(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2023년12월13일
(11) 등록번호 10-2613593
(24) 등록일자 2023년12월11일

(51) 국제특허분류(Int. Cl.)
H04N 19/70 (2014.01) *H04N 19/46* (2014.01)
H04N 21/6332 (2011.01) *H04N 21/854* (2011.01)
(52) CPC특허분류
H04N 19/70 (2015.01)
H04N 19/46 (2015.01)
(21) 출원번호 10-2019-7030601
(22) 출원일자(국제) 2018년03월21일
심사청구일자 2021년03월04일
(85) 번역문제출일자 2019년10월17일
(65) 공개번호 10-2019-0122867
(43) 공개일자 2019년10월30일
(86) 국제출원번호 PCT/US2018/023598
(87) 국제공개번호 WO 2018/175609
국제공개일자 2018년09월27일
(30) 우선권주장
62/474,540 2017년03월21일 미국(US)
15/926,833 2018년03월20일 미국(US)
(56) 선행기술조사문헌
KR1020160094998 A*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
(72) 발명자
왕 예-쿠이
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
슈토크하머 토마스
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
(74) 대리인
특허법인코리아나

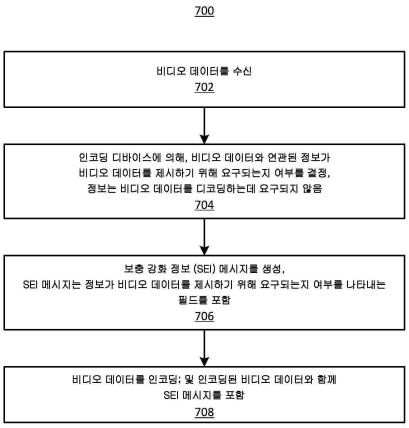
전체 청구항 수 : 총 26 항 심사관 : 김혜린

(54) 발명의 명칭 **필수 및 비필수 비디오 보충 정보의 시그널링**

(57) 요 약

여러 구현들에서, 비필수 정보로부터 필수 정보를 구별하는 SEI 메시지들을 위한 기법들 뿐아니라 이들 기법들을 구현하는 시스템들이 제공된다. 여러 구현들에서, 인코더는 비디오 데이터와 연관된 정보가 필수적인지 또는 필수적이지 않은지 여부를 결정하도록 구성될 수 있다. 여러 구현들에서, 상이한 유형들의 SEI 메시지들이 정의될 수 있으며, 여기서 이들 유형들의 SEI 메시지들 중 하나 이상은 특정의 비디오와 연관된 정보가 필수적인지 또는 필수적이지 않은지 여부를 표시할 수 있다. 필수 정보는 비디오 데이터를 제시하기 위해 요구되고, 비디오 데이터를 디코딩하기 위해 요구되지 않는다.

대 표 도 - 도7



(52) CPC특허분류
H04N 21/6332 (2013.01)
H04N 21/85406 (2013.01)

명 세 서

청구범위

청구항 1

회로에 구현된 인코딩 디바이스에 의해, 비디오 데이터를 인코딩하는 방법으로서,

인코딩할 상기 비디오 데이터를 얻는 단계;

상기 인코딩 디바이스에 의해 구현된 기준에 기초하여, 디코딩된 비디오 비트 스트림의 코딩된 비디오 시퀀스 (CVS) 의 컨텐츠를 프로세스하는 것과 연관된 어느 보충 강화 정보 (SEI) 메시지가 바람직한 사용자 경험을 가능하게 하기 위해 필수적인 것으로 고려되는지 여부를 결정하는 단계로서, 상기 디코딩된 비디오 비트 스트림은 상기 비디오 데이터와 연관된 인코딩된 비디오 비트 스트림을 디코딩한 결과인, 상기 결정하는 단계;

상기 CVS 내 SEI 메시지들의 유형들의 세트가 필수적인 것으로서 고려된다는 상기 인코딩 디바이스의 결정에 기초하여:

SEI 메시지를 생성하는 단계로서, 상기 SEI 메시지는 상기 인코딩 디바이스에 의해 상기 CVS 내 상기 SEI 메시지들의 유형들의 세트가 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 나타내는 정보를 포함하고, 상기 SEI 메시지들의 유형들의 세트는 상기 인코딩된 비디오 비트 스트림의 디코딩에 영향을 미치지 않는, 상기 SEI 메시지를 생성하는 단계;

상기 SEI 메시지 내에 포함시키기 위해, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 나타내는 제 1 신택스 엘리먼트를 생성하는 단계;

상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지의 각 유형에 대해, 상기 SEI 메시지 내에 포함시키기 위해, SEI 메시지의 유형을 나타내는 제 2 신택스 엘리먼트를 생성하는 단계;

상기 SEI 메시지 내에 제 1 신택스 엘리먼트와 제 2 신택스 엘리먼트를 포함시키는 단계;

상기 인코딩된 비디오 비트 스트림을 생성하기 위해 상기 비디오 데이터를 인코딩하는 단계; 및

상기 인코딩된 비디오 비트 스트림과 상기 SEI 메시지를 송신하는 단계를 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 2

제 1 항에 있어서,

요건에 기초하여, 상기 CVS의 제 1 액세스 유닛 내에 상기 SEI 메시지를 포함시키는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 3

제 1 항에 있어서,

요건에 기초하여, 상기 SEI 메시지를 현재의 액세스 유닛에 포함시키는 단계 및 상기 현재의 액세스 유닛에서 상기 CVS의 종단까지 각 액세스 유닛에 상기 SEI 메시지를 포함시키는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 4

제 1 항에 있어서,

컨텐츠 제공자로부터 수신된 입력에 기초하여, 상기 인코딩 디바이스에 의해 구현된 상기 기준을 정의하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 5

제 1 항에 있어서,

스트리밍을 위해 상기 인코딩된 비디오 비트 스트림을 캡슐화하는 단계로서, 상기 인코딩된 비디오 비트 스트림은 스트리밍 포맷에 따라 캡슐화되는, 상기 캡슐화하는 단계; 및

상기 스트리밍 포맷에 따라, 인코딩된 비디오 비트 스트림의 디스크립션을 생성하는 단계로서, 상기 디스크립션은 상기 SEI 메시지를 포함하는, 상기 디스크립션을 생성하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 6

제 1 항에 있어서,

네트워크를 통한 송신을 위해 상기 인코딩된 비디오 비트 스트림을 캡슐화하는 단계로서, 상기 인코딩된 비디오 비트 스트림은 컨테이너 포맷에 따라 캡슐화되는, 상기 캡슐화하는 단계;

상기 컨테이너 포맷에 따라 제 1 패킷을 생성하는 단계로서, 상기 제 1 패킷의 페이로드 부분은 디스크립터로서 상기 SEI 메시지를 포함하는, 상기 제 1 패킷을 생성하는 단계; 및

상기 인코딩된 비디오 비트 스트림을 포함하는 하나 이상의 패킷들을 생성하는 단계로서, 상기 하나 이상의 패킷들은 송신 순서에서 상기 제 1 패킷에 후속하는, 상기 하나 이상의 패킷들을 생성하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 7

제 1 항에 있어서,

네트워크를 통한 송신을 위해 상기 인코딩된 비디오 비트 스트림을 캡슐화하는 단계로서, 상기 인코딩된 비디오 비트 스트림은 실시간 전송 프로토콜에 따라 캡슐화되는, 상기 캡슐화하는 단계; 및

세션 디스크립션 프로토콜에 따라 세션 디스크립션을 생성하는 단계로서, 상기 세션 디스크립션은 상기 SEI 메시지를 포함하는, 상기 세션 디스크립션을 생성하는 단계를 더 포함하는, 비디오 데이터를 인코딩하는 방법.

청구항 8

인코딩 디바이스로서,

인코딩할 비디오 데이터를 저장하도록 구성된 메모리; 및

회로에 구현되고 상기 메모리와 통신하는 프로세서를 포함하고,

상기 프로세서는,

상기 인코딩 디바이스에 의해 구현된 기준에 기초하여, 디코딩된 비디오 비트 스트림의 코딩된 비디오 시퀀스 (CVS) 의 컨텐츠를 프로세스하는 것과 연관된 어느 보충 강화 정보 (SEI) 메시지가 바람직한 사용자 경험을 가능하게 하기 위해 필수적인 것으로 고려되는지 여부를 결정하는 것으로서, 상기 디코딩된 비디오 비트 스트림은 상기 비디오 데이터와 연관된 인코딩된 비디오 비트 스트림을 디코딩한 결과인, 상기 어느 보충 강화 정보 (SEI) 메시지가 필수적인 것으로 고려되는지 여부를 결정하고;

상기 CVS 내 SEI 메시지들의 유형들의 세트가 필수적인 것으로서 고려된다는 상기 인코딩 디바이스의 결정에 기초하여:

SEI 메시지를 생성하는 것으로서, 상기 SEI 메시지는 상기 인코딩 디바이스에 의해 상기 CVS 내 상기 SEI 메시지들의 유형들의 세트가 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 나타내는 정보를 포함하고, 상기 SEI 메시지들의 유형들의 세트는 상기 인코딩된 비디오 비트 스트림의 디코딩에 영향을 미치지 않는, 상기 SEI 메시지를 생성하고;

상기 SEI 메시지 내에 포함시키기 위해, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 나타내는 제 1 신택스 엘리먼트를 생성하고;

상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지의 각 유형에 대해, 상기 SEI 메시지 내에 포함시키기 위해, SEI 메시지의 유형을 나타내는 제 2 신택스 엘리먼트를 생성하고;

상기 SEI 메시지 내 제 1 신택스 엘리먼트와 제 2 신택스 엘리먼트를 포함시키고;

상기 인코딩된 비디오 비트 스트림을 생성하기 위해 상기 비디오 데이터를 인코딩하고; 및

상기 인코딩된 비디오 비트 스트림과 상기 SEI 메시지를 송신하도록 구성되는, 인코딩 디바이스.

청구항 9

제 8 항에 있어서,

상기 프로세서는 또한 요건에 기초하여, 상기 CVS의 제 1 액세스 유닛 내에 상기 SEI 메시지를 포함시키도록 구성되는, 인코딩 디바이스.

청구항 10

제 8 항에 있어서,

상기 프로세서는 또한 요건에 기초하여, 상기 SEI 메시지를 현재의 액세스 유닛에 포함시키고, 상기 현재의 액세스 유닛에서 상기 CVS의 종단까지 각 액세스 유닛에 상기 SEI 메시지를 포함시키도록 구성되는, 인코딩 디바이스.

청구항 11

제 8 항에 있어서,

상기 프로세서는 또한 컨텐츠 제공자로부터 수신된 입력에 기초하여, 상기 인코딩 디바이스에 의해 구현된 상기 기준을 정의하도록 구성되는, 인코딩 디바이스.

청구항 12

명령들을 포함하는 비일시적 컴퓨터 판독가능 저장 매체로서,

상기 명령들은, 하나 이상의 프로세서들에 의해 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

인코딩할 비디오 데이터를 수신하게 하고;

상기 인코딩 디바이스에 의해 구현된 기준에 기초하여, 디코딩된 비디오 비트 스트림의 코딩된 비디오 시퀀스 (CVS) 의 컨텐츠를 프로세스하는 것과 연관된 어느 보충 강화 정보 (SEI) 메시지가 바람직한 사용자 경험을 가능하게 하기 위해 필수적인 것으로 고려되는지 여부를 결정하게 하는 것으로서, 상기 디코딩된 비디오 비트 스트림은 상기 비디오 데이터와 연관된 인코딩된 비디오 비트 스트림을 디코딩한 결과인, 상기 어느 보충 강화 정보 (SEI) 메시지가 필수적인 것으로 고려되는지 여부를 결정하게 하고;

상기 CVS 내 SEI 메시지들의 유형들의 세트가 필수적인 것으로서 고려된다는 상기 인코딩 디바이스의 결정에 기초하여:

SEI 메시지를 생성하게 하는 것으로서, 상기 SEI 메시지는 상기 인코딩 디바이스에 의해 상기 CVS 내 상기 SEI 메시지들의 유형들의 세트가 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 나타내는 정보를 포함하고, 상기 SEI 메시지들의 유형들의 세트는 상기 인코딩된 비디오 비트 스트림의 디코딩에 영향을 미치지 않는, 상기 SEI 메시지를 생성하게 하고;

상기 SEI 메시지 내에 포함시키기 위해, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 나타내는 제 1 신택스 엘리먼트를 생성하게 하고;

상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지의 각 유형에 대해, 상기 SEI 메시지 내에 포함시키기 위해, SEI 메시지의 유형을 나타내는 제 2 신택스 엘리먼트를 생성하게 하고;

상기 SEI 메시지 내 제 1 신택스 엘리먼트와 제 2 신택스 엘리먼트를 포함시키게 하고;

상기 인코딩된 비디오 비트 스트림을 생성하기 위해 상기 비디오 데이터를 인코딩하게 하고; 및

상기 인코딩된 비디오 비트 스트림과 상기 SEI 메시지를 송신하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 13

비디오 데이터를 디코딩하는 방법으로서,

디코딩 디바이스에서, 인코딩된 비디오 비트 스트림 및 상기 인코딩된 비디오 비트 스트림과 연관된 보충 강화 정보 (SEI) 메시지를 수신하는 단계;

상기 인코딩된 비디오 비트 스트림의 인코딩 디바이스에 의해, 상기 인코딩된 비디오 비트 스트림의 코딩된 비디오 시퀀스 (CVS) 내의 SEI 메시지들의 유형들의 세트가 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 나타내는 정보를 포함하는 상기 SEI 메시지를 프로세스하는 단계로서, 상기 SEI 메시지들의 유형들의 세트는 상기 인코딩된 비디오 비트 스트림의 디코딩에 영향을 미치지 않는, 상기 프로세스하는 단계;

상기 SEI 메시지 내에서, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 나타내는 제 1 신택스 엘리먼트를 수신하는 단계;

상기 제 1 신택스 엘리먼트에 기초하여, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 결정하는 단계;

상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지의 각 유형에 대해, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수의 결정에 대응하여, SEI 메시지의 유형을 나타내는 제 2 신택스 엘리먼트를 수신하는 단계;

상기 인코딩된 디바이스에 의해, 상기 SEI 메시지들의 유형들의 세트 중 각 SEI 메시지의 유형이 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 결정하는 단계; 및

디코딩된 비디오 데이터를 생성하기 위해 상기 인코딩된 비디오 비트 스트림을 디코딩하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 14

제 13 항에 있어서,

상기 CVS의 제 1 액세스 유닛에서 상기 SEI 메시지를 수신하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 15

제 13 항에 있어서,

현재의 액세스 유닛 및 상기 현재의 액세스 유닛에서 상기 CVS의 종단까지 각 액세스 유닛에서 상기 SEI 메시지를 수신하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 16

제 13 항에 있어서,

상기 SEI 메시지의 컨텐츠에 기초하여, 상기 디코딩 디바이스가 상기 바람직한 사용자 경험을 가능하게 하는 것을 제공하기 위해 상기 CVS를 프로세스하기 위해 구성되는 시스템 레벨 프로세스 엔티티를 포함하는지를 결정하는 단계를 더 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 17

제 13 항에 있어서,

상기 인코딩된 비디오 비트 스트림 및 상기 인코딩된 비디오 비트 스트림과 연관된 상기 SEI 메시지는 파일로 수신되고, 상기 파일은 파일 포맷에 따라 포맷되고, 상기 파일 포맷에 따라, 상기 SEI 메시지는 상기 인코딩된 비디오 비트 스트림을 포함하는 상기 파일의 부분 앞의 상기 디코딩 장치에 의해 판독되는 상기 파일의 부분에 존재하는, 비디오 데이터를 디코딩하는 방법.

청구항 18

제 13 항에 있어서,

상기 인코딩된 비디오 비트 스트림 및 상기 인코딩된 비디오 비트 스트림과 연관된 상기 SEI 메시지는 데이터 스트림으로 수신되며, 상기 데이터 스트림은 스트리밍 포맷에 따라 포맷되고;

상기 데이터 스트림으로부터, 상기 인코딩된 비디오 비트 스트림의 디스크립션을 판독하는 단계를 더 포함하며, 상기 디스크립션은 상기 SEI 메시지를 포함하는, 비디오 데이터를 디코딩하는 방법.

청구항 19

제 13 항에 있어서,

상기 인코딩된 비디오 비트 스트림 및 상기 인코딩된 비디오 비트 스트림과 연관된 상기 SEI 메시지는 복수의 네트워크 패킷들에서 수신되고;

상기 복수의 네트워크 패킷들로부터 제 1 패킷의 페이로드 부분으로부터 상기 SEI 메시지를 판독하는 단계; 및

상기 복수의 네트워크 패킷들로부터의 하나 이상의 패킷들로부터 상기 인코딩된 비디오 비트 스트림을 판독하는 단계를 더 포함하며,

상기 하나 이상의 패킷들은 송신 순서에서 상기 제 1 패킷에 후속하는, 비디오 데이터를 디코딩하는 방법.

청구항 20

제 13 항에 있어서,

상기 인코딩된 비디오 비트 스트림 및 상기 인코딩된 비디오 비트 스트림과 연관된 상기 SEI 메시지는 네트워크를 통해 수신되고, 상기 인코딩된 비디오 비트 스트림은 실시간 전송 프로토콜에 따라 캡슐화되며, 상기 SEI 메시지는 세션 디스트립션에 포함되고, 상기 세션 디스크립션은 세션 디스크립션 프로토콜에 따라 포맷되는, 비디오 데이터를 디코딩하는 방법.

청구항 21

디코딩 디바이스로서,

인코딩된 비디오 비트 스트림 및 상기 인코딩된 비디오 비트 스트림과 연관된 상기 SEI 메시지를 저장하도록 구성된 메모리; 및

프로세서를 포함하고,

상기 프로세서는,

상기 인코딩된 비디오 비트 스트림의 인코딩 디바이스에 의해, 상기 인코딩된 비디오 비트 스트림의 코딩된 비디오 시퀀스 (CVS) 내의 SEI 메시지들의 유형들의 세트가 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 나타내는 정보를 포함하는 상기 SEI 메시지를 프로세스하는 것으로서, 상기 SEI 메시지들의 유형들의 세트는 상기 인코딩된 비디오 비트 스트림의 디코딩에 영향을 미치지 않는, 상기 정보를 포함하는 상기 SEI 메시지를 프로세스하고;

상기 SEI 메시지 내에서, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 나타내는 제 1 신택스 엘리먼트를 수신하고;

상기 제 1 신택스 엘리먼트에 기초하여, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 결정하고;

상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지의 각 유형에 대해, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수의 결정에 대응하여, SEI 메시지의 유형을 나타내는 제 2 신택스 엘리먼트를 수신하고;

상기 인코딩된 디바이스에 의해, 상기 SEI 메시지들의 유형들의 세트 중 각 SEI 메시지의 유형이 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 결정하고; 및

디코딩된 비디오 데이터를 생성하기 위해 상기 인코딩된 비디오 비트 스트림을 디코딩하도록 구성되는, 디코딩

디바이스.

청구항 22

제 21 항에 있어서,

상기 CVS의 제 1 액세스 유닛에서 상기 SEI 메시지를 수신하도록 더 구성되는, 디코딩 디바이스.

청구항 23

제 21 항에 있어서,

현재의 액세스 유닛 및 상기 현재의 액세스 유닛에서 상기 CVS의 종단까지 각 액세스 유닛에서 상기 SEI 메시지를 수신하도록 더 구성되는, 디코딩 디바이스.

청구항 24

제 21 항에 있어서,

상기 SEI 메시지의 컨텐츠에 기초하여, 상기 디코딩 디바이스가 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS를 프로세스하기 위해 구성되는 시스템 레벨 프로세스 엔티티를 포함하는지를 결정하도록 더 구성되는, 디코딩 디바이스.

청구항 25

명령들을 포함하는 비일시적 컴퓨터 판독가능 저장 매체로서,

상기 명령들은, 하나 이상의 프로세서들에 의해 실행될 때, 상기 하나 이상의 프로세서들로 하여금,

인코딩된 비디오 비트 스트림 및 상기 인코딩된 비디오 비트 스트림과 연관된 보충 강화 정보 (SEI) 메시지를 수신하게 하고;

상기 인코딩된 비디오 비트 스트림의 인코딩 디바이스에 의해, 상기 인코딩된 비디오 비트 스트림의 코딩된 비디오 시퀀스 (CVS) 내의 SEI 메시지들의 유형들의 세트가 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 나타내는 정보를 포함하는 상기 SEI 메시지를 프로세스하게 하는 것으로서, 상기 SEI 메시지들의 유형들의 세트는 상기 인코딩된 비디오 비트 스트림의 디코딩에 영향을 미치지 않는, 상기 정보를 포함하는 상기 SEI 메시지를 프로세스하게 하고;

상기 SEI 메시지 내에서, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 나타내는 제 1 신택스 엘리먼트를 수신하게 하고;

상기 제 1 신택스 엘리먼트에 기초하여, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 결정하게 하고;

상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지의 각 유형에 대해, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수의 결정에 대응하여, SEI 메시지의 유형을 나타내는 제 2 신택스 엘리먼트를 수신하게 하고;

상기 인코딩된 디바이스에 의해, 상기 SEI 메시지들의 유형들의 세트 중 각 SEI 메시지의 유형이 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 결정하게 하고;

디코딩된 비디오 데이터를 생성하기 위해 상기 인코딩된 비디오 비트 스트림을 디코딩하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 26

장치로서,

인코딩된 비디오 비트 스트림, 보충 강화 정보 (SEI) 메시지, 및 상기 인코딩된 비디오 비트 스트림과 연관된 정보를 수신하는 수단;

상기 인코딩된 비디오 비트 스트림의 인코딩 디바이스에 의해, 상기 인코딩된 비디오 비트 스트림의 코딩된 비

디오 시퀀스 (CVS) 내의 SEI 메시지들의 유형들의 세트가 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 나타내는 정보를 포함하는 상기 SEI 메시지를 프로세스하는 수단으로서, 상기 SEI 메시지들의 유형들의 세트는 상기 인코딩된 비디오 비트 스트림의 디코딩에 영향을 미치지 않는, 상기 정보를 포함하는 상기 SEI 메시지를 프로세스하는 수단;

상기 SEI 메시지 내에서, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 나타내는 제 1 신택스 엘리먼트를 수신하는 수단;

상기 제 1 신택스 엘리먼트에 기초하여, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수를 결정하는 수단;

상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지의 각 유형에 대해, 상기 SEI 메시지들의 유형들의 세트 내에 포함된 SEI 메시지들의 유형들의 수의 결정에 대응하여, SEI 메시지의 유형을 나타내는 제 2 신택스 엘리먼트를 수신하는 수단;

상기 인코딩된 디바이스에 의해, 상기 SEI 메시지들의 유형들의 세트 중 각 SEI 메시지의 유형이 바람직한 사용자 경험을 가능하게 하기 위해 상기 CVS의 컨텐츠를 프로세스하기 위해 필수적인 것으로 결정됨을 결정하는 수단; 및

디코딩된 비디오 데이터를 생성하기 위해 상기 인코딩된 비디오 비트 스트림을 디코딩하는 수단을 포함하는, 장치.

청구항 27

삭제

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

청구항 31

삭제

청구항 32

삭제

발명의 설명

기 술 분 야

[0001] 본 출원은 일반적으로 비디오 코딩에 관한 것으로서, 예를 들어, 보충 강화 정보의 인코딩 및 디코딩을 위한 시스템 및 방법이 기술된다.

배 경 기 술

[0002] 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, ITU-T H.264 또는 ISO/IEC MPEG-4 AVC (그 스케일러블 비디오 코딩 (SVC) 및 멀티-뷰 비디오 코딩 (MVC) 확장물들을 포함함), 및 또한 ITU-T H.265 및 ISO/IEC 23008-2 로 알려진 고효율 비디오 코딩 (HEVC) (그 스케일러블 코딩 확장물 (즉, 스케일러블 고효율 비디오 코딩, SHVC) 및 멀티뷰 확장물

(즉, 멀티뷰 고효율 비디오 코딩, MV-HEVC) 를 포함함) 을 포함한다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0003] 여러 구현들에서, SEI 메시지에서 필수 정보를 시그널링하기 위한 기법들이 제공된다. 정보는 인코더가 그 정보가 비디오 데이터를 디스플레이하기 위해 요구된다고 결정하는 경우 필수적이다. 그 정보는 비디오 데이터를 디코딩하기 위해 요구되지 않는다. 비디오 데이터를 디스플레이 또는 제시하는 것과 관련된 정보는 SEI 메시지에서 제공될 수 있지만, SEI 메시지는 각 메시지에서 반송되는 데이터가 비디오 데이터의 제시에 필수적인지 여부를 표시할 수 없을 수도 있다. 추가적으로, SEI 메시지를 사용하는 코딩 표준은 디코딩 디바이스가 필수 정보를 포함하는 SEI 메시지를 빠르게 식별 및 로케이팅하는 방법을 제공하지 않을 수도 있다.

[0004] 여러 구현들에서, 필수 정보가 인코딩된 비디오 비트스트림에 포함된다고 표시할 수 있는 SEI 메시지가 제공된다. 이들 SEI 메시지는 디코딩 디바이스가 필수 정보가 제공된다고 결정하고, 그 필수 정보를 로케이팅하는 것을 가능하게 한다.

[0005] 적어도 하나의 예에 따르면, 비디오 데이터를 수신하는 단계를 포함하는 비디오 데이터를 인코딩하는 방법이 제공된다. 방법은 인코딩 디바이스에 의해, 비디오 데이터와 연관된 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하는 단계를 더 포함하고, 여기서 그 정보는 비디오 데이터를 디코딩하기 위해 요구되지 않는다. 방법은 보충 강화 정보 (SEI) 메시지를 생성하는 단계를 더 포함하고, 여기서 SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 나타내는 필드를 포함한다. 방법은 비디오 데이터를 인코딩하는 단계를 더 포함한다. 방법은 인코딩된 비디오 데이터와 함께 SEI 메시지를 포함시키는 단계를 더 포함한다.

[0006] 다른 예에서, 비디오 데이터를 저장하도록 구성된 메모리 및 프로세서를 포함하는 인코딩 디바이스가 제공된다. 프로세서는 비디오 데이터와 연관된 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하도록 구성되고 결정할 수 있고, 여기서 그 정보는 비디오 데이터를 디코딩하기 위해 요구되지 않는다. 프로세서는 보충 강화 정보 (SEI) 메시지를 생성하도록 구성되고 생성할 수 있고, 여기서 SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 나타내는 필드를 포함한다. 프로세서는 비디오 데이터를 인코딩하도록 구성되고 인코딩할 수 있다. 프로세서는 인코딩된 비디오 데이터와 함께 SEI 메시지를 포함시키도록 구성되고 포함시킬 수 있다.

[0007] 다른 예에서, 하나 이상의 프로세서들에 의해 실행될 때, 하나 이상의 프로세서들로 하여금 비디오 데이터를 수신하게 하는 명령들을 포함하는 비일시적 컴퓨터 판독가능 매체가 제공된다. 명령들은 또한 하나 이상의 프로세서들로 하여금 비디오 데이터와 연관된 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하게 하며, 여기서 그 정보는 비디오 데이터를 디코딩하기 위해 요구되지 않는다. 명령들은 또한 하나 이상의 프로세서들로 하여금 보충 강화 정보 (SEI) 메시지를 생성하게 하며, 여기서 SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 나타내는 필드를 포함한다. 명령들은 또한 하나 이상의 프로세서들로 하여금 비디오 데이터를 인코딩하게 한다. 명령들은 또한 하나 이상의 프로세서들로 하여금 인코딩된 비디오 데이터와 함께 SEI 메시지를 포함시키게 한다.

[0008] 다른 예에서, 비디오 데이터를 수신하는 수단을 포함하는 장치가 제공된다. 장치는 비디오 데이터와 연관된 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하는 수단을 더 포함하고, 여기서 그 정보는 비디오 데이터를 디코딩하기 위해 요구되지 않는다. 장치는 보충 강화 정보 (SEI) 메시지를 생성하는 수단을 더 포함하고, 여기서 SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 나타내는 필드를 포함한다. 장치는 비디오 데이터를 인코딩하는 수단을 더 포함한다. 장치는 인코딩된 비디오 데이터와 함께 SEI 메시지를 포함시키는 수단을 더 포함한다.

[0009] 일부 양태들에서, 인코딩 디바이스는 인코딩 디바이스가 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하기 위해 사용할 수 있는 하나 이상의 파라미터로 구성된다.

[0010] 일부 양태들에서, 인코딩 디바이스는 정보가 비디오 데이터를 제시하기 위해 요구된다고 결정한다.

[0011] 일부 양태들에서, 인코딩 디바이스는 정보가 비디오 데이터를 제시하기 위해 요구되지 않는다고 결정한다.

[0012] 일부 양태들에서, SEI 메시지의 신택스 엘리먼트는 SEI 메시지의 유형을 표시한다.

[0013] 일부 양태들에서, 정보는 인코딩된 비디오 데이터와 함께 포함된다.

[0014] 일부 양태들에서, 상술된 방법, 장치, 및 컴퓨터 판독가능 매체는 파일에 인코딩된 비디오 데이터 및 SEI 메시지를 기입하는 것을 더 포함하고, 여기서 인코딩된 비디오 데이터 및 SEI 메시지는 파일 포맷에 따라 파일에 기입된다. 양태들은 인코딩된 비디오 데이터가 판독되기 전에 SEI 가 판독될 수 있는 파일 내에 SEI 메시지를 배치하는 파일 포맷을 사용하는 것을 더 포함한다.

[0015] 일부 양태들에서, 상술된 방법, 장치, 및 컴퓨터 판독가능 매체는 스트리밍을 위해 인코딩된 비디오 데이터를 캡슐화하는 것을 더 포함하고, 여기서 인코딩된 비디오 데이터는 스트리밍 포맷에 따라 캡슐화된다. 이들 양태들은 스트리밍 포맷에 따라 인코딩된 비디오 데이터의 디스크립션을 생성하는 것을 더 포함하고, 여기서 그 디스크립션은 SEI 메시지를 포함한다.

[0016] 일부 양태들에서, 상술된 방법, 장치, 및 컴퓨터 판독가능 매체는 네트워크를 통한 송신을 위해 인코딩된 비디오 데이터를 캡슐화하는 것을 더 포함하고, 여기서 인코딩된 비디오 데이터는 컨테이너 포맷에 따라 캡슐화된다. 이들 양태들은 컨테이너 포맷에 따라 제 1 패킷을 생성하는 것을 더 포함하고, 여기서 제 1 패킷의 페이로드 부분은 디스크립터로서 SEI 메시지를 포함한다. 이들 양태들은 인코딩된 비디오 데이터를 포함하는 하나 이상의 패킷을 생성하는 것을 더 포함하고, 여기서 하나 이상의 패킷은 송신 순서에서 제 1 패킷에 후속한다.

[0017] 일부 양태들에서, 상술된 방법, 장치, 및 컴퓨터 판독가능 매체는 네트워크를 통한 송신을 위해 인코딩된 비디오 데이터를 캡슐화하는 것을 더 포함하고, 여기서 인코딩된 비디오 데이터는 실시간 전송 프로토콜에 따라 캡슐화된다. 이들 양태들은 세션 디스크립션 프로토콜에 따라 세션 디스크립션을 생성하는 것을 더 포함하고, 여기서 세션 디스크립션은 SEI 메시지를 포함한다.

[0018] 적어도 하나의 예에 따르면, 디코딩 디바이스에서, 인코딩된 비디오 비트스트림, 보충 강화 정보 (SEI) 메시지, 및 인코딩된 비디오 비트스트림과 연관된 정보를 수신하는 단계를 포함하는, 비디오 데이터를 디코딩하는 방법이 제공된다. 방법은 SEI 메시지를 사용하여, 정보가 인코딩된 비디오 비트스트림에서의 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하는 단계를 더 포함하고, 여기서 그 정보는 비디오 데이터를 디코딩하기 위해 요구되지 않고, 여기서 SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 나타내는 필드를 포함한다. 방법은 비디오 데이터를 생성하기 위해 인코딩된 비디오 비트스트림을 디코딩하는 단계를 더 포함하고, 여기서 인코딩된 비디오 비트스트림은 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하는 것의 결과에 따라 디코딩된다.

[0019] 다른 예에서, 인코딩된 비디오 데이터, 보충 강화 정보 (SEI) 메시지, 및 인코딩된 비디오 비트스트림과 연관된 정보를 저장하도록 구성된 메모리, 및 프로세서를 포함하는 장치가 제공된다. 프로세서는 SEI 메시지를 사용하여, 정보가 인코딩된 비디오 비트스트림에서의 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하도록 구성되고 결정할 수 있으며, 여기서 그 정보는 비디오 데이터를 디코딩하기 위해 요구되지 않고, 여기서 SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 나타내는 필드를 포함한다. 프로세서는 비디오 데이터를 생성하기 위해 인코딩된 비디오 비트스트림을 디코딩하도록 구성되고 디코딩할 수 있으며, 여기서 인코딩된 비디오 비트스트림은 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하는 것의 결과에 따라 디코딩된다.

[0020] 다른 예에서, 하나 이상의 프로세서들에 의해 실행될 때, 하나 이상의 프로세서들로 하여금, 인코딩된 비디오 비트스트림, 보충 강화 정보 (SEI) 메시지, 및 인코딩된 비디오 비트스트림과 연관된 정보를 수신하게 하는 명령들을 포함하는 비일시적 컴퓨터 판독가능 매체가 제공된다. 명령들은 또한 하나 이상의 프로세서들로 하여금 SEI 메시지를 사용하여, 정보가 인코딩된 비디오 비트스트림에서의 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하게 하며, 여기서 그 정보는 비디오 데이터를 디코딩하기 위해 요구되지 않고, 여기서 SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 나타내는 필드를 포함한다. 명령들은 또한 하나 이상의 프로세서들로 하여금 비디오 데이터를 생성하기 위해 인코딩된 비디오 비트스트림을 디코딩하게 하며, 여기서 인코딩된 비디오 비트스트림은 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하는 것의 결과에 따라 디코딩된다.

