(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2020년10월15일
(11) 등록번호 10-2166016
(24) 등록일자 2020년10월08일

(51) 국제특허분류(Int. Cl.)
G06T 3/00 (2019.01) *G06T 7/11* (2017.01)
(52) CPC특허분류
G06T 3/00 (2019.01)
G06N 20/00 (2019.01)
(21) 출원번호 10-2019-0112094
(22) 출원일자 2019년09월10일
심사청구일자 2019년09월10일
(56) 선행기술조사문헌
Azadi et al. Compositional GAN: Learning Image-Conditional Binary Composition. arXiv, 2019년 3월, pp. 1-15. 1부.*
Mo et al. INSTAGAN: INSTANCE-AWARE IMAGE-TO-IMAGE TRANSLATION. arXiv, 2019년 1월, pp. 1-26. 1부.*
Shen et al. Towards Instance-level Image-to-Image Translation. CVPR, 2019년 6월, pp. 3683-3692. 1부.*
JP2019101789 A
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
인하대학교 산학협력단
인천광역시 미추홀구 인하로 100(용현동, 인하대학교)
(72) 발명자
신병석
서울특별시 서초구
이연
인천광역시 부평구 주부토로 78, 902호(부평동, 뉴에이스)
윤욱
인천광역시 미추홀구 인하로 100(용현동, 인하대학교)
(74) 대리인
양성보

전체 청구항 수 : 총 5 항 심사관 : 김창원

(54) 발명의 명칭 **이미지 대 이미지의 변환을 위한 도메인 인식 기반의 변환 기법**

(57) 요 약

이미지 대 이미지의 변환을 위한 도메인 인식 기반의 변환 기법이 개시된다. 일 실시예에 따른 도메인 변환 방법은, 입력 이미지와 목표 이미지 각각을 백그라운드 영역 및 인스턴스 영역으로 각각 분리하는 단계; 상기 분리된 입력 이미지와 목표 이미지 각각의 백그라운드 영역 및 인스턴스 영역을 이미지 변환을 위하여 생성된 변환 모델에 입력하여 학습시키는 단계; 및 상기 학습을 통하여 획득된 변환 이미지를 상기 분리된 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 기초하여 생성된 라벨(label)을 사용하여 통합하는 단계를 포함할 수 있다.

대 표 도 - 도1



(52) CPC특허분류
G06T 7/11 (2017.01)
G06T 2207/20081 (2013.01)
G06T 2207/20084 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711077423(세부과제번호: 2017M3C8A8091768)
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	기초 . 원천기술 개발사업
연구과제명	[Ezbaro] 디지털 컴패니언 기반의 고령자 모니터링 및 대응 소프트웨어 기술 개발
기 여 율	1/1
과제수행기관명	인하대학교
연구기간	2018.09.20 ~ 2019.09.19

명 세 서

청구범위

청구항 1

도메인 변환 방법에 있어서,

입력 이미지와 목표 이미지 각각을 백그라운드 영역 및 인스턴스 영역으로 각각 분리하는 단계;

상기 분리된 입력 이미지와 목표 이미지 각각의 백그라운드 영역 및 인스턴스 영역을 이미지 변환을 위하여 생성된 변환 모델에 입력하여 학습시키는 단계; 및

상기 학습을 통하여 획득된 변환 이미지를 상기 분리된 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 기초하여 생성된 라벨(label)을 사용하여 통합하는 단계

를 포함하고,

상기 학습시키는 단계는,

상기 이미지 변환을 위하여 백그라운드 모델 및 인스턴스 모델을 포함하는 변환 모델을 생성하고, 상기 입력 이미지와 목표 이미지 각각의 백그라운드 영역을 상기 변환 모델에 구성된 공통의 잠재적 공간으로 맵핑시키고, 상기 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 각각의 인스턴스 영역에 의해 공유되는 스타일 코드 및 컨텐츠 코드를 매핑하여 복수 개의 인스턴스 영역을 생성하는 단계

를 포함하는 도메인 변환 방법.

청구항 2

제1항에 있어서,

상기 각각 분리하는 단계는,

상기 입력 이미지와 상기 목표 이미지 각각으로부터 분리된 인스턴스 영역을 획득하고, 상기 인스턴스 영역을 상기 획득된 인스턴스 영역 이외의 나머지 영역을 픽셀의 평균으로 패딩(padding)하여 백그라운드 영역을 획득하는 단계

를 포함하는 도메인 변환 방법.

청구항 3

제1항에 있어서,

상기 통합하는 단계는,

상기 입력 이미지와 목표 이미지 각각을 분리함에 따라 각각의 인스턴스 영역의 위치 값을 단위 행렬로 저장하여 라벨로 저장하고, 상기 저장된 라벨을 사용하여 상기 변환 이미지를 통합하여 이미지를 변환시키는 단계

를 포함하는 도메인 변환 방법.

청구항 4

삭제

청구항 5

제1항에 있어서,

상기 백그라운드 모델은, 상기 입력 이미지와 목표 이미지 각각의 백그라운드 영역을 백그라운드 네트워크에 입력하여 크로스 도메인 변환(cross-domain transformation)을 위한 학습을 수행하는 것을 포함하고,

상기 인스턴스 모델은, 복원(reconstruction) 모드 및 다중 출력(multi-output) 모드로 구성되는 것을

포함하고,

상기 복원 모드는, 상기 입력 이미지와 목표 이미지 각각의 백그라운드 영역이 변화될 때, 인스턴스 네트워크를 통하여 상기 입력 이미지와 목표 이미지 각각의 인스턴스 영역을 변화시키지 않으면서 통합 이미지가 기 설정된 기준 이상 일치하도록 학습을 수행하고,

상기 다중 출력 모드는, 상기 입력 이미지와 목표 이미지 각각의 인스턴스 영역을 인스턴스 네트워크에 학습시킴에 따라 각각의 인스턴스 영역에 임의의 코드를 추가하여 복수 개의 스타일을 변화시켜 다중 이미지의 출력을 획득하기 위하여 학습시키는

것을 특징으로 하는 도메인 변환 방법.

청구항 6

도메인 변환 시스템에 있어서,

입력 이미지와 목표 이미지 각각을 백그라운드 영역 및 인스턴스 영역으로 각각 분리하는 분리부;

상기 분리된 입력 이미지와 목표 이미지 각각의 백그라운드 영역 및 인스턴스 영역을 이미지 변환을 위하여 생성된 변환 모델에 입력하여 학습시키는 학습부; 및

상기 학습을 통하여 획득된 변환 이미지를 상기 분리된 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 기초하여 생성된 라벨(label)을 사용하여 통합하는 통합부

를 포함하고,

상기 학습부는,

상기 이미지 변환을 위하여 백그라운드 모델 및 인스턴스 모델을 포함하는 변환 모델을 생성하고, 상기 입력 이미지와 목표 이미지 각각의 백그라운드 영역을 상기 변환 모델에 구성된 공통의 잠재적 공간으로 맵핑시키고, 상기 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 각각의 인스턴스 영역에 의해 공유되는 스타일 코드 및 컨텐츠 코드를 매핑하여 복수 개의 인스턴스 영역을 생성하는

도메인 변환 시스템.

