



## 기술분야

본 발명은 화상 등의 이차원 데이터에 전자무늬를 매립하는 방법, 매립된 전자무늬를 복호하는 복호방법 및 이를 행하는 장치에 관한 것이다.

## 배경기술

인터넷 등의 컴퓨터 네트워크의 발전에 동반하여, 정보의 디지털화가 진행되고, 많은 사용자가 간단히 필요로 하는 정보에 액세스 할 수 있도록 되어 있다. 그 반면, 그 디지털 정보에 저작권이 발생하고 있는 디지털 콘텐츠에 관해, 그 저작자에게 사전 양해 없이 용이하게 데이터를 복제할 수 있는 바와 같은 환경이 되고 있으며, 부정 카피(copy)에 동반하는 저작권 침해의 문제가 주목되어 오고 있다. 그래서, 디지털 콘텐츠의 주된 정보인 컬러 화상에 관한 저작권 침해를 방지하는 것을 목적으로 하여, 저작권정보를 제삼자에게 보이지 않는 형태로 몰래 컬러 화상의 데이터에 매립하고, 언제라도 그 화상으로부터 저작권정보를 추출할 수 있는 바와 같은 기술, 소위 전자무늬의 기술이 주목되고 있다.

화상 등의 이차원적인 데이터에 전자무늬를 매립하는 방법으로서, 예컨대 화상 데이터에 대해 이산 코사인 변환등의 직교 변환을 행하고, 얻어진 계수를 소정의 알고리즘으로 변화시킨다고 한 수법(手法)이 알려져 있다. 이 경우에는 원화상을 비닉(秘匿:몰래 감춤)해 두고, 원화상과 복사되었다고 생각되는 화상을 동일하게 직교 변환하여, 그 계수를 비교함으로써, 양자의 일치 여부를 검증하고 있다.

그렇지만, 이러한 수법에서는 저작권정보를 추출하는 데 비닉된 원화상을 필요로 하게 되어, 원화상의 관리라고 하는 새로운 문제를 야기한다. 원화상의 수가 늘어남에 따라, 그 관리는 점차 곤란한 것으로 된다. 따라서, 전자무늬는 가능한 한, 화상으로부터 직접적으로 취출(取出)할 수 있는 형태로 매립하는 것이 바람직하다.

한 편, 이러한 전자무늬에 대해서는 화상의 일부의 오버라이팅(overwriting) 공격 등에 대해 견딜 수 있는 지어떤지 라고 하는 평가가 필요하게 된다. 근년, 직교 변환을 이용한 전자무늬에 대해서는 라플라시언(Laplacian) 공격 등, 여러 가지 공격수법이 밝혀져 있다. 보다 많은 정보를 화상측에 매립한 경우, 이러한 공격이나 화상의 일부 이용이라고 한 문제에 대해, 충분한 유효성을 갖는 수법은 아직 제안되어 있지 않다.

## 발명의 상세한 설명

본 발명은 직교 변환을 이용한 전자무늬의 매립과 복호에 관한 상기한 문제를 해결하고, 여러 가지의 공격에 대해 충분한 내성(耐性)을 갖는 전자무늬의 매립과 복호의 기술을 제안하는 것을 목적으로 한다.

본 발명은 화상에 전자무늬 정보를 매립하는 방법으로서,

원화상으로부터 소정의 크기의 블록을 절출(切出:베어 냄)하고,

그 각 블록에 대응한 화상 데이터에 대해 직교 변환을 행하며,

소정의 관계를 갖는 두개 이상의 블록의 상기 직교 변환후의 대응하는 계수를 비교하고, 전자무늬로 하는 비트 정보에 의해, 그 계수의 한 쪽을 다른 쪽에 대해 소정의 대소 관계로 함으로써, 그 비트 정보를 매립하고,

상기 비트 정보가 매립된 블록을 역 직교변환 함으로써, 전자무늬가 매립된 화상을 출력하는

것을 요지로 하고 있다.

이 전자무늬의 매립방법에 의하면, 소정의 관계에 있는 두개 이상의 블록을 이용하고, 이들의 블록을 직교 변환한 후의 계수를 비교하고, 양(兩)계수의 대소 관계를 이용하여 비트 정보를 매립하고 있다. 따라서, 블록의 관계와, 이 계수 사이의 대소관계를 알고 있으면, 전자무늬를 화상으로부터 언제라도 취출할 수 있고, 대소 관계를 모르면, 전자무늬를 취출할 수 없다. 이 때문에, 취급이 용이하고, 또한 공격에 강한 전자무늬의 매립방법을 실현할 수 있다. 또, 전자무늬를 매립한 원화상을 비닉해 두고, 이것과의 비교에 의해 전자무늬를 확인한다고 하는 사용법도 물론 가능하다.

한 편, 전자무늬가 매립된 화상 데이터로부터 전자무늬를 취출하는 복호방법의 발명은,

원화상으로부터 소정의 크기의 블록을 절출하고,

그 각 블록에 대응한 화상 데이터에 대해 직교 변환을 행하며,

소정의 관계를 갖는 두개 이상의 블록의 상기 직교 변환후의 대응하는 계수를 비교하고, 그 계수의 한 쪽의 다른 쪽에 대한 대소 관계에 의거하여, 비트 정보를 취출하는

것을 요지로 하고 있다.

이 전자무늬의 복호방법에 의하면, 용이하게 전자무늬를 취출할 수 있다.

이러한 원리를 기본으로 하여, 여러 가지의 구체화가 가능하다. 예컨대, 소정의 관계에 있는 블록으로서 서로 이웃한 블록을 이용할 수 있다. 화상에서, 인접 화상에는 소정의 상관(相關)이 존재하는 경우가 많기 때문에, 블록 사이의 계수의 대소 관계에 일정한 경향이 생기는 경우가 있고, 본 발명의 전자무늬의 매립에 적합한 경우가 있기 때문이다.

또한, 화상 데이터에 대한 직교 변환으로서, 이산 코사인 변환을 이용할 수 있다. 이산 코사인 변환은 JPEG 등에도 채용되어 있고, 화상의 압축 등이 뛰어난 특성을 나타내기 때문이다. 직교 변환후의 계수를 양자화 테이블을 이용하여 양자화하고, 상기 정보 비트의 매립을 그 양자화된 계수를 이용하여 행하는 것도 바람직하다. 양자 테이블을 조정함으로써, 압축 효율을 자유롭게 가변가능하기 때문이다.

이러한 화상 변환의 일례로서, 화상 데이터를 상기 직교 변환에 앞서, 휘도(Y), 색차(Cb, Cr)로 변환하고, 그 휘도(Y), 색차(Cb, Cr)의 각각에 대해 상기 직교 변환으로서 이산 코사인 변환을 행하고, 상기 휘도(Y)를 이산 코사인 변환하여 얻어진 계수에 대해, 상기 비트 정보의 매립을 행하는 구성을 생각할 수 있다.

또, 양자화후의 양 블록의 대응하는 계수가 공히(모두) 0의 값이 아닌 경우에, 비트 정보의 매립을 행하는 것으로 하는 것도 바람직하다. 양 블록이 0의 값인 경우에 비트 정보를 매립하여, 계수를 0의 값 이외의 값으로 변경하면, 데이터의 압축 효율이 저하할 염려가 생기기 때문이다.

상기한 전자무늬의 매립에서, 상기 소정의 관계에 있는 두개 이상의 블록의 직교 변환후의 계수의 차가 소정의 범위에 있는 경우를 진(眞)으로 하는 논리 함수를 도입하는 것도, 연산을 간이화하는데 있어서 바람직하다. 이 경우에는 논리 함수의 진위(眞僞)에 의거하여, 상기 비트 정보의 매립의 수법을 변경하는 것으로 하면 좋다. 이 소정의 범위를 가변하는 것이 용이하여 진다.

또, 이 전자무늬의 매립방법에 있어서,

직교 변환에 의해 얻어지는 계수에 대응시켜진 비밀 키(key)를 미리 준비하고,

상기 계수마다의 그 비밀 키와 상기 계수에 대한 논리함수의 진위 상태에 의거하여, 상기 비트 정보의 매립의 수법을 변경하는 것으로 하는 것도 바람직하다. 이 경우에는, 비밀 키가 없으면 어떠한 규칙으로 비트 정보가 매립되어 있는지를 특정하는 것은 지극히 곤란하게 된다. 통상 이러한 비밀 키는 화상마다 관리해도 좋지만, 저작자마다 관리할 수도 있어, 전자무늬의 관리가 용이해 진다.

여기서, 2치(値) 정보의 조합으로서 이차원적으로 정의된 기본 패턴을 전자무늬의 정보로서 준비하고, 이 기본 패턴을 구성하는 각 2치 정보를 매립되는 비트 정보로 하며, 소정의 관계에 있는 블록 끼리를 하나의 단위로 하여, 이 기본 패턴을 구성하는 상기 2치 정보를 매립하는 것도 유용하다. 이 경우에는 전자무늬가 이차원적으로 정의된 기본 패턴으로서 매립되게 되고, 이 기본 패턴을 직접 저작권정보 등으로서 이용할 수 있기 때문이다. 이러한 기본 패턴으로서의 용장성(冗長性)을 갖는 패턴을 이용하는 것이 바람직하다. 패턴의 용장성을 살려 오류(예러) 검출 등을 행할 수 있기 때문이다. 전자무늬를 매립하고자 하는 화상의 성질에 따라, 전자무늬에 이용하는 기본 패턴의 비트 정보를 변경한다고 한 응용도 가능하다. 이러한 용장성을 갖는 기본 패턴으로서 예컨대 농도 패턴법을 이용하여 정할 수 있다. 농도 패턴은,  $n \times n$  화소의 영역을 생각하고, 그 농도가 동일하게 되는 도트의 배치가 복수 존재한다고 하는 원리에 의거하는 것이며 높은 용장성을 갖는다. 따라서, 농도 패턴을 이용하면, 그 높은 용장성을 살려 오류 검출 등을 실현할 수 있다.

또, 이 기본 패턴을 구성하는 요소의 수가 상기 절출된 블록수 보다 충분히 클 때, 그 기본 패턴을 상기 화상 데이터에 소정 횟수 반복하여 매립하는 것이 가능하다. 이렇게 해 놓으면, 화상의 일부를 오려 내거나 해도, 기본 패턴을 복호하는 것이 가능해지고, 전자무늬의 내성을 향상할 수 있다.

