



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2016년08월22일

(11) 등록번호 10-1649882

(24) 등록일자 2016년08월16일

- (51) 국제특허분류(Int. Cl.)  
**HO4N 7/24** (2011.01) **HO4N 5/21** (2016.01)
- (21) 출원번호 10-2011-7021340
- (22) 출원일자(국제) 2010년02월10일  
 심사청구일자 2015년02월09일
- (85) 번역문제출일자 2011년09월09일
- (65) 공개번호 10-2011-0116055
- (43) 공개일자 2011년10월24일
- (86) 국제출원번호 PCT/US2010/023775
- (87) 국제공개번호 WO 2010/093709  
 국제공개일자 2010년08월19일
- (30) 우선권주장  
 61/151,434 2009년02월10일 미국(US)
- (56) 선행기술조사문헌  
 JP11196419 A\*  
 JP2007312370 A\*
- \*는 심사관에 의하여 인용된 문헌

- (73) 특허권자  
래티스세미컨덕터코퍼레이션  
미국 오리건 (우편번호 97204) 포틀랜드 에스더블  
유 피프스 애비뉴 111 스위트 700
- (72) 발명자  
아담스, 데일, 리차드  
미국 95445 캘리포니아주 구알라라 피.오.박스  
448  
톰프슨, 로렌스, 에이.  
미국 95070 캘리포니아주 사라토가 칼레 타쿠마  
13785
- (74) 대리인  
특허법인에이아이피

전체 청구항 수 : 총 26 항

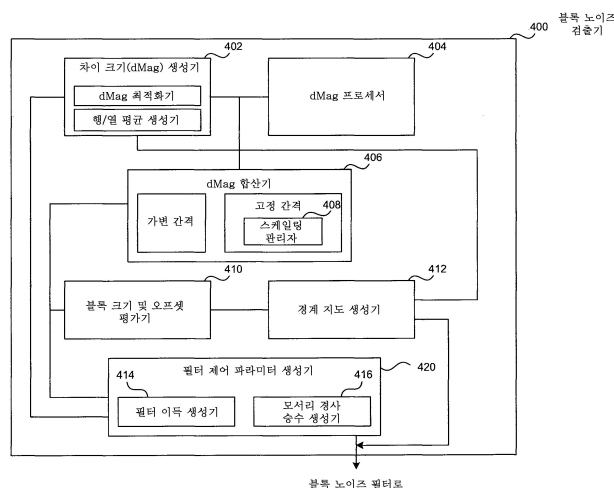
심사관 : 박상철

## (54) 발명의 명칭 블록 노이즈 검출 및 필터링

(57) 요약

블록 노이즈 검출 및 필터링을 위한 시스템 및 방법이 개시된다. 일 실시예는 이미지 내의 인접 화소들에 대한 화소값들의 차이 크기를 계산하는 단계를 포함한다. 차이 크기는 수평으로 인접한 화소들에 대한 수평 차이 크기 및 수직으로 인접한 화소들에 대한 수직 차이 크기를 포함할 수 있다. 일 실시예는 차이 크기의 정규화된 합을 사용하여 블록 노이즈의 노이즈 특성의 집합 및 이미지의 이미지 특성의 집합을 결정하는 단계 및 노이즈 및 이미지 특성의 집합을 사용하여 블록 노이즈 필터에 대한 입력을 구성하는 단계를 더 포함한다.

대표도 - 도4



## 명세서

### 청구범위

#### 청구항 1

이미지 내 블록 노이즈 검출 방법으로서,

상기 이미지 내의 인접 화소들에 대한 화소값들의 차이 크기(difference magnitude)들을 계산하는 단계;

감소된 차이 크기들의 집합을 생성하기 위해서 상기 이미지 내의 특징(feature)들로부터 비롯된 차이 크기들로부터의 차이 크기들의 집합을 제거하는 단계;

평균 차이 크기들의 집합(averaged set of difference magnitude)을 생성하기 위해서 상기 감소된 차이 크기들의 집합을 평균하는 단계;

필터링된 차이 크기들의 집합을 생성하기 위해서 평균된 차이 크기들의 집합에 중간값 필터(median filter)를 적용하는 단계;

차이 크기들의 결과 집합(result set of difference magnitude)을 생성하기 위해서 상기 평균된 차이 크기들의 집합으로부터 상기 필터링된 차이 크기들의 집합을 감산하는 단계;

모서리 화소 위치(edge pixel location)들을 검출하기 위해서 상기 이미지에 모서리 검출을 수행하는 단계; 및  
억제된 차이 크기들의 집합을 생성하기 위해서 상기 모서리 화소 위치들에 해당하는 차이 크기들의 결과 집합에서 크기들을 억제하는 단계;

를 포함하는 블록 노이즈 검출 방법.

#### 청구항 2

제1항에 있어서,

상기 차이 크기들은 수평으로 인접하는 화소들에 대한 수평 차이 크기들 및 수직으로 인접하는 화소들에 대한 수직 차이 크기들을 포함하고,

상기 평균하는 단계는

상기 수평 차이 크기들을 수직 방향으로 평균하여 상기 이미지의 폭의 수평 차이 크기들의 평균값들의 제1 벡터를 생성하는 단계; 및

상기 수직 차이 크기들을 수평 방향으로 평균하여 상기 이미지의 높이의 수직 차이 크기들의 평균값들의 제2 벡터를 생성하는 단계

를 포함하는 블록 노이즈 검출 방법.

#### 청구항 3

제2항에 있어서,

상기 이미지의 폭을 따르는 화소 위치들에 대해 상기 제1 벡터를 그래프로 플롯(plot)하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 4

제2항에 있어서,

상기 이미지의 높이를 따르는 화소 위치들에 대해 상기 제2 벡터를 그래프로 플롯하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 5

제2항에 있어서,

상기 수직 차이 크기들의 평균값들 또는 상기 수평 차이 크기들의 평균값들로부터 DC 레벨을 제거하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 6

제5항에 있어서,

상기 DC 레벨은 상기 평균값들에 중간값 필터를 적용하고 상기 평균값들로부터 그 결과를 감산함으로써 제거되는 블록 노이즈 검출 방법.

#### 청구항 7

제2항에 있어서,

상기 이미지의 모서리들 근처에서 상기 제1 및 제2 벡터들 내의 값들을 억제하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 8

제2항에 있어서,

후보 구간 크기들의 집합 및 후보 오프셋들의 집합에 대한 차이 크기들의 정규화된 합들을 결정하는 단계; 및  
상기 정규화된 합들의 가장 큰 값을 사용하여 상기 블록 노이즈의 구간 크기 및 오프셋을 결정하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 9

제8항에 있어서,

상기 구간 크기 및 오프셋을 사용하여 공칭 블록 크기 및 시작 경계 블록 위치를 결정하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 10

제8항에 있어서,

상기 차이 크기들의 3개의 정규화된 합들을 결정하는 단계를 더 포함하고,

상기 3개의 정규화된 합 중 제1 합산에서, 공칭 위치 및 상기 공칭 위치에서 1을 뺀 것을 사용하여 합산값들을 계산하고 상기 합산값들 중 더 큰 것을 산출하는 위치를 선택하고,

상기 3개의 정규화된 합 중 제2 합산에서, 공칭 위치 및 상기 공칭 위치에 1을 더한 것을 사용하여 합산값들을 계산하고 상기 합산값들 중 더 큰 것을 산출하는 위치를 선택하고,

상기 3개의 정규화된 합 중 제3 합산에서, 상기 공칭 위치를 사용하여 합산값을 계산하고 상기 공칭 위치를 선택하는 블록 노이즈 검출 방법.

#### 청구항 11

제10항에 있어서,

차이 크기들의 최대 정규화 합을 생성하는 후보 구간 크기 및 오프셋값을 식별하는 단계; 및

상기 최대 정규화 합을 상기 후보 구간 크기의 다른 두 오프셋값에 대해 계산된 합들과 비교하여 상기 블록 노이즈의 블록 크기를 결정하고 상기 공칭 블록 크기 및 시작 경계 위치를 사용하여 최종 블록 크기 및 오프셋값을 선택하는 단계

를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 12

제9항에 있어서,  
 후속 경계 블록 위치를 식별하는 단계를 더 포함하고,  
 상기 후속 경계 블록 위치는,  
 상기 시작 블록 위치를 상기 공칭 블록 크기만큼 증가시키는 단계;  
 상기 후속 경계 블록 위치 주위의 윈도우에서 최대 또는 근사 최대 차이 크기값의 위치를 식별하는 단계; 및  
 상기 최대 또는 근사 최대값의 위치를 상기 후속 블록 경계 위치로서 선택하는 단계  
 에 의해 결정되는 블록 노이즈 검출 방법.

#### 청구항 13

제12항에 있어서,  
 복수의 후속 경계 블록 위치를 사용하여 블록 경계 지도를 생성하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 14

제1항에 있어서,  
 상기 차이 크기들을 사용하여 상기 이미지가 스케일링되는지를 결정하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 15

제1항에 있어서,  
 수평 및 수직으로 인접하는 화소들은 1 화소 차분들을 갖는 블록 노이즈 검출 방법.

#### 청구항 16

제1항에 있어서,  
 문턱값 이상인 차이 크기들의 집합을 식별하고 상기 차이 크기들의 집합을 상기 문턱값으로 설정함으로써 상기 차이 크기들의 집합을 감소시키는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 17

제1항에 있어서,  
 상기 차이 크기들로부터 상기 중간값 필터의 출력을 감산하는 단계는, 0 미만의 값들을 0에서 클리핑(clip)하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 18

제1항에 있어서,  
 상기 모서리 검출은 소벨(Sobel) 필터들을 사용하여 수행되는 블록 노이즈 검출 방법.

#### 청구항 19

제18항에 있어서,  
 상기 소벨 필터들은  $3 \times 3$ 인 블록 노이즈 검출 방법.

#### 청구항 20

제2항에 있어서,  
 상기 모서리 화소 위치들에서 상기 모서리 차이 크기들을 억제하는 단계는,

해당 경사 크기가 경사 문턱값( $\text{gradient threshold value}$ ) 이상인 상기 모서리 화소 위치들의 위치에서 상기 모서리 차이 크기를 0으로 설정하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 21

제2항에 있어서,

상기 차이 크기들은 휘도 및 채도에 대해 계산되는 블록 노이즈 검출 방법.

#### 청구항 22

제21항에 있어서,

상기 채도에 대해 계산된 차이 크기들을 사용하여 채도 블록 경계 지도를 생성하고 상기 휘도에 대해 계산된 차이 크기들을 사용하여 휘도 블록 경계 지도를 생성하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 23

제8항에 있어서,

정규화된 합들을 사용하여 상기 블록 노이즈의 노이즈 특성들의 집합을 결정하는 단계 및 차이 크기 데이터를 사용하여 상기 이미지의 이미지 특성들의 집합을 결정하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 24

제23항에 있어서,

상기 노이즈 특성들의 집합은 블록 노이즈의 세기 및 블록 크기를 포함하고,

상기 이미지 특성들의 집합은 상기 이미지 내의 디테일( $\text{detail}$ )의 양을 포함하는 블록 노이즈 검출 방법.

#### 청구항 25

제23항에 있어서,

상기 노이즈 및 이미지 특성들의 집합을 사용하여 블록 노이즈 필터에 대한 이득값을 결정하는 단계를 더 포함하는 블록 노이즈 검출 방법.

#### 청구항 26

이미지 내 블록 노이즈 검출 시스템으로서,

상기 이미지 내의 인접 화소들에 대한 화소값들의 차이 크기( $\text{difference magnitude}$ )들을 계산하는 수단;

감소된 차이 크기들의 집합을 생성하기 위해서 상기 이미지 내의 특징( $\text{feature}$ )들로부터 비롯된 차이 크기들로부터의 차이 크기들의 집합을 제거하는 수단;

평균 차이 크기들의 집합( $\text{averaged set of difference magnitude}$ )을 생성하기 위해서 상기 감소된 차이 크기들의 집합을 평균하는 수단;

필터링된 차이 크기들의 집합을 생성하기 위해서 평균된 차이 크기들의 집합에 중간값 필터( $\text{median filter}$ )를 적용하는 수단;

차이 크기들의 결과 집합( $\text{result set of difference magnitude}$ )을 생성하기 위해서 상기 평균된 차이 크기들의 집합으로부터 상기 필터링된 차이 크기들의 집합을 감산하는 수단;

모서리 화소 위치( $\text{edge pixel location}$ )들을 검출하기 위해서 상기 이미지에 모서리 검출을 수행하는 수단; 및

억제된 차이 크기들의 집합을 생성하기 위해서 상기 모서리 화소 위치들에 해당하는 차이 크기들의 결과 집합에서 크기들을 억제하는 수단;

를 포함하는 블록 노이즈 검출 시스템.

#### 청구항 27

삭제

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

청구항 31

삭제

청구항 32

삭제

청구항 33

삭제

청구항 34

삭제

청구항 35

삭제

청구항 36

삭제

청구항 37

삭제

청구항 38

삭제

청구항 39

삭제

청구항 40

삭제

청구항 41

삭제

청구항 42

삭제

청구항 43

삭제

청구항 44

삭제

청구항 45

삭제

청구항 46

삭제

청구항 47

삭제

청구항 48

삭제

## 발명의 설명

### 기술 분야

[0001] 본 출원은 그 내용이 본 명세서에 명시적으로 참고 문헌으로서 포함되는 2009년 2월 10일에 출원된 미국 임시 특허 출원 제61/151,434호(발명의 명칭: "BLOCK NOISE DETECTION AND FILTER")에 대한 우선권을 주장한다.

### 배경 기술

[0002] 이미지, 오디오, 또는 비디오 콘텐츠와 같은 디지털 콘텐츠의 압축은 압축된 디지털 콘텐츠에 대한 아티팩트(artifact)를 도입할 수 있다. 도입되는 아티팩트의 유형 및 정도는 데이터 유형 및 압축 기법에 좌우된다. 압축 아티팩트는 전형적으로 DVD와 같은 매체, MPEG-1, MPEG-2, MPEG-4, JPEG 등과 같은 압축된 비디오 형식에서 발생한다. 특히, 디지털로 압축된 비디오 또는 이미지 파일에 블록 노이즈(block noise)가 존재할 수 있다.

### 도면의 간단한 설명

[0003] 도 1은 블록 노이즈가 어떻게 디지털 이미지 및/또는 비디오에 도입되는지와 블록 노이즈 필터가 블록 노이즈 검출기에 의해 결정되는 제어 파라미터를 사용하여 어떻게 적용되는지를 도시하는 블록도.

도 2는 블록 노이즈 검출기 및/또는 블록 노이즈 필터를 갖는 이미지 또는 비디오 프로세서를 갖는 시스템의 예를 도시하는 블록도.

도 3은 블록 노이즈 검출기 및/또는 블록 노이즈 필터를 갖는 이미지 또는 비디오 프로세서를 갖는 광학 디스크 장치의 예를 도시하는 블록도.

도 4는 블록 노이즈 검출기 내의 구성요소의 예를 도시하는 블록도.

도 5는 블록 노이즈 필터 내의 구성요소의 예를 도시하는 블록도.

도 6은 행 또는 열에 걸친 차이 크기의 평균의 예를 도시하는 도면.

도 7은 평균 차이 크기에 적용되는 중간값 필터의 출력의 예를 도시하는 도면.

도 8은 DC 레벨이 제거된 평균 차이 크기의 예를 도시하는 도면.

도 9는 DC 레벨이 제거되고 모서리 스파이크(edge spike)가 억제된 평균 차이 크기의 예를 도시하는 도면.

도 10은 다양한 블록 간격 및 오프셋(offset)을 갖는 정규화 합산 차이 크기의 예를 도시하는 도면.

도 11은 스케일링(scale)된 이미지에 대한 평균 차이 크기의 예를 도시하는 도면.

도 12는 스케일링된 이미지에 대한 다양한 블록 간격 및 오프셋에서의 정규화 합산 차이 크기의 예를 도시하는

도면.

도 13은 인접하는 0이 아닌 구간값들이 합산되는, 스케일링된 이미지에 대한 다양한 블록 간격 및 오프셋에서의 정규화된 합산 차이 크기의 예를 도시하는 도면.

도 14는 행 또는 열에 걸친 평균 차이 크기의 예를 도시하는 도면.

도 15는 다중 화소 폭을 갖는 피크를 나타내는, 가변 오프셋 및 고정 간격에서의 정규화된 합산 차이 크기의 예를 도시하는 도면.

도 16a는 인접하는 0이 아닌 구간값들이 합산되는, 가변 오프셋 및 고정 간격에서의 정규화된 합산 차이 크기의 예를 도시하는 도면.

도 16b는 도 15 및 16a의 합산 차이 크기의 조작으로부터 획득되는 값의 예를 갖는 테이블.

도 17 및 18은 스케일링의 효과가 있고 없는 경우에 블록 노이즈 필터를 노이즈 경계에 적용하는 효과의 예를 도시하는 도면.

도 19는 블록 노이즈 검출을 위한 예시 프로세스의 흐름도.

도 20은 블록 노이즈 필터에 대한 제어 파라미터를 선택하기 위한 예시 프로세스의 흐름도.

도 21은 블록 노이즈 필터를 이미지에 적용하는 흐름도.

도 22는 본 명세서에서 논의되는 방법들 중 임의의 하나 이상을 머신이 실행하게 하기 위한 명령어들의 집합이 실행될 수 있는 컴퓨터 시스템의 예시 형태로 머신을 표현한 도표.

### 발명을 실시하기 위한 구체적인 내용

[0004] 아래의 설명 및 도면은 예시적이며, 한정적으로 해석되지 않아야 한다. 본 개시 내용의 철저한 이해를 제공하기 위해 다수의 특정한 세부사항이 기술된다. 그러나, 어떤 경우에는 공지된 또는 종래의 세부 사항은 설명을 불명료하게 하지 않기 위해 기술되지 않는다. 본 개시 내용의 일 실시예를 언급하는 것은 동일한 실시예를 언급하는 것일 수 있으나 반드시 그러하지는 않으며, 이러한 언급은 실시예들 중 적어도 하나를 의미한다.

[0005]본 명세서에서 "일 실시예" 또는 "실시예"를 언급하는 것은 그 실시예와 관련하여 기술된 특정한 특징, 구조, 또는 특성이 본 개시 내용의 적어도 하나의 실시예에 포함됨을 의미한다. 본 명세서 내의 다양한 위치에서 "일 실시예에서"라는 문구가 나타나는 것은 반드시 모두 동일한 실시예를 언급하고 있는 것은 아니며, 다른 실시예들과 상호 배타적인 별개의 또는 대안적인 실시예인 것도 아니다. 더욱이, 일부 실시예들에 의해 표현될 수 있고 다른 실시예들에 의해서는 표현될 수 없는 다양한 특징이 기술된다. 유사하게, 일부 실시예들에 대해 요건될 수 있지만 다른 실시예들에 대해서는 그렇지 않을 수 있는 다양한 요건이 기술된다.

[0006]본 명세서에서 사용되는 용어는 일반적으로 본 개시 내용의 맥락 내에서, 그리고 각 용어가 사용되는 특정한 맥락 내에서 본 기술 분야에서의 통상적인 의미를 갖는다. 본 개시 내용의 설명에 관한 추가적인 안내를 실시자에게 제공하기 위해, 본 개시 내용을 기술하는 데 사용되는 소정의 용어들이 아래에서, 또는 본 명세서의 다른 곳에서 논의된다. 편의상, 소정의 용어들은 예컨대 이탤릭체 및/또는 인용 부호를 사용하여 강조될 수 있다. 강조를 사용하는 것은 용어의 범위 및 의미에 영향을 미치지 않는다. 용어의 범위 및 의미는 동일한 맥락 내에서 강조 여부에 관계없이 동일하다. 동일한 대상이 둘 이상의 방식으로 서술될 수 있음을 이해할 것이다.

[0007]결과적으로, 대안적인 언어 및 동의어가 본 명세서에서 논의되는 용어들 중 임의의 하나 이상에 대해 사용될 수 있으며, 용어가 본 명세서에서 상술되거나 논의되는지 여부에 대해 어떠한 특별한 중요성도 부과되지 않는다. 소정의 용어에 대한 동의어가 제공된다. 하나 이상의 동의어를 열거하는 것은 다른 동의어들을 사용하는 것을 배제하지 않는다. 본 명세서의 임의의 위치에서 본 명세서에서 논의되는 임의의 용어에 관한 예시를 포함하는 예시를 사용하는 것은 예시적일 뿐이며, 본 개시 내용 또는 임의의 예시된 용어의 범위 및 의미를 더 한정하기 위한 것이 아니다. 마찬가지로, 본 개시 내용은 본 명세서에서 주어지는 다양한 실시예로 한정되지 않는다.

[0008]본 개시 내용의 범위를 더 한정하려는 의도 없이, 본 개시 내용의 실시예들에 따른 기구, 장치, 방법 및 이들의 관련된 결과에 관한 예시가 아래에서 주어진다. 예시에서 표제 또는 부제가 독자의 편의를 위해 사용될 수 있으나, 결코 본 개시 내용의 범위를 한정하는 것이 아님에 주목한다. 달리 정의되지 않는 한, 본 명세서에서 사용되는 모든 기술 및 과학 용어는 본 개시 내용과 관련된 기술 분야의 당업자에 의해 통상적으로 이해되는 것과

동일한 의미를 갖는다. 분쟁의 경우, 정의를 포함하는 본 문서가 대조될 것이다.

