IDS 705 Final Report

Detecting AI-Generated Fake Images

Towards a Generalizable Fake Facial Image Detector via Neural Networks

Team Members:
Kabber, Pooja
Lai, Scott
Oh, Song Young
Wang, Emma

Team Number: 5

Abstract

With the steady rise in innovative new deep learning techniques, the detection of AI-generated fake images has become a critical research problem. In this project, we aim to apply a convolutional neural network based model to detect a particular kind of fakes - facial images. This idea was developed from previous studies done in this space, including Sabithaa et al. (2020) and Zhang et al. (2020). We use the current state-of-the-art pre-trained ResNet model to address vanishing gradients in deep neural networks, as proposed by He et al. (2016). We also run experiments using the light-weight pre-trained MobileNetV2. Our models are fine-tuned and evaluated using the FFHQ (Flickr-Faces-HQ) dataset as the benchmark. Through our experimentation, we achieve high accuracy in detecting fake images, and analyze the impact of different categories on model performance, obtaining an overall accuracy of 80% with MobileNetV2 and an accuracy of 89% with ResNet50. Our findings can be applied to real-world applications where fake image detection is critical.

Introduction

Image-generation AI models have made significant progress in recent years, particularly with the advances in neural network techniques. This has led to the development of several impressive models including DALL-E 2 developed by OpenAI. DALL-E 2 is a transformer autoregressive model in which the model learns to map the input text to a latent space representation, which is then used to generate a corresponding image. Thus, DALL-E 2 can be effectively used for fake image generation through textual descriptions. DALL-E 2's ability to generate high-quality images has significant implications for a wide range of applications, including entertainment, advertising, digital art, and even architecture design.

However, the use of AI-generated images raises important questions about the ethical implications of using such images. For example, AI-generated images can be misused to create fake images or videos of individuals, which may potentially lead to serious consequences such as identity theft or reputational harm. For this reason, this project aims to build a machine learning model that can differentiate between AI-generated images created by DALL-E 2 and real human face images. Our project is driven by the potential ethical implications of using AI-generated images in various contexts, such as deep fakes and racial bias. By uncovering valuable insights into these differences, we hope to contribute to this significant area of study and promote ethical practices in the use of AI-generated images.

Background

The detection of AI-generated images is crucial due to the potential for their use in various malicious activities, including deep fake videos, synthetic identity fraud, propaganda, and art fraud. However, the increasing sophistication and realism of AI-generated images have made it challenging for humans to

distinguish between real and fake images. To explore the potential for fraud detection in images, we are using DALLE-2 to generate fake images from our real image dataset in our study, given its advancement in generating images that are difficult to detect as fraud by human eyes.

Deep fake videos are a compelling example of the misuse of AI-generated images. These videos manipulate existing footage using AI technology to replace the face of one person with another, creating fake news, spreading propaganda, and damaging the reputation of individuals. Similarly, AI-generated images can also be used for synthetic identity fraud, where criminals create fake identities using stolen personal information and AI-generated images to commit fraudulent activities like opening bank accounts or applying for loans. Additionally, the recent controversy surrounding AI-generated artwork in Taiwan has highlighted the potential for fraud in art sales. This debate centers on the use of computer-generated tools to create art and the concerns that this undermines the work of traditional artists who rely solely on their own skills and expertise. As AI-generated images become increasingly sophisticated, it is becoming more challenging for the public to differentiate between real and fake artwork, creating opportunities for fraud.

Our research aims to develop a machine learning pipeline that can detect fraud in images, thereby safeguarding against the negative consequences of AI-generated images. By gaining a deeper understanding of the differences between AI-generated facial images and real human faces, we hope to contribute to the development of more responsible and ethical practices for using AI-generated images across different fields.

In the field of computer vision research, there has been a surge of publications related to DALL-E 2 over the past few months. These papers have garnered an increasing amount of attention due to the model's impressive capabilities and its potential applications in various fields. In one recent study, Sha et al. (2022) proposed a novel method for detecting and attributing fake images. The authors conducted extensive experiments on four popular text-to-image generation models, namely DALL-E 2, Stable Diffusion, GLIDE, and Latent Diffusion, using two benchmark prompt-image datasets. The analysis results revealed that fake images generated by various models can be distinguished from real ones based on a common artifact shared by fake images from different models.

Another paper by Ojha et al. (2023) addressed the problem of training neural networks to detect fake images and evaluated recent text-to-image generation models, including Latent Diffusion, GLIDE, and DALL-E. The results suggested that there is something common between fake images generated from a GAN and those from a diffusion model, although the specific similarity remains an open question. Meanwhile, Corvi et al. (2022) studied the performance of current detectors, developed for GAN-generated images, on twelve kinds of new synthetic images, including DALL-E 2 images. The study particularly focused on challenging social-networks scenarios involving image compression and resizing.

Title	Model Used	Accuracy	Architecture	Dataset	Code/Source
Enhanced model for fake image detection (EMFID) using convolutional neural networks with histogram and wavelet based feature extractions	Deep convolutional neural network (CNN) designed to detect fake images using a combination of histogram and wavelet-based feature extraction.	96.45%	The model consists of multiple convolutional and max pooling layers, followed by fully connected layers and a softmax output layer that produces a probability distribution over the two classes (fake and real).	Real: FFHQ	Sabithaa et al. (2020)
Deep residual Learning for Image Recognition	ResNet	The authors evaluated ResNet on the ImageNet dataset and achieved state-of-the–art performance, with a top-5 error rate of 3.57% for ResNet-152.	The authors experimented with various depths of ResNet, from 18 layers to 152 layers, and found that increasing the depth beyond a certain point led to diminishing returns in performance.	ImageNet dataset https://www.image-net.org/download.php	https://github.com/pytorch/vision/blob/main/torchvision/models/resnet.py
DE-FAKE: Detection and Attribution of Fake Images Generated by Text-to-Image Generation Models	ResNet18	0.871	Their 18-layer ResNet has no extra parameter compared to its plain counterpart.	- Real: MSCOCO (20,000 images) - Fake: Generated 20,000 images by using the prompts of real images through Stable Diffusion	Kaiming He, et al. (2016)
On the Detection of Synthetic Images Generated by Diffusion Models	ResNet50	0.595	block = BottleNeck layers = [3,4,6,3] num_classes = 1000 zero_init_residual = False	ProGAN images	S.-Y. Wang, et al. (2020) https://github.com/peterwang512/CNNDetection
Towards Universal Fake Image Detectors that Generalize Across Generative Models	ResNet50	0.8425	They vary the voting pool size from k=1 to k=9. More details will be released soon.	ProGAN images	https://github.com/Yuheng-Li/UniversalFakeDetect (Data has not been released yet)
Detecting Computer Generated Images with Deep Convolutional Neural Networks	ResNet50	0.941	They transfer the weights of layers pre-trained on ImageNet data to their deep model, using a transfer learning approach.	Public dataset comprising of 9,700 images	N/A
Distinguishing Computer-Generated Graphics from Natural Images Based on Sensor Pattern Noise and Deep Learning	CNN	0.9337 (50 epochs) 0.9312 (80 epochs)	Each of five convolutional layers is followed by a batch normalization layer, a ReLU layer, and an average pooling layer.	- Real: RAISE dataset (1,800 images) - Fake: Level-design reference database (1,800 images)	N/A

Multi-attentional Deepfake Detection	EfficientNet-b4	0.976 (high-quality) 0.8869 (low-quality)	in_channels = 3 out_channels = 32	- FaceForensics++ - DFDC	https://github.com/yoctta/multiple-attention
Face Forensics++: Learning to Detect Manipulated Facial Images	XceptionNet	0.701	They replace the final FC layer with two outputs and the other layers are initialized with the ImageNet weights.	FaceForensics++	Francois Chollet (2017)

Table 1. Comparison of Related Work

Data

The dataset we use for our experiments consists of two classes: real and fake, the latter of which is generated by Dall-E 2 from the auto-generation pipeline we build. For the real dataset, we adopt the Flickr-Faces-HQ (FFHQ) dataset, a high-quality collection of 70,000 PNG images at 1024×1024 resolution. These images were crawled from their google drive to Amazon S3, a cloud-based storage service provided by AWS. This dataset comprises real human images with considerable variation in terms of age, ethnicity, and image background, and it also includes a broad coverage of accessories such as eyeglasses, sunglasses, hats, and more. Since our data was originally unlabeled with respect to these categories and human annotation is time-consuming, we annotate our images using machine learning models in cloud-based development environments such as GitHub Codespaces and Google Colab. Meanwhile, to ensure fair comparisons, the fake images in our dataset were also set to a resolution of 1024×1024, matching the resolution of the real images. We also generate corresponding fake images for our set of real images to ensure that the learning of the model is to differentiate between real and fake images and not to separate the images using some other criteria.

