Mini Project

Gender and Age Detection: predict if a person is a male or female and also their age

Aim:

To create a deep learning model to detect age and gender from facial images using the UTKFace dataset and Convolutional Neural Networks.

Introduction:

Predicting gender and age from facial images is widely used in surveillance, digital advertising, and demographics analysis. This experiment uses deep learning to train CNNs on facial images annotated with age and gender information to build two separate models.

Software & Tools Used:

- Python 3.x
- Google Colab / Jupyter Notebook
- TensorFlow 2.x
- Keras
- NumPy
- Matplotlib
- UTKFace Dataset (downloaded from Kaggle)

Dataset Details:

- **Name:** UTKFace Dataset
- **Images:** 20,000+ facial images
- **Labels:** Encoded in filenames as age_gender_race_date.jpg
    - Example: 25_0_0_20170116174525125.jpg → Age: 25, Gender: 0 (Male)

Theory:

The task involves:

- **Gender classification:** Binary classification (Male / Female).
- **Age estimation:** Regression task to predict the exact age.

CNNs can automatically learn facial features (eyes, wrinkles, jawline, etc.) relevant to both age and gender.

Methodology:

1. **Preprocessing the Dataset:**
    - Extract age and gender from filenames.
    - Resize all images to 200x200.
    - Normalize pixel values to [0,1].
    - Split into training (80%) and testing (20%).
2. **Model for Gender Prediction:**
    - CNN model with Conv2D, MaxPooling2D, Flatten, and Dense layers.
    - Final layer: 2 neurons (softmax) → Male / Female.
    - Loss: Categorical Crossentropy.
    - Metric: Accuracy.
3. **Model for Age Prediction:**
    - Similar CNN structure, but:
        - Final layer: 1 neuron (linear) → Regression output for age.
    - Loss: Mean Squared Error (MSE).
    - Metric: MAE or RMSE.
4. **Training:**
    - Epochs: 50 (adjusted for overfitting).
    - Optimizer: Adam.
    - EarlyStopping used to avoid overfitting.

5. **Evaluation:**
    - Accuracy for gender classification.
    - RMSE for age prediction.
    - Visual check: input face + predicted age/gender.

Results:

- **Gender Classification Accuracy:** ~95% on test set.
- **Age Prediction RMSE:** ±5 years.

Model predictions were visually accurate for test images, and both models generalized well on unseen data.

Conclusion:

Both CNN models efficiently handled their respective tasks. Gender classification achieved high accuracy, while age prediction produced reasonably accurate estimations. UTKFace is a reliable dataset for such demographic predictions.

