



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0006510  
(43) 공개일자 2022년01월17일

- (51) 국제특허분류(Int. Cl.)  
G10L 19/032 (2013.01) G10L 19/025 (2013.01)  
G10L 25/21 (2013.01) G10L 25/51 (2013.01)  
G10L 25/93 (2013.01)
- (52) CPC특허분류  
G10L 19/032 (2013.01)  
G10L 19/025 (2013.01)
- (21) 출원번호 10-2021-7034717
- (22) 출원일자(국제) 2020년05월01일  
심사청구일자 없음
- (85) 번역문제출일자 2021년10월26일
- (86) 국제출원번호 PCT/CA2020/050582
- (87) 국제공개번호 WO 2020/223797  
국제공개일자 2020년11월12일
- (30) 우선권주장  
62/844,225 2019년05월07일 미국(US)

- (71) 출원인  
보이세지 코포레이션  
캐나다 퀘벡 에이치3알 2에이치6 빌레 몬트-로알  
케민 루션 750 스위트 250
- (72) 발명자  
엑슬러 바츨라브  
홀라브니 145/110, 624 00 브르노 체코
- (74) 대리인  
신성특허법인(유한)

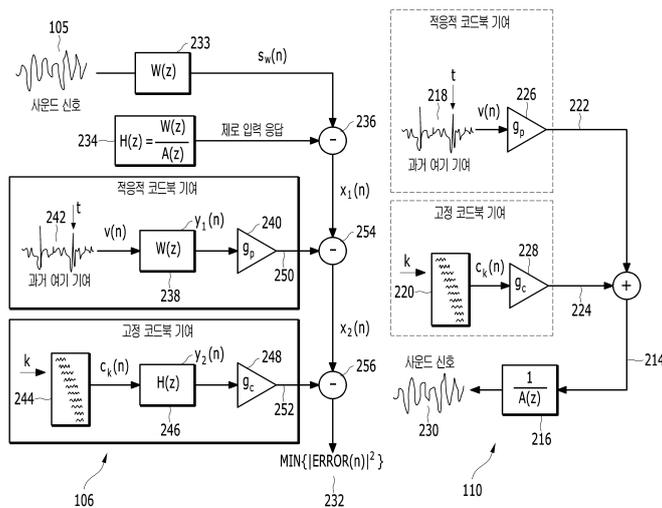
전체 청구항 수 : 총 40 항

(54) 발명의 명칭 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들

(57) 요약

코딩될 사운드 신호에 있어서의 어택을 검출하는 방법 및 디바이스로서, 사운드 신호는 다수의 서브-프레임들을 각각으로 포함하는 연속하는 프레임들로 프로세싱된다. 디바이스는, 현재 프레임의 마지막 서브-프레임에 있어서의 어택을 검출하는 제 1 단계 어택 검출기; 및 마지막 서브-프레임보다 선행하는 서브-프레임들을 포함하는, 현재 프레임의 서브-프레임들중 하나에 있어서의 어택을 검출하는 제 2 단계 어택 검출기를 구비한다. 현재 프레임이 체네틱 코딩 모드를 이용하여 코딩되는 것으로 이전에 분류된 활성 프레임이 아니면 검출되는 어택은 없다. 사운드 신호에 있어서의 어택을 코딩하는 방법 및 디바이스가 제공된다. 코딩 디바이스는, 상술한 어택 검출 디바이스와, 성문음 임펄스 형상들로 채워진 성문음-형상 코드북을 이용하는 전이 코딩 모드를 이용하여 검출된 어택을 구비한 서브-프레임을 인코딩하는 인코더를 구비한다.

대표도



(52) CPC특허분류

*G10L 25/21* (2013.01)

*G10L 25/51* (2013.01)

*G10L 25/93* (2013.01)

*G10L 2025/935* (2013.01)

---

## 명세서

### 청구범위

#### 청구항 1

코딩될 사운드 신호(sound signal)에 있어서의 어택(attack)을 검출하는 디바이스로서,  
 사운드 신호는 다수의 서브-프레임들을 각각으로 포함하는 연속하는 프레임들로 프로세싱되고,  
 상기 디바이스는,  
 현재 프레임의 마지막 서브-프레임에 있어서의 어택을 검출하는 제 1 단계 어택 검출기; 및  
 마지막 서브-프레임보다 선행하는 서브-프레임들을 포함하는, 현재 프레임의 서브-프레임들중 하나에 있어서의 어택을 검출하는 제 2 단계 어택 검출기를 구비하는  
 어택 검출 디바이스.

#### 청구항 2

제 1 항에 있어서,  
 현재 프레임이 제네릭 코딩 모드(generic coding mode)를 이용하여 코딩되도록 이전에 분류된 활성 프레임(active frame)인지를 판정하고, 현재 프레임이 제네릭 코딩 모드를 이용하여 코딩되도록 이전에 분류된 활성 프레임으로서 판정되지 않으면, 검출되는 어택이 없음을 나타내는 결정 모듈을 구비하는  
 어택 검출 디바이스.

#### 청구항 3

제 1 항 또는 제 2 항에 있어서,  
 현재 프레임의 다수의 분석 세그먼트들(analysis segments)에 있어서의 사운드 신호의 에너지의 계산기; 및  
 제 1 단계 및 제 2 단계 어택 검출기들에 의해 인증될 후보 어택 위치를 나타내는 최대 에너지를 가진 분석 세그먼트들 중의 한 세그먼트의 탐지기(finder)를 구비하는  
 어택 검출 디바이스.

#### 청구항 4

제 3 항에 있어서,  
 제 1 단계 어택 검출기는,  
 현재 프레임에 있어서의 마지막 서브-프레임 전의 분석 세그먼트들에 걸쳐서의 제 1 평균 에너지의 계산기; 및  
 최대 에너지를 가진 분석 세그먼트에서부터 현재 프레임의 마지막 분석 세그먼트까지의, 현재 프레임의 분석 세그먼트들에 걸쳐서의 제 2 평균 에너지의 계산기를 구비하는  
 어택 검출 디바이스.

#### 청구항 5

제 4 항에 있어서,

제 1 단계 어택 검출기는,

제 1 평균 에너지와 제 2 평균 에너지간의 비율을, 제 1 임계치와 비교하거나, 이전 프레임의 분류가 유성음(VOICED)이면, 제 1 평균 에너지와 제 2 평균 에너지간의 비율을 제 2 임계치와 비교하는 제 1 비교기를 구비하는

어택 검출 디바이스.

#### 청구항 6

제 5 항에 있어서,

제 1 단계 어택 검출기는,

제 1 비교기에 의한 비교가, 제 1 단계 어택이 검출됨을 나타내면,

최대 에너지의 분석 세그먼트의 에너지와 현재 프레임의 다른 분석 세그먼트들의 에너지간의 비율을, 제 3 임계치와 비교하는 제 2 비교기를 구비하는

어택 검출 디바이스.

#### 청구항 7

제 6 항에 있어서,

제 1 및 제 2 비교기에 의한 비교가, 제 1 단계 어택 위치가 후보 어택 위치를 나타내는 최대 에너지를 가진 분석 세그먼트임을 나타내면,

제 1 단계 어택 위치가 현재 프레임의 마지막 서브-프레임 전의 다수의 분석 세그먼트들과 동일하거나 그보다 더 큰지를 판정하고, 제 1 단계 어택 위치가 마지막 서브-프레임 전의 다수의 분석 세그먼트들과 동일하거나 그보다 더 큰 경우, 검출된 어택 위치를, 현재 프레임의 마지막 서브-프레임에 있어서의 제 1 단계 어택 위치로서 판정하는 결정 모듈을 구비하는

어택 검출 디바이스.

#### 청구항 8

제 1 항 내지 제 7 항 중 어느 한 항에 있어서,

제 2 단계 어택 검출기는, 제 1 단계 어택 검출기에 의해 검출되는 어택이 없을 때 이용되는

어택 검출 디바이스.

#### 청구항 9

제 8 항에 있어서,

현재 프레임이 유성음(VOICED)으로서 분류되는지를 판정하는 결정 모듈을 구비하고,

현재 프레임이 유성음으로서 분류되지 않으면, 제 2 단계 어택 검출기가 이용되는

어택 검출 디바이스.

#### 청구항 10

제 8 항 또는 제 9 항에 있어서,

제 2 단계 어택 검출기는, 후보 어택 위치를 나타내는 최대 에너지를 가진 분석 세그먼트 전의 분석 세그먼트들에 걸쳐서의 사운드 신호의 평균 에너지를 계산하는 계산기를 구비하는 어택 검출 디바이스.

#### 청구항 11

제 10 항에 있어서,  
후보 어택 위치를 나타내는 최대 에너지를 가진 분석 세그먼트 전의 분석 세그먼트들은 이전 프레임으로부터의 분석 세그먼트들을 구비하는 어택 검출 디바이스.

#### 청구항 12

제 11 항에 있어서,  
제 2 단계 어택 검출기는,  
후보 어택 위치를 나타내는 분석 세그먼트의 에너지와 계산된 평균 에너지간의 비율을 제 1 임계치와 비교하거나, 이전 프레임의 분류가 무성음(UNVOICED)이면, 후보 어택 위치를 나타내는 분석 세그먼트의 에너지와 계산된 평균 에너지간의 비율을 제 2 임계치와 비교하는 제 1 비교기를 구비하는 어택 검출 디바이스.

#### 청구항 13

제 12 항에 있어서,  
제 2 단계 어택 검출기는,  
제 2 단계 어택 검출기의 제 1 비교기에 의한 비교가, 제 2 단계 어택이 검출됨을 나타내면,  
후보 어택 위치를 나타내는 분석 세그먼트의 에너지와 분석 세그먼트들의 장기 에너지(long-term energy)간의 비율을 제 3 임계치와 비교하는 제 2 비교기를 구비하는 어택 검출 디바이스.

#### 청구항 14

제 13 항에 있어서,  
제 2 단계 어택 검출기의 제 2 비교기는, 이전 프레임에서 어택이 검출되었으면, 어택이 없는 것으로 검출하는 어택 검출 디바이스.

#### 청구항 15

제 13 항에 있어서,  
제 2 단계 어택 검출기의 제 1 및 제 2 비교기에 의한 비교가, 제 2 단계 어택 위치가 후보 어택 위치를 나타내는 최대 에너지를 가진 분석 세그먼트임을 나타내면,  
검출된 어택의 위치를 제 2 단계 어택 위치로서 판정하는 결정 모듈을 구비하는

어택 검출 디바이스.

#### 청구항 16

사운드 신호에 있어서의 어택을 코딩하는 디바이스로서,  
청구항 제 1 항 내지 제 15 항 중 어느 한 항에 따른 어택 검출 디바이스; 및  
비-예측 코드북(non-predictive codebook)을 가진 코딩 모드(coding mode)를 이용하여 검출된 어택을 구비한 서브-프레임을 인코딩하는 인코더를 구비하는  
어택 코딩 디바이스.

#### 청구항 17

제 16 항에 있어서,  
코딩 모드는 전이 코딩 모드(transition coding mode)인  
어택 코딩 디바이스.

#### 청구항 18

제 17 항에 있어서,  
비-예측 코드북은 성문음 임펄스 형상(glottal impulse shape)들로 채워진 성문음-형상 코드북(glottal shape codebook)인  
어택 코딩 디바이스.

#### 청구항 19

제 17 항에 있어서,  
어택 검출 디바이스는, 검출된 어택의 위치에 기초하여 전이 코딩 모드로 코딩된 서브-프레임을 판정하는  
어택 코딩 디바이스.

#### 청구항 20

코딩될 사운드 신호에 있어서의 어택을 검출하는 디바이스로서,  
사운드 신호는 다수의 서브-프레임들을 각각으로 포함하는 연속하는 프레임들로 프로세싱되고,  
상기 디바이스는,  
적어도 하나의 프로세서; 및  
프로세서에 결합되고 비-일시적 명령어들을 구비한 메모리를 구비하며,  
상기 비-일시적 명령어들은, 실행시에, 상기 프로세서가,  
현재 프레임의 마지막 서브-프레임에 있어서의 어택을 검출하는 제 1 단계 어택 검출기; 및  
마지막 서브-프레임보다 선행하는 현재 프레임의 서브-프레임에 있어서의 어택을 검출하는 제 2 단계 어택 검출  
기를 구현하게 하는

어택 검출 디바이스.

**청구항 21**

코딩될 사운드 신호에 있어서의 어택을 검출하는 디바이스로서,  
 사운드 신호는 다수의 서브-프레임들을 각각으로 포함하는 연속하는 프레임들로 프로세싱되고,  
 상기 디바이스는,  
 적어도 하나의 프로세서; 및  
 프로세서에 결합되고 비-일시적 명령어들을 구비한 메모리를 구비하며,  
 상기 비-일시적 명령어들은, 실행시에, 상기 프로세서가,  
 제 1 단계에서, 현재 프레임의 마지막 서브-프레임에 위치한 어택을 검출하게 하고,  
 제 2 단계에서, 마지막 서브-프레임보다 선행하는 현재 프레임의 서브-프레임에 위치한 어택을 검출하게 하는  
 어택 검출 디바이스.

**청구항 22**

코딩될 사운드 신호에 있어서의 어택을 검출하는 방법으로서,  
 사운드 신호는 다수의 서브-프레임들을 각각으로 포함하는 연속하는 프레임들로 프로세싱되고,  
 상기 방법은,  
 현재 프레임의 마지막 서브-프레임에 있어서의 어택을 검출하는 제 1 단계 어택 검출; 및  
 마지막 서브-프레임보다 선행하는 서브-프레임들을 포함하는, 현재 프레임의 서브-프레임들중 하나에 있어서의  
 어택을 검출하는 제 2 단계 어택 검출을 구비하는  
 어택 검출 방법.

**청구항 23**

제 22 항에 있어서,  
 현재 프레임이 제네릭 코딩 모드(generic coding mode)를 이용하여 코딩되도록 이전에 분류된 활성 프레임  
 (active frame)인지를 판정하고, 현재 프레임이 제네릭 코딩 모드를 이용하여 코딩되도록 이전에 분류된 활성  
 프레임으로서 판정되지 않으면, 검출되는 어택이 없음을 나타내는 것을 구비하는  
 어택 검출 방법.