[0021] 다른 예에서, 인코딩된 비디오 비트스트림, 보충 강화 정보 (SEI) 메시지, 및 인코딩된 비디오 비트스트림과 연관된 정보를 수신하는 수단을 포함하는 장치가 제공된다. 장치는 SEI 메시지를 사용하여, 정보가 인코딩된 비디오 비트스트림에서의 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하는 수단을 더 포함하고, 여기서 그 정보는 비디오 데이터를 디코딩하기 위해 요구되지 않고, 여기서 SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 나타내는 필드를 포함한다. 장치는 비디오 데이터를 생성하기 위해 인코딩된 비디오 비트스트림을 디코딩하는 수단을 더 포함하고, 여기서 인코딩된 비디오 비트스트림은 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 결정하는 것의 결과에 따라 디코딩된다.

[0022] 일부 양태들에서, 디코딩 디바이스는 정보가 비디오 데이터를 제시하기 위해 요구된다고 결정하고, 디코딩 디바이스는 그 정보에 따라 비디오 데이터를 구성한다.

[0023] 일부 양태들에서, 디코딩 디바이스는 정보가 비디오 데이터를 제시하기 위해 요구되지 않는다고 결정하고, 디코딩 디바이스는 그 정보를 사용하지 않고 인코딩 비디오 데이터를 디코딩한다.

[0024] 일부 양태들에서, SEI 메시지의 신택스 엘리먼트는 SEI 메시지의 유형을 표시한다.

[0025] 일부 양태들에서, 인코딩된 비디오 비트스트림, SEI 메시지, 및 인코딩된 비디오 비트스트림과 연관된 정보는 파일에서 수신되고, 여기서 파일은 파일 포맷에 따라 포맷팅되며, 파일 포맷에 따라, SEI 메시지가 인코딩된 비디오 비트스트림을 포함하는 파일의 부분 앞의 디코딩 디바이스에 의해 판독되는 파일의 부분에 존재한다.

[0026] 일부 양태들에서, 인코딩된 비디오 비트스트림, SEI 메시지, 및 인코딩된 비디오 비트스트림과 연관된 정보는 데이터 스트림에서 수신되고, 여기서 데이터 스트림은 스트리밍 포맷에 따라 포맷팅된다. 이들 양태들은 데이터 스트림으로터, 인코딩된 비디오 데이터의 디스크립션을 판독하는 것을 더 포함하고, 여기서 디스크립션은 SEI 메시지를 포함한다.

[0027] 일부 양태들에서, 인코딩된 비디오 비트스트림, SEI 메시지, 및 인코딩된 비디오 비트스트림과 연관된 정보는 복수의 네트워크 패킷들에서 수신된다. 이들 양태들은 복수의 네트워크 패킷들로부터의 제 1 패킷의 페이로드 부분으로부터 SEI 메시지를 판독하는 것을 더 포함한다. 이들 양태들은 복수의 네트워크 패킷들로부터의 하나 이상의 패킷들로부터 인코딩된 비디오 데이터를 판독하는 것을 더 포함하며, 여기서 하나 이상의 패킷들은 송신 순서에서 제 1 패킷에 후속한다.

[0028] 일부 양태들에서, 인코딩된 비디오 비트스트림, SEI 메시지, 및 인코딩된 비디오 비트스트림과 연관된 정보는 네트워크를 통해 수신되고, 여기서 인코딩된 비디오 비트스트림은 실시간 전송 프로토콜에 따라 캡슐화되며, 여기서 SEI 메시지는 세션 디스크립션에 포함되고, 세션 디스크립션은 세션 디스크립션 프로토콜에 따라 포맷팅된다.

도면의 간단한 설명

[0029] 본 출원의 예시적인 예들이 다음의 도면을 참조하여 이하에 상세히 기술된다.

도 1 은 비디오 코딩 시스템의 예를 도시하는 블록도이다.

도 2 는 액세스 유닛의 예를 도시한다.

도 3 은 ISO 베이스 미디어 파일의 예를 도시한다.

도 4 는 스트리밍 비디오 컨텐츠에 대한 DASH 제시의 예를 도시한다.

도 5 는 MPEG 전송 스트림의 예를 도시한다.

도 6 은 네트워크 통신을 위한 개방 시스템 인터커넥트 (OSI) 모델 600 의 예를 도시한다.

도 7 은 비디오 데이터를 인코딩하기 위한 프로세스의 예를 도시한다.

도 8 은 비디오 데이터를 디코딩하기 위한 프로세스의 예를 도시한다.

도 9 는 예시의 인코딩 디바이스를 도시하는 블록도이다.

도 10 은 예시의 디코딩 디바이스를 도시하는 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0030] 본 개시의 특정 양태들 및 예들이 이하에 제공된다. 이들 양태들 및 예들 중 일부는 독립적으로 적용될 수 있고 그들 중 일부는 당업자에게 자명한 바와 같이 조합하여 적용될 수도 있다. 다음의 설명에서, 설명의 목적으로, 다양한 가능한 구현들의 완전한 이해를 제공하기 위해 다수의 특정 세부 사항들이 제시된다. 하지만, 여러 예들은 이들 특정 상세들없이도 실시될 수도 있음이 명백할 것이다. 도면 및 설명은 제한하려는 것이 아니다.

[0031] 다음의 설명은 단지 예시적인 구현들을 제공하며, 본 개시의 범위, 적용가능성 또는 구성을 제한하려도록 의도되지 않는다. 다만, 예시적 구현들의 다음의 설명은 일 예를 구현하기 위한 실시가능한 설명을 당업자에게 제공할 것이다. 첨부된 청구범위에 설명된 바와 같이 본 출원의 사상 및 범위를 벗어나지 않으면서 요소들의 기능 및 배열에 다양한 변경이 가해질 수 있음을 이해해야한다.

[0032] 특정 상세들이 예들의 철저한 이해를 제공하기 위해 다음의 설명에서 주어진다. 그러나, 예들은 이러한 특정 세부사항들이 없이 실시될 수도 있음이 당업자에 의해 이해될 것이다. 예를 들면, 회로, 시스템, 네트워크, 프로세스 및 다른 컴포넌트들은, 실시형태들을 불필요한 상세들에서 불분명하게 하지 않기 위하여 블록도 형태의 컴포넌트들로서 보여질 수도 있다. 다른 예들에서, 잘 알려진 회로, 프로세스들, 알고리즘, 구조, 및 기술들은 실시형태들을 불분명하게 하는 것을 피하기 위하여 불필요한 상세 없이 도시될 수도 있다.

[0033] 개개의 예들은, 플로우차트, 플로우 다이어그램, 데이터 흐름도, 구조 다이어그램, 또는 블록 다이어그램으로서 도시된 프로세스로서 설명될 수도 있음이 주목된다. 비록 플로우차트가 동작들을 순차적인 프로세스로서 기술할 수도 있지만, 동작들 중 다수는 병렬로 또는 동시에 수행될 수 있다. 부가적으로, 동작들의 순서가 재배열될 수도 있다. 프로세스는 그의 동작들이 완료되면 종료되지만 도면에 포함되지 않은 추가 단계들을 가질 수 있다. 프로세스는 방법, 함수, 절차, 서브루틴, 서브프로그램 등에 대응할 수도 있다. 프로세스가 함수에 대응할 경우, 그 종료는 그 함수의 호출 함수 또는 메인 함수로의 복귀에 대응할 수 있다.

[0034] 용어 "컴퓨터 판독가능 매체" 는 휴대용 또는 고정식 저장 디바이스들, 광학 저장 디바이스들, 및 명령(들) 및/또는 데이터를 저장, 포함, 또는 반송할 수 있는 다양한 다른 매체들을 포함하지만 이에 한정되지 않는다. 컴퓨터 판독 가능 매체는 데이터가 저장될 수 있고 캐리어 웨이브 및/또는 무선 또는 유선 연결을 통해 전파되는 일시적인 전자 신호를 포함하지 않는 비일시적인 매체를 포함할 수도 있다. 비일시적인 매체의 예는 자기 디스크 또는 테이프, 콤팩트 디스크 (CD) 또는 디지털 다용도 디스크 (DVD)와 같은 광학 저장 매체, 플래시 메모리, 메모리 또는 메모리 장치를 포함하나, 이에 한정되는 것은 아니다. 컴퓨터 판독가능 매체는 절차, 함수, 서브프로그램, 프로그램, 루틴, 서브루틴, 모듈, 소프트웨어 패키지, 클래스, 또는 명령들, 데이터 구조들, 또는 프로그램 스테이트먼트들의 임의의 조합을 나타낼 수도 있는 코드 및/또는 머신 실행가능 명령들을 저장할 수도 있다. 코드 세그먼트는, 정보, 데이터, 인수들 (arguments), 파라미터들, 또는 메모리 콘텐츠를 전달 및/또는 수신함으로써 다른 코드 세그먼트 또는 하드웨어 회로에 커플링될 수도 있다. 정보, 인수들, 파라미터들, 데이터 등은 메모리 공유, 메시지 전달, 토큰 전달, 네트워크 전송 등을 포함한 임의의 적합한 수단을 통해 전달, 포워딩, 또는 전송될 수도 있다.

[0035] 또한, 여기서 논의된 예들은, 하드웨어, 소프트웨어, 펌웨어, 미들웨어, 마이크로코드, 하드웨어 기술 언어, 또는 이들의 임의의 조합에 의해 구현될 수도 있다. 소프트웨어, 펌웨어, 미들웨어, 또는 마이크로코드로 구현될 경우, 필요한 태스크들을 수행하기 위한 프로그램 코드 또는 코드 세그먼트들 (예를 들어, 컴퓨터 프로그램 제품) 은 컴퓨터 판독가능 또는 머신 판독가능 매체에 저장될 수도 있다. 프로세서(들) 은 필요한 작업들을 수행할 수도 있다.

[0036] 더 많은 디바이스들 및 시스템들이 소비자에게 디지털 비디오 데이터를 소비 할 능력을 제공함에 따라, 효율적인 비디오 코딩 기술에 대한 필요가 더 중요 해지고 있다. 비디오 코딩은 디지털 비디오 데이터에 존재하는 다량의 데이터를 처리하는 데 필요한 저장 및 전송 요건들을 줄이기 위해 필요하다. 다양한 비디오 코딩 기술은 높은 비디오 품질을 유지하면서 더 낮은 비트 레이트를 사용하는 형태로 비디오 데이터를 압축하는데 사용될 수도 있다. 본 명세서에서 사용되는 바와 같이, "코딩"은 "인코딩" 및 "디코딩"을 지칭한다.

[0037] 인코딩된 비트스트림에서의 보충 강화 정보 (SEI) 메시지는 인코딩된 비트 스트림의 디코딩 및/또는 디스플레이를 보조하거나 다른 목적으로 사용될 수있다. SEI 메시지는 디코딩 프로세스 동안 루마 또는 크로마 샘플을 구성하기 위해 SEI 메시지의 데이터가 필요하지 않다는 점에서 "보충적"이다. 또한, SEI 메시지를 지원하는 비디오 코딩 표준을 따르는 디코더는 컨포밍 (conforming) 하기 위해 SEI 메시지를 처리할 필요가 없다. 일부 코딩 표준의 경우, 비트 스트림 적합성을 확인하거나 타이밍 디코더 적합성을 출력하기 위해 일부 SEI 메시지 정보가 필요할 수있다.

[0038] SEI 메시지는 인코딩된 비트 스트림에 대한 추가 정보를 제공하는데 사용될 수 있으며, 이는 일단 비트 스트림이 디코딩되면 비트 스트림의 제시를 변경하거나 디코더에 정보를 제공하는데 사용될 수있다. 예를 들어, SEI 메시지는 무엇보다도 프레임 패킹 정보 (예를 들어, 비디오 데이터가 비디오 프레임에 배열되는 방식을 기술함), 컨텐츠 디스크립션 (예를 들어, 인코딩 된 비트 스트림이 예를 들어 360 도 비디오임을 나타 내기 위해), 컬러 정보 (예를 들어, 색 영역 및/또는 색 범위) 을 제공하는데 사용되어 왔다.

[0039] 비디오 코딩 표준이 SEI 메시지가 인코딩 된 비트 스트림을 디코딩하는데 필요하지 않다고 제공하더라도, 일부 경우에, SEI 메시지로 반송되는 정보는 디코딩 프로세싱에 필수적 일 수있다. 예를 들어, SEI 메시지는 비트 스트림으로 인코딩 된 비디오 데이터의 포맷을 설명하기 위해 사용될 수 있으며, 그것이 없이는 디코더는 의도 된 대로 비디오를 제시하지 못할 수도 있다. 예를 들어, SEI 메시지는 인코딩 된 비트 스트림이 360도 비디오를 포함한다는 것을 디코더에 시그널링하기 위해 사용될 수있다. 이 예에서, 디코더는 360도 제시를 위한 비디오 데이터를 렌더링하기 위해 이 정보를 사용할 수있다. 대안적으로, 디코더가 360도 비디오를 렌더링 할 수 없다면, 디코더는 비디오 데이터를 렌더링하려고 시도하고 심하게 왜곡 된 비디오 시퀀스를 제시하기보다는 이 정보를 사용하여 비디오 데이터를 렌더링하지 않을 수있다. 다른 예로서, 컨텐츠 생성기는 인코딩 된 비디오 데이터가 특정 색 영역으로 렌더링되도록 특정했을 수있다. 이 예에서, SEI 메시지에 데이터가 없는 경우, 디코더는 비디오를 다른 색 영역으로 렌더링하여 의도 된 것과 다른 방식으로 비디오 데이터를 제시할 수도 있다.

[0040] 다양한 구현에서, "필수"정보와 "비 필수"정보를 구별하는 SEI 메시지에 대한 기술 및 이러한 기술을 구현하는 시스템이 제공된다. 다양한 구현에서, 비디오 데이터와 관련된 정보가 필수적인지 아닌지를 결정하도록 인코더가 구성 될 수있다. 예를 들어, 필수 정보를 식별하는 매개 변수로 인코더를 구성 할 수 있다. 파라미터는 특정 비디오에 대해 프레임 패킹 배열 또는 컬러 볼륨과 같은 정보 또는 다른 정보가 특정 비디오에 필수적임을 나타낼 수있다.

[0041] 다양한 구현에서, 상이한 유형의 SEI 메시지가 정의 될 수 있으며, 여기서 이들 유형의 SEI 메시지 중 하나 이상은 특정 비디오와 관련된 정보가 예를 들어 인코딩 장치에 의해 결정된 바와 같이 비디오 데이터를 제시 (예를 들어, 인코딩 디바이스에 의해 의도된 대로 제시) 하기 위해, 필수적인지 아닌지를 나타낼 수있다. 본 명세서에 제공된 바와 같이, 용어 "필수"및 "비 필수"는 특정 방식으로 비디오의 제시에 필요하지 않은 정보로부터의 그 특정 방식으로 비디오를 제시 (예를 들어, 의도된 제시 또는 렌더링) 하기 위해 요구되는 것으로 (예를 들어, 인코딩 디바이스 및/또는 컨텐츠 제공자/생성기에 의해) 간주된 정보를 구별하기 위해 사용된다. 정보가 필수적인지 아닌지 여부는 컨텐츠 생성기에 의해 정의 (예를 들어, 결정 또는 설정) 될 수있다. 예를 들어, 일부 비디오 데이터의 경우, 프레임 패킹 정보 또는 색 변환 정보는 컨텐츠 생성기에 의해 필수적인 것으로 간주 될 수 있는 반면, 계층 정보는 필수적이지 않을 수도 있다. 다른 예로서, 일부 비디오 데이터에 대해, 필터링 정보는 필수적 일 수 있지만, 타이밍 정보는 필수적이지 않다. 비트 스트림에서 비디오를 디코딩하는 데 필수 및 비 필수 정보가 필요하지 않다.

[0042] 일례에서, 메니페스트 SEI 메시지 및 프리픽스 SEI 메시지가 제공된다. 이 예에서, 매니페스트 SEI 메시지는 비트 스트림에 존재하거나 존재하지 않는 SEI 메시지의 유형들의 리스트 및 비트 스트림 내의 임의의 SEI 메시지가 필수 정보를 포함하는지 여부를 포함한다. 또한 이 예에서, 프리픽스 SEI 메시지는 프리픽스 SEI 메시지에서 반송되는 페이로드의 유형을 나타낼 수있는 프리픽스 표시를 포함 할 수있다. 페이로드는 필수적이거나 필수적이지 않을 수도 있는 정보를 포함할 수 있다.

[0043] 다른 예에서, SEI 메시지는 필수 정보에 대해 정의되며, 이것은 필수 보충 정보 (ESI) SEI 메시지로 지칭되고, SEI 메시지는 비 필수 정보에 대해 정의되며, 이는 비 필수 보충 정보 (NSI) SEI 메시지로 지칭된다. ESI SEI 및 NSI SEI 를 사용하여, 비디오 코딩 표준은 이들 SEI 메시지 유형을 지원하는 디코더에 의해 우선 순위가 정해질 수 있는 2 가지 상이한 종류의 SEI 메시지를 지원할 수 있다.

[0044] 다양한 예시적인 비디오 코딩 표준에서, SEI 메시지는 인코딩 된 비트 스트림의 특정 부분에 배치 될 수 있지만, 임의의 특정 지점에 배치될 필요는 없다. 예를 들어, SEI 메시지는 인코딩 된 비트 스트림에서 초기에 배치 될 필요는 없다. 이러한 예에서, 디코더가 비디오 데이터를 인코딩 된 비트 스트림으로 렌더링하기 시작하기 전에 필수 정보를 포함하는 SEI 메시지가 디코더에 의해 보이지 않을 수도 있다.

[0045] 다양한 구현에서, SEI 메시지가 디코더에 의해 조기에 보여 질 수 있도록 캡슐화 포맷을 사용하기 위한 기법들, 및 이들 기법들을 구현하는 시스템이 제공된다. 캡슐화 형식에는 무엇보다도 예를 들어 ISOBMFF (Organization for Standardization (ISO) base media file format), SDP (Session Description Protocol), DASH (Dynamic Adaptive Streaming over Hypertext Transfer Protocol) 및 MPEG-TS (Motion Picture Experts Group

Transport Stream) 등이 있다. 다양한 구현들에서, 이들 캡슐화 포맷들에 의해 정의 된 포맷들은 SEI 메시지들을 배치하기 위해 사용될 수 있으며, 여기서 디코더는 비디오 데이터를 디코딩 및 판독하는 프로세스의 초기에 SEI 메시지들을 발견할 수 있다. 따라서, 필수 정보에 대한 SEI 메시지를 지원하는 디코더는 화상의 디코딩을 시작하기 전에 이들 메시지에 의해 제공된 정보를 얻을 수있다.

[0046] 필수 또는 비 필수 정보를 나타낼 수 있는 SEI 메시지의 사용자는 디코더의 동작을 향상시킬 수 있다. 예를 들어, 필수 정보를 플래깅할 수 있는 SEI 메시지가 없으면, 일부 디코더는 시스템이 비디오를 디스플레이 할 수 있는지 여부를 먼저 결정하지 않고 비트 스트림을 디코딩 및 렌더링하려고 시도할 수도 있다. 이 예에서, 결과는 디스플레이 될 때 비디오가 심하게 왜곡되거나 다르게는 컨텐츠 생성기에서 의도 된 방식으로 제시되지 않는 결과일 수 있다.

[0047] 일부 예들에서, 디코더는 비디오를 디코딩하기 시작한 후, 예를 들어 비트 스트림에 포함 된 SEI 메시지에서 필수 정보를 만날 수도 있다. 이러한 예에서, 일부 디코더는 출력을 보정하고 의도 한대로 비디오를 디스플레이하기 시작할 수 있다. 그러나 일부 디코더는 SEI 메시지로부터 시스템이 비디오를 지원하지 않고 비디오를 디스플레이 할 수 없다고 결정할 수있다. 이러한 경우, 시스템은 비디오를 디코딩 및 렌더링하는 것을 중지할지 여부를 결정하거나 시스템이 디스플레이할 수 있는 비디오의 버전을 찾아야 할 수도 있다. 시스템이 시스템이 디스플레이 할 수 있는 비디오의 버전을 찾을 수 있다면, 시스템은 비디오를 처음부터 디코딩을 시작할지, 또는 뷰어가 본 마지막 시점 근처에서 시작하기 위해 검색 동작을 수행할지를 결정해야 할 수도 있다. 어느 경우 든, 디코더는 시작하는데 처리 시간을 소비한다. 또한 적절하게 디스플레이될 수 없는 비디오를 디코딩하는 데 소요된 처리 시간이 낭비된다.

[0048] 필수 정보를 얻는 문제에 대한 한 가지 해결책은 디코더가 필수 정보를 위해 인코딩 된 비트 스트림을 검색하는 것이다. 그러나, 이 솔루션은 디코더가 인코딩 된 비트 스트림을 버퍼링하기 위해 상당한 양의 메모리를 요구할 수 있고, 시스템이 비디오를 재생하기 시작할 수 있기 전에 상당한 지연을 야기 할 수도 있다.

[0049] 상기 각각의 예에서, 디코더가 필수 정보를 신속하게 식별하고 찾을 수 있게 하는 방식으로 필수 정보를 제공함으로써, 디코더는 필수 정보를 검색 할 필요가 없고, 시스템이 디스플레이할 수 없는 비트 스트림의 디코딩을 피할 수 있으며, 시스템이 의도된 것과 다른 방식으로 디코딩 된 비트 스트림을 디스플레이하는 것을 피할 수 있다. 디코더는 또한 아마도 더 적은 시간 낭비로, 더욱 효율적으로 작동될 수 있다.

[0050] 도 1 은 인코딩 디바이스 (104) 및 디코딩 디바이스 (112) 를 포함하는 비디오 코딩 시스템 (100) 의 예를 도시한 블록도이다. 일부 예에서, 비디오 코딩 시스템 (100) 의 인코딩 디바이스 (100)는 ESI SEI 메시지 및/또는 NSI SEI 메시지를 생성 할 수 있고, 디코딩 디바이스 (112) 는 ESI SEI 메시지 및/또는 NSI SEI 메시지를 디코딩하고 이들 메시지들에 포함 된 정보를 이용할 수있다. 인코딩 디바이스 (104) 는 소스 디바이스의 일부일 수 있고, 디코딩 디바이스 (112)는 수신 디바이스의 일부일 수 있다. 소스 디바이스 및/또는 수신 디바이스는 모바일 또는 고정 전화 핸드셋 (예를 들어, 스마트 폰, 셀룰러 폰 등), 데스크탑 컴퓨터, 랩탑 또는 노트북 컴퓨터, 태블릿 컴퓨터, 셋톱 박스, 텔레비전, 카메라, 디스플레이 장치, 디지털 미디어 플레이어, 비디오 게임 콘솔, 비디오 스트리밍 장치, 인터넷 프로토콜 (IP) 카메라, 또는 임의의 다른 적절한 전자 장치와 같은 전자 장치를 포함할 수도 있다. 일부 예에서, 소스 디바이스 및 수신 디바이스는 무선 통신을 위한 하나 이상의 무선 송수신기를 포함할 수도 있다. 여기에 기술된 코딩 기술은 (예를 들면, 인터넷을 통한) 스트리밍 비디오 전송, 텔레비전 방송 또는 전송, 데이터 저장 매체상의 저장을 위한 디지털 비디오의 인코딩, 데이터 저장 매체상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션을 포함하는 다양한 멀티미디어 애플리케이션에서의 비디오 코딩에 적용가능하다. 일부 예들에 있어서, 시스템 (100) 은 비디오 컨퍼런싱, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 게이밍 및/또는 비디오 전화와 같은 애플리케이션들을 지원하기 위해 일방향 또는 양방향 비디오 송신을 지원할 수 있다.

[0051] 인코딩 디바이스 (104) (또는 인코더) 는 비디오 코딩 표준 또는 프로토콜을 사용하여 비디오 데이터를 인코딩하여 인코딩된 비디오 비트 스트림을 생성하는 데 사용될 수 있다. 비디오 코딩 표준들의 예들은, 그 스케일러블 비디오 코딩 (SVC) 및 멀티-뷰 비디오 코딩 (MVC) 확장들, 및 고효율 비디오 코딩 (HEVC) 또는 ITU-T H.265를 포함하여, 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, ITU-T H.264 (ISO/IEC MPEG-4 AVC 로서도 알려짐) 를 포함한다. 범위 및 스크린 콘텐츠 코딩 확장본들, 3D 비디오 코딩(3D-HEVC) 및 멀티뷰 확장본들(MV-HEVC) 그리고 스케일러블 확장본(scalable extension, SHVC)을 포함하는, 다층 비디오 코딩을 다루는 HEVC에 대한 다양한 확장본들이 존재한다. HEVC 및 그것의 확장본들은 JCT-VC(Joint Collaboration Team on Video Coding) 뿐만 아니라 ITU-T 비디오 코딩 전문가

그룹(VCEG) 및 ISO/IEC 모션 픽처 전문가 그룹(MPEG)의 JCT-3V(Joint Collaboration Team on 3D Video Coding Extension Development)에 의해 개발되었다. MPEG 및 ITU-T VCEG는 차세대의 비디오 코딩 표준을 위한 새로운 코딩 도구들을 탐구하기 위해 JVET(joint exploration video team)을 또한 형성했다. 참조 소프트웨어는 JEM(joint exploration model)이라 지칭된다.

[0052] 여기에 기술된 많은 예는 JEM 모델, HEVC 표준 및/또는 그 확장을 사용하는 예를 제공한다. 그러나, 여기서 설명된 기술들 및 시스템들은 또한 AVC, MPEG, 그 확장들, 또는 현재 존재하는 다른 적절한 코딩 표준들 또는 장래의 코딩 표준들과 같은 다른 코딩 표준들에도 적용될 수도 있다. 따라서, 여기에 기술된 기술들 및 시스템들은 특정 비디오 코딩 표준을 참조하여 기술될 수도 있지만, 당업자는 설명이 그러한 특정 표준에만 적용되는 것으로 해석되어서는 안됨을 이해할 것이다.

[0053] 도 2 를 참조하면, 비디오 소스 (102) 는 비디오 데이터를 인코딩 디바이스 (104) 에 제공할 수도 있다. 비디오 소스 (102) 는 소스 디바이스의 일부일 수도 있거나 소스 디바이스 이외의 디바이스의 일부일 수도 있다. 비디오 소스 (102) 는 비디오 캡처 디바이스 (예를 들어, 비디오 카메라, 카메라 폰, 비디오 폰 등), 저장된 비디오를 포함하는 비디오 아카이브, 비디오 데이터를 제공하는 비디오 서버 또는 컨텐츠 제공자, 비디오 서버 또는 컨텐츠 제공자로부터 비디오를 수신하는 비디오 피드 인터페이스, 컴퓨터 그래픽 비디오 데이터를 생성하기 위한 컴퓨터 그래픽 시스템, 그러한 소스들의 조합, 또는 임의의 다른 적합한 비디오 소스를 포함할 수도 있다.

[0054] 비디오 소스 (102) 로부터의 비디오 데이터는 하나 이상의 입력 화상들 또는 프레임들을 포함할 수도 있다. 비디오의 화상 또는 프레임은 장면의 정지 이미지이다. 인코딩 디바이스 (104) 의 인코더 엔진 (106) (또는 인코더) 은 인코딩된 비디오 비트 스트림을 생성하기 위해 비디오 데이터를 인코딩한다. 일부 예에서, 인코딩된 비디오 비트 스트림 (또는 "비디오 비트 스트림" 또는 "비트 스트림") 은 일련의 하나 이상의 코딩된 비디오 시퀀스들이다. 코딩된 비디오 시퀀스 (CVS) 는 특정 특성을 갖는 및 기본 계층에서의 랜덤 액세스 포인트 화상을 갖는 AU 로 시작하여 특정 특성을 갖는 및 기본 계층에서의 랜덤 액세스 포인트 화상을 갖는 다음 AU 까지의 및 다음 AU 를 포함하지 않는 일련의 액세스 유닛들 (Aus) 을 포함한다. 예를 들어, CVS 를 시작하는 랜덤 액세스 포인트 화상의 특정 속성은 1과 같은 RASL 플래그 (예: NoRaslOutputFlag) 를 포함할 수도 있다. 그렇지 않으면, (0 과 동일한 RASL 플래그를 갖는) 랜덤 액세스 포인트 화상은 CVS를 시작하지 않는다. 액세스 유닛 (AU) 은 동일한 출력 시간을 공유하는 코딩된 화상들에 대응하는 하나 이상의 코딩된 화상들 및 제어 정보를 포함한다. 화상들의 코딩된 슬라이스들은 비트 스트림 수준에서 NAL (Network Abstraction Layer) 유닛들이라고 하는 데이터 유닛들로 캡슐화된다. 예를 들어, HEVC 비디오 비트 스트림은 NAL 유닛들을 포함하는 하나 이상의 CVS 들을 포함할 수도 있다. NAL 유닛들 각각은 NAL 유닛 헤더를 갖는다. 일례에서, 헤더는 H.264/AVC (다층 확장을 제외) 에 대해 1 바이트 및 HEVC 에 대해 2 바이트이다. NAL 유닛 헤더의 신택스 엘리먼트들은 지정된 비트를 취하므로, 무엇보다도 전송 스트림, RTP (Real-time Transport) 프로토콜, 파일 형식과 같은 모든 종류의 시스템 및 전송 계층에서 볼 수 있다.

[0055] 비디오 코딩 계층 (VCL) NAL 유닛들 및 비-VCL NAL 유닛들을 포함하는 2개의 클래스들의 NAL 유닛들이 HEVC 표준에 존재한다. VCL NAL 유닛은 코딩된 화상 데이터의 하나의 슬라이스 또는 슬라이스 세그먼트 (이하 설명됨) 를 포함하고, 비-VCL NAL 유닛은 하나 이상의 코딩 된 화상과 관련된 제어 정보를 포함한다. 일부 경우들에서, NAL 유닛은 패킷이라고 지칭될 수 있다. HEVC AU 는 코딩된 화상 데이터를 포함하는 VCL NAL 유닛들 및 코딩된 화상 데이터에 대응하는 비-VCL NAL 유닛들 (존재하는 경우) 을 포함한다.

[0056] NAL 유닛들은 비디오 내의 화상들의 코딩된 표현과 같은 비디오 데이터의 코딩된 표현 (예를 들어, 인코딩된 비디오 비트 스트림, 비트 스트림의 CVS 등) 을 형성하는 비트들의 시퀀스를 포함할 수도 있다. 인코더 엔진 (106) 은 각각의 화상을 다수의 슬라이스들로 파티셔닝함으로써 화상들의 코딩된 표현들을 생성한다. 슬라이스는 그 슬라이스 내의 정보가 동일한 화상 내의 다른 슬라이스들로부터의 데이터에 종속되지 않고 코딩되도록 다른 슬라이스들에 대해 독립적이다. 슬라이스는 독립 슬라이스 세그먼트를 포함하는 하나 이상의 슬라이스 세그먼트들, 및 존재하는 경우, 이전 슬라이스 세그먼트들에 의존하는 하나 이상의 종속 슬라이스 세그먼트들을 포함한다. 슬라이스들은 그 후 루마 샘플들 및 크로마 샘플들의 코딩 트리 블록들 (CTBs) 로 파티셔닝된다. 샘플들에 대한 신택스와 함께, 루마 샘플들의 CTB 와 크로마 샘플들의 하나 이상의 CTB 들은 코딩 트리 유닛 (CTU) 으로서 지칭된다. CTU 는 HEVC 인코딩을 위한 기본 프로세싱 단위이다. CTU는 다양한 크기의 다중 코딩 유닛들 (Cus) 로 분할될 수 있다. CU 는 코딩 블록들 (CBs) 이라고 지칭되는 루마 및 크로마 샘플 배열들을 포함한다.

[0057] 루마 및 크로마 CB 들은 예측 블록들 (PB) 로 더 분할될 수 있다. PB 는 (이용 가능하거나 사용을 위해 인에이블될 때) 인터 예측 또는 인트라 블록 복사 예측에 대해 동일한 모션 파라미터들을 사용하는 루마 성분 또는 크

로마 성분의 샘플들의 블록이다. 루마 PB 및 하나 이상의 크로마 PB 들은, 연관된 신택스와 함께, 예측 유닛(PU) 을 형성한다. 인터 예측을 위해, 모션 파라미터들의 세트 (예를 들어, 하나 이상의 모션 벡터들, 참조 인덱스들 등) 가 각각의 PU 에 대해 비트 스트림으로 시그널링되고, 루마 PB 및 하나 이상의 크로마 PB 들의 인터 예측을 위해 사용된다. 모션 파라미터들은 모션 정보로서도 지칭될 수 있다. CB 는 또한 하나 이상의 변환 블록들 (TB) 로 파티셔닝될 수 있다. TB 는 예측 잔차 신호를 코딩하기 위해 동일한 2 차원 변환이 적용되는 컬러 성분의 샘플들의 정사각형 블록을 나타낸다. 변환 유닛 (TU) 은 루마 및 크로마 샘플들의 TB 들 및 대응하는 신택스 엘리먼트들을 나타낸다.

[0058] CU 의 사이즈는 코딩 모드의 사이즈에 대응하고, 형상이 정사각형일 수도 있다. 예를 들어, CU 의 크기는 8 x 8 샘플, 16 x 16 샘플, 32 x 32 샘플, 64 x 64 샘플 또는 해당 CTU 의 크기까지의 임의의 다른 적절한 크기일 수도 있다. 어구 "NxN" 은 수직 및 수평 치수들의 관점에서의 비디오 블록의 픽셀 치수들 (예를 들어, 8 픽셀 x 8 픽셀) 을 지칭하기 위해 여기서 사용된다. 블록에 있어서의 픽셀들은 행 및 열로 배열될 수도 있다. 일부 예들에서, 블록들은 수평 방향에서 수직 방향에서와 동일한 수의 픽셀들을 갖지 않을 수도 있다. CU 와 연관된 신택스 데이터는, 예를 들어, 하나 이상의 PU들로의 CU 의 파티셔닝을 기술할 수도 있다. 파티셔닝 모드들은 CU 가 인트라-예측 모드 인코딩되거나, 또는 인터-예측 모드 인코딩되는지 여부 간에 상이할 수도 있다. PU들은 형상이 비-정방형이도록 파티셔닝될 수도 있다. CU 와 연관된 신택스 데이터는 또한, 예를 들어, CTU 에 따라 하나 이상의 TU들로의 CU 의 파티셔닝을 기술할 수도 있다. TU 는 형상이 정방형이거나 비-정방형일 수 있다.

