



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2016년12월30일  
(11) 등록번호 10-1691199  
(24) 등록일자 2016년12월23일

- (51) 국제특허분류(Int. Cl.)  
*HO4N 19/50* (2014.01)
- (21) 출원번호 10-2010-7025282
- (22) 출원일자(국제) 2009년04월07일  
심사청구일자 2014년02월26일
- (85) 번역문제출일자 2010년11월10일
- (65) 공개번호 10-2011-00009141
- (43) 공개일자 2011년01월27일
- (86) 국제출원번호 PCT/US2009/002177
- (87) 국제공개번호 WO 2009/126260  
국제공개일자 2009년10월15일
- (30) 우선권주장  
61/044,193 2008년04월11일 미국(US)
- (56) 선행기술조사문헌  
KR1020050067083 A\*  
WO2007125856 A1\*

\*는 심사관에 의하여 인용된 문헌

전체 청구항 수 : 총 41 항

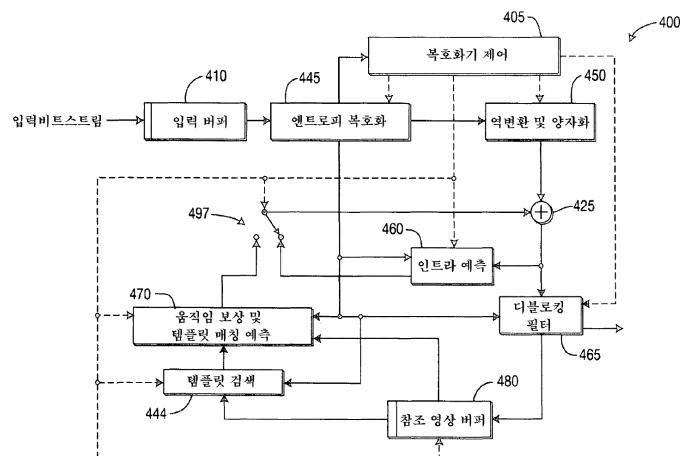
심사관 : 박상철

(54) 발명의 명칭 **비디오 부호화 및 복호화에서의 템플릿 매칭 예측을 위한 방법 및 장치**

**(57) 요약**

비디오 부호화 및 복호화에서의 템플릿 매칭 예측을 위한 방법 및 장치가 제공된다. 장치는 영상의 타겟 블러를 상기 타겟 블러에 대한 적어도 하나의 예측기를 결정하고 상기 타겟 블러에 대한 적어도 하나의 부가의 예측기를 검색하도록 상기 적어도 하나의 예측기를 템플릿으로 각각 이용함으로써 부호화하기 위한 부호화기(300)를 포함한다.

**대 표 도**



(72) 발명자  
인, 평  
미국 뉴저지주 08536 플래인즈버로 소로 드라이브  
49

솔, 조엘  
미국 뉴저지주 08536 플래인즈버로 퀘일 럿지 드라이브  
1006

---

## 명세서

### 청구범위

#### 청구항 1

장치에 있어서:

타겟 블럭에 대한 복수의 예측기를 결정하고 상기 타겟 블럭에 대한 복수의 부가의 예측기를 검색하도록 상기 복수의 예측기를 템플릿 매칭 예측 프로세스에서 템플릿으로 각각 이용함으로써 영상 내의 상기 타겟 블럭을 부호화하고,

상기 타겟 블럭을 예측하기 위해 다수 가설 움직임 보상 예측 프로세스에서 상기 부가의 예측기를 사용하는 부호화기(300)를 포함하되,

상기 복수의 예측기는 참조 영상에서 참조 블럭인 장치.

#### 청구항 2

제1항에 있어서, 상기 복수의 예측기는 인트라 예측기, 인터 예측기, 또는 그 조합인 장치.

#### 청구항 3

제2항에 있어서, 상기 복수의 예측기에 대한 부가 정보는 암시적 부호화 및 명시적 부호화된 것 중 적어도 하나인 장치.

#### 청구항 4

제1항에 있어서, 상기 복수의 부가의 예측기는 템플릿 매칭 예측 프로세스 및 다수 가설(multi-hypothesis) 움직임 보상 예측 프로세스를 이용하여 취득되는 장치.

#### 청구항 5

제4항에 있어서, 명시적 움직임 추정 프로세스는 상기 복수의 예측기를 결정하는 데에 이용되고 상기 템플릿 매칭 예측 프로세스는 다수의 참조 목록과 관련하여 상기 복수의 부가의 예측기를 검색하는 데에 이용되는 장치.

#### 청구항 6

제5항에 있어서, 상기 명시적 움직임 추정 프로세스는 상기 다수의 참조 목록 중 특정의 것으로 제한되고, 상기 템플릿 매칭 예측 프로세스는 상기 다수의 참조 목록 중 다른 것으로 제한되는 장치.

#### 청구항 7

제5항에 있어서, 상기 복수의 예측기는 상기 다수의 참조 목록 중에서 제1 참조 목록과 제2 참조 목록과 관련하여 상기 명시적 움직임 추정 프로세스와 암시적 움직임 추정 프로세스를 각각 적용하여 취득된 제1 예측기 및 제2 예측기를 포함하고, 상기 복수의 부가의 예측기는 상기 제2 참조 목록과 상기 제1 참조 목록에 관련하여 상기 템플릿 매칭 예측 프로세스에서 상기 제1 예측기와 상기 제2 예측기를 각각 이용하여 취득된 제1 부가 예측기 및 제2 부가 예측기를 포함하는 장치.

**청구항 8**

제7항에 있어서, 상기 제1 예측기 및 제2 부가 예측기는 상이한 참조 영상에 놓이는 장치.

**청구항 9**

제5항에 있어서, 상기 명시적 움직임 추정 프로세스는 상기 다수의 참조 목록 중 특정한 것에 대응하는 특정한 하나의 참조 영상에서 이용되고, 상기 템플릿 매칭 예측 프로세스는 상기 특정한 하나의 참조 영상 또는 상기 다수의 참조 목록 중 다른 것에 대응하는 다른 참조 영상에서 이용되는 장치.

**청구항 10**

제4항에 있어서, 적어도 하나의 예측은 움직임 벡터 예측기를 이용하여 유도되거나 추정되는(estimated) 장치.

**청구항 11**

방법에 있어서:

타겟 블럭에 대한 복수의 예측기를 결정하고 상기 타겟 블럭에 대한 복수의 부가의 예측기를 검색하도록 상기 복수의 예측기를 템플릿 매칭 예측 프로세스에서 템플릿으로 각각 이용하여(615, 630, 915, 918, 1115, 1420, 1625) 영상 내의 상기 타겟 블럭을 부호화하는 단계, 및

상기 타겟 블럭을 예측하기 위해 다수 가설 움직임 보상 예측 프로세스에서 상기 부가의 예측기를 사용하는 단계를 포함하되,

상기 복수의 예측기는 참조 영상에서 참조 블럭인 방법.

**청구항 12**

제11항에 있어서, 상기 복수의 예측기는 인트라 예측기(1620, 1680), 인터 예측기(612, 627, 909, 912, 1109, 1112, 1415), 또는 그 조합(1115)인 방법.

**청구항 13**

제12항에 있어서, 상기 복수의 예측기에 대한 부가 정보는 암시적 부호화 및 명시적 부호화된 것 중 적어도 하나(651, 948, 1139, 1455, 1660)인 방법.

**청구항 14**

제11항에 있어서, 상기 복수의 부가의 예측기는 템플릿 매칭 예측 프로세스 및 다수 가설 움직임 보상 예측 프로세스를 이용하여 취득되는(618, 633, 921, 1118, 1425, 1630) 방법.

**청구항 15**

제14항에 있어서, 명시적 움직임 추정 프로세스(612, 627, 909, 912, 1109, 1112, 1415)는 상기 복수의 예측기를 결정하는 데에 이용되고 상기 템플릿 매칭 예측 프로세스(615, 630, 915, 918, 1115, 1420, 1625)는 다수의 참조 목록과 관련하여 상기 복수의 부가의 예측기를 검색하는 데에 이용되는 방법.

**청구항 16**

제15항에 있어서, 상기 명시적 움직임 추정 프로세스(612, 627, 909, 912, 1109, 1112, 1415)는 상기 다수의 참조 목록 중 특정의 것으로 제한되고, 상기 템플릿 매칭 예측 프로세스(615, 630, 915, 918, 1115, 1625)는 상기 다수의 참조 목록 중 다른 것으로 제한되는 방법.

**청구항 17**

제15항에 있어서, 상기 복수의 예측기는 상기 다수의 참조 목록 중에서 제1 참조 목록과 제2 참조 목록과 관련하여 상기 명시적 움직임 추정 프로세스(612, 627, 909, 912, 1109, 1112, 1415) 및 암시적 움직임 추정 프로세스(615, 630, 915, 918, 1115, 1625)를 각각 적용하여 취득된 제1 예측기 및 제2 예측기를 포함하고, 상기 복수의 부가의 예측기는 상기 제2 참조 목록과 상기 제1 참조 목록에 관련하여 상기 템플릿 매칭 예측 프로세스(615, 630, 915, 918, 1115, 1625)에서 상기 제1 예측기와 상기 제2 예측기를 각각 이용하여 취득된 제1 부가 예측기 및 제2 부가 예측기를 포함하는 방법.

**청구항 18**

제17항에 있어서, 상기 제1 예측기 및 제2 부가 예측기는 다른 참조 영상에 놓이는(1420) 방법.

**청구항 19**

제15항에 있어서, 상기 명시적 움직임 추정 프로세스(612, 627, 909, 912, 1109, 1112, 1415)는 상기 다수의 참조 목록 중 특정한 것에 대응하는 특정한 하나의 참조 영상에서 이용되고, 상기 템플릿 매칭 예측 프로세스(615, 630, 915, 918, 1115, 1420, 1625)는 상기 특정한 하나의 참조 영상 또는 상기 다수의 참조 목록 중 다른 것에 대응하는 다른 참조 영상에서 이용되는 방법.

**청구항 20**

제14항에 있어서, 적어도 하나의 예측은 움직임 벡터 예측기를 이용하여 유도되거나 추정되는 방법.

**청구항 21**

장치에 있어서,

타겟 블럭에 대한 복수의 예측기를 결정하고 상기 타겟 블럭에 대한 복수의 부가의 예측기를 검색하도록 상기 복수의 예측기를 템플릿 매칭 예측 프로세스에서 템플릿으로 각각 이용하여 영상 내의 상기 타겟 블럭을 복호화하고,

상기 타겟 블럭을 예측하기 위해 다수 가설 움직임 보상 예측 프로세스에서 상기 부가의 예측기를 사용하는 복호화기(400)를 포함하되,

상기 복수의 예측기는 참조 영상에서 참조 블럭인 장치.

**청구항 22**

제21항에 있어서, 상기 복수의 예측기는 인트라 예측기, 인터 예측기, 또는 그 조합인 장치.

**청구항 23**

제22항에 있어서, 상기 복수의 예측기에 대한 부가 정보는 유도된 것 및 명시적 부호화된 것 중 적어도 하나인 장치.

**청구항 24**

제21항에 있어서, 상기 복수의 부가의 예측기는 템플릿 매칭 예측 프로세스 및 다수 가설 움직임 보상 예측 프로세스를 이용하여 취득되는 장치.

**청구항 25**

제24항에 있어서, 명시적 움직임 추정 프로세스는 상기 복수의 예측기를 결정하는 데에 이용되고 상기 템플릿 매칭 예측 프로세스는 다수의 참조 목록과 관련하여 상기 복수의 부가의 예측기를 검색하는 데에 이용되는 장치.

**청구항 26**

제25항에 있어서, 상기 명시적 움직임 추정 프로세스는 상기 다수의 참조 목록 중 특정의 것으로 제한되고, 상기 템플릿 매칭 예측 프로세스는 상기 다수의 참조 목록 중 다른 것으로 제한되는 장치.

**청구항 27**

제25항에 있어서, 상기 복수의 예측기는 상기 다수의 참조 목록 중에서 제1 참조 목록과 제2 참조 목록과 관련하여 상기 명시적 움직임 추정 프로세스와 암시적 움직임 추정 프로세스를 각각 적용하여 취득된 제1 예측기 및 제2 예측기를 포함하고, 상기 복수의 부가의 예측기는 상기 제2 참조 목록과 상기 제1 참조 목록에 관련하여 상기 템플릿 매칭 예측 프로세스에서 상기 제1 예측기와 상기 제2 예측기를 각각 이용하여 취득된 제1 부가 예측기 및 제2 부가 예측기를 포함하는 장치.

**청구항 28**

제27항에 있어서, 상기 제1 예측기 및 제2 부가 예측기는 다른 참조 영상에 놓이는 장치.

**청구항 29**

제25항에 있어서, 상기 명시적 움직임 추정 프로세스는 상기 다수의 참조 목록 중 특정한 것에 대응하는 특정한 하나의 참조 영상에서 이용되고, 상기 템플릿 매칭 예측 프로세스는 상기 특정한 하나의 참조 영상 또는 상기 다수의 참조 목록 중 다른 것에 대응하는 다른 참조 영상에서 이용되는 장치.

**청구항 30**

제24항에 있어서, 적어도 하나의 예측은 움직임 벡터 예측기를 이용하여 유도되거나 추정되는 장치.

**청구항 31**

방법에 있어서:

타겟 블럭에 대한 복수의 예측기를 결정하고 상기 타겟 블럭에 대한 복수의 부가의 예측기를 검색하도록 상기 복수의 예측기를 템플릿 매칭 예측 프로세스에서 템플릿으로서 각각 이용하여(724, 736, 1021, 1027, 1235, 1535, 1735) 영상 내의 타겟 블럭을 부호화하는 단계, 및

상기 타겟 블럭을 예측하기 위해 다수 가설 움직임 보상 예측 프로세스에서 상기 부가의 예측기를 사용하는 단계를 포함하되,

상기 복수의 예측기는 참조 영상에서 참조 블럭인 방법.

### 청구항 32

제31항에 있어서, 상기 복수의 예측기는 인트라 예측기(1725, 1750), 인터 예측기(718, 730, 1012, 1220, 1525), 또는 그 조합인 방법.

### 청구항 33

제32항에 있어서, 상기 복수의 예측기에 대한 부가 정보는 유도된 것 및 명시적 부호화된 것 중 적어도 하나(709, 712, 721, 733, 1009, 1015, 1215, 1225, 1515, 1520, 1715, 1730)인 방법.

### 청구항 34

제31항에 있어서, 상기 복수의 부가의 예측기는 템플릿 매칭 예측 프로세스(724, 736, 1021, 1027, 1235, 1535, 1735) 및 다수 가설 움직임 보상 예측 프로세스를 이용하여 취득되는(727, 739, 1018, 1024, 1030, 1039, 1230, 1240, 1540, 1740) 방법.

### 청구항 35

제34항에 있어서, 명시적 움직임 추정 프로세스(718, 730, 1012, 1220, 1525)는 상기 복수의 예측기를 결정하는 데에 이용되고 상기 템플릿 매칭 예측 프로세스(724, 736, 1021, 1027, 1235, 1735)는 다수의 참조 목록과 관련하여 상기 복수의 부가의 예측기를 검색하는 데에 이용되는 방법.

### 청구항 36

제35항에 있어서, 상기 명시적 움직임 추정 프로세스(718, 730, 1012, 1220, 1525)는 상기 다수의 참조 목록 중 특정의 것으로 제한되고, 상기 템플릿 매칭 예측 프로세스(724, 736, 1021, 1027, 1235, 1735)는 상기 다수의 참조 목록 중 다른 것으로 제한되는 방법.

### 청구항 37

제35항에 있어서, 상기 복수의 예측기는 상기 다수의 참조 목록 중에서 제1 참조 목록과 제2 참조 목록과 관련하여 상기 명시적 움직임 추정 프로세스(718, 730, 1012, 1220, 1525) 및 암시적 움직임 추정 프로세스(724, 736, 1021, 1027, 1235, 1735)를 각각 적용하여 취득된 제1 예측기 및 제2 예측기를 포함하고, 상기 복수의 부가의 예측기는 상기 제2 참조 목록과 상기 제1 참조 목록에 관련하여 상기 템플릿 매칭 예측 프로세스(724, 736, 1021, 1027, 1235, 1735)에서 상기 제1 예측기와 상기 제2 예측기를 각각 이용하여 취득된 제1 부가 예측기 및 제2 부가 예측기를 포함하는 방법.

### 청구항 38

제37항에 있어서, 상기 제1 예측기 및 제2 부가 예측기는 다른 참조 영상에 놓이는(1535) 방법.

### 청구항 39

제35항에 있어서, 상기 명시적 움직임 추정 프로세스(718, 730, 1012, 1220, 1525)는 상기 다수의 참조 목록 중 특정한 것에 대응하는 특정한 하나의 참조 영상에서 이용되고, 상기 템플릿 매칭 예측 프로세스(724, 736, 1021, 1027, 1235, 1535, 1735)는 상기 특정한 하나의 참조 영상 또는 상기 다수의 참조 목록 중 다른 것에 대응하는 다른 참조 영상에서 이용되는 방법.

### 청구항 40

제34항에 있어서, 적어도 하나의 예측은 움직임 벡터 예측기를 이용하여 유도되거나 추정되는 방법.

### 청구항 41

부호화된 비디오 신호 데이터를 저장하고 있는 저장 매체에 있어서:

타겟 블럭에 대한 복수의 예측기를 결정하고 상기 타겟 블럭에 대한 복수의 부가의 예측기를 검색하도록 상기 복수의 예측기를 템플릿 매칭 예측 프로세스에서 템플릿으로 각각 이용함으로써 부호화된 영상 내의 타겟 블럭을 포함하고,

상기 타겟 블럭을 예측하기 위해 다수 가설 움직임 보상 예측 프로세스에서 상기 부가의 예측기를 사용하되, 상기 복수의 예측기는 참조 영상에서 참조 블럭인 저장 매체.

## 발명의 설명

### 기술 분야

[0001] 이 출원은 여기에서 전체적으로 참조로 결합되는 미국 예비 출원 No.61/044,193 (대리인 일람 번호 PU080048)의 이점을 청구한다.

[0002] 본 발명은 일반적으로 비디오 부호화 및 복호화에 관한 것으로, 더욱 특히 비디오 부호화 및 복호화에서의 템플릿 매칭 예측(template matching prediction; TMP)을 위한 방법 및 장치에 관한 것이다.

### 배경 기술

[0003] 비디오 부호화 및 복호화 기구에 이용되는, 템플릿 매칭 예측(TMP)은 움직임 벡터/변위 벡터의 전송을 방지함으로써 인터 예측 및/또는 인트라 예측을 위한 부호화 효율의 이득을 이룰 수 있다. 그러나, 템플릿 매칭 예측의 예측 성능은 타겟 블럭과 그 대응하는 템플릿 간의 상관 관계에 매우 의존한다. 도 1에서 나타낸 바와 같이, 상부에 및/또는 좌측으로, 타겟 블럭(target block)을 둘러싸는 재구성된 화소를 이용하는 것이 본 기술에서 잘 알려져 있다. 도 1을 참조하면, 템플릿 매칭 예측을 수반하는 암시적 움직임 추정 기구(implicit motion estimation scheme)는 보통 참조 부호 100으로 나타내었다. 암시적 움직임 추정 기구(100)는 검색 영역(111), 검색 영역(111) 내의 예측(112), 및 예측(112)과 관련한 인접부(113)를 갖는 재구성된 참조 프레임(reference frame; 110)을 수반한다. 암시적 움직임 추정 기구(100)는 또한 타겟 블럭(151), 타겟 블럭(151)과 관련한 템플릿(152), 및 재구성된 영역(153)을 갖는 현재 프레임(150)을 수반한다. 그러나, 이런 방법으로 템플릿 매칭 예측을 이용하게 되면 특히 템플릿이 타겟 신호와의 상관 관계가 낮을 때 부호화기 및 복호화기 성능을 떨어뜨리게 된다.

