



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2015년10월07일

(11) 등록번호 10-1558627

(24) 등록일자 2015년10월01일

(51) 국제특허분류(Int. Cl.)

H04N 19/597 (2014.01) H04N 13/00 (2006.01)

H04N 19/70 (2014.01)

(21) 출원번호 10-2010-7009367

(22) 출원일자(국제) 2008년09월16일

심사청구일자 2013년06월24일

(85) 번역문제출일자 2010년04월28일

(65) 공개번호 10-2010-0085078

(43) 공개일자 2010년07월28일

(86) 국제출원번호 PCT/US2008/010775

(87) 국제공개번호 WO 2009/048502

국제공개일자 2009년04월16일

(30) 우선권주장

60/977,709 2007년10월05일 미국(US)

(56) 선행기술조사문헌

US20070211796 A1*

*는 심사관에 의하여 인용된 문헌

(73) 특허권자

툼슨 라이선싱

프랑스 92130 이씨레물리노 루 잔다르크 1-5

(72) 발명자

루오, 지양총

미국 뉴저지주 08536 플레인즈버로 헌터스 글렌
드라이브 3404

인, 평

미국 뉴저지주 08536 플레인즈버로 소로 드라이브
49

(74) 대리인

양영준, 전경석, 백만기

전체 청구항 수 : 총 2 항

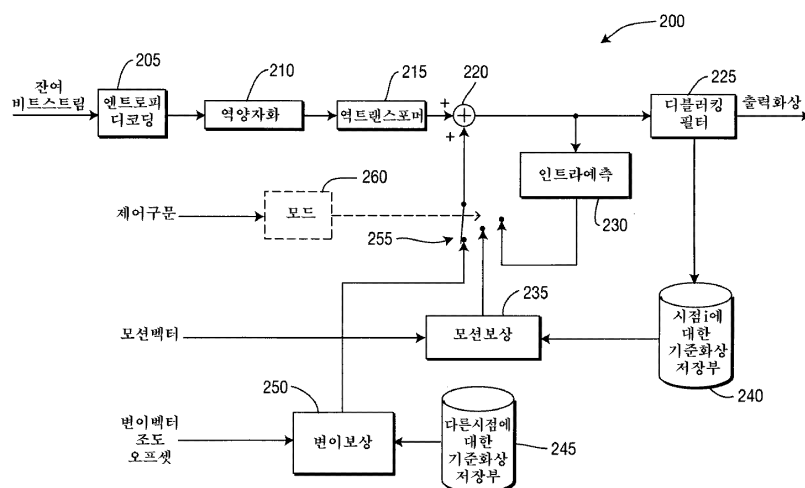
심사관 : 박상철

(54) 발명의 명칭 다시점 비디오 코딩시스템에 있어 비디오 사용성 정보(VUI)를 통합하기 위한 방법 및 장치

(57) 요약

다시점 비디오 코딩(multi-view video coding, MVC) 내 비디오 사용성 정보(VUI)를 통합하기 위한 방법 및 장치가 제공된다. 일 장치(100)가 개별적인 시점(300), 시점 내 개별적인 시간 레벨(500), 및 개별적인 동작점(700)으로부터 선택된 적어도 하나에 대한 비디오 사용성 정보를 규정함으로써 다시점 비디오 콘텐츠를 인코딩하기 위한 인코더(100)를 포함한다. 또한, 일 장치(200)가 개별적인 시점(400), 시점 내 개별적인 시간 레벨(600), 및 개별적인 동작점(800)으로부터 선택된 적어도 하나에 대한 비디오 사용성 정보를 규정함으로써 다시점 비디오 콘텐츠를 디코딩하기 위한 디코더를 포함한다.

대표도



명세서

청구범위

청구항 1

다시점(multi-view) 비디오 코딩에서 개별적인 동작점(operating point)들에 대한 비트스트림 제한 정보를 특정하여 다시점 비디오 콘텐츠를 디코딩하기 위한 디코더(200) - 동작점은 하나 이상의 시점들 및 시간 레벨의 조합에 대응함 -

를 포함하고,

상기 비트스트림 제한 정보는,

- (1) 모션 벡터가 화상 경계(picture boundary)에 걸쳐 있는지 여부;
- (2) 매크로블록 당 최대 비트;
- (3) 최대 수평 모션 벡터 길이 및 최대 수직 모션 벡터 길이; 및
- (4) 프레임 재배치 수(number of reordering frames)

중 하나 이상을 나타내는,

장치.

청구항 2

다시점 비디오 코딩에서 개별적인 동작점들에 대한 비트스트림 제한 정보를 특정하여 다시점 비디오 콘텐츠를 디코딩하기 위한 단계(800) - 동작점은 하나 이상의 시점들 및 시간 레벨의 조합에 대응함 -

를 포함하고,

상기 비트스트림 제한 정보는,

- (1) 모션 벡터가 화상 경계(picture boundary)에 걸쳐 있는지 여부;
- (2) 매크로블록 당 최대 비트;
- (3) 최대 수평 모션 벡터 길이 및 최대 수직 모션 벡터 길이; 및
- (4) 프레임 재배치 수(number of reordering frames)

중 하나 이상을 나타내는,

방법.

청구항 3

삭제

청구항 4

삭제

청구항 5

삭제

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

발명의 설명

기술 분야

[0001]

본 출원은 U.S. Provisional Application Serial No. 60/977,709, filed October 5, 2007에 대한 우선권을 주장하며, 이것은 여기서 그 전체가 참조로서 통합된다. 또한, 본 출원은 non-provisional application, Attorney Docket No. PU070239, entitled "METHOD AND APPARATUS FOR INCORPORATING VIDEO USABILITY INFORMATION(VUI) WITHIN A MULTI-VIEW VIDEO(MVC) CODING SYSTEM"과 관련되고, 이 또한 U.S. Provisional Application Serial No. 60/977,709, filed October 5, 2007에 대한 우선권을 주장하며, 이것은 여기서 참조로서 널리 할당 및 통합되고 이와함께 동시에 제출된다.

[0002]

본 발명은 일반적으로 비디오 인코딩 및 디코딩에 관련되고, 더욱 구체적으로는 다시점 비디오 코딩(multi-view video coding, MVC)에 있어 비디오 사용성 정보(video usability information, VUI)를 통합하기 위한 방법 및 장치에 관련된다.

배경 기술

[0003]

ISO/IEC(International Organization for Standardization/International Electrotechnical Commission) MPEG-4(Moving Picture Experts Group-4) Part10 AVC(Advanced Video Coding) standard/ITU-T(International Telecommunication Union, Telecommunication Sector) H.264 recommendation(이하, "MPEG-4 AVC 표준"이라 함)은 시퀀스 파라미터 세트의 비디오 사용성 정보(VUI) 파라미터의 구문법(syntax)과 의미(semantics)를 상술한다. 비디오 사용성 정보는 화상비(aspect ratio), 오버스캐닝, 비디오 신호 타입, 색도 위치(chroma location), 타이밍, NAL(네트워크 추상계층, network abstraction layer) 가상 레퍼런스 디코더(hypothetical reference decoder, HRD) 파라미터, VCL(video coding layer) 가상 레퍼런스 디코더 파라미터, 비트스트림 제한 등의 정보를 포함한다. 비디오 사용성 정보는 사용자를 위한 보다 폭넓은 어플리케이션을 가능하게 하기 위하여 대응 비트스트림에 대한 추가적인 정보를 제공한다. 예를 들어, 비트 스트림 제한 정보(bitstream restriction information)에서, 비디오 사용성 정보는 (1) 모션이 하나의 화상 경계(picture boundary)에 걸쳐 있는지 여부; (2) 화상(picture) 당 최대 바이트; (3) 매크로블럭 당 최대 비트; (4) 최대 모션 벡터 길이(수평 및 수직); (5) 프레임 재배치 수(the number of reordering frames); 및 최대 디코딩된 프레임 버퍼 크기를 상술한다. 디코더가 상기 정보를 확인하면, 디코딩 요구조건을 설정하기 위해 "레벨(level)" 정보를 사용하는 대신(이것은 비트스트림이 실제로 요구하는 것보다 통상적으로 더 높음), 디코더는 더 엄격한 제한요건에 기초하여 디코딩 동작을 맞춤형설정(customize)할 수 있다.

[0004] 다시점 비디오 코딩(MVC)은 MPEG-4 AVC 표준에 대한 확장이다. 다시점 비디오 코딩에서, 다수의 시점(multiple views)에 대한 비디오 이미지는 시점들 간의 상호 관련성을 이용하여 인코딩될 수 있다. 모든 시점 중에서, 하나의 시점은 기본 시점(base view)인데, 이것은 MPEG-4 AVC 표준 호환성이 있고 다른 시점들로부터는 예측될 수 없다. 다른 시점들은 비기본 시점(non-base views)으로 일컬어진다. 비기본 시점은 상기 기본 시점 및 다른 비기본 시점으로부터 예측적으로 인코딩될 수 있다. 각 시점은 시간적으로(temporally) 서브샘플(sub-sample)링될 수 있다. 어떤 시점의 시간 서브세트(temporal subset)는 일시적_id(temporal_id) 구문 요소에 의해 확인될 수 있다. 어떤 시점의 시간 레벨(temporal level)은 비디오 신호의 한 표현이다. 다시점 비디오 코딩된 스트림 내에는 시점 및 시간 레벨의 상이한 조합이 있을 수 있다. 각각의 조합은 동작점(operation point)라 불린다. 상기 동작점에 대응하는 서브-비트스트림은 비트스트림으로부터 추출될 수 있다.

발명의 내용

해결하려는 과제

[0005] 종래 기술에 대한 상기 및 다른 단점 및 불편함이 본 발명 사상에 의해 제기되며, 본 발명 사상은 다시점 비디오 코딩(MVC)에 있어 비디오 사용성 정보(VUI)를 통합하기 위한 방법 및 장치와 관련된다.

과제의 해결 수단

[0006] 본 발명의 일 측면에 따라 일 장치가 제공된다. 그 장치는 개별적인 시점, 시점 내 개별적인 시간 레벨, 및 개별적인 동작점 중의 적어도 하나에 대한 비디오 사용성 정보를 규정함(specifying)으로써 다시점 비디오 콘텐츠를 인코딩하기 위한 인코더를 포함한다.

[0007] 본 발명의 또 다른 측면에 따라 일 방법이 제공된다. 그 방법은 개별적인 시점, 시점 내 개별적인 시간 레벨, 및 개별적인 동작점 중의 적어도 하나에 대한 비디오 사용성 정보를 규정함으로써 다시점 비디오 콘텐츠를 인코딩하는 단계를 포함한다.

[0008] 본 발명의 또 다른 측면에 따라, 일 장치가 제공된다. 그 장치는 개별적인 시점, 시점 내 개별적인 시간 레벨, 및 개별적인 동작점 중의 적어도 하나에 대한 비디오 사용성 정보를 규정함으로써 다시점 비디오 콘텐츠를 디코딩하기 위한 디코더를 포함한다.

[0009] 본 발명의 또 다른 측면에 따라 일 방법이 제공된다. 그 방법은 개별적인 시점, 시점 내 개별적인 시간 레벨, 및 개별적인 동작점 중의 적어도 하나에 대한 비디오 사용성 정보를 규정함으로써 다시점 비디오 콘텐츠를 디코딩하는 단계를 포함한다.

[0010] 본 발명의 상기 및 다른 측면, 특징, 장점은 예시적인 실시예에 관한 이하의 상세한 설명으로부터 명백해질 것이며, 이는 첨부된 도면과 관련하여 설명될 것이다.