[0059] HEVC 표준에 따르면, 변환은 변환 유닛 (TU) 을 사용하여 수행될 수도 있다. TU 들은 상이한 CU 들에 대해 다를 수도 있다. TU 들은 주어진 CU 내의 PU 들의 크기에 기초하여 크기가 정해질 수있다. TU들은 PU들과 동일한 사이즈이거나 그 보다 더 작을 수도 있다. 일부 예들에 있어서, CU 에 대응하는 잔차 샘플들은 잔차 쿼드 트리 (RQT) 로서 알려진 쿼드트리 구조를 이용하여 더 작은 유닛들로 세분될 수도 있다. RQT 의 리프 노드들은 TU들에 대응할 수도 있다. TU들과 연관된 픽셀 차이 값들은 변환 계수들을 생성하도록 변환될 수도 있다. 변환 계수는 그 후 인코더 엔진 (106) 에 의해 양자화될 수 있다.

[0060] 일단 비디오 데이터의 화상들이 CU 들로 파티셔닝되면, 인코더 엔진 (106) 은 예측 모드를 사용하여 각각의 PU 를 예측한다. 그 후, 예측 유닛 또는 예측 블록은 잔차들 (후술 됨)를 얻기 위해 원래의 비디오 데이터로부터 감산된다. 각각의 CU 에 대해, 예측 모드는 신택스 데이터를 사용하여 비트 스트림 내부에서 시그널링될 수도 있다. 예측 모드는 인트라 예측 (또는 인트라-픽처 예측) 또는 인터-예측 (또는 인터-픽처 예측) 을 포함할 수도 있다. 인트라 예측은 화상 내에서 공간적으로 이웃하는 샘플 간의 상관을 이용한다. 예를 들어, 인트라-예측을 사용하여, 각각의 PU 는, 예를 들어, PU 에 대한 평균값을 발견하기 위한 DC 예측, PU 에 대해 평면 표면을 피팅 (fitting) 하기 위한 평면 예측, 이웃하는 데이터로부터 외삽하기 위한 방향 예측, 또는 임의의 다른 적절한 유형의 예측을 사용하여 동일한 화상 내의 이웃하는 이미지 데이터로부터 예측된다. 인터 예측은 이미지 샘플들의 블록에 대한 모션 보상된 예측을 도출하기 위해 화상들 간의 시간 상관을 사용한다. 예를 들어, 인터-예측을 사용하여, 각각의 PU 는 (출력 순서로 현재 화상의 전 또는 후의) 하나 이상의 기준 화상들에서의 이미지 데이터로부터의 모션 보상 예측을 사용하여 예측된다. 인터 화상 또는 인트라 화상 예측을 사용하여 화상 영역을 코딩할지 여부의 결정은 예를 들어 CU 레벨에서 행해질 수도 있다.

[0061] 일부 예에서, 화상의 하나 이상의 슬라이스는 슬라이스 유형이 할당된다. 슬라이스 유형에는 I 슬라이스, P 슬라이스 및 B 슬라이스가 포함된다. I 슬라이스 (인트라-프레임, 독립적으로 디코딩 가능) 는 인트라-예측에 의해서만 코딩되는 화상의 슬라이스이며, 따라서 I 슬라이스는 슬라이스의 임의의 예측 유닛 또는 예측 블록을 예측하기 위해 프레임 내의 데이터만을 필요로하기 때문에 독립적으로 디코딩 가능하다. P 슬라이스 (단방향 예측 프레임들) 는 인트라-예측 및 단방향 인터-예측으로 코딩될 수도 있는 화상의 슬라이스이다. P 슬라이스 내의 각각의 예측 유닛 또는 예측 블록은 인트라 예측 또는 인터-예측으로 코딩된다. 인터 예측이 적용될 때, 예측 유닛 또는 예측 블록은 하나의 기준 화상에 의해서만 예측되므로, 기준 샘플은 한 프레임의 하나의 기준 영역으로부터만 나온다. B 슬라이스 (양방향 예측 프레임들) 는 인트라-예측 및 인터-예측 (예를 들어, 양방향 예측 또는 단방향 예측) 으로 코딩될 수도 있는 화상의 슬라이스이다. B 슬라이스의 예측 유닛 또는 예측 블록은 2 개의 기준 화상으로부터 양방향적으로 예측될 수도 있으며, 여기서 각각의 화상은 하나의 기준 영역에 기여하고 2 개의 기준 영역들의 샘플 세트들은 (예를 들어, 동일한 가중치 또는 상이한 가중치를 사용하여) 가중되어, 양방향 예측된 블록의 예측 신호를 생성한다. 전술한 바와 같이, 하나의 화상의 슬라이스들은 독립적으로 코딩된다. 일부 경우들에서, 화상은 단 하나의 슬라이스로서 코딩될 수 있다.

[0062] PU 는 예측 프로세스와 관련된 데이터 (예: 모션 파라미터들 또는 기타 적절한 데이터)를 포함할 수도 있다. 예를 들어, PU 가 인트라-예측을 사용하여 인코딩될 경우, PU 는 그 PU 에 대한 인트라-예측 모드를 기술하는 데

이터를 포함할 수도 있다. 다른 예로서, PU 가 인터-예측을 사용하여 인코딩될 경우, PU 는 그 PU 에 대한 모션 벡터를 정의하는 데이터를 포함할 수도 있다. PU에 대한 모션 벡터를 정의하는 데이터는, 예를 들어, 모션 벡터의 수평 성분(x), 모션 벡터의 수직 성분(y), 모션 벡터에 대한 분해능(예컨대, 정수 정밀도, 1/4 화소 정밀도, 또는 1/8 화소 정밀도), 모션 벡터가 가리키는 참조 픽처, 참조 인덱스, 모션 벡터에 대한 참조 픽처 리스트(예컨대, List 0, List 1, 또는 List C), 또는 그것들의 임의의 조합을 기술할 수도 있다.

[0063] 그 다음, 인코딩 디바이스 (104) 는 변환 및 양자화를 수행할 수도 있다. 예를 들어, 예측에 후속하여, 인코더 엔진 (106) 은 PU 에 대응하는 잔차 값들을 계산할 수도 있다. 잔차 값들은 코딩되는 픽셀들의 현재 블록 (PU) 과 현재 블록을 예측하는데 사용된 예측 블록 (예를 들어, 현재 블록의 예측된 버전) 간의 픽셀 차이 값들을 포함할 수도 있다. 예를 들어, 예측 블록을 생성한 후 (예를 들어, 인터-예측 또는 인트라-예측을 발행한 후), 인코더 엔진 (106) 은 현재 블록으로부터 예측 유닛에 의해 생성 된 예측 블록을 감산함으로써 잔차 블록을 생성할 수 있다. 잔차 블록은 현재 블록의 픽셀 값과 예측 블록의 픽셀 값 사이의 차이를 정량화하는 픽셀 차이 값들의 세트를 포함한다. 일부 예들에서, 잔차 블록은 2 차원 블록 포맷 (예를 들어, 2 차원 매트릭스 또는 어레이의 픽셀 값들) 으로 표현될 수도 있다. 이러한 예에서, 잔차 블록은 픽셀 값들의 2 차원 표현이다.

[0064] 예측이 수행된 후에 남을 수 있는 임의의 잔차 데이터는 이산 코사인 변환, 이산 사인 변환, 정수 변환, 웨이브렛 변환, 다른 적절한 변환 함수 또는 이들의 임의의 조합에 기초할 수도 있는 블록 변환을 사용하여 변환된다. 일부 경우들에서, 하나 이상의 블록 변환들 (예를 들어, 크기 32 x 32, 16 x 16, 8 x 8, 4 x 4 등) 이 각 CU 의 잔차 데이터에 적용될 수도 있다. 일부 예에서, TU 는 인코더 엔진 (106) 에 의해 구현되는 변환 및 양자화 프로세스에 사용될 수있다. 하나 이상의 PU 들을 갖는 소정의 CU 는 또한 하나 이상의 TU들을 포함할 수도 있다. 아래에 더 상세히 기술되는 바와 같이, 잔차 값들은 블록 변환을 사용하여 변환 계수로 변환될 수 있고, 그 후 TU를 사용하여 양자화되고 스캔되어 엔트로피 코딩을 위한 직렬화된 변환 계수를 생성할 수도 있다.

[0065] 일부 예들에서, CU 의 PU들을 이용한 인트라-예측 또는 인터-예측 코딩에 이어서, 인코더 엔진 (106) 은 CU 의 TU들을 위한 잔차 데이터를 계산할 수도 있다. PU 는 공간 도메인 (또는 픽셀 도메인)의 픽셀 데이터를 포함할 수도 있다. TU 는 블록 변환의 적용에 이어서 변환 도메인에 계수들을 포함할 수도 있다. 이전에 주지된 바와 같이, 잔차 데이터는 인코딩되지 않은 픽처의 픽셀들과 PU들에 대응하는 예측 값들 사이의 픽셀 차이 값들에 대응할 수도 있다. 인코더 엔진 (106) 은 CU 에 대한 잔차 데이터를 포함하는 TU들을 형성하고, 그 후, TU들을 변환하여 CU 에 대한 변환 계수들을 생성할 수도 있다.

[0066] 인코더 엔진 (106) 은 변환 계수의 양자화를 수행할 수도 있다. 양자화는 변환 계수를 양자화하여 그 계수를 나타내는 데 사용되는 데이터의 양을 줄임으로써 추가의 압축을 제공한다. 예를 들어, 양자화는 그 계수들의 일부 또는 모두와 연관되는 비트 심도를 감소시킬 수도 있다. 하나의 예에서, n-비트 값을 갖는 계수가 양자화 동안 n이 m 더 큰 m-비트 값으로 버림될 수도 있다.

[0067] 일단 양자화가 수행되면, 코딩된 비디오 비트 스트림은 양자화된 변환 계수, 예측 정보 (예를 들어, 예측 모드, 모션 벡터, 블록 벡터 등), 파티셔닝 정보, 및 다른 신택스 데이터와 같은 임의의 다른 적절한 데이터를 포함한다. 코딩된 비디오 비트 스트림의 상이한 엘리먼트들은 인코더 엔진 (106) 에 의해 엔트로피 인코딩될 수도 있다. 일부 예들에 있어서, 인코더 엔진 (106) 은 양자화된 변환 계수들을 스캔하기 위한 미리정의된 스캔 순서를 이용하여, 엔트로피 인코딩될 수 있는 직렬화된 벡터를 생성할 수도 있다. 일부 예들에 있어서, 인코더 엔진 (106) 은 적응적 스캔을 수행할 수도 있다. 양자화된 변환 계수를 스캐닝하여 벡터 (예를 들어, 1 차원 벡터) 를 형성한 후, 인코더 엔진 (106) 은 벡터를 엔트로피 인코딩할 수 있다. 예를 들어, 인코더 엔진 (106) 은 컨텍스트 적응적 가변 길이 코딩, 컨텍스트 적응적 바이너리 산술 코딩, 신택스 기반 컨텍스트 적응적 바이너리 산술 코딩, 확률 간격 파티셔닝 엔트로피 코딩, 또는 다른 적절한 엔트로피 인코딩 기술을 사용할 수도 있다.

[0068] 전술한 바와 같이, HEVC 비트 스트림은 VCL NAL 유닛들 및 비-VCL NAL 유닛들을 포함하는 NAL 유닛들의 그룹을 포함한다. VCL NAL 유닛들은 코딩된 비디오 비트스트림을 형성하는 코딩된 화상 데이터를 포함한다. 예를 들어, 코딩된 비디오 비트스트림을 형성하는 비트들의 시퀀스는 VCL NAL 유닛들에서 재송신된다. 비-VCL NAL 유닛은 다른 정보 이외에 인코딩된 비디오 비트 스트림에 관한 상위 레벨 정보를 갖는 파라미터 세트들을 포함할 수도 있다. 예를 들어, 파라미터 세트는 비디오 파라미터 세트 (VPS), 시퀀스 파라미터 세트 (SPS) 및 화상 파라미터 세트 (PPS) 를 포함할 수도 있다. 파라미터 세트들의 목표의 예는 비트 레이트 효율성, 오류 복원력 및 시스템 계층 인터페이스들의 제공을 포함한다. 각각의 슬라이스는 단일의 활성 PPS, SPS 및 VPS를 참조하여, 디코딩 디바이스 (112) 가 슬라이스를 디코딩하는데 사용할 수도 있는 정보에 액세스한다. 식별자 (ID) 는 VPS ID, SPS ID 및 PPS ID 를 포함하여 각 파라미터 세트에 대해 코딩될 수도 있다. SPS 는 SPS ID 와 VPS ID 를 포함한다.

PPS 는 PPS ID 와 SPS ID 를 포함한다. 각 슬라이스 헤더는 PPS ID를 포함한다. ID 들을 사용하여, 주어진 슬라이스에 대해 활성 파라미터 세트들이 식별될 수 있다.

[0069] PPS는 주어진 화상의 모든 슬라이스들에 적용되는 정보를 포함한다. 이 때문에, 화상의 모든 슬라이스들은 동일한 PPS 를 참조한다. 상이한 화상들의 슬라이스들은 동일한 PPS 를 또한 참조할 수도 있다. SPS는 동일한 코딩된 비디오 시퀀스 (CVS) 또는 비트 스트림의 모든 화상들에 적용되는 정보를 포함한다. 전술한 바와 같이, 코딩된 비디오 시퀀스 (CVS) 는 특정 특성 (상술됨) 을 갖는 및 기본 계층에서의 랜덤 액세스 포인트 화상 (예를 들어, IDR (instantaneous decode reference) 화상 또는 BLA (broken link access) 화상, 또는 다른 적절한 랜덤 액세스 포인트 화상) 으로 시작하여 특정 특성을 갖는 및 기본 계층에서의 랜덤 액세스 포인트 화상을 갖는 다음 AU 까지의 및 다음 AU 를 포함하지 않는 (또는 비트스트림의 종단까지의) 일련의 액세스 유닛들 (Aus) 이다. SPS 의 정보는 코딩된 비디오 시퀀스 내에서 화상간에 변경되지 않을 수도 있다. 코딩된 비디오 시퀀스에서의 화상들은 동일한 SPS를 사용할 수도 있다. VPS 는 코딩된 비디오 시퀀스 또는 비트 스트림의 모든 계층들에 적용되는 정보를 포함한다. VPS 는 전체 코딩된 비디오 시퀀스에 적용되는 신택스 엘리먼트들을 갖는 신택스 구조를 포함한다. 일부 예들에서, VPS, SPS 또는 PPS 는 인코딩된 비트스트림과 함께 대역 내 전송될 수도 있다. 일부 예에서, VPS, SPS 또는 PPS는 코딩된 비디오 데이터를 포함하는 NAL 유닛과는 별도의 송신으로 대역 외 (out-of-band) 로 전송될 수도 있다.

[0070] 비디오 비트 스트림은 또한 SEI (Supplemental Enhancement Information) 메시지를 포함할 수 있다. 예를 들어, SEI NAL 유닛은 비디오 비트 스트림의 일부일 수 있다. 일부 경우들에서, SEI 메시지는 디코딩 프로세스에서 필요하지 않은 정보를 포함할 수 있다. 예를 들어, SEI 메시지 내의 정보는 디코더가 비트 스트림의 비디오 화상을 디코딩하는 데 필수적이지 않을 수 있지만, 디코더는 그 정보를 사용하여 화상 (예를 들어, 디코딩된 출력) 의 디스플레이 또는 프로세싱을 향상시킬 수 있다. SEI 메시지 내의 정보는 임베딩된 메타데이터일 수 있다. 하나의 예시적인 예에서, SEI 메시지 내의 정보는 디코더 측 엔티티에 의해 이용되어 콘텐츠의 가시성을 향상시킬 수 있다. 일부 예들에서, 특정 애플리케이션 표준은 비트 스트림 내의 이러한 SEI 메시지의 존재를 요구하여, 그 애플리케이션 표준을 준수하는 모든 디바이스들에 품질 향상을 가져올 수 있다 (예를 들어, 다수의 다른 예들에 더하여, SEI 메시지가 비디오의 모든 프레임에 대해 반송되는, 프레임-호환형 평면-입체 3DTV 비디오 포맷에 대한 프레임-팩킹 SEI 메시지의 캐리지, 복구 포인트 SEI 메시지의 핸들링, DVB에서의 팬-스캔 스캔 사각형 SEI 메시지의 사용).

[0071] 인코딩 디바이스 (104) 의 출력 (110) 은 인코딩된 비디오 데이터를 구성하는 NAL 유닛들을 통신 링크 (120) 를 통해 수신 디바이스의 디코딩 디바이스 (112) 로 전송할 수도 있다. 디코딩 디바이스 (112) 의 입력 (114) 은 NAL 유닛들을 수신할 수도 있다. 통신 링크 (120) 는 무선 네트워크, 유선 네트워크, 또는 유선 및 무선 네트워크의 조합에 의해 제공되는 채널을 포함할 수도 있다. 무선 네트워크는 임의의 무선 인터페이스 또는 무선 인터페이스의 조합을 포함할 수 있으며, 임의의 적합한 무선 네트워크 (예를 들어, 인터넷 또는 다른 광역 네트워크, 패킷 기반 네트워크, WiFiTM, 무선 주파수 (RF), UWB, WiFi-다이렉트, 셀룰러, LTE (Long-Term Evolution), WiMaxTM 등) 을 포함할 수도 있다. 유선 네트워크는 임의의 유선 인터페이스 (예 : 광섬유, 이더넷, 전력선 이더넷, 동축 케이블을 통한 이더넷, DSL (Digital Signal Line) 등) 를 포함할 수도 있다. 유선 및/또는 무선 네트워크는 기지국, 라우터, 액세스 포인트, 브리지, 게이트웨이, 스위치 등과 같은 다양한 장비를 사용하여 구현될 수있다. 인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라 변조되고, 수신 디바이스로 송신될 수도 있다.

[0072] 일부 예에서, 인코딩 장치 (104) 는 인코딩된 비디오 데이터를 저장 장치 (108) 에 저장할 수있다. 출력 (110) 은 인코더 엔진 (106) 또는 저장 장치 (108)로부터 인코딩된 비디오 데이터를 취출할 수있다. 저장 장치 (108) 는 다양한 분산 또는 로컬 액세스 데이터 저장 매체 중 임의의 것을 포함할 수있다. 예를 들어, 저장 장치 (108)는 하드 드라이브, 저장 디스크, 플래시 메모리, 휘발성 또는 비 휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적절한 디지털 저장 매체를 포함할 수있다.

[0073] 디코딩 디바이스 (112) 의 입력 (114) 은 인코딩된 비디오 비트 스트림 데이터를 수신하고 디코더 엔진 (116)에 의해 나중에 사용하기 위해 비디오 비트 스트림 데이터를 디코더 엔진 (116) 또는 저장 장치 (118)에 제공 할 수있다. 디코더 엔진 (116)은 인코딩 된 비디오 데이터를 구성하는 하나 이상의 코딩된 비디오 시퀀스의 엘리먼트들을 엔트로피 디코딩 (예를 들어, 엔트로피 디코더를 사용) 하고 추출함으로써 인코딩 된 비디오 비트 스트림 데이터를 디코딩할 수있다. 그 다음, 디코더 엔진 (116) 은 인코딩 된 비디오 비트 스트림 데이터에 대해 리스케일링 및 역변환을 수행할 수있다. 잔차 데이터는 그 후 디코더 엔진 (116) 의 예측 스테이지로 전달된다. 디코더 엔진 (116)은 픽셀들의 블록 (예를 들어, PU) 을 예측한다. 일부 예에서, 그 예측은 역변환의 출력 (잔

차 데이터) 에 추가된다.

[0074] 디코딩 디바이스 (112) 는 디코딩된 비디오 데이터를 컨텐츠의 소비자에게 디스플레이하기 위한 디스플레이 또는 다른 출력 디바이스를 포함할 수 있는 비디오 목적지 디바이스로 디코딩된 비디오를 출력할 수있다. 일부 양태들에서, 비디오 목적지 디바이스 (122)는 디코딩 디바이스 (112)를 포함하는 수신 디바이스의 일부일 수있다. 일부 양태들에서, 비디오 목적지 디바이스 (122) 는 수신 디바이스 이외의 별도의 디바이스의 일부일 수있다.

[0075] 일부 예에서, 비디오 인코딩 디바이스 (104) 및/또는 비디오 디코딩 디바이스 (112) 는 각각 오디오 인코딩 디바이스 및 오디오 디코딩 디바이스와 통합될 수있다. 비디오 인코딩 디바이스 (104) 및/또는 비디오 디코딩 디바이스 (112) 는 또한 하나 이상의 마이크로프로세서들, 디지털 신호 프로세서들 (DSP들), 주문형 집적 회로들 (ASIC들), 필드 프로그래밍가능 게이트 어레이들 (FPGA들), 이산 로직, 소프트웨어, 하드웨어, 펌웨어 또는 이들의 임의의 조합들과 같은, 상술된 코딩 기술들을 구현하는데 필요한 다른 하드웨어 또는 소프트웨어를 포함할 수도 있다. 비디오 인코딩 디바이스 (104) 및 비디오 디코딩 디바이스 (112) 는 각각의 디바이스에서 결합된 인코더/디코더 (코덱) 의 일부로서 통합될 수있다. 인코딩 디바이스 (104) 의 특정 세부 사항의 예가 도 9 를 참조하여 아래에서 설명된다. 디코딩 디바이스 (112) 의 특정 세부 사항의 예가 도 10 을 참조하여 아래에서 설명된다.

[0076] HEVC 표준의 확장들은 MV-HEVC 라고하는 멀티뷰 비디오 코딩 확장과 SHVC 라고하는 스케일러블 비디오 코딩 확장을 포함한다. MV-HEVC 및 SHVC 확장은 계층화된 코딩의 개념을 공유하며, 인코딩된 비디오 비트 스트림에 상이한 계층들이 포함된다. 코딩된 비디오 시퀀스의 각 계층은 고유 계층 식별자 (ID) 로 어드레싱된다. 계층 ID 는 NAL 유닛이 연관되는 계층을 식별하기 위해 NAL 유닛의 헤더 내에 존재할 수도 있다. MV-HEVC 에서, 상이한 계층들은 일반적으로 비디오 비트 스트림의 동일한 장면에 대한 상이한 뷰들 (views) 을 나타낸다. SHVC 에서, 서로 다른 공간 해상도 (또는 화상 해상도) 또는 서로 다른 재구성 충실도로 비디오 비트 스트림을 나타내는 서로 다른 스케일러블 계층들이 제공된다. 스케일러블 계층들은 베이스 계층 (계층 ID = 0) 및 하나 이상의 향상 계층 (계층 ID = 1, 2, ... N) 을 포함할 수있다. 기본 계층은 HEVC 의 제 1 버전의 프로파일을 따르고, 비트 스트림에서 최저 이용가능 계층을 나타낸다. 향상 계층은 기본 계층에 비해 증가된 공간 해상도, 시간 해상도 또는 프레임 레이트 및/또는 재구성 충실도 (또는 품질) 를 갖는다. 향상 계층들은 계층적으로 구성되며 하위 계층들에 종속될 수도 있고 그렇지 않을 수도 있다. 일부 예에서, 상이한 계층들은 단일 표준 코덱을 사용하여 코딩될 수도 있다 (예를 들어, 모든 계층들이 HEVC, SHVC 또는 다른 코딩 표준을 사용하여 인코딩 됨). 일부 예들에서, 상이한 계층들은 다중 표준 코덱을 사용하여 코딩될 수있다. 예를 들어, 기본 계층은 AVC를 사용하여 코딩될 수 있는 반면, 하나 이상의 향상 계층은 HEVC 표준에 대한 SHVC 및/또는 MV-HEVC 확장을 사용하여 코딩될 수있다.

[0077] 위에서 언급 한 바와 같이, SEI 메시지는 인코딩 된 비디오 비트 스트림으로부터 루마 및 크로마 값을 생성하는데 필요하지 않은 정보를 제공 할 수 있다. 그러나, SEI 메시지는 그럼에도 불구하고 컨텐츠 생성기에 의해 의도 된 바와 같이 비트 스트림을 렌더링하는데 필수적인 것으로 간주 될 수 있는 정보를 반송할 수도 있다.

[0078] 도 2 는 비트 스트림의 일부일 수 있는 액세스 유닛 (200) (AU) 의 예를 도시한다. 비트 스트림은 일련의 액세스 유닛을 포함 할 수 있으며, 각각은 코딩 된 픽처 및 가능하면 비트 스트림에 대한 일부 정보를 제공한다. 각각의 액세스 유닛을 디코딩하면 하나의 디코딩 된 화상이 생성된다. 여기에 도시 된 예시적인 액세스 유닛 (200)과 같은 액세스 유닛은 H.264/AVC 및 H.265/HEVC 표준에 포함된다.

[0079] 도 2 의 예에 예시된 바와 같이, 액세스 유닛 (200) 은 일차 코딩 된 픽처를 함께 구성하는 NAL 유닛의 세트를 포함 할 수 있다. 액세스 유닛 (200)은 여기서 점선으로 도시 된 선택적 NAL 유닛을 포함 할 수있다. 액세스 유닛 (200)은 선택적으로 액세스 유닛 구분자 (202)로 시작할 수있다. 액세스 유닛 구분자는 액세스 유닛의 시작을 로케이팅하는 것을 도울 수있다. 액세스 유닛 (200)은 또한 1 차 코딩 된 픽처 (206)에 선행하는 보충 강화 정보 (204) (예를 들어, SEI 메시지)를 선택적으로 포함 할 수있다. SEI는 픽쳐 타이밍 정보와 같은 데이터를 포함 할 수있다. 1 차 코딩 된 화상 (206)은 비디오 코딩 계층 (VCL) NAL 유닛을 포함 할 수있다. VCL NAL 유닛은 비디오 픽처의 샘플을 나타내는 슬라이스 또는 슬라이스 데이터 파티션을 포함한다. 1 차 코딩 된 픽처 (206) 다음에는 중복 코딩 된 픽처 (208)를 포함하는 일부 추가 VCL NAL 유닛이 있을 수있다. 중복 코딩 된 픽처 (208) 는 동일한 비디오 픽처의 영역의 여분의 표현을 포함 할 수 있고, 일차 코딩 된 픽처 (206)에서 데이터의 손실 또는 손상을 복구하는 데 디코더에 의한 사용을 위해 이용가능하다. 중복 코딩 된 픽처가 존재하면, 디코더는 중복 코딩 된 픽처를 디코딩 할 필요가 없다.

[0080] 일차 코딩 된 픽처 (206)가 코딩 된 비디오 시퀀스의 마지막 픽처 인 경우, 액세스 유닛 (200)은 시퀀스의 종단

을 표시하기 위해 시퀀스의 종단 (210) NAL 유닛을 포함 할 수있다. 일차 코딩 된 픽처 (206)가 또한 비트 스트림에서 마지막 코딩 된 픽처 인 경우, 액세스 유닛 (200)은 또한 스트림의 종단 (212) NAL 유닛 (비트 스트림의 종단 NAL 유닛이라고도 함)을 포함 할 수있다.

[0081] SEI NAL 유닛은 다양한 정보를 디코더에 제공하기 위해 사용될 수있다. 아래의 표 1 은 ITU-T (International Telecommunications Union) H.265 “시리즈 H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS, Infrastructure of audiovisual services - Coding of moving video (이하 추천 ITU-T H.265) 에 의해서 제공된 SEI 메시지의 일반적인 신택스의 예를 제공한다. 표 1 에서, SEI 메시지에 포함될 수있는 서로 다른 정보에 대해 서로 다른 페이로드 유형이 정의되었다. 아래에 제공된 신택스는 HEVC 표준에 정의 및/또는 포함된 SEI 메시지 유형의 서브세트 만 포함한다.

표 1

[0082]

sei_payload (payloadType, payloadSize) {	디스크립터
if(nal_unit_type = = PREFIX_SEI_NUT)	
if(payloadType = = 0)	
buffering_period(payloadSize)	
else if(payloadType = = 1)	
pic_timing(payloadSize)	
else if(payloadType = = 2)	
pan_scan_rect(payloadSize)	
else if(payloadType = = 3)	
filler_payload(payloadSize)	
else if(payloadType = = 4)	
user_data_registered_itu_t_t35(payloadSize)	
else if(payloadType = = 5)	
user_data_unregistered(payloadSize)	
else if(payloadType = = 6)	
recovery_point(payloadSize)	
else if(payloadType = = 9)	
scene_info(payloadSize)	
else if(payloadType = = 15)	
picture_snapshot(payloadSize)	
else if(payloadType = = 16)	
progressive_refinement_segment_start(payloadSize)	
else if(payloadType = = 17)	
progressive_refinement_segment_end(payloadSize)	
else if(payloadType = = 19)	
film_grain_characteristics(payloadSize)	
else if(payloadType = = 22)	
post_filter_hint(payloadSize)	
else if(payloadType = = 23)	
tone_mapping_info(payloadSize)	
else if(payloadType = = 45)	
frame_packing_arrangement(payloadSize)	
else if(payloadType = = 47)	
display_orientation(payloadSize)	
else if(payloadType = = 128)	
structure_of_pictures_info(payloadSize)	
else if(payloadType = = 129)	
active_parameter_sets(payloadSize)	
else if(payloadType = = 130)	
decoding_unit_info(payloadSize)	
else if(payloadType = = 131)	
temporal_sub_layer_zero_index(payloadSize)	
else if(payloadType = = 133)	
scalable_nesting(payloadSize)	
else if(payloadType = = 134)	

region_refresh_info(payloadSize)	
else if(payloadType = = 135)	
no_display(payloadSize)	
else if(payloadType = = 136)	
time_code(payloadSize)	
else if(payloadType = = 137)	
mastering_display_colour_volume(payloadSize)	
else if(payloadType = = 138)	
segmented_rect_frame_packing_arrangement(payloadSize)	
else if(payloadType = = 139)	
temporal_motion_constrained_tile_sets(payloadSize)	
else if(payloadType = = 140)	
chroma_resampling_filter_hint(payloadSize)	
else if(payloadType = = 141)	
knee_function_info(payloadSize)	
else if(payloadType = = 142)	
colour_remapping_info(payloadSize)	
else if(payloadType = = 143)	
deinterlaced_field_identification(payloadSize)	
else if(payloadType = = 160)	
layers_not_present(payloadSize)	
else if(payloadType = = 161)	
inter_layer_constrained_tile_sets(payloadSize)	
else if(payloadType = = 162)	
bsp_nesting(payloadSize)	
else if(payloadType = = 163)	
bsp_initial_arrival_time(payloadSize)	
else if(payloadType = = 164)	
sub_bitstream_property(payloadSize)	
else if(payloadType = = 165)	
alpha_channel_info(payloadSize)	
else if(payloadType = = 166)	
overlay_info(payloadSize)	
else if(payloadType = = 167)	
temporal_mv_prediction_constraints(payloadSize)	
else if(payloadType = = 168)	
frame_field_info(payloadSize)	
else if(payloadType = = 176)	
three_dimensional_reference_displays_info(payloadSize)	
else if(payloadType = = 177)	
depth_representation_info(payloadSize)	
else if(payloadType = = 178)	
multiview_scene_info(payloadSize)	
else if(payloadType = = 179)	
multiview_acquisition_info(payloadSize)	
else if(payloadType = = 180)	
multiview_view_position(payloadSize)	
else if(payloadType = = 181)	
alternative_depth_info(payloadSize)	
else	
reserved_sei_message(payloadSize)	
else /* nal_unit_type = = SUFFIX_SEI_NUT */	
if(payloadType = = 3)	
filler_payload(payloadSize)	
else if(payloadType = = 4)	
user_data_registered_itu_t_t35(payloadSize)	
else if(payloadType = = 5)	
user_data_unregistered(payloadSize)	

else if(payloadType = = 17)	
progressive_refinement_segment_end(payloadSize)	
else if(payloadType = = 22)	
post_filter_hint(payloadSize)	
else if(payloadType = = 132)	
decoded_picture_hash(payloadSize)	
else	
reserved_sei_message(payloadSize)	
if(more_data_in_payload( )) {	
if(payload_extension_present( ))	
reserved_payload_extension_data	u(v)
payload_bit_equal_to_one	f(1)
while(!byte_aligned( ))	
payload_bit_equal_to_zero	f(1)
}	
}	

[0083] 여러 비디오 코딩 사양에서, 정보의 피스 (piece) 가 픽쳐 샘플의 디코딩에 영향을 주지 않는 한, 그 정보는 SEI 메시지에 포함될 수 있다. SEI 메시지에서 반송되는 일부 정보는 최적화 목적을 위한 것일 수도 있고 수용 불가한 방식으로 제시되는 비디오의 렌더링 없이 디코더에 의해 무시될 수도 있다. 그러나, 프레임 팩킹 정보, 전방향 투영 표시 정보, 및 서브타이틀링 또는 캡셔닝 정보와 같은, SEI 메시지에서 반송되는 일부 다른 정보는 수용가능한 비디오 시스템 성능을 보장하기 위해 필수적일 수도 있다. 예를 들어, 디팩킹 또는 디워핑과 같은 특정의 포스트 디코딩 동작들은 프레임 팩킹되거나 360 도 전방향 투영된 비디오가 디스플레이될 수 있기 전에 적용되어야 한다. 유사하게, 서브타이틀링 또는 캡셔닝은 비디오에 수반하는 오디오가 뷰어가 이해하는 언어가 아닌 경우 필요할 수도 있다.

