Deepfake Localization

Alexandru Pascu, Alexandru Vișan and Elisabeta Oneață*
University of Bucharest, Romania
*Bitdefender, Romania

pascuionutalexandru@gmail.com, alexandruvisan44@yahoo.com, eoneata@bitdefender.com

UNIVERSITY OF BUCHAREST
VIRTUTE ET SAPIENTIA

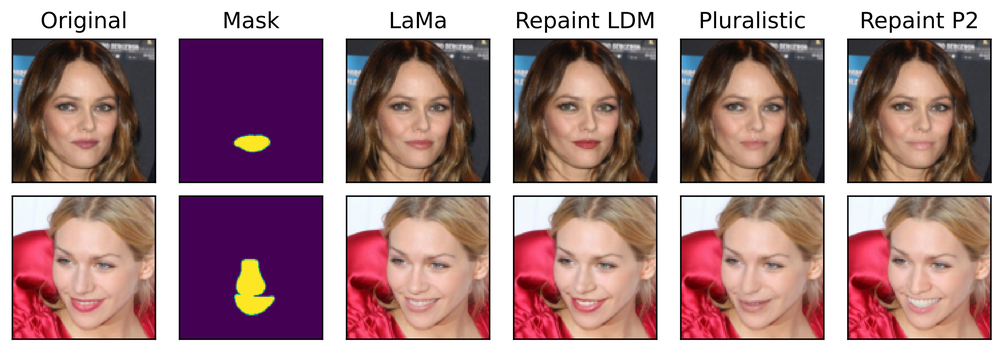
1. Introduction

- This project focuses on the development of deepfake localization techniques to detect and identify manipulated areas within images.
- The main challenges involve designing models that can generalize across manipulations produced by different generators, and managing the significant computational demands required for training robust models.
- Our approach tests various architectures to optimize performance and generalization, addressing critical needs in digital media authentication.

2. Dataset

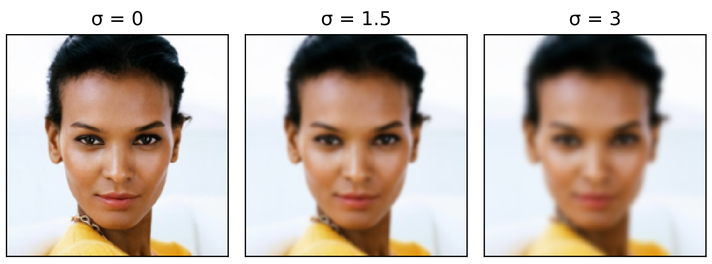
In order to train and evaluate our models, we use the locally manipulated datasets described in [4]. The source dataset is CelebA-HQ [1], consisting of 30K images that were selected and processed from the CelebA dataset. We train our models in a fully supervised setting, so we use both the locally manipulated images and the ground-truth localization masks

Local manipulations were created using four generators:



3. Data Augmentation

It has been shown that adding Gaussian blur to the dataset improves model generalization across generators [3].
We augment our dataset using Gaussian blur with 50% probability. Images are blurred with $\sigma \sim Uniform[0, 3]$.



4. Models

Architectures Tested: We have used segmentation models from Segmentation Models PyTorch repository[2], which provide flexible modular architectures consisting of two main components: a backbone and an encoder. Our approach consisted of trying different backbones (FPN, Unet, Unet++) with encoders (Resnet variants, Inception).

Loss Functions: We have experimented with different loss functions, such as Dice loss, Focal loss, and Jaccard loss; Jaccard loss provided superior performance due to its relevance to the IOU metric.

Training Details: All models were trained with a learning rate of 3×10^{-5} over 15 epochs, focusing on optimizing localization accuracy. The primary objective of the training was to maximize the Intersection Over Union (IOU), a critical performance metric for evaluating the precision of localization. The constraint to 15 epochs was imposed by limitations in computational resources, as training the larger models required upwards of six hours per iteration.

Optimal Configuration: Optimal results were obtained using the Unet++ architecture coupled with the Resnet-50 encoder. This configuration outperformed the next most effective model by several percentage points, demonstrating its superior efficacy in localization accuracy.

