



## 특허청구의 범위

### 청구항 1

각각이 복수의 블록을 가지는 복수의 프레임을 포함하는 인코딩된 비디오 정보의 스트림을 디코딩하는 방법으로서,

적어도 제1 프레임에 대해, 제1 프레임 내의 적어도 하나의 블록을 적어도 하나의 세그먼트 식별자와 연관시키는 세그먼트 맵을 비디오 정보로부터 판독하는 단계 - 세그먼트 식별자는 이진 산술 디코더를 사용하여 디코딩됨 -,

세그먼트 식별자와 연관된 적어도 하나의 세그먼트 파라미터를 인코딩된 비디오 정보로부터 판독하는 단계,

세그먼트 파라미터를 사용하여 블록을 디코딩하는 단계, 및

적어도 제2 프레임에 대해, 세그먼트 식별자를 사용하여 제1 프레임 내의 블록에 대응하는 제2 프레임 내의 블록을 디코딩하는 단계를 포함하는 방법.

### 청구항 2

제1항에 있어서, 제2 프레임 내의 블록을 디코딩하는 단계가

제2 프레임과 연관된 레코드를 인코딩된 비디오 정보로부터 판독하는 단계 - 레코드는 제2 프레임에 대한 세그먼트화 조건을 나타냄 -, 및

제2 프레임에 대해 세그먼트화 조건이 존재할 때, 세그먼트 식별자를 사용하여 제1 프레임 내의 블록에 대응하는 제2 프레임 내의 블록을 디코딩하는 단계를 추가로 포함하는 것인 방법.

### 청구항 3

제1항 또는 제2항에 있어서, 세그먼트 파라미터가 양자화 파라미터, 루프 필터 유형, 루프 필터 강도값, 및 서브-픽셀 보간 필터 중 적어도 하나를 포함하는 것인 방법.

### 청구항 4

제1항 또는 제2항에 있어서, 복수의 세그먼트 파라미터가 세그먼트 식별자와 연관되어 있고, 복수의 세그먼트 파라미터 중 제1 세그먼트 파라미터가 대안의 참조 프레임을 나타내는 값을 포함하며, 상기 방법이

세그먼트 식별자와 연관된 복수의 세그먼트 파라미터 중 적어도 제2 세그먼트 파라미터를 사용하여, 지정된 대안의 참조 프레임을 업데이트하는 단계를 추가로 포함하는 것인 방법.

### 청구항 5

제1항 또는 제2항에 있어서, 디코딩되고 있는 프레임이 세그먼트들 중 특정의 세그먼트에 대한 업데이트된 세그먼트 파라미터를 지정하는지 여부를 판정하는 단계, 및

그 지정된 업데이트된 세그먼트 파라미터를 세그먼트들 중 특정의 세그먼트와 연관시키는 단계를 추가로 포함하는 방법.

### 청구항 6

제5항에 있어서, 업데이트된 세그먼트 파라미터가 세그먼트 파라미터에 가산될 증분값인 방법.

### 청구항 7

제1항 또는 제2항에 있어서, 세그먼트 맵을 판독하는 단계가

블록에 고유한 데이터와 인터리빙되는 비디오 정보의 일부분에서 세그먼트 식별자를 판독하는 단계를 추가로 포함하는 것인 방법.

**청구항 8**

제1항 또는 제2항에 있어서, 세그먼트 맵을 판독하는 단계가 적어도 하나의 프레임과 연관된 헤더 파일에서 세그먼트 식별자를 판독하는 단계를 추가로 포함하는 것인 방법.

**청구항 9**

제1항 또는 제2항에 있어서, 프레임이 키 프레임인지 여부를 판정하는 단계, 및 프레임이 키 프레임일 때, 세그먼트 파라미터를 기본값으로 설정하는 단계를 추가로 포함하는 방법.

**청구항 10**

각각이 복수의 블록을 가지는 복수의 프레임을 포함하는 압축된 비디오 정보를 디코딩하는 방법으로서, 적어도 하나의 프레임 내의 적어도 하나의 블록을 적어도 하나의 세그먼트 식별자와 연관시키는 세그먼트 맵을 비디오 정보로부터 판독하는 단계 - 세그먼트 식별자는 이진 산술 디코더를 사용하여 디코딩됨 -, 세그먼트 식별자와 연관된 적어도 하나의 세그먼트 파라미터를 비디오 정보로부터 판독하는 단계, 및 세그먼트 파라미터를 사용하여 블록을 디코딩하는 단계를 포함하는 방법.

**청구항 11**

제10항에 있어서, 세그먼트 파라미터가 양자화 파라미터, 루프 필터 유형, 루프 필터 강도값, 및 서브-픽셀 보간 필터 중 적어도 하나를 포함하는 것인 방법.

**청구항 12**

제10항 또는 제11항에 있어서, 복수의 세그먼트 파라미터가 세그먼트 식별자와 연관되어 있고, 복수의 세그먼트 파라미터 중 제1 세그먼트 파라미터가 대안의 참조 프레임을 나타내는 값을 포함하며, 상기 방법이 세그먼트 식별자와 연관된 복수의 세그먼트 파라미터 중 적어도 제2 세그먼트 파라미터를 사용하여, 지정된 대안의 참조 프레임을 업데이트하는 단계를 추가로 포함하는 것인 방법.

**청구항 13**

제10항 또는 제11항에 있어서, 디코딩되고 있는 프레임이 세그먼트들 중 특정의 세그먼트에 대한 업데이트된 세그먼트 파라미터를 지정하는지 여부를 판정하는 단계, 및 그 지정된 업데이트된 세그먼트 파라미터를 세그먼트들 중 특정의 세그먼트와 연관시키는 단계를 추가로 포함하는 방법.

**청구항 14**

제13항에 있어서, 업데이트된 세그먼트 파라미터가 세그먼트 파라미터에 가산될 충분한 증분값인 방법.

**청구항 15**

제10항 또는 제11항에 있어서, 특정 프레임과 연관된 비디오 정보로부터 세그먼트화 조건을 나타내는 레코드를 판독하는 단계, 및 세그먼트화 조건이 거짓일 때, 세그먼트 파라미터를 기본 파라미터로 설정하고 세그먼트 맵의 판독 및 세그먼트 파라미터의 판독을 억제하는 단계를 추가로 포함하는 방법.

**청구항 16**

제10항 또는 제11항에 있어서, 세그먼트 맵을 판독하는 단계가 블록에 고유한 데이터와 인터리빙되는 비디오 정보의 일부분에서 세그먼트 식별자를 판독하는 단계를 추가로 포함하는 것인 방법.

**청구항 17**

제10항 또는 제11항에 있어서, 세그먼트 맵을 판독하는 단계가 프레임과 연관된 헤더 파일에서 세그먼트 식별자를 판독하는 단계를 추가로 포함하는 것인 방법.

**청구항 18**

제10항 또는 제11항에 있어서, 프레임이 키 프레임인지 여부를 판정하는 단계, 및 프레임이 키 프레임일 때, 세그먼트 파라미터를 기본값으로 설정하는 단계를 추가로 포함하는 방법.

**청구항 19**

각각이 복수의 블록을 가지는 복수의 프레임을 포함하는 인코딩된 비디오 정보의 스트림을 생성하는 방법으로서,

적어도 제1 프레임에 대해, 제1 프레임 내의 적어도 하나의 블록을 적어도 하나의 세그먼트 식별자와 연관시키는 세그먼트 맵을 생성하고 적어도 하나의 세그먼트 식별자를 적어도 하나의 세그먼트 파라미터와 연관시키는 단계,

이진 산술 인코더를 사용하여 세그먼트 식별자를 인코딩하는 단계, 및 세그먼트 파라미터를 사용하여 블록을 디코딩하는 단계를 포함하는 방법.

**청구항 20**

제19항에 있어서, 적어도 제2 프레임에 대해, 세그먼트화 수정 조건을 감지하는 단계, 및 세그먼트화 수정 조건이 감지될 때, 세그먼트 파라미터를 업데이트하고 업데이트된 세그먼트 파라미터를 제2 프레임과 연관된 비디오 정보와 함께 저장하는 단계를 추가로 포함하는 방법.

**청구항 21**

제20항에 있어서, 업데이트된 세그먼트 파라미터가 충분값인 방법.

**청구항 22**

제19항 또는 제20항에 있어서, 적어도 제2 프레임에 대해, 세그먼트화 조건을 감지하는 단계, 및 세그먼트화 조건이 감지될 때, 세그먼트화 플래그를 제2 프레임과 연관된 인코딩된 비디오 정보와 함께 저장하는 단계를 추가로 포함하는 방법.

**청구항 23**

제19항 또는 제20항에 있어서, 세그먼트 파라미터가 양자화 파라미터, 루프 필터 유형, 루프 필터 강도, 및 서브-픽셀 보간 필터 중 적어도 하나를 포함하는 것인 방법.

**청구항 24**

제23항에 있어서, 세그먼트 파라미터가 대안의 참조 프레임을 나타내는 값을 포함하며, 상기 방법이 세그먼트 파라미터를 사용하여 지정된 대안의 참조 프레임을 업데이트하는 단계를 추가로 포함하는 것인 방법.

**청구항 25**

제19항 또는 제20항에 있어서, 블록에 고유한 데이터와 인터리빙되는 비디오 정보의 일부분에 세그먼트 식별자를 저장하는 단계를 추가로 포함하는 방법.

**청구항 26**

제19항 또는 제20항에 있어서, 제1 프레임의 헤더 부분에 세그먼트 식별자를 저장하는 단계를 추가로 포함하는 방법.