상기의 전자무늬의 매립방법 및 전자무늬의 복호방법은 이를 기억한 기억매체로서 실시할 수 있다. 이러한 기억매체로서는 CD-ROM이나 메모리 카드, 플렉시블 디스크 등의 형태를 생각할 수 있다. 또한, 통신회선을 거쳐 이들의 프로그램을 다운 로드하는 형태도 생각할 수 있다. 또, 이러한 전자무늬의 매립방법은 화상 데이터를 취급하는 리터치 소프트 등에, 일체로 혹은 플러그 인(plug-in) 소프트로서 내장할 수 있다. 물론, 전자무늬의 매립이나 복호만을 행하는 독립된 프로그램 으로서 실현하는 것도 지장은 없다. 또, 컬러 화상이 디지털 콘텐츠로서 유통되는 형태를 생각하면, JPEG 등의 여러 가지의 압축기술을 이용하여 이를 압축하는 경우도 많기 때문에, 화상 데이터를 압축하는 소프트웨어에 일체 또는 플러그 인의 형태로 내장하는 것도 지장은 없다. 혹은, 인쇄된 화상 데이터를 디지털 데이터로서 판독입력하는 스캐너나 판독을 행하는 소프트웨어에 일체로 내장하는 것도 생각된다. 반대로, 디지털 데이터를 인쇄하는 프린터 드라이버나 프린터에, 내장해서 이용하는 것도 가능하다. 또한, 이들의 매립방법 및 복호방법은 장치로서 구현화하는 것도 용이하다. 이 경우에는 전용의 장치로서 실현하는 것도 가능하고, 상기의 기록매체로부터 프로그램을 판독입력함으로써 컴퓨터를 전자무늬의 매립장치 나 복호장치로서 이용하는 것도 가능하다.

### 도면의 간단한 설명

도 1은 본 발명의 일 실시예인 전자무늬정보의 매립 및 복호처리의 원리설명도.

도 2는 실시예로서의 전자무늬 매립장치(10)의 구성예를 나타내는 블록도.

도 3은 무늬 패턴 생성처리 루틴을 나타내는 플로우 차트.

도 4는 매립되는 무늬 패턴에 관해 설명하는 설명도.

도 5는 패턴에 의해 표현 가능한 정보의 종류를 나타내는 설명도.

도 6은 전자무늬 복호처리의 개요를 설명하는 플로우 차트.

도 7은 비트 정보 매립처리 루틴을 나타내는 플로우 차트.

도 8은 매립하는 비트 정보(s)와, 논리변수(D)와, 키 정보(K)로부터, 출력논리함수(E1, E2)를 구하는 테이블을 나타내는 설명도.

도 9는 무늬 복호처리 루틴을 나타내는 플로우 차트.

도 10은 매립 비트 복호처리 루틴을 나타내는 플로우 차트.

도 11은 복호용의 출력논리함수(EE)를 구하기 위한 테이블을 나타내는 설명도.

도 12는 저작권 관리정보의 복호처리 루틴을 나타내는 플로우 차트.

도 13은 실험예에서 이용한 기본 패턴을 나타내는 설명도.

도 14는 실험예에서 이용한 반복 패턴의 일례를 나타내는 설명도.

도 15는 실험예로서의 무늬 패턴을 매립한 화상의 일례를 나타내는 설명도.

도 16은 라플라시언(Laplacian) 공격된 후의 화상 데이터를 나타내는 설명도.

도 17은 라플라시언 공격된 후의 화상 데이터로부터 복호된 무늬 패턴의 일례를 나타내는 설명도.

실시예

<전자무늬 매립, 복호의 원리설명>

도 1은 실시예인 디지털 화상으로의 전자무늬의 매립방법과 그 처리 원리의 설명도이다. 원화상의 화상 데이터는 디지털 정보로서 보존되어 있고, 도시하는 바와 같이, 원화상에 대해 8×8화소를 하나의 단위로 하는 블록을 상정한다. 이는 JPEG 등의 화상 압축 등에서 널리 이용되고 있는 수법과 같다. 이 예에서 블록의 크기는 8×8 화소로 하고 있지만, 반드시 이 크기에 한정되는 것은 아니고, 다른 크기이더라도 지장은 없다.

다음에 이 원화상 데이터에 대해 색 변환의 처리를 행한다. 이 색 변환의 처리는 반드시 필요하지는 않지만, 통상 컴퓨터 상에서 취급되는 화상 데이터는 RGB 형식이고, 이를 그 후의 처리 상황에 맞추어 색차신호계(YCrCb) 등의 형식으로 변환하는 것이다. 이는 컬러 화상의 경우이고, 원화상이 모노톤(monotone)인 경우에는 휘도 신호만을 취급하게 되므로, 색변환 등의 처리는 필요하지 않다. 또한, 데이터 처리의 효율에 구애되지 않으면, 원화상의 형식대로 취급하는 것도 가능하다.

이 예에서는 컬러 화상을 YCrCb 형식으로 변환하고, 그 중의 휘도 신호를 나타내는 Y평면(plane)에 관해, 직교 변환을 행한다. 이 예에서, 직교 변환으로서 이산 코사인 변환(이하, DCT라고 한다)을 행하고 있지만, 다른 변환이라도 채용 가능하다. 이 DCT는 먼저 상정한 8 x 8화소를 단위로 하는 블록마다 행하여진다. 통상, DCT에 의해 얻어진 DCT 계수(8×8)에 대해, 이를 양자화 테이블로서 준비한 계수(양자화 레벨)로 계산(나눗셈)함으로써, 양자화된 DCT 계수(8×8)를 얻는다. 양자화 테이블로 계산(除算)하는 것은 데이터를 압축하기 위함이고, 전자무늬의 매립을 위해 원리적으로 필요하게 되는 것은 아니다. 또한, 양자화 테이블의 작성 방법에 따라 양자화의 조도(거칠음)를 설정할 수 있지만, 전자무늬의 매립에 의해 화질이 거칠어지는 것은 반드시 바람직하지 않기 때문에, 양자화 테이블을 이용하는 경우에는 화질의 저하가 작은(압축율이 낮은) 테이블을 채용하는 것도 바람직하다.

이렇게 해서 8×8화소를 한개의 블록으로 하여 DCT가 행하여지고, 양자화된 DCT 계수가 얻어진다. 다음에, 인접한 두개의 블록에 관해 얻어진 이 DCT 계수를 비교함으로써, 비트 정보를 매립하는 처리를 행한다. DCT 계수가 비교되는 두개의 블록은 처리의 단순함과 원화상의 연속성이라는 관점에서부터, 통상은 인접한 블록이 선택되지만, 반드시 인접한 블록에 한정되는 것은 아니고, 또한 가로 방향으로 인접하는 블록 대신에 세로 방향으로 인접하는 블록을 채용하는 것도 지장은 없다. 또한, 세개 이상의 블록의 DCT 계수를 이용하여 비트 정보를 매립하는 것으로 하는 것도 가능하다. 여기서 매립되는 비트 정보는 미리 기본 패턴을 생성해 두고, 이 기본 패턴으로부터 구해진다. 비트 정보의 근원이 되는 기본 패턴의 생성방법과 비트 정보의 매립에 대한 상세한 내용에 관해서는 후술한다.

이렇게 해서 비트 정보가 매립된 각 블록에 대해, 다음에 역 이산 코사인 변환(이하, IDCT라고 한다)을 행한다. 이 때, DCT에 이용한 것과 동일한 양자화 테이블을 이용한다. 역 변환에 의해 원래의 색차신호계(YCrCb)의 화상 데이터로 변환된다. 또, 양자화 테이블을 이용하면, 통상은 화상 데이터의 고주파성분이 일부 소실되어 버리게 되고 화질은 열화한다. 그 후, 역 색변환을 행하여 원래의 RGB계의 화상으로 복원할 수 있다. 이렇게 해서 기본 패턴을 전자무늬로 하고, 이것이 매립된 화상을 얻을 수 있다.

이상의 설명에서, 화상은 원래의 형식(여기서는 RGB 표색계)으로 복원했지만, 반드시 원래의 형식으로 복원할 필요는 없고, 예컨대 JPEG 등의 형식으로 보존한 채로 해도 좋다. 요컨대, 화상을 JPEG 등의 형식으로 보존할 때에, 미리 준비한 무늬 정보에 대응한 기본 패턴을 매립하는 것으로 해도 좋다.

이렇게 해서 얻어진 화상 데이터로부터, 무늬 정보를 판독출력하는 수법은 상기한 절차를 역으로 거치게(따르게) 된다. 즉, 먼저 화상 데이터로부터 두개의 블록에 대한 양자화된 DCT 계수를 구하고, 양 계수의 관계를 조사하여, 여기에 매립된 비트 정보를 추출하고, 이를 배열함으로써 기본 패턴을 복원하는 것이다. 이 수법에 관해서도 나중에 자세히 설명한다.

<장치의 전체구성>

다음에, 실시예에서 이용한 장치구성에 관해 간단히 설명한다. 도 2는 실시예로서의 전자무늬 매립방법을 실시하는 전자무늬 매립장치(10)의 하드웨어 구성을 나타내는 블록도이다. 도시한 바와 같이, 이 전자무늬 매립장치(10)는 컴퓨터(20)와, 이에 접속된 스캐너(12), CD-ROM 등의 외부기억장치(14), 모뎀(16) 및 표시용의 모니터(18)로 구성되어 있다. 컴퓨터(20)는 처리를 실행하는 CPU(22)와, 모니터 프로그램 등을 기억한 ROM(23)과, 프로그램이 로드되거나 연산중인 데이터 등이 일시적으로 기억되는 RAM(24)과, 모니터(18)에 화상을 표시하기 위한 프레임 메모리(25)와, 외부기억장치(14)와의 데이터의 교환을 담당하는 컨트롤러(H DC)(26)와, 모뎀(16)과의 데이터의 교환을 담당하는 시리얼 입출력회로(SIO)(28)

와, 스캐너(12)로부터 화상을 판독입력하는 입력 인터페이스(입력 IF)(29) 등으로 구성되어 있다. 컴퓨터에는, 키보드나 마우스 등도 접속되어 있지만, 이들의 도시는 생략하였다. 모뎀(16)은 통신회선(NT)을 거쳐 컴퓨터 네트워크에 접속되어 있다. 컴퓨터(20)는 이 모뎀(16)을 거쳐, 컴퓨터 네트워크의 도시하지 않은 서버로부터, 화상처리 프로그램을 다운 로드하는 것도 가능하다. 또한, 이 네트워크를 거쳐, 전자무늬를 매립한 화상 데이터를 배신(配信)한다고 하는 것도 가능하다.