발명의 효과

[0011] 본 발명에 따르면, 종래의 문제점을 효과적으로 개선한 비디오 사용성 정보(VUI)를 통합하기 위한 장치, 방법, 비디오 신호구조 등을 제공할 수 있다.

도면의 간단한 설명

[0012] 본 발명은 이하의 실시예적인 도면에 따라 더 잘 이해될 수 있다.

도 1은, 본 발명의 일 실시예에 따라, 본 발명이 적용될 수 있는 예시적인 다시점 비디오 코딩(MVC) 인코더에 대한 블록도이다.

도 2는, 본 발명의 일 실시예에 따라, 본 발명이 적용될 수 있는 예시적인 다시점 비디오 코딩(MVC) 디코더에 대한 블록도이다.

도 3은, 본 발명의 일 실시예에 따라, `mvc_vui_parameters_extension()` 구문요소를 이용하여, 각 시점에 대한 비트스트림 제한 파라미터를 인코딩하기 위한 예시적인 방법에 관한 흐름도이다.

도 4는, 본 발명의 일 실시예에 따라, `mvc_vui_parameters_extension()` 구문요소를 이용하여, 각 시점에 대한 비트스트림 제한 파라미터를 디코딩하기 위한 예시적인 방법에 관한 흐름도이다.

도 5는, 본 발명의 일 실시예에 따라, `mvc_vui_parameters_extension()` 구문요소를 이용하여, 각 시점에서의 각각의 시간 레벨(temporal level)에 대한 비트스트림 제한 파라미터를 인코딩하기 위한 예시적인 방법에 관한 흐름도이다.

도 6은, 본 발명의 일 실시예에 따라, `mvc_vui_parameters_extension()` 구문요소를 이용하여, 각 시점에서의 각각의 시간 레벨에 대한 비트스트림 제한 파라미터를 디코딩하기 위한 예시적인 방법에 관한 흐름도이다.

도 7은, 본 발명의 일 실시예에 따라, `view_scalability_parameters_extension()` 구문요소를 이용하여, 각 동작점에 대한 비트스트림 제한 파라미터를 인코딩하기 위한 예시적인 방법에 관한 흐름도이다.

도 8은, 본 발명의 일 실시예에 따라, `view_scalability_parameters_extension()` 구문요소를 이용하여, 각 동작점에 대한 비트스트림 제한 파라미터를 디코딩하기 위한 예시적인 방법에 관한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

- [0013] 본 발명은 다시점 비디오 코딩(MVC)에 있어 비디오 사용성 정보(VUI)를 통합하기 위한 방법 및 장치에 관련된다.
- [0014] 본 명세서는 본 발명에 대하여 설명한다. 여기에 명확하게 기술되거나 도시되지 않았다 하더라도, 본 발명의 당업자라면 본 발명을 구체화하고 그 사상 및 범위에 속하는 다양한 구성들을 고안할 수 있다는 것을 알 수 있을 것이다.
- [0015] 여기에 언급된 모든 실시예와 조건적인 언어는 독자가 발명자에 의하여 제공되는 본 발명의 사상 및 개념들을 이해하는데 도움을 주기 위한 교육적인 목적으로 제공되며, 상세하게 언급되는 실시예들과 조건들에 한정되지 않는 것으로 이해될 것이다.
- [0016] 또한, 구체적인 실시예뿐만 아니라 본 발명의 사상, 측면 및 실시예들을 언급하는 모든 기재내용은 구조적 및 기능적인 균등물을 포함한다. 또한, 상기 균등물은 미래에 개발될 균등물뿐만 아니라 현재 알려진 균등물, 즉 구조와 상관없이 동일한 작용을 수행하는 개발된 어떠한 구성요소들도 포함한다.
- [0017] 따라서, 예를 들어, 당업자라면 여기에 제시된 블록도가 본 발명을 구체화하는 도시적인 회로에 관한 개념적인 관점을 제공한다는 점도 알 수 있을 것이다. 마찬가지로, 어떠한 흐름도, 상태 천이도, 의사코드(pseudocode) 등도, 컴퓨터 판독 가능 매체에 실질적으로 제공될 수 있고 컴퓨터나 프로세서에 의해 실행가능한, 다양한 프로세스를 (이러한 컴퓨터나 프로세서가 명시적으로 개시되어 있는지에 상관없이) 나타낸다.
- [0018] 도면에 도시된 다양한 구성요소의 작용들은 적절한 소프트웨어와 관련하여 소프트웨어를 실행할 수 있는 하드웨어뿐만 아니라 전용 하드웨어를 사용함으로써 제공될 수 있다. 프로세서에 의해 제공될 때, 상기 작용들은 단일 전용 프로세서, 단일 공용 프로세서 또는 다수의 개별 프로세서(이 중 일부는 공유될 수 있다)에 의하여 제공될 수 있다. 또한, "프로세서" 또는 "제어기"라는 용어의 명시적인 사용은 소프트웨어를 실행할 수 있는 하드웨어를 배타적으로 언급하는 것으로 해석되면 안 되고, 어떤 제한도 없이 디지털 신호 프로세서("DSP") 하드웨어, 소프트웨어를 저장하기 위한 ROM(read-only memory), RAM(random access memory) 및 비휘발성 저장매체를 암시적으로 포함할 수 있다.
- [0019] 통상적 및/또는 맞춤형의 다른 하드웨어도 포함될 수 있다. 마찬가지로, 도면에 도시된 어떠한 스위치도 단지 개념적인 것이다. 그 기능은 프로그램 로직의 동작을 통해, 전용 로직을 통해, 프로그램 제어의 상호작용과 전용 로직을 통해, 또는 심지어 수동적으로 수행될 수 있으며, 그 구체적인 기술은 문맥으로부터 더욱 자세하게 이해될 수 있는 바와 같이 실행자에 의하여 선택될 수 있다.
- [0020] 청구항에 있어서, 어떤 특정 기능을 수행하기 위한 수단으로 표현된 어떠한 구성요소도 상기 기능을 수행하는 어떠한 방법도 포함하도록 의도되는데, 상기 기능을 수행하는 어떠한 방법은 예를 들어 a) 그 기능을 수행하는 회로요소들의 조합, 또는 b) 그 기능을 수행하기 위한 소프트웨어를 실행하기 위한 적절한 회로와 결합되는, 펌웨어, 마이크로코드 등을 포함하는, 어떠한 형태의 소프트웨어도 포함한다. 상기와 같은 청구항에 의하여 정의

되는 본 발명은, 언급된 다양한 수단에 의하여 제공되는 기능들이 상기 청구항이 요구하는 바에 따라 결합되고 통합된다는 사실에 귀속된다. 따라서, 상기 기능을 제공할 수 있는 어떠한 수단이라도 여기에 도시되는 것과 균등한 것으로 이해된다.

- [0021] 명세서에서 본 발명의 "일 실시예" 또는 "어떤 실시예"로 언급된 것은 그 실시예와 관련하여 언급된 특정 특징, 구조, 특성 등이 본 발명의 적어도 하나의 실시예에 포함된다는 것을 의미한다. 따라서, 명세서를 통하여 다양한 곳에서 나타나는 "일 실시예에서" 또는 "어떤 실시예에서"라는 문구는 모두 동일한 실시예를 필연적으로 언급하는 것은 아니다.
- [0022] 예를 들어 "A 및/또는 B" 및 "A와 B 중의 적어도 하나"와 같이, "및/또는"과 "적어도 하나"라는 용어의 사용은 첫번째로 나열된 옵션 (A)만의 선택 또는 두번째로 나열된 옵션 (B)만의 선택, 또는 옵션 (A와 B) 모두의 선택을 포함하도록 의도된다. 또 다른 예로서 "A, B 및/또는 C" 및 "A, B 및 C 중의 적어도 하나"의 경우, 이러한 문구는 첫번째로 나열된 옵션 (A)만의 선택 또는 두번째로 나열된 옵션 (B)만의 선택, 또는 세번째로 나열된 옵션 (C)만의 선택, 또는 첫번째와 두번째로 나열된 옵션 (A와 B)만의 선택, 또는 첫번째와 세번째로 나열된 옵션 (A와 C)만의 선택, 또는 두번째와 세번째로 나열된 옵션 (B와 C)만의 선택, 또는 세가지 옵션 모두 (A와 B와 C)의 선택을 포함하도록 의도된다. 이것은, 본 기술분야 또는 관련기술분야의 당업자에게 명확한 바와 같이, 많은 항목이 나열된 경우에 대해 확장될 수 있다.
- [0023] 다시점 비디오 코딩(MVC)은 다시점 시퀀스의 인코딩에 대한 압축 체제(compression framework)이다. 다시점 비디오 코딩(MVC) 시퀀스는 상이한 시점으로부터의 동일한 장면을 캡처한 2 또는 그 이상의 비디오 시퀀스의 조합이다.
- [0024] 여기서 호환적으로 사용된 바와 같이, "크로스 뷰(cross-view)" 및 "시점 간(inter-view)"이란 현재 시점 외의 어떤 한 시점에 속하는 화상을 말한다.
- [0025] 또한, 여기서 사용되는 바와 같이, "고급 레벨 구문(high level syntax)"은 매크로블럭 계층 상에 계층적으로 존재하는 비트스트림 내에 존재하는 구문(syntax)을 말한다. 예를 들어, 여기서 사용된 바와 같이, 고급 레벨 구문은 슬라이스 헤더 레벨(slice header level), 보충강화정보(supplemental enhancement information, SEI) 레벨, 화상 파라미터 세트(picture parameter set, PPS) 레벨, 시퀀스 파라미터 세트(PPS) 레벨 및 네트워크 추상계층(network abstraction layer, NAL) 유닛 헤더 레벨에서의 구문을 말할 수 있으나, 이에 한정되지는 않는다.
- [0026] 또한, 본 발명의 하나 또는 그 이상의 실시예가 MPEG-4 AVC 표준의 다시점 비디오 코딩 확장과 관련된 예시적인 목적에 대해 여기에 기재되어 있지만, 본 발명은 이러한 확장 및/또는 상기 표준에만 한정되지는 않으며, 따라서 본 발명의 사상을 유지하는 한 다른 비디오 코딩표준, 권고 및 그 확장과 관련하여 이용될 수 있다.
- [0027] 또한, 본 발명의 하나 또는 그 이상의 실시예가 비트스트림 제한 정보와 관련된 예시적인 목적에 대하여 여기에 기재되어 있지만, 본 발명은 비트스트림 제한 정보를 비디오 사용성 정보의 한 타입으로서 사용하는 것에만 한정되지는 않으며, 따라서 본 발명의 사상을 유지하는 한, 다시점 비디오 코딩에 관한 사용에 대해 확장될 수 있는 다른 타입의 비디오 사용성 정보도 본 발명에 따라 사용될 수 있다.
- [0028] 도 1에 있어, 실시예적인 다시점 비디오 코딩(MVC) 인코더가 참조번호 100에 의하여 전체적으로 지시되어 있다. 인코더(100)는 트랜스포머(transformer, 110)의 입력과 신호 통신적으로 연결된 출력을 갖는 결합기(combiner, 105)를 포함한다. 트랜스포머(110)의 출력은 양자화기(quantizer, 115)의 입력과 신호 통신적으로 연결된다. 양자화기(115)의 출력은 엔트로피 코더(entropy coder, 120)의 입력 및 역양자화기(inverse quantizer, 125)의 입력과 신호 통신적으로 연결된다. 역양자화기(125)의 출력은 역트랜스포머(inverse transformer, 130)의 입력과 신호통신적으로 연결된다. 역트랜스포머(130)의 출력은 결합기(135)의 제 1 비반전 입력과 신호통신적으로 연결된다. 결합기(135)의 출력은 인트라예측기(intra predictor, 145)의 입력 및 디블러킹 필터(deblocking filter, 150)의 입력과 신호통신적으로 연결된다. 디블러킹 필터(150)의 출력은 (시점 i에 대한) 기준화상 저장부(reference picture store, 155)의 입력과 신호통신적으로 연결된다. 기준화상 저장부(155)의 출력은 모션보상기(175)의 제 1 입력 및 모션 평가기(motion estimator, 180)의 제 1 입력과 신호통신적으로 연결된다. 모션 평가기(180)의 출력은 모션보상기(175)의 제 2 입력과 신호통신적으로 연결된다.
- [0029] (다른 시점에 대한) 기준화상 저장부(160)의 출력은 변이/조도 평가기(disparity/illumination estimator, 170)의 제 1 입력 및 변이/조도 보상기(165)의 제 1 입력과 신호통신적으로 연결된다. 변이/조도 평가기(170)의 출력은 변이/조도 보상기(165)의 제 2 입력과 신호통신적으로 연결된다.

