들의 세트, 즉, 코딩된 인덱스 값에 대응하는 필터 계수들의 세트를 사용할 것이다.

[0070] 추가적으로 또는 대안적으로, 비디오 코더는 블록들의 적어도 하나의 클래스에 대해, 필터가 동일한 필터를 이용하지만, 필터의 필터 서포트 영역 또는 필터 계수들에 적용되는 기하학적 변환을 이용하여 발생되어야 한다는 것을 표시할 수도 있다. 따라서, 심지어 동일한 클래스 인덱스를 갖는 블록들에 대해, 필터 서포트는 기하학적 변환들로 인해 상이할 수도 있다. 기능적으로, 기하학적 변환을 필터 서포트 영역에 적용하는 것은 기하학적 변환을 필터 계수들에 적용하는 것과 동일한 수학적 결과들을 산출하며, 따라서, 비디오 코더가 기하학적 변환을 필터 서포트 영역 또는 필터 계수들에 적용할 수도 있다. 이 예에서, 비디오 코더는 블록의 현재의 클래스에 대해, 기하학적 변환 유형 (예컨대, 회전, 대각선 플립, 또는 수직 플립, 또는 비-기하학적 변환)

뿐만 아니라, 필터 계수들이 블록의 현재의 클래스에 대한 발생된 필터에 사용될 이전에 구성된 필터를 식별하는 데이터를 코딩할 수도 있다.

[0071] 추가적으로 또는 대안적으로, 비디오 코더는 적어도 하나의 고정 필터가 블록의 클래스에 대한 필터 계수들을 예측하는데 사용되는지 여부를 표시하는 (예컨대, 플래그의) 값을 코딩할 수도 있다. 즉, 플래그는 예를 들어, 블록의 클래스의 필터에 대한 필터 계수들이 고정 필터로부터 예측되거나, 또는 대신에, 이전에 코딩된 픽처의 필터들로부터, 또는 이전에 코딩된 픽처들로부터의 고정 필터 또는 필터들로부터의 예측 없이, 예측된다 (또는, 전승된다) 것을 표시할 수도 있다. 일부 예들에서, 비디오 코더는 블록의 각각의 클래스에 대한 복수의 고정 필터들로 구성될 수도 있다. 이러한 예들에서, 그 값이 필터 계수들이 고정 필터로부터 예측된다는 것을 표시할 때, 비디오 코더는, 블록의 클래스에 대한 현재의 필터의 필터 계수들을 예측하는데 사용될 복수의 고정 필터들 중 하나를 식별하는, 복수의 고정 필터들로의 인덱스를 추가로 코딩할 수도 있다.

[0072] 일부 예들에서, 비디오 코더는 클래스들 중 임의의 클래스에 대한 필터 계수들 중 어느 것도 임의의 고정 필터로부터 예측되지 않는다는 것, 또는 모든 필터 계수들이 고정 필터들로부터 예측된다는 것, 또는 고정 필터들로부터 예측된 일부 필터 계수들이 존재하지만 다른 필터 계수들은 고정 필터들로부터 예측되지 않는다는 것을 표시하는 정보를 추가로 코딩할 수도 있다. 필터 계수들 중 어느 것도 고정 필터로부터 예측되지 않을 때, 비디오 코더는 복수의 고정 필터들로의 인덱스를 나타내는 데이터의 코딩을 회피할 (예컨대, 생략하거나 또는 방지할) 수도 있다. 한편, 필터 계수들 중 일부 또는 모두가 고정 필터로부터 예측될 때, 비디오 코더는 각각의 클래스 하나의 클래스에 대한 필터가 고정 필터로부터 예측되는지 여부에 대한 플래그를 추가로 코딩할 수도 있다. 게다가, 비디오 코더는 고정 필터로부터 예측되는 이들 필터 계수들의 세트들에 대한 개별 복수의 고정 필터들로의 인덱스들을 코딩할 수도 있으며, 여기서, 인덱스들은 대응하는 필터 계수들이 예측될 참조 고정 필터를 식별한다. 더욱이, 비디오 코더는 인덱스들에 의해 식별된 고정 필터들을 이용하여 대응하는 필터 계수들을 예측할 수도 있다. 예측 후, 비디오 코더는 예측된 필터 계수들과 실제 필터 계수 값들 사이의 차이들을 나타내는 잔차 값들 (예컨대, 오프셋 값들) 을 추가로 코딩할 수도 있다.

[0073] 비디오 코더는 이들 기법들을 단독으로 또는 조합하여 적용할 수도 있다. 예를 들어, 비디오 코더는 필터 계수들의 복수의 세트들을 명시적으로 코딩하고, 필터 계수들의 복수의 세트들로의, 블록들의 클래스들의 제 1 세트에 대한 인덱스들을 코딩하고, 그리고 고정 필터들 또는 하나 이상의 참조 픽처들의 필터들로부터 블록들의 클래스들의 제 2 세트의 필터들에 대한 필터 계수들을 예측 코딩할 수도 있다. 대안적으로, 비디오 코더는 고정 필터들로부터 또는 하나 이상의 이전에 코딩된 픽처들의 필터로부터 예측된 필터 계수들의 복수의 세트들의 각각을 코딩하고, 그후, 블록들의 클래스들의 각각에 대해, 필터 계수들의 세트들 중 하나를 식별하는 인덱스를 코딩할 수도 있다. 또한, 다른 예로서, 비디오 코더는 상기 기법들 중 임의의 기법을 이용하여 블록들의 클래스들의 제 1 세트에 대한 필터 계수들을 코딩하고, 그리고, 기하학적 변환을 식별하는 데이터 및 필터 계수들을 전승할 블록들의 클래스들의 제 1 세트 중 하나를 식별하는 데이터를 포함한, 블록들의 클래스들의 제 2 세트에 대한 기하학적 변환 정보를 코딩할 수도 있다.

[0074] 도 2 는 BA (블록 기반의 적응) 분류에 사용되는 15 개의 그룹들 (또한, 클래스들로 불림) 을 예시하는 개념도이다. 도 2 의 예에서, 필터들은 활동도 메트릭 (즉, 범위 0 내지 범위 4) 및 방향 메트릭에 대한 값들의 범위들에 맵핑된다. 도 2 에서의 방향 메트릭은 상기 수식 3 으로부터 0, 1, 및 2 의 값들에 대응할 수도 있는, 방향 없음, 수평, 및 수직의 값들을 갖는 것으로 도시된다. 도 2 의 구체적인 예는 15 개의 그룹들에 맵핑되는 것으로 6개의 상이한 필터들 (즉, 필터 1, 필터 2... 필터 6) 을 나타내지만, 더 많거나 또는 더 적은 필터들이 유사하게 사용될 수도 있다. 도 2 는 그룹들 221 내지 235 로서 식별되는 15 개의 그룹들을 갖는 예를 나타내지만, 더 많거나 또는 더 적은 그룹들이 또한 사용될 수도 있다. 예를 들어, 활동도 메트릭에 대한 5개의 범위들 대신, 더 많거나 또는 더 적은 그룹들을 초래하는, 더 많거나 또는 더 적은 범위들이 사용될 수도 있다. 추가적으로 또는 대안적으로, 단지 3개의 방향들 대신, 추가적인 방향들 (예컨대, 45-도 방향 및 135-도 방향) 이 또한 사용될 수도 있다. 예를 들어, 활동도 메트릭의 5 개의 범위들이 5개의 방향들 (예컨대, 방향 없음, 수평, 수직, 45-대각선, 및 135-대각선) 과 함께 사용되면, 25 개의 그룹들이 분류에 사용될 것이며, 25 개의 그룹들의 각각이 필터에 맵핑될 것이다.

[0075] 픽처에서의 크로마 성분들 양쪽에 대해, 필터 계수들의 단일 세트가 적용될 수도 있다. 참조 픽처들의 필터 계수들은 저장되어 현재의 픽처의 필터 계수들로서 재사용되도록 허용될 수도 있다. 현재의 픽처는 참조 픽처들에 대해 저장된 필터 계수들을 이용하도록 선택하고, 필터 계수들 시그널링을 바이패스할 수도 있다. 이 경우, 단지 참조 픽처들 중 하나로의 인덱스만이 시그널링될 필요가 있으며, 표시된 참조 픽처의 저장된 필

터 계수들은 현재의 픽처에 대해 전승된다.

[0076] 본 개시물의 기법들에 따르면, 필터링 전에, 비디오 코더 (예컨대, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 는 ALF 를 수행하기 전에, 복원된 픽셀(들) 의 기울기의 방위에 따라서, 회전, 대각선, 및 수직 플립과 같은, 어떤 기하학적 변환들을, 필터 서포트 영역 내 픽셀들 (수식 (1) 에서 필터링된 계수들로 곱해지는 픽셀들) 에 적용할 수도 있다. 이들 변환들은 픽처 내 상이한 영역들 사이의 유사성, 예컨대, 이들의 방향성을 증가시킬 수도 있다. 이것은 디코더로 전송되어야 하는 필터들의 개수를 감소시키고, 따라서 그들을 표현하는데 요구되는 비트수를 감소시키거나, 또는 대안적으로, 복원 에러를 감소시킬 수 있다. 변환들을 필터 서포트 영역들에 적용하는 것은 변환들을 필터 계수들에 직접 적용하는 것과 동등하다. 따라서, 비디오 코더는 변환들을, 필터 서포트 영역들 대신, 필터 계수들에 대신 적용할 수도 있다.

[0077] 이미 소개된 다른 기법들과 공동으로 또는 이와 독립적으로 사용될 수도 있는, 본 개시물의 다른 기법에 따르면, 필터 계수들을 표현하는데 요구되는 비트수를 감소시키기 위해, 비디오 코더는 블록들의 상이한 클래스들을 병합할 수도 있다. 그러나, HM-3.0 에서와 달리, 클래스들의 임의의 세트는, 심지어 클래스들이 C 의 비-연속된 값들을 갖더라도, 병합될 수 있다. 클래스들이 병합되는 정보는 클래스들의 각각 (예컨대, 25 개의 클래스들의 각각) 에 대해, 인덱스 i_C 를 전송함으로써 제공된다. 동일한 인덱스 i_C 를 갖는 클래스들은 동일한 필터 f(k,l) 를 공유한다. 따라서, 비디오 코더는 필터 계수들의 복수의 세트들을 발생시키고, 그리고 복수의 클래스들의 각각에 대해, 인덱스 값 i_C 를 나타내는 데이터를 코딩할 수도 있으며, 여기서, C 는 1 과 클래스들의 개수 사이의 값일 수도 있으며, i_C 는 클래스 C 에 대한 필터 계수들의 세트를 식별하는, 발생된 필터 계수들의 세트들로의 인덱스이다.

[0078] 이미 소개된 다른 기법들과 공동으로 또는 이와 독립적으로 사용될 수도 있는, 본 개시물의 다른 기법에 따르면, HM-3.0 에서 설명된 바와 같은 ALF 에 대한 다른 변경은, 비디오 코더 (예컨대, 비디오 인코더 (20) 또는 비디오 디코더 (30)) 가 예를 들어, 오프라인 트레인에 의해, 획득되어, 필터 계수들을 예측 코딩하기 위해 각각의 클래스에 할당된, n개의 고정 필터들의 세트를 이용할 수도 있다는 것이다. 필터 계수들이 고정 필터로부터 예측되는 각각의 클래스에 대해, 비디오 코더는 고정 필터들 중 어느 것이 사용되는지를 나타내는 데이터를 코딩할 수도 있다. 다른 예에서, 일부 클래스들에는 동일한 n 개의 고정 필터들의 세트가 할당될 수 있거나 또는 심지어 n개의 고정 필터들의 동일한 세트가 모든 클래스들에 할당될 수 있다. 심지어 고정 필터가 주어진 클래스에 대해 선택될 때, 적응 필터의 계수들이 여전히 이 클래스에 대해 시그널링될 수 있다. 복원된 이미지에 적용될 수도 있는 필터의 계수들은 이 경우, 시그널링된 계수들 f(k,l) 과 고정 필터 계수들의 총합이다. 클래스들의 개수는, 심지어 상이한 고정 필터들이 이들 중에서 선택되더라도, 동일한 시그널링된 계수들 f(k,l) 을 공유할 수도 있다.

[0079] 위에서 소개된 기법들의 예시적인 구현예들이 이하 설명될 것이다. 일 예에서, 각각의 N x N 블록은 다음과 같이 그의 방향성 D 및 활동도 $\hat{A}$ 의 양자화된 값에 기초하여 25 개의 클래스들 중 하나로 분류된다:

[0080] $$C = 5D + \hat{A}. \quad (6)$$

[0081] 수평 기울기, 수직 기울기, 및 2개의 대각선 기울기들의 값들은 1-D 라플라시안을 이용하여 다음과 같이 계산된다:

[0082] $$g_v = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} V_{k,l}$$, 여기서,

[0083] $$V_{k,l} = |2R(k,l) - R(k,l-1) - R(k,l+1)|, \quad (7)$$

[0084] $$g_h = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} H_{k,l}$$, 여기서,

[0085] $$H_{k,l} = |2R(k,l) - R(k-1,l) - R(k+1,l)|$$, (8)

[0086] $$g_{d1} = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} D1_{k,l}$$, 여기서,

[0087] $$D1_{k,l} = |2R(k,l) - R(k-1,l-1) - R(k+1,l+1)|$$, (9)

[0088] $$g_{d2} = \sum_{k=i-M}^{i+N+M-1} \sum_{j=j-M}^{j+N+M-1} D2_{k,l}$$, 여기서,

[0089] $$D2_{k,l} = |2R(k,l) - R(k-1,l+1) - R(k+1,l-1)|$$. (10)

[0090] 이들 예들에서, g_v 는 수직 기울기를 나타내며, g_h 는 수평 기울기를 나타내며, g_{d1} 은 45-도 대각선 기울기를 나타내며, g_{d2} 는 135-도 대각선 기울기를 나타낸다.

[0091] 인덱스들 i 및 j 는 N x N 블록에서의 좌상단 픽셀의 좌표들을 지칭한다. 방향성 D 를 할당하기 위해, 수평 및 수직 기울기들의 최대 및 최소의 비

[0092] $$g_{h,v}^{max} = max(g_h, g_v),\ g_{h,v}^{min} = min(g_h, g_v)$$, (11)

[0093] 및 2개의 대각선 기울기들의 최대 및 최소의 비

[0094] $$g_{d0,d1}^{max} = max(g_{d0}, g_{d1}),\ g_{d0,d1}^{min} = min(g_{d0}, g_{d1})$$, (12)

[0095] 가 서로에 대해 그리고 임계치들 t_1 및 t_2 의 세트에 대해 다음과 같이 비교된다:

[0096] 단계 1. 양쪽 다 $g_{h,v}^{max} \le t_1 \cdot g_{h,v}^{min}$ 및 $g_{d0,d1}^{max} \le t_1 \cdot g_{d0,d1}^{min}$ 이면, D 는 0 으로 설정된다 (블록이 "텍스쳐" 로서 분류된다).

[0097] 단계 2. $g_{h,v}^{max}/g_{h,v}^{min} > g_{d0,d1}^{max}/g_{d0,d1}^{min}$ 이면, 단계 3 으로부터 속행하고, 그렇지 않으면 단계 4 로부터 속행한다.