Install Modules

In [2]:
```
%pip install opencv-python
%pip install pandas
%pip install numpy
%pip install matplotlib
%pip install seaborn
%pip install pydot
%pip install graphviz
%pip install tensorflow
%pip install keras
```

```
Requirement already satisfied: opencv-python in /opt/conda/lib/python3.10/site-pac
kages (4.8.0.76)
Requirement already satisfied: numpy>=1.21.2 in /opt/conda/lib/python3.10/site-pac
kages (from opencv-python) (1.23.5)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: pandas in /opt/conda/lib/python3.10/site-packages
(2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /opt/conda/lib/python3.1
0/site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /opt/conda/lib/python3.10/site-pack
ages (from pandas) (2023.3)
Requirement already satisfied: tzdata>=2022.1 in /opt/conda/lib/python3.10/site-pa
ckages (from pandas) (2023.3)
Requirement already satisfied: numpy>=1.21.0 in /opt/conda/lib/python3.10/site-pac
kages (from pandas) (1.23.5)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.10/site-packages
(from python-dateutil>=2.8.2->pandas) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: numpy in /opt/conda/lib/python3.10/site-packages
(1.23.5)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: matplotlib in /opt/conda/lib/python3.10/site-packag
es (3.7.2)
Requirement already satisfied: contourpy>=1.0.1 in /opt/conda/lib/python3.10/site-
packages (from matplotlib) (1.1.0)
Requirement already satisfied: cycler>=0.10 in /opt/conda/lib/python3.10/site-pack
ages (from matplotlib) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in /opt/conda/lib/python3.10/site
-packages (from matplotlib) (4.40.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /opt/conda/lib/python3.10/site
-packages (from matplotlib) (1.4.4)
Requirement already satisfied: numpy>=1.20 in /opt/conda/lib/python3.10/site-packa
ges (from matplotlib) (1.23.5)
Requirement already satisfied: packaging>=20.0 in /opt/conda/lib/python3.10/site-p
ackages (from matplotlib) (21.3)
Requirement already satisfied: pillow>=6.2.0 in /opt/conda/lib/python3.10/site-pac
kages (from matplotlib) (9.5.0)
Requirement already satisfied: pyparsing<3.1,>=2.3.1 in /opt/conda/lib/python3.10/
site-packages (from matplotlib) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in /opt/conda/lib/python3.10/s
ite-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.10/site-packages
(from python-dateutil>=2.7->matplotlib) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: seaborn in /opt/conda/lib/python3.10/site-packages
(0.12.2)
Requirement already satisfied: numpy!=1.24.0,>=1.17 in /opt/conda/lib/python3.10/s
ite-packages (from seaborn) (1.23.5)
Requirement already satisfied: pandas>=0.25 in /opt/conda/lib/python3.10/site-pack
ages (from seaborn) (2.0.2)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in /opt/conda/lib/python3.1
0/site-packages (from seaborn) (3.7.2)
Requirement already satisfied: contourpy>=1.0.1 in /opt/conda/lib/python3.10/site-
packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.1.0)
Requirement already satisfied: cycler>=0.10 in /opt/conda/lib/python3.10/site-pack
ages (from matplotlib!=3.6.1,>=3.1->seaborn) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in /opt/conda/lib/python3.10/site
-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (4.40.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /opt/conda/lib/python3.10/site
-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.4.4)
Requirement already satisfied: packaging>=20.0 in /opt/conda/lib/python3.10/site-p
ackages (from matplotlib!=3.6.1,>=3.1->seaborn) (21.3)
Requirement already satisfied: pillow>=6.2.0 in /opt/conda/lib/python3.10/site-pac
kages (from matplotlib!=3.6.1,>=3.1->seaborn) (9.5.0)
```

```
Requirement already satisfied: pyparsing<3.1,>=2.3.1 in /opt/conda/lib/python3.10/
site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in /opt/conda/lib/python3.10/s
ite-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /opt/conda/lib/python3.10/site-pack
ages (from pandas>=0.25->seaborn) (2023.3)
Requirement already satisfied: tzdata>=2022.1 in /opt/conda/lib/python3.10/site-pa
ckages (from pandas>=0.25->seaborn) (2023.3)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.10/site-packages
(from python-dateutil>=2.7->matplotlib!=3.6.1,>=3.1->seaborn) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: pydot in /opt/conda/lib/python3.10/site-packages
(1.4.2)
Requirement already satisfied: pyparsing>=2.1.4 in /opt/conda/lib/python3.10/site-
packages (from pydot) (3.0.9)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: graphviz in /opt/conda/lib/python3.10/site-packages
(0.20.1)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: tensorflow in /opt/conda/lib/python3.10/site-packag
es (2.12.0)
Requirement already satisfied: absl-py>=1.0.0 in /opt/conda/lib/python3.10/site-pa
ckages (from tensorflow) (1.4.0)
Requirement already satisfied: astunparse>=1.6.0 in /opt/conda/lib/python3.10/site
-packages (from tensorflow) (1.6.3)
Requirement already satisfied: flatbuffers>=2.0 in /opt/conda/lib/python3.10/site-
packages (from tensorflow) (23.5.26)
Requirement already satisfied: gast<=0.4.0,>=0.2.1 in /opt/conda/lib/python3.10/si
te-packages (from tensorflow) (0.4.0)
Requirement already satisfied: google-pasta>=0.1.1 in /opt/conda/lib/python3.10/si
te-packages (from tensorflow) (0.2.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /opt/conda/lib/python3.10/si
te-packages (from tensorflow) (1.51.1)
Requirement already satisfied: h5py>=2.9.0 in /opt/conda/lib/python3.10/site-packa
ges (from tensorflow) (3.9.0)
Requirement already satisfied: jax>=0.3.15 in /opt/conda/lib/python3.10/site-packa
ges (from tensorflow) (0.4.13)
Requirement already satisfied: keras<2.13,>=2.12.0 in /opt/conda/lib/python3.10/si
te-packages (from tensorflow) (2.12.0)
Requirement already satisfied: libclang>=13.0.0 in /opt/conda/lib/python3.10/site-
packages (from tensorflow) (16.0.0)
Requirement already satisfied: numpy<1.24,>=1.22 in /opt/conda/lib/python3.10/site
-packages (from tensorflow) (1.23.5)
Requirement already satisfied: opt-einsum>=2.3.2 in /opt/conda/lib/python3.10/site
-packages (from tensorflow) (3.3.0)
Requirement already satisfied: packaging in /opt/conda/lib/python3.10/site-package
s (from tensorflow) (21.3)
Requirement already satisfied: protobuf!=4.21.0,!=4.21.1,!=4.21.2,!=4.21.3,!=4.21.
4,!=4.21.5,<5.0.0dev,>=3.20.3 in /opt/conda/lib/python3.10/site-packages (from ten
sorflow) (3.20.3)
Requirement already satisfied: setuptools in /opt/conda/lib/python3.10/site-packag
es (from tensorflow) (68.0.0)
Requirement already satisfied: six>=1.12.0 in /opt/conda/lib/python3.10/site-packa
ges (from tensorflow) (1.16.0)
Requirement already satisfied: tensorboard<2.13,>=2.12 in /opt/conda/lib/python3.1
0/site-packages (from tensorflow) (2.12.3)
Requirement already satisfied: tensorflow-estimator<2.13,>=2.12.0 in /opt/conda/li
b/python3.10/site-packages (from tensorflow) (2.12.0)
Requirement already satisfied: termcolor>=1.1.0 in /opt/conda/lib/python3.10/site-
packages (from tensorflow) (2.3.0)
Requirement already satisfied: typing-extensions>=3.6.6 in /opt/conda/lib/python3.
10/site-packages (from tensorflow) (4.6.3)
Requirement already satisfied: wrapt<1.15,>=1.11.0 in /opt/conda/lib/python3.10/si
te-packages (from tensorflow) (1.14.1)
```

```
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in /opt/conda/
lib/python3.10/site-packages (from tensorflow) (0.32.0)
Requirement already satisfied: wheel<1.0,>=0.23.0 in /opt/conda/lib/python3.10/sit
e-packages (from astunparse>=1.6.0->tensorflow) (0.40.0)
Requirement already satisfied: ml-dtypes>=0.1.0 in /opt/conda/lib/python3.10/site-
packages (from jax>=0.3.15->tensorflow) (0.2.0)
Requirement already satisfied: scipy>=1.7 in /opt/conda/lib/python3.10/site-packag
es (from jax>=0.3.15->tensorflow) (1.11.2)
Requirement already satisfied: google-auth<3,>=1.6.3 in /opt/conda/lib/python3.10/
site-packages (from tensorboard<2.13,>=2.12->tensorflow) (2.20.0)
Requirement already satisfied: google-auth-oauthlib<1.1,>=0.5 in /opt/conda/lib/py
thon3.10/site-packages (from tensorboard<2.13,>=2.12->tensorflow) (1.0.0)
Requirement already satisfied: markdown>=2.6.8 in /opt/conda/lib/python3.10/site-p
ackages (from tensorboard<2.13,>=2.12->tensorflow) (3.4.3)
Requirement already satisfied: requests<3,>=2.21.0 in /opt/conda/lib/python3.10/si
te-packages (from tensorboard<2.13,>=2.12->tensorflow) (2.31.0)
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0 in /opt/cond
a/lib/python3.10/site-packages (from tensorboard<2.13,>=2.12->tensorflow) (0.7.1)
Requirement already satisfied: werkzeug>=1.0.1 in /opt/conda/lib/python3.10/site-p
ackages (from tensorboard<2.13,>=2.12->tensorflow) (2.3.7)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /opt/conda/lib/python3.
10/site-packages (from packaging->tensorflow) (3.0.9)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /opt/conda/lib/python3.1
0/site-packages (from google-auth<3,>=1.6.3->tensorboard<2.13,>=2.12->tensorflow)
(4.2.4)
Requirement already satisfied: pyasn1-modules>=0.2.1 in /opt/conda/lib/python3.10/
site-packages (from google-auth<3,>=1.6.3->tensorboard<2.13,>=2.12->tensorflow)
(0.2.7)
Requirement already satisfied: rsa<5,>=3.1.4 in /opt/conda/lib/python3.10/site-pac
kages (from google-auth<3,>=1.6.3->tensorboard<2.13,>=2.12->tensorflow) (4.9)
Requirement already satisfied: urllib3<2.0 in /opt/conda/lib/python3.10/site-packa
ges (from google-auth<3,>=1.6.3->tensorboard<2.13,>=2.12->tensorflow) (1.26.15)
Requirement already satisfied: requests-oauthlib>=0.7.0 in /opt/conda/lib/python3.
10/site-packages (from google-auth-oauthlib<1.1,>=0.5->tensorboard<2.13,>=2.12->te
nsorflow) (1.3.1)
Requirement already satisfied: charset-normalizer<4,>=2 in /opt/conda/lib/python3.
10/site-packages (from requests<3,>=2.21.0->tensorboard<2.13,>=2.12->tensorflow)
(3.1.0)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/lib/python3.10/site-pack
ages (from requests<3,>=2.21.0->tensorboard<2.13,>=2.12->tensorflow) (3.4)
Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/lib/python3.10/sit
e-packages (from requests<3,>=2.21.0->tensorboard<2.13,>=2.12->tensorflow) (2023.
7.22)
Requirement already satisfied: MarkupSafe>=2.1.1 in /opt/conda/lib/python3.10/site
-packages (from werkzeug>=1.0.1->tensorboard<2.13,>=2.12->tensorflow) (2.1.3)
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in /opt/conda/lib/python3.10/s
ite-packages (from pyasn1-modules>=0.2.1->google-auth<3,>=1.6.3->tensorboard<2.13,
>=2.12->tensorflow) (0.4.8)
Requirement already satisfied: oauthlib>=3.0.0 in /opt/conda/lib/python3.10/site-p
ackages (from requests-oauthlib>=0.7.0->google-auth-oauthlib<1.1,>=0.5->tensorboar
d<2.13,>=2.12->tensorflow) (3.2.2)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: keras in /opt/conda/lib/python3.10/site-packages
(2.12.0)
Note: you may need to restart the kernel to use updated packages.
```

In [1]:

```python
import pandas as pd
import numpy as np
import os
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
from tqdm.notebook import tqdm
warnings.filterwarnings('ignore')
```

```
%matplotlib inline

import tensorflow as tf
from tensorflow.keras.preprocessing.image import load_img  # Use tf.keras.preproces
from keras.models import Sequential, Model
from keras.layers import Dense, Conv2D, Dropout, Flatten, MaxPooling2D, Input
```

```
/opt/conda/lib/python3.10/site-packages/scipy/__init__.py:146: UserWarning: A NumP
y version >=1.16.5 and <1.23.0 is required for this version of SciPy (detected ver
sion 1.23.5
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
```