발명의 설명

기 술 분 야

[0001] 아래의 설명은 이미지 변환 기술에 관한 것이다.

배 경 기 술

[0003] Generative adversarial networks (GANs)는 인공신경망(neural network)의 일종으로, 가짜(fake) 이미지를 생성하는 제너레이터(Generator)와 제너레이터(Generator)가 생성한 이미지와 실제 이미지를 구별하는 Discriminator 두 개의 네트워크를 동시에 훈련시킨다. GANs는 도메인 변환 (domain transfer)에서 좋은 효과를 내고 있으며, text-to-image synthesis, super-resolution, image-to-image translation 등 분야에서 state-of-art의 성능을 보이고 있다.

[0004] 의료 영상에 GANs를 활용한 연구는 다양하게 진행되고 있다. 대부분의 연구는 synthesis, segmentation, reconstruction 에 집중되어 있다. 또한 CT와 MR 데이터를 타겟으로 하는 연구가 대부분이며, 적은 비율이나 Retinal Fundus Imaging, histopathology, X-ray를 대상으로 한 연구도 진행되고 있다.

[0005] Image to Image(I2I)는 이미지 잡음제거(denoising), 이미지 합성(synthetics), 해상도 향상(resolution enhancement)과 같은 다양한 컴퓨터 비전(computer vision) 분야에 활용되고 있다.

발명의 내용

해결하려는 과제

[0007] 단위 행렬(identity matrix)을 이용하여 원본 이미지의 인스턴스(instance)를 구역화(segregate) 하는 도메인 인식(domain-aware) 적대적 생성 신경망(Generative adversarial network, GAN) 프레임워크 (domain-aware GAN, DAGAN)를 제안한다.

과제의 해결 수단

[0009] 도메인 변환 방법은, 입력 이미지와 목표 이미지 각각을 백그라운드 영역 및 인스턴스 영역으로 각각 분리하는 단계; 상기 분리된 입력 이미지와 목표 이미지 각각의 백그라운드 영역 및 인스턴스 영역을 이미지 변환을 위하여 생성된 변환 모델에 입력하여 학습시키는 단계; 및 상기 학습을 통하여 획득된 변환 이미지를 상기 분리된 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 기초하여 생성된 라벨(label)을 사용하여 통합하는 단계를 포함할 수 있다.

[0010] 상기 각각 분리하는 단계는, 상기 입력 이미지와 상기 목표 이미지 각각으로부터 분리된 인스턴스 영역을 획득하고, 상기 인스턴스 영역을 상기 획득된 인스턴스 영역 이외의 나머지 영역을 픽셀의 평균으로 패딩(padding)하여 백그라운드 영역을 획득하는 단계를 포함할 수 있다.

[0011] 상기 통합하는 단계는, 상기 입력 이미지와 목표 이미지 각각을 분리함에 따라 각각의 인스턴스 영역의 위치 값을 단위 행렬로 저장하여 라벨로 저장하고, 상기 저장된 라벨을 사용하여 상기 변환 이미지를 통합하여 이미지를 변환시키는 단계를 포함할 수 있다.

[0012] 상기 학습시키는 단계는, 상기 이미지 변환을 위하여 백그라운드 모델 및 인스턴스 모델을 포함하는 변환 모델을 생성하고, 상기 입력 이미지와 목표 이미지 각각의 백그라운드 영역을 상기 변환 모델에 구성된 공통의 잠재적 공간으로 맵핑시키고, 상기 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 각각의 인스턴스 영역에 의해 공유되는 스타일 코드 및 컨텐츠 코드를 매핑하여 복수 개의 인스턴스 영역을 생성하는 단계를 포함할 수 있다.

[0013] 상기 백그라운드 모델은, 상기 입력 이미지와 목표 이미지 각각의 백그라운드 영역을 백그라운드 네트워크에 입력하여 크로스 도메인 변환(cross-domain transformation)을 위한 학습을 수행하는 것을 포함하고, 상기 인스턴스 모델은, 복원(reconstruction) 모드 및 다중 출력(multi-output) 모드로 구성되는 것을 포함하고, 상기 복원 모드는, 상기 입력 이미지와 목표 이미지 각각의 백그라운드 영역이 변화될 때, 인스턴스 네트워크를 통하여 상기 입력 이미지와 목표 이미지 각각의 인스턴스 영역을 변화시키지 않으면서 통합 이미지가 기 설정된 기준 이상 일치하도록 학습을 수행하고, 상기 다중 출력 모드는, 상기 입력 이미지와 목표 이미지 각각의 인스턴스 영역을 인스턴스 네트워크에 학습시킴에 따라 각각의 인스턴스 영역에 임의의 코드를 추가하여 복수 개의 스타일을 변화시켜 다중 이미지의 출력을 획득하기 위하여 학습시킬 수 있다.

[0014] 도메인 변환 시스템은, 입력 이미지와 목표 이미지 각각을 백그라운드 영역 및 인스턴스 영역으로 각각 분리하는 분리부; 상기 분리된 입력 이미지와 목표 이미지 각각의 백그라운드 영역 및 인스턴스 영역을 이미지 변환을 위하여 생성된 변환 모델에 입력하여 학습시키는 학습부; 및 상기 학습을 통하여 획득된 변환 이미지를 상기 분리된 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 기초하여 생성된 라벨(label)을 사용하여 통합하는 통합부를 포함할 수 있다.

발명의 효과

[0016] 일 실시예에 따른 도메인 변환 시스템은 단위행렬을 이용하여 원본 이미지로부터 백그라운드 영역과 인스턴스 영역을 분리하고, 전체적으로 UNIT을 기본 모델로 하여 공통의 잠재적 공간을 구하고, 이를 통하여 모델의 안정성을 유지할 수 있다. 또한, 학습을 통하여 보다 유연하게 비대응 데이터 셋의 도메인 변환을 수행할 수 있다.

도면의 간단한 설명

[0018] 도 1은 일 실시예에 따른 비대응 이미지의 도메인 인식 변환을 나타낸 예이다.

도 2는 일 실시예에 따른 도메인 변환 시스템의 백그라운드와 인스턴스 모델의 두 파트로 구성된 것을 나타낸 예이다.

도 3은 일 실시예에 따른 도메인 변환 시스템에서 백그라운드 모델을 설명하기 위한 도면이다.

도 4 및 도 5는 일 실시예에 따른 도메인 변환 시스템에서 인스턴스 모델을 설명하기 위한 도면이다.

도 6은 일 실시예에 따른 도메인 변환 시스템에서 이미지를 변환하는 것을 설명하기 위한 도면이다.

도 7은 일 실시예에 따른 도메인 변환 시스템에서 smoothed 판별기(discriminator)를 설명하기 위한 도면이다.

도 8은 일 실시예에 따른 도메인 변환 시스템의 구성을 설명하기 위한 블록도이다.

도 9는 일 실시예에 따른 도메인 변환 시스템에서 이미지 대 이미지 변환을 수행하는 방법을 설명하기 위한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0019] 이하, 실시예를 첨부한 도면을 참조하여 상세히 설명한다.

[0021] 도 1은 비대응 이미지의 도메인 인식 변환을 나타낸 예이다.

