(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2018년02월06일
(11) 등록번호 10-1826294
(24) 등록일자 2018년01월31일

(51) 국제특허분류(Int. Cl.)
H04N 19/13 (2014.01) *H04N 19/174* (2014.01)
H04N 19/176 (2014.01) *H04N 19/25* (2014.01)
H04N 19/436 (2014.01) *H04N 19/50* (2014.01)
H04N 19/51 (2014.01) *H04N 19/61* (2014.01)
H04N 19/91 (2014.01)
(52) CPC특허분류
H04N 19/13 (2015.01)
H04N 19/174 (2015.01)
(21) 출원번호 10-2015-7008104(분할)
(22) 출원일자(국제) 2012년02월23일
심사청구일자 2017년02월23일
(85) 번역문제출일자 2015년03월30일
(65) 공개번호 10-2015-0042295
(43) 공개일자 2015년04월20일
(62) 원출원 특허 10-2013-7026437
원출원일자(국제) 2012년02월23일
심사청구일자 2015년03월18일
(86) 국제출원번호 PCT/FR2012/050380
(87) 국제공개번호 WO 2012/120217
국제공개일자 2012년09월13일
(30) 우선권주장
11/51,849 2011년03월07일 프랑스(FR)

(56) 선행기술조사문헌
Y-W Huang, et al. ITU-T SG 16 Q 6 Video Coding Experts Group 38th Meeting. VCEG-AL25 rev.1, Jul. 5, 2009, pp.1-10*

(뒷면에 계속)

(73) 특허권자
돌비 인터네셔널 에이비
네덜란드 1101 씨엔 암스트레담 주이두스트 헤리 커베르그벡 1-35 3이 아폴로 빌딩
(72) 발명자
헨리 펠릭스
프랑스 에프-35700 렌 스퀘어 그란데 샤르보니에르 11
패테우스 스티븐
프랑스 에프-35700 렌 뤼 드 사르저우 102
(74) 대리인
김태홍, 김진회

전체 청구항 수 : 총 2 항 심사관 : 이상래

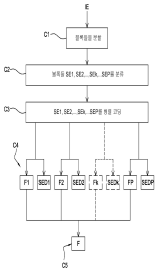
(54) 발명의 명칭 **이미지들을 코딩 및 디코딩하는 방법, 코딩 및 디코딩 디바이스 및 그에 대응하는 컴퓨터 프로그램들**

(57) 요 약

본 발명은 이미지를 다수의 블록들(MB)로 분할하는 단계, 상기 블록들을 미리 결정된 수의 블록들의 서브셋들로 그룹화하는 단계, 블록들의 상기 서브셋들의 각각을 병렬로 코딩하는 단계 — 고려된 서브셋의 블록들은 트래버설(traversal)의 미리 결정된 순차적 순서에 따라 코딩됨 — 를 포함하는 적어도 하나의 이미지를 코딩하는 방법

(뒷면에 계속)

대 표 도



에 관한 것이다. 코딩 단계는, 고려된 서브셋의 현재 블록에 대하여, 적어도 하나의 이전에 코딩 및 디코딩된 블록에 대하여 상기 현재 블록을 예측 코딩하는 서브-단계(C341), 및 적어도 하나의 심벌의 발생의 확률에 기반하여 상기 현재 블록을 엔트로피 코딩하는 서브-단계(C345)를 포함한다. 현재 블록이 고려된 서브셋의 코딩될 첫 번째 블록인 경우, 심벌의 발생의 상기 확률은 적어도 하나의 다른 서브셋의 코딩 및 디코딩된 미리 결정된 블록에 대하여 연산되었던 것이다. 현재 블록이 서브셋의 상기 첫 번째 블록 이외에 고려된 상기 서브셋의 블록인 경우, 심벌의 발생의 상기 확률은 상기 동일 서브셋에 속하는 적어도 하나의 코딩 및 디코딩된 블록에 대하여 연산되었던 것이다.

(52) CPC특허분류

H04N 19/176 (2015.01)
H04N 19/25 (2015.01)
H04N 19/436 (2015.01)
H04N 19/50 (2015.01)
H04N 19/51 (2015.01)
H04N 19/61 (2015.01)
H04N 19/91 (2015.01)

(56) 선행기술조사문헌

D. Marpe, et al. Context-based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard. IEEE Trans. on CSVT. Jul. 2003, Vol.13, No.7, pp.620-636*

V. Sze, et al. ITU-T SG 16 Q 6 Video Coding Experts Group 38th Meeting. VCEG-AL21, Jul. 3, 2009, pp.1-10

WO2010063184 A1

*는 심사관에 의하여 인용된 문헌

명 세 서

청구범위

청구항 1

하나 이상의 컴퓨터에 의해 실행가능한 명령어들을 포함하는 소프트웨어를 저장한 컴퓨터 판독가능한 저장 매체에 있어서, 상기 명령어들은 이러한 실행시 상기 하나 이상의 컴퓨터로 하여금,

디코더에 의해, 적어도 하나의 코딩된 이미지를 나타내는 스트림을 수신하는 동작;

상기 스트림으로부터 상기 적어도 하나의 코딩된 이미지를 나타내는 복수의 행들(rows)의 블록들을 식별하는 동작;

상기 디코더에 의해, 제1 행의 제1 블록을 엔트로피 디코딩하는 동작으로서, 상기 제1 블록은 상기 제1 행의 블록 순서에서 첫 번째이지 않으며, 상기 제1 블록의 엔트로피 디코딩은 상기 제1 블록의 바로 왼쪽에 위치하는 블록의 엔트로피 디코딩에 의해 결정된 제1 확률 데이터 세트에 기초하는 것인, 상기 제1 행의 제1 블록을 엔트로피 디코딩하는 동작; 및

상기 디코더에 의해, 제2 행의 제2 블록을 엔트로피 디코딩하는 동작으로서, 상기 제2 블록의 엔트로피 디코딩은 상기 제1 행의 미리결정된 블록의 엔트로피 디코딩에 의해 결정된 제2 확률 데이터 세트에 기초하고, 상기 제2 블록은 상기 제2 행의 블록 순서에서 첫 번째이며, 상기 미리결정된 블록은 상기 제1 행의 블록 순서에서 두 번째이고, 상기 제2 행은 상기 제1 행 아래에 위치한 것인, 상기 제2 행의 제2 블록을 엔트로피 디코딩하는 동작

을 포함하는 동작들을 수행하게 하는 것인, 컴퓨터 판독가능한 저장 매체.

청구항 2

청구항 1에 있어서, 상기 동작들은,

상기 미리결정된 블록의 엔트로피 디코딩 후에, 메모리에 상기 제2 확률 데이터 세트를 저장하는 동작; 및

상기 제2 행의 제2 블록을 엔트로피 디코딩하기 전에, 상기 메모리로부터 상기 제2 확률 데이터 세트를 검색(retrieve)하는 동작

을 더 포함하는 것인, 컴퓨터 판독가능한 저장 매체.

청구항 3

삭제

발명의 설명

기 술 분 야

[0001] 본 발명은 일반적으로 이미지 프로세싱의 분야에 관한 것으로, 보다 정확하게는 디지털 이미지들 및 디지털 이미지들의 시퀀스들의 코딩 및 디코딩에 관한 것이다.

[0002] 따라서, 본 발명은 특히 현재의 비디오 코더들(MPEG, H.264 등) 또는 근간의 비디오 코더들(ITU-T/VCEG(H.265) 또는 ISO/MPEG(HVC))에서 구현되는 비디오 코딩에 적용될 수 있다.

배 경 기 술

[0003] 현재의 비디오 코더들(MPEG, H264 등)은 비디오 시퀀스의 블록-방식(blockwise) 표현을 이용한다. 이미지들은 매크로-블록들로 분할되고, 각 매크로-블록은 그 자체를 블록들로 분할되며, 각 블록, 또는 매크로-블록은 인트라(intra)-이미지 또는 인터(inter)-이미지 예측에 의해 코딩된다. 따라서, 당업자에 의해 알려진 움직임 보상의 도움으로 하나 또는 둘 이상의 코딩-디코딩된 기준 이미지들에 대하여, 어떤 이미지들은 공간 예측(인트라 예측)에 의해 코딩되는 반면, 다른 이미지들은 시간 예측(인터 예측)에 의해 코딩된다. 게다가, 각 블록에 대

하여, 예측에 의해 감소된 원래 블록에 대응하는 잔차 블록(residual block)이 코딩될 수 있다. 이런 블록의 계수들은 선택적인 변환(transformation) 후에 양자화되고, 그 다음 엔트로피 코더에 의해 코딩된다.

[0004] 인트라 예측 및 인터 예측은 이전에 코딩 및 디코딩되었던 어떤 블록들이 현재 블록을 예측하기 위해 디코더 또는 코더 중 어느 하나에서 사용될 수 있도록 이용가능한 것을 필요로 한다. 이런 예측 코딩의 개략적 예는 도 1a에 도시되는데, 여기서, 이미지 I_N는 블록들로 분할되고, 이런 이미지의 현재 블록 MB_i은 해칭된 화살표에 의해 표시된 바와 같은 이전에 코딩 및 디코딩된 미리 결정된 수의 3개의 블록들 MBr_1, MBr_2 및 MBr_3에 대하여 예측 코딩이 이루어진다. 상술한 3개의 블록들은 특히 현재 블록 MB_i의 좌측에 바로 위치된 블록 MBr_1, 및 현재 블록 MB_i의 바로 위 및 우측 위에 각각 위치된 2개의 블록들 MBr_2 및 MBr_3을 포함한다.

[0005] 엔트로피 코더가 여기서 더욱 특별한 관심대상이다. 엔트로피 코더는 정보의 도착의 순서로 정보를 인코딩한다. 이미지의 상단 좌측의 블록에서 시작하는 참조표시 PRS에 의해 도 1a에 도시된 바와 같이, 전형적으로 블록들의 행 단위(row-by-row) 트래버설(traversal)은 "래스터-주사(raster-scan)" 형태로 수행된다. 각 블록에 대하여 블록의 표현을 위한 필요한 정보의 다양한 항목들(블록의 유형, 예측의 모드, 잔차 계수(residual coefficient)들 등)은 엔트로피 코더로 순차적으로 보내진다.

[0006] 합리적인 복잡성의 효율적인 산술 코더(arithmetical coder)는 AVC 압축 표준(또한 ISO-MPEG4 part 10 및 ITU-T H.264로 알려짐)으로 도입된 소위 "CABAC"("Context Adaptive Binary Arithmetic Coder")로 이미 알려져 있다.

[0007] 이런 엔트로피 코더는 다양한 개념들을 구현한다:

[0008] - 산술 코딩 : 문서 J. Rissanen and G. G. Langdon Jr, "Universal modeling and coding," IEEE Trans. Inform. Theory, vol. IT-27, pp. 12-23, Jan. 1981에 처음으로 설명된 바와 같은 코더는 심벌을 코딩하기 위해 이런 심벌의 발생의 확률을 사용한다;

[0009] - 콘텍스트(context)에 적응 : 이는 코딩될 심벌들의 발생의 확률의 적응을 수반한다. 한편, 온 더 플라이(on-the-fly) 학습이 수행된다. 다른 한편, 이전 코딩된 정보의 상태에 따라 특정 콘텍스트가 코딩을 위해 사용된다. 각 콘텍스트에 대하여, 심벌의 발생의 고유(inherent) 확률이 대응되어 있다. 예를 들면, 콘텍스트는 주어진 구성 또는 이웃의 상태(예를 들면, 이웃으로부터 선택된 "인트라" 모드들의 수 등)에 따른 코딩된 심벌의 유형(잔차의 계수의 표시 또는 코딩 모드의 시그널링(signaling) 등)에 대응한다;

[0010] 이진화(binarization) : 코딩될 심벌들은 비트들의 스트링(string)의 형태로 만들어진다. 그 후에, 이들 다양한 비트들은 연속하여 이진 엔트로피 코더로 전송된다.