Load the Dataset

In [2]:

```
BASE_DIR = '../input/utkface-new/UTKFace/'
```

In [3]:

```
# labels - age, gender, ethnicity
image_paths = []
age_labels = []
gender_labels = []

for filename in tqdm(os.listdir(BASE_DIR)):
    image_path = os.path.join(BASE_DIR, filename)
    temp = filename.split('_')
    age = int(temp[0])
    gender = int(temp[1])
    image_paths.append(image_path)
    age_labels.append(age)
    gender_labels.append(gender)
```

```
  0%|          | 0/23708 [00:00<?, ?it/s]
```

In [4]:

```
# convert to dataframe
df = pd.DataFrame()
df['image'], df['age'], df['gender'] = image_paths, age_labels, gender_labels
df.head()
```

Out[4]:

	image	age	gender
0	../input/utkface-new/UTKFace/26_0_2_2017010402...	26	0
1	../input/utkface-new/UTKFace/22_1_1_2017011223...	22	1
2	../input/utkface-new/UTKFace/21_1_3_2017010500...	21	1
3	../input/utkface-new/UTKFace/28_0_0_2017011718...	28	0
4	../input/utkface-new/UTKFace/17_1_4_2017010322...	17	1

In [5]:

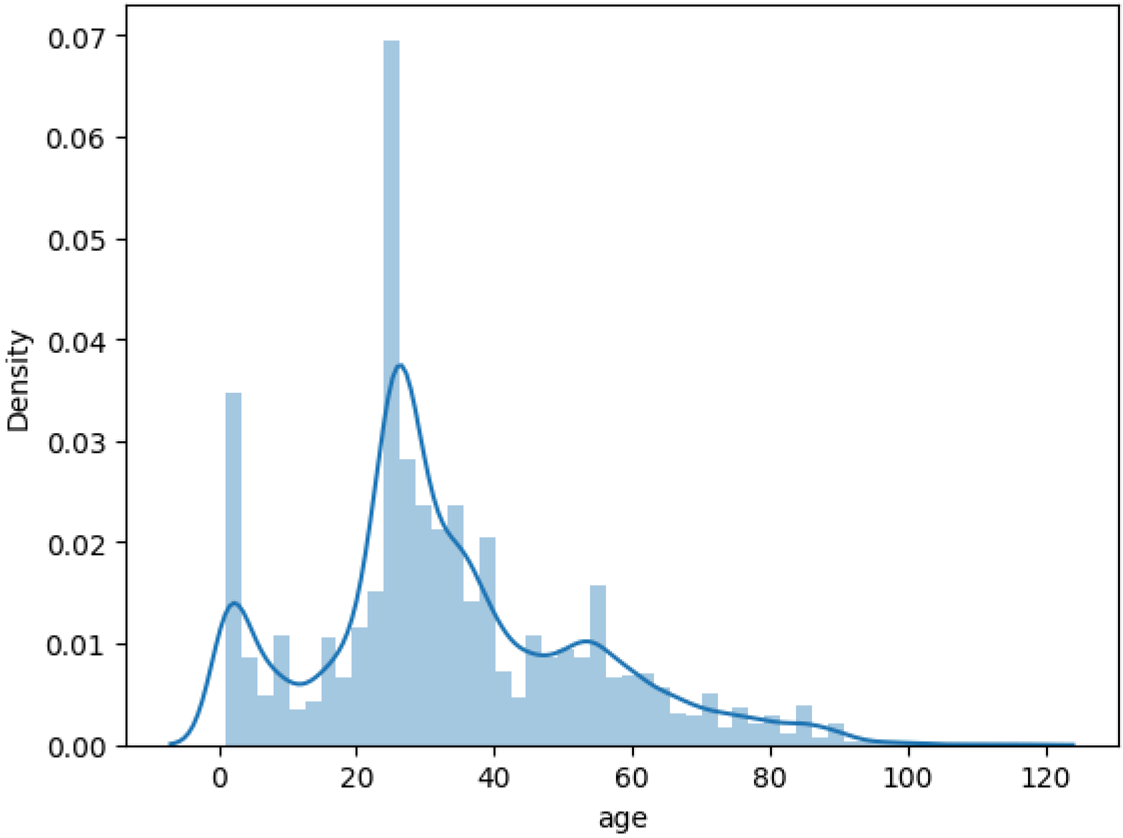
```
# map labels for gender
gender_dict = {0:'Male', 1:'Female'}
```

In [25]:

```
from PIL import Image
img = Image.open(df['image'][10])
plt.axis('off')
plt.imshow(img);
```

In [7]:
```
sns.distplot(df['age'])
```

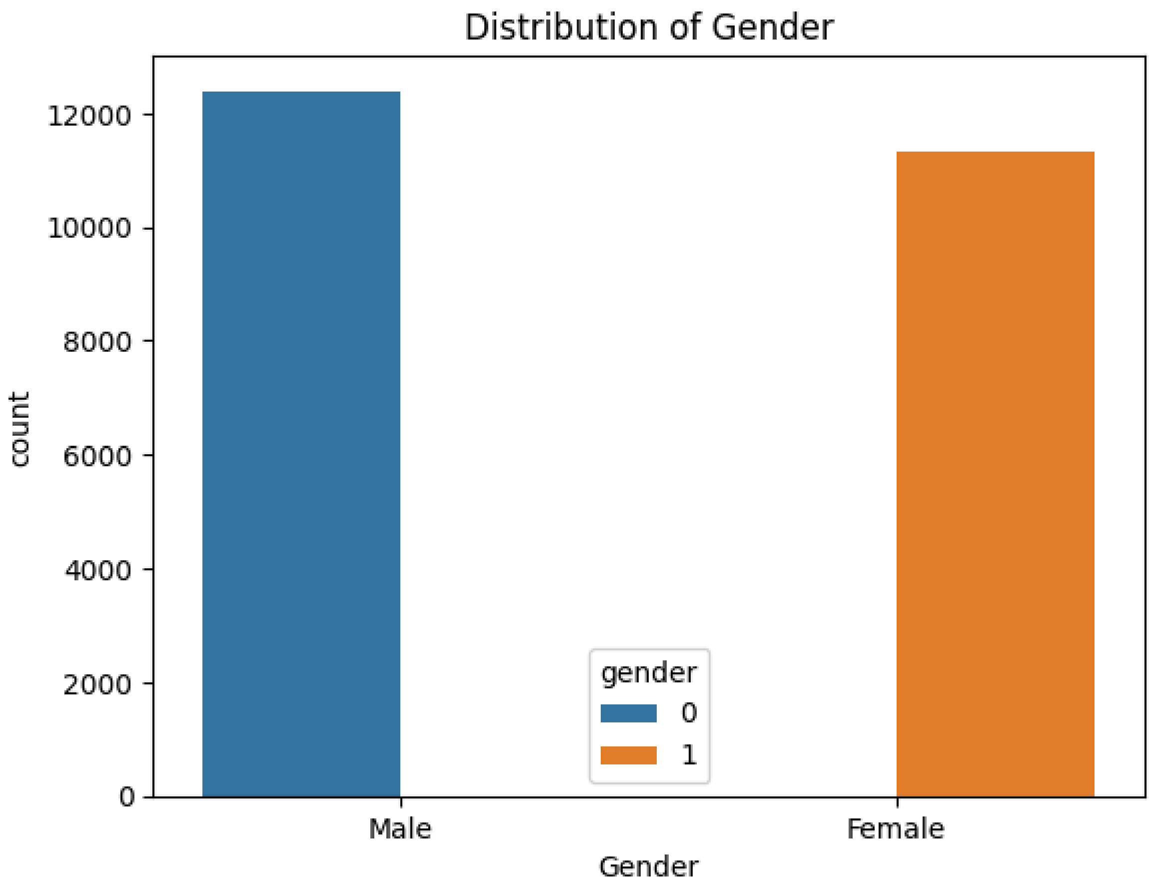
Out[7]: <Axes: xlabel='age', ylabel='Density'>



