



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2018년03월07일

(11) 등록번호 10-1835342

(24) 등록일자 2018년02월28일

- (51) 국제특허분류(Int. Cl.)
HO4N 19/50 (2014.01) *HO4N 19/503* (2014.01)
HO4N 19/513 (2014.01) *HO4N 19/593* (2014.01)
- (52) CPC특허분류
HO4N 19/50 (2015.01)
HO4N 19/503 (2015.01)
- (21) 출원번호 10-2017-7005439(분할)
- (22) 출원일자(국제) 2008년10월16일
 심사청구일자 2017년02월24일
- (85) 번역문제출일자 2017년02월24일
- (65) 공개번호 10-2017-0024168
- (43) 공개일자 2017년03월06일
- (62) 원출원 특허 10-2016-7002786
 원출원일자(국제) 2008년10월16일
 심사청구일자 2016년02월01일
- (86) 국제출원번호 PCT/KR2008/006113
- (87) 국제공개번호 WO 2009/051419
 국제공개일자 2009년04월23일

(30) 우선권주장
 60/980,432 2007년10월16일 미국(US)

(뒷면에 계속)

(56) 선행기술조사문헌

Competition-Based Scheme for Motion Vector Selection and Coding(VCEG-AC06), Video Coding Experts Group (VCEG) 29thMeeting
 (2006.07.18.)*

*는 심사관에 의하여 인용된 문헌

전체 청구항 수 : 총 8 항

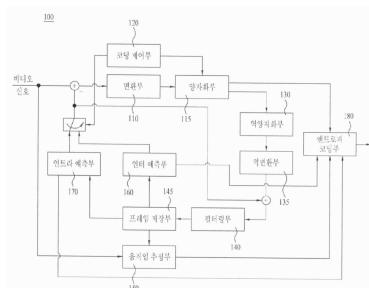
심사관 : 장석환

(54) 발명의 명칭 비디오 신호 처리 방법 및 장치

(57) 요약

본 발명의 비디오 신호의 디코딩 방법에 따르면, 예측 모드 정보, 인터폴레이팅 정보, 및 현재 블록의 레지듀얼을 수신하는 단계; 상기 인터폴레이팅 정보와 주변 블록을 이용하여 인터폴레이팅 픽셀을 복원하는 단계; 및 상기 인터폴레이팅 픽셀, 상기 예측 모드 정보, 및 상기 레지듀얼을 이용하여 현재 블록을 복원하는 단계를 포함하

(뒷면에 계속)

대 표 도

며, 상기 인터플레이팅 정보는 상기 현재 블록의 위치를 기초로 생성되는 것이 개시된다.

본 발명의 비디오 신호 처리 방법 및 장치에 따르면, 인터플레이팅 정보에 기초하여 인터플레이팅 퍽셀을 복원하여 이용함으로써, 인코딩시 화면의 경계 영역에 위치하는 현재 블록에서 이용가능한 인트라 예측 모드가 한정되었던 종래의 방법을 개선하여, 높은 복원율을 가질 수 있게 된다.

(52) CPC특허분류

H04N 19/513 (2015.01)*H04N 19/593* (2015.01)

(72) 발명자

박준영서울특별시 서초구 바우뫼로 38, LG전자 전자기술
원 (우면동)**성재원**서울특별시 서초구 바우뫼로 38, LG전자 전자기술
원 (우면동)**박현옥**대전광역시 유성구 대학로 291, KAIST 전자전기공
학부 (구성동)**이지홍**대전광역시 유성구 대학로 291, KAIST 전자전기공
학부 (구성동)**이진영**대전광역시 유성구 대학로 291, KAIST 전자전기공
학부 (구성동)**전동산**대전광역시 유성구 대학로 291, KAIST 전자전기공
학부 (구성동)**피아오 인지**대전광역시 유성구 대학로 291, KAIST 전자전기공
학부 (구성동)

(30) 우선권주장

60/980,434 2007년10월16일 미국(US)

60/980,435 2007년10월16일 미국(US)

61/016,808 2007년12월27일 미국(US)

61/029,577 2008년02월19일 미국(US)

61/031,705 2008년02월27일 미국(US)

61/075,759 2008년06월26일 미국(US)

명세서

청구범위

청구항 1

현재 블록을 위한 움직임 벡터 차이값(motion vector difference)을 획득하는 단계;

비디오 신호로부터 상기 현재 블록을 위한 움직임 벡터 예측값(motion vector predictor)을 지시하는 움직임 벡터 지시 정보를 획득하는 단계;

상기 현재 블록에 인접한 이웃 블록으로부터 획득된 공간적 움직임 벡터 후보를 포함하여 복수의 움직임 벡터 후보를 포함하는 움직임 벡터 후보 세트를 구성하는 단계;

상기 구성된 움직임 벡터 후보 세트 중에서 상기 움직임 벡터 지시 정보에 대응하는 움직임 벡터 후보를 상기 현재 블록을 위한 움직임 벡터 예측값으로 설정하는 단계; 및

상기 움직임 벡터 예측값 및 상기 움직임 벡터 차이값에 기초하여 상기 현재 블록을 디코딩하는 단계를 포함하되,

상기 공간적 움직임 벡터 후보는,

상기 현재 블록의 참조 픽처 넘버와 상기 현재 블록을 포함하는 현재 픽처 넘버 간의 차이 및 상기 이웃 블록의 참조 픽처 넘버와 상기 현재 블록을 포함하는 현재 픽처 넘버 간의 차이를 기반으로 상기 이웃 블록의 움직임 벡터를 스케일링하여 획득되는, 비디오 신호 디코딩 방법.

청구항 2

제1항에 있어서,

상기 이웃 블록은 상기 현재 블록에 인접한 좌측 이웃 블록 또는 상단 이웃 블록인, 비디오 신호 디코딩 방법.

청구항 3

제1항에 있어서,

상기 현재 블록을 포함하는 현재 픽처와 서로 다른 픽처 내에서 상기 현재 블록에 대응하는 위치의 블록으로부터 시간적 움직임 벡터 후보를 획득하는 단계를 더 포함하며,

상기 움직임 벡터 후보 세트는 상기 획득한 시간적 움직임 벡터 후보를 더 포함하는, 비디오 신호 디코딩 방법.

청구항 4

제1항에 있어서,

상기 현재 블록은 인터 예측 모드를 이용하여 코딩된 블록인, 비디오 신호 디코딩 방법.

청구항 5

현재 블록을 위한 움직임 벡터 차이값(motion vector difference)을 획득하고, 비디오 신호로부터 상기 현재 블록을 위한 움직임 벡터 예측값(motion vector predictor)을 지시하는 움직임 벡터 지시 정보를 획득하도록 구성된 엔트로피 디코딩부; 및

상기 현재 블록에 인접한 이웃 블록으로부터 획득된 공간적 움직임 벡터 후보를 포함하여 복수의 움직임 벡터 후보를 포함하는 움직임 벡터 후보 세트를 구성하고, 상기 구성된 움직임 벡터 후보 세트 중에서 상기 움직임 벡터 지시 정보에 대응하는 움직임 벡터 후보를 상기 현재 블록을 위한 움직임 벡터 예측값으로 설정하고, 상기 움직임 벡터 예측값 및 상기 움직임 벡터 차이값에 기초하여 상기 현재 블록을 디코딩하도록 구성된 인터 예측부를 포함하되,

상기 공간적 움직임 벡터 후보는,

상기 현재 블록의 참조 픽처 넘버와 상기 현재 블록을 포함하는 현재 픽처 넘버 간의 차이 및 상기 이웃 블록의 참조 픽처 넘버와 상기 현재 블록을 포함하는 현재 픽처 넘버 간의 차이를 기반으로 상기 이웃 블록의 움직임 벡터를 스케일링하여 획득되는, 비디오 신호 디코딩 장치.

청구항 6

제5항에 있어서,

상기 이웃 블록은 상기 현재 블록에 인접한 좌측 이웃 블록 또는 상단 이웃 블록인, 비디오 신호 디코딩 장치.

청구항 7

제5항에 있어서,

상기 인터 예측부는 또한 상기 현재 블록을 포함하는 현재 픽처와 서로 다른 픽처 내에서 상기 현재 블록에 대응하는 위치의 블록으로부터 시간적 움직임 벡터 후보를 획득하도록 구성되며,

상기 움직임 벡터 후보 세트는 상기 획득한 시간적 움직임 벡터 후보를 더 포함하는, 비디오 신호 디코딩 장치.

청구항 8

제5항에 있어서,

상기 현재 블록은 인터 예측 모드를 이용하여 코딩된 블록인, 비디오 신호 디코딩 장치.

발명의 설명

기술 분야

[0001] 본 발명은 비디오신호의 처리 방법 및 장치에 관한 것으로, 보다 상세하게는 비디오 신호를 인코딩하거나 디코딩하는 비디오 신호 처리 방법 및 장치에 관한 것이다.

배경 기술

[0002] 압축 부호화란 디지털화한 정보를 통신 회선을 통해 전송하거나, 저장 매체에 적합한 형태로 저장하기 위한 일련의 신호 처리 기술을 의미한다. 압축 부호화의 대상에는 음성, 영상, 문자 등의 대상이 존재하며, 특히 영상을 대상으로 압축 부호화를 수행하는 기술을 비디오 영상 압축이라고 일컫는다. 비디오 영상의 일반적인 특징은 공간적 중복성, 시간적 중복성을 지니고 있는 점에 특징이 있다.

발명의 내용

해결하려는 과제

[0003] 이와 같이 공간적 중복성 및 시간적 중복성을 충분히 제거하지 않으면, 비디오 신호를 부호화하는데 있어서 압축율이 낮아질 뿐만 아니라, 공간적 중복성 및 시간적 중복성을 과도하게 제거하는 경우, 비디오 신호를 디코딩하는 데 있어서 필요한 정보를 생성하지 못하기 때문에 복원율이 나빠지는 문제점이 있었다.

[0004] 본 발명은 상기와 같은 문제점을 해결하기 위해 창안된 것으로서, 화면의 경계 영역에 위치하는 블록들도 제한되지 아니한 9 가지 인트라 예측 모두를 모두 이용하도록 하여 압축 효율을 증가시키는 비디오 신호 처리 방법 및 장치를 제공하는 데 그 목적이 있다.

[0005] 본 발명의 또 다른 목적은, 다양한 움직임 벡터 프리딕터 후보를 이용함으로써, 움직임 벡터 예측의 정확도를 증가시키는 비디오 신호 처리 방법 및 장치를 제공하는 데 있다.

과제의 해결 수단

[0006] 상기와 같은 목적을 달성하기 위하여 본 발명에 따른 비디오 신호 처리 방법은, 예측 모드 정보, 인터플레이팅 정보, 및 현재 블록의 레지듀얼을 수신하는 단계; 상기 인터플레이팅 정보와 주변 블록을 이용하여 인터플레이팅 픽셀을 복원하는 단계; 및 상기 인터플레이팅 픽셀, 상기 예측 모드 정보, 및 상기 레지듀얼을 이용하여 현재 블록을 복원하는 단계를 포함하며, 상기 인터플레이팅 정보는 상기 현재 블록의 위치를 기초로 생성되는 것

을 포함한다.

- [0007] 본 발명에 따르면, 상기 인터플레이팅 정보는, 상기 인터플레이팅 픽셀을 생성하기 위하여 이용된 상기 주변 블록의 픽셀 정보 및 상기 인터플레이팅 픽셀을 생성하는 방법에 관한 정보를 포함할 수 있다.
- [0008] 본 발명에 따르면, 상기 인터플레이팅 정보는 상기 현재 블록이 화면의 경계 영역에 위치하는지 여부를 나타내는 플래그 정보를 더 포함할 수 있다.
- [0009] 본 발명에 따르면, 픽셀 예측 정보, 레지듀얼을 포함하는 비디오 신호를 수신하는 단계; 상기 픽셀 예측 정보 및 상기 레지듀얼을 이용하여 현재 블록의 하나의 픽셀을 복원하는 단계; 및 상기 복원된 픽셀, 상기 픽셀 예측 정보, 및 상기 레지듀얼을 이용하여 인접 픽셀을 복원하는 단계를 포함하며, 상기 픽셀 예측 정보는 인접 픽셀의 인트라 예측에 이용된 예측 픽셀, 예측 방법, 및 예측 방향을 포함하고, 상기 인접 픽셀은 현재 블록에 포함되는 픽셀로 현재 블록에 포함되는 다른 픽셀들을 이용하여 예측된 것일 수 있다.
- [0010] 본 발명에 따르면, 상기 예측 픽셀은 상기 현재 블록 또는 복원된 주변 블록에 포함되는 픽셀일 수 있다.
- [0011] 본 발명에 따르면, 상기 예측 방법은 인접 픽셀들의 평균값을 이용하는 방법 및 방향성에 따라 예측하는 방법일 수 있다.
- [0012] 본 발명에 따르면, 상기 방향성에 따라 예측하는 방법은, 상기 현재 픽셀을 중심으로 각 방향에서 가장 가까운 두 픽셀의 차이를 계산하는 단계; 및 상기 차이가 가장 작은 방향의 두 픽셀의 평균값을 이용하여 예측하는 단계를 더 포함하는 것일 수 있다.
- [0013] 본 발명에 따르면, 상기 예측 방향은 DC 모드를 제외한 인트라 예측 모드의 방향 중 하나이고, 현재 픽셀을 중심으로 각 방향에서 상기 현재 픽셀과 가장 가까운 두 픽셀의 차이가 가장 작은 값을 갖는 방향일 수 있다.
- [0014] 본 발명의 또 다른 측면에 따른 비디오 신호의 처리 방법에 따르면, 움직임 벡터 프리딕터 정보, 움직임 벡터의 차이값을 포함하는 비디오 신호를 수신하는 단계; 공간적 프리딕터 및 시간적 프리딕터를 이용하여 움직임 벡터 프리딕터 후보군 세트를 결정하는 단계; 상기 움직임 벡터 프리딕터 정보를 이용하여 움직임 벡터 프리딕터들을 결정하는 단계; 및 상기 움직임 벡터 프리딕터 및 상기 움직임 벡터의 차이값을 이용하여 현재 블록을 움직임 보상하는 단계를 포함하며, 상기 움직임 벡터 프리딕터는 상기 움직임 벡터 프리딕터 후보군 세트를 이용하여 생성되는 것일 수 있다.
- [0015] 본 발명에 따르면, 상기 움직임 벡터 프리딕터 후보군 세트를 결정하는 단계는, 현재 블록이 스kip 모드를 채택하는지 여부에 따라 결정되는 것일 수 있다.
- [0016] 본 발명에 따르면, 상기 스kip 모드는 8*8 블록, 8*16 블록, 16*8 블록, 및 16*16 블록 중 어느 하나를 단위 블록으로 포함하는 것일 수 있다.
- [0017] 본 발명에 따르면, 상기 현재 블록이 스kip 모드인 경우, 상기 움직임 벡터 프리딕터 후보군 세트를 결정하는 단계는, 상기 공간적 프리딕터의 x 성분과 상기 시간적 프리딕터의 x 성분 및 상기 공간적 프리딕터의 y 성분과 상기 공간적 프리딕터의 y 분의 일치여부에 기초하는 것일 수 있다.
- [0018] 본 발명의 비디오 신호 처리 방법에 따르면, 상기 현재 블록이 스kip 모드가 아닌 경우, 상기 움직임 벡터 프리딕터 후보군 세트를 결정하는 단계는, 상기 공간적 프리딕터의 x 성분과 상기 시간적 프리딕터의 x 성분 사이의 거리 및 상기 공간적 프리딕터의 y 성분과 상기 공간적 프리딕터의 y 성분 사이의 거리에 기초하는 것일 수 있다.
- [0019] 본 발명에 따르면, 상기 움직임 벡터 프리딕터 정보는 상기 현재 블록의 움직임 벡터 예측시 이용된 상기 움직임 벡터 프리딕터에 관한 것을 나타낼 수 있다.
- [0020] 본 발명에 따르면, 상기 움직임 벡터 프리딕터 후보군 세트를 결정하는 단계 이전에, 공간적 프리딕터 및 시간적 프리딕터를 이용하여 움직임 벡터 프리딕터 결정 방법을 선택하는 단계를 더 포함할 수 있다.
- [0021] 본 발명에 따르면, 상기 움직임 벡터 프리딕터 결정 방법을 선택하는 단계는, 상기 공간적 프리딕터 및 상기 시간적 프리딕터의 크기에 기초하여 결정되는 것일 수 있다.
- [0022] 본 발명에 따르면, 상기 움직임 벡터 프리딕터 후보군 세트를 결정하는 단계 이전에, 이전 화면의 움직임 벡터 및 현재 블록의 타입에 기초하여 움직임 벡터 프리딕터 결정 방법을 선택하는 단계를 더 포함할 수 있다.
- [0023] 본 발명에 따르면, 상기 이전 화면의 움직임 벡터 크기의 평균값이 0 내지 2 이하인 경우, 상기 공간적 프리딕

터 및 상기 시간적 프리딕터를 포함하는 상기 움직임 벡터 프리딕터 후보군 세트를 결정할 수 있다.

[0024] 본 발명에 따르면, 상기 이전 화면의 움직임 벡터 크기의 평균값이 2 초과이고, 상기 현재 블록의 타입이 스킵 모드 또는 16*16 블록인 경우, 상기 공간적 프리딕터 및 상기 시간적 프리딕터를 포함하는 상기 움직임 벡터 프리딕터 후보군 세트를 결정할 수 있다.

[0025] 본 발명에 따르면, 상기 움직임 벡터 프리딕터를 결정하는 단계는, 현재 블록과 움직임 추정을 수행하는 블록의 거리를 고려하여 스케일된 움직임 벡터 프리딕터를 생성하는 단계를 더 포함할 수 있다.

[0026] 본 발명의 또 다른 측면에 따른 비디오 신호의 처리 장치에 따르면, 예측 모드 정보, 인터폴레이팅 정보, 및 현재 블록의 레지듀얼을 포함하는 비디오 신호를 수신하는 비디오 신호 수신부; 상기 인터폴레이팅 정보 및 주변 블록을 이용하여 인터폴레이팅 픽셀을 복원하는 인터폴레이팅 픽셀 복원부; 및 상기 인터폴레이팅 픽셀, 상기 예측 모드 정보, 및 상기 레지듀얼을 이용하여 현재 블록을 복원하는 현재 블록 복원부를 포함하며, 상기 인터폴레이팅 픽셀 복원부는 상기 현재 블록의 위치를 기초로 생성된 상기 인터폴레이팅 정보를 이용하는 것일 수 있다.

발명의 효과

[0027] 본 발명은 다음과 같은 효과와 이점을 제공한다.

[0028] 첫째, 본 발명의 비디오 신호 처리 방법은, 화면의 경계 영역에 위치하는 블록들도 제한되지 아니한 9 가지 인트라 예측 모두를 모두 이용하도록 하여 압축 효율을 증가시킬 수 있다.

[0029] 둘째, 본 발명의 비디오 신호 처리 방법은, 현재 블록이 주변블록의 픽셀뿐만 아니라, 현재 블록 내의 인접 픽셀을 인트라 예측에 이용함으로써 예측의 정확도 및 코딩 효율을 높일 수 있다.

[0030] 셋째, 본 발명의 비디오 신호 처리 방법은, 인트라 스kip 모드를 이용함으로써, 비트율을 감소시킬 수 있다.

[0031] 넷째, 본 발명의 비디오 신호 처리 방법은, 분리 인트라 예측 모드를 이용함으로써, 예측 블록이 현재 블록의 상부 및 왼쪽에 위치하는 블록으로 한정되지 아니하여 현재 프레임의 정확한 예측을 가능하게 할 수 있다.

[0032] 다섯째, 본 발명의 비디오 신호 처리 방법은, 다양한 움직임 벡터 프리딕터 후보들을 이용함으로써, 현재 블록의 움직임 벡터와 유사한 프리딕터를 채택하여 정확한 움직임 예측이 가능할수 있다.

[0033] 여섯째, 본 발명의 비디오 신호 처리 방법은, 현재 블록과 이전 프레임의 유사성 및 현재 블록의 타입에 따라 움직임 벡터 프리딕터 후보를 결정함으로써, 현재 블록의 움직임 예측의 정확성을 증가시킬 수 있다.

[0034] 일곱째, 본 발명의 비디오 신호 처리 방법은, 현재 블록의 레퍼런스 인덱스가 나타내는 프레임과 주변 블록의 레퍼런스 인덱스가 나타내는 프레임 사이의 거리를 고려하여 움직임 벡터 프리딕터를 결정함으로써, 결정된 움직임 벡터 프리딕터가 현재 블록의 움직임 벡터와 더욱 유사한 값을 갖도록 하여, 예측의 정확도를 증가시킬 수 있다.

[0035] 여덟번째, 본 발명의 비디오 신호 처리 방법은, 부분 스kip 모드를 이용함으로써, 스kip 모드가 더욱 활성화되어 코딩 효율을 높일 수 있으며, 종래의 스kip 모드보다 더 많은 움직임 벡터를 이용함으로써 현재 블록의 복원율을 높일 수 있다.

[0036] 아홉번째, 본 발명의 비디오 신호 처리 방법은, 8*8 공간적 다이렉트 모드를 이용함으로써, 16*16 블록 내에서 두 개 이상의 움직임 벡터를 이용할 수 있어 현재 블록이 경계 영역에 위치하거나 작은 물체가 움직이는 경우에서 보다 높은 복원율을 가질 수 있는 효과가 있다.

[0037] 열번째, 본 발명의 비디오 신호 처리 방법은, 이산 코사인 변환 이전에 열 또는 행에서 적은 레지듀얼 값을 갖는 블록이 왼쪽 또는 위쪽에 위치하도록 레지듀얼 신호들을 재배열하여, 이산 코사인 변환을 수행하는 경우 이산 코사인 변환계수들이 좌측(직류성분 근처)로 몰려 분포하게 되므로 보다 높은 압축효과를 얻을 수 있다.

도면의 간단한 설명

[0038] 도 1 은 본 발명의 일실시예에 따른 비디오 신호의 인코딩 장치를 나타내는 개략적인 블록도이다.

도 2 는 본 발명의 일실시예에 따른 비디오 신호의 디코딩 장치를 나타내는 개략적인 블록도이다.

도 3 는 본 발명의 제 1 실시예에 따른 확장 예측 모드(Extension Prediction Mode)시 사용 되는 주변 픽셀 영

역을 나타내는 것이다.