[0098] 단계 3. $g_{h,v}^{max} > t_2 \cdot g_{h,v}^{min}$ 이면, D 는 2 로 설정되며, 그렇지 않으면, D 는 1 로 설정된다 (블록이 "강한 수평/수직" 또는 "수평/수직" 으로 각각 분류된다).

[0099] 단계 4. $g_{d0,d1}^{max} > t_2 \cdot g_{d0,d1}^{min}$ 이면, D 는 4 로 설정되며, 그렇지 않으면 D 는 3 으로 설정된다 (블록이 "강한 대각선" 또는 "대각선" 으로 각각 분류된다).

[0100] 활동도 값 A 는 다음과 같이 계산된다:

[0101] $$A = \sum_{k=i-M}^{i+N+M-1} \sum_{l=j-M}^{j+N+M-1} (V_{k,l} + H_{k,l})$$ (13)

[0102] A 는 0 내지 4 를 포함한 범위로 추가로 양자화되며, 양자화된 값은 $\hat{A}$ 로서 표시된다:

[0103] $A = \min(15, (24 \cdot A) \gg 13)$ 및 $\hat{A} = Q[A]$, 여기서,

[0104] $Q = \{0, 1, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 3, 4\}$. (14)

[0105] 최고 25 개의 필터들이 각각의 클래스 당 하나씩 시그널링될 수 있다. 필터 계수들을 표현하는데 요구되는 비트수를 감소시키기 위해, 상이한 클래스들이 병합될 수 있다. 그러나, HM-3.0 에서와는 달리, 심지어 클래스들이 C 의 비-연속된 값들을 갖더라도, 클래스들의 임의의 세트가 병합될 수 있다. 클래스들이 병합되는 정보가, 25 개의 클래스들의 각각에 대해, 인덱스 i_C 를 코딩함으로써 제공된다. 동일한 인덱스 i_C 를 가지는 클래스들은 동일한 필터 f(k,l) 을 공유한다.

[0106] 각각의 블록을 필터링하기 전에, 회전 또는 대각선 및 수직 플립과 같은 간단한 기하학적 변환들이 그 블록에 대해 계산된 기울기 값들에 따라서, 필터 계수들 f(k,l) 에 적용될 수 있다. 이것은 이들 변환들을 필터 서포트 영역 내 픽셀들에 적용하는 것과 동등하다. 아이디어는, 이들의 방향성을 정렬함으로써 ALF 가 적용되는 상이한 블록들을 더 유사하게 만드는 것이다. 기하학적 변환들은 다음과 같이 정의되며,

[0107] FlipDiagonal: $f_D(k,l)=f(l,k)$,

[0108] FlipVertical: $f_V(k,l)=f(k,K-l-1)$,

[0109] RotateRight: $f_R(k,l)=f(K-l-1,k)$, (15)

[0110] 여기서, K 는 필터의 사이즈이며, $0 \le k,l \le K-1$ 는 계수들 좌표들이며, 따라서 로케이션 (0,0) 은 좌상단 모서리에 있고 로케이션 (K-1,K-1) 은 우하단 모서리에 있다. 수식들 (7)-(10) 을 이용하여 계산된 기울기 값들과 수식 (15) 에서 규정된 기하학적 변환들 사이의 맵핑이 테이블 1 에 주어진다.

[0111] 테이블 1: N x N 블록 및 필터에 대해 계산된 기울기의 맵핑

기울기 값들	필터 계수들
$g_{d2} < g_{d1}$ 및 $g_h < g_v$	$f(k,l)$
$g_{d2} < g_{d1}$ 및 $g_v < g_h$	$f_D(k,l)$
$g_{d1} < g_{d2}$ 및 $g_h < g_v$	$f_V(k,l)$
$g_{d1} < g_{d2}$ 및 $g_v < g_h$	$f_R(k,l)$

[0112]

[0113] 일부 예들에서, 필터 계수들의 예측은 고정 필터들로부터 유래될 수도 있다. 필터 계수들의 시간 예측은 인터 코딩된 프레임들에 대한 코딩 효율을 향상시킬 수도 있다. 시간 예측이 이용불가능할 때 (인트라 프레임들) 코딩 효율을 향상시키기 위해, n개의 고정 필터들의 세트가 각각의 클래스에 할당된다. 일부 예들에서, 25 개의 클래스들의 각각에 대해, 고정 필터가 사용되면 플래그가, 필요할 경우, 선택된 필터의 인덱스가 시그널링된다. 일부 예들에서, 플래그의 값은 고정 필터가 사용되는지 여부를 표시할 수도 있다. 일부 예들에서, 필요할 경우, 선택된 필터를 나타내는 인덱스가 시그널링될 수도 있다. 대안적으로, 다음 3개의 예시적인 정의된 경우들 중 하나가 사용될 수도 있다. 첫번째 경우, 25 개의 클래스들의 필터들 중 어느 것도 고정 필터로부터 예측되지 않는다. 두번째 경우, 클래스들의 모든 필터들이 고정 필터로부터 예측된다. 세번째 경우, 일부 클래스들과 연관된 필터들이 고정 필터들로부터 예측되고 클래스들의 나머지와 연관된 필터들은 고정 필터들로부터 예측되지 않는다.

[0114] 상기 3개의 예시적인 경우들 중 어느 것이 적용되는지를 표시하기 위해 변수가 코딩될 수도 있다. 또한, 다음이 적용될 수도 있다:

[0115] ● 변수가 경우 1 을 표시하면, 고정 필터의 인덱스를 추가로 시그널링할 필요가 없다.

[0116] ● 그렇지 않고, 변수가 경우 2 를 표시하면, 각각의 클래스에 대해 선택된 고정 필터의 인덱스가 코딩된

다.

[0117] ● 그렇지 않고, 변수가 경우 3 을 표시하면, 각각의 클래스에 대해 1 비트가 먼저 시그널링될 수도 있으며, 고정 필터가 사용되면, 인덱스가 추가로 시그널링될 수도 있다.

[0118] 심지어 고정 필터가 주어진 클래스에 대해 선택되더라도, 적응 필터의 계수들이 이 클래스에 대해 여전히 전송될 수도 있다. 복원된 이미지에 적용될 필터의 계수들은, 이 경우, 시그널링된 계수들 f(k,l) 과 고정 필터 계수들의 총합이다. 클래스들의 개수는 심지어 이들에 대해 상이한 고정 필터들이 선택되더라도, 동일한 계수들 f(k,l) 을 공유할 수 있다. 하나의 대안에서, 인트라 프레임들에 적용된 고정 필터들은 또한 P 또는 B 슬라이스들과 같은, 인터 코딩된 프레임들에 적용될 수도 있다.

[0119] 도 3 은 필터 계수 차이들을 시그널링하는 예시적인 패턴을 예시하는 개념도이다. 하나 이상의 고정 필터들, 예컨대, 고정 필터들에 대한 계수 값들을 정의하기 위해 다음 기법들이 사용될 수도 있다. 일 예에서, 고정 필터들은 오프-라인 트레이닝으로부터 유도되며, 단지 가장 큰 필터 탭 (예컨대, 9x9 다이아몬드 필터 형상) 에 대한 필터 계수들 만이 정의된다. 유도된 필터 계수들과 선택된 고정 필터들 사이의 차이들을 코딩하기 전에, 비디오 코더는 필터 계수 차이들을 시그널링하는 패턴을 표시하는 인덱스를 먼저 코딩할 수도 있다. 예를 들어, 9x9 대칭 다이아몬드 필터 형상에 대한 고정 필터가 도 3 에 도시되며, 여기서, 필터 계수는 F_i 로 표시된다 (i 는 0 내지 20 을 포함한 범위이다). 도 3 에 나타낸 바와 같이, 시그널링된 패턴 인덱스가 5x5 다이아몬드 필터 형상을 표시할 때, 그리고 그 필터 계수들이 고정 필터로부터 예측될 때, 도 3 의 그레이 영역들에 로케이트된 유도된 필터 계수들의 값들은 변경되지 않고 유지되며 이들 위치들에 대한 차이들을 시그널링할 필요가 없다 (즉, 이들 그레이 위치들에 대한 차이들은 0 과 동일하다). 이 경우, 비디오 코더는 유도된 계수와 고정 필터 계수들 (F0 ~ F6) 사이의 차이들을 정의하는 데이터를 코딩한다. 대안적으로, 비디오 코더는 DC 계수 (즉, F6) 의 차이를 정의하는 데이터를 코딩할 필요가 없지만, 대신, 다른 코딩된 차이 값들에 대한 DC 계수의 차이를 유도한다. 이 경우, 필터 계수들이 고정 필터로부터 예측될 때 9x9 다이아몬드 필터링 프로세스가 항상 적용된다는 점에 유의한다.

[0120] 다른 예에서, 고정 필터들의 M 개의 버전들이 사전-정의될 수도 있으며, 여기서, M 은 지원되는 필터 형상들의 총 개수를 나타낸다. 비디오 코더는 선택된 필터 형상의 인덱스를 먼저 코딩하고, 그후, 필터 형상과 연관된 선택된 고정 필터들의 인덱스들을 나타내는 데이터를 코딩할 수도 있다. 이 경우, (M 의 값에 따라) 다수의 상이한 필터링 프로세스들이 지원된다는 점에 유의한다.

[0121] 도 4 는 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 비디오 인코더 (20) 를 예시하는 블록도이다. 비디오 인코더 (20) 는 비디오 슬라이스들 내 비디오 블록들의 인트라-코딩 및 인터-코딩을 수행할 수도 있다. 인트라 코딩은 주어진 비디오 프레임 또는 픽처 내 비디오에서 공간 리던던시를 감소시키거나 또는 제거하기 위해, 공간 예측에 의존한다. 인터-코딩은 비디오 시퀀스의 인접 프레임들 또는 픽처들 내 비디오에서 시간 리던던시를 감소시키거나 또는 제거하기 위해, 시간 예측에 의존한다. 인트라-모드 (I 모드) 는 여러 공간 기반의 압축 모드들 중 임의의 압축 모드를 지칭할 수도 있다. 단방향 예측 (P 모드) 또는 양방향-예측 (B 모드) 과 같은 인터-모드들은 여러 시간-기반의 압축 모드들 중 임의의 모드를 지칭할 수도 있다.

[0122] 도 4 의 예에서, 비디오 인코더 (20) 는 비디오 데이터 메모리 (33), 파티셔닝 유닛 (35), 예측 프로세싱 유닛 (41), 합산기 (50), 변환 프로세싱 유닛 (52), 양자화 유닛 (54), 엔트로피 인코딩 유닛 (56) 을 포함한다. 예측 프로세싱 유닛 (41) 은 모션 추정 유닛 (MEU) (42), 모션 보상 유닛 (MCU) (44), 및 인트라 예측 프로세싱 유닛 (46) 을 포함한다. 비디오 블록 복원을 위해, 비디오 인코더 (20) 는 또한 역양자화 유닛 (58), 역변환 프로세싱 유닛 (60), 합산기 (62), ALF 유닛 (64), 및 디코딩된 픽처 버퍼 (DPB) (66) 를 포함한다.

[0123] 도 4 에 나타낸 바와 같이, 비디오 인코더 (20) 는 비디오 데이터를 수신하고 수신된 비디오 데이터를 비디오 데이터 메모리 (33) 에 저장한다. 비디오 데이터 메모리 (33) 는 비디오 인코더 (20) 의 컴포넌트들에 의해 인코딩될 비디오 데이터를 저장할 수도 있다. 비디오 데이터 메모리 (33) 에 저장된 비디오 데이터는 예를 들어, 비디오 소스 (18) 로부터 획득될 수도 있다. DPB (66) 는 예컨대, 인트라- 또는 인터-코딩 모드들에서 비디오 인코더 (20) 에 의해 비디오 데이터를 인코딩할 때에 사용하기 위해 참조 비디오 데이터를 저장하는 참조 픽처 메모리일 수도 있다. 비디오 데이터 메모리 (33) 및 DPB (66) 는 동기식 DRAM (SDRAM) 을 포함한 동적 랜덤 액세스 메모리 (DRAM), 자기저항 RAM (MRAM), 저항 RAM (RRAM), 또는 다른 유형들의 메모리 디바이스들과 같은, 다양한 메모리 디바이스들 중 임의의 메모리 디바이스에 의해 형성될 수도 있다. 비디오 데이터 메모리 (33) 및 DPB (66) 는 동일한 메모리 디바이스 또는 별개의 메모리 디바이스들에 의해 제공될 수도 있다.

다양한 예들에서, 비디오 데이터 메모리 (33) 는 비디오 인코더 (20) 의 다른 컴포넌트들과의 온칩, 또는 그들 컴포넌트들에 대한 오프-칩일 수도 있다.

[0124] 파티셔닝 유닛 (35) 은 비디오 데이터 메모리 (33) 로부터 비디오 데이터를 취출하고 그 비디오 데이터를 비디오 블록들로 파티셔닝한다. 이 파티셔닝은 또한 예컨대, LCU들 및 CU들의 쿼드트리 구조에 따라서, 슬라이스들, 타일들, 또는 다른 더 큰 유닛들로 파티셔닝하는 것 뿐만 아니라, 비디오 블록 파티셔닝을 포함할 수도 있다. 비디오 인코더 (20) 는 일반적으로 인코딩되는 비디오 슬라이스 내 비디오 블록들을 인코딩하는 컴포넌트들을 예시한다. 슬라이스는 다수의 비디오 블록들로 (그리고, 아마도, 타일들로서 지칭되는 비디오 블록들의 세트들로) 분할될 수도 있다. 예측 프로세싱 유닛 (41) 은 에러 결과들 (예컨대, 코딩 레이트 및 왜곡의 레벨) 에 기초하여, 현재의 비디오 블록에 대해, 복수의 인트라 코딩 모드들 중 하나 또는 복수의 인터 코딩 모드들 중 하나와 같은, 복수의 가능한 코딩 모드들 중 하나를 선택할 수도 있다. 예측 프로세싱 유닛 (41) 은 최종 인트라- 또는 인터-코딩된 블록을, 합산기 (50) 에 제공하여 잔차 블록 데이터를 발생하고, 그리고 합산기 (62) 에 제공하여, 참조 픽처로서 사용하기 위한 인코딩된 블록을 복원할 수도 있다.

[0125] 예측 프로세싱 유닛 (41) 내의 인트라 예측 유닛 (46) 은 코딩될 현재의 블록과 동일한 프레임 또는 슬라이스에서의 하나 이상의 이웃하는 블록들에 대해, 현재의 비디오 블록의 인트라-예측을 수행하여, 공간 압축을 제공할 수도 있다. 예측 프로세싱 유닛 (41) 내 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 하나 이상의 참조 픽처들에서의 하나 이상의 참조 블록들에 대해, 현재의 비디오 블록의 인터-예측을 수행하여, 시간 압축을 제공한다.

