(19)대한민국특허청(KR)
(12) 등록특허공보(B1)

(51) 。Int. Cl.[7]
H04N 1/41

(45) 공고일자 2003년12월24일
(11) 등록번호 10-0412176
(24) 등록일자 2003년12월10일

(21) 출원번호 10-2002-0004196
(22) 출원일자 2002년01월24일

(65) 공개번호 특2003-0063850
(43) 공개일자 2003년07월31일

(73) 특허권자 이호석
경기도 안양시 동안구 평촌동 933 꿈마을우성아파트 201-405

(72) 발명자 이호석
경기도 안양시 동안구 평촌동 933 꿈마을우성아파트 201-405

(74) 대리인 이종일

심사관 : 정성중

(54) 문자와 이미지가 포함된 문서의 압축, 복원 시스템 및방법

요약

본 발명은 문자와 이미지가 포함된 문서의 분할, 압축 그리고 복원 시스템 및 방법에 관한 것으로서, 더 상세하게는 문자(text)와 이미지(image)가 함께 구성되는 문서(document)의 경우에 문자와 이미지를 분할(segmentation)하여 문자는 문자대로 이미지는 이미지대로 각각 부호화하고 다시 복호화하는 문서의 분할, 압축 그리고 복원 시스템 및 방법에 관한 것이다.
본 발명에 의한 문서의 압축 시스템은, 문자와 이미지가 포함된 문서를 문자에 해당하는 포그라운드(foreground) 영상과, 배경 및 그림의 이미지에 해당하는 백그라운드(background) 영상과, 비트맵(bitmap) 영상으로 분리하여 생성하는 문서분할 모듈과; 상기 비트맵 영상을 입력으로 받아 부호화를 하여 JBG 파일을 생성하는 JBIG 부호기와; 상기 포그라운드 영상과 백그라운드 영상을 입력으로 받아 웨이블릿 리프팅 방식으로 웨이블릿 변환을 수행하여 웨이블릿 계수를 출력하는 FDWT부와; 상기 웨이블릿 계수를 입력으로 받아 부호화하여 비트스트림을 생성하는 비트평면 부호기로 구성된다.

대표도

도 1

색인어

문서분할, 포그라운드, 백그라운드, 비트맵, 웨이블릿 리프팅, FDWT/IDWT 부호기, 복호기

명세서

도면의 간단한 설명

도 1은 본 발명에 따른 문서의 압축 시스템의 블록도이다.
도 2는 본 발명에 따른 문서의 복원 시스템의 블록도이다.
도 3은 본 발명에 따른 문서의 압축시에 FDWT를 수행하는 TSFB의 구조이다.

도 4는 본 발명에 따른 TSFB의 분석과정을 나타내는 도면이다.
도 5는 본 발명에 따른 변환에 대한 1차원의 가역적인 리프팅 변환과정을 나타내는 도면이다.
도 6은 본 발명에 따른 1차원의 각 픽셀에 대해서 리프팅을 순차수만큼 수행하는 의사코드부이다.
도 7은 본 발명에 따른 색차신호 다운샘플링 과정을 나타내는 도면이다.
도 8은 본 발명에 따른 웨이블릿 계수의 양자화 과정을 나타내는 도면이다.
도 9a와 도 9b는 본 발명에 따른 웨이블릿 계수값의 비트평면과 서브블록에 대한 처리과정을 나타내는 도면이다.
도 10은 본 발명에 따른 TSFB의 합성과정을 나타내는 도면이다.
도 11은 본 발명에 따른 영상 복원과정을 나타내는 도면이다.
도 12a에서 도 12c는 원래의 실험영상이다.
도 13a 및 도 13b는 본 발명에 따른 백그라운드 영상과 포그라운드 영상의 압축률이다.
도 14a 및 도 14b는 본 발명에 따른 백그라운드 영상과 포그라운드 영상의 PSNR 수치이다.
도 15a 내지 도 15c는 원래의 실험영상을 압축하고 복원한 후의 복원영상이다.
<도면의 주요부분에 대한 부호의 설명>
100 : 문서분할 모듈 110 : JBIG 부호기
120,132 : FDWT부 122,134 : 비트평면 부호기
124,135 : 반복길이 부호기 126,136 : 산술 부호기
210 : JBIG 복호기 220,230 : 산술 복호기
222,232 : 반복길이 복호기 224,234 : 비트평면 복호기
226,236 : IDWT부 240 : 문서 재구성부

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 문자와 이미지가 포함된 문서의 분할, 압축 그리고 복원 시스템 및 방법에 관한 것으로서, 더 상세하게는 문자(text)와 이미지(image)가 함께 구성되는 문서(document)의 경우에 문자와 이미지를 분할(segmentation)하여 문자는 문자대로 이미지는 이미지대로 각각 부호화하고 다시 복호화하는 문서의 분할, 압축 그리고 복원 시스템 및 방법에 관한 것이다.
현재 가장 많이 사용하고 있는 정지 영상을 위한 압축 방법은 JPEG(Joint Photographic Experts Group) 부호화이다.
JPEG와 같은 손실 압축 방법은 압축률을 높임에 따라 원 영상에 대한 손실 또한 높아진다.
일반적으로 영상을 JPEG와 같은 손실 압축법으로 압축했을 경우, 복원한 영상은 사람의 눈에 보기에 무리가 없다.
그러나 신문, 잡지 등과 같은 높은 해상도의 문서를 스캔한 영상, 즉 문자가 포함되어 있고, 문자에 대한 중요도가 비교적 높은 영상을 JPEG과 같은 손실 압축법으로 압축했을 경우 복원한 영상에서 문자를 알아보기는 힘들다.
문자는 일반적으로 이미지에 비해 영상에서 차지하는 면적이 극히 적고 미세하기 때문에 영상의 적은 손실도 문자에 대해서는 큰 영향을 끼치기 때문이다.
즉, 종래에는 문자와 이미지를 함께 압축하고 복원하였으며, 이미지를 압축하고 복원하는 방법과 같은 방법으로 문자를 압축하고 복원하는 경우에 문자가 많이 뭉개지고 희미해져서 눈으로 읽기가 어렵게 되는 경우가 많이 있었다.
또한 종래에는 웨이블릿 리프팅 방식의 정수 변환(integer transform)을 수행하는 데 있어서, 1) 문자만을 포함한 문자 영상, 2) 간단한 흑백 영상, 3) 복잡한 흑백 영상, 4) 간단한 컬러 영상, 5) 자연 영상, 총 5가지 경우만을 고려하여 처리하였기 때문에 영상의 종류에 따라서 적합하게 정수 변환을 선택할 수 없어 손실이 많이 발생하게 되었다.

발명이 이루고자 하는 기술적 과제

본 발명은 상술한 문제점을 해결하기 위하여 안출된 것으로서, 문서를 압축하기 전에 문서 분할 부분에서 문자와 이미지를 분할하여 문자는 문자대로 이미지는 이미지대로 각각 부호화하고 다시 복호화함으로써 문자와 이미지가 포함된 문서를 동시에 압축하고 복원할 수 있고 입력되는 영상에 적합하게 정수변환을 선택할 수 있어 손실을 감소시킬 수 있으며, 따라서 문서의 높은 압축 효과와 높은 품질을 유지할 수 있는 문자와 이미지가 포함된 문서의 압축, 복원 시스템 및 방법을 제공하는데 그 목적이 있는 것이다.
상술한 목적을 달성하기 위하여 본 발명은, 문자와 이미지가 포함된 문서를, 문자에 해당하는 포그라운드 영상과, 배경 및 그림의 이미지에 해당하는 백그라운드 영상과, 비트맵 영상으로 분리하여 생성하는 문서분할 모듈과; 상기 비트맵 영상을 입력으로 받아 부호화를 하여 JBG 파일을 생성하는 JBIG 부호기와; 상기 포그라운드 영상과 백그라운드 영상을 입력으로 받아 색차 변환과 다운샘플링을 수행하고, 리프팅 방식으로 웨이블릿 변환을 수행하여 웨이블릿 계수를 출력하는 FDWT부와; 상기 웨이블릿 계수를 입력으로 받아 그레이코드(graycode) 변환과 양자화를 수행하고,

비트평면 방식에 의하여 부호화를 수행화하여 비트스트림을 생성하는 비트평면(Bit-plane) 부호기를 포함하는 문자와 이미지가 포함된 문서의 압축 시스템을 제공하고자 한다.
또한 본 발명은, 포그라운드 영상과 백그라운드 영상의 비트스트림을 입력으로 받아 웨이블릿 계수를 생성하고 역 그레이코드 변환을 수행하는 비트평면 복호 기와; JBG 파일의 비트스트림을 입력으로 받아 비트맵 영상을 생성하는 JBIG 복호기와; 상기 웨이블릿 계수를 입력으로 받아 역 웨이블릿 변환, 업샘플링, 색차 변환을 수행하여 문자에 해당하는 포그라운드 영상과, 이미지에 해당하는 백그라운드 영상을 생성하는 IDWT부와; 상기 비트맵 영상과, 포그라운드 영상과, 백그라운드 영상을 원래의 문서로 복원하는 문서 재구성부를 포함하는 문자와 이미지가 포함된 문서의 압축 시스템을 제공하고자 한다.
또한 본 발명은, 문자와 이미지가 포함된 문서의 전체 영상을, 비트맵 영상과, 문자의 포그라운드 영상과, 이미지의 백그라운드 영상으로 분리하는 제1과정과; 상기 비트맵 영상을 부호화하여 JBG 파일로 생성하는 제2과정과; 상기 포그라운드 영상에 대해 리프팅 방식으로 웨이블릿 변환을 수행하여 웨이블릿 계수를 출력하고, 상기 웨이블릿 계수를 부호화하여 비트스트림을 생성하는 제3과정과; 상기 백그라운드 영상에 대해 리프팅 방식으로 웨이블릿 변환을 수행하여 웨이블릿 계수를 출력하고, 상기 웨이블릿 계수를 부호화하여 비트스트림을 생성하는 제4과정을; 포함하는 문자와 이미지가 포함된 문서의 압축 방법을 제공하고자 한다.
또한 본 발명은, JBG 파일을 복호화하여 비트맵 영상을 생성하는 제1과정과; 포그라운드 영상의 비트스트림을 복호화하여 웨이블릿 계수를 생성하고, 상기 웨이블릿 계수에 대해 역 웨이블릿 변환을 수행하여 포그라운드 영상을 생성하는 제2과정과; 백그라운드 영상의 비트스트림을 복호화하여 웨이블릿 계수를 생성하고, 상기 웨이블릿 계수에 대해 역 웨이블릿 변환을 수행하여 백그라운드 영상을 생성하는 제3과정과; 상기 비트맵 영상과, 포그라운드 영상과 백그라운드 영상을 원래의 문서로 재구성하여 복원하는 제4과정을; 포함하는 문자와 이미지가 포함된 문서의 복원 방법을 제공하고자 한다.

