# 비디오의 25개의 프레임들을 이용한 시계열 학습 딥페이크 비디오 탐지 연구

## Deepfake video detection via time-series training based on 25 Frames of Video

황영준, 김영섭(단국대학교 전자전기공학부)

## INTRODUCTION

최근 딥페이크 미디어의 데이터양은 기하급수적으로 늘어나고 있으며, 공개된 오픈소스를 통해 대중들도 손쉽게 딥페이크 미디어를 제작할 수 있다. 딥페이크 생성모델의 지속적인 발전으로 딥페이크 미디어는 사람의 눈으로 식별하기 어려울 정도로 실제와 닮아가고 있다. 이러한 딥페이크 미디어는 단순히 유희를 위한 수단으로 사용될 뿐만 아니라, 사회, 경제적으로 큰 문제를 야기하여 딥페이크 비디오 탐지모델에 대한 수요가 증가하고 있다.

본 논문에서 제안하는 시계열 학습 딥페이크 탐지모델은 MTCNN(Multi-task cascaded CNN)을 통해 입력 비디오에서 사람의 얼굴 영역을 탐지하고, 각 프레임별로 추출한다. 추출된 얼굴 영역을 얼굴 인식 및 군집화를 위한 FaceNet에 입력하여 512차원의 임베딩으로 변환한다. 변환된 임베딩을 프레임 방향으로 쌓아 (25,512) 시계열 정보가 포함된 임베딩을 형성한 후 LSTM-FC 레이어를 거쳐 학습을 진행한다. 모델 학습을 위해 최신의 딥페이크 조작 기법을 이용한 Celeb-DF 데이터셋을 이용한다.

## DEEPFAKE VIDEO DETECTION MODEL

### 1. Framework

본 논문에서 제안하는 모델의 전체적인 흐름은 그림 2의 Framework와 같다. 입력된 비디오를 MTCNN-FaceNet-LSTM-FC 순서로 처리하고, FC Layer의 끝단에서 활성화 함수 SoftMax를 거쳐 진실 혹은 거짓일 확률을 출력한다.

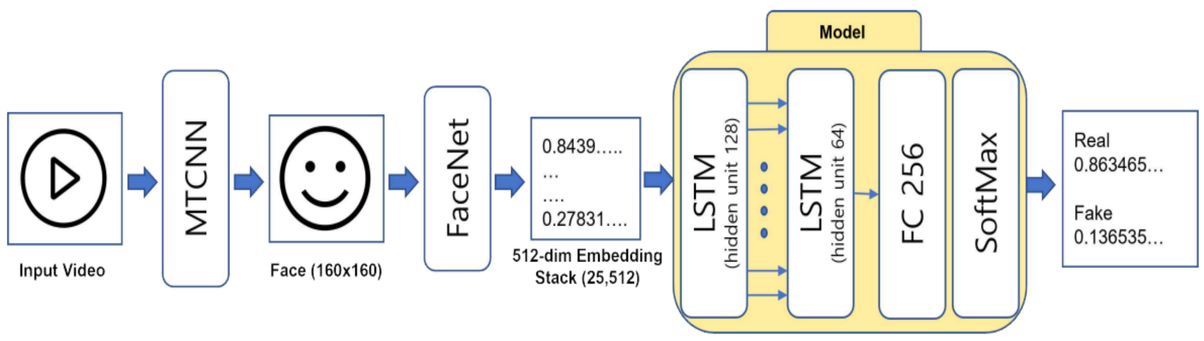


Figure 1. Framework

- **MTCNN**

딥페이크 비디오는 얼굴 영역에 대해 조작을 한다. 따라서, 얼굴 영역을 추출하여 네트워크에 입력해야 한다. 본 논문은 Celeb-DF 데이터셋을 이용하여 다음의 과정을 통해 전처리하였다. 비디오의 각 프레임을 이미지로 변환하여 저장한다. 저장된 이미지에서 딥페이크 탐지에 불필요한 영역은 버리고 얼굴 영역만 추출한다. 본 과정에서 MTCNN을 통해 얼굴 영역을 추출한다. MTCNN은 다양한 해상도의 이미지에서 정확한 얼굴 추출 수행 능력을 보여준다. 본 MTCNN은 3단계로 구성되며, 각 단계에서 이진 분류, 얼굴의 Bounding box(Bbox), 얼굴 대표적인 눈, 코와 입 등의 랜드마크의 위치를 출력한다. 각 단계의 출력에 대해 NMS(Non-Maximum Suppression) 및 Bbox의 위치에 대한 회귀를 진행하여 최종 얼굴 영역에 대한 Bbox를 출력한다. 단순히 얼굴 영역의 Bbox만 추측하는 것보다 얼굴의 대표적인 랜드마크와 같이 추측함으로써 얼굴 검출 정확도를 높였다.