Data Annotation

To ensure that our dataset covers a diverse range of human faces, we employed four pre-trained classification models for the following categories: facial expression (Trpakov, 2022), race (Parkhi et al.), gender (Parkhi et al.), and glasses (Mandgi, 2021). We obtained the facial expression model from HuggingFace, which recognizes seven distinct emotions. For more convenient analysis and adequate representation of all classes, we converted the seven expressions to the positive, negative and neutral scale. For glasses detection, we utilized a binary classification approach that involved cropping each image based on facial landmark coordinates for the nasal area and converting the image into black and white to identify the presence of a strip of black pixels (0s) on the face. For race and gender, we used the VGG-Face model pre-trained on ImageNet. The race categories were asian, white, middle eastern, indian, latino and black. We absorbed middle eastern and indian into asian since they were not represented adequately individually. Gender has two categories: male and female.

These labeling results were then aggregated into a single data frame, with one column for the image file name and four columns for the labeling results. This rigorous process allowed us to create a more

representative dataset by selecting a sample of sample size 21k from the 70k real images such that each category and class were adequately represented and in a comparable ratio to each other. The final ratios for the categories and classes are as follows:

Facial Expression	Positive	0.52
	Negative	0.25
	Neutral	0.23
Glasses	Glasses	0.67
	No Glasses	0.33
Gender	Man	0.59
	Woman	0.41
Race	White	0.31
	Asian	0.3
	Latino Hispanic	0.3
	Black	0.11

Table 2. Ratios of classes for all facial categories

Another reason for selecting these four facial characteristics is because we aim to analyze if images belonging to these categories may be more challenging for DALLE-2 to generate vivid images, thereby making it easier for our model to detect fakes, for example, a face with eyeglasses. Our objective is to investigate whether other facial features would impact its performance (gender, glasses, etc.). Nonetheless, it is important to note that our annotations only cover some types of human faces and do not represent all categories.

During the annotation process, we manually labeled a few hundred images to estimate the performance of the models. We observed that the facial expression model had an accuracy of approximately 70%. Meanwhile, the models for detecting faces wearing glasses and classifying gender were more accurate, with an accuracy of 91%. Additionally, the model for race classification had an accuracy of 86% and around 96% for gender. Although some images were misclassified, we still wanted to maintain the machine labeling results due to limited time and the primary aim of our data annotation, which was to ensure we have enough images in many facial categories. Nevertheless, we acknowledge that our annotations may not be completely accurate and this can affect the efficacy of our analysis of the detector’s performance.

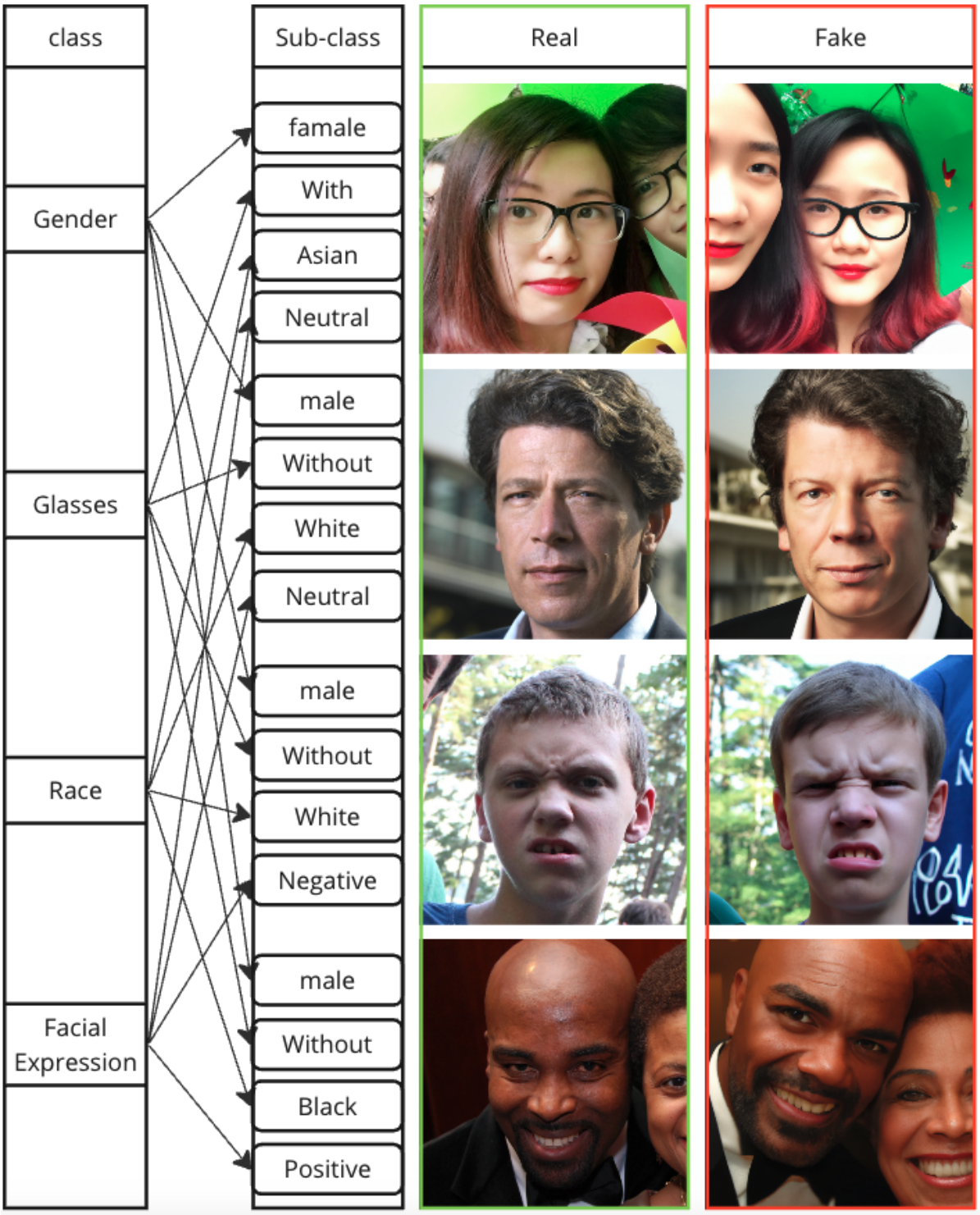

Figure 1. Characteristics for Real Image vs Fake Image. The first column show the four class that we use to do the picture classification: Gender, Glasses, Race and Facial Expression; Second column contains the sub-classes for each class; third column contains the real image we use for AI image generation; the last column is the fake images generate from DallE-2.

