Detección de armas en imágenes usando YOLO

Alejandro G. Reyes-Aldeco, Kelsey A. Ramírez-Gutiérrez, Ignacio Algredo-Badillo
Coordinación de Ciencias en Tecnologías de Seguridad
Instituto Nacional de Astrofísica, Óptica y Electrónica
Luis Enrique Erro 1, Sta María Tonanzintla, 72840 San Andrés Cholula, Puebla, México
agreyesaldeco@gmail.com, kramirez@inaoep.mx, algredobadillo@inaoep.mx

***Resumen*—El número creciente de delitos cometidos en los Estados Unidos Mexicanos tienen como principal herramienta el uso de armas de fuego. Varias soluciones tecnológicas se han implementado en los centros de monitoreo dentro del país, donde las soluciones basadas en visión artificial son una de las más importantes. La detección automática de armas puede garantizar la prevención de delitos. En este artículo se presenta un sistema basado en la red neuronal YOLO V3 con el objetivo de detectar armas en un conjunto de imágenes de asaltos, prácticas de tiro y documentales. Adicionalmente, se genera un conjunto de imágenes, las cuales son etiquetadas para que el sistema sea entrenado, probado y validado. Los resultados establecen una opción para que sea implementado en sistemas de videovigilancia ya que se tiene un 84.46 % de exactitud.**

***Index Terms*—CNN, YOLO, Gun-detection**

I. Introducción

Actualmente, en México, el 44.2 % de actividades ilícitas son cometidas con armas de fuego [1]. El crimen y las actividades ilícitas pueden ser reducidas al monitorear e identificar el comportamiento, vestimenta, gestos, entre otros, que comúnmente tienen los delincuentes. Adicionalmente, la percepción de las personas sobre la inseguridad ha ido en aumento en comparación con los años anteriores. Una posible solución a los problemas anteriormente descritos seria desplegar sistemas de control de vigilancia en vehículos y/o edificios con detección de personas armadas además de una alerta a las autoridades.

Los diversos estados de la república mexicana han implementado Centros de Comando y Control, donde se capta información integral para la toma de decisiones en materia de seguridad pública, urgencias médicas, medio ambiente, protección civil, movilidad y servicios a la comunidad a través del vídeo monitoreo, de la captación de llamadas telefónicas y de aplicaciones informáticas de inteligencia, enfocadas a mejorar la calidad de los habitantes. Entre estos se tienen a los C2Móvil (Centros de Comando y Control Móviles) que son vehículos con cámaras desplegables que permiten el monitoreo en lugares de difícil acceso y el envío de imágenes en todo momento a un centro de comando central, y los C4 (Centros de Comando, control, comunicación y computo), C5 (Centros de Comando, Control, Cómputo, Comunicaciones y Contacto Ciudadano) y C5i (Centro de Comando, Control, Cómputo, Comunicaciones, Coordinación e inteligencia) donde se realiza el video monitoreo con la finalidad de prevenir y alertar inmediatamente a las autoridades de seguridad y de emergencias sobre cualquier situación de riesgo.

Los Centros de Comando y control proporcionan imágenes en tiempo real, sin embargo, tan sólo en la ciudad de México han sido instaladas más de 15 mil cámaras de vigilancia, por lo cual resulta complicado anticiparse a los delitos antes de que sucedan debido a la exigencia visual y se necesita una gran número de operadores para observar la totalidad de cámaras.

Con estas fuentes de datos y con ayuda de tecnología es importante tener operaciones basadas en visión artificial (VA) para proporcionar seguridad. En este caso, la VA es un campo de la inteligencia artificial (IA) donde un conjunto de algoritmos son destinados para el procesamiento de imágenes. *Machine Learning* o aprendizaje automático, como parte de la IA, usa algoritmos para extraer información de datos sin procesar, reconocer patrones y representarla en algún tipo de modelo. El *Deep learning* o aprendizaje profundo, forma parte del *machine learning*, cuya meta es llegar a un aprendizaje profundo más avanzado. Entre los algoritmos más populares en el uso del *Deep learning* se encuentran las redes neuronales convolucionales (CNN o ConvNet), que son redes neuronales que son capaces de construir funciones completas a partir de otras menos complejas como puede ser el de reconocimiento de patrones.