발명의 구성 및 작용

이하 본 발명을 첨부된 도면을 참고로 하여 설명하면 다음과 같다.
도 1은 본 발명에 따른 문서의 압축 시스템의 블록도이다.
먼저 본 발명에 따른 부호 시스템의 기본적인 구성을 살펴보면, 문자와 이미지가 포함된 문서를, 문자에 해당하는 포그라운드 영상과, 배경 및 그림의 이미지에 해당하는 백그라운드 영상과, 비트맵 영상으로 분리하여 생성하는 문서분할 모듈(100)과;
상기 비트맵 영상을 입력으로 받아 부호화를 하여 JBG 파일을 생성하는 JBIG 부호기와(110);
상기 포그라운드 영상과 백그라운드 영상을 입력으로 받아 색차 변환, 다운샘플링, 리프팅 방식으로 웨이블릿 변환을 수행하여 웨이블릿 계수를 출력하는 FDWT부(120,132)와;
상기 웨이블릿 계수를 입력으로 받아 그레이코드(graycode) 변환과 양자화를 수행하고, 비트평면 방식에 의하여 부호화를 수행화하여 비트스트림을 생성하는 비트평면(Bit-plane) 부호기(122,134)로 이루어진다.
문자와 이미지가 포함된 문서의 압축에 있어서, 문서의 전체 영상을 문자에 해당하는 포그라운드(foreground) 영역과 배경 및 그림의 이미지에 해당하는 백그라운드(background) 영역으로 분리하여 이들 영역에 대하여 각각 다른 압축 방법을 사용하는 것이 필요하고, 이때 문서를 포그라운드 영역과 백그라운드 영역 및 비트맵 영역으로 분리하는 작업의 수행은 문서분할 모듈(100)에 의해서 이루어진다.
다음은 본 발명에 따른 시스템의 압축 과정이다.
(1) 입력 영상을 포그라운드 영역과 백그라운드 영역 그리고 비트맵으로 분할한다.
(2) 분할된 포그라운드 영역과 백그라운드 영역에 대하여 색차 변환을 수행한다.
(3) 색차 변환된 Y,U,V 신호에서 색차 값에 해당되는 U,V 값을 다운샘플링한다.
이때 휘도값 Y와 색차값 U,V의 다운샘플링 비율은 4:1:1 이다.
(4) 각 Y,U,V 콤포넌트에 대하여 웨이블릿 변환을 수행한다.
(5) 생성된 웨이블릿 계수에 대하여 그레이코드 변환을 수행한다.
(6) 그레이 코드로 변환된 웨이블릿 계수에 대하여, 각 서브밴드별로 양자화를 수행한다.
이때 LL, LH, HL, HH 서브밴드에 대하여 각기 다른 양자화를 적용한다.
(7) 다음에 양자화된 웨이블릿 계수에 대하여 비트평면을 구성하고, 각 비트평면 내에 서브블록을 구성한다.
(8) 비트평면 내의 각 서브블록에 대하여 서브블록 부호화를 수행한다.
(9) 생성된 비트스트림에 대하여 반복길이 부호화와 적응 산술 부호화를 수행한다.
문자와 이미지가 포함된 문서의 전체적인 영상은 .bmp 파일, .jpg 파일, .tiff 파일로 문서분할 모듈(100)에 입력된다.
도 12a, 도 12b, 도 12c는 원래 문서의 영상이다.
상기 문서분할 모듈(100)은 K-평균(K-means) 알고리즘을 기반으로 컬러 클러스터링을 한다.
상기 K-평균 알고리즘을 이용한 컬러 클러스터링은 RGB 좌표상에 어떠한 영상의 컬러 분포를 놓고 보았을 때 영상의 모든 픽셀(Pixel)값과 각 클러스터에 속해 있는 픽셀들의 평균값과의 거리를 비교하여 클러스터링하는 알고리즘으로서, 처음에 나눌 클러스터의 개수만큼 각 클러스터의 평균값을 초기화한다.
본 발명에서는 전체 영상을 포그라운드 컬러와 백그라운드 컬러로 클러스터링하며, K-평균 알고리즘을 이용한 컬러 클러스터링은 다음과 같은 순서로 작동한다.
1. 첫째 포그라운드 컬러는 검은색으로 백그라운드 컬러는 흰색으로 초기화한다.
2. 영상의 모든 픽셀값과 포그라운드 컬러와 백그라운드 컬러와의 거리를 각각 비교하여 더 가까운 쪽에 포함시킨다.

3. 포그라운드에 포함되는 픽셀과 백그라운드에 포함되는 픽셀에 대한 각각의 평균을 구하여 포그라운드 컬러와 백그라운드 컬러로 갱신하여 준다.
4. 각각의 컬러가 모두 모아질 때까지 2와 3을 반복한다.
그런데 일반적인 문서의 이미지는 두 가지 컬러로 제한되어 있는 경우가 드 물고, 문서의 디자인과 명암의 상태는 영상 전체에 걸쳐서 백그라운드 컬러와 포그라운드 컬러의 변화를 유발시키므로 컬러 변화 문제에 대응할 필요가 생긴다.
이 알고리즘이 상기 K-평균 컬러 클러스터링 알고리즘을 확장한 블록 바이컬러 클러스터링(block bicolor clustering) 알고리즘이다.
상기 블록 바이컬러 클러스터링 알고리즘은 전체 영상을 여러 개의 정사각형 블록으로 나누어 K-평균 컬러 클러스터링 알고리즘을 적용시킨 것으로서 영상 전체에 걸친 컬러 변화 문제에 대응할 수 있다.
정사각형 블록의 크기는 한 문장 안에서 문자의 색깔 변화를 감지할 수 있을 만큼 충분히 작아야 한다.
이 경우에 작은 블록의 블록 전체가 포그라운드 영역 혹은 백그라운드 영역에 위치하는 문제가 발생할 수 있고, 이때에는 가장 밝은 픽셀의 컬러값을 백그라운드 컬러로 설정한다.
이 방법은 문자 혹은 이미지중 어느 한쪽이 포함되어 있지 않은 그리고 매우 연속적인 픽셀을 포함한 블록에서, 연속적인 픽셀값의 특성상 포그라운드로 할 수 없는 픽셀을 포그라운드로 선택하여 블록의 픽셀을 포그라운드와 백그라운드로 구분해야만 하며, 이에 대응하기 위하여 블록 바이컬러 클러스터링 알고리즘을 확장한 알고리즘이 멀티스케일 바이컬러 클러스터링(multiscale bicolor clustering) 알고리즘이다.
상기 멀티스케일 바이컬러 클러스터링 알고리즘은 하나의 블록 크기 대신 블록 크기가 작아지는 연속적인 그리드(grid)를 고려한 것으로써 다음과 같이 수행된 다.
1. 전체 영상을 일정한 블록 크기의 그리드로 나누어 블록 바이컬러 클러스터링 알고리즘을 적용한다.
2. 전체 영상을 이전 그리드의 블록 크기보다 작은 블록 크기의 그리드로 나눈다.
3. 각 블록들의 포그라운드 컬러와 백그라운드 컬러를 현재 블록들이 속해 있는 이전 그리드 블록의 포그라운드 컬러와 백그라운드 컬러로 초기화 한다.
4. 각 블록들에 대하여 블록 안의 모든 픽셀값과 포그라운드 컬러와 백그라운드 컬러와의 거리를 각각 비교하여 더 가까운 쪽에 포함시킨다.
5. 포그라운드에 포함되는 픽셀들과 백그라운드에 포함되는 픽셀들 각각에 이전 그리드 블록의 포그라운드 컬러와 백그라운드 컬러값을 더하여 그 평균값을 포그라운드 컬러와 백그라운드 컬러로 갱신한다.
6. 각각의 컬러가 모두 모아질 때까지 4와 5를 반복하고, 일정 블록의 크기가 될 때까지 2부터 반복한다.
상기 멀티스케일 바이컬러 클러스터링 알고리즘은 블록을 초기화할 때, 이전 블록의 포그라운드 컬러와 백그라운드 컬러를 사용한다.
이러한 컬러 초기화 과정은 이전 블록에서 선택된 포그라운드 컬러와 백그라운드 컬러에 가까운 컬러를 선택하도록 하는 경향을 만들어 내며, 상기 블록 바이컬러 클러스터링 알고리즘은 작은 블록 크기 문제를 해결하여 준다.
상기 멀티스케일 바이컬러 클러스터링 알고리즘은 처음 그리드 블록의 포그 라운드 컬러를 검은색으로, 백그라운드 컬러를 흰색으로 초기화하여 클러스터링함으로써, 결과적으로는 검은색에 가까운 컬러값을 가진 픽셀은 포그라운드 컬러로, 흰색에 가까운 컬러값을 가진 픽셀은 백그라운드 컬러로 추출된다.
예를 들어 검은색 바탕에 흰색 문자가 있는 경우는 반전해야 할 필요성이 있다.
이 경우에는 포그라운드/백그라운드 반전(foreground/background inverting) 알고리즘을 통하여 반전을 수행한다.
상기 포그라운드/백그라운드 반전 알고리즘은 추출된 포그라운드 영역에서 하나의 포그라운드 또는 백그라운드 픽셀들의 모임인 객체를 추출하여 하나의 객체가 전체 영상에서 차지하는 면적과, 각 객체들 간의 위치관계를 이용하여 백그라운드로 반전할 것인지를 결정하고 반전을 수행한다.
상기한 K-평균 컬러 클러스터링 알고리즘을 기반으로 멀티스케일 바이컬러 클러스터링 알고리즘과, 포그라운드/백그라운드 반전 알고리즘으로 구성된 문서분할 모듈(100)에 의해 .bmp, .jpg, .tiff 파일 영상을 입력으로 받아 전체 영상을 분할하여 문자에 해당하는 포그라운드 영상과, 이미지에 해당하는 백그라운드 영상을 생성하고 또한 비트맵 영상을 생성한다.
일반적으로 많이 사용되는 컬러 모델에는 RGB, CMY, YUV, YIQ 컬러 모델들이 있다.
예를 들어 하드웨어나 범용적인 컬러 비디오 카메라에서는 RGB 모델이 채택되고, 컬러 프린터에서는 CMY(Cyan,Magenta,Blue) 모델이 채택되며, 컬러 프린터에 서는 YIQ 모델을 사용한다.
YUV 컬러 모델과 같은 휘도신호와 색차신호는 RGB 컬러모델에 비해서 압축을 용이하게 할 수 있다.
인간의 시각은 색차신호보다 휘도신호에 더 민감하기 때문에 색차신호값과 휘도신호값을 같은 비중으로 압축을 수행할 필요는 없다.
상기한 이유로 본 발명에서는 백그라운드 영상과 포그라운드 영상에 대해서 각각 RGB 컬러모델을 YUV 컬러모델로 변환하는 색차변환을 수행한다.
아래의 수학식 1은 RGB 신호를 YUV로 변환하는 식이다.