## 명세서

### 기술분야

- [0001] 관련 출원들의 상호 참조
- [0002] 이 출원은 2008년 12월 5일자로 출원된 미국 특허 출원 제12/329,177호(2008년 9월 11일자로 출원된 미국 특허 출원 제61/096,242호를 기초로 우선권을 주장함)를 기초로 우선권을 주장하며, 양 출원은 참조 문헌으로서 그 전체 내용이 본 명세서에 포함된다.
- [0003] 본 발명은 일반적으로 비디오 인코딩에 관한 것이다.

### 배경기술

- [0004] 현재 점점 많은 수의 응용 프로그램이, 예를 들어, 화상 회의를 통한 원격 업무 회의, 고선명 비디오 엔터테인먼트, 비디오 광고, 및 사용자-생성 비디오의 공유를 비롯한 다양한 목적으로 디지털 비디오를 사용하고 있다. 기술이 발전함에 따라, 사람들은 비디오 품질에 대한 더 많은 기대를 가지고 있고 높은 프레임 레이트에서 원활하게 재생되는 고휘상도 비디오를 기대하고 있다.
- [0005] 디지털 비디오를 보기 위한 비디오 코더를 선택할 때 고려할 많은 요인들이 있을 수 있다. 일부 응용 프로그램은 우수한 비디오 품질을 필요로 할 수 있고, 다른 것들은, 예를 들어, 대역폭 또는 저장 장치 요구사항을 비롯한 다양한 제약조건에 부합해야만 할지도 모른다. 대역폭 사용을 제한하면서 고품질의 비디오 전송을 허용하기 위해, VPx(미국 뉴욕주 클리프턴 파크 소재의 On2 Technologies, Inc.에 의해 발표됨), ITU-T VCEG(Video Coding Experts Group)에 의해 발표된 H.264 표준, 및 ISO/IEC MPEG(Moving Picture Experts Group)(이들의 현재 및 미래의 버전을 포함함) 등의 독점 포맷을 비롯한 다수의 비디오 압축 방식이 주목을 받고 있다. H.264는 또한 MPEG-4 Part 10 또는 MPEG-4 AVC(정식 명칭은 ISO/IEC 14496-10임)라고도 한다.
- [0006] 최근의 비디오 코딩 방법은 전체 프레임보다 작은 세분성 레벨(level of granularity)에서 특정 압축 파라미터를 조정한다. 예를 들어, 일부 압축 방식은 관찰자에게 보여지는 비디오 품질을 향상시키기 위해 매크로블록 또는 블록 경계에 적용되는 루프 필터의 유형 또는 강도를 조정한다. 게다가, 예를 들어, 프레임 내의 각각의 매크로블록에 상이한 양자화 레벨이 적용될 수 있다. 이러한 방식으로, 중요한 정보를 가지는 매크로블록이 높은 비트 레이트로 전송될 수 있는 반면, 덜 중요한 정보를 가지는 매크로블록은 낮은 비트 레이트로 전송될 수 있다.

### 발명의 내용

#### 해결하려는 과제

- [0007] 기술이 발전함에 따라, 사람들은 비디오 품질에 대한 더 많은 기대를 가지고 있고 높은 프레임 레이트에서 원활하게 재생되는 고휘상도 비디오를 기대하고 있다.

#### 과제의 해결 수단

- [0008] 비디오 정보를 인코딩 및 디코딩하는 방법의 실시예에 대해 본 명세서에 개시되어 있다. 일 실시예에서, 방법은 각각이 복수의 블록을 가지는 복수의 프레임을 포함하는 압축된 비디오 정보를 디코딩한다. 이 방법은 적어도 하나의 프레임 내의 적어도 하나의 블록을 적어도 하나의 세그먼트 식별자와 연관시키는 세그먼트 맵을 비디오 정보로부터 판독하는 단계를 포함한다. 세그먼트 식별자는 이진 산술 디코더를 사용하여 디코딩된다. 이 방법은 또한 세그먼트와 연관된 적어도 하나의 세그먼트 파라미터를 정보로부터 판독하는 단계, 및 일련의 세그먼트 파라미터를 사용하여 블록을 디코딩하는 단계를 포함한다. 일부 실시예에서, 세그먼트 파라미터는 양자화 파라미터, 루프 필터 유형, 루프 필터 강도, 및 서브-픽셀 보간 필터 중 적어도 하나를 포함한다.
- [0009] 다른 실시예에서, 각각이 복수의 블록을 가지는 복수의 프레임을 포함하는 인코딩된 비디오 정보의 스트림을 디코딩하는 방법이 개시되어 있다. 이 방법은, 적어도 제1 프레임에 대해, 제1 프레임 내의 적어도 하나의 블록을 적어도 하나의 세그먼트 식별자와 연관시키는 세그먼트 맵을 비디오 정보로부터 판독하는 단계를 포함한다. 세그먼트 식별자는 이진 산술 디코더를 사용하여 디코딩된다. 적어도 하나의 세그먼트 파라미터는 세그먼트 식별자와 연관되어 있는 인코딩된 비디오 정보로부터 판독된다. 블록은 세그먼트 파라미터를 사용하여 디코딩된다. 이 방법은, 적어도 제2 프레임에 대해, 세그먼트 식별자를 사용하여 제1 프레임 내의 블록에 대응하는 제2 프레

임 내의 블록을 디코딩하는 단계를 추가로 포함한다.

[0010] 다른 실시예에서, 각각이 복수의 블록을 가지는 복수의 프레임을 포함하는 인코딩된 비디오 정보의 스트림을 생성하는 방법이 개시되어 있다. 이 방법은, 적어도 제1 프레임에 대해, 제1 프레임 내의 적어도 하나의 블록을 적어도 하나의 세그먼트 식별자와 연관시키는 세그먼트 맵을 생성하는 단계, 및 세그먼트 식별자를 적어도 하나의 세그먼트 파라미터와 연관시키는 단계를 포함한다. 세그먼트 식별자는 이진 산술 인코더를 사용하여 인코딩된다. 블록은 세그먼트 파라미터를 사용하여 인코딩된다.

[0011] 이후부터 본 발명의 이들 및 기타 실시예에 대해 더 상세히 기술한다.

**발명의 효과**

[0012] 비디오 정보를 인코딩 및 디코딩하는 방법이 제공된다.

**도면의 간단한 설명**

[0013] 본 명세서에서의 설명은 유사한 참조 번호가 몇개의 도면에 걸쳐 유사한 부분을 지칭하고 있는 첨부 도면을 참조한다.

도 1은 본 발명의 일 실시예에 따른 비디오 압축 시스템의 블록도.

도 2는 본 발명의 일 실시예에 따른 비디오 압축 해제 시스템의 블록도.

도 3은 도 1 및 도 2의 비디오 압축 및 압축 해제 시스템에서 사용되는 인트라-예측 모드 및 인터-예측 모드의 개략도.

도 4a는 도 1 및 도 2의 비디오 압축 및 압축 해제 시스템에서 사용되는 예시적인 세그먼트화 맵.

도 4b는 도 1 및 도 2의 비디오 압축 및 압축 해제 시스템에서 사용되는 대안의 예시적인 세그먼트화 맵.

도 5는 도 1의 비디오 압축 시스템에서 사용되는 비디오의 세그먼트화를 제어하는 방법의 플로우차트도.

도 6은 도 1 및 도 2의 비디오 압축 및 압축 해제 시스템에서 사용되는 세그먼트 식별자를 코딩하는 데 사용되는 이진 트리의 개략도.

도 7은 도 1의 비디오 압축 시스템에서 사용되는 세그먼트 식별자를 인코딩하는 방법의 플로우차트도.

**발명을 실시하기 위한 구체적인 내용**

[0014] 비디오 압축은 보통 비디오 신호 내의 특정 중복 정보를 영구적으로 제거함으로써 비디오 신호를 나타내는 데이터의 양을 감소시키는 것을 말한다. 압축이, 예를 들어, 대역폭 또는 저장소 요구사항을 비롯한 다양한 제약조건에 부합해야만 할지도 모른다. 이러한 압축 표준의 일례는 MPEG 및 H.264을 포함한다.

[0015] 블록-기반 비디오 압축은, 예를 들어, 고정된 형상의 이웃 픽셀 그룹(종종 매크로블록이라고 함)에 대해 동작한다. 예를 들어, H.264 표준은 16x16 픽셀의 매크로블록을 지정한다. 일반적으로, 각각의 비디오 프레임은 매크로블록으로 분할될 수 있으며, 각각의 매크로블록은 복수의 보다 작은 크기의 블록으로 이루어져 있다. 매크로블록 및 블록 내의 이들 픽셀 그룹은, 예측 데이터 및 오차 신호를 작성하기 위해, 현재 프레임에서 발견되는 데이터 또는 다른 프레임에서 발견되는 데이터와 비교된다.

[0016] 각각의 블록에 대한 오차 신호는 이어서, 예를 들어, DCT(discrete cosine transform)를 사용하여 변환된다. 각각의 블록에 대해 얻어지는 변환 계수는 이어서 양자화되고, 이어서 양자화된 계수는 이어서 CABAC(context-adaptive binary arithmetic coding) 등의 인코딩 방식에 따라 인코딩된다.