Pipeline of Preprocessing

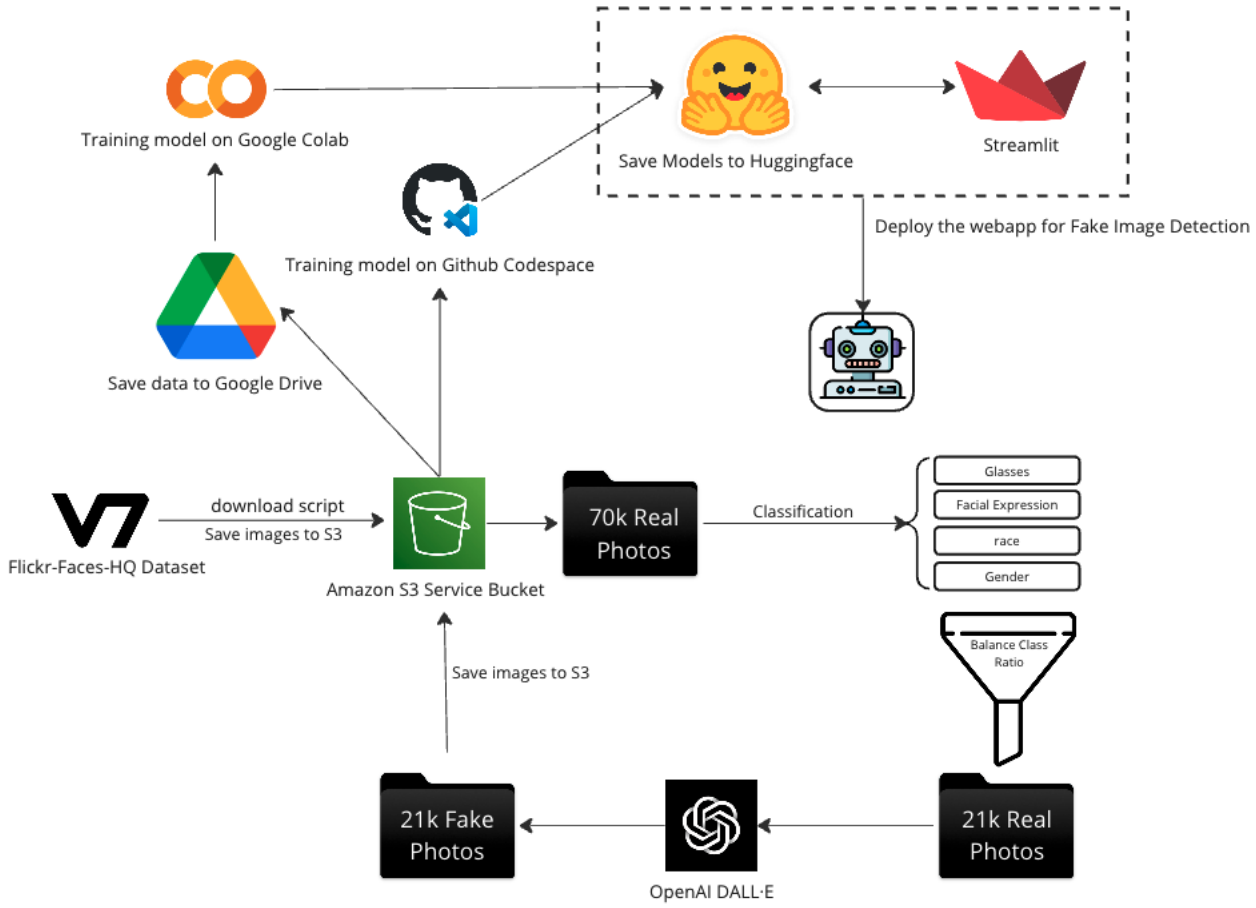

Figure 2. Workflow diagram

To ensure a balanced dataset, we performed four separate classification analyses on the original dataset of 70,000 images. The images were classified into classes for four categories: glasses, facial expressions, race, and gender. Using the number of images in each category, we randomly selected a total of 21,000 images from across all four categories based on the class weight for each class. This approach allowed us to create a well-balanced dataset for training our models.

The AI images were generated using OpenAI's DALL-E 2 model, ensuring consistency with the real images in our dataset. We believe that this may allow us to ensure that the learning of the model is in fact the difference between real and fake images and not differences in other characteristics. To facilitate efficient model training, we utilized both Google Colab and CodeSpaces with AWS S3 to run multiple models, and stored both the real and fake images in Google Drive and the AWS S3 bucket. We also made the data publicly available in Google Drive, to be used by the community for other research.

After training the models, we saved them on Hugging Face, a popular platform for sharing and deploying NLP and ML models. To enable easy access and utilization of our models, we built a web

application using Streamlit and Hugging Face. This application allows users to detect fake images and classify them into the four categories we identified during the selection process.

In conclusion, our preprocessing pipeline was carefully designed to ensure a balanced dataset and efficient model training. We believe that open access to data and models is crucial for advancing research and enhancing model accuracy. Therefore, we are excited to make our dataset and models available to the research community to facilitate further studies in this field.

Experiments

Project Architecture

Our project aims to build a robust detector of fake images with the following workflow. First, we acquire 70,000 images from the FFHQ dataset and formulate our dataset which is stored in AWS S3. We then annotate the 70,000 images, including attributes such as facial expression, wearing glasses, races, and gender. We select 21k real images and build an image-to-image pipeline that generates a fake dataset of 21k images. We then perform image preprocessing (ensuring RGB color model and resizing the images to 180x180 for input to the model), and feature extraction, followed by a model training process. The classification will be done using two types of neural network models MobileNetV2 and ResNet50, and the results will be evaluated.

Methodology

Our work is intended to build a neural network model to detect fake AI images, based on previous work done including Sabithaa et al. (2020), Zhang et al. (2020), and He et al. (2016). Sabithaa et al. (2020) used an enhanced model for detecting fake images generated by artificial intelligence using convolutional neural networks with histogram and wavelet-based feature extractions. In our study, the real dataset we used is the same as one of their evaluation datasets, FFHQ dataset, which would be a good reference for our model building process. Zhang et al. (2020) also adopted a CNN-based architecture for detecting fake images. We particularly paid attention to He et al. (2016) which proposed the ResNet model developed for addressing the problem of vanishing gradients in deep neural networks. Our study can fine-tune their pre-trained model and visualize outputs to gain insights into the model's predictions. This visualization would be highly useful to our study, because we aim to achieve high accuracy while understanding the factors influencing detection performance. Based on this research, we will be using ResNet50. We also chose to use MobileNetV2 since it is a lightweight version of ResNet50, and to explore its performance as compared to ResNet50.

MobileNetV2 (Visualization displayed in Appendix 1.)

MobileNetV2 model, as a pre-trained base model for transfer learning, is a deep learning architecture that is designed for mobile devices and has a small computational footprint. The model was trained on the ImageNet dataset, which has over 1 million images with 1000 different classes. The model has been shown to achieve high accuracy on image classification tasks while requiring fewer parameters and computation compared to other popular architectures. We adopted the pre-trained MobileNetV2 model with its weights trained on the ImageNet dataset, and the top layer of the model is excluded by setting include_top to False. This allows us to reuse the feature extraction capabilities of the model while adding our custom classification layer on top.
During the training process, we freeze all layers in the base model, so only the newly added classification layers are trainable during training. The added layers include a data augmentation layer to artificially increase the dataset's size and prevent overfitting, the base MobileNetV2 model, followed by a Global Average Pooling layer, and three fully connected layers with ReLU activation, Dropout layers, and L2 regularization. The final layer has a sigmoid activation function since this is a binary classification task.

The model is compiled with the Adam optimizer, binary cross-entropy loss function, and accuracy metric. The goal is to train the model to classify images into two classes, fake and real, with high accuracy, more specifically high score in precision of fake class images.

Given the limited computation power, the fine-tuning process was done using a subset of 5300 images. The two hyperparameters that are tuned are dropout rate (0.05,0.1,0.2) and learning rate (1e-4, 5e-4, 1e-3), the best set of parameters turns out to be dropout rate = 0.05 and learning rate = 5e-4. After the model was fine-tuned, it was trained again using the large dataset of 21439 images.

