Deep Fake Localization

Andriciuc Andreea-Cristina, Moroșan Eric-Alexandru and Oneață Elisabeta*

University of Bucharest, Romania

*Bitdefender, Romania

andreea-cristina.andriciuc@s.unibuc.ro, morosan.eric2002@gmail.com and eoneata@bitdefender.com

1. Introduction

1. **Short Description**
   The quality at which images are generated and altered is astounding, and it is getting more and harder for people to tell the difference between real and fake. Besides recognizing an image as real or fake, we should also seek to find out which parts are counterfeit in these abnormal images we are identifying. In this regard, there were multiple neural networks specially developed to better classify between fake and real images and even find the subtle artifacts which make these images doctored.

2. **Objective**
   In our project, we seek to better understand the manner in which images are generated so different methods can find the traces of the manipulation behind the pixels with both the training on the same generative model or distinct ones.

2. Dataset and Preprocessing

Dataset Description:
The dataset consists of 9k modified images from the **CelebA-HQ dataset**. The CelebA-HQ dataset has 30k images from the CelebA dataset which consists of large-scale face attributes images of celebrities. The modified images come from 4 different image modifier methods: **Repaint** (the images crawled from Flickr and automatically aligned but up to 15 pixels), **LAMA** (large-mask inpainting model), **Pluralistic** (image completion model), **LDM** (latent diffusion model). All the images followed the same preprocessing steps (input resolution, reshape) such that there is no bias for the classifier model to take in consideration.
For any of these images, the dataset also offers a mask which highlights the artifacts that make the image fake.

3. Models

We chose a U-Net architecture, widely used for image segmentation tasks. It has a symmetrical structure:

1. **Encoder (Contracting Path)**: progressively downsample the input image, capturing features at different scales.

2. **Decoder (Expanding Path)**: upsamples the feature maps, combining them with skip connections from the encoder to recover spatial details.

Baseline: We created a U-Net architecture having the following layers, in order:

- convolutional block (consisting of 2 x Conv2D + BatchNorm2D + ReLU, 0.2 Dropout at the end), followed by Maxpool
- second convolutional block, same as first + Maxpool
- third convolutional block, same as second
- first Up block (consisting of Upsample + Conv2D + BathNorm2D + ReLU), followed by concat with the second conv block for connection skipping
- second up block + concat with the first conv block
- third up block + final Conv2D layer

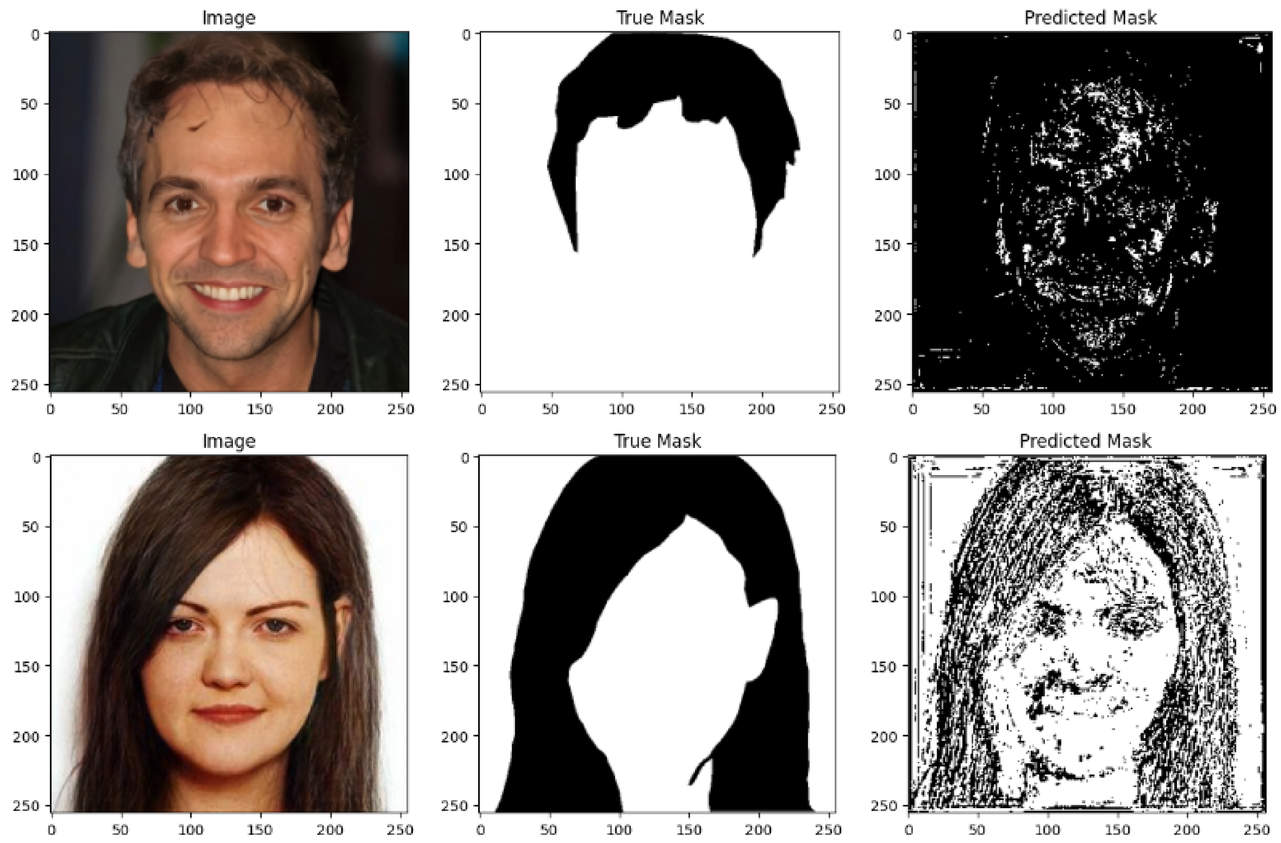
We used *Dice Loss Function*, as it is specific for binary segmentation.

4. Results

Our evaluation metric was *Intersection over Union*, and we obtained the following results:

Train on	Test on	IOU
Repaint	Repaint	0.2344
LAMA	LAMA	0.4792
Pluralistic	Pluralistic	0.7557
LDM	LDM	0.1299

Train on	Test on	IOU
Repaint + LAMA + LDM	Pluralistic	0.0064
Repaint + Pluralistic + LDM	LAMA	0.6674
LAMA + Pluralistic + LDM	Repaint	0.3832
Pluralistic + LAMA + Repaint	LDM	0.6599



5. Conclusion

Although we didn't get a high accuracy based on the masks that we had been given for a various of reasons such as a high volume dataset which takes a lot of time to be loaded and run through every model, the limitations of Google Colab and Kaggle resources (lots of crashes, insufficient free resources), small number for epochs for training (also because of the excessive time), a similarity between our results and the perfect ones is observable. Our model seems to struggle to find the entire sought area, but it succeeds in finding the outline in many cases. It is clear that the model better identifies the mask when the dataset used for the training is generated from the same model as the dataset used for testing. This fact shows that specific image are easier to be labeled as fake and divided in real and fake zones when we know what generative model may behind the photo we see.