- **Facenet**

얼굴 영역의 (160,160,3) 이미지를 CNN 기반 네트워크를 거쳐 특징이 추출된 512차원의 임베딩으로 변환한다. 변환된 임베딩은 비디오의 한 프레임에 대한 임베딩이므로, 한 비디오의 25개의 프레임 이미지를 Concatenate하여 (25, 512)의 시계열 정보가 포함된 임베딩을 형성하였다.

- **LSTM-FC Layer**

Long short term memory(LSTM)는 기존 Recurrent Neural Network(RNN)의 장기 의존성 문제를 네트워크 내부의 cell state 및 이전 셀의 정보에 얼마나 의존할지를 조정하는 게이트를 두어 해결하였다. 본 논문에서 제안하는 모델은 (25, 512)의 임베딩을 LSTM에 입력하고, Fully-connected Layer를 거쳐 학습을 진행하였다. 하나의 LSTM Layer를 통해 학습용 데이터를 학습한 결과 Training Loss 및 Validation Loss가 줄어들지 않아 모델이 임베딩을 잘 학습하지 못함을 확인하였다. 따라서, LSTM을 3-Layer로 쌓는 구조를 고안하였다. 처음 두 개의 LSTM Layer는 각 프레임의 hidden state를 출력하며, 마지막 LSTM Layer를 통해 하나의 hidden state를 출력한다.

출력된 hidden state를 Fully Connected(FC) Layer에 입력하고, 활성화 함수로 SoftMax를 적용하여 입력된 비디오가 진실 혹은 거짓일 확률을 출력한다.

### 2. Dataset

Celeb-DF Dataset을 이용하였다. 총 1720개의 비디오를 1376:344로 분할하여 Training/Validation Dataset을 구축하고 학습을 진행하였다. Test Dataset으로는 Celeb-DF에 주어진 Test Video 리스트를 통해 작업하였으며, 총 300개의 비디오를 이용하였다.

## TRAINING AND RESULTS

최적의 학습 정확도 및 학습 속도를 얻기 위해 Hyperparameter를 정의하였다. Hyperparameter로 batch_size, epochs, learning_rate를 설정하였다. 학습을 위한 Optimizer로 Nadam optimizer를 사용하였으며, 초기 학습 과정에서 일정 epoch 이상에서 학습이 진행되지 않아 learning_rate decay를 1e-6으로 적용하였다. 또한, 학습 중간 과정에서 Training data에 과적합(Overfitting) 하는 경향을 보여 LSTM Layer 사이에 Dropout을 0.5의 비율로 적용하였다. 데이터의 Label은 One-hot encoding을 통해 (N, 2)의 배열로 변환한 후 categorical_crossentropy loss function을 적용하였다.

최종 학습은 batch_size=32, epochs=50, learning _rate=8e-4를 이용하여 학습하였으며, 학습 과정의 Loss는 그림 3과 같다. 최종 Loss 및 Accuracy는 Table 1과 같다.