[0022] 백그라운드(background)와 인스턴스(instance) 도메인의 변환을 각각 수행하고, 변환된 각각의 백그라운드 도메인 및 인스턴스 도메인을 스무스 라벨(smoothed label)을 통해 통합(intergration)할 수 있다. 이하, 이미지를 통합하는 방법에 대하여 구체적으로 설명하기로 한다.

[0023] 도 2는 일 실시예에 따른 도메인 변환 시스템의 백그라운드와 인스턴스 모델의 두 파트로 구성된 것을 나타낸 예이다.

[0024] 도메인 변환 시스템에서 제안하는 모델은 입력 이미지 A와 목표 이미지 B 사이의 인스턴스 인식 변환(instance-aware translation)을 수행할 수 있다. 이때, 각각의 이미지는 다음과 같이, 백그라운드 영역과 인스턴스 영역으로 이루어졌다고 가정하기로 한다.

[0025] $A: \{A_{bgr}, A_{ins}\},\ B: \{B_{bgr},\ B_{ins}\}$

[0026] 도메인 변환 시스템에서 제안하는 변환 모델은 백그라운드 모델과 인스턴스 모델로 구성될 수 있다. 도메인 변환 시스템은 전체 네트워크를 훈련하기 전에, 입력 이미지 A와 목표 이미지 B를 포함하는 각각의 원본 이미지로부터 인스턴스 영역 A_{ins} 와 B_{ins}를 잘라낼(cutting) 수 있다. 도메인 변환 시스템은 각각의 원본 이미지로부터 인스턴스 영역을 잘라낸 위치 값을 단위 행렬로 저장하여 라벨(label)로 사용할 수 있다. 잘려진 인스턴스 영역들은 나머지 이미지의 평균 값으로 대체될 수 있다. 변환 모델에 구성된 백그라운드 네트워크와 인스턴스 네트워크를 훈련한 후, 변환된 이미지를 복원하는 통합(intergration)과정이 수행될 수 있다.

[0027] 도 3은 일 실시예에 따른 도메인 변환 시스템에서 백그라운드 모델을 설명하기 위한 도면이다.

[0028] 백그라운드 모델은 다른 부분(different part)을 위해 다른 변환(different translation)을 적용하기 위한 것이다. 도 3을 참고하면, 백그라운드 모델은 두 백그라운드 입력 사이의 크로스 도메인 변환(cross-domain transformation) 학습을 목표로 한다.

[0029] 인스턴스 도메인과 백그라운드 도메인을 예시로 사용하여 변환을 수행할 수 있다. 입력 이미지를 분리함에 따라 인스턴스 영역이 획득될 수 있다. 백그라운드 영역은 픽셀 값의 평균으로 패딩(padding)하여 획득될 수 있다. 백그라운드 모델은 전체 프레임워크에서 독립적이며, 백그라운드 모델은 인스턴스 도메인과 백그라운드 도메인을 포함하는 두 도메인 사이의 시각적 변환을 학습할 수 있다. 이러한 추론을 통하여 인코더와 생성기(generator)는 각각 다음과 같이 정의될 수 있다.

[0030] $$VAE^{A}_{bgr}: \{E^{A}_{bgr}, G^{A}_{bgr}\} \quad E^{A}_{bgr}, G^{A}_{bgr} \in C$$

$$VAE^{B}_{bgr}: \{E^{B}_{bgr}, G^{A}_{bgr}\} \quad E^{A}_{bgr}, G^{A}_{bgr} \in C \#(1)$$

[0031] UNIT과 유사하게, 인코더 E^{A}_{bgr} E^{B}_{bgr} 를 통해 주어진 목표 백그라운드(target background)

$C_{bgr}^{A} = E_{bgr}^{A}(A_{bgr})$ and $C_{bgr}^{B} = E_{bgr}^{B}(B_{bgr})$ 일 때, 기본적인 잠재적 공간을 맵핑할 수 있다. C_{bgr}^{A} 와 C_{bgr}^{B} 는 각각 잠재 코드(latent code)를 나타낸다. 실시예에서는 마지막 두 레이어 E_{bgr}^{A}, E_{bgr}^{B} 와 첫번째 레이어 G_{bgr}^{A}, G_{bgr}^{B} 의 가중치 공유(weight sharing)를 수행할 수 있다. 동시에 두 개의 판별기(discriminator) D_{bgr}^{A}, D_{bgr}^{B} 를 추가할 수 있다. 판별기들은 두 백그라운드 도메인 사이의 변환이 적절한 것인지를 판별하고 $G_{bgr}^{A}(C_{bgr}^{B})$ 와 A_{bgr} 의 적재적 프로세스를 수행할 수 있다. 이때, 적대적 프로세스란 $G_{bgr}^{A}(C_{bgr}^{B})$ 와 A_{bgr} 가 D_{bgr}^{A} 에 의해 판별되는 것을 의미한다. 동시에 $G_{bgr}^{B}(C_{bgr}^{A})$ 와 B_{bgr} 는 D_{bgr}^{B} 에 의해 판별될 수 있다.

[0032] 도 4 및 도 5는 일 실시예에 따른 도메인 변환 시스템에서 인스턴스 모델을 설명하기 위한 도면이다.

[0033] 인스턴스 네트워크는 복원(reconstruction)와 다중 출력(multi-output)모드로 구성될 수 있다. 인스턴스 모델의 목표는 인스턴스 영역의 독립성(independent)을 유지하는 것이다. 다시 말해서, 백그라운드의 변환에 영향을 받지 않도록 하는 것이다. 첫 번째, DCGAN (Deep Convolution Generative Adversarial Networks)과 유사한 구조로 단순한 인스턴스 복원을 수행하여 가우시안 분포 N(0,1)을 따르는 잡음 벡터(noise vector)를 끌어낸다. 이것은 전체 프레임워크(백그라운드 네트워크 포함)가 백그라운드 영역을 다른 스타일로 변환하되, 인스턴스 영역은 변화없이 남기도록 한다. 이미지는 스타일과 컨텐츠 코드들(content codes)로 구성되며, 인스턴스 부분을 다중 출력 모델로 변환할 수 있다.

[0034] 복원 모드(reconstruction mode)는 인스턴스 영역을 변화시키지 않으면서 마지막 통합 이미지가 최대한 현실적이도록 생성할 수 있다. 인스턴스 영역의 사이즈를 고려하여 생성기와 판별기 내부의 convolutional 레이어 복수 개(예를 들면, 3개)를 두고, 정규 분포에서 샘플링한 랜덤 노이즈 Z ~ N (0, 1)를 입력 벡터로 사용할 수 있다. 도 4를 참고하면, 복원 모드를 나타낸 것으로, 세 개의 블럭(block)으로 구성된 판별기는 원본 이미지의 입력과 원본 이미지의 입력으로부터 생성된 결과 G(z)를 구분할 수 있다. 생성기의 각 블럭은 "Transposed convolution + Batch normalization + Relu activation"으로 구성될 수 있다. 판별기의 블럭이 "Convolutional + Batch normalization + leakly Relu activation"으로 구성된 것과 구분된다.