Learning Rate	**Dropout Rate**	**Accuracy**	**Loss**
0.0001	0.05	0.7826	0.4993
0.0005	0.05	0.7948	0.4943
0.001	0.05	0.7892	0.4937
0.0001	0.1	0.7771	0.5116
0.0005	0.1	0.7910	0.4944
0.001	0.1	0.7826	0.5066
0.0001	0.2	0.7715	0.5198
0.0005	0.2	0.7799	0.5028
0.001	0.2	0.7817	0.4897

Table 3. Results from hyperparameter tuning for MobileNetV2.

ResNet50 (Visualization displayed in Appendix 1.)

We used the ResNet50 model from the library Keras, pre-trained on the ImageNet dataset. We removed the input and output layers and froze the rest of the weights since re-training the whole network would be time-consuming. We then added an input layer of size (180, 180, 3). To the end of the network, we added a Dense layer of size 512 and activation ReLU and an output Dense layer of size 1 and activation Sigmoid, since this is a binary classification problem. We also chose to use the Adam optimizer and Binary Crossentropy as our loss function. We chose our batch size to be 300.

To get the best values for the hyperparameters learning rate and number of epochs, we performed hyperparameter tuning using GridSearch. Since the training time was high for the full dataset, we performed hyperparameter tuning on a subset of the data. For this process, we used the same ratio of training and validation data with 4k training images and 1k validation images.

Learning Rate	No. of Epochs	Accuracy	Precision	Recall
0.01	10	0.85	0.93	0.74
0.001	10	0.86	0.91	0.8
0.0001	10	0.87	0.86	0.89
0.01	20	0.88	0.88	0.88
0.001	20	0.88	0.88	0.89
0.0001	20	0.88	0.87	0.9
0.01	50	0.88	0.87	0.9

Table 4. Results for hyperparameter tuning for ResNet50.

We limited hyperparameter tuning to the above hyperparameters and their values due to time

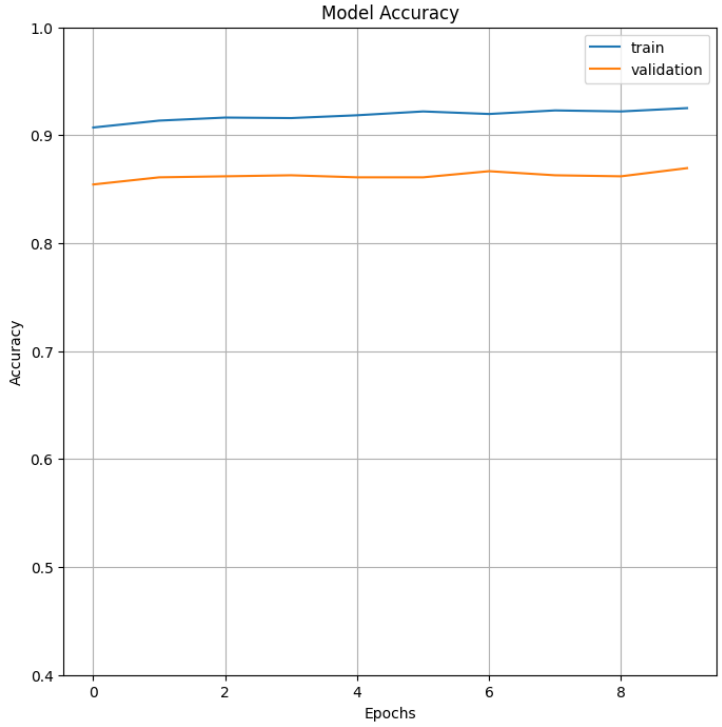

Figure 3. Accuracy vs. Epochs for ResNet50 with no. of epochs of 10 and learning rate smaller than 0.01.

constraints and large training times. We plan to include batch size, additional layers and input image size to the process. Our initial aim was to pick the combination that gave us the best possible trade-off between precision and recall (since we wanted to minimize misclassifying a real image as fake but at the same time, identify as many fakes as possible) but there were a few challenges. We could not pick epoch size greater than 10 due to increase in training time. We also noticed that the Accuracy vs. Epoch graph for a learning rate smaller than 0.01 looked like Figure 3., where with more epochs the accuracy wasn't changing much, that is, the learning was too slow.

For these reasons, we chose the combination of learning rate of 0.01 and no. of epochs of 10. Had we had access to more resources, we would have chosen no. of epochs to be 20.

For our final model, we used a training:validation ratio of the dataset to be 80:20, with 13k training images, 3k validation images and 4k testing images.

Results

Classification Model

The training of the ResNet model to classify real and fake images is in progress. We will use the accuracy metric, ROC curves, PR curves and a confusion matrix to evaluate the model and its variations on a validation set. We will hold out a test set to get an estimate of the final generalization performance after training on both the training and validation sets.

MobileNet

Evaluation

In total, 21439 images were used and separated into train sets (80%) and test sets (20%). The overall accuracy is 80% with AUC being 0.88, precision being 0.74, recall being 0.91. Our aim was to get the best possible trade-off between precision and recall since we wanted to minimize misclassifying a real image as fake but at the same time, identify as many fakes as possible.

	Precision	Recall	F1-score	support
Real	0.88	0.65	0.75	2145
Fake	0.72	0.91	0.80	2143
Overall	0.8	0.78	0.78	4288

Table 5. Results for MobileNetV2.

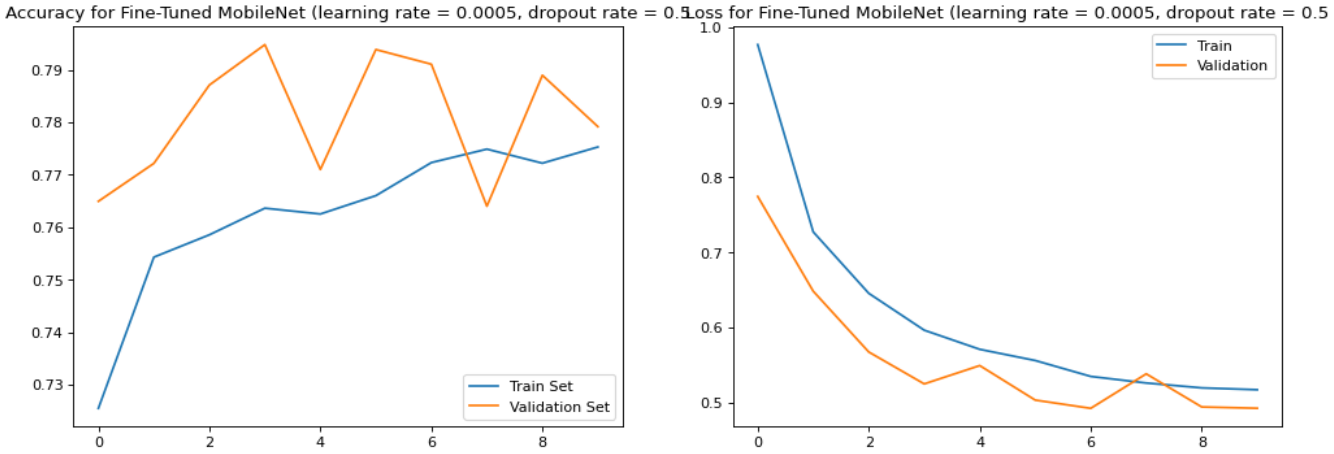

Figure 4. Accuracy over the 10 epochs (left). Loss over the 10 epochs (right) of MobileNet V2 model.

Mapping

In this study, we consider four facial attributes: glasses, race (White, Asian, Black), emotion (positive, negative, neutral), and gender (woman, man). Due to limitations in computing power, we randomly selected 200 samples for each facial attribute and calculated the prediction accuracy of our model. The results are summarized in Table 3.