**청구항 24**

제 22 항 또는 제 23 항에 있어서,  
 현재 프레임의 다수의 분석 세그먼트들(analysis segments)에 있어서의 사운드 신호의 에너지를 계산하고;  
 제 1 단계 및 제 2 단계 어택 검출들에 의해 인증될 후보 어택 위치를 나타내는 최대 에너지를 가진 분석 세그  
 먼트들 중의 한 세그먼트를 탐지하는 것을 구비하는  
 어택 검출 방법.

**청구항 25**

제 24 항에 있어서,

제 1 단계 어택 검출은,

현재 프레임에 있어서의 마지막 서브-프레임 전의 분석 세그먼트들에 걸쳐서의 제 1 평균 에너지를 계산하고;

최대 에너지를 가진 분석 세그먼트에서부터 현재 프레임의 마지막 분석 세그먼트까지의, 현재 프레임의 분석 세그먼트들에 걸쳐서의 제 2 평균 에너지를 계산하는 것을 구비하는

어택 검출 방법.

**청구항 26**

제 25 항에 있어서,

제 1 단계 어택 검출은,

제 1 비교기를 이용하여, 제 1 평균 에너지와 제 2 평균 에너지간의 비율을, 제 1 임계치와 비교하거나, 이전 프레임의 분류가 유성음(VOICED)이면, 제 1 평균 에너지와 제 2 평균 에너지간의 비율을 제 2 임계치와 비교하는 것을 구비하는

어택 검출 방법.

**청구항 27**

제 26 항에 있어서,

제 1 단계 어택 검출은,

제 1 비교기에 의한 비교가, 제 1 단계 어택이 검출됨을 나타내면,

제 2 비교기를 이용하여, 최대 에너지의 분석 세그먼트의 에너지와 현재 프레임의 다른 분석 세그먼트들의 에너지간의 비율을, 제 3 임계치와 비교하는 것을 구비하는

어택 검출 방법.

**청구항 28**

제 27 항에 있어서,

제 1 및 제 2 비교기에 의한 비교가, 제 1 단계 어택 위치가 후보 어택 위치를 나타내는 최대 에너지를 가진 분석 세그먼트임을 나타내면,

제 1 단계 어택 위치가 현재 프레임의 마지막 서브-프레임 전의 다수의 분석 세그먼트들과 동일하거나 그보다 더 큰지를 판정하고, 제 1 단계 어택 위치가 마지막 서브-프레임 전의 다수의 분석 세그먼트들과 동일하거나 그보다 더 큰 경우, 검출된 어택 위치를, 현재 프레임의 마지막 서브-프레임에 있어서의 제 1 단계 어택 위치로서 판정하는 것을 구비하는

어택 검출 방법.

**청구항 29**

제 22 항 내지 제 28 항 중 어느 한 항에 있어서,

제 2 단계 어택 검출은 제 1 단계 어택 검출에 의해 검출되는 어택이 없을 때 이용되는

어택 검출 방법.

### 청구항 30

제 29 항에 있어서,

현재 프레임이 유성음(VOICED)으로서 분류되는지를 판정하는 것을 구비하고,

현재 프레임이 유성음으로서 분류되지 않으면, 제 2 단계 어택 검출이 이용되는

어택 검출 방법.

### 청구항 31

제 29 항 또는 제 30 항에 있어서,

제 2 단계 어택 검출은, 후보 어택 위치를 나타내는 최대 에너지를 가진 분석 세그먼트 전의 분석 세그먼트들에 걸쳐서의 사운드 신호의 평균 에너지를 계산하는 것을 구비하는

어택 검출 방법.

### 청구항 32

제 31 항에 있어서,

후보 어택 위치를 나타내는 최대 에너지를 가진 분석 세그먼트 전의 분석 세그먼트들은 이전 프레임으로부터의 분석 세그먼트들을 구비하는

어택 검출 방법.

### 청구항 33

제 32 항에 있어서,

제 2 단계 어택 검출은,

제 1 비교기를 이용하여, 후보 어택 위치를 나타내는 분석 세그먼트의 에너지와 계산된 평균 에너지간의 비율을 제 1 임계치와 비교하거나, 이전 프레임의 분류가 무성음(UNVOICED)이면, 후보 어택 위치를 나타내는 분석 세그먼트의 에너지와 계산된 평균 에너지간의 비율을 제 2 임계치와 비교하는 것을 구비하는

어택 검출 방법.

### 청구항 34

제 33 항에 있어서,

제 2 단계 어택 검출은,

제 2 단계 어택 검출의 제 1 비교기에 의한 비교가, 제 2 단계 어택이 검출됨을 나타내면,

제 2 비교기를 이용하여, 후보 어택 위치를 나타내는 분석 세그먼트의 에너지와 분석 세그먼트들의 장기 에너지(long-term energy)간의 비율을 제 3 임계치와 비교하는 것을 구비하는

어택 검출 방법.

**청구항 35**

제 34 항에 있어서,

제 2 단계 어택 검출의 제 2 비교기에 의한 비교는, 이전 프레임에서 어택이 검출되었으면, 어택이 없는 것으로 검출하는

어택 검출 방법.

**청구항 36**

제 34 항에 있어서,

제 2 단계 어택 검출의 제 1 및 제 2 비교기에 의한 비교가, 제 2 단계 어택 위치가 후보 어택 위치를 나타내는 최대 에너지를 가진 분석 세그먼트임을 나타내면,

검출된 어택의 위치를 제 2 단계 어택 위치로서 판정하는 것을 구비하는

어택 검출 방법.

**청구항 37**

사운드 신호에 있어서의 어택을 코딩하는 방법으로서,

청구항 제 22 항 내지 제 36 항 중 어느 한 항에 따른 어택 검출 방법; 및

비-예측 코드북(non-predictive codebook)을 가진 코딩 모드(coding mode)를 이용하여 검출된 어택을 구비한 서브-프레임을 인코딩하는 것을 구비하는

어택 코딩 방법.

**청구항 38**

제 37 항에 있어서,

코딩 모드는 전이 코딩 모드(transition coding mode)인

어택 코딩 방법.

**청구항 39**

제 38 항에 있어서,

비-예측 코드북은 성문음 임펄스 형상(glottal impulse shape)들로 채워진 성문음-형상 코드북(glottal shape codebook)인

어택 코딩 방법.

**청구항 40**

제 38 항에 있어서,

검출된 어택의 위치에 기초하여 전이 코딩 모드로 코딩된 서브-프레임을 판정하는 것을 구비하는

어택 코딩 방법.

**발명의 설명**

**기술 분야**

- [0001] 본 개시는, 예를 들어, 스피치(speech) 또는 오디오 신호와 같은 사운드 신호의 전송 및 합성의 견지에서, 사운드 신호를 코딩하는 기술에 관한 것이다.
- [0002] 보다 구체적으로, 본 개시는, 예를 들어, 스피치 또는 오디오 신호와 같은, 코딩될 사운드 신호에 있어서의 어택(attack)을 검출하고, 검출된 어택을 코딩하는 방법들 및 디바이스들에 관한 것이지만, 배타적인 것은 아니다.
- [0003] 본 개시 및 첨부된 청구항들에 있어서,
- [0004] - 용어 "어택"은, 예를 들어, 유성음 온셋들(voiced onsets)(무성음 스피치 세그먼트(unvoiced speech segment)로부터 유성음 스피치 세그먼트(voiced speech segment)로의 전이(transition)), 다른 사운드 온셋들, 전이들(transitions), 파열음(plosive)들 등과 같은, 신호의 저-고 에너지 변경(low-to-high energy change)을 지칭하는 것으로, 사운드 신호 세그먼트(sound signal segment)내의 급격한 에너지 증가를 특징으로 한다.
- [0005] - 용어 "온셋"은, 예를 들어, 스피치, 뮤지컬 노트(musical note), 또는 다른 사운드와 같은, 유의적 사운드 이벤트(significant sound event)의 시작을 지칭한다.
- [0006] - 용어 "파열음"은, 음성학에 있어서, 모든 기류(airflow)가 중단되도록 보컬 트랙(vocal track)이 차단되는 자음(consonant)을 지칭한다.
- [0007] - 용어 "검출된 어택의 코딩"은 어택의 시작 후 길이가 대체적으로 수밀리초(few millisecond)인 사운드 신호 세그먼트의 코딩을 지칭한다.

**배경 기술**

- [0008] 스피치 인코더(speech encoder)는 스피치 신호를 디지털 비트 스트림으로 변환하는데, 그 디지털 비트 스트림은 통신 채널을 통해 전송되거나 저장 매체에 저장된다. 스피치 신호는 디지털화되며, 즉, 샘플링되고 통상적으로 샘플당 16비트로 양자화된다. 스피치 인코더는 소수의 비트들을 가진 이 디지털 샘플들을 나타내면서 양호한 주관적 스피치 품질(good subjective speech quality)을 유지시키는 역할을 한다. 스피치 디코더 또는 합성기는 전송되거나 저장된 디지털 비트 스트림에 대해 작용하여 그것을 스피치 신호로 되변환한다.
- [0009] CELP(Code-Excited Linear Prediction) 코딩은 주관적 품질과 비트율간의 양호한 절충을 달성하는 최상의 기법들 중 하나이다. 이 코딩 기법은 무선 및 유선 애플리케이션의 둘 모두에 있어서 여러 스피치 코딩 표준들의 기초를 형성한다. CELP 코딩에 있어서, 샘플링된 스피치 신호는 통상적으로 프레임이라고 지칭하는 M개 샘플들의 연속하는 블록들로 프로세싱(processing)되며, 여기서, M은 전형적으로 10-30ms에 대응하는 스피치 샘플들의 사전 결정된 개수이다. LP(Liner Prediction) 필터는 매 프레임마다 계산되고 전송된다. LP 필터의 계산은, 전형적으로, 예를 들어, 후속하는 프레임으로부터 5-15ms의 스피치 세그먼트와 같은, 룩어헤드(lookahead)를 필요로 한다. 각각의 M-샘플 프레임은, 서브-프레임이라고 하는, 보다 적은 블록들로 분할된다. 통상, 서브-프레임들의 개수는 2개 내지 5개로서, 4-10ms의 서브-프레임들로 결과한다. 각 서브-프레임에 있어서, 여기(excitation)는, 통상적으로, 과거 여기 기여(past excitation contribution)와 이노베이티브 고정 코드북 여기 기여(innovative, fixed codebook excitation contribution)와 같은 2개의 성분으로부터 획득된다. 과거 여기 기여는, 피치(pitch) 또는 적응적 코드북 여기 기여라고 지칭하기도 한다. 여기를 특징짓는 파라메타들은 코드화되어 디코더로 전송되며, 그 여기는 재구성되어 LP 합성 필터에 대한 입력으로서 공급된다.
- [0010] CELP-기반 스피치 코덱(codec)들은 그들의 높은 성능을 달성하기 위해 예측에 많이 의존한다. 그러한 예측은 다른 유형들일 수 있지만 통상적으로 이전 프레임들로부터 선택된 적응적 코드북 여기 기여를 저장하는 적응적 코드북의 이용을 구비한다. CELP 인코더는 과거 적응적 코드북 여기 기여에서, 현재 코딩되고 있는 세그먼트와 가장 유사한 세그먼트를 탐색함에 의해 유성음 스피치의 주기(quasi periodicity)를 활용한다. 동일한 과거 적응적 코드북 여기 기여가 또한 디코더에 저장된다. 그 다음, 인코더에서 이용된 것과 동일한 적응적 코드북 여기 기여를 디코더가 재구성하도록 하기 위해 인코더는 피치 지연 및 피치 이득을 송신하면 충분하다. 추가로, 이전 스피치 세그먼트와 현재 코딩된 스피치 세그먼트간의 에볼루션(evolution)(차이)은 고정 코드북으로부터 선택된 고정 코드북 여기 기여를 이용하여 모델링된다.

[0011] CELP-기반 스피치 코덱들에 고유한 예측에 관한 문제는, 인코더의 상태 및 디코더의 상태가 비동기화될 때의 전송 에러(소거된 프레임들 또는 패킷들)의 존재시에 나타난다. 예측으로 인해, 소거된 프레임의 영향은 소거된 프레임에 국한되는 것이 아니라, 프레임 소거 이후, 여러개의 후속하는 프레임들 동안에 계속적으로 전파된다. 당연하게도, 인지 영향(perceptual impact)은 매우 짜증나는 것일 수 있다. 무성음 스피치 세그먼트에서 유성음 스피치 세그먼트로의 전이(예를 들어, 불활성 스피치의 기간 또는 자음과 모음(vowel)간의 전이) 또는 2개의 서로 다른 유성음 세그먼트들간의 전이(예를 들어, 2개의 모음들간의 전이)와 같은 어택들은, 프레임 소거 은폐(frame erasure concealment)를 위한 가장 큰 과제들 중 하나이다. 무성음 스피치 세그먼트에서 유성음 스피치 세그먼트로의 전이(유성음 온셋)가 손실되면, 유성음 온셋 프레임 직전의 프레임은 무성음화되거나 불활성화되고, 그에 따라 적응적 코드북의 버퍼에서 의미있는 여기 기여가 발견되지 않게 된다. 인코더에서, 유성음 온셋 프레임동안 적응적 코드북에 과거 여기 기여가 구축되며, 뒤이은 유성음 프레임은 이러한 과거 적응적 코드북 여기 기여를 이용하여 코딩된다. 대부분의 프레임 에러 은폐 기술은 누락 프레임을 은폐하기 위해 정확하게 수신된 마지막 프레임으로부터의 정보를 이용한다. 유성음 온셋 프레임이 손실되면 디코더에 있는 적응적 코드북의 버퍼는 이전 프레임(무성음 또는 불활성 프레임)의 잡음성(noise-like) 적응적 코드북 여기 기여를 이용하여 갱신될 것이다. 따라서, 여기의 주기적 부분(periodic part)(적응적 코드북 여기 기여)은 손실된 유성음 온셋 이후, 디코더에 있는 적응적 코드북에서 완전히 누락되며, 디코더가 이러한 손실을 복구하는데에 여러 개의 프레임들이 소요될 수 있다. 손실된 유성음에서 유성음으로의 전이의 경우에 유사한 상황이 발생한다. 그 경우, 전이 프레임 전에 적응적 코드북에 저장된 여기 기여는, 전형적으로, 전이 이후에 적응적 코드북에 저장된 여기 기여와 매우 다른 특성들을 가지다. 다시, 통상적으로 디코더가 과거 프레임 정보의 이용으로 손실된 프레임을 은폐함에 따라, 인코더의 상태와 디코더의 상태는 매우 다를 것이며, 합성된 신호는 중요한 왜곡을 겪을 수 있다. 이러한 문제에 대한 해법은, 참조 [2]에 소개되어 있으며, 거기에서는, 전이 프레임을 뒤따르는 프레임에 있어서, 프레임간 종속성 적응적 코드북(inter-frame dependent adaptive codebook)이 비-예측 성문음 형상 코드북(non-predictive glottal-shape codebook)으로 대체된다.