[0126] 모션 추정 유닛 (42) 은 비디오 시퀀스에 대한 미리 결정된 패턴에 따라서 비디오 슬라이스에 대한 인터-예측 모드를 결정하도록 구성될 수도 있다. 미리 결정된 패턴은 그 시퀀스에서의 비디오 슬라이스들을 P 슬라이스들 또는 B 슬라이스들로서 지시할 수도 있다. 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 고도로 통합될 수도 있지만, 개념적인 목적들을 위해 별개로 예시된다. 모션 추정 유닛 (42) 에 의해 수행되는 모션 추정은 모션 벡터들을 발생시키는 프로세스이며, 이 프로세스는 비디오 블록들에 대한 모션을 추정한다. 모션 벡터는, 예를 들어, 참조 픽처 내 예측 블록에 대한, 현재의 비디오 프레임 또는 픽처 내 비디오 블록의 PU 의 변위를 나타낼 수도 있다.

[0127] 예측 블록은 픽셀 차이의 관점에서 코딩되는 비디오 블록의 PU 에 가깝게 매칭하는 것으로 발견되는 블록이며, SAD, SSD, 또는 다른 차이 메트릭들에 의해 결정될 수도 있다. 일부 예들에서, 비디오 인코더 (20) 는 DPB (66) 에 저장된 참조 픽처들의 서브-정수 픽셀 위치들에 대한 값들을 계산할 수도 있다. 예를 들어, 비디오 인코더 (20) 는 참조 픽처의 1/4 픽셀 위치들, 1/8 픽셀 위치들, 또는 다른 분수 픽셀 위치들의 값들을 내삽할 수도 있다. 따라서, 모션 추정 유닛 (42) 은 풀 픽셀 위치들 및 분수 픽셀 위치들에 대해, 모션 탐색을 수행하고, 분수 픽셀 정밀도를 가진 모션 벡터를 출력할 수도 있다.

[0128] 모션 추정 유닛 (42) 은 PU 의 위치를 참조 픽처의 예측 블록의 위치와 비교함으로써 인터-코딩된 슬라이스에서의 비디오 블록의 PU 에 대한 모션 벡터를 계산한다. 참조 픽처는 제 1 참조 픽처 리스트 (List 0) 또는 제 2 참조 픽처 리스트 (List 1) 로부터 선택될 수도 있으며, 이 리스트들 각각은 DPB (66) 에 저장된 하나 이상의 참조 픽처들을 식별한다. 모션 추정 유닛 (42) 은 그 계산된 모션 벡터를 엔트로피 인코딩 유닛 (56) 및 모션 보상 유닛 (44) 으로 전송한다.

[0129] 모션 보상 유닛 (44) 에 의해 수행되는 모션 보상은 가능한 한, 서브-픽셀 정밀도까지 내삽들을 수행함으로써, 모션 추정에 의해 결정된 모션 벡터에 기초하여 예측 블록을 페치하거나 또는 발생시키는 것을 수반할 수도 있다. 현재의 비디오 블록의 PU 에 대한 모션 벡터를 수신하자 마자, 모션 보상 유닛 (44) 은 모션 벡터가 참조 픽처 리스트들 중 하나에서 가리키는 예측 블록을 로케이트할 수도 있다. 비디오 인코더 (20) 는 코딩중인 현재의 비디오 블록의 픽셀 값들로부터 예측 블록의 픽셀 값들을 감산하여 픽셀 차이 값들을 형성함으로써 잔차 비디오 블록을 형성한다. 픽셀 차이 값들은 블록에 대한 잔차 데이터를 형성하며, 루마 및 크로마 차이 성분들 양쪽을 포함할 수도 있다. 합산기 (50) 는 이 감산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 모션 보상 유닛 (44) 은 또한 비디오 슬라이스의 비디오 블록들을 디코딩할 때에 비디오 디코더 (30) 에 의한 사용을 위해 비디오 블록들 및 비디오 슬라이스와 연관되는 신택스 엘리먼트들을 발생시킬 수도 있다.

[0130] 예측 프로세싱 유닛 (41) 이 현재의 비디오 블록에 대한 예측 블록을 인트라 예측 또는 인터 예측을 통해서 발생시킨 후, 비디오 인코더 (20) 는 현재의 비디오 블록으로부터 예측 블록을 감산함으로써 잔차 비디오 블록을 형성한다. 잔차 블록에서의 잔차 비디오 데이터는 하나 이상의 TU들에 포함되어, 변환 프로세싱 유닛 (52)

에 적용될 수도 있다. 변환 프로세싱 유닛 (52) 은 이산 코사인 변환 (DCT) 또는 개념적으로 유사한 변환과 같은 변환을 이용하여 잔차 비디오 데이터를 잔차 변환 계수들로 변환한다. 변환 프로세싱 유닛 (52) 은 잔차 비디오 데이터를 픽셀 도메인으로부터 주파수 도메인과 같은 변환 도메인으로 변환할 수도 있다.

[0131] 변환 프로세싱 유닛 (52) 은 최종 변환 계수들을 양자화 유닛 (54) 으로 전송할 수도 있다. 양자화 유닛 (54) 은 비트 레이트를 추가로 감소시키기 위해 변환 계수들을 양자화한다. 양자화 프로세스는 그 계수들의 일부 또는 모두와 연관되는 비트 심도를 감소시킬 수도 있다. 양자화의 정도는 양자화 파라미터를 조정함으로써 변경될 수도 있다. 일부 예들에서, 양자화 유닛 (54) 은 그후 양자화된 변환 계수들을 포함하는 매트릭스의 스캐닝을 수행할 수도 있다. 대안적으로, 엔트로피 인코딩 유닛 (56) 이 그 스캐닝을 수행할 수도 있다.

[0132] 양자화 이후, 엔트로피 인코딩 유닛 (56) 은 양자화된 변환 계수들을 엔트로피 인코딩한다. 예를 들어, 엔트로피 인코딩 유닛 (56) 은 컨텍스트 적응 가변 길이 코딩 (CAVLC), 컨텍스트 적응 2진 산술 코딩 (CABAC), 신택스-기반 컨텍스트-적응 2진 산술 코딩 (SBAC), 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩 또는 또 다른 엔트로피 인코딩 방법론 또는 기법을 수행할 수도 있다. 엔트로피 인코딩 유닛 (56) 에 의한 엔트로피 인코딩에 이어서, 인코딩된 비트스트림은 비디오 디코더 (30) 로 송신되거나, 또는 비디오 디코더 (30) 에 의한 추후 송신 또는 취출을 위해 아카이브될 수도 있다. 엔트로피 인코딩 유닛 (56) 은 또한 코딩중인 현재의 비디오 슬라이스에 대한 모션 벡터들 및 다른 신택스 엘리먼트들을 엔트로피 인코딩할 수도 있다.

[0133] 역양자화 유닛 (58) 및 역변환 프로세싱 유닛 (60) 은 역양자화 및 역변환을 각각 적용하여, 참조 픽처의 참조 블록으로 추후 사용을 위해 픽셀 도메인에서 잔차 블록을 복원한다. 모션 보상 유닛 (44) 은 잔차 블록을 참조 픽처 리스트들 중 하나 내 참조 픽처들 중 하나의 예측 블록에 가산함으로써 참조 블록을 계산할 수도 있다. 모션 보상 유닛 (44) 은 또한 하나 이상의 내삽 필터들을 그 복원된 잔차 블록에 적용하여, 모션 추정에 사용하기 위한 서브-정수 픽셀 값들을 계산할 수도 있다. 합산기 (62) 는 복원된 잔차 블록을 모션 보상 유닛 (44) 에 의해 발생되는 모션 보상된 예측 블록에 가산하여, 복원된 블록을 발생한다.

[0134] ALF 유닛 (64) 은 복원된 블록 (예컨대, 합산기 (62) 의 출력) 을 필터링하고 참조 블록으로서의 사용을 위해, 필터링된 복원된 블록을 DPB (66) 에 저장한다. 참조 블록은 후속 비디오 프레임 또는 픽처에서 블록을 인터-예측하기 위해 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 참조 블록으로서 사용될 수도 있다.
도 4 에 명시적으로 나타내지는 않았지만, 비디오 인코더 (20) 는 디블록 필터, 샘플 적응 오프셋 (SAO) 필터, 또는 다른 유형의 루프 필터 중 하나 이상과 같은 추가적인 필터들을 포함할 수도 있다. 디블록 필터는 블록 경계들에 디블록 필터링을 적용하여, 복원된 비디오로부터 블록킹 현상 아티팩트들을 제거할 수도 있다. SAO 필터는 전체 코딩 품질을 향상시키기 위해 오프셋들을 복원된 픽셀 값들에 적용할 수도 있다.
일부 구현예들에서, SAO 는 필터링의 특수한 경우 또는 특수 모드일 수도 있다. 추가적인 루프 필터들 (인 루프 또는 포스트 루프) 이 또한 사용될 수도 있다.

[0135] ALF 유닛 (64) 은, 단독으로 또는 비디오 인코더 (20) 의 다른 컴포넌트들과 함께, 본 개시물에서 설명된 다양한 ALF 기법들을 수행하도록 구성될 수도 있다. 예를 들어, 합산기 (62) 의 출력은 (예측 프로세싱 유닛 (41), 변환 프로세싱 유닛 (52), 및 양자화 유닛 (54) 과 같은) 비디오 인코더 (20) 의 다른 엘리먼트들에 의해 이전에 인코딩되었던 디코딩된 비디오 데이터이다. 따라서, ALF 유닛 (64) 은 디코딩된 비디오 데이터를 본 개시물의 기법들에 따라서 필터링할 수도 있다. 특히, ALF 유닛 (64) 은, ALF 유닛 (64) 이 후속 픽처들을 예측하기 위해 예측 프로세싱 유닛 (41) 에 의해 사용될, DPB (66) 에 저장된, 참조 픽처들로서 이후 사용되는 디코딩된 픽처들을 필터링한다는 점에서, "인-루프" 필터링 프로세스로서 필터링을 수행한다.

[0136] 본 개시물의 기법들에 따르면, ALF 유닛 (64) 은 블록들의 다양한 클래스들에 대한 필터들을 구성할 수도 있다.
ALF 유닛 (64) 은 예를 들어, 블록의 픽셀들로부터 계산된 기울기의 방향성에 기초하여 블록을 분류할 수도 있다. 일반적으로, ALF 유닛 (64) 은 블록의 각각의 클래스에 대해 상이한 필터를 구성할 수도 있다. 그러나, 일부의 경우, ALF 유닛 (64) 은 블록들의 2개 이상의 클래스들에 의해 사용되는 필터 계수들의 세트를 계산할 수도 있다. 예를 들어, ALF 유닛 (64) 은 필터 계수들의 복수의 세트들을 계산할 수도 있다. ALF 유닛 (64) 은 필터 계수들의 복수의 세트들에서 하나의 필터 계수들의 세트를 식별하는 블록들의 클래스들의 각각 (또는, 블록들의 클래스들의 서브세트) 에 대한 인덱스 값을 추가로 인코딩할 수도 있다. ALF 유닛 (64) 은 블록들의 클래스들의 각각에 대한 인덱스들을 규정하는 데이터 구조를 구성하고, 그 데이터 구조를, 엔트로피 인코딩되어 비트스트림에 포함되도록, 엔트로피 인코딩 유닛 (56) 에 제공할 수도 있다. 엔트로피 인코딩 유닛 (56) 은 데이터 구조를 엔트로피 인코딩하고, 그 데이터 구조를 예컨대, 슬라이스 헤더, 픽처 파라

미터 세트, 또는 다른 이러한 데이터 구조에 추가할 수도 있다. 따라서, ALF 유닛 (64) 이 블록들의 2개 이상의 클래스들에 대해 동일한 인덱스를 인코딩하는 경우, 이들 블록들의 클래스들은 필터 계수들의 동일한 세트, 즉, 인코딩된 인덱스 값에 대응하는 필터 계수들의 세트를 사용할 것이다.

[0137] 추가적으로 또는 대안적으로, ALF 유닛 (64) 은 블록의 적어도 하나의 클래스에 대해, 블록의 상이한 클래스와 동일한 필터 계수들을 이용하지만 필터의 필터 서포트 영역 또는 필터 계수들에 적용되는 기하학적 변환을 이용함으로써 필터가 발생될 것이라는 것을 표시할 수도 있다. 기능적으로, 기하학적 변환을 필터 서포트 영역에 적용하는 것은 기하학적 변환을 필터 계수들에 적용하는 것과 동일한 수학적 결과들을 산출하며, 따라서, ALF 유닛 (64) 은 기하학적 변환을 필터 서포트 영역 또는 필터 계수들에 적용할 수도 있다. 이 예에서, ALF 유닛 (64) 은 블록의 현재의 클래스에 대해, 기하학적 변환 유형 (예컨대, 회전, 대각선 플립, 또는 수직 플립) 뿐만 아니라, 필터 계수들이 블록의 현재의 클래스에 대한 발생된 필터에 사용될 이전에 구성된 필터를 식별하는 데이터를 인코딩할 수도 있다. 또, ALF 유닛 (64) 은 이러한 데이터를 엔트로피 인코딩 유닛 (56) 에 제공할 수도 있으며, 이 엔트로피 인코딩 유닛은 그 데이터를 엔트로피 인코딩하여, 그 데이터를 예를 들어, 슬라이스 헤더 또는 픽처 파라미터 세트와 같은 데이터 구조에 추가할 수도 있다.

[0138] 추가적으로 또는 대안적으로, ALF 유닛 (64) 은 고정 필터가 블록의 클래스에 대한 필터 계수들의 세트를 예측하는데 사용되는지 여부를 표시하는 (예컨대, 플래그의) 값을 인코딩할 수도 있다. 이와 유사하게, ALF 유닛 (64) 은 이 값들을, 예컨대, 슬라이스 헤더, 픽처 파라미터 세트, 또는 다른 이러한 데이터 구조의 부분으로서 엔트로피 인코딩되도록, 엔트로피 인코딩 유닛 (56) 에 제공할 수도 있다. 플래그는 예를 들어, 블록의 클래스의 필터에 대한 필터 계수들이 고정 필터로부터 예측되거나, 또는 대신에, 참조 픽처의 필터로부터 예측된다 (또는, 전승된다) 는 것을 표시할 수도 있다. 일부 예들에서, ALF 유닛 (64) 은 블록의 각각의 클래스에 대한 복수의 고정 필터들로 구성될 수도 있다. 이러한 예들에서, ALF 유닛 (64) 은 그 값이 필터 계수들이 고정 필터로부터 예측된다는 것을 표시할 때, 비디오 코더는, 블록의 클래스에 대한 현재의 필터의 필터 계수들을 예측하는데 사용될 복수의 고정 필터들 중 하나를 식별하는, 복수의 고정 필터들로의 인덱스를 추가로 인코딩할 수도 있다.

[0139] 일부 예들에서, ALF 유닛 (64) 은 클래스들 중 임의의 클래스에 대한 필터 계수들 중 어느 것도 임의의 고정 필터로부터 예측되지 않는다는 것, 모든 필터 계수들이 고정 필터로부터 예측된다는 것, 또는 고정 필터로부터 예측되는 일부 필터 계수들이 존재하지만 다른 필터 계수들은 고정 필터로부터 예측되지 않는다는 것을 표시하는 정보를 추가로 인코딩할 수도 있다. 필터 계수들 중 어느 것도 고정 필터로부터 예측되지 않을 때, ALF 유닛 (64) 은 복수의 고정 필터들로의 인덱스를 나타내는 데이터의 인코딩을 회피할 (예컨대, 생략하거나 또는 방지할) 수도 있다. 한편, 필터 계수들 중 일부 또는 모두가 고정 필터로부터 예측될 때, ALF 유닛 (64) 은 고정 필터로부터 예측되는 이들 필터 계수들의 세트들에 대한 개별 복수의 고정 필터들로의 인덱스들을 인코딩할 수도 있으며, 여기서, 인덱스들은 대응하는 필터 계수들이 예측될 참조 고정 필터를 식별한다. 더욱이, ALF 유닛 (64) 은 인덱스들에 의해 식별된 고정 필터들을 이용하여 대응하는 필터 계수들을 예측할 수도 있다.
예측 후, ALF 유닛 (64) 은 예측된 필터 계수들과 실제 필터 계수 값들 사이의 차이들을 나타내는 잔차 값들 (예컨대, 오프셋 값들) 을 추가로 인코딩할 수도 있다.

