



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0065306  
(43) 공개일자 2022년05월20일

(51) 국제특허분류(Int. Cl.)  
G06N 3/08 (2006.01) G06N 3/04 (2006.01)  
G06T 9/00 (2019.01)  
(52) CPC특허분류  
G06N 3/08 (2013.01)  
G06N 3/04 (2013.01)  
(21) 출원번호 10-2020-0151675  
(22) 출원일자 2020년11월13일  
심사청구일자 2020년11월13일

(71) 출원인  
네이버웹툰 유한회사  
경기도 성남시 분당구 분당내곡로 117, 9층(백현동)  
연세대학교 산학협력단  
서울특별시 서대문구 연세로 50 (신촌동, 연세대학교)  
(72) 발명자  
서충현  
경기도 성남시 분당구 분당내곡로 117, 9층(백현동, 크래프톤타워)  
장재혁  
경기도 성남시 분당구 분당내곡로 117, 9층(백현동, 크래프톤타워)  
(뒀면에 계속)  
(74) 대리인  
양성보

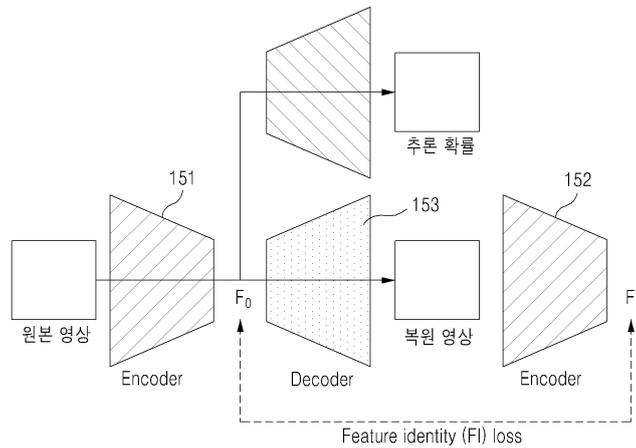
전체 청구항 수 : 총 11 항

(54) 발명의 명칭 연속 심층 영상 압축을 위한 장치 및 방법

(57) 요약

연속 심층 영상 압축을 위한 장치 및 방법이 개시된다. 연속 심층 영상 압축을 위한 장치는 딥러닝 네트워크 구조의 영상 압축 모델로서 이미지를 인코딩하기 위한 제1 인코더와, 인코딩된 이미지를 디코딩하기 위한 디코더, 및 디코딩된 이미지를 다시 인코딩하기 위한 제2 인코더를 포함하고, 상기 제1 인코더에서 추출된 제1 특징과 상기 제2 인코더에서 추출된 제2 특징에 기초한 손실 함수를 이용하여 상기 영상 압축 모델을 학습한다.

대표도 - 도15



$$L_{total} = L_R (\text{추론 확률}) + \lambda_1 \times L_D (\text{원본 영상, 복원 영상}) + \lambda_2 \times L_{FI} (F_0, F_1)$$

(52) CPC특허분류  
*G06T 9/002* (2013.01)

(72) 발명자

**이중석**

서울특별시 서대문구 연세로 50 (신촌동, 연세대학교)

---

**김준혁**

서울특별시 서대문구 연세로 50 (신촌동, 연세대학교)

## 명세서

### 청구범위

#### 청구항 1

컴퓨터 장치에 있어서,

메모리에 포함된 컴퓨터 관독가능한 명령들을 실행하도록 구성된 적어도 하나의 프로세서

를 포함하고,

상기 적어도 하나의 프로세서는,

딤러닝 네트워크 구조의 영상 압축 모델을 통해 인코딩 단계와 디코딩 단계를 포함한 압축 과정을 여러 번 반복하는 것으로,

상기 영상 압축 모델은 이미지를 인코딩하기 위한 제1 인코더와, 인코딩된 이미지를 디코딩하기 위한 디코더, 및 디코딩된 이미지를 다시 인코딩하기 위한 제2 인코더를 포함하고,

상기 제1 인코더에서 추출된 제1 특징과 상기 제2 인코더에서 추출된 제2 특징에 기초한 손실 함수를 이용하여 상기 영상 압축 모델을 학습하는 것

을 특징으로 하는 컴퓨터 장치.

#### 청구항 2

제1항에 있어서,

상기 손실 함수로 상기 제1 인코더에서 추출된 잠재적 특징(latent feature)과 상기 제2 인코더에서 추출된 잠재적 특징의 차이에 관한 손실 함수를 사용하는 것

을 특징으로 하는 컴퓨터 장치.

#### 청구항 3

제1항에 있어서,

상기 손실 함수는 수학식 1과 같이 정의되는 것

을 특징으로 하는 컴퓨터 장치.

[수학식 1]

$$\mathcal{L}_{FI}(\mathbf{F}_0, \mathbf{F}_1) = \|\mathbf{F}_0 - \mathbf{F}_1\|_2$$

(여기서,  $\mathbf{F}_0$ 은 제1 인코더에서 추출된 잠재적 특징을 나타내고,  $\mathbf{F}_1$ 은 제2 인코더에서 추출된 잠재적 특징을 나타낸다.)

#### 청구항 4

제1항에 있어서,

상기 영상 압축 모델을 구성하는 딤러닝 네트워크의 목적 함수는 수학식 2와 같이 정의되는 것

을 특징으로 하는 컴퓨터 장치.

[수학식 2]

$$\mathcal{L}_{total} = \mathcal{L}_R + \lambda_1 \mathcal{L}_D + \lambda_2 \mathcal{L}_{FI}$$

(여기서,  $\mathcal{L}_R$ 은 이미지의 압축 비율(rate)과 관련된 손실 조건을 나타내고,  $\mathcal{L}_D$ 는 이미지의 왜곡(distortion)과 관련된 손실 조건을 나타내며,  $\mathcal{L}_{FI}$ 은 이미지의 잠재적 특징과 관련된 손실 조건을 나타내고,  $\lambda_1$ 과  $\lambda_2$ 는 가중치를 나타낸다.)

**청구항 5**

제1항에 있어서,  
 상기 제1 인코더와 상기 제2 인코더는 컨볼루션 오토 인코더로 구성되는 것을 특징으로 하는 컴퓨터 장치.

**청구항 6**

컴퓨터 장치에서 실행되는 영상 압축 방법에 있어서,  
 상기 컴퓨터 장치는 메모리에 포함된 컴퓨터 관독가능한 명령들을 실행하도록 구성된 적어도 하나의 프로세서를 포함하고,  
 상기 적어도 하나의 프로세서는 딥러닝 네트워크 구조의 영상 압축 모델을 포함하고,  
 상기 영상 압축 모델은 이미지를 인코딩하기 위한 제1 인코더와, 인코딩된 이미지를 디코딩하기 위한 디코더, 및 디코딩된 이미지를 다시 인코딩하기 위한 제2 인코더를 포함하고,  
 상기 영상 압축 방법은,  
 상기 적어도 하나의 프로세서에 의해, 상기 제1 인코더에서 추출된 제1 특징과 상기 제2 인코더에서 추출된 제2 특징에 기초한 손실 함수를 이용하여 상기 영상 압축 모델을 학습하는 단계; 및  
 상기 적어도 하나의 프로세서에 의해, 상기 영상 압축 모델을 통해 인코딩 단계와 디코딩 단계를 포함한 압축 과정을 여러 번 반복 수행하는 단계를 포함하는 영상 압축 방법.

**청구항 7**

제6항에 있어서,  
 상기 손실 함수로 상기 제1 인코더에서 추출된 잠재적 특징과 상기 제2 인코더에서 추출된 잠재적 특징의 차이에 관한 손실 함수를 사용하는 것을 특징으로 하는 영상 압축 방법.

**청구항 8**

제6항에 있어서,  
 상기 손실 함수는 수학식 3과 같이 정의되는 것을 특징으로 하는 영상 압축 방법.

[수학식 3]

$$\mathcal{L}_{FI}(\mathbf{F}_0, \mathbf{F}_1) = \|\mathbf{F}_0 - \mathbf{F}_1\|_2$$

(여기서,  $\mathbf{F}_0$ 은 제1 인코더에서 추출된 잠재적 특징을 나타내고,  $\mathbf{F}_1$ 은 제2 인코더에서 추출된 잠재적 특징을 나타낸다.)

**청구항 9**

제6항에 있어서,

상기 영상 압축 모델을 구성하는 딥러닝 네트워크의 목적 함수는 수학식 4와 같이 정의되는 것을 특징으로 하는 영상 압축 방법.