### 인터 예측

[0005] 비디오 부호화시, 인터 예측은 타겟 프레임과 기준 프레임 간의 시간적 중복성(temporal redundancy)을 감소하기 위해 광범위하게 이용된다. 움직임 추정과 보상이 인터 예측을 위한 성분이다. 일반적으로, 우리는 움직임

모델 및 대응하는 움직임 추정 기법을 두 카테고리로 분류할 수 있다. 제1 카테고리는 소위 명시적 움직임 추정이거나, 명시적 움직임 모델이며, 이 때 움직임 벡터는 명시적으로 전송된다. 제2 카테고리는 소위 암시적 움직임 추정과, 여기에서 움직임 정보는 재구성된 화소에 기초하여 추정(estimated)되므로, 어떠한 움직임 벡터도 전송될 필요가 없다.

[0006] 도 2에 나타낸 바와 같이, 블럭 매칭 기법에 기초한 제1 카테고리는 예를 들어, 국제 표준화 기구/국제 전기기계 위원회 (International Organization for Standardization/International Electrotechnical Commission; ISO/IEC) 동화상 전문가 그룹-4(MPEG-4) 파트 10 어드밴스드 비디오 코딩 (AVC) 표준/국제 텔레커뮤니케이션 협회, 텔레커뮤니케이션 섹터 (ITU-T) H.264 권장 (이하 "MPEG-4 AVC 표준")과 같은 현재 비디오 부호화 표준에서 잘 적용된다. 도 2를 참조하면, 블럭 매칭을 포함하는 명시적 움직임 추정 기구의 일 예는 일반적으로 참조 부호 200으로 나타내었다. 명시적 움직임 추정 기구(200)는 검색 영역(201) 및 검색 영역(201) 내의 예측부(202)를 갖는 재구성된 기준 프레임(210)을 수반한다. 명시적 움직임 추정 기구(200)는 또한 타겟 블럭(251) 및 재구성된 영역(252)을 갖는 현재 프레임(250)을 수반한다. 움직임 벡터 MV는 타겟 블럭(251)과 예측(202) 간의 움직임을 나타내는 데에 이용된다. 제1 카테고리는 보통 두 단계에서 실행된다. 타겟(현재) 블럭과 기준 프레임 간의 움직임 벡터가 추정된다. 다음에 움직임 정보가 부호화되어 명시적으로 복호화기에 보내진다. 복호화 기에서, 움직임 정보가 복호화되어 이전에 복호화된 재구성된 기준 프레임으로부터 타겟 블럭을 예측하는 데에 이용된다.

[0007] 제2 카테고리는 비트스트림으로 명시적으로 움직임 정보를 부호화하지 않는 움직임 추정 방법의 클래스를 말한다. 대신에, 부호화기에서 실행된 것과 동일한 움직임 정보 변이가 복호화기에서 실행된다. 하나의 실제적인 암시적 움직임 추정 기구는 공간-시간 자동회귀 모델을 이용하는 것이며, 이 때 최소 제곱 예측법 (least-square prediction; LSP)이 적용된다. 다른 공통의 대중적인 방법은 도 1에 나타낸 템플릿 매칭 예측법과 같은 패치(patch) 기반의 모델을 이용하는 것이다.

[0008] 일반적으로, 명시적 움직임 추정은 특히 고속의 부자연스런 움직임에 대해서, 하지만 움직임 벡터에 대한 부호화 비트를 희생하고, 더 정확해질 수 있다. 암시적 움직임 추정은 저속의 자연스런 움직임에 대해 더 적합하지만, 이것은 부호화기가 행한 것과 같이 움직임 정보를 연산해야 하기 때문에 복호화기에 복잡성의 부담을 준다.

#### MPEG-4 AVC 표준 인터 예측

[0010] MPEG-4 AVC 표준은 트리 구조의 계층적 매크로블럭 구획(macroblock partition)을 이용한다. 인터 부호화된 16x16 화소 매크로블럭은 크기 16x8, 8x16 또는 8x8의 매크로블럭 구획으로 나누어진다. 8x8 화소의 매크로블럭 구획은 또한 서브매크로블럭으로 알려져 있다. 서브매크로블럭은 또한 크기 8x4, 4x8 및 4x4의 서브매크로블럭 구획으로 나누어진다. 부호화기는 압축 효율과 본질적인 품질을 최대화하기 위해서, 특정 매크로블럭의 특성에 기초하여 매크로블럭을 구획 및 서브매크로 구획으로 분할하는 방법을 선택할 수 있다.

[0011] 다수의 참조 영상(multiple reference pictures)은 인터 예측에 이용될 수 있으며, 참조 영상 인덱스는 다수의 참조 영상 중에서 어느 것이 이용되는지를 나타내기 위해 부호화된다. P 영상 (또는 P 슬라이스)에 대해, 오직 단일 방향의 예측만이 이용되고, 허용 가능한 참조 영상이 목록 0에서 관리된다. B 영상 (또는 B 슬라이스)에 대해, 두 목록의 참조 영상들이 목록 0 및 목록 1에서 관리된다. B 영상 (또는 B 슬라이스)에 대해, 목록 0 또는 목록 1을 이용한 단일 방향 예측이 허용되거나, 두 목록 0 및 목록 1을 이용한 양방향 예측이 허용된다. 양방향이 이용되면, 목록 0 및 목록 1 예측기는 최종 예측기를 형성하기 위해 함께 평균화된다.

[0012] 각 매크로블럭 구획은 독립된 참조 영상 지수, 예측 유형 (목록 0, 목록 1, bipred) 및 독립된 움직임 벡터를 가질 수 있다. 각 서브매크로블럭 구획은 독립된 움직임 벡터를 가지지만, 동일한 서브매크로블럭의 모든 서브매크로블럭 구획은 동일한 참조 영상 지수와 예측 유형을 이용한다.

[0013] MPEG-4 AVC 표준 조인트 모델 (JM) 참조 소프트웨어에서, 비트율 왜곡 최적화(Rate Distortion Optimization RDO) 프레임워크가 모드 판정에 이용된다. 인터 모드에 대해, 움직임 추정은 모드 판정과는 별개로 고려된다. 움직임 추정은 인터 모드의 모든 블럭 유형에 대해 먼저 실행된 다음에, 모드 판정은 각 인터 모드와 인트라 모드의 코스트를 비교하여 행해진다. 최소한의 코스트의 모드가 최상의 모드로 선택된다. 다음 모드는 P 프레임과 B 프레임에 각각 관련된다.

$$[0014] \quad MODE \in \left\{ \begin{array}{l} INTRA4x4, INTRA16x16, SKIP, \\ 16x16, 16x8, 8x16, 8x8, 8x4, 4x8, 4x4 \end{array} \right\}, \\ P \text{ 프레임:}$$

$$MODE \in \left\{ \begin{array}{l} INTRA4x4, INTRA16x16, DIRECT, \\ 16x16L0, 16x8L0, 8x16L0, 8x8L0, 8x4L0, \\ 4x8L0, 4x4L0, 16x16L1, 16x8L1, 8x16L1, \\ 8x8L1, 8x4L1, 4x8L1, 4x4L1, 16x16BI, 16x8BI, 8x16BI, \\ 8x8BI, 8x4BI, 4x8BI, 4x4BI \end{array} \right\}$$

[0015] B 프레임:

### 0016] 템플릿 매칭

[0017] 템플릿 매칭 예측은 움직임 벡터를 보내지 않고 타겟 화소를 예측하는 한 방법이다. 프레임의 타겟 블럭이 주어지면, 블럭 내의 타겟 화소는 한 세트의 참조 샘플로부터 최적의 화소를 찾아 결정되고, 이 때 최적 화소의 인접 화소는 타겟 화소의 것과 최고의 상관 관계를 갖는다. 타겟 화소의 이들 인접 화소는 템플릿으로 불린다. 종래 기술은 도 1에서 나타낸 바와 같이, 템플릿은 항상 타겟 화소의 재구성된 주변 화소로부터 취한다는 것을 보여준다. 템플릿 매칭 예측은 인트라 및 인터 예측 둘 다에 적용된다.

[0018] 인터 예측의 경우, 템플릿 매칭 프로세스는 복호화기 측에서 움직임 벡터 검색으로 볼 수 있다. 여기에서, 템플릿 매칭은 종래의 움직임 추정 기법과 매우 유사하게 실행된다. 움직임 벡터는 기준 프레임에서 상응하여 변위된 템플릿 형상의 영역에 대한 코스트 함수를 연산하여 추정된다. 템플릿의 최상의 움직임 벡터는 타겟 영역을 예측하는 데에 이용된다. 재구성 또는 적어도 예측 신호가 이미 존재하는 화상의 영역들만이 검색을 위해 액세스된다. 따라서, 복호화기는 템플릿 매칭 프로세스를 실행할 수 있으며 부가 정보 없이 타겟 영역을 예측 할 수 있다.

[0019] 템플릿 매칭은 움직임 벡터의 전송 없이 타겟 블럭 내의 화소를 예측할 수 있다. 템플릿 매칭 예측의 예측 성능은 타겟 블럭과 그 템플릿 간의 상관 관계가 높은 경우 블럭 매칭 기구의 것에 필적할 만하다고 예측된다. 종래 기술에서, 템플릿은 항상 타겟 화소의 재구성된 공간 인접 화소로부터 취해진다. 인접 화소는 때로 타겟 화소와의 상관 관계가 적으므로, 템플릿 매칭 예측의 성능은 종래의 블럭 매칭 기구 보다 더 낮을 수 있다.

### [0020] 다수 가설 움직임 보상 예측(Motion Compensated Prediction (MCP))

[0021] 다수 가설은 타겟 블럭의 예측 신호가 덜 노이지하게 만들거나, 타겟 블럭과 더욱 상관되게 만들기 위해 다수의 예측 후보들이 조합된 것이다. 양방향 예측은 다수의 가설의 가장 일반적인 경우이다. 종래 기술에서, 다수 가설 움직임 보상된 예측은 명시된 예측과 암시된 예측과 함께, 두 카테고리로 분류될 수 있다. 제1 카테고리에서, 모든 움직임 벡터는 각 가설 움직임 보상된 예측을 위해 명시적으로 전송된다. 제2 카테고리에서는, 어떤 움직임 벡터도 템플릿 매칭 평균화와 같이, 비트스트림으로 구현되지 않는다. 제1 방법의 단점은 움직임에 대한 비트율 오버헤드(bitrate overhead)이다. 제2 카테고리의 단점은 매칭 프로세스가 인접 화소에 근거하기 때문에 가능하게 수반되는 아웃라이어(outlier)에 의해 생기는 부정확성이다.

## 발명의 내용

### 해결하려는 과제

[0022] 종래 기술의 이들 및 그 외 결점과 단점들은 비디오 부호화 및 복호화에서의 템플릿 매칭 예측(TMP)을 위한 방법 및 장치에 관한 본 원리에 의해 해결된다.

### 과제의 해결 수단

[0023] 본 원리의 형태에 따르면, 장치가 제공된다. 이 장치는 영상의 타겟 블럭을 타겟 블럭에 대한 적어도 하나의 예측기를 결정하고 타겟 블럭에 대한 적어도 하나의 부가의 예측기를 검색하도록 적어도 하나의 예측기를 템플릿으로 각각 이용하여 부호화하기 위한 부호화기를 포함한다.

[0024] 본 원리의 다른 형태에 따르면, 방법이 제공된다. 이 방법은 영상의 타겟 블럭을, 타겟 블럭에 대한 적어도 하나의 예측기를 결정하고 타겟 블럭에 대한 적어도 하나의 부가의 예측기를 검색하도록 적어도 하나의 예측기를 템플릿으로 각각 이용하여 부호화하는 단계를 포함한다.

[0025] 본 원리의 또 다른 형태에 따르면, 장치가 제공된다. 이 장치는 영상의 타겟 블럭을 타겟 블럭에 대한 적어도 하나의 예측기를 결정하고 타겟 블럭에 대한 적어도 하나의 부가의 예측기를 검색하도록 적어도 하나의 예측기를 템플릿으로 각각 이용하여 복호화하기 위한 복호화기를 포함하는 장치.

[0026] 본 원리의 다른 형태에 따르면, 방법이 제공된다. 이 방법은 영상의 타겟 블럭을 타겟 블럭에 대한 적어도 하나의 예측기를 결정하고 타겟 블럭에 대한 적어도 하나의 부가의 예측기를 검색하도록 적어도 하나의 예측기를 템플릿으로 각각 이용하여 복호화하는 단계를 포함한다.

[0027] 본 원리의 이들 및 그 외 형태, 특성 및 장점들은 첨부한 도면과 관련하여 쉽게 되는 다음 예시의 실시예의 상세한 설명으로부터 명백하게 될 것이다.

### 도면의 간단한 설명

[0028] 본 원리는 다음의 예시의 도면에 따라 잘 이해될 것이다.

도 1은 템플릿 매칭 예측을 수반하는 암시적 움직임 추정 기구의 일 예를 나타내는 도면이다.

도 2는 블럭 매칭을 수반하는 명시적 움직임 추정 기구의 일 예를 나타내는 도면이다.

도 3은 본 발명의 실시예에 따라 본 원리가 적용되는 예시의 비디오 부호화기를 나타내는 블럭도이다.

도 4는 본 원리의 실시예에 따라 본 원리가 적용되는 예시의 비디오 복호화기를 나타내는 블럭도이다.

도 5는 본 원리의 실시예에 따라 단일 예측 (단방향 예측)과 조합된 템플릿 매칭 예측(TMP) 기구의 예를 나타내는 도면이다.

도 6은 본 원리의 실시예에 따라, 단일 참조 목록 예측 (단방향 예측)으로 템플릿 매칭 예측을 이용하는 비디오 부호화 방법을 나타내는 흐름도이다.

도 7은 본 발명의 실시예에 따라, 단일 참조 목록 예측 (단방향 예측)으로 템플릿 매칭 예측을 이용한 비디오 부호화 방법을 나타내는 흐름도이다.

도 8은 본 원리의 실시예에 따라, 양방향 예측과 조합된 템플릿 매칭 예측의 일 예를 나타내는 도면이다.

도 9는 본 원리의 실시예에 따라, 양방향 예측으로 템플릿 매칭 예측을 이용한 비디오 부호화 방법을 나타내는 흐름도이다.

도 10은 본 원리의 실시예에 따라, 양방향 예측으로 템플릿 매칭 예측을 이용한 비디오 복호화 방법을 나타내는 흐름도이다.

도 11은 본 원리의 실시예에 따라, 양방향 예측으로 템플릿 매칭 예측을 이용한 다른 비디오 부호화 방법을 나타내는 흐름도이다.

도 12는 본 원리의 실시예에 따라, 양방향 예측으로 템플릿 매칭 예측을 이용한 다른 비디오 복호화 방법을 나타내는 흐름도이다.

도 13은 본 원리의 실시예에 따라, 다수의 참조 영상을 갖는 템플릿 매칭 예측의 예를 나타내는 흐름도이다.

도 14는 본 원리의 실시예에 따라, 다수의 참조 영상을 갖는 템플릿 매칭 예측을 이용하는 비디오 부호화 방법을 나타내는 흐름도이다.

도 15는 본 원리의 실시예에 따라, 다수의 참조 영상을 갖는 템플릿 매칭 예측을 이용한 비디오 복호화 방법을 나타내는 흐름도이다.

도 16은 본 원리의 실시예에 따라, 인트라 예측을 위한 템플릿 매칭 예측을 이용한 다른 비디오 부호화 방법을 나타내는 흐름도이다.

도 17은 본 원리의 실시예에 따라, 다수의 참조 영상을 갖는 템플릿 매칭 예측을 이용한 다른 비디오 복호화 방법을 나타내는 흐름도이다.

### 발명을 실시하기 위한 구체적인 내용

[0029] 본 원리는 비디오 부호화 및 복호화에서의 템플릿 매칭 예측(TMP)을 위한 방법 및 장치에 관한 것이다.

[0030] 본 상세한 설명은 본 원리를 설명한다. 따라서, 당업자라면 여기에서 명시적으로 기재하거나 나타내지 않았어도, 본 원리를 구현하고 그 정신과 영역 내에 포함되는 여러 가지 구성을 고안할 수 있으리라는 것이 이해될 것이다.

- [0031] 여기 기재된 모든 예들 및 조건부 언어들은 독자가 본 원리를 이해하는 데에 도움을 주기 위한 것이며, 이렇게 특정하여 기재된 예와 조건들에 제한하지 않는 것으로 해야 한다.
- [0032] 더구나, 본 원리의 원리, 형태 및 실시예를 여기에 기재한 모든 문구들 뿐만 아니라, 그 특정 예시들은 그 구조적 및 기능적 균등물을 포괄하고자 하는 것이다. 부가하여, 이러한 균등물은 현재 알려진 균등물 뿐만 아니라 장차 개발될 균등물, 즉 구조와는 상관 없이 동일한 기능을 실행하는 개발 요소들 둘 다를 포함하는 것이다.
- [0033] 따라서, 예를 들어, 당업자라면 여기에서 제시된 블럭도가 본 원리를 구체화하는 설명적 회로의 개념도를 나타낸다는 것을 이해할 것이다. 유사하게, 임의의 플로챠트, 흐름도, 상태 전이도, 의사코드 등이 컴퓨터 판독 가능 매체에서 실질적으로 표현되어 컴퓨터나 프로세서에 의해 실행될 수 있는 다양한 프로세스를 나타낸다는 것이 이해될 것이며, 이런 컴퓨터나 프로세서는 명백하게 나타나 있던지 아니던지 상관 없다.
- [0034] 도면에서 나타낸 여러가지 요소들의 기능은 전용 하드웨어 뿐만 아니라 적당한 소프트웨어와 연계하여 소프트웨어를 실행할 수 있는 하드웨어를 이용하여 제공될 수 있다. 프로세서에 의해 제공될 때, 이 기능들은 하나의 전용 프로세서에 의해, 단일의 공용 프로세서에 의해, 또는 복수의 개별 프로세서에 의해 제공될 수 있으며, 이들 중 몇은 공유될 수 있다. 더구나, 용어 "프로세서" 또는 "제어기"의 명백한 이용은 소프트웨어를 실행할 수 있는 하드웨어를 배타적으로 언급하는 것으로 생각되어서는 안되며, 제한 없이, 디지털 신호 프로세서("DSP") 하드웨어, 소프트웨어의 저장을 위한 판독 전용 메모리(read-only memory "ROM"), 랜덤 액세스 메모리(random access memory "RAM"), 및 비휘발성 저장소를 내재적으로 포함할 수 있다.
- [0035] 다른 종래 및/또는 관용적인 하드웨어가 또한 포함될 수 있다. 유사하게, 도면에서 나타낸 스위치들은 또한 개념적이다. 이들의 기능은 프로그램 로직의 운용으로, 전용 로직으로, 프로그램 제어 및 전용 로직의 상호 작용으로, 또는 수동적으로 실행될 수 있으며, 이 특정 기술들은 컨텍스트로부터 더욱 명확하게 이해되는 바와 같이 구현자에 의해 선택 가능하다.
- [0036] 그 청구범위에서, 특정 기능을 실행하기 위한 수단으로 표현된 요소들은 예를 들어 a) 그 기능을 실행하는 회로 요소들의 조합 또는 b) 기능을 실행하기 위해 그 소프트웨어를 실행하는 데에 적합한 회로와 조합된 펌웨어, 마이크로코드 등을 포함하는 어느 형태나 가능한 소프트웨어를 포함하여, 그 기능을 실행하는 어느 방식이나 포함하고자 하는 것이다. 이러한 청구범위에 의해 정의되는 본 원리는 청구범위가 요구하는 방식으로 여러 인용된 수단으로 제공되는 기능들이 조합되어 함께 하게 된다는 사실에 있는 것이다. 따라서 이들 기능들을 제공할 수 있는 어느 수단들이나 여기 나타낸 것들과 균등한 것으로 간주된다.
- [0037] 명세서에서 본 원리의 "일 실시예" 또는 "실시예" 뿐만 아니라, 그의 변형들의 언급은 본 실시예와 관련하여 설명된 특정한 특성, 구조, 특징들은 본 원리의 최소한의 일 실시예에 포함된다는 것을 의미한다. 따라서, 명세서 전체에 걸쳐 여러 위치에 나타나는 문구 "일 실시예" 또는 "실시예" 뿐만 아니라 그 외 다른 변형들의 출현은 반드시 모두 동일한 실시예를 언급하는 것이 아니다.
- [0038] 예를 들어, "A/B", "A 및/또는 B" 및 "A 및 B 중 적어도 하나"의 경우에서와 같이, 다음 "/", "및/또는" 및 "적어도 하나"의 이용은 제1 목록의 옵션 (A)만, 또는 제2 목록의 옵션 (B)만, 또는 두 옵션 (A 및 B)의 선택을 포함하고자 하는 것임이 이해되어야 한다. 다른 예로서, "A, B, 및/또는 C"의 경우 및 "A, B, 및 C 중 적어도 하나"의 경우에, 이런 문구들은 제1 목록의 옵션 (A)만, 또는 제2 목록의 옵션 (B)만, 제3 목록의 옵션 (C)만, 또는 제1 및 제2 목록의 옵션 (A 및 B)만, 또는 제1 및 제3 목록의 옵션 (A 및 C)만, 또는 제2 및 제3 목록의 옵션 (B 및 C)만, 또는 모든 세 목록 (A 및 B 및 C)의 선택을 포함하고자 하는 것이다. 이는 이 기술 및 관련 기술의 당업자에 의해 명백하게 되는 바와 같이, 많은 목록의 항목들에도 확장될 수 있다.
- [0039] 더구나, 본 원리의 하나 이상의 실시예들은 MPEG-4 AVC 표준과 관련하여 여기 개시되어 있는 반면, 본 원리는 이 표준에만 제한되는 것이 아니므로, 본 원리의 정신을 유지하면서, 그 외 다른 비디오 코딩 표준, 권장 및 MPEG-4 AVC 표준의 확장을 포함하는 확장과 관련하여 이용될 수 있다.
- [0040] 또한, 용어 "예측", "예측기" 및 "가설"은 여기에서 상호 교환적으로 이용된다는 것이 이해되어야 한다.
- [0041] 또한, 여기에서 이용되는 용어 "블럭"은 매크로블럭, 매크로블럭 구획, 서브매크로블럭 및 서브매크로블럭 구획 중 임의의 것을 말하는 것이 이해되어야 한다. 더구나, 용어 "블럭"은 또한 예를 들어, 매크로블럭, 매크로블럭 구획, 서브매크로블럭 및 서브매크로블럭 구획 중 임의의 것으로부터 세그먼트된 웨지를 말할 수 있다.
- [0042] 부가적으로, 용어 "명시적 부호화" 및 "명시적 복호화"는 각각 부호화나 복호화에 관한 대응하는 문장이 비트스트림(예를 들어, 문구가 대응하는 비디오 컨텐트를 포함하는 비트스트림 또는 개별의 비트스트림)에 포함되도록

