FACEAUTH

REAL AND AI GENERATED IMAGE CLASSIFICATION

Kush, Udbhav
Ukush4@gwu.edu

Table of Contents

Abstract

In the rapidly evolving landscape of artificial intelligence, the need to discern between human-generated and AI-generated images, particularly faces, has become paramount. This project addresses this challenge by developing an image classification model specifically designed to distinguish between authentic human faces and those generated by advanced AI techniques. Additionally, an auxiliary classification model is proposed to identify the specific AI architecture responsible for generating the image, including cutting-edge models such as DALL-E, Imagen, and others.

This research aims to fill the gap by proposing models for the discrimination of AI-generated faces from genuine human faces and identifying the underlying AI architecture. The development of such models holds significance in addressing the rising concerns associated with the misuse of AI-generated content, especially in the realm of deepfakes. The outcomes of this project can contribute to the advancement of AI ethics, digital forensics, and the mitigation of potential societal challenges arising from the proliferation of manipulated visual content.

Data

Even though there exist various data sources on Kaggle and other sources we wanted to make sure that the sources we chose explicitly tell us what models were used to generate these images, moreover we wanted at least 20k images from each source to make sure we have enough data. Moreover, since we are classifying between fake and real, we need both fake and real human faces.

Datasets Used For Real Faces:

1. Celeb-HQ-2
2. Wiki Celeb

There are fairly clean datasets both scraped from the internet and containing mostly celebrity photos, this gives us some surety that the collection of these datasets did not involve any violation of privacy. This was also the reason we did not end up using the Flicker dataset, even though the data publisher ensured that the images were extracted legally. Moreover, both these datasets contain high-resolution color images but since training on high-res images would be very resource-consuming we decided to settle on a resolution of 256x256 pixels, and all these images were cropped to face so we got away by not doing much pre-processing.

Datasets Used For Fake Faces:

1. 1millon fake faces: This dataset contains a million fake faces generated by the StyleGan model developed by Nvidia.
2. iFakeFaceDB: This dataset is a collection of 87000 images which are again generated by the styleGan model but are transformed with the GANprintR approach. GANprintR is a transformation to remove the GAN fingerprint from the images (Refer to paper for more details).
3. DeepFakeFace: This dataset contains fake face images which are generated by diffusion-based models. This dataset contains images generated by the Stable Diffusion Impainting model, Insightface toolbox, and Stable Diffusion V1.5.

These datasets were chosen as we thought they encompass both GAN and Diffusion-based models.

Splits

We did a standard split of training, test, and dev. The training set was used for training the model and the test set was used to make decisions while training the model. The dev set was untouched and only evaluated right before the project presentation. After splitting the dataset in training, test, and dev, we had 120,000 images to train our model.

Moreover, we ensured the following conditions:

- An equal number of fake and real images.
- If more than one fake dataset was used to train the model they have an equal number of images from each fake dataset.
- The Dev set is not used until the day of submission to get the results.

Using Discriminator of a GAN Model

Our initial idea was to first find the state-of-the-art GAN models that generate realistic human faces and then use the Discriminator section of the model to detect fake faces from the real ones. This was a very naive approach, upon reading the literature about training GAN-based models we realized that usually a GAN is trained to a saddle point where the Discriminator can no longer differentiate between the real and generated images, this by itself would make the discriminator not so useful for our purposes. But we held onto the idea as we thought we could further train the discriminator to do our job.

We landed on two GAN models namely StyleGAN3 and StyleSwin but both these models were very challenging to work with since we could not dissect the published code to use the discriminator for our purpose.

Hence after much deliberation, we made a team decision to use other models due to the time constrain of the project.