[0017] 변환 계수가 양자화되는 정도(양자화 레벨이라고 함)는 이미지 데이터를 표현하는 데 사용되는 비트의 수는 물론 얻어지는 디코딩된 이미지의 품질에 영향을 준다. 일반적으로, 양자화 레벨이 높은 값으로 설정될 때, 더 많은 계수가 0으로 설정되며, 그 결과 더 높은 압축이 얻어지지만, 또한 이미지의 품질을 떨어뜨린다. 이와 달리, 양자화 레벨이 낮은 값으로 설정될 때, 더 적은 계수가 0으로 설정되며, 그 결과 향상된 이미지 품질이 얻어지지만, 또한 덜 압축된다.

[0018] 비디오의 시각적 품질이 양자화 레벨이 비디오 시퀀스 내의 프레임들에 걸쳐 어떻게 분산되는지에 의존할 수 있기 때문에, 각각의 프레임에 걸쳐 양자화 레벨을 제어하는 것이 유용하다. 일부 현재 기술은 각각의 프레임에

걸쳐 균일한 양자화 레벨을 적용한다. 그렇지만, 이들 기법은 낮은 양자화 레벨을 시각적으로 더 중요한 프레임의 일부(사람의 얼굴 등)에 적용하는 것을 고려하고 있지 않다. 이와 유사하게, 이들 균일한 양자화 레벨 기법은 또한 높은 양자화 레벨을 시각적으로 덜 중요한 프레임의 일부(배경 영역 등)에 적용하는 것을 고려하고 있지 않다.

- [0019] 이들 어려움을 극복하기 위해, 인코더/디코더에서의 오버헤드 및 계산 복잡도를 증가시키는 일 없이, 각각의 매크로블록에 대한 양자화 레벨을 효율적으로 선택하는 것이 바람직할 것이다.
- [0020] 블록 아티팩트는 인접한 블록들 간의 불연속으로서 나타난다. 이러한 불연속은 시각적으로 거스리며, 또한 후속 프레임에 대한 예측자로서의 재구성된 프레임의 유효성을 감소시킨다.
- [0021] 이들 불연속을 제거하기 위해, 재구성 경로 동안에 루프 필터링이 재구성된 프레임에 적용될 수 있으며, 이에 대해서는 이하에서 더 상세히 기술한다. 루프 필터의 선택 및 루프 필터의 강도는 이미지 품질에 상당한 영향을 미칠 수 있다. 너무 강한 필터는 블러링(blurring) 및 디테일의 손실(loss of detail)을 야기할 수 있다. 너무 약한 필터는 인접 블록들 간의 불연속을 적절히 억압하지 못할지도 모른다.
- [0022] 그에 따라, 세그먼트화 맵 및 세그먼트화 맵 내의 각각의 세그먼트에 적용될 일련의 파라미터(양자화 레벨 및 루프 필터 유형 및/또는 강도 등)를 효율적으로 전송하는 효율적인 적응적 세그먼트화 방식의 실시예가 본 명세서에 기술되어 있다.
- [0023] 적응적 세그먼트화 발명의 실시예들에 대한 설명이 VP8 비디오 코딩 포맷과 관련하여 기술되어 있지만, 본 발명의 대안의 실시예가 다른 비디오 코딩 알고리즘과 관련하여 구현될 수 있다.
- [0024] 도 1은 본 발명의 일 실시예에 따른 루프 필터(34)를 사용하는 일반화된 비디오 인코더(14)의 블록도이다. 도 1을 참조하면, 입력 비디오 스트림(16)을 인코딩하기 위해, 인코더(14)는 전방향 경로(실선 연결선으로 나타냄)에서 인코딩된 비트스트림(26)을 생성하기 위해 이하의 기능, 즉 인트라/인터 예측(18), 변환(19), 양자화(22) 및 엔트로피 인코딩(24)을 수행한다. 인코더(14)는 또한 추가의 매크로블록의 인코딩을 위한 프레임을 재구성하는 재구성 경로(점선 연결선으로 나타냄)도 포함하고 있다. 인코더(14)는 재구성 경로에서 이하의 기능, 즉 역양자화(28), 역변환(30), 재구성(32) 및 루프 필터링(34)을 수행한다. 인코더(14)의 다른 구조적 변형이 비트스트림(26)을 인코딩하는 데 사용될 수 있다.
- [0025] 도 1을 참조하면, 입력 비디오 스트림(16)이 인코딩을 위해 제공될 때, 입력 비디오 스트림(16) 내의 각각의 프레임은 매크로블록 단위로 처리될 수 있다. 인트라/인터 예측 단계(18)에서, 각각의 매크로블록은 인트라-예측 모드 또는 인터-예측 모드 중 어느 하나를 사용하여 인코딩될 수 있다. 어느 경우든지, 재구성된 프레임에 기초하여 예측 매크로블록이 형성될 수 있다. 인트라-예측의 경우에, 이전에 인코딩되고 재구성된 현재 프레임 내의 샘플들로부터 예측 매크로블록이 형성될 수 있다. 인터-예측의 경우에, 이미 인코딩되고 재구성된 하나 이상의 참조 프레임(이전의 또는 장래의 프레임 등)으로부터 예측 매크로블록이 형성될 수 있다.
- [0026] 그 다음에, 여전히 도 1을 참조하면, 예측 매크로블록이 현재 매크로블록으로부터 차감되어 잔차 매크로블록(잔차)을 생성할 수 있다. 변환 단계(19)는 잔차를 변환 코딩하고 양자화 단계(22)는 잔차를 양자화하여 일련의 양자화된 변환된 계수를 생성한다. 양자화된 변환된 계수는 이어서 엔트로피 인코딩 단계(24)에 의해 엔트로피 코딩된다. 엔트로피 코딩된 계수는, 매크로블록을 디코딩하는 데 필요한 정보(사용되는 예측 모드의 유형, 움직임 벡터 및 양자화기 값 등)와 함께, 압축된 비트스트림(26)으로 출력된다.
- [0027] 도 1의 재구성 경로는 인코더 및 디코더 둘다가 매크로블록을 디코딩하는 데 필요한 동일한 참조 프레임을 사용하는 것을 허용하기 위해 존재한다. 이하에서 더 상세히 기술되는 디코딩 프로세스 동안에 일어나는 기능들과 유사한 재구성 경로는 역양자화 단계(28)에 의해 변환된 계수를 역양자화하는 것 및 역변환 단계(30)에 의해 계수를 역변환하여 파생 잔차 매크로블록(derivative residual macroblock)(파생 잔차)을 생성하는 것을 포함한다. 재구성 단계(32)에서, 예측 매크로블록이 파생 잔차에 부가되어 재구성된 매크로블록을 생성할 수 있다. 블록화 아티팩트를 감소시키기 위해, 루프 필터(34)가 재구성된 매크로블록에 적용될 수 있다.
- [0028] 도 2를 참조하면, 일 실시예에 따르면, 압축된 비트스트림(26)을 디코딩하기 위해, 이전에 기술한 인코더(14)의 재구성 경로와 유사한 디코더(21)는 출력 비디오 스트림(35)을 생성하기 위해 이하의 기능, 즉 엔트로피 디코딩(25), 역양자화(27), 역변환(29), 인트라/인터 예측(23), 재구성(31), 루프 필터(34) 및 디블록킹 필터링(33)을 수행한다. 디코더(21)의 다른 구조적 변형이 압축된 비트스트림(26)을 디코딩하는 데 사용될 수 있다.
- [0029] 압축된 비트스트림(26)이 디코딩을 위해 제공될 때, 데이터 요소는 엔트로피 디코딩 단계(25)에 의해 엔트로피

디코딩되어 일련의 양자화된 계수를 생성할 수 있다. 역양자화 단계(27)는 계수를 역양자화하고 역변환 단계(29)는 계수를 역변환하여, 인코더(14)에서의 재구성 단계에서 생성된 것과 동일한 과생 잔차를 생성한다. 압축된 비트스트림(26)으로부터 디코딩된 헤더 정보를 사용하여, 인트라/인터 예측 단계(23)에서, 디코더(21)는 인코더(14)에서 생성된 것과 동일한 예측 매크로블록을 생성한다. 재구성 단계(33)에서, 예측 매크로블록이 과생 잔차에 부가되어 재구성된 매크로블록을 생성할 수 있다. 블록화 아티팩트를 감소시키기 위해, 루프 필터(34)가 재구성된 매크로블록에 적용될 수 있다. 디블록킹 필터(33)는 재구성된 매크로블록에 적용되어, 블록화 왜곡을 추가적으로 감소시킬 수 있고, 그 결과가 출력 비디오 스트림(35)으로 출력될 수 있다.

[0030] 도 3은 참조 프레임(44, 48) 및 현재 인코딩 또는 디코딩되고 있는 현재 프레임(36)을 나타내고 있다. 앞서 기술한 바와 같이, 각각의 프레임은 매크로블록 단위로 처리될 수 있으며, 인트라/인터 예측 단계(18)에서, 각각의 매크로블록은 인트라 예측 또는 인터 예측 모드를 사용하여 인코딩될 수 있다. 예를 들어, 현재 매크로블록(38)은 이전에 코딩된 참조 프레임(44)으로부터의 매크로블록(46)으로부터 인터 예측을 사용하여 인코딩되거나 디코딩될 수 있다. 이와 유사하게, 현재 매크로블록(38')은 이전에 인코딩된 참조 프레임(48)으로부터의 매크로블록(50)으로부터 인터 예측을 사용하여 인코딩되거나 디코딩될 수 있다. 또한, 예를 들어, 현재 매크로블록(38'')은 현재 프레임(36) 내의 매크로블록(52)으로부터 인트라 예측을 사용하여 인코딩되거나 디코딩될 수 있다.

