정 보 보 호 학 회 지
제 30 권 제5호, 2020. 10

데이터 기반 딥페이크 탐지기법에 관한 최신 기술 동향 조사

김 정 호*, 안 재 주*, 양 보 성**, 정 주 연***, 우사이먼성일****

요 약

최근 전 세계적으로 '가짜뉴스', '가짜 연예인 음란 동영상' 및 '지인 능욕'에 사용되는 인공지능 기반의 딥페이크(Deepfakes)기술이 사회적인 이슈로 대두되고 있다. 딥페이크 기술이란 딥러닝 기술을 이용해 악의적으로 조작된 음성, 영상, 이미지 등을 만들어 내는 방법으로, 인공지능 기술의 발전에 맞추어 더욱더 빠르고 정교한 생성 기술이 등장하고 있다. 이러한 딥페이크 기술은 빠른 개발 속도와 쉬운 접근성을 기반으로 다양한 범죄에 악용되고 있다. 본 논문에서는 다양한 딥페이크 생성 기술을 설명하고, 이를 효율적으로 탐지 할 수 있는 다양한 데이터 기반 딥페이크 탐지 기술의 현황을 설명한다.

Ⅰ. 서 론

최근 전 세계적으로 정치인 및 연예인 동영상 합성 동영상에 사용되는 인공지능 기반의 딥페이크(Deepfakes) 기술은 점점 더 큰 문제가 되고 있다. 특히, 딥페이크기술이 단순 재미, 유희 및 풍자를 넘어서 가짜 뉴스 생성[1], 언론 조작[2], 폭동 생성[3], 및 음란물[4] 생성에 악용되고 있다. 특히 전체 딥페이크 음란물 사이트의 포르노 영상 중에 약 25%가 한국의 K-POP 여가수를 대상으로 한 것으로 조사되었으며[5], 주위의 일반인을 대상으로 영상을 조작, 유포하는 범죄인 '지인 능욕'은 N번방 사건에서 보여지듯, 새로운 디지털 성범죄 유형으로 추가되었다. 그럼으로 딥페이크는 앞으로 심각한 사회적 뿐만 아니라 법적 문제를 야기할 것으로 예상된다.

딥페이크라는 용어는 2017년 미국의 소셜네트워크 플랫폼 레딧에서 'Deepfakes'라는 아이디를 가진 회원이 유명 배우의 얼굴로 조작된 가짜 음란 동영상을 올리면서 사용되기 시작하였다[6]. 이처럼 딥페이크 악용 기술이란 GAN(Generative Adversarial Network)이나 오토인코더(Autoencoder) 등의 딥러닝 기술을 이용해 나쁜 의도를 가지고 조작된 음성, 영상, 이미지 등을 만들어 내는 방법을 말한다. 딥페이크 악용 기술은 인공지능 기술의 발전에 기생하여 더 빠르고 정교한 방법으로 변화하고 있으며, 특히 다음과 같은 문제점들이 있다. 첫째로는 딥페이크 기술의 긍정적인 면에 비하여 부정적인 의도로 악용되는 경우가 압도적으로 높다는 것이다. 의료, 교육 등 여러 분야에서 긍정적으로 사용될 가능성이 있음에도 불구하고, 현재 온라인에 존재하는 딥페이크 영상의 약 96%가 불법 음란 동영상으로 확인되고 있다[7]. 두 번째는 기술이 전문 기술자가 아닌 일반인도 쉽게 사용할 수 있도록 퍼져있다는 점이다. 컴퓨터 및 모바일 애플리케이션은 물론, 딥페이크 영상을 비용을 받고 바로 만들어주는 서비스도 온라인을 통해서 쉽게 접근할 수 있다. 이런 기술의 공개성을 통해 누구나 쉽게 악용할 수 있는 환경이 구성되었다.

이 논문에서는 현재 개발된 최신의 딥페이크 데이터셋 및 생성알고리즘을 중점적으로 살펴보고, 데이터 및 인공지능 기반의 최신 탐지 기술들을 설명한다.

* 성균관대학교 수학과(대학생, rlawjdghek@g.skku.edu), 을지대학교 의료IT마케팅학과(대학생,anjaeju@gmail.com)
** 아주대학교 사이버보안학과(대학생, ghtlda1@gmail.com)
*** 숙명여자대학교 컴퓨터과학과(대학생, jy364@naver.com)
**** 성균관대학교 데이터사이언스융합학과/소프트웨어학과 (조교수 swoo@skku.edu)

Ⅱ. 딥페이크 데이터셋 및 생성 알고리즘

2.1. FaceForensics++

딥페이크 영상은 컴퓨터 그래픽 또는 인공지능 기술에 의해 만들어진 실제로는 존재하지 않는 영상을 말하며 대표적으로 원본(source) 영상의 얼굴을 목표(target) 영상의 얼굴로 바꾸어 가짜 영상을 생성한다.

이러한 가짜 영상은 한 사람의 표정 또는 감정을 다른 사람의 얼굴에 전달하는 얼굴 재연 방법(Face reenactment)과 한 사람의 얼굴을 아예 다른 사람의 얼굴로 바꾸는 얼굴 교체 방법(Face replacement)으로 크게 두 가지 범주로 나눌 수 있다[8].

[그림 1]에 따라, 저자[8]는 이러한 가짜 영상물을 탐지하는 알고리즘의 개발을 돕기 위해 대규모 얼굴 위조 데이터베이스(Large Scale Facial Forgery Database)를 만들고 이를 자동적으로 탐지하는 알고리즘을 소개한다. 추가로 제안된 데이터베이스에서 특정한 기준에 따라 선별한 데이터를 딥페이크 탐지 벤치마크(benchmark)로 제안한다.

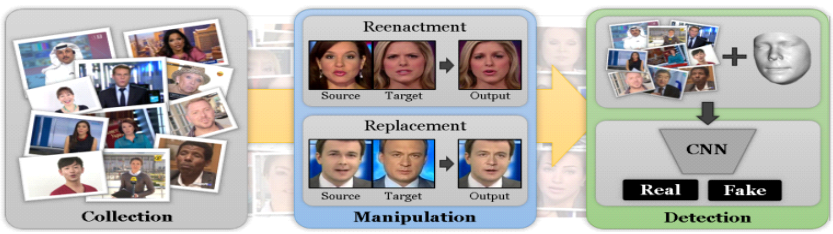
현재 딥페이크 영상 생성 기술은 크게 컴퓨터 그래픽스 기반 방법(computer graphics-based methods)과 학습 기반 방법(learning-based approaches)으로 나눌 수 있다.

FaceForensics++ 데이터셋은 컴퓨터 그래픽스 기반 방법은 Face2Face와 FaceSwap을 사용했고 학습 기반 방법은 DeepFakes와 NeuralTextures를 사용했다. 언급한 생성 방법들은 모두 입력 데이터로 한 쌍의 원본 영상과 목표 영상을 필요로 하며 아웃풋은 합성 이미지 또는 비디오이다. 저자는 현실적인 환경을 위해 유튜브에서 수집하였으며 수집한 영상 속 얼굴이 가려지는 문제를 피하기 위해 비디오를 수동 선별 하였다. 최종적으로 509,914개의 이미지를 포함하는 1,000개의 비디오를 선택하여 사용하였다.

Face2Face방법은 얼굴 재연 방법으로 목표 영상의 얼굴을 그대로 유지하면서 원본 영상의 얼굴 표정을 목표 영상에 전송하는 방법이다. 해당 방법은 원본과 목표 두 개의 동영상 프레임들을 이용한다. 저자는 다음의 처리 과정을 통해 딥페이크를 생성하였다. 먼저 3D model 작업을 위한 임시 얼굴로 첫 번째 프레임을 사용하고, 나머지 프레임에 대하여 표정을 추적하였다. 키 프레임을 선택할 때는 가장 왼쪽 각도와 오른쪽 각도의 얼굴을 선택하였다. 비디오의 전체 프레임에 대하여 파라미터로 표정, 포즈 그리고 조명을 추적한 뒤 원본 영상의 파라미터를 목표 영상에 전송하여 표정을 재구성하였다. 더욱 자세한 방법은 원 논문에서 찾아볼 수 있다[9].