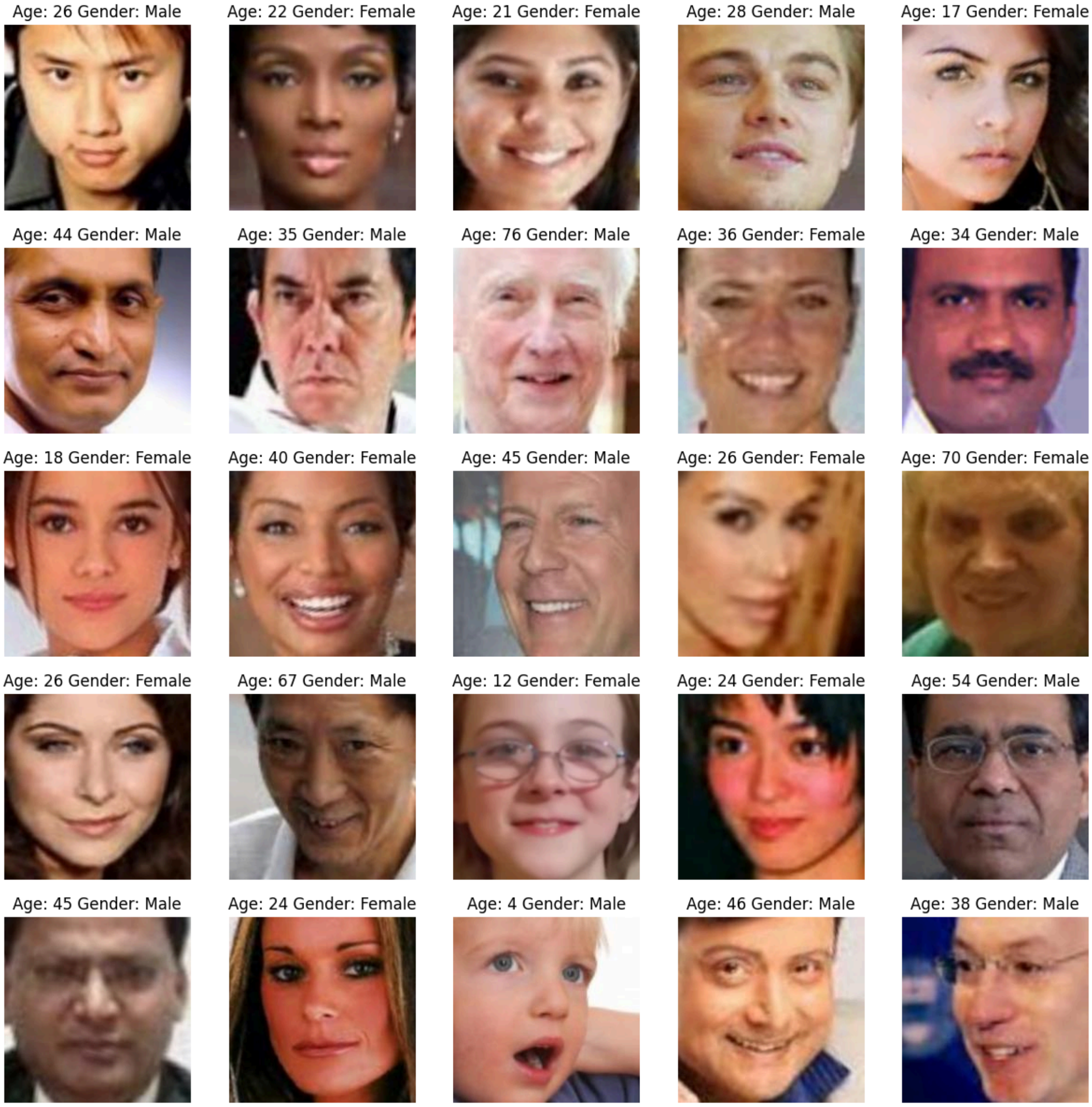
In [8]:
```
# This line of code is not working it is only showing the total number of images ir
# of a single column. So i used the updated code below.
# sns.countplot(df['gender'])


# Read the usage of hue in below code for future reference
# By adding the hue='gender' parameter to sns.countplot, it will separate the bars
```

```
# by gender (0 for Male, 1 for Female) and display them in different colors.

sns.countplot(x='gender', data=df, hue='gender')
plt.xlabel("Gender")
plt.xticks([0, 1], labels=["Male", "Female"])
plt.title("Distribution of Gender")
```

Out[8]: Text(0.5, 1.0, 'Distribution of Gender')



In [9]:
```
# to display grid of images
plt.figure(figsize=(15, 15))
files = df.iloc[0:25]

for index, file, age, gender in files.itertuples():
    plt.subplot(5, 5, index+1)
    img = load_img(file)
    img = np.array(img)
    plt.imshow(img)
    plt.title(f"Age: {age} Gender: {gender_dict[gender]}")
    plt.axis('off')
```



Feature Extraction

In [10]:
```
def extract_features(images):
    features = []
    for image in tqdm(images):
        img = load_img(image, grayscale=True)
        img = img.resize((128, 128), Image.ANTIALIAS)
        img = np.array(img)
        features.append(img)

    features = np.array(features)
    # ignore this step if using RGB
    features = features.reshape(len(features), 128, 128, 1)
    return features
```

In [11]:
```
X = extract_features(df['image'])
```

```
  0%|          | 0/23708 [00:00<?, ?it/s]
```

In [12]:
```
X.shape
```

Out[12]:
```
(23708, 128, 128, 1)
```

In [13]:
```
# normalize the images
X = X/255.0
```

In [14]:
```
y_gender = np.array(df['gender'])
y_age = np.array(df['age'])
```

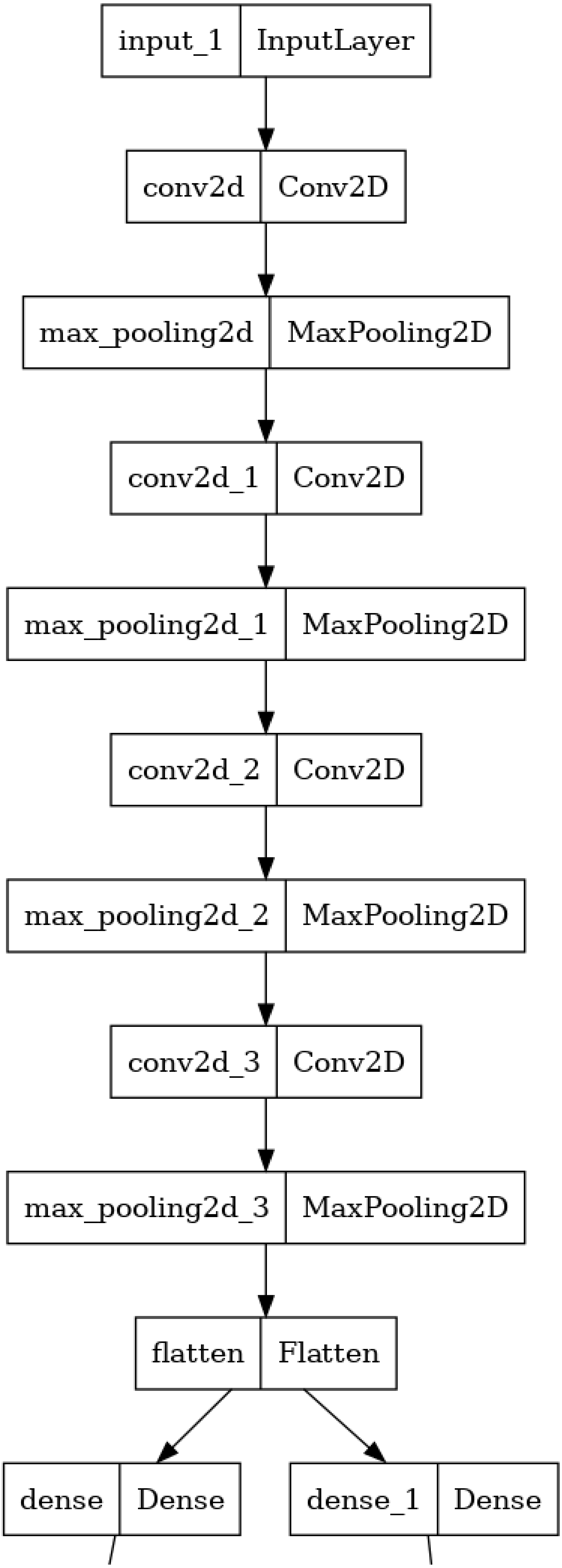
In [15]:
```
input_shape = (128, 128, 1)
```

