



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2018-0032775
(43) 공개일자 2018년04월02일

(51) 국제특허분류(Int. Cl.)

H04N 19/86 (2014.01) H04N 19/117 (2014.01)
H04N 19/119 (2014.01) H04N 19/176 (2014.01)
H04N 19/44 (2014.01)

(52) CPC특허분류

H04N 19/86 (2015.01)
H04N 19/117 (2015.01)

(21) 출원번호 10-2016-0121827

(22) 출원일자 2016년09월23일

심사청구일자 없음

(71) 출원인

세종대학교산학협력단

서울특별시 광진구 능동로 209 (군자동, 세종대학교)

(72) 발명자

이영렬

서울특별시 송파구 올림픽로 135, 224동 802호(잠실동, 리센즈아파트)

김남욱

서울특별시 광진구 동일로28길 8, 402호

(뒷면에 계속)

(74) 대리인

성병기, 강남석

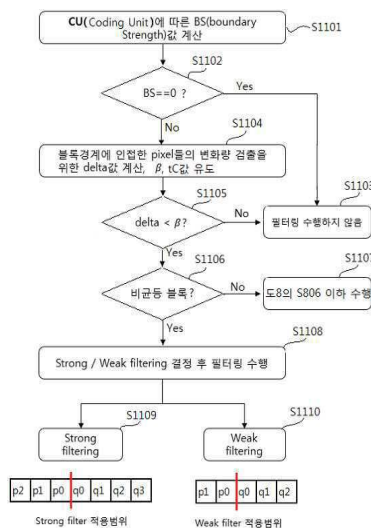
전체 청구항 수 : 총 9 항

(54) 발명의 명칭 적응적 블록 분할에 기반한 비디오 신호 처리 방법 및 장치

(57) 요약

본 발명에 따른, 입력 영상을 블록 단위로 분할하여 부호화하는 비디오 신호 처리 방법은 현재 블록에 대한 잔차 블록을 생성하고, 상기 잔차 블록을 부호화하고, 상기 부호화된 잔차 블록을 복호화하고, 상기 복호화된 잔차 블록을 이용하여 상기 현재 블록을 복원하고, 상기 복원된 현재 블록을 포함하는 복원 영상에 대해 필터링을 수행하는 단계를 포함한다. 상기 필터링은, 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여 수행된다.

대표도 - 도11



(52) CPC특허분류

H04N 19/119 (2015.01)

H04N 19/176 (2015.01)

H04N 19/44 (2015.01)

(72) 발명자

고경환

서울특별시 강동구 천중로16길 37, 2-101(천호동,
동서울연립)

유영환

서울특별시 송파구 동남로 225, 105동 302호(가락
동, 래미안파크팰리스아파트)

이 발명을 지원한 국가연구개발사업

과제고유번호 NRF-2015R1A2A2A01006085

부처명 미래창조과학부

연구관리전문기관 한국연구재단

연구사업명 중견 연구자 사업

연구과제명 낮은 복잡도를 갖는 모바일 차세대 비디오코딩 기술연구

기 여 율 1/1

주관기관 세종대학교 산학협력단

연구기간 2015.05.01 ~ 2018.04.30

명세서

청구범위

청구항 1

입력 영상을 블록 단위로 분할하여 부호화하는 비디오 신호 처리 방법에 있어서,
 현재 블록에 대한 잔차 블록을 생성하는 단계;
 상기 잔차 블록을 부호화하는 단계;
 상기 부호화된 잔차 블록을 복호화하는 단계;
 상기 복호화된 잔차 블록을 이용하여 상기 현재 블록을 복원하는 단계; 및
 상기 복원된 현재 블록을 포함하는 복원 영상에 대해 필터링을 수행하는 단계를 포함하고,
 상기 필터링은, 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여 수행되는 것을 특징으로 하는 비디오 신호 처리 방법.

청구항 2

제1 항에 있어서,
 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여, 필터링되는 화소의 개수 또는 필터링 강도를 결정하는 것을 특징으로 하는 비디오 신호 처리 방법.

청구항 3

제2 항에 있어서,
 블록 경계에 인접한 두 블록 중 적어도 하나의 블록이 비정방형인 경우, 두 블록 중 크기가 더 큰 블록에 대해 더 많은 화소를 필터링하는 것을 특징으로 하는 비디오 신호 처리 방법.

청구항 4

제2 항에 있어서,
 블록 경계에 인접한 두 블록 중 적어도 하나의 블록이 비정방형인 경우, 두 블록 중 크기가 더 큰 블록에 대해 더 강한 필터링을 적용하는 것을 특징으로 하는 비디오 신호 처리 방법.

청구항 5

제 2항에 있어서,
 블록 경계에 인접한 두 블록의 크기가 상이한 경우, 두 블록 중 크기가 더 큰 블록에 대해 더 많은 화소를 필터링하는 것을 특징으로 하는 비디오 신호 처리 방법.

청구항 6

제 2항에 있어서,
 블록 경계에 인접한 두 블록의 크기가 상이한 경우, 두 블록 중 크기가 더 큰 블록에 대해 더 강한 필터링을 적용하는 것을 특징으로 하는 비디오 신호 처리 방법.

청구항 7

입력 영상을 블록 단위로 분할하여 복호화하는 비디오 신호 처리 방법에 있어서,
 현재 블록에 대한 잔차 블록을 비트스트림으로부터 복호화하는 단계;

상기 복호화된 잔차 블록을 이용하여 상기 현재 블록을 복원하는 단계; 및

상기 복원된 현재 블록을 포함하는 복원 영상에 대해 필터링을 수행하는 단계를 포함하고,

상기 필터링은, 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여 수행되는 것을 특징으로 하는 비디오 신호 처리 방법.

청구항 8

입력 영상을 블록 단위로 분할하여 부호화하는 비디오 신호 처리 장치에 있어서,

현재 블록에 대한 잔차 블록을 생성하는 잔차 블록 생성부;

상기 잔차 블록을 부호화하는 잔차 블록 부호화부;

상기 부호화된 잔차 블록을 복호화하는 잔차 블록 복호화부;

상기 복호화된 잔차 블록을 이용하여 상기 현재 블록을 복원하는 현재 블록 복원부; 및

상기 복원된 현재 블록을 포함하는 복원 영상에 대해 필터링을 수행하는 필터링부를 포함하고,

상기 필터링부는, 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여 상기 블록 경계에 대해 필터링을 수행하는 것을 특징으로 하는 비디오 신호 처리 장치.

청구항 9

입력 영상을 블록 단위로 분할하여 복호화하는 비디오 신호 처리 장치에 있어서,

현재 블록에 대한 잔차 블록을 비트스트림으로부터 복호화하는 잔차 블록 복호화부;

상기 복호화된 잔차 블록을 이용하여 상기 현재 블록을 복원하는 현재 블록 복원부; 및

상기 복원된 현재 블록을 포함하는 복원 영상에 대해 필터링을 수행하는 필터링부를 포함하고,

상기 필터링부는, 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여 상기 블록 경계에 대해 필터링을 수행하는 것을 특징으로 하는 비디오 신호 처리 장치.

발명의 설명

기술 분야

[0001] 본 발명은 적응적 블록 분할에 기반한 비디오 신호 처리 방법 및 장치에 관한 것이다.

배경 기술

[0002] 최근 HD(High Definition) 영상 및 UHD(Ultra High Definition) 영상과 같은 고해상도, 고품질의 영상에 대한 수요가 다양한 응용 분야에서 증가하고 있다. 영상 데이터가 고해상도, 고품질이 될수록 기존의 영상 데이터에 비해 상대적으로 데이터량이 증가하기 때문에 기존의 유무선 광대역 회선과 같은 매체를 이용하여 영상 데이터를 전송하거나 기존의 저장 매체를 이용해 저장하는 경우, 전송 비용과 저장 비용이 증가하게 된다. 영상 데이터가 고해상도, 고품질화 됨에 따라 발생하는 이러한 문제들을 해결하기 위해서는 고효율의 영상 압축 기술들이 활용될 수 있다.

[0003] 영상 압축 기술로 현재 픽처의 이전 또는 이후 픽처로부터 현재 픽처에 포함된 화소값을 예측하는 화면 간 예측 기술, 현재 픽처 내의 화소 정보를 이용하여 현재 픽처에 포함된 화소값을 예측하는 화면 내 예측 기술, 출현 빈도가 높은 값에 짧은 부호를 할당하고 출현 빈도가 낮은 값에 긴 부호를 할당하는 엔트로피 부호화 기술 등 다양한 기술이 존재하고 이러한 영상 압축 기술을 이용해 영상 데이터를 효과적으로 압축하여 전송 또는 저장할 수 있다.

[0004] 한편, 고해상도 영상에 대한 수요가 증가함과 함께, 새로운 영상 서비스로서 입체 영상 콘텐츠에 대한 수요도 함께 증가하고 있다. 고해상도 및 초고해상도의 입체 영상 콘텐츠를 효과적으로 제공하기 위한 비디오 압축 기술에 대하여 논의가 진행되고 있다.

발명의 내용

해결하려는 과제

- [0005] 본 발명은 입력 영상의 적응적 분할에 기반하여 입력 영상을 부호화/복호화하는 방법 및 장치를 제공하는 것을 목적으로 한다.
- [0006] 또한, 본 발명은 입력 영상의 적응적 분할을 시그널링하는 방법 및 장치를 제공하는 것을 목적으로 한다.
- [0007] 또한, 본 발명은 입력 영상의 적응적 분할에 따른 블록에 대해 변환 및/또는 필터링을 수행하는 방법 및 장치를 제공하는 것을 목적으로 한다.

과제의 해결 수단

- [0008] 본 발명에 따른, 입력 영상을 블록 단위로 분할하여 부호화하는 비디오 신호 처리 방법은, 현재 블록에 대한 잔차 블록을 생성하고, 상기 잔차 블록을 부호화하고, 상기 부호화된 잔차 블록을 복호화하고, 상기 복호화된 잔차 블록을 이용하여 상기 현재 블록을 복원하고, 상기 복원된 현재 블록을 포함하는 복원 영상에 대해 필터링을 수행하며, 상기 필터링은, 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여 수행되는 것을 특징으로 한다.
- [0009] 본 발명에 따른 비디오 신호 처리 방법에 있어서, 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여, 필터링되는 화소의 개수 또는 필터링 강도를 결정하는 것을 특징으로 한다.
- [0010] 본 발명에 따른 비디오 신호 처리 방법에 있어서, 블록 경계에 인접한 두 블록 중 적어도 하나의 블록이 비정방형인 경우, 두 블록 중 크기가 더 큰 블록에 대해 더 많은 화소를 필터링하는 것을 특징으로 한다.
- [0011] 본 발명에 따른 비디오 신호 처리 방법에 있어서, 블록 경계에 인접한 두 블록 중 적어도 하나의 블록이 비정방형인 경우, 두 블록 중 크기가 더 큰 블록에 대해 더 강한 필터링을 적용하는 것을 특징으로 한다.
- [0012] 본 발명에 따른 비디오 신호 처리 방법에 있어서, 블록 경계에 인접한 두 블록의 크기가 상이한 경우, 두 블록 중 크기가 더 큰 블록에 대해 더 많은 화소를 필터링하는 것을 특징으로 한다.
- [0013] 본 발명에 따른 비디오 신호 처리 방법에 있어서, 블록 경계에 인접한 두 블록의 크기가 상이한 경우, 두 블록 중 크기가 더 큰 블록에 대해 더 강한 필터링을 적용하는 것을 특징으로 한다.
- [0014] 본 발명에 따른, 입력 영상을 블록 단위로 분할하여 복호화하는 비디오 신호 처리 방법은, 현재 블록에 대한 잔차 블록을 비트스트림으로부터 복호화하고, 상기 복호화된 잔차 블록을 이용하여 상기 현재 블록을 복원하고, 상기 복원된 현재 블록을 포함하는 복원 영상에 대해 필터링을 수행하며, 상기 필터링은, 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여 수행되는 것을 특징으로 한다.
- [0015] 본 발명에 따른, 입력 영상을 블록 단위로 분할하여 부호화하는 비디오 신호 처리 장치는, 현재 블록에 대한 잔차 블록을 생성하는 잔차 블록 생성부, 상기 잔차 블록을 부호화하는 잔차 블록 부호화부, 상기 부호화된 잔차 블록을 복호화하는 잔차 블록 복호화부, 상기 복호화된 잔차 블록을 이용하여 상기 현재 블록을 복원하는 현재 블록 복원부, 및 상기 복원된 현재 블록을 포함하는 복원 영상에 대해 필터링을 수행하는 필터링부를 포함하고, 상기 필터링부는, 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여 상기 블록 경계에 대해 필터링을 수행하는 것을 특징으로 한다.
- [0016] 본 발명에 따른, 입력 영상을 블록 단위로 분할하여 복호화하는 비디오 신호 처리 장치는, 현재 블록에 대한 잔차 블록을 비트스트림으로부터 복호화하는 잔차 블록 복호화부, 상기 복호화된 잔차 블록을 이용하여 상기 현재 블록을 복원하는 현재 블록 복원부, 및 상기 복원된 현재 블록을 포함하는 복원 영상에 대해 필터링을 수행하는 필터링부를 포함하고, 상기 필터링부는, 블록 경계에 인접한 두 블록의 형태 또는 크기에 기초하여 상기 블록 경계에 대해 필터링을 수행하는 것을 특징으로 한다.

발명의 효과

- [0017] 본 발명에 의하면, 쿼드 트리 구조 및/또는 이진 트리 구조에 기반하여 블록을 적응적으로 분할함으로써 부호화 효율을 향상시킬 수 있다.
- [0018] 또한, 본 발명에 의하면, 입력 영상의 적응적 분할에 따른 블록의 분할 정보를 효율적으로 시그널링하여 부호화

효율을 향상시킬 수 있다.

[0019] 또한, 본 발명에 의하면, 입력 영상의 적응적 분할에 따라, 임의의 형태를 갖는 블록에 대한 변환 및/또는 필터링을 효율적으로 수행하여 부호화 효율을 향상시킬 수 있다.

도면의 간단한 설명

[0020] 도 1은 본 발명의 일 실시예에 따른 영상 부호화 장치를 나타낸 블록도이다.

도 2는 본 발명의 일 실시예에 따른 영상 복호화 장치를 나타낸 블록도이다.

도 3(a)는 입력 영상의 기본 블록을 쿼드 트리 구조 (Quad Tree Structure)를 이용하여 분할한 구조를 예시적으로 나타내는 도면이다.

도 3(b)는 입력 영상의 기본 블록을 쿼드 트리 구조 및/또는 이진 트리 구조를 사용하여 분할한 구조를 예시적으로 나타내는 도면이다.

도 4(a)는 쿼드 트리 구조를 주 분할 구조로, 이진 트리 구조를 부 분할 구조로 이용하여 입력 영상의 기본 블록을 복수의 서브 블록으로 분할한 구조를 예시적으로 나타내는 도면이다.

도 4(b)는 본 발명의 일 실시예에 따라, 도 4(a)에 도시된 블록 분할 구조에 대한 블록 분할 정보를 트리 구조를 이용하여 예시적으로 나타내는 도면이다.

도 4(c)는 본 발명의 다른 실시예에 따라, 도 4(a)에 도시된 블록 분할 구조에 대한 블록 분할 정보를 트리 구조를 이용하여 예시적으로 나타내는 도면이다.

도 5는 입력 영상의 기본 블록을 쿼드 트리 구조 및/또는 이진 트리 구조를 사용하여 복수의 서브 블록으로 분할한 경우, 분할된 블록이 취할 수 있는 다양한 블록의 크기 및 형태를 예시적으로 나타내는 도면이다.

도 6은 변환에 사용될 수 있는 DCT-II의 실수 기저 및 실수 기저에 소정의 값을 곱하여 얻어지는 정수 기저를 나타내는 예시적인 도면이다.

도 7은 변환에 사용될 수 있는 DST-VII의 실수 기저 및 실수 기저에 소정의 값을 곱하여 얻어지는 정수 기저를 나타내는 예시적인 도면이다.

도 8은 본 발명의 일 실시예에 따른, 필터링을 설명하기 위한 플로우 차트이다.

도 9는 도 8에 따른 필터링을 수행하기 위해 사용되는, 블록 경계에 인접한 두 블록 및 그 내부의 픽셀을 나타내는 도면이다.

도 10은 본 발명에 따라 쿼드 트리 구조 및/또는 이진 트리 구조를 이용하여 분할된 블록의 구조 및 그에 따른 필터링이 적용되는 블록 경계들을 나타낸 도면이다.

도 11은 본 발명의 다른 실시예에 따른, 필터링을 설명하기 위한 플로우 차트이다.

도 12는 본 발명의 실시예에 따라 강한 필터링이 적용되는 화소를 예시하는 도면이다.

도 13은 본 발명의 실시예에 따라 약한 필터링이 적용되는 화소를 예시하는 도면이다.

도 14는 본 발명에 따라 약한 필터링이 수행되는 경우에 필터링이 적용되는 픽셀의 범위를 예시적으로 나타내는 도면이다.

발명을 실시하기 위한 구체적인 내용

[0021] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세한 설명에 상세하게 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 각 도면을 설명하면서 유사한 참조부호를 유사한 구성요소에 대해 사용하였다.

[0022] 제1, 제2 등의 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다. 및/또는 이라는 용어는 복수의 관련된 기재된 항목들의

조합 또는 복수의 관련된 기재된 항목들 중의 어느 항목을 포함한다.

