



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2010년03월09일  
(11) 등록번호 10-0946600  
(24) 등록일자 2010년03월03일

(51) Int. Cl.

H04N 7/24 (2006.01)

(21) 출원번호 10-2003-7017297

(22) 출원일자 2002년07월02일

심사청구일자 2007년07월02일

(85) 번역문제출일자 2003년12월31일

(65) 공개번호 10-2004-0018414

(43) 공개일자 2004년03월03일

(86) 국제출원번호 PCT/US2002/021151

(87) 국제공개번호 WO 2003/005626

국제공개일자 2003년01월16일

(30) 우선권주장

60/302,853 2001년07월02일 미국(US)

10/180,828 2002년06월26일 미국(US)

(56) 선행기술조사문헌

US04903317 A1\*

WO0135673 A1\*

DCC '96. Proceedings; Weinberger, M.J.; Seroussi, G.; Sapiro, G.; "LOCO-I: a low complexity, context-based, lossless image compression algorithm", (31 March-3 Apr. 1996 Page(s):140-149)\*

\*는 심사관에 의하여 인용된 문헌

(73) 특허권자

켈컴 인코퍼레이티드

미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775

(72) 발명자

라빈드랜비제이아라크시미알

미국92130캘리포니아주샌디에고칼레마르테발레나스4272

시야가라얀카다암

미국92130캘리포니아주샌디에고설스턴플레이스482  
(뒷면에 계속)

(74) 대리인

특허법인코리어나

전체 청구항 수 : 총 14 항

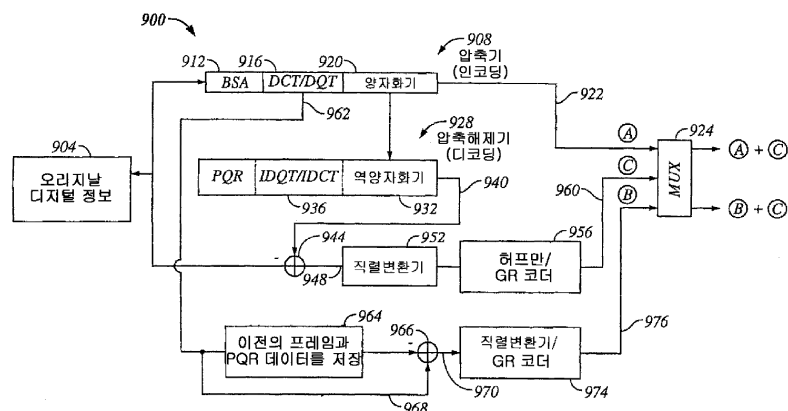
심사관 : 조우연

(54) 무손실 방식으로 디지털 이미지 데이터를 인코딩하기 위한 장치 및 방법

(57) 요약

이미지 정보를 표현하는 신호를 무손실 압축 및 인코딩하는 방법이 청구된다. 무손실 압축된 데이터 파일 및 리지듀얼 압축된 데이터 파일이 생성된다. 손실 압축된 데이터 파일 및 리지듀얼 압축된 데이터 파일이 결합될 때, 오리지널 데이터 파일과 실질적으로 동일한 무손실 데이터 파일이 생성된다.

대표도



(72) 발명자

**라셀존**

미국92129캘리포니아주샌디에고팜퍼널웨이12823

**몰리스티븐에이**

미국92025

캘리포니아주에스콘디도로스트오크레인540

---

## 특허청구의 범위

### 청구항 1

각각이 오리지널 이미지를 포함하는 복수의 프레임을 포함하는 비디오 데이터를 인코딩하는 방법으로서,

상기 복수의 프레임 내 각각의 프레임의 상기 오리지널 이미지와 관련된 손실 압축된 데이터 파일을 생성하는 단계;

상기 복수의 프레임 내 각각의 프레임의 상기 오리지널 이미지와 관련된 리지듀얼 (residual) 압축된 데이터 파일을 생성하는 단계로서, 각각의 리지듀얼 압축된 데이터 파일의 생성은 각각의 오리지널 이미지를 각각 적응적으로 블록 사이징함으로써 이전에 생성된 데이터를 이용하는, 단계; 및

무손실 데이터 파일을 생성하기 위해 각각의 손실 압축된 데이터 파일을 각각의 리지듀얼 압축된 데이터 파일과 결합하는 단계를 포함하고,

상기 무손실 데이터 파일은, 각각의 프레임의 이미지가 상기 복수의 프레임 내 각각의 프레임의 상기 오리지널 이미지와 실질적으로 동일한, 이미지를 산출하도록 이용될 수 있는, 신호의 무손실 압축 및 인코딩 방법.

### 청구항 2

제 1 항에 있어서,

상기 손실 압축된 데이터 파일 및 상기 리지듀얼 압축된 데이터 파일은 인트라프레임 기반 또는 인터프레임 기반으로 생성되는, 신호의 무손실 압축 및 인코딩 방법.

### 청구항 3

제 1 항에 있어서,

상기 생성은 DCT (discrete cosine transform) 및 DQT (differential quadtree transform) 기술의 결합을 이용하는, 신호의 무손실 압축 및 인코딩 방법.

### 청구항 4

제 1 항에 있어서,

상기 생성은 GR (Golomb-Rice) 코딩 기술을 이용하는, 신호의 무손실 압축 및 인코딩 방법.

### 청구항 5

각각이 오리지널 이미지를 포함하는 복수의 프레임을 포함하는 비디오 데이터를 인코딩하는 장치로서,

상기 복수의 프레임 내 각각의 프레임의 상기 오리지널 이미지와 관련된 손실 압축된 데이터 파일을 생성하는 수단;

상기 복수의 프레임 내 각각의 프레임의 상기 오리지널 이미지와 관련된 리지듀얼 압축된 데이터 파일을 생성하는 수단으로서, 각각의 리지듀얼 압축된 데이터 파일은 각각의 오리지널 이미지를 적응적으로 블록 사이징함으로써 이전에 생성된 데이터를 이용하여 생성되는, 수단; 및

무손실 데이터 파일을 생성하기 위해 각각의 손실 압축된 데이터 파일을 각각의 리지듀얼 압축된 데이터 파일과 결합하는 수단을 포함하고,

상기 무손실 데이터 파일은, 각각의 프레임의 이미지가 상기 복수의 프레임 내 각각의 프레임의 상기 오리지널 이미지와 실질적으로 동일한, 이미지를 산출하도록 이용될 수 있는, 신호의 무손실 압축 및 인코딩 장치.

### 청구항 6

제 5 항에 있어서,

상기 손실 압축된 데이터 파일을 생성하는 수단 및 상기 리지듀얼 압축된 데이터 파일을 생성하는 수단은 인트

라프레임 기반 또는 인터프레임 기반으로 생성되는, 신호의 무손실 압축 및 인코딩 장치.

#### 청구항 7

제 5 항에 있어서,

상기 생성 수단은 DCT (discrete cosine transform) 및 DQT (differential quadtree transform) 기술의 결합을 이용하는, 신호의 무손실 압축 및 인코딩 장치.

#### 청구항 8

제 5 항에 있어서,

상기 생성 수단은 GR (Golomb-Rice) 코딩 기술을 이용하는, 신호의 무손실 압축 및 인코딩 장치.

#### 청구항 9

각각이 오리지널 이미지를 포함하는 복수의 프레임을 포함하는 비디오 데이터를 인코딩하는 방법으로서,

각각의 오리지널 이미지를 표현하는 데이터를 압축하여 각각의 오리지널 이미지의 압축된 버전을 생성하는 단계로서, 상기 압축은 각각의 오리지널 이미지를 적응적으로 블록 사이징함으로써 생성된 데이터를 이용하는, 단계;

각각의 오리지널 이미지의 각각의 압축된 버전을 양자화하여 각각의 오리지널 이미지의 손실 버전을 생성하는 단계;

각각의 압축된 버전의 압축해제된 이미지를 생성하기 위해 각각의 오리지널 이미지의 각각의 압축된 버전을 압축해제(decompress)하는 단계로서, 상기 압축해제는 각각의 오리지널 이미지를 적응적으로 블록 사이징함으로써 생성된 데이터를 이용하는, 단계;

각각의 오리지널 이미지와 각각의 압축해제된 이미지 사이의 차이를 결정하여, 각각의 오리지널 이미지와 관련된 리지듀얼 데이터를 생성하는 단계; 및

각각의 오리지널 이미지의 각각의 손실 버전 및 각각의 오리지널 이미지와 관련된 각각의 리지듀얼 데이터를 출력하는 단계를 포함하고,

각각의 오리지널 이미지의 각각의 손실 버전 및 각각의 오리지널 이미지와 관련된 각각의 리지듀얼 데이터는, 각각의 프레임의 이미지가 상기 복수의 프레임 내 각각의 프레임의 상기 오리지널 이미지와 실질적으로 동일한, 이미지를 생성하도록 이용될 수 있는, 신호의 무손실 압축 및 인코딩 방법.

#### 청구항 10

제 9 항에 있어서,

상기 압축은 인트라프레임 기반으로 이루어지는, 신호의 무손실 압축 및 인코딩 방법.

#### 청구항 11

제 9 항에 있어서,

상기 압축은 DCT (discrete cosine transform) 및 DQT (differential quadtree transform) 기술의 결합을 이용하는, 신호의 무손실 압축 및 인코딩 방법.

#### 청구항 12

삭제

#### 청구항 13

삭제

#### 청구항 14

삭제

#### 청구항 15

삭제

#### 청구항 16

삭제

#### 청구항 17

각각이 오리지널 이미지를 포함하는 복수의 프레임을 포함하는 비디오 데이터를 인코딩하는 장치로서,

각각의 오리지널 이미지를 표현하는 데이터를 압축하여 각각의 오리지널 이미지의 압축된 버전을 생성하는 수단으로서, 상기 압축은 각각의 오리지널 이미지를 적응적으로 블록 사이징함으로써 생성된 데이터를 이용하는, 수단;

각각의 오리지널 이미지의 각각의 압축된 버전을 양자화하여 각각의 오리지널 이미지의 손실 버전을 생성하는 수단;

각각의 압축된 버전의 압축해제된 이미지를 생성하기 위해 각각의 오리지널 이미지의 각각의 압축된 버전을 압축해제(decompress)하는 수단으로서, 상기 압축해제는 각각의 오리지널 이미지를 적응적으로 블록 사이징함으로써 생성된 데이터를 이용하는, 수단;

각각의 오리지널 이미지와 각각의 압축해제된 이미지 사이의 차이를 결정하여, 각각의 오리지널 이미지와 관련된 리지듀얼 데이터를 생성하는 수단; 및

각각의 오리지널 이미지의 각각의 손실 버전 및 각각의 오리지널 이미지와 관련된 각각의 리지듀얼 데이터를 출력하는 수단을 포함하고,

각각의 오리지널 이미지의 각각의 손실 버전 및 각각의 오리지널 이미지와 관련된 각각의 리지듀얼 데이터는, 각각의 프레임의 이미지가 상기 복수의 프레임 내 각각의 프레임의 상기 오리지널 이미지와 실질적으로 동일한, 이미지를 생성하도록 이용될 수 있는, 신호의 무손실 압축 및 인코딩 장치.

#### 청구항 18

제 17 항에 있어서,

상기 압축은 인트라프레임 기반으로 이루어지는, 신호의 무손실 압축 및 인코딩 장치.

#### 청구항 19

제 17 항에 있어서,

상기 압축 수단은 DCT (discrete cosine transform) 및 DQT (differential quadtree transform) 기술의 결합을 이용하는, 신호의 무손실 압축 및 인코딩 장치.

#### 청구항 20

삭제

#### 청구항 21

삭제

#### 청구항 22

삭제

명세서

[0001]

## 발명의 배경

[0002]

### I. 발명의 분야

[0003]

본 발명은 이미지 프로세싱 및 압축에 관련된 것이다. 보다 상세하게는, 본 발명은 주파수 도메인에서의 비디오 이미지 및 오디오 정보의 무손실 인코딩에 관한 것이다.

[0004]

### II. 관련된 기술의 설명

[0005]

디지털 화상 프로세싱은 디지털 신호 프로세싱의 일반적인 분야에서 중요한 위치를 차지한다. 인간의 시각적 인지력의 중요성은 디지털 화상 프로세싱 기술 분야 및 학문에 있어서 많은 관심과 진보를 촉진해왔다. 필름이나 영화를 투영하기 위해 사용되는 것들과 같은 비디오 신호들의 송신 및 수신 분야에서, 이미지 압축 기술에 다양한 향상이 있어 왔다. 현재 제안된 비디오 시스템들 중 많은 것들이 디지털 인코딩 기술을 사용한다. 이러한 분야의 태양은 이미지 코딩, 이미지 복원, 및 이미지 특징 선택을 포함한다. 이미지 코딩은 요구되는 밴드 폭을 최소화할 수 있도록 가능한 적은 비트들을 사용하여, 한편으로 동시에 일정한 제한내의 왜곡을 유지하면서 효율적인 방식으로 디지털 통신 채널의 화상을 송신하고자 하는 시도를 나타낸다. 이미지 복원은 대상의 실제 이미지를 회복하기 위한 노력을 나타낸다. 통신 채널을 통해 송신되는 코딩된 이미지는 다양한 팩터에 의해 왜곡될 수 있다. 열화의 소스는 대상으로부터 이미지를 형성할 때 최초로 발생할 수 있다. 특징 선택은 화상에 대한 임의의 특성 선택을 지칭한다. 그러한 특성들은 더 넓은 문맥에서는 인식, 분류, 및 선택이 요구되어야 한다.