5. Results

The results from our image manipulation detection experiments are provided below. Table 1 shows high performance for models trained and tested on the same generator, such as LaMa and Pluralistic, with IOU scores around 0.90. Conversely, Table 2 illustrates a significant drop in performance when models are trained on three generators and tested on a fourth, highlighting the difficulties in generalizing across varied manipulation techniques.

Train on	Test on	Performance (IOU)
LaMa	LaMa	0.90
Repaint LDM	Repaint LDM	0.51
Pluralistic	Pluralistic	0.90
Repaint P2	Repaint P2	0.74

Table 1: Training and testing on the same generator

Train on	Test on	Performance (IOU)
w/o LaMa	LaMa	0.19
w/o Repaint LDM	Repaint LDM	0.24
w/o Pluralistic	Pluralistic	0.37
w/o Repaint P2	Repaint P2	0.27

Table 2: Training on 3 generators and testing on the 4th

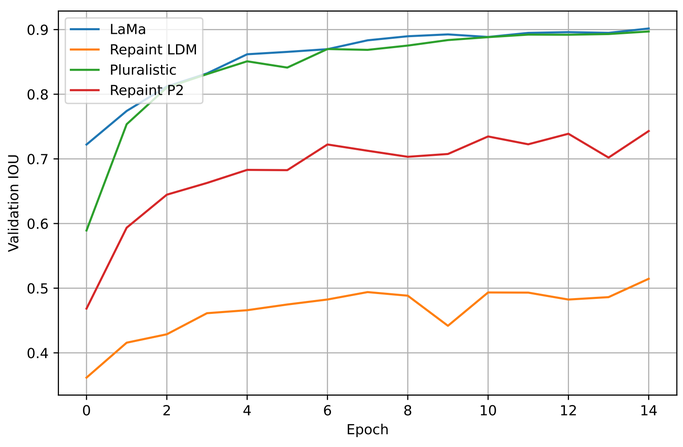


Figure 1: Validation for training on the same generator

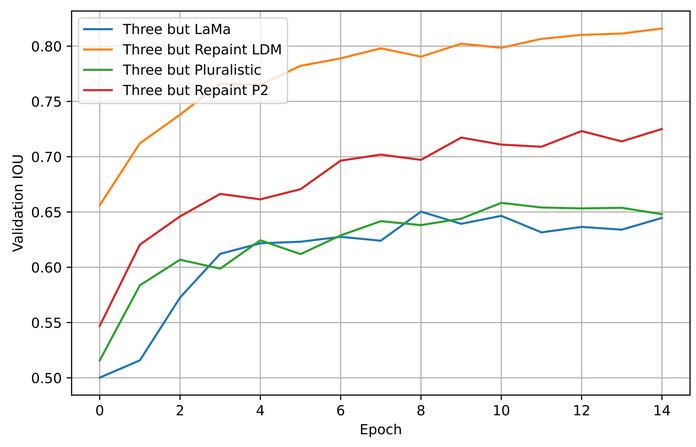


Figure 2: Validation for training on three generators

The analysis highlights certain interesting findings. The LDM generator demonstrated a low IOU of 0.51, indicating that traces of the latent manipulation are more difficult to detect. Additionally, evaluations on the LaMa dataset without prior training resulted in a poor IOU of 0.19, suggesting that algorithmic differences matter for generalization.

6. References

References

[1] Tero Karras et al. *Progressive Growing of GANs for Improved Quality, Stability, and Variation*. 2017. eprint: `arXiv:1710.10196`.

[2] Pavel Iakubovskii. *Segmentation Models Pytorch*. `https://github.com/qubvel/segmentation_models.pytorch`. 2019.

[3] Sheng-Yu Wang et al. *CNN-generated images are surprisingly easy to spot... for now*. 2019. eprint: `arXiv:1912.11035`.

[4] Dragos Tantaru, Elisabeta Oneata, and Dan Oneata. *Weakly-supervised deepfake localization in diffusion-generated images*. 2023. eprint: `arXiv:2311.04584`.

7. Conclusion

- Our study emphasizes the challenges of generalization across different generators for deepfake localization.
- Results consistently favor the Unet++ architecture paired with a Resnet-50 encoder as the most effective combination.
- Localization of manipulations in the latent space is inaccurate, even when training on the same generator.