[0011] 따라서, 이들 엔트로피 코더는, 사용된 각 콘텍스트에 대하여, 고려된(considered) 콘텍스트를 위해 이전에 코딩된 심벌들에 대하여 확률들을 온 더 플라이 학습하기 위한 시스템을 구현한다. 이런 학습은 이들 심벌들의 코딩의 순서에 기반한다. 전형적으로, 이미지는 위에서 설명된 "래스터-주사" 유형의 순서에 따라 트래버스된다(traversed).

[0012] 0 또는 1과 같을 수 있는 주어진 심벌 b의 코딩 동안, 이들 심벌의 발생의 확률들 P_i의 학습은 다음 방식에서 현재 블록 MB_i에 대하여 업데이트 된다 :

[0013]
$$p_i(b=0) = \alpha E p_{i-1}(b=0) + \begin{cases} (1-\alpha) & \text{코딩된 비트가 0이면} \\ 0 & \text{그외} \end{cases}$$

[0014] 여기서, α는 미리 결정된 값, 예를 들면, 0.95이고 P_{i-1}은 이런 심벌의 마지막 발생 동안 연산된 심벌 발생의 확률이다.

[0015] 이런 엔트로피 코더의 개략적 예가 도 1a에 도시되는데, 여기서 이미지 I_N의 현재 블록 MB_i는 엔트로피 코딩이 이루어진다. 블록 MB_i의 엔트로피 코팅이 시작될 때, 사용된 심벌 발생 확률들은 이전에 코딩 및 디코딩된 블록의 코딩 이후에 획득되는 것들이고, 이는 "래스터 주사" 유형의 블록들의 상술한 행 단위 트래버설에 따라 현재 블록 MB_i에 바로 선행하는 것이다. 블록 단위(block-by-block) 의존에 기반한 이런 학습은 어떤 블록들에 대하

여 단지 도면의 명료성을 위해, 얇은 선 화살표들로 도 1a에 표시된다.

[0016] 이런 유형의 엔트로피 코딩의 문제점은, 행의 시작에 위치된 심벌의 코딩 동안, 블록들의 "래스터 주사" 트래버설을 고려하면, 사용된 확률들이 주로 이전 행의 마지막에 위치된 심벌들에 대하여 관측된 것들에 대응한다는 사실에 있다. 이제, 심벌들의 확률들의 가능한 공간 변화 때문에(예를 들면, 움직임 정보의 항목과 관련된 심벌에 대하여, 이미지의 우측 부분에 위치된 움직임은 좌측 부분에서 관찰된 것과 상이할 수 있고, 따라서 그로부터 발생하는 국부(local) 확률들과 유사할 수 있음), 코딩 동안 효율성의 손실을 초래하는 위험이 있는 확률들의 국부 타당성(appropriateness)의 부족이 관찰될 수 있다.

[0017] 인터넷 주소 http://wftp3.itu.int/av-arch/jctvcsite/2010 04 A Dresden/JCTVC-A114-AnnexA.doc (on 8 February 2011)에서 입수할 수 있는 문서 "Annex A: CDCM Video Codec Decoder Specification"는 상술한 문제점을 완화시키는 코딩 방법을 설명한다. 위의 문서에서 설명된 코딩 방법은, 도 1b에 도시된 바와 같이,

[0018] - 이미지 IN을 다수의 블록들로 분할하는 단계,

[0019] - 해칭된(hatched) 화살표들에 의해 표시된 바와 같은, 이전에 코딩 및 디코딩된 미리 결정된 수의 3개의 블록들 MBr_1, MBr_2, 및 MBr_3에 대하여 이들 이미지의 현재 블록 MB_i를 예측 코딩하는 단계. 상술한 3개의 블록들은 특히 현재 블록 MB_i의 좌측에 바로 위치된 블록 MBr_1, 및 현재 블록 MB_i의 바로 위 및 우측 위에 각각 위치된 2개의 블록들 MBr_2 및 MBr_3을 포함함,

[0020] - 각 블록이 현재 블록의 바로 위에 위치되는 코딩 및 디코딩된 블록에 대하여 및 현재 블록의 바로 좌측에 위치되는 코딩 및 디코딩된 블록에 대하여, 이들 블록들이 이용가능할 때, 각각 연산된 심벌 발생의 확률들을 사용하는 것에 따라 이미지 I_N의 블록들을 엔트로피 코딩하는 단계. 심벌 발생의 확률들의 이런 이용은 도 1b의 명료성을 위해 얇은 선 화살표들로 도 1b에 부분적으로 표시된다.

[0021] 이런 엔트로피 코딩의 장점은 그것이 현재 블록의 당면한 환경에서 발생하는 확률들을 이용하고 그에 의해 더 높은 코딩 성능을 달성하는 것을 가능하게 하는 것이다. 게다가, 사용된 코딩 기법은 미리 결정된 수의 쌍으로(pairwise) 이웃하는 블록들의 서브셋들을 병렬로 코딩하는 것을 가능하게 한다. 도 1b에 나타낸 예에서, 3개의 서브셋들 SE1, SE2 및 SE3은 병렬로 코딩되고, 각 서브셋은 이런 예에서 대시로 표시된 블록들의 행으로 구성된다. 물론, 이런 코딩 기법은, 현재 블록의 위 및 우측 위에 각각 위치된 블록들이 이용가능한 것을 필요로 한다.

[0022] 이런 병렬 코딩 기법의 문제점은 현재 블록의 바로 위에 위치된 블록에 대하여 연산된 심벌 발생의 확률에 접근하도록 허용하기 위해, 블록들의 열과 관련된 다량의 확률들을 저장하는 것이 필요하다는 것이다. 블록들의 두 번째 열 SE2가 예를 들면, 도 1b에서 고려되면, 이런 열의 첫 번째 블록은 이전 첫 번째 열 SE1의 첫 번째 블록에 대하여 연산된 심벌 발생의 확률들을 이용함으로써 엔트로피 코딩이 이루어진다. 두 번째 열의 첫 번째 블록의 코딩이 완료되면, 발생의 확률의 상태의 값 V1은 버퍼 메모리(MT)에 저장된다. 두 번째 열 SE 2의 두 번째 블록은 그 후에 첫 번째 열 SE1의 두 번째 블록 및 두 번째 열 SE2의 첫 번째 블록에 대하여 한번에 및 동시에 연산된 심벌 발생의 확률들을 이용함으로써 엔트로피 코딩이 이루어진다. 두 번째 열의 첫 번째 블록의 코딩이 완료되면, 발생의 확률의 상태의 값 V2가 버퍼 메모리(MT)에 저장된다. 이런 절차는 두 번째 열 SE2의 마지막 블록까지 수행된다. 확률들의 양이 매우 크기 때문에(구문 성분(syntax element)의 수와, 관련된 콘텍스트의 수의 조합만큼 많은 확률들이 존재함), 전체 열에 걸친 이들 확률들의 저장은 메모리 자원 면에서 고가이다.

발명의 내용

[0023] 본 발명의 목적들 중 하나는 상술한 종래 기술들의 문제점들을 해결하는 것이다.

[0024] 이를 위해, 본 발명의 주제는 적어도 하나의 이미지를 코딩하는 방법에 관한 것이며, 상기 방법은

[0025] - 상기 이미지를 다수의 블록들로 분할하는 단계,

[0026] - 블록들을 미리 결정된 수의 블록들의 서브셋들로 그룹화하는 단계,

[0027] - 블록들의 상기 서브셋들의 각각을 병렬로 코딩하는 단계 - 고려된 서브셋의 상기 블록들은 트래버설의 미리 결정된 순차적 순서에 따라 코딩됨 - 포함하고,

[0028] 상기 코딩 단계는, 고려된 서브셋의 현재 블록에 대하여,

[0029] ● 적어도 하나의 이전에 코딩 및 디코딩된 블록에 대하여 상기 현재 블록을 예측 코딩하는 서브-단계,

[0030] ● 적어도 하나의 심벌 발생의 확률에 기반하여 상기 현재 블록을 엔트로피 코딩하는 서브-단계(C345)를 포함한다.

[0031] 본 발명에 따른 방법은 다음에 주목할 만하다:

[0032] - 상기 현재 블록이 고려된 서브셋의 코딩될 첫 번째 블록인 경우, 상기 심벌 발생의 확률은 적어도 하나의 다른 서브셋의 코딩 및 디코딩된 미리 결정된 블록에 대하여 연산되었던 것이고,

[0033] - 상기 현재 블록이 상기 첫 번째 블록 이외에 고려된 상기 서브셋의 블록인 경우, 상기 심벌 발생의 확률은 상기 동일 서브셋에 속하는 적어도 하나의 코딩 및 디코딩된 블록에 대하여 연산되었던 것이다.

[0034] 그러한 배열은, 블록들의 서브셋의 첫 번째 블록 이외의 현재 블록의 엔트로피 코딩이 다른 서브셋에서 현재 블록 위에 위치되는 이전에 코딩 및 디코딩된 블록에 대하여 연산된 심벌 발생의 확률들의 사용을 더 이상 반드시 필요로 하지 않기 때문에, 코더의 버퍼 메모리에 훨씬 적은 양의 심벌들의 발생의 확률들을 저장하는 것을 가능하게 한다.

[0035] 그러한 배열은 게다가 현재 블록의 엔트로피 코딩이 현재 블록에 속하는 또 다른 이전에 코딩 및 디코딩된 서브셋의 블록에 대하여 연산된 심벌 발생의 확률들을 이용하고, 결과적으로 확률들이 비디오 신호의 통계에 따르도록 학습이 확률들의 갱신을 통하여 이미 수행되었기 때문에, 기존의 압축 성능을 유지하는 것을 가능하게 한다.

[0036] 고려된 블록들의 서브셋의 첫 번째 현재 블록을 엔트로피 코딩하는 동안 상기 다른 서브셋의 첫 번째 블록에 대하여 연산된 심벌 발생의 확률들을 이용하는 주요한 장점은, 상기 다른 서브셋의 다른 연속하는 블록들에 의해 학습된 심벌 발생 확률들을 고려하지 않고, 심벌들의 발생의 상기 확률들의 갱신만을 버퍼 메모리에 저장함으로써 코더의 버퍼 메모리를 절약하는 것이다.

[0037] 고려된 블록들의 서브셋의 첫 번째 현재 블록의 엔트로피 코딩 동안, 첫 번째 블록 이외에 상기 다른 서브셋의 블록, 예를 들면, 두 번째 블록에 대하여 연산된 심벌 발생 확률들의 사용의 주요한 장점은, 심벌들의 발생의 확률들을 더 정확하게 획득하고, 따라서 더 우수하게 학습하며, 그에 의해 더 우수한 비디오 압축 성능을 야기하는 것이다.

[0038] 특정 실시예에서, 상기 서브셋의 상기 첫 번째 블록 이외에 코딩될 상기 현재 블록과 동일한 서브셋에 속하는 상기 코딩 및 디코딩된 블록은 코딩될 상기 현재 블록의 가장 근접한 이웃인 것이다.

[0039] 따라서, 이런 방식은, 이런 특정 경우에, 첫 번째 현재 블록 위에 위치되고 다른 서브셋에 속하는 블록에 대하여 연산된 심벌 발생 확률만이 고려되기 때문에, 고려된 서브셋의 첫 번째 블록의 엔트로피 코딩 동안 학습된 심벌 발생 확률들만 저장하는 것을 가능하게 한다. 이는 코더의 메모리 자원들의 크기의 감소의 최적화를 야기한다.

[0040] 다른 특정 실시예에서, 고려된 서브셋의 블록의 예측 코딩이 고려된 상기 서브셋 이외에 미리 결정된 수의 이전에 코딩 및 디코딩된 서브셋의 블록들에 대하여 수행되도록 의도되는 경우, 고려된 상기 서브셋의 블록들의 병렬 코딩은 상기 병렬 코딩이 수행되는 순서에서 바로 이전의 블록들의 서브셋에 대하여 상기 미리 결정된 수의 블록들만큼 시프트하여 수행된다.