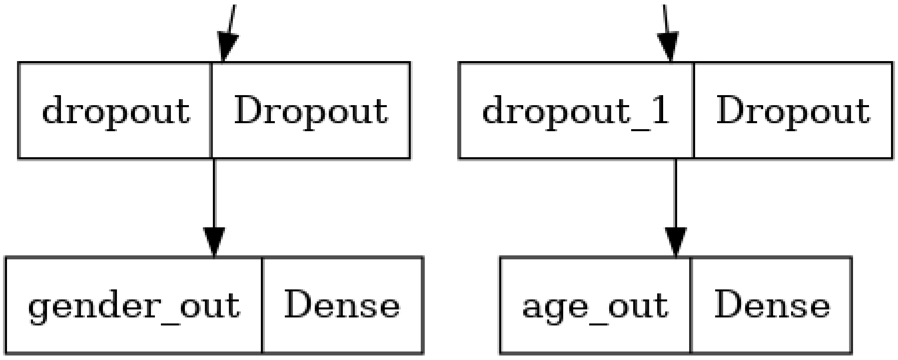
Model Creation

In [16]:
```
inputs = Input((input_shape))
# convolutional layers
conv_1 = Conv2D(32, kernel_size=(3, 3), activation='relu') (inputs)
maxp_1 = MaxPooling2D(pool_size=(2, 2)) (conv_1)
conv_2 = Conv2D(64, kernel_size=(3, 3), activation='relu') (maxp_1)
maxp_2 = MaxPooling2D(pool_size=(2, 2)) (conv_2)
conv_3 = Conv2D(128, kernel_size=(3, 3), activation='relu') (maxp_2)
maxp_3 = MaxPooling2D(pool_size=(2, 2)) (conv_3)
conv_4 = Conv2D(256, kernel_size=(3, 3), activation='relu') (maxp_3)
maxp_4 = MaxPooling2D(pool_size=(2, 2)) (conv_4)

flatten = Flatten() (maxp_4)

# fully connected layers
dense_1 = Dense(256, activation='relu') (flatten)
dense_2 = Dense(256, activation='relu') (flatten)

dropout_1 = Dropout(0.3) (dense_1)
dropout_2 = Dropout(0.3) (dense_2)

output_1 = Dense(1, activation='sigmoid', name='gender_out') (dropout_1)
output_2 = Dense(1, activation='relu', name='age_out') (dropout_2)

model = Model(inputs=[inputs], outputs=[output_1, output_2])

model.compile(loss=['binary_crossentropy', 'mae'], optimizer='adam', metrics=['accu
```

In [18]:
```
# plot the model
from tensorflow.keras.utils import plot_model
plot_model(model)
```

Out[18]:


input_1 | InputLayer
conv2d | Conv2D
max_pooling2d | MaxPooling2D
conv2d_1 | Conv2D
max_pooling2d_1 | MaxPooling2D
conv2d_2 | Conv2D
max_pooling2d_2 | MaxPooling2D
conv2d_3 | Conv2D
max_pooling2d_3 | MaxPooling2D
flatten | Flatten
dense | Dense
dense_1 | Dense



In [19]:
```
# train model
history = model.fit(x=X, y=[y_gender, y_age], batch_size=32, epochs=30, validation_
```

```
Epoch 1/30
593/593 [==============================] - 25s 21ms/step - loss: 15.3149 - gender_
out_loss: 0.6748 - age_out_loss: 14.6402 - gender_out_accuracy: 0.5999 - age_out_a
ccuracy: 0.0476 - val_loss: 13.0185 - val_gender_out_loss: 0.5370 - val_age_out_lo
ss: 12.4815 - val_gender_out_accuracy: 0.7273 - val_age_out_accuracy: 0.0394
Epoch 2/30
593/593 [==============================] - 11s 19ms/step - loss: 11.1110 - gender_
out_loss: 0.4776 - age_out_loss: 10.6334 - gender_out_accuracy: 0.7697 - age_out_a
ccuracy: 0.0284 - val_loss: 9.4736 - val_gender_out_loss: 0.4306 - val_age_out_los
s: 9.0430 - val_gender_out_accuracy: 0.8013 - val_age_out_accuracy: 0.0154
Epoch 3/30
593/593 [==============================] - 11s 19ms/step - loss: 9.5563 - gender_o
ut_loss: 0.3977 - age_out_loss: 9.1586 - gender_out_accuracy: 0.8136 - age_out_acc
uracy: 0.0159 - val_loss: 8.6304 - val_gender_out_loss: 0.3534 - val_age_out_loss:
8.2770 - val_gender_out_accuracy: 0.8397 - val_age_out_accuracy: 0.0105
Epoch 4/30
593/593 [==============================] - 11s 19ms/step - loss: 8.5816 - gender_o
ut_loss: 0.3449 - age_out_loss: 8.2368 - gender_out_accuracy: 0.8426 - age_out_acc
uracy: 0.0122 - val_loss: 8.4899 - val_gender_out_loss: 0.3333 - val_age_out_loss:
8.1566 - val_gender_out_accuracy: 0.8372 - val_age_out_accuracy: 0.0070
Epoch 5/30
593/593 [==============================] - 11s 18ms/step - loss: 7.9002 - gender_o
ut_loss: 0.3125 - age_out_loss: 7.5877 - gender_out_accuracy: 0.8555 - age_out_acc
uracy: 0.0096 - val_loss: 7.3850 - val_gender_out_loss: 0.3187 - val_age_out_loss:
7.0663 - val_gender_out_accuracy: 0.8663 - val_age_out_accuracy: 0.0091
Epoch 6/30
593/593 [==============================] - 12s 20ms/step - loss: 7.5394 - gender_o
ut_loss: 0.2911 - age_out_loss: 7.2483 - gender_out_accuracy: 0.8711 - age_out_acc
uracy: 0.0094 - val_loss: 7.2500 - val_gender_out_loss: 0.2942 - val_age_out_loss:
6.9559 - val_gender_out_accuracy: 0.8739 - val_age_out_accuracy: 0.0067
Epoch 7/30
593/593 [==============================] - 11s 18ms/step - loss: 7.1528 - gender_o
ut_loss: 0.2784 - age_out_loss: 6.8744 - gender_out_accuracy: 0.8776 - age_out_acc
uracy: 0.0087 - val_loss: 7.8145 - val_gender_out_loss: 0.2707 - val_age_out_loss:
7.5439 - val_gender_out_accuracy: 0.8815 - val_age_out_accuracy: 0.0055
Epoch 8/30
593/593 [==============================] - 11s 19ms/step - loss: 6.9654 - gender_o
ut_loss: 0.2589 - age_out_loss: 6.7065 - gender_out_accuracy: 0.8863 - age_out_acc
uracy: 0.0084 - val_loss: 7.2427 - val_gender_out_loss: 0.2779 - val_age_out_loss:
6.9649 - val_gender_out_accuracy: 0.8817 - val_age_out_accuracy: 0.0038
Epoch 9/30
593/593 [==============================] - 11s 19ms/step - loss: 6.6096 - gender_o
ut_loss: 0.2490 - age_out_loss: 6.3607 - gender_out_accuracy: 0.8920 - age_out_acc
uracy: 0.0067 - val_loss: 7.6145 - val_gender_out_loss: 0.2615 - val_age_out_loss:
7.3530 - val_gender_out_accuracy: 0.8887 - val_age_out_accuracy: 0.0049
Epoch 10/30
593/593 [==============================] - 11s 19ms/step - loss: 6.3424 - gender_o
ut_loss: 0.2342 - age_out_loss: 6.1082 - gender_out_accuracy: 0.8993 - age_out_acc
uracy: 0.0075 - val_loss: 6.8519 - val_gender_out_loss: 0.2631 - val_age_out_loss:
6.5889 - val_gender_out_accuracy: 0.8849 - val_age_out_accuracy: 0.0038
Epoch 11/30
593/593 [==============================] - 11s 19ms/step - loss: 6.0887 - gender_o
ut_loss: 0.2243 - age_out_loss: 5.8644 - gender_out_accuracy: 0.9025 - age_out_acc
uracy: 0.0061 - val_loss: 6.8296 - val_gender_out_loss: 0.2609 - val_age_out_loss:
6.5687 - val_gender_out_accuracy: 0.8912 - val_age_out_accuracy: 0.0051
Epoch 12/30
593/593 [==============================] - 11s 19ms/step - loss: 5.8805 - gender_o
ut_loss: 0.2127 - age_out_loss: 5.6678 - gender_out_accuracy: 0.9091 - age_out_acc
uracy: 0.0063 - val_loss: 6.8389 - val_gender_out_loss: 0.2578 - val_age_out_loss:
6.5811 - val_gender_out_accuracy: 0.8874 - val_age_out_accuracy: 0.0038
Epoch 13/30
593/593 [==============================] - 11s 19ms/step - loss: 5.6329 - gender_o
ut_loss: 0.2008 - age_out_loss: 5.4322 - gender_out_accuracy: 0.9172 - age_out_acc
uracy: 0.0065 - val_loss: 6.7650 - val_gender_out_loss: 0.2749 - val_age_out_loss:
```