CPU(22)는 RAM(24) 상에 로드된 프로그램을 실행함으로써, 블록 절출부(31), 변환부(32), 비트 정보 매립부(33), 출력부(34) 등의 기능을 실현하고 있다. 이들 각부는 청구항의 블록 절출수단, 변환수단, 비트 정보 매립수단, 출력수단에 각각 대응하고 있지만, 각부의 기능에 관해서는 CPU(22)가 실행하는 처리로서 나중에 일괄해서 설명한다.

다음에, 전자무늬의 매립방법에 관해서 설명한다. 화상 데이터에 전자무늬를 매립하기 전에, 먼저 매립하고자 하는 무늬 패턴을 생성한다. 도 3은 이 무늬 패턴의 생성처리 루틴을 나타내는 설명도이다. 이 루틴을 기동하면, 먼저 저작권정보를 입력하고, 이를 10진수로 전개하는 처리를 행한다(스텝 S50). 여기서 저작권정보란, 매립하려고 하고 있는 화상의 권리자를 나타내는 정보이고, 이 실시예에서는 한 사람의 저작자에 대해 하나의 코드 정보를 설정하는 것으로 하고 있다. 하나의 작품에 관해서 하나의 코드를 할당하더라도 지장은 없다.

저작자마다 고유의 10진수를 전개하는 처리는 컴퓨터(20)내에서 자동적으로 실행하는 것으로 해도 좋다. 저작자가 스스로 원하는 코드를 입력하는 것으로 해도 좋다. 또, 저작자정보의 일원적인 관리라는 점에서는 예컨대 저작자명을 컴퓨터(20)에 입력하면, 모뎀(16)을 거쳐 저작권 처리센터라고 하는 총괄적인 관리센터로 필요한 정보를 송신하고, 이 센터로부터 저작자마다 혹은 작품마다 고유의 코드의 배신을 받는 것으로 해도 좋다.

삭제

다음에, 이렇게 해서 얻어진 10진수에 의거하여, 기본 패턴을 생성하는 처리를 행한다(스텝 S60). 본 실시예에서는 전자무늬를 단순히 비트 정보로서 매립하는 것이 아니라, 이것을 임의의 패턴으로서 매립하고 있다. 여기서는 농도 패턴법(특히 나카무라의 방법)을 채용하여, 10진수로부터 기본 패턴을 생성하고 있다. 무늬 정보를 문자나 숫자 비트 계열로서 화상에 그 대로의 모습으로 매립하면, 용장성이 없기 때문에 공격을 받은 경우에 복호 오류가 많아진다. 그것을 막기 위해서 무늬에 오류정정 부호 등을 적용하는 대응책도 있지만, 여기서는 패턴의 용장성을 이용하고 있는 것이다.

기본 패턴은 다음과 같이 하여 생성된다. 먼저, 소정의 크기(n)(이 예에서는 도 4에 나타내는 바와 같이 4×4, 즉 n= 4)로 이루어지는 2차 도형에 어느 정도의 정보를 실을 수 있는 가라고 하는 점에 관해 설명한다. 도 4에 나타내는 4×4의 예에서는 기본 패턴이 16개의 무늬 신호{s(u), (u= 0, 1, …15)}로 구성되어 있다. 화상신호{s(u)}는 0 또는 1의 값을 취하는 2차 데이터이다. 또한, s(u)= 1의 개수를 m으로 나타낸다. 다만, 이 m은 2차 패턴을, m-out-of-n2의 용장부호계를 이용하여 데이터를 대응시키기 위해 도입한다. 이 때, 매립할 무늬 신호(S) 위의 n2개의 각 요소를, 조합의 수로 가중(加重)된 2진수로 간주하고,

수학식 1

$$n2 \prod_{m=0}^{n2-1} m-1 + n2-1 \prod_{m=0}^{n2-1} m$$

의 관계를 이용하면, 재귀적(再歸的)으로 s(u)의 각 요소의 값을 결정할 수 있다. 또 여기서, 식 「a!nb」는 a개의 요소로부터 b개의 요소를 추출할 경우의 조합의 수를 나타내는 것으로 한다(이하 동일).

먼저 u= 0의 경우를 생각하면, 수학식 1의 우변(右邊)의 제 1항은 s(0) ←1로 되는 배열의 총수를, 제 2항은 s(0) ←0으로 되는 배열의 총수를 각각 나타내고 있다. 따라서, s(0) = 1인 배열을, 각각

$$\{0,1,2, \dots, m2-1 \prod_{m=0}^{n2-1} m-1\}$$

의 각 정수(整數)에, s(0) = 0인 배열을 각각

$$\{n2-1 \prod_{m=0}^{n2-1} m-1, n2-1 \prod_{m=0}^{n2-1} m-1+1, \dots, m2 \prod_{m=0}^{n2-1} m\}$$

의 각 정수에 대응시킬 수 있다. 이 규칙을 이용하고,

$$bm = \log_2(n2 \prod_{m=0}^{n2-1} m)$$

의 저작권 관리정보(t)(10진수)와,  $n2-1 \ll m-1$ 을 비교하여

$t < n2-1 \ll m-1$ 이면,  $s(u) \leftarrow 1$

$t < n2-1 \ll m-1$ 이 아니면,  $s(u) \leftarrow 0$

으로 한다.

만약  $s(0) = 1$ 이면,  $s(0)$  이외의  $s(u)$  ( $u = 1, 2, \dots, n2-1$ )에 나머지  $m-1$ 개의 신호 1을 배치하기 위해,  $m$ 의 값을 1만큼 저감하여, 또  $t$ 와  $n2-2 \ll m-1$ 를 비교한다. 역으로,  $s(0) = 0$ 이면,  $s(0)$  이외의  $s(u)$  ( $u = 1, 2, \dots, n2-1$ )에 나머지  $m$ 개의 신호 1을 배치하기 위해,  $t$ 의 값을  $n2-1 \ll m-1$ 만큼 저감하여, 또  $t$ 와  $n2-2 \ll m-1$ 을 비교한다. 이 조작을,  $m = 0$ 으로 될 때까지 반복한다. 이 순서에 의해, 기본 패턴 중에서 1의 값으로 할 위치는 전부 결정되기 때문에, 나머지의  $s(u)$ 에는 0의 값을 매립한다. 이상의 순서에 의해, 주어진  $m$ 과  $t$ 에 1대 1로 대응하는 무늬 패턴(S)을 구할 수 있다.

도 5에 이 순서로 표현 가능한 기본 패턴(S)의 종류 수를 나타내었다.  $m$ 을 7 내지 9의 값으로 하면, 1개의 기본 패턴으로 13비트 분의 10진수 정수( $t$ )를 표현 가능한 것이 이해된다. 이 실시예에서 이용한 농도 패턴과 같이, 용장성이 높은 패턴을 이용하고 있는 경우에는 이 용장성을 살린 오류검출 등의 다른 처리를 내장(편입)하는 것이 가능해진다. 농도 패턴의 경우, 같은 농도가 되는 패턴은 복수 존재한다. 따라서, 같은 농도의 패턴을 이용하는 경우라도, 그 중에서 온(on)(흑)으로 하는 비트의 배치는 높은 자유도로 설정할 수 있다. 그래서, 예컨대 같은 농도이면서, 1래스터(raster)분의 화소(x 방향의 총 화소)중의 온(흑)화소의 수를 미리 짝수개로 결정해 둔다고 하는 사용방법이 가능해진다. 또한, 화질로의 영향을 배려하여, 어느 패턴을 이용하는 가를 결정한다고 한 사용방법도 가능하다. 본 실시예에서는 상술한 바와 같이, 10진수를 표현할 수 있는 크기의 농도 패턴을 이용하고 있지만, 이와 같이  $t$ 의 값이  $m$ -out-of- $n2$ 의 용장 시스템으로서 표현 가능한 범위에 있으면, 1개의 기본 패턴으로 충분하다. 관리해야 할 저작자 정보가 이 범위를 넘는 경우에는 이상의 기본 패턴을 2이상으로 나누어 매립하는 것으로 하면 좋다.

본 실시예에서는 10진수가 표현 가능하므로, 도 3의 스텝(S60)에서는 스텝(S

50)에서 얻어진 10진수에 대응시켜, 상기한 기본 패턴중의 하나를 선택하고, 이를 생성하는 것이다. 계속해서, 이 기본 패턴을 화상의 전역에 확대하고, 무늬 패턴을 생성하는 처리를 행한다(스텝 S70). 이 처리는 다음과 같이 행한다.

무늬 패턴을 매립하고자 하고 있는 원화상(P)이 가령 256화소×256화소로 이루어진다고 한다. 이 때, 원화상(P) 중에 8×8화소의 블록(Bu)은 32×32개 존재한다. 이들의 블록 중 인접하는 Bu, Bu+1을 한 조로 하면, 이 조는 32×16조 존재하게 된다. 이 각 조에  $s(u)$ 를 1비트씩 사상(寫像)하고 있기 때문에 원화상(P) 중에는 4×4의 기본 패턴이 8×4영역분 확보된 것으로 된다. 만약에 저작권 관리정보를 표현할 수 있는 무늬 기본 패턴이 1개로 충분하면, 화상(P)의 내부에 무늬 정보(S)를, 가로 방향으로 4회, 세로 방향으로 8회, 합계 32회 중복하여 기록할 수 있게 된다. 이와 같이, 기본 패턴을 반복 무늬 패턴으로 하고 있는 것은 무늬 패턴의 용장성을 높여, 무늬를 매립한 화상 데이터에 대한 라플라시언 공격이나 일부의 오류냄 이용에 대해 강한 내성을 실현하기 위해서이다. 또, 저작권 관리정보가 많으므로 기본 패턴(S)이 2개 이상 필요한 경우에는 서로 식별 가능하도록  $m$ 의 값을 변경한 기본 패턴(S)을 2개 이상 작성하고, 화상이 오려내어질 가능성을 고려하여, 8×4개의 영역에 복수개의 기본 패턴(S)을 랜덤하게 배분하면 좋다. 이와 같이하면, 10진수 뿐 만 아니라, 문자나 문자열을 매립하는 것도 가능하다. 물론, 보다 큰 화상에 관해서는 기본 패턴의 반복 횟수를 크게 하고, 또한 저작권 관리정보를 나타내는 자세한 문자 정보 등을 매립하는 것도 가능하다.

