



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2020년02월11일
(11) 등록번호 10-2068383
(24) 등록일자 2020년01월14일

(51) 국제특허분류(Int. Cl.)
H03M 7/30 (2006.01)
(52) CPC특허분류
H03M 7/3068 (2013.01)
(21) 출원번호 10-2015-7024996
(22) 출원일자(국제) 2014년03월01일
심사청구일자 2019년02월28일
(85) 번역문제출일자 2015년09월11일
(65) 공개번호 10-2015-0126858
(43) 공개일자 2015년11월13일
(86) 국제출원번호 PCT/EP2014/000529
(87) 국제공개번호 WO 2014/131526
국제공개일자 2014년09월04일
(30) 우선권주장
1303658.7 2013년03월01일 영국(GB)
(56) 선행기술조사문헌
KR1020110094046 A
KR1020130006689 A

(73) 특허권자
구루로직 마이크로시스템스 오이
핀란드 투르쿠 20100 린난카투 34
(72) 발명자
칼레보 오씨
핀란드 에프아이-37800 토이자라 케툰헨타 1
(74) 대리인
김태홍, 김진희

전체 청구항 수 : 총 45 항

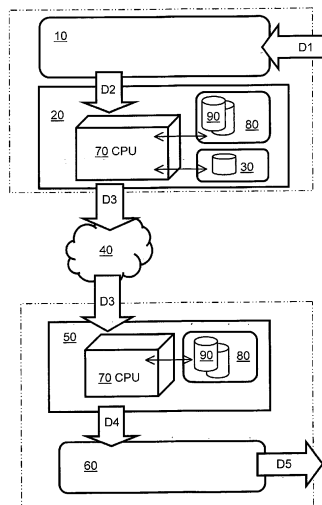
심사관 : 조춘근

(54) 발명의 명칭 엔트로피 변형기 및 방법

(57) 요약

제2 엔트로피를 갖는 대응 엔트로피 변형 출력 데이터를 생성하기 위한 제1 엔트로피를 갖는 입력 데이터 비트 스트림을 인코딩 또는 디코딩하도록 동작 가능한 엔트로피 변형기(10, 60)가 제공되는데, 여기서 엔트로피 변형기는 상호 유사 비트들의 그룹들 및 제1 비트의 값을 처리하기 위해 입력 데이터 비트 스트림을 프로세싱하고, 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이를 프로세싱하도록 동작 가능하다. 선택적으로, 엔트로피 변형기(10, 60)는 적어도 하나의 이스케이프 코드(escape code)를 이용함으로써 상호 유사 비트들의 그룹의 하나 이상의 최대 런 길이를 제어하도록 동작 가능하다.

대표도 - 도1



명세서

청구범위

청구항 1

비트 단위 방식(bit-by-bit manner)으로 제1 엔트로피를 갖는 입력 데이터 비트(D1)의 스트림을 인코딩하여, 제2 엔트로피를 갖는 대응하는 엔트로피 변형된 출력 데이터(D2)를 생성하도록 동작 가능한 엔트로피 변형기(10)에 있어서,

(a) 상기 엔트로피 변형기(10)는, 상기 출력 데이터(D2)에 포함된 후속 데이터를 위한 참조 값으로서 기능하는 제1 비트 값의 정보를 포함하기 위해 상기 출력 데이터(D2)를 비트 단위 방식으로 생성하도록 동작 가능하며;

(b) 상기 출력 데이터(D2)는 상기 입력 데이터 비트(D1)에 존재하는 상호 유사 비트들의 런 길이를 나타내는 데이터를 더 포함하고, 상기 상호 유사 비트들의 런 길이는 최대 런 길이 값(MaxRun)을 초과하지 않도록 사이즈가 한정되며;

(c) 상기 최대 런 길이 값(MaxRun)을 초과하는 상기 입력 데이터(D1)에서의 상기 런 길이의 하나 이상의 출현이, 상기 출력 데이터(D2)에 포함된 적어도 하나의 이스케이프 코드에 의해 표시되는 것을 특징으로 하는 엔트로피 변형기(10).

청구항 2

제1항에 있어서, 상기 입력 데이터 비트(D1)는 상호 유사 비트들의 그룹들을 포함하고, 상기 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이 값(MaxRun)은 복수의 이스케이프 코드에 의해 제어되는 것을 특징으로 하는 엔트로피 변형기(10).

청구항 3

제1항 또는 제2항에 있어서, 상기 엔트로피 변형기(10)는, 상기 출력 데이터(D2) 내에서의 상기 적어도 하나의 이스케이프 코드를 "0" 데이터 값으로 구현하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(10).

청구항 4

제1항에 있어서, 상기 엔트로피 변형기(10)는, 상기 입력 데이터 비트(D1)의 스트림의 특성(nature)의 함수로서 동적으로 변하는 상기 최대 런 길이 값(MaxRun)을 이용하여 상기 입력 데이터 비트(D1)의 스트림을 처리하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(10).

청구항 5

제1항에 있어서, 상기 엔트로피 변형기(10)는, 압축 데이터(D3)를 생성하기 위한 하나 이상의 코딩 장치(20)에서의 후속 압축을 위해 상기 입력 데이터(D1)를 상기 출력 데이터(D2)로 포맷화하는 데이터 사전 프로세서로서 기능하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(10).

청구항 6

제1항에 있어서, 상기 엔트로피 변형기(10)는 별도로 프로세싱되는 복수의 부분에서 상기 입력 데이터 비트(D1)의 스트림을 처리하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(10).

청구항 7

제6항에 있어서, 상기 엔트로피 변형기(10)는 시간 병렬 방식으로, 즉 병렬 실행을 통해 상기 복수의 부분을 처리하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(10).

청구항 8

제6항에 있어서, 상기 엔트로피 변형기(10)는, 상기 복수의 부분 중 하나 이상에 대해 서로 상이한 최대 런 길이 값(MaxRun)을 이용함으로써 상기 복수의 부분을 처리하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(10).

기(10).

청구항 9

제1항에 있어서, 상기 엔트로피 변형기(10)는, 상기 입력 데이터(D1)를 압축하여 상기 엔트로피 변형된 출력 데이터(D2 또는 D3)를 생성하는 인코더(10, 20)로서 기능하도록 구현되는 것을 특징으로 하는 엔트로피 변형기(10).

청구항 10

제1항에 있어서, 상기 엔트로피 변형기(10)는, 상기 입력 데이터(D1)에 존재하는 비트 스트링을 상기 엔트로피 변형된 출력 데이터(D2 또는 D3)에 존재하는 심볼들로 변환하는 것을 보조하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(10).

청구항 11

제1항에 있어서, 상기 엔트로피 변형기(10)는 컴퓨팅 하드웨어를 포함하고, 상기 컴퓨팅 하드웨어는, 상기 입력 데이터 비트(D1)의 스트림을 프로세싱하여 상기 출력 데이터(D2)를 생성하기 위해 기계 판독 가능 데이터 저장 매체 상에 기록된 하나 이상의 컴퓨터 프로그램을 실행하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(10).

청구항 12

제1 엔트로피를 갖는 입력 데이터 비트(D3 또는 D4)의 스트림을 디코딩하여, 제2 엔트로피를 갖는 대응하는 엔트로피 변형된 출력 데이터(D5)를 생성하도록 동작 가능한 엔트로피 변형기(60)에 있어서,

(a) 상기 엔트로피 변형기(60)는, 상기 입력 데이터(D3 또는 D4)에 포함된 후속 데이터를 위한 참조 값으로서 기능하는 제1 비트 값의 정보를 이용함으로써 상기 입력 데이터(D3 또는 D4)를 비트 단위 방식으로 프로세싱하도록 동작 가능하며;

(b) 상기 엔트로피 변형기(60)는, 오리지널 데이터(D1)에 존재하는 상호 유사 비트들의 런 길이를 나타내는 상기 입력 데이터(D3 또는 D4)에 존재하는 데이터를 프로세싱하도록 동작 가능하며, 상기 상호 유사 비트들의 런 길이는 최대 런 길이 값(MaxRun)을 초과하지 않도록 사이즈가 한정되고, 상기 입력 데이터는 상기 오리지널 데이터를 인코딩함으로써 생성된 데이터이며;

(c) 상기 엔트로피 변형기(60)는, 상기 입력 데이터(D3 또는 D4)에 포함된 적어도 하나의 이스케이프 코드에 의해 표시되는 상기 최대 런 길이 값을 초과하는 상기 오리지널 데이터(D1)에서의 상기 런 길이의 하나 이상의 출현을 검출하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 13

제12항에 있어서, 상기 오리지널 데이터(D1)는 상호 유사 비트들의 그룹들을 포함하고, 상기 엔트로피 변형기(60)는 적어도 하나의 이스케이프 코드를 이용함으로써 상기 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이 값(MaxRun)을 프로세싱하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 14

제12항에 있어서, 상기 엔트로피 변형기(60)는, 상기 입력 데이터 비트(D3 또는 D4)의 스트림의 특성의 함수로서 동적으로 변하는 상기 최대 런 길이 값(MaxRun)을 이용하여, 상기 입력 데이터 비트(D3 또는 D4)의 스트림을 처리하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 15

제12항에 있어서, 상기 엔트로피 변형기(60)는 상기 후속 데이터와는 별도로 상기 입력 데이터 비트(D3 또는 D4)의 스트림에서 상기 제1 비트 값의 정보를 처리하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 16

제12항에 있어서, 상기 엔트로피 변형기(60)는, 상기 입력 데이터(D3 또는 D4)를 디코딩하고, 디코딩된 비트 스

트림으로서 상기 출력 데이터(D5)를 생성하기 위해 인버스 엔트로피 변형을 적용하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 17

제12항에 있어서, 상기 엔트로피 변형기(60)는, 유사한 비트들의 연속량을 나타내는 상기 후속 데이터의 값의 시퀀스에 대해 별도로 인코딩되는 상기 제1 비트 값의 정보를 처리하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 18

제12항에 있어서, 상기 엔트로피 변형기(60)는 별도로 디코딩되는 복수의 부분에서 상기 입력 데이터 비트(D3 또는 D4)의 스트림을 처리하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 19

제18항에 있어서, 상기 엔트로피 변형기(60)는 시간 병렬 방식으로, 즉 병렬 실행을 통해 상기 복수의 부분을 처리하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 20

제18항에 있어서, 상기 엔트로피 변형기(60)는, 상기 복수의 부분 중 하나 이상에 대해 서로 상이한 최대 런 길이 값(MaxRun)을 사용함으로써 상기 복수의 부분을 처리하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 21

제12항에 있어서, 상기 엔트로피 변형기(60)는, 상기 입력 데이터(D3 또는 D4)를 압축해제하여 엔트로피 변형된 출력 데이터(D5)를 생성하기 위한 디코더(50, 60)로서 기능하도록 구현되는 것을 특징으로 하는 엔트로피 변형기(60).

청구항 22

제12항에 있어서, 상기 엔트로피 변형기(60)는 상기 입력 데이터(D3 또는 D4)에 존재하는 심볼들을 상기 출력 데이터(D5)에 존재하는 비트 스트림으로 변환하는 것을 보조하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 23

제12항에 있어서, 상기 엔트로피 변형기(60)는 컴퓨팅 하드웨어를 포함하고, 상기 컴퓨팅 하드웨어는 상기 입력 데이터 비트(D3 또는 D4)의 스트림을 프로세싱하여 상기 출력 데이터(D5)를 생성하기 위해 기계 판독 가능 데이터 저장 매체 상에 기록된 하나 이상의 컴퓨터 프로그램을 실행하도록 동작 가능한 것을 특징으로 하는 엔트로피 변형기(60).

