

# 얼굴 탐지 기술을 활용한 딥페이크 영상의 저작권 침해 여부 탐지

김 동 엽\*, 이 상 진\*\*  
고려대학교 정보보안학과 (대학원생)\*  
고려대학교 정보보호대학원 (교수)\*\*

## Detection of Copyright Infringement in Deepfake Video using Face Detector

Dong-Yeob Kim\*, Sang-Jin Lee\*\*  
Dept. of Information Security, Korea University (Graduate Student)\*  
School of Cybersecurity, Korea University (Professor)\*\*

### 요 약

최근 딥페이크 기술을 활용하여 영상저작물에 다른 얼굴을 합성하고, 이를 배포하는 사례가 발생하고 있다. 모바일 앱과 상용프로그램을 통해 영상저작물을 무단으로 변경하고 영상 공유 플랫폼에 게시할 경우, 저작권 침해 문제가 발생한다. 영상저작물을 이용하여 제작된 딥페이크 영상을 탐지하기 위해, 얼굴 탐지 기술을 활용한 탐지 시스템을 제안한다. 영상저작물과 딥페이크 영상의 화면 내 얼굴 비율을 얼굴 탐지 기술을 이용하여 측정하고, 두 값의 비교를 통해 딥페이크 영상이 영상저작물을 활용하여 제작되었는지 판단할 수 있다. 이를 통해, 영상 공유 플랫폼에 무분별하게 게시되고 있는 저작권 침해 딥페이크 영상을 효과적으로 탐지 가능하다.

주제어 : 딥페이크, 딥러닝, 얼굴 탐지, 영상저작물, 저작권

### ABSTRACT

Recently, there have been many cases of manipulating faces using deepfake technology and distributing video productions. If a video production is changed through a mobile app or commercial program without permissions and posted on a video sharing platform, a copyright infringement problem occurs. We propose a detection system using face detection technology to detect deepfake videos produced using video productions. The ratio of the face in the screen of the video production and the deepfake video can be measured using face detection technology, and by comparing the two values, it can be determined whether the deepfake video was produced using the video productions. Through this study, it is expected that it will be possible to effectively detect copyright infringement deepfake videos that are indiscriminately posted on video sharing platforms.

**Key Words** : Deepfake, Deep Learning, Face Detection, Video Productions, Copyright

---

※ 이 논문은 2022년도 정부(문화체육관광부)의 재원으로 한국저작권보호원의 지원을 받아 수행된 연구임(No 2022. 저작권 특화 디지털포렌식 전문인력 양성사업)

▪ Received 21 February 2022, Revised 23 February 2022, Accepted 21 March 2022  
▪ 제1저자(First Author) : DongYeob Kim (Email: kdylove96@korea.ac.kr)  
▪ 교신저자(Corresponding Author) : Sangjin Lee (Email : sangjin@korea.ac.kr)

## I. 서 론

최근 영상편집물에 인공지능을 활용하여 타인의 얼굴이나 특정 신체부위를 합성한 딥페이크가 많이 제작되고 있다. 이러한 기술은 실존하지 않는 인물을 영상에 등장시키고, 신원보호가 필요한 인물의 얼굴 노출을 방지할 수 있을 뿐만 아니라 교육 영상물에도 활용될 수 있다. 하지만 아직까지는 이러한 순기능보다, 딥페이크를 악용한 사례가 빈번하게 발생하여 사람들이 딥페이크에 대해 부정적으로 인식하고 있는 것이 사실이다. 특히, 연예인, 아동의 얼굴을 음란물에 합성한 영상물이 디지털 성범죄로서 심각한 문제가 되고 있다. 대한민국은 이를 처벌하기 위해 「성폭력범죄의 처벌 등에 관한 특례법 제14조 제2항: 허위영상물 등의 반포등」을 2020년 6월부터 시행하였다. 또한, 2016년 미국 대통령 선거 이후, 선거 때마다 등장하는 딥페이크 영상은 공정한 선거를 방해하고 있다. 이러한 영상은 주로 후보가 허위발언을 하는 내용으로 제작되고 있어, 유권자가 잘못된 선택을 하게 유도할 수 있다.

한편, 모바일 앱과 상용 프로그램을 통해 누구나 쉽게 딥페이크 영상제작이 가능해짐에 따라, 영화, 뮤직비디오, 방송, 광고 등의 다양한 콘텐츠에 다른 얼굴을 합성하여 무단으로 배포하는 새로운 저작권 침해유형이 등장하였다. 이러한 저작물 콘텐츠를 활용하여 무단으로 영상을 제작·배포하는 행위는 저작권법 「제13조: 동일성 유지권」, 「제16조: 복제권」, 「제22조: 2차적저작물작성권」, 「제35조 2항: 저작물 이용과정에서의 일시적 복제」 등을 침해할 소지가 있다. 이에 대해 한국저작권보호원[1]은 딥페이크 기술을 이용한 저작권 침해 사례에 대해 조사하였다. 이 보고서에 따르면, 유튜브와 인스타그램에 게시된 다수 딥페이크 영상이 저작권법을 위반한 사실을 확인하였다.

본 논문은 유튜브, 인스타그램, 틱톡 등에 무분별하게 배포되는 딥페이크 영상의 저작권 침해 유형을 탐지하는 방안을 제시한다. 특정 영상을 탐지하기 위하여, 이미지 분류 기술을 기반으로 측정된 영상 유사도에 따라 분류하는 방법이 널리 사용된다. 이 중에서 영상저작물을 활용하여 제작된 딥페이크 영상은 원본 영상의 특성과 유사한 면이 있고, 이를 활용한 탐지 방법은 기존의 영상 유사도 측정에 필요한 연산수를 획기적으로 감소시켰다.