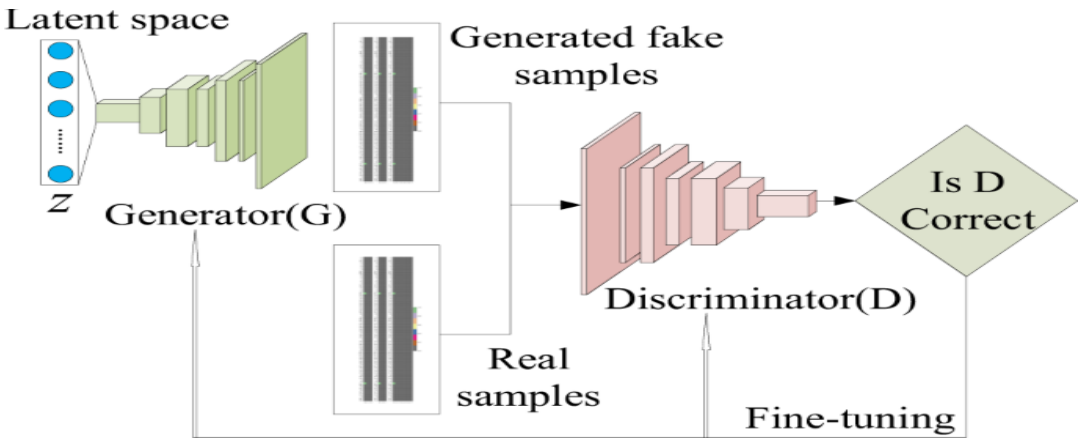

Figure 1. GAN Architecture

Transformer

During our literature review, we found one paper. The model gave us an understanding of the Vision Transformer and its architecture. On further research, we found a pre-trained transformer model called vit-basepatch16-224 (https://huggingface.co/google/vit-base-patch16-224). The model was the most trending model on huggingface for the image classification task. On further studying the architecture, we found that the model has a transformer base with a linear layer as the classification head. We trained this model on our dataset and the model gave very good results on our test set.

Breaking down the architecture of Google's ViT, the image is fed into the model of fixed-size patches of 16x16 that are linearly embedded. The linear embedding of the image along with positional embedding of the image are fed to the transformer encoder. The transformer head is connected to the MLP head to classify images into different classes. Since the model is transformer-based, we use the concept of positional embedding (like it is done for textual data) along with the [CLS] token which is taken as the image representation.

```
def __init__(self, list_IDs, type_data):
    self.type_data = type_data
    self.list_IDs = list_IDs
    self.processor = AutoImageProcessor.from_pretrained("google/vit-base-patch16-224")
```

Figure 2. Transformer Image Processor

In the code above, we define the processor object of the model. This processor takes care of all the

preprocessing required for the image. We just need to pass the image to the processor object. Linear embedding, positional embedding, and [CLS] token, all are taken care of to pass to the model to inference and train our model.