[0012] CELP-기반 코덱들에 있어서 전이 프레임들을 코딩할 때의 또 다른 문제는 코딩 효율이다. 코덱이 이전 및 현재 세그먼트 여기가 매우 다른 전이를 프로세싱할 경우, 코딩 효율은 감소한다. 이러한 경우들은, 예를 들어, 유성음 온셋들(무성음 스피치 세그먼트에서 유성음 스피치 세그먼트로의 전이), 다른 사운드 온셋들, 2개의 서로 다른 유성음 세그먼트들간의 전이(예를 들어, 2개의 모음들간의 전이), 파열음 등과 같은 어택을 인코딩하는 프레임들에서 발생한다. 이하의 2가지 문제들은 효율성에 있어서의 그러한 감소에 주로 기여한다(주로 참조 [1]). 첫번째 문제로서, 장기 예측(long-time prediction)의 효율은 빈약하며, 따라서, 전체 여기에 대한 적응적 코드북 여기의 기여도 약하다. 두번째 문제는, 통상적으로 프레임내의 급격한 에너지 증가에 적절하게 반응할 수 없는, 제한된 비트-예산(limited bit-budget)을 이용하는 벡터 양자화기들로서 고안되는, 이득 양자화기들과 관련된다. 이러한 급격한 에너지 증가가 프레임의 중단에 가까운 곳에 발생할수록 두 번째 문제는 더욱 심화된다.

**발명의 내용**

**해결하려는 과제**

[0013] 상술한 문제점을 극복하기 위하여, 온셋 프레임들 및 전이 프레임들과 같은 어택을 포함하는 프레임들의 코딩 효율을 개선하는, 그리고, 보다 일반적으로는, CELP-기반 코덱에 있어서 코딩 품질을 개선하기 위한 방법 및 디바이스가 필요하다.

**과제의 해결 수단**

[0014] 제 1 측면에 따르면, 본 개시는 코딩될 사운드 신호에 있어서 어택을 검출하는 방법에 관한 것이며, 여기에서, 사운드 신호는 각각이 다수의 서브-프레임들을 포함하는 연속하는 프레임들로 프로세싱된다. 그 방법은, 현재 프레임의 마지막 서브-프레임에 있어서의 어택을 검출하는 제 1 단계 어택 검출과, 마지막 서브-프레임에 선행하는 서브-프레임을 포함하는 현재 프레임의 서브-프레임들 중 하나에 있어서의 어택을 검출하는 제 2 단계 어택 검출을 구비한다.

[0015] 본 개시는, 사운드 신호에 있어서의 어택을 코딩하는 방법에 관한 것으로, 그 방법은 상기에서 정의한 어택 검출 방법을 구비한다. 코딩 방법은 비-예측 코드북(non-predictive codebook)을 가진 코딩 모드를 이용하여 검출된 어택을 구비한 서브-프레임을 인코딩하는 것을 구비한다.

[0016] 또 다른 측면에 따르면, 본 개시는 코딩될 사운드 신호에 있어서 어택을 검출하는 디바이스에 관한 것이며, 여

기에서, 사운드 신호는 각각이 다수의 서브-프레임들을 포함하는 연속하는 프레임들로 프로세싱된다. 그 디바이스는, 현재 프레임의 마지막 서브-프레임에 있어서의 어택을 검출하는 제 1 단계 어택 검출기와, 마지막 서브-프레임에 선행하는 서브-프레임을 포함하는 현재 프레임의 서브-프레임들 중 하나에 있어서의 어택을 검출하는 제 2 단계 어택 검출기를 구비한다.

[0017] 본 개시는 사운드 신호에 있어서의 어택을 코딩하는 디바이스에 관한 것으로, 그 디바이스는 상기에서 정의한 어택 검출 디바이스와, 비-예측 코드북을 가진 코딩 모드를 이용하여 검출된 어택을 구비한 서브-프레임의 인코더를 구비한다.

[0018] 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들의 상술한 및 다른 목적들, 장점들 및 특징들은, 첨부된 도면을 참조하여 예시에 의해 주어진 예시적인 실시 예의 이하의 비-제한적 설명을 읽으면 보다 명백해질 것이다.

**도면의 간단한 설명**

[0019] 첨부 도면에 있어서:

도 1은 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들의 구현의 가능한 맥락을 도시한 사운드 프로세싱 및 통신 시스템의 개략적인 블록도이고,

도 2는, 도 1의 사운드 프로세싱 및 통신 시스템의 일부를 형성하는, CELP-기반 인코더 및 디코더의 구조를 도시한 개략적인 블록도이고,

도 3은 EVS(Enhanced Voice Services) 코딩 모드 분류기의 모듈들 및 EVS(Enhanced Voice Services) 코딩 모드 분류 방법의 동작들을 동시에 도시한 블록도이고,

도 4는 코딩될 사운드 신호에 있어서의 어택을 검출하는 방법 및 그 방법을 구현한 어택 검출기의 모듈들의 동작들을 동시에 도시한 블록도이고,

도 5는 도 4의 어택 검출기의 영향과, 디코딩된 스피치 신호의 품질에 대한 TC(Transition Coding) 코딩 모드를 보여주는 제 1의 비 제한적 예시의 그래프로서, 곡선 a)는 입력 스피치 신호를 나타내고, 곡선 b)는 참조 스피치 신호 합성을 나타내며, 곡선 c)는 도 4의 어택 검출기와 TC 코딩 모드가 온셋 프레임을 프로세싱하는데 이용될 때, 개선된 스피치 신호 합성을 나타내고,

도 6은 도 4의 어택 검출기의 영향과, 디코딩된 스피치 신호의 품질에 대한 TC(Transition Coding) 코딩 모드를 보여주는 제 2의 비 제한적 예시의 그래프로서, 곡선 a)는 입력 스피치 신호를 나타내고, 곡선 b)는 참조 스피치 신호 합성을 나타내며, 곡선 c)는 도 4의 어택 검출기와 TC 코딩 모드가 온셋 프레임을 프로세싱하는데 이용될 때, 개선된 스피치 신호 합성을 나타내고,

도 7은 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들을 구현한 하드웨어 부품들의 예시적인 구성의 단순화된 블록도이다.

**발명을 실시하기 위한 구체적인 내용**

[0020] 이하의 설명에서는, 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들의 비제한적인 예시적 실시 예들이 스피치 신호 및 CELP-기반 코덱과 관련하여 설명될 것이지만, 이 방법들 및 디바이스들은 스피치 신호들 및 CELP-기반 코덱들에 대한 애플리케이션에 국한되는 것은 아니며, 그들의 원리 및 개념들이 임의의 다른 유형의 사운드 신호들 및 코덱들에 적용될 수 있음을 알아야 한다.

[0021] 이하의 설명은, 예를 들어, 스피치 또는 오디오 신호와 같은 사운드 신호에 있어서의 어택을 검출하고, 어택이 검출되는 서브-프레임들에 TC(Transition Coding) 모드가 이용되게 하는 것과 연관된다. 어택의 검출은, 서브-프레임을 선택하는데 이용될 수 있는데, 거기에서는 TC 코딩 모드의 일부로서 성문음-형상 코드북(glottal-shape codebook)이 적응적 코드북을 대신하여 채용된다.

[0022] 참조 [4]에 설명된 바와 같은 EVS 코덱에서는, 검출 알고리즘이 현재 프레임의 마지막 서브-프레임에서 어택을 검출할 경우, 이 마지막 서브-프레임에 TC 코딩 모드의 성문음-형상 코드북이 이용된다. 본 개시에 있어서, 검출 알고리즘에는, 어택을 포함하는 상당수의 프레임들을 검출하고, 그러한 서브-프레임들의 코딩시에 어택이 검출되는 모든 서브-프레임들에 TC 코딩 모드 및 대응하는 성문음-형상 코드북을 이용하게 하기 위하여, 제 2 단

계 로직(second-stage logic)이 보완된다.

- [0023] 상술한 기법은 코딩될 사운드 신호에서 검출된 어택과 특정 뮤직 세그먼트(예를 들어, 캐스터네츠(castanets))의 코딩 효율을 개선한다. 보다 일반적으로는, 코딩 품질이 개선된다.
- [0024] 도 1은 이하의 설명에서 개시된 바와 같이 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들의 구현의 가능한 맥락을 도시한 사운드 프로세싱 및 통신 시스템(100)의 개략적인 블럭도이다.
- [0025] 도 1의 사운드 프로세싱 및 통신 시스템(100)은 통신 채널(101)을 통한 사운드 신호의 전송을 지원한다. 통신 채널(101)은, 예를 들어, 유선 또는 광섬유 링크를 구비할 수 있다. 대안적으로, 통신 채널(101)은, 적어도 부분적으로, 무선 주파수 링크를 구비할 수 있다. 무선 주파수 링크는, 셀룰러 폰(cellular telephone)에서 발견될 수 있는 것과 같은 공유된 대역폭 리소스(shared bandwidth resource)를 필요로 하는 다수의 동시 통신들을 지원하기도 한다. 도시되지는 않았지만, 통신 채널(101)은, 시스템(100)의 단일 디바이스 구현시에, 추후 재생을 위해 인코딩된 사운드 신호를 기록 및 저장하는 저장 디바이스로 대체될 수 있다.
- [0026] 도 1을 참조하면, 예를 들어, 마이크로폰(102)은 원시 아날로그 사운드 신호(103)를 생성한다. 상기에서 지적한 바와 같이, 사운드 신호(103)는, 특히, 스피치 및/또는 오디오를 구비할 수 있지만, 배타적인 것은 아니다.
- [0027] 아날로그 사운드 신호(103)는 그를 원시 디지털 사운드 신호(105)로 변환하는 아날로그-디지털(A/D) 변환기(104)로 공급된다. 원시 디지털 사운드 신호(105)는 기록될 수 있고 저장 디바이스(도시되지 않음)로부터 공급될 수 있다.
- [0028] 사운드 인코더(106)는 디지털 사운드 신호(105)를 인코딩하며, 그에 의해 광학 에러-정정 채널 인코더(108)로 운송되는 비트 스트림(107) 형태로 다중화되는 인코딩 파라메타들의 세트를 생성한다. 광학 에러 정정 채널 인코더(108)는, 존재할 경우, 비트 스트림(107)에 있어서의 인코딩 파라메타들의 이진 표현에 용장성을 추가하는데, 이는 통신 채널(101)을 통해 결과하는 비트 스트림(111)을 전송하기 전에 이루어진다.
- [0029] 수신측 상에서는, 광학 에러-정정 채널 디코더(109)가 수신된 디지털 비트 스트림(111)에 있어서의 상술한 용장성 정보를 이용하여, 통신 채널(101)을 통한 전송동안에 발생되었을 수 있는 에러를 검출 및 정정함으로써, 수신된 인코딩 파라메타들을 가진 에러 정정된 비트 스트림(112)을 생성한다. 사운드 디코더(110)는 비트 스트림(112)에 있어서의 수신된 인코딩 파라메타들을 변환하여 합성된 디지털 사운드 신호(113)를 생성한다. 사운드 디코더(110)에서 재구성된 디지털 사운드 신호(113)는 디지털-아날로그(D/A) 변환기(115)에서 합성된 아날로그 사운드 신호(114)로 변환된다.
- [0030] 합성된 아날로그 사운드 신호(114)는 확성기 유닛(116)에서 재생된다(확성기 유닛(116)은 명백하게 헤드폰으로 대체될 수 있다). 대안적으로, 사운드 디코더(110)로부터의 디지털 사운드 신호(113)는, 또한, 저장 디바이스(도시되지 않음)에 공급되어 기록될 수 있다.
- [0031] 비-제한적 예시로서, 본 개시에 따라 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들은 도 1의 사운드 인코더(106) 및 디코더(110)에 구현될 수 있다. 도 1의 사운드 프로세싱 및 통신 시스템(100)은, 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들과 함께, 인코더(106)의 입력과 디코더(110)의 출력이 스테레오 사운드 신호의 좌우측 채널들로 이루어진 스테레오 음향(stereophony)의 경우를 포괄하도록 확장될 수 있다. 도 1의 사운드 프로세싱 및 통신 시스템(100)은, 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들과 함께, 멀티-채널(multi-channel) 및/또는 장면-기반 오디오(scene-based audio) 및/또는 독립적인 스트림 인코딩 및 디코딩(예를 들어, 서라운드 및 수준높은 앰비소닉스(surround and high-order ambisonics))의 경우를 포괄하도록 추가 확장될 수 있다.
- [0032] 도 2는, 예시적인 실시 예에 따라, 도 1의 사운드 프로세싱 및 통신 시스템(100)의 일부를 형성하는, CELP-기반 인코더 및 디코더의 구조를 도시한 개략적인 블럭도이다. 도 2에 도시된 바와 같이, 사운드 코덱은, 2개의 기본 부분들, 즉, 도 1의 이전 설명에서 소개한 사운드 인코더(106)와 사운드 디코더(110)를 구비한다. 인코더(106)는 원시 디지털 사운드 신호(105)를 공급받고, 원시 아날로그 신호(103)를 나타내는 인코딩 파라메타들(107)을 판정하는데, 이에 대해서는 이하에서 설명하겠다. 이 파라메타들(107)은, 디지털 비트 스트림(111)으로 인코딩된다. 이미 설명한 바와 같이, 비트 스트림(111)은, 예를 들어, 도 1의 통신 채널(101)과 같은 통신 채널을 이용하여, 디코더(110)로 전송된다. 사운드 디코더(110)는 원시 디지털 사운드 신호(105)와 가능한 유사하게 되도

록 합성된 디지털 신호(113)를 재구성한다.