청구항 24

엔트로피 변형기(10)를 사용해서, 비트 단위 방식으로 제1 엔트로피를 갖는 입력 데이터 비트(D1)의 스트림을 인코딩하여, 제2 엔트로피를 갖는 대응하는 엔트로피 변형된 출력 데이터(D2)를 생성하는 방법에 있어서,

- (a) 상기 엔트로피 변형기(10)를 동작시켜서, 상기 출력 데이터(D2)에 포함된 후속 데이터를 위한 참조 값으로서 기능하는 제1 비트 값의 정보를 포함하도록 상기 출력 데이터(D2)를 비트 단위 방식으로 생성하는 단계;
- (b) 상기 출력 데이터(D2)에, 상기 입력 데이터 비트(D1)에 존재하는 상호 유사 비트들의 런 길이를 나타내는 추가적인 데이터를 포함시키는 단계—상기 상호 유사 비트들의 런 길이는 최대 런 길이 값(MaxRun)을 초과하지 않도록 사이즈가 한정됨—; 및
- (c) 상기 최대 런 길이 값을 초과하는 상기 입력 데이터(D1)에서의 상기 런 길이의 하나 이상의 출현을, 상기 출력 데이터(D2)에 포함된 적어도 하나의 이스케이프 코드에 의해 표시하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 25

제24항에 있어서, 상기 입력 데이터 비트(D1)는 상호 유사 비트들의 그룹들을 포함하고, 복수의 이스케이프 코드를 사용하여 상기 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이 값(MaxRun)을 제어하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 26

제24항에 있어서, 상기 엔트로피 변형기(10)를 사용해서, 상기 입력 데이터 비트(D1)의 스트림의 특성의 함수로서 동적으로 변하는 상기 최대 런 길이 값(MaxRun)을 사용하여 상기 입력 데이터 비트(D1)의 스트림을 처리하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 27

제24항에 있어서, 상기 엔트로피 변형기(10)를 사용해서, 압축 데이터(D3)를 생성하기 위한 하나 이상의 코딩 장치(20)에서의 후속 압축을 위해 상기 입력 데이터(D1)를 상기 출력 데이터(D2)로 포맷화하는 데이터 사전 프로세서로서 기능하게 하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 28

제24항에 있어서, 상기 엔트로피 변형기(10)를 사용하여, 별도로 프로세싱되는 복수의 부분에서 상기 입력 데이터 비트(D1)의 스트림을 처리하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 29

제28항에 있어서, 상기 엔트로피 변형기(10)를 사용하여 시간 병렬 방식으로, 즉 병렬 실행을 통해 상기 복수의 부분을 처리하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 30

제28항에 있어서, 상기 엔트로피 변형기(10)를 사용해서, 상기 복수의 부분 중 하나 이상에 대해 서로 상이한 최대 런 길이 값(MaxRun)을 사용함으로써 상기 복수의 부분을 처리하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 31

제24항에 있어서, 상기 입력 데이터(D1)를 압축하여 상기 엔트로피 변형된 출력 데이터(D2 또는 D3)를 생성하는 인코더(10, 20)로서 기능하도록 상기 엔트로피 변형기(10)를 구현하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 32

제24항에 있어서, 상기 엔트로피 변형기(10)를 사용하여, 상기 입력 데이터(D1)에 존재하는 비트 스트링을 상기 엔트로피 변형된 출력 데이터(D2 또는 D3)에 존재하는 심볼들로 변환하는 것을 보조하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 33

엔트로피 변형기(60)를 사용해서, 제1 엔트로피를 갖는 입력 데이터 비트(D3 또는 D4)의 스트림을 디코딩하여, 제2 엔트로피를 갖는 대응하는 엔트로피 변형된 출력 데이터(D5)를 생성하는 방법에 있어서,

(a) 상기 엔트로피 변형기(60)를 사용해서, 상기 입력 데이터(D3 또는 D4)에 포함된 후속 데이터를 위한 참조 값으로서 기능하는 제1 비트 값의 정보를 사용하여 비트 단위 방식으로 상기 입력 데이터(D3 또는 D4)를 프로세싱하는 단계;

(b) 상기 엔트로피 변형기(60)를 사용해서, 오리지널 데이터(D1)에 존재하는 상호 유사 비트들의 런 길이를 나타내는, 상기 입력 데이터(D3 또는 D4)에 존재하는 데이터를 프로세싱하는 단계—상기 상호 유사 비트들의 런 길이는 최대 런 길이 값(MaxRun)을 초과하지 않도록 사이즈가 한정됨—; 및

(c) 상기 엔트로피 변형기(60)를 사용해서, 상기 입력 데이터(D3 또는 D4)에 포함된 적어도 하나의 이스케이프

코드에 의해 표시되는 상기 최대 런 길이 값을 초과하는 상기 오리지널 데이터(D1)에서의 상기 런 길이의 하나 이상의 출현을 검출하는 단계-상기 입력 데이터는 상기 오리지널 데이터를 인코딩함으로써 생성된 데이터임-를 포함하는 것을 특징으로 하는 방법.

청구항 34

제33항에 있어서, 상기 오리지널 데이터(D1)는 상호 유사 비트들의 그룹들을 포함하고, 상기 엔트로피 변형기(60)를 사용해서, 적어도 하나의 이스케이프 코드를 사용하여 상기 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이 값(MaxRun)을 프로세싱하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 35

제33항 또는 제34항에 있어서, 상기 적어도 하나의 이스케이프 코드를, 상기 입력 데이터(D3 또는 D4) 내에서 "0" 데이터 값으로서 구현하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 36

제33항에 있어서, 상기 엔트로피 변형기(60)를 사용해서, 상기 입력 데이터 비트(D3 또는 D4)의 스트림의 특성의 함수로서 동적으로 변하는 상기 최대 런 길이 값(MaxRun)을 사용하여 상기 입력 데이터(D3 또는 D4)의 스트림을 처리하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 37

제33항에 있어서, 상기 엔트로피 변형기(60)를 사용해서, 상기 후속 데이터와는 별도로 상기 입력 데이터 비트(D3 또는 D4)의 스트림에서 상기 제1 비트 값의 정보를 처리하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 38

제33항에 있어서, 상기 엔트로피 변형기(60)를 사용해서, 상기 입력 데이터(D3 또는 D4)를 디코딩하고, 디코딩된 비트 스트림으로서 상기 출력 데이터(D5)를 생성하기 위해 인버스 엔트로피 변형을 적용하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 39

제37항에 있어서, 상기 엔트로피 변형기(60)를 사용해서, 유사한 비트의 연속량을 나타내는 상기 후속 데이터의 값의 시퀀스에 관해 별도로 인코딩되는 상기 제1 비트 값의 정보를 처리하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 40

제33항에 있어서, 상기 엔트로피 변형기(60)를 사용해서, 별도로 디코딩되는 복수의 부분에서 상기 입력 데이터 비트(D3 또는 D4)의 스트림을 처리하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 41

제40항에 있어서, 상기 엔트로피 변형기(60)를 사용해서 시간 병렬 방식으로, 즉 병렬 실행을 통해 상기 복수의 부분을 처리하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 42

제40항에 있어서, 상기 엔트로피 변형기(60)를 사용해서 상기 복수의 부분 중 하나 이상에 대해 서로 상이한 최대 런 길이 값(MaxRun)을 사용함으로써 상기 복수의 부분을 처리하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 43

제33항에 있어서, 상기 엔트로피 변형기(60)를 사용해서, 상기 입력 데이터(D3 또는 D4)를 압축 해제하여 상기 엔트로피 변형된 출력 데이터(D5)를 생성하는 디코더(50, 60)로서 기능하게 하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 44

제33항에 있어서, 상기 엔트로피 변형기(60)를 사용해서, 상기 입력 데이터(D3 또는 D4)에 존재하는 심볼들을 상기 출력 데이터(D5)에 존재하는 비트 스트링으로 변환하는 것을 보조하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 45

장치로 하여금, 상기 장치의 프로세싱 수단 상에서 실행될 때에, 제24항에 기재된 방법을 수행하게 하도록 구성되는 프로그램 명령어를 포함하는 컴퓨터 프로그램을 저장하는 기계 판독 가능한 저장 매체.

청구항 46

삭제

청구항 47

삭제

청구항 48

삭제

청구항 49

삭제

청구항 50

삭제

청구항 51

삭제

청구항 52

삭제

청구항 53

삭제

청구항 54

삭제

청구항 55

삭제

청구항 56

삭제

청구항 57

삭제

발명의 설명

기술 분야

[0001] 본 발명은 예를 들어, 데이터 압축 및/또는 데이터 압축 해제 목적으로, 입력 데이터를 수신하고, 입력 데이터에 대해 엔트로피가 변형된 대응하는 출력 데이터를 생성하도록 동작 가능한 엔트로피 변형기에 관한 것으로서, 엔트로피 변형기들은, 기계 판독 가능 데이터 저장 매체에 저장되고, 컴퓨팅 하드웨어에서 실행 가능한 전자 하드웨어 및/또는 소프트웨어 제품들을 이용하여 구현 가능하다. 더구나, 본 발명은 또한 입력 데이터에 대해 변형된 정도의 엔트로피를 갖는 대응 출력 데이터를 생성하기 위해 입력 데이터의 엔트로피를 변형하는 방법에 관한 것이다. 게다가, 본 발명은 기계 판독 가능 데이터 저장 매체에 기록된 소프트웨어 제품들에 관한 것으로서, 여기서 소프트웨어 제품들은 후술하는 방법들을 실행하기 위한 컴퓨팅 하드웨어에서 실행 가능하다. 예를 들어, 엔트로피 변형기들은 데이터 통신 시스템들 및 데이터 공급 시스템들, 예를 들어 미디어 공급 시스템들의 컴포넌트 부분들로서 사용 가능하다.

배경 기술

[0002] 개괄적으로 설명하면, 정보 이론에서, 엔트로피(entropy)는 랜덤 변수에서의 불확실성의 측정치로서, 정보 시스템, 예를 들어 데이터 통신 시스템뿐 아니라 열역학에서와 같이 물리적 연속 시스템에 관한 것이다. 더구나, 데이터 통신 시스템의 상황에서, 엔트로피는 더 적절하게는 샤논 엔트로피(Shannon entropy)라고 지칭되며, 참조로서 여기에 통합된 Claude E. Shannon의 1948년 과학 논문 "A Mathematical Theory of Communication"를 참조하고, 샤논 엔트로피는 소정의 메시지에 포함된 예측 정보 값을 정량화하기 위해 사용될 수 있다. 더구나, 샤논 엔트로피는 비트들뿐 아니라 내트(nat) 및 반(ban)의 측면에서 표현 가능하다.

[0003] 샤논 엔트로피는 임의의 통신의 최선의 가능한 무손실 인코딩 또는 압축에 관한 절대적 한계를 제공하는데, 이 때 소정의 통신이, 독립적이고 동일하게 분포된 랜덤 변수들의 시퀀스로서 용이하게 나타낼 수 있다고 가정한다. 더구나, 샤논의 이론은, 소정의 알파벳으로 된 소정의 메시지를 인코딩하기 위해 가장 짧은 가능한 표현의 평균 길이(L)가 그 알파벳에 존재하는 심볼 N의 개수의 로그에 의해 나누어진 엔트로피(E)라는 것을 확인하였는데, 즉 수학적 1에 정의된 바와 같다.

수학식 1

$$L = \frac{E}{\log_{10} N}$$

[0004]

[0005] 이로 인해, 엔트로피(E)는 예측 불가능성 또는 정보 내용의 척도이다. 무손실 데이터 압축 방법에서, 대응하는 입력 데이터에 이 방법을 적용함으로써 생성되는 압축 출력 데이터는 입력 데이터와 유사한 양의 정보를 갖지만, 출력 데이터는 입력 데이터에 비해 더 적은 데이터 비트를 포함한다. 그 결과, 압축 출력 데이터는 더 적은 정보 리던던시(redundancy of information)를 포함하기 때문에 더 예측 불가능하다.

[0006] 샤논의 이론(Shannon's theory)은 공지된 데이터 인코딩 장치의 설계 시에 고려되어 왔다. 예를 들어, 공개된 국제 특허 출원 제W02010/050157A1호(PCT/JP2009/005548, "Image encoding apparatus, image encoding method, and image encoding program", 출원인 - Thomson Licensing)에는, 이미지 인코딩 장치, 이미지 인코딩 방법, 및 이미지 인코딩 프로그램이 기술되어 있는데, 이들은 인코딩 효율을 낮추지 않고 전체적으로 이미지의 이미지 품질을 균일하게 하도록 동작 가능하며, 고속으로 동작 가능하고, 슬라이스 구조를 변경하지 않고서 매크로블록 셔플링(macroblock shuffling)을 수행함으로써 회로 스케일의 사이즈를 축소시킬 수 있다. 더구나,

[0007] (i) 이미지 내의 각각의 위치들로부터 이미지 데이터를 구성하는 복수의 매크로블록을 수집 및 셔플링하는 셔플링 부분;

[0008] (ii) 셔플링 부분에 의해 수집 및 셔플링된 복수의 매크로블록에 대한 공간 주파수 변환 및 엔트로피 인코딩을 수행하는 인코딩 부분; 및

[0009] (iii) 인코딩이 실행된 이후 복수의 매크로블록의 레이트를 조정하도록 인코딩 부분을 제어하는 레이트 제어 부분을 포함하는 이미지 인코딩 장치 또한 제공된다.

[0010] 데이터의 타입과 무관하게, 데이터는 하나의 공간 위치로부터 다른 위치로 이동하는 경우 데이터 저장 공간 및 통신 네트워크 용량의 대역폭을 필요로 한다. 이러한 대역폭은 실제로 에너지의 활용 및 통신 인프라구조의 투자에 대응한다. 데이터 통신 양이 장래에 증가할 것으로 예측됨에 따라, 더 많은 데이터 저장 공간 및 더 많은

통신 시스템 용량이 필요할 것이고, 종종 더 많은 에너지가 필요할 것이다. 동시대의 인터넷에서, 종종 다수의 사본에 막대한 양의 데이터를 저장하고 있다. 이로 인해, 데이터를 압축할 수 있는 임의의 접근법은, 특히 이러한 압축이 무손실인 경우 잠재적 그리고 기술적 경제적 측면에서 크게 유리하다. 현재, 데이터 세트를 압축하기 위해 데이터 세트 내의 엔트로피를 감소시키기 위한 여러 공지의 방법들이 존재한다. 더구나, 데이터 세트, 예를 들어 델타 코딩(Delta coding) 및 런 길이 인코딩(Run-Length-Encoding, RLE)에 존재하는 엔트로피를 변형하는 공지의 방법들이 존재하지만, 더 효과적인 데이터 압축을 제공하는 새로운 방법들이 여전히 요구된다.