[0035] 도 5를 참고하면, 다중 출력 모드(multiple-output mode)는 원본 이미지로부터 잘려진 인스턴스 영역은 완전히 독립적이다는 조건 하에서, 인스턴스 영역에 대하여 원하는 모든 변환 작업을 수행할 수 있다. 일반적인 접근법에서, 다양한 출력을 획득하기 위하여 이미지를 스타일과 컨텐츠 정보의 조합으로 본다. 특히, 만약 적절한 속성이나 스타일 코드를 추가한다면, 이미지를 임의의 스타일로 변환할 수 있다. MUNIT(multimodal Unsupervised Image-to-image Translation) 구조와 유사하게, 목표(target) 인스턴스 영역에서 가져온 스타일 코드를 무작위로 샘플링하고, 이를 컨텐츠 코드와 재결합하여 복수의 인스턴스를 생성할 수 있다.

[0036] 입력 인스턴스 영역A_{ins} 와 목표 인스턴스 영역 B_{ins}가 주어지면, 인코더 $E_{A_ins}^{S}$, $E_{A_ins}^{C}$, $E_{B_ins}^{S}$, $E_{B_ins}^{C}$에서 입력 인스턴스 영역A_{ins} 와 목표 인스턴스 영역 B_{ins}를 스타일과 컨텐츠 공간(content space)으로 각각 맵핑할 수 있다(S_{ins}^{A}, C_{ins}^{A}, S_{ins}^{B}, C_{ins}^{B} 는 입력 인스턴스 영역A_{ins} 와 목표 인스턴스 영역 B_{ins}의 스타일과 컨텐츠 공간을 나타낸다.).

$$\{S_{ins}^{A}, C_{ins}^{A}\} = \{E_{A_{ins}}^{S}(A_{ins}), E_{A_{ins}}^{C}(A_{ins})\} \quad C_{ins}^{A} \in C$$

$$\{S_{ins}^{B}, C_{ins}^{B}\} = \{E_{B_{ins}}^{S}(B_{ins}), E_{B_{ins}}^{C}(B_{ins})\} \quad C_{ins}^{B} \in C \#(2)$$

[0037]

[0038] 백그라운드 모델의 공통 공간(common space) 셋팅처럼, 컨텐츠 공간 역시 입력 인스턴스 영역 및 목표 인스턴스 영역, 즉, 두 인스턴스 영역에 의해 공유될 수 있다. 그 후, 목표 인스턴스 영역 B_{ins}의 스타일 S_{ins}^{B} 및 입력 인스턴스 영역A_{ins}의 스타일 S_{ins}^{A} 에서 무작위로 샘플링 된 두 스타일 코드 $\sigma(S_{ins}^{B}),\ \sigma(S_{ins}^{A})$ 와 입력 인스턴스 영역A_{ins} 와 목표 인스턴스 영역 B_{ins}의 컨텐츠 코드 C_{ins}^{A} , C_{ins}^{B} 를 통합할 수 있다. 학습 과정에서 획득된 순환적 의미(Cycle sementics)를 통해 양방향 인스턴스 변환을 수행할 수 있다. 도 5는 Pascal VOC 데이터 셋에 다중 출력 모델을 적용한 예이다.

[0039] 실시예에서 제안된 모델의 손실 함수는 백그라운드 모델과 인스턴스 모델에 따라 다르게 사용될 수 있다. 백그라운드와 인스턴스 변환 문제를 서로 연관 지어 해결하기 위한 전체 목적 함수(full object function)는 다음과 같다.

$$Loss = \min_{G} \max_{D} [\min_{G_{ins}, E_{ins}} \max_{D_{ins}} L_{ins} + \min_{G_{bgr}, E_{bgr}} \max_{D_{bgr}} L_{bgr}] \#(3)$$

[0040]

[0041] 인스턴스 모델은 유연한 변환(flexible transformation)을 위해 복원 모드(reconstruction mode)와 다중 출력 모드(multi-output mode)로 구성될 수 있다. 복원 모드는 백그라운드 영역이 변환될 때 인스턴스 영역이 사라지지 않도록 하고, 다중 출력 모드는 다양한 인스턴스 영역의 변환을 생성하기 위해 사용될 수 있다. 예를 들면, 입력 이미지에서 인스턴스 영역을 자른 후, 상기 인스턴스 영역에 임의의 코드를 추가하여 스타일을 변환할 수 있다.

[0042] 도 4에 도시된 바와 같이, 복원 모드의 인스턴스 네트워크는 잡음 벡터 $Z_A,\ Z_B$로 구성되며, 두 개의 생성기와 판별기 $G_{ins}^{A},\ G_{ins}^{B},\ D_{ins}^{A},\ D_{ins}^{B}$ 로 구성될 수 있다. 이때, 각 생성기와 판별기는 3개의 레이어로 구성될 수 있다. 적대적 손실(adversarial loss)은 아래와 같이 나타난다.

$$L_{ins}^{A} = E_{A_{ins}\sim p(A_{ins}),\, Z_B\sim p(Z_B)}[\frac{1}{2} log\, D_{ins}^{A}\left(G_{ins}^{A}(Z_A)\right) + \frac{1}{2} log(1 - D_{ins}^{A}(A_{ins}))] \qquad (4)$$

[0043]

[0044] 다중 출력 모드가 도시된 도 5 에서 볼 수 있듯, 입력 인스턴스 영역 A_{ins}은 인코더-생성기(encoder-generator)의 집합 $\{E_{A_ins}^{S}, E_{A_ins}^{C}, G_{ins}^{A}\}$ 으로 구성되며, 다중 출력 모드에서의 L_{ins}은 복원 손실(reconstruction loss)과 적대적 손실(adversarial loss)로 구성될 수 있다.

[0045] 손실 함수 및 학습(loss function and training)에 대하여 설명하기로 한다.

[0046] 복원 손실이란 일반적으로 인스턴스 영역은 잠재 코드(latent code)에 대한 이미지로 변환되어 처리될 수 있다. 그러나, 스타일 코드는 정규 분포 N(0,1)에서 샘플링되지 않는다. 실시예에 따른 모델에서는 스타일 인코더와 컨텐츠 인코더를 사용하여 변환된 인스턴스 영역이 목표 도메인의 속성을 유지하게 한다. 이러한 손실 함수의 목표는 학습 후 이미지가 잠재 코드와 의미론적 일관성을 복원할 수 있도록 하는 것이다.

[0047] 적대적 손실에서 GAN을 사용하는 주요한 목적은 적대적 학습 과정을 통해 목표 도메인에 최대한 일치하는 결과

를 획득할 수 있기 때문이다. 판별기 D_{ins}^{A} 와 D_{ins}^{B} 를 사용하여 변환된 인스턴스 영역과 실제 인스턴스를 구분할 수 있다.

$$L_{adv}^{A_{ins}} = E_{A_{ins}\sim p(A_{ins}),C_{ins}^{A}\sim p(C_{ins}^{A})}[\frac{1}{2}log\, D_{ins}^{A}\left(G_{ins}^{A}(\sigma(S_{ins}^{A}),C_{ins}^{A})\right) + \frac{1}{2}log\,(1-D_{ins}^{A}(A_{ins}))] \quad (6)$$

The L_{ins}^{A} term is the sum of $L_{recon}^{A_{ins}}$ and $L_{adv}^{A_{ins}}$; further, L_{ins}^{B} is defined similarly.