[0041] 그러한 배열은, 예를 들면, 코딩될 블록들의 현재 서브셋에 대하여, 병렬 코딩이 수행되는 순서에서 현재 서브셋에 선행하는 블록들의 서브셋의 블록들의 처리의 진행의 동기화를 허용하고, 그에 의해 현재 블록의 코딩을 위해 사용되는 이전 서브셋의 블록 또는 블록들의 이용가능성(availability)을 보장하는 것을 가능하게 한다. 이런 방식으로, 종래 기술의 병렬 코더들에서 구현되는 바와 같은 이전 서브셋의 이런 또는 이들 블록들의 이용가능성을 확인하는 단계가 유리하게 생략될 수 있고, 그에 의해 본 발명에 따라 코더에서 블록들을 처리하기 위해 필요한 처리 시간의 가속을 허용한다.

[0042] 상관적으로, 본 발명은 적어도 하나의 이미지를 코딩하기 위한 디바이스에 관한 것이며, 상기 디바이스는,

[0043] - 상기 이미지를 다수의 블록들로 분할하기 위한 수단,

[0044] - 상기 블록들을 미리 결정된 수의 블록들의 서브셋들로 그룹화하기 위한 수단,

[0045] - 블록들의 상기 서브셋들의 각각을 병렬 코딩하는 수단 - 고려된 서브셋의 상기 블록들은 트래버설의 미리 결정된 순차적 순서에 따라 코딩됨 -을 포함하고,

[0046] 상기 코딩 수단은, 고려된 서브셋의 현재 블록에 대하여,

[0047] ● 적어도 하나의 이전에 코딩 및 디코딩된 블록에 대하여 상기 현재 블록을 예측 코딩하는 서브-수단,

[0048] ● 적어도 하나의 심벌 발생의 확률에 기반하여 상기 현재 블록을 엔트로피 코딩하는 서브-수단을 포함한다.

[0049] 이런 코딩 디바이스는 다음에 주목할 만하다:

[0050] - 상기 현재 블록이 고려된 서브셋의 코딩될 첫 번째 블록인 경우, 상기 엔트로피 코딩하는 서브-수단은, 상기 첫 번째 현재 블록을 엔트로피 코딩하기 위해, 적어도 하나의 다른 서브셋의 코딩 및 디코딩된 미리 결정된 블록에 대하여 연산되었던 심벌 발생의 확률을 고려하고,

[0051] - 상기 현재 블록이 고려된 상기 서브셋의 상기 첫 번째 블록 이외에 고려된 상기 서브셋의 블록인 경우, 상기 엔트로피 코딩하는 서브-수단은, 상기 현재 블록을 엔트로피 코딩하기 위해, 상기 동일 서브셋에 속하는 적어도 하나의 코딩 및 디코딩된 블록에 대하여 연산되었던 심벌 발생의 확률을 고려한다.

[0052] 대응하는 방식으로, 본 발명은 또한 적어도 하나의 코딩된 이미지를 나타내는 스트림을 디코딩하는 방법에 관한 것이며, 상기 방법은,

[0053] - 미리 결정된 수의 디코딩될 블록들의 서브셋들의 상기 이미지를 식별하는 단계,

[0054] - 블록들의 상기 서브셋들의 각각과 관련되는 상기 스트림의 일부들을 병렬로 디코딩하는 단계 - 고려된 상기 서브셋의 블록들은 트래버설의 미리 결정된 순차적 순서에 따라 디코딩됨 -를 포함하고,

[0055] 상기 디코딩 단계는, 고려된 서브셋의 현재 블록에 대하여,

[0056] ● 적어도 하나의 심벌 발생의 확률에 기반하여 상기 현재 블록을 엔트로피 디코딩하는 서브-단계,

[0057] ● 적어도 하나의 이전에 디코딩된 블록에 대하여 상기 현재 블록을 예측 디코딩하는 서브-단계를 포함한다.

[0058] 이런 디코딩 방법은 다음에 주목할 만하다:

[0059] - 상기 현재 블록이 고려된 서브셋의 디코딩될 첫 번째 블록인 경우, 상기 심벌 발생의 확률은 적어도 하나의 다른 서브셋의 디코딩된 미리 결정된 블록에 대하여 연산되었던 것이고,

[0060] - 상기 현재 블록이 고려된 상기 서브셋의 상기 첫 번째 블록 이외에 고려된 상기 서브셋의 블록인 경우, 상기 심벌 발생의 확률은 상기 동일 서브셋에 속하는 적어도 하나의 디코딩된 블록에 대하여 연산되었던 것이다.

[0061] 특정 실시예에서, 상기 서브셋의 상기 첫 번째 블록 이외에 디코딩될 상기 현재 블록과 동일한 서브셋에 속하는 상기 디코딩된 블록은 디코딩될 상기 현재의 블록의 가장 근접한 이웃인 것이다.

[0062] 다른 특정 실시예에서, 고려된 서브셋의 블록의 예측 디코딩이 고려된 상기 서브셋 이외에 미리 결정된 수의 이전에 디코딩된 서브셋의 블록들에 대하여 수행되도록 의도되는 경우, 고려된 상기 서브셋의 블록들의 병렬 디코딩은 상기 병렬 디코딩이 수행되는 순서에서 바로 이전의 블록들의 서브셋에 대하여 상기 미리 결정된 수의 블록들만큼 시프트하여 수행된다.

[0063] 상관적으로, 본 발명은 또한 적어도 하나의 코딩된 이미지를 나타내는 스트림을 디코딩하기 위한 디바이스에 관한 것이며, 상기 디바이스는,

[0064] - 미리 결정된 수의 디코딩될 블록들의 서브셋들을 상기 이미지에서 식별하는 수단,

[0065] - 블록들의 상기 서브셋들의 각각과 관련되는 상기 스트림의 일부들을 병렬 디코딩하는 수단 - 고려된 상기 서브셋의 블록들은 트래버설의 미리 결정된 순차적 순서에 따라 디코딩됨 -을 포함하고,

[0066] 상기 디코딩 수단은, 고려된 서브셋의 현재 블록에 대하여,

[0067] ● 적어도 하나의 심벌 발생의 확률에 기반하여 상기 현재 블록을 엔트로피 디코딩하는 서브-수단,

[0068] ● 적어도 하나의 이전에 디코딩된 블록에 대하여 상기 현재 블록을 예측 디코딩하는 서브-수단을 포함한다.

[0069] 이런 디코딩 디바이스는 다음에 주목할 만하다:

[0070] - 상기 현재 블록이 고려된 서브셋의 디코딩될 첫 번째 블록인 경우, 엔트로피 디코딩하는 서브-수단은, 상기 첫 번째 현재 블록을 엔트로피 디코딩하기 위해, 적어도 하나의 다른 서브셋의 디코딩된 미리 결정된 블록에 대하여 연산되었던 심벌 발생의 확률을 고려하고,

[0071] - 상기 현재 블록이 고려된 상기 서브셋의 상기 첫 번째 블록 이외에 고려된 상기 서브셋의 블록인 경우, 상기 엔트로피 디코딩하는 서브-수단은, 상기 현재 블록을 엔트로피 디코딩하기 위해, 상기 동일 서브셋에 속하는 적어도 하나의 디코딩된 블록에 대하여 연산되었던 심벌 발생의 확률을 고려한다.

[0072] 본 발명은 또한 프로그램이 컴퓨터에 의해 실행되는 경우, 위에서 설명한 코딩 또는 디코딩 방법의 단계들의 실행을 위한 명령들을 포함하는 컴퓨터 프로그램을 목적으로 한다.

[0073] 이런 프로그램은 임의의 프로그래밍 언어를 사용할 수 있고, 소스 코드, 목적 코드(object code), 또는 부분적으로 컴파일된 형태와 같은 소스 코드와 목적 코드 사이의 중간 코드의 형태 또는 임의의 다른 바람직한 형태일 수 있다.

[0074] 본 발명의 또 다른 주제는 또한 컴퓨터에 의해 판독 가능하고 상술한 바와 같은 컴퓨터 프로그램 명령들을 포함하는 기록 매체를 목적으로 한다.

[0075] 기록 매체는 프로그램을 저장할 수 있는 임의의 엔티티 또는 디바이스일 수 있다. 예를 들면, 이런 매체는 ROM과 같은 저장 수단, 예를 들면, CD ROM 또는 마이크로 전자 회로 ROM, 또는 그 외의 자기 기록 수단, 예를 들면, 디스켓(플로피 디스크) 또는 하드 디스크를 포함할 수 있다.

[0076] 게다가, 이런 기록 매체는 전기 또는 광학 케이블을 통하여, 무선에 의해, 또는 다른 수단에 의해 전달될 수 있는 전기 또는 광학 신호와 같은 전달 가능한 매체일 수 있다. 본 발명에 따른 프로그램은 특히 인터넷 유형의 네트워크 상에서 다운로드될 수 있다.

[0077] 대안적으로, 이런 기록 매체는 프로그램이 그 안에 통합되는 집적 회로일 수 있고, 집적 회로는 해당 방법을 실행하기 위해 또는 방법의 실행에 사용되도록 적응된다.

[0078] 상술한 코딩 디바이스, 디코딩 방법, 디코딩 디바이스 및 컴퓨터 프로그램들은 본 발명에 따른 코딩 방법에 의해 부여되는 것들과 적어도 동일한 장점을 제시한다.

[0079] 다른 특징들 및 장점들은 도면들을 참조하여 설명된 2개의 바람직한 실시예들을 읽을 때 명백하게 될 것이다.

도면의 간단한 설명

[0080] - 도 1a는 제 1 예에 따른 종래 기술의 이미지 코딩 다이어그램을 도시하고,

- 도 1b는 제 2 예에 따른 종래 기술의 이미지 코딩 다이어그램을 도시하며,

- 도 2a는 본 발명에 따른 코딩 방법의 주요 단계들을 도시하고,

- 도 2b는 도 2a의 코딩 방법에서 구현되는 병렬 코딩을 상세하게 도시하며,

- 도 3a는 본 발명에 따른 코딩 디바이스의 실시예를 도시하고,

- 도 3b는 도 3a의 코딩 디바이스에 대한 코딩 유닛을 도시하며,

- 도 4a는 제 1 바람직한 실시예에 따른 이미지 코딩/디코딩 다이어그램을 도시하고,

- 도 4b는 제 2 바람직한 실시예에 따른 이미지 코딩/디코딩 다이어그램을 도시하며,

- 도 5a는 본 발명에 따른 디코딩 방법의 주요 단계들을 도시하고,

- 도 5b는 도 5a의 디코딩 방법에서 구현되는 병렬 디코딩을 상세하게 도시하며,

- 도 6a는 본 발명에 따른 디코딩 디바이스의 실시예를 도시하고,

- 도 6b는 도 6a의 디코딩 디바이스의 디코딩 유닛을 도시한다.

발명을 실시하기 위한 구체적인 내용

[0081] <u>코딩 부분의 실시예의 상세한 설명</u>

[0082] 본 발명의 실시예가 이제 설명될 것이고, 여기서, 본 발명에 따른 코딩 방법은 H.264/MPEG-4 AVC 표준에 따른 코딩에 의해 획득되는 것과 근사한 이진 스트림에 따른 이미지들의 시퀀스를 코딩하도록 사용된다. 이런 실시예에서, 본 발명에 따른 코딩 방법은 예를 들면, 초기에 H.264/MPEG-4 AVC 표준에 따른 코더의 수정들에 의해 소프트웨어 또는 하드웨어 방식으로 구현된다. 본 발명에 따른 코딩 방법은 도 2a에 도시된 단계들(C1 내지 C5)을 포함하는 알고리즘의 형태로 도시된다.

[0083] 본 발명의 실시예에 따르면, 본 발명에 따른 코딩 방법은 도 3a에 도시된 코딩 디바이스(CO)에서 구현된다.

