

Proyecto I

Clasificación y localización de objetos

13 de febrero de 2023

1. Introducción

El reto que se propone en este proyecto consiste en construir un modelo para el problema de clasificación y localización de objetos en imágenes. Para entrenar, validar y probar el modelo, utilizaremos el conjunto de datos **Tiny Imagenet** que tiene 200 clases. Cada clase tiene 500 imágenes de entrenamiento, 50 imágenes de validación y 50 imágenes de prueba.

1.1. Conjunto de datos

Puedes descargar los datos utilizando el siguiente enlace [\[dataset\]](#). La figura 1 muestra el contenido del archivo descargado y descomprimido.



Figura 1: Contenido del archivo dataset.zip

El archivo *wnids.txt* contiene los identificadores de las 200 clases a las que pertenecen las imágenes, mientras que en el archivo *words.txt* se encuentra la descripción de las clases. En la figura 2 y 3 puedes observar una muestra del contenido de los archivos:

```
1 n02124075
2 n04067472
3 n04540053
4 n04099969
5 n07749582
6 n01641577
7 n02802426
8 n09246464
9 n07920052
10 n03970156
11 n03891332
12 n02106662
13 n03201208
14 n02279972
```

Figura 2: Muestra del contenido del archivo *wnids.txt*

1	n00001740	entity
2	n00001930	physical entity
3	n00002137	abstraction, abstract entity
4	n00002452	thing
5	n00002684	object, physical object
6	n00003553	whole, unit
7	n00003993	congener
8	n00004258	living thing, animate thing
9	n00004475	organism, being
10	n00005787	benthos
11	n00005930	dwarf
12	n00006024	heterotroph
13	n00006150	parent
14	n00006269	life

Figura 3: Muestra del contenido del archivo *words.txt*

1.2. Datos de entrenamiento

Para el desarrollo de tu proyecto se te proporciona un conjunto de datos para entrenamiento que incluye imágenes y anotaciones. En la figura 4 puedes observar el contenido de la carpeta *train*, en donde cada subcarpeta representa una clase (el nombre de la carpeta es el identificador de la clase, por ejemplo: *n01443537*). En la figura 5 puedes ver un ejemplo del contenido de la subcarpeta *n01443537*.

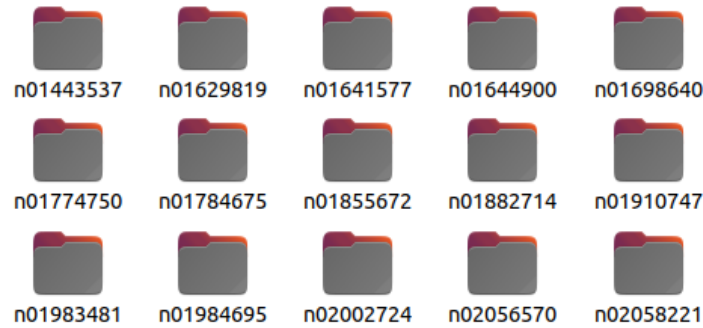


Figura 4: Contenido de la carpeta *train*

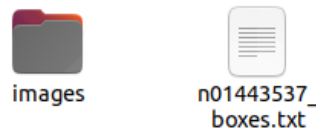


Figura 5: Contenido de la carpeta *n01443537*

En la figura 6 puede observar una muestra de las imágenes que pertenecen a la clase *n01443537*, mientras que en la figura 7 puede ver las anotaciones de cada una de las imágenes (recuerde que las anotaciones representan el cuadro que delimita la localización de un objeto dentro de la imagen).

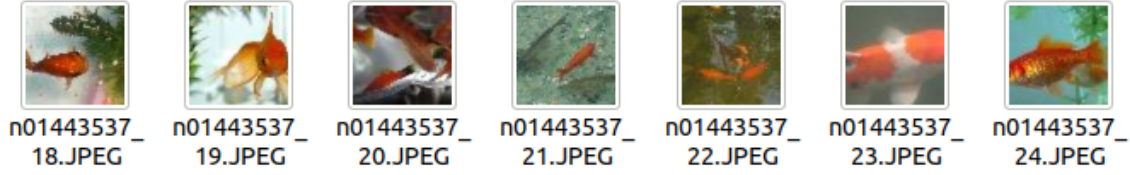


Figura 6: Contenido de la carpeta *n01443537/images*

1	n01443537_0.JPEG	0	10	63	58
2	n01443537_1.JPEG	14	2	54	49
3	n01443537_2.JPEG	0	0	63	63
4	n01443537_3.JPEG	0	16	48	54
5	n01443537_4.JPEG	15	0	51	63
6	n01443537_5.JPEG	0	0	63	63
7	n01443537_6.JPEG	13	21	52	41
8	n01443537_7.JPEG	0	0	63	63
9	n01443537_8.JPEG	0	14	37	36
10	n01443537_9.JPEG	16	18	56	53
11	n01443537_10.JPEG	1	17	58	56
12	n01443537_11.JPEG	0	0	61	52
13	n01443537_12.JPEG	4	3	61	63
14	n01443537_13.JPEG	51	7	63	16

Figura 7: Contenido del archivo *n01443537-boxes.txt*

1.3. Datos de validación

Para validar tu modelo, se te proporciona un conjunto de datos que incluye imágenes, clase a la que pertenecen y sus anotaciones. Estos datos se encuentran en la carpeta *val* (figura 8).

