```
import numpy as np
import matplotlib.pyplot as plt
import os
import tensorflow as tf
import keras
from keras import utils
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout
from tensorflow.keras.optimizers import Adam

data_dir = '/content/drive/MyDrive/Bone Break Classification'


train_data = utils.image_dataset_from_directory(
    data_dir,
    labels="inferred",
    label_mode="int",
    validation_split=0.1,
    subset="training",
    shuffle=True,
    color_mode="rgb",
    image_size=(256, 256),
    batch_size=64,
    seed=40,
)

vald_data = utils.image_dataset_from_directory(
    data_dir,
    labels="inferred",
    label_mode="int",
    validation_split=0.1,
    subset="validation",
    color_mode="rgb",
    image_size=(256, 256),
    batch_size=64,
    seed=40,
)

def normalize(image, label):
    return image / 255.0, label

train_data = train_data.map(normalize)
vald_data = vald_data.map(normalize)

train_x = []
train_y = []
for image, label in train_data:
    train_x.append(image)
    train_y.append(label)

train_x = tf.concat(train_x, axis=0)
train_y = tf.concat(train_y, axis=0)

val_x = []
val_y = []
for image, label in vald_data:
    val_x.append(image)
    val_y.append(label)

val_x = tf.concat(val_x, axis=0)
val_y = tf.concat(val_y, axis=0)

num_classes = 10
train_y = tf.keras.utils.to_categorical(train_y, num_classes=num_classes)
val_y = tf.keras.utils.to_categorical(val_y, num_classes=num_classes)

class_labels = [
    "Avulsion fracture", "Comminuted fracture", "Fracture Dislocation", "Greenstick fracture",
    "Hairline Fracture", "Impacted fracture", "Longitudinal fracture", "Oblique fracture",
    "Pathological fracture", "Spiral Fracture"
]

fig, axes = plt.subplots(2, 4, figsize=(15, 5))
```

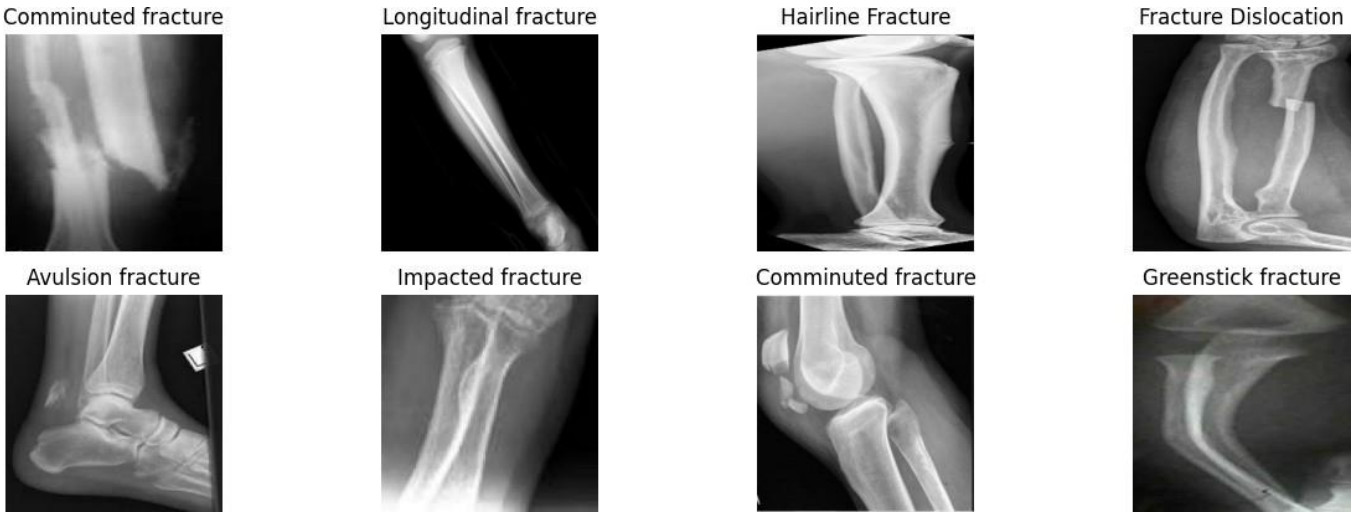
```
for i, ax in enumerate(axes.flat):
    image, label = train_x[i], train_y[i]

    ax.imshow(image, cmap='gray')

    ax.set_title(f"{class_labels[np.argmax(label)]}")
    ax.axis('off')

plt.show()
```

```
Found 1139 files belonging to 10 classes.
Using 1026 files for training.
Found 1139 files belonging to 10 classes.
Using 113 files for validation.
```



Akash Dutta