## II. 관련 연구

유사한 이미지를 찾거나 비슷한 이미지끼리 분류하는 연구는 많이 진행되어 왔다. 이미지 인식과 분류 경진 대회인 ILSVRC(ImageNet Large Scale Visual Recognition Challenge)에서 2012년에 AlexNet[2] 모델이 우수한 것을 시작으로, 최근에는 합성곱 신경망(CNN)을 기반으로 이미지의 피쳐(feature)를 추출하여 이미지 분류를 시행하고 있다.

합성곱 신경망은 다양한 크기의 이미지에 대한 피쳐를 추출하기 위해, 분류 필터가 존재하는 다수의 층(layer)으로 구성되는데, 각각을 합성곱층(convolutional layer)이라고 한다. 하나의 합성곱층마다 피쳐를 추출하고, 최종적으로 다차원의 피쳐 벡터(feature vector)를 만들어낸다. 결국 이미지 분류 모델은 이 피쳐 벡터간의 비교를 통해, 유사한 이미지를 찾거나 비슷한 이미지끼리 분류하는 작업을 수행한다.

이미지의 피쳐를 추출하기 위해 많이 쓰이고 있는 모델로는 대표적으로 AlexNet[2], GoogLeNet[3], VGG-19[4], ResNet[5]이 있다. 이 4가지 모델 모두 합성곱층에 존재하는 필터를 통해 1000개 이상의 피쳐 벡터를 생성한다. 결국, 이미지 유사도를 측정하기 위해서는 1000개 이상의 피쳐 벡터 간의 연산을 수행하여야 한다. 이러한 방식은 후처리 과정인, 어떤 이미지를 탐지하는가의 목적에 따라 다르지만 높은 정확도로 유사한 이미지를 찾거나 비슷한 이미지끼리 분류할 수 있다. 이를테면, 대표적인 얼굴 탐지 기술인 RetinaFace[6]는 95% 이상의 정확도로 얼굴을 탐지할 수 있다.

이미지 분류 기술을 활용하여 유사한 이미지를 찾는 것은 위와 같이 많은 연구를 통해 준수한 속도와 높은 정확도로 시행할 수 있지만, 영상을 대상으로 한다면 동일한 모델을 사용하여 해결할 수 없다. 초당 60프레임의 10초짜리 영상만 하더라도 600개의 이미지에 대한 피쳐 벡터를 생성해야 하며, 1,000개의 피쳐 벡터를 생성하는 모델을 사용했다면 비교해야 하는 연산수는 600,000번에 달한다. 이를 극복하기 위해, 3D 합성곱을 사용하여 영상 분류의 속도와 정확도를 높인 C3D[7] 모델이 제안되었다. 3D 합성곱은 기존의 2D 합성곱에 인접한 프레임과의 합성곱 연산이 결합된 구조로, 시간적 특성이 반영됐다고 볼 수 있다. 2D 합성곱은 새로운 프레임이 들어올 때마다 기존의 프레임에 대한 피쳐 정보가 매번 사라지지만, 3D 합성곱은 이를 보존할 수 있다. 다수의 프레임에 대해 동시에 합성곱 연산을 진행함으로써 연속된 프레임에 대한 정보를 반영하여 정확도를 향상시키고, 동시처리를 통한 속도를 개선했다. 이때, 3D 합성곱에서 한 번에 처리되는 단위는 클립으로 표현하며, C3D의 경우 1클립은 16프레임으로 구성된다.

영상의 피처를 효과적으로 추출하기 위하여, C3D 모델이 제안한 방법으로 유사한 영상을 탐색하기 위한 피처 벡터의 수를 유의미하게 줄이기는 했지만, 여전히 영상플랫폼에 존재하는 수많은 영상을 비교하기 위해서는 비교 연산을 해야 하는 피처의 수가 획기적으로 감소될 필요가 있다.

본 연구에서 다루는 딥페이크 영상은 영상 내에 인물의 얼굴이 반드시 존재하고, 특히 영상저작물의 경우에는 일반적으로 얼굴의 크기가 동적으로 변하는 특징을 가진다. 이러한 특징에 착안해, 영상 프레임 내 얼굴 비율이라는 피처를 사용하여 하나의 프레임에 비교 연산이 필요한 피처를 오직 실수(float) 하나로 표현할 수 있다. 이미지 분류 모델(ResNet)과 영상 분류 모델(C3D)을 기반으로 한 영상 유사도 측정 방법과 제시된 방법에 필요한 피처 수를 비교한 결과는 [표 1]과 같다. 이처럼 영상저작물을 활용한 딥페이크 영상을 분류하는데 있어서는, 제시된 방법을 통해 프레임 당 적은 피처를 사용하여 가능하다.

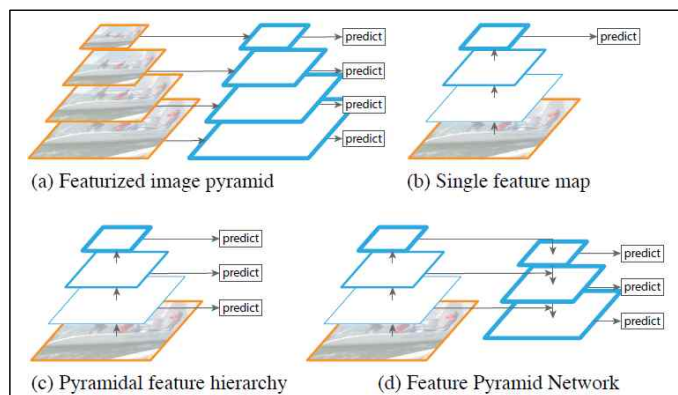
〈Table 1〉 Comparison of number of features per frame between models