부호화나 복호화가 실행되는 경우를 말한다는 것이 이해되어야 한다. 반대로, 용어 "암시적 부호화" 및 "암시적 복호화"는 각각 부호화나 복호화에 관한 대응하는 문장이 비트스트림(예를 들어, 구문이 대응하는 비디오 컨텐트를 포함하는 비트스트림이나 개별의 비트스트림)에 포함되지 않도록 부호화나 복호화가 실행되는 경우를 말한다. 그보다, 대응하는 구문은 다른 구문으로부터 유도될 수 있는 등이다.

[0043] 도 3을 참조하면, 본 원리가 적용될 수 있는 예시의 비디오 부호화기는 일반적으로 참조 부호 300으로 나타낸다.

[0044] 비디오 부호화기(300)는 조합기(385)의 비반전 입력과 신호 통신되는 출력을 갖는 프레임 오더링 버퍼(310)를 포함한다. 조합기(385)의 출력은 변환기 및 양자화기(325)의 제1 입력과 신호 통신되게 접속된다. 변환기 및 양자화기(325)의 출력은 엔트로피 부호화기(345)의 제1 입력 및 역변환기 및 역양자화기(350)의 제1 입력과 신호 통신되게 접속된다. 엔트로피 부호화기(345)의 출력은 조합기(390)의 제1 비반전 입력과 신호 통신되게 접속된다. 조합기(390)의 출력은 출력 버퍼(335)의 제1 입력과 신호 조합되게 접속된다.

[0045] 부호화기 제어기(305)의 제1 출력은 프레임 오더링 버퍼(310)의 제2 입력, 역변환기 및 역양자화기(350)의 제2 입력, 영상 타입 판정 모듈(315)의 입력, 매크로블럭 타입 (MB-type) 판정 모듈(320)의 제1 입력, 인트라 예측 모듈(360)의 제2 입력, 디블로킹 필터(deblocking filter; 365)의 제2 입력, 움직임 보상기 및 템플릿 매칭 예측기(370)의 제1 입력, 움직임 예측기(375)의 제1 입력 및 참조 영상 버퍼(reference picture buffer, 380)의 제2 입력과 신호 통신되게 접속된다.

[0046] 부호화기 제어기(305)의 제2 출력은 서플리멘털 인핸스먼트 인포메이션 (Supplemental Enhancement Information (SEI)) 삽입기(330)의 제1 입력, 변환기 및 양자화기(325)의 제2 입력, 엔트로피 부호화기(345)의 제2 입력, 출력 버퍼(335)의 제2 입력 및 시퀀스 파라미터 세트 및 영상 파라미터 세트 (Sequence Parameter Set(SPS) Picture Parameter Set(PPS)) 삽입기(340)의 입력과 신호 통신 접속된다.

[0047] SEI 삽입기(330)의 출력은 조합기(390)의 제2 비반전 입력과 신호 통신되게 접속된다.

[0048] 영상 타입 판정 모듈(315)의 제1 출력은 프레임 오더링 버퍼(310)의 제3 입력과 신호 통신되게 접속된다. 영상 타입 판정 모듈(315)의 제2 출력은 매크로블럭 타입 판정 모듈(320)의 제2 입력과 신호 통신되게 접속된다.

[0049] 시퀀스 파라미터 세트(SPS) 및 영상 파라미터 세트(PPS) 삽입기(340)의 출력은 조합기(390)의 제3 비반전 입력과 신호 통신되게 접속된다.

[0050] 역양자화기 및 역변환기(350)의 출력은 조합기(319)의 제1 비반전 입력과 신호 통신되게 접속된다. 조합기(319)의 출력은 인트라 예측 모듈(360)의 제1 입력과 디블로킹 필터(365)의 제1 입력과 신호 통신되게 접속된다. 디블로킹 필터(365)의 출력은 참조 영상 버퍼(380)의 제1 입력과 신호 통신되게 접속된다. 참조 영상 버퍼(380)의 출력은 움직임 추정기(375)의 제2 입력, 움직임 보상기 및 템플릿 매칭 예측기(370)의 제3 입력, 및 템플릿 검색기(344)의 제1 입력과 신호 통신되게 접속된다. 템플릿 검색기(344)의 출력은 움직임 보상기 및 템플릿 매칭 예측기(370)의 제4 입력과 신호 통신되게 접속된다. 움직임 추정기(375)의 제1 출력은 움직임 보상기 및 템플릿 매칭 예측기(370)의 제2 입력과 신호 통신되게 접속된다. 움직임 추정기(375)의 제2 출력은 엔트로피 부호화기(345)의 제3 입력과 신호 통신되게 접속된다. 움직임 추정기(375)의 제3 출력은 템플릿 검색기(344)의 제2 입력과 신호 통신되게 접속된다. 부호화기 제어기(305)의 제3 출력은 템플릿 검색기(344)의 제3 입력과 신호 통신되게 접속된다.

[0051] 움직임 보상기 및 템플릿 매칭 예측기(370)의 출력은 스위치(397)의 제1 입력과 신호 통신되게 접속된다. 인트라 예측 모듈(360)의 출력은 스위치(397)의 제2 입력과 신호 통신되게 접속된다. 매크로블럭 타입 판정 모듈(320)의 출력은 스위치(397)의 제3 입력과 신호 통신되게 접속된다. 스위치(397)의 제3 입력은 스위치의 "데이터" 입력 (제어 입력, 즉 제3 입력과 비교하여)이 움직임 보상기 및 템플릿 매칭 예측기(370) 또는 인트라 예측 모듈(360)에 의해 제공되는지의 여부를 판정한다. 스위치(397)의 출력은 조합기(319)의 제2 비반전 입력 및 조합기(385)의 반전 입력과 신호 통신되게 접속된다.

[0052] 프레임 오더링 버퍼(310)의 제1 입력 및 부호화기 제어기(305)의 입력은 입력 영상을 수신하기 위해서, 부호화기(100)의 입력으로 이용 가능하다. 더구나, 서플리멘털 인핸스먼트 인포메이션(SEI) 삽입기(330)의 제2 입력은 메타데이터를 수신하기 위해서, 부호화기(300)의 입력으로 이용 가능하다. 출력 버퍼(335)의 출력은 비트스트림을 출력하기 위해서, 부호화기(300)의 출력으로 이용 가능하다.

[0053] 도 4를 참조하면, 본 원리가 적용되는 예시의 비디오 복화하기는 일반적으로 참조 부호 400으로 나타낸다.

- [0054] 비디오 복호화기(400)는 엔트로피 복호화기(445)와 신호 통신되게 접속된 출력을 갖는 입력 버퍼(410)을 포함한다. 엔트로피 복호화기(445)의 제1 출력은 역변환기 및 역양자화기(450)의 제1 입력과 신호 통신되게 접속된다. 역변환기 및 역양자화기(450)의 출력은 조합기(425)의 제2 비반전 입력과 신호 통신되게 접속된다. 조합기(425)의 출력은 디블로킹 필터(465)의 제2 입력 및 인트라 예측 모듈(460)의 제1 입력과 신호 통신되게 접속된다. 디블로킹 필터(465)의 제2 출력은 참조 영상 버퍼(480)의 제1 입력과 신호 통신되게 접속된다. 참조 영상 버퍼(480)의 출력은 템플릿 검색기(444)의 제1 입력과 신호 통신되게 접속된다. 템플릿 검색기(444)의 출력은 움직임 보상기 및 템플릿 매칭 예측기(470)의 제2 입력과 신호 통신되게 접속된다.
- [0055] 엔트로피 복호화기(445)의 제2 출력은 움직임 보상기 및 템플릿 매칭 예측기(470)의 제3 입력, 디블로킹 필터(465)의 제1 입력 및 템플릿 검색기(444)의 제2 입력과 신호 통신되게 접속된다. 엔트로피 복호화기(445)의 제3 출력은 복호화기 제어기(405)의 입력과 신호 통신 접속된다. 복호화기 제어기(405)의 제1 입력은 엔트로피 복호화기(445)의 제2 입력과 신호 통신 접속된다. 복호화기 제어기(405)의 제2 출력은 역변환기 및 역양자화기(450)의 제2 입력과 신호 통신 접속된다. 복호화기 제어기(405)의 제3 출력은 디블로킹 필터(465)의 제3 입력과 신호 통신되게 접속된다. 복호화기 제어기(405)의 제4 출력은 인트라 예측 모듈(460)의 제2 입력, 움직임 보상기 및 템플릿 매칭 예측기(470)의 제1 입력, 참조 영상 버퍼(480)의 제2 입력, 및 템플릿 검색기(444)의 제3 입력과 신호 통신되게 접속된다.
- [0056] 움직임 보상기 및 템플릿 매칭 예측기(470)의 출력은 스위치(497)의 제1 입력과 신호 통신되게 접속된다. 인트라 예측 모듈(460)의 출력은 스위치(497)의 제2 입력과 신호 통신되게 접속된다. 스위치(497)의 출력은 조합기(425)의 제1 비반전 입력과 신호 통신되게 접속된다.
- [0057] 입력 버퍼(410)의 입력은 입력 비트스트림을 수신하기 위해서, 복호화기(400)의 입력으로 이용 가능하다. 디블로킹 필터(465)의 제1 출력은 출력 영상을 출력하기 위해서, 복호화기(400)의 출력으로 이용 가능하다.
- [0058] 상기한 바와 같이, 본 원리는 비디오 부호화 및 복호화에서의 템플릿 매칭 예측(TMP)을 위한 방법 및 장치에 관한 것이다.
- [0059] 또한 상기한 바와 같이, 템플릿 매칭 예측의 예측 성능은 타겟 블럭과 그 대응 템플릿 간의 상관 관계에 매우 의존한다. 도 1에 나타낸 바와 같이, 본 기술에서는 상부에 및/또는 좌측으로 타겟 블럭을 둘러싸는 재구성된 화소를 이용하는 것이 잘 알려져 있다. 그러나, 이러한 방식으로 템플릿 매칭 예측을 이용하게 되면 특히 템플릿이 타겟의 신호와 상관 관계가 적을 때 부호화기 및 복호화기 성능을 떨어뜨리게 된다.
- [0060] 우리는 이 한계점을 인식하고 이에 따라 암시적 움직임 유도를 위해 개선된 템플릿 매칭 예측 방법을 요구한다. 따라서 우리는 다수 가설 인터 예측을 위한 새로운 템플릿 매칭 기술을 설명하며, 이 때 몇 가설에 대한 움직임 정보는 템플릿 매칭에 의해 유도된다. 이를 행하는 데에는, 템플릿 매칭 예측 및 명시적 움직임 예측의 상조적인 조합이 효과적이다. 이는 움직임 정보를 위한 비트율 오버헤드를 줄이므로, 부호화 효율을 개선한다. 본 원리의 실시예가 인터 예측의 관점에서 주로 설명되고 있지만, 여기에서 본 원리의 개시가 제공되면, 이 기술 및 관련 기술의 당업자라면 본 원리가 또한 인트라 예측의 이용에도 적용될 수 있다는 것이 용이하게 인식될 것이라는 것이 이해되어야 한다.
- [0061] 따라서, 본 원리에 따르면, 우리는 템플릿 매칭 예측의 한계점과 명시적 및 암시적 복수 가설의 움직임 보상 예측의 한계점을 해결하는 방법과 장치를 설명한다. 또한, 특히 우리는 복수 가설의 움직임 보상된 예측을 개선하기 위해 조인트 암시적 및 명시적 예측을 행하기 위해 템플릿 매칭 예측을 이용하여 설명한다. 이전의 방법에서, 템플릿은 목표의 화소/블럭의 재구성된 주변 화소로부터 취한다. 주변 화소는 때로 타겟 화소를 나타내지 못하므로, 템플릿 매칭 예측의 성능은 종래의 블럭 매칭 기구 보다 더 낮아질 수 있다. 그러나, 본 원리의 하나 또는 그 이상의 실시예에 따르면, 우리는 템플릿이 도 5에서 나타낸 바와 같이, 타겟 화소/블럭의 예측이 될 수 있는 방법 및 장치를 설명한다. 도 5를 참조하면, 단일 예측(단방향 예측)과 결합된 템플릿 매칭 예측(TMP) 기구는 보통 참조 부호 500으로 나타내었다. 단일 예측인 템플릿 매칭 예측 기구(500)는 현재 프레임(510), 목록 0의 참조 프레임(520), 목록 1의 참조 프레임(530)을 수반한다. 현재 프레임(510)은 타겟 블럭(511)을 포함한다. 목록 0의 참조 프레임(520)은 예측(521) (또한 "P<sub>0</sub>"으로 표시)을 포함한다. 목록 1의 참조 프레임은 예측(531) (또한 "P<sub>1</sub>"로 표시)를 포함한다. 움직임 벡터 MVO은 타겟 블럭(511)과 예측부(521) 사이의 움직임을 지정하는 데에 이용된다. 예측 P<sub>0</sub>가 타겟 블럭과 매우 가까우면, 템플릿 매칭 예측의 성능은 예측 P<sub>0</sub>에 기초하여 템플릿을 형성하는 것이 매우 효율적일 수 있다. 제1 실시예에서, 템플릿은 현재 블럭의 인터 예측일 수 있다. 제2 실시예에서, 템플릿은 현재 블럭의 인트라 예측일 수 있다. 우리가 이용하는 나머지 예측

을 나타내도록 약간의 부가 정보를 보낼 수 있거나, 디폴트 예측을 가정할 수도 있다. 예를 들어, 제1 실시예에서, 우리는 인터 예측이 항상 움직임 벡터 예측기로부터 유도된 것, 또는 스kip 또는 다이렉트 모드로부터 유도된 움직임 벡터로서 취해진다고 가정할 수 있거나, 우리는 예측을 위해 움직임 벡터 정보를 명시적으로 보낼 수 있다. 제2 실시예에서, 우리는 항상 DC 인트라 예측이 이용된다고 가정할 수 있거나 우리는 MPEG-4 AVC 표준에서 어느 인트라 예측이 이용되는지를 나타내기 위해 부가 정보를 보낼 수 있다. 제3 실시예에서, 우리는 공간 인접 화소로부터 템플릿 검색을 이용하여 타겟 블럭에 대해 템플릿을 채울 수 있다.

[0062] **템플릿 매칭을 이용한 다수 가설**

[0063] 이 기구에서, 템플릿은 명시적 움직임 추정에 의한 인터 예측으로 성취된다. 움직임 벡터는 MPEG-4 AVC 표준에서와 같이 움직임 벡터 예측기로부터 유도될 있거나, 비트스트림으로 명시적으로 보내질 수 있다. 다음에, 우리는 움직임 보상된 예측을 얻기 위해서 이용 가능한 참조 영상을 검색하도록 템플릿을 이용한다. 이런 식으로, 우리는 명시적 움직임 추정과 암시적 움직임 추정 둘다를 이용할 수 있다. 오버헤드를 더욱 초래하지 않고 동일하거나 다른 참조 화상으로부터 다수의 예측을 얻을 수 있기 때문에, 이 방법으로 다수의 가설을 구현할 수 있다.

[0064] 다음에서, 우리는 템플릿 매칭 예측이 다수 가설 움직임 보상된 예측을 개선하기 위해 이용되는 네가지 다른 실시예를 설명한다. 처음 두 실시예들은 B 프레임에 촛점을 둔다. 제3 실시예는 다수의 참조 영상의 관점에서 설명한다. 본 원리는 P 프레임에 또한 적용 가능하다. 제4 실시예에서, 우리는 인트라 영상 부호화를 위해 이 방법이 또한 어떻게 이용될 수 있는지를 설명한다.

[0065] **실시예 1: 단일 참조 목록 예측 (단방향 예측)과 결합된 TMP**

[0066] 이 실시예에서, 명시적 움직임 추정은 타겟 블럭에 대한 예측 데이터 ( $P_0$ )를 얻기 위해 하나의 참조 목록에 대해 행해진 다음에, 이는 예측 데이터 ( $P_1$ )를 얻기 위해 다른 참조 목록에서 검색할 템플릿으로 이용된다. 다음에, 최종 예측 ( $P$ )는  $P_0$ 과  $P_1$ 의 가중 평균치가 된다. 가중치는 단순 평균 (0.5와 동일한 가중치) 또는 그 외 왜곡 치수, 현재 영상과의 기준 거리 등에 기초한 가중치일 수 있다. 예를 도 5에 나타내었다.

[0067] 이 방법은 예를 들어, 이에만 제한되는 것은 아니지만, 모드 MxNL0 또는 NxNL1과 같은 MPEG-4 AVC 표준의 단방향 예측 모드와 용이하게 조합될 수 있으며, 이 때 M, N=16, 8, 4이고, L0 및 L1은 각각 목록 0 또는 목록 1로부터 나온 예측을 나타낸다. 우리가 템플릿 매칭 예측된 단방향 예측이나 양방향 예측이 이 모드에 이용되는지를 나타내기 위해 매크로블럭 레벨이나 서브매크로블럭 레벨의 플랙을 부가할 수 있다.

[0068] 1인 bi\_tmp\_flag는 단방향 예측과 템플릿 매칭 예측을 이용한 양방향 예측이 적용된 것을 특정한다. 0인 bi\_tmp\_flag는 단방향 예측이 적용된 것을 특정한다.