수학식 1

$$Y=0.30R+0.59G+0.11B$$
$$U=(B-Y)\times 0.493$$
$$V=(R-Y)\times 0.877$$

수학식 1에 의해 RGB 컬러를 YUV 컬러로 변환한 후, Y,U,V 컴포넌트에 대하여 Y:U:V는 4:1:1로 다운샘플링(down sampling)을 수행한다.
여기서 다운샘플링은 영상 샘플의 갯수를 줄인다는 의미로, 샘플의 갯수를 감소시켜 전체 영상의 크기를 줄임으로써 영상을 부호화하는 데 소요되는 시간을 단축하기 위해서이다.
이와 같은 방법으로 상기 문서분할 모듈(100)을 거쳐서 생성된 포그라운드 영상 및 백그라운드 영상은 색차 변환, 다운샘플링, FDWT부(120,132)를 거쳐서 웨이블릿 변환을 수행한다.
상기 FDWT부((Forward Discrete Wavelet Transform),120,132)에서는 리프팅 방식(Lifting scheme)의 웨이블릿 변환을 수행한다.
웨이블릿 변환을 구현하는 데 사용되는 리프팅 방식(lifting scheme)은 최근에 제안된 방식이다.
웨이블릿 리프팅 방식은 순방향 웨이블릿 변환(forward wavelet transform)으로부터 간단하게 역방향(inverse) 웨이블릿 변환을 얻을 수 있는 가역적(reversible)인 방법이고, 웨이블릿 계수가 정수가 될 수 있는 정수 변환이며, 그리고 웨이블릿 변환을 수행하는 과정에서 별도의 기억장소를 사용하지 않는다는 장점이 있다.
본 발명에서는 다음 12가지 웨이블릿 변환의 수식을 제공한다.
여기서 x[n]은 입력신호, s[n]은 로우패스 서브밴드(lowpass subband) 신호, d[n]은 하이패스 서브밴드(highpass subband) 신호를 나타나고, $s_0[n]$은 x[2n], $d_0[n]$은 x[2n+1]을 나타낸다.

수학식 2
(5/3) 변환

순방향 변환

$$d[n]=d_0[n]-\left\lfloor \frac{1}{2}(s_0[n+1]+s_0[n]) \right\rfloor$$

$$s[n]=s_0[n]+\left\lfloor \frac{1}{4}(d[n]+d[n-1])+\frac{1}{2} \right\rfloor$$

역방향 변환

$$s[n]=s_0[n]-\left\lfloor \frac{1}{4}(d[n]+d[n-1])+\frac{1}{2} \right\rfloor$$

$$d[n]=d_0[n]+\left\lfloor \frac{1}{2}(s_0[n+1]+s_0[n]) \right\rfloor$$

수학식 3
(2/6) 변환

순방향 변환

$$d_1[n]=d_0[n]-s_0[n]$$

$$s[n]=s_0[n]+\left\lfloor \frac{1}{2}d_1[n] \right\rfloor$$

$$d[n]=d_1[n]+\left\lfloor \frac{1}{4}(-s[n+1]+s[n-1])+\frac{1}{2} \right\rfloor$$

역방향 변환

$$d[n]=d_1[n]-\left\lfloor \frac{1}{4}(-s[n+1]+s[n-1])+\frac{1}{2} \right\rfloor$$

$$s[n]=s_0[n]-\left\lfloor \frac{1}{2}d_1[n] \right\rfloor$$

$$d_1[n]=d_0[n]+s_0[n]$$

수학식 4
(SP+B) 변환

순방향 변환

$$d_1[n]=d_0[n]-s_0[n]$$

$$s[n]=s_0[n]+\lfloor \frac{1}{2} d_1[n] \rfloor$$

$$d[n]=d_1[n]+\lfloor \frac{1}{8}(-3s[n+1]+s[n]+2s[n-1]+2d_1[n+1])+\frac{1}{2} \rfloor$$

역방향 변환

$$d[n]=d_1[n]-\lfloor \frac{1}{8}(-3s[n+1]+s[n]+2s[n-1]+2d_1[n+1])+\frac{1}{2} \rfloor$$

$$s[n]=s_0[n]-\lfloor \frac{1}{2} d_1[n] \rfloor$$

$$d_1[n]=d_0[n]+s_0[n]$$

수학식 5

(9/7-M) 변환

순방향 변환

$$d[n]=d_0[n]+\lfloor \frac{1}{16}((s_0[n+2]+s_0[n-1])-9(s_0[n+1]+s_0[n]))+\frac{1}{2} \rfloor$$

$$s[n]=s_0[n]+\lfloor \frac{1}{4}(d[n]+d[n-1])+\frac{1}{2} \rfloor$$

역방향 변환

$$s[n]=s_0[n]-\lfloor \frac{1}{4}(d[n]+d[n-1])+\frac{1}{2} \rfloor$$

$$d[n]=d_0[n]-\lfloor \frac{1}{16}((s_0[n+2]+s_0[n-1])-9(s_0[n+1]+s_0[n]))+\frac{1}{2} \rfloor$$

수학식 6

(2/10) 변환

순방향 변환

$$d_1[n]=d_0[n]-s_0[n]$$

$$s[n]=s_0[n]+\lfloor \frac{1}{2} d_1[n] \rfloor$$

$$d[n]=d_1[n]+\lfloor \frac{1}{64}(22(s[n-1]-s[n+1])+3(s[n+2]-s[n-2]))+\frac{1}{2} \rfloor$$

역방향 변환

$$d[n]=d_1[n]-\lfloor \frac{1}{64}(22(s[n-1]-s[n+1])+3(s[n+2]-s[n-2]))+\frac{1}{2} \rfloor$$

$$s[n]=s_0[n]-\lfloor \frac{1}{2} d_1[n] \rfloor$$

$$d_1[n]=d_0[n]+s_0[n]$$

수학식 7

(5/11-C) 변환

순방향 변환

$$d_1[n]=d_0[n]-\lfloor \frac{1}{2}(s_0[n+1]+s_0[n]) \rfloor$$

$$s[n]=s_0[n]+\lfloor \frac{1}{4}(d_1[n]+d_1[n-1])+\frac{1}{2} \rfloor$$

$$d[n]=d_1[n]+\lfloor \frac{1}{16}(s_1[n+2]-s_1[n+1]-s_1[n]+s_1[n-1])+\frac{1}{2} \rfloor$$

역방향 변환

$$d[n]=d_1[n]-\lfloor\frac{1}{16}(s_1[n+2]-s_1[n+1]-s_1[n]+s_1[n-1])+\frac{1}{2}\rfloor$$

$$s[n]=s_0[n]-\lfloor\frac{1}{4}(d_1[n]+d_1[n-1])+\frac{1}{2}\rfloor$$

$$d_1[n]=d_0[n]+\lfloor\frac{1}{2}(s_0[n+1]+s_0[n])\rfloor$$

수학식 8

(5/11-A) 변환

순방향 변환

$$d_1[n]=d_0[n]-\lfloor\frac{1}{2}(s_0[n+1]+s_0[n])\rfloor$$

$$s[n]=s_0[n]+\lfloor\frac{1}{4}(d_1[n]+d_1[n-1])+\frac{1}{2}\rfloor$$

$$d[n]=d_1[n]+\lfloor\frac{1}{32}(s_1[n+2]-s_1[n+1]-s_1[n]+s_1[n-1])+\frac{1}{2}\rfloor$$

역방향 변환

$$d[n]=d_1[n]-\lfloor\frac{1}{32}(s_1[n+2]-s_1[n+1]-s_1[n]+s_1[n-1])+\frac{1}{2}\rfloor$$

$$s[n]=s_0[n]-\lfloor\frac{1}{4}(d_1[n]+d_1[n-1])+\frac{1}{2}\rfloor$$

$$d_1[n]=d_0[n]+\lfloor\frac{1}{2}(s_0[n+1]+s_0[n])\rfloor$$

수학식 9

(6/14) 변환

순방향 변환

$$d_1[n]=d_0[n]-s_0[n]$$

$$s[n]=s_0[n]+\lfloor\frac{1}{16}(-d_1[n+1]+d_1[n-1]+8d_1[n])+\frac{1}{2}\rfloor$$

$$d[n]=d_1[n]+\lfloor\frac{1}{16}(s_1[n+2]-s_1[n-2]+6(-s_1[n+1]+s_1[n-1]))+\frac{1}{2}\rfloor$$

역방향 변환

$$d[n]=d_1[n]-\lfloor\frac{1}{16}(s_1[n+2]-s_1[n-2]+6(-s_1[n+1]+s_1[n-1]))+\frac{1}{2}\rfloor$$

$$s[n]=s_0[n]-\lfloor\frac{1}{16}(-d_1[n+1]+d_1[n-1]+8d_1[n])+\frac{1}{2}\rfloor$$

$$d_1[n]=d_0[n]+s_0[n]$$

수학식 10

SPC+C 변환

순방향 변환

$$d_1[n]=d_0[n]-s_0[n]$$

$$s[n]=s_0[n]+\lfloor\frac{1}{2}d_1[n]\rfloor$$

$$d[n]=d_1[n]+\lfloor\frac{1}{16}(-8s[n+1]+4s[n]+5s[n-1]-s[n-2]+6d_1[n+1])\frac{1}{2}\rfloor$$

역방향 변환

$$d[n]=d_1[n]-\lfloor\frac{1}{16}(-8s[n+1]+4s[n]+5s[n-1]-s[n-2]+6d_1[n+1])+\frac{1}{2}\rfloor$$

$$s[n]=s_0[n]-\lfloor\frac{1}{2}d_1[n]\rfloor$$

$$d_1[n]=d_0[n]+s_0[n]$$

수학식 11

(13/7-T) 변환

순방향 변환

$$d[n]=d_0[n]+\lfloor\frac{1}{16}((s_0[n+2]+s_0[n-1])-9(s_0[n+1]+s_0[n]))+\frac{1}{2}\rfloor$$

$$s[n]=s_0[n]+\lfloor\frac{1}{32}((-d[n+1]-d[n-2])+9(d[n]+d[n-1]))+\frac{1}{2}\rfloor$$