Model	Features per frame
ResNet	4,096 vectors
C3D	256 vectors
<b>Proposed Method</b>	1 float

### III. 배경 지식

영상의 화면 내 얼굴 비율을 측정하기 위해 얼굴 인식, 추적 등 얼굴 분석 과정의 전 처리 단계인 얼굴 탐지 기술을 활용하였다. 그 중, 단일 단계 얼굴 탐지기(single-stage face detector)인 RetinaFace [6]를 사용하였다. 단일 단계 얼굴 탐지기란 2단계 탐지기(two-stage detector)와 달리 얼굴이 있을 만한 영역을 찾아내는 얼굴 위치 탐색 과정(Region Proposal)과 얼굴의 특징을 표현하여 분류하는 과정(Classification)을 동시에 진행한다. 이러한 방식은 비교적 빠르지만 정확도가 낮다. RetinaFace는 이러한 한계점을 극복하기 위해, 여러 얼굴 탐지 기술들을 채택하여 성능을 높였다.

RetinaFace는 FPN(Feature Pyramid Networks) [8]에서 설계하였다. 이미지에서 원하는 물체를 탐지하기 위해 여러 객체 탐지 방법이 연구되어 왔는데, [그림 1]은 이를 비교한 것이다. 먼저, Featurized image pyramid [9]는 입력 이미지의 크기를 다양하게 나누고 이를 독립적으로 계산하는 방식이다. 이는 서로 다른 크기의 이미지에 각각 합성곱 신경망(CNN)을 적용해야 하므로 속도가 매우 느리다. Single feature map [10]은 합성곱 신경망을 거친 최종의 피처 맵(feature map)만을 대상으로 객체 탐지를 수행한다. 이러한 방식은 신경망을 거칠수록 이미지에 존재하는 피처가 사라져 정확도가 낮은 문제점을 가진다. 피처가 손실되는 점을 고려하여, Pyramidal feature hierarchy는 신경망을 거치며 생성된 다양한 크기의 피처를 모두 사용한다. 하지만, 하위 층(layer)에서 생성된 피처는 상위 층에서 생성된 피처의 정보를 담지 못하고, 이에 따라 하위 층과 상위 층에서 생성된 피처 맵 간의 semantic gap이 발생한다. 이러한 문제를 해결하기 위해 [11]에서는 하위 층에서 생성된 피처 맵은 사용하지 않는 방법을 제시했으나, 이는 고해상도의 이미지에서 탐지할 수 있는 작은 물체를 탐지하기 어렵게 한다.



〈Figure 1〉 Comparison between object detection

FPN은 Pyramidal feature hierarchy와 마찬가지로 입력 이미지의 해상도를 낮추며 층마다 순차적으로 피쳐 맵을 생성한다. 그리고 상위 층 피쳐 맵의 해상도를 하위 층의 해상도와 맞추고 하위 피쳐 맵과 결합한다. 이로써 최종적으로 생성된 피쳐 피라미드(feature pyramid)에는 모든 층의 피쳐를 담고 있게 된다.

RetinaFace는 이러한 FPN 구조를 백본(backbone)으로 사용하여, 다양한 항목들을 병렬적으로 학습하였다. 기존의 연구들은 얼굴인지 배경인지 판단하는 Face classification과 얼굴의 범위를 나타내는 Face box regression만을 사용하여 학습시켰다. RetinaFace는 추가적으로 양 눈, 코 끝, 두 입꼬리로 이루어진 얼굴의 5가지 랜드마크를 예측한다. 이 방법을 통해 획기적으로 얼굴 탐지 성능을 높였다. 또한, 렌더링된 3D face 이미지를 원본 이미지와 픽셀단위로 비교하는 예측을 수행한다. 이 네 가지를 동시에 학습하지만, 단일 프로세스의 손실 함수(Multi-task Loss)를 구현하여 단일 단계로 구성된 얼굴 탐지 프레임워크가 RetinaFace이다. 성능 측정 결과, VGA(Video Graphic Array) 해상도 이미지 탐색을 단일 cpu 코어에서 실시간으로 실행 가능하였고, 최신 2단계 기술인 ISRN [12]보다 평균 정확도(Average Precision) 성능이 1.1% 우수하였다.

기계학습에서 모델의 분류 성능 평가를 위해 분류성능평가지표를 활용하는데, 이 지표를 활용하여 현재 모델의 성능을 측정하고 적절한 피드백을 통해 모델의 성능을 개선시킬 수 있다. 평가지표 중 정확도(Accuracy), 재현율(Recall), 거짓 인식률(False Positive Rate)을 사용하였다.

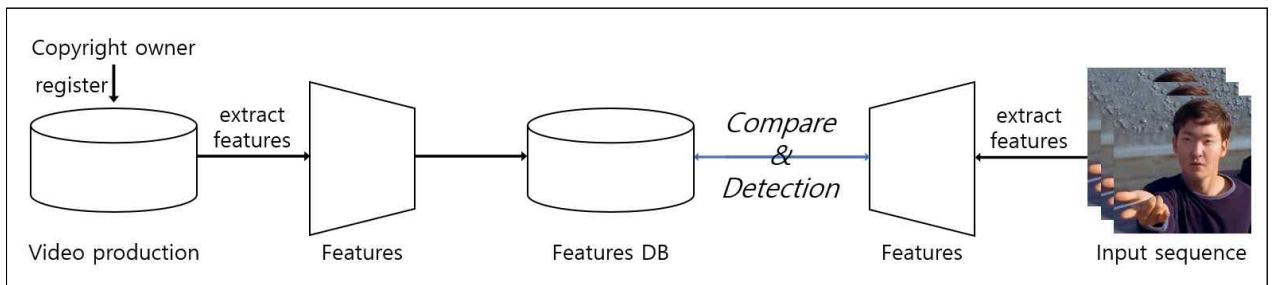
정확도는 탐지기가 옳은 탐지 결과를 낸 경우이다. 즉, 참(True)을 참이라 예측하고, 거짓(False)을 거짓으로 예측한 지표이다. 정확도는 탐지기의 성능을 가장 직관적으로 나타낼 수 있지만, 실제 값이 참인 경우가 거짓인 경우보다 상대적으로 매우 많거나 그 반대의 경우, 적은 표본을 탐지하는 성능이 낮을 수밖에 없다. 따라서 재현율과 거짓 인식률을 추가로 측정하였다. 재현율은 실제 값이 참인 경우 중에서 탐지 모델이 참이라고 예측한 경우이다. 마지막으로, 거짓 인식률은 실제 데이터가 거짓인 경우 중, 탐지 모델이 참이라고 예측한 비율이다. 성능평가지표를 측정하여 정확도와 재현율은 높을수록, 거짓 인식률은 낮을수록 탐지기의 성능이 좋다고 할 수 있다.