[0031] 도 4a 및 도 4b는 본 발명의 일 실시예에 따른 샘플 프레임에 대한 예시적인 세그먼트화 맵을 나타낸 것이다.

[0032] 도 4a를 참조하면, 예시적인 세그먼트화 맵(100)은 프레임(110)을 4개의 세그먼트[3개의 전경 세그먼트(112, 114, 116) 및 하나의 배경 세그먼트(118)]로 세그먼트화한다. 본 발명의 일 실시예에 따르면, 전경 세그먼트(112)에 대응하는 모든 매크로블록(백색 매크로블록으로 나타냄)은 동일한 양자화 레벨 및 루프 필터 유형 및/또는 강도를 사용하여 인코딩되고, 전경 세그먼트(114)에 대응하는 모든 매크로블록(흑색 매크로블록으로 나타냄)은 동일한 양자화 레벨 및 루프 필터 유형 및/또는 강도를 사용하여 인코딩되며, 전경 세그먼트(116)에 대응하는 모든 매크로블록(점을 찍은 매크로블록으로 나타냄)은 동일한 양자화 레벨 및 루프 필터 유형 및/또는 강도를 사용하여 인코딩되고, 전경 세그먼트(118)에 대응하는 모든 매크로블록(선이 그어진 매크로블록으로 나타냄)은 동일한 양자화 레벨 및 루프 필터 유형 및/또는 강도를 사용하여 인코딩된다. 따라서, 각각의 세그먼트는 상이한 양자화 레벨 및 루프 필터 유형 및/또는 강도를 가질 수 있다.

[0033] 인코더(14)가 전경 및 배경 이미지를 가지는(이 경우 전경 내의 이미지 데이터가 프레임마다 배경 이미지보다 더 많이 변함) 이미지 데이터를 포함하는 프레임을 코딩할 때, 세그먼트화 맵(100)이 유용할 수 있다. 프레임 시퀀스는 집 앞에서 대화를 하는 3명의 사람의 얼굴으로 포함할 수 있다. 3개의 얼굴은 전경 이미지의 일례일 것이고, 집은 배경 이미지의 일례일 것이다. 비디오 품질이 배경 세그먼트(118)에서보다 전경 세그먼트(112, 114, 116)에서 더 중요할 수 있기 때문에, 낮은 양자화 레벨 및 양호한 루프 필터 유형 및/또는 강도가 전경 세그먼트(112, 114, 116)에 적용될 수 있다.

[0034] 도 4b를 참조하면, 예시적인 세그먼트화 맵(200)은 프레임(210)을 2개의 세그먼트[하나의 몸체 세그먼트(212) 및 하나의 경계 세그먼트(214)]로 세그먼트화한다. 본 발명의 일 실시예에 따르면, 몸체 세그먼트(212)에 대응하는 모든 매크로블록(점을 찍은 매크로블록으로 나타냄)은 동일한 양자화 레벨 및 루프 필터 유형 및/또는 강도를 사용하여 인코딩되고, 경계 세그먼트(214)에 대응하는 모든 매크로블록(백색 매크로블록으로 나타냄)은 동일한 양자화 레벨 및 루프 필터 유형 및/또는 강도를 사용하여 인코딩된다.

[0035] 인코더(14)가 줌인이 있는 이미지 시퀀스를 코딩할 때, 세그먼트화 맵(200)이 유용할 수 있다. 비디오 품질이 경계 세그먼트(221)에서보다 몸체 세그먼트(212)에서 더 중요할 수 있기 때문에, 낮은 양자화 레벨 및 양호한 루프 필터 유형 및/또는 강도가 몸체 세그먼트(212)에 적용될 수 있다.

[0036] 도 4a 및 도 4b는 단지 세그먼트화 맵의 일례를 제공한다. 예를 들어, 다수의 세그먼트를 포함하거나, 동일한 크기 또는 동일하지 않은 크기의 세그먼트를 포함하거나, 연속적인 또는 비연속적인 매크로블록을 포함하거나, 사용자-정의된 또는 인코더 내에 사전 설정된 다른 적당한 세그먼트화 맵도 이용가능하다.

[0037] 세그먼트화 인에이블/디스에이블(즉, 세그먼트화 조건)의 선택 및 세그먼트화 맵의 정의는 인코더에 의해 결정되거나, 외부 응용 프로그램에 의해 전달되거나, 사용자에게 의해 지정될 수 있다. 일례로서, 화상 회의 응용 프로그램은 사람의 머리가 위치하는 곳의 세그먼트를 정의하는 맵을 전달될 수 있다.

[0038] 도 5는 본 발명의 일 실시예에 따른, 비디오 데이터의 세그먼트화를 제어하는 방법의 플로차트이다. 도 5를 참조하면, 단계(300)에서, 인코더(14)는 프레임 입력을 수신한다. 단계(302)에서, 인코더(14)는 이어서 현재 프

레이미 그 자신을 제외한 어떤 다른 프레임도 참조하지 않고 코딩된 프레임(흔히 키 프레임이라고 함)인지 여부를 판정한다. 현재 프레임이 키 프레임인 경우, 단계(304)에서 인코더(14)는 파라미터를 0으로 리셋하고, 모든 매크로블록을 동일한 세그먼트에 매핑하기 위해 단계(306)에서 세그먼트 맵을 리셋한다. 파라미터는, 예를 들어, 양자화 레벨 및 루프 필터 강도이다. 인코더(14)는 이어서 단계(308)로 진행한다.

[0039] 현재 프레임이 키 프레임이 아닌 경우 또는 현재 프레임이 키 프레임이고 인코더(14)가 파라미터 및 세그먼트 맵을 리셋한 경우, 단계(308)에서 인코더(14)는 세그먼트화가 인에이블되어 있는지를 판정한다. 각각의 프레임에 대한 비트스트림 헤더는 세그먼트화를 인에이블 또는 디스에이블하는 단일 비트를 포함한다. 그렇지만, 비트 대신에, 임의의 적당한 플래그, 데이터 구조 또는 기타 데이터 레코드가 사용될 수 있다. 세그먼트화가 디스에이블되어 있는 경우, 인코더(14)는 전체 프레임에 단계(310)에서 기본 양자화 파라미터를 적용할 수 있고, 단계(312)에서 기본 루프 필터 강도 파라미터를 적용할 수 있다.

[0040] 세그먼트화 파라미터는, 예를 들어, 양자화 레벨, 루프 필터 유형, 루프 필터 강도, 참조 프레임 유형 또는 서브-픽셀 보간 필터의 선택일 수 있다. 기타 세그먼트화 파라미터도 역시 이용가능하다.

[0041] 일례로서, 세그먼트 파라미터들 중 하나가 참조 프레임 유형일 수 있다. 참조 프레임 유형은 이미 인코딩되어 재구성된 하나 이상의 이전 프레임, 장래 프레임 또는 이들의 어떤 조합을 나타내는 값을 가질 수 있다. 따라서, 참조 프레임 유형은, 예를 들어, 마지막 프레임, 골든 프레임(golden frame) 또는 대안의 참조 프레임을 포함할 수 있다. 마지막 프레임은 현재 프레임보다 앞서 이전에 인코딩된 프레임일 수 있다. 골든 프레임은 후속 프레임에 대한 예측자로서 사용하기 위해 멀리 떨어진 과거로부터 임의적으로 선택된 과거 프레임일 수 있다. 대안의 참조 프레임은 마지막 프레임 또는 골든 프레임이 아닌 임의의 프레임을 포함할 수 있다. 예를 들어, 대안의 참조는 과거 프레임, 장래 프레임, 또는 구성된 참조 프레임일 수 있다. 게다가, 예를 들어, 구성된 참조 프레임은 본 발명의 양수인에게 양도되고 본 출원과 동시에 출원된, 발명의 명칭이 "System and Method for Video Encoding Using Constructed Reference Frame(구성된 참조 프레임을 사용하는 비디오 인코딩 시스템 및 방법)"인 특허 출원(참조 문헌으로서 본 명세서에 그 전체 내용이 포함됨)에서 개시된 참조 프레임일 수 있다.

[0042] 인코더(14) 및 디코더(21)는 이어서, 세그먼트 파라미터로 나타낸 바와 같이, 하나 이상의 참조 프레임 내의 주어진 세그먼트 id에 대응하는 매크로블록에 선택적인 업데이트를 적용할 수 있다. 이러한 방식으로, 전체 참조 프레임의 전체적인 업데이트에 의존할 필요없이, 현재 프레임으로부터의 정보를 사용하여 하나 이상의 참조 프레임의 일부(예를 들어, 배경)를 업데이트하는 방법으로서 세그먼트화가 사용될 수 있다.

[0043] 적응적 세그먼트화 발명의 실시예가 단지 2개의 세그먼트화 파라미터(즉, 양자화 레벨 및 루프 필터 강도)를 기술하고 있지만, 본 발명의 대안의 실시예는 상이한 세그먼트화 파라미터 또는 다수의 세그먼트화 파라미터를 사용하여 구현될 수 있다.

[0044] 세그먼트화가 디스에이블되어 있는 경우, 인코더(14)는 임의의 이전에 전송된 세그먼트화 맵은 물론 임의의 이전에 전송된 세그먼트화 파라미터를 무시할 수 있다. 이 세그먼트화 인에이블/디스에이블 이진 스위치 또는 세그먼트화 조건은 오버헤드를 최소화하고 세그먼트화가 이점을 거의 또는 전혀 제공하지 않는 프레임들에 대해 프레임마다 1 비트 이하를 생성한다.