[0140] ALF 유닛 (64) 은 이들 기법들을 단독으로 또는 조합하여 적용할 수도 있다. 예를 들어, ALF 유닛 (64) 은 필터 계수들의 복수의 세트들을 명시적으로 인코딩하고, 필터 계수들의 복수의 세트들로의, 블록들의 클래스들의 제 1 세트에 대한 인덱스들을 인코딩하고, 그리고, 고정 필터들 또는 DPB (66) 내 하나 이상의 참조 픽처들의 필터들로부터 블록들의 클래스들의 제 2 세트의 필터들에 대한 필터 계수들을 예측 인코딩할 수도 있다. 대안적으로, ALF 유닛 (64) 은 고정 필터로부터 또는 DPB (66) 내 하나 이상의 참조 픽처들의 필터로부터 예측된 필터 계수들의 복수의 세트들의 각각을 인코딩하고, 그후, 블록들의 클래스들의 각각에 대해, 필터 계수들의 세트들 중 하나를 식별하는 인덱스를 인코딩할 수도 있다. 또한, 다른 예로서, ALF 유닛 (64) 은 상기 기법들 중 임의의 기법을 이용하여 블록들의 클래스들의 제 1 세트에 대한 필터 계수들을 인코딩하고, 그리고, 기하학적 변환을 식별하는 데이터 및 필터 계수들을 전승할 블록들의 클래스들의 제 1 세트 중 하나를 식별하는 데이터를 포함한, 블록들의 클래스들의 제 2 세트에 대한 기하학적 변환 정보를 인코딩할 수도 있다.

[0141] 어떻게 블록들의 클래스들에 대한 인덱스들을 선택할지, DPB (66) 의 고정 필터들 또는 참조 픽처들로부터 필터 계수들을 예측할지 여부, 및/또는 위에서 설명한 바와 같은 기하학적 변환을 이용하여 블록의 클래스에 대한 필터를 발생시킬지 여부를 결정하기 위해, 비디오 인코더 (20) 는 픽처 또는 픽처들의 시퀀스에 대해 다양한 인코딩 과정들을 실행하고, ALF 유닛 (64) 으로 하여금, 다양한 필터링 기법들을 픽처들 내 블록들의 다양한 클래스들에 적용하게 할 수도 있다. 비디오 인코더 (20) 는 디코딩된, 필터링된 픽처들에 대한 레이트 왜곡 최적

화 (RDO) 메트릭들을 계산하여, 필터 계수들과 필터 계수 예측 기법들의 어느 조합이 최상의 RDO 메트릭들을 초래하는지를 결정할 수도 있다. 비디오 인코더 (20) 는 그후, 각각의 픽처에 대한 최상의 RDO 메트릭들을 초래하는 필터링 기법들의 조합에 기초하여, 각각의 픽처에 대한 필터 계수들 및 필터링 기법들을 선택할 수도 있다.

[0142] 도 5 는 본 개시물에서 설명하는 기법들을 구현할 수도 있는 예시적인 비디오 디코더 (30) 를 예시하는 블록도이다. 도 5 의 예에서, 비디오 디코더 (30) 는 비디오 데이터 메모리 (78), 엔트로피 디코딩 유닛 (80), 예측 프로세싱 유닛 (81), 역양자화 유닛 (86), 역변환 프로세싱 유닛 (88), 합산기 (90), 및 디코딩된 픽처 버퍼 (DPB) (94) 를 포함한다. 예측 프로세싱 유닛 (81) 은 모션 보상 유닛 (82) 및 인트라 예측 유닛 (84) 을 포함한다. 비디오 디코더 (30) 는 일부 예들에서, 도 4 로부터의 비디오 인코더 (20) 에 대해 설명된 인코딩 과정과는 일반적으로 반대인 디코딩 과정을 수행할 수도 있다.

[0143] 디코딩 프로세스 동안, 비디오 디코더 (30) 는 인코딩된 비디오 슬라이스의 비디오 블록들 및 연관되는 신택스 엘리먼트들을 나타내는 인코딩된 비디오 비트스트림을 비디오 인코더 (20) 로부터 수신한다. 비디오 디코더 (20) 는 수신된 인코딩된 비디오 비트스트림을 비디오 데이터 메모리 (78) 에 저장한다. 비디오 데이터 메모리 (78) 는 비디오 디코더 (30) 의 컴포넌트들에 의해 디코딩될, 인코딩된 비디오 비트스트림과 같은, 비디오 데이터를 저장할 수도 있다. 비디오 데이터 메모리 (78) 에 저장된 비디오 데이터는 예를 들어, 저장 디바이스 (26) 로부터, 또는 카메라와 같은 로컬 비디오 소스로부터, 링크 (16) 를 통해서, 또는 물리적인 데이터 저장 매체들에 액세스함으로써 획득될 수도 있다. 비디오 데이터 메모리 (78) 는 인코딩된 비디오 비트스트림으로부터 인코딩된 비디오 데이터를 저장하는 코딩된 픽처 버퍼 (CPB) 를 형성할 수도 있다. DPB (94) 는 예컨대, 인트라- 또는 인터-코딩 모드들에서 비디오 디코더 (30) 에 의해 비디오 데이터를 디코딩할 때에 사용하기 위해 참조 비디오 데이터를 저장하는 참조 픽처 메모리일 수도 있다. 비디오 데이터 메모리 (78) 및 DPB (94) 는 DRAM, SDRAM, MRAM, RRAM, 또는 다른 유형들의 메모리 디바이스들과 같은, 다양한 메모리 디바이스들 중 임의의 메모리 디바이스에 의해 형성될 수도 있다. 비디오 데이터 메모리 (78) 및 DPB (94) 는 동일한 메모리 디바이스 또는 별개의 메모리 디바이스들에 의해 제공될 수도 있다. 다양한 예들에서, 비디오 데이터 메모리 (78) 는 비디오 디코더 (30) 의 다른 컴포넌트들과의 온칩, 또는 그들 컴포넌트들에 대한 오프-칩일 수도 있다.

[0144] 비디오 디코더 (30) 의 엔트로피 디코딩 유닛 (80) 은 비디오 데이터 메모리 (78) 에 저장된 비디오 데이터를 엔트로피 디코딩하여, 양자화된 계수들, 모션 벡터들, 및 다른 신택스 엘리먼트들을 발생시킨다. 엔트로피 디코딩 유닛 (80) 은 모션 벡터들 및 다른 신택스 엘리먼트들을 예측 프로세싱 유닛 (81) 으로 포워딩한다. 비디오 디코더 (30) 는 신택스 엘리먼트들을 비디오 슬라이스 레벨 및/또는 비디오 블록 레벨에서 수신할 수도 있다.

[0145] 비디오 슬라이스가 인트라-코딩된 (I) 슬라이스로서 코딩될 때, 예측 프로세싱 유닛 (81) 의 인트라 예측 유닛 (84) 은 시그널링된 인트라 예측 모드 및 현재의 프레임 또는 픽처의 이전에 디코딩된 블록들로부터의 데이터에 기초하여, 현재의 비디오 슬라이스의 비디오 블록에 대한 예측 데이터를 발생할 수도 있다. 비디오 프레임이 인터-코딩된 슬라이스 (즉, B 슬라이스 또는 P 슬라이스) 로서 코딩될 때, 예측 프로세싱 유닛 (81) 의 모션 보상 유닛 (82) 은 엔트로피 디코딩 유닛 (80) 으로부터 수신된 모션 벡터들 및 다른 신택스 엘리먼트들에 기초하여 현재의 비디오 슬라이스의 비디오 블록에 대한 예측 블록들을 발생한다. 예측 블록들은 참조 픽처 리스트들 중 하나 내 참조 픽처들 중 하나로부터 발생될 수도 있다. 비디오 디코더 (30) 는 디폴트 구성 기법들을 이용하여, DPB (94) 에 저장된 참조 픽처들에 기초하여, 참조 프레임 리스트들, 즉, List 0 및 List 1 을 구성할 수도 있다.

[0146] 모션 보상 유닛 (82) 은 모션 벡터들 및 다른 신택스 엘리먼트들을 파싱하여 현재의 비디오 슬라이스의 비디오 블록에 대한 예측 정보를 결정하고, 그리고, 그 예측 정보를 이용하여, 디코딩중인 현재의 비디오 블록에 대한 예측 블록들을 발생한다. 예를 들어, 모션 보상 유닛 (82) 은 그 수신된 신택스 엘리먼트들 중 일부를 이용하여, 비디오 슬라이스의 비디오 블록들을 코딩하는 사용되는 예측 모드 (예컨대, 인트라- 또는 인터-예측), 인터-예측 슬라이스 유형 (예컨대, B 슬라이스 또는 P 슬라이스), 슬라이스에 대한 참조 픽처 리스트들 중 하나 이상에 대한 구성 정보, 슬라이스의 각각의 인터-인코딩된 비디오 블록에 대한 모션 벡터들, 슬라이스의 각각의 인터-코딩된 비디오 블록에 대한 인터-예측 상태, 및 다른 정보를 결정하여, 현재의 비디오 슬라이스에서의 비디오 블록들을 디코딩한다.

[0147] 모션 보상 유닛 (82) 은 또한 내삽 필터들에 기초하여 내삽을 수행할 수도 있다. 모션 보상 유닛 (82) 은

비디오 블록들의 인코딩 동안 비디오 인코더 (20) 에 의해 사용되는 것과 같은 내삽 필터들을 이용하여, 참조 블록들의 서브-정수 픽셀들에 대해 내삽된 값들을 계산할 수도 있다. 이 경우, 모션 보상 유닛 (82) 은 수신된 신택스 엘리먼트들로부터 비디오 인코더 (20) 에 의해 사용되는 내삽 필터들을 결정하고 그 내삽 필터들을 이용하여 예측 블록들을 발생할 수도 있다.

[0148] 역양자화 유닛 (86) 은 비트스트림으로 제공되어 엔트로피 디코딩 유닛 (80) 에 의해 디코딩되는 양자화된 변환 계수들을 역양자화한다, 즉 양자화 해제한다. 역양자화 프로세스는 양자화의 정도와, 마찬가지로, 적용되어야 하는 역양자화의 정도를 결정하기 위해, 비디오 슬라이스에서의 각각의 비디오 블록에 대해 비디오 인코더 (20) 에 의해 계산된 양자화 파라미터의 사용을 포함할 수도 있다. 역변환 프로세싱 유닛 (88) 은 픽셀 도메인에서 잔차 블록들을 발생하기 위해 역변환, 예컨대, 역 DCT, 역 정수 변환, 또는 개념적으로 유사한 역변환 프로세스를 변환 계수들에 적용한다.

[0149] 예측 프로세싱 유닛이 예를 들어, 인트라 또는 인터 예측을 이용하여 현재의 비디오 블록에 대한 예측 블록을 발생시킨 후, 비디오 디코더 (30) 는 역변환 프로세싱 유닛 (88) 으로부터의 잔차 블록들을 모션 보상 유닛 (82) 에 의해 발생된 대응하는 예측 블록들과 합산함으로써, 복원된 비디오 블록을 형성한다. 합산기 (90) 는 이 합산 동작을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. ALF 유닛 (92) 은 예를 들어, 본 개시물에서 설명된 필터 기법들 중 하나 이상을 이용하여, 복원된 비디오 블록을 필터링한다.

[0150] 도 5 에 명시적으로 나타내지는 않았지만, 비디오 디코더 (30) 는 또한 디블록킹 필터, SAO 필터, 또는 다른 유형들의 필터들 중 하나 이상을 포함할 수도 있다. (코딩 루프 중에 또는 코딩 루프 이후에) 다른 루프 필터들이 또한 픽셀 전환들 (pixel transitions) 을 평활화하거나 또는 아니면 비디오 품질을 향상시키기 위해 사용될 수도 있다. 주어진 프레임 또는 픽처에서의 디코딩된 비디오 블록들은 그후 DPB (94) 에 저장되며, 이 메모리는 후속 모션 보상을 위해 사용되는 참조 픽처들을 저장한다. DPB (94) 는 도 1 의 디스플레이 디바이스 (32) 와 같은 디스플레이 디바이스 상에의 추후 프리젠테이션을 위해, 디코딩된 비디오를 저장하는 추가적인 메모리의 일부이거나 또는 그와 분리될 수도 있다.

[0151] ALF 유닛 (92) 은, 단독으로 또는 비디오 디코더 (30) 의 다른 컴포넌트들과 함께, 청구범위 섹션에서 뿐만 아니라 다른 어딘가에서 설명되는 기법들을 포함한, 본 개시물에서 설명되는 다양한 기법들을 수행하도록 구성될 수도 있다. 예를 들어, 합산기 (90) 의 출력은 디코딩된 비디오 데이터이다. 따라서, ALF 유닛 (92) 은 디코딩된 비디오 데이터를, 본 개시물의 기법들에 따라서 필터링할 수도 있다. 특히, ALF 유닛 (92) 은, ALF 유닛 (92) 이 후속 픽처들을 예측하기 위해 예측 프로세싱 유닛 (81) 에 의해 사용될, DPB (94) 에 저장된, 참조 픽처들로서 이후 사용되는 디코딩된 픽처들을 필터링한다는 점에서, "인-루프" 필터링 프로세스로서 필터링을 수행한다.