- [0030] 엔트로피 디코더(120)의 출력은 인코더(100)의 출력으로서 사용가능하다. 결합기(105)의 비반전 입력은 인코더(100)의 입력으로서 사용가능하고, 변이/조도 평가기(170)의 제 2 입력 및 모션 평가기(180)의 제 2 입력과 신호통신적으로 연결된다. 스위치(185)의 출력은 결합기(135)의 제 2 비반전 입력 및 결합기(105)의 반전 입력과 신호통신적으로 연결된다. 스위치(185)는 모션 보상기(175)의 출력과 신호통신적으로 연결되는 제 1 입력과, 변이/조도 보상기(165)의 출력과 신호통신적으로 연결되는 제 2 입력, 및 인트라예측기(145)의 출력과 신호통신적으로 연결되는 제 3 입력을 포함한다.
- [0031] 모드결정 모듈(mode decision module, 140)은 어떤 입력이 스위치(185)에 의하여 선택되는지를 제어하기 위하여 스위치(185)에 연결되는 출력을 갖는다.
- [0032] 도 2에 있어서, 실시예적인 다시점 비디오 코딩(MVC) 디코더가 참조번호 200에 의하여 전체적으로 지시되어 있다. 디코더(200)는 역양자화기(210)의 입력과 신호통신적으로 연결된 출력을 갖는 엔트로피 디코더(entropy decoder, 205)를 포함한다. 상기 역양자화기의 출력은 역트랜스포머(215)의 입력과 신호통신적으로 연결된다. 역트랜스포머(215)의 출력은 결합기(220)의 제 1 비반전입력과 신호통신적으로 연결된다. 결합기(220)의 출력은 디블러킹 필터(225)의 입력 및 인트라예측기(230)의 입력과 신호통신적으로 연결된다. 디블러킹 필터(225)의 출력은 (시점 i에 대한) 기준화상 저장부(240)의 입력과 신호통신적으로 연결된다. 기준화상 저장부(240)의 출력은 모션보상기(235)의 제 1 입력과 신호통신적으로 연결된다.
- [0033] (다른 시점에 대한)기준화상 저장부(245)의 출력은 변이/조도 보상기 (250)의 제 1 입력과 신호통신적으로 연결된다.
- [0034] 엔트로피 코더(205)의 입력은 잔여 비트스트림(residue bitstream)을 수신하기 위하여 디코더(200)의 입력으로서 사용가능하다. 또한, 모드모듈(mode module, 260)의 입력은 어떤 입력이 스위치(255)에 의하여 선택되지는 제어하기 위한 제어구문(control syntax)을 수신하기 위해 디코더(200)의 입력으로서도 사용가능하다. 그리고, 모션 보상기(235)의 제 2 입력은 모션벡터를 수신하기 위해 디코더(200)의 입력으로서 사용가능하다. 또한, 변이/조도 보상기(250)의 제 2 입력은 변이벡터(disparity vectors)와 조도보상 구문(syntax)를 수신하기 위해, 디코더(200)으로의 입력으로서 사용가능하다.
- [0035] 스위치(255)의 출력은 결합기(220)의 제 2 비반전입력과 신호통신적으로 연결된다. 스위치(255)의 제 1 입력은 변이/조도 보상기(250)의 출력과 신호통신적으로 연결된다. 스위치(255)의 제 2 입력은 모션보상기(235)의 출력과 신호통신적으로 연결된다. 스위치(255)의 제 3입력은 인트라예측기(230)의 출력과 신호통신적으로 연결된다. 모드모듈(260)의 출력은 어떤 입력이 스위치(255)에 의하여 선택되는지 제어하기 위하여 스위치(255)와 신호통신적으로 연결된다. 디블러킹 필터(225)의 출력은 상기 디코더의 출력으로서 사용가능하다.
- [0036] MPEG-4 AVC 표준에는 시퀀스 파라미터 세트의 구문 및 의미 파라미터(syntax and semantic parameters)가 비디오 사용성 정보(VUI)를 위해 규정되어 있다. 이것은 폭넓은 다양한 목적을 위해 비디오의 사용성을 강화하기 위하여 비트스트림에 삽입될 수 있는 추가적인 정보를 나타낸다. 비디오 사용성 정보는 화상비(aspect ratio), 오버스캐닝, 비디오 신호 타입, 색도 위치(chroma location), 타이밍, NAL(network abstraction layer) 가상 레퍼런스 디코더(hypothetical reference decoder, HRD) 파라미터, VCL(video coding layer) 가상 레퍼런스 디코더 파라미터, 비트스트림 제한 등의 정보를 포함한다.
- [0037] 본 발명의 하나 또는 그 이상의 실시예에 따르면, 우리는 종래 기술과는 다른 새로운 목적을 위해 상기 현존하는 비디오 사용성 정보를 사용하고, 그 사용을 다시점 비디오 코딩(MVC)으로 확장한다. 본 다시점 비디오 코딩 스킵에서, 비디오 사용성 정보는 예를 들어 상이한 시점 간, 한 시점 내의 상이한 시간 레벨 간, 및 상이한 동작점 간에 다를 수 있도록 확장된다. 따라서, 일 실시예에 따라, 우리는 이하의 단계 중 하나 또는 그 이상에 따르되 이에 한정되지는 않는 비디오 사용성 정보를 규정한다: 개별적인 시점에 대한 상기 비디오 사용성 정보를 규정(specifying)하는 단계; 시점 내 개별적인 시간 레벨에 대한 상기 비디오 사용성 정보를 규정하는 단계; 및 개별 동작점에 대한 상기 비디오 사용성 정보를 개별적으로 규정하는 단계.
- [0038] MPEG-4 AVC 표준에서, 비디오 사용성 정보를 포함하는 한 세트가 시퀀스 파라미터 세트(SPS) 내에 전송될 수 있다. 일 실시예에 따라, 우리는 다시점 비디오 코딩(MVC) 환경에서의 사용을 위해 비디오 사용성 정보의 개념을 확장한다. 유익하게도, 이것은 다시점 비디오 코딩에서 상이한 시점, 시점 내 상이한 시간 레벨 또는 상이한 동작점에 대하여 상이한 비디오 사용성 정보가 규정될 수 있도록 한다. 일 실시예에서, 우리는 다시점 비디오 코딩에 대한 비디오 사용성 정보 내의 비트스트림 제한 정보를 고려, 수정 및 사용함에 있어 새로운 접근법을 제공한다.

[0039] MPEG-4 AVC 표준에서 비트스트림 제한 정보는 sequence_parameter_set()의 일부분인 vui_parameters() 구문요소에 규정되어 있다. 표 1은 vui_parameters()의 MPEG-4 AVC 표준 구문을 나타낸다.

표 1

vui_parameters() {	C	Descriptor
aspect_ratio_info_present_flag	0	u(1)
...		
bitstream_restriction_flag	0	u(1)
if(bitstream_restriction_flag) {		
motion_vectors_over_pic_boundaries_flag	0	u(1)
max_bytes_per_pic_denom	0	ue(v)
max_bits_per_mb_denom	0	ue(v)
log2_max_mv_length_horizontal	0	ue(v)
log2_max_mv_length_vertical	0	ue(v)
num_reorder_frames	0	ue(v)
max_dec_frame_buffering	0	ue(v)
}		
}		

[0040]

[0041] 비트스트림 제한 정보의 구문 요소의 의미는 다음과 같다:

[0042] 1과 같은 값을 갖는 bitstream_restriction_flag는 다음의 코딩된 비디오 시퀀스 비트스트림 제한 파라미터가 존재한다는 것을 규정한다.

[0043] 0과 같은 값을 갖는 bitstream_restriction_flag는 다음의 코딩된 비디오 시퀀스 비트스트림 제한 파라미터가 존재하지 않는다는 것을 규정한다.

[0044] 0과 같은 값을 갖는 motion_vectors_over_pic_boundaries_flag는, 화상 경계(picture boundaries) 외부의 어떠한 샘플 및 부분적 샘플 지점(fractional sample position)에서의 어떠한 샘플도 어떤 샘플을 인터 예측하기 위하여 사용되지 않는다는 것을 나타내며, 상기 부분적 샘플 지점의 값은 화상 경계 외부의 하나 또는 그 이상의 샘플을 이용하여 얻어진다.

[0045] 1과 같은 값을 갖는 motion_vectors_over_pic_boundaries_flag는, 화상 경계 외부의 하나 또는 그 이상의 샘플이 인터 예측(inter prediction)에 사용될 수 있다는 것을 나타낸다. motion_vectors_over_pic_boundaries_flag 구문요소가 존재하지 않으면, motion_vectors_over_pic_boundaries_flag 값은 1과 같은 것으로 추정될 것이다.

[0046] max_bytes_per_pic_denom은, 코딩된 비디오 시퀀스 내의 코딩된 화상과 관련된 VCL(가상 코딩계층, virtual coding layer) NAL(네트워크 추상계층, network abstraction layer) 유닛의 크기의 합에 의해 초과되지 않는, 바이트 수를 나타낸다.

[0047] 네트워크 추상계층 유닛 스트림(network abstraction layer unit stream) 내의 화상을 나타내는 상기 바이트 수는, 본 목적을 위해 상기 화상에 대한 가상 코딩계층 네트워크 추상계층(virtual coding layer network abstraction layer) 유닛 데이터의 총 바이트 수(즉, 가상 코딩계층 네트워크 추상계층 유닛에 대한 NumBytesInNALunit 변수의 총계)로서 규정된다. max_bytes_per_pic_denom의 값은 0 내지 16(포함)의 범위가 된다.

[0048] max_bytes_per_pic_denom에 따라 다음이 적용된다:

[0049] -만약 max_bytes_per_pic_denom가 0과 같으면, 제한 사항이 없다.

[0050] -만약 그렇지 않으면(max_bytes_per_pic_denom이 0이 아니면), 이하의 바이트 수 이상에 걸쳐서, 어떠한 코딩된 화상도 상기 코딩된 비디오 시퀀스 내에 나타나지 않는다.