[0048]

$$L_{ins}^{A} = L_{recon}^{A_{ins}} + L_{adv}^{A_{ins}}$$
$$= \left[L_{s_{re}}^{A_{ins}} + L_{l_{re}}^{A_{ins}}\right] + L_{adv}^{A_{ins}} \#(7)$$

[0049]

[0050] 백그라운드 손실(background loss)에 대하여 설명하기로 한다.

[0051] 백그라운드 모델은 $VAE_{bgr}^{A},\ VAE_{bgr}^{B},\ G_{bgr}^{A},\ G_{bgr}^{B}$ 로 구성될 수 있다. 백그라운드 모델과 인스턴스 모델은 통합 이전에는 서로 독립적이다. 백그라운드 모델의 시각적 도메인은 복원과 변환 스트림(reconstruction and translation stream)을 따른다. 백그라운드 모델은 VAEs와 GANs를 표현하는 $L_{VAE},\ L_{GAN},\ L_{sc}$ 와 의미론적 일관성 손실(semantic-consistency loss)를 사용할 수 있다. 세 개의 가중치 파라미터 $\lambda_{v},\ \lambda_{G},\ \lambda_{sc}$ 는 각각의 컴포넌트의 영향을 측정하는데 적용될 수 있다.

$$L_{bgr}^{A} = \lambda_{v} L_{VAE_{bgr}^{A}}(E_{bgr}^{A},G_{bgr}^{A}) + \lambda_{G} L_{GAN_{bgr}^{A}}(C_{bgr}^{A},G_{bgr}^{A},D_{bgr}^{A}) + \lambda_{SC} L_{SC_{bgr}^{A}}(C_{bgr}^{A},G_{bgr}^{A},E_{bgr}^{B},G_{bgr}^{B}) \#(8)$$

[0052]

[0053] VAE 구조는 학습 데이터셋의 주변 로그 우도(marginal log-likelihood) 근사를 통해 잠재 모델(latent model) 학습을 목표로 하는 ELBO(the lower bound of latent codes)를 사용하며, 목적 함수는 다음과 같다.

$$L_{VAE_{bgr}^{A}}(E_{bgr}^{A},G_{bgr}^{A}) = \lambda_{1} KL(q_{bgr}^{A}(C_{bgr}^{A}|A_{bgr}) \,\|\, p_{C}(C_{bgr})) -$$
$$\lambda_{2} E_{A_{bgr}\sim p(A_{bgr}),C_{bgr}^{A}\sim p(C_{bgr}^{A})}[log\, p_{G_{bgr}^{A}}(A_{bgr}|C_{bgr}^{A}) \#9$$

[0054]

[0055] 상기 목적 함수에서 가중치 파라미터인 $\lambda_{1},\ \lambda_{2}$ 는 목적 함수와 쿨백-라이블러 발산(KL divergence) 의 영향력을 조절한다. 쿨백-라이블러 발산은 $q_{bgr}^{A}(C_{bgr}^{A}|A_{bgr})$ 이 얼마나 사전 확률 $p_{C}(C_{bgr})$ 와 비슷한가를 추정하는 평가 지표이다. 사전 확률은 공통 컨텐츠 공간(common content space) C_{bgr}^{A} 의 분포를 나타낸다. 실시예에서는 샘플링의 성능을 높이기 위하여 라플라시안 분포(Laplacian distributions)를 따르는 $p_{G_{bgr}^{A}}$ 와 표준 분포를 따르는 $\{q_{bgr}^{A}(C_{bgr}^{A}|A_{bgr}),\ p_{C}(C_{bgr})\}$ 를 각각 모델링 할 수 있다.

$$L_{GAN^A_{bgr}}(C^A_{bgr}, G^A_{bgr}, D^A_{bgr}) = E_{A_{bgr}\sim p(A_{bgr})}\left[log\, D^A_{bgr}(A_{bgr})\right] + E_{C^A_{bgr}\sim p(C^A_{bgr})}\left[log(1 - D^A_{bgr}\left(G^A_{bgr}(C^A_{bgr})\right))\right] \#(10)$$

[0056]

[0057] GAN의 목적 함수는 적대적 프로세스에서 이미지를 변환하고 복원하는 것을 목표로 한다.

$$L_{SC^A_{bgr}}(E^A_{bgr}, G^A_{bgr}, E^B_{bgr}, G^B_{bgr}) = E\left[\left\|E^S_{A_{bgr}}(G^A_{bgr}(C^A_{bgr})) - C^A_{bgr}\right\|_1\right] + E\left[\left\|E^B_{bgr}(G^B_{bgr}(C^A_{bgr})) - C^B_{bgr}\right\|_1\right] \#(11)$$

[0058]

[0059] 의미론적 일관성 손실 함수(semantic-consistency loss function)는 이미지가 원래의 잠재적 공간으로 다시 맵핑될 수 있도록 하며, 이때, 목표 도메인의 특징은 잃지 않도록 한다. 실시예에 따른 모델은 UNIT 처럼 L1 norm (l_1) 을 사용하여, 잠재적 공간의 거리를 측정하는 쿨백-라이블러 발산을 대신하여 바로 의미론적 차이를 비교할 수 있다.

[0060] 도 6은 일 실시예에 따른 도메인 변환 시스템에서 이미지를 변환하는 것을 설명하기 위한 도면이다.

[0061] 일 실시예에 따른 도메인 변환 시스템은 이미지를 백그라운드 영역과 인스턴스 영역으로 분리한 다음, 분리된 두 영역을 독립적으로 변환 네트워크에 입력할 수 있다. 그리고 나서, 저장된 라벨을 사용하여 변환된 두 영역을 통합할 수 있다. 이때, 통합 과정을 통해 어떻게 실제처럼 보이게 할 것인지, 단순히 자르고 나서 각각을 변환하고 통합하기만 한다면 백그라운드 영역과 인스턴스 부분이 완전히 다른 방향을 갖기에 통합 결과가 좋지 않을 것이라는 점을 고려해야 한다.

[0062] 도 6을 참고하면, "smoothing"이 없는 경우의 변환을 나타낸 예로서, 이미지는 백그라운드 영역(Bgr)과 인스턴스 영역(Ins)으로 나뉘고, 백그라운드 영역 및 인스턴스 영역이 백그라운드 네트워크인 $model_{bgr}$과 인스턴스 네트워크인 $model_{ins}$으로 입력될 수 있다. 변환이 종료된 후, 두 네트워크에서 생성된 결과가 전체 이미지로 통합될 수 있다.