FaceSwap은 얼굴 부위를 원본 영상에서 목표 영상으로 전송하는 그래픽 기반 접근 방식이다. 우선 얼굴의 랜드마크(landmark)를 탐지하여 얼굴 지역을 추출한다. 추출에 사용된 랜드마크를 이용해, 3D 템플릿 모델을 만든 후 혼합 형상(blenshapes)방법을 사용해 랜드마크를 적용시킨다. 3D 템플릿 모델은 투영된 모양과 랜드마크 간의 차이를 최소화시키며 대상 이미지에 투영된다. 마지막으로, 렌더링 된 모델은 이미지 혼합과 색상 보정이 적용되면서 자연스럽게 만들어진다[10].

DeepFakes는 현재 일반적인 딥러닝에 기반을 둔 얼굴 교체 방법과 동의어로 사용되고 있으나, 이 또한 특정한 얼굴 조작 방법으로 사용된다. 이 방법은 목표 영상 속 얼굴을 원본 영상이나 이미지속의 얼굴로 바꾸는 방법이다. 해당 방법은 하나의 인코더(encoder)를 공유하는 두 개의 오토인코더로 구성되어있다. 인코더는 입력 데이터를 압축된 공간에 변환시키고 디코더는 압축된 공간의 정보를 출력 데이터로 변환시키는데, 이때의



〔그림 1〕 FaceForensics++〔8〕의 프로세스 흐름도. 동영상을 SNS를 통해 수집 후 가짜 동영상을 제작 및 배포함. 알고리즘을 이용해 가짜 영상을 탐지하는 방법까지 제시.

입력과 출력은 모두 이미지다. 하나의 공유 인코더는 원본과 목표를 각각 재구성하도록 학습된다. 따라서 원본 재구성용 오토인코더 1개, 목표 재구성용 오토인코더 1개, 총 2개의 오토인코더를 학습하는 것이다. 가짜 영상을 생성할 때, 각각의 압축된 공간을 서로 바꿔 출력시킴으로써 원본 영상에 대해서 학습된 인코더와 디코더가 목표 얼굴에 적용된다. 그 후, 오토인코더의 아웃풋은 이미지를 자연스럽게 합성하는 Poisson 수정방법을 통해 나머지 부분이 처리된다[11].

NeuralTextures는 얼굴 재연 방법으로, Thies[12]가 NeuralTextures에 기반을 둔 렌더링 접근법을 소개하였다. 해당 방법은 목표 얼굴의 neural texture을 측광 재구성 손실(photometric reconstruction loss)과 적대적 손실(adversarial loss)의 구성으로 학습한다. 저자는 구현으로 Pix2Pix[13]에서 사용된 패치 기반(patch-based) GAN-loss를 적용하였다. NeuralTextures는 학습 및 테스트하는 동안에 사용된 추적 기하학적 정보(tracked geometry)에 의존하며, 저자[12]는 Face2Face 모듈을 사용해 추적하였다.

실제와 비슷한 환경의 데이터를 생성할 수 있도록 아웃풋 영상의 해상도를 소셜 네트워크(social network)에서 비디오 처리에 사용되는 수준과 유사하게 생성하였다. 특히 소셜 네트워크나 비디오 공유 사이트에서 자주 사용되는 H.264 코덱을 이용해 압축하였다, 고화질 영상(HQ)의 경우 23을 정량화 매개변수로 압축하였고, 이는 거의 시각적으로 잃는 정보가 없다고 한다. 저화질 영상(LQ)은 40으로 정량화가 진행되었다.

대규모 데이터 셋과 더불어, 저자는 얼굴 위조 탐지를 위한 벤치마크(benchmark)를 배포하였다. 벤치마크 데이터를 위해 추가로 1천개의 동영상을 수집하였고, 앞서 설명한 가짜 영상 방법을 통해 생성하였다. 조작된 각각의 영상을 시각적 검사(visual inspection)를 통해 탐지하기 힘든 하나의 프레임을 직접 골랐으며 총 1,000 개의 이미지를 수집하였다.

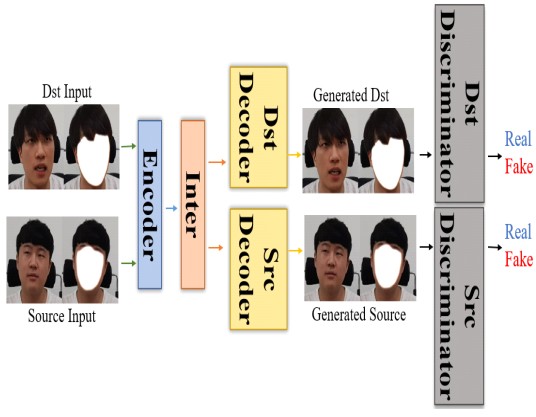
FaceForesics++[8]데이터셋은 대부분 서양인 얼굴 데이터셋을 기반으로 제작되었기 때문에 한국인 얼굴 데이터셋은 적은 것이 문제점이다. 또한 딥페이크 생성 기술들은 매우 빠르게 발전하고 있어 벤치마크 데이터셋이 발전되는 기술들을 포함하지 못한다는 것도 문제점으로 파악된다.

2.2. DeepFaceLab

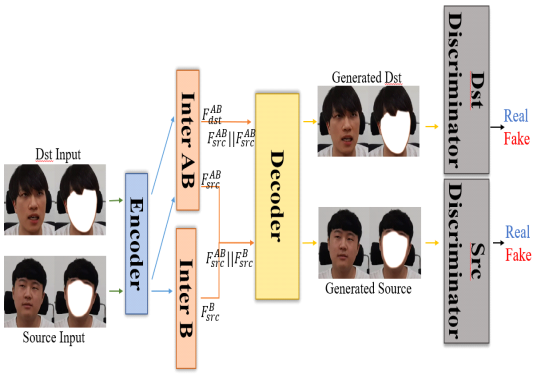
다음으로 GAN[14] 을 기반으로 하는 최신 딥페이크 영상 생성 알고리즘 DeepFaceLab[15]에 대하여 논의한다. DeepFaceLab은 딥페이크 영상 생성 기법 중 DeepFakes[8]에 해당 된다. 즉 원본 영상의 눈, 코, 입 등 얼굴의 특징을 목표 영상의 표정, 반응으로 나타내는 기법이다.

DeepFaceLab은 크게 특징 추출, 학습, 전환 세 가지 단계로 구분할 수 있다. 먼저 특징 추출 단계에서는 전체 이미지에서 얼굴이 있는 부분을 포괄하여 탐지(Detection) 한 뒤, 얼굴의 특징에 따라 랜드마크를 표시하여 표정을 나타낸다. 이 과정은 목표 영상의 표정을 결과물에 명확히 드러나게 하기위한 것으로 눈, 코, 입의 모양에 따라 표시된 랜드마크가 표정을 잡는데 큰 도움을 준다. 마지막으로 딥페이크 생성에서 중요한 역할을 하는 분할(Segmentation) 작업에서 DeepFaceLab은 Xseg프로그램을 적용해 뛰어난 성능을 보인다. 다양한 구도와 표정에서 전체 이미지의 3~5%의 수작업으로 추출된 마스크(Mask)로 전체 이미지를 훈련하여 머리카락과 얼굴의 경계를 확실히 구분하는 최종 마스크를 생성한다.

원본 이미지와 추출 단계에서 생성된 마스크(Mask)를 입력으로 학습을 시작한다. LIAE 모델보다 상대적으로 가벼운 DF 모델은 인코더-디코더 구조에서 Inter 레이어의 삽입으로 변화를 주었다. 합성곱(Convolutional Neural Networks) 신경망으로 구성된



〔그림 2〕 DF 구조, 가중치를 공유하는 인코더와 Inter 레이어를 학습시킨 뒤 원본 영상과 목표 영상 각각의 출력에 대하여 학습시킨 뒤, 판별기로 진위 여부를 분류한다.