도 4a 내지 도 4c 는 본 발명의 제 1 실시예에 따른 확장 예측 모드를 이용하기 위하여 인터플레이팅된 주변 픽셀을 나타내는 것이다.

도 5 는 본 발명의 제 1 실시예를 따른 확장 예측 모드를 채용한 비디오 신호의 인코딩/디코딩 순서를 나타내는 순서도이다.

도 6a 및 제 6b 는 본 발명의 제 1 실시예에 따른 인코딩/디코딩 장치 중 인트라 예측부를 나타내는 블록도이다.

도 7은 본 발명의 제 2 실시예에 따른 프레임-픽셀 예측 모드를 수행하는 순서를 나타내는 순서도이다.

도 8a 내지 도 8b 는 본 발명의 제 2 실시예에 따른 인코딩/디코딩 장치 중 인트라 예측부에서 현재 블록을 예측 및 복원하는 방법을 나타내는 것이다.

도 9 는 본 발명의 제 3 실시예에 따르는 비디오 영상을 나타내는 것이다.

도 10 은 본 발명의 제 3 실시예에 따른 비디오 신호의 인코딩 및 디코딩 순서를 나타내는 순서도이다.

도 11 은 본 발명의 제 4 실시예에 따른 비디오 신호의 인코딩 및 디코딩 순서를 나타내는 순서도이다.

도 12a 내지 도 12c 는 본 발명의 제 4 실시예의 분리 인트라 예측 모드(Separate Intra Prediction Mode)를 나타내는 블록의 예들이다.

도 13 은 본 발명의 제 5 실시예를 채택한 비디오 신호의 인코딩 장치를 나타내는 것이다.

도 14 는 본 발명의 제 5 실시예를 채택한 비디오 신호의 디코딩 장치를 나타내는 것이다.

도 15 는 본 발명의 제 5 실시예를 따르는 현재 블록의 움직임 벡터 예측을 위하여 이용되는 현재 프레임 및 이전 프레임의 움직임 벡터를 나타내는 것이다.

도 16a 및 도 16b 는 본 발명의 제 5 실시예의 움직임 벡터 프리딕터 결정법에서 가능한 움직임 벡터 프리딕터의 조합을 나타내는 것이다.

도 17 는 본 발명의 제 5 실시예를 따르는 움직임 벡터 프리딕터의 결정 및 움직임 보상의 순서를 나타내는 순서도이다.

도 18은 본 발명의 제 6 실시예를 따른 움직임 벡터 프리딕터 후보 결정방법을 나타내는 것이다.

도 19 는 본 발명의 제 6 실시예를 따른 경우의 BD rate 압축율을 나타내는 그래프이다.

도 20 및 도 21 은 본 발명의 제 7 실시예의 스케일된 움직임 벡터 프리딕터를 결정하는 방법을 나타내는 것이다.

도 22 는 본 발명의 제 8 실시예의 부분 스윕 모드에서 이용하는 블록 사이즈를 나타내는 것이다.

도 23a 는 종래의 공간적 다이렉트 모드를 채택한 P8*8 블록들의 움직임 벡터를 나타내는 것이다.

도 23b 는 본 발명의 제 9 실시예를 따른 8*8 블록의 움직임 벡터를 나타낸 것이다.

도 24 는 일반적인 DCT 방법을 수행하는 것을 나타내는 것이고, 도 25 는 본 발명의 제 10 실시예에 따른 이산 코사인 재배열 변환 방법을 나타내는 것이다.

발명을 실시하기 위한 구체적인 내용

[0039] 이하 첨부된 도면을 참조로 본 발명의 바람직한 실시예를 상세히 설명하기로 한다. 이에 앞서, 본 명세서 및 청구범위에 사용된 용어나 단어는 통상적이거나 사전적인 의미로 한정해서 해석되어서는 아니되며, 발명자는 그 자신의 발명을 가장 최선의 방법으로 설명하기 위해 용어의 개념을 적절하게 정의할 수 있다는 원칙에 입각하여 본 발명의 기술적 사상에 부합하는 의미와 개념으로 해석되어야만 한다. 따라서, 본 명세서에 기재된 실시예와 도면에 도시된 구성은 본 발명의 가장 바람직한 일 실시예에 불과할 뿐이고 본 발명의 기술적 사상을 모두 대변하는 것은 아니므로, 본 출원시점에 있어서 이들을 대체할 수 있는 다양한 균등물과 변형예들이 있을 수 있음을 이해하여야 한다.

- [0040] 특히, 본 발명에서 코딩이라 함은, 인코딩 및 디코딩을 모두 포함하는 개념으로 이해되어야 한다. 또한, 본 발명에서 주변 블록이라 함은, 디코딩이 이미 완료된 현재 블록의 주변에 위치하는 블록으로 이해되어야 한다.
- [0041] 도 1은 본 발명의 일 실시예에 따른 비디오 신호 인코딩 장치(100)의 개략적인 블록도이다. 도 1을 참조하면, 변환부(110), 양자화부(115), 코딩 제어부(120), 역양자화부(130), 역변환부(135), 필터링부(140), 프레임 저장부(145), 움직임 추정부(150), 인터 예측부(160), 인트라 예측부(170), 및 엔트로피 코딩부(180)을 포함한다.
- [0042] 변환부(110)는 화소값을 변환하여 변환 계수값을 획득하는데, 이 때 이산 코사인 변환(DCT: Discrete Cosine Transform) 또는 웨이블릿 변환 방식이 사용될 수 있다. 특히, 이산 코사인 변환은 입력된 영상신호를 8×8 블록으로 나누어 적은 수의 영상 신호에 신호를 집중시켜 압축 효율을 증가시키는데, 본 발명에서 제안하는 이산 코사인 변환의 실시예에 관하여는, 추후 도 23 및 도 24를 참고하여 설명하고자 한다. 양자화부(115)는 변환부(110)에서 출력된 변환 계수값을 양자화한다. 코딩 제어부(120)는 특정 블록 또는 프레임을 화면내 부호화 할 것인지, 화면간 부호화를 할 것인지를 제어한다. 역양자화부(130) 및 역변환부(135)는 변환 계수값을 역양자화하고, 역양자화된 변환 계수값을 이용하여 원래의 화소값을 복원한다.
- [0043] 필터링부(140)는 블록 왜곡 현상을 감소시키기 위하여 각각의 코딩된 매크로블록에 적용된다. 필터는 블록의 가장자리를 부드럽게 하여 디코딩된 픽처의 화질을 향상시킨다. 필터링 과정의 선택은 경계 세기(boundary strength)와 경계 주위의 이미지 샘플의 변화(gradients)에 의해 좌우된다. 필터링을 거친 픽처는 출력되거나 참조 픽처로 이용하기 위해 프레임 저장부(145)에 저장된다.
- [0044] 움직임 추정부(150)는 프레임 저장부(145)에 저장된 참조 픽처를 이용하여, 참조 픽처 중에서 현재 블록과 가장 유사한 참조 블록을 탐색한다. 움직임 추정부(150)는 탐색된 참조 블록의 위치 정보 등을 엔트로피 코딩부(180)로 전달하여 비트스트림에 포함되도록 할 수 있다. 인터 예측부(160)는 참조 픽처를 이용하여 현재 픽처의 예측을 수행하고, 화면간 부호화정보를 엔트로피 코딩부(180)에 전달한다. 여기서 화면간 부호화정보는 현재 블록과 예측된 블록(프리딕터, predictor)과의 차이인 레지듀얼을 포함한다. 인터 예측부(160)에 대한 제 4 실시예 내지 제 9 실시예는 추후 도면을 참조하여 구체적으로 설명하기로 한다.
- [0045] 또한, 인트라 예측부(170)는 현재 픽처 내의 디코딩된 픽셀로부터 화면내 예측을 수행하며, 화면내 부호화정보를 엔트로피 코딩부(180)에 전달한다. 상세하게는, 현재 픽처 내에서 주변 블록의 픽셀을 이용하여 현재 블록을 예측하고, 현재 블록 및 예측된 현재 블록과의 차이인 레지듀얼을 생성한다. 인트라 예측부(170)에 대한 제 1 실시예 내지 제 3 실시예는 도 3A 내지 도 11C를 참조하여 구체적으로 설명하기로 한다.
- [0046] 엔트로피 코딩부(180)는 양자화된 변환 계수, 화면간 부호화 정보, 화면내 부호화 정보, 및 움직임 추정부(160)로부터 입력된 참조 블록 정보 등을 엔트로피 코딩하여 비디오 신호 비트스트림을 생성한다. 여기서 엔트로피 코딩부(180)에서는 가변 길이 코딩(VLC: Variable Length Coding) 방식과 산술 코딩(arithmetic coding)이 사용될 수 있다. 가변 길이 코딩(VLC) 방식은 입력되는 심볼들을 연속적인 코드워드로 변환하는데, 코드워드의 길이는 가변적일 수 있다. 예를 들어, 자주 발생하는 심볼들을 짧은 코드워드로 자주 발생하지 않은 심볼들은 긴 코드워드로 표현하는 것이다. 가변 길이 코딩 방식으로서 컨텍스트 기반 적응형 가변 길이 코딩(CAVLC: Context-based Adaptive Variable Length Coding) 방식이 사용될 수 있다. 산술 코딩은 연속적인 데이터 심볼들을 하나의 소수로 변환하는 데, 산술 코딩은 각 심볼을 표현하기 위해 필요한 최적의 소수 비트를 얻을 수 있다. 산술 코딩으로서 컨텍스트 기반 적응형 산술 부호화(CABAC: Context-based Adaptive Binary Arithmetic Code)가 이용될 수 있다.
- [0047] 도 2는 본 발명의 일 실시예에 따른 비디오 신호 디코딩 장치의 개략적인 블록도이다. 도 2를 참조하면, 본 발명의 비디오 신호 디코딩 장치는 크게 엔트로피 디코딩부(210), 역양자화부(220), 역변환부(225), 필터링부(230), 프레임 저장부(240), 인터 예측부(250), 인트라 예측부(260)를 포함한다.
- [0048] 엔트로피 디코딩부(210)는 비디오 신호 비트스트림을 엔트로피 디코딩하여, 각 매크로블록의 변환 계수, 움직임 벡터 등을 추출한다. 역양자화부(220)는 엔트로피 디코딩된 변환 계수를 역양자화하고, 역변환부(225)는 역양자화된 변환 계수를 이용하여 원래의 화소값을 복원한다. 한편, 필터링부(230)는 블록 왜곡 현상을 감소시키기 위해 각각의 코딩된 매크로블록에 적용된다. 필터는 블록의 가장자리를 부드럽게 하여 디코딩된 픽처의 화질을 향상시킨다. 필터링을 거친 픽처는 출력되거나 참조 픽처로 이용하기 위하여 프레임 저장부(240)에 저장된다. 인터 예측부(250)는 프레임 저장부(240)에 저장된 참조 픽처를 이용하여 현재 픽처를 예측하는데, 이 때 도 1과 함께 설명된 바와 마찬가지로, 참조 픽처를 이용하게 된다. 또한, 엔트로피 디코딩부(210)로부터 전달받은 화면간 예측정보(참조 픽처 인덱스 정보, 움직임 벡터 정보 등)를 이용할 수 있다.

- [0049] 한편, 인트라 예측부(260)에서는 현재 블록 내의 디코딩된 픽셀로부터 화면내 예측을 수행하게 된다. 인트라 예측부(265) 또는 인터 예측부(260)로부터 출력된 예측값, 및 역변환부(225)로부터 출력된 화소값이 더해져서 복원된 비디오 프레임이 생성된다.
- [0050] 이하, 본 발명에서는 먼저 인코딩 장치의 인트라 예측부(170), 인터 예측부(160), 디코딩 장치의 인트라 예측부(260) 및 인터 예측부(250)에서 비디오 신호를 효율적으로 디코딩하기 위한 새로운 여러가지 실시예를 제시한다. 먼저, 인트라 예측부(170, 260)의 제 1 실시예 내지 제 4 실시예에 대하여 설명하고, 그 후 인터 예측부(160, 250)의 제 5 실시예 내지 제 9 실시예에 대하여 설명하고자 한다. 제 1 실시예는 도 3 내지 도 6b를 참조하면서, 제 2 실시예는 도 7 및 도 8b를 참조하면서, 제 3 실시예는 도 9 및 도 10을 참조하면서, 제 4 실시예는 도 11 내지 도 12를 참조하면서 설명하도록 한다.
- [0051] 도 3은 본 발명의 제 1 실시예의 확장 예측 모드(Extension Prediction Mode)가 채용되는 주변 픽셀 영역을 나타내는 것이고, 도 4a 내지 도 4c는 확장 예측 모드를 이용하기 위하여 인터polate된 주변 픽셀을 나타내는 것이다. 또한, 도 5는 확장 예측 모드를 채용한 비디오 신호의 인코딩/디코딩 순서를 나타내는 순서도이다. 도 6a는 비디오 신호 인코딩 장치 중 인트라 예측부의 제 1 실시예의 구성을 나타내는 것이고, 도 6b는 비디오 신호 디코딩 장치 중 인트라 예측부의 제 1 실시예의 구성을 나타내는 것이다.
- [0052] 먼저, 도 3를 참조하면, 하나의 프레임은 복수개의 인트라 블록 단위들의 집합으로 구성될 수 있다. 상기 단위 블록은 인트라 4*4 블록일 수 있고, 인트라 8*8 또는 인트라 16*16 일 수 있다. 일반적으로 인트라 N*N 블록은 9 가지의 예측 모드(Prediction Mode)를 이용하여 인트라 예측을 수행할 수 있으며, 상기 예측 모드(Prediction Mode)는 현재 블록의 왼쪽에 위치하는 4 개의 픽셀 및 상단의 8 개의 픽셀을 이용하여 예측(Prediction)을 수행한다. 그러나 도 3에 도시된 바와 같이, 현재 블록이 경계 영역에 위치하는 경우 인트라 예측을 위하여 이용되는 픽셀이 존재하지 아니할 수 있다. 이러한 경계 영역은 상단부 중 가장 왼쪽 블록(A), 상단부(B), 좌측부(C), 및 우측부(D) 일 수 있다.
- [0053] 프레임의 상단부 중 가장 왼쪽 블록(A)은, 예측이 이용되는 현재 블록의 왼쪽 픽셀 및 상부 픽셀이 모두 존재하지 아니하기 때문에, 9 가지의 예측 모드 중 평균값(Mean value)를 이용하는 DC 모드(모드 2)만이 채택될 수 있다. 상단부(B)는, 현재 블록의 상부 픽셀이 존재하지 아니하므로, 예측 모드 1(Horizontal direction), 모드 2(DC), 및 모드 8(Horizontal Up direction)만이 이용될 수 있으며, 좌측부(C)는 모드 0(Vertical direction), 모드 2(DC), 모드 3(Diagonal Down Left), 및 모드 7(Vertical Left)만이 이용 가능하다. 또한, 우측부(D)에 해당하는 블록은, 모드 3(Daigonal Down Left) 및 모드 7(Vertical Left)을 제외한 6 가지 예측 모드만을 이용할 수 있다.
- [0054] 이와 같이, 경계 영역에 위치하는 블록들(A, B, C, D)은 이용할 수 있는 예측 모드에 제한을 갖지만, 현재 엔트로피 코딩부(180, 210)에서 가변 길이 방식(VLC: Variable Length Coding)으로 코딩되는 경우, 무조건 최대 4비트가 전송되도록 제한되어 있다. 따라서, 비어있는 예측 모드만큼 비트손실이 발생하게 된다.
- [0055] 본 발명의 제 1 실시예에서는, 현재 블록이 경계 영역(A, B, C, D)에 위치하는 경우, 현재 존재하는 주변 픽셀을 이용하여 비어있는 주변 픽셀을 인터polate하는 단계를 더 포함한다. 따라서, 경계 영역에 위치하는 블록들도 제한되지 아니한 9 가지 예측 모드를 모두 이용할 수 있게 되며, 이러한 예측 모드를 확장 인트라 예측 모드(Extension Intra Prediction Mode)라고 지칭한다.
- [0056] 본 발명의 확장 인트라 예측 모드(Extension Intra Prediction Mode)를 이용하기 위하여, 경계 영역에 위치하는 블록의 주변 픽셀을 인터polate하는 방법을 이하에서 살펴보기로 한다.
- [0057] 현재 블록이 상단부 중 가장 왼쪽 블록(A)인 경우, 비어있는 모드 0, 1, 3~7에 각각 다른 DC 예측값(predictor)를 대입하여 전송한다. 예를 들어, 현재 모드 2에서 갖는 128의 DC 예측값 이외에, 모드 0에는 108의 DC 예측값을, 모드 1에는 118을, 모드 3에는 138, 모드 4에는 148, 모드 5에는 98, 모드 6에는 88, 모드 7에는 158, 모드 8에는 168을 대입하여 전송함으로써, 가장 좋은 코딩 효율 및 복원율이 좋은 값을 선택할 수 있도록 한다.
- [0058] 현재 블록이 상단부(B), 좌측부(C), 및 우측부(D)에 위치하는 경우, 인터polate된 주변 픽셀에 대하여 도 4a 내지 도 4c에 도시한다.
- [0059] 먼저, 도 4a는 현재 블록이 상단부(B)에 위치하는 경우의 인터polate된 픽셀을 나타낸 것이다. 도 4a를 참조하면, 현재 블록의 왼쪽에 위치하는 I, J, K, L 픽셀은 A 블록에 속하는 픽셀로 기준에 존재하지만, 상부의

x, a ~ h 픽셀은 원래 존재하지 아니한다. 따라서, 기존에 존재하는 I, J, K, L 픽셀을 이용하여 상부의 x, a ~ h 픽셀을 인터폴레이팅함으로써 생성할 수 있다.

[0060] x, a ~ h 픽셀을 인터폴레이팅하기 위하여 여러가지 방법을 이용할수 있다. 예를 들어, x, a ~ h 픽셀에 I 픽셀을 그대로 인터폴레이팅하여 이용할 수 있으며, I, J 픽셀의 평균값(Mean Value)를 x, a ~ h 픽셀에 인터폴레이팅할 수 있다. 또한, I 내지 L 픽셀값을 a 내지 d 픽셀과 e 내지 h 픽셀값에 그대로 반영하고, I 내지 L 픽셀의 평균값(Mean Value)을 x 픽셀로 인터폴레이팅 하는 방법을 채택할 수 있다. x, a 내지 h 픽셀을 인터폴레이팅하는 방법은 상기의 I 내지 L 픽셀을 이용하는 방법에 한정되지 아니하며, 복원된 A 블록(현재 블록의 왼쪽 블록)에 나머지 픽셀을 이용할수도 있으며, 이러한 방법에 한정되지 아니한다.

[0061] 먼저, 도 4b 는 현재 블록이 좌측부(B)에 위치하는 경우의 인터폴레이팅된 픽셀을 나타낸 것이다. 도 4b를 참조하면, 현재 블록의 상부에 위치하는 a 내지 h 픽셀은 복원된 A, B 블록에 속하는 픽셀로 기존에 존재하지만, 왼쪽의 x, I 내지 L 픽셀은 기존에 존재하지 아니하는 픽셀이다. 따라서, 기존에 존재하는 a 내지 h 픽셀을 이용하여 왼쪽의 x, I 내지 L 픽셀을 인터폴레이팅함으로써 생성할 수 있다.

[0062] 상기 x 픽셀 및 I 내지 L 픽셀을 생성하기 위하여, 예를 들어, a 픽셀을 x 픽셀 및 I 내지 L 픽셀에 그대로 인터폴레이팅하여 이용할 수 있고, a 내지 d 픽셀의 평균값을 x 픽셀 및 I 내지 L 픽셀에 인터폴레이팅 할 수 있으며, a 내지 d 픽셀의 평균값을 x 픽셀로 인터폴레이팅하고, a 내지 d 픽셀값을 I 내지 L 픽셀값에 그대로 반영할 수 있다. 그러나, 상기 인터폴레이팅 방법은 이에 한정되지 아니하고, 복원된 A 블록 및 B 블록의 나머지 픽셀을 이용할 수 있다.

[0063] 먼저, 도 4c 는 현재 블록이 우측부(C)에 위치하는 경우의 인터폴레이팅된 픽셀을 나타낸 것이다. 도 4c를 참조하면, 현재 블록의 상부 및 왼쪽에 위치하는 a 내지 h 픽셀과 I 내지 L 픽셀은 복원된 A, B 블록에 속하는 픽셀로 기존에 존재하지만, 상부 오른쪽의 e 내지 h 픽셀은 기존에 존재하지 아니하는 픽셀이다. 따라서, 기존에 존재하는 a 내지 h 픽셀 및 I 내지 L 픽셀을 이용하여 e 내지 h 픽셀을 인터폴레이팅함으로써 생성할 수 있다.

[0064] 상기 e 내지 h 픽셀을 생성하기 위하여 여러가지 인터폴레이팅 방법이 이용될 수 있다. 예를 들어, a 내지 d 픽셀의 평균값을 e 내지 h 픽셀값으로 이용할 수 있고, I 내지 L 픽셀의 평균값을 e 내지 h 픽셀값으로 이용할 수 있으며, d 픽셀값을 e 내지 h 픽셀값으로 이용할수도 있다. 상기 인터폴레이팅 방법은 이에 한정되지 아니하며, 복원된 A 블록 및 B 블록의 나머지 픽셀을 이용할 수 있다.

[0065] 도 5 및 도 6a 을 참조하면, 비디오 신호의 인코딩 장치 중 인트라 예측부(170)는 인터폴레이팅 결정부(610), 인터폴레이팅 블록 생성부(620), 및 예측 모드 결정부(630)를 포함한다. 먼저, 현재 블록에 대한 정보가 입력되면, 인터폴레이팅 결정부(610)는 현재 블록이 경계 영역(화면 상단부 중 가장 왼쪽 블록, 상단부, 좌측부, 우측부)에 해당되는지 판단하여 인터폴레이팅을 수행할지 여부를 결정한다. 현재 블록이 경계 영역에 해당되는지 여부는 현재 블록의 위치 정보에 의하여 판단한다(S510단계). 예를 들어, 현재 블록의 위치는 현재 블록을 구성하는 픽셀값을 이용하여 알 수 있다. 한편, 인터폴레이팅 결정부(510)는 인터폴레이팅이 결정되는 경우, 현재 블록의 주변 픽셀이 인터폴레이팅을 이용하여 생성되었는지 여부를 나타내는 인터폴레이팅 정보(interpolating information)을 생성하여 비트스트림에 포함시킬 수 있다. 상기 인터폴레이팅 정보는 현재 블록이 경계 영역 중 어느 위치에 존재하는지를 나타낼 수 있고, 인터폴레이팅이 수행되었는지 여부를 나타낼 수도 있으며, 예를 들어, 플래그 형태로 존재할 수도 있다.