[수학식 4]

$$\mathcal{L}_{total} = \mathcal{L}_R + \lambda_1 \mathcal{L}_D + \lambda_2 \mathcal{L}_{FI}$$

(여기서,  $\mathcal{L}_R$ 은 이미지의 압축 비율(rate)과 관련된 손실 조건을 나타내고,  $\mathcal{L}_D$ 는 이미지의 왜곡(distortion)과 관련된 손실 조건을 나타내며,  $\mathcal{L}_{FI}$ 은 이미지의 잠재적 특징과 관련된 손실 조건을 나타내고,  $\lambda_1$ 과  $\lambda_2$ 는 가중치를 나타낸다.)

**청구항 10**

제6항 내지 제9항 중 어느 한 항의 영상 압축 방법을 컴퓨터 장치에 실행시키기 위해 컴퓨터 판독가능한 기록 매체에 저장되는 컴퓨터 프로그램.

**청구항 11**

제6항 내지 제9항 중 어느 한 항의 영상 압축 방법을 컴퓨터에 실행시키기 위한 프로그램이 기록되어 있는 컴퓨터 판독 가능한 기록 매체.

**발명의 설명**

**기술 분야**

[0001] 아래의 설명은 딥러닝 기반의 영상 압축 기술에 관한 것이다.

**배경 기술**

[0002] 영상 압축은 멀티미디어 분야의 핵심 기술 중 하나로, 무손실 압축과 손실 압축으로 분류할 수 있다. 무손실 압축의 목적은 영상의 정보 손실 없이 최소한의 용량(속도)으로 영상을 인코딩하여 인코딩된 비트스트림에서 원본 영상을 완벽하게 재구성할 수 있도록 하는 것이다. 반면에, 손실 압축은 속도를 더욱 최소화하기 위해 영상의 일부 정보를 삭제하므로 디코딩된 영상이 불가피하게 왜곡될 수 있다. 손실 압축은 뛰어난 압축 효율성으로 인해 대부분의 실제 어플리케이션에서 사용되고 있다.

[0003] 연속 영상 압축(successive image compression, SIC라 칭함)은 다양한 어플리케이션에서 발생할 수 있는 영상의 인코딩과 디코딩을 반복하는 과정을 말한다. 예를 들어, 영상 처리 어플리케이션은 우선 카메라가 찍거나 다른 사용자가 공유한 압축 입력 영상을 해독한 다음, 영상을 조작하고(예를 들어, 자르기, 밝기 조정), 마지막으로 SIC가 압축적으로 관여하는 영상을 재인코딩한다.

[0004] 최근에는 멀티미디어와 통신기술이 발달함에 따라 영상 제작, 편집, 유통 등의 환경이 일반 이용자에게도 쉽게 보급되고 있어 영상 콘텐츠는 이전보다 SIC의 대상이 되는 경우가 많아지고 있다.

[0005] 이상적으로는 일정한 압축 조건에서 여러 번 연산해도 결과가 달라지지 않아야 한다. 즉, 각 반복해서 디코딩된 영상은 첫 번째 디코딩된 영상과 동일해야 한다. 그렇지 않으면 SIC 동안 정보 손실이 누적되게 되는데, 이를 세대손실이라고 한다. 따라서, SIC에 대한 강건성은 중요한 이슈이다.

**발명의 내용**

**해결하려는 과제**

[0006] 딥러닝 기반의 영상 압축 모델을 통해 여러 번의 반복적인 압축 과정(인코딩 및 디코딩)을 거치는 연속 심층 영상 압축 기술(successive deep image compression, SDIC라 칭함)을 제공한다.

[0007] 딥러닝 학습 모델의 본질로 인해 SDIC에서 왜곡 문제로 나타나는 불안정성(instability)을 해소할 수 있는 솔루션을 제공한다.

**과제의 해결 수단**

[0008] 컴퓨터 장치에 있어서, 메모리에 포함된 컴퓨터 관독가능한 명령들을 실행하도록 구성된 적어도 하나의 프로세서를 포함하고, 상기 적어도 하나의 프로세서는, 딥러닝 네트워크 구조의 영상 압축 모델을 통해 인코딩 단계와 디코딩 단계를 포함한 압축 과정을 여러 번 반복하는 것으로, 상기 영상 압축 모델은 이미지를 인코딩하기 위한 제1 인코더와, 인코딩된 이미지를 디코딩하기 위한 디코더, 및 디코딩된 이미지를 다시 인코딩하기 위한 제2 인코더를 포함하고, 상기 제1 인코더에서 추출된 제1 특징과 상기 제2 인코더에서 추출된 제2 특징에 기초한 손실 함수를 이용하여 상기 영상 압축 모델을 학습하는 것을 특징으로 하는 컴퓨터 장치를 제공한다.

[0009] 일 측면에 따르면, 상기 손실 함수로 상기 제1 인코더에서 추출된 잠재적 특징(latent feature)과 상기 제2 인코더에서 추출된 잠재적 특징의 차이에 관한 손실 함수를 사용할 수 있다.

[0010] 다른 측면에 따르면, 상기 손실 함수는 수학식 1과 같이 정의된다.

[0011] [수학식 1]

$$[0012] \mathcal{L}_{FI}(\mathbf{F}_0, \mathbf{F}_1) = \|\mathbf{F}_0 - \mathbf{F}_1\|_2$$

[0013] (여기서,  $\mathbf{F}_0$ 은 제1 인코더에서 추출된 잠재적 특징을 나타내고,  $\mathbf{F}_1$ 은 제2 인코더에서 추출된 잠재적 특징을 나타낸다.)

[0014] 또 다른 측면에 따르면, 상기 영상 압축 모델을 구성하는 딥러닝 네트워크의 목적 함수는 수학식 2와 같이 정의된다.

[0015] [수학식 2]

$$[0016] \mathcal{L}_{total} = \mathcal{L}_R + \lambda_1 \mathcal{L}_D + \lambda_2 \mathcal{L}_{FI}$$

[0017] (여기서,  $\mathcal{L}_R$ 은 이미지의 압축 비율(rate)과 관련된 손실 조건을 나타내고,  $\mathcal{L}_D$ 는 이미지의 왜곡(distortion)과 관련된 손실 조건을 나타내며,  $\mathcal{L}_{FI}$ 은 이미지의 잠재적 특징과 관련된 손실 조건을 나타내고,  $\lambda_1$ 과  $\lambda_2$ 는 가중치를 나타낸다.)

[0018] 또 다른 측면에 따르면, 상기 제1 인코더와 상기 제2 인코더는 컨볼루션 오토 인코더로 구성될 수 있다.

[0019] 컴퓨터 장치에서 실행되는 영상 압축 방법에 있어서, 상기 컴퓨터 장치는 메모리에 포함된 컴퓨터 관독가능한 명령들을 실행하도록 구성된 적어도 하나의 프로세서를 포함하고, 상기 적어도 하나의 프로세서는 딥러닝 네트워크 구조의 영상 압축 모델을 포함하고, 상기 영상 압축 모델은 이미지를 인코딩하기 위한 제1 인코더와, 인코딩된 이미지를 디코딩하기 위한 디코더, 및 디코딩된 이미지를 다시 인코딩하기 위한 제2 인코더를 포함하고, 상기 영상 압축 방법은, 상기 적어도 하나의 프로세서에 의해, 상기 제1 인코더에서 추출된 제1 특징과 상기 제2 인코더에서 추출된 제2 특징에 기초한 손실 함수를 이용하여 상기 영상 압축 모델을 학습하는 단계; 및 상기 적어도 하나의 프로세서에 의해, 상기 영상 압축 모델을 통해 인코딩 단계와 디코딩 단계를 포함한 압축 과정을 여러 번 반복 수행하는 단계를 포함하는 영상 압축 방법을 제공한다.

**발명의 효과**

[0020] 본 발명의 실시예들에 따르면, 딥러닝 기반의 영상 압축 모델을 통해 여러 번의 반복적인 압축 과정(인코딩 및 디코딩)을 거치는 SDIC에서 인코더를 구성하는 딥러닝 네트워크의 목적 함수로서 특징 정체성(feature identity) 손실이라고 불리는 심층 압축 모델을 학습에 같이 사용하여 SDIC의 불안정성을 해소할 수 있다.

**도면의 간단한 설명**

[0021] 도 1은 본 발명의 실시예에 따른 네트워크 환경의 예를 도시한 도면이다.

도 2는 본 발명의 실시예에 따른 컴퓨터 장치의 예를 도시한 블록도이다.

도 3은 영상 압축 모델의 학습 방법을 설명하기 위한 도면이다.

도 4는 본 발명의 일실시예에 있어서 연속 영상 압축 과정을 설명하기 위한 도면이다.

도 5 내지 도 14는 본 발명의 일실시예에 있어서 연속 심층 영상 압축의 불안정성을 설명하기 위한 예시 도면이다.

도 15는 본 발명의 일실시예에 있어서 연속 심층 영상 압축 모델의 학습 방법을 설명하기 위한 도면이다.