[0152] ALF 유닛 (92) 은 본 개시물의 기법들을 단독으로 또는 조합하여 적용할 수도 있다. 예를 들어, ALF 유닛 (92) 은 필터 계수들의 복수의 세트들을 명시적으로 디코딩하고, 필터 계수들의 복수의 세트들로의, 블록들의 클래스들의 제 1 세트에 대한 인덱스들을 디코딩하고, 그리고, 슬라이스 헤더 또는 픽처 파라미터 세트와 같은, 비트스트림으로부터의 디코딩된 데이터에 기초하여, 고정 필터들 또는 DPB (94) 내 하나 이상의 참조 픽처들의 필터들로부터 블록들의 클래스들의 제 2 세트의 필터들에 대한 필터 계수들을 예측 디코딩할 수도 있다. 대안적으로, ALF 유닛 (92) 은 고정 필터로부터 또는 DPB (94) 내 하나 이상의 참조 픽처들의 필터로부터 예측된 필터 계수들의 복수의 세트들의 각각을 디코딩하고, 그후, 블록들의 클래스들의 각각에 대해, 슬라이스 헤더 또는 픽처 파라미터 세트와 같은, 비트스트림으로부터의 디코딩된 데이터에 기초하여, 필터 계수들의 세트들 중 하나를 식별하는 인덱스를 디코딩할 수도 있다. 또한, 다른 예로서, ALF 유닛 (92) 은 상기 기법들 중 임의의 기법을 이용하여 블록들의 클래스들의 제 1 세트에 대한 필터 계수들을 디코딩하고, 그리고, 슬라이스 헤더 또는 픽처 파라미터 세트와 같은, 비트스트림으로부터의 디코딩된 데이터에 기초하여, 기하학적 변환을 식별하는 데이터 및 필터 계수들을 전승할 블록들의 클래스들의 제 1 세트 중 하나를 식별하는 데이터를 포함한, 블록들의 클래스들의 제 2 세트에 대한 기하학적 변환 정보를 디코딩할 수도 있다. 따라서, 일반적으로,

[0153] 일부 예들에서, ALF 유닛 (92) 은 블록들의 다양한 클래스들에 대한 필터들을 구성할 수도 있다. ALF 유닛 (92) 은 예를 들어, 블록의 픽셀들로부터 계산된 기울기의 방향성에 기초하여 블록을 분류할 수도 있다. 일반적으로, ALF 유닛 (92) 은 블록의 각각의 클래스에 대해 상이한 ALF 를 구성할 수도 있다. 그러나, 일부의 경우, ALF 유닛 (92) 은 블록들의 2개 이상의 클래스들에 의해 사용되는 필터 계수들의 세트를 계산할 수도 있다. 예를 들어, ALF 유닛 (92) 은 필터 계수들의 복수의 세트들을 계산할 수도 있다. ALF 유닛 (92) 은 필터 계수들의 복수의 세트들에서 하나의 필터 계수들의 세트를 식별하는 블록들의 클래스들의 각각 (또는,

블록들의 클래스들의 서브세트) 에 대한 인덱스 값을 추가로 디코딩할 수도 있다. 엔트로피 디코딩 유닛 (80) 은 블록들의 클래스들의 각각에 대한 인덱스들을 규정하는 데이터 구조를 엔트로피 디코딩하고, 그 데이터 구조를 ALF 유닛 (92) 에 제공할 수도 있다. 엔트로피 디코딩 유닛 (56) 은 예컨대, 슬라이스 헤더, 픽처 파라미터 세트, 또는 다른 이러한 데이터 구조로부터 데이터 구조를 엔트로피 디코딩할 수도 있다. 따라서, ALF 유닛 (92) 이 블록들의 2개 이상의 클래스들에 대해 동일한 인덱스를 디코딩하는 경우, 이들 블록들의 클래스들은 필터 계수들의 동일한 세트, 즉, 디코딩된 인덱스 값에 대응하는 필터 계수들의 세트를 사용할 것이다.

[0154] 추가적으로 또는 대안적으로, ALF 유닛 (92) 은 블록의 적어도 하나의 클래스에 대해, 블록의 상이한 클래스와 동일한 필터 계수들을 이용하지만, 필터의 필터 서포트 영역 또는 필터 계수들에 적용되는 기하학적 변환을 이용함으로써, 필터가 발생될 것이라는 것을 표시할 수도 있다. 기능적으로, 기하학적 변환을 필터 서포트 영역에 적용하는 것은 기하학적 변환을 필터 계수들에 적용하는 것과 동일한 수학적 결과들을 산출하며, 따라서, ALF 유닛 (92) 은 기하학적 변환을 필터 서포트 영역 또는 필터 계수들에 적용할 수도 있다. 이 예에서, ALF 유닛 (92) 은 블록의 현재의 클래스에 대해, 기하학적 변환 유형 (예컨대, 회전, 대각선 플립, 또는 수직 플립) 뿐만 아니라, 필터 계수들이 블록의 현재의 클래스에 대한 발생된 필터에 사용될 이전에 구성된 필터를 식별하는 데이터를 디코딩할 수도 있다. 또, 엔트로피 디코딩 유닛 (80) 은 이러한 데이터를 엔트로피 디코딩하고 디코딩된 데이터를 ALF 유닛 (92) 에 제공할 수도 있다. 데이터는 예를 들어, 슬라이스 헤더 또는 픽처 파라미터 세트와 같은, 데이터 구조에 포함될 수도 있다.

[0155] 추가적으로 또는 대안적으로, 엔트로피 디코딩 유닛 (80) 은 고정 필터가 블록의 클래스에 대한 필터 계수들의 세트를 예측하는데 사용되는지 여부를 표시하는 (예컨대, 플래그의) 값을 엔트로피 디코딩할 수도 있다. 이와 유사하게, 엔트로피 디코딩 유닛 (80) 은 이 값들을 ALF 유닛 (92) 에 제공할 수도 있다. 엔트로피 디코딩 유닛 (80) 은 슬라이스 헤더, 픽처 파라미터 세트, 또는 다른 이러한 데이터 구조의 부분으로부터 데이터를 디코딩할 수도 있다. 플래그는 예를 들어, 블록의 클래스의 필터에 대한 필터 계수들이 고정 필터로부터 예측되거나, 또는 대신에, 참조 픽처의 필터로부터 예측된다 (또는, 전승된다) 는 것을 표시할 수도 있다. 일부 예들에서, ALF 유닛 (92) 은 블록의 각각의 클래스에 대한 복수의 고정 필터들로 구성될 수도 있다. 이러한 예들에서, ALF 유닛 (92) 은 그 값이 필터 계수들이 고정 필터로부터 예측된다는 것을 표시할 때, 비디오 코더는, 블록의 클래스에 대한 현재의 필터의 필터 계수들을 예측하는데 사용될 복수의 고정 필터들 중 하나를 식별하는, 복수의 고정 필터들로의 인덱스를 추가로 디코딩할 수도 있다.

[0156] 일부 예들에서, ALF 유닛 (92) 은 클래스들 중 임의의 클래스에 대한 필터 계수들 중 어느 것도 임의의 고정 필터로부터 예측되지 않는다는 것, 모든 필터 계수들이 고정 필터로부터 예측된다는 것, 또는 고정 필터로부터 예측되는 일부 필터 계수들이 존재하지만 다른 필터 계수들은 고정 필터로부터 예측되지 않는다는 것을 표시하는 엔트로피 디코딩된 정보를 추가로 수신할 수도 있다. 필터 계수들 중 어느 것도 고정 필터로부터 예측되지 않을 때, ALF 유닛 (92) (또는, 엔트로피 디코딩 유닛 (80)) 은 복수의 고정 필터들로의 인덱스를 나타내는 데이터의 디코딩을 회피할 (예컨대, 생략하거나 또는 방지할) 수도 있다. 한편, 필터 계수들 중 일부 또는 모두가 고정 필터로부터 예측될 때, ALF 유닛 (92) 은 고정 필터로부터 예측되는 이들 필터 계수들의 세트들에 대한, 개별 복수의 고정 필터들로의 인덱스들을 디코딩할 수도 있으며, 여기서, 인덱스들은 대응하는 필터 계수들이 예측될 참조 고정 필터를 식별한다. 더욱이, ALF 유닛 (92) 은 인덱스들에 의해 식별된 고정 필터들을 이용하여 대응하는 필터 계수들을 예측할 수도 있다. 예측 후, ALF 유닛 (92) 은 예측된 필터 계수들과 실제 필터 계수 값들 사이의 차이들을 나타내는 잔차 값들 (예컨대, 오프셋 값들) 을 추가로 디코딩할 수도 있다.

[0157] 도 6 은 비디오 인코더에 의해 (예컨대, 비디오 인코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다. 예 및 설명의 목적들을 위해, 도 6 의 방법은 도 4 의 비디오 인코더 (20) 와 관련하여 설명된다.

[0158] 먼저, 비디오 인코더 (20) 는 픽처를 인코딩한다 (100). 예를 들어, 비디오 인코더 (20) 는 도 4 와 관련하여 위에서 설명한 바와 같이, 픽처의 블록들을 인트라- 또는 인터-예측하고, 예측된 블록들에 대한 잔차들을 계산하고, 잔차 데이터를 변환 및 양자화하고, 그리고, 픽처에 대한 예측 정보 및 양자화된 변환 계수들을 엔트로피 인코딩할 수도 있다. 그후, 비디오 인코더 (20) 는 예컨대, 양자화된 변환 계수들을 역양자화하고 역변환하여 잔차 값들을 복원하고 그후 잔차 값들을 예측된 블록들에 가산함으로써, 픽처를 디코딩할 수도 있다 (102).

[0159] ALF 유닛 (64) 은 그후 블록들의 다양한 클래스들에 대한 필터 계수들을 계산할 수도 있다 (104). 예를 들어, ALF 유닛 (64) 은 블록들에 대한 잠재적인 기울기들에 기초하여 블록들의 클래스들을 결정할 수도 있다.

ALF 유닛 (64) 은 그후, 계산된 필터 계수들로부터 블록들의 클래스들에 대한 필터들을 발생시킬 수도 있다 (106). ALF 유닛 (64) 은 본 개시물의 필터들을 발생시키는 다양한 기법들 중 임의의 기법을 이용할 수도 있다. 예를 들어, 블록들의 적어도 일부 클래스들에 대해, ALF 유닛 (64) 은 계산된 필터 계수들의 복수의 세트들로의 인덱스들을 선택할 수도 있으며, 여기서, 블록의 각각의 클래스에 대한 인덱스는 블록의 대응하는 클래스에 대한 필터를 발생시키는데 사용할 계산된 필터 계수들의 세트를 식별한다. 추가적으로 또는 대안적으로, 블록들의 적어도 일부 클래스들에 대해, ALF 유닛 (64) 은 사용될 이전에 발생된 필터 뿐만 아니라, 필터 서포트 영역 또는 이전에 발생된 필터의 필터 계수들 중 적어도 하나에 적용될 기하학적 변환을 표시하는 데이터를 발생시킬 수도 있다. 추가적으로 또는 대안적으로, 블록들의 적어도 일부 클래스들에 대해, ALF 유닛 (64) 은 고정 필터 또는 참조 픽처의 필터로부터 필터 계수들을 예측하고, 필터 계수들이 예측될 고정 필터 또는 참조 픽처를 표시하는 데이터를 발생시킬 수도 있다.

[0160] 더욱이, 비디오 인코더 (20) 는 블록들의 다양한 클래스들에 대한 필터들을 정의하는 데이터를 인코딩할 수도 있다 (108). 예를 들어, 필터들이 필터 계수들의 복수의 세트들로의 인덱스들을 이용하여 발생되는 블록의 임의의 클래스에 대해, 엔트로피 인코딩 유닛 (56) 은 이들 블록들의 클래스들에 대한 인덱스들을 엔트로피 인코딩할 수도 있다. 추가적으로 또는 대안적으로, 필터들이 이전에 발생된 필터들 및 기하학적 변환을 이용하여 발생되는 블록의 임의의 클래스에 대해, 엔트로피 인코딩 유닛 (56) 은 이전에 발생된 필터들 및 적용될 기하학적 변환 (예컨대, 회전, 수직 플립, 또는 대각선 플립) 을 식별하는 데이터를 엔트로피 인코딩할 수도 있다. 추가적으로 또는 대안적으로, 필터들이 고정 필터로부터 예측된 필터 계수들을 이용하여 발생되는 블록의 임의의 클래스에 대해, 엔트로피 인코딩 유닛 (56) 은 필터 계수들이 고정 필터 또는 참조 픽처의 필터 뿐만 아니라, 고정 필터 또는 참조 픽처의 표시로부터 예측되는지 여부를 표시하는 데이터를 엔트로피 인코딩할 수도 있다.

[0161] ALF 유닛 (64) 은 또한 대응하는 필터들을 이용하여 디코딩된 픽처의 블록들을 필터링할 수도 있다 (110). 즉, 픽처의 각각의 블록에 대해, ALF 유닛 (64) 은 기울기를 계산하고, 기울기에 기초하여 블록에 대한 클래스를 결정할 수도 있다. ALF 유닛 (64) 은 클래스에 대응하는 필터를 추가로 선택할 수도 있다. ALF 유닛 (64) 은 그후 블록에 대한 클래스에 대응하는 필터를 이용하여 블록의 각각의 픽셀을 필터링할 수도 있다.

[0162] 이와 같이, 도 6 의 방법은, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 단계로서, 상기 복수의 필터들을 구성하는 단계는 클래스들의 각각에 대해, 고정 필터가 클래스의 필터 계수들의 세트를 예측하는데 사용되는지 여부를 결정하는 단계, 및 고정 필터가 필터 계수들의 세트를 예측하는데 사용된다고 결정하는 것에 응답하여, 클래스의 고정 필터들의 세트로의 인덱스 값을 결정하는 단계 및 인덱스 값에 의해 식별된 고정 필터들의 세트의 고정 필터를 이용하여 클래스의 필터 계수들의 세트를 예측하는 단계를 포함하는, 상기 복수의 필터들을 구성하는 단계; 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하는 단계; 현재의 블록에 대한 클래스를 결정하는 단계; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하는 단계; 및 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법의 일 예를 나타낸다.

[0163] 도 7 은 비디오 디코더에 의해 (예컨대, 비디오 디코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다. 예 및 설명의 목적들을 위해, 도 6 의 방법은 도 5 의 비디오 디코더 (30) 와 관련하여 설명된다.

[0164] 먼저, 비디오 디코더 (30) 는 픽처를 디코딩한다 (120). 예를 들어, 비디오 디코더 (30) 는 도 5 와 관련하여 위에서 설명한 바와 같이, 픽처에 대한 예측 정보 및 양자화된 변환 계수들을 엔트로피 디코딩하고, 예측 정보에 기초하여 픽처의 블록들을 인트라- 또는 인터-예측하고, 양자화된 변환 계수들을 역양자화 및 역변환하여 잔차 데이터를 복원하고, 그리고 예측된 블록들과 잔차 데이터를 결합하여 블록들을 복원할 수도 있다.

[0165] 엔트로피 디코딩 유닛 (80) 은 블록들의 다양한 클래스들에 대한 필터들을 정의하는 데이터를 추가로 엔트로피 디코딩할 수도 있다 (122). 예를 들어, 엔트로피 디코딩 유닛 (80) 은 필터들이 필터 계수들의 복수의 세트들로의 인덱스들을 이용하여 발생되는 블록들의 클래스들을 표시하는 데이터를 엔트로피 디코딩하고, 이들 블록들의 클래스들에 대한 인덱스들을 엔트로피 디코딩할 수도 있다. 추가적으로 또는 대안적으로, 엔트로피 디코딩 유닛 (80) 은 필터들이 이전에 발생된 필터들 및 기하학적 변환을 이용하여 발생되는 블록의 클래스들을 표시하는 데이터 뿐만 아니라, 이전에 발생된 필터들 및 적용될 기하학적 변환 (예컨대, 회전, 수직 플립, 또는 대각선 플립) 을 식별하는 데이터를 엔트로피 디코딩할 수도 있다. 추가적으로 또는 대안적으로, 엔트로피 디코딩 유닛 (80) 은 필터들이 고정 필터로부터 예측된 필터 계수들을 이용하여 발생되는 블록의 클래스들을 표

시하는 데이터, 및 필터 계수들이 고정 필터 또는 참조 픽처의 필터 뿐만 아니라, 고정 필터 또는 참조 픽처의 표시로부터 예측되는지 여부를 표시하는 데이터를 엔트로피 디코딩할 수도 있다.