[0066] 인터폴레이팅 픽셀 생성부(620)는 현재 블록의 위치 정보를 이용하여 현재 블록이 경계 영역에 해당된다고 판단되는 경우, 주변 블록 및 현재 블록에 대한 정보를 입력받아, 주변 블록의 픽셀을 이용하여 인터폴레이팅 픽셀을 생성한다(S520 단계), 인터폴레이팅 블록 생성부(620)에서 기존에 존재하지 아니하였던 인터폴레이팅 픽셀을 생성하는 방법은 도 4a 내지 도 4c를 참조하여 위에 언급한 방법과 같다.

[0067] 만일 인터폴레이팅 결정부(610)에서 현재 블록이 경계 영역에 위치하지 아니한다고 판단되는 경우, 현재 블록의 인트라 예측을 위한 주변 픽셀이 모두 존재하므로 현재 블록에 대한 정보는 예측 모드 결정부(630)로 입력된다.

[0068] 인터폴레이팅을 수행함으로써, 현재 블록의 모든 주변 픽셀이 생성된 후, 주변 블록에 대한 정보, 현재 블록에 대한 정보는 예측 모드 결정부(630)로 입력된다. 예측 모드 결정부(630)는 인터폴레이팅된 주변 픽셀들을 이용하여 현재 블록의 9 가지의 인트라 예측을 수행한 후, 예측 모드를 결정한다(S530 단계). 또한, 결정된 예측 모드에 따라 예측된 블록(프리딕터)과, 현재 블록과의 차이인 레지듀얼을 생성하고, 예측 모드 정보와 레지듀얼 및 앞서 생성된 인터폴레이팅 정보를 비트스트림에 포함시켜 디코딩 장치에 전송한다(S540단계).

[0069] 도 5 및 도 6b 를 참조하면, 디코딩 장치(의 엔트로피 디코딩부)는 예측 모드 정보, 인터폴레이팅 정보 및 레지

듀얼을 수신한다(S550 단계). 이 후, 디코딩 장치의 인트라 예측부(260)의 인터폴레이팅 블록 복원부(640)는 수신된 인터폴레이팅 정보 및 주변블록을 이용하여 인터폴레이팅 블록을 복원한다(S560 단계). 디코딩 장치의 인터폴레이팅 블록을 복원하는 단계는 도 4A 내지 도 4C를 참조하여 설명한 바와 같다.

[0070] 현재 블록 복원부(650)는 S560 단계에서 생성된 인터폴레이팅 블록, 및 S550에서 수신된 예측 모드 정보를 이용하여 현재 블록의 인트라 프리딕션을 생성하고(S570 단계), 상기 인트라 프리딕션과 수신된 레지듀얼을 이용하여 현재 블록을 복원한다(S580 단계).

[0071] 이와 같이, 본 발명의 제 1 실시예는 주변 픽셀을 인터폴레이팅함으로써 현재 블록의 인트라 예측에 이용함으로써, 현재 블록이 경계 영역에 위치하더라도 인트라 예측을 이용하여 코딩할 수 있는 이점이 있다.

[0072] 또한, 본 발명의 제 2 실시예는 인트라 예측을 이용하여 현재 프레임을 복원하는데 있어서, 블록 단위로써 지그재그(zig-zag) 방향으로 복원하는 것과 달리, 현재 프레임을 구성하는 복수개의 블록들이 동시에 픽셀 단위로 복원되는 새로운 인트라 예측 모드를 제안한다. 즉, 현재 프레임에서 좌측 상단에 위치하는 블록이 인트라 예측을 수행한 후, 우측 블록의 인트라 예측을 순차적으로 진행하는 것이 아니라, 현재 프레임을 구성하는 복수개의 블록들에서 기 설정된 순서에 따라 픽셀들이 순차적으로 인트라 예측을 수행한다. 이러한 현재 프레임의 인트라 예측 방법을 프레임-픽셀 예측 모드(Frame-pixel Prediction Mode)라고 지칭하며, 도 7 내지 도 8p를 참조하여 자세히 설명하기로 한다.

[0073] 도 7은 본 발명의 제 2 실시예에 따른 프레임-픽셀 예측 모드를 수행하는 순서를 나타내는 순서도이다. 도 7에 나타난 바와 같이, 현재 프레임이 프레임-픽셀 예측 모드(Frame-pixel Prediction Mode)를 채택하는 경우, 현재 블록은 먼저 복원된 이전 프레임에 위치하는 주변 블록을 이용하여 현재 블록의 제 1 픽셀을 예측한다(S710 단계). 이 때, 현재 블록에서 가장 먼저 인트라 예측되는 제 1 픽셀은 현재 블록에서 가장 오른쪽 하단에 위치하는 픽셀일 수 있으나, 이에 한정되지는 아니한다. 그 후, 제 1 픽셀을 이용하여 인접 픽셀의 인트라 예측을 수행하고, 픽셀 예측 정보(Pixel Prediction Information) 및 레지듀얼을 생성한다(S720 단계). 상기 픽셀 예측 정보(Pixel Prediction Information, 이하 PPI라고 함)은 인접 픽셀의 인트라 예측에 이용된 예측 픽셀(프리딕터), 예측 방법(Prediction Method), 예측 방향(Prediction Direction) 중 어느 하나 이상을 포함할 수 있다. 또한, 레지듀얼은 픽셀 예측 정보에 따라 예측된 픽셀(프리딕터)과, 현재 픽셀과의 차이이며, 픽셀 단위로 생성될 수 있다.

[0074] 인접 픽셀의 인트라 예측이 수행되면, 본 발명의 인코딩 장치(100)의 인트라 예측부(170) 및 디코딩 장치의 인트라 예측부(260)는 현재 블록의 인트라 예측이 모두 완료되었는지 판단(S730 단계)하고, 완료된 경우에는 S720 단계에서 생성된 픽셀 예측 정보(PPI) 및 레지듀얼을 전송한다(S740 단계).

[0075] 반면, 현재 블록의 인트라 예측이 모두 완료되지 아니한 경우에는, 이미 예측된 픽셀들을 이용하여 인접 픽셀을 예측하고, 픽셀 예측 정보(PPI) 및 레지듀얼을 생성하는 단계를 반복한다. 이 때, 예측이 완료된 픽셀을 이용하여 예측되는 인접 픽셀들을 예측 순서에 따라 제 2 픽셀, 제 3 픽셀, ..., 제 N 픽셀이라고 지칭한다. 상기 제 1 픽셀 내지 제 N 픽셀의 인트라 예측을 수행하는 방법에 대하여는 추후 도 8a 내지 도 8p를 참조하여 상세히 살펴보기로 한다.

[0076] 본 발명의 제 2 실시예에 따른 디코딩 장치의 인터 예측부(260)는, 먼저 픽셀 예측 정보(PPI) 및 레지듀얼을 수신하고(S750 단계), 수신된 픽셀 예측 정보(PPI) 및 레지듀얼을 이용하여 현재 블록의 제 1 픽셀을 복원한다(S760 단계). 상기 제 1 픽셀의 복원을 위하여 이용되는 레지듀얼은 다른 픽셀과는 달리, 이전 프레임에 위치하는 픽셀과 제 1 픽셀과의 차이값으로 이루어진다.

[0077] 그 후, 복원된 제 1 픽셀, 픽셀 예측 정보, 및 레지듀얼을 이용하여 제 2 픽셀을 복원하고(S770 단계), 현재 블록의 복원이 완료되었는지 판단한 후(S780 단계), 완료되지 아니한 경우 다시 복원된 제 1 픽셀 및 제 2 픽셀, 픽셀 예측 정보, 및 레지듀얼을 이용하여 제 3 픽셀을 복원하며, 이는 현재 블록의 복원이 완료될 때까지 계속된다. 본 발명의 제 2 실시예에 따른 인코딩 장치 및 디코딩 장치의 인트라 예측부(170, 260)에서 현재 블록을 예측 및 복원하는 방법에 관하여는 도 8A 내지 도 8P를 참조하여 서술한다.

[0078] 본 발명의 제 2 실시예에 따른 프레임-픽셀 인트라 예측 모드(Frame-pixel Intra Prediction Mode)는, 도 8a에 나타난 바와 같이, 이전 프레임에 위치하는 픽셀들을 이용하여 현재 블록 중 가장 오른쪽 하단에 위치한 제 1 픽셀(도 8a의 픽셀 0)을 가장 먼저 인트라 예측한다. 이 때, 제 1 픽셀을 예측하기 위한 예측 모드는 종래의 9 가지 예측 모드(Prediction Mode)를 모두 이용할 수 있으며, 인트라 예측과 동시에 인코더 장치의 인트라 예측부(170)에서는 픽셀 예측 정보(PPI)와 레지듀얼이 생성된다. 상기 제 1 픽셀의 픽셀 예측 정보는 이전 프레임

에 위치하며 픽셀 예측에 이용된 픽셀들(프리딕터)일 수 있고, 예측 방법(Prediction Method)는 9 가지 예측 모두 중 가장 효율성이 높은 예측 모드일 수 있으며, 레지듀얼은 제 1 픽셀의 예측에 이용된 픽셀들과, 제 1 픽셀 간의 차이일 수 있다.

[0079] 제 2 픽셀의 인트라 예측은 제 1 픽셀(0)을 이용하여 수행될 수 있다. 본 발명의 제 2 실시예에 따른 프레임-픽셀 예측 모드는 제 2 픽셀 내지 제 N 픽셀(제 1 픽셀은 제외)의 인트라 예측 모드로 다음과 같은 두 가지 방법을 이용하나, 이에 한정되지는 아니한다.

[0080] 첫번째 방법은, 인접한 픽셀들의 평균값을 이용하여 예측하는 방법(Averaging Mode)이고, 두번째 방법은 방향성에 따라 예측하는 방법(Directional Mode)이다. 즉, 첫번째 방법은 인트라 예측을 수행하려는 현재 픽셀과 가장 인접한 픽셀들의 평균값을 예측 픽셀(프리딕터)로 하여, 현재 픽셀의 인트라 예측을 수행한다. 두번째 방법은 먼저 현재 픽셀을 중심으로 가능한 8 가지의 방향 각각에 대하여 가장 가까운 두 픽셀간의 차이를 계산하여, 가장 작은 차이값을 갖는 방향을 예측 방향(Prediction Direction)으로 결정한다. 선택된 예측 방향에서 현재 픽셀과 가장 가까운 두 픽셀의 평균값이 현재 픽셀의 예측 픽셀(프리딕터)로 하여, 현재 픽셀의 인트라 예측을 수행한다. 또한, 인트라 예측과 동시에 생성되는 픽셀 예측 정보와 레지듀얼이 생성된다.

[0081] 도 8b 를 참조하면, 제 2 픽셀(1)은 제 1 픽셀(0)들의 평균값을 이용하여 인트라 예측을 수행할 수 있고, 제 2 픽셀(1)을 중심으로 제 1 방향(제 2 방향의 계산값이 동일하므로 생략)에 위치한 제 1 픽셀(0)들간의 차이를 계산하고, 제 1 픽셀(0)들의 평균값을 예측 픽셀로 하여 인트라 예측을 수행한다. 또한, 인트라 예측과 동시에 생성되는 픽셀 예측 정보와 레지듀얼이 생성된다.

[0082] 도 8c 를 참조하면, 제 3 픽셀(2)은 주변의 예측이 완료된 제 1 픽셀(0) 및 제 2 픽셀(1)을 제 3 픽셀의 예측 픽셀(프리딕터)로 이용하여, 제 3 픽셀의 인트라 예측을 수행한다. 도 8c 에 나타난 바와 같이, 제 3 픽셀(2)는 인접한 제 1 픽셀(0) 및 제 2 픽셀(1)의 평균값을 예측 픽셀로 이용할 수 있다. 또는, 도 8c 의 제 1 방향 및 제 2 방향 각각에서의 가까운 픽셀간의 차이를 구해, 예측 방향을 결정하고, 예측 방향에서 가까운 두 개의 픽셀(제 1 방향 선택시 두 개의 제 1 픽셀(0), 제 2 방향 선택시 두 개의 제 2 픽셀(1))의 평균을 예측 픽셀로 이용할 수 있다. 또한, 인트라 예측과 동시에 생성되는 픽셀 예측 정보와 레지듀얼이 생성된다.

[0083] 도 8d 를 참조하면, 제 4 픽셀(3)은 예측이 완료된 제 1 픽셀(0) 내지 제 3 픽셀(2)을 제 4 픽셀의 예측 픽셀(프리딕터)로 이용하여, 제 4 픽셀의 인트라 예측을 수행한다. 도 8D 에 나타난 바와 같이, 제 4 픽셀(3)은 인접한 제 1 픽셀(0), 제 2 픽셀(1), 및 제 3 픽셀(2)들의 평균값을 예측 픽셀로 이용할 수 있다. 또는, 도 8d 의 제 1 방향 내지 제 3 방향 각각에서 가까운 픽셀간의 차이를 구하여, 예를 들어, 제 1 방향에서는 두 개의 제 2 픽셀(1)간의 차이를 구하여, 이를 비교하여 예측 방향을 결정하고(예를 들어, 제 2 방향), 예측 방향(제 2 방향)에서 가까운 두 개의 픽셀(제 3 픽셀(2))의 평균을 예측 픽셀로 이용할 수 있다. 또한, 인트라 예측과 동시에 생성되는 픽셀 예측 정보와 레지듀얼이 생성된다.

[0084] 또한, 도 8e 를 참조하면, 제 5 픽셀(4)은 예측이 완료된 제 1 픽셀(0) 내지 제 4 픽셀(3)을 제 5 픽셀의 예측 픽셀(프리딕터)로 이용하여, 제 5 픽셀의 인트라 예측을 수행한다. 도 8e 에 나타난 바와 같이, 제 5 픽셀(4)은 인접한 제 1 픽셀(0), 제 2 픽셀(1), 및 제 3 픽셀(2)들의 평균값을 예측 픽셀로 이용할 수 있다. 또는, 도 8e 의 제 1 방향 및 제 2 방향 각각에서 가까운 픽셀간의 차이를 구하고, 이를 비교하여 예측 방향을 결정하며, 결정된 예측 방향에서 가까운 두 개의 픽셀의 평균을 예측 픽셀로 이용할 수 있다.

[0085] 예를 들어, 도 8e 를 참조하면, 제 3 픽셀(2) 및 제 4 픽셀(3)을 포함하는 제 1 방향과 제 1 픽셀(0) 및 제 2 픽셀(1)을 포함하는 제 2 방향 각각에서의 두 픽셀간의 차이를 구하고, 이를 비교하여 더 작은 값을 갖는 방향을 예측 방향으로 결정한다. 만일 제 1 방향이 예측 방향으로 선택되는 경우, 제 5 픽셀(4)를 인트라 예측하기 위한 예측 픽셀(프리딕터)는, 제 5 픽셀(4)과 제 1 방향에서 인접한 제 3 픽셀(2)과 제 4 픽셀(3)의 평균일 수 있다. 또한, 인트라 예측과 동시에 생성되는 픽셀 예측 정보와 레지듀얼이 생성된다.

[0086] 이와 같은 방법으로, 도 8f 내지 도 8p 에 나타난 바와 같이, 예측이 완료된 주변 픽셀을 이용하여 제 6 픽셀(5) 내지 제 16 픽셀(15)까지 인트라 예측을 수행할 수 있으며, 인트라 예측과 동시에 각 픽셀과 관련된 픽셀 예측 정보와 레지듀얼이 생성될 수 있다.

[0087] 또한, 제 1 픽셀 내지 제 N 픽셀의 인트라 예측이 수행되고, 각 픽셀에 관한 픽셀 예측 정보와 레지듀얼이 생성될 때마다. 현재 블록의 인트라 예측이 완료되었는지 여부를 판단한다. 현재 블록의 인트라 예측이 완료된 경우, 제 1 픽셀 내지 제 N 픽셀에 대한 픽셀 예측 정보와 레지듀얼을 전송하게 된다.

[0088] 본 발명의 비디오신호의 디코딩 장치의 인트라 예측부(270)는, 제 1 픽셀 내지 제 N 픽셀에 대한 픽셀 예측 정

보와 레지듀얼을 수신받아, 현재 블록의 복원을 수행한다. 먼저, 수신된 제 1 픽셀의 픽셀 예측 정보와 레지듀얼을 이용하여, 현재 블록의 제 1 픽셀을 복원하고, 복원된 제 1 픽셀과 제 2 픽셀의 픽셀 예측 정보 및 레지듀얼을 이용하여 현재 블록의 제 2 픽셀을 복원한다. 이와 같은 방식으로, 현재 블록의 제 N 픽셀까지 순차적으로 복원하여 현재 블록의 복원을 완료할 수 있다.

[0089] 이와 같이, 본 발명의 제 2 실시예에 따른 비디오 신호의 디코딩 방법은 현재 블록이 주변 블록의 픽셀뿐만 아니라, 현재 블록의 인접 픽셀을 인트라 예측에 이용 이용함으로써 보다 정확한 예측이 가능하게 된다. 또한, 인접 픽셀들은 두 개 이상의 방향으로부터 선택된 프리딕터 값을 이용하여 예측하므로, 이산 코사인 변환시 보다 많은 0 의 계수를 갖는 블록이 생성되므로 코딩 효율을 높일 수 있는 이점이 있다.

[0090] 본 발명의 제 3 실시예는 현재 블록을 인트라 예측함에 있어서, 일정한 경우 주변 블록으로부터 예측을 수행하지 아니하고, 주변 블록의 픽셀값을 그대로 이용하는 인트라 스kip 모드(inter skip mode)를 제안한다. 도 9 는 본 발명의 제 3 실시예에 따르는 비디오 영상을 나타내는 것이고, 도 10 은 본 발명의 제 3 실시예에 따른 비디오 신호의 인코딩 및 디코딩 순서를 나타내는 순서도이다.

[0091] 종래의 인트라 예측은 16*16 블록, 8*8 블록, 및 4*4 블록을 이용하며, 9 가지의 인트라 예측 모드를 이용하여 수행된다. 그러나, 화면간 상관도(correlation)가 떨어져 인트라 예측 모드를 이용해야 하는 경우 중, 현재 블록과 주변 블록이 유사한(homogeneous) 경우에는 종래의 인트라 예측을 수행하는 것보다 주변 블록을 그대로 이용하는 것이 더 효율적일 수 있다.

[0092] 도 9 를 참조하면, 도 9의 모자에 나타난 박스(Box A)나 얼굴 윈편의 박스(Box B) 모두 이전 화면과의 상관도가 떨어져 인터 예측(inter prediction)을 이용할 수 없는 경우라 가정한다. 이러한 경우, 인트라 예측을 수행하는데, 박스 내부의 블록들은 유사성(homogeneous)를 가지므로, 주변 블록의 픽셀값을 그대로 현재 블록의 픽셀값으로 하고, 따로 인트라 예측을 수행하지 아니하는 인트라 스kip 모드(inter skip mode)를 이용하는 것이 효율적일 수 있다. 이 때, 예측 모드 정보 및 레지듀얼이 전송되지 아니하는 것은 당연하다.

[0093] 도 10 을 참조하면, 먼저 현재 블록은 인트라 예측과 인터 예측 중 어떤 예측을 수행할지 여부를 결정한다(S1010 단계). 현재 블록의 인트라 예측 또는 인터 예측을 결정하는 것은 종래의 방법과 동일하다. 상기 단계에서 현재 블록이 인트라 예측을 수행할 것으로 결정되는 경우, 인트라 스kip 모드(intra skip mode)는 현재 블록이 종래의 인트라 예측을 수행하는 경우보다 효율이 더 좋은 경우만 채택될 수 있다. 먼저, 현재 블록은 인트라 예측 모드를 이용한 경우와 인트라 스kip 모드를 이용한 경우의 RD cost 를 모두 계산한다(S1020 단계). 이 때, RD cost(이하, 효율) 는 하기 수학식 1 과 같이 계산된다.

[0094] [수학식 1]

$$J = D + \lambda R$$

[0095] 여기서, D 는 SSD(sum of square distortion)이고, R 은 요구비트수이다.

[0096] 효율을 계산하여, 현재 블록이 인트라 스kip 모드를 이용하는 경우, 효율이 더 좋은 경우 인트라 스kip 모드를 채택(S1030 단계의 Yes)하고, 예측 블록을 결정하며, 인트라 스kip 정보 및 선택된 예측 블록 정보를 전송한다(S1040 단계). 상기 인트라 스kip 정보는 현재 블록이 인트라 스kip 모드를 이용하였는지 여부를 나타내는 플래그 정보(intra_Skip_flag)일 수 있다.

[0097] 본 발명의 디코딩 장치의 인트라 예측부(260)는 인트라 스kip 정보 및 예측 블록 정보를 수신한다(S1050 단계). 상기 인트라 스kip 정보가 플래그 정보의 형태인 경우, 인트라 스kip 모드 플래그 정보의 의미는 하기 표 1 과 같다.