- [0033] 현재, 가장 광범위한 스피치 코딩 기법은 LP(Linear Prediction), 특히, CELP에 기반한다. LP-기반 코딩에 있어서, 합성된 디지털 사운드 신호(230)(도 2)는, 전달 함수  $1/A(z)$ 를 가진 LP 합성 필터(216)를 통해 여기(214)를 필터링함으로써 생성된다. LP 필터의 필터 파라미터  $A(z)$ 를 탐지하기 위한 절차는, 예를 들어, 참조 [4]에서 발견될 수 있다.
- [0034] CELP에 있어서, 여기(214)는, 전형적으로, 2 부분들, 즉, 인덱스  $t$ (피치 레그(pitch lag))에 응답하여 적응적 코드북(218)으로부터 과거 여기 신호  $v(n)$ 를 선택하고, 과거 여기 신호  $v(n)$ 를 적응적-코드북 이득  $g_p$ (226)만큼 증폭시킴에 의해 생성되는 제 1 단계의 적응적-코드북 기여(222)와; 인덱스  $k$ 에 응답하여 고정 코드북(220)으로부터 이노베이트브 코드벡터  $c_k(n)$ 를 선택하고, 이노베이트브 코드벡터  $c_k(n)$ 를 고정-코드북 이득  $g_c$ (228)만큼 증폭시킴에 의해 생성되는 제 2 단계의 고정-코드북 기여(224)로 이루어진다. 일반적으로, 적응적 코드북 기여(222)는 여기의 주기적 부분을 모델링하고, 고정 코드북 여기 기여(224)는 사운드 신호의 에볼루션(evolution)을 모델링하기 위해 추가된다.
- [0035] 사운드 신호는 전형적으로 20ms의 프레임들에 의해 프로세싱되며, LP 필터의 필터 파라미터  $A(z)$ 는 프레임당 1 회씩 인코더(106)로부터 디코더(110)로 전송된다. CELP에 있어서, 프레임들은 여기를 인코딩하기 위해 여러 개의 서브-프레임들로 추가 분할된다. 서브-프레임 길이는 전형적으로 5ms이다.
- [0036] CELP는, 가능한 디코더 출력들이, 인코더(106)에서의 코딩 프로세싱동안 이미 시도(합성)되고 원시 디지털 사운드 신호(105)와 비교되는, 합성에 의한 분석(Analysis-by-Synthesis)이라고 하는 원리를 이용한다. 따라서, 인코더(106)는 디코더(110)의 요소들과 유사한 요소들을 포함한다. 이 요소들은 (디코더(110)에서의 적응적 코드북(218)에 대응하는) 적응적 코드북(242)으로부터의 인덱스  $t$ (피치 레그)에 응답하여 선택된 (디코더(110)에서의 적응적-코드북 기여(222)에 대응하는) 적응적 코드북 여기 기여(250)를 포함하며, 적응적 코드북(242)는 (LP 합성 필터  $1/A(z)$ 와 인지 가중 필터  $W(z)$ 의 캐스케이드(cascade)인) 가중 합성 필터  $H(z)$ (238)의 임펄스 응답과 콘볼루션되는 과거 여기 신호  $v(n)$ 를 공급하고, 가중 합성 필터(238)의 출력  $y_1(n)$ 는 (디코더(110)에서의 적응적-코드북 이득(226)에 대응하는) 적응적-코드북 이득  $g_p$ (240)에 의해 증폭된다. 이 요소들은 (디코더(110)에서의 고정 코드북(220)에 대응하는) 고정 코드북(244)으로부터의 인덱스  $k$ 에 응답하여 선택된 (디코더(110)에서의 고정 코드북 기여(224)에 대응하는) 고정 코드북 여기 기여(252)를 포함하고, 고정 코드북(244)은 가중 합성 필터  $H(z)$ (246)의 임펄스 응답과 콘볼루션되는 이노베이트브 코드벡터  $c_k(n)$ 를 공급하며, 가중 합성 필터(246)의 출력  $y_2(n)$ 은 (디코더(110)에서의 고정-코드북 이득(228)에 대응하는) 고정 코드북 이득  $g_c$ (248)에 의해 증폭된다.
- [0037] 인코더(106)는 인지 가중 필터  $W(z)$ (233)와, LP 합성 필터  $1/A(z)$ 와 인지 가중 필터  $W(z)$ 의 캐스케이드( $H(z)$ )의 제로-입력 응답(zero-input response)의 계산기(234)를 구비한다. 감산기(236, 254 및 256)는, 각각, 계산기(234)로부터의 제로-입력 응답, 적응적 코드북 기여(250) 및 고정 코드북 기여(252)을 인지 가중 필터(233)에 의해 필터링된 원시 디지털 사운드 신호(105)로부터 감산하여, 원시 디지털 사운드 신호(105)와 합성된 디지털 사운드 신호(113)간의 평균-제곱근 에러(mean-squared error)(232)를 계산하는데 이용되는 에러 신호를 제공한다(도 1 참조).
- [0038] 적응적 코드북(242) 및 고정 코드북(244)은 인지 가중 도메인(perceptually weighted domain)에서 원시 디지털 사운드 신호(105)와 합성 디지털 사운드 신호(113)간의 평균-제곱근 에러(232)를 최소화하도록 탐색되며, 이산 시간 인덱스  $n = 0, 1, \dots, N-1$ 이고,  $N$ 은 서브-프레임의 길이이다. 평균-제곱근 에러의 최소화는 디지털 사운드 신호(105)를 코딩하기 위한 최상 후보의 (인덱스  $t$ 에 의해 식별되는) 과거 여기 신호  $v(n)$  및 (인덱스  $k$ 에 의해 식별되는) 이노베이트브 코드벡터  $c_k(n)$ 를 제공한다. 인지 가중 필터  $W(z)$ 는 주파수 마스킹 효과(frequency masking effect)를 활용하고, 전형적으로 LP 필터  $A(z)$ 로부터 도출된다. WB(광대역, 전형적으로 50-7000Hz의 대역폭) 신호들에 대한 인지 가중 필터  $W(z)$ 의 예시는 참조 [4]에서 발견될 수 있다.
- [0039] LP 합성 필터  $1/A(z)$ 와 가중 필터  $W(z)$ 의 메모리가 탐색된 이노베이트브 코드벡터  $c_k(n)$ 로부터 독립적이기 때문에, 이 메모리(LP 합성 필터  $1/A(z)$ 와 인지 가중 필터  $W(z)$ 의 캐스케이드( $H(z)$ )의 제로-입력 응답)는 고정된 코드북 탐색에 앞서 원시 디지털 사운드 신호(105)로부터 감산될 수 있다(감산기(236)). 후보 이노베이트브 코드벡터  $c_k(n)$ 의 필터링은, 도 2에  $H(z)$ 로 표시된, 필터  $1/A(z)$ 와  $W(z)$ 의 캐스케이드의 임펄스 응답과의 콘볼루션

에 의해 수행될 수 있다.

- [0040] 인코더(106)에서 디코더(110)로 전송되는 디지털 비트 스트림(111)은 전형적으로 다음과 같은 파라메타들(107), 즉, LP 필터  $A(z)$ 의 양자화된 파라메타들, 적응적 코드북(242)의 인덱스  $t$ 와 고정 코드북(244)의 인덱스  $k$ , 적응적 코드북(242)과 고정 코드북(244)의 이득들  $g_b(240)$  및  $g_c(248)$ 을 포함한다. 디코더(110)에서는,
- [0041] - LP 필터  $A(z)$ 의 수신된 양자화된 파라메타들이 LP 합성 필터(216)를 구축하는데 이용되고,
- [0042] - 수신된 인덱스  $t$ 는 적응적 코드북(218)에 적용되고,
- [0043] - 수신된 인덱스  $k$ 는 고정 코드북(220)에 적용되고,
- [0044] - 수신된 이득  $g_b$ 는 적응적-코드북 이득(226)으로서 이용되고,
- [0045] - 수신된 이득  $g_c$ 는 고정-코드북 이득(228)으로서 이용된다.
- [0046] CELP-기반 인코더 및 디코더의 구조 및 동작의 추가적인 설명은, 예를 들어, 참조 [4]에서 발견될 수 있다.
- [0047] 또한, 이하의 설명은 EVS 표준(참조 [4])을 참조하여 이루어지지만, 거기에 설명된 개념, 원리, 구조 및 동작들은 다른 사운드/스피치 프로세싱 및 통신 표준에 적용될 수 있음을 알아야 한다.
- [0048] **유성음 온셋들의 코딩**
- [0049] 보다 나은 코딩 성능을 획득하기 위하여, 참조 [4]에 설명된 EVS 코덱의 LP-기반 코어는 신호 분류 알고리즘과, 신호의 각 카테고리에 맞춤화된 6개의 개별적인 코딩 모드들, 즉, IC(Inactive Coding) 모드, UC(Unvoiced Coding) 모드, TC(Transition Coding) 모드, VC(Voiced Coding) 모드, GC(Generic Coding) 모드, AC(Audio Coding) 모드(도시되지 않음)를 이용한다.
- [0050] 도 3은 EVS(Enhanced Voice Services) 코딩 모드 분류기(320)의 모듈들 및 EVS(Enhanced Voice Services) 코딩 모드 분류 방법(300)의 동작들을 동시에 도시한 단순화된 상위 레벨 블록도(high-level block)이다.
- [0051] 도 3을 참조하면, 코딩 모드 분류 방법(300)은 활성 프레임 검출 동작(301), 무성음 프레임 검출 동작(302), 온셋 후 프레임 검출(frame after onset) 동작(303) 및 안정(stable) 유성음 프레임 검출 동작(304)을 구비한다.
- [0052] 활성 프레임 검출 동작(301)을 수행하기 위하여, 활성 프레임 검출기(31)는 현재 프레임이 활성인지 불활성인지 판정한다. 이를 위해, SAD(Sound Activity Detection) 또는 VAD(Voice Activity Detection)가 이용될 수 있다. 불활성 프레임이 검출되면, IC 코딩 모드(321)가 선택되고 절차가 종료된다.
- [0053] 검출기(311)가 활성 프레임 검출 동작(301) 동안에 활성 프레임을 검출하면, 무성음 프레임 검출기(312)를 이용하여 무성음 프레임 검출 동작(302)이 수행된다. 특히, 무성음 프레임이 검출되면, 무성음 프레임 검출기(312)는 검출된 무성음 프레임을 코딩하기 위해 UC 코딩 모드(322)를 선택한다. UC 코딩 모드는 무성음 프레임을 코딩하도록 고안된다. UC 코딩 모드에서는 적응적 코드북이 이용되지 않으며, 여기는 선형 가우시안 코드북(linear Gaussian codebook)으로부터 선택된 2개의 벡터들로 이루어진다. 대안적으로, UC에 있어서의 코딩 모드는 고정 대수 코드북(fixed algebraic codebook) 및 가우시안 코드북으로 이루어질 수 있다.
- [0054] 현재 프레임이 검출기(312)에 의해 무성음으로서 분류되지 않으면, 온셋 후 프레임 검출 동작(303) 및 대응하는 온셋 후 프레임(Frame after Onset) 검출기(313)와, 안정 유성음 프레임 검출 동작(304) 및 대응하는 안정 유성음 프레임 검출기(314)가 이용된다.
- [0055] 온셋 후 프레임 검출 동작(303)에서는, 검출기(313)가 유성음 온셋에 뒤이은 유성음 프레임들을 검출하고, TC 코딩 모드(323)를 선택하여 이 프레임들을 코딩한다. TC 코딩 모드(323)는 과거 정보(적응적 코드북)의 이용을 제한함으로써 프레임 소거의 존재시에 코덱 성능을 개선하도록 고안된다. 그와 동시에 (프레임 소거없는) 클린 채널 성능(clean channel performance)에 대한 TC 코딩 모드(323)의 영향을 최소화하기 위해, 프레임 소거 관점에서 가장 중요한 프레임들(most critical frames)에 대해서만 모드(323)가 이용된다. 가장 중요한 프레임들은 유성음 온셋에 뒤이은 유성음 프레임들이다.
- [0056] 현재 프레임이 유성음 온셋에 뒤이은 유성음 프레임이 아니면, 안정 유성음 프레임 검출 동작(304)이 수행된다. 이 동작 동안에, 안정 유성음 프레임 검출기(314)는 준-주기 안정 유성음 프레임을 검출하도록 고안된다. 현재 프레임이 준-주기 안정 유성음 프레임으로서 검출되면, 검출기(314)는 안정 유성음 프레임을 인코딩하기 위해 VC 코딩 모드(324)를 선택한다. 검출기(314)에 의한 VC 코딩 모드의 선택은 매끄러운 피치 에블루션(smooth

pitch evolution)에 의해 조절된다. 이것은 ACELP(Algebraic Code-Excited Linear Prediction) 기술을 이용하지만, 피치 에블루션이 프레임에 걸쳐 매끄러우면, GC 코딩 모드에서보다 보다 많은 비트들이 고정(대수) 코드북에 할당된다.