[0084] 일부 코딩 시스템 및 애플리케이션의 경우, 프레임 팩킹 정보, 전방향 투영 표시 정보, 및 서브타이틀링 또는 캡셔닝 정보와 같은 정보는 비디오 애플리케이션 시스템이 효율적으로 및/또는 올바르게 수행하는데 (예를 들어, 비디오가 보여지도록 의도되었던 때에 비디오를 제시하는데) 필수적이다. 이들 시스템 및 애플리케이션의 경우, 그러한 정보는 따라서 (예를 들어, 시스템 동작들에서 코덱, 프로파일, 및 레벨로서) 비디오 디코딩 인터오퍼러빌리티에 중요한 정보로서 유사하게 다루어져야 한다. 예를 들어, 이러한 필수 정보는 코딩된 비디오 비트스트림을 예를 들어 ISO 베이스 미디어 파일 포맷 (ISOBMFF) 에 따른 미디어 파일 또는 HTTP 를 통한 동적 적응 스트리밍 (DASH) 에 따른 미디어 제시로 캡슐화할 때 고려되어야 한다.

[0085] 코딩된 비디오 비트스트림을 ISOBMFF 파일 또는 DASH 미디어 제시로 캡슐화할 때, 상술된 바와 같은 필수 정보는 예를 들어 파일 포맷 샘플 엔트리, DASH 미디어 제시 디스트립션 (MPD), 및/또는 MIME 유형 파라미터들에서와 같은 높은 레벨에서 노출될 수 있다. 캡슐화 형성에서 높은 레벨에서 필수 정보를 배치함으로써, 디코더는 비트스트림 내의 SEI 메시지에 무슨 정보가 포함될 수도 있는지를 결정하기 위해 전체 비트스트림을 스캔해야 하는 것으르 피할 수 있다. 또한, 디코더는 필수성 조건에 기초하여 SEI 메시지를 카테고리화해야 하는 것을 피할 수 있다. 대신에, 하나의 SEI 메시지에서 집합적으로, (예를 들어, 프로파일 및 레벨 정보가 시그널링되는 경우), 비트스트림에서 조기에 쉽게 액세스가능한, 또는 기존 표준들에 대한 그러한 정보를 갖는 것은 디코더의 구현을 단순화할 수 있을 것이다. 필수 정보에 대해 조기에 액세스하는 것은 디코더가 디코딩된 비디오를 최선으로 다룰 방법에 대한 선제적 결정을 행하는 것을 가능하게 할 수 있다. 또한, 비디오 코딩 표준들은 필수 정보를 포함하는 SEI 메시지가 정의될 때마다 변경될 필요가 없다.

[0086] 여러 구현들에서, SEI 메시지는 필수 보충 정보 (ESI) SEI 메시지로서 지칭되는 필수 정보를 위해 설계되고, SEI 메시지는 비필수 보충 정보 (NSI) SEI 메시지로서 지칭되는 비필수 정보를 위해 설계된다. 상술된 바와 같이, 용어들 “필수” 및 “비필수” 는 SEI 메시지들의 2 개의 상이한 유형들을 분류하는데 사용되고 있다. ESI SEI 및 NSI SEI 메시지들에서 전달되는 정보는 예를 들어 비디오 제시를 인코딩하고 있는 인코더의 구성에서 컨텐츠 생성기에 의해 결정된다. ESI SEI 및 NSI SEI 는 임의의 특정의 정보를 전달할 필요가 없고, 다른 데이터를 프로세싱하기 전에 디코더가 프로세싱해야 하는 정보를 식별하는데 있어서 디코더에 의해 사용될 수 있다.

[0087] 여러 구현들에서, 인코더는 필수인 정보를 결정할 수 있다. 예를 들어, 인코더는 특정의 정보를 필수로서 식별하도록 구성될 수 있다. 이들 예에서, 인코더는 필수 정보를 위해 ESI SEI 를 이용할 수 있다. 인코더는 또한

선택적으로 비필수 정보를 위해 NSI ESI 를 이용할 수 있다.

[0088] 여러 구현들에서, 디코더는 ESI SEI 및 NSI ESI 메시지들을 식별하도록, 그리고 이들 메시지에 포함된 정보를 이용하도록 구성될 수 있다. 디코더가 이들 메시지들 중 어느 것도 발견하지 않는 경우, 디코더는 적용가능한 표준에 따라 비트스트림을 디코딩하는 것을 진행할 수 있다. ESI SEI 또는 NSI SEI 메시지들을 프로세싱하도록 구성되지 않은 디코더는 이들 메시지들을 식별 불가능한 SEI 메시지들로서 다룰 수 있다. 일부 예들에서, 디코더가 식별 불가능한 SEI 메시지를 만나는 경우, 디코더는 비트스트림을 프로세싱하는 것을 중단하도록 구성될 수 있다. 대안적으로, 일부 예들에서, 디코더는 비트스트림을 디코딩하는 것을 진행할 수도 있다.

[0089] 다음의 예들에서, ESI SEI 및 NSI SEI 메시지들은 HEVC 의 콘텍스트에서 기술되고, 이들 예들은 또한 AVC 와 같은 SEI 메시지들을 지원하는 다른 비디오 코딩 표준들에 적용될 수 있다는 것이 이해된다.

[0090] ESI SEI 메시지

[0091] 다음은 ESI SEI 메시지에 대한 예시의 신택스 및 시맨틱스를 제공한다. 유사한 신택스 및 시맨틱스가 여러 비디오 코딩 표준들과 함께 사용될 수 있다.

[0092] ESI SEI 메시지 신택스

[0093] ESI SEI 메시지 신택스의 예는 다음과 같다:

[0094]

essential_supplemental_information(payloadSize) {	Descriptor
num_essential_sei_msg_types	u(16)
for(i = 0; i < num_essential_sei_msg_types; i++) {	
essential_sei_payload_type[i]	u(16)
num_esei_init_payload_databits[i]	u(16)
for(j = 0; j < num_esei_init_pldbits; j++) {	
esei_payload_data_bit[i][j]	u(1)
while(!byte_aligned( ))	
esei_alignment_bit_equal_to_zero /* equal to 0 */	f(1)
}	
}	

[0095] 대안적으로, 3 개의 u(16)-코딩된 신택스 엘리먼트들의 코딩은 ue(v) 코딩으로 대체된다 (즉, 위의 "u(16)" 의 모든 인스턴스들을 "ue(v)" 로 대체한다).

[0096] ESI SEI 메시지 시맨틱스

[0097] 필수 보충 정보 (ESI) SEI 메시지는 비트스트림에 존재하는 모든 필수 SEI 메시지들에 관한 정보를 전달한다. 필수 SEI 메시지는 반송된 정보가 인코더 (즉, 컨텐츠 생성기) 에 의해 디코더측이 바람직한 사용자 경험을 가능하게 하기 위해 적적하게 프로세싱하기 위해 필수적인 것으로 고려되는 그러한 SEI 메시지이다. 어느 유형의 SEI 메시지들이 특정의 비트스트림에서 필수 SEI 메시지들인지를 결정하는 것은 인코더 (즉, 컨텐츠 생성기) 의 재량이다.

[0098] ESI SEI 메시지가 CVS 의 임의의 액세스 유닛 (AU) 에 존재하는 경우, ESI SEI 메시지는 CVS 의 제 1 액세스 유닛에 존재할 것이다. ESI SEI 메시지는 디코딩 순서에서 현재의 액세스 유닛으로부터 CVS 의 종단까지 지속된다.

[0099] ESI SEI 메시지를 포함하는 SEI NAL 유닛은 비필수 보충 정보 SEI 메시지가 아닌 임의의 다른 SEI 메시지를 포함하지 않을 것이다. CVS 에 존재하는 다수의 ESI 메시지가 있는 경우, 그들은 동일한 컨텐츠를 가질 것이다.

[0100] num_essential_sei_msg_types 은 필수 SEI 메시지들로서 고려되는 CVS 에 존재하는 SEI 메시지들의 유형들의 수를 특정한다.

[0101] essential_sei_payload_type[i] 는 필수 SEI 메시지들의 제 i 유형의 payloadType 값을 나타낸다. 임의의 2 개의 상이한 값들 m 및 n 의 경우, essential_sei_payload_type[m] 및 essential_sei_payload_type[n] 의 값들은 동일하지 않을 것이다.

[0102] num_esei_init_payload_databits[i] 는 CVS 에 존재하는 필수 SEI 메시지들의 제 i 유형에 대한 추가 정보를

제공하는 다음의 비트들의 수를 특정한다.

[0103] **esei_payload_data_bit**[i][j] 는 CVS 에 존재하는 필수 SEI 메시지들의 제 i 유형에 대한 추가 정보를 제공하는 비트들의 제 j 비트를 특정한다.

[0104] num_esei_init_payload_databits[i] 비트들은 필수 SEI 메시지들의 제 i 유형의 payloadType 에 대한 SEI 페이로드 신택스에 후속하고, SEI 페이로드 신택스에서의 제 1 신택스 엘리먼튱서 시작하는 다수의 신택스 엘리먼트들을 포함하지만, SEI 페이로드 신택스에서의 모든 신택스 엘리먼트들을 포함할 수도 있거나 포함하지 않을 수도 있다. 마지막 비트는 SEI 페이로드 신택스 내의 신택스 엘리먼트의 마지막 비트일 것이다. 이들 비트들에 포함되는 신택스 엘리먼트들은 여기의 정보가 전체 CVS 에 적용된다는 것을 제외하고, 마치 그들이 payloadType 을 갖는 SEI 메시지 내에 있는 것처럼 동일한 시맨틱스를 갖는다.

[0105] 이들 비트들은 디코더측이 바람직한 사용자 경험을 가능하게 하기 위해 비트스트림을 적절하게 프로세싱할 수 있는지 여부를 어느 시스템 계층 엔티티가 결정할 수 있는지에 기초하여, 프로세싱 능력 관점에서 디코더측에서 어떤 유형들의 프로세싱이 필요한지를 표시하기 위한 충분한 정보를 제공해야 한다. 예를 들어, payloadType 이 45 (즉, 프레임 팩킹 배열 SEI 메시지) 인 경우, 이들 비트들은 적어도 신택스 엘리먼트 frame_packing_arrangement_type 을 포함해야 한다. 다른 예의 경우, payloadType 이 150 (즉, JCTVC-Z1005 에서 특정된 바와 같은 전방향 투영 표시 SEI 메시지) 인 경우, 이들 비트들은 적어도 신택스 엘리먼트 projection_type 을 포함해야 한다.

[0106] **esei_alignment_bit_equal_to_zero** 는 0 과 동일할 것이다.

[0107] NSI SEI 메시지

[0108] 다음은 NSI SEI 메시지에 대한 예시의 신택스 및 시맨틱스를 제공한다. 유사한 신택스 및 시맨틱스가 여러 비디오 코딩 표준들과 함께 사용될 수 있다.

[0109] NSI SEI 메시지 신택스

[0110] NSI SEI 메시지 신택스의 예는 다음과 같다:

[0111]

nonessential_supplemental_information(payloadSize) {	Descriptor
num_nonessential_sei_msg_types	u(16)
for(i = 0; i < num_nonessential_sei_msg_types; i++) {	
nonessential_sei_payload_type[i]	u(16)
num_nesei_init_payload_databits[i]	u(16)
for(j = 0; j < num_nesei_init_pldbits; j++) {	
nesei_payload_data_bit[i][j]	u(1)
while(!byte_aligned( ))	
nesei_alignment_bit_equal_to_zero /* equal to 0 */	f(1)
}	
}	

[0112] 대안적으로, 3 개의 u(16)-코딩된 신택스 엘리먼트들의 코딩은 ue(v) 코딩으로 대체되며, 즉, 위의 “u(16)” 의 모든 인스턴스들을 “ue(v)” 로 대체한다.

[0113] NSI SEI 메시지 시맨틱스

[0114] 비필수 보충 정보 (NSI) SEI 메시지는 비트스트림에 존재하는 모든 비필수 SEI 메시지들에 관한 정보를 전달한다. 비필수 SEI 메시지는 인코더 (즉, 컨텐츠 생성기) 에 의해 필수 SEI 메시지로서 고려되지 않는 그러한 SEI 메시지이다. 어느 유형의 SEI 메시지들이 특정의 비트스트림에서 비필수 SEI 메시지들인지를 결정하는 것은 인코더 (즉, 컨텐츠 생성기) 의 재량이다.

[0115] NSI SEI 메시지가 CVS 의 임의의 액세스 유닛에 존재하는 경우, NSI SEI 메시지는 CVS 의 제 1 액세스 유닛에 존재할 것이다. NSI SEI 메시지는 디코딩 순서에서 현재의 액세스 유닛으로부터 CVS 의 종단까지 지속된다.

[0116] NSI SEI 메시지를 포함하는 SEI NAL 유닛은 ESI SEI 메시지가 아닌 임의의 다른 SEI 메시지를 포함하지 않을 것이다. CVS 에 존재하는 다수의 NSI 메시지가 있는 경우, 그들은 동일한 컨텐츠를 가질 것이다.

[0117] 특정의 유형의 SEI 메시지들이 CVS 에서의 ESI SEI 메시지에 의해 필수 SEI 메시지들로서 식별되는 경우, 동일

한 유형의 SEI 메시지들은 CVS 에서의 NSI SEI 메시지에 의해 비필수 SEI 메시지들로서 식별되지 않을 것이다.

[0118] **num_nonessential_sei_msg_types** 은 비필수 SEI 메시지들로서 고려되는 CVS 에 존재하는 SEI 메시지들의 유형들의 수를 특정한다.

[0119] **nonessential_sei_payload_type**[i] 는 비필수 SEI 메시지들의 제 i 유형의 payloadType 값을 나타낸다. 임의의 2 개의 상이한 값들 m 및 n 의 경우, nonessential_sei_payload_type[m] 및 nonessential_sei_payload_type[n] 의 값들은 동일하지 않을 것이다.

[0120] **num_nesei_init_payload_databits**[i] 는 CVS 에 존재하는 비필수 SEI 메시지들의 제 i 유형에 대한 추가 정보를 제공하는 다음의 비트들의 수를 특정한다.

[0121] **nesei_payload_data_bit**[i][j] 는 CVS 에 존재하는 비필수 SEI 메시지들의 제 i 유형에 대한 추가 정보를 제공하는 비트들의 제 j 비트를 특정한다.

[0122] num_nesei_init_payload_databits[i] 비트들은 비필수 SEI 메시지들의 제 i 유형의 payloadType 에 대한 SEI 페이로드 신택스에 후속하고, SEI 페이로드 신택스에서의 제 1 신택스 엘리먼트에서 시작하는 다수의 신택스 엘리먼트들을 포함하지만, SEI 페이로드 신택스에서의 모든 신택스 엘리먼트들을 포함할 수도 있거나 포함하지 않을 수도 있다. 마지막 비트는 SEI 페이로드 신택스 내의 신택스 엘리먼트의 마지막 비트일 것이다. 이들 비트들에 포함되는 신택스 엘리먼트들은 여기의 정보가 전체 CVS 에 적용된다는 것을 제외하고, 마치 그들이 payloadType 을 갖는 SEI 메시지 내에 있는 것처럼 동일한 시맨틱스를 갖는다.

[0123] **nesei_alignment_bit_equal_to_zero** 는 0 과 동일할 것이다.

[0124] 여러 구현들에서, 신택스 및 시맨틱스는 SEI 메시지들의 존재, 부재, 및/또는 필수성을 표시하기 위해 제공된다. 여러 구현들에서, SEI 프리픽스 표시들을 포함하는 SEI 메시지 뿐아니라 SEI 메니페스트가 정의된다. SEI 메니페스트는 비트스트림에 포함되는 SEI 메시지들에 대한 정보를 전달하는데 사용될 수 있다. SEI 프리픽스 표시들은 SEI 메시지 내의 페이로드를 식별하는데 사용될 수 있고, 비트스트림 내의 컨텐츠에 필요한 프로세스의 유형을 결정하기 위해 디코더에 의해 사용될 수 있다.

[0125] 여러 구현들에서, 인코더는 필수인 정보를 결정할 수 있다. 예를 들어, 인코더는 비디오의 디스플레이에 필수인 특정의 정보를 식별하도록 구성될 수 있다. 이들 예들에서, 인코더는 필수 정보를 표시하기 위해 SEI 프리픽스 표시 SEI 메시지를, 및 비트스트림에 포함된 SEI 메시지를 기술하는 SEI 메니페스트 SEI 메시지를 이용할 수 있다.

[0126] 여러 구현들에서, 디코더는 SEI 메니페스트 SEI 메시지 및 SEI 프리픽스 SEI 메시지를 식별하도록, 그리고 이들 메시지들에 포함된 정보를 이용하도록 구성될 수 있다. 디코더가 이들 메시지들 중 어느 것도 발견하지 않는 경우, 디코더는 적용가능한 표준에 따라 비트스트림을 디코딩하는 것을 진행할 수 있다. SEI 메니페스트 및 SEI 프리픽스 메시지들을 프로세싱하도록 구성되지 않은 디코더는 이들 메시지들을 식별 불가능한 SEI 메시지들로서 다룰 수 있다. 일부 예들에서, 디코더가 식별 불가능한 SEI 메시지를 만나는 경우, 디코더는 비트스트림을 프로세싱하는 것을 중단하도록 구성될 수 있다. 대안적으로, 일부 예들에서, 디코더는 비트스트림을 디코딩하는 것을 진행할 수도 있다.

[0127] 제 1 예는 SEI 메니페스트 SEI 메시지 및 SEI 프리픽스 표시 SEI 메시지를 제공한다. 이들 SEI 메시지들 각각에 대한 신택스 및 시맨틱스가 이하에 제공된다.

[0128] **SEI 메니페스트 SEI 메시지**

[0129] SEI 메니페스트 SEI 메시지 신택스

sei_manifest(payloadSize) {	Descriptor
manifest_num_sei_msg_types	u(16)
for(i = 0; i < manifest_num_sei_msg_types; i++) {	
manifest_sei_payload_type[i]	u(16)
manifest_sei_description[i]	u(8)
}	
}	

[0130]

[0131] SEI 메니페스트 SEI 메시지 시맨틱스

[0132] SEI 메니페스트 SEI 메시지는 비트스트림의 수신자에 의해 존재하는 것으로 또는 존재하지 않는 것으로 가정될 수도 있는 SEI 메시지들에 대한 정보를 전달한다.

[0133] 1) 비트스트림의 수신자는 특정 타입들의 SEI 메시지들이 CVS 에 존재한다고 가정할 수도 있다.

[0134] 주의 1 - 사용되는 그러한 "가정한다" 라는 표현에 의해, 그것은 "약속들" 이 깨지는 경우 오류가 아닐 것이며, "약속들" 이 깨지는지 여부에 관계없이, 클라이언트 가정들은 여전히 올바른 것으로서 고려될 수 있다.

[0135] 2) CVS 에 존재하는 것으로 가정될 수도 있는 각각의 유형의 SEI 메시지의 경우, 이러한 유형의 SEI 메시지들의 필수성.

[0136] 필수성은 필수, 비필수, 또는 미결정으로서 표시될 수도 있다.

[0137] SEI 메시지는 반송된 정보가 컨텐츠를 적절하게 프로세싱하고 바람직한 사용자 경험을 가능하게 하기 위해 디코더측으로부터 필수인 것으로서 고려되는 경우 인코더 (즉, 컨텐츠 생성기) 에 의해 필수인 것으로서 고려된다. 어느 SEI 메시지들이 특정의 비트스트림에서 필수인 것으로서 고려되는지를 결정하는 것은 인코더의 재량이다. 그러나, 프레임 팩킹 배열, 세그멘팅된 직사각형 프레임 팩킹 배열, 디스플레이 배향, 및 전방향 투영 표시 SEI 메시지와 같은 일부 SEI 메시지들은 항상 필수적인 것으로서 고려되어야 한다.

[0138] 3) 비트스트림의 수신자는 특정의 유형들의 SEI 메시지들이 CVS 에 존재하지 않는다고 가정할 수도 있다.

[0139] 주의 2 - 예를 들어, 시그널링된 SEI 메니페스트 SEI 메시지에 기초하여, 비트스트림의 수신자는 비트스트림에 프레임 팩킹 배열 SEI 메시지, 세그멘팅된 직사각형 프레임 팩킹 배열 SEI 메시지, 디스플레이 배향 SEI 메시지, 또는 전방향 투영 표시 SEI 메시지가 존재하지 않는다고 가정할 수 있을 것이다. 따라서, 디코딩된 비디오 픽쳐들은 그러한 SEI 메시지들이 존재하는 경우 통상 요구되는 임의의 추가의 포스트 프로세싱의 필요 없이 뷰잉을 위해 렌더링될 수 있을 것이다.

[0140] SEI 메니페스트 SEI 메시지가 CVS 의 임의의 액세스 유닛에 존재하는 경우, SEI 메니페스트 SEI 메시지는 CVS 의 제 1 액세스 유닛에 존재할 것이다. SEI 메니페스트 SEI 메시지는 디코딩 순서에서 현재의 액세스 유닛으로부터 CVS 의 종단까지 지속된다. CVS 에 존재하는 다수의 SEI 메니페스트 SEI 메시지들이 있는 경우, 그들은 동일한 컨텐츠를 가질 것이다.

[0141] SEI 메니페스트 SEI 메시지를 포함하는 SEI NAL 유닛은 SEI 프리픽스 표시SEI 메시지들과는 임의의 다른 SEI 메시지를 포함하지 않을 것이다. SEI NAL 유닛에 존재하는 경우, SEI 메니페스트 SEI 메시지는 SEI NAL 유닛 내의 제 1 SEI 메시지일 것이다.

[0142] **manifest_num_sei_msg_types** 은 정보가 SEI 메니페스트 SEI 메시지에서 제공되는 SEI 메시지들의 유형들의 수를 특정한다.

[0143] **manifest_sei_payload_type**[i] 은 정보가 SEI 메니페스트 SEI 메시지에서 제공되는 SEI 메시지의 제 i 유형의 payloadType 값을 표시한다. manifest_sei_payload_type[m] 및 manifest_sei_payload_type[n] 의 값들은 m 이 n 과 동일하지 않은 경우 동일하지 않을 것이다.

[0144] **manifest_sei_description**[i] 은 표 2 에 특정된 바와 같은 manifest_sei_payload_type[i] 과 동일한 payloadType 을 갖는 SEI 메시지들에 관한 정보를 제공한다.

표 2

[0145] 표 2 - manifest_sei_description[i] 값들

값	디스크립션
0	CVS 에 존재하는 manifest_sei_payload_type[i] 와 동일한 payloadType 을 갖는 SEI 메시지가 존재하지 않는다고 가정될 수도 있다는 것을 표시
1	1) CVS 에 존재하는 manifest_sei_payload_type[i] 와 동일한 payloadType 을 갖는 SEI 메시지가 존재한다고 가정될 수도 있고, 2) 이들 SEI 메시지는 필수인 것으로 고려된다는 것을 표시.

2	1) CVS 에 존재하는 manifest_sei_payload_type[i] 와 동일한 payloadType 을 갖는 SEI 메시지가 존재한다고 가정될 수도 있고, 2) 이들 SEI 메시지는 비필수인 것으로 고려된다는 것을 표시.
3	1) CVS 에 존재하는 manifest_sei_payload_type[i] 와 동일한 payloadType 을 갖는 SEI 메시지가 존재한다고 가정될 수도 있고, 2) 이들 SEI 메시지의 필수성은 결정되지 않았음을 표시.
4-255	예약됨

[0146] manifest_sei_description[i] 의 값은 이러한 사양의 이러한 버전에 따르는 비트스트림들에서 (0 과 3 을 포함하여) 0 내지 3 의 범위에 있을 것이다. manifest_sei_description[i] 에 대한 다른 값들은 ITU-T | ISO/IEC 에 의한 장래의 사용을 위해 예약된다. 디코더들은 4 이상의 manifest_sei_description[i] 의 값이 신택스에서 나타내는 것을 허용할 것이고, SEI 메니페스트 SEI 메시지에서 시그널링된 manifest_sei_payload_type[i] 과 동일한 payloadType 에 대한 모든 정보를 무시할 것이고, manifest_sei_payload_type[i] 과 동일한 prefix_sei_payload_type 을 갖는 모든 SEI 프리픽스 표시 SEI 메시지들을 무시할 것이다.

[0147] **SEI 프리픽스 표시 SEI 메시지**

[0148] SEI 프리픽스 표시 SEI 메시지 신택스

[0149]

sei_prefix_indication(payloadSize) {	Descriptor
prefix_sei_payload_type	u(16)
num_sei_prefix_indications_minus1	u(8)
for(i = 0; i <= num_sei_prefix_indications_minus1; i++) {	
num_bits_in_prefix_indication_minus1[i]	u(16)
for(j = 0; j <= num_bits_in_prefix_indication_minus1[i]; j++)	
sei_prefix_databit[i][j]	u(1)
while(!byte_aligned( ))	
byte_alignment_bit_equal_to_zero /* equal to 0 */	f(1)
}	
}	

[0150] SEI 프리픽스 표시 SEI 메시지 시맨틱스

[0151] SEI 프리픽스 표시 SEI 메시지는 특정의 payloadType 의 SEI 메시지들에 대한 하나 이상의 SEI 프리픽스 표시들을 반송한다. 각 SEI 프리픽스 표시는 그러한 payloadType 의 SEI 페이로드 신택스에 후속하고 SEI 페이로드에서의 제 1 신택스 엘리먼트로부터 시작하는 다수의 완전한 신택스 엘리먼트들을 포함하는 비트 스트링이다.

[0152] 수신자는 CVS 에서의 이러한 payloadType 의 SEI 메시지들의 일부 또는 전부가 이들 비트 스트링들로 시작한다고 가정할 수도 있다. 시작 비트 스트링은 통상 그러한 유형의 SEI 메시지들의 SEI 페이로드의 진정한 서브세트만을 포함하고, 전체 SEI 페이로드를 포함할 수도 있지만, 전체 SEI 페이로드 초과를 결코 포함하지 않는다.

[0153] 주의 - 이러한 payloadType 의 일부 SEI 메시지들은 임의의 표시된 비트 스트링들로 시작하지 않는 것이 허용된다.

[0154] 이들 SEI 프리픽스 표시들은 어떤 타입의 프로세싱이 필요한지 또는 어떤 유형의 컨텐츠가 포함되는지를 표시하기 위한 충분한 정보를 제공해야 한다. 전자 (유형의 프로세싱) 는 디코더측 프로세싱 능력, 예를 들어 일부 타입의 프레임 언팩킹이 필요한지 여부를 표시한다. 후자 (유형의 컨텐츠) 는 예를 들어 비트스트림이 특정의 언어로 된 서브타이틀 캡션들을 포함하는지 여부를 나타낸다. 정보 시스템 계층에 기초하여, 엔티티들은 디코더측이 바람직한 사용자 경험을 가능하게 하기 위해 비트스트림을 적절하게 프로세싱할 수 있는지 여부, 또는 비트스트림이 애플리케이션 필요들을 만족시키는지 여부를 결정할 수 있다.

[0155] 하나의 예에서, payloadType 이 프레임 팩킹 배열 SEI 메시지를 나타내는 경우, SEI 프리픽스 표시는 적어도 신택스 엘리먼트 frame_packing_arrangement_type 까지 포함해야 한다; payloadType 이 전방향 투영 표시 SEI 메시지를 나타내는 경우, SEI 프리픽스 표시는 적어도 신택스 엘리먼트 projection_type 까지 포함해야 한다.

[0156] 다른 예에서, 캡셔닝 정보를 반송하는 데 사용되는 사용자 데이터 등록 SEI 메시지들의 경우, SEI 프리픽스 표시는 적어도 언어 코드까지 포함해야 하고; 사적인 사용을 위해 확장된 사용자 데이터 미등록 SEI 메시지들의

경우, SEI 프리픽스 표시는 적어도 UUID 까지 포함해야 한다.

[0157] SEI 프리픽스 표시 SEI 메시지가 CVS 의 임의의 액세스 유닛에 존재하는 경우, SEI 프리픽스 표시 SEI 메시지는 CVS 의 제 1 액세스 유닛에 존재할 것이다. SEI 프리픽스 표시 SEI 메시지는 디코딩 순서에서 현재의 액세스 유닛으로부터 CVS 의 종단까지 지속된다. CVS 에 존재하는 다수의 SEI 프리픽스 표시 SEI 메시지들이 있는 경우, 그들은 동일한 컨텐츠를 가질 것이다.

[0158] SEI 프리픽스 표시 SEI 메시지를 포함하는 SEI NAL 유닛은 SEI 메니페스트 SEI 메시지와는 임의의 다른 SEI 메시지를 포함하지 않을 것이다.

[0159] **prefix_sei_payload_type** 은 하나 이상의 SEI 프리픽스 표시들이 SEI 프리픽스 표시 SEI 메시지에서 제공되는 SEI 메시지들의 payloadType 값을 나타낸다. prefix_sei_payload_type 의 값은 manifest_sei_description[m] 이 CVS 에 적용되는 SEI 메니페스트 SEI 메시지에 의해 표시되는 바와 같이, (1 과 3 을 포함하는) 1 내지 3 과 동일한 manifest_sei_payload_type[m] 값들 중 하나와 동일할 것이다.

[0160] **num_sei_prefix_indications_minus1** 플러스 1 은 SEI 프리픽스 표시들의 수를 특정한다.

[0161] **num_bits_in_prefix_indication_minus1**[i] 플러스 1 은 제 i SEI 프리픽스 표시에서의 비트들의 수를 특정한다.

[0162] **sei_prefix_databit**[i][j] 는 제 i SEI 프리픽스 표시의 제 j 비트를 특정한다.

[0163] (0 과 num_bits_in_prefix_indication_minus1[i] 을 포함하여) 0 으로부터 num_bits_in_prefix_indication_minus1[i] 까지의 범위의 j 에 대한 비트들 sei_prefix_databit[i][j] 는 prefix_sei_payload_type 과 동일한 payloadType 을 갖는 SEI 페이로드의 신택스에 후속하고, SEI 페이로드 신택스에서의 제 1 신택스 엘리먼트로부터 시작하는 다수의 완전한 신택스 엘리먼트들을 포함하고, SEI 페이로드 신택스에서의 모든 신택스 엘리먼트들을 포함할 수도 있거나 포함하지 않을 수도 있다. 마지막 비트는 SEI 페이로드 신택스에서의 신택스 엘리먼트의 마지막 비트일 것이다.

[0164] **byte_alignment_bit_equal_to_zero** 는 0 과 동일할 것이다.

[0165] 선행의 예에서, 크기가 작을 수 있는 SEI 메니페스트 디스크립션들은 별개의 SEI 메시지에 존재할 수 있다. 이러한 SEI 메시지의 작은 크기는 SEI 메시지가 예를 들어 MIME 유형 파라미터로서와 같이, 시스템 레벨에서 반송되는 것을 가능하게 한다. 선행의 예는 또한 프리픽스 표시가 아직 정의되지 않은 SEI 유형들에 대한 프리픽스 표시들의 부가를 허용한다. 예를 들어, 새로운 SEI 메시지는 기존의 SEI 메시지를 변경하지 않고 부가될 수 있다. 또한, 다수의 SEI 메시지가 하나의 특정의 SEI 유형에 대한 SEI 프리픽스 표시들을 위해 필요하지 않다. 프리픽스 표시들이 이미 정의된 SEI 유형에 대한 프리픽스 표시들을 부가할 때, 기존 SEI 메시지에 대한 변경이 필요할 수도 있다.

[0166] 여러 예들에서, 대안적인 신택스가 사용될 수 있다.

[0167] 제 1 예에서, 2 개의 별개의 SEI 메시지들을 정의하는 대신에, SEI 메니페스트 디스크립션 및 SEI 프리픽스들은 하나의 SEI 메시지에서 반송될 수 있다. 그러한 SEI 메시지에 대한 예시의 신택스는 다음과 같다:

sei_manifest(payloadSize) {	Descriptor
manifest_num_sei_msg_types	u(16)
for(i = 0; i < manifest_num_sei_msg_types; i++) {	
manifest_sei_payload_type[i]	u(16)
manifest_sei_description[i]	u(8)
}	
for(i = 0; i < manifest_num_sei_msg_types; i++) {	
num_sei_prefix_indications[i]	u(8)
for(j = 0; j < num_sei_prefix_indications[i]; j++) {	
num_bits_in_prefix_indication[i][j]	u(16)
for(k = 0; k < num_bits_in_prefix_indication[i][j]; k++)	
sei_prefix_databit[i][j][k]	u(1)
while(!byte_aligned( ))	
byte_alignment_bit_equal_to_zero /* equal to 0 */	f(1)
}	
}	
}	

[0168]

[0169] 위의 예에서, (예를 들어, 모든 SEI 유형들에 대한 SEI 메니페스트 디스크립션들 및 SEI 프리픽스 표시들에 대한) 다수의 SEI 메시지들은 정의될 필요가 없다. 그러나, 추가의 정보가 필요한 경우, SEI 메시지는 변경될 필요가 있을 수도 있다.

[0170] 제 2 의 예에서, 하나의 SEI 프리픽스 표시 SEI 메시지에서 다수의 SEI 프리픽스 표시들을 허용하는 대신에, SEI 프리픽스 메시지는 하나의 SEI 프리픽스 표시만을 포함할 수 있다. 그러한 SEI 메시지에 대한 예시의 신택스는 다음과 같다:

sei_prefix_indication(payloadSize) {	Descriptor
prefix_sei_payload_type	u(16)
num_bits_in_prefix_indication_minus1	u(16)
for(i = 0; i <= num_bits_in_prefix_indication_minus1; i++)	
sei_prefix_databit[i]	u(1)
}	

[0171]

[0172] 위의 예에서, 기존 SEI 메시지는 프리픽스 표시들이 이미 정의된 SEI 유형에 대한 프리픽스 표시들을 부가할 때 변경될 필요가 없다. 그러나, 하나의 특정의 SEI 유형에 대한 SEI 프리픽스 표시들을 시그널링하기 위한 다수의 SEI 메시지들이 필요할 수도 있다.

[0173] 여러 예들에서, 대안적인 기법들이 SEI 메시지들의 지속성 범위를 정의하는데 사용될 수 있다.

[0174] 제 1 예에서, SEI 메시지들의 지속성 범위는 활성 SPS 의 그것과 동일할 수 있다. 이러한 예에서, 활성 SPS 가 다수의 CVS 들에 적용되는 경우, SEI 메시지의 지속성 범위는 또한 다수의 CVS 들일 것이다.