[0069] 일 실시예에 따른 16x16L0에 대한 일 예의 부호화 과정은 다음과 같이 설명된다:

[0070] (1)  $P_0$ 을 얻기 위해 참조 목록 0에서의 매크로블럭 16x16의 움직임 추정을 실행하고  $P_0$ 에 대한 RDcost를 연산한다.

[0071] (2)  $P_1$ 을 얻기 위해서 참조 목록 1에서 검색하도록  $P_0$ 을 템플릿으로 이용하고,  $0.5(P_0+P_1)$ 에 대한 RDcost를 연산한다.

[0072] (3) (1)과 (2)로부터 RDcost를 비교하여 bi\_tmp\_flag을 세트한다. (2)의 RDcost가 (1)의 것보다 작으면, bi\_tmp\_flag=1로 세트한다; 그렇지 않으면, bi\_tmp\_flag=0으로 세트한다.

[0073] 일 실시예에 따른 16x16L0에 대한 대응하는 복호화 과정은 다음과 같이 설명한다:

[0074] (1) bi\_tmp\_flag의 구문, 16x16L0에 대한 움직임 벡터를 파싱(parse) 및 복호화한다.

[0075] (2)  $P_0$ 을 얻기 위해 참조 목록 0에서 매크로블럭 16x16에 대한 움직임 보상을 실행한다.

[0076] (3) bi\_tmp\_flag이 0과 같으면, 중지한다. 그렇지 않으면,

[0077] (4) 다음을 실행한다:  $P_1$ 을 얻기 위해 참조 목록 1에서 검색하도록  $P_0$ 을 템플릿으로 이용하여 최종 예측을  $0.5(P_0+P_1)$ 으로 얻는다.

- [0078] 우리는 템플릿 검색을 위해 템플릿 크기는  $P_0$ 의 크기와 다를 수 있다는 것에 주목했다. 예를 들어, 우리는  $16 \times 16$ 인  $P_1$ 의 검색을 위해  $16 \times 16$  블럭 크기 대신에  $8 \times 8$  블럭 크기를 이용한 템플릿 매칭 예측을 실행할 수 있다. 다르게, 우리는  $P_0$ 의 주변 화소와 결합하여 템플릿의 크기를 증가시킬 수 있는  $20 \times 20$ 을 이용한 템플릿 매칭 예측을 실행할 수 있다.
- [0079] 도 6을 참조하여, 단일 참조 목록 예측 (단방향 예측)으로 템플릿 매칭 예측을 이용한 비디오 부호화 방법은 일반적으로 참조 부호 600으로 나타내었다. 방법 600은 판정 블럭(606)으로 제어를 보내는 시작 블럭(603)을 포함한다. 판정 블럭(606)은 현재 모드가 템플릿 매칭 모드(TMP)인지의 여부를 판정한다. 그렇다면, 제어는 판정 블럭(609)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(663)으로 보내진다.
- [0080] 판정 블럭(609)은 명시적인 움직임 추정 방향 (정방향  $L_0$  또는 역방향  $L_1$ )을 판정한다. 명시적인 움직임 추정 방향이 역방향(BAK)이면, 제어는 기능 블럭(612)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(627)으로 보내진다.
- [0081] 기능 블럭(612)은 예측  $P_1$  및 움직임 벡터  $Mv$ 를 얻기 위해 참조 목록 1에서 움직임 추정을 실행하고, 제어를 기능 블럭(615)으로 보낸다. 기능 블럭(615)은  $P_0$ 을 얻기 위해  $P_1$ 에 기초하여 참조 목록 0에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(618)으로 보낸다. 기능 블럭(618)은  $P = (P_0 + P_1) / 2$ 를 연산하고, 제어를 판정 블럭(621)으로 보낸다. 판정 블럭(621)은  $P_1$ 의 비트율 왜곡치(rate-distortion cost), 즉  $Rd(P_1)$ 이  $P$ 의 비트율 왜곡치, 즉  $Rd(P)$  보다 작은지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(624)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(657)으로 보내진다.
- [0082] 기능 블럭(624)는  $bi\_tmp\_flag=0$ 을 세트하고, 예측= $P_1$ 을 세트하고, 제어를 기능 블럭(642)으로 보낸다.
- [0083] 기능 블럭(642)는 나머지를 연산하고, 제어를 기능 블럭(645)으로 보낸다. 기능 블럭(645)은 예측의 비트율 왜곡, 즉  $Rd$ (예측)을 연산하고, 제어를 기능 블럭(648)으로 보낸다. 기능 블럭(648)은 모드 선택을 실행하고, 제어를 기능 블럭(651)으로 보낸다. 기능 블럭(651)은 움직임 벡터  $Mv$  및 그 외 구문을 부호화하고, 제어를 기능 블럭(654)으로 보낸다. 기능 블럭(654)은 나머지를 부호화하고, 제어를 종료 블럭(699)으로 보낸다.
- [0084] 기능 블럭(657)은  $bi\_tmp\_flag=1$ 을 세트하고, 예측= $P$ 을 세트하고, 제어를 기능 블럭(642)으로 보낸다.
- [0085] 기능 블럭(627)은 예측  $P_0$  및 움직임 벡터  $Mv$ 를 얻기 위해 참조 목록 0에서 움직임 추정을 실행하고, 제어를 기능 블럭(630)으로 보낸다. 기능 블럭(630)은  $P_1$ 을 얻기 위해  $P_0$ 에 기초하여 참조 목록 1에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(633)으로 보낸다. 기능 블럭(633)은  $P = (P_0 + P_1) / 2$ 를 연산하고, 제어를 판정 블럭(636)으로 보낸다. 판정 블럭(636)은  $P_0$ 의 비트율 왜곡치, 즉  $Rd(P_0)$ 이  $P$ 의 비트율 왜곡치, 즉  $Rd(P)$  보다 작은지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(639)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(660)으로 보내진다.
- [0086] 기능 블럭(639)은  $bi\_tmp\_flag=0$ 로 세트하고, 예측= $P_0$ 으로 세트하고, 제어를 기능 블럭(642)으로 보낸다.
- [0087] 기능 블럭(660)은  $bi\_tmp\_flag=1$ 로 세트하고, 예측= $P$ 를 세트하고, 제어를 기능 블럭(642)으로 보낸다.
- [0088] 기능 블럭(663)은 non-TMP 모드(들)를 이용하여 부호화하고, 제어를 기능 블럭(648)으로 보낸다.
- [0089] 도 7을 참조하여, 단일 참조 목록 예측 (단방향 예측)된 템플릿 매칭 예측을 이용한 비디오 복호화 방법은 일반적으로 참조 부호 700으로 나타낸다. 방법(700)은 기능 블럭(706)으로 제어를 보내는 시작 블럭(703)을 포함한다. 기능 블럭(706)은 구문을 파싱하고, 제어를 판정 블럭(709)으로 보낸다. 판정 블럭(709)은 현재 모드가 템플릿 매칭 예측 (TMP) 모드인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(712)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(751)으로 보내진다.
- [0090] 기능 블럭(712)은 움직임 벡터  $Mv$ , 나머지, 및  $bi\_tmp\_flag$ 를 복호화하고, 제어를 판정 블럭(715)으로 보낸다. 판정 블럭(715)은 명시적 움직임 추정 방향을 결정한다. 명시적 움직임 추정 방향이 역방향이면(BAK), 제어는 기능 블럭(718)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(730)으로 보내진다.
- [0091] 기능 블럭(718)은  $Mv$ 로 참조 목록 1에서  $P_1$ 을 얻고, 제어를 판정 블럭(721)으로 보낸다. 판정 블럭(721)은

bi\_tmp\_flag=1인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(724)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(745)으로 보내진다.

[0092] 기능 블럭(724)은  $P_0$ 을 얻기 위해  $P_1$ 에 기초하여 참조 목록 0에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(727)으로 보낸다. 기능 블럭(727)은 예측=( $P_0+P_1$ )/2를 연산하고, 제어를 기능 블럭(742)으로 보낸다.

[0093] 기능 블럭(742)은 나머지를 예측에 부가하여, 현재 블럭을 보상하고, 제어를 종료 블럭(799)으로 보낸다.

[0094] 기능 블럭(745)은 예측= $P_0$ 로 세트하고, 제어를 기능 블럭(742)으로 보낸다.

[0095] 기능 블럭(730)은  $Mv$ 로 참조 목록 0에서  $P_0$ 을 얻고, 제어를 판정 블럭(733)으로 보낸다. 판정 블럭(733)은 bi\_tmp\_flag=1인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(736)으로 보낸다. 그렇지 않으면, 제어는 기능 블럭(743)으로 보내진다.

[0096] 기능 블럭(736)은  $P_1$ 을 얻기 위해  $P_0$ 에 기초하여 참조 목록 1에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(739)으로 보낸다. 기능 블럭(739)은 예측=( $P_0+P_1$ )/2를 연산하고, 제어를 기능 블럭(742)으로 보낸다.

[0097] 기능 블럭(748)은 예측= $P_0$ 으로 세트하고 제어를 기능 블럭(742)으로 보낸다.

[0098] 기능 블럭(751)은 non-TMP 모드를 이용하여 복호화하고, 제어를 종료 블럭(799)으로 보낸다.

#### 실시예 2: 양방향 예측과 조합된 TMP

[0100] 일 실시예에서, 종래의 양방향 예측이 실행된 다음에 일 참조 목록의 각 예측에 대해, 우리는 그 대응하는 템플릿 매칭 예측을 다른 참조 목록에서 찾을 수 있다. 최종 결과는 3 또는 4개의 가설 예측이 생긴다. 우리는 이 방법에서 다수의 참조 영상이 가능할 때, 템플릿 매칭 예측을 명시적 예측에 대한 것과 동일한 참조 영상에 제한하지 않는다. 도 8은 이런 예를 나타낸다. 도 8을 참조하면, 양방향 예측과 조합되는 템플릿 매칭 예측의 예를 참조 부호 800으로 나타낸다. 양방향 예측된 템플릿 매칭 예측(800)은 현재 프레임(810), 목록 0의 참조 프레임 1(820), 목록 0의 참조 프레임 0(830), 및 목록 1의 참조 프레임(840)을 수반한다. 현재 프레임(810)은 타겟 블럭(811)을 포함하고, 목록 0의 참조 프레임 1(820)은 예측(821) (또는 "P<sub>3</sub>"로 지정)을 포함하고, 목록 0의 참조 프레임 0(830)은 예측(831) (또는 "P<sub>3</sub>"로 지정) 및 예측(832) (또는 "P<sub>0</sub>"로 지정)을 포함하고, 목록 1의 참조 프레임(840)은 예측(841) (또는 "P<sub>1</sub>"로 지정) 및 예측(842) (또는 "P<sub>2</sub>"로 지정)을 포함한다. 우리는  $P_0$ 을 얻기 위해서 참조 목록 0에서 움직임 추정을 먼저 실행하고, 다음에  $P_2$ 를 얻기 위해서 참조 목록 1에서 검색하도록  $P_0$ 을 템플릿으로 이용하고,  $P_3$ 를 얻기 위해서 참조 목록 0에서 검색하도록  $P_1$ 을 템플릿으로 이용한다. 다수의 참조 영상이 이용되면,  $P_3$  및  $P_0$ 은 동일한 참조 영상에 존재할 필요는 없다. 움직임 벡터 MV0이 타겟 블럭(811)과 예측(832) 간의 움직임을 나타내는 데에 이용되고, 움직임 벡터 MV1은 타겟 블럭(811)과 예측(841) 간의 움직임을 나타내는 데에 이용된다.

[0101] 이 방법은 예를 들어, 이들에만 제한하는 것은 아니지만 모드 MxNB1 또는 다이렉트 모드와 같은 MPEG-4 AVC 표준에 대한 양방향 예측 모드와 용이하게 조합될 수 있으며, 이 때 M, N=16, 8, 4이고, B1은 양방향 예측을 나타낸다. 우리는 이 모드에 대해 얼마나 많은 가설들이 템플릿 매칭 예측의 이용에 실행되는지를 특정하기 위해 매크로블럭 레벨이나 서브매크로블럭 레벨의 표시자를 부가한다.

[0102] 0과 동일한 multi\_tmp\_mode는 종래의 양방향 예측이 적용된 것을 특정한다. 1과 동일한 multi\_tmp\_mode는 3개의 가설들이 적용된 것을 특정하고, 종래의 양방향 예측에다 템플릿으로 목록 0 예측을 이용한 템플릿 매칭 예측과 목록 1에서의 검색을 더한 것이 실행된다. 2인 multi\_tmp\_mode는 3개의 가설이 적용된 것을 특정하고, 종래의 양방향 예측에다 템플릿으로 목록 1 예측을 이용한 템플릿 매칭 예측과 목록 0에서의 검색을 더한 것이 실행된다. 3인 multi\_tmp\_mode는 4개의 가설이 적용된 것을 특정하며, 종래의 양방향 예측에다 템플릿으로 목록 1 예측을 이용한 템플릿 매칭 예측, 목록 0에서의 검색, 템플릿으로 목록 0 예측을 이용한 템플릿 매칭 예측, 및 목록 1에서의 검색을 더한 것이 실행된다.

[0103] 일 실시예에 따른 16x16B1에 대한 일 예의 부호화 과정은 다음과 같이 설명된다.

[0104] (1)  $P_0$ 을 얻기 위해 참조 목록 0에서 및  $P_1$ 을 얻기 위해 참조 목록 1에서 매크로블럭 16x16에 대한 움직임 추정

을 실행하여,  $(P_0+P_1)/2$ 에 대한 RDcost를 연산한다.

[0105] (2)  $P_2$ 를 얻기 위해 참조 목록 1에서 검색하도록  $P_0$ 을 템플릿으로 이용하고,  $(P_0+P_1+P_2)/3$ 에 대한 RDcost를 연산한다.

[0106] (3)  $P_3$ 를 얻기 위해 참조 목록 0에서 검색하도록 템플릿으로  $P_1$ 을 이용하고,  $(P_0+P_1+P_2)/3$ 에 대한 RDcost를 연산한다.

[0107] (4)  $(P_0+P_1+P_2+P_3)/4$ 에 대한 RDcost를 연산한다.

[0108] (5) (1), (2), (3) 및 (4)로부터의 RDcost를 비교한다.

[0109] (1)의 RDcost가 최소이면, multi\_tmp\_mode를 0로 세트한다.

[0110] (2)의 RDcost가 최소이면, multi\_tmp\_mode를 1로 세트한다.

[0111] (3)의 RDcost가 최소이면, multi\_tmp\_mode를 2로 세트한다.

[0112] (4)의 RDcost가 최소이면, multi\_tmp\_mode를 3로 세트한다.

[0113] 일 실시예에 따른 16x16B1에 대한 대응하는 복호화 과정을 다음과 같이 설명한다:

[0114] (1) multi\_tmp\_mode의 구문 및 16x16B1에 대한 움직임 벡터를 파싱 및 복호화한다.

[0115] (2)  $P_0$ 을 얻기 위해 참조 목록 0에서 및  $P_1$ 을 얻기 위해 참조 목록 1에서 매크로블럭 16x16에 대한 움직임 보상을 실행한다.

[0116] (3) multi\_tmp\_mode가 0과 같으면, 최종 예측은  $(P_0+P_1)/2$ 이고, 그렇지 않으면,

[0117] (4) multi\_tmp\_mode가 1과 같으면,  $P_2$ 를 얻기 위해 참조 목록 1에서 검색하도록 템플릿으로  $P_0$ 을 이용하고, 최종 예측을  $(P_0+P_1+P_2)/3$ 으로 얻는다.

[0118] (5) multi\_tmp\_mode가 2과 같으면,  $P_3$ 를 얻기 위해 참조 목록 1에서 검색하도록  $P_0$ 을 템플릿으로 이용하고, 최종 예측을  $(P_0+P_1+P_3)/3$ 으로 얻는다.

[0119] (6) 그렇지 않으면, 최종 예측을  $(P_0+P_1+P_2+P_3)/4$ 으로 얻는다.

[0120] 도 9를 참조하면, 양방향 예측된 템플릿 매칭 예측을 이용한 비디오 부호화 방법은 일반적으로 참조 부호 900으로 나타낸다. 이 방법 900은 판정 블럭(906)으로 제어를 보내는 시작 블럭(903)을 포함한다. 판정 블럭(906)은 현재 모드가 템플릿 매칭 예측(TMP) 모드인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(909)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(954)으로 보내진다.

[0121] 기능 블럭(909)은 예측  $P_0$ 와 움직임 벡터 Mv0을 얻도록 참조 목록 0에서 움직임 추정을 실행하고, 제어를 기능 블럭(912)으로 보낸다. 기능 블럭(912)은 예측  $P_1$ 과 움직임 벡터 Mv1을 얻도록 참조 목록 1에서 움직임 추정을 실행하고, 제어를 기능 블럭(915)으로 보낸다. 기능 블럭(915)은  $P_2$ 를 얻기 위해  $P_0$ 에 기초하여 참조 목록 1에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(918)으로 보낸다. 기능 블럭(918)은  $P_3$ 를 얻기 위해  $P_1$ 에 기초하여 참조 목록 0에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(921)으로 보낸다. 기능 블럭(921)은 Pred0=( $P_0+P_1$ )/2, Pred1=( $P_0+P_1+P_2$ )/3, Pred2=( $P_0+P_1+P_3$ )/3, 및 Pred3=( $P_0+P_1+P_2+P_3$ )/4를 연산하고, 제어를 판정 블럭(924)으로 보낸다. 판정 블럭(924)는 Pred0, Pred1, Pred2 및 Pred3 중 어느 예측이 최소 비트율 왜곡 (Rd)을 제공하는지를 판정한다. Pred0이 최소 Rd를 제공하면, 제어는 기능 블럭(927)으로 보내진다. Pred1이 최소 Rd를 제공하면, 제어는 기능 블럭(930)으로 보내진다. Pred2가 최소 Rd를 제공하면, 제어는 기능 블럭(933)으로 보내진다. Pred3이 최소 Rd를 제공하면, 제어는 기능 블럭(936)으로 보내진다.

[0122] 기능 블럭(927)은 Multi\_TMP\_Mode=0으로 세트하고, 제어를 기능 블럭(939)으로 보낸다.

[0123] 기능 블럭(930)은 Multi\_TMP\_Mode=1으로 세트하고, 제어를 기능 블럭(939)으로 보낸다.

[0124] 기능 블럭(933)은 Multi\_TMP\_Mode=2으로 세트하고, 제어를 기능 블럭(939)으로 보낸다.