[0063] 통합 후 이미지를 사실적으로 유지하기 위하여 라벨 스무딩(label smoothing) 기술이 사용될 수 있다. 구체적으로, GAN은 판별기가 입력과 모델 데이터 사이의 비율을 추정할 수 있을 때 효과적임이 알려져 있다. x 로 표현되는 어느 포인트에 대하여 아래와 같이 정리될 수 있다.

$$\frac{P_{model}(x)}{P_{img}(x)} \#(12)$$

[0064]

[0065] 이전 단계의 조건에 따라 판별기를 아래와 같이 학습시킬 수 있다.

$$D^A_{bgr} \rightarrow \frac{P_{G^A_{bgr}}(B_{bgr})}{P(A_{bgr})}$$
$$D^B_{bgr} \rightarrow \frac{P_{G^B_{bgr}}(A_{bgr})}{P(B_{bgr})} \#(13)$$

[0066]

[0067] (인스턴스) 복원 모드:

$$D_{ins}^{A} \rightarrow \frac{P_{G_{ins}^{A}}(A_{ins})}{P(A_{ins})}$$

$$D_{ins}^{B} \rightarrow \frac{P_{G_{ins}^{B}}(A_{ins})}{P(B_{ins})} \#(14)$$

[0068]

[0069] (인스턴스) 다중 출력 모드:

$$D_{ins}^{A} \rightarrow \frac{P_{G_{ins}^{A}}(B_{ins})}{P_{ins}(A_{ins})}$$

$$D_{ins}^{B} \rightarrow \frac{P_{G_{ins}^{B}}(A_{ins})}{P_{ins}(B_{ins})} \#(15)$$

[0070]

[0071] $model_{bgr}$이 백그라운드 네트워크를 나타내고, $model_{ins}$이 인스턴스 네트워크를 나타낸다고 하자. 모델에 의해 생성된 출력이 통합 과정을 거친 경우, 전체 이미지 I_A, I_B로 나타낸다고 하면, 다음과 같이 나타낼 수 있다.

$$I_A : \{model_{bgr}(A_{bgr}), model_{bgr}(A_{ins})\}$$

$$I_B : \{model_{bgr}(B_{bgr}), model_{bgr}(B_{ins})\} \#(16)$$

[0072]

[0073] 통합 후, B, I_A와 A, I_B를 구분하기 위해 다른 한 쌍의 판별기 {D_A, D_B}가 추가될 수 있다. 통합이 정확히 원본 도메인 A와 B같지 않으므로, 파라미터 α를 추가하여 훈련 데이터를 스무딩(smoothing)한다. 파라미터는 아래와 같이 주어질 수 있다.

$$\alpha^{A} = \frac{Area(A_{ins})}{Area(A)}$$

$$\alpha^{B} = \frac{Area(B_{ins})}{Area(B)} \#(17)$$

[0074]

[0075] 그런 다음, 파라미터 α를 판별 프로세스에 추가하기 위한 한 쌍의 판별기 {D_A, D_B}를 이용하여 비율을 측정할 수 있다.

$$D_A \rightarrow \frac{P_{G_{bgr}^{A}}(B_{bgr})}{(1-\alpha^{A})p(A) + \alpha^{A} P_{G_{bgr}^{A}}(B_{bgr})}$$

$$D_B \rightarrow \frac{P_{G_{bgr}^{B}}(A_{bgr})}{(1-\alpha^{B})p(B) + \alpha^{A} P_{G_{bgr}^{A}}(A_{bgr})} \#(18)$$

[0076]

[0077] Smoothing 파라미터 α 가 원본 이미지와 비교하여 출력을 덜 확신하게(less confident)하고 출력의 스타일에 영향을 주기는 하지만, 이 조절(adjustment)은 통합을 더 실제적으로 만들고, 이미지가 완전히 다른 두 이미지에서 합성된 것을 눈치챌 수 없게 한다.

[0078] 도 7을 참고하면, smoothed 판별기(discriminator)를 설명하기 위한 도면이다. 입력 이미지 A로부터 인스턴스 영역 A_{ins}와 백그라운드 영역 A_{bgr}를 분리하고, 인스턴스 모델 $model_{ins}$와 백그라운드 모델 $model_{bgr}$을 통해 전체

이미지 I_A가 라벨을 통해 생성될 수 있다. 실시예에 따른 모델은 새로운 판별기 D_A를 I_A와 목표 이미지 B를 구분하기 위하여 추가할 수 있다.

[0079] 도 8은 일 실시예에 따른 도메인 변환 시스템의 구성을 설명하기 위한 블록도이고, 도 9는 일 실시예에 따른 도메인 변환 시스템에서 이미지 대 이미지 변환을 수행하는 방법을 설명하기 위한 흐름도이다.

[0080] 도메인 변환 시스템(800)의 프로세서는 분리부(810), 학습부(820) 및 통합부(830)를 포함할 수 있다. 이러한 프로세서의 구성요소들은 도메인 변환 시스템(800)에 저장된 프로그램 코드가 제공하는 제어 명령에 따라 프로세서에 의해 수행되는 서로 다른 기능들(different functions)의 표현들일 수 있다. 프로세서 및 프로세서의 구성요소들은 도 9의 이미지 대 이미지 변환을 수행하는 방법이 포함하는 단계들(910 내지 930)을 수행하도록 도메인 변환 시스템(800)을 제어할 수 있다. 이때, 프로세서 및 프로세서의 구성요소들은 메모리가 포함하는 운영체제의 코드와 적어도 하나의 프로그램의 코드에 따른 명령(instruction)을 실행하도록 구현될 수 있다.

[0081] 프로세서는 이미지 대 이미지 변환을 수행하는 방법을 위한 프로그램의 파일에 저장된 프로그램 코드를 메모리에 로딩할 수 있다. 예를 들면, 도메인 변환 시스템(800)에서 프로그램이 실행되면, 프로세서는 운영체제의 제어에 따라 프로그램의 파일로부터 프로그램 코드를 메모리에 로딩하도록 도메인 변환 시스템(800)을 제어할 수 있다. 이때, 프로세서 및 프로세서가 포함하는 프로세서는 분리부(810), 학습부(820) 및 통합부(830) 각각은 메모리에 로딩된 프로그램 코드 중 대응하는 부분의 명령을 실행하여 이후 단계들(910 내지 930)을 실행하기 위한 프로세서의 서로 다른 기능적 표현들일 수 있다.

[0082] 단계(910)에서 분리부(810)는 입력 이미지와 목표 이미지 각각을 백그라운드 영역 및 인스턴스 영역으로 각각 분리할 수 있다. 분리부(810)는 입력 이미지와 상기 목표 이미지 각각으로부터 분리된 인스턴스 영역을 획득하고, 인스턴스 영역을 획득된 인스턴스 영역 이외의 나머지 영역을 픽셀의 평균으로 패딩(padding)하여 백그라운드 영역을 획득할 수 있다.

[0083] 단계(920)에서 학습부(820)는 분리된 입력 이미지와 목표 이미지 각각의 백그라운드 영역 및 인스턴스 영역을 이미지 변환을 위하여 생성된 변환 모델에 입력하여 학습시킬 수 있다. 학습부(820)는 이미지 변환을 위하여 백그라운드 모델 및 인스턴스 모델을 포함하는 변환 모델을 생성하고, 입력 이미지와 목표 이미지 각각의 백그라운드 영역을 변환 모델에 구성된 공통의 잠재적 공간으로 맵핑시키고, 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 각각의 인스턴스 영역에 의해 공유되는 스타일 코드 및 컨텐츠 코드를 매핑하여 복수 개의 인스턴스 영역을 생성할 수 있다. 이때, 백그라운드 모델은, 입력 이미지와 목표 이미지 각각의 백그라운드 영역을 백그라운드 네트워크에 입력하여 크로스 도메인 변환(cross-domain transformation)을 위한 학습을 수행할 수 있다. 인스턴스 모델은, 복원(reconstruction) 모드 및 다중 출력(multi-output) 모드로 구성되는 것을 포함하고, 복원 모드는, 입력 이미지와 목표 이미지 각각의 백그라운드 영역이 변화될 때, 인스턴스 네트워크를 통하여 입력 이미지와 목표 이미지 각각의 인스턴스 영역을 변화시키지 않으면서 통합 이미지가 기 설정된 기준 이상 일치하도록 학습을 수행하고, 다중 출력 모드는, 입력 이미지와 목표 이미지 각각의 인스턴스 영역을 인스턴스 네트워크에 학습시킴에 따라 각각의 인스턴스 영역에 임의의 코드를 추가하여 복수 개의 스타일을 변화시켜 다중 이미지의 출력을 획득하기 위하여 학습시킬 수 있다.