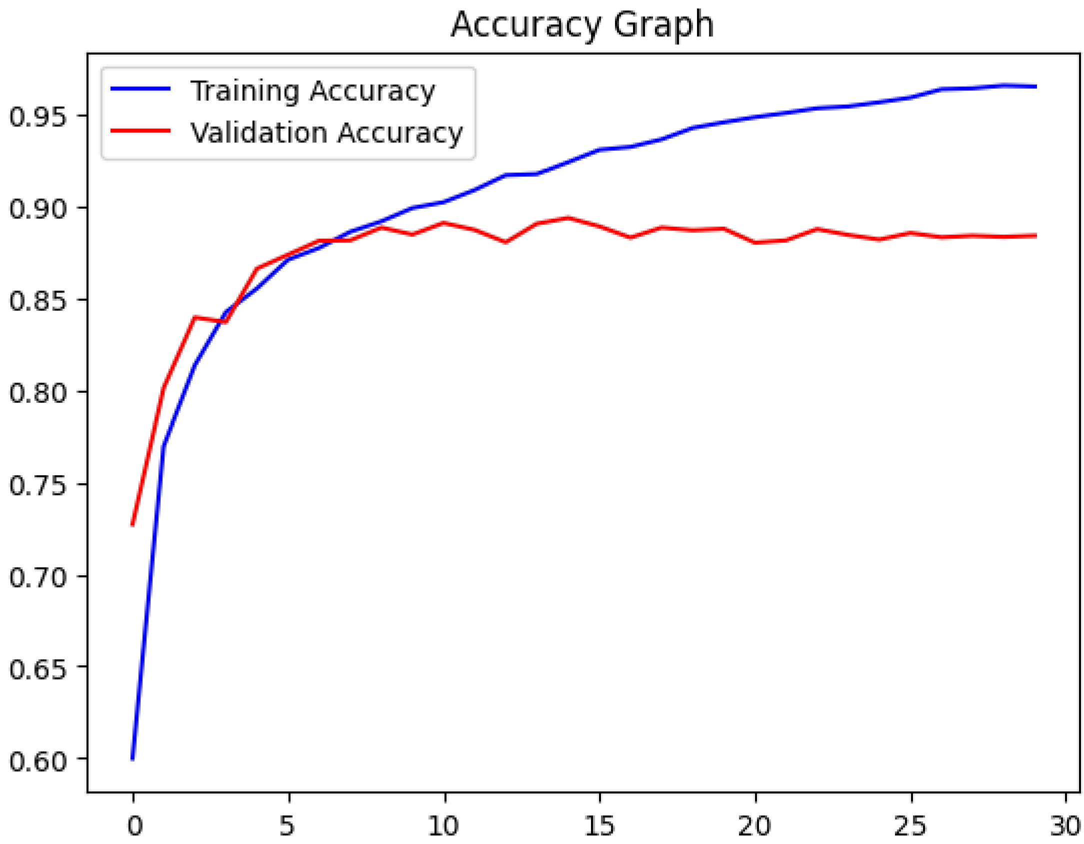
```
6.4901 - val_gender_out_accuracy: 0.8806 - val_age_out_accuracy: 0.0042
Epoch 14/30
593/593 [==============================] - 12s 20ms/step - loss: 5.5173 - gender_o
ut_loss: 0.1945 - age_out_loss: 5.3228 - gender_out_accuracy: 0.9178 - age_out_acc
uracy: 0.0064 - val_loss: 6.7859 - val_gender_out_loss: 0.2525 - val_age_out_loss:
6.5334 - val_gender_out_accuracy: 0.8908 - val_age_out_accuracy: 0.0023
Epoch 15/30
593/593 [==============================] - 11s 19ms/step - loss: 5.3792 - gender_o
ut_loss: 0.1807 - age_out_loss: 5.1986 - gender_out_accuracy: 0.9243 - age_out_acc
uracy: 0.0057 - val_loss: 6.7218 - val_gender_out_loss: 0.2672 - val_age_out_loss:
6.4546 - val_gender_out_accuracy: 0.8939 - val_age_out_accuracy: 0.0040
Epoch 16/30
593/593 [==============================] - 11s 19ms/step - loss: 5.1514 - gender_o
ut_loss: 0.1701 - age_out_loss: 4.9813 - gender_out_accuracy: 0.9310 - age_out_acc
uracy: 0.0053 - val_loss: 6.9818 - val_gender_out_loss: 0.2757 - val_age_out_loss:
6.7061 - val_gender_out_accuracy: 0.8893 - val_age_out_accuracy: 0.0032
Epoch 17/30
593/593 [==============================] - 11s 19ms/step - loss: 5.0442 - gender_o
ut_loss: 0.1636 - age_out_loss: 4.8806 - gender_out_accuracy: 0.9326 - age_out_acc
uracy: 0.0058 - val_loss: 6.8968 - val_gender_out_loss: 0.2777 - val_age_out_loss:
6.6191 - val_gender_out_accuracy: 0.8832 - val_age_out_accuracy: 0.0044
Epoch 18/30
593/593 [==============================] - 11s 19ms/step - loss: 4.8921 - gender_o
ut_loss: 0.1557 - age_out_loss: 4.7363 - gender_out_accuracy: 0.9367 - age_out_acc
uracy: 0.0060 - val_loss: 7.0129 - val_gender_out_loss: 0.2804 - val_age_out_loss:
6.7325 - val_gender_out_accuracy: 0.8887 - val_age_out_accuracy: 0.0046
Epoch 19/30
593/593 [==============================] - 11s 19ms/step - loss: 4.7872 - gender_o
ut_loss: 0.1465 - age_out_loss: 4.6406 - gender_out_accuracy: 0.9429 - age_out_acc
uracy: 0.0062 - val_loss: 6.9136 - val_gender_out_loss: 0.2832 - val_age_out_loss:
6.6304 - val_gender_out_accuracy: 0.8872 - val_age_out_accuracy: 0.0049
Epoch 20/30
593/593 [==============================] - 11s 19ms/step - loss: 4.6335 - gender_o
ut_loss: 0.1334 - age_out_loss: 4.5001 - gender_out_accuracy: 0.9460 - age_out_acc
uracy: 0.0076 - val_loss: 6.9305 - val_gender_out_loss: 0.3056 - val_age_out_loss:
6.6249 - val_gender_out_accuracy: 0.8880 - val_age_out_accuracy: 0.0061
Epoch 21/30
593/593 [==============================] - 11s 19ms/step - loss: 4.5393 - gender_o
ut_loss: 0.1273 - age_out_loss: 4.4120 - gender_out_accuracy: 0.9488 - age_out_acc
uracy: 0.0083 - val_loss: 6.8502 - val_gender_out_loss: 0.3047 - val_age_out_loss:
6.5455 - val_gender_out_accuracy: 0.8804 - val_age_out_accuracy: 0.0091
Epoch 22/30
593/593 [==============================] - 11s 18ms/step - loss: 4.3865 - gender_o
ut_loss: 0.1222 - age_out_loss: 4.2643 - gender_out_accuracy: 0.9511 - age_out_acc
uracy: 0.0098 - val_loss: 6.8959 - val_gender_out_loss: 0.3685 - val_age_out_loss:
6.5275 - val_gender_out_accuracy: 0.8817 - val_age_out_accuracy: 0.0078
Epoch 23/30
593/593 [==============================] - 12s 20ms/step - loss: 4.3326 - gender_o
ut_loss: 0.1145 - age_out_loss: 4.2181 - gender_out_accuracy: 0.9536 - age_out_acc
uracy: 0.0149 - val_loss: 7.3191 - val_gender_out_loss: 0.3126 - val_age_out_loss:
7.0065 - val_gender_out_accuracy: 0.8878 - val_age_out_accuracy: 0.0099
Epoch 24/30
593/593 [==============================] - 11s 19ms/step - loss: 4.2218 - gender_o
ut_loss: 0.1090 - age_out_loss: 4.1128 - gender_out_accuracy: 0.9546 - age_out_acc
uracy: 0.0209 - val_loss: 6.9357 - val_gender_out_loss: 0.3510 - val_age_out_loss:
6.5846 - val_gender_out_accuracy: 0.8846 - val_age_out_accuracy: 0.0291
Epoch 25/30
593/593 [==============================] - 11s 19ms/step - loss: 4.2507 - gender_o
ut_loss: 0.1035 - age_out_loss: 4.1472 - gender_out_accuracy: 0.9569 - age_out_acc
uracy: 0.0222 - val_loss: 7.0294 - val_gender_out_loss: 0.4417 - val_age_out_loss:
6.5877 - val_gender_out_accuracy: 0.8821 - val_age_out_accuracy: 0.0179
Epoch 26/30
593/593 [==============================] - 11s 19ms/step - loss: 4.0607 - gender_o
ut_loss: 0.0997 - age_out_loss: 3.9610 - gender_out_accuracy: 0.9594 - age_out_acc
```