[0175] 제 2 예에서, 지속성 범위는 비트스트림에서 특정될 수 있다. 이러한 예에서, SEI 메시지들은 다수의 비트스트림들이 하나의 비트스트림으로 스플라이싱될 때 변경될 필요가 있을 수도 있다.

[0176] 여러 예들에서, 시맨틱스들에 대한 대안적인 표현이 사용될 수 있다.

[0177] 제 1 예에서, “특정 유형들의 SEI 메시지들이 존재하거나 부재하는지 여부를 표시하다” 와 같은 적극적인 표현은 예를 들어 SEI 메니페스트에 대한 시맨틱스에서 사용될 수 있다.

[0178] 제 2 의 예로서, “특정 유형들의 SEI 메시지들이 존재하는 것으로 또는 부재하는 것으로 가정될 수도 있는지 여부를 표시하다” 와 같은, (디코더의 관점에서 본) 비적극적인 표현이 사용될 수 있다.

[0179] 제 3 예로서, “특정 유형들의 SEI 메시지들이 존재하는 것으로 또는 부재하는 것으로 인코더에 의해 진술되는지 여부를 표시하다” 와 같은, (인코더의 관점에서 본) 비적극적인 표현이 사용될 수 있다. 이러한 예에서, 이러한 정보를 프로세싱하는 방법은 디코더에게 달려있을 수 있지만, 디코더가 이러한 정보를 프로세싱하는 방법에 관해 결정할 수도 있다는 것이 추가로 진술될 수 있을 것이다.

[0180] 위의 제 1 예의 경우, SEI 메니페스트가 특정의 유형의 SEI 메시지가 비트스트림에 존재한다고 표시할 수도 있는 인스턴스들이 존재할 수도 있지만, 그러한 유형의 SEI 메시지들은 제거되었을 수도 있다. 이들 예에서, 비트스트림의 시맨틱스는 올바르지 않고, 비트스트림은 넌-컨포밍 (non-conforming) 일 수도 있다.

[0181] 위의 제 2 및 제 3 예들에 의해, 비트스트림 내의 SEI 메시지들의 제거는비트스트림이 넌-컨포밍이 되게 하지 않는다. 예를 들어, SEI 메니페스트가 특정 유형의 SEI 메시지가 비트스트림에 존재하지만 SEI 메시지가 (비트스트림이 인코딩된 후) 제거되었다고 표시하는 경우, 제 2 및 제 3 예들의 비적극적인 표현은 부정확하게 되지 않을 것이다. 제 3 의 예에서, 비적극적인 표현은 제 2 예의 비적극적인 표현보다 덜 절박하다.

[0182] **필수 및 비필수 SEI 메시지들**

[0183] 여러 SEI 메시지들이 예를 들어 HEVC 사양의 부록 D 에서, (JCTVC-Z1005 에서의 드래프트 보정 텍스트와 같은) HEVC 사양에 대한 드래프트 보정에서, 및 그 사양의 부록 F 및 후속 부록들에서 정의되었다.

[0184] 필수 정보를 포함하는 것으로서 고려될 수도 있는 SEI 메시지들의 예들이 이하에 제공된다. 이들 SEI 메시지들은, 존재하는 (예를 들어, 본래 존재하는 또는 네스팅 SEI 메시지에 포함되는) 경우, 디코딩된 픽처들의 렌더링에 직접적인 영향을 줄 수도 있다. 이들 SEI 메시지들은 필수인 것으로 고려될 수도 있지만, 이러한 정보를 포함하는 SEI 메시지들이 필수인 것으로 라벨링되는지 여부는 인코더에게 달려있다.

[0185] - 프레임 팩킹 배열 SEI 메시지

[0186] - 디스플레이 배향 SEI 메시지

[0187] - 세그멘팅된 직사각형 프레임 팩킹 배열 SEI 메시지

[0188] - 전방향 투영 표시 SEI 메시지

[0189] 다음의 SEI 메시지들은, 존재하는 (예를 들어, 본래 존재하는 또는 네스팅 SEI 메시지에 포함되는) 경우, 디코딩된 픽처들의 렌더링에 영향을 줄 수도 있다. 이들 SEI 메시지들은 또한 필수인 것으로 고려될 수 있는 SEI 메시지들의 예들이다.

[0190] - 필름 그레인 특성 SEI 메시지

[0191] - 포스트-필터 힌트 SEI 메시지

[0192] - 톤 맵핑 정보 SEI 메시지

[0193] - 무 디스플레이 SEI 메시지

[0194] - 무 디스플레이 SEI 메시지

[0195] - 마스터링 디스플레이 컬러 볼륨 SEI 메시지

[0196] - 크로마 리샘플링 필터 힌트 SEI 메시지

[0197] - 무릎 함수 정보 SEI 메시지

[0198] - 컬러 리샘플링 정보 SEI 메시지

[0199] - 디인터레이싱된 픽처 정보 SEI 메시지

[0200] - 컨텐츠 광 레벨 정보 SEI 메시지

[0201] - 대안적인 전달 특징 SEI 메시지

[0202] - 주변 뷰잉 환경 SEI 메시지

[0203] - 컨텐츠 컬러 볼륨 SEI 메시지

[0204] 디코딩된 픽처의 렌더링에 영향을 줄 가능성이 더 적은 다른 SEI 메시지들은 인코더의 재량으로 필수 SEI 메시지들의 리스트 내로 포함될 수도 있거나 포함되지 않을 수도 있다. 예를 들어, 인코더는 추천 ITU-T T.35 SEI 메시지에 의해 등록된 사용자 데이터 또는 사용자 데이터 미등록 SEI 메시지가 필수 정보를 반송한다고 결정하고 결과적으로 그러한 SEI 페이로드 유형을 필수 SEI 메시지들의 리스트 내로 포함시키고 요구된 프로세싱 능력을 충분히 표시하기 위해 일부 추가 정보를 제공할 수도 있다.

[0205] **필수 보충 정보의 시스템 레벨 노출**

[0206] 여러 구현들에서, 상술된 것들과 같은 SEI 메시지들 (예를 들어, SEI 를 반송하는 필수 정보로서 집합적으로 여기에서 지칭될 ESI SEI, NSI SEI, SEI 메니페스트 SEI, 및 SEI 프리픽스 SEI) 은 디코더가 비디오 세그먼트를 디코딩하는 프로세스에서 조기에 이들 SEI 를 액세스할 수 있도록 캡슐화 포맷들에 배치될 수 있다. 캡슐화 포맷들은 예를 들어 파일 포맷들, 스트리밍 포맷들, 네트워크 포맷들, 및 다른 포맷들을 포함한다.

[0207] 파일 포맷 표준들은 비디오 (및 가능하게는 또한 오디오) 데이터를 하나 이상의 파일들 내로 팩킹 및 언팩킹하기 위한 포맷을 정의할 수 있다. 파일 포맷 표준들은 (ISO/IEC 14496-15 에서 정의된) 동화상 전문가 그룹 (MPEG) MPEG-4 파일 포맷, (3GPP TS 26.244 에서 정의된) 3 세대 파트너쉽 프로젝트 (3GPP) 파일 포맷 및 (양자 모두 ISO/IEC 14496-15 에서 정의된) 진보된 비디오 코딩 (AVC) 파일 포맷 및 고효율 비디오 코딩 (HEVC) 파일 포맷을 포함하여, (ISO/IEC 14496-12 에서 정의된) 표준화를 위한 국제 기구 (ISO) 베이스 미디어 파일 포맷 (ISOBMFF) 및 ISOBMFF 로부터 도출된 다른 파일 포맷들을 포함한다. ISO/IEC 14496-12 및 14496-15 에 대한 최근의 새로운 에디션들의 드래프트 텍스트들은 각각 http://phenix.int-evry.fr/mpeg/doc_end_user/documents/111_Geneva/wg11/w15177-v6-w15177.zip 및 http://phenix.int-evry.fr/mpeg/doc_end_user/documents/112_Warsaw/wg11/w15479-v2-w15479.zip 에서 이용가능하다.

[0208] ISOBMFF 는 다수의 멀티미디어 컨테이너 포맷들 (예를 들어, MPEG-4 파일 포맷, 3GPP 파일 포맷 (3GPP), 디지털 비디오 브로드캐스팅 (DVB) 파일 포맷, 또는 임의의 다른 적절한 멀티미디어 컨테이너 포맷) 에 대해서 뿐아니라 다수의 코덱 캡슐화 포맷 (예를 들어, AVC 파일 포맷 또는 임의의 다른 적절한 코덱 캡슐화 포맷) 에 대한 기초로서 사용된다. ISOBMFF 기반 파일 포맷들은 스트리밍 미디어로서도 지칭되는 연속 미디어를 위해 사용될 수 있다.

[0209] 연속 미디어 (예를 들어, 오디오 및 비디오) 에 더하여, 정적 미디어 (예를 들어, 이미지들) 및 메타데이터가 ISOBMFF 에 따르는 파일에 저장될 수 있다. ISOBMFF 에 따라 구조화된 파일들은 로컬 미디어 파일 재생, 원격 파일의 점진적 다운로딩를 포함하는 다수의 목적을 위해, HTTP 를 통한 동적 적응형 스트리밍 (DASH) 을 위한 세그먼트들로서, 스트리밍될 컨텐츠를 위한 컨테이너로서 (이 경우, 컨테이너는 패킷화 명령들을 포함함), 수신된 실시간 미디어 스트림들의 기록을 위해, 또는 다른 사용들을 위해 사용될 수도 있다.

[0210] ISOBMFF 및 그것의 도출된 파일 포맷들 (예를들어, AVC 파일 포맷 또는 다른 도출된 파일 포맷들) 은 다수의 멀티미디어 애플리케이션들에서 (예를 들어, 비디오, 오디오, 및 타이밍된 텍스트를 포함하는) 미디어 컨텐츠의 저장 및 캡슐화를 위해 널리 사용된다. 여러 구현들에서, ISOBMFF 파일 포맷은 필수 정보가 적용되는 비디오 데이터에 앞서 SEI 메시지를 반송하는 필수 정보를 제공하는데 사용될 수 있다. 예를 들어, SEI 를 반송하는 필수 정보는 이하에 더 논의되는 바와 같이, 샘플 엔트리에 포함될 수 있다.

[0211] 도 3 은 ISOBMFF 에 따라 포맷팅된, 비디오 제시를 위한 데이터 및 메타데이터를 포함하는 ISO 베이스 미디어 파일 (300) 의 예를 도시한다. ISOBMFF 는 미디어의 교환, 관리, 편집, 및 제시를 용이하게 하는 유연하고 확장가능한 포맷으로 타이밍된 미디어 정보를 포함하도록 설계된다. 미디어의 제시는 그 제시를 포함하는 시스템에 대해 “로컬” 일 수도 있거나 그 제시는 네트워크 또는 다른 스트림 전달 메커니즘을 통할 수도 있다.

[0212] ISOBMFF 사양에 의해 정의된 바와 같은 “제시” 는 종종 비디오 캡쳐 디바이스에 의해 순차적으로 캡쳐된 것에 의해 관련된, 또는 일부 다른 이유로 관련된 픽쳐들의 시퀀스이다. 여기서, 제시는 또한 무비 또는 비디오 제시로서 지칭될 수도 있다. 제시는 오디오를 포함할 수도 있다. 단일의 제시는 하나 이상의 파일들에 포함될 수도 있으며, 하나의 파일은 전체 제시를 위한 메타데이터를 포함한다. 메타데이터는 타이밍 및 프레이밍 데이터, 디스크립터들, 포인터들, 파라미터들, 및 제시를 기술하는 다른 정보와 같은 정보를 포함한다. 메타데이터는 비디오 및/또는 오디오 데이터 자체를 포함하지 않는다. 메타데이터를 포함하는 파일 이외의 파일은 ISOBMFF 에 따라 포맷팅될 필요가 없고, 단지 이들 파일들이 메타데이터에 의해 참조될 수 있도록 포맷팅될 필요가 있다.

[0213] ISO 베이스 미디어 파일의 파일 구조는 객체 지향이고, 파일 내의 개개의 객체의 구조는 객체의 유형으로부터 직접 추론될 수 있다. ISO 베이스 미디어 파일 내의 객체들은 ISOBMFF 사양에 의해 “박스들” 로서 지칭된다. ISO 베이스 미디어 파일은 다른 박스들을 포함할 수 있는 박스들의 시퀀스로서 구조화된다. 박스들은 일반적으로 박스의 크기 및 유형을 제공하는 헤더를 포함한다. 크기는 헤더, 필드들, 및 박스 내에 포함된 모든 박스들을 포함하여, 박스의 전체 크기를 기술한다. 재생 디바이스에 의해 인식되지 않는 유형을 갖는 박스들은 통상 무시되고 스킵된다.

[0214] 도 3 의 예에 의해 도시된 바와 같이, 파일의 상부 레벨에서, ISO 베이스 미디어 파일 (300) 은 파일 유형 박스

(310), 무비 박스 (320), 및 하나 이상의 무비 단편들 (330a, 330n) 을 포함할 수 있다. 이러한 레벨에서 포함될 수 있는 그러나 이러한 예에서 나타내지 않는 다른 박스들은 무엇보다도 자유 공간 박스들, 메타데이터 박스들, 및 미디어 데이터 박스들을 포함한다.

[0215] ISO 베이스 미디어 파일은 박스 타입 “ftyp” 에 의해 식별된 파일 유형 박스 (310) 를 포함할 수 있다. 파일 유형 박스 (310) 는 파일을 파싱하기 위해 가장 적절한 ISOBMFF 사양을 식별한다. 이러한 예에서 “가장” 은 ISO 베이스 미디어 파일 (300) 이 특정의 ISOBMFF 사양에 따라 포맷팅되었을 수도 있지만, 그 사양의 다른 신판들과 아마도 호환가능하다는 것을 의미한다. 이러한 가장 적절한 사양은 주요 브랜드로서 지칭된다. 재생 디바이스는 디바이스가 파일의 컨텐츠를 디코딩 및 디스플레이할 수 있는지 여부를 결정하기 위해 그 주요 브랜드를 사용할 수 있다. 파일 유형 박스 (310) 는 또한 ISOBMFF 사양의 버전을 표시하기 위해 사용될 수 있는 버전 번호를 포함할 수 있다. 파일 유형 박스 (310) 는 또한 파일이 호환가능한 다른 브랜드들의 리스트를 포함하는 호환가능 브랜드들의 리스트를 포함할 수 있다. ISO 베이스 미디어 파일은 2 이상의 주요 브랜드와 호환가능할 수 있다.

[0216] ISO 베이스 미디어 파일 (300) 이 파일 유형 박스 (310) 를 포함할 때 하나의 파일 유형 박스만이 존재한다. ISO 베이스 미디어 파일 (300) 은 더 오래된 플레이어 디바이스와 호환가능하기 위하여, 파일 유형 박스 (310) 를 생략할 수도 있다. ISO 베이스 미디어 파일 (300) 이 파일 유형 박스 (310) 를 포함하지 않을 때 재생 디바이스는 디폴트 메이저 브랜드 (예를 들어, “mp41”), 마이너 버전 (예를 들어, "0") 및 호환가능한 브랜드 (예를 들어, “mp41”) 를 가정할 수 있다. 파일 유형 박스 (310) 는 전형적으로 ISO 베이스 미디어 파일 (300) 에 가능한 일찍 배치된다.

[0217] ISO 베이스 미디어 파일은 추가로, 제시에 대한 메타데이터를 포함하는 무비 박스 (320) 를 포함할 수 있다. 무비 박스 (320) 는 박스 유형 "moov" 에 의해 식별된다. ISO/IEC 14496-12 는 제시가, 하나의 파일 또는 다수의 파일들에 포함되든지 간에, 오직 하나의 무비 박스 (320) 를 포함할 수 있는 것을 규정한다. 빈번하게, 무비 박스 (320) 는 ISO 베이스 미디어 파일의 시작부 부근에 있다. 무비 박스 (320) 는 무비 헤더 박스 (322) 를 포함하고, 하나 이상의 트랙 박스들 (324) 뿐만 아니라 다른 박스들을 포함할 수 있다.

[0218] 박스 타입 "mvhd" 에 의해 식별되는 무비 헤더 박스 (322) 는 미디어-독립적이고 전체적으로 제시에 관련된 정보를 포함할 수 있다. 예를 들어, 무비 헤더 박스 (322) 는, 무엇보다도, 생성 시간, 수정 시간, 타임스케일, 및/또는 프리젠테이션을 위한 지속기간과 같은 정보를 포함할 수 있다. 무비 헤더 박스 (322) 는 또한, 제시에서 다음 트랙을 식별하는 식별자를 포함할 수 있다. 예를 들어, 식별자는 예시된 예에서 무비 박스 (320) 에 의해 포함된 트랙 박스 (324) 에 포인팅할 수 있다.

[0219] 박스 타입 "trak" 에 의해 식별되는 트랙 박스 (324) 는 제시를 위한 트랙에 대한 정보를 포함할 수 있다. 제시는 하나 이상의 트랙들을 포함할 수 있고, 여기서, 각각의 트랙은 제시에서 다른 트랙들에 독립적이다. 각 트랙은 트랙에서의 컨텐츠에 특정적인 시간 및 공간 정보를 포함할 수 있고, 각 트랙은 미디어 박스와 연관될 수 있다. 트랙에서의 데이터는 미디어 데이터일 수 있고, 이 경우에, 트랜은 미디어 트랙이고, 또는, 데이터는 스트리밍 프로토콜들을 위한 패킷화 정보일 수 있고, 이 경우에 트랙은 힌트 트랙이다. 미디어 데이터는 예를 들어 비디오 및 오디오 데이터를 포함한다. 도시된 예에서, 예시적인 트랙 박스 (324) 는 트랙 헤더 박스 (324a) 및 미디어 박스 (324b) 를 포함한다. 트랙 박스는 트랙 참조 박스, 트랙 그룹 박스, 편집 박스, 사용자 데이터 박스, 메타 박스, 및 기타와 같은 다른 박스들을 포함할 수 있다.

[0220] 박스 타입 "tkhd" 에 의해 식별되는 트랙 헤더 박스 (324a) 는 트랙 박스 (324) 에 포함된 트랙의 특성들을 특정할 수 있다. 예를 들어, 트랙 헤더 박스 (324a) 는, 무엇보다도 생성 시간, 수정 시간, 지속기간, 트랙 식별자, 계층 식별자, 그룹 식별자, 트랙의 볼륨, 폭, 및/또는 높이를 포함할 수 있다. 미디어 트랙에 대해, 트랙 헤더 박스 (324a) 는 추가적으로, 다른 것들도 있지만, 트랙이 인에이블되는지 여부, 트랙이 제시의 일부로서 플레이되어야 하는지 여부, 또는 트랙이 제시를 프리뷰하기 위해 사용될 수 있는지 여부를 식별할 수 있다. 트랙의 제시는 일반적으로 제시의 시작부에 있는 것으로 가정된다. 트랙 박스 (324) 는 명시적 타임라인 맵을 포함할 수 있는, 여기서 나타내지 않은, 편집 리스트 박스를 포함할 수 있다. 타임라인은, 무엇보다도 트랙에 대한 오프셋 시간을 명시할 수 있고, 여기서, 오프셋은 트랙에 대한, 제시의 시작 후의 시작 시간을 표시한다.

[0221] 나타낸 예에서, 트랙 박스 (324) 는 또한, 박스 타입 "mdia" 에 의해 식별된 미디어 박스 (324b) 를 포함한다. 미디어 박스 (324b) 는 트랙 내에서 미디어 데이터에 관한 정보 및 오브젝트들을 포함할 수 있다. 예를 들어, 미디어 박스 (324b) 는 핸들러 참조 박스를 포함할 수 있고, 이 핸들러 참조 박스는 트랙의 미디어 타입, 및 트랙에서의 미디어가 제시되는 프로세스를 식별할 수 있다. 다른 예로서, 미디어 박스 (324b) 는 트랙에서의 미디

어의 특성들을 명시할 수 있는 미디어 정보 박스를 포함할 수 있다. 미디어 정보 박스는 샘플들의 테이블을 추가로 포함할 수 있고, 여기서, 각 샘플은 예를 들어 그 샘플에 대한 데이터의 로케이션을 포함하는 미디어 데이터 (예컨대, 비디오 또는 오디오 데이터) 의 청크 (chunk) 를 기술한다. 샘플에 대한 데이터는 이하에서 추가로 논의되는 미디어 데이터 박스에 저장된다. 대부분의 다른 박스들과 같이, 미디어 박스 (324b) 는 미디어 헤더 박스를 또한 포함할 수 있다.

[0222] 예시된 예에서, 예시적인 ISO 기본 미디어 파일 (300) 은 또한 제시의 다수의 단편들 (330a, 330b, 330c, 330n) 을 포함한다. 단편들 (330a, 330b, 330c, 330n) 은 ISOBMFF 박스가 아니며, 오히려 무비 단편 박스 (332) 및 무비 단편 박스 (332) 에 의해 참조되는 미디어 데이터 박스 (338) 를 기술한다. 무비 프래그먼트 박스 (332) 및 미디어 데이터 박스들 (338) 은 최상위-레벨 박스들이지만, 여기서 무비 프래그먼트 박스 (332) 와 미디어 데이터 박스 (338) 사이의 관계를 나타내기 위해 그룹핑된다.

[0223] 박스 타입 "moof"에 의해 식별되는 무비 프래그먼트 박스 (332) 는 무비 박스 (320) 에 달리 저장된 추가 정보를 포함함으로써 프리젠테이션을 확장할 수 있다. 무비 프래그먼트 박스 (332) 를 사용하여, 프리젠테이션이 점진적으로 구축될 수 있다. 무비 프래그먼트 박스 (332) 는 여기에 예시되지 않은 다른 박스뿐만 아니라 무비 프래그먼트 헤더 박스 (334) 및 트랙 프래그먼트 박스 (336) 를 포함할 수 있다.

[0224] 박스 타입 "mfhd" 에 의해 식별되는 무비 프래그먼트 박스 (334) 는 시퀀스 넘버를 포함할 수 있다. 재생 디바이스는 프래그먼트 (330a) 가 프리젠테이션을 위한 데이터의 다음 피스 (piece) 를 포함하는 것을 확인하기 위해 시퀀스 넘버를 사용할 수 있다. 일부 경우들에서, 프리젠테이션을 위한 파일 또는 파일들의 컨텐츠들은 순서 외로 플레이어 디바이스에 제공될 수 있다. 예를 들어, 네트워크 패킷들은 패킷들이 원래 송신되었던 순서와는 다른 순서로 빈번하게 도달할 수 있다. 이들 경우들에서, 시퀀스 넘버는 프래그먼트들에 대한 정확한 순서를 결정함에 있어서 플레이어 디바이스를 도울 수 있다.

[0225] 무비 프래그먼트 박스 (332) 는 또한, 박스 타입 "traf" 에 의해 식별되는 하나 이상의 트랙 프래그먼트 박스들 (336) 을 포함할 수 있다. 무비 프래그먼트 박스 (332) 는 트랙 당 제로 또는 그보다 많은 트랙 프래그먼트들의 세트를 포함할 수 있다. 트랙 프래그먼트들은, 제로 또는 그보다 많은 트랙 런들 (runs) 을 포함할 수 있고, 이 트랙 런들의 각각은 트랙에 대한 샘플들의 인접한 런을 기술한다. 트랙 프래그먼트들은 트랙에 샘플들을 부가하는 것에 추가하여 트랙에 엠프티 타임 (empty time) 을 부가하기 위해 사용될 수 있다.

[0226] 박스 타입 "mdat" 에 의해 식별되는 미디어 데이터 박스 (338) 는 미디어 데이터를 포함한다. 비디오 트랙들에서, 미디어 데이터 박스 (338) 는 비디오 프레임들을 포함할 것이다. 미디어 데이터 박스는 대안적으로 또는 추가적으로 오디오 데이터를 포함할 수 있다. 프리젠테이션은 하나 이상의 개별 파일들에 포함된 하나 이상의 미디어 데이터 박스들을 포함할 수 있다. 미디어 데이터는 메타데이터에 의해 기술된다. 도시된 예에서, 미디어 데이터 박스 (338) 에서의 미디어 데이터는 트랙 프래그먼트 박스 (336) 에 포함된 메타데이터에 의해 기술될 수 있다. 다른 예들에서, 미디어 데이터 박스에서의 미디어 데이터는 무비 박스 (320) 에서의 메타데이터에 의해 기술될 수 있다. 메타데이터는 파일 (300) 내의 절대 오프셋에 의한 특정 미디어 데이터를 가리킬 수 있어서, 미디어 데이터 박스 (338) 내의 미디어 데이터 헤더 및/또는 자유 공간은 스킵될 수 있다.

[0227] ISO 베이스 미디어 파일 (300) 에서의 다른 프래그먼트들 (330b, 330c, 330n) 은 제 1 프래그먼트 (330a) 에 대해 예시된 것들과 유사한 박스들을 포함할 수 있고, 및/또는, 다른 박스들을 포함할 수 있다.

[0228] ISOBMFF 는 미디어의 로컬 재생을 지원하는 것에 더하여, 네트워크를 통한 미디어 데이터의 스트리밍에 대한 지원을 포함한다. 하나의 무비 프리젠테이션을 포함하는 파일 또는 파일들은 힌트 트랙이라 불리는 추가적인 트랙들을 포함할 수 있고, 이들은 패킷들로서 파일 또는 파일들을 형성하고 송신하는데 있어 스트리밍 서버을 보조할 수 있는 명령들을 포함한다. 이들 명령들은, 예를 들어, 서버가 전송할 데이터 (예를 들어, 헤더 정보) 및 미디어 데이터의 세그먼트들에 대한 레퍼런스들을 포함할 수 있다. 파일은 상이한 스트리밍 프로토콜들에 대해 별개의 힌트 트랙들을 포함할 수 있다. 힌트 트랙들은 또한, 파일을 재포맷팅할 필요 없이 파일에 부가될 수 있다.

[0229] 도 3 에 예시된 예시의 미디어 파일 (300) 에서, SEI 를 운반하는 필수 정보는 예를 들어 샘플 입력에 포함될 수있다. 예를 들어, SEI 메시지를 운반하는 필수 정보를 포함하는 SEI NAL 유닛은 HEVC 샘플 엔트리의 디코더 구성 레코드에서 SEI NAL 유닛의 어레이에 직접 포함될 수있다. 일부 경우에, SEI 메시지를 운반하는 필수 정보를 포함하는 SEI NAL 유닛은 디코더 구성 레코드에서 SEI NAL 유닛 어레이의 제 1 SEI NAL 유닛으로서 포함될 수있다. 다양한 예들에서, SEI 메시지들을 운반하는 필수 정보는 대안 적으로 또는 부가 적으로 다른 것들 중에

서 무비 헤더 박스 (322) 또는 트랙 헤더 박스 (324a) 와 같은 상위 레벨 박스들에서 운반 될 수있다.

[0230] DASH와 같은 스트리밍 캡슐화 형식은 또한 SEI 메시지를 반송하는 필수 정보를 디코더에 제시하는데 사용될 수 있다. ISO/IEC 23009-1 에 명시된 DASH 는 HTTP (적응형) 스트리밍 애플리케이션들에 대한 표준이다. DASH는 매니페스트라고도하는 MPD (Media Presentation Description)라고하는 포맷과 미디어 세그먼트 포맷을 지정한다. MPD 는 서버에서 사용 가능한 미디어를 설명하고 DASH 클라이언트가 미디어에서 특정 시간에 특정 미디어 버전을 자율적으로 다운로드하는 것을 가능하게 한다.

[0231] DASH 기반 HTTP 스트리밍은 다음 단계를 포함할 수 있다.

[0232] 1) 클라이언트가 스트리밍 컨텐츠 (예를 들어, 무비) 의 MPD 를 얻는다. MPD 에는 예를 들어 스트리밍 콘텐츠의 비트 레이트, 비디오 해상도, 프레임 속도, 오디오 언어 뿐아니라 HTTP 리소스 (예를 들어, 초기화 세그먼트 및 미디어 세그먼트) 의 URL 과 같은 상이한 대안적인 표현에 대한 정보가 포함된다.

[0233] 2) MPD 에서의 정보와 클라이언트의 로컬 정보, (예를 들어 네트워크 대역폭, 디코딩/디스플레이 능력, 및/또는 사용자 선호) 에 기초하여, 클라이언트는 한 번에 하나의 세그먼트 (또는 그 일부) 씩, 원하는 리프리젠테이션(들) 을 요청한다.

[0234] 3) 클라이언트가 네트워크 대역폭 변경을 검출하면, 그것은 랜덤 액세스 포인트로 시작하는 세그먼트부터 이상적으로 시작하는, 더 나은 매칭 비트레이트를 가진 상이한 표현의 세그먼트를 요청한다.

[0235] HTTP 스트리밍 세션 동안, 과거 위치로 뒤로 또는 미래 위치로 앞으로 탐색하라는 사용자 요청에 응답하기 위해, 클라이언트는 원하는 위치에 가까운 세그먼트부터 시작하여 과거 또는 미래의 세그먼트를 요청한다. 이 시점부터 클라이언트는 랜덤 액세스 지점에서 시작할 수 있습니다. 사용자는 또한 인트라 코딩된 비디오 픽쳐만을 또는 비디오 스트림의 시간 서브 세트만을 디코딩하기 위해 충분히 데이터를 요구함으로써 실현될 수도 있는 콘텐츠를 고속 포워딩할 것을 요청할 수도 있다.

[0236] 도 4 는 비디오 컨텐츠를 스트리밍하기 위한 DASH 제시 (400) 의 일 예를 예시한다. 비디오 컨텐츠는 연속 비디오 스트림을 기술 할 수있는 미디어 프리젠테이션 (402)에 포함된다. 미디어 프리젠테이션 (402)은 매니페스트라고도하는 미디어 프리젠테이션 디스크립션 (MPD)을 포함 할 수있다. MPD는 예를 들어 XML (eXtensible Markup Language)을 사용하여 포맷된 문서이며, 미디어 프리젠 테이션 (402)의 미디어 세그먼트에 관한 정보를 포함한다. 이 정보는 예를 들어 세그먼트 간의 관계 및 세그먼트 사이에서 선택하는 데 사용될 수있는 정보를 포함 할 수 있다. MPD는 또한 클라이언트 디바이스 (예를 들어, 컨텐츠를 수신하는 디바이스) 가 사용할 수있는 다른 데이터를 포함 할 수있다.

[0237] 미디어 프리젠 테이션 (402)은 기간들 (404a-404c)으로 분할 될 수있다. DASH에 의해 정의 된 기간은 미디어 프리젠 테이션 (402) 내의 시간 간격이다. 따라서 프리젠 테이션은 연속적인 시퀀스의 기간들 (404a-404c) 로 구성된다. 기간 (404a-404c) 은 예를 들어 시작 시간에 의해 설명 될 수 있으며, 여기서 시작 시간은 기간 내의 시간 간격이 시작되는 시간을 나타낸다. 달리 말하면, 시작 시간은 시간 0으로부터의 오프셋이다.

[0238] 기간 (404b) (예를 들어, 도시된 예에서 기간 2) 내에서, 미디어 컨텐츠는 예를 들어 평균 비트레이트, 언어, 캡션 설정, 자막 설정 등을 포함하는 인코딩들의 일관된 세트를 가질 수있다. 기간 (404b) 또한 콘텐츠가 스트리밍 될 수있는 소스 (예를 들어, 베이스 URL (Uniform Resource Locator))를 제공 할 수있다. 기간 (404b) 의 컨텐츠는 적응 세트 (424, 426) 로 배열 될 수있다. 적응 세트는 하나 이상의 미디어 컨텐츠 컴포넌트의 교환 가능한 인코딩 된 버전들의 세트를 나타낸다. 예를 들어, 기간은 메인 비디오 컴포넌트에 대한 하나의 적응 세트 (424) 및 메인 오디오 컴포넌트에 대한 별도의 적응 세트 (426)를 포함 할 수있다. 캡션 또는 오디오 디스프립트들과 같은 사용 가능한 다른 컨텐츠가 있는 경우, 이들 각각은 별도의 적응 세트를 가질 수 있다.

[0239] 적응 세트 (예를 들어, 비디오 적응 세트 (424)) 는 다수의 대체 표현 (432)을 포함 할 수있다. 표현은 하나 또는 수 개의 미디어 컨텐츠 컴포넌트들의 전달 가능한 인코딩된 버전을 기술한다. 적응 세트 내의 임의의 단일 표현은 기간 내에 미디어 컨텐츠 컴포넌트를 렌더링하는데 사용될 수있다. 전송 된 하나의 적응에서의 상이한 표현은 지각적으로 동등한 것으로 간주 될 수 있는데, 이는 클라이언트 디바이스가 네트워크 조건 또는 다른 요인에 적응하기 위해 적응 세트 내의 하나의 표현으로부터 다른 표현으로 동적으로 전환 할 수 있음을 의미한다. 예를 들어, 각각의 표현은 특정 대역폭, 프레임 높이 및 폭뿐만 아니라 프레임 레이트 또는 인코딩 유형과 같은 다른 정보를 가질 수있다. 표현 (432) 은 미디어 프리젠 테이션 (402)에서 세그먼트를 설명하는 세그먼트 정보 (442)를 더 포함 할 수있다.

[0240] 세그먼트 정보 (442)는 초기화 세그먼트 (444) 및 하나 이상의 미디어 세그먼트 (446a-446c)를 기술 할 수있다. 초기화 세그먼트 (444)는 미디어 자체에 선행하는 일부 컨텐츠를 포함 할 수있다. 각각의 미디어 세그먼트 (446a-446c)는 전체 기간 (404b)의 일부를 포함한다. 미디어 세그먼트들 (446a-446c)은 기간 (404b)의 시작에 대한 시작 시간 및 소스 위치 (예를 들어, URL)에 의해 설명 될 수있다.

