UNIVERSITÀ
DEGLI STUDI
FIRENZE

IMT SCHOOL
FOR ADVANCED
STUDIES
LUCCA

University of Florence | IMT School for Advanced Studies Lucca

II Level Master | Data Science & Statistical Learning

Master Thesis

DeepFake Detection Exploiting Self-Attention Maps

Supervisor

Prof. Lorenzo Seidenari

Co-Supervisor

Luca Cultrera PhD

Author

Gianmarco Santoro

Academic Year 2022-2023

SUMMARY

Acronyms & Abbreviations

- AI | *Artificial Intelligence*
- CV | *Computer Vision*
- DL | *Deep Learning*
- ML | *Machine Learning*
- NN | *Neural Networks*
- CNNs | *Convolutional Neural Networks*
- RNN | *Recurrent Neural Network*
- NLP | *Natural Language Processing*
- GAN | *Generative Adversarial Network*
- MLP | *Multi-Layer Perceptron*
- GPT | *Generative Pre-trained Transformer*
- ROC | *Receiver Operating Characteristic*
- AUC | *Area Under the Curve*
- ViT | *Vision Transformer*
- OOD | *Out-of-Distribution*
- CAE | *Convolutional Auto-Encoder*
- GIT | *Global Information Tracker*
- MSE | *Mean Squared Error*
- CLS | *Classification*

Abstract

In “*Leveraging Visual Attention for out-of-Distribution Detection*" [1] a method for learning to recognize samples from classes, which are not used in training sessions, has been developed exploiting ViT self attention maps.

An autoencoder is trained on attention heatmaps and the reconstruction error is used to flag samples as In-Distribution or Out-of-Distribution thus providing an estimate of the uncertainty of the classifier prediction.

Image forgery detectors are often trained on a specific technique and are not robust in the cross-forgery scenario, in this thesis an experimental DeepFake detection as an Anomaly Detection task has been performed.

It is proposed to exploit a face identification module "*Face transformer for recognition*" [2] or a face attribute estimation module "*MiVOLO: Multi-input Transformer for Age and Gender Estimation*" [3] in a similar fashion to "*Leveraging Visual Attention for out-of-Distribution Detection*" [1] to predict the presence of face forgery. The Autoencoder should be trained on heatmaps of REAL samples and then tested on REAL/FAKE ones.

A focus of the research is on extracting the feature "self-attention maps," specific heatmaps of the processed images. This information is then used as a feature to perform "Out of Distribution Detection," a discrimination process, in this case, distinguishing between real images and those generated by a "Generative Artificial Intelligence."

1. Introduction, State of the Art & Objective

1.1. Artificial intelligence

Artificial Intelligence is a current fast-growing field in computer science, statistics and engineering, enabling computers to learn from data and perform tasks that typically require human cognitive abilities. AI has evolved into a multidisciplinary domain, encompassing machine learning, natural language processing, computer vision, and more.

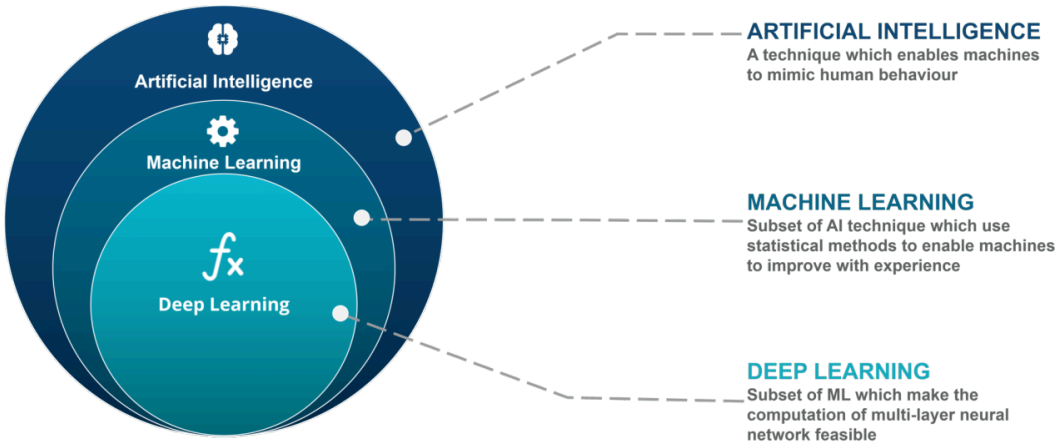

Figure 1 – Deep Learning vs Machine Learning vs Artificial Intelligence relationship. [7]

AI is leading to transformative changes across various industries:

- In healthcare: AI algorithms can help to analyze medical images for early disease detection and assist in personalized treatment plans.
- In transportation: AI can power autonomous vehicles, optimizing routes, and lead safety.

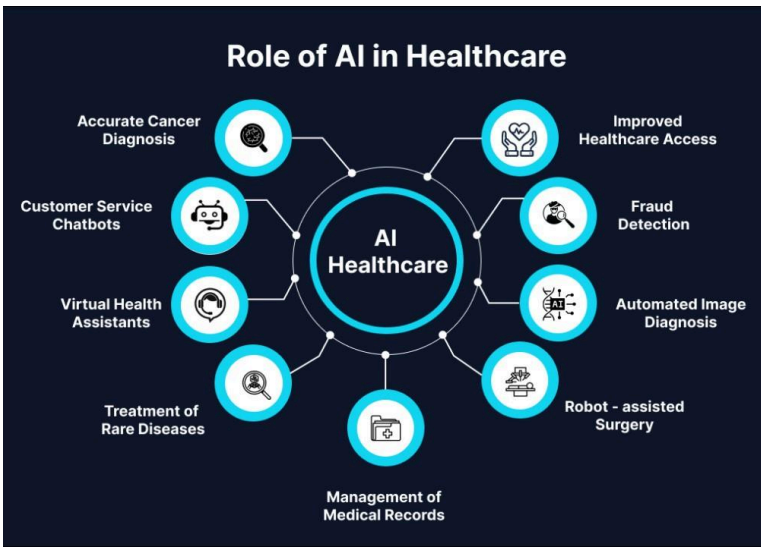

Figure 2 – AI in Healthcare. [8]

1.2. Deep Learning

Deep Learning stands out as one of the most remarkable advancements in AI. DL models are inspired by the structure and function of the human brain's neural networks, called Artificial Neural Networks, consisting of multiple layers of interconnected nodes called neurons. Each layer processes and transforms the input data, learning complex representations as information flows through the network.

The depth, number of hidden layers, of neural networks allows them to learn patterns and features in vast amounts of data. Unlike traditional machine learning algorithms that require handcrafted features, deep learning models learn these features directly from the raw data. This capability has led to unprecedented successes in various fields, including Computer Vision, Natural Language Processing, and speech recognition.

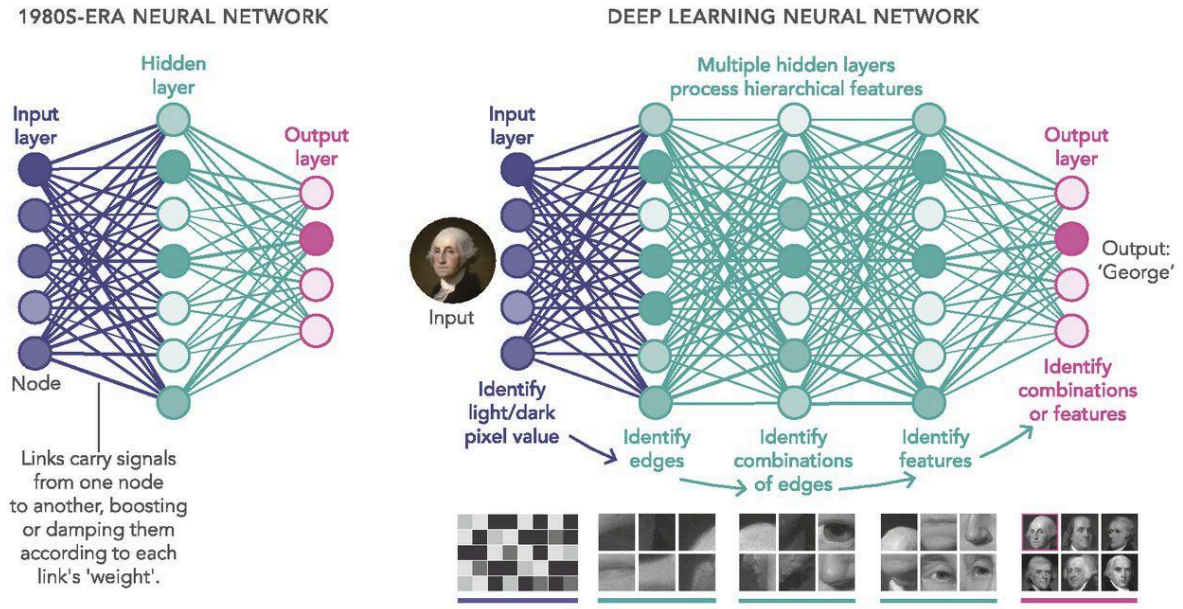

Figure 3 – Example of a Deep Learning Neural Network. [9]

1.3. Computer Vision

Computer Vision is the Artificial Intelligence sector for image processing, for example to recognize objects, scenes, faces, and gestures.

Some CV tasks are:

- Image classification: identify an object in an image.
- Detecting object location: locate and classify multiple objects within an image.
- The typical example is in autonomous vehicles, computer vision systems analyze live camera feeds to detect pedestrians, vehicles, and road signs, enabling safe navigation.

Convolutional Neural Networks (CNNs), has been a cornerstone of modern computer vision systems, able to capture features at different levels of abstraction, enabling the achievement mentioned above.

Figure 4 – Example of a frame from Tesla Autopilot vision system. [10]

1.4. DeepFake and its Detection

DeepFake enables the creation of highly realistic yet entirely fabricated images, and so videos, relying on Generative Adversarial Networks (GANs), to generate media that mimic real human appearances and behaviors.

Figure 5 – This deepfake of comedian Alec Baldwin, morphed into former President Trump, shows how real the manipulated images are. (Source: TikTok). [11]

The implications of DeepFake technology are multifaceted:

- It opens up new possibilities for entertainment, e.g. special effects in movies.
- On the other hand, can help misinformation and propaganda, e.g. with fake videos of public figures saying things they never did, which can spread rapidly across social media platforms, potentially causing political confusion.

The societal impact, ethical considerations, and the responsibility that comes with these transformative technologies are the reasons of the importance of this kind of research.

Detecting DeepFake media presents a significant challenge due to their high realism. Researchers are continuously developing methods for detection, often involving the analysis of inconsistencies in facial features, discrepancies in audio-visual synchrony, and abnormal patterns in the data.

In this work, the aim is to detect DeepFake videos via Out-of-Distribution detection.

Out-of-Distribution (OOD) refers to instances where the input data significantly deviates from the distribution of the training data. When a machine learning model encounters OOD samples, it may struggle to make accurate predictions, leading to unreliable behavior.