- [0023] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.
- [0024] 본 출원에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 출원에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0025] 이하, 첨부한 도면들을 참조하여, 본 발명의 바람직한 실시예를 보다 상세하게 설명하고자 한다. 이하, 도면상의 동일한 구성요소에 대해서는 동일한 참조부호를 사용하고 동일한 구성요소에 대해서 중복된 설명은 생략한다.
- [0026] 도 1은 본 발명의 일실시예에 따른 영상 부호화 장치를 나타낸 블록도이다.
- [0027] 도 1을 참조하면, 영상 부호화 장치(100)는 픽처 분할부(110), 예측부(120, 125), 변환부(130), 양자화부(135), 재정렬부(160), 엔트로피 부호화부(165), 역양자화부(140), 역변환부(145), 필터부(150) 및 메모리(155)를 포함할 수 있다.
- [0028] 도 1에 나타난 각 구성부들은 영상 부호화 장치에서 서로 다른 특징적인 기능들을 나타내기 위해 독립적으로 도시한 것으로, 각 구성부들이 분리된 하드웨어나 하나의 소프트웨어 구성단위로 이루어짐을 의미하지 않는다. 즉, 각 구성부는 설명의 편의상 각각의 구성부로 나열하여 포함한 것으로 각 구성부 중 적어도 두 개의 구성부가 합쳐져 하나의 구성부로 이루어지거나, 하나의 구성부가 복수개의 구성부로 나뉘어져 기능을 수행할 수 있고 이러한 각 구성부의 통합된 실시예 및 분리된 실시예도 본 발명의 본질에서 벗어나지 않는 한 본 발명의 권리범위에 포함된다.
- [0029] 또한, 일부의 구성 요소는 본 발명에서 본질적인 기능을 수행하는 필수적인 구성 요소는 아니고 단지 성능을 향상시키기 위한 선택적 구성 요소일 수 있다. 본 발명은 단지 성능 향상을 위해 사용되는 구성 요소를 제외한 본 발명의 본질을 구현하는데 필수적인 구성부만을 포함하여 구현될 수 있고, 단지 성능 향상을 위해 사용되는 선택적 구성 요소를 제외한 필수 구성 요소만을 포함한 구조도 본 발명의 권리범위에 포함된다.
- [0030] 픽처 분할부(110)는 입력된 픽처를 적어도 하나의 처리 단위로 분할할 수 있다. 이때, 처리 단위는 예측 단위(Prediction Unit: PU)일 수도 있고, 변환 단위(Transform Unit: TU)일 수도 있으며, 부호화 단위(Coding Unit: CU)일 수도 있다. 픽처 분할부(110)에서는 하나의 픽처에 대해 복수의 부호화 단위, 예측 단위 및 변환 단위의 조합으로 분할하고 소정의 기준(예를 들어, 비용 함수)으로 하나의 부호화 단위, 예측 단위 및 변환 단위의 조합을 선택하여 픽처를 부호화 할 수 있다.
- [0031] 예를 들어, 하나의 픽처는 복수개의 부호화 단위로 분할될 수 있다. 픽처에서 부호화 단위를 분할하기 위해서는 쿼드 트리 구조(Quad Tree Structure)와 같은 재귀적인 트리 구조를 사용할 수 있는데 하나의 영상 또는 최대 크기 부호화 단위(largest coding unit)를 루트로 하여 다른 부호화 단위로 분할되는 부호화 유닛은 분할된 부호화 단위의 개수만큼의 자식 노드를 가지고 분할될 수 있다. 일정한 제한에 따라 더 이상 분할되지 않는 부호화 단위는 리프 노드가 된다. 즉, 하나의 코딩 유닛에 대하여 정방형 분할만이 가능하다고 가정하는 경우, 하나의 부호화 단위는 최대 4개의 다른 부호화 단위로 분할될 수 있다.
- [0032] 픽처에서 부호화 단위를 분할하기 위해서는 쿼드 트리 구조 및/또는 이진 트리 구조(Binary Tree Structure)를 사용할 수 있다. 하나의 영상 또는 최대 크기 부호화 단위를 루트로 하여 쿼드 트리 구조 및/또는 이진 트리 구조에 따른 분할이 가능하다. 분할된 블록에 대해서는 다시 쿼드 트리 구조 및/또는 이진 트리 구조를 재귀적으로 또는 계층적으로 적용할 수 있다. 더 이상 분할되지 않는 블록은 리프 노드로서, 예측, 변환 및/또는 양자화의 단위가 될 수 있다. 쿼드 트리 구조 및/또는 이진 트리 구조를 사용하여 블록을 분할하는 경우, 리프 노드는 정방형뿐만 아니라 비정방형일 수 있다.
- [0033] 이하, 본 발명의 실시예에서는 부호화 단위는 부호화를 수행하는 단위의 의미로 사용할 수도 있고, 복호화를 수행하는 단위의 의미로 사용할 수도 있다.

- [0034] 예측 단위는 하나의 부호화 단위 내에서 동일한 크기의 적어도 하나의 정사각형 또는 직사각형 등의 형태를 가지고 분할된 것일 수도 있고, 하나의 부호화 단위 내에서 분할된 예측 단위 중 어느 하나의 예측 단위가 다른 하나의 예측 단위와 상이한 형태 및/또는 크기를 가지도록 분할된 것일 수도 있다.
- [0035] 부호화 단위를 기초로 인트라 예측을 수행하는 예측 단위를 생성시 최소 부호화 단위가 아닌 경우, 복수의 예측 단위 NxN 으로 분할하지 않고 인트라 예측을 수행할 수 있다.
- [0036] 예측부(120, 125)는 인터 예측을 수행하는 인터 예측부(120)와 인트라 예측을 수행하는 인트라 예측부(125)를 포함할 수 있다. 예측 단위에 대해 인터 예측을 사용할 것인지 또는 인트라 예측을 수행할 것인지를 결정하고, 각 예측 방법에 따른 구체적인 정보(예컨대, 인트라 예측 모드, 모션 벡터, 참조 픽처 등)를 결정할 수 있다. 이때, 예측이 수행되는 처리 단위와 예측 방법 및 구체적인 내용이 정해지는 처리 단위는 다를 수 있다. 예컨대, 예측의 방법과 예측 모드 등은 예측 단위로 결정되고, 예측의 수행은 변환 단위로 수행될 수도 있다. 생성된 예측 블록과 원본 블록 사이의 잔차값(잔차 블록)은 변환부(130)로 입력될 수 있다. 또한, 예측을 위해 사용한 예측 모드 정보, 모션 벡터 정보 등은 잔차값과 함께 엔트로피 부호화부(165)에서 부호화되어 복호화기에 전달될 수 있다. 특정한 부호화 모드를 사용할 경우, 예측부(120, 125)를 통해 예측 블록을 생성하지 않고, 원본 블록을 그대로 부호화하여 복호화부에 전송하는 것도 가능하다.
- [0037] 인터 예측부(120)는 현재 픽처의 이전 픽처 또는 이후 픽처 중 적어도 하나의 픽처의 정보를 기초로 예측 단위를 예측할 수도 있고, 경우에 따라서는 현재 픽처 내의 부호화가 완료된 일부 영역의 정보를 기초로 예측 단위를 예측할 수도 있다. 인터 예측부(120)는 참조 픽처 보간부, 모션 예측부, 움직임 보상부를 포함할 수 있다.
- [0038] 참조 픽처 보간부에서는 메모리(155)로부터 참조 픽처 정보를 제공받고 참조 픽처에서 정수 화소 이하의 화소 정보를 생성할 수 있다. 휘도 화소의 경우, 1/4 화소 단위로 정수 화소 이하의 화소 정보를 생성하기 위해 필터 계수를 달리하는 DCT 기반의 8탭 보간 필터(DCT-based Interpolation Filter)가 사용될 수 있다. 색차 신호의 경우 1/8 화소 단위로 정수 화소 이하의 화소 정보를 생성하기 위해 필터 계수를 달리하는 DCT 기반의 4탭 보간 필터(DCT-based Interpolation Filter)가 사용될 수 있다.
- [0039] 모션 예측부는 참조 픽처 보간부에 의해 보간된 참조 픽처를 기초로 모션 예측을 수행할 수 있다. 모션 벡터를 산출하기 위한 방법으로 FBMA(Full search-based Block Matching Algorithm), TSS(Three Step Search), NTS(New Three-Step Search Algorithm) 등 다양한 방법이 사용될 수 있다. 모션 벡터는 보간된 화소를 기초로 1/2 또는 1/4 화소 단위의 모션 벡터값을 가질 수 있다. 모션 예측부에서는 모션 예측 방법을 다르게 하여 현재 예측 단위를 예측할 수 있다. 모션 예측 방법으로 스킵(Skip) 방법, 머지(Merge) 방법, AMVP(Advanced Motion Vector Prediction) 방법, 인트라 블록 카피(Intra Block Copy) 방법 등 다양한 방법이 사용될 수 있다.
- [0040] 인트라 예측부(125)는 현재 픽처 내의 화소 정보인 현재 블록 주변의 참조 픽셀 정보를 기초로 예측 단위를 생성할 수 있다. 현재 예측 단위의 주변 블록이 인터 예측을 수행한 블록이어서, 참조 픽셀이 인터 예측을 수행한 픽셀일 경우, 인터 예측을 수행한 블록에 포함되는 참조 픽셀을 주변의 인트라 예측을 수행한 블록의 참조 픽셀 정보로 대체하여 사용할 수 있다. 즉, 참조 픽셀이 가용하지 않는 경우, 가용하지 않은 참조 픽셀 정보를 가용한 참조 픽셀 중 적어도 하나의 참조 픽셀로 대체하여 사용할 수 있다.
- [0041] 인트라 예측에서 예측 모드는 참조 픽셀 정보를 예측 방향에 따라 사용하는 방향성 예측 모드와 예측을 수행시 방향성 정보를 사용하지 않는 비방향성 모드를 가질 수 있다. 휘도 정보를 예측하기 위한 모드와 색차 정보를 예측하기 위한 모드가 상이할 수 있고, 색차 정보를 예측하기 위해 휘도 정보를 예측하기 위해 사용된 인트라 예측 모드 정보 또는 예측된 휘도 신호 정보를 활용할 수 있다.
- [0042] 인트라 예측을 수행할 때 예측 단위의 크기와 변환 단위의 크기가 동일할 경우, 예측 단위의 좌측에 존재하는 픽셀, 좌측 상단에 존재하는 픽셀, 상단에 존재하는 픽셀을 기초로 예측 단위에 대한 인트라 예측을 수행할 수 있다. 그러나 인트라 예측을 수행할 때 예측 단위의 크기와 변환 단위의 크기가 상이할 경우, 변환 단위를 기초로 한 참조 픽셀을 이용하여 인트라 예측을 수행할 수 있다. 또한, 최소 부호화 단위에 대해서만 N x N 분할을 사용하는 인트라 예측을 사용할 수 있다.
- [0043] 인트라 예측 방법은 예측 모드에 따라 참조 화소에 AIS(Adaptive Intra Smoothing) 필터를 적용한 후 예측 블록을 생성할 수 있다. 참조 화소에 적용되는 AIS 필터의 종류는 상이할 수 있다. 인트라 예측 방법을 수행하기 위해 현재 예측 단위의 인트라 예측 모드는 현재 예측 단위의 주변에 존재하는 예측 단위의 인트라 예측 모드로부터 예측할 수 있다. 주변 예측 단위로부터 예측된 모드 정보를 이용하여 현재 예측 단위의 예측 모드를 예측하는 경우, 현재 예측 단위와 주변 예측 단위의 인트라 예측 모드가 동일하면 소정의 플래그 정보를 이용하여 현

재 예측 단위와 주변 예측 단위의 예측 모드가 동일하다는 정보를 전송할 수 있고, 만약 현재 예측 단위와 주변 예측 단위의 예측 모드가 상이하면 엔트로피 부호화를 수행하여 현재 블록의 예측 모드 정보를 부호화할 수 있다.

- [0044] 또한, 예측부(120, 125)에서 생성된 예측 단위를 기초로 예측을 수행한 예측 단위와 예측 단위의 원본 블록과 차이값인 잔차값(Residual) 정보를 포함하는 잔차 블록이 생성될 수 있다. 생성된 잔차 블록은 변환부(130)로 입력될 수 있다.
- [0045] 변환부(130)에서는 원본 블록과 예측부(120, 125)를 통해 생성된 예측 단위의 잔차값(residual)정보를 포함한 잔차 블록을 DCT(Discrete Cosine Transform), DST(Discrete Sine Transform), KLT와 같은 변환 방법을 사용하여 변환시킬 수 있다. 잔차 블록을 변환하기 위해 DCT를 적용할지, DST를 적용할지 또는 KLT를 적용할지는 잔차 블록을 생성하기 위해 사용된 예측 단위의 인트라 예측 모드 정보를 기초로 결정할 수 있다.
- [0046] 양자화부(135)는 변환부(130)에서 주파수 영역으로 변환된 값들을 양자화할 수 있다. 블록에 따라 또는 영상의 중요도에 따라 양자화 계수는 변할 수 있다. 양자화부(135)에서 산출된 값은 역양자화부(140)와 재정렬부(160)에 제공될 수 있다.
- [0047] 재정렬부(160)는 양자화된 잔차값에 대해 계수값의 재정렬을 수행할 수 있다.
- [0048] 재정렬부(160)는 계수 스캐닝(Coefficient Scanning) 방법을 통해 2차원의 블록 형태 계수를 1차원의 벡터 형태로 변경할 수 있다. 예를 들어, 재정렬부(160)에서는 지그-재그 스캔(Zig-Zag Scan)방법을 이용하여 DC 계수부터 고주파수 영역의 계수까지 스캔하여 1차원 벡터 형태로 변경시킬 수 있다. 변환 단위의 크기 및 인트라 예측 모드에 따라 지그-재그 스캔 대신 2차원의 블록 형태 계수를 열 방향으로 스캔하는 수직 스캔, 2차원의 블록 형태 계수를 행 방향으로 스캔하는 수평 스캔이 사용될 수도 있다. 즉, 변환 단위의 크기 및 인트라 예측 모드에 따라 지그-재그 스캔, 수직 방향 스캔 및 수평 방향 스캔 중 어떠한 스캔 방법이 사용될지 여부를 결정할 수 있다.
- [0049] 엔트로피 부호화부(165)는 재정렬부(160)에 의해 산출된 값들을 기초로 엔트로피 부호화를 수행할 수 있다. 엔트로피 부호화는 예를 들어, 지수 골롬(Exponential Golomb), CAVLC(Context-Adaptive Variable Length Coding), CABAC(Context-Adaptive Binary Arithmetic Coding)과 같은 다양한 부호화 방법을 사용할 수 있다.
- [0050] 엔트로피 부호화부(165)는 재정렬부(160) 및 예측부(120, 125)로부터 부호화 단위의 잔차값 계수 정보 및 블록 타입 정보, 예측 모드 정보, 분할 단위 정보, 예측 단위 정보 및 전송 단위 정보, 모션 벡터 정보, 참조 프레임 정보, 블록의 보간 정보, 필터링 정보 등 다양한 정보를 부호화할 수 있다.
- [0051] 엔트로피 부호화부(165)에서는 재정렬부(160)에서 입력된 부호화 단위의 계수값을 엔트로피 부호화할 수 있다.
- [0052] 역양자화부(140) 및 역변환부(145)에서는 양자화부(135)에서 양자화된 값들을 역양자화하고 변환부(130)에서 변환된 값들을 역변환한다. 역양자화부(140) 및 역변환부(145)에서 생성된 잔차값(Residual)은 예측부(120, 125)에 포함된 움직임 추정부, 움직임 보상부 및 인트라 예측부를 통해서 예측된 예측 단위와 합쳐져 복원 블록(Reconstructed Block)을 생성할 수 있다.
- [0053] 필터부(150)는 디블록킹 필터, 오프셋 보정부, ALF(Adaptive Loop Filter)중 적어도 하나를 포함할 수 있다.
- [0054] 디블록킹 필터는 복원된 픽처에서 블록간의 경계로 인해 생긴 블록 왜곡을 제거할 수 있다. 디블록킹을 수행할지 여부를 판단하기 위해 블록에 포함된 몇 개의 열 또는 행에 포함된 픽셀을 기초로 현재 블록에 디블록킹 필터 적용할지 여부를 판단할 수 있다. 블록에 디블록킹 필터를 적용하는 경우 필요한 디블록킹 필터링 강도에 따라 강한 필터(Strong Filter) 또는 약한 필터(Weak Filter)를 적용할 수 있다. 또한 디블록킹 필터를 적용함에 있어 수직 필터링 및 수평 필터링 수행시 수평 방향 필터링 및 수직 방향 필터링이 병행 처리되도록 할 수 있다.
- [0055] 디블록킹 필터를 수행함에 있어, 블록 경계에 인접한 두 블록 P, Q의 형태, 크기 및/또는 특성에 따라 적응적인 필터링을 수행할 수 있다. 예컨대, 두 블록 P, Q의 크기가 상이할 경우, 크기가 더 큰 블록에 대해서는 크기가 작은 블록보다 더 많은 픽셀을 필터링할 수 있다. 또한, 두 블록 P, Q의 적어도 하나가 비정방형 블록인지의 여부에 기초하여, 적응적인 필터링을 수행할 수 있다. 예컨대, 블록 P가 8x8 블록이고, 블록 Q가 8x16블록인 경우, P, Q가 서로 인접한 블록 경계의 필터링에 있어서, 블록 Q에 대해서는 블록 P보다 더 많은 픽셀을 필터링할 수 있다.

- [0056] 블록 경계에 인접한 두 블록 P, Q의 크기가 상이하거나, 또는 적어도 어느 하나가 비정방형 블록인 경우, 두 블록 P, Q의 필터링되는 픽셀의 수를 동일하게 하되, P, Q에 대해 각각 서로 다른 강도의 필터링을 수행할 수 있다. 또는 필터링되는 픽셀의 수와 필터링 강도를 두 블록 P, Q에 대해 모두 달리 적용할 수도 있다.
- [0057] 오프셋 보정부는 디블록킹을 수행한 영상에 대해 픽셀 단위로 원본 영상과의 오프셋을 보정할 수 있다. 특정 픽처에 대한 오프셋 보정을 수행하기 위해 영상에 포함된 픽셀을 일정한 수의 영역으로 구분한 후 오프셋을 수행할 영역을 결정하고 해당 영역에 오프셋을 적용하는 방법 또는 각 픽셀의 에지 정보를 고려하여 오프셋을 적용하는 방법을 사용할 수 있다.
- [0058] ALF(Adaptive Loop Filtering)는 필터링한 복원 영상과 원래의 영상을 비교한 값을 기초로 수행될 수 있다. 영상에 포함된 픽셀을 소정의 그룹으로 나눈 후 해당 그룹에 적용될 하나의 필터를 결정하여 그룹마다 차별적으로 필터링을 수행할 수 있다. ALF를 적용할지 여부에 관련된 정보는 휘도 신호는 부호화 단위(Coding Unit, CU) 별로 전송될 수 있고, 각각의 블록에 따라 적용될 ALF 필터의 모양 및 필터 계수는 달라질 수 있다. 또한, 적용 대상 블록의 특성에 상관없이 동일한 형태(고정된 형태)의 ALF 필터가 적용될 수도 있다.
- [0059] 메모리(155)는 필터부(150)를 통해 산출된 복원 블록 또는 픽처를 저장할 수 있고, 저장된 복원 블록 또는 픽처는 인터 예측을 수행 시 예측부(120, 125)에 제공될 수 있다.
- [0060] 도 2는 본 발명의 일실시예에 따른 영상 복호화 장치를 나타낸 블록도이다.
- [0061] 도 2를 참조하면, 영상 복호화기(200)는 엔트로피 복호화부(210), 재정렬부(215), 역양자화부(220), 역변환부(225), 예측부(230, 235), 필터부(240), 메모리(245)가 포함될 수 있다.
- [0062] 영상 부호화기에서 영상 비트스트림이 입력된 경우, 입력된 비트스트림은 영상 부호화기와 반대의 절차로 복호화될 수 있다.
- [0063] 엔트로피 복호화부(210)는 영상 부호화기의 엔트로피 부호화부에서 엔트로피 부호화를 수행한 것과 반대의 절차로 엔트로피 복호화를 수행할 수 있다. 예를 들어, 영상 부호화기에서 수행된 방법에 대응하여 지수 골롬(Exponential Golomb), CAVLC(Context-Adaptive Variable Length Coding), CABAC(Context-Adaptive Binary Arithmetic Coding)과 같은 다양한 방법이 적용될 수 있다.
- [0064] 엔트로피 복호화부(210)에서는 부호화기에서 수행된 인트라 예측 및 인터 예측에 관련된 정보를 복호화할 수 있다.
- [0065] 재정렬부(215)는 엔트로피 복호화부(210)에서 엔트로피 복호화된 비트스트림을 부호화부에서 재정렬한 방법을 기초로 재정렬을 수행할 수 있다. 1차원 벡터 형태로 표현된 계수들을 다시 2차원의 블록 형태의 계수로 복원하여 재정렬할 수 있다. 재정렬부(215)에서는 부호화부에서 수행된 계수 스캐닝에 관련된 정보를 제공받고 해당 부호화부에서 수행된 스캐닝 순서에 기초하여 역으로 스캐닝하는 방법을 통해 재정렬을 수행할 수 있다.
- [0066] 역양자화부(220)는 부호화기에서 제공된 양자화 파라미터와 재정렬된 블록의 계수값을 기초로 역양자화를 수행할 수 있다.
- [0067] 역변환부(225)는 영상 부호화기에서 수행한 양자화 결과에 대해 변환부에서 수행한 변환 즉, DCT, DST, 및 KLT에 대해 역변환 즉, 역 DCT, 역 DST 및 역 KLT를 수행할 수 있다. 역변환은 영상 부호화기에서 결정된 전송 단위를 기초로 수행될 수 있다. 영상 복호화기의 역변환부(225)에서는 예측 방법, 현재 블록의 크기 및 예측 방향 등 복수의 정보에 따라 변환 기법(예를 들어, DCT, DST, KLT)이 선택적으로 수행될 수 있다.
- [0068] 예측부(230, 235)는 엔트로피 복호화부(210)에서 제공된 예측 블록 생성 관련 정보와 메모리(245)에서 제공된 이전에 복호화된 블록 또는 픽처 정보를 기초로 예측 블록을 생성할 수 있다.
- [0069] 전술한 바와 같이 영상 부호화기에서의 동작과 동일하게 인트라 예측을 수행시 예측 단위의 크기와 변환 단위의 크기가 동일할 경우, 예측 단위의 좌측에 존재하는 픽셀, 좌측 상단에 존재하는 픽셀, 상단에 존재하는 픽셀을 기초로 예측 단위에 대한 인트라 예측을 수행하지만, 인트라 예측을 수행시 예측 단위의 크기와 변환 단위의 크기가 상이할 경우, 변환 단위를 기초로 한 참조 픽셀을 이용하여 인트라 예측을 수행할 수 있다. 또한, 최소 부호화 단위에 대해서만 $N \times N$ 분할을 사용하는 인트라 예측을 사용할 수도 있다.
- [0070] 예측부(230, 235)는 예측 단위 판별부, 인터 예측부 및 인트라 예측부를 포함할 수 있다. 예측 단위 판별부는 엔트로피 복호화부(210)에서 입력되는 예측 단위 정보, 인트라 예측 방법의 예측 모드 정보, 인터 예측 방법의 모션 예측 관련 정보 등 다양한 정보를 입력 받고 현재 부호화 단위에서 예측 단위를 구분하고, 예측 단위가 인

터 예측을 수행하는지 아니면 인트라 예측을 수행하는지 여부를 판별할 수 있다. 인터 예측부(230)는 영상 부호화기에서 제공된 현재 예측 단위의 인터 예측에 필요한 정보를 이용해 현재 예측 단위가 포함된 현재 픽처의 이전 픽처 또는 이후 픽처 중 적어도 하나의 픽처에 포함된 정보를 기초로 현재 예측 단위에 대한 인터 예측을 수행할 수 있다. 또는, 현재 예측 단위가 포함된 현재 픽처 내에서 기-복원된 일부 영역의 정보를 기초로 인터 예측을 수행할 수도 있다.