**발명을 실시하기 위한 구체적인 내용**

- [0022] 이하, 본 발명의 실시예를 첨부된 도면을 참조하여 상세하게 설명한다.
- [0024] 본 발명의 실시예들은 딥러닝 기반의 영상 압축 기술에 관한 것이다.
- [0025] 본 명세서에서 구체적으로 개시되는 것들을 포함하는 실시예들은 딥러닝 학습 모델의 본질로 인해 SDIC에서 왜곡 문제로 나타나는 불안정성을 개선할 수 있다.
- [0026] 본 발명의 실시예들에 따른 SDIC 시스템은 적어도 하나의 컴퓨터 장치에 의해 구현될 수 있으며, 본 발명의 실시예들에 따른 SDIC 방법은 SDIC 시스템에 포함되는 적어도 하나의 컴퓨터 장치를 통해 수행될 수 있다. 이때, 컴퓨터 장치에는 본 발명의 일실시예에 따른 컴퓨터 프로그램이 설치 및 구동될 수 있고, 컴퓨터 장치는 구동된 컴퓨터 프로그램의 제어에 따라 본 발명의 실시예들에 따른 SDIC 방법을 수행할 수 있다. 상술한 컴퓨터 프로그램은 컴퓨터 장치와 결합되어 SDIC 방법을 컴퓨터에 실행시키기 위해 컴퓨터 판독 가능한 기록매체에 저장될 수 있다.
- [0027] 도 1은 본 발명의 일실시예에 따른 네트워크 환경의 예를 도시한 도면이다. 도 1의 네트워크 환경은 복수의 전자 기기들(110, 120, 130, 140), 복수의 서버들(150, 160) 및 네트워크(170)를 포함하는 예를 나타내고 있다. 이러한 도 1은 발명의 설명을 위한 일례로 전자 기기의 수나 서버의 수가 도 1과 같이 한정되는 것은 아니다. 또한, 도 1의 네트워크 환경은 본 실시예들에 적용 가능한 환경들 중 하나의 예를 설명하는 것일 뿐, 본 실시예들에 적용 가능한 환경이 도 1의 네트워크 환경으로 한정되는 것은 아니다.
- [0028] 복수의 전자 기기들(110, 120, 130, 140)은 컴퓨터 장치로 구현되는 고정형 단말이거나 이동형 단말일 수 있다. 복수의 전자 기기들(110, 120, 130, 140)의 예를 들면, 스마트폰(smart phone), 휴대폰, 내비게이션, 컴퓨터, 노트북, 디지털방송용 단말, PDA(Personal Digital Assistants), PMP(Portable Multimedia Player), 태블릿 PC 등이 있다. 일례로 도 1에서는 전자 기기(110)의 예로 스마트폰의 형상을 나타내고 있으나, 본 발명의 실시예들에서 전자 기기(110)는 실질적으로 무선 또는 유선 통신 방식을 이용하여 네트워크(170)를 통해 다른 전자 기기들(120, 130, 140) 및/또는 서버(150, 160)와 통신할 수 있는 다양한 물리적인 컴퓨터 장치들 중 하나를 의미할 수 있다.
- [0029] 통신 방식은 제한되지 않으며, 네트워크(170)가 포함할 수 있는 통신망(일례로, 이동통신망, 유선 인터넷, 무선 인터넷, 방송망)을 활용하는 통신 방식뿐만 아니라 기기들간의 근거리 무선 통신 역시 포함될 수 있다. 예를 들어, 네트워크(170)는, PAN(personal area network), LAN(local area network), CAN(campus area network), MAN(metropolitan area network), WAN(wide area network), BBN(broadband network), 인터넷 등의 네트워크 중 하나 이상의 임의의 네트워크를 포함할 수 있다. 또한, 네트워크(170)는 버스 네트워크, 스타 네트워크, 링 네트워크, 메쉬 네트워크, 스타-버스 네트워크, 트리 또는 계층적(hierarchical) 네트워크 등을 포함하는 네트워크 토폴로지 중 임의의 하나 이상을 포함할 수 있으나, 이에 제한되지 않는다.
- [0030] 서버(150, 160) 각각은 복수의 전자 기기들(110, 120, 130, 140)과 네트워크(170)를 통해 통신하여 명령, 코드, 파일, 콘텐츠, 서비스 등을 제공하는 컴퓨터 장치 또는 복수의 컴퓨터 장치들로 구현될 수 있다. 예를 들어, 서버(150)는 네트워크(170)를 통해 접속한 복수의 전자 기기들(110, 120, 130, 140)로 서비스(일례로, 콘텐츠 제공 서비스, 영상/음성 통화 서비스, 메시징 서비스, 메일 서비스, 소셜 네트워크 서비스, 지도 서비스, 번역 서비스, 금융 서비스, 결제 서비스, 검색 서비스 등)를 제공하는 시스템일 수 있다.
- [0031] 도 2는 본 발명의 일실시예에 따른 컴퓨터 장치의 예를 도시한 블록도이다. 앞서 설명한 복수의 전자 기기들(110, 120, 130, 140) 각각이나 서버들(150, 160) 각각은 도 2를 통해 도시된 컴퓨터 장치(200)에 의해 구현될 수 있다.
- [0032] 이러한 컴퓨터 장치(200)는 도 2에 도시된 바와 같이, 메모리(210), 프로세서(220), 통신 인터페이스(230) 그리

고 입출력 인터페이스(240)를 포함할 수 있다. 메모리(210)는 컴퓨터에서 판독 가능한 기록매체로서, RAM(random access memory), ROM(read only memory) 및 디스크 드라이브와 같은 비소멸성 대용량 기록장치(permanent mass storage device)를 포함할 수 있다. 여기서 ROM과 디스크 드라이브와 같은 비소멸성 대용량 기록장치는 메모리(210)와는 구분되는 별도의 영구 저장 장치로서 컴퓨터 장치(200)에 포함될 수도 있다. 또한, 메모리(210)에는 운영체제와 적어도 하나의 프로그램 코드가 저장될 수 있다. 이러한 소프트웨어 구성요소들은 메모리(210)와는 별도의 컴퓨터에서 판독 가능한 기록매체로부터 메모리(210)로 로딩될 수 있다. 이러한 별도의 컴퓨터에서 판독 가능한 기록매체는 플로피 드라이브, 디스크, 테이프, DVD/CD-ROM 드라이브, 메모리 카드 등의 컴퓨터에서 판독 가능한 기록매체를 포함할 수 있다. 다른 실시예에서 소프트웨어 구성요소들은 컴퓨터에서 판독 가능한 기록매체가 아닌 통신 인터페이스(230)를 통해 메모리(210)에 로딩될 수도 있다. 예를 들어, 소프트웨어 구성요소들은 네트워크(170)를 통해 수신되는 파일들에 의해 설치되는 컴퓨터 프로그램에 기반하여 컴퓨터 장치(200)의 메모리(210)에 로딩될 수 있다.

[0033] 프로세서(220)는 기본적인 산술, 로직 및 입출력 연산을 수행함으로써, 컴퓨터 프로그램의 명령을 처리하도록 구성될 수 있다. 명령은 메모리(210) 또는 통신 인터페이스(230)에 의해 프로세서(220)로 제공될 수 있다. 예를 들어 프로세서(220)는 메모리(210)와 같은 기록 장치에 저장된 프로그램 코드에 따라 수신되는 명령을 실행하도록 구성될 수 있다.

[0034] 통신 인터페이스(230)는 네트워크(170)를 통해 컴퓨터 장치(200)가 다른 장치(일례로, 앞서 설명한 저장 장치들)와 서로 통신하기 위한 기능을 제공할 수 있다. 일례로, 컴퓨터 장치(200)의 프로세서(220)가 메모리(210)와 같은 기록 장치에 저장된 프로그램 코드에 따라 생성한 요청이나 명령, 데이터, 파일 등이 통신 인터페이스(230)의 제어에 따라 네트워크(170)를 통해 다른 장치들로 전달될 수 있다. 역으로, 다른 장치로부터의 신호나 명령, 데이터, 파일 등이 네트워크(170)를 거쳐 컴퓨터 장치(200)의 통신 인터페이스(230)를 통해 컴퓨터 장치(200)로 수신될 수 있다. 통신 인터페이스(230)를 통해 수신된 신호나 명령, 데이터 등은 프로세서(220)나 메모리(210)로 전달될 수 있고, 파일 등은 컴퓨터 장치(200)가 더 포함할 수 있는 저장 매체(상술한 영구 저장 장치)로 저장될 수 있다.

[0035] 입출력 인터페이스(240)는 입출력 장치(250)와의 인터페이스를 위한 수단일 수 있다. 예를 들어, 입력 장치는 마이크, 키보드 또는 마우스 등의 장치를, 그리고 출력 장치는 디스플레이, 스피커와 같은 장치를 포함할 수 있다. 다른 예로 입출력 인터페이스(240)는 터치스크린과 같이 입력과 출력을 위한 기능이 하나로 통합된 장치와의 인터페이스를 위한 수단일 수도 있다. 입출력 장치(250)는 컴퓨터 장치(200)와 하나의 장치로 구성될 수도 있다.