[0045] 여전히 도 5를 참조하면, 그렇지만, 세그먼트화가 인에이블되어 있는 경우, 단계(314)에서 인코더(14)는 세그먼트화 파라미터가 업데이트되어야만 하는지를 판정할 수 있다. 세그먼트화 파라미터에 대한 어떤 업데이트도 표시되지 않은 경우, 단계(316)에서 인코더(14)는 이전에 전송된 세그먼트화 파라미터 값을 사용할 수 있다. 환언하면, 현재 비디오 프레임에 대해 세그먼트화 파라미터는 업데이트될 수 없다. 이 업데이트 세그먼트화 파라미터 이진 스위치는 세그먼트화 파라미터를 재전송하지 않음으로써 오버헤드를 최소화하며, 이 경우 이전 프레임으로부터의 세그먼트화 파라미터가 현재 프레임에 대해 사용될 수 있다.

[0046] 세그먼트화 파라미터에 대한 업데이트가 표시된 경우, 인코더(14)는 세그먼트화 파라미터를 업데이트할 수 있다. 구체적으로는, 예시적인 실시예에서, 단계(318)에서 양자화 레벨 파라미터가 업데이트될 수 있고, 단계(320)에서 루프 필터 강도 파라미터가 업데이트될 수 있다. 업데이트된 세그먼트화 파라미터 값은 절대값이거나, 다른 대안으로서 기준선 값에 가산되는 델타(즉, 증분 또는 감분) 값일 수 있다. 파라미터 값이 델타인 경우, 세그먼트화 파라미터 값이 허용 범위를 초과하는 것을 방지하기 위해 클램핑 메커니즘이 구현될 수 있다.

[0047] 여전히 도 5를 참조하면, 단계(314)에서 세그먼트화 파라미터가 업데이트되는지 여부에 상관없이, 단계(322)에

서 인코더(14)는 세그먼트화 맵(100, 200)이 업데이트되어야 하는지를 판정할 수 있다. 세그먼트화 맵(100, 200)에 대한 어떤 업데이트도 표시되지 않은 경우, 단계(324)에서 인코더(14)는 이전에 전송된 세그먼트화 맵을 사용할 수 있다. 이 업데이트 세그먼트화 맵 이진 스위치는 세그먼트화 맵(100, 200)을 재전송하지 않음으로써 오버헤드를 최소화하며, 이 경우 이전 프레임으로부터의 세그먼트화 맵이 현재 프레임에 대해 사용될 수 있다.

[0048] 세그먼트화 맵(100, 200)에 대한 업데이트가 표시된 경우, 단계(326)에서 인코더(14)는 세그먼트 식별자 비트 각각의 발생 가능성을 업데이트할 수 있다. 발생 가능성이 업데이트된 후에, 단계(328)에서 인코더(14)는 현재 프레임 내의 매크로블록에 대한 세그먼트 식별자를 업데이트할 수 있다. 다시 도 4a를 참조하면, 예를 들어, 세그먼트 식별자는 세그먼트화 맵(100)에서 각각의 매크로블록을 특정의 세그먼트(112, 114, 116, 118)와 연관시킨다. 따라서, 4개의 세그먼트(112, 114, 116, 118)가 있는 경우, 예를 들어, 00, 01, 10 및 11로서 식별될 수 있는 4개의 세그먼트 식별자가 있을 수 있다. 도 5의 프로세스는 인코더(14)에 의해 수신되는 각각의 프레임에 대해 반복될 수 있다.

[0049] 적응적 세그먼트화에서 세그먼트화 파라미터 및 세그먼트 맵의 업데이트를 제어하기 위해 프레임 레벨 헤더 정보를 구현하는 예시적인 의사 코드가 이하에 예시되어 있다.

```
[0050] // Is Segmentation Enabled
[0051] WriteBit( SegmentationEnabled );
[0052] if ( SegmentationEnabled )
[0053] {
[0054]     // Signal whether or not the segmentation map and parameters are to be updated in this frame
[0055]     WriteBit( UpdateMbSegmentationMap );
[0056]     WriteBit( UpdateMbSegmentationParams );
[0057]     if (UpdateMbSegmentationParams)
[0058]     {
[0059]         WriteBit( MbSegmentAbsOrDelta );           // 0 indicates parameters are delta values, 1 absolute
values
[0060]         // For each segment parameter (quantizer and loop filter strength)
[0061]         for ( i = 0; i < 2; i++ )
[0062]         {
[0063]             for ( j = 0; j < 4; j++ )                // For each of the segments
[0064]             {
[0065]                 Data = SegmentFeatureData[i][j];
[0066]                 // Frame level data
[0067]                 if ( Data )                          // parameter is non zero
[0068]                 {
[0069]                     WriteBit (1);
[0070]                     if (Data < 0)
[0071]                     {
[0072]                         WriteBit (1);                // - value (sign bit)
[0073]                         Data = -Data
[0074]                     }

```

```

[0075]     else
[0076]         WriteBit (0);           // + value (sign bit)
[0077]                                     // In VP8 7 bits for quantizer parameter and 6 for loop filter
[0078]         WriteLiteral( Data, MbFeatureDataBits[i]);
[0079]     }
[0080]     else
[0081]         WriteBit(0);           // Parameter was set to 0.
[0082]     }
[0083] }
[0084] }
[0085]
[0086] if ( UpdateMbSegmentationMap )
[0087] {
[0088]     // Write out the tree node probabilities used to decode the segment id for each macro block.
[0089]     for ( i = 0; i < 3; i++ )
[0090]     {
[0091]         Data = MbSegmentTreeProbs[i];
[0092]         if ( Data != 255)
[0093]         {
[0094]             WriteBit(1);
[0095]             WriteLiteral(Data, 8);
[0096]         }
[0097]         else
[0098]             WriteBit(0);
[0099]     }
[0100] }
[0101] }

```

[0102] 상기 의사 코드를 참조하면, 이하는 변수 및 어레이의 목록과 본 발명의 실시예에 대한 그의 의미이다.

[0103] SegmentationEnabled: 세그먼트화 인에이블/디스에이블 이진 스위치.

[0104] UpdateMbSegmentationMap: 세그먼트화 맵 업데이트/업데이트 없음 이진 스위치.

[0105] UpdateMbSegmentationParams: 세그먼트화 파라미터 업데이트/업데이트 없음 이진 스위치.

[0106] MbSegmentAbsOrDelta: SegmentFeatureData 어레이 내의 세그먼트화 파라미터의 값이 델타값인지 절대값인지를 나타내는 이진 스위치.

[0107] SegmentFeatureData[i][j]: 세그먼트화 파라미터의 값(예를 들어, 양자화기 또는 루프 필터 강도)

[0108] MbSegmentTreeProbs[i]: 트리 노드 확률(이하 도 6 참조).