[0166] ALF 유닛 (92) 은 그후 블록들의 다양한 클래스들에 대한 필터 계수들을 계산할 수도 있다 (124). 예를 들어, ALF 유닛 (92) 은 블록들에 대한 잠재적인 기울기들에 기초하여 블록들의 클래스들을 결정할 수도 있다. ALF 유닛 (92) 은 그후 위에서 설명한 바와 같이 필터들을 발생시키는 방법을 나타내는 엔트로피 디코딩된 데이터에 기초하여, 계산된 필터 계수들로부터 블록들의 클래스들에 대한 필터들을 발생시킬 수도 있다 (126). ALF 유닛 (92) 은 본 개시물의 필터들을 발생시키는 다양한 기법들 중 임의의 기법을 이용할 수도 있다. 예를 들어, 블록들의 적어도 일부 클래스들에 대해, ALF 유닛 (92) 은 계산된 필터 계수들의 복수의 세트들로의 인덱스들을 엔트로피 디코딩 유닛 (80) 으로부터 수신할 수도 있으며, 여기서, 블록의 각각의 클래스에 대한 인덱스는 블록의 대응하는 클래스에 대한 필터를 발생시키는데 사용할 계산된 필터 계수들의 세트를 식별한다. 추가적으로 또는 대안적으로, 블록들의 적어도 일부 클래스들에 대해, ALF 유닛 (92) 은 사용될 이전에 발생된 필터 뿐만 아니라, 필터 서포트 영역 또는 이전에 발생된 필터의 필터 계수들 중 적어도 하나에 적용될 기하학적 변환을 표시하는 데이터를 엔트로피 디코딩 유닛 (80) 으로부터 수신할 수도 있다. 추가적으로 또는 대안적으로, 블록들의 적어도 일부 클래스들에 대해, ALF 유닛 (92) 은 고정 필터 또는 참조 픽처의 필터로부터 필터 계수들을 예측하고, 엔트로피 디코딩 유닛 (80) 으로부터 수신되는 데이터에 의해 표시되는 바와 같이, 필터 계수들이 예측되는 고정 필터 또는 참조 픽처를 표시하는 데이터를 발생시킬 수도 있다.

[0167] ALF 유닛 (92) 은 그후 대응하는 필터들을 이용하여, 디코딩된 픽처의 블록들을 필터링할 수도 있다 (128). 즉, 픽처의 각각의 블록에 대해, ALF 유닛 (92) 은 기울기를 계산하고, 기울기에 기초하여 블록에 대한 클래스를 결정할 수도 있다. ALF 유닛 (92) 은 클래스에 대응하는 필터를 추가로 선택할 수도 있다. ALF 유닛 (92) 은 그후 블록에 대한 클래스에 대응하는 필터를 이용하여 블록의 각각의 픽셀을 필터링할 수도 있다.

[0168] 이와 같이, 도 7 의 방법은, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 단계로서, 상기 복수의 필터들을 구성하는 단계는 클래스들의 각각에 대해, 고정 필터가 클래스의 필터 계수들의 세트를 예측하는데 사용되는지 여부를 결정하는 단계, 및 고정 필터가 필터 계수들의 세트를 예측하는데 사용된다고 결정하는 것에 응답하여, 클래스의 고정 필터들의 세트로의 인덱스 값을 결정하는 단계 및 인덱스 값에 의해 식별된 고정 필터들의 세트의 고정 필터를 이용하여 클래스의 필터 계수들의 세트를 예측하는 단계를 포함하는, 상기 복수의 필터들을 구성하는 단계; 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하는 단계; 현재의 블록에 대한 클래스를 결정하는 단계; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하는 단계; 및 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법의 일 예를 나타낸다.

[0169] 도 8 은 비디오 인코더에 의해 (예컨대, 비디오 인코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다. 예 및 설명의 목적들을 위해, 도 8 의 방법은 도 4 의 비디오 인코더 (20) 와 관련하여 설명된다.

[0170] 먼저, 비디오 인코더 (20) 는 픽처를 인코딩한다 (140). 예를 들어, 비디오 인코더 (20) 는 도 4 와 관련하여 위에서 설명한 바와 같이, 픽처의 블록들을 인트라- 또는 인터-예측하고, 예측된 블록들에 대한 잔차들을 계산하고, 잔차 데이터를 변환 및 양자화하고, 그리고, 픽처에 대한 예측 정보 및 양자화된 변환 계수들을 엔트로피 인코딩할 수도 있다. 그후, 비디오 인코더 (20) 는 예컨대, 양자화된 변환 계수들을 역양자화하고 역변환하여 잔차 값들을 복원하고 그후 잔차 값들을 예측된 블록들에 가산함으로써, 픽처를 디코딩할 수도 있다 (142).

[0171] ALF 유닛 (64) 은 그후 픽처의 블록의 클래스에 대한 (예컨대, 적응 루프 필터링 (ALF) 에 따른) 필터를 선택할 수도 있다 (144). ALF 유닛 (64) 은 예컨대, 블록의 하나 이상의 픽셀들에 대해 계산된 기울기에 기초하여, 클래스를 선택할 수도 있다. 대안적으로, ALF 유닛 (64) 은 블록의 다른 특성들에 기초하여 블록에 대한 클래스를 결정할 수도 있다.

[0172] 본 개시물의 기법들에 따르면, ALF 유닛 (64) 은 블록의 기울기를 계산할 수도 있다 (146). ALF 유닛 (64) 은 계산된 기울기에 기초하여 기하학적 변환을 추가로 결정하고 수행할 수도 있다 (148). 예를 들어, ALF 유닛 (64) 은 위에서 설명한 바와 같은 테이블 1 의 기법들에 따라서 기하학적 변환을 결정할 수도 있다. ALF 유닛 (64) 은 기하학적 변환을 선택된 필터의 계수들에, 또는 선택된 필터의 필터 서포트 영역 (예컨대, 선택된 필터를 이용하여 필터링될 픽셀에 이웃하는 픽셀들) 에 적용할 수도 있다. ALF 유닛 (64) 은 그후, 선택된 필터를 이용하여, 디코딩된 픽처의 블록을 필터링할 수도 있다 (150). 즉, ALF 유닛 (64) 은 선택된

필터를 이용하여 블록의 각각의 픽셀을 필터링한다.

[0173] 이와 같이, 도 8 의 방법은 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하는 단계; 현재의 블록의 픽셀들을 필터링하는데 사용될 (예컨대, 적응 루프 필터링 (ALF) 에 따른) 필터를 선택하는 단계; 선택된 필터의 필터 서포트 영역 또는 계수들 중 하나에 대해 수행될 기하학적 변환을 선택하는 단계; 선택된 필터의 필터 서포트 영역 또는 계수들에 대해 기하학적 변환을 수행하는 단계; 및 기하학적 변환을 수행한 후, 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법의 일 예를 나타낸다.

[0174] 도 9 는 비디오 디코더에 의해 (예컨대, 비디오 디코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다. 예 및 설명의 목적들을 위해, 도 9 의 방법은 도 5 의 비디오 디코더 (30) 와 관련하여 설명된다.

[0175] 먼저, 비디오 디코더 (30) 는 예컨대, 예측을 위한 신택스 엘리먼트들의 값들 및 양자화된 변환 계수들을 엔트로피 디코딩하고, 양자화된 변환 계수들을 역양자화 및 역변환하여 잔차 값들을 복원하고, 그후 잔차 값들을 예측된 블록들에 가산함으로써, 픽처를 디코딩한다 (160). ALF 유닛 (92) 은 그후 픽처의 블록의 클래스에 대한 (예컨대, 적응 루프 필터링 (ALF) 에 따른) 필터를 선택할 수도 있다 (162). ALF 유닛 (92) 은 예컨대, 블록의 하나 이상의 픽셀들에 대해 계산된 기울기에 기초하여, 클래스를 선택할 수도 있다. 대안적으로, ALF 유닛 (92) 은 블록의 다른 특성들에 기초하여 블록에 대한 클래스를 결정할 수도 있다.

[0176] 본 개시물의 기법들에 따르면, ALF 유닛 (92) 은 블록의 기울기를 계산할 수도 있다 (164). ALF 유닛 (92) 은 계산된 기울기에 기초하여 기하학적 변환을 추가로 결정하고 수행할 수도 있다 (166). 예를 들어, ALF 유닛 (92) 은 위에서 설명한 바와 같은 테이블 1 의 기법들에 따라서 기하학적 변환을 결정할 수도 있다. ALF 유닛 (92) 은 기하학적 변환을 선택된 필터의 계수들에, 또는 선택된 필터의 필터 서포트 영역 (예컨대, 선택된 필터를 이용하여 필터링될 픽셀에 이웃하는 픽셀들) 에 적용할 수도 있다. ALF 유닛 (92) 은 그후, 선택된 필터를 이용하여, 디코딩된 픽처의 블록을 필터링할 수도 있다 (168). 즉, ALF 유닛 (92) 은 선택된 필터를 이용하여 블록의 각각의 픽셀을 필터링한다.

[0177] 이와 같이, 도 9 의 방법은, 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하는 단계; 현재의 블록의 픽셀들을 필터링하는데 사용될 (예컨대, 적응 루프 필터링 (ALF) 에 따른) 필터를 선택하는 단계; 선택된 필터의 필터 서포트 영역 또는 계수들 중 하나에 대해 수행될 기하학적 변환을 선택하는 단계; 선택된 필터의 필터 서포트 영역 또는 계수들에 대해 기하학적 변환을 수행하는 단계; 및 기하학적 변환을 수행한 후, 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법의 일 예를 나타낸다.

[0178] 도 10 은 비디오 인코더에 의해 (예컨대, 비디오 인코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다. 예 및 설명의 목적들을 위해, 도 10 의 방법은 도 4 의 비디오 인코더 (20) 와 관련하여 설명된다.

[0179] 먼저, 비디오 인코더 (20) 는 픽처를 인코딩한다 (170). 예를 들어, 비디오 인코더 (20) 는 도 4 와 관련하여 위에서 설명한 바와 같이, 픽처의 블록들을 인트라- 또는 인터-예측하고, 예측된 블록들에 대한 잔차들을 계산하고, 잔차 데이터를 변환 및 양자화하고, 그리고, 픽처에 대한 예측 정보 및 양자화된 변환 계수들을 엔트로피 인코딩할 수도 있다. 그후, 비디오 인코더 (20) 는 예컨대, 양자화된 변환 계수들을 역양자화하고 역변환하여 잔차 값들을 복원하고, 그후 잔차 값들을 예측된 블록들에 가산함으로써, 픽처를 디코딩할 수도 있다 (172).

[0180] ALF 유닛 (64) 은 그후 고정 필터를 이용하여 픽처의 블록의 클래스에 대한 (예컨대, 적응 루프 필터링 (ALF) 에 따른) 필터의 계수들을 예측할 수도 있다 (174). ALF 유닛 (64) 은 예컨대, 블록의 하나 이상의 픽셀들에 대해 계산된 기울기에 기초하여, 클래스를 선택할 수도 있다. 대안적으로, ALF 유닛 (64) 은 블록의 다른 특성들에 기초하여 블록에 대한 클래스를 결정할 수도 있다. 일부 예들에서, 다수의 고정 필터들이 블록의 각각의 클래스에 대해 이용가능할 수도 있으며, ALF 유닛 (64) 은 이용가능한 고정 필터들의 각각을 테스트하여, 어느 것이 최상의 레이트-왜곡 성능을 초래하는지를 결정할 수도 있다. ALF 유닛 (64) 은 그후 블록의 각각의 클래스에 대해 최상의 레이트-왜곡 성능을 산출하는 고정 필터들, 또는 참조 필터가 더 나은 레이트-왜곡 성능을 산출하는 경우 참조 픽처의 참조 필터를 선택할 수도 있다. ALF 유닛 (64) 은 그후 고정 필터들 중 어느 것이 블록의 각각의 클래스에 사용되는지 (그리고, 추가적으로 또는 대안적으로, 특정의 클래스에

대한 필터가 참조 픽처의 참조 필터로부터 예측되는지 여부) 를 표시하는 플래그를 인코딩한다 (176).

[0181] ALF 유닛 (64) 은 그후, 디코딩된 픽처의 블록의 각각의 클래스에 대한 필터들을 선택하고 (178), 예컨대, 블록들에 대해 계산된 기울기들에 기초하여, 디코딩된 픽처의 블록들의 각각에 대한 클래스들을 결정하고 (180), 그리고, 선택된 필터들을 이용하여 블록들을 필터링할 수도 있다 (182). 또, ALF 유닛 (64) 은 블록의 각각의 클래스에 대해 상이한 필터들을 테스트하여, 어느 필터가 최상의 레이트-왜곡 성능을 산출하는지를 결정할 수도 있다. 본 개시물의 기법들에 따르면, 일부 예들에서, ALF 유닛 (64) 은 병합된 클래스들의 각각이 동일한 필터를 이용하도록, 블록들의 2개 이상의 클래스들을 병합할 수도 있다. 더욱이, 비디오 인코더 (20) 는 필터들 중 어느 것이 블록의 각각의 클래스에 사용되는지를 표시하는 데이터를 인코딩할 수도 있다.

[0182] 이와 같이, 도 10 의 방법은, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 단계로서, 상기 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 단계는, 상기 복수의 필터들을 구성하는 단계는, 클래스들의 각각에 대해, 고정 필터가 클래스의 필터 계수들의 세트를 예측하는데 사용되는지 여부를 결정하는 단계, 및 고정 필터가 필터 계수들의 세트를 예측하는데 사용된다고 결정하는 것에 응답하여, 클래스의 고정 필터들의 세트로의 인덱스 값을 결정하는 단계 및 인덱스 값에 의해 식별된 고정 필터들의 세트의 고정 필터를 이용하여 클래스의 필터 계수들의 세트를 예측하는 단계를 포함하는, 상기 복수의 필터들을 구성하는 단계; 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하는 단계; 현재의 블록에 대한 클래스를 결정하는 단계; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하는 단계; 및 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법의 일 예를 나타낸다.

[0183] 도 11 은 비디오 디코더에 의해 (예컨대, 비디오 디코딩 프로세스 동안) 디코딩된 픽처의 블록들을 필터링하는 예시적인 방법을 예시하는 플로우차트이다. 예 및 설명의 목적들을 위해, 도 11 의 방법은 도 5 의 비디오 디코더 (30) 와 관련하여 설명된다.

[0184] 먼저, 비디오 디코더 (30) 는 예컨대, 예측을 위한 신택스 엘리먼트들의 값들 및 양자화된 변환 계수들을 엔트로피 디코딩하고, 양자화된 변환 계수들을 역양자화 및 역변환하여 잔차 값들을 복원하고, 그후 잔차 값들을 예측된 블록들에 가산함으로써, 비디오 데이터의 픽처를 디코딩한다 (190).

[0185] ALF 유닛 (92) 은 그후 필터의 계수들이 고정 필터로부터 예측되는지 여부를 표시하는 플래그를 디코딩할 수도 있으며 (192), 그리고, 일부 예들에서는, 필터 계수들이 고정 필터로부터 예측되는 경우, 고정 필터를 식별하는 데이터를 디코딩할 수도 있다. ALF 유닛 (92) 은 그후, (이 예에서는, 필터 계수들이 고정 필터로부터 예측된다고 가정하여) 고정 필터를 이용하여 필터의 계수들을 예측할 수도 있다 (194). 다른 예들에서, ALF 유닛 (92) 은 필터 계수들이 고정 필터로부터 예측되지 않는다는 것을 플래그가 표시하면, 참조 픽처의 참조 필터로부터 필터 계수들을 예측할 수도 있다.