[0006]

디지털 시네마와 같은 비디오의 디지털 인코딩은 개선된 이미지 압축 기술들에 의해 수혜받는 영역이다. 디지털 이미지 압축은 일반적으로 무손실과 손실 방법들의 두 개의 카테고리들로 구분될 수 있다. 무손실 이미지는 정보의 손실 없이 복구된다. 손실 방법은 압축 비율, 압축 알고리즘의 품질 및 알고리즘의 구현에 따른 회복 불가능한 정보의 손실을 수반한다. 일반적으로, 손실있는 압축 방식은 비용-효율적인 디지털 시네마 접근에 요구되는 압축 비율들을 획득하기 위해 고려된다. 디지털 시네마 품질 수준을 달성하기 위해 이러한 압축 방식은 시각적으로 무손실 성능 레벨을 제공하여야 한다. 압축 프로세스의 결과로서 정보의 수학적 손실이 있을 수는 있지만, 이러한 손실에 의해 야기되는 이미지 왜곡은 정상적인 관측 조건 하에서 관측자에게 인지되지 않아야 한다.

[0007]

기존의 디지털 이미지 압축 기술들은, 다른 응용들, 즉 텔레비전 시스템들을 위해 발달되어 왔다. 그러한 기술들은 의도하는 응용들에 적합한 설계 절충 들을 만들어 왔지만, 시네마 상에 요구되는 품질 조건을 만족시키지는 못했다.

[0008]

디지털 시네마 압축 기술은 영화 관람객들이 이전에 경험했던 시각적 품질을 제공해야 한다. 이상적으로, 디지털 시네마의 시각적 품질은 고품질, 방출 프린트 필름의 품질을 능가하고자 하는 시도를 해야한다. 동시에, 압축 기술은 실질적으로 높은 코딩 효율을 가져야 한다. 여기에서 정의된 바와 같이, 코딩 효율은 임의의 품질 수준을 만족하기 위한 압축된 이미지 품질에 요구되는 비트 레이트를 지칭한다. 게다가, 시스템과 코딩기술은 다른 포맷들을 조정하기 위한 빌트-인 유연성(built-in flexibility)을 가져야 하며, 비용 효율적이어야 한다; 즉, 작은 사이즈의 효율적인 디코더 또는 인코더 프로세스이다.

[0009]

사용 가능한 많은 압축 기술들은 엄청난 수준의 압축을 제공하지만, 비디오 신호 품질의 열화를 가져온다. 통상적으로, 압축된 정보를 전달하기 위한 기술들은 압축된 기술들이 일정 비트 레이트로 전달될 것이 필요하다

[0010]

비디오 신호들에 대해 목적하는 수준의 품질을 보증하면서 엄청난 수준의 압축을 제공할 수 있는 한가지 압축 기술은 인코딩된 이산 코사인 변환(DCT) 계수 데이터에 적응적인 사이즈의 블록들과 서브-블록들을 활용한다. 이하에서 이 기술을 ABSDCT(적응적 블록 사이즈 이산 코사인 변환, adaptive block size discrete cosine transform)이라고 한다. 이 기술은 미국특허 제 5,021,891 호인 "Adaptive Block Size Image Compression Method And System"에 개시되어 있으며, 본 발명의 양수인에게 양도되어 여기에서 함께 참조된다. 또한, DCT는 미국특허 제 5,107,345 호인 "Adaptive Block Size Image Compression Method And System"에 개시되어 있으며, 본 발명의 양수인에게 양도되어 여기에서 함께 참조된다. 또한, ABSDCT 기술과 DQT(Differential Quadtree Transform) 기술을 함께 사용하는 것은 미국특허 제 5,452,104 호인 "Adaptive

Block Size Image Compression Method And System"에 개시되어 있으며, 본 발명의 양수인에게 양도되어 여기에서 함께 참조된다. 이러한 특허들에 개시된 시스템들은 "인트라-프레임 (intra-frame)" 인코딩이라고 지칭된 것을 활용하는데, 거기서 이미지 데이터의 각각의 프레임은 다른 프레임의 내용과 관계없이 인코딩된다.

ABSDCT 기술을 사용하여, 달성 가능한 데이터 비율은 이미지 품질의 식별가능한 열화 없이, 대략 초당 15억 비트에서 대략 초당 5천만 비트로 감소될 것이다.

[0011] ABSDCT 기술은 흑백 또는 컬러 이미지 또는 이미지를 나타내는 신호를 압축하는데 사용할 수 있다. 컬러 입력 신호는 YIQ 포맷일 수 있는데, 여기서, 각 4:4:4 또는 대안 포맷에 대해, Y는 휘도 또는 밝기 샘플이고 I와 Q는 색상 또는 컬러 샘플들이다. YUV, YCbCr 또는 RGB 포맷 같은 다른 알려진 포맷들이 또한 사용될 수 있다. 눈의 컬러에 대한 낮은 공간 감도 때문에 대부분의 연구는 수평 및 수직 방향으로 4 라는 팩터에 의한 컬러 성분의 서브-샘플이 바람직하다는 것을 보여주고 있다. 따라서, 비디오 신호는 4 개의 휘도 샘플들과 2 개의 색상 성샘플들에 의해 표현될 수 있다.

[0012] ABSDCT 를 사용하여 비디오 신호는 프로세싱을 위해서 일반적으로 픽셀들의 블록들로 분할될 것이다. 각각의 블록에 대해 휘도와 색상 성분들은 블록 사이즈 할당 요소 또는 블록 인터리버를 통과해야 한다. 예를 들어, 16 x 16 (픽셀) 블록은 블록 인터리버에 제시되어야 하는데, 블록 인터리버는 각각의 16 x 16 블록 내의 이미지 샘플들을 정렬하거나 조직화하여 DCT 분석을 위한 데이터의 블록과 서브-블록들의 결합을 생성한다. DCT 오퍼레이터는 시간과 공간 샘플화된 신호를 동일한 신호의 주파수 표현으로 전환하는 한가지 방법이다. 주파수 표현으로 변환함으로써, DCT 기술들은 높은 수준의 압축을 허용하고 있음을 보여주는데, 이는 양자화가 이미지에 주파수 분산 특성들을 이용할 수 있도록 설계될 수 있기 때문이다. 바람직한 실시형태에서 하나의 16 x 16 DCT 는 제 1 배열에 적용되고, 4 개의 8 x 8 DCT 는 제 2 배열에 적용되고, 16 개의 4 x 4 DCT 는 제 3 배열에 적용되며, 64 개의 2 x 2 DCT 는 제 4 배열에 적용된다.

[0013] DCT 동작은 비디오 소스 내에 내재하는 공간 중복성을 줄인다. DCT 가 수행되고 난 후 대부분의 비디오 신호 에너지는 몇 개의 DCT 계수 내에 집중하는 경향이 있다. 부가적인 변환인, DQT (Differential Quad-Tree Transform) 는 DCT 계수들간의 중복성을 줄이기 위해 사용될 수 있다.

[0014] 16x16 블록과 각각의 서브-블록에 대해, 블록 또는 서브 블록을 인코딩하기 위해 요구되는 비트들의 수를 판단하기 위해 DCT 계수 값들과 DQT 값 (DQT가 사용되었다면) 이 분석된다. 그리고 나서, 인코딩할 최소한의 수의 비트들을 요구하는 블록 또는 서브 블록들의 조합이 이미지 분할을 나타내기 위해 선택된다. 예를 들어, 2 개의 8x8 서브 블록들, 6 개의 4x4 서브 블록들, 및 8개의 2x2 서브 블록들이 이미지 분할을 나타내기 위해 선택될 수 있다.

[0015] 그 후, 선택된 블록 또는 서브 블록들의 조합은 순서대로 16x16 블록에 적당하게 배열된다. 그 후, DCT/DQT 계수 값들은 송신을 준비하기 위한 주파수 가중화, 양자화, 및 코딩 (예를 들어, 가변적인 길이 코딩)을 경험한다. 상기에서 기술한 ABSDCT 기술을 놀라울 정도로 잘 수행한다고 하더라도 그것은 계산적으로 집약적 (computationally intensive) 이다.

[0016] 또한, ABSDCT의 사용으로 시각적 손실이 없더라도, 때때로, 인코딩된 것과 똑같은 방식으로 데이터를 복구하는 것이 바람직하다. 예를 들어, 마스터링과 기록을 위해서는 본래의 도메인 (native domain) 내에 그것을 정확하게 복구할 수 있는 방식으로 데이터들을 압축할 것이 요구된다.

[0017] 통상, 이미지에 대한 무손실 압축 시스템은, 인코딩될 현재 픽셀 값을 추정하는 프리딕터 (predictor) 로 이루어진다. 리지듀얼 픽셀(residual pixel)은 실제 픽셀 및 예측되는 픽셀 간의 차이로 획득된다. 다음으로, 리지듀얼 픽셀은 엔트로피 인코딩되고 저장 또는 전송된다. 예측은 픽셀 상관성을 제거하기 때문에, 리지듀얼 픽셀들은 특징적인 양측 (라플라시안) 분포를 갖는 감소된 동적 범위를 갖는다. 이제, 압축된다. 리지듀얼의 압축량은 예측 및 이후의 엔트로피 인코딩 방법에 의존한다. 가장 일반적으로 이용되는 예측 방법은 DPCM (differential pulse code modulation), 및 ADPCM (adaptive differential pulse code modulation) 등의 그 변형예이다.

[0018] 화소-기반(pixel-based) 예측의 문제점은 리지듀얼이 여전히 높은 에너지를 갖는다는 것이다. 이는 작은 수의 이웃하는 픽셀만이 예측 프로세스에 이용된다는 사실에서 기인되는 것이다. 따라서, 화소-기반 예측 방식의 코딩 효율을 향상시킬 필요가 있다.



[0019]

## 발명의 요약

[0020]

본 발명의 실시예는 압축을 달성하는데 있어 무손실 방식으로 디지털 이미지 및 비디오 데이터를 인코딩하기 위한 시스템을 설명한다. 이 시스템은 하이브리드(hybrid)이며, 이는 손실 방식으로 데이터를 압축하는 일부분을 갖는 것 및 무손실 방식으로 리지듀얼 데이터를 압축하는 일부분을 갖는 것을 의미한다. 손실부에서, 이 시스템은 ABSDCT (adaptive block size discrete cosine transform) 알고리즘을 이용한다. ABSDCT 시스템은 고 시각적 품질 및 압축 비로 상기 데이터를 압축한다. ABSDCT 시스템으로부터 오리지널 및 압축해제(decompress)된 데이터 간의 차이로서 리지듀얼 이미지가 획득된다. 이 리지듀얼은 골롬-라이스(Golomb-Rice) 코딩 알고리즘을 이용하여 무손실 인코딩된다. DCT 계수의 양자화 및 시각 기반 적응적 블록 사이즈로 인하여, 리지듀얼은 매우 낮은 에너지를 가지므로, 따라서, 우수한 전체 무손실 압축비를 나타낸다.

[0021]

ABSDCT 시스템은 시네마 품질에서 높은 압축률을 달성한다. 블록-기반 이기 때문에, 픽셀 상관성이 어떠한 화소-기반 방식에서보다도 매우 많이 제거된다. 따라서, 이는 여기에서 설명될 무손실 시스템에서 프리딕터로서 이용된다. 프리딕터와 관련하여, 무손실 인코딩 시스템이 추가되어 하이브리드 무손실 압축 시스템을 형성한다. 이 시스템은 동화상뿐만 아니라 정지 화상을 압축할 수 있다. 정지 화상인 경우, ABSDCT 압축된 데이터와 엔트로피 인코딩된 리지듀얼 데이터는 압축된 출력으로 이용된다. 모션 시퀀스에서, 인트라-프레임 또는 인터-프레임 압축을 사용할지가 결정된다. 예를 들면,  $f(t)$  는  $t$  순간에서의 이미지 프레임을,  $F(t)$  및  $F(t+\Delta t)$  는  $t+\Delta t$  순간에서의 이미지 프레임의 DCT를 각각 나타낸다.  $\Delta t$ 는 2 개의 연속적인 프레임 사이의 시간 간격에 대응한다.