For example, an image classification model trained on photos of cats and dogs may confidently classify a picture of a cat but fail when presented with an image of a lion, a species it has never seen before.

The Out-of-Distribution detection task is to identify instances where a machine learning model encounters OOD samples. The goal is to equip models with the capability to recognize when they are presented with data that significantly differs from their training distribution.

In this work, to perform OOD detection, a method concerning a convolutional-Auto-Encoder is used:

- Conv-AE is trained on in-distribution samples, in this case, real videos.
- Once trained, the conv-AE is fed by real videos and fake videos. What is expected is that during reconstruction, the MSE error calculated for fake videos is higher than the one computed for real videos.
- These differences can underline which videos are out-of-distribution and so, which are DeepFake forged videos.

1.5. Tools of this research

1.5.1. Architectures

In this work, some Deep Learning architectures have been used, in particular Transformer and AutoEncoder.

1.5.1.1. Transformers

The Transformers architecture was introduced in the paper "Attention is All You Need" [4] for sequence-to-sequence NLP tasks and represents a shift in the field of deep learning. The core of Transformers is the self-attention mechanism, a mechanism that allows the model to weigh the importance of different parts of the input when making predictions. Unlike traditional recurrent neural networks (RNNs) that process sequences sequentially, Transformers can process all input tokens simultaneously in parallel. This parallelization enables Transformers to capture long-range dependencies and contextual information more effectively.

In image classification, Transformers process image patches in parallel, leveraging self-attention to model spatial relationships and capture global context.

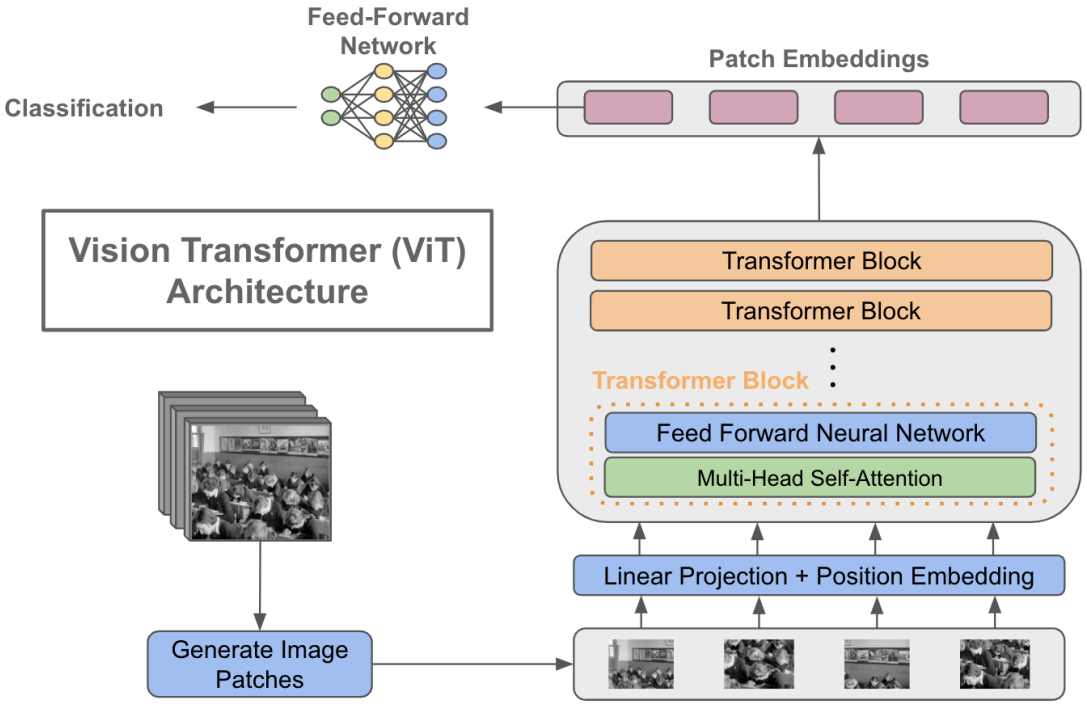

Figure 6 – A basic description of a vision transformer architecture. [12]

1.5.1.2. Auto-Encoders

Auto-Encoders are specifically neural network architectures designed for unsupervised learning and feature learning: the objective is to reconstruct its input data accurately.

It consists of two main components: an encoder network that compresses the input data into a lower-dimensional latent space representation, and a decoder network that attempts to reconstruct the original input from this compressed representation.

Convolutional Auto-Encoders (CAEs) leverage convolutional layers for both the encoder and decoder components. This design enables CAEs to learn hierarchical representations of images, capturing spatial hierarchies of features.

By training on a specific dataset, an auto-encoder can learn to extract meaningful features that best represent the input data. This learned representation can then be used for tasks such as image generation or denoising, anomaly detection, and data compression.

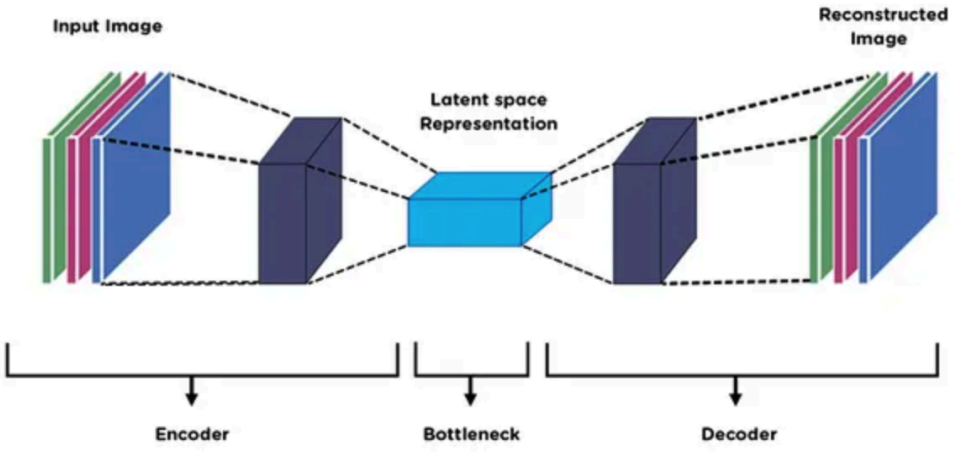

Figure 7 – Basic autoencoder architecture. [6]

1.5.2. Dataset

The data used in this research come first from *FaceForensics++* [4] dataset, which is a widely used benchmark dataset in the field of deepfake detection and face manipulation detection.

The dataset includes a variety of facial manipulations, videos are of high resolution, and includes different lighting conditions, poses, and facial expressions, mimicking real-world scenarios.

Figure 8-a – Image from FF++ dataset, in particular from video 523, crop 0052, forgery: ORIGINAL, not forged. [4]

Face modification methods[1], present in the dataset, are:

1. Deepfakes: create highly realistic fake videos where a person's likeness is convincingly superimposed onto another person's body.

Figure 8-b – Image from FF++ dataset, in particular from video 523_541, crop 0052, forgery: DeepFakes. [4]

2. Face2Face: a facial reenactment method that enables the transfer of facial expressions from a source actor to a target actor in real-time video.

[1] *Face modification methods are also called "forgeries".*

Figure 8-c – Image from FF++ dataset, in particular from video 523_541, crop 0052, forgery: Face2Face. [4]

3. FaceSwap: a technique that involves swapping the faces of two or more individuals in a video or image.

Figure 8-d – Image from FF++ dataset, in particular from video 523_541, crop 0052, forgery: FaceSwap. [4]

4. FaceShifter: another method for face swapping.

Figure 8-e – Image from FF++ dataset, in particular from video 523_541, crop 0052, forgery: FaceShifter. [4]

5. NeuralTextures: a method that focuses on generating highly realistic textures for face synthesis. It aims to improve the visual quality of synthetic faces, making them more difficult to distinguish from real ones. It is the most challenging forgery to be detected among these ones.

Figure 8-f – Image from FF++ dataset, in video 523_541, crop 0052, forgery: NeuralTextures[2]. [4]

Datasets like FaceForensics++ are used to train models to recognize the revealing signs of manipulation specific to each method.

Each face modification method leaves behind distinct artifacts or inconsistencies that can be exploited for detection. For example, deepfakes may exhibit unnatural eye movements or lack of blinking, while FaceSwap videos may show mismatches in skin tone or facial contours.

Researchers have developed deep learning models that leverage these artifacts to detect manipulated videos with high accuracy. These models often combine techniques such as analyzing facial landmarks, examining temporal inconsistencies, and assessing image quality metrics.

In this work, attention mechanism is used to focus on facial regions prone to manipulation, to detect forgeries in videos' frames.

FaceForensics++ dataset provides in this case a valuable resource for training and evaluating detection algorithms since it is composed of 1000 videos, in the version used in this research. From each video the first 100 frames have been extracted and cropped in as a rectangular frame considering the face of the person in the image. Thus, the total amount of images in the dataset is of about 600'000, made by: 1000 videos, 100 images each, and every image exists in original form (real) and manipulated by the 5 forgeries before-mentioned.

[2] *In this specific frame, the forgery worked very well, as typically does NeuralTextures, but it has created some fancy color between the lips.*

1.6. Objective

As said above, the research revolves around Computer Vision, specifically focusing on "DeepFake Detection". The goal is to verify the feasibility of developing a methodology that can, with a certain level of performance, recognize whether a video, or more simply an image, has been generated using a "DeepFake" AI system, where faces, expressions, and images in general are artificially produced.

The approach of this work is based on the methodology applied in "*Leveraging Visual Attention for out-of-Distribution Detection*" [1], where researchers have been able to recognize animals outside from the training set classes, as out-of-distribution classes. In a similar fashion, DeepFake videos made by different kinds of forgeries are the out-of-distribution classes for the detector-architecture trained exclusively on the real-video, which are the in-distribution in this case. It is a binary classification problem in the sense that the aim is to classify a video (images) as real or forged by an AI system.

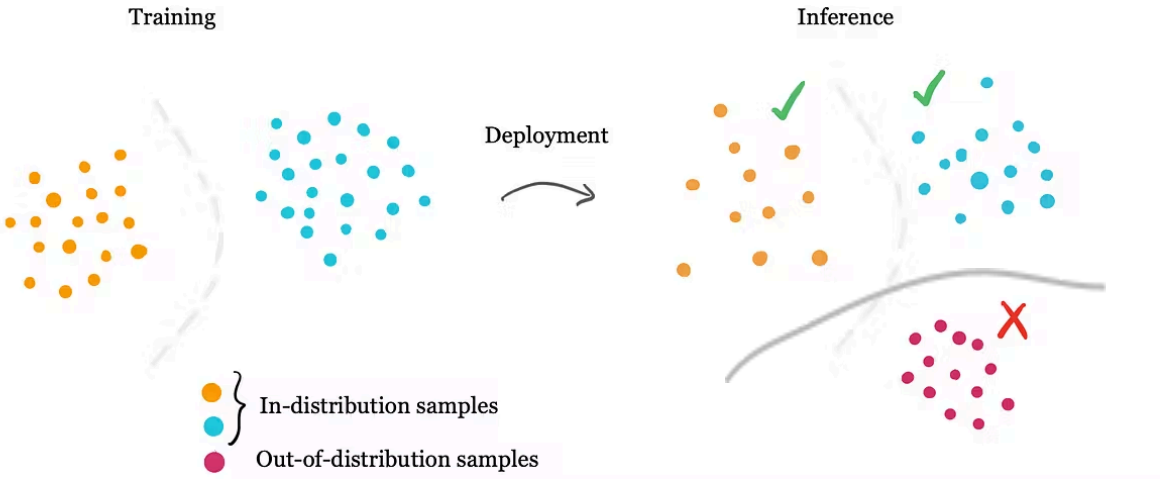

Figure 9 – Out-of-distribution detection in a general example; gray line is the decision boundary. [13]

1.7. State of the Art

The methodology considered has proved to be effective in "*Leveraging Visual Attention for out-of-Distribution Detection*" [1].