#### IV. 딥페이크 영상의 저작권 침해 여부 탐지 방법

딥페이크 영상의 저작권 침해 여부를 탐지하기 위해서는 영상저작물과의 유사도 측정 과정이 필요하다. 이러한 과정을 직접 영상원본데이터를 통해 실행할 경우, 영상저작물의 데이터의 양이 매우 방대할 뿐만 아니라, 유사도 측정도 많은 시간이 소요된다. 따라서 영상원본데이터와 직접 유사도를 측정하는 방법이 아닌, 영상의 특징을 추출해 이에 대한 유사도를 측정하는 방법을 제안한다.

##### 4.1 탐지 모델

새로운 딥페이크 영상이 수집됐을 때, 저작물을 무단으로 이용했는지 탐지하는 모델은 [그림 2]와 같다.



〈Figure 2〉 Detection model of copyright infringement in deepfake videos

해당 모델은 존재하는 모든 영상저작물에 대해 저작권 침해여부를 탐지하려는 것은 아니다. 저작권자는 패러디로써 본인 소유의 일부 영상저작물을 딥페이크 영상으로 활용하는 것을 허용할 수 있다. 이러한 패러디 영상은 원본 영상과는 독립적인 창작물로 인정되는 경우, 법적으로 허용될 수 있다. 따라서 제시한 모델에서는 저작권자가 영상공유플랫폼에 공유되고 있는 영상저작물 중 딥페이크를 통해 재생산·배포되지 않길 원하는 영상을 등록한다. 등록된 모든 저작물에 대해 프레임 내 얼굴 비율이라는 피쳐를 측정하고 이를 기록한다. 저작물 무단 이용 여부를 탐지하고 싶은 영상이 수집되면, 이 영상의 피쳐를 측정하고 시스템에 저장되어 있는 저작물

의 피처와 비교한다. 이때, 특정 기준치보다 유사도가 높게 측정된다면 저작물을 무단으로 이용하여 딥페이크 영상을 제작했다고 판단한다.

일반적으로, 구글 이미지 검색 등과 같이 이미지 유사도 측정과 이미지 분류를 하는 방법으로, 합성곱 신경망을 이용한다. 두 이미지의 피처 맵을 구하고 고차원의 피처 벡터 간의 거리를 비교하여 유사도를 측정한다. 이러한 방식은 영상의 유사도를 측정할 때는 부적합하다. 영상의 유사도를 측정하기 위해, 영상에 존재하는 수많은 프레임마다 고차원 피처 벡터를 저장하고, 이를 비교해야하기 때문이다. 따라서 본 논문에서는 얼굴 비율이라는 계산 값을 오직 실수 하나로 저장하고, 이를 비교하여 영상의 유사도를 측정하는 방식을 제안한다.

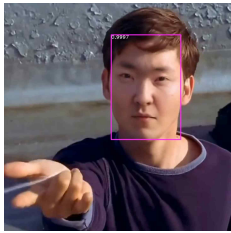
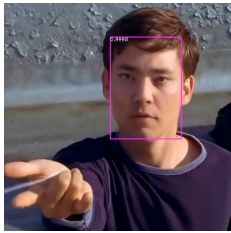
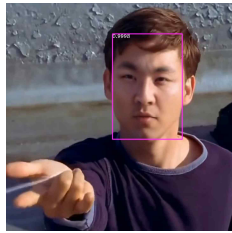
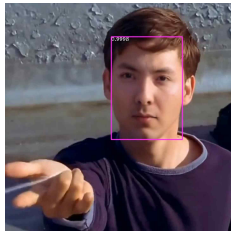
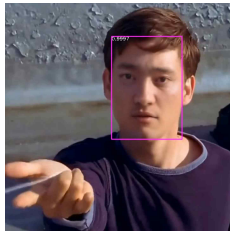
## 4.2 얼굴 비율 측정

같은 이미지에 각각 다른 얼굴을 합성하더라도 이미지 내 얼굴 비율은 매우 유사함을 [표 2]를 통해 알 수 있다. 딥페이크는 얼굴 외곽선을 포함한 얼굴 전체를 다른 얼굴로 바꾸는 것이 아닌, 얼굴 외곽선 안쪽의 얼굴 부분만을 합성하는 것이다. 특히, 머리 모양, 얼굴 형태, 귀, 피부톤, 그림자, 표정은 원본 영상의 것을 살리고 눈, 눈썹, 코, 입만을 합성하는 것이 일반적이다. 따라서 딥페이크 기술을 통해 타인의 얼굴로 바꾼 영상에서도 얼굴 외곽선을 따라 계산하는 화면 내 얼굴 비율은 원본 영상과 큰 차이를 보이지 않는다. 이러한 특징을 이용하여 저작물 영상 내 얼굴 비율을 주기적으로 기록한 데이터베이스를 구축하고, 새로운 영상이 수집되면 얼굴 비율을 프레임마다 비교하여 같은 영상이 데이터베이스에 존재하는지 탐색할 수 있다.