[0022]

본 발명은 데이터가 인코딩되었던 방식과 정확히 동일하게 데이터를 복원할 수 있도록 데이터를 압축하는 장치 및 방법으로 구현된다. 실시예는 인트라프레임 코딩, 인터프레임 코딩, 또는 두가지의 하이브리드를 수행하는 시스템을 포함한다. 이 시스템은 DCT 계수 데이터의 적응 사이즈 블록 및 서브 블록을 이용하는 품질-기반 시스템이다. 픽셀 데이터의 블록은 인코더에 입력된다. 이 인코더는 프로세싱을 위한 픽셀의 입력 블록을 분할하는 BSA(block size assignment) 요소를 포함한다. 블록 사이즈 할당은 입력 블록 및 추가적으로 세분화된 분산에 기초한다. 일반적으로, 블록 및 서브 블록 평균 값이 상이한 소정의 범위에 드는 경우에, 더 큰 분산을 갖는 영역은 더욱 작은 블록으로 세분되고, 더욱 적은 분산을 갖는 영역은 세분되지 않게 된다. 따라서, 먼저, 블록의 분산 임계값은 평균값에 따라 공칭값으로부터 변화되고, 다음으로, 블록의 분산이 이 임계값과 비교되며, 분산이 임계값보다 큰 경우 블록은 세분된다.

[0023]

블록 사이즈 할당은 변환 요소에 제공되는데, 그것은 픽셀 데이터를 주파수 도메인 데이터로 변형한다. 변환은 블록 사이즈 할당을 통해 선택된 블록과 서브 블록들에 대해서만 수행된다. AC 성분들을 위해서 변환 데이터는 그 후 양자화와 직렬 변환(serialization)을 통한 스케일링을 경험한다. 변환 데이터의 양자화는 콘트라스트, 계수 카운트, 레이트 왜곡, 블록 사이즈 할당의 밀도, 및/또는 과거 스케일 팩터들에 대해 조정되는 스케일 팩터와 같은 이미지 품질 메트릭에 기초하여 양자화된다. 지그-재그 스케일링과 같은 직렬 변환은 같은 값의 가능한 긴 런 렉스(run length)를 생성하는 것에 기초한다. 데이터의 스트림은 그 후 송신을 준비하기 위한 가변적인 길이 코더에 의해 코딩된다. 코딩은 허프만 코딩이거나, 또는 코딩은 골롬-라이스(GR) 인코딩같은 지수 분포에 기초할 수 있다.

[0024]

ABSDCT 와 같은 하이브리드 압축 시스템의 이용은 픽셀 또는 DCT 값의 우수한 프리딕터로서 동작한다. 따라서, 화소-기반 예측을 이용하는 시스템보다 더욱 높은 무손실 압축률을 발생한다. 손실 부분은 디지털 시네마 품질 결과, 즉, 시각적으로 무손실되는 파일에서의 압축 결과를 제공한다. 무손실 부분에 대해, 허프만 코드와 다르게, GR 코딩은 어떠한 사전(priori) 코드 생성을 요구하지 않는다. 즉, 허프만 코딩으로 저장될 광범위의 코드북을 필요로 하지 않는다. 이는 칩 자원(chip real estate)의 효과적인 이용을 유발한다. 따라서, 칩 사이즈는 하드웨어 구현 시 감소된다. 또한, GR 인코딩은 허프만 코딩보다 구현하는 것이 더욱 간단하다. 또한, DCT 계수 또는 리지듀얼이 지수 분포 성질을 가짐에 따라 GR 코딩은 허프만 코딩보다 더욱 높은 코딩 효율을 달성한다. 또한, 압축 시스템의 손실부가 블록 세분 시 시각적으로 현저한 정보를 이용하기 때문에, 콘텍스트 모델링(context modeling)이 리지듀얼 인코딩에 내재된다. 이는 리지듀얼 인코딩에 대해 콘텍스트 데이터를 축적하기 위해 추가의 저장 레지스터가 필요하지 않다는 점에서 중요하다. 또한, 모션 추정을 이용하지 않기 때문에, 이 시스템은 구현하는 것이 매우 간단하다.

[0025]

이미지 정보를 표현하는 신호를 무손실 압축 및 인코딩하기 위한 장치 및 방법을 청구한다. 이미지 정보를



표현하는 신호가 압축되어 이미지의 압축된 버전을 형성한다. 이미지의 압축된 버전은 양자화됨으로써 이미지의 손실 버전을 형성한다. 또한, 이미지의 압축된 버전은 직렬 변환되어 양자화된 이미지의 직렬 변환된 양자화 버전이 생성된다. 다음으로, 이 버전이 복원되고, 오리지널 이미지와 복원된 버전 사이의 차이를 결정함으로써, 이미지의 리지듀얼 버전이 생성된다. 이미지의 손실 버전 및 이미지의 리지듀얼 버전은 각각 출력되거나 결합되며, 여기서, 이미지의 복원된 손실 버전과 이미지의 리지듀얼 버전의 결합은 오리지널 이미지와 실질적으로 유사하다.

[0026] 이미지 정보를 표현하는 신호를 무손실 압축 및 인코딩하는 방법이 청구된다. 손실 압축된 데이터 파일 및 리지듀얼 압축된 데이터 파일이 생성된다. 손실 압축 데이터 파일 및 리지듀얼 압축된 데이터 파일이 결합될 때, 오리지널 데이터 파일과 실질적으로 동일한 무손실 데이터 파일이 생성된다.

[0027] 따라서, 효율적으로 무손실 압축을 수행하는 장치와 방법을 제공하는 것이 실시형태의 일 태양이다.

[0028] 본 실시형태의 다른 태양은 마스터링과 기록보관을 위해 이용할 수 있는 방식으로 손실없이 디지털 이미지와 오디오 정보를 압축하는데 있다.

[0029] 본 실시예의 또 다른 태양은 인터프레임 기반 상에서 무손실 압축 시스템을 제공하는데 있다.

[0030] 본 실시예의 또 다른 태양은 인트라프레임 기반 상에서 무손실 압축 시스템을 제공하는데 있다.

[0031] **도면의 간단한 설명**

[0032] 이하, 도면을 참조하여 본 발명의 특징 및 잇점들을 보다 상세하게 설명하며, 도면 중 동일부호는 명세서 전체를 통해 동일부재를 나타낸다.

[0033] 도 1 은 이미지 압축 및 프로세싱 시스템의 인코더 부분에 대한 블록도이다.

[0034] 도 2 는 이미지 압축 및 프로세싱 시스템의 디코더 부분에 대한 블록도이다.

[0035] 도 3 은 분산 (variance) 기반의 BSA 와 관련된 프로세싱 단계들을 나타내는 흐름도이다.

[0036] 도 4a 는 DCT 계수 매트릭스에서 제로 런 령스의 Y 성분의 지수 분포를 나타내는 도면이다.

[0037] 도 4b 는 DCT 계수 매트릭스에서 제로 런 령스의  $C_b$  성분의 지수 분포를 나타내는 도면이다.

[0038] 도 4c 는 DCT 계수 매트릭스에서 제로 런 령스의  $C_r$  성분의 지수 분포를 나타내는 도면이다.

[0039] 도 5a 는 DCT 계수 매트릭스에서 Y 성분의 진폭 사이즈 또는 Y 성분의 진폭 사이즈의 지수 분포를 나타낸다.

[0040] 도 5b 는 DCT 계수 매트릭스에서  $C_b$  성분의 진폭 사이즈 또는  $C_b$  성분의 진폭 사이즈의 지수 분포를 나타낸다.

[0041] 도 5c 는 DCT 계수 매트릭스에서  $C_r$  성분의 진폭 사이즈 또는  $C_r$  성분의 진폭 사이즈의 지수 분포를 나타낸다.

[0042] 도 6 은 곱셈-라이트 인코딩 프로세스를 나타내는 도면이다.

[0043] 도 7 은 곱셈-라이트 인코딩을 위한 장치를 나타내는 도면이다.

[0044] 도 8 은 DC 성분값들을 인코딩하는 프로세스를 나타내는 도면이다.

[0045] 도 9 는 무손실 압축을 위한 장치를 나타내는 도면이다.

[0046] 도 10 은 하이브리드 무손실 압축의 방법을 나타내는 도면이다.

[0047] **바람직한 실시형태들에 대한 상세한 설명**

[0048] 디지털 신호들의 디지털 전송을 편리하게 하고 그에 따른 이익들을 누리기 위해서는, 일반적으로 어떤 형태의 신호 압축을 이용할 필요가 있다. 결과 이미지에 높은 압축을 실현하기 위해서는, 높은 이미지 품질을 유지하는 것 또한 중요하다. 또한, 많은 애플리케이션에서 중요하게 생각하는 콤팩트한 하드웨어 구현을 위해 효율적인 계산이 바람직하다.

[0049] 본 발명의 일 실시형태에 대해 상세하게 설명하기 전에, 도면에 도시되거나 또는 다음의 상세한 설명에서 설명

될 컴포넌트들의 배열이나 구성의 세목들에 본 발명의 응용이 제한되지 않는다는 점이 이해되어야 할 것이다.

본 발명은 다른 실시형태들이 가능하며, 다양한 방법으로 실시될 수 있다. 또한, 여기에서 사용되는 어구와 전문용어는 설명을 목적으로 하는 것이지 제한으로 간주되어서는 안된다는 것도 이해되어야 할 것이다.

[0050]

일 실시형태의 일 태양에 이용되는 이미지 압축은, 본 출원의 양도인에게 양도되었으며 여기에서 함께 참조되는, 1999년 11월 8일 출원되고 함께 계류중인 미국 특허출원 제 09/436,085 호인 "Contrast Sensitive Variance Based Adaptive Block Size DCT Image Compression" 에 개시되어 있는 DCT (DCT) 기술에 기초한다.

DCT 를 이용하는 이미지 압축 및 복원 시스템은, 본 출원의 양도인에게 양도되고 여기에서 함께 참조되는, 2000년 1월 28일 출원되어 함께 계류중인 미국 특허출원 제 09/494,192 호 "Quality Based Image Compression" 에 개시되어 있다. 일반적으로, 디지털 도메인에서 프로세싱되는 이미지는  $N \times N$  사이즈의 오버랩되지 않는 어레이로 분할된 화소 데이터로 구성된다. 2 차원 DCT는 각각의 블록에서 수행될 수 있다. 2 차원 DCT 는 다음의 관계식,

$$X(k,l) = \frac{\alpha(k)\beta(l)}{\sqrt{N \cdot M}} \sum_{m=0}^{N-1} \sum_{n=0}^{M-1} x(m,n) \cos\left[\frac{(2m+1)\pi k}{2N}\right] \cos\left[\frac{(2n+1)\pi l}{2M}\right], 0 \leq k, l \leq N-1$$

[0051]

으로 정의되고,

$$\alpha(k), \beta(l) = \begin{cases} 1, & \text{if } k=0 \\ \sqrt{2}, & \text{if } k \neq 0 \end{cases}$$

[0052]

여기서,  $x(m,n)$  은  $N \times M$  블록 내에서  $(m,n)$  위치에서의 픽셀이며, 그리고  $X(k,l)$  은 대응하는 DCT 계수이다.

[0053]

픽셀 값은 음이 아니기 때문에, DCT 성분  $X(0,0)$  은 항상 양이고 보통 대부분의 에너지를 갖는다. 사실, 통상적인 이미지들에 있어서, 변환 에너지의 대부분은 성분  $X(0,0)$  주위에 집중되어 있다. 이러한 에너지 집중 특성은 DCT 기술을 매우 유용한 압축 방법이 되게 한다.

[0054]

이미지 압축 기술은 콘트라스트 (contrast) 적응형 코딩을 이용하여 더 높은 비트 레이트 감소를 얻을 수 있다. 가장 자연스러운 이미지들은 상대적으로 천천히 변하는 평평한 영역들 및 물체의 경계들과 하이-콘트라스트 텍스처와 같은 비지 영역 (busy area) 으로 이루어지는 것을 알 수 있다. 콘트라스트 적응형 코딩 방식은 비지 영역들에 더 많은 비트를 할당하고 약한 비지 영역들에 더 적은 비트들을 할당하는 것에 의해 이러한 팩터를 이용한다.

[0055]

콘트라스트 적응형 방법들은 인터프레임 코딩 (공간-시간 프로세싱) 을 대신하여 인트라프레임 (intraframe) 코딩 (공간 프로세싱) 을 이용한다. 인터프레임 코딩은 본래부터 더 복잡한 프로세싱 회로들에 더하여 다중 프레임 버퍼들을 필요로 한다. 많은 응용들에서, 실제적인 구현을 위해서는 감소된 복잡도를 필요로 한다. 인트라프레임 코딩은 또한 공간-시간 코딩 이론이 동작하지 않고 잘 실행되지 않는 상황에서 유용하다. 예를 들어, 초당 24 프레임의 영화들은 기계적인 셔터 때문에 통합 (integration) 시간이 상대적으로 짧기 때문에 이러한 카테고리에 해당될 수 있다. 짧은 통합 시간은 더 높은 정도의 시간 에일리어징 (temporal aliasing) 을 허용한다. 프레임 대 프레임 상호상관의 가정은 갑자기 움직일수록 빠른 움직임에 대해 동작을 하지 않는다. 인트라프레임 코딩은 또한 50 Hz 및 60 Hz 전력선 주파수가 포함되는 경우에 표준화하기에 더 쉽게 된다. 텔레비전은 현재 50 Hz 또는 60 Hz 에서 신호를 전송한다. 디지털 접근에서 인트라프레임 방식의 이용은 프레임 레이트 대 공간 해상도 (spatial resolution) 의 트레이드 오프를 이용하여 50 Hz 및 60 Hz 동작 또는 초당 24 프레임의 영화에서조차 모두에 적용될 수 있다.