[0051] $(PicSizeInMbs * RawMbBits) \div (8 * max_bytes_per_pic_denom)$

- [0052] max_bytes_per_pic_denom 구문요소가 존재하지 않는 경우, max_bytes_per_pic_denom의 값은 2와 같은 것으로 추정된다. 변수 PicSizeInMbs는 화상 내의 매크로블록의 수이다. 변수 RawMbits는 MPEG-4 AVC 표준의 하위조항(sub-clause) 7.4.2.1에서와 같이 얻어진다.
- [0053] max_bits_per_mb_denom은 코딩된 비디오 시퀀스의 임의의 화상 내의 임의의 매크로블록에 대한 macroblock_layer() 데이터의 코딩된 최대 비트 수를 나타낸다. max_bits_per_mb_denom의 값은 0 내지 16(포함)의 범위에 있다.
- [0054] max_bits_per_mb_denom에 따라 다음이 적용된다:
- [0055] - 만약 max_bits_per_mb_denom가 0과 같으면, 제한 사항이 없다.
- [0056] - 만약 그렇지 않으면(max_bits_per_mb_denom이 0이 아니면), 이하의 비트 수 이상에 걸쳐서, 어떠한 코딩된 macroblock_layer()도 상기 비트스트림 내에 나타나지 않는다.
- [0057] $(128 + \text{RawMbits}) \div \text{max_bits_per_mb_denom}$
- [0058] entropy_coding_mode_flag에 따라, macroblock_layer() 데이터의 비트가 다음과 같이 카운팅된다.
- [0059] -만약 entropy_coding_mode_flag가 0과 같으면, macroblock_layer() 데이터의 비트 수는 매크로블록에 대한 macroblock_layer() 구문 구조 내의 비트 수에 의해 주어진다.
- [0060] -만약 그렇지 않으면(entropy_coding_mode_flag가 1과 같으면), 매크로블록에 대한 macroblock_layer() 데이터의 비트 수는, 상기 매크로블록과 관련된 macroblock_layer()를 분석할 때, MPEG-4 AVC 표준의 하위조항 9.3.3.2.2 및 9.3.3.2.3에서 read_bits(1)가 호출되는 회수에 의해 주어진다.
- [0061] max_bits_per_mb_denom이 존재하지 않는 경우, max_bits_per_mb_denom의 값은 1과 같은 것으로 추정될 것이다.
- [0062] log2_max_mv_length_horizontal 및 log2_max_mv_length_vertical은 코딩된 비디오 시퀀스 내의 모든 화상에 대하여 1/4 휘도샘플 유닛(luma sample units) 내 디코딩된 수평 및 수직 모션벡터 성분의 최대 절대값을 각각 나타낸다. n 값은, 모션 벡터 성분의 어떤 값도 1/4 휘도 샘플 배치 유닛 내에서 -2^n 내지 2^n-1 (포함)의 범위를 넘어서지 않는다는 것을 말한다. log2_max_mv_length_horizontal의 값은 0 내지 16(포함)의 범위에 있다. log2_max_mv_length_vertical의 값은 0 내지 16(포함)의 범위에 있다. log2_max_mv_length_horizontal이 존재하지 않는 경우, log2_max_mv_length_horizontal 및 log2_max_mv_length_vertical의 값은 16과 같은 것으로 추정된다. 디코딩된 수직 또는 수평 모션 벡터 성분의 최대 절대값은 또한 MPEG-4 AVC 표준의 Annex A에 규정된 바와 같이 프로파일 및 레벨 제한(profile and level limits)에 의해 구속된다.
- [0063] num_reorder_frames는, 상기 코딩된 비디오 시퀀스 내의 어떤 프레임, 상보적 필드쌍(complementary field pair) 또는 논페어드 필드(non-paired field)보다도 디코딩 순위에 있어 각각 선행하고 출력순위에 있어 그것을 따르는, 프레임, 상보적 필드쌍 또는 논페어드 필드의 최대 개수를 나타낸다. num_reorder_frames의 값은 0 내지 max_dec_frame_buffering(포함)의 범위에 있다. num_reorder_frames 구문 요소가 존재하지 않는 경우, num_reorder_frames의 값은 다음과 같이 추정된다:
- [0064] -만약 profile_idc가 44, 100, 110, 122 또는 244와 같고 constraint_set3_flag가 1과 같으면, num_reorder_frames의 값은 0과 같은 것으로 추정된다.
- [0065] -만약 그렇지 않으면(profile_idc가 44, 100, 110, 122 또는 244와 같지 않거나 constraint_set3_flag가 0과 같으면), num_reorder_frames의 값은 max_dec_frame_bufferingMaxDpbSize와 같은 것으로 추정된다.
- [0066] max_dec_frame_buffering은 프레임 버퍼 유닛 내 가상 레퍼런스 디코더 디코딩 화상 버퍼(decoded picture buffer, DPB)의 요구되는 크기를 규정한다. 디코딩된 비디오 시퀀스는, 화상 타이밍 SEI(Supplemental Enhancement Information) 메시지의 dpb_output_delay에 의해 규정되는 출력 횟수로 디코딩 화상의 출력을 가능하게 하기 위하여, $\text{Max}(1, \text{max_dec_frame_buffering})$ 프레임 버퍼보다 더 큰 크기의 디코딩 화상 버퍼를 요구하지는 않는다. max_dec_frame_buffering의 값은 (MPEG-4 AVC 표준의 하위조항 A.3.1 또는 A.3.2에 규정된 바와 같이) num_ref_frames 내지 MaxDpbSize(포함)의 범위에 있다. max_dec_frame_buffering 구문요소가 존재하지 않으면, max_dec_frame_buffering의 값은 다음과 같이 추정된다:
- [0067] -만약 profile_idc가 44 또는 244와 같고 constraint_set3_flag가 1과 같으면, max_dec_frame_buffering의 값은 0과 같은 것으로 추정된다.

[0068] -만약 그렇지 않으면(profile_idc가 44 또는 244와 같지 않거나 constraint_set3_flag가 0과 같으면), max_dec_frame_buffering의 값은 MaxDpbsize와 같은 것으로 추정된다.

[0069] 다시점 비디오 코딩에서, 비트스트림 제한 파라미터(bitstream restriction parameters)는 보다 엄격한 제한조건에 기초하여 서브-스트림의 디코딩동작을 맞춤형설정(customize)할 수 있다. 따라서, 비트스트림 제한 파라미터는 다시점 비디오 코딩된 비트스트림의 각 추출가능한 서브-스트림에 대하여 규정될 수 있도록 할 수 있을 것이다. 일 실시예에 따르면, 우리는 각각의 시점, 시점 내 각각의 시간 레벨 및/또는 각각의 동작점에 대하여 비트스트림 제한 정보를 규정할 것을 제안한다.

[0070] 각 시점에 대한 비트스트림 제한 파라미터 규정(Specifying bitstream restriction parameters for each view)

[0071] 비트스트림 제한 파라미터는 각각의 시점에 대하여 규정될 수 있다. 우리는 mvc_vui_parameters_extension 구문을 제안하는데, 이것은 subset_sequence_parameter_set의 일부분이다. 표 2는 mvc_vui_parameters_extension 구문을 나타낸다.

[0072] mvc_vui_parameters_extension()는 상기 subset_sequence_parameter_set과 관련된 모든 시점에 걸쳐 루프를 이룬다. 각 시점의 view_id 및 각 시점의 비트스트림 제한 파라미터는 상기 루프 내에 규정된다.

표 2

mvc_vui_parameters_extension() {	C	Descriptor
num_views_minus1	0	ue(v)
for(i = 0; i <= num_views_minus1; i++) {		
view_id[i]	0	u(3)
bitstream_restriction_flag[i]	0	u(1)
if(bitstream_restriction_flag[i]) {		
motion_vectors_over_pic_boundaries_flag[i]	0	u(1)
max_bytes_per_pic_denom[i]	0	ue(v)
max_bits_per_mb_denom[i]	0	ue(v)
log2_max_mv_length_horizontal[i]	0	ue(v)
log2_max_mv_length_vertical[i]	0	ue(v)
num_reorder_frames[i]	0	ue(v)
max_dec_frame_buffering[i]	0	ue(v)
}		
}		
}		

[0073]

[0074] 비트스트림 제한 구문 요소의 의미는 다음과 같다:

[0075] bitstream_restriction_flag[i]는 view_id와 같은 view_id[i]를 갖는 시점의 bitstream restriction_flag의 값을 규정한다.

[0076] motion_vectors_over_pic_boundaries_flag[i]는 view_id와 같은 view_id[i]를 갖는 시점의 motion_vectors_over_pic_boundaries_flag의 값을 규정한다. motion_vectors_over_pic_boundaries_flag[i] 구문요소가 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점에 대한 motion_vectors_over_pic_boundaries_flag의 값은 1과 같은 것으로 추정된다.

[0077] max_bytes_per_pic_denom[i]는 view_id와 같은 view_id[i]를 갖는 시점의 max_bytes_per_pic_denom 값을 규정한다. max_bytes_per_pic_denom[i] 구문요소가 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점의 max_bytes_per_pic_denom의 값은 2와 같은 것으로 추정된다.

[0078] max_bits_per_mb_denom[i]는 view_id와 같은 view_id[i]를 갖는 시점의 max_bits_per_mb_denom 값을 규정한다. max_bits_per_mb_denom[i]이 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점의

max_bits_per_mb_denom의 값은 1과 같은 것으로 추정된다.

- [0079] log2_max_mv_length_horizontal[i] 및 log2_max_mv_length_vertical[i]는 각각 view_id와 같은 view_id[i]를 갖는 시점의 log2_max_mv_length_horizontal 및 log2_max_mv_length_vertical의 값을 규정한다. log2_max_mv_length_horizontal[i]이 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점의 log2_max_mv_length_horizontal 및 log2_max_mv_length_vertical의 값은 16과 같은 것으로 추정된다.
- [0080] num_reorder_frames[i]는 view_id와 같은 view_id[i]를 갖는 시점의 num_reorder_frames의 값을 규정한다. num_reorder_frames[i]의 값은 0 내지 max_dec_frame_buffering(포함)의 범위에 있다. num_reorder_frames[i] 구문요소가 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점의 num_reorder_frames의 값은 max_dec_frame_buffering과 같은 것으로 추정된다.
- [0081] max_dec_frame_buffering[i]는 view_id와 같은 view_id[i]를 갖는 시점의 max_dec_frame_buffering의 값을 규정한다. max_dec_frame_buffering[i]의 값은 (MPEG-4 AVC 표준의 하위조항 A.3.1 또는 A.3.2에 규정된 바와 같이) num_ref_frames[i] 내지 MaxDpbSize(포함)의 범위에 있다. max_dec_frame_buffering[i] 구문요소가 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점의 max_dec_frame_buffering의 값은 MaxDpbSize와 같은 것으로 추정된다.
- [0082] 도 3에서, mvc_vui_parameters_extension() 구문요소를 이용하여, 각 시점에 대한 비트스트림 제한 파라미터를 인코딩하기 위한 예시적인 방법이 참조번호 300에 의해 전반적으로 도시되어 있다.
- [0083] 상기 방법(300)은 기능블럭(310)에 제어를 넘기는 시작블럭(305)을 포함한다. 기능블럭(310)은 변수 M을 (시점 개수 - 1)과 같도록 설정하며, 기능블럭(315)으로 넘어가도록 제어된다. 기능블럭(315)은 비트스트림에 상기 변수 M을 기입하고, 기능블럭(320)으로 넘어가도록 제어된다. 기능블럭(320)은 변수 i를 0으로 설정하며, 기능블럭(325)으로 넘어가도록 제어된다. 기능블럭(325)은 view_id[i] 구문요소를 기입하고, 기능블럭(330)으로 넘어가도록 제어된다. 기능블럭(330)은 bitstream_restriction_flag[i] 구문요소를 기입하고, 결정블럭(335)으로 넘어가도록 제어된다. 결정블럭(335)은 bitstream_restriction_flag[i] 구문요소가 0과 같은지 여부를 결정한다. 만약 그렇다면, 결정블럭(345)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(340)으로 넘어가도록 제어된다.
- [0084] 기능블럭(340)은 시점 i의 비트스트림 제한 파라미터를 기입하고, 결정블럭(345)으로 넘어가도록 제어된다. 결정블럭(345)은 상기 변수 i가 변수 M과 같은지 여부를 결정한다. 만약 그렇다면, 종료블럭(399)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(350)으로 넘어가도록 제어된다.
- [0085] 기능블럭(350)은 변수 i를 (i + 1)과 같도록 설정하며, 기능블럭(325)으로 되돌아가도록 제어된다.
- [0086] 도 4에서, mvc_vui_parameters_extension() 구문요소를 이용하여, 각 시점에 대한 비트스트림 제한 파라미터를 디코딩하기 위한 예시적인 방법이 참조번호 400에 의하여 전반적으로 도시되어 있다.
- [0087] 상기 방법(400)은 기능블럭(407)에 제어를 넘기는 시작블럭(405)을 포함한다. 기능블럭(407)은 비트스트림으로부터 변수 M을 판독(read)하고, 기능블럭(410)으로 넘어가도록 제어된다. 기능블럭(410)은 시점의 개수를 (변수 M + 1)과 같도록 설정하며, 기능블럭(420)으로 넘어가도록 제어된다. 기능블럭(420)은 변수 i를 0으로 설정하며, 기능블럭(425)으로 넘어가도록 제어된다. 기능블럭(425)은 view_id[i] 구문요소를 판독하고, 기능블럭(430)으로 넘어가도록 제어된다. 기능블럭(430)은 bitstream_restriction_flag[i] 구문요소를 판독하고, 결정블럭(435)으로 넘어가도록 제어된다. 결정블럭(435)은 bitstream_restriction_flag[i] 구문요소가 0과 같은지 여부를 결정한다. 만약 그렇다면, 결정블럭(445)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(440)으로 넘어가도록 제어된다.
- [0088] 기능블럭(440)은 시점 i의 비트스트림 제한 파라미터를 판독하고, 결정블럭(445)으로 넘어가도록 제어된다. 결정블럭(445)은 상기 변수 i가 변수 M과 같은지 여부를 결정한다. 만약 그렇다면, 종료블럭(499)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(450)으로 넘어가도록 제어된다.
- [0089] 기능블럭(450)은 변수 i를 (i + 1)과 같도록 설정하며, 기능블럭(425)으로 되돌아가도록 제어된다.
- [0090] 각 시점의 각각의 시간 레벨에 대한 비트스트림 제한 파라미터 규정(Specifying bitstream restriction