[0084] 도 2a를 참조하여, 제 1 코딩 단계(C1)는, 도 4a 또는 도 4b에 도시된 바와 같이, 코딩될 이미지들의 시퀀스의 이미지(IE)의 다수의 블록들 또는 매크로-블록들(MB)로 분할하는 단계이다. 도시된 예들에서, 상기 블록들(MB)은 정사각형 형상을 갖고 모두가 동일한 크기를 갖는다. 반드시 블록들의 크기의 배수가 아닌 이미지의 크기의 함수로서, 좌측 상의 마지막 블록들 및 하부에서 마지막 블록들은 정사각형이 아닐 수 있다. 대안적인 실시예에서, 블록들은 예를 들면, 직사각형 크기 및/또는 서로 정렬되지 않을 수 있다.

[0085] 각 블록 또는 매크로블록은 게다가 그 자체로 그것들 자체가 세분될 수 있는(subdividable) 서브-블록들로 분할될 수 있다.

[0086] 이런 분할은 예를 들면, 이와 같이 알려진 파티션 알고리즘을 사용하는 도 3a에 도시된 파티션(partition) 모듈(PCO)에 의해 수행된다.

[0087] 도 2a를 참조하여, 제 2 코딩 단계(C2)는 상술한 블록들을 병렬로 코딩되도록 의도된 미리 결정된 수 P개의 연속하는 블록들의 서브셋들 SE1, SE2,...,SEk,...SEP로 그룹화하는 단계이다. 도 4a 및 도 4b에 도시된 예들에서, 미리 결정된 수 P개는 4와 같고 대시로 도시된 4개의 서브셋들 SE1, SE2, SE3, SE4는 각각 이미지 IE의 첫 번째 4개의 블록들의 열들로 구성된다.

[0088] 이런 그룹화는 그 자체로 잘 알려진 알고리즘의 도움으로 도 3a에 도시된 연산 모듈(GRCO)에 의해 수행된다.

[0089] 도 2a를 참조하여, 제 3 코딩 단계(C3)는 블록들의 상기 서브셋들 SE1, SE2, SE3 및 SE4의 각각을 병렬로 코딩하는 단계에 있고, 고려된 서브셋의 블록들은 트래버설 PS의 미리 결정된 순차적 순서에 따라 코딩된다. 도 4a 및 도 4b에 도시된 예들에서, 현재 서브셋 SEk($1 \le k \le 4$)의 블록들은 화살표 PS에 의해 지시된 바와 같이, 좌측에서 우측으로 차례로 코딩된다.

[0090] 이런 병렬 코딩은 도 3a에 도시된 바와 같이, R=4인 다수의 R개의 코딩 유닛들(UCk($1 \le k \le R$))에 의해 구현되고, 코딩 방법의 상당한 가속을 허용한다. 이와 같이 알려진 방법으로, 코더(CO)는 현재 블록의 코딩과 협력하여 계속적으로 갱신되는 바와 같은 심벌 발생 확률들을 포함하도록 적응되는 버퍼 메모리(MT)를 포함한다.

[0091] 도 3b에 더 상세하게 도시된 바와 같이, 코딩 유닛들(UCk)의 각각은,

[0092] ● SUCPk로 지시된, 적어도 하나의 이전 코딩 및 디코딩된 블록에 대하여 현재 블록을 예측 코딩하기 위한 서브-유닛;

[0093] ● SUCEk로 지시된, 상기 이전 코딩 및 디코딩된 블록들에 대하여 연산된 적어도 하나의 심벌 발생의 확률을 이용함으로써 상기 현재 블록을 엔트로피 코딩하기 위한 서브-유닛을 포함한다.

[0094] 예측 코딩 서브-유닛(SUCPk)은 예를 들면, 인트라 및/또는 인터 모드에서와 같은 종래의 예측 기법들에 따라 현재 블록의 예측 코딩을 수행할 수 있다.

[0095] 엔트로피 코딩 서브-유닛(SUCEk)은 CABAC 유형의 그 일부이지만, 명세서(description)에서 더 설명될 것과 같이, 본 발명에 따라 수정된다.

[0096] 변형예로서, 엔트로피 코더 서브-유닛(SUCEk)은 이와 같이 알려진 허프만(Huffman) 코더일 수 있다.

[0097] 도 4a 및 도 4b에 도시된 예들에서, 제 1 유닛(UC1)은 첫 번째 열 SE1의 블록들을 좌측에서 우측으로 코딩한다. 그것이 첫 번째 열 SE1의 마지막 블록에 도달하면, 그것은 (N+1) 번째 열, 여기서 5 번째 열 등의 첫 번째 블록으로 이동한다. 제 2 유닛(UC2)은 두 번째 열 SE2의 블록들을 좌측에서 우측으로 코딩한다. 그것이 두 번째 열 SE2의 마지막 블록에 도달하면, 그것은 (N+2) 번째 열, 여기서 6 번째 열 등의 첫 번째 블록으로 이동한다. 이런 트래버설은 네 번째 열 SE4의 블록들을 좌측에서 우측으로 코딩하는 유닛(UC4)까지 반복된다. 그것이 첫 번째 열의 마지막 블록에 도달될 때, 이미지 IE의 마지막 블록이 코딩될 때까지, 그것은 (N+4) 번째 열, 여기서 8 번째 열 등등의 첫 번째 블록으로 이동한다.

[0098] 위에서 방금 설명되었던 것과 다른 유형들의 트래버설도 물론 가능하다. 따라서, 이미지 IE를 다수의 서브-이미지들로 분할하고, 이런 유형의 분할을 각각의 서브-이미지에 독립적으로 적용하는 것이 가능하다. 각 코딩 유닛이, 위에서 설명된 바와 같이, 네스팅된(nested) 열들이 아니라 네스팅된 행들을 처리하는 것이 또한 가능하다. 열들 또는 행들 중 어느 하나의 방향으로 트래버스하는 것이 또한 가능하다.

[0099] 도 2a를 참조하여, 제 4 코딩 단계(C4)는 상술한 코딩 유닛들의 각각에 의해 압축된 처리된 블록들을 나타내는 N 개의 서브-비트스트림들 Fk(1≤k≤N) 뿐만 아니라 각각의 서브셋 SEk의 처리된 블록들의 디코딩된 버전의 생성이다. SED1, SED2,...,SEDk,...,SEDP로 표시된 고려된 서브셋의 디코딩된 처리된 블록들은 명세서에서 더 상세하게 될 동기화 메커니즘에 따라 도 3a에 도시된 코딩 유닛들(UC1, UC2,..., UCk,..., UCP) 중 일부에 의해 재사용될 수 있다.

[0100] 도 2a를 참조하여, 제 5 코딩 단계(C5)는 글로벌(global) 스트림 F를 상술한 서브-스트림들 Fk에 기반하여 구성하는 단계에 있다. 일 실시예에 따르면, 서브-스트림 Fk는 글로벌 스트림 F에서 각각의 서브-스트림 Fk의 위치를 디코더로 나타내도록 의도된 추가 정보 항목과 단지 병렬(juxtraposed)된다. 글로벌 스트림은 통신 네트워크(미도시)에 의해 원격 단말로 전송된다. 원격 단말은 도 6a에 도시된 디코더(DO)를 포함한다.

[0101] 따라서, 명세서에서 더 상세하게 설명될 것과 같이, 본 발명에 따른 디코더는 글로벌 스트림 F 내의 서브-스트림들 Fk를 분리시키고 디코더의 각각의 컴포넌트 디코딩 유닛으로 그것들을 할당할 수 있다. 서브-스트림들의 글로벌 스트림으로의 이런 분해는 병렬로 동작하는 몇몇 코딩 유닛들의 사용의 선택과 독립적이고, 이런 접근법에 의해 병렬로 동작하는 유닛들을 포함하는 코더만 또는 디코더만을 갖는 것이 가능한 것에 유의할 것이다.

[0102] 글로벌 스트림 F의 이런 구성은 도 3a에 도시된 바와 같은 스트림 구성 모듈(CF)에서 구현된다.

[0103] 상술한 병렬 코딩 단계(C3) 동안, 코딩 유닛(UCk)에서 구현되는 바와 같은 본 발명의 다양한 특정 서브-단계들은 도 2b를 참조하여 이제 설명될 것이다.

[0104] 단계 C31의 과정에서, 코딩 유닛(UCk)은 도 4a 또는 도 4b에 도시된 현재 열 SEk의 코딩될 첫 번째 블록을 현재 블록으로서 선택한다.

[0105] 단계 C32의 과정에서, 유닛(UCk)은 현재 블록이 상술한 단계 C1에서 블록들로 분할되었던 이미지 IE의 첫 번째 블록(상부에서 좌측에 위치된)인지의 여부를 검사한다.

[0106] 그것이 사실이면, 단계 C33의 과정에서, 코딩 확률들이 도 3a의 코더(CO)에서 이전에 정의된 값들 Pinit로 초기화된다.

[0107] 그것이 사실이 아니면, 다음의 명세서에서 나중에 설명될 단계 C40의 과정에서, 필요한 이전에 코딩 및 디코딩된 블록들의 이용가능성의 결정이 착수된다.

[0108] 단계 C34의 과정에서, 도 4a 또는 도 4b에 도시된 첫 번째 열 SE1의 첫 번째 현재 블록 MB1의 코딩이 착수된다. 이런 단계 C34는 이하에 설명될 다수의 서브-단계들 C341 내지 C348을 포함한다.

[0109] 첫 번째 서브-단계 C341의 과정에서, 블록 MB1이 적어도 하나의 이전에 코딩 및 디코딩된 블록들에 대하여 예측되는 과정에서, 현재 블록 MB1의 예측 코딩이 인트라 및/또는 인터 예측의 알려진 기법들에 의해 착수된다.

[0110] H.264 표준에서 제안된 바와 같은 인트라 예측의 다른 모드들이 가능함은 말할 필요도 없다.

[0111] 현재 블록 MB1에는, 현재 블록이 이전에 코딩 및 디코딩된 이미지로부터 발생된 블록에 대하여 예측되는 과정에서, 또한 인터 모드로 예측 코딩이 이루어질 수 있다. 다른 유형들의 예측이 물론 구상 가능하다. 현재 블록에 대하여 가능한 예측들 중에서, 최적 예측이 당업자에게 잘 알려진 비율 왜곡(rate distortion) 기준에 따라 선택된다.

[0112] 상기 상술한 예측 코딩 단계는 현재 블록 MB_1의 근삿값인 예측된 블록 MBp_1을 구성하는 것을 가능하게 한다. 이런 예측 코딩과 관련된 정보는 나중에 디코더(DO)로 전송되는 스트림 F 안에 기록될 것이다. 이런 정보는 특히 예측의 유형(인터 또는 인트라), 및 적합하면, 인트라 예측의 모드, 블록 또는 매크로블록이 세분되었던 경우, 블록 또는 매크로블록의 파티션의 유형, 기준 이미지 인덱스 및 인터 예측 모드에서 사용된 변위 벡터를 포함한다. 이런 정보는 코더(CO)에 의해 압축된다.

[0113] 다음의 서브-단계 C342의 과정에서, 잔차 블록 MBr_1을 생성하기 위해 현재 블록 MB_1에서 예측된 블록 MBp_1의 감산이 착수된다.

[0114] 다음의 서브-단계 C343의 과정에서, 변환된 블록 MBt_1을 생성하기 위해, 예를 들면, DCT 유형의 이산 코사인 변환과 같은 직접 변환의 종래의 연산에 따라 잔차 블록 MBr_1의 변환이 착수된다.

[0115] 다음의 서브-단계 C344의 과정에서, 예를 들면, 스칼라 양자화와 같은 종래의 양자화 연산에 따라 변환된 블록 MBt_1의 양자화가 착수된다. 이어서, 양자화된 계수들의 블록 MBq_1이 획득된다.

[0116] 다음의 서브-단계 C345의 과정에서, 양자화된 계수들의 블록 MBq_1의 엔트로피 코딩이 착수된다. 바람직한 실시예에서, 이는 CABAC 엔트로피 코딩을 수반한다.