[0056]

이미지 프로세싱의 목적을 위해, DCT 동작은 오버랩되지 않는 블록들의 어레이로 나뉘어진 픽셀 데이터 상에서 수행된다. 비록 여기에서는 블록사이즈가  $N \times N$  사이즈로 논의되지만, 다양한 블록 사이즈들이 사용될 수 있도록 계획될 수 있다. 예를 들어,  $N \times M$  블록 사이즈가 이용될 수도 있는데, 여기서,  $N$  과  $M$  은 정수이고  $M$  이  $N$  보다 더 크거나 또는 더 작을 수 있다. 다른 중요한 태양은, 블록이  $i$  와  $j$  가 정수들일 때  $N/i \times N/i$ ,  $N/i \times N/j$ ,  $N/i \times M/j$  등과 같은 적어도 하나의 레벨의 서브-블록들로 나뉠 수 있는 것이다. 또한, 여기에서 논의되는 것과 같은 예시적인 블록 사이즈는 DCT 계수의 대응하는 블록과 서브-블록을 갖는  $16 \times 16$  픽셀 블록이다. 또한, 예를 들어  $9 \times 9$  와 같이 둘 모두 짝수 또는 홀수인 다양한 다른 정수들이 사용될 수도 있도록 계획될 수 있다.

[0057] 도 1 및 도 2 는 구현 가능한 직렬 변환기 (serializer) 개념이 통합된 이미지 프로세싱 시스템 (100) 을 나타낸다. 이미지 프로세싱 시스템 (100) 은 수신된 비디오 신호를 압축하는 인코더 (104) 를 포함한다. 압축된 신호는 전송 채널 또는 물리적 매체 (108)를 사용하여 송신되어, 디코더 (112) 에 의해 수신된다. 디코더 (112) 는 수신된 인코딩된 데이터를, 이후 표시될 수도 있는 이미지 샘플들로 디코딩한다.

[0058] 일반적으로, 이미지는 프로세싱을 위해 픽셀들의 블록들로 나뉜다. 컬러 신호는 RGB-YC<sub>1</sub>C<sub>2</sub> 변환기 (116) 에 의해 RGB 공간으로부터 YC<sub>1</sub>C<sub>2</sub> 공간으로 변환될 수도 있으며, 여기서 Y 는 휘도 또는 밝기 성분, 그리고 C<sub>1</sub> 및 C<sub>2</sub> 는 색상 또는 컬러 성분이다. 컬러에 대한 눈의 낮은 공간 감도 때문에 많은 시스템들은 수직 및 수평 방향으로 4 의 팩터로 C<sub>1</sub> 및 C<sub>2</sub> 성분들을 서브-샘플링한다. 그러나, 서브-샘플링은 필수적인 것은 아니다. 4:4:4 포맷으로 알려진 전체 해상도 이미지는 "디지털 시네마"를 포함하는 것들과 같은 몇몇의 응용들에서 매우 유용하거나 필요할 수 있다. 두가지 가능한 YC<sub>1</sub>C<sub>2</sub> 표현들은 YIQ 표현과 YUV 표현으로, 둘 모두는 본 기술분야에 잘 알려져 있다. 또한 YCbCr로 알려진 다양한 YUV 표현이 이용될 수도 있다. 또한, 이것은 홀수와 짝수 성분으로 분리될 수도 있다. 따라서, 실시 형태에서 표현 Y-짝수, Y-홀수, Cb-짝수, Cb-홀수, Cr-짝수, Cr-홀수가 사용된다.

[0059] 바람직한 실시형태에서, 각각의 홀수 및 짝수 Y, Cb, 그리고 Cr 성분들은 서브-샘플링 없이 프로세싱된다. 그러므로, 픽셀들의 16x16 블록의 6개의 성분들은 각각은 인코더 (104) 로 제공된다. 예시의 목적으로, Y-짝수 성분을 위한 인코더 (104) 가 도시되어 있다. 유사하게, Y-홀수 성분, 그리고 짝수 및 홀수 Cb 및 Cr 성분들을 위해 인코더들이 사용된다. 인코더 (104) 는 비디오 압축을 위한 준비에서 블록 사이즈 할당을 수행하는 블록 사이즈 할당 요소 (120)를 포함한다. 블록 사이즈 할당 요소 (120) 는 블록 내의 이미지의 지각 특성 (perceptual characteristics) 에 기초하여 16x16 블록의 블록 분해를 결정한다. 블록 사이즈 할당은 16x16 블록 내에서의 활동에 의존하는 쿼드-트리 (quad-tree) 내에서, 각각의 16x16 블록을 8x8, 4x4, 그리고 2x2와 같이 더 작은 블록들로 다시 나눈다. 블록 사이즈 할당 요소 (120) 는 그 길이가 1 과 21 비트 사이일 수 있는 PQR 데이터로 불리는 쿼드-트리 데이터를 생성한다. 그러므로, 블록 사이즈 할당이 16x16 블록이 나뉘어지는 것으로 결정한다면, PQR 데이터의 R 비트가 설정되고, 4개로 나뉜 8x8 블록들에 대응하는 Q 데이터의 4개의 추가적인 비트들이 이어진다. 블록 사이즈 할당이 8x8 블록들 중 어떤 하나가 다시 나뉘어지는 것으로 결정한다면, 그 후 각각의 8x8 블록이 다시 나뉘어지도록 하기 위해서 P 데이터의 4 개의 추가적인 비트들이 추가된다.

[0060] 도 3 을 참조하면, 블록 사이즈 할당 요소 (120) 의 구체적인 동작을 보여주는 플로우 다이어그램이 제공된다. 블록의 분산이 블록을 다시 나누는 결정에 있어서 메트릭으로서 사용된다. 단계 (202)에서 시작하면, 픽셀의 16x16 블록이 판독된다. 단계 (204)에서, 16x16 블록의 분산  $\sigma^2$  이 계산된다. 분산은 다음 식에 의해 계산된다:

$$\text{var} = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} x_{i,j}^2 - \left( \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} x_{i,j} \right)^2$$

[0061] 으로 계산되며,

[0062] 여기서, N=16, 그리고  $x_{i,j}$ 는 NxN 블록 내에서 i 번째 행, j 번째 열의 화소값이다. 단계 206 에서, 먼저 블록의 평균값이 2개의 소정의 값들 사이인 경우에 분산 임계값 (T16) 이 새로운 임계값 (T'16) 을 제공하도록 변경하고, 그 후 블록 분산을 새로운 임계값, T'16 에 대해서 비교한다.

[0063] 만약 분산 ( $\sigma^2$ ) 이 임계값 (T16) 보다 크지 않으면, 단계 208 에서, 16x16 블록의 시작 어드레스를 임시 저장 장치에 기록하고, 16x 16 블록이 세분되지 않는 것을 지시하기 위해 PQR 데이터의 R 비트를 0 으로 설정한다. 그 다음, 알고리즘은 화소의 후속되는 16x16 블록을 판독한다. 단계 210 에서 분산 ( $\sigma^2$ ) 이 임계값 (T16) 보다 크면, 16x16 블록이 4개의 8x8 블록들로 세분되는 것을 알리기 위해 PQR 데이터의 R 비트를 1 로 설정된다.

[0064] 단계 212 에 도시된 바와 같이, 4개의 8x8 블록 (i=1:4) 에 대해 더이상의 세분을 순차적으로 고려한다. 단계 214 에서, 각각의 8x8 블록들에 대해 분산 ( $\sigma^2_i$ ) 가 계산된다. 단계 216 에서, 블록의 평균값이 두 개의 소정의 값들 사이에 있다면, 먼저 분산 임계값 (T8) 을 새로운 임계값 (T'8) 을 제공하도록 변경하고, 그 다음, 블록 분산을 이러한 새로운 임계값과 비교한다.

- [0065] 분산 ( $v_{8i}$ ) 가 임계값 (T8) 보다 크지 않다면, 단계 218 에서, 8x8 블록의 시작 어드레스는 임시 저장 장치로 기록되고, 대응하는 Q 비트인  $Q_i$  는 0 으로 설정된다. 그 다음, 후속의 8x8 블록이 프로세싱된다. 만약 분산 ( $v_{8i}$ ) 이 임계값 (T8) 보다 크다면, 단계 220 에서, 8x8 블록이 4 개의 4x4 블록들로 세분될 것을 지시하기 위해 대응하는 Q 비트인  $Q_i$  는 1 로 설정된다.
- [0066] 단계 222 에 도시된 바와 같이, 4 개의 4x4 블록 ( $j_i=1:4$ ) 에 대한 추가적인 세분을 순차적으로 고려한다. 단계 224 에서, 각각의 4x4 블록에 대해서, 분산  $v_{4ij}$  를 계산한다. 단계 226 에서, 만약 블록의 평균값이 두 개의 소정의 값들 사이이면, 먼저 분산 임계값 (T4) 을 새로운 분산 (T'4) 를 제공하도록 변경하고, 그 후 블록 분산을 새로운 임계값과 비교한다.
- [0067] 만약 분산  $v_{4ij}$  가 임계값 T4 보다 크지 않다면, 단계 (228)에서 4x4 블록의 어드레스가 기록되고, 그리고 대응하는 P 비트인  $P_{ij}$  가 0으로 설정된다. 그 후, 다음 4x4 블록이 프로세싱된다. 만약 분산  $v_{4ij}$  가 임계값 T4보다 크다면, 그 후 단계 (230)에서 대응하는 P 비트인  $P_{ij}$  는 4x4 블록이 4개의 2x2블록들로 다시 나뉘는 것을 알리기 위해 1로 설정된다. 또한, 4개의 2x2 블록들의 어드레스가 임시 저장장소에 기록된다.
- [0068] 임계값 T16, T8, 및 T4는 소정의 상수들일 수도 있다. 이것은 하드 디지전 (hard decision) 으로 알려져 있다. 다른 방법으로, 적응적 또는 소프트 디지전이 구현될 수도있다. 예를 들어, 소프트 디지전은 N이 8, 4, 2일 수 있을 때  $2N \times 2N$  블록들의 평균 픽셀 값에 의존하여 분산에 대한 임계값이 변한다. 그러므로, 평균 픽셀값의 함수들은 임계값으로 사용될 수도있다.
- [0069] 예시의 목적으로, 다음의 예를 고려한다. Y 성분에 대한 소정의 분산 임계값은 16x16, 8x8, 및 4x4 블록들에 대해 각각 50, 1100, 및 880으로 한다. 즉, T16=50, T8=1100, 및 T4=880이다. 평균값의 범위는 80 과 100이다. 16x16 블록에 대해 계산된 분산이 60이라고 가정하자. 60은 T16보다 크고, 평균값 90은 80 과 100 사이에 있기 때문에 16x16 블록은 4개의 8x8 블록들로 다시 나누어진다. 8x8 블록들에 대한 계산된 분산이 1180, 935, 980, 1210 이라고 가정하자. 8x8 블록들 중 2개는 T8을 초과하기 때문에, 이 두 개의 블록들은 더 나뉘어져서 전체적으로 8개의 4x4 서브-블록들을 만든다. 마지막으로, 8개의 4x4 블록들의 분산이 620, 630, 670, 610, 590, 525, 930, 및 690이고 대응하는 평균값이 90, 120, 110, 115라고 가정하자. 첫 번째 4x4 블록의 평균값이 (80, 100) 범위에 있기 때문에, 그의 임계값은 880보다 낮은 T'4=200으로 낮아지게 될 것이다. 그러므로, 이 4x4 블록 7번째 4x4 블록과 마찬가지로 다시 나뉘게된다.
- [0070] 휘도 성분  $Y_{\text{홀수}}$ 와 컬러 성분들,  $C_b$  및  $C_r$  에 대한 블록 사이즈 할당에 유사한 절차가 사용되었음을 주목한다. 컬러 성분들은 수평적으로, 수직적으로, 또는 양쪽 모두로 데시메이션 될 수도 있다.
- [0071] 또한, BSA 가 가장 큰 블록 (본 실시예에서는 16x16) 을 먼저 평가하는 탑 다운 접근 (top down approach) 으로 설명되었지만, 대신에 바텀 업 접근 (bottom up approach) 이 사용될 수도 있다. 바텀 업 접근은 가장 작은 적응 블록들 (본 실시예에서는 2x2) 을 먼저 평가할 것이다.
- [0072] 도 1을 다시 참조하면, 선택된 블록들에 대한 어드레스들과 함께 PQR 데이터가 DCT 요소 (124) 에 제공된다. DCT 요소 (124) 는 선택된 블록들에 대해 적당한 사이즈들의 이산 코사인 변환들을 수행하기 위해 PQR 데이터를 사용한다. 단지 선택된 블록들만이 DCT 프로세싱을 경험해야 한다.
- [0073] 이미지 프로세싱 시스템 (100) 은 또한 DCT 들의 DC 계수들간의 중복성을 줄이기 위한 DQT 요소 (128)를 포함한다. DC 계수는 각각의 DCT 블록의 좌상부 코너에서 발견된다. DC 계수들은 일반적으로 AC 계수들과 크게 비교된다. 사이즈들 간의 불일치는 효율적인 가변길이 코더의 설계를 어렵게한다. 따라서, DC 계수들 간의 중복성을 줄이는 것이 유리하다.
- [0074] DQT 요소 (128) 은 한번에 2x2를 택해, DC 계수들 상에 2-D DCT 들을 수행한다. 4x4 블록들 내에 2x2 블록들로 시작하여, 하나의 2-D DCT가 4개의 DC 계수들 상에서 수행된다. 이러한 2x2 DCT는 4개의 DC 계수들의 차동 쿼드-트리 변환 또는 DQT라고 불린다. 다음으로, 8x8 블록 내의 세 개의 인접한 DC 계수들과 함께 DQT 의 DC 계수가 다음 레벨의 DQT를 계산하기 위해 사용된다. 최종적으로, 16x16 블록 내의 4개의 8x8 블록들의 DC 계수들이 DQT를 계산하기 위해 사용된다. 또한, 16x16 블록 내에는 하나의 진정한 DC 계수가 있고 나머지는 DCT와 DQT 에 해당하는 AC 계수들이다.
- [0075] 양자화를 위해 변환 계수들 (DCT와 DQT 모두) 이 이 양자화기에 제공된다. 바람직한 실시형태에서, DCT 계



수들은 주파수 가중화 마스크 (frequency weighting mask : FWM) 와 양자화 스케일 팩터 (factor) 를 사용하여 양자화된다. FWM 은 입력 DCT 계수들의 블록과 같은 크기의 주파수 가중치들의 테이블이다. 주파수 가중치들은 다른 DCT 계수들에 다른 가중치들을 적용한다. 가중치들은 인간 시각, 또는 광학 시스템이 더 민감한 주파수 내용을 갖는 입력 샘플들을 강조하고, 시각 또는 광학 시스템에 덜 민감한 주파수 내용을 갖는 샘플들을 경시하도록 고안(design)된다. 가중치들은 관찰 거리 등과 같은 팩터들에 기초하여 또한 고안될 수 있다.