역방향 변환

$$s[n]=s_0[n]-\lfloor\frac{1}{32}((-d[n+1]-d[n-2])+9(d[n]+d[n-1]))+\frac{1}{2}\rfloor$$

$$d[n]=d_0[n]-\lfloor\frac{1}{16}((s_0[n+2]+s_0[n-1])-9(s_0[n+1]+s_0[n]))+\frac{1}{2}\rfloor$$

수학식 12

(13/7-C) 변환

순방향 변환

$$d[n]=d_0[n]+\lfloor\frac{1}{16}(s_0[n+2]+s_0[n-1]-9(s_0[n+1]+s_0[n]))+\frac{1}{2}\rfloor$$

$$s[n]=s_0[n]+\lfloor\frac{1}{16}(5(d[n]+d[n-1])-(d[n+1]+d[n-2]))+\frac{1}{2}\rfloor$$

역방향 변환

$$s[n]=s_0[n]-\lfloor\frac{1}{16}(5(d[n]+d[n-1])-(d[n+1]+d[n-2]))+\frac{1}{2}\rfloor$$

$$d[n]=d_0[n]-\lfloor\frac{1}{16}(s_0[n+2]+s_0[n-1]-9(s_0[n+1]+s_0[n]))+\frac{1}{2}\rfloor$$

수학식 13

(9/7-F) 변환

순방향 변환

$$d_1[n]=d_0[n]+\lfloor\frac{1}{128}(203(-s_0[n+1]-s_0[n]))+\frac{1}{2}\rfloor$$

$$s_1[n]=s_0[n]+\lfloor\frac{1}{4096}(217(-d_1[n]-d_1[n-1]))+\frac{1}{2}\rfloor$$

$$d[n]=d_1[n]+\lfloor\frac{1}{128}(113(s_1[n+1]+s_1[n]))+\frac{1}{2}\rfloor$$

$$s[n]=s_1[n]+\lfloor\frac{1}{4096}(1817(d_1[n]+d_1[n-1]))+\frac{1}{2}\rfloor$$

역방향 변환

$$s[n]=s_1[n]-\lfloor\frac{1}{4096}(1817(d_1[n]+d_1[n-1]))+\frac{1}{2}\rfloor$$

$$d[n]=d_1[n]-\lfloor\frac{1}{128}(113(s_1[n+1]+s_1[n]))+\frac{1}{2}\rfloor$$

$$s_1[n]=s_0[n]-\lfloor\frac{1}{4096}(217(-d_1[n]-d_1[n-1]))+\frac{1}{2}\rfloor$$

$$d_1[n]=d_0[n]-\lfloor\frac{1}{128}(203(-s_0[n+1]-s_0[n]))+\frac{1}{2}\rfloor$$

이러한 웨이블릿 변환을 수행하기 위한 핵심 자료구조로는 Enc_t로서, 이 자료구조는 웨이블릿 변환을 위한 트리 구조(Tree Structure)와 이미지의 각 컴포넌트에 대한 실제 데이터 및 변환을 위한 모든 정보를 포함하고 있는 구조체이다.
상기 트리 구조를 보이고 있는 TSFB(Tree-Structured Filter Bank)는 입력 영상을 분해(decomposition) 레벨에 따라서 트리 구조를 생성한다.
예를 들어 분해 레벨이 3인 경우 도 3의 트리 구조가 생성된다.
트리 구조에서 각 노드들은 해당 레벨의 밴드(band)에 대한 정보를 포함하며 영상에 대한 버퍼 포인트와 크기 등에 관한 모든 정보를 포함하고 있다.
Enc_t 구조체가 설정된 후에, 이 구조체의 데이터를 바탕으로 웨이블릿 리프팅 알고리즘을 수행하는 TSFB 분석(analyze) 과정을 거치게 된다.
도 4에서 실제 영상 데이터에 대해서 스플릿(Split)을 수행한다.
짝수(even)에 해당하는 부분은 상위부분으로, 홀수(odd)부분은 하위부분으로 분할된 후, 상위 부분은 LIFT1을 통하여 로우패스(lowpass) 필터를 거치고, 하위 부분은 LIFT0을 통하여 하이패스(highpass) 필터를 거친다.
이와 같은 과정을 순차수(number_of_sequence)만큼 반복하여 수행함으로써 1차원(수직)에 대해서 웨이블릿 리프팅이 수행된다.
수직축에 대한 1차원 리프팅 수행 후에 수평축에 대하여 리프팅을 수행하면, 전체 이미지 즉 2차원 영상에 대하여 리프팅이 수행된 결과를 얻어낼 수 있다.
도 5는 상기 수학식 2의 (5/3) 변환에 대한 1차원의 가역적인(reversible) 리프팅 변환 과정을 나타내고 그 변환 수식은 수학식 14와 같다.

수학식 14

$$LIFT\ 0 : Y_h(n)=X_h(n)-\left\lfloor \frac{X_L(n)+X_L(n+1)}{2} \right\rfloor$$

$$LIFT\ 1 : Y_L(n)=X_L(n)+\left\lfloor \frac{Y_h(n)+Y_h(n+1)}{4}+\frac{1}{2} \right\rfloor$$

도 6은 1차원의 각 픽셀에 대해서 웨이블릿 리프팅을 순차수만큼 수행하는 의사(擬似)코드(pseudo-code)부이다.
이를 수직, 수평에 대하여 각각 수행함으로써 4개의 분할된 밴드 즉, LL, HL, LH, HH를 얻을 수 있다.
이상과 같이 포그라운드 영상과 백그라운드 영상은 각각 색차 변환, 다운샘플링, FDWT부(120,132)를 거쳐 웨이블릿 계수로 변환된다.
출력된 웨이블릿 계수는 다시 그레이코드로 변환되고 양자화 과정을 거친다.
양자화 과정을 거친 후에 포그라운드 영상과 백그라운드 영상의 웨이블릿 계수들은 비트평면 부호기(134,122)를 거쳐 부호화되어 비트스트림으로 생성된다.
상기 비트평면 부호기(122,134)는 영상을 비트들로 구성된 여러개의 비트평 면(Bit-plane)으로 나누는 데서 시작한다.
가장 중요한 비트(most significant bit)를 중심으로 비트평면 0을 생성하고, 가장 덜 중요한 비트(least significant bit)를 마지막 비트평면으로 생성한다.
하나의 비트평면은 다시 적절한 크기(2x2 64x64)의 서브블록으로 나누어진다.
상기 비트평면 부호기(122,134)는 서브블록들로 나누어진 비트평면에 대하여 각 서브블록마다 반복길이 부호화와 산술 부호화 과정을 거쳐서 비트스트림을 생성한다.
이를 자세히 설명하면, 일반적으로 비트평면 부호화는 컬러 영상으로부터 비트평면을 구성하여 이진 영상(binary image)을 만들고 각 이진 영상에 대해서 압축을 수행하는 것이다.
본 발명에서는 웨이블릿 변환을 수행한 웨이블릿 계수값에 대해서 비트평면을 구성하여 이진 데이터를 생성한 다음에 서브블록 부호화를 수행한다.
그러나, 이러한 비트평면 부호화의 단점은 계수값의 조그마한 변화가 전체 비트평면을 복잡하게 만들 수 있다는 것으다.
예를 들어, 한 계수값이 127(01111111)이고 인접한 계수값이 128(100000000)이라면 계수값은 단지 1이지만 전체 비트평면에서 인접한 모든 비트값이 다르게 나타난다.
이러한 경우는 비트의 중복성을 통한 압축에 효과적이지 못하므로 서브블록 부호화를 수행하기 전에 각 필셀의 값을 그레이코드(graycode)로 변환해야 한다.
아래의 수학식 15는 m-비트의 비트평면에 대해서 m개의 그레이코드 $g_{m-1}\cdots g_2 g_1 g_0$ 으로 변환하는 경우에 변환식이다.