[0117] 다음의 서브-단계 C346의 과정에서, 블록 MBq_1의 역양자화(dequantization)가 종래의 역양자화 연산에 따라 착수되는데, 이는 단계 C344에서 수행된 양자화의 반대 연산이다. 이어서, 역양자화된 계수들의 블록 $MBDq_1$이 획득된다.

[0118] 다음의 서브-단계 C347의 과정에서, 위의 단계 C343에서 수행된 직접 변환의 반대의 연산인 역양자화된 계수들의 블록 $MBDq_1$의 역변환이 착수된다. 이어서, 디코딩된 잔차 블록 $MBDr_1$이 획득된다.

[0119] 다음의 서브-단계 C348의 과정에서, 예측된 블록 MBp_1에 디코딩된 잔차 블록 $MBDr_1$을 가산함으로써 디코딩된 블록 MBD_1의 구성이 착수된다. 디코딩된 잔차 블록은 명세서에서 더 상세하게 설명될 이미지 IE를 디코딩하는 방법의 완료 시 획득된 디코딩된 블록과 동일한 것임에 유의해야 한다. 따라서, 디코딩된 블록 MBD_1은 코딩 유닛(UC1) 또는 미리 결정된 수 R개의 코딩 유닛들 중 일부를 형성하는 임의의 다른 코딩 유닛에 의해 사용되도록 이용가능하게 만들어진다.

[0120] 상술한 코딩 단계 C34의 완료 시, 도 3b에 도시된 바와 같은 엔트로피 코딩 서브-유닛(SUCEk)은 첫 번째 블록의 코딩과 협력하여 계속하여 갱신된 바와 같은 모든 확률들을 포함한다. 이들 확률들은 다양한 가능한 구문 성분들 및 다양한 관련 코딩 콘텍스트들에 대응한다.

[0121] 상술한 코딩 단계 C34 후에, 단계 C35의 과정에서, 현재 블록이 이런 동일 열의 j 번째 블록인지의 여부를 결정하기 위한 검사가 수행되는데, 여기서 j는 적어도 1과 동일한 코더(CO)의 알려진 미리 결정된 값이다.

[0122] 그것이 사실이면, 단계 C36의 과정에서, j 번째 블록에 대하여 연산된 확률들의 세트가 도 3a 및 도 4a 및 도 4b에 도시된 바와 같은 코더(CO)의 버퍼 메모리(MT)에 저장되고, 상기 메모리의 크기는 연산된 수의 확률들을 저장하기 위해 적합하다.

[0123] 단계 C37의 과정에서, 유닛(UCk)은 방금 코딩되었던 열 SEk의 현재 블록이 이미지 IE의 마지막 블록인지의 여부를 검사한다.

[0124] 그것이 사실이면, 단계 C38의 과정에서, 코딩 방법은 종료된다.

[0125] 그것이 사실이 아니면, 단계 C39의 과정에서, 도 4a 또는 도 4b의 화살표 PS에 의해 도시된 트래버설의 순서에 따라 코딩될 다음 블록 MB_i의 선택이 착수된다.

[0126] 단계 C35의 과정에서, 현재 블록이 고려된 열 SEk의 j 번째 블록이 아니면, 위의 단계 C37이 착수된다.

[0127] 단계 C40의 과정에서, 현재 블록 MB_i을 코딩하기 위해 필요한 이전에 코딩 및 디코딩된 블록들의 이용가능성의 결정이 착수된다. 이것이 상이한 코딩 유닛들(UCk)에 의해 이미지 IE의 블록들의 병렬 코딩을 수반한다는 사실을 고려하면, 그것은 이들 블록들이 이들 블록들의 코딩에 할당된 코딩 유닛에 의해 코딩 및 디코딩되지 않았고, 따라서, 그것들이 아직 이용가능하지 않다는 것일 수 있다. 상기 결정 단계는 이전 열 SEk-1에 위치된 미리 결정된 수 N'개의 블록들, 예를 들면, 현재 블록의 위 및 우측 위에 각각 위치된 2개의 블록들이 현재 블록의 코딩을 위해 이용가능한지의 여부, 즉, 그것들이 이미 코딩되었고 그 다음 그것들의 코딩에 할당된 코딩 유닛(UCk-1)에 의해 디코딩되었는지의 여부를 확인하는 단계에 있다. 상기 결정 단계는 또한 코딩될 현재 블록 MB_i의 좌측에 위치된 적어도 하나의 블록의 이용가능성을 확인하는 단계에 있다. 그러나, 도 4a 또는 도 4b에 도시된 실시예에서 선택된 트래버설 PS의 순서를 고려하면, 블록들은 고려된 열 SEk에서 번갈아 코딩된다. 결과적으로, 좌측의 코딩 및 디코딩된 블록은 항상 이용가능하다(열의 첫 번째 블록을 제외하고). 도 4a 또는 도 4b에 도시된 예에서, 이는 코딩될 현재 블록의 바로 좌측에 위치된 블록을 수반한다. 이런 목적을 위해, 현재

블록의 위 및 우측 위에 각각 위치된 2개의 블록들의 이용가능성만이 검사된다.

[0128] 이런 검사 단계는 코딩 방법을 느리게 하기 쉽고, 본 발명에 따른 대안적인 방식으로, 도 3a에 도시된 클록(CLK)이, 현재 블록의 위 및 우측 위 각각에 위치된 2개의 블록들의 이용가능성을 확인할 필요 없이 이들 2개의 블록들의 이용가능성을 보장하도록 블록들의 코딩의 진행을 동기화하기 위해 적응된다. 따라서, 도 4a 또는 4b에 도시된 바와 같이, 코딩 유닛(UCk)은 항상 첫 번째 블록을 현재 블록의 코딩을 위해 사용되는 이전 열 SEk-1의 미리 결정된 수 N'(여기서 N'=2)개의 코딩 및 디코딩된 블록들만큼 시프트하여 코딩하기 시작한다. 소프트웨어 관점에서, 이런 클록의 구현은 코더(CO)에서 이미지 IE의 블록들을 처리하는데 필요한 프로세싱 시간을 현저하게 가속하는 것을 가능하게 한다.

[0129] 단계 C41의 과정에서, 현재 블록이 고려된 열 SEk의 첫 번째 블록인지의 여부를 결정하기 위한 검사가 수행된다.

[0130] 그것이 사실이면, 단계 C42의 과정에서, 이전 열 SEk-1의 j 번째 블록의 코딩 동안 연산된 심벌 발생 확률들만의 버퍼 메모리(MT)에서의 판독이 착수된다.

[0131] 도 4a에 도시된 제 1 변형예에 따르면, j 번째 블록은 이전 열 SEk-1(j=1)의 첫 번째 블록이다. 이런 판독은 CABAC 코더의 확률들을 버퍼 메모리(MT)에 존재하는 것으로 대체하는 것에 있다. 두 번째, 세 번째 및 네 번째 열들 SE2, SE3 및 SE4의 첫 번째 각 블록들을 갖는 것으로서 취급하면, 이런 판독 단계는 얇은 선들에 의해 표시된 화살표들에 의해 도 4a에 도시된다.

[0132] 도 4b에 도시되는 상술한 단계 C42의 제 2 변형예에 따르면, j 번째 블록은 이전 열 SEk-1(j=2)의 두 번째 블록이다. 이런 판독은 CABAC 코더의 확률들을 버퍼 메모리(MT)에 존재하는 것으로 대체하는 것에 있다. 두 번째, 세 번째 및 네 번째 열들 SE2, SE3 및 SE4의 첫 번째 각 블록들을 갖는 것으로서 취급하면, 이런 판독 단계는 얇은 대시 선들에 의해 표시된 화살표들에 의해 도 4b에 도시된다.

[0133] 단계 C42의 다음에, 현재 블록이 위에서 설명된 단계들 C34 내지 C38의 반복에 의해 코딩되고 그 다음 디코딩된다.

[0134] 상술한 단계 C41의 다음에, 현재 블록이 고려된 열 SEk의 첫 번째 블록이 아니면, 동일 열 SEk에 위치되는 이전에 코딩 및 디코딩된 블록, 즉, 도시된 예에서 현재 블록의 바로 좌측에 위치된 코딩 및 디코딩된 블록으로부터 발생하는 확률들의 판독은 유리하게 착수되지 않는다. 사실은, 도 4a 또는 도 4b에 도시된 바와 같이, 동일 열에 위치된 블록들의 판독의 순차적 트래버설 PS를 고려하면, CABAC 코더에 존재하는 심벌 발생 확률들은, 현재의 블록의 코딩이 시작할 때, 틀림없이 이런 동일 열의 이전 블록의 코딩/디코딩 이후에 존재하는 것이다.

[0135] 결과적으로, 단계 C43의 과정에서, 상기 현재 블록의 엔트로피 코딩에 대한 심벌 발생의 확률들의 학습이 착수되는데, 이런 확률들은 도 4a 또는 도 4b의 이중 실선 화살표들에 의해 도시된 바와 같이, 동일 열의 상기 이전 블록에 대하여 연산되었던 것들에만 대응한다.

[0136] 단계 C43의 다음에, 현재 블록이 위에 설명된 단계들 C34 내지 C38의 반복에 의해 코딩되고 그 다음 디코딩된다.

[0137] <u>디코딩 부분의 실시예의 상세한 설명</u>

[0138] 본 발명에 따른 디코딩 방법의 실시예가 이제 설명될 것이고, 여기서, 디코딩 방법은 초기에 H.264/MPEG-4 AVC 표준에 따른 디코더의 수정들에 의해 소프트웨어 또는 하드웨어 방식으로 구현된다.

[0139] 본 발명에 따른 디코딩 방법은 도 5a에 도시된 단계들(D1 내지 D4)을 포함하는 알고리즘의 형태로 도시된다.

[0140] 본 발명의 실시예들에 따르면, 본 발명에 따른 디코딩 방법은 도 6a에 도시된 디코딩 디바이스(DO)에서 구현된다.

[0141] 도 5a를 참조하여, 제 1 디코딩 단계(D1)는, 도 4a 또는 도 4b에 도시된 바와 같이, 이전에 코딩된 블록들 또는 매크로-블록(MB)의 N 개의 서브셋들 SE1, SE2,..., SEk,...,SEP 각각을 포함하는 N 개의 서브-스트림들 F1, F2,...,Fk,..., FP의 상기 스트림 F의 식별이다. 이런 목적을 위해, 스트림 F의 각 서브-스트림 Fk는 디코더(DO)가 스트림 F의 각각의 서브-스트림 Fk의 위치를 결정하게 하도록 의도된 표시자(indicator)와 관련된다. 도시된 예에서, 상기 블록들(MB)은 정사각형 형상을 갖고 모두가 동일한 크기를 갖는다. 반드시 블록들의 크기의 배수가 아닌 이미지의 크기의 함수로서, 좌측 상의 마지막 블록들 및 하부에서 마지막 블록들은 정사각형이 아닐 수 있다. 대안적인 실시예에서, 블록들은, 예를 들면, 직사각형 크기 및/또는 서로 정렬되지 않을 수 있

다.

[0142] 각 블록 또는 매크로블록은 게다가 그 자체로 그것들 자체가 세분될 수 있는 서브-블록들로 분할될 수 있다.

[0143] 이런 식별은 도 6a에 도시된 바와 같은 스트림 추출 모듈(EXDO)에 의해 수행된다.

[0144] 도 4a 또는 도 4b에 도시된 예에서, 미리 결정된 수는 4와 같고, 4개의 서브셋들 SE1, SE2, SE3, SE4는 대시로 도시된다.

[0145] 도 5a를 참조하여, 제 2 디코딩 단계(D2)는 블록들의 상기 서브셋들 SE1, SE2, SE3 및 SE4의 각각을 병렬로 디코딩하는 단계이고, 고려된 서브셋의 블록들은 트래버설 PS의 미리 결정된 순차적 순서에 따라 코딩된다. 도 4a 또는 도 4b에 도시된 예들에서, 현재 서브셋 SEk(1≤k≤4)의 블록들은 화살표 PS에 의해 지시된 바와 같이, 좌측에서 우측으로 번갈아 디코딩된다. 단계 D2의 완료 시, 디코딩된 블록들의 서브셋들 SED1, SED2, SED3,....,SEDk,..., SEDP가 획득된다.