이상으로 무늬 패턴이 생성된 것으로 된다. 그래서, 다음에 이 무늬 패턴을 화상 데이터에 매립하는 수법에 관해서 자세하게 설명한다. 도 6은 무늬 매립처리의 개요를 나타내는 플로우 차트이다. 도시하는 바와 같이, 이 처리가 개시되면, 먼저 원화상 데이터를 판독입력하는 처리를 행한다(스텝 S100). 다음에 판독입력한 원화상 데이터를 색 변환하는 처리를 행한다(스텝 S110). 이미 설명한 바와 같이, 이 처리는 컴퓨터(20) 상에서 화상을 취급하는 일반적인 신호계인 RGB의 신호를, 색차신호계인 YCrCb의 형식으로 변경하는 처리이다. 원화상이 컬러 신호이면, 이 처리에 의해, 휘도신호(Y), 색차신호(Cr, Cb)의 3개의 평면으로 분할된 화상 데이터가 얻어진다.

이 3개의 평면중, 휘도 신호에 상당하는 Y평면으로부터 8×8화소를 단위로 하는 블록을 절출하는 처리를 행한다(스텝 S120). 원화상이 예컨대 256×256 화소로 이루어지는 경우, 그 왼쪽 위 모퉁이를 원점으로 하여, 수평방향(x)으로 순차 32개의 블록을 절출하고, 끝에 이르면, 수직방향(y)으로 어긋나게 하여 다음의 32개의 블록을 절출하며, 이 처리를 32회 반

복합으로써, 32×32개의 블록을 원화상으로부터 절출하는 것이다. 이렇게 해서 절출한 Y평면의 각 블록에 관해, 이산 코사인 변환(DCT)을 행한다. DCT의 상세는 주지의 것이기 때문에 여기서는 설명하지 않는다. 본 실시예에서, JPEG의 휘도 성분용의 양자화 테이블을 이용하여 DCT 계수를 양자화 하였다.

다음에, 먼저 설정한 무늬 패턴의 기본 패턴에 따라 결정한 비트 정보를, 이 블록의 DCT 계수에 매립하는 처리를 행한다(스텝 S150). 이 처리에 관해서는 나중에 도 7을 이용하여 설명한다. 비트 정보의 매립을 행한 후, 32×32개의 블록에 대해, 이번은 역이산 코사인 변환(IDCT)을 행하고(스텝 S170), 다시 역 색변환을 행하며(스텝 S180), 원래의 RGB계의 화상 데이터로 복원하고 나서, 화상을 외부로 출력하는 처리를 행한다(스텝 S190). 화상의 출력은 컬러 프린터 등을 이용하여 인쇄해도 좋고, 모뎀(16)을 이용하여 네트워크 상의 사용자에게 배신해도 좋다. 또, 이 실시예에서 무늬 패턴을 매립한 화상은 RGB 형식으로 복원하고 있지만, JPEG 형식으로 압축한 채로 출력해도 좋다.

다음에, 비트 정보 매립처리(스텝 S150)의 상세에 관해 설명한다. 비트 정보 매립처리 루틴에서는 도 7에 도시하는 바와 같이, 이 처리가 기동되면, 먼저 Y평면의 서로 인접하는 u번째와 u+1번째의 블록에 대한 DCT 계수{ $D_i(u, v)$  와  $D_{i+1}(u, v)$ }를 입력한다(스텝 S151). 다음에, 이것을 양자화 테이블을 이용하여 제산하여 양자화를 행하고, 주파수 계수영역(F)을 구하며, 이 F의 구성요소{ $f_u(i, j)$ ,  $f_{u+1}(i, j)$ }를 구하는 처리를 행한다(스텝 S152). 여기서,  $i, j = 0, 1, \dots, 7$ 이다. 계속해서, 같은 어드레스(i, j)의 계수의 크기의 차의 절대값{ $d(i, j)$ }을 구하는 처리를 행한다(스텝 S153).

다음에, 키 정보(K)를 결정하는 처리를 행한다(스텝 S154). 키 정보(K)는 이하에 설명하는 비트 정보의 매립에서, 매립의 룰을 비닉하기 위한 것이고, 미리  $K = \{k(i, j) \mid i, j=0, 1, 2, \dots, 7\}$ 로서 주어지고, 비밀로 되는 것이다. 한편, 이 실시예에서  $k(i, j)$ 는 0 또는 1의 값을 취하는 것으로 하고 있다.

다음에 매립하고자 하고 있는 비트 정보{ $s(u)$ }를, 무늬 패턴(S)으로부터 추출하는 처리를 행한다(스텝 S155). 비트 정보는 이 실시예에서 인접하는 두 개의 블록에서 단 하나이다. 먼저 설명한 무늬 패턴(도 3 참조)으로부터, 순차 어느 블록의 조에 어느 비트 정보를 매립할지가 결정된다. 또, 논리변수(D)를 연산하는 처리를 행한다(스텝 S156). 논리변수(D)는 이하의 조건에 의해 결정된다. 즉,  $d(i, j)$ 가 a의 값 이상이면  $D = 1$ 로 하고,  $d(i, j)$ 가 a의 값 미만이면  $D = 0$ 로 하는 것이다. 또, 이 a의 값은 무늬의 내성강도를 나타내는 값이고, 일반적으로 클수록, 외부로부터의 공격에 대해 무늬는 강한 내성을 나타낸다. 그러나, 너무 큰 값이 되면, 화질에 영향을 주기 쉬워진다.

이상으로, 두개의 블록에 관해 그 양자화된 DCT 계수의 차의 절대값에 의거하는 논리변수(D)와, 키 정보(K)와, 매립하고자 하고 있는 비트 정보{ $s(u)$ }가 결정된 것으로 된다. 그래서, 다음에 이들 3개의 정보로부터, 출력논리함수(E1)를 연산한다(스텝 S161). 이 출력논리함수(E1)는 논리 테이블로서 나타내면, 도 8과 같이 되지만, 논리식으로서서는,

$$E1 = S \setminus \cdot D \setminus \cdot K \setminus + S \cdot D \setminus \cdot K$$

이다. 또, 부호「\」는 직전(直前)의 변수가 부(負)논리인 것을 나타내고 있다.

출력논리함수(E1)가 1의 값인지 아닌지를 판단하고(스텝 S162),  $E1 = 1$ 이 아니면 아무것도 행하지 않고,  $E1 = 1$ 이면, 스텝(S163)에서 계수{ $f_u(i, j)$ 와  $f_{u+1}(i, j)$ }가 큰 쪽에, 소정값{ $e(i, j)$ }을 가산하는 처리를 행한다. 또, 이 소정값{ $e(i, j)$ }은 통상, 상술한 무늬의 내성강도를 나타내는 값(a)과 동일하게 설정된다.

다음에, 출력논리함수(E2)를 연산하는 처리를 행한다(스텝 S165). 이것은 도 8의 란(欄)(E2)을 구하는 것과 같고, 그 논리식은,

$$E2 = S \setminus \cdot D \cdot K + S \cdot D \cdot K \setminus$$

이다. 그 후, 출력논리함수(E1)와 마찬가지로 그 값이 1인지 아닌지를 판별하고(스텝 S166), 1의 값이면, 스텝(S167)에서, 계수{ $f_u(i, j)$  와  $f_{u+1}(i, j)$ }가 큰 쪽에서 소정값{ $e(i, j)$ }을 감산하는 처리를 행한다. 그 후, 이상의 처리를 원화상의 모든 블록에 관해 행했는지의 여부를 판단하고(스텝 S169), 모든 블록에 관해 완료할 때까지 반복한다. 원화상이 256×256 화소의 화상인 경우, 2개의 인접하는 블록을 조로 하여 상기의 처리를 행하기 때문에, 32×16회의 처리가 반복되게 된다. 이 처리의 과정에서, 4×4의 기본 패턴이 4×8회 반복 매립되는 것은 이미 설명한 대로이다.

이상, 원리도인 도 1, 무늬 패턴의 매립처리의 개요를 나타내는 도 6, 비트 정보 매립처리 루틴의 상세를 나타내는 도 7을 이용하여 설명한 바와 같이, 본 실시예에 의하면, 소정의 관계(실시예에서는 x방향의 인접관계)를 갖는 두개의 블록의 이산 코사인 변환(DCT)되고 양자화된 계수를 이용하여, 이 계수의 크기에 소정의 차가 있을 때, 이것에 비트 정보{ $s(u)$ }를,

키 정보(K)를 참조하면서, 매립하고 있다. 즉, 이 방법에 의하면, 블록의 조의 양자화된 계수를 사용하여, 1비트의 무늬 신호  $\{s(u)\}$ 를 주파수영역의 요소 쌍  $\{f_u(i, j), f_{u+1}(i, j)\}$ 의 차분값으로서 용장성을 갖게 하여 매립하고 있는 것이다. 더구나, 무늬 패턴을 기본 패턴의 반복으로서 구성하고, 말하자면 무늬 정보를 2중구조로 하고 있기 때문에, 풍부한 용장성을 도입하는 것이 가능해지고 있다. 따라서, 오러냄이나 라플라시언 공격에 대한 강한 내성을 나타낸다. 또한, 이 내성강도를 소정 값  $\{e(i, j)\}$ 에 의해 용이하게 설정할 수 있다. 이러한 공격에 대한 내성은 말미에 실험예로서 나타내었다. 또, 주파수영역의 요소 쌍  $\{f_u(i, j), f_{u+1}(i, j)\}$ 을 변경하고 있기 때문에, 화상은 전주파수 대역에 걸쳐 미소한 주파수성분의 변동을 받고, 얼마되지 않지만 그 화질은 열화한다. 따라서, 고품질의 출력화상을 필요로 하는 경우에는 강도 파라미터  $\{e(i, j) = a\}$ 의 값을 작게하거나, 비트 정보의 매립에서  $i$  및  $j$ 가 0의 값의 계수, 즉 직류성분이나, 또 일부의 저주파성분을 매립대상으로부터 제외하는 것도, 화질을 높게 유지한다고 하는 점에서 바람직하다. 이 경우에서도, 원래 비트 정보는 블록의 다수의 요소에 매립하고 있기 때문에,  $i, j = 0$ 의 몇 개의 요소로의 매립을 행하지 않은 것으로 해도, 특히 충분히 무늬를 복원할 수 있다.