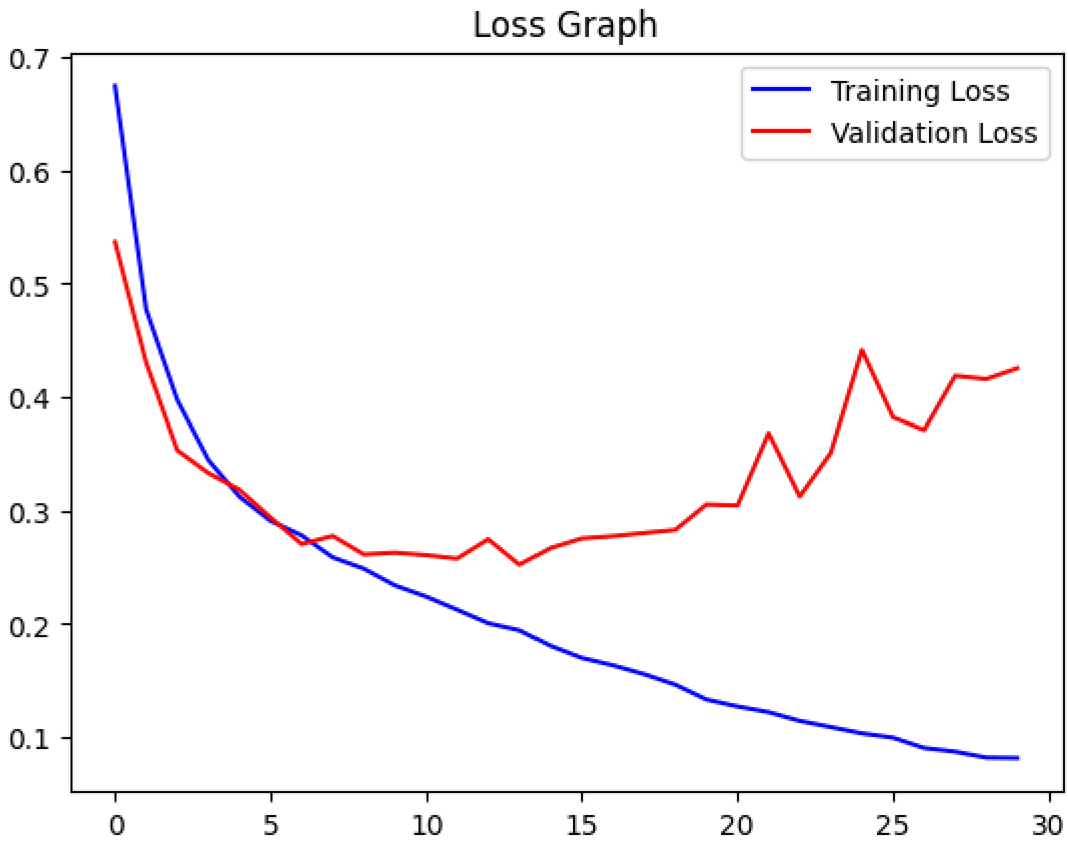
```
uracy: 0.0274 - val_loss: 6.9872 - val_gender_out_loss: 0.3825 - val_age_out_loss:
6.6047 - val_gender_out_accuracy: 0.8857 - val_age_out_accuracy: 0.0255
Epoch 27/30
593/593 [==============================] - 11s 19ms/step - loss: 4.0060 - gender_o
ut_loss: 0.0906 - age_out_loss: 3.9155 - gender_out_accuracy: 0.9639 - age_out_acc
uracy: 0.0334 - val_loss: 6.9073 - val_gender_out_loss: 0.3711 - val_age_out_loss:
6.5362 - val_gender_out_accuracy: 0.8834 - val_age_out_accuracy: 0.0257
Epoch 28/30
593/593 [==============================] - 12s 20ms/step - loss: 3.9119 - gender_o
ut_loss: 0.0873 - age_out_loss: 3.8246 - gender_out_accuracy: 0.9645 - age_out_acc
uracy: 0.0282 - val_loss: 7.1981 - val_gender_out_loss: 0.4189 - val_age_out_loss:
6.7792 - val_gender_out_accuracy: 0.8842 - val_age_out_accuracy: 0.0238
Epoch 29/30
593/593 [==============================] - 11s 19ms/step - loss: 3.8396 - gender_o
ut_loss: 0.0821 - age_out_loss: 3.7575 - gender_out_accuracy: 0.9660 - age_out_acc
uracy: 0.0312 - val_loss: 6.9846 - val_gender_out_loss: 0.4160 - val_age_out_loss:
6.5686 - val_gender_out_accuracy: 0.8836 - val_age_out_accuracy: 0.0329
Epoch 30/30
593/593 [==============================] - 12s 20ms/step - loss: 3.7756 - gender_o
ut_loss: 0.0817 - age_out_loss: 3.6939 - gender_out_accuracy: 0.9655 - age_out_acc
uracy: 0.0346 - val_loss: 7.0667 - val_gender_out_loss: 0.4255 - val_age_out_loss:
6.6412 - val_gender_out_accuracy: 0.8842 - val_age_out_accuracy: 0.0375
```