[0011] 이에 따라, 소정의 데이터에 존재하는 엔트로피를 감소시킴으로써 소정의 데이터를 압축하기 위해 채택 가능한 여러 상이한 압축 방법들, 예를 들어 전술한 RLE 코딩, 전술한 VLC 코딩, 허프만 코딩(Huffman coding), 델타 코딩(Delta coding), 산술 코딩(Arithmetic coding) 및 레인지 코딩(Range coding)이 존재한다. 이들 방법은 통상적으로 알파벳, 숫자, 바이트, 및 워드를 압축하도록 설계된다. 그러나 이러한 방법들은 개별 비트를 압축하기에 특히 적합하지 않고, 이러한 이유로 이상적으로, 비트별 방식으로 변할 수 있는 데이터, 예를 들어, 데이터 비트 스트림들을 압축하는데 적합하지 않다.

[0012] 종래의 RLE 코딩의 경우, 소정의 값(비트), 또는 그 값의 2배(비트), 그 다음 다수의 유사 값(비트들)이 저장된다. RLE는 선택적으로 적용될 수 있는데, 예를 들어, 즉 그 코딩은 다수의 유사한 값들에 대해 공지의 비트량을 포함하는 런(run)을 코딩하기 위해서만 예약될 수 있다. RLE의 이러한 선택적 적용은, 각각의 새로운 런을 가진 소정의 데이터 스트림에 동일한 값이 한번씩 또는 두번씩 다시 포함되는 것을 필요로 한다. 그러나 데이터의 타입과 무관하게, 데이터 세트에서의 엔트리를 감소시키는, 데이터 압축에 대한 더 나은 접근법들이 현재 요구된다.

발명의 내용

해결하려는 과제

[0013] 본 발명의 제1 목적은 예를 들어, 개선된 데이터 압축을 제공하기 위해 개선된 엔트로피 변형기를 제공하는 것이다.

[0014] 본 발명의 제2 목적은 예를 들어, 개선된 데이터 압축을 제공하기 위해 개선된 엔트로피 변형 방법을 제공하는 것이다.

과제의 해결 수단

[0015] 본 발명의 제1 양태에 따르면, 제1항에 청구된 바와 같은 엔트로피 변형기가 제공되는데, 제2 엔트로피를 갖는 대응하는 엔트로피 변형 출력 데이터(D2)를 생성하기 위해 제1 엔트로피를 갖는 입력 데이터 비트(D1)의 스트림을 인코딩하도록 동작 가능한 엔트로피 변형기가 제공되는데,

[0016] (a) 엔트로피 변형기(10)는 출력 데이터(D2)에 포함된 후속 데이터를 위한 참조 값으로서 기능하는 제1 비트의 정보를 포함하기 위해 출력 데이터(D2)를 생성하도록 동작 가능하며;

[0017] (b) 출력 데이터(D2)는 입력 데이터 비트(D1)에 존재하는 상호 유사 비트들의 런 길이들을 나타내는 데이터를 더 포함하고, 상호 유사 비트들의 런 길이들은 최대 런 길이 값(MaxRun)을 초과하지 않도록 사이즈가 실질적으로 한정되며;

[0018] (c) 최대 런 길이 값을 실질적으로 초과하는 입력 데이터(D1)에서의 런 길이들의 하나 이상의 생성은 출력 데이터(D2)에 포함된 적어도 하나의 이스케이프 코드에 의해 표시된다.

[0019] 본 발명은 데이터의 런 길이 표현의 선택적 사용으로, 향상된 엔트로피 변형, 예를 들어, 향상된 데이터 압축을 제공할 수 있다는 점에서 이점을 갖는다.

[0020] 선택적으로, 엔트로피 변형기는 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이가 복수의 이스케이프 코드에 의해 제어되도록 구현된다.

[0021] 채택된 일 종류의 이스케이프 코드 솔루션은 최대 런 길이 값, 및 소정 양의 유사 비트들이 연속된다는 표시를 이용한다. 이러한 종류의 솔루션에서, 모든 심볼들은 하나의 데이터 스트림으로서 코딩될 수 있다. 다른 종류의 이스케이프 코드는 코딩된 심볼들과 별도의 여러 상이한 솔루션들의 활용을 가능하게 하고, 최대 런 길이 값보다 큰 유사 비트 값의 연속량을 나타낸다. 이러한 종류의 솔루션은 2개의 별도의 스트림으로서 처리되거나, 그 수를 나타내는 임의의 값이 최대 런보다 더 크지 않도록 설계될 수 있으며, 이로써 이들 솔루션은 또한 다른 심

볼들의 동일한 스트림을 이용하여 코딩될 수 있다. 또한, 다른 이스케이프 방법들이 선택적으로 채택된다.

- [0022] 선택적으로, 엔트로피 변형기는 입력 데이터(D1)의 스트림의 특성의 함수에 따라 동적으로 달라지는 방식으로 최대 런 길이를 이용하여 입력 데이터(D1)의 스트림을 처리하도록 동작 가능하다.
- [0023] 선택적으로, 엔트로피 변형기는, 압축 데이터(D3)를 생성하기 위한 하나 이상의 코딩 배열들에서의 후속 압축을 위해 입력 데이터(D1)를 출력 데이터(D2)로 포맷화하기 위한 데이터 사전 프로세서로서 기능하도록 동작 가능하다.
- [0024] 선택적으로, 적어도 하나의 이스케이프 코드는 엔트로피 변형기에 의해 생성된 출력 데이터(D2) 내에 "0" 값으로서 구현되도록 엔트로피 변형기가 구현된다.
- [0025] 선택적으로, 하나 이상의 코딩 배열들은 오프셋(OFFSET) 코더, RLE 코딩, VLC 코딩, 허프만 코딩, 델타 코딩, 산술 코딩, 레인지 코딩 중 하나 이상을 포함한다.
- [0026] 선택적으로, 엔트로피 변형기는 별도로 프로세싱되는 복수의 부분에서 입력 데이터 비트(D1)의 스트림을 처리하도록 동작 가능하다. 추가 선택적으로, 엔트로피 변형기는 시간 병렬 방식으로, 즉 병렬 실행을 통해 복수의 부분들을 처리하도록 동작 가능하다. 추가 선택적으로, 엔트로피 변형기는 복수의 부분 중 하나 이상에 대해 상호 상이한 최대 런 길이들을 이용함으로써 복수의 부분들을 처리하도록 동작 가능하다.
- [0027] 제1 비트의 정보는 자기 자신의 데이터 스트림으로 인코딩되거나, 런 길이 정보를 이용하여 동일한 데이터 스트림으로 인코딩될 수 있으며, 사용된 코딩 방법의 정보와 함께 제1 비트의 정보가 포함될 수 있음은 물론이다.
- [0028] 선택적으로, 엔트로피 변형기는 입력 데이터(D1)의 스트림을 압축하여 엔트로피 변형 출력(D2, D3)을 생성하기 위한 인코더로서 기능하도록 구현된다.
- [0029] 선택적으로, 엔트로피 변형기는 입력 데이터(D1)에 존재하는 비트 스트림을 엔트로피 변형 출력 데이터(D2)에 존재하는 심볼들로 변환하는 것을 보조하도록 동작 가능하다.
- [0030] 선택적으로, 엔트로피 변형기는 컴퓨팅 하드웨어를 포함하고, 컴퓨팅 하드웨어는 입력 데이터(D1)의 스트림을 프로세싱하여 출력 데이터(D2)를 생성하기 위해 기계 판독 가능 데이터 저장 매체에 기록된 하나 이상의 소프트웨어 제품을 실행하도록 동작 가능하다.
- [0031] 본 발명의 제2 양태에 따르면, 제2 엔트로피를 갖는 대응하는 엔트로피 변형 출력 데이터(D5)를 생성하기 위해 제1 엔트로피를 갖는 입력 데이터 비트(D3 또는 D4)의 스트림을 디코딩하도록 동작 가능한 엔트로피 변형기가 제공되는데,
- [0032] (a) 엔트로피 변형기(60)는 입력 데이터(D3 또는 D4)에 포함된 후속 데이터를 위한 참조 값으로서 기능하는 제1 비트의 정보를 이용함으로써 입력 데이터(D3 또는 D4)를 프로세싱하도록 동작 가능하며;
- [0033] (b) 엔트로피 변형기(60)는 입력 데이터 비트(D3 또는 D4)에 존재하는 상호 유사 비트들의 런 길이들을 나타내는 입력 데이터(D3 또는 D4)에 존재하는 데이터를 프로세싱하도록 동작 가능하며, 상호 유사 비트의 최대 길이는 최대 런 길이 값(MaxRun)을 초과하지 않도록 사이즈가 실질적으로 한정되며;
- [0034] (c) 엔트로피 변형기(60)는 입력 데이터(D3 또는 D4)에 포함된 적어도 하나의 이스케이프 코드에 의해 표시되는 것인, 최대 런 길이 값을 실질적으로 초과하는 입력 데이터(D3 또는 D4)에서의 런 길이들의 하나 이상의 생성을 검출하도록 동작 가능하다.
- [0035] 선택적으로, 엔트로피 변형기는 적어도 하나의 이스케이프 코드를 이용함으로써 상호 유사 비트들의 그룹의 하나 이상의 최대 런 길이를 프로세싱하도록 동작 가능하다. 추가 선택적으로, 엔트로피 변형기는 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이가 복수의 이스케이프 코드를 이용함으로써 프로세싱되도록 구현된다.
- [0036] 채택된 일 종류의 이스케이프 코드 솔루션은 최대 런 길이 값, 및 소정 양의 유사 비트들이 연속된다는 표시를 이용한다. 이러한 종류의 솔루션에서, 모든 심볼들은 하나의 데이터 스트림으로서 코딩될 수 있으며, 마찬가지로 하나의 데이터 스트림으로 디코딩된다. 다른 종류의 이스케이프 코드는 코딩된 심볼들과 별도의 여러 상이한 솔루션들의 활용을 가능하게 하고, 최대 런 길이 값보다 큰 유사 비트 값의 연속량을 나타낸다. 이러한 종류의 솔루션은 2개의 별도의 스트림으로서 처리되거나, 그 수를 나타내는 임의의 값이 최대 런보다 더 크지 않도록 설계될 수 있으며, 이로써 이들 솔루션은 또한 다른 심볼들의 동일한 스트림을 이용하여 코딩될 수 있다. 또한, 다른 이스케이프 방법들이 선택적으로 채택된다.