[0084] 단계(930)에서 통합부(830)는 학습을 통하여 획득된 변환 이미지를 분리된 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 기초하여 생성된 라벨(label)을 사용하여 통합할 수 있다. 통합부(830)는 입력 이미지와 목표 이미지 각각을 분리함에 따라 각각의 인스턴스 영역의 위치 값을 단위 행렬로 저장하여 라벨로 저장하고, 저장된 라벨을 사용하여 상기 변환 이미지를 통합하여 이미지를 변환시킬 수 있다.

[0085] 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, 예를 들어, 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 애플리케이션을 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.

[0086] 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, 가상 장치(virtual equipment), 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.

[0087] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다.

[0088] 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.

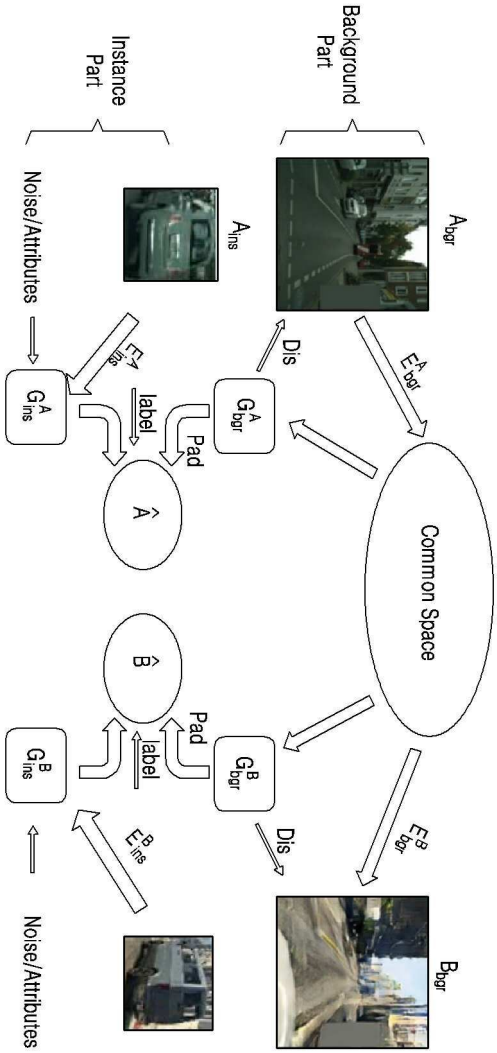
[0089] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

도면

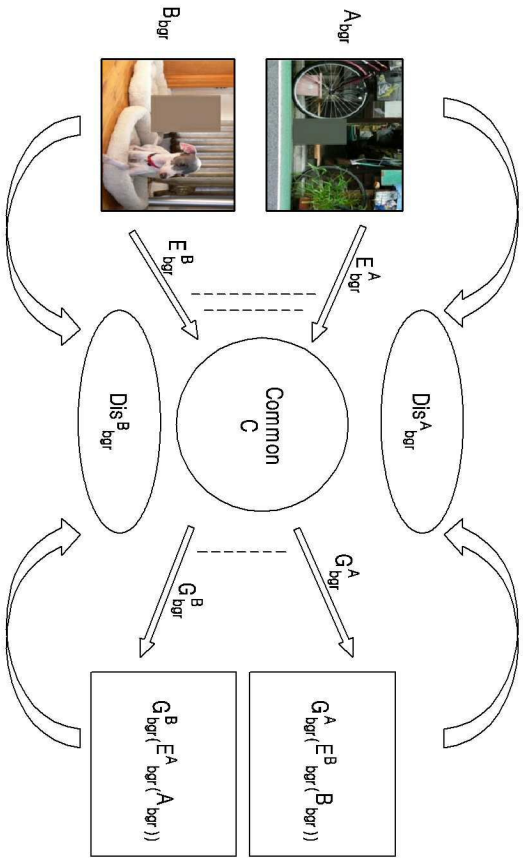
도면1



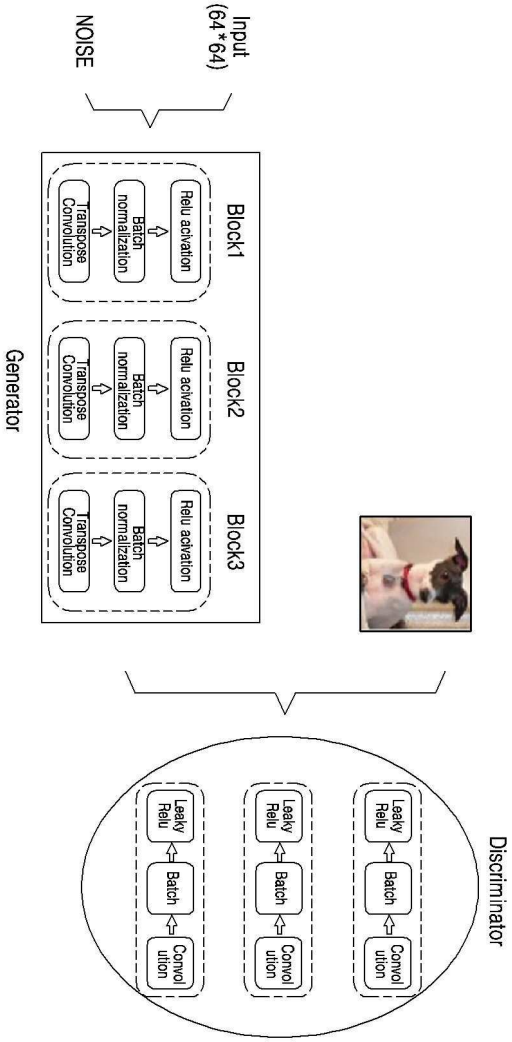
도면2


Background Part
Instance Part
Noise/Attributes
Common Space
label
Pad
Dis
Noise/Attributes