- [0109] 의사 코드를 참조하면, 세그먼트화가 인에이블되어 있는지 여부를 나타내기 위해 SegmentationEnabled 변수가 비트스트림에 기입될 수 있다. 세그먼트화가 인에이블되어 있지 않은 경우, 세그먼트화 파라미터도 세그먼트화 맵도 업데이트되지 않으며, 앞서 기술한 바와 같이, 임의의 이전에 전송된 세그먼트화 파라미터 또는 세그먼트화 맵이 인코더(14) 및 디코더(21)에 의해 무시된다. 그렇지만, 세그먼트화가 인에이블되어 있는 경우, 세그먼트화 파라미터가 업데이트되어야만 하는지 여부를 나타내기 위해 UpdateMbSegmentationParams 비트가 기입될 수 있고, 세그먼트 맵이 업데이트되어야만 하는지 여부를 나타내기 위해 UpdateMbSegmentationMap이 기입될 수 있다. 비트 대신에, 임의의 적당한 플래그, 데이터 구조 또는 기타 데이터 레코드도 역시 사용될 수 있다.
- [0110] UpdateMbSegmentationParams가 참인 경우, 세그먼트화 파라미터가 델타값인지 절대값인지를 나타내기 위해 MbSegmentAbsOrDelta 비트가 비트스트림에 기입될 수 있다. 이어서, 4개의 세그먼트 각각에 대한 2개의 세그먼트 파라미터(즉, 양자화기 값 및 루프 필터 강도) 각각에 대해, SegmentFeatureData[i][j] 내의 데이터는 그의 대응하는 부호 비트(있는 경우)와 함께 비트스트림에 기입될 수 있다. 비트 대신에, 임의의 적당한 플래그, 데이터 구조 또는 기타 데이터 레코드도 역시 사용될 수 있다.
- [0111] UpdateMbSegmentationMap이 참인 경우, 인코더(14)는 MbSegmentTreeProbs[i] 어레이 내의 임의의 값이 255인지 여부를 테스트함으로써 MbSegmentTreeProbs[i] 내의 데이터가 업데이트될 수 있는지 여부를 판정한다. 어레이 내의 값들 중 하나가 255인 경우, 특정의 노드 확률이 업데이트되어서는 안된다. 그렇지 않은 경우, MbSegmentTreeProbs[i] 내의 각각의 노드 확률이 비트스트림에 기입될 수 있다.
- [0112] 노드 확률을 업데이트하는 비용이 프레임 레벨에서 발생될 수 있고 최소일 수 있다. 상기 의사 코드에 따르면, 비용은 몇개의 확률이 업데이트되는지에 따라 프레임마다 3 비트와 27 비트 사이의 어딘가일 수 있다. 하한 비용에서, 조건이 노드 확률 중 어느 것도 업데이트되어서는 안된다는 것을 나타낼 때 3 비트가 얻어진다. 따라서, 인코더(14)는 Writebit(0) 동작을 3회 수행할 수 있다. 상한 비용에서, 조건이 노드 확률 중 3개 모두가 업데이트되어야 한다는 것을 나타낼 때 27 비트가 얻어진다. 따라서, 인코더(14)는 Writebit(1) 동작을 3회 수행하고 각각 8 비트의 비용으로 MbSegmentTreeProbs[i]로부터 3개의 확률 값을 기입할 수 있다.
- [0113] 다른 대안으로서, 일 실시예에서, UpdateMbSegmentationMap이 참인 경우, 각각 8 비트의 비용으로 되어 있는 MbSegmentTreeProbs[i]로부터의 3개의 확률 모두가 항상 24 비트의 비용으로 비트스트림에 기입된다. 이 실시예를 나타내는 예시적인 의사 코드가 이하에 나타내어져 있다:
- [0114] if ( UpdateMbSegmentationMap )
- [0115] {
- [0116] // Write out the tree node probabilities used to decode the segment id for each macro block.
- [0117] for ( i = 0; i < 3; i++ )
- [0118] {
- [0119] Data = MbSegmentTreeProbs[i];
- [0120] WriteLiteral(Data, 8);
- [0121] }
- [0122] }
- [0123] 도 6은 특정의 결과(406, 408, 410, 412)가 나올 수 있는 확률을 사용하여 세그먼트 식별자를 코딩하는 데 사용되는 이진 트리를 나타낸 것이다. 각각의 결과(406, 408, 410, 412)는 4개의 세그먼트 식별자: 00, 01, 10 및 11 중 하나를 나타낸다. 앞서 언급한 바와 같이, 본 발명의 실시예는 상이한 수의 세그먼트 식별자를 가질 수 있다. 다양한 실시예에서 세그먼트의 수에 따라, 특정 수의 세그먼트를 코딩하는 데 필요한 확률의 수를 나타내기 위해 도 6의 이진 트리가 그에 따라 수정될 것이다. 따라서, 예를 들어, 일 실시예가 8개의 세그먼트를 포함하고 있는 경우, 이진 트리는 8개의 특정 결과를 코딩하기 위해 7개의 상이한 노드에 7개의 확률을 포함할 것이다.
- [0124] 각각의 노드(400, 402, 404)는, 예를 들어, 좌측 분기가 취해질 수 있는 가능성을 정의하는 연관된 확률을 가진다. 도 6에 나타낸 바와 같이, 좌측 분기는 0을 나타내는 비트에 대응하는 반면, 우측 분기는 1을 나타내는 비트에 대응한다. 좌측 분기가 취해질 수 있는 가능성은 값이 각각의 노드(400, 402, 404)에 표현되는 것에 대응

할 수 있는 확률이다. 따라서, 노드 400에서의 확률 1은 값이 2보다 작을 수 있는 가능성을 나타낸다. 환언하면, 확률 1은 결과(406, 408)가 나올 수 있는 가능성을 나타내는데, 그 이유는 이들 결과가 이진 00 및 이진 01(각각 2보다 작음)의 값을 가지기 때문이다. 수식 1 - 확률 1은 노드(400)의 우측 분기가 나올 수 있는 가능성을 계산하는 데 사용될 수 있다. 환언하면, 1 - 확률 1은 결과(410, 412)가 나올 수 있는 가능성을 나타내는데, 그 이유는 이들 결과가 이진 10 및 이진 11(각각 2보다 큼)의 값을 가지기 때문이다.

[0125] 노드 402에서의 확률 2는 값이 0일 수 있는 가능성을 나타낸다. 환언하면, 확률 2는 결과(406)가 나올 수 있는 가능성을 나타내는데, 그 이유는 이 결과가 이진 00(0임)의 값을 가지기 때문이다. 수식 1 - 확률 2는 노드 (402)의 우측 분기가 나올 수 있는 가능성을 계산하는 데 사용될 수 있다. 환언하면, 1- 확률 2는 결과(408)가 나올 수 있는 가능성을 나타내는데, 그 이유는 이 결과가 이진 01(0이 아님)의 값을 가지기 때문이다.

[0126] 노드 404에서의 확률 3은 값이 2일 수 있는 가능성을 나타낸다. 환언하면, 확률 3은 결과(410)가 나올 수 있는 가능성을 나타내는데, 그 이유는 이 결과가 이진 10(2임)의 값을 가지기 때문이다. 수식 1 - 확률 3은 노드 (404)의 우측 분기가 나올 수 있는 가능성을 계산하는 데 사용될 수 있다. 환언하면, 1- 확률 3은 결과(412)가 나올 수 있는 가능성을 나타내는데, 그 이유는 이 결과가 이진 11(2가 아님)의 값을 가지기 때문이다.

[0127] 확률이, 예를 들어, 1-254 범위의 정수로서 표현될 수 있으며, 여기서 1은 각각의 노드(400, 402, 404)의 좌측 분기가 나올 수 있는 가능성이 거의 없다는 것을 나타내고, 254는 각각의 노드(400, 402, 404)의 좌측 분기가 나올 수 있는 가능성이 아주 많다는 것을 나타낸다. 각각의 확률은 각각의 세그먼트에 할당된 매크로블록의 수를 반영한다. 따라서, 대부분의 매크로블록이 세그먼트 00에 할당되는 경우, 확률 1 및 확률 2가 높을 수 있다. 확률은 또한 다른 범위 또는 값을 사용하여 표현되고 우측 분기가 취해질 수 있는 가능성 등의 다른 의미를 암시할 수 있다.

[0128] 다양한 유형의 코딩 방식(산술 코딩 또는 허프만 코딩 등)이 있다. 산술 코딩이 전체 메시지 시퀀스를 하나의 숫자로 변환하는 반면 허프만 코딩이 메시지의 각각의 심볼을 일련의 숫자로 변환하기 때문에, 산술 코딩이 허프만 코딩보다 더 효율적일 수 있다. 따라서, 본 발명의 바람직한 실시예에서 산술 코딩이 사용될 수 있다. 본 발명의 대안의 실시예에서 허프만 코딩과 같은 다른 코딩 방식 또는 코딩 방식들의 조합이 사용된다.

[0129] 4개의 세그먼트 식별자 중 하나를 표현하기 위해 3개의 확률 각각을 사용하여 0 또는 1을 엔트로피 인코딩 단계 (24)(도 1 참조)에서 인코딩하거나 엔트로피 디코딩 단계(25)(도 2 참조)에서 디코딩하는 데 산술 인코더가 사용될 수 있다. 산술 코딩은 일반적으로 메시지 시퀀스를 주어진 범위 내의 하나의 부동 소수점 출력 숫자로서 표현하는 것을 포함한다. 통상적으로, 이 범위는 0부터 1까지의 확률선(probability line)일 수 있다. 메시지 시퀀스 내의 각각의 심볼(즉, 세그먼트 식별자)은 확률선의 일부를 차지하는 범위에 의해 표현될 수 있다. 심볼이 높은 확률을 가지는 경우, 그 심볼은 확률선 상의 큰 범위를 받을 수 있고, 심볼이 낮은 확률을 가지는 경우, 그 심볼은 확률선 상의 작은 범위를 받을 수 있다.

[0130] 코딩 동안에, 메시지 시퀀스를 표현하는 데 사용되는 실수는 각각의 심볼이 어느 범위에 속하는지에 기초하여 작성될 수 있다. 메시지 내의 각각의 심볼이 처리될 때, 범위가 심볼에 할당된 그 부분으로 좁혀질 수 있다. 메시지가 더 길어짐에 따라, 그를 표현하는 데 필요한 범위가 더 작아지고, 그 범위를 지정하는 데 필요한 비트의 수가 늘어난다. 가능성이 많은 심볼(즉, 높은 확률을 가지는 심볼)은 가능성이 적은 심볼(즉, 낮은 확률을 가지는 심볼)보다 적게 범위를 감소시키며, 따라서 적은 수의 비트를 메시지에 추가한다.

[0131] 다시 도 6을 참조하면, 각각의 세그먼트 식별자가 개별적으로 코딩되는 경우, 세그먼트 식별자당 2 비트의 비용일 것이다. 그렇지만, 상기한 산술 코딩 방식을 사용할 때, 4개의 세그먼트 식별자 모두가 사용되고 있지만 일부가 다른 것보다 높은 확률을 가지는 경우, 세그먼트 식별자를 코딩하는 평균 비용은 2 비트보다 훨씬 더 적을 수 있다. 4개 미만의 세그먼트 식별자가 사용되는 경우, 각각의 세그먼트 식별자를 코딩하는 비용은 훨씬 더 적을 수 있다.

[0132] 도 7은 본 발명의 일 실시예에 따른, 각각의 매크로블록에 대한 세그먼트 식별자를 인코딩하는 방법의 플로우차트이다. 단계(500)에서, 인코더(14)는 제1 매크로블록을 ID=1로 설정한다. ID는 프레임에서 현재 인코딩되고 있는 매크로블록 식별 번호를 나타낸다.

[0133] 이어서, 단계(502)에서 인코더(14)는 세그먼트화가 인에이블되어 있는지 여부를 판정한다. 세그먼트화가 인에이블되어 있지 않은 경우, 매크로블록에 대해 세그먼트 식별자가 인코딩되지 않는다. 세그먼트화가 인에이블되어 있는 경우, 단계(504)에서 인코더(14)는 세그먼트화 맵이 업데이트되어야 하는지 여부를 판정한다. 세그먼트화 맵에 대한 업데이트가 표시되어 있지 않은 경우, 매크로블록에 대한 새로운 세그먼트 식별자가 인코딩되지 않음

며, 이전의 세그먼트 맵이 코딩하는 데 사용될 수 있다.

