



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



TFG del Grado en Ingeniería
Informática

Aplicaciones de Visión
Artificial en Dispositivos de
Edge Computing



Presentado por Miriam Torres Calvo
en Universidad de Burgos — 23 de junio
de 2022

Tutor: Bruno Baruque Zanón



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. Bruno Baruque Zanón, profesor del departamento de Ingeniería Informática., área de Ciencia de la Computación e Inteligencia Artificial.

Expone:

Que la alumna D^a. Miriam Torres Calvo, con DNI 45575901K, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado título de TFG.

Y que dicho trabajo ha sido realizado por la alumna bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 23 de junio de 2022

Vº. Bº. del Tutor:

D. Bruno Baruque Zanón

Resumen

La tecnología avanza continuamente y a velocidades que hace unos años eran inexplicables, contamos su ayuda en diferentes entornos de la sociedad y cada vez en mayor medida. Pero también, contamos con situaciones en las cuáles está no se encuentra tan fácil de acceder, ya que no se cuenta con la facilidad para transportarla y así poder usarla en muchos mas lugares y mas comodamente.

Así surge este proyecto, con la idea de poder usar un modelo de Machine Learning en dispositivos de Edge Computing, como puede ser la Jetson Nano de NVIDIA.

Para su desarrollo, se contara con el lenguaje Python y el modelo escogido para entrenar ha sido YOLO en su cuarta versión.

Descriptores

Deep Learning, Edge Computing, Jetson Nano, YOLO, Python Object Detection, Trasnsfer Learning

Abstract

Technology is advancing continuously and at speeds that were inexplicable a few years ago, we count on your help in different environments of society and to a greater extent. But also, we have situations in which it is not so easy to access, since it is not easy to transport it and thus be able to use it in many more places. and more comfortably.

This is how this project arose, with the idea of being able to use a Machine Learning model in Edge Computing devices, such as the NVIDIA Jetson Nano.

For its development, the Python language will be used and the model chosen for training has been YOLO in its fourth version.

Keywords

Deep Learning, Edge Computing, Jetson Nano, YOLO, Python Object Detection, Trasnsfer Learning

Índice general

Índice general	iii
Índice de figuras	iv
Índice de tablas	v
Introducción	1
Objetivos del proyecto	3
Conceptos teóricos	5
3.1. Deep Learning	5
3.2. Edge Computing	6
3.3. Jetson Nano	6
3.4. YOLO	7
3.5. Object Detection	8
Técnicas y herramientas	9
Aspectos relevantes del desarrollo del proyecto	11
Trabajos relacionados	13
Conclusiones y Líneas de trabajo futuras	15
Bibliografía	17

Índice de figuras

3.1. Red neuronal convolucional	6
3.2. Jetson Nano	7
3.3. Explicación YOLO	8
3.4. Detección de objetos	8

Índice de tablas

Introducción

Hoy en día la tecnología ha avanzado tanto, que es muy fácil contar con ella a la hora de realizar ciertas tareas, pero cada vez la demandamos más para poder trabajar codo con codo con ella, es decir, recibir su ayuda de tal forma que les podamos asignar tareas asegurando que tendrán un porcentaje de acierto igual o superior al que tendría si lo realizásemos cualquiera de nosotros.

Pero generalmente, para poder llevar a cabo estas tareas, se necesitan dispositivos con una gran cantidad de compute, ya que necesitaremos entrenarlo con el objeto u objetos a predecir, siendo está la tarea más importante y la que más capacidad de compute va a necesitar y la que más recursos va consumir. Tras su entrenamiento, volveremos a consumir recursos para su detección, de tal forma que necesitaremos un equipo lo suficientemente potente para poder realizar ambas tareas con efectividad y poder obtener buenos resultados.

Debido a esto, el poder entrenar el modelo en un ordenador lo suficientemente potente y seguidamente poder adaptarlo para poder ser utilizado en dispositivos pequeños como puede ser la Jetson Nano de NVIDIA, y que este dispositivo lo ejecute, sacrificando el porcentaje de acierto pero respetando los tiempos de ejecución, puede facilitar a muchos trabajadores y/o investigadores en sus trabajos ya que pueden tener una herramienta funcional en poco espacio y además fácil de transportar para poder usarla en diferentes lugares.

Objetivos del proyecto

Este apartado explica de forma precisa y concisa cuales son los objetivos que se persiguen con la realización del proyecto. Se puede distinguir entre los objetivos marcados por los requisitos del software a construir y los objetivos de carácter técnico que plantea a la hora de llevar a la práctica el proyecto.

Conceptos teóricos

Para la compresion de este proyecto, se deben conocer los siguientes conceptos:

3.1. Deep Learning

El deep learning [1] es una rama del MachineLearning, donde los algoritmos inspirados en el funcionamiento del cerebro humano (redes neurales) aprenden a partir de grandes cantidades de datos y tratan con un alto número de unidades computacionales.

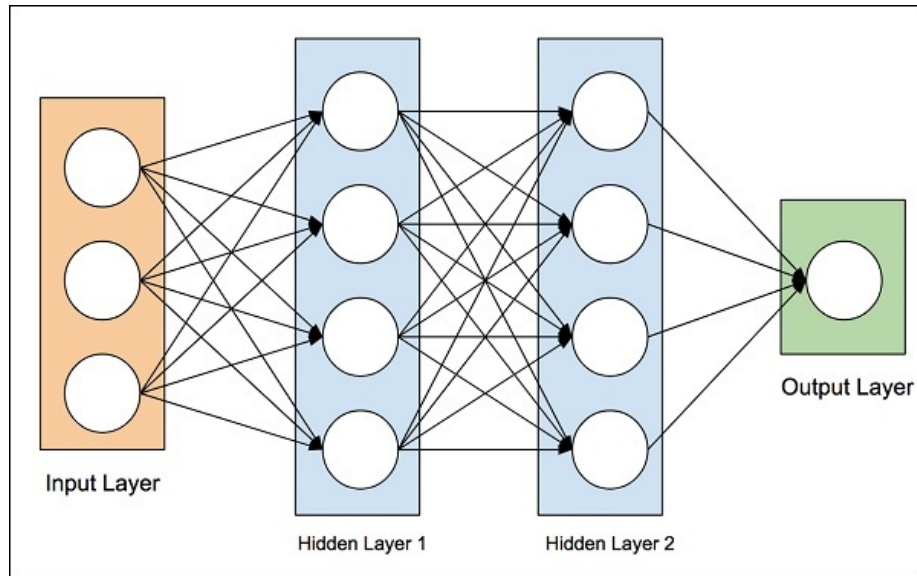


Figura 3.1: Red neuronal convolucional

3.2. Edge Computing

El edge computing [2] es un tipo de informática que ocurre, en la ubicación física del usuario, en la ubicación de la fuente de los datos o cerca de estas. Permitiendo que los usuarios obtengan servicios mas rápidos y fiables.

3.3. Jetson Nano

Una Jetson Nano [3] es un mini PC de bajo coste, el cual cabe en una mano. Se encuentra compuesto por un SoC, procesador ARM de 64 bits de 4 núcleos y una GPU con arquitectura Maxvell con 128 núcleos de procesameinto gráfico, conectividad de red, contando con una potencia total de 472 Gflops, cuenta a su vez, con puertos USB-A, salidas de vídeo HDMI y DisplayPort y un puerto para su conexión a Internet.



Figura 3.2: Jetson Nano

3.4. YOLO

You Only Look Once (YOLO) [4] es un algoritmo de detección que usa Deep Learning y CNN para ello, como su nombre indica sólo necesita mirar la imagen una única vez, de tal forma que la detección es mucho más rápida que en otros algoritmos, pero a cambio de sacrificar rendimiento a la hora de predecir. Para llevar a cabo la detección, divide la imagen en una cuadrícula de $S \times S$ (imagen de la izquierda). Por cada cuadrícula, predice N posibles "bounding boxes" calcula la probabilidad de cada una de ellas, es decir, en total se predicen $S \times S \times N$ cajas diferentes (la gran mayoría con una probabilidad muy baja) (imagen del centro). Por último, se eliminan las cajas que están por debajo de un límite, conocido este como non-max-suppression, de tal forma, que se eliminan los objetos detectados por duplicado, dejando los que poseen un mayor valor de predicción (imagen de la derecha).

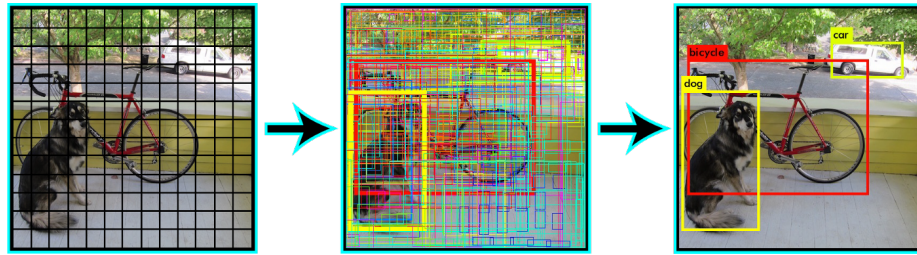


Figura 3.3: Explicación YOLO

3.5. Object Detection

El Object Detection [5] es una técnica de visión por ordenador que permite localizar imágenes y/o vídeos. Estos algoritmos se aprovechan del aprendizaje automático o del profundo con el objetivo de obtener resultados significativos, es decir, intentan replicar la inteligencia humana a la hora de reconocer un objeto.

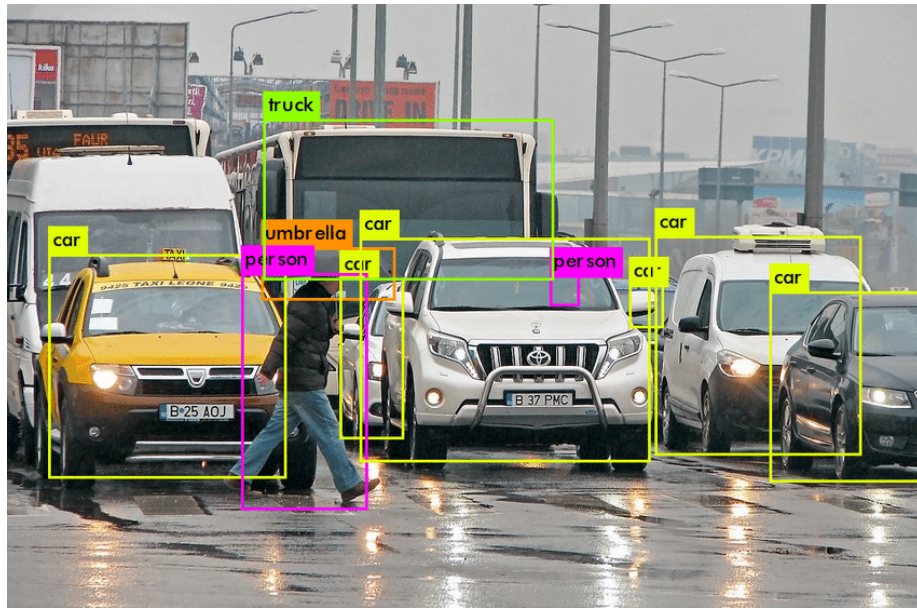


Figura 3.4: Detección de objetos

Técnicas y herramientas

Esta parte de la memoria tiene como objetivo presentar las técnicas metodológicas y las herramientas de desarrollo que se han utilizado para llevar a cabo el proyecto. Si se han estudiado diferentes alternativas de metodologías, herramientas, bibliotecas se puede hacer un resumen de los aspectos más destacados de cada alternativa, incluyendo comparativas entre las distintas opciones y una justificación de las elecciones realizadas. No se pretende que este apartado se convierta en un capítulo de un libro dedicado a cada una de las alternativas, sino comentar los aspectos más destacados de cada opción, con un repaso somero a los fundamentos esenciales y referencias bibliográficas para que el lector pueda ampliar su conocimiento sobre el tema.

Aspectos relevantes del desarrollo del proyecto

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros³, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.

Trabajos relacionados

Este apartado sería parecido a un estado del arte de una tesis o tesina. En un trabajo final grado no parece obligada su presencia, aunque se puede dejar a juicio del tutor el incluir un pequeño resumen comentado de los trabajos y proyectos ya realizados en el campo del proyecto en curso.

Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

Bibliografía

- [1] IBM, “Qué es deep learning.” [Online]. Available: <https://www.ibm.com/es-es/cloud/watson-studio/deep-learning>
- [2] R. Hat, “Qué es edge computing,” 2021. [Online]. Available: <https://www.redhat.com/es/topics/edge-computing/what-is-edge-computing>
- [3] M. Computer, “Nvidia jetson nano, una raspberry pi para ia,” 2019. [Online]. Available: <https://www.muycomputer.com/2019/03/19/nvidia-jetson-nano/>
- [4] enrique a., “Detección de objetos con yolo: implementaciones y como usarlas.” [Online]. Available: <https://medium.com/@enriqueav/detecci%C3%B3n-de-objetos-con-yolo-implementaciones-y-como-usarlas-c73ca2489246>
- [5] Mathworks, “What is object detection?” [Online]. Available: <https://es.mathworks.com/discovery/object-detection.html>