〔그림 3〕 LIAE 구조, 모델의 크기에서 큰 비중을 차지하는 완전 연결층이 포함된 Inter 레이어를 늘려 무겁지만 높은 해상도를 출력한다.

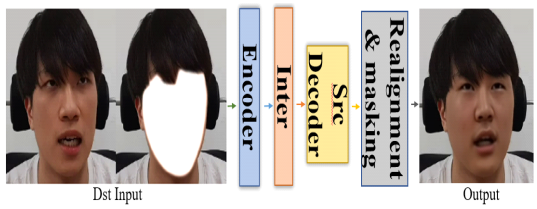
인코더, 디코더와는 달리 LIAE 모델은 두 개의 FC(Fully-Connected) 레이어와 특징맵의 업샘플링(up-sampling)으로 원본 영상과 목표 영상의 가중치를 공유한다. 좀 더 복잡한 LIAE모델은 두 개의 Inter 레이어를 사용함으로써 목표 영상에서 인코더, Inter 레이어를 거친 출력이 두 개가 되며, 이를 연결하여 원본 영상과 목표 영상 각각의 방향성을 더욱 뚜렷하게 표현한다. DeepFaceLab은 학습에서의 손실 함수로 원본 영상과 목표 영상 두 얼굴의 일반적인 특징을 추출하기 위한 SSIM(Structure Similarity)과 영상의 명확성을 추출하는 MSE(Mean Square Error)를 사용한다. 이 때 손실 함수를 0으로 수렴시키기 위하여 아래와 같이 DSSIM(Structure Dissimilarity)를 사용한다.

$$SSIM = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (1)$$

$$DSSIM = \frac{1 - SSIM}{2} \quad (2)$$

$$LOSS = DSSIM + MSE \quad (3)$$

훈련을 마친 후, DF 모델을 기준으로 훈련된 6개의 레이어를 획득한다. 우리는 얼굴이 변환된 영상을 얻기 위하여 Deepfakes[16]와 같은 다른 생성 모델과 마찬가지로 목표 영상을 인코더, Inter 레이어, 원본 디코더를 거친 출력을 생성한다. 마지막으로 분할 작업에서 추출한 목표 영상의 마스크를 제외한 나머지 부분은 모두



〔그림 4〕 출력 결과, 목표 영상을 입력으로 원본 영상의 얼굴을 합성하기 위하여 목표 영상의 파이프라인에서 원본 영상의 디코더로 교체한다.

일치해야하므로 최종 손실 함수는 다음과 같다.

$$I_{output} = M_t \odot I_t' + (1 - M_t) \odot I_t \quad (4)$$

더 사실적인 딥페이크 영상을 얻기 위하여 DeepFaceLab은 최종 결과물에서 원본 영상의 마스크 영역을 감소시키거나 피부색 명암 조절 등 다양한 후처리 작업도 지원한다.

하지만 높은 완성도를 보이는 DeepFaceLab은 하나의 합성 영상을 만들기 위해 대응되는 하나의 모델을 만들어야 하고, 전체 생성 과정에서 마스크를 만드는 수동 작업이 필요하기 때문에 많은 합성 영상을 만드는 작업에서 자동화가 불가하다는 단점이 있다. 또한 [그림 5]에서 볼 수 있듯이, 안경과 같은 액세서리를 한 목표에 얼굴을 합성한다면 원본영상에서 학습할 수 없는 안경은 마스크 내에서 반영이 되지 않는다.

반면, DeepFaceLab은 짧은 수렴 시간으로 전체적인 생성 시간은 짧다는 장점과 액세서리가 없는 영상에서 수동 후처리 작업으로 미세한 경계까지 보완하는 높은 해상도의 영상을 생성한다.

〔그림 5〕 실패 예시. 첫 번째, 세 번째 이미지에서 얼굴에 해당하는 부분에서 귀와 안경의 연결이 끊어짐을 볼 수 있다. 마찬가지로, 가운데 이미지에서도 얼굴 내에서 학습되지 않은 안경은 표현되지 않는다.

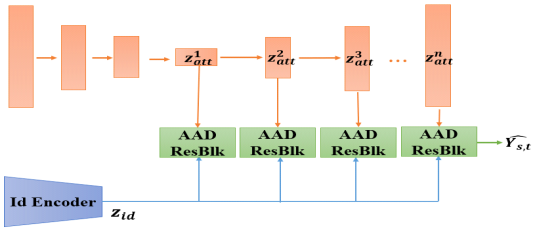
2.3. Faceshifter

다음으로 FaceShifter[17]는 [그림 5]와 같은 가림 현상을 방지하고, 높은 해상도의 딥페이크 이미지를 생성하기 위하여 두 단계의 모델 구조를 제시한다.

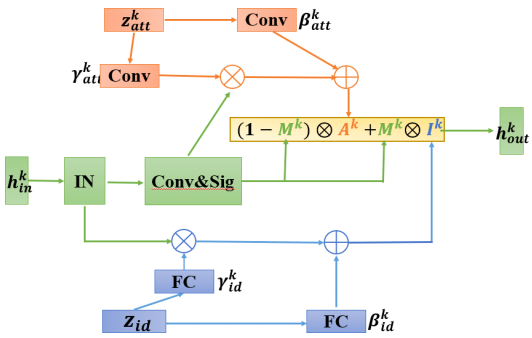
첫 번째 단계에서, [그림 6]에서 볼 수 있듯이, AEI-Net은 얼굴의 특징을 담은 원본 영상을 인코더로 입력하여 특징을 추출한다. 마찬가지로 표현을 담은 목표 영상을 입력으로 인코더-디코더 구조의 모델에 넣은 후 레이어에서 각각의 표현 벡터들을 얼굴의 변환이 완성된 이미지 $\widehat{Y_{s,t}}$를 생성한다. 주목할 점은 모든 레이어에 특징 벡터를 넣음으로써 얼굴의 특징을 보존하고, 다중 레이어의 출력을 모두 사용함으로써 다각도의 표현들을 학습할 수 있다.

다음으로 FaceShifter에서 핵심이 되는 AAD 블록은 이전 AAD블록의 출력에서 합성곱 레이어와 시그모이드 함수를 취하여 얼굴의 특징을 잡는 마스크를 생성한다. 표현벡터는 특징의 나머지에 해당하는 부분을 추출하여 더해준다.

AEI-Net의 특징 손실(L_{id})은 코사인 유사도로 생성된 이미지와 원본 영상(X_s)의 유사도를 구한다.



〔그림 6〕 AEI-Net〔17〕



〔그림 7〕 AAD 블록〔17〕

$$L_{id} = 1 - \cos(z_{id}(\widehat{Y_{s,t}}), z_{id}(X_s)) \quad (5)$$

FaceShifter는 목표 영상(X_t)의 표현을 학습하기 위하여 다중 레이어의 개별 출력을 사용하므로 표현 손실(L_{att})을 다음과 같이 정의하였다.

$$L_{att} = \frac{1}{2}\sum_{k=1}^{n} \| z^k_{att}(\widehat{Y_{s,t}}) - z^k_{att}(X_t) \|^2_2 \quad (6)$$

재구성 손실(L_{rec})은 장애물 문제를 해결하기 위한 HEAR-Net에서 사용된다. 재구성 손실과 AEI-Net의 손실 함수는

$$L_{rec} = \begin{cases} \frac{1}{2} \| \widehat{Y_{s,t}} - X_t \|^2_2 & \text{if } X_t = X_s \\ 0 & otherwise \end{cases} \quad (7)$$

$$L_{AEI-Net} = L_{adv} + \lambda_{id} L_{id} + \lambda_{att} L_{att} + \lambda_{rec} L_{rec}, \quad (8)$$

이며, 가중치 계수 $\lambda_{id} = 5, \lambda_{att} = \lambda_{rec} = 10$를 갖는다.