[0076] 가중치들은 실험에 의한 데이터에 기초하여 선택된다. 8x8 DCT 계수들을 위한 가중화 마스크들을 고안하기 위한 방법은 본 발명에서 참고문헌으로 인용된 ISO/IEC JTC1 CD 10918, "Digital compression and encoding of continuous/tonal still images / part 1 : Requirements and guideline" International Standards Organization, 1994 에 개시된다. 일반적으로 2개의 FWM이 고안되는데, 하나는 휘도 성분을 위한 것이고, 하나는 색상 성분들을 위한 것이다. 블록 사이즈 2x2와 4x4를 위한 FWM 테이블들은 8x8블록의 데시메이션(decimation)에 의해 얻어지고, 16x16은 8x8 블록을 위한 보간(interpolation)에 의해 얻어진다. 스케일 팩터는 양자화된 계수들의 품질과 비트 레이트를 조정한다.

[0077] 그러므로, 각각의 계수는 하기 관련식,

$$DCT_q(i, j) = \left\lfloor \frac{8 * DCT(i, j)}{fwm(i, j) * q} \pm \frac{1}{2} \right\rfloor$$

[0078] 에 의해 양자화되고,

[0079] 여기서, DCT(i, j)는 입력 DCT 계수, fwm(i, j)는 주파수 가중화 마스크, q는 스케일 팩터, 그리고 DCT<sub>q</sub>(i, j)는 양자화된 계수이다. DCT 계수의 부호에 따라서 괄호 안의 첫 번째 항이 라운드 업 또는 다운된다. DQT 계수들은 또한 적당한 가중화 마스크를 사용하여 양자화된다. 그러나, 다수의 테이블 또는 마스크가 사용될 수 있고, 각각의 Y, Cb, 및 Cr 성분들에 대해 적용될 수 있다.

[0080] 그 후, AC 값들은 DC 값들로부터 분리되어 (130) 별도로 프로세싱된다. DC 성분들을 위해서 각각의 슬라이스의 제 1 DC 성분값이 인코딩된다. 각각의 슬라이스의 각각의 후속의 DC 성분값은 그 후 그것과 그것에 선행하는 성분값간의 차이에 의해 표현되고 인코딩된다 (134). 무손실 인코딩을 위해 각각의 슬라이스의 DC 성분값과 차이들은 도 6과 도 8 에 기재된 바와 같이 GR 을 사용하여 인코딩된다. 연속하는 DC 성분 값들간의 차이를 인코딩하기 위해 GR 을 사용하는 것은 DC 성분 값들의 차이들 (differentials) 이 양-측면의 지수분포를 갖는 경향이 있다는 점에서 유리하다. 그 후 데이터를 버퍼 (142)를 사용하여 일시적으로 저장될 수 있고, 그 후 전송 채널 (108)을 통해 디코더 (112) 로 전송 또는 송신된다.

[0081] 도 8 은 DC 성분값의 인코딩 프로세스를 나타낸다. 본 프로세스는 정지 이미지, (동영상 또는 고화질 TV 등으로 한정되는 것이 아닌) 비디오 이미지, 및 오디오에도 동일하게 적용될 수 있다. 소정의 데이터 슬라이스 (804) 에 대해, 슬라이스의 첫번째 DC 성분값을 검색한다 (808). 그 다음, 첫번째 DC 성분값을 코딩한다 (812). AC 성분값과 달리, DC 성분값은 양자화될 필요가 없다. 일 실시형태에서는, BSA 장애에 상관없이, 16 × 16 블록에 대한 단일의 DC 값이 이용된다. 8 × 8 또는 4 × 4 와 같은 임의의 고정 사이즈 블록이 나 BSA에 의해 정의되는 임의의 가변 블록 사이즈가 이용될 수 있음을 알 수 있다. 그 다음, 소정 슬라이스의 두번째, 또는 후속의 DC 성분값을 검색한다 (816). 그 다음, 두번째 DC 성분값을 첫번째 DC 성분값과 비교하고, 그 차이, 또는 리지듀얼 (residual) 을 인코딩한다 (820). 따라서, 두 번째 DC 성분값은 첫 번째 값과 두 번째 값간의 DC 성분값간의 차이로서 표현할 필요는 없다. 슬라이스의 DC 성분값 각각에 대해, 이러한 프로세스가 반복된다. 그 후, 슬라이스의 끝 (마지막 블록 및 그에 따른 마지막 DC 값) 에 도달했는지의 여부에 대한 검사가 수행된다 (824). 그렇지 않다면 (828), 슬라이스의 후속 DC 값이 검색되며 (816), 프로세스가 반복된다. 그렇다면 (832), 후속 슬라이스가 검색되며 (804), 프레임의 모든 슬라이스와 파일의 모든 프레임이 프로세싱될 때까지, 프로세스가 반복된다.

[0082] DC 성분값을 무손실 인코딩하는 목적은 리지듀얼 값이 낮은 분산을 갖도록 하기 위한 것이다. DCT 의 이용시, DC 계수의 성분값이 최대 화소 에너지에 기여한다. 따라서, DC 성분값을 양자화하지 않음으로써, 리지듀얼의 분산은 감소한다.

[0083] AC 성분의 경우, 그 다음, 데이터 블록 및 주파수 가중화 마스크가 양자화기 (146), 또는 스케일 팩터 요소에 의해 스케일링된다. DCT 계수의 양자화로 인해 이들 중 다수가 0 으로 감소되며, 이로 인해, 압축이 발생한

다. 바람직한 일 실시형태에는, 평균 비트 레이트에 대응하는 32 개의 스케일 팩터가 있다. MPEG 2 와 같은 다른 압축 방법과 달리, 평균 비트 레이트는, 목표 비트 레이트와 버퍼 상태 대신에, 프로세싱된 이미지에 기초하여 제어된다.

[0084] 압축을 더 증가시키기 위해, 양자화된 계수는 스캔 직렬 변환기 (150 ; scan serializer) 에 제공된다. 직렬 변환기 (150) 는 양자화된 계수 블록을 스캐닝하여 양자화된 계수의 직렬 변환된 스트림을 생성한다. 지그-재그 스캐닝, 컬럼 스캐닝, 또는 로우 스캐닝이 이용될 수 있다. 지그재그 이외의 패턴 뿐만 아니라 다수의 상이한 지그재그 스캐닝 패턴이 선택될 수 있다. 바람직한 일 기술은 지그재그 스캐닝에 8×8 블록 사이즈를 이용한다. 양자화된 계수의 지그재그 스캐닝은 제로값의 큰 런 (large run of zero values) 에 접할 기회를 증가시킨다. 이러한 제로 런은 본질적으로, 감소하는 확률을 가지며, 허프만 부호를 이용해 효율적으로 인코딩될 수 있다.

[0085] 직렬 변환되고 양자화된 AC 계수의 스트림은 가변 렱스 코더 (154) 에 제공된다. AC 성분값은 허프만 (Huffman) 인코딩 또는 골롬-라이스 (Golomb-Rice) 인코딩을 이용해 인코딩될 수 있다. DC 성분값에 대해서는, 골롬-라이스 인코딩을 이용한다. 런-렱스 코더 (run-length coder) 는 계수를 제로 계수와 논-제로 계수로 분리하는데, 도 6 을 참조하여 이를 상세히 설명한다. 일 실시형태에서는, 골롬-라이스 코딩이 이용된다. 골롬-라이스 코딩은 제로가 아닌 정수 분포를 이용해 코딩하기에 효율적이다. 지수 분포의 변수에 보다 짧은 렱스의 코드를 제공하는 압축을 위해서는 골롬 코드를 이용하는 것이 보다 적절하다.

[0086] Golomb 인코딩 런-렱스에서, Golomb 코드는 제로가 아닌 정수  $m$  에 의해 파라미터화 된다. 예를 들어, 소정 파라미터  $m$  에 대해, 양의 정수  $n$  의 Golomb 코딩은 변형된 2 진 코드로 표현되는 나머지가 수반되는 1 진 코드의  $n/m$  의 몫으로 표현되는데, 나머지가  $2^{\lceil \log_2 m \rceil} - m$  이하이면  $\lceil \log_2 m \rceil$  비트 길이이고, 그렇지 않으면,  $\lceil \log_2 m \rceil$  비트 길이이다. GR 코딩은, 파라미터  $m$  이  $m = 2^k$  으로 표현되는 Golomb 코딩의 특수한 경우이다. 이러한 경우,  $n/m$  의 몫은 정수  $n$  의 2 진 표현을 오른쪽으로  $k$  비트만큼 시프트시켜 얻으며,  $n/m$  의 나머지는  $n$  의 하위  $k$  비트로 표현된다. 따라서, GR 코드는 이 둘의 접합 (concatenation) 이다. GR 코딩은 수학식 1 로써 주어지는 양측 기하 (지수) 분포를 가진 양의 정수와 음의 정수 모두를 인코딩하는데 이용될 수 있다.

## 수학식 1

[0087] 
$$P_{\alpha}(x) = c \alpha^{|x|}$$

[0088] [수학식 1] 에서,  $\alpha$  는  $x$  의 확률 감쇠를 특징짓는 파라미터이고,  $c$  는 정규화 상수이다.  $P_{\alpha}(x)$  가 단조 (monotonic) 이기 때문에, 정수값의 시퀀스가 다음의 [수학식 2] 를 만족시켜야 함을 알 수 있다.

## 수학식 2

[0089] 
$$p_{\alpha}(x_i = 0) \geq p_{\alpha}(x_i = -1) \geq p_{\alpha}(x_i = +1) \geq p_{\alpha}(x_i = -2) \geq \dots$$

[0090] 도 4a, 4b, 4c 및 5a, 5b, 5c 에 나타난 바와 같이, 양자화된 DCT 계수 매트릭스의 제로-런과 진폭 모두는 지수 분포를 갖는다. 이들 도면에 도시된 분포는 실제 이미지로부터의 데이터에 기초한다. 도 4a 는 제로 런-렱스 대 상대 주파수의 Y 성분 분포 (400) 를 나타낸다. 마찬가지로, 도 4b 및 4c 각각은 제로 런-렱스 대 상대 주파수의 Cb 및 Cr 성분 분포 (410 및 420) 를 나타낸다. 도 5a 는 진폭 사이즈 대 상대 주파수의 Y 성분 분포 (500) 를 나타낸다. 마찬가지로, 도 5b 및 5c 각각은 진폭 사이즈 대 상대 주파수의 Cb 및 Cr 성분 분포 (510 및 520) 를 나타낸다. 도 5a, 5b 및 5c 에서, 플롯은 DCT 계수의 사이즈 분포를 나타낸다. 각 사이즈는 계수 값의 범위를 나타낸다. 예를 들어, 4 의 사이즈 값은  $\{-15, -14, \dots, -8, 8, \dots, 14, 15\}$  범위, 총 16 개 값을 갖는다. 마찬가지로, 10 의 사이즈 값은  $\{-1023, -1022, \dots, -512, 512, \dots, 1022, 1023\}$  범위, 총 1024 개 값을 갖는다. 도 4a, 4b, 4c, 5a, 5b 및 5c 로부터, 런-렱스와 진폭 사이즈 모두 지수 분포를 갖는다는 것을 알 수 있다. 진폭의 실제 분포는 다음의 [수학식 3] 을 따른다는 것을 알 수 있다.

