



YOLOV3: UNA SOLUCIÓN AVANZADA PARA EL CONTEO DE OBJETOS EN TIEMPO REAL.

YOLOV3: AN ADVANCED SOLUTION FOR REAL-TIME OBJECT COUNTING.

José Antonio Fuentes Velásquez.
Licenciado en matemáticas.
antonio.velasquez@ues.edu.sv
[https: 0000-0001-7013-108X](https://0000-0001-7013-108X)

Eduardo José Vásquez Flores.
Ingeniero en Sistemas Informáticos.
eduardo_vasquez@univo.edu.sv
[https: 0000-0003-0864-7274](https://0000-0003-0864-7274)

Resumen

Con la evolución de la modernidad y los nuevos desafíos que esta trae consigo se presenta una solución avanzada utilizando el algoritmo de YOLOv3 para el conteo de objetos en tiempo real. El cual consiste en la detección y el conteo precisos de objetos en imágenes y videos esto deriva a diversas aplicaciones, como seguridad, logística, agricultura y control de calidad. El algoritmo de YOLOv3, un sistema de reconocimiento de objetos basado en aprendizaje automático, el cual ha demostrado ser una herramienta eficiente en el proceso de detección de múltiples clases de objetos con una alta velocidad de procesamiento en una sola imagen o cuadros en videos. En este estudio, se utiliza el algoritmo de YOLOv3 para abordar el problema del conteo de objetos, aprovechando su capacidad para detectar y localizar objetos en una imagen y proporcionar un recuento preciso en tiempo real. YOLOv3, el proceso de entrenamiento del

modelo y la evaluación de su rendimiento en distintos conjuntos de datos. Los resultados experimentales muestran que la solución propuesta ofrece una precisión prometedora a la detección y el conteo de objetos, superando las limitaciones de los enfoques tradicionales. Este estudio contribuye al avance de la visión computacional y proporciona una herramienta efectiva para aplicaciones que requieren conteo de objetos en tiempo real.

Palabras Clave:

YOLOv3, Aprendizaje automático, redes neuronales, Conteo de objetos, Reconocimiento de objetos, Visión computacional.

Abstract

With the evolution of modernity and the new challenges it brings, an advanced solution is presented using the YOLOv3 algorithm for real-time object counting. This consists of the precise detection and counting of objects in





images and videos, which leads to diverse applications, such as security, logistics, agriculture, and quality control. The YOLOv3 algorithm, an object recognition system based on machine learning, has proven to be an efficient tool in the process of detecting multiple classes of objects with a high processing speed in a single image or frames in videos. In this study, the YOLOv3 algorithm is used to address the problem of object counting, taking advantage of its ability to detect and locate objects in an image and provide an accurate real-time count. Describing the YOLOv3 configuration process, the model training process, and the evaluation of its performance on different datasets. The experimental results show that the proposed solution offers promising accuracy to object detection and counting, overcoming the limitations of traditional approaches. This study contributes to the advancement of computer vision and provides an effective tool for applications that require real-time object counting.

Keywords:

YOLOv3, Machine learning, neural networks, Object counting, Object recognition, Computer vision.

Introducción

El reconocimiento y conteo preciso de objetos en imágenes y videos son desafíos fundamentales en el campo del aprendizaje automático y la visión computacional. Entre

los sistemas más destacados, YOLO (You Only Look Once) ha ganado reconocimiento debido a su capacidad para detectar objetos utilizando deep learning y redes neuronales convolucionales (CNN). A diferencia de otros enfoques, YOLO se distingue por su velocidad, ya que puede “ver” la imagen una sola vez, lo que le permite ser el más rápido en su clase, aunque con un pequeño sacrificio en termino de precisión.

En este artículo, nos centraremos en las diferentes implementaciones y usos de YOLO, sin embargo, no se profundizará en detalles técnicos. Exploraremos como aprovechar la rapidez de YOLO para detectar objetos en tiempo real en videos, alcanzando hasta 30 cuadros por segundo (FPS). Si bien se sacrifica un poco de exactitud en comparación con enfoques más lentos, esta capacidad de detección en tiempo real abre una amplia gama de aplicaciones en diversos campos.

A lo largo del artículo, examinaremos las implementaciones prácticas de YOLO en diferentes áreas como la seguridad, la logística, agricultura de precisión y la investigación científica. Analizaremos cómo utilizar YOLO de manera efectiva y cómo optimizar su rendimiento para cumplir con los requisitos específicos de cada aplicación.

A lo largo de este artículo de expondrán las capacidades y limitaciones que trae consigo el uso de YOLO, esperamos proporcionar a los lectores una visión general de esta tecnología





y su potencial para la automatización, la toma de decisiones inteligente y la mejora de la eficiencia en diversos escenarios. Sin duda, YOLO representa un avance importante en el campo del reconocimiento de objetos en tiempo real y su aplicación práctica tiene el potencial de transformar una amplia gama de industrias y optimizar procesos.

1 Aprendizaje Automático y YOLOv3

El aprendizaje automático con YOLOv3 ha revolucionado el reconocimiento de objetos en imágenes y videos. YOLOv3, basado en el aprendizaje profundo y las redes neuronales convolucionales, los cuales agilizan el proceso de interpretación y aprendizaje, de este modo YOLOv3 se destaca por su capacidad de “ver” y analizar la imagen de una sola vez logrando así una detección de objetos extremadamente rápida. Aunque podría haber una ligera disminución en la precisión en comparación con otros enfoques más lentos, la velocidad de YOLOv3 lo convierte en una opción ideal para aplicaciones de análisis y detección en tiempo real.

El proceso de aprendizaje automático con YOLOv3 implica utilizar conjuntos de datos anotados para entrenar la red neuronal convolucional. Durante el entrenamiento, la red aprende a reconocer y localizar objetos, así como a asignar probabilidades de clasificación. Una vez entrenado, el modelo de la red neuronal convolucional de YOLOv3 se puede utilizar para detectar objetos en nuevas

imágenes o videos en tiempo real, aprovechando su capacidad para procesar rápidamente las características y patrones de toda la imagen.

YOLOv3 al ser una herramienta versátil su implementación se encuentra presente en una amplia gama de casos de uso con enfoques en diferentes disciplinas, desde seguridad y logística hasta control de calidad. Su combinación de velocidad y precisión equilibradas lo ha convertido en una herramienta valiosa en diversas industrias, ofreciendo soluciones avanzadas y eficiente para el reconocimiento y conteo de objetos, proporcionando una detección en tiempo real y un conteo preciso.

2 Antecedentes Históricos.

El marco YOLO v3 fue propuesto por (Redmon y otros, 2016) y ha evolucionado para incorporar varias características avanzadas, allanando el camino para algoritmos de detección de objetos más precisos y de última generación. El algoritmo YOLO v3 implementa Darknet-53 como una columna vertebral para extraer características de las imágenes de entrada. Darknet-53 es una red neuronal profunda escrita en C y Compute Unified Device Architecture (CUDA), compuesta por capas convolucionales utilizadas para la extracción de características. Implementa una Red de Pirámide de Características (FPN) [7], lo que permite la extracción de mapas de características de las





imágenes de entrada. Admite cálculos de detección en CPU y GPU, y está fácilmente accesible en Github.

3.1 Descripción de YOLO

YOLO es un enfoque de detección de objetos que utiliza una única red convolucional para predecir simultáneamente múltiples cuadros delimitadores y las probabilidades de clase asociadas a esos cuadros. En lugar de trabajar con regiones de interés propuestas, YOLO utiliza características de toda la imagen para realizar predicciones precisas de cada cuadro delimitador. Además, es capaz de predecir todos los cuadros delimitadores de todas las clases en una imagen al mismo tiempo.

Para lograr esto, YOLO divide la imagen de entrada en una cuadrícula de Tamaño $n \times n$. Cada celda de la cuadrícula es responsable de detectar los objetos cuyos centros están ubicados dentro de esa celda. Este enfoque permite una detección eficiente y precisa de objetos en diferentes partes de la imagen.

Una de las ventajas y características de YOLO es que se entrena utilizando imágenes completas y optimiza directamente el rendimiento de detección. Además, posee la capacidad de procesamiento de video en tiempo real con una baja latencia de menos de 25 milisegundos.

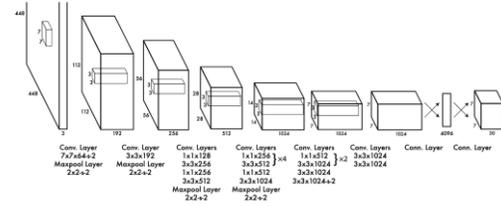


Figura 1 Arquitectura del sistema YOLO. Figura tomada de (Redmon y otros, 2016).

N posibles “bounding boxes” y calcula el nivel de certidumbre (o probabilidad) de cada una de ellas (imagen del centro), es decir, se calculan $S \times S \times N$ diferentes cajas, la gran mayoría de ellas con un nivel de certidumbre muy bajo. Después de obtener estas predicciones se procede a eliminar las cajas que estén por debajo de un límite. A las cajas restantes se les aplica un paso de “non-max suppression” que sirve para eliminar posibles objetos que fueron detectados por duplicado y así dejar únicamente el más exacto de ellos.

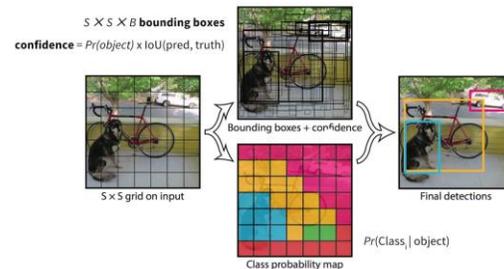


Figura 2 Cuadros delimitadores.

Cada celda es responsable de predecir las probabilidades de clase. Esto no quiere decir que alguna celda de la cuadrícula contenga algún objeto, es una probabilidad en que la celda puede o no contener un objeto. Entonces, si una celda de la grilla predice un automóvil, no está diciendo que haya un automóvil, solo





está diciendo que, si hay un objeto, identificando que ese objeto es un automóvil.

Describamos con más detalles cómo se ve la salida.

La idea principal de las cajas de anclaje es predefinir dos formas diferentes. Se llaman cajas de anclaje o formas de caja de anclaje. De esta manera, podremos asociar dos predicciones con los dos cuadros de anclaje. En general, podríamos usar incluso más cuadros de anclaje (cinco o incluso más). Las anclas se calcularon en el conjunto de datos COCO usando el agrupamiento de k-means.

Tenemos una cuadrícula y cada celda va a predecir:

- En el sistema YOLO, se utiliza una cuadrícula para realizar
- predicciones de cuadro delimitador.
- Cada celda de la cuadrícula predice las siguientes características:
- 4 coordenadas: t_x , t_y , t_w , t_h , que representan las coordenadas y el tamaño del cuadro delimitador.
- 1 error de objetividad, que es la puntuación de confianza de si hay un objeto presente en la celda.
- Un conjunto de probabilidades de clase, que indican la probabilidad de que el objeto pertenezca a cada clase posible.

La siguiente fórmula describe el proceso de cómo se transforma la salida de la red para obtener predicciones de cuadro delimitador:

$$b_x = \sigma(t_x) + c_x$$

$$b_y = \sigma(t_y) + c_y$$

$$b_w = p_w e^{t_w}$$

$$b_h = p_h e^{t_h}$$

Donde b representa el cuadro delimitador final predicho, c_x y c_y son las coordenadas de la celda en la cuadrícula, p_w y p_h son los tamaños de la celda, y σ es la función sigmoide para mapear los valores a un rango entre 0 y 1.

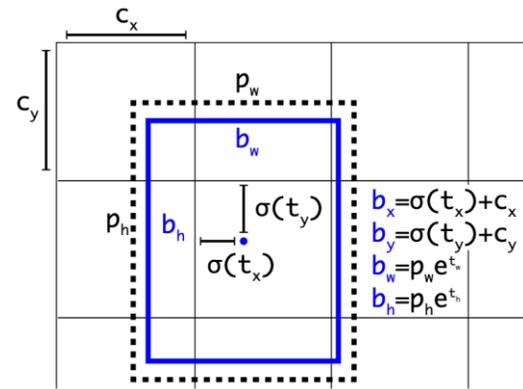


Figura 3 Cuadros delimitadores con dimensiones previas y predicción de ubicación.

De donde p_c , b_x , b_y , b_h , b_w son las coordenadas centrales x , y , el ancho y la altura de nuestra predicción. t_x , t_y , t_w , t_h es lo que produce la red. c_x y c_y son las coordenadas superiores izquierda de la cuadrícula. p_w y p_h son anclas de dimensiones para la caja.

3.2.2 Intersecciones sobre uniones

La métrica de IoU (Intersección sobre Unión) es una característica esencial del clasificador YOLO v3 implementada por la mayoría de los modelos de detección de objetos de última generación para describir la superposición de





cuadros delimitadores. Proporciona una medida para evaluar la similitud entre el cuadro delimitador predicho y el cuadro delimitador de referencia. La idea es comparar la proporción del área de superposición con la región combinada total de los dos cuadros, como se muestra en la ecuación:

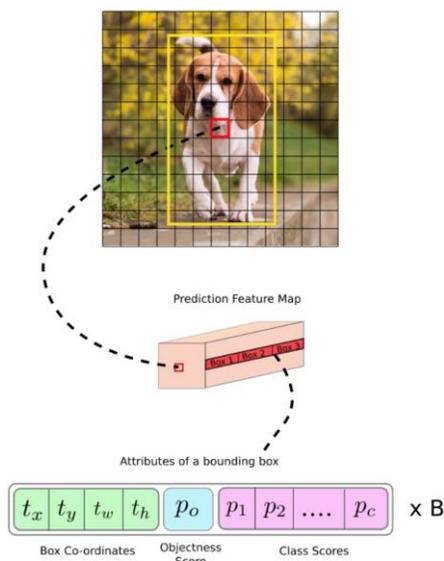


Figura 4 Proceso de detección

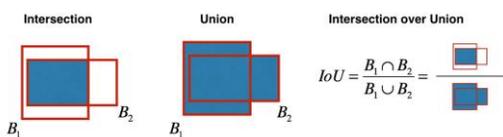


Figura 5 IOU proceso.

$$IoU = \frac{\text{Área de supervisión}}{\text{Región combinada.}}$$

En el clasificador YOLO v3, en el proceso de detección de objetos se utilizan cuadros delimitadores y el principio de Intersección sobre Unión (IoU). Durante la detección de

objetos, un puntaje de 1 indica que el cuadro delimitador predicho coincide precisamente con el cuadro de referencia (a encontrado una coincidencia con el objeto). Un puntaje relativo de 0 implica que los cuadros predichos y de referencia no se superponen (existe coincidencia con el cuadro de referencia).

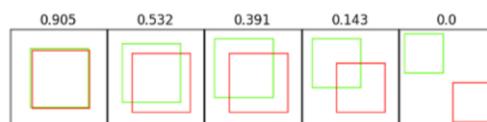


Figura 6 IoU

3.2.3 Supresión no máxima

Ahora, para implementar la supresión no máxima, los pasos clave son:

- Seleccione la casilla que tenga la puntuación más alta.
- Calcule su superposición con todos los demás cuadros y elimine los cuadros que se superponen más de iou_threshold.
- Regrese al paso 1 e itere hasta que no haya más cuadros con una puntuación más baja que el cuadro seleccionado actualmente.

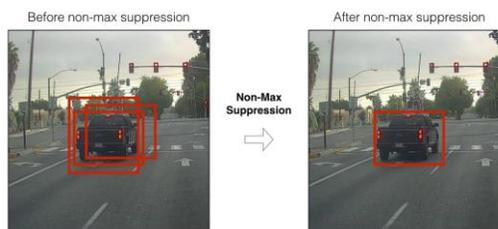


Figura 7 Supresión no máxima.





3.2.4 Predicciones a través de escalas

YOLOv3 predice cuadros delimitadores en 3 escalas diferentes. Nuestro sistema extrae características de esas escalas utilizando un concepto similar a las redes de pirámide de características [8]. A partir de nuestro extractor de características base, agregamos varias capas convolucionales. La última de estas capas predice un tensor tridimensional que codifica los cuadros delimitadores, la probabilidad de objeto y las predicciones de clase. En nuestros experimentos con COCO, predecimos 3 cuadros en cada escala, por lo que el tensor tiene una forma de $N \times N \times [3 * (4 + 1 + 80)]$ para los 4 desplazamientos de cuadro delimitador, 1 predicción de probabilidad de objeto y 80 predicciones de clase.

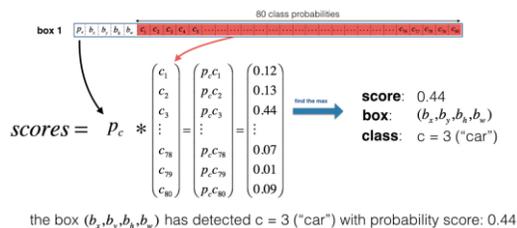


Figura 8 Caption

A continuación, tomamos el mapa de características de las 2 capas anteriores y lo aumentamos en un factor de $2 \times$. también tomamos un mapa de características anterior en la red y lo fusionamos con nuestras características aumentadas utilizando concatenación. Este método nos permite obtener información semántica más significativa de las características aumentadas

y una información más detallada del mapa de características anterior. Luego, agregamos algunas capas convolucionales adicionales para procesar este mapa de características combinado y, finalmente, predecir un tensor similar, pero ahora con el doble tamaño.

Realizamos el mismo diseño una vez más para predecir cuadros delimitadores en la escala final. Por lo tanto, nuestras predicciones para la tercera escala se benefician de todos los cálculos anteriores, así como de características detalladas desde el inicio de la red.

Aun utilizamos el agrupamiento k-means para determinar nuestras dimensiones previas de cuadros delimitadores. Simplemente elegimos arbitrariamente 9 grupos y 3 escalas, y luego distribuimos los grupos de manera equitativa entre las escalas. En el conjunto de datos COCO, los 9 grupos fueron: (10×13) , (16×30) , (33×23) , (30×61) , (62×45) , (59×119) , (116×90) , (156×198) , (373×326) .

3.3.5 Extractor de Características

Utilizamos una nueva red para realizar la extracción de características. Nuestra nueva red es un enfoque híbrido entre la red utilizada en YOLOv2, Darknet-19, y ese novedoso concepto de redes residuales.





	Type	Filters	Size	Output
1x	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
2x	Convolutional	128	3 × 3 / 2	64 × 64
	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
8x	Convolutional	256	3 × 3 / 2	32 × 32
	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
8x	Convolutional	512	3 × 3 / 2	16 × 16
	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
4x	Convolutional	1024	3 × 3 / 2	8 × 8
	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figura 9 Darknet-53

4. Resultados

En este estudio, se evaluó la solución propuesta utilizando el algoritmo YOLOv3 para el conteo de objetos en tiempo real. Se realizaron experimentos utilizando diversos conjuntos de datos para evaluar y determinar la precisión y el rendimiento del modelo entrenado.

Los resultados obtenidos fueron prometedores en términos de precisión en el conteo de objetos. Se observó una alta concordancia entre el conteo realizado por el modelo YOLOv3 y el conteo manual realizado por expertos humanos lo cual brinda una mayor fiabilidad de los resultados. Esto indica que el modelo tiene la capacidad de realizar conteos precisos y en tiempo real en diferentes escenarios y con diferentes clases de objetos.



Figura 10 Primera imagen

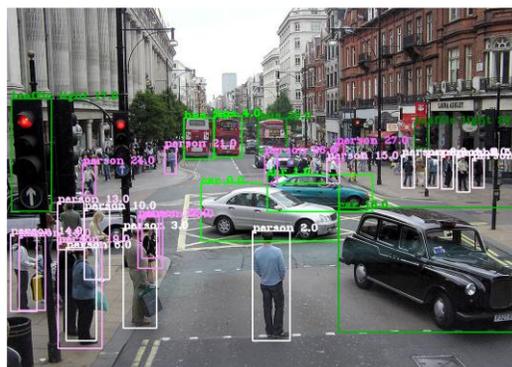


Figura 11 Segunda imagen

5. Conclusiones

En este reporte, se ha presentado una solución avanzada utilizando el algoritmo YOLOv3 para el conteo de objetos en tiempo real. La detección y el conteo precisos de objetos en imágenes y videos son desafíos fundamentales en diversas aplicaciones, y YOLOv3 ha demostrado ser una herramienta eficaz en este sentido. Aprovechando su capacidad para detectar y localizar objetos en una imagen y proporcionar un conteo preciso en tiempo real, YOLOv3 ha mostrado resultados prometedores en términos de precisión.

La implementación de YOLOv3 se basa en el aprendizaje automático y las redes neuronales convolucionales (CNN), lo que permite una





detección rápida y eficiente de múltiples clases de objetos. Aunque se pueda sacrificar un poco de precisión en comparación con enfoques más lentos, la velocidad de YOLOv3 lo convierte en una solución ideal para aplicaciones que requieren detección en tiempo real.

A través de este estudio, se ha demostrado que YOLOv3 es una herramienta versátil y aplicable en diversas áreas, como seguridad, logística, agricultura y control de calidad. Su facilidad de implementación, junto con su capacidad para realizar detecciones rápidas y precisas, lo convierten en una opción viable para diversas aplicaciones prácticas.

Bibliografía

- Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature Pyramid Networks for Object Detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. <https://doi.org/https://doi.org/10.48550/arXiv.1612.03144>
- Lin, T.-Y., Goyal, P., Girshick, R., Él, K., & Dollár, P. (2017). pérdida focal para detección de objetos densos. *Conferencia Internacional IEEE 2017 sobre Visión por Computadora (ICCV)*, 2999-3007.

<https://doi.org/10.1109/ICCV.2017.324>

- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., . . . Zitnick, C. (2014). Microsoft COCO: Common Objects in Context. *European Conference on Computer Vision (ECCV)*, 740-755. https://doi.org/https://doi.org/10.1007/978-3-319-10602-1_48
- Redmon, J., & Farhadi, A. (2017). YOLO9000: Mejor, más rápido, más fuerte. *Conferencia IEEE 2017 sobre visión por computadora y reconocimiento de patrones (CVPR)*, 6517-6525. <https://doi.org/10.1109/CVPR.2017.690>
- Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *arXiv preprint arXiv:1804.02767*. <https://doi.org/https://doi.org/10.48550/arXiv.1804.02767>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779-788. <https://doi.org/10.1109/CVPR.2016.91>.