[0186] ALF 유닛 (92) 은 그후, 디코딩된 픽처의 블록의 각각의 클래스에 대한 필터들을 선택하고 (196), 예컨대, 블록들에 대해 계산된 기울기들에 기초하여, 디코딩된 픽처의 블록들의 각각에 대한 클래스들을 결정하고 (198), 그리고 선택된 필터들을 이용하여 블록들을 필터링할 수도 있다 (200). ALF 유닛 (92) 은 그 클래스의 블록들의 픽셀들을 필터링하는데 사용될 필터를 식별하는, 블록의 각각의 클래스에 대한 데이터를 디코딩할 수도 있다.

[0187] 이와 같이, 도 11 의 방법은, 비디오 데이터의 현재의 픽처의 블록들의 클래스들에 대한 복수의 필터들을 구성하는 단계로서, 상기 복수의 필터들을 구성하는 단계는 클래스들의 각각에 대해, 고정 필터가 클래스의 필터 계수들의 세트를 예측하는데 사용되는지 여부를 결정하는 단계, 및 고정 필터가 필터 계수들의 세트를 예측하는데 사용된다고 결정하는 것에 응답하여, 클래스의 고정 필터들의 세트로의 인덱스 값을 결정하는 단계 및 인덱스 값에 의해 식별된 고정 필터들의 세트의 고정 필터를 이용하여 클래스의 필터 계수들의 세트를 예측하는 단계를 포함하는, 상기 복수의 필터들을 구성하는 단계; 비디오 데이터의 현재의 픽처의 현재의 블록을 디코딩하는 단계; 현재의 블록에 대한 클래스를 결정하는 단계; 복수의 필터들 중, 현재의 블록에 대한 클래스에 대응하는 필터를 선택하는 단계; 및 선택된 필터를 이용하여 현재의 블록의 적어도 하나의 픽셀을 필터링하는 단계를 포함하는, 비디오 데이터의 디코딩된 블록을 필터링하는 방법의 일 예를 나타낸다.

[0188] 하나 이상의 예들에서, 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현되는 경우, 그 기능들은 하나 이상의 명령들 또는 코드로서, 컴퓨터-판독가능 매체 상에 저장되거나 또는 컴퓨터-판독가능 매체를 통해서 송신될 수도 있으며, 하드웨어-기반의 프로세싱 유

닛에 의해 실행될 수도 있다. 컴퓨터-판독가능 매체는 데이터 저장 매체들과 같은 유형의 매체에 대응하는 컴퓨터-판독가능 저장 매체들, 또는 예를 들어, 통신 프로토콜에 따라서 한 장소로부터 다른 장소로의 컴퓨터 프로그램의 전송을 용이하게 하는 임의의 매체를 포함하는 통신 매체들을 포함할 수도 있다. 이런 방법으로, 컴퓨터-판독가능 매체들은 일반적으로 (1) 비일시성 유형의 컴퓨터-판독가능 저장 매체, 또는 (2) 신호 또는 캐리어 파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체는 본 개시물에서 설명하는 기법들의 구현을 위한 명령들, 코드 및/또는 데이터 구조들을 취출하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 가용 매체들일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터-판독가능 매체를 포함할 수도 있다.

[0189] 일 예로서, 이에 한정하지 않고, 이런 컴퓨터-판독가능 저장 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광디스크 스토리지, 자기디스크 스토리지, 또는 다른 자기 저장 디바이스들, 플래시 메모리, 또는 원하는 프로그램 코드를 명령들 또는 데이터 구조들의 형태로 저장하는데 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 접속이 컴퓨터-판독가능 매체로 적절히 지칭된다. 예를 들어, 동축 케이블, 광섬유 케이블, 연선, 디지털 가입자 회선 (DSL), 또는 무선 기술들, 예컨대 적외선, 라디오, 및 마이크로파를 이용하여 명령들이 웹사이트, 서버, 또는 다른 원격 소스로부터 송신되는 경우, 동축 케이블, 광섬유 케이블, 연선, DSL, 또는 무선 기술들 예컨대 적외선, 라디오, 및 마이크로파가 그 매체의 정의에 포함된다. 그러나, 컴퓨터-판독가능 저장 매체 및 데이터 저장 매체는 접속부들, 반송파들, 신호들, 또는 다른 일시성 매체를 포함하지 않고, 그 대신, 비-일시성 유형의 저장 매체로 송신되는 것으로 해석되어야 한다. 디스크 (disk) 및 디스크 (disc) 는, 본원에서 사용할 때, 컴팩트 디스크 (CD), 레이저 디스크, 광 디스크, 디지털 다기능 디스크 (DVD), 플로피 디스크 및 블루-레이 디스크를 포함하며, 디스크들 (disks) 은 데이터를 자기적으로 보통 재생하지만, 디스크들 (discs) 은 레이저로 데이터를 광학적으로 재생한다. 앞에서 언급한 것들의 결합들이 또한 컴퓨터-판독가능 매체들의 범위 내에 포함되어야 한다.

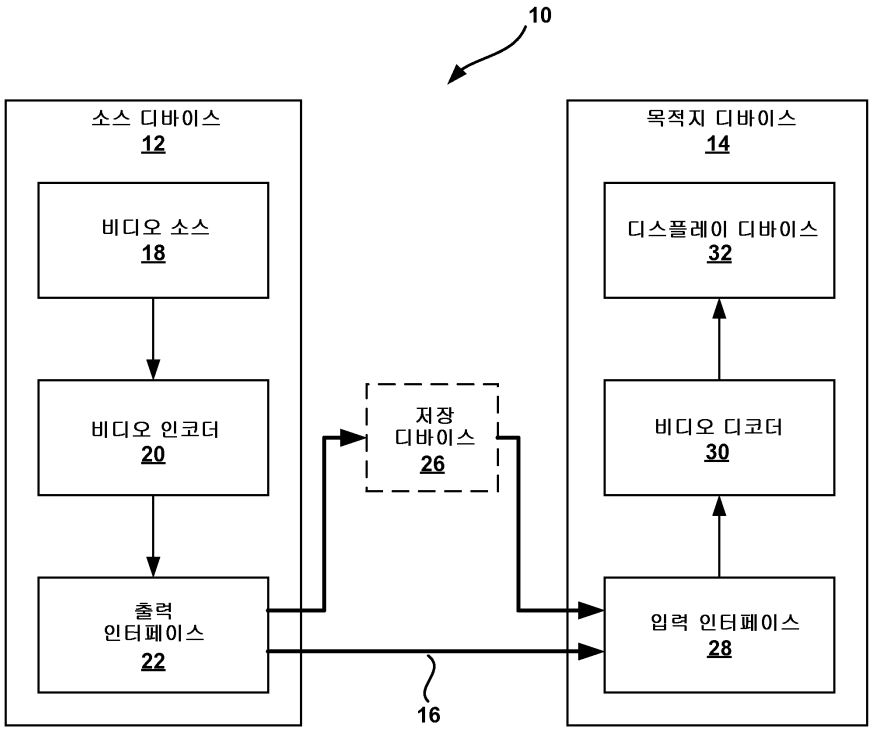
[0190] 명령들은 하나 이상의 디지털 신호 프로세서들 (DSPs), 범용 마이크로프로세서들, 주문형 집적회로들 (ASICs), 필드 프로그래밍가능 로직 어레이들 (FPGAs), 또는 다른 등가의 집적 또는 이산 로직 회로와 같은, 하나 이상의 프로세서들에 의해 실행될 수도 있다. 따라서, 용어 "프로세서" 는, 본원에서 사용될 때 전술한 구조 중 임의의 구조 또는 본원에서 설명하는 기법들의 구현에 적합한 임의의 다른 구조를 지칭할 수도 있다. 게다가, 일부 양태들에서, 본원에서 설명하는 기능은 전용 하드웨어 및/또는 인코딩 및 디코딩을 위해 구성되는 소프트웨어 모듈들 내에 제공되거나, 또는 결합된 코덱에 포함될 수도 있다. 또한, 이 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들로 전적으로 구현될 수 있다. 따라서, 프로세서는 고정된 하드웨어 프로세싱 회로부, 프로그래밍가능 프로세싱 회로부 및/또는 고정된 프로세싱 회로부와 프로그래밍가능 프로세싱 회로부 양쪽의 조합으로서 구현되는, 하나 이상의 프로세서들을 포함하는 다양한 통합된 프로세싱 회로부 중 임의의 프로세싱 회로부로 형성될 수도 있다.

[0191] 본 개시물의 기법들은 무선 핸드셋, 집적 회로 (IC) 또는 IC들의 세트 (예컨대, 칩 세트) 를 포함한, 매우 다양한 디바이스들 또는 장치들로 구현될 수도 있다. 개시한 기법들을 수행하도록 구성되는 디바이스들의 기능적 양태들을 강조하기 위해서 다양한 컴포넌트들, 모듈들, 또는 유닛들이 본 개시물에서 설명되지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 필요로 하지는 않는다. 대신, 위에서 설명한 바와 같이, 다양한 유닛들이 코덱 하드웨어 유닛에 결합되거나 또는 적합한 소프트웨어 및/또는 펌웨어와 함께, 위에서 설명한 바와 같은 하나 이상의 프로세서들을 포함한, 상호작용하는 하드웨어 유닛들의 컬렉션으로 제공될 수도 있다.

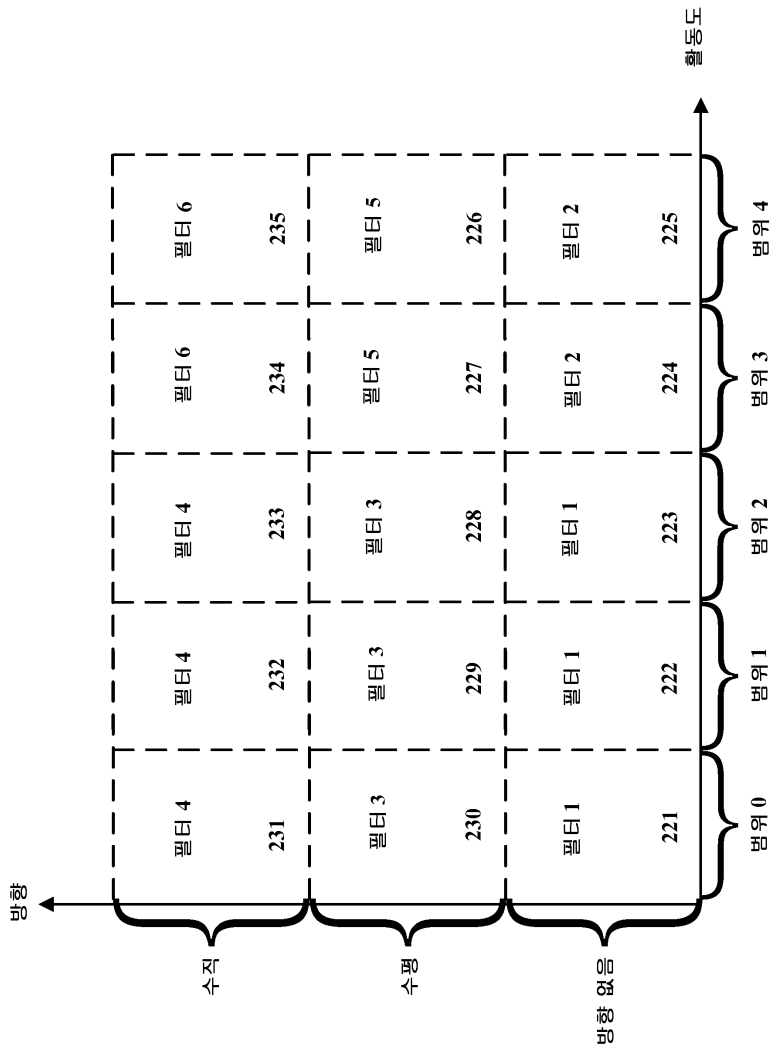
[0192] 다양한 예들이 설명되었다. 이들 및 다른 예들은 다음 청구항들의 범위 이내이다.

도면

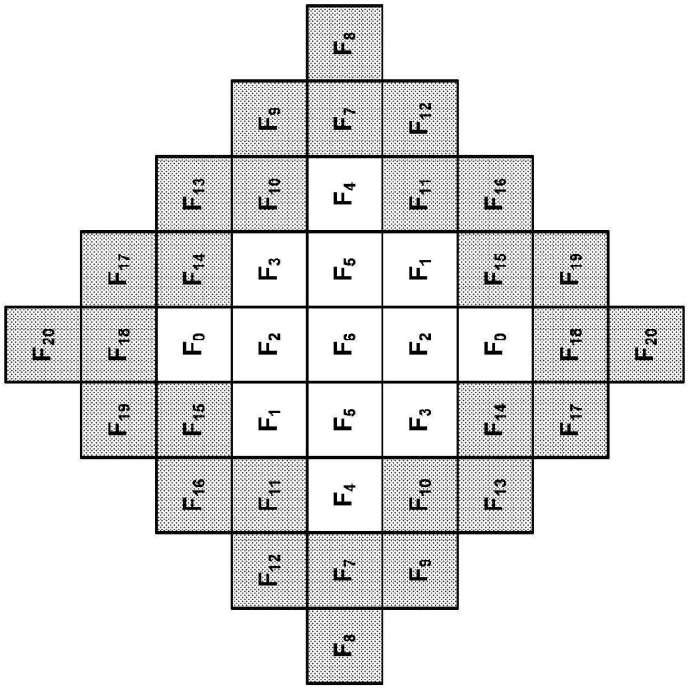
도면1


10
소스 디바이스
12
비디오 소스
18
비디오 인코더
20
출력
인터페이스
22
저장
디바이스
26
16
목적지 디바이스
14
디스플레이 디바이스
32
비디오 디코더
30
입력 인터페이스
28