[0036] 또한, 다른 실시예들에서 컴퓨터 장치(200)는 도 2의 구성요소들보다 더 적은 혹은 더 많은 구성요소들을 포함할 수도 있다. 그러나, 대부분의 종래기술적 구성요소들을 명확하게 도시할 필요성은 없다. 예를 들어, 컴퓨터 장치(200)는 상술한 입출력 장치(250) 중 적어도 일부를 포함하도록 구현되거나 또는 트랜시버(transceiver), 데이터베이스 등과 같은 다른 구성요소들을 더 포함할 수도 있다.

[0037] 이하에서는 연속 심층 영상 압축을 위한 방법 및 시스템의 구체적인 실시예를 설명하기로 한다.

[0038] 본 명세서에서는 SDIC에 대한 종합적인 분석을 실시한다. SDIC의 동작이 JPEG 압축을 사용한 SIC의 동작과 현저하게 다르다는 문제를 소개하면서, SDIC의 불안정성이 실제 상황에서 악화될 수 있음을 보여준다. 마지막으로, SIC로 인한 왜곡을 완화하기 위해 심층 영상 압축 모델을 학습하기 위한 새로운 손실함수를 제안한다.

[0039] 먼저, 관련 연구로서 손실 영상 압축 기술을 설명한다.

[0040] 변환 코딩은 JPEG 압축을 포함한 손실 영상 압축에 일반적으로 사용되는 전통적인 데이터 압축 방식의 일종이다. 변환, 양자화, 엔트로피 코딩 등 3단계로 구성된다. 첫째, 주어진 데이터를 압축에 효과적인 연속 값 표현으로 선형 변환한다. 변환된 표현은 그 후 정량화되며, 이후 손실 없는 엔트로피 코딩이 적용되어 압축 효율을 높인다. 예를 들어, JPEG의 인코더 측에서는 8×8개의 분할된 영상 패치가 이산 코사인 변환(DCT)을 거치고, 그 결과 주파수 성분을 정량화하여 지각 중복성을 감소시킨다. 정량화된 패치는 허프만 코딩을 사용하여 손실 없이 압축된다.

[0041] 최근에는 심층 신경망의 비선형 특성을 활용하여 기존의 선형 변환 기반 방법을 능가하는 딥러닝 기반 손실 영상 압축 방법이 제안되고 있다.

[0042] 딥러닝 기반 손실 영상 압축은 크게 세 부분, 즉 영상으로부터 압축에 의미있는 특징들을 추출하는 인코더, 추출된 특징들의 확률을 모델링하여 엔트로피를 예측하는 엔트로피 모델, 추출된 특징들을 복원하는 디코더로 나

된다. 복원 영상의 왜곡(distortion)과 압축 비율(rate) 모두를 최적화하기 위해 컨볼루션 오토 인코더를 사용한다.

[0043] 복원 영상의 왜곡을 최소화하기 위한 손실 조건(loss term)( $\mathcal{L}_D$ )은 원본 영상과 복원 영상의 차이를 기초로 정의할 수 있고, 압축 비율과 관련된 손실 조건( $\mathcal{L}_R$ )은 잠재된 표현들의 엔트로피를 기초로 정의할 수 있다.

[0044] 도 3을 참조하면, 딥러닝 모델의 학습 과정에서는 복원 영상의 왜곡 정도와 관련된 손실 조건( $\mathcal{L}_D$ )과 압축 비율과 관련된 손실 조건( $\mathcal{L}_R$ )의 조합인  $\mathcal{L}_{total} = \mathcal{L}_R + \lambda\mathcal{L}_D$  를 손실 함수(loss function)로 하여 학습이 진행된다.  $\mathcal{L}_R$ 은 추론 확률을 나타내는 엔트로피를 계산하는 함수,  $\mathcal{L}_D$ 는 원본 영상과 복원 영상의 차이에 관한 함수(예를 들어, 평균 제곱 오차(MSE))를 의미한다. 여기서,  $\lambda$ 는 두 함수 사이의 중요도를 결정하는 가중치로, 큰 값을 가질수록 압축이 덜 되고 복원 영상의 화질은 좋다.

[0045] 다음으로, SDIC에 대해 구체적으로 설명한다.

[0046] 도 4를 참조하면, SDIC 시스템(400)은 딥러닝 기반 영상 압축 모델(40)을 통해 여러 번의 압축 과정(인코딩 및 디코딩)을 반복하는 것이다.

[0047] 원본 영상( $I_0$ )에 따라 손실 압축의 인코딩과 디코딩 단계는 다음과 같이 수행된다.

[0048] [수학식 1]

$$F_0 = f_q \circ f_e(I_0)$$

[0049]

[0050] [수학식 2]

$$I_1 = f_{rc} \circ f_d(F_0) = f_{ed}(I_0)$$

[0051]

[0052] 여기서,  $F_0$ 는 최초의 잠재적 특징,  $f_e(\cdot)$ 와  $f_q(\cdot)$ 는 각각 인코딩 단계에서의 변환 함수와 양자화 연산,  $I_1$ 은 디코딩된 영상,  $f_d(\cdot)$ 와  $f_{rc}(\cdot)$ 는 각각 디코딩 단계에서의 변환 함수와 라운딩(rounding) 및 클리핑(clipping) 연산을 의미하고,  $f_{ed}(\cdot)$ 는 인코딩과 디코딩 단계를 포함하는 전체 과정을 나타낸 것이다. 는 함수 구성요소이다.

[0053]  $n(\geq 2)$ 회 반복되는 SIC를 수학식 3과 같이 공식화할 수 있다.

[0054] [수학식 3]

$$I_n = f_{SIC,n}(I_0) = \overbrace{f_{ed} \circ f_{ed} \cdots \circ f_{ed}}^{n \text{ times}}(I_0)$$

[0055]

[0056]  $I_n$ 은  $n$ 회 반복 후 디코딩된 영상을 의미한다. 여기서, 각 압축이 정확히 동일한 조건에서 수행된다고 가정한다. 또한, 전체 SIC 프로세스 동안 영상 조작(예를 들어, 자르기 등)은 관여하지 않는 것으로 가정한다.

[0057] SIC 동안 관측된 현상을 조사하기 위해 다양한 딥러닝 기반 방법을 사용하여 데이터셋의 샘플 영상 두 개에  $f_{SIC,50}(\cdot)$ 을 적용한다. 또한, 디코딩된 영상의 품질 제어에 사용되는 품질 계수가 10인 JPEG 압축 기술을 사용한다(일반적으로, 1~100 범위). JPEG의 경우와 달리, 딥러닝을 이용하여 SIC의 상당한 불안정성을 관찰할 수 있으며, 그 결과는 도 5와 도 6에 나타나 있다. 도 5와 도 6은 딥러닝 기반 SIC의 불안정성을 설명하기 위한 도면으로, 도 5는 영상 압축을 50회 반복한 시각적 결과를 나타낸 것이고, 도 6은 영상 압축의 반복 중 PSNR(dB)의 변화를 나타낸 것이다. 딥러닝 기반의 모델들이 연속 압축 상황에서 JPEG에 비해 큰 불안정성을 보임을 확인할 수 있다.

[0058] 도 5를 참조하면, 딥러닝 기반 압축 방식은 JPEG를 이용한 SIC에 비해 상대적으로 많은 왜곡을 발생시켜 SDIC의 불안정성을 보여준다. 게다가, 왜곡은 매우 다양한 형태로 일어나며, 첫째 약한 수준의 흐림부터 영상의 일부 내용을 이해할 수 없는 심각한 수준까지 품질 저하 정도가 다양하다. 둘째, 왜곡은 영상 전체에 걸쳐 또는 국소적으로 나타나고 있다. 마지막으로, 왜곡의 종류는 흐림, 부정확한 색, 왜곡된 윤곽, 두드러진 색 패턴 등 다양하다.

[0059] 딥러닝 기반 압축 방식의 경우 같은 모델이더라도 영상에 따라 완전히 다른 시각적 결과를 보인다는 점에서 SDIC의 불안정성은 영상 내용에 따라 좌우된다는 것을 알 수 있다.

[0060] 도 6의 각 곡선 그래프는 각 심층 압축 모델의 반복과 관련하여 디코딩된 영상의 PSNR 변화를 보여준다. PSNR 값이 허용 가능한 품질 저하로 수렴되는 추세, 상당한 품질 저하(10dB 이상)로 분산되는 추세, 수렴되다가 결국 분산되어 많은 양의 강하를 보이는 추세 등이 관찰된다. 이러한 관찰은 영상 압축 반복 횟수와 관련하여 SDIC의 불안정성을 보여준다.