Race	
Asian	0.76
Black	0.82
White	0.80

Race	
Asian	0.76
Hispanic	0.85
Gender	
Man	0.83
Woman	0.87
Glasses	
Glasses	0.74
No Glasses	0.78
Emotions	
Positive	0.76
Neutral	0.78
Negative	0.75

Table 6. Accuracy for MobileNetV2 by facial category.

We see from the table above that our model does best for faces belonging to women under the category gender, to race Asian under the category race, for faces without glasses and with neutral expressions. After manually analyzing the fakes generated by the rest, we believe that DALL-E 2 performs worse when generating glasses with facial features like frowns or visible teeth in a smile.

ResNet50

Evaluation

In total, 13291 images were used for training, 3322 for validation and 4210 for testing with hyperparameter values of 0.01 for learning rate and 10 for number of epochs. The overall accuracy is 89% with AUC of 0.96, precision of 0.86 and recall of 0.94 on the test set.

	Accuracy	**AUC**	**Precision**	**Recall**
ResNet50	0.89	0.96	0.86	0.94

Table 7. Results for ResNet50.

	Real	**Fake**
Real	1785.0	320
Fake	126	1979

Table 8. Confusion matrix for ResNet50.

Mapping

In this study, we consider four facial attributes: glasses, race (White, Asian, Black, Latino Hispanic), emotion (positive, negative, neutral), and gender (woman, man). Due to limitations in computing power, we randomly selected 200 samples for each facial attribute and calculated the prediction accuracy of our model. The results are summarized in Table 6.

Race	
Asian	0.835
Black	0.9
White	0.82
Hispanic	0.85
Gender	
Man	0.85
Woman	0.84
Glasses	
Glasses	0.80
No Glasses	0.84
Emotions	
Positive	0.82
Neutral	0.86
Negative	0.825

Table 9. Accuracy for ResNet50 by facial category.

We see from the table above that our model does best for faces belonging to men under the category gender, to race Black under the category race, for faces without glasses and with neutral expressions. The results are similar for expressions and glasses for both models. After manually analyzing the fakes generated by the rest, we believe that DALL-E 2 performs worse when generating glasses with facial features like frowns or visible teeth in a smile.

Below we have included an example of a fake image generated by DALL-E 2 that both models could not correctly classify. We believe that this is because it is a headshot with a neutral expression of the face and a clean background.

Figure 5. Corresponding real image (left) and fake image generated by DALL-E 2 (right).

Analysis of ResNet50

To better understand the working of the ResNet50 model and analyze what parts of the image the model uses to differentiate between real and fake images, we studied the outputs of the convolutional layers, that is the feature maps of the model.

Figure 6 is one of the images we used to do this analysis. We chose this particular image for the poorly reconstructed reflection in the glasses by DALL-E 2.

We then ran the image forward through our model and collected the feature maps at each layer. Figure 7 shows the plotted feature maps. We can see in the first few layer outputs that the model is initially learning the features of a face, the shape, the eyes, nose, mouth, forehead, chin etc. (due to the ResNet being trained on a larger, more general ImageNet dataset) but the learning in the last layers of the model are too small to visualize and interpret.

For this reason, we constructed a heatmap using the output of the resnet's Global Average Pooling layer in Figure 8. We see that for this image the model does indeed focus on the poorly regenerated reflection on the glasses and the sides of the face.

Figure 6. Corresponding real image (right) and fake image generated by DALL-E 2 (left).

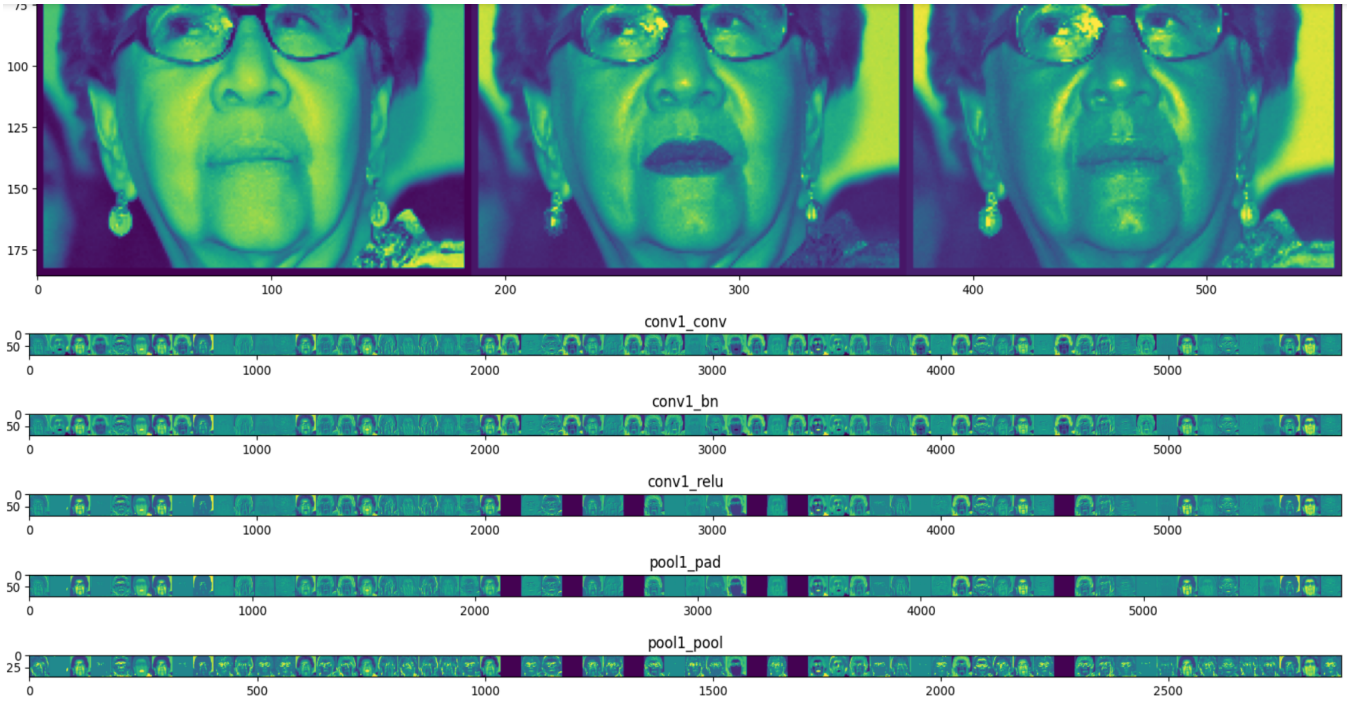

Figure 7. Feature maps generated by ResNet50.