In this paper, a convolutional autoencoder is trained on attention heatmaps, produced by a ViT classifier, and the reconstruction error is used to flag samples as In-Distribution or Out-of-Distribution, thus providing an estimate of the uncertainty of the classifier prediction.

Moreover, the method does not require additional labels during training, ensuring efficiency and ease of implementation.

The architecture of this approach is shown in the following image.

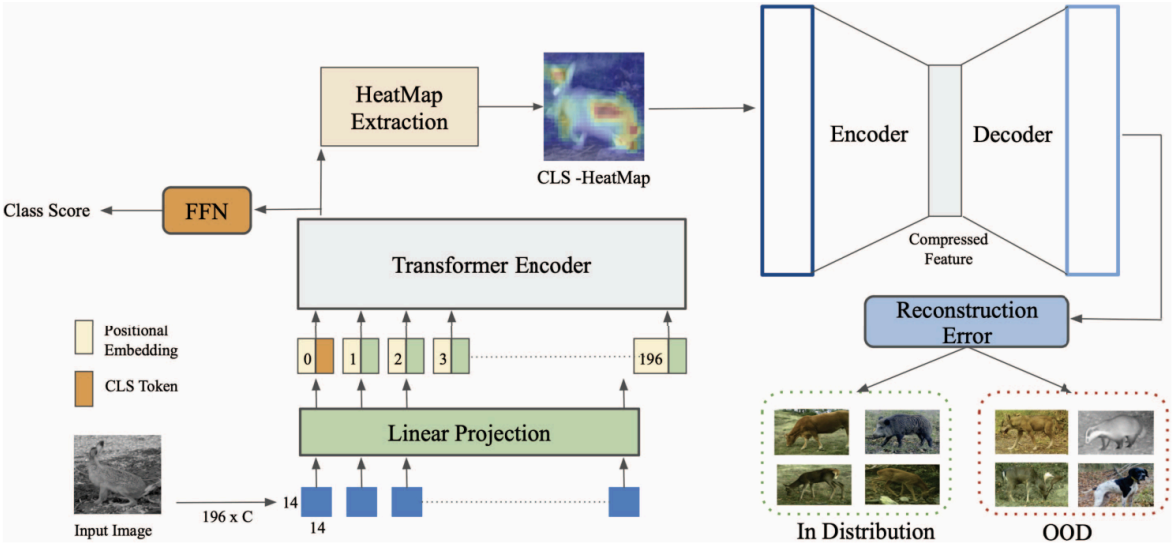

Figure 10 – ViT encoder is used to classify samples and feed the CLS token heatmap to the Convolutional Autoencoder, then the reconstruction error can be used to perform OOD detection. [1]

By leveraging attention-based mechanisms and autoencoder-based techniques, this model captures fine-grained features and class-specific patterns, significantly enhancing OOD detection performance.

In order to exploit a face identification module, the model developed in "*Face transformer for recognition*" [2] is adapted. In this paper, transformer models are used to perform face recognition tasks.

In particular, the patch generation process has been modified to consider inter-patch information. The model trained on MS-Celeb-1M dataset has achieved comparable performance as CNNs benchmarks.

The architecture of this model is the following.

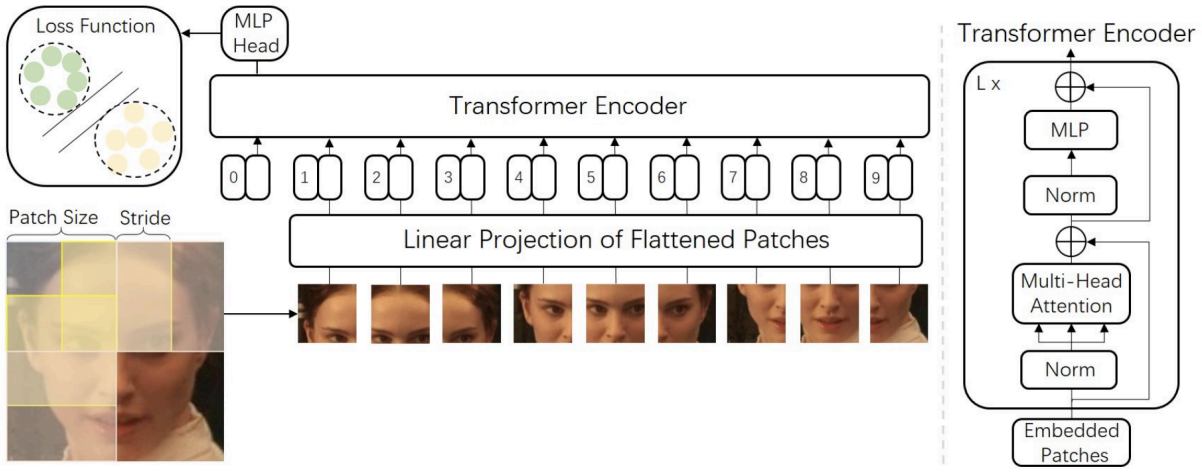

Figure 11 – Images are split into multiple patches as input-tokens to the transformer encoder. [2]

So this transformer architecture has been used to extract visual self-attention maps from images of FF++ dataset.

Part of the code developed by the authors of the paper "Face transformer for recognition" is utilized and implemented as a starting point to be adapted to the needs of the research.

As presented above, *FaceForensics++* is a dataset composed of 1000 original videos[3] that have been manipulated using five automatic face modification methods: Deepfakes, Face2Face, FaceSwap, FaceShifter and NeuralTextures [4].

Another dataset used for initial code testing is LFW, which contains about 13000 face images.

[3] *In this work the first 100 frames of each video are considered, in a total amount of 600'000 images in the dataset.*

2. Method, Model, Experiments & Results

2.1. Self-Attention Heatmaps Extraction

This work is the result of an experimentation research on DeepFakes detection task.

The first step in this process has been the construction of a system to extract visual attentions of the CLS token[4], which can be read in the form of heatmaps.

Here, an example of heatmap, plotted in colors:

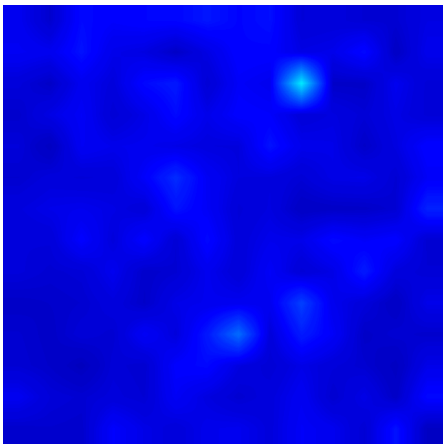

Figure 12 – Heatmap representation of the visual attention mechanism given on a face-crop frame.

If overlapped to the original frame, it shows the areas where the ViT focus more:

Figure 13 – Heatmap representation overlapped to the original frame.

[4] *The CLS token, short for "classification" token, is a special token used in ViT to perform classification tasks. It is inserted at the beginning of the token sequence and serves as a representative token for the entire input image. During the training process, the self-attention mechanism allows the model to attend to different parts of the input image and capture relevant features. The output corresponding to the CLS token is then used for classification tasks, such as determining whether an image contains a particular object or classifying facial expressions. The CLS-attention map is a visual representation of the attention weights assigned to different parts of the input image by the self-attention mechanism, specifically focusing on the CLS token.*

As said above, attention mechanisms focus on specific regions of an image or input sequence to process relevant information:

- Allow neural networks to dynamically focus on relevant parts of the input data while suppressing irrelevant information.
- Assign learnable weights to different parts of the input data, indicating their importance in the task at hand.

In the context of computer vision, visual attention mechanisms help analyzing images efficiently, as follow:

1. Neural networks extract features from the input image, capturing hierarchical representations of visual information.
2. Attention mechanisms are integrated into the network architecture to dynamically focus on salient regions of the image, enhancing the representation of relevant features.
3. The attention mechanism assigns higher weights to regions of the image containing important facial features, such as eyes, nose, and mouth.

This enables the model to prioritize relevant information during facial recognition tasks, like handling different poses, expressions and lighting conditions. Furthermore, visualizing the attention weights enables interpretability, allowing researchers to understand which regions of the input image contribute most to the model's predictions, like in the example above, where the higher values of attentions are near the mouth and the eyes on the face considering the cropped image.

To extract the above-mentioned heatmaps, and consequently the attention weights of importance of the areas in a face-picture, has been taken advantage of the work "*Face transformer for recognition*" [2] where researchers created a ViT architecture to perform a facial recognition task. Face Transformer model follows the architecture of ViT, which applies the original Transformer. The only difference is that they have modified the tokens generation method of ViT, to generate tokens with sliding patches, i.e., to make the image patch overlaps, for the better description of the inter-patch information, as shown in the following Figure.

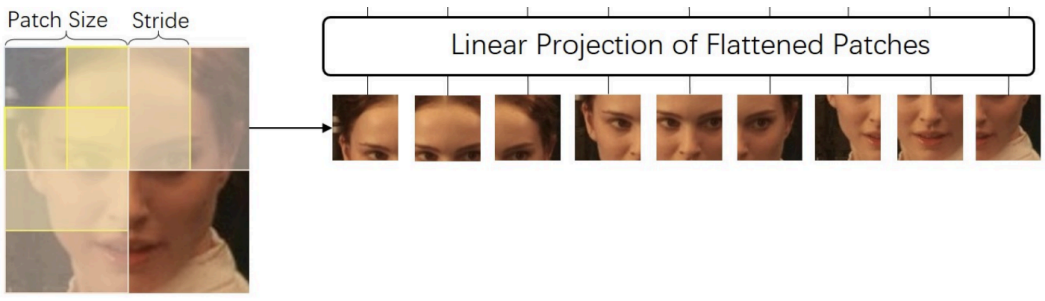

Figure 14 – Image path overlaps for better description of inter-patch information.

The original Vision Transformer (ViT), introduced in the paper "*An Image is Worth 16 x 16 Words: Transformers for Image Recognition at Scale*" [19], is a deep learning architecture that applies the transformer architecture, shown in the following figure.