[0071] 인터 예측을 수행하기 위해 부호화 단위를 기준으로 해당 부호화 단위에 포함된 예측 단위의 모션 예측 방법이 스킵 모드(Skip Mode), 머지 모드(Merge 모드), AMVP 모드(AMVP Mode), 인트라 블록 카피 모드 중 어떠한 방법인지 여부를 판단할 수 있다.

[0072] 인트라 예측부(235)는 현재 픽처 내의 화소 정보를 기초로 예측 블록을 생성할 수 있다. 예측 단위가 인트라 예측을 수행한 예측 단위인 경우, 영상 부호화기에서 제공된 예측 단위의 인트라 예측 모드 정보를 기초로 인트라 예측을 수행할 수 있다. 인트라 예측부(235)에는 AIS(Adaptive Intra Smoothing) 필터, 참조 화소 보간부, DC 필터를 포함할 수 있다. AIS 필터는 현재 블록의 참조 화소에 필터링을 수행하는 부분으로써 현재 예측 단위의 예측 모드에 따라 필터의 적용 여부를 결정하여 적용할 수 있다. 영상 부호화기에서 제공된 예측 단위의 예측 모드 및 AIS 필터 정보를 이용하여 현재 블록의 참조 화소에 AIS 필터링을 수행할 수 있다. 현재 블록의 예측 모드가 AIS 필터링을 수행하지 않는 모드일 경우, AIS 필터는 적용되지 않을 수 있다.

[0073] 참조 화소 보간부는 예측 단위의 예측 모드가 참조 화소를 보간한 화소값을 기초로 인트라 예측을 수행하는 예측 단위일 경우, 참조 화소를 보간하여 정수값 이하의 화소 단위의 참조 화소를 생성할 수 있다. 현재 예측 단위의 예측 모드가 참조 화소를 보간하지 않고 예측 블록을 생성하는 예측 모드일 경우 참조 화소는 보간되지 않을 수 있다. DC 필터는 현재 블록의 예측 모드가 DC 모드일 경우 필터링을 통해서 예측 블록을 생성할 수 있다.

[0074] 복원된 블록 또는 픽처는 필터부(240)로 제공될 수 있다. 필터부(240)는 디블록킹 필터, 오프셋 보정부, ALF를 포함할 수 있다.

[0075] 영상 부호화기로부터 해당 블록 또는 픽처에 디블록킹 필터를 적용하였는지 여부에 대한 정보 및 디블록킹 필터를 적용하였을 경우, 강한 필터를 적용하였는지 또는 약한 필터를 적용하였는지에 대한 정보를 제공받을 수 있다. 영상 복호화기의 디블록킹 필터에서는 영상 부호화기에서 제공된 디블록킹 필터 관련 정보를 제공받고 영상 복호화기에서 해당 블록에 대한 디블록킹 필터링을 수행할 수 있다.

[0076] 오프셋 보정부는 부호화시 영상에 적용된 오프셋 보정의 종류 및 오프셋 값 정보 등을 기초로 복원된 영상에 오프셋 보정을 수행할 수 있다.

[0077] ALF는 부호화기로부터 제공된 ALF 적용 여부 정보, ALF 계수 정보 등을 기초로 부호화 단위에 적용될 수 있다. 이러한 ALF 정보는 특정한 파라미터 셋에 포함되어 제공될 수 있다.

[0078] 메모리(245)는 복원된 픽처 또는 블록을 저장하여 참조 픽처 또는 참조 블록으로 사용할 수 있도록 할 수 있고 또한 복원된 픽처를 출력부로 제공할 수 있다.

[0079] 전술한 바와 같이 이하, 본 발명의 실시예에서는 설명의 편의상 코딩 유닛(Coding Unit)을 부호화 단위라는 용어로 사용하지만, 부호화뿐만 아니라 복호화를 수행하는 단위가 될 수도 있다.

[0080] 도 3(a)는 본 발명의 일실시예에 따라, 입력 영상의 기본 블록을 쿼드 트리 구조를 이용하여 복수의 서브 블록으로 분할한 구조를 예시적으로 나타내는 도면이다.

[0081] 도 3(b)는 입력 영상의 기본 블록을 쿼드 트리 구조 및/또는 이진 트리 구조를 사용하여 복수의 서브 블록으로 분할한 구조를 예시적으로 나타내는 도면이다.

[0082] 부호화 하고자 하는 입력 영상은 효율적으로 부호화하기 위하여 기본 블록 단위로 분할되어 부호화될 수 있다. 본 발명의 기본 블록은 코딩 트리 유닛(Coding Tree Unit, CTU)으로 정의될 수 있다. 기본 블록들은 모두 $M \times N$ 의 일정한 직사각형 혹은 정사각형 크기를 가질 수 있다. M 과 N 은 2^n (n 은 1보다 큰 정수)의 값을 가지는 정수일 수 있으며 M 은 블록의 가로 길이, N 은 블록의 세로 길이를 의미한다. 이러한 기본 블록들은 영상 압축을 효율적으로 수행하기 위해 추가적으로 분할될 수 있다.

[0083] 영상 압축을 효율적으로 수행하기 위해서는 영상을 균질도에 따라 균질 영역으로 분할하는 것이 바람직하다. 균질 영역이란, 그 영역 내에 포함된 샘플의 휘도 및/또는 색차 값 사이의 변화가 없거나, 그 변화가 소정의 임계치 이하인 경우를 의미한다. 즉, 균질 영역은 균질한 샘플 값을 갖는 샘플들로 구성되며, 균질도는 소정의 판단

기준에 따라 결정될 수 있다. 영상의 균질성을 고려하여 기본 블록을 균질 영역인 복수의 서브 블록으로 분할하면, 서브 블록의 예측 잔차 신호(residual 신호) 에너지를 보다 효율적으로 집중시킬 수 있으며, 그에 따라 변환(transform)과 양자화(quantization)시 압축 효율의 향상을 얻을 수 있다.

- [0084] 입력 영상 내의 기본 블록을 균질도에 따라 복수의 서브 블록으로 분할하기 위해서는 이진 트리 구조, 쿼드 트리 구조, 팔진 트리 구조(Octree Structure) 및/또는 일반적인 N 진 트리 구조(N-ary Tree Structure) 등을 이용할 수 있다.
- [0085] 도 3(a)는 쿼드 트리 구조만을 이용하여 기본 블록을 복수의 서브 블록으로 분할한 예이다. 쿼드 트리 구조는 분할의 대상이 되는 블록을 같은 블록 크기를 갖는 4개의 블록으로 분할하는 하는 것으로서, 분할된 4개의 블록에 대해서는 다시 쿼드 트리 구조에 따른 블록의 분할이 가능하다.
- [0086] 도 3(b)는 쿼드 트리 구조 및/또는 이진 트리 구조를 이용하여 기본 블록을 복수의 서브 블록으로 분할한 예이다. 분할된 각각의 서브 블록 내에 표시된 알파벳은 균질도를 나타내기 위한 인덱스이다. 예컨대, 알파벳 a 가 표시된 서브 블록들은 모두 동일한 균질도를 갖는 영역을 나타낸다. 또한, 도 3(b)에서, 블록의 내부에 표시된 도면 부호는 해당 블록을 지시하며, 블록의 경계에 표시된 도면 부호는 해당 경계에 의해 분할되는 블록을 지시한다.
- [0087] 도 3(a), 도 3(b)에 나타난 바와 같이, 기본 블록은 복수의 서브 블록으로 분할될 수 있다. 더 이상 분할되지 않는 서브 블록은 부호화 단위(코딩 유닛)로 결정될 수 있다. 도 3(a), 도 3(b)에 있어서, 부호화 단위는 예측, 변환 및/또는 양자화 등의 부호화의 단위가 될 수 있다. 또는 예측, 변환 및/또는 양자화를 수행하기 위해 부호화 단위는 다시 분할될 수 있다. 예컨대, 부호화 단위의 예측을 위해, 쿼드 트리 구조에 따른 분할, 이진 트리 구조에 따른 분할, 또는 비대칭형 분할이 수행될 수 있고, 정사각형 또는 직사각형의 형태가 아닌 다양한 형태의 분할도 고려될 수 있다. 또한, 부호화 단위의 변환 및/또는 양자화를 위해, 상기 예측을 위해 사용될 수 있는 분할 방법이 마찬가지로 적용될 수 있다.
- [0088] 입력 영상의 기본 블록을 분할할 때, 쿼드 트리 구조 및/또는 이진 트리 구조를 사용할지를 나타내는 정보는 비트스트림을 통해 시그널링될 수 있다. 입력 영상의 기본 블록의 분할 구조에 관한 정보는 예를 들어, 시퀀스 단위, 픽처 단위, 슬라이스 단위, 타일 단위 및/또는 기본 블록 단위로 시그널링될 수 있다. 예컨대, 상기 정보가 픽처 단위로 시그널링되는 경우에는, 해당 픽처에 포함된 모든 기본 블록들 또는 일부 기본 블록들에 대해 쿼드 트리 구조와 이진 트리 구조를 모두 이용할지 또는 쿼드 트리 구조만을 이용할지를 나타낼 수 있다. 쿼드 트리 구조만을 이용하는 것으로 결정된 경우, 기본 블록의 블록 분할 정보는 쿼드 트리 구조에 따른 분할 정보를 포함하고, 이진 트리 구조에 따른 분할 정보는 포함하지 않을 수 있다.
- [0089] 전술한 바와 같이, 쿼드 트리 구조는 분할의 대상이 되는 블록을 같은 블록 크기를 갖는 4개의 블록으로 분할하는 것이다. 또한, 이진 트리 구조는 분할의 대상이 되는 블록을 같은 크기를 갖는 2개의 블록으로 분할하는 것이다. 이진 트리 구조를 이용하여 블록을 분할하는 경우에는 수평 방향 분할인지 또는 수직 방향 분할인지를 나타내는 분할 방향에 관한 정보를 함께 부호화/복호화할 필요가 있다. 블록의 분할 정보를 부호화/복호화하는 방법에 대해서는 도 4를 참조하여 상세히 설명한다.
- [0090] 도 3(b)에 도시된 바와 같이, 예컨대, 기본 블록(300)은 쿼드 트리 구조를 이용하여 4개의 서브 블록으로 분할될 수 있다. 트리 구조를 이용하여 블록을 분할하는 경우, 트리 구조 상에서의 깊이에 기반하여, 각 블록의 깊이를 결정할 수 있다. 기본 블록(300)은 트리 구조 상에서 깊이가 0인 블록에 해당한다. 기본 블록(300)을 분할하여 얻어지는 서브 블록은 깊이가 1인 블록에 해당한다. 분할에 의해 얻어진 4개의 서브 블록(깊이 = 1)은 서로 같은 크기를 가질 수 있다. 4개의 서브 블록 중 좌상측에 위치하는 서브 블록(깊이 = 1)(301)은 더 이상 분할되지 않을 수 있다. 예컨대, 좌상측에 위치하는 서브 블록(깊이 = 1)(301)이 균질 영역으로 판단되는 경우, 서브 블록(301)은 더 이상 분할되지 않을 수 있다. 더 이상 분할되지 않는 서브 블록(깊이 = 1)(301)은 부호화 단위로 결정될 수 있다.
- [0091] 예컨대, 기본 블록(깊이 = 0)(300)으로부터 쿼드 트리 구조를 이용하여 분할된 4개의 서브 블록 중 우상측에 위치하는 서브 블록(깊이 = 1)(302)은 재귀적으로 또는 계층적으로 쿼드 트리 구조를 이용하여 4개의 서브 블록(깊이 = 2)으로 재차 분할될 수 있다. 재차 분할된 서브 블록(깊이 = 2)의 각각은 부호화 단위로 결정될 수 있다. 또는, 기본 블록(깊이 = 0)(300)의 우상측 서브 블록(깊이 = 1)(302)의 좌상측 서브 블록(깊이 = 2)(302-1)과 같이 이진 트리 구조를 이용하여 다시 분할될 수 있다. 분할된 블록은 더 이상 분할될 수 없거나, 또는 분할될 필요가 없는 균질 영역으로 판단되는 경우, 부호화 단위로 결정될 수 있다.

- [0092] 도 3(b)에 도시된 바와 같이, 쿼드 트리 구조 및/또는 이진 트리 구조를 이용하여 기본 블록을 분할하면, 쿼드 트리 구조만을 이용하여 분할한 도 3(a)의 경우와 비교했을 때, 보다 적응적으로, 다양한 크기를 갖는 균질 영역으로 블록을 분할할 수 있다.
- [0093] 전술한 바와 같이, 부호화 단위는 예측, 변환 및/또는 양자화를 위해 다시 분할하는 것이 가능하다. 그러나, 도 3(b)에 도시된 방법으로 분할하면, 매우 높은 정확도를 갖는 균질 영역으로 부호화 단위를 결정할 수 있다. 따라서, 예측, 변환 및/또는 양자화를 위하여, 부호화 단위를 다시 분할하지 않을 수 있다. 즉, 부호화 단위 그 자체가 예측의 단위가 되는 예측 유닛(prediction unit) 및/또는 변환의 단위가 되는 변환 유닛(transform unit)이 될 수 있다. 부호화 단위를 예측 유닛 및/또는 변환 유닛으로 그대로 사용할 수 있으므로, 부호화 단위를 예측 유닛 및/또는 변환 유닛으로 다시 분할함에 따르는 비용을 절약할 수 있다. 특히, 부호화 단위를 예측 유닛 및/또는 변환 유닛으로 분할하는 형태에 관한 분할 정보를 선택스 요소(syntax element)로서 부호화할 필요가 없으므로, 압축 효율의 향상을 기대할 수 있다. 또한, 전술한 바와 같이, 도 3(b)를 참조하여 설명한 블록 분할 방법에 따르면, 부호화 단위로 결정된 각각의 서브 블록은 매우 높은 정확도를 갖는 균질 영역이므로, 잔차 신호의 에너지의 집중이 매우 효율적이 되어, 변환 및/또는 양자화시 압축 효율이 향상될 수 있다.
- [0094] 본 발명에 따르면, 기본 블록은 쿼드 트리 구조 및/또는 이진 트리 구조를 사용하여 복수의 부호화 단위로 분할될 수 있다. 쿼드 트리 구조와 이진 트리 구조는 순서에 상관없이 필요에 따라 적절히 선택되어 이용될 수 있다. 또는 쿼드 트리 구조를 주 분할 구조로 이용하고, 이진 트리 구조를 부 분할 구조로 이용할 수 있다. 또는 이진 트리 구조를 주 분할 구조로 이용하고, 쿼드 트리 구조를 부 분할 구조로 이용할 수 있다. 어느 하나를 주 분할 구조로 하고, 나머지 하나를 부 분할 구조로 하는 경우, 먼저 주 분할 구조에 따른 분할이 수행될 수 있다. 주 분할 구조의 종단 노드(lead node)에 도달하면, 종단 노드는 부 분할 구조의 루트 노드(root node)가 되어 부 분할 구조에 따른 분할이 수행될 수 있다.
- [0095] 도 4(a)는 쿼드 트리 구조를 주 분할 구조로, 이진 트리 구조를 부 분할 구조로 이용하여 입력 영상의 기본 블록을 복수의 서브 블록으로 분할한 구조를 예시적으로 나타내는 도면이다. 도 4(a)에서, 블록의 내부에 표시된 도면 부호는 해당 블록을 지시하며, 블록의 경계에 표시된 도면 부호는 해당 경계에 의해 분할되는 블록을 지시한다.
- [0096] 도 4(b)는 본 발명의 일 실시예에 따라, 도 4(a)에 도시된 블록 분할 구조에 대한 블록 분할 정보를 트리 구조를 이용하여 예시적으로 나타내는 도면이다.
- [0097] 도 4(c)는 본 발명의 다른 실시예에 따라, 도 4(a)에 도시된 블록 분할 구조에 대한 블록 분할 정보를 트리 구조를 이용하여 예시적으로 나타내는 도면이다.
- [0098] 도 4(a), (b) 및 (c)에서, 쿼드 트리 구조에 따른 블록의 분할은 실선으로 표시하고, 이진 트리 구조에 따른 블록의 분할은 점선으로 표시한다.
- [0099] 도 4(a)에 도시된 바와 같이, 기본 블록(깊이 = 0)(400)에 대해 먼저 쿼드 트리 구조를 이용하여 크기를 갖는 4개의 서브 블록들(깊이 = 1)로 분할한다. 4개의 서브 블록들(깊이 = 1)의 각각은 쿼드 트리 구조를 이용하여, 재차 분할될 수 있다. 더 이상 쿼드 트리 구조에 따른 분할이 수행되지 않는 서브 블록은 쿼드 트리 구조의 종단 노드에 해당한다. 쿼드 트리 구조의 종단 노드에 대해서는 이진 트리 구조에 따른 분할이 수행될 수도 있고, 수행되지 않을 수도 있다. 쿼드 트리 구조의 종단 노드에 대해 이진 트리 구조에 따른 분할이 수행되지 않으면, 해당 종단 노드는 부호화 단위로 결정될 수 있다.
- [0100] 쿼드 트리 구조의 종단 노드에 대해서는 이진 트리 구조에 따른 분할이 더 수행될 수 있다. 즉, 쿼드 트리 구조의 종단 노드가 이진 트리 구조의 루트 노드가 될 수 있다. 예컨대, 도 4(a)에서, 기본 블록(깊이 = 0)(400)을 분할하여 얻어진 4개의 서브 블록 중 우하측에 위치하는 서브 블록(깊이 = 1)(401)에 대해서는 더 이상 쿼드 트리 구조에 따른 분할이 수행되지 않는다. 상기 우하측 서브 블록(깊이 = 1)(401)은 이진 트리 구조의 루트 노드로서 이진 트리 구조에 따른 분할이 수행될 수 있다. 도 4(a)에 도시된 바와 같이, 상기 우하측 서브 블록(깊이 = 1)(401)에 대해 이진 트리 구조에 따른 수직 방향 분할이 수행되어 두 개의 서브 블록(깊이 = 2)으로 분할될 수 있다. 또한, 두 개의 서브 블록 중 좌측에 위치한 서브 블록(깊이 = 2)(401-1)에 대해 다시 이진 트리 구조에 따른 수직 방향 분할이 수행되어 두 개의 서브 블록(깊이 = 3)(401-1a, 401-1b)으로 분할될 수 있다. 두 개의 서브 블록(깊이 = 3)(401-1a, 401-1b)의 각각은 이진 트리 구조의 종단 노드에 해당할 수 있다. 더 이상 분할되지 않는 두 개의 서브 블록(깊이 = 3)(401-1a, 401-1b)의 각각은 부호화 단위로 결정될 수 있다.
- [0101] 전술한 바와 같이, 본 발명에 따라, 쿼드 트리 구조를 주 분할 구조로, 이진 트리 구조를 부 분할 구조로 이용

하여 기본 블록(400)을 분할할 수 있다. 이 때, 기본 블록(400)을 퀴드 트리 구조의 루트 노드로 결정할 수 있다. 퀴드 트리 구조의 루트 노드는 퀴드 트리 구조의 중단 노드에 도달하기 전까지 퀴드 트리 구조를 재귀적으로 또는 계층적으로 이용하여 분할될 수 있다. 퀴드 트리의 중단 노드는 이진 트리의 루트 노드가 될 수 있다. 이진 트리 구조의 루트 노드는 이진 트리 구조의 중단 노드에 도달하기 전까지 이진 트리 구조를 재귀적으로 또는 계층적으로 이용하여 분할할 수 있다. 퀴드 트리 구조에 따른 분할 및/또는 이진 트리 구조에 따른 분할을 포함하여 현재 블록이 더 이상 분할되지 않는 경우, 현재 블록은 부호화 단위로 결정될 수 있다.