[0098] [표1] 인트라 스kip 모드 플래그 정보

분리 인트라 예측 플래그 (intra_skip_flag)	의미(meaning)
0	종래의 인트라 예측 모드를 수행
1	인트라 스kip 모드를 수행

[0100]

- [0101] 수신된 인트라 스kip 모드 플래그 정보(infra_skip_flag)가 1인 경우, 본 발명의 디코딩 장치 중 인트라 예측부(260)는, 예측 블록 정보에 따라 예측 블록의 픽셀값을 현재 블록의 픽셀값으로 그대로 이용한다(S1060 단계).
- [0102] 반면, 현재 블록이 인트라 예측 모드를 이용하도록 결정된 경우,(S1030 단계에서의 No) 인트라 N*N 예측을 수행하고, 레지듀얼 및 예측 모드 정보를 전송한다(S1070 단계). 또한, 본 발명의 디코딩 장치 중 인트라 예측부(260)에서는 상기 레지듀얼 및 예측 모드 정보를 수신하여, 종래의 인트라 예측 방법을 이용하여 현재 블록을 복원한다(S1080 단계).
- [0103] 이와 같이, 본 발명의 제 3 실시예에 따라 현재 블록의 인트라 예측을 위하여, 인트라 스kip 모드를 이용하는 경우, 인터 예측이 비효율적이며 주변 블록과 유사(homogeneous)한 비디오 신호의 예측 및 복원에 효율적일 수 있다. 또한, 인트라 예측을 수행할 필요가 없고, 레지듀얼 및 CBP 를 전송할 필요가 없으므로 사용되는 비트율을 감소시킬 수 있다.
- [0104] 인트라 예측을 수행하는 프레임은 인트라 16*16 블록, 인트라 8*8 블록, 및 인트라 4*4 블록으로 코딩된다. 각각의 모드에서, 특정 블록은 이미 복원된 상부 및 왼쪽에 위치한 블록을 이용하여 예측된다. 특히, 인트라 4*4 블록을 이용하는 모드는 고주파 영역에서 이용되는데, 현재 블록의 인트라 예측을 위하여 예측 블록(프리딕터)를 상부 및 왼쪽에 위치한 블록만을 이용하도록 제한된다. 그러나, 이는 현재 블록을 예측 및 복원하는데 충분하지 못하고, 현재 블록이 경계에 위치하는 경우 비효율적이다. 따라서, 현재 블록의 예측을 위하여 이전에 복원된 블록뿐만 아니라, 다음에 복원될 블록을 이용하는 것이 더 정확한 예측을 가능하게 할 수 있다.
- [0105] 따라서, 본 발명의 제 4 실시예는 인트라 예측 모드의 이러한 문제점을 해결하기 위하여 현재 블록을 포함하는 현재 프레임의 새로운 인트라 예측 방법인 분리 인트라 예측 모드(Separate Intra Prediction Mode)를 제안한다. 하나의 인트라 프레임은 두 개 또는 네 개의 독립적인 서브프레임(sub-frame)으로 구분가능함에 따라, 분리된 프레임에서의 여러가지 분리 인트라 예측 모드가 존재 할 수 있다. 모션을 갖는 연속적인 프레임들과 달리 서브-픽셀이 이동하는 서브 프레임 사이에는 높은 임여성(redundancy)가 존재하므로, 인트라 프레임의 서브 프레임에서의 분리 인트라 예측 모드는 굉장히 효율적일 수 있다.
- [0106] 도 11 은 본 발명의 제 4 실시예에 따른 비디오 신호의 인코딩 및 디코딩 순서를 나타내는 순서도이며, 도 12a 내지 도 12c 는 본 발명의 제 4 실시예의 분리 인트라 예측 모드(Separate Intra Prediction Mode)를 나타내는 예들이다.
- [0107] 먼저, 도 11 을 참조하면, 본 발명의 인코딩 장치의 인트라 예측부(170)는 현재 프레임에 대하여 종래의 인트라 예측을 수행하여, RD cost 를 계산한다(S1110단계). 또한, 본 발명의 제 4 실시예에 따른 인트라 예측부는 분리 인트라 모드 구조 정보(Separate Intra Mode Structure Information, SIMSI)를 입력받는다. 상기 분리 인트라 모드 구조 정보(SIMSI)는 서브 프레임의 종류 및 분리 인트라 모드의 구조에 관한 것을 포함하며, 상기 서브 프레임의 종류는 복수개의 16*16 블록, 8*8 블록, 4*4 블록일 수 있으며, 하나의 16*16 블록, 8*8 블록, 4*4 블록일 수도 있고, 하나 또는 복수개의 픽셀 단위일 수도 있으나, 이에 한정되지 아니한다. 또한, 분리 인트라 모드의 구조는 IP 구조, IPPP 구조, IBBP 구조일 수 있고, 이에 관하여는 추후 도 12a 내지 도 12c 를 참조하여 상세히 서술하기로 한다.
- [0108] 이 후, 현재 프레임에 대하여 종래의 인트라 예측을 수행한 후, 상기 분리 인트라 모드 구조 정보(SIMSI)를 이용하여 분리 인트라 모드의 구조에 따른 현재 프레임의 분리 인트라 예측을 수행한다(S1120 단계). 현재 프레임에 대한 종래의 인트라 예측과 분리 인트라 모드 예측의 경우 효율(RD cost)을 계산하여, 분리 인트라 모드 예측의 효율이 더 좋은 경우, 이를 선택한다(S1130 단계). 그 후, 선택된 분리 인트라 모드의 종류 및 분리 인트라 모드 구조를 포함하는 분리 인트라 모드 정보(separate intra mode information, SIDI 라 함) 및 예측에 이용된 블록(프리딕터)와 현재 블록간의 레지듀얼을 전송한다(S1140 단계). 또한, 분리 인트라 예측 모드가 채택되었는지 여부를 나타내는 분리 인트라 예측 플래그 정보(separate_intra_prediction_flag)를 전송할 수 있다.
- [0109] 본 발명의 디코딩 장치의 인트라 예측부(260)는 만일 분리 인트라 예측 플래그 정보가 인코더로부터 전송된 경우, 이를 수신한다. 분리 인트라 예측 플래그 정보의 의미는 하기 표 2 와 같다.

[0110]

[표2] 분리 인트라 예측 플래그 정보

분리 인트라 예측 플래그 (separate_intra_prediction_flag)	의미(meaning)
0	종래의 인트라 예측 모드를 수행
1	분리 인트라 예측 모드를 수행

[0111]

[0112] 분리 인트라 예측 플래그가 0인 경우, 현재 프레임의 복원을 위하여 종래의 인트라 예측 모드를 수행한다. 반면, 1인 경우에는, 분리 인트라 예측 모드를 수행하므로 상기 분리 인트라 모드 정보(SIMI)와 레지듀얼을 수신 받고(S1150 단계), 분리 인트라 모드 정보, (SIMI) 레지듀얼 및 이전 프레임을 이용하여 현재 프레임의 제 1 서브 프레임 그룹을 복원한다(S160 단계). 또한, 분리 인트라 모드 정보, 레지듀얼 및 상기 제 1 서브 프레임 그룹을 이용하여 제 2 서브 프레임 그룹을 복원한다. S1170 단계).

[0113]

만일, 선택된 분리 인트라 모드 구조가 IPPP 구조인 경우에는, 분리 인트라 모드 정보, 레지듀얼, 상기 제 1 서브 프레임 그룹 및 상기 제 2 서브 프레임 그룹을 이용하여 제 3 서브 프레임 그룹을 복원하며, 다시 분리 인트라 모드 정보, 레지듀얼, 상기 제 1 서브 프레임 그룹, 상기 제 2 서브 프레임 그룹, 상기 제 3 서브 프레임 그룹을 이용하여 제 4 서브 프레임 그룹을 복원한다. 또한, IBBP 구조인 경우는, 분리 인트라 모드 정보, 레지듀얼, 상기 제 1 서브 프레임 그룹 및 상기 제 2 서브 프레임 그룹을 이용하여 제 3 서브 프레임 그룹을 복원하며, 분리 인트라 모드 정보, 레지듀얼, 상기 제 1 서브 프레임 그룹 및 상기 제 2 서브 프레임 그룹을 이용하여 제 4 서브 프레임 그룹을 복원한다. 이하에서, 분리 인트라 모드 구조에 따른 분리 인트라 예측 모드의 예측 방법을 도 12a 내지 도 12c를 참조하여 설명한다.

[0114]

도 12a를 참조하면, 현재 프레임(I frame)은 서브 프레임 단위로 제 1 서브 프레임 및 제 2 서브 프레임으로 구분될 수 있으며, 격행주사 방식으로 위치할 수 있으며, 현재 프레임을 양분하여 좌측은 제 1 서브 프레임, 우측은 제 2 서브 프레임일 수도 있으나, 이에 한정되지 아니한다. 도 12a에 나타난 바와 같이, 제 1 서브 프레임과 제 2 서브 프레임이 번갈아가면서 위치할 수도 있으며, 이러한 제 1 서브 프레임들의 집합을 제 1 서브 프레임 그룹으로, 제 2 서브 프레임들의 집합을 제 2 서브 프레임 그룹이라 지칭한다. 먼저, 제 1 서브 프레임 그룹은 종래의 인트라 예측 모드를 이용하여 코딩될 수 있고, 제 2 서브 프레임 그룹은 제 1 서브 프레임 그룹을 참조로 하여 현재 프레임내에서 인트라 예측을 이용하여 코딩될 수 있다.

[0115]

도 12b는 분리 인트라 예측 모드 중 IPPP 구조 및 IPPP 예측 방법을 나타내는 것으로, 현재 프레임은 총 4 가지의 서브 프레임으로 분리될 수 있으며, 각각의 서브 프레임의 집합은 제 1 서브 프레임 그룹 내지 제 4 서브 프레임 그룹일 수 있다. 도 12b에 나타난 바와 같이, 제 1 서브 프레임 그룹이 종래의 인트라 예측 모드를 이용하여 예측을 수행하여 코딩된다. 그 후, 제 1 서브프레임 그룹을 이용하여 제 2 서브 프레임 그룹이 인트라 예측을 수행하여 코딩되고, 제 3 서브 프레임 그룹은 제 1 서브 프레임 그룹 및 제 2 서브 프레임 그룹을 이용하여, 제 4 서브 프레임 그룹은 제 1 서브 프레임 그룹 내지 제 3 서브 프레임 그룹을 이용하여 인트라 예측을 수행한다.

[0116]

도 12c는 분리 인트라 예측 모드 중 IBBP 구조 및 IBBP 예측 방법을 나타내는 것으로, 현재 프레임은 도 12b와 마찬가지로 총 4 가지의 서브 프레임으로 분리될 수 있다. 도 12c에 나타난 바와 같이, IBBP 예측 방법은 먼저 제 1 서브프레임 그룹이 종래의 인트라 예측 모드를 이용하여 인트라 예측을 수행하고, 제 2 서브 프레임 그룹은 제 1 서브 프레임 그룹을 이용하여, 제 3 서브 프레임 그룹은 제 1 서브 프레임 그룹 및 제 2 서브 프레임 그룹을 이용하여, 제 4 서브 프레임 그룹은 제 1 서브 프레임 그룹 및 제 2 서브 프레임 그룹을 이용하여 인트라 예측을 수행한다.

[0117]

이 때, 분리 인트라 예측 모드에서의 서브 프레임의 배치 순서는, 도 12a 내지 도 12c에 나타난 바에 한정되지 아니하며, 여러가지 순서를 선택할 수 있다. 또한, 상기 구조 및 예측 방법은 위에 언급한 바와 같이 분리 인트라 모드 정보에 포함될 수 있다.

[0118]

이와 같이, 본 발명의 제 4 실시예에 따른 분리 인트라 예측 모드를 이용하는 경우, 예측 블록이 현재 블록의 상부 및 왼쪽에 위치한 블록에 한정되지 아니하므로, 현재 프레임의 정확한 예측을 가능하게 할 수 있다.

[0119]

본 발명의 비디오신호의 디코딩 방법은 새로운 인트라 예측 모드 뿐만 아니라, 움직임 벡터 추정 및 보상과 관

련하여 높은 효율을 위한 새로운 모드들을 제안한다. 이하에서는 인터 예측부(160, 250)의 제 5 실시예 내지 제 9 실시예에 대하여 설명한다. 제 5 실시예는 도 13 내지 도 17 를 참조하면서, 제 6 실시예는 도 18 및 도 19 를 참조하면서, 제 7 실시예는 도 20 및 도 21 을 참조하면서, 제 8 실시예는 도 22 를 참조하면서, 도 9 실시 예는 도 23a 및 도 23b 를 참조하면서 설명하도록 한다.

[0120] 비디오 신호의 디코딩 장치 중 인터 예측부(160, 250)에서는 움직임 벡터 프리딕터(Motion Vector Predictor, 이하 mvp 라 함)를 결정하기 위하여 평균값 필터(median filter)를 이용한다. 그러나, 움직임 벡터 프리딕터(mvp)와 추정된 움직임 벡터 사이에 오차가 존재하기 때문에, 정확한 움직임 벡터 프리딕터(mvp)를 찾는 것이 인코더의 중요한 역할이다. 따라서, 본 발명의 제 5 실시예에서는 보다 정확한 움직임 벡터 프리딕터(mvp)를 결정하기 위한 방법(이하, 움직임 벡터 프리딕터 결정법(Motion Vector Competition)이라함)에 대하여 도 13 내지 도 17 을 참조하여 설명하기로 한다.

[0121] 도 13 및 도 14 는 본 발명의 비디오 신호의 인코딩 장치 및 디코딩 장치를 나타내는 것으로, 움직임 벡터 프리딕터 생성부(1350, 1445)를 더 포함한다. 인코딩 장치의 움직임 벡터 프리딕터 생성부(1350)는 기 설정된 방법에 의하여 움직임 벡터 프리딕터 후보 세트(MVP candidate set)를 결정하고, 결정된 세트에서 가능한 움직임 벡터 프리딕터 중 가장 효율이 좋은 움직임 벡터 프리딕터를 결정하며, 결정된 움직임 벡터 프리딕터와 관련된 움직임 벡터 프리딕터 정보(Motion Vector Predictor Information, MVPI)를 전송한다.

[0122] 이와 같이, 보다 정확한 움직임 벡터 프리딕터(mvp)를 움직임 벡터 추정 및 보상을 위하여 이용하기 위하여, 본 발명의 제 5 실시예는 보다 많은 움직임 벡터 프리딕터(mvp)의 후보군을 설정하고, 이 중 가장 효율적인 움직임 벡터 프리딕터(mvp)를 이용한다. 먼저, 움직임 벡터 프리딕터(mvp)는 종류에 따라크게 공간적 프리딕터(spatial predictor), 시간적 프리딕터(temporal predictor), 및 공간-시간적 프리딕터(spatial-temporal predictor)로 나눌 수 있다.

[0123] 도 15 는 현재 블록의 움직임 벡터 예측을 위하여 이용되는 현재 프레임 및 이전 프레임의 움직임 벡터를 나타내는 것이다. 도 15 를 참조하면, 공간적 프리딕터(spatial predictor)는 현재 프레임(N 프레임)의 움직임 벡터를 이용하는 것으로, 현재 블록의 움직임 벡터를 mv 라고 할 때, N 프레임의 a, b, c 블록의 움직임 벡터를 이용한 mva, mvb, mvc, mvd 및 3 개의 움직임 벡터 프리딕터 중 중간값(median)을 채택하는 mv_med 및 3 개 이상의 움직임 벡터 프리딕터 중 중간값(median)을 채택하는 mv_spaEXT 를 포함할 수 있다. 도 15 를 참조하면, 시간적 프리딕터(temporal predictor)는 이전 프레임(N-1 프레임)의 움직임 벡터를 이용하는 것으로, 이전 프레임에서 현재 블록과 동일한 위치에 존재하는 블록의 모션 벡터를 이용하는 mv_col 과 mv_col, mvt0, mvt3 의 중간값(median)을 이용하는 프리딕터 및 MVcol, MVt0...MVt8 의 중간값(median)을 이용하는 프리딕터를 포함할 수 있다. 또한, 공간-시간적 프리딕터(spatial-temporal predictor)는 mv_col, mv_col, mva, mvb, mvc의 평균값을 이용하는 움직임 벡터 프리딕터(mv_spf)일 수 있다. 움직임 벡터 프리딕터 중 필수적으로 이용되는 공간적 프리딕터를 제 1 움직임 벡터 프리딕터라고 하고, 본 발명의 제 5 실시예에서 추가적으로 이용하는 공간적 프리딕터 또는 시간적 프리딕터를 제 2 움직임 벡터 프리딕터라고 지칭한다. 도 14에 나타난 바와 같이, 상기 제 1 움직임 벡터 프리딕터는 중간 움직임 벡터(mv_med)일 수 있고, 상기 제 2 움직임 벡터 프리딕터는 현재 블록의 왼쪽에 위치하는 블록의 움직임 벡터를 이용하는 a-움직임 벡터 프리딕터(mva) 또는 mv_col 일 수 있으나, 이에 한정되지 아니한다.

[0124] 상기와 같이, 고정된 중간값을 이용하여 움직임 벡터 프리딕터를 결정하지 아니하고, 다양한 후보군(candidate)로부터 움직임 벡터 프리딕터를 결정하므로 움직임 벡터 프리딕터의 정확도가 높아질 수 있다. 그러나, 상기와 같이 움직임 벡터 프리딕터로 10 가지 후보군을 고정하는 것은 다양한 움직임을 갖는 화면을 코딩하는데 있어 비효율적일 수 있다.

[0125] 따라서, 본 발명의 제 5 실시예는 새로운 움직임 벡터 프리딕터 후보군 세트(Motion Vecfor Predictor Candidate Set, MVPCS)를 설정하고, 이를 결정하기 위한 조건을 제시한다. 상기 움직임 벡터 프리딕터 후보군 세트는 하기 표 3 과 같다.

[0126]

[표3] 움직임 벡터 프리딕터 후보군 세트의 종류

Set	움직임 벡터-x 성분	움직임 벡터-y 성분
1	Sx	Sy
2	Sx	Sy or Ty
3	Sx or Tx	Sy
4	Sx or Tx	Sy or Ty

[0127]

[0128]

여기서, S 는 공간적 프리딕터(spatial predictor)를 나타내며, T 는 시간적 프리딕터(temporal predictor)를 나타낸다.

[0129]

또한, 상기 움직임 벡터 프리딕터 후보군 세트를 위한 공간적 프리딕터는 mv_med 만을 이용하고, 시간적 프리딕터로는 mvcol를 이용할 수 있으나, 이에 한정되지는 아니한다. 만일 상기 움직임 벡터 프리딕터 후보군 세트로 세트 4 가 결정되는 경우, 가능한 움직임 벡터 프리딕터는 (Sx-Sy, Sx-Ty, Tx-Sy, Tx-Ty) 4 가지 경우가 존재하므로, 각 경우의 움직임 비용(Motion cost)를 기초로 하여 더욱 명확한 움직임 벡터 프리딕터를 결정할 수 있게 된다.

[0130]

도 16a 및 도 16b 는 본 발명의 제 5 실시예의 움직임 벡터 프리딕터 결정법을 따르는 경우, 가능한 움직임 벡터 프리딕터의 조합을 나타내는 것이다. 도 16a 는 현재 블록이 스kip 모드인 경우를 나타내고, 도 16b 는 스kip 모드 이외의 경우를 나타낸다.

[0131]

종래의 스kip 모드는 이전 프레임의 움직임 벡터 프리딕터를 현재 블록의 움직임 벡터로 이용하는 것으로, 하나의 움직임 벡터 프리딕터 후보만이 존재한다. 반면, 본 발명의 움직임 벡터 프리딕터 결정법에 따른 방법은 움직임 벡터 프리딕터의 후보가 제 1 움직임 벡터 프리딕터 뿐만 아니라, 제 2 움직임 벡터 프리딕터까지 이용하므로, 최대 4 개의 조합가능한 후보가 존재할 수 있다. 도 16a 를 참조하면, 현재 블록의 제 1 움직임 벡터 프리딕터가 A 이고, 제 2 움직임 벡터 프리딕터가 B 일 수 있으며, 이들의 조합으로 움직임 벡터 후보는 a(제 1 움직임 벡터 프리딕터), b(제 2 움직임 벡터 프리딕터), c(제 1 움직임 벡터 프리딕터 x 성분-제 2 움직임 벡터 프리딕터 y 성분) 또는 d(제 2 움직임 벡터 프리딕터 x 성분-제 1 움직임 벡터 프리딕터 y 성분)일 수 있다. 이 경우, 움직임 벡터 프리딕터를 위한 움직임 벡터 프리딕터 정보는 2 비트를 요구할 수 있다.

[0132]

또한, 도 16b 를 참조하면, 현재 블록이 스kip 모드가 아니며, 분할(partitioning)된 경우에는, 움직임 벡터 프리딕터 정보가 보다 많은 비트를 요구하므로, 먼저 제 1 움직임 벡터 프리딕터와 제 2 움직임 벡터 프리딕터 사이의 거리(mv1-mv2)를 측정한다. 측정된 제 1 움직임 벡터 프리딕터 및 제 2 움직임 벡터 프리딕터 사이의 거리가 1/2 픽셀 이하로 이하로 두 개의 프리딕터가 가까우면, 제 1 움직임 벡터 프리딕터를 현재 블록의 움직임 벡터 프리딕터로 결정하여, 종래의 움직임 벡터 예측 방법을 이용한다. 이 때, 현재 블록이 종래의 움직임 벡터 예측 방법을 이용하는지 움직임 벡터 프리딕터 결정법(motion vector competition)을 이용하였는지 나타내는 움직임 벡터 프리딕터 결정 플래그(mv_competition_flag)를 이용할 수 있다. 제 1 움직임 벡터 프리딕터가 현재 블록의 움직임 벡터 프리딕터로 결정되면, 종래의 움직임 벡터 예측 방법을 이용하였음을 나타내는 움직임 벡터 프리딕터 결정 플래그(mv_predictor_competition_flag)에 0 을 셋팅하여 전송할 수 있다. 상기 움직임 벡터 프리딕터 결정 플래그 정보의 의미는 다음 표 4 와 같다.

[0133]

[표 4] 움직임 벡터 프리딕터 결정 플래그 정보

Mv_competition_flag	의미(meaning)
0	현재 블록이 MV competition 을 이용하지 아니함 (종래의 MV 결정법 사용)
1	현재 블록이 MV competition 을 이용함

[0134]

도 17는 본 발명의 제 5실시예를 따르는 움직임 벡터 프리딕터의 결정 및 움직임 보상의 순서를 나타내는 순서도이다. 도 17 를 참조하면, 먼저 움직임 벡터 프리딕터 후보군 세트를 결정하기 이전에, 하기 수학식 2 를 계산한다(S1710 단계).

[0136] [수학식 2]

$$MV_{col} > 2 * \max(MV_{spa_max}, MV_{temp_max})$$

[0137]

[0138] 이전 프레임에서 현재 블록과 동일한 위치에 존재하는 블록의 움직임 벡터를 이용한 mv_{col} 의 값이 너무 큰 경우, 이는 이전 프레임과 현재 블록과 상관도(correlation)가 너무 떨어지는 것을 의미한다. 따라서, 상기 수학식 2의 조건을 만족하는 경우(S1710 단계의 Yes)에는 시간적 프리딕터를 이용하지 않고, 공간적 프리딕터만 이용하여 움직임 벡터 프리딕터를 결정한다(S1720 단계). 움직임 벡터 프리딕터로 공간적 프리딕터만 이용할지 여부는 x 성분과 y 성분에 대하여 따로 결정될 수 있다.