[0134] 그렇지만, 세그먼트화 맵에 대한 업데이트가 표시되어 있는 경우, 매크로블록이 세그먼트 맵 내에서의 그의 위치에 기초하여 세그먼트 ID를 할당받을 수 있다. 따라서, 단계(506)에서 인코더(14)는 식별 번호 ID를 가지는 매크로블록이 세그먼트 0에 매핑되어야 하는지 여부를 판정한다. 식별 번호 ID를 가지는 매크로블록이 세그먼트 0에 매핑되는 경우, 단계(508)에서 인코더(14)는 식별 번호 ID를 가지는 매크로블록에 대해 세그먼트 0(이진 00)을 인코딩한다. 식별 번호 ID를 가지는 매크로블록이 세그먼트 0에 매핑되지 않는 경우, 단계(510)에서 인코더(14)는 식별 번호 ID를 가지는 매크로블록이 세그먼트 1에 매핑되어야 하는지 여부를 판정한다. 식별 번호 ID를 가지는 매크로블록이 세그먼트 1에 매핑되는 경우, 단계(512)에서 인코더(14)는 식별 번호 ID를 가지는 매크로블록에 대해 세그먼트 1(이진 01)을 인코딩한다. 식별 번호 ID를 가지는 매크로블록이 세그먼트 1에 매핑되지 않는 경우, 단계(514)에서 인코더(14)는 식별 번호 ID를 가지는 매크로블록이 세그먼트 2에 매핑되어야 하는지 여부를 판정한다. 식별 번호 ID를 가지는 매크로블록이 세그먼트 2에 매핑되는 경우, 단계(516)에서 인코더(14)는 식별 번호 ID를 가지는 매크로블록에 대해 세그먼트 2(이진 10)를 인코딩한다. 식별 번호 ID를 가지는 매크로블록이 세그먼트 2에 매핑되지 않는 경우, 단계(518)에서 인코더(14)는 식별 번호 ID를 가지는 매크로블록에 대해 세그먼트 3(이진 11)을 인코딩한다.

[0135] 단계(508, 512, 516 또는 518)에서 매크로블록에 대해 세그먼트 식별자가 인코딩된 경우, 인코더(14)는 단계(520)에서 ID가 현재 프레임에서의 매크로블록의 총수를 나타내는 NUM\_OF\_MBS보다 작은지 여부를 판정한다. ID가 매크로블록의 총수보다 작지 않은 경우, 현재 프레임 내의 각각의 매크로블록에 대해 세그먼트 식별자가 더 이상 인코딩될 필요가 없다. 그렇지 않고, ID가 NUM\_OF\_MBS보다 작은 경우, 인코더(14)는 단계(522)에서 ID를 1만큼 증가시키고, 모든 매크로블록 세그먼트 식별자가 현재 프레임에 대해 인코딩될 때까지 프로세스를 반복한다.

[0136] 도 7의 방법의 단계들을 구현하는 예시적인 의사 코드가 이하에 나타내어져 있다:

[0137] 프레임 내의 각각의 매크로블록에 대해,

[0138] {

[0139] ...

[0140] ...

[0141] // If segmentation is enabled and a map update is indicate then

[0142] // encode the segment id for this macroblock

[0143] if ( SegmentationEnabled && UpdateMbSegmentationMap )

[0144] {

[0145] switch ( SegmentId )

[0146] {

[0147] case 0:

[0148] EncodeBit (0, MbSegmentTreeProbs[0]);

[0149] EncodeBit (0, MbSegmentTreeProbs[1]);

[0150] break;

[0151] case 1:

[0152] EncodeBit (0, MbSegmentTreeProbs[0]);

[0153] EncodeBit (1, MbSegmentTreeProbs[1]);

[0154] break;

[0155] case 2:

```

[0156]         EncodeBit (1, MbSegmentTreeProbs[0]);
[0157]         EncodeBit (0, MbSegmentTreeProbs[2]);
[0158]         break;
[0159]     case 3:
[0160]         EncodeBit (1, MbSegmentTreeProbs[0]);
[0161]         EncodeBit (1, MbSegmentTreeProbs[2]);
[0162]         break;
[0163]     }
[0164] }
[0165] }
    
```

[0166] 상기 의사 코드를 참조하면, 프레임 내의 각각의 매크로블록에 대해, 세그먼트화가 인에이블되어 있고 세그먼트화 맵이 업데이트되는 경우, 세그먼트 식별자는, 예를 들어, EncodeBit(value, probability) 함수를 사용하여 인코딩된다. 인코더에서 EncodeBit 함수를 2번 호출함으로써, 각각의 매크로블록에 대해 세그먼트 식별자가 인코딩된다. EncodeBit 함수에 대한 각각의 호출에서, 인코더는 코딩 프로세스에 대한 컨텍스트로서 3개의 "확률" 값 중 하나를 사용하여 0이거나 1인 입력 비트 "값"을 코딩한다. 3개의 확률 값은 MbSegmentTreeProbs라고 하는 3개의 값의 어레이로서 지정된다. 컨텍스트는 각각의 세그먼트 식별자가 입력 시퀀스에서 나올 가능성이 얼마나 되느냐에 따라 각각의 세그먼트 식별자의 확률 분포에 관한 안내를 인코더에게 제공한다.

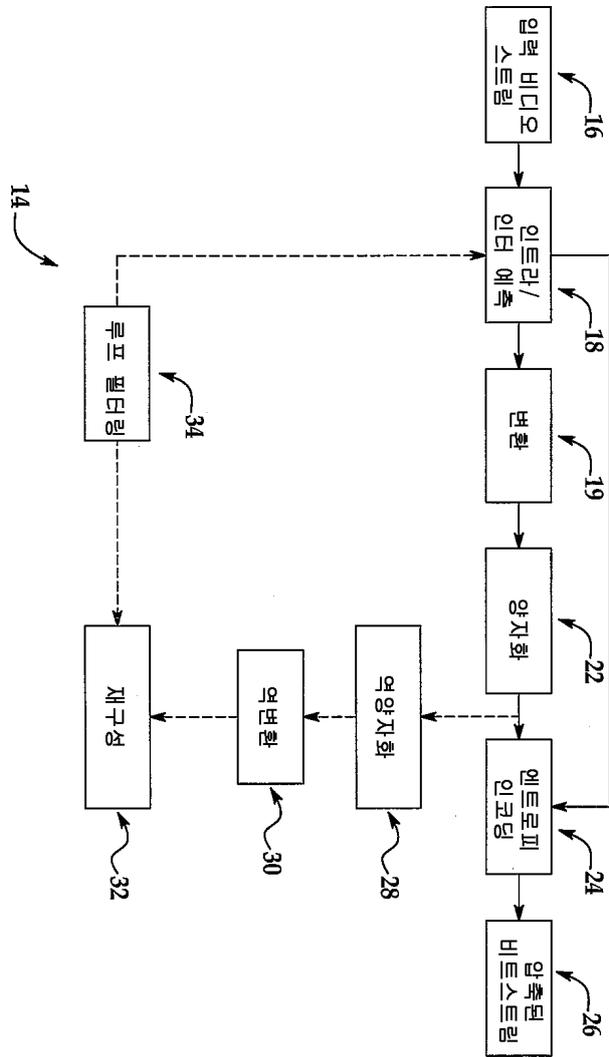
[0167] 상기한 예시적인 의사 코드는 4개의 세그먼트 식별자를 지원하는 구현예이다. 앞서 기술한 바와 같이, 다른 수의 세그먼트가 지원될 수 있고, 의사 코드는 정확한 수의 세그먼트를 나타내기 위해 그에 따라 수정될 수 있다. 게다가, 예시적인 의사 코드는 임의의 특정의 프로그래밍 언어 및 그의 구현으로 제한되는 것으로 보아서는 안 된다. 본 명세서에 기술된 본 발명의 실시예의 개시 내용을 구현하는 데 각종의 프로그래밍 언어 및 그의 구현이 사용될 수 있다는 것을 잘 알 수 있다.

[0168] 본 발명이 특정 실시예와 관련하여 기술되어 있지만, 본 발명이 개시된 실시예로 제한되지 않으며 그와 달리 첨부된 특허청구범위의 사상 및 범위 내에 포함되는 다양한 수정예 및 등가 구성을 포함하기 위한 것이며, 법이 허용하는 모든 이러한 수정예 및 등가 구조를 포함하기 위해 그 범위가 최광의로 해석되어야 한다는 것을 잘 알 것이다.