[0102] 도 4(b)는 본 발명의 일 실시예에 따라, 도 4(a)에 도시된 블록 분할 구조를 부호화/복호화하기 위한 블록 분할 정보를 나타내는 도면이다.

[0103] 도 4(b), 도 4(c)의 트리 구조에서, 괄호안의 도면 부호는 도 4(a)에 도시된 블록의 도면 부호이며, 0 및/또는 1로 나타낸 정보는 해당 블록에 대한 블록 분할 정보의 일례를 나타낸다.

[0104] 이하에서, 퀴드 트리 구조에 따른 블록 분할 여부를 나타내는 정보를 “퀴드 분할 정보”라 한다. 퀴드 분할 정보는 제1 비트 길이로 부호화될 수 있다. 제1 비트 길이는 1 비트일 수 있다. 현재 블록이 퀴드 트리 구조에 따라 분할되는 경우, 현재 블록의 퀴드 분할 정보를 “1”로 부호화할 수 있다. 현재 블록이 퀴드 트리 구조에 따라 분할되지 않는 경우에는, 퀴드 분할 정보를 “0”으로 부호화할 수 있다. 즉, 현재 블록이 퀴드 트리 구조에 따라 분할되는지의 여부는 1 비트의 퀴드 분할 정보로 부호화할 수 있다.

[0105] 이하에서, 이진 트리 구조에 따른 블록 분할 정보를 “이진 분할 정보”라 한다. 이진 분할 정보는 이진 트리 구조에 따른 블록 분할 여부를 나타내는 정보 및 이진 트리 구조에 따른 분할 방향에 관한 정보 중 적어도 하나를 포함할 수 있다. 이진 분할 정보는 제2 비트 길이로 부호화될 수 있다. 제2 비트 길이는 1비트 또는 2 비트일 수 있다. 현재 블록이 이진 트리 구조에 따른 분할 대상인 경우, 후술하는 바와 같이 2 비트의 이진 분할 정보가 사용될 수 있다. 현재 블록이 이진 트리 구조에 따라 분할되는 경우, 상기 2 비트의 이진 분할 정보의 첫 번째 비트는 “1”로 부호화될 수 있다. 현재 블록이 이진 트리 구조를 이용하여 분할되지 않는 경우, 상기 2 비트의 이진 분할 정보의 첫 번째 비트는 “0”으로 부호화될 수 있다. 퀴드 트리 구조와는 달리, 이진 트리 구조는 블록의 분할 방향에 관한 정보를 추가적으로 부호화할 필요가 있다. 즉, 이진 트리 구조에 따른 분할이 수평 방향 분할인 경우에는, 상기 2 비트의 이진 분할 정보의 두 번째 비트를 “0”으로 부호화할 수 있다. 이진 트리 구조에 따른 분할이 수직 방향 분할인 경우에는, 상기 2 비트의 이진 분할 정보의 두 번째 비트를 “1”로 부호화할 수 있다. 현재 블록에 대해 이진 트리 구조에 따른 분할이 수행되지 않는 경우, 상기 2 비트의 이진 분할 정보의 두 번째 비트는 사용되지 않을 수 있다.

[0106] 상기 제1 비트 길이와 제2 비트 길이는 서로 동일할 수 있으며, 상기 실시예에 한정되지 않는다. 또한, 상기 실시예에서, 퀴드 분할 정보와 이진 분할 정보 각각의 비트 값의 의미를 반대로 정의할 수 있다. 예컨대, 퀴드 분할 정보 및/또는 이진 분할 정보의 첫 번째 비트에 “0”을 사용하여 해당 구조에 따른 분할이 수행됨을 나타내고, “1”을 사용하여 분할이 수행되지 않음을 나타낼 수 있다. 또는 이진 분할 정보의 두 번째 비트가 “0”인 경우가 수직 방향 분할, “1”인 경우가 수평 방향 분할로 각각 정의할 수 있다.

[0107] 현재 블록의 블록 분할 정보는 상기 제1 비트 길이의 퀴드 분할 정보와 상기 제2 비트 길이의 이진 분할 정보를 함께 사용하여 아래의 표 1과 같이 부호화될 수 있다.

표 1

퀴드 분할 정보	이진 분할 정보	블록 분할 정보	의미
0	0	00	현재 블록에 대해 퀴드 트리 구조에 따른 분할과 이진 트리 구조에 따른 분할을 수행하지 않음
0	10	010	현재 블록에 대해 이진 트리 구조에 따라 수평 방향으로 분할을 수행
0	11	011	현재 블록에 대해 이진 트리 구조에 따라 수직 방향으로 분할을 수행
1	-	1	현재 블록에 대해 퀴드 트리 구조에 따른 분할을 수행
-	0	0	이진 트리 구조를 부 분할 구조로 이용한 분할을 통해 얻어진 현재 블록에 대해 이진 트리 구조에 따른 분할을 수행하지 않음
-	10	10	이진 트리 구조를 부 분할 구조로 이용한 분할을 통해 얻어진 현재 블록에 대해 이진 트리 구조에 따라 수평 방향으로 분할을 수행
-	11	11	이진 트리 구조를 부 분할 구조로 이용한 분할을 통해 얻어진 현재 블록에 대해 이진 트리 구조에 따라 수직 방향으로 분할을 수행

- [0109] 상기 표 1에서 쿼드 분할 정보는 “0” 또는 “1” 이 될 수 있으며, 이진 분할 정보는 “0”, “10” 또는 “11” 로 부호화될 수 있다. 블록 분할 정보는 블록의 분할 여부, 분할 타입 및 분할 방향을 나타내는 정보로서 쿼드 분할 정보와 이진 분할 정보를 결합한 정보 또는 쿼드 분할 정보와 이진 분할 정보의 조합을 의미하는 정보 일 수 있다. 상기 표 1에서, 블록 분할 정보 “00”은 현재 블록에 대해 쿼드 트리 구조에 따른 분할과 이진 트리 구조에 따른 분할을 수행하지 않음을 의미한다. 블록 분할 정보 “010”은 현재 블록에 대해 쿼드 트리 구조에 따른 분할은 수행되지 않고(첫 번째 비트 = 0), 이진 트리 구조에 따른 분할은 수행되며(두 번째 비트 = 1), 이진 트리 구조에 따른 분할의 방향은 수평 방향임(세 번째 비트 = 0)을 의미한다. 블록 분할 정보 “011”은 현재 블록에 대해 쿼드 트리 구조에 따른 분할은 수행되지 않고(첫 번째 비트 = 0), 이진 트리 구조에 따른 분할은 수행되며(두 번째 비트 = 1), 이진 트리 구조에 따른 분할의 방향은 수직 방향임(세 번째 비트 = 1)을 의미한다. 블록 분할 정보 “1”은 현재 블록에 대해 쿼드 트리 구조에 따른 분할이 수행됨을 의미한다.
- [0110] 상기 표 1에서, 쿼드 트리 구조를 주 분할 구조로 하고, 이진 트리 구조를 부 분할 구조로 이용하는 경우, 쿼드 트리 구조에 따라 분할된 블록에 대해 일단 이진 트리 구조에 따른 분할이 수행되면, 다시 쿼드 트리 구조에 따른 분할이 수행되지 않는다. 따라서, 이진 트리 구조에 따른 분할로 얻어진 블록에 대해서는 더 이상 쿼드 트리 구조에 따른 분할 수행 여부를 나타내는 쿼드 분할 정보를 부호화/복호화할 필요가 없을 수 있다. 이 경우, 이진 트리 구조에 따른 분할로 얻어진 블록의 블록 분할 정보는 이진 분할 정보만을 포함할 수 있다. 즉, 쿼드 분할 정보가 없이, 이진 분할 정보가 그대로 블록 분할 정보로 사용될 수 있으며, 블록 분할 정보 “10”은 현재 블록에 대해 이진 트리 구조에 따른 분할이 수행되며(첫 번째 비트 = 1), 이진 트리 구조에 따른 분할의 방향은 수평 방향임(두 번째 비트 = 0)을 의미한다. 블록 분할 정보 “11”은 현재 블록에 대해 이진 트리 구조에 따른 분할이 수행되며(첫 번째 비트 = 1), 이진 트리 구조에 따른 분할의 방향은 수직 방향임(두 번째 비트 = 1)을 의미한다. 블록 분할 정보 “0”은 이진 트리 구조에 따른 분할로 얻어진 현재 블록에 대해 더 이상 이진 트리 구조에 따른 분할이 수행되지 않음을 의미한다.
- [0111] 도 4(b)는 상기 표 1의 블록 분할 정보 부호화 방법을 이용하여, 도 4(a)에 도시된 블록 분할 구조에 대한 분할 정보를 트리 구조로 도시한 도면이다. 트리 구조의 깊이는 도 4(a)의 블록 분할 구조의 깊이에 대응한다. 도 4(b)의 각 노드에 나타난 정보는 그 노드에 대응하는 블록의 블록 분할 정보를 의미한다. 예컨대, 도 4(b)의 루트 노드에 표시된 “1”은 그 노드에 대응하는 블록, 즉 도 4(a)의 기본 블록(400)의 블록 분할 정보로서, 기본 블록(400)이 쿼드 트리 구조에 따라 4개의 서브 블록으로 분할됨을 나타낸다. 하나의 노드가 복수의 자식 노드를 갖는 경우, 도 4(b)에서 자식 노드를 배열한 순서는 도 4(a)에 도시된 블록 분할 구조의 래스터 스캔 순서(또는 z 스캔 순서)에 따른다. 즉, 도 4(a)의 기본 블록(깊이 = 0)(400)을 쿼드 트리 구조에 따라 4개의 서브 블록(깊이 = 1)으로 분할한 경우, 예컨대, 우하측 서브 블록(깊이 = 1)(401)은 래스터 스캔 순서상 4개의 서브 블록(깊이 = 1) 중 가장 마지막에 위치한다. 따라서, 도 4(b)의 트리 구조상에서 깊이가 1인 자식 노드 중 가장 마지막에 배열된 노드의 위치에 표시된 “011”이 도 4(a)의 우하측 서브 블록(깊이 = 1)(401)의 블록 분할 정보에 대응된다.
- [0112] 블록 분할 정보는 표 1에 나타난 바와 같이, 쿼드 분할 정보와 이진 분할 정보를 결합하거나 또는 조합한 형태의 정보로서 부호화될 수 있다. 또는 쿼드 분할 정보와 이진 분할 정보를 각각 별도의 선택스 요소로서 부호화할 수도 있다. 또한, 이진 분할 정보에 포함된 이진 트리 구조에 따른 블록 분할 여부를 나타내는 정보와 이진 트리 구조에 따른 분할 방향에 관한 정보를 각각 별도의 선택스 요소로 부호화할 수도 있다.
- [0113] 도 4(c)는 본 발명의 다른 실시예에 따라, 도 4(a)에 도시된 블록 분할 구조를 부호화하기 위한 블록 분할 정보를 나타내는 도면이다.
- [0114] 전술한 바와 같이, 쿼드 분할 정보를 위해서는 1 비트가 필요하고 이진 분할 정보를 위해서는 2 비트가 필요할 수 있다. 즉, 이진 분할 정보를 시그널링 하기 위해서는 분할 방향에 관한 정보가 추가적으로 필요하므로, 이진 분할 정보는 쿼드 분할 정보보다 더 많은 비트수가 요구될 수 있다. 도 4(c)를 통해 설명되는 블록 분할 정보 부호화 방법은, 블록 분할 정보의 비트수를 고려하여, 아래의 표 2와 같이, 현재 블록에 적용될 분할 방식이 쿼드 트리 구조에 따른 분할인지 이진 트리 구조에 따른 분할인지 부호화할 수 있다.

표 2

[0115]

블록 분할 정보	의미
0	현재 블록에 대해 쿼드 트리 구조에 따른 분할과 이진 트리 구조에 따른 분할을 수행하지 않음
1	현재 블록에 대해 쿼드 트리 구조에 따른 분할을 수행
10	현재 블록에 대해 이진 트리 구조에 따른 분할을 수평 방향으로 수행
11	현재 블록에 대해 이진 트리 구조에 따른 분할을 수직 방향으로 수행

[0116]

상기 표 2에서 블록 분할 정보 “0”은 현재 블록에 대해 쿼드 트리 구조에 따른 분할과 이진 트리 구조에 따른 분할을 수행하지 않음을 의미한다. 즉, 현재 블록이 분할되지 않는 블록임을 의미한다. 블록 분할 정보 “1”은 현재 블록에 대해 쿼드 트리 구조에 따른 분할이 수행됨을 의미한다. 블록 분할 정보 “10”은 현재 블록에 대해 이진 트리 구조에 따른 분할이 수행되며, 이진 트리 구조에 따른 분할의 방향은 수평 방향임을 의미한다. 블록 분할 정보 “11”은 현재 블록에 대해 이진 트리 구조에 따른 분할이 수행되며, 이진 트리 구조에 따른 분할의 방향은 수직 방향임을 의미한다.

[0117]

표 2의 블록 분할 정보 부호화 방법은 블록 분할 정보의 비트수에 기초하여, 블록의 분할 타입(쿼드 트리 구조 또는 이진 트리 구조)을 시그널링한다. 1 비트인 경우는 블록의 분할 여부 및/또는 쿼드 트리 구조에 따른 분할이 수행됨을 나타낸다. 2 비트인 경우는 이진 트리 구조에 따른 분할이 수행됨을 나타낸다. 따라서, 표 1의 블록 분할 정보 부호화 방법보다 더 적은 비트수로 블록 분할 정보를 부호화하는 것이 가능하다.

[0118]

도 4(c)는 상기 표 2의 블록 분할 정보 부호화 방법을 이용하여, 도 4(a)에 도시된 블록 분할 구조에 대한 분할 정보를 트리 구조로 도시한 도면이다. 도 4(b)와 도 4(c)는 각각 표1과 표2의 서로 다른 블록 분할 정보 부호화 방법을 이용하고 있는 것 외에는 동일한 트리 구조를 갖는다. 따라서, 도 4(b)에 대한 설명 중, 일부, 특히, 자식 노드의 배열 순서에 대한 설명은 도 4(c)에 대해서도 동일하게 적용될 수 있다.

[0119]

블록 분할 정보의 부호화 방법은 표 1 및 표 2에 나타난 방법으로 제한되지 않으며, 예컨대, 표 1 및/또는 표 2의 방법을 혼합하여 사용하거나, 일부를 생략하여 사용하는 것이 가능하다. 블록 분할 정보는 현재 블록이 분할되는지 여부, 현재 블록에 적용되는 분할이 쿼드 트리 분할인지 이진 트리 분할인지 여부 및/또는 이진 트리 분할이 적용될 경우, 분할 방향이 가로인지 세로인지 여부를 나타낼 수 있는 모든 형태의 정보를 의미한다.

[0120]

도 5는 입력 영상의 기본 블록을, 쿼드 트리 구조 및/또는 이진 트리 구조에 따라 복수의 서브 블록으로 분할한 경우, 분할된 서브 블록이 취할 수 있는 다양한 블록의 크기 및 형태를 예시적으로 나타내는 도면이다.

[0121]

부호화 단위의 크기가 너무 작은 경우에는 오히려 부호화(예측, 변환 및/또는 양자화)의 효율이 저하되는 경우가 있다. 또한 블록 분할 정보의 부호화에 따라 전송될 데이터의 양이 증가할 수 있다. 따라서, 더 작은 블록으로 분할될 수 블록의 크기를 제한할 필요가 있다. 예컨대, 분할된 블록의 길이(세로 및/또는 가로)가 소정의 값 이하인 경우에는 더 이상 분할되지 않는 것으로 할 수 있다. 상기 소정의 값은 4, 8, 16과 같이 임의의 크기로 설정할 수 있다. 상기 소정의 값은 비트스트림을 통해 시그널링할 수 있다. 상기 소정의 값은 시퀀스 단위, 픽처 단위, 슬라이스 단위 또는 타일 단위로 적응적으로 시그널링할 수 있다. 또는, 상기 소정의 값은 부호화기와 복호화기가 미리 약속한 값으로 설정할 수도 있다.

[0122]

또는, 블록의 세로 또는 가로의 한 쪽만 상기 소정의 값 이하인 경우, 이진 트리 구조에 따른 분할을 한쪽 방향으로만 수행할 수 있다. 예컨대, 블록의 가로 길이는 소정의 값 이하로서 더 이상 분할될 수 없는 크기에 해당되나, 블록의 세로 길이는 소정의 값을 넘는 경우, 수평 방향의 이진 트리 분할만이 가능하도록 결정할 수 있다. 보다 구체적으로, 블록에 대한 분할이 수행되는 최소 길이가 4인 경우, 도 5에 도시된 4x32, 4x16, 4x8 블록에 대해서는 이진 트리 구조에 따른 수평 방향 분할이 가능한 것으로 취급할 수 있다. 이 때, 4x32, 4x16, 4x8 블록의 사이즈로부터 이진 트리 구조에 따른 수평 방향 분할만이 가능하다는 것을 알 수 있으므로, 블록 분할 정보는 “10” 또는 “1”로 부호화될 수 있고, 더 이상 분할되지 않는 경우 블록 분할 정보는 “0”으로 부호화될 수 있다. 마찬가지로, 32x4, 16x4, 8x4 블록에 대해서는 이진 트리 구조에 따른 수직 방향 분할만 가능한 것으로 취급할 수 있다. 이 때, 32x4, 16x4, 8x4 블록의 사이즈로부터 이진 트리 구조에 따른 수직 방향 분할만이 가능하다는 것을 알 수 있으므로, 블록 분할 정보는 “11” 또는 “1”로 부호화될 수 있고, 더 이상 분할되지 않는 경우 블록 분할 정보는 “0”으로 부호화될 수 있다.