수학식 15

$$g_i=a_i \oplus a_{i+1} \qquad 0\le i\le m-2$$
$$g_{m-1}=a_{m-1}$$

여기서, $\oplus$ 는 XOR 연산을 나타내고, a_i 와 a_{i+1} 는 웨이블릿 계수에 대한 비트값들이다.
다시 설명하면 웨이블릿 계수 비트값들을 그레이코드로 변환된 후, 최상위 비트는 그대로 그레이코드가 되고($g_{m-1}=a_{m-1}$), 그 다름 그레이코드 값은 다음 하위 두개의 비트값들을 XOR한 것으로 대체되는 것이다($g_i=a_i \oplus a_{i+1}$).
상기 수학식 15에 의해 127(01111111)은 그레이코드로 변환후 01000000값으로 바뀌고, 이웃하는 값이 128(10000000)일 경우 그레이코드로 변환후 11000000값으로 변환된다.
즉, 비트평면에서 이웃하는 비트 사이의 값의 변화는 현저히 줄어든다.
이는 비트간의 중복성을 증가시켜 압축을 더욱 효과적으로 할 수 있게 한다.
그레이코드로 변환된 웨이블릿 계수는 최소 2바이트의 저장공간을 요구하며, 계수값을 줄여서 비트평면 부호화의 효율을 높이고 압축율을 최대화하기 위하여 양자화(quantization)를 수행한다.
본 발명에서는 웨이블릿 계수값을 쉬프트(shift)하는 방식으로 양자화을 구현한다.
웨이블릿 변환을 통한 서브밴드들중에서 이미지의 평균값을 담고 있는 LL밴드의 쉬프트는 LH, HL, HH밴드보다 상대적으로 적게 하여 복원된 영상의 왜곡현상을 최소화하고, 또한 상대적으로 LH, HL, HH밴드의 쉬프트를 크게 함으로서 압축률을 높일 수 있다.
또한 색차변환을 통한 Y, U, V신호값에서 색차값을 담고 있는 U, V신호의 쉬프트를 Y신호에 비해 상대적으로 크게 하여 압축률을 높일 수 있다.
도 8은 웨이블릿 계수의 양자화 과정을 나타낸다.
이미지의 평균값을 나타내는 LL밴드의 계수값을 n비트만큼 쉬프트하고, LH, HL밴드에 추가적으로 1비트(n+1)를, HH 밴드에 대해서는 추가적으로 2비트(n+2)를 더 쉬프트한다.
쉬프트 연산을 수행한 후 다시 역방향으로 쉬프트 연산을 수행하면, 결과적으로 LL밴드는 n비트만큼의 0값이 설정된다.
상술한 양자화를 거친 후에 웨이블릿 계수를 비트평면 기반하에서 서브블록으로 분할하고, 각 서브블륵에 대해서 비트평면 부호화를 수행한다.
이러한 쉬프트 양자화 과정을 거치면 계수값의 비트에서 0값이 증가하게 되고, 이러한 0값의 증가는 서브블록 부호화에서 "중요하지 않은(insignificant)" 서브블록의 증가를 가져오게 된다.
도 9a는 웨이블릿 계수값의 비트평면을 나타내고 도 9b는 각 비트평면에 대 해 서브블록의 처리과정을 나타낸다.
비트평면 부호기(122,134)는 비트평면이 포함하고 있는 각 서브블록에 대해서 "중요성(significance)" 여부를 조사한다.
중요성 여부는 서브블록에서 1의 존재 여부에 따라 결정된다.
1값이 단 한개라도 존재하면 그 서브블록은 "중요한(significant)" 서브블록으로 간주된다.
상기 서브블록 내의 값이 모두 0이라면 그 서브블록은 "중요하지 않은(insignificant)" 서브블록으로 간주되어 부호화에서 제외되고, 단지 전체 비트평면에서 그 서브블록 위치에 대한 정보만이 서브블록 맵에 기록된다.
중요한 서브블록은 서브블록 전체가 부호화되며 서브블록 맵에는 중요한 서브블록을 나타내는 값으로 1값을 저장한다.
비트평면에 기반한 서브블록 부호화는 전체 비트평면에 대해서 각각 수행되고, 중요 서브블록에 대해서는 비트스트림을 출력한다.
이때 서브블록 내의 비트는 좌측상단에서 우측하단으로 스캔하면서 부호화를 한다.
중요하지 않은 서브블록은 어떠한 비트스트림도 출력하지 않으며, 단지 서브블록 맵에 그 서브블록의 위치정보를 기록한다.
한편, 비트맵 영상을 부호화하여 .jbg 파일을 생성하는 JBIG 부호기(110)는 이진 영상을 고능률로 부호화할 수 있다.
여기서 JBIG 부호기(110)는 JBIG-1 또는 JBIG-2를 의미한다.
JBIG(Joint Bi-level Image Experts Group)는 이진 영상(Bi-level Image) 그리고 그레이스케일(grayscale) 영상을 부호화한다.
JBIG는 이진 영상의 손실/무손실 압출을 위해 ISO/IEC 11544(ITU-T T.82) 권고안으로 규정되어 있다.
JBIG는 주위의 화소로부터 부호화 화소를 예측하면서 부호화를 수행하기 때문에 이진 영상을 고능률로 부호화할 수 있다.
상기 비트평면 부호화기(122,134)를 거쳐서 부호화된 포그라운드 영상과 백그라운드 영상은 반복길이 부호기(124,135)와 산술부호기(126,136)를 거친다.
반복길이 부호화 방법은 픽셀 블록을 하나의 픽셀값과 그 반복횟수로 나타내는 방법으로 동일한 값을 갖는 픽셀들을 하나의 코드로 사용하여 나타냄으로서 영상 내에 존재하는 값의 중복성을 감소시키는 방법이다.
예를 들어, 첫번째 픽셀값이 234이고 동일한 값을 갖는 픽셀이 연속적으로 이웃하여 14픽셀이 나타난다면 234@14로 부호화를 한다.
원래의 영상은 15픽셀의 값을 표현하기 위하여 15바이트가 필요한 것에 반하여 위와 같이 반복길이 부호화를 수행한 다음에는 2바이트로 표현이 가능하다.
산술부호기(126,136)의 적응 산술(adaptive arithmetic) 부호화 알고리즘을 사용한다.
상술한 바와 같은 부호화 방법은, 문자와 이미지가 포함된 .bmp, .jpg, .tiff 파일의 문서는 문서분할 모듈(100)에 의해서 비트맵 영상과 문자의 포그라운드 영상과 이미지의 백그라운드 영상으로 분할된다(제1과정).
상기 비트맵 영상은 JBIG 부호기(110)를 거쳐 부호화되어 .jbg 파일로 생성된다(제2과정).

상기 포그라운드 영상은 리프팅 방식을 사용하여 웨이블릿 변환을 수행하는 FDWT부(120)를 거쳐 웨이블릿 계수로 출력되고, 상기 웨이블릿 계수를 입력으로 받는 비트평면 부호기(122)를 거쳐 부호화되어 비트스트림을 생성한다(제3과정).
상기 백그라운드 영상은 리프팅 방식을 사용하여 웨이블릿 변환을 수행하는 FDWT부(122)를 거쳐 웨이블릿 계수로 출력되고, 상기 웨이블릿 계수를 입력으로 받는 비트평면 부호기(134)를 거쳐 부호화되어 비트스트림을 생성한다(제4과정).
제3과정과 제4과정을 거친 포그라운드 영상과 백그라운드 영상은 계속하여 반복길이 부호기(124,135)에 의해 영상 내에 존재하는 값의 중복성이 감소되고, 적응 산술 부호기(126,136)에 의해 구간의 크기가 줄어들어 이 구간 내에 존재하는 값만 부호화되어 압축이 완료된다.
복호화 과정은 부호화의 반대 과정을 거치며, 이하는 복호화 과정을 나타낸다.
(1) 포그라운드와 백그라운드에 대한 비트스트림에 대하여 적응 산술 복호화와 반복길이 복호화를 수행한다.
(2) 비트맵에 대한 비트스트림은 JBIG 복호기(210)에서 복호화되어 비트맵을 생성한다.
(3) 포그라운드와 백그라운드 영역에 대한 비트스트림은 비트평면 복호화를 수행하여 비트평면을 구성한다.
(4) 생성된 웨이블릿 계수에 대하여 역 그레이코드 변환을 수행한다.
(5) 생성된 Y,U,V 각 컴포넌트에 대하여 역 웨이블릿 변환을 수행한다.
(6) 생성된 Y,U,V 컴포넌트에서 U,V 컴포넌트 값을 업샘플링한다.
이때 휘도값 Y와 색차값 U,V의 업샘플링 비율은 1:4:4 이다.
(7) YUV 컬러에서 RGB 컬러로 색차 변환을 수행하고, 색차 변환은 포그라운드 영상과 백그라운드 영상에 대하여 각각 수행한다.
(8) 색차 변환된 포그라운드 영상, 백그라운드 영상, 그리고 비트맵으로 원 영상을 복원한다.
이러한 복원 시스템의 구조는 도 2에서 처럼, 포그라운드 영상과 백그라운드 영상의 비트스트림을 입력으로 받아 웨이블릿 계수를 생성하고 역 그레이코드 변환을 수행하는 비트평면 복호기(220,230)와;
JBG 파일의 비트스트림을 입력으로 받아 비트맵 영상을 생성하는 JBIG 복호기(210)와;
상기 웨이블릿 계수를 입력으로 받아 역 웨이블릿 변환, 업샘플링, YUV에서 RGB로의 색차 변환을 수행하여 문자에 해당하는 포그라운드 영상과, 이미지에 해당하는 백그라운드 영상을 생성하는 IDWT부(226,236)와;
상기 비트맵 영상과, 포그라운드 영상과, 백그라운드 영상을 원래의 문서로 재구성하는 재구성부(240)로 구성된다.
상기 JBIG 복호기(210)는 비트스트림을 입력으로 받아 복호화를 수행하여 비트맵 영상을 생성한다.
도 2에서 비트스트림을 입력으로 하여 적응 산술 복호기(220,230)에 의해 상술한 복호화 과정을 거친다.
반복길이 복호기(222,232)는 동일한 값을 갖는 픽셀들을 하나의 코드를 사용하여 나타내는 부호화의 역과정으로 복호화를 수행한다.
예를 들어 234@12로 부호화한 값을 첫번째 픽셀값이 234이고 동일한 값을 픽셀이 연속적으로 이웃하여 14의 픽셀이 나타난 것으로 표시하여 복호화한다.
상기 비트평면 복호기(224,234)는 비트스트림을 입력으로 받아 웨이블릿 계수를 생성하고 역 그레이코드 변환을 수행하여 웨이블릿 계수를 생성한다.
IDWT부(226,236)는 웨이블릿 계수를 입력으로 받아 문자에 해당하는 포그라운드 영상과 이미지에 해당되는 백그라운드 영상을 생성한다.
상기 IDWT(Inverse Discrete Wavelet Transform)의 자료구조는 도 3의 FDWT 자료구조와 같다.
도 10은 TSFB의 합성(synthesize)과정을 나타낸다.
상기 합성 과정은 도 4의 분석과정의 역과정으로서, LIFT1, LIFT0, 합성을 순차수만큼 1차원(수직)에 대해서 수행하고, 또 1차원(수평)에 대해 반복 수행함으로써 원래의 영상을 얻을 수 있다.
이를 수직, 수평에 대하여 각각 수행함으로서 도 11에서와 같이 포그라운드 영상과 백그라운드 영상에 대한 원래의 영상을 얻는다.
이와 같은 방법으로, 비트맵 영상과, 포그라운드 영상과 백그라운드 영상이 생성되면, 이는 문서 재구성부((document reconstruction),240)에 의해 원래(original) 문서의 영상으로 재구성되어 복원이 완료된다.
상술한 바와 같은 압축된 문서의 복원은, JBG 파일을 복호화하여 비트맵 영상을 생성한다(제1과정).
포그라운드 영상의 부호화된 비트스트림을 복호화하여 웨이블릿 계수를 생성하고, 역 그레이코드 변환을 거친 후, 역 웨이블릿 변환을 수행하고, YUV 컴포넌트에 대하여 1:4:4로 업샘플링을 수행하고, YUV 컬러를 RGB 컬러로 색차 변환을 수행하여 포그라운드 영상을 생성한다(제2과정).
백그라운드 영상의 부호화된 비트스트림을 복호화하여 웨이블릿 계수를 생성하고, 역 그레이코드 변환을 거친 후, 역 웨이블릿 변환을 수행하고, YUV 컴포넌트에 대하여 1:4:4로 업샘플링을 수행하고, YUV 컬러를 RGB 컬러로 색차 변환을 수행하여 백그라운드 영상을 생성한다(제3과정).
상기 비트맵 영상과, 포그라운드 영상과 백그라운드 영상을 원래의 문서로 재구성하여 복원한다(제4과정).
이와 같은 본 발명에 따른 웨이블릿 변환과 비트평면 부호화를 이용한 문서 압축 Codec의 성능을 실험하기 위해서 사용된 원래의 실험영상은 임의의 크기를 가진 영상으로 한다.
도 12a(Lena 영상(400x255))는 이미지만 담고 있으며, 어떠한 문자 정보도 없는 영상이다.
도 12b(Hobby 영상(791x565))는 이미지와 문자가 분리된 형태의 영상이다.
도 12c(ATT 영상(512x512))는 이미지와 문자가 혼용된 형태의 영상이다.
상기한 원래의 실험영상들은 RGB형태로 저장된 컬러 영상들이다.