[0146] 이런 병렬 디코딩은 도 6a에 도시된 바와 같이, R=4인 다수의 R개의 디코딩 유닛들(UDk(1≤k≤R))에 의해 구현되고, 디코딩 방법의 상당한 가속을 가능하게 한다. 이와 같이 알려진 방법으로, 디코더(DO)는 현재 블록의 디코딩과 협력하여 계속적으로 갱신되는 바와 같은 심벌들의 발생의 확률들을 포함하도록 적응되는 버퍼 메모리(MT)를 포함한다.

[0147] 도 6b에 더 상세하게 도시된 바와 같이, 디코딩 유닛들(UDk)의 각각은,

[0148] ● SUDEk로 지시된, 적어도 하나의 이전 디코딩된 블록에 대하여 연산된 적어도 하나의 심벌 발생의 확률을 학습함으로써 상기 현재 블록을 엔트로피 디코딩하기 위한 서브-유닛,

[0149] ● SUDPk로 지시된, 상기 이전 디코딩된 블록들에 대하여 현재 블록의 예측 디코딩을 하기 위한 서브-유닛을 포함한다.

[0150] 예측 디코딩 서브-유닛(SUDPk)은 예를 들면, 인트라 및/또는 인터 모드에서와 같은 종래의 예측 기법들에 따라 현재 블록의 예측 디코딩을 수행할 수 있다.

[0151] 엔트로피 디코딩 서브-유닛(SUDEk)은 CABAC 유형의 그 일부이지만, 명세서에서 더 설명될 것과 같이, 본 발명에 따라 수정된다.

[0152] 변형예로서, 엔트로피 디코더 서브-유닛(SUDEk)은 이와 같이 알려진 허프만 디코더일 수 있다.

[0153] 도 4a 또는 도 4b에 도시된 예들에서, 제 1 유닛(UD1)은 첫 번째 열 SE1의 블록들을 좌측에서 우측으로 디코딩한다. 그것이 첫 번째 열 SE1의 마지막 블록에 도달하면, 그것은 (N+1) 번째 열, 여기서 5 번째 열 등의 첫 번째 블록으로 이동한다. 제 2 유닛(UC2)은 두 번째 열 SE2의 블록들을 좌측에서 우측으로 디코딩한다. 그것이 두 번째 열 SE2의 마지막 블록에 도달하면, 그것은 (N+2) 번째 열, 여기서 6 번째 열 등의 첫 번째 블록으로 이동한다. 이런 트래버설은 네 번째 열 SE4의 블록들을 좌측에서 우측으로 디코딩하는 유닛(UD4)까지 반복된다. 그것이 첫 번째 열의 마지막 블록에 도달하면, 마지막 식별된 서브-스트림의 마지막 블록이 디코딩될 때까지, 그것은 (N+4) 번째 열, 여기서 8 번째 열 등등의 첫 번째 블록으로 이동한다.

[0154] 위에서 방금 설명되었던 것과 다른 유형들의 트래버설도 물론 가능하다. 예를 들면, 각 디코딩 유닛이, 위에서 설명된 바와 같이, 네스팅된 열들이 아니라 네스팅된 행들을 처리할 수 있다. 열들 또는 행들 중 어느 하나의 방향으로 트래버스하는 것이 또한 가능하다.

[0155] 도 5a를 참조하여, 제 3 디코딩 단계(D3)는 디코딩 단계 D2에서 획득된 각 디코딩된 서브셋 SED1, SED2,...,SEDk,...,SEDP에 기반하여 디코딩된 이미지의 재구성이다. 보다 정확하게는, 각 디코딩된 서브셋 SED1, SED2,...,SEDk,...,SEDP의 디코딩된 블록들은 도 6a에 도시된 바와 같은 이미지 재구성 유닛(URI)으로 전송된다. 이런 단계 D3의 과정에서, 유닛(URI)은 이런 블록들이 이용 가능하게 되기 때문에 그리고 이런 경우, 디코딩된 블록들을 디코딩된 이미지에 기록한다.

[0156] 도 5a에 도시된 제 4 디코딩 단계(D4)의 과정에서, 완전히 디코딩된 이미지 ID는 도 6a에 도시된 유닛(URI)에 의해 전달된다.

[0157] 상술한 병렬 디코딩 단계(D2) 동안, 디코딩 유닛(UDk)에서 구현되는 바와 같은 본 발명의 다양한 특정 서브-단계들은 도 5b를 참조하여 이제 설명될 것이다.

[0158] 단계 D21의 과정에서, 디코딩 유닛(UDk)은 도 4a 또는 도 4b에 도시된 현재 열 SEk의 디코딩될 첫 번째 블록을

현재 블록으로서 선택한다.

[0159] 단계 D22의 과정에서, 유닛(UDk)은 현재 블록이 디코딩된 이미지의 첫 번째 블록, 이러한 예에서 서브-스트림 F1의 첫 번째 블록인지의 여부를 검사한다.

[0160] 그것이 사실이면, 단계 D23의 과정에서, 디코딩 확률들이 도 6a의 디코더(DO)에서 이전에 정의된 값들 Pinit로 초기화된다.

[0161] 그것이 사실이 아니면, 다음의 명세서에서 나중에 설명될 단계 D30의 과정에서, 필요한 이전에 디코딩된 블록들의 이용가능성의 결정이 착수된다.

[0162] 단계 D24의 과정에서, 도 4a 또는 도 4b에 도시된 첫 번째 열 SE1의 첫 번째 현재 블록 MB1의 디코딩이 착수된다. 이런 단계 D24는 이하에 설명될 다수의 서브-단계들 D241 내지 D246을 포함한다.

[0163] 첫 번째 서브-단계 D241의 과정에서, 현재 블록과 관련된 구문 성분들의 엔트로피 디코딩이 착수된다. 보다 정확하게는 현재 블록과 관련된 구문 성분들은 도 6b에 도시된 바와 같은 CABAC 엔트로피 디코딩 서브-유닛(SUDE1)에 의해 디코딩된다. 엔트로피 디코딩 서브-유닛은 구문 성분들을 생성하기 위해 압축된 파일의 서브-비트스트림 F1을 디코딩하고, 동시에 이런 서브-유닛이 심벌을 디코딩하는 때에, 이런 심벌의 발생의 확률들이, 상술한 엔트로피 코딩 단계 C345 동안, 이런 동일한 심벌의 코딩 동안 획득된 것과 동일한 것과 같은 방법으로 그것의 확률들을 갱신한다.

[0164] 다음의 서브-단계 D242의 과정에서, 블록 MB1이 적어도 하나의 이전에 디코딩된 블록에 대하여 예측되는 과정에서, 현재 블록 MB1의 예측 디코딩이 인트라 및/또는 인터 예측의 알려진 기법들에 의해 착수된다.

[0165] H.264 표준에서 제안된 바와 같은 인트라 예측의 다른 모드들이 가능함은 말할 필요도 없다.

[0166] 이런 단계의 과정에서, 예측 디코딩은 이전 단계에서 디코딩된 구문 성분들의 도움으로 수행되고, 특히 예측의 유형(인터 또는 인트라), 및 적합하면, 인트라 예측의 모드, 블록 또는 매크로블록이 세분되었던 경우, 블록 또는 매크로블록의 파티션의 유형, 기준 이미지 인덱스 및 인터 예측 모드에서 사용된 변위 벡터를 포함한다.

[0167] 상기 상술한 예측 디코딩 단계는 예측된 블록 MBp_1을 구성하는 것을 가능하게 한다.

[0168] 다음의 서브-단계 D243의 과정에서, 이전에 디코딩된 구문 성분들의 도움으로 양자화된 잔차 블록 MBq_1의 구성이 착수된다.

[0169] 다음의 서브-단계 D244의 과정에서, 디코딩된 역양자화된 블록 $MBDt_1$을 생성하기 위해 상술한 단계 C344에서 수행된 양자화의 반대 연산인 종래의 역양자화 연산에 따라 양자화된 잔차 블록 MBq_1의 역양자화가 착수된다.

[0170] 다음의 서브-단계 D245의 과정에서, 위의 단계 C343에서 수행된 직접 변환의 반대의 연산인 역양자화된 블록 $MBDt_1$의 역변환이 착수된다. 이어서, 디코딩된 잔차 블록 $MBDr_1$이 획득된다.

[0171] 다음의 서브-단계 D246의 과정에서, 예측된 블록 MBp_1에 디코딩된 잔차 블록 $MBDr_1$을 가산함으로써 디코딩된 블록 MBD_1의 구성이 착수된다. 따라서, 디코딩된 블록 MBD_1은 디코딩 유닛(UD1) 또는 미리 결정된 수 N개의 디코딩 유닛들 중 일부를 형성하는 임의의 다른 디코딩 유닛에 의해 사용되도록 이용가능하게 만들어진다.

[0172] 상술한 디코딩 단계 D246의 완료 시, 도 6b에 도시된 바와 같은 엔트로피 디코딩 서브-유닛(SUDE1)은 첫 번째 블록의 디코딩과 협력하여 계속하여 갱신된 바와 같은 모든 확률들을 포함한다. 이들 확률들은 다양한 가능한 구문 성분들 및 다양한 관련된 디코딩 콘텍스트들에 대응한다.

[0173] 상술한 디코딩 단계 D24 후에, 단계 D25의 과정에서, 현재 블록이 이런 동일 열의 j 번째 블록인지의 여부를 결정하기 위한 검사가 수행되는데, 여기서 j는 적어도 1과 같은 디코더(DO)의 알려진 미리 결정된 값이다.

[0174] 그것이 사실이면, 단계 D26의 과정에서, j 번째 블록에 대하여 연산된 확률들의 세트가 도 6a 및 도 4a 및 도 4b에 도시된 바와 같은 디코더(DO)의 버퍼 메모리(MT)에 저장되고, 상기 메모리의 크기는 연산된 수의 확률들을 저장하기 위해 적합하다.

[0175] 단계 D27의 과정에서, 유닛(UDk)은 방금 디코딩되었던 현재 블록이 마지막 서브-스트림의 마지막 블록인지의 여부를 검사한다.

[0176] 그것이 사실이면, 단계 D28의 과정에서, 디코딩 방법은 종료된다.

[0177] 그것이 사실이 아니면, 단계 D29의 과정에서, 도 4a 또는 도 4b의 화살표 PS에 의해 도시된 트래버설의 순서에 따라 디코딩될 다음 블록 MB_i의 선택이 착수된다.

[0178] 상술한 단계 D25의 과정에서, 현재 블록이 고려된 열 SEDk의 j 번째 블록이 아니면, 위의 단계 D27이 착수된다.

[0179] 상술한 단계 D29에 뒤이은 단계 D30의 과정에서, 현재 블록 MB_i를 디코딩하기 위해 필요한 이전에 디코딩된 블록들의 이용가능성의 결정이 착수된다. 이것이 상이한 디코딩 유닛들(UDk)에 의해 블록들의 병렬 디코딩을 수반한다는 사실을 고려하면, 그것은 이들 블록들이 이들 블록들의 디코딩에 할당된 디코딩 유닛에 의해 디코딩되지 않았고 따라서, 그것들이 아직 이용가능하지 않다는 것일 수 있다. 상기 결정 단계는 이전 열 SEk-1에 위치된 미리 결정된 수 N'개의 블록들, 예를 들면, 현재 블록의 위 및 우측 위에 각각 위치된 2개의 블록들이 현재 블록의 디코딩을 위해 이용가능한지의 여부, 즉, 그것들이 그것들의 디코딩에 할당된 디코딩 유닛(UDk-1)에 의해 이미 디코딩되었는지의 여부를 확인하는 단계에 있다. 상기 결정 단계는 또한 디코딩될 현재 블록 MB_i의 좌측에 위치된 적어도 하나의 블록의 이용가능성을 확인하는 단계에 있다. 그러나, 도 4a 또는 도 4b에 도시된 실시예에서 선택된 트래버설 PS의 순서를 고려하면, 블록들은 고려된 열 SEk에서 번갈아 디코딩된다. 결과적으로, 좌측의 디코딩된 블록은 항상 이용가능하다(열의 첫 번째 블록을 제외하고). 도 4a 또는 도 4b에 도시된 예에서, 이는 디코딩될 현재 불록의 바로 좌측에 위치된 블록을 수반한다. 이런 목적을 위해, 현재 블록의 위 및 우측 위에 각각 위치된 2개의 블록들의 이용가능성만이 검사된다.