[0123]

또는, 블록 분할이 가능한 최대 깊이를 제한하는 것도 가능하다. 예컨대, 블록의 분할이 재귀적 또는 계층적으

로 수행되어 소정 깊이에 도달한 경우에는 더 이상 분할되지 않는 것으로 설정할 수 있다. 상기 소정 깊이의 설정 및 부호화 방법에 대해서는 상기 블록 분할이 가능한 블록의 최소 크기의 설정 및 부호화 방법이 적용될 수 있다.

[0124] 전술한 바와 같이, 쿼드 트리 구조에 따른 분할 및/또는 이진 트리 구조에 따른 분할을 이용하여 기본 블록에 대한 분할을 수행하고, 더 이상 분할되지 않는 복수의 서브 블록이 결정될 수 있다. 더 이상 분할되지 않는 서브 블록은 부호화 단위로 결정되어 예측, 변환 및/또는 양자화의 단위가 될 수 있다. 부호화 단계에서는 각각의 부호화 단위에 대하여 인터 예측 또는 인트라 예측을 수행하여 예측 신호를 구할 수 있다. 구해진 예측 신호와 부호화 단위의 원본 신호의 차이로부터 잔차 신호를 계산할 수 있다. 계산된 잔차 신호에 대해서는 에너지 집중을 위해 변환이 수행될 수 있다.

[0125] 본 발명에 따른 부호화 단위는 4x4, 4x8, 8x4, 8x8, 16x4, 4x16 등 다양한 크기의 직사각형 또는 정사각형 형태이므로, 본 발명에 따른 부호화 단위의 잔차 신호를 변환하기 위해서는 정방형 변환뿐만 아니라 비정방형 변환에 대한 정의가 필요하다. 변환에 사용되는 수식은 다음과 같다.

[0126] [수학식 1]

$$Y = AXB^T$$

[0128] X는 m x n 크기의 2 차원 잔차 신호 블록이고, A는 수평 방향 1차원 n 포인트 변환을 의미하며, B^T는 수직 방향 1차원 m 포인트 변환을 의미한다. B^T는 B의 전치 행렬을 의미한다. 상기 m과 n은 다른 크기를 가질 수도 있고, 같은 크기를 가질 수도 있다. 또한, A와 B는 같은 변환 기저일 수도 있고, 서로 다른 변환 기저일 수도 있다. Y는 잔차 신호 블록 X를 변환하여 얻어지는 변환 블록을 의미한다.

[0129] 변환 블록 Y를 역 변환하는 과정에 사용되는 수식은 다음과 같다.

[0130] [수학식 2]

$$X = A^T Y B$$

[0132] 상기 수학식 1 및 수학식 2에 있어서, 수직 방향 변환과 수평 방향 변환은 그 수행 순서에 상관없이 유사한 결과를 얻을 수 있다. 그러나, 변환 계수들의 표현 범위가 16 비트와 같이 제한된 비트 정확도(bit precision)를 가지면, 수직 방향 변환과 수평 방향 변환의 수행 순서를 부호화기와 복호화기에서 동일하게 사용해야 한다. 이는 제한된 비트 정확도에 기인하여 연산의 중간에 버려지는 데이터가 발생하기 때문이다. 부호화기/복호화기에서 발생할 수 있는 미스 매치(mismatch)를 방지하기 위해서는 부호화기/복호화기에서 동일한 순서로 수직 방향 변환과 수평 방향 변환을 수행해야 한다.

[0133] 상기 수학식 1의 변환과 수학식 2의 역변환 수식이 성립되기 위해서 변환 기저들은 분리 특성(separability)과 직교성(orthogonality)을 만족할 필요가 있다. 이러한 제약이 필요한 이유는 계산량이 O(n⁴)에서 O(n³)으로 감소하며 A^T = A⁻¹을 만족하기 때문이다.

[0134] 변환 기저(transform basis vectors)들로 사용될 수 있는 기저(basis vectors, kernel)들의 종류에는 DCT-II(Discrete Cosine Transform type-II), DCT-V, DCT-VIII, DST-I(Discrete Sine Transform type-I), DST-VII 등이 있다. 실제 부호화기/복호화기에서는 계산 속도와 계산의 정확성을 위해 변환 기저들을 정수로 근사화하여 사용할 수 있다.

[0135] 부호화 단위의 크기가 m x n 인 경우, 분리 특성(separability)에 따라, 1차원 n 포인트 변환과 1차원 m 포인트 변환이 필요하다. 아래의 수학식 3은 4 ≤ m, n ≤ 64 인 경우 모든 부호화 단위의 크기에 대하여 적용될 수 있는 1차원 DCT-II 변환 기저의 예이다.

[0136] [수학식 3]

$$\cos\left(\frac{\pi \cdot (i+0.5) \cdot v}{N}\right) * w_0 ; \begin{cases} w_0 = \sqrt{\frac{1}{N}}, & \text{when } v = 0 \\ w_0 = \sqrt{\frac{2}{N}}, & \text{when } v \neq 0 \end{cases}$$

[0137]

- [0138] 부호화기/복호화기의 변환에 사용하기 위하여 실수 값을 갖는 변환 기저 각각의 원소들에 대하여 $\sqrt{N} * K_0$ 곱한 후 정수 단위로 반올림하여 정수 변환 기저를 생성할 수 있다. 도 6은 변환에 사용될 수 있는 DCT-II의 실수 기저 및 실수 기저에 소정의 값을 곱하여 얻어지는 정수 기저를 나타내는 예시적인 도면이다. 도 6에서는 K 값으로 64를 적용하여 얻어지는 정수 변환 기저를 도시한다. K 값이 커질수록 변환의 정확도는 향상될 수 있지만 부호화기/복호화기의 메모리 사용량이 증가할 수 있으므로 상황에 맞게 K 값이 선택될 수 있다. 역변환시에는 반대로 $\sqrt{N} * K$ 로 나누어 준다. 쉬프트 연산을 이용하여 계산량을 줄이기 위해 $K = 2^K$ (K는 1 보다 큰 정수)를 사용할 수 있다.
- [0139] 도 7은 변환에 사용될 수 있는 DST-VII의 실수 변환 기저 및 실수 변환 기저에 소정의 값을 곱하여 얻어지는 정수 변환 기저를 나타내는 예시적인 도면이다. 도 6을 참조하여 설명한 바와 같이, 부호화 단위의 크기가 $m \times n$ 인 경우 1차원 n 포인트 변환과 1차원 m 포인트 변환이 필요하다. 도 7에 도시한 바와 같이, $4 \leq m, n \leq 64$ 인 경우 모든 크기의 부호화 단위에 대하여 1차원 DST 변환도 적용이 가능하다.
- [0140] 도 8은 본 발명의 일 실시예에 따른, 인루프 필터링을 설명하기 위한 플로우 차트이다. 인루프 필터는 더블록킹 필터, 오프셋 보정부, ALF(Adaptive Loop Filter)중 적어도 하나를 포함할 수 있다. 또는 인루프 필터링은 복원 샘플 및/또는 예측 샘플에 적용되는 필터링을 포함할 수 있다.
- [0141] 도 9는 도 8에 따른 인루프 필터링을 수행하기 위해 사용되는, 블록 경계에 인접한 두 블록 및 그 내부의 픽셀을 나타내는 도면이다.
- [0142] 인루프 필터링은 블록 단위로 수행되는 변환 및 양자화 과정에 의해 블록 경계에서 발생하는 블록화 현상(Blocking Artifacts)를 감소시키기 위한 방법일 수 있다. 인루프 필터링은, 임의의 크기 이상의 수직(vertical) 블록 경계에 대해 수평 방향 필터링(Horizontal Filtering)을 수행한 후 수평(horizontal) 블록 경계에 대해 수직 방향 필터링(Vertical Filtering)을 실시할 수 있다. 또는 수직 방향 필터링을 수행한 후 수평 방향 필터링을 수행할 수도 있다. 먼저 실시된 필터링의 결과가 다음에 실시되는 필터링의 입력이 되기 때문에, 필터링의 순서는 부호화기/복호화기에서 동일해야 한다. 필터링의 순서가 동일하지 않으면, 부호화/복호화에서, 필터링된 화소값들 사이에 불일치(mismatch)가 발생할 수 있다.
- [0143] 인루프 필터링이 적용되는 임의의 크기는 부호화기/복호화기에서 미리 설정될 수도 있고, 비트스트림을 통해 시그널링되는 정보로부터 결정될 수도 있다. 예컨대, 임의의 크기는 4×4 , 8×8 , 16×16 등의 크기를 가질 수 있고, 가로와 세로의 길이가 상이하게 결정될 수도 있다.
- [0144] 인루프 필터링은 선택적으로 수행될 수 있으며, 인루프 필터링을 수행할 지 여부의 결정을 선결적으로 수행할 수 있다. 예를 들어, 필터링 수행 여부의 결정은 필터링의 수행 여부를 지시하는 정보, BS(Boundary Strength) 값, 블록 경계에 인접한 픽셀들의 변화량에 관한 적어도 하나의 변수 값 중 적어도 하나에 기초하여 결정될 수 있다. 필터링의 수행 여부를 지시하는 정보는 시퀀스, 픽처, 슬라이스, 타일 및 블록 레벨 중 적어도 하나 이상을 통해 시그널링 될 수 있다.
- [0145] 도 8에 도시된 방법에 따르면, 필터링 수행 여부를 결정하기 위해, 필터링되는 블록 경계에 인접한 적어도 하나의 블록의 부호화 정보를 이용할 수 있다. 여기서 부호화 정보는 블록의 예측 모드, 움직임 정보 및/또는 변환 계수 등을 포함할 수 있다.
- [0146] 단계 S801에서, 예측 유닛(Prediction Unit, PU) 및/또는 변환 유닛(Transform Unit, TU)의 경계에 대해, 블록 경계의 세기를 의미하는 BS(Boundary Strength) 값을 계산할 수 있다(S801). 예측 유닛 및/또는 변환 유닛은 블록 경계에 인접한 두 블록의 종류를 의미하며, BS 값의 계산에 이용되는 블록의 종류는 이에 한정되지 않는다. 예컨대, 후술하는 바와 같이 부호화 유닛(Coding Unit, CU)의 경계에 대해 BS 값을 계산할 수도 있다.
- [0147] 예컨대, 블록 경계에 인접한 두 블록 중 적어도 하나의 블록이 화면내 예측 모드로 부호화된 경우, BS 값은 제1 값으로 결정될 수 있다. 제1 값은 예를 들어, 2 또는 그 이상의 상수로 결정될 수 있다. 블록 경계에 인접한 두 블록이 모두 화면간 예측 모드로 부호화된 경우, 두 블록의 움직임 벡터(motion vector)의 값, 움직임 벡터의 개수, 참조 픽처(reference picture)의 동일 여부 및/또는 두 블록 중 적어도 한 블록이 0이 아닌 양자화된 잔차 신호 계수(quantized residual coefficient)를 갖는지의 여부 등에 따라 BS 값은 제2 값을 가질 수 있다. 제2 값은 제1 값보다 작은 상수(예를 들어, 0, 1 등)일 수 있다. BS 값은 제1 값 및 제2 값에 한정되지 않고, 예컨대, BS 값의 결정 기준을 보다 세분화하여, 제3 값, 제4 값 등 다양한 단계의 값을 갖도록 할 수 있다. 이하에서는 BS 값이 0, 1 및 2 중 하나의 값을 갖는 실시예를 기준으로 설명한다. 그러나 전술한 바와 같이, BS

값은 그 이상의 다양한 값을 가질 수 있으므로, 후술하는 실시예에 한정되지 않는다.

- [0148] 단계 S802에서는 BS값이 0인지 아닌지에 대한 판단이 수행될 수 있다. BS값이 0인 경우에는 필터링을 수행하지 않을 수 있다(S803).
- [0149] BS값이 0이 아닌 경우, 단계 S804가 수행될 수 있다. 단계 S804에서, 블록 경계에 인접한 픽셀들의 픽셀 변화량을 측정하는 델타(delta)값을 계산할 수 있다. 델타값은 소정의 블록 단위로 계산될 수 있다. 이때 소정의 블록 단위에 속하는 전부 또는 일부 라인(행 또는 열)이 이용될 수 있다. 상기 일부 라인의 위치 및/또는 개수는 부호화기/복호화기에 기-약속된 고정된 것일 수도 있고, 블록의 크기/형태에 따라 가변적으로 결정될 수도 있다. 또한, 라인 상에 위치한 하나 또는 그 이상의 픽셀들이 이용될 수 있고, 이때 복수의 픽셀들은 연속적으로 배열된 것일 수도 있고, 일정 간격으로 떨어진 비연속적인 픽셀들일 수도 있다. 예를 들어, 델타값은 4x4 블록단위마다 도 9의 블록 내부의 밝은 회색으로 표시된 영역의 픽셀들의 변화량에 의해 계산할 수 있다.
- [0150] 도 9에서, 중앙의 붉은 수직선은 좌우의 블록 P와 블록 Q 사이의 경계를 의미한다. 블록 P 및 블록 Q 내부의 격자의 각각은 하나의 화소(픽셀)를 나타낸다. 각 격자 내부에 표시된 부호는 해당 격자에 대응하는 화소를 지시하기 위해 사용되는 인덱스이다.
- [0151] 델타값의 계산은 블록 경계 주변 픽셀들의 밝기 값의 변화량에 기초하여 계산될 수 있다. 블록 경계 주변 픽셀들의 밝기 값의 변화량은 블록 경계를 중심으로 좌측 및/또는 우측(또는 상측 및/또는 하측)에 위치하는 픽셀들의 밝기 값의 변화량에 기초하여 계산될 수 있다. 예를 들어, 델타값의 계산은, 도 9에 도시된 블록 P와 블록 Q의 화소들을 이용하여, 아래의 수학적 식 4에 따라 계산할 수 있다. 수학적 식 4는 해당 블록 경계에서의 에지(edge)의 세기를 계산하는 식일 수 있다.
- [0152] [수학적 식 4]
- [0153] $dp0 = \text{abs}(p2,0 - 2*p1,0 + p0,0)$
- [0154] $dp3 = \text{abs}(p2,3 - 2*p1,3 + p0,3)$
- [0155] $dq0 = \text{abs}(q2,0 - 2*q1,0 + q0,0)$
- [0156] $dq3 = \text{abs}(q2,3 - 2*q1,3 + q0,3)$
- [0157] $dp = dp0 + dp3$
- [0158] $dq = dq0 + dq3$
- [0159] $dpq0 = dp0 + dq0$
- [0160] $dpq3 = dp3 + dq3$
- [0161] $\text{delta} = dpq0 + dpq3$
- [0162] 또한, 단계 S804에서, 2 가지의 임계값인 베타(β)값과 t_c 값을 유도할 수 있다. 베타값과 t_c 값은 양자화 파라미터(Quantization Parameter, QP)와 아래의 표3에 기초하여 유도될 수 있다. 상기 양자화 파라미터는 블록 경계에 인접한 두 블록 또는 두 블록 중 적어도 하나의 블록의 양자화와 관련된 파라미터로부터 유도될 수 있다. 표 3의 Q는 양자화 파라미터로부터 유도되는 값이다. 양자화 파라미터로부터 Q값이 결정되면, 표3을 참조하여 β' 와 t_c' 값을 결정할 수 있다. 베타값과 t_c 값은 결정된 β' 와 t_c' 에 기초하여 결정될 수 있다.

표 3

Q	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
β'	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	7	8
t_c'	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Q	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37
β'	9	10	11	12	13	14	15	16	17	18	20	22	24	26	28	30	32	34	36
t_c'	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	4	4	4
Q	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53			
β'	38	40	42	44	46	48	50	52	54	56	58	60	62	64	-	-			
t_c'	5	5	6	6	7	8	9	10	11	13	14	16	18	20	22	24			

[0163]

[0164]

베타값 및/또는 t_c 값은 필터링 수행 여부를 결정하기 위해 사용될 수 있다. 또한 베타값 및/또는 t_c 값은 필터링이 수행되는 것으로 결정된 경우, 필터링 타입을 선택하는데 사용될 수 있다. 필터링 타입은 필터링이 적용되는 픽셀들의 범위, 필터링의 강도 등에 따라 달라질 수 있다. 예컨대, 필터링 타입에는 강한 필터링(Strong filtering)과 약한 필터링(Weak filtering)이 포함될 수 있다. 그러나, 필터링 타입은 이에 한정되지 않으며, 예컨대, 필터링의 강도가 각각 상이한 3 종류 이상의 필터링 타입을 가질 수도 있다. 또는, 후술하는 바와 같이, 블록 경계에 인접한 두 블록의 형태, 크기 및/또는 특성에 따라 각각 다른 필터링이 적용되는 경우, 각각을 별개의 필터링 타입으로 설정할 수 있다. 또한 베타값 및/또는 t_c 값은 필터링 수행 시, 필터링된 픽셀의 클리핑(clipping)을 위해 사용될 수 있다.

[0165]

단계 S805에서, 델타값과 베타값에 대한 비교가 수행될 수 있다. 단계 S805의 비교 결과에 기초하여 필터링 수행 여부를 결정할 수 있다. 델타값이 베타값보다 작지 않으면, 필터링을 수행하지 않을 수 있다(S803). 델타값이 베타값보다 작으면, 필터링을 수행하는 것으로 결정하고, 해당 블록 경계에 대해 적용될 필터링 타입을 결정할 수 있다. 또는 델타값이 베타값보다 작은 경우에 필터링을 수행하지 않는 것으로 결정하고, 델타값이 베타값보다 작지 않은 경우에, 필터링을 수행하는 것으로 결정할 수도 있다. 도 8에 도시된 실시예에서는 필터링 타입은 강한 필터링과 약한 필터링의 두가지를 포함하며, 단계 S806에서는 강한 필터링과 약한 필터링 중 어느 필터링을 수행할지의 여부를 결정할 수 있다. 강한 필터링과 약한 필터링 중 어느 필터링을 수행할 지 여부는 블록 경계에 인접한 두 블록 내부의 픽셀값의 변화량과 소정의 임계치를 비교하여 결정될 수 있다. 이때 사용되는 블록 내부의 픽셀값의 변화량은 델타값의 계산에서 이용된 변화량일 수 있다. 또는 블록 경계에 인접한 각각의 블록 내부의 픽셀값들 사이의 변화량과 소정의 임계치가 비교될 수도 있다. 또는 블록 경계를 중심으로 인접하는 둘 이상의 픽셀값들 사이의 변화량과 소정의 임계치가 비교될 수도 있다. 상기 소정의 임계치는 기정의된 상수값, 비트스트림을 통해 시그널링되는 값, 블록의 형태, 크기 등의 특성에 따라 결정되는 값 전술한 베타값 및 t_c 값 중 적어도 하나 이상의 값이 될 수 있다. 또는 베타값 및/또는 t_c 값을 스케일링한 값이 사용될 수도 있다. 또한, 상기 픽셀값들 사이의 변화량은 상기 임계치를 이용하여 스케일링 된 후 상기 임계치와 비교될 수 있다. 예컨대, 강한 필터링과 약한 필터링 중 어느 필터링을 수행할지의 여부는 아래의 수학식 5 및/또는 수학식 6의 조건을 만족하는지 여부에 의해 결정될 수 있다.

