



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2014년12월30일

(11) 등록번호 10-1477435

(24) 등록일자 2014년12월22일

(51) 국제특허분류(Int. Cl.)

H04N 19/122 (2014.01) H04N 19/157 (2014.01)
H04N 19/42 (2014.01)

(21) 출원번호 10-2013-7003945

(22) 출원일자(국제) 2011년07월08일

심사청구일자 2013년02월15일

(85) 번역문제출일자 2013년02월15일

(65) 공개번호 10-2013-0054354

(43) 공개일자 2013년05월24일

(86) 국제출원번호 PCT/US2011/043462

(87) 국제공개번호 WO 2012/009237

국제공개일자 2012년01월19일

(30) 우선권주장

13/159,063 2011년06월13일 미국(US)

(뒷면에 계속)

(56) 선행기술조사문헌

KR1020090027670 A*

M. Karczewicz, et al. Video coding technology proposal by Qualcomm Inc. JCT-VC of ITU-T SG16 WP3 & ISO/IEC JTC1/SC29/WG11.

JCTVC-A121. 1st Meeting: Dresden, Apr. 15-23, 2010, pp.1-24*

*는 심사관에 의하여 인용된 문헌

(73) 특허권자

퀄컴 인코포레이티드

미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775

(72) 발명자

조쉬 라잔 엘

미국 92121 캘리포니아주 샌디에고 모어하우스 드라이브 5775

치엔 웨이-정

미국 92121 캘리포니아주 샌디에고 모어하우스 드라이브 5775

(뒷면에 계속)

(74) 대리인

특허법인코리아나

전체 청구항 수 : 총 38 항

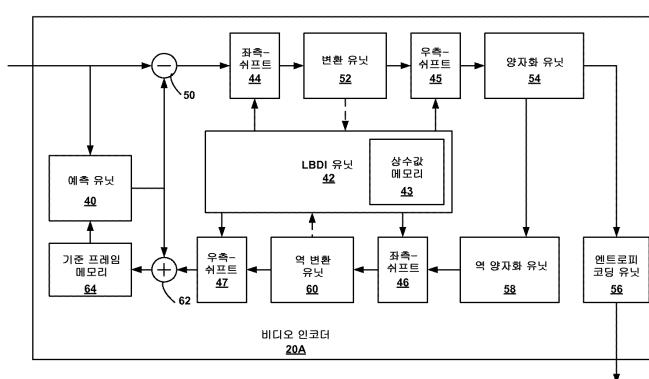
심사관 : 이상래

(54) 발명의 명칭 비디오 코딩에서 고정 소수점 변환들을 위한 가변 로컬화 비트 깊이 증가

(57) 요약

본 개시는 비디오 코딩과 연관된 고정 소수점 변환에서 반올림 오차들을, 그 변환에 가변 로컬화 비트-깊이 증가를 적용함으로써 경감시키는 기술들을 설명한다. 더욱 상세하게는, 그 기술들은 비디오 코딩 디바이스에서 비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상수 값들 중 하나의 상수 값을 선택하고, 상

(뒷면에 계속)

대 표 도

수 값과 동일한 값을 이용하여 변환에 가변 로컬화 비트-깊이 증가를 적용하는 것을 포함한다. 가변 로컬화 비트-깊이 증가를 적용하는 것은 고정 소수점 변환 이전에 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-쉬프트하고, 고정 소수점 변환 이후에 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-쉬프트하는 것을 포함한다. 상수 값은 비디오 코딩 디바이스에 저장된 복수의 상수 값들로부터 선택된다. 상수 값을 각각은 비디오 코딩에 의해 지원되는 복수의 상이한 변환 사이즈들 중 하나에 대하여 미리 계산된다.

(72) 발명자

카르체비츠 마르타

미국 92121 캘리포니아주 샌디에고 모어하우스 드
라이브 5775

레즈닉 유리

미국 92121 캘리포니아주 샌디에고 모어하우스 드
라이브 5775

천 폐이송

미국 92121 캘리포니아주 샌디에고 모어하우스 드
라이브 5775

추양 샤오-창

미국 92121 캘리포니아주 샌디에고 모어하우스 드
라이브 5775

(30) 우선권주장

61/364,784 2010년07월15일 미국(US)

61/370,760 2010년08월04일 미국(US)

특허청구의 범위

청구항 1

비디오 신호를 코딩하는 방법으로서,

복수의 상수 값을 저장하는 단계로서, 상기 상수 값을 각각은 비디오 코딩과 연관된 복수의 상이한 변환 사이즈들 중 하나의 변환 사이즈와 연관되는, 상기 복수의 상수 값을 저장하는 단계;

비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상기 상수 값을 중 하나의 상수 값을 선택하는 단계; 및

상기 비디오 코딩 디바이스에서, 상기 고정 소수점 변환 이전에 상기 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-ши프트하고, 상기 고정 소수점 변환 이후에 상기 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-ши프트함으로써, 상기 상수 값과 동일한 값을 이용하여 상기 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하는 단계를 포함하는, 비디오 신호를 코딩하는 방법.

청구항 2

제 1 항에 있어서,

상기 고정 소수점 변환은 2 차원 고정 소수점 변환을 포함하고,

상기 상수 값을 중 하나의 상수 값을 선택하는 단계는, 상기 2 차원 고정 소수점 변환의 일 스테이지의 사이즈에 기초하여 상기 상수 값을 중 하나의 상수 값을 선택하고, 상기 상수 값을 2 배가 되도록 하는 단계를 포함하며, 그리고

상기 가변 로컬화 비트-깊이 증가를 적용하는 단계는 상기 2 배된 상수 값 이하인 값을 이용하여 상기 2 차원 고정 소수점 변환에 상기 가변 로컬화 비트-깊이 증가를 적용하는 단계를 포함하는, 비디오 신호를 코딩하는 방법.

청구항 3

제 1 항에 있어서,

내부 비트-깊이 증가가 상기 비디오 신호에 적용되는지 여부를 결정하는 단계; 및

상기 내부 비트-깊이 증가가 상기 비디오 신호에 적용될 경우, 상기 내부 비트-깊이 증가의 값을 기초하여 상기 상수 값을 조정하는 단계를 더 포함하며,

상기 가변 로컬화 비트-깊이 증가를 적용하는 단계는, 상기 조정된 상수 값과 동일한 값을 이용하여 상기 고정 소수점 변환에 상기 가변 로컬화 비트-깊이 증가를 적용하는 단계를 포함하는, 비디오 신호를 코딩하는 방법.

청구항 4

제 3 항에 있어서,

상기 상수 값을 조정하는 단계는, 상기 상수 값 N 을 상기 내부 비트-깊이 증가의 값 B_d 만큼 감소시켜 상기 조정된 상수 값이 $N - B_d$ 와 동일하도록 하는 단계를 포함하는, 비디오 신호를 코딩하는 방법.

청구항 5

제 4 항에 있어서,

상기 상수 값을 조정하는 단계는, 상기 내부 비트-깊이 증가의 값 B_d 이 상기 상수 값 N 이하일 경우에, 상기 상수 값 N 을 상기 내부 비트-깊이 증가의 값 B_d 만큼 감소시켜 $(N - B_d) \geq 0$ 이 되도록 하는 단계를 포함하는, 비디오 신호를 코딩하는 방법.

청구항 6

제 4 항에 있어서,

상기 상수 값을 조정하는 단계는, 상기 내부 비트-깊이 증가의 값 B_d 이 상기 상수 값 N 보다 클 경우에, 상기 상수 값 N 을 0 으로 감소시켜 $(N - B_d) < 0$ 이 되도록 하는 단계를 포함하는, 비디오 신호를 코딩하는 방법.

청구항 7

제 1 항에 있어서,

상기 고정 소수점 변환 이후에 상기 상수 값과 동일한 비트 수만큼 상기 변환 출력 신호를 우측-쉬프트하기 전에 상기 변환 출력 신호에 오프셋을 적용하는 단계를 더 포함하는, 비디오 신호를 코딩하는 방법.

청구항 8

제 7 항에 있어서,

상기 상수 값은 N 과 동일하고,

상기 변환 출력 신호에 오프셋을 적용하는 단계는 상기 변환 출력 신호에 2^{N-1} 과 동일한 오프셋을 적용하는 단계를 포함하는, 비디오 신호를 코딩하는 방법.

청구항 9

제 7 항에 있어서,

상기 가변 로컬화 비트-깊이 증가에 대한 오프셋 및 우측-쉬프트와 상기 변환 출력 신호에 적용된 다른 오프셋들 및 우측-쉬프트들을 결합하는 단계를 더 포함하는, 비디오 신호를 코딩하는 방법.

청구항 10

제 1 항에 있어서,

상기 비디오 코딩 디바이스에서 상기 복수의 상수 값을 저장하기 전에, 상기 복수의 상이한 변환 사이즈들 중 하나의 변환 사이즈와 연관된 상수 값을 각각은 상기 비디오 신호의 입력 비트-깊이 값, 상기 변환 사이즈와 연관된 변환 비트-깊이 값, 및 상기 변환 사이즈와 연관된 변환 비트-정확도 값에 기초하여 미리 계산되는, 비디오 신호를 코딩하는 방법.

청구항 11

제 10 항에 있어서,

상기 상수 값들의 각각, N 은 $B_i + B_{TR} + N \leq Z$ 를 만족하도록 미리 계산되고, 상기 B_i 는 상기 비디오 신호의 입력 비트-깊이 값이고, B_{TR} 는 상기 변환 사이즈와 연관된 변환 비트-깊이 값이고, Z 는 상기 변환 사이즈와 연관된 변환 비트-정확도 값인, 비디오 신호를 코딩하는 방법.

청구항 12

제 1 항에 있어서,

상기 복수의 상이한 변환 사이즈들은 정사각 변환 사이즈들 및 직사각 변환 사이즈들 중 하나 이상을 포함하는, 비디오 신호를 코딩하는 방법.

청구항 13

제 12 항에 있어서,

상기 복수의 상이한 변환 사이즈들은 4×4 변환, 4×8 변환, 8×4 변환, 8×8 변환, 8×16 변환, 16×8 변환, 16×16 변환, 16×32 변환, 32×16 변환, 32×32 변환, 32×64 변환, 64×32 변환, 64×64 변환, 64×128 변환, 128×64 변환 및 128×128 변환 중 하나 이상을 포함하는, 비디오 신호를 코딩하는 방법.

청구항 14

비디오 신호를 코딩하는 비디오 코딩 디바이스로서,

복수의 상수 값들을 저장하는 메모리로서, 상기 상수 값들의 각각은 비디오 코딩과 연관된 복수의 상이한 변환 사이즈들 중 하나의 변환 사이즈와 연관되는, 상기 메모리;

상기 비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상기 상수 값을 선택하고, 또한, 상기 고정 소수점 변환 이전에 상기 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-쉬프트하고, 상기 고정 소수점 변환 이후에 상기 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-쉬프트함으로써, 상기 상수 값과 동일한 값을 이용하여 상기 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하는, 프로세서를 포함하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 15

제 14 항에 있어서,

상기 비디오 코딩 디바이스는 비디오 인코딩 디바이스를 포함하고,

상기 비디오 인코딩 디바이스는 상기 비디오 신호를 인코딩하는 엔트로피 인코딩 유닛을 더 포함하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 16

제 14 항에 있어서,

상기 비디오 코딩 디바이스는 비디오 디코딩 디바이스를 포함하고,

상기 비디오 디코딩 디바이스는 상기 비디오 신호를 디코딩하는 엔트로피 디코딩 유닛을 더 포함하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 17

제 14 항에 있어서,

상기 고정 소수점 변환은 2 차원 고정 소수점 변환을 포함하고,

상기 프로세서는, 상기 2 차원 고정 소수점 변환의 일 스테이지의 사이즈에 기초하여 상기 상수 값을 중 하나의 상수 값을 선택하고, 상기 상수 값을 2 배가 되도록 하며, 상기 2 배된 상수 값을 이용하여 상기 2 차원 고정 소수점 변환에 상기 가변 로컬화 비트-깊이 증가를 적용하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 18

제 14 항에 있어서,

상기 프로세서는,

내부 비트-깊이 증가가 상기 비디오 신호에 적용되는지 여부를 결정하고,

상기 내부 비트-깊이 증가가 상기 비디오 신호에 적용될 경우, 상기 내부 비트-깊이 증가의 값을 기초하여 상기 상수 값을 조정하며, 그리고

상기 조정된 상수 값과 동일한 값을 이용하여 상기 고정 소수점 변환에 상기 가변 로컬화 비트-깊이 증가를 적용하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 19

제 18 항에 있어서,

상기 프로세서는, 상기 상수 값 N 을 상기 내부 비트-깊이 증가의 값 B_d 만큼 감소시킴으로써 상기 조정된 상수 값이 $N - B_d$ 와 동일하도록 상기 상수 값을 조정하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 20

제 19 항에 있어서,

상기 프로세서는, 상기 내부 비트-깊이 증가의 값 B_d 이 상기 상수 값 N 이하일 경우에, 상기 상수 값 N 을 상기 내부 비트-깊이 증가의 값 B_d 만큼 감소시켜 $(N - B_d) \geq 0$ 이 되도록 하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 21

제 19 항에 있어서,

상기 프로세서는, 상기 내부 비트-깊이 증가의 값 B_d 이 상기 상수 값 N 보다 클 경우에, 상기 상수 값 N 을 0 으로 감소시켜 $(N - B_d) < 0$ 이 되도록 하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 22

제 14 항에 있어서,

상기 프로세서는, 상기 고정 소수점 변환 이후에 상기 상수 값과 동일한 비트 수만큼 상기 변환 출력 신호를 우측-수프트하기 전에 상기 변환 출력 신호에 오프셋을 적용하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 23

제 22 항에 있어서,

상기 상수 값은 N 과 동일하고,

상기 프로세서는 상기 변환 출력 신호에 2^{N-1} 과 동일한 오프셋을 적용하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 24

제 22 항에 있어서,

상기 프로세서는, 상기 가변 로컬화 비트-깊이 증가에 대한 오프셋 및 우측-수프트와 상기 변환 출력 신호에 적용된 다른 오프셋들 및 우측-수프트들을 결합하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 25

제 14 항에 있어서,

상기 메모리 내에 저장되기 전에, 상기 복수의 상이한 변환 사이즈들 중 하나의 변환 사이즈와 연관된 상수 값들의 각각은 상기 비디오 신호의 입력 비트-깊이 값, 상기 변환 사이즈와 연관된 변환 비트-깊이 값, 및 상기 변환 사이즈와 연관된 변환 비트-정확도 값에 기초하여 미리 계산되는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 26

제 25 항에 있어서,

상기 상수 값들의 각각, N 은 $B_i + B_{TR} + N \leq Z$ 를 만족하도록 미리 계산되고, 상기 B_i 는 상기 비디오 신호의 입력 비트-깊이 값이고, B_{TR} 는 상기 변환 사이즈와 연관된 변환 비트-깊이 값이고, Z 는 상기 변환 사이즈와 연관된 변환 비트-정확도 값인, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 27

제 14 항에 있어서,

상기 복수의 상이한 변환 사이즈들은 정사각 변환 사이즈들 및 직사각 변환 사이즈들 중 하나 이상을 포함하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 28

제 27 항에 있어서,

상기 복수의 상이한 변환 사이즈들은 4×4 변환, 4×8 변환, 8×4 변환, 8×8 변환, 8×16 변환, 16×8 변환, 16×16 변환, 16×32 변환, 32×16 변환, 32×32 변환, 32×64 변환, 64×32 변환, 64×64 변환, 64×128 변환, 128×64 변환 및 128×128 변환 중 하나 이상을 포함하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 29

비디오 신호를 코딩하는 비디오 코딩 디바이스로서,

복수의 상수 값들을 저장하는 수단으로서, 상기 상수 값들의 각각은 비디오 코딩과 연관된 복수의 상이한 변환 사이즈들 중 하나의 변환 사이즈와 연관되는, 상기 복수의 상수 값들을 저장하는 수단;

상기 비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상기 상수 값을 중 하나의 상수 값을 선택하는 수단; 및

상기 고정 소수점 변환 이전에 상기 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-ши프트하고, 상기 고정 소수점 변환 이후에 상기 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-ши프트함으로써, 상기 상수 값과 동일한 값을 이용하여 상기 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하는 수단을 포함하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 30

제 29 항에 있어서,

상기 고정 소수점 변환은 2 차원 고정 소수점 변환을 포함하고,

상기 2 차원 고정 소수점 변환의 일 스테이지의 사이즈에 기초하여 상기 상수 값을 중 하나의 상수 값을 선택하는 수단;

상기 상수 값을 2 배가 되도록 하는 수단; 및

상기 2 배된 상수 값을 이용하여 상기 2 차원 고정 소수점 변환에 상기 가변 로컬화 비트-깊이 증가를 적용하는 수단을 포함하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 31

제 29 항에 있어서,

내부 비트-깊이 증가가 상기 비디오 신호에 적용되는지 여부를 결정하는 수단;

상기 내부 비트-깊이 증가가 상기 비디오 신호에 적용될 경우, 상기 내부 비트-깊이 증가의 값에 기초하여 상기 상수 값을 조정하는 수단; 및

상기 조정된 상수 값과 동일한 값을 이용하여 상기 고정 소수점 변환에 상기 가변 로컬화 비트-깊이 증가를 적용하는 수단을 더 포함하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 32

제 29 항에 있어서,

상기 고정 소수점 변환 이후에 상기 상수 값과 동일한 비트 수만큼 상기 변환 출력 신호를 우측-ши프트하기 전에 상기 변환 출력 신호에 오프셋을 적용하는 수단을 더 포함하는, 비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 33

제 29 항에 있어서,

상기 복수의 상이한 변환 사이즈들은 정사각 변환 사이즈들 및 직사각 변환 사이즈들 중 하나 이상을 포함하는,

비디오 신호를 코딩하는 비디오 코딩 디바이스.

청구항 34

비디오 신호를 코딩하기 위한 명령을 포함하는 컴퓨터 판독가능 저장 매체로서,

상기 명령은, 프로세서에서 실행시, 상기 프로세서로 하여금,

복수의 상수 값들을 저장하게 하는 것으로서, 상기 상수 값들의 각각은 비디오 코딩과 연관된 복수의 상이한 변환 사이즈들 중 하나의 변환 사이즈와 연관되는, 상기 복수의 상수 값을 저장하게 하고;

비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상기 상수 값을 중 하나의 상수 값을 선택하게 하며; 그리고

상기 비디오 코딩 디바이스에서, 상기 고정 소수점 변환 이전에 상기 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-쉬프트하고, 상기 고정 소수점 변환 이후에 상기 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-쉬프트함으로써, 상기 상수 값과 동일한 값을 이용하여 상기 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하게 하는, 컴퓨터 판독가능 저장 매체.

청구항 35

제 34 항에 있어서,

상기 고정 소수점 변환은 2 차원 고정 소수점 변환을 포함하고,

상기 명령들은 상기 프로세서로 하여금,

상기 2 차원 고정 소수점 변환의 일 스테이지의 사이즈에 기초하여 상기 상수 값을 중 하나의 상수 값을 선택하게 하고;

상기 상수 값을 2 배가 되게 하고; 그리고

상기 2 배된 상수 값을 이용하여 상기 2 차원 고정 소수점 변환에 상기 가변 로컬화 비트-깊이 증가를 적용하게 하는, 컴퓨터 판독가능 저장 매체.

청구항 36

제 34 항에 있어서,

상기 프로세서로 하여금,

내부 비트-깊이 증가가 상기 비디오 신호에 적용되는지 여부를 결정하게 하고;

상기 내부 비트-깊이 증가가 상기 비디오 신호에 적용될 경우, 상기 내부 비트-깊이 증가의 값을 기초하여 상기 상수 값을 조정하게 하고; 그리고

상기 조정된 상수 값을 동일한 값을 이용하여 상기 고정 소수점 변환에 상기 가변 로컬화 비트-깊이 증가를 적용하게 하는

명령들을 더 포함하는, 컴퓨터 판독가능 저장 매체.

청구항 37

제 34 항에 있어서,

상기 프로세서로 하여금,

상기 고정 소수점 변환 이후에 상기 상수 값을 동일한 비트 수만큼 상기 변환 출력 신호를 우측-쉬프트하기 전에 상기 변환 출력 신호에 오프셋을 적용하게 하는

명령들을 더 포함하는, 컴퓨터 판독가능 저장 매체.

청구항 38

제 34 항에 있어서,

상기 복수의 상이한 변환 사이즈들은 정사각 변환 사이즈들 및 직사각 변환 사이즈들 중 하나 이상을 포함하는, 컴퓨터 판독가능 저장 매체.

명세서

기술분야

[0001] 본 개시는 비디오 코딩에 관한 것이고, 더욱 상세하게는, 비디오 코딩에서 고정 소수점 변환들에 관한 것이다.

배경기술

[0002] 디지털 비디오 성능들이, 디지털 텔레비전, 디지털 다이렉트 브로드캐스트 시스템, 무선 브로드캐스트 시스템, PDA (personal digital assistant), 랩톱 또는 데스크톱 컴퓨터, 태블릿 컴퓨터, 디지털 카메라, 디지털 기록 디바이스, 디지털 미디어 플레이어, 비디오 게이밍 디바이스, 비디오 게임 콘솔, 셀룰러 또는 위성 라디오 전화, 비디오 회상회의 디바이스들, 등을 포함하는 광범위한 디바이스들에 통합될 수 있다. 디지털 비디오 디바이스들은, MPEG-2, MPEG-4, ITU-T H.263, ITU-T H.264/MPEG-4, Part 10, AVC (Advanced Video Coding) 또는 제안된 ITU-T H.265, HEVC (High Efficiency Video Coding) 및 이러한 표준들의 확장들과 같은 비디오 압축 기술들을 구현하여 디지털 비디오 정보를 더 효율적으로 송신 및 수신한다.