또, 본 실시예의 키 정보(K)를 이용하는 것은 반드시 필수적이지 않지만, 이 실시예에서는 키 정보(K)를 이용하여, 매립 규칙의 비독화(비밀화)를 꾀하고 있기 때문에, 화상을 입수한 자가 부정하게 이 매립 정보를 추출하여, 이것을 파괴한다고 하는 것을 곤란하게 하고 있다.

다음에, 화상에 매립된 무늬를 복호하는 처리에 관해 설명한다. 도 9는 무늬복호처리 루틴을 나타내는 플로우 차트이다. 또한, 도 10은 매립 비트의 복호처리의 상세를 나타내는 플로우 차트이다. 양 도면은, 무늬의 매립처리의 플로우 차트인 도 6 및 도 7에 대응하고 있으므로 간략히 설명한다.

이 루틴이 기동되면, 먼저 무늬를 복호하고자 하는 화상 데이터를 관독입력하고(스텝 S200), 이를 색 변환한다(스텝 S210). 그 후, Y평면으로부터  $8 \times 8$  화소의 블록을 절출하고(스텝 S220), 이들을 이산 코사인 변환(DCT)한다(스텝 S230). 이렇게 해서 얻어진 양자화된 DCT 계수로부터, 매립 비트를 복호하는 처리를 행하고(스텝 S250), 복호한 비트 정보로부터, 무늬의 기본 패턴을 복원하는 처리를 행한다(스텝 S270). 그 후, 복원한 무늬를 출력한다(스텝 S290).

스텝(S250)의 매립 비트의 복호처리는 도 10에 나타난 바와 같이, Y평면의 인접하는 블록의 DCT 계수  $\{D_i'(u, v)$  와  $D_{i+1}'(u, v)\}$ 를 입력한다(스텝 S251). 다음에, 이를 양자화 테이블을 이용하여 제산하고 양자화를 행하고, 주파수 계수영역(F')을 구하며, 이 F'의 구성요소  $\{f'u(i, j), f'_{u+1}(i, j)\}$ 를 구하는 처리를 행한다(스텝 S252). 계속해서, 같은 어드레스  $\{(i, j)\}$ 의 계수의 크기의 차의 절대값  $\{d'(i, j)\}$ 을 구하는 처리를 행한다(스텝 S253).

다음에, 키 정보(K)를 결정하는 처리를 행하고(스텝 S254), 상기의 절대값  $\{d'(i, j)\}$ 으로부터, 논리변수(D)를 구하는 처리를 행한다(스텝 S256). 논리변수(D

)는 이하의 조건에 의해 결정된다. 즉,  $d'(i, j)$ 가  $a$ 의 값 이상이면  $D = 1$ 로 하고,  $d'(i, j)$ 가  $a$ 의 값 미만이면  $D = 0$ 으로 하는 것이다.

이상으로, 두개의 블록에 관해 그 양자화된 DCT 계수의 차의 절대값에 의거하는 논리변수(D)와, 키 정보(K)가 구해진 것으로 된다. 그래서, 다음에 이들 2개의 정보로부터, 출력논리함수(EE)를 연산한다(스텝 S261). 이 출력논리함수(EE)는 논리 테이블로 나타내면, 도 11과 같이 되지만, 논리식으로서,

$$EE = D \cdot K + \bar{D} \cdot \bar{K}$$

이다.

출력논리함수(EE)가 1의 값인지 아닌지를 판단하고(스텝 S262),  $EE = 1$ 이면, 변수(g)를 1의 값만큼 인크리먼트(increment)하는 처리(스텝 S263)를, 출력논리함수( $EE = 0$ )가 아니면, 변수(z)를 1의 값만큼 인크리먼트하는 처리(스텝 S264)를 각각 행한다. 그 후, 이상의 처리를, 원화상의 모든 블록에 관해서 행했는지의 여부를 판단하고(스텝 S266), 모든 블록에 관해 완료할 때까지 반복한다.

전 블록에 관해, 이상의 처리가 완료되면, 다음에 변수(g 와 z)의 대소관계를 판단하고, 소위 다수결에 의해, 이 두개의 블록에 매립된 비트 정보  $\{s(u)\}$ 를 1의 값 또는 0의 값으로 결정한다(스텝 S268, 269).

이상의 처리에 의해, 용이하고 또한 안정되게 인접하는 두개의 블록에 매립된 비트 정보  $\{s(u)\}$ 를 취출할 수 있다. 그래서,  $u = 0, 1, 2, \dots, 15$ 에 대응한 비트 정보  $\{s(u)\}$ 가 전부 구해지고, 전자무늬로서의 기본 패턴이 확정되었으므로, 저작권의 관리정보를 복호하는 처리에 관해 설명한다. 도 12는 저작권 관리정보의 복호처리 루틴을 나타내는 플로우 차트이다. 도시하

는 바와 같이, 이 루틴이 기동되면, 먼저, 1 내지 16의 값을 취할 수 있는 변수(x, y)를 이용하여, 그 순열 조합의 수( $\pi(y)$ )를, 배열{M(x, y)}에 대입하는 처리를 행한다(스텝 S271). 다음에, x, y, u, t에 초기값을 대입하고(스텝 S272), 이하의 처리를 개시한다.

스텝(S273)에서는 초기값으로서 값(m)이 대입된 변수(y)가 0의 값으로 되어있는 지 어떤지를 판단한다. 이것은 탐색이 종료했는지 어떤지를 판단하기 위해서이다. 처리가 개시된 직후에는 변수(y)의 값은 0으로 되어 있지 않기 때문에, 처리는 스텝(S274) 이하로 진행하고, 먼저 비트 정보{s(u)}가 0의 값인지 아닌지의 판단이 행하여진다. 비트 정보{s(u)}가 0의 값이면, 저작권의 관리정보에 대응한 10진수에 배열{M(x, y-1)}을 가산하고(스텝 S275), s(u) = 0이 아니면 변수(y)를 1의 값만큼 디크리먼트(decrement)하는 처리를 행한다(스텝 S276). 어느 경우라도, 그 후 변수(x)를 1의 값만큼 디크리먼트하고, 변수(u)를 1의 값만큼 인크리먼트하는 처리를 행한다(스텝 S278).

그 후, 변수(u)가 15의 값보다 커졌는지 아닌지를 판단하고(스텝 S279), 커져 있으면, 변수(y)가 0의 값으로 된 경우와 함께, 스텝(S280)으로 이행하고, 관리정보(t)의 하위 비트를,  $\log_2 M(16, j)$ 로서 복호하는 처리를 행한다. 이상 설명한 순서에 의해 무늬의 기본 패턴에 매립되어 있던 수치 데이터, 즉 저작권 등의 관리정보(t)를 복호 할 수 있다. 본 실시예의 방법으로 매립된 무늬는 기본 패턴으로 되고, 또 이를 복수회 매립한 2중 구조로 되어 있기 때문에, 풍부한 용장성을 도입하는 것이 가능해져, 용이하고 또한 안정되게 이 무늬를 복호할 수 있다.

다음에 이 방법을 이용한 실험결과를 나타낸다. 상술한 실시예에 따라, 저작권 관리정보를 매립했다. 이 실험에서는 저작권 관리정보(T)로서 t = 123(10진수)을 이용했다. 이 값(t)을 표현하기 위해, 도 5에 나타난 결과를 감안하여, m = 3을 이용했다. 상술한 실시예의 수법에 의해 기본 패턴작성을 구한 결과를, 도 13에 나타내었다. 이 기본 패턴을 도 3에 나타낸 바와 같이, 2 블록에 1개의 비율로 256 × 256의 영역에 반복 배열한 경우의 무늬 패턴을, 도 14에 나타내었다.

이 2차 화상을 무늬 정보로서 농담화상(P)의 DCT된 주파수 계수영역에 매립했다. 그 결과를 도 15에 나타낸다. 이 무늬를 매립한 화상(P')에 대해 라플라시언 공격을 가한 결과를, 도 16에 나타내었다. 이 화상으로부터 무늬 정보를 복호한 결과를, 도 17에 나타내었다. 도시하는 바와 같이, 라플라시언 공격에 대해서도, 원래의 무늬 패턴은 대부분 보존되어 있는 것을 알 수 있다. 또한 일부의 패턴은 소실되어 있지만, 다수결 원리 등을 이용함으로써, 이 반복 패턴으로부터, 원래의 기본 패턴을 정확하게 복원하는 것은 용이하였다.

이상, 본 발명의 실시예와 그 실험결과에 관해 설명하였지만, 이 블록 절출부(31), 변환부(32), 비트 정보 매립부(33), 출력부(34) 등의 기능을 실현하는 컴퓨터 프로그램은 플렉시블 디스크나 CD-ROM 등의 컴퓨터 판독 가능한 기록매체에 기록된 형태로 제공된다. 컴퓨터는 그 기록매체로부터 컴퓨터 프로그램을 판독하여 내부기억장치 또는 외부기억장치에 전송한다. 혹은, 통신경로를 거쳐 컴퓨터에 컴퓨터 프로그램을 공급하도록 해도 좋다. 컴퓨터 프로그램의 기능을 실현할 때에는 내부기억장치에 격납된 컴퓨터 프로그램이 컴퓨터의 마이크로 프로세서에 의해 실행된다. 또한, 기록매체에 기록된 컴퓨터 프로그램을 컴퓨터가 판독하여 직접 실행하도록 해도 좋다.

본 명세서에서 컴퓨터란, 하드웨어 장치와 오퍼레이션 시스템을 포함하는 개념이고, 오퍼레이션 시스템의 제어하에서 동작하는 하드웨어 장치를 의미하고 있다. 또한, 오퍼레이션 시스템이 불필요하고 어플리케이션 프로그램 단독으로 하드웨어 장치를 동작시키는 바와 같은 경우에는 그 하드웨어 장치 자체가 컴퓨터에 상당한다. 하드웨어 장치는 CPU 등의 마이크로 프로세서와, 기록매체에 기록된 컴퓨터 프로그램을 판독하기 위한 수단을 적어도 구비하고 있다. 컴퓨터 프로그램은 이러한 컴퓨터에 상술한 각 수단의 기능을 실현시키는 프로그램 코드를 포함하고 있다. 또, 상술한 기능의 일부는 어플리케이션 프로그램이 아니고, 오퍼레이션 시스템에 의해 실현되어 있어도 좋다. 또, 전자무늬의 매립처리나 복호처리를 행하는 프로그램은 화상처리를 행하는 프로그램에 대해, 플러그 인의 형식으로 부가되는 것으로 해도 좋다.