[0166]

[수학식 5]

[0167]

조건 A: $2 * dpq0 < \beta / 4$

[0168]

조건 B: $abs(p3,0 - p0,0) + abs(q3,0 - q0,0) < \beta / 8$

[0169]

조건 C: $abs(p0,0 - q0,0) < (5 * t_c + 1) / 2$

[0170]

[수학식 6]

[0171]

조건 A: $2 * dpq3 < \beta / 4$

[0172]

조건 B: $abs(p3,3 - p0,3) + abs(q3,3 - q0,3) < \beta / 8$

[0173]

조건 C: $abs(p0,3 - q0,3) < (5 * t_c + 1) / 2$

- [0174] 상기 수학식 5는 도 9의 첫 번째 열에 대한 식이고, 상기 수학식 6은 도 9의 네 번째 열에 대한 식이다. 상기 수학식 5 및/또는 수학식 6의 조건이 모두 만족하면, 해당 블록 경계에 대해 강한 필터링을 수행할 수 있다(S807). 그렇지 않으면 약한 필터링을 수행할 수 있다(S808). 델타값의 계산과 관련하여 전술한 바와 같이, 필터링 타입의 결정에는 블록 경계에 인접한 소정의 블록 단위에 속하는 전부 또는 일부 라인(행 또는 열)이 이용될 수 있다. 상기 일부 라인의 위치 및/또는 개수는 부호화기/복호화기에 기-약속된 고정된 것일 수도 있고, 블록의 크기/형태에 따라 가변적으로 결정될 수도 있다. 또한, 라인 상에 위치한 하나 또는 그 이상의 픽셀들이 이용될 수 있고, 이때 복수의 픽셀들은 연속적으로 배열된 것일 수도 있고, 일정 간격으로 떨어진 비연속적인 픽셀들일 수도 있다. 블록 경계에 인접한 두 블록의 형태 및/또는 크기가 상이한 경우, 본 명세서에서 픽셀의 위치를 특정하기 위해 사용된 인덱스(예컨대, $p0,0$, $p1,0$, $q2,0$ 등)는 적응적으로 달라질 수 있다.
- [0175] 강한 필터링이 수행되는 경우(S807), 블록 경계에 인접한 두 블록을 모두 사용하여, 블록당 m 개(여기서, m 은 3 또는 그 이상의 상수)의 픽셀이 필터링될 수 있다. 약한 필터링이 수행되는 경우(S808), 블록 경계에 인접한 두 블록을 모두 사용하거나, 혹은 하나의 블록을 사용하여 블록당 n 개(여기서, n 은 m 보다 작은 상수)의 픽셀이 필터링될 수 있다. 도 8의 단계 S807 및 S808의 하단에 도시한 필터링 적용 범위는 m 은 3, n 은 2인 경우를 예시적으로 나타내며, m , n 은 이에 한정되지 않는다. 또한, 도 8에 도시한 필터링 방법은 반드시 이에 한정되지 않는다. 즉, 각 단계는 필요에 따라 스킵될 수 있으며, 2 이상의 단계를 하나의 단계로 병합하거나, 하나의 단계를 2 이상의 단계로 분할하는 것도 가능하다. 또한, 일부 단계의 수행 순서가 변경될 수 있다.
- [0176] 본 발명에 따라 기본 블록을 쿼드 트리 구조 및/또는 이진 트리 구조를 이용하여 분할하는 경우, 도 3(b), 도 4(a)에 도시된 바와 같이, 부호화 단위가 되는 블록의 모양 및 크기가 매우 가변적이며, 직사각형 형태의 부호화 단위를 포함하도록 분할될 수 있다.
- [0177] 본 발명에 따라 쿼드 트리 구조 및/또는 이진 트리 구조를 이용하여 블록을 분할하면, 더 이상 분할되지 않는 블록은 부호화 단위로서 예측, 변환 및/또는 양자화의 단위가 될 수 있다. 즉, 기본 블록은 직사각형 또는 직사각형의 형태를 갖는 다양한 크기의 부호화 단위로 분할될 수 있고, 각 부호화 단위는 예측 또는 변환을 위해 다시 분할될 필요가 없이, 그 자체로 예측 유닛 및/또는 변환 유닛이 될 수 있다.
- [0178] 부호화 단위는 곧바로 변환 유닛이 될 수 있으므로, 부호화 단위의 크기가 다양해짐에 따라 변환 유닛도 매우 다양한 크기를 가질 수 있다. 따라서 변환 유닛 경계에서 발생하는 블록화 현상이 블록의 형태, 크기 및/또는 특성에 따라 보다 다양하게 나타날 수 있다. 그러므로 인루프 필터링의 강도 및/또는 적용 범위를 블록 경계에 인접한 블록의 형태, 크기 및/또는 특성에 맞추어 수행할 필요가 있다.
- [0179] 도 10은 본 발명에 따라 쿼드 트리 구조 및/또는 이진 트리 구조를 이용하여 분할된 블록의 구조 및 그에 따른 인루프 필터링이 적용되는 블록 경계들을 나타낸 도면이다. 도 10에서 붉은 선으로 표시된 블록 경계는 인루프 필터링이 적용될 수 있는 블록 경계를 예시적으로 나타낸 것이다.
- [0180] 부호화기와 복호화기의 복잡도를 고려하여 블록의 너비(width)나 높이(height)가 4인 블록(ex. 4×8 , 8×4 , 4×4 등)들의 블록 경계에 대해서는 필터링을 수행하지 않을 수 있다. 즉, 도 10에서 검정 선으로 표시된 블록 경계에 대해서는 필터링이 수행되지 않을 수 있다.
- [0181] 도 11은 본 발명의 다른 실시예에 따른 필터링을 설명하기 위한 플로우 차트이다. 도 11을 참조하여, 도 10에 도시된 바와 같이, 비균등 블록을 포함하는 영상에 대한 수정된 인루프 필터링 방법을 설명한다. 여기서 비균등 블록이란, 블록의 너비와 높이가 상이한 블록을 의미한다.
- [0182] 도 11의 각 단계의 설명에 있어서, 도 8의 각 단계에 대한 설명과 공통되는 부분은 중복하여 설명하지 않고, 이하에서는, 도 11의 방법과 도 8의 방법과의 차이점을 위주로 설명한다. 도 8의 단계 S801에서는 PU와 TU를 기준으로 BS값을 계산하였으나, 도 11의 단계 S1101에서의 BS의 계산은 블록 경계에 인접한 두 부호화 단위(Coding Unit, CU)를 기준으로 수행될 수 있다.
- [0183] BS 값은 전술한 바와 같이, 제1 값 및/또는 제2 값을 갖거나 그 이상의 값을 가질 수 있다. 단계 S1102에서, BS 값이 0인지의 여부의 판단에 따라, 필터링이 수행되지 않거나(S1103), 단계 S1104에 따라 필터링이 수행될 수 있다.
- [0184] 단계 S1104에서는 델타값, 베타값 및 t_c 값이 계산/유도될 수 있다. 단계 S1105에서 델타값과 베타값을 비교할 수 있다. 단계 S1105에서 델타값이 베타값보다 작은 것으로 판단되면, 블록 경계에 인접한 블록 중 적어도 하나의 블록이 비균등 블록인지 확인할 수 있다(S1106). 블록 경계에 인접한 두 개의 블록이 모두 정방형 블록인 경

우, 도 8의 S806 내지 S808에 따른 필터링이 수행될 수 있다(S1107). 블록 경계에 인접한 두 개의 블록 중 적어도 하나의 블록이 비균등 블록인 경우, 본 발명의 다른 실시예에 따라, S1108 이후의 과정이 수행될 수 있다. 또는, 블록 경계에 인접한 두 개의 블록의 형태 및/또는 크기가 서로 상이한 경우에도 본 발명의 다른 실시예에 따라, S1108 이후의 과정이 수행될 수 있다. 단계 S1108에서, 소정의 기준에 따라 필터링 타입을 선택할 수 있다. 필터링 타입은 예컨대, 강한 필터링과 약한 필터링을 포함할 수 있으나, 전술한 바와 같이 이에 한정되지 않는다. 강한 필터링이 선택되면, 단계 S1109에서 강한 필터링을 수행하며, 약한 필터링이 선택되면, S1110에서 약한 필터링을 수행할 수 있다. 단계 S1109, S1110의 하단에 예시적으로 도시한 바와 같이, 블록 경계에 인접한 두 블록에 대해 필터링이 적용되는 범위가 상이할 수 있다. 구체적인 필터링 적용 범위에 대해서는 후술한다. 도 11에 도시한 필터링 방법은 반드시 이에 한정되지 않는다. 즉, 각 단계는 필요에 따라 스킵될 수 있으며, 2 이상의 단계를 하나의 단계로 병합하거나, 하나의 단계를 2 이상의 단계로 분할하는 것도 가능하다. 또한, 일부 단계의 수행 순서가 변경될 수 있다. 또한, 도 11을 참조하여 설명한 실시예와 상충되지 않는 범위에서, 도 8을 참조하여 설명한 내용이 동일하게 적용될 수 있다.

[0185] 도 12는 도 11의 S1109에 따라 강한 필터링이 적용되는 화소를 예시하는 도면이다. 도 12에 도시된 바와 같이, 블록 경계(붉은 선으로 표시)에 인접한 두 블록은 각각 8x8 및 16x8일 수 있다. 16x8 크기를 갖는 블록은 비균등 블록에 해당한다. 도 8을 참조하여 설명한 바와 같이, 블록 경계에 강한 필터링을 적용할지 약한 필터링을 적용할지 여부는 두 블록 또는 두 블록 중 적어도 한 블록의 양자화 파라미터에 기초하여 결정될 수 있다. 블록 경계에 인접한 두 블록 중 적어도 하나의 블록이 비균등 블록인 경우, 블록 경계에 인접한 두 블록의 너비 및/또는 높이가 함께 고려될 수 있다. 구체적으로, 블록 경계에 인접한 두 블록 P와 블록 Q의 너비와 높이를 고려하여, 블록의 크기가 더 큰 블록에는 블록 경계 주변 픽셀들을 더 많이 필터링할 수 있다. 이는, 변환 및 양자화로 인해 나타나는 블록화 현상이 양자화 파라미터뿐 아니라, 블록 경계에 인접한 두 블록의 크기와도 관련이 있기 때문이다. 예컨대, 블록 경계에 인접한 두 블록 중, 상대적으로 큰 블록은 블록의 DC성분을 제외한 AC성분들이 양자화 스텝 사이즈(Quantization step, Qstep)로 인해 양자화되어 제거될 수 있으며, 이 때, 상대적으로 큰 블록 내부에서 AC부분이 제거된 해당 위치가, 인접하는 블록의 DC 부분과 인접할 수도 있다. 이와 같은 경우, 블록화 현상이 심각하게 나타날 수 있다. 본 발명의 다른 실시예에 따른 필터링 방법은 이러한 불일치를 줄이기 위해 블록의 형태, 크기 및/또는 특성을 고려하여 필터링을 수행할 수 있다.

[0186] 블록 경계의 강도(BS)의 유도, 필터링의 수행 여부 결정, 강한 필터링과 약한 필터링의 선택 등의 과정은 블록 경계에 인접한 두 블록의 크기에 기초하여 수행될 수도 있고, 도 8을 참조하여 설명한 방법에 의해 수행될 수도 있다.

[0187] 도 12에 도시된 바와 같이, 블록 경계에 인접한 두 블록의 크기가 각각 8x8 및 16x8이고, 해당 블록 경계에 대해 강한 필터링이 수행되는 경우, 블록의 크기가 더 큰 블록에 대해서는 더 많은 화소를 필터링할 수 있다. 예컨대, 8x8 블록에 대해서는 3개의 화소(b, c, d)를 필터링하고, 이보다 상대적으로 크기가 큰 16x8 블록에 대해서는 4개의 화소(e, f, g, h)를 필터링할 수 있다. 필터링되는 화소의 개수는 이에 한정되지 않으며, 상대적으로 크기가 더 큰 블록에 대해서는 상대적으로 크기가 작은 블록보다 더 많은 화소가 필터링될 수 있다. 도 12에 도시된 필터링 대상 화소들에 대한 필터링은 블록 경계에 인접한 두 블록 내부의 픽셀값들의 평균 및/또는 가중 평균을 이용하거나, 이에 준하는 중간값의 계산에 의해 수행될 수 있다. 예컨대, 아래의 수학식 7을 이용하여 필터링을 수행할 수 있다.

[0188] [수학식 7]

[0189]
$$b' = (2a + 2b + c + d + e + f)/8$$

[0190]
$$c' = (b + c + d + e + f)/5$$

[0191]
$$d' = (b + 3c + 5d + 3e + f + 8)/16$$

[0192]
$$e' = (c + 3d + 5e + 3f + g + 8)/16$$

[0193]
$$f' = (c + d + e + f + g)/5$$

[0194]
$$g' = (c + d + e + f + g + 2g + 2h)/8$$

[0195]
$$h' = (2d + 2e + f + g + h + i + 4)/8$$

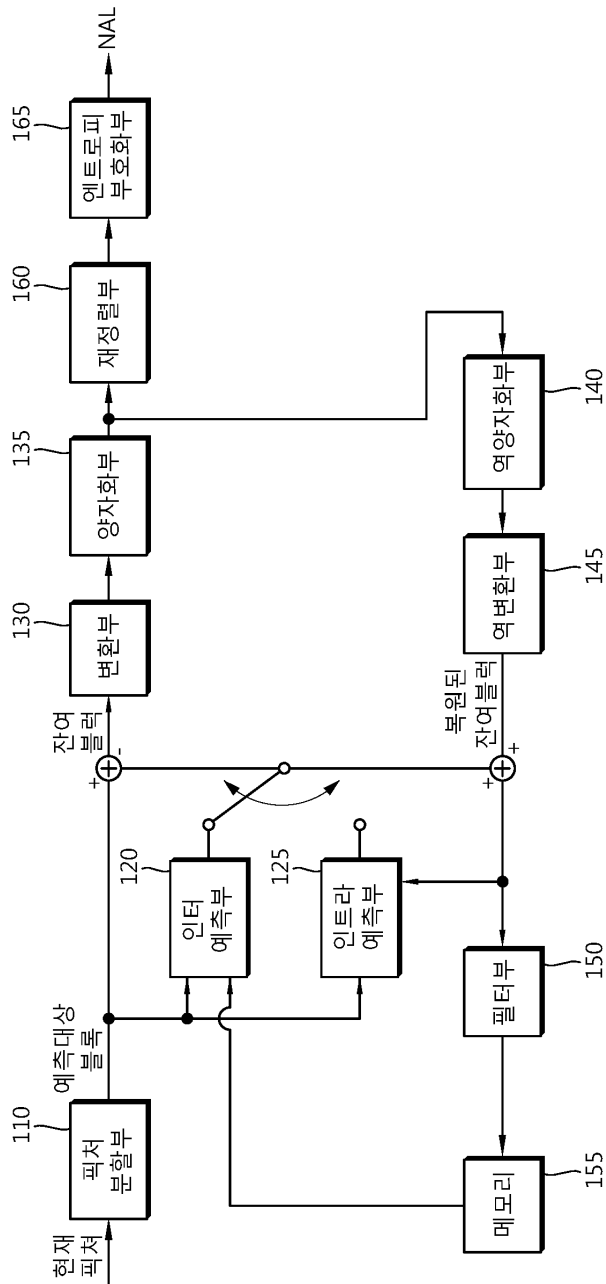
[0196] 상기 수학식 7에서, 알파벳 a, b, c, d, e, f, g, h, i 로 표현한 부분은 예측 신호 및 복호화된 잔차 신호가 더해져 복원된 화소값을 의미하며, b', c', d', e', f', g', h'로 표현된 알파벳은 필터 계수를 통해 필터링이

수행된 후 수정된 화소값을 의미한다.

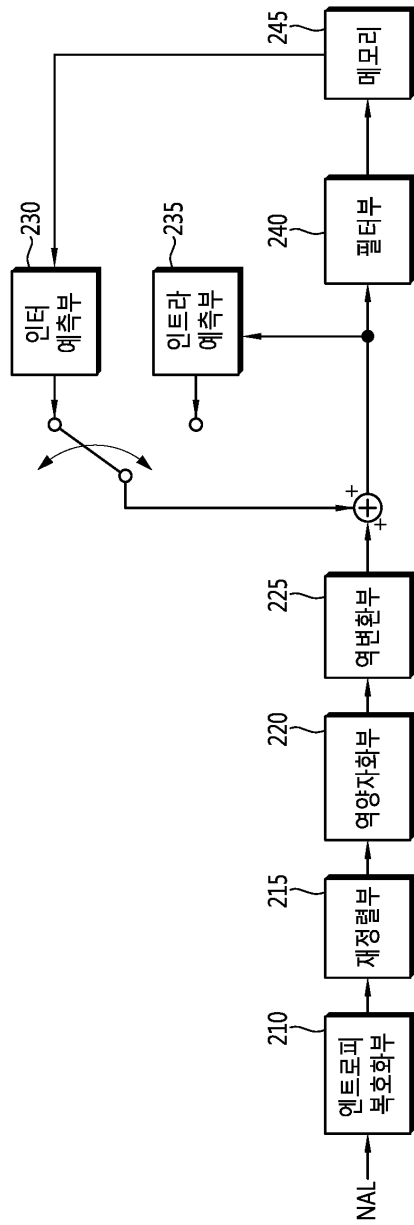
- [0197] 도 13은 도 11의 S1110에 따라 약한 필터링이 적용되는 화소를 예시하는 도면이다. 도 13에 도시된 바와 같이, 블록 경계에 인접한 두 블록의 크기가 각각 8x8 및 16x8이고, 해당 블록 경계에 대해 약한 필터링이 수행되는 경우, 예컨대, 8x8 블록에 대해서는 2개의 화소(c, d)를 필터링하고, 이보다 상대적으로 크기가 큰 16x8 블록에 대해서는 3개의 화소(e, f, g)를 필터링할 수 있다.
- [0198] 필터링의 수행 과정은 블록 경계에 인접한 두 블록 내부의 픽셀값들의 차이값에 기초하여 Δ 값을 계산함으로써 수행될 수 있다. 이때, 블록 경계에 인접한 두 픽셀값들의 차이값에 보다 큰 가중치를 두어 Δ 값을 계산할 수 있다. 이때, 블록 경계에 인접한 두 픽셀값들의 차이값은 Δ 값과 비례하는 관계일 수 있다. 또한, 블록 경계에 직접적으로 인접하지 않은 픽셀값들의 차이값을 이용하여 Δ 값을 계산할 수 있다. 이때, 블록 경계에 직접적으로 인접하지 않은 픽셀값들의 차이값은 Δ 값과 반비례하는 관계일 수 있다. Δ 값은 상기 픽셀값들의 차이값들을 하나 이상 이용하여 계산될 수 있으며, 계산된 값은 기정의된 상수, 블록의 형태, 크기 등의 특성에 따라 결정되는 값 및/또는 비트스트림을 통해 시그널링 되는 값을 이용하여 스케일링될 수 있다. 예컨대, 아래의 수학적식 8과 같이 Δ 값을 계산할 수 있다. 계산된 Δ 값은 필터링의 대상이 되는 픽셀값에 더하거나 빼줌으로써 필터링된 픽셀값을 계산할 수 있다.
- [0199] [수학적식 8]
- [0200]
$$\Delta = (9 \cdot (q_0 - p_0) - 3 \cdot (q_1 - p_1) + 8) \gg 4$$
- [0201] 약한 필터링은 하나의 블록에 대해서만 수행할 수도 있고, 두 블록 모두에 대해 수행될 수도 있다. 블록 경계에 인접한 두 블록 중 적어도 하나의 블록이 비균등 블록인 블록 경계에 대해 약한 필터링이 적용되는 경우, 블록 경계의 픽셀인 d와 e는, 각각의 픽셀값에 수학적식 8에 의해 계산된 차이값 Δ 를 더하거나 빼줌으로써 필터링될 수 있다. 비균등 블록 내부에 위치한 픽셀들 (f, g)의 경우, 블록 경계에 인접한 두 블록 내부의 픽셀값들의 평균 및/또는 가중 평균을 이용하거나, 이에 준하는 중간값의 계산에 의해 수행될 수 있다. 예로서 아래의 수학적식 9와 같이, 블록 경계의 주변 픽셀값을 이용하여 필터링을 수행할 수 있다.
- [0202] [수학적식 9]
- [0203]
$$f' = (e + 5f + 2g) \gg 3$$
- [0204]
$$g' = (2f + 4g + 2h) \gg 3$$
- [0205] 도 13에서, 만약 블록 경계의 왼쪽이 16x8 블록인 경우는 두 픽셀 b와 c에 대해, 상기 수학적식 9를 유사하게 적용할 수 있다.
- [0206] 본 발명의 다른 실시예에 따른 필터링은 블록 경계에 위치한 두 블록 중 적어도 하나의 블록이 비균등 블록인 경우에 수행될 수 있다. 또한, 본 발명의 다른 실시예에 따른 필터링은 블록 경계에 위치한 두 블록의 크기가 다른 경우에도 적용될 수 있다. 블록 경계에 인접한 두 블록의 크기가 같은 경우 또는 두 블록 모두 정방형인 경우, 도 8을 참조하여 설명한 필터링이 수행될 수 있다.
- [0207] 도 14는 본 발명에 따라 약한 필터링이 수행되는 경우에 필터링이 적용되는 픽셀의 범위를 예시적으로 나타내는 도면이다. 도 14에서 굵은 선은 블록의 블록 경계를 나타낸다. 굵은 선의 내부의 격자는 각각의 블록 내부의 픽셀을 의미한다. 도 14에 있어서, 예컨대, 필터링이 수행되는 최소 블록 크기는 8x8이다. 따라서, 가로 및/또는 세로의 크기가 4인 블록의 경계에 대해서는 필터링이 수행되지 않을 수 있다. 도 14에서, 진한 회색으로 표시한 부분은, 도 8을 참조하여 설명된 필터링이 적용되는 화소 범위를 나타낸다. 도 14에서, 밝은 회색으로 표시한 부분은 비균등 블록(16x8)의 내부의 영역으로서, 도 11을 참조하여 설명된 필터링이 적용되는 화소 범위를 나타낸다.