[0003] 비디오 압축 기술들은 공간적 예측 및/또는 시간적 예측을 수행하여 비디오 시퀀스들에 내재하는 중복성 (redundancy) 을 감소 또는 제거할 수도 있다. 블록-기반 비디오 코딩을 위해, 비디오 프레임 또는 슬라이스는 매크로블록들로 분할될 수도 있다. 각각의 매크로블록은 추가로 분할될 수 있다. 인트라 코딩된 (I) 프레임 또는 슬라이스 내의 매크로블록들은 이웃하는 매크로블록들에 대하여 공간적 예측을 이용하여 인코딩된다. 인터 코딩된 (P 또는 B 또는 일반화된 P/B) 프레임 또는 슬라이스 내의 매크로블록들은 동일한 프레임 또는 슬라이스 내의 이웃하는 매크로블록들에 대하여 공간적 예측을 이용하고 다른 기준 프레임들에 대하여 시간적 예측을 이용할 수도 있다.

[0004] 비디오 압축 기술들은 또한 코딩될 현재 프레임의 비디오 블록과 기준 프레임의 예측형 비디오 블록 사이의 차이인 잔여 비디오 블록 데이터를 코딩을 위한 효율적인 방식으로 나타내기 위해 선형 변환들을 수행할 수도 있다. 일부 경우에, 비디오 코딩 디바이스는 캐스케이드 버터플라이 구조 (cascaded butterfly structure) 들을 이용하여 수행되는 선형 변환들의 고속 구현들을 이용할 수도 있다. 변환을 나타내기 위해 이용된 캐스케이드 버터플라이 구조들의 수는 변환의 사이즈에 의존한다. 비디오 코딩 디바이스들 내에서의 계산 복잡도를 감소시키기 위해, 선형 변환들은 부동 소수점 변환들 대신에 고정 소수점 변환들로서 구현될 수도 있다. 고정 소수점 변환들은, 그러나, 그 변환을 나타내는데 이용된 버터플라이 구조들의 각각의 캐스케이드 레벨들에 반올림 오차들을 도입할 수도 있다. 그러므로, 고정 소수점 변환들의 경우에, 누적되는 반올림 오차는 변환의 사이즈에 따라 증가한다.

발명의 내용

[0005] 본 개시는 비디오 코딩과 연관된 고정 소수점 변환에서 반올림 오차들을, 그 변환에 가변 로컬화 비트-깊이 증가를 적용함으로써 경감시키는 기술들을 설명한다. 더욱 상세하게는, 그 기술들은 비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상수 값을 선택하고, 상수 값과 동일한 값을 이용하여 변환에 가변 로컬화 비트-깊이 증가를 적용하는 것을 포함한다. 가변 로컬화 비트-깊이 증가를 적용하는 것은 고정 소수점 변환 이전에 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-ши프트하고, 고정 소수점 변환 이후에 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-ши프트하는 것을 포함한다.

[0006] 그 기술들에 따라, 고정 소수점 변환의 사이즈에 대한 상수 값은 비디오 코딩 디바이스에 저장된 복수의 상이한 변환 사이즈들에 대한 복수의 상수 값을로부터 선택된다. 상수 값을 각각은 비디오 코딩에 의해 지원되는 복수의 상이한 변환 사이즈들 중 하나에 대하여 미리 계산된다. 고정 소수점 변환은 그 변환의 사이즈에 의존하는 비트-깊이의 손실로 인해 변환 출력 신호에 반올림 오차를 도입할 수도 있다. 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하는 것은 변환을 위해 더 높은 비트-깊이의 변환 입력 신호를 제공한다. 더욱이, 고정 소수점 변환의 사이즈에 기초하여 계산된 값을 이용하여 가변 로컬화 비트-깊이 증가를 적용하는 것은 그 변환의 고정 소수점 구현으로 인한 반올림 오차를 감소시키거나 제거할 수도 있다.

[0007] 일 실시예에서, 본 개시는 비디오 신호를 코딩하는 방법과 관련되며, 그 방법은, 복수의 상수 값을 저장하는 단계로서, 상수 값을 각각은 비디오 코딩과 연관된 복수의 상이한 변환 사이즈들 중 하나의 변환 사이즈와 연관

되는, 복수의 상수 값들을 저장하는 단계 및 비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상수 값들 중 하나의 상수 값을 선택하는 단계를 포함한다. 그 방법은 또한, 비디오 코딩 디바이스에서, 고정 소수점 변환 이전에 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-쉬프트하고, 고정 소수점 변환 이후에 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-쉬프트함으로써, 상수 값과 동일한 값을 이용하여 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하는 단계를 포함한다.

[0008] 다른 실시예에서, 본 개시는 비디오 신호를 코딩하는 비디오 코딩 디바이스에 관한 것이며, 그 비디오 코딩 디바이스는, 복수의 상수 값들을 저장하는 메모리로서, 상수 값들 각각은 비디오 코딩과 연관된 복수의 상이한 변환 사이즈들 중 하나의 변환 사이즈와 연관되는, 메모리; 및 비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상수 값들 중 하나의 상수 값을 선택하고, 고정 소수점 변환 이전에 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-쉬프트하고, 고정 소수점 변환 이후에 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-쉬프트함으로써, 상수 값과 동일한 값을 이용하여 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하는 프로세서를 포함한다.

[0009] 추가의 실시예에서, 본 개시는 비디오 신호를 코딩하는 비디오 코딩 디바이스에 관한 것이며, 그 비디오 코딩 디바이스는, 복수의 상수 값들을 저장하는 수단으로서, 상수 값들 각각은 비디오 코딩과 연관된 복수의 상이한 변환 사이즈들 중 하나의 변환 사이즈와 연관되는, 복수의 상수 값들을 저장하는 수단; 및 비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상수 값들 중 하나의 상수 값을 선택하는 수단을 포함한다. 비디오 코딩 디바이스는 또한 고정 소수점 변환 이전에 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-쉬프트하고, 고정 소수점 변환 이후에 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-쉬프트함으로써, 상수 값과 동일한 값을 이용하여 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하는 수단을 포함한다.

[0010] 다른 실시예에서, 본 개시는 비디오 신호를 코딩하기 위한 명령을 포함하는 컴퓨터 판독가능 저장 매체에 관한 것이며, 그 명령은 프로세서에서 실행시, 프로세서로 하여금, 복수의 상수 값들을 저장하는 것으로서, 상수 값들 각각은 비디오 코딩과 연관된 복수의 상이한 변환 사이즈들 중 하나의 변환 사이즈와 연관되는, 복수의 상수 값들을 저장하게 하고; 비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상수 값들 중 하나의 상수 값을 선택하게 하며; 비디오 코딩 디바이스에서, 고정 소수점 변환 이전에 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-쉬프트하고, 고정 소수점 변환 이후에 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-쉬프트함으로써, 상수 값과 동일한 값을 이용하여 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하게 한다.

도면의 간단한 설명

[0011] 도 1 은 예시적인 비디오 인코딩 및 디코딩 시스템을 도시하는 블록도이며, 예시적인 비디오 인코딩 및 디코딩 시스템은 그 시스템 내에서 소스 디바이스의 비디오 인코더 및 목적지 디바이스의 비디오 디코더와 연관된 고정 소수점 변환들에 가변 로컬화 비트-깊이 증가들을 적용하는 기술들을 활용할 수도 있다.

도 2 는 비디오 코딩과 연관된 변환을 위한 버터플라이 구조를 도시하는 개념도이다.

도 3 은 비디오 코딩과 연관된 변환을 위한 캐스케이드 버터플라이 구조들을 도시하는 개념도이다.

도 4 는 변환 유닛의 고정 소수점 구현에서 반올림 오차들을 경감시키기 위해 변환 유닛에 가변 로컬화 비트-깊이 증가를 적용하는 기술들을 구현할 수도 있는 예시적인 비디오 인코더를 도시하는 블록도이다.

도 5 는 역변환 유닛의 고정 소수점 변환에서 반올림 오차들을 경감시키기 위해 역변환 유닛에 가변 로컬화 비트-깊이 증가를 적용하는 기술들을 구현할 수도 있는 예시적인 비디오 디코더를 도시하는 블록도이다.

도 6 은 변환 유닛에 가변 로컬화 비트-깊이 증가를 적용하는 기술들을 구현할 수도 있는, 내부 비트-깊이 증가 (IBDI) 가 이용되는 예시적인 비디오 인코더를 도시하는 블록도이다.

도 7 은 역변환 유닛에 가변 로컬화 비트-깊이 증가를 적용하는 기술들을 구현할 수도 있는, 내부 비트-깊이 증가 (IBDI) 가 이용되는 예시적인 비디오 디코더를 도시하는 블록도이다.

도 8 은 변환 유닛의 고정 소수점 구현에서 반올림 오차들을 경감시키기 위해 변환 유닛에 가변 로컬화 비트-깊이 증가를 적용하는 예시적인 방법을 도시하는 흐름도이다.

도 9 는 역변환 유닛의 고정 소수점 구현에서 반올림 오차들을 경감시키기 위해 역변환 유닛에 가변 로컬화 비트-깊이 증가를 적용하는 예시적인 방법을 도시하는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0012]

본 개시에는, 비디오 코딩과 연관된 변환의 고정 소수점 구현에 의해 도입되는 반올림 오차들을, 그 변환에 가변 로컬화 비트-깊이 증가를 적용함으로써 경감시키는 기술들이 설명된다. 더욱 상세하게는, 그 기술들은 비디오 코딩 디바이스에서 고정 소수점 변환의 사이즈에 기초하여 상수 값을 선택하고, 그 상수 값과 동일한 값을 이용하여 그 변환에 가변 로컬화 비트-깊이 증가를 적용하는 것을 포함한다. 가변 로컬화 비트-깊이 증가를 적용하기 위한 기술들은 고정 소수점 변환 이전에 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-ши프트 하고, 고정 소수점 변환 이후에 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-ши프트하는 것을 포함한다. 고정 소수점 변환의 사이즈에 대한 상수 값은 비디오 코딩 디바이스에 저장된 복수의 상이한 변환 사이즈들에 대한 복수의 상수 값을로부터 선택된다. 상수 값을 각각은 비디오 코딩에 의해 지원된 복수의 상이한 변환 사이즈들 중 하나에 대하여 미리 계산된다.

[0013]

도 1은 예시적인 비디오 인코딩 및 디코딩 시스템 (10)을 도시하는 블록도이며, 예시적인 비디오 인코더 및 디코더 시스템 (10)은 그 시스템 (10) 내에서 소스 디바이스 (12)의 비디오 인코더 (20) 및 목적지 디바이스 (14)의 비디오 디코더 (30)와 연관된 고정 소수점 변환들에 가변 로컬화 비트-깊이 증가들을 적용하는 기술들을 활용할 수도 있다. 가변 로컬화 비트-깊이 증가들의 값들은 복수의 상이한 변환 사이즈들에 대하여 미리 계산되고, 비디오 인코더 (20) 및 비디오 디코더 (30)에 저장된다. 적절한 가변 로컬화 비트-깊이 증가들은 개별 고정 소수점 변환들의 사이즈들에 기초하여 비디오 인코더 (20)와 비디오 디코더 (30)에 의해 선택될 수도 있다. 이러한 방식으로, 가변 로컬화 비트-깊이 증가들은 시스템 (10) 내의 특정 변환들의 고정 소수점 구현들에 의해 도입된 반올림 오차들을 경감시킬 수도 있다.

[0014]

도 1에 도시된 것과 같이, 시스템 (10)은 인코딩된 비디오를 통신 채널 (16)을 통해 목적지 디바이스 (14)로 송신하는 소스 디바이스 (12)를 포함한다. 소스 디바이스 (12) 및 목적지 디바이스 (14)는 디지털 텔레비전, 디지털 디렉트 브로드캐스트 시스템, 무선 브로드캐스트 시스템, PDA (personal digital assistant), 랩톱 또는 데스크톱 컴퓨터, 태블릿 컴퓨터, 디지털 카메라, 디지털 기록 디바이스, 디지털 미디어 플레이어, 비디오 게이밍 디바이스, 비디오 게임 콘솔, 셀룰러 또는 위성 라디오 전화, 비디오 화상회의 디바이스들, 등을 포함하는 광범위한 디바이스들 중 임의의 디바이스를 포함할 수도 있다. 일부 경우에, 소스 디바이스 (12) 및 목적지 디바이스 (14)는 통신 채널 (16)을 통해 비디오 정보를 통신할 수 있는 무선 통신 디바이스들을 포함할 수도 있고, 이 경우 통신 채널 (16)은 무선이다.

[0015]

그러나, 비디오 코딩과 연관된 고정 소수점 변환들에 가변 로컬화 비트-깊이 증가들을 적용하는 것과 관련된 본 개시의 기술들은, 무선 애플리케이션들 또는 세팅들에 제한될 필요는 없다. 예를 들어, 이러한 기술들은 공중-경유 텔레비전 브로드캐스트, 케이블 텔레비전 송신, 위성 텔레비전 송신, 인터넷 비디오 송신, 저장 매체 상에 인코딩된 인코딩 디지털 비디오, 또는 다른 시나리오들에 적용할 수도 있다. 따라서, 통신 채널 (16)은 인코딩된 비디오 데이터의 송신에 적합한 무선 또는 유선 매체의 임의의 조합을 포함할 수도 있다.

[0016]

도 1의 실시예에서, 소스 디바이스 (12)는 비디오 소스 (18), 비디오 인코더 (20), 변조기/복조기 (모뎀) (22) 및 송신기 (24)를 포함할 수도 있다. 목적지 디바이스 (14)는 수신기 (26), 모뎀 (28), 비디오 디코더 (30) 및 디스플레이 디바이스 (32)를 포함할 수도 있다. 본 개시에 따르면, 소스 디바이스 (12)의 비디오 인코더 (20)는 비디오 인코더 (20)에 저장된 복수의 상수 값을로부터 고정 소수점 변환의 사이즈에 기초하여 상수 값을 선택함으로써 비디오 인코더 (20)와 연관된 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하고, 그 상수 값과 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 변환에 적용하는 기술들을 수행하도록 구성될 수도 있다. 또한, 목적지 디바이스 (14)의 비디오 디코더 (30)는 유사하게 비디오 디코더 (30)와 연관된 고정 소수점 역변환에 가변 로컬화 비트-깊이 증가를 적용하는 기술들을 수행하도록 구성될 수도 있다. 다른 예들에서, 소스 디바이스 및 목적지 디바이스는 다른 컴포넌트들 또는 배열들을 포함할 수도 있다. 예를 들면, 소스 디바이스 (12)는 외부 카메라와 같은 외부 비디오 소스 (18)로부터 비디오 데이터를 수신할 수도 있다. 유사하게, 목적지 디바이스 (14)는 집적 디스플레이 디바이스를 포함하는 것이 아니라 외부 디스플레이 디바이스와 인터페이싱할 수도 있다.

[0017]

도 1의 예시된 시스템 (10)은 단지 하나의 실시예이다. 비디오 코딩과 연관된 고정 소수점 변환들에 가변 로컬화 비트-깊이 증가들을 적용하는 기술들은 임의의 디지털 비디오 인코딩 및/또는 디코딩 디바이스에 의해 수행될 수도 있다. 본 개시의 기술들은 또한 통상적으로 "코덱 (CODEC)"이라 지칭되는 비디오 인코더/디코더에 의해 수행될 수도 있다. 추가로, 본 개시의 기술들은 비디오 프리프로세서에 의해 수행될 수도 있다. 소스 디바이스 (12) 및 목적지 디바이스 (14)는, 소스 디바이스 (12)가 목적지 디바이스 (14)로의 송신을

위해 코딩된 비디오 데이터를 생성하는 이러한 코딩 디바이스들의 단지 실시예들이다. 일부 실시예들에서, 디바이스들 (12, 14)은 디바이스들 (12, 13)의 각각이 비디오 인코딩 및 디코딩 컴포넌트들을 포함하도록, 실질적으로 대칭적인 방식으로 동작할 수도 있다. 따라서, 시스템 (10)은 예컨대, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 또는 비디오 전화를 위해 비디오 디바이스들 (12, 14) 사이에 1 방향 또는 2 방향 비디오 송신을 지원할 수도 있다.

[0018] 소스 디바이스 (12)의 비디오 소스 (18)는 비디오 카메라와 같은 비디오 캡처 디바이스, 이전에 캡처된 비디오를 포함하는 비디오 아카이브 (video archive), 및/또는 비디오 콘텐트 제공자로부터의 비디오 퍼드 (video feed)를 포함할 수도 있다. 추가의 대안으로서, 비디오 소스 (18)는 소스 비디오로서 컴퓨터 그래픽 기반 데이터, 또는 라이브 비디오, 아카이브 비디오 및 컴퓨터 생성된 비디오의 조합을 생성할 수도 있다. 몇몇 경우들에서, 비디오 소스 (18)가 비디오 카메라이면, 소스 디바이스 (12) 및 목적지 디바이스 (14)는 소위 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 그러나, 전술된 것과 같이, 본 개시에 설명된 기술들은 일반적으로 비디오 코딩에 적용가능할 수도 있고, 무선 및/또는 유선 애플리케이션들에 적용될 수도 있다. 각 경우에서, 캡처되거나, 미리 캡처되거나 또는 컴퓨터 생성된 비디오는 비디오 인코더 (20)에 의해 인코딩될 수도 있다. 그 다음 인코딩된 비디오 정보는 통신 표준에 따라 모뎀 (22)에 의해 변조되고, 송신기 (24)를 통해 목적지 디바이스 (14)로 송신될 수도 있다. 모뎀 (22)은 다양한 맵서들, 필터들, 증폭기들 또는 신호 변조를 위해 설계된 다른 컴포넌트들을 포함할 수도 있다. 송신기 (24)는 증폭기들, 필터들 및 하나 이상의 안테나들을 포함하는, 데이터를 송신하기 위해 설계된 회로들을 포함할 수도 있다.

[0019] 목적지 디바이스 (14)의 수신기 (26)는 채널 (16)을 통해 정보를 수신하고 모뎀 (28)은 정보를 복조한다. 채널 (16)을 통해 통신되는 정보는 비디오 인코더 (20)에 의해 정의되고, 또한 비디오 디코더 (30)에 의해 사용되는 신팩스 정보를 포함할 수도 있고, 이 정보는 매크로 블록들 및 다른 코딩된 유닛들, 예컨대 GOP들의 특징들 및/또는 프로세싱을 설명하는 신팩스 엘리먼트들을 포함한다. 디스플레이 디바이스 (32)는 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 음극선관 (CRT), 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 다른 타입의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 것을 포함할 수도 있다.

[0020] 도 1의 예에서, 통신 채널 (16)은 임의의 무선 또는 유선 통신 매체, 이를테면 무선 주파수 (RF) 스펙트럼 또는 하나 이상의 물리적 송신 라인들, 또는 무선 및 유선 매체의 임의의 조합을 포함할 수도 있다. 통신 채널 (16)은, 로컬 영역 네트워크 (LAN), 와이드 영역 네트워크 (WAN), 또는 인터넷과 같은 글로벌 네트워크와 같은 패킷 기반 네트워크의 일 부분을 형성할 수도 있다. 일반적으로 통신 채널 (16)은, 소스 디바이스 (12)로부터 목적지 디바이스 (14)로 비디오 데이터를 송신하기 위한, 임의의 적합한 통신 매체, 또는 상이한 통신 매체들의 집합 (collection)을 나타내며, 이는 유선 또는 무선 매체의 임의의 적합한 조합을 포함한다. 통신 채널 (16)은 라우터, 스위치, 기지국, 또는 소스 디바이스 (12)로부터 목적지 디바이스 (14)로 통신을 용이하게 하기 위해 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0021] 비디오 인코더 (20) 및 비디오 디코더 (30)는 대안적으로 MPEG-4, Part 10, AVC (Advanced Video Coding)로 지정되는 ITU-T H.264 표준, 또는 고 효율 비디오 코딩 (HEVC)으로 지정되는 최신 차세대 비디오 코딩 표준과 같은 비디오 압축 표준에 따라 동작할 수도 있다. 하지만, 본 개시의 기술들은 임의의 특정 코딩 표준에 한정되지 않는다. 다른 실시예들은 MPEG-2 및 ITU-T H.263을 포함한다. 비록 도 1에 도시되지는 않았지만, 몇몇 양태들에서, 비디오 인코더 (20) 및 비디오 디코더 (30)는 각각 오디오 인코더 및 디코더와 통합될 수 있고, 적절한 MUX-DEMUX 유닛들 또는 다른 하드웨어 및 소프트웨어를 포함하여 공통 데이터 스트림 또는 개별 데이터 스트림들에서 오디오 및 비디오 양쪽 모두의 인코딩을 핸들링 (handling) 할 수도 있다. 적용가능하다면, MUX-DEMUX 유닛들은 ITU H.223 멀티플렉서 프로토콜 또는 다른 프로토콜들 이를테면 사용자 데이터그램 프로토콜 (UDP)을 따를 수도 있다.

[0022] 비디오 시퀀스는 통상적으로 비디오 프레임들의 시리즈를 포함한다. 꾹처들의 그룹 (GOP)은 일반적으로 하나 이상의 비디오 프레임들의 시리즈를 포함한다. GOP는 GOP의 헤더, 즉 GOP의 하나 이상의 프레임들의 헤더에서 신팩스 데이터를 포함할 수도 있거나, 그렇지 않으면 GOP 내에 포함된 복수의 프레임들을 설명할 수도 있다. 각각의 프레임은 개별 프레임에 대한 인코딩 모드를 설명하는 프레임 신팩스 데이터를 포함할 수도 있다. 비디오 인코더 (20)는 통상적으로 비디오 데이터를 인코딩하기 위해 개별 비디오 프레임들 내의 비디오 블록들에 동작한다. 비디오 블록은 매크로블록 또는 매크로블록의 파티션에 대응할 수도 있다. 예를 들면, 매크로블록의 사이즈 미만의 블록 사이즈들은 매크로블록들의 파티션들이라 지칭될 수도 있다. 비디오 블록들은 고정되거나 변화하는 사이즈들을 가질 수도 있고, 명시된 코딩 표준에 따라 사이즈가 상이할 수

도 있다. 각각의 비디오 프레임은 복수의 슬라이스들을 포함할 수도 있다. 각각의 슬라이스는 서브-블록들로 지칭되는 파티션들로 배열될 수도 있는 복수의 매크로블록들을 포함할 수도 있다.

[0023] 일 예로서, ITU-T H.264 표준은 루마 성분 (luma component) 들을 위해 16×16 , 8×8 , 또는 4×4 그리고 크로마 성분 (chroma component) 들을 위해 8×8 와 같은 다양한 블록 사이즈들에서 인트라 예측을 지원하고, 루마 성분들을 위해 16×16 , 16×8 , 8×16 , 8×8 , 8×4 , 4×8 및 4×4 그리고 크로마 성분들을 위해 대응하는 스케일의 사이즈들과 같은 다양한 블록 사이즈들로 인터 예측을 지원한다. HEVC 표준은 32×32 , 64×64 또는 128×128 까지의 블록 사이즈들로 인트라 및 인터 예측을 지원할 수도 있다. HEVC 표준은 코딩 유닛들 (CU), 예측 유닛들 (PU), 및 변환 유닛들 (TU) 에 기초하여 더 효율적이고 플렉시블한 비디오 코딩을 제공한다. CU 는 H.264 표준에서 이용된 매크로블록과 유사한 기본 압축 유닛이다. HEVC 에서, 32×32 , 64×64 또는 128×128 까지의 CU 사이즈들은 고해상도 비디오 콘텐츠를 효율적으로 압축하기 위해 지원된다. PU 는 인터 /인트라 예측의 유닛이며, 단일 CU 에서 복수의 PU 들이 존재할 수도 있다. TU 는 변환 유닛이며, 하나 이상의 PU 들을 포함할 수도 있다. 종래의 4×4 및 8×8 변환들에 부가하여, 16×16 , 32×32 , 64×64 및 128×128 정수 변환들이 TU 들을 위해 지원될 수도 있다. 추가로, 4×8 , 8×4 , 8×16 , 16×8 , 16×32 , 32×16 , 32×64 , 64×32 , 64×128 , 및 128×64 를 포함하는 직사각 변환이 또한 TU 들을 위해 지원될 수도 있다.

[0024] 본 개시에서, "N×N" 및 "N 바이 N" 은 예컨대, 16×16 픽셀 또는 16 바이 16 픽셀들과 같이 블록들의 픽셀 크기들을 수직 및 수평 크기들과 관련하여 지칭하도록 상호교환가능하게 이용될 수도 있다. 일반적으로, 16×16 블록은 수직 방향에서 16개 픽셀들 ($y = 16$) 및 수평 방향에서 16개 픽셀들 ($x = 16$) 을 가질 것이다. 유사하게, N×N 블록은 일반적으로 수직 방향에서 N개의 픽셀들 및 수평 방향에서 N개의 픽셀들을 가지며, 여기서 N 은 양의 정수값을 나타낸다. 블록 내의 픽셀들은 행과 열로 배열될 수도 있다. 추가로, 블록들은 반드시 수평 방향에서 픽셀들의 수와 동일한 수의 픽셀들을 수직 방향에서 가져야할 필요는 없다. 예를 들면, 블록들은 N×M 픽셀들을 포함할 수도 있고, 상기 M 은 반드시 N 과 동일하지 않다.

[0025] 비디오 블록들은 픽셀 영역에서 픽셀 데이터의 블록들 또는 변환 영역에서 변환 계수들의 블록들 다음에, 코딩된 비디오 블록들과 예측 비디오 블록들 사이의 픽셀 차이들을 나타내는 잔여 비디오 블록 데이터의 변환의 애플리케이션을 포함할 수도 있다. 일부 경우에, 비디오 블록은 변환 영역에서 양자화된 변환 계수들의 블록들 다음에, 변환으로부터 출력된 변환 계수들로의 양자화기의 적용을 포함할 수도 있다.

[0026] 비디오 블록들이 작을수록 더 양호한 해상도를 제공할 수 있고, 고 레벨의 세부사항을 포함하는 비디오 프레임의 로케이션들을 위해 이용될 수도 있다. 일반적으로, 매크로블록들 및 때때로 서브-블록들이라 지칭되는 다양한 파티션들은 비디오 블록들로 간주될 수도 있다. 추가로, 슬라이스는 매크로블록들 및/또는 서브-블록들과 같은 복수의 비디오 블록들로 간주될 수도 있다. 각각의 슬라이스는 비디오 프레임의 독립적으로 디코딩 가능한 유닛일 수도 있다. 대안적으로, 프레임은 자체적으로 디코딩 가능한 유닛들일 수도 있거나, 프레임의 다른 부분들이 디코딩 가능한 유닛들로 정의될 수도 있다. 전술된 용어 코딩 유닛 (CU) 은, 전체 프레임과 같은 비디오 프레임의 임의의 독립적으로 디코딩 가능한 유닛, 일 프레임의 슬라이스, 시퀀스로 지칭되는 픽처들의 그룹 (GOP), 또는 적용 가능한 코딩 기술들에 따라 정의된 다른 독립적으로 디코딩 가능한 유닛을 지칭할 수도 있다.

[0027] 예측 데이터 및 잔여 데이터를 생성하기 위한 인트라-예측 또는 인터-예측 코딩 이후에, 코딩을 위한 더 효율적인 방식으로 잔여 데이터를 나타내는 잔여 변환 계수들을 생성하기 위해 잔여 데이터의 변환이 수행될 수도 있다. 변환들은 순방향 또는 역방향의 이산 코사인 변환 (DCT), 정수 변환, 웨이브릿 (wavelet) 변환, 또는 다른 개념적으로 유사한 선형 변환을 포함할 수도 있다. 일부 경우에, 변환들은 변환 사이즈와 관련된 복수의 캐스케이드 버터플라이 구조들을 이용하여 수행되는 고속 구현들을 가질 수도 있다. 추가로, 계산상의 복잡도를 감소시키기 위해, 변환들은 고정 소수점 구현들을 가질 수도 있다. 그러나, 고정 소수점 변환은 변환 출력 신호에 반올림 오차들을 도입할 수도 있다. 이는 캐스케이드 버터플라이 스테이지들이 반올림에 의해 또는 반올림 없이 변환 출력 신호를 복수의 비트들만큼 우측-ши프트하는 경우에 발생할 수도 있다. 예컨대, n-비트 값은 고정 소수점 변환 동안 m-비트 값으로 우측-ши프트될 수도 있고, 상기 n 은 m 보다 크다. 비트 손실량은 통상적으로 변환 사이즈에 대응하는 캐스케이드 버터플라이 스테이지들의 수에 의존한다.

[0028] 본 개시의 기술들에 따르면, 비디오 인코더 (20) 및/또는 비디오 디코더 (30) 는 고정 소수점 변환에서 반올림 오차들을, 그 변환에 가변 로컬화 비트-깊이 증가를 적용함으로써 경감시킬 수도 있다. 더욱 상세하게는, 기술들은 고정 소수점 변환의 사이즈에 기초하여 상수 값을 선택하고, 그 상수 값과 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 변환에 적용하는 것을 포함한다. 가변 로컬화 비트-깊이 증가는 고정 소수점 변환

이전의 상수 값과 동일한 비트 수만큼 변환 입력 신호를 좌측-쉬프트 하고, 고정 소수점 변환 이후의 상수 값과 동일한 비트 수만큼 변환 출력 신호를 우측-쉬프트함으로써 적용된다. 고정 소수점 변환의 사이즈에 대한 상수 값은 비디오 인코더 (20) 및/또는 비디오 디코더 (30)에 저장된 복수의 상수 값을로부터 선택될 수도 있다. 상수 값을 각각은 비디오 코딩에 의해 지원된 복수의 상이한 변환 사이즈들 중 하나에 대하여 미리 계산된다.

[0029] 통상적으로, 비디오 코딩과 연관된 고정 소수점 변환들은 2 차원 변환들을 포함한다. 일 예로서, 2 차원 변환은 비디오 블록 내의 픽셀 데이터들의 행들에 제 1 스테이지 1 차원 변환을 적용한 후, 비디오 블록 내의 픽셀 데이터의 열들에 제 2 스테이지 1 차원 변환을 적용하거나, 그 반대로 적용함으로써 수행될 수도 있다. 그러므로, 본 개시의 기술들은 2 차원 고정 소수점 변환의 일 스테이지의 사이즈에 기초하여 상수 값을 선택하고, 상수 값을 2 배가 되게 하고, 그리고 2 배된 상수 값 이하의 값을 이용하여 가변 로컬화 비트-깊이 증가를 2 차원 변환에 적용하는 것을 포함할 수도 있다. 다른 실시예들에서, 로컬화 비트-깊이 증가는 2 차원 변환의 각 스테이지에 적용될 수도 있고, 2 차원 변환의 일 스테이지를 위해 선택된 상수 값을 동일한 값을 갖는다.

[0030] 기술들에 따라, 비디오 코딩 표준에 의해 지원되는 각각의 상이한 변환 사이즈에 대하여 상이한 상수 값이 미리 계산된다. 예를 들면, ITU-T H.264 (AVC) 표준은 사이즈들 4×4 및 8×8 의 변환들을 지원할 수도 있다.

최신 HEVC 표준은 TU 들을 위해 4×4 , 4×8 , 8×4 , 8×8 , 8×16 , 16×8 , 16×16 , 16×32 , 32×16 , 32×32 , 32×64 , 64×32 , 64×64 , 64×128 , 128×64 , 또는 128×128 을 포함하는 변환들을 지원할 수도 있다. 복수의 상수 값을 비디오 인코더 (20) 및/또는 비디오 디코더 (30)에 저장될 수도 있고, 상수 값을 중 적절한 상수 값이 비디오 인코더 (20) 및/또는 비디오 디코더 (30)에 적용될 고정 소수점 변환의 사이즈에 기초하여 선택될 수 있다. 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용하는 것은 변환 입력 신호에 더 높은 비트-정확도를 제공하여 변환 동안 비트-깊이의 손실을 오프셋한다. 추가로, 고정 소수점 변환의 사이즈에 기초하여 계산된 값을 이용하여 가변 로컬화 비트-깊이 증가를 적용하는 것은 변환 사이즈에 대응하는 복수의 캐스케이드 버퍼플라이 스테이지들 만큼 변환 출력 신호를 우측-쉬프트함으로 인한 반올림 오차를 감소시키거나 제거한다.

[0031] 일 실시예에서, 비디오 인코더 (20) 및/또는 비디오 디코더 (30)는 인코딩될 비디오 프레임 또는 슬라이스의 소정의 CU 에 대한 TU 의 사이즈에 기초하여 비디오 코딩과 연관된 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 선택적으로 적용한다. 예를 들면, 가변 로컬화 비트-깊이 증가는 TU 가 32×32 , 64×64 , 또는 128×128 과 같이 상대적으로 클 경우에 변환에 적용될 수도 있지만, TU 가 4×4 또는 8×8 과 같이 작은 경우에 변환에 적용되지 않는다.

[0032] 일부 실시예들에서, 비디오 인코더 (20)는 임의의 비디오 블록 코딩을 수행하기 전에 비디오 소스 (18)로부터 수신된 비디오 신호에 내부 비트-깊이 증가 (IBDI) 를 적용할 수도 있다. 예를 들면, 비디오 인코더 (20)는 더 높은 비트-정확도의 비디오 코딩을 수행하기 위해 8 비트에서 12 비트로 입력 비트-깊이로부터 비디오 신호의 비트-깊이를 증가시킬 수도 있다. 비디오 인코더 (20)는 비디오 신호의 비트-깊이를 증가시킬 수도 있고, 비디오 디코더 (30)는 이후에 재구성된 비디오 신호 블록의 비트-깊이를 그 입력 비트-깊이로 감소시킬 수도 있다. 이 경우, 기술들에 따라, 고정 소수점 변환의 사이즈에 기초하여 선택된 상수 값을 IBDI 의 값에 기초하여 조정될 수도 있다. 예를 들어, IBDI 가 비디오 신호에 적용된다면, 그 후에 고정 소수점 변환의 사이즈에 기초하여 선택된 상수 값을 IBDI 의 값만큼 감소될 수도 있다. 그러나, IBDI 의 값이 상수 값보다 크고 그 감소로 인해 가변 로컬화 비트-깊이 증가에 대하여 음의 값을 발생한다면, 상수 값을 0 으로 감소되고 어떤 가변 로컬화 비트-깊이 증가도 고정 소수점 변환에 적용되지 않는다.

[0033] 변환 이후에, 변환 계수들의 양자화가 수행된다. 일반적으로 양자화는, 변환 계수들이 그 계수들을 나타내기 위해 사용된 데이터량을 감소시킬 수 있도록 양자화되는 프로세스를 지칭한다. 양자화 프로세스는 또한 계수들의 일부 또는 전부와 연관된 비트 깊이를 감소시킬 수도 있다. 양자화 다음에, 양자화된 데이터의 엔트로피 코딩이 예를 들면 CAVLC (content adaptive variable length coding), CABAC (context adaptive binary arithmetic coding), 또는 다른 엔트로피 코딩 방법론에 따라 수행될 수도 있다. 엔트로피 코딩을 위해 구성된 프로세싱 유닛 또는 다른 프로세싱 유닛은 양자화된 계수들의 제로 런 랜스 코딩 (zero run length coding) 및/또는 코딩된 블록 패턴 (CBP) 값을, 매크로블록 타입, 코딩 모드, 코딩된 유닛 (예컨대, 프레임, 슬라이스, 매크로블록 또는 시퀀스)의 최대 매크로블록 사이즈, 또는 기타 등등과 같은 신박스 정보의 생성과 같은 다른 프로세싱 기능들을 수행할 수도 있다.

[0034] 비디오 인코더 (20)는 추가로 예컨대, 프레임 헤더, 블록 헤더, 슬라이스 헤더, 또는 GOP 헤더에서 블록-기반

의 선택스 데이터, 프레임-기반의 선택스 데이터, 및 GOP-기반의 선택스 데이터와 같은 선택스 데이터를 비디오 디코더 (30)로 송신할 수도 있다. GOP 선택스 데이터는 개별 GOP 내의 복수의 프레임들을 설명할 수도 있고, 프레임 선택스 데이터는 대응하는 프레임을 인코딩하는데 이용된 인코딩/예측 모드를 나타낼 수도 있다.

IBDI가 비디오 신호에 적용될 경우, 비디오 인코더 (20)는 IBDI의 값을 비디오 디코더 (30)로 명확하게 시그널링하여 비디오 디코더 (30)가 디코딩된 비디오 신호 블록을 그 입력 비트-깊이로 가져가도록 IBDI를 제거할 수도 있다. 일부 실시예들에서, 비디오 인코더 (20)는 역변환에서 가변 로컬화 비트-깊이 증가를 위해 선택된 상수 값에 관한 정보를 비디오 디코더 (30)로 명확히 시그널링 할 수도 있다.

[0035] 본 개시에서, 용어 "코딩"은 인코딩 또는 디코딩을 지칭한다. 유사하게, 용어 "코더"는 일반적으로 임의의 비디오 인코더, 비디오 디코더, 또는 결합된 인코더/디코더 (CODEC)를 지칭한다. 따라서, 용어 "코더"는 본 명세서에서 비디오 인코딩 또는 비디오 디코딩을 수행하는 특수 컴퓨터 디바이스 또는 장치를 지칭하도록 이용된다. 본 개시의 기술들은 인코더들 또는 디코더들에 적용 가능할 수도 있다. 예를 들면, 비디오 인코더 (20)와 비디오 디코더 (30) 각각은 개별 카메라, 컴퓨터, 모바일 디바이스, 가입자 디바이스, 브로드캐스트 디바이스, 셋톱 박스, 서버, 또는 기타 등등에 포함될 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 각각은 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSPs), 애플리케이션용 집적 회로들 (ASICs), 필드 프로그래머블 게이트 어레이들 (FPGAs), 이산 로직, 소프트웨어, 하드웨어, 펌웨어, 또는 이들의 임의의 조합들과 같은, 임의의 다양한 적절한 인코더 회로로서 구현될 수도 있다.

[0036] 도 2는 비디오 코딩과 연관된 변환을 위한 버터플라이 구조 (34)를 도시하는 개념도이다. 전술된 것과 같이, 비디오 코딩과 연관된 변환들은 도 1의 비디오 인코더 (20) 또는 비디오 디코더 (30)와 같은 비디오 코딩 디바이스 내의 고속 구현들을 가질 수도 있다. 변환들의 이러한 고속 구현들은 변환 사이즈와 관련된 복수의 캐스케이드 버터플라이 구조들을 이용하여 수행될 수도 있다. 도 2에 도시된 버터플라이 구조 (34)는 이러한 버터플라이 구조의 일 실시예이다. 예를 들어 1 차원 16-포인트 변환의 경우, 그 변환은 버터플라이 구조 (34; 도 3에 도시)와 유사한 복수의 캐스케이드 버터플라이 구조들에 의해 표현될 수도 있다. 버터플라이 구조 (34)에 의해 적어도 부분적으로 표현된 변환은 순방향 이산 코사인 변환 (DCT), 정수 변환, 웨이브릿 변환, 또는 다른 개념적으로 유사한 선형 변환을 포함할 수도 있다.

[0037] 버터플라이 구조 (34)에 의해 구현된 변환의 부분은 하기의 식들에 의해 표현될 수도 있다:

$$Y_i = W_i * X_i + W_{i+1} * X_{i+1} \text{ 및}$$

$$Y_{i+1} = W_{i+1} * X_i - W_i * X_{i+1},$$

[0040] 상기 X_i 및 X_{i+1} 는 변환 입력 값들이고, Y_i 및 Y_{i+1} 는 변환 출력 값들이고, W_i 및 W_{i+1} 는 각종 또는 곱셈 인자들이다. 변환 입력 신호는 $X = (X_0, X_1, X_2 \dots X_M)$ 으로 표현될 수도 있고, 변환 출력 신호는 $Y = (Y_0, Y_1, Y_2 \dots Y_M)$ 으로 표현될 수도 있다. 비디오 인코딩의 경우에, 변환 입력 값들은 잔여 비디오 블록의 픽셀 값들을 포함할 수도 있고, 변환 출력 값들은 변환 계수들을 포함할 수도 있다. 비디오 디코딩의 경우에, 변환 입력 값들은 역양자화된 변환 계수들을 포함할 수도 있고, 변환 출력 값들은 잔여 비디오 블록의 픽셀 값들을 포함할 수도 있다. 변환이 DCT인 경우, 예컨대, 곱셈 인자들은 사인들 및 코사인들이다. 다른 변환 타입들의 경우, 곱셈 인자들은 정수들, 웨이브릿 또는 기타 등등의 형태를 취할 수도 있다.

[0041] 도 1의 비디오 인코더 (20) 또는 비디오 디코더 (30)와 같은 비디오 코딩 디바이스 내의 계산 복잡도를 감소시키기 위해, 비디오 코딩과 연관된 변환들은 부동 소수점 변환들 고정 소수점 변환들로서 구현될 수도 있다.

고정 소수점 변환들의 경우, 곱셈 인자들은 정수들 또는 이항 유리수 (dyadic rationals) 들일 수도 있다. 이항 유리수는 분자가 상수이고, 분모가 2의 거듭제곱인 유리수이다. 예를 들어, 버터플라이 구조 (34)에 의해 적어도 부분적으로 표현된 고정 소수점 변환 구현에서, 곱셈 인자들은 $W_i = A/2^m$ 및 $W_{i+1} = B/2^m$ 이고, 상기 A, B, 및 m은 정수들이다. 고정 소수점에서, 2^m 으로 나누는 것은 m비트들 만큼의 우측-수프트로 표현될 수 있다. 그러므로, 버터플라이 구조 (34)의 변환 출력 값들은 하기의 식들에 의해 표현될 수도 있다:

$$Y_i = (A * X_i + B * X_{i+1}) \gg m, \text{ 및}$$

$$Y_{i+1} = (B * X_i - A * X_{i+1}) \gg m,$$

[0044] 상기 ($>> m$) 은 반올림 없이 m 개 비트들만큼의 우측-쉬프트를 나타낸다.

[0045] 고정 소수점 변환 구현에서 우측-쉬프트는 그 변환을 나타내는데 이용된 캐스케이드 버터플라이 구조들의 각각에 반올림 오차 또는 비트-깊이의 손실을 도입한다. 이러한 비트-깊이의 손실에 대한 한가지 추가의 이유는 버터플라이 구조가 반올림 대신에 클리핑을 수행한다는 것이다. 클리핑은 각각의 버터플라이 스테이지에서 반올림을 수행하는 것이 계산적으로 비용이 들 수 있기 때문에 수행된다. 그러므로, 고정 소수점 변환의 변환 출력 신호에서 누적된 반올림 오차는 변환 사이즈에 따라 증가한다. 본 개시의 기술들은 고정 소수점 변환의 입력에 가변 로컬화 비트-깊이 증가를 적용시킴으로써 고정 소수점 변환에서 반올림 오차들을 경감시킨다.

[0046] 도 3 은 비디오 코딩과 연관된 변환을 위한 캐스케이드 버터플라이 구조들 (36) 을 도시하는 개념도이다. 일 실시예에서, 캐스케이드 버터플라이 구조들 (36) 은 1 차원 16-포인트 변환을 나타낼 수도 있다. 캐스케이드 버터플라이 구조들 (36) 은 변환의 단지 예시적인 표현일 뿐이다. 다른 실시예들에서, 버터플라이 구조들의 다수의 다른 변형들 및 조합들이 함께 캐스케이드되어 변환들을 나타낼 수도 있다. 캐스케이드 버터플라이 구조들 (36) 에 의해 표현된 변환은 순방향 또는 역방향 이산 코사인 변환 (DCT), 정수 변환, 웨이브릿 변환, 또는 다른 개념적으로 유사한 선형 변환을 포함할 수도 있다.