DeepFaceLab은 원본 영상과 목표 영상의 경계가 불연속적이다. 즉, 각 영상마다 별도의 모델을 사용함으로써 얼굴 경계 내에서는 경계 밖에 존재하는 치장 등 학습되지 않은 물체에 대해서는 표현할 수 없다. 반면, FaceShifter는 이 문제를 해결하기 위하여 U-Net[18] 구조의 HEAR-Net을 사용한다. 학습되지 않은 물체를 합성 영상에 입히기 위하여 재구성 영상($\triangle Y_t$)을 구한 뒤,

$$\triangle Y_t = X_t - AEINet(X_t, X_t) \quad (9)$$

최종 이미지($Y_{s,t}$)를 생성한다.

$$Y_{s,t} = HEARNet(\widehat{Y_{s,t}}, \triangle Y_t) \quad (10)$$

AEI-Net과 마찬가지로, HEAR-Net의 특징 손실(L'_{id}), 변동 손실(L'_{chg}), 재구성 손실(L'_{rec})을

$$L'_{id} = 1 - \cos(z_{id}(Y_{s,t}), z_{id}(X_s)) \quad (11)$$

$$L'_{chg} = \left| \widehat{Y_{s,t}} - Y_{s,t} \right| \tag{12}$$

$$L_{rec}{}' = \begin{cases} \frac{1}{2} \| Y_{s,t} - X_t \|_2^2 & \text{if } X_t = X_s \\ 0 & otherwise \end{cases} \tag{13}$$

로 정의한다. 결론적으로, HEAR-Net의 전체 손실 함수는 (14)로 정의한다.

$$L_{HEAR-Net} = L'_{id} + L'_{chg} + L'_{rec} \tag{14}$$

선명한 해상도와 물체의 가림 현상을 보완한 FaceShifter를 소개하였다. AEI-Net은 AAD 블록을 활용하여 별도의 마스킹 작업 없이 얼굴의 특징과 표현을 섞을 수 있고, 두 번째 단계인 HEAR-Net으로 합성 영상에 치장을 더한다. 하지만, 공개된 코드가 없어 정확한 성능을 파악할 수 없다는 점과 합성 과정이 길기 때문에 이미지를 합성하는 데 시간이 오래 걸린다는 단점에 비하여 원본 영상과 목표 영상 선택 폭의 확대, 별도의 수작업이 없는 단일 연속적인 파이프라인을 갖고 있다. 또한 영상과 모델의 일대일 대응이 아닌 합성 영상 생성의 일반화된 모델이라는 점에서 큰 이점을 갖는다.

Ⅲ. 데이터 기반 딥페이크 탐지 알고리즘

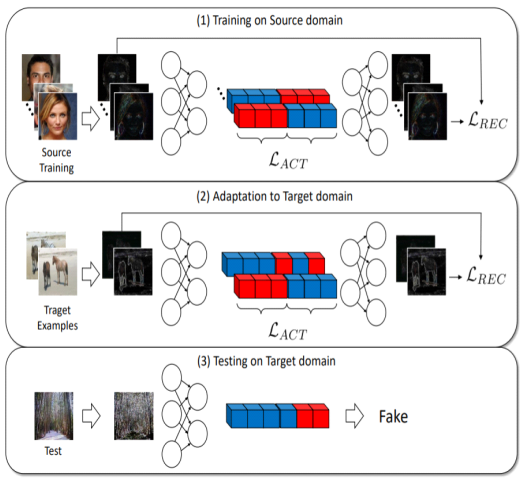
3.1. ForensicTransfer: Weakly-supervised Domain Adaptation for Forgery Detection

3.1.1. ForensicTransfer의 이론적 배경

현재 다양한 딥페이크 영상 생성 방법으로 생성한 콘텐츠를 탐지할 수 있는 모델의 연구가 활발히 진행되고 있다. 합성곱 신경망이 위조 탐지에 매우 효과적인 성능을 보이는 것이 증명되었다[8, 31, 32, 33, 34].

하지만, 일반적으로 이러한 방법들은 모두 학습 데이터에 의존하며 하나의 특정한 생성 방법에 과적합 되기 때문에 학습 시 보지 못했던 생성 방법에 대해서는 성능이 눈에 띄게 감소하는 문제가 있다. 디지털 콘텐츠분야에서 이미지 합성 및 생성방법은 매우 다양하기 때문에, 이를 해결하기 위해서는 각 영상 생성 방법마다 모델을 하나씩 만들어야하는 큰 단점이 있다. 이상적인 모델은 하나의 네트워크가 다양한 생성 방법으로 생성된 가짜 이미지를 구분할 수 있어야 한다. 이를 위한 방법 중 하나를 도메인 적응(domain adaptation)이라고 한다. ForensicTransfer[36]는 도메인 적응방법으로 이미 특정한 영상 생성 방법을 학습한 모델이 새로운 유형의 생성 방법으로 만든 이미지를 소량만 재학습 후 탐지에 사용하는 것이다. 전체적인 학습구조는 [그림 8]에 제시되어있다.

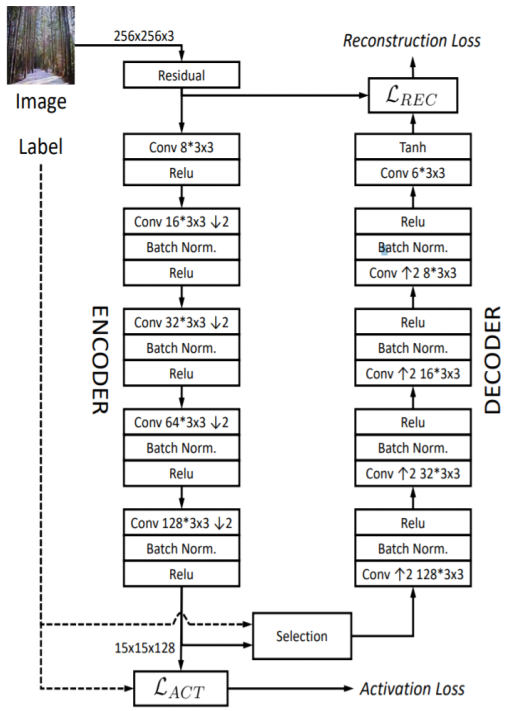
원본(source) 도메인과 목표(target) 도메인은 서로 다른 이미지 생성 방법을 통해 만들어진 데이터 셋을 가리킨다. ForensicTransfer는 우선 원본 도메인에 대해서 모델을 학습시킨 후, 목표 도메인에 대해서도 아주 소량 학습시킨다. 이렇게 두 가지 생성 방법을 학습한 모델을 평가하는데 사용한다. 해당 방법은 모델이 원본 도메인을 학습할 때 데이터를 진짜 혹은 가짜라고 분류하는데 이용한 정보를 새로운 유형의 목표 도메인에서 활용하자는 접근이다.



[그림 8] ForensicTranfer[36] 스키마. (1) 원본 도메인에 대해서 오토인코더 구조 네트워크를 학습시킨다. 이때, 네트워크는 latent space속 가짜와 진짜 이미지의 표현 방법을 학습한다. (2) 적은 양의 목표 도메인 데이터를 이용해 모델을 목표 도메인에 적응시킨다. (3) 학습된 latene space가 테스트 이미지를 분류하는데 사용된다.

3.1.2. ForensicTransfer의 구조 및 학습 과정

저자는 네트워크의 구조로 오토인코더를 사용하였으며 [그림 9]에 제시되어있다. 인코더와 디코더는 같은 구조로 3x3 커널을 갖는 5개의 합성곱 레이어로 구성되



〔그림 9〕 ForensicTranfer〔36〕 의 신경망 구조. 입력으로 먼저 잔차 학습을 위한 이미지를 추출한다. 그 후, 왼쪽의 인코더에 이미지를 넣어 진행한다. 학습 된 임베딩은 손실 함수와 디코더의 재구성 함수에 의해 제한된다

어 있다. 인코더에서 첫 번째 레이어를 제외한 모든 레이어에서 스트라이드(stride)를 2로 설정해 입력 대비 1/16의 해상도를 갖는다. 디코더에서는 원래의 해상도로 만들기 위해 마지막 레이어를 제외한 모든 레이어에서 2x2의 최근접 이웃 업샘플링(up-sampling)을 사용하였다. 활성화 함수의 경우 마지막 레이어에서는 hyperbolic tangent를 사용하였고, 나머지 모든 레이어에서 ReLU(Recified Linear Units)를 사용하였다.