[0061] SDIC의 불안정성 원인을 조사하기 위해  $f_{ed}(\cdot)$ , 즉  $f_{ed} \circ f_{ed}(\cdot) = f_{ed}(\cdot)$ 의 멱등법칙(idempotence)을 달성하기 위한 조건을 분석한다.

[0062] [수학식 4]

[0063] 
$$f_{ed} \circ f_{ed}(\cdot) = f_{rc} \circ f_d \circ f_q \circ \underline{f_e \circ f_{rc} \circ f_d} \circ f_q \circ f_e(\cdot)$$

[0064] 수학식 2와 수학식 4를 고려하면, 수학식 4에서 밑줄 친 부분이 정체성 함수(identity function)일 경우  $f_q(\cdot)$ 는 멱등성(idempotent), 즉  $f_q(\cdot) \circ f_q(\cdot) = f_q(\cdot)$ 이기 때문에 멱등법칙이 달성된다. JPEG의 경우  $f_e(\cdot)$ 와  $f_d(\cdot)$ 는 역관계(DCT와 역DCT)를 가지므로  $f_{rc}(\cdot)$ 에 의한 오류를 무시할 수 있다면 멱등법칙을 보장한다.

[0065] 이런 점에서 SDIC의 불안정성은  $f_e \circ f_d(\cdot)$ 가 딥러닝 기반 방법의 정체성 함수가 아니라는 데서 비롯된다. 무엇보다 기존 모델은  $F_i$ 와  $F_{i+1} = f_e \circ f_d(F_i)$  간의 차이를 줄이기 위해 훈련되지 않았으며, 이러한 차이를 훈련 중에 고려하지 않더라도  $F_{i+1}$ 이  $f_d(\cdot)$ 가 훈련되는 잠재표현의 영역 내에 머무른다면  $F_{i+2} = f_d(F_{i+1})$ 은 어느 정도 품질을 보장하고 그렇지 않다면 상당한 품질 저하가 발생할 수 있다.

[0066] 도 7을 참조하면, 영상 압축을 반복하는 과정에서 이전 복원 영상인 복원 영상\_1과 다음 복원 영상인 복원 영상\_2가 같으려면 괄호 안 영역(70)이 정체성 함수가 되어야 한다. JPEG은 디코더와 인코더의 변환 함수가 각각 역DCT와 DCT로 명시적으로 정의되는 반면, 딥러닝 기반의 모델들은 학습을 기반으로 하는 본질로 인해 각각의 함수가 학습에 의해 결정되는 특성이 있고, 이러한 이유로 안정성을 보장하지 못한다. 이는 불안정성의 원인이 된다.

[0067] 디코더가 출력한 영상이 인코더가 다룰 수 있는 도메인에서 벗어나는 상황에서 불안정성이 심화될 것이라는 예측 하에 두 가지 모델을 훈련시킴으로써 불안정성을 증명할 수 있다. 하나는 모든 MNIST 훈련 데이터를 사용하고 다른 하나는 테스트 영상만을 사용하여 의도적으로 모델이 과적합(overfitting) 되도록 하고  $f_d(\cdot)$ 가 훈련되는 극히 작은 도메인을 부과한다. 도 8은 테스트 영상에서 SIC의 시각적 결과를 보여주는 것으로, 과적합된 모델에서 심각한 품질 저하를 보여준다. 과적합 모델의 경우 압축으로 인해 원본 영상과 비교하여 일정한 반복에서 디코딩된 영상의 손실로 인해  $f_e(\cdot)$ 에 의한 영상의 잠재표현이 훈련 중에 보이는 것과 크게 달라지는데, 이때  $f_d(\cdot)$ 에 의해 왜곡된 영상이 생성된다.

[0068] 딥러닝 기반의 모델들의 연속 압축 상황에서의 벤치마크를 진행하여 도 9에 도시한 바와 같이 압축율에 따라서 4가지 구역(Very Low, Low, Mid, High)으로 나눠서 불안정성에 영향을 미치는 요인에 대한 심층적인 분석을 제공한다.

[0069] 도 10은 압축율이 'Low'인 그룹에서 심층 압축 모델에 의해 영상 압축을 50회 반복한 SIC의 시각적 결과를 비교

한 것이고, 도 11은 압축율이 'High'인 그룹에서 세 가지 다른 엔트로피 추정 모델을 사용한 SIC의 시각적 결과를 비교한 것이다.

[0070] 상기한 분석을 통해 압축 방법 간의 가장 큰 차이는 각 압축 방법이 사용하는 엔트로피 추정 모델에 있다는 점을 확인할 수 있다. 첫째, 엔트로피 추정치가 없는 방법은 엔트로피 추정을 사용하는 방법과 비교하여 SIC로 인한 왜곡이 심하다. 엔트로피 추정에 기초한 손실 함수를 부과하면 인코더 부분이 훈련 데이터의 확률 분포를 학습하게 된다. 이것은 압축을 위해 어떤 정보를 제거해야 하는지, 즉 덜 자주 발생하는 정보가 손실되어 디코딩된 영상이 훈련용 영상의 배포에 포함되도록 돕는 효과에 대하여 유용한 가이드를 제공한다. 둘째, 컨텍스트 모델을 가진 방법은 SIC에 대해 상대적으로 높은 강건성을 보여준다. 컨텍스트 모델은 전체 압축 모델 내에서 잠재된 표현의 분포를 학습하는 효과가 있는 자기 회귀 방식으로 확률을 추정한다. 셋째, 서로 다른 엔트로피 추정 모델은 특히 압축 효율을 위해 폐기할 정보를 신중하게 결정하는 압축율이 'High'인 그룹에서 서로 다른 성능을 보여준다.

[0071] 또한, 학습에 사용되는 왜곡과 관련된 손실 조건의 종류에 따라서 불안정성의 패턴이 달라지는 것을 확인할 수 있다. 도 12는 MSE 및 MS-SSIM 최적화 모델 간의 영상 압축 50회 후 시각적 결과를 비교한 것이다. 전체적으로 MSE로 최적화된 모델은 흐릿한 영상을 연출하는 경향이 있는 반면, MS-SSIM 최적화 모델에서는 색상 정보 손실이 자주 발생한다. 압축율이 'High'인 그룹의 경우 MSE 최적화 모델의 경우 모자 중간에 왜곡이 나타나는 반면, MS-SSIM 최적화 모델의 경우 영상 테두리 근처의 왜곡이 관찰된다. 영상 화질 측정을 위한 구조적 왜곡에 중점을 둔 MS-SSIM에 최적화된 모델은 SIC 동안 영상 구조(예를 들어, 가장자리) 주변의 왜곡이 덜 발생하는 경향이 있다. 반면, MS-SSIM은 색상 정보를 고려하지 않은 채 휘도 성분을 측정하기 때문에, MS-SSIM에 최적화된 모델에서 생성된 영상은 흐릿함 등 구조적 왜곡보다 색상 왜곡이 더 많이 나타난다. 또한, MS-SSIM은 슬라이딩 가우스 윈도우를 영상에 적용한 후 계산되어 영상의 테두리 부근에 있는 픽셀의 상대적 중요도가 감소하기 때문에 영상 테두리 주변의 왜곡이 자주 관찰된다.

[0072] 추가적으로, 실제 상황(practical situations)에서 불안정성이 더 심화된다는 것을 확인할 수 있다. 도 13은 영상을 한 픽셀 단위로 옮겨가며 자른 후 각각을 반복적 압축을 진행한 시각적 결과를 나타낸 것이다. 절단면적이 변경됨에 따라 SIC에 의한 왜곡 패턴이 눈에 띄게 달라짐을 확인할 수 있다. SIC의 불안정성이 한 픽셀 이동 수준에서도 영상 내용에 민감하다는 것을 나타내며, 이는 미세한 변화가  $f_e \circ f_d(\cdot)$ 를 통해 현저하게 다른 잠재적 표현으로 이어질 수 있기 때문이다.

[0073] 또한, 실제 상황으로 변화하는 압축 환경에서 불안정성이 더 심화되는 것을 확인할 수 있다. 변화하는 압축 환경이란, 매번 압축 과정에서 타겟 압축율이 다른 경우를 의미한다. 예를 들어, JPEG의 경우는 품질 계수(quality factor)가 90, 70, 90 순으로 매번 바뀌는 상황을 의미한다. 도 14는 다양한 압축 조건에 대한 SIC 결과를 비교한 것으로, 일정 조건(constant) 대 가변 조건(variable)을 비교한 것이다. 가변 압축 조건("Very Low"+"Low", "Low"+"Mid", "Mid"+"High")에서는 일정한 압축 조건("Very Low", "Low", "Mid")에 비해 SIC에 의한 왜곡 정도가 증가하는 것으로 관측된다. 흐림이 심하여 영상 디테일을 구분하기 어렵거나, 색상 정보가 변경 또는 두드러지는 패턴 등이 나타난다. 이러한 결과는 실제 상황에서 SDIC의 불안정성이 심화될 가능성을 시사한다.