- [0037] 선택적으로, 엔트로피 변형기는 입력 데이터(D3 또는 D4)의 스트림의 특성의 함수에 따라 동적으로 달라지는 방식으로 최대 런 길이를 이용하여 입력 데이터(D3 또는 D4)의 스트림을 처리하도록 동작 가능하다.
- [0038] 선택적으로, 엔트로피 변형기는 데이터 비트 시퀀스와 별도로 입력 데이터(D3 또는 D4)의 스트림에서 제1 비트 값의 정보를 처리하도록 동작 가능하다. 추가 선택적으로, 엔트로피 변형기는 연속적인 유사 비트들의 양을 나타내는 데이터 값들의 시퀀스에 관해 별도로 인코딩되는 것으로서 제1 비트의 값을 처리하도록 동작 가능하다. 추가 선택적으로, 적어도 하나의 이스케이프 코드가 엔트로피 변형기 내에 "0" 값으로서 구현되도록 엔트로피 변형기가 구현된다.
- [0039] 선택적으로, 엔트로피 변형기는 입력 데이터(D3 또는 D4)에 존재하는 데이터 요소들을 디코딩하고, 디코딩된 비트 스트림으로서 출력 데이터(D5)를 생성하도록 인버스 엔트로피 변형을 적용하도록 동작 가능하다.
- [0040] 선택적으로, 엔트로피 변형기는 입력 데이터 비트(D3)의 스트림을 처리하기 위한 하나 이상의 코딩 배열과 함께 사후 프로세싱하는데 활용되며, 하나 이상의 코딩 배열들은 오델타 코딩, RLE 코딩, VLC 코딩, 허프만 코딩, 델타 코딩, 산술 코딩, 레인지 코딩 중 하나 이상을 포함한다.
- [0041] 선택적으로, 엔트로피 변형기는 별도로 인코딩 또는 디코딩되는 복수의 부분에서 입력 데이터 비트(D3 또는 D4)의 스트림을 처리하도록 동작 가능하다. 추가 선택적으로, 엔트로피 변형기는 시간 병렬 방식으로, 즉 병렬 실행을 통해 복수의 부분들을 처리하도록 동작 가능하다. 추가 선택적으로, 엔트로피 변형기는 복수의 부분 중 하나 이상에 대해 상호 상이한 최대 런 길이들을 이용함으로써 복수의 부분들을 처리하도록 동작 가능하다.
- [0042] 선택적으로, 엔트로피 변형기는 엔트로피 변형된 출력(D5)을 생성하기 위해 디코딩한 이후 입력 데이터(D3 또는 D4)의 스트림을 엔트로피 변형하기 위한 사후 프로세서로서 기능하도록 구현된다.
- [0043] 선택적으로, 엔트로피 변형기는 입력 데이터(D3 또는 D4)에 존재하는 심볼들을 출력 데이터(D5)에 존재하는 비트 스트림으로 변환하는 것을 보조하도록 동작 가능하다.
- [0044] 선택적으로, 엔트로피 변형기는 컴퓨팅 하드웨어를 포함하고, 컴퓨팅 하드웨어는 입력 데이터(D3 또는 D4)의 스트림을 프로세싱하여 출력 데이터(D5)를 생성하기 위해 기계 관독 가능 데이터 저장 매체에 기록된 하나 이상의 소프트웨어 제품을 실행하도록 동작 가능하다.
- [0045] 본 발명의 제3 양태에 따르면, 엔트로피 변형기를 이용하여 제2 엔트로피를 갖는 대응하는 엔트로피 변형 출력 데이터(D2)를 생성하기 위해 제1 엔트로피를 갖는 입력 데이터 비트(D1)의 스트림을 인코딩하는 방법이 제공되는데, 이 방법은,
- [0046] (a) 출력 데이터(D2)에 포함된 후속 데이터를 위한 참조 값으로서 기능하는 제1 비트의 정보를 포함하기 위해 엔트로피 변형기(10)를 동작시켜서 출력 데이터(D2)를 생성하는 단계;
- [0047] (b) 출력 데이터(D2)에, 입력 데이터 비트(D1)에 존재하는 상호 유사 비트들의 런 길이들을 나타내는 데이터를 더 포함시키는 단계로서, 상호 유사 비트들의 런 길이들은 최대 런 길이 값(MaxRun)을 초과하지 않도록 사이즈가 실질적으로 한정되는 것인 상기 단계; 및
- [0048] (c) 최대 런 길이 값을 실질적으로 초과하는 입력 데이터(D1)에서의 런 길이들의 하나 이상의 생성을 출력 데이터(D2)에 포함된 적어도 하나의 이스케이프 코드에 의해 표시하는 단계를 포함한다.
- [0049] 선택적으로, 그 방법은 적어도 하나의 이스케이프 코드를 이용함으로써 엔트로피 변형기를 사용하여, 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이를 제어하는 단계를 포함한다. 추가 선택적으로, 그 방법은 복수의 이스케이프 코드를 이용하여 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이를 제어하는 단계를 포함한다. 추가 선택적으로, 그 방법은 입력 데이터(D1)의 스트림의 특성의 함수에 따라 동적으로 달라지는 방식으로 최대 런 길이를 이용함으로써 엔트로피 변형기를 사용하여 입력 데이터(D1)의 스트림을 처리하는 단계를 포함한다.
- [0050] 선택적으로, 그 방법은 엔트로피 변형기를 사용하여, 데이터 비트 시퀀스와 별도로 입력 데이터(D1)의 스트림에서 제1 비트의 값의 정보를 처리하는 단계를 포함한다. 추가 선택적으로, 그 방법은 엔트로피 변형기를 사용하여, 유사 비트들의 연속 양을 나타내는 데이터 값들의 시퀀스에 관해 별도로 인코딩되는 것으로서 제1 비트의 값을 처리하는 단계를 포함한다.
- [0051] 선택적으로, 그 방법은 적어도 하나의 이스케이프 코드가 엔트로피 변형기 내에 "0" 값으로서 구현되도록 구현

된다.

- [0052] 선택적으로, 그 방법은 입력 데이터 비트(D1)의 스트림을 처리하기 위한 하나 이상의 코딩 배열과 함께 사후 프로세싱하기 위해 엔트로피 변형기를 사용하는 단계를 포함하며, 하나 이상의 코딩 배열들은 오델타 코딩, RLE 코딩, VLC 코딩, 허프만 코딩, 델타 코딩, 산술 코딩, 레인지 코딩 중 하나 이상을 포함한다.
- [0053] 선택적으로, 그 방법은 엔트로피 변형기를 사용하여, 별도로 인코딩 또는 디코딩되는 복수의 부분에서 입력 데이터 비트(D1)의 스트림을 처리하는 단계를 포함한다.
- [0054] 선택적으로, 그 방법은 엔트로피 변형기를 사용하여, 시간 병렬 방식으로, 즉 병렬 실행을 통해 복수의 부분들을 처리하는 단계를 포함한다.
- [0055] 선택적으로, 그 방법은 복수의 부분 중 하나 이상에 대해 상호 상이한 최대 런 길이들을 이용함으로써 엔트로피 변형기를 사용하여 복수의 부분들을 처리하는 단계를 포함한다.
- [0056] 선택적으로, 그 방법은 입력 데이터(D1)의 스트림을 압축하여 엔트로피 변형 출력(D2, D3)을 생성하기 위한 인코더로서 기능하도록 엔트로피 변형기를 구현하는 단계를 포함한다. 추가 선택적으로, 그 방법은 엔트로피 변형기를 사용하여 입력 데이터(D1)에 존재하는 비트 스트림을 엔트로피 변형 출력 데이터(D2)에 존재하는 심볼들로 변환하는 것을 보조하는 단계를 포함한다. 추가 선택적으로, 그 방법은 컴퓨팅 하드웨어를 포함하도록 엔트로피 변형기를 구현하는 단계를 포함하고, 여기서 컴퓨팅 하드웨어는 입력 데이터(D1)의 스트림을 프로세싱하여 출력 데이터(D2)를 생성하기 위해 기계 판독 가능 데이터 저장 매체에 기록된 하나 이상의 소프트웨어 제품을 실행하도록 동작 가능하다.
- [0057] 본 발명의 제4 양태에 따르면, 엔트로피 변형기를 사용하여, 제2 엔트로피를 갖는 대응하는 엔트로피 변형 출력 데이터(D5)를 생성하기 위해 제1 엔트로피를 갖는 입력 데이터 비트(D3 또는 D4)의 스트림을 디코딩하는 방법이 제공되는데, 이 방법은,
- [0058] (a) 입력 데이터(D3 또는 D3)에 포함된 후속 데이터를 위한 참조 값으로서 기능하는 제1 비트의 정보를 이용함으로써 엔트로피 변형기(60)를 사용하여 입력 데이터(D3 또는 D4)를 프로세싱하는 단계;
- [0059] (b) 엔트로피 변형기(60)를 사용하여, 입력 데이터 비트(D3 또는 D4)에 존재하는 상호 유사 비트들의 런 길이들을 나타내는 입력 데이터(D3 또는 D4)에 존재하는 데이터를 프로세싱하는 단계로서, 상호 유사 비트들의 런 길이들은 최대 런 길이 값(MaxRun)을 초과하지 않도록 사이즈가 실질적으로 한정되는 것인 상기 단계; 및
- [0060] (c) 엔트로피 변형기(60)를 사용하여, 입력 데이터(D3 또는 D4)에 포함된 적어도 하나의 이스케이프 코드에 의해 표시되는 것인, 최대 런 길이 값을 실질적으로 초과하는 입력 데이터(D1)에서의 런 길이들의 하나 이상의 생성을 검출하는 단계를 포함한다.
- [0061] 선택적으로, 그 방법은 적어도 하나의 이스케이프 코드를 이용함으로써 엔트로피 변형기를 사용하여, 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이를 프로세싱하는 단계를 포함한다. 추가 선택적으로, 그 방법은 복수의 이스케이프 코드를 이용함으로써 상호 유사 비트들의 그룹들의 하나 이상의 최대 런 길이를 프로세싱하는 단계를 포함한다.
- [0062] 선택적으로, 그 방법은 입력 데이터(D3 또는 D4)의 스트림의 특성의 함수에 따라 동적으로 달라지는 방식으로 최대 런 길이를 이용함으로써 엔트로피 변형기를 사용하여 입력 데이터(D3 또는 D4)의 스트림을 처리하는 단계를 포함한다.
- [0063] 선택적으로, 그 방법은 엔트로피 변형기를 사용하여, 데이터 비트 시퀀스와 별도로 입력 데이터(D3 또는 D4)의 스트림에서 제1 비트의 값의 정보를 처리하는 단계를 포함한다. 추가 선택적으로, 그 방법은 엔트로피 변형기를 사용하여, 연속적인 유사 비트들의 양을 나타내는 데이터 값들의 시퀀스에 관해 별도로 인코딩되는 바와 같이 제1 비트의 값을 처리하는 단계를 포함한다. 추가 선택적으로, 그 방법은 적어도 하나의 이스케이프 코드가 엔트로피 변형기 내에 "0" 값으로서 구현되도록 구현된다.
- [0064] 선택적으로, 그 방법은 엔트로피 변형기를 사용하여, 입력 데이터(D3 또는 D4)에 존재하는 데이터 요소들을 디코딩하고, 디코딩된 비트 스트림으로서 출력 데이터(D5)를 생성하도록 인버스 엔트로피 변형을 적용하는 단계를 포함한다.
- [0065] 선택적으로, 그 방법은 입력 데이터 비트(D3)의 스트림을 처리하기 위한 하나 이상의 코딩 배열과 함께 사후 프로세싱하기 위해 엔트로피 변형기를 사용하는 단계를 포함하며, 하나 이상의 코딩 배열들은 오델타 코딩, RLE

코딩, VLC 코딩, 허프만 코딩, 델타 코딩, 산술 코딩, 레인지 코딩 중 하나 이상을 포함한다.

- [0066] 선택적으로, 그 방법은 엔트로피 변형기를 사용하여, 별도로 프로세싱되는 복수의 부분에서 입력 데이터 비트(D3 또는 D4)의 스트림을 처리하는 단계를 포함한다. 추가 선택적으로, 그 방법은 엔트로피 변형기를 사용하여, 시간 병렬 방식으로, 즉 병렬 실행을 통해 복수의 부분들을 처리하는 단계를 포함한다.
- [0067] 디코더는 런 길이 정보를 이용하여 자신의 데이터 스트림 또는 동일한 데이터 스트림으로부터 제1 비트(값)의 정보를 얻거나, 이 정보는 사용된 코딩 배열(들)의 정보와 함께 포함될 수 있음은 물론이다. 선택적으로, 그 방법은 복수의 부분 중 하나 이상에 대해 상호 상이한 최대 런 길이들을 이용함으로써 엔트로피 변형기를 사용하여 복수의 부분들을 처리하는 단계를 포함한다.
- [0068] 선택적으로, 그 방법은 입력 데이터(D3 또는 D4)의 스트림을 압축 해제하여 엔트로피 변형 출력(D5)을 생성하기 위한 디코더로서 기능하도록 상기 엔트로피 변형기를 사용하는 단계를 포함한다.
- [0069] 선택적으로, 그 방법은 엔트로피 변형기를 사용하여, 입력 데이터(D3 또는 D4)에 존재하는 심볼들을 출력 데이터(D5)에 존재하는 비트 스트림으로 변환하는 것을 보조하는 단계를 포함한다.
- [0070] 선택적으로, 그 방법은 컴퓨팅 하드웨어를 포함하도록 엔트로피 변형기를 구현하는 단계를 포함하고, 컴퓨팅 하드웨어는 입력 데이터(D3 또는 D4)의 스트림을 프로세싱하여 출력 데이터(D5)를 생성하기 위해 기계 판독 가능 데이터 저장 매체에 기록된 하나 이상의 소프트웨어 제품을 실행하도록 동작 가능하다.
- [0071] 본 발명의 제5 양태에 따르면, 기계 판독 가능 데이터 저장 매체에 기록된 소프트웨어 제품이 제공되는데, 여기서 소프트웨어 제품은 본 발명의 제3 양태에 따른 방법을 실행하기 위한 컴퓨팅 하드웨어에서 실행 가능하다.
- [0072] 본 발명의 제6 양태에 따르면, 기계 판독 가능 데이터 저장 매체에 기록된 소프트웨어 제품이 제공되는데, 여기서 소프트웨어 제품은 본 발명의 제4 양태에 따른 방법을 실행하기 위한 컴퓨팅 하드웨어에서 실행 가능하다.
- [0073] 본 발명의 특징들이 첨부된 청구범위에 의해 정의된 바와 같이 본 발명의 범위로부터 벗어나지 않고 다양한 조합으로 용이하게 결합될 수 있음은 물론이다.