[0241] ISOBMFF는 AVC 및 HEVC 및 그들의 확장들에 대한 MIME 유형 'codecs' 파라미터에 대한 하위 파라미터를 지정한다. DASH를 사용하여 컨텐츠를 스트리밍 할 때, MIME 유형 파라미터는 MPD에서 속성 (예를 들어, @mimeType)으로 제공 될 수 있다. SEI 메시지를 전달하는 필수 정보를 포함하는 SEI NAL 유닛은 'codecs' 파라미터 또는 다른 MIME 유형 파라미터의 일부로 포함될 수 있으며, 따라서 @mimeType 속성을 통해 MPD에서 이러한 SEI 메시지가 추출가능하게 할 수 있다.

[0242] MPEG-TS (MPEG transport stream) 는 인코딩 된 비트 스트림이 네트워크를 통해 전송 될 수있는 캡슐화 포맷이다. FIG. 도 5 는 MPEG 전송 스트림 (500) 의 예를 도시한다. MPEG-TS는 오디오, 비디오 및 프로그래밍 데이터를 전송하고 저장하는 데 사용되는 컨테이너 포맷이다. MPEG-TS는 신호가 열화 될 때 송신 무결성을 유지하기 위한 오류 수정 및 스트림 동기화 특징들과 함께 패킷화된 기본 스트림을 캡슐화하는 컨테이너 포맷을 지정한다. MPEG-TS는 지상파 및 위성 방송과 같은 신뢰할 수없는 전송 매체를 통해 스트리밍하도록 의도된다.

[0243] MPEG 전송 스트림 (500)은 다수의 전송 패킷 (502)을 포함한다. 패킷은 전송 스트림에서 기본 데이터 단위입니다. 전송 스트림 자체는 임의의 전역 헤더가 없는 패킷들의 시퀀스이다. 각각의 패킷 (502) 은 헤더 (504) 및 페이로드 (506)를 포함한다. 헤더 (504)는 예를 들어 페이로드의 컨텐츠, 다른 패킷에 대한 패킷 시퀀스, 에러 정보 등을 기술하는 것을 포함하여 패킷 (502)에 관한 정보를 제공한다.

[0244] 다른 필드들 중에서, 헤더 (504)는 패킷 식별자 (508) (PID)를 포함 할 수있다. 패킷 식별자 (508)는 페이로드 (506)에 포함 된 데이터를 기술 할 수있다. 예를 들어, PID = 2는 페이로드 (506)가 전송 스트림 디스크립터 테이블 (TSDT)의 일부 또는 전부를 포함 함을 나타낸다. 전송 스트림 디스크립터 테이블은 전송 스트림 (500)과 관련된 디스크립터 (510)를 포함 할 수있다.

[0245] 일부 예들에서, SEI 메시지들을 운반하는 필수 정보는 유사한 디스크립터로서 비디오 코덱 프로파일 및 레벨 정보를 포함하는 디스크립터의 일부로서 포함될 수있다. 이들 디스크립터에 포함 됨으로써, 디코더는 필수 디코딩 정보와 함께 SEI 메시지를 운반하는 필수 정보를 파싱 할 수있다.

[0246] 일부 예시적인 애플리케이션은 미디어 전송을 위해 RTP (Real-time Transfer Protocol) 를 및 미디어에 대한 정보를 시그널링하기 위해 SDP (Session Description Protocol)를 사용하고 있을 수도 있다. SDP 는 멀티미디어 통신 세션을 기술하는데 사용될 수 있다. 이러한 설명은 예를 들어 세션 공지, 세션 초대 및 파라미터 협상에 사용될 수있다. SDP는 미디어 자체를 전달하는 데 사용되지 않지만 미디어 유형, 포맷 및 관련 속성을 협상하기 위해 종단점 간에서 사용될 수 있다. 특성들 및 파라미터들의 세트는 종종 세션 프로파일이라고 한다. SDP는 원래 SAP (Session Announcement Protocol)의 구성 요소 였지만 RTP (Real-time Transfer Protocol), RTSP (Real-time Streaming Protocol), SIP (Session Initiation Protocol) 및 멀티 캐스트 세션을 설명하기 위한 독립형 포맷과 관련하여 다른 용도를 발견했다. SDP는 RFC 4566에 설명되어 있다.

[0247] 도 6 은 네트워크 통신을 위한 OSI (Open Systems Interconnect) 모델 (600)의 예를 도시한다. 이 모델 내에서, SDP는 세션 계층 (610)에서 구현되며, 연결 관리, 오류 복구, 보안, 원격 작동 및 기타 기능이 발생한다. 세션 속성 (620)은 세션 계층 (610)에서 교환 될 수있다.

[0248] OSI 모델 (600) 은 통신 기능을 표준화하고 컴플라이언스는 다양한 통신 시스템 간의 상호 운용성을 가능하게 한다. 모델 (600) 의 각 계층은 상부의 계층을 서빙하고, 하부의 계층에 의해 서빙된다. 물리 계층 (602) 은 물리 매체를 통한 원시 비트 스트림의 전송 및 수신을 정의한다. 데이터 링크 계층 (604)은 두 노드 사이에서 데이터 프레임의 안정적인 전송을 정의한다. 물리 어드레싱과 같은 동작은 데이터 링크 계층 (604)에서 처리된다. 네트워크 계층 (606)은 어드레싱, 라우팅 및 트래픽 제어를 포함하는 다중 노드 네트워크의 구조 및 관리를 정의한다. 패킷 단편화 및 논리 어드레싱과 같은 동작은 네트워크 계층 (606)에서 발생한다. 전송 계층 (608)은 세그먼트화, 확인 응답 및 멀티플렉싱을 포함하여 네트워크상의 포인트들 사이에서 데이터 세그먼트의 안정적인 송신을 정의한다. 엔드-투-엔드 연결 관리, 메시지 세그멘트화, 메시지 시퀀싱, 신뢰성 및 흐름 제어와 같은 동작들은 전송 계층 (608)에서 발생할 수있다. 세션 계층 (610)은 세션들의 관리를 정의하는데, 세션들은 두 노드 사이에서의 다수의 연속 송신들의 형태의 정보의 지속적 교환들이다. 전술 한 바와 같이, 이전에, 연결 관리, 에러 복구, 보안 및 원격 동작과 같은 동작이 세션 계층 (610)에서 발생할 수있다. 프리젠 테이션 계층 (612)은

문자 인코딩, 데이터 압축, 및 암호화 및/또는 암호 해독을 포함하여 네트워킹 서비스와 애플리케이션 간의 데이터 변환을 정의한다. 애플리케이션 계층 (614)은 리소스 공유, 원격 파일 액세스 및 다른 동작을 포함하는 고급 애플리케이션 프로그래밍 인터페이스 (API) 를 포함한다.

[0249] 일부 예에서, SEI 메시지를 운반하는 필수 정보는 SDP 파일에 SDP 속성으로서 포함될 수있다. 아래는 세션 디스크립션의 필드들의 예이다. 선택적 필드는 "= *" 로 지정되며 아래 예에서 SEI 메시지를 전달하는 필수 정보는 세션 속성 라인 및/또는 미디어 속성 라인을 사용하여 지정될 수 있다.

[0250] 세션 디스크립션

[0251] v = (프로토콜 버전 번호, 현재는 단지 0)

[0252] o = (발생지 및 세션 식별자 : 사용자 이름, 아이디, 버전 번호, 네트워크 주소)

[0253] s = (세션 이름 : 적어도 하나의 UTF-8 인코딩 문자가 필수)

[0254] i = * (세션 제목 또는 간단한 정보)

[0255] u = * (디스크립션의 URI)

[0256] e = * (선택적인 연락처 이름을 가진 0 개 이상의 이메일 주소)

[0257] p = * (선택적인 연락처 이름을 가진 0 개 이상의 전화 번호)

[0258] c = * (연결 정보 — 모든 미디어에 포함 된 경우 필요하지 않음)

[0259] b = * (0 개 이상의 대역폭 정보 라인)

[0260] *하나 이상의 시간 디스크립션들 ("t =" 및 "r =" 라인들; 아래 참조)*

[0261] z = * (시간대 조정)

[0262] k = * (암호화 키)

[0263] a = * (0 개 이상의 세션 속성 라인들)

[0264] *0 개 이상의 미디어 디스크립션들 (각각의 것은 "m =" 라인으로 시작; 아래 참조)*

[0265] 시간 디스크립션

[0266] t = (세션이 활성화인 시간)

[0267] r = * (0 회 이상의 반복 시간)

[0268] 미디어 디스크립션

[0269] m = (미디어 이름 및 전송 주소)

[0270] i = * (미디어 제목 또는 정보 필드)

[0271] c = * (연결 정보-세션 수준에 포함된 경우 선택 사항)

[0272] b = * (제로 이상의 대역폭 정보 라인)

[0273] k = * (암호화 키)

[0274] a = * (0 개 이상의 미디어 속성 라인들-세션 속성 라인들을 오버라이딩함)

[0275] 도 7 은 비디오 데이터를 인코딩하기 위한 프로세스 (700) 의 일 예를 예시한다. 방법은 여기에 설명 된 것과 같은 비디오 코딩 시스템에 의해 구현 될 수있다.

[0276] 702 에서, 프로세스 (700) 는 비디오 데이터를 수신하는 것을 포함한다. 비디오 데이터는 예를 들어 비디오 코딩 시스템의 비디오 캡처 디바이스로부터 수신될 수 있다. 다른 예로서, 비디오 데이터는 하드 드라이브 또는 플래시 메모리 장치와 같은 저장 장치 및/또는 네트워크를 통해 수신 될 수있다.

[0277] 704에서, 프로세스 (700)는 인코딩 데이터에 의해 비디오 데이터와 관련된 정보가 비디오 데이터를 제시하는데 필요한지 여부를 결정하는 것을 포함하며, 여기서 정보는 비디오 데이터를 디코딩하는데 필요하지 않다. 인코딩 디바이스는 비디오 코딩 시스템의 일부일 수있다. 비디오 데이터가 수신되는 동시에 정보가 수신 될 수있다. 대

안적으로, 정보는 인코딩 디바이스가 비디오 데이터를 수신하기 전에 인코딩 디바이스에 제공 될 수있다. 예를 들어, 인코딩 디바이스는 정보와 함께 미리 구성 될 수있다. 대안적으로, 정보는 비디오 데이터의 구성 요소 일 수있다.

[0278] 다양한 예들에서, 디코딩 디바이스에 의해 사용될 때, 정보는 디코딩 디바이스가 비디오 데이터를 특정 방식으로 처리 및/또는 디스플레이할 수 있게 할 수 있다. 예를 들어, 정보는 프레임 패킹 배열을 포함 할 수 있으며, 이는 프레임이 인코딩되기 전에 픽쳐가 프레임으로 패킹된 방식을 표시한다. 다른 예로서, 정보는 비디오 데이터가 제시 될 디스플레이의 원하는 배향 (예를 들어, 회전)을 나타내는 디스플레이 배향을 포함 할 수있다. 다른 예로서, 정보는 프레임으로 패킹되기 전에 픽쳐가 세그먼트화되었다는 것을 나타낼 수 있는 세그먼트화된 직사각형 프레임 패킹 배열, 및 프레임이 인코딩되기 전에 세그먼트들이 프레임으로 패킹된 방식을 포함한다. 다른 예로서, 정보는 비디오 데이터가 360도 비디오 데이터임을 디코딩 디바이스에 표시할 수 있는 전방향 투영 표시를 포함할 수 있다. 이들 및 다른 예에서, 일부 경우에, 디코딩 디바이스는 정보를 사용하여 디코딩 디바이스가 비디오 데이터를 디스플레이 할 수 있는지 여부를 판정할 수있다.

[0279] 다양한 예들에서, 인코딩 장치는 정보가 비디오 데이터를 제시하는데 필요한지 여부를 결정하기 위해 인코딩 장치가 사용할 수있는 하나 이상의 파라미터들로 구성된다. 이들 파라미터는 무엇보다도 예를 들어, 비디오 데이터에 적용하기 위한 필름 그레인 특성 또는 톤 매핑, 비디오 데이터를 생성하는데 사용되는 컬러 볼륨 또는 컬러 매핑, 및/또는 프레임 패킹 구성을 포함 할 수있다.

[0280] 706에서, 프로세스 (700)는 보충 강화 정보 (SEI) 메시지를 생성하는 것을 포함하고, SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 필요한지 여부를 나타내는 필드를 포함한다. 예를 들어, SEI 메시지는 비디오 데이터에 포함 된 SEI 메시지의 유형을 식별하는 필드 및 이들 SEI 메시지 각각의 필수성을 나타내는 필드를 포함할 수 있는 SEI 매니페스트 메시지일 수있다. 다른 예로서, SEI 메시지는 필수 정보를 포함 할 수있는 프리픽스 SEI 메시지 일 수있다. 다른 예로서, SEI 메시지는 ESI SEI 메시지일 수 있다. 다른 예로서, SEI 메시지는 NSI SEI 메시지일 수 있다.

[0281] 일부 예에서, SEI 메시지의 신택스 요소는 SEI 메시지의 유형을 나타낸다. 이러한 예에서, 디코딩 장치는 신택스 엘리먼트를 사용하여 SEI 메시지를 표 1 에 정의된 것과 같은 SEI 메시지와 다른 것으로 식별 할 수있다.

[0282] 708 에서, 프로세스 (700) 는 비디오 데이터를 인코딩하는 것을 포함한다. 비디오 데이터는, 예를 들어, AVC 또는 HEVC 표준, 또는 다른 비디오 코딩 표준을 사용하여 인코딩될 수 있다. 다양한 예들에서, 비디오 데이터를 인코딩하는 것은 인코딩 된 비디오 비트 스트림을 생성 할 수 있고, 인코딩 된 비디오 비트 스트림은 비디오 데이터의 압축 된 버전을 포함한다.

[0283] 710에서, 프로세스 (700)는 인코딩 된 비디오 데이터와 함께 SEI 메시지를 포함시키는 것을 포함한다. 일부 예들에서, SEI 메시지는 인코딩 된 비디오 데이터에 포함될 수있다. 일부 예들에서, SEI 메시지는 인코딩된 비디오 데이터와 함께 캡슐화된 포맷으로 포함될 수있다.

[0284] 예를 들어, 프로세스 (700)는 인코딩 된 비디오 데이터 및 SEI 메시지를 파일에 기입하는 것을 포함 할 수 있고, 인코딩 된 비디오 데이터 및 SEI 메시지는 파일 포맷에 따라 파일에 기입된다. 이들 예들에서, 프로세스 (700)는 인코딩 된 비디오 데이터가 판독되기 전에 SEI가 판독 될 수있는 파일에 SEI 메시지를 배치하기 위해 파일 포맷을 사용하는 것을 더 포함 할 수있다.

[0285] 다른 예로서, 프로세스 (700)는 스트리밍을 위해 인코딩 된 비디오 데이터를 캡슐화하는 것을 포함 할 수 있고, 인코딩 된 비디오 데이터는 스트리밍 포맷에 따라 캡슐화된다. 이 예에서, 프로세스 (700)는 스트리밍 포맷에 따라 인코딩 된 비디오 데이터의 디스크립션을 생성하는 것을 더 포함 할 수 있으며, 여기서 디스크립션은 SEI 메시지를 포함한다.

[0286] 다른 예로서, 프로세스 (700)는 네트워크를 통한 송신을 위해 인코딩 된 비디오 데이터를 캡슐화하는 것을 포함할 수 있고, 인코딩 된 비디오 데이터는 컨테이너 포맷에 따라 캡슐화된다. 이 예에서, 프로세스 (700)는 컨테이너 포맷에 따라 제 1 패킷을 생성하는 것을 더 포함할 수 있으며, 제 1 패킷의 페이로드 부분은 디스크립터로서 SEI 메시지를 포함한다. 프로세스 (700)는 인코딩 된 비디오 데이터를 포함하는 하나 이상의 패킷을 생성하는 것을 더 포함 할 수 있으며, 하나 이상의 패킷은 전송 순서로 제 1 패킷을 따른다.

[0287] 다른 예로서, 프로세스 (700)는 네트워크를 통한 송신을 위해 인코딩 된 비디오 데이터를 캡슐화하는 것을 더 포함 할 수 있고, 인코딩 된 비디오 데이터는 실시간 전송 프로토콜에 따라 캡슐화된다. 이 예에서, 프로세스 (700)는 세션 디스크립션 프로토콜에 따라 세션 디스크립션을 생성하는 것을 더 포함 할 수 있으며, 여기서 세

션 디스크립션은 SEI 메시지를 포함한다.

[0288] 일부 예들에서, 인코딩 디바이스는 정보가 비디오 데이터를 제시하기 위해 필요하다고 결정한다. 이들 예에서, SEI 메시지의 필드는 필수 정보가 인코딩 된 비디오 데이터에 포함되거나 인코딩 된 비디오 데이터와 함께 제공됨을 나타낼 수있다. 이러한 예에서, 디코딩 장치는 SEI 메시지를 사용하여 필수 정보를 식별하고 로케이팅할 수있다.

[0289] 일부 예들에서, 인코딩 디바이스는 정보가 비디오 데이터를 제시하기 위해 필요하다고 결정한다. 이들 예에서, SEI 메시지의 필드는 필수 정보가 인코딩 된 비디오 데이터에 포함되거나 인코딩 된 비디오 데이터와 함께 제공되지 않음을 나타낼 수있다.

[0290] 일부 예들에서, 정보는 인코딩 된 비디오 데이터와 함께 포함된다. 예를 들어, 정보는 SEI 메시지에 포함될 수 있다. 대안적으로 또는 추가적으로, 정보는 하나 이상의 추가 SEI 메시지에 포함될 수있다. 일부 예들에서, 추가 SEI 메시지들은 각각의 SEI 메시지가 운반하는 유형 정보를 식별 할 수있다. 일부 예에서, 추가의 SEI 메시지 각각은 각각의 SEI 메시지에 의해 운반되는 정보의 필수성을 나타낼 수있다. 일부 예들에서, 각각의 SEI 메시지의 필수성은 위에서 논의 된 바와 같이 생성된 SEI 메시지에서 표시된다.

[0291] 도 8 은 비디오 데이터를 디코딩하기 위한 프로세스 (800) 의 일 예를 예시한다. 프로세스 (800) 는 예를 들어 여기에 설명 된 것과 같은 비디오 코딩 디바이스에 의해 구현 될 수있다.

[0292] 802에서, 프로세스 (800)는 인코딩 된 비디오 비트 스트림, 보충 강화 정보 (SEI) 메시지, 및 인코딩 된 비디오 비트 스트림과 관련된 정보를 디코딩 장치에서 수신하는 것을 포함한다. 일부 예들에서, 디코딩 장치는 비디오 코딩 디바이스의 일부이다. 일부 예들에서, 디코딩 장치는 저장 디바이스로부터 및/또는 네트워크를 통해 인코딩 된 비디오 비트 스트림을 수신한다. 일부 예들에서, 디코딩 디바이스는 인코딩 디바이스로부터 인코딩된 비디오 비트 스트림을 수신한다.

[0293] 일부 예들에서, SEI 메시지 및/또는 인코딩 된 비디오 비트 스트림과 관련된 정보는 인코딩된 비디오 비트 스트림에 포함된다. 일부 예들에서, SEI 메시지 및/또는 정보는 인코딩된 비디오 비트 스트림과 함께, 예를 들어, 인코딩된 비디오 비트스트림을 포함하는 캡슐화된 포맷에서 제공된다.

[0294] 예를 들어, 인코딩 된 비디오 비트 스트림, SEI 메시지 및 인코딩 된 비디오 비트 스트림과 관련된 정보는 파일로 수신 될 수 있으며, 여기서 파일은 파일 포맷에 따라 포맷된다. 이 예에서, 파일 포맷은 SEI 메시지가 인코딩 된 비디오 비트 스트림을 포함하는 파일의 일부에 앞서 디코딩 장치에 의해 판독 된 파일의 일부에 있음을 제공 할 수있다.

[0295] 다른 예로서, 인코딩 된 비디오 비트 스트림, SEI 메시지, 및 인코딩 된 비디오 비트 스트림과 관련된 정보는 데이터 스트림으로 수신 될 수 있으며, 여기서 데이터 스트림은 스트리밍 포맷에 따라 포맷된다. 이 예에서, 디코딩 디바이스는 데이터 스트림에 따라 인코딩 된 비디오 데이터의 디스크립션을 판독할 수 있으며, 여기서 디스크립션은 SEI 메시지를 포함한다.

[0296] 다른 예로서, 인코딩 된 비디오 비트 스트림, SEI 메시지, 및 인코딩 된 비디오 비트 스트림과 관련된 정보는 복수의 네트워크 패킷들에서 수신 될 수 있다. 이 예에서, 디코딩 장치는 복수의 네트워크 패킷으로부터 제 1 패킷의 페이로드 부분으로부터 SEI 메시지를 판독 할 수있다. 예를 들어, SEI 메시지는 페이로드에서 디스크립터로서 제공 될 수있다. 이 예에서, 디코딩 장치는 복수의 네트워크 패킷으로부터의 하나 이상의 패킷으로부터 인코딩 된 비디오 데이터를 추가로 판독 할 수 있으며, 여기서 하나 이상의 패킷은 송신 순서로 제 1 패킷을 따른다.

[0297] 다른 예로서, 인코딩 된 비디오 비트 스트림, SEI 메시지, 및 인코딩 된 비디오 비트 스트림과 관련된 정보는 네트워크를 통해 수신 될 수 있다. 이 예에서, 인코딩 된 비디오 비트 스트림은 실시간 전송 프로토콜에 따라 캡슐화 될 수 있고, SEI 메시지는 세션 디스크립션에 포함될 수 있으며, 세션 디스크립션은 세션 디스크립션 프로토콜에 따라 포맷된다.

[0298] 804에서, 프로세스 (800)는 SEI 메시지를 사용하여, 정보가 인코딩 된 비디오 비트 스트림에서의 비디오 데이터를 제시하는데 필요한지 여부를 결정하는 것을 포함하며, 여기서 정보는 비디오 데이터를 디코딩하는데 필요하지 않으며, SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 필요한지 여부를 나타내는 필드를 포함한다.

[0299] 일부 예에서, SEI 메시지의 신택스 요소는 SEI 메시지의 유형을 나타낸다. 이들 예들에서, 디코딩 디바이스는 SEI 메시지가 정보가 비디오 데이터를 제시하는 데 필요하지 여부의 표시를 포함할 수 있다는 것을 결정하기 위

해 신택스 엘리먼트를 사용할 수있다.

[0300] 일부 예들에서, 정보는 SEI 메시지에, 및/또는 인코딩 된 비디오 데이터와 함께 포함된 하나 이상의 추가 SEI 메시지에 포함된다. 예를 들어, SEI 메시지는 인코딩 된 비디오 데이터와 함께 포함 된 SEI 메시지들의 유형들의 리스팅을 포함 할 수있다. 이 예에서, 리스팅은 각 SEI 메시지에 포함 된 데이터의 필수성을 추가로 나타낼 수있다. 다른 예로서, SEI 메시지 유형은 SEI 메시지가 필수 정보를 포함하는지 또는 포함하지 않는지 여부를 나타낼 수있다.

[0301] 806에서, 프로세스 (800)는 비디오 데이터를 생성하기 위해 인코딩 된 비디오 비트 스트림을 디코딩하는 것을 포함하고, 인코딩 된 비디오 비트 스트림은 비디오 데이터를 제시하기 위해 정보가 필요한지를 결정하는 것의 결과에 따라 디코딩된다.

[0302] 일부 예들에서, 디코딩 디바이스는 정보가 비디오 데이터를 제시하기 위해 필요하다고 결정한다. 이들 예에서, 디코딩 장치는 정보에 따라 비디오 데이터를 구성한다. 예를 들어, 인코딩 된 비디오 데이터를 디코딩 할 때, 디코딩 디바이스는 프레임 패킹 포맷에 따라 비디오 프레임의 컨텐츠를 재배열 할 수있다. 다른 예로서, 디코딩 장치는 360도 디스플레이를 사용하여 디스플레이 할 비디오 프레임을 배열 할 수있다. 다른 예로서, 디코딩 장치는 컬러 정보를 디스플레이 장치에 제공 할 수 있고, 여기서 비디오 데이터는 컬러 정보를 사용하여 디스플레이 될 수있다.

[0303] 일부 예들에서, 디코딩 디바이스는 정보가 비디오 데이터를 제시하기 위해 필요하다고 결정한다. 이들 예에서, 디코딩 장치는 정보를 사용하지 않고 인코딩된 비디오 데이터를 디코딩할 수 있다.

[0304] 일부 예에서, 프로세스 (700, 800) 는 컴퓨팅 디바이스 또는 장치, 이를테면 비디오 코딩 시스템 (100) 에 의해 수행될 수도 있다. 예를 들어, 프로세스들 (700, 800) 는 도 1 에 도시된 비디오 코딩 시스템 (100) 에 의해 수행될 수 있다. 일부 경우들에서, 컴퓨팅 디바이스 또는 장치는 프로세스 (700, 800) 의 단계들을 수행하도록 구성되는 디바이스의 프로세서, 마이크로프로세서, 마이크로컴퓨터, 또는 다른 컴포넌트를 포함할 수도 있다. 일부 예에서, 컴퓨팅 디바이스 또는 장치는 비디오 프레임들을 포함하는 비디오 데이터 (예를 들어, 비디오 시퀀스) 를 캡처하도록 구성된 카메라를 포함할 수도 있다. 예를 들어, 컴퓨팅 디바이스는 비디오 코덱을 포함할 수도 있는 카메라 디바이스 (예를 들어, IP 카메라 또는 다른 유형의 카메라 디바이스) 를 포함할 수도 있다. 일부 예에서, 비디오 데이터를 캡처하는 카메라 또는 다른 캡처 디바이스는 컴퓨팅 디바이스로부터 분리되어 있으며, 이 경우 컴퓨팅 디바이스는 캡처된 비디오 데이터를 수신한다. 컴퓨팅 디바이스는 비디오 데이터를 통신하도록 구성된 네트워크 인터페이스를 더 포함할 수도 있다. 네트워크 인터페이스는 인터넷 프로토콜 (IP) 기반 데이터를 통신하도록 구성될 수도 있다.

[0305] 프로세스 (700, 800) 는 논리 흐름도로서 나타나 있고, 그의 동작들은 하드웨어, 컴퓨터 명령 또는 이들의 조합으로 구현될 수 있는 동작들의 시퀀스를 표현한다. 컴퓨터 명령들의 맥락에서, 그 동작들은 하나 이상의 프로세서들에 의해 실행될 때 열거된 동작들을 수행하는, 하나 이상의 컴퓨터 판독 가능 저장 매체에 저장된 컴퓨터 실행 가능 명령들을 나타낸다. 일반적으로, 컴퓨터 실행 가능 명령들은 특정 기능들을 수행하거나 또는 특정 데이터 유형을 구현하는 루틴, 프로그램, 오브젝트, 컴포넌트, 데이터 구조 등을 포함한다. 동작들이 기재되는 순서는 제한으로서 해석되게 의도되지 않으며, 기재된 동작들의 임의의 수는 임의의 순서 및/또는 병렬로 조합되어 프로세스를 구현할 수 있다.

[0306] 추가적으로, 프로세스들 (700 및 800) 은 실행가능 명령들로 구성된 하나 이상의 컴퓨터 시스템들의 제어 하에서 수행될 수도 있고, 집합적으로 하나 이상의 프로세서 상에서 실행하는 코드 (예를 들어, 실행가능 명령들, 하나 이상의 컴퓨터 프로그램들, 또는 하나 이상의 애플리케이션들) 로서, 하드웨어에 의해, 또는 이들의 조합으로 구현될 수도 있다. 상기 언급된 바와 같이, 코드는 컴퓨터 판독가능 또는 머신 판독가능 저장 매체 상에, 예를 들어, 하나 이상의 프로세서들에 의해 실행가능한 복수의 명령들을 포함하는 컴퓨터 프로그램의 형태로 저장될 수도 있다. 컴퓨터 판독가능 또는 머신 판독가능 저장 매체는 비일시적일 수도 있다.

[0307] 본 명세서에서 논의되는 방법들 및 동작들은 압축된 비디오를 사용하여 구현될 수도 있고, 예시적인 비디오 인코딩 및 디코딩 시스템(예컨대, 시스템(100))에서 구현될 수도 있다. 일부 예들에서, 시스템이 목적지 디바이스에 의해 나중에 디코딩될 인코딩된 비디오 데이터를 제공하는 소스 디바이스를 포함한다. 특히, 소스 디바이스는 비디오 데이터를 목적지 디바이스에 컴퓨터 판독가능 매체를 통해 제공한다. 소스 디바이스 및 목적지 디바이스는 데스크탑 컴퓨터들, 노트북 (즉, 랩탑) 컴퓨터들, 태블릿 컴퓨터들, 셋탑 박스들, 소위 "스마트" 폰들과 같은 전화기 핸드셋들, 소위 "스마트" 패드들, 텔레비전들, 카메라들, 디스플레이 디바이스들, 디지털 미디어

플레이어들, 비디오 게이밍 콘솔들, 비디오 스트리밍 디바이스 등을 포함한 광범위한 디바이스들 중 임의의 디바이스를 포함할 수도 있다. 일부 경우들에 있어서, 소스 디바이스 및 목적지 디바이스는 무선 통신을 위해 장비될 수도 있다.

[0308] 목적지 디바이스는 디코딩될 인코딩된 비디오 데이터를 컴퓨터 판독가능 매체를 통해 수신할 수도 있다. 컴퓨터 판독가능 매체는 인코딩된 비디오 데이터를 소스 디바이스로부터 목적지 디바이스로 이동시킬 수 있는 임의의 타입의 매체 또는 디바이스를 포함할 수도 있다. 일 예에 있어서, 컴퓨터 판독가능 매체는, 소스 디바이스로 하여금 인코딩된 비디오 데이터를 직접 목적지 디바이스로 실시간으로 송신할 수 있게 하는 통신 매체를 포함할 수도 있다. 인코딩된 비디오 데이터는 무선 통신 프로토콜과 같은 통신 표준에 따라 변조되고, 목적지 디바이스로 송신될 수도 있다. 통신 매체는 무선 주파수 (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 부분을 형성할 수도 있다. 통신 매체는 라우터들, 스위치들, 기지국들, 또는 소스 디바이스로부터 목적지 디바이스로의 통신을 용이하게 하는데 유용할 수도 있는 임의의 다른 장비를 포함할 수도 있다.

[0309] 일부 예들에 있어서, 인코딩된 데이터는 출력 인터페이스로부터 저장 디바이스로 출력될 수도 있다. 유사하게, 인코딩된 데이터는 입력 인터페이스에 의해 저장 디바이스로부터 액세스될 수도 있다. 저장 디바이스는 하드 드라이브, 블루-레이 디스크들, DVD들, CD-ROM들, 플래시 메모리, 휘발성 또는 비휘발성 메모리, 또는 인코딩된 비디오 데이터를 저장하기 위한 임의의 다른 적합한 디지털 저장 매체들과 같은 다양한 분산된 또는 국부적으로 액세스된 데이터 저장 매체들 중 임의의 데이터 저장 매체를 포함할 수도 있다. 추가의 예에 있어서, 저장 디바이스는, 소스 디바이스에 의해 생성된 인코딩된 비디오를 저장할 수도 있는 파일 서버 또는 다른 중간 저장 디바이스에 대응할 수도 있다. 목적지 디바이스는, 스트리밍 또는 다운로드를 통해 저장 디바이스로부터 저장된 비디오 데이터에 액세스할 수도 있다. 파일 서버는, 인코딩된 비디오 데이터를 저장하고 그리고 그 인코딩된 비디오 데이터를 목적지 디바이스에 송신하는 것이 가능한 임의의 타입의 서버일 수도 있다. 예시적인 파일 서버들은 웹 서버 (예를 들어, 웹 사이트용), FTP 서버, 네트워크 접속형 저장 (NAS) 디바이스들, 또는 로컬 디스크 드라이브를 포함한다. 목적지 디바이스는 인코딩된 비디오 데이터에, 인터넷 커넥션을 포함한 임의의 표준 데이터 커넥션을 통해 액세스할 수도 있다. 이는 파일 서버 상에 저장되는 인코딩된 비디오 데이터에 액세스하기에 적합한 무선 채널 (예를 들어, Wi-Fi 커넥션), 유선 커넥션 (예를 들어, DSL, 케이블 모뎀 등), 또는 이들 양자의 조합을 포함할 수도 있다. 인코딩된 비디오 데이터의 저장 디바이스로부터의 송신은 스트리밍 송신, 다운로드 송신, 또는 이들의 조합일 수도 있다.

[0310] 본 개시의 기법들은 무선 어플리케이션들 또는 설정들로 반드시 한정되는 것은 아니다. 그 기법들은, 공중 경유 (over-the-air) 텔레비전 브로드캐스트, 케이블 텔레비전 송신, 위성 텔레비전 송신, DASH (dynamic adaptive streaming over HTTP) 와 같은 인터넷 스트리밍 비디오 송신, 데이터 저장 매체 상에 인코딩되는 디지털 비디오, 데이터 저장 매체 상에 저장된 디지털 비디오의 디코딩, 또는 다른 애플리케이션 등의 다양한 멀티미디어 애플리케이션들 중 어느 것을 지원하는 비디오 코딩에 적용될 수도 있다. 일부 예들에서, 시스템은, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅 및/또는 픽처 통화등의 애플리케이션들을 지원하기 위하여 일방향 또는 바이래터럴 비디오 송신을 지원하도록 구성될 수도 있다.

[0311] 하나의 예에서, 소스 디바이스는 비디오 소스, 비디오 인코더, 및 출력 인터페이스를 포함한다. 목적지 디바이스는 입력 인터페이스, 비디오 디코더, 및 디스플레이 디바이스를 포함할 수도 있다. 소스 디바이스의 비디오 인코더는 본 명세서에서 개시된 기법들을 적용하도록 구성될 수도 있다. 다른 예들에서, 소스 디바이스와 목적지 디바이스가 다른 컴포넌트들 또는 배열체들을 포함할 수도 있다. 예를 들어, 소스 디바이스는 외부 카메라와 같은 외부 비디오 소스로부터 비디오 데이터를 수신할 수도 있다. 유사하게, 목적지 디바이스는 통합된 디스플레이 디바이스를 포함하는 것보다는 외부 디스플레이 디바이스와 인터페이싱할 수도 있다.