[0139]

반면, 상기 수학식 2의 조건을 만족시키지 못하는 경우(S1710 단계의 No), 하기 수학식 3 또는 수학식 4의 조건을 계산하여 이용할 움직임 벡터 프리딕터 후보군 세트를 결정한다(S1730 단계). 수학식 3은 움직임 벡터 스kip 모드(Motion Vector Skip Mode)를 위한 조건이고, 수학식 4은 스kip모드를 제외한 경우를 위한 조건이다.

[0140]

[수학식 3]

$$Set1: S_x = T_x, S_y = T_y$$

$$Set2: S_x = T_x, S_y \neq T_y$$

$$Set3: S_x \neq T_x, S_y = T_y$$

$$Set4: S_x \neq T_x, S_y \neq T_y$$

[0141]

[수학식 4]

$$Set1: |S_x - T_x| < 3, |S_y - T_y| < 3$$

$$Set2: |S_x - T_x| < 3, |S_y - T_y| \geq 3$$

$$Set3: |S_x - T_x| \geq 3, |S_y - T_y| < 3$$

$$Set4: |S_x - T_x| \geq 3, |S_y - T_y| \geq 3$$

[0143]

[0144] 움직임 벡터 프리딕터 생성부(1350, 1445)는 상기 조건을 이용하여 현재 블록의 가장 효율적인 움직임 벡터 프리딕터 후보군을 결정한다. 예를 들어, $mv_med(Sx, Sy)$ 및 $mv_Col(Tx, Ty)$ 의 조건을 검토한 결과, 세트 3이 현재 블록의 움직임 벡터 프리딕터 후보군 세트로 결정된 경우, 가능한 움직임 벡터 프리딕터(mvp) 2 가지($Sx-Sy$ 또는 $Tx-Sy$) 일 수 있다. 이 때, 각 움직임 벡터 프리딕터를 이용할 때의 움직임 비용(Motion cost)를 측정하여, 현재 블록의 움직임 벡터 프리딕터($Sx-Sy$)를 결정한다(S1740 단계). 그 후, 어떤 조합의 움직임 벡터 프리딕터($Sx-Sy$)를 이용하였는지를 나타내는 움직임 벡터 프리딕터 정보(mvp information)을 전송한다(S1750 단계). 이 때, 움직임 벡터 프리딕터 정보는 mv_{col} 이 mv_{spa_max} 및 mv_{temp_max} 중 최대값의 두배 보다 큰 경우(S1710 단계의 Yes)에는, 공간적 프리딕터(Spatial Predictor)에 관한 정보를 포함할 수 있다.

[0145]

[0145] 본 발명의 디코딩 장치 중 움직임 보상부(1445)는 움직임 벡터 프리딕터 정보(mvp Information)을 수신하고, 자체적으로 현재 블록에 대하여 수학식 3 또는 수학식 4의 조건을 계산하여 움직임 벡터 프리딕터 후보군 세트를 결정한다(S1760 단계). 그 후, 수신한 움직임 벡터 프리딕터 정보 및 결정된 세트의 조건을 이용하여, 현재 블록의 움직임 보상을 수행한다(S1770 단계). 이러한 다양한 움직임 벡터 프리딕터 후보를 이용하여, 보다 현재 블록의 움직임 벡터와 유사한 프리딕터를 채택함으로써, 더욱 정확한 움직임 벡터 예측이 가능할 수 있다.

[0146]

[0146] 종래의 움직임 벡터 프리딕터 결정법을 이용하는 경우, 기존의 AVC 방법을 이용하는 경우에 비하여 비트율의 감소가 약 -0.5%인데 반해, 본 발명의 제 5 실시예에 따른 움직임 벡터 프리딕터 결정법을 이용하는 경우에는, 약 -0.8%정도 비트율이 감소하였다. 또한, PSNR(peak signal to noise ratio)에 있어서도, 본 발명의 제 5 실시예에 따른 움직임 벡터 프리딕터 결정법을 이용하는 경우 종래의 방법을 이용하는 경우보다 약 0.028dB 향상되었다.

[0147]

[0147] 이와 같이, 본 발명의 제 5 실시예의 움직임 벡터 프리딕터 결정법은, 다양한 움직임 벡터 프리딕터 후보를 이용하여, 현재 블록의 움직임 벡터와 보다 유사한 움직임 벡터 프리딕터를 채택함으로써, 더욱 정확한 움직임 벡터 예측이 가능한 효과를 갖는다.

- [0148] 또한, 본 발명의 또 다른 실시예인 제 6 실시예에서는 이전 프레임의 움직임 벡터의 평균값 및 현재 블록의 타입에 따라 움직임 벡터 프리디터 후보(Motion Vector Predictor Candidate)를 결정할 수 있다. 움직임 벡터 프리디터 후보 결정방법은 도 18 을 참조하여 설명하기로 한다.
- [0149] 일반적으로 움직임이 큰 프레임들은 연속하기 때문에, 이전 프레임의 움직임 벡터들의 평균적인 크기에 따라 현재 프레임이 이전 프레임과 상관도(correlation)를 갖는지 판단할 수 있다. 이전 프레임의 움직임 벡터들의 평균값이 일정값(threshold) 이상이면, 현재 프레임은 이전 프레임에 비하여 움직임이 크므로, 이전 프레임과의 상관도가 떨어지게 된다. 따라서, 움직임 벡터 프리디터 후보로 두 개 이상의 공간적 프리디터를 이용할 수 있다. 반면, 이전 프레임의 움직임 벡터들의 평균값이 일정값 이하이면, 현재 프레임은 이전 프레임에 비하여 움직임이 적게 되므로, 이전 프레임과 상관도가 커서 이전 프레임의 움직임 벡터를 이용하는 시간적 프리디터를 이용하는 것이 효율적이다. 이 경우, 움직임 벡터 프리디터 후보는 하나 이상의 공간적 프리디터 뿐만 아니라, 하나 이상의 시간적 프리디터를 포함할 수 있다.
- [0150] 따라서, 본 발명의 제 6 실시예에 따른 움직임 벡터 프리디터 후보 결정 방법은 이전 프레임의 움직임 벡터 크기의 평균값(mv_pre_avg)이 1 이하인지 여부를 판단한다(S1810 단계). 이전 프레임의 움직임 벡터의 평균값(mv_pre_avg)가 1 보다 큰 경우, 이는 현재 블록을 포함하는 현재 프레임이 움직임이 큰 프레임임을 나타낸다. 이 경우, 본 발명의 제 6 실시예에서와 같이, 현재 블록과 이전 프레임이 상관도가 적으므로 공간적 프리디터(spatial predictor) 두 개를 이용하는 것이 효율적이다. 그러나, 현재 블록이 서로 유사성이 적은 서브 블록으로 이루어진 경우는 그러하지 아니하다. 따라서, 현재 블록의 블록 타입이 스kip 모드(Skip mode) 또는 16*16 mode 인지 검토한다(S1820 단계), 스kip 모드 또는 16*16 모드가 아닌 경우 S1820 단계의 No, 현재 블록은 서브 블록으로 분할(partitioning)되어 있으며, 분할된 서브 블록 간의 상관도(correlation)가 적다는 의미이다. 이 경우는 제 2 움직임 벡터 프리디터로서 공간적 프리디터(spatial predictor)를 이용하는 것(S1830 단계)보다는 시간적 프리디터(temporal predictor)를 이용하는 것이 효율적이다. 현재 블록이 스kip 모드 또는 16*16 모드인 경우에는, 제 1 움직임 벡터 프리디터로 공간적 프리디터를 이용함과 더불어 제 2 움직임 벡터 프리디터로도 공간적 프리디터를 이용하는 것(S1840 단계)이 효율적임은 당연하다.
- [0151] 이전 프레임의 움직임 벡터의 평균값이 1 미만이면(S1810 단계의 Yes) 현재 프레임은 현재 블록과 이전 프레임의 유사성이 큰 느린 영상이므로, 제 1 움직임 벡터 프리디터 및 제 2 움직임 벡터 프리디터로 공간적 프리디터를 이용하는 것이 효율적이다.
- [0152] 이와 같이, 현재 블록과 이전 프레임의 유사성 및 현재 블록의 타입에 따라 이용하는 움직임 벡터 프리디터 후보를 다르게 결정함으로써, 현재 블록의 움직임 예측의 명확성을 증가시킬 수 있게 된다.
- [0153] 본 발명의 제 5 실시예 및 제 6 실시예의 움직임 벡터 프리디터 결정법에서 생성된 움직임 벡터 프리디터 정보는 현재 블록이 8*8 블록인 경우를 제외하고 2 비트를 요구한다. 현재 블록이 8*8 블록이면, 분할(partitioning)에 따라 움직임 벡터 프리디터 정보를 최대 16 비트까지 요구할 수 있으므로, 오히려 종래의 움직임 벡터 예측 방법이 효율적일 수 있다. 따라서, 블록마다 움직임 벡터 프리디터 결정법을 따르는지 여부를 나타내는 움직임 벡터 프리디터 결정 플래그 정보(mv_competition_flag)를 포함할 수 있다.
- [0154] 하기 표 5 는 본 발명의 제 6 실시예의 움직임 벡터 프리디터 후보 결정법과 종래의 방법을 이용하는 경우의 효과를 비교하기 위한 실험에서 사용된 조건을 나타내는 것이고, 표 6 은 표 5 의 조건하에서 움직임 예측을 수행할 때 이용되는 움직임 벡터 후보를 나타내는 것이다.

[0155]

[표 5]

프레임수	150(15Hz)/300(30Hz)
움직임 추정	Full Search
레퍼런스의 수	4
움직임 벡터 탐색 범위	± 32
GOP 구조	IPPP
엔트로피 코딩법	CAVLC
양자화 계수(QP)	22/27/32/37 for I slice 23/28/33/38 for P slice

[0156]

[0157]

[표 6]

Case 1	스킵 모드	mv_spaEXT, mva
	스킵 모드 이외	mv_med, mv_col
Case 2	스킵 모드	mv_med, mva
	스킵 모드 이외	mv_med, mv_col
Case 3	모든 경우	mv_med, mv_col

[0158]

[0159]

여기서, mv_spaEXT 는 3 개의 벡터를 이용할 수 있는 경우에는, 도 15 의 mva, mvb, mvc 를 이용하고, 그렇지 아니한 경우에는, mva, mvb, mvc 중 가능한 움직임 벡터를 이용하여 결정된다.

[0160]

도 19 는 상기 case 1 내지 3 의 움직임 벡터를 움직임 벡터 프리디터 후보로 갖는 경우와 본 발명의 제 6 실시 예에 따라 결정된 움직임 벡터 프리디터를 이용한 경우의 BD rate 이득을 나타내는 것이다. 상기 BD rate 는 부호화 성능을 나타내는 지표로써 같은 화질을 유지하면서 비트율을 얼마나 압축하였는가를 의미한다.

[0161]

도 19 를 참조하면, 거의 대부분의 경우에서 case 1 내지 3 의 움직임 벡터 프리디터 후보를 이용한 경우보다 본 발명의 제 6 실시예를 따른 경우 비트율의 압축정도가 증가하였음을 알 수 있고, 실질적으로 본 발명의 제 5 실시예에 따른 비디오 신호의 디코딩 방법에 비하여, 약 0.1dB 정도의 PSNR[dB]가 증가됨을 알 수 있다.

[0162]

또한, 본 발명의 또 다른 실시예인 제 7 실시예에서는 현재 블록의 움직임 추정이 수행되는 블록과 주변 블록의 거리를 고려하여 생성된 움직임 벡터 프리디터를 이용하는 새로운 방법을 도 20 및 도 21 을 참조하여 설명하기로 한다.

[0163]

종래의 현재 블록의 움직임 벡터 프리디터는 현재 블록의 주변 3 개의 블록을 이용하여 생성한다. 도 20 은 현재 블록 및 주변 3 개의 블록을 나타내는 것이고, 상기 주변 3 개의 블록은 현재 블록의 왼쪽 블록, 상부 블록 및 상부 오른쪽 블록을 나타낸다. 도 20 을 참조하면, 현재 블록의 움직임 벡터를 mvc 라고 지칭하고, 현재 블록의 왼쪽 블록의 움직임 벡터를 mvL, 상부 블록의 움직임 벡터를 mvU, 상부 오른쪽의 움직임 벡터를 mvUR 이라고 지칭한다. 현재 블록의 움직임 벡터 프리디터를 결정하기 위하여는, 주변 3 개 블록의 레퍼런스 인덱스 중 현재 블록의 레퍼런스 인덱스와 동일한 것이 있는지 살펴본다. 현재 블록의 레퍼런스 인덱스와 같은 레퍼런스 인덱스를 갖는 주변 블록이 1 개인 경우, 상기 주변 블록의 움직임 벡터를 움직임 벡터 프리디터로 결정한다. 그렇지 아니하면(동일한 레퍼런스 인덱스를 가리키는 주변 블록이 1개가 아닌 경우), 주변 3 개의 블록의 움직임 벡터 중 중간값(median)을 선택하여 현재 블록의 움직임 벡터 프리디터로 결정하고, 레퍼런스 인덱스는 주변 3 개의 블록의 레퍼런스 인덱스 중 가장 작은 수로 결정한다.

[0164]

그러나, 현재 블록의 레퍼런스 인덱스가 주변 3 개의 블록의 레퍼런스 인덱스가 아닌 다른 레퍼런스 인덱스를

갖는 경우, 상기와 같은 방법으로 결정된 움직임 벡터 프리딕터는 현재 블록의 움직임 벡터와 큰 차이를 가질 수 있다. 따라서, 결정되는 움직임 벡터 프리딕터가 현재 블록의 움직임 벡터와 유사하게 하기 위하여는, 본 발명의 제 7 실시예에 따른 '스케일된 움직임 벡터 프리딕터(Scaled Motion Vector Predictor)를 이용하는 효율적이다.

[0165] 도 21은 본 발명의 제 7 실시예에 따른 스케일된 움직임 벡터 프리딕터를 이용하여 움직임 예측을 수행하는 방법을 나타내는 순서도이다. 도 21을 참조하면, 움직임 벡터 예측을 수행하는데 있어서, 현재 블록의 레퍼런스 인덱스와 동일한 값을 갖는 주변 블록이 있는지 검토한다(S2110 단계). 만일, 현재 블록의 레퍼런스 인덱스와 동일한 값을 갖는 주변 블록이 있는 경우에는 종래의 움직임 벡터 프리딕터 결정방법을 따른다(S2120 단계). 그러나, 현재 블록의 레퍼런스 인덱스가 주변 블록의 레퍼런스 인덱스와 일치하지 아니하는 경우(S2110 단계의 No)에는, 스케일된 움직임 벡터 프리딕터(Scaled Motion Vector Predictor, SMVP)들을 계산한다(S2130 단계). 스케일된 움직임 벡터 프리딕터는 하기 수학식 5를 이용하여 구할 수 있다.

[수학식 5]

$$\begin{aligned} MV_L' &= \frac{f_{ME} - f_c}{f_L - f_c} * MV_L \\ MV_U' &= \frac{f_{ME} - f_c}{f_U - f_c} * MV_U \\ MV_{UR}' &= \frac{f_{ME} - f_c}{f_{UR} - f_c} * MV_{UR} \end{aligned}$$

[0167]

[0168] 여기서, f_{ME} 는 현재 블록에 대하여 움직임 추정이 수행되는 블록의 프레임 넘버를 나타내며, f_c 는 현재 블록의 프레임 넘버, f_L, f_U, f_{UR} 는 각각 현재 블록의 왼쪽 블록, 상부 블록, 상부 오른쪽 블록의 프레임 넘버를 나타낸다.

[0169] 현재 블록의 움직임 벡터 프리딕터는 상기 스케일된 움직임 벡터 프리딕터들로부터 결정되고, 결정된 스케일된 움직임 벡터 프리딕터를 이용하여 현재 블록의 움직임 벡터 예측을 수행하며(S2140 단계), 현재 블록의 움직임 벡터 차이값(mvd), 레퍼런스 인덱스 및 스케일된 움직임 벡터 프리딕터 정보를 전송한다(S2150 단계). 또한, 스케일된 움직임 벡터 프리딕터를 현재 블록의 움직임 벡터 예측에 이용하였는지 여부를 나타내는 스케일된 움직임 벡터 프리딕터 플래그 정보(scaled_mv_predictor_flag)를 전송할 수도 있다. 상기 스케일된 움직임 벡터 프리딕터 플래그 정보가 이용되는 경우, 상기 플래그 정보의 의미는 하기 표 7과 같다.

[표 7] 스케일된 움직임 벡터 프리딕터 플래그 정보

scaled_mv_predictor_flag	의미(meaning)
0	현재 블록이 스케일된 MV predictor 를 이용하지 아니함 (종래의 MV predictor 결정법 사용)
1	현재 블록이 스케일된 MV predictor 를 이용함

[0171]

[0172] 본 발명의 디코딩 장치의 인터 예측부(250)에서는, 스케일된 움직임 벡터 프리딕터 정보, mvd 및 레퍼런스 인덱스를 수신하고(S2160 단계), 이들을 이용하여 현재 블록의 움직임 벡터를 복원한다(S2170 단계). 상기 스케일된 움직임 벡터 프리딕터 플래그 정보(scaled_mv_predictor_flag)를 전송하는 경우, 스케일된 움직임 벡터 프리딕터 정보가 전송되지 아니할 수 있다. 이 경우, 스케일된 움직임 벡터 프리딕터 플래그 정보(scaled_mv_predictor_flag)의 값이 1이면, 수신된 레퍼런스 인덱스와 mvd 와 주변 블록의 움직임 벡터를 이용하여 스케일된 움직임 벡터 프리딕터를 구할 수 있고, 이를 이용하여 현재 블록의 움직임 벡터를 복원한다.

[0173] 따라서, 본 발명의 제 7 실시예에 따른 현재 블록의 움직임 벡터를 예측하는 방법은, 현재 블록의 레퍼런스 인덱스가 나타내는 프레임과 주변 블록의 레퍼런스 인덱스가 나타내는 프레임 사이의 거리를 고려하여 움직임 벡터 프리딕터를 결정함으로써, 결정된 움직임 벡터 프리딕터가 현재 블록의 움직임 벡터와 더욱 유사한 값을 갖도록 하여, 예측의 정확도를 증가시킬 수 있다.

[0174] 움직임 벡터를 이용하여 현재 블록을 예측하는 방법은, 움직임이 큰 화면에서 보다 정확하게 현재 블록을 코딩

할 수 있으나 비트 요구율이 크므로, 일정 조건에서는 스kip 모드를 이용한다. 스kip 모드(Skip Mode)는 주변 블록을 움직임 벡터 프리디터로 이용하여 움직임 벡터 차이값(MV difference)를 계산하지 아니하고, 주변 블록의 움직임 벡터(mvP)를 현재 블록의 움직임 벡터로 이용하는 것이다. 스kip 모드를 이용하는 경우, 레지듀얼, 레퍼런스 인덱스, CBP(Coded Block Pattern)등 아무런 정보도 코딩하지 아니한다. 그러나, 현재 H.264에서는 스kip 모드를 채택하는 경우 16*16 블록만을 이용하도록 제한하고 있으므로, 16*16 블록 당 두 개 이상의 움직임 벡터를 갖지 못한다.

[0175] 본 발명의 제 8 실시예에서는 스kip 모드에서 움직임 벡터를 갖는 단위 블록이 다양한 크기를 가질 수 있는 부분 스kip 모드(Partially Skip Mode)를 제안한다. 도 22은 부분 스kip 모드에서 이용하는 블록 사이즈를 나타내는 것이다.

[0176] 도 22를 참조하면, 부분 스kip 모드는 16*16 블록뿐만 아니라, 16*8 블록, 8*16 블록, 8*8 블록을 이용할 수 있다. 종래의 스kip 모드는 16*16 블록당 하나의 움직임 벡터를 갖는 반면, 부분 스kip 모드는 16*16 블록 내에서 최대 4 개의 움직임 벡터(8*8 블록 사용시)를 갖는 것이 가능하다. 따라서, 부분 스kip 모드를 이용하는 현재 블록의 디코딩 방법은 영상의 왜곡을 최소화하는 동시에, 스kip 모드를 더욱 활성화함으로써 코딩 효율을 증가시킬 수 있게 된다.

[0177] 본 발명의 제 8 실시예에 따른 부분 스kip 모드는 새로운 블록 타입을 이용하기 때문에, 부분 스kip 모드를 이용한 블록의 타입에 관한 정보를 더 전송할 수 있다. 블록 타입에 관한 정보는 블록 크기에 따라 skip_16*16, skip_16*8, skip_8*16, skip_P8*8 일 수 있으나 이에 한정되지 아니한다. 또한, 종래의 스kip 모드 또는 본 발명의 부분 스kip 모드를 이용하였는지 여부를 나타내는 부분 스kip 모드 플래그 정보(partial_skip_flag)를 전송할 수 있다. 상기 플래그 정보의 의미는 하기 표 8과 같다.

[표 8] 부분 스kip 모드 플래그 정보

Partially skip_flag	의미(meaning)
0	현재 블록이 부분 스kip 모드를 이용하지 아니함 (종래의 스kip 모드를 이용)
1	현재 블록이 부분 스kip 모드를 이용함

[0179]

[0180] 본 발명의 디코딩 장치의 인터 예측부(250)에서는, 블록 타입 정보, 부분 스kip 모드 플래그 정보를 수신하여, 부분 스kip 모드가 이용되는 경우, 이전 프레임에서 현재 블록과 동일한 위치에 있는 블록의 움직임 벡터를 현재 블록의 움직임 벡터로 이용한다. 이 경우, 스kip 모드가 더욱 활성화되어 코딩 효율을 높일 수 있으며, 종래의 스kip 모드보다 더 많은 움직임 벡터를 이용함으로써 현재 블록의 복원율을 높일 수 있다.