도면의 간단한 설명

- [0074] 본 발명의 실시예들은 단지 일례로서 다음의 첨부 도면을 참조하여 이하 설명될 것이다.
도 1은 본 발명에 따른 엔트로피 변형기(entropy modifier)들을 포함하는 코덱 시스템의 예시이다.
첨부 도면에서, 밑줄 있는 번호는 밑줄 있는 번호가 위치하는 아이템 또는 밑줄 있는 번호가 인접한 아이템을 나타내기 위해 채택된다. 밑줄 없는 번호는 밑줄 없는 번호를 아이템에 링크하는 라인에 의해 식별된 아이템에 관한 것이다. 번호가 밑줄 없이 관련 화살표에 의해 수반되는 경우, 밑줄 없는 번호는 화살표가 가리키는 일반적인 아이템을 식별하기 위해 사용된다.

발명을 실시하기 위한 구체적인 내용

- [0075] 개괄적으로 설명하면, 본 발명은 엔트로피 변형기에 관한 것이다. 도 1을 참조하면, 입력 데이터(D1)는 입력 데이터를 변환하여 대응하는 변환 데이터(D2)를 생성하는 제1 엔트로피 변형기(10)에 공급되고, 선택적으로 데이터(D2)는 제1 엔트로피 변형기(10)에 채택된 변환의 특성에 따라 데이터(D1)에 대해 서로 사이즈가 상이하고, 데이터(D1)에 비해 감소된 엔트로피를 갖는다. 선택적으로 데이터(D2)는 인코딩 스테이지(20)를 통해 더 코딩되어 최종 인코딩 출력 데이터(D3)를 생성하는데, 인코딩 스테이지(20)가 채택되는 않은 경우, 데이터(D2) 및 D3)는 서로 유사하다. 인코딩된 출력 데이터(D3)는 데이터 메모리(30)에 저장되고/되거나 데이터 통신 네트워크(40)를 통해 송신된다. 더구나, 데이터(D4)가 데이터(D2)와 실질적으로 유사하도록 전송한 인코딩 스테이지(20)가 채택되는 경우, 출력 데이터(D3)는 제2 엔트로피 변형기(60)에 직접 전달되거나, 디코딩 스테이지(50)를 통해 제2 엔트로피 변형기(60)에 전달된다. 제2 엔트로피 변형기(60)는 입력 데이터와 실질적으로 유사한 디코딩 출력 데이터(D5)를 생성한다. 이로써, 제1 엔트로피 변형기(10) 및 제2 엔트로피 변형기(60)는 예를 들어 코덱 구성에서와 같이, 상호 인버스(inverse) 엔트로피 변형 동작들을 수행한다.
- [0076] 선택적으로, 제1 및 제2 엔트로피 변형기(10, 60)는 고속 엔트로피 변형, 예를 들어, 고속 인코딩 및/또는 디코딩을 달성하기 위한 복수의 병렬 데이터 스트림으로서 데이터를 프로세싱하도록 동작 가능하다. 대량의 데이터, 예를 들어, 데이터 서버들, 예를 들어 케이블 TV 서비스들을 제공하는 인터넷 기반 서버들부터의 스트리밍 비디오

오 콘텐츠를 처리하는 경우, 복수의 병렬 데이터 스트림을 이용하는 이러한 고속 프로세싱이 유리하다.

[0077] 그러므로 제1 엔트로피 변형기(10)는 데이터(D1)에 존재하는 비트 스트림을 소비하고, 그로부터 입력 데이터(D1)의 엔트로피에 비해 엔트로피가 감소된 데이터 요소들을 생성한다. 선택적으로, 이들 변형된 데이터 요소는 가변 길이 코딩(Variable-Length-Coding, VLC), 허프만 코딩(Huffman coding), 산술 코딩, 레인지 코딩, 런 길이 인코딩(Run-Length-Encoding, RLE) 등과 같은 방법들을 채택함으로써 인코딩 스테이지(30)에서 추가로 인코딩된다. 제1 엔트로피 변형기(10) 내에서 실행되는 엔트로피 변형은 가역적이고, 바람직하게는 무손실적이며, 필요한 부분만 약간 변형되어 제2 엔트로피 변형기(6)에서도 실행된다. 게다가, 제1 엔트로피 변형기(10) 내에서 실행되는 엔트로피 변형은 비트 값의 매 변경 이후 연속적인 유사 비트의 양을 포함하는 새로운 데이터 요소를 적어도 저장한다.

[0078] 전술한 바와 같이 엔트로피 감소를 실행할 때, 임의의 소정의 시퀀스로 제1 비트의 표시가 제공되어야 한다. 게다가, 데이터 요소들이 압축될 때 데이터 요소들을 나타내기 위한 제한적 수의 코드들, 즉 최대 런(maximum run)이 채택되게 하는 이스케이프 코드가 제공되어야 한다. 선택적으로, 이스케이프 코드는 디지털 "0"에 의해 표현되고, 이스케이프 코드는 데이터 요소가 나타낼 수 있는 최고 값, 즉 최대 런을 표시하고, 그리고 비트 값이 계속되는 정보를 추가적으로 전달한다. 바람직하게는, 본 발명의 실시예들을 구현하는 경우, 최대 비트 런에 대응하는 하나 이상의 코드는 데이터 및 그 특성에 따라 달라지는데, 즉 최대 런에 대응하는 하나 이상의 코드는 압축될 데이터에 따라 적응적으로 달라진다. 선택적으로, 엔트로피 변형기(10, 60)는 사전 및/또는 사후 프로세싱 방법들, 예를 들어 1 비트 데이터에 대해 동작하는 오델타 오퍼레이터(ODelta operator)를 활용한다. 그러나 엔트로피 변형기들(10, 60)의 경우, 이를 통해 프로세싱되는 데이터는 임의의 종류, 예를 들어, 이미지 데이터, 비디오 데이터, 오디오 데이터, 참조 데이터, 마스크들, 분할 비트들, 부호 비트들, 및 압축 데이터, 즉 비트로 프로세싱될 수 있는 임의의 부류의 데이터일 수 있다. 선택적으로, 임의의 데이터의 바이트 또는 워드들이 비트들을 포함하는 스트림으로 삽입될 수 있고, 엔트로피 변형기(10, 60)를 통해 프로세싱될 수 있다. 다음으로, "엔트로피 변형기" 및 "인버스 엔트로피 변형기"는 각각 EM 및 IEM으로 축약될 것이며, 즉 제1 및 제2 엔트로피 변형기(10, 60)를 통해 실행되는 데이터 압축 및 데이터 압축 해제를 제공하는데 사용하기 위한 것이다.

[0079] 선택적으로, 엔트로피 변형기(10, 60)는 본 발명에 따른 방법들을 구현하기 위한 기계 판독 가능 데이터 저장 매체(90)에 기록된 하나 이상의 소프트웨어 제품(80)을 실행하도록 동작 가능한 컴퓨팅 하드웨어(CPU; 70)를 이용하여 구현된다.

[0080] 이하, 본 발명의 실시예들에 대해 더 상세히 설명할 것이다. 엔트로피 변형기(10, 60)는 예를 들어, 데이터 비트 스트림에 존재하는 0 또는 1의 최대 런 개수를 정의하는 하나의 파라미터를 채택한다. 게다가, 엔트로피 변형기(10, 60)는 또한 비트 스트림 내의 비트들의 런이 비트 스트림 내의 최대 런보다 더 큰 경우에 사용되는 이스케이프 요소를 채택한다. 바람직하게는, 최대 런 길이(L)는 다음의 시리즈에서 하나 이상으로 설정된다.

수학식 2

[0081]
$$L = 2^n - 1$$

[0082] 여기서, n은 n = 2, 3, ...의 값을 갖는 정수이다.

[0083] 이는 시리즈 L = 3, 7, 15, 31에 대응하지만, 바람직하게는 1보다 큰 임의의 수이다. 바람직하게는, 이스케이프 요소는 "0"의 값으로서 선택되는데, 그 이유는 복수의 제로(0) 유사 비트가 본 발명을 구현할 때 가능하지 않기 때문이다. 유사 비트의 개수가 전술한 바와 같이 1 카운트씩 감소되면, 바람직하게는 이스케이프 요소가 최대 런인 것이 이해될 것이다. 엔트로피 변형기(10)는 비트 스트림의 압축을 바람직한 경우, 그 스트림에서의 제1 비트가 공지될 것을 요구하며, 이로써 대응하는 압축 비트 스트림이 후속으로 디코딩, 즉 엔트로피 변형기(60)에서 가역적으로 압축 해제될 수 있다.

[0084] 본 발명의 실시예들을 더 설명하기 위해, 이하 일례가 설명될 것이다. 이하, 본 발명에 따른 엔트로피 인코딩이, 최대 런 길이가 3 비트이고 이스케이프 코드가 "0"인 경우에 기능하는 방식의 제1 예시가 제시된다.

[0085] 원 비트 스트림(original stream of bits)은 다음과 같다.

[0086] 0 1 1 0 0 1 1 1 1 1 1 0 0 0 0 1 0 0 0 1 1 1 1 1 1

[0087] 이 스트림에는 총 26개의 비트가 포함되는데 16개의 1과 10개의 0이 있다.

[0088] 원 비트 스트림과 관련된 엔트로피(E)는 다음과 같이 계산된다.

수학식 3

$$E = 16 * \log_{10}\left(\frac{26}{16}\right) + 10 * \log_{10}\left(\frac{26}{10}\right) = 7.52339$$

[0089]

[0090] 수학식 3으로부터, 이러한 관련 엔트로피(E)를 갖는 원 비트 스트림을 인코딩하는데 필요한 비트의 최소 개수 (M_B)는 다음과 같이 소스 코딩 이론에 따라 계산 가능하다.

수학식 4

$$M_B = \frac{E}{\log_{10}(2) \text{ 비트}} = 24.99125$$

[0091]

[0092] 전술한 원 비트 스트림은 대응하는 코드 시퀀스를 생성하기 위해 본 발명에 따른 방법을 적용함으로써 인코딩 가능하다.

[0093] 제1 비트 값은 "0"이고, 연속적인 유사 비트들(= 엔트로피 변형 심볼들)의 양은 다음과 같다.

[0094] 1 2 2 7 4 1 3 6

[0095] 이스케이프 심볼 "0"이, 상이한 심볼의 개수를 0(이스케이프) 내지 3(최대 런 길이)인 4개의 상이한 값으로 한정하기 위해 사용되고, 또한 제1 비트 값이 제1 심볼로서 스트림에 추가되면, 다음과 같다.

[0096] 0 1 2 2 0 0 1 0 1 1 3 0 3

[0097] 이 스트림에는 총 13개의 코드가 포함되는데, 5개의 0, 4개의 1, 2개의 2, 및 2개의 3이 존재한다.

[0098] 이러한 코드 스트림과 관련된 엔트로피(E)는 다음과 같이 계산 가능하다.

수학식 5

$$E = 5 * \log_{10}\left(\frac{13}{5}\right) + 4 * \left(\frac{13}{4}\right) + 2 * 2 * \log_{10}\left(\frac{13}{2}\right) = 7.37405$$

[0099]

[0100] 이로부터, 무손실 방식으로 이러한 엔트로피(E)를 표현하기 위해 비트의 최소 개수(M_B)는 다음과 같이 계산 가능하다.

수학식 6

$$M_B = \frac{E}{\log_{10}(2) \text{ 비트}} = 24.49608$$

[0101]

[0102] 이 예시로부터, 일련의 코드에서의 엔트로피가 원 비트 스트림보다 작고, 더 적은 비트들이 일련의 코드를 나타내는데 필요함은 물론이다. 선택적으로, 엔트로피 변형된 일련의 코드는 표 1에서 개요로 나타내는 바와 같이 생성된다.

[0103] 표 1: 엔트로피 변형 코드들의 인버스

코드	지시
0	제1 비트 값
1	제1 비트의 개수: 1개의 "0"
2	변화 이후 비트의 개수: 2개의 "1"
2	변화 이후 비트의 개수: 2개의 "0"
0	4 이상의 상호 유사 비트를 나타내는 이스케이프 코드, 예를 들어, 3개의 "1"
0	4 이상의 상호 유사 비트를 나타내는 이스케이프 코드, 예를 들어, 3개의 "1"
1	변화 이후 비트의 개수, $(7)-2*(\text{이스케이프 비트})$ 여기서 3개의 이스케이프 비트, 즉 1개의 "1"이 존재한다.
0	4 이상의 상호 유사 비트를 나타내는 이스케이프 코드, 예를 들어, 3개의 "0"
1	변화 이후 비트의 개수, $(4)-1*(\text{이스케이프 비트})$, 여기서 3개의 이스케이프 비트, 즉 1개의 "0"이 존재한다.
1	변화 이후 비트의 개수: 1개의 "1"
3	변화 이후 비트의 개수: 3개의 "0"
0	4개 이상의 상호 유사 비트를 나타내는 이스케이프 코드, 즉 3개의 1
3	변화 이후 비트의 개수 $(6)-1*(\text{이스케이프 비트들})$, 여기서 3개의 이스케이프 비트들, 즉 3개의 1이 존재한다.