또, 본 발명에서의 「기록매체」로서는 플렉시블 디스크나 CD-ROM, 광자기 디스크, IC 카드, ROM 카트리지, 펀치 카드, 바코드 등의 부호가 인쇄된 인쇄물, 컴퓨터의 내부기억장치(RAM이나 ROM 등의 메모리) 및 외부 기억장치 등의 컴퓨터가 판독 가능한 여러 가지의 매체를 이용할 수 있다.

이상 본 발명의 실시예에 관해 설명하였지만, 본 발명은 이러한 실시예에 하등 한정되는 것이 아니고, 다양한 형태로 실시할 수 있는 것은 물론이다.

### 산업상 이용 가능성

본 발명은 전자 데이터로의 무늬 정보의 매립장치나 전자인증 등의 장치로서 적용 가능하다. 이러한 장치는 저작권정보의 매립 등에도 적용할 수 있다. 또한, 암호장치나 암호의 매립, 송신, 복호 등의 분야에서도 이용할 수 있다.

(57) 청구의 범위

**청구항 1.**

화상에 전자무늬 정보를 매립하는 방법으로서,

원화상으로부터 소정의 크기의 블록을 절출(切出)하고,

그 각 블록에 대응한 화상 데이터에 대해 직교 변환을 행하며,

소정의 관계를 갖는 두개 이상의 블록의 상기 직교 변환후의 대응하는 계수를 비교하고, 전자무늬로 하는 비트 정보에 의해 그 계수의 한 쪽을 다른 쪽에 대해 소정의 대소관계로 함으로써, 그 비트 정보를 매립하고,

상기 비트 정보가 매립된 블록을 역 직교변환 함으로써, 전자무늬가 매립된 화상을 출력하는,

전자무늬의 매립방법.

**청구항 2.**

제 1항에 있어서,

상기 블록 사이의 소정의 관계는 인접 관계인, 전자무늬의 매립방법.

**청구항 3.**

제 1항에 있어서,

상기 직교변환은 이산 코사인 변환인, 전자무늬 매립방법.

**청구항 4.**

제 1항에 있어서,

상기 직교 변환후의 계수를 양자화 테이블을 이용하여 양자화하고, 상기 정보 비트의 매립을 그 양자화 된 계수를 이용하여 행하는, 전자무늬의 매립방법.

**청구항 5.**

제 4항에 있어서,

상기 화상 데이터를 상기 직교 변환에 앞서, 휘도(Y), 색차(Cb, Cr)로 변환하고,

그 휘도(Y), 색차(Cb, Cr)의 각각에 대해, 상기 직교 변환으로서 이산 코사인 변환을 행하며,

상기 휘도(Y)를 이산 코사인 변환하여 얻어진 계수에 대해, 상기 비트 정보의 매립을 행하는,  
전자무늬의 매립방법.

### 청구항 6.

제 4항에 있어서,

상기 양자화 후의 양 블록의 대응하는 계수가 공히 0의 값이 아닌 경우에는 비트 정보의 매립을 행하는, 전자무늬의 매립 방법.

### 청구항 7.

제1 항에 있어서,

상기 소정의 관계에 있는 두개 이상의 블록의 상기 직교 변환후의 계수의 차가 소정의 범위에 있는 경우를 진(眞)으로 하는 논리함수를 도입하고,

그 논리함수의 진위(眞僞)에 의거하여, 상기 비트 정보의 매립의 수법(手法)을 변경하는,

전자무늬의 매립방법.

### 청구항 8.

제 7항에 있어서,

상기 계수에 대응지어진 비밀 키(key)를 미리 준비하고,

상기 계수마다의 그 비밀 키와 상기 계수에 대한 논리함수의 진위의 상태에 의거하여, 상기 비트 정보의 매립의 수법을 변경하는, 전자무늬의 매립방법.

### 청구항 9.

제 1항에 있어서,

2치(値) 정보의 조합으로서 이차원적으로 정의된 기본 패턴을 전자무늬의 정보로서 준비하고,

그 준비된 기본 패턴을 구성하는 각 2치 정보를 상기 매립되는 비트 정보로 하며,

상기 소정의 관계에 있는 블록 끼리를 하나의 단위로 하여, 상기 기본 패턴을 구성하는 상기 2치 정보를 매립하는,

전자무늬의 매립방법.

### 청구항 10.

제 9항에 있어서,

상기 기본 패턴을 구성하는 요소의 수가 상기 절출된 블록수 보다 충분히 클 때, 그 기본 패턴을 상기 화상 데이터에 소정 횟수 반복하여 매립하는, 전자무늬의 매립방법.

### 청구항 11.

제 9항에 있어서,

상기 기본 패턴이 용장성(冗長性)을 갖는 패턴인, 전자무늬의 매립방법.

### 청구항 12.

제 11항에 있어서,

상기 용장성을 갖는 기본 패턴이 농도 패턴인, 전자무늬의 매립방법.

### 청구항 13.

전자무늬가 매립된 화상으로부터 전자무늬를 취출(取出)하는 복호방법으로서,

원화상으로부터 소정의 크기의 블록을 절출하고,

그 각 블록에 대응한 화상 데이터에 대해 직교 변환을 행하며,

소정의 관계를 갖는 두개 이상의 블록의 상기 직교 변환후의 대응하는 계수를 비교하고, 그 계수의 한 쪽의 다른 쪽에 대한 대소관계에 의거하여, 비트 정보를 취출하는,

전자무늬의 복호방법.

### 청구항 14.

제 13항에 있어서,

상기 블록 사이의 소정의 관계는 인접 관계인, 전자무늬의 복호방법.

### 청구항 15.

제 13항에 있어서,

상기 화상 데이터를 상기 직교 변환에 앞서, 휘도(Y), 색차(Cb, Cr)로 변환하고,

그 휘도(Y), 색차(Cb, Cr)의 각각에 대해, 상기 직교 변환으로서 이산 코사인 변환을 행하고,

상기 휘도(Y)를 이산 코사인 변환하여 얻어진 계수에 의거하여, 상기 비트 정보의 취출을 행하는,

전자무늬의 복호방법.

**청구항 16.**

제 13항에 있어서,  
 상기 추출된 비트 정보를 배열하고, 그 비트 정보로부터 기본 패턴을 복원하고,  
 그 기본 패턴으로부터 상기 무늬 정보를 추출하는,  
 전자무늬의 복호방법.

**청구항 17.**

화상에 전자무늬 정보를 매립하는 장치로서,  
 원화상으로부터 소정의 크기의 블록을 절출하는 블록 절출수단과,  
 그 각 블록에 대응한 화상 데이터에 대해 직교 변환을 행하는 변환수단과,  
 소정의 관계를 갖는 두개 이상의 블록의 상기 직교 변환후의 대응하는 계수를 비교하고, 전자무늬로 하는 비트 정보에 의  
 해, 그 계수의 한 쪽을 다른 쪽에 대해 소정의 대소관계로 함으로써, 그 비트 정보를 매립하는 비트 정보 매립수단과,  
 상기 비트 정보가 매립된 블록을 역 직교변환 함으로써, 전자무늬가 매립된 화상을 출력하는 출력수단  
 을 구비한, 전자무늬 매립장치.

**청구항 18.**

전자무늬가 매립된 화상으로부터 전자무늬를 복호하는 장치로서,  
 원화상으로부터 소정의 크기의 블록을 절출하는 블록 절출수단과,  
 그 절출된 각 블록에 대응한 화상 데이터에 대해 직교 변환을 행하는 변환수단과,  
 소정의 관계를 갖는 두개 이상의 블록의 상기 직교 변환후의 대응하는 계수를 비교하고, 그 계수의 한 쪽의 다른 쪽에 대한  
 대소관계에 의거하여, 비트 정보를 추출하는 비트 정보 추출수단  
 을 구비한, 전자무늬 복호장치.

**청구항 19.**

화상에 전자무늬 정보를 매립하는 프로그램을 컴퓨터에 의해 관독 가능하게 기록한 기록매체로서,  
 원화상으로부터 소정의 크기의 블록을 절출하는 기능과,  
 그 각 블록에 대응한 화상 데이터에 대해 직교 변환을 행하는 기능과,  
 소정의 관계를 갖는 두개 이상의 블록의 상기 직교 변환후의 대응하는 계수를 비교하고, 전자무늬로 하는 비트 정보에 의  
 해, 그 계수의 한 쪽을 다른 쪽에 대해 소정의 대소관계로 함으로써, 그 비트정보를 매립하는 기능과,  
 상기 비트 정보가 매립된 블록을 역 직교변환 함으로써, 전자무늬가 매립된 화상을 출력하는 기능

을 컴퓨터에 의해 실현하는 프로그램을 기록한, 기록매체.

## 청구항 20.

전자무늬가 매립된 화상으로부터, 그 전자무늬를 추출하는 프로그램을 컴퓨터에 의해 판독 가능하게 기록한 기록매체로서,

원화상으로부터 소정의 크기의 블록을 절출하는 기능과,

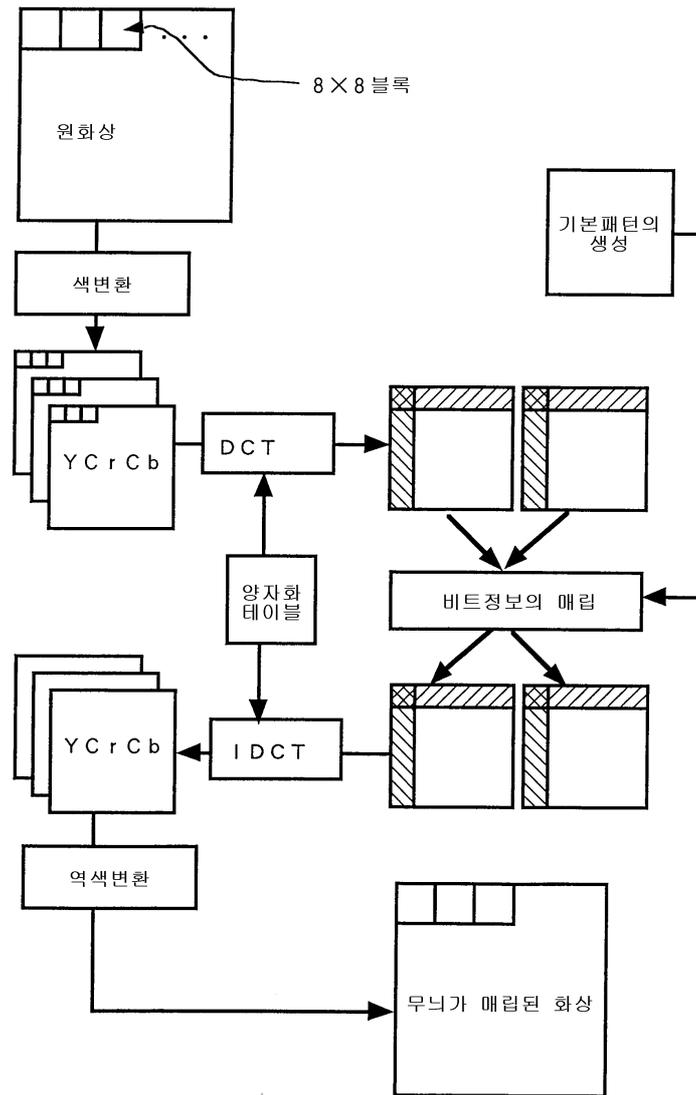
그 각 블록에 대응한 화상 데이터에 대해 직교 변환을 행하는 기능과,

소정의 관계를 갖는 두개 이상의 블록의 상기 직교 변환후의 대응하는 계수를 비교하고, 그 계수의 한 쪽의 다른 쪽에 대한 대소관계에 의거하여, 비트 정보를 추출하는 기능