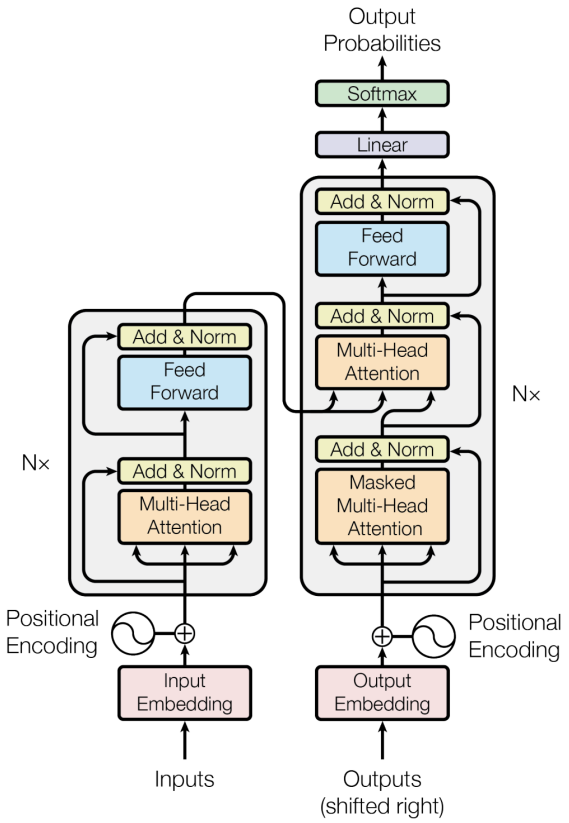

Figure 15-a – The original transformer-model architecture for NLP. [5]

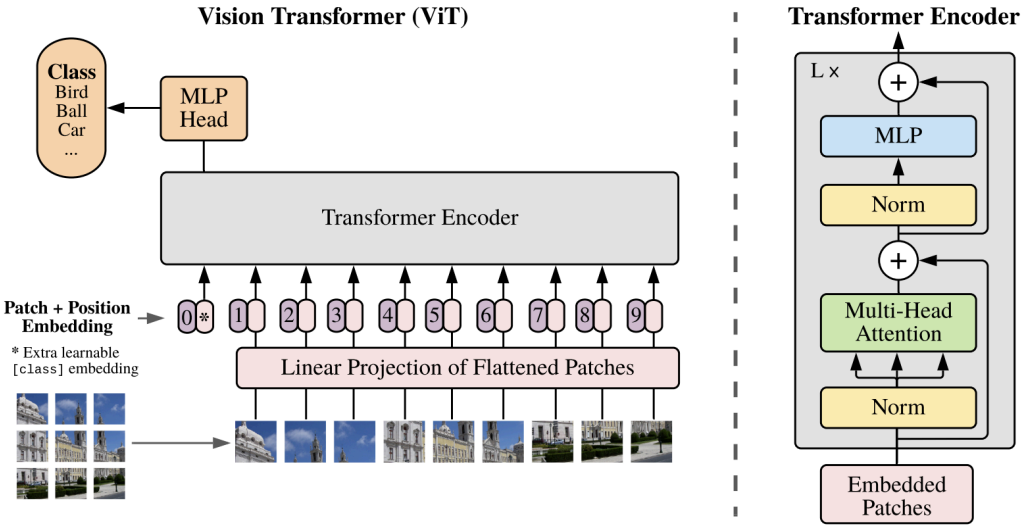

Figure 15-b – The original Vision-Transformer model. Split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, the standard approach of adding an extra learnable "classification token" to the sequence is used. [19]

Unlike traditional convolutional neural networks (CNNs) that process images in a hierarchical manner, ViT treats images as sequences of tokens and utilizes self-attention mechanisms to capture global and local dependencies, as follow:

- Tokenization: ViT divides the input image into fixed-size patches, which are then flattened into sequences of tokens. Each token represents a patch of the image, preserving spatial information.
- Transformer Blocks: ViT consists of a stack of transformer blocks, each comprising multi-head self-attention layers and feedforward neural networks. These blocks allow the model to capture interactions between different parts of the input image, by the self-attention mechanism:
    - As said above, enables the model to weight the importance of different elements in the input sequence. Each token attends to all other elements, learning a weighted representation that captures contextual information.
- Multi-Head Attention: Transformers typically employ multi-head attention, where the input sequence is projected into multiple linear subspaces. Self-attention is performed independently in each subspace, allowing the model to capture diverse interactions and attend to different aspects of the input sequence simultaneously.
- Positional Encoding: Since transformers do not inherently understand the sequential order of tokens, positional encoding is added to the input embeddings to convey positional information. This enables the model to differentiate between tokens based on their positions in the sequence.

In particular, "*Face transformer for recognition*" *[2]* architecture has been used as a pre-trained[5] model on the large scale face recognition dataset *MS-Celeb-1M*. Once cloned[6] and assessed[7] its fluid behavior in the work environment, the code has been adapted to the research first aim of extracting CLS token self-attention heatmaps from the *FF++* dataset. The heatmaps have been drawn out from every frame in the dataset, for an exact amount of 599'694 images to be evaluated.

[5] *Pre-training and Fine-tuning: ViT-based architectures are typically pre-trained on large-scale image datasets using self-supervised learning objectives. After pre-training, the model can be fine-tuned on specific downstream tasks, such as image classification, object detection, or image segmentation.*

[6] *A GIT term to say "download the project-folder from an online git repository".*

[7] *Assessed on the LFW benchmark database.*

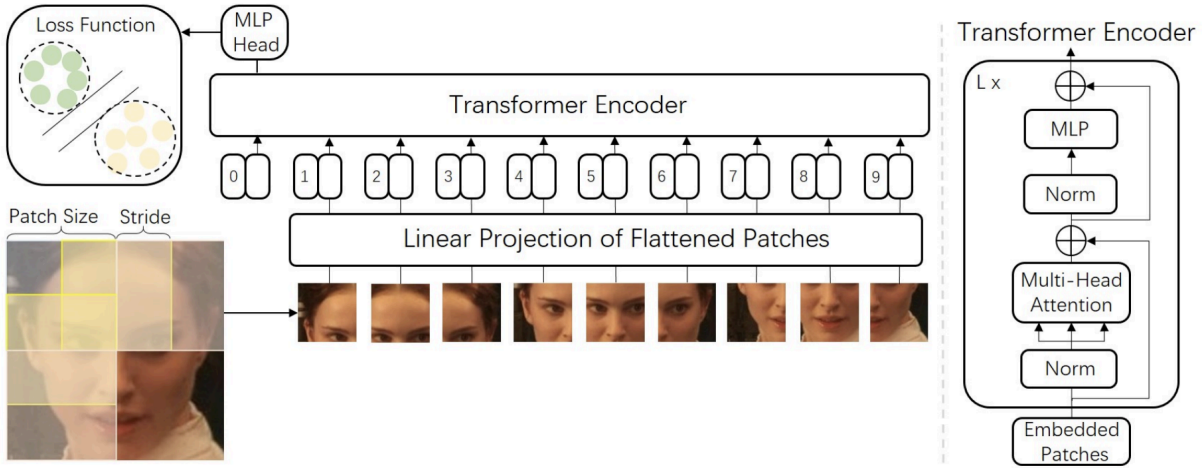

Figure 11 – Images are split into multiple patches as input-tokens to the transformer encoder. [2]

Technically, the heatmaps have been extracted from the last layer of the transformer, performing the test-forward in the model, for each image in the dataset and collecting all these features.

At this point the research question of this work can be explained: is it possible to use features from pre-trained models to obtain an unsupervised model to perform OOD detection?

This work wants to show if it is possible to do so, using the self-attention from a pre-trained supervised model, in this case which scope is to classify faces.

2.2. Convolutional-AutoEncoder Training

The second step in the process flow, is to train a convolutional autoencoder. This is of crucial importance since the error of reconstruction given by the trained conv-AE during the test phase is the one used as a metric to perform the out-of-distribution detection task, which should discern real-images from fake-ones.

The conv-AE used is the one produced by researchers in the paper "*Leveraging Visual Attention for out-of-Distribution Detection*" [1] where it is trained on attention heatmaps, produced by a ViT classifier, and the reconstruction error is used to flag samples as In-Distribution or Out-of-Distribution, thus providing an estimate of the uncertainty of the classifier prediction. In the case, the classifier has been tested using benchmarks CIFAR10 and CIFAR100 datasets, and to test OOD-Detection in a real-world setting, they also collected a novel dataset, called "*WildCapture*".

Here an example of attention-heatmaps created in the paper:

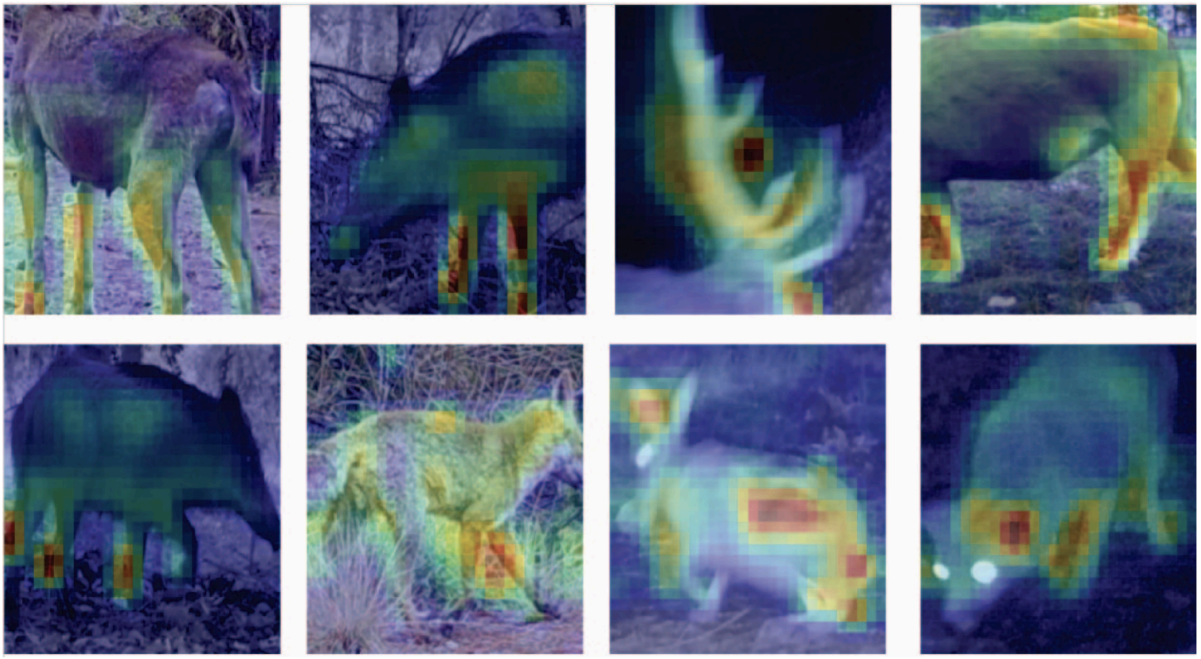

Figure 16 – Attention heatmaps examples. The visual classifier attend mostly to legs and antlers. [1]

For sake of clarity, the architecture of the data flow method is displayed again.