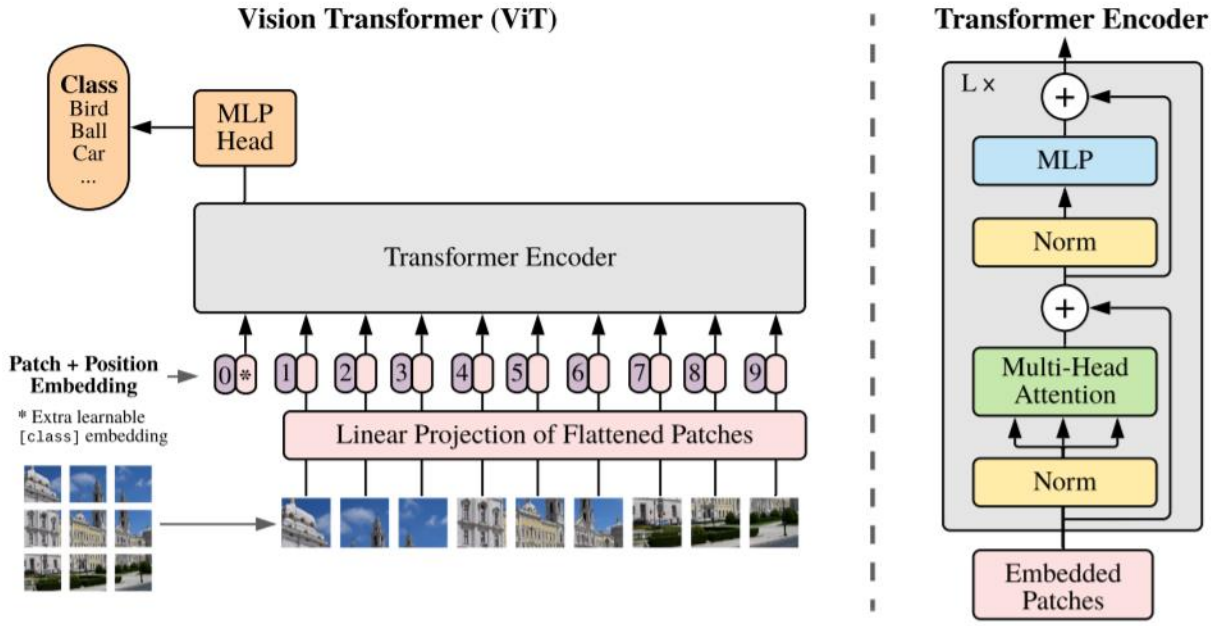

Figure 3. Architecture of the Vision Transformer

Results

Table 1. Results of Google's ViT model

Model	Accuracy	Precision	Recall	AUROC	F1Score
ViT	0.83616	0.84791	0.84919	0.83501	0.84885
ViT_diffusion	0.75791	0.75311	0.82134	0.75232	0.78575
ViT_GAN	0.56116	0.55192	0.9995	0.52254	0.71115
ViT_GANPrintR	0.66407	0.6167	1	0.63447	0.76291

Google's ViT models performed the best on the dev split dataset. As mentioned in the training section of the group report, each model was trained on the complete dataset (equal proportions of fake images from each model) and individual fake images dataset too. Google ViT generalized the best among all the models tried for this project.

Summary

This project focuses on developing an image classification model to distinguish between human-generated and AI-generated faces. The Vision Transformer (ViT) model, particularly vit-basepatch16-224, emerged as the most effective choice among various models. Utilizing datasets like Celeb-HQ-2 and iFakeFaceDB, the

project aimed to address concerns related to AI-generated content misuse, particularly in deepfake scenarios. ViT demonstrated superior performance in accuracy, precision, recall, AUROC, and F1Score, showcasing its potential for AI ethics, digital forensics, and mitigating societal challenges associated with manipulated visual content.

Conclusion

In conclusion, the project successfully addresses the challenge of discerning AI-generated faces from genuine human faces. The ViT model, with its transformer architecture, proved to be effective in achieving high accuracy and generalization. The outcomes of this project hold significance in the context of AI ethics, digital forensics, and addressing concerns related to the misuse of AI-generated content, especially deepfakes. The proposed models contribute to advancing the field and mitigating potential societal challenges arising from manipulated visual content.

Percentage of Code lines copied from the internet: (646-131)/646 = 0.79721 or 79%.

References

1. Zhang, B., Gu, S., Zhang, B., Bao, J., Chen, D., Wen, F., Wang, Y., & Guo, B. (2022, July 21). *StyleSwin: Transformer-based gan for high-resolution image generation*. arXiv.org. https://arxiv.org/abs/2112.10762
2. Tsang, S.-H. (2022, August 14). *Review-DEIT: Data Efficient Image Transformer*. Medium. https://sh-tsang.medium.com/review-deit-data-efficient-image-transformer-b5b6ee5357d0#:~:text=DeiT%20has%20the%20same%20architecture,is%20a%20way%20to%20train
3. Tsang, S.-H. (2022, August 14). *Review-DEIT: Data Efficient Image Transformer*. Medium. https://sh-tsang.medium.com/review-deit-data-efficient-image-transformer-b5b6ee5357d0#:~:text=DeiT%20has%20the%20same%20architecture,is%20a%20way%20to%20train
4. *Google/VIT-base-patch16-224 · hugging face*. google/vit-base-patch16-224 · Hugging Face. (n.d.). https://huggingface.co/google/vit-base-patch16-224
5. *Papers with code - ifakefacedb dataset*. Dataset | Papers With Code. (n.d.). https://paperswithcode.com/dataset/ifakefacedb
6. OpenRL. (n.d.). *OpenRL/deepfakeface · datasets at hugging face*. OpenRL/DeepFakeFace · Datasets at Hugging Face. https://huggingface.co/datasets/OpenRL/DeepFakeFace