- [0125] 기능 블럭(936)은 Multi\_TMP\_Mode=3으로 세트하고, 제어를 기능 블럭(939)으로 보낸다.
- [0126] 기능 블럭(939)은 나머지를 연산하고, 제어를 기능 블럭(942)으로 보낸다. 기능 블럭(942)은 예측의 비트율 왜곡치, 즉 Rd(예측)을 연산하고, 제어를 기능 블럭(945)으로 보낸다. 기능 블럭(945)은 모드 선택을 실행하고, 제어를 기능 블럭(948)으로 보낸다. 기능 블럭(948)은 움직임 벡터와 그 외 구문을 부호화하고, 제어를 기능 블럭(951)으로 보낸다. 기능 블럭(951)은 잔여를 부호화하고, 제어를 종료 블럭(999)으로 보낸다.
- [0127] 기능 블럭(954)은 non-TMP 모드(들)을 이용하여 부호화하고, 제어를 기능 블럭(945)으로 보낸다.
- [0128] 도 10을 참조하면, 양방향 예측의 템플릿 매칭 예측을 이용한 비디오 복호화 방법은 일반적으로 참조 부호 1000으로 나타낸다. 이 방법(1000)은 시작 블럭(1003)을 포함하고 제어를 기능 블럭(1006)으로 보낸다. 기능 블럭(1006)은 구문을 파싱하고, 제어를 판정 블럭(1009)으로 보낸다. 판정 블럭(1009)는 현재 모드가 템플릿 매칭 예측(TMP) 모드인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1012)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1045)으로 보내진다.
- [0129] 기능 블럭(1012)는 목록 0의  $P_0$  및 목록 1의  $P_1$ 을 얻기 위해서 움직임 보상을 실행하고, 제어를 판정 블럭(1015)으로 보낸다. 판정 블럭(1015)은 어느 Multi\_TMP\_Mode인지를 판정한다. 모드 0이 Multi\_TMP\_Mode이면, 제어는 기능 블럭(1018)으로 보내진다. 모드 1이 Multi\_TMP\_Mode이면, 제어는 기능 블럭(1021)으로 보내진다. 모드 2가 Multi\_TMP\_Mode이면, 제어는 기능 블럭(1027)으로 보내진다. 모드 3가 Multi\_TMP\_Mode이면, 제어는 기능 블럭(1033)으로 보내진다.
- [0130] 기능 블럭(1018)은  $Pred=(P_0+P_1)/2$ 를 연산하고, 제어를 기능 블럭(1042)으로 보낸다.
- [0131] 기능 블럭(1021)은  $P_2$ 를 얻기 위해  $P_0$ 에 기초하여 참조 목록 1에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(1024)으로 보낸다. 기능 블럭(1024)은  $Pred=(P_0+P_1+P_2)/3$ 를 연산하고, 제어를 기능 블럭(1042)으로 보낸다.
- [0132] 기능 블럭(1027)은  $P_3$ 를 얻기 위해  $P_1$ 에 기초하여 참조 목록 0에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(1030)으로 보낸다. 기능 블럭(1030)은  $Pred=(P_0+P_1+P_3)/3$ 를 연산하고, 제어를 기능 블럭(1042)으로 보낸다.
- [0133] 기능 블럭(1033)은  $P_2$ 를 얻기 위해  $P_0$ 에 기초하여 참조 목록 1에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(1036)으로 보낸다. 기능 블럭(1036)은  $P_3$ 를 얻기 위해  $P_1$ 에 기초하여 참조 목록 0에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(1039)으로 보낸다. 기능 블럭(1039)은  $Pred=(P_0+P_1+P_2+P_3)/4$ 를 연산하고, 제어를 기능 블럭(1042)으로 보낸다.
- [0134] 기능 블럭(1042)은 나머지를 예측에 부가하여, 현재 블럭을 보상하고, 제어를 종료 블럭(1099)으로 보낸다.
- [0135] 기능 블럭(1045)은 non-TMP 모드(들)을 이용하여 복호화하고, 제어를 종료 블럭(1099)으로 보낸다.
- [0136] 다른 실시예에서, 우리는 여전히 bi\_tmp\_flag를 이용하고, 이 때 1인 bi\_tmp\_flag은 최대 (4 까지) 여러 움직임 보상된 예측에 대한 다수 가설의 움직임 보상된 예측을 특정한다.
- [0137] 또 다른 실시예에서, 우리는 실시예 1로부터 생각을 확장하지만, 두 예측기의 평균  $P_2=(P_0+P_1)/2$ 으로 템플릿을 설정한 다음에, 이 템플릿을 이용하여 두 리스트의 참조 영상으로부터 템플릿 매칭 예측  $P_3$ 를 검색한다. 마지막으로, 우리는 템플릿 매칭 예측을 평균화한다  $(P_2+P_3)/2$ .
- [0138] 도 11을 참조하면, 양방향 예측의 템플릿 매칭 예측을 이용한 다른 비디오 부호화 방법은 일반적으로 참조 부호 1100으로 나타낸다. 방법(1100)은 제어를 판정 블럭(1106)으로 보내는 시작 블럭(1103)을 포함한다. 판정 블럭(1106)은 현재 모드가 템플릿 매칭 예측(TMP) 모드인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1109)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1145)으로 보내진다.
- [0139] 기능 블럭(1109)은 예측  $P_0$ 과 움직임 벡터  $Mv0$ 을 얻기 위해 참조 목록 0에서 움직임 추정을 실행하고, 제어를 기능 블럭(1112)으로 보낸다. 기능 블럭(1112)은 예측  $P_1$  및 움직임 벡터  $Mv1$ 을 얻기 위해 참조 목록 1에서 움직임 추정을 실행하고, 제어를 기능 블럭(1115)으로 보낸다. 기능 블럭(1115)은 템플릿  $P_2=(P_0+P_1)/2$ 를 세트하고,  $P_3$ 를 얻기 위해 두 참조 목록에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(1118)으로 보낸다. 기능 블

력(1118)은  $\text{Pred0}=(P_0+P_1)/2$  및  $\text{Pred1}=(P_2+P_3)/2$ 를 연산하고, 제어를 판정 블럭(1121)으로 보낸다. 판정 블럭(1121)은 어느 예측이 최소 비트율 왜곡치 (Rd)를 제공하는지를 판정한다.  $\text{Pred0}$ 이 최소 Rd를 제공하면, 제어는 기능 블럭(1124)으로 보내진다.  $\text{Pred1}$ 이 최소 Rd를 제공하면, 제어는 기능 블럭(1124)으로 보내진다.  $\text{Pred1}$ 이 최소 Rd를 제공하면, 제어는 기능 블럭(1127)으로 보내진다.

[0140] 기능 블럭(1124)는  $\text{bi\_tmp\_flag}=0$ 를 세트하고, 제어를 기능 블럭(1130)으로 보낸다.

[0141] 기능 블럭(1130)은 나머지를 연산하고, 제어를 기능 블럭(1133)으로 보낸다. 기능 블럭(1133)은 예측에 대한 비트율 왜곡치, 즉  $\text{Rd}$ (예측)을 연산하고, 제어를 기능 블럭(1136)으로 보낸다. 기능 블럭(1136)은 모드 선택을 실행하고, 제어를 기능 블럭(1139)으로 보낸다. 기능 블럭(1139)은 움직임 백터와 그 외 구문을 부호화하고, 제어를 기능 블럭(1142)으로 보낸다. 기능 블럭(1142)는 나머지를 부호화하고, 제어를 종료 블럭(1199)으로 보낸다.

[0142] 기능 블럭(1127)은  $\text{bi\_tmp\_flag}=1$ 을 세트하고, 제어를 기능 블럭(1130)으로 보낸다.

[0143] 기능 블럭(1145)는 non-TMP 모드(들)을 이용하여 부호화하고, 제어를 기능 블럭(1136)으로 보낸다.

[0144] 도 12를 참조하면, 양방향 예측의 템플릿 매칭 예측을 이용한 다른 비디오 부호화 방법은 일반적으로 참조 부호 1200으로 나타낸다. 방법(1200)은 제어를 기능 블럭(1210)으로 보내는 시작 블럭(1205)을 포함한다. 기능 블럭(1210)은 구문을 파싱하고, 제어를 판정 블럭(1215)으로 보낸다. 판정 블럭(1215)은 현재 모드가 템플릿 매칭 예측(TMP) 모드인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1220)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1250)으로 보내진다.

[0145] 기능 블럭(1220)은 목록 0의  $P_0$  및 목록 1의  $P_1$ 을 연기 위해서 움직임 보상을 실행하고, 제어를 판정 블럭(1225)으로 보낸다. 판정 블럭(1225)은  $\text{bi\_tmp\_flag}$ 의 현재 값이 0 또는 1인지의 여부를 판정한다.  $\text{bi\_tmp\_flag}$ 의 현재 값이 0이면, 제어는 기능 블럭(1230)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1235)으로 보내진다.

[0146] 기능 블럭(1230)은  $\text{Pred}=(P_0+P_1)/2$ 를 연산하고, 제어를 기능 블럭(1245)으로 보낸다.

[0147] 기능 블럭(1245)는 나머지를 예측에 부가하여, 현재 블럭을 보상하고, 제어를 종료 블럭(1299)으로 보낸다.

[0148] 기능 블럭(1235)는 템플릿  $P_2+(P_0+P_1)/2$ 를 세트하고,  $P_3$ 을 연기 위해 두 참조 목록에서 템플릿 매칭을 실행하고 제어를 기능 블럭(1240)에 보낸다. 기능 블럭(1240)은  $\text{Pred}=(P_2+P_3)/2$ 를 연산하고, 제어를 기능 블럭(1245)으로 보낸다.

[0149] 기능 블럭(1250)은 non-TMP 모드(들)을 이용하여 복호화하고, 제어를 종료 블럭(1299)으로 보낸다.

[0150] 유사하게, 우리는 이 방법으로 하나 이상의 템플릿 매칭 예측을 얻을 수 있다. 또한, 다른 실시예에서, 우리는 다음 평균화로 최종 예측을 성취할 수 있다:  $(P_0+P_1+P_3)/3$ . 부가의 실시예는  $P_2$  평균화에 대해 N개의 가장 유사한 패치를 고려하는 것이며 최종 예측기는 평균일 수 있다: 다른 많은 확률 중에서  $(P_2+P_3+\dots+P_N)/(N-1)$  또는  $(P_0+P_1+P_3+\dots+P_N)/N$ .

[0151] 우리는 이 실시예에서, 템플릿 매칭 예측이 움직임 보상된 예측과 동일한 경우, 템플릿 매칭 예측을 더 이상을 부가할 필요가 없으며, 즉 모든 다수 가설 움직임 보상된 예측은 달라야 한다는 것에 유의해야 한다.

[0152] **실시예 3: 다수 참조 영상과 조합된 TMP**

[0153] MPEG-4 AVC 표준에서, 다수 참조 영상은 부호화 효율을 더욱 개선하기 위해 이용될 수 있다. 따라서, 상기 실시예 1과 실시예 2와 관련하여 설명되는 바와 같이 본 발명을 다른 참조 목록에 대해 제한할 필요가 없다. 우리는 하나의 참조 영상으로부터 템플릿을 성취한 다음에, 동일한 참조 영상이나 그 외 동일한 영상 목록에 있거나 있지 않은 참조 영상에서 템플릿 매칭 예측 검색을 실행할 수 있다. 이것은 하나의 참조 목록으로 본 발명을 구현할 수 있게 한다. 이는 또한 임의의 개수의 다수 가설 움직임 보상된 예측에 대해 본 발명을 구현할 수 있게 한다. 도 13을 참조하면, 다수의 참조 영상을 갖는 템플릿 매칭 예측의 일 예는 일반적으로 참조 부호 1300으로 나타낸다. 다수 참조 영상을 갖는 템플릿 매칭 예측(1300)은 현재 프레임(1310), 목록 0의 참조 프레임 1(1320) 및 목록 0의 참조 프레임 0(1330)을 수반한다. 현재 프레임(1310)은 타겟 블럭(1311)을 포함하고,

목록 0의 참조 프레임 1(1320)은 예측(1321)을 포함하고 (또한 "P<sub>1</sub>"로 지정), 및 목록 0의 참조 프레임 0(1330)은 예측(1331) (또한 "P<sub>0</sub>"으로 지정)을 포함한다. 움직임 벡터(1340)은 타겟 블럭(1311)과 예측(1331) 간의 움직임을 나타내기 위해 이용된다. 우리는 먼저 목록 0의 참조 프레임 0에서 P<sub>0</sub>을 얻기 위해 움직임 예측을 실행한다. 다음에 P<sub>1</sub>을 얻기 위해 참조 목록 0의 다른 참조 영상 (참조 프레임 1)에서 검색하기 위해 템플릿으로 P<sub>0</sub>을 이용한다.

[0154] 이 방법은 다수의 참조 영상이 이용될 때 MPEG-4 AVC 표준의 P 영상에서의 예측 모드와 쉽게 조합될 수 있다. 예를 들어, 이 방법은 모드 MxN 또는 스킵 모드에 이용될 수 있으며, 이 때 M 또는 N은 16, 8 및 4 중 어느 것 일 수 있다. 우리는 이 모드에 대해 템플릿 매칭 예측의 이용에 얼마나 많은 가설들이 실행되는지를 나타내기 위해 매크로블럭 레벨이나 서브매크로블럭 레벨의 플랙을 부가할 수 있다.

[0155] 1인 bi\_tmp\_flag은 단방향 예측과 템플릿 매칭 예측을 이용한 양방향 예측이 적용된 것을 특정한다. 0인 bi\_tmp\_flag는 단방향 예측이 적용된 것을 특정한다.

[0156] 일 실시예에 따른 16x16에 대한 일 예의 부호화 과정은 다음과 같이 나타낸다:

[0157] (1) P<sub>0</sub>을 얻기 위해 참조 목록 0에서 매크로블럭 16x16에 대한 움직임 추정을 실행하고 P<sub>0</sub>에 대한 RDcost를 연산 한다.

[0158] (2) P<sub>1</sub>을 얻기 위해 참조 목록 0의 다른 참조 영상에서 검색하도록 P<sub>0</sub>을 템플릿으로 이용하고, (P<sub>0</sub>+P<sub>1</sub>)/2에 대한 RDcost를 연산한다.

[0159] (3) (1)과 (2)로부터 RDcost를 연산하고 bi\_tmp\_flag를 세트한다. (2)의 RDcost가 (1)의 것보다 작으면, bi\_tmp\_flag=1로 세트하고, 그렇지 않으면 bi\_tmp\_flag=0으로 세트한다.

[0160] 일 실시예에 따른 16x16에 대한 대응 복호화 과정은 다음과 같이 설명된다:

[0161] (1) bi\_tmp\_flag의 구문 및 16x16에 대한 움직임 벡터를 파싱 및 복호화한다.

[0162] (2) P<sub>0</sub>을 얻기 위해 참조 목록 0에서 매크로블럭 16x16에 대한 움직임 보상을 실행한다.

[0163] (3) bi\_tmp\_flag이 0이면, 중지한다. 그렇지 않으면,

[0164] (4) 다음을 실행한다: P<sub>1</sub>을 얻기 위해 참조 목록 0의 다른 참조 영상에서 검색하도록 템플릿으로 P<sub>0</sub>을 이용하고, 0.5(P<sub>0</sub>+P<sub>1</sub>)으로 최종 예측을 얻는다.

[0165] 도 14를 참조하면, 다수의 참조 영상을 갖는 템플릿 매칭 예측을 이용한 비디오 부호화 방법은 일반적으로 참조 부호 1400으로 나타낸다. 방법(1400)은 판정 블럭(1410)으로 제어를 보내는 시작 블럭(1405)를 포함한다. 판정 블럭(1410)은 현재 모드가 TMP 모드인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1415)으로 전달된다. 그렇지 않으면, 제어는 기능 블럭(1470)으로 보내진다.

[0166] 기능 블럭(1415)는 예측 P<sub>0</sub> 및 움직임 벡터 Mv를 얻기 위해 참조 목록 0에서 움직임 추정을 실행하고, 제어를 기능 블럭(1420)으로 보낸다. 기능 블럭(1420)은 P<sub>1</sub>을 얻기 위해 P<sub>0</sub>에 기초하여 목록 0의 다른 참조 영상에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(1425)을 보낸다. 기능 블럭(1425)은 P=(P<sub>0</sub>+P<sub>1</sub>)/2를 연산하고, 제어를 판정 블럭(1430)으로 보낸다. 판정 블럭(1430)은 예측 P<sub>0</sub>의 비트율 왜곡치가 예측 P의 비트율 왜곡치 보다 낮은지, 즉 Rd(P<sub>0</sub>)<Rd(P)인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1435)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1465)으로 보내진다.

[0167] 기능 블럭(1435)은 bi\_tmp\_flag=0를 세트하고, 예측=P<sub>0</sub>를 세트하고, 제어를 기능 블럭(1440)으로 보낸다.

[0168] 기능 블럭(1440)은 나머지를 연산하고, 제어를 기능 블럭(1445)으로 보낸다. 기능 블럭(1445)는 RD(예측)을 연산하고, 제어를 기능 블럭(1450)으로 보낸다. 기능 블럭(1450)은 모드 선택을 실행하고, 제어를 기능 블럭(1455)으로 보낸다. 기능 블럭(1455)은 움직임 벡터 Mv 및 그 외 구문을 부호화하고, 제어를 기능 블럭(1460)으로 보낸다. 기능 블럭(1460)은 나머지를 부호화하고, 제어를 종료 블럭(1499)으로 보낸다.

- [0169] 기능 블럭(1465)은  $bi\_tmp\_flag=1$ 을 세트하고, 예측=P를 세트하고, 제어를 기능 블럭(1440)으로 보낸다.
- [0170] 기능 블럭(1470)은 non-TMP 모드(들)을 이용하여 부호화하고, 제어를 기능 블럭(1450)으로 보낸다.
- [0171] 도 15를 참조하며, 다수 참조 영상을 갖는 템플릿 매칭 예측을 이용한 비디오 복호화 방법은 일반적으로 참조 부호 1500으로 나타낸다. 이 방법 1500은 제어를 기능 블럭(1510)으로 보내는 시작 블럭(1505)을 포함한다. 기능 블럭(1510)은 구문을 파싱하고, 제어를 판정 블럭(1515)으로 보낸다. 판정 블럭(1515)는 현재 모드가 TMP 모드인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1520)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1555)으로 보내진다.
- [0172] 기능 블럭(1520)은 움직임 벡터  $Mv$ , 나머지, 및  $bi\_tmp\_flag$ 를 복호화하고, 제어를 기능 블럭(1525)으로 보낸다. 기능 블럭(1525)는 움직임 벡터  $Mv$ 로 참조 목록 0에서  $P_0$ 을 얻고, 제어를 판정 블럭(1530)으로 보낸다. 판정 블럭(1530)은  $bi\_tmp\_flag$ 가 1인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1535)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1550)으로 보내진다.
- [0173] 기능 블럭(1535)은  $P_1$ 을 얻기 위해  $P_0$ 에 기초하여 목록 0의 다른 참조 영상에서 템플릿 매칭을 실행하고, 제어를 기능 블럭(1540)으로 보낸다. 기능 블럭(1540)은 예측= $(P_0+P_1)/2$ 를 연산하고, 제어를 기능 블럭(1545)으로 보낸다.
- [0174] 기능 블럭(1545)은 나머지를 예측에 부가하여, 현재 블럭을 보상하고, 제어를 종료 블럭(1599)으로 보낸다.
- [0175] 기능 블럭(1550)은 예측= $P_0$ 을 연산하고, 제어를 기능 블럭(1545)으로 보낸다.
- [0176] 기능 블럭(1555)은 non-TMP 모드(들)을 이용하여 복호화하고, 제어를 종료 블럭(1599)으로 보낸다.
- [0177] **실시예 4: 인트라 영상 부호화를 위한 방향성 인트라 예측이나 변위된 인트라 예측과 조합된 TMP**
- [0178] 이 실시예에서, 인트라 영상 부호화를 위해, 우리는 MPEG-4 AVC 표준에서 방향성 인트라 예측을 이용하거나, 변위된 인트라 예측(DIP)을 이용하여 타겟 블럭에 템플릿을 채울 수 있다. 다음에, 우리는 템플릿을 현재 영상의 재구성된 화소로부터 검색한다. 다음에 템플릿 검색된 예측 및 방향성/변위 인트라 예측을 조합하여 다수의 가설 예측을 적용한다.
- [0179] 도 16을 참조하면, 인트라 예측을 위해 템플릿 매칭 예측을 이용한 다른 비디오 부호화 방법은 일반적으로 참조 부호 1600으로 나타낸다. 방법 1600은 판정 블럭(1610)으로 제어를 보내는 시작 블럭(1605)을 포함한다. 판정 블럭(1610)은 현재 모드가 TMP 모드인지의 여부를 판정한다. 그렇다면, 제어는 판정 블럭(1615)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1675)으로 보내진다.
- [0180] 판정 블럭(1615)은 방향성/변위된 인트라 예측(DIP)이 허용되는지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1620)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1680)으로 보내진다.
- [0181] 기능 블럭(1620)은  $P_0$ 을 얻기 위해 인트라 움직임 추정을 실행하고, 제어를 기능 블럭(1625)으로 보낸다.
- [0182] 기능 블럭(1625)은  $P_1$ 을 얻기 위해  $P_0$ 에 기초하여 인트라 템플릿 매칭을 실행하고 (즉, 현재 프레임의 재구성된 영역으로부터 템플릿을 검색하고), 제어를 기능 블럭(1630)으로 보낸다. 기능 블럭(1630)은  $P=(P_0+P_1)/2$ 를 연산하고, 제어를 판정 블럭(1635)으로 보낸다. 판정 블럭(1635)은 예측  $P_0$ 의 비트율 왜곡치가 예측  $P$ 의 비트율 왜곡치 보다 작은지, 즉  $Rd(P_0) < Rd(P)$ 인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1640)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1670)으로 보내진다.
- [0183] 기능 블럭(1640)은  $bi\_tmp\_flag=0$ 을 세트하고,  $Pred=P_0$ 을 세트하고, 제어를 기능 블럭(1645)으로 보낸다.
- [0184] 기능 블럭(1645)은 나머지를 연산하고, 제어를 기능 블럭(1650)으로 보낸다. 기능 블럭(1650)은  $Rd(Pred)$ 를 연산하고, 제어를 기능 블럭(1655)으로 보낸다. 기능 블럭(1655)는 모드 선택을 실행하고, 제어를 기능 블럭(1660)으로 보낸다. 기능 블럭(1660)은 예측 모드, 움직임 벡터 및 그 외 구문을 부호화하고, 제어를 기능 블럭(1665)으로 보낸다. 기능 블럭(1665)은 나머지를 부호화하고, 제어를 종료 블럭(1699)으로 보낸다.
- [0185] 기능 블럭(1670)은  $bi\_tmp\_flag=1$ 을 세트하고,  $Pred=P$ 을 세트하고, 제어를 기능 블럭(1645)으로 보낸다.