도면

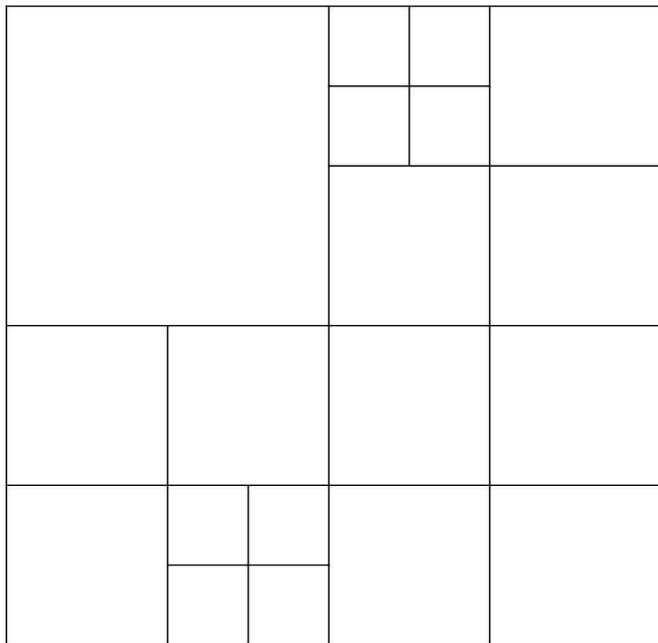
도면1



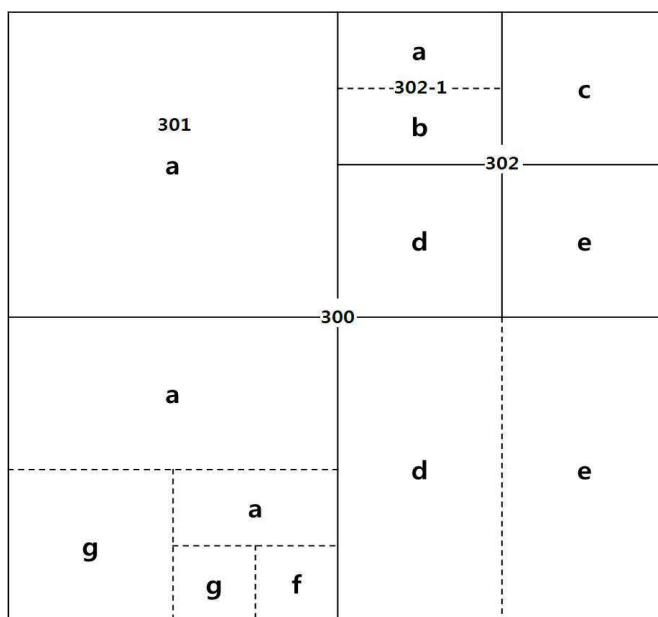
도면2



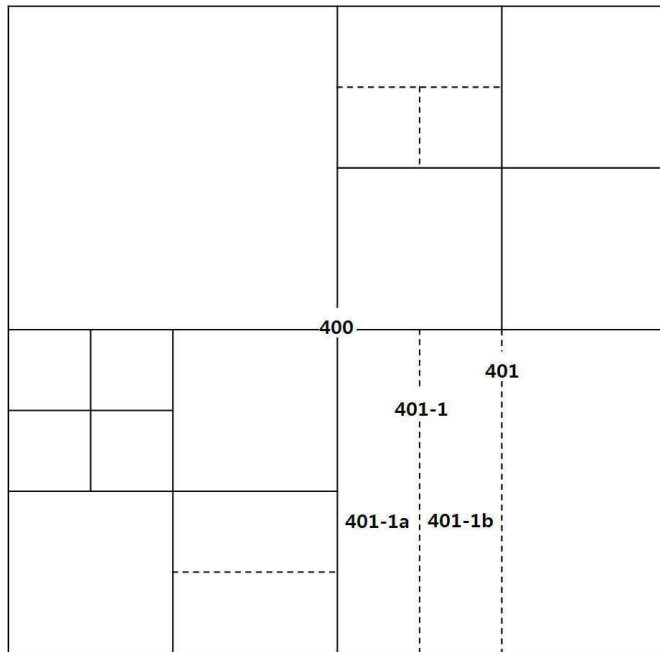
도면3a



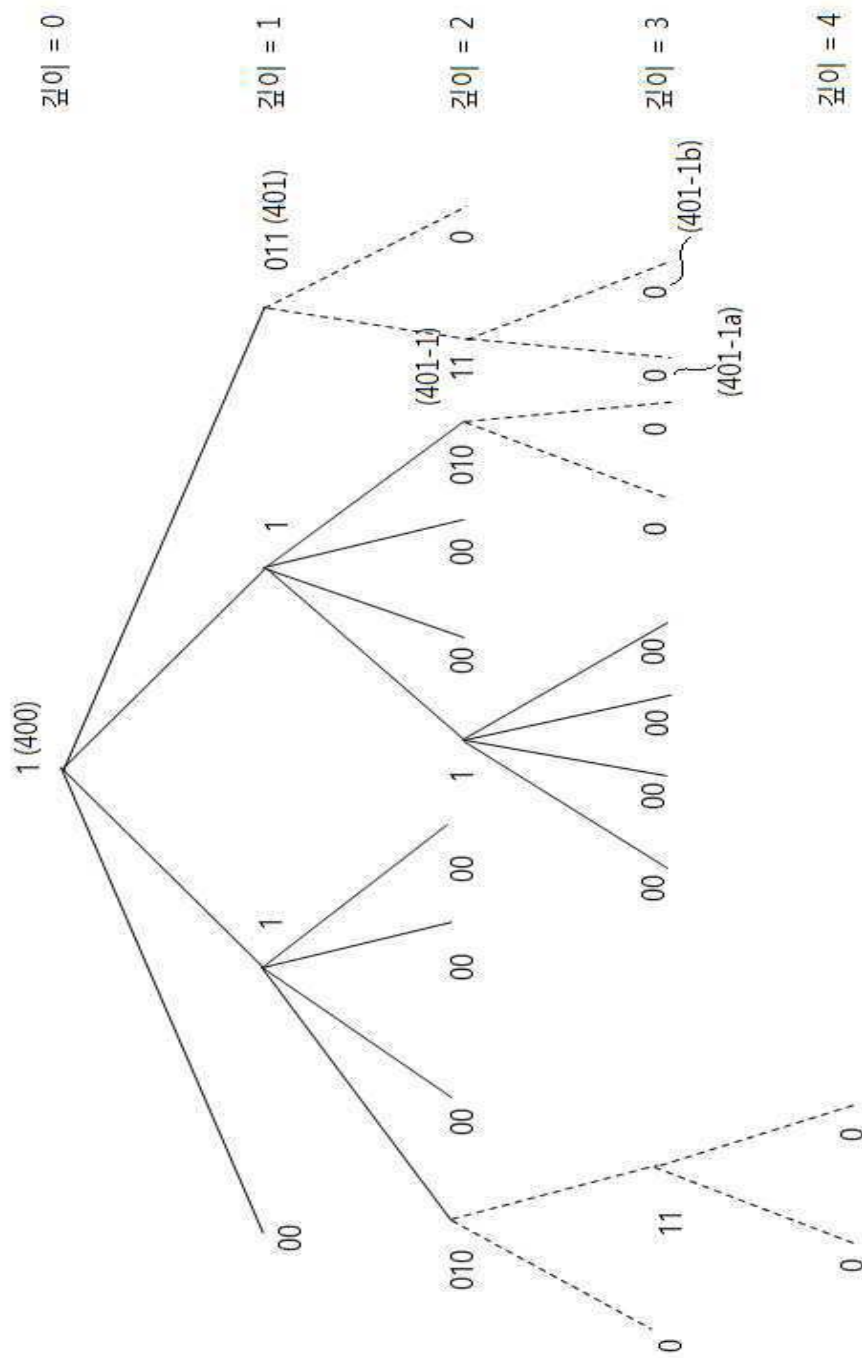
도면3b



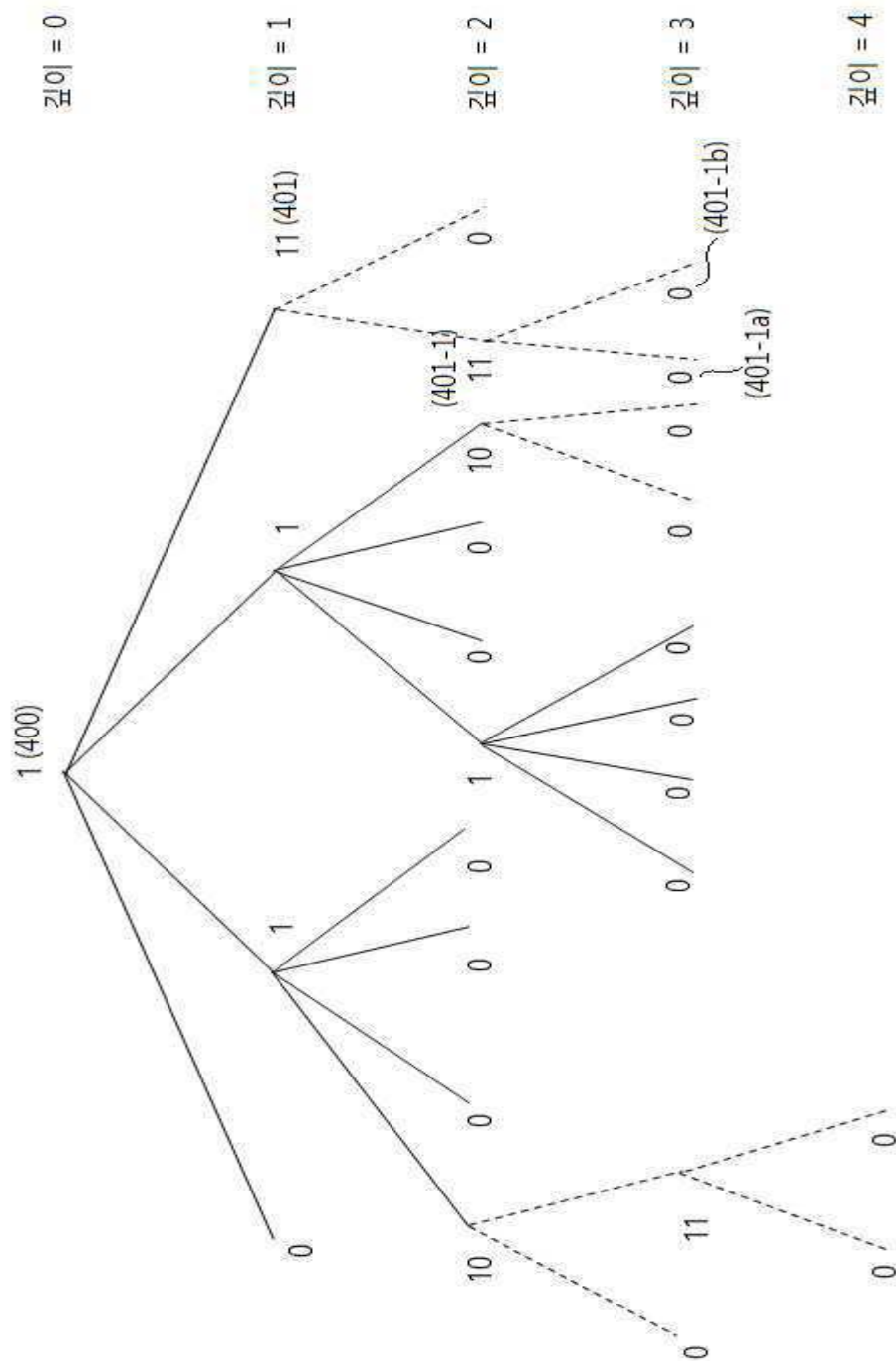
도면4a



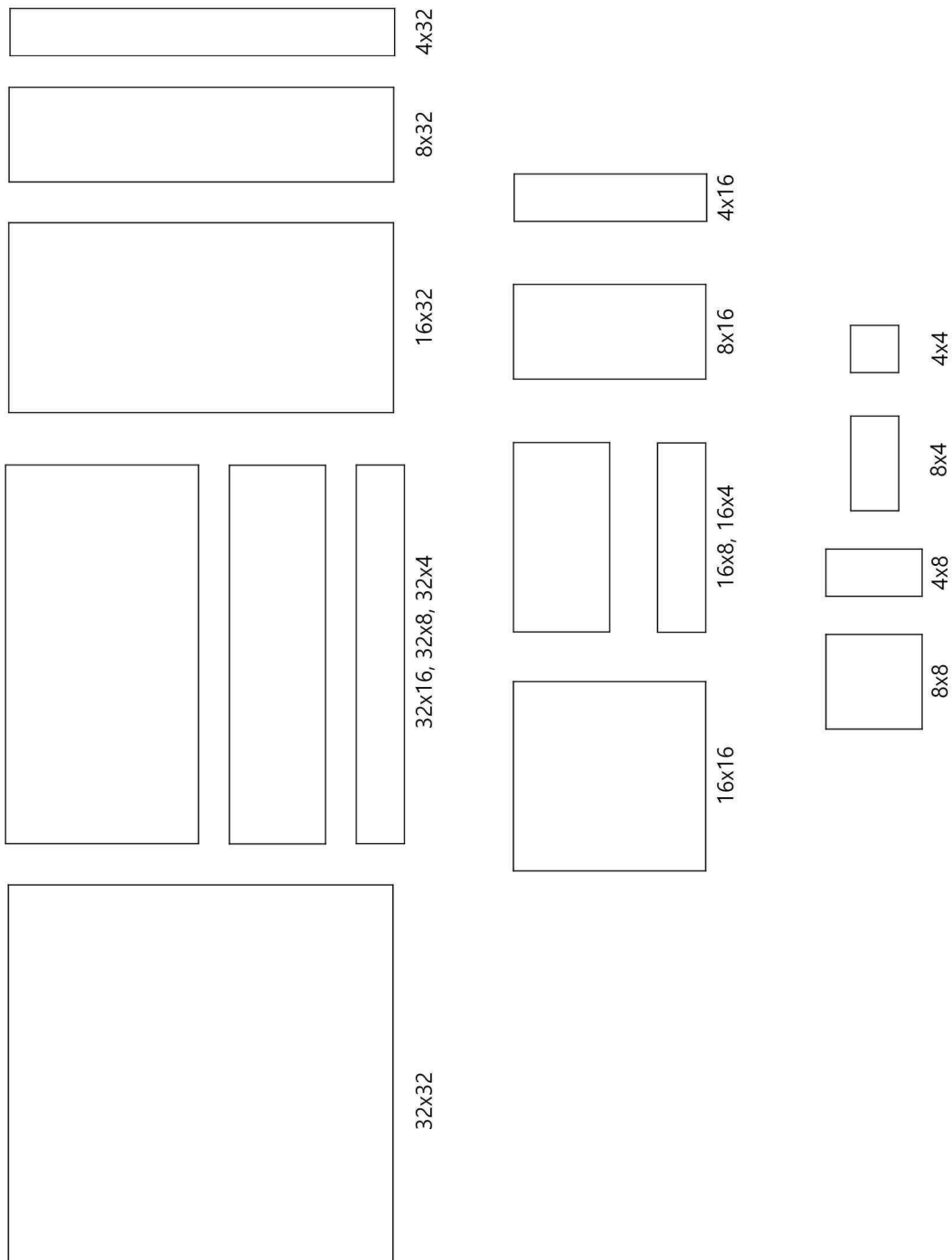
도면4b



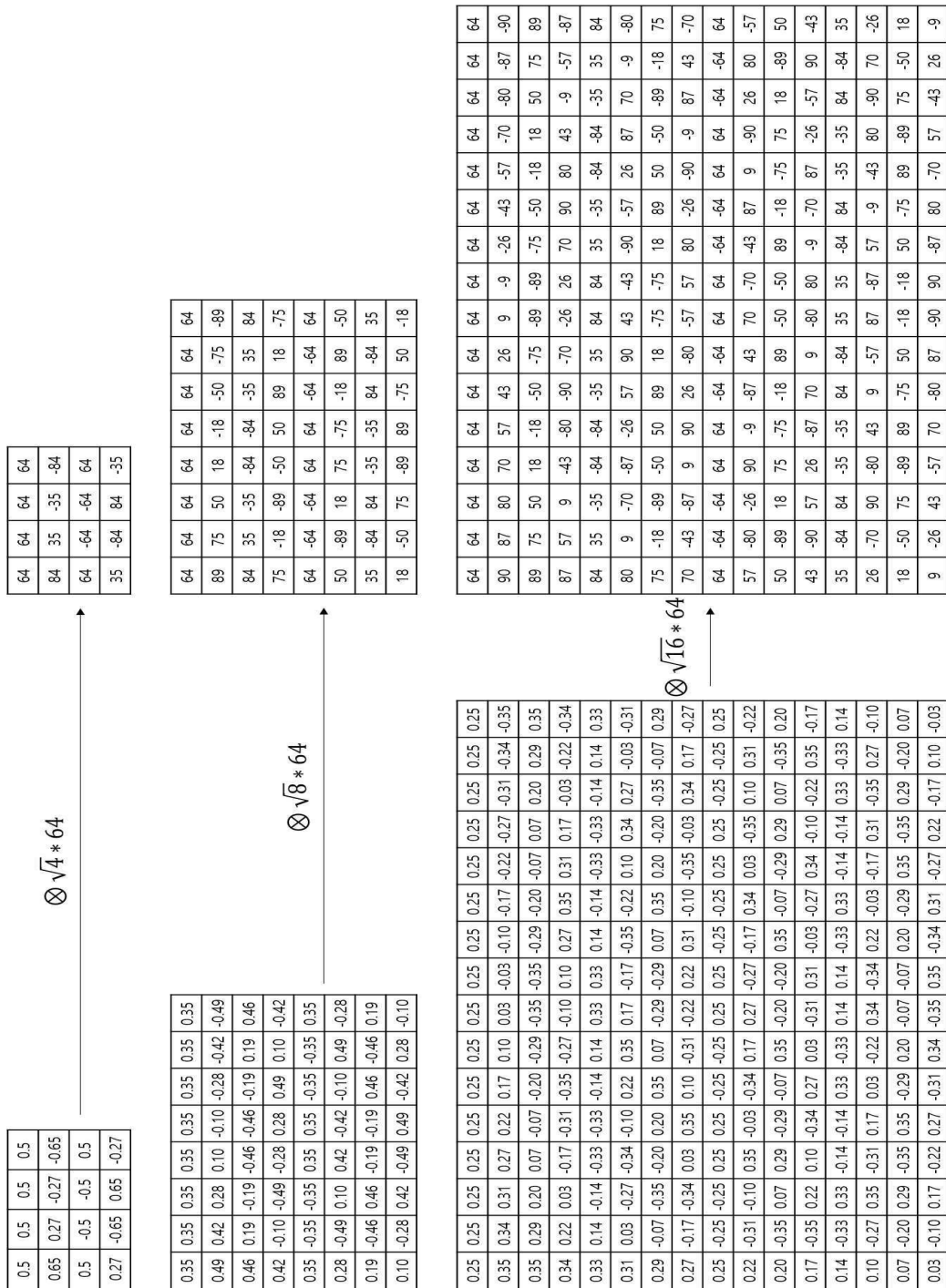
도면4c