[0312] 예시의 시스템은 단지 일 예일 뿐이다. 비디오 데이터를 병렬로 프로세싱하기 위한 기술들은 임의의 디지털 비디오 인코딩 및/또는 디코딩 디바이스에 의해 수행될 수도 있다. 비록 일반적으로 본 개시의 기술들이 비디오 인코딩 디바이스에 의해 수행되지만, 그 기술들은 또한, 통상적으로 "CODEC" 로서 지칭되는 비디오 인코더/디코더에 의해 수행될 수도 있다. 또한, 본 개시의 기술들은 비디오 프리프로세서에 의해 수행 될 수도 있다. 소스 디바이스 및 목적지 디바이스는, 단지, 소스 디바이스가 목적지 디바이스로의 송신을 위한 코딩된 비디오 데이터를 생성하는 그러한 코딩 디바이스들의 예들일 뿐이다. 일부 예들에 있어서, 소스 및 목적지 디바이스들은, 디바이스들 각각이 비디오 인코딩 및 디코딩 컴포넌트들을 포함하도록 실질적으로 대칭적인 방식으로 동작할 수도 있다. 따라서, 예시의 시스템들은 예를 들어, 비디오 스트리밍, 비디오 플레이백, 비디오 브로드캐스팅, 또

는 비디오 전화를 위해, 비디오 디바이스들 간의 일방향 또는 양방향 비디오 송신을 지원할 수도 있다.

[0313] 비디오 소스는 비디오 카메라와 같은 비디오 캡처 디바이스, 이전에 캡처된 비디오를 포함하는 비디오 아카이브, 및/또는 비디오 콘텐츠 제공자로부터 비디오를 수신하기 위한 비디오 피드 인터페이스를 포함할 수도 있다. 추가적인 대안으로서, 비디오 소스는 컴퓨터 그래픽 기반 데이터를 소스 비디오로서, 또는 라이브 비디오, 아카이브된 비디오, 및 컴퓨터 생성된 비디오의 조합으로서 생성할 수도 있다. 일부 경우들에 있어서, 비디오 소스가 비디오 카메라이면, 소스 디바이스 및 목적지 디바이스는 소위 카메라 폰들 또는 비디오 폰들을 형성할 수도 있다. 하지만, 상기 언급된 바와 같이, 본 개시에서 설명된 기술들은 일반적으로 비디오 코딩에 적용가능할 수도 있으며, 무선 및/또는 유선 어플리케이션들에 적용될 수도 있다. 각각의 경우에 있어서, 캡처되거나 사전-캡처되거나 또는 컴퓨터 생성된 비디오는 비디오 인코더에 의해 인코딩될 수도 있다. 그 후, 인코딩된 비디오 정보는 출력 인터페이스에 의해 컴퓨터 판독가능 매체상으로 출력될 수도 있다.

[0314] 상술된 바와 같이, 컴퓨터 판독가능 매체는 무선 브로드캐스트 또는 유선 네트워크 송신과 같은 일시적인 매체들, 또는 하드 디스크, 플래시 드라이브, 컴팩트 디스크, 디지털 비디오 디스크, 블루-레이 디스크, 또는 다른 컴퓨터 판독가능 매체들과 같은 저장 매체들 (즉, 비일시적인 저장 매체들) 을 포함할 수도 있다. 일부 예들에 있어서, 네트워크 서버 (도시 안됨) 는 인코딩된 비디오 데이터를 소스 디바이스로부터 수신하고, 인코딩된 비디오 데이터를, 예를 들어, 네트워크 송신을 통해 목적지 디바이스에 제공할 수도 있다. 유사하게, 디스크 스탬핑 설비와 같은 매체 생성 설비의 컴퓨팅 디바이스는 인코딩된 비디오 데이터를 소스 디바이스로부터 수신하고, 인코딩된 비디오 데이터를 포함하는 디스크를 생성할 수도 있다. 따라서, 컴퓨터 판독가능 매체는, 다양한 예들에 있어서, 다양한 형태들의 하나 이상의 컴퓨터 판독가능 매체들을 포함하도록 이해될 수도 있다.

[0315] 목적지 디바이스의 입력 인터페이스는 컴퓨터 판독 가능 매체로부터 정보를 수신한다. 컴퓨터 판독가능 매체의 정보는 비디오 인코더에 의해 정의되고 또한 비디오 디코더에 의해 이용되는 신택스 정보를 포함할 수도 있으며, 이 신택스 정보는 블록들 및 다른 코딩된 유닛들, 예를 들어, 화상들의 그룹 (GOP) 의 특성 및/또는 프로세싱을 기술하는 신택스 엘리먼트들을 포함한다. 디스플레이 디바이스는 디코딩된 비디오 데이터를 사용자에게 디스플레이하며, 음극선관 (CRT), 액정 디스플레이 (LCD), 플라즈마 디스플레이, 유기 발광 다이오드 (OLED) 디스플레이, 또는 다른 타입의 디스플레이 디바이스와 같은 다양한 디스플레이 디바이스들 중 임의의 디바이스를 포함할 수도 있다. 다양한 예들이 설명되었다.

[0316] 인코딩 디바이스 (104) 및 디코딩 디바이스 (112) 의 특정의 상세들은 각각 도 9 및 도 10 에 도시된다. FIG. 도 9 는 본 개시에서 설명된 기법들의 하나 이상을 구현할 수도 있는 예시적인 인코딩 디바이스 (104) 를 예시한 블록도이다. 인코딩 디바이스 (104) 는 예를 들어, 여기에 기술된 신택스 구조들 (예를 들어, VPS, SPS, PPS 또는 다른 신택스 엘리먼트들의 신택스 구조들) 을 생성 할 수있다. 인코딩 디바이스 (104) 는 비디오 슬라이스들 내에서 비디오 블록들의 인트라-예측 및 인터-예측을 수행할 수도 있다. 이전에 기술된 바와 같이, 인트라-코딩은 소정의 비디오 프레임 또는 화상 내의 공간 리던던시를 감소 또는 제거하기 위해 공간 예측에 적어도 부분적으로 의존한다. 인터-코딩은 비디오 시퀀스의 인접한 또는 주위의 프레임들 내의 시간 리던던시를 감소 또는 제거하기 위해 시간 예측에 적어도 부분적으로 의존한다. 인트라-모드 (I 모드) 는 수개의 공간 기반 압축 모드들 중 임의의 모드를 지칭할 수도 있다. 단방향 예측 (P 모드) 또는 양방향 예측 (B 모드) 과 같은 인터-모드들은 수개의 시간 기반 압축 모드들 중 임의의 모드를 지칭할 수도 있다.

[0317] 인코딩 디바이스 (104) 는 파티셔닝 유닛 (35), 예측 프로세싱 유닛 (41), 필터 유닛 (63), 화상 메모리 (64), 합산기 (50), 변환 프로세싱 유닛 (52), 양자화 유닛 (54), 및 엔트로피 인코딩 유닛 (56) 을 포함한다. 예측 프로세싱 유닛 (41) 은 모션 추정 유닛 (42), 모션 보상 유닛 (44) 및 인트라 예측 프로세싱 유닛 (46) 을 포함한다. 비디오 블록 재구성을 위해, 인코딩 디바이스 (104) 는 또한 역양자화 유닛 (58), 역변환 프로세싱 유닛 (60), 및 합산기 (62) 를 포함한다. 필터 유닛 (63) 은 디블록킹 (deblocking) 필터, 적응 루프 필터 (ALF), 및 샘플 적응 오프셋 (SAO) 필터와 같은 하나 이상의 루프 필터들을 나타내도록 의도된다. 필터 유닛 (63) 이 도 10 에 인 루프 필터인 것으로 도시되어 있지만, 필터 유닛 (63) 은 인 루프 (in loop) 필터인 것으로서 도 9 에 도시되지만, 다른 구성들에 있어서, 필터 유닛 (63) 은 포스트 루프 필터로서 구현될 수도 있다. 포스트 프로세싱 디바이스 (57) 는 인코딩 디바이스 (104) 에 의해 생성되는 인코딩된 비디오 데이터에 대해 추가적인 프로세싱을 수행할 수도 있다. 이 개시의 기술들은 일부 인스턴스들에서 인코딩 디바이스 (104) 에 의해 구현될 수도 있다. 그러나, 다른 인스턴스들에서, 본 개시의 기술들 중 하나 이상은 포스트 프로세싱 디바이스 (57) 에 의해 구현될 수도 있다.

[0318] 도 9 에 도시된 바와 같이, 인코딩 디바이스 (104) 는 비디오 데이터를 수신하고, 파티셔닝 유닛 (35) 은 그 데

이터를 비디오 블록들로 파티셔닝한다. 파티셔닝은 또한, 예를 들어, LCU들 및 CU들의 쿼드트리 구조에 따른 비디오 블록 파티셔닝 뿐아니라 슬라이스들, 슬라이스 세그먼트들, 타일들, 또는 다른 더 큰 유닛들로의 파티셔닝을 포함할 수도 있다. 인코딩 디바이스 (104) 는 일반적으로, 인코딩될 비디오 슬라이스 내의 비디오 블록들을 인코딩하는 컴포넌트들을 예시한다. 슬라이스는 다중의 비디오 블록들로 (및 가능하게는 타일들로서 지칭되는 비디오 블록들의 세트들로) 분할될 수도 있다. 예측 프로세싱 유닛 (41) 은 에러 결과들 (예를 들어, 코딩 레이트 및 왜곡의 레벨 등) 에 기초하여 현재의 비디오 블록에 대해 복수의 인트라 예측 코딩 모드들 중 하나 또는 복수의 인터 예측 코딩 모드들 중 하나와 같은, 복수의 가능한 코딩 모드들 중 하나를 선택할 수도 있다. 예측 프로세싱 유닛 (41) 은 결과적인 인트라- 또는 인터-코딩된 블록을 합산기 (50) 에 제공하여 잔차 블록 데이터를 생성하고, 합산기 (62) 에 제공하여 참조 화상으로서의 사용을 위한 인코딩된 블록을 복원할 수도 있다.

[0319] 예측 프로세싱 유닛 (41) 내의 인트라 예측 프로세싱 유닛 (46) 은 공간 압축을 제공하기 위해, 코딩될 현재 블록과 동일한 프레임 또는 슬라이스에서의 하나 이상의 이웃 블록들에 관해 현재 비디오 블록의 인트라-예측 코딩을 수행할 수도 있다. 예측 프로세싱 유닛 (41) 내의 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 시간 압축을 제공하기 위해 하나 이상의 레퍼런스 픽처들에서의 하나 이상의 예측 블록들에 관해 현재 비디오 블록의 인터-예측 코딩을 수행한다.

[0320] 모션 추정 유닛 (42) 은 비디오 시퀀스에 대한 미리결정된 패턴에 따라 비디오 슬라이스에 대한 인터-예측 모드를 결정하도록 구성될 수도 있다. 미리결정된 패턴은 시퀀스에서의 비디오 슬라이스들을, P 슬라이스들, B 슬라이스들 또는 GPB 슬라이스들로서 지정할 수도 있다. 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 은 고도로 통합될 수도 있지만, 개념적인 목적을 위해 별도로 도시된다. 모션 추정 유닛 (42) 에 의해 수행된 모션 추정은, 비디오 블록들에 대한 모션을 추정하는 모션 벡터들을 생성하는 프로세스이다. 모션 벡터는, 예를 들어, 참조 화상 내의 예측 블록에 관한 현재 비디오 프레임 또는 화상 내의 비디오 블록의 예측 유닛 (PU) 의 변위를 표시할 수도 있다.

[0321] 예측 블록은 픽셀 차이의 관점에서 코딩될 비디오 블록의 PU 와 밀접하게 매칭하도록 발견되는 블록이며, 이 픽셀 차이는 절대 차이의 합 (SAD), 제곱 차이의 합 (SSD), 또는 다른 상이한 메트릭들에 의해 결정될 수도 있다. 일부 예들에 있어서, 인코딩 디바이스 (104) 는 화상 메모리 (64) 에 저장된 참조 화상들의 서브-정수 픽셀 포지션들에 대한 값들을 계산할 수도 있다. 예를 들어, 인코딩 디바이스 (104) 는 참조 화상의 1/4 픽셀 포지션들, 1/8 픽셀 포지션들, 또는 다른 분수 픽셀 포지션들의 값들을 보간할 수도 있다. 따라서, 모션 추정 유닛 (42) 은 풀 픽셀 포지션들 및 분수 픽셀 포지션들에 관한 모션 탐색을 수행하고, 분수 픽셀 정밀도로 모션 벡터를 출력할 수도 있다.

[0322] 모션 추정 유닛 (42) 은 인터-코딩된 슬라이스에서의 비디오 블록의 PU 에 대한 모션 벡터를, 그 PU 의 포지션을 레퍼런스 픽처의 예측 블록의 포지션과 비교함으로써 계산한다. 레퍼런스 픽처는 제 1 레퍼런스 픽처 리스트 (List 0) 또는 제 2 레퍼런스 픽처 리스트 (List 1) 로부터 선택될 수도 있고, 이들의 각각은 픽처 메모리 (64) 내에 저장된 하나 이상의 레퍼런스 픽처들을 식별한다. 모션 추정 유닛 (42) 은 계산된 모션 벡터를 엔트로피 인코딩 유닛 (56) 및 모션 보상 유닛 (44) 으로 전송한다.

[0323] 모션 보상 유닛 (44) 에 의해 수행된 모션 보상은 모션 추정에 의해 결정된 모션 벡터에 기초하여 예측 블록을 페치 또는 생성하여, 가능하게는 서브픽셀 정밀도로 보간들을 수행하는 것을 수반할 수도 있다. 현재 비디오 블록의 PU 에 대한 모션 벡터를 수신할 시, 모션 보상 유닛 (44) 은, 모션 벡터가 참조 화상 리스트에서 포인팅하는 예측 블록을 로케이팅할 수도 있다. 인코딩 디바이스 (104) 는, 코딩되고 있는 현재 비디오 블록의 픽셀 값들로부터 예측 블록의 픽셀 값들을 감산하여 픽셀 차이 값들을 형성함으로써 잔차 비디오 블록을 형성한다. 픽셀 차이 값들은 블록에 대한 잔차 데이터를 형성하고, 루마 및 크로마 차이 컴포넌트들 양자를 포함할 수도 있다. 합산기 (50) 는 이러한 감산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 모션 보상 유닛 (44) 은 또한, 비디오 슬라이스의 비디오 블록들을 디코딩함에 있어서 디코딩 디바이스 (112) 에 의한 사용을 위해 비디오 블록들 및 비디오 슬라이스와 연관된 신택스 엘리먼트들을 생성할 수도 있다.

[0324] 인트라-예측 프로세싱 유닛 (46) 은 현재 블록을, 상기 설명된 바와 같은 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 수행된 인터-예측에 대한 대안으로서 인트라-예측할 수도 있다. 특히, 인트라-예측 프로세싱 유닛 (46) 은 현재 블록을 인코딩하는데 이용하기 위한 인트라-예측 모드를 결정할 수도 있다. 일부 예들에 있어서, 인트라 예측 프로세싱 유닛 (46) 은 예를 들어 별도의 인코딩 패스들 동안에 다양한 인트라-예측 모드들을 이용하여 현재 블록을 인코딩할 수도 있으며, 인트라 예측 프로세싱 유닛 (46) 은 테스팅된 모드들로부터의 이용을 위해 적절한 인트라-예측 모드를 선택할 수도 있다. 예를 들어, 인트라-예측 프로세싱 유닛 (46) 은 다

양한 테스팅된 인트라-예측 모드들에 대한 레이트-왜곡 분석을 이용하여 레이트-왜곡 값들을 계산하고, 테스팅된 모드들 중 최상의 레이트-왜곡 특성들을 갖는 인트라-예측 모드를 선택할 수도 있다. 레이트-왜곡 분석은 일반적으로, 인코딩된 블록과 그 인코딩된 블록을 생성하도록 인코딩되었던 오리지널의 인코딩되지 않은 블록 간의 왜곡 (또는 에러) 의 양뿐 아니라 인코딩된 블록을 생성하는데 사용된 비트 레이트 (즉, 비트들의 수) 를 결정한다. 인트라-예측 프로세싱 유닛 (46) 은 다양한 인코딩된 블록들에 대한 왜곡들 및 레이트들로부터의 비율들을 계산하여, 어느 인트라-예측 모드가 그 블록에 대한 최상의 레이트-왜곡 값을 나타내는지를 결정할 수도 있다.

[0325] 어떤 경우든, 블록에 대한 인트라-예측 모드를 선택한 이후, 인트라-예측 프로세싱 유닛 (46) 은 블록에 대한 선택된 인트라-예측 모드를 표시하는 정보를 엔트로피 인코딩 유닛 (56) 에 제공할 수도 있다. 엔트로피 인코딩 유닛 (56) 은 선택된 인트라-예측 모드를 표시한 정보를 인코딩할 수도 있다. 인코딩 디바이스 (104) 는 다양한 블록들에 대한 인코딩 컨텍스트들의 정의들 뿐아니라 그 컨텍스트들 각각에 대해 사용할 가장 가능성있는 인트라-예측 모드, 인트라-예측 모드 인덱스 테이블 및 변형된 인트라-예측 모드 인덱스 테이블의 표시들을 전송된 비트 스트림 구성 데이터에 포함할 수있다. 비트 스트림 구성 데이터는 복수의 인트라 예측 모드 인덱스 테이블 및 복수의 변형된 인트라 예측 모드 인덱스 테이블 (코드워드 맵핑 테이블이라고도 함) 을 포함 할 수있다.

[0326] 예측 프로세싱 유닛 (41) 이 인터-예측 또는 인트라-예측 중 어느 하나를 통해 현재 비디오 블록에 대한 예측 블록을 생성한 후, 인코딩 디바이스 (104) 는 현재 비디오 블록으로부터 예측 블록을 감산함으로써 잔차 비디오 블록을 형성한다. 잔차 블록에서의 잔차 비디오 데이터는 하나 이상의 TU들에 포함되고 변환 프로세싱 유닛 (52) 에 적용될 수도 있다. 변환 프로세싱 유닛 (52) 은 이산 코사인 변환 (DCT) 또는 개념적으로 유사한 변환과 같은 변환을 이용하여 잔차 비디오 데이터를 잔차 변환 계수들로 변환한다. 변환 프로세싱 유닛 (52) 은 잔차 비디오 데이터를 픽셀 도메인으로부터 주파수 도메인과 같은 변환 도메인으로 컨버팅할 수도 있다.

[0327] 변환 프로세싱 유닛 (52) 은 결과적인 변환 계수들을 양자화 유닛 (54) 으로 전송할 수도 있다. 양자화 유닛 (54) 은 비트 레이트를 추가로 감소시키기 위해 변환 계수들을 양자화한다. 양자화 프로세스는 그 계수들의 일부 또는 모두와 연관되는 비트 심도를 감소시킬 수도 있다. 양자화의 정도는 양자화 파라미터를 조정함으로써 수정될 수도 있다. 그 후, 일부 예들에 있어서, 양자화 유닛 (54) 은, 양자화된 변환 계수들을 포함하는 매트릭스의 스캔을 수행할 수도 있다. 대안적으로, 엔트로피 인코딩 유닛 (56) 은 스캔을 수행할 수도 있다.

[0328] 양자화 이후, 엔트로피 인코딩 유닛 (56) 은 양자화된 변환 계수들을 엔트로피 인코딩한다. 예를 들어, 엔트로피 인코딩 유닛 (56) 은 컨텍스트 적응적 가변 길이 코딩 (CAVLC), 컨텍스트 적응적 바이너리 산술 코딩 (CABAC), 신택스 기반 컨텍스트 적응적 바이너리 산술 코딩 (SBAC), 확률 간격 파티셔닝 엔트로피 (PIPE) 코딩, 또는 다른 엔트로피 인코딩 기술을 수행할 수도 있다. 엔트로피 인코딩 유닛 (56) 에 의한 엔트로피 인코딩 이후, 인코딩된 비트스트림은 디코딩 디바이스 (112) 로 송신되거나, 또는 디코딩 디바이스 (112) 에 의한 나중의 송신 또는 취출을 위해 아카이브될 수도 있다. 엔트로피 인코딩 유닛 (56) 은 또한, 코딩되고 있는 현재 비디오 슬라이스에 대한 모션 벡터들 및 다른 신택스 엘리먼트들을 엔트로피 인코딩할 수도 있다.

[0329] 역양자화 유닛 (58) 및 역변환 프로세싱 유닛 (60) 은, 각각, 역양자화 및 역변환을 적용하여, 레퍼런스 픽처의 레퍼런스 블록으로서의 더 나중 사용을 위해 픽셀 도메인에서 잔차 블록을 재구성한다. 모션 보상 유닛 (44) 은, 레퍼런스 픽처 리스트 내 하나의 레퍼런스 픽처의 예측 블록에 잔차 블록을 가산함으로써 레퍼런스 블록을 계산할 수도 있다. 모션 보상 유닛 (44) 은 또한, 하나 이상의 보간 필터들을 복원된 잔차 블록에 적용하여, 모션 추정에서의 사용을 위한 서브-정수 픽셀 값들을 계산할 수도 있다. 합산기 (62) 는 복원된 잔차 블록을, 모션 보상 유닛 (44) 에 의해 생성된 모션 보상된 예측 블록에 부가하여, 참조 화상 메모리 (64) 로의 저장을 위한 참조 블록을 생성한다. 참조 블록은, 후속 비디오 프레임 또는 화상에서의 블록을 인터-예측하기 위해 참조 블록으로서 모션 추정 유닛 (42) 및 모션 보상 유닛 (44) 에 의해 사용될 수도 있다.

[0330] 이러한 방식으로, 도 9 의 인코딩 디바이스 (104) 는 인코딩된 비디오 비트 스트림에 대한 신택스를 생성하도록 구성된 비디오 인코더의 예를 나타낸다. 인코딩 디바이스 (104) 는 예를 들어, 전술한 바와 같이 VPS, SPS 및 PPS 파라미터 세트들을 생성할 수있다. 인코딩 디바이스(104)는 도 7 및 도 8 을 참조하여 위에서 설명된 프로세스들을 포함하는 본 명세서에서 설명되는 기법들 중 임의의 것을 수행할 수도 있다. 본 개시의 기술들은 일반적으로 인코딩 디바이스 (104) 와 관련하여 설명되었지만, 전술한 바와 같이, 본 개시의 기술 중 일부는 또한 포스트 프로세싱 디바이스 (57) 에 의해 구현될 수도 있다.

[0331] 도 10 은 예시적인 인코딩 디바이스 (112) 를 도시한 블록도이다. 디코딩 디바이스 (112) 는 엔트로피 디코딩 유닛 (80), 예측 프로세싱 유닛 (81), 역양자화 유닛 (86), 역변환 프로세싱 유닛 (88), 합산기 (90), 필터 유

닛 (91), 및 화상 메모리 (92) 를 포함한다. 예측 프로세싱 유닛 (81) 은 모션 보상 유닛 (82) 및 인트라 예측 프로세싱 유닛 (84) 을 포함한다. 일부 예들에 있어서, 디코딩 디바이스 (112) 는 도 9 으로부터의 인코딩 디바이스 (104) 에 대하여 설명된 인코딩 패스에 일반적으로 상호적인 디코딩 패스를 수행할 수도 있다.

[0332] 디코딩 프로세스 동안, 디코딩 디바이스 (112) 는 인코딩 디바이스 (104) 에 의해 전송된 인코딩된 비디오 슬라이스의 비디오 블록들 및 관련 신택스 엘리먼트들을 나타내는 인코딩된 비디오 비트스트림을 수신한다. 일부 예들에서, 디코딩 디바이스 (112) 는 인코딩 디바이스 (104) 로부터 인코딩된 비디오 비트 스트림을 수신할 수있다. 일부 예들서, 디코딩 디바이스 (112) 는 서버, 미디어 인식 네트워크 엘리먼트 (MANE), 비디오 편집자/스플라이서 (splicer), 또는 상술된 기술들 중 하나 이상을 구현하도록 구성된 다른 그러한 디바이스와 같은 네트워크 엔티티 (79) 로부터 인코딩된 비디오 비트스트림을 수신할 수도 있다. 네트워크 엔티티 (79) 는 인코딩 디바이스 (104) 를 포함할 수도 있거나 포함하지 않을 수도 있다. 본 개시에 설명된 기법들 중 일부는 네트워크 엔티티 (79) 가 인코딩된 비디오 비트스트림을 디코딩 디바이스 (112) 로 송신하는 것 이전에 네트워크 엔티티 (79) 에 의해 구현될 수도 있다. 일부 비디오 디코딩 시스템들에 있어서, 네트워크 엔티티 (79) 및 디코딩 디바이스 (112) 는 별개의 디바이스들의 부분들일 수도 있지만, 다른 예들에 있어서, 네트워크 엔티티 (79) 에 대하여 설명된 기능은 디코딩 디바이스 (112) 를 포함하는 동일한 디바이스에 의해 수행될 수도 있다.

[0333] 디코딩 디바이스 (112) 의 엔트로피 디코딩 유닛 (80) 은 비트스트림을 엔트로피 디코딩하여, 양자화된 계수들, 모션 벡터들, 및 다른 신택스 엘리먼트들을 생성한다. 엔트로피 디코딩 유닛 (80) 은 모션 벡터들 및 다른 신택스 엘리먼트들을 예측 프로세싱 유닛 (81) 으로 포워딩한다. 디코딩 디바이스 (112) 는 신택스 엘리먼트들을 비디오 슬라이스 레벨 및/또는 비디오 블록 레벨로 수신할 수도 있다. 엔트로피 디코딩 유닛 (80)은 VPS, SPS 및 PPS와 같은 하나 이상의 파라미터 세트에서 고정 길이 신택스 엘리먼트들 및 가변 길이 신택스 엘리먼트들을 처리 및 파싱할 수있다.

[0334] 비디오 슬라이스가 인트라-코딩된 (I) 슬라이스로서 코딩된 경우, 예측 프로세싱 유닛 (81) 의 인트라 예측 프로세싱 유닛 (84) 은 현재 프레임 또는 화상의 이전에 디코딩된 블록들로부터의 데이터 및 시그널링된 인트라 예측 모드에 기초하여 현재 비디오 슬라이스의 비디오 블록에 대한 예측 데이터를 생성할 수도 있다. 비디오 프레임이 인터-코딩된 (즉, B, P 또는 GPB) 슬라이스로서 코딩된 경우, 예측 프로세싱 유닛 (81) 의 모션 보상 유닛 (82) 은 엔트로피 디코딩 유닛 (80) 으로부터 수신된 모션 벡터들 및 다른 신택스 엘리먼트들에 기초하여 현재 비디오 슬라이스의 비디오 블록에 대한 예측 블록들을 생성한다. 예측 블록들은 참조 화상 리스트 내의 참조 화상들 중 하나로부터 생성될 수도 있다. 디코딩 디바이스 (112) 는 화상 메모리 (92) 에 저장된 참조 화상들에 기초한 디폴트 구성 기법들을 이용하여 레퍼런스 프레임 리스트들, 즉, 리스트 0 및 리스트 1 을 구성할 수도 있다.

[0335] 모션 보상 유닛 (82) 은 모션 벡터들 및 다른 신택스 엘리먼트들을 파싱함으로써 현재 비디오 슬라이스의 비디오 블록에 대한 예측 정보를 결정하고, 그 예측 정보를 사용하여, 디코딩되고 있는 현재 비디오 블록에 대한 예측 블록들을 생성한다. 예를 들어, 모션 보상 유닛 (82) 은 파라미터 세트 내의 하나 이상의 신택스 엘리먼트들을 사용하여, 비디오 슬라이스의 비디오 블록들을 코딩하는데 사용되는 예측 모드 (예를 들어, 인트라-예측 또는 인터-예측), 인터-예측 슬라이스 타입 (예를 들어, B 슬라이스, P 슬라이스, 또는 GPB 슬라이스), 슬라이스에 대한 하나 이상의 참조 화상 리스트들에 대한 구성 정보, 슬라이스의 각각의 인터-인코딩된 비디오 블록에 대한 모션 벡터들, 슬라이스의 각각의 인터-코딩된 비디오 블록에 대한 인터-예측 상태, 및 현재 비디오 슬라이스에서의 비디오 블록들을 디코딩하기 위한 다른 정보를 결정할 수도 있다.

[0336] 모션 보상 유닛 (82) 은 또한, 보간 필터들에 기초하여 보간을 수행할 수도 있다. 모션 보상 유닛 (82) 은 비디오 블록들의 인코딩 동안 인코딩 디바이스 (104) 에 의해 사용된 바와 같은 보간 필터들을 이용하여, 참조 블록들의 서브-정수 픽셀들에 대한 보간된 값들을 계산할 수도 있다. 이 경우, 모션 보상 유닛 (82) 은 수신된 신택스 엘리먼트들로부터 인코딩 디바이스 (104) 에 의해 사용된 보간 필터들을 결정할 수도 있고, 보간 필터들을 이용하여 예측 블록들을 생성할 수도 있다.

[0337] 역양자화 유닛 (86) 은 비트스트림에서 제공되고 엔트로피 디코딩 유닛 (80) 에 의해 디코딩되는 양자화된 변환 계수들을 역양자화, 또는 탈양자화한다. 역양자화 프로세스는 적용되어야 하는 양자화의 정도 및 유사하게 역양자화의 정도를 결정하기 위해 비디오 슬라이스에서의 각각의 비디오 블록에 대해 인코딩 디바이스 (104) 에 의해 계산된 양자화 파라미터의 사용을 포함할 수도 있다. 역변환 프로세싱 유닛 (88) 은 픽셀 도메인에서 잔차 블록들을 생성하기 위해, 역변환, (예를 들어, 역 DCT, 또는 다른 적합한 역변환), 역 정수 변환, 또는 개념적으로 유사한 역변환 프로세스를 변환 계수들에 적용한다.

[0338] 모션 보상 유닛 (82) 이 모션 벡터들 및 다른 신택스 엘리먼트들에 기초하여 현재 비디오 블록에 대한 예측 블록을 생성한 후, 디코딩 디바이스 (112) 는 역변환 프로세싱 유닛 (88) 으로부터의 잔차 블록들을 모션 보상 유닛 (82) 에 의해 생성된 대응하는 예측 블록들과 합산함으로써 디코딩된 비디오 블록을 형성한다. 합산기 (90) 는 이러한 합산 연산을 수행하는 컴포넌트 또는 컴포넌트들을 나타낸다. 요구된다면, (코딩 루프에 있어서 또는 코딩 루프 이후에) 루프 필터들이 또한 픽셀 천이들을 평활하게 하거나 그렇지 않으면 비디오 품질을 개선하기 위해 이용될 수도 있다. 필터 유닛 (91) 은 디블록킹 필터, ALF (adaptive loop filter) 및 SAO (sample adaptive offset) 필터 등의 하나 이상의 루프 필터들을 나타내도록 의도된다. 필터 유닛 (91) 이 도 10 에 인 루프 필터인 것으로 도시되어 있지만, 다른 구성들에 있어서, 필터 유닛 (91) 은 포스트 루프 (post loop) 필터로서 구현될 수도 있다. 그 후, 소정의 프레임 또는 화상에서의 디코딩된 비디오 블록들이 후속적인 모션 보상을 위해 사용되는 참조 화상들을 저장하는 화상 메모리 (92) 에 저장된다. 화상 메모리 (92) 는 또한, 도 1 의 비디오 목적지 디바이스 (122) 와 같은 디스플레이 디바이스 상의 나중의 제시를 위해 디코딩된 비디오를 저장한다.

[0339] 전술한 설명에서, 본 출원의 양태는 특정 예들을 참조하여 기재되었지만, 당업자는 본 출원이 이에 제한되지 않는다는 것을 인식 할 것이다. 따라서, 본 출원의 예들이 본원에 상세히 설명되었지만, 본 발명의 개념은 달리 다양하게 구체화되고 채택 될 수 있으며, 첨부된 청구 범위는 선행 기술에 의해 제한되는 것을 제외하고는 그러한 변형을 포함하는 것으로 해석되도록 의도된다. 전술한 예들의 다양한 특징 및 양태는 개별적으로 또는 공동으로 사용될 수있다. 또한, 기술된 예들은 본 명세서의 더 넓은 사상 및 범위를 벗어나지 않으면서 본 명세서에 기재된 것 이외의 임의의 수의 환경 및 애플리케이션에서 이용 될 수 있다. 이에 따라, 명세서 및 도면들은 한정적 의미보다는 예시적 의미로 간주되어야 한다. 예시의 목적 상, 방법은 특정 순서로 기술되었다. 대안적인 예들에서, 상기 방법은 설명된 것과 다른 순서로 수행될 수 있다는 것을 이해해야한다.

[0340] 컴포넌트들이 특정 동작을 수행"하도록 구성된" 것으로 기술되는 경우, 그러한 구성은 예를 들어, 전자 회로 또는 다른 하드웨어를 동작을 수행하도록 설계함으로써, 프로그래밍 가능한 전자 회로 (예를 들어, 마이크로 프로세서 또는 다른 적절한 전자 회로) 를 동작을 수행하도록 프로그래밍함으로써, 또는 이들의 임의의 조합으로써 달성될 수 있다.

[0341] 본 명세서에 개시된 예들과 관련하여 설명된 다양한 예시적인 논리 블록들, 모듈들, 회로들, 및 알고리즘 단계들은 전자 하드웨어, 컴퓨터 소프트웨어, 펌웨어, 또는 이들의 조합들로서 구현될 수도 있다. 하드웨어와 소프트웨어의 이러한 상호대체 가능성을 분명히 예시하기 위해, 다양한 예시적인 컴포넌트들, 블록들, 모듈들, 회로들 및 단계들이 일반적으로 그들의 기능의 관점에서 상기 설명되었다. 그러한 기능이 하드웨어로서 구현될지 또는 소프트웨어로서 구현될지는 전체 시스템에 부과된 설계 제약들 및 특정 어플리케이션에 의존한다. 당업자는 설명된 기능을 각각의 특정 어플리케이션에 대하여 다양한 방식들로 구현할 수도 있지만, 그러한 구현의 결정들이 본 출원의 범위로부터의 일탈을 야기하는 것으로서 해석되지는 않아야 한다.