- [0186] 기능 블럭(1675)는 non-TMP 모드(들)을 이용하여 부호화하고, 제어를 기능 블럭(1655)으로 보낸다.
- [0187] 도 17을 참조하여, 다수의 참조 영상을 갖는 템플릿 매칭 예측을 이용한 다른 비디오 복호화 방법은 참조 부호 1700으로 나타낸다. 이 방법 1700은 제어를 기능 블럭(1710)으로 보내는 시작 블럭(1705)를 포함한다. 기능 블럭(1710)은 구문을 파싱하고, 제어를 판정 블럭(1715)으로 보낸다. 판정 블럭(1715)는 현재 모드가 TMP 모드 인지의 여부를 판정한다. 그렇다면, 제어는 판정 블럭(1720)으로 보내진다. 그렇지 않으면 제어는 기능 블럭(1760)으로 보내진다.
- [0188] 판정 블럭(1720)은 방향성/변위 인트라 예측(DIP)이 가능한지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1725)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1750)으로 보내진다.
- [0189] 기능 블럭(1725)는  $P_0$ 을 얻기 위해 인트라 움직임 보상을 실행하고, 제어를 판정 블럭(1730)으로 보낸다.
- [0190] 판정 블럭(1730)은  $bi\_tmp\_flag=1$ 인지의 여부를 판정한다. 그렇다면, 제어는 기능 블럭(1735)으로 보내진다. 그렇지 않으면, 제어는 기능 블럭(1755)으로 보내진다.
- [0191] 기능 블럭(1735)는  $P_1$ 을 얻기 위해  $P_0$ 에 기초하여 인트라 템플릿 매칭을 실행하고, 제어를 기능 블럭(1740)으로 보낸다. 기능 블럭(1740)은  $P=(P_0+P_1)/2$ 를 연산하고, 제어를 기능 블럭(1745)으로 보낸다.
- [0192] 기능 블럭(1745)은 나머지를  $P$ 에 부가하여 현재 블럭을 보상하고, 제어를 종료 블럭(1799)으로 보낸다.
- [0193] 기능 블럭(1750)은  $P_0$ 을 얻기 위해 인트라 예측을 실행하고 제어를 판정 블럭(1730)으로 보낸다.
- [0194] 기능 블럭(1765)은  $P=P_0$ 을 세트하고, 제어를 기능 블럭(1745)으로 보낸다.
- [0195] 기능 블럭(1760)은 non-TMP 인트라 모드(들)를 이용하여 복호화하고, 제어를 종료 블럭(1799)으로 보낸다.
- [0196] **템플릿 검색 간략화**
- [0197] 본 원리의 실시예에 따르면, 템플릿 검색은 부호화기 및 복호화기 둘 다에서 실행된다. 그러나, 이 방법의 연산 복잡성은 매우 상당하다. 복잡성을 줄이기 위해서, 우리는 검색 알고리즘을 간략화하기 위해 몇 가지 방법을 설명 및 제안한다. 이 중 한 실시예로, 우리는 예를 들어, 다이아몬드 검색, 알고리즘 검색 및 인핸스드 프리딕티브 조널 서치 (enhance predictive zonal search (EPZS)) 등과 같은 고속 검속 알고리즘을 이용한다. 두 번째 실시예로, 움직임이 선형이라고 가정하면, 우리는 템플릿이 이용했던 움직임 벡터로부터 템플릿 검색 예측 기를 유도하고, 참조 영상으로부터 타겟 영상으로의 거리 및 방향에 따라 템플릿 검색 예측기를 스케일(scale) 할 수 있다. 다음에 이 예측기를 중심으로 이용하여 소량의 정화를 행한다.
- [0198] **다른 부호화 블럭에 미치는 영향**
- [0199] **TMP에 대한 움직임 벡터**
- [0200] MPEG-4 AVC 표준에서, 현재 블럭에 대한 움직임 벡터는 인접 블럭으로부터 예측된다. 따라서, 현재 블럭의 움직임 벡터의 값은 장차 인접하는 블럭에 영향을 주게 된다. 이는 명시적 움직임 벡터와 템플릿 검색된 움직임 벡터를 갖기 때문에, 어느 움직임 벡터를 이용해야 하는지에 대해서 템플릿 검색 블럭에 관한 질문을 야기한다. 제1 실시예에서, 템플릿 검색이 행해진 후에, 템플릿에 대한 명시적 움직임 벡터를 검색할 수 있다. 이 움직임 벡터는 장차 움직임 벡터 예측에 이용된다. 제2 실시예에서, 우리는 움직임 벡터가 0과 같은 몇 예측기, 또는 현재 블럭의 중간 예측기라고 가정한다.
- [0201] **디블로킹 필터**
- [0202] 디블로킹 필터에 대해, 제1 실시예에서, 우리는 템플릿 매칭 예측 블럭이 명시적 움직임 추정 블럭과 동일한 것으로 처리하고, 상술한 바와 같이 템플릿 매칭 예측을 위해 움직임 벡터를 이용할 수 있다. 다음에, 디블로킹 프로세스는 변경되지 않는다. 제2 실시예에서, 템플릿 매칭 예측은 명시적 움직임 추정 블럭과는 다른 특성을 갖기 때문에, 우리는 블럭 강도, 필터 유형 및/또는 이에 따라 필터 길이를 조정할 수 있다.
- [0203] **독립적 파싱**
- [0204] 파싱을 복호화 프로세스와 분리하기 위해서, 우리는 파싱 시 템플릿 검색을 실행할 수 없다. 따라서, 파싱 프로세스에서, 템플릿 매칭 예측을 위한 움직임 벡터 편차를 위해, (상술한 바와 같이) TMP에 대한 움직임 벡터의 실시예 1이 이용되면, 우리는 그 움직임 벡터 예측기에 의해 제2 움직임 벡터를 세트하는 것을 수반하는 파싱을

위해 실시예 2를 이용하고, 실제 복호화를 위해 실시예 1을 이용할 수 있다.

- [0205] 표 1은 본 원리의 실시예에 따른, 예시의 슬라이스 헤더 구문을 나타낸다. 표 2는 본 원리의 실시예에 따른 예시의 매크로블럭 총 구문을 나타낸다.

豆 1

slice_header(){	C	디스크립터
first_mb_in_slice	2	ue(v)
slice_type	2	ue(v)
pic_parameter_set_id	2	ue(v)
...		
if (slice_type!=1)		
tmp_enable_flag	2	u(1)
...		

五 2

macroblock_layer(){ mb_type if(MbBPartPredMode(mb_type,0)!=intra_4x4 && MbBPartPredMode(mb_type,0)!=intra_8x8 && MbBPartPredMode(mb_type,0)!=intra_16x16 && bi_tmp_flag ...}	2	ue(v) ae(v) u(1)
--	---	------------------------

- [0208] 표 1 및 표 2의 구문 요소들 중 약간의 의미를 이하 다음과 같이 더욱 상세히 설명한다.

[0209] 1인 tmp\_enable\_flag는 템플릿 매칭 예측이 슬라이스에 대해 가능한 것을 특정한다. 0인 tmp\_enable\_flag는 템플릿 매칭 예측이 슬라이스에 대해 가능하지 않은 것을 특정한다.

[0210] 0인 bi\_tmp\_flag는 템플릿 매칭 예측이 매크로블럭에 가능하지 않은 것을 특정한다. 1인 bi\_tmp\_flag는 템플릿 매칭 예측이 매크로블럭에 가능한 것을 특정한다. bi\_tmp\_flag가 1일 때, 단방향 예측 및 템플릿 매칭 예측이 단일 예측 모드에 적용되고, 최대 (4 까지) 다른 움직임 보상된 예측을 위한 다수 가설 움직임 보상 예측이 양방향 예측 모드에 적용된다.

[0211] 본 발명의 많은 부수적인 장점/특성들에 대해 이하 설명하며, 이를 중 약간은 상술한 바 있다. 예를 들어, 일장점/특성은 타겟 블럭에 대한 적어도 하나의 예측기를 결정하고 타겟 블럭에 대한 적어도 하나의 부가의 예측기를 검색하기 위해 템플릿으로 적어도 하나의 예측기를 각각 이용하는 것을 영상의 타겟 블럭을 부호화하기 위한 부호화기를 갖는 장치이다.

[0212] 다른 장치/특성은 적어도 하나의 예측기는 인트라 예측기, 인터 예측기, 또는 그 조합인, 상술한 바와 같은 부호화기를 갖는 장치이다.

[0213] 또 다른 장치/특성은 적어도 하나의 예측기는 인트라 예측기, 인터 예측기, 또는 그 조합이고, 적어도 하나의 예측기에 대한 부가 정보는 암시적 부호화 및 명시적 부호화된 것 중 적어도 하나인, 상술한 바와 같은 부호화기를 갖는 장치이다.

[0214] 또 다른 장치/특성은 적어도 하나의 부가의 예측기는 템플릿 매칭 예측 프로세스 및 다수 가설 움직임 보상 예측 프로세스를 이용하여 취득되는, 상술한 바와 같은 부호화기를 갖는 장치이다.

[0215] 더욱, 다른 장점/특성은 적어도 하나의 부가의 예측기는 템플릿 매칭 예측 프로세스 및 다수 가설 움직임 보상 예측 프로세스를 이용하여 취득되고, 명시적 움직임 추정 프로세스는 상기 적어도 하나의 예측기를 결정하는 데에 이용되고 상기 템플릿 매칭 예측 프로세스는 다수의 참조 목록과 관련하여 상기 적어도 하나의 부가의 예측기를 검색하는 데에 이용되는, 상술한 바와 같은 부호화기를 갖는 장치이다.

[0216] 또한, 다른 장점/특성은, 명시적 움직임 추정 프로세스 및 템플릿 매칭 예측 프로세스가 상술한 바와 같이 이용

되고, 명시적 움직임 추정 프로세스는 상기 다수의 참조 목록 중 특정의 것으로 제한되고 상기 템플릿 매칭 예측 프로세스는 상기 다수의 참조 목록 중 다른 것으로 제한되는, 부호화기를 갖는 장치이다.

[0217] 또한, 다른 장점/특성은 명시적 움직임 추정 프로세스 및 템플릿 매칭 예측 프로세스가 상술한 바와 같이 이용되고, 적어도 하나의 예측기는 상기 다수의 참조 목록 중에서 제1 참조 목록과 제2 참조 목록과 관련하여 상기 명시적 움직임 추정 프로세스와 암시적 움직임 추정 프로세스를 각각 적용하여 취득된 제1 예측기 및 제2 예측기를 포함하고, 상기 적어도 하나의 부가의 예측기는 상기 제2 참조 목록과 상기 제1 참조 목록에 관련하여 상기 템플릿 매칭 예측 프로세스에서 상기 제1 예측기와 상기 제2 예측기를 각각 이용하여 취득된 제1 부가 예측기 및 제2 부가 예측기를 포함하는, 부호화기를 갖는 장치이다.

[0218] 부가하여, 다른 장점/특성은 상술한 바와 같이 적어도 하나의 예측기는 제1 예측기 및 제2 예측기를 포함하고 적어도 하나의 부가의 예측기는 제1 부가의 예측기 및 제2 부가의 예측기를 포함하고, 제1 예측기 및 제2 부가 예측기는 다른 참조 영상에 놓이는, 부호화기를 갖는 장치이다.

[0219] 더구나, 다른 장점/특성은 명시적 움직임 추정 프로세스와 템플릿 매칭 예측 프로세스가 상술한 바와 같이 이용되고, 명시적 움직임 추정 프로세스는 상기 다수의 참조 목록 중 특정한 것에 대응하는 특정한 하나의 참조 영상에서 이용되고, 템플릿 매칭 예측 프로세스는 상기 특정한 하나의 참조 영상 또는 상기 다수의 참조 목록 중 다른 것에 대응하는 다른 참조 영상에서 이용되는, 부호화기를 갖는 장치이다.

[0220] 또한, 다른 장점/특성은 상술한 바와 같이 적어도 하나의 부가의 예측기는 템플릿 매칭 예측 프로세스 및 다수 가설 움직임 보상 예측 프로세스를 이용하여 취득되고, 적어도 하나의 예측은 움직임 벡터 예측기를 이용하여 유도되거나 추정되는, 부호화기를 갖는 장치이다.

[0221] 본 원리의 이들 및 그 외 특성 및 장점들은 여기 개시된 것에 비추어 관련 기술의 당업자에 의해서 쉽게 인식될 것이다. 본 원리의 개시는 하드웨어, 소프트웨어, 펌웨어, 특수 목적의 프로세서, 또는 이들의 조합의 여러 형태로 구현될 수 있다는 것이 이해되어야 한다.

[0222] 가장 바람직하게, 본 원리의 개시는 하드웨어와 소프트웨어의 조합으로 구현된다. 더구나, 소프트웨어는 프로그램 저장 유닛에 실체적으로 구현된 응용 프로그램으로 구현될 수 있다. 이 응용 프로그램은 임의의 적합한 아키텍처를 포함하는 머신에 업로드되거나 이에 의해 실행될 수 있다. 바람직하게, 이 머신은 하나 이상의 중앙 처리 장치 ("CPU"), 랜덤 액세스 메모리 ("RAM") 및 입/출력 ("I/O") 인터페이스와 같은 하드웨어를 갖는 컴퓨터 플랫폼에 구현된다. 컴퓨터 플랫폼은 또한 운영 시스템과 마이크로인스트럭션 코드를 포함할 수 있다. 여기 개시된 여러 프로세스와 기능들은 CPU에 의해 실행될 수 있는, 마이크로인스트럭션 코드의 일부나 응용 프로그램의 일부이거나, 이의 조합일 수 있다. 부가하여, 여러 다른 주변 장치들은 부가의 데이터 저장 유닛과 프린팅 유닛과 같은 컴퓨터 플랫폼에 연결될 수 있다.

[0223] 첨부한 도면에서 설명된 구성 시스템 성분과 방법들 중 몇은 소프트웨어로 바람직하게 구현되기 때문에, 시스템 구성 요소나 프로세스 기능 블럭 간의 실제 접속은 본 원리가 프로그램되는 방식에 따라 달라질 수 있다는 것이 이해되어야 한다. 여기 개시된 바에 의하면, 관련 기술의 당업자라면 본 원리의 이들 및 그 유사한 구현이나 구성을 생각해낼 수 있을 것이다.

[0224] 설명된 실시예는 첨부한 도면을 참조하여 설명되었지만, 본 원리는 이들 명확한 실시예에만 제한되는 것은 아니며, 여러 변경과 수정들이 본 원리의 영역이나 정신에서 벗어나지 않고 관련 기술의 당업자라면 실시할 수 있다는 것이 이해되어야 한다. 이런 변경 및 수정들은 첨부한 청구범위에 기재된 바와 같은 본 원리의 영역 내에 포함되는 것이다.