Entre las diversas aplicaciones de las redes neuronales, se ha encontrado que ayudan en la detección de patrones, sin embargo, debido a la gran cantidad de algoritmos e implementaciones de redes neuronales convolucionales, no existe una metodología que explique cuál es la mejor red o cuál es el número idóneo de capas que debe contener para realizar una efectiva detección de objetos. Para realizar este tipo de tareas, se necesita un conjunto de datos muy grande, además de realizar numerosas operaciones por lo que el uso de recursos computacionales es muy alto ya que requiere de abundante tiempo para procesarlos y obtener resultados.

La investigación de detección de armas se ha centrado principalmente en la detección de armas y cuchillos, donde se pueden usar sensores costosos y especializados. En la detección de armas, se destacan los sistemas utilizados en el equipaje, basado un escáner de rayos X, como en [2], presenta un método basado en una segmentación robusta [3] y vectores de característicos basados en bordes para la detección automática de potenciales armas en el escaneo de equipaje.

De manera similar, los autores en [4] detectan armas basadas en características de forma en imágenes de rayos X de alta energía, este método tiene una exactitud del 98 %, dando una tasa de alarma más baja y reduciendo el tiempo de inspección

ISSN: 3061-8991

de equipaje.

Los autores en [5] presentan un algoritmo de detección de armas basado en la fusión de imágenes. Las imágenes se obtienen utilizando diferentes sensores y se descomponen en bandas de baja y alta frecuencia con la transformada compleja de doble árbol de doble densidad de Wavelet (DDDTCWT).

Verma [6] implementó un método visual de detección de armas en imágenes usando SIFT (Scale Invariant Feature Transform), el detector de puntos de interés de Harris y Fast Retina Keypoint (FREAK). Alcanzado una precisión de 84.26 %.

Por otro lado, un método híbrido que utiliza segmentación basada en color y un detector de puntos de interés SURF (Speed Up Robust Features), con 88.67 % de precisión alcanzada fue propuesto en [7].

Los trabajos mencionados anteriormente se centran únicamente en la búsqueda de armas por separado (no tienen interacción con personas), sin embargo, [8] ha realizado una implementación de detección de armas utilizando MatConvNet [9] y alcanza una precisión del 93 %. En este sentido, el trabajo propuesto en este artículo se centra en hallar armas aumque otros objetos como personas se encuentren en la escena.

Este artículo se centra en la detección de armas en vídeos en tiempo real por lo que se entrenó una red que analiza vídeo a una velocidad mínima de 15 frames por segundo. El artículo se divide de la siguiente manera: Sección II presenta los antecedentes del trabajo, Sección III detalla el sistema propuesto, Sección IV reporta los resultados y comparaciones y Sección V enuncia las conclusiones y el trabajo a futuro.

II. Fundamentos teóricos

A continuación se presentan los elementos teóricos principales utilizados en el sistema propuesto.

II-A. Redes Neuronales para la detección de objetos

Una tendencia en la detección de objetos en imágenes son las redes neuronales profundas. Además, varios modelos basados en redes neuronales son compartidos públicamente tales como tales GoogLeNet [10], ResNet [11] y YOLO [12] y que se pueden utilizar como inicialización para tareas de entrenamiento, detección y clasificación de objetos.

Las redes neuronales convolucionales han permitido una mejora significativa en el rendimiento de los detectores de objetos, como las Redes neuronales convolucionales basadas en regiones (R-CNNs) [13], que solucionan el problema de la localización con un paradigma basado en regiones.

Un inconveniente de las R-CNN [13] es que toman mucho tiempo de entrenamiento ya que tiene que clasificar demasiadas regiones propuestas por imagen, por lo que está actividad es compleja para ser implementada en tareas de tiempo real y el algoritmo de búsqueda selectiva es un algoritmo fijo. Para mejorar estos problemas, el autor de R-CNN [13] propuso un nuevo algoritmo llamado Fast R-CNN [14], que en vez de alimentar a la CNN con regiones propuestas, se alimenta la CNN con la imagen de entrada para generar un mapa de características convolucionales.

Estos algoritmos usan búsqueda selectiva para encontrar las regiones propuestas. Esta búsqueda es lenta y consume tiempo de procesamiento afectando el desempeño de la red, por lo tanto, Faster R-CNN [15] viene con un algoritmo de detección de objetos que elimina el algoritmo de búsqueda selectiva y deja a la red aprender las regiones propuestas.