[0104]

[0105]

표 1은 단지 일례일 뿐이며, 본 발명의 실시예들의 구현은 여러 대체적인 예시들을 생성할 수 있다. 예를 들어, 다른 최대 런 값들 및 다른 이스케이프 코드들이 채택될 수 있다. 더구나, 제1 비트는 선택적으로 별도로 삽입되고, 채택된 실제 코드들 및 이들의 관련 압축에 어떤 유효한 정도로도 영향을 미치지 않는다.

[0106]

선택적으로, 엔트로피 변형기(10, 60)는 소정의 상황에 있어서 엔트로피를 증가시키도록 동작 가능하며, 그렇기 때문에, 채택된 코딩 레짐(coding regime)이 원하는 정도의 데이터 압축을 제공하는지 여부를 결정하기 위해, 예를 들어 신호 모니터링 배열(arrangement)을 이용하여 엔트로피 변형기(10, 60)로부터의 출력 데이터가 모니터링되는 것이 바람직하다. 전달될 소정의 타입의 데이터에 대해 엔트로피가 증가하는 이벤트에서, 신호 모니터링 배열은 엔트로피에서의 원하는 변형이 동작 중에 달성되는 것을 보장하도록 적응형 방식으로 엔트로피 변형기(10, 60)의 동작을 제어하기 위해 채택될 수 있다. 예를 들어, 소정 타입의 데이터의 경우, 최대 런의 최적 값을 검색하여 최적 데이터 압축을 획득하도록 채택하기 위해 데이터 스트림의 인코딩을 개시할 때 하나 이상의 테스트 코딩 레짐들을 적용하기 위해 사용되는 것이 바람직하며, 여기서 최대 런은 엔트로피(E)에 주요한 영향을 미칠 수 있다. 추가적으로, 소정의 데이터 스트림의 길이가 상대적으로 짧은 경우, 예를 들어, 제1 비트 값이 바람직하게는 소정의 데이터 스트림을 나타내는데 사용되는 다른 코드 워드들과 별도로 송신되는 경우, 소정의 데이터 스트림에서 제1 비트 값과 관련된 엔트로피(E)는 엔트로피(E)에 주요한 영향을 미친다.

[0107]

도 1에 예시된 바와 같이, 가변 길이 코딩(Variable-Length-Coding, VLC), 허프만 코딩(Huffman coding), 런 길이 인코딩(Run-Length-Encoding, RLE) 등과 같은 엔트로피 코딩 방법들과 함께 엔트로피 변형기(10, 60)가 채택되는 것이 바람직하다. 다른 방법으로, 바람직하게는, 델타 코딩(Delta coding)과 같은 다른 타입의 엔트로피 변형기들이 이들과 함께 채택된다. 바람직하게는, 이러한 엔트로피 코딩 방법은 엔트로피 변형기(10) 다음 그리고 엔트로피 변형기(60) 앞에 채택되는데, 다른 방법으로 또는 추가적으로, 이러한 엔트로피 코딩 방법들은 엔

트로피 변형기(10) 앞에 그리고 엔트로피 변형기(60) 다음에 채택되며, 예를 들어, 인코딩될 원 데이터가 영숫자, 숫자, 바이트, 또는 워드가 이벤트에서, 원 데이터는 표 2에 제공되는 바와 같이 추가 코딩을 이용하여 효율적으로 코딩된다.

[0108] 표 2: 본 발명에 따른 엔트로피 변형기를 이용하여 채택 가능한 추가 코딩

추가 인코딩 방법	사용 상황
RLE	소정의 데이터 스트림이 동일한 데이터 값들의 연속 스트림을 포함하는 경우
델타	소정의 데이터 스트림이 유사하거나 상호 실질적으로 유사한 데이터 값들을 포함하거나 유사하게 변하는 경우
VLC/허프만/산술/레인지	소정의 데이터 스트림이 여러 상호 유사한 데이터 값들을 포함한 경우

[0109]

[0110] 선택적으로, 엔트로피 변형기(10, 60)는 반복적 방식으로 채택된다.

[0111] 진술한 바와 같이, 바람직하게는, 본 발명에 따른 엔트로피 변형기들이 다른 타입의 엔트로피 변형기들과 함께 채택되며, 여기서 선택적으로 다른 타입의 엔트로피 변형기들은 공지된 타입이다. 예를 들어, 공지 타입들은 델타(Delta) 코딩을 포함한다. 통상적으로, 델타 코딩은 바이트들 또는 워드들의 포맷으로 제공되는 수치 값들을 프로세싱하기 위해 채택된다. 다음으로, 델타 코딩과 함께 채택되는 본 발명의 엔트로피 변형기의 일례가 설명될 것이다.

[0112] 본 발명을 더 설명하기 위해, 두 가지 예시가 설명될 것이며, 일례는 본 발명에 따른 엔트로피 변형기의 사용하지 않고, 다른 예는 본 발명에 따른 엔트로피 변형기를 활용한다. 이 예시에서, 7 비트의 최대 런 길이를, "0"인 관련 이스케이프 코드와 함께 채택한다. 1 비트 오프셋 타입 오퍼레이터를 채택하는 경우, 인코딩된 시퀀스에서의 제1 비트는 인코딩될 원 데이터 스트림에 존재하는 대응하는 원 데이터 비트와 동일하다. 선택적으로, 1 비트 오프셋 연산이 반복적으로 실행될 수 있다.

[0113] 예시 1:

[0114] 원 데이터 비트 스트림에서, 총 37개의 비트, 즉 17개의 1 및 20개의 0이 포함된다.

[0115] 0 1 0 1 0 1 1 0 0 1 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1

[0116] 여기서, 대응하는 관련 엔트로피(E)는 다음과 같이 계산 가능하다.

수학식 7

$$E = 17 * \log_{10}\left(\frac{37}{17}\right) + 20 * \log_{10}\left(\frac{37}{20}\right) = 11.08523$$

[0117]

[0118] 계산된 M_B 를 위한 36.82개의 비트, 즉 앞서 제공된 바와 같이 실질적으로 37개의 비트에 대응한다. 수학식 7이 원 비트 스트림에서의 비트량에 대응함은 물론이다.

[0119] 등가의 엔트로피 변형(EM) 코드는, 표 1을 생성하기 위해 채택된 것과 일반적으로 유사한 방식으로 다음과 같다.

[0120] 0 1 1 1 1 1 2 2 1 3 1 1 1 0 4 0 3

[0121] 여기서, 관련 엔트로피(E)는 다음과 같이 계산 가능하다.

수학식 8

$$E = 3 * \log_{10}\left(\frac{17}{3}\right) + 9 * \log_{10}\left(\frac{17}{9}\right) + 2 * 2 * \log_{10}\left(\frac{17}{2}\right) + 1 * \log_{10}\left(\frac{17}{1}\right) = 9.69397$$

[0122]

[0123]

수학식 8이 엔트로피 변형 이후의 비트량에 대응한다는 점은 물론이다. 게다가, 이 비트량이 수학식 7과 관련된 비트량 미만임은 물론이다.

[0124]

그러나, 전술한 원 비트 스트림이 오델타 연산의 대상인 경우, 37개의 비트를 포함하는 대응하는 변형 비트 스트림이 생성되는데, 여기서 13개의 "1" 및 24개의 "0"이 존재한다.

[0125]

0 1 1 1 1 1 0 1 0 1 1 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0

[0126]

여기서, 대응하는 엔트로피(E)는 다음과 같이 계산 가능하다.

수학식 9

$$E = 13 * \log_{10}\left(\frac{37}{13}\right) + 24 * \log_{10}\left(\frac{37}{24}\right) = 10.41713$$

[0127]

[0128]

즉, 원 비트 스트림과 관련된 엔트로피(E)(도 7)보다 더 작다. 전술한 이 오델타 프로세싱된 데이터 비트 스트림이 본 발명에 따른 엔트로피 변형의 대상이면, 이로써 아래와 같이 대응하는 인코딩된 코드 시퀀스가 획득된다.

[0129]

0 1 5 1 1 1 2 2 4 0 3 1 0 2

[0130]

여기서, 대응하는 엔트로피(E)는 다음과 같이 계산 가능하다.

수학식 10

$$E = 2 * 3 * \log_{10}\left(\frac{14}{3}\right) + 5 * \log_{10}\left(\frac{14}{5}\right) + 3 * 1 * \log_{10}\left(\frac{14}{1}\right) = 9.68822$$

[0131]

[0132]

총 37개의 비트를 포함하는 원 데이터 비트 스트림(수학식 7)의 엔트로피(E) 미만이다. 게다가, 이러한 비트량이 수학식 8 또는 수학식 9과 관련된 비트량 미만임은 물론이다.

[0133]

이 예시는 원 데이터 스트림이 1 비트 오델타 오퍼레이터를 이용하여 먼저 압축되어 본 발명에 따른 엔트로피 변형기를 이용하여 압축되는 경우 원 데이터 비트 스트림을 효율적으로 압축하는 것이 가능하다는 것을 나타낸다.

[0134]

예시 2:

[0135]

예시 1에서의 전술한 원 비트 스트림은 2개의 부분으로 분할되고, 본 발명에 따른 엔트로피 변형의 이점들은 이들 2개의 최종 데이터 스트림의 최상 조합을 선택함으로써 최적화된다. 엔트로피 변형기의 제1 비트의 값은 별도로 송신되며, 결과적인 엔트로피 감소한 일련의 코드로 삽입되지 않는다. 획득 가능한 제1 엔트로피 변형 결과는 예시 1과 유사하며, 여기서 제1 비트 또한 "1" 코드 워드라고 설정된다. 표 3에 제공된 바와 같이, 7개의 비트의 유사 최대 런이 채택되고, "0"이 이스케이프 코드에 채택된다.

[0136] 표 3: 예시 2 엔트로피 변형

	비트 스트림	엔트로피(E)	인코딩할 최대 비트 (M_B)
원 비트 부분 1 (16비트, 7개의 1 및 9개의 0)	010101100100010 1	4.7621	15.82 비트
엔트로피 변형 (EM)	0111112213111	5.2910	17.58 비트
엔트로피 변형 (EM)	(0)1111122131111	3.7599	13.49 비트
오델타 변형	0111110101100111	4.3158	14.34 비트
EM	(0)15111223	4.2144	15.00 비트
원 비트 부분 2 (21비트, 11개의 1 및 10개의 0)	0000000000011111 11111	6.3113	20.97 비트
엔트로피 변형 (EM)	(0)0403	1.08062	6.00/7.00 비트
오델타	000000000001000 000000	1.7460	5.80 비트
EM	(0)04102	2.8928	9.61/10.61비트

[0137]

[0138]

표 3으로부터, 엔트로피 감소가 달성되며, 이것이 데이터 압축이 달성될 때 유리한 것은 물론이다. 데이터의 제 1 부분 스트림이 엔트로피 변형된, 즉 13.49개의 비트를 이용하여 나타낸 원 데이터 스트림으로부터 생성된 데이터로부터 엔트로피 코딩되는 경우 데이터의 최선 압축이 달성되고, 제2 부분은 요구되는 원래 37개의 비트와 비교해 19.29개의 비트를 이용하여 표현 가능하도록 1 비트 오델타 코딩을 통해 엔트로피 변형되는데, 즉 5.80개의 비트를 이용하여 나타내진다. 제2 데이터 스트림이 코딩되는 경우, 제1 비트의 값을 전송할 필요가 없는데, 그 이유는 제1 데이터 스트림에 기반하여 공지될 수 있다. 이 때문에, "/"에 의해 구분된 표 3에 도시된 2개의 상이한 값들이 존재하는 경우, 제1 값은 연속 상황(즉, 전송되지 않은 제1 비트 값)을 위한 것이고, 제2 값은 이러한 종류의 비트 조합에서 데이터 스트림이 개시된 경우에 관한 것이다. 이 예시에서, 오델타 코딩만이 데이터의 엔트로피를 즉 대개 데이터의 제2 부분을 위해, 감소시킴에도 불구하고, 엔트로피 변형기(10, 60)와 함께 바람직한 엔트로피 감소를 제공함은 물론이다. 여기에서, 엔트로피 변형된 원 데이터는 오델타 코딩을 이용하여 먼저 프로세싱되는 엔트로피 변형된 데이터보다 작은 엔트로피, 즉 6.00 개의 비트들 < 9.61 개의 비트들을 초래한다. 실제 코딩 솔루션에서, 선택적으로, 오델타 코딩 데이터의 엔트로피가 더 작음에도 불구하고, 바람직하게는 엔트로피 변형된 원 데이터는 오델타 코딩 데이터보다 더 효율적으로 압축되는데; 무엇보다 산술 코딩이 바람직하게는 1 비트 데이터를 압축하기 위해 채택됨에도, 1 비트 데이터는 엔트로피 인코더들의 구현을 위해 실용적인 것은 아니다. 선택적으로, 데이터 압축을 달성하기 위한 엔트로피 변형 프로세스들의 대상이 되기 이전에, 원 비트 스트림이 3개의 부분 이상으로 분할된다.