[0074] 본 실시예에서는 SIC에 대해 심층 영상 압축의 강건성을 개선하여 불안정성을 완화시키기 위한 효과적인 방법을 제안한다.

[0075] 도 15를 참조하면, 본 발명에 따른 SDIC 시스템(400)은 딥러닝 기반 영상 압축 모델로서 이미지의 압축을 위한 딥러닝 네트워크 구조의 인코더(151), 및 이미지의 압축 해제를 위한 딥러닝 네트워크 구조의 디코더(153)를 포함할 수 있다.

[0076] 본 실시예에서는 딥러닝 기반 영상 압축 모델에 대해 새로운 손실 함수로서 FI를 정의할 수 있으며, 이를 위하여 디코더(153)를 통해 복원된 이미지의 압축을 위한 딥러닝 네트워크 구조의 인코더(152)가 추가될 수 있다.

[0077] 인코더(151, 152)는 이미지의 왜곡과 압축 비율 모두를 최적화하기 위해 컨볼루션 오토 인코더를 사용할 수 있다.

[0078] 인코더(151)의 파라미터 재사용을 통해서 복원 영상을 추가 인코더(152)를 통해 한번 더 인코딩하여  $F_1$ 을 추출함에 따라  $F_1$ 과  $F_0$  사이에 새로운 손실 FI를 정의한다. 다양한 형태의 손실이 적용 가능하며, 일례로 평균 제곱

오차(MSE), 다시 말해  $F_1$ 과  $F_0$ 의 차이에 관한 손실 함수를 사용할 수 있다.

[0079]

상세하게,  $f_e \circ f_d(\cdot)$ 와 함께 정체성 함수를 구현하기 위해 심층 압축 모델을 명시적으로 시행할 수 있다.  $F_1 = f_e \circ f_d(F_0)$ 을 고려하면, 특징 정체성(feature identity)(FI) 손실  $\mathcal{L}_{FI}(\cdot)$ 라 불리는 모델을 훈련시키기 위한 새로운 손실 함수를 제안한다.

[0080]

[수학식 5]

[0081]

$$\mathcal{L}_{FI}(F_0, F_1) = \|F_0 - F_1\|_2$$

[0082]

여기서,  $F_0$ 은 수학식 1과 같고,  $F_1$ 은  $F_1 = f_q \circ f_e(I_1)$ 와 같다. 제안된 손실 함수  $\mathcal{L}_{FI}(\cdot)$ 는 원본 비율 손실 함수( $\mathcal{L}_R$ )와 왜곡 손실 함수( $\mathcal{L}_D$ )와 함께 쉽게 사용될 수 있다.

[0083]

[수학식 6]

[0084]

$$\mathcal{L}_{total} = \mathcal{L}_R + \lambda_1 \mathcal{L}_D + \lambda_2 \mathcal{L}_{FI}$$

[0085]

여기서,  $\lambda_1$ 과  $\lambda_2$ 는 두 함수 사이의 중요도를 결정하는 가중치를 의미한다.

[0086]

일례로,  $\lambda_2$ 를 1로 설정할 수 있으며, FI 손실 함수로 훈련된 모델은 모든 구성에 대해 전체 SIC 과정 동안 향상된 강건성을 보이며, 영상 전체의 성능 변동을 최소화할 수 있다.

[0087]

이처럼 본 발명의 실시예들에 따르면, SDIC의 불안정성을 줄이기 위한 솔루션을 통해 SDIC의 품질 저하를 최소화할 수 있고 훈련을 통해 심층 영상 압축의 인코딩 및 디코딩 체인의 먹등법칙을 강제하여 SDIC의 강건성을 개선할 수 있다.

[0088]

이상에서 설명된 장치는 하드웨어 구성요소, 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성요소는, 프로세서, 콘트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPGA(field programmable gate array), PLU(programmable logic unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 어플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(processing element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 콘트롤러를 포함할 수 있다. 또한, 병렬 프로세서(parallel processor)와 같은, 다른 처리 구성(processing configuration)도 가능하다.

[0089]

소프트웨어는 컴퓨터 프로그램(computer program), 코드(code), 명령(instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성요소(component), 물리적 장치, 컴퓨터 저장 매체 또는 장치에 구체화(embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.

[0090]

실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 이때, 매체는 컴퓨터로 실행 가능한 프로그램을 계속 저장하거나, 실행 또는 다운로드를 위해 임시 저장하는 것일 수도 있다. 또한, 매체는 단일 또는 수 개의 하드웨어가 결합된 형태의 다양한 기록수단 또는 저장수단일 수 있는데, 어떤 컴퓨터 시스템에 직접 접속되는 매체에 한정되지 않고, 네트워크 상에 분산 존재하는 것일 수도 있다. 매체의 예시로는, 하드 디스크, 플로피 디스크 및 자기 테이프와 같

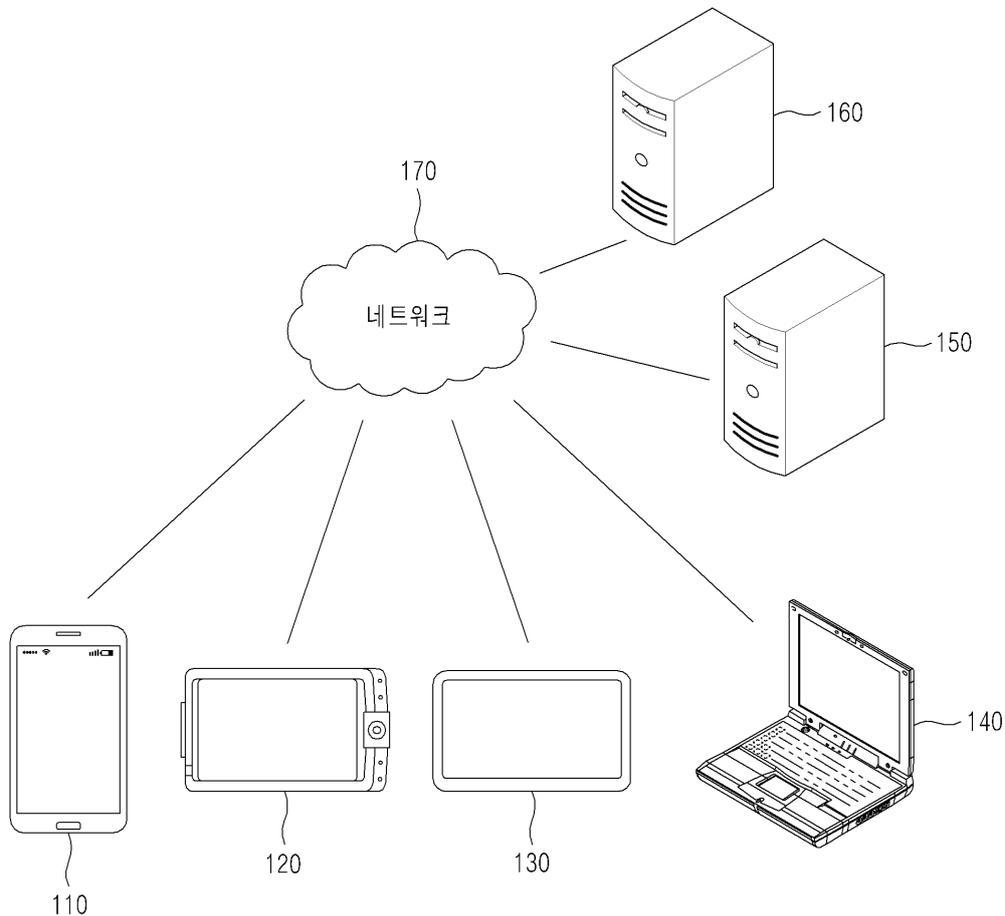
은 자기 매체, CD-ROM 및 DVD와 같은 광기록 매체, 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체 (magneto-optical medium), 및 ROM, RAM, 플래시 메모리 등을 포함하여 프로그램 명령어가 저장되도록 구성된 것이 있을 수 있다. 또한, 다른 매체의 예시로, 어플리케이션을 유통하는 앱 스토어나 기타 다양한 소프트웨어를 공급 내지 유통하는 사이트, 서버 등에서 관리하는 기록매체 내지 저장매체도 들 수 있다.