영상 내 얼굴 비율을 계산하기 위해, [6]의 예측 값인 Face-bounding-box를 이용하였다. Face-bounding-box는 얼굴 외곽선으로 예측되는 지점을 따라 직사각형 모양의 구역을 표시한 것이다. 이 직사각형의 넓이를 계산하여 얼굴 면적을 구할 수 있다. 만약 영상 안에 얼굴이 여러 개 존재한다면, Face-bounding-box는 얼굴 개수만큼 표현되는데 이 중 가장 큰 값을 대푯값으로 설정하여 측정하였다.

또한, [6]에서는 얼굴이라고 판단한 것들 중에 실제 얼굴일 확률을 Face-detection-score로 정의한다. 본 연구에서는 1초에 10번 얼굴 비율을 측정하였고, Face-detection-score가 70 이상인 경우에 실제 얼굴이라고 판단하였다.

〈Table 2〉 Comparison of face proportions in the videos

Video 1	Video 2	Video 3	Video 4	Video 5
				

## 4.3 얼굴 비율 비교를 통한 영상 유사도 측정

동영상은 연속된 이미지들의 모음인데, 각각의 이미지 하나를  $F$ (Frame)라고 정의한다. 두 개의 영상 ( $V_1, V_2$ )이 있을 때, 각각의 영상은  $F$ 의 집합으로 (1)과 같이 표현한다.  $n, m$ 은 각각의 영상에 존재하는  $F$ 의 개수이다.

$$V_1 = \{F_{1,1}, F_{1,2}, \dots, F_{1,n}\}, \quad V_2 = \{F_{2,1}, F_{2,2}, \dots, F_{2,m}\} \quad (n, m = \text{total number of frames in the video}) \quad (1)$$

$F$  안의 얼굴 비율을 4.2에서 기술한 것과 같이 측정하여 그 값을  $P(F)$ 로 정의한다.

$$\text{if } P(F_{2,k}) \geq P(F_{1,k}), \quad \text{Similarity}(F_{1,k}, F_{2,k})(\%) = \frac{P(F_{1,k}) + c}{P(F_{2,k}) + c} \times 100 \quad (2)$$

두 영상의  $k$ 번째 프레임( $F_{1,k}, F_{2,k}$ ) 간의 얼굴 비율 비교를 통해 유사도(Similarity)를 측정하는 식은 (2)와



같이 계산하였다.  $P(F_{2,k}) \geq P(F_{1,k})$  일 때, 두 이미지 간의 얼굴 비율의 비는  $\frac{P(F_{1,k})}{P(F_{2,k})}$ 로 구할 수 있으나,  $P(F_{2,k})$ 가 0에 수렴할수록 *Similarity*가 급격하게 높아지기 때문에 보정 값( $c$ )을 추가하였다.

if)  $n = m$ , (3)

$$Similarity(V_1, V_2)(\%) = \frac{\sum_{k=1}^n Similarity(F_{1,k}, F_{2,k})}{n}$$

if)  $n \neq m$ , (4)

$$Similarity(V_1, V_2)(\%) = \max_{t=0,1,\dots,m-n} \frac{\sum_{k=1}^n Similarity(F_{1,k}, F_{2,k+t})}{n} \quad (n < m)$$

두 영상의 길이가 같다면, 영상의 유사도는 프레임 간의 모든 유사도의 평균을 구하여 (3)과 같이 측정할 수 있다. 하지만 본 연구에서는 서로 다른 영상 간의 유사도를 측정해야 하므로, 비교 대상인 두 영상의 길이가 다를 가능성이 매우 높다. 따라서 (4)와 같이 구간별로 측정하여 측정 값 중 최댓값을 유사도로 계산한다. 이러한 유사도 측정 방법은 두 영상 중 짧은 동영상에 긴 동영상에 포함되어 있는지 판단할 수 있는 지표가 된다.

## V. 실험

영상의 연속된 프레임마다 시행하는 얼굴 탐지는 Wider-face dataset [13]으로 사전 훈련된 Resnet50을 백본 네트워크로 구성하였다. Wider-face dataset에는 32,203개의 이미지와 393,703개의 Face-bounding-box가 존재한다. 또한, 일반적인 얼굴뿐만 아니라, 다양한 표정, 자세, 움직임을 포함한 이미지를 가지고 있다. 데이터셋의 훈련, 검증, 테스트 구성 비율은 40%, 10%, 50%이고, 용량은 4GB이다.

코드는 python을 사용하여 개발하였고, RetinaFace를 기반으로 Resnet50의 detector로 재구성한 코드 [14]를 영상에 기록이 가능하도록 수정하였다. 또한, Face-bounding-box를 이용한 얼굴 비율 측정과 영상 저작물과의 얼굴 비율 유사도 측정 코드를 덧붙여 작성하였다. 실험에 사용한 PC의 환경은 [표 3]과 같다.

〈Table 3〉 Experiment environment



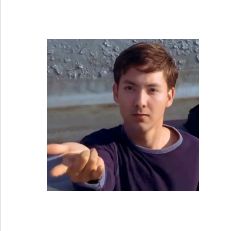

	Information
CPU	Intel(R) Core(TM) i9-9900KF (16 core, 3.60GHz)
Memory	32GB
GPU	NVIDIA GeForce RTX 3060Ti (8GB)
Motherboard	ASUS TUF Z390-PLUS GAMING
OS	Windows 10 Home 64 bits(10.0)

### 5.1 영상 데이터 셋

딥페이크 영상을 쉽게 제작하고 공유하는 안드로이드 앱 [15]을 사용하여 영상저작물을 생성하였다. 해당 앱은 1억명 이상의 사용자가 다운로드 받은 가장 대표적인 딥페이크 앱이다. 'Movie'라는 키워드로 검색하여 상위 검색 결과 중 5~20초 이내의 영상 25개를 사용하였다.