## 부호의 설명

[0225] 405: 복호화기 제어기

410: 입력 버퍼

444: 템플릿 검색기

445: 엔트로피 복호화기

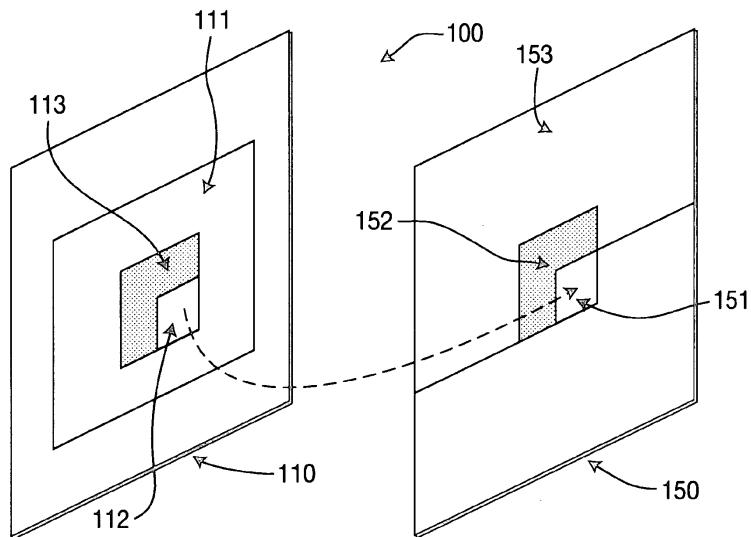
460: 인트라 예측 모듈

465: 디블로킹 필터

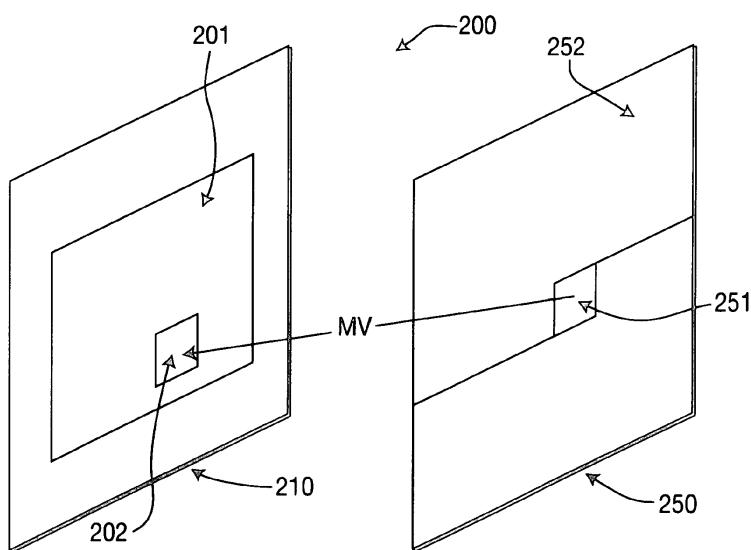
480: 참조 영상 베파

## 도면

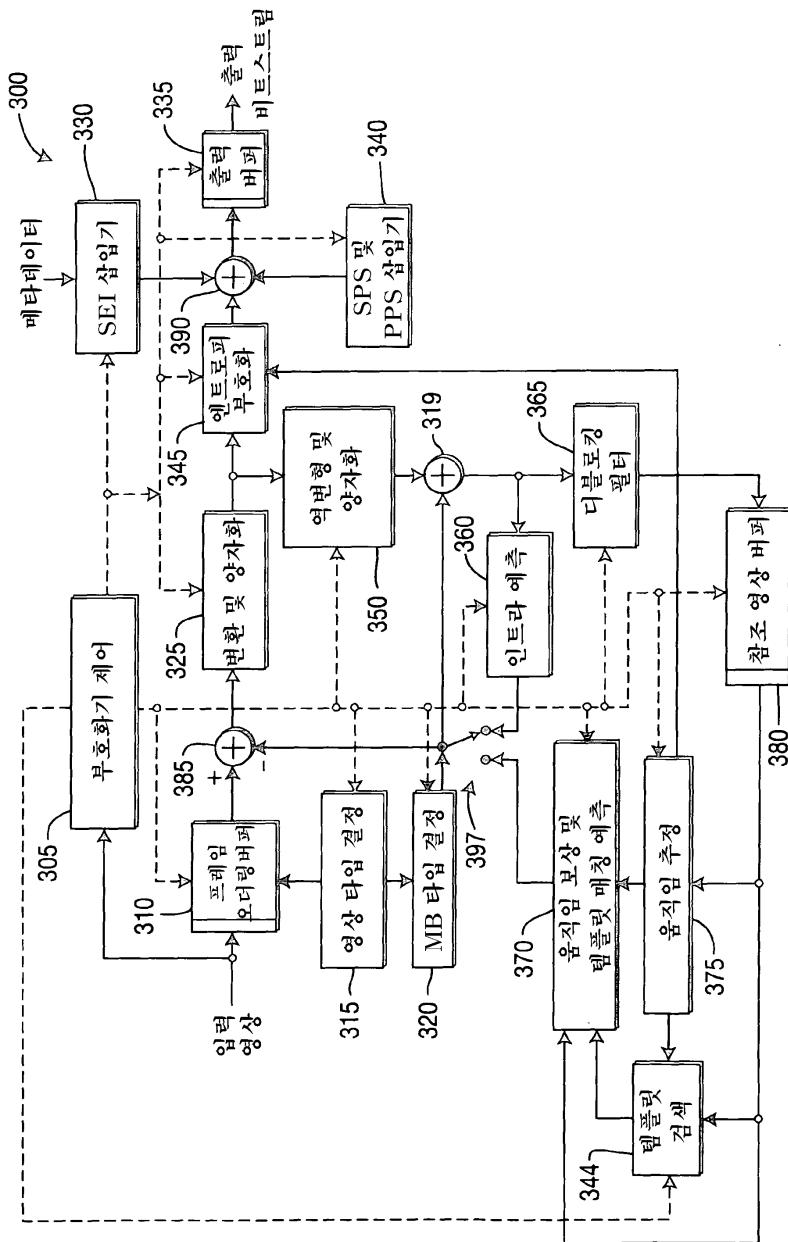
## 도면1



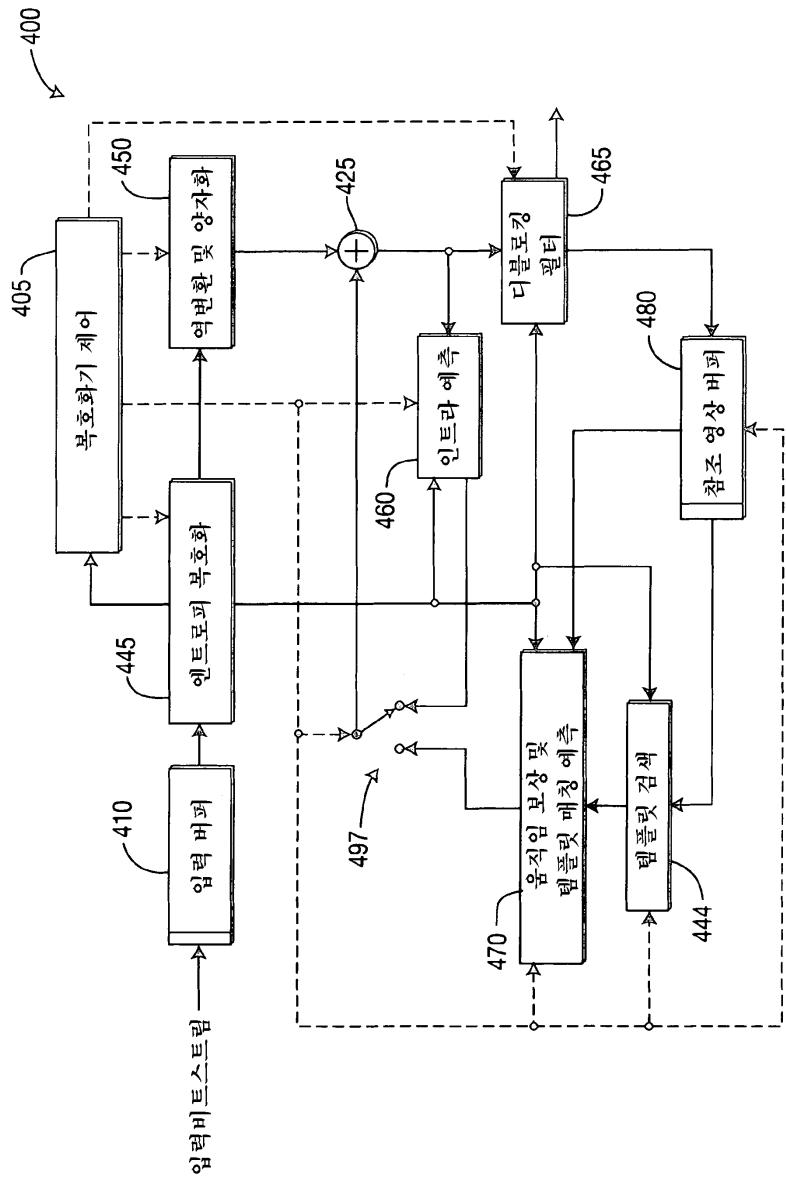
## 도면2



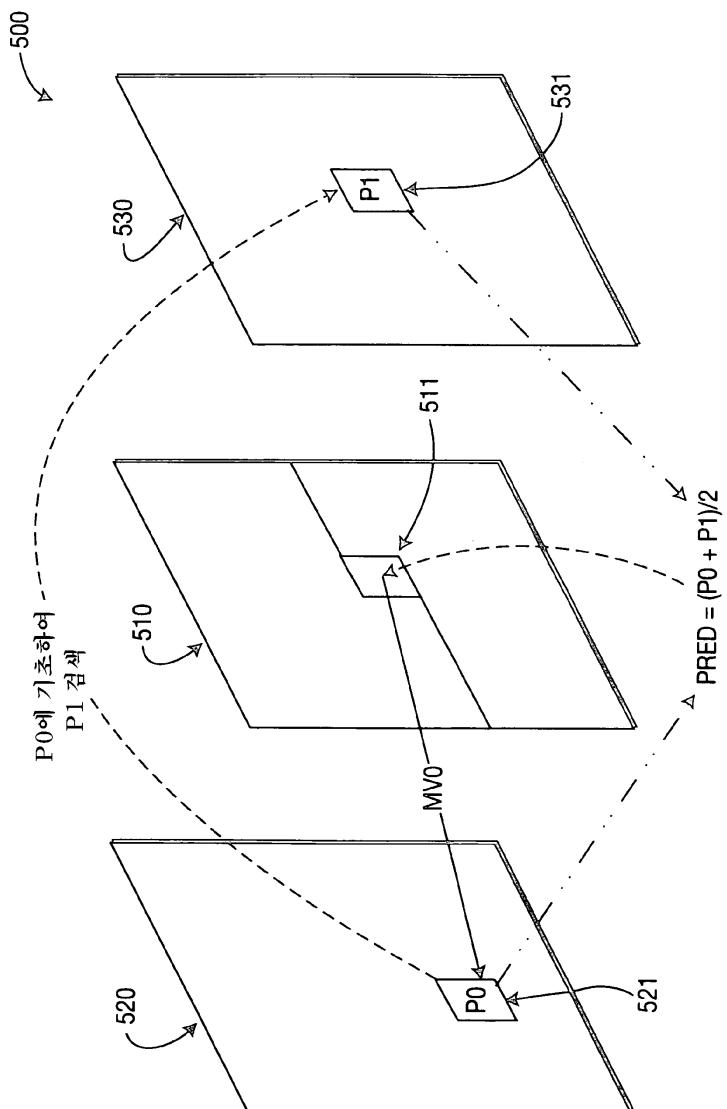
도면3



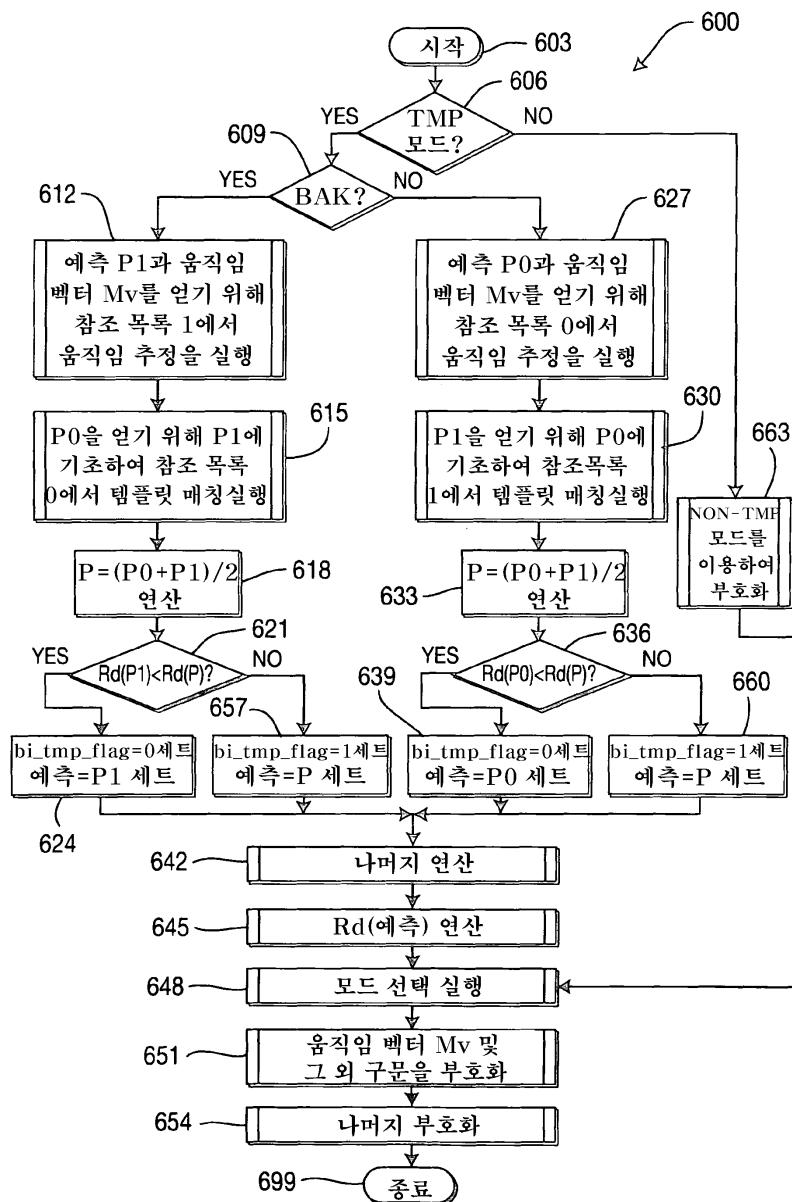
## 도면4



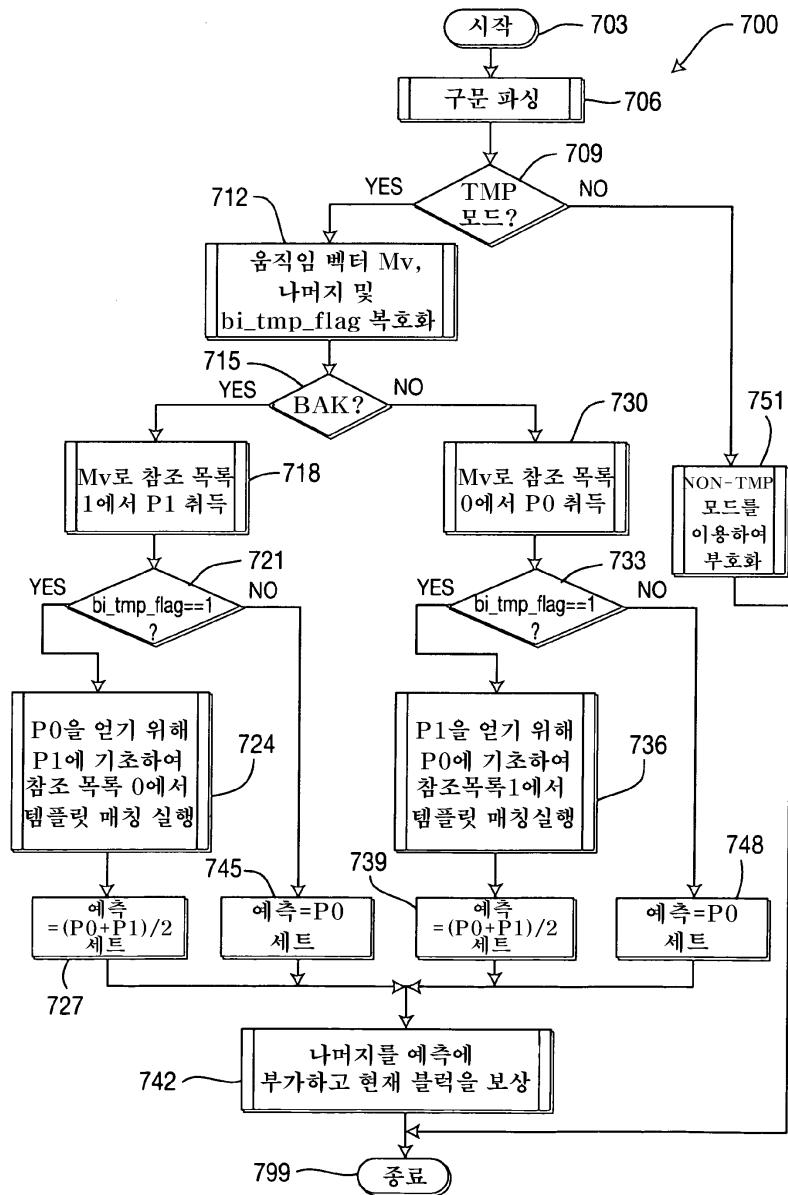
도면5