인코더의 아웃풋인 latent space는 15x15 이미지가 128개로 구성되어 있는 특징맵이다. 128개 중 64개는 진짜 클래스와 연관되어있고, 또 다른 64개는 가짜 클래스와 연관이 있다. 선택 블록(selection block)은 학습하는 데이터의 클래스와 맞지 않는 부분을 0으로 설정하는 역할을 한다. 예를 들면, 학습하는 데이터가 진짜 클래스인 경우 가짜 클래스와 연관되어있는 64개의 특징맵의 값을 0으로 설정하는 것이다.

선택 블록은 디코더가 오직 같은 클래스에 해당하는 latent space로 부터 샘플 이미지를 재구성 하도록 하는 역할을 한다.

학습 시, 사용하는 손실 함수는 다음과 같다.

$$L = \gamma L_{rec} + L_{act} \tag{21}$$

오토인코더의 아웃풋 이미지에는 L_{rec}를, latent space에는 L_{act}를 적용하였다. λ는 L_{rec}의 영향력을 조절하는 가중치로 모든 실험에서 0.1로 설정하였다. L_{rec}은 인풋 이미지 x와 아웃풋 x'에 대한 차이를 측정한 것이다.

$$L_{rec} = \frac{1}{K} \sum_{x \in S_0 \cup S_1} \| x + x' \|_1 \tag{22}$$

S0와 S1은 각각 원본 도메인 데이터와 목표 도메인 데이터를 뜻한다. K는 인풋 샘플 수이다.

활성화 손실 함수 L_{rec}는 다음과 같이 정의된다.

$$L_{act} = \sum_{x \in S_0} |a_0(x) - 1| + |a_1(x)| + \sum_{x \in S_1} |a_1(x) - 1| + |a_0(x)| \tag{23}$$

여기서 a_0와 a_1은 각각 진짜와 가짜에 대한 특징맵의 L1-norm을 통해 구할 수 있다.

$$a_c(x) = \frac{1}{K_c} \| encoder(x) \|_1 \tag{24}$$

여기서 c는 클래스 {0, 1}이고 K_c는 각 클래스에 속한 샘플 수이다.

테스트 시, 위조 여부의 결정은 L_{ACT}의 활성화 된 강도에 의존한다. 만약 샘플 x가 가짜라고 여겨진다면 a1(x) > a0(x)이며, 진짜의 경우는 그 반대이다. 분류 문제에서 널리 사용되는 크로스 엔트로피 손실 함수와는 달리, 제안된 손실 함수는 두 개의 클래스에 대한 분류뿐만 아니라 클래스 내부적으로 분산을 줄인다.

ForensicTransfer는 도메인 적응 방법으로 목표 도메인에 대한 재학습이 필요하며, 재학습을 하지 않을 시 성능이 잘 나오지 않는 단점이 있다. 이는 모델이 목표 도메인의 데이터를 알고 있어야 한다는 것으로 현실적인 환경에서는 사용하기 힘든 기법이다. 미래에는 목표 도

메인에 대한 재학습 없이 다양한 딥페이크 영상을 탐지하는 일반화된 모델에 대한 연구가 이뤄져야 할 것이다.

3.2. T-GD

기존의 딥페이크 탐지 방법은 학습하지 않은 데이터에 대해서 낮은 성능을 보인다. T-GD(Transferable GAN-generated Images Detection Framework)[19]는 이러한 문제점을 해결하기 위해 전이학습을 활용한 GAN기반 딥페이크 탐지 방법이다. 본 장에서는 T-GD의 이론적 배경 및 구조 및 학습방법 그리고 T-GD의 성능 및 평가에 대하여 서술한다.

3.2.1. T-GD의 이론적 배경

전이학습은 기존의 잘 훈련된 모델을 활용하여 해당 모델과 유사한 문제를 해결하는 방법으로 이미 학습된 가중치를 활용하여 새로운 모델을 빠르게 학습시킬 수 있다. T-GD는 합성곱 신경망기반의 딥페이크 탐지 모델을 활용하여 다른 데이터셋의 딥페이크를 탐지하기 위해 L^2-SP를 적용한 전이학습을 사용한다.

Self training[20]은 Teacher Model의 예상 결과를 활용하여 Student Model을 학습시켜 도메인 적응(Domain Adaption)을 높이기 위한 방법이다. Student Model을 학습시킬 때 노이즈를 주입하여 과대적합을 방지한다. T-GD는 전이능력(transferability)을 향상시키고 규제(regularization)정도를 조절하기 위하여 Self Training을 사용한다.

3.2.2. T-GD의 구조 및 학습방법

T-GD는 전이능력을 향상시키기 위해 규제, 데이터 증감기법 (data argumentation), Self training 및 학습 전략을 활용한다.

우선, T-GD는 전이학습에 L^2-SP 규제를 활용하여 FC 레이어에만 L^2 규제를 가하여, 과대적합을 방지하고 Pre-trained model의 가중치를 보존한다.

L^2-SP규제를 적용한 손실함수는 다음과 같다.

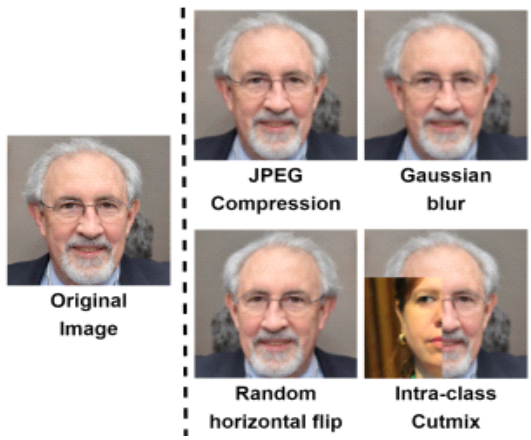
$$\min \frac{1}{n}\sum_{i=1}^{n} J(\hat{y_i}, y_i) + \alpha \cdot \Omega_{sp}(w, \acute{w}) + \beta \cdot \Omega_{l2}(w_{fc}) \quad (15)$$

(15)에서 J는 크로스엔트로피(cross-entropy)함수, Ω는 L2 norm, w는 미세조정(fine-tuning)된 가중치, $\acute{w}$는 Pre-trained model의 가중치이며, α와 β는 Self training 과정에서 자동으로 조정된다.

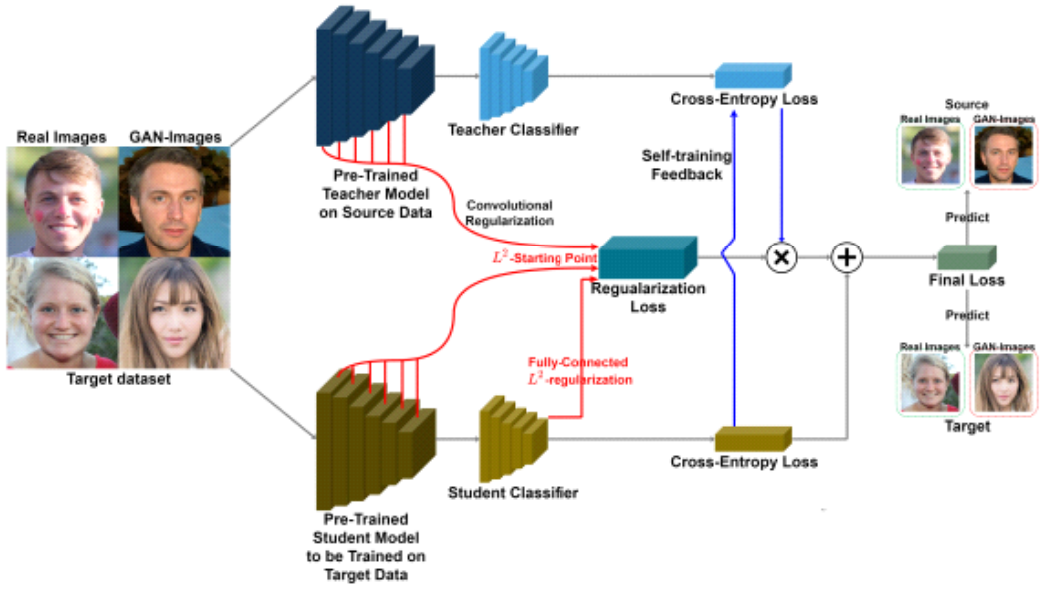
T-GD는 Self training시 데이터 증강 및 잡음을 추가하고, Drop out[22] 및 Stochastic depth[23]를 활용하여 과대적합 문제를 해결한다. 데이터 증강에는 JPEG compression[24], Gaussian blur[25], random horizontal flip, Cutmix[26]를 활용하였으며 그 예시는 [그림 10]과 같다.