영상저작물 25개를 각각 Reface 앱을 활용하여 딥페이크 영상 25개를 제작하였다. 또한, 영상저작물을 무단으로 딥페이크 영상으로 제작하고 배포하는 과정에서, 영상의 특징이 바뀔 수 있으므로 3가지 변경을 추가로 시행하였다. 영상저작물에 각각 색상, 화질, 크기를 변경하였고 변경한 영상의 예시는 [표 4]와 같다. 색상은 흑백으로 변경하였고, 화질은 영상 압축을 20~50% 시행하였다. 마지막으로, 크기는 원본 영상의 70~90%의 해상도로 변경하였다. 요약하면, 영상저작물에 다른 얼굴을 합성한 영상 25개와 추가로 영상의 특징을 변경한 영상 75개를 영상저작물을 활용한 딥페이크 영상의 데이터셋으로 사용하였다.

〈Table 4〉 Data sets of deepfake videos using video productions

Face swap	Face swap + Gray	Face swap + Quality	Face swap + Size
			

영상저작물이 아닌 딥페이크 영상은 두 가지 방법으로 수집하였다. 총 영상 데이터셋은 1000개로 결과는 [표 5]와 같다.

〈Table 5〉 Data sets of deepfake videos not using video productions

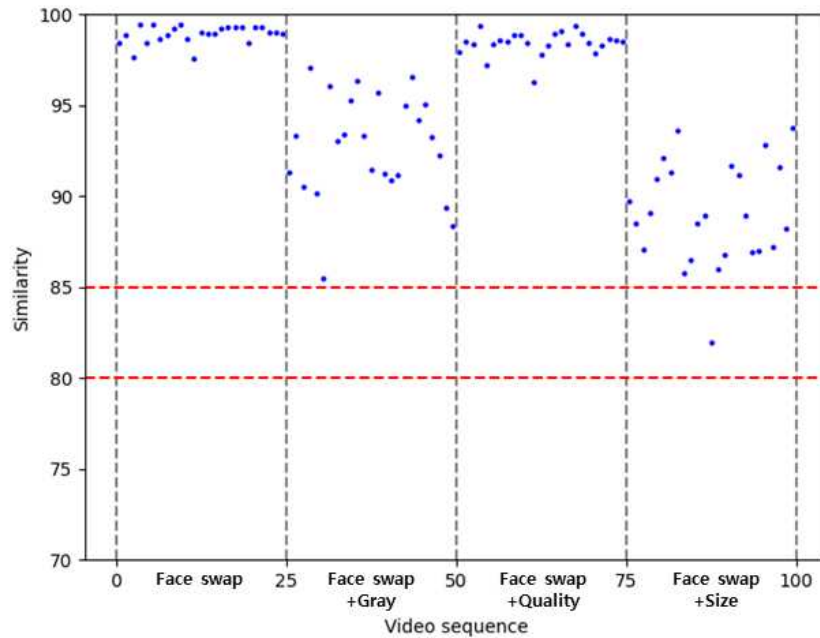
Total	Meta AI	FaceForensics++				
	Deepfake Detection Challenge	Deepfakes	Face2Face	FaceShifter	FaceSwap	Neural Textures
1000	500	100	100	100	100	100

첫 번째로, Meta AI에서 딥페이크 기술을 악용하는 것을 막기 위해 딥페이크 식별 챌린지 [16]를 개최하였는데, 이때 공개한 영상 샘플 데이터 셋 [17] 500개를 사용하였다.

또한, 딥페이크 식별 방법의 평가를 표준화하기 위한 연구 [18]가 진행되었다. 딥페이크 영상 압축률, 크기를 랜덤으로 생성한 데이터 셋을 공개하였는데, 이 중 Deepfakes [19], Face2Face [20], FaceShifter [21], FaceSwap [22], Neural Textures [23] 모델로 생성된 각각 100개, 총 500개의 영상을 추가로 수집하였다.