[0181] 한편, 인코더에서 움직임 벡터와 관련된 어떠한 정보도 전송받지 아니하고, 디코더에서 독자적으로 이를 유도하여 이용하는 것이 더 효율적일 수도 있다. 이와 같이, 비디오 신호의 디코딩 장치에서는 현재 블록의 움직임 예측에 필요한 정보를 직접 유도하는 다이렉트 모드(direct mode)를 이용할 수 있다. 이러한 다이렉트 모드는 공간적 다이렉트 모드(spatial direct mode) 및 시간적 다이렉트 모드(temporal direct mode)를 포함하며, 공간적 다이렉트 모드는 디코더상에서 현재 블록의 움직임 벡터 및 레퍼런스 인덱스를 주변 블록으로부터 유도하고, 시간적 다이렉트 모드는 종래의 B 퍽처 다이렉트 모드와 동일한 방법으로 수행된다. 공간적 다이렉트 모드는 시간적 다이렉트 모드와 달리, P8*8 블록이 공간적 다이렉트 모드를 채택하는 경우에도 16*16 블록 단위로 하나의 움직임 벡터만을 유도하여 이용하도록 제한된다.

[0182] 따라서, 본 발명의 제 9 실시예에서는 공간적 다이렉트 모드에서 다양한 움직임 벡터를 이용할 수 있는 8*8 공간적 다이렉트 모드(8*8-based spatial direct mode)를 제안한다. 도 23a 및 도 23b는 종래의 공간적 다이렉트 모드를 채택한 P8*8 블록들의 움직임 벡터와 본 발명의 제 9 실시예를 따른 8*8 블록의 움직임 벡터를 나타낸 것이다.

[0183] 도 23a를 참조하면, 종래의 방법에서 현재 블록이 P8*8 블록이고, 다이렉트 모드를 이용하는 경우, 16*16 블록을 구성하는 P8*8 블록들은 모두 동일한 움직임 벡터 및 레퍼런스 인덱스를 갖는다. 도 23a에 나타난 바와 같이, 블록 A, B, C, D는 16*16 블록의 왼쪽에 위치하는 블록인 a 블록, 상단의 b 블록, 상단 오른쪽의 c 블록을 이용하여 중간값 필터링을 통하여 동일한 움직임 벡터(mv)와 레퍼런스 인덱스를 유도하여 이용한다.

[0184] 반면, 도 23b를 참조하면, 본 발명의 8*8 공간적 다이렉트 모드는 주변 블록을 이용하여 다양한 움직임 벡터를 유도하여 이용한다. 도 23b를 참조하면, A' 블록은 주변 블록인 a' 블록, c' 블록 및 d' 블록의 움직임 벡터

중 중간값(median)을 A' 블록의 움직임 벡터(mva')로 이용한다. B' 블록은 주변 블록인 a' 블록, d' 블록 및 e' 블록의 움직임 벡터를 이용하여 움직임 벡터(mvb')로 이용하고, C' 블록은 b' 블록, c' 블록 및 d' 블록을 이용하여, D' 블록은 b' 블록, d' 블록 및 e' 블록을 이용하여 움직임 벡터(mvc' , mvd')를 유도한다. 이 때, 레퍼런스 인덱스는 코딩 효율을 높이기 위하여 종래의 방법과 같이 이용된 주변 블록의 레퍼런스 인덱스 중 가장 작은 값을 선택한다.

[0185] 이와 같이, 본 발명의 8*8 공간적 다이렉트 모드는 P8*8 블록마다 움직임 벡터를 계산하여 16*16 블록 내에서 두 개 이상의 움직임 벡터를 이용함으로써, 현재 블록이 경계 영역에 위치하거나 작은 물체가 움직이는 경우에 보다 높은 복원율을 가질 수 있는 이점이 있다.

[0186] 한편, 본 발명의 또 다른 일실시예에서는 이산 코사인 변환 단계 사이에 비디오 신호들을 재배열하는 단계를 더 포함함으로써 비트 전송율을 감소시키고, 원 영상과의 오차를 줄이는 비디오 신호의 처리 방법에 대하여 도 24 및 도 25를 참고하여 살펴보기로 한다.

[0187] 도 24를 참조하면, 종래의 이산 코사인 변환은 비디오 신호를 2 차원 주파수 성분으로 변환하며, 변환시 비디오 신호의 블록에 포함된 기저성분들 각각의 크기를 구한 후, 양자화하고 지그재그 스캔(zig-zag scan)을 수행한다. 이러한 이산 코사인 변환을 수행하는 비디오 신호는 입력된 비디오 신호일 수 있고, 레지듀얼 신호일 수 있다. 이산 코사인 변환은 2 차원 변환 방법을 이용하며, 먼저 열(row)방향으로 변환을 수행하고, 행(column)방향으로의 변환을 수행한다.

[0188] 반면, 각 방향으로의 변환 이전에 계수의 값이 적은 블록이 직류(DC)성분으로 모이도록 유도하여 압축율을 더욱 향상시킬 수 있다. 따라서, 본 발명의 제 10 실시예는 각 블록의 레지듀얼 값 및 계수를 고려하여 각 방향으로의 변환이전에 블록을 재배열하는 단계를 더 포함함으로써, 일반적인 이산 코사인 변환을 수행하는 경우 보다 이산 코사인 변환 계수가 직류성분 근처에 모이도록 유도하여 압축율을 향상시킬 수 있다. 이하에서는 재배열을 수행하는 블록들이 레지듀얼 신호일 때에 대하여 설명한다.

[0189] 도 25를 참조하면, 본 발명의 제 10 실시예에 따른 이산 코사인 재배열 변환은 열 방향 및 행 방향으로의 변환 단계 이전에 제 1 재배열 단계 및 제 2 재배열 단계를 포함한다. 상기 제 1 재배열 단계는 열(row) 방향에서 각 블록의 레지듀얼 값을 비교하여 왼쪽에 작은 값을 갖는 블록을 배열하고, 상기 재배열 순서를 나타내는 제 1 재배열 정보를 생성한다. 그 후, 열(row) 방향으로의 이산 코사인 변환을 수행한다. 제 2 재배열 단계는 열(row) 방향으로 이산 코사인 변환된 레지듀얼 값을 기준으로 행(column)방향에서 위쪽에 작은 값을 갖는 블록을 재배열하고, 상기 재배열 순서를 나타내는 제 2 재배열 정보를 생성함으로써 수행된다. 본 발명의 실시예에서는 각 블록의 레지듀얼 값을 비교하여 재배열 하는 방법에 대하여 기재하였으나, 이에 한정되지는 아니한다.

[0190] 이와 같이, 이산 코사인 변환 이전에 열 또는 행에서 적은 레지듀얼 값을 갖는 블록이 왼쪽 또는 위쪽에 위치하도록 레지듀얼 신호들을 재배열하여, 이산 코사인 변환을 수행하는 경우 이산 코사인 변환계수들이 좌측(직류성분 근처)로 몰려 분포하게 되므로 보다 높은 압축효과를 얻을 수 있다.

[0191] 또한, 본 발명이 적용되는 디코딩/인코딩 방법은 컴퓨터에서 실행되기 위한 프로그램으로 제작되어 컴퓨터가 읽을 수 있는 기록 매체에 저장될 수 있으며, 본 발명에 따른 데이터 구조를 가지는 멀티미디어 데이터도 컴퓨터가 읽을 수 있는 기록 매체에 저장될 수 있다. 상기 컴퓨터가 읽을 수 있는 기록 매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 저장 장치를 포함한다. 컴퓨터가 읽을 수 있는 기록 매체의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피디스크, 광 데이터 저장장치 등이 있으며, 또한 캐리어 웨이브(예를 들어 인터넷을 통한 전송)의 형태로 구현되는 것도 포함한다. 또한, 상기 인코딩 방법에 의해 생성된 비트스트림은 컴퓨터가 읽을 수 있는 기록 매체에 저장되거나, 유/무선 통신망을 이용해 전송될 수 있다.

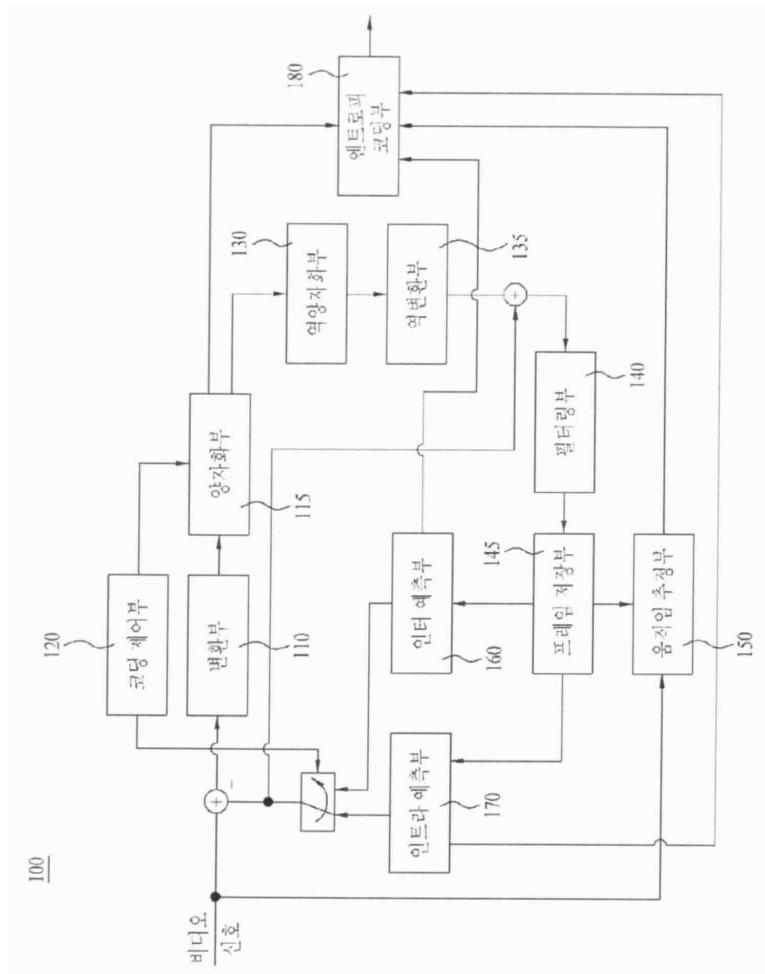
[0192] 이상과 같이, 본 발명은 비록 한정된 실시예와 도면에 의해 설명되었으나, 본 발명은 이것에 의해 한정되지 않으며 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에 의해 본 발명의 기술사상과 아래에 기재될 특허 청구범위의 균등범위 내에서 다양한 수정 및 변형이 가능함은 물론이다.

산업상 이용가능성

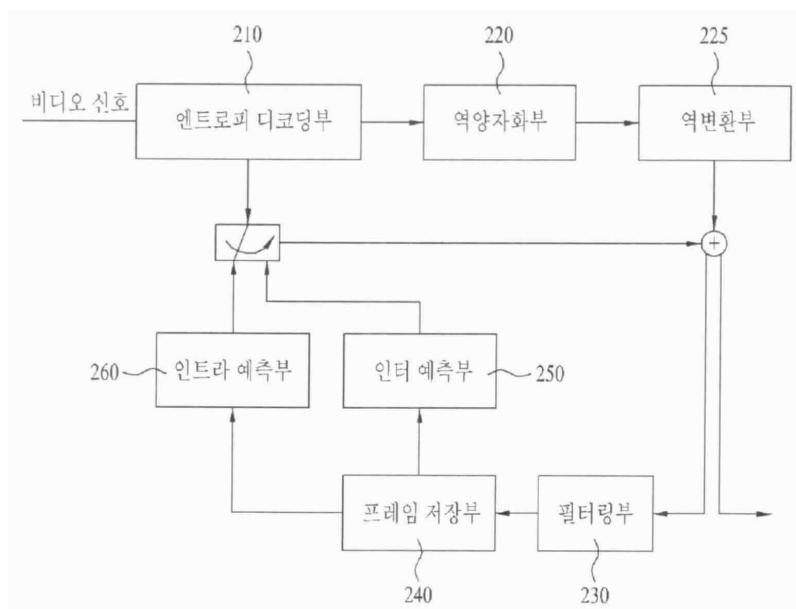
[0193] 본 발명은 비디오를 인코딩하고 디코딩하는 데 적응될 수 있다.

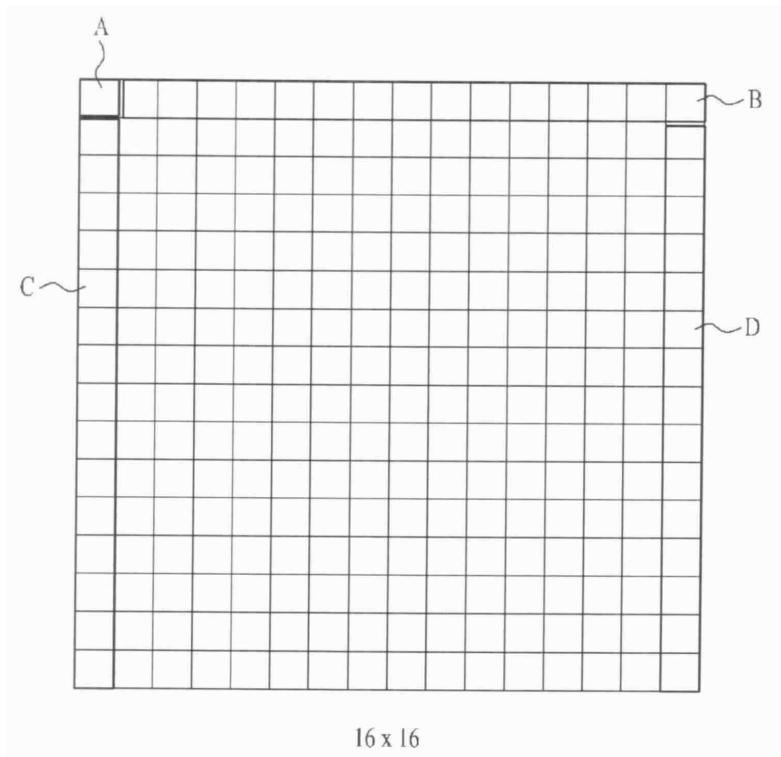
도면

도면1

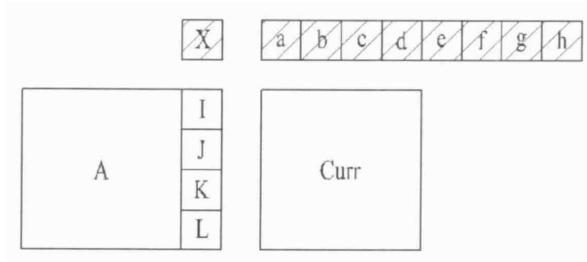
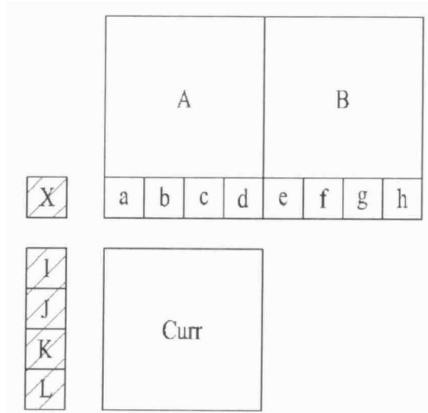


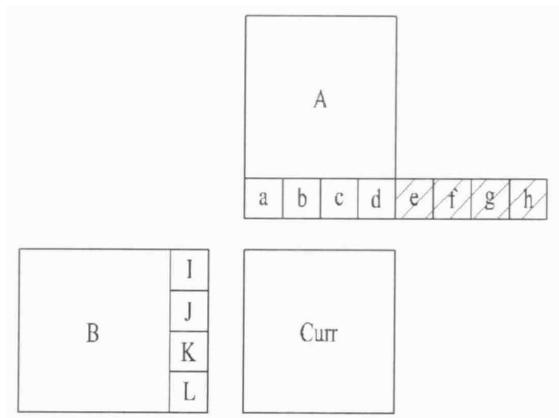
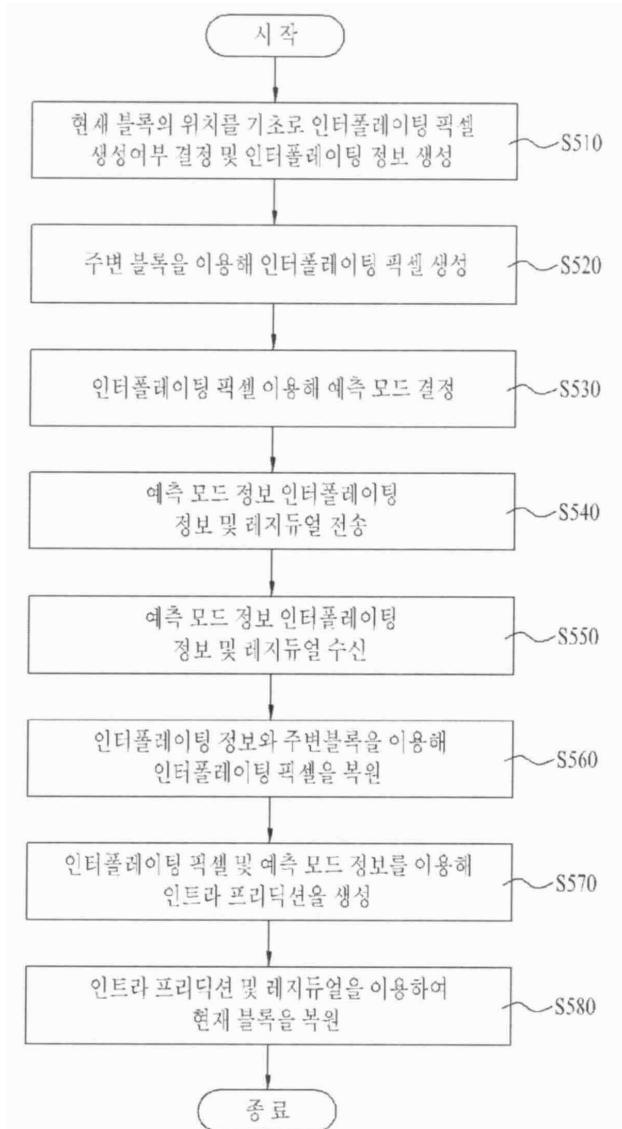
도면2