상기 각 영상에 대해서 웨이블릿 변환을 수행하고, 웨이블릿 계수값에 대해서 비트평면 부호화를 수행한 다음 압축된 영상에 대해서 복원을 수행하고 이에 따른 PSNR(Peak Signal-to-Noise Ratio)을 구하기 위한 식은 수학식 16과 같다.

수학식 16

$$PSNR=20\log_{10}(\frac{255}{MSE})$$

여기에서, 255는 픽셀에서의 최대값이고, MSE(Mean Squared Error)는 모든 픽셀에 대한 원래의 실험영상과 복원영상과의 오차의 합 제곱으로 수학식 17에 의해 구해진다.

수학식 17

$$MSE=\frac{\sum[f(i,j)-F(i,j)]^2}{W\times H}$$

여기서 W는 이미지 폭이고 H는 이미지의 높이이다.
도 13a와 도 13b는 백그라운드와 포그라운드 영상에 대한 압축비율을 나타낸고 도 14a와 14b는 복원된 영상에 대한 PSNR 수치를 나타낸다.
여기서 백그라운드 영상에 대해서는 평균적으로 100:9 정도의 압축률을 보이고 있고, 포그라운드 영상에 대해서는 평균적으로 100:7정도의 압축률을 보이고 있음을 알 수 있다.
백그라운드 영상에 대해서는 (5/3)과 (5/11-C) 웨이블릿 리프팅이 가장 높은 압축률을 보인다.
(9/7-F)는 압축률이 비교적 좋지 못하지만, PSNR은 가장 높은 값을 나타내고 있다.
(5/3)과 (5/11-C) 웨이블릿 리프팅 역시 높은 압축률을 나타내면서도 비교적 높은 PSNR을 보이고 있다.
포그라운드 영상에서는 (5/3)과 (2/6) 웨이블릿 리프팅이 가장 높은 압축률을 나타낸다.
(5/3) 웨이블릿 리프팅은 압축률과 동시에 높은 PSNR을 나타내고 있다.
반면에 (9/7-F)는 압축율과 PSNR 모두에서 가장 비효율적임을 볼 수 있다.
도 15a, 도 15b, 도 15c는 실험영상 도 12a, 도 12b, 12c에 대한 압축된 파일을 복원한 복원영상을 나타낸다.
상기 도 12와 도 15를 비교해 보면, 실험영상과 복원영상이 시각적으로 거의 차이가 없음을 알 수 있다.

발명의 효과

이상에서 살펴본 바와 같이 본 발명에 의하면, 문자와 이미지를 분리하여 문자는 문자대로 그리고 영상은 영상대로 압축하고 복원할 수 있다.
또한, 본 발명에 의하면, 12가지의 정수 변환(integer transform)을 수행할 수 있어 입력되는 영상에 적합하게 정수 변환을 더욱 세밀하게 선택할 수 있으므로 영상을 압축하고 복원하였을 경우에 발생하는 손실을 상당히 많이 감소시킬 수 있다.
또한 웨이블릿 계수에 대한 그레이코드로의 변환을 통해 비트평면에서 중복 성을 높임으로써 압축효과를 높이고, 각 서브밴드에 대해 다른 비율로 양자화를 적용함으로서 이미지의 시각적 질(quality)을 낮추지 않으면서 압축률을 높일 수 있다.

(57) 청구의 범위

청구항 1.

문자와 이미지가 포함된 문서를, 문자에 해당하는 포그라운드 영상과, 배경 및 그림의 이미지에 해당하는 백그라운드 영상과, 비트맵 영상으로 분리하여 생성하는 문서분할 모듈과;
상기 비트맵 영상을 입력으로 받아 부호화를 하여 JBG 파일을 생성하는 JBIG 부호기와;
상기 포그라운드 영상과 백그라운드 영상을 입력으로 받아 색차 변환과 다운샘플링을 수행하고, 리프팅 방식으로 웨이블릿 변환을 수행하여 웨이블릿 계수를 출력하는 FDWT부와;
상기 웨이블릿 계수를 입력으로 받아 그레이코드 변환과 양자화를 수행하고, 비트평면 방식에 의하여 부호화를 수행화하여 비트스트림을 생성하는 비트평면 부호기를;
포함하는 문자와 이미지가 포함된 문서의 압축 시스템.

청구항 2.

포그라운드 영상과 백그라운드 영상의 비트스트림을 입력으로 받아 웨이블릿 계수를 생성하고 역 그레이코드 변환을 수행하는 비트평면 복호기와;
JBG 파일의 비트스트림을 입력으로 받아 비트맵 영상을 생성하는 JBIG 복호기와;
상기 웨이블릿 계수를 입력으로 받아 역 웨이블릿 변환, 업샘플링, 색차 변 환을 수행하여 문자에 해당하는 포그라운드 영상과, 이미지에 해당하는 백그라운드 영상을 생성하는 IDWT부와;

상기 비트맵 영상과, 포그라운드 영상과, 백그라운드 영상을 원래의 문서로 복원하는 문서 재구성부를;
포함하는 문자와 이미지가 포함된 문서의 복원 시스템.

청구항 3.

문자와 이미지가 포함된 문서의 전체 영상을, 비트맵 영상과, 문자의 포그라운드 영상과, 이미지의 백그라운드 영상으로 분할하는 제1과정과;
상기 비트맵 영상을 부호화하여 JBG 파일로 생성하는 제2과정과;
상기 포그라운드 영상에 대해 리프팅 방식으로 웨이블릿 변환을 수행하여 웨이블릿 계수를 출력하고, 상기 웨이블릿 계수를 부호화하여 비트스트림을 생성하는 제3과정과;
상기 백그라운드 영상에 대해 리프팅 방식으로 웨이블릿 변환을 수행하여 웨이블릿 계수를 출력하고, 상기 웨이블릿 계수를 부호화하여 비트스트림을 생성하는 제4과정을;
포함하는 문자와 이미지가 포함된 문서의 압축 방법.

청구항 4.

청구항 3에 있어서, 상기 제1과정에서 문서의 분할은,
전체의 영상을 포그라운드 컬러와, 백그라운드 컬러로 클러스터링하며,
1. 포그라운드 컬러는 검은색으로 백그라운드 컬러는 흰색으로 초기화하는 과정,
2. 영상의 모든 픽셀값과 포그라운드 컬러와 백그라운드 컬러와의 거리를 각각 비교하여 더 가까운 쪽에 포함시키는 과정,
3. 포그라운드에 포함되는 픽셀과 백그라운드에 포함되는 픽셀에 대한 각각의 평균을 구하여 포그라운드 컬러와 백그라운드 컬러로 갱신하여 주는 과정,
4. 각각의 컬러가 모두 모아질 때까지 2과정과 3과정을 반복하는 과정을,
거쳐 이루어짐을 특징으로 하는 문자와 이미지가 포함된 문서의 압축 방법.

청구항 5.

청구항 3 또는 청구항 4에 있어서, 상기 제1과정에서 분할된 백그라운드 영상과 포그라운드 영상에 대해서 RGB 컬러모델을 YUV 컬러모델로 변환하는 색차변환을 수행하고, Y와 U와 V 컴포넌트에 대해서 4:1:1로 다운샘플링을 수행함을 특징으로 하는 문자와 이미지가 포함된 문서의 압축방법.

청구항 6.

청구항 3에 있어서, 상기 웨이블릿 계수값을 그레이코드로 변환하고 상기 그레이코드로 변환된 계수를 각 밴드별(LL, LH,HL,HH)로 다른 양자화를 수행한 후, 부호화를 행함을 특징으로 하는 문자와 이미지가 포함된 문서의 압축방법.

청구항 7.

청구항 6에 있어서, 상기 양자화는 계수값을 쉬프트(shift)하는 방식으로 구현됨을 특징으로 하는 문자와 이미지가 포함된 문서의 압축방법.

청구항 8.

청구항 7에 있어서, 상기 밴드중에서 이미지의 평균값을 담고 있는 LL밴드의 쉬프트는 LH, HL, HH밴드보다 상대적으로 적게함을 특징으로 하는 문자와 이미지가 포함된 문서의 압축방법.

청구항 9.

청구항 3 또는 청구항 6에 있어서, 상기 부호화는 웨이블릿 계수를 비트평면 기반하에서 서브블록으로 분할하고 각 서브블록에 대해서 비트평면 부호화를 수행하는 것임을 특징으로 하는 문자와 이미지가 포함된 문서의 압축 방법.

청구항 10.

청구항 9에 있어서, 상기 서브블록에 대해서 서브블록에 존재하는 1의 유무에 따라서 고려되는 중요성(significance) 여부를 조사하여, 서브블록이 1의 값을 포함하고 있는 중요한(signficant) 서브블록인 경우는 부호화하여 비트스트림을 생성하고 서브블록이 0으로만 구성된 중요하지 않은(insignificant) 서브블록인 경우에는 부호화에서 제외하고 전체 비트평면에서 그 서브블록 위치에 대한 정보만 서브블록 맵에 저장함을 특징으로 하는 문자와 이미지가 포함된 문서의 압축 방법.

청구항 11.

청구항 10에 있어서, 상기 서브블록이 0으로만 구성된 중요하지 않은 서브블록인 경우에 어떠한 비트스트림도 출력하지 않음을 특징으로 하는 문자와 이미지가 포함된 문서의 압축 방법.

청구항 12.

청구항 3에 있어서, 상기 제3,4과정의 웨이블릿 리프팅 수행은,
실제 영상 데이터에 대해 스플릿트를 수행하여 짝수부분은 상위부분으로 홀수부분은 하위부분으로 분할한 후,
상위부분은 로우패스 필터(lowpass filter)를 거치고, 하위부분은 하이패스 필터(highpass filter)를 거치는 과정을,
순차수만큼 반복하여 수행함을 특징으로 하는 문자와 이미지가 포함된 문서의 압축 방법.