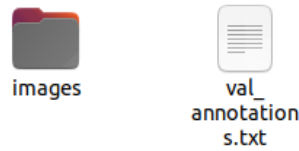


Figura 8: Datos de validación en la carpeta *val*

Las figuras 9 y 10 muestran un ejemplo de las imágenes y sus anotaciones, respectivamente.

1.4. Datos de prueba

Al explorar el conjunto de datos, observarás que los datos de prueba que se proporcionan en la carpeta *test* no incluyen etiquetas ni anotaciones, únicamente tendrás las imágenes para que tu modelo realice la predicción de la ubicación en donde se encuentra el objeto para cada imagen.

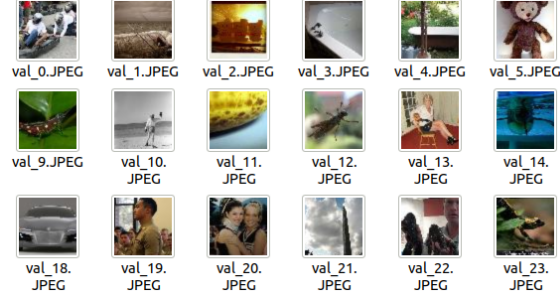


Figura 9: Imágenes de validación que se encuentran en *val/images*

1 val_0.JPEG	n03444034	0	32	44	62
2 val_1.JPEG	n04067472	52	55	57	59
3 val_2.JPEG	n04070727	4	0	60	55
4 val_3.JPEG	n02808440	3	3	63	63
5 val_4.JPEG	n02808440	9	27	63	48
6 val_5.JPEG	n04399382	7	0	59	63
7 val_6.JPEG	n04179913	0	0	63	56
8 val_7.JPEG	n02823428	5	0	57	63
9 val_8.JPEG	n04146614	0	31	60	60
10 val_9.JPEG	n02226429	0	3	63	57
11 val_10.JPEG	n04371430	37	38	44	45
12 val_11.JPEG	n07753592	0	1	63	47
13 val_12.JPEG	n02226429	5	12	57	53
14 val_13.JPEG	n03770439	21	33	36	43

Figura 10: Anotaciones que tiene cada imagen *val_annotations.txt*

2. Evaluación del desempeño

Para este problema, el error en cada predicción se medirá como:

$$e = \max(d_{ij}, f_{ij}) \quad (1)$$

donde

- i es el *bounding box* predicho;
- j es el *bounding box* verdadero.
- $d = 0$ si las etiquetas de los dos *bounding box* son iguales y $d = 1$ en caso contrario;
- $f = 0$ si la superposición de los dos *bounding box* $\geq 50\%$ y $f = 1$ en caso contrario;

Por ejemplo, supongamos que para una imagen dada hay un *bounding box* b_0 , y nosotros predecimos el *bounding box* p_0 . Debemos verificar si se encuentra una coincidencia (match) entre p_0 y b_0 . Si hay una coincidencia, entonces el error mínimo para esta imagen es 0; de lo contrario, el error mínimo es 1.

Para este trabajo, se considera una coincidencia (match) cuando:

- la clase predicha es igual a la clase verdadera del objeto en la imagen, y

- el *bounding box* $p0$ que fue predicho tenga más del 50 % de coincidencia con el área del *bounding box* $b0$ verdadero.

El error total será calculado como la media de los errores resultantes de realizar las predicciones de todas las imágenes del conjunto de datos de prueba.

3. Productos esperados

- *Reporte técnico.* Descripción del desarrollo de su trabajo en el que se documente la su modelo con mejor desempeño, arquitectura y configuración (capas, funciones de activación, funciones de pérdida, métricas, etc.). En este reporte se debe incluir la documentación de desempeño del modelo en las etapas de entrenamiento y validación (gráficas: métricas vs épocas).
- *Archivo del modelo final.* Su archivo debe incluir los pesos aprendidos, parámetros de entrenamiento y arquitectura del modelo.
- *Archivo de predicciones.* Debe generar un archivo de texto plano (*.txt) con las predicciones realizadas por su modelo en el conjunto de datos de prueba. El archivo debe contener dos columnas con 10,000 líneas. Cada línea contiene el nombre del archivo de una imagen de prueba, el identificador de la clase que fue predicha, y cuatro números enteros del *bounding box* predicho. Veamos dos líneas de ejemplo:

```
test_534.JPEG    n01910747    25 25 57 40
test_156.JPEG    n07614500    5 6 63 58
```

4. Calificación

Para determinar que tan bueno es su modelo, se utilizará un conjunto de imágenes de prueba que no forma parte del conjunto de datos proporcionado al estudiante. Si su modelo funciona sin errores de lógica y sintaxis, la calificación se asignará acorde a la siguiente tabla:

Predicciones correctas	Puntos
0 % - 24 %	30
25 % - 54 %	50
55 % - 64 %	60
65 % - 74 %	70
75 % - 84 %	80
85 % - 94 %	90
95 % - 100 %	100