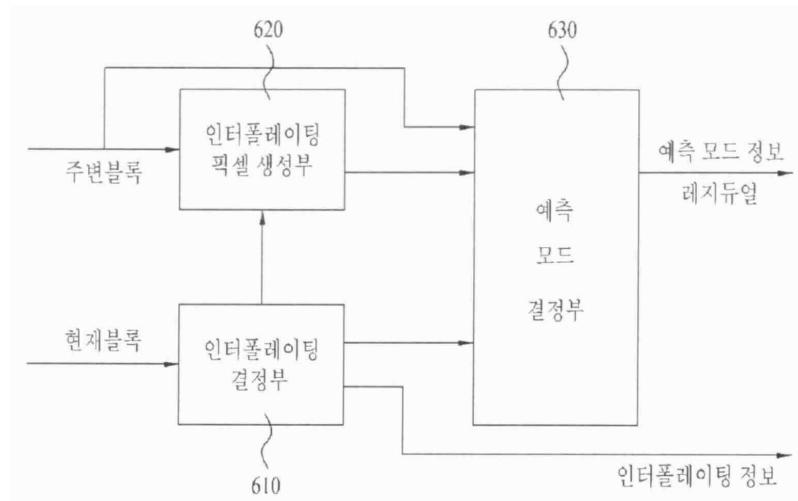
도면3

16 x 16

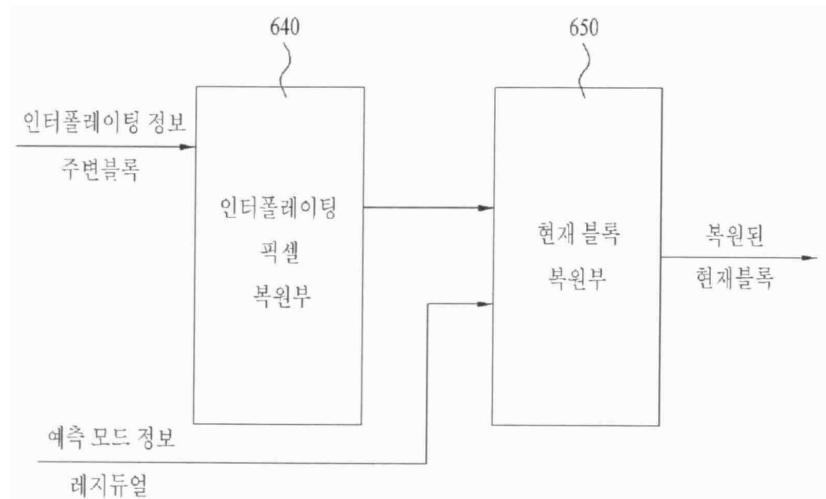
도면4a**도면4b**

도면4c**도면5**

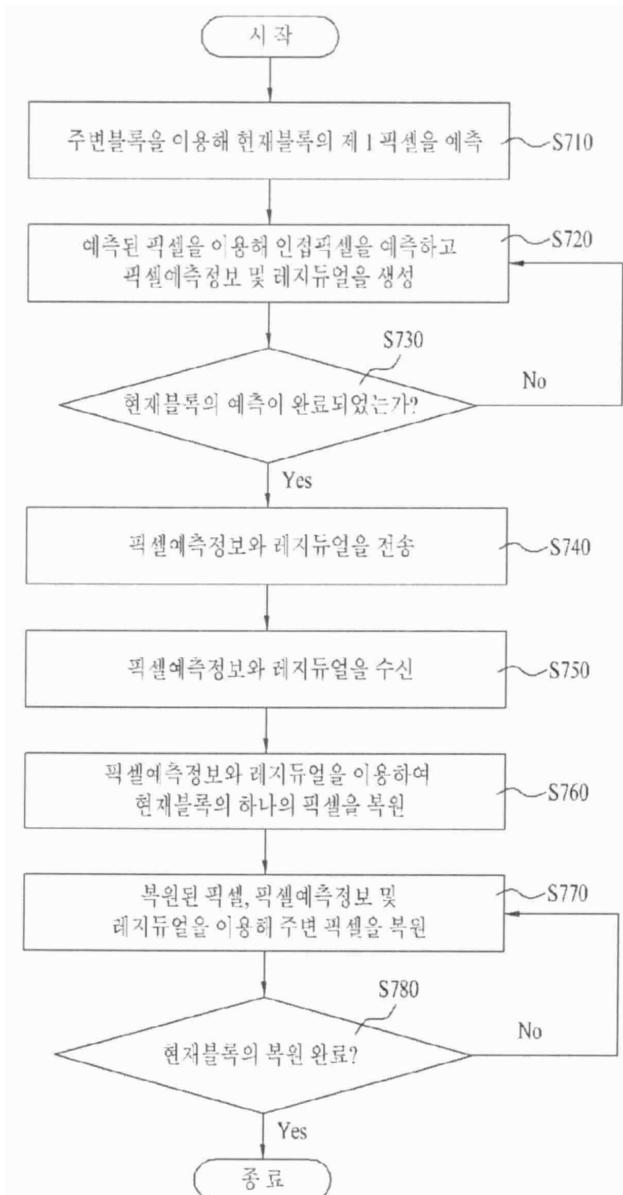
도면6a



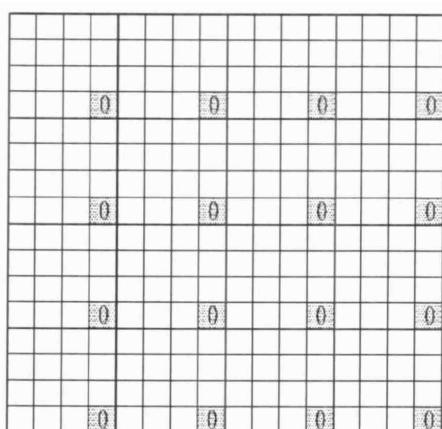
도면6b



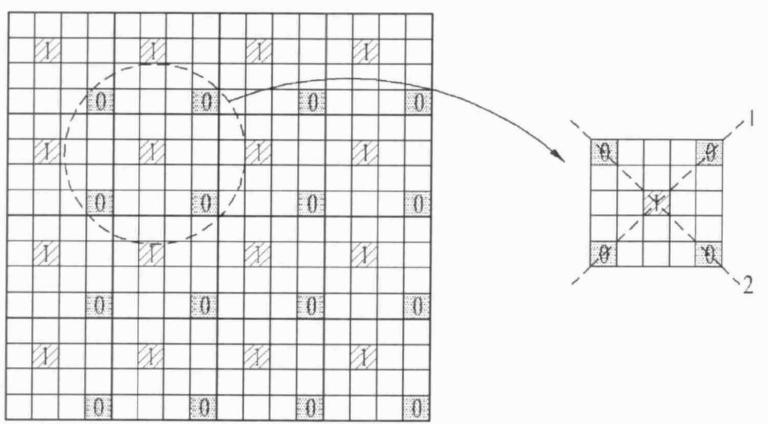
도면7



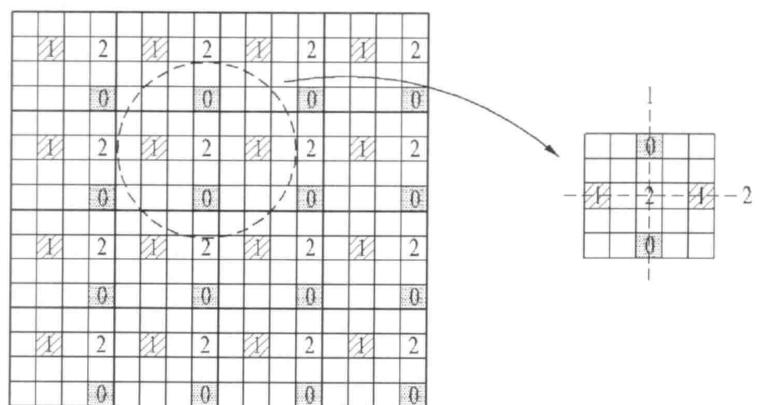
도면8a



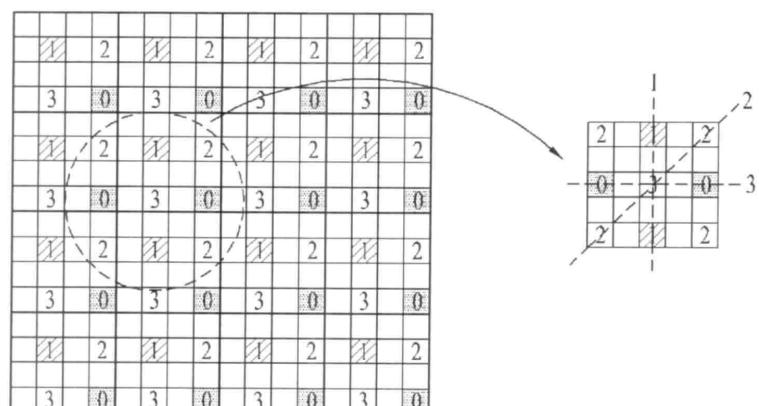
도면8b



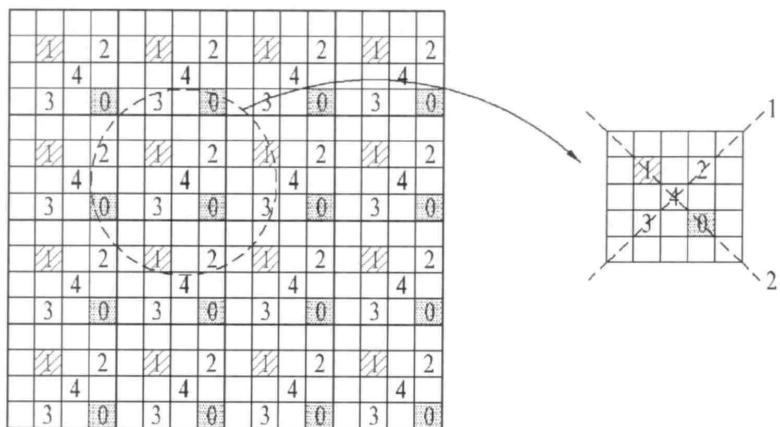
도면8c



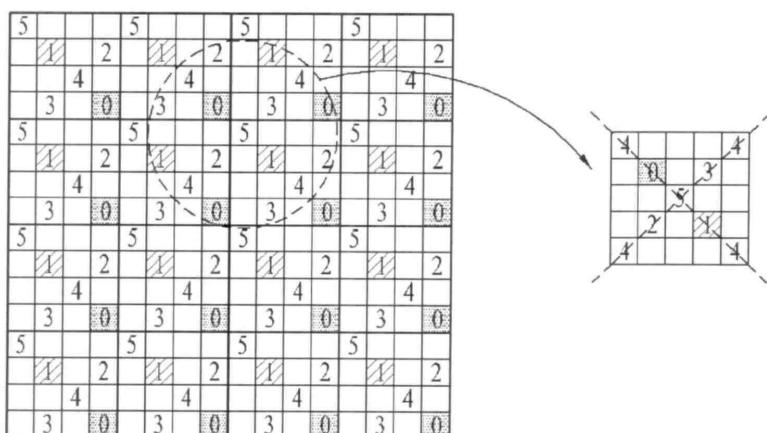
도면8d



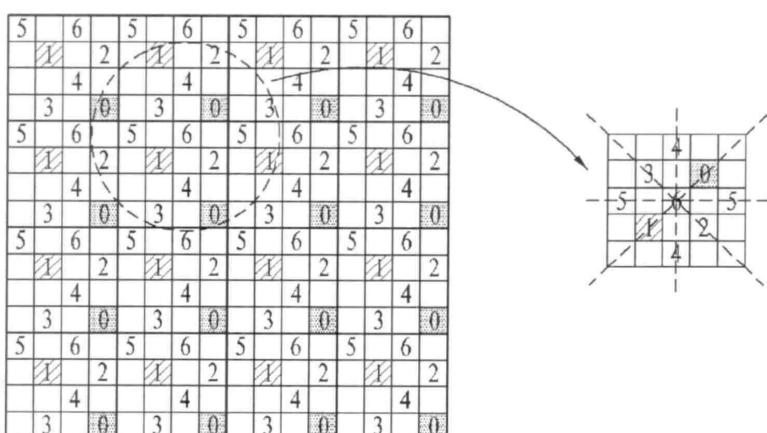
도면8e



도면8f



도면8g



도면8h

5	6	5	6	5	6	5	6
X	2	X	2	X	2	X	2
7	4	7	4	7	4	7	4
3	0	3	0	3	0	3	0
5	6	5	6	5	6	5	6
X	2	X	2	X	2	X	2
7	4	7	4	7	4	7	4
3	0	3	0	3	0	3	0
5	6	5	6	5	6	5	6
X	2	X	2	X	2	X	2
7	4	7	4	7	4	7	4
3	0	3	0	3	0	3	0
5	6	5	6	5	6	5	6
X	2	X	2	X	2	X	2
7	4	7	4	7	4	7	4
3	0	3	0	3	0	3	0

1			
6	5	6	
2	X		
4	3	4	4

도면8i

5	6	5	6	5	6	5	6
X	2	X	2	X	2	X	2
7	4	8	7	4	8	7	4
3	0	3	0	3	0	3	0
5	6	5	6	5	6	5	6
X	2	X	2	X	2	X	2
7	4	8	7	4	8	7	4
3	0	3	0	3	0	3	0
5	6	5	6	5	6	5	6
X	2	X	2	X	2	X	2
7	4	8	7	4	8	7	4
3	0	3	0	3	0	3	0

1			
6	5	6	
2	X		
4	3	7	3

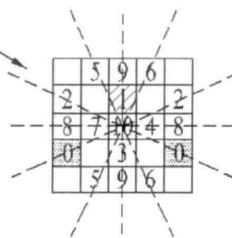
도면8j

5	9	6	5	9	6	5	9	6
X	2	X	2	X	2	X	2	2
7	4	8	7	4	8	7	4	8
3	0	3	0	3	0	3	0	0
5	9	6	5	9	6	5	9	6
X	2	X	2	X	2	X	2	2
7	4	8	7	4	8	7	4	8
3	0	3	0	3	0	3	0	0
5	9	6	5	9	6	5	9	6
X	2	X	2	X	2	X	2	2
7	4	8	7	4	8	7	4	8
3	0	3	0	3	0	3	0	0

1			
8	7	1	4
0	3	0	0
5	6	2	2

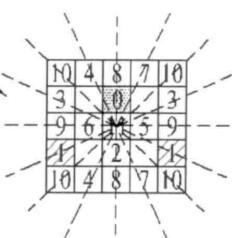
도면8k

5	9	6	5	9	6	5	9	6	5	9	6
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	0	3	0	3	0	3	0	3	0	3	0
5	9	6	5	9	6	5	9	6	5	9	6
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	0	3	0	3	0	3	0	3	0	3	0
5	9	6	5	9	6	5	9	6	5	9	6
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	0	3	0	3	0	3	0	3	0	3	0
5	9	6	5	9	6	5	9	6	5	9	6
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	0	3	0	3	0	3	0	3	0	3	0



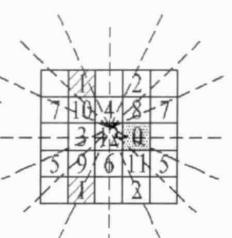
도면8l

5	9	6	11	5	9	6	11	5	9	6	11
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	0	3	0	3	0	3	0	3	0	3	0
5	9	6	11	5	9	6	11	5	9	6	11
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	0	3	0	3	0	3	0	3	0	3	0
5	9	6	11	5	9	6	11	5	9	6	11
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	0	3	0	3	0	3	0	3	0	3	0
5	9	6	11	5	9	6	11	5	9	6	11
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	0	3	0	3	0	3	0	3	0	3	0



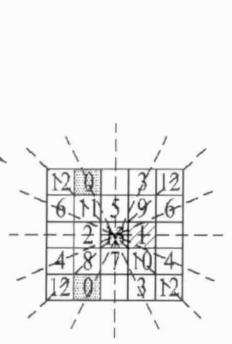
도면8m

5	9	6	11	5	9	6	11	5	9	6	11
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	12	0	3	12	0	3	12	0	3	12	0
5	9	6	11	5	9	6	11	5	9	6	11
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	12	0	3	12	0	3	12	0	3	12	0
5	9	6	11	5	9	6	11	5	9	6	11
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	12	0	3	12	0	3	12	0	3	12	0
5	9	6	11	5	9	6	11	5	9	6	11
2	2	2	2	2	2	2	2	2	2	2	2
7	10	4	8	7	10	4	8	7	10	4	8
3	12	0	3	12	0	3	12	0	3	12	0



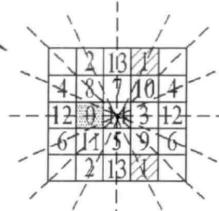
도면8n

5	9	6	11	5	9	6	11	5	9	6	11	5	9	6	11
13		2	13	1		2	13	1		2	13	1		2	
7	10	4	8	7	10	4	8	7	10	4	8	7	10	4	8
	3	12	0		3	12	0		3	12	0		3	12	0
5	9	6	11	5	9	6	11	5	9	6	11	5	9	6	11
13		2	13	1		2	13	1		2	13	1		2	
7	10	4	8	7	10	4	8	7	10	4	8	7	10	4	8
	3	12	0		3	12	0		3	12	0		3	12	0
5	9	6	11	5	9	6	11	5	9	6	11	5	9	6	11
13		2	13	1		2	13	1		2	13	1		2	
7	10	4	8	7	10	4	8	7	10	4	8	7	10	4	8
	3	12	0		3	12	0		3	12	0		3	12	0
5	9	6	11	5	9	6	11	5	9	6	11	5	9	6	11
13		2	13	1		2	13	1		2	13	1		2	
7	10	4	8	7	10	4	8	7	10	4	8	7	10	4	8
	3	12	0		3	12	0		3	12	0		3	12	0



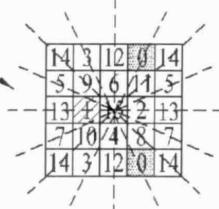
도면8o

5	9	6	11	5	9	6	11	5	9	6	11	5	9	6	11
13		2	13	1		2	13	1		2	13	1		2	
7	10	4	8	7	10	4	8	7	10	4	8	7	10	4	8
14	3	12	0	14	3	12	0	14	3	12	0	14	3	12	0
5	9	6	11	5	9	6	11	5	9	6	11	5	9	6	11
13		2	13	1		2	13	1		2	13	1		2	
7	10	4	8	7	10	4	8	7	10	4	8	7	10	4	8
14	3	12	0	14	3	12	0	14	3	12	0	14	3	12	0
5	9	6	11	5	9	6	11	5	9	6	11	5	9	6	11
13		2	13	1		2	13	1		2	13	1		2	
7	10	4	8	7	10	4	8	7	10	4	8	7	10	4	8
14	3	12	0	14	3	12	0	14	3	12	0	14	3	12	0
5	9	6	11	5	9	6	11	5	9	6	11	5	9	6	11
13		2	13	1		2	13	1		2	13	1		2	
7	10	4	8	7	10	4	8	7	10	4	8	7	10	4	8
14	3	12	0	14	3	12	0	14	3	12	0	14	3	12	0

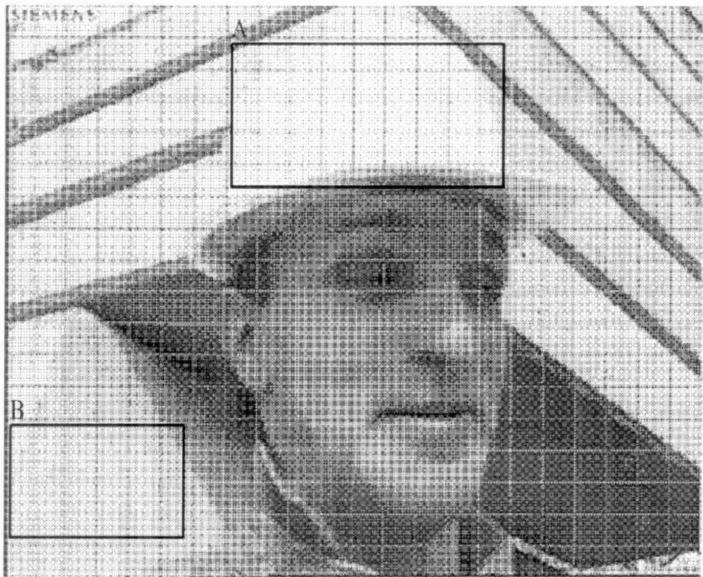


도면8p

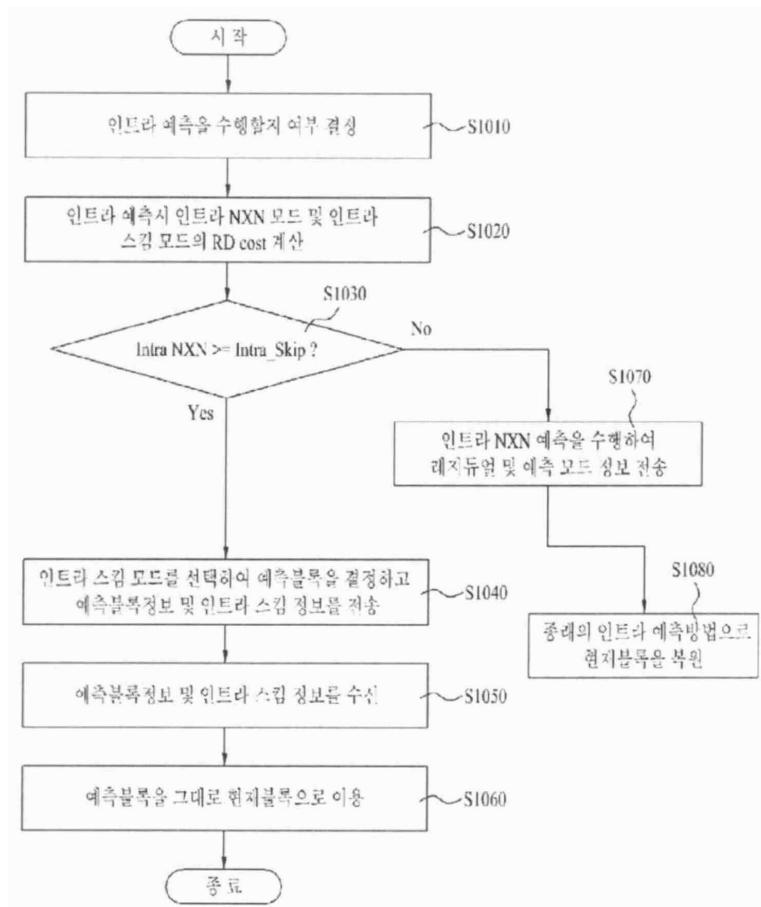
5	9	6	11	5	9	6	11	5	9	6	11	5	9	6	11
13		15	2	13	1	15	2	13	1	15	2	13	1	15	2
7	10	4	8	7	10	4	8	7	10	4	8	7	10	4	8
14	3	12	0	14	3	12	0	14	3	12	0	14	3	12	0
5	9	6	11	5	9	6	11	5	9	6	11	5	9	6	11
13		15	2	13	1	15	2	13	1	15	2	13	1	15	2
7	10	4	8	7	10	4	8	7	10	4	8	7	10	4	8
14	3	12	0	14	3	12	0	14	3	12	0	14	3	12	0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

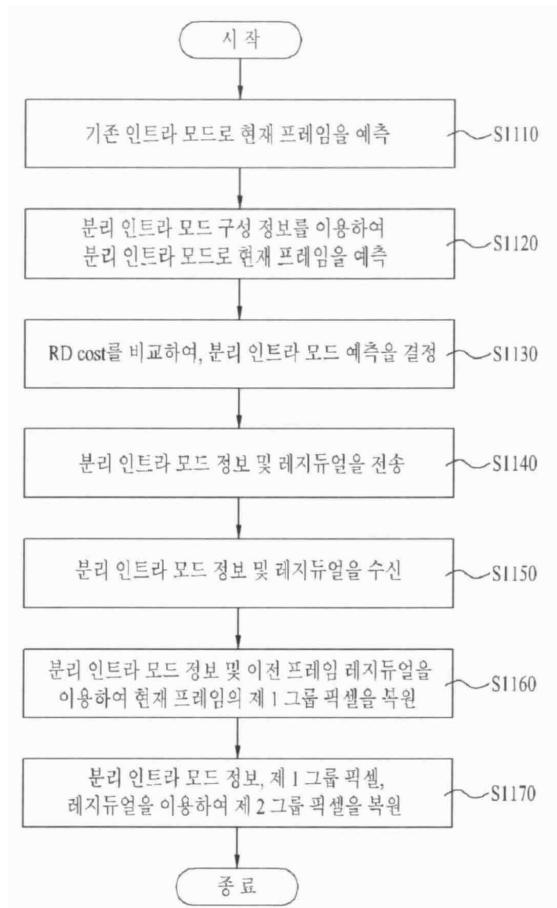
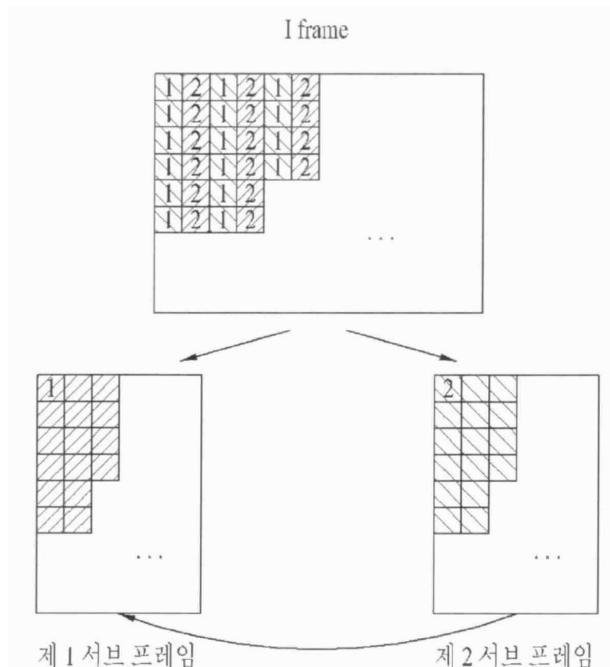