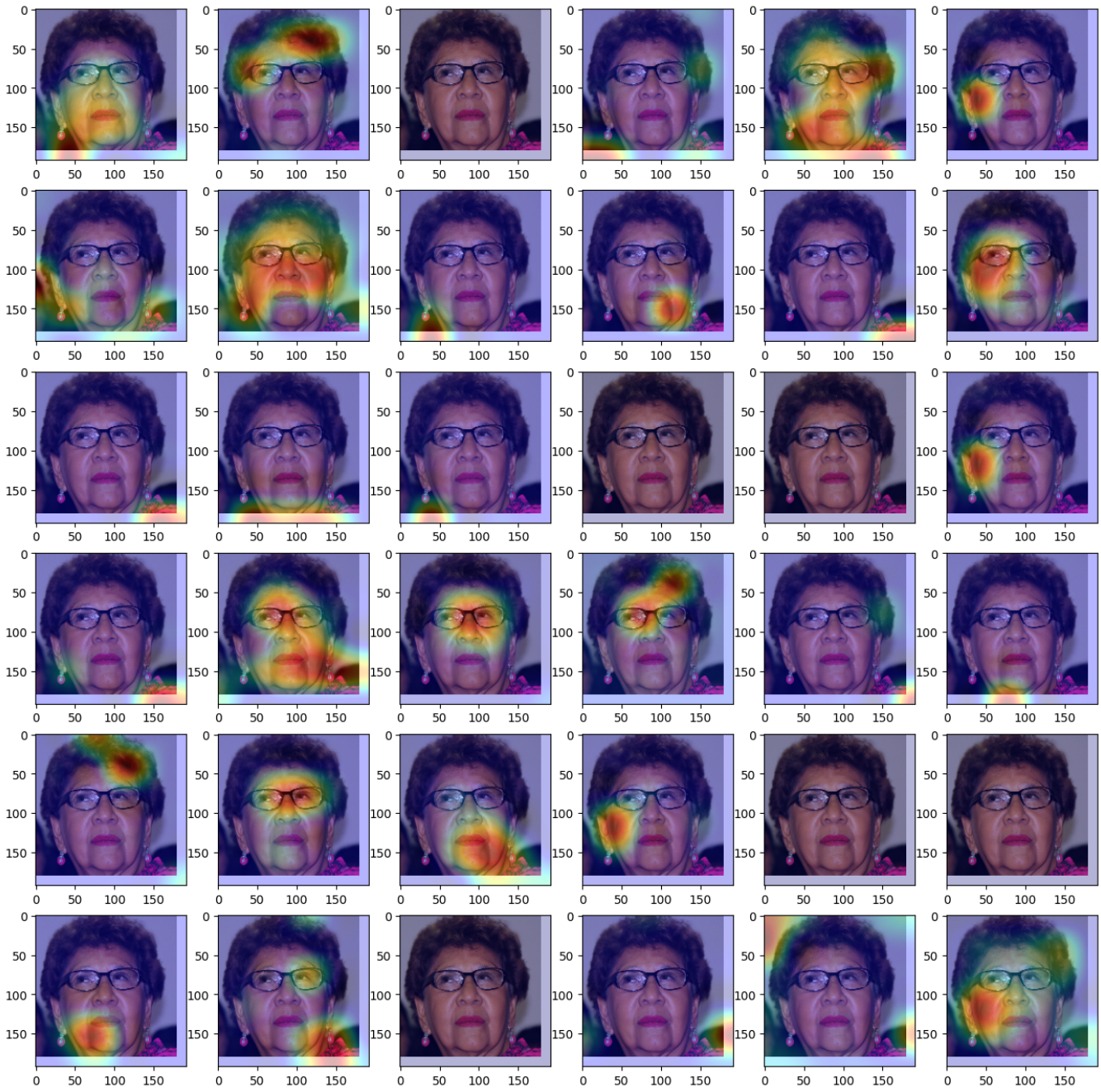

Figure 8. Heatmap generated by the Global Average Pooling layer of ResNet50.

Visualization

After we train the ResNet50, we deploy the pre-trained Keras model on HuggingFace, and use it for visualization. Here, we cluster and plot the feature maps and their corresponding images from the final layer of ResNet to be able to visualize the model's predictions.

We retrieve a specified number of real and fake images, preparing them for processing. The images are resized to a uniform size, and their pixel values are normalized to a range between 0 and 1.

To extract features from the images, we take the feature map generated by the final layer of the pre-trained Keras model (ResNet50). The extracted features are then utilized to create the visualization. The images are organized in a grid-like structure, with each image accompanied by a label indication whether it is real or fake. The labels are color-coded, with red for fake and green for real ones.

We see from the below plot that similar images (faces with similar attributes are grouped together) as are the predicted fake or real images. From this we can conclude that the ResNet model learns the attributes of the faces and whether the image is real or fake and this information is encoded in the feature map of the final model layer.

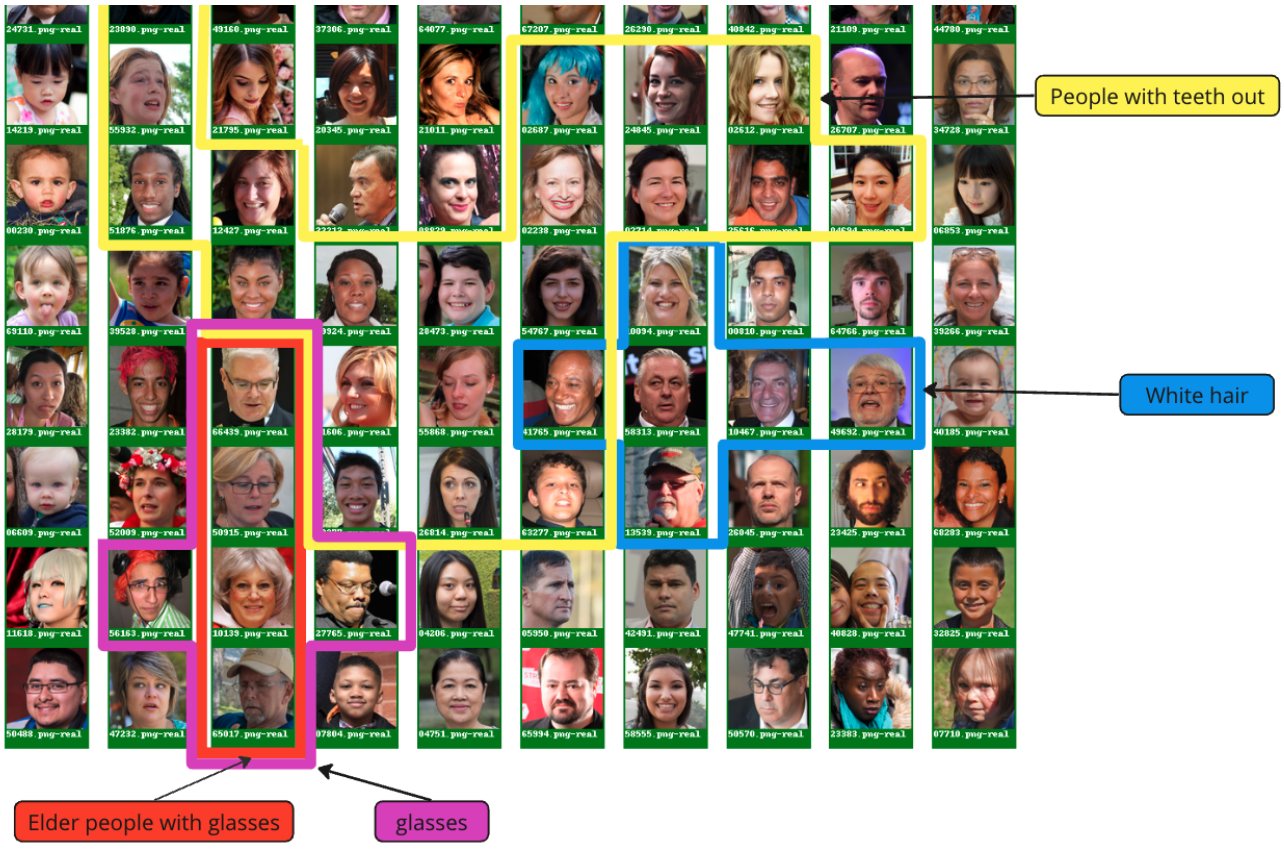

Figure 9. Similar characteristics placed together.

Figure 10. Visualization result based on ResNet50 Model on higer demensional.

Dimensionality Reduction

We also tried the images in a lower-dimensional space (2D) obtained by UMAP, which helps in identifying patterns and relations between the images. UMAP is an unsupervised dimensionality reduction technique that preserves the global structure of the data while also revealing local pattern. In the Figure 11, each image is placed based on its UMAP coordinates, and the border color represents the cluster label from HDBSCAN (with separate color maps for real and fake images). UMAP is applied to the extracted features to reduce their dimensionality to 2D, making it possible to visualize the images in a 2D plot.