T-GD는 학습 전략으로 가중치 표준화(Weight Standardization), 그룹 정규화(Group Normalization)을 활용하여 배치사이즈에 관계없이 높은 성능을 기록하도록 하였다. 또한 낮은 학습률과 낮은 관성(momentum)을 전이학습에 적용하여 다른 도메인에도 성공적으로 전이가 가능하도록 설계되었다.

전체적인 T-GD 학습 절차는 다음과 같다. 첫째, CNN기반의 딥페이크 탐지 모델을 학습한다. 이때 학습한 모델을 전이학습을 위한 Pre-trained model로 사용한다. 둘째, L^2-SP을 적용하여 전이학습을 진행한다. 이를 통해 Pre-trained model의 가중치 손상을 방지할 수 있다. 셋째, 전이학습 프레임워크를 Self training 프

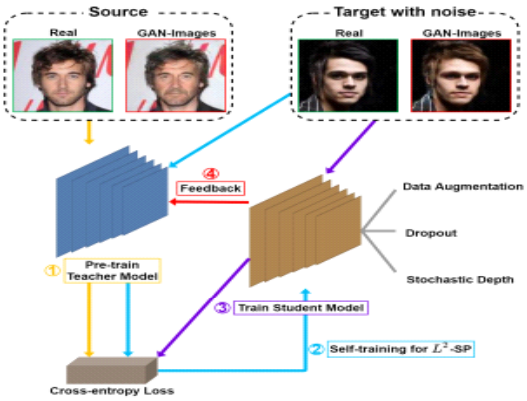


〔그림 10〕 데이터 증감기법이 적용된 이미지 예시[19],데이터 증감기법이 적용된 이미지는 원본과 비교하였을 때 각 기법에 따른 차이가 있음을 알 수 있다.



〔그림 11〕 T-GD 전체 구조[19]. 효율적인 전이학습을 위하여 빨간 부분은 L^2-SP를 사용하였고, 파란 부분은 셀프트레이닝을 사용하였다.

레임워크로 변환한다. 즉, Pre-trained model과 목표 모델의 관계를 Teacher model과 Student model의 관계로 변환한다. 이때 Teacher model은 (15)의 α와 β를 조정하는 추가적인 역할을 수행한다. 또한 전이학습을 수행할 때 이미지의 노이즈를 섞어 학습하여 과대적합을 방지한다. [그림 11]는 전체적인 T-GD구조를 나타내며 [그림 12]은 Self training구조를 나타낸다.



〔그림 12〕 셀프 트레이닝 구조[19]. 그림에 표시된 순서대로 셀프트레이닝 과정이 진행된다.

3.2.3. T-GD 성능평가

기존 전이학습 방법과 비교한 T-GD의 전이능력과 Self training, 데이터 증강기법의 효용성에 대하여 서술한다.

T-GD의 성능을 평가하는 기준은 AUROC(the Area Under a Receiver Operating Characteristic)이며 일반적인 전이학습 방법과 데이터 증감기법과 학습전략을 다르게 한 전이학습방법인 Forensic Transfer, 두 가지 전이학습 방법이 비교대상이다.

또한, T-GD의 Pre-trained model은 ResNet과 EfficientNet 두 가지 모델을 기반으로 구현되었다.

일반적인 전이학습 방법과 T-GD를 비교하였을 때, 일반적인 전이학습 방법은 원본(Source) 데이터셋과 목표(Target) 데이터셋 간의 성능간의 Trade-off가 존재한다. 예를 들어, 일반적인 전이학습방법 사용 시 PG-GAN은 99.86%의 AUROC를 기록하였으나, 목표 데이터셋에서 약 54%의 저조한 AUROC를 기록하였다. 반대로 StyleGAN는 전이학습 이후 47.37%의 낮은 AUROC를 기록하였지만, 목표 데이터셋인 StyleGAN, StarGAN에서는 약 90%의 높은 AUROC점수를 기록하였다. 반면에, T-GD는 전이학습 후 원본 데이터셋과 목표 데이터셋 모두 약 95%의 높은 성능을 기록하였다.

ForensicTransfer와 T-GD의 성능을 비교하였을 때, ForensicTransfer는 전체적으로 약 50%~75%의 낮은 AUROC를 기록한 반면, T-GD는 약 95%의 높은 AUROC를 기록하였다.

T-GD의 Pre-trained model에 따른 차이를 비교하였을 때 EfficientNet-B0가 ResNet보다 일반적으로 높은 AUROC를 기록하였다. 반면에 EfficientNet-B0의 파라미터 수는 ResNet의 1/7수준이다. 이를 통해 T-GD의 성능은 파라미터 개수와 직접적인 연관이 없음을 알 수 있다.

Self training 유무는 T-GD에서 유의미한 차이를 이끌어낸다. 전이학습 시 원본 데이터셋과 목표 데이터셋에서 AUROC 점수가 전체적으로 향상된다. 데이터 증강 기법 여부 또한 T-GD에서 유의미한 결과를 도출한다. 데이터 증강 기법을 적용하였을 경우 원본 데이터셋에서 약 10%의 AUROC가 증가된다.

T-GD는 다른 전이학습 대비 높은 전이능력을 기록하였고, 원본 데이터셋에 대한 정보손실이 없다. T-GD는 높은 전이능력을 바탕으로 새롭게 나타나는 딥페이크에 대해 전이학습을 통해 빠르게 대응할 수 있을 것으로 보인다. 또한 T-GD 이전 연구는 GAN의 메타데이터에 주목하였지만, T-GD는 GAN의 메타데이터를 활용하지 않기 때문에 더욱 보편적으로 사용될 수 있다. 하지만, GAN 탐지 기법의 발전과 더불어 GAN이미지 생성 기술 또한 발전하고 있어, 발전한 이미지 생성기술에 대한 데이터 수집은 여전히 해결해야 할 문제로 남아있다.

3.3. OC-FakeDect

이 연구[36]에서는 VAE(Variational Autoencoder)를 기반으로 딥페이크와 같은 이상 탐지 문제를 감지하는 one-class 분류 모델인 OC-FakeDect를 제안한다. 이 연구에서 사용한 방법은 가짜 딥페이크 이미지가 아닌, 실제 정상 사람 얼굴 이미지를 학습 데이터로 사용하여 딥페이크를 탐지하는데 일반화 할 수 있다는 큰 장점이 있다. 또한, one-class 모델 평가에 FaceForensics++ 벤치마크 데이터셋을 사용하였으며 실제 이미지만 가지고 훈련시켰지만 높은 정확도를 보인다.

3.3.1. OC-VAE(One-class Variational Autoencoder) 소개

One-class 분류 모델은 모든 관측치가 오직 하나의 클래스 “정상”(normal)에 속하며, 나머지 관측치는 “비정상 또는 ”이상치”(anomalies)로 간주된다. 오토인코더는 one-class 분류 혹은 이상 탐지에 사용된다. 오토인코더는 입력 이미지의 잠재적인 특징을 학습하면서 입력 데이터를 재구성하는데, 낮은 재구성 오류율을 보인다. 즉, “정상”(normal) 데이터는 학습에 사용되고, “비정상 또는 이상치”(abnormal) 데이터는 이상 점수로 재구성 오류율을 기록할 때 발견된다. 하지만, 오토인코더는 확률적인 기초가 없는 결정론적이고 차별적인 모델이다. 이러한 오토인코더를 개선하기 위해서 나온 모델이 VAE(Variational Autoencoder)[27] 이다. VAE는 확률적인 생성 모델로, 보정(calibrated) 확률을 제공하여 딥페이크 이미지 탐지 성능을 높인다. 따라서 이 연구에서는 OC-FakeDect를 적용하여 오토인코더보다 향상된 탐지 성능을 입증한다.