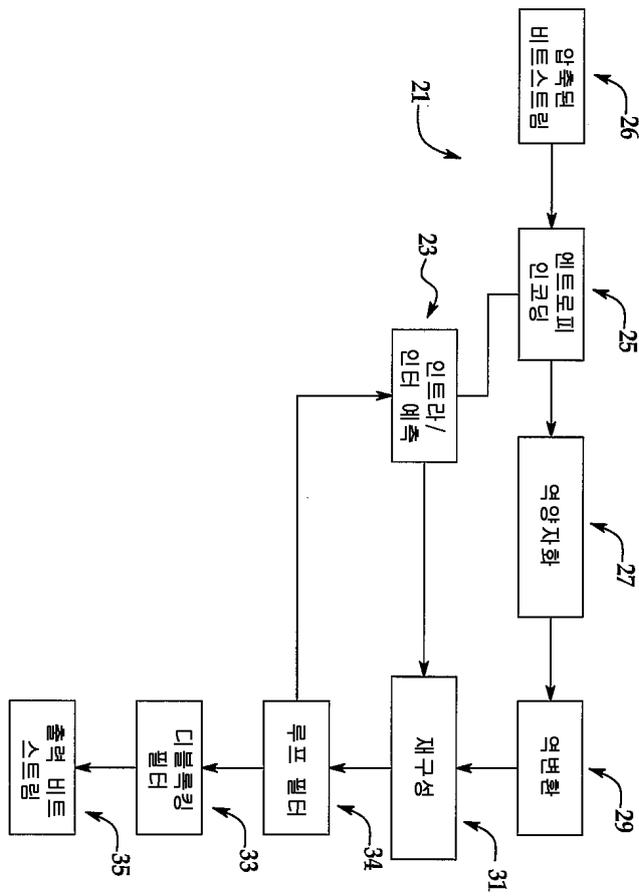
**부호의 설명**

- [0169] 16: 입력 비디오 스트림
- 18: 인트라/인터 예측
- 19: 변환
- 22: 양자화
- 24: 엔트로피 인코딩
- 26: 압축된 비트스트림
- 28: 역양자화
- 30: 역변환
- 32: 재구성
- 34: 루프 필터링

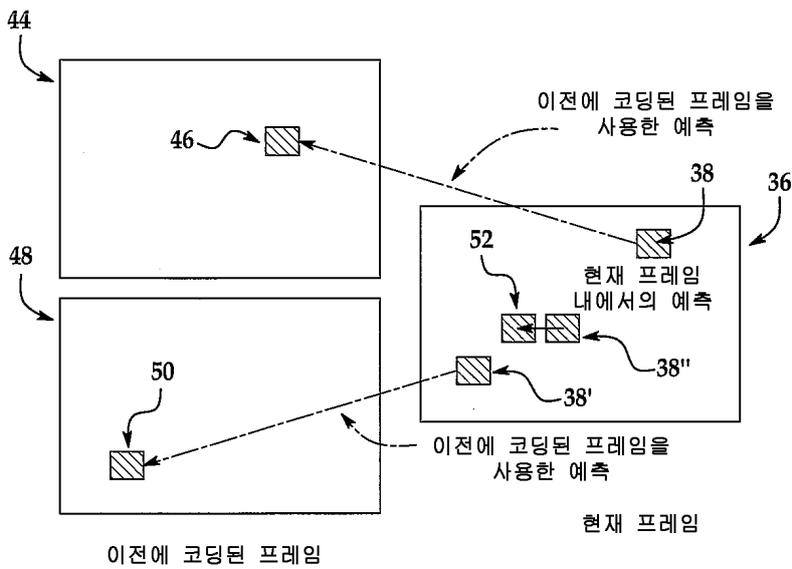
도면  
도면1



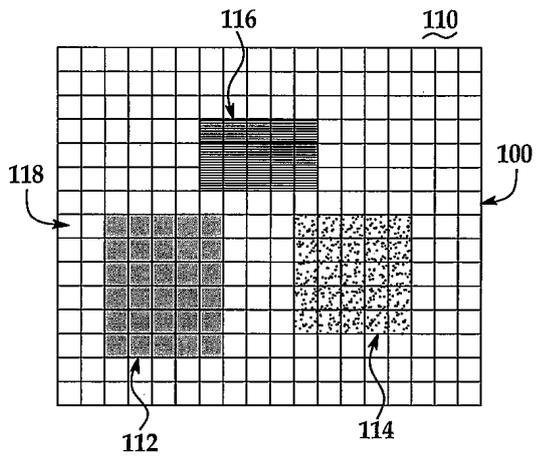
도면2



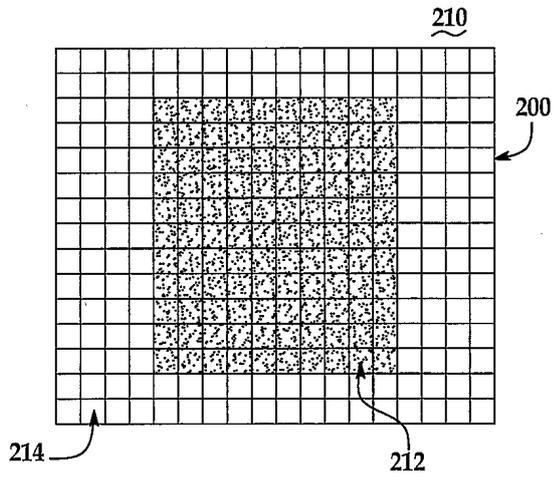
도면3



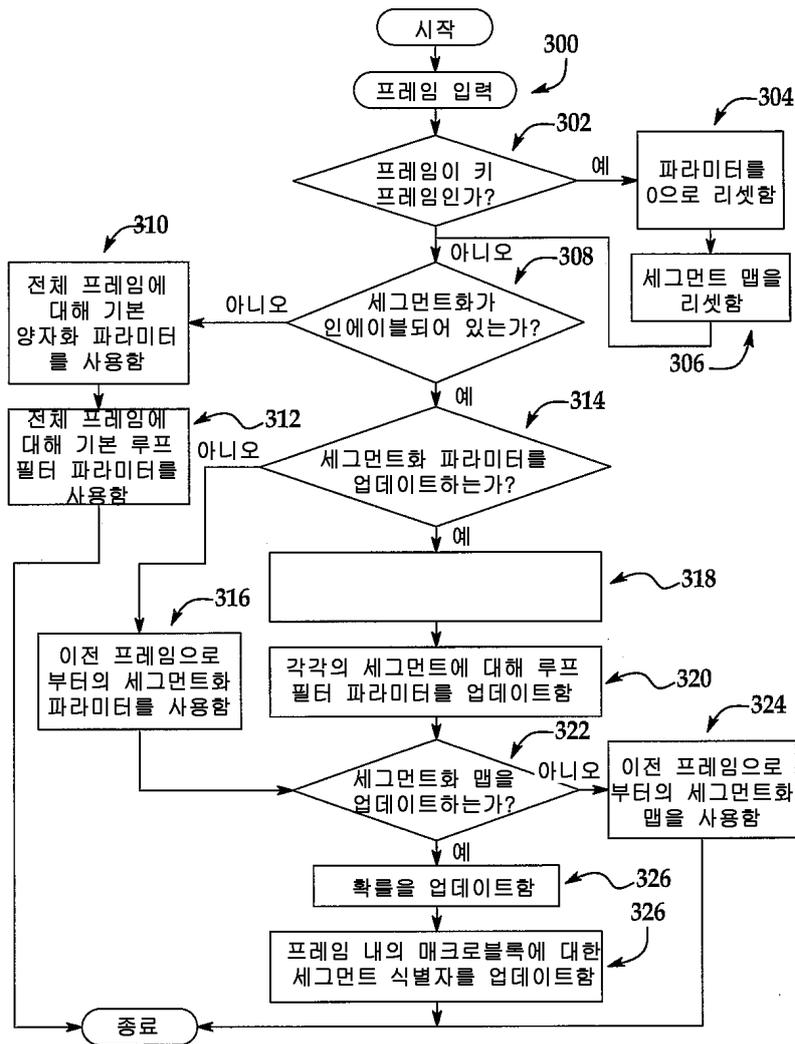
도면4a



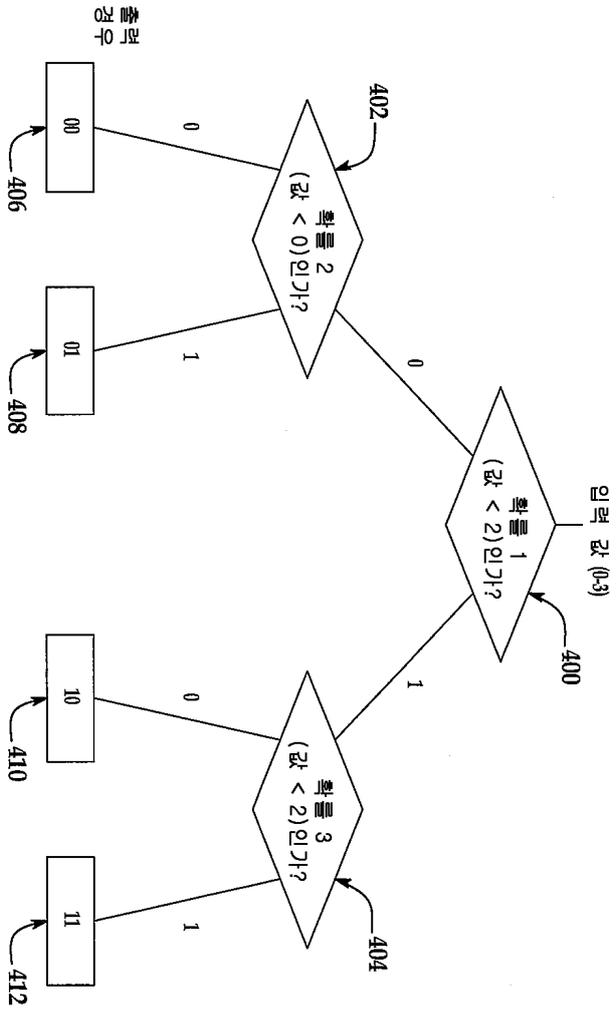
도면4b



도면5



도면6



도면7