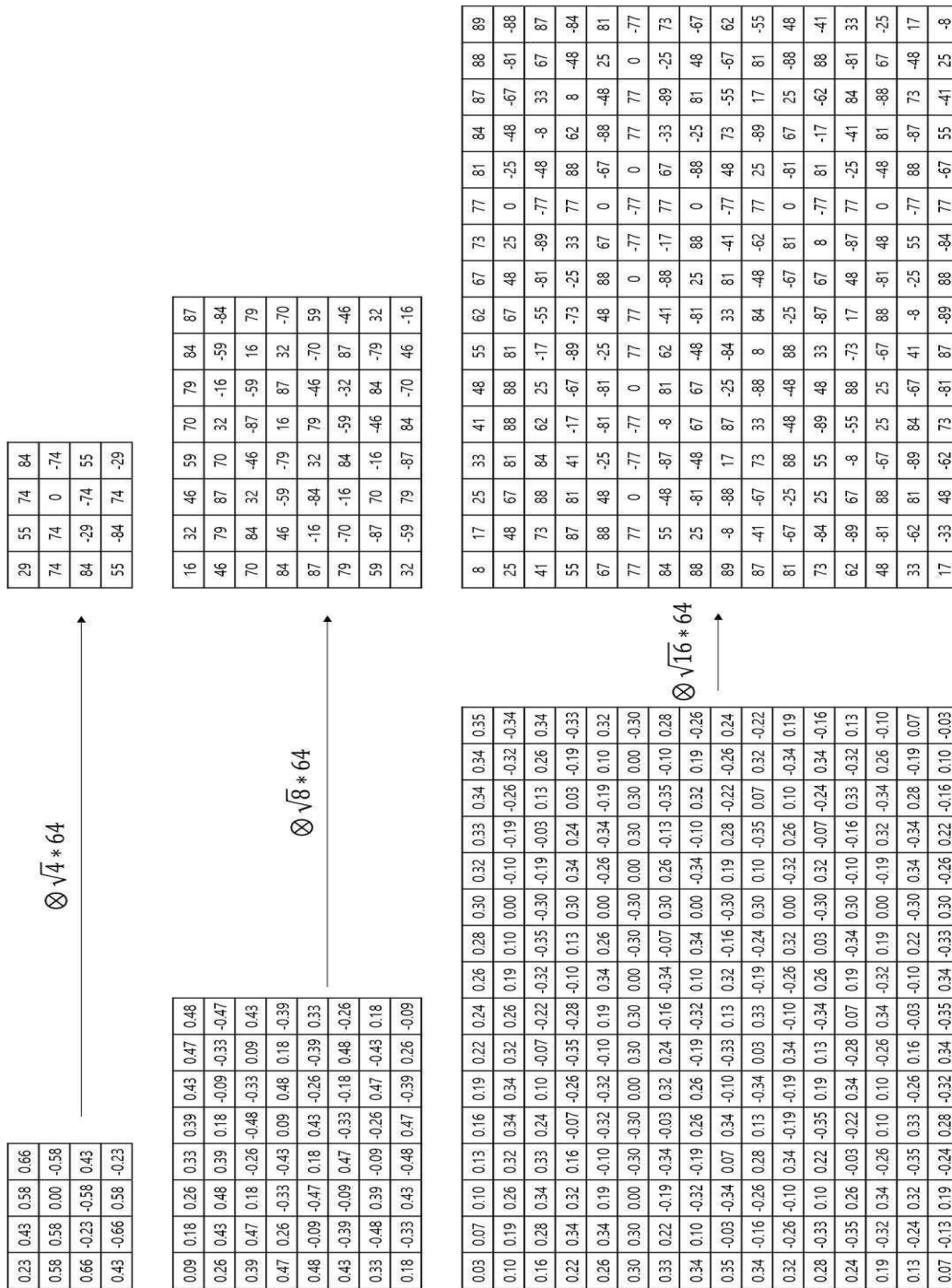
도면5



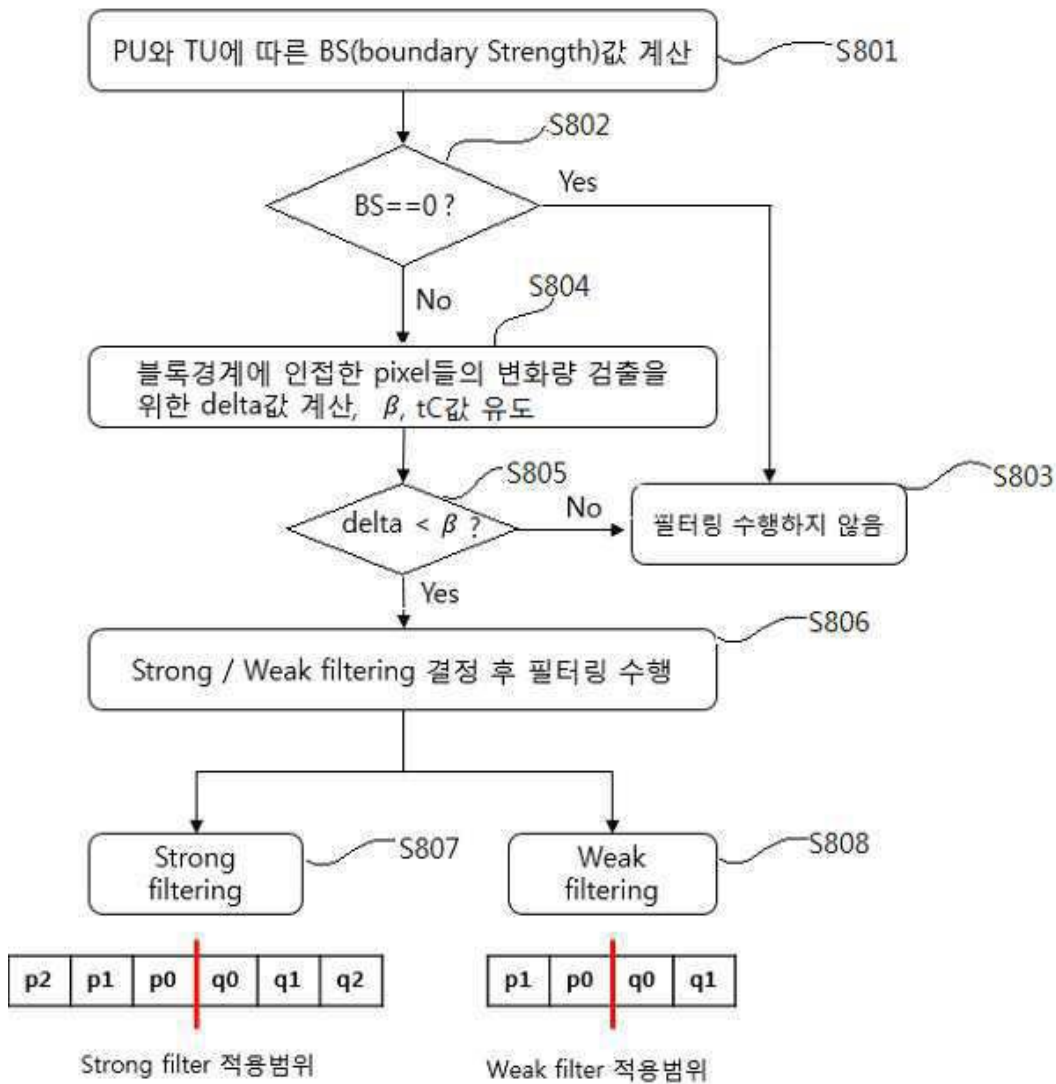
도면6



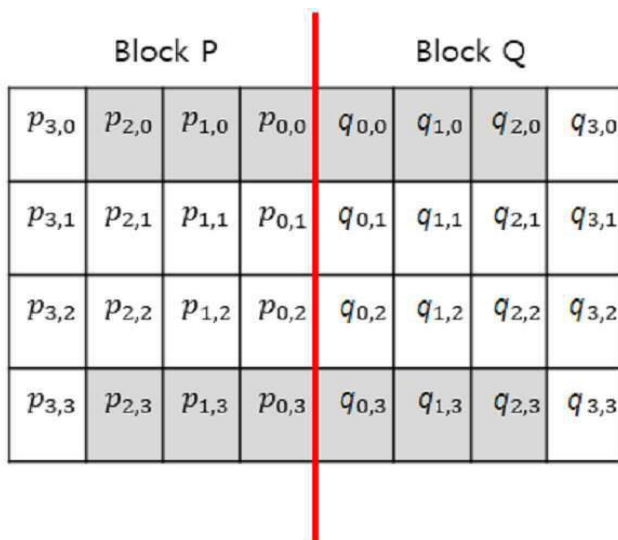
도면7



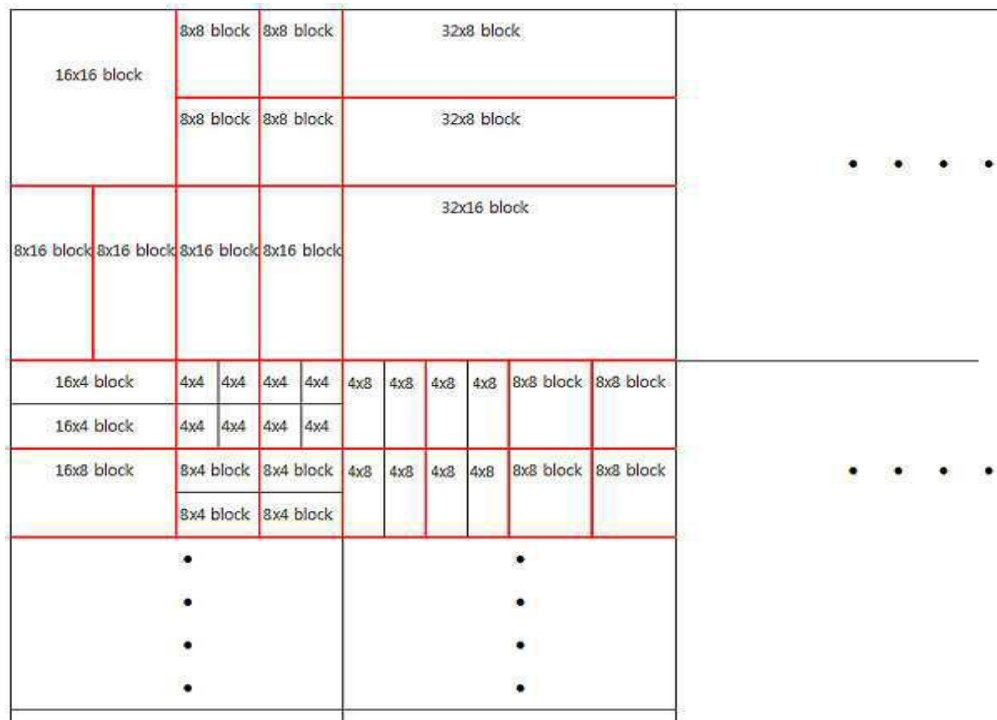
도면8



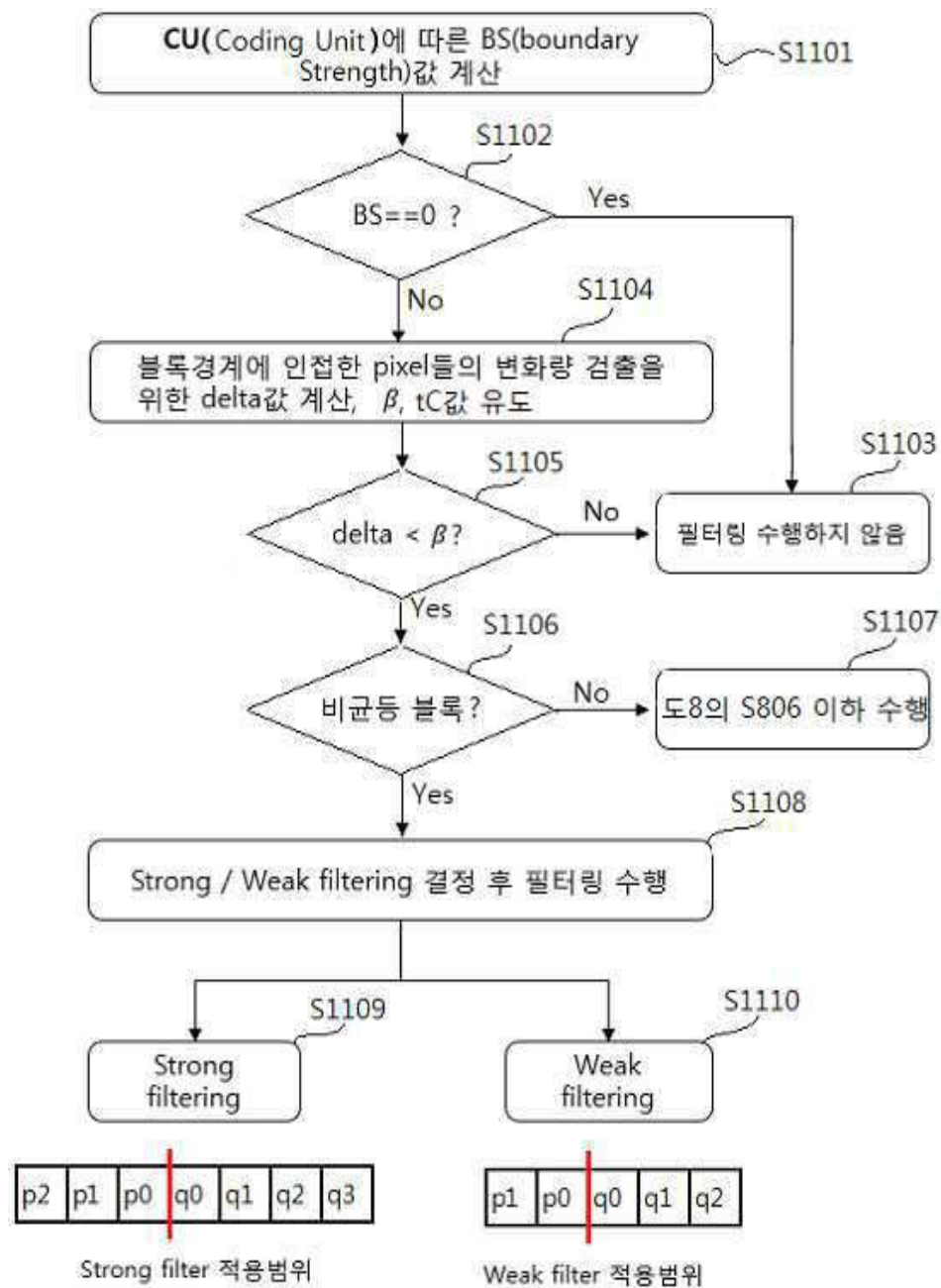
도면9



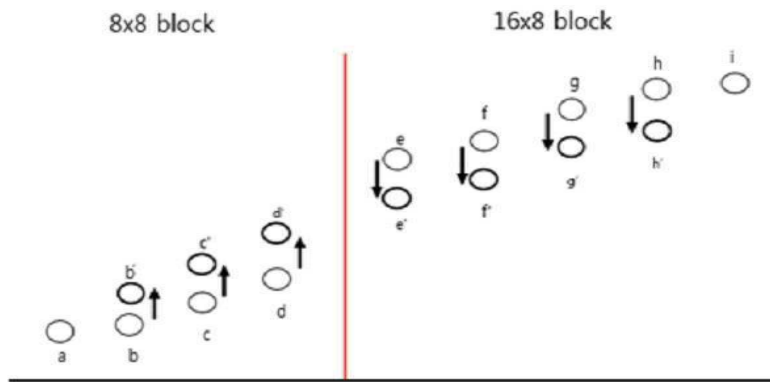
도면10



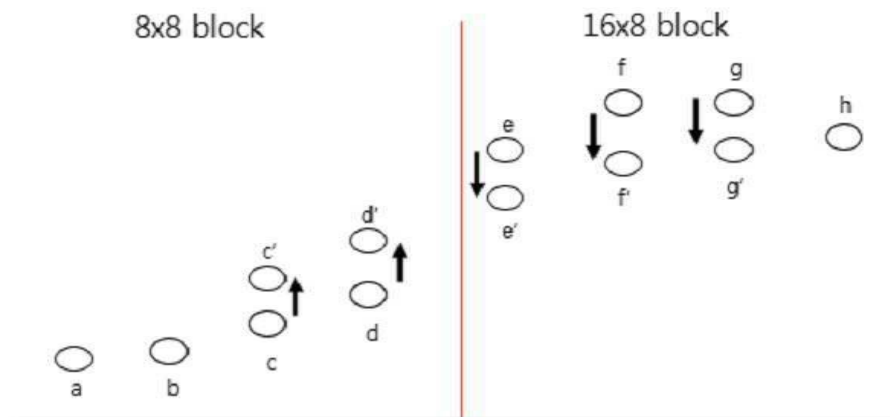
도면11



도면12



도면13



도면14