The plot shows how images are related to each other based on their extracted features, and the color-coded borders help you understand which cluster each image belongs to and whether the image is real or fake. The generated plot makes it easier to analyze the distribution of images and identify clusters of similar images.

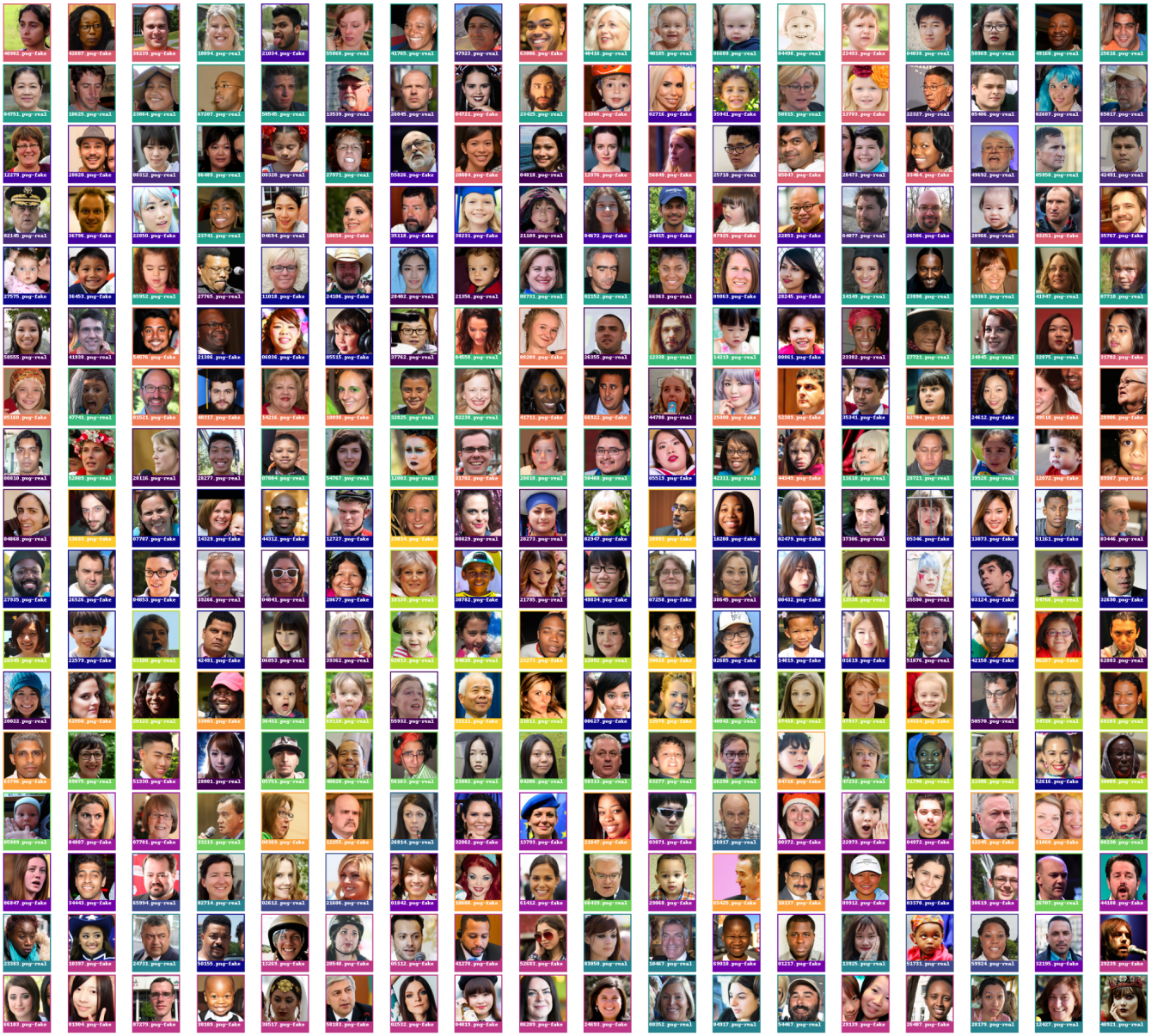

Figure 11. Visualization result based on ResNet50 Model on lower demensional using UMAP.

Deployment

After training the MobileNet model on a dataset of 21439 images stored in AWS S3, the trained models was saved in the TensorFlow format and pushed to HuggingFace's model repositories (Emmawang / mobilenet_v2_fake_image_detection and poojakabber1997/ResNetDallE2Fakes).

To enable practical usage of the model in real-world scenarios, the model was deployed using Streamlit, an open-source Python library for building web applications, and hosted on HuggingFace Space (spaces/Emmawang/Fake_image_detection). This enables users to interact with the model through a user-friendly web interface and receive predictions on new images uploaded to the application.

Deploying the model in this way enables wider accessibility and usability, allowing users to utilize the model's capabilities without requiring specialized knowledge or expertise in machine learning. HuggingFace Space provides an accessible and scalable platform for hosting models and making them available to a wider audience:
https://huggingface.co/spaces/Emmawang/Fake_image_detection.

After thorough research and experimentation, our team has selected ResNet as the optimal model for our web application. The ResNet model has been saved on HuggingFace and in order to seamlessly integrate the model into our webapp, we have leveraged Streamlit, a cutting-edge tool for building and deploying data-driven apps. Our webapp not only utilizes the ResNet model for AI picture detection, providing a prediction and the corresponding confidence score, but also includes a DALL-E 2 AI picture generation function, allowing users to generate their own unique images.

The webapp can be accessed at https://fake-image-generator-detector.streamlit.app/. Please note that in order to generate AI pictures, users will need to apply for their own OpenAI API keys. We are excited to offer this innovative tool to users and believe it will have a positive impact on a wide range of industries, from marketing to entertainment.

Conclusion

In conclusion, our project aims to address the challenges and ethical implications of using AI-generated facial images by developing a machine learning model that can distinguish between real human faces and images generated by DALL-E 2. Through the analysis of a diverse dataset that includes real human images from the FFHQ dataset and fake images generated by Dall-E 2, we hope to gain valuable insights into the differences between AI-generated and real human faces. This research has the potential to contribute to the development of responsible and ethical practices for the use of AI-generated images in various fields, such as criminal detection and prevention.

The studies conducted by Sabithaa et al. (2020), Zhang et al. (2020), and He et al. (2016) provide valuable insights into the development of a fake image detection model using convolutional neural

networks (CNNs). He et al. (2016) presented the use of a pre-trained ResNet model to address the issue of vanishing gradients in deep neural networks, which can be fine-tuned and visualized in our study to gain insights into the learned features and prediction process.

Limitation

1. Limited classification accuracy: Due to time constraints, we could not achieve an ideal classification accuracy for the original data. With an accuracy of around 80% for each class (glass, race, facial expression, gender), we had to train the model without optimizing it further. To improve accuracy, a more extensive dataset with over 40k images for each class could be analyzed, but this was not feasible within our time limitations.

2. Model comparison and selection: Although we compared different models, we did not have the opportunity to explore the best possible model for our task. Combining MobileNetV2 and ResNet might have yielded a more accurate and efficient model, but due to time constraints, this possibility was not fully explored.

By addressing the challenges and ethical implications of using AI-generated images, this research contributes to the development of responsible and ethical practices in various fields.

By drawing upon these studies, we can enhance the architecture and performance of our fake image detection model. Our study aims not only to achieve high accuracy in detecting fake images, but also to understand the underlying features and categories that significantly impact the model's performance, such as race, gender, facial expressions, and wearing accessories. By incorporating insights from these studies, our model can be further refined and optimized for real-world applications where the detection of fake images generated by AI is crucial.

Roles

Our overall high-level roles include

1. Project management
2. Dataset acquisition
3. Building the model pipeline
4. Model training
5. Model experiments
6. Evaluation

We want to ensure that each member takes part in all the above roles. To break down, we first divided the project and its deliverables into steps and tasks that can be assigned and distributed.