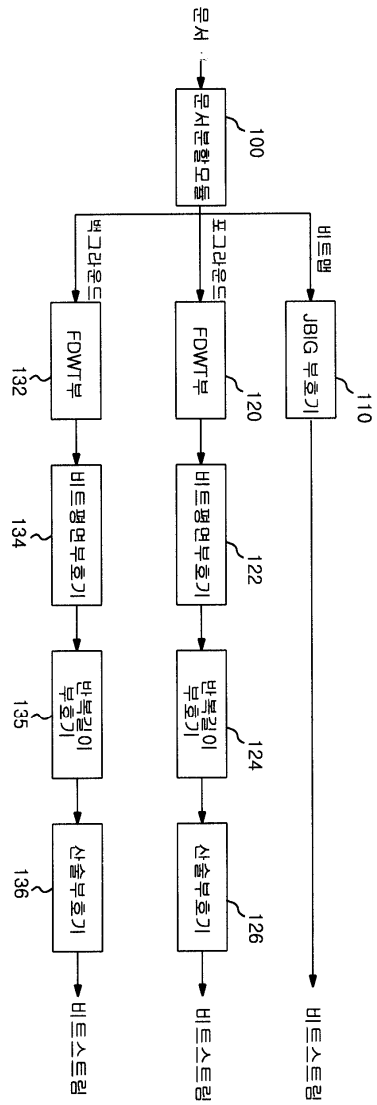
청구항 13.

부호화된 JBG 파일을 복호화하여 비트맵 영상을 생성하는 제1과정과;
포그라운드 영상의 부호화된 비트스트림을 복호화하여 웨이블릿 계수를 생성하고, 역 그레이코드 변환을 거친 후, 역 웨이블릿 변환을 수행하고, YUV 컴포넌트에 대하여 1:4:4로 업샘플링을 수행하고, YUV 컬러를 RGB 컬러로 색차 변환을 수행하여 포그라운드 영상을 생성하는 제2과정과;
백그라운드 영상의 부호화된 비트스트림을 복호화하여 웨이블릿 계수를 생성하고, 역 그레이코드 변환을 거친 후, 역 웨이블릿 변환을 수행하고, YUV 컴포넌트에 대하여 1:4:4로 업샘플링을 수행하고, YUV 컬러를 RGB 컬러로 색차

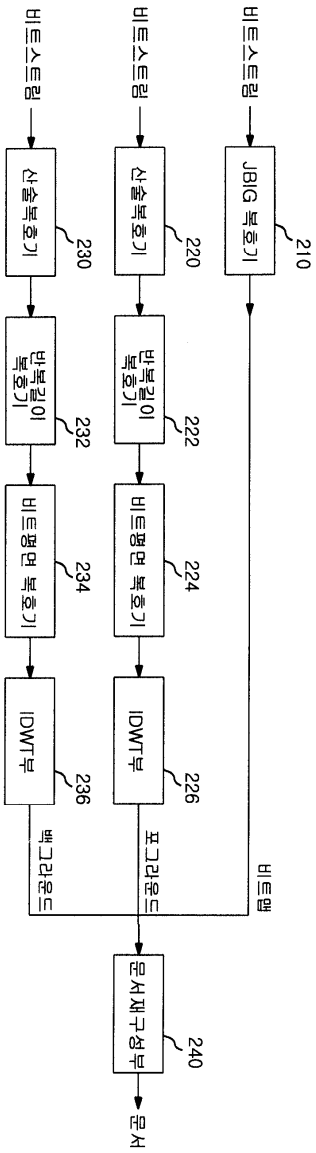
변환을 수행하여 백그라운드 영상을 생성하는 제3과정과;
상기 비트맵 영상과 포그라운드 영상과 백그라운드 영상을 원래의 문서로 재구성하여 복원하는 제4과정을;
포함하는 문자와 이미지가 포함된 문서의 복원 방법.

도면

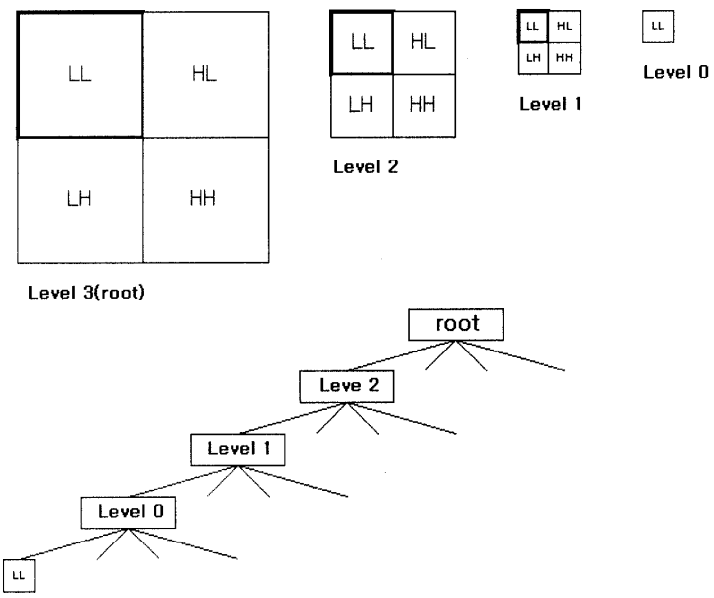
도면1



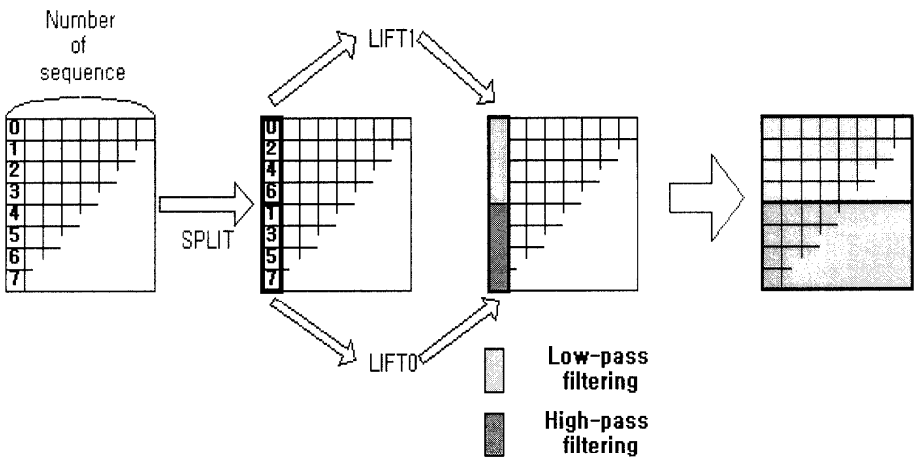
도면2


비트스트림
비트스트림
비트스트림
산술복호기
230
산술복호기
220
JBIG 복호기
210
반복길이 복호기
232
반복길이 복호기
222
비트평면 복호기
234
비트평면 복호기
224
IDWT부
236
IDWT부
226
백그라운드
포그라운드
비트맵
문서재구성부
240
문서

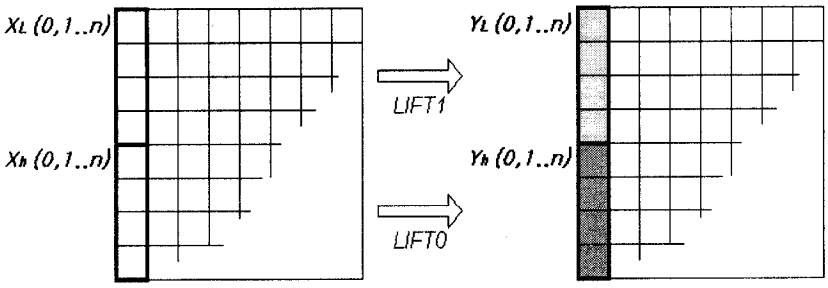
도면3


Level 3(root)
Level 2
LL
HL
LH
HH
Level 1
LL
Level 0
root
Leve 2
Level 1
Level 0

도면4



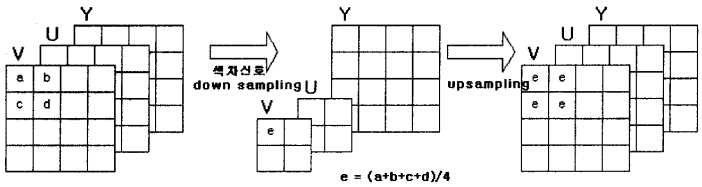
도면5



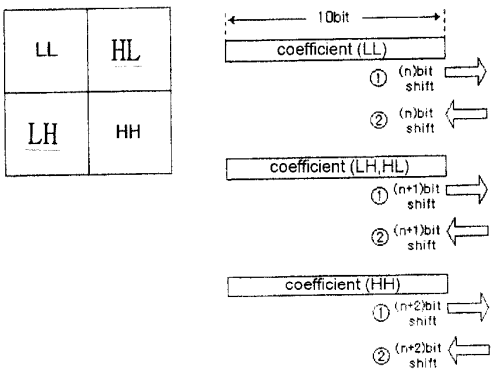
도면6

```
while(numseq)
{
        setup : // 1차원(수직)평면에 대한 이미지픽셀 포인터 설정
        split  : // 짝수번째 픽셀은 상위로, 홀수번째 픽셀은 하위로 분할
        LIFT0 : // 홀번번째 픽셀에 대한 High-pass filtering
        LIFT1 : // 짝수번째 픽셀에 대한 Low_pass filtering
}
```