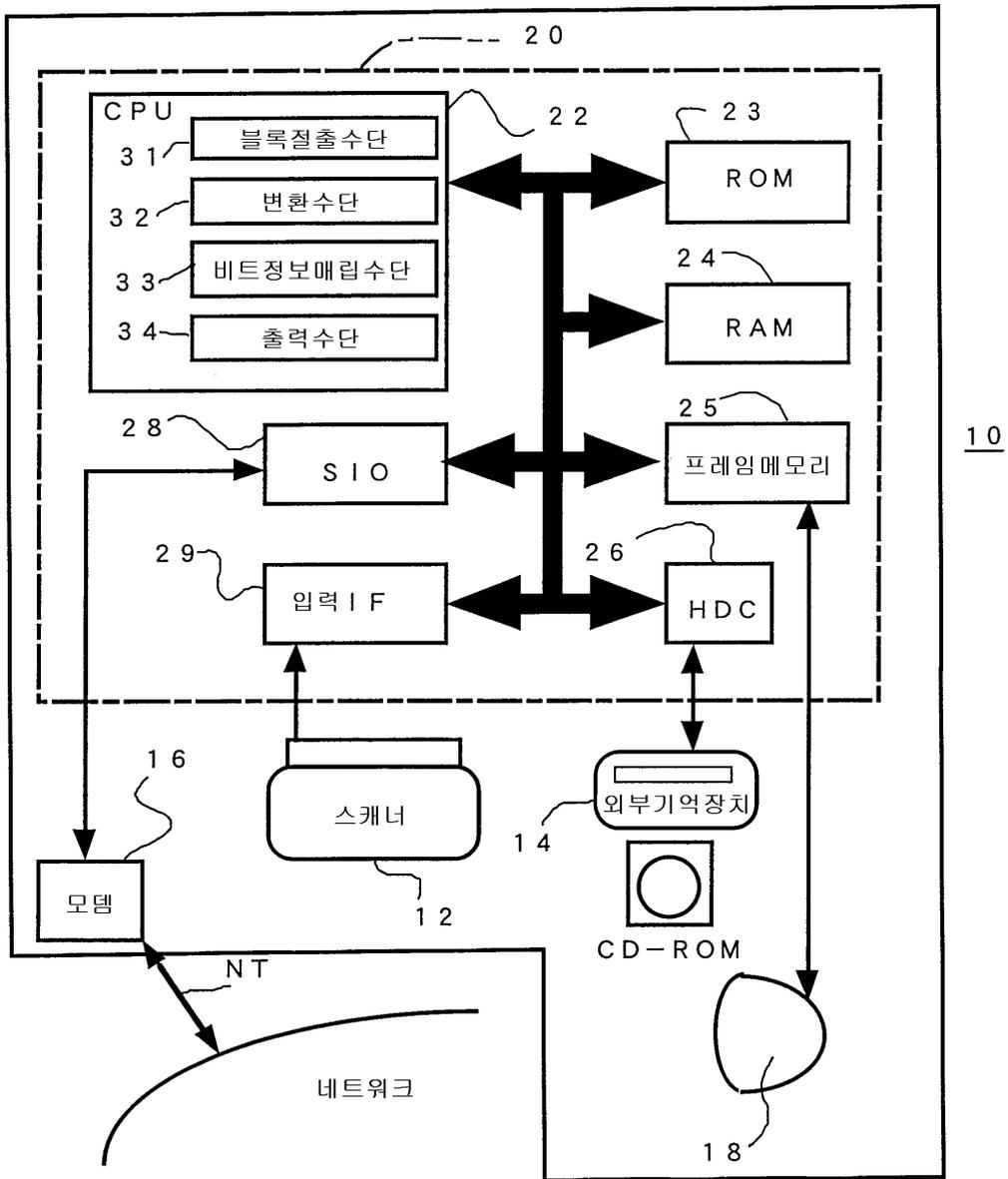
을 컴퓨터에 의해 실현하는 프로그램을 기록한, 기록매체.

도면

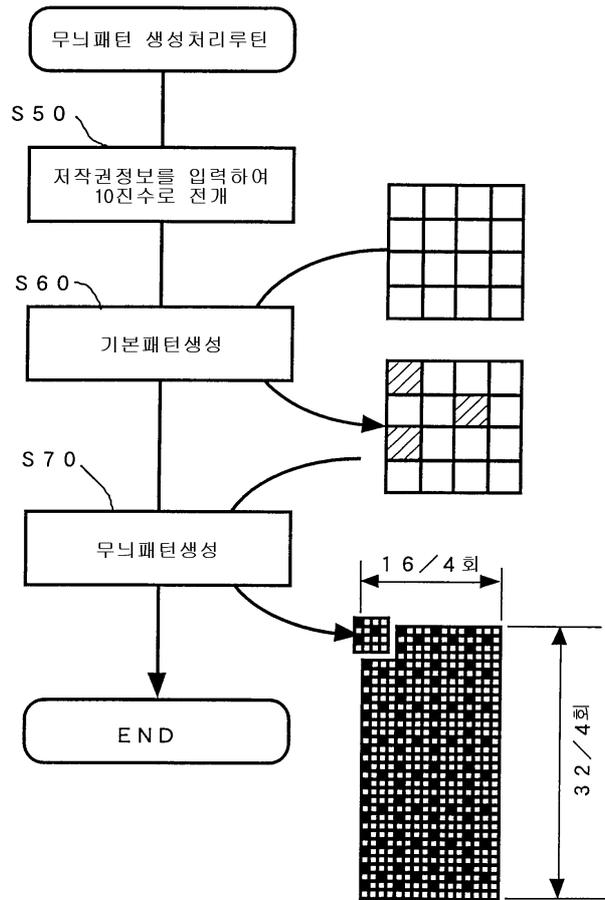
도면1



도면2



도면3



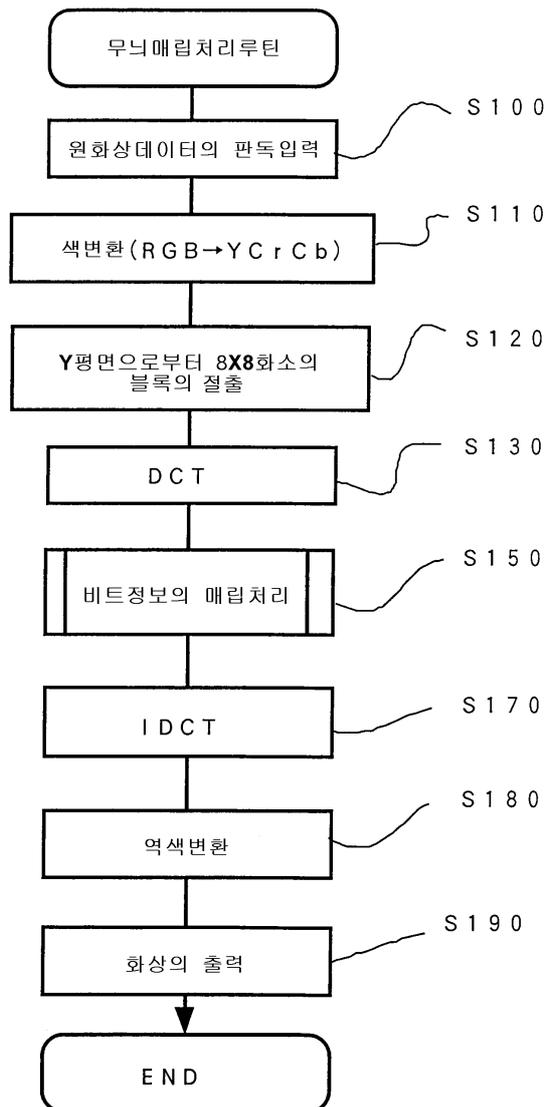
도면4

s (0)	s (1)	s (2)	s (3)
s (4)	s (5)	s (6)	s (7)
s (8)	s (9)	s (10)	s (11)
s (12)	s (13)	s (14)	s (15)