[0180] 이런 검사 단계는 디코딩 방법을 느리게 하기 쉽고, 본 발명에 따른 대안적인 방식으로, 도 6a에 도시된 클록(CLK)이, 현재 블록의 위 및 우측 위 각각에 위치된 2개의 블록들의 이용가능성을 확인할 필요 없이 이들 2개의 블록들의 이용가능성을 보장하도록 블록들의 디코딩의 진행을 동기화하기 위해 적응된다. 따라서, 도 4a 또는 4b에 도시된 바와 같이, 디코딩 유닛(UDk)은 항상 첫 번째 블록을 현재 블록의 디코딩을 위해 사용되는 이전 열 SEk-1의 미리 결정된 수 N'(여기서 N'=2)개의 디코딩된 블록들만큼 시프트하여 디코딩하기 시작한다. 소프트웨어 관점에서, 이런 클록의 구현은 디코더(DO)에서 각 서브셋 SEk의 블록들을 처리하는데 필요한 프로세싱 시간을 현저하게 가속하는 것을 가능하게 한다.

[0181] 단계 D31의 과정에서, 현재 블록이 고려된 열 SEk의 첫 번째 블록인지의 여부를 결정하기 위한 검사가 수행된다.

[0182] 그것이 사실이면, 단계 D32의 과정에서, 이전 열 SEk-1의 j 번째 블록의 디코딩 동안 연산된 심벌 발생 확률들만의 버퍼 메모리(MT)에서의 판독이 착수된다.

[0183] 도 4a에 도시된 제 1 변형예에 따르면, j 번째 블록은 이전 열 SEk-1(j=1)의 첫 번째 블록이다. 이런 판독은 CABAC 디코더의 확률들을 버퍼 메모리(MT)에 존재하는 것으로 대체하는 것에 있다. 두 번째, 세 번째 및 네 번째 열들 SE2, SE3 및 SE4의 첫 번째 각 블록들을 갖는 것으로서 취급하면, 이런 판독 단계는 얇은 선들에 의해 표시된 화살표들에 의해 도 4a에 도시된다.

[0184] 도 4b에 도시되는 상술한 단계 D32의 제 2 변형예에 따르면, j 번째 블록은 이전 열 SEk-1(j=2)의 두 번째 블록이다. 이런 판독은 CABAC 디코더의 확률들을 버퍼 메모리(MT)에 존재하는 것으로 대체하는 것에 있다. 두 번째, 세 번째 및 네 번째 열들 SE2, SE3 및 SE4의 첫 번째 각 블록들을 갖는 것으로서 취급하면, 이런 판독 단계는 얇은 대시 선들에 의해 표시된 화살표들에 의해 도 4b에 도시된다.

[0185] 단계 D32의 다음에, 현재 블록이 위에서 설명된 단계 D24 내지 D28의 반복에 의해 디코딩된다.

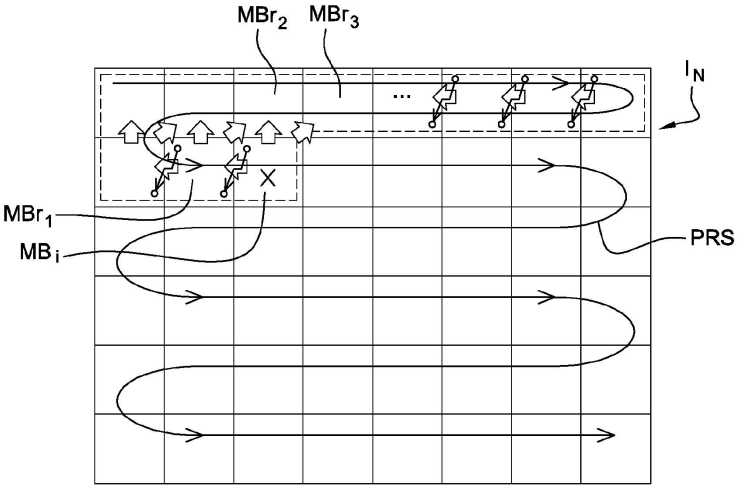
[0186] 상술한 단계 D31의 다음에, 현재 블록이 고려된 열 SEk의 첫 번째 블록이 아니면, 동일 열 SEk에 위치되는 이전에 디코딩된 블록, 즉, 도시된 예에서 현재 블록의 바로 좌측에 위치된 디코딩된 블록으로부터 발생하는 확률들의 판독은 유리하게 착수되지 않는다. 실제로, 도 4a 또는 도 4b에 도시된 바와 같이, 동일 열에 위치된 블록들의 판독의 순차적 트래버설 PS를 고려하면, CABAC 디코더에 존재하는 심벌 발생 확률들은, 현재의 블록의 코딩을 시작할 때, 틀림없이 이런 동일 열의 이전 블록의 디코딩 이후에 존재하는 것이다.

[0187] 결과적으로, 단계 D33의 과정에서, 상기 현재 블록의 엔트로피 디코딩에 대한 심벌 발생의 확률들의 학습이 착수되는데, 이런 확률들은 도 4a 또는 도 4b의 이중 실선 화살표들에 의해 도시된 바와 같이, 동일 열의 상기 이전 블록에 대하여 연산되었던 것들에만 대응한다.

[0188] 단계 D33의 다음에, 현재 블록이 위에 설명된 단계 D24 내지 D28의 반복에 의해 디코딩된다.