- [0057] 현재 프레임이 동작들(301-304) 동안에 상기 프레임 카테고리들 중 하나로 분류되지 않으면, 이 프레임은 아마도 비-정적 스피치 세그먼트(non-stationary speech segment)를 포함할 가능성이 있으며, 검출기(314)는 그러한 프레임의 인코딩을 위해, GC 코딩 모드(325), 예를 들어, 제네릭(generic) ACELP 코딩 모드를 선택한다.
- [0058] 마지막으로, 현재 프레임이 AC 모드를 이용하여 코딩될지를 결정하기 위해 EVS 표준의 스피치/뮤직 분류 알고리즘(speech/music classification algorithm, 도시되지 않음)이 운영된다. AC 모드는 제네릭 오디오 신호들, 특히, 배타적인 아니지만 뮤직을 효과적으로 코딩하도록 고안되었다.
- [0059] 잡음 채널에 대한 코덱 성능을 개선하기 위하여, FEC(Frame Error Concealment)을 위한 프레임 분류라고 하는, 도 3을 참조하여 이전 문단에서 설명한 코딩 모드 분류 방법의 개선(refinement)이 적용된다(참조 [4]). FEC를 위한 다른 프레임 분류 방식을 이용하는 배후의 기본적인 생각은, FEC에 대한 이상적인 전략이 신속한 변경 특성을 가진 스피치 세그먼트에 대해 및 준-정적 스피치 세그먼트에 대해 달라야 한다는 사실이다. EVS 표준에 있어서(참조 [4]), 인코더에서 이용된 FEC를 위한 프레임 분류는 다음과 같이 5개의 개별적인 클래스(class)를 정의한다. 무성음 클래스(UNVOICED class)는 활성 스피치없는 모든 프레임들 및 모든 무성음 스피치 프레임을 구비한다. 유성음 오프셋 프레임(voiced offset frame)은, 그의 중단이 무성음으로 되기 쉬우면, 무성음 클래스(UNVOICED class)로서 분류될 수 있다. 무성음 전이 클래스(UNVOICED TRANSITION class)는 프레임 중단에서 가능한 유성음 온셋을 가진 무성음 프레임을 구비한다. 유성음 전이 클래스(VOICED TRANSITION class)는 상대적으로 약한 유성음 특성을 가진 유성음 프레임들을 구비한다. 유성음 클래스(VOICED class)는 안정 특성을 가진 유성음 프레임들을 구비한다. 온셋 클래스(ONSET class)는 무성음 클래스(UNVOICED class) 또는 무성음 전이 클래스(UNVOICED TRANSITION class)로서 분류된 프레임에 뒤이은 안정 특성을 가진 모든 유성음 프레임들을 구비한다.
- [0060] 도 3의 EVS 코딩 모드 분류 방법(300) 및 EVS 코딩 모드 분류기(320)에 대한 추가적인 설명은, 예를 들어, 참조 [4]에서 발견될 수 있다.
- [0061] 원래, TC 코딩 모드는 전이 프레임이 상실된 경우에 에러 전파를 중지시키는데 도움을 주기 위해 전이에 뒤이은 프레임들에 이용되기 위해 도입되었다(참조 [4]). 추가적으로, TC 코딩 모드는 코딩 효율을 증가시키기 위해 전이 프레임들에 이용될 수 있다. 특히, 유성음 온셋 직전, 적응적 코드북은, 통상적으로, 유성음 세그먼트의 시작을 코딩하는데 있어서 매우 유용하거나 효율적이지는 않는 잡음형 신호를 포함한다. 목표는 유성음 온셋들을 인코딩하기 위해 성문음 임펄스 형상들의 단순화된 양자화 버전들로 채워진, 보다 나은 비-예측 코드북으로 적응적 코드북을 보완하는 것이다. 성문음-형상 코드북은, 프레임내에 제 1 성문음 임펄스를 포함하는 하나의 서브-프레임, 보다 구체적으로는, LP 잔차 신호(도 2에 있어서의  $s_n(n)$ )가 프레임의 제 1 피치 기간내에 최대 에너지를 가진 서브-프레임에만 이용된다. 도 3의 TC 코딩 모드에 대한 추가적인 설명은, 예를 들어, 참조 [4]에서 발견될 수 있다.
- [0062] 본 개시는 TC 코딩 모드의 성문음-형상 코드북을 이용하여 유성음 온셋들을 코딩하는 EVS 개념을 더욱 확장할 것을 제안한다. 프레임의 중단쯤에서 어택이 발생하면, 프레임의 중단쯤에서 여기를 코딩하기 위한 비트-예산(이용 가능한 비트들의 수)을 가능한 한 많이 이용하게 하는 것이 제안되는데, 이는, 적은 수의 비트들을 가진 프레임의 선행 부분(어택을 포함하는 서브-프레임들의 서브-프레임들)의 코딩이면 충분하기 때문이다. 참조 [4]에 설명된 EVS의 TC 코딩 모드와의 차이점은, 성문음-형상 코드북이, 프레임의 제 1 피치 기간내의 LP 잔차 신호의 실제 최대 에너지와 무관하게, 프레임내의 마지막 서브-프레임(들)에 이용된다는 것이다.
- [0063] 프레임의 종단을 인코딩하기 위한 비트-예산의 대부분이 이용되게 함에 의해, 프레임의 시작에 있는 사운드 신호의 파형은, 고정 코드북이, 예를 들어, 서브-프레임당 하나 또는 2개의 펄스로 이루어진 저 비트-레이트로 잘 모델링되지 않을 수 있다. 그러나, 인간 귀의 감도가 여기에 활용될 수 있다. 인간의 귀는 어택전의 사운드 신호의 부정확한 코딩에 많이 민감하지 않지만, 그러한 어택 후에, 예를 들어, 유성음 세그먼트와 같은 사운드 신호 세그먼트를 코딩하는데 있어서의 임의 결함에 대해서는 훨씬 더 민감하다. 어택을 구성하기 위해 많은 수의 비트들이 이용되게 함에 의해, 후속하는 사운드 신호 프레임들에 있어서의 적응적 코드북은 보다 효율적으로 되는데, 그 이유는 그것이 잘 모델링되는 어택 세그먼트에 대응하는 과거 여기로부터 이득을 취하기 때문이다. 결론적으로 주관적 품질이 개선된다.

[0064] 본 개시는, 이 프레임들이 TC 코딩 모드로 인코딩되어야 하는지를 판정하기 위해 GC 코딩 모드로 코딩될 프레임들에 대해 작용하는 어택을 검출하는 방법 및 그에 대응하는 어택 검출기를 제안한다. 특히, 어택이 검출되면, 이 프레임들은 TC 코딩 모드를 이용하여 코딩된다. 따라서, TC 코딩 모드를 이용하여 코딩된 프레임들의 상대적인 개수가 증가된다. 더욱이, TC 코딩 모드가 과거 여기를 이용하지 않음에 따라, 이 방식으로 프레임 소거에 대한 코덱의 본질적인 견고성이 증가된다.

[0065] 어택 검출 방법 및 어택 검출기

[0066] 도 4는 어택 검출 방법(400) 및 어택 검출기(450)의 모듈들의 동작들을 동시에 도시한 블록도이다.

[0067] 어택 검출 방법(400) 및 어택 검출기(450)은 TC 코딩 모드를 이용하여 코딩될 프레임들을 적절하게 선택한다. 이하의 설명은, 20ms의 길이를 가지며 4개의 서브-프레임들로 이루어진 프레임 및 12.8kbps의 내부 샘플링 레이트를 가진, 본 예시에 있어서 CELP 코덱과 같은 코덱에 이용될 수 있는, 어택 검출 방법(400) 및 어택 검출기(450)의 예시를 도 4와 관련하여 이루어진다. 예를 들어, 그러한 코덱은 저 비트-레이트( $\leq 13.2$ kbps)의 EVS 코덱(참조 [4])이다. 다른 내부 비트-레이트, 프레임 길이 및 서브-프레임들의 개수를 가진 다른 유형의 코덱에 대한 애플리케이션이 고려될 수 있다.

[0068] 어택의 검출은, 현재 프레임에 있어서의 입력 사운드 신호의 여러 세그먼트들의 에너지가 계산되는 전치 프로세싱과 함께 시작되고, 뒤이어, 연속적인 2 단계들로 수행되는 검출과 최종 결정이 이루어진다. 제 1 단계 검출은 현재 프레임에 있어서의 계산된 에너지를 비교하는 것에 기초하며, 제 2 단계 검출은 과거 프레임 에너지를 고려한다.

[0069] 세그먼트들의 에너지들

[0070] 도 4의 에너지 계산 동작(401)에 있어서, 에너지 계산기(451)는 인지 가중 입력 사운드 신호(perceptually weighted, input sound signal)  $s_w(n)$ 의 다수의 연속하는 분석 세그먼트들에 있어서의 에너지를 계산하며,  $n$ 은  $0, \dots, N-1$ 이고,  $N$ 은 샘플들에 있어서의 프레임의 길이이다. 그러한 에너지를 계산하기 위해, 계산기(451)는, 예를 들어, 다음의 수학적 식 (1)을 이용할 수 있다.

[0071] 
$$E_{seg}(i) = \sum_{k=0}^{K-1} s_w^2(i \cdot K + k), \quad i = 0, \dots, (N/K) - 1 \quad (1)$$

[0072]  $K$ 는 분석 사운드 신호 세그먼트의 샘플들에 있어서의 길이이고,  $i$ 는 세그먼트의 인덱스이고,  $N/K$ 는 세그먼트들의 전체 개수이다. 12.8kbps의 내부 샘플링 레이트로 동작하는 EVS 표준에 있어서, 프레임의 길이는  $N=256$  샘플들이고, 세그먼트의 길이는, 예를 들어,  $N/K=32$  분석 세그먼트들의 전체 개수로 결과하는,  $K=8$ 로 설정될 수 있다. 따라서, 세그먼트들  $i=0, \dots, 7$ 은 제 1 서브-프레임에 대응하고, 세그먼트들  $i=8, \dots, 15$ 는 제 2 서브-프레임에 대응하며, 세그먼트  $i=16, \dots, 23$ 은 제 3 서브-프레임에 대응하고, 마지막으로 세그먼트  $i=24, \dots, 31$ 은 현재 프레임의 마지막(4번째) 서브-프레임에 대응한다. 수학적 식 (1)의 비제한적 예시에 있어서, 세그먼트들은 연속적이다. 또 다른 가능한 실시 예에 있어서, 부분적으로 겹쳐진 세그먼트들이 채용될 수 있다.

[0073] 다음, 최대 에너지 세그먼트 탐지 동작(402)에서는, 최대 에너지 세그먼트 탐지기(452)가 최대 에너지를 가진 세그먼트  $i$ 를 탐지한다. 이를 위해, 탐지기(452)는, 예를 들어, 이하의 수학적 식 (2)를 이용할 수 있다.

[0074] 
$$I_{att} = \max_i (E_{seg}(i)), \quad i = 0, \dots, (N/K) - 1 \quad (2)$$

[0075] 최대 에너지를 가진 세그먼트는 이하의 2 단계(이하에서는 제 1 단계 및 제 2 단계)에서 인증되는 후보 어택의 위치를 나타낸다.

[0076] 본 설명에서 예시적인 주어진 예시적인 실시 예에 있어서, GC 코딩 모드를 이용하여 프로세싱되는 것으로 이전에 분류된 활성 프레임들(VAD=1, 국소적 VAD는 현재 프레임내에서 고려됨)에 대해서만 이하의 제 1 단계 및 제 2 단계 어택 검출이 이루어진다. VAC(Voice Activity Detection)에 대한 추가적인 설명은, 예를 들어, 참조 [4]에서 발견될 수 있다. 결정 동작(403)에 있어서, 결정 모듈(453)은, VAD=1 및 현재 프레임이 GC 코딩 모드를 이용하여 프로세싱되는 것으로 분류되었는지를 판정한다. 그렇다면, 현재 프레임에 대해 제 1 단계 어택 검출이 수행된다. 그렇지 않으면, 검출되는 어택은 없으며, 현재 프레임은 도 3에 도시된 바와 같이 이전 분류에 따라 프로세싱된다.

[0077] 스피치 및 뮤직 프레임들은 GC 코딩 모드로 분류될 수 있으며, 그러므로 스피치 신호들 및 일반적인 사운드 신호

호들을 코딩하는데 어택 검출이 적용된다.

[0078] 제 1 단계 어택 검출

[0079] 제 1 단계 어택 검출 동작(404)과 대응하는 제 1 단계 어택 검출기(454)를 도 4를 참조하여 설명하겠다.

[0080] 제 1 단계 어택 검출 동작(404)은 평균 에너지 계산 동작(405)을 구비한다. 동작(405)을 수행하기 위하여, 제 1 단계 어택 검출기(454)는, 예를 들어, 이하의 수학적 식 (3)을 이용하여 현재 프레임내의 마지막 서브-프레임 전의 분석 세그먼트들에 걸쳐서의 평균 에너지의 계산기(455)를 구비한다.

[0081] 
$$E_1 = \frac{1}{P} \sum_{i=0}^{P-1} E_{seg}(i) \quad (3)$$

[0082] P는 마지막 서브-프레임 전의 세그먼트들의 개수이다. 비 제한적인 예시적 구현에 있어서, N/K=32이고, 파라메타 P는 24이다.

[0083] 유사하게, 평균 에너지 계산 동작(405)에 있어서, 계산기(455)는 예를 들어 이하의 수학적 식 (4)를 이용하여, 세그먼트  $I_{att}$ 에서 현재 프레임의 마지막 프레임까지의 분석 세그먼트들에 걸쳐서의 평균 에너지를 계산한다.

[0084] 
$$E_2 = \frac{1}{(N/K) - I_{att}} \sum_{i=I_{att}}^{(N/K)-1} E_{seg}(i) \quad (4)$$

[0085] 제 1 단계 어택 검출 동작(404)은 비교 동작(406)을 추가로 구비한다. 비교 동작(406)을 수행하기 위해, 제 1 단계 어택 검출기(454)는, 수학적 식 (3)으로부터의 평균 에너지  $E_1$ 과 수학적 식 (4)로부터의 평균 에너지  $E_2$ 의 비율을, 상기에서 설명한 FEC(Frame Error Concealment)을 위한 프레임 분류(참조 [4])에 의해 수행되는, "last\_class"로서 표시된, 이전 프레임의 신호 분류에 의존하는 임계치와 비교하는 비교기(456)를 구비한다. 비교기(456)는, 비 제한적 예시로서, 이하의 수학적 식 (5)의 로직을 이용하여, 제 1 단계 어택 검출로부터의 어택 위치  $I_{att1}$ 을 판정한다.