### 수학식 3

$$p(X_{k,l}) = \frac{\sqrt{2\lambda}}{2} \exp\left[-\sqrt{2\lambda}|X_{k,l}|\right], k, l \neq 0$$

[0091]

[0092]

수학식 3 에서,  $X_{k,l}$  는 각각 수직 차원과 수평 차원의 주파수  $k$  와  $l$  에 대응하는 DCT 계수를 나타내고, 평균

$\mu_x = \frac{1}{\sqrt{2\lambda}}$  이며, 분산  $\sigma^2_x = \frac{1}{2\lambda}$  이다. 따라서, 상술한 방식으로 GR 코딩을 이용하는 것이 DCT 에서의 데이터 프로세싱에 보다 적절하다.

[0093]

이하에서는 이미지 데이터의 압축에 대해 설명하였지만, 본 실시형태들은 오디오 데이터를 압축하는 실시형태에도 동일하게 적용될 수 있다. 이미지 데이터 압축의 경우, 이미지 또는 비디오 신호는, 예를 들어, 선형 또는 로그 인코딩된 화소값을 갖는 RGB, YIQ, YUV, 또는 YCbCr 성분일 수 있다.

[0094]

도 6 은 제로 및 논-제로 계수의 인코딩 프로세스 (600) 를 나타낸다. DCT 매트릭스가 스캐닝됨에 따라, 제로와 논-제로 계수를 개별적으로 프로세싱하여 분리한다 (604). 제로 데이터의 경우, 제로 런의 렘스가 결정된다. 런-렘스는 양의 정수이다. 예를 들어, 런-렘스가  $n$  임을 알게 되면, 곱셈 파라미터  $m$  이 결정된다 (612). 일 실시형태에서, 곱셈 파라미터는 런 렘스의 함수로서 결정된다. 다른 실시형태에서, 곱셈 파라미터 ( $m$ ) 는 다음의 [수학식 4] 에 의해 결정된다.

### 수학식 4

$$m = \lceil \log_2 n \rceil$$

[0095]

[0096]

선택적으로, 런-렘스의 길이와 관련 곱셈 파라미터가 카운터 또는 레지스터에 의해 카운팅된다 (616). 제로의 런-렘스 ( $n$ ) 를 인코딩하기 위해, 몫이 인코딩된다 (620). 일 실시형태에서, 몫은 제로의 런-렘스와 곱셈 파라미터의 함수로서 결정된다. 다른 실시형태에서, 몫 ( $Q$ ) 은 다음의 [수학식 5] 에 의해 결정된다.

### 수학식 5

$$Q = \lfloor n/2^m \rfloor$$

[0097]

[0098]

일 실시형태에서, 몫 ( $Q$ ) 은 1 진 코드로 인코딩되는데, 이는  $Q+1$  비트를 요한다. 다음으로, 나머지가 인코딩된다 (624). 일 실시형태에서, 나머지는 런 렘스와 몫의 함수로서 인코딩된다. 다른 실시형태에서, 나머지 ( $R$ ) 는 다음의 [수학식 6] 을 이용해 결정된다.

### 수학식 6

$$R = n - 2^m Q$$

[0099]

[0100]

일 실시형태에서, 나머지 ( $R$ ) 는  $m$ -비트의 2 진 코드로 인코딩된다. 몫 ( $Q$ ) 과 나머지 ( $R$ ) 를 결정한 후,  $Q$  와  $R$  에 대한 코드를 접합하여 (628) 제로의 런-렘스 ( $n$ ) 에 대한 전체 코드를 나타낸다.

[0101]

논-제로 계수도 GR 을 이용해 인코딩된다. 계수 진폭은 양 또는 음일 수 있으므로, 부호 비트를 이용해야 하며 소정 진폭의 절대값을 인코딩해야 한다. 논-제로 계수의 진폭이  $x$  일 경우, 진폭은 진폭의 절대값과 부호의 함수로 표현될 수 있다. 따라서, 진폭은 다음의 수학식 7 을 이용하여  $y$  로 표현될 수 있다.

### 수학식 7

$$y = \begin{cases} 2x, & x \geq 0 \text{이면} \\ 2|x| - 1, & \text{그렇지 않으면} \end{cases}$$

[0102]

[0103]

따라서, 논-제로 계수값이 카운터 또는 레지스터에 의해 적절하게 카운팅된다 (632). 그 다음, 진폭이 제로 이상인지의 여부가 결정된다 (636). 그렇다면, 그 값은 소정값의 2 배로 인코딩된다 (640). 그렇지 않으면, 그 값은 절대값의 2 배보다 1 작은 값으로 인코딩된다 (644). 다른 매핑 방식도 이용될 수 있음을 알



수 있다. 요점은 그 값의 부호를 구별하기 위한 추가 비트가 불필요하다는 것이다.

[0104] [수학식 7] 로 표현된 바와 같이 진폭을 인코딩함으로써,  $x$  의 양의 값은 짝수가 되고 음의 값은 홀수가 된다. 또한, 이러한 매핑은 [수학식 2] 에서와 같은  $x$  의 확률 할당을 보존한다. [수학식 7] 에 나타난 바와 같이, 인코딩하는 것의 이점은 양수와 음수를 나타내기 위한 부호 비트를 이용하지 않아도 된다는 점이다. 매핑이 이루어지고 난 후,  $y$  는 제로-런에 대해 수행된 것과 동일한 방식으로 인코딩된다. 현재 블록의 모든 계수가 스캐닝될 때까지, 이 과정이 계속된다.

[0105] 본 발명의 실시형태들이 [수학식 1] 내지 [수학식 7] 의 함수로서 계수의 값과 런-렌스를 결정하긴 하지만, 정확한 [수학식 1] 내지 [수학식 7] 을 이용할 필요는 없다. 골롬-라이스 인코딩과 DCT 계수의 지수 분포를 이용함으로써, 이미지 및 오디오 데이터를 보다 효율적으로 압축할 수 있다.

[0106] 인코딩 후의 제로-런은 논-제로 진폭과 구별이 불가능하므로, 고정 길이의 특수한 프리픽스 코드를 이용하여 첫 번째 제로-런의 발생을 마킹할 필요가 있다. 논-제로 진폭을 처리한 후에 블록의 모든 제로를 처리하는 것이 일반적이다. 이러한 경우, GR 코드보다 EOB (end-of-block) 라는 코드를 이용하는 것이 더 효율적일 수 있다. EOB 코드 또한, 선택적으로, 특수한 고정 길이의 코드이다.

[0107] [수학식 1] 또는 [수학식 3] 에 따르면, DCT 계수 매트릭스의 진폭 또는 런-렌스의 확률 분포가  $\alpha$  또는  $\lambda$  에 의해 파라미터화 된다. 이는 특정한 DCT 계수 블록이 발생한다면, 코딩 효율성이 향상될 수 있음을 암시한다. 그 다음, 관심있는 양 (quantity of interest) 을 인코딩하기 위한 적절한 골롬-라이스 파라미터가 이용될 수 있다. 일 실시형태에서, 각각의 누적값 및 이러한 값이 발생하는 대응 횟수를 계산하기 위해, 각각의 런-렌스 및 진폭 사이즈 값에 대해 카운터 또는 레지스터가 이용된다. 예를 들어, 레지스터가 누적값 및 누적된 요소의 수를 저장하기 위한 레지스터가 각각  $R_{r1}$  와  $N_{r1}$  라면, 런-렌스를 인코딩하기 위한 라이스-골롬 (Rice-Golomb) 파라미터로서 다음의 [수학식 8] 이 이용될 수 있다.

### 수학식 8

$$\left\lceil \log_2 \frac{R_r}{N_r} \right\rceil$$

[0108]

[0109] 진폭에 대해서도 유사한 과정이 이용될 수 있다.

[0110]

리지듀얼 픽셀들은 ABSDCT 디코더를 이용하여 압축된 데이터를 압축해제함으로써 생성되고, 다음으로, 오리지널 데이터로부터 이를 감산한다. 리지듀얼 동적 범위가 작아질수록, 압축은 크게 된다. 압축이 블록-기반이기 때문에, 리지듀얼 또한 블록 기반으로 생성된다. 통상, 제로에 중심을 두고 리지듀얼 픽셀이 양측 지수 분포를 갖는 것은 공지된 사실이다. GR 코드가 이러한 데이터에 대해 최적이기 때문에, 이 리지듀얼 데이터를 압축하기 위해 GR 코딩 절차가 이용된다. 그러나, 인코딩된 런-렌스가 없기 때문에 특별한 코드가 필요한 것은 아니다. 또한, EOB 코드가 필요하지 않다. 따라서, 압축된 데이터는 2 개의 컴포넌트로 이루어진다. 하나는 손실 압축기로부터의 컴포넌트이고, 나머지는 무손실 압축기로부터의 컴포넌트이다.

[0111] 모션 시퀀스의 인코딩 시, 일시적인 상관성을 이용하는 것이 이로울 수 있다. 일시적인 상관성을 모두 이용하기 위해, 먼저 모션으로 인한 픽셀 전치가 설정되고, 다음으로, 모션 보상된 예측이 리지듀얼 픽셀을 획득하기 위해 수행된다. ABSDCT 가 적합한 블록 사이즈 인코딩을 수행함에 따라, 블록 사이즈 정보가 모션으로 인한 전치의 측정으로 이용될 수 있다. 더욱 간단하게 하기 위해, 신호 변화 검출이 이용되지 않는다. 그 대신, 일련의 각각의 프레임에 대해, 압축된 데이터가 획득된다. 다음으로, 현재 및 이전의 프레임 DCT 사이의 차이는 블록X블록에 기초하여 생성된다. 이는 여기에 참조로서 포함되는 2001년 6월 7일자 미국 특허 출원 09/877,578 호에서 더욱 설명된다. DCT 도메인의 이러한 리지듀얼은 허프만 및 GR 코딩 절차를 모두 이용함으로써 인코딩된다. 다음으로, 마지막으로 압축되는 출력은 프레임 당 최소 비트를 이용하는 출력에 대응된다.

[0112] 무손실 압축 알고리즘은 무손실 부분을 제거함으로써 용도변경(repurposing) 및 트랜스코딩에 적합하게 되는 하이브리드 방식이다. 따라서, ABSDCT 를 이용함으로써 공간 도메인의 픽셀 상관성을 최대화하여, 예측 방식에서 이용되는 것보다 낮은 분산을 갖는 리지듀얼 픽셀을 발생시킨다. 전체 시스템의 손실부는, 사용자가 인터프레임 프로세싱에 의존함이 없이 분배 목적을 위한 필요한 품질 및 데이터 속도를 달성하도록 하여, 관련 화질 열화(related motion artifact)를 제거하고 구현 복잡성을 현저하게 감소시킨다. 특히, 압축된 재료의

손실 부분이 그 분배 시 더욱 높은 레벨의 품질을 필요로 하기 때문에, 이는 디지털 시네마 응용을 위해 분배될 프로그램에 대해 중요하다.