[0139]

선택적으로, 원 데이터 비트 스트림을 하나 이상의 스트림, 즉 부분으로 분할하는 것은,

[0140]

(a) 원 데이터 비트 스트림과 관련된 엔트로피를 분석하는 단계;

[0141]

(b) 원 데이터 비트 스트림의 분석된 엔트로피의 함수에 따라 원 데이터 스트림을 2개 이상의 부분으로 분할하는 단계;

[0142]

(c) 하나 이상의 엔트로피 변형 연산을 2 이상의 부분에 적용하여 하나 이상의 대응하는 엔트로피 변형 부분을

생성하는 단계; 및

- [0143] (d) 하나 이상의 엔트로피 변형 부분을 결합하여 압축된 인코딩 출력 데이터를 생성하는 단계를 포함하는 방법에 의해 자동 실행될 수 있다.
- [0144] 본 발명은 또한 인코딩된 출력 데이터를 디코딩하는 방법에 관한 것으로서, 단계들 (a) 내지 (d)의 인버스가 활용된다. 선택적으로, 이 방법은 원 비트 스트림에 존재하는 데이터 블록들을 분할 또는 결합하는 단계를 포함하는데, 이로써 예를 들어 향상된 압축비를 달성하고/하거나 실질적으로 무손실 압축을 달성하기 위해 인코딩이 더 효율적으로 구현된다.
- [0145] 선택적으로, 단계 (b)는 조약한 방식으로 구현되는데, 즉 다수의 롱(long) 런 섹션들이 원 데이터 비트 스트림에서 마주치는 경우, 즉 데이터 비트가 급속한 방식으로 순차적으로 달라지는 데이터 비트 스트림의 충분히 큰 영역이 존재하는 원 데이터 비트 스트림의 영역 이후 또는 그 사이에, 원 데이터 비트 스트림을 단지 분할한다.
- [0146] 선택적으로, 본 발명에 따른 인코딩된 비트 스트림을 생성하기 위해, 원 데이터 비트 스트림이 예를 들어, 엔트로피가 압축 및 감소된 출력 인코딩 데이터를 생성하기 위한 본 발명에 따른 엔트로피 변형의 대상이 되는 VLC, 산술 또는 레인지 코딩 및/또는 RLE의 적용을 통해 중간 인코딩 데이터를 제공하도록 프로세싱된다. 이로써, 데이터 압축 중의 상당한 속도 최적화가 달성되는데, 이는 본 발명이 기계 판독 가능 데이터 저장 매체에 기록된 하나 이상의 소프트웨어 제품에 의해 구현되는 경우 필요한 더 적은 컴퓨팅 전력에 대응하며, 여기서 하나 이상의 소프트웨어 제품은 엔트로피 변형 인코더를 구현하고, 필요한 부분만 약간 변형되어 엔트로피 변형 디코더를 구현하기 위한 컴퓨팅 하드웨어에서 실행 가능하다. 선택적으로, 원 데이터 비트 스트림에서의 원 비트, 알파벳, 숫자, 바이트, 워드가 일부 다른 방법을 이용하여 먼저 코딩되고, 그 후 본 발명에 따른 엔트로피 변형이 인코딩된 출력 데이터를 생성하도록 엔트로피 최적화가 구현된다.
- [0147] 동시대의 공지된 런 길이 인코딩(RLE) 방법들에서, 소정의 값이 생성되고, 다른 후속 값이 따르기 전에, 그 후 이러한 소정의 값의 생성 횟수를 나타내는 대응 데이터가 생성된다. 인코딩될 입력 데이터가 이용 가능한 한 이러한 RLE 방법이 계속되는데, 즉 다른 값 이후의 하나의 값이 순차적으로 인코딩된다. 대조적으로, 본 발명은 단지 1 비트 값들이 채택되면, 이들 값이 차례로 번갈아 일어나기 때문에 인코딩될 데이터 스트림으로 "0" 및 "1" 데이터 값들을 추가할 필요가 없다는 관측으로부터 도출되며, 이로써 인코딩 데이터의 전송 시에 소정의 시퀀스에서의 제1 값만이 표현되면 충분하다. 이로 인해, 본 발명은 RLE 코딩과 유사하지만, 전술한 바와 같이 1 비트 데이터 스트림을 채택한다.
- [0148] 본 발명의 실시예들을 구현하는 경우, 이스케이프 코드는 실제 인코딩 방법들의 더 효율적인 사용이 채택될 수 있게 하는 부가물이다. 바람직하게는, 본 발명의 실시예들은 인코더로부터 대응하는 디코더로 하나 이상의 코딩 테이블을 전송하여 인코더로부터 전송되어 디코더에서 수신된 인코딩 데이터의 디코딩을 보조하는 단계를 포함한다. 이러한 코딩 테이블들을 전달하는 일부 매우 유리한 방법들은, 예를 들어 여기에 참조로서 통합된 2014년 2월 20일에 출원인에 의해 출원된 특허출원 제GB1403039.9호에 개시되어 있다. 또한, 참조로서 통합된 2014년 2월 20일에 출원인에 의해 출원된 특허출원 제GB403038.1호에 개시되어 있다. 앞서 개시된 양자의 방법은 본 발명의 실시예들과 함께 사용하기에 잘 어울리며, 엔트로피 코딩에 필요한 추가 인코딩 데이터를 최소화하는 것을 보조한다.
- [0149] 본 발명의 실시예들을 구현하기 위해 채택된 이스케이프 코드들은 여러 상이한 방식으로 용이하게 구현될 수 있다. 예를 들어, "0" 값이 계속되는 런-길이 심볼 또는 디지털의 경우, 만족스러운 인코딩이 달성 가능하다. 그러나 이들 양의 값들이 작은 상황들이 잠재적으로 일어나는데, 반대로 이들 상황에서 매우 큰 값들이 가끔 발생할 수 있으며, 바람직하게는, 이스케이프 코드가, 아직 사용되지 않거나, 코드 워드로서 사용하기 위해 할당된 다음 값으로 설정된다. 예를 들어, 채택된 최대 런-길이가 7개의 비트 또는 요소인 경우, 및 이스케이프 코드가 "0"이면, 코드 워드 "8"은 인코딩된 데이터 스트림의 생성 중에 사용되는데, 이는 새로운 이스케이프 코드는 하나의 새로운 코드 워드만을 추가하고, 그 후 심볼의 실제 개수는 예를 들어, 1, 2, 및/또는 3 바이트에서의 임의의 방법을 이용하거나, 하프 바이트(즉, 니블(nibble))를 이용하여 표현 가능하다. 선택적으로, 심볼들의 실제 개수는 압축되지 않고 송신된다. 선택적으로, 심볼들의 실제 개수는 최대 런 길이 값과 조합된 다음, 엔트로피 압축기, 즉 엔트로피 변형기 내의 다른 데이터 값과 함께 압축된다.
- [0150] 본 발명의 실시예들에서 사용하기 위한 적절한 이스케이프 값을 선택하는 경우, 일단 총 개수의 인접 요소들을 표현하는 것이 바람직하며, 이로써 이러한 이스케이프 값을 불러온 이후, 동일 비트가 이스케이프 값의 사용과 대조적으로 후속해서 연속되지 않는다. 이러한 새로운 종결 형태는 전술한 바와 같은 연속적인 이스케이프 방법

뿐만 아니라 사용 가능한 이스케이프, 즉 코드 워드를 제공하며, 다른 방법으로서, 이러한 새로운 이스케이프 코드가 하나의 대안으로서 사용 가능하며, 이 경우 자신의 코드 워드는 바람직하게는 "0"으로 설정 가능하다. 이와 유사하게, 이러한 종결 코드의 소정의 데이터 값이 종종 작지만, 가끔 크면, 교번하는 개수의 바이트 또는 니블들을 이용하여 데이터에 존재하는 연속적인 유사한 심볼들의 개수를 표현하는 것이 바람직하다. 이러한 바이트 중 하나의 비트, 예를 들어, 최하위 비트(least significant bit, LSB) 또는 최상위 비트(most significant bit, MSB)는 그 수, 즉 비트 스트링의 종결을 지시하기 위해 바람직하게는 예약되며, 이로써 값이 종료된 경우, 또는 다른 경우에 값이 계속되는 경우 자신의 코드 워드가 값 "0" 또는 "1"로 설정된다.

[0151] 전술한 내용을 설명하기 위해, 다음으로, 이스케이프 코드의 기능성의 일례를, 양쪽 타입의 이스케이프가 활용되는 상황에서 설명할 것이다. 일례에서, 14개의 요소의 최대 런이 채택되는데, 연속적인 이스케이프를 위한 코드 워드가 15에 의해 표시되고, 여기서 종결 이스케이프를 위한 코드는 0이다. 게다가, 연속적인 유사 심볼 또는 비트들의 양에 대한 값들은 다음과 같다.

[0152] 9, 1024, 16, 9, 12, 2000, 7, 20, 21, 6, 8, 120, 12, ...

[0153] 또한, 이 예시는 값 "1"의 최상위 비트(MSB)가 그 수, 즉 비트 스트림의 종결을 지시하기 위해 채택된 하프 바이트들(예를 들어, 니블들)을 활용한다. 바람직하게는, 종결 이스케이프가 채택되면, 그 수, 즉 비트 스트림으로부터 차감하는 것이 다음과 같이 실행 가능하다.

[0154] (비트 스트링) - (MAXRUN + 1)

[0155] 예를 들어:

$$1024_{\text{base } 10} \Rightarrow 1024_{\text{base } 10} - (14_{\text{base } 10} + 1) \Rightarrow 1009_{\text{base } 10} \Rightarrow 1 \ 111 \ 110 \ 001_{\text{base } 10 \text{ number expressed for nibbles}} \Rightarrow 1 + 8(\text{Terminates}) \ 7 \ 6 \ 1).$$

[0156] [0157] 이로써, 이 예시는 다음과 같이 코딩될 수 있다.

[0158] 9 0 1 6 7 9 1 5 2 9 12 0 1 0 7 11 7 1 5 6 15 7 6 8 0 1 5 9 12

[0159] 이 예시에서, 엔트로피 변형기(10, 60)는 0 내지 15 범위의 값을 갖는 코드 워드들을 해석할 수 있어야 한다. 단일 이스케이프 코드만을 이용함으로써 이러한 예시를 표현하는 것이 실행 가능하며, 여기서 코드 워드 "0"는 또한 종결 이스케이프를 지시하는 이스케이프 코드에 대응하는데, 단일 타입의 이스케이프 코드의 이러한 대체 예시에서, 연속적인 이스케이프가 사용되지 않고, 다른 파라미터들은 전술한 바와 같이, 즉 다음과 같다.

[0160] 9 0 1 6 7 9 0 9 9 12 0 1 0 7 11 7 0 13 0 14 6 8 0 1 5 9 12

[0161] 여기서, 엔트로피 변형기(10, 60)는 0 내지 14의 범위에 있는 코드 워드들을 인식하는 것이 요구된다. 대체 예는 코드 워드 "0"에 의해 표현된 연속적인 이스케이프만을 활용하고, 14의 최대 런 길이를 채택함으로써 선택적으로 표현되며, 이로써 결과적으로 다음 형태의 코드 시퀀스를 초래한다.

[0162] 9 73x0 2 0 2 9 12 142x0 12 7 0 6 7 6 8 8x0 8 12

[0163] 최선 코딩 솔루션은 3개의 모든 솔루션에 대해 별도로 계산되는 엔트로피에 기반하여 선택될 수 있다.