3.3.2. VAE와 손실 함수 및 재구성 점수

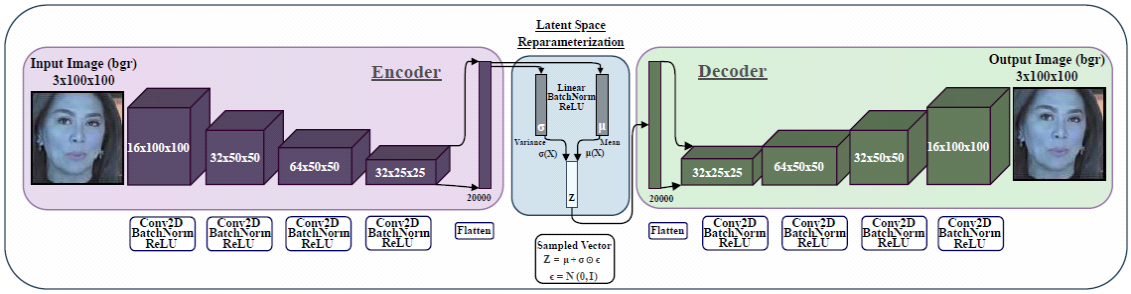
[그림 13]은 일반적인 OC-VAE의 구조를 나타낸다. OC-VAE는 DPGM(Directed Probabilistic Graphical Model)[28]으로 인코더와 발전기(디코더)로 구성되어 있다. VAE의 손실함수는 다음과 같다.

$$L(\sigma,\theta,x) = D_{KL}(q_\phi(z|x) \parallel p_\theta(z)) - E_{q_\phi(z|x)}(p_\theta(x|z)) \quad (17)$$

(17)의 첫 번째 항은 KL divergence[29]으로 latent space의 대략적인 앞과 뒷부분이다. 그리고 두 번째 항은 Monte Carlo 방법[30]으로 계산된다. 마찬가지로, OC-FakeDect의 손실 함수는 다음과 같다.

$$L_{OC-FakeDect} = D_{KL}[N(\mu(x),\sigma(x)), N(0,I)] + \parallel X - p_\theta^*(Z) \parallel^2 \quad (18)$$

(18)에서 X는 입력, $p_\theta(X|Z)$ 혹은 $p_\theta^*(Z)$은 디코더의 출력이다. D_{KL}은 잠재 공간에서 가우시안 분포인 $N(\mu(x),\sigma(x))$와 유사하도록 네트워크를 강화한다. [그림 13]의 하늘색 가운데 블록에 나타난 Z는 가우시



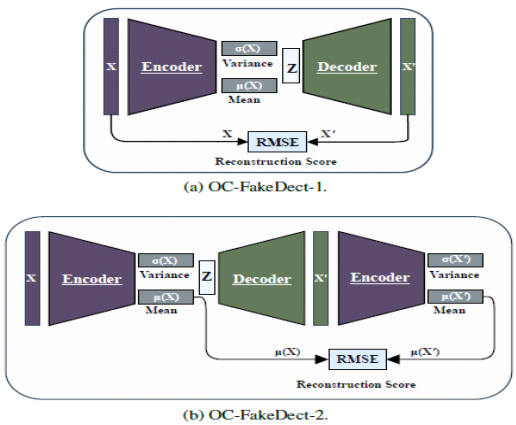
〔그림 13〕 FaceForensics++ dataset에서 추출한 실제 및 가짜 인간 얼굴 이미지의 예. 첫 번째 행은 실제 이미지를 포함한 반면 아래 다른 행은 DF, F2F, FS, NT 및 DFD dataset의 가짜 이미지를 포함한다.

안 분포로부터 얻은 확률 변수이다.

$$Z = \sigma(x) \times \varsigma + \mu(x), \varsigma \sim N(0, I) \tag{19}$$

(19)는 잠재 벡터 Z가 뒤쪽 분포를 따르게 하며, VAE를 훈련시키는 것과 유사하게 이 연구에서 제시한 모델을 훈련시킬 수 있도록 한다. 실제와 가짜 얼굴 이미지를 구별하기 위해서는 이미지들을 평가할 때 동일한 평가 방법을 사용하는 것이 중요하다. 이 논문에서는 재구성 점수를 계산할 때 RMSE(Root Mean Squared Error)로 계산한다.

$$\mathrm{RMSE} = \sqrt{\left(\frac{1}{n}\right)\sum_{i=1}^{n}(X_i' - X_i)^2} \tag{20}$$



〔그림 14〕 OC-FakeDect 구조: (a) OC-FakeDect-1은 입출력 이미지로부터 재구성 점수를 즉시 계산한다. (b) OC-FakeDect-2는 추가된 인코더 구조와 함께 입출력 잠재 정보로부터 재구성 점수를 계산한다.

(20)에서 X는 기존 입력, X'은 재구성된 출력이다.

3.3.3. OC-FakeDect 구조

[그림 13]에 나타난 OC-VAE 구조를 기반으로 논문에서는 OC-FakeDect-1와 OC-FakeDect-2를 제안한다. OC-FakeDect-1은 동일한 인코더와 디코더 블록을 사용한다. 또한 재구성 점수를 계산할 때는 기존 입력 X와 재구성된 출력 X'를 이용하여 (20)에 제시된 RMSE로 계산한다. OC-FakeDect-2는 디코더 뒤에 추가적인 인코더 블록이 필요하다. 그리고 첫 번째 인코더의 출력인 $\mu(X)$와 입력 이미지의 두 번째 인코더의 출력인 $\mu(X')$를 이용하여 RMSE를 계산한다. 이 연구에서는 더 높은 재구성 점수를 얻기 위해 추가적인 인코더 블록을 이용하여 디코더의 출력으로부터 실제 이미지 특징을 더 효과적으로 추출하도록 한다.

3.3.4. OC-FakeDect 성능평가

성능평가는 FaceForensics++ dataset으로부터 precision, recall 그리고 F1 score을 이용하였다. OC-FakeDect의 성능을 평가를 위해 OC-AE를 베이스라인으로 설정하였다. 두 OC-FakeDect은 0.465부터 0.669의 낮은 F1 점수를 받은 OC-AE보다 더 좋은 성능을 보인다. OC-FakeDect-2는 모든 데이터셋에서 가장 높은 F1 점수를 얻었다. 특히, OC-FakeDect-2는 DFD(Deepfake-detection)에서 가장 높은 정확도를 보였으며 NT(NeuralTexture)에서는 두 번째로 높은 정확도를 보였다. 따라서 OC-FakeDect은 다양한 유형의 가

짜 이미지를 탐지하는데 오토인코더보다 더 나은 수행을 보인다고 할 수 있다.

결과적으로 이 연구에서 제안한 OC-FakeDect는 추가적인 인코더 블록을 사용하여 실제 이미지의 특징을 배우는 것과 딥페이크와 같은 이상 탐지를 효과적으로 수행한다. 하지만, OC-FakeDect의 성능 평가를 위한 재구성 점수를 계산할 때 RMSE에만 의존하고 있다. 더 일반적인 성능 결과를 보이기 위해서는 재구성 점수 혹은 이상 점수를 계산할 때 더 개선된 성능 평가 방법을 찾을 필요가 있다. 또한, 연구에서는 모델을 평가할 때 Forensics++의 진짜와 가짜 이미지 데이터셋만 사용하였다. 다양한 실제 이미지를 사용한다면 OC-FakeDect를 사용하여 일반화되고 확장된 딥페이크 탐지를 할 수 있을 것이다.