- [0009] 본 개시 내용의 실시에는 블록 노이즈 검출 및 필터링을 위한 시스템 및 방법을 포함한다.
- [0010] 도 1은 블록 노이즈(106)가 어떻게 디지털 이미지 및/또는 비디오(102)에 도입되는지와 블록 노이즈 필터(154)가 블록 노이즈 검출기(152)에 의해 결정되는 제어 파라미터를 사용하여 어떻게 적용되는지를 표시하는 블록도(100)를 도시한다.
- [0011] 전형적인 이미지 또는 비디오 압축 방식은 디지털 이미지/비디오(102)에 블록 노이즈와 같은 아티팩트를 도입시킨다. 특히, 손실 압축 방식을 적용함으로써, 블록 노이즈가 이미지/비디오(106)에 도입될 수 있다.
- [0012] 본 명세서에서 기술되는 기법은 영향을 받은 디지털 이미지/비디오(106)를 분석 및 처리하고 블록 노이즈의 속성/특성{예컨대 블록 크기, 오프셋, 경계 위치, 블록 노이즈의 세기(예컨대 블록 노이즈의 강도)} 및/또는 이미지 특성(예컨대 이미지 내의 디테일의 양)을 식별하는 블록 노이즈 검출기(152)와 관련된다.
- [0013] 블록 노이즈 검출기(152)는 블록 노이즈 및/또는 이미지 특성을 사용하여 블록 노이즈 필터(154)에 대한 제어 파라미터(108)를 생성할 수 있다. 제어 파라미터(108)는 블록 노이즈 필터 이득값, 모서리 세기 승수(edge strength multiplier) 등을 포함할 수 있다. 블록 노이즈 아티팩트의 가시도를 감소 또는 제거하기 위해 블록 노이즈 필터(154)가 원래의 이미지에 대해 작용할 수 있다.
- [0014] 일 실시예에서, 디지털 압축을 통해 도입된 블록 노이즈는 (대략) 고정된 구간으로 이격된 이미지 내의 세로만의 또는 가로만의 모서리를 식별함으로써 검출될 수 있다. 검출은 수평 및 수직 휘도(luma) 화소 차이를 분석하여 규칙적으로 이격된 피크를 검출함으로써 수행될 수 있다. 이러한 검출에 기초하여, 블록 노이즈 경계의 위치를 가리키는 수평 및 수직 맵이 생성된다.
- [0015] 검출된 휘도 블록 크기로부터 채도(chroma) 블록 크기가 추론될 수 있고, 채도 화소 차이 데이터를 사용하여 채도에 대한 별개의 경계 지도가 만들어진다. 검출 전에 해상도가 스케일링된 이미지가 또한 크기 및 오프셋을 검출하기 위한 수정된 방식을 사용하여 수용되고, 이러한 스케일링으로 인한 블록 경계의 흐려짐에 대한 대비가 블록 경계 지도에서 이루어진다. 이러한 블록 경계 지도는 지도 내용에 기초하여 이미지 내의 블록 노이즈를 필터링하는 노이즈 필터 기능에 전달될 수 있다.
- [0016] 일반적으로, 블록 노이즈 검출은 이미지 프레임의 전체 또는 상당 부분에 대해 수행되고, 검출의 결과는 전형적으로 전체 프레임 또는 대부분의 프레임이 분석될 때까지 완전하지 않다. 이는 분석에 더 많은 데이터를 포함 시킴으로써 검출의 통계적인 신뢰성을 향상시킨다. 비디오 소스의 경우, 검출 결과는 프레임의 끝까지 이용 가능하지 않을 수 있고, 그럼으로써 검출 프로세스에서 1 프레임의 지연을 초래한다. 따라서 블록 노이즈 필터 동작은 통상적으로 다음의 프레임 기간에 적용될 것이다. 블록 노이즈 경계의 검출은 시간적인 양상 없이 공간적일 뿐이다. 즉, 이는 비디오 프레임/필드 시퀀스 내에서 각각의 단일 이미지 프레임에 대해 개별적으로 수행된다.
- [0017] 수평 및 수직 블록 경계의 위치 외에도, 이러한 기능은 또한 블록 노이즈 필터 기능에 대한 제어 파라미터를 자동으로 결정한다. 이러한 파라미터는 수평 및 수직 필터의 세기 및 이미지 내의 물체의 모서리를 따라 노이즈 및 필터 아티팩트를 억제하기 위해 필터에 의해 사용되는 수평 및 수직 경사(gradient) 승수를 포함한다.
- [0018] 블록 노이즈를 검출하고 제어 파라미터를 계산하기 위한 프로세스가 도 6 내지 16 및 도 19 내지 20과 연관된 설명을 더 참조하여 보다 상세히 기술된다.
- [0019] 도 2는 블록 노이즈 검출기(252) 및/또는 블록 노이즈 필터(254)를 갖는 이미지 또는 비디오 프로세서(250)를 갖는 시스템(200)의 예를 도시하는 블록도를 나타낸다.
- [0020] 시스템(200)은 처리 유닛(202), 메모리(204) 및/또는 이미지/비디오 프로세서(250)를 포함한다. 일부 실시예에서, 위에서 언급된 모듈/장치 또는 이를 나타내는 기능 중 임의의 것은 그 전체 또는 일부가 시스템(200)의 외부에 있을 수 있다.
- [0021] 시스템(200)은 일반적으로 인터넷 콘텐츠(스트리밍되고/되거나 국지적으로 저장됨), 디지털 이미징 장치에 의해 기록된 콘텐츠(예컨대 캠코더, 카메라 등), 사용자 생성 콘텐츠, 저장 매체 상에 저장된 콘텐츠(예컨대 VHS, 자기 매체 또는 광학 매체) 등을 포함하지만 이에 한정되지 않는 다양한 소스로부터 유래하는 디지털 이미지, 디지털 비디오 및/또는 디지털 오디오를 수신 및 처리할 수 있다. 호환 가능한 유형의 광학 매체 형식은 CD, DVD, HD-DVD, Blu-ray 및 HDV를 포함할 수 있지만, 이에 한정되지 않는다. 호환 가능한 광학 디스크는 또한

ROM(Read Only Memory), WORM(Write Once-Read Many), I(Interactive) 및/또는 E(Erasable)일 수 있다. 블록 노이즈를 갖는 비디오의 추가적인 소스는 예컨대 디지털 방송 텔레비전, 위성 TV, 디지털 케이블 TV 등을 포함한다.

- [0022] 시스템(200)의 일 실시예는 메모리(204)에 결합된 처리 유닛(202)을 포함한다. 처리 유닛(202)은 시스템(200)에 의해 사용되는 명령어를 실행할 수 있는 소프트웨어 에이전트(software agent) 및/또는 하드웨어 구성요소의 임의의 조합일 수 있다. 명령어는 메모리(204)와 통신함으로써 액세스될 수 있다.
- [0023] 일 실시예에서, 처리 유닛(202)은 내부 및/또는 외부 요청을 처리하고, 필요한 계산을 수행한다. 처리 유닛(202)은 메모리(204)와 통신할 수 있으며, 메모리(204)는 일부 실시예에서 처리 유닛(202) 및/또는 이미지 프로세서(250)에 의해 실행될 수 있는 명령어의 시퀀스를 저장한다.
- [0024] 메모리(204)는 이미지 프로세서(250)에 대한 데이터 입력/출력 버퍼로서 작용하는 소프트웨어 에이전트 및/또는 하드웨어 구성요소의 임의의 조합일 수 있다. 메모리(204) 내의 데이터는 시스템(200)의 내부 및/또는 외부에 있는 임의의 주체로부터 수신될 수 있다.
- [0025] 시스템(200)의 일 실시예는 메모리(204)에 결합될 수 있는 이미지/비디오 프로세서(250)를 포함한다. 이미지 프로세서(250)는 예컨대 블록 노이즈 검출 및 블록 노이즈 필터링을 포함하는 이미지 및/또는 비디오 처리를 수행할 수 있는 소프트웨어 에이전트 및/또는 하드웨어 구성요소의 임의의 조합일 수 있다. 이미지 프로세서(250)의 일 실시예는 블록 노이즈 검출기(252) 및/또는 블록 노이즈 필터(254)를 포함한다. 디지털 이미지/비디오는 메모리(204)로부터 수신 또는 인출될 수 있다.
- [0026] 또한, 도 6 내지 16 및 도 19 내지 20의 예를 더 참조하여 기술되는 블록 노이즈 검출 및/또는 블록 노이즈 필터링에 대해, 이미지 프로세서(250)에 의해 적용될 수 있는 프로세스는 예컨대 기하학적 변형(예컨대 확대, 스케일링, 축소, 회전 등), 색 조정, 밝기 조정, 대비 조정, 양자화, 상이한 색 공간으로의 변환, 디지털 합성, 광학적 합성, 보간(interpolation), 에일리어스(alias) 감소, 필터링, 이미지 편집, 이미지 등록, 이미지 안정화, 이미지 분할 등을 포함하지만, 이에 한정되지 않는다.
- [0027] 블록 노이즈 검출기(252)의 예시 구성요소/모듈이 도 4의 예를 더 참조하여 예시되고, 블록 노이즈 필터의 예시 구성요소가 도 5의 예를 더 참조하여 예시된다.
- [0028] 도 3은 블록 노이즈 검출기(352) 및/또는 블록 노이즈 필터(354)를 갖는 이미지 또는 비디오 프로세서(350)를 갖는 광학 디스크 장치(300)의 예를 도시하는 블록도를 나타낸다.
- [0029] 매체 재생기{예컨대 광학 디스크 장치(300)}는 광학 디스크 판독기(302), 처리 유닛(304), 메모리 유닛(306) 및/또는 이미지/비디오 프로세서를 포함할 수 있다. 일부 실시예에서, 위에서 언급된 모듈/장치 또는 이를 나타내는 기능 중 임의의 것은 그 전체 또는 일부가 매체 재생기{예컨대 광학 디스크 장치(300)}의 외부에 있을 수 있다. 대안적인 실시예에서, 광학 디스크 장치 이외의 매체 재생기의 경우, 광학 디스크 판독기(302)는 본 개시 내용의 신규한 기술로부터 벗어나지 않고 임의의 다른 적합한 매체 판독기로 대체될 수 있음을 이해한다. 다른 유형의 적합한 매체 판독기는 공지된 및/또는 편리한 수단으로 동작할 수 있다.
- [0030] 광학 디스크 판독기(302)는 광학 디스크 상에 빛을 비추고/비추거나, 광학 디스크로부터 반사되는 빛을 검출하고/하거나, 검출된 빛을 전기 신호로 변환할 수 있는 소프트웨어 에이전트 및/또는 하드웨어 구성요소의 임의의 조합일 수 있다. 광학 디스크 판독기(302)는 CD, DVD, HD-DVD, Blu-ray 및 HDV를 포함하지만 이에 한정되지 않는 임의의 광학 디스크를 판독할 수 있다. 호환 가능한 광학 디스크는 또한 ROM(Read Only Memory), WORM(Write Once-Read Many), I(Interactive) 및/또는 E(Erasable)일 수 있다.
- [0031] 일 실시예에서, 광학 디스크 장치(300)는 광학 디스크 판독기(302)에 결합된 메모리(306)를 포함한다. 메모리(306)는 광학 디스크 판독기(302)에 대한 데이터 입력/출력 버퍼로서 작용하는 소프트웨어 에이전트 및/또는 하드웨어 구성요소의 임의의 조합일 수 있다. 광학 디스크 장치(300)의 일 실시예는 광학 디스크 판독기(302)에 결합된 처리 유닛(304)을 포함한다. 일부 경우에, 처리 유닛(304)은 메모리(306) 및/또는 이미지/비디오 프로세서(350)에 결합된다. 처리 유닛(304)은 예컨대 광학 디스크 장치(300)에 의해 사용되는 명령어를 실행할 수 있는 소프트웨어 에이전트 및/또는 하드웨어 구성요소의 임의의 조합일 수 있다.
- [0032] 일 실시예에서, 처리 유닛(304)은 내부 및/또는 외부 요청을 처리하고 필요한 계산을 수행한다. 처리 유닛(304)은 메모리(406)와 통신할 수 있으며, 메모리(406)는 일부 실시예에서 처리 유닛(304) 및/또는 이미지/비디오 프로세서(350)에 의해 실행될 수 있는 명령어의 시퀀스를 저장한다. 이미지/비디오 프로세서(350)는

이미지, 비디오 처리 및/또는 디지털 오디오 처리를 수행할 수 있는 소프트웨어 에이전트 및/또는 하드웨어 구성요소의 임의의 조합일 수 있다.

- [0033] 일 실시예에서, 프로세서(350)는 블록 노이즈 검출 및/또는 필터링과 연관된 작업의 일부 또는 전부를 수행하기 위한 블록 노이즈 검출기(352) 및/또는 블록 노이즈 필터(354)를 포함한다. 블록 노이즈 검출기(252)의 예시 구성요소/모듈이 도 4의 예를 더 참조하여 예시되고, 블록 노이즈 필터의 예시 구성요소가 도 5의 예를 더 참조하여 예시된다.
- [0034] 도 4는 블록 노이즈 검출기(452) 내의 구성요소의 예를 도시하는 블록도를 나타낸다.
- [0035] 블록 노이즈 검출기(452)는 차이 크기(difference magnitude; dMag) 생성기(402), dMag 프로세서(404), 가변 간격/고정 간격 모듈 및 스케일링 모듈(408)을 갖는 dMag 합산기(406), 블록 크기 및 오프셋 평가기(410), 경계 지도 생성기(412), 필터 이득 생성기(414) 및/또는 모서리 경사 승수 생성기(416){예컨대 이들은 필터 제어 파라미터 생성기(420)의 구성요소일 수 있음}를 포함한다. 일부 실시예에서, 위에서 언급된 모듈/장치 또는 이를 나타내는 기능 중 임의의 것은 그 전체 또는 일부가 검출기(452)의 외부에 있을 수 있다.
- [0036] 차이 크기(dMag) 생성기(402)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.
- [0037] dMag 생성기(402)는 이미지(예컨대 프레임 또는 비디오의 이미지 프레임) 내의 인접한 화소에 대한 화소값의 차이 크기를 계산할 수 있는 소프트웨어 에이전트 및/또는 하드웨어 구성요소의 임의의 조합일 수 있다. 일 실시예에서, dMag 생성기(402)는 최적화기 및 행/열 평균 생성기를 포함한다.
- [0038] 일 실시예에서, 이미지의 차이 크기(dMag)는 이미지 또는 비디오의 블록 노이즈 검출에서 사용된다. 차이 크기는 생성기(402)가 화소값의 수평 및 수직 1화소 차분(1-pixel differential)을 계산함으로써 결정될 수 있다. 이는 인접한 화소 위치 사이의 차이의 크기이다. 즉, 화소 위치 N에 대해, 차이 크기는 화소 N에서 화소 N+1을 뺀 절대 화소값이다. 차이 크기는 휘도 및 채도 양자에 대해 계산될 수 있다.
- [0039] 계산된 dMag 값은 블록 노이즈를 가리키기 위한 신호 대 노이즈 비(SNR)를 향상시키기 위한 일련의 동작들 중 하나 이상에 의해 최적화될 수 있다(예컨대 dMag 최적화기에 의해). 예컨대, 블록 노이즈로부터 비롯되는 차이 값은 전형적으로 이미지 내의 주요 특징으로부터 비롯되는 차이값에 비해 비교적 작은 크기를 갖는다. 예컨대, 노이즈 블록 사이의 경계로부터 비롯되는 수직 및 수평 라인의 망상선(crosshatch)은 전형적으로 이미지 내의 다른 특징의 윤곽에 비해 희미하다.
- [0040] 큰 크기의 이미지 특징이 블록 노이즈 차이를 제거하여 가능하게는 잘못된 양성 블록 노이즈 경계 식별을 야기하는 것을 방지하게 위해, 소정의 문턱값보다 큰 차이 크기의 값은 그 문턱값으로 또는 근처로 한정될 수 있다. 예컨대, 화소값으로 8 비트 표현을 사용하는 이미지의 경우, 이러한 문턱값은 (8 비트의 최대값인 255 중에서) 10일 수 있다.
- [0041] 또한, 블록 노이즈 경계로 인한 차이 크기의 작은 피크는 전형적으로 1 화소의 폭을 갖는다(일부 경우에 있어서 이는 상향 스케일링, 채도 상향 샘플링의 효과로 인해, 또는 단순히 이미지 화소값으로 인해 약간 더 클 수 있다). 결과적으로, 따라서 이러한 특징은 검출되고 격리될 수 있다.
- [0042] 이를 수행하는 한 가지 방식은 이러한 특징을 배제하는 필터 프로세스를 사용하는 것이다. 이러한 필터의 출력이 이미지로부터 감해지는 경우, 이는 블록 노이즈에 기인하는 차이 크기값과 같은 더 작은 특징만을 남긴다. 이러한 프로세스는 차이 크기 이미지의 신호 대 노이즈 비를 증가시키는데, 이러한 경우 '신호'는 노이즈 블록 사이의 경계의 차이 크기이며, 전형적으로 블록 노이즈의 결과이다.
- [0043] 일 실시예에서, 행/열 평균 생성기는 서로 수직인 방향으로 평균 수평 및 수직 차이 크기를 합산한다(예컨대 수평 차이 크기 내의 행들이 평균되고(수직으로) 수직 차이 크기 내의 열들이 평균된다(수평으로)). 그 결과를 사용하여, 행/열 평균 생성기는 이미지의 폭인 수평 차이 크기의 벡터 및 이미지의 높이인 수직 차이 크기의 벡터를 생성할 수 있다. 평균 프로세스는 규칙적인 블록 경계 특징을 강화하는 경향이 있다. 그러나, 보다 불규칙적으로 배치되는 잔여 이미지 데이터 또는 무작위 노이즈 데이터는 강화되지 않으며, 차이 크기 평균의 배경 레벨이 변화하는 결과를 낳는다.

- [0044] 도 6은 행 또는 열에 걸친 차이 크기(dMag)의 평균의 예를 나타내며, 평균 수평 차이 크기의 플롯(plot)(600)을 도시한다. 각각의 수평 위치에 대한 차이 크기가 수직축(604) 상에 도시되고, 이미지의 폭을 가로지르는 화소 위치가 수평축(602)을 따라 도시된다. 이미지 내의 규칙적으로 이격된 피크(606)는 전형적으로 블록 사이의 수평 경계에 대응한다.
- [0045] 플롯의 평균 또는 'DC' 레벨(608)은 주로 나머지 무작위 노이즈 및 저레벨 이미지 데이터로부터 비롯된다. 이미지(600)의 말단에 있는 큰 스파이크(610)는 원래의 이미지와 이미지의 왼쪽 및 오른쪽에 있는 좁은 수직 흑색 막대 사이의 큰 차이에 의해 야기된다. 예를 위해, 수평 차이 크기 플롯만이 여기에(그리고 아래의 단락들에서) 도시된다. 유사한 결과가 수직 플롯으로 얻어진다.
- [0046] 이러한 방법은 일반적으로 모든 열에 걸친 수직 차이 크기에 대해, 그리고 모든 행에 걸친 수평 크기 차이에 대해 동일한 블록 경계 위치에 대해 수행된다. 블록 크기가 이미지에 걸쳐 현저하게 변하는 경우, 수정된 절차가 수행될 수 있다.
- [0047] 도 6의 예에 도시된 바처럼, 평균 차이 크기 벡터에는 전형적으로 배경 또는 'DC' 레벨(608)이 존재한다. 이는 이전의 처리 단계와 유사한 중간값 필터 접근법을 사용하여 대체로 제거된다. 중간값 필터는 행/열 평균 생성기에 의해 평균 차이 크기에 적용될 수 있고, 평균 차이 크기로부터 결과가 감해진다. 중간값 필터는  $1 \times N$  화소의 영역에 적용되는데, N은 고정되거나 프로그램 가능할 수 있다. 이전의 경우에서와 같이, 경험적으로 7의 N 값이 잘 작용하는 것으로 나타났다.
- [0048] 도 7은 평균 차이 크기에 적용되는 중간값 필터의 출력(706)의 예를 도시한다. {이미지(700) 내의 수직 척도(704)는 플롯의 데이터 내용으로 정규화되고, 따라서 중간값 필터 출력은 이전의 플롯에 비해 과장되게 나타남에 주목한다.}
- [0049] 도 8은 평균 차이 크기로부터 중간값 필터의 출력을 감산함으로써 DC 레벨이 제거된 평균 차이 크기의 예를 도시한다. 배경 또는 'DC' 레벨이 제거되었음을 쉽게 확인할 수 있다.
- [0050] 도 8의 예는 원래 이미지의 흑색 경계로 인해 평균 차이 크기의 모서리에 여전히 스파이크(810)가 존재함을 나타낸다. 이러한 경계는 영상비(aspect ratio) 정정{즉 레터박스(letterbox) 또는 필러박스(pillarbox)}에 의해 야기될 수 있거나, 또는 때때로 이미지 모서리에 있는 작은 흑색 띠일 뿐이다. 후속 단계들은 평균 차이 크기를 분석하여 데이터 내의 피크로 표현되는 블록 경계의 크기 및 위치를 결정할 것이다. 그러나, 이러한 매우 큰 스파이크의 존재는 그 크기로 인해 블록 크기 및 위치의 분석을 벗어날 수 있다. 그 결과, 블록 크기 및 위치의 검출 전에 이러한 스파이크를 제거하거나 그 크기를 현저히 감소시키는 것이 유익하다.
- [0051] 이러한 스파이크를 처리하는 하나의 방법은 단순히 차이 크기 벡터의 모서리 근처의 모든 값을 억제하는 것이다. 이러한 방법이 종종 잘 작용하지만, 이는 이미지의 실제 모서리가 어디에 위치하는지에 관해 알고 있다고 가정한다. 그러나, 이는 항상 신뢰성 있게 알려지지 않는다. 여기에서 사용되는 기법은 벡터 내의 처음 몇몇 최대값을 식별하고, 이러한 최대값들의 크기를 비교하며, 비교 결과에 기초하여 가장 큰 피크들의 진폭을 감소시키는 것이다.
- [0052] 평균 차이 크기에서 중간값을 뺀 것에 알고리즘을 적용(예컨대 행/열 평균 생성기에 의해)한 결과가 도 9의 예에 도시된다. {이미지 내의 수직 척도(904)는 플롯 데이터 내용으로 정규화되고, 따라서 큰 피크의 감소는 전체 데이터 집합에 대해 척도가 보다 균일해지는 결과를 낳음에 주목한다.}
- [0053] 검출기(404)의 일 실시예는 dMag 프로세서(404)를 포함한다. dMag 프로세서(404)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.
- [0054] dMag 프로세서(404)는 이미지 또는 비디오 프레임 내의 블록 노이즈를 가리키는 SNR을 증가시키기 위해 차이 크기 데이터를 분석, 처리 및/또는 필터링할 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있다.
- [0055] 일 실시예에서, dMag 프로세서(404)는 블록 노이즈에 대응하는 특징을 격리시키기 위한 필터를 구현한다. 예컨대, 중간값 필터가 특징을 격리시키는 데 사용될 수 있다. 중간값 필터, 또는 다른 유사한 필터는 작은 영역에서 중간값 화소를 선택할 수 있다. 수평 차이 크기에 대해, 프로세서(404)는  $1 \times N$  화소의 수평 방향 영역에 중

간값 필터를 적용할 수 있다. 수직 차이 크기에 대해, 프로세서(404)는  $N \times 1$  화소의 수직 방향 영역에 중간값 필터를 적용할 수 있다. 이 경우  $N$ 은 고정되거나 프로그램 가능할 수 있다. 예컨대, 7의  $N$  값이 사용될 수 있는데, 이는 가장 흔한 블록 크기가 8 화소이기 때문이라는 점에 주로 기인한다. 일 실시예에서, 프로세서(404)는 필터링되지 않은 차이 크기로부터 중간값 필터의 출력을 감산하는데, 0 미만의 값은 0으로 클리핑(clip)된다. 이러한 프로세스는 이미지 특징의 대부분을 제거할 수 있고, 따라서 노이즈 블록 사이의 경계의 검출을 위한 신호 대 노이즈 비를 증가시킬 수 있다.

[0056] 일 실시예에서, dMag 프로세서(404)는 이미지 내의 물체 모서리 주위의 차이 크기 값을 억제할 수 있다. 차이 크기로부터 중간값을 감한 후에도, 원래 이미지로부터의 물체 모서리의 나머지가 여전히 존재한다. 이는 원래 이미지 내의 모서리를 검출하고 그 위치에서 차이 크기를 억제함으로써 더 감소될 수 있다.

[0057] 일 실시예에서, 프로세서(404)는 예컨대 수직 및 수평 필터{예컨대  $3 \times 3$  소벨(Sobel) 필터}를 원래 이미지의 휘도 부분에 적용한 후 각 결과의 절대값을 취하여 수직 및 수평 경사 크기를 산출함으로써 모서리 검출을 수행한다. 소벨 필터는 예컨대 아래와 같다.

$$\begin{aligned} \text{HorSobelFilter3x3} &= \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}; \\ \text{VertSobelFilter3x3} &= \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}; \end{aligned}$$

[0058] (필터의 출력은 4로 나눔으로써 정규화되어야 함에 주목한다.)

[0059]

[0060] 경사 크기는 문턱값과 비교되고, 차이 크기는 경사 크기가 문턱값을 초과하는 위치에서 0으로 설정된다. 수평 및 수직 차이 크기가 적절한 경사 크기를 사용하여 별개로 처리된다. 전형적인 문턱값은 8 비트 경사 크기 표현에 대해 20으로 설정될 수 있다. 이러한 프로세스는 차이 크기 데이터의 신호 대 노이즈 비를 더 향상시킨다.