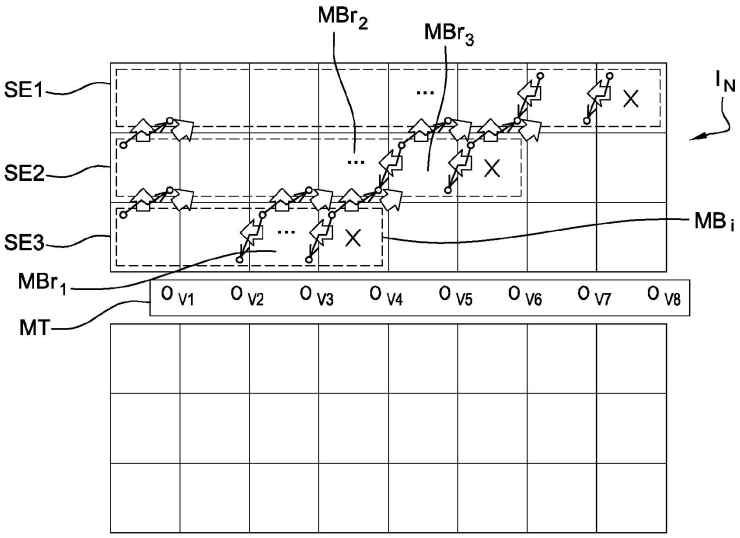
도면

도면1a



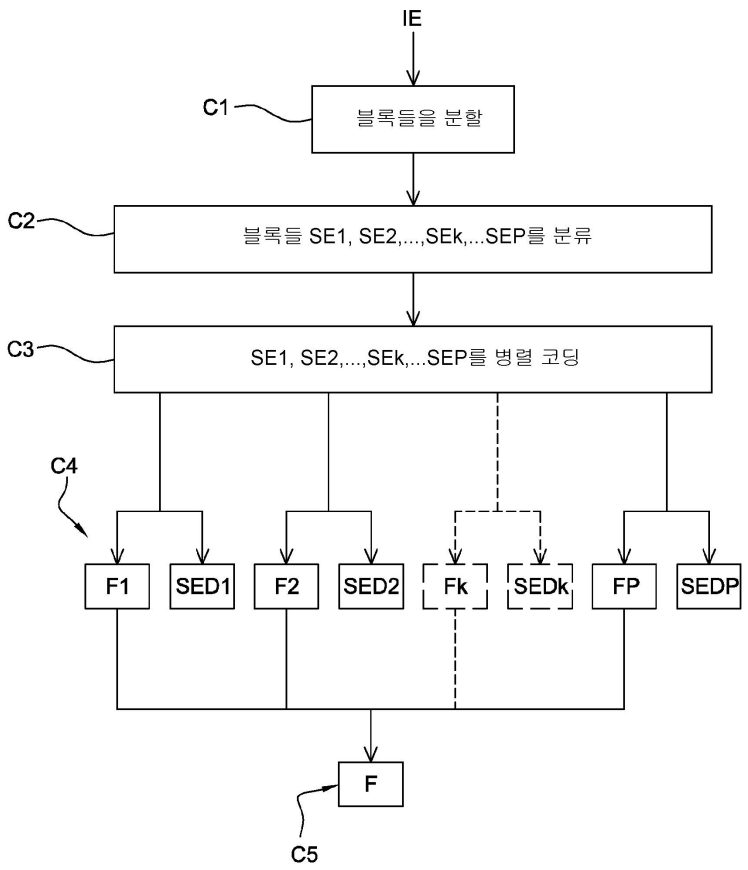
종래 기술

도면1b

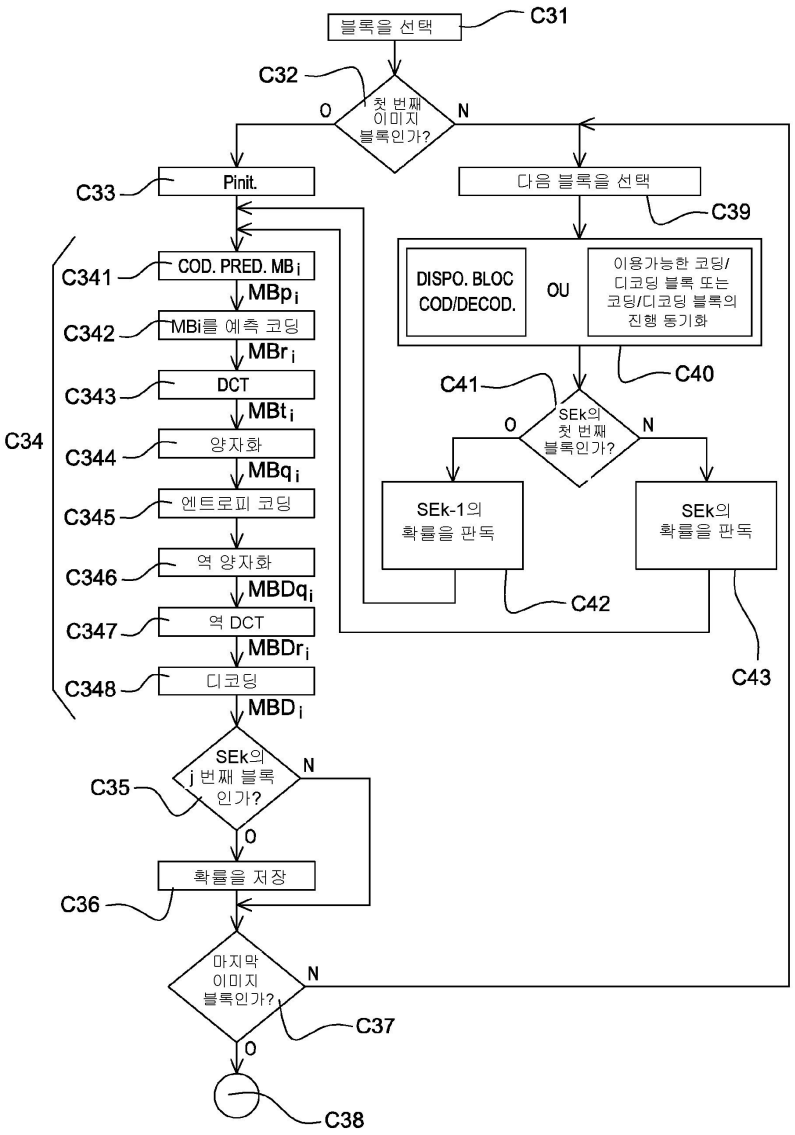


종래 기술

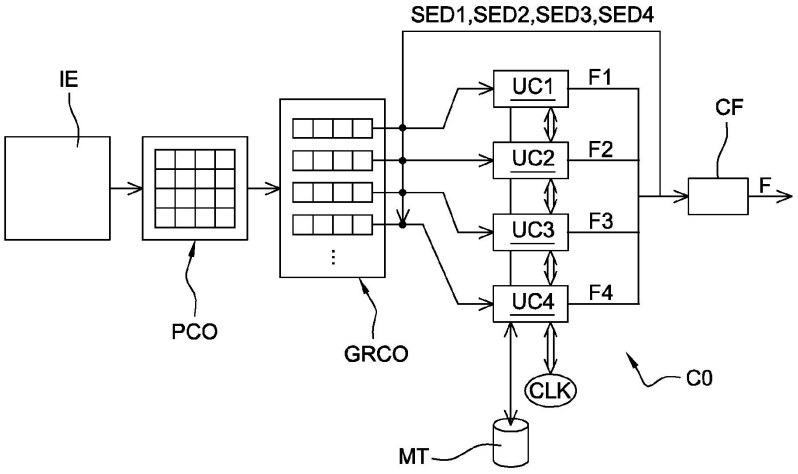
도면2a


IE
C1
블록들을 분할
C2
블록들 SE1, SE2,...,SEk,...SEP를 분류
C3
SE1, SE2,...,SEk,...SEP를 병렬 코딩
C4
F1
SED1
F2
SED2
Fk
SEDk
FP
SEDP
F
C5

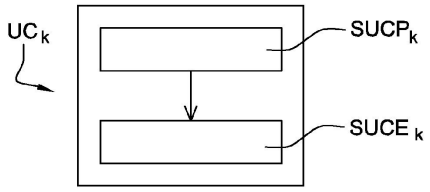
도면2b


블록을 선택
C31
C32
첫 번째 이미지 블록인가?
O
N
C33
Pinit.
다음 블록을 선택
C39
C341
COD. PRED. MBi
MBpi
C342
MBi를 예측 코딩
MBri
C343
DCT
MBti
C34
C344
양자화
MBqi
C345
엔트로피 코딩
C346
역 양자화
MBDqi
C347
역 DCT
MBDri
C348
디코딩
MBDi
DISPO. BLOC COD/DECOD.
OU
이용가능한 코딩/디코딩 블록 또는 코딩/디코딩 블록의 진행 동기화
C40
C41
SEk의 첫 번째 블록인가?
SEk-1의 확률을 판독
SEk의 확률을 판독
C42
C43
SEk의 j 번째 블록 인가?
C35
확률을 저장
C36
마지막 이미지 블록인가?
C37
C38

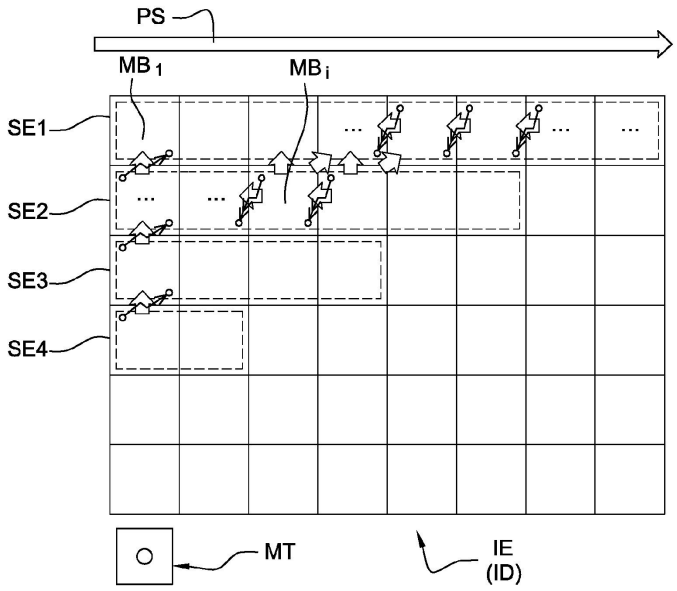
도면3a


SED1,SED2,SED3,SED4
IE
UC1
UC2
UC3
UC4
F1
F2
F3
F4
CF
F
PCO
GRCO
CLK
C0
MT

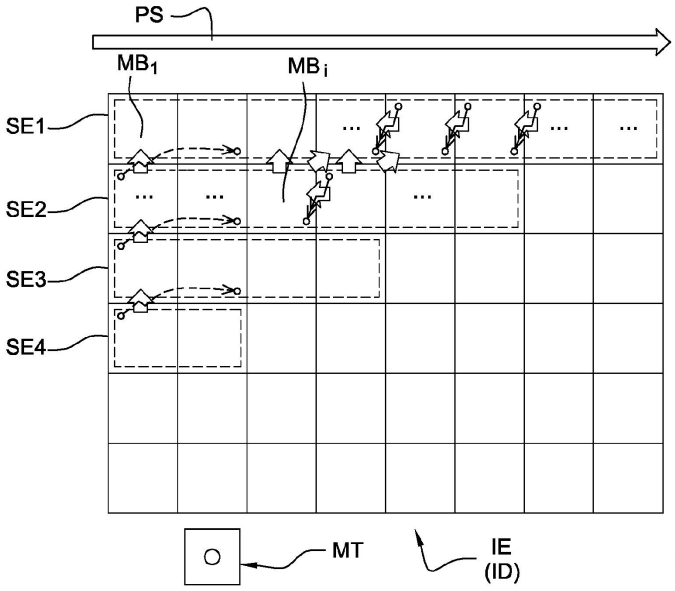
도면3b



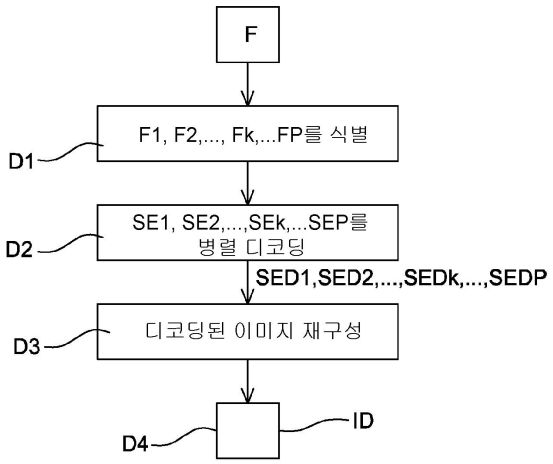
도면4a



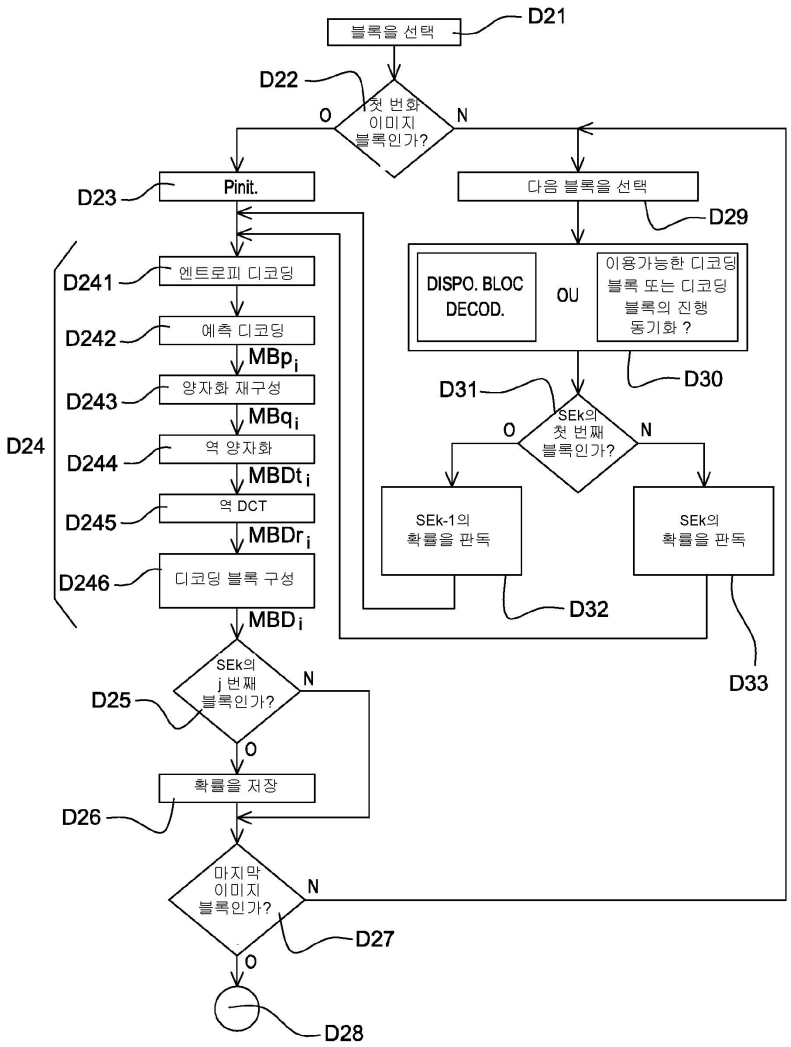
도면4b



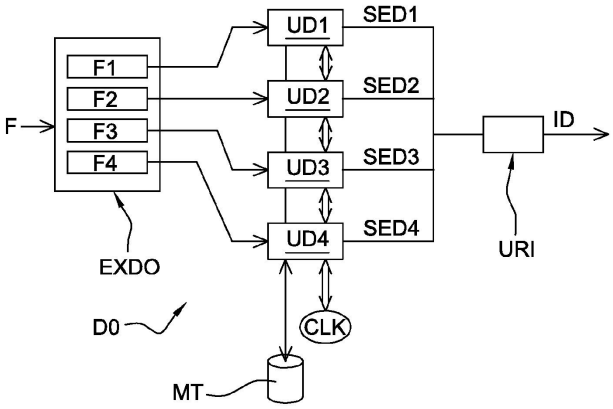
도면5a


F
F1, F2,..., Fk,...FP를 식별
D1
SE1, SE2,...,SEk,...SEP를
병렬 디코딩
D2
SED1,SED2,...,SEDk,...,SEDP
디코딩된 이미지 재구성
D3
D4
ID

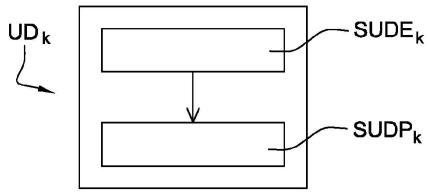
도면5b


블록을 선택
D21
D22
첫 번째
이미지
블록인가?
O
N
D23
Pinit.
다음 블록을 선택
D29
D241
엔트로피 디코딩
D242
예측 디코딩
MBp_i
D243
양자화 재구성
MBq_i
D24
D244
역 양자화
MBDt_i
D245
역 DCT
MBDr_i
D246
디코딩 블록 구성
MBD_i
DISPO. BLOC
DECOD.
OU
이용가능한 디코딩
블록 또는 디코딩
블록의 진행
동기화 ?
D30
D31
SEk의
첫 번째
블록인가?
SEk-1의
확률을 판독
SEk의
확률을 판독
D32
D33
SEk의
j 번째
블록인가?
D25
확률을 저장
D26
마지막
이미지
블록인가?
D27
D28

도면6a


F
F1
F2
F3
F4
EXDO
UD1
UD2
UD3
UD4
SED1
SED2
SED3
SED4
ID
URI
D0
CLK
MT

도면6b


UDk
SUDEk
SUDPk