[0091] 이상과 같이 실시예들이 비록 한정된 실시예와 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기의 기재로부터 다양한 수정 및 변형이 가능하다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

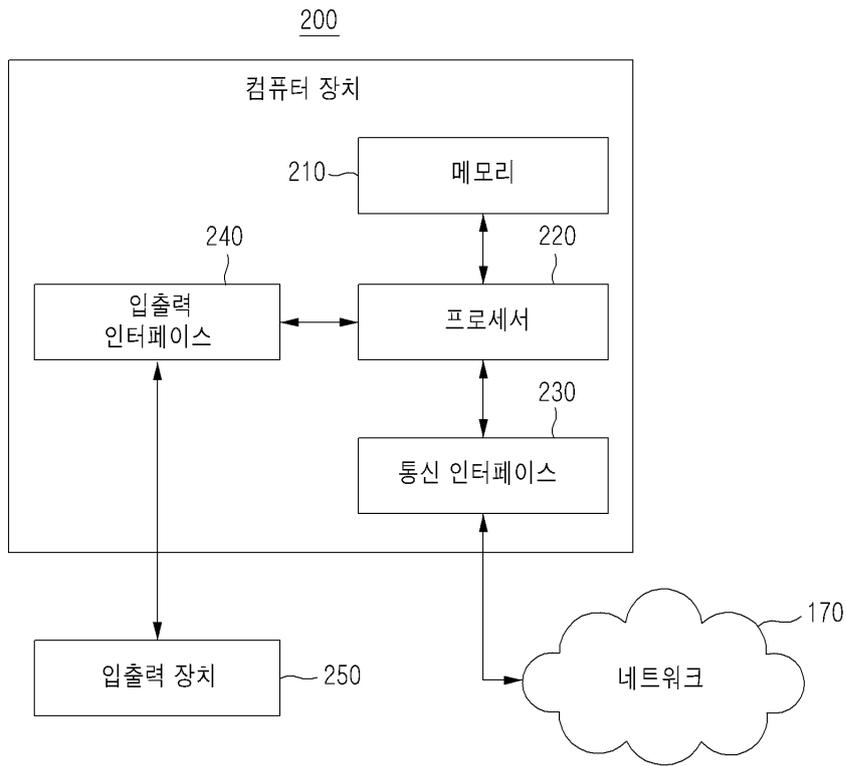
[0092] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