[0061] 검출기(404)의 일 실시예는 dMag 합산기(406)를 포함한다. dMag 합산기(406)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.

[0062] dMag 합산기(406)는 고정 및/또는 가변 구간/간격 합을 생성할 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있다.

[0063] 합산기(406)는 가변 간격 및 오프셋에서 차이 크기를 합산하여 노이즈 블록의 크기 및 그 위치를 식별할 수 있다. 합산기(406)는 데이터의 시작으로부터 소정 범위의 오프셋에서 소정 범위의 구간 간격으로 차이 크기 벡터를 가로질러 균일하게(또는 실질적으로 균일하게) 이격되는 차이 크기의 정규화된 합을 계산할 수 있다.

[0064] 구간 크기는 최소값으로부터 최대값까지의 블록 크기 범위(MinSize:MaxSize)일 수 있고, 오프셋은 최소값으로부터 최대값까지의 시작 위치 범위(MinOffset:MaxOffset)일 수 있다. 구간 크기는 블록의 크기를 식별하고, 한편으로 오프셋은 블록의 위치를 식별한다. 합산기(406)는 블록 크기의 범위 내의 각 항목에 대해, 그리고 오프셋의 범위 내의 각 항목에 대해 개별적으로 차이 크기 데이터를 합산할 수 있다. 이후 합산기(406)는 차이 크기 벡터 내의 위치의 개수에 걸쳐 존재할 수 있는 각 크기의 블록의 개수에 기초하여 합을 정규화할 수 있다.

[0065] 모든 구간 크기 및 오프셋에 대한 합이 계산되면, 모든 합의 평균이 합산기(406)에 의해 계산된다. 이는 SNR을 향상시키기 위해 구간/오프셋 합 각각으로부터 감해진다. 음의 결과는 0으로 클리핑될 수 있다.

[0066] 차이 크기 벡터 내의 블록 경계 피크의 실제 위치와 정렬되는 구간 크기 및 오프셋이 전형적으로 최대의(또는 근사 최대의) 합을 갖는다. 이는 그 크기 구간의 정수 배수에 대해서도 마찬가지일 것이다. 이러한 구간 크기

(그리고 그 배수)는 또한 이들이 차이 크기 벡터 내의 블록 경계 피크와 정렬되므로 구간 크기의 오프셋에서 반복된다.

- [0067] 도 10은 다양한 블록 간격 및 오프셋에서의 정규화 합산 차이 크기의 예를 나타내며, 수평 차이 크기 벡터에 대한 결과적인 구간/오프셋 합을 표시한다. 축(1002)은 오프셋값을 표시하고, 축(1004)은 정규화 합산 차이 크기의 진폭을 표시하며, 상이한 구간 크기에 대해 플롯(1000)에 도시된 상이한 곡선들이 존재한다.
- [0068] 합산기(406)는 이 예에서 4 내지 16의 모든 구간 크기에 대해, 그리고 1 내지 32의 오프셋에 대해 합을 계산할 수 있다. 이 플롯에서 가장 큰 진폭값은 8 및 16의 블록 크기에 대한 것이다. 두 크기 모두 매 8 화소에서 반복되는 피크를 가지며, 블록 크기가 8일 가능성이 높음을 가리킨다. 처음 값의 오프셋은 8이며, 제1 블록 경계가 8번째 화소 위치에 또는 그 근처에 위치함을 가리킨다(이는 제1 블록이 이미지의 왼쪽 모서리에 정렬되는 8 화소 폭의 블록에 대해 예상됨).
- [0069] 블록 노이즈 검출 전에 이미지가 스케일링되는 경우, 블록 사이의 뾰족한 경계가 복수의(예컨대 두 개의) 화소에 걸쳐 스미어링(smear)될 수 있다. 이는 대부분의 스케일러(scaler)에 내재하는 저역 통과 필터 및 이미지 내의 화소의 개수 — 그리고 따라서 블록 내의 화소의 개수 — 가 변한다는 사실에 기인하여 발생한다. 그 결과, 차이 크기 벡터에서 검출된 블록 경계 피크는 단일 화소 폭의 피크가 아닐 수 있지만, 2 화소 이상의 폭을 가질 수 있다. 도 11은 스케일링된 이미지에 대한 평균 차이 크기의 예를 도시하고, 발생할 수 있는 피크 스미어링을 도시한다.
- [0070] 플롯(1100)은 2/3의 스케일링 비율로 하향 스케일링된 이미지의 수직 차이 크기의 확대된 부분이다. 그 결과, 원래의 8 화소 블록을 3개 포함하는 24 화소가 16 화소로 스케일링되어 3개의 5⅓ 화소 블록(평균적으로)을 낳는다. 도 11의 예의 플롯(1100)에서, 하향 스케일링 프로세스에서의 블록 경계의 스미어링으로 인해 매 3번째 피크는 실제로 2 화소 폭이라는 점이 확인될 수 있다.
- [0071] 이전 단계에서 기술되는 구간/오프셋 합을 계산하기 위해 플롯(1100)에서 도시되는 크기 차이 벡터가 처리되는 경우, 스미어링된 블록 경계는 2 화소 이상의 폭을 가질 수 있는 구간 피크를 낳는다. 도 12는 스케일링된 이미지에 대한 다양한 블록 간격 및 오프셋에서의 정규화 합산 차이 크기의 예를 도시한다.
- [0072] 예시 플롯(1200)에서, 최대 진폭을 갖는 구간 크기는 16이고, 이는 3개의 5⅓ 화소 블록의 그룹에 대응한다. 2 화소 폭의 구간 피크의 어느 부분이 다음 피크까지의 거리를 결정하는 데 사용되는지에 따라, 구간-16 피크 사이의 오프셋 간격은 5 화소와 6 화소 사이에서 변한다. 더 작은 진폭 구간은 또한 유사한 성질을 보인다. 예컨대, 8 화소 구간은 또한 명확히 스미어링된 구간 피크를 갖는다. 이러한 스미어링된 구간 피크는 피크 위치에 불확실성을 더하고 구간 피크 진폭을 감소시킴으로써 블록 크기가 실제로 얼마인지를 결정하기가 더 어렵게 만든다(공칭 오프셋에 대한 값이 둘 이상의 위치에 걸쳐 퍼져 있기 때문임).
- [0073] 이러한 문제를 우회하는 하나의 방법은 스미어링된 구간 피크를 하나의 값으로 병합하는 것이다(예컨대 스케일링 관리자(408)에 의해). 합산기(406)의 일 실시예는 스케일링 관리자(408)를 포함한다. 스케일링 관리자(408)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다. 스케일링 관리자(408)는 합산 프로세스를 스케일링된 이미지에 적응시킬 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있다.
- [0074] 예컨대, 스케일링 관리자(408)는 양측이 0의 값에 접하지 않는 구간 피크의 사례를 단일 피크로 병합할 수 있다. 이는 예컨대 피크의 양측에 있는 두 값을 피크값으로 더한 후 경계값을 0으로 설정함으로써 수행될 수 있다. 합산 동작은 인접하는 두 값보다 큰 구간/오프셋값에 대해서만 수행된다.
- [0075] 인접하는 0이 아닌 구간값들을 합산하는 결과의 예가 도 13의 예시 플롯(1300)에 도시된다. 이는 도 12의 플롯(1200)과 동일한 데이터에 기초하지만, 구간값들이 인접하는 0이 아닌 위치들에 대해 합산되었다. 각각의 피크 위치가 이제 더욱 명확히 정의되며, 변화하는 오프셋에 걸쳐 규칙적인 진폭을 갖는다.
- [0076] 이미지의 스케일링이 이루어지지 않은 경우 각각의 피크는 전형적으로 깔끔하고, 0의 값이 양측에 있다. 그 결과, 위에서 기술된 피크 데이터 병합 동작을 수행하는 것은 거의 또는 전혀 영향이 없고, 구간/오프셋 데이터가 본질적으로 변하지 않게 한다. 일반적으로, 병합된 구간/오프셋 데이터 집합은 원래의 구간/오프셋 집합과 동

일하거나 더 나은 결과를 제공하였다.

- [0077] 도 14는 행 또는 열에 걸친 평균 차이 크기의 예를 도시한다.
- [0078] 플롯(1400)은 일견 블록 노이즈를 가리키는 규칙적인 간격의 패턴으로 보이는 피크를 도시한다. 플롯(1400)의 피크 사이의 간격을 면밀하게 조사하면, 이들은 7 또는 8 화소만큼 떨어져 있음이 판명된다. 그러나, 피크 간격의 패턴에는 규칙성이 없는 것으로 보인다.
- [0079] 도 15는 도 14의 예에 도시된 평균 차이 크기 데이터에 대한 다중 화소 폭 피크를 나타내는, 가변 오프셋을 갖는 고정 간격에서의 정규화된 합산 차이 크기의 플롯(1500)의 예를 도시한다.
- [0080] 플롯(1500)은 피크 및 가변 간격이 뒤죽박죽인 것으로 보이고, 명료한 패턴을 보이지 않는다. 그러나, 이러한 플롯으로부터, 병합된 피크 플롯이 조사되어야 함을 가리키는 다중 화소 폭 피크가 존재한다는 점이 명확해진다. 도 16a는 인접하는 0이 아닌 구간값들이 합산되는, 가변 오프셋을 갖는 고정 간격에서의 정규화된 합산 차이 크기의 예를 도시한다.
- [0081] 플롯(1600)에서, 가장 큰 피크값을 갖는 구간 및 가장 큰 두 개의 구간 피크 사이의 15의 간격으로 표시된 바처럼 15의 블록 크기가 존재함을 관찰할 수 있다. 그러나, 15 화소 구간에 대해 플롯 내에 몇몇 다른 피크가 존재하며, 이들은 15의 블록 크기 선택을 불확실하게 할 정도로 큰 진폭을 갖는다. 이는 블록 크기가 15가 아님을 명확히 가리키는, 차이 크기 플롯에 도시된 피크 간격에 의해 야기된다. 또한, 이 예에 대한 구간/오프셋 데이터는 스미어링된 피크를 병합한 후에도 종종 구분 가능한 패턴이 전혀 존재하지 않는 많은 경우보다 실제로 더 잘 작용한다.
- [0082] 그 결과, 이미지의 스케일링이 이미지의 폭 및/또는 높이에 걸쳐 가변 블록 크기를 야기하는 경우에 대해 공칭 블록 크기를 식별하기 위해 다른 방법이 사용될 수 있다.
- [0083] 이러한 유형의 스케일링된 이미지의 한 가지 주요 특성은 블록 크기가 공칭값을 갖지만 사용되는 스케일링 비율에 따라 하나의 블록에서 다음 블록까지  $\pm 1$  화소만큼 변할 수 있다는 점이다. 변분(variation)은 전형적으로 1 화소이고, 공칭 블록 크기로부터 한쪽 방향이다. 따라서, 일 실시예에서, 피크가 합산됨에 따라(예컨대 dMag 합산기(406)를 통해) 구간 크기의 단일 화소 변화량을 사용하여 합산이 수행될 수 있다.
- [0084] 합산기(406)는 시작점을 선택한 후 구간 크기의 범위를 사용하여(즉 이전의 구간/오프셋 계산에서와 같이 MinSize로부터 MaxSize까지) 차이 크기값을 단계적으로 통과할 수 있다. 이후 합산기(406)는 이러한 위치에서 발견되는 값들을 합산할 수 있다. 각각의 구간 크기에 대해, 합산기(406)는 차이 합의 집합을 결정한다. 예컨대, 세 개의 상이한 합이 계산될 수 있는데, 하나는 현재 구간 크기로부터 -1 화소의 가능한 오프셋을 허용하고, 하나는 오프셋을 허용하지 않으며, 하나는 +1 화소의 가능한 오프셋을 허용한다.
- [0085] 값이 합에 더해지는 각각의 위치에서, 허용되는 차이 크기값(예컨대 공칭값 및 '-1' 오프셋에 대하여 공칭값에서 1을 뺀 값)이 조사되고 가장 큰 것이 선택된다. 선택된 위치에서의 값이 그 구간에 대한 합에 더해진다. 그 위치는 또한 시작 위치가 되며, 여기에 다음 위치를 결정하도록 구간 크기가 더해진다. 모든 구간 크기에 대해, 그리고 각각의 구간에 대한 -1, 0 및 +1 화소 오프셋에 대해 이러한 방식으로 차이 크기값이 단계적으로 통과될 수 있다. 합이 계산되면 이를 조사하여 어느 블록 크기를 가리키는지(존재하는 경우)를 결정할 수 있다.
- [0086] dMag 합산기(406)에 의해 수행되는 아래의 단계들은 이 절차를 보다 상세히 기술한다.
- [0087] *수평 가변 구간 프로세스*
- [0088] A) 시작점을 식별한다.
- [0089] 1) 벡터 내의 최대 차이 크기값을 결정한다. 모서리의 16 화소 내에 있는 검색점(무효 국외부(invalid outlier)라고 간주됨)으로부터 제외한다. 막 식별된 최대값의 1/4인 문턱값을 설정한다.
- [0090] 2) 최소 블록 크기(4)의 오프셋에서 시작하여, 문턱값보다 크지만 (무효 국외부로 간주됨) 이전에 결정된 최대값 이하인 위치에 도달할 때까지 값들을 단계적으로 통과한다. 이는 제1 후보 시작 위치이다.
- [0091] 3) 후보 시작 위치를 검증한다. 이는 이것의 양측까지의 값보다 커야 하고(즉 진정한 피크) 또한 문턱값보다 커야 한다. 현재 후보가 검증되지 않는 경우, 유효한 시작 위치가 발견될 때까지 계속 값들을 단계적으로 통과한다. 시작점이 차이 크기 어레이에서 지나치게 깊은 경우(즉 어레이의 끝에 가까운 경우), 차이 크기 데이터

는 식별 가능한 블록 경계를 가리지 않으며, 따라서 모든 합을 0으로 설정한다. 이것이 참일 조건은 ( $Width - 1 - MaxSize - HorizontalStartLocation$ )이 0 이하인 것이다.

[0092] B) MinSize로부터 MaxSize까지의 각각의 구간 크기에 대해 아래를 수행한다.

[0093] 1) 각각의 구간에 대해 -1, 0 및 +1의 오프셋 변분을 허용하면서, 구간값에 의해 시작점에서 시작하여 차이 크기값들을 단계적으로 통과한다. 각각의 구간 및 -1, 0, +1 오프셋 변분에 대한 값들을 합산한다. 이는 아래와 같이 이루어진다.

[0094] -1의 오프셋 변분

[0095] 현재 구간 크기의 증분(increment)을 사용하여 값들을 단계적으로 통과한다. 공칭 위치에서의 값이 공칭 위치에서 1을 뺀 것보다 작으면, 그 위치에서 1을 뺀 위치에서의 값을 -1 오프셋 변분에 대한 구간 합에 더하고, 그렇지 않으면 공칭 위치의 값을 더한다. 식별된 최대값 위치를 다음 단계에 대한 시작점으로 사용한다.

[0096] 0의 오프셋 변분

[0097] 현재 구간 크기의 증분을 사용하여 값들을 단계적으로 통과한다. 오프셋이 0이므로 공칭 구간 크기에서의 값만이 사용된다. 식별된 위치를 다음 단계에 대한 시작점으로 사용한다.

[0098] +1의 오프셋 변분

[0099] 현재 구간 크기의 증분을 사용하여 값들을 단계적으로 통과한다. 공칭 위치에서의 값이 공칭 위치에 1을 더한 것보다 작으면, 그 위치에서 1을 더한 위치에서의 값을 +1 오프셋 변분에 대한 구간 합에 더하고, 그렇지 않으면 공칭 위치의 값을 더한다. 식별된 최대값 위치를 다음 단계에 대한 시작점으로 사용한다.

[0100] 2) 합을 (폭 / 구간 크기)로 나눔으로써 각각의 합을 정규화한다.

[0101] 수직 가변 구간 프로세스

[0102] A) 수평 프로세스를 반복하지만, 이미지 폭 대신에 이미지 높이(이는 수직 벡터 길이임)와 수직 차이 크기 벡터를 사용한다. 수직 블록 경계에 대해, 시작점이 차이 크기 벡터의 끝에 지나치게 가까워 유효한 블록 크기가 존재하지 않을 조건은 ( $Height - 1 - MaxSize - VerticalStartLocation$ )이 0 이하인 것이다.

[0103] 도 16b의 예시 표는 도 15 및 도 16a의 예의 플롯(1500) 및 플롯(1600)에서 사용된 예에 대한 차이 크기 데이터에 대해 이러한 절차를 실행한 결과를 가리킨다.

[0104] 8-화소 구간의 -1 오프셋과 7-화소 구간의 +1 오프셋에 대해 가장 큰 구간 합이 발견된다. 이들은 도 16b의 표에서 요소(1610)로서 식별된다. 이러한 유형의 관계 — 즉 구간 N에 대한 -1 오프셋 및 구간 N-1에 대한 +1 오프셋에 대한 합이 둘 다 크고 유사한 경우 — 는 스케일링된 이미지의 평균 블록 크기가 두 구간 크기 사이에 있음을 가리킨다. 따라서 이 예에서, 블록 크기가 7과 8 화소 사이로 스케일링된 것으로 보인다. 이러한 유형의 경우에 있어서, 더 낮은 구간을 공칭 블록 크기로 사용하는 것이 통상적이다.

[0105] 후속 단계는 이러한 블록 크기가 특정한 블록 경계 위치(불균일한 간격으로 나타남)를 식별하는 데 어떻게 사용되는지를 나타낼 것이다. 또한 값들(1610)과 크기 및 구조가 유사하지만 최대값들보다 구간 크기가 두 배인, 1620으로서 식별되는 값들에 주목한다. 비 가변 오프셋을 사용하는 구간/오프셋 합산 방법과 마찬가지로, 최대값을 생성하도록 합산되는 동일한 데이터 지점들에 2X 구간 크기가 걸쳐 있을 것이기 때문에 이것은 전형적인 패턴이다.

[0106] 가변 오프셋 합은 평가기(410)에 의해 블록 크기를 결정하는 데 사용될 수 있다.

[0107] 검출기(404)의 일 실시예는 블록 크기 및 오프셋 평가기(410)를 포함한다. 블록 크기 및 오프셋 평가기(410)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.

[0108] 평가기(410)는 차이 크기 데이터로부터 블록 노이즈 특성을 결정할 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있다.

- [0109] 예컨대, 합산기(406)에 의해 생성되는 차이 크기의 가변 오프셋 합은 평가기(410)에 의해 블록 크기를 결정하는데 사용될 수 있다. 평가기(410)는 수직 및 수평 합 둘 다에 대한 프로세스를 수행할 수 있고, 아래의 예시 단계에서 상세히 기술된다.
- [0110] A) 제1 및 최후 구간 합 — 즉 MinSize 구간의 -1 오프셋 및 MaxSize 구간의 +1 오프셋 — 은 이들을 인접하는 값으로 설정함으로써 사실상 폐기될 수 있다. 이는 이러한 블록 크기들이 MinSize 내지 MaxSize의 블록 크기 범위(예컨대 4:16) 밖에 있기 때문에 수행된다.
- [0111] B) 최대 차이 크기 합을 포함하는 가변 오프셋값 및 구간 크기를 결정한다.
- [0112] C) 최대 합이 0이 아닌지를 검사한다(이러한 조건은 구간 시작 위치가 오른쪽/아래 이미지 모서리에 지나치게 가까운 경우에 발생하며, 이전 단계에서 모든 합이 0으로 설정되는 결과를 낳는다). 0인 경우, 블록 크기를 0으로 설정하고(이는 유효한 블록 크기가 발견되지 않음을 가리킴) 종료한다.
- [0113] D) 식별된 최대 합을 동일한 구간의 다른 두 개의 가변 오프셋에 대한 합과 비교한다.
- [0114] E) 블록 크기를 아래와 같이 결정한다('현재 구간'은 최대값의 구간을 가리킴).
- [0115] IF 중심(0) 오프셋이 그 구간에 대한 최대값이거나 그 구간에 대한 모든 오프셋이 동일한 경우, THEN 현재 구간이 짝수이고 현재 구간의 절반의 중심 오프셋의 값이 최대값에 매우 가까운(예컨대 7/8) 경우 블록 크기를 현재 구간의 절반으로 설정하며, ELSE 블록 크기를 현재 구간으로 설정한다.
- [0116] ELSE -1 오프셋이 최대값이므로, 다음의 더 낮은 구간값을 조사한다. 이전 단계에서 정규화 프로세스가 수행되는 방식으로 인해 다음의 더 큰 구간의 -1 오프셋이 항상 더 클 것이므로, +1 오프셋은 최대값일 수 없다.
- [0117] IF 다음의 더 낮은 구간 크기의 +1 오프셋이 동일 구간의 0 및 -1 오프셋과 유사하거나(예컨대 15/16) 더 크고, 현재 구간의 +1 오프셋에 가까운(예컨대 3/4) 경우 THEN
- [0118] IF 최대값 구간(N)이 홀수이고 (N+1)/2 구간의 -1 오프셋이 최대값에 가까운(예컨대 7/8) 경우 THEN
- [0119] IF (N-1)/2 구간의 +1 오프셋이 (N+1)/2 구간의 -1 오프셋에 가까운(3/4) 경우 THEN
- [0120] 블록 크기를 (N-1)/2로 설정하고
- [0121] ELSE 블록 크기를 N-1로 설정하며,
- [0122] ELSE IF 최대값 구간(N)이 짝수이고 N/2 구간의 -1 오프셋이 최대값에 가까운(7/8) 경우 THEN
- [0123] IF (N/2)-1 구간의 +1 오프셋이 N/2 구간의 -1 오프셋에 가까운(3/4) 경우 THEN
- [0124] 블록 크기를 (N/2)-1로 설정하고
- [0125] ELSE 블록 크기를 N-1로 설정하며,
- [0126] ELSE 블록 크기를 N-1로 설정하고,
- [0127] ELSE 블록 크기를 N으로 설정한다.
- [0128] 일 실시예에서, 평가기(410)는 또한 합산기(406)에 의해 생성되는 구간/오프셋 합을 조사하여 오프셋 피크 사이의 간격을 결정한다. 예컨대, 평가기(410)는 다양한 구간 및 오프셋에 걸쳐 차이 크기의 정규화된 합을 조사하여 가장 높은(또는 근사 최고의) 진폭 피크 및 피크 사이의 간격을 식별할 수 있다.
- [0129] 평가기(410)는 피크 진폭 및 간격 데이터를 사용하여 블록 크기 및 오프셋을 식별할 수 있다. 아래의 절차는 어느 피크 진폭이 현저한지와 그러한 피크 사이의 간격을 식별하는 데 사용된다.
- [0130] A) 최대 피크 진폭을 발견한다. 이러한 값으로부터, 한 쌍의 문턱값을 생성한다 — 상부 문턱값은 최대값의 5/8이고 하부 문턱값은 최대값의 3/8이다. 나중에 블록 노이즈 필터 파라미터를 설정하는 데 사용하기 위해 최대값을 저장하고, 블록 크기 및 최초 블록 경계 위치를 알고리즘적으로 결정하는 데 사용하기 위해 최대값을 포함하는 오프셋을 저장한다.
- [0131] B) 각각의 구간 크기에 대해, 각각의 오프셋에서 값들을 단계적으로 통과하고, 상부 문턱값보다 큰 값을 갖는 모든 피크와 하부 문턱값보다 큰 값을 갖는 모든 피크를 식별한다.
- [0132] C) 상부 문턱값보다 큰 적어도 하나의 피크를 갖는 각각의 구간에 대해, 그 구간에 대한 오프셋들을 단계적으로

통과하고 하부 문턱값보다 큰 피크 사이의 간격을 식별한다. 피크 사이의 간격은 단순히 그러한 피크의 오프셋 값 사이의 차이이다. 문턱값을 넘는 피크 사이의 각각의 간격을 '오프셋 어레이'에 저장한다.