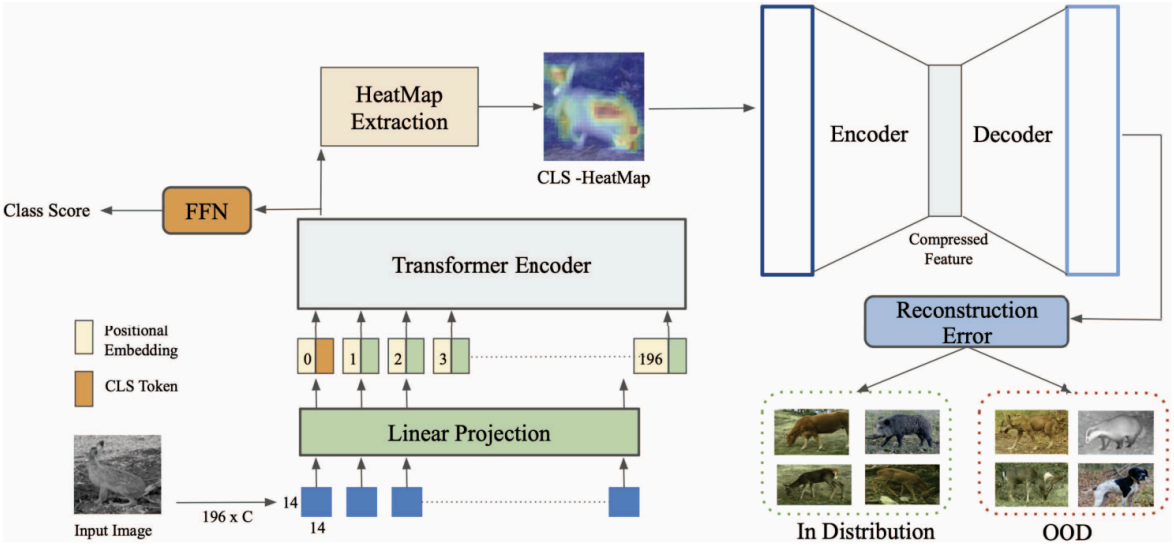

Figure 10 – ViT encoder is used to classify samples and feed the CLS token heatmap to the Convolutional Autoencoder, then the reconstruction error can be used to perform OOD detection. [1]

Data pipeline:

- Images, first, go through Transformer where visual self-attention heatmaps are bringed out:
    - In this step a dataset of images' self-attentions and related labels are collected.
- Heatmaps, then, are used to train the Convolutional Auto-Encoder considering only the In-distribution classes, e.g. in the case of the paper, the animal classes already known and labeled; while in the case of this work the Real images (non-manipulated). The autoencoder learns to encode the meaningful and distinctive representations of the attention maps, facilitating precise image reconstruction.
- Lastly, every image on the dataset, including the one of the out-of-distribution classes, passes through the conv-AE which returns a reconstruction error.
- It is expected that the reconstruction error of Out-of-distribution images is on average higher than the one displayed on In-distribution classes images. This difference can be used as a threshold to discriminate between In-distribution and out-of-distribution images, thus, in this research, detect forgery-manipulated images with a certain confidentiality.

By leveraging attention-based mechanisms and autoencoder-based techniques, this method captures fine-grained features and class-specific patterns, significantly enhancing OOD detection performance.

Considering so, the architecture has been re-implemented in this work as follows.

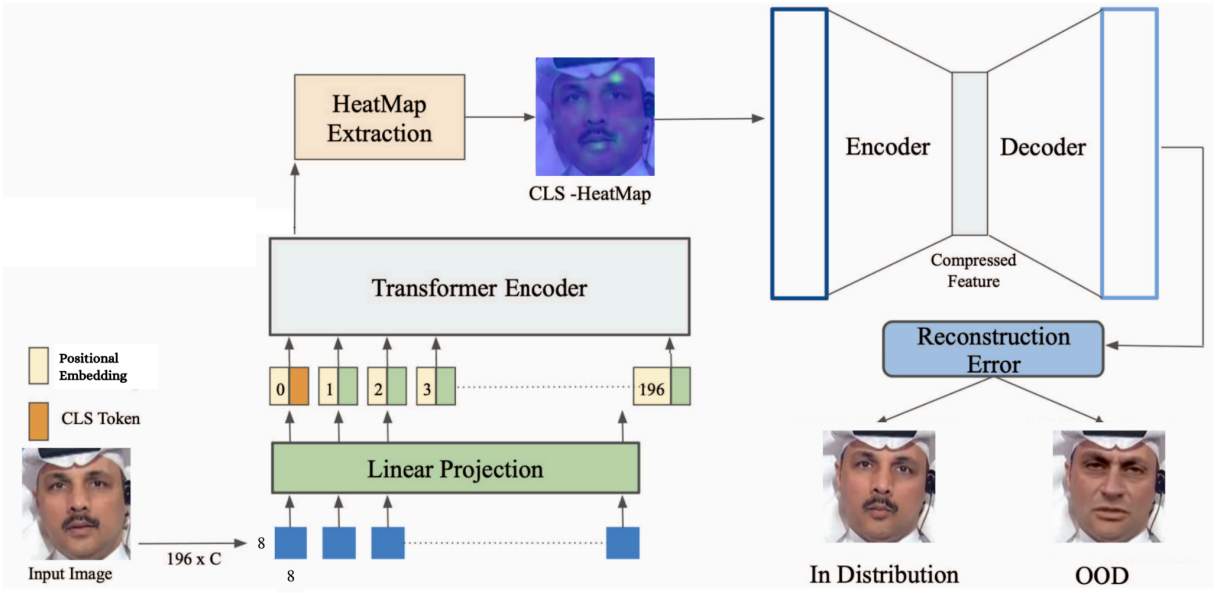

Figure 17 – ViT encoder is used to classify samples and feed the visual-attention heatmaps to the Convolutional Autoencoder, then the reconstruction error can be used to perform OOD detection.

2.3. Data Processing Details

The Vision Transformer classifier is implemented as pre-trained on the large scale face recognition *MS-Celeb-1M* dataset. The model takes input images of size 112 × 112 × 3, starting from 224 × 224 × 3 images on the FF++ dataset which have been resized, and divides them into patches of size 8 × 8, with overlap. This way, each image is split into a grid of 14 × 14 patches, resulting in a total of 196 patches. To facilitate efficient storage and analysis, the attention heatmaps have been resized to a standardized size of 128 × 128 × 1.

With all visual attention heatmaps extracted, the dataset to train, validate and test the Convolutional AutoEncoder is formed, with a total amount of 599'694 images, divided into 80% for training, 10% for validation and 10% for test. More in detail, to create the subsets, the dataset has been divided by videos:

- 80% of the videos are used as training, from the total of 1000 folders of videos, the first 800 form the training-set, for total amount of 79'954 images[8].
- Similarly, the following 10% of videos folders have been used as a validation-set, with 9995 frames.
- Finally, the last 10% of videos have been used as a test-set, with a total of 10'000 images for each kind of forgery and real ones in FF++ (5 forged + 1 real as shown in figure 8) for a total of 60'000 frames.

LEGEND: • IN DISTRIBUTION • OUT-OF-DISTRIBUTION		DATASETS		
		TRAINING	VALIDATION	TEST
IMAGES	REAL	79'954	9'995	10'000
	DEEPFAKES	✖	✖	10'000
	FACE2FACE	✖	✖	10'000
	FACESHIFTER	✖	✖	10'000
	FACESWAP	✖	✖	10'000
	NEURAL TEXTURES	✖	✖	10'000

Figure 18 – Dataset partition into [80-10-10]% for [Train - Val - Test] sets.

[8] *Not all videos have exactly 100 frames.*

Note that, the training and validation sets are formed only by real images, since it is the in-distribution classes needed to train the conv-AE.

Convolutional Autoencoder is trained for the task of out-of-distribution (OOD) detection, leveraging visual attention extracted from the pre-trained Vision Transformer. The Convolutional AutoEncoder is designed to reconstruct input images. The architecture, as in figure 7, consists of an encoder and decoder, each comprising several convolutional layers, with Leaky ReLU activation functions to introduce a regularization effect.

The encoder takes grayscale input images of size 128 × 128 × 1 and progressively reduces the spatial dimensions while increasing the number of channels. It culminates in a bottleneck layer of size 512 × 1 × 1. The decoder then upscales and progressively reconstructs the original input image through transposed convolutions and activations. During the training process, the model is optimized to minimize the Mean Squared Error (MSE) loss between the reconstructed heatmaps and the original input.

During Convolutional Autoencoder training, grayscale visual CLS token attention heatmaps extracted from the Vision Transformer have been used, as explained above, with an input size of 128 × 128 × 1. The Autoencoder architecture comprises encoder and decoder blocks[9]

To regularize each convolutional layer, Leaky ReLU activation has been engaged with a negative slope of 0.2. For optimization, the Adam optimizer with an initial learning rate of 0.0001 has been imposed. To enhance convergence stability and overall model performance, a linear learning rate scheduler has been implemented. During the first 40 epochs, the learning rate halved every 10 epochs, after which it remained constant.

The Autoencoder's primary objective during training was to minimize the Mean Squared Error (MSE) loss between the reconstructed output and the input images. This training setup empowered the Autoencoder to learn meaningful representations of the input data, facilitating precise image reconstruction and substantially contributing to the subsequent out-of-distribution Detection process.

[9] *Specific details for each layer can be found in the Appendix.*

2.4. Experiments & Considerations

Once the architecture is complete to process the data-flow, some tests have been performed.

As depicted above, the core training involved the 80% of all videos available in FF++, it took about 1 hour to be trained[10], with a batch size = 200 on a machine equipped with NVIDIA TITAN RTX[11].

Performing the test of the conv-AE, on videos never seen by the model during the training process, the model returns a pickle[12] file containing the MSE[13] values computed on every frame, between the input image and the one reconstructed by the Autoencoder.

The test has been performed for each of the 6 test-sets, the one for real images and the 5 for forged images.

To have a comparison model, the convolutional AutoEncoder has been trained also among the same exact dataset and partitions, following same protocol but using the RGB[14] images without performing any feature extraction, as in the core model with self-attention maps created by the ViT.

[10] *With a lowest loss value* = 0.0003302.

[11] *For details about this, go check the appendix.*

[12] *Pickle file extension: .pkl*

[13] *MSE: Mean Square Error, in this case calculated between the input image and the reconstructed one.*

[14] *RGB: Red Green Blue, is one way to encode colors of each pixel in a 3 channels image. Each color channel can have values ranging from 0 to 255, where 0 represents no intensity and 255 represents full intensity, with a large number of color combinations.*

2.5. Results & Comments

2.5.1. Exploiting Self-Attention Maps

In order to compare results, ROC[15] curves of the tests have been created from the MSE pickle files built among all different forgery, to measure the ability of this model and approach to detect every specific manipulation method under consideration.

As a direct consequence, AUC[16] of the ROC curves (AUROC) has been calculated for each case and used as a comparison metric of the model's performance in detection tasks.

