

PYTORCH OBJECT DETECTION WITH PRE-TRAINED NETWORKS

UNIVERSIDAD CENTRAL DEL ECUADOR

FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

INGENIERÍA INFORMATICA



CRISTIAN XAVIER VISCAINO
VIII SEMESTRE

OPTATIVA II

Contenido

1. ¿Qué es PyTorch?.....	2
2. Modelos de detección de objetos de PyTorch	2
3. Arquitectura Faster R-CNN	2
5. Conjunto de Datos “COCO”	3
6. Requisitos de Instalación.....	3
7. Qué es CUDA.....	4
8. Conjunto de Imágenes Analizar	4
9. Conclusiones Iniciales.....	5
10. Referencias	5
11. Notas.....	5

1. ¿Qué es PyTorch?

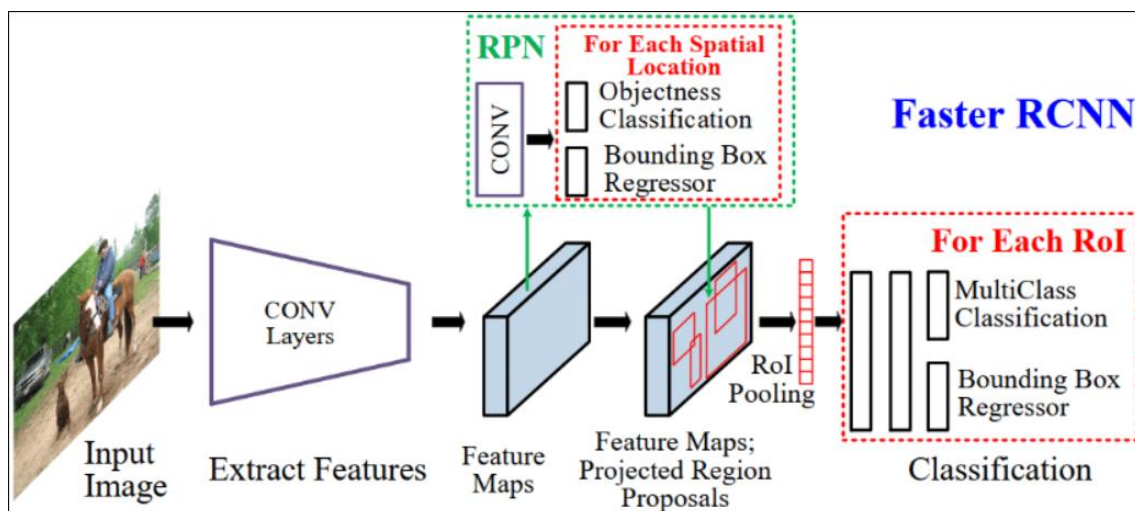
Es una biblioteca de aprendizaje automático de código abierto que se especializa en cálculos de tensores, diferenciación automática y aceleración de GPU. Por esas razones, PyTorch es una de las bibliotecas de Deep Learning más popular.

2. Modelos de detección de objetos de PyTorch

PyTorch nos proporciona tres modelos de detección de objetos:

- Faster R-CNN with a ResNet50 backbone (más precisa, pero más lenta).
- Faster R-CNN with a MobileNet v3 backbone (más rápido, pero menos preciso).
- RetinaNet with a ResNet50 backbone (buen equilibrio entre velocidad y precisión).

3. Arquitectura Faster R-CNN

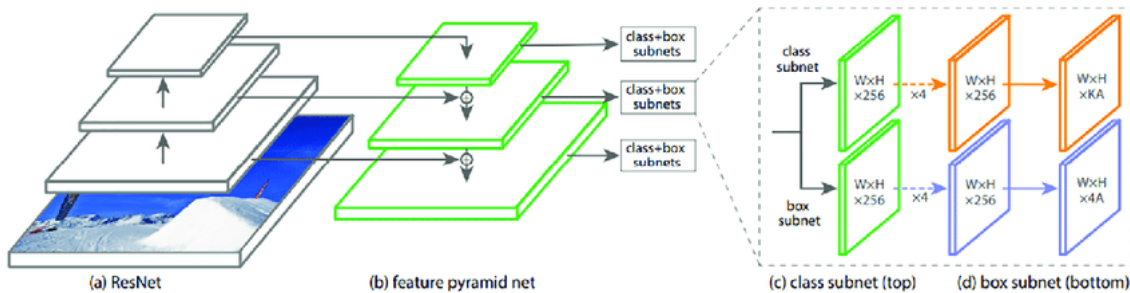


4. Arquitectura RetinaNet

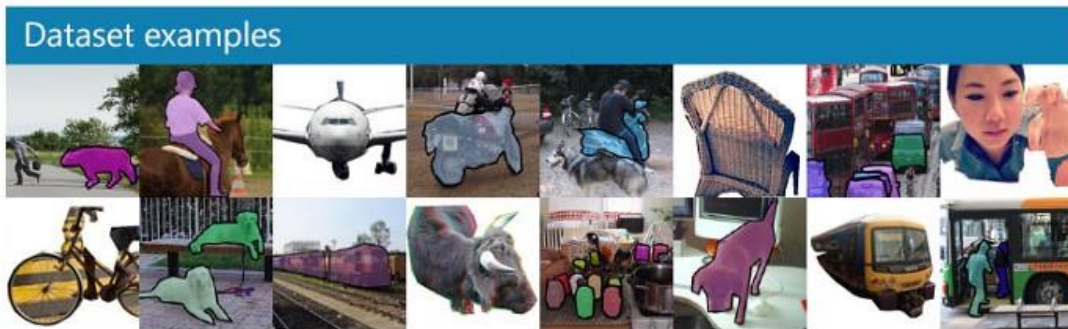
La arquitectura de la Red neuronal convolucional (CNN) RetinaNet consta de 4 partes principales, cada una de las cuales tiene su propio propósito:

- Backbone:** la red principal (básica) utilizada para extraer características de la imagen de entrada. Esta parte de la red es variable y puede incluir redes neuronales de clasificación como ResNet, VGG, EfficientNet y otras.
- Feature Pyramid Net (FPN):** una red neuronal convolucional, construida en forma de pirámide, que sirve para combinar las ventajas de los mapas de características de los niveles inferior y superior de la red; los primeros tienen una capacidad de generalización de alta resolución, pero de baja semántica; los últimos son lo contrario.

- c) **Subred de clasificación:** una subred que extrae información sobre las clases de objetos de FPN, resolviendo el problema de clasificación;
- d) **Subred de regresión:** una subred que extrae información sobre las coordenadas de objetos en la imagen desde FPN, resolviendo el problema de regresión.



El conjunto de datos COCO (Common Objects in Context) tiende a ser el estándar para la evaluación comparativa de detección de objetos.



```
$ pip install torch torchvision
```

El paquete de torchvision consta de conjuntos de datos populares, arquitecturas de modelos y transformaciones de imágenes comunes para la visión por computadora.

OpenCV provee una infraestructura para aplicaciones de visión artificial.










7. Qué es CUDA



CUDA® es una plataforma de computación paralela y un modelo de programación desarrollado por NVIDIA para computación general en unidades de procesamiento gráfico (GPU). Con CUDA, los desarrolladores pueden acelerar drásticamente las aplicaciones informáticas aprovechando la potencia de las GPU.

En las aplicaciones aceleradas por GPU, la parte secuencial de la carga de trabajo se ejecuta en la CPU, que está optimizada para el rendimiento de un solo subproceso, mientras que la parte de procesamiento intensivo de la aplicación se ejecuta en miles de núcleos de GPU en paralelo. Al usar CUDA, los desarrolladores programan en lenguajes populares como C, C ++, Fortran, Python y MATLAB y expresan el paralelismo a través de extensiones en forma de algunas palabras clave básicas.

El kit de herramientas CUDA de NVIDIA proporciona todo lo que necesita para desarrollar aplicaciones aceleradas por GPU. El kit de herramientas CUDA incluye bibliotecas aceleradas por GPU, un compilador, herramientas de desarrollo y el tiempo de ejecución de CUDA.

8. Conjunto de Imágenes Analizar

Example_01	Example_02	Example_03
		
Example_04	Example_05	Example_06
		
Example_07	Example_08	Example_09
		

Example_010	Example_011	
		

9. Conclusiones Iniciales

PyTorch tiende a ser *especialmente* popular entre la comunidad de investigación debido a su naturaleza Pythonic y facilidad de extensibilidad (es decir, implementación de tipos de capas personalizados, arquitecturas de red, etc.).

Dentro de los tres modelos que ofrece PyTorch, podemos observar que el modelo Faster R-CNN ofrece un mayor porcentaje en la detección de objetos en comparación con los demás modelos, además ofrece un mayor nivel en lo que se refiere a números de objetos detectados en una imagen.

Se incluye un documento Excel de nombre “Cuadro Comparativo 1.0” que contiene una comparación de los tres modelos con sus respectivos porcentajes de detección de objetos.

10. Referencias

<https://www.pyimagesearch.com/2021/08/02/pytorch-object-detection-with-pre-trained-networks/>

<https://tech-es.netlify.app/articles/es510560/index.html>

11. Notas

El conjunto de Imágenes se incluye dentro de la carpeta “images” dentro de la carpeta raíz “pytorch-object-detection” que pertenece al Proyecto Individual “Proyecto_019”.

Adicionalmente el código para la detección de objetos, se encuentra dentro de la carpeta raíz “pytorch-object-detection” con el nombre “pytorch_object_detection.ipynb”.

Dentro del código para probar las imágenes y los tres modelos que ofrece PyTorch por defecto se encuentra usando la imagen `"image": "images/example_01.jpg"` y el modelo `"model": "frcnn-resnet"`. Para probar las demás imágenes y modelos es necesario comentar la imagen y el modelo que se encuentran con sus respectivos comentarios dentro del código.