[0133] D) 상기 프로세스를 병합된 구간/오프셋값들의 수평 및 수직 집합 둘 다에 대해 수행한다(즉 이미지 스케일링으로 인해 인접한 피크들이 단일 화소 폭의 피크로 병합된 경우).

[0134] 이러한 절차가 도 10의 예에 도시된 구간/오프셋 데이터에 대해 수행되는 경우, 아래의 피크 간격이 발견된다.

[0135] 구간에서의 최대 피크: 16

[0136] 오프셋에서의 최대 피크: 32

[0137] 구간 크기 8에 대한 수평 오프셋 간격

[0138] 8 8 8

[0139] 구간 크기 16에 대한 수평 오프셋 간격

[0140] 8 8 8

[0141] 이러한 데이터에 기초하여, 블록 크기가 8임이 상당히 명백하다. 그러나, 동일한 절차가 도 16a에 도시된 것로부터의 구간/오프셋 데이터에 대해 수행되는 경우, 아래의 피크 간격이 발견된다.

[0142] 구간에서의 최대 피크: 15

[0143] 오프셋에서의 최대 피크: 11

[0144] 구간 크기 14에 대한 수평 오프셋 간격

[0145] 4 3 5 2 4 3

[0146] 구간 크기 15에 대한 수평 오프셋 간격

[0147] 2 6 7 8

[0148] 구간 크기 16에 대한 수평 오프셋 간격

[0149] 3 6 7 3 6

[0150] 이 예에 있어서, 블록 크기가 무엇일 수 있는지에 관한 명확한 지표가 없다. 구간 크기 중에서 표시된 상충하는 간격들이 존재한다. 이러한 유형의 비결정적인 데이터는 가변 오프셋 절차가 또한 사용되는 이유 중 하나이다.

[0151] 이와 같이 결정되는 피크 간격 측정치들(예컨대 오프셋 어레이)은 평가기(410)에 의해 블록 크기 및 오프셋을 결정하는 데 사용될 수 있다.

[0152] 블록 크기는 구간/오프셋 어레이 내의 피크값 사이의 간격 및 구간 크기에 의해 결정되고, 한편으로 초기 오프셋은 구간/오프셋 어레이 내의 최대 피크의 오프셋 및 블록 크기에 좌우된다.

[0153] 우선시되는 단계들의 아래와 같은 예시 시퀀스가 블록 크기 및 오프셋을 결정하도록 수행될 수 있다. 이러한 단계들은 오프셋 어레이 데이터에 대해 수행된다. 'MaxPeakOffset'이라는 용어는 이전 단계로부터의 병합된 피크 데이터 집합을 사용하여 최대 진폭을 갖는 피크의 오프셋을 가리킨다. 동일한 일련의 단계들이 수평 및 수직 오프셋 어레이에 적용될 수 있다.

[0154] A) 표준 8 화소 블록 크기를 찾는다. 이것이 참일 조건은 아래와 같다.

[0155] • 0이 아닌 항목을 갖는 오프셋 어레이 내의 가장 큰 구간 크기가 16이다.

[0156] • 최대 피크값이 8의 배수인 구간 크기에서 발생한다.

[0157] • 오프셋 어레이 행들 중 하나에 적어도 세 개의 측정된 피크 대 피크 간격이 존재한다.

[0158] • 구간 크기 8에 대한 오프셋 어레이 내의 처음 세 항목이 모두 8이다.

- [0159]     • 구간 크기 16에 대한 오프셋 어레이 내의 처음 세 항목이 모두 8이다.
- [0160]     블록 크기를 8로 설정한다.
- [0161]     최초 오프셋을  $(\text{MaxPeakOffset} \bmod 8) + 8$ 로 설정한다.
- [0162]     B) 4, 8, 12 및 16 구간 크기에서 나타나는 4 화소 블록 크기를 찾는다. 이것이 참일 조건은 아래와 같다.
- [0163]     • 0이 아닌 항목을 갖는 오프셋 어레이 내의 가장 큰 구간 크기가 16이다.
- [0164]     • 최대 피크가 4의 배수인 구간 크기에서 발생한다.
- [0165]     • 오프셋 어레이 내의 4의 배수 구간들 전부가 적어도 하나의 0이 아닌 항목을 갖는다.
- [0166]     • 오프셋 어레이 내의 4의 배수 구간들 중 적어도 3개가 4의 평균 간격을 갖는다.
- [0167]     블록 크기를 4로 설정한다.
- [0168]     최초 오프셋을  $(\text{MaxPeakOffset} \bmod 4) + 4$ 로 설정한다.
- [0169]     C) 구간 크기의 절반이 동일한 피크 간격을 갖는 짝수 구간 크기를 찾는다. 이것이 참일 조건은 아래와 같다.
- [0170]     • 최대 차이값을 갖는 구간 크기가 짝수이다.
- [0171]     • 그 구간 크기에 대한 오프셋 어레이 내의 모든 항목이 동일하다.
- [0172]     • 그 구간 크기에 대한 오프셋 어레이 내의 모든 항목이 그 구간 크기의 절반에 대한 오프셋 어레이 항목과 동일하다.
- [0173]     블록 크기를 그 구간의 피크 대 피크 간격으로 설정한다.
- [0174]     최초 오프셋을  $(\text{MaxPeakOffset} \bmod \text{BlockSize}) + \text{BlockSize}$ 로 설정한다.
- [0175]     D) 구간 크기의 2/3 및 1/3이 동일한 블록 크기를 갖는 3의 배수 구간 크기를 찾는다. 이것이 참일 조건은 아래와 같다.
- [0176]     • 최대 차이값을 갖는 구간 크기가 3의 배수이다.
- [0177]     • 그 구간 크기가 12 이상이다(따라서 그 구간 크기의 1/3은 4 이상이다).
- [0178]     • 그 구간 크기에 대한 오프셋 어레이 내의 모든 항목이 동일하다.
- [0179]     • 그 구간 크기에 대한 오프셋 어레이 내의 모든 항목이 그 구간 크기의 2/3에 대한 오프셋 어레이 항목과 동일하고 그 구간 크기의 1/3에 대한 오프셋 어레이 항목과 동일하다.
- [0180]     공칭 블록 크기를 그 구간의 피크 대 피크 간격으로 설정한다.
- [0181]     최초 오프셋을  $(\text{MaxPeakOffset} \bmod \text{BlockSize}) + \text{BlockSize}$ 로 설정한다.
- [0182]     E) 구간 크기가 최대 피크값을 포함하는 구간 크기와 같은, 0이 아닌 측정된 오프셋(들)을 갖는 단일 구간 크기를 찾는다. 이것이 참일 조건은 아래와 같다.
- [0183]     • 0이 아닌 항목을 갖는 오프셋 어레이에 단일 구간 크기가 존재한다.
- [0184]     • 그 구간 크기에 대한 오프셋 어레이 내의 처음 항목이 구간 크기와 동일하다.
- [0185]     공칭 블록 크기를 그 구간 크기로 설정한다.
- [0186]     최초 오프셋을  $(\text{MaxPeakOffset} \bmod \text{BlockSize}) + \text{BlockSize}$ 로 설정한다.
- [0187]     F) 4-4-3의 블록을 갖는 11의 '매크로(macro)' 블록의 특별한 경우를 찾는다. 이러한 상황은 VCD 이미지를 스케일링하는 소정의 경우에 발견된다. 이것이 참일 조건은 아래와 같다.

- [0188]     • 최대 또는 근사 최대 피크가 11의 구간 크기에서 발생한다.
- [0189]     • 오프셋 어레이에 6개 이상의 피크 대 피크 간격 항목이 존재한다.
- [0190]     • 11-화소 구간 행 내의 처음 6개의 오프셋 어레이 항목이 수열 '4 4 3 4 4 3'의 순환 변화이다.
- [0191]     블록 크기를 4로 설정한다.
- [0192]     가장 큰 피크의 오프셋을 그것이 발견된 구간 크기(즉 11)로 나눈 후의 나머지로 최초 오프셋값을 설정하고, 그 나머지가 공칭 블록 크기(4)보다 작은 경우 여기에 공칭 블록 크기(4)를 더한다.
- [0193]     G) 구간 16의 오프셋 어레이 항목 내의 피크 대 피크 간격이 5-11 또는 11-5인, 5의 블록을 갖는 16의 '매크로' 블록의 특별한 경우를 찾는다. 이러한 경우는 흔히 1920에서 1280으로 하향 스케일링하는 것과 1080에서 720으로 하향 스케일링함으로써 야기된다(즉, 3개의 8화소 블록이 16 화소 그룹 내의 3개의 블록으로 하향 스케일링 됨). 이것이 참일 조건은 아래와 같다.
- [0194]     • 0이 아닌 항목을 갖는 오프셋 어레이 내의 가장 큰 구간 크기가 16이다.
- [0195]     • 최대 피크가 16의 구간 크기에서 발생한다.
- [0196]     • 오프셋 어레이에 2 이상의 피크 대 피크 간격 항목이 존재한다.
- [0197]     • 16 화소 구간 행 내의 처음 두 개의 오프셋 어레이 항목이 5-11 또는 11-5이다.
- [0198]     블록 크기를 5로 설정한다.
- [0199]     가장 큰 피크의 오프셋을 그것이 발견된 구간 크기(즉 16)로 나눈 후의 나머지로 최초 오프셋값을 설정하고, 그 나머지가 공칭 블록 크기(5)보다 작은 경우 여기에 공칭 블록 크기(5)를 더한다.
- [0200]     H) 0이 아닌 측정된 오프셋(들)을 갖는 단일 구간 크기만을 찾는다.
- [0201]     단일 구간에 대한 오프셋 어레이 내의 항목들의 (반올림된) 평균으로 블록 크기를 설정한다.
- [0202]     평가기(410)는 먼저 단일 구간 크기에 대한 병합된 구간/오프셋 합 내의 오프셋 항목을 조사한 후 최대값 항목의 오프셋을 발견함으로써 최초 오프셋값을 결정할 수 있다. 이후 최초 오프셋은 그 오프셋을 단일 구간 크기로 나눈 후의 나머지로 설정될 수 있고, 그 나머지가 공칭 블록 크기보다 작은 경우 여기에 공칭 블록 크기가 더해진다.
- [0203]     I) 그 밖의 경우, 블록 크기 및 오프셋을 0으로 설정한다(이는 블록이 검출되지 않았음을 가리킨다).
- [0204]     이후 평가기(410)는 최종 블록 크기 및 오프셋을 오프셋 어레이 내의 항목들을 조사함으로써 이전 단계에서 결정된 것들로부터, 그리고 가변 오프셋(-1/0/+1) 방법에 기초한 것들로부터 선택할 수 있다. 오프셋 어레이 내의 항목들로부터 결정된 블록 크기가 0인 경우(이는 유효한 블록 크기가 발견되지 않았음을 가리킴), 가변 오프셋 방법에 의해 결정된 값들을 사용한다.
- [0205]     수직 블록 크기 및 오프셋을 결정하기 위한 한 가지 특별한 경우가 존재한다. 오프셋 어레이 내의 항목들로부터 결정된 수직 블록 크기가 2인 경우, 원래의 이미지 데이터는 시간차를 갖는 두 개의 인터레이싱(interlace)된 필드(예컨대 이들은 동일한 시점으로부터 유래하지 않음)의 위브(weave)임이 거의 확실하다. 이러한 경우일 때 블록 크기를 0으로 설정한다(유효한 블록 크기가 없음을 가리킴).
- [0206]     검출기(404)의 일 실시예는 경계 지도 생성기(412)를 포함한다. 경계 지도 생성기(412)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.
- [0207]     경계 지도 생성기(412)는 공칭 블록 크기 및 시작 블록 경계 위치를 사용하여 블록 경계의 지도를 생성할 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있다.

- [0208] 생성기(412)는 차이 크기 벡터 내의 피크의 간격에 기초하여 휘도 블록 경계를 결정할 수 있다. 사용되는 절차는 제1 블록 경계의 이미 결정된 위치에서 시작한 후 공칭 블록 크기에 의해 차이 크기 데이터를 단계적으로 통과할 수 있다. 각각의 단계에서, 작은 윈도우(window)에서 최대 피크 위치가 검색되고, 이는 이후 블록 경계로서 표기된다. 그 위치는 다음 검색을 위한 시작 위치로 만들어지고, 프로세스는 데이터의 끝까지 계속된다.
- [0209] 생성기(412)는 이러한 프로세스 중에 식별되는 블록 경계의 위치를 포함하는 블록 경계 지도를 생성할 수 있다. 경계 지도는 차이 크기 벡터와 동일한 크기의 벡터이다. 경계 지도 항목은 3가이며, 0은 블록 경계가 어떤 위치에 존재하지 않음을 나타내고, 1은 블록 경계가 어떤 위치에 존재함을 나타내며, 1/2의 값은 2차 경계 특징이 존재함을 가리킨다. 2차 경계 특징은 스케일링으로 인한 스미어링된 경계를 포함하고, 따라서 경계가 2 이상의 화소 폭을 가짐을 가리킨다.
- [0210] 아래의 절차는 이것이 어떻게 수행되는지에 대한 추가적인 세부 사항을 제공한다. 블록 경계 지도는 절차에 앞서 모두 0으로 초기화될 수 있다는 점 및 프로세스의 각 단계에 대한 디폴트(default) 경계 윈도우는  $\pm 2$  화소라는 점에 주목한다. 이러한 절차는 수평 및 수직 휘도 블록 경계 지도 둘 다에 대해 동일하다.
- [0211] A) 맨 먼저, 블록 크기가 0인지 여부를 검사한다. 그러한 경우, 발견된 블록 노이즈 경계가 없으며 경계 지도는 모두 0으로 남는다.
- [0212] B) 제1 경계의 위치가 초기 오프셋값으로 설정된다. 이 위치에 대한 경계 지도값은 1.0의 최대값으로 설정된다. 이는 또한 제1 피크 검색에 대한 기선(baseline)이 된다.
- [0213] C) 스케일링에 의해 야기되는 '2배 폭 피크'를 시작 위치에서 찾는다. 시작 위치에 인접하는 위치들 중 어느 것이라도 시작 위치의 크기보다 큰 경우, 그 인접 블록 경계 지도 위치의 값을 1.0으로 설정한다. 그렇지 않으면, 시작 위치에 인접하는 위치들 중 어느 것이라도 시작 위치의 1/10 크기보다 큰 경우, 블록 경계 지도의 값을 0.5로 설정한다.
- [0214] D) 아래와 같이 각각의 후보 블록 경계 위치 주위의 윈도우에서 피크를 찾으면서 차이 크기 데이터를 단계적으로 통과한다.
- [0215] 공칭 블록 크기를 기선에 더하여 새로운 피크 검색에 대한 윈도우의 중심 위치를 결정한다.
- [0216] 검색 윈도우 내에서 최대값(즉 피크)의 위치를 식별한다.
- [0217] IF 새롭게 식별된 피크의 크기가 이전 위치에서의 피크 크기의 1/5보다 큰 경우 THEN
- [0218] • 블록 경계 지도 내의 이러한 위치를 1.0으로 설정한다.
- [0219] • 이것을 현재 피크 및 다음 검색에 대한 기선으로서 표기한다.
- [0220] • 스케일링에 의해 야기되는 '2배 폭 피크'를 시작 위치에서 찾는다. 시작 위치에 인접하는 위치들 중 어느 것이라도 시작 위치의 1/10 크기보다 큰 경우, 그 위치에 대한 블록 경계 지도의 값을 0.5로 설정한다.
- [0221] ELSE IF 새롭게 식별된 피크의 크기가 이전 피크의 1/20 보다 크고 다음으로 더 높은 위치에서의 크기는 새로운 피크 크기의 4배보다 작은 경우(주의: 이는 진정한 피크가 아니지만 바로 검색 윈도우의 한계에 있는 피크 위치를 제거한다) THEN
- [0222] • 블록 경계 위치 어레이 내의 이러한 위치를 변경하지 않고, 이를 다음 검색에 대한 기선으로서 설정한다.
- [0223] ELSE 이전의 기선에 공칭 블록 크기를 더한 것을 다음 검색에 대한 기선으로서 사용한다.
- [0224] 차이 크기 벡터의 끝에 도달할 때까지 상기 단계들의 시퀀스를 반복한다.
- [0225] 생성기(412)는 또한 채도에 대한 지도를 생성할 수 있다. 채도 블록 경계 지도를 생성하기 위한 절차는 휘도 지도에 대한 것과 유사하다. 차이점은 채도 차이 크기 데이터가 휘도 대신에 사용된다는 점이고, 채도 블록 경계 지도는 채도 상향 샘플링 후의 이미지 스케일링의 가능성으로 인해 이중 분리 피크를 가능하게 한다. 따라서 수정된 절차에 관한 아래의 예가 사용될 수 있다.
- [0226] A) 맨 먼저, 블록 크기가 0인지 여부를 검사한다. 그러한 경우, 발견된 블록 노이즈 경계가 없으며 경계 지도는 모두 0으로 남는다.

- [0227] B) 제1 경계의 위치가 초기 오프셋값으로 설정된다. 이 위치에 대한 경계 지도값은 1.0의 최대값으로 설정된다. 이는 또한 제1 피크 검색에 대한 기선이 된다.
- [0228] C) 스케일링에 의해 야기되는 '2배 폭 피크'를 시작 위치에서 찾는다. 시작 위치에 인접하는 위치들 중 어느 것이라도 시작 위치의 크기보다 큰 경우, 그 인접 블록 경계 지도 위치의 값을 1.0으로 설정한다. 그렇지 않으면, 시작 위치에 인접하는 위치들 중 어느 것이라도 시작 위치의 1/10 크기보다 큰 경우, 블록 경계 지도의 값을 0.5로 설정한다.
- [0229] D) 아래와 같이 각각의 후보 블록 경계 위치 주위의 윈도우에서 피크를 찾으면서 차이 크기 데이터를 단계적으로 통과한다.
- [0230] 공칭 블록 크기를 기선에 더하여 새로운 피크 검색에 대한 윈도우의 중심 위치를 결정한다.
- [0231] 검색 윈도우 내에서 최대값(즉 피크)의 위치를 식별한다.
- [0232] IF 새롭게 식별된 피크의 크기가 이전 위치에서의 피크 크기의 1/5보다 큰 경우 THEN
- [0233] • 블록 경계 지도 내의 이러한 위치를 1.0으로 설정한다.
- [0234] • 이것을 현재 피크 및 다음 검색에 대한 기선으로서 표기한다.
- [0235] • 스케일링에 의해 야기되는 '2배 폭 피크' 및 '이중 분리 피크'를 시작 위치에서 찾는다. 시작 위치에 인접하는 위치들 중 어느 것이라도 시작 위치의 1/10 크기보다 큰 경우, 그 위치에 대한 블록 경계 지도의 값을 0.5로 설정한다. 바로 인접하는 위치가 유효한 값을 갖지 않는 경우, 경계 지도값을 변경하기 위한 동일한 기준을 사용하여 중심으로부터 2 화소에 있는 인접 위치를 검사한다.
- [0236] • 복수의 이중 피크 선택을 실행시킨다.
- [0237] IF 복수의 분리된 피크가 존재하는 경우, 둘 중 가장 큰 것을 선택한다.
- [0238] ELSE IF 분리된 피크가 왼쪽에 존재하고 연결된 피크가 오른쪽에 존재하는 경우, 더 큰 것을 선택한다.
- [0239] ELSE IF 분리된 피크가 오른쪽에 존재하고 연결된 피크가 왼쪽에 존재하는 경우, 더 큰 것을 선택한다.
- [0240] ELSE IF 새롭게 식별된 피크의 크기가 이전 피크의 1/20 보다 크고 다음으로 더 높은 위치에서의 크기는 새로운 피크 크기의 4배보다 작은 경우(주의: 이는 진정한 피크가 아니지만 바로 검색 윈도우의 한계에 있는 피크 위치를 제거한다) THEN
- [0241] • 블록 경계 위치 어레이 내의 이러한 위치를 변경하지 않고, 이를 다음 검색에 대한 기선으로서 설정한다.
- [0242] ELSE 이전의 기선에 공칭 블록 크기를 더한 것을 다음 검색에 대한 기선으로서 사용한다.
- [0243] 차이 크기 벡터의 끝에 도달할 때까지 상기 단계들의 시퀀스를 반복한다.
- [0244] 검출기(404)의 일 실시예는 필터 이득 생성기(414)를 포함한다.
- [0245] 필터 이득 생성기(414)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.
- [0246] 필터 이득 생성기(414)는 블록 노이즈 필터에 대한 필터 이득값들을 계산, 결정 및/또는 생성할 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있다.
- [0247] 필터 이득 생성기(414)는 필터 이득을 계산할 수 있는데, 이는 기선값에서 시작하여 그것을 다수의 계산된 '수정자(modifier)'에 기초하여 수정함으로써 이루어진다. 이러한 수정자는 아래에 열거되는 인자에 기초하고, 최종 이득값을 결정하기 위해 기선 이득에 더해질 수 있다. 일반적으로, 별개의 수정자 값들이 수평 및 수직 필터 이득에 대해 계산될 수 있다. 아래는 이득 계산에서 인자들로서 사용될 수 있다.