Tasks:

1. Pooja Kabber

    a. Selecting the right model type and architecture
    b. Experimenting with the model results
2. Scott Lai
    a. Building the model pipeline
    b. Experimenting with the model results
3. Song Young Oh
    a. Training the model and model selection by hyperparameter optimization
    b. Model evaluation
4. Emma Wang
    a. Building / Finding and preparing the dataset
    b. Model training/ fine tuning
    c. Model evaluation

References

1. Sabithaa, R., Aruna, A., Karthik, S., & Shanthini, J. (2020). Enhanced model for fake image detection (EMFID) using convolutional neural networks with histogram and wavelet based feature extractions. Journal of Ambient Intelligence and Humanized Computing, 11(7), 3005-3018.

2. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

3. Zeyang S., Zheng L., Ning Y., and Yang Z. (2022). DE-FAKE: Detection and Attribution of Fake Images Generated by Text-to-Image Generation Models. arXiv preprint arXiv:2210.06998.

4. Riccardo C., Davide C., Giada Z., Giovanni P., Koki N., and Luisa V. (2022). On the Detection of Synthetic Images Generated by Diffusion Models. arXiv preprint arXiv:2211.00680v1.

5. Utkarsh O., Yuheng L., and Yong Jae L. (2023). Towards Universal Fake Image Detectors that Generalize Across Generative Models. arXiv preprint arXiv:2302.10174v1.

6. Rezende, E.R.S.D.; Ruppert, G.C.S.; Carvalho, T. Detecting Computer Generated Images with Deep Convolutional Neural Networks. In Proceedings of the 30th SIBGRAPI Conference on Graphics, Patterns and Images, Niteroi, Brazil, 17–20 October 2017; pp. 71–78.

7. Ye Y., Weitong H., Wei Z., Ting W., and Yun-Qing S. (2018). Distinguishing Computer-generated Graphics from Natural Images Based on Sensor Pattern Noise and Deep Learning. arXiv preprint arXiv:1803.09403.

8. Bekhet S. & Alahmer H. A Robust Deep Learning Approach for Glass Detection in Non-standard Facial Images. IET Biom. 2021;10:74-86.

9. Falko M., Christian R., and Marc S. Exploiting visual artifacts to expose deepfakes and face manipulations. In 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW), pages 83–92. IEEE, 2019.

10. Andreas R., Davide C., Luisa V., Christian R., Justus T., and Matthias N. Faceforensics++: Learning to detect manipulated facial images. In Proceedings of the IEEE International Conference on Computer Vision, pages 1–11, 2019.

11. Lingzhi L., Jianmin B., Ting Z., Hao Y., Dong C., Fang W., and Baining G. Face x-ray for more general face forgery detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5001–5010, 2020.

12. Yuyang Q., Guojun Y., Lu S., Zixuan C., and Jing S. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In the European Conference on Computer Vision, pages 86–103. Springer, 2020.

13. Xi W., Zhen X., YuTao G., and Yu X. Sstnet: Detecting manipulated faces through spatial, steganalysis and temporal features. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 2952–2956. IEEE, 2020.

14. Xin Y., Yuezun L., and Siwei L. Exposing deep fakes using inconsistent head poses. In IEEE International Confer- ence on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019, pages 8261–8265. IEEE, 2019.

15. Mingxing T. and Quoc V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Ka- malika Chaudhuri and Ruslan Salakhutdinov, editors, Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, volume 97 of Proceedings of Machine Learning Research, pages 6105–6114. PMLR, 2019.

16. Francois Chollet. Xception: Deep Learning with Depthwise Separable Convolutions. In IEEE Conference on Computer Vision and Pattern Recognition, 2017.

17. Trpakov, D. (2022). Vit Face Expression Model. Hugging Face. https://huggingface.co/trpakov/vit-face-expression

18. Siddharth Mandgi (2021). Glasses Detection - OpenCV, DLIB & Edge Detection. MLearning.ai. https://medium.com/mlearning-ai/glasses-detection-opencv-dlib-bf4cd50856da

19. O. M. Parkhi, A. Vedaldi, A. Zisserman: Deep Face Recognition. British Machine Vision Conference, 2015

Appendix

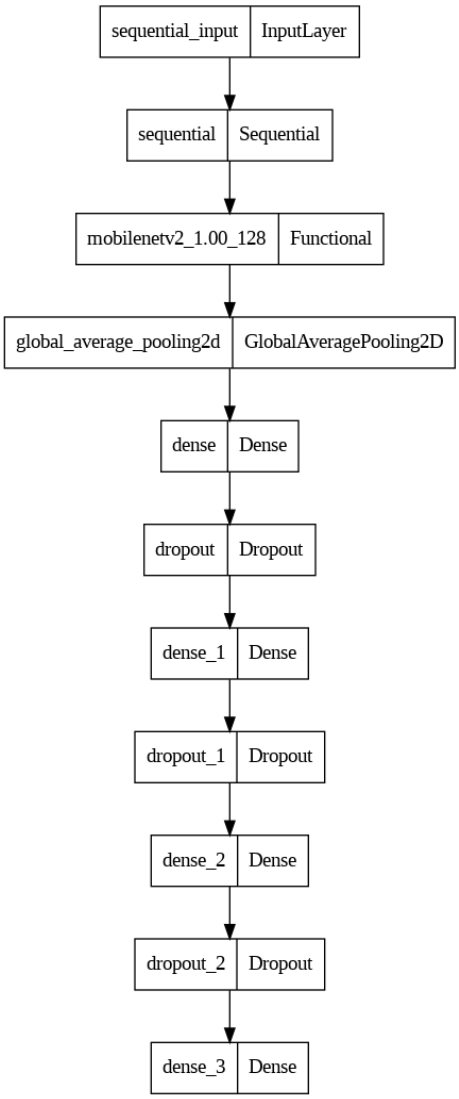

Figure 10. Architecture of MobileNetV2.

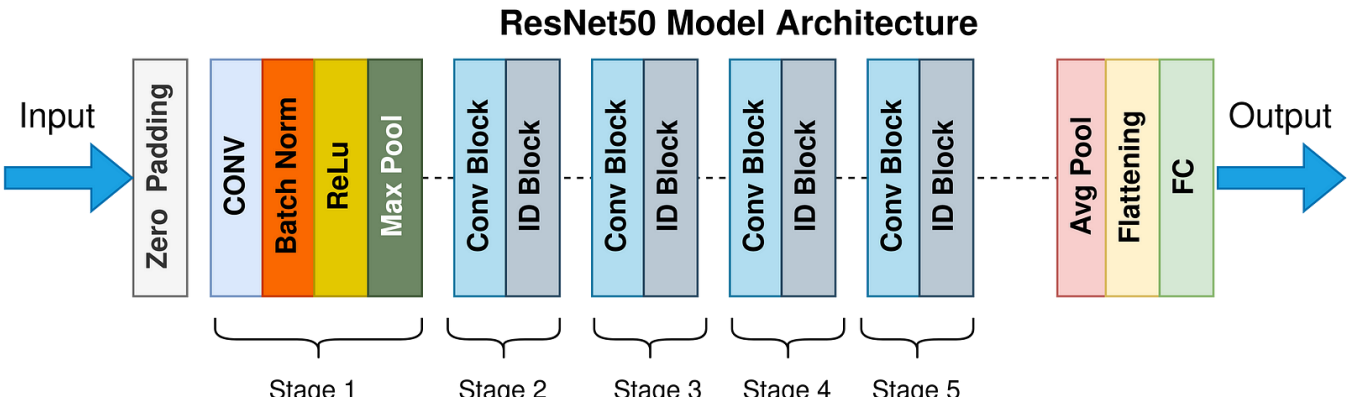

Figure 11. Architecture of ResNet50