- [0113] 도 9 는 하이브리드 무손실 인코딩 장치 (900) 를 나타낸다. 도 10 은 이러한 장치에서 동작될 수 있는 프로세스를 나타낸다. 오리지널 디지털 정보 (904) 는 저장 장치 상에 위치되거나, 또는 전송된다. 도 9 의 대부분의 요소는 도 1 및 도 2 와 관련하여 보다 상세하게 설명된다. 데이터 프레임은 BSA 요소 (912), DCT/DQT 변환 요소 (916), 및 양자화기 (920) 를 포함하는 압축기 (908) 로 전송된다. DCT/DQT 를 데이터 상에서 수행한 후, 데이터는 주파수 도메인으로 변환된다. 출력 (922) 에서, 데이터는 양자화기 (920) 에 의해 양자화되고, 저장 및/또는 스위칭을 포함할 수 있는 출력 (924) 으로 전송된다. 전송한 프로세싱 모두는 인트라프레임 기반에 있다.
- [0114] 또한, 양자화 출력은 압축해제기 (928) 로 전송된다. 압축해제기 (928) 는 BSA 에 의해 정의되는 바와 같이 PQR 데이터의 지식을 따르는 역양자화기 (932) 와 IDQT/IDCT (936) 를 통과하는 압축기의 프로세스를 취소한다. 감산기 (940) 의 결과는 오리지널과 비교되는 감산기 (944) 로 제공된다. 감산기 (944) 는 각각의 블록에 대해 미압축된 픽셀과 압축되고 압축해제된 픽셀간의 차이로서 리지듀얼 픽셀을 계산하는 차이기 (differencer) 와 같은 다양한 요소일 수 있다. 또한, 차이기는 조건부 인터프레임 코딩에 대해 각각의 블록에 대한 DCT 도메인의 리지듀얼을 획득할 수 있다. 압축해제된 데이터와 오리지널 데이터 사이의 비교 결과 (948) 가 픽셀 리지듀얼 파일이다. 즉, 결과 (948) 는 압축 및 미압축되는 데이터에 의해 표현되는 손실을 나타낸다. 따라서, 오리지널 데이터는 결과 (948) 와 결합되는 출력 (922) 과 동일하다. 다음으로, 결과 (948) 는 직렬 변환기 (952) 및 허프만 및/또는 GR 인코더 (956) 이고, 제 2 출력 (960) 으로 제공된다. 허프만 및/또는 GR 인코더 (956) 는 GR 코딩을 이용하여 리지듀얼 화픽셀을 인코딩하는 엔트로피 인코더 타입이 될 수 있다. 결정은 각 프레임에 대한 최소 비트에 기초하여 인트라프레임 또는 인터프레임을 이용하기 위해 이루어진다. 리지듀얼의 GR 코딩의 이용은 시스템의 전체 압축비를 더욱 높게 한다.
- [0115] 따라서, 무손실 인터프레임 출력은 손실 고품질 이미지 파일 (922 또는 A) 과 리지듀얼 파일 (960 또는 C) 의 2 세트의 데이터의 결합 또는 하이브리드이다.
- [0116] 또한, 인터프레임 코딩이 이용될 수 있다. 양자화기의 출력은 BSA 의 지식과 함께 저장부 (964) 로 전송된다. 데이터의 프레임의 가치를 축적하면서, 감산기 (966) 는 저장된 프레임 (964) 을 다음의 프레임 (968) 과 비교한다. 차이 결과는 직렬 변환기 및/또는 GR 인코더 (974) 되는 DCT 리지듀얼 (970) 을 초래하고, 제 3 출력 데이터 세트 (976) 를 출력 (924) 으로 제공한다. 따라서, B 및 C 의 인터프레임 무손실 파일이 컴파일된다. 따라서, 결합 (A+B 또는 B+C) 은 사이즈를 고려하여 선택될 수 있다. 또한, 순수 인트라프레임 출력은 편집 목적에서 바람직하다.
- [0117] 다시 도 1 을 참조하면, 인코더 (104) 에 의해 생성된 압축 이미지 신호는 버퍼 (142) 를 이용하여 임시 저장된 다음, 전송 채널 (108) 을 이용하여 디코더 (112) 로 전송될 수 있다. 전송 채널 (108) 은 자기적 또는 광학적 저장 장치와 같은 물리적 매체이거나 유선 또는 무선의 전달 프로세스나 장치일 수 있다. BSA 정보를 포함하는 PQR 데이터도 디코더 (112) 에 제공된다 (도 2). 디코더 (112) 는 버퍼 (164) 및 런-렝스 값과 논-제로 값을 디코딩하는 가변 렝스 디코더 (168) 를 구비한다. 가변 렝스 디코더 (168) 는 유사하지만 도 6 에서 설명한 것과 반대 방식으로 동작한다.
- [0118] 가변 렝스 디코더 (168) 의 출력은, 이용된 스캐닝 방식에 따라 계수의 순서를 지정하는 역 직렬 변환기 (172 ; inverse serializer) 에 제공된다. 예를 들어, 지그 재그 스캐닝, 수직 스캐닝, 및 수평 스캐닝의 결합이 이용되었다면, 역 직렬 변환기 (172) 는 이용된 스캐닝 타입에 따라 계수를 재지정한다. 역 직렬 변환기 (172) 는 계수를 복합 계수 블록으로 적절하게 순서 지정하는 것을 돕는 PQR 데이터를 수신한다.
- [0119] 양자화 스케일 팩터 및 주파수 가중화 마스크의 이용으로 인한 프로세싱을 취소하기 위해, 복합 블록은 역 직렬 변환기 (174) 로 제공된다.
- [0120] 그 다음, DQT (Differential Quad-tree Transform) 가 이용되었다면, 복합 블록은 IDCT 요소 (190) 가 수반되는 IDQT 요소 (186) 에 제공된다. 그렇지 않다면, 계수 블록은 IDCT 요소 (190) 에 직접 제공된다. IDQT 요소 (186) 와 IDCT 요소 (190) 는 계수를 역변환하여 픽셀 데이터의 블록을 생성한다. 그 다음, 픽셀 데이터는 RGB 형태로 보관되고 변환된 후, 장치의 디스플레이를 위해 저장되어야 한다.
- [0121] 도 7 은 콜롬-라이스 인코딩을 위한 장치 (700) 를 나타낸다. 도 7 의 장치는 바람직하게도 도 6 을 참조하여 설명한 바와 같은 프로세스를 구현한다. 결정기 (704) 는 런 렝스 (n) 와 콜롬 파라미터 (m) 를 결정한다

다. 옵션으로, 개개의 누적값과 이러한 값이 발생하는 대응 횟수를 계산하기 위해, 각각의 런-렝스와 진폭 사이즈 값에 대해 카운터 또는 레지스터 (708) 를 이용할 수 있다. 인코더 (712) 는 몫 (Q) 을 런 렝스와 곱셈 파라미터의 함수로서 인코딩한다. 인코더 (712) 는 또한, 나머지 (R) 를 런 렝스, 곱셈 파라미터, 및 몫의 함수로서 인코딩한다. 다른 실시형태에서, 인코더 (712) 는 또한, 데이터를 논-제로 데이터값과 논-제로 데이터값의 코드의 함수로서 논제로 인코딩한다. 접합기 (716) 를 이용하여, Q 값을 R 값과 접합한다.

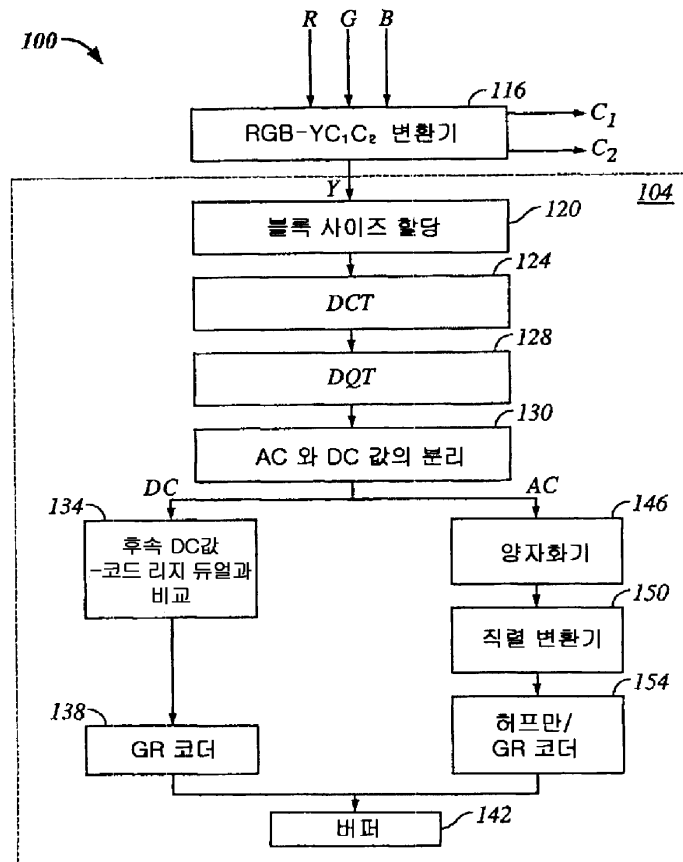
[0122] 일례로서, 여기에 개시된 실시형태와 관련하여 설명된 다양한 예시적 논리 블록, 흐름도, 및 단계들은, ASIC (application-specific integrated circuit), 프로그램가능한 로직 장치, 개별적인 게이트나 트랜지스터 로직, 예를 들어, 레지스터 및 FIFO 와 같은 개별적인 하드웨어 컴포넌트, 한 세트의 펌웨어 명령어를 실행하는 프로세서, 종래의 프로그램 가능한 소프트웨어 및 프로세서, 또는 이들의 임의 조합을 이용하여 하드웨어 또는 소프트웨어로 구현되거나 수행될 수 있다. 프로세서는 마이크로프로세서인 것이 바람직할 수 있지만, 다른 방법으로, 임의의 종래 프로세서, 컨트롤러, 마이크로컨트롤러, 또는 스테이트 머신일 수 있다. 소프트웨어는 RAM 메모리, 플래시 메모리, ROM 메모리, 레지스터, 하드 디스크, 이동가능한 디스크, CD-ROM, DVD-ROM, 또는 업계에 공지되어 있는 임의의 다른 형태의 저장 매체에 상주할 수 있다.

[0123] 바람직한 실시형태에 대한 상기 설명은 당업자가 본 발명을 이용할 수 있도록 하기 위해 제공되었다. 당업자는 이들 실시형태를 다양하게 변형할 수 있으며, 창의력을 이용하지 않고도, 여기에서 정의된 일반적인 원리를 다른 실시형태에 적용할 수 있다. 따라서, 본 발명은 여기에 개시된 실시형태들에 한정되지 않으며, 여기에 개시된 원리와 신규한 특징에 부합하는 최광의로 해석되어야 한다.

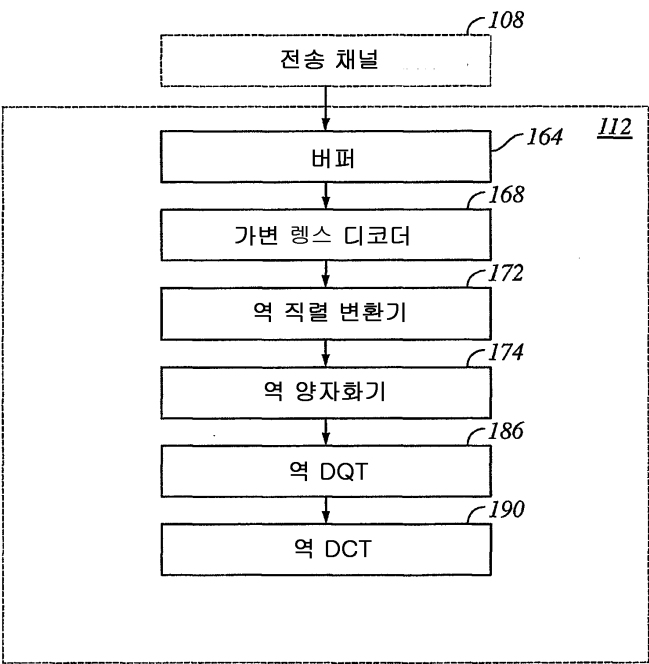
[0124] 본 발명의 다른 특징 및 이점을 이하의 청구범위에 기술한다.