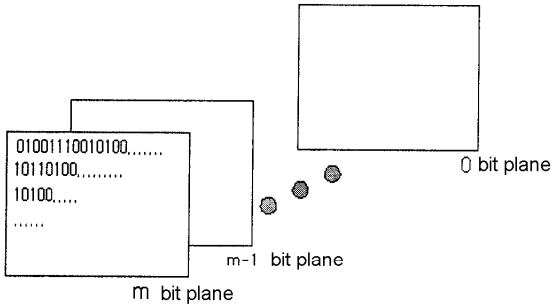
도면7



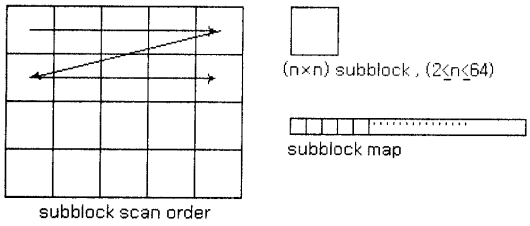
도면8



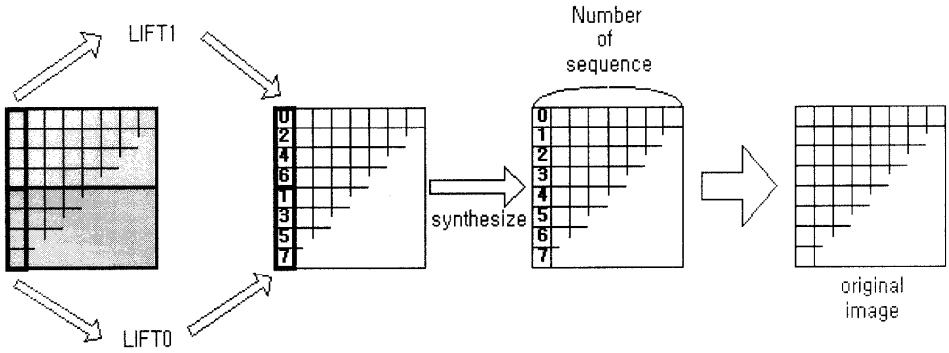
도면9a



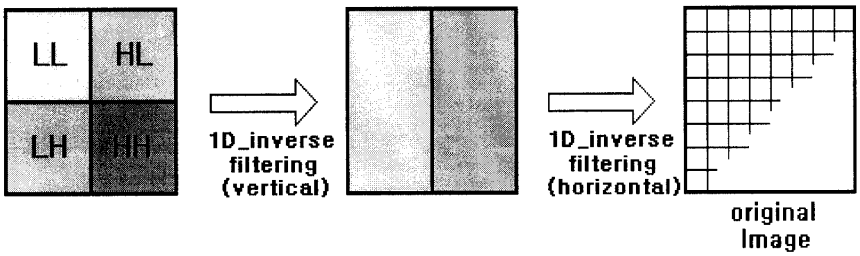
도면9b



도면10



도면11

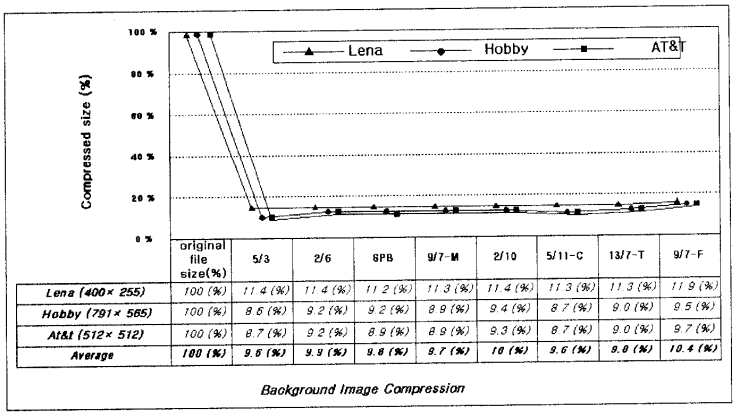


도면12a

도면12b

도면12c

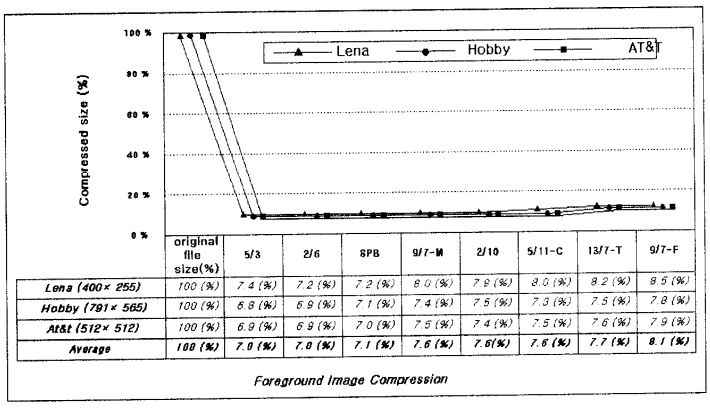
도면13a



	original file size(%)	5/3	2/6	8PB	9/7-M	2/10	5/11-C	13/7-T	9/7-F
Lena (400× 255)	100 (%)	11.4 (%)	11.4 (%)	11.2 (%)	11.3 (%)	11.4 (%)	11.3 (%)	11.3 (%)	11.9 (%)
Hobby (791× 565)	100 (%)	8.6 (%)	9.2 (%)	9.2 (%)	8.9 (%)	9.4 (%)	8.7 (%)	9.0 (%)	9.5 (%)
At&t (512× 512)	100 (%)	8.7 (%)	9.2 (%)	8.9 (%)	8.9 (%)	9.3 (%)	8.7 (%)	9.0 (%)	9.7 (%)
Average	100 (%)	9.6 (%)	9.9 (%)	9.8 (%)	9.7 (%)	10 (%)	9.6 (%)	9.8 (%)	10.4 (%)

Background Image Compression

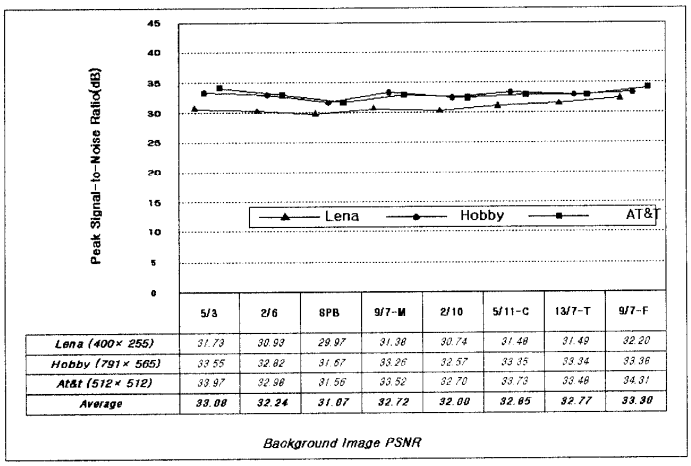
도면13b



	original file size(%)	5/3	2/6	8PB	9/7-M	2/10	5/11-C	13/7-T	9/7-F
Lena (400× 255)	100 (%)	7.4 (%)	7.2 (%)	7.2 (%)	8.0 (%)	7.8 (%)	8.0 (%)	8.2 (%)	8.5 (%)
Hobby (791× 565)	100 (%)	6.8 (%)	6.9 (%)	7.1 (%)	7.4 (%)	7.5 (%)	7.3 (%)	7.5 (%)	7.8 (%)
At&t (512× 512)	100 (%)	6.9 (%)	6.9 (%)	7.0 (%)	7.5 (%)	7.4 (%)	7.5 (%)	7.6 (%)	7.9 (%)
Average	100 (%)	7.0 (%)	7.0 (%)	7.1 (%)	7.6 (%)	7.6(%)	7.6 (%)	7.7 (%)	8.1 (%)

Foreground Image Compression

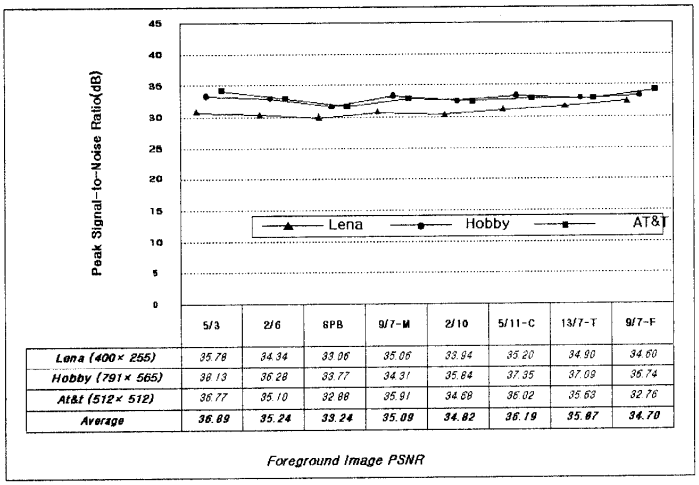
도면14a



	5/3	2/6	SPB	9/7-M	2/10	5/11-C	13/7-T	9/7-F
Lena (400× 255)	31.73	30.93	29.97	31.38	30.74	31.48	31.49	32.20
Hobby (791× 565)	33.55	32.82	31.57	33.26	32.57	33.35	33.34	33.36
At&t (512× 512)	33.97	32.98	31.56	33.52	32.70	33.73	33.48	34.31
Average	33.08	32.24	31.07	32.72	32.00	32.85	32.77	33.30

Background Image PSNR

도면14b



	5/3	2/6	SPB	9/7-M	2/10	5/11-C	13/7-T	9/7-F
Lena (400× 255)	35.78	34.34	33.06	35.05	33.94	35.20	34.90	34.60
Hobby (791× 565)	38.13	36.28	33.77	34.31	35.84	37.35	37.09	36.74
At&t (512× 512)	36.77	35.10	32.88	35.91	34.08	36.02	35.63	32.76
Average	36.89	35.24	33.24	35.09	34.82	36.19	35.87	34.70

Foreground Image PSNR

도면15a

도면15b


NEWS ITEM! Standard ECO establishes world record for endurance with 63 minute flight!
On August 17, 1996 a standard ECO helicopter was flown for 63.29 minutes in one flight, by Norbert Gruntjens. This flight was witnessed by 150 people including 30 active RC helicopter pilots at Schramberg, Germany.
ECO: For expert or beginner -- your best choice in electric helicopters!
NEW! Buy ECO Electric Helicopter for only $199.00
IK001 Ikarus ECO Electric Helicopter kit, regular price $299.00
SPECIAL PRICE $199.00 (in effect until withdrawn by Hobby Lobby)
ECO cut-performs most glow powered helicopters! And, it's electric so you can fly anywhere, anytime. No noise, no pollution. Fly indoors when it's cold and rainy outside. ECO is a contest performance helicopter that is
•ECO is a strong, fully aerobatic helicopter with collective pitch. It can do all helicopter aerobatics -- aggressively! You can choose Duration Motor that will give ECO more flying time than any other electric helicopter. Sport Motor is the best choice for good overall performance for the beginner. ECO needs only a 8 cell nicad pack! ECO was designed and is manufactured in Germany by two time German RC helicopter champion Norbert Gruntjens.
•ECO is the best choice for a helicopter beginner:
It flies better than other electric's, has longer flight times on a battery charge, is extremely durable, and it includes a mechanical mixing system that lets you use inexpensive RC airplane radios.
•ECO has the clearest instructions we've seen. You will be able to assemble it perfectly and install your radio equipment in a few evenings.
•ECO includes two mixing systems: With the mechanical mixing system ECO can use any standard 4 channel RC airplane radio with mini servos. Or you can use an expert helicopter radio that has CCPM mixing with the 90° CCPM collective/cyclic pitch control system that's also included.
Rotor diameter: 41.75", Length overall: 35.85", Weight without nicad pack:

도면15c


documents on the Web with high resolution ... another useful invention from AT&T Labs.
http://www.djvu.att.com
AT&T
pression techniques only begin to solve the problem. As a result, Web site content developers have been unable to leverage existing printed materials.
Until now.
Introducing DjVu℠
DjVu℠ (pronounced "déjà vu") is a highly sophisticated, ground-breaking technology developed by AT&T Labs. DjVu℠ is the Next Generation