- [0248]     • 공칭 이득은 1.0이다.
- [0249]     • 블록 경계 세기(Block Boundary Strength; BBS)는 이득을 증가시킨다.
- [0250]     • 낮은 디테일(합산된 차이 크기)은 이득을 증가시킨다.
- [0251]     • 더 큰 블록 크기는 이득을 증가시킨다.
- [0252]     • 1.0 미만의 블록 경계 세기는 이득을 빠르게 감소시킨다.
- [0253]     • 문턱값을 넘는 구간의 집계와 블록 경계 세기는 블록 노이즈가 심지어 존재하는지 여부를 결정할 수 있다. 즉, 현저한 블록 노이즈가 없는 것으로 간주되는 경우 이득이 0으로 설정될 수 있다.
- [0254]     수정자들이 모두 계산되었으면, 이들은 수평 및 수직 필터 이득을 생성하도록 기선 이득값들과 병합될 수 있다. 첫 번째 단계는 블록 노이즈가 존재하지 않음을 가리키는 조건을 식별하는 것이다. 이러한 조건이 존재하지 않는 경우, 다양한 이득 수정자가 기선 이득값에 더해지고, 결과가 상한 및 하한(각각 2.3 및 0)으로 한정된다. 이후, 이득 수정자들의 효과가 0의 이득을 초래하고 이미지 디테일 수정자의 크기가 수정자들의 전체 효과의 대부분을 구성하는 특별한 경우에 대해, 필터 이득이 1.0으로 설정된다. 이는 높은 이미지 디테일이 자체적으로 필터 이득을 0으로 강제하는 경우를 인지하기 위한 것이다.
- [0255]     수평 및 수직 이득이 둘 다 계산되었으면, 하나 이상의 특별한 조건을 찾는다. 아래의 조건이 충족되는 경우 수직 및 수평 필터 이득이 둘 다 0으로 설정된다.
- [0256]     1) 구간/오프셋 방법을 통한 블록 크기 결정이 수평 및 수직 둘 다에 대해 0이다.
- [0257]     2) 필터 이득들 중 하나가 0이거나 둘 다가 매우 낮다(낮음 = 0.25).
- [0258]     검출기(404)의 일 실시예는 모서리 경사 승수 생성기(416)를 포함한다. 모서리 경사 승수 생성기(416)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.
- [0259]     모서리 경사 승수 생성기(416)는 블록 노이즈 필터에 대한 모서리 경사 승수를 계산, 결정 및/또는 생성할 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있고, 블록 노이즈 필터의 구성요소 및 기능은 도 5의 예를 더 참조하여 기술된다. 일 실시예에서, 필터 이득 생성기(414) 및/또는 모서리 경사 승수 생성기는 필터 제어 파라미터 생성기(420)의 구성요소이다.
- [0260]     모서리 경사 승수 생성기(416)는 모서리 경사 승수를 결정할 수 있는데, 이는 기선값에서 시작하여 그것을 한 쌍의 계산된 '수정자'에 기초하여 수정함으로써 이루어진다. 이러한 수정자는 예컨대 블록 세기 및 이미지 내의 디테일의 레벨에 기초하여 결정될 수 있고, 기선 경사 승수에 더해져 최종 경사 승수값을 낳는다. 별개의 수정자 값들이 수평 및 수직 필터 이득에 대해 계산된다. 아래는 경사 승수 계산에서 인자들로서 사용될 수 있다.
- [0261]     • 공칭 승수는 2.0이다.
- [0262]     • 높은 블록 경계 세기는 승수를 감소시키고, 매우 낮은 블록 경계 세기는 승수를 증가시킨다.
- [0263]     • 높은 디테일은 승수를 증가시킨다.
- [0264]     수정자들이 모두 계산되었으면, 생성기(416)는 수평 및 수직 경사 승수를 생성하도록 기선 경사 승수값을 병합할 수 있다. 두 개의 수정자가 기선 경사 승수값에 더해질 수 있고, 결과가 1.0의 하한으로 한정된다.
- [0265]     도 5는 블록 노이즈 필터(554)의 구성요소의 예를 도시하는 블록도를 나타낸다.
- [0266]     블록 노이즈의 처리에 있어서 다음 단계는 블록 노이즈 경계를 걸러 내는 것이다. 이를 효과적으로 수행하기 위해, 블록 노이즈 필터는 이미지 내의 블록 노이즈의 특성에 기초하여 적절히 구성될 수 있다. 필터에 대한

제어 입력은 예컨대 휘도 및/또는 채도 경계 지도(도 4의 예에서 블록 노이즈 검출기에 의해 결정됨), 수직 및 수평 필터 이득값 및 수직 및 수평 모서리 세기 승수(예컨대 모서리 경사 승수)를 포함한다. 필터 입력 관리자(502)는 필터(554)에 대한 이러한 입력들을 계산 또는 관리할 수 있다. 이러한 후자의 값들은 예컨대 아래를 포함하는 인자들에 기초하여 자동으로 결정될 수 있다.

- [0267]     • 블록 검출 세기(피크 차이 크기값)
- [0268]     • 블록 크기
- [0269]     • 이미지 내의 디테일의 레벨(전체 이미지에 걸친 차이 크기값들의 정규화된 합)
- [0270]     • 문턱값보다 큰 크기를 갖는 구간의 개수
- [0271]     필터 제어값을 계산하기 위한 프로토콜(protocol)은 복수의 이미지 유형의 분석으로부터의 경험적 데이터에 기초할 수 있다.
- [0272]     필터 입력 관리자(502)는 필터(554)에 대한 아래의 입력들을 계산 또는 관리할 수 있다.
- [0273]     수평 휘도 블록 경계 지도: 수평 휘도 블록 경계 위치의 지도. 이는 1차 및 2차 경계를 가리키는 항목을 갖는, 입력 이미지의 폭인 어레이이다.
- [0274]     수직 휘도 블록 경계 지도: 수직 휘도 블록 경계 위치의 지도. 이는 1차 및 2차 경계를 가리키는 항목을 갖는, 입력 이미지의 폭인 어레이이다.
- [0275]     수평 채도 블록 경계 지도: 수평 채도 블록 경계 위치의 지도. 이는 1차 및 2차 경계를 가리키는 항목을 갖는, 입력 이미지의 폭인 어레이이다.
- [0276]     수직 채도 블록 경계 지도: 수직 채도 블록 경계 위치의 지도. 이는 1차 및 2차 경계를 가리키는 항목을 갖는, 입력 이미지의 폭인 어레이이다.
- [0277]     수평 필터 이득: 수평 필터에 대한 이득값. 이 이득값은 수평 필터 계수 및 필터의 폭의 선택에 영향을 미친다.
- [0278]     수직 필터 이득: 수직 필터에 대한 이득값. 이 이득값은 수직 필터 계수 및 필터의 높이의 선택에 영향을 미친다.
- [0279]     수평 경사 승수: 수평 경사는 수평 이미지 특징 모서리에서 필터링을 억제하는 데 사용된다. 이 경사 승수는 억제가 수행되는 정도에 영향을 미친다.
- [0280]     수직 경사 승수: 수직 경사는 수직 이미지 특징 모서리에서 필터링을 억제하는 데 사용된다. 이 경사 승수는 억제가 수행되는 정도에 영향을 미친다.
- [0281]     블록 노이즈 필터(554)의 일 실시예는 이미지 경사 생성기(504)를 포함한다. 이미지 경사 생성기(504)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.
- [0282]     이미지 경사 생성기(504)는 이미지 내의 경사를 계산, 결정 및/또는 생성할 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있다.
- [0283]     이미지 경사는 이미지 내의 물체의 모서리를 따라 아티팩트를 억제하는 데 사용될 수 있다. 경사는  $5 \times 5$  소벨 필터를 사용하여 휘도 화소 데이터를 가지고 계산된다. 소벨 필터는 예컨대 아래와 같이 정의된다.

HorSobelFilter5x5 = [-1 0 0 0 1

-2 0 0 0 2

-4 0 0 0 4

-2 0 0 0 2

-1 0 0 0 1];

VertSobelFilter5x5 = [ 1 2 4 2 1

0 0 0 0 0

0 0 0 0 0

0 0 0 0 0

-1 -2 -4 -2 -1];

[0284]

[0285]

(필터들의 출력은 10으로 나눔으로써 정규화됨에 주목한다.)

[0286]

2차원 가우스 필터가 먼저 휘도 데이터에 적용될 수 있고, 수직 및 수평 이미지 경사를 생성하기 위해 소벨 필터가 적용될 수 있다. 결과의 절대값을 취하여 수직 및 수평 경사 크기를 생성할 수 있다.

[0287]

블록 노이즈 필터(554)의 일 실시예는 수직 필터(506)를 포함한다. 수직 필터(506)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.

[0288]

수직 필터(506)는 수직 차원을 따라 이미지를 필터링할 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있다.

[0289]

수직 필터(506)는 블록 경계 지도, 필터 이득 및/또는 경사 승수 입력값들 중 하나 이상에 기초하여 수직 차원을 따라 이미지를 필터링할 수 있다. 필터는 경계에 인접한 블록들의 높이의 일부에 걸쳐 수직 블록 경계 지도에 의해 표시되는 블록 경계 위치에서 Y, Cb 및 Cr 수직 차이를 분포시킨다.

[0290]

필터(506)는 수직 필터 이득 인자에 기초하여 수직 휘도 필터에 대한 계수를 결정할 수 있다. 휘도 필터는 예컨대 블록 경계 위치의 양측에 반영되는 3개의 계수로 이루어진다. 더 많거나 더 적은 계수가 또한 사용될 수 있다. 추가적인 계수는 더 나은 결과를 생성할 수 있다. 아래의 등식은 필터 계수가 어떻게 계산될 수 있는지에 관한 예를 예시한다.

```

if (FilterGain > 1.0) then
    LCoefficient1 = (5 + ((FilterGain-1) * 1.5)) / 16;
    LCoefficient2 = (2 + ((FilterGain-1) * 2.0)) / 16;
    LCoefficient3 = ((FilterGain-1) * 2.0) / 16;
else
    LCoefficient1 = (5 + ((FilterGain-1) * 5)) / 16;
    LCoefficient2 = (2 + ((FilterGain-1) * 2)) / 16;
    LCoefficient3 = 0;
end if

```

[0291]

[0292] 필터는 휘도 블록 경계 지도 내의 피크를 갖는 위치에 적용될 수 있다. 경계 지도 내의 인접하는 위치가 동등한 0이 아닌 진폭을 갖는 경우, 가장 높은(또는 근사 최고의) 색인을 갖는 것이 중심 피크의 위치로서 선택될 수 있다.

[0293] 일 실시예에서, 세 개의 휘도 차이값이 계산되는데, 이후 이들은 각각 필터 계수의 집합에 의해 곱해진다. 첫 번째 차이값은 중심 피크 위치에서의 인접 화소들 사이의 차이에 기초하여 결정될 수 있다. 나머지 두 차이값은 처음으로 계산된 차이값, 각각의 인접 위치에서의 인접 화소값들 사이의 차이 및 인접 위치들에 대한 블록 경계 지도 크기값에 기초하여 계산될 수 있다.

[0294] 중심 블록 경계 차이값은 중심 위치에서의 화소값과 다음으로 더 높은 위치에서의 화소값 사이의 차이이다. 인접값들은 무엇보다도 각각의 인접 위치에 대한 경계 지도값이 0인지 여부에 기초하여 결정될 수 있다. 그러한 경우, 인접 차이값은 또한 0이다. 그렇지 않은 경우, 인접값은 중심 차이값 및 인접 화소 차이값의 크기에 인접 화소 차이값의 부호( $\pm 1$ )를 곱한 것 중의 최소값(또는 근사 최소값)이다.

[0295] 중심 화소가 위치 N에 있는 경우, 아래의 등식은 세 개의 차이값 각각이 어떻게 계산되는지에 관한 예를 기술한다(여기서 'Diff1'은 중심 위치에 있고, 한편으로 'Diff0' 및 'Diff2'는 각각 중심 전과 후의 위치에 있다).

Diff1 = LumaImage(N) - LumaImage(N+1);

Diff0Temp = LumaImage(N-1) - LumaImage(N);

Diff2Temp = LumaImage(N+1) - LumaImage(N+2);

if (LumaVerticalBlockBoundaryMap(N-1) > 0) then

Diff0 = min(abs(Diff1), abs(Diff0Temp)) \* sign(Diff0Temp);

else

Diff0 = 0;

end if

if (LumaVerticalBlockBoundaryMap(N+1) > 0) then

Diff2 = min(abs(Diff1), abs(Diff2Temp)) \* sign(Diff2Temp);

else

Diff2 = 0;

end if

[0296]

[0297]

이후, 계산된 필터 계수들 및 각 화소 위치에 대한 블록 경계 차이값들을 사용하여 수직 필터(506)가 이미지에 적용될 수 있다. 블록 경계 피크 위치의  $\pm 4$  화소 인근 이내에서 경계에 수직인 방향으로 위치하는 각각의 화소에 대해, 이는 필터의 효과를 계산하고 필터 계산에 기초하여 화소값을 수정한다. 아래의 등식은 필터 계산이 어떻게 이루어지는지에 관한 예를 가리키는데, N은 블록 경계(이는 두 화소 사이에 위치하는 것으로 가정됨) 바로 앞(위)의 화소 위치라고 가정한다. 계산은 경계와 직교하는 단일 화소 열에 대한 것이다.

$$\begin{aligned}
 \text{FilteredImage}(N-3) &= \text{OriginalImage}(N-3) - (\text{Diff0} * \text{LCoefficient3}); \\
 \text{FilteredImage}(N-2) &= \text{OriginalImage}(N-2) - (\text{Diff0} * \text{LCoefficient2}) \\
 &\quad - (\text{Diff1} * \text{LCoefficient3}); \\
 \text{FilteredImage}(N-1) &= \text{OriginalImage}(N-1) - (\text{Diff0} * \text{LCoefficient1}) \\
 &\quad - (\text{Diff1} * \text{LCoefficient2}) \\
 &\quad - (\text{Diff2} * (\text{LCoefficient2} / 4)); \\
 \text{FilteredImage}(N) &= \text{OriginalImage}(N) - (\text{Diff0} * \text{LCoefficient1}) \\
 &\quad - (\text{Diff1} * \text{LCoefficient1}) \\
 &\quad - (\text{Diff2} * (\text{LCoefficient1} / 2)); \\
 \text{FilteredImage}(N+1) &= \text{OriginalImage}(N+1) + (\text{Diff0} * (\text{LCoefficient1} / 2)) \\
 &\quad + (\text{Diff1} * \text{LCoefficient1}) \\
 &\quad + (\text{Diff2} * \text{LCoefficient1}); \\
 \text{FilteredImage}(N+2) &= \text{OriginalImage}(N+2) + (\text{Diff0} * (\text{LCoefficient2} / 4)) \\
 &\quad + (\text{Diff1} * \text{LCoefficient2}) \\
 &\quad + (\text{Diff2} * \text{LCoefficient1}); \\
 \text{FilteredImage}(N+3) &= \text{OriginalImage}(N+3) + (\text{Diff1} * \text{LCoefficient3}) \\
 &\quad + (\text{Diff2} * \text{LCoefficient2}); \\
 \text{FilteredImage}(N+4) &= \text{OriginalImage}(N+4) + (\text{Diff2} * \text{LCoefficient3});
 \end{aligned}$$

[0298]

[0299]

일 실시예에서, 수직 필터(506)는 예컨대 수직 필터 이득 인자에 기초하여 수직 채도에 대한 계수를 계산한다. 채도는 전형적으로 휘도보다 더 심하게 필터링되는데, 그 까닭은 채도 대역폭이 전형적으로 더 낮고 블록 크기가 전형적으로 더 크기 때문이다. 채도 필터는 블록 경계 위치의 양측에 반영되는 4개 이하의 계수로 이루어질 수 있다. 더 많거나 더 적은 계수가 또한 사용될 수 있다. 추가적인 계수는 더 나은 결과를 생성할 수 있다. 채도 필터는 더 많은 계수를 갖는데, 그 까닭은 채도 블록이 전형적으로 휘도 블록보다 크고 그 결과 효과적인 제거를 위해 더 큰 필터를 필요로 하기 때문이다. 또한, 채도에 대한 스미어링된 블록 경계는 휘도에 대한 1 화소 대신에 2 화소에 걸쳐 있을 수 있다. 아래의 등식은 필터 계수가 어떻게 계산될 수 있는지에 관한 예를 예시한다.

```

FilterGain = min(VerticalFilterGain * 1.25, 2.5);
if (FilterGain > 1.0) then
    CCoefficient1 = (5 + ((FilterGain-1) * 1.5)) / 16;
    CCoefficient2 = (2 + ((FilterGain-1) * 2.0)) / 16;
    CCoefficient3 = ( (FilterGain-0.5) * 1.5) / 16;
    CCoefficient4 = ( (FilterGain-1) * 0.875) / 16;
elseif (FilterGain < 0.5) then
    CCoefficient1 = (5 + ((FilterGain-1) * 5)) / 16;
    CCoefficient2 = (2 + ((FilterGain-1) * 2)) / 16;
    CCoefficient3 = 0;
    CCoefficient4 = 0;
else
    CCoefficient1 = (5 + ((FilterGain-1) * 5)) / 16;
    CCoefficient2 = (2 + ((FilterGain-1) * 2)) / 16;
    CCoefficient3 = ( (FilterGain-0.5) * 1.5) / 16;
    CCoefficient4 = 0;
end if

```

[0300]

[0301] 수직 필터(506)는 채도 블록 경계 지도 내의 피크를 갖는 위치에 적용될 수 있다. 경계 지도 내의 인접하는 위치가 동등한 0이 아닌 진폭을 갖는 경우, 가장 높은 색인을 갖는 것이 중심 피크의 위치로서 선택된다. 채도 블록 경계 지도는 Cb 및 Cr 둘 다에 대해 사용되고, 필터는 양자에 대해 동일하게 구현된다.

[0302] 휘도에 대해 사용되는 3개의 차이값 대신에 5개의 차이값이 사용된다는 점을 제외하고는, 휘도에 대해서와 동일하게 채도에 대해서 차이값들이 계산된다. 두 가지 이유로 인해 더 많은 개수의 차이값이 전형적으로 사용된다.

[0303] 1) 채도 블록이 전형적으로 휘도 블록보다 크기 때문에, 채도 필터는 다 많은 개수의 화소 위치에 걸쳐 있다.

[0304] 2) 채도 필터는 불연속 인접 블록 경계값들이 상향 샘플링된 채도의 스케일링을 설명할 수 있게 한다.

[0305] 중심 화소가 위치 N에 있는 경우, 아래의 등식은 5개의 차이값 각각이 어떻게 계산되는지에 관한 예를 기술한다 (여기서 'Diff2'는 중심 위치에 있고, 다른 'Diff' 값들은 중심에 인접하는 위치에 있다).

```

Diff2 = ChromaImage(N) - ChromaImage(N+1);
Diff0Temp = ChromaImage(N-2) - ChromaImage(N-1);
Diff1Temp = ChromaImage(N-1) - ChromaImage(N);
Diff3Temp = ChromaImage(N+1) - ChromaImage(N+2);
Diff4Temp = ChromaImage(N+2) - ChromaImage(N+3);
if (ChromaVerticalBlockBoundaryMap(N-2) > 0) then
    Diff0 = min(abs(Diff2), abs(Diff0Temp)) * sign(Diff0Temp);
else
    Diff0 = 0;
end if
if (ChromaVerticalBlockBoundaryMap(N-1) > 0) then
    Diff1 = min(abs(Diff2), abs(Diff1Temp)) * sign(Diff1Temp);
else
    Diff1 = 0;
end if
if (ChromaVerticalBlockBoundaryMap(N+1) > 0) then
    Diff3 = min(abs(Diff2), abs(Diff3Temp)) * sign(Diff3Temp);
else
    Diff3 = 0;
end if
if (ChromaVerticalBlockBoundaryMap(N+2) > 0) then
    Diff4 = min(abs(Diff2), abs(Diff4Temp)) * sign(Diff4Temp);
else
    Diff4 = 0;
end if

```

[0306]

[0307]

계산된 필터 계수들 및 각 화소 위치에 대한 블록 경계 차이값들을 사용하여 필터 자체가 이미지에 적용된다. 블록 경계 피크 위치의  $\pm 6$  화소 인근 이내에서 경계에 수직인 방향으로 위치하는 각각의 화소에 대해, 이는 필터의 효과를 계산하고 필터 계산에 기초하여 화소값을 수정한다.

[0308]

아래의 등식은 필터 계산이 어떻게 수행되는지에 관한 예를 가리키는데, N은 블록 경계(이는 두 화소 사이에 위치하는 것으로 가정됨) 바로 앞(위)의 화소 위치라고 가정한다. 계산은 경계에 따른 단일 열에 대한 것이다. 두 가지 별개 유형의 필터 계산이 존재하는데, 하나는 블록 경계 지도에 분리된 피크가 존재하지 않는 경우(즉 '인접하는' 값들이 중심 피크로부터 오직 1 화소 위치만큼 떨어져 있음)에 사용되고, 다른 것은 적어도 하나의 분리된 피크가 존재하는 경우(즉 '인접하는' 값들이 중심 피크로부터 2 화소 위치에 있음)에 사용된다.

[0309] 아래의 등식은 분리된 이중 피크가 존재하지 않는 경우에 사용된다.

$$\begin{aligned}
 \text{FilteredImage}(N-4) &= \text{OriginalImage}(N-4) - (\text{Diff1} * \text{CCoefficient4}); \\
 \text{FilteredImage}(N-3) &= \text{OriginalImage}(N-3) - (\text{Diff1} * \text{CCoefficient3}) \\
 &\quad - (\text{Diff2} * \text{CCoefficient4}); \\
 \text{FilteredImage}(N-2) &= \text{OriginalImage}(N-2) - (\text{Diff1} * \text{CCoefficient2}) \\
 &\quad - (\text{Diff2} * \text{CCoefficient3}); \\
 \text{FilteredImage}(N-1) &= \text{OriginalImage}(N-1) - (\text{Diff1} * \text{CCoefficient1}) \\
 &\quad - (\text{Diff2} * \text{CCoefficient2}) \\
 &\quad - (\text{Diff3} * (\text{CCoefficient2} / 4)); \\
 \text{FilteredImage}(N-0) &= \text{OriginalImage}(N-0) - (\text{Diff1} * -\text{CCoefficient1}) \\
 &\quad - (\text{Diff2} * \text{CCoefficient1}) \\
 &\quad - (\text{Diff3} * (\text{CCoefficient1} / 2)); \\
 \text{FilteredImage}(N+1) &= \text{OriginalImage}(N+1) + (\text{Diff1} * (\text{CCoefficient1} / 2)) \\
 &\quad + (\text{Diff2} * \text{CCoefficient1}) \\
 &\quad + (\text{Diff3} * -\text{CCoefficient1}); \\
 \text{FilteredImage}(N+2) &= \text{OriginalImage}(N+2) + (\text{Diff1} * (\text{CCoefficient2} / 4)) \\
 &\quad + (\text{Diff2} * \text{CCoefficient2}) \\
 &\quad + (\text{Diff3} * \text{CCoefficient1}); \\
 \text{FilteredImage}(N+3) &= \text{OriginalImage}(N+3) + (\text{Diff2} * \text{CCoefficient3}) \\
 &\quad + (\text{Diff3} * \text{CCoefficient2}); \\
 \text{FilteredImage}(N+4) &= \text{OriginalImage}(N+4) + (\text{Diff2} * \text{CCoefficient4}) \\
 &\quad + (\text{Diff3} * \text{CCoefficient3}); \\
 \text{FilteredImage}(N+5) &= \text{OriginalImage}(N+5) + (\text{Diff3} * \text{CCoefficient4});
 \end{aligned}$$

[0310]

[0311] 아래의 등식은 분리된 이중 피크가 존재하는 경우에 사용된다.

$$\begin{aligned}
 \text{FilteredImage}(N-5) &= \text{OriginalImage}(N-5) - (\text{Diff0} * \text{CCoefficient4}); \\
 \text{FilteredImage}(N-4) &= \text{OriginalImage}(N-4) - (\text{Diff0} * \text{CCoefficient3}); \\
 \text{FilteredImage}(N-3) &= \text{OriginalImage}(N-3) - (\text{Diff0} * \text{CCoefficient2}) \\
 &\quad - (\text{Diff2} * \text{CCoefficient4}); \\
 \text{FilteredImage}(N-2) &= \text{OriginalImage}(N-2) - (\text{Diff0} * \text{CCoefficient1}) \\
 &\quad - (\text{Diff2} * \text{CCoefficient3}); \\
 \text{FilteredImage}(N-1) &= \text{OriginalImage}(N-1) + (\text{Diff0} * \text{CCoefficient1}) \\
 &\quad - (\text{Diff2} * \text{CCoefficient2}) \\
 &\quad - (\text{Diff4} * \text{CCoefficient4}); \\
 \text{FilteredImage}(N) &= \text{OriginalImage}(N) + (\text{Diff0} * \text{CCoefficient2}) \\
 &\quad - (\text{Diff2} * \text{CCoefficient1}) \\
 &\quad - (\text{Diff4} * \text{CCoefficient3}); \\
 \text{FilteredImage}(N+1) &= \text{OriginalImage}(N+1) + (\text{Diff0} * \text{CCoefficient3}) \\
 &\quad + (\text{Diff2} * \text{CCoefficient1}) \\
 &\quad - (\text{Diff4} * \text{CCoefficient2}); \\
 \text{FilteredImage}(N+2) &= \text{OriginalImage}(N+2) + (\text{Diff0} * \text{CCoefficient4}) \\
 &\quad + (\text{Diff2} * \text{CCoefficient2}) \\
 &\quad - (\text{Diff4} * \text{CCoefficient1}); \\
 \text{FilteredImage}(N+3) &= \text{OriginalImage}(N+3) + (\text{Diff2} * \text{CCoefficient3}) \\
 &\quad + (\text{Diff4} * \text{CCoefficient1}); \\
 \text{FilteredImage}(N+4) &= \text{OriginalImage}(N+4) + (\text{Diff2} * \text{CCoefficient4}) \\
 &\quad + (\text{Diff4} * \text{CCoefficient2}); \\
 \text{FilteredImage}(N+5) &= \text{OriginalImage}(N+5) + (\text{Diff4} * \text{CCoefficient3}); \\
 \text{FilteredImage}(N+6) &= \text{OriginalImage}(N+6) + (\text{Diff4} * \text{CCoefficient4});
 \end{aligned}$$

[0312]

[0313] 블록 노이즈 필터(554)의 일 실시예는 수평 필터(510)를 포함한다. 수평 필터(510)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.