## 도면

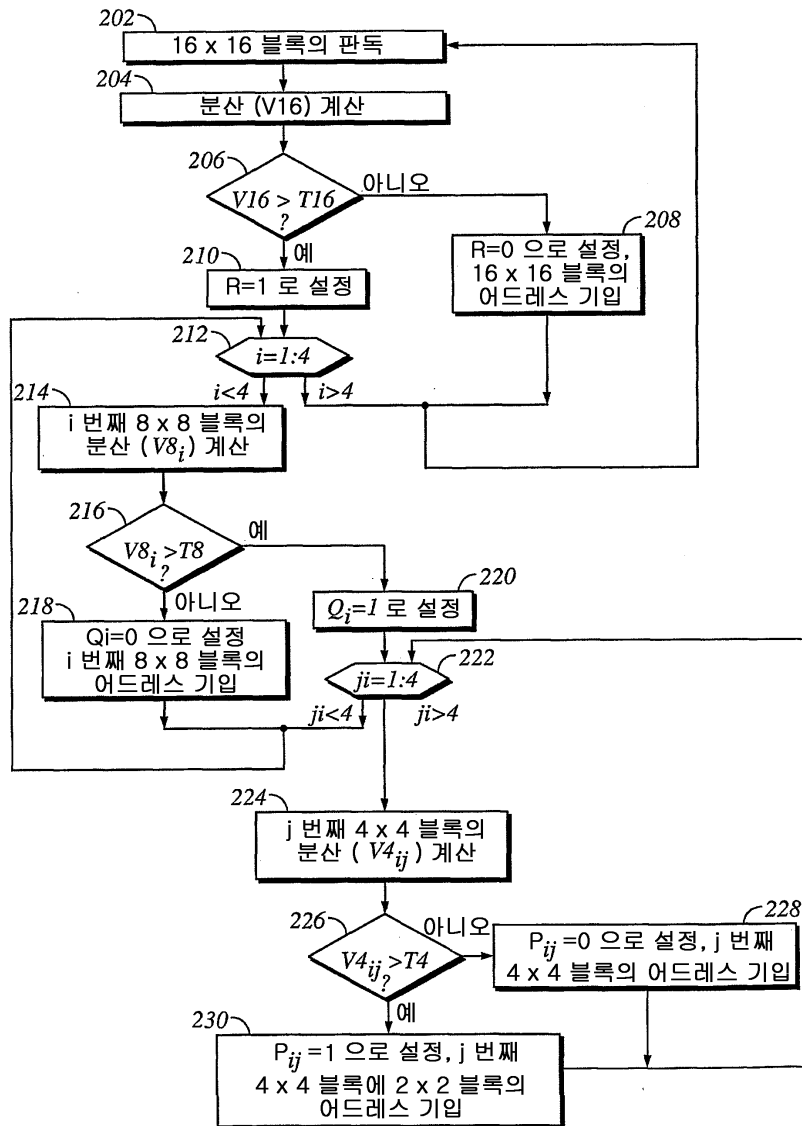
### 도면1



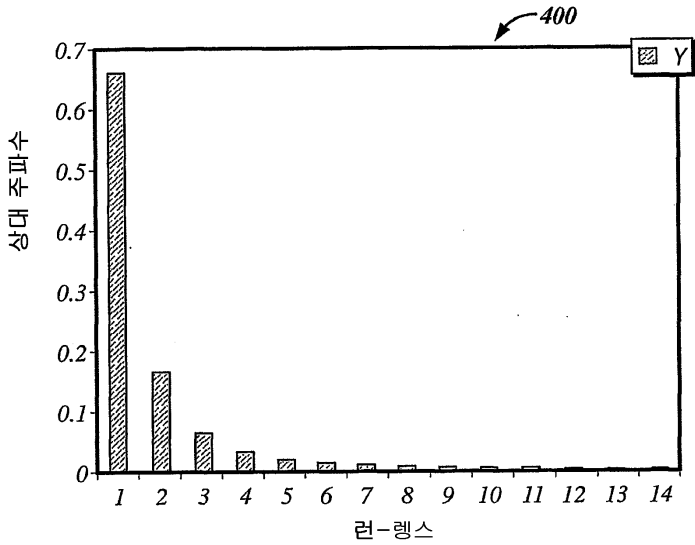
도면2



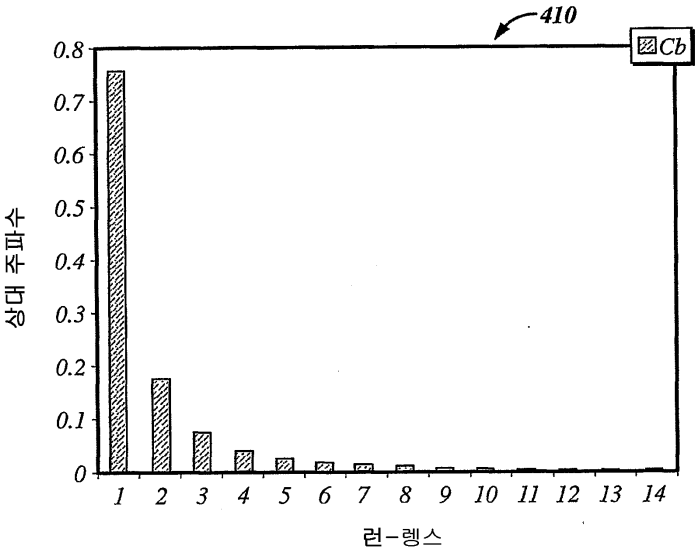
도면3



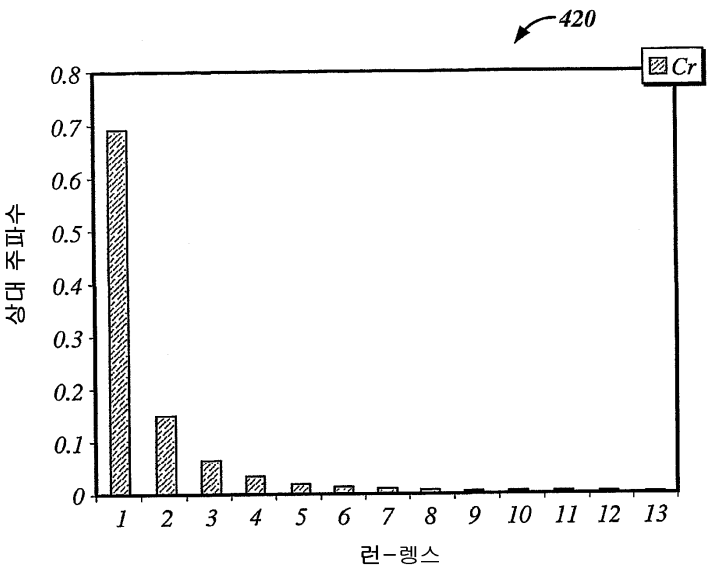
도면4a



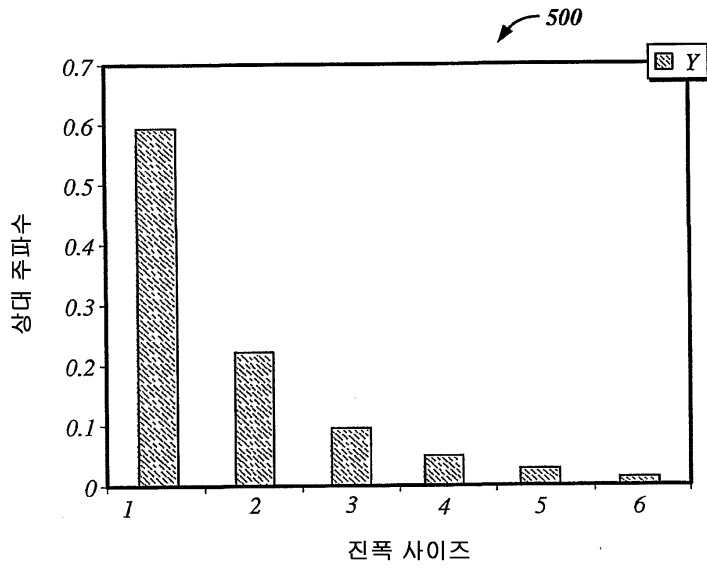
도면4b



도면4c

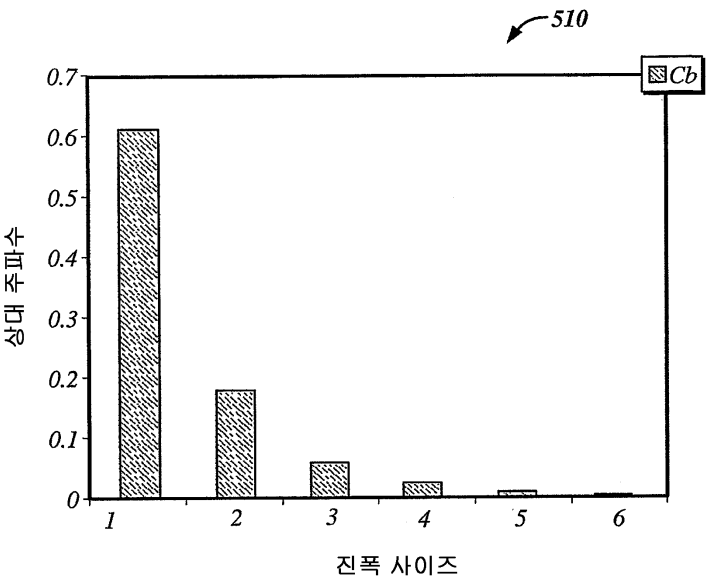


도면5a

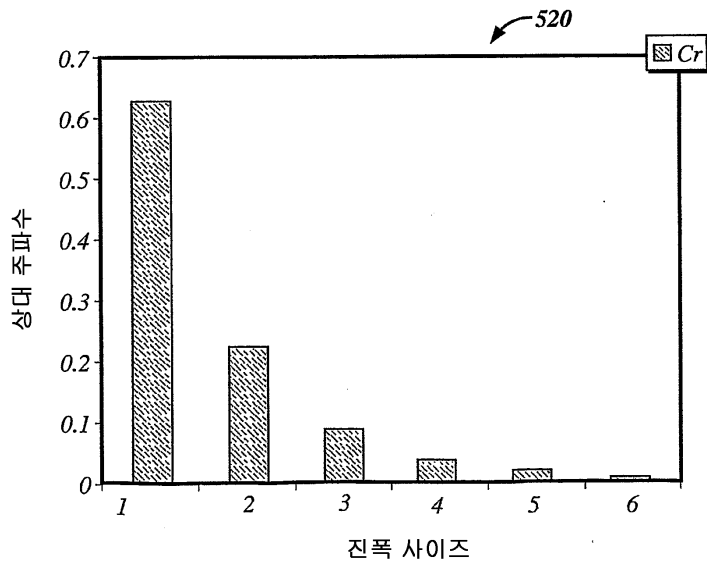




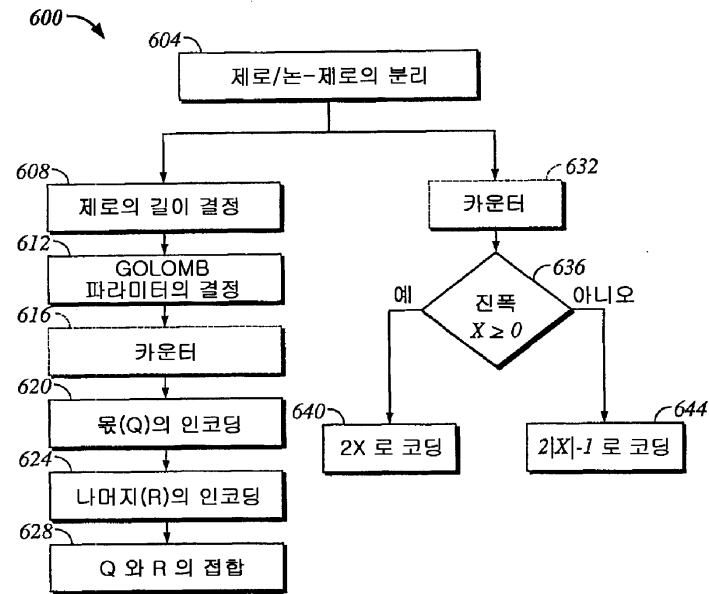
도면5b



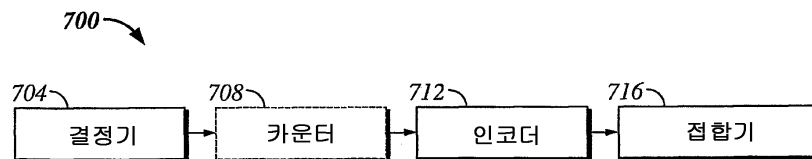
도면5c



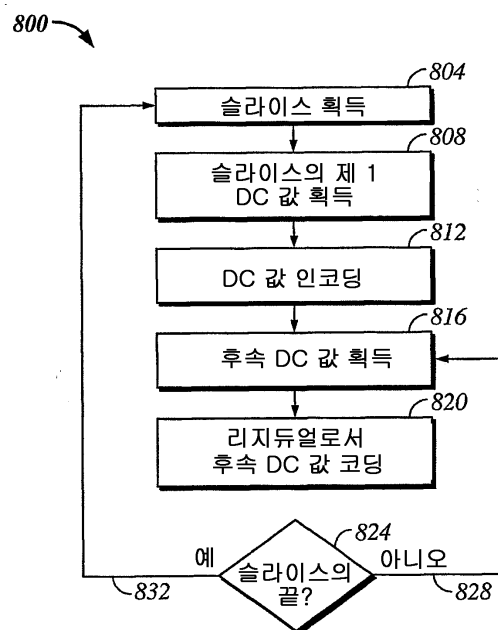
도면6



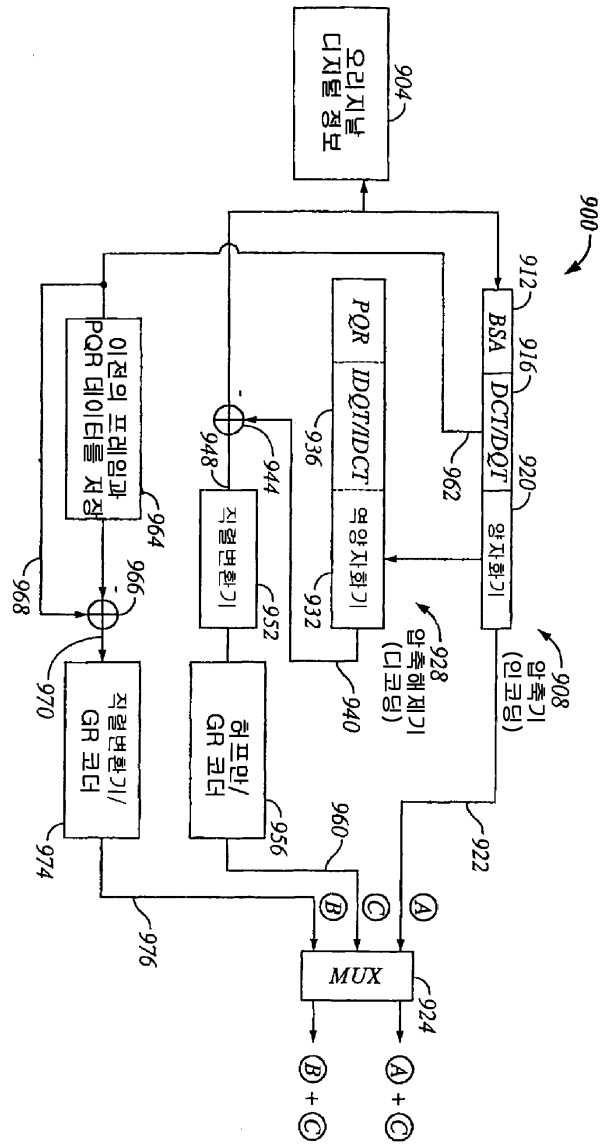
도면7



도면8



도면9



도면10