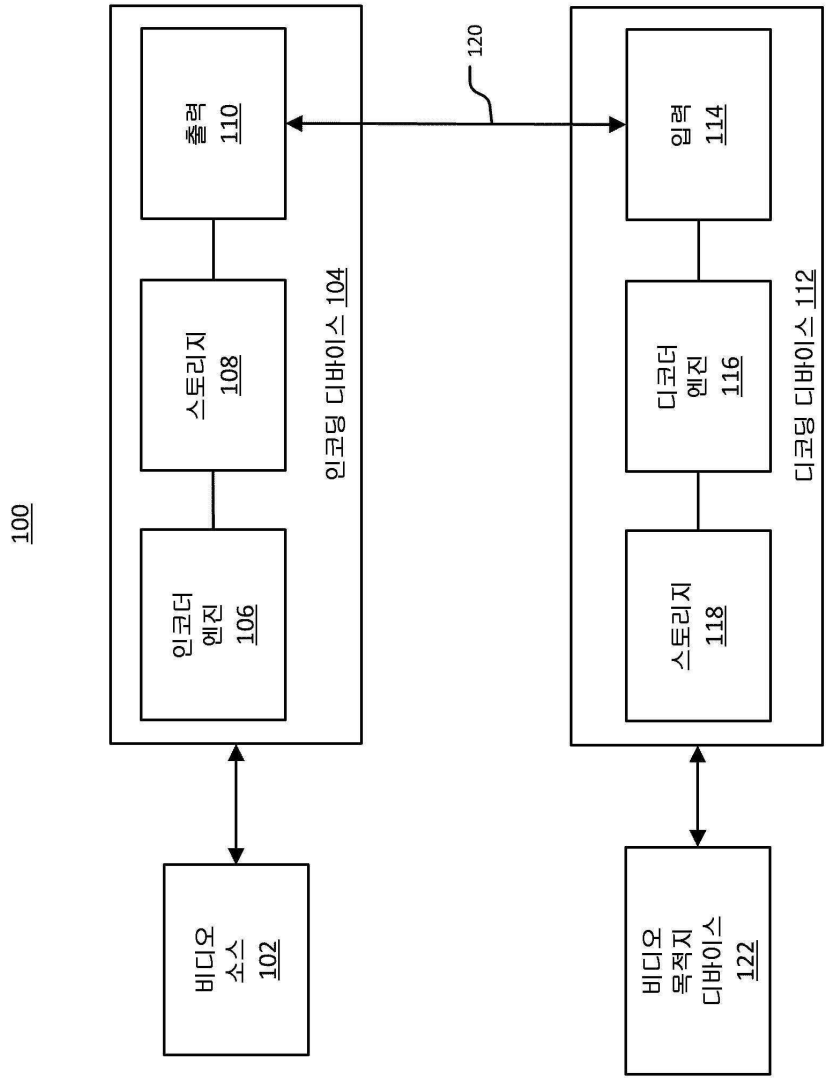
[0342] 본 명세서에서 설명된 기술들은 또한 전자 하드웨어, 컴퓨터 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 그러한 기술들은 범용 컴퓨터들, 무선 통신 디바이스 핸드셋들, 또는 무선 통신 디바이스 핸드셋들 및 다른 디바이스들에서의 애플리케이션을 포함하여 다중의 이용들을 갖는 집적 회로 디바이스들과 같은 임의의 다양한 디바이스들에서 구현될 수도 있다. 모듈들 또는 컴포넌트들로서 설명된 임의의 특징들은 집적된 로직 디바이스에서 함께 또는 별개지만 상호운용가능한 로직 디바이스들로서 별도로 구현될 수도 있다. 소프트웨어에서 구현되면, 그 기법들은, 실행될 경우 상기 설명된 방법들 중 하나 이상을 수행하는 명령들을 포함하는 프로그램 코드를 포함하는 컴퓨터 판독가능 데이터 저장 매체에 의해 적어도 부분적으로 실현될 수도 있다. 컴퓨터 판독가능 데이터 저장 매체는 컴퓨터 프로그램 제품의 부분을 형성할 수도 있으며, 이는 패키징 재료들을 포함할 수도 있다. 컴퓨터 판독가능 매체는 동기식 동적 랜덤 액세스 메모리 (SDRAM) 와 같은 랜덤 액세스 메모리 (RAM), 판독 전용 메모리 (ROM), 비휘발성 랜덤 액세스 메모리 (NVRAM), 전기적으로 소거가능한 프로그래밍 가능 판독 전용 메모리 (EEPROM), 플래시 메모리, 자기 또는 광학 데이터 저장 매체들 등과 같은 메모리 또는 데이터 저장 매체들을 포함할 수도 있다. 그 기법들은, 부가적으로 또는 대안적으로, 전파된 신호들 또는 파동들과 같이, 명령들 또는 데이터 구조들의 형태로 프로그램 코드를 수록하거나 통신하고 그리고 컴퓨터에 의해 액세스, 판독 및/또는 실행될 수 있는 컴퓨터 판독가능 통신 매체에 의해 적어도 부분적으로 실현될 수도 있다.

[0343] 프로그램 코드는, 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로 프로세서들, 주문형 집적 회로들 (ASIC들), 필드 프로그래밍가능 로직 어레이들 (FPGA들), 또는 다른 균등한 집적된 또는 별개의 로직 회로부와 같은 하나 이상의 프로세서들을 포함할 수도 있는 프로세서에 의해 실행될 수도 있다. 그러한 프로세서는 본 개시에서 설명된 기법들 중 임의의 기법을 수행하도록 구성될 수도 있다. 범용 프로세서는 마이크로프로세서일

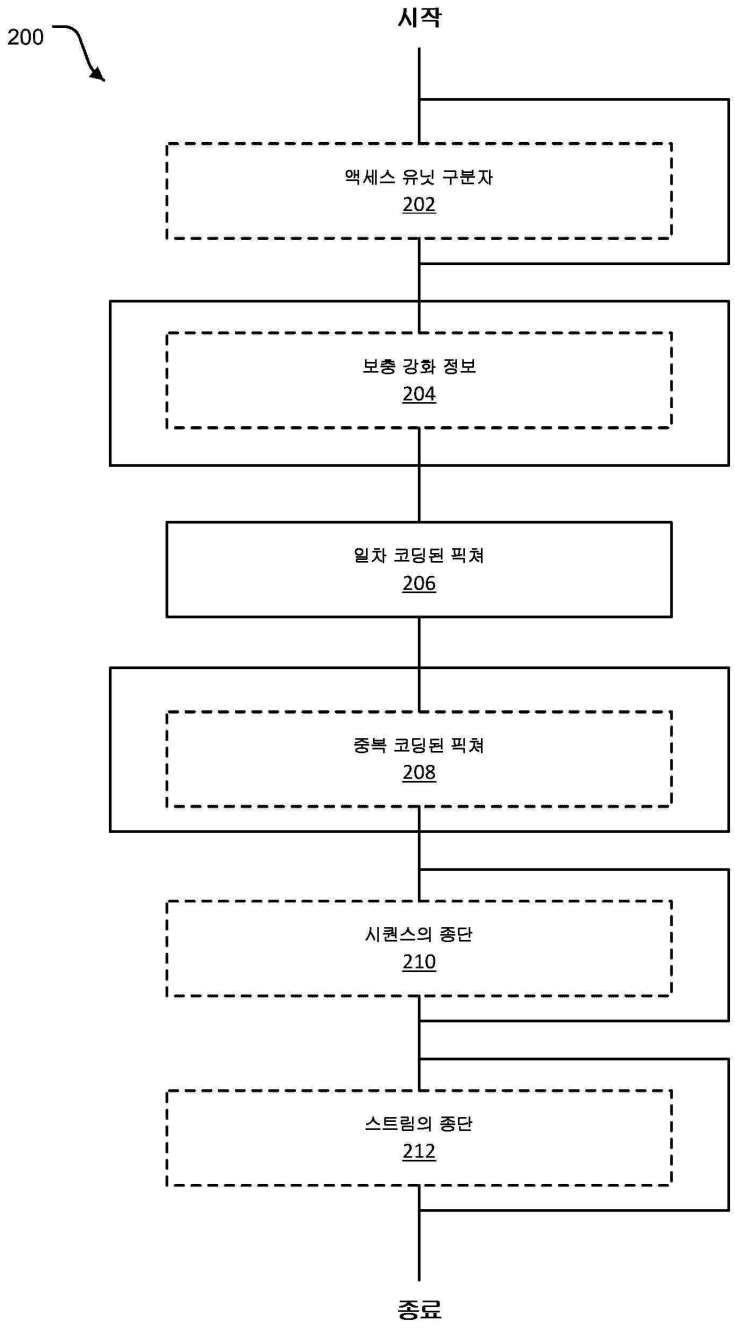
수도 있지만, 대안적으로, 그 프로세서는 임의의 종래의 프로세서, 제어기, 마이크로 제어기, 또는 상태 머신일 수도 있다. 프로세서는 또한, 컴퓨팅 디바이스들의 조합 ,예를 들어, DSP 와 마이크로프로세서의 조합, 복수의 마이크로프로세서들, DSP 코어와 결합된 하나 이상의 마이크로프로세서들, 또는 임의의 그러한 다른 구성물 으로서 구현될 수도 있다. 이에 따라, 본 명세서에서 사용된 바와 같은 용어 "프로세서" 는 전술한 구조, 전술한 구조의 임의의 조합, 또는 본 명세서에서 설명된 기법들의 구현에 적합한 임의의 다른 구조 또는 장치 중 임의의 것을 지칭할 수도 있다. 부가적으로, 일부 양태들에 있어서, 본 명세서에서 설명된 기능은, 인코딩 및 디코딩을 위해 구성된 전용 소프트웨어 모듈들 또는 하드웨어 모듈들 내에서 제공되거나, 또는 결합된 비디오 인코더-디코더 (CODEC) 에 통합될 수도 있다.

도면

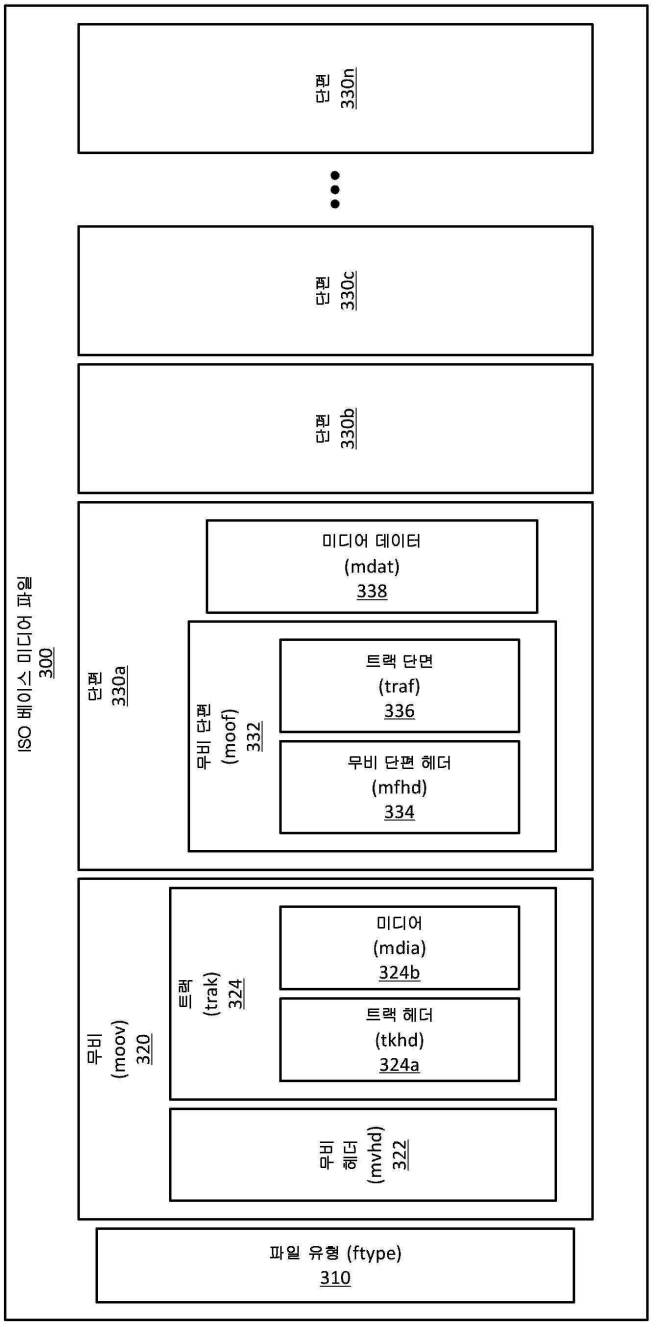
도면1



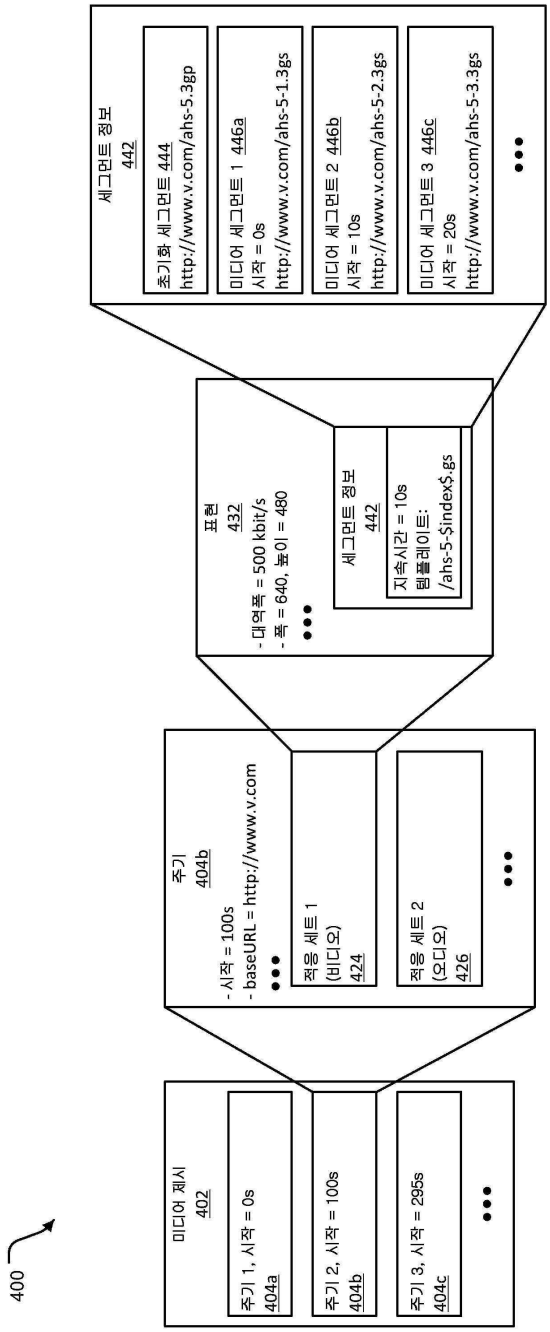
도면2


200
시작
액세스 유닛 구분자
202
보충 강화 정보
204
일차 코딩된 픽쳐
206
중복 코딩된 픽쳐
208
시퀀스의 종단
210
스트림의 종단
212
종료

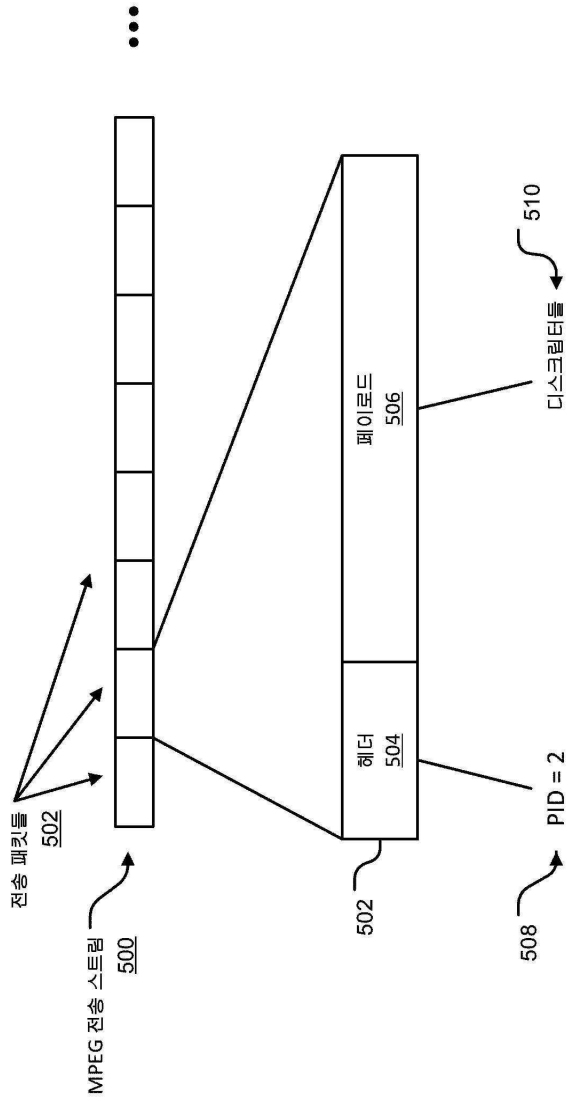
도면3


ISO 베이스 미디어 파일
300
파일 유형 (ftype)
310
무비
(moov)
320
무비 헤더
(mvhd)
322
트랙
(trak)
324
트랙 헤더
(tkhd)
324a
미디어
(mdia)
324b
단편
330a
무비 단편
(moof)
332
무비 단편 헤더
(mfhd)
334
트랙 단면
(traf)
336
미디어 데이터
(mdat)
338
단편
330b
단편
330c
단편
330n

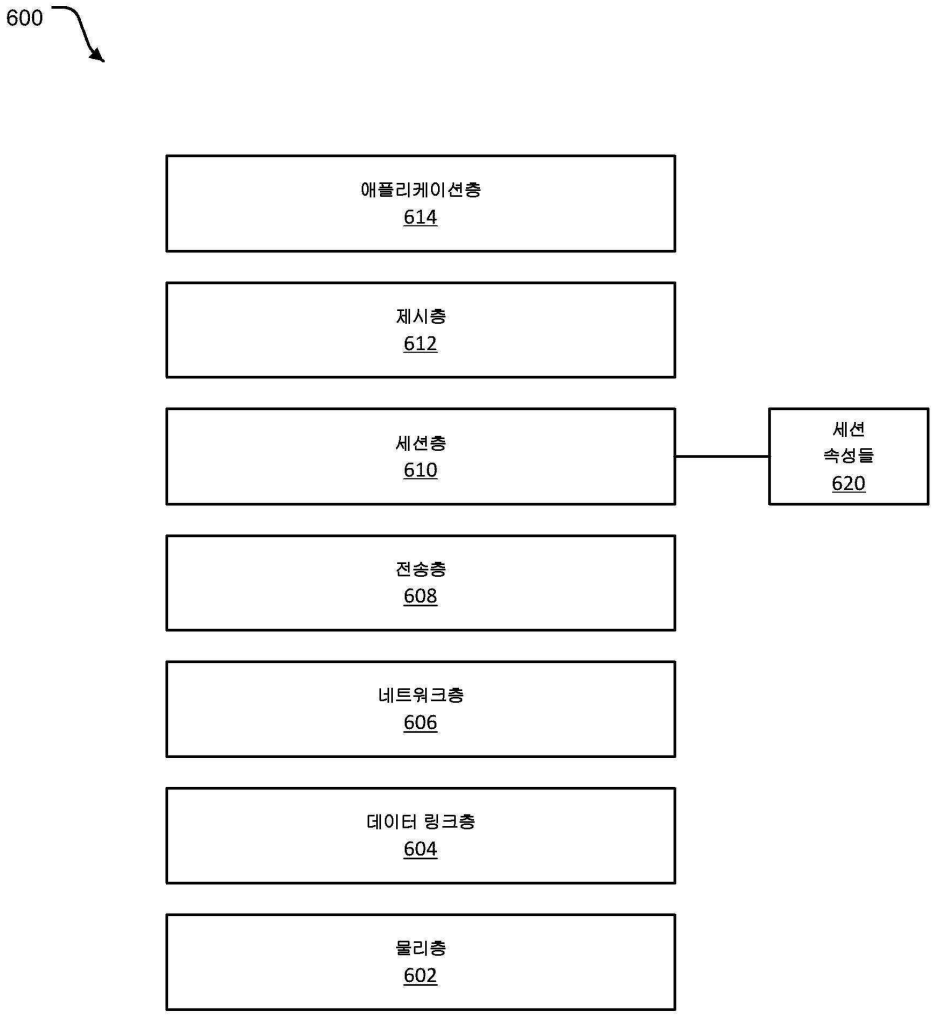
도면4


400
미디어 제시
402
주기 1, 시작 = 0s
404a
주기 2, 시작 = 100s
404b
주기 3, 시작 = 295s
404c
주기
404b
- 시작 = 100s
- baseURL = http://www.v.com
적응 세트 1
(비디오)
424
적응 세트 2
(오디오)
426
표현
432
- 대역폭 = 500 kbit/s
- 폭 = 640, 높이 = 480
세그먼트 정보
442
지속시간 = 10s
템플레이트:
/ahs-5-$index$.gs
세그먼트 정보
442
초기화 세그먼트 444
http://www.v.com/ahs-5.3gp
미디어 세그먼트 1 446a
시작 = 0s
http://www.v.com/ahs-5-1.3gs
미디어 세그먼트 2 446b
시작 = 10s
http://www.v.com/ahs-5-2.3gs
미디어 세그먼트 3 446c
시작 = 20s
http://www.v.com/ahs-5-3.3gs

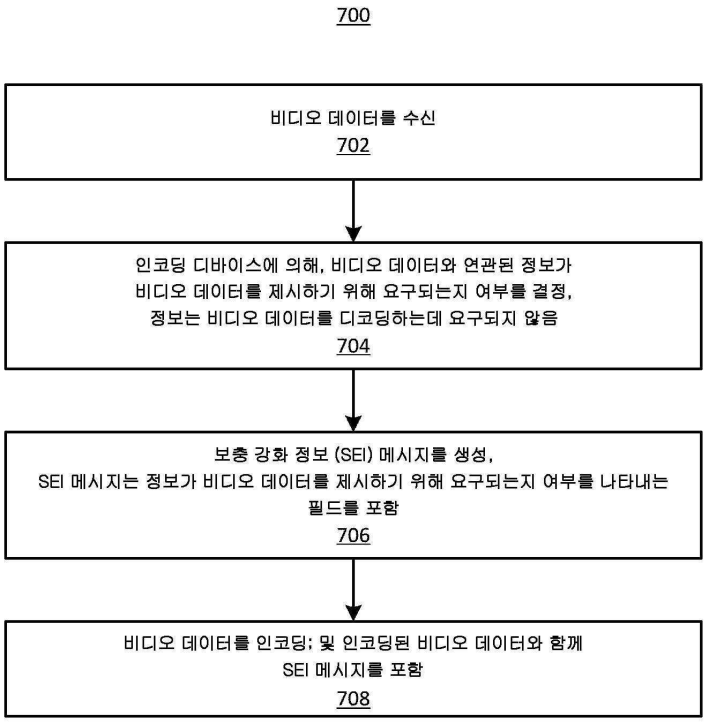
도면5


전송 패킷들
502
MPEG 전송 스트림
500
헤더
504
페이로드
506
502
PID = 2
508
디스크립터들
510

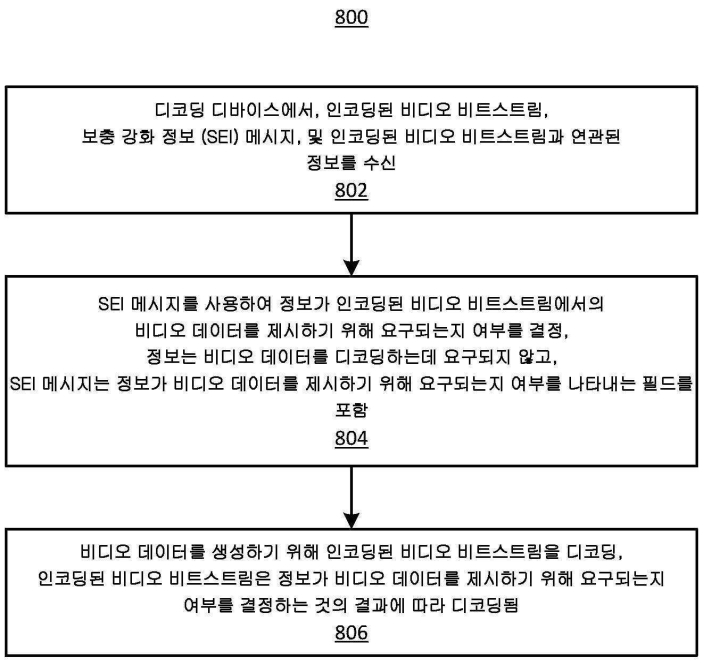
도면6


600
애플리케이션층
614
제시층
612
세션층
610
세션
속성들
620
전송층
608
네트워크층
606
데이터 링크층
604
물리층
602

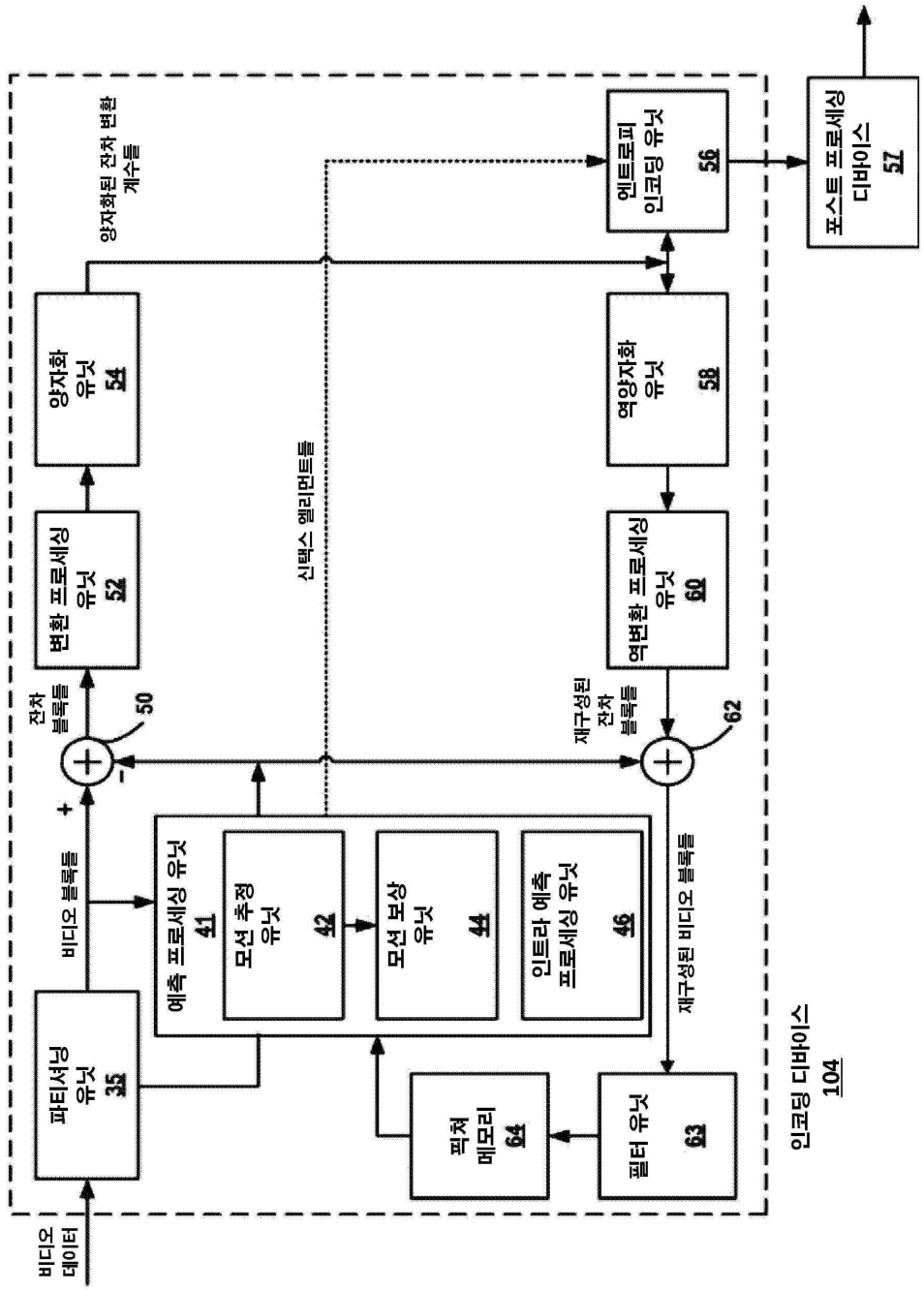
도면7


700
비디오 데이터를 수신
702
인코딩 디바이스에 의해, 비디오 데이터와 연관된 정보가
비디오 데이터를 제시하기 위해 요구되는지 여부를 결정,
정보는 비디오 데이터를 디코딩하는데 요구되지 않음
704
보충 강화 정보 (SEI) 메시지를 생성,
SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 나타내는
필드를 포함
706
비디오 데이터를 인코딩; 및 인코딩된 비디오 데이터와 함께
SEI 메시지를 포함
708

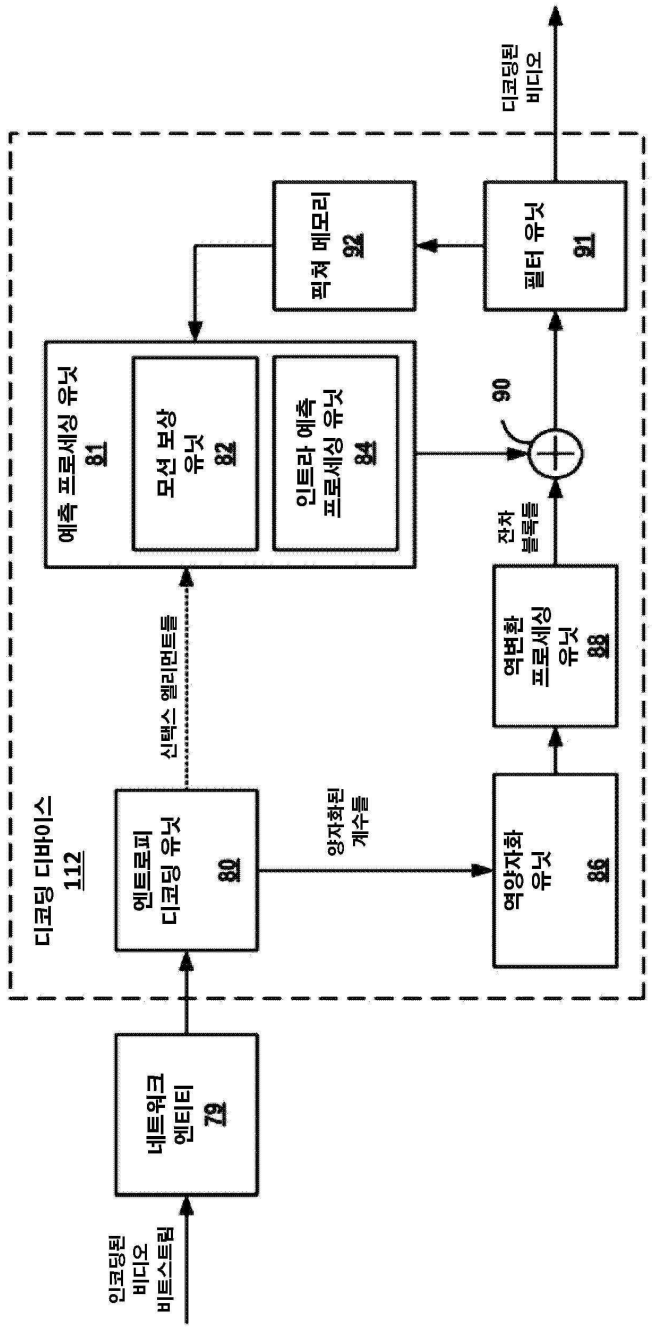
도면8


800
디코딩 디바이스에서, 인코딩된 비디오 비트스트림,
보충 강화 정보 (SEI) 메시지, 및 인코딩된 비디오 비트스트림과 연관된
정보를 수신
802
SEI 메시지를 사용하여 정보가 인코딩된 비디오 비트스트림에서의
비디오 데이터를 제시하기 위해 요구되는지 여부를 결정,
정보는 비디오 데이터를 디코딩하는데 요구되지 않고,
SEI 메시지는 정보가 비디오 데이터를 제시하기 위해 요구되는지 여부를 나타내는 필드를
포함
804
비디오 데이터를 생성하기 위해 인코딩된 비디오 비트스트림을 디코딩,
인코딩된 비디오 비트스트림은 정보가 비디오 데이터를 제시하기 위해 요구되는지
여부를 결정하는 것의 결과에 따라 디코딩됨
806

도면9


비디오 데이터
파티셔닝 유닛 35
픽처 메모리 64
필터 유닛 63
예측 프로세싱 유닛 41
모션 추정 유닛 42
모션 보상 유닛 44
인트라 예측 프로세싱 유닛 46
비디오 블록들
잔차 블록들
50
변환 프로세싱 유닛 52
양자화 유닛 54
양자화된 잔차 변환 계수들
신택스 엘리먼트들
엔트로피 인코딩 유닛 56
포스트 프로세싱 디바이스 57
역양자화 유닛 58
역변환 프로세싱 유닛 60
재구성된 잔차 블록들
62
재구성된 비디오 블록들
인코딩 디바이스 104

도면10


디코딩된 비디오
필터 유닛 91
픽처 메모리 92
예측 프로세싱 유닛 81
모션 보상 유닛 82
인트라 예측 프로세싱 유닛 84
90
잔차 블록들
역변환 프로세싱 유닛 88
신택스 엘리먼트들
양자화된 계수들
엔트로피 디코딩 유닛 80
역양자화 유닛 86
디코딩 디바이스 112
네트워크 엔티티 79
인코딩된 비디오 비트스트림