parameters for each temporal level of each view)

[0091] 비트스트림 제한 파라미터는 각 시점의 각각의 시간 레벨(temporal level)에 대하여 규정될 수 있다. 우리는 mvc_vui_parameters_extension 구문을 제안하는데, 이것은 subset_sequence_parameter_set의 일부분이다. 표 3은 mvc_vui_parameters_extension 구문을 나타낸다.

표 3

mvc_vui_parameters_extension() {	C	Descriptor
num_views_minus1	0	ue(v)
for(i = 0; i <= num_views_minus1; i++) {		
view_id[i]	0	u(3)
num_temporal_layers_in_view_minus1[i]	0	ue(v)
for(j = 0; j <= num_temporal_level_in_view_minus1; j++) {		
temporal_id[i][j]		
bitstream_restriction_flag[i][j]	0	u(1)
if(bitstream_restriction_flag[i][j]) {		
motion_vectors_over_pic_boundaries_flag[i][j]	0	u(1)
max_bytes_per_pic_denom[i][j]	0	ue(v)
max_bits_per_mb_denom[i][j]	0	ue(v)
log2_max_mv_length_horizontal[i][j]	0	ue(v)
log2_max_mv_length_vertical[i][j]	0	ue(v)
num_reorder_frames[i][j]	0	ue(v)
max_dec_frame_buffering[i][j]	0	ue(v)
}		
}		
}		
}		

[0092]

[0093] 비트스트림 제한 구문 요소의 의미는 다음과 같다:

[0094] bitstream_restriction_flag[i][j]는 view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨의 bitstream restriction_flag 값을 규정한다.

[0095] motion_vectors_over_pic_boundaries_flag[i][j]는 view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨의 motion_vectors_over_pic_boundaries_flag의 값을 규정한다. motion_vectors_over_pic_boundaries_flag[i] 구문요소가 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 motion_vectors_over_pic_boundaries_flag 값은 1과 같은 것으로 추정된다.

[0096] max_bytes_per_pic_denom[i][j]는 view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 max_bytes_per_pic_denom 값을 규정한다. max_bytes_per_pic_denom[i] 구문요소가 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 max_bytes_per_pic_denom의 값은 2와 같은 것으로 추정된다.

[0097] max_bits_per_mb_denom[i][j]는 view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 max_bits_per_mb_denom 값을 규정한다. max_bits_per_mb_denom[i]이 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 max_bits_per_mb_denom의 값은 1과 같은 것으로 추정된다.

[0098] log2_max_mv_length_horizontal[i][j] 및 log2_max_mv_length_vertical[i][j]는 각각 view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 log2_max_mv_length_horizontal 및 log2_max_mv_length_vertical의 값을 규정한다. log2_max_mv_length_

horizontal[i]이 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 log2_max_mv_length_horizontal 및 log2_max_mv_length_vertical의 값은 16과 같은 것으로 추정된다.

[0099] num_reorder_frames[i][j]는 view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 num_reorder_frames의 값을 규정한다. num_reorder_frames[i]의 값은 0 내지 max_dec_frame_buffering(포함)의 범위에 있다. num_reorder_frames[i] 구문요소가 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 num_reorder_frames의 값은 max_dec_frame_buffering과 같은 것으로 추정된다.

[0100] max_dec_frame_buffering[i][j]는 view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 max_dec_frame_buffering의 값을 규정한다. max_dec_frame_buffering[i]의 값은 (MPEG-4 AVC 표준의 하위조항 A.3.1 또는 A.3.2에 규정된 바와 같이) num_ref_frames[i] 내지 MaxDpbSize(포함)의 범위에 있다. max_dec_frame_buffering[i] 구문요소가 존재하지 않는 경우, view_id와 같은 view_id[i]를 갖는 시점에서 temporal_id와 같은 temporal_id[i][j]를 갖는 시간 레벨에 대한 max_dec_frame_buffering의 값은 MaxDpbSize와 같은 것으로 추정된다.

[0101] mvc_vui_parameters_extension()에서는 2개의 루프가 실행된다. 외부 루프는 subset_sequence_parameter_set과 관련된 모든 시점에 걸쳐 루프를 이룬다. 각 시점의 시간 레벨의 개수에 대한 view_id는 상기 외부 루프에서 규정된다. 내부 루프는 한 시점의 모든 시간 레벨에 걸쳐 루프를 이룬다. 비트스트림 제한 정보는 상기 내부 루프에서 규정된다.

[0102] 도 5에서, mvc_vui_parameters_extension() 구문요소를 이용하여, 각 시점에서의 각각의 시간 레벨에 대한 비트스트림 제한 파라미터를 인코딩하기 위한 예시적인 방법이 참조번호 500에 의해 전반적으로 도시되어 있다.

[0103] 상기 방법(500)은 기능블럭(510)에 제어를 넘기는 시작블럭(505)을 포함한다. 기능블럭(510)은 변수 M을 (시점 개수 - 1)과 같도록 설정하며, 기능블럭(515)으로 넘어가도록 제어된다. 기능블럭(515)은 비트스트림에 상기 변수 M을 기입하며, 기능블럭(520)으로 넘어가도록 제어된다. 기능블럭(520)은 변수 i를 0으로 설정하며, 기능블럭(525)으로 넘어가도록 제어된다. 기능블럭(525)은 view_id[i] 구문요소를 기입하고, 기능블럭(530)으로 넘어가도록 제어된다. 기능블럭(530)은 변수 N을 (시점 i 내 시간 레벨의 개수 - 1)과 같도록 설정하며, 기능블럭(535)으로 넘어가도록 제어된다. 기능블럭(535)은 비트스트림에 상기 변수 N을 기입하며, 기능블럭(540)으로 넘어가도록 제어된다. 기능블럭(540)은 변수 j를 0으로 설정하며, 기능블럭(545)으로 넘어가도록 제어된다. 기능블럭(545)은 temporal_id[i][j] 구문요소를 기입하며, 기능블럭(550)으로 넘어가도록 제어된다. 기능블럭(550)은 bitstream_restriction_flag[i][j] 구문요소를 기입하고, 결정블럭(555)으로 넘어가도록 제어된다. 결정블럭(555)은 bitstream_restriction_flag[i][j] 구문요소가 0과 같은지 여부를 결정한다. 만약 그렇다면, 결정블럭(565)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(560)으로 넘어가도록 제어된다.

[0104] 기능블럭(560)은 시점 i 내 시간 레벨 j의 비트스트림 제한 파라미터를 기입하며, 결정블럭(565)으로 넘어가도록 제어된다. 결정블럭(565)은 상기 변수 j가 변수 N과 같은지 여부를 결정한다. 만약 그렇다면, 결정블럭(570)으로 넘어가도록 제어된다. 그렇지 않다면, 기능블럭(575)으로 넘어가도록 제어된다.

[0105] 결정블럭(570)은 변수 i가 변수 M과 같은지 여부를 결정한다. 만약 그렇다면, 종료블럭(599)으로 넘어가도록 제어된다. 그렇지 않다면, 기능블럭(580)으로 넘어가도록 제어된다.

[0106] 기능블럭(580)은 변수 i를 (i + 1)과 같도록 설정하며, 기능블럭(525)으로 되돌아가도록 제어된다.

[0107] 기능블럭(575)은 변수 j를 (j + 1)과 같도록 설정하며, 기능블럭(545)으로 되돌아가도록 제어된다.

[0108] 도 6에서, mvc_vui_parameters_extension() 구문요소를 이용하여, 각 시점 내 각 시간 레벨에 대한 비트스트림 제한 파라미터를 디코딩하기 위한 예시적인 방법이 참조번호 600에 의하여 전반적으로 도시되어 있다.

[0109] 상기 방법(600)은 기능블럭(607)에 제어를 넘기는 시작블럭(605)을 포함한다. 기능블럭(607)은 비트스트림으로부터 변수 M을 판독(read)하며, 기능블럭(610)으로 넘어가도록 제어된다. 기능블럭(610)은 시점의 개수를 (M + 1)과 같도록 설정하며, 기능블럭(620)으로 넘어가도록 제어된다. 기능블럭(620)은 변수 i를 0으로 설정하며, 기능블럭(625)으로 넘어가도록 제어된다. 기능블럭(625)은 view_id[i] 구문요소를 판독하며, 기능블럭(627)으로 넘어가도록 제어된다. 기능블럭(627)은 비트스트림으로부터 변수 N을 판독하며, 기능블럭(630)으로 넘어가도록

제어된다. 기능블럭(630)은 시점 i 내 시간 레벨의 수를 $(N + 1)$ 과 같도록 설정하며, 기능블럭(640)으로 넘어가도록 제어된다. 기능블럭(640)은 변수 j 를 0으로 설정하며, 기능블럭(645)으로 넘어가도록 제어된다. 기능블럭(645)은 `temporal_id[i][j]` 구문요소를 판독하며, 기능블럭(650)으로 넘어가도록 제어된다. 기능블럭(650)은 `bitstream_restriction_flag[i][j]` 구문요소를 판독하며, 결정블럭(655)으로 넘어가도록 제어된다. 결정블럭(655)은 `bitstream_restriction_flag[i][j]` 구문요소가 0과 같은지 여부를 결정한다. 만약 그렇다면, 결정블럭(665)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(660)으로 넘어가도록 제어된다.