YOLO [12] es una arquitectura que proporciona un nuevo enfoque en la detección de objetos, desarrollada en la universidad de Washington, la cual se basa en redes neuronales convolucionales simultáneas que predice múltiples cajas delimitadoras. La red neuronal base trabaja a 45 frames por segundo, esto significa que al procesar video en tiempo real obtiene menos de 25 milisegundos de latencia [16],

A diferencia de los algoritmos basados en regiones mencionados anteriormente, YOLO toma la detección de objetos como un problema único de regresión, en vez de examinar toda la imagen, examina partes de la imagen que tiene altas probabilidades de tener un objeto. Divide la imagen en una cuadrícula de SxS y si el centro de un objeto se encuentra dentro de un cuadrante, ese cuadrante es el responsable de detectar ese objeto. Una sola red convolucional predice simultáneamente múltiples cajas delimitadoras que enmarcan los objetos de la imagen y realiza un mapa de probabilidades por cada clase como lo muestra la fig. 1.

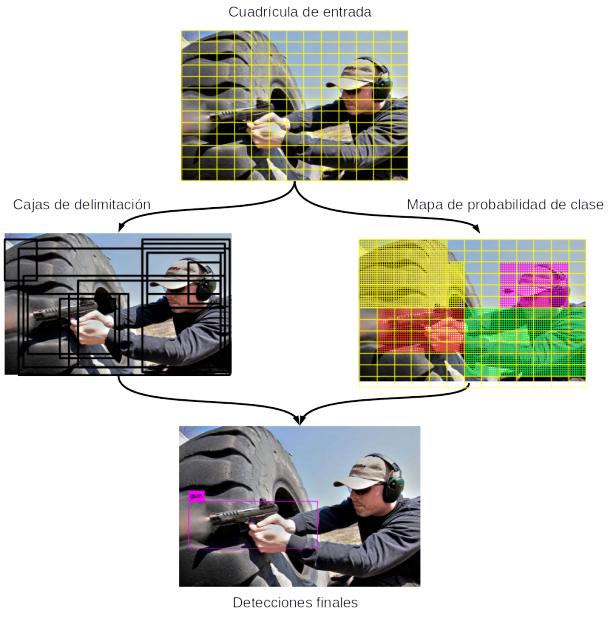

Figura 1: Esquema generar para YOLO.

Su arquitectura se inspiró en el modelo GoogLeNet [10] para la clasificación de imágenes. La red de YOLO tiene 53 capas convolucionales que son llamadas Darknet-53 [17], donde cada capa es idénticamente entrenada con los mismos valores y probadas en cuadriculas de 256 x 256. Se desempeña a la par de clasificadores de vanguardia, pero realiza menos operaciones de punto flotante, lo cual la hace más rápida [17].

III. Diseño y desarrollo del sistema

En este trabajo se propone un sistema para detectar armas de fuego o revólveres basado en las características de YOLO para detectar objetos.

III-A. Selección de imágenes

Consideramos dos conjuntos de imágenes (positivas y negativas), el primer conjunto de imágenes positivas, ver Figuras 2 (a)-(c), son aquellas en las que se presenta el objeto y el segundo conjunto de imágenes negativas, ver Figuras 2 (d)-(f), son aquellas en las que hay ausencia del objeto a identificar y se seleccionaron principalmente donde aparecían personas sosteniendo un objeto. Se seleccionaron un total de 600 imágenes y se dividieron como se muestra en el Cuadro I.

Cuadro I: Distribución de imágenes usadas en este trabajo

Tipo	Positivas	Negativas	Total
Entrenamiento	150	150	300
Pruebas	50	50	100
Validación	100	100	200

III-B. Recolección de imágenes

Las imágenes se obtuvieron de las siguientes fuentes:

III-B1. Internet Movie Firearms Database (IMFDb) [18]: Es una base de datos de imágenes de armas de fuego que aparecen en películas, series de televisión, videojuegos y series animadas. Aunque la base de datos contiene una gran cantidad de armas, este proyecto se centró únicamente en la detección de armas pequeñas como revólveres y pistolas.

III-C. Procesamiento de imágenes

Para cada una de las 300 imágenes utilizadas en el entrenamiento, se debe extraer las posiciones de los objetos, enmarcándolos dentro de un cuadrante interno en la imagen, que se representa mediante coordenadas en píxeles, realizando los siguientes pasos:

1. Seleccionar las imágenes de entrenamiento.
2. Realizar el etiquetado manual de cada imagen, usando la herramienta YOLO MARK [19].
3. Guardar las coordenadas del objeto en una archivo de texto, el archivo se deberá llamar igual que la imagen.