**도면**

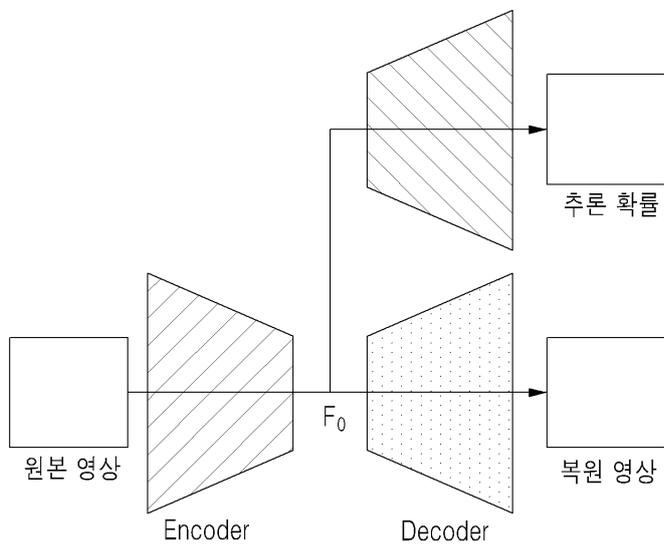
**도면1**



도면2

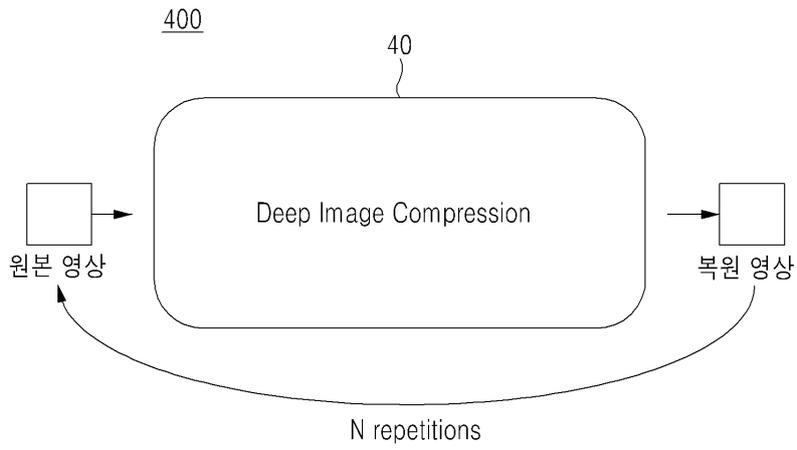


도면3

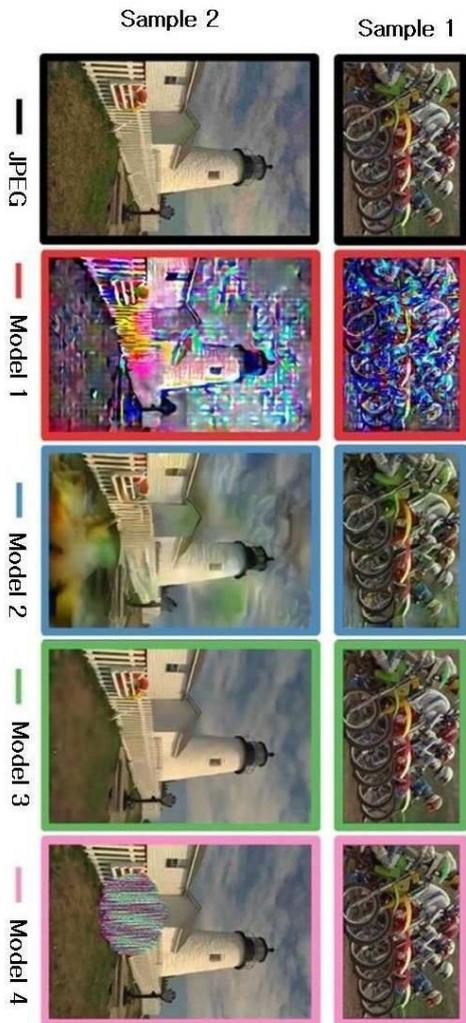


$$L_{total} = L_R (\text{추론 확률}) + \lambda \times L_D (\text{원본 영상, 복원 영상})$$

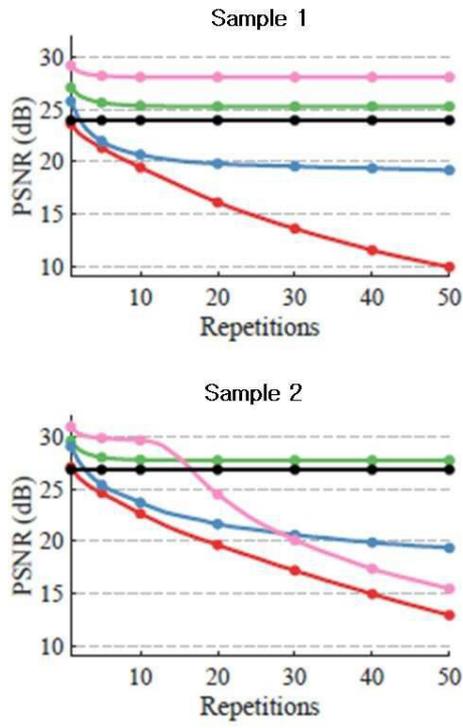
도면4



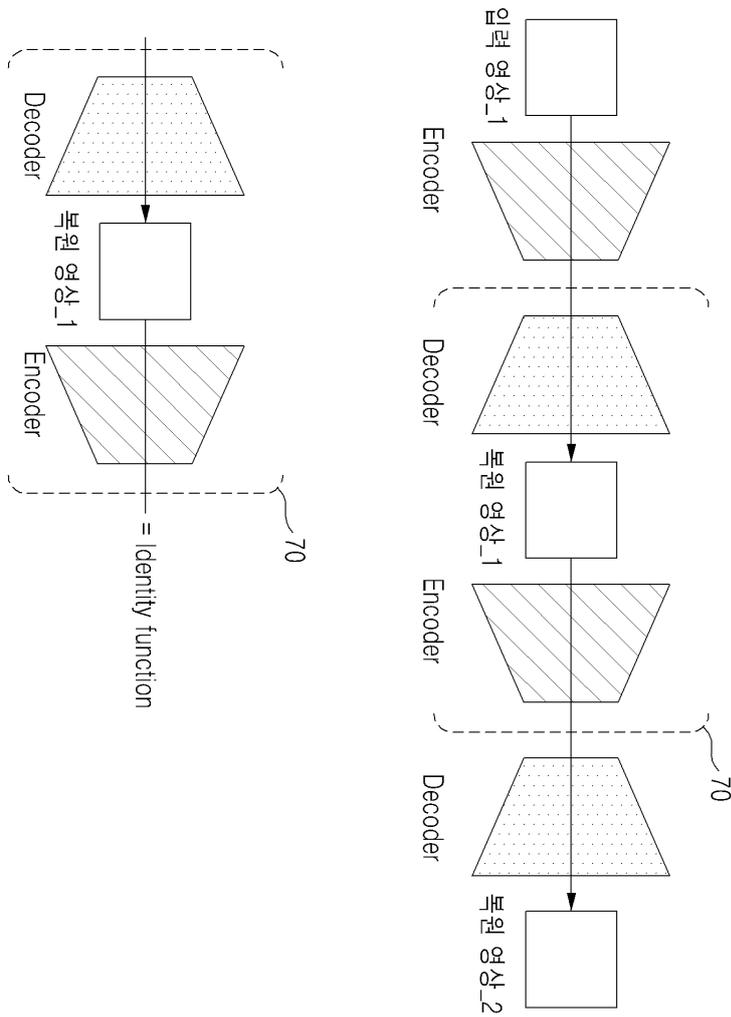
도면5



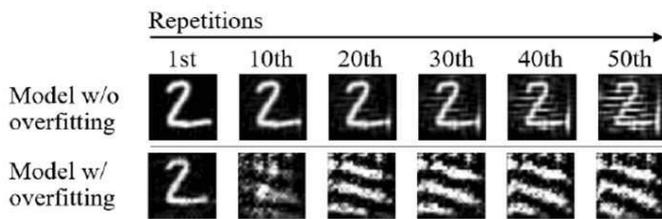
도면6



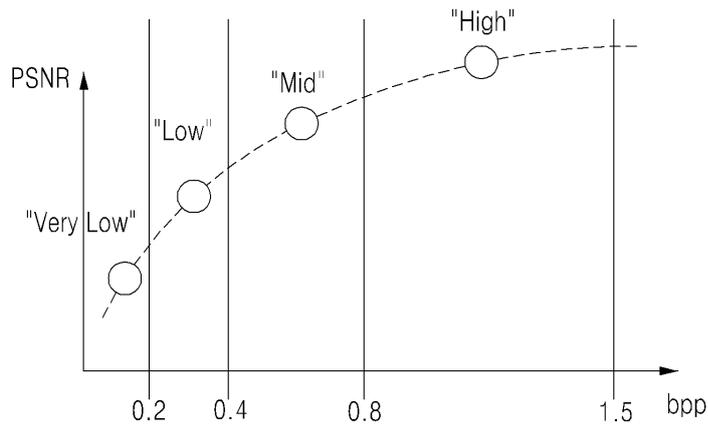
도면7



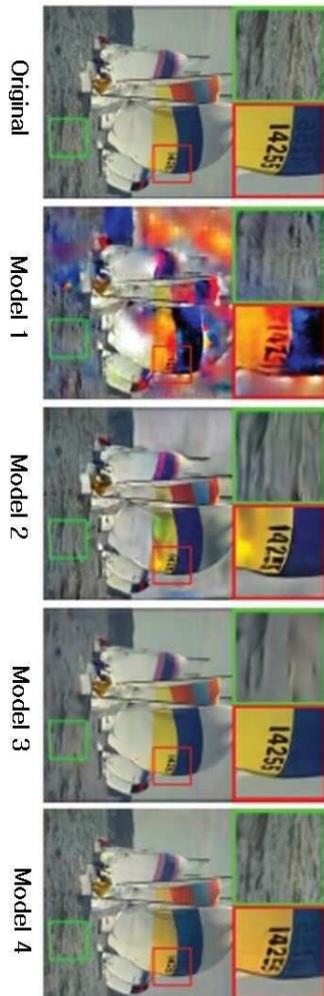
도면8



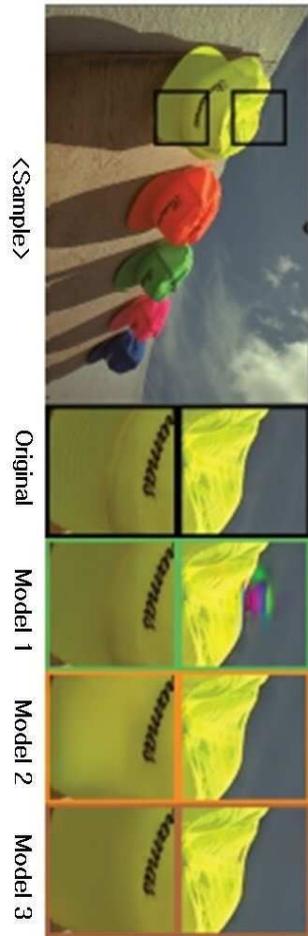
도면9



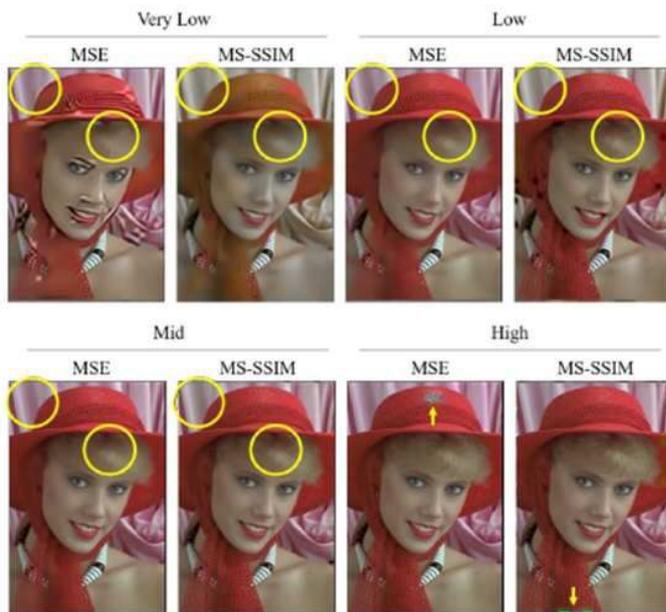
도면10



도면11



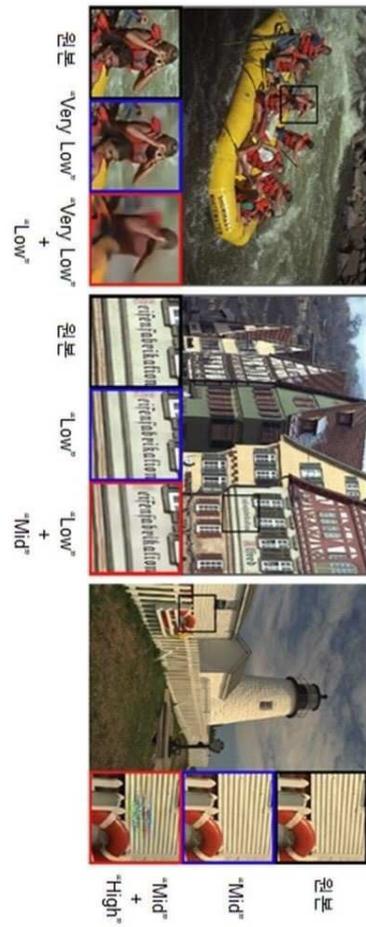
도면12



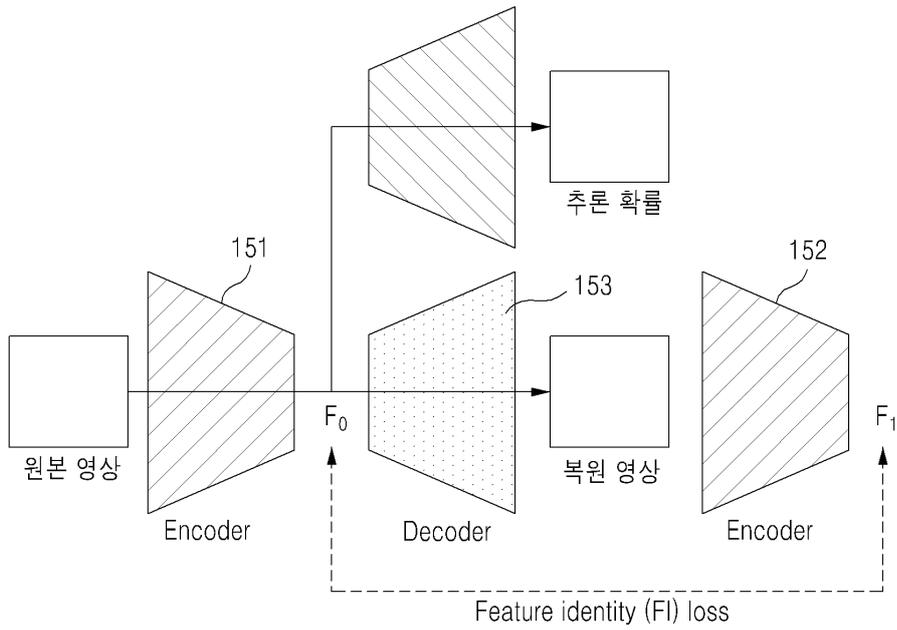
도면13



도면14



도면15



$$L_{\text{total}} = L_R (\text{추론 확률}) + \lambda_1 \times L_D (\text{원본 영상, 복원 영상}) + \lambda_2 \times L_{FI} (F_0, F_1)$$