[0110] 기능블럭(660)은 시점 i 내 시간 레벨 j 의 비트스트림 제한 파라미터를 판독하고, 결정블럭(665)으로 넘어가도록 제어된다. 결정블럭(665)은 상기 변수 j 가 변수 N 과 같은지 여부를 결정한다. 만약 그렇다면, 결정블럭(670)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(675)으로 넘어가도록 제어된다.

[0111] 결정블럭(670)은 상기 변수 i 가 변수 M 과 같은지 여부를 결정한다. 만약 그렇다면, 종료블럭(699)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(680)으로 넘어가도록 제어된다.

[0112] 기능블럭(680)은 변수 i 를 $(i + 1)$ 과 같도록 설정하며, 기능블럭(625)으로 되돌아가도록 제어된다.

[0113] 기능블럭(675)은 변수 j 를 $(j + 1)$ 과 같도록 설정하며, 기능블럭(645)으로 되돌아가도록 제어된다.

[0114] 각 동작점에 대한 비트스트림 제한 정보 규정(Specifying bitstream restriction information for each operation point)

[0115] 비트스트림 제한 파라미터는 각 동작점(operation point)에 대하여 규정될 수 있다. 우리는 시점 확장성 정보(view scalability information) SEI 메시지 내 각 동작점의 비트스트림 제한 파라미터를 운반하는 것을 제안한다. 시점 확장성 정보 SEI 메시지의 구문은 표 4에서와 같이 수정될 수 있다. 비트스트림 제한 정보의 구문은 모든 동작점에 걸쳐 루프를 이루고 있는 루프에 삽입된다.

표 4

view_scalability_info(payloadSize) {	C	Descriptor
num_operation_points_minus1	5	ue(v)
for(i = 0; i <= num_operation_points_minus1; i++) {		
operation_point_id[i]	5	ue(v)
priority_id[i]	5	u(5)
temporal_id[i]	5	u(3)
num_active_views_minus1[i]	5	ue(v)
for(j = 0; j <= num_active_views_minus1[i]; j++)		
view_id[i][j]	5	ue(v)
profile_level_info_present_flag[i]	5	u(1)
bitrate_info_present_flag[i]	5	u(1)
frm_rate_info_present_flag[i]	5	u(1)
op_dependency_info_present_flag[i]	5	u(1)
init_parameter_sets_info_present_flag[i]	5	u(1)
bitstream_restriction_flag[i]		
if(profile_level_info_present_flag[i]) {		
op_profile_idc[i]	5	u(8)
op_constraint_set0_flag[i]	5	u(1)
op_constraint_set1_flag[i]	5	u(1)
op_constraint_set2_flag[i]	5	u(1)
op_constraint_set3_flag[i]	5	u(1)
reserved_zero_4bits /* equal to 0 */	5	u(4)
op_level_idc[i]	5	u(8)
} else		
profile_level_info_src_op_id_delta[i]		ue(v)
if(bitrate_info_present_flag[i]) {		
avg_bitrate[i]	5	u(16)
max_bitrate[i]	5	u(16)
max_bitrate_calc_window[i]	5	u(16)
}		
if(frm_rate_info_present_flag[i]) {		
constant_frm_rate_idc[i]	5	u(2)
avg_frm_rate[i]	5	u(16)
} else		
frm_rate_info_src_op_id_delta[i]	5	ue(v)
if(op_dependency_info_present_flag[i]) {		
num_directly_dependent_ops[i]	5	ue(v)
for(j = 0; j < num_directly_dependent_ops[i]; j++) {		
directly_dependent_op_id_delta_minus1[i][j]	5	ue(v)
} else		
op_dependency_info_src_op_id_delta[i]	5	ue(v)
if(init_parameter_sets_info_present_flag[i]) {		

[0116]

num_init_seq_parameter_set_minus1[i]	5	ue(v)
for(j = 0; j <= num_init_seq_parameter_set_minus1[i]; j++)		
init_seq_parameter_set_id_delta[i][j]	5	ue(v)
num_init_pic_parameter_set_minus1[i]	5	ue(v)
for(j = 0; j <= num_init_pic_parameter_set_minus1[i]; j++)		
init_pic_parameter_set_id_delta[i][j]	5	ue(v)
} else		
init_parameter_sets_info_src_op_id_delta[i]	5	ue(v)
if (bitstream_restriction_flag[i]) {		
motion_vectors_over_pic_boundaries_flag[i]	0	u(1)
max_bytes_per_pic_denom[i]	0	ue(v)
max_bits_per_mb_denom[i]	0	ue(v)
log2_max_mv_length_horizontal[i]	0	ue(v)
log2_max_mv_length_vertical[i]	0	ue(v)
num_reorder_frames[i]	0	ue(v)
max_dec_frame_buffering[i]	0	ue(v)
}		
}		
}		

[0117]

[0118]

비트스트림 제한 구문 요소의 의미는 다음과 같다:

[0119]

bitstream_restriction_flag[i]는 operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 bitstream_restriction_flag 값을 규정한다.

[0120]

motion_vectors_over_pic_boundaries_flag[i]는 operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 motion_vectors_over_pic_boundaries_flag의 값을 규정한다. motion_vectors_over_pic_boundaries_flag[i] 구문요소가 존재하지 않는 경우, operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 motion_vectors_over_pic_boundaries_flag 값은 1과 같은 것으로 추정된다.

[0121]

max_bytes_per_pic_denom[i]는 operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 max_bytes_per_pic_denom 값을 규정한다. max_bytes_per_pic_denom[i] 구문요소가 존재하지 않는 경우, operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 max_bytes_per_pic_denom의 값은 2와 같은 것으로 추정된다.

[0122]

max_bits_per_mb_denom[i]는 operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 max_bits_per_mb_denom 값을 규정한다. max_bits_per_mb_denom[i]가 존재하지 않는 경우, operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 max_bits_per_mb_denom의 값은 1과 같은 것으로 추정된다.

[0123]

log2_max_mv_length_horizontal[i] 및 log2_max_mv_length_vertical[i]는 각각 operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 log2_max_mv_length_horizontal의 값 및 log2_max_mv_length_vertical의 값을 규정한다. log2_max_mv_length_horizontal[i]가 존재하지 않는 경우, operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 log2_max_mv_length_horizontal 및 log2_max_mv_length_vertical의 값은 16과 같은 것으로 추정된다.

[0124]

num_reorder_frames[i]는 operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 num_reorder_frames의 값을 규정한다. num_reorder_frames[i]의 값은 0 내지 max_dec_frame_buffering(포함)의 범위에 있다. num_reorder_frames[i] 구문요소가 존재하지 않는 경우, operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 num_reorder_frames의 값은 max_dec_frame_buffering과 같은 것으로 추정된다.

[0125]

max_dec_frame_buffering[i]는 operation_point_id와 같은 operation_point_id[i]를 갖는 동작점의 max_dec_frame_buffering의 값을 규정한다. max_dec_frame_buffering[i]의 값은 (MPEG-4 AVC 표준의 하위조항 A.3.1 또는 A.3.2에 규정된 바와 같이) num_ref_frames[i] 내지 MaxDpbSize(포함)의 범위에 있다. max_dec_frame_buffering[i] 구문요소가 존재하지 않는 경우, operation_point_id와 같은

operation_point_id[i]를 갖는 동작점의 max_dec_frame_buffering의 값은 MaxDpbSize와 같은 것으로 추정된다.

- [0126] 도 7에서, view_scalability_parameters_extension() 구문요소를 이용하여, 각 동작점에 대한 비트스트림 제한 파라미터를 인코딩하기 위한 예시적인 방법이 참조번호 700에 의해 전반적으로 도시되어 있다.
- [0127] 상기 방법(700)은 기능블럭(710)에 제어를 넘기는 시작블럭(705)을 포함한다. 기능블럭(710)은 변수 M을 (동작점 개수 - 1)과 같도록 설정하며, 기능블럭(715)으로 넘어가도록 제어된다. 기능블럭(715)은 비트스트림에 상기 변수 M을 기입하고, 기능블럭(720)으로 넘어가도록 제어된다. 기능블럭(720)은 변수 i를 0으로 설정하며, 기능블럭(725)으로 넘어가도록 제어된다. 기능블럭(725)은 operation_point_id[i] 구문요소를 기입하고, 기능블럭(730)으로 넘어가도록 제어된다. 기능블럭(730)은 bitstream_restriction_flag[i] 구문요소를 기입하고, 결정블럭(735)으로 넘어가도록 제어된다. 결정블럭(735)은 bitstream_restriction_flag[i] 구문요소가 0과 같은지 여부를 결정한다. 만약 그렇다면, 결정블럭(745)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(740)으로 넘어가도록 제어된다.
- [0128] 기능블럭(740)은 동작점 i의 비트스트림 제한 파라미터를 기입하고, 결정블럭(745)으로 넘어가도록 제어된다. 결정블럭(745)은 상기 변수 i가 변수 M과 같은지 여부를 결정한다. 만약 그렇다면, 종료블럭(799)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(750)으로 넘어가도록 제어된다.
- [0129] 기능블럭(750)은 변수 i를 (i + 1)과 같도록 설정하며, 기능블럭(725)으로 되돌아가도록 제어된다.
- [0130] 도 8에서, view_scalability_parameters_extension() 구문요소를 이용하여, 각 동작점에 대한 비트스트림 제한 파라미터를 디코딩하기 위한 예시적인 방법이 참조번호 800에 의하여 전반적으로 도시되어 있다.
- [0131] 상기 방법(800)은 기능블럭(807)에 제어를 넘기는 시작블럭(805)을 포함한다. 기능블럭(807)은 비트스트림으로부터 변수 M을 판독하고, 기능블럭(810)으로 넘어가도록 제어된다. 기능블럭(810)은 동작점의 개수를 (변수 M + 1)과 같도록 설정하며, 기능블럭(820)으로 넘어가도록 제어된다. 기능블럭(820)은 변수 i를 0으로 설정하며, 기능블럭(825)으로 넘어가도록 제어된다. 기능블럭(825)은 operation_point_id[i] 구문요소를 판독하고, 기능블럭(830)으로 넘어가도록 제어된다. 기능블럭(830)은 bitstream_restriction_flag[i] 구문요소를 판독하고, 결정블럭(835)으로 넘어가도록 제어된다. 결정블럭(835)은 bitstream_restriction_flag[i] 구문요소가 0과 같은지 여부를 결정한다. 만약 그렇다면, 결정블럭(845)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(840)으로 넘어가도록 제어된다.
- [0132] 기능블럭(840)은 동작점 i의 비트스트림 제한 파라미터를 판독하고, 결정블럭(845)으로 넘어가도록 제어된다. 결정블럭(845)은 상기 변수 i가 변수 M과 같은지 여부를 결정한다. 만약 그렇다면, 종료블럭(899)으로 넘어가도록 제어된다. 만약 그렇지 않다면, 기능블럭(850)으로 넘어가도록 제어된다.
- [0133] 기능블럭(850)은 변수 i를 (i + 1)과 같도록 설정하며, 기능블럭(825)으로 되돌아가도록 제어된다.
- [0134] 이제 본 발명의 많은 부수적인 장점/특징 중 일부에 대하여 기재될 것이다. 예를 들어, 하나의 장점/특징은, 개별적인 시점, 시점 내 개별적인 시간 레벨, 및 개별적인 동작점 중의 적어도 하나에 대한 비디오 사용성 정보를 규정함으로써, 다시점 비디오 콘텐츠를 인코딩하기 위한 인코더를 포함하는 장치이다.
- [0135] 또 다른 장점/특징은 상기 기재된 인코더를 구비한 장치로서, 파라미터가 적어도 하나의 고급 레벨 구문(high level syntax) 요소에 규정된다.
- [0136] 또한, 또 다른 장점/특징은 상기 기재된 인코더를 구비한 장치로서, 상기 적어도 하나의 고급 레벨 구문요소는 mvc_vui_parameters_extension() 구문요소, mvc_scalability_info 보충강화정보 구문 메시지, 시퀀스 파라미터 세트의 적어도 일부분, 화상 파라미터 세트 및 보충강화정보 중의 적어도 하나를 포함한다.
- [0137] 나아가, 또 다른 장점/특징은 상기 기재된 바와 같은 인코더를 구비한 장치로서, 비디오 사용성 정보의 적어도 일부분은 비트스트림 제한 파라미터를 포함한다.
- [0138] 본 발명의 이러한 및 다른 특징과 장점은 관련 기술분야의 통상의 지식을 가진 자에 의해 상기 기재내용에 기초하여 쉽게 확인될 수 있다. 본 발명에 관한 상기 기재내용은 하드웨어, 소프트웨어, 펌웨어, 특수 목적 프로세서 또는 그들의 조합의 다양한 형태로 실행될 수 있다는 점을 이해해야 한다.
- [0139] 매우 바람직하게는, 본 발명의 상기 기재내용은 하드웨어 및 소프트웨어의 조합으로서 실행된다. 나아가, 소프