[0314] 수평 필터(510)는 수평 차원을 따라 이미지를 필터링할 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있다.

[0315] 일 실시예에서, 수평 필터(510)는 수평 블록 경계 지도, 필터 이득 및/또는 경사 승수 입력값들에 기초하여 수평 차원을 따라 수직으로 필터링된 이미지를 필터링할 수 있다. 필터는 경계에 인접한 블록들의 폭의 일부에 걸쳐 수평 블록 경계 지도에 의해 지정되는 블록 경계 위치에서 Y, Cb 및 Cr 수평 차이를 분포시킨다.

- [0316] 수평 필터(510)는 수직 필터와 유사하게 구현될 수 있는데, 차이점은 이것이 원래 이미지 대신에 수직 필터의 출력에 적용된다는 점 및 이것이 수평 블록 경계 지도에 의해 지정되는 위치에서 수평 방향으로 적용된다는 점이다.
- [0317] 각각의 수평 블록 경계 위치에 대한 필터링이 완료된 후에, 수직으로 필터링된 이미지와 원래의 이미지는 예컨대 모서리 혼합기(508)에 의해 수직 경사 크기값에 기초하여 혼합될 수 있다. 출력은 수평 필터에 송신될 수 있다. 혼합기(508)는 입력 이미지의 수평 경사를 사용하여 수직으로 필터링된 이미지와 수평으로 필터링된 이미지를 혼합할 수 있다. 일 실시예에서, 모서리 혼합기(508)는 수평 이미지 경사의 크기에 수평 경사 승수를 곱한 것에 기초하여 혼합 인자를 생성한다. 1보다 큰 결과적인 혼합 지도값은 1로 클리핑될 수 있다. 이후 혼합 지도 및 혼합된 이미지는 수직으로 필터링된 이미지에 대해 사용되는 것과 유사한 방식으로 계산될 수 있다.
- [0318] 블록 노이즈 필터(554)의 일 실시예는 모서리 평활기(512)를 포함한다. 모서리 평활기(512)는 예컨대 머신 상의 컴퓨터 판독 가능 매체 또는 컴퓨터 판독 가능 저장 매체, 펌웨어, 하드웨어, 이들의 조합, 또는 임의의 적용 가능한 공지된 또는 편리한 장치 또는 시스템 내에 구현되는 소프트웨어로서 구현될 수 있다. 본 명세서에서 기술되는 이러한 그리고 다른 엔진은 본 명세서에 암시적으로, 명시적으로, 또는 내재적으로 기술된 기능 중 적어도 일부를 실행하고/하거나 동등한 기능을 실행할 수 있는 임의의 머신, 제품, 또는 구성을 포함하고자 하는 것이다.
- [0319] 모서리 평활기(512)는 블록 경계 및/또는 필터 아티팩트를 제거할 수 있는 하드웨어 구성요소 및/또는 소프트웨어 에이전트의 임의의 조합일 수 있다. 모서리 평활기는 2008년 9월 4일에 출원된 "SYSTEM, METHOD, AND APPARATUS FOR SMOOTHING OF EDGES IN IMAGES TO REMOVE IRREGULARITIES"라는 명칭의 동시 출원 계속중인 미국 특허 출원 제12/204,760호에 기술된 바처럼 구현될 수 있으며, 상기 출원의 내용은 본 명세서에 참고 문헌으로서 포함된다.
- [0320] 일 실시예에서, 모서리 평활기(512)는 결과적인 이미지에 대해 모서리 평활을 수행하여 이미지 내의 물체 모서리를 따라 블록 경계 및 필터 아티팩트를 제거한다. 임의의 적합한 모서리 평활 필터가 사용될 수 있다. 필터 커널(kernel) 크기는 블록 필터 이득값에 기초하여 결정될 수 있다. 일 실시예에서, 수직 및 수평 필터 이득값이 둘 다 0인 경우, 모서리 평활 단계를 건너뛴다. 두 이득값이 2보다 작은 경우, 작은 필터 커널이 사용된다. 그렇지 않으면, 큰 모서리 평활 필터 커널이 사용된다.
- [0321] 도 17 및 18은 스케일링의 효과가 있고 없는 경우에 블록 노이즈 필터를 노이즈 경계에 적용하는 효과의 예를 도시한다.
- [0322] 블록 노이즈 필터는 경계에 인접하고 수직인 화소들 가운데에 인접 블록들 사이의 경계에 걸친 차이를 분포시킴으로써 블록 노이즈를 감소시킨다. 필터 계수는 필터에 대한 제어 입력인 필터 이득값에 기초한다. 필터 위치는 입력 블록 경계 지도에 의해 결정되는데, 블록 경계는 두 화소 위치 사이에 위치하도록 지정된다.
- [0323] 필터의 효과에 관한 예가 도 17의 예에 도시된다. 도면은 '화소 0'과 '화소 1' 사이에 위치하는 가상의 블록 경계를 가로지르는 비디오 신호의 크기를 플로팅(plot)하는데, 왼쪽 블록의 레벨은 0이고 오른쪽 블록은 1이다. 플롯 내의 적색선은 원래의 사전 필터링된 신호이다. 필터는 블록 경계에 인접하는 화소들 가운데에 왼쪽과 오른쪽 블록 사이의 차이(0으로부터 1까지의, 원래 신호의 큰 단계)를 분포시키도록 동작한다. 청색선은 필터링된 결과이다.
- [0324] 필터는 또한 블록 경계 상의 스케일링의 효과를 처리할 수 있다. 이는 두 번 스케일링된 신호를 도시하는 도 18의 예에서 예시된다. 이 예는 먼저 4:2:0에서 4:4:4로 상향 샘플링된 후 상이한 해상도로 스케일링된 채도 신호를 시뮬레이션한다. 이는 블록 경계의 인근에 있는 화소들에서 2 화소 폭을 갖는 셸프(shelf)(즉 도면의 적색선에서 화소 -1 및 화소 0)를 생성한다. 그 결과, 화소값이 블록 경계를 가로질러 왼쪽으로부터 오른쪽으로 횡단하는 경우 화소값에는 두 단계가 존재한다. 즉, 화소 -2와 화소 -1 사이의 왼쪽에 부 단계가, 화소 0과 화소 1 사이의 오른쪽에 더 큰 주 단계가 존재한다. 원래의 신호는 적색으로 표시되고, 최종적인 필터링된 결과는 청색으로 표시된다. 부 단계 및 주 단계의 전체 필터에 대한 효과는 각각 갈색과 황색으로 표시된다. 전체 필터 응답은 두 단계에 대한 응답의 합이다.
- [0325] 필터가 적용된 후에, 필터링된 이미지 및 필터링되지 않은 이미지로 모서리 혼합 동작이 수행될 수 있는데, 혼합의 정도는 경사 승수 제어 입력에 의해 결정된다. 일부 경우에 있어서, 모서리 평활 함수가 적용되어 이미지 내의 물체 모서리를 따라 블록 노이즈 및 필터 아티팩트를 제거한다.

- [0326] 도 19는 블록 노이즈 검출을 위한 예시 프로세스의 흐름도를 도시한다.
- [0327] 프로세스 1902에서, 차이 크기(dMag)가 계산된다. 프로세스 1904에서, 고레벨 dMag 값이 억제된다. 프로세스 1906에서, 중간값이 dMag으로부터 제거된다. 프로세스 1908에서, 물체 모서리 주위에서 dMag 값이 억제된다. 프로세스 1910에서, 행 및 열에 걸쳐 dMag의 평균이 계산되고, 중간값을 감산하도록 평균이 최적화된다. 프로세스 1912에서, dMag의 합이 계산되고, 일부 경우에는 정규화된다. 프로세스 1914는 프로세스 블록 1912에서 계산된 고정 오프셋을 갖는 구간/오프셋 합으로부터 피크를 병합한다. 이후 이러한 병합된 피크 구간/오프셋 합을 조사하여 프로세스 블록 1918에서 각각의 구간 크기에 대한 상이한 오프셋값의 피크 사이의 간격을 결정한다. 이후 피크 간격은 프로세스 블록 1920에서 고정 구간을 갖는 구간 오프셋 합에 기초하여 블록 크기 및 오프셋을 결정하는 기초로서 사용된다. 프로세스 1914, 1918 및 1920과 병행하여, 가변 오프셋을 사용하여 계산된 정규화된 구간/오프셋 합을 조사하여 프로세스 블록 1916에서 블록 크기를 결정한다. 프로세스 1922에서, 프로세스 1920 및 1916에서 두 가지 상이한 방법에 의해 결정된 블록 크기 및 오프셋값을 조사하여 최종 블록 크기 및 오프셋을 선택한다.
- [0328] 도 20은 블록 노이즈 필터에 대한 제어 파라미터를 선택하기 위한 예시 프로세스의 흐름도를 도시한다.
- [0329] 프로세스 2002에서, 블록 노이즈 필터 이득이 결정된다. 프로세스 2004에서, 모서리 경사 승수가 결정된다. 프로세스 2006에서, 블록 노이즈가 필터링된다.
- [0330] 도 21은 블록 노이즈 필터를 이미지에 적용하는 흐름도를 도시한다.
- [0331] 프로세스 2102에서, 이미지 내의 블록 노이즈 특성에 기초하여 블록 노이즈 필터에 대한 이득 설정이 선택된다. 프로세스 2104에서, 수직 이득 인자를 사용하여 수직 휘도 필터에 대한 계수가 계산된다.
- [0332] 프로세스 2106에서, 수직 이득 인자를 사용하여 수직 채도 필터에 대한 계수가 계산된다. 프로세스 2108에서, 수직 및 채도 필터에 대한 계수를 사용하여 수직 차원을 따라 이미지가 필터링되어 수직으로 필터링된 이미지를 생성하고, 수직으로 필터링된 이미지와 원래의 이미지를 혼합함으로써 혼합된 이미지를 생성한다.
- [0333] 프로세스 2110에서, 수평 이득 인자를 사용하여 수평 휘도 필터에 대한 계수가 계산된다. 프로세스 2112에서, 수평 이득 인자를 사용하여 수평 채도 필터에 대한 계수가 계산된다. 프로세스 2114에서, 수평 및 채도 필터에 대한 계수를 사용하여 수평 차원을 따라 수직으로 필터링된 이미지가 필터링되어 수평으로 필터링된 이미지를 생성한다. 프로세스 2116에서, 수직으로 필터링된 이미지와 수평으로 필터링된 이미지를 혼합함으로써 혼합된 이미지가 생성된다.
- [0334] 도 22는 본 명세서에서 논의되는 방법들 중 임의의 하나 이상을 머신이 실행하게 하기 위한 명령어들의 집합이 실행될 수 있는 컴퓨터 시스템의 예시 형태로 머신을 표현한 도표이다.
- [0335] 도 22의 예에서, 컴퓨터 시스템(2200)은 프로세서, 메모리, 비휘발성 메모리 및 인터페이스 장치를 포함한다. 다양한 통상적인 구성요소(예컨대 캐시 메모리)가 예시의 편의상 생략된다. 컴퓨터 시스템(900)은 본 명세서에 기술된 구성요소들 중 임의의 것이 구현될 수 있는 하드웨어 장치를 예시하기 위한 것이다. 컴퓨터 시스템(2200)은 임의의 적용 가능한 공지된 또는 종래의 유형의 것일 수 있다. 컴퓨터 시스템(2200)의 구성요소들은 버스를 통해 또는 소정의 다른 공지된 또는 종래의 장치를 통해 함께 결합될 수 있다.
- [0336] 프로세서는 예컨대 Intel Pentium 마이크로프로세서 또는 Motorola Power PC 마이크로프로세서와 같은 종래의 마이크로프로세서일 수 있다. 관련 기술 분야의 당업자는 "머신 판독 가능 (저장) 매체" 또는 "컴퓨터 판독 가능 (저장) 매체"라는 용어가 프로세서에 의해 액세스 가능한 임의의 유형의 장치를 포함함을 인식할 것이다.
- [0337] 메모리는 예컨대 버스에 의해 프로세서에 결합된다. 메모리는 예컨대 동적 RAM(Dynamic Random Access Memory; DRAM) 및 정적 RAM(SRAM)과 같은 RAM을 포함할 수 있지만, 이에 한정되지 않는다. 메모리는 국지적이거나, 이격되거나, 또는 분산될 수 있다.
- [0338] 버스는 또한 프로세서를 비휘발성 메모리 및 드라이브 유닛에 결합시킨다. 비휘발성 메모리는 종종 자기 플로피 또는 하드 디스크, 자기-광학 디스크, 광학 디스크, CD-ROM, EPROM, 또는 EEPROM과 같은 ROM(Read-Only Memory), 자기 또는 광학 카드, 또는 많은 양의 데이터를 위한 다른 형태의 저장소이다. 이러한 데이터 중 일부는 종종 컴퓨터(2200)에서의 소프트웨어 실행 중에 직접 메모리 액세스 프로세스에 의해 메모리에 기입된다. 비휘발성 저장소는 국지적이거나, 이격되거나, 또는 분산될 수 있다. 시스템은 메모리에서 입수 가능한 모든 적용 가능한 데이터로 생성될 수 있기 때문에, 비휘발성 메모리는 선택적이다. 전형적인 컴퓨터 시스템은 대개

적어도 하나의 프로세서, 메모리 및 메모리를 프로세서에 결합시키는 장치(예컨대 버스)를 포함할 것이다.

- [0339] 소프트웨어는 전형적으로 비휘발성 메모리 및/또는 드라이브 유닛에 저장된다. 사실, 대용량 프로그램의 경우, 전체 프로그램을 메모리에 저장하는 것조차 가능하지 않을 수 있다. 그럼에도 불구하고, 소프트웨어가 실행되기 위해서는 필요한 경우 소프트웨어가 처리에 적합한 컴퓨터 판독 가능 위치로 이동되어야 한다는 점 및 예시를 위해 그러한 위치는 본 명세서에서 메모리라고 지칭된다는 점을 이해해야 한다. 소프트웨어가 실행을 위해 메모리로 이동되는 경우에도, 프로세서는 전형적으로 소프트웨어와 연관된 값들을 저장하기 위한 하드웨어 레지스터(register) 및 이상적으로는 실행 속도를 높이는 역할을 하는 로컬 캐시(local cache)를 사용할 것이다. 본 명세서에서 사용되는 바처럼, 소프트웨어 프로그램은 소프트웨어 프로그램이 "컴퓨터 판독 가능 매체에 구현"되는 것으로 지칭되는 경우 임의의 공지된 또는 종래의 위치에(비휘발성 저장소로부터 하드웨어 레지스터로) 저장되는 것으로 가정된다. 프로세서는 프로그램과 연관된 적어도 하나의 값이 프로세서에 의해 판독 가능한 레지스터에 저장되는 경우 "프로그램을 실행하도록 구성되는" 것으로 간주된다.
- [0340] 버스는 또한 프로세서를 네트워크 인터페이스 장치에 결합시킨다. 인터페이스는 모뎀 또는 네트워크 인터페이스 중 하나 이상을 포함할 수 있다. 모뎀 또는 네트워크 인터페이스는 컴퓨터 시스템(2200)의 일부로 간주될 수 있음을 이해할 것이다. 인터페이스는 아날로그 모뎀, ISDN 모뎀, 케이블 모뎀, 토큰 링(token ring) 인터페이스, 위성 송신 인터페이스(예컨대 "직접 PC"), 또는 컴퓨터 시스템을 다른 컴퓨터 시스템에 결합시키기 위한 다른 인터페이스를 포함할 수 있다. 인터페이스(208)는 하나 이상의 입력 및/또는 출력 장치를 포함할 수 있다. I/O 장치는 예컨대 키보드, 마우스 또는 다른 포인팅 장치, 디스크 드라이브, 프린터, 스캐너 및 디스플레이를 포함하는 다른 입력 및/또는 출력 장치를 포함할 수 있지만, 이에 한정되지 않는다. 디스플레이 장치는 예컨대 CRT(Cathode Ray Tube), LCD(Liquid Crystal Display), 또는 소정의 다른 적용 가능한 공지된 또는 종래의 디스플레이 장치를 포함할 수 있지만, 이에 한정되지 않는다. 편의상 도 22의 예에 도시되지 않은 임의의 장치의 제어가 인터페이스에 존재한다고 가정한다.
- [0341] 동작시에, 컴퓨터 시스템(900)은 디스크 운영 체제와 같은 파일 관리 시스템을 포함하는 운영 체제 소프트웨어에 의해 제어될 수 있다. 연관된 파일 관리 시스템 소프트웨어를 갖는 운영 체제 소프트웨어의 일례는 미국 워싱턴주 레드몬드(Redmond)에 소재한 Microsoft사의 Windows®로 알려진 운영 체제군이다. 연관된 파일 관리 시스템 소프트웨어를 갖는 운영 체제 소프트웨어의 다른 예는 Linux 운영 체제 및 이것의 연관된 파일 관리 시스템이다. 파일 관리 시스템은 전형적으로 비휘발성 메모리 및/또는 드라이브 유닛에 저장되고, 데이터를 입력 및 출력하고 메모리에 데이터를 저장하기 위해 운영 체제가 필요로 하는 다양한 동작(비휘발성 메모리 및/또는 드라이브 유닛에 파일을 저장하는 것을 포함함)을 프로세서로 하여금 실행하게 한다.
- [0342] 상세한 설명의 일부는 컴퓨터 메모리 내의 데이터 비트에 대한 동작의 알고리즘 및 기호적 표현과 관련하여 제공될 수 있다. 이러한 알고리즘 설명 및 표현은 데이터 처리 기술 분야의 당업자가 이들의 작업의 내용을 그 기술 분야의 다른 당업자에게 가장 효과적으로 전달하기 위해 사용하는 수단이다. 알고리즘은 여기에서, 그리고 일반적으로 원하는 결과를 낳는 동작들의 자기 모순이 없는 시퀀스로 간주된다. 동작들은 물리량의 물리적 조작을 필요로 하는 것들이다. 대개, 반드시 그러하지는 않더라도, 이러한 양은 저장, 전송, 결합, 비교 및 기타 조작이 가능한 전기 또는 자기 신호의 형태를 취한다. 때로는 주로 통상적인 용법을 이유로 이러한 신호를 비트, 값, 요소, 기호, 문자, 항, 또는 숫자 등으로 지칭하는 것이 편리한 것으로 드러났다.
- [0343] 그러나, 명심해야 할 것은 이러한 그리고 유사한 용어들 전부가 적절한 물리량과 연관되는 것이고, 이러한 양에 적용되는 편리한 표지일 뿐이라는 점이다. 아래의 논의에서 명확해지는 것과 달리 구체적으로 언급되지 않는 한, 설명 전체에 걸쳐 "처리" 또는 "계산" 또는 "연산" 또는 "결정" 또는 "디스플레이" 등과 같은 용어를 사용하는 논의는 컴퓨터 시스템의 레지스터 및 메모리 내의 물리(전기)량으로 표현되는 데이터를 조작하여 컴퓨터 시스템 메모리 또는 레지스터 또는 다른 이러한 정보 저장, 송신 또는 디스플레이 장치 내의 물리량으로 유사하게 표현되는 다른 데이터로 변환하는 컴퓨터 시스템 또는 유사한 전자 컴퓨팅 장치의 동작 및 프로세스를 지칭한다.
- [0344] 본 명세서에서 제공되는 알고리즘 및 디스플레이는 임의의 특정한 컴퓨터 또는 다른 장치와 고유하게 관련되지 않는다. 다양한 범용 시스템이 본 명세서의 설명에 따른 프로그램과 함께 사용될 수 있거나, 또는 일부 실시예의 방법을 수행하기 위한 보다 특수한 장치를 구축하는 것이 편리할 수 있다. 다양한 이러한 시스템을 위해 필요한 구조는 아래의 설명으로부터 드러날 것이다. 또한, 기법들은 임의의 특정한 프로그래밍 언어를 참조하여 기술되지 않고, 따라서 다양한 실시예는 다양한 프로그래밍 언어를 사용하여 구현될 수 있다.
- [0345] 대안적인 실시예에서, 머신은 독립형(standalone) 장치로서 동작하거나 다른 머신에 접속(예컨대 네트워크)될

수 있다. 네트워크된 배치에서, 머신은 클라이언트-서버 네트워크 환경의 서버 또는 클라이언트 머신의 용량 내에서, 또는 피어 대 피어(peer-to-peer)(또는 분산형) 네트워크 환경의 피어 머신으로서 동작할 수 있다.

[0346] 머신은 서버 컴퓨터, 클라이언트 컴퓨터, 개인용 컴퓨터(PC), 태블릿(tablet) PC, 랩톱 컴퓨터, 셋톱 박스(set-top box; STB), PDA(Personal Digital Assistant), 휴대 전화, iPhone, Blackberry, 프로세서, 전화, 웹 응용 기기(web appliance), 네트워크 라우터(network router), 스위치 또는 브리지(bridge), 또는 그러한 머신에 의해 취해질 동작들을 지정하는 명령들의 집합(순차적이거나 그렇지 않을 수 있음)을 실행할 수 있는 임의의 머신일 수 있다.

[0347] 머신 판독 가능 매체 또는 머신 판독 가능 저장 매체는 예시 실시예에서 단일 매체인 것으로 표시되지만, "머신 판독 가능 매체" 및 "머신 판독 가능 저장 매체"라는 용어는 하나 이상의 명령어 집합을 저장하는 단일 매체 또는 복수의 매체(예컨대 집중형 또는 분산형 데이터베이스 및/또는 연관된 캐시 및 서버)를 포함하는 것으로 간주되어야 한다. "머신 판독 가능 매체" 및 "머신 판독 가능 저장 매체"라는 용어는 또한 머신에 의한 실행을 위한 것으로서 현재 개시된 기법 및 혁신의 방법들 중 임의의 하나 이상을 머신으로 하여금 수행하게 하는 명령어들의 집합을 저장, 인코딩 또는 운반할 수 있는 임의의 매체를 포함하는 것으로 간주되어야 한다.

[0348] 일반적으로, 본 개시 내용의 실시예를 구현하기 위해 실행되는 루틴(routine)은 "컴퓨터 프로그램"이라고 지칭되는 운영 체제 또는 특정한 애플리케이션(application), 구성요소, 프로그램, 객체, 모듈 또는 명령어 시퀀스의 일부로서 구현될 수 있다. 컴퓨터 프로그램은 전형적으로 다양한 시간에 컴퓨터 내의 다양한 메모리 및 저장 장치에 있고 컴퓨터 내의 하나 이상의 처리 유닛 또는 프로세서에 의해 판독 및 실행되는 경우 컴퓨터로 하여금 본 개시 내용의 다양한 태양을 수반하는 요소들을 실행하기 위한 동작들을 수행하게 하는 하나 이상의 명령어 집합을 포함한다.