Ⅳ. 결 론

본 논문에서는 최신 딥페이크 데이터셋 및 생성 알고리즘과 탐지방법에 대하여 살펴보았다. 딥페이크 생성 방법이 고도화되어감에 따라 더욱 실제 같은 가짜 동영상, 이미지 등이 사회에 악영향을 미칠 수 있다. 현재 개발된 딥페이크 탐지 방법은 준수한 성능을 갖고 있으나, 새로운 딥페이크 생성 방법 및 데이터셋에 대하여 취약할 수 있다는 문제점이 있으며 향후 연구를 통해 극복해야하는 과제이다.

또한, 딥페이크와 관련된 문제를 해결하기 위해서는 단순히 딥페이크 탐지 방법의 성능을 향상시키는 것 외에 체계화된 딥페이크 데이터셋 수집, 딥페이크 유포 방지 대책 및 처벌 방안 마련 등의 여러 전문 기관협력 및 논의가 필요하다.

참 고 문 헌

[1] Oscar Schwartz, "You thought fake news was bad? Deepfakes are where truth goes to die.", Nov 2018, 2020년 9월 접속, https://www.theguardian.com/technology/2018/nov/12/deep-fakes-fake-news-truth

[2] Regina Rini, "Deepfakes Are Coming. We Can No Longer Believe What We See.", June 2019, 2020년 9월 접속 https://www.nytimes.com/2019/06/10/opinion/deepfake-pelosi-video.html

[3] BBC NEWS, "Deepfake videos could 'spark violent social unrest", June 2019, 2020년 9월 접속, https://www.bbc.com/news/technology-48621452

[4] Lux Alptraum, "Deepfake Pron Harms Adult Performers, Too.", Jan 2020, 2020년 9월 접속, https://www.wired.com/story/deepfake-porn-harms-adult-performers-too/

[5] Alex Moltzau, "What Strategy Does Europe Have to Tackle Deepfakes?", 2020년 9월 접속, https://medium.com/dataseries/what-strategy-does-europe-have-to-tackle-deepfakes-fb159040f0c

[6] Wikipedia contributors. Deepfake —Wikipedia, the free encyclopedia, 2020.

[7] RIvan Mehta, "A new study says nearly 96% of deepfake videos are porn.", Oct 2019, 2020년 9월 접속, https://thenextweb.com/apps/2019/10/07/a-new-study-says-nearly-96-of-deepfake-videos-are-porn/

[8] Rossler, Andreas, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. "Faceforensics++: Learning to detect manipulated facial images." In Proceedings of the IEEE International Conference on Computer Vision, pp. 1-11. 2019

[9] Justus Thies, Michael Zollhofer, Marc Stamminger, Chris- ¨tian Theobalt, and Matthias Nießner. Face2Face: Real-Time Face Capture and Reenactment of RGB Videos. In IEEE pages 2387 –2395, June 2016

[10] Faceswap. https://github.com/Marek Kowalski/FaceSwap/.

[11] Deepfakes https://github.com/deepfa kes/faceswap. Accessed: 2018-10-29

[12] Justus Thies, Michael Zollhofer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. ACM Transactions on Graphics 2019 (TOG), 2019

[13] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. CVPR, 2017

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courvile, and Yoshua Bengio, “Generative adversarial nets”, In Advances in neural information processing systems, 27, June 2014.

[15] Ivan Perov, Daiheng Gao, Nikolay Chervoniy, Kunlin Liu, Sugasa Maranggonda, Chris Ume, Mr. dpfks, carl Shift Facenheim, Luis RP, Jian Jiang, Sheng Zhang, Pingyu Wo, Bo zhou, Weiming Zhang, “DeepFaceLab: A simple, flexible and extensible face swapping framework”, Computer Vision and Pattern Recognition, May 2020.

[16] Deepfakes. Deepfakes. https://github.co m/deepfakes/faceswap, 2017.

[17] Lingzhi Li, Jianmin Bao, Hao Yang, Dong Chen, Fang Wen, "FaceShifter: Towards High Fidelity And Occlusion Aware Face Swapping", Computer Vision and Pattern Recognition. Dec 2019.

[18] Olaf Ronneberger, Philipp Fischer, Thomas Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", Computer Vision and Pattern Recognition. May 2015.

[19] Hyeonseong Jeon Youngoh Bang, Junyaup Kim and Simon S.Woo, "T-GD: Transferable GAN-generated Images Detection Framewok", ICML, 2020

[20] RoyChowdhury, A., Chakrabarty, P., Singh, A., Jin, S., Jiang, H., Cao, L., and Learned-Miller, E. Automatic adaptation of object detectors to new domains using selftraining. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 780–790, 2019.

[21] Li, X., Grandvalet, Y., and Davoine, F. Explicit inductive bias for transfer learning with convolutional networks. arXiv preprint arXiv:1802.01483, 2018.

[22] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1):1929–1958, 2014.

[23] Huang, G., Sun, Y., Liu, Z., Sedra, D., and Weinberger, K. Q. Deep networks with stochastic depth. In European conference on computer vision, pp. 646–661. Springer, 2016.

[24] Wang, S.-Y., Wang, O., Zhang, R., Owens, A., and Efros, A. A. Cnn-generated images are surprisingly easy to spot... for now. arXiv preprint arXiv:1912.11035, 2019

[25] Xuan, X., Peng, B., Wang, W., and Dong, J. On the generalization of gan image forensics. In Chinese Conference on Biometric Recognition, pp. 134–141. Springer, 2019.

[26] Yun, S., Han, D., Oh, S. J., Chun, S., Choe, J., and Yoo, Y. Cutmix: Regularization strategy to train strong classifiers with localizable features. In Proceedings of the IEEE International Conference on Computer Vision, pp. 6023–6032, 2019.

[27] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[28] Wikipedia contributors. Graphical model — Wikipedia, the free encyclopedia, 2020. [Online; accessed 15-February-2020].

[29] Wikipedia contributors. Kullback–leibler divergence —Wikipedia, the free encyclopedia, 2020.

[30] Wikipedia contributors. Monte carlo method — Wikipedia, the free encyclopedia, 2020. [Online; accessed 13-March-2020].

[31] Davide Cozzolino, Giovanni Poggi, and Luisa Verdoliva. Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection. In ACM Workshop on Information Hiding and Multimedia Security, pages 1–6, 2017

[32] Belhassen Bayar and Matthew C. Stamm. A deep learning approach to universal image manipulation detection using a new convolutional layer. In ACM Workshop on Information Hiding and

Multimedia Security, pages 5 - 10, 2016

[33] Nicolas Rahmouni, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Distinguishing computer graphics from natural images using convolution neural networks. In IEEE Workshop on Information Forensics and Security, pages 1 - 6, 2017

[34] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Mesonet: a compact facial video forgery detection network. arXiv preprint arXiv:1809.00888, 2018

[35] Cozzolino, Davide, Justus Thies, Andreas Rössler, Christian Riess, Matthias Nießner, and Luisa Verdoliva. "Forensictransfer: Weakly-supervised domain adaptation for forgery detection." arXiv preprint arXiv:1812.02510 (2018)

[36] Hasam Khalid and Simon S. Woo. OC-FakeDect: Classifying Deepfakes Using One-class Variational Autoencoder. CVPR, 2020

〈저 자 소 개〉

김 정 호 (Jeongho Kim)
2015년 3월 ~ 현재 : 성균관대학교 수학과 학사과정
<관심분야> 컴퓨터 비전, 딥페이크 탐지

안 재 주 (Jaeju An)
2014년 3월 ~ 현재 : 을지대학교 의료IT마케팅학과 학사과정
<관심분야> 컴퓨터 비전, 딥페이크 탐지

양 보 성 (Bosung Yang)
2017년 3월~현재 : 아주대학교 사이버보안학과 학사과정
<관심분야> 정보보호, AI보안

정 주 연 (Jooyeon Jung)
2017년 3월~현재 : 숙명여자대학교 소프트웨어학부 컴퓨터과학전공 학사과정
<관심분야> 컴퓨터비전, AI보안

우사이먼성일 (Simon S. Woo)
2019년 3월~현재 : 성균관대학교 데이터사이언스융합학과/소프트웨어학과 조교수
<관심분야> 딥페이크 탐지