The results are displayed below.

The first outcome is the case considering *Deepfakes* forgery and real images in the binary classification problem. The curve suggests a threshold where the model can discern, at least more than in a random chance choice (dash line), between the real images and manipulated ones. Indeed, in this case the AUROC value is 0.62 > 0.50.

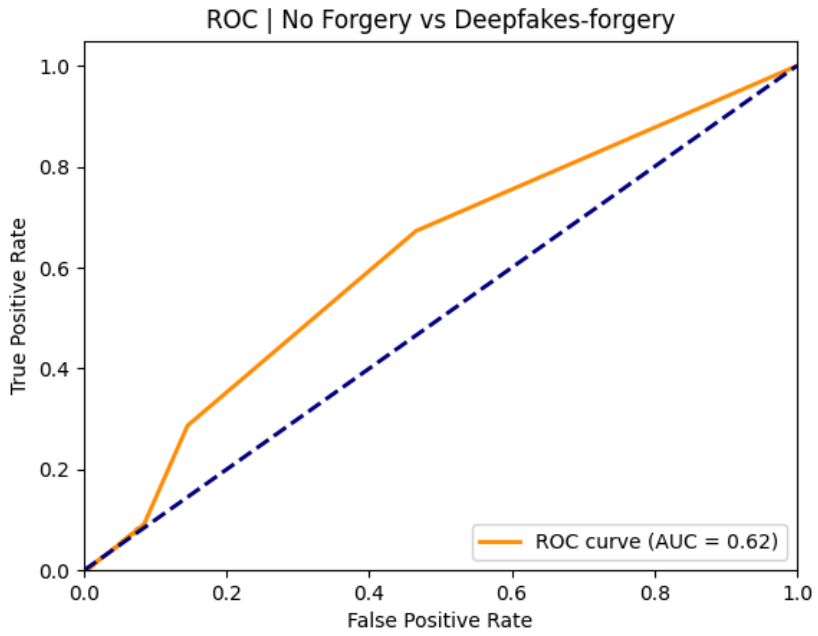

Figure 19 – ROC & AUC of the binary classification problem involving Deepfakes vs Real images.

[15] *ROC (Receiver Operating Characteristic) curves are graphical plots that illustrate the diagnostic ability of a binary classification model across different threshold settings. They display the trade-off between the true positive rate (sensitivity) and the false positive rate (1 - specificity). ROC curves summarize the performance of a classifier across all possible thresholds, providing insight into its ability to discriminate between the positive and negative classes.*

[16] *The AUC (Area Under the Curve) or AUROC (Area Under the Receiver Operating Characteristic Curve) is a single scalar value that quantifies the overall performance of a binary classification model. It represents the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance. AUC measures the classifier's ability to distinguish between the two classes, with higher values indicating better performance. AUC ranges from 0 to 1, where 0.5 corresponds to random guessing and 1 represents perfect discrimination.*

For the next 3 kinds of forgeries *Face2Face*, *FaceShifter*, *FaceSwap* the results are similar to this one, with an AUC value around 0.60.

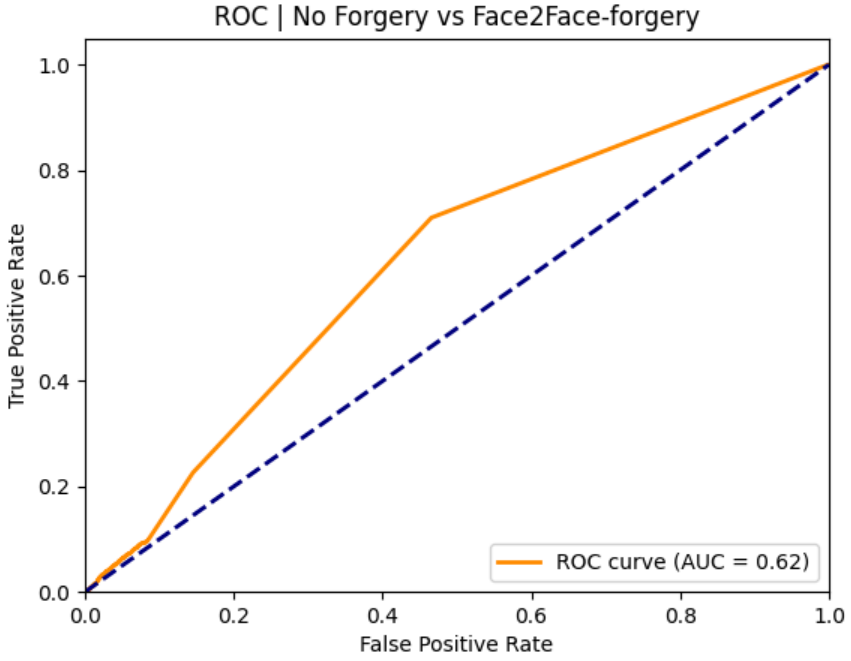

Figure 20 – ROC & AUC of the binary classification problem involving Face2Face vs Real images.

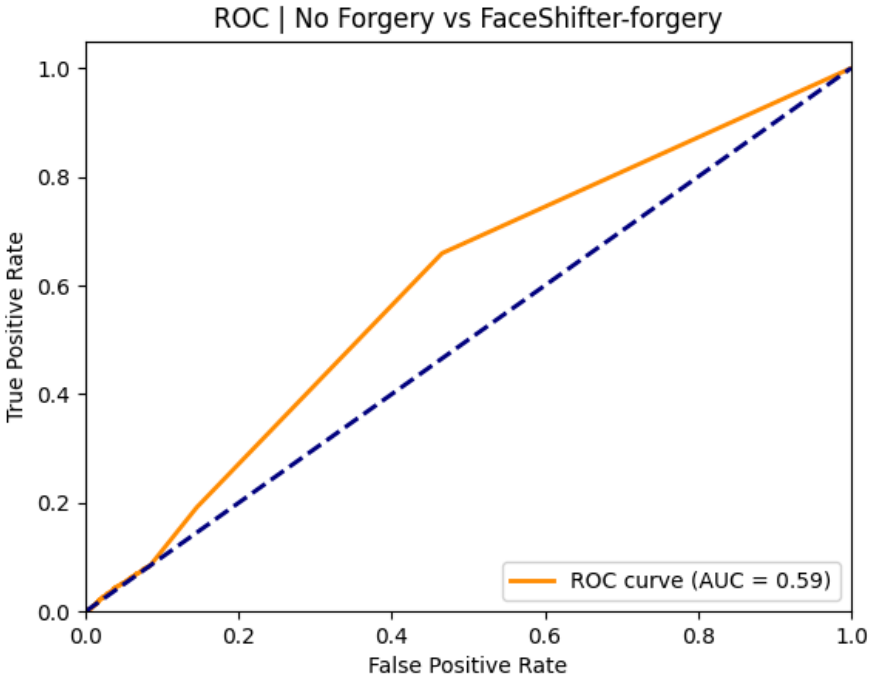

Figure 21 – ROC & AUC of the binary classification problem involving FaceShifter vs Real images.

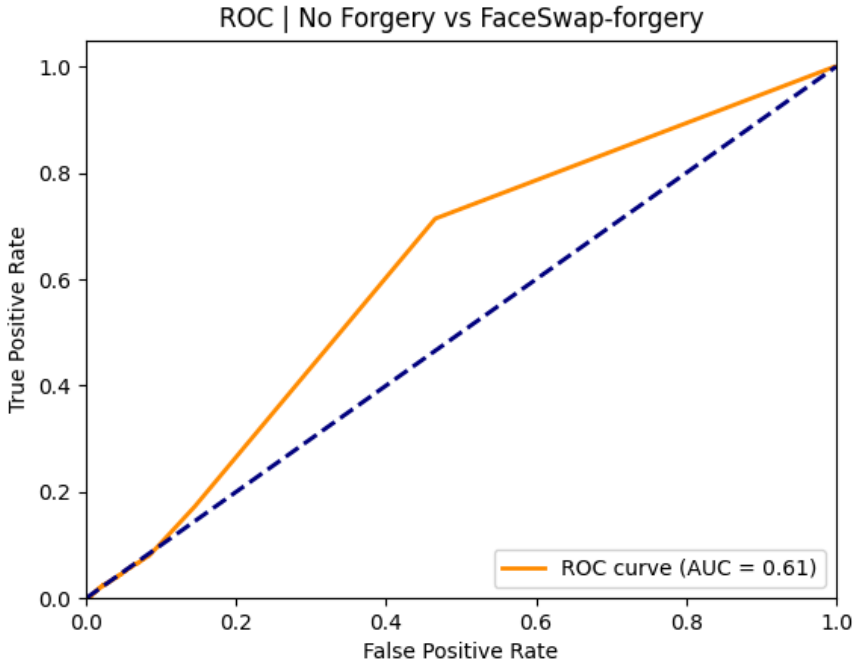

Figure 22 – ROC & AUC of the binary classification problem involving FaceSwap vs Real images.

For the last forgery considered, *Neural Textures*, the detection ability measured as the AUROC value in the binary classification problem drops down by about 0.06, suggesting that this forgery is harder to be detected.

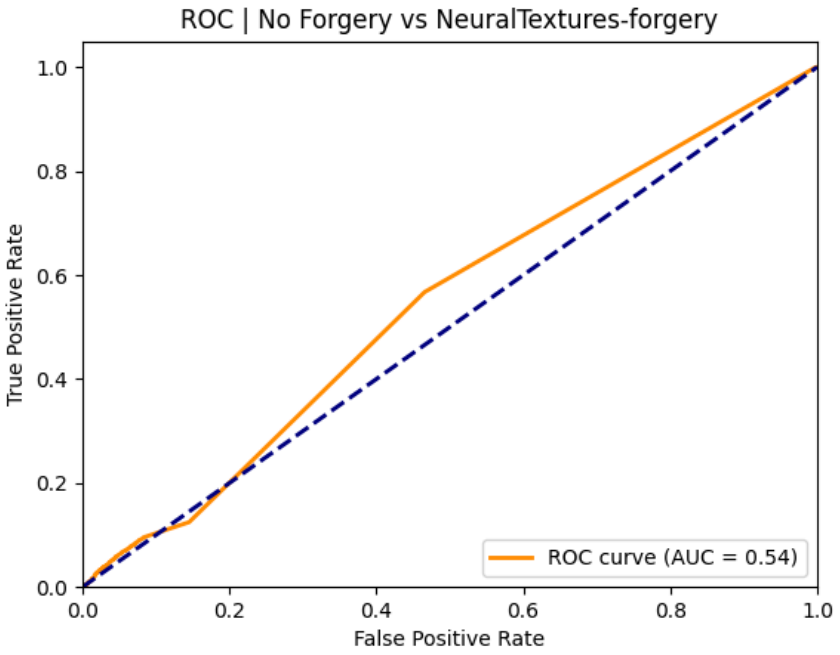

Figure 23 – ROC & AUC of the binary classification problem involving NeuralTextures vs Real images.