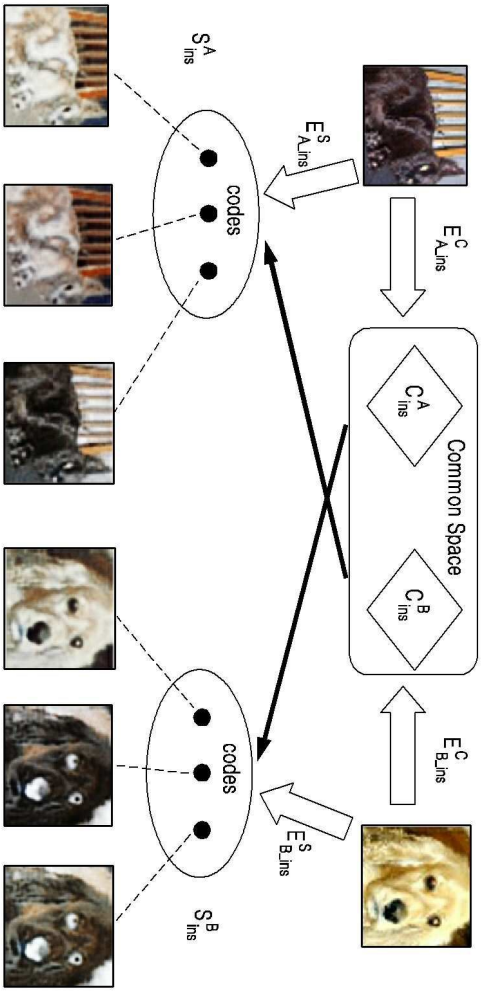
도면3


B bgr
A bgr
E B bgr
E A bgr
Dis B bgr
Common C
Dis A bgr
G B bgr
G A bgr
G B bgr(E A bgr(A bgr))
G A bgr(E B bgr(B bgr))

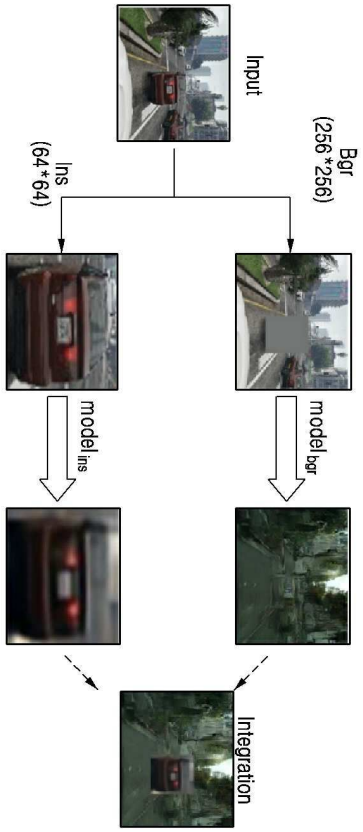
도면4


NOISE
Input
(64*64)
Generator
Block1
Block2
Block3
Relu activation
Batch normalization
Transpose Convolution
Discriminator
Leaky Relu
Batch
Convol ution

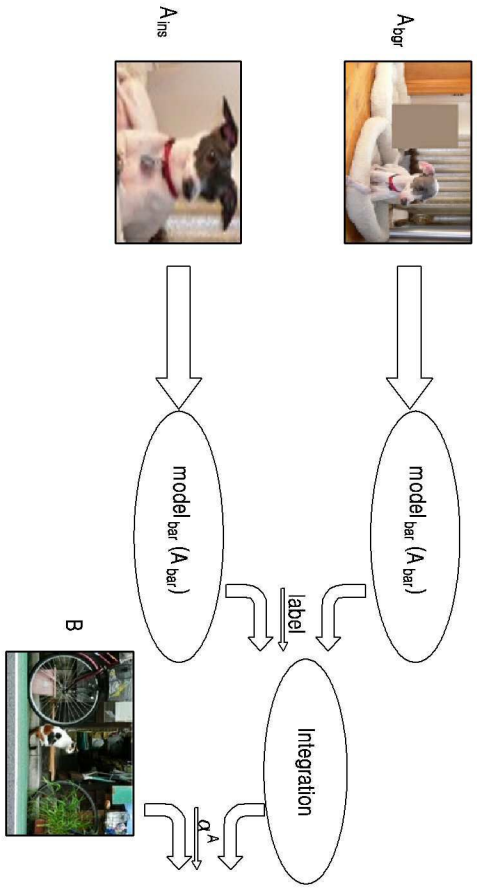
도면5


S^A_{ins}
$E^S_{A_ins}$
codes
$E^C_{A_ins}$
C^A_{ins}
Common Space
C^B_{ins}
$E^C_{B_ins}$
codes
$E^S_{B_ins}$
S^B_{ins}

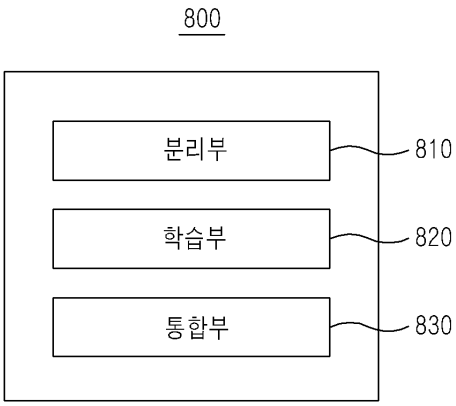
도면6



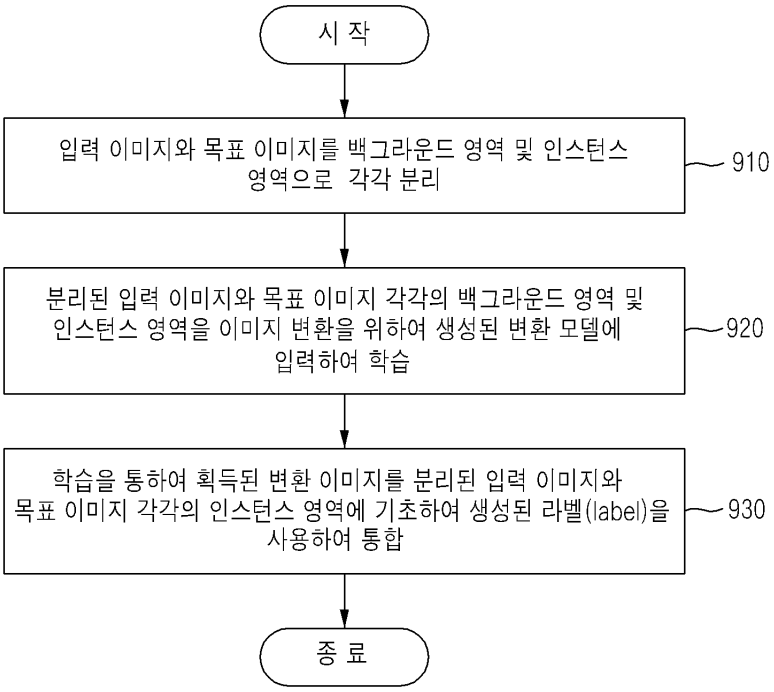
도면7


A ins
A bgr
model bar (A bar)
model bar (A bar)
label
B
Integration
α A

도면8


800
분리부
810
학습부
820
통합부
830

도면9


시 작
입력 이미지와 목표 이미지를 백그라운드 영역 및 인스턴스 영역으로 각각 분리
910
분리된 입력 이미지와 목표 이미지 각각의 백그라운드 영역 및 인스턴스 영역을 이미지 변환을 위하여 생성된 변환 모델에 입력하여 학습
920
학습을 통하여 획득된 변환 이미지를 분리된 입력 이미지와 목표 이미지 각각의 인스턴스 영역에 기초하여 생성된 라벨(label)을 사용하여 통합
930
종 료