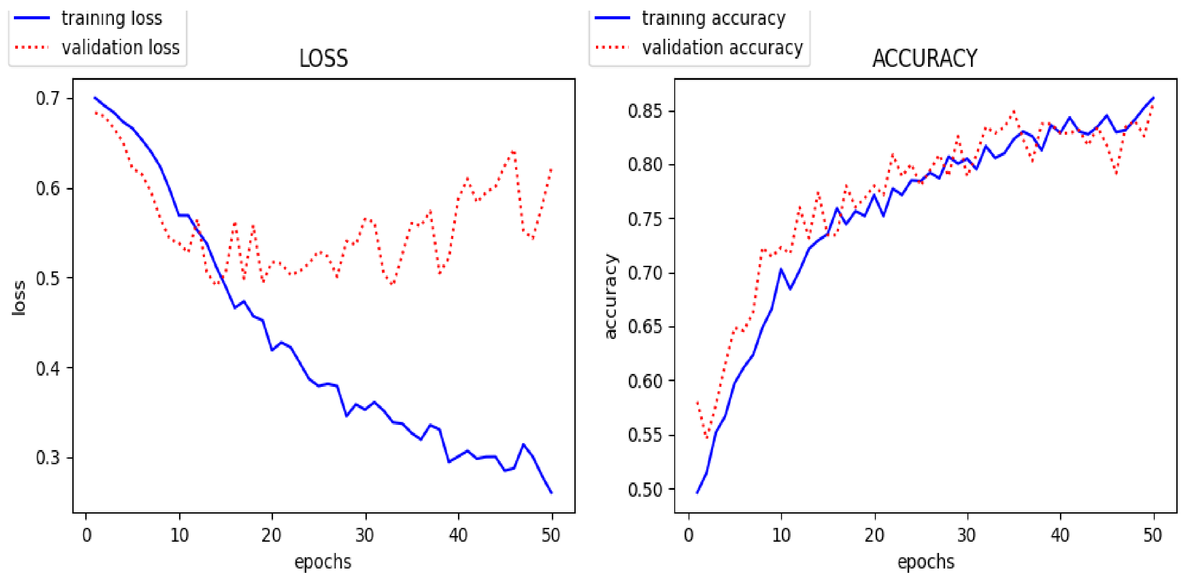


Figure 2. Train/Validation Loss and Accuracy

| | Training | Validation | Test |
|---|---|---|---|
| Loss | 0.2606 | 0.6208 | |
| Accuracy | 0.8611 | 0.8571 | 0.8562 |

Table 1. Loss and Accuracy

## CONCLUSION

본 모델은 FaceNet을 통해 추출한 각 얼굴 영역 이미지의 임베딩을 군집화(Clustering)하기 위한 Triplet Loss를 적용한 모델에서 아이디어를 얻어 실험을 진행하였다. 기존 모델은 한 프레임 단위의 임베딩에 대해 진위 판별 학습을 진행하며, 추론(Inference) 과정에 사용될 프레임 수를 정하여 각 프레임의 확률 평균값을 이용하여 추론을 진행하였다. 그러나, 위와 같은 Framework는 비디오의 시계열 정보를 학습할 수 없다. 따라서 LSTM을 사용한 시계열 정보 학습 모델을 구축하였다.

본 모델은 입력된 이미지를 512차원의 임베딩으로 변환하며 Network의 Parameter를 감소할 수 있었다. 또한, 각 프레임에서 연속 혹은 불연속적인 정보를 LSTM을 이용하여 학습한 후 확률을 출력하여 딥페이크 진위를 판별할 수 있다.

기존 모델의 FaceNet은 사람의 얼굴을 일정 차원의 임베딩으로 변환한 후 Triplet Loss를 적용하여 유클리드 공간상에서 서로 다른 Label의 임베딩을 일정 Margin 이상의 거리를 두도록 군집화한다. 그러나, 본 모델은 FaceNet에서 각 얼굴의 특징점을 추출하여 임베딩으로 변환한 후 군집화를 적용하지 않고 LSTM-FC Layer에 입력하여 유클리드 공간상에서 분류되지 않은 임베딩의 시계열 정보를 학습하였다. 따라서 정제되지 않은 임베딩을 시계열 정보만을 통해 딥페이크 탐지를 시도하여 기존 모델보다 낮은 성능을 보였다. 본 모델은 기존 모델이 인지하지 못하였던 시계열 정보를 활용하여 딥페이크 탐지를 시도하였으며, 얼굴 영역의 시각적 특징에 대한 학습 과정이 없었음에도 준수한 성능을 보였다.

또한, 아래 표 2를 참고하면 Celeb-DF Dataset을 이용한 기존 딥페이크 검출 방법과 비교하여 준수한 성능을 보이는 것을 확인하였다. Xception 및 Multi-task는 Base-Line Model로, Convolutional Neural Network 기반의 단일 Xception Network 및 Capsule Network를 적용한 모델이다.

| | Accuracy[%] |
|---|---|
| Ours | 85.6 |
| Xception | 65.5 |
| Capsule | 57.5 |

Table 2. Comparison with Base-model