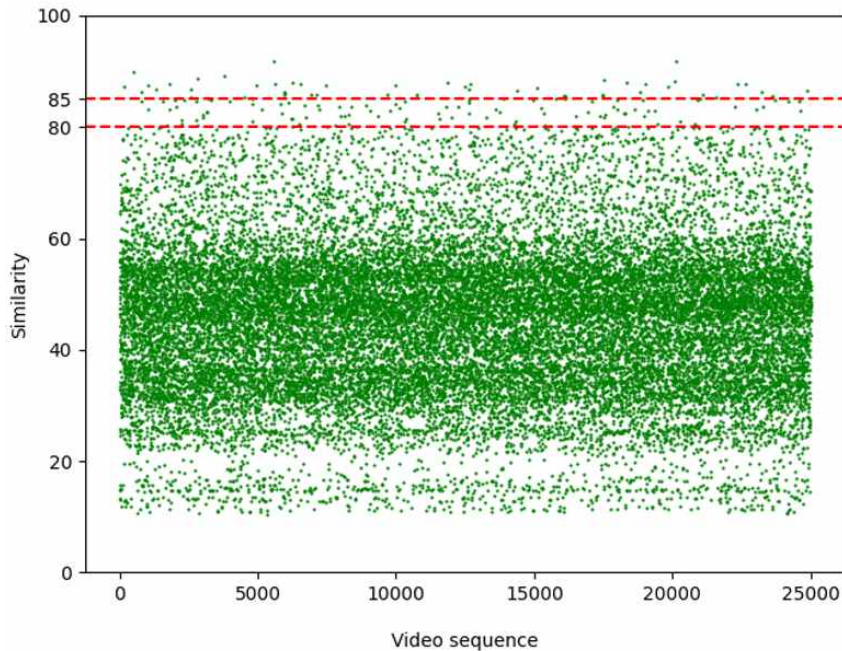
## 5.2 유사도 측정 및 저작권 침해 여부 탐지

영상저작물과 이를 딥페이크 영상으로 변형한 영상 간의 유사도를 측정한 결과는 [그림 3]과 같다. 유사도 측정식에 대한 보정 값은(c) 1.0으로 설정하여 계산하였다. 영상저작물의 얼굴만 다른 얼굴로 바꿀 경우, 25개의 영상 모두 유사도가 95% 이상으로 높게 측정되었다. 얼굴과 화질을 변경한 영상 역시, 유사도가 소폭 낮아지긴 했지만 여전히 95% 이상이었다. 흑백으로 추가 변경한 영상은 얼굴 비율 측정을 하기 위해 얼굴 범위를 인식하는데 있어, 소수 프레임에서 얼굴을 탐지하지 못하는 경우가 발생하여 유사도가 조금 낮아졌지만, 대부분의 영상에서 유사도가 90% 이상으로 측정되었다. 마지막으로, 얼굴을 다른 얼굴로 조작하고 프레임 크기를 변경한 영상에서는 전체 화면에서의 얼굴 비율이 전체적으로 높아짐에 따라 그만큼 유사도가 낮게 측정되었다.



〈Figure 3〉 Similarity between video productions and deepfake videos using them

영상저작물과 관련 없는 딥페이크 영상의 유사도를 측정한 결과는 [그림 4]와 같다. 25개의 영상 저작물과 1,000개의 딥페이크 영상의 얼굴 비율을 측정하고 각각 비교하여 25,000개의 유사도를 측정하였다. 25,000번 유사도를 10회 측정했을 때, 평균 74.83초가 소요되었다.



〈Figure 4〉 Similarity between video productions and deepfake videos not using them

영상저작물과 관련 없는 영상 25,000개 중 24,844개의 영상, 99.38%가 영상저작물과의 유사도가 80% 보다 낮게 측정되어 영상저작물을 딥페이크 영상으로 재제작한 것들과 차이를 보였다. 이를 통해 영상저작물을 활용하여 딥페이크 영상을 제작하였는지 여부를 유사도에 따라 적절한 기준으로 분류하여 탐지할 수 있음을 알 수 있다.



〈Table 6〉 Classification evaluation in detection of copyright infringement

구분	Accuracy(%)	Recall(%)	FPR(%)
80	95.94	100.0	0.62
85	96.32	99.0	0.23

얼굴비율 유사도를 측정하고 적절한 기준에 따라 분류하여 영상저작물 활용 여부를 탐지하는 탐지기의 성능은 [표 6]과 같다. 유사도를 80, 85의 기준으로 분류할 때 각각 95.94%, 96.32%의 높은 성능으로 실제 영상저작물 활용 여부를 옳게 반환하였다. 또한, 영상저작물을 활용한 딥페이크 영상은 두 기준 모두 99% 이상의 확률로 탐지기에서 영상저작물이라고 분류하였다. 영상저작물과 관련 없는 영상 중 탐지기가 영상저작물이라고 잘못 판단할 경우는 유사도 80으로 분류할 때 0.62%, 85로 분류할 때는 0.23%로 오탐률이 매우 작은 것을 알 수 있다. 이러한 성능 평가 결과를 보면 알 수 있듯이, 화면 내 얼굴 비율 비교를 통한 분류 방식은 실제 영상저작물을 이용하여 딥페이크 영상을 생성했는지의 여부를 높은 신뢰도로 탐지할 수 있다.

## VI. 결 론

지난 5년간 딥페이크 기술을 악용한 음란물 배포, 정치적 선동, 가짜뉴스 유포, 명예훼손 등의 사례가 다수 발생하였다. 최근에는 영상저작물을 저작권자의 동의 없이 무단으로 활용하여 딥페이크 영상을 제작하고, 이를 유튜브, 인스타그램, 틱톡, 개인 방송 등에 게시하는 저작권 침해 사례가 보고되었다. 저작권자가 자신의 영상저작물이 무단으로 게시되는 것을 막기 위해 일일이 모니터링하고 삭제를 요청하는 것은 무리가 있다. 따라서 본 연구에서는 딥페이크 영상으로 재생산되는 것을 금지하고 싶은 영상저작물을 저작권자가 시스템에 등록하면, 이후부터는 그것을 활용한 딥페이크 영상인지의 여부를 자동으로 탐지하는 모델을 제안하였다.

탐지를 위해 영상저작물과 딥페이크 영상을 프레임 단위로 구분하여 얼굴 비율을 측정하고, 이를 비교하여 유사도를 측정하였다. 이러한 방법은 2장에서 살펴본 것과 같이, 기존의 이미지 분류 모델에서 생성되는 피처 벡터를 통한 영상 유사도 측정 방식에 비해 비교 연산의 수가 현저히 적어 유사도 측정 속도가 매우 빠르다.