[0086] 
$$\left\{ \left( \frac{E_2}{E_1} < \beta_1 \right) \text{ OR } \left( \left( \frac{E_2}{E_1} < \beta_2 \right) \text{ AND } (\text{last\_class} = \text{VOICED}) \right) \right\}$$
 이면,

[0087]  $I_{att1} = 0$  이고,

[0088] 그렇지 않으면  $I_{att1} = I_{att}$  (5)

[0089]  $\beta_1$  및  $\beta_2$ 는, 비 제한적 예시에 따라, 각각,  $\beta_1=8$  및  $\beta_2=20$ 으로 설정될 수 있는 임계치들이다.  $I_{att1}=0$ 이면, 검출되는 어택은 없다. 수학적 식 (5)의 로직을 이용하면, 충분히 강하지는 않는 모든 어택들이 제거된다.

[0090] 잘못 검출된 어택들의 개수를 추가로 감소시키기 위해, 제 1 단계 어택 검출 동작(404)은, 세그먼트 에너지 비교 동작(407)을 더 구비한다. 세그먼트 에너지 비교 동작(407)을 수행하기 위하여, 제 1 단계 어택 검출기(454)는 현재 프레임의 다른 분석 세그먼트의 에너지  $E_{seg}(i)$ 와 최대 에너지  $E_{seg}(I_{att})$ 를 가진 세그먼트를 비교하는 세그먼트 에너지 비교기(457)를 구비한다. 동작(406)과 비교기(456)에 의해  $I_{att} > 0$ 으로 판정되면, 비교기(457)는, 비 제한적 예시로서,  $i=2, \dots, P-3$ 에 대해 수학적 식 (6)의 비교를 수행한다.

[0091] 
$$\left\{ \frac{E_{seg}(I_{att})}{E_{seg}(i)} < \beta_3 \right\}$$
 이면,  $I_{att1} = 0$  (6)

[0092] 임계치  $\beta_3$ 은 진실한 어택의 검출 효율을 방해하지 않은 채, 잘못 검출된 어택들을 가능한 한 많이 줄이도록 실험적으로 결정된다. 비 제한적인 실험적 구현에 있어서, 임계치  $\beta_3$ 은 2로 설정된다. 다시,  $I_{att1}=0$ 이면 검출되는 어택은 없다.

[0093] 제 2 단계 어택 검출

[0094] 제 2 단계 어택 검출 동작(410) 및 대응하는 제 2 단계 어택 검출기(460)를 도 4를 참조하여 설명하겠다.

[0095] 제 2 단계 어택 검출 동작(410)은 유성음 클래스 비교 동작(411)을 구비한다. 유성음 클래스 비교 동작(411)을 수행하기 위하여, 제 2 단계 어택 검출기(460)는 현재 프레임 클래스가 유성음(VOICED)인지 여부를 판정하기 위해 상술한 EVS FEC 분류 방법으로부터 정보를 얻도록 유성음 클래스 결정 모듈(461)을 구비한다. 현재 프레임 클래스가 유성음이면, 결정 모듈(461)은 검출되는 어택이 없다는 결정을 출력한다.

[0096] 제 1 단계 어택 검출 동작(404) 및 제 1 단계 어택 검출기(454)(특히, 비교 동작(406) 및 비교기(456) 또는 비교 동작(407) 및 비교기(457))에서 어택이 검출되지 않았고, 즉,  $I_{att1}=0$ 이고, 현재 프레임의 클래스가 유성음과 다른 것이면, 제 2 단계 어택 검출 동작(410) 및 제 2 단계 어택 검출기(460)가 적용된다.

[0097] 제 2 단계 어택 검출 동작(410)은 평균 에너지 계산 동작(412)을 구비한다. 동작(412)을 수행하기 위해, 제 2 단계 어택 검출기(460)는, 예를 들어, 수학적 식 (7)을 이용하여 (이전 프레임으로부터의 세그먼트들을 포함하는) 후보 어택  $I_{att}$  전의 N/K 분석 세그먼트들에 걸쳐서의 평균 에너지를 계산하는 평균 에너지 계산기(462)를 구비한다.

[0098] 
$$E_{mean} = \frac{1}{N/K} \left( \sum_{i=I_{att}}^{(N/K)-1} E_{seg,past}(i) + \sum_{i=0}^{I_{att}-1} E_{seg}(i) \right) \quad (7)$$

[0099]  $E_{seg,past}(i)$ 는 이전 프레임으로부터의 세그먼트들당 에너지들이다.

[0100] 제 2 단계 어택 검출 동작(410)은 로직 결정 동작(413)을 구비한다. 동작(413)을 수행하기 위하여, 제 2 단계 어택 검출기(460)는, 예를 들어, 수학적 식 (7)로부터의 평균 에너지에 수학적 식 (8)의 이하의 로직을 적용함에 의해, 제 2 단계 어택 검출기로부터의 어택 위치  $I_{att2}$ 를 탐지하기 위한 로직 결정 모듈(463)을 구비한다.

[0101]  $\left\{ \left( \frac{E_{seg}(I_{att})}{E_{mean}} > \beta_4 \right) \text{ OR } \left( \left( \frac{E_{seg}(I_{att})}{E_{mean}} > \beta_5 \right) \text{ AND } (\text{last\_class} = \text{UNVOICED}) \right) \right\}$  이면,

[0102]  $I_{att2} = I_{att}$  이고,

[0103] 그렇지 않으면,  $I_{att2} = 0$  (8)

[0104]  $I_{att}$ 는 수학적 식 (2)에서 발견되었고,  $\beta_4$  및  $\beta_5$ 는, 본 비-제한적인 예시적 구현에서, 각각  $\beta_4=16$  및  $\beta_5=12$ 로 설정되는 임계치들이다. 비교 동작(413) 및 비교기(463)는, 검출되는 어택이 없는,  $I_{att2} = 0$ 을 판정한다.

[0105] 제 2 단계 어택 검출 동작(410)은, 최종적으로, 에너지 비교 동작(414)을 구비한다. 동작(414)을 수행하기 위해, 제 2 단계 어택 검출기(460)는, 비교 동작(413) 및 비교기(463)에서 판정된  $I_{att}$ 가 0보다 클때, 잘못 검출된 어택들의 개수를 추가로 줄이기 위하여, 예를 들어 수학적 식 (9)에 보여진 이하의 임계치와 이하의 비율을 비교하기 위한 에너지 비교기(464)를 구비한다.

[0106]  $\left\{ \frac{E_{seg}(I_{att})}{E_{LT}} < \beta_6 \right\}$  이면,  $I_{att2} = 0$  (9)

[0107] 본 비-제한적인 예시적 구현에 있어서,  $\beta_6$ 은  $\beta_6=20$ 으로 설정된 임계치이고,  $E_{LT}$ 는, 비-제한적 예시로서, 수학적 식 (10)을 이용하여 계산된 장기 에너지(long-term energy)이다.

[0108] 
$$E_{LT} = \alpha \cdot E_{LT} + (1-\alpha) \cdot \frac{1}{N/K} \sum_{i=0}^{N/K-1} E_{seg}(i). \quad (10)$$

[0109] 본 비-제한적인 예시적 구현에 있어서, 파라메타  $\alpha$ 는 0.95로 설정된다. 다시,  $I_{att2} = 0$ 일 경우, 검출되는 어택은 없다.

[0110] 마지막으로, 에너지 비교 동작(414)에 있어서, 이전 프레임에서 어택이 검출되었으면, 에너지 비교기(464)는 어택 위치  $I_{att2}$ 를 0으로 설정한다. 이 경우, 검출되는 어택은 없다.

[0111] 최종 어택 검출 결정

[0112] 현재 프레임이 TC 코딩 모드를 이용하여 코딩될 어택 프레임으로서 판정되는지의 최종 결정은, 각각, 제 1 단계 (404) 및 제 2 단계(410) 검출 동작 동안에 획득한 어택의 위치들  $I_{att1}$  및  $I_{att2}$ 에 기초하여 수행된다.

[0113] 현재 프레임이 활성화(VAD=1)이고, 결정 동작(403)과 결정 모듈(453)에서 결정된 바와 같이 GC 코딩 모드에서 코딩하는 것으로 이전에 분류되었으면, 예를 들어, 수학적 식 (11)의 이하의 로직이 적용된다.

[0114]  $I_{att} \geq P$ 이면,  $I_{att,final} = I_{att1}$ 이고,  $I_{att2} > 0$ 이면,  $I_{att,final} = I_{att2}$  (11)

[0115] 특히, 어택 검출 방법(400)은 제 1 단계 어택 결정 동작(430)을 구비한다. 동작(430)을 수행하기 위해, 현재 프레임이, 활성화(VAD=1)이고, 결정 동작(403) 및 결정 모듈(453)에서 판정된 바와 같이 GC 코딩 모드에서 코딩하는 것으로 이전에 분류되었으면, 어택 검출기(450)는  $I_{att1} \geq P$ 인지를 판정하기 위해 제 1 단계 어택 결정 모듈(470)을 추가로 구비한다.  $I_{att1} \geq P$ 이면,  $I_{att1}$ 은 현재 프레임의 마지막 서브-프레임에 있어서의 검출된 어택의 위치  $I_{att1,final}$ 로서, TC 코딩 모드의 성문음 형상 코드북이 이러한 마지막 서브-프레임에 이용됨을 판정하는데 이용된다. 그렇지 않을 경우, 검출되는 어택은 없다.

[0116] 제 2 단계 어택 검출과 관련하여, 수학적 식 (9)의 비교가 참이거나, 어택이 에너지 비교 동작(414)과 에너지 비교기(464)에서 판정된 바와 같이 이전 프레임에서 검출되었으면,  $I_{att2}=0$ 이고 검출되는 어택은 없다. 그렇지 않을 경우, 어택 검출 방법(400)의 어택 결정 동작(440)에 있어서, 어택 검출기(450)의 어택 결정 모듈(480)은, 어택이 위치  $I_{att,final} = I_{att2}$ 에 있는 현재 프레임에서 검출된다고 판정한다. 검출된 어택의 위치  $I_{att,final}$ 는 TC 코딩 모드의 성문음-형상 코드북이 어느 서브-프레임에서 이용되는지를 판정하는데 이용된다.

[0117] 검출된 어택의 최종 위치  $I_{att,final}$ 에 대한 정보는 TC 코딩 모드내의 성문음-형상 코드북이 현재 프레임의 어느 서브-프레임에 채용되는지 및 어느 TC 모드 구성(참조 [3])이 이용되는지 판정하는데 이용된다. 예를 들어, 4개의 서브-프레임들과  $N/K=32$  분석 세그먼트들로 분할된,  $N=256$  샘플들의 프레임의 경우에 있어서, 성문음-형상 코드북은, 최종 어택 위치  $I_{att,final}$ 가 세그먼트들 1-7에서 검출되면 제 1 서브-프레임에 이용되고, 최종 어택 위치  $I_{att,final}$ 가 세그먼트들 8-15에서 검출되면 제 2 서브-프레임에 이용되며, 최종 어택 위치  $I_{att,final}$ 가 세그먼트들 16-23에서 검출되면 제 3 서브-프레임에 이용되고, 마지막으로, 최종 어택 위치  $I_{att,final}$ 가 세그먼트들 24-31에서 검출되면 현재 프레임의 마지막(제 4) 서브-프레임에 이용된다. 값  $I_{att,final}=0$ 은, 어택이 발견되지 않았으며, 현재 프레임이 (통상적으로 GC 코딩 모드를 이용하는) 원래의 분류에 따라 코딩됨을 시사한다.

[0118] 몰입형 보이스/오디오 코덱(immersive voice/audio codec)에 있어서의 예시적인 구현

[0119] 어택 검출 방법(400)은 성문음-형상 코드북 할당 동작(445)을 구비한다. 동작(445)을 수행하기 위해, 어택 검출기(450)는, 이하의 수학적 식 (12)의 로직을 이용하여, 4개의 서브-프레임로 이루어진 현재 프레임의 주어진 서브-프레임에 TC 코딩 모드내의 성문음-형상 코드북을 할당하기 위한 성문음-형상 코드북 할당 모듈(485)를 구비한다.

[0120] 
$$sbfr = 4 \cdot \frac{I_{att,final}}{N / K} \quad (12)$$

[0121] sbfr은 서브-프레임 인덱스로서,  $sbfr = 0, \dots, 3$ 이고, 인덱스 0은 제 1 서브-프레임을 나타내고, 인덱스 1은 제 2 서브-프레임을 나타내며, 인덱스 2는 제 3 서브-프레임을 나타내고, 인덱스 3은 제 4 서브-프레임을 나타낸다.

[0122] 비-제한적 예시적 구현의 이전 설명은 12.8kHz의 내부 샘플링 레이트로 동작하고, 4개의 서브-프레임들 및 그에 따라  $N=256$ 의 샘플링 개수를 가진 프레임을 가진 전치-프로세싱 모듈을 상정한다. 코어 코덱이 12.8kHz의 내부 샘플링 레이트로 동작하는 ACELP를 이용하면, 최종 어택 위치  $I_{att,final}$ 은 수학적 식 (12)에서 정의된 바와 같은 서브-프레임에 할당된다. 그러나, 코어 코덱이 다른 내부 샘플링 레이트, 예를 들어, 내부 샘플링 레이트가 16kHz인, 보다 높은 비트-레이트(EVS의 경우에는 16.4kbps 이상)로 동작할 때에는 상황이 다르다. 프레임 길이가 20ms인 경우, 그 프레임은 이 경우에 5개의 서브-프레임들로 구성되며, 그러한 프레임의 길이는  $N_{16}=320$  샘플들이다. 구현의 이러한 예시에서는, 전치-프로세싱 분류 및 분석이 여전히 12.8kHz의 내부 샘플링 레이트의 도메인에서 수행되기 때문에, 성문음-형상 코드북 할당 모듈(485)은, 성문음-형상 코드북 할당 동작(445)에서, 이하

의 수학적 식 (13)의 로직을 이용하여 TC 코딩 모드내의 성문음-형상 코드북을 이용하여 코딩될 서브-프레임을 선택한다.

