

Proyecto Final

Sistemas Inteligentes

Integrantes:

Natalia González	A01382007
Óscar Peña García	A01176808
Ernesto Guzmán	A00824855
Alexa Mercado Sánchez	A01634000

Presentación

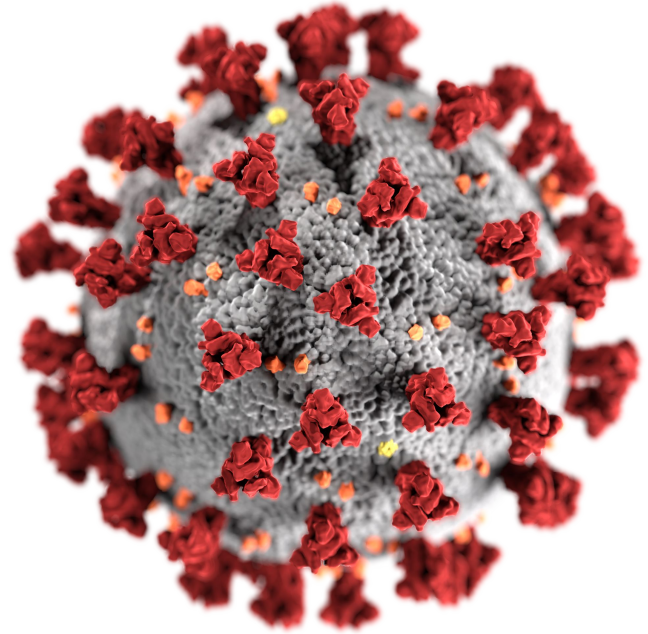
Cada equipo debe grabar una presentación de 20 minutos exponiendo el proyecto que desarrollaron. En la presentación indiquen:

- La problemática que estudiaron
- Los datos que recolectaron y el procesamiento que les efectuaron
- El componente de innovación que agregaron al proyecto
- La demostración de su modelo corriendo

El video lo deben subir a youtube en alta calidad.

Problemática

- El COVID-19 ha tenido un impacto significativo en el mundo, haciéndonos partícipes de una nueva realidad. Como una de las medidas reconocidas para la prevención de los contagios, se recomienda el uso de cubrebocas que evite la propagación del virus.



Objetivo

- Este proyecto tiene como objetivo el clasificar correctamente si una persona porta un cubrebocas ya sea a partir de fotos o vídeos, mediante visión artificial.

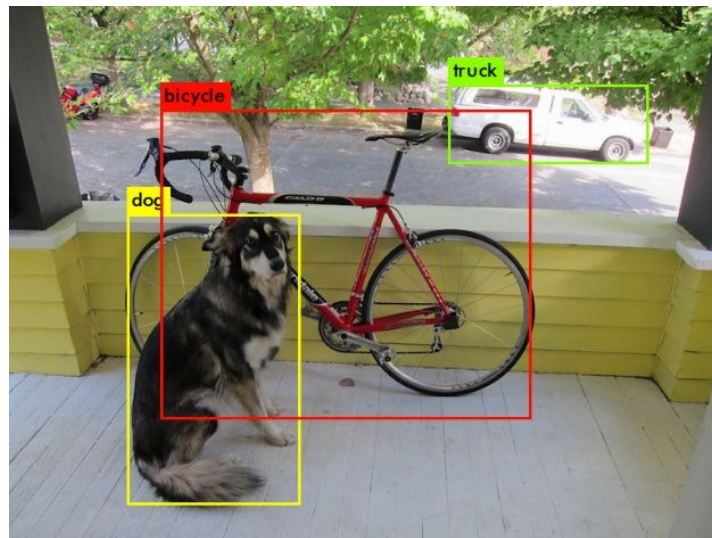


Visión artificial

- La visión artificial es una disciplina que persigue que los ordenadores alcancen la capacidad de percibir y comprender una o varias imágenes y actuar de una manera determinada.
- El objetivo último de la visión artificial es conseguir el desarrollo de estrategias automáticas para el reconocimiento de patrones complejos en imágenes de múltiples dominios.

Detección de objetos

- Detección de objetos es la parte de la visión artificial que estudia cómo detectar la presencia de objetos en una imagen sobre la base de su apariencia visual
- A la obtención de modelos matemáticos compactos que "resuman" el contenido de la imagen con el fin de simplificar el proceso de aprendizaje de los objetos a reconocer se le llama *extracción de descriptores*.



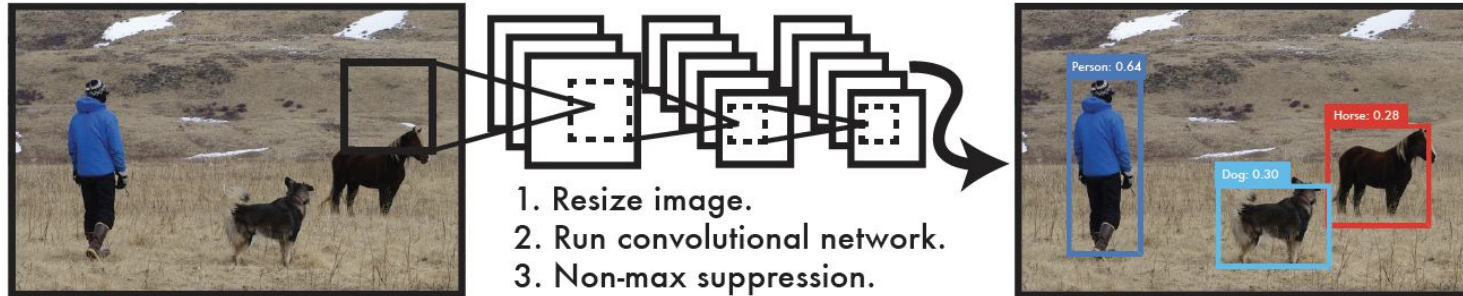
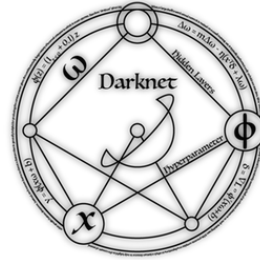
Algunos ejemplos de algoritmos para detección de objetos:

- **YOLO**, SSD, RetinaNet

¿Por qué YOLO? ^[1,6]

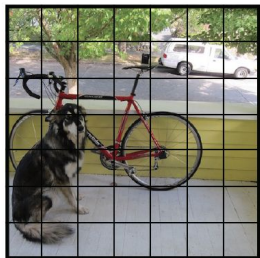
YOLO (***You Only Look Once***) es uno de los algoritmos de detección por visión computacional más rápidos y exactos que hay. Su arquitectura está construida en el framework de Darknet. [1]

YOLO



Funcionamiento de YOLO.

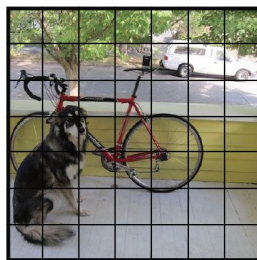
1. La imagen se divide en $S \times S$ celdas.



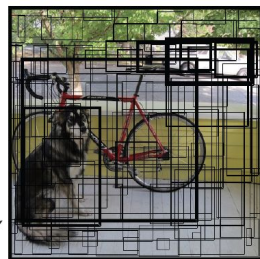
5×5 grid on input

Funcionamiento de YOLO.

2. Cada celda predice B cajas delimitantes con respectivos puntajes de confianza.



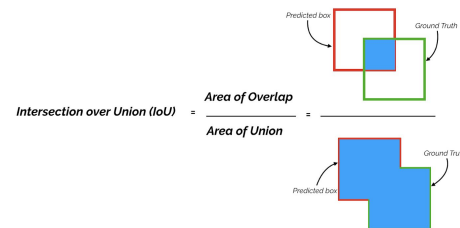
5 x 5 grid on input



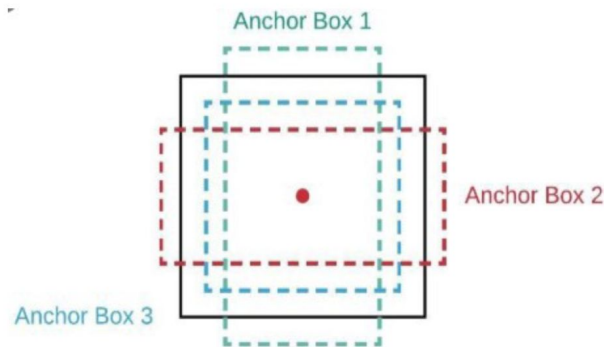
Bounding boxes + confidence

Cada caja consta de 5 parámetros::

- **(b_x, b_y)** :
Coordenadas del centro de la caja.
- **(b_w, b_h)** :
Dimensiones de la caja.
- **Confidence:** \hat{b}
IoU entre b y \hat{b}

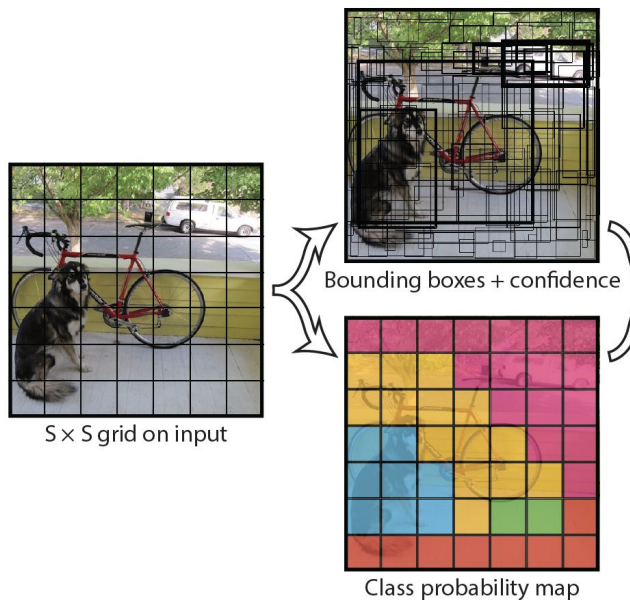


Para generarlas se utilizan *anchor boxes*. *Tamaños predefinidos de cajas.* [2]



Funcionamiento de YOLO.

3. Cada celda predice la probabilidad condicional de una clase dada un objeto.



Probabilidad de que el objeto pertenezca a una clase.

Probabilidad de que la caja contenga un objeto.

La caja generada intersecta a la caja que contiene al objeto.

$$\Pr(\text{Class}_i | \text{Object}) * \Pr(\text{Object}) * \text{IOU}_{\text{pred}}^{\text{truth}}$$
$$= \Pr(\text{Class}_i) * \text{IOU}_{\text{pred}}^{\text{truth}} \quad (1)$$

Funcionamiento de YOLO.

4. Las predicciones se codifican en un tensor de las siguientes dimensiones:

En YOLOv3 se atribuye la probabilidad de clases a las cajas b y no a las celdas. [5]

$$S \times S \times (B * (5 + C))$$

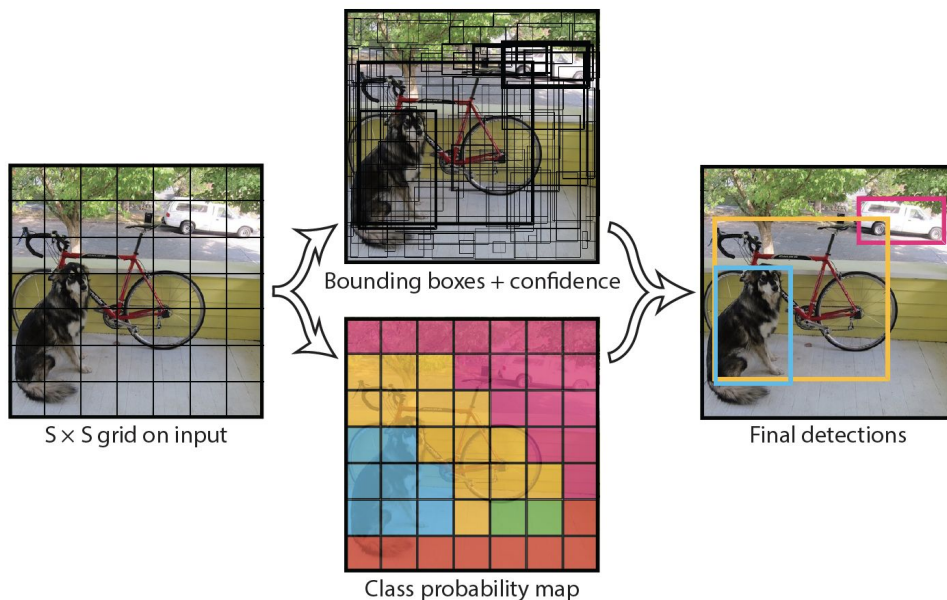
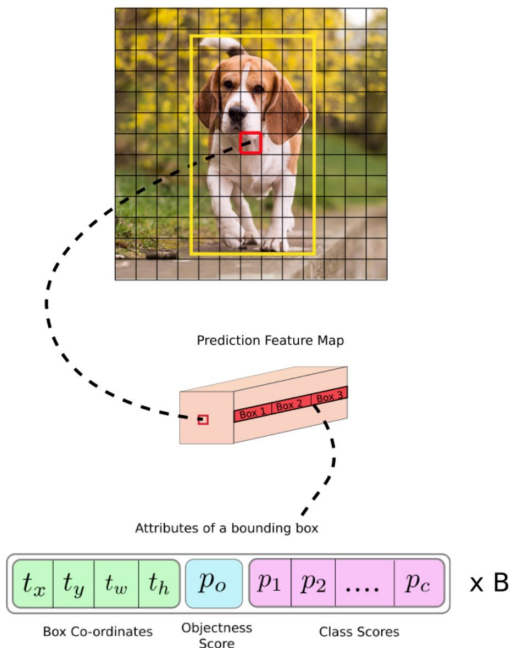
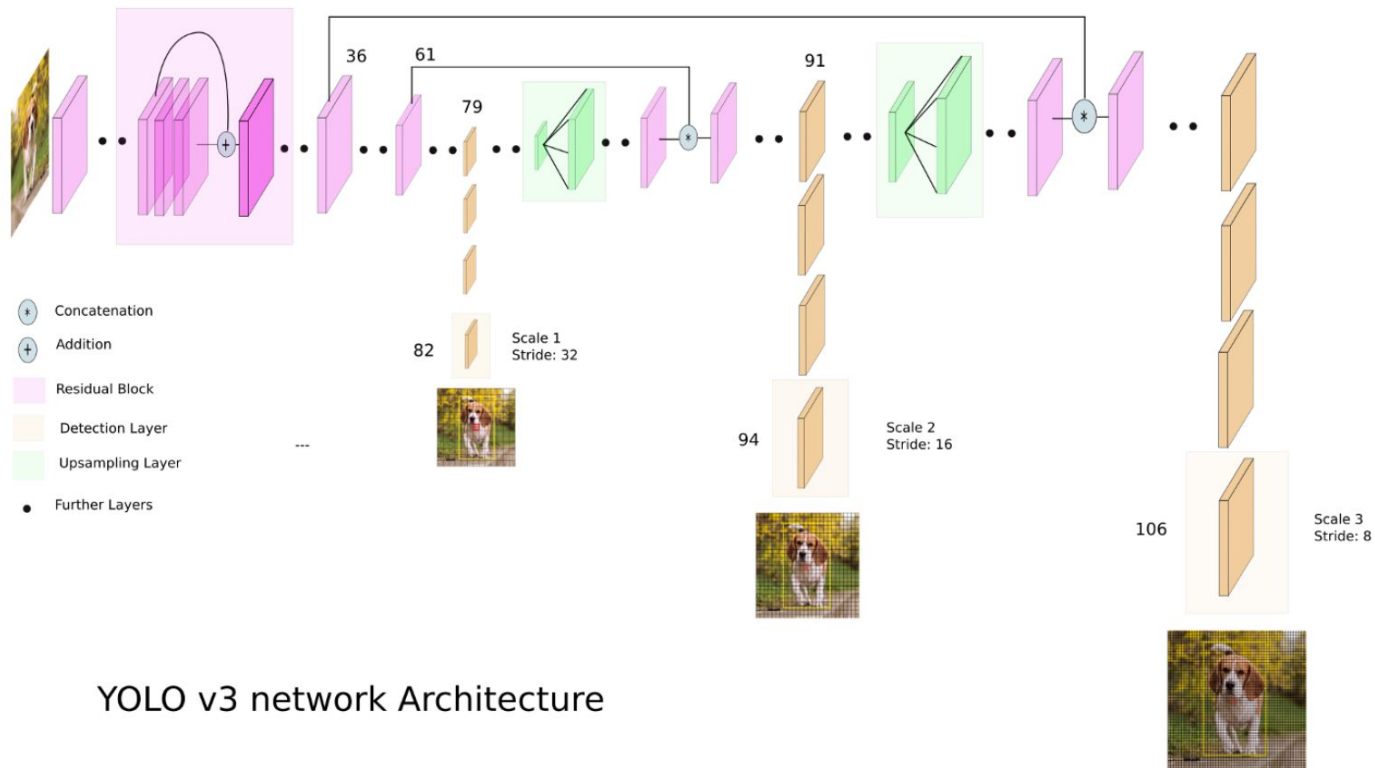


Image Grid. The Red Grid is responsible for detecting the dog



Red Convolutional (YOLOv3). [6]



YOLO v3 network Architecture

Función de Costo. [1]

Otra particularidad de YOLO es que trata la detección como un problema de regresión. Se definen las ponderaciones: $\lambda_{\text{coord}} > \lambda_{\text{noobj}}$. Donde λ_{noobj} es una caja sin detectar objetos, por lo tanto se le da un peso menor.

Regression
loss

$$\lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\ + \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right]$$

Confidence
loss

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{obj}} (C_i - \hat{C}_i)^2 \\ + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^B \mathbb{1}_{ij}^{\text{noobj}} (C_i - \hat{C}_i)^2$$

Classification
loss

$$+ \sum_{i=0}^{S^2} \mathbb{1}_i^{\text{obj}} \sum_{c \in \text{classes}} (p_i(c) - \hat{p}_i(c))^2$$

Sobre las versiones de Yol:



Elegimos YOLOv4 dado que está optimizado para entrenarse mejor en GPUs. Esta versión ya no es desarrollada por Joseph Redmon.

Ver video para información sobre:

- Los datos que recolectamos y el procesamiento que efectuamos.
- El componente de innovación que agregaron al proyecto
 - Tiempo real con webcam
 - Labeling de imágenes nuestras nuevas, para hacer el proceso completo de machine learning.
- Demostración del modelo corriendo.

Referencias

- [1] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).
- [2] Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
- [3] Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.
- [4] Yohanandan, S. (2020, June 9). Map (mean average precision) might confuse you! Medium. Retrieved June 3, 2022, from <https://towardsdatascience.com/map-mean-average-precision-might-confuse-you-5956f1bfa9e2>
- [5] Jonnalagadda, V. K. (2019, January 31). Object detection yolo V1 , V2, V3. Medium. Retrieved June 3, 2022, from <https://medium.com/@venkatakrishna.jonnalagadda/object-detection-yolo-v1-v2-v3-c3d5eca2312a>
- [6] Kathuria, A. (2018, April 29). *What's new in yolo v3?* Medium. Retrieved June 3, 2022, from <https://towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b>