The case of real vs forged, without distinguishing between one forge and another, has been reported, and as expected it displays a mean value compared to the results mentioned above with a corresponding AUROC value of 0.60.

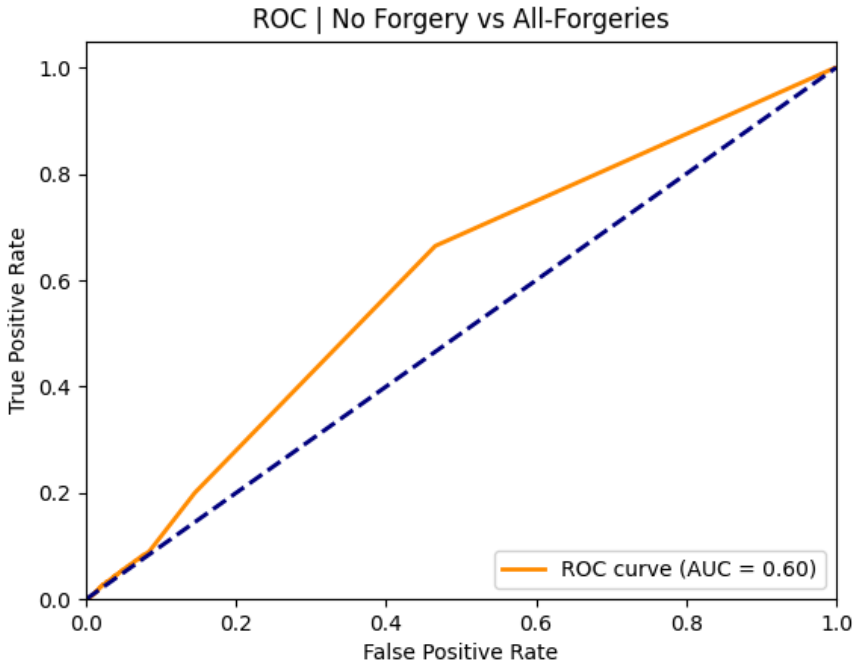

Figure 24 – ROC & AUC of the binary classification problem involving All-Forgeries vs Real images.

The following table shows the synthesis of these results.

Real vs Forged Images Model Detection Ability	
Forgery	**AUROC**
Deepfakes	0.62
Face2Face	0.62
FaceShifter	0.59
FaceSwap	0.61
NeuralTextures	0.54
All forgeries	0.60

Table 1 - Results of the model performance in the detection for the 5 forgeries.

The method shows a performance lower than the one achievable via supervised models in state of the art [14], but it has transfer learning ability and it is independent to the kind of forgery, in the sense that it can detect forgeries never seen before, so it does not need any new data to perform the detection of new forgery, differently to the supervised methods [16].

2.5.2. Trained on RGB images

Briefly, the results of the RGB-based model as a comparison are presented. It is possible to note that the model shows no detection ability, given ROC curves following the dash-line of random guessing, so with AUROC values around 0.50.

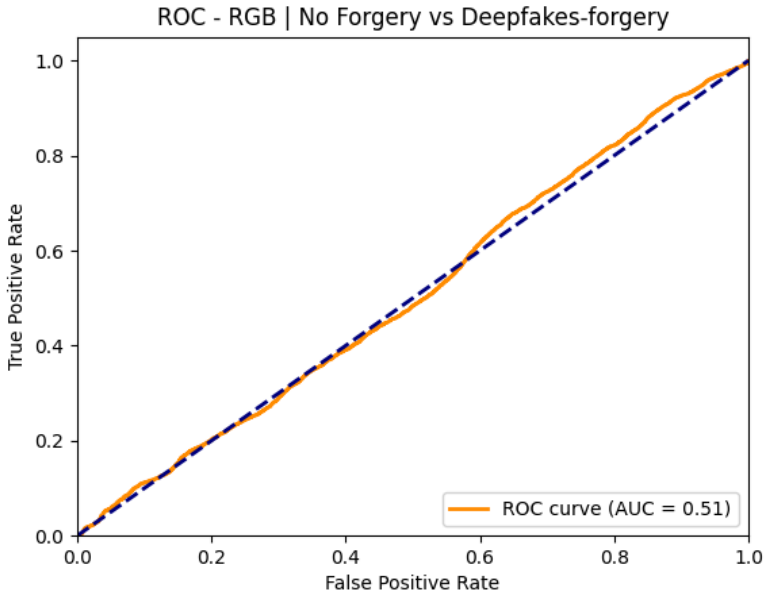

Figure 25 – ROC & AUC of the binary classification problem involving Deepfakes-Forgeries vs Real RGB images.

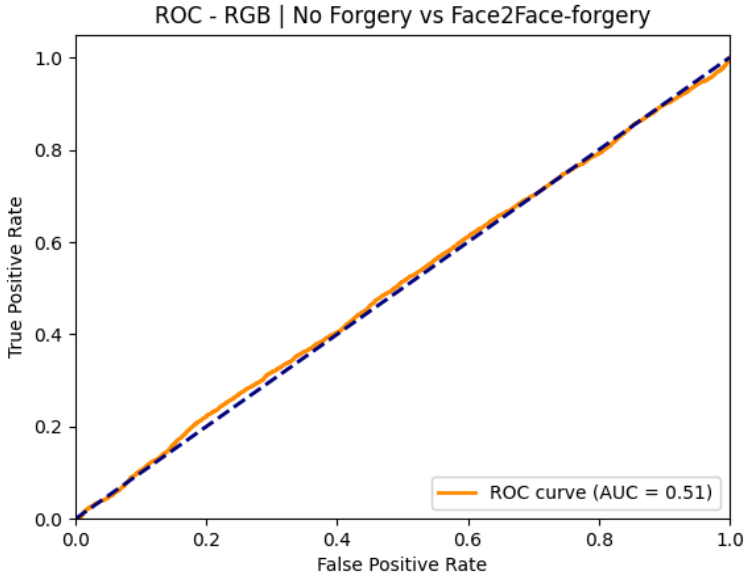

Figure 26 – ROC & AUC of the binary classification problem involving Face2Face-Forgeries vs Real RGB images.

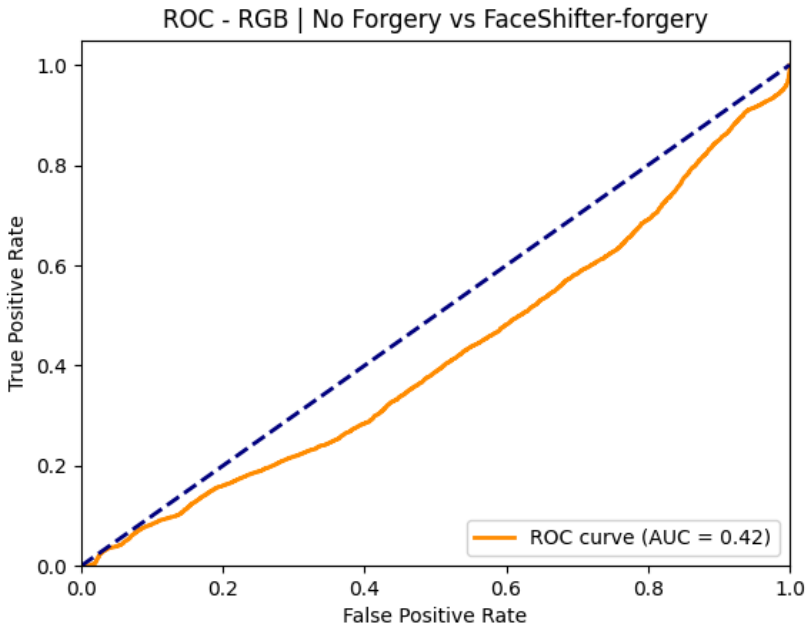

Figure 27 – ROC & AUC of the binary classification problem involving FaceShifter-Forgeries vs Real RGB images.

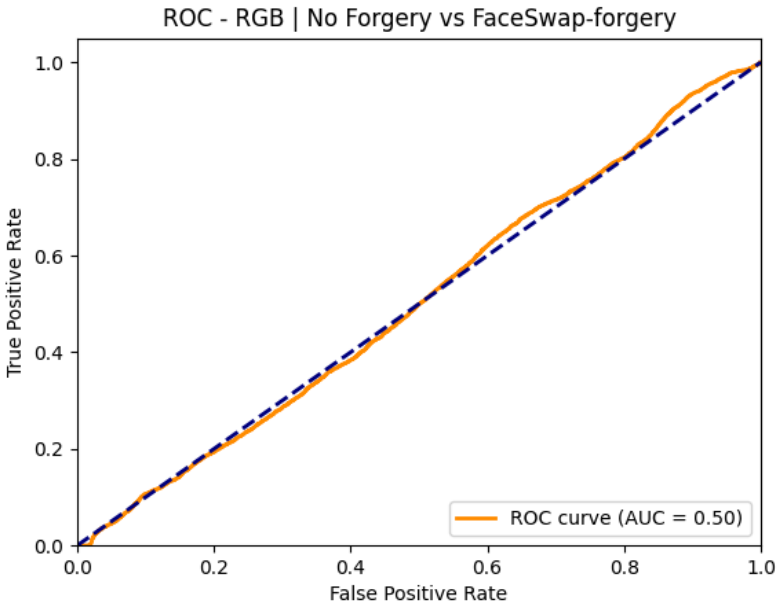

Figure 28 – ROC & AUC of the binary classification problem involving FaceSwap-Forgeries vs Real RGB images.

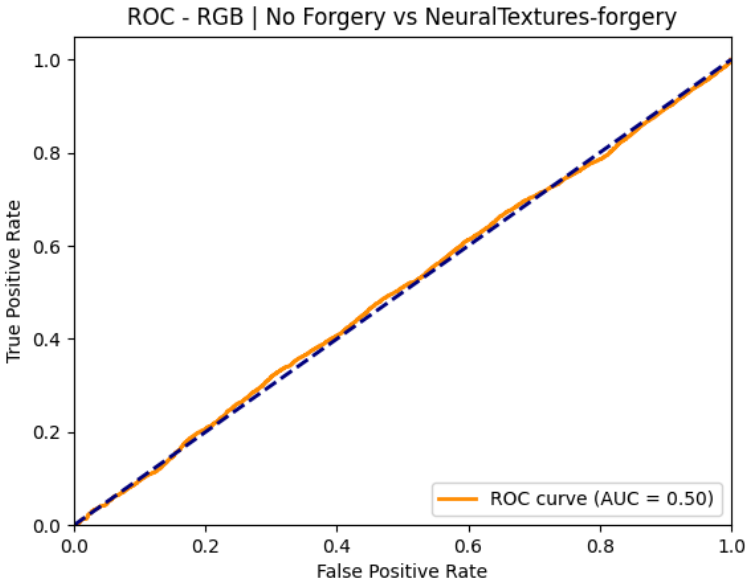

Figure 29 – ROC & AUC of the binary classification problem involving NeuralTextures-Forgeries vs Real RGB images.