[0349] 더욱이, 완전히 동작하는 컴퓨터 및 컴퓨터 시스템의 맥락에서 실시예들이 기술되었지만, 본 기술 분야의 당업자는 다양한 실시예가 다양한 형태로 프로그램 제품으로서 배포될 수 있다는 점 및 본 개시 내용이 배포를 실제로 실행하는 데 사용되는 머신 또는 컴퓨터 판독 가능 매체의 특정한 유형에 관계없이 동등하게 적용된다는 점을 이해할 것이다.

[0350] 머신 판독 가능 저장 매체, 머신 판독 가능 매체, 또는 컴퓨터 판독 가능 (저장) 매체의 추가적인 예는 다른 것들 중에서도 휘발성 및 비휘발성 메모리 장치, 플로피 및 다른 이동식 디스크, 하드 디스크 드라이브, 광학 디스크{예컨대 CD ROM(Compact Disk Read-Only Memory), DVD(Digital Versatile Disk) 등}와 같은 기록 가능형 매체 및 디지털 및 아날로그 통신 링크와 같은 송신형 매체를 포함하지만, 이에 한정되지 않는다.

[0351] 문맥이 명확히 달리 요구하지 않는 한, 본 명세서 및 청구범위 전체에 걸쳐, "포함한다" 및 "포함하는" 등의 단어는 배타적이거나 총망라적인 의미가 아닌 포괄적인 의미로 해석되어야 한다. 즉, "포함하지만 이에 한정되지 않는다"는 의미이다. 본 명세서에서 사용되는 바처럼 "접속", "결합"이라는 용어 또는 이것의 임의의 변형은 둘 이상의 요소 사이의 직접적이거나 간접적인 임의의 접속 또는 결합을 의미한다. 요소들 사이의 접속의 결합은 물리적, 논리적, 또는 이들의 조합일 수 있다. 또한, "여기서", "위에서", "아래에서"와 같은 단어 및 유사한 의미의 단어는 본 출원에서 사용되는 경우 본 출원을 전체적으로 지칭하고 본 출원의 임의의 특정한 부분을 지칭하지 않는다. 맥락이 허락하는 곳에서, 단수 또는 복수를 사용하는 상기 상세한 설명 내의 단어들은 또한 복수 또는 단수를 각각 포함할 수 있다. "또는"이라는 단어는, 둘 이상의 항목의 목록을 참조하자면, 단어의 해석 전부, 즉 목록 내의 항목들 중 임의의 것, 목록 내의 항목들 전부 및 목록 내의 항목들의 임의의 조합을 포괄한다.

[0352] 본 개시 내용의 실시예에 관한 상기 상세한 설명은 위에서 개시된 정확한 형태로 설명을 한정하거나 총망라적인 것이 아니다. 개시 내용의 특정한 실시예 및 예시가 예시를 위해 위에서 기술되었지만, 관련 기술 분야의 당업자가 인식하는 바처럼 다양한 동등한 수정이 본 개시 내용의 범위 내에서 가능하다. 예컨대, 프로세스 또는 블록이 주어진 순서로 제시되었지만, 대안적인 실시예는 상이한 순서로 단계들을 갖는 루틴을 수행하거나 블록들을 갖는 시스템을 이용할 수 있으며, 일부 프로세스 또는 블록이 삭제, 이동, 추가, 세분화, 결합 및/또는 수정되어 대안 또는 하위 조합을 제공할 수 있다. 이러한 프로세스들 또는 블록들 각각은 다양한 상이한 방식으로 구현될 수 있다. 또한, 프로세스들 또는 블록들이 때때로 순차적으로 수행되는 것으로 표시되지만, 이러한 프로세스들 또는 블록들은 그 대신 병렬적으로 수행될 수 있거나, 상이한 시간에 수행될 수 있다. 또한, 본 명세서에 표기된 임의의 특정한 숫자는 예시일 뿐이며, 대안적인 구현에는 상이한 값 또는 범위를 이용할 수 있다.

[0353] 본 명세서에 제공되는 개시 내용의 설명은 반드시 위에서 기술된 시스템에 적용되는 것이 아니라 다른 시스템에

도 적용될 수 있다. 위에서 기술된 다양한 실시예의 요소 및 동작은 추가 실시예를 제공하도록 조합될 수 있다.

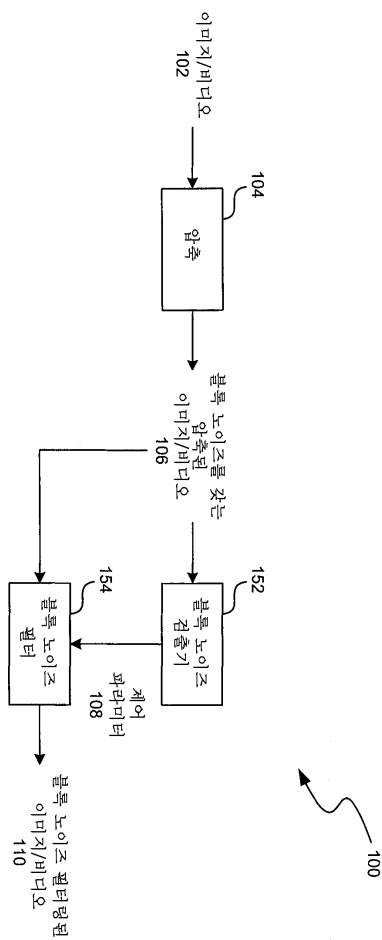
[0354] 첨부된 출원 서류에 열거될 수 있는 임의의 것을 포함하는 임의의 특허 및 출원 및 다른 참고 문헌이 본 명세서에 참고로서 포함된다. 본 개시 내용의 태양은 필요한 경우 위에서 기술된 다양한 참고 문헌의 시스템, 기능 및 개념을 이용하도록 수정되어 본 개시 내용의 더 추가적인 실시예를 제공할 수 있다.

[0355] 상기 상세한 설명에 비추어 이러한 그리고 다른 변화가 본 개시 내용에 대해 이루어질 수 있다. 상기 설명은 본 개시 내용의 소정의 실시예를 기술하고 예상되는 최상의 모드를 기술하지만, 상기 설명이 본문에서 얼마나 상세히 나타나는지에 관계없이 설명은 많은 방식으로 실시될 수 있다. 시스템의 세부 사항은 그 구현 세부 사항에서 상당히 달라질 수 있지만, 여전히 본 명세서에 개시된 주제에 의해 포괄된다. 위에서 주목한 바처럼, 본 개시 내용의 특정한 특징 또는 태양을 기술하는 경우에 사용되는 특정한 용어는 상기 용어가 그 용어와 연관된 본 개시 내용의 임의의 특정한 특성, 특징, 또는 태양으로 제한되는 것으로 본 명세서에서 재정의됨을 의미하는 것으로 간주되지 않아야 한다. 일반적으로, 아래의 청구범위에서 사용되는 용어는 상기 상세한 설명 부분이 이러한 용어를 명시적으로 정의하지 않는 한 본 개시 내용을 본 명세서에 개시된 특정한 실시예로 한정하는 것으로 해석되지 않아야 한다. 따라서, 본 개시 내용의 실제 범위는 개시된 실시예뿐만 아니라 청구범위 하에서 본 개시 내용을 실시 또는 구현하는 모든 동등한 방식을 포괄한다.

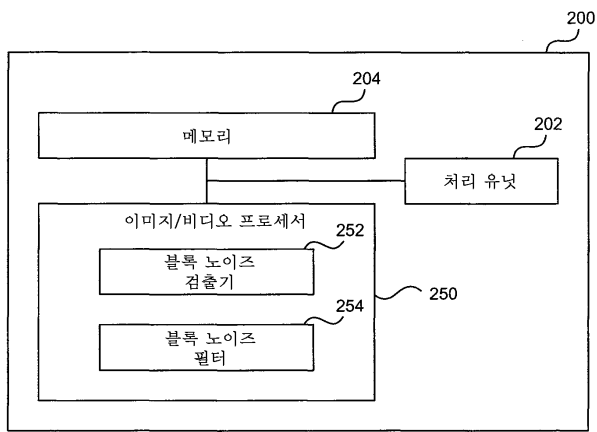
[0356] 본 개시 내용의 소정의 태양이 아래에서 소정의 청구항 형식으로 제시되지만, 발명자는 임의의 개수의 청구항 형식으로 본 개시 내용의 다양한 태양을 예상한다. 예컨대, 본 개시 내용의 하나의 태양만이 35 U.S.C. § 112, ¶ 13에 따른 기능식 청구항으로 열거되지만, 다른 태양들이 마찬가지로 기능식 청구항으로, 또는 컴퓨터 판독 가능 매체로 구현되는 것과 같은 다른 형태로 구현될 수 있다. (35 U.S.C. § 112, ¶ 13에 따라 취급되고자 하는 임의의 청구항은 "~를 위한 수단"이라는 단어로 시작할 것이다). 따라서, 출원인은 본 출원을 출원한 후에 본 개시 내용의 다른 태양에 대한 그러한 추가적인 청구항 형태를 추구하기 위해 추가 청구항을 추가할 권리를 유보한다.