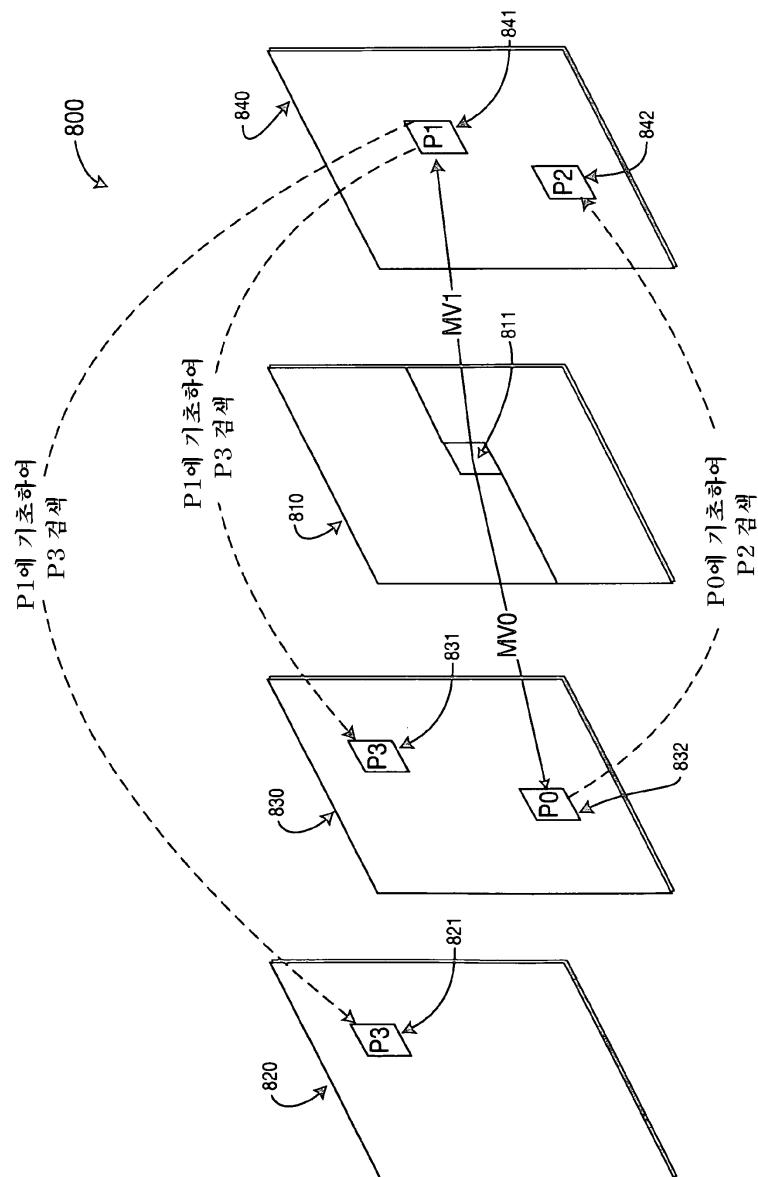
## 도면6



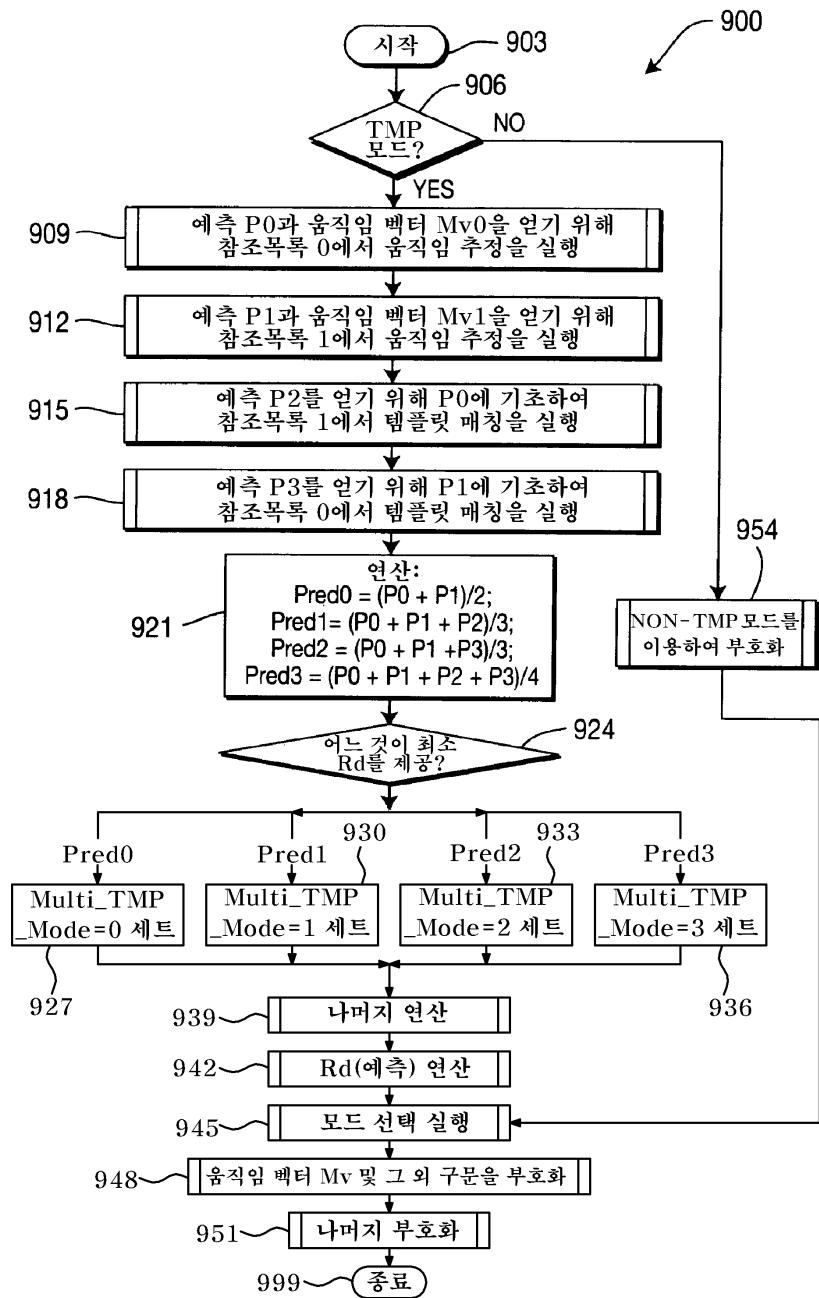
## 도면7



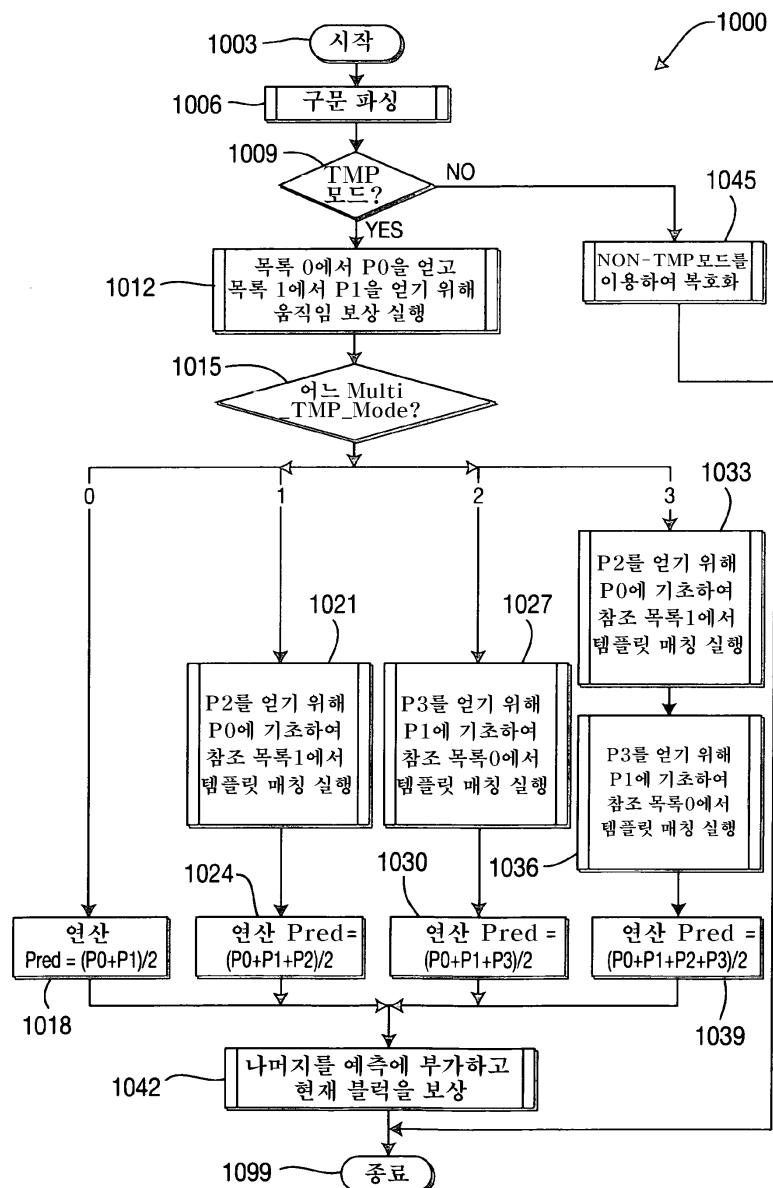
도면8



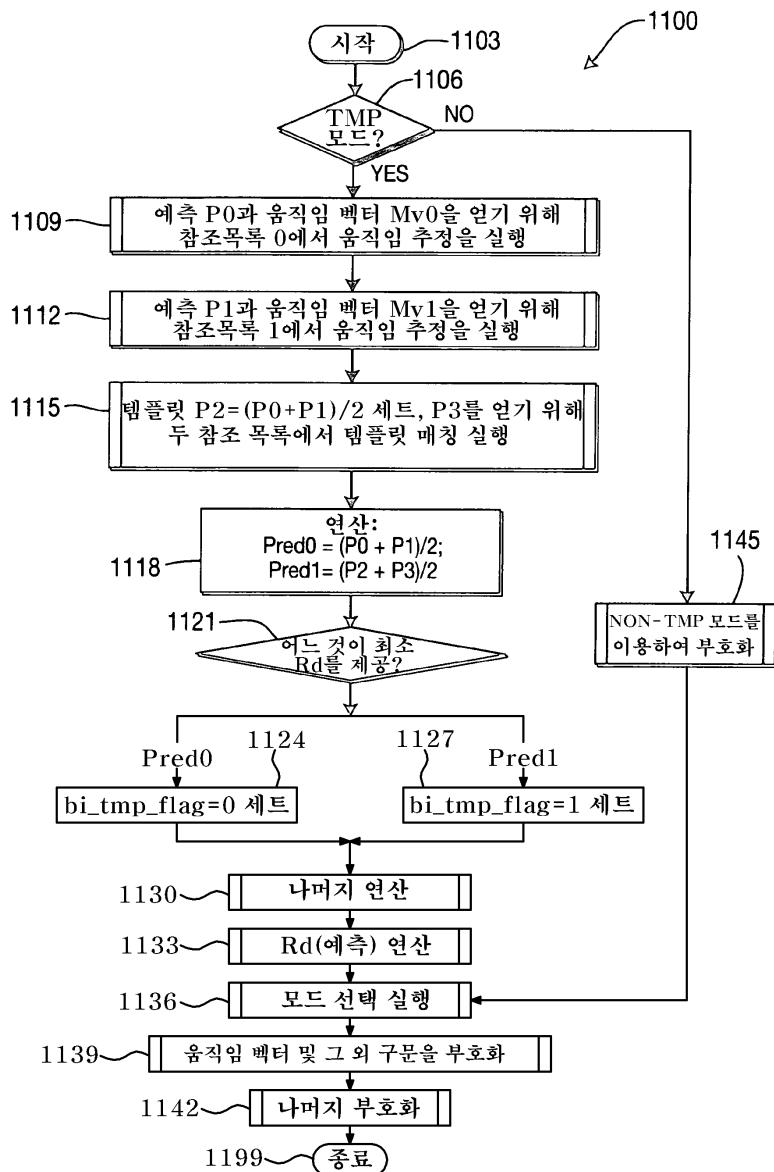
## 도면9



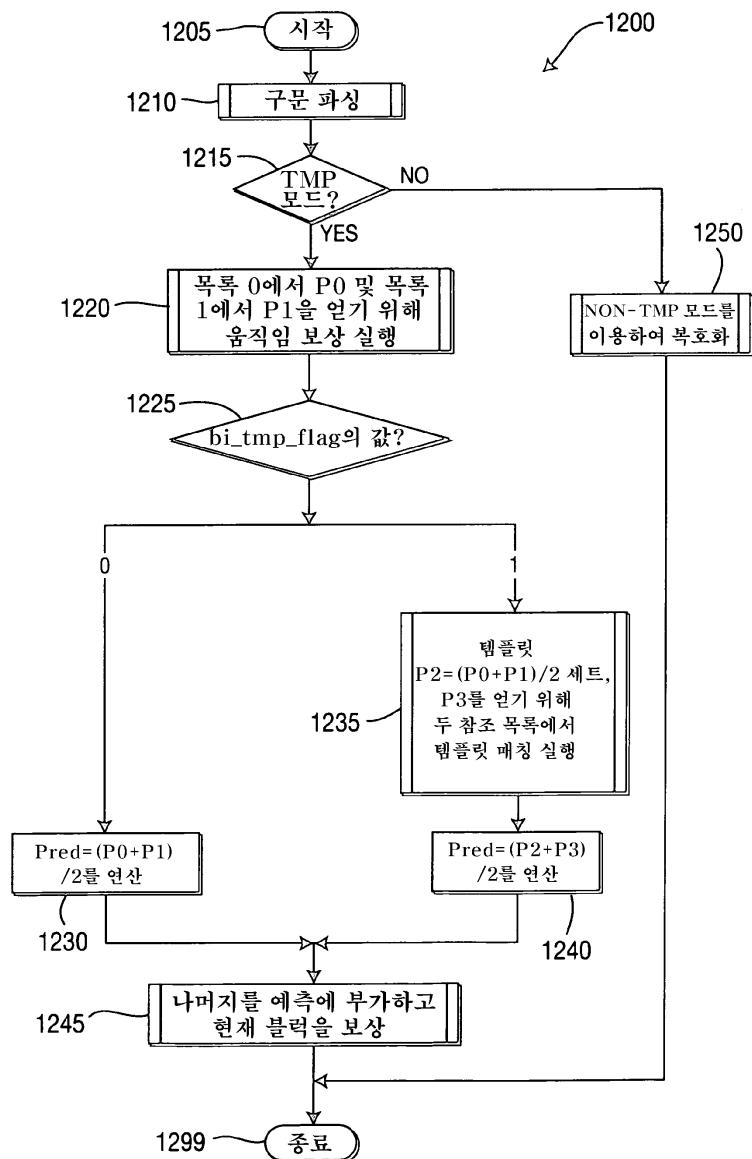
## 도면10



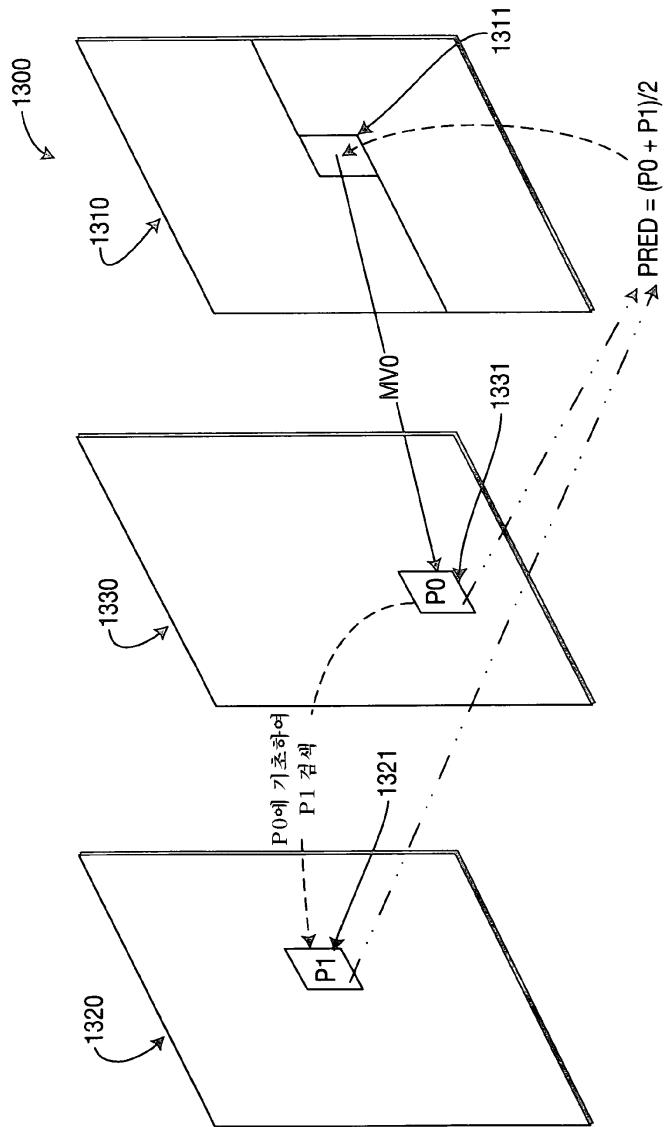
## 도면11



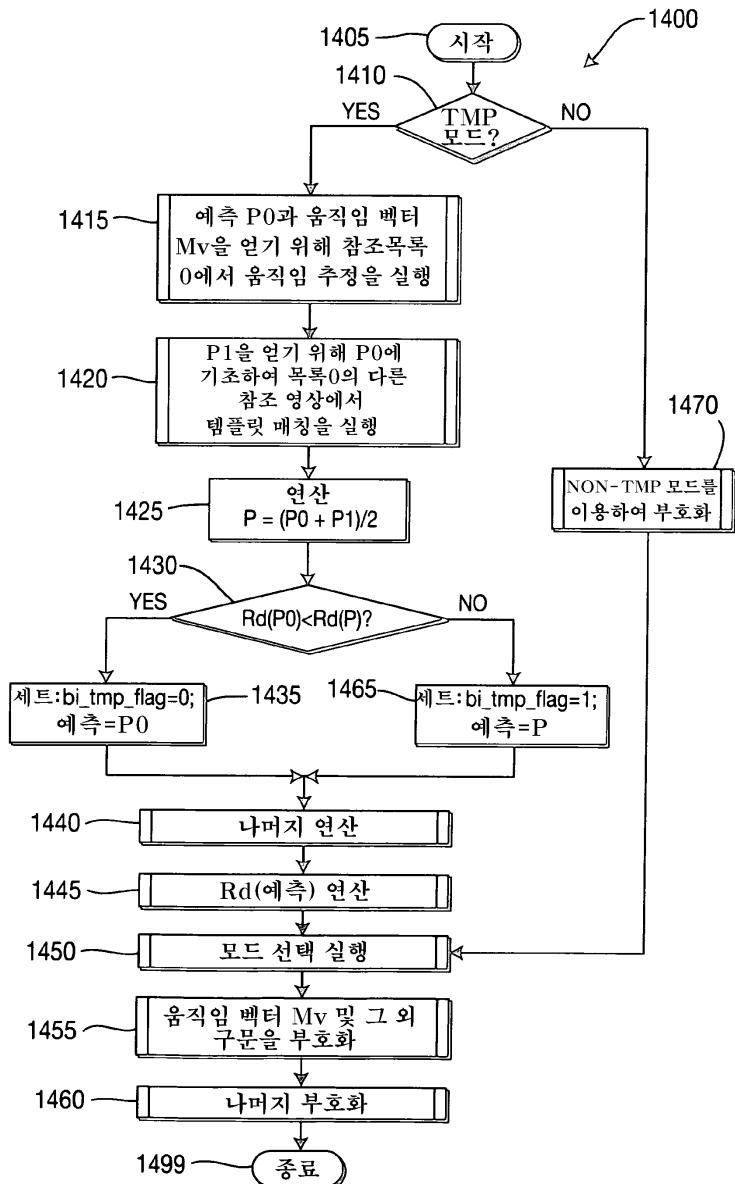
## 도면12



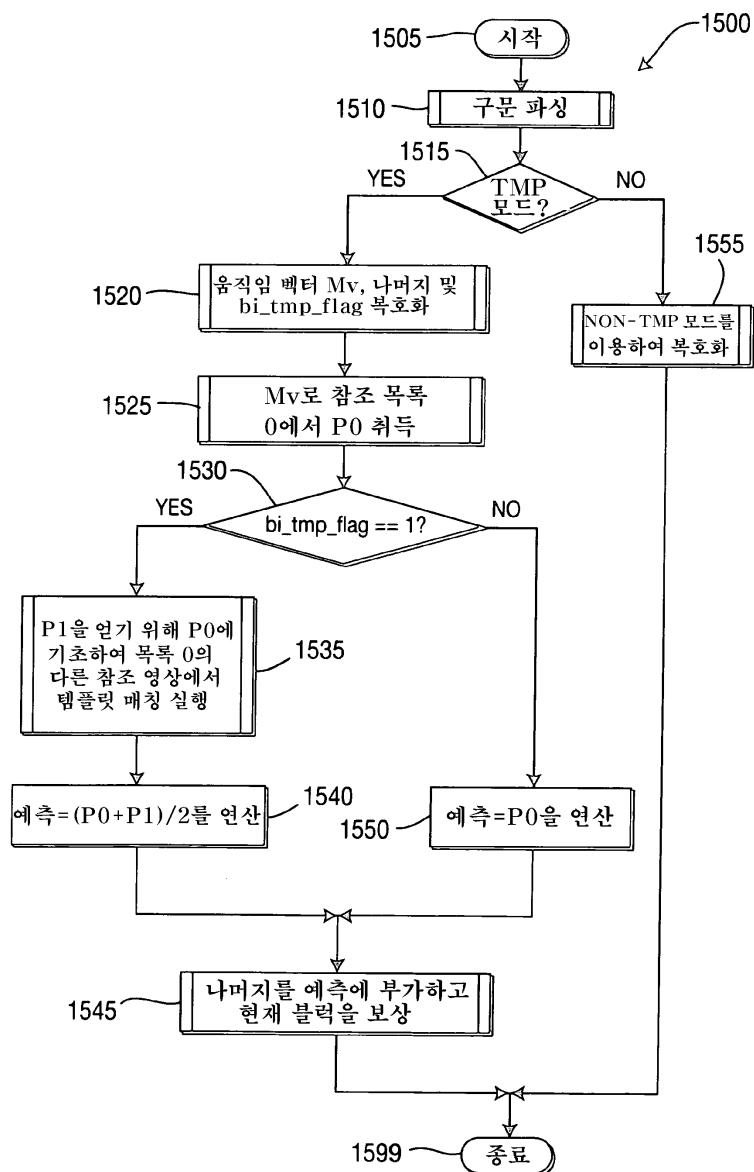
도면13



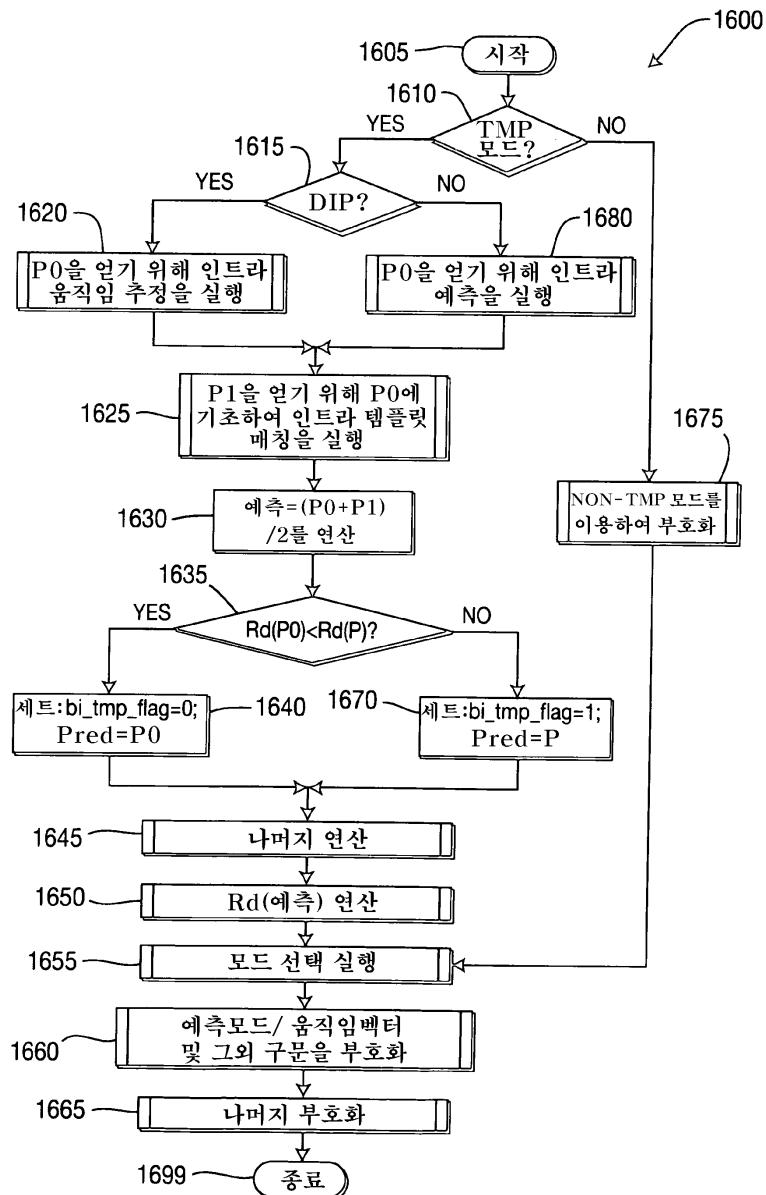
## 도면14



## 도면15



## 도면16



## 도면17