[0123] 
$$sbfr = \left\lfloor 5 \cdot \frac{I_{att,final}}{N/K} \right\rfloor \quad (13)$$

[0124] 연산자  $\lfloor x \rfloor$ 은  $x$  이하의 최대 정수를 나타낸다. 수학적 식 (13)의 경우,  $sbfr = 0, \dots, 4$ 는 수학적 식 (12)와 다르지만, 분석 세그먼트들의 개수는 수학적 식 (12)에서와 동일하다. 즉,  $N/K = 32$ 이다. 따라서, 성문음-형상 코드북은, 최종 어택 위치  $I_{att,final}$ 가 세그먼트들 1-6에서 검출되면 제 1 서브-프레임에 이용되고, 최종 어택 위치  $I_{att,final}$ 가 세그먼트들 7-12에서 검출되면 제 2 서브-프레임에 이용되며, 최종 어택 위치  $I_{att,final}$ 가 세그먼트들 13-19에서 검출되면 제 3 서브-프레임에 이용되고, 최종 어택 위치  $I_{att,final}$ 가 세그먼트들 20-25에서 검출되면 제 4 서브-프레임에 이용되며, 최종 어택 위치  $I_{att,final}$ 가 세그먼트들 26-31에서 검출되면 현재 프레임의 마지막(제 5) 서브-프레임에 이용된다.

[0125] 도 5는 디코딩된 뮤직 신호의 품질에 대한 TC 코딩 모드 및 도 4의 어택 검출기의 영향을 보여주는 제 1의 비제한적 예시의 그래프이다. 특히, 도 5에는 캐스터네즈들의 뮤직 세그먼트가 도시되는데, 거기에서, 곡선 a)는 입력 (코딩되지 않은) 뮤직 신호를 나타내고, 곡선 b)는 단지 제 1 단계 어택 검출만이 채용되었을 경우의 디코딩된 참조 신호 합성을 나타내며, 곡선 c)는 전체적인 제 1 단계 및 제 2 단계 어택 검출과 TC 코딩 모드를 이용한 코딩이 채용될 경우의 디코딩되고 개선된 합성을 나타낸다. 곡선 b)와 c)를 비교하면, 온셋들의 시작에서의 캐스터네즈 신호들의 에너지 및 예리함(sharpness)을 유지하는 견지에서, 곡선 c)의 합성에 있어서의 어택들(도 5의 500과 같은 저-고 진폭 온셋들)이 보다 정확하고 크게 재구성됨을 알 수 있을 것이다.

[0126] 도 6은 디코딩된 스피치 신호의 품질에 대한 TC 코딩 모드 및 도 4의 어택 검출기의 영향을 보여주는 제 2의 비제한적 예시의 그래프로서, 곡선 a)는 입력 (코딩되지 않은) 스피치 신호를 나타내고, 곡선 b)는 온셋 프레임이 GC 코딩 모드를 이용하여 코딩될 경우의 디코딩된 참조 스피치 신호 합성을 나타내며, 곡선 c)는 전체적인 제 1 단계 및 제 2 단계 어택 검출과 TC 코딩 모드를 이용한 코딩이 온셋 프레임에 채용될 경우의 디코딩되고 개선된 스피치 신호 합성을 나타낸다. 곡선 b)와 c)를 비교하면, 어택들(도 6의 600과 같은 저-고 진폭 온셋들)의 코딩은, 어택 검출 동작(400) 및 어택 검출기(450)와 TC 코딩 모드가 온셋 프레임에 채용될 때, 개선됨을 알 수 있을 것이다. 또한, 온셋 후 프레임은 곡선 b) 및 c)에 있어서 GC 코딩 모드를 이용하여 코딩되며, 온셋 후 프레임의 코딩 품질은 곡선 c)에서 개선됨을 알 수 있을 것이다. 이것은, 온셋 프레임이 TC 코딩 모드를 이용하여 코딩될 때, 온셋 후 프레임에 있어서의 GC 코딩 모드의 적응적 코드북이, 잘 구축된 여기를 활용하기 때문이다.

[0127] 도 7은 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 디바이스들을 형성하고 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들을 구현한 하드웨어 부품들의 예시적인 구성의 단순화된 블럭도이다.

[0128] 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 디바이스들은 이동 단말의 일부로서, 휴대용 매체 플레이어의 일부로서 또는 임의 유사한 디바이스에 구현될 수 있다. 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 디바이스들(도 7에서 700으로 표시됨)은 입력(702), 출력(704), 프로세서(706) 및 메모리(708)를 구비한다.

[0129] 입력(702)은, 예를 들어, 디지털 입력 사운드 신호(105)를 수신하도록 구성된다(도 1). 출력(704)은 인코딩된 비트-스트림(111)을 공급하도록 구성된다. 입력(702)과 출력(704)은 공통 모듈, 예를 들어, 직렬 입력/출력 디바이스로 구현될 수 있다.

[0130] 프로세서(706)는 입력(702), 출력(704) 및 메모리(708)에 동작 가능하게 접속된다. 프로세서(706)는 도 2, 3 및 4의 모듈들을 포함하는, 사운드 인코더(106)의 여러 모듈들의 기능을 지원하여 코드 명령어들을 실행하는 하나 이상의 프로세서들로서 실현된다.

[0131] 메모리(708)는 프로세서(706)에 의해 실행될 수 있는 코드 명령어들을 저장하는 비-일시적 메모리, 특히, 실행 시에, 프로세서가 도 2, 3 및 4의 동작들과 모듈들을 포함하는, 사운드 인코더(106)의 동작들과 모듈들을 구현하게 하는 비-일시적 명령어들을 구비하는 프로세서-관독 가능 메모리를 구비할 수 있다. 메모리(708)는, 프로

세서(706)에 의해 수행되는 여러 기능들로부터 중간 프로세싱 데이터를 저장하기 위해 랜덤 액세스 메모리 또는 버퍼(들)를 구비할 수 있다.

- [0132] 본 기술 분야의 숙련자라면, 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들의 설명이 단지 예시적인 것이며, 임의 방식으로 제한하기 위한 것은 아님을 알 것이다. 다른 실시 예들은 본 개시의 혜택을 받은 본 기술 분야의 숙련자에게 쉽게 제안될 것이다. 또한, 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 개시된 방법들 및 디바이스들은 비트-예산의 할당 또는 분배와 관련된 기존의 필요성 및 문제점들에 대한 가치있는 해법을 제공하도록 맞춤화될 수 있다.
- [0133] 명확성을 위해, 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들의 구현들의 일상적인 특징들의 전부가 도시되고 설명된 것은 아니다. 물론, 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들의 임의의 그러한 실제적 구현의 개발에 있어서, 애플리케이션, 시스템, 네트워크 및 사업 관련 제약의 준수와 같은, 개발자의 특정 목표를 달성하기 위해 많은 구현 특정적 결정들이 이루어질 필요가 있으며, 이들 특정 목표들은 구현마다 가변될 것이고, 개발자마다 가변될 것임을 알 것이다. 또한, 개발 노력은 복잡하고 시간-소모적이지만, 그럼에도 본 개시의 혜택을 본 사운드 프로세싱 분야의 숙련자에게는 엔지니어링의 일상적인 처리임을 알 것이다.
- [0134] 본 개시에 따르면, 본 명세서에 설명된 모듈들, 프로세싱 동작들 및/또는 데이터 구조들은 다양한 유형의 운영 시스템들, 컴퓨팅 플랫폼들, 네트워크 디바이스들, 컴퓨터 프로그램들 및/또는 범용 기계들을 이용하여 구현될 수 있다. 또한, 본 기술 분야의 숙련자라면, 하드웨어 디바이스들, FPGA(Field Programmable Gate Array)들, ASIC(Application Specific Integrated Circuit)들 등과 같은 보다 덜 범용적인 디바이스들이 이용될 수도 있음을 알 것이다. 일련의 동작들 및 서브-동작들을 구비한 방법이 프로세서, 컴퓨터 또는 기계에 의해 구현되고, 이들의 동작과 서브-동작들이 프로세서, 컴퓨터 또는 기계에 의해 관독 가능한 일련의 비-일시적 코드 명령어들로서 저장되면, 그들은 유형의 및/또는 비-일시적인 매체상에 저장될 수 있다.
- [0135] 본 명세서에서 설명한 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들의 모듈들은, 본 명세서에서 설명한 목적에 적당한 하드웨어, 소프트웨어 또는 펌웨어의 임의 조합들, 펌웨어 또는 소프트웨어를 구비할 수 있다.
- [0136] 본 명세서에서 설명한 코딩될 사운드 신호에 있어서의 어택을 검출하고 검출된 어택을 코딩하는 방법들 및 디바이스들에 있어서, 여러 동작들 및 서브-동작들은 다양한 순서로 수행될 수 있으며, 그 동작들 및 서브-동작들의 일부는 선택적일 수 있다.
- [0137] 본 개시가 비 제한적이고 예시적인 실시 예들로 이루어지지만, 이 실시 예들은 본 개시의 사상 및 본질을 벗어나지 않은 채 첨부된 청구범위의 범주내에서 마음대로 수정될 수 있다.
- [0138] **참조들**
- [0139] 이하의 참조들은 본 명세서에서 언급되며, 그의 전체 콘텐츠는 본 명세서에 참조로서 인용된다.
- [0140] [1] V. Eksler, R. Salami, and M. Jelinek, "Efficient handling of mode switching and speech transitions in the EVS codec," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Brisbane, Australia, 2015.
- [0141] [2] V. Eksler, M. Jelinek, and R. Salami, "Method and Device for the Encoding of Transition Frames in Speech and Audio," WIPO Patent Application No. WO/2008/049221, 24 Oct. 2006.
- [0142] [3] V. Eksler and M. Jelinek, "Glottal-Shape Codebook to Improve Robustness of CELP Codecs," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 18, no. 6, pp. 1208 - 1217, Aug. 2010.
- [0143] [4] 3GPP TS 26.445: "Codec for Enhanced Voice Services (EVS); Detailed Algorithmic Description".
- [0144] 추가적인 개시로서, 아래는 몰입형 보이스 및 오디오 서비스(Immersive Voice and Audio Service: IVAS) 코덱에 구현된 개시된 어택 검출기의 비-제한적 예시의 의사-코드이다.

[0145] 의사-코드는 EVS에 기반한다. 새로운 IVAS 로직인 음영 배경에 하이라이트된다.

```

void detector( . . . )
{
    attack_flag = 0;          /* initialization */

    attack = attack_det( . . . ); /* attack detection */

    . . .

    if (localVAD == 1 && *coder_type == GENERIC && attack > 0 &&
        !(*sp_aud_decision2 == 1 && ton > 0.65f))
    {
        /* change coder_type to TC if attack has been detected */
        *sp_aud_decision1 = 0;
        *sp_aud_decision2 = 0;
        *coder_type = TRANSITION;
        *attack_flag = attack + 1;
    }

    return attack_flag;
}

static short attack_det(
    const float *inp,          /* i : input signal          */
    const short last_clas,    /* i : last signal clas      */
    const short localVAD,    /* i : local VAD flag       */
    const short coder_type,   /* i : coder type           */
    const long total_brate,   /* i : total bit-rate       */
    const short element_mode, /* i : IVAS element mode    */
    const short clas,        /* i : signal class         */
    float finc_prev[],       /* i/o: previous finc       */
    float *lt_finc,         /* i/o: long-term mean finc */
    short *last_strong_attack /* i/o: last strong attack flag */
)
{

```

[0146]

```

short i, attack;
float etmp, etmp2, finc[ATT_NSEG];
short att_3lsub_pos;
short attack1;

att_3lsub_pos = ATT_3LSUB_POS;
if( total_brate >= ACELP_24k40 )
{
    att_3lsub_pos = ATT_3LSUB_POS_16k; /* applicable only in EVS */
}

/* compute energy per section */
for( i=0; i<ATT_NSEG; i++ )
{
    finc[i] = sum2_f( inp + i*ATT_SEG_LEN, ATT_SEG_LEN );
}

attack = maximum( finc, ATT_NSEG, &etmp );
attack1 = attack;

if( localVAD == 1 && coder_type == GENERIC )
{
    /* compute mean energy in the first three sub-frames */
    etmp = mean( finc, att_3lsub_pos );

    /* compute mean energy after the attack */
    etmp2 = mean( finc + attack, ATT_NSEG - attack );

    /* and compare them */
    if( etmp * 8 > etmp2 )
    {
        /* stop, if the attack is not sufficiently strong */
        attack = 0;
    }

    if( last_clas == VOICED_CLAS && etmp * 20 > etmp2 )
    {
        /* stop, if the signal was voiced and the attack is not
sufficiently strong*/
        attack = 0;
    }

    /* compare wrt. other sections (reduces miss-classification) */
    if( attack > 0 )
    {
        etmp2 = finc[attack];

        for( i=2; i<att_3lsub_pos-2; i++ )
        {
            if( finc[i] * 2.0f > etmp2 )
            {
                /* stop, if the attack is not sufficiently strong *,
                attack = 0;
                break;
            }
        }
    }

    if( attack == 0 && element_mode > EVS_MONO && (clas <
VOICED_TRANSITION || clas == ONSET) )
    {
        mvr2r( finc, finc_prev, attack1 );
    }
}