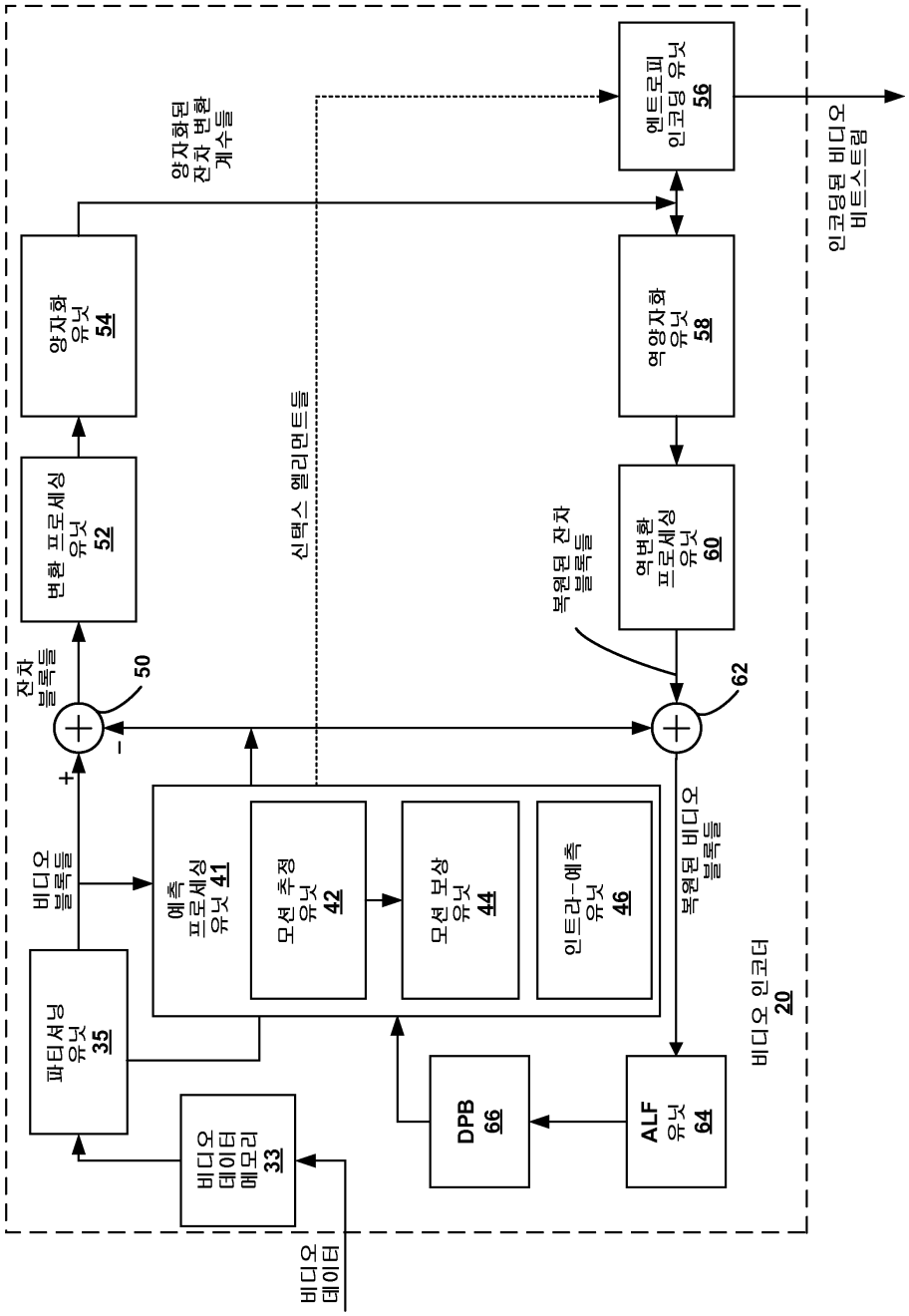
도면2


활동도
방향
수직
수평
방향 없음
범위 0
범위 1
범위 2
범위 3
범위 4
221
222
223
224
225
230
229
228
227
226
231
232
233
234
235
필터 1
필터 2
필터 3
필터 4
필터 5
필터 6

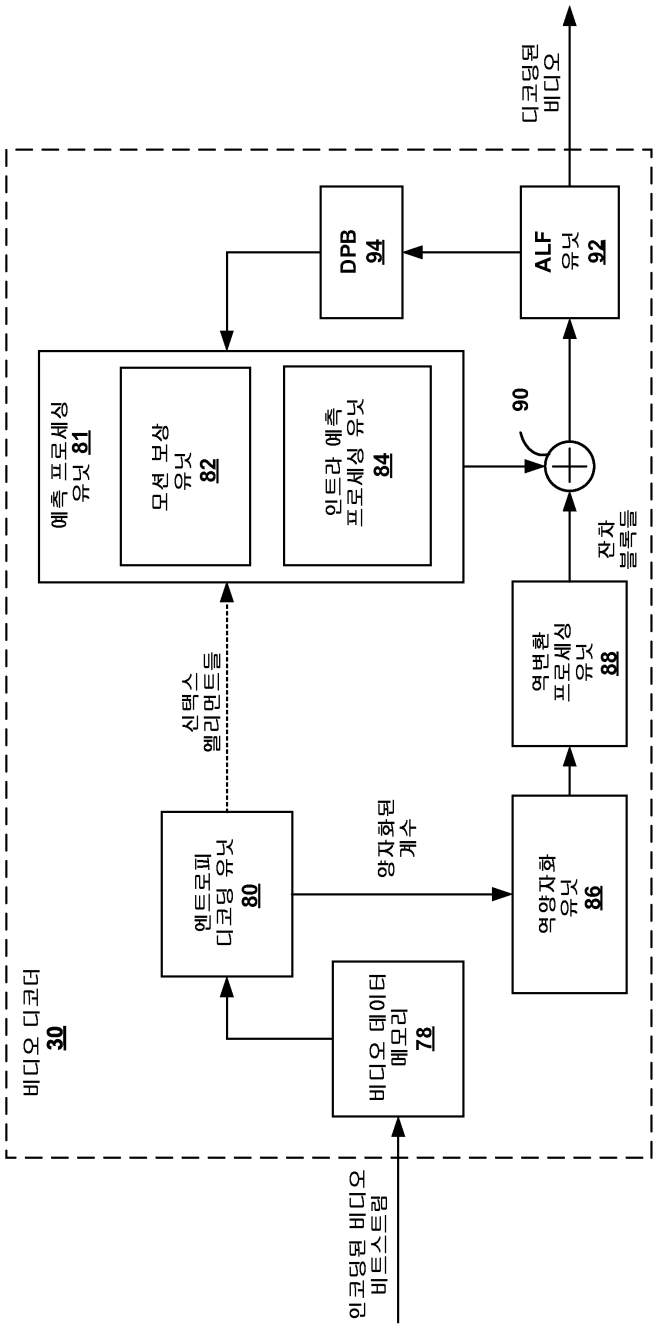
도면3


F8
F9
F7
F12
F13
F10
F4
F11
F16
F17
F14
F3
F5
F1
F15
F19
F20
F18
F0
F2
F6
F2
F0
F18
F20
F19
F15
F1
F5
F3
F14
F17
F16
F11
F4
F10
F13
F12
F7
F8
F8

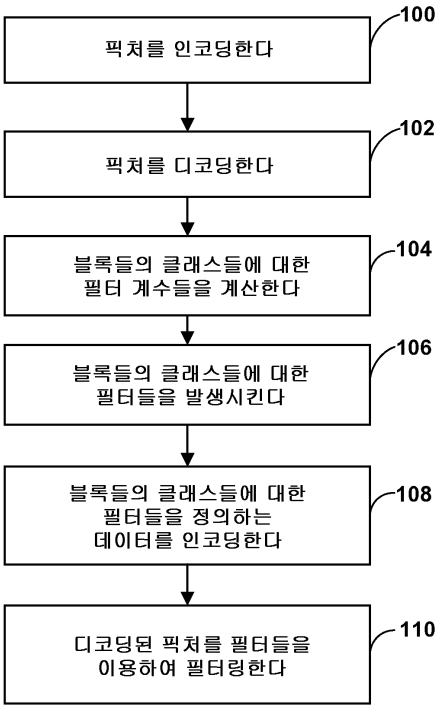
도면4


비디오 인코더
20
비디오 데이터
비디오 데이터 메모리
33
파티셔닝 유닛
35
비디오 블록들
예측 프로세싱 유닛
41
모션 추정 유닛
42
모션 보상 유닛
44
인트라-예측 유닛
46
DPB
66
ALF 유닛
64
복원된 비디오 블록들
잔차 블록들
50
변환 프로세싱 유닛
52
양자화 유닛
54
양자화된 잔차 변환 계수들
신택스 엘리먼트들
엔트로피 인코딩 유닛
56
인코딩된 비디오 비트스트림
역양자화 유닛
58
역변환 프로세싱 유닛
60
복원된 잔차 블록들
62

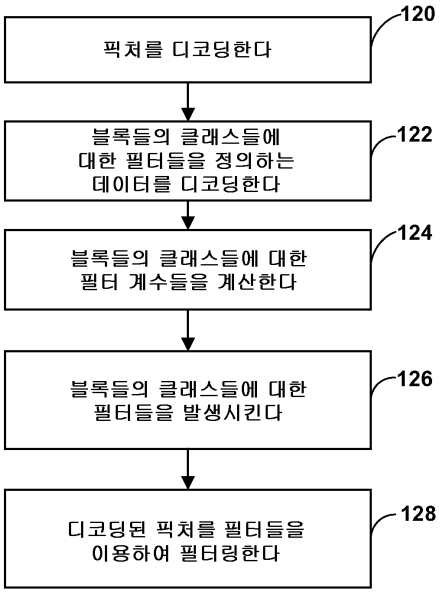
도면5


인코딩된 비디오 비트스트림
비디오 디코더 30
비디오 데이터 메모리 78
엔트로피 디코딩 유닛 80
신택스 엘리먼트들
예측 프로세싱 유닛 81
모션 보상 유닛 82
인트라 예측 프로세싱 유닛 84
양자화된 계수
역양자화 유닛 86
역변환 프로세싱 유닛 88
90
잔차 블록들
DPB 94
ALF 유닛 92
디코딩된 비디오

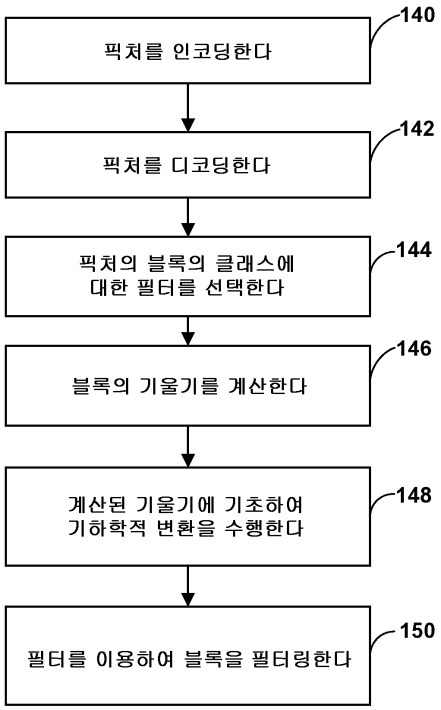
도면6


픽처를 인코딩한다
100
픽처를 디코딩한다
102
블록들의 클래스들에 대한 필터 계수들을 계산한다
104
블록들의 클래스들에 대한 필터들을 발생시킨다
106
블록들의 클래스들에 대한 필터들을 정의하는 데이터를 인코딩한다
108
디코딩된 픽처를 필터들을 이용하여 필터링한다
110

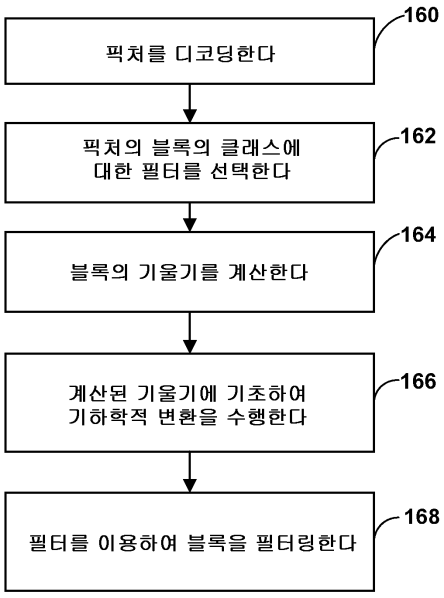
도면7


픽처를 디코딩한다
120
블록들의 클래스들에 대한 필터들을 정의하는 데이터를 디코딩한다
122
블록들의 클래스들에 대한 필터 계수들을 계산한다
124
블록들의 클래스들에 대한 필터들을 발생시킨다
126
디코딩된 픽처를 필터들을 이용하여 필터링한다
128

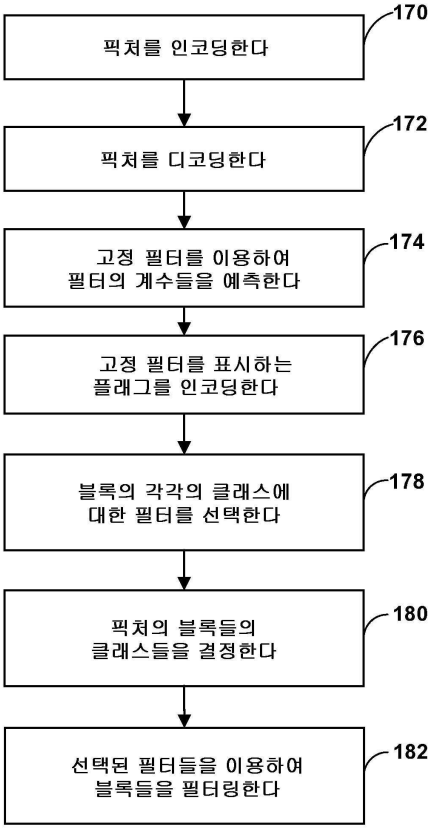
도면8


픽처를 인코딩한다
140
픽처를 디코딩한다
142
픽처의 블록의 클래스에 대한 필터를 선택한다
144
블록의 기울기를 계산한다
146
계산된 기울기에 기초하여 기하학적 변환을 수행한다
148
필터를 이용하여 블록을 필터링한다
150

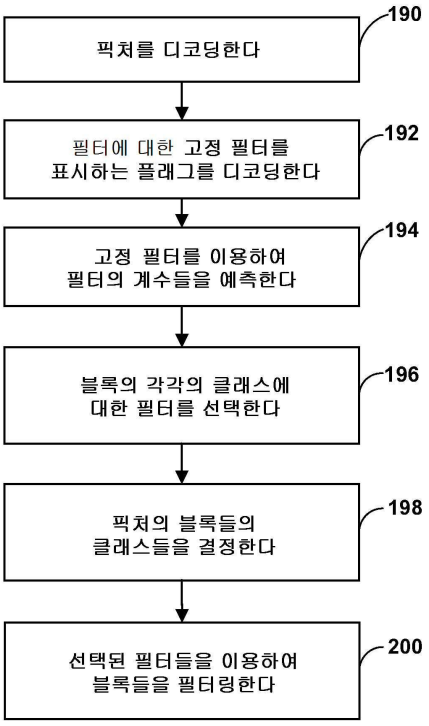
도면9


픽처를 디코딩한다
160
픽처의 블록의 클래스에 대한 필터를 선택한다
162
블록의 기울기를 계산한다
164
계산된 기울기에 기초하여 기하학적 변환을 수행한다
166
필터를 이용하여 블록을 필터링한다
168

도면10


픽쳐를 인코딩한다
170
픽쳐를 디코딩한다
172
고정 필터를 이용하여
필터의 계수들을 예측한다
174
고정 필터를 표시하는
플래그를 인코딩한다
176
블록의 각각의 클래스에
대한 필터를 선택한다
178
픽쳐의 블록들의
클래스들을 결정한다
180
선택된 필터들을 이용하여
블록들을 필터링한다
182

도면11


픽쳐를 디코딩한다
190
필터에 대한 고정 필터를
표시하는 플래그를 디코딩한다
192
고정 필터를 이용하여
필터의 계수들을 예측한다
194
블록의 각각의 클래스에
대한 필터를 선택한다
196
픽쳐의 블록들의
클래스들을 결정한다
198
선택된 필터들을 이용하여
블록들을 필터링한다
200