실험을 통해 영상저작물을 활용한 딥페이크 영상은 얼굴, 색상, 화질, 크기가 변하더라도 얼굴 비율이 영상저작물과 유사하게 변화함을 알 수 있었다. 따라서 영상저작물과 딥페이크 영상의 얼굴 비율 유사도를 프레임 별로 구하고, 그 값을 이용하여 영상 간 얼굴 비율 유사도를 계산하였다. 이 계산 값에 따라 적절한 기준으로 분류하여, 영상저작물을 무단으로 악용하여 딥페이크 영상을 제작하였는지의 여부를 높은 확률로 판단할 수 있었다. 다만 영상저작물을 딥페이크 영상으로 재제작하는 과정에서 해상도를 크게 변경하는 경우, 실험의 결과보다 유사도가 낮게 측정되어 탐지 정확도가 낮아질 수 있다. 실험에서는 가장 많이 사용되는 16:9, 4:3, 1:1의 해상도로 서로 변경된다는 가정으로 영상 데이터셋을 생성하였지만, 영상의 화면을 상당 부분 잘라내는 정도로 해상도를 변경한다면 제안한 화면 내 얼굴 비율 비교 방법이 효과적이지 못하다. 이를 극복하기 위해, 배경을 활용하여 피처를 추가로 생성하거나, 이미지 분류모델을 활용하여 얻어진 피처 벡터들을 해싱(hashing)하여 생성된 값을 추가로 활용하는 방법이 필요할 것으로 판단된다. 향후 연구에서는 이러한 방법들을 추가로 활용하여, 각각의 방법들을 적용한 단계적 탐지 모델을 통해 영상의 색상, 화질, 크기, 왜곡 등의 변수에 조금 더 강인하도록 발전시키려 한다.

본 연구에서 제안한 방법으로 영상플랫폼에 무분별하게 개제되고 있는 딥페이크 영상의 저작권 침해 여부를 탐지하여, 딥페이크 기술의 올바른 사용에 기여할 수 있기를 기대한다.

## 참 고 문 헌 (References)

- [1] Chan-Sol Kim, "Deepfakes and Copyright Inringement," Korea Copyright Protection Agency, pp.1-6, 2021.
- [2] Krizhevsky A, Sutskever I, and Hinton G-E, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, 25, 2012.
- [3] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, and Rabinovich A, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.1-9, 2015.
- [4] Simonyan K, and Zisserman A, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.
- [5] He K, Zhang X, Ren S, and Sun J, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.770-778, 2016.
- [6] Deng J, Guo J, Zhou Y, Yu J, Kotsia I, and Zafeiriou S, "Retinaface: Single-shot multi-level face localisation in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.5203-5212, 2020.
- [7] Tran D, Bourdev L, Fergus R, Torresani L, and Paluri M, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, pp.4489-4497, 2015.
- [8] Lin T-Y, Dollár P, Girshick R, He K, Hariharan B, and Belongie S, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.2117-2125, 2017.
- [9] Adelson E-H, Anderson C-H, Bergen J-R, Burt P-J, and Ogden J-M, "Pyramid methods in image processing," *RCA engineer*, 29(6), pp.33-41, 1984.
- [10] He K, Zhang X, Ren S, and Sun J, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, 37(9), pp.1904-1916, 2015.
- [11] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, and Berg A-C, "Ssd: Single shot multibox detector," in *European conference on computer vision*, pp.21-37, 2016.
- [12] Zhang S, Zhu R, Wang X, Shi H, Fu T, Wang S, Mei T, Li S-Z, "Improved selective refinement network for face detection," arXiv preprint arXiv:1901.06651, 2019.
- [13] Yang S, Luo P, Loy C-C, and Tang X, "Wider face: A face detection benchmark," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.5525-5533, 2016.
- [14] Pytorch\_Retinaface github, [https://github.com/biubug6/Pytorch\\_Retinaface](https://github.com/biubug6/Pytorch_Retinaface), Accessed February, 2022.
- [15] Reface: Swap your faces now, <https://reface.app>, Accessed February, 2022.
- [16] Deepfake Detection Challenge, <https://www.kaggle.com/c/deepfake-detection-challenge>, Accessed February, 2022.
- [17] Dolhansky B, Bitton, J, Pflaum, B, Lu J, Howes R, Wang M, and Ferrer C-C, "The deepfake detection challenge (dfdc) dataset," arXiv preprint arXiv:2006.07397, 2020.
- [18] Rossler A, Cozzolino D, Verdoliva L, Riess C, Thies J, and Nießner M, "Faceforensics++: Learning to detect manipulated facial images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp.1-11, 2019.
- [19] Deepfakes github, <https://github.com/deepfakes/faceswap>, Accessed February, 2022.
- [20] Thies J, Zollhofer M, Stamminger M, Theobalt C, and Nießner M, "Face2face:

- Real-time face capture and reenactment of rgb videos,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp.2387-2395, 2016.
- [21] Li L, Bao J, Yang H, Chen D, and Wen F, “Faceshifter: Towards high fidelity and occlusion aware face swapping,” arXiv preprint arXiv:1912.13457, 2019.
- [22] Faceswap github, <https://github.com/MareKowalski/FaceSwap/>, Accessed February, 2022.
- [23] Thies J, Zollhöfer M, and Nießner M, “Deferred neural rendering: Image synthesis using neural textures,” *ACM Transactions on Graphics (TOG)*, 38.4, pp.1-12, 2019.

## 저 자 소 개



**김 동 엽 (Dongyeob Kim)**

준회원

2019년 2월 : 고려대학교 정보보호학부 졸업

2020년 9월 ~ 현재 : 고려대학교 정보보호학과 석사과정

관심분야 : 디지털 포렌식, 정보보호 등



**이 상 진 (Sangjin Lee)**

평생회원

1989년 10월 ~ 1999년 2월 : 한국 전자통신연구원 선임연구원

1999년 3월 ~ 2001년 8월 : 고려대학교 자연과학대학 조교수

2001년 9월 ~ 현재 : 고려대학교 정보보호대학원 교수

2017년 3월 ~ 현재 : 고려대학교 정보보호대학원 원장

관심분야 : 대칭키 암호, 정보은닉 이론, 디지털 포렌식