[0164] 엔트로피 변형(EM) 및 인버스 엔트로피 변형(IEM)을 제공하기 위한 본 발명의 실시예들은 다음에 설명될 것이다. 그러나, 다른 실시예들, 예를 들어, 가변 상태 기계, ASIC 등과 같은 디지털 하드웨어에 기반한 대응하는 엔트로피 변형기들은 또한 본 발명의 범위 내에서 실행 가능하다. 다음의 예시들에서 "GetBit", "SetBit" 및 "ClearBit" 명령어들은 항상 "HeaderBits" 값들을 업데이트하는 기능을 한다. 더구나, 다음 비트가 다음 바이트에 있을 경우 "HeaderIndex" 값 또한 업데이트된다. 소프트웨어 제품에 기반한 다음의 예시들에서, 제1 비트는 하나의 코드로서 저장된다. 선택적으로, 더 나은 구현예들에서, 제1 비트 값은 예를 들어, 도 3을 참조하여 전술한 바와 같이 코드들로부터 분리된다. 바람직하게는, 디코딩된 출력 데이터에 기여한 목적지 바이트에 다수의 비트를 함께 기입함으로써 대응하는 디코더가 최적화된다. 다음의 예시적인 실시예들에서 제공되는 "MaxRun"가 (최대 런 길이 + 1)와 유사함은 물론이며, 예를 들어, 7 비트의 최대 런 길이가 채택되는 경우, 8의 값이 다음 실시예들에서 함수 파라미터 "AMaxRun"로서 주어진다.

```

function EncodeBitRLE1u(APtrSrc : PByte; ASrcBitLen : Cardinal; APtrDst : PByte; var ADstBitOffset :
Cardinal; AMaxRun : Cardinal) : Boolean;
var
  iHeaderIndex, iRunLength, iIndex : Cardinal;
  bBit, bLastBit : Boolean;
begin
  // Reset offsets
  ADstBitOffset := 0;
  iHeaderIndex := 0;
  iHeaderBits := 0;

  // Read first bit, write it to destination and initialize run
  bLastBit := GetBit(APtrSrc, @iHeaderIndex, @iHeaderBits);
  APtrDst[(ADstBitOffset + 7) div 8] := Byte(bLastBit);
  Inc(ADstBitOffset, 8);
  iRunLength := 1;

  // Go through bits
  for iIndex := 1 to ASrcBitLen-1 do
  begin
    // Read bit
    bBit := GetBit(APtrSrc, @iHeaderIndex, @iHeaderBits);

    // Same bit as previous
    if (bBit = bLastBit) then
    begin
      // Increment run of bits
      Inc(iRunLength, 1);

      // Escape (same bits continuous over maximum run)
      if(iRunLength = AMaxRun) then
      begin
        // Write escape code to destination and initialize run
        APtrDst[(ADstBitOffset + 7) div 8] := 0;
        Inc(ADstBitOffset, 8);
        iRunLength := 1;
      end
    end

    // Different bit as previous
  end
end

```

[0165]

```

else
begin
  // Change bit, write run of bits to destination and initialize run
  bLastBit := bBit;
  APtrDst[(ADstBitOffset + 7) div 8] := iRunLength;
  Inc(ADstBitOffset, 8);
  iRunLength := 1;
end;
end;

// Write last bits
APtrDst[(ADstBitOffset + 7) div 8] := iRunLength;
Inc(ADstBitOffset, 8);
end;

function DecodeBitRLE1u(APtrSrc : PByte; ASrcBitOffset : Cardinal; APtrDst : PByte; var ADstBitLen :
Cardinal; AMaxRun : Cardinal) : Boolean;
var
  bBit : Boolean;
  Value : Byte;
  iSrcBitOffset, iIndex, iRunLength, iHeaderIndex, iHeaderBits, iDstBitOffset : Cardinal;
begin
  // Reset srcbitoffset, dstbitoffset and clear first destination byte
  iSrcBitOffset := 0;
  iDstBitOffset := 0;
  iHeaderIndex := 0;
  iHeaderBits := 0;
  APtrDst[0] := 0;

  // Read first bit value (= byte) and update srcbitoffset
  bBit := (APtrSrc[(iSrcBitOffset + 7) div 8] > 0);
  Inc(iSrcBitOffset, 8);

  // Go through all Bytes until source is finished or destination buffer is full
  while ((iSrcBitOffset < ASrcBitOffset) and (iDstBitOffset < ADstBitLen)) do
  begin
    // Read runlength and update srcbitoffset
    iRunLength := APtrSrc[(iSrcBitOffset + 7) div 8];
    Inc(iSrcBitOffset, 8);

    // Process escape code

```

[0166]

```

if (iRunLength = 0) then
begin
  // Set bits for escape
  for iIndex := 1 to AMaxRun - 1 do
  begin
    // Set bit if destination buffer is not full
    if (iDstBitOffset < ADstBitLen) then
    begin
      // Set bit and/or go to next bit
      if (bBit) then
        SetBit(APtrDst, @iHeaderIndex, @iHeaderBits)
      else ClearBit(APtrDst, @iHeaderIndex, @iHeaderBits);
      Inc(iDstBitOffset, 1);
    end;
  end;
end

// Process runlength
else
begin
  // Set bits for run
  for iIndex := 0 to iRunLength - 1 do
  begin
    // Set bit if destination buffer is not full
    if (iDstBitOffset < ADstBitLen) then
    begin
      // Set bit and/or go to next bit
      if (bBit) then
        SetBit(APtrDst, @iHeaderIndex, @iHeaderBits)
      else ClearBit(APtrDst, @iHeaderIndex, @iHeaderBits);
      Inc(iDstBitOffset, 1);
    end;
  end;

  // Change bit value
  if (bBit) then
    bBit := False
  else bBit := True;
end;
end;

```

[0167]

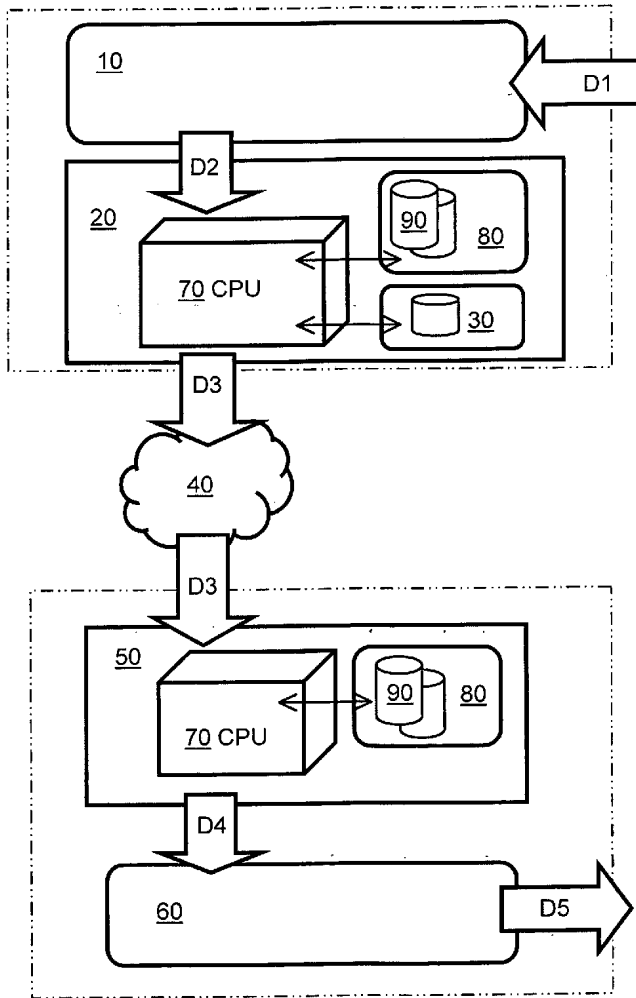
[0168]

예를 들어, VLC, 허프만 코딩, 산술 코딩, 레인지 코딩, RLE 코딩 등과 같은 엔트로피 코딩 방법이 즉, 예측 또는 다른 추가 정보의 형태를 이용하지 않거나, 정보의 손실이 일어나지 않고, 자신의 엔트로피 허용치보다 더 작은 사이즈로 데이터를 압축하는 것이 가능하지 않음에도 불구하고, 이는 모든 경우에 실제로 최대 데이터 압축이 최소 엔트로피에 대응하는 것을 의미하지 않는데, 다시 말하면, 실제로 엔트로피 및 데이터 압축은 정확하게 동일한 개념은 아니다. 인코딩 데이터를 자신의 대응하는 원 형태로 다시 디코딩하는 것은 일부 추가 정보를 항상 요구하지 않고, 때때로 이들 추가 정보의 전송은 압축될 실제 데이터뿐 아니라 불합리한 양의 여분 정보를 요구한다. 대량의 데이터가 처리될 경우, 가장 작은 엔트로피가 최선의 데이터 압축에 대응하지만, 작은 양의 데이터가 처리될 때, 데이터를 압축하는 것은 항상 그렇게 직접적인 것은 아니며, 예를 들어, 1 비트 데이터는 대개 효율적으로 압축하는 것이 어렵고, 그러므로 1 비트 정보를 처리하도록 동작 가능한 엔트로피 변형기(10, 60)는 잠재적 그리고 상업적으로 매우 가치 있는 발명이다. 더구나, 더 쉽게 압축 가능한 포맷으로 데이터를 프로세싱하는 것은 종종, 프로세스들을 포맷화하는 것이 적용되기 이전에 대응하는 원 데이터에 비해 포맷화된 데이터의 엔트로피의 감소를 초래한다.

- [0169] 전술한 바와 같이, 인코딩 데이터를 나타내는 코드 워드들의 시퀀스에서의 제1 비트 값이 연속 심볼 또는 비트의 개수를 나타내는 이들 값과 별도로 전송되면, 종종 효율적이다. 그러나, 이러한 별도의 전송은 완전히 별도의 통신에 대응할 필요가 없다. 다음의 예시에서, 제1 비트의 별도의 처리에도 불구하고, 연속 요소들의 총량에 제1 비트량을 추가할 필요가 여전히 존재하는데, 그 이유는 다음 예시들을 구별하는 것이 가능하지 않을 수 있기 때문이다.
- [0170] 0 0 1 및 0 1 또는
- [0171] 1 1 0 및 1 0,
- [0172] 제1 요소의 개수를 0으로 설정하지 않고, 전술한 이스케이프 코드를 구현하기 위해 채택될 수 있으면, 즉 연속적인 유사 신호들의 개수가 예시들에서 최대 런 길이를 초과할 때 사용하기 위해 채택될 수 있으면 바람직하지 않을 수 있다.
- [0173] 도 1에서, 데이터(D2)의 사이즈는 데이터(D1)의 사이즈보다 필수적으로 더 작은 것은 아니지만, 데이터(D1 및 D2)가 상호 유사한 정보를 포함한 경우에도 데이터(D2)의 엔트로피가 더 작다. 데이터(D2)의 사이즈는 채택된 런 길이에 종속적이다. 예를 들어, 데이터(D1)에서, 데이터 요소들의 사이즈는 1 비트인 반면, 데이터(D2)의 사이즈는 채택된 런 길이들의 사이즈에 의존한다. 예를 들어, 데이터(D1)가 104 비트의 사이즈를 갖고, 엔트로피(E) = 80이면, 데이터(D1)는 데이터(D2) 내의 7개의 요소들의 최대 런 길이를 이용함으로써 엔트로피 변형기(10)에 의해 변환될 수 있으며, 여기서 데이터(D2)는 40개의 요소, 즉 0 내지 7의 범위에 있는 값을 갖는 3 비트 요소를 갖고, 엔트로피(E) = 50이며, 그 후 데이터(D2)는 40*3 비트 = 120 비트의 사이즈를 갖는데; 이러한 예시에서, 데이터(D2)는 데이터(D1)보다 사이즈가 크다. 바람직하게는, 인코딩 스테이지(20)는 데이터(D2)를 52 비트로 압축하여 데이터(D3)를 제공할 수 있는 VLC 변환을 채택한다. 이에 따라, 엔트로피 변형기(10, 60)는 각각 데이터 압축 및 압축 해제를 항상 제공하지 않지만, 다른 인코딩 스테이지들을 이용하여 채택되는 경우 데이터 압축 및 압축 해제를 제공할 수 있음은 물론이다.
- [0174] 바람직하게는, 본 발명의 실시예들은 몇몇 예시들을 언급하기 위해 통신 시스템들, 시청각 소비자 제품들, 과학적 이미지 프로세싱 장비, 컴퓨팅 디바이스들, 스마트폰과 같은 무선 가능 통신 디바이스들, 디지털 카메라, 디지털 감시 장비, 및 쌍방향 컴퓨터 게임의 컴포넌트 부분들로서 포함된다. 데이터 압축이 증가하면 데이터 저장 조건들을 감축시키고, 에너지를 절감할 수 있으며, 고속 통신이 달성될 수 있게 하는데, 이들 모두는 예를 들어, 산업에서 채택 가능한 유리한 기술적 효과들로서 간주될 것이다. 더구나, 실시예들은 소프트웨어 제품들, 하드웨어, 또는 이들의 임의의 조합으로 구현 가능하다. 선택적으로, 소프트웨어 제품들은 소프트웨어 애플리케이션, 즉 "소프트웨어 앱"으로서 다운로드 가능하다.
- [0175] 전술한 본 발명의 실시예들에 대한 변형은 다음의 청구범위에 의해 정의된 본 발명의 범위에서 벗어나지 않고 가능하다. 본 발명을 설명하고 주장하는데 사용되는 "포함하는(including)", "포함하지만 이로 제한되지 않는(comprising)", "통합하는(incorporating)", "구성되는(consisting of)", "갖다(have)", "이다(is)"와 같은 표현들은 비-배타적인 방식으로 간주되는 것을 의도하는데, 즉 명확히 설명되지 않은 아이템들, 컴포넌트들, 또는 요소들이 제시되는 것을 가능하게 한다. 단수에 대한 참조는 또한 복수에 관한 것으로 간주될 수 있다. 첨부된 청구범위에서 괄호 내에 포함된 숫자들은 청구범위의 이해를 보조하기 위한 것으로서, 이 청구범위에 의해 주장되는 주제를 한정하는 것으로 임의의 간주되지 않아야 한다.

도면

도면1



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 45

【변경전】

저장 매체

【변경후】

기계 판독 가능한 저장 매체