도면

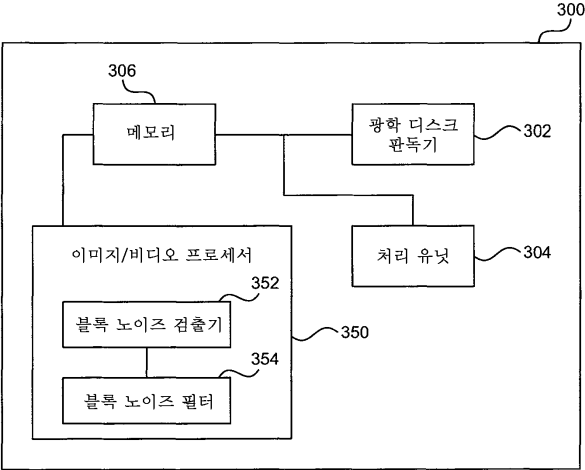
도면1



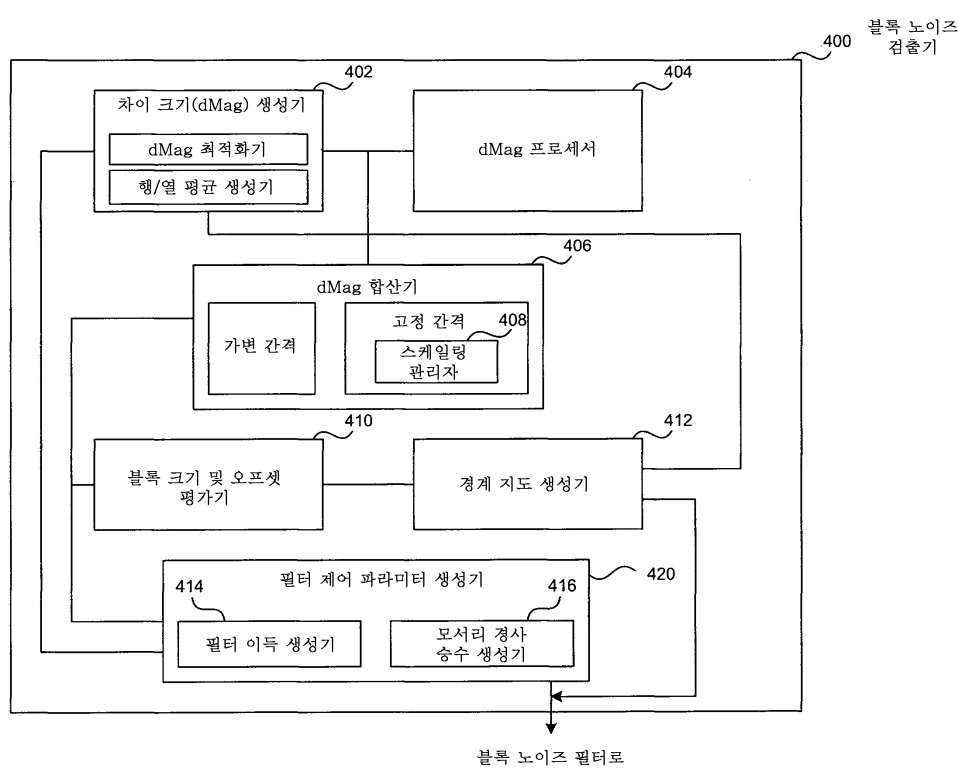
도면2



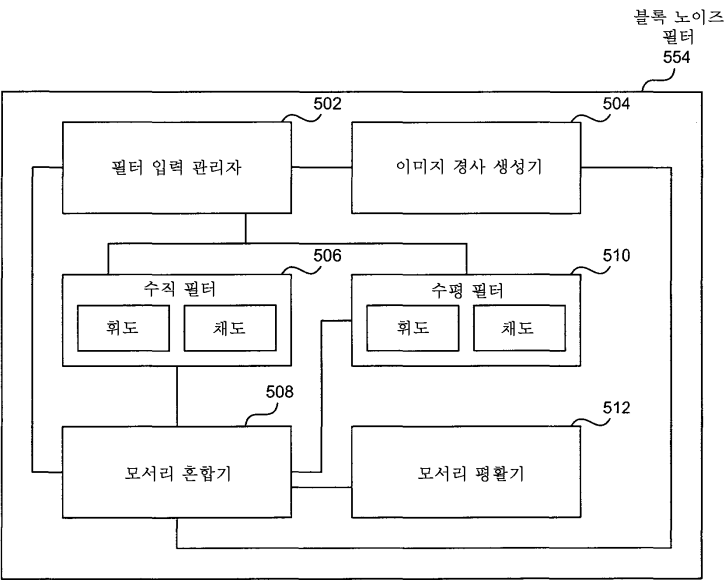
도면3



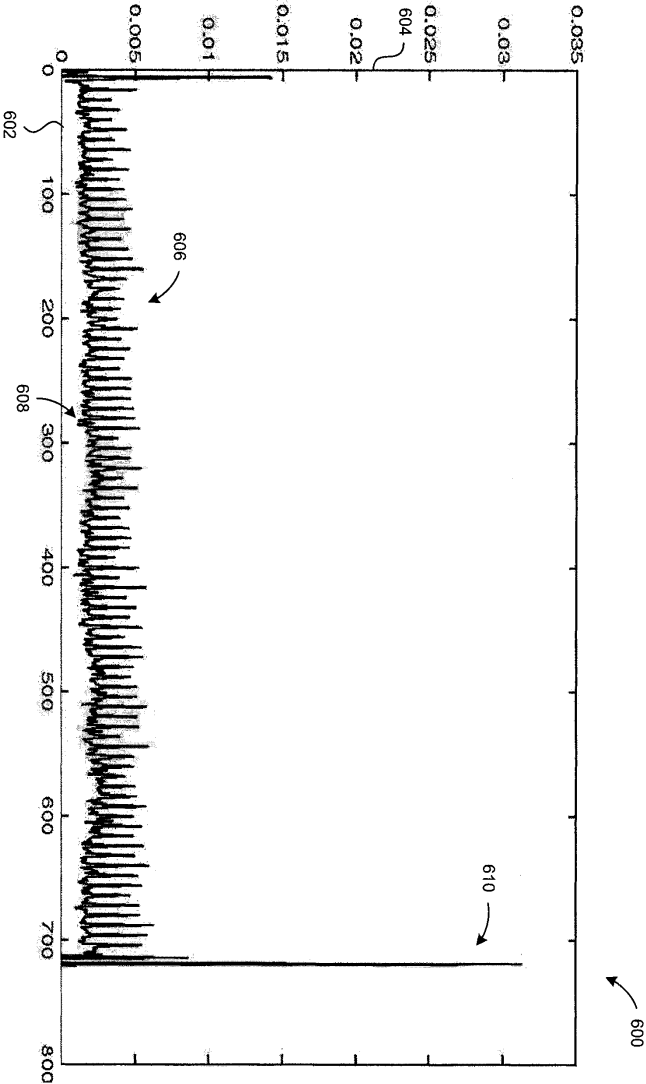
도면4



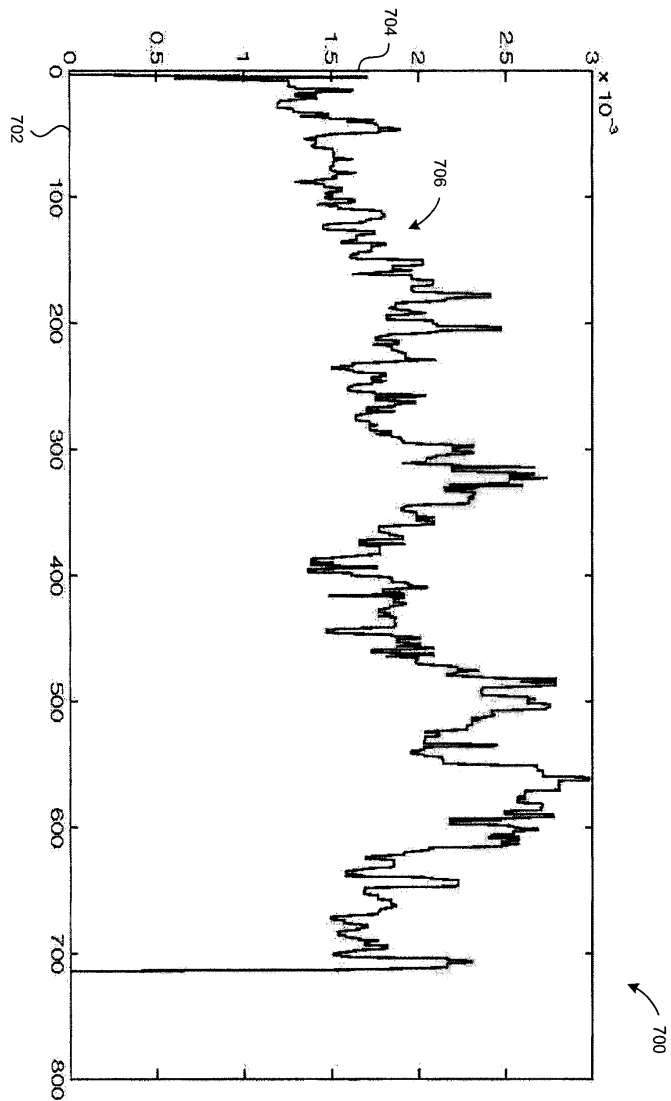
도면5



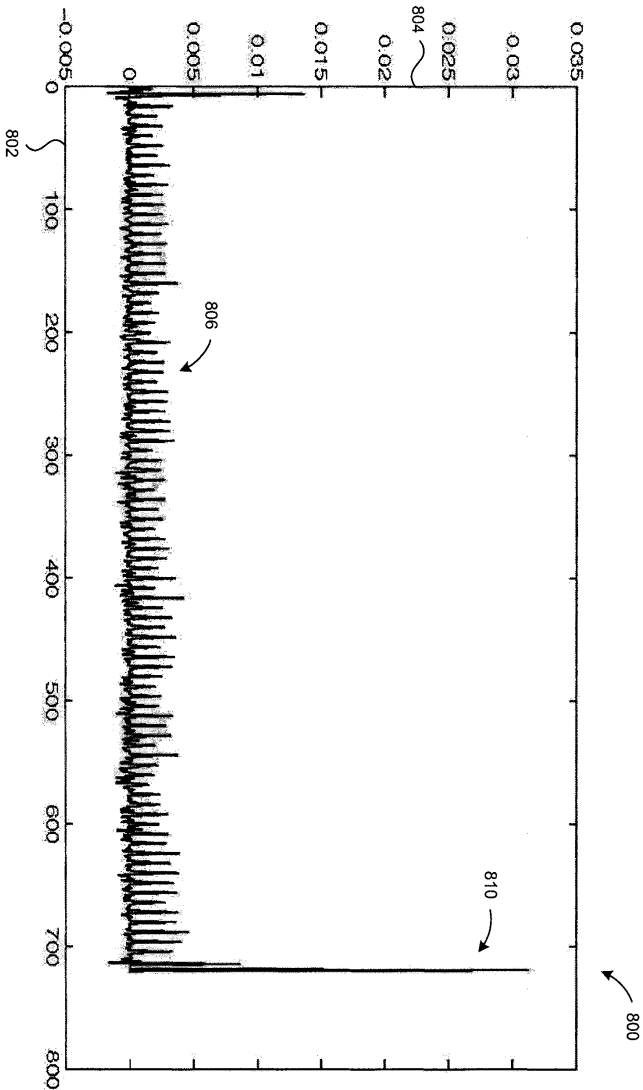
도면6



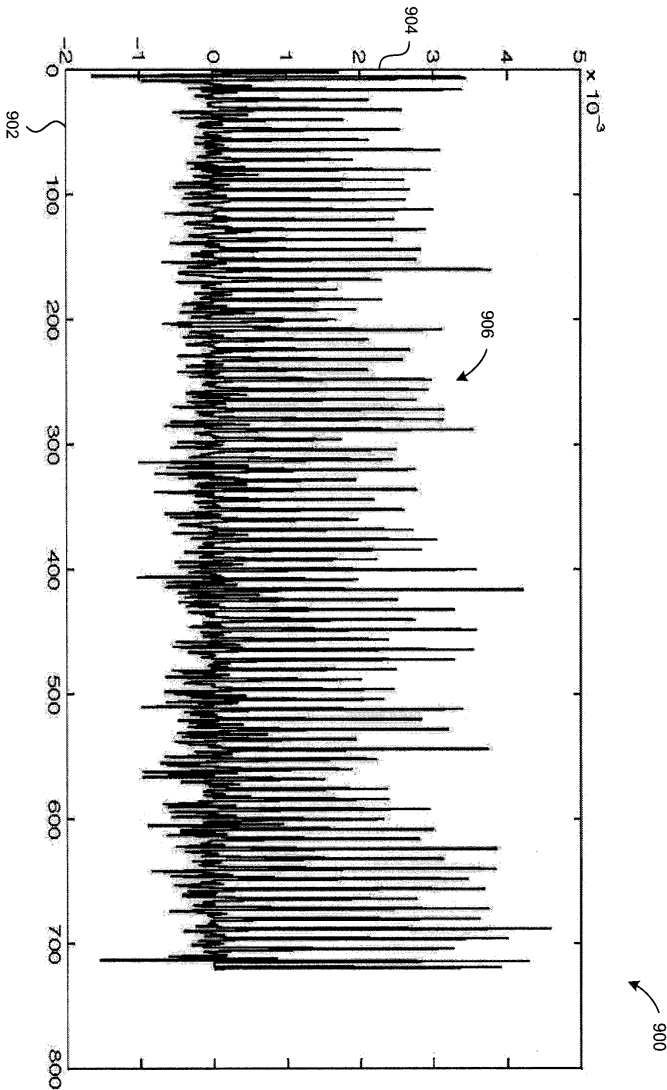
도면7



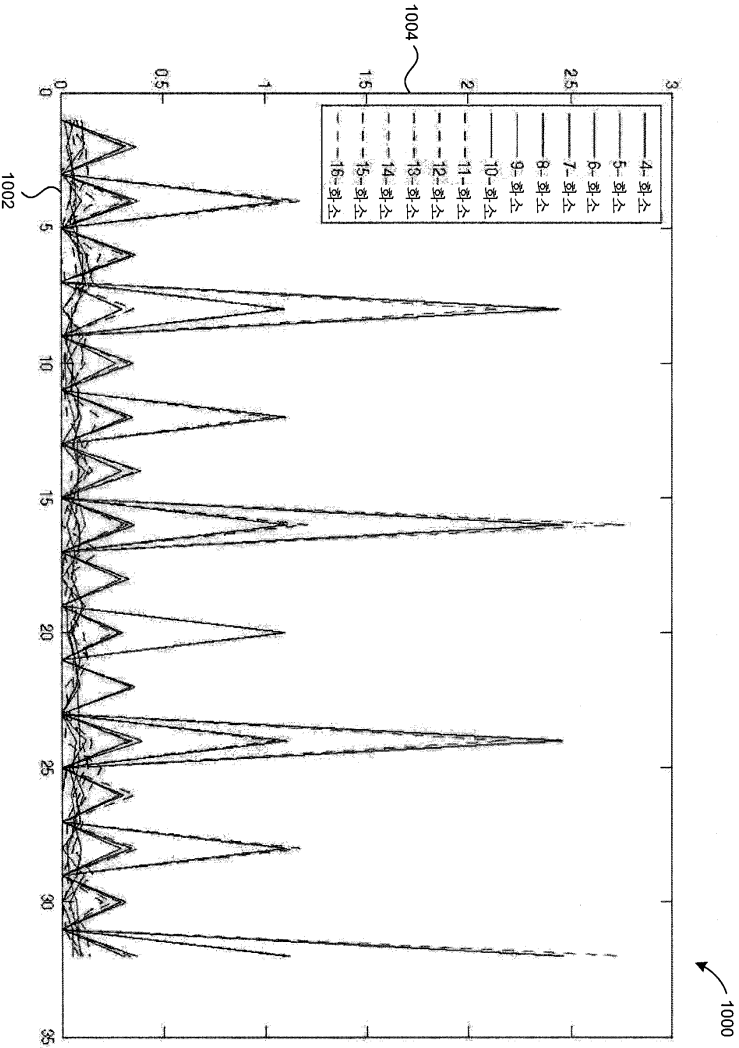
도면8



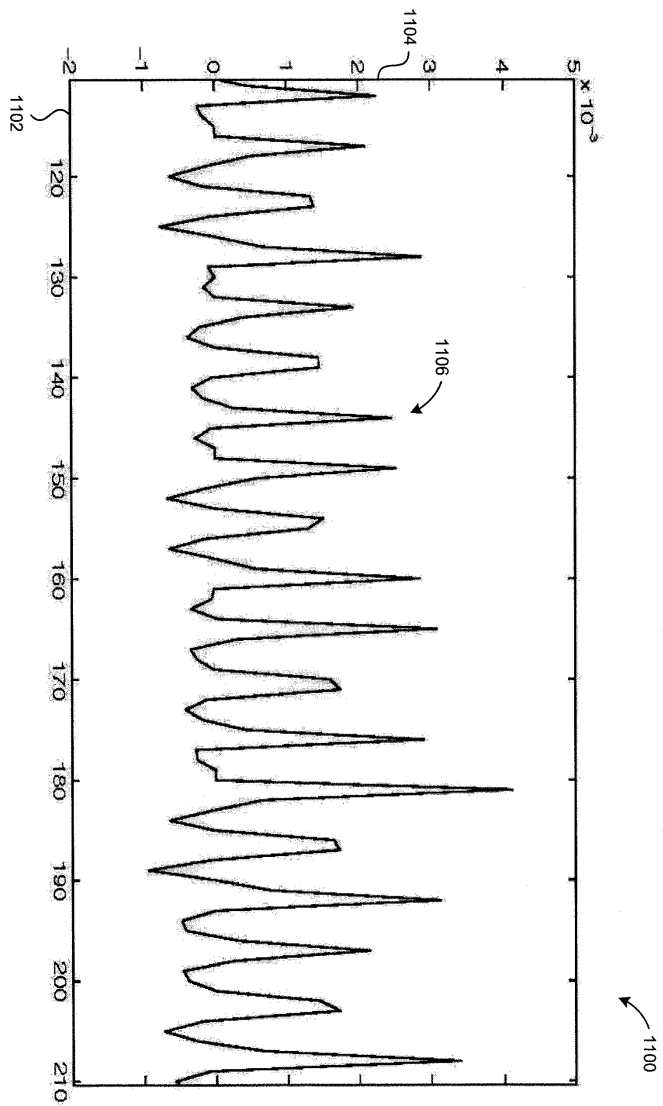
도면9



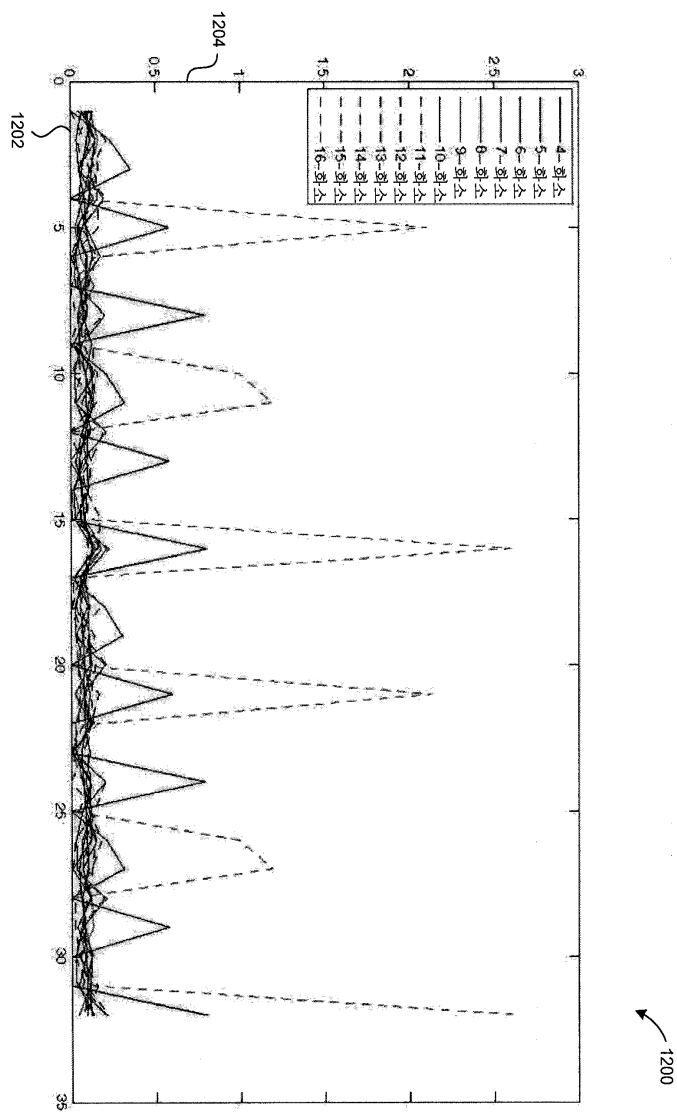
도면10



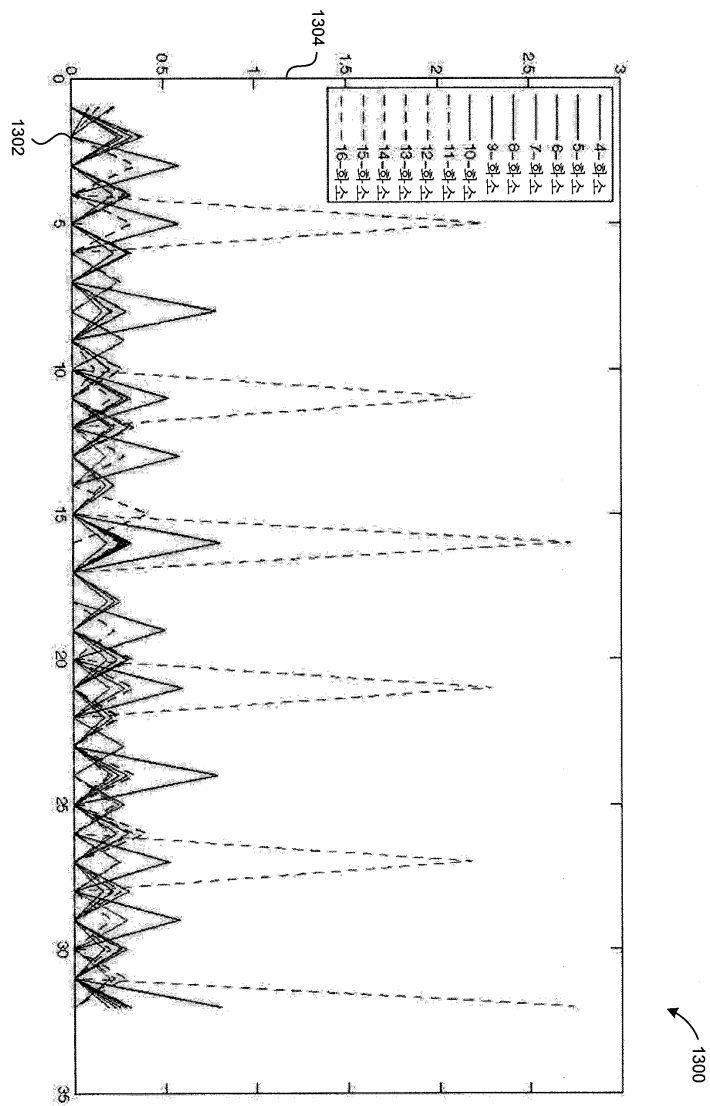
도면11



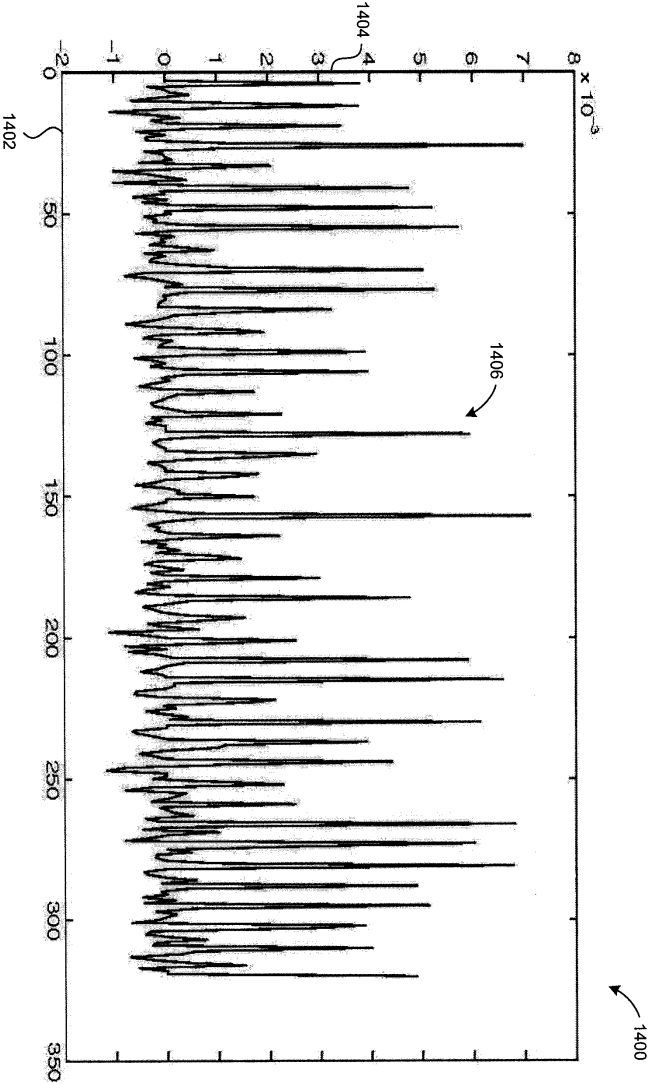
도면12



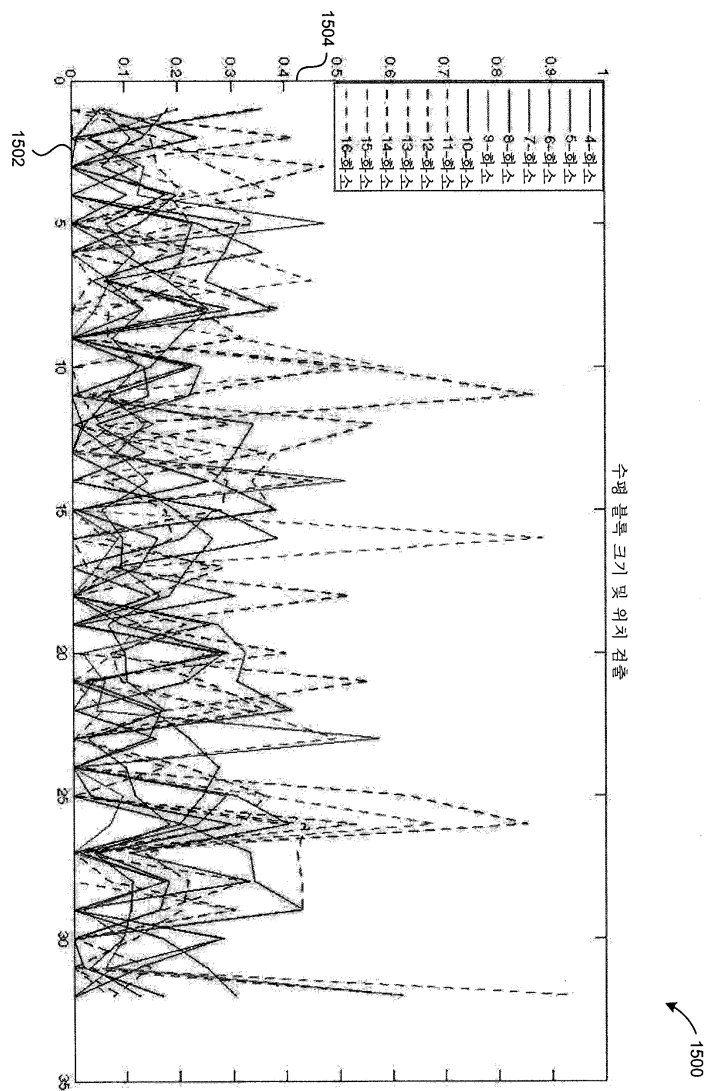
도면13



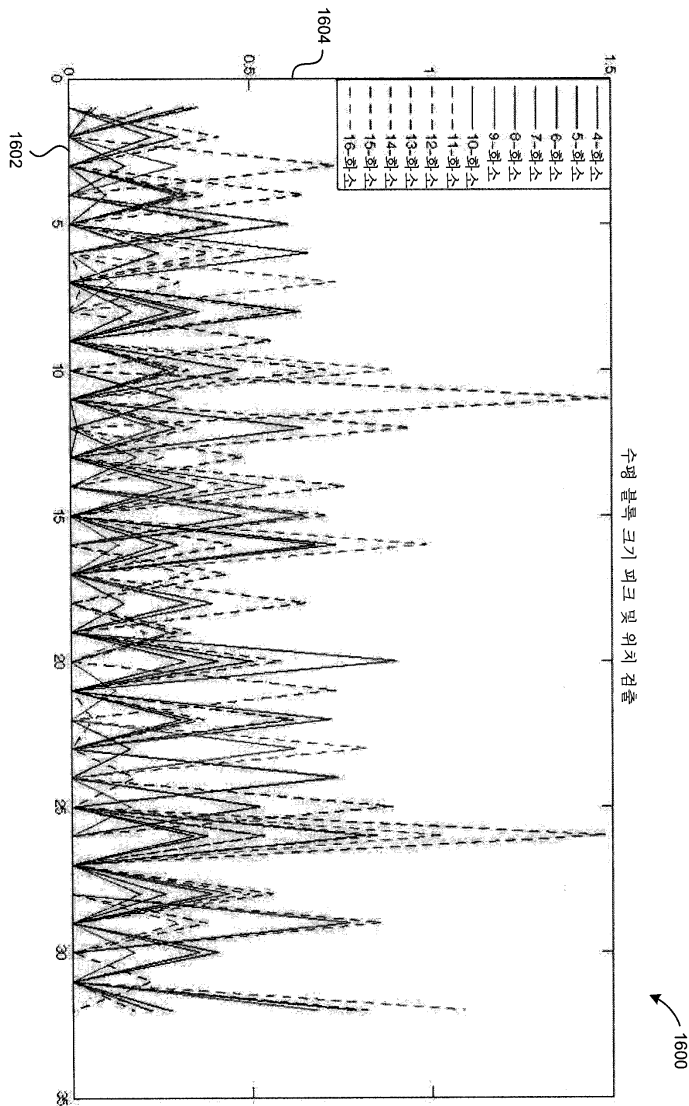
도면14



도면15



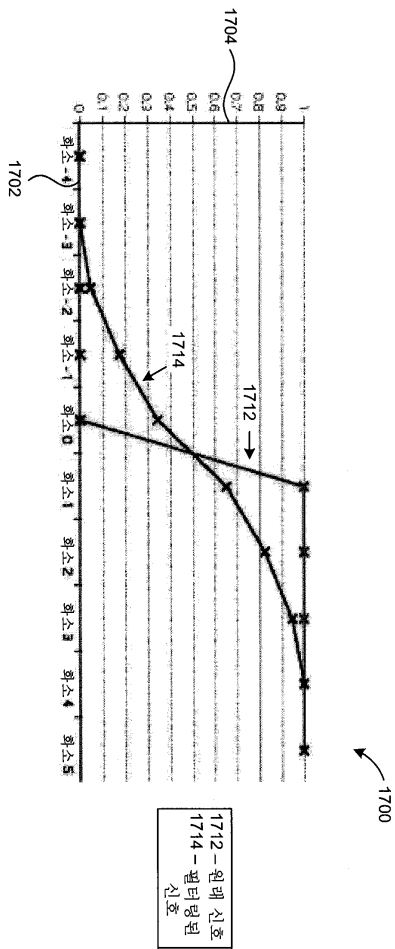
도면16a



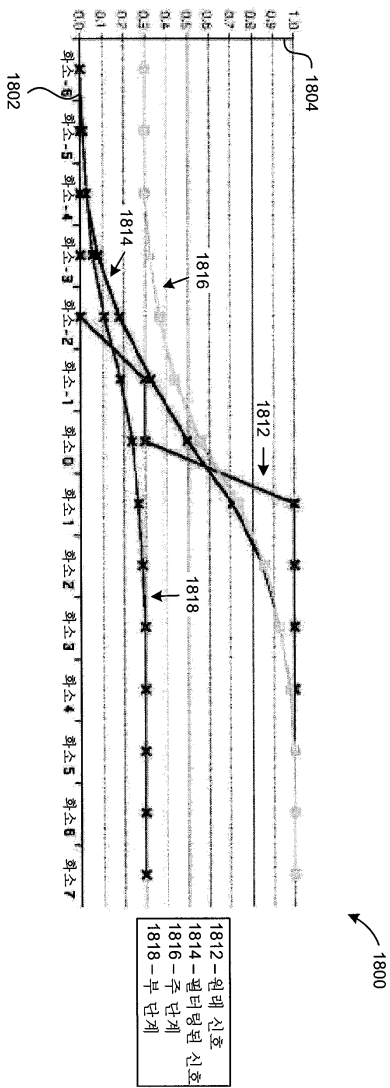
도면16b

| 구간 | -1     | 0      | +1     |
|----|--------|--------|--------|
| 4  | 1.7985 | 0.5825 | 1.0069 |
| 5  | 1.2627 | 0.3895 | 0.9570 |
| 6  | 1.1521 | 0.6720 | 1.0173 |
| 7  | 1.1908 | 0.3525 | 3.869  |
| 8  | 4.4361 | 0.5801 | 1.1058 |
| 9  | 1.2481 | 0.6040 | 1.6725 |
| 10 | 1.8644 | 0.3320 | 1.4620 |
| 11 | 1.6135 | 0.8803 | 0.6557 |
| 12 | 0.7176 | 0.6727 | 1.1608 |
| 13 | 1.2617 | 0.7048 | 0.8090 |
| 14 | 0.8741 | 0.4372 | 3.5881 |
| 15 | 3.8572 | 0.7577 | 1.0587 |
| 16 | 1.1330 | 0.7459 | 1.3802 |

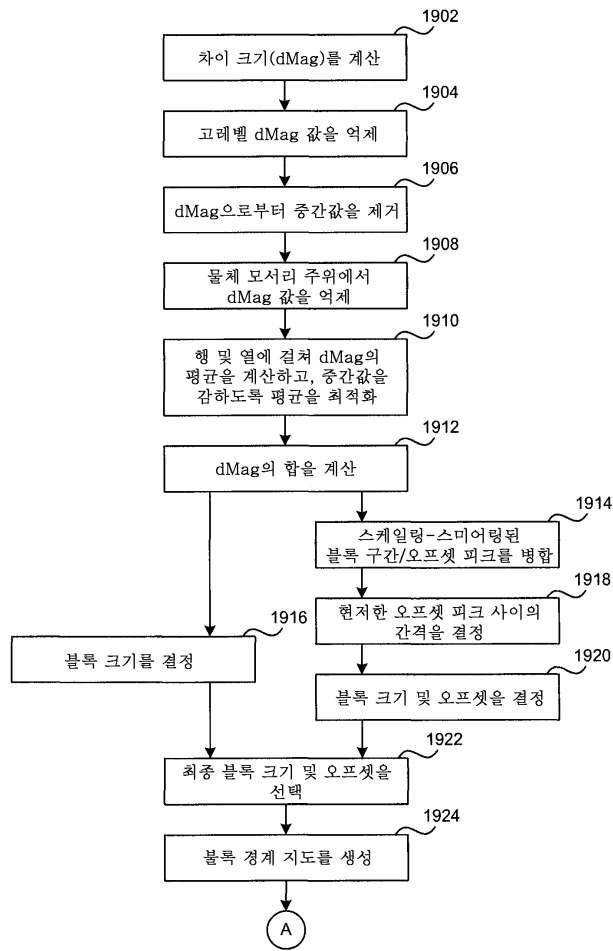
도면17



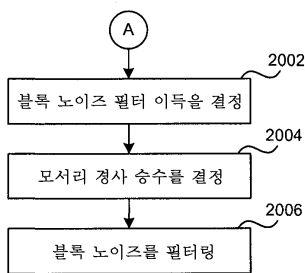
도면18



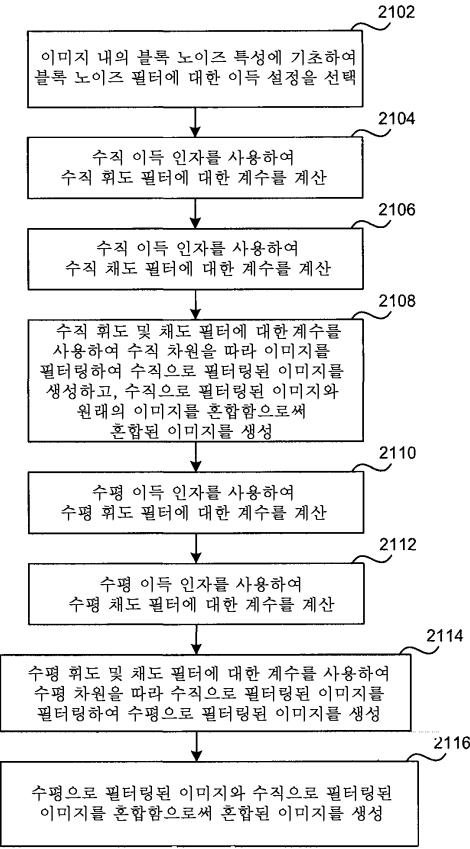
도면19



도면20



도면21



도면22