트웨어는 프로그램 저장유닛 상에 실재적으로 구체화되는 애플리케이션 프로그램으로서 실행될 수 있다. 그 애플리케이션 프로그램은 어떤 적당한 아키텍처를 포함하는 장치에 업로드되고 그 장치에 의해 실행될 수 있다. 바람직하게는, 그 장치는 하나 또는 그 이상의 중앙처리유닛("CPU"), 램덤 액세스 메모리("RAM") 및 입력/출력("I/O") 인터페이스와 같은 하드웨어를 갖는 컴퓨터 플랫폼 상에서 실행된다. 상기 컴퓨터 플랫폼은 또한 운영 시스템(operating system) 및 마이크로명령 코드를 포함할 수 있다. 여기에 기재된 다양한 프로세스 및 기능은 마이크로명령 코드의 일부, 응용 프로그램의 일부 또는 그들의 조합일 수 있으며, 그것은 CPU에 의해 실행될 수 있다. 또한, 부가적 데이터 저장유닛 및 프린팅 유닛과 같은 다양한 다른 주변 장치가 컴퓨터 플랫폼에 연결될 수 있다.

[0140] 첨부된 도면에 도시된 구성적 시스템 요소 및 방법의 몇몇은 바람직하게는 소프트웨어에서 실행되기 때문에, 시스템 요소 간 또는 프로세스 기능블럭 간의 실제 연결은 본 발명이 프로그램되는 방식에 따라 달라질 수 있다. 여기에 기재된 내용이 일단 제공되면, 관련분야의 통상의 지식을 가진 자라면 본 발명의 상기 및 이와 유사한 실행에 또는 구성을 생각해 낼 수 있을 것이다.

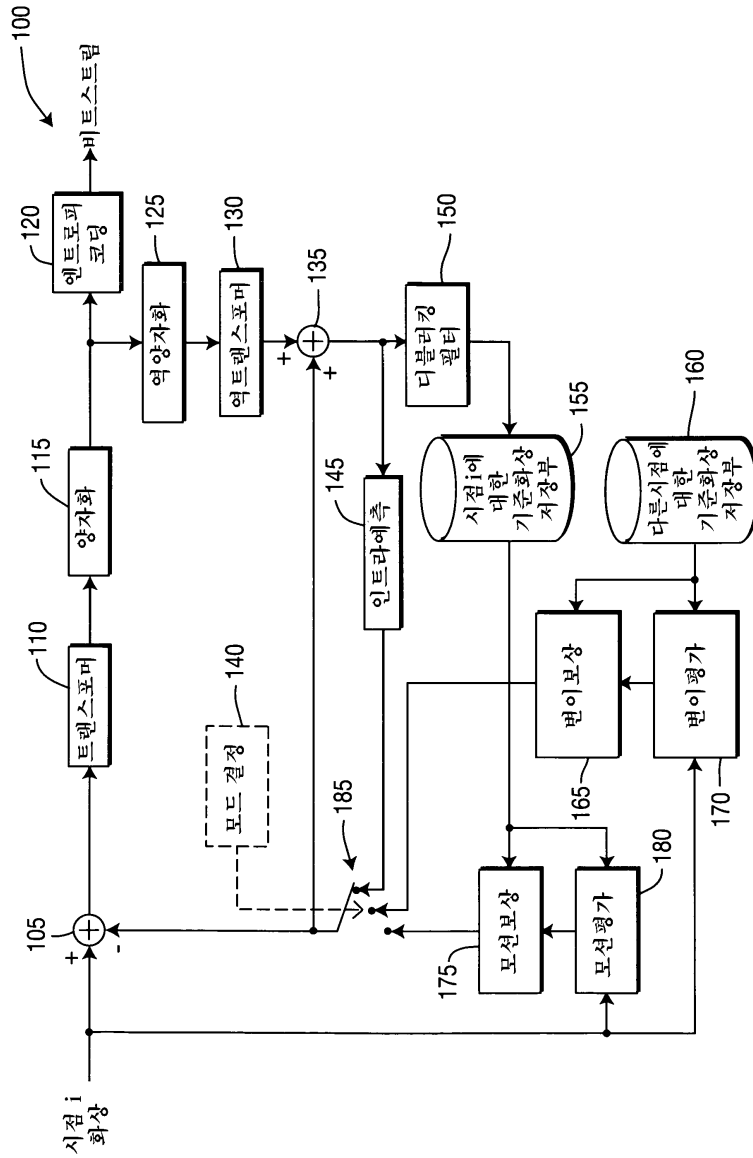
[0141] 첨부된 도면을 참조하여 예시적인 실시예가 여기에 기재되었지만, 본 발명은 그러한 구체적인 실시예에 한정되지 않으며, 본 발명의 범위 또는 사상을 벗어나지 않는 한 관련 기술분야의 통상의 지식을 가진 자에 의하여 다양한 변경 및 수정이 거기에 행해질 수 있다는 점을 이해해야 한다. 그러한 모든 변경 및 수정은 첨부된 청구범위에 기재된 바와 같은 본 발명의 범위에 포함되도록 의도된다.

부호의 설명

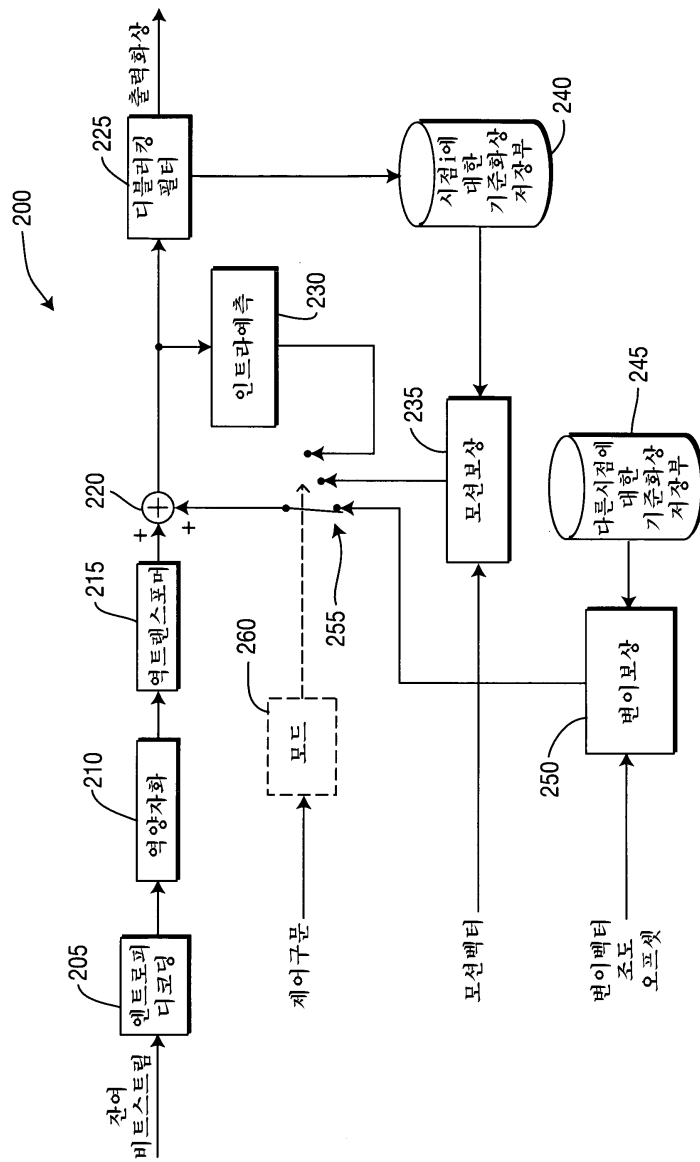
[0142]	100 : 인코더	110 : 트랜스포머
	105 : 결합기	115 : 양자화기
	120 : 엔트로피 코더	125 : 역양자화기
	130 : 역트랜스포머	135 : 결합기
	140 : 모드 결정 모듈	
	145 : 인트라예측기	150 : 디블러킹 필터
	155 : 기준화상 저장부	160 : 기준화상 저장부
	165 : 변이/조도 보상기	170 : 변이/조도 평가기
	175 : 모션보상기	180 : 모션 평가기
	185 : 스위치	
	200 : 디코더	205 : 엔트로피 디코더
	210 : 역양자화기	215 : 역트랜스포머
	220 : 결합기	225 : 디블러킹 필터
	230 : 인트라예측기	240 : 기준화상 저장부
	235 : 모션보상기	245 : 기준화상 저장부
	250 : 변이/조도 보상기	255 : 스위치
	260 : 모드모듈	