III-D. Entrenamiento

Con el entrenamiento se busca reducir la pérdida de precisión en la detección de objetos, requirió 2000 ciclos de entrenamiento y se utilizó una computadora con 8 GB de RAM, una tarjeta NVIDIA GeForce GTX 1050 TI, un procesador Intel Core I7, CUDA 10.1 y YOLO V3, a través de Darknet-53 [20].

El sistema propuesto basado en YOLO permite detectar, de manera concurrente, armas localizados en diferentes puntos de la imagen que está siendo procesada. Esto es debido a la característica de la red que cada caja delimitadora permite un análisis para definir si hay un arma en ella, es decir, se pueden detectar hasta 5 objetos (armas), una por cada una de las cajas delimitadoras.

IV. Validación y resultados

Las pruebas se realizaron con 100 muestras de imágenes (50 positivas y 50 negativas) diferentes de las utilizadas en el entrenamiento.

Para llevar a cabo la validación del sistema, de 100 imágenes positivas donde se encontraban 106 objetos se detectaron correctamente 95 objetos; por otra parte de 100 imágenes negativas se detectaron erróneamente 21 objetos. Se consideran los siguientes tipos de errores para la evaluación:

1. Error de tipo I. La predicción es positiva cuando el valor debe ser negativo, siendo 21 las ocurrencias de este tipo.
2. Error de tipo 2. La predicción es negativa cuando el valor debe ser positivo, siendo 11 las ocurrencias de este tipo.

Se seleccionaron las siguientes métricas de calidad [21]: *Exactitud (Ac)*, es la proporción del número total de predicciones que fueron correctas. *Tasa de verdaderos positivos (TPR)*, la proporción de que un caso positivo fueran correctamente identificadas. *Tasa de falsos positivos (FPR)*, es la proporción de que un caso negativo haya sido clasificado como positivo incorrectamente. *Tasa de verdadero negativos (TNR)*, la proporción de que los casos negativos fueron correctamente identificados. *Tasa de falsos negativos(FNR)*, la proporción de casos positivos que fueron incorrectamente clasificados como negativos. *Precisión (P)*, la proporción de casos positivos predichos que fueron correctos. El Cuadro II muestra los resultados obtenidos.

Cuadro II: Resultados de las métricas de calidad.

Ac	TPR	FPR	TNR	FNR	P
84.4660	0.89622	0.21	0.79	0.10377	0.818965

En el proceso de evaluación, se obtuvieron los siguientes resultados: *Verdadero positivo*, cuando el objeto es correctamente reconocido como lo muestra la Figura 3, sin embargo, es necesario implementar un supresor para encontrar los valores máximos y así evitar que un objeto sea reconocido más de dos veces. *Falso-Negativo*, cuando el objeto no es reconocido satisfactoriamente, como se muestra en la Figura 4 y *Falso-Positivo* cuando se identifican objetos incorrectos como correctos, como se puede ver en la Figura 5.

V. Conclusiones

De acuerdo a los resultados cuando se entrena el detector YOLO en varios escenarios y condiciones es robusto, respecto a las imágenes el tiempo de detección es de 47,5 milisegundos en promedio.

En la Figura 6 se muestra la comparación de exactitud de este trabajo con otros similares. En [6] y [7], donde no se aplica entrenamiento, utilizan segmentación basada en colores y su cantidad de imágenes es menor (88 imágenes para el primer caso y 25 imágenes para el segundo), este trabajo obtiene una exactitud similar a [6] y 4.2 % menor respecto a [7], sin embargo, el tiempo de detección de ambos es proporcional al numero de objetos que se encuentran en la imagen.

(a) (b) (c)

(d) (e) (f)

Figura 2: Muestra de imágenes

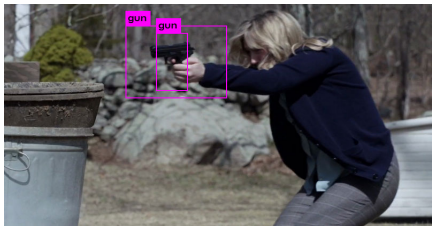

Figura 3: Positivo verdadero.

Figura 4: Falso Negativo.

Figura 5: Falso positivo.