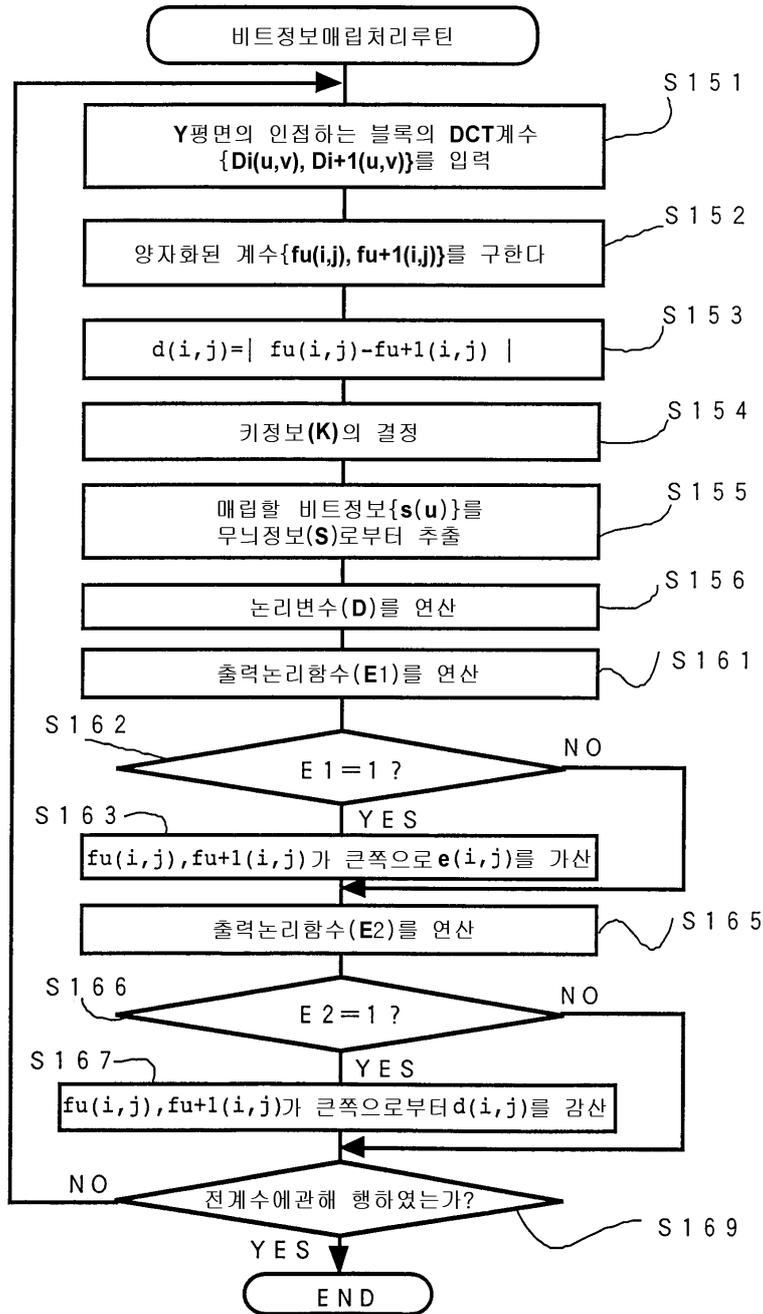
도면5

m	16Tm	b <sub>m</sub>
1	1 6	4
2	1 2 0	6
3	5 6 0	9
4	1 8 2 0	1 0
5	4 3 6 8	1 2
6	8 0 0 8	1 2
7	1 1 4 4 0	1 3
8	1 2 8 7 0	1 3
9	1 1 4 4 0	1 3
1 0	8 0 0 8	1 2
1 1	4 3 6 8	1 2
1 2	1 8 2 0	1 0
1 3	5 6 0	9
1 4	1 2 0	6
1 5	1 6	4
1 6	1	0

도면6



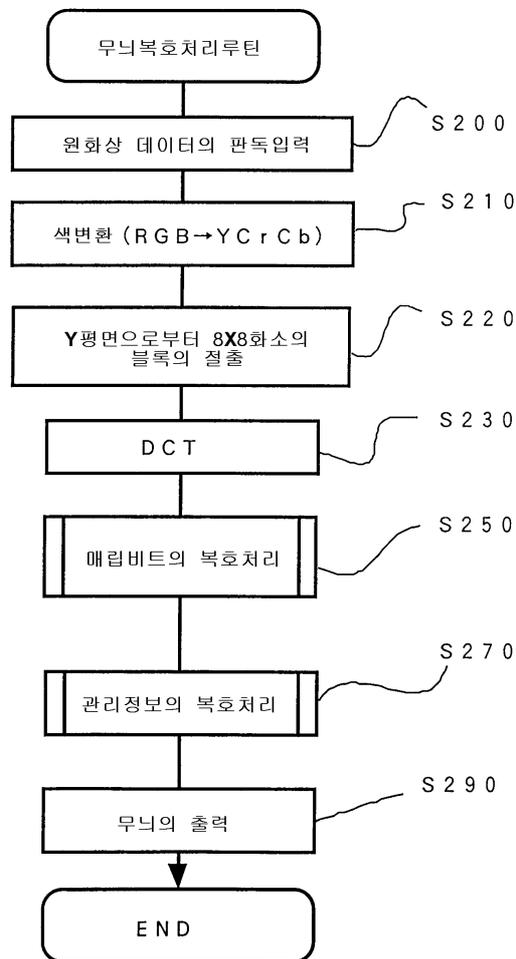
도면7



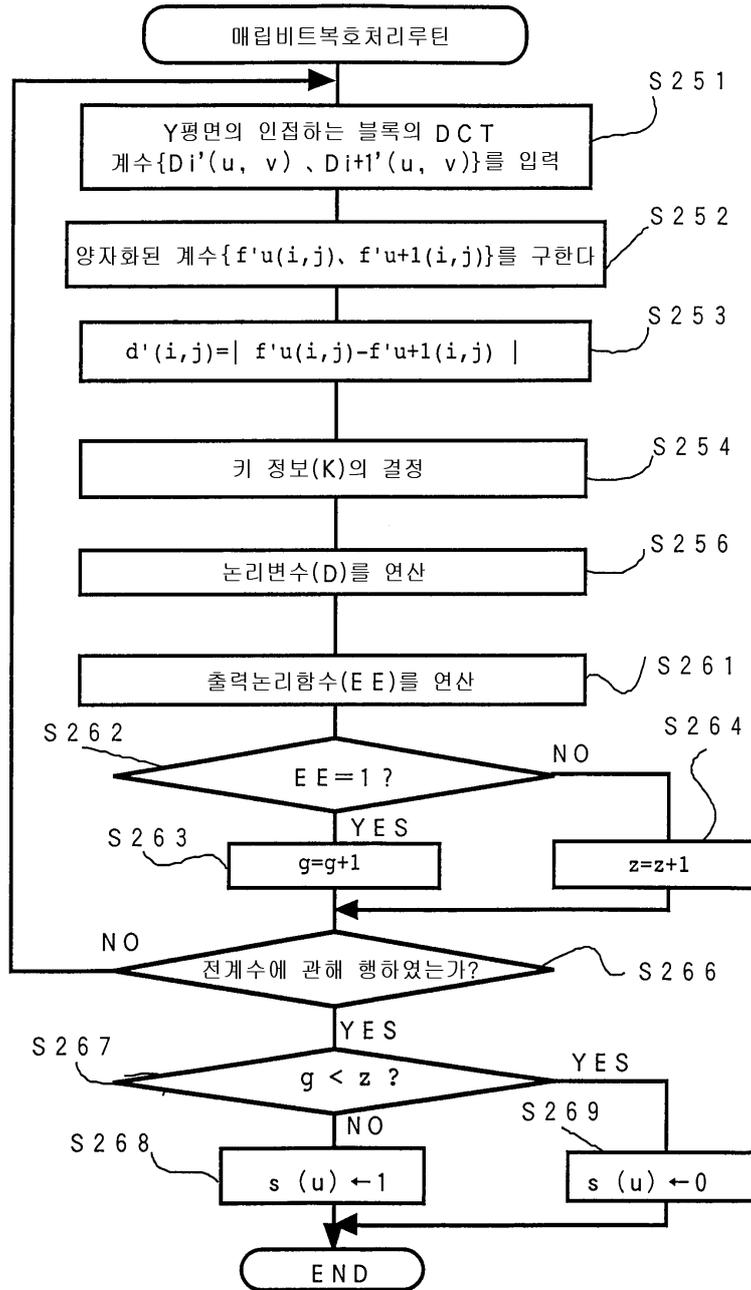
도면8

무늬 차분		S				출력 논리함수
		0		1		
D	0	1	0	1	0	E 1
	1	0	1	0	1	E 2
키	0	1		0	K	

도면9



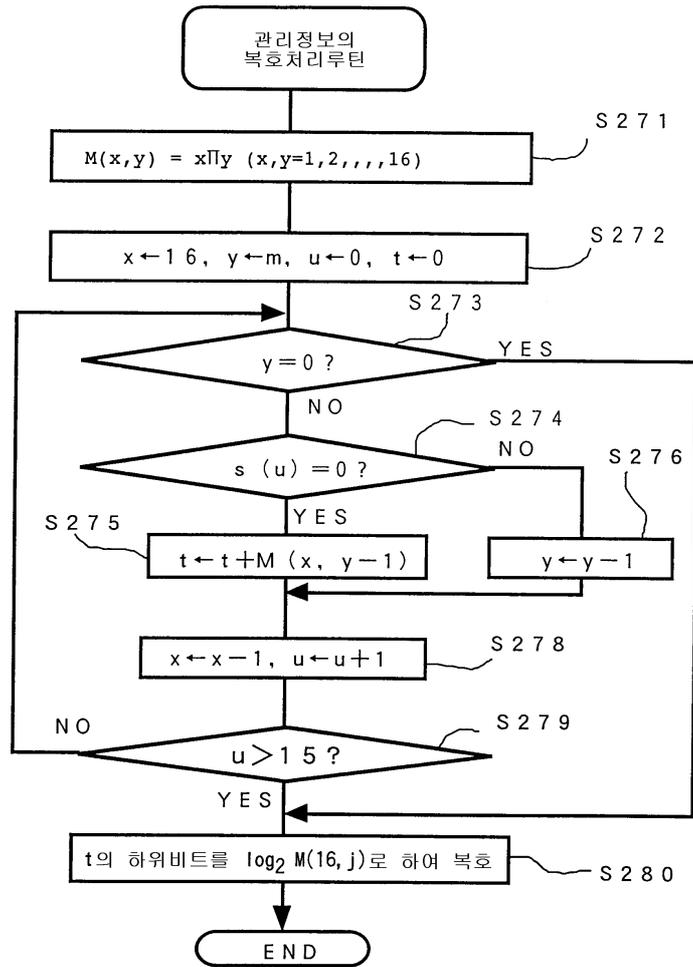
도면10



도면11

		키	
		K	
차	0	1	0
	1	0	1

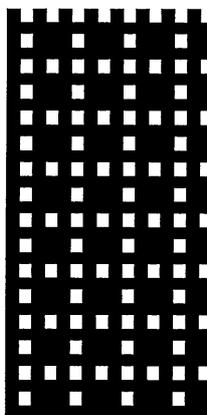
도면12



도면13



도면14



도면15



도면16



도면17