도면

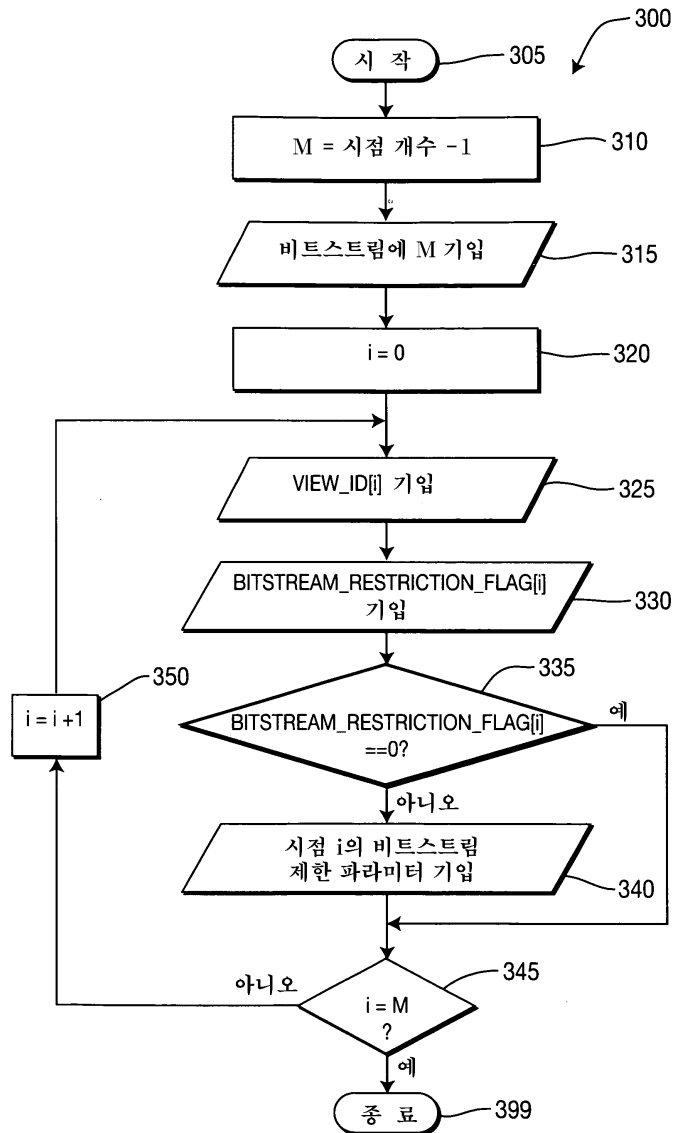
도면1



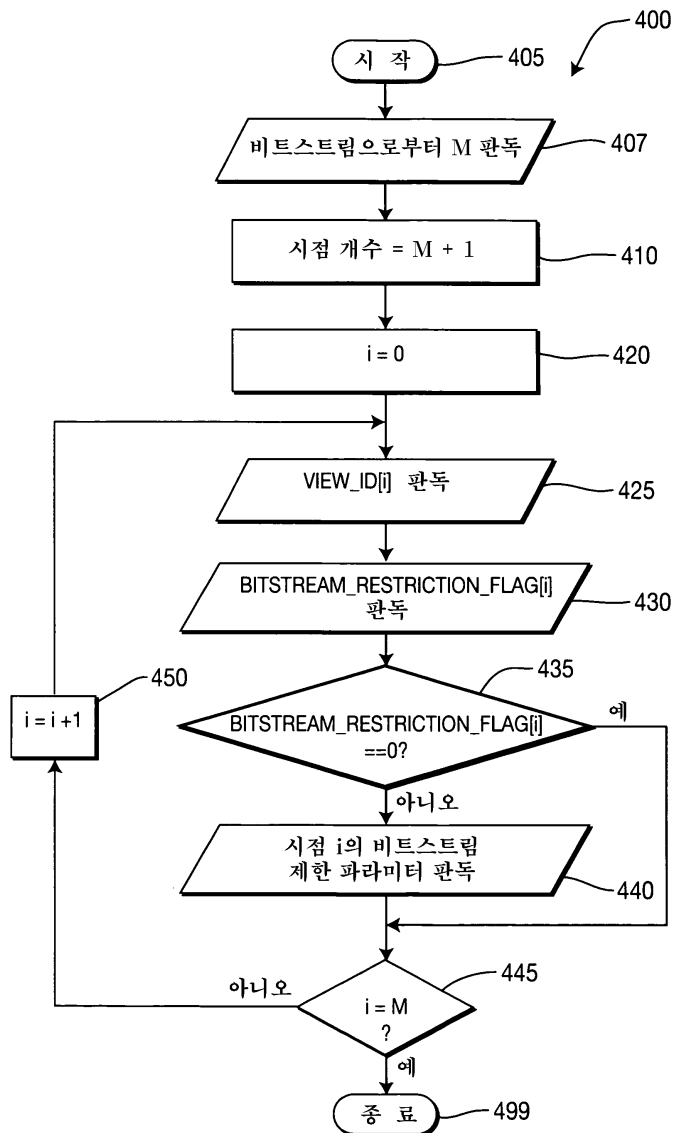
도면2



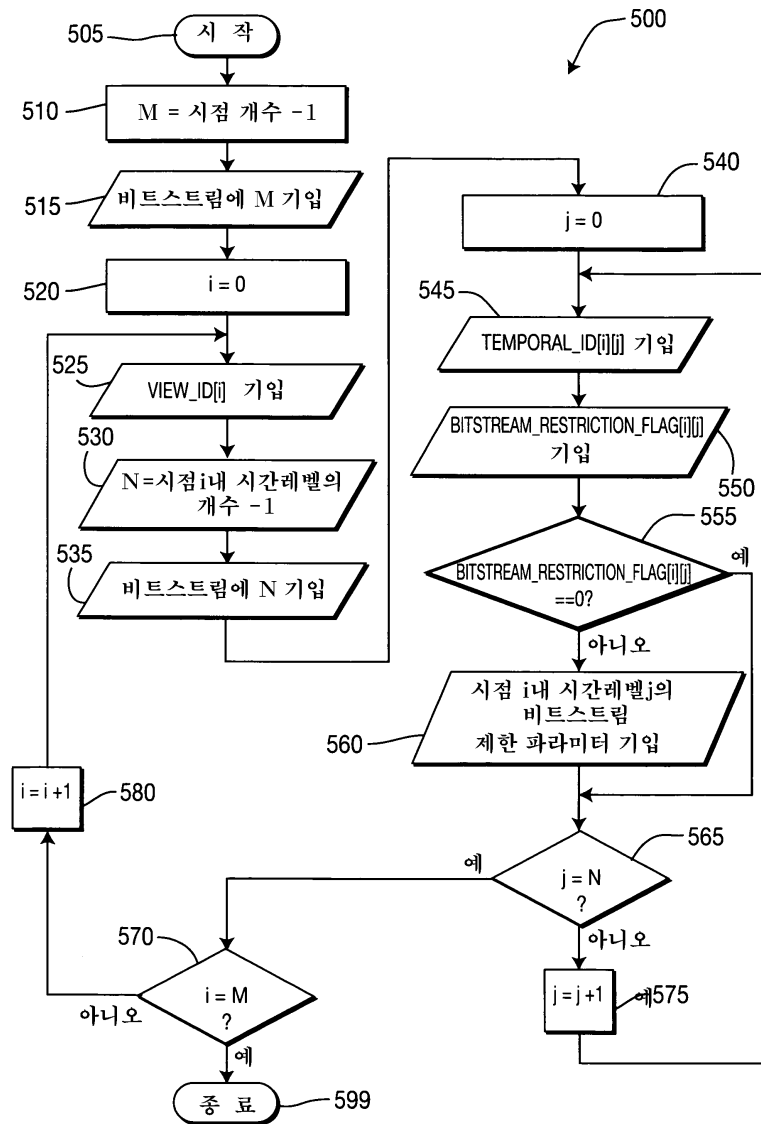
도면3



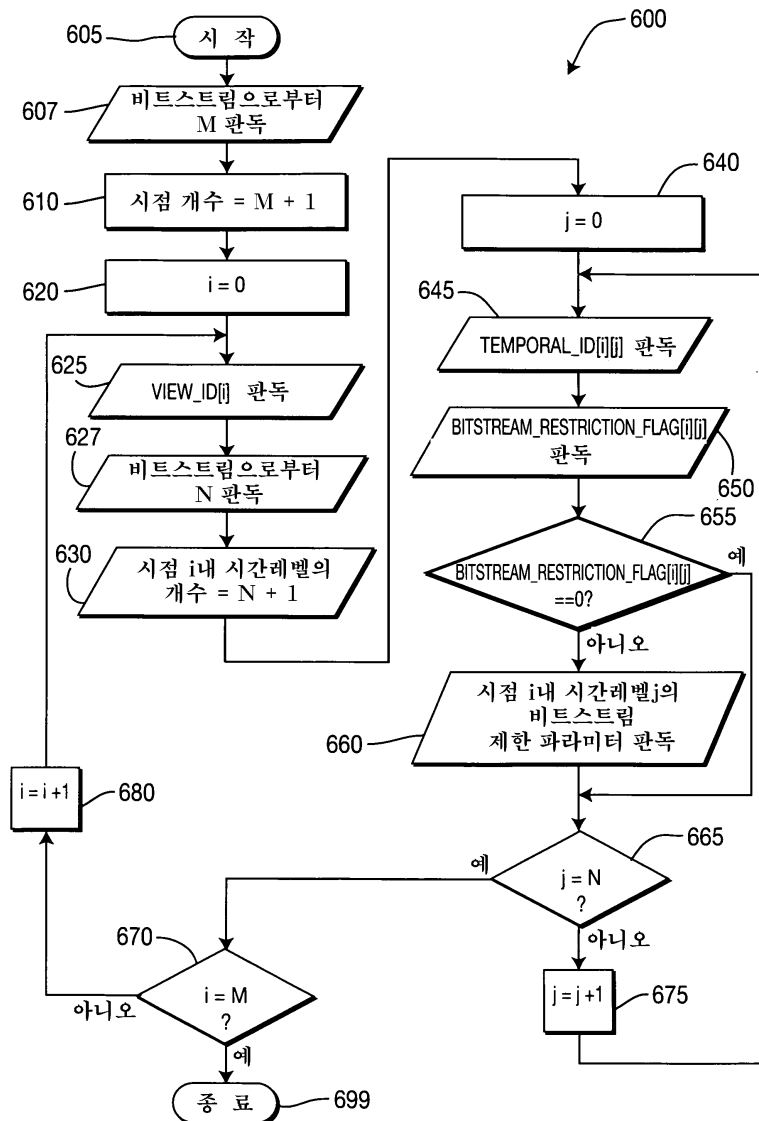
도면4



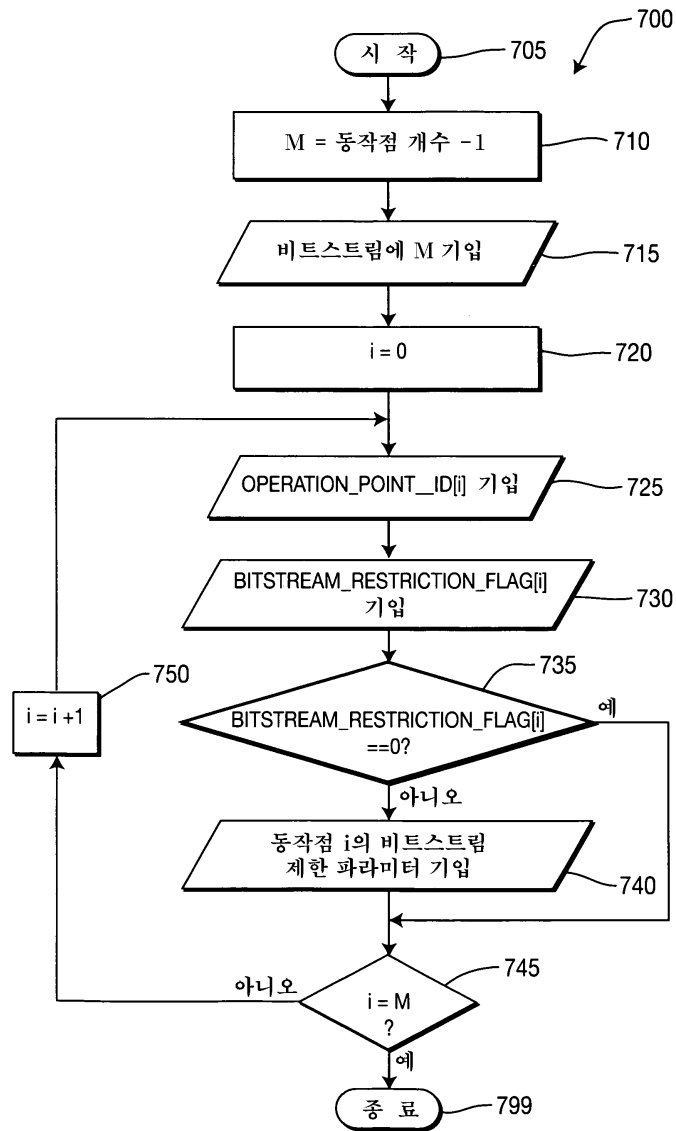
도면5



도면6



도면7



도면8

