



**TECNOLÓGICO
NACIONAL DE MÉXICO**



Instituto Tecnológico de Culiacán

Carrera: Ingeniería en Sistemas Computacionales

Materia: Inteligencia Artificial

Profesor: Zuriel Dathan Mora Félix

Tarea: Adquisición de imágenes para la clasificación de emociones
y preprocesamiento

Grupo:

11:00 AM – 12:00 PM

Equipo:

García Pérez José Ángel

Verdugo Bermúdez Sebastián

Adquisición de Imágenes para la clasificación de emociones

Sign InRegister

MANAS SAMBARE · UPDATED 5 YEARS AGO

1463CodeDownload

FER-2013

Learn facial expressions from an image

FearHappyNeutral

Data CardCode (579)Discussion (8)Suggestions (6)

About Dataset

The data consists of 48×48 pixel grayscale images of faces. The faces have been automatically registered so that the face is more or less centred and occupies about the same amount of space in each image.

The task is to categorize each face based on the emotion shown in the facial expression into one of seven categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). The training set consists of 28,709 examples and the public test set consists of 3,589 examples.

test (7 directories)

Usability

7.50

License

Database: Open Database, Cont...

Expected update frequency

Not specified

Tags

Arts and Entertainment

Art

Data Explorer

Version 1 (56.51 MB)

test

angry

disgust

fear

happy

Utilizamos este dataset ya que nos resultó muy útil debido a la gran variedad de imágenes para el reconocedor de emociones.

Preprocesamiento

```
import fiftyone as fo
import os
import cv2
import albumentations as A

# Directorio raíz donde se encuentran los datos organizados por carpetas de emociones
dataset_dir = "./fer2013_raw"

# Función para cargar imágenes desde carpetas etiquetadas (una carpeta por clase)
def cargar_dataset(ruta):
    nombre = os.path.basename(ruta) # Nombre del conjunto (train o test)
    return fo.Dataset.from_dir(
        dataset_dir=ruta,
        dataset_type=fo.types.ImageClassificationDirectoryTree, # Carga etiquetas desde la estructura de carpetas
        name=nombre
    )

# Cargar datasets de entrenamiento y prueba
train_dataset = cargar_dataset(os.path.join(dataset_dir, "train"))
test_dataset = cargar_dataset(os.path.join(dataset_dir, "test"))

# Crear un nuevo dataset combinando train y test
dataset = fo.Dataset(name="fer2013_full")
dataset.add_samples(train_dataset) # Añadir muestras de entrenamiento
dataset.add_samples(test_dataset) # Añadir muestras de prueba

# Definir las transformaciones de aumento de datos usando Albumentations
transform = A.Compose([
    A.Rotate(limit=25, p=0.8), # Rotar aleatoriamente hasta 25 grados
    A.RandomScale(scale_limit=0.2, p=0.8), # Escalar aleatoriamente ±20%
    A.RandomBrightnessContrast(brightness_limit=0.3, contrast_limit=0.3, p=0.9), # Cambiar brillo y contraste
    A.HorizontalFlip(p=0.5), # Voltear horizontalmente
])

# Directorio donde se guardarán las imágenes aumentadas
output_dir = "fer2013_augmented"
os.makedirs(output_dir, exist_ok=True)

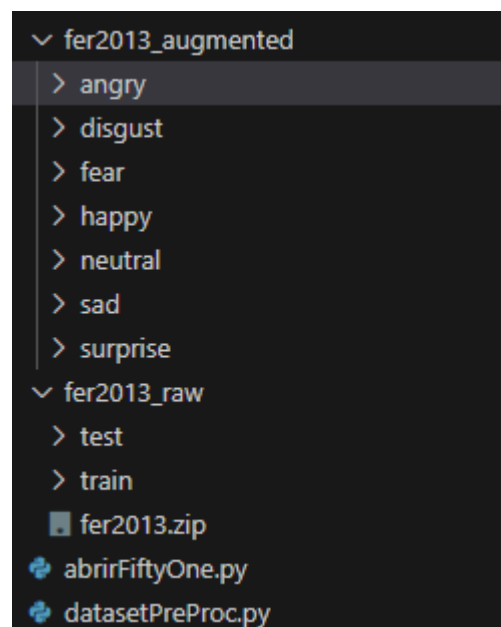
new_samples = [] # Lista para almacenar las nuevas muestras generadas
```

```

1 # Aplicar transformaciones y guardar nuevas imágenes
2 for i, sample in enumerate(dataset):
3     label = sample.ground_truth.label # Obtener la etiqueta de la emoción
4     label_dir = os.path.join(output_dir, label) # Crear subcarpeta por clase
5     os.makedirs(label_dir, exist_ok=True)
6
7     # Leer y convertir imagen
8     img = cv2.imread(sample.filepath)
9     img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
10
11     # Aplicar transformaciones
12     transformed = transform(image=img)["image"]
13     transformed = cv2.cvtColor(transformed, cv2.COLOR_RGB2BGR)
14
15     # Guardar imagen transformada en carpeta correspondiente
16     filename = f"{i:05d}.jpg"
17     new_path = os.path.join(label_dir, filename)
18     cv2.imwrite(new_path, transformed)
19
20     # Crear nuevo sample con la misma etiqueta
21     new_sample = fo.Sample(filepath=new_path)
22     new_sample["ground_truth"] = fo.Classification(label=label)
23     new_sample.tags.append("augmented") # Etiqueta adicional para identificar muestras aumentadas
24     new_samples.append(new_sample)
25
26 # Crear un nuevo dataset con las imágenes aumentadas
27 processed_dataset = fo.Dataset(name="fer2013_augmented")
28 processed_dataset.add_samples(new_samples)
29
30 # Iniciar aplicación de FiftyOne para visualizar las muestras
31 session = fo.launch_app(processed_dataset, port=5151)
32 session.wait()

```

Utilizamos este código que nos ayuda a realizar el preprocesamiento de las imágenes para saber a qué emoción pertenece, por ejemplo, si es cambio de intensidad de luz, rotaciones o cambios de escala. En la siguiente imagen se puede observar cómo nos agregó una carpeta en donde aparecen las imágenes ya clasificadas por su tipo de emoción, y esto fue para cada emoción registrada.





En esta última imagen están clasificadas las imágenes por su tipo de emoción en fiftyone optamos por esta aplicación ya que nos visualizar y la gestión del dataset que seleccionamos, se puede ver que esta etiquetada por su emoción correspondiente en este caso tristeza.