```

[0147]

```

/* compute mean energy before the attack */
etmp = mean( finc_prev, ATT_NSEG );

etmp2 = finc[attack1];

if((etmp * 16 < etmp2) || (etmp * 12 < etmp2 && last_clas ==
UNVOICED_CLAS))
{
    attack = attack1;
}

if( 20 * *lt_finc > etmp2 || *last_strong_attack )
{
    attack = 0;
}
}
*last_strong_attack = attack;
}

/* compare wrt. other sections (reduces miss-classification) */
else if( attack > 0 )
{
    etmp2 = finc[attack];

    for( i=2; i<att_3lsub_pos-2; i++ )
    {
        if( i != attack && finc[i] * 1.3f > etmp2 )
        {
            /* stop, if the attack is not sufficiently strong */
            attack = 0;
            break;
        }
    }
    *last_strong_attack = 0;
}

/* updates */
mvr2r( finc, finc_prev, ATT_NSEG );
*lt_finc = 0.95f * *lt_finc + 0.05f * mean( finc, ATT_NSEG );

return attack;
}

/* function to determine the sub-frame with glottal-shape codebook in TC mode
frame */
void tc_classif_enc(
    const short L_frame,          /* i : length of the frame          */
    short *tc_subfr,             /* o : TC sub-frame index          */
    short *position,             /* o : maximum of residual signal index */
    const short attack_flag,     /* i : attack flag                  */
    const short T_op[],          /* i : open loop pitch estimates    */
    const float *res             /* i : LP residual signal           */
)
{
    float temp;

    *tc_subfr = -1;
    if( attack_flag )
    {
        *tc_subfr = 3*L_SUBFR;

        if( attack_flag > 0 )
        {
            if( L_frame == L_FRAME )
            {
                *tc_subfr = NB_SUBFR * (attack_flag-1) / 32 /*ATT_NSEG*/;
            }
            else
            {
                *tc_subfr = NB_SUBFR16k * (attack_flag-1) / 32 /*ATT_NSEG*/;
            }
            *tc_subfr *= L_SUBFR;
        }
    }

    if( attack_flag )
    {
        *position = emaximum( res + *tc_subfr, min(T_op[0]+2, L_SUBFR), &temp )
+ *tc_subfr;
    }
    else
    . . .

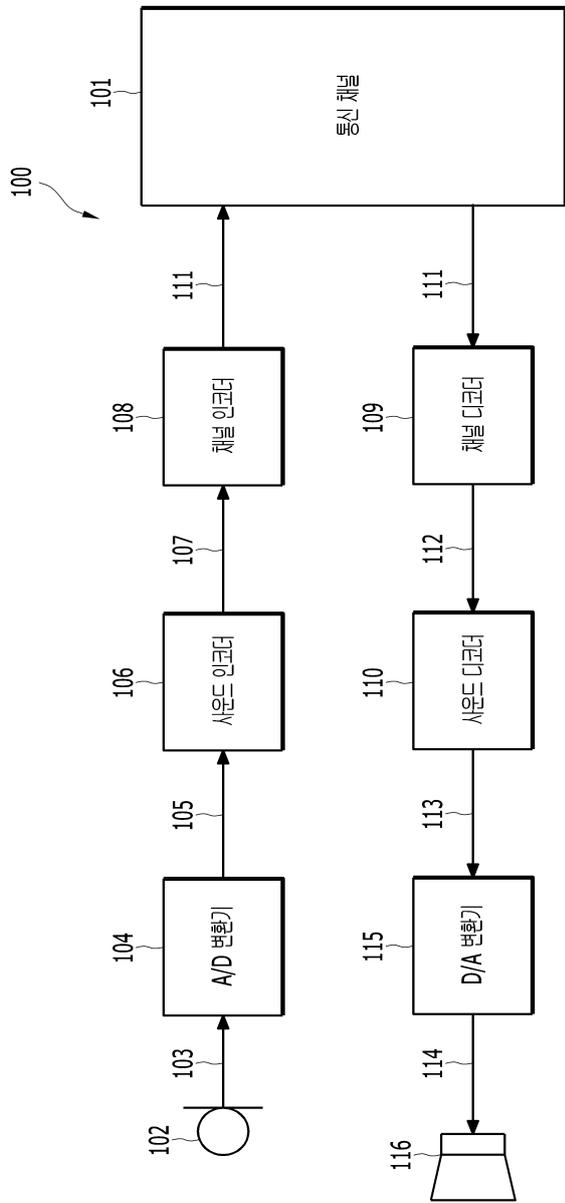
```

[0148]

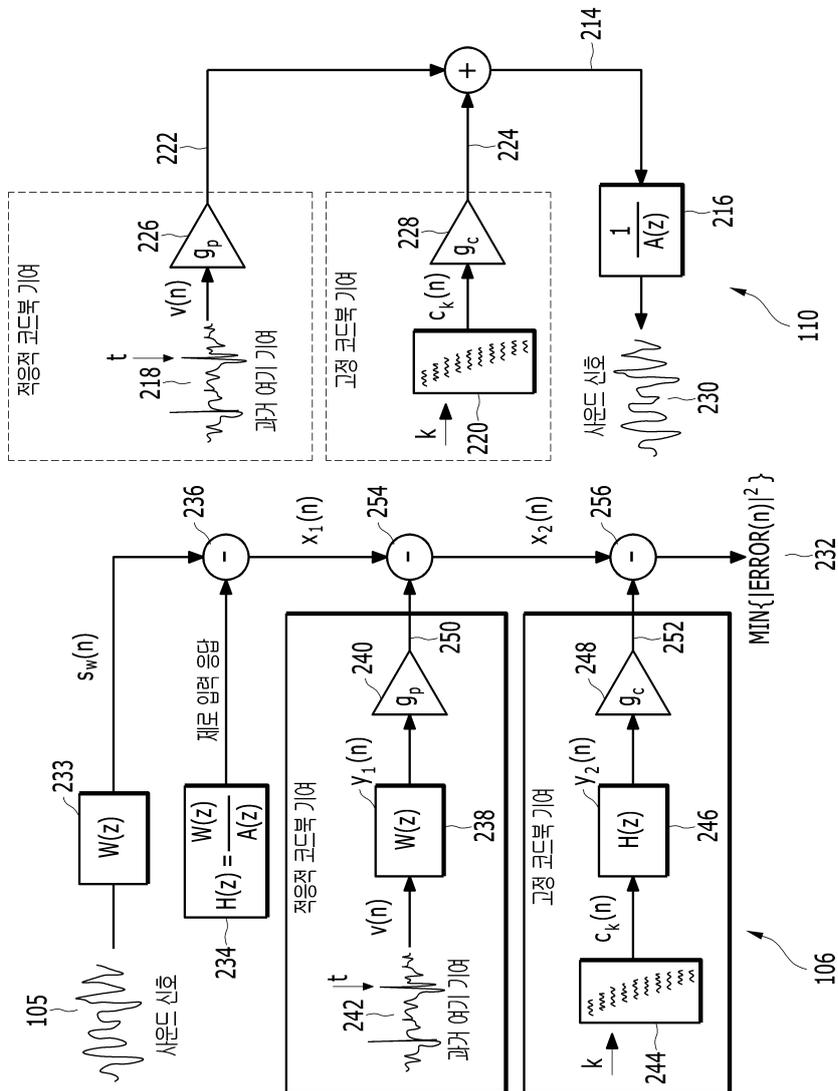
[0149]

도면

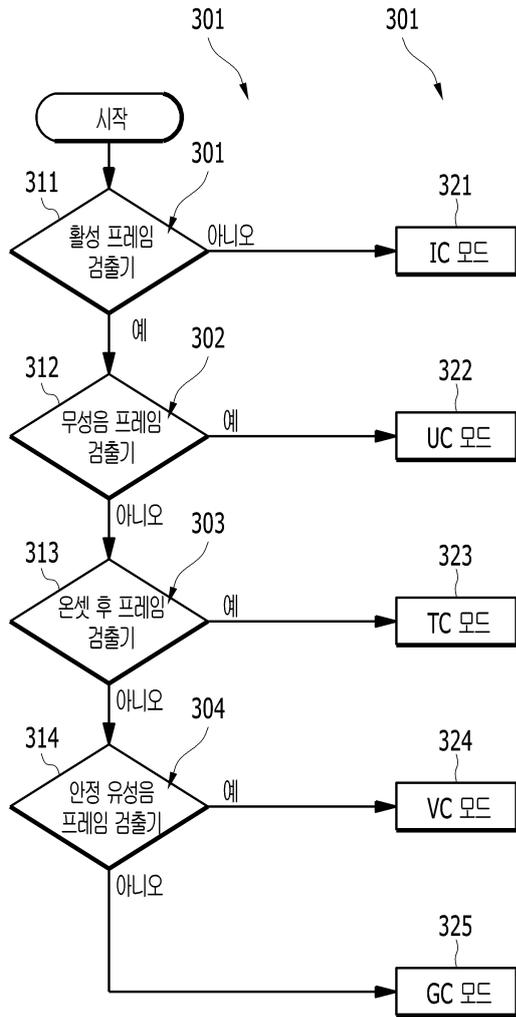
도면1



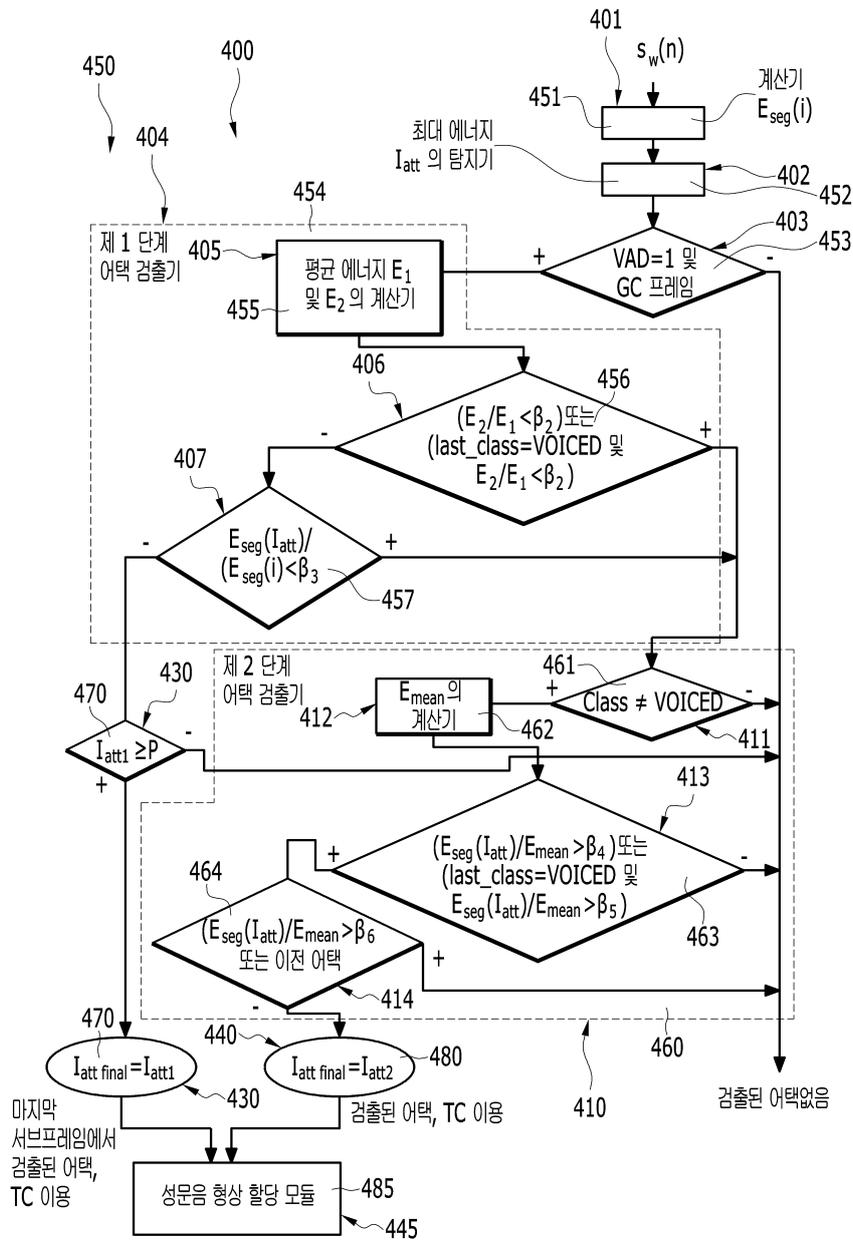
도면2



도면3



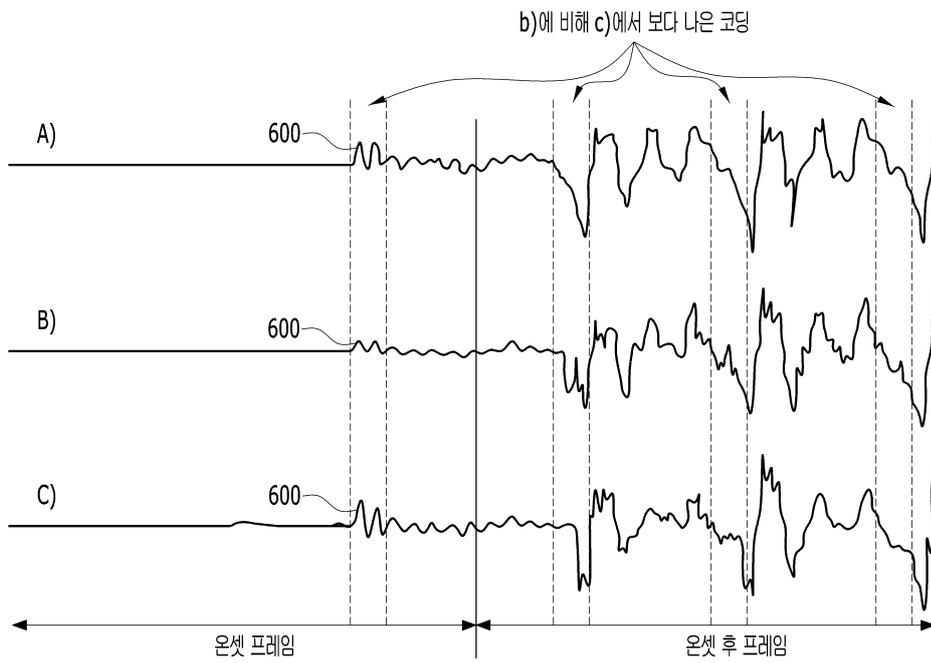
도면4



도면5



도면6



도면7