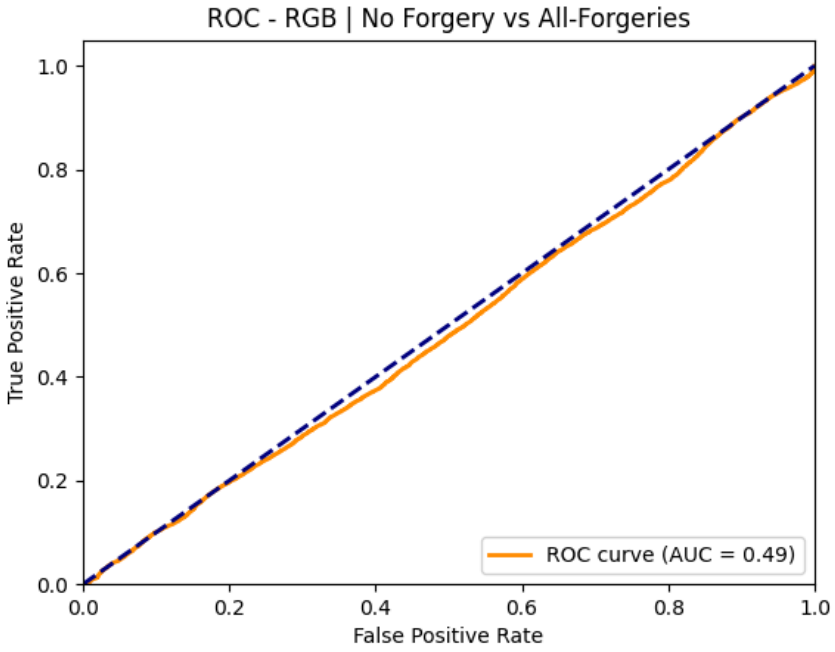

Figure 30 – ROC & AUC of the binary classification problem involving All-Forgeries vs Real RGB images.

All the results are summarized in the following table.

AUROC | Real vs Forged Images Models Detection Ability		
Forgery	**Attention-based**	**RGB-based**
Deepfakes	0.62	0.51
Face2Face	0.62	0.51
FaceShifter	0.59	0.42
FaceSwap	0.61	0.50
NeuralTextures	0.54	0.50
All forgeries	0.60	0.49

Table 2 - Results of the attention-based and RGB-based models performance in the detection for the 5 forgeries.

3. Conclusions

The experimentation research conducted in this thesis work focuses on DeepFakes detection, employing a two-step methodology involving the extraction of CLS token self-attention heatmaps and the training of a convolutional autoencoder.

By leveraging attention mechanisms and autoencoder-based techniques, the approach captures fine-grained features and class-specific patterns, significantly enhancing out-of-distribution (OOD) detection performance.

The first step involves constructing a system to extract visual attentions, represented as weights-maps, which dynamically focus on salient regions of images. These attention mechanisms prioritize relevant facial features during facial recognition tasks, aiding in interpretation and understanding of model predictions.

The Vision Transformer (ViT) architecture, pre-trained on a large-scale face recognition dataset, serves as the foundation for attentions extraction. While, training a convolutional autoencoder on the extracted visual-attention heatmaps drive the discern between real and manipulated images.

The autoencoder learns to encode meaningful representations, facilitating precise image reconstruction and enabling OOD detection based on reconstruction error.

Experimental results demonstrate the model's ability to detect various manipulation methods, even with low AUROC values ranging from 0.54 to 0.62, outperforming the RGB-based model which scores no-detection ability .

While the performance may be lower than supervised methods, the transfer learning capability and independence from specific forgery types highlight the potential of the proposed approach in addressing novel forgery detection challenges.

4. Future Developments

This work can be extended to achieve better performance and improve its utility, for example, to real-time DeepFake detection and deployment in practical scenarios, such as social media platforms or video streaming services. This could be crucial for combating the spread of misinformation.

Given some considerations that arose during the research, the model can also be evaluated using different metrics and scenarios, as it is possible to treat single frames or entire videos, leading to different results.

One way to improve this model could be to implement a more robust data flow, from data gathering to data engineering. For example, the frames cropping phase can be enhanced and standardized.

Another improvement could involve training a Vision Transformer with different data more aligned with the necessity of a deepfake-detection task, maybe incorporating diverse datasets, including real-world scenarios images, could enhance the model's ability to generalize.

Alternatively, utilizing a face attribute estimation module like "MiVOLO: Multi-input Transformer for Age and Gender Estimation" [3] could be explored and studied for future tests.

Exploring alternative feature extraction methods, such as using a ResNet-n and then training an AutoEncoder, could also be beneficial.

Implementing ensemble learning techniques by combining multiple models, each trained on different aspects or representations of the data, could potentially enhance detection accuracy and robustness.

5. References

[1] Luca Cultrera, Lorenzo Seidenari, and Alberto Del Bimbo; "*Leveraging Visual Attention for out-of-Distribution Detection*"; Proceedings of the IEEE/CVF International Conference on Computer Vision; 2023.

[2] Zhong, Yaoyao, and Weihong Deng; "*Face transformer for recognition*"; arXiv preprint arXiv:2103.14803 (2021).

[3] Kuprashevich, Maksim, and Irina Tolstykh; "*MiVOLO: Multi-input Transformer for Age and Gender Estimation*"; arXiv preprint arXiv:2307.04616; (2023).

[4] Andreas Rössler and Davide Cozzolino and Luisa Verdoliva and Christian Riess and Justus Thies and Matthias Nießner; "*FaceForensics++: Learning to Detect Manipulated Facial Images*"; International Conference on Computer Vision (ICCV); 2019; Available @ https://paperswithcode.com/dataset/faceforensics-1

& https://github.com/ondyari/FaceForensics

[5] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). "*Attention is All You Need*". In Advances in Neural Information Processing Systems (pp. 5998-6008).

[6] https://medium.com/@birla.deepak26/autoencoders-76bb49ae6a8f

[7] https://www.linkedin.com/pulse/ai-vs-ml-dl-harmanjeet-singh-gdi8c

[8] https://www.linkedin.com/pulse/role-ai-healthcare-infosense-ai

[9] https://www.nextbigfuture.com/2019/12/what-are-the-limits-of-deep-learning-going-beyond-deep-learning.html

[10] https://www.youtube.com/watch?v=fKXztwtXaGo

[11] https://seeflection.com/22579/viral-deepfake-on-tiktok-causes-outrage/

[12] https://towardsdatascience.com/using-transformers-for-computer-vision-6f764c5a078b

[13] https://encord.com/blog/what-is-out-of-distribution-ood-detection/

[14] Sohail Ahmed Khan and Duc-Tien Dang-Nguyen; "*Deepfake Detection: A Comparative Analysis*"; August 2023; available at: https://arxiv.org/pdf/2308.03471.pdf

[15] Karima Omar, Rasha H. Sakr, Mohammed F. Alrahmawy; "*An ensemble of CNNs with self-attention mechanism for DeepFake video detection*"; Mansoura University, Egypt; 23 November 2023.

[16] Christian Rathgeb, Ruben Tolosana, Ruben Vera-Rodriguez, Christoph Busch, Editors; "*Advances in Computer Vision and Pattern Recognition, Handbook of Digital Face Manipulation and Detection, From DeepFakes to Morphing Attacks, Chapter 11, Deepfake Detection Using Multiple Data Modalities*"; Hanxiang Hao, Emily R. Bartusiak, David Güera, Daniel Mas Montserrat, Sriram Baireddy, Ziyue Xiang, Sri Kalyan Yarlagadda, Ruiting Shao, János Horváth, Justin Yang, Fengqing Zhu, and Edward J. Delp.

[17] Gragnaniello D, Cozzolino D, Marra F, Poggi G, Verdoliva L (2021); "*Are GAN generated images easy to detect? A critical analysis of the state-of-the-art*". In: Proceedings of the IEEE international conference on multimedia and expo, July 2021.

[18] Cozzolino D, Rössler A, Thies J, Nießner M, Verdoliva L (2021); "*Id-reveal: Identity-aware deepfake video detection*". In: arXiv preprint arXiv:2012.02512, December 2021.

[19] Alexey Dosovitskiy*,† , Lucas Beyer* , Alexander Kolesnikov* , Dirk Weissenborn* , Xiaohua Zhai* , Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby*,† * equal technical contribution, † equal advising Google Research, Brain Team; "*An Image is Worth 16 x 16 Words: Transformers for Image Recognition at Scale*"; *at ICLR 2021.*

6. Appendix

6.1. Code

The codes and further details of this work might be available at ***https://github.com/Gianmarco-San/DeepFake-Detection***[17] in the near future.

6.2. Architecture details

The Convolutional Autoencoder layers used in this research are displayed.

Encoder					
Layer	Input Shape	Output Shape	Kernel	Stride	Padding
Conv2d (1 → 32)	3×128×128	32×64×64	4×4	2	1
Conv2d (32 → 32)	32×64×64	32×32×32	4×4	2	1
Conv2d (32 → 32)	32×32×32	32×32×32	3×3	1	1
Conv2d (32 → 64)	32×32×32	64×16×16	4×4	2	1
Conv2d (64 → 64)	64×16×16	64×16×16	3×3	1	1
Conv2d (64 → 128)	64×16×16	128×8×8	4×4	2	1
Conv2d (128 → 64)	128×8×8	64×8×8	3×3	1	1
Conv2d (64 → 32)	64×8×8	32×8×8	3×3	1	1
Conv2d (32 → 512)	32×8×8	512×1×1	8×8	1	0
Decoder					
ConvTranspose2d (512 → 32)	512×1×1	32×8×8	8×8	1	0
Conv2d (32 → 64)	32×8×8	64×8×8	3×3	1	1
Conv2d (64 → 128)	64×8×8	128×8×8	3×3	1	1
ConvTranspose2d (128 → 64)	128×8×8	64×16×16	4×4	2	1
Conv2d (64 → 64)	64×16×16	64×16×16	3×3	1	1
ConvTranspose2d (64 → 32)	64×16×16	32×32×32	4×4	2	1
Conv2d (32 → 32)	32×32×32	32×32×32	3×3	1	1
ConvTranspose2d (32 → 32)	32×32×32	32×64×64	4×4	2	1
ConvTranspose2d (32 → 1)	32×64×64	1×128×128	4×4	2	1

Figure 31 – Convolutional Autoencoder layers. [1]

[17] *Note: the correct link may be different, in case it will be possible to find it from my github repository https://github.com/Gianmarco-San/.*

6.3. Machine

Ultron computer details:

- NVIDIA-SMI 535.161.07
- Driver Version: 535.161.07
- CUDA Version: 12.2
- GPU Name: 0 NVIDIA TITAN RTX
- GPU Name: 1 NVIDIA TITAN RTX
- GPUs details: example of CUDA device 1, at (gsEnv) gsantoro@ultron:~/DeepFake-Detection$, Sun Apr 21 15:14:14 2024, nvidia-smi displayed:
    - Fan: 55%
    - Temp: 72C
    - Perf: P2
    - Pwr:Usage/Cap: 189W / 280W
    - Memory-Usage: 3453MiB / 24576MiB
    - GPU-Util: 44%
    - Compute M.: Default