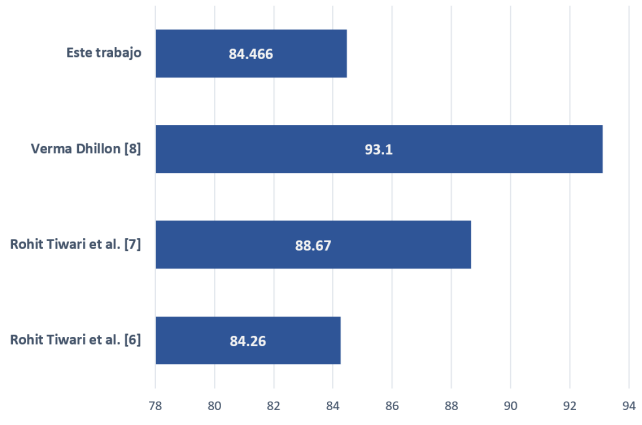

Figura 6: Comparación de la exactitud obtenida con trabajos similares.

La precisión de [8] es mayor en 8.6 % donde se aplica una arquitectura VGG-16 basada en CNN como un extractor de características. Aunque las imágenes positivas son similares, las imágenes negativas de [8] se eligieron de forma aleatoria, mientras que nuestras imágenes se centraron principalmente en personas que tenían diversos objetos en las manos.

Se observó que al entrenar la red neuronal con una muestra pequeña de imágenes ocasionan que frecuentemente se detecten como positivos algunos objetos negativos, por lo que es necesario volver a realizar el entrenamiento con una muestra mas grande de imágenes para mejorar la precisión.

Referencias

[1] INEGI, "Encuesta nacional de victimización y percepción sobre seguridad pública principales resultados(envipe) 2018," 2018.

[2] S. Nercessian, K. Panetta, and S. Agaian, "Automatic detection of potential threat objects in x-ray luggage scan images," 06 2008, pp. 504 – 509.

[3] Maneesha Singh and Sameer Singh, "Image segmentation optimisation for x-ray images of airline luggage," in *Proceedings of the 2004 IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety, 2004. CIHSPS 2004.*, July 2004, pp. 10–17.

[4] A. D. Lopez, E. S. Kollialil, and K. G. Gopan, "Adaptive neuro-fuzzy classifier for weapon detection in x-ray images of luggage using zernike moments and shape context descriptor," in *2013 Third International Conference on Advances in Computing and Communications*, Aug 2013, pp. 46–49.

[5] T. Xu and Q. M. Jonathan Wu, "Multisensor concealed weapon detection using the image fusion approach," in *6th International Conference on Imaging for Crime Prevention and Detection (ICDP-15)*, July 2015, pp. 1–7.

[6] G. Verma and R. Tiwari, "A computer vision based framework for visual gun detection using harris interest point detector," vol. 54, 08 2015.

[7] G. Verma, "A computer vision based framework for visual gun detection using surf," 01 2015.

[8] G. Verma and A. Dhillon, "A handheld gun detection using faster r-cnn deep learning," 11 2017, pp. 84–88.

[9] A. Vedaldi and K. Lenc, "Matconvnet - convolutional neural networks for MATLAB," *CoRR*, vol. abs/1412.4564, 2014. [Online]. Available: http://arxiv.org/abs/1412.4564

[10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014. [Online]. Available: http://arxiv.org/abs/1409.4842
[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385
[12] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," *CoRR*, vol. abs/1612.08242, 2016. [Online]. Available: http://arxiv.org/abs/1612.08242
[13] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *CoRR*, vol. abs/1311.2524, 2013. [Online]. Available: http://arxiv.org/abs/1311.2524
[14] R. Girshick, "Fast rcnn," *IEEE International Conference onComputer Vision (ICCV)*, 2015.
[15] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, June 2017.
[16] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, 2015. [Online]. Available: http://arxiv.org/abs/1506.02640
[17] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *CoRR*, vol. abs/1804.02767, 2018. [Online]. Available: http://arxiv.org/abs/1804.02767
[18] (2019) Internet movie firearms database imfdb. [Online]. Available: http://www.imfdb.org/wiki/Main_Page
[19] AlexeyAB, "Yolo mark," 2019. [Online]. Available: https://github.com/AlexeyAB/Yolo_mark
[20] ——, "Darknet-53," 2019. [Online]. Available: https://github.com/AlexeyAB/darknet
[21] EcuRed, "Matrices de confusión," 2019. [Online]. Available: