



(52) CPC특허분류

*H04N 19/124* (2015.01)

*H04N 19/174* (2015.01)

*H04N 19/176* (2015.01)

(56) 선행기술조사문헌

US20090116555 A1

KR1019990022849 A

KR1020110105875 A

VESA Display Stream Compression (DSC) Standard  
Version 1.1, Aug. 1, 2014. pp.1-125

---

## 명세서

### 청구범위

#### 청구항 1

디스플레이 링크 비디오 압축의 방법으로서,

비디오 데이터의 슬라이스 내의 블록을 코딩하는 단계로서, 상기 슬라이스는 상기 비디오 데이터의 하나 이상의 블록들을 포함하는, 상기 블록을 코딩하는 단계;

상기 블록에 대응하는 비트들을 버퍼에 저장하는 단계;

상기 슬라이스 내의 상기 블록의 위치를 나타내는 정보를 결정하는 단계;

상기 슬라이스 내의 상기 블록의 위치에 적어도 부분적으로 기초하여 상기 버퍼의 충만도를 결정하는 단계; 및

상기 슬라이스 내의 임계 위치를 결정하는 단계로서, 상기 임계 위치에 도달하기 전에 제 1 위치에서 코딩된 각각의 블록은 제 1 값과 동일한 임시 최대 버퍼 크기를 갖고, 상기 임계 위치에 도달한 후에 제 2 위치에서 코딩된 각각의 블록은 조절가능한 값과 동일한 임시 최대 버퍼 크기를 가지며, 상기 조절가능한 값은 상기 제 1 값보다 작은, 상기 슬라이스 내의 임계 위치를 결정하는 단계를 포함하는, 디스플레이 링크 비디오 압축의 방법.

#### 청구항 2

제 1 항에 있어서,

상기 버퍼에 저장된 비트 수를 결정하는 단계를 더 포함하고, 상기 버퍼의 충만도를 결정하는 단계는 또한 상기 버퍼에 저장된 결정된 상기 비트 수에 적어도 부분적으로 기초하는, 디스플레이 링크 비디오 압축의 방법.

#### 청구항 3

제 2 항에 있어서,

상기 임시 최대 버퍼 크기는 상기 블록의 위치에 대해 상기 버퍼에 저장될 수 있는 원하는 최대 비트 수를 나타내고, 그리고 상기 버퍼의 충만도를 결정하는 단계는 상기 버퍼에 저장된 상기 비트 수를 상기 임시 최대 버퍼 크기로 나누는 단계를 포함하는, 디스플레이 링크 비디오 압축의 방법.

#### 청구항 4

제 1 항에 있어서,

상기 버퍼의 최대 용량 및 상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 원하는 최대 비트 수를 결정하는 단계를 더 포함하고, 상기 버퍼의 충만도를 결정하는 단계는 또한 상기 버퍼의 상기 최대 용량 및 상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 원하는 최대 비트 수에 적어도 부분적으로 기초하는, 디스플레이 링크 비디오 압축의 방법.

#### 청구항 5

제 4 항에 있어서,

상기 제 1 값은 상기 버퍼의 상기 최대 용량과 동일한, 디스플레이 링크 비디오 압축의 방법.

#### 청구항 6

제 5 항에 있어서,

상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 상기 원하는 최대 비트 수에 기초하여, 상기 임계 위치에 도달한 후에 상기 제 2 위치에서 코딩된 각각의 블록에 대한 상기 조절가능한 값을 감소시키는 단계를 더 포함하는, 디스플레이 링크 비디오 압축의 방법.

**청구항 7**

제 1 항에 있어서,

상기 슬라이스 내의 상기 블록의 위치를 나타내는 상기 정보를 결정하는 단계는 코딩된 상기 슬라이스에서의 블록들의 수를 결정하는 단계를 포함하는, 디스플레이 링크 비디오 압축의 방법.

**청구항 8**

디스플레이 링크 비디오 압축을 위한 장치로서,

비디오 데이터의 슬라이스 내의 블록을 코딩하도록 구성된 프로세서로서, 상기 슬라이스는 상기 비디오 데이터의 하나 이상의 블록들을 포함하는, 상기 프로세서; 및

상기 프로세서에 커플링되고 상기 블록에 대응하는 비트들을 저장하도록 구성된 버퍼를 포함하고,

상기 프로세서는 또한,

상기 슬라이스 내의 상기 블록의 위치를 나타내는 정보를 결정하고 상기 슬라이스 내의 상기 블록의 위치에 적어도 부분적으로 기초하여 상기 버퍼의 충만도를 결정하고, 그리고

상기 슬라이스 내의 임계 위치를 결정하는 것으로서, 상기 임계 위치에 도달하기 전에 제 1 위치에서 코딩된 각각의 블록은 제 1 값과 동일한 임시 최대 버퍼 크기를 갖고, 상기 임계 위치에 도달한 후에 제 2 위치에서 코딩된 각각의 블록은 조절가능한 값과 동일한 임시 최대 버퍼 크기를 가지며, 상기 조절가능한 값은 상기 제 1 값보다 작은, 상기 슬라이스 내의 임계 위치를 결정하도록 구성되는, 디스플레이 링크 비디오 압축을 위한 장치.

**청구항 9**

제 8 항에 있어서,

상기 프로세서는 또한 상기 버퍼에 저장된 비트 수를 결정하도록 구성되고, 상기 버퍼의 충만도를 결정하는 것은 또한 상기 버퍼에 저장된 결정된 상기 비트 수에 적어도 부분적으로 기초하는, 디스플레이 링크 비디오 압축을 위한 장치.

**청구항 10**

제 9 항에 있어서,

상기 임시 최대 버퍼 크기는 상기 블록의 위치에 대해 상기 버퍼에 저장될 수 있는 원하는 최대 비트 수를 나타내고, 그리고 상기 버퍼의 충만도를 결정하는 것은 상기 버퍼에 저장된 상기 비트 수를 상기 임시 최대 버퍼 크기로 나누는 것을 포함하는, 디스플레이 링크 비디오 압축을 위한 장치.

**청구항 11**

제 8 항에 있어서,

상기 프로세서는 또한 상기 버퍼의 최대 용량 및 상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 원하는 최대 비트 수를 결정하도록 구성되고, 상기 버퍼의 충만도를 결정하는 것은 또한 상기 버퍼의 상기 최대 용량 및 상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 원하는 최대 비트 수에 적어도 부분적으로 기초하는, 디스플레이 링크 비디오 압축을 위한 장치.

**청구항 12**

제 11 항에 있어서,

상기 제 1 값은 상기 버퍼의 상기 최대 용량과 동일한, 디스플레이 링크 비디오 압축을 위한 장치.

**청구항 13**

제 12 항에 있어서,

상기 프로세서는 또한, 상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 상기 원하는 최대 비트 수에

기초하여, 상기 임계 위치에 도달한 후에 상기 제 2 위치에서 코딩된 각각의 블록에 대한 상기 조절가능한 값을 감소시키도록 구성되는, 디스플레이 링크 비디오 압축을 위한 장치.

**청구항 14**

제 13 항에 있어서,

상기 슬라이스 내의 상기 블록의 위치를 나타내는 상기 정보를 결정하는 것은 코딩된 상기 슬라이스에서의 블록들의 수를 결정하는 것을 포함하는, 디스플레이 링크 비디오 압축을 위한 장치.

**청구항 15**

명령들을 저장하는 비밀시적 컴퓨터 판독가능 저장 매체로서,

상기 명령들이 실행될 때 디바이스의 프로세서로 하여금,

비디오 데이터의 슬라이스 내의 블록을 코딩하게 하는 것으로서, 상기 슬라이스는 상기 비디오 데이터의 하나 이상의 블록들을 포함하는, 상기 블록을 코딩하게 하고;

상기 블록에 대응하는 비트들을 버퍼에 저장하게 하고;

상기 슬라이스 내의 상기 블록의 위치를 나타내는 정보를 결정하게 하고;

상기 슬라이스 내의 상기 블록의 위치에 적어도 부분적으로 기초하여 상기 버퍼의 충만도를 결정하게 하고; 그리고

상기 슬라이스 내의 임계 위치를 결정하게 하는 것으로서, 상기 임계 위치에 도달하기 전에 제 1 위치에서 코딩된 각각의 블록은 제 1 값과 동일한 임시 최대 버퍼 크기를 갖고, 상기 임계 위치에 도달한 후에 제 2 위치에서 코딩된 각각의 블록은 조절가능한 값과 동일한 임시 최대 버퍼 크기를 가지며, 상기 조절가능한 값은 상기 제 1 값보다 작은, 상기 슬라이스 내의 임계 위치를 결정하게 하는, 비밀시적 컴퓨터 판독가능 저장 매체.

**청구항 16**

제 15 항에 있어서,

상기 명령들이 실행될 때 상기 프로세서로 하여금 상기 버퍼에 저장된 비트 수를 결정하게 하는 명령들을 또한 저장하고, 상기 버퍼의 충만도를 결정하는 것은 또한 상기 버퍼에 저장된 결정된 상기 비트 수에 적어도 부분적으로 기초하는, 비밀시적 컴퓨터 판독가능 저장 매체.

**청구항 17**

제 16 항에 있어서,

상기 임시 최대 버퍼 크기는 상기 블록의 위치에 대해 상기 버퍼에 저장될 수 있는 원하는 최대 비트 수를 나타내고, 그리고 상기 버퍼의 충만도를 결정하는 것은 상기 버퍼에 저장된 상기 비트 수를 상기 임시 최대 버퍼 크기로 나누는 것을 포함하는, 비밀시적 컴퓨터 판독가능 저장 매체.

**청구항 18**

제 15 항에 있어서,

상기 명령들이 실행될 때 상기 프로세서로 하여금 상기 버퍼의 최대 용량 및 상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 원하는 최대 비트 수를 결정하게 하는 명령들을 또한 저장하고, 상기 버퍼의 충만도를 결정하는 것은 또한 상기 버퍼의 상기 최대 용량 및 상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 원하는 최대 비트 수에 적어도 부분적으로 기초하는, 비밀시적 컴퓨터 판독가능 저장 매체.

**청구항 19**

제 18 항에 있어서,

상기 제 1 값은 상기 버퍼의 상기 최대 용량과 동일한, 비밀시적 컴퓨터 판독가능 저장 매체.

**청구항 20**

제 19 항에 있어서,

상기 명령들이 실행될 때 상기 프로세서로 하여금, 상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 상기 원하는 최대 비트 수에 기초하여, 상기 임계 위치에 도달한 후에 상기 제 2 위치에서 코딩된 각각의 블록에 대한 상기 조절가능한 값을 감소시키게 하는 명령들을 또한 저장하는, 비밀시적 컴퓨터 판독가능 저장 매체.

**청구항 21**

비디오 코딩 디바이스로서,

비디오 데이터의 슬라이스 내의 블록을 코딩하는 수단으로서, 상기 슬라이스는 상기 비디오 데이터의 하나 이상의 블록들을 포함하는, 상기 블록을 코딩하는 수단;

상기 블록에 대응하는 비트들을 버퍼에 저장하는 수단;

상기 슬라이스 내의 상기 블록의 위치를 나타내는 정보를 결정하는 수단;

상기 슬라이스 내의 상기 블록의 위치에 적어도 부분적으로 기초하여 상기 버퍼의 충만도를 결정하는 수단; 및

상기 슬라이스 내의 임계 위치를 결정하는 수단으로서, 상기 임계 위치에 도달하기 전에 제 1 위치에서 코딩된 각각의 블록은 제 1 값과 동일한 임시 최대 버퍼 크기를 갖고, 상기 임계 위치에 도달한 후에 제 2 위치에서 코딩된 각각의 블록은 조절가능한 값과 동일한 임시 최대 버퍼 크기를 가지며, 상기 조절가능한 값은 상기 제 1 값보다 작은, 상기 슬라이스 내의 임계 위치를 결정하는 수단을 포함하는, 비디오 코딩 디바이스.

**청구항 22**

제 21 항에 있어서,

상기 버퍼에 저장된 비트 수를 결정하는 수단을 더 포함하고, 상기 버퍼의 충만도를 결정하는 것은 또한 상기 버퍼에 저장된 결정된 상기 비트 수에 적어도 부분적으로 기초하는, 비디오 코딩 디바이스.

**청구항 23**

제 22 항에 있어서,

상기 임시 최대 버퍼 크기는 상기 블록의 위치에 대해 상기 버퍼에 저장될 수 있는 원하는 최대 비트 수를 나타내고, 그리고 상기 버퍼의 충만도를 결정하는 것은 상기 버퍼에 저장된 상기 비트 수를 상기 임시 최대 버퍼 크기로 나누는 것을 포함하는, 비디오 코딩 디바이스.

**청구항 24**

제 21 항에 있어서,

상기 버퍼의 최대 용량 및 상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 원하는 최대 비트 수를 결정하는 수단을 더 포함하고, 상기 버퍼의 충만도를 결정하는 것은 또한 상기 버퍼의 상기 최대 용량 및 상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 원하는 최대 비트 수에 적어도 부분적으로 기초하는, 비디오 코딩 디바이스.

**청구항 25**

제 24 항에 있어서,

상기 제 1 값은 상기 버퍼의 상기 최대 용량과 동일한, 비디오 코딩 디바이스.

**청구항 26**

제 25 항에 있어서,

상기 슬라이스가 완전히 코딩된 후에 상기 버퍼에 저장될 상기 원하는 최대 비트 수에 기초하여 상기 임계 위치에 도달한 후에 상기 제 2 위치에서 코딩된 각각의 블록에 대해 조절가능한 값을 감소시키는 수단을 더 포함하

는, 비디오 코딩 디바이스.

**청구항 27**

삭제

**청구항 28**

삭제

**청구항 29**

삭제

**청구항 30**

삭제

**발명의 설명**

**기술 분야**

[0001] 이 개시물은 비디오 코딩 및 압축, 특히, 디스플레이 링크 비디오 압축과 같이, 디스플레이 링크들 상에서의 송신을 위한 비디오 압축에 관한 것이다.

**배경 기술**

[0002] 디지털 비디오 기능들은, 디지털 텔레비전들, 개인 정보 단말 (personal digital assistant; PDA) 들, 랩톱 컴퓨터들, 데스크톱 모니터들, 디지털 카메라들, 디지털 레코딩 디바이스들, 디지털 미디어 플레이어들, 비디오 게임용 디바이스들, 비디오 게임 콘솔들, 셀룰러 또는 위성 라디오 전화들, 화상 원격회의 디바이스들 등등을 포함하는 광범위한 디스플레이들 내로 편입될 수 있다. 디스플레이 링크들은 디스플레이들을 적합한 소스 디바이스들에 접속하기 위하여 이용된다. 디스플레이 링크들의 대역폭 조건들은 디스플레이들의 해상도에 비례하고, 이에 따라, 고해상도 디스플레이들은 큰 대역폭 디스플레이 링크들을 요구한다. 일부 디스플레이 링크들은 고해상도 디스플레이들을 지원하기 위한 대역폭을 가지지 않는다. 비디오 압축은 더 낮은 대역폭의 디스플레이 링크들이 디지털 비디오를 고해상도 디스플레이들에 제공하기 위하여 이용될 수 있도록, 대역폭 조건들을 감소시키기 위하여 이용될 수 있다.

[0003] 다른 사람들은 픽셀 데이터에 관한 이미지 압축을 사용하도록 노력하였다. 그러나, 이러한 방식들은 때때로 시각적으로 무손실이 아니거나, 기존의 디스플레이 디바이스들에서 구현하기가 어려울 수 있고 고가일 수 있다.

[0004] 비디오 전자 표준 협회 (Video Electronics Standards Association; VESA) 는 디스플레이 스트림 압축 (Display Stream Compression; DSC) 을 디스플레이 링크 비디오 압축을 위한 표준으로서 개발하였다. DSC 와 같은 디스플레이 링크 비디오 압축 기법은 무엇보다도, 시각적으로 무손실인 픽처 품질 (즉, 사용자들이 압축이 활성인 것으로 말할 수 없도록 하는 품질의 레벨을 가지는 픽처들) 을 제공해야 한다. 디스플레이 링크 비디오 압축 기법은 또한, 기존의 하드웨어로 실시간으로 구현하기가 용이하고 덜 고가인 방식을 제공해야 한다.

**발명의 내용**

**해결하려는 과제**

**과제의 해결 수단**

[0005] 이 개시물의 시스템들, 방법들, 및 디바이스들은 각각 몇몇 혁신적인 양태들을 가지며, 그 단 하나가 본원에서 개시된 바람직한 속성들을 전적으로 담당하지는 않는다.

[0006] 일 양태에서, 비디오 데이터의 슬라이스 내의 블록을 코딩하는 단계로서, 슬라이스가 비디오 데이터의 하나 이상의 블록들을 포함하는, 상기 코딩하는 단계, 블록에 대응하는 비트들을 버퍼에 저장하는 단계, 슬라이스 내의

블록의 위치를 나타내는 정보를 결정하는 단계, 및 슬라이스 내의 블록의 위치에 적어도 부분적으로 기초하여 버퍼의 충만도를 결정하는 단계를 포함하는, 디스플레이 링크 비디오 압축의 방법이 제공된다.

[0007] 다른 양태에서, 디스플레이 링크 비디오 압축을 위한 장치는, 비디오 데이터의 슬라이스의 블록을 코딩하도록 구성된 프로세서로서, 슬라이스가 상기 비디오 데이터의 하나 이상의 블록들을 포함하는, 상기 코딩하도록 구성된 프로세서, 블록에 대응하는 비트들을 버퍼에 저장하도록 구성된 버퍼, 및 비디오 코딩 디바이스 및 버퍼에 작동되도록 커플링된 프로세서를 포함한다. 프로세서는 또한 슬라이스 내의 블록의 위치를 나타내는 정보를 결정하고 슬라이스 내의 블록의 위치에 적어도 부분적으로 기초하여 버퍼의 충만도를 결정하도록 구성된다.

[0008] 또 다른 양태에서, 비일시적 컴퓨터 판독가능 저장 매체는 명령들을 포함하며, 명령들은 실행되는 경우 디바이스의 프로세서로 하여금, 비디오 데이터의 슬라이스 내의 블록을 코딩하게 하는 것으로서, 슬라이스가 비디오 데이터의 하나 이상의 블록들을 포함하는, 상기 코딩하게 하고, 블록에 대응하는 비트들을 버퍼에 저장하게 하고, 슬라이스 내의 블록의 위치를 나타내는 정보를 결정하게 하고, 그리고 슬라이스 내의 블록의 위치에 적어도 부분적으로 기초하여 버퍼의 충만도를 결정하게 한다.

[0009] 또 다른 양태에서, 비디오 데이터의 슬라이스 내의 블록을 코딩하는 수단으로서, 슬라이스가 비디오 데이터의 하나 이상의 블록들을 포함하는, 상기 코딩하는 수단, 블록에 대응하는 비트들을 버퍼에 저장하는 수단, 슬라이스 내의 블록의 위치를 나타내는 정보를 결정하는 수단, 및 슬라이스 내의 블록의 위치에 적어도 부분적으로 기초하여 버퍼의 충만도를 결정하는 수단을 포함하는, 비디오 코딩 디바이스가 제공된다.

**도면의 간단한 설명**

[0010] 도 1a 는 이 개시물에서 설명된 양태들에 따라 기법들을 사용할 수도 있는 일 예의 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.

도 1b 는 이 개시물에서 설명된 양태들에 따라 기법들을 수행할 수도 있는 또 다른 예의 비디오 인코딩 및 디코딩 시스템을 예시하는 블록도이다.

도 2a 는 이 개시물에서 설명된 양태들에 따라 기법들을 구현할 수도 있는 비디오 인코더의 예를 예시하는 블록도이다.

도 2b 는 이 개시물에서 설명된 양태들에 따라 기법들을 구현할 수도 있는 비디오 디코더의 예를 예시하는 블록도이다.

도 3 은 데이터의 슬라이스들을 인코딩하기 위한 예시적인 프로세스를 도시한 그래프이다.

도 4 는 슬라이스의 인코딩 동안 버퍼 충만도 (buffer fullness; BF) 를 조절하기 위한 예시적인 기법을 나타낸다.

도 5 는 버퍼에서 고정 비트 레이트 (constant bit rate; CBR) 출력을 제공하기 위한 예시적인 방법을 나타낸 플로우차트이다.

**발명을 실시하기 위한 구체적인 내용**

[0011] 일반적으로, 이 개시물은 디스플레이 링크 비디오 압축에서 이용되는 것들과 같은 비디오 압축 기법들을 개선하는 방법에 관한 것이다. 보다 구체적으로, 본 개시물은 디스플레이 링크 비디오 압축을 위한 고정 비트 레이트 (CBR) 출력 및/또는 버퍼 (예를 들어, 레이트 버퍼) 의 입력을 제공하는 시스템 및 방법에 관한 것이다.

[0012] 종래의 디스플레이 기술에서, 디스플레이 링크 비디오 압축의 일레인 비디오 전자 표준 협회 (VESA) 에 의한 디스플레이 스트림 압축 (DSC) v1.0 솔루션에 의해 제공되는 3:1 압축은, 특히 4K 해상도 (4K라고도 함) 와 같은 고해상도 디스플레이와 관련된 요구 사항에 대해 미래의 모바일 시장 요구 사항을 유도하는데 불충분하다. 따라서 미래의 요구를 충족시키기 위해, 예를 들어 4:1 압축 이상을 제공하는 차세대 DSC 표준에 통합하기 위한 방법론을 개발하는 것이 바람직하다.

[0013] 이러한 맥락에서, DSC 방법론의 성능을 결정함에 있어서 레이트 제어를 관리하는 것이 바람직하다. 레이트 제어의 하나의 목적은, 레이트 왜곡 성능을 최대화하면서 레이트에 대한 순간 및 평균 제약을 만족시키는, 양자화 파라미터 (quantization parameter; QP), 코딩 모드 등과 같은 코딩 파라미터들의 세트를 결정하는 것이다.

[0014] 비디오 데이터의 각 블록을 인코딩하는데 소비되는 비트 (즉, 인코딩하는데 이용되는 비트) 는 블록의 특성들

(예를 들어, 크기, 비트 수 등)에 기초하여 실질적으로 변환될 수 있다. 따라서, 버퍼는 출력 비트 스트림의 레이트 변화를 평활화하기 위해 레이트 제어 메커니즘의 일부로서 사용될 수 있다. CBR 버퍼 모델에서, 비트는 유선 물리적 링크를 통해 데이터를 전송하는 동안 일정한 레이트로 버퍼로부터 제거될 수 있다. 이러한 버퍼 모델에서, 인코더가 버퍼에서 제거된 비트들에 비해 너무 많은 비트들을 추가하면, 버퍼에서의 비트 수가 버퍼의 용량을 초과하여 오버플로우를 야기시킬 수 있다. 다른 한편, 인코더는 언더플로우를 방지하기 위해 충분한 레이트로 비트를 추가해야 한다. 디코더 측에서, 비트는 버퍼에 일정한 레이트로 추가되고, 디코더는 각 블록에 대한 가변 비트 수를 제거한다. 적절한 디코딩을 보장하기 위해, 버퍼는 압축된 비트스트림의 디코딩 동안 임의의 지점에서 언더플로우 또는 오버플로우하지 않아야 한다.

[0015] 변수 BufferCurrentSize는 버퍼 내에 현재 저장된 비트 수를 나타내고, 변수 BufferMaxSize는 버퍼의 크기(즉, 용량), 즉 버퍼에 저장될 수 있는 비트의 전체 최대 수를 나타내는 것으로 한다. 버퍼의 "충만도" (버퍼 충만도 (BF) 라고도 함)는 아래 식 1에서 나타낸 바와 같이 계산될 수 있다. BF는 특정 시점에서 비트 저장에 사용되는 버퍼 용량의 백분율을 나타낸다.

[0016] 
$$BF = ((BufferCurrentSize * 100) / BufferMaxSize) \text{ (식 1)}$$

[0017] 또는, 다음을 사용:

[0018] 
$$BF = (((BufferCurrentSize * 100) + (BufferMaxSize \gg 1)) / BufferMaxSize) \text{ (식 2)}$$

[0019] 하지만, 특정 실시형태에서는, 비디오 데이터의 슬라이스의 끝에서, 즉 슬라이스에서 모든 블록을 코딩한 후에 버퍼에 포함된(즉, 저장된) 비트의 수가 BufferMaxSize보다 작은 것이 바람직하다. 변수 maxBufferBitsAtSliceEnd가 슬라이스의 끝에 있는 버퍼에서 원하는 최대 비트 수를 나타내는 것으로 한다. 일례에서, 언더플로우 및/또는 오버플로우를 방지하기 위해, 슬라이스의 끝에서 버퍼의 비트 수가 BufferMaxSize보다 작을 수 있는 maxBufferBitsAtSliceEnd와 동일한 것이 최적일 수 있다. 그러한 실시형태에서, maxBufferBitsAtSliceEnd는 슬라이스의 끝에서 BF를 계산할 때 BufferMaxSize 대신에 사용될 것이고, 전술한 식 1 및/또는 2는 최적이지 않을 것이다.

[0020] 그러한 실시형태에서, 슬라이스 내의 주어진 블록의 코딩 위치에 따라(즉, 기초하여) BF를 조절하는 것이 바람직하다. 보다 정확하게, 슬라이스의 끝에서 BufferCurrentSize = maxBufferBitsAtSliceEnd인 경우 BF가 100%가 되는 방식으로, BF는 슬라이스에서의 일부 고정된 수의 블록(즉, 미리결정된 또는 구성가능한 수의 블록)을 코딩한 후 일정한 레이트로 선형적으로 감소한다.

[0021] 어떤 실시형태들은 DSC 표준의 맥락에서 본원에서 설명되지만, 당해 분야의 당업자는 본원에서 개시된 시스템들 및 방법들이 임의의 적당한 비디오 코딩 표준에 적용가능할 수도 있다는 것을 인식할 것이다. 예를 들어, 본원에서 개시된 실시형태들은 다음의 표준들 중의 하나 이상에 적용가능할 수도 있다: 국제 전기통신 연합(International Telecommunication Union; ITU) 전기통신 표준화 섹터(Telecommunication Standardization Sector) (ITU-T) H.261, 표준화를 위한 국제 기구(International Organization for Standardization; ISO) / 국제 전기기술 위원회(International Electrotechnical Commission; IEC) (ISO/IEC) 동화상 전문가 그룹-1(Moving Picture Experts Group-1; MPEG-1) 비주얼(Visual), ITU-T H.262 또는 ISO/IEC MPEG-2 비주얼, ITU-T H.263, ISO/IEC MPEG-4 비주얼, (또한, ISO/IEC MPEG-4 AVC 로서 알려진) ITU-T H.264, 및 이러한 표준들에 대한 임의의 확장들. 또한, 이 개시물에서 설명된 기법들은 미래에 개발된 표준들의 일부가 될 수도 있다. 다시 말해서, 이 개시물에서 설명된 기법들은 이전에 개발된 비디오 코딩 표준들, 현재 개발 중인 비디오 코딩 표준들, 및 당면할 비디오 코딩 표준들에 적용가능할 수도 있다.

[0022] **비디오 코딩 표준들**

[0023] 비디오 이미지, TV 이미지, 스틸 이미지(still image), 또는 비디오 레코더 또는 컴퓨터에 의해 생성된 이미지와 같은 디지털 이미지는 수평 및 수직 라인들로 배열된 픽셀들 또는 샘플들을 포함할 수도 있다. 단일 이미지에서의 픽셀들의 수는 전형적으로 수만 개이다. 각각의 픽셀은 전형적으로 휘도(luminance) 및 색차(chrominance) 정보를 포함한다. 압축이 없다면, 이미지 인코더로부터 이미지 디코더로 전달되어야 할 정보의 순수한 분량은 실시간 이미지 송신을 비실용적으로 할 것이다. 송신되어야 할 정보의 양을 감소시키기 위하여, JPEG, MPEG, 및 H.263 표준들과 같은 다수의 상이한 압축 방법들이 개발되었다.

[0024] 비디오 코딩 표준들은 ITU-T H.261, ISO/IEC MPEG-1 비주얼, ITU-T H.262 또는 ISO/IEC MPEG-2 비주얼, ITU-T H.263, ISO/IEC MPEG-4 비주얼, (또한, ISO/IEC MPEG-4 AVC 로서 알려진) ITU-T H.264, 및 이러한 표준들의 확

장들을 포함하는 HEVC 를 포함한다.

[0025] 게다가, 비디오 코딩 표준, 즉, DSC 는 VESA 에 의해 개발되었다. DSC 표준은 디스플레이 링크들을 통한 송신을 위하여 비디오를 압축할 수 있는 비디오 압축 표준이다. 디스플레이들의 해상도가 증가함에 따라, 디스플레이들을 구동하기 위하여 요구된 비디오 데이터의 대역폭이 대응하여 증가한다. 일부 디스플레이 링크들은 비디오 데이터의 전부를 이러한 해상도들을 위한 디스플레이로 송신하기 위한 대역폭을 가지지 않을 수도 있다. 따라서, DSC 표준은 디스플레이 링크들 상에서의 상호동작가능한, 시각적 무손실 압축을 위한 압축 표준을 특정한다.

[0026] DSC 표준은 H.264 및 HEVC 와 같은 다른 비디오 코딩 표준들과는 상이하다. DSC 는 인트라-프레임 (intra-frame) 압축을 포함하지만, 인터-프레임 (inter-frame) 압축을 포함하지 않으며, 이것은 시간적 정보가 비디오 데이터를 코딩할 시에 DSC 표준에 의해 이용되지 않을 수도 있다는 것을 의미한다. 대조적으로, 다른 비디오 코딩 표준들은 그 비디오 코딩 기법들에서 인터-프레임 압축을 채용할 수도 있다.

[0027] **비디오 코딩 시스템**

[0028] 신규한 시스템들, 장치들, 및 방법들의 다양한 양태들은 동반된 도면들을 참조하여 이하에서 더욱 완전하게 설명된다. 그러나, 이 개시물은 많은 상이한 형태들로 구체화될 수도 있고, 이 개시물의 전반에 걸쳐 제시된 임의의 특정 구조 또는 기능으로 제한된 것으로서 해석되지 않아야 한다. 오히려, 이 양태들은, 이 개시물이 철저하고 완전할 것이며, 개시물의 범위를 당해 분야의 당업자들에게 완전히 전달하도록 제공된다. 본원에서의 교시사항들에 기초하여, 당해 분야의 당업자는 개시물의 범위가, 본 개시물의 임의의 다른 양태에 독립적으로 또는 이와 조합하여 구현되든지 간에, 본원에서 개시된 신규한 시스템들, 장치들, 및 방법들의 임의의 양태를 커버 (cover) 하도록 의도되는 것을 인식해야 한다. 예를 들어, 본원에서 기재된 임의의 수의 양태들을 이용하여 장치가 구현될 수도 있거나 방법이 실시될 수도 있다. 게다가, 본 개시물의 범위는 본원에서 기재된 본 개시물의 다양한 양태들에 추가하여, 또는 이 다양한 양태들 이외에, 다른 구조, 기능성, 또는 구조 및 기능성을 이용하여 실시되는 이러한 장치 또는 방법을 커버하도록 의도된다. 본원에서 개시된 임의의 양태는 청구항의 하나 이상의 구성요소들에 의해 구체화될 수도 있다는 것을 이해해야 한다.

[0029] 특정한 양태들이 본원에서 설명되지만, 이 양태들의 많은 변형들 및 치환들은 개시물의 범위 내에 속한다. 바람직한 양태들의 일부 이득들 및 장점들이 언급되지만, 개시물의 범위는 특정한 이득들, 용도들, 또는 목적들에 제한되도록 의도된 것이 아니다. 오히려, 개시물의 양태들은 상이한 무선 기술들, 시스템 구성들, 네트워크들, 및 송신 프로토콜들에 폭넓게 적용가능하도록 의도되며, 이들의 일부는 바람직한 양태들의 도면들 및 다음의 설명에서 예로서 예시되어 있다. 상세한 설명 및 도면들은 제한하는 것이 아니라 개시물의 예시에 불과하고, 개시물의 범위는 첨부된 청구항들 및 그 등가물들에 의해 정의된다.

[0030] 첨부된 도면들은 예들을 예시한다. 첨부된 도면들에서 참조 번호들에 의해 표시된 구성요소들은 다음의 설명에서 유사한 참조 번호들에 의해 표시된 구성요소들에 대응한다. 이 개시물에서, 서수 단어들 (예컨대, "제 1", "제 2", "제 3" 및 등등) 로 시작하는 명칭들을 가지는 구성요소들은 구성요소들이 특정한 순서를 가지는 것을 반드시 암시하지는 않는다. 오히려, 이러한 서수 단어들은 동일하거나 유사한 타입의 상이한 구성요소들을 지칭하기 위하여 이용되는 것에 불과하다.

[0031] 도 1a 는 이 개시물에서 설명된 양태들에 따라 기법들을 사용할 수도 있는 일 예의 비디오 코딩 시스템 (10) 을 예시하는 블록도이다. 본원에서 이용되고 설명된 바와 같이, 용어 "비디오 코더" 또는 "코더" 는 일반적으로 비디오 인코더들 및 비디오 디코더들의 양자를 지칭한다. 이 개시물에서, 용어들 "비디오 코딩" 또는 "코딩" 은 일반적으로 비디오 인코딩 및 비디오 디코딩을 지칭할 수도 있다. 비디오 인코더들 및 비디오 디코더들에 추가하여, 본 출원에서 설명된 양태들은 트랜스코더 (transcoder) 들 (예컨대, 비트스트림을 디코딩할 수 있고 또 다른 비트스트림을 재-인코딩 (re-encode) 할 수 있는 디바이스들) 및 미들박스 (middlebox) 들 (예컨대, 비트스트림을 수정할 수 있고, 변환할 수 있고, 및/또는 그렇지 않을 경우에 조작할 수 있는 디바이스들) 과 같은 다른 관련된 디바이스들로 확장될 수도 있다.

[0032] 도 1a 에서 도시된 바와 같이, 비디오 코딩 시스템 (10) 은 목적지 디바이스 (14) (즉, "비디오 코딩 디바이스 (14)" 또는 "코딩 디바이스 (14)") 에 의해 더 이후의 시간에 디코딩되어야 할 인코딩된 비디오 데이터를 생성하는 소스 디바이스 (12) (즉, "비디오 코딩 디바이스 (12)" 또는 "코딩 디바이스 (12)") 를 포함한다. 도 1a 의 예에서, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 별도의 디바이스들을 구성한다. 그러나, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 도 1b 의 예에서 도시된 바와 같이, 동일한 디바이스 상에 있을

수도 있거나 동일한 디바이스의 일부일 수도 있다는 것에 주목한다.

- [0033] 도 1a 를 다시 한번 참조하면, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 각각, 데스크톱 컴퓨터들, 노트북 (예컨대, 랩톱) 컴퓨터들, 태블릿 컴퓨터들, 셋톱 박스 (set-top box) 들, 소위 "스마트" 폰들과 같은 전화 핸드셋들, 소위 "스마트" 패드들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어 플레이어들, 비디오 게임용 콘솔들, 비디오 스트리밍 디바이스 등등을 포함하는 (또한 비디오 코딩 디바이스으로도 불리는) 광범위한 디바이스들 중의 임의의 것을 포함할 수도 있다. 다양한 실시형태들에서, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 무선 통신을 위해 구비 (즉 무선 통신을 통해 통신하도록 구성) 될 수도 있다.
- [0034] 비디오 코딩 시스템 (10) 의 비디오 코딩 디바이스들 (12, 14) 은 무선 네트워크 및 라디오 기술들, 예컨대 무선 광역 네트워크 (WWAN; wireless wide area network) (예를 들어, 셀룰러) 및 무선 근거리 네트워크 (WLAN; wireless local area network) 캐리어들을 통해 통신하도록 구성될 수 있다. 용어들 "네트워크" 및 "시스템" 은 종종 상호 교환가능하게 이용된다. 비디오 코딩 디바이스들 (12, 14) 의 각각은 사용자 장비 (UE), 무선 디바이스, 단말기, 모바일 스테이션, 가입자 유닛 등일 수 있다.
- [0035] WWAN 캐리어들은 예를 들어 코드 분할 다중 액세스 (CDMA), 시분할 다중 액세스 (TDMA), 주파수 분할 다중 액세스 (FDMA), 직교 FDMA (OFDMA), 단일 캐리어 FDMA (SC-FDMA) 및 다른 네트워크들과 같은 무선 통신 네트워크를 포함할 수 있다. CDMA 네트워크는 유니버설 지상 라디오 액세스 (Universal Terrestrial Radio Access; UTRA), CDMA 2000 등과 같은 라디오 기술 (radio technology) 을 구현할 수도 있다. UTRA 는 광대역 CDMA (Wideband CDMA; WCDMA), 및 CDMA 의 다른 변형들을 포함한다. CDMA2000 은 IS-2000, IS-95 및 IS-856 표준들을 커버한다. TDMA 네트워크는 이동 통신을 위한 글로벌 시스템 (Global System for Mobile Communications; GSM) 과 같은 라디오 기술을 구현할 수도 있다. OFDMA 네트워크는 진화형 UTRA (Evolved UTRA; E-UTRA), 울트라 이동 브로드밴드 (Ultra Mobile Broadband; UMB), IEEE 802.11 (Wi-Fi), IEEE 802.16 (WiMAX), IEEE 802.20, 플래시-OFDMA (Flash-OFDMA) 등과 같은 라디오 기술을 구현할 수도 있다. UTRA 및 E-UTRA 는 범용 이동 통신 시스템 (Universal Mobile Telecommunication System; UMTS) 의 일부이다. 3GPP 롱 텀 에볼루션 (LTE) 및 LTE-어드밴스는 E-UTRA 를 사용하는 UMTS 의 새로운 릴리즈이다. UTRA, E-UTRA, UMTS, LTE, LTE-A 및 GSM 은 "3 세대 파트너십 프로젝트 (3rd Generation Partnership Project)" (3GPP) 라는 명칭의 기구로부터의 문서들에서 설명되어 있다. CDMA2000 및 UMB 는 "3rd Generation Partnership Project 2" (3GPP2) 로 명명된 기구로부터의 문서들에서 설명된다.
- [0036] 비디오 코딩 시스템 (10) 의 비디오 코딩 디바이스들 (12, 14) 은 또한, 예를 들어, 이들 보정서들: 802.11a-1999 (통상 "802.11a"로 불림), 802.11b-1999 (통상 "802.11b"로 불림), 802.11g-2003 (통상 "802.11g"로 불림) 등을 포함하는, IEEE 802.11 표준과 같은 하나 이상의 표준들에 따라 WLAN 기지국을 통해 서로 통신할 수 있다.
- [0037] 목적지 디바이스 (14) 는 링크 (16) 를 통해, 디코딩되어야 할 인코딩된 비디오 데이터를 수신할 수도 있다. 링크 (16) 는 인코딩된 비디오 데이터를 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로 이동시킬 수 있는 임의의 타입의 매체 또는 디바이스를 포함할 수도 있다. 도 1a 의 예에서, 링크 (16) 는 소스 디바이스 (12) 가 인코딩된 비디오 데이터를 실시간으로 목적지 디바이스 (14) 로 송신하는 것을 가능하게 하기 위한 통신 매체를 포함할 수도 있다. 인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라 변조될 수도 있고, 목적지 디바이스 (14) 로 송신될 수도 있다. 통신 매체는 라디오 주파수 (radio frequency; RF) 스펙트럼 또는 하나 이상의 물리적 송신 라인들과 같은 임의의 무선 또는 유선 통신 매체를 포함할 수도 있다. 통신 매체는 로컬 영역 네트워크, 광역 네트워크, 또는 인터넷과 같은 글로벌 네트워크와 같은 패킷-기반 네트워크의 일부를 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스 (12) 로부터 목적지 디바이스 (14) 로의 통신을 용이하게 하기 위해 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.
- [0038] 도 1a 의 예에서, 소스 디바이스 (12) 는 비디오 소스 (18), 비디오 인코더 (20), 및 출력 인터페이스 (22) 를 포함한다. 일부의 경우들에는, 출력 인터페이스 (22) 가 변조기/복조기 (모뎀) 및/또는 송신기를 포함할 수도 있다. 소스 디바이스 (12) 에서, 비디오 소스 (18) 는 비디오 캡처 디바이스, 예컨대, 비디오 카메라, 이전에 캡처된 비디오를 포함하는 비디오 아카이브 (video archive), 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 공급 인터페이스, 및/또는 소스 비디오로서 컴퓨터 그래픽 데이터를 생성하기 위한 컴퓨터 그래픽 시스템과 같은 소스, 또는 이러한 소스들의 조합을 포함할 수도 있다. 하나의 예로서, 비디오 소스 (18) 가 비디오 카메라일 경우, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 도 1b 의 예에서 예시된 바와

같이, 소위 "카메라 폰들" 또는 "비디오 폰들" 을 형성할 수도 있다. 그러나, 이 개시물에서 설명된 기법들은 일반적으로 비디오 코딩에 적용가능할 수도 있고, 무선 및/또는 유선 애플리케이션들에 적용될 수도 있다.

[0039] 캡처된 (captured), 프리-캡처된 (pre-captured), 또는 컴퓨터-생성된 비디오는 비디오 인코더 (20) 에 의해 인코딩될 수도 있다. 인코딩된 비디오 데이터는 소스 디바이스 (12) 의 출력 인터페이스 (22) 를 통해 목적지 디바이스 (14) 로 송신될 수도 있다. 인코딩된 비디오 데이터는 또한 (또는 대안적으로), 디코딩 및/또는 재생을 위하여, 목적지 디바이스 (14) 또는 다른 디바이스들에 의한 더 이후의 액세스를 위해 저장 디바이스 (31) 상으로 저장될 수도 있다. 도 1a 및 도 1b 에서 예시된 비디오 인코더 (20) 는 도 2a 에서 예시된 비디오 인코더 (20), 또는 본원에서 설명된 임의의 다른 비디오 인코더를 포함할 수도 있다.

[0040] 도 1a 의 예에서, 목적지 디바이스 (14) 는 입력 인터페이스 (28), 비디오 디코더 (30), 및 디스플레이 디바이스 (32) 를 포함한다. 일부 경우에는, 입력 인터페이스 (28) 가 수신기 및/또는 모뎀을 포함할 수도 있다. 목적지 디바이스 (14) 의 입력 인터페이스 (28) 는 링크 (16) 를 통해, 및/또는 저장 디바이스 (31) 로부터 인코딩된 비디오 데이터를 수신할 수도 있다. 링크 (16) 를 통해 통신되거나 저장 디바이스 (31) 상에서 제공된 인코딩된 비디오 데이터는 비디오 데이터를 디코딩할 시에, 비디오 디코더 (30) 와 같은 비디오 디코더에 의한 이용을 위하여 비디오 인코더 (20) 에 의해 생성된 다양한 신택스 엘리먼트들을 포함할 수도 있다. 이러한 신택스 엘리먼트들은 통신 매체 상에서 송신되거나, 저장 매체 상에 저장되거나, 파일 서버에 저장된 인코딩된 비디오 데이터와 함께 포함될 수도 있다. 도 1a 및 도 1b 에서 예시된 비디오 디코더 (30) 는 도 2b 에서 예시된 비디오 디코더 (30), 또는 본원에서 설명된 임의의 다른 비디오 디코더를 포함할 수도 있다.

[0041] 디스플레이 디바이스 (32) 는 목적지 디바이스 (14) 와 통합될 수도 있거나, 목적지 디바이스 (14) 의 외부에 있을 수도 있다. 일부 예들에서, 목적지 디바이스 (14) 는 통합된 디스플레이 디바이스를 포함할 수도 있고, 또한, 외부의 디스플레이 디바이스와 인터페이스하도록 구성될 수도 있다. 다른 예들에서, 목적지 디바이스 (14) 는 디스플레이 디바이스일 수도 있다. 일반적으로, 디스플레이 디바이스 (32) 는 디코딩된 비디오 데이터를 사용자에게 디스플레이하고, 액정 디스플레이 (liquid crystal display; LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (organic light emitting diode; OLED) 디스플레이, 또는 또 다른 타입의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중의 임의의 것을 포함할 수도 있다.

[0042] 관련된 양태들에서, 도 1b 는 일 예의 비디오 코딩 시스템 (10') 을 도시하고, 여기서, 소스 디바이스 (12) 및 목적지 디바이스 (14) 는 디바이스 (11) 상에 있거나, 디바이스 (11) 의 일부이다. 디바이스 (11) 는 "스마트" 폰 등등과 같은 전화 핸드셋일 수도 있다. 디바이스 (11) 는 소스 디바이스 (12) 및 목적지 디바이스 (14) 와 동작가능하게 통신하는 (선택적으로 존재하는) 프로세서/제어기 디바이스 (13) 를 포함할 수도 있다. 도 1b 의 비디오 코딩 시스템 (10') 및 그 컴포넌트들은 그 외에는, 도 1a 의 비디오 코딩 시스템 (10) 및 그 컴포넌트들과 유사하다.

[0043] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 DSC 와 같은 비디오 압축 표준에 따라 동작할 수도 있다. 대안적으로, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 MPEG-4, Part 10, AVC, HEVC, 또는 이러한 표준들의 확장들로서 대안적으로 지칭된 ITU-T H.264 표준과 같은 다른 독점적 또는 산업 표준들에 따라 동작할 수도 있다. 그러나, 이 개시물의 기법들은 임의의 특정한 코딩 표준으로 제한되지는 않는다. 비디오 압축 표준들의 다른 예들은 MPEG-2 및 ITU-T H.263 을 포함한다.

[0044] 도 1a 및 도 1b 의 예들에서 도시되지 않았지만, 비디오 인코더 (20) 및 비디오 디코더 (30) 는 각각 오디오 인코더 및 디코더와 통합될 수도 있고, 공통의 데이터 스트림 또는 별도의 데이터 스트림들에서 오디오 및 비디오 양자의 인코딩을 처리하기 위한 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 사용자 데이터그램 프로토콜 (user datagram protocol; UDP) 과 같은 다른 프로토콜들을 준수할 수도 있다.

[0045] 비디오 인코더 (20) 및 비디오 디코더 (30) 는 각각, 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서 (digital signal processor; DSP) 들, 애플리케이션 특정 집적 회로 (application specific integrated circuit; ASIC) 들, 필드 프로그래밍가능 게이트 어레이 (field programmable gate array; FPGA) 들, 개별 로직, 소프트웨어, 하드웨어, 펌웨어, 또는 그 임의의 조합들과 같은 다양한 적당한 인코더 회로부 중의 임의의 것으로서 구현될 수도 있다. 기법들이 소프트웨어로 부분적으로 구현될 때, 디바이스는 소프트웨어를 위한 명령들을 적당한 비-일시적 컴퓨터-관독가능 매체 내에 저장할 수도 있고, 이 개시물의 기법들을 수행하기 위하여 하나 이상의 프로세서들을 이용하여 명령들을 하드웨어로 실행할 수도 있다. 비디오 인코더 (20) 및 비디오 디코더 (30) 의 각각은 하나 이상의 인코더들 또는 디코더들 내에 포함될 수도 있고, 그 어느 하나는 조합

된 인코더/디코더의 일부로서 개개의 디바이스 내에 통합될 수도 있다.

[0046] **비디오 코딩 프로세스**

[0047] 위에서 간단히 언급된 바와 같이, 비디오 인코더 (20) 는 비디오 데이터를 인코딩한다. 비디오 데이터는 하나 이상의 픽처들을 포함할 수도 있다. 픽처들의 각각은 비디오의 일부를 형성하는 스틸 이미지이다. 일부 사례들에서, 픽처는 비디오 "프레임" 으로서 지칭될 수도 있다. 비디오 인코더 (20) 가 비디오 데이터 (예를 들어, 비디오 코딩 계층 (VCL) 데이터 및/또는 비-VCL 데이터) 를 인코딩할 때, 비디오 인코더 (20) 는 비트스트림을 생성할 수도 있다. 비트스트림은 비디오 데이터의 코딩된 표현을 형성하는 비트들의 시퀀스를 포함할 수도 있다. 비트스트림은 코딩된 픽처들 및 연관된 데이터를 포함할 수도 있다. 코딩된 픽처는 픽처의 코딩된 표현이다. VCL 데이터는 코딩된 픽처 데이터 (즉, 코딩된 픽처(들)의 샘플들과 관련된 정보) 를 포함할 수 있고 비-VCL 데이터는 하나 이상의 코딩된 픽처(들)와 관련된 제어 정보 (예를 들어, 파라미터 세트들 및/또는 보충 강화 정보) 를 포함할 수 있다.

[0048] 비트스트림을 생성하기 위하여, 비디오 인코더 (20) 는 비디오 데이터에서의 각각의 픽처에 대해 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 픽처들에 대해 인코딩 동작들을 수행할 때, 비디오 인코더 (20) 는 일련의 코딩된 픽처들 및 연관된 데이터를 생성할 수도 있다. 연관된 데이터는 양자화 파라미터 (quantization parameter; QP) 와 같은 코딩 파라미터들의 세트를 포함할 수도 있다. 코딩된 픽처를 생성하기 위하여, 비디오 인코더 (20) 는 픽처를 동일한 크기의 비디오 블록들로 파티셔닝할 수도 있다. 비디오 블록은 샘플들의 2 차원 어레이일 수도 있다. 코딩 파라미터들은 비디오 데이터의 매 블록에 대한 코딩 옵션 (예컨대, 코딩 모드) 을 정의할 수도 있다. 코딩 옵션은 희망하는 레이트-왜곡 성능을 달성하기 위하여 선택될 수도 있다.

[0049] 일부 예들에서, 비디오 인코더 (20) 는 픽처를 복수의 슬라이스들로 파티셔닝할 수도 있다. 슬라이스들의 각각은 이미지 또는 프레임에서의 영역들의 나머지로부터의 정보 없이 독립적으로 디코딩될 수 있는 이미지 (예컨대, 프레임) 에서의 공간적으로 별개의 영역을 포함할 수도 있다. 각각의 이미지 또는 비디오 프레임은 단일 슬라이스에서 인코딩될 수도 있거나, 또는 각각의 이미지 또는 비디오 프레임은 몇몇 슬라이스들에서 인코딩될 수도 있다. DSC 에서, 각각의 슬라이스를 인코딩하기 위하여 할당된 비트 수는 실질적으로 일정할 수도 있다. 픽처에 대해 인코딩 동작을 수행하는 것의 일부로서, 비디오 인코더 (20) 는 픽처의 각각의 슬라이스에 대해 인코딩 동작들을 수행할 수도 있다. 비디오 인코더 (20) 가 슬라이스에 대해 인코딩 동작을 수행할 때, 비디오 인코더 (20) 는 슬라이스와 연관된 인코딩된 데이터를 생성할 수도 있다. 슬라이스와 연관된 인코딩된 데이터는 "코딩된 슬라이스" 로서 지칭될 수도 있다.

[0050] **DSC 비디오 인코더**

[0051] 도 2a 는 이 개시물에서 설명된 양태들에 따라 기법들을 구현할 수도 있는 비디오 인코더 (20) 의 예를 예시하는 블록도이다. 비디오 인코더 (20) 는 이 개시물의 기법들 중의 일부 또는 전부를 수행하도록 구성될 수도 있다. 일부 예들에서, 이 개시물에서 설명된 기법들은 비디오 인코더 (20) 의 다양한 컴포넌트들 사이에서 공유될 수도 있다. 일부 예들에서, 추가적으로 또는 대안적으로, 프로세서 (도시되지 않음) 는 이 개시물에서 설명된 기법들 중의 일부 또는 전부를 수행하도록 구성될 수도 있다.

[0052] 설명의 목적들을 위하여, 이 개시물은 DSC 코딩의 맥락에서 비디오 인코더 (20) 를 설명한다. 그러나, 이 개시물의 기법들은 다른 코딩 표준들 또는 방법들에 적용가능할 수도 있다.

[0053] 도 2a 의 예에서, 비디오 인코더 (20) 는 복수의 기능적 컴포넌트들을 포함한다. 비디오 인코더 (20) 의 기능적 컴포넌트들은 컬러-공간 변환기 (105), 버퍼 (110), 평탄성 검출기 (115), 레이트 제어기 (120), 예측기, 양자화기, 및 재구성기 컴포넌트 (125), 라인 버퍼 (130), 인덱싱된 컬러 히스토리 (indexed color history; 135), 엔트로피 인코더 (140), 서브스트림 멀티플렉서 (145), 및 레이트 버퍼 (150) 를 포함한다. 다른 예들에서, 비디오 인코더 (20) 는 더 많거나, 더 적거나, 또는 상이한 기능적 컴포넌트들을 포함할 수도 있다.

[0054] 컬러-공간 변환기 (105) 는 입력 컬러-공간을, 코딩 구현에서 이용된 컬러-공간으로 변환할 수도 있다. 예를 들어, 하나의 예시적인 실시형태에서, 입력 비디오 데이터의 컬러-공간은 적색, 녹색, 및 청색 (RGB) 컬러-공간에 있고, 코딩은 휘도 Y, 색차 녹색 Cg, 및 색차 황색 Co (YCbCr) 컬러-공간에서 구현된다. 컬러-공간 변환은 비디오 데이터에 대한 시프트 (shift) 들 및 추가들을 포함하는 방법 (들) 에 의해 수행될 수도 있다. 다른 컬러-공간들에서의 입력 비디오 데이터가 프로세싱될 수도 있고 다른 컬러-공간들로의 변환들이 또한 수행될 수도 있다는 것에 주목한다.

- [0055] 관련된 양태들에서, 비디오 인코더 (20) 는 버퍼 (110), 라인 버퍼 (130), 및/또는 레이트 버퍼 (150) 를 포함할 수도 있다. 예를 들어, 버퍼 (110) 는 비디오 인코더 (20) 의 다른 부분들에 의한 그 이용 이전에 컬러-공간 변환된 비디오 데이터를 유지할 수도 있다. 또 다른 예에서, 비디오 데이터는 RGB 컬러-공간에서 저장될 수도 있고, 컬러-공간 변환된 데이터는 더 많은 비트들을 요구할 수도 있으므로, 컬러-공간 변환은 필요한 대로 수행될 수도 있다.
- [0056] 레이트 버퍼 (150) 는, 레이트 제어기 (120) 와 관련하여 이하에서 더욱 상세하게 설명될 비디오 인코더 (20) 에서의 레이트 제어 메커니즘의 일부로서 기능할 수도 있다. 각각의 블록을 인코딩하는 것에 소비된 비트들은 블록의 본질에 기초하여 고도로 실질적으로 변동될 수 있다. 레이트 버퍼 (150) 는 압축된 비디오에서 레이트 변동들을 평활화 (smooth) 할 수 있다. 일부 실시형태들에서는, 비트들이 CBR에서 버퍼로부터 추출되는 CBR 버퍼 모델이 채용된다. CBR 버퍼 모델에서는, 비디오 인코더 (20) 가 너무 많은 비트들을 비트 스트림에 추가할 경우, 레이트 버퍼 (150) 가 오버플로우 (overflow) 할 수도 있다. 다른 한편으로, 비디오 인코더 (20) 는 레이트 버퍼 (150) 의 언더플로우 (underflow) 를 방지하기 위하여 충분한 비트들을 추가해야 한다.
- [0057] 비디오 디코더 측 상에서는, 비트들이 CBR 에서 비디오 디코더 (30) 의 레이트 버퍼 (155) (이하에서 더욱 상세하게 설명되는 도 2b 참조) 에 추가될 수도 있고, 비디오 디코더 (30) 는 각각의 블록에 대한 가변적인 수들의 비트들을 제거할 수도 있다. 적절한 디코딩을 보장하기 위하여, 비디오 디코더 (30) 의 레이트 버퍼 (155) 는 압축된 비트 스트림의 디코딩 동안에 "언더플로우" 또는 "오버플로우" 하지 않아야 한다.
- [0058] 상기에서 논의된 바와 같이, BF 는, 버퍼에서 현재 비트들의 수를 나타내는 BufferCurrentSize 와, 레이트 버퍼 (150) 의 크기, 즉, 임의의 시간 포인트에서 레이트 버퍼 (150) 내에 저장될 수 있는 비트들의 최대 수를 나타내는 BufferMaxSize 의 값들에 기초하여 정의될 수 있다. BF 는 상기 식 1 또는 식 2에 따라 계산될 수도 있지만, BF 를 계산하는 추가적인 방법이 후술된다.
- [0059] 평탄성 검출기 (115) 는 비디오 데이터에서의 복잡한 (즉, 비-평탄한 (non-flat)) 구역들로부터 비디오 데이터에서의 평탄한 (즉, 간단하거나 균일한) 구역들로의 변경들을 검출할 수 있다. 용어들 "복잡한" 및 "평탄한" 은 비디오 인코더 (20) 가 비디오 데이터의 개개의 영역들을 인코딩하기 위한 어려움을 일반적으로 지적하기 위하여 본원에서 이용될 것이다. 이에 따라, 본원에서 이용된 바와 같은 용어 복잡은 비디오 데이터의 영역을, 비디오 인코더 (20) 가 인코딩하기가 복잡한 것으로서 일반적으로 설명하고, 예를 들어, 텍스처링된 비디오 데이터 (textured video data), 높은 공간적 주파수, 및/또는 인코딩하기가 복잡한 다른 특징들을 포함할 수도 있다. 본원에서 이용된 바와 같은 용어 평탄은 비디오 데이터의 영역을, 비디오 인코더 (20) 가 인코딩하기가 간단한 것으로서 일반적으로 설명하고, 예를 들어, 비디오 데이터에서의 평활한 경도 (smooth gradient), 낮은 공간적 주파수, 및/또는 인코딩하기가 간단한 다른 특징들을 포함할 수도 있다. 복잡한 그리고 평탄한 영역들 사이의 전이 (transition) 들은 인코딩된 비디오 데이터에서 양자화 아티팩트 (quantization artifact) 들을 감소시키기 위하여 비디오 인코더 (20) 에 의해 이용될 수도 있다. 구체적으로, 레이트 제어기 (120) 및 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 복잡한 영역들로부터 평탄한 영역들로의 전이들이 식별될 때에 이러한 양자화 아티팩트들을 감소시킬 수 있다.
- [0060] 레이트 제어기 (120) 는 코딩 파라미터들, 예컨대, QP 의 세트를 결정한다. 레이트 버퍼 (150) 가 오버플로우하거나 언더플로우하지 않는다는 것을 보장하는 타겟 비트레이트에 대한 픽처 품질을 최대화하기 위하여, QP 는 레이트 버퍼 (150) 의 BF 및 비디오 데이터의 이미지 활성에 기초하여 레이트 제어기 (120) 에 의해 조절될 수도 있다. 레이트 제어기 (120) 는 또한, 최적의 레이트-왜곡 성능을 달성하기 위하여, 비디오 데이터의 각각의 블록에 대한 특정한 코딩 옵션 (예컨대, 특정한 모드) 을 선택한다. 레이트 제어기 (120) 는 왜곡이 비트-레이트 제약을 충족시키도록, 즉, 전체적인 실제의 코딩 레이트가 타겟 비트 레이트 내에서 적합하도록, 재구성된 이미지들의 왜곡을 최소화한다.
- [0061] 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 비디오 인코더 (20) 의 적어도 3 개의 인코딩 동작들을 수행할 수도 있다. 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 다수의 상이한 모드들에서 예측을 수행할 수도 있다. 하나의 예의 예측 모드는 중간-적응적 예측 (median-adaptive prediction) 의 수정된 버전이다. 중간-적응적 예측은 무손실 JPEG 표준 (lossless JPEG standard; JPEG-LS) 에 의해 구현될 수도 있다. 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 에 의해 수행될 수도 있는 중간-적응적 예측의 수정된 버전은 3 개의 연속 샘플 값들의 병렬 예측을 허용할 수도 있다. 또 다른 예의 예측 모드는 블록 예측이다. 블록 예측에서는, 샘플들이 상기 라인에서의 이전에 재구성된 픽셀들로부터 동일한 라인에서의 좌측까

지 예측된다. 일부 실시형태에서, 비디오 인코더 (20) 및 비디오 디코더 (30) 양자는 블록 예측 사용량들을 결정하기 위하여 재구성된 픽셀들에 대해 동일한 검색을 수행할 수 있고, 이에 따라, 어떤 비트들도 블록 예측 모드에서 전송될 필요가 없다. 다른 실시형태에서, 비디오 인코더 (20) 는, 비디오 디코더 (30) 가 별도의 검색을 수행할 필요가 없도록 비트스트림에서 검색 및 신호 블록 예측 벡터들을 수행할 수 있다. 샘플들이 컴포넌트 범위의 중점을 이용하여 예측되는 중점 예측 모드 (midpoint prediction mode) 가 또한 구현될 수도 있다. 중점 예측 모드는 심지어 최악-경우의 샘플에서의 압축된 비디오에 대하여 요구된 비트들의 수의 경계결정을 가능하게 할 수도 있다.

[0062] 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 또한 양자화를 수행한다. 예를 들어, 양자화는 시프터 (shifter) 를 이용하여 구현될 수도 있는 2-제곱 (power-of-2) 양자화기를 통해 수행될 수도 있다. 다른 양자화 기법들이 2-제곱 양자화기 대신에 구현될 수도 있다는 것에 주목한다. 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 에 의해 수행된 양자화는 레이트 제어기 (120) 에 의해 결정된 QP 에 기초할 수도 있다. 최종적으로, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 또한, 역 양자화된 잔차를 예측된 값에 추가하는 것과, 결과가 샘플 값들의 유효한 범위의 외부에 속하지 않는다는 것을 보장하는 것을 포함하는 재구성을 수행한다.

[0063] 예측기, 양자화기, 재구성기 컴포넌트 (125) 에 의해 수행된 예측, 양자화, 및 재구성에 대한 상기 설명된 예의 접근법들은 예시적인 것에 불과하고 다른 접근법들이 구현될 수도 있다는 것에 주목한다. 또한, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 는 예측, 양자화, 및/또는 재구성을 수행하기 위한 하위컴포넌트 (들) 를 포함할 수도 있다는 것에 주목한다. 또한, 예측, 양자화, 및/또는 재구성은 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 대신에 몇몇 별도의 인코더 컴포넌트들에 의해 수행될 수도 있다는 것에 주목한다.

[0064] 라인 버퍼 (130) 는 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 로부터의 출력을 유지하여, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 및 인덱싱된 컬러 히스토리 (135) 는 버퍼링된 비디오 데이터를 이용할 수 있다. 인덱싱된 컬러 히스토리 (135) 는 최근에 이용된 픽셀 값들을 저장한다. 이 최근에 이용된 픽셀 값들은 전용 선택스를 통해 비디오 인코더 (20) 에 의해 직접적으로 참조될 수 있다.

[0065] 엔트로피 인코더 (140) 는 인덱싱된 컬러 히스토리 (135) 와, 평탄성 검출기 (115) 에 의해 식별된 평탄성 전이들에 기초하여, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 로부터 수신된 예측 잔차들 및 임의의 다른 데이터 (예컨대, 예측기, 양자화기, 및 재구성기 컴포넌트 (125) 에 의해 식별된 인덱스들) 를 인코딩한다. 일부 예들에서, 엔트로피 인코더 (140) 는 서브스트림 인코더에 대하여 이 클럭 당 3 개의 샘플들을 인코딩할 수도 있다. 서브스트림 멀티플렉서 (145) 는 무헤더 패킷 (headless packet) 멀티플렉싱 방식에 기초하여 비트스트림을 멀티플렉싱할 수도 있다. 이것은 비디오 디코더 (30) 가 이 3 개의 엔트로피 디코더들을 병렬로 실행하도록 하여, 이 클럭 당 3 개의 픽셀들의 디코딩을 용이하게 한다. 서브스트림 멀티플렉서 (145) 는 패킷 순서를 최적화할 수도 있어서, 패킷들은 비디오 디코더 (30) 에 의해 효율적으로 디코딩될 수 있다. 클럭 당 2-제곱 픽셀들 (예컨대, 2 픽셀들/클럭 또는 4 픽셀들/클럭) 의 디코딩을 용이하게 할 수도 있는 엔트로피 코딩에 대한 상이한 접근법들이 구현될 수도 있다는 것에 주목한다.

[0066] **DSC 비디오 디코더**

[0067] 도 2b 는 이 개시물에서 설명된 양태들에 따라 기법들을 구현할 수도 있는 비디오 디코더 (30) 의 예를 예시하는 블록도이다. 비디오 디코더 (30) 는 이 개시물의 기법들 중의 일부 또는 전부를 수행하도록 구성될 수도 있다. 일부 예들에서, 이 개시물에서 설명된 기법들은 비디오 디코더 (30) 의 다양한 컴포넌트들 사이에서 공유될 수도 있다. 일부 예들에서, 추가적으로 또는 대안적으로, 프로세서 (도시되지 않음) 는 이 개시물에서 설명된 기법들 중의 일부 또는 전부를 수행하도록 구성될 수도 있다.

[0068] 설명의 목적들을 위하여, 이 개시물은 DSC 코딩의 맥락에서 비디오 디코더 (30) 를 설명한다. 그러나, 이 개시물의 기법들은 다른 코딩 표준들 또는 방법들에 적용가능할 수도 있다.

[0069] 도 2b 의 예에서, 비디오 디코더 (30) 는 복수의 기능적 컴포넌트들을 포함한다. 비디오 디코더 (30) 의 기능적 컴포넌트들은 레이트 버퍼 (155), 서브스트림 디멀티플렉서 (160), 엔트로피 디코더 (165), 레이트 제어기 (170), 예측기, 양자화기, 및 재구성기 컴포넌트 (175), 인덱싱된 컬러 히스토리 (180), 라인 버퍼 (185), 및 컬러-공간 변환기 (190) 를 포함한다. 비디오 디코더 (30) 의 예시된 컴포넌트들은 도 2a 에서의 비디오 인코더 (20) 와 관련하여 위에서 설명된 대응하는 컴포넌트들과 유사하다. 이와 같이, 비디오 디코더 (30) 의 컴포넌트들의 각각은 위에서 설명된 바와 같은 비디오 인코더 (20) 의 대응하는 컴포넌트들과 유사한 방식으로

동작할 수도 있다.

[0070] **DSC 에서의 슬라이스들**

[0071] 위에서 언급된 바와 같이, 슬라이스는 일반적으로, 이미지 또는 프레임에서의 영역들의 나머지로부터의 정보 없이 독립적으로 디코딩될 수 있는 이미지 또는 프레임에서의 공간적으로 별개의 영역을 지칭한다. 각각의 이미지 또는 비디오 프레임은 단일 슬라이스에서 인코딩될 수도 있거나, 또는 각각의 이미지 또는 비디오 프레임은 몇몇 슬라이스들에서 인코딩될 수도 있다. DSC 에서, 각각의 슬라이스를 인코딩하기 위하여 할당된 비트 수는 실질적으로 일정할 수도 있다. 슬라이스는 다수의 블록들로 구성될 수 있다.

[0072] **레이트 제어 버퍼**

[0073] 도 3 은 데이터의 슬라이스들을 인코딩하기 위한 예시적인 프로세스를 도시한 그래프이다. 도 3 을 참조하면, 수직축 (305) 은 레이트 버퍼 (예를 들어, 도 2a 의 비디오 인코더 (20) 상의 레이트 버퍼 (150), 또는 도 2b 의 비디오 디코더 (30) 의 레이트 버퍼 (155)) 이고, 수평축 (310) 은 시간을 나타낸다. 도 3 은 비디오 데이터의 제 1 슬라이스 (315) 및 비디오 데이터의 제 2 슬라이스 (320) 의 인코딩을 도시한다. 제 2 슬라이스 (320) 는, 제 1 슬라이스 (315) 의 인코딩 및 제 2 슬라이스 (320) 의 인코딩 동안 발생하는 시간적 오버랩을 단지 보여주기 위해 제 1 슬라이스 (315) 아래에 위치되는 것으로 도시되어 있으며, 제 2 슬라이스 (320) 가 제 1 슬라이스 (315) 의 비트 수보다 작다는 것을 나타내는 것으로 의도된다.

[0074] 설명을 목적으로, 본 개시물의 양태들은 도 2a 의 비디오 인코더 (20) 의 레이트 버퍼 (150) 를 참조하여 설명될 것이지만; 그러한 양태들은 또한 도 2b 의 비디오 디코더 (30) 또는 이들의 컴포넌트(들) (레이트 버퍼 (155)) 를 포함하지만 이에 한정되지 않음) 에 적용가능하다는 것을 이해할 것이다.

[0075] 레이트 버퍼 (150) 는 한정된 수의 비트만을 유지 (즉, 저장) 할 수 있다. 상기에서 논의된 바와 같이, 변수 BufferMaxSize는 레이트 버퍼의 전체 용량을 나타낼 수 있다. 그러나, 레이트 버퍼 (150) 에서의 언더플로우 및/또는 오버플로우를 방지하기 위해, 레이트 버퍼 (150) 를 BufferMaxSize보다 낮은 비트 수로 제한하는 것이 유리하다. 이와 같이, 레이트 버퍼 (150) 의 원하는 용량을 반영하도록 레이트 버퍼 (150) 에 대해 임시 최대 비트 레벨 (변수 tempBufferMaxSize 및 maxBufferBitsAtSliceEnd로 나타냄) 이 정의될 수 있다.

[0076] 325에서, 제 1 슬라이스 (315) 의 인코딩이 시작된다. 인코딩 동안, 비트는 레이트 버퍼 (150) 에 저장될 수 있고 레이트 버퍼 (150) 에 이전에 저장된 비트는 레이트 버퍼 (150) 로부터 제거될 수 있다. 예를 들어, 레이트 버퍼 (150) 로부터 제거된 비트는 (예를 들어, 소스 디바이스 (12) 에 의해, 링크 (16) 를 통해) 송신될 수 있다. 이와 같이, 레이트 버퍼 (150) 로의 비트들의 저장 및/또는 비트들을 제거하는 동작으로 인해, 레이트 버퍼 (150) 에서의 총 비트 수는 다양한 시점에서 증가 및/또는 감소할 수 있다. 일부 경우, 비트는 인코딩, 송신 등을 위해 함께 그룹화될 수 있다. 예를 들어, 비트는 픽셀들 (예를 들어, 픽셀당 6 비트) 또는 블록들 (예를 들어, 픽셀당 6 비트와 2x8 블록 크기에 대해 블록당 96 비트) 로 송신될 수 있다.

[0077] 330에서, 저장된 비트의 송신이 시작된다. 송신은, 도 3에 Init\_enc\_delay로 표시된 초기 인코딩 지연 기간을 지연 이후 시작할 수 있다. Init\_enc\_delay는 블록들 또는 픽셀들 단위로 측정될 수 있고 선택된 값 및/또는 고정된 값일 수 있거나 각 슬라이스에 대해 계산될 수 있다. 일 실시형태에서, Init\_enc\_delay는 maxBufferBitsAtSliceEnd에 기초할 수 있다. 예를 들어, maxBufferBitsAtSliceEnd가 4128이고 Init\_enc\_delay가 블록 단위로 측정되는 경우, Init\_enc\_delay는 블록당 목표 비트 수 (예를 들어, 96) 로 나눈 4128일 수 있다. 이것은, 제 1 슬라이스 (315) 가 완전히 디코딩된 후에 버퍼 (150) 가 제 1 슬라이스 (315) 로부터 나머지 블록들을 전송할 수 있도록 수행 될 수 있다.

[0078] 계속해서 도 3을 참조하면, 335에서, 제 1 슬라이스 (315) 의 인코딩이 완료될 수 있다. 즉, 335에서, 버퍼 (150) 에서의 비트 수는 maxBufferBitsAtSliceEnd와 동일해야 한다. 예를 들어, maxBufferBitsAtSliceEnd 가 4128이면, 제 1 슬라이스 (315) 의 인코딩이 완료될 때, 버퍼 (150) 의 비트 수는 이벤트 (335) 에서 4128과 동일해야 한다. 340에서, 제 1 슬라이스 (315) 의 송신이 완료된다. 도 3에 도시된 바와 같이, 제 1 슬라이스 (315) 에 대한 인코딩 (335) 의 끝은 제 2 슬라이스 (320) 에 대한 인코딩의 시작에 대응할 수 있는 한편, 제 1 슬라이스 (315) 에 대한 송신의 끝 (340) 은 제 2 슬라이스 (320) 에 대한 송신 (350) 의 시작에 대응할 수 있다. Init\_enc\_delay는 제 1 슬라이스 (315) 의 종료 이벤트 (예를 들어, 인코딩 및 송신의 끝) 를 제 2 슬라이스 (320) 의 시작 이벤트 (예를 들어, 인코딩 및 전송의 시작) 와 정렬하도록 선택될 수 있다. 즉, 제 2 슬라이스 (320) 에 대한 Init\_enc\_delay 기간은 제 1 슬라이스 (315) 에 대한 플러시 기간 (Flush\_time) 에 대응할 수 있다.

- [0079] 일 실시형태에서, 비트들은 Init\_enc\_delay 이후에 일정한 레이트로 버퍼 (150) 로부터 제거될 수 있다. 그렇게 함으로써, 슬라이스의 끝에 있는 버퍼 (150) 에서의 비트 수가 maxBufferBitsAtSliceEnd와 동일하다는 것을 보장할 수 있다. 또한, 비트들은 Flush\_time 동안 버퍼 (150) 로부터 일정한 레이트로 제거될 수 있다. 제 2 슬라이스 (320) 로부터의 압축된 블록들은, 제 2 슬라이스 (320) 에 대한 Init\_enc\_delay 기간이 종료된 후에 버퍼 (150) 로부터 제거된다.
- [0080] **BF 계산**
- [0081] maxBufferBitsAtSliceEnd가 BufferMaxSize보다 작다면, 얼마나 많은 슬라이스가 인코딩되었는지에 기초하여 BF 계산을 조절하는 것이 유리하다. 도 4는 슬라이스의 코딩 동안 BF를 조절하기 위한 예시적인 방법을 도시한다. 설명을 목적으로, 도 4는 도 2a의 비디오 인코더 (20) 의 레이트 버퍼 (150) 를 참조하여 설명될 것이다; 그러나, 도 4의 컴포넌트는 또한 도 2b의 비디오 디코더 (30) 또는 그 컴포넌트(들) (레이트 버퍼 (155) 를 포함하지만 이에 한정되지 않음) 에 적용가능하다는 것을 이해할 것이다.
- [0082] BF를 계산하기 위해, 변수 (tempBufferMaxSize) 가 정의될 수 있으며, 이는 슬라이스의 특정 지점에서의 버퍼 (150) 에 대한 임시 최대 비트 수를 나타낸다. 예를 들어, 슬라이스의 끝에서 tempBufferMaxSize는 maxBufferBitsAtSliceEnd와 동일하다. 또 다른 변수 (Offset) 는 특정 시점에서 얼마나 많은 BufferMaxSize가 사용불가능한지를 나타낸다.
- [0083] 도 4의 그래프는 그 수직축에 따라 오프셋 (405) 을 도시하고 그 수평축을 따라 슬라이스에서의 각 블록의 위치 (즉, 수) 를 도시한다. 블록은 순서대로 코딩될 수 있으므로, 각 개별 블록의 위치는 코딩된 블록 수 (이하, numBlocksCoded) 와 같을 수 있다. 예를 들어, numBlocksCoded의 값은 0에서  $N_T$ 로 표기되는 슬라이스에서의 총 블록 수 (즉, #blocksInSlice) 까지의 범위일 수 있다. 일 실시형태에서,  $N_p$ 로 표기된 블록들의 수는 BF가 조절되기 전에 코딩될 수 있다. 즉, numBlocksCoded는  $N_p$ 보다 작거나 동일하지만, 420에 나타낸 바와 같이 오프셋 (405) 은 제로와 동일하고 tempBufferMaxSize는 BufferMaxSize와 동일하다. 일 실시형태에서,  $N_p$ 는 임계치 위치를 제공하여,  $N_p$  또는 그 아래의 위치를 갖는 블록들이 BufferMaxSize와 동일한 tempBufferMaxSize를 가질 수 있게 하는 한편,  $N_p$  위의 위치를 갖는 블록들이 BufferMaxSize보다 작은 tempBufferMaxSize를 가질 수 있게 한다. 도 4에 도시된 바와 같이, tempBufferMaxSize의 값은  $N_p$  위의 각 후속 블록마다 선형적으로 감소할 수 있다. 즉,  $N_p$  이후의 위치를 갖는 블록은 슬라이스 내의 임의의 후속 블록의 tempBufferMaxSize보다 큰 고유한 개개의 tempBufferMaxSize를 가질 수 있다. 바꾸어 말하면, 도 4에 도시된 바와 같이,  $N_p$  이후의 위치를 갖는 블록들은,  $N_p$ 와  $N_T$  사이의 각 블록에 대해 감소하는 조절가능 값과 동일한 tempBufferMaxSize를 가질 수 있다.
- [0084] 도 4를 계속 참조하면, 예시적인 슬라이스 내의 블록들의 총 개수는  $N_T$ 로 표기된다. 따라서, numBlocksCoded가  $N_T$ 와 동일한 경우, tempBufferMaxSize는 maxBufferBitsAtSliceEnd와 동일해야 하고, 다른 말로 표현하면, 오프셋 (405) 은 MaxBufferSize - maxBufferBitsAtSliceEnd와 동일해야 한다. 예를 들어, 426에 나타낸 바와 같이, maxBufferBitsAtSliceEnd가 4128이면, tempBufferMaxSize는  $N_T$ 에서 4128과 동일해야 하고, 오프셋 (405) 은 MaxBufferSize - 4128과 동일해야 한다.
- [0085] 이를 달성하기 위해, tempBufferMaxSize 및 오프셋 (405) 은 numBlocksCoded =  $N_p$  와 numBlocksCoded =  $N_T$  사이에서  $B_R$ 로 표기되는 일정한 레이트로 조절될 수 있다.  $N_p$ 와  $N_T$ 의 차이는 도 3에서  $N_R$ 로 표기된다.  $B_R$ 은 지점들 (numBlocksCoded =  $N_p$ , 오프셋 = 0) 과 (numBlocksCoded =  $N_T$ , 오프셋 = bufferMaxSize - maxBufferBitsAtSliceEnd) 사이에 선을 형성한다. 예를 들어, maxBufferBitsAtSliceEnd가 4128이면,  $B_R$ 은 (maxBufferSize - 4128)/ $N_R$ 로 정의될 수 있다.  $B_R$ 에 따른 예시적인 포인트에 대한 오프셋 (405) 및 tempBufferMaxSize의 값은 422 및 424로 도시된다.
- [0086] 이러한 방식으로,  $B_R$ 에 따른 지점들은 슬라이스의 개별 블록들에 대해 대응하는 tempBufferMaxSize를 제공할 수 있다. 대응하는 tempBufferMaxSize는 이후 슬라이스에서의 각 블록에 대한 개별 BF를 계산할 수 있다. 일 실시형태에서, 슬라이스에서의 블록들은 슬라이스 내의 그들의 위치에 기초하여 정의될 수 있다.

[0087] 일 실시형태에서, 위의 식 1 및 2는  $N_p$  또는 그 아래에 위치하는 슬라이스 내의 블록들에 대한 BF를 계산하는데 사용될 수 있다. 예를 들어,  $N_p$ 가 2와 동일한 경우, BF는 제 1 및 제 2 블록들에 대해 식 1 및 2에 따라 계산될 수 있다. 임의의 나머지 블록들에 대해, BF는 식 1 및/또는 2에서 BufferMaxSize 대신에 tempBufferMaxSize를 치환함으로써 계산될 수 있다. 즉, BF는 다음을 사용하여 계산될 수 있다:

[0088]  $BF = ((BufferCurrentSize * 100) / tempBufferMaxSize)$  (식 3)

[0089] 또는, 다음을 사용:

[0090]  $BF = ((BufferCurrentSize * 100) + (tempBufferMaxSize >> 1) / tempBufferMaxSize)$  (식 4)

[0091] 또한, tempBufferMaxSize는 다음을 사용하여 계산될 수 있다:

[0092]  $tempBufferMaxSize = BufferMaxSize - 오프셋$  (식 5),

[0093] 여기서

[0094]  $오프셋 = B_R * (numBlocksCoded - N_p + 1)$  (식 6)

[0095] 그리고 여기서

[0096]  $BR = (BufferMaxSize - tempBufferMaxSize) / (numBlocksCoded - N_p)$  (식 7)

[0097] 요약하면, 다음의 예시적인 코드가 BF를 계산하는데 사용될 수 있다:

```
numBlocksCoded = 0; tempBufferMaxSize = BufferMaxSize;
BR = (BufferMaxSize - maxBufferBitsAtSliceEnd) / (NT - NP);
for i = 1: NT
{
    code ith block
    numBlocksCoded += 1;
    if(numBlocksCoded >= Np)
    {
        tempBufferMaxSize = BufferMaxSize - (BR * (numBlocksCoded - Np + 1));
    }
    calculate BF
}
```

[0098]

[0099] **CBR을 제공하기 위한 예시적인 플로우차트**

[0100] 도 5를 참조하여, 버퍼 (예를 들어, 도 2a에서의 비디오 인코더 (20) 상의 레이트 버퍼 (150), 또는 도 2b에서의 비디오 디코더 (30)의 레이트 버퍼 (155))에서 버퍼 충만도를 결정하기 위한 예시적인 절차를 설명할 것이다. 설명을 목적으로, 도 5의 단계들은 도 2a에서의 비디오 인코더 (20)의 레이트 버퍼 (150)를 참조하여 설명할 것이다. 하지만, 도 5에 나타난 단계들은 비디오 인코더 (예를 들어, 도 2a에서의 비디오 인코더 (20)), 비디오 디코더 (예를 들어, 도 2b에서의 비디오 디코더 (30)), 또는 이들의 컴포넌트(들)에 의해 수행될 수도 있다. 편의를 위해, 방법 (500)은, 비디오 인코더 (20), 비디오 디코더 (30), 또는 다른 컴포넌트일 수도 있는, 비디오 코더 (또한 간단히 코더라고도 함)에 의해 수행되는 것으로 기재된다.

[0101] 방법 (500)은 블록 501에서 시작한다. 블록 505에서, 프로세스 (500)는 데이터의 슬라이스 내에서 데이터의 블록을 코딩하는 것을 수반할 수 있다. 일 실시형태에서, 코딩은 프로세서에 의해 수행될 수 있다. 상술한 바와 같이, 슬라이스는, 이미지 또는 프레임에서의 나머지 영역들로부터의 정보를 사용하지 않고 독립적으로 디코딩될 수 있는 이미지 또는 프레임에서의 임의의 공간적으로 구별되는 영역일 수 있다. 슬라이스는 하나 이상의 블록들 및/또는 픽셀들로 구성될 수 있다.

[0102] 블록 510에서, 프로세스 (500)는 그 블록에 대응하는 하나 이상의 비트들을 레이트 버퍼 (150)에 저장하는 것을 수반할 수 있다. 예를 들어, 블록들은 압축될 수 있고 대응하는 비트들은 레이트 버퍼 (150)에 일시적으로 저장될 수 있다. 일 실시형태에서, 개별 비트들은 코딩된 후에 버퍼 (150)에 추가될 수 있다.

대안의 실시형태에서, 다수의 비트들이 동시에 버퍼 (150) 에 추가될 수 있다.

[0103] 블록 515 에서, 프로세스 (500) 는 슬라이스에서의 다른 블록들에 대한 하나 이상의 블록들의 위치를 나타내는 정보를 결정하는 것을 수반할 수 있다. 예를 들어, 주어진 블록이 제 3 코딩된 블록이고, 그러한 블록이 슬라이스의 제 3 블록으로 정의될 수 있다는 것을 결정할 수 있다. 블록의 위치를 결정함으로써, 슬라이스 내의 이전에 코딩된 블록들의 수 및/또는 코딩되지 않은 슬라이스 내의 블록들의 수를 결정할 수 있다. 일 실시형태에서, 위치를 나타내는 정보를 결정하는 것은 프로세서에 의해 수행될 수 있다.

[0104] 블록 520 에서, 프로세스 (500) 는 하나 이상의 블록 위치들에서 BF를 결정하는 것을 수반할 수 있다. 이 결정은 레이트 제어에 이용될 수 있다. 예를 들어, 도 4를 참조하면, 블록의 위치는 블록과 관련된  $B_R$ 에 따르는 지점을 나타낼 수 있다. 이를 결정하기 위해,  $B_R$  계산에 사용될 수 있는  $\text{maxBufferBitsAtSliceEnd}$ 를 알 필요가 있을 수 있다.  $B_R$ 에 따른 관련 위치는 블록에 대해 대응하는  $\text{tempBufferMaxSize}$ 를 나타낼 수 있다. 다음, BF는 블록에 대해  $\text{tempBufferMaxSize}$ 로  $\text{BufferCurrentSize}$ 를 나눠 계산할 수 있다. 일 실시형태에서, BF를 결정하는 것은 프로세서에 의해 수행될 수 있다. 프로세스 (500) 는 블록 525 에서 종료한다.

[0105] 일 실시형태들에서,  $\text{tempBufferMaxSize}$ 는 정수 값,  $c$ 에 의해 조절될 수 있다. 예를 들어,  $\text{tempBufferMaxSize}$ 는 다음을 사용하여 계산할 수 있다:

[0106] 
$$\text{tempBufferMaxSize} = \text{BufferMaxSize} - (B_R * (\text{numBlocksCoded} - N_P + c)) \quad (\text{식 } 8)$$

[0107] 일 예에서,  $c$ 는 0, 1, -1 또는 임의의 다른 정수 값과 동일할 수 있다. 다른 실시형태에서,  $B_R$ 은 정수 값,  $d$ 에 의해 조절될 수 있다. 예를 들어,  $B_R$  은 다음을 사용하여 계산할 수 있다:

[0108] 
$$B_R = (\text{BufferMaxSize} - \text{maxBufferBitsAtSliceEnd}) / (N_T - N_P + d) \quad (\text{식 } 9)$$

[0109] 여기서  $d$  는 0, 1, -1 또는 임의의 다른 정수 값과 동일할 수 있다. 상기 식의 다른 변수에 대한 유사한 조절이 또한 이루어질 수 있다.

[0110] **다른 고려사항들**

[0111] 본원에서 개시된 정보 및 신호들은 다양한 상이한 기술들 및 기법들 중의 임의의 것을 이용하여 표현될 수도 있다. 예를 들어, 상기 설명의 전반에 걸쳐 참조될 수도 있는 데이터, 명령들, 커맨드 (command) 들, 정보, 신호들, 비트들, 심볼들, 및 칩들은 전압들, 전류들, 전자기파들, 자기장들 또는 입자들, 광학 필드들 또는 입자들, 또는 그 임의의 조합에 의해 표현될 수도 있다.

[0112] 본원에서 개시된 실시형태들과 관련하여 설명된 다양한 예시적인 논리적 블록들 및 알고리즘 단계들은 전자 하드웨어, 컴퓨터 소프트웨어, 또는 양자의 조합들로서 구현될 수도 있다. 하드웨어 및 소프트웨어의 이 교환 가능성을 명확하게 예시하기 위하여, 다양한 예시적인 컴포넌트들, 블록들, 및 단계들은 일반적으로 그 기능성의 측면에서 위에서 설명되었다. 이러한 기능성이 하드웨어 또는 소프트웨어로서 구현되는지 여부는 특정한 애플리케이션과, 전체 시스템에 부과된 설계 제약들에 종속된다. 당업자들은 각각의 특별한 애플리케이션을 위한 다양한 방법들로 설명된 기능성을 구현할 수도 있지만, 이러한 구현 판단들은 본 개시물의 범위로부터의 이탈을 야기시키는 것으로 해석되지 않아야 한다.

[0113] 본원에서 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 그 임의의 조합으로 구현될 수도 있다. 이러한 기법들은 범용 컴퓨터들, 무선 통신 디바이스 핸드셋들, 또는 무선 통신 디바이스 핸드셋들 및 다른 디바이스들에서의 애플리케이션을 포함하는 다수의 용도들을 가지는 집적 회로 디바이스들과 같은 다양한 디바이스들 중의 임의의 것에서 구현될 수도 있다. 디바이스들 또는 컴포넌트들로서 설명된 임의의 특징들은 집적된 로직 디바이스에서 함께, 또는 개별적이지만 상호 동작가능한 로직 디바이스들로서 별도로 구현될 수도 있다. 소프트웨어로 구현될 경우, 기법들은, 실행될 때, 위에서 설명된 방법들 중의 하나 이상을 수행하는 명령들을 포함하는 프로그램 코드를 포함하는 컴퓨터-판독가능 데이터 저장 매체에 의해 적어도 부분적으로 실현될 수도 있다. 컴퓨터-판독가능 데이터 저장 매체는 패키징 재료들을 포함할 수도 있는 컴퓨터 프로그램 제품의 일부를 형성할 수도 있다. 컴퓨터-판독가능 매체는 동기식 랜덤 액세스 메모리 (synchronous dynamic random access memory; SDRAM) 와 같은 랜덤 액세스 메모리 (random access memory; RAM), 판독-전용 메모리 (read-only memory; ROM), 비-휘발성 랜덤 액세스 메모리 (non-volatile random access memory; NVRAM), 전기

적 소거가능 프로그래밍가능 판독-전용 메모리 (electrically erasable programmable read-only memory; EEPROM), 플래시 메모리 (FLASH memory), 자기 또는 광학 데이터 저장 매체들 등과 같은 메모리 또는 데이터 저장 매체들을 포함할 수도 있다. 추가적으로 또는 대안적으로, 기법들은 전파된 신호들 또는 파 (wave) 들과 같이, 명령들 또는 데이터 구조들의 형태로 프로그램 코드를 반송하거나 통신하며 컴퓨터에 의해 액세스, 판독, 및/또는 실행될 수 있는 컴퓨터-판독가능 통신 매체에 의해 적어도 부분적으로 실현될 수도 있다.

[0114] 프로그램 코드는, 하나 이상의 디지털 신호 프로세서 (digital signal processor; DSP) 들, 범용 마이크로프로세서들, 애플리케이션 특정 집적 회로 (application specific integrated circuit; ASIC) 들, 필드 프로그래밍 가능한 로직 어레이 (field programmable logic array; FPGA) 들, 또는 다른 등가의 집적 또는 개별 로직 회로 부와 같은 하나 이상의 프로세서들을 포함할 수도 있는 프로세서에 의해 실행될 수도 있다. 이러한 프로세서는 이 개시물에서 설명된 기법들 중의 임의의 것을 수행하도록 구성될 수도 있다. 범용 프로세서는 마이크로프로세서일 수도 있지만, 대안적으로, 프로세서는 임의의 기존의 프로세서, 제어기, 마이크로제어기, 또는 상태 머신일 수도 있다. 프로세서는 또한, 컴퓨팅 디바이스들의 조합, 예컨대, DSP 및 마이크로프로세서, 복수의 마이크로프로세서들, DSP 코어와 함께 하나 이상의 마이크로프로세서들, 또는 임의의 다른 이러한 구성의 조합으로서 구현될 수도 있다. 따라서, 본원에서 이용된 바와 같은 용어 "프로세서" 는 상기한 구조, 상기 구조의 임의의 조합, 또는 본원에서 설명된 기법들의 구현을 위해 적당한 임의의 다른 구조 또는 장치 중의 임의의 것을 지칭할 수도 있다. 게다가, 일부의 양태들에서, 본원에서 설명된 기능성은, 인코딩 및 디코딩을 위해 구성되거나, 조합된 비디오 인코더-디코더 (combined video encoder-decoder; CODEC) 내에 편입된 전용 소프트웨어 또는 하드웨어 내에서 제공될 수도 있다. 또한, 기법들은 하나 이상의 회로들 또는 로직 엘리먼트들에서 완전히 구현될 수도 있다.

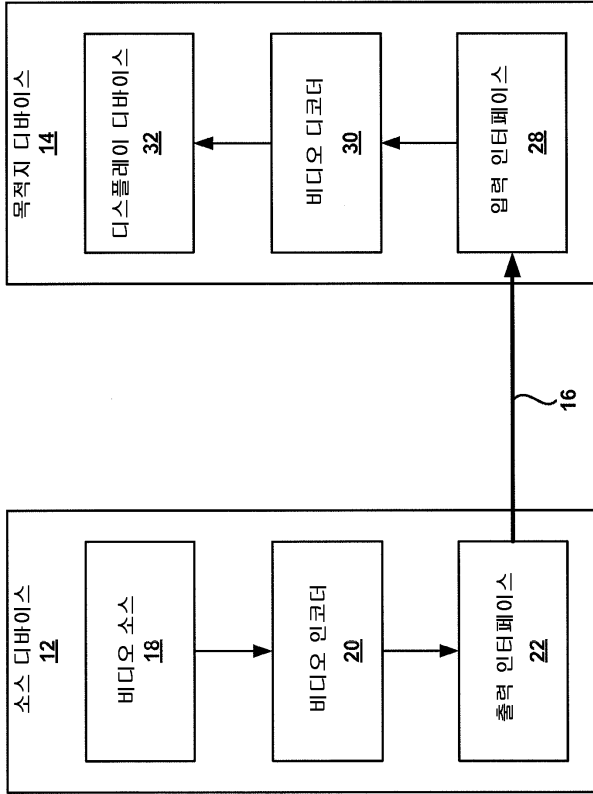
[0115] 이 개시물의 기법들은 무선 핸드셋 (wireless handset), 집적 회로 (integrated circuit; IC) 또는 IC 들의 세트 (예를 들어, 칩세트) 를 포함하는 광범위한 디바이스들 또는 장치들에서 구현될 수도 있다. 다양한 컴포넌트들 또는 유닛들은 개시된 기법들을 수행하도록 구성된 디바이스들의 기능적 양태들을 강조하기 위하여 이 개시물에서 설명되어 있지만, 상이한 하드웨어 유닛들에 의한 실현을 반드시 요구하지는 않는다. 오히려, 위에서 설명된 바와 같이, 다양한 유닛들은 코텍 하드웨어 유닛 내에 조합될 수도 있거나, 적당한 소프트웨어 및/또는 펌웨어와 함께, 위에서 설명된 바와 같은 하나 이상의 프로세서들을 포함하는 상호동작하는 하드웨어 유닛들의 집합에 의해 제공될 수도 있다.

[0116] 상기한 것은 다양한 상이한 실시형태들과 관련하여 설명되었지만, 하나의 실시형태로부터의 특징들 또는 구성요소들은 이 개시물의 교시사항들로부터 이탈하지 않으면서 다른 실시형태들과 조합될 수도 있다. 그러나, 개개의 실시형태들 사이의 특징들의 조합들은 반드시 그것으로 제한되지는 않는다. 개시물의 다양한 실시형태들이 설명되었다. 이러한, 그리고 다른 양태들은 다음의 청구항들의 범위 내에 있다.

도면

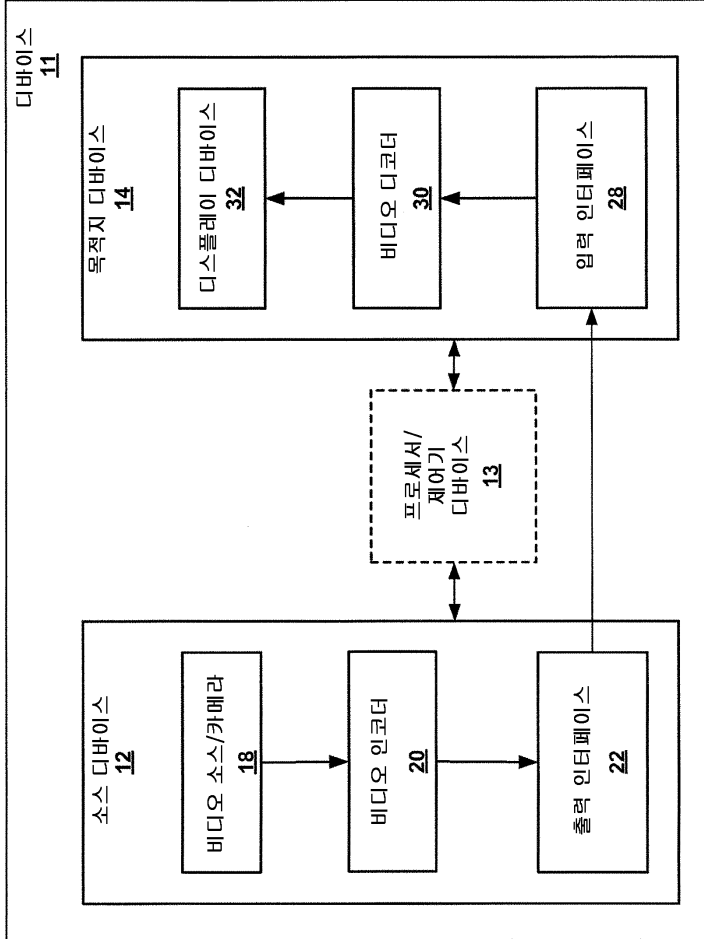
도면1a

10

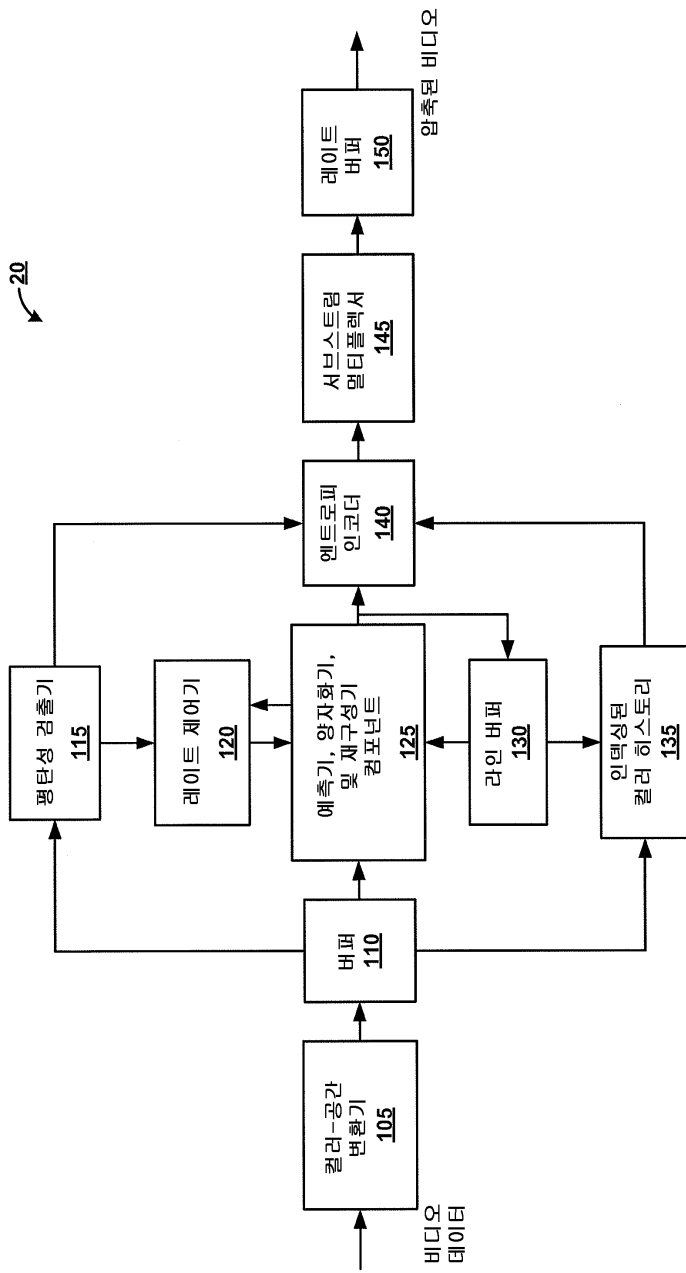


도면1b

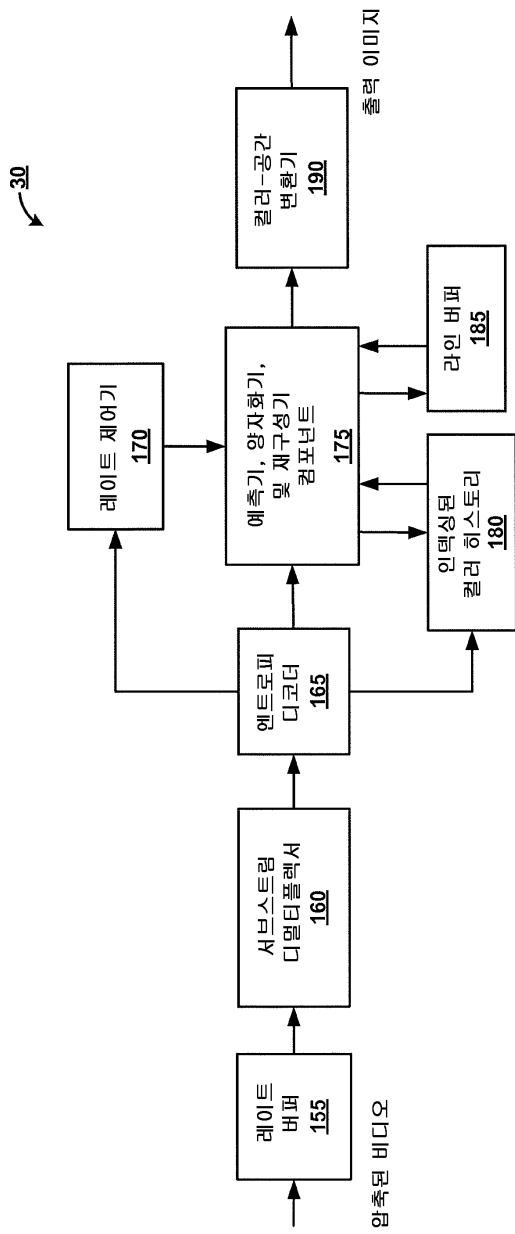
10'



도면2a

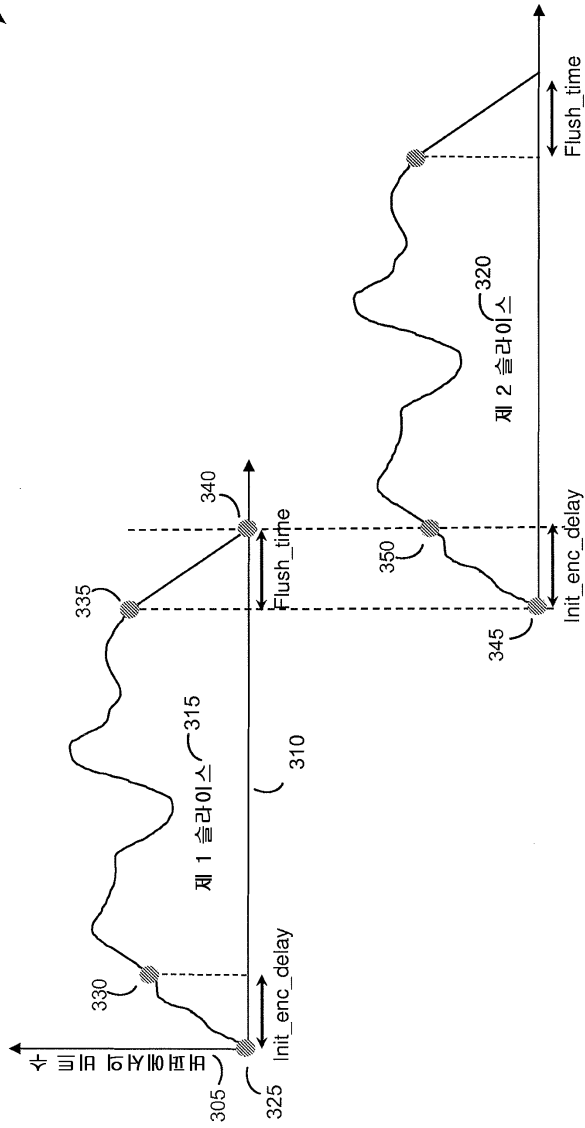


도면2b

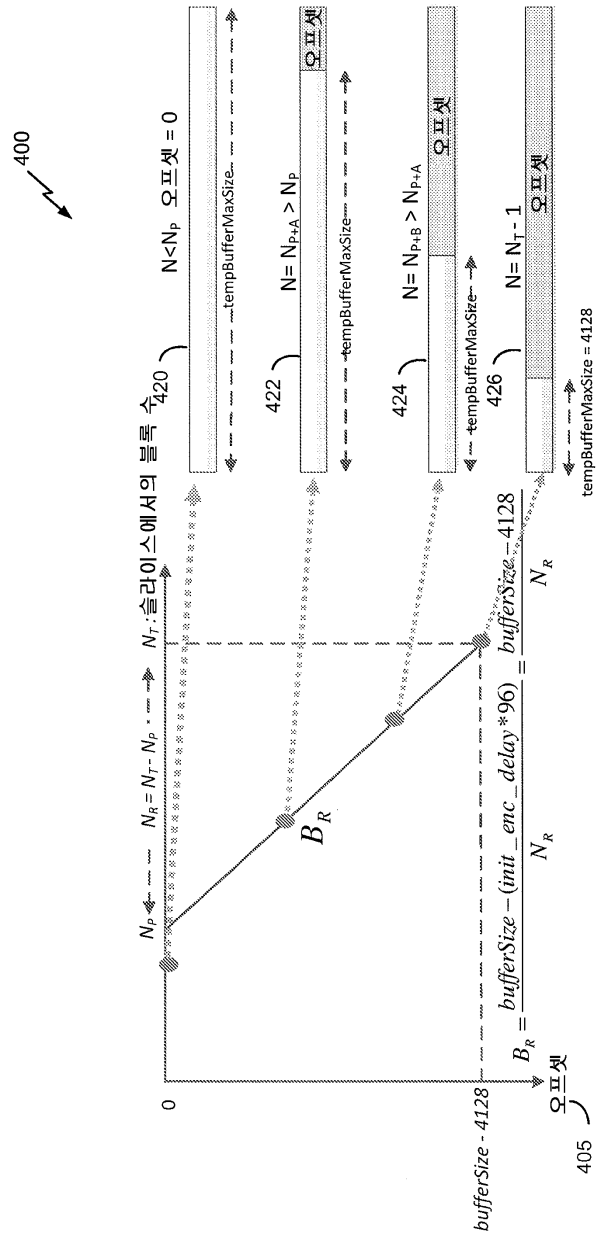


도면3

300 ↗



도면4



도면5

500 ↘