[0047] 도시된 실시예에서, 캐스케이드 버터플라이 구조들 (36) 은 16-포인트 변환을 나타내기 위해 함께 캐스케이드된, 도 2 의 버터플라이 구조 (34) 와 유사한 복수의 개별 버터플라이 구조들을 포함한다. 예를 들면, 캐스케이드 버터플라이 구조들 (36) 은 변환 입력 값들 X_0 내지 X_{15} 의 쌍들 사이의 버터플라이 구조들을 포함한다. 추가로, 캐스케이드 버터플라이 구조들 (36) 은 16-포인트 변환을 나타내기 위해 36 개 곱셈들을 이용한다. 다른 실시예들에서, 캐스케이드 버터플라이 구조들 (36) 은 상이한 배열들로 캐스케이드된, 변환 입력 값들의 상이한 쌍들 사이, 상이한 수의 변환 입력 값들 사이에 개별 버터플라이 구조들을 포함하며, 변환 입력 값들을 이용하여 상이한 수학적 연산들, 예컨대 덧셈 또는 곱셈을 수행한다.

[0048] 전술된 것과 같이, 캐스케이드 버터플라이 구조들 (36) 에 의해 표현된 고정 소수점 변환 구현은 변환 출력 신호를 우측-쉬프트하기 때문에 개별 버터플라이 구조들 각각에 반올림 오차 또는 비트-깊이 손실을 도입한다. 이러한 비트-깊이 손실의 하나의 추가의 이유는 버터플라이 구조들 각각이 반올림 대신 클리핑을 수행하기 때문이다. 도시된 실시예에서, 표현된 16-포인트 고정 소수점 변환은, 따라서, 변환을 나타내는데 이용된 캐스케이드 버터플라이 구조들의 수에 의존하는 누적된 반올림 오차를 변환 출력 신호에서 갖는다. 다른 실시예들에서, 고정 소수점 변환의 누적 반올림 오차는, 비디오 코딩 표준들이 32×32 , 64×64 , 또는 128×128 까지의 사이즈들의 변환들을 지원할 수도 있기 때문에, 심지어 더 클 수도 있다.

[0049] 추가로, 변환 사이즈가 증가함에 따라, 버터플라이 구조들 각각에서 가중 또는 곱셈 인자들, 예컨대, $W_i = A/2^m$ 은 감소될 수도 있고, 이는 이항 유리수들에서 m 의 값에서의 증가로서 표현될 수 있다. 증가된 m 값으로 인해 고정 소수점 변환의 버터플라이 구조들의 각각에서 더 큰 우측-쉬프트가 발생하며, 따라서 더 큰 누적된 반올림 오차가 발생한다. 비디오 인코딩 디바이스에서의 순방향 변환과 비디오 디코딩 디바이스에서의 역방향 변환 양자에서 비트-깊이의 연관된 손실은 디스플레이를 위한 비디오 블록들의 낮은 비트-정확도의 재구성들을 발생 할 수도 있다.

[0050] 본 개시의 기술들은 고정 소수점 변환에 가변 로컬화 비트-깊이 증가를 적용시킴으로써 고정 소수점 변환에서 반올림 오차들을 경감시킨다. 그 기술들은 고정 소수점 변환의 사이즈에 기초하여 상수 값, N 을 선택하고, 그 상수 값과 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 고정 소수점 변환에 적용하는 것을 포함한다. 상수 값은 복수의 상이한 변환 사이즈들에 대하여 미리 계산된 복수의 상수 값들로부터 선택될 수도 있다. 가변 로컬화 비트-깊이 증가를 적용하는 것은 고정 소수점 변환 이전의 상수 값과 동일한 비트 수만큼 변환 입력 신호, X 를 좌측-쉬프트 하고, 고정 소수점 변환 이후의 상수 값과 동일한 비트 수만큼 변환 출력 신호, Y 를 우측-쉬프트하는 것을 포함한다.

[0051] 가변 로컬화 비트-깊이 증가가 고정 소수점 변환에 적용될 경우, 변환 입력 신호는 $X' = X \ll N$ 으로 표현될 수도 있고, 상기 ($\ll N$) 는 N 개 비트들 만큼의 좌측-쉬프트를 나타내며, 변환 출력 신호는 $Y' = (Y + 2^{N-1}) \gg N$ 으로 표현될 수도 있어, 상기 2^{N-1} 은 오프셋을 나타내고, ($\gg N$) 은 N 개 비트만큼의 우측-쉬프트를 나타낸다. 다른 실시예들에서, 상이한 오프셋들이 N 개 비트만큼의 우측-쉬프트 이전에 변환 출력 신호에 적용될 수도 있다. 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들은 가변 로컬화 비트-깊이 증가와 함께 변환 출력 신호에 적용될 수도 있다. 상기 경우에, 오프셋들 및 우측-쉬프트들은 함께 결합될 수도 있다. 예를

들면, 추가의 오프셋이 2^{N-1} 의 값을 가지고 추가의 우측-쉬프트가 M 의 값을 갖는다면, 결합된 오프셋은 2^{M+N-1} 과 동일한 값을 가질 수도 있고, 결합된 우측-쉬프트는 (M+N) 과 동일한 값을 가질 수도 있다.

[0052] 고정 소수점 변환의 입력에 가변 로컬화 비트-깊이 증가를 적용하는 것은 변환에 대하여 더 높은 비트-깊이의 변환 입력 신호를 제공한다. 추가로, 고정 소수점 변환의 사이즈에 기초하여 계산된 값을 이용하여 가변 로컬화 비트-깊이 증가를 적용하는 것은 고정 소수점 변환 구현으로 인해 반올림 오차들을 감소시키거나 제거한다.

[0053] 전술된 것과 같이, 캐스케이드 버터플라이 구조들 (36) 은 1 차원 16-포인트 변환을 나타낼 수도 있다. 통상적으로, 비디오 코딩과 연관된 고정 소수점 변환들은 2 차원 변환들을 포함한다. 2 차원 변환은 비디오 블록 내의 픽셀 데이터의 행들에 제 1 스테이지 1 차원 변환을 적용한 후, 비디오 블록 내의 픽셀 데이터의 열들에 제 2 스테이지 1 차원 변환을 적용하거나, 그 반대로 적용함으로써 수행될 수도 있다. 예를 들면, 2 차원 16×16 변환은 캐스케이드 버터플라이 구조들 (36) 과 유사한 캐스케이드 버터플라이 구조들의 2 개 세트들로 표현될 수도 있다. 본 개시의 기술들은 2 차원 고정 소수점 변환의 일 스테이지의 사이즈에 기초하여 상수 값을 선택하고, 상수 값을 2 배가 되게 하고, 그리고 2 배된 상수 값을 이용하여 가변 로컬화 비트-깊이 증가를 2 차원 변환에 적용하는 것을 포함할 수도 있다. 2 배된 상수 값을 미만의 값을 이용하여 가변 로컬화 비트-깊이는 더 적은 상수 값을 반올림 오차들을 경감시키기에 충분히 클 경우 및/또는 더 적은 상수 값을 변환들을 구현하는데 필요한 비트-깊이를 제한하는데 필수적일 경우에 적용될 수도 있다. 다른 실시예들에서, 로컬화 비트-깊이 증가는 2 차원 변환의 각 스테이지에 적용될 수도 있고, 2 차원 변환의 일 스텝이지를 위해 선택된 상수 값을 동일한 값을 갖는다.

[0054] 도 4 는 변환 유닛 (52) 의 고정 소수점 구현에서 반올림 오차들을 경감시키기 위해 변환 유닛 (52) 에 가변 로컬화 비트-깊이 증가를 적용하는 기술들을 구현할 수도 있는 예시적인 비디오 인코더 (20A) 를 도시하는 블록도이다. 비디오 인코더 (20A) 는 도 1 을 참조하여 설명된 소스 디바이스 (12) 내의 비디오 인코더 (20) 의 일 실시예를 포함할 수도 있다.

[0055] 비디오 인코더 (20A) 는 매크로블록들, 또는 매크로블록들의 파티션들 또는 서브-파티션들을 포함하는 비디오 신호의 비디오 프레임들 내의 블록들의 인트라 코딩 및 인터 코딩을 수행할 수도 있다. 인트라 코딩은 소정의 비디오 프레임 내의 비디오 블록들에서 공간적 중복성을 감소 또는 제거하기 위하여 공간적 예측에 의존한다. 인터 코딩은 시간적 예측에 의존하여 비디오 시퀀스의 인접하는 프레임들 내의 비디오 블록들에서 시간적 중복성을 감소 또는 제거한다. 인트라 모드 (I-모드) 는 몇몇 공간 기반 압축 모드들 중 임의의 모드를 지칭할 수도 있고, 단방향성 예측 (P-모드), 양방향성 예측 (B-모드), 또는 일반화된 P/B 예측 (GPB-모드) 와 같은 인터 모드들은 몇몇 시간 기반 압축 모드들 중 임의의 모드를 지칭할 수도 있다.

[0056] 비디오 인코더 (20A) 는 인코딩될 비디오 신호의 비디오 프레임 내의 현재 비디오 블록을 수신한다. 도 4 의 실시예에서, 비디오 인코더 (20A) 는 예측 유닛 (40), 기준 프레임 메모리 (64), 가산기 (50), 변환 유닛 (52), 양자화 유닛 (54) 및 엔트로피 코딩 유닛 (56) 을 포함한다. 비디오 블록 재구성을 위해, 비디오 인코더 (20A) 는 또한 역 양자화 유닛 (58), 역변환 유닛 (60) 및 가산기 (62) 를 포함한다. 또한 재구성된 비디오로부터 블록화 아티팩트 (blockiness artifact) 들을 제거하기 위하여 블록 경계들을 필터링하는 디블록킹 필터 (deblocking filter; 도 4 에 미도시) 를 포함할 수도 있다. 원하는 경우, 디블록킹 필터는 통상적으로 가산기 (62) 의 출력을 필터링한다. 변환 유닛 (52) 의 고정 소수점 구현의 반올림 오차를 경감시키기 위해 변환 유닛 (52) 에 가변 로컬화 비트-깊이 증가를 적용하는 기술들에 따라, 비디오 인코더 (20A) 는 상수 값을 메모리 (43) 를 갖는 로컬화된 비트-깊이 증가 (LBDI) 유닛 (42) 을 포함한다.

[0057] 인코딩 프로세스 동안, 비디오 인코더 (20A) 는 코딩될 비디오 프레임 또는 슬라이스를 수신한다. 프레임 또는 슬라이스는 복수의 비디오 블록들로 분할될 수도 있다. 예측 유닛 (40) 은 시간 압축을 제공하기 위하여 하나 이상의 기준 프레임들 내의 하나 이상의 블록들에 대하여 수신된 비디오 블록의 인터 예측 코딩을 수행한다. 인트라 예측 유닛은 또한 공간 압축을 제공하기 위해 코딩될 블록과 동일한 프레임 또는 슬라이스 내의 하나 이상의 이웃하는 블록들에 대하여 수신된 비디오 블록의 인트라-예측 코딩을 수행할 수도 있다. 예측 유닛 (40) 은 예컨대 에러 결과들에 기초하여 코딩 모드들 중 하나, 인트라 코딩 모드 또는 인터 코딩 모드를 선택하기 위해 모드 선택을 수행할 수도 있고, 결과적인 인트라 또는 인터 코딩 블록을, 잔여 블록 데이터를 생성하기 위해 가산기 (50) 로, 및 기준 프레임으로 이용하기 위한 인코딩된 블록을 재구성하기 위해 가산기 (62) 로 제공한다.

[0058] 예측 유닛 (40) 은 모션 추정 및 모션 보상 양자를 수행할 수도 있다. 다른 실시예들에서, 모션 추정 및 모

선 보상은 개별 기능 유닛을 포함할 수도 있다. 모션 추정은 비디오 블록들을 위한 모션을 추정하는 모션 벡터들을 생성하는 프로세스이다. 모션 벡터는, 예를 들어, 현재 프레임 내에서 코딩되는 현재 블록 (또는 다른 코딩된 유닛)에 대하여 예측 기준 프레임 내의 예측 블록 (또는 다른 코딩된 유닛)의 변위를 나타낼 수 있다. 예측 블록은 절대 차이의 합 (SAD), 제곱 차이의 합 (SSD), 또는 다른 차이 메트릭들에 의해 결정될 수도 있는, 픽셀 차이와 관련하여 코딩될 블록을 밀접하게 매칭하도록 발견된 블록이다. 모션 벡터는 또한 매크로블록의 파티션의 변위를 나타낼 수도 있다. 모션 보상은 모션 추정에 의해 결정된 모션 벡터에 기초하여 예측 블록을 인출 (fetch)하거나 생성하는 것을 수반할 수도 있다.

[0059] 모션 추정을 수행할 경우, 예측 유닛 (40)은 인터 코딩된 프레임의 비디오 블록을, 기준 프레임 메모리 (64) 내의 기준 프레임의 비디오 블록들과 비교함으로써, 그 비디오 블록에 대한 모션 벡터를 계산한다. 예측 유닛 (40)은 엔트로피 코딩 유닛 (56)으로 계산된 모션 벡터를 송신한다. 모션 벡터에 의해 식별된 기준 프레임 블록은 예측 블록으로 지정될 수도 있다. 예측 유닛 (40)은 기준 프레임의 예측 블록에 대한 에러 값들을 계산한다. 예측 유닛 (40)은 또한 예측 블록에 기초하여 예측 데이터를 계산할 수도 있다. 비디오 인코더 (20A)는 코딩되고 있는 원래의 비디오 블록에서 예측 유닛 (40)으로부터의 예측 데이터를 감산함으로써 잔여 비디오 블록을 형성한다. 가산기 (50)는 이러한 감산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다.

[0060] 변환 유닛 (52)은 변환, 이를테면 이산 코사인 변환 (DCT), 정수 변환, 웨이브릿 변환, 또는 개념적으로 유사한 변환을 잔여 비디오 블록에 적용하여 잔여 변환 계수 값을 포함하는 비디오 블록을 생성한다. 변환 유닛 (52)은 변환을 잔여 블록에 적용하며, 잔여 변환 계수들의 블록을 생성한다. 변환 유닛 (51)은 잔여 정보를 픽셀 값 도메인으로부터 변환 도메인, 예컨대 주파수 도메인으로 변환할 수도 있다.

[0061] 전술된 것과 같이, 변환 유닛 (52)의 고정 소수점 구현은 변환 유닛 (52)의 고정 소수점 변환의 사이즈에 대응하는 다수의 캐스케이드 버터플라이 구조들의 우측-쉬프트들로 인해 그 변환 출력 신호에 반올림 오차들 또는 비트-깊이의 손실을 도입할 수도 있다. 변환 유닛 (52)의 고정 소수점 구현에서 반올림 오차들을 감소시키거나 제거하기 위해, LBDI 유닛 (42)은 상수 값 메모리 (43)로부터 변환 유닛 (52)의 고정 소수점 변환의 사이즈에 기초하여 상수 값을 선택하며, 그 상수 값과 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 변환 유닛 (52)에 적용한다. 더욱 상세하게는, LBDI 유닛 (42)은 상수 값과 동일한 비트 수만큼, 변환 유닛 (52)에 입력되는 잔여 비디오 블록 데이터에 좌측-쉬프트 (44)를 도입한다. LBDI 유닛 (42)은 또한 상수 값과 동일한 비트 수만큼, 변환 유닛 (52)으로부터 출력되는 잔여 변환 계수들에 우측-쉬프트 (45)를 적용한다. LBDI 유닛 (42)은 또한 잔여 변환 계수들에 오프셋을 적용한 후에 오프셋된 잔여 변환 계수들에 우측-쉬프트 (45)를 적용할 수도 있다. 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들이 변환 출력 신호에 적용될 수도 있고, 가변 로컬화 비트-깊이 증가의 오프셋 및 우측-쉬프트와 결합될 수도 있다.

[0062] 잔여 비디오 블록들의 비트-깊이를 증가시키는 것은 비트 정확도를 증가시키지만, 또한 높은 비트 정확도의 비디오 블록들을 처리하기 위한 계산 복잡도를 증가시킨다. 그러므로, LBDI 유닛 (42)은 변환 유닛 (52)의 이전 및 이후에 직접 좌측-쉬프트 (44) 및 우측-쉬프트 (45)를 적용하여 가변 로컬화 비트-깊이 증가가 변환 유닛 (52)에 로컬화되는 것을 유지한다. 이러한 방식으로, 높은 복잡도의 프로세싱이 변환 유닛 (52)에서 요구될 수도 있지만, 비디오 인코더 (20A) 내의 나머지 기능 유닛들에서는 요구되지 않는다.

[0063] LBDI 유닛 (42) 내의 상수 값 메모리 (43)는 비디오 인코더 (20A)에 의해 이용된 비디오 코딩 표준에 의해 지원되는 복수의 상이한 변환 사이즈들에 대한 복수의 상수 값을 저장한다. 상수 값 메모리 (43)에 저장된 상수 값을 각각은 복수의 상이한 변환 사이즈들 중 하나에 대하여 미리 계산된다. 이러한 방식으로, LBDI 유닛 (42)은 임의의 사이즈의 고정 소수점 변환들에서 반올림 오차들을 효율적으로 감소시키기 위해 변환 유닛 (52)에 의해 적용된 각각의 상이한 변환 사이즈에 대하여 상이한 상수 값을 선택할 수도 있다. 예를 들면, H.264 (AVC) 표준은 4×4 및 8×8 의 사이즈들의 변환들을 지원할 수도 있다. 최신의 HEVC 표준은 16×16 , 32×32 , 64×64 , 또는 128×128 까지의 사이즈들의 정사각 및/또는 직사각 변환들을 지원할 수도 있다.

[0064] 상수 값 메모리 (43)에 저장된 복수의 상이한 변환 사이즈들 중 하나에 대한 상수 값을 각각은 비디오 신호의 입력 비트-깊이 값 B_i , 고정 소수점 변환에서 오버플로우를 회피하는데 필요한 추가 비트 수인 변환 비트-깊이 값 B_{TR} , 및 고정 소수점 변환과 연관된 변환 깊이-정확도 값 Z 에 기초하여 미리 계산된다. 상수 값을 N 각각은 식 $B_i + B_{TR} + N \leq Z$ 를 만족하도록 미리 계산될 수도 있다.

[0065] 일부 경우에, 상수 값은, 그 상수 값과 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 적용함으로써 비디

오 인코더 (20A)에 의해 지원되는 전체 비트-깊이 보다 큰 비트-깊이를 가지는 변환 입력 신호를 생성하도록, 큰 값을 가지도록 미리 계산될 수도 있다. 예를 들면, 비디오 인코더 (20A)가 32-비트 인코더를 포함한다면, 상수 값이 미리 계산될 수도 있고, 그 결과 32 비트보다 큰 비트-깊이를 가지는 변환 입력 신호를 생성한다. 이러한 경우에, 상수 값은 변환 입력 신호의 비트-깊이가 비디오 인코더 (20A)에 대한 전체 비트-깊이를 초과하지 않도록 보장하기 위해 상수 값 메모리 (43)에 저장되기 전에 감소될 수도 있다.

[0066] 복수의 상이한 변환 사이즈들 중 하나에 대한 상수 값들의 각각을 미리 계산하기 위해, 변환 비트-정확도 값 Z은 비디오 인코더 및/또는 비디오 디코더 양자에서 고정 소수점 변환의 구현에 기초하여 선택된다. 예를 들면, 변환 비트-정확도 값은 16, 32 또는 64 비트들과 동일할 수도 있다. 변환 비트-정확도 값은 비디오 코딩 디바이스들 내에서 구현되는, 고정 소수점 변환의 사이즈 및 그 변환과 연관된 저항들의 사이즈에 의존할 수도 있다.

[0067] 추가로, 복수의 상이한 변환 사이즈들 중 하나에 대한 상수 값들 각각을 미리 계산하기 위해, 고정 소수점 변환에서 오버플로우를 회피하는데 필요한 변환 비트-깊이 값 B_{TR} 은 고정 소수점 변환의 사이즈에 기초하여 결정된다. 더욱 상세하게, 변환 비트-깊이 값은 고정 소수점 변환 유닛을 나타내는 캐스케이드 버터플라이 구조들의 각각에서 변환 출력 값들의 최악의 경우 분석에 기초하여 결정될 수도 있다. 변환 출력들, Y_i 및 Y_{i+1} 에 대한 가능한 값들의 범위는 개별 버터플라이 구조들의 각각에서 콥셈 인자들, W_i 및 W_{i+1} 에 기초하여 계산될 수도 있다. 전체 고정 소수점 변환의 변환 출력 신호에 대한 누적 비트 범위는 고정 소수점 변환에서 오버플로우를 회피하는데 필요한 최대 수의 추가 비트들, 즉 변환 비트-깊이 값, B_{TR} 을 제공한다.

[0068] 일부 실시예들에서, 변환 유닛 (52)은 2 차원 고정 소수점 변환을 포함할 수도 있다. 예를 들면, 변환 유닛 (52)은 잔여 비디오 블록들 내의 픽셀 데이터의 행들에 제 1 스테이지 1 차원 변환을 적용하고, 그 후에 잔여 비디오 블록들 내의 픽셀 데이터의 열들에 제 2 스테이지 1 차원 변환을 적용하거나, 그 반대로 적용함으로써 2 차원 변환을 수행할 수도 있다. 일 실시예로서, 변환 유닛 (52)은 16×16 고정 소수점 변환을 포함할 수도 있다. LBDI 유닛 (42)은 상수 값 메모리 (43)에 저장된 복수의 상수 값들로부터 변환 유닛 (52)의 고정 소수점 변환을 위한 사이즈 16×16 에 기초하여 상수 값을 선택할 수도 있다. 사이즈 16×16 에 대하여 저장된 상수 값은 예컨대, 12 비트들과 동일할 수도 있다. 일 실시예로서, 저장된 상수 값은 비디오 신호의 입력 비트-깊이 값 B_i 이 8 비트와 동일하고, 변환 비트-정확도 값 Z 이 32 비트와 동일하고, 변환 비트-깊이 값 B_{TR} 이 12 비트와 동일한, 사이즈 16×16 변환에 대한 미리 계산에 기초할 수도 있다. 상수 값 N은 식 $B_i + B_{TR} + N \leq Z$ 을 만족하도록 12 비트와 동일하게 미리 계산된다.

[0069] 이러한 실시예에서, LBDI 유닛 (42)은 변환 유닛 (52)에 입력되는 잔여 비디오 블록들에 12 비트 좌측-쉬프트 (44)를 적용한다. LBDI 유닛 (42)은 유사하게 변환 유닛 (52)으로부터 출력되는 잔여 변환 계수들에 12 비트 우측-쉬프트 (45)를 적용한다. LBDI (42)는 먼저 변환 유닛 (52)으로부터 출력되는 잔여 변환 계수들에 오프셋을 적용한 후에, 오프셋된 잔여 변환 계수들에 우측-쉬프트 (45)를 적용할 수도 있다. 일 실시예에서, LBDI (42)는 2^{N-1} 과 동일한 오프셋을 적용할 수도 있고, 여기서 N은 상수 값이다. 다른 실시예들에서, 우측-쉬프트 (45)를 적용하기 전에 변환 유닛 (52)으로부터 출력된 잔여 변환 계수들에 상이한 오프셋들이 적용될 수도 있다. 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들은 변환 출력 신호와 함께 가변로컬화 비트-깊이 증가에 적용될 수도 있다. 이러한 경우에, 오프셋들 및 우측-쉬프트들은 함께 결합될 수도 있다. 예를 들어, 추가의 오프셋이 2^{M-1} 의 값을 가지고, 추가의 우측-쉬프트가 M의 값을 가진다면, 결합된 오프셋은 2^{M+N-1} 과 동일한 값을 가질 수도 있고, 결합된 우측-쉬프트는 $(M+N)$ 과 동일한 값을 가질 수도 있다

[0070] 일부 경우에, 상수 값 메모리 (43)는 1 차원 변환 사이즈들에 대응하는 상수 값들을 저장할 수도 있다. 변환 유닛 (52)이 2 차원 고정 소수점 변환을 포함할 경우, LBDI 유닛 (42)은 변환 유닛 (52)의 2 차원 고정 소수점 변환의 일 스테이지의 사이즈에 기초하여 상수 값을 선택하고, 그 상수 값을 2 배가 되도록 하여 2 차원 고정 소수점 변환의 양자의 스테이지들을 고려한다. 그 후에, LBDI 유닛 (42)은 2 배된 상수 값 이하의 값을 이용하여 가변로컬화 비트-깊이 증가를 변환 유닛 (52)에 적용할 수도 있다. 2 배된 상수 값 미만의 값을 이용하여 가변로컬화 비트-깊이는 더 작은 상수 값이 반올림 오차들을 경감시키기에 충분히 큰 경우 및/또는 더 작은 상수 값이 변환들을 구현하는데 필요한 비트-깊이를 한정하는데 필요할 경우 적용될 수도 있다. 다른 실시예들에서, LBDI 유닛 (42)은 2 차원 고정 소수점 변환의 일 스테이지에 대하여 선택된 상수 값과 동일한 값을 이용하여 로컬화 비트-깊이 증가를 변환 유닛 (52)의 2 차원 고정 소수점 변환의 각 스테이지에

적용할 수도 있다.

[0071] LBDI 유닛 (42)은 변환을 위해 더 높은 비트-깊이의 변환 입력 신호를 제공하도록 변환 유닛 (52)에 가변 로컬화 비트-깊이 증가를 적용함으로써 변환 유닛 (52)에서 반올림 오차들을 감소시키거나 제거할 수도 있다.

변환 유닛 (52)에 입력되는 더 높은 비트-깊이의 잔여 비디오 블록 데이터는 변환 유닛 (52)의 고정 소수점 변환을 나타내는데 이용된 캐스케이드 버터플라이 구조들에 의해 수행된 우측-쉬프트의 영향들을 경감시킨다.

또한, 변환 유닛 (52)의 고정 소수점 변환의 사이즈에 기초하여 계산된 값을 이용하여 가변 로컬화 비트-깊이 증가를 적용하는 것은 잔여 비디오 블록 데이터에 적용된 특정 고정 소수점 변환으로 인한 변환 유닛 (52)에서의 반올림 오차들을 더 효율적으로 감소시킨다.

[0072] 양자화 유닛 (54)은 잔여 변환 계수들을 양자화하여 비트 레이트를 더 감소시킨다. 양자화 프로세스는 계수들의 일부 또는 전부와 연관된 비트 깊이를 감소시킬 수도 있다. 양자화 정도는 양자화 파라미터를 조정함으로써 수정될 수도 있다. 특히, 양자화 유닛 (54)은 매크로블록 또는 파티션에서 변환 계수들을 스캔하기 위해 지그재그 스캔 또는 다른 스캔 패턴을 적용할 수도 있다. 양자화 다음에, 엔트로피 코딩 유닛 (56)은 양자화된 변환 계수들을 엔트로피 코딩한다. 예컨대, 엔트로피 코딩 유닛 (56)은 CAVLC (content adaptive variable length coding), CABAC (context adaptive binary arithmetic coding), 또는 다른 엔트로피 코딩 기술을 수행할 수도 있다. 엔트로피 코딩 유닛 (56)에 의한 엔트로피 코딩 다음에, 인코딩된 비디오는 다른 디바이스로 송신되거나 또는 나중의 송신 또는 취출을 위해 보관될 수도 있다. 콘텍스트 적응형 이진 산술 코딩의 경우, 콘텍스트는 이웃하는 매크로블록들에 기초할 수도 있다.

[0073] 일부 경우에, 엔트로피 코딩 유닛 (56) 또는 비디오 인코더 (20A)의 다른 유닛은 엔트로피 코딩 이외에 다른 코딩 기능들을 수행하도록 구성될 수도 있다. 예를 들면, 엔트로피 코딩 (56)은 매크로블록들 및 파티션들에 대한 CBP 값을 결정하도록 구성될 수도 있다. 또한, 일부 경우에 엔트로피 코딩 유닛 (56)은 매크로블록 또는 그 파티션에서 계수들의 런 렌스 코딩을 수행할 수도 있다. 엔트로피 코딩 유닛 (56)은 또한 인코딩된 비디오 비트스트림에서 송신을 위해 적절한 선택스 앤리먼트들을 갖는 헤더 정보를 구성할 수도 있다.

[0074] 역 양자화 유닛 (58) 및 역변환 유닛 (60)은 역 양자화 및 역변환을 각각 적용하여 예를 들면 참조 블록으로서 나중에 사용하기 위해 픽셀 도메인에서 잔여 블록을 재구성한다. 변환 유닛 (52)을 참조하여 전술된 프로세싱과 유사하게, LBDI 유닛 (42)은 상수 값을 메모리 (43)로부터 역변환 유닛 (60)의 고정 소수점 역변환의 사이즈에 기초하여 상수 값을 선택하고, 그 상수 값을 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 역변환 유닛 (60)에 적용한다.

[0075] 일부 실시예들에서, LBDI 유닛 (42)은 변환 유닛 (52)에서와 역변환 유닛 (60)에서 가변 로컬화 비트-깊이 증가를 위한 상이한 상수 값을 선택할 수도 있다. 일부 경우에, 역변환 유닛 (60)에 의해 적용되는 고정 소수점 역변환은 변환 유닛 (52)에 의해 적용되는 고정 소수점 순변환과 상이한 사이즈를 갖는다. 다른 경우에, 비디오 인코더 (20A)의 디코더 부분은 비디오 인코더 (20A)의 인코더 부분 보다 더 낮은 비트 정확도 값을 가질 수도 있거나, 역 양자화 유닛 (58)은 역변환 유닛 (60)에서의 반올림 오차들이 변환 유닛 (52)에서 보다 덜 심각하도록 좌측-쉬프트를 생성할 수도 있다. 그러므로, 상수 값을 메모리 (43)는 상이한 비트 정확도 값을 및 상이한 역-양자화 매트릭스들을 갖는 코딩 디바이스들에 대하여 상이한 상수 값을 세트들을 저장할 수도 있다. 그러나, LBDI 유닛 (42)은 드리프트를 회피하기 위해 대응하는 비디오 디코더에서 역변환에서의 가변 로컬화 비트-깊이 증가와 동일한, 역변환 유닛 (60)에서의 가변 로컬화 비트-깊이 증가를 위한 상수 값을 선택한다.

[0076] LBDI 유닛 (42)은 역변환 유닛 (60)에 입력되는 잔여 변환 계수들에 상수 값을 동일한 비트 수만큼 좌측-쉬프트 (46)를 도입하고, 역변환 유닛 (60)으로부터 출력된 잔여 비디오 블록 데이터에 상수 값을 동일한 비트 수만큼 우측-쉬프트 (47)를 적용한다. LBDI (42)는 또한 역변환 유닛 (60)으로부터 출력된 잔여 비디오 블록 데이터에 오프셋을 적용한 후에, 오프셋된 잔여 비디오 블록 데이터에 우측-쉬프트 (47)를 적용할 수도 있다. 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들은 변환 출력 신호에 적용될 수도 있고, 가변 로컬화 비트-깊이 증가의 오프셋 및 우측-쉬프트와 결합될 수도 있다. 다시 말해서, LBDI 유닛 (42)은 역변환 유닛 (60) 이전 및 이후에 직접 좌측-쉬프트 (46) 및 우측-쉬프트 (47)를 적용하여 가변 로컬화 비트-깊이 증가가 역변환 유닛 (60)에 로컬화되는 것을 유지할 수도 있다.

[0077] 예측 유닛 (40)은 기준 프레임 메모리 (64)의 프레임들 중 하나의 예측 블록에 잔여 블록을 추가함으로써 기준 블록을 계산할 수도 있다. 예측 유닛 (40)은 또한 재구성된 잔여 블록에 하나 이상의 보간 필터들을 적용하여 모션 추정에서 이용하기 위한 서브-정수 (sub-integer) 픽셀 값을 계산할 수도 있다. 가산기 (62)

는 재구성된 잔여 블록을 예측 유닛 (40)에 의해 생성되는 모션 보상된 예측 블록에 가산하여 기준 프레임 메모리 (64)에 저장하기 위한 재구성된 비디오 블록을 생성한다. 재구성된 비디오 블록은 예측 유닛 (40)에 의해 참조 블록으로서 사용되어 후속 비디오 프레임에서 블록을 인터 인코딩할 수도 있다.

[0078] 도 5는 역변환 유닛 (78)의 고정 소수점 구현에서 반올림 오차들을 경감시키기 위해 역변환 유닛 (78)에 가변 로컬화 비트-깊이 증가를 적용하는 기술을 구현할 수도 있는 비디오 디코더 (30A)의 일 실시예를 도시하는 블록도이다. 비디오 디코더 (30A)는 도 1을 참조하여 설명된 목적지 디바이스 (14) 내의 비디오 디코더 (30)의 일 실시예를 포함할 수도 있다.

[0079] 비디오 디코더 (30A)는 도 4의 비디오 인코더 (20A)와 같은 비디오 인코더로부터의 비트스트림에서 수신된 인코딩된 비디오 시퀀스를 디코딩한다. 도 5의 실시예에서, 비디오 디코더 (30A)는 엔트로피 디코딩 유닛 (70), 예측 유닛 (72), 역양자화 유닛 (76), 역변환 유닛 (78), 기준 프레임 메모리 (82) 및 가산기 (80)를 포함한다. 역변환 유닛 (78)의 고정 소수점 구현에서 반올림 오차들을 경감시키기 위해 역변환 유닛 (78)에 가변 로컬화 비트-깊이 증가를 적용하는 기술들에 따르면, 비디오 디코더 (30A)는 또한 상수 값 메모리 (85)를 갖는 로컬화 비트-깊이 증가 (LBDI) 유닛 (84)을 포함한다.

[0080] 일부 실시예들에서, 비디오 디코더 (30A)는 (도 4의) 비디오 인코더 (20A)와 관련하여 설명된 인코딩 패스와 일반적으로 상호적인 디코딩 패스를 수행한다. 예측 유닛 (72)은 엔트로피 디코딩 유닛 (70)으로부터 수신된 모션 벡터들에 기초하여 예측 데이터를 생성할 수도 있다. 예측 유닛 (72)은 기준 프레임 메모리 (82) 내의 기준 프레임들에서 예측 블록을 식별하기 위해 비트스트림에서 수신된 모션 벡터들을 이용할 수도 있다. 예측 유닛 (72)은 또한 공간적으로 인접하는 블록들로부터 예측 블록을 형성하기 위해 비트스트림에서 수신된 인트라 예측 모드들을 이용할 수도 있다. 역양자화 유닛 (76)은 비트스트림에서 제공되고 엔트로피 디코딩 유닛 (70)에 의해 디코딩된 양자화된 블록 계수들을 역 양자화, 즉 탈-양자화한다. 역양자화 프로세스는 예컨대, H.264 디코딩 표준에 의해 정의된 것과 같은 종래의 프로세스를 포함할 수도 있다. 역양자화 프로세스는 또한 양자화 정도 및 유사하게 적용되어야 하는 역양자화 정도를 결정하기 위해 각각의 매크로블록에 대하여 인코더 (20A)에 의해 계산된 양자화 파라미터의 이용을 포함할 수도 있다.

[0081] 역변환 유닛 (78)은 잔여 비디오 블록을 생성하기 위해 역 DCT, 역 정수 변환, 역 웨이브릿 변환, 또는 개념적으로 유사한 역 선형 변환과 같은 역변환을 변환 계수들에 적용한다. 역변환 유닛 (78)은 변환 계수들을 변환 도메인, 예컨대 주파수 도메인으로부터 픽셀 값 도메인으로 변환할 수도 있다.

[0082] 도 4의 변환 유닛 (52)과 관련하여 전술된 것과 같이, 역변환 유닛 (78)의 고정 소수점 구현은 역변환 유닛 (78)의 고정 소수점 역변환의 사이즈에 의존하는, 반올림 오차들 또는 비트-깊이의 손실을 그 변환 출력 신호에 도입할 수도 있다. 역변환 유닛 (78)의 고정 소수점 구현에서 반올림 오차들을 감소시키거나 제거하기 위해, LBDI 유닛 (84)은 상수 값 메모리 (85)로부터 역변환 유닛 (78)의 고정 소수점 역변환의 사이즈에 기초하여 상수 값을 선택하며, 그 상수 값과 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 역변환 유닛 (78)에 적용하기 위해 도 4의 LBDI 유닛 (42)과 실질적으로 유사한 기술들을 수행한다.

[0083] 더욱 상세하게는, LBDI 유닛 (84)은 상수 값과 동일한 비트 수만큼, 역변환 유닛 (78)에 입력되는 변환 계수들에 좌측-쉬프트 (86)를 도입한다. LBDI 유닛 (84)은 또한 상수 값과 동일한 비트 수만큼, 역변환 유닛 (78)으로부터 출력되는 잔여 비디오 블록들에 우측-쉬프트 (87)를 적용한다. LBDI 유닛 (84)은 또한 역변환 유닛 (78)으로부터 출력된 잔여 비디오 블록들에 오프셋을 적용한 후에 오프셋된 잔여 비디오 블록들에 우측-쉬프트 (87)를 적용할 수도 있다. 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들이 변환 출력 신호에 적용될 수도 있고, 가변 로컬화 비트-깊이 증가의 오프셋 및 우측-쉬프트와 결합될 수도 있다. LBDI 유닛 (84)은 역변환 유닛 (78)의 이전 및 이후에 직접 좌측-쉬프트 (86) 및 우측-쉬프트 (87)를 적용하여 가변 로컬화 비트-깊이 증가가 역변환 유닛 (78)에 로컬화되는 것을 유지한다.

[0084] LBDI 유닛 (84) 내의 상수 값 메모리 (85)는 비디오 디코더 (30A)에 의해 이용된 비디오 코딩 표준에 의해 지원되는 복수의 상이한 역변환 사이즈들에 대한 복수의 상수 값을 저장한다. 상수 값 메모리 (85)에 저장된 상수 값을 각각은 복수의 상이한 역변환 사이즈들 중 하나에 대하여 미리 계산된다. 이러한 방식으로, LBDI 유닛 (84)은 임의의 사이즈의 고정 소수점 역변환들에서 반올림 오차들을 효율적으로 감소시키기 위해 역변환 유닛 (78)에 의해 적용된 각각의 상이한 역변환 사이즈에 대하여 상이한 상수 값을 선택한다.

[0085] 상수 값 메모리 (85)에 저장된 복수의 상이한 변환 사이즈들 중 하나에 대한 상수 값을 각각은 비디오 신호의 입력 비트-깊이 값 B_i , 고정 소수점 역변환에서 오버플로우를 회피하는데 필요한 추가 비트 수인 변환 비트-깊이

값 B_{TR} , 및 고정 소수점 역변환과 연관된 변환 깊이-정확도 값 Z 에 기초하여 미리 계산된다. 상수 값들 N 각각은 식 $B_i + B_{TR} + N \leq Z$ 를 만족하도록 미리 계산된 상수 값일 수도 있다.

[0086] 일부 경우에, 상수 값은, 그 상수 값과 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 적용함으로써 비디오 디코더 (30A)에 의해 지원되는 전체 비트-깊이 보다 큰 비트-깊이를 가지는 변환 입력 신호를 생성하도록, 큰 값을 가지도록 미리 계산될 수도 있다. 예를 들면, 비디오 디코더 (30A)가 32-비트 디코더를 포함한다면, 상수 값이 미리 계산될 수도 있고, 그 결과 32 비트보다 큰 비트-깊이를 가지는 변환 입력 신호를 생성한다. 이러한 경우에, 상수 값은 변환 입력 신호의 비트-깊이가 비디오 디코더 (30A)에 대한 전체 비트-깊이를 초과하지 않도록 보장하기 위해 상수 값 메모리 (85)에 저장되기 전에 감소될 수도 있다.

[0087] 일부 경우에, 비디오 디코더 (30A)는 역변환 유닛 (78)에서 가변 로컬화 비트-깊이 증가를 위한 상수 값을 선택하기 위해 비디오 인코더 (20A)로부터 명시적으로 시그널링된 정보를 수신할 수도 있다. 예를 들면, 비디오 디코더 (30A)는 비디오 인코더 (20A)의 역변환 유닛 (60)에 적용된 가변 로컬화 비트-깊이 증가를 위해 LBDI 유닛 (42)에 의해 선택된 상수 값을 명시적으로 수신할 수도 있다. LBDI 유닛 (84)이 인코더와 디코더 사이의 드리프트를 회피하기 위해 비디오 인코더 (20A) 내의 역변환 유닛 (60)에서의 가변 로컬화 비트-깊이 증가와 동일한, 역변환 유닛 (78)에서의 가변 로컬화 비트-깊이 증가를 위한 상수 값을 선택하는 것은 중요하다.

[0088] 일부 실시예들에서, 역변환 유닛 (78)은 2 차원 고정 소수점 변환을 포함할 수도 있다. 일 실시예로서, 역변환 유닛 (78)은 16×16 고정 소수점 역변환을 포함할 수도 있다. LBDI 유닛 (84)은 상수 값 메모리 (85)에 저장된 복수의 상수 값들로부터 역변환 유닛 (78)의 고정 소수점 역변환을 위한 사이즈 16×16 에 기초하여 상수 값을 선택할 수도 있다. 사이즈 16×16 에 대하여 저장된 상수 값은 예컨대, 12 비트들과 동일 할 수도 있다. 일 실시예로서, 저장된 상수 값은 비디오 신호의 입력 비트-깊이 값 B_i 이 8 비트와 동일하고, 변환 비트-정확도 값 Z 이 32 비트와 동일하고, 변환 비트-깊이 값 B_{TR} 이 12 비트와 동일한, 사이즈 16×16 변환에 대한 미리 계산에 기초할 수도 있다. 상수 값 N 은 식 $B_i + B_{TR} + N \leq Z$ 를 만족하도록 12 비트와 동일하게 미리 계산된다.

[0089] 이러한 실시예에서, LBDI 유닛 (84)은 역변환 유닛 (78)에 입력되는 변환 계수들에 12 비트 좌측-쉬프트 (86)를 적용한다. LBDI 유닛 (84)은 유사하게 역변환 유닛 (78)으로부터 출력되는 잔여 비디오 블록들에 12 비트 우측-쉬프트 (87)를 적용한다. LBDI 유닛 (84)는 먼저 역변환 유닛 (78)으로부터 출력되는 잔여 비디오 블록들에 오프셋을 적용한 후에, 오프셋된 잔여 비디오 블록들에 우측-쉬프트 (87)를 적용할 수도 있다.

일 실시예에서, LBDI 유닛 (84)은 2^{N-1} 과 동일한 오프셋을 적용할 수도 있고, 여기서 N 은 상수 값이다. 다른 실시예들에서, 우측-쉬프트 (87)를 적용하기 전에 역변환 유닛 (78)으로부터 출력된 잔여 비디오 블록들에 상이한 오프셋들이 적용될 수도 있다. 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들은 변환 출력 신호와 함께 가변 로컬화 비트-깊이 증가에 적용될 수도 있다. 이러한 경우에, 오프셋들 및 우측-쉬프트들을 함께 결합될 수도 있다. 예를 들어, 추가의 오프셋이 2^{M-1} 의 값을 가지고, 추가의 우측-쉬프트가 M 의 값을 가진다면, 결합된 오프셋은 2^{M+N-1} 과 동일한 값을 가질 수도 있고, 결합된 우측-쉬프트는 $(M+N)$ 과 동일한 값을 가질 수도 있다.

[0090] 일부 경우에, 상수 값 메모리 (85)는 1 차원 변환 사이즈들에 대응하는 상수 값을 저장할 수도 있다. 역변환 유닛 (78)이 2 차원 고정 소수점 변환을 포함할 경우, LBDI 유닛 (84)은 역변환 유닛 (78)의 2 차원 고정 소수점 변환의 일 스테이지의 사이즈에 기초하여 상수 값을 선택하고, 그 상수 값을 2 배가 되도록 하여 2 차원 고정 소수점 변환의 양자의 스테이지들을 고려한다. 그 후에, LBDI 유닛 (84)은 2 배된 상수 값을 이하의 값을 이용하여 가변 로컬화 비트-깊이 증가를 역변환 유닛 (78)에 적용할 수도 있다. 2 배된 상수 값 미만의 값을 이용하여 가변 로컬화 비트-깊이는 더 작은 상수 값이 반올림 오차들을 경감시키기에 충분히 큰 경우 및/또는 더 작은 상수 값이 변환들을 구현하는데 필요한 비트-깊이를 한정하는데 필요할 경우 적용될 수도 있다. 다른 실시예들에서, LBDI 유닛 (84)은 2 차원 고정 소수점 변환의 일 스테이지에 대하여 선택된 상수 값과 동일한 값을 이용하여 로컬화 비트-깊이 증가를 역변환 유닛 (78)의 2 차원 고정 소수점 변환의 각 스테이지에 적용할 수도 있다.

[0091] 예측 유닛 (72)은 보간 필터들에 기초하여 보간을 수행할 수 있는 모션 보상 블록들을 생성한다. 서브-픽셀 정확도를 갖는 모션 보상을 위해 이용될 보간 필터들에 대한 식별자들은 싱크스 엘리먼트들에 포함될 수도

있다. 예측 유닛 (72) 은 기준 블록의 서브-정수 픽셀들에 대한 보간된 값들을 계산하기 위해 비디오 블록의 인코딩 동안 비디오 인코더 (20A) 에 의해 이용된 것과 같은 보간 필터들을 이용할 수도 있다. 예측 유닛 (72) 은 수신된 신택스 정보에 따라 비디오 인코더 (20A) 에 의해 이용된 보간 필터들을 결정하며, 그 보간 필터들을 이용하여 예측 블록들을 생성할 수도 있다.

[0092] 예측 유닛 (72) 은 인코딩된 비디오 시퀀스의 프레임(들)을 인코딩하는데 이용된 매크로블록들의 사이즈들, 인코딩된 비디오 시퀀스의 일 프레임의 각각의 매크로블록이 분할되는 방식을 설명하는 파티션 정보, 각각의 파티션이 인코딩되는 방식을 나타내는 모드들, 각각의 인터-인코딩된 매크로블록 또는 파티션에 대한 하나 이상의 기준 프레임들 (또는 리스트들), 및 인코딩된 비디오 시퀀스를 디코딩하기 위한 다른 정보를 결정하기 위해 신택스 정보의 일부를 이용할 수도 있다.

[0093] 가산기 (80) 는 예측 유닛 (72) 에 의해 생성된 대응하는 예측 블록들과 잔여 블록들을 합산하여 디코딩된 블록들을 형성한다. 원하는 경우에, 디블록킹 필터는 또한 블록화 아티팩트들을 제거하기 위해 디코딩된 블록들을 필터링하도록 적용될 수도 있다. 그 후에, 디코딩된 비디오 블록들은 후속 모션 보상을 위해 기준 블록 드를 제공하고, (도 1 의 디스플레이 디바이스 (32) 와 같은) 디스플레이 디바이스 상의 프리젠테이션을 위한 디코딩된 비디오를 생성하는 기준 프레임 메모리 (82) 에 저장된다.

[0094] 도 6 은 내부 비트-깊이 증가 (IBDI) 가 이용되고, 또한 변환 유닛 (52) 에 가변 로컬화 비트-깊이 증가를 적용하는 기술들을 구현할 수도 있는 예시적인 비디오 인코더 (20B) 를 도시하는 블록도이다. 비디오 인코더 (20B) 는 도 1 을 참조하여 설명된 소스 디바이스 (12) 내의 비디오 인코더 (20) 의 일 실시예를 포함할 수도 있다. 추가로, 비디오 인코더 (20B) 는 도 4 의 비디오 인코더 (20A) 와 실질적으로 유사하게 동작하고, IBDI 에 대한 변경들을 가질 수도 있다.

[0095] 도 4 의 비디오 인코더 (20A) 와 유사하게, 비디오 인코더 (20B) 는 예측 유닛 (40), 기준 프레임 메모리 (64), 가산기 (50), 변환 유닛 (52), 양자화 유닛 (54) 및 엔트로피 코딩 유닛 (56) 을 포함한다. 비디오 블록 재구성을 위해, 비디오 인코더 (20B) 는 또한 역 양자화 유닛 (58), 역변환 유닛 (60) 및 가산기 (62) 를 포함한다. 변환 유닛 (52) 의 고정 소수점 구현에서 반올림 오차를 경감시키기 위해 변환 유닛 (52) 에 가변 로컬화 비트-깊이 증가를 적용하는 기술들에 따라, 비디오 인코더 (20B) 는 상수 값 메모리 (43) 를 갖는 로컬화된 비트-깊이 증가 (LBDI) 유닛 (42) 을 포함한다.

[0096] 도 6 의 도시된 실시예에서, 비디오 인코더 (20B) 는 비디오 인코더 (20B) 가 임의의 비디오 블록 코딩을 수행하기 전에 IBDI 를 비디오 신호에 적용하는 내부 비트 깊이 증가 (IBDI) 유닛 (90) 을 포함한다. 일 실시예에서, IBDI 유닛 (90) 은 더 높은 비트 정확도의 비디오 코딩을 수행하기 위해 비디오 신호의 비트 깊이를 8 비트의 입력 비트 깊이로부터 12 비트로 증가시키기 위해 4 비트 IBDI 를 비디오 신호에 적용할 수도 있다. 비디오 신호에 IBDI 가 적용될 경우, 높은 복잡도와 높은 비트 정확도의 비디오 프로세싱이 비디오 인코더 (20B) 내의 모든 기능 유닛에서 요구된다. IBDI 의 애플리케이션이 비디오 신호의 더 정확한 인코딩을 발생하지만, IBDI (90) 는 높은 비트-깊이 신호를 처리하고 버퍼링할 수 있는 인코더에만 IBDI 를 적용할 수도 있다. 또한, IBDI (90) 가 비디오 신호에 IBDI 를 적용할 경우에, 비디오 인코더 (20B) 는 IBDI 의 값을 도 7 의 비디오 디코더 (30B) 와 같은 연관된 비디오 디코더에 명시적으로 시그널링하여 비디오 디코더가 디코딩된 비디오 신호를 다시 그 입력 비트-깊이로 가져오기 위해 IBDI 를 제거할 수 있게 한다.

[0097] 변환 유닛 (52) 의 고정 소수점 구현에서 반올림 오차들을 감소시키거나 제거하기 위해, LBDI 유닛 (42) 은 상수 값 메모리 (43) 로부터 변환 유닛 (52) 의 고정 소수점 변환의 사이즈에 기초하여 상수 값을 선택하며, IBDI 유닛 (90) 에 의해 적용된 IBDI 의 값을 기초하여 상수 값을 조정한다. 그 후에, LBDI 유닛 (42) 은 조정된 상수 값을 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 변환 유닛 (52) 에 적용한다. 더욱 상세하게는, LBDI 유닛 (42) 은 조정된 상수 값을 동일한 비트 수만큼, 변환 유닛 (52) 에 입력되는 잔여 비디오 블록 데이터에 좌측-쉬프트 (94) 를 도입한다. LBDI 유닛 (42) 은 또한 조정된 상수 값을 동일한 비트 수만큼, 변환 유닛 (52) 으로부터 출력되는 잔여 변환 계수들에 우측-쉬프트 (95) 를 적용한다. LBDI 유닛 (42) 은 변환 유닛 (52) 으로부터 출력된 잔여 변환 계수들에 오프셋을 적용한 후에 오프셋된 잔여 변환 계수들에 우측-쉬프트 (95) 를 적용할 수도 있다. 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들이 변환 출력 신호에 적용될 수도 있고, 가변 로컬화 비트-깊이 증가의 오프셋 및 우측-쉬프트와 결합될 수도 있다.

[0098] 도 4 와 관련하여 전술된 것과 같이, LBDI 유닛 (42) 내의 상수 값 메모리 (43) 는 비디오 인코더 (20B) 에 의해 이용된 비디오 코딩 표준에 의해 지원되는 복수의 상이한 변환 사이즈들에 대한 복수의 상수 값을 저장한다. 상수 값 메모리 (43) 에 저장된 상수 값을 각각은 복수의 상이한 변환 사이즈들 중 하나에 대하여 미리

계산된다. IBDI 가 비디오 신호에 적용될 경우, LBDI 유닛 (42) 은 상수 값 메모리 (43) 로부터의 변환 유닛 (52) 의 고정 소수점 변환의 사이즈에 기초하여 상수 값을 선택하는 것을 계속한다. 그러나, 상수 값이 선택되면, LBDI (42) 는 IBDI 유닛 (90) 에 의해 적용된 IBDI 의 값만큼 상수 값을 감소시킬 수도 있다.

[0099] 그러나, LBDI 유닛 (42) 은 오직 상수 값을 음이 아닌 값으로 감소시킬 수도 있고, 그렇지 않으면 상수 값을 0 과 동일하게 세팅한다. 음의 상수 값을 LBDI 유닛 (42) 이 변환 유닛 (42) 에 입력되는 잔여 비디오 블록들에 좌측-쉬프트 (94) 대신 우측 쉬프트를 적용하여 변환 출력 신호에서 반올림 오차들이 심지어 더 커지게 할 것이다. 그러므로, LBDI 유닛 (42) 은 상수 값 N 을 IBDI 유닛 (90) 에 의해 적용된 IBDI 의 값 B_d 만큼 감소시켜, IBDI 의 값이 상수 값 이하라면, 즉 $(N-B_d) \geq 0$ 이라면, 조정된 상수 값이 $N-B_d$ 와 동일하게 되도록 한다. IBDI 의 값 B_d 이 상수 값 N 보다 크면, 즉 $(N-B_d) < 0$ 이라면, LBDI 유닛 (42) 은 상수 값 N 을 0 으로 감소시킨다. 이 경우, LBDI 유닛 (42) 은 변환 유닛 (52) 에 어떤 가변 로컬화 비트-깊이 증가도 적용하지 않는다.

[0100] 일 실시예로서, IBDI 유닛 (90) 은 비디오 인코더 (20B) 에 입력되는 비디오 신호에 4 비트 IBDI 를 적용할 수도 있고, 변환 유닛 (52) 은 고정 소수점 16×16 변환을 포함할 수도 있다. LBDI 유닛 (42) 은 상수 값 N 을 사이즈 16×16 변환을 위한 8 비트와 동일하게 선택할 수도 있다. 그 후에, LBDI 유닛 (42) 은 상수 값 N 을 IBDI 유닛 (90) 에 적용된 IBDI 의 값 B_d 만큼 감소시켜 조정된 상수 값이 4 비트와 동일하게 한다. 이러한 실시예에서, LBDI 유닛 (42) 은 변환 유닛 (52) 에 입력되는 잔여 비디오 블록들에 4 비트 좌측-쉬프트 (94) 를 적용하고, 변환 유닛 (52) 으로부터 출력되는 잔여 변환 계수들에 4 비트 우측-쉬프트 (95) 를 적용한다. 다른 실시예에서, IBDI 유닛 (90) 은 비디오 신호에 10 비트 IBDI 를 적용할 수도 있다. 이 경우, LBDI 유닛 (42) 은 상수 값 N 을 0 으로 감소시키며, 이는 상수 값 N 을 IBDI 의 값 B_d 만큼 감소시키는 것이 음의 값을 발생하기 때문이다. 이러한 실시예에서, LBDI 유닛 (42) 은 변환 유닛 (52) 에 어떤 가변 로컬화 비트-깊이 증가도 적용하지 않는다.

[0101] 변화 유닛 (52) 을 참조하여 전술된 프로세싱과 유사하게, LBDI 유닛 (42) 은 역변환 유닛 (60) 의 고정 소수점 역변환의 사이즈에 기초하여 상수 값을 선택하며, IBDI 유닛 (90) 에 의해 적용된 IBDI 의 값에 기초하여 상수 값을 조정한다. 그 후에, LBDI 유닛 (42) 은 조정된 상수 값과 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 역변환 유닛 (60) 에 적용한다. 일부 실시예들에서, LBDI 유닛 (42) 은 변환 유닛 (52) 에 보다 역변환 유닛 (60) 에 가변 로컬화 비트-깊이 증가를 적용함, 따라서 상이한 조정된 상수 값을 발생한다. 그러나, LBDI 유닛 (42) 이 드리프트를 회피하기 위해 대응하는 비디오 디코더의 역변환에서의 가변 로컬화 비트-깊이 증가와 동일한, 역변환 유닛 (60) 에서의 가변 로컬화 비트-깊이 증가에 대한 상수 값을 선택하여 조정하는 것은 중요하다.

[0102] LBDI 유닛 (42) 은 역변환 유닛 (60) 에 입력되는 잔여 변환 계수들에 상수 값과 동일한 비트 수만큼 좌측-쉬프트 (96) 를 도입하고, 역변환 유닛 (60) 으로부터 출력된 잔여 비디오 블록 데이터에 상수 값과 동일한 비트수 만큼 우측-쉬프트 (97) 를 적용한다. LBDI 유닛 (42) 은 또한 역변환 유닛 (60) 으로부터 출력된 잔여 비디오 블록 데이터에 오프셋을 적용한 후에, 오프셋된 잔여 비디오 블록 데이터에 우측-쉬프트 (97) 를 적용할 수도 있다. 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들은 변환 출력 신호에 적용될 수도 있고, 가변 로컬화 비트-깊이 증가의 오프셋 및 우측-쉬프트와 결합될 수도 있다.

[0103] 도 7 은 내부 비트-깊이 증가 (IBDI) 가 이용되고, 또한 역변환 유닛 (78) 에 가변 로컬화 비트-깊이 증가를 적용하는 기술을 구현할 수도 있는 비디오 디코더 (30B) 의 일 실시예를 도시하는 블록도이다. 비디오 디코더 (30B) 는 도 1 을 참조하여 설명된 목적지 디바이스 (14) 내의 비디오 디코더 (30) 의 일 실시예를 포함할 수도 있다. 추가로, 비디오 디코더 (30B) 는 도 5 의 비디오 디코더 (30A) 와 실질적으로 유사하게 동작하고, IBDI 에 대한 변경들을 가질 수도 있다.

[0104] 도 5 의 비디오 디코더 (30A) 와 유사하게, 비디오 디코더 (30B) 는 엔트로피 디코딩 유닛 (70), 예측 유닛 (72), 역양자화 유닛 (76), 역변환 유닛 (78), 기준 프레임 메모리 (82) 및 가산기 (80) 를 포함한다. 역변환 유닛 (78) 의 고정 소수점 구현에서 반올림 오차들을 경감시키기 위해 역변환 유닛 (78) 에 가변 로컬화 비트-깊이 증가를 적용하는 기술들에 따르면, 비디오 디코더 (30B) 는 또한 상수 값 메모리 (85) 를 갖는 LBDI 유닛 (84) 을 포함한다.

[0105] 비디오 디코더 (30B) 는 도 6 의 비디오 인코더 (20B) 와 같은 비디오 인코더로부터의 비트스트림에서 수신된 내부 비트-깊이 증가 (IBDI) 를 갖는 인코딩된 비디오 시퀀스를 디코딩한다. 일부 경우에, 비디오 디코더

(30B) 는 역변환 유닛 (78)에서 가변 로컬화 비트-깊이 증가를 위한 상수 값을 선택하고 조정하기 위해 비디오 인코더 (20B)로부터 명시적으로 시그널링된 정보를 수신할 수도 있다. 예를 들면, 비디오 디코더 (30B)는 비디오 인코더 (20B)의 역변환 유닛 (60)에 적용된 가변 로컬화 비트-깊이 증가를 위한 상수 값을 조정하기 위해 LBDI 유닛 (42)에 의해 선택된 상수 값 및 LBDI 유닛 (42)에 의해 이용된 IBDI의 값을 명시적으로 수신할 수도 있다. LBDI 유닛 (84)이 인코더와 디코더 사이의 드리프트를 회피하기 위해 비디오 인코더 (20A) 내의 역변환 유닛 (60)에서의 가변 로컬화 비트-깊이 증가와 동일한, 역변환 유닛 (78)에서의 가변 로컬화 비트-깊이 증가를 위한 상수 값을 선택하고 조정하는 것은 중요하다.

[0106] 도 7의 도시된 실시예에서, 비디오 디코더 (30B)는 비디오 디코더 (30B)가 비디오 블록 디코딩을 수행한 후에 역 IBDI를 디코딩된 비디오 신호에 적용하는 역 IBDI 유닛 (100)을 포함한다. 역 IBDI 유닛 (100)은 비디오 인코더 (20B) 내의 IBDI 유닛 (90)에 의해 비디오 신호에 적용된 IBDI의 명시적으로 수신된 값과 동일한 값을 이용하여 역 IBDI를 디코딩된 비디오 신호에 적용할 수도 있다. 이러한 방식으로, 비디오 디코더 (30B)는 IBDI를 제거하여 디코딩된 비디오 신호를 다시 그 입력 비트-깊이에 가져올 수도 있다. 일 실시예에서, 역 IBDI 유닛 (100)은 디코딩된 비디오 신호의 비트 깊이를 12비트로부터 8비트의 입력 비트 깊이로 감소시키기 위해 4비트 역 IBDI를 디코딩된 비디오 신호에 적용할 수도 있다.

[0107] 역변환 유닛 (78)의 고정 소수점 구현에서 반올림 오차들을 감소시키거나 제거하기 위해, LBDI 유닛 (82)은 상수 값을 메모리 (85)로부터 역변환 유닛 (78)의 고정 소수점 역변환의 사이즈에 기초하여 상수 값을 선택하며, 비디오 인코더 (20B) 내의 IBDI 유닛 (90)에 의해 적용되고 비디오 디코더 (30B)로 명시적으로 시그널링되는 IBDI의 값에 기초하여 상수 값을 조정한다. 그 후에, LBDI 유닛 (84)은 조정된 상수 값을 동일한 값을 이용하여 가변 로컬화 비트-깊이 증가를 역변환 유닛 (78)에 적용한다.

[0108] 더욱 상세하게는, LBDI 유닛 (84)은 조정된 상수 값과 동일한 비트 수만큼, 역변환 유닛 (78)에 입력되는 변환 계수들에 좌측-ши프트 (106)를 도입한다. LBDI 유닛 (84)은 또한 조정된 상수 값과 동일한 비트 수만큼, 역변환 유닛 (78)으로부터 출력되는 잔여 비디오 블록들에 우측-ши프트 (107)를 적용한다. LBDI 유닛 (84)은 또한 역변환 유닛 (78)으로부터 출력된 잔여 비디오 블록들에 오프셋을 적용한 후에 오프셋된 잔여 비디오 블록들에 우측-ши프트 (107)를 적용할 수도 있다. 일부 경우에, 추가의 오프셋들 및 우측-ши프트들이 변환 출력 신호에 적용될 수도 있고, 가변 로컬화 비트-깊이 증가의 오프셋 및 우측-ши프트와 결합될 수도 있다.

[0109] 도 5와 관련하여 전술된 것과 같이, LBDI 유닛 (84) 내의 상수 값을 메모리 (85)는 비디오 디코더 (30B)에 의해 이용된 비디오 코딩 표준에 의해 지원되는 복수의 상이한 변환 사이즈들에 대한 복수의 상수 값을 저장한다. 상수 값을 메모리 (85)에 저장된 상수 값을 각각은 복수의 상이한 역변환 사이즈들 중 하나에 대하여 미리 계산된다. IBDI가 비디오 신호에 적용될 경우, LBDI 유닛 (84)은 상수 값을 메모리 (85)로부터의 역변환 유닛 (78)의 고정 소수점 역변환의 사이즈에 기초하여 상수 값을 선택하는 것을 계속한다. 그러나, 상수 값이 선택되면, LBDI (84)는 IBDI 유닛 (90)에 의해 적용되고 비디오 디코더 (30B)로 명시적으로 시그널링되는 IBDI의 값만큼 상수 값을 감소시킬 수도 있다.

[0110] 그러나, LBDI 유닛 (84)은 오직 상수 값을 음이 아닌 값으로 감소시킬 수도 있고, 그렇지 않으면 상수 값을 0과 동일하게 세팅한다. 음의 상수 값을 LBDI 유닛 (84)이 좌측-ши프트 (106) 대신 역변환 유닛 (78)에 입력되는 변환 계수들에 우측-ши프트를 적용하여 변환 출력 신호에서 반올림 오차들이 더 커지게 할 것이다. 그러므로, LBDI 유닛 (84)은 상수 값 N 을 IBDI의 값 B_d 만큼 감소시켜, IBDI의 값이 상수 값 이하라면, 즉 $(N-B_d) \geq 0$ 이라면, 조정된 상수 값이 $N-B_d$ 와 동일하게 되도록 한다. IBDI의 값 B_d 이 상수 값 N 보다 크면, 즉 $(N-B_d) < 0$ 이라면, LBDI 유닛 (84)은 상수 값 N 을 0으로 감소시킨다. 이 경우, LBDI 유닛 (84)은 역변환 유닛 (78)에 어떤 가변 로컬화 비트-깊이 증가도 적용하지 않는다.

[0111] 일 실시예로서, IBDI 유닛 (90)은 비디오 인코더 (20B) 내의 비디오 신호에 적용된 4비트 IBDI를 가질 수도 있고, 역변환 유닛 (78)은 고정 소수점 16×16 변환을 포함할 수도 있다. LBDI 유닛 (84)은 상수 값을 N 을 사이즈 16×16 변환을 위한 8비트와 동일하게 선택할 수도 있다. 그 후에, LBDI 유닛 (84)은 상수 값을 N 을 IBDI의 값 B_d 만큼 감소시켜 조정된 상수 값이 4비트와 동일하게 한다. 이러한 실시예에서, LBDI 유닛 (84)은 역변환 유닛 (78)에 입력되는 변환 계수들에 4비트 좌측-ши프트 (106)를 적용하고, 역변환 유닛 (78)으로부터 출력되는 잔여 비디오 블록들에 4비트 우측-ши프트 (95)를 적용한다. 다른 실시예에서, IBDI 유닛 (90)은 비디오 인코더 (20B) 내의 비디오 신호에 적용된 10비트 IBDI를 가질 수도 있다. 이

경우, LBDI 유닛 (84)은 상수 값 N 을 0으로 감소시키며, 이는 상수 값 N 을 IBDI의 값 B_d 만큼 감소시키는 것이 음의 값을 발생하기 때문이다. 이러한 실시예에서, LBDI 유닛 (84)은 역변환 유닛 (78)에 어떤 가변 로컬화 비트-깊이 증가도 적용하지 않는다.

[0112] 도 8은 변환 유닛의 고정 소수점 구현에서 반올림 오차들을 경감시키기 위해 변환 유닛에 가변 로컬화 비트-깊이 증가를 적용하는 예시적인 방법을 도시하는 흐름도이다. 도 8에 도시된 방법은 도 4로부터의 비디오 인코더 (20A) 및 도 6으로부터의 비디오 인코더 (20B)와 관련하여 설명될 것이다.

[0113] 비디오 인코더 (20A 또는 20B)는 LBDI 유닛 (42) 내의 상수 값 메모리 (43)에 복수의 상이한 변환 사이즈들에 대한 복수의 상수 값을 저장한다 (130). 전술된 것과 같이, 복수의 상이한 변환 사이즈들 중 하나와 연관된 상수 값을 각각은 비디오 신호의 입력 비트-깊이 값 B_i , 변환 사이즈와 연관된 변환 비트-깊이 값 B_{TR} , 및 변환 사이즈와 연관된 변환 깊이-정확도 값 Z 에 기초하여 미리 계산된다. 예를 들어, 각각의 상수 값 N 은 식 $B_i + B_{TR} + N \leq Z$ 를 만족하도록 미리 계산될 수도 있다.

[0114] 그 후에, 비디오 인코더 (20A 또는 20B)는 인코딩될 비디오 신호를 수신한다 (132). 비디오 신호를 수신하면, 비디오 인코더 (20A 또는 20B)의 LBDI 유닛 (42)은 변환 유닛 (52)에 의해 적용될 고정 소수점 변환의 사이즈를 결정하고, 고정 소수점 변환의 사이즈에 기초하여 상수 값 메모리 (53)에 저장된 상수 값을 선택한다 (134).

[0115] 그 후에, LBDI 유닛 (42)은 내부 비트-깊이 증가 (IBDI)가 수신된 비디오 신호에 적용되는지 여부를 결정한다 (136). IBDI가 적용되지 않았다면 (136의 아니오), 도 4로부터의 비디오 인코더 (20A)의 실시예에서와 같이, 변환 유닛 (52) 바로 이전에, LBDI 유닛 (42)은 변환 유닛 (52)에 입력되는 잔여 비디오 블록들에 좌측-쉬프트 (44)를, 선택된 상수 값과 동일한 비트 수만큼 적용한다 (140). 그 후에, 변환 유닛 (52)은 좌측-쉬프트된 잔여 비디오 블록 데이터를 변환 계수들로 변환한다 (142). 변환 유닛 (52) 바로 이후에, LBDI 유닛 (42)은 변환 유닛 (52)으로부터 출력되는 변환 계수들에 오프셋을 적용할 수도 있다 (144). 예를 들어, LBDI (42)는 2^{N-1} 과 동일한 오프셋을 변환 계수들에 적용할 수도 있다. 그 후에, LBDI 유닛 (42)은 선택된 상수 값과 동일한 비트 수만큼 오프셋 변환 계수들에 우측-쉬프트 (45)를 적용한다 (146).

[0116] 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들은 변환 출력 신호와 함께 가변 로컬화 비트-깊이 증가에 적용될 수도 있다. 이러한 경우에, 오프셋들 및 우측-쉬프트들은 함께 결합될 수도 있다. 예를 들어, 추가의 오프셋이 2^{M-1} 의 값을 가지고, 추가의 우측-쉬프트가 M 의 값을 가진다면, 결합된 오프셋은 2^{M+N-1} 과 동일한 값을 가질 수도 있고, 결합된 우측-쉬프트는 $(M+N)$ 과 동일한 값을 가질 수도 있다.

[0117] IBDI가 적용되었다면 (136의 예), 도 6으로부터의 비디오 인코더 (20B)의 일 실시예에서와 같이, LBDI 유닛 (42)은 IBDI 유닛 (90)에 의해 비디오 신호에 적용된 IBDI의 값을 결정한다. 그 후에, LBDI 유닛 (42)은 IBDI의 값을 기초하여 선택된 상수 값을 조정한다 (138). 이 경우에, 변환 유닛 (52)의 고정 소수점 변환의 사이즈에 대하여 선택된 상수 값 N 은 조정된 상수 값이 $N - B_d$ 와 동일하도록 IBDI의 값 B_d 만큼 감소될 것이다.

[0118] LBDI 유닛 (42)은 오직 선택된 상수 값을 음이 아닌 값으로 감소시킬 수도 있다. 예를 들면, LBDI 유닛 (42)은 IBDI의 값을 선택된 상수 값 이하이면, $(N - B_d) \geq 0$ 이 되도록, 선택된 상수 값을 IBDI의 값만큼 감소시킨다. IBDI의 값을 선택된 상수 값보다 크면, $(N - B_d) < 0$ 이 되도록, LBDI 유닛 (42)은 선택된 상수 값을 0으로 감소시킨다.

[0119] 변환 유닛 (52) 바로 이전에, LBDI 유닛 (42)은 조정된 상수 값과 동일한 비트 수만큼 변환 유닛 (52)에 입력되는 잔여 비디오 블록 데이터에 좌측-쉬프트 (94)를 적용한다 (140). 변환 유닛 (52)은 좌측-쉬프트된 잔여 비디오 블록 데이터를 변환 계수들로 변환한다 (142). 변환 유닛 (52) 바로 이후에, LBDI 유닛 (42)은 변환 유닛 (52)으로부터 출력되는 변환 계수들에 오프셋을 적용할 수도 있다 (144). 예를 들어, LBDI (42)는 2^{N-1} 과 동일한 오프셋을 변환 계수들에 적용할 수도 있다. 그 후에, LBDI 유닛 (42)은 조정된 상수 값과 동일한 비트 수만큼 오프셋 변환 계수들에 우측-쉬프트 (95)를 적용할 수도 있다 (146).

[0120] 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들은 변환 출력 신호와 함께 가변 로컬화 비트-깊이 증가에 적용될 수도 있다. 이러한 경우에, 오프셋들 및 우측-쉬프트들은 함께 결합될 수도 있다. 예를 들어, 추가

의 오프셋이 2^{M-1} 의 값을 가지고, 추가의 우측-쉬프트가 M 의 값을 가진다면, 결합된 오프셋은 2^{M+N-1} 과 동일한 값을 가질 수도 있고, 결합된 우측-쉬프트는 (M+N) 과 동일한 값을 가질 수도 있다.

[0121] LBDI 유닛 (42) 은 변환을 위한 더 높은 비트-깊이의 변환 입력 신호를 제공하기 위해 변환 유닛 (52) 에 가변 로컬화 비트-깊이 증가를 적용함으로써 변환 유닛 (52) 의 반올림 오차들을 감소시키거나 제거할 수도 있다.

변환 유닛 (52) 에 입력되는 잔여 비디오 블록 데이터의 더 높은 비트-깊이는 변환 유닛 (52) 의 고정 소수점 변환을 나타내는데 이용된 캐스케이드 버터플라이 구조들에 의해 수행된 우측-쉬프트의 양만큼 덜 오염된다.

추가로, 변환 유닛 (52) 의 고정 소수점 변환의 사이즈에 기초하여 계산된 값을 이용하여 가변 로컬화 비트-깊이 증가를 적용하는 것은 잔여 비디오 블록 데이터에 적용된 특정 고정 소수점 변환으로 인해 변환 유닛 (52) 의 반올림 오차를 더 효율적으로 감소시킨다.

[0122] 도 9 는 역변환 유닛의 고정 소수점 구현에서 반올림 오차들을 경감시키기 위해 역변환 유닛에 가변 로컬화 비트-깊이 증가를 적용하는 예시적인 방법을 도시하는 흐름도이다. 도 9 에 도시된 방법은 도 5로부터의 비디오 디코더 (30A) 및 도 7로부터의 비디오 디코더 (30B) 와 관련하여 설명될 것이다.

[0123] 비디오 디코더 (30A 또는 30B) 는 LBDI 유닛 (84) 내의 상수 값 메모리 (85) 에 복수의 상이한 변환 사이즈들에 대한 복수의 상수 값들을 저장한다 (150). 전술된 것과 같이, 복수의 상이한 변환 사이즈들 중 하나와 연관된 상수 값들 각각은 비디오 신호의 입력 비트-깊이 값 B_i , 변환 사이즈와 연관된 변환 비트-깊이 값 B_{TR} , 및 변환 사이즈와 연관된 변환 깊이-정확도 값 Z에 기초하여 미리 계산된다. 예를 들어, 각각의 상수 값 N 은 식 $B_i + B_{TR} + N \leq Z$ 를 만족하도록 미리 계산될 수도 있다.

[0124] 그 후에, 비디오 디코더 (30A 또는 30B) 는 디코딩된 비디오 신호를 수신한다 (152). 인코딩된 비디오 신호를 수신하면, 비디오 디코더 (30A 또는 30B) 의 LBDI 유닛 (84) 은 역변환 유닛 (78) 에 의해 적용될 고정 소수점 변환의 사이즈를 결정하고, 고정 소수점 변환의 사이즈에 기초하여 상수 값 메모리 (85) 에 저장된 상수 값을 선택한다 (154).

[0125] 그 후에, LBDI 유닛 (84) 은 내부 비트-깊이 증가 (IBDI) 가 비디오 인코더에서 비디오 신호에 적용되었는지 여부를 결정한다 (156). IBDI 가 비디오 신호에 적용된 경우에, 비디오 인코더는 IBDI 의 값을 비디오 디코더 (30A 또는 30B) 로 명시적으로 시그널링 할 수도 있다. IBDI 가 적용되지 않았다면 (156 의 아니오), 도 5로부터의 비디오 디코더 (30A) 의 실시예에서와 같이, 역변환 유닛 (78) 바로 이전에, LBDI 유닛 (84) 은 역변환 유닛 (78) 에 입력되는 변환 계수들에 좌측-쉬프트 (86) 를, 선택된 상수 값과 동일한 비트 수만큼 적용한다 (160). 그 후에, 역변환 유닛 (78) 은 좌측-쉬프트된 변환 계수들을 잔여 비디오 블록 데이터로 변환한다 (162). 역변환 유닛 (78) 바로 이후에, LBDI 유닛 (84) 은 역변환 유닛 (78) 으로부터 출력되는 잔여 비디오 블록 데이터에 오프셋을 적용할 수도 있다 (164). 예를 들어, LBDI (84) 는 2^{N-1} 과 동일한 오프셋을 잔여 비디오 블록 데이터에 적용할 수도 있다. 그 후에, LBDI 유닛 (84) 은 선택된 상수 값과 동일한 비트 수만큼 오프셋된 잔여 비디오 블록 데이터에 우측-쉬프트 (87) 를 적용한다 (166).

[0126] 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들은 변환 출력 신호와 함께 가변 로컬화 비트-깊이 증가에 적용될 수도 있다. 이러한 경우에, 오프셋들 및 우측-쉬프트들은 함께 결합될 수도 있다. 예를 들어, 추가의 오프셋이 2^{M-1} 의 값을 가지고, 추가의 우측-쉬프트가 M 의 값을 가진다면, 결합된 오프셋은 2^{M+N-1} 과 동일한 값을 가질 수도 있고, 결합된 우측-쉬프트는 (M+N) 과 동일한 값을 가질 수도 있다.

[0127] IBDI 가 적용되었다면 (156 의 예), 도 7로부터의 비디오 디코더 (30B) 의 일 실시예에서와 같이, LBDI 유닛 (84) 은 비디오 인코더 (20B) 에 의해 명시적으로 시그널링된 것과 같은 IBDI 의 값을 결정한다. 그 후에, LBDI 유닛 (84) 은 IBDI 의 값에 기초하여 선택된 상수 값을 조정한다 (158). 이 경우에, 변환 유닛 (52) 의 고정 소수점 변환의 사이즈에 대하여 선택된 상수 값 N 은 조정된 상수 값이 $N - B_d$ 와 동일하도록 IBDI 의 값 B_d 만큼 감소될 것이다.

[0128] LBDI 유닛 (84) 은 오직 선택된 상수 값을 음이 아닌 값으로 감소시킬 수도 있다. 예를 들면, LBDI 유닛 (84) 은 IBDI 의 값이 선택된 상수 값 이하이면, $(N-B_d) \geq 0$ 이 되도록, 선택된 상수 값을 IBDI 의 값만큼 감소시킨다. IBDI 의 값이 선택된 상수 값보다 크면, $(N-B_d) < 0$ 이 되도록, LBDI 유닛 (84) 은 선택된 상수 값을 0 으로 감소시킨다.

[0129] 역변환 유닛 (78) 바로 이전에, LBDI 유닛 (84) 은 조정된 상수 값과 동일한 비트 수만큼 역변환 유닛 (78) 에 입력되는 변환 계수들에 좌측-쉬프트 (106) 를 적용한다 (160). 그 후에, 역변환 유닛 (78) 은 좌측-쉬프트 된 변환 계수들을 잔여 비디오 블록 데이터로 변환한다 (162). 역변환 유닛 (84) 바로 이후에, LBDI 유닛 (84) 은 역변환 유닛 (78) 으로부터 출력되는 잔여 비디오 블록에 오프셋을 적용할 수도 있다 (164). 예를 들어, LBDI 유닛 (84) 은 2^{N-1} 과 동일한 오프셋을 잔여 비디오 블록 데이터에 적용할 수도 있다. 그 후에, LBDI 유닛 (84) 은 조정된 상수 값과 동일한 비트 수만큼 오프셋된 잔여 비디오 블록 데이터에 우측-쉬프트 (107) 를 적용한다 (166).

[0130] 일부 경우에, 추가의 오프셋들 및 우측-쉬프트들은 변환 출력 신호와 함께 가변 로컬화 비트-깊이 증가에 적용될 수도 있다. 이러한 경우에, 오프셋들 및 우측-쉬프트들은 함께 결합될 수도 있다. 예를 들어, 추가의 오프셋이 2^{M-1} 의 값을 가지고, 추가의 우측-쉬프트가 M 의 값을 가진다면, 결합된 오프셋은 2^{M+N-1} 과 동일한 값을 가질 수도 있고, 결합된 우측-쉬프트는 $(M+N)$ 과 동일한 값을 가질 수도 있다.

[0131] LBDI 유닛 (84) 은 변환을 위한 더 높은 비트-깊이의 변환 입력 신호를 제공하기 위해 역변환 유닛 (78) 에 가변 로컬화 비트-깊이 증가를 적용함으로써 역변환 유닛 (78) 의 반올림 오차를 감소시키거나 제거할 수도 있다. 역변환 유닛 (78) 에 입력되는 변환 계수들의 더 높은 비트-깊이는 역변환 유닛 (78) 의 고정 소수점 역변환을 나타내는데 이용된 캐스케이드 버터플라이 구조들에 의해 수행된 우측-쉬프트의 양만큼 덜 오염된다. 추가로, 역변환 유닛 (78) 의 고정 소수점 역변환의 사이즈에 기초하여 계산된 값을 이용하여 가변 로컬화 비트-깊이 증가를 적용하는 것은 변환 계수들에 적용된 특정 고정 소수점 변환으로 인해 역변환 유닛 (78) 의 반올림 오차를 더 효율적으로 감소시킨다.

[0132] 하기에 제시된 표 1 및 표 2 는 내부 비트-깊이 증가 (IBDI) 가 적용되지 않을 경우에 선형 변환에 로컬화된 비트-깊이 증가를 적용한 초기 결과들을 제공한다. 표 1 및 표 2 양자는 IBDI 가 적용될 경우에 로컬화된 비트-깊이 증가 없이 및 IBDI 가 적용되지 않을 경우에 로컬화된 비트-깊이 증가로 몇몇 비디오 시퀀스들에 대한 비트-레이트 감소 퍼센트율을 제공한다. 표 1 은 저-지연 고효율, 즉 일반화된 P/B (GPB) 구성에 대한 결과들을 제시하고, 표 2 는 랜덤 액세스 고효율, 즉 계층적-B 구성에 대한 결과들을 제시한다. 로컬화된 비트-깊이 증가에 의한 평균 비트-레이트 감소는 저지연 고효율 구성 (표 1) 에 대하여 1.26% 이고, 랜덤 액세스 고효율 구성 (표 2) 에 대하여 1.03% 이다.

표 1

저-지연 고효율 구성에 대한 가변 로컬화 비트-깊이 증가로 인한 비트-레이트 감소 [%]

		IBDI 온	IBDI 오프, 로컬화된 비트-깊이 증가
WQVGA	RaceHorses	-1.88	-0.38
	Basketball	-2.29	-0.90
	PartyScene	-2.18	-0.11
	BQSquare	-2.51	-0.18
	WQVGA_AVG	-2.22	-0.39
WVGA	RaceHorses	-2.54	-0.79
	Basketball	-4.51	-1.29
	PartyScene	-2.14	-0.40
	BQSquare	-4.52	-0.96
	WVGA_AVG	-3.43	-0.86
720p	Vidvo1	-11.84	-1.85
	Vidvo3	-12.26	-1.07
	Vidvo4	-12.80	-2.03
	720p_AVG	-12.30	-1.65
1080p	ParkScene	-4.11	-0.94
	Kimono	-4.79	-3.22
	Basketball	-7.01	-3.39
	Cactus	-4.64	-1.75
	BQSquare	-5.91	-0.93
	1080p_AVG	-5.29	-2.05
	AVG	-5.37	-1.26

표 2

[0134]

랜덤 액세스 고효율 구성에 대한 가변 로컬화 비트-깊이 증가로 인한 비트-레이트 감소 [%]

		IBDI 온	IBDI 오프, 로컬화된 비트-깊이 증가
WQVGA	RaceHorses	-1.04	-0.29
	Basketball	-1.37	-0.50
	PartyScene	-1.65	-0.35
	BQSquare	-2.28	-0.39
	WQVGA_AVG	-1.59	-0.38
WVGA	RaceHorses	-1.55	-0.54
	Basketball	-3.42	-1.10
	PartyScene	-1.72	-0.32
	BQSquare	-2.60	-0.55
	WVGA_AVG	-2.32	-0.63
720p	Vidyo1	-8.38	-1.89
	Vidyo3	-6.89	-1.24
	Vidyo4	-7.76	-1.72
	720p_AVG	-7.68	-1.62
1080p	ParkScene	-3.00	-0.68
	Kimono	-3.82	-2.49
	Basketball	-4.78	-2.54
	Cactus	-3.22	-1.07
	BQSquare	-4.74	-0.86
	1080p_AVG	-3.91	-1.53
	AVG	-3.64	-1.03

[0135]

하나 이상의 실시예들에서, 본 개시에 설명된 기능들 또는 동작들은 하드웨어, 소프트웨어, 펌웨어, 또는 그것들의 임의의 조합으로 구현될 수도 있다. 소프트웨어로 구현된다면, 그 기능들 또는 동작들은 하나 이상의 명령들 또는 코드를 통해 컴퓨터 판독가능 매체 상에 저장되거나 전송될 수도 있고, 하드웨어-기반의 프로세싱 유닛에 의해 실행될 수도 있다. 컴퓨터 판독가능 매체는 데이터 저장 매체와 같은 유형의 매체에 대응하는 컴퓨터 판독가능 저장 매체, 또는 예컨대 통신 프로토콜에 따라 한 장소에서 다른 장소로의 컴퓨터 프로그램의 전송을 용이하게 하는 임의의 매체를 포함하는 통신 매체를 포함할 수도 있다. 이러한 방식으로, 컴퓨터 판독가능 매체는 일반적으로 (1) 비-일시적인 유형의 컴퓨터 판독가능 저장 매체 또는 (2) 신호 또는 반송파와 같은 통신 매체에 대응할 수도 있다. 데이터 저장 매체는 본 개시에 설명된 기술들의 구현을 위해 명령들, 코드 및/또는 데이터 구조들을 검색하기 위해 하나 이상의 컴퓨터들 또는 하나 이상의 프로세서들에 의해 액세스될 수 있는 임의의 사용가능한 매체일 수도 있다. 컴퓨터 프로그램 제품은 컴퓨터 판독가능 매체를 포함할 수도 있다.

[0136]

제한이 아닌 예로서, 이러한 컴퓨터 판독가능 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광 디스크 스토리지, 자기 디스크 스토리지, 또는 다른 자기 저장 디바이스들, 또는 소망의 프로그램 코드를 컴퓨터에 의해 액세스될 수도 있는 명령들 또는 데이터 구조들의 형태로 운반하거나 저장하는데 사용될 수도 있는 임의의 다른 매체와 같은 비-일시적인 매체를 포함할 수도 있다. 또한, 어떤 관련된 것이라도 적절히 컴퓨터 판독가능 매체라고 지칭된다. 예를 들어, 명령들이 웹사이트, 서버, 또는 다른 원격 자원으로부터, 동축 케이블, 광섬유 케이블, 꼬임 쌍 (twisted pair), 디지털가입자 회선 (DSL), 또는 무선 기술들 이를테면 적외선, 라디오, 및/또는 마이크로파를 이용하여 전송된다면, 동축 케이블, 광섬유 케이블, 연선, DSL, 또는 적외선, 라디오, 및 마이크로파와 같은 무선 기술들은 매체의 정의에 포함된다. 그러나, 컴퓨터 판독가능 저장 매체 및 데이터 저장 매체는 접속들, 반송파들, 신호들 또는 다른 일시적인 매체를 포함하는 것이 아니라, 그 대신 비-일시적인 유형의 저장 매체와 관련되는 것이 이해될 것이다. 디스크 (Disk 및 disc)는 여기서 사용되는 바와 같이, 콤팩트 디스크 (CD), 레이저 디스크, 광 디스크, 디지털 다용도 디스크 (DVD), 플로피 디스크 및 블루레이 디스크를 포함하는데, 디스크 (disk) 들은 보통 데이터를 자기적으로 재생하지만, 디스크 (disc) 들은 레이저들로 광학적

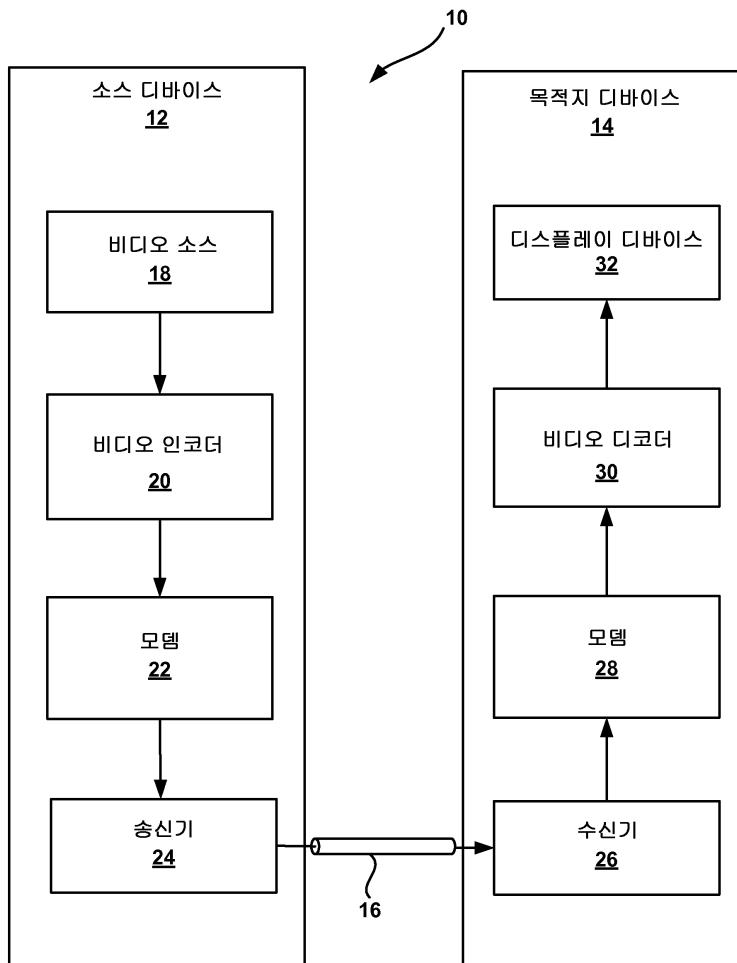
으로 데이터를 재생한다. 상기한 것들의 조합들도 컴퓨터 판독가능 매체의 범위 내에 포함되어야 한다.

[0137] 명령들은 하나 이상의 프로세서들, 예컨대 하나 이상의 디지털 신호 프로세서들 (DSPs), 범용 마이크로프로세서들, 애플리케이션용 집적 회로들 (ASICs), 필드 프로그래머블 로직 어레이들 (FPGAs) 또는 다른 등가의 집적 또는 이산 로직 회로에 의해 실행될 수도 있다. 따라서, 용어 "프로세서"는 본 명세서에서 이용되는 것과 같이, 전술된 구조 중 임의의 것 또는 본 명세서에 설명된 기술들의 구현에 적합한 임의의 다른 구조를 지칭할 수도 있다. 추가로, 일부 양태들에서, 본 명세서에 설명된 기능은 인코딩 및 디코딩을 위해 구성되거나 결합된 코덱에 통합된 전용 하드웨어 및/또는 소프트웨어 모듈들 내에 제공될 수도 있다. 또한, 기술들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수 있다.

[0138] 본 개시물의 기술들은 무선 핸드셋, 집적 회로 (IC) 또는 IC 세트 (예컨대, 칩 세트)를 포함하는 광범위한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들, 모듈들, 또는 유닛들은 본 개시물에서 개시된 기술들을 수행하도록 구성된 디바이스들의 기능적인 양태들을 강조하기 위해 설명되지만, 상이한 하드웨어 유닛들에 의한 구현을 반드시 필요로 하지는 않는다. 오히려, 전술된 것과 같이, 다양한 유닛들은 코덱 하드웨어 유닛에서 결합되거나 상호동작하는 하드웨어 유닛들의 집합에 의해 제공되며, 전술된 것과 같은 하나 이상의 프로세스들과 결합하여 적절한 소프트웨어 및/또는 펌웨어를 포함한다.

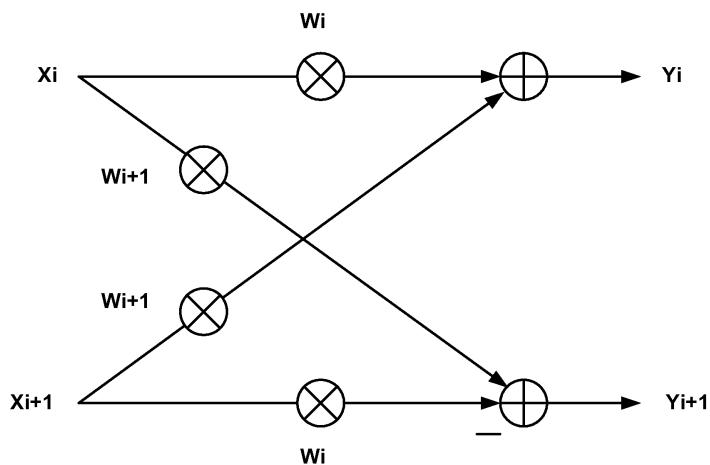
도면

도면1



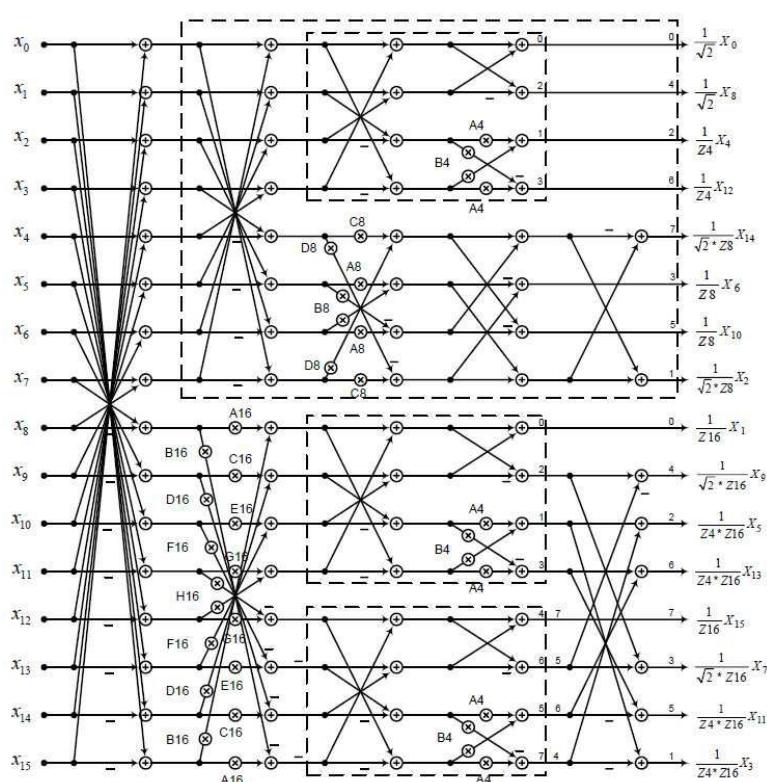
도면2

34

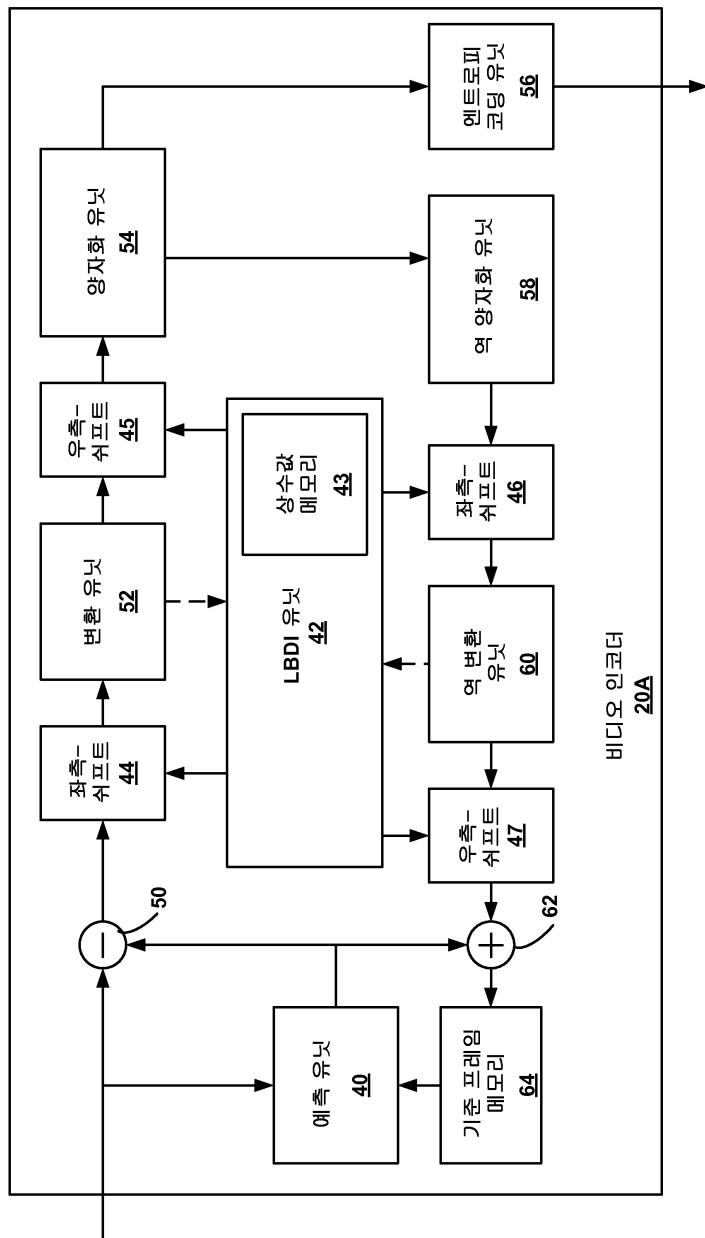


도면3

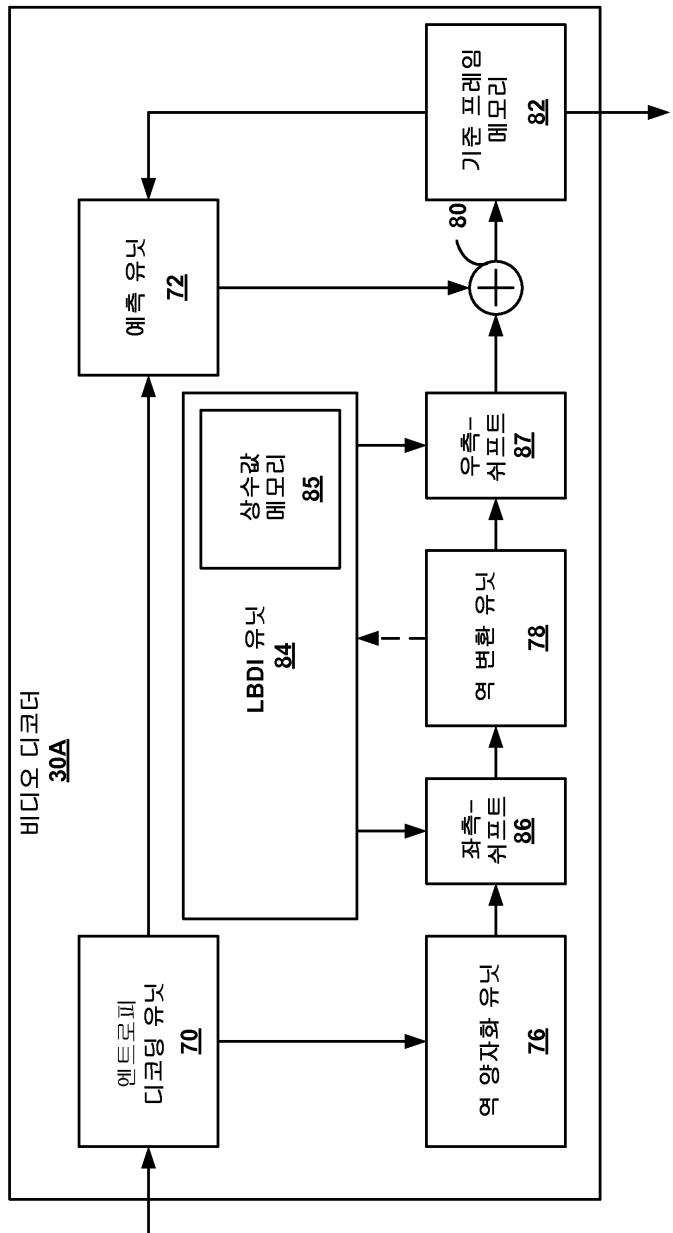
36



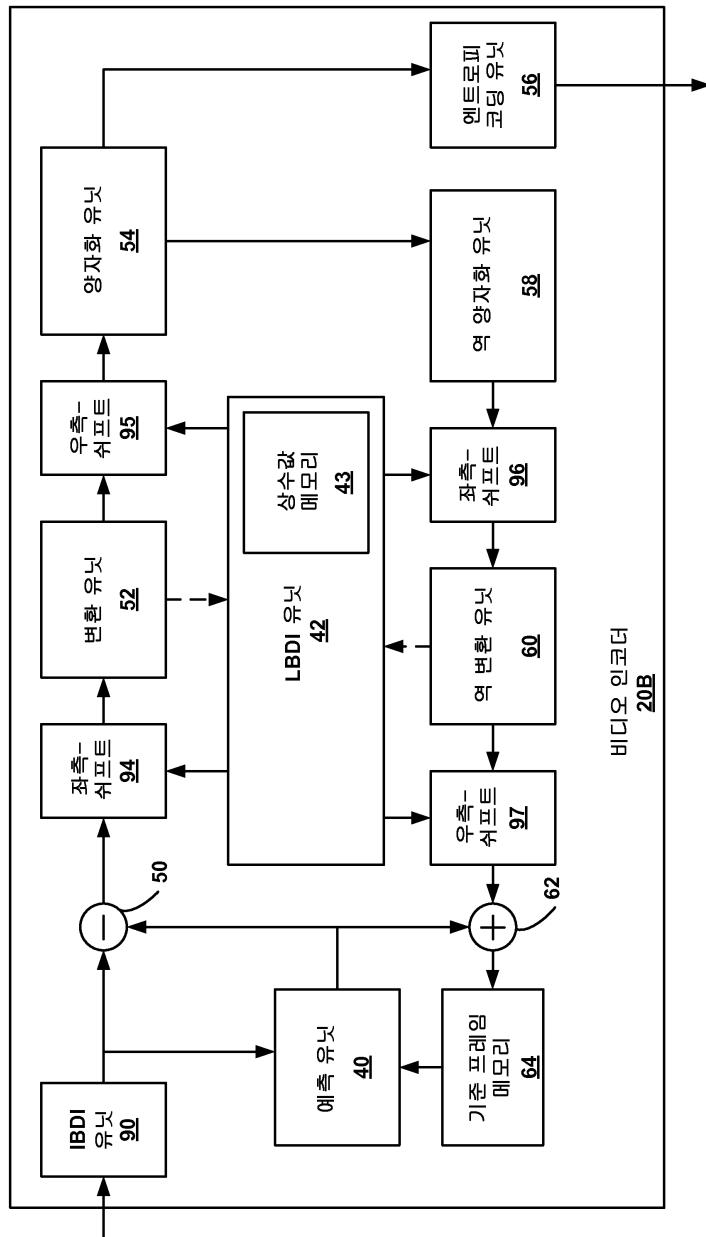
도면4



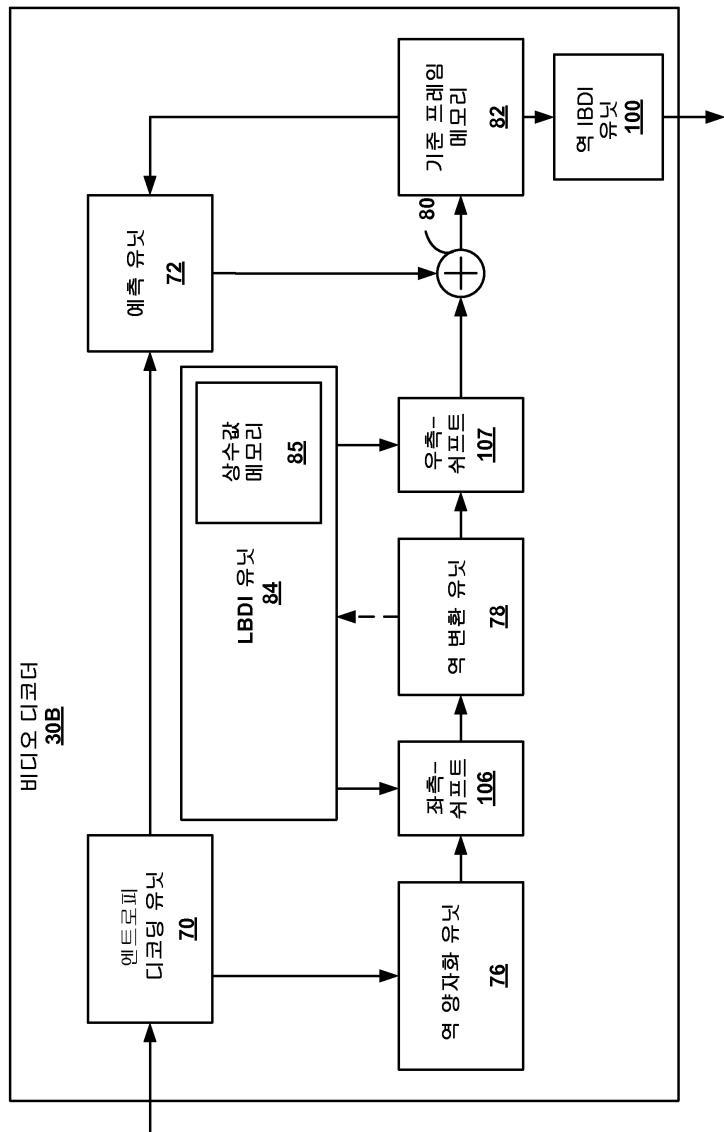
도면5



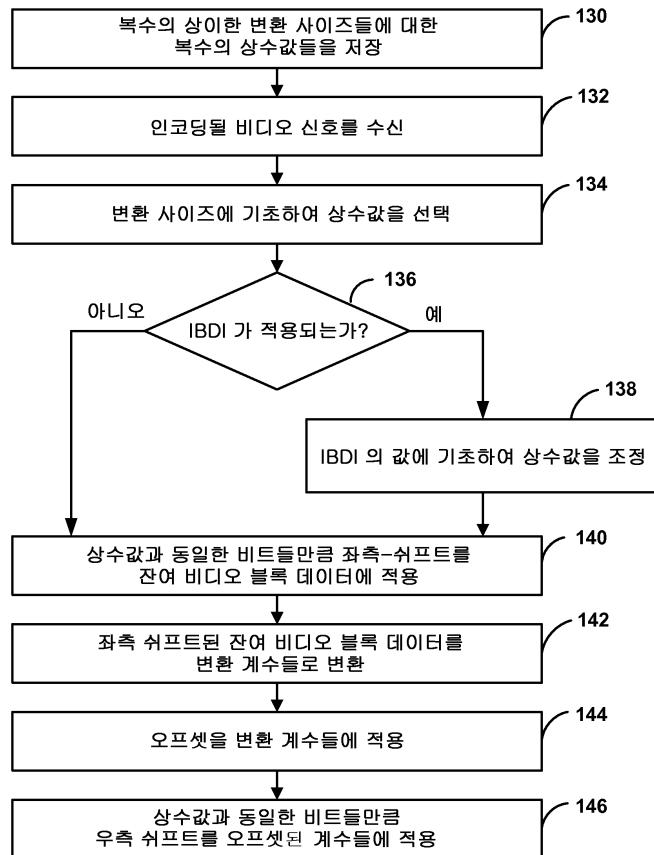
도면6



도면7



도면8



도면9