도면9

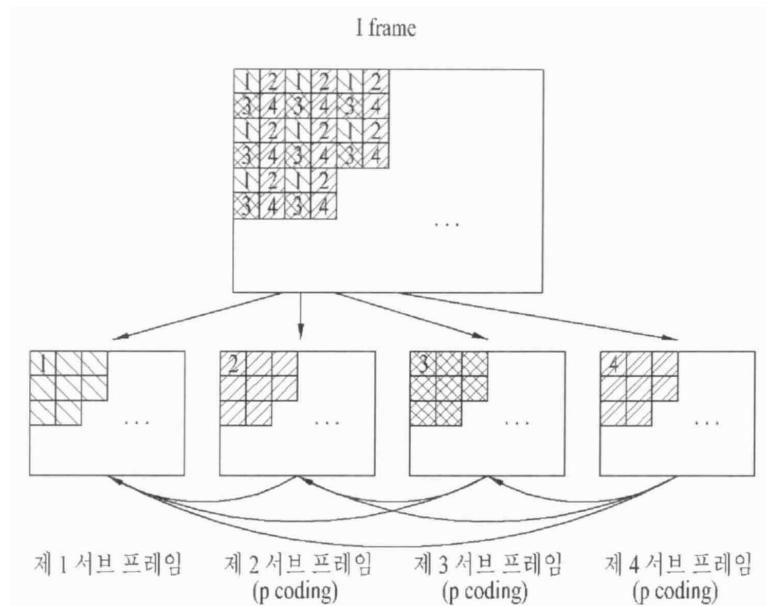


도면10

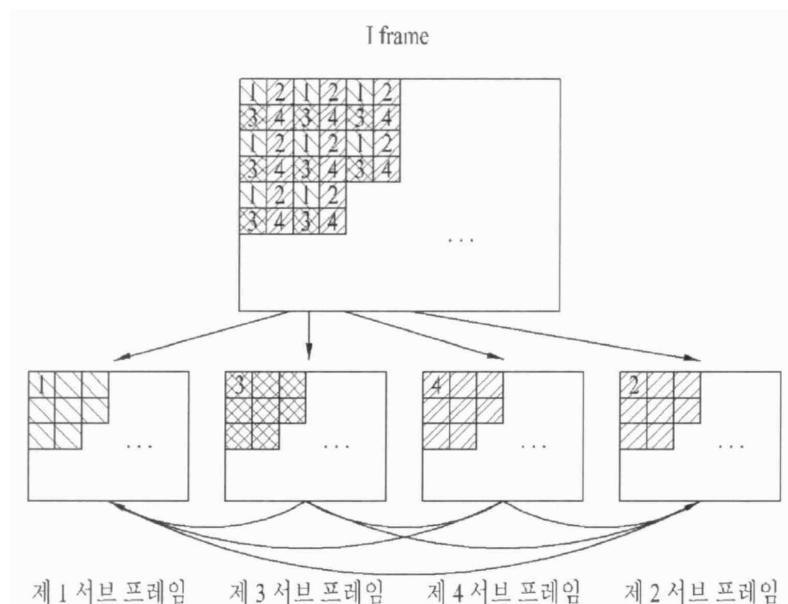


도면11**도면12a**

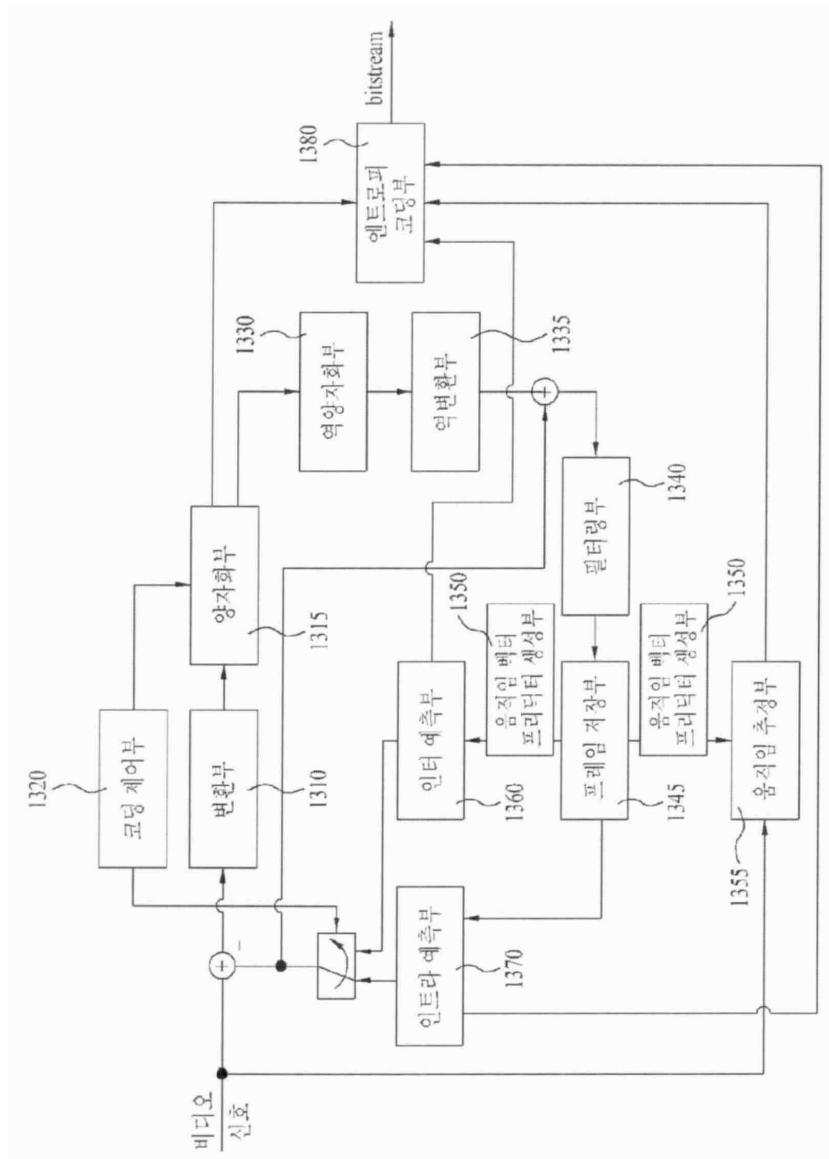
도면12b



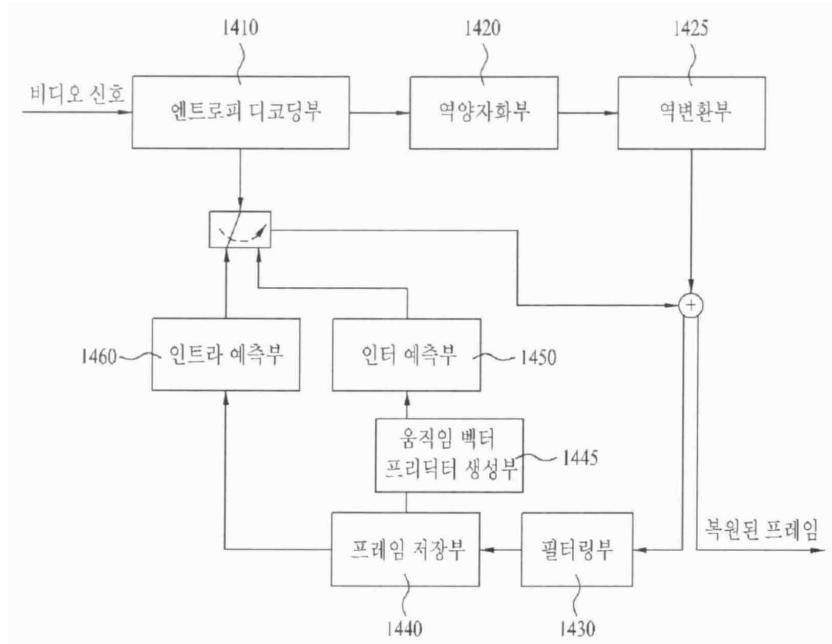
도면12c



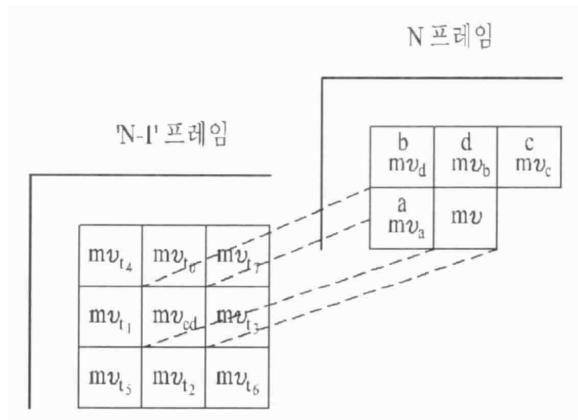
도면13



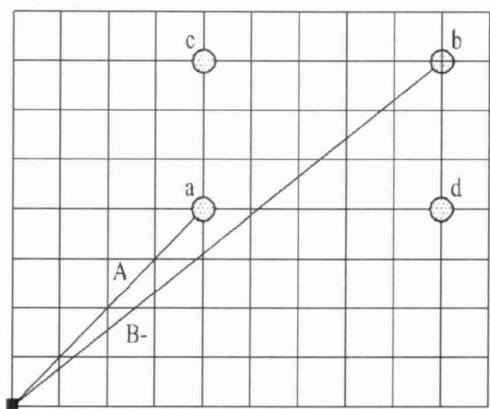
도면14



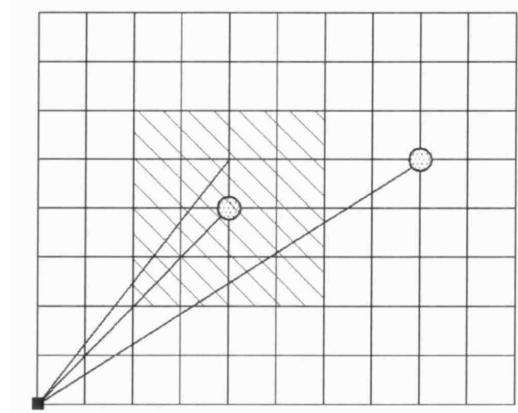
도면15



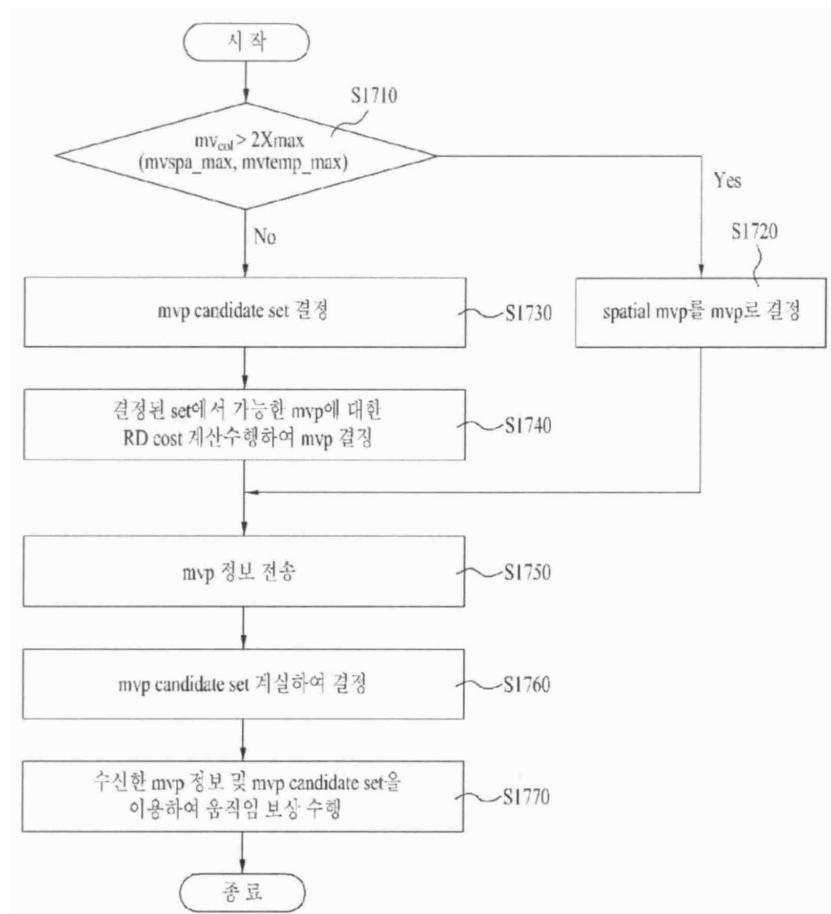
도면16a



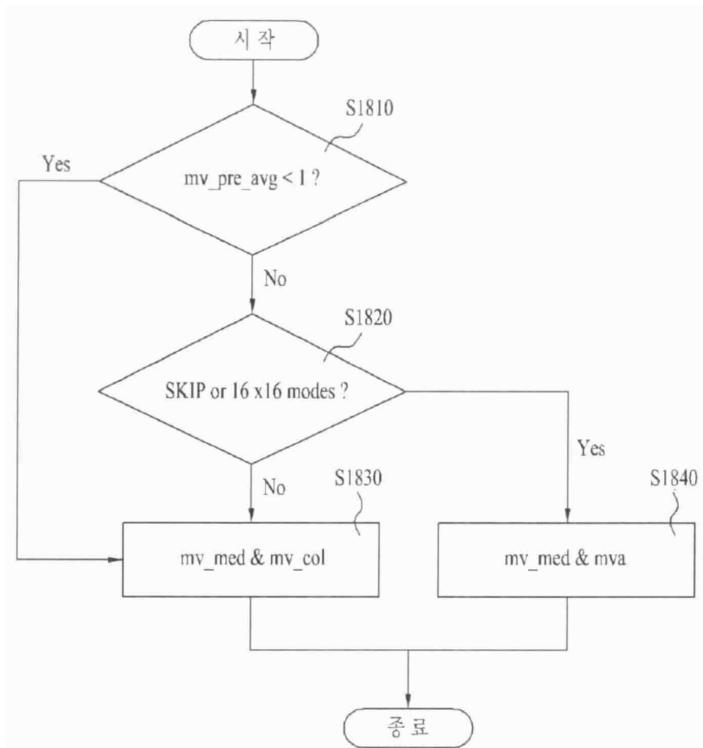
도면16b



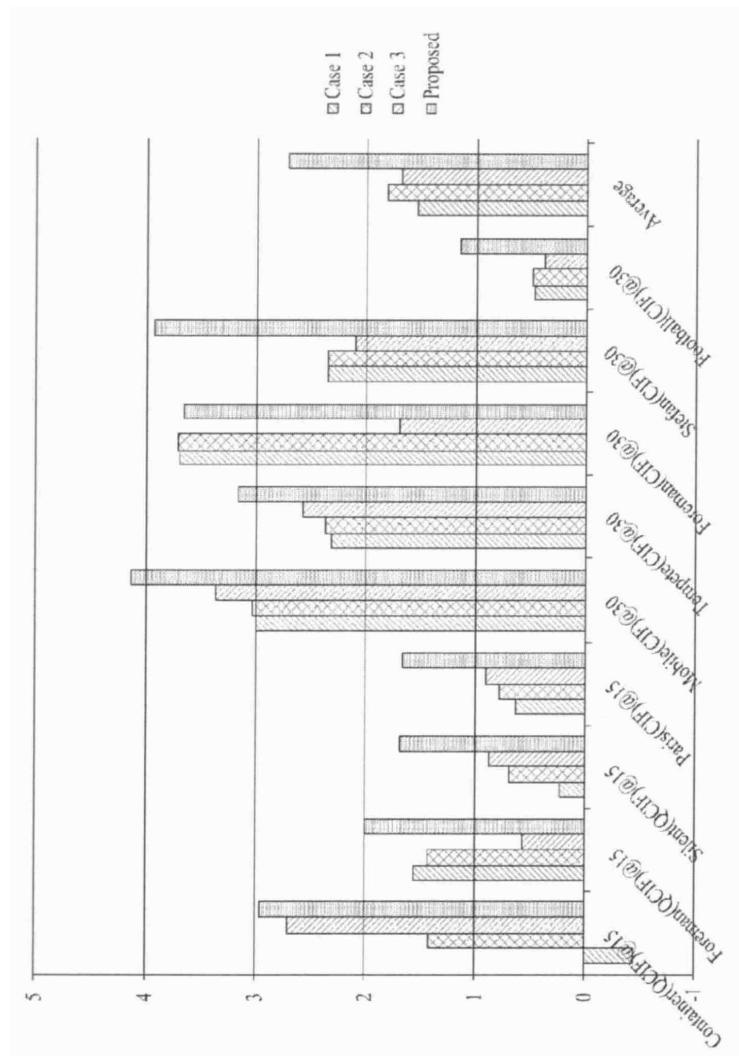
도면17



도면18



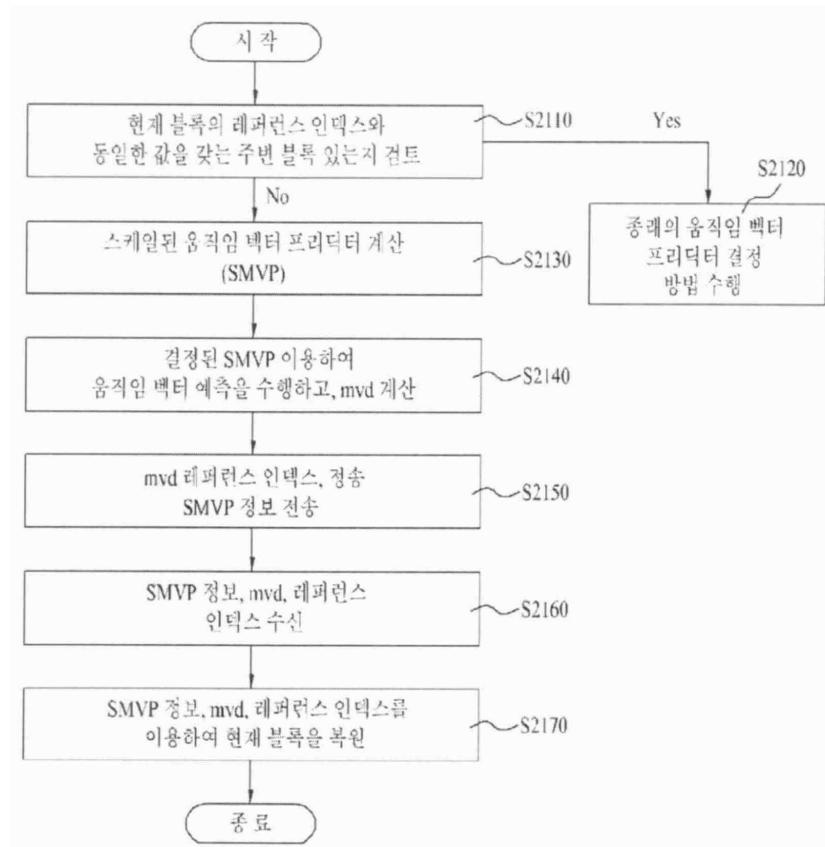
도면19



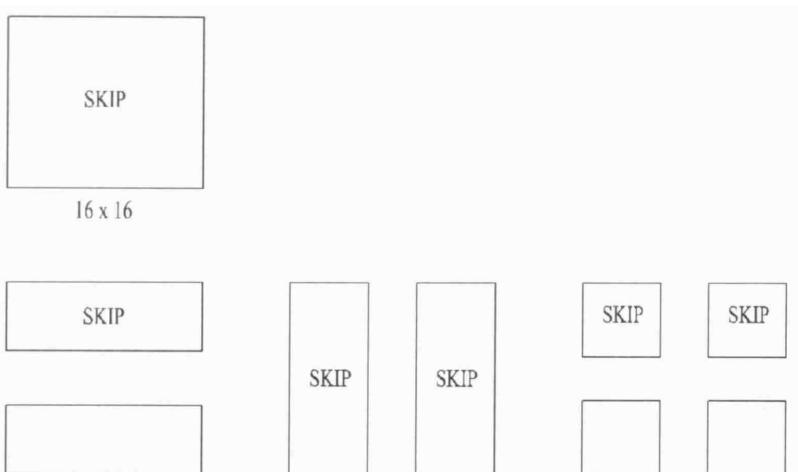
도면20

	mv _v Ref. # 1	mvUR Ref. # 1
mv ₂ Ref. # 0	mv _c	

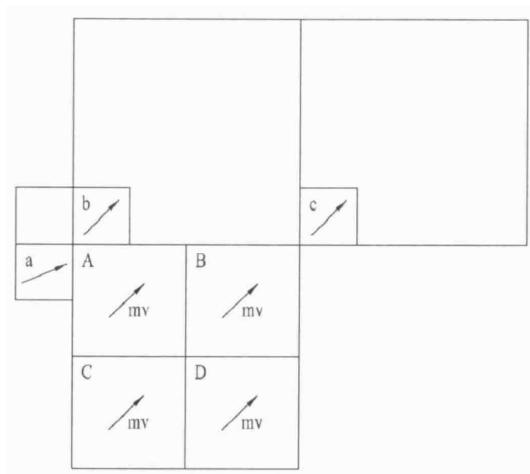
도면21



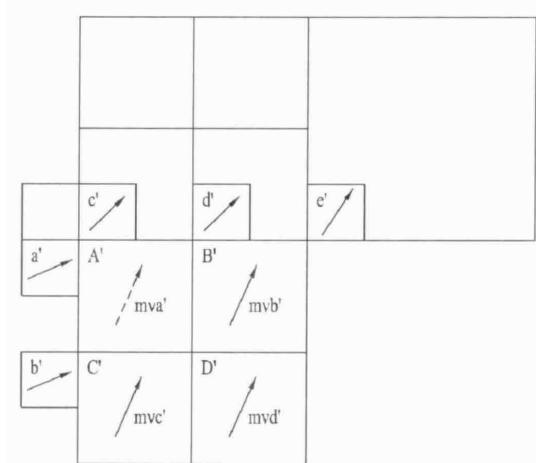
도면22



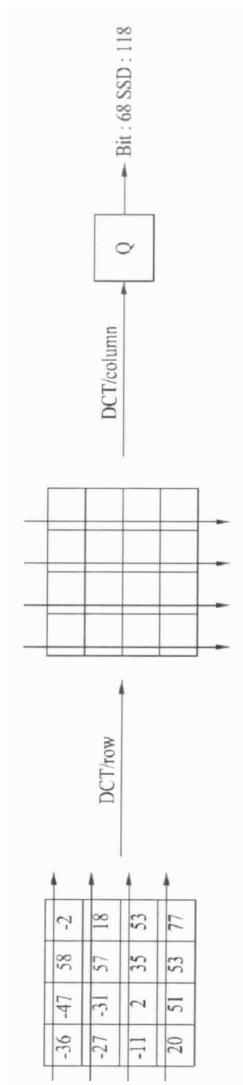
도면23a



도면23b



도면24



도면25