Plot the Results

In [20]:

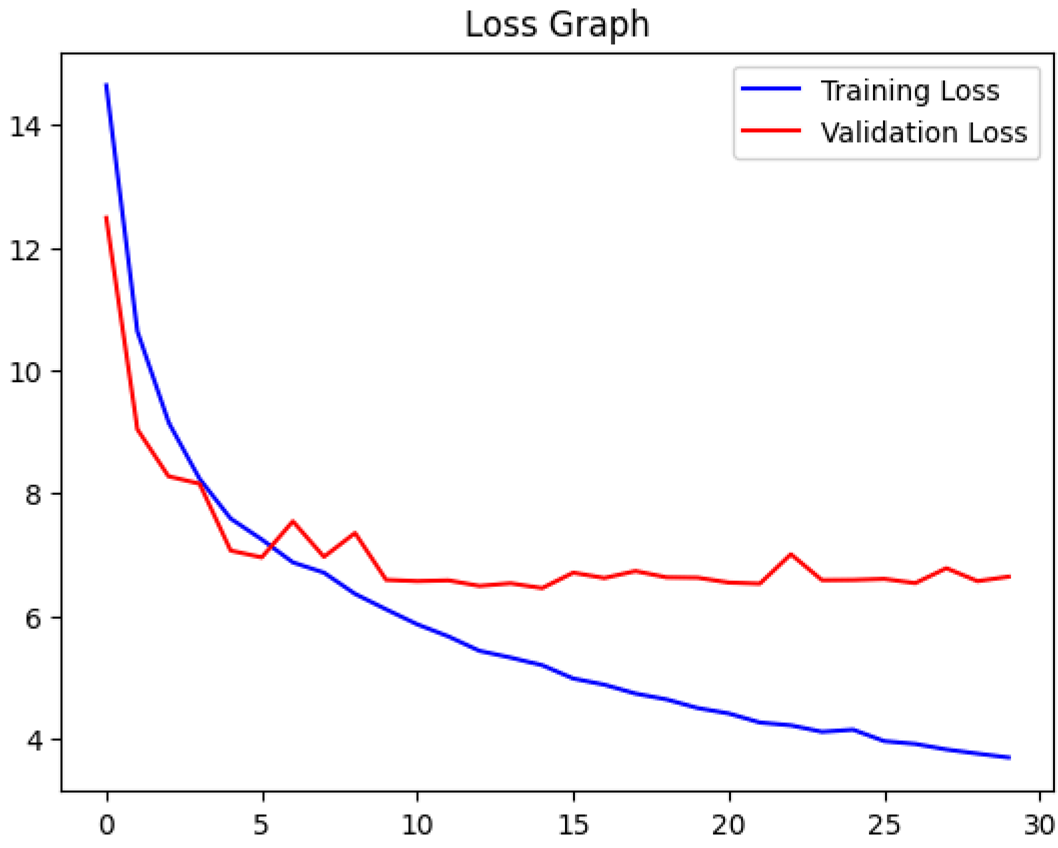
```python
# plot results for gender
acc = history.history['gender_out_accuracy']
val_acc = history.history['val_gender_out_accuracy']
epochs = range(len(acc))

plt.plot(epochs, acc, 'b', label='Training Accuracy')
plt.plot(epochs, val_acc, 'r', label='Validation Accuracy')
plt.title('Accuracy Graph')
plt.legend()
plt.figure()

loss = history.history['gender_out_loss']
val_loss = history.history['val_gender_out_loss']

plt.plot(epochs, loss, 'b', label='Training Loss')
plt.plot(epochs, val_loss, 'r', label='Validation Loss')
plt.title('Loss Graph')
plt.legend()
plt.show()
```





In [21]:
```
# plot results for age
loss = history.history['age_out_loss']
val_loss = history.history['val_age_out_loss']
epochs = range(len(loss))

plt.plot(epochs, loss, 'b', label='Training Loss')
plt.plot(epochs, val_loss, 'r', label='Validation Loss')
plt.title('Loss Graph')
```

```
plt.legend()
plt.show()
```



In [26]:
```
image_index = 10
print("Original Gender:", gender_dict[y_gender[image_index]], "Original Age:", y_ag
# predict from model
pred = model.predict(X[image_index].reshape(1, 128, 128, 1))
pred_gender = gender_dict[round(pred[0][0][0])]
pred_age = round(pred[1][0][0])
print("Predicted Gender:", pred_gender, "Predicted Age:", pred_age)
plt.axis('off')
plt.imshow(X[image_index].reshape(128, 128), cmap='gray');
```

```
Original Gender: Female Original Age: 18
1/1 [==============================] - 0s 19ms/step
Predicted Gender: Female Predicted Age: 17
```

In [27]:

```
image_index = 300
print("Original Gender:", gender_dict[y_gender[image_index]], "Original Age:", y_ag
# predict from model
pred = model.predict(X[image_index].reshape(1, 128, 128, 1))
pred_gender = gender_dict[round(pred[0][0][0])]
pred_age = round(pred[1][0][0])
print("Predicted Gender:", pred_gender, "Predicted Age:", pred_age)
plt.axis('off')
plt.imshow(X[image_index].reshape(128, 128), cmap='gray');
```

```
Original Gender: Male Original Age: 21
1/1 [==============================] - 0s 20ms/step
Predicted Gender: Male Predicted Age: 22
```

In [30]:
```
image_index = 2500
print("Original Gender:", gender_dict[y_gender[image_index]], "Original Age:", y_ag
# predict from model
pred = model.predict(X[image_index].reshape(1, 128, 128, 1))
pred_gender = gender_dict[round(pred[0][0][0])]
pred_age = round(pred[1][0][0])
print("Predicted Gender:", pred_gender, "Predicted Age:", pred_age)
plt.axis('off')
plt.imshow(X[image_index].reshape(128, 128), cmap='gray');
```

```
Original Gender: Male Original Age: 56
1/1 [==============================] - 0s 20ms/step
Predicted Gender: Male Predicted Age: 56
```

In [31]:
```
image_index = 2567
print("Original Gender:", gender_dict[y_gender[image_index]], "Original Age:", y_ag
# predict from model
pred = model.predict(X[image_index].reshape(1, 128, 128, 1))
pred_gender = gender_dict[round(pred[0][0][0])]
pred_age = round(pred[1][0][0])
print("Predicted Gender:", pred_gender, "Predicted Age:", pred_age)
plt.axis('off')
plt.imshow(X[image_index].reshape(128, 128), cmap='gray');
```

```
Original Gender: Female Original Age: 10
1/1 [==============================] - 0s 20ms/step
Predicted Gender: Female Predicted Age: 10
```

In [34]:
```
image_index = 1577
print("Original Gender:", gender_dict[y_gender[image_index]], "Original Age:", y_ag
# predict from model
pred = model.predict(X[image_index].reshape(1, 128, 128, 1))
pred_gender = gender_dict[round(pred[0][0][0])]
pred_age = round(pred[1][0][0])
print("Predicted Gender:", pred_gender, "Predicted Age:", pred_age)
plt.axis('off')
plt.imshow(X[image_index].reshape(128, 128), cmap='gray');
```

```
Original Gender: Female Original Age: 26
1/1 [==============================] - 0s 27ms/step
